ASYNCHRONOUS TRANSFER MODE NETWORKS

Asynchronous transfer mode, or ATM, is a network transfer technique capable of supporting a wide variety of multimedia applications with diverse service and performance requirements. It supports traffic bandwidths ranging from a few kilobits per second (e.g., a text terminal) to several hundred megabits per second (e.g., high-definition video) and traffic types ranging from continuous, fixed-rate traffic (e.g., traditional telephony and file transfer) to highly bursty traffic (e.g., interactive data and video). Because of its support for such a wide range of traffic, ATM was designated by the telecommunication standardization sector of the International Telecommunications Union (ITU-T, formerly CCITT) as the multiplexing and switching technique for Broadband, or high-speed, ISDN (B-ISDN) (1).

ATM is a form of packet-switching technology. That is, ATM networks transmit their information in small, fixed-length packets called cells, each of which contains 48 octets (or bytes) of data and 5 octets of header information. The small, fixed cell size was chosen to facilitate the rapid processing of packets in hardware and to minimize the amount of time required to fill a single packet. This is particularly important for real-time applications such as voice and video that require short packetization delays.

ATM is also connection-oriented. In other words, a virtual circuit must be established before a call can take place, where a call is defined as the transfer of information between two or more endpoints. The establishment of a virtual circuit entails the initiation of a signaling process, during which a route is selected according to the call's quality of service requirements, connection identifiers at each switch on the route are established, and network resources such as bandwidth and buffer space may be reserved for the connection.

Another important characteristic of ATM is that its network functions are typically implemented in hardware. With the introduction of high-speed fiber optic transmission lines, the communication bottleneck has shifted from the communication links to the processing at switching nodes and at terminal equipment. Hardware implementation is necessary to overcome this bottleneck because it minimizes the cell-processing overhead, thereby allowing the network to match link rates on the order of gigabits per second.

Finally, as its name indicates, ATM is asynchronous. Time is slotted into cell-sized intervals, and slots are assigned to calls in an asynchronous, demand-based manner. Because slots are allocated to calls on demand, ATM can easily accommodate traffic whose bit rate fluctuates over time. Moreover, in ATM, no bandwidth is consumed unless information is actually transmitted. ATM also gains bandwidth efficiency by being able to multiplex bursty traffic sources statistically. Because bursty traffic does not require continuous allocation of the bandwidth at its peak rate, statistical multiplexing allows a large number of bursty sources to share the network's bandwidth.

Since its birth in the mid-1980s, ATM has been fortified by a number of robust standards and realized by a significant number of network equipment manufacturers. International standards-making bodies such as the ITU and independent consortia like the ATM Forum have developed a significant body of standards and implementation agreements for ATM (1,4). As networks and network services continue to evolve toward greater speeds and diversities, ATM will undoubtedly continue to proliferate.

ATM STANDARDS

The telecommunication standardization sector of the ITU, the international standards agency commissioned by the United Nations for the global standardization of telecommunications, has developed a number of standards for ATM networks. Other standards bodies and consortia (e.g., the ATM Forum, ANSI) have also contributed to the development of ATM standards. This section presents an overview of the standards, with particular emphasis on the protocol reference model used by ATM (2).

Protocol Reference Model

The B-ISDN protocol reference model, defined in ITU-T recommendation I.321, is shown in Fig. 1 (1). The purpose of the protocol reference model is to clarify the functions that ATM networks perform by grouping them into a set of interrelated, function-specific layers and planes. The reference model consists of a user plane, a control plane, and a management plane. Within the user and control planes is a hierarchical set of layers. The user plane defines a set of functions for the transfer of user information between communication endpoints; the control plane defines control functions such as call establishment, call maintenance, and call release; and the management plane defines the operations necessary to control information flow between planes and layers and to maintain accurate and fault-tolerant network operation.

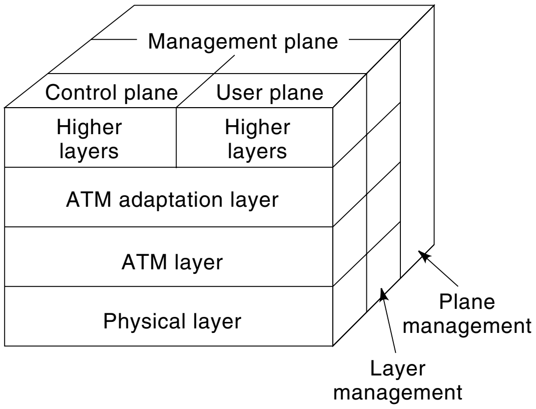

Figure 1. Protocol reference model for ATM.

Layer management	Higher layer functions	Higher layers	
	Convergence	CS	AAL
	Segmentation and reassembly	SAR	
	Generic flow control Cell header generation/extraction Cell VPI/VCI translation Cell multiplex and demultiplex	ATM	
	Cell rate decoupling Header error control (HEC) Cell delineation Transmission frame adaptation Transmission frame generation/recovery	TC	Physical layer
	Bit timing Physical medium	PM	

Figure 2. Functions of each layer in the protocol reference model.

Within the user and control planes, there are three layers: the physical layer, the ATM layer, and the ATM adaptation layer (AAL). Figure 2 summarizes the functions of each layer (1). The physical layer performs primarily bit-level functions, the ATM layer is primarily responsible for the switching of ATM cells, and the ATM adaptation layer is responsible for the conversion of higher-layer protocol frames into ATM cells. The functions that the physical, ATM, and adaptation layers perform are described in more detail next.

Physical Layer

The physical layer is divided into two sublayers: the physical medium sublayer and the transmission convergence sublayer (1).

Physical Medium Sublayer. The physical medium (PM) sublayer performs medium-dependent functions. For example, it provides bit transmission capabilities including bit alignment, line coding and electrical/optical conversion. The PM sublayer is also responsible for bit timing (i.e., the insertion and extraction of bit timing information). The PM sublayer currently supports two types of interface: optical and electrical.

Transmission Convergence Sublayer. Above the physical medium sublayer is the transmission convergence (TC) sublayer, which is primarily responsible for the framing of data transported over the physical medium. The ITU-T recommendation specifies two options for TC sublayer transmission frame structure: cell-based and synchronous digital hierarchy (SDH). In the cell-based case, cells are transported continuously without any regular frame structure. Under SDH, cells are carried in a special frame structure based on the North American SONET (synchronous optical network) protocol (3). Regardless of which transmission frame structure is used, the TC sublayer is responsible for the following four functions: cell rate decoupling, header error control, cell delineation, and transmission frame adaptation. Cell rate decoupling is the insertion of idle cells at the sending side to adapt the ATM cell

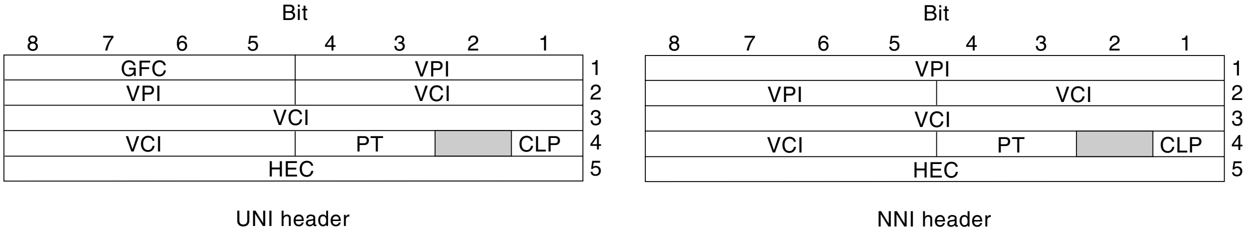

Figure 3. ATM cell header structure.

stream's rate to the rate of the transmission path. Header error control is the insertion of an 8-bit CRC in the ATM cell header to protect the contents of the ATM cell header. Cell delineation is the detection of cell boundaries. Transmission frame adaptation is the encapsulation of departing cells into an appropriate framing structure (either cell-based or SDH-based).

ATM Layer

The ATM layer lies atop the physical layer and specifies the functions required for the switching and flow control of ATM cells (1).

There are two interfaces in an ATM network: the user-network interface (UNI) between the ATM endpoint and the ATM switch, and the network-network interface (NNI) between two ATM switches. Although a 48-octet cell payload is used at both interfaces, the 5-octet cell header differs slightly at these interfaces. Figure 3 shows the cell header structures used at the UNI and NNI (1). At the UNI, the header contains a 4-bit generic flow control (GFC) field, a 24-bit label field containing virtual path identifier (VPI) and virtual channel identifier (VCI) subfields (8 bits for the VPI and 16 bits for the VCI), a 2-bit payload type (PT) field, a 1-bit cell loss priority (CLP) field, and an 8-bit header error check (HEC) field. The cell header for an NNI cell is identical to that for the UNI cell, except that it lacks the GFC field; these four bits are used for an additional 4 VPI bits in the NNI cell header.

The VCI and VPI fields are identifier values for virtual channel (VC) and virtual path (VP), respectively. A virtual channel connects two ATM communication endpoints. A virtual path connects two ATM devices, which can be switches or endpoints, and several virtual channels may be multiplexed onto the same virtual path. The 2-bit PT field identifies whether the cell payload contains data or control information. The CLP bit is used by the user for explicit indication of cell loss priority. If the value of the CLP is 1, then the cell is subject to discarding in case of congestion. The HEC field is an 8-bit CRC that protects the contents of the cell header. The GFC field, which appears only at the UNI, is used to assist the customer premises network in controlling the traffic flow. At the time of writing, the exact procedures for use of this field have not been agreed upon.

ATM Layer Functions

The primary function of the ATM layer is VPI/VCI translation. As ATM cells arrive at ATM switches, the VPI and VCI values contained in their headers are examined by the switch to determine which outport port should be used to forward the cell. In the process, the switch translates the cell's original VPI and VCI values into new outgoing VPI and VCI values, which are used in turn by the next ATM switch to send the cell toward its intended destination. The table used to perform this translation is initialized during the establishment of the call.

An ATM switch may either be a VP switch, in which case it translates only the VPI values contained in cell headers, or it may be a VP/VC switch, in which case it translates the incoming VPI/VCI value into an outgoing VPI/VCI pair. Because VPI and VCI values do not represent a unique end-to-end virtual connection, they can be reused at different switches through the network. This is important because the VPI and VCI fields are limited in length and would be quickly exhausted if they were used simply as destination addresses.

The ATM layer supports two types of virtual connections: switched virtual connections (SVC) and permanent, or semipermanent, virtual connections (PVC). Switched virtual connections are established and torn down dynamically by an ATM signaling procedure. That is, they exist only for the duration of a single call. Permanent virtual connections, on the other hand, are established by network administrators and continue to exist as long as the administrator leaves them up, even if they are not used to transmit data.

Other important functions of the ATM layer include cell multiplexing and demultiplexing, cell header creation and extraction, and generic flow control. Cell multiplexing is the merging of cells from several calls onto a single transmission path, cell header creation is the attachment of a 5-octet cell header to each 48-octet block of user payload, and generic flow control is used at the UNI to prevent short-term overload conditions from occurring within the network.

ATM Layer Service Categories

The ATM Forum and ITU-T have defined several distinct service categories at the ATM layer (1,4). The categories defined by the ATM Forum include constant bit rate (CBR), real-time variable bit rate (VBR-rt), non-real-time variable bit rate (VBR-nrt), available bit rate (ABR), and unspecified bit rate (UBR). ITU-T defines four service categories, namely, deterministic bit rate (DBR), statistical bit rate (SBR), available bit rate (ABR), and ATM block transfer (ABT). The first of the three ITU-T service categories correspond roughly to the ATM Forum's CBR, VBR, and ABR classifications, respectively. The fourth service category, ABT, is solely defined by ITU-T and is intended for bursty data applications. The UBR category defined by the ATM Forum is for calls that request no quality of service guarantees at all. Figure 4 lists the ATM service categories, their quality of service (QoS) parameters,

ITU-T service categories →	DBR	SBR	ABT	ABR	
ATM forum service categories →	CBR	VBR-rt	VBR-nrt	ABR	UBR
Cell loss rate	Specified				Unspecified
Cell transfer delay	Specified			Unspecified	
Cell delay variation	Specified		Unspecified		
Traffic descriptors (contract)	PCR/CDVT	PCR/CDVT SCR/BT		PCR/CDVT MCR/ACR	PCR/CDVT

PCR = Peak Cell Rate; SCR = Sustained Cell Rate; CDVT = Cell Delay Variation Tolerance; BT = Burst Tolerance; MCR = Minimum Cell Rate; ACR = Allowed Cell Rate.

Figure 4. ATM layer service categories.

and the traffic descriptors required by the service category during call establishment (1,4).

The constant bit rate (or deterministic bit rate) service category provides a very strict QoS guarantee. It is targeted at real-time applications, such as voice and raw video, which mandate severe restrictions on delay, delay variance (jitter), and cell loss rate. The only traffic descriptors required by the CBR service are the peak cell rate and the cell delay variation tolerance. A fixed amount of bandwidth, determined primarily by the call's peak cell rate, is reserved for each CBR connection.

The real-time variable bit rate (or statistical bit rate) service category is intended for real-time bursty applications (e.g., compressed video), which also require strict QoS guarantees. The primary difference between CBR and VBR-rt is in the traffic descriptors they use. The VBR-rt service requires the specification of the sustained (or average) cell rate and burst tolerance (i.e., burst length) in addition to the peak cell rate and the cell delay variation tolerance. The ATM Forum also defines a VBR-nrt service category, in which cell delay variance is not guaranteed.

The available bit rate service category is defined to exploit the network's unused bandwidth. It is intended for non-real-time data applications in which the source is amenable to enforced adjustment of its transmission rate. A minimum cell rate is reserved for the ABR connection and therefore guaranteed by the network. When the network has unused bandwidth, ABR sources are allowed to increase their cell rates up to an allowed cell rate (ACR), a value that is periodically updated by the ABR flow control mechanism (to be described in the section entitled "ATM Traffic Control"). The value of ACR always falls between the minimum and the peak cell rate for the connection and is determined by the network.

The ATM Forum defines another service category for non-real-time applications called the unspecified bit rate (UBR) service category. The UBR service is entirely best effort; the call is provided with no QoS guarantees. The ITU-T also defines an additional service category for non-real-time data applications. The ATM block transfer service category is intended for the transmission of short bursts, or blocks, of data. Before transmitting a block, the source requests a reservation of bandwidth from the network. If the ABT service is being used with the immediate transmission option (ABT/IT), the block of data is sent at the same time as the reservation request. If bandwidth is not available for transporting the block, then it is simply discarded, and the source must retransmit it. In the ABT service with delayed transmission (ABT/DT), the source waits for a confirmation from the network that enough bandwidth is available before transmitting the block of data. In both cases, the network temporarily reserves bandwidth according to the peak cell rate for each block. Immediately after transporting the block, the network releases the reserved bandwidth.

ATM Adaptation Layer

The ATM adaptation layer, which resides atop the ATM layer, is responsible for mapping the requirements of higher layer protocols onto the ATM network (1). It operates in ATM devices at the edge of the ATM network and is totally absent in ATM switches. The adaptation layer is divided into two sublayers: the convergence sublayer (CS), which performs error detection and handling, timing, and clock recovery; and the segmentation and reassembly (SAR) sublayer, which performs segmentation of convergence sublayer protocol data units (PDUs) into ATM cell-sized SAR sublayer service data units (SDUs) and vice versa.

In order to support different service requirements, the ITU-T has proposed four AAL-specific service classes. Figure 5 depicts the four service classes defined in recommendation I.362 (1). Note that even though these AAL service classes are similar in many ways to the ATM layer service categories defined in the previous section, they are not the same; each exists at a different layer of the protocol reference model, and each requires a different set of functions.

AAL service class A corresponds to constant bit rate services with a timing relation required between source and destination. The connection mode is connection-oriented. The CBR audio and video belong to this class. Class B corresponds to variable bit rate (VBR) services. This class also requires timing between source and destination, and its mode is connection-oriented. The VBR audio and video are examples of class B services. Class C also corresponds to VBR connection-oriented services, but the timing between source and destination needs not be related. Class C includes connection-oriented data transfer such as X.25, signaling, and future high-speed data services. Class D corresponds to connectionless services. Connectionless data services such as those supported by LANs and MANs are examples of class D services.

Four AAL types (Types 1, 2, 3/4, and 5), each with a unique SAR sublayer and CS sublayer, are defined to support the four service classes. AAL Type 1 supports constant bit rate services (class A), and AAL Type 2 supports variable bit rate services with a timing relation between source and desti-

	Class A	Class B	Class C	Class D
Timing relation between source and destination	Required		Not required	
Bit rate	Constant	Variable		
Connection mode	Connection oriented		Connectionless	

Figure 5. Service classification for AAL.

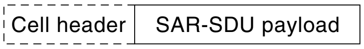

Figure 6. SAR-SDU format for AAL Type 5.

nation (class B). AAL Type 3/4 was originally specified as two different AAL types (Type 3 and Type 4), but because of their inherent similarities, they were eventually merged to support both class C and class D services. AAL Type 5 also supports class C and class D services.

AAL Type 5. Currently, the most widely used adaptation layer is AAL Type 5. AAL Type 5 supports connection-oriented and connectionless services in which there is no timing relation between source and destination (classes C and D). Its functionality was intentionally made simple in order to support high-speed data transfer. AAL Type 5 assumes that the layers above the ATM adaptation layer can perform error recovery, retransmission, and sequence numbering when required, and thus, it does not provide these functions. Therefore, only nonassured operation is provided; lost or corrupted AAL Type 5 packets will not be corrected by retransmission.

Figure 6 depicts the SAR-SDU format for AAL Type 5 (5,6). The SAR sublayer of AAL Type 5 performs segmentation of a CS-PDU into a size suitable for the SAR-SDU payload. Unlike other AAL types, Type 5 devotes the entire 48-octet payload of the ATM cell to the SAR-SDU; there is no overhead. An AAL specific flag (end-of-frame) in the ATM PT field of the cell header is set when the last cell of a CS-PDU is sent. The reassembly of CS-PDU frames at the destination is controlled by using this flag.

Figure 7 depicts the CS-PDU format for AAL Type 5 (5,6). It contains the user data payload, along with any necessary padding bits (PAD) and a CS-PDU trailer, which are added by the CS sublayer when it receives the user information from the higher layer. The CS-PDU is padded using 0 to 47 bytes of PAD field to make the length of the CS-PDU an integral multiple of 48 bytes (the size of the SAR-SDU payload). At the receiving end, a reassembled PDU is passed to the CS sublayer from the SAR sublayer, and CRC values are then calculated and compared. If there is no error, the PAD field is removed by using the value of length field (LF) in the CS-PDU trailer, and user data is passed to the higher layer. If an error is detected, the erroneous information is either delivered to the user or discarded according to the user's choice. The use of the CF field is for further study.

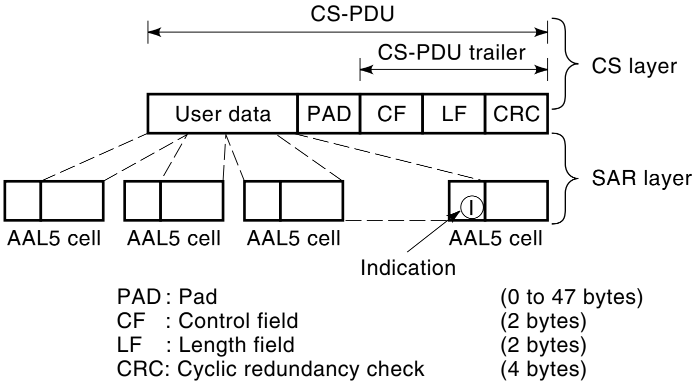
CS-PDU
CS-PDU trailer
CS layer
User data
PAD
CF
LF
CRC
SAR layer
AAL5 cell
AAL5 cell
AAL5 cell
AAL5 cell
1
Indication
PAD : Pad (0 to 47 bytes)
CF : Control field (2 bytes)
LF : Length field (2 bytes)
CRC: Cyclic redundancy check (4 bytes)

Figure 7. CS-PDU format, segmentation and reassembly of AAL Type 5.

AAL Type 1. AAL Type 1 supports constant bit rate services with a fixed timing relation between source and destination users (class A). At the SAR sublayer, it defines a 48-octet service data unit (SDU), which contains 47 octets of user payload, 4 bits for a sequence number, and a 4-bit CRC value to detect errors in the sequence number field. AAL Type 1 performs the following services at the CS sublayer: forward error correction to ensure high quality of audio and video applications, clock recovery by monitoring the buffer filling, explicit time indication by inserting a time stamp in the CS-PDU, and handling of lost and misinserted cells that are recognized by the SAR. At the time of writing, the CS-PDU format has not been decided.

AAL Type 2. AAL Type 2 supports variable bit rate services with a timing relation between source and destination (class B). AAL Type 2 is nearly identical to AAL Type 1, except that it transfers service data units at a variable bit rate, not at a constant bit rate. Furthermore, AAL Type 2 accepts variable length CS-PDUs, and thus, there may exist some SAR-SDUs that are not completely filled with user data. The CS sublayer for AAL Type 2 performs the following functions: forward error correction for audio and video services, clock recovery by inserting a time stamp in the CS-PDU, and handling of lost and misinserted cells. At the time of writing, both the SAR-SDU and CS-PDU formats for AAL Type 2 are still under discussion.

AAL Type 3/4. AAL Type 3/4 mainly supports services that require no timing relation between the source and destination (classes C and D). At the SAR sublayer, it defines a 48-octet service data unit, with 44 octets of user payload; a 2-bit payload type field to indicate whether the SDU is at the beginning, middle, or end of a CS-PDU; a 4-bit cell sequence number; a 10-bit multiplexing identifier that allows several CS-PDUs to be multiplexed over a single VC; a 6-bit cell payload length indicator; and a 10-bit CRC code that covers the payload. The CS-PDU format allows for up to 65535 octets of user payload and contains a header and trailer to delineate the PDU.

The functions that AAL Type 3/4 performs include segmentation and reassembly of variable-length user data and error handling. It supports message mode (for framed data transfer) as well as streaming mode (for streamed data transfer). Because Type 3/4 is mainly intended for data services, it provides a retransmission mechanism if necessary.

ATM Signaling

ATM follows the principle of out-of-band signaling that was established for N-ISDN. In other words, signaling and data channels are separate. The main purposes of signaling are (1) to establish, maintain, and release ATM virtual connections and (2) to negotiate (or renegotiate) the traffic parameters of new (or existing) connections (7). The ATM signaling standards support the creation of point-to-point as well as multicast connections. Typically, certain VCI and VPI values are reserved by ATM networks for signaling messages. If additional signaling VCs are required, they may be established through the process of metasignaling.

ATM TRAFFIC CONTROL

The control of ATM traffic is complicated as a result of ATM's high-link speed and small cell size, the diverse service requirements of ATM applications, and the diverse characteristics of ATM traffic. Furthermore, the configuration and size of the ATM environment, either local or wide area, has a significant impact on the choice of traffic control mechanisms.

The factor that most complicates traffic control in ATM is its high-link speed. Typical ATM link speeds are 155.52 Mbit/s and 622.08 Mbit/s. At these high-link speeds, 53-byte ATM cells must be switched at rates greater than one cell per 2.726 μs or 0.682 μs, respectively. It is apparent that the cell processing required by traffic control must perform at speeds comparable to these cell-switching rates. Thus, traffic control should be simple and efficient, without excessive software processing.

Such high speeds render many traditional traffic control mechanisms inadequate for use in ATM because of their reactive nature. Traditional reactive traffic control mechanisms attempt to control network congestion by responding to it after it occurs and usually involves sending feedback to the source in the form of a choke packet. However, a large bandwidth-delay product (i.e., the amount of traffic that can be sent in a single propagation delay time) renders many reactive control schemes ineffective in high-speed networks. When a node receives feedback, it may have already transmitted a large amount of data. Consider a cross-continental 622 Mbit/s connection with a propagation delay of 20 ms (propagation-bandwidth product of 12.4 Mbit). If a node at one end of the connection experiences congestion and attempts to throttle the source at the other end by sending it a feedback packet, the source will already have transmitted over 12 Mb of information before feedback arrives. This example illustrates the ineffectiveness of traditional reactive traffic control mechanisms in high-speed networks and argues for novel mechanisms that take into account high propagation-bandwidth products.

Not only is traffic control complicated by high speeds, but it also is made more difficult by the diverse QoS requirements of ATM applications. For example, many applications have strict delay requirements and must be delivered within a specified amount of time. Other applications have strict loss requirements and must be delivered reliably without an inordinate amount of loss. Traffic controls must address the diverse requirements of such applications.

Another factor complicating traffic control in ATM networks is the diversity of ATM traffic characteristics. In ATM networks, continuous bit rate traffic is accompanied by bursty traffic. Bursty traffic generates cells at a peak rate for a very short period of time and then immediately becomes less active, generating fewer cells. To improve the efficiency of ATM network utilization, bursty calls should be allocated an amount of bandwidth that is less than their peak rate. This allows the network to multiplex more calls by taking advantage of the small probability that a large number of bursty calls will be simultaneously active. This type of multiplexing is referred to as statistical multiplexing. The problem then becomes one of determining how best to multiplex bursty calls statistically such that the number of cells dropped as a result of excessive burstiness is balanced with the number of bursty traffic streams allowed. Addressing the unique demands of bursty traffic is an important function of ATM traffic control.

For these reasons, many traffic control mechanisms developed for existing networks may not be applicable to ATM networks, and therefore novel forms of traffic control are required (8,9). One such class of novel mechanisms that work well in high-speed networks falls under the heading of preventive control mechanisms. Preventive control attempts to manage congestion by preventing it before it occurs. Preventive traffic control is targeted primarily at real-time traffic. Another class of traffic control mechanisms has been targeted toward non-real-time data traffic and relies on novel reactive feedback mechanisms.

Preventive Traffic Control

Preventive control for ATM has two major components: call admission control and usage parameter control (8). Admission control determines whether to accept or reject a new call at the time of call set-up. This decision is based on the traffic characteristics of the new call and the current network load. Usage parameter control enforces the traffic parameters of the call after it has been accepted into the network. This enforcement is necessary to ensure that the call's actual traffic flow conforms with that reported during call admission.

Before describing call admission and usage parameter control in more detail, it is important to first discuss the nature of multimedia traffic. Most ATM traffic belongs to one of two general classes of traffic: continuous traffic and bursty traffic. Sources of continuous traffic (e.g., constant bit rate video, voice without silence detection) are easily handled because their resource utilization is predictable and they can be deterministically multiplexed. However, bursty traffic (e.g., voice with silence detection, variable bit rate video) is characterized by its unpredictability, and this kind of traffic complicates preventive traffic control.

Burstiness is a parameter describing how densely or sparsely cell arrivals occur. There are a number of ways to express traffic burstiness, the most typical of which are the ratio of peak bit rate to average bit rate and the average burst length. Several other measures of burstiness have also been proposed (8). It is well known that burstiness plays a critical role in determining network performance, and thus, it is critical for traffic control mechanisms to reduce the negative impact of bursty traffic.

Call Admission Control. Call admission control is the process by which the network decides whether to accept or reject a new call. When a new call requests access to the network, it provides a set of traffic descriptors (e.g., peak rate, average rate, average burst length) and a set of quality of service requirements (e.g., acceptable cell loss rate, acceptable cell delay variance, acceptable delay). The network then determines, through signaling, if it has enough resources (e.g., bandwidth, buffer space) to support the new call's requirements. If it does, the call is immediately accepted and allowed to transmit data into the network. Otherwise it is rejected. Call admission control prevents network congestion by limiting the number of active connections in the network to a level where the network resources are adequate to maintain quality of service guarantees.

One of the most common ways for an ATM network to make a call admission decision is to use the call's traffic descriptors and quality of service requirements to predict the "equivalent bandwidth" required by the call. The equivalent bandwidth determines how many resources need to be reserved by the network to support the new call at its requested quality of service. For continuous, constant bit rate calls, determining the equivalent bandwidth is simple. It is merely equal to the peak bit rate of the call. For bursty connections, however, the process of determining the equivalent bandwidth should take into account such factors as a call's burstiness ratio (the ratio of peak bit rate to average bit rate), burst length, and burst interarrival time. The equivalent bandwidth for bursty connections must be chosen carefully to ameliorate congestion and cell loss while maximizing the number of connections that can be statistically multiplexed.

Usage Parameter Control. Call admission control is responsible for admitting or rejecting new calls. However, call admission by itself is ineffective if the call does not transmit data according to the traffic parameters it provided. Users may intentionally or accidentally exceed the traffic parameters declared during call admission, thereby overloading the network. In order to prevent the network users from violating their traffic contracts and causing the network to enter a congested state, each call's traffic flow is monitored and, if necessary, restricted. This is the purpose of usage parameter control. (Usage parameter control is also commonly referred to as policing, bandwidth enforcement, or flow enforcement.)

To monitor a call's traffic efficiently, the usage parameter control function must be located as close as possible to the actual source of the traffic. An ideal usage parameter control mechanism should have the ability to detect parameter-violating cells, appear transparent to connections respecting their admission parameters, and rapidly respond to parameter violations. It should also be simple, fast, and cost effective to implement in hardware. To meet these requirements, several mechanisms have been proposed and implemented (8).

The leaky bucket mechanism (originally proposed in Ref. 10) is a typical usage parameter control mechanism used for ATM networks. It can simultaneously enforce the average bandwidth and the burst factor of a traffic source. One possible implementation of the leaky bucket mechanism is to control the traffic flow by means of tokens. A conceptual model for the leaky bucket mechanism is illustrated in Fig. 5.

In Fig. 8, an arriving cell first enters a queue. If the queue is full, cells are simply discarded. To enter the network, a cell must first obtain a token from the token pool; if there is no token, a cell must wait in the queue until a new token is generated. Tokens are generated at a fixed rate corresponding to the average bit rate declared during call admission. If the number of tokens in the token pool exceeds some predefined threshold value, token generation stops. This threshold value corresponds to the burstiness of the transmission declared at call admission time; for larger threshold values, a greater degree of burstiness is allowed. This method enforces the average input rate while allowing for a certain degree of burstiness.

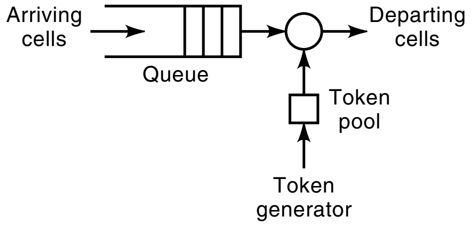

Figure 8. Leaky bucket mechanism.

One disadvantage of the leaky bucket mechanism is that the bandwidth enforcement introduced by the token pool is in effect even when the network load is light and there is no need for enforcement. Another disadvantage of the leaky bucket mechanism is that it may mistake nonviolating cells for violating cells. When traffic is bursty, a large number of cells may be generated in a short period of time, while conforming to the traffic parameters claimed at the time of call admission. In such situations, none of these cells should be considered violating cells. Yet in actual practice, leaky bucket may erroneously identify such cells as violations of admission parameters. A virtual leaky bucket mechanism (also referred to as a marking method) alleviates these disadvantages (11). In this mechanism, violating cells, rather than being discarded or buffered, are permitted to enter the network at a lower priority (CLP = 1). These violating cells are discarded only when they arrive at a congested node. If there are no congested nodes along the routes to their destinations, the violating cells are transmitted without being discarded. The virtual leaky bucket mechanism can easily be implemented using the leaky bucket method described earlier. When the queue length exceeds a threshold, cells are marked as "droppable" instead of being discarded. The virtual leaky bucket method not only allows the user to take advantage of a light network load but also allows a larger margin of error in determining the token pool parameters.

Reactive Traffic Control

Preventive control is appropriate for most types of ATM traffic. However, there are cases where reactive control is beneficial. For instance, reactive control is useful for service classes like ABR, which allow sources to use bandwidth not being used by calls in other service classes. Such a service would be impossible with preventive control because the amount of unused bandwidth in the network changes dynamically, and the sources can only be made aware of the amount through reactive feedback.

There are two major classes of reactive traffic control mechanisms: rate-based and credit-based (12,13). Most rate-based traffic control mechanisms establish a closed feedback loop in which the source periodically transmits special control cells, called resource management cells, to the destination (or destinations). The destination closes the feedback loop by returning the resource management cells to the source. As the feedback cells traverse the network, the intermediate switches examine their current congestion state and mark the feedback cells accordingly. When the source receives a returning feedback cell, it adjusts its rate, either by decreasing it in the case of network congestion or increasing it in the case of network underuse. An example of a rate-based ABR algorithm is the Enhanced Proportional Rate Control Algorithm (EPRCA), which was proposed, developed, and tested through the course of ATM Forum activities (12).

Credit-based mechanisms use link-by-link traffic control to eliminate loss and optimize use. Intermediate switches exchange resource management cells that contain "credits," which reflect the amount of buffer space available at the next downstream switch. A source cannot transmit a new data cell unless it has received at least one credit from its downstream neighbor. An example of a credit-based mechanism is the Quantum Flow Control (QFC) algorithm, developed by a consortium of reseachers and ATM equipment manufacturers (13).

HARDWARE SWITCH ARCHITECTURES FOR ATM NETWORKS

In ATM networks, information is segmented into fixed-length cells, and cells are asynchronously transmitted through the network. To match the transmission speed of the network links and to minimize the protocol processing overhead, ATM performs the switching of cells in hardware-switching fabrics, unlike traditional packet switching networks, where switching is largely performed in software.

A large number of designs has been proposed and implemented for ATM switches (14). Although many differences exist, ATM switch architectures can be broadly classified into two categories: asynchronous time division (ATD) and space-division architectures.

Asynchronous Time Division Switches

The ATD, or single path, architectures provide a single, multiplexed path through the ATM switch for all cells. Typically a bus or ring is used. Figure 9 shows the basic structure of the ATM switch proposed in (15). In Fig. 6, four input ports are connected to four output ports by a time-division multiplexing (TDM) bus. Each input port is allocated a fixed time slot on the TDM bus, and the bus is designated to operate at a speed equal to the sum of the incoming bit rates at all input ports. The TDM slot sizes are fixed and equal in length to the time it takes to transmit one ATM cell. Thus, during one TDM cycle, the four input ports can transfer four ATM cells to four output ports.

In ATD switches, the maximum throughput is determined by a single, multiplexed path. Switches with N input ports and N output ports must run at a rate N times faster than the transmission links. Therefore, the total throughput of ATD ATM switches is bounded by the current capabilities of device logic technology. Commercial examples of ATD switches are the Fore Systems ASX switch and Digital's VNswitch.

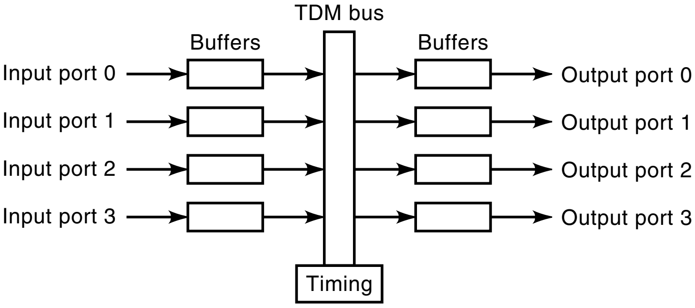

Figure 9. A 4 × 4 asynchronous time division switch.

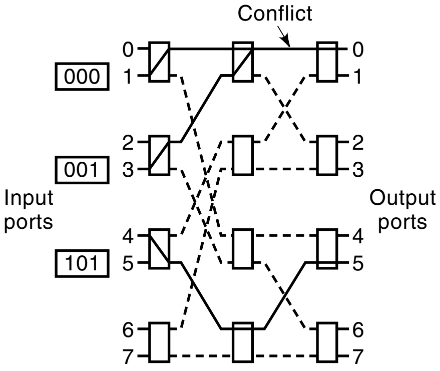

Figure 10. A 8 × 8 Banyan switch with binary switching elements.

Space-Division Switches

To eliminate the single-path limitation and increase total throughput, space-division ATM switches implement multiple paths through switching fabrics. Most space-division switches are based on multistage interconnection networks, where small switching elements (usually 2 × 2 cross-point switches) are organized into stages and provide multiple paths through a switching fabric. Rather than being multiplexed onto a single path, ATM cells are space-switched through the fabric. Three typical types of space-division switches are described next.

Banyan Switches. Banyan switches are examples of space-division switches. An $N \times N$ Banyan switch is constructed by arranging a number of binary switching elements into several stages ($\log_2 N$ stages). Figure 10 depicts an 8 × 8 self-routing Banyan switch (14). The switch fabric is composed of twelve 2 × 2 switching elements assembled into three stages. From any of the eight input ports, it is possible to reach all the eight output ports. One desirable characteristic of the Banyan switch is that it is self-routing. Because each cross-point switch has only two output lines, only one bit is required to specify the correct output path. Very simply, if the desired output addresses of a ATM cell is stored in the cell header in binary code, routing decisions for the cell can be made at each cross-point switch by examining the appropriate bit of the destination address.

Although the Banyan switch is simple and possesses attractive features such as modularity, which makes it suitable for VLSI implementation, it also has some disadvantages. One of its disadvantages is that it is internally blocking. In other words, cells destined for different output ports may contend for a common link within the switch. This results in

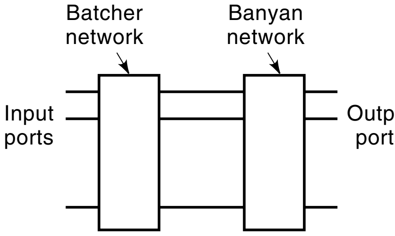

Figure 11. Batcher–Banyan switch.

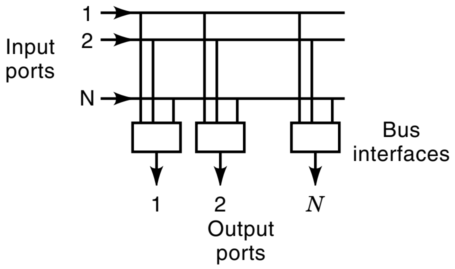

Figure 12. A knockout (crossbar) switch.

blocking all cells that wish to use that link, except for one. Hence, the Banyan switch is referred to as a blocking switch. In Fig. 10, three cells are shown arriving on input ports 1, 3, and 4 with destination port addresses of 0, 1, and 5, respectively. The cell destined for output port 0 and the cell destined for output port 1 end up contending for the link between the second and third stages. As a result, only one of them (the cell from input port 1 in this example) actually reaches its destination (output port 0), while the other is blocked.

Batcher–Banyan Switches. Another example of space-division switches is the Batcher–Banyan switch (14). (See Fig. 11.) It consists of two multistage interconnection networks: a Banyan self-routing network and a Batcher sorting network. In the Batcher–Banyan switch, the incoming cells first enter the sorting network, which takes the cells and sorts them into ascending order according to their output addresses. Cells then enter the Banyan network, which routes the cells to their correct output ports.

As shown earlier, the Banyan switch is internally blocking. However, the Banyan switch possesses an interesting feature. Namely, internal blocking can be avoided if the cells arriving at the Banyan switch's input ports are sorted in ascending order by their destination addresses. The Batcher–Banyan switch takes advantage of this fact and uses the Batcher sorting network to sort the cells, thereby making the Batcher–Banyan switch internally nonblocking. The Starlite switch, designed by Bellcore, is based on the Batcher–Banyan architecture (16).

Crossbar Switches. The crossbar switch interconnects N inputs and N outputs into a fully meshed topology; that is, there are N^2 cross points within the switch (14). (See Fig. 12.) Because it is always possible to establish a connection between any arbitrary input and output pair, internal blocking is impossible in a crossbar switch.

The architecture of the crossbar switch has some advantages. First, it uses a simple two-state cross-point switch (open and connected state), which is easy to implement. Second, the modularity of the switch design allows simple expansion. One can build a larger switch by simply adding more cross-point switches. Lastly, compared to Banyan-based switches, the crossbar switch design results in low transfer latency, because it has the smallest number of connecting points between input and output ports. One disadvantage to this design, however, is the fact that it uses the maximum number of cross points (cross-point switches) needed to implement an $N \times N$ switch.

The knockout switch by AT&T Bell Labs is a nonblocking switch based on the crossbar design (17,18). It has N inputs and N outputs and consists of a crossbar-based switch with a bus interface module at each output (Fig. 12).

Nonblocking Buffered Switches

Although some switches such as Batcher–Banyan and crossbar switches are internally nonblocking, two or more cells may still contend for the same output port in a nonblocking switch, resulting in the dropping of all but one cell. In order to prevent such loss, the buffering of cells by the switch is necessary. Figure 13 illustrates that buffers may be placed (1) in the inputs to the switch, (2) in the outputs to the switch, or (3) within the switching fabric itself, as a shared buffer (14). Some switches put buffers in both the input and output ports of a switch.

The first approach to eliminating output contention is to place buffers in the output ports of the switch (14). In the worst case, cells arriving simultaneously at all input ports can be destined for a single output port. To ensure that no cells are lost in this case, the cell transfer must be performed at N times the speed of the input links, and the switch must be able to write N cells into the output buffer during one cell transmission time. Examples of output buffered switches include the knockout switch by AT&T Bell Labs, the Siemens & Newbridge MainStreetXpress switches, the ATML's VIRATA switch, and Bay Networks' Lattis switch.

The second approach to buffering in ATM switches is to place the buffers in the input ports of the switch (14). Each input has a dedicated buffer, and cells that would otherwise be blocked at the output ports of the switch are stored in input buffers. Commercial examples of switches with input buffers as well as output buffers are IBM's 8285 Nways switches, and Cisco's Lightstream 2020 switches.

A third approach is to use a shared buffer within the switch fabric. In a shared buffer switch, there is no buffer at the input or output ports (14). Arriving cells are immediately injected into the switch. When output contention happens, the winning cell goes through the switch, while the losing cells are stored for later transmission in a shared buffer common to all of the input ports. Cells just arriving at the switch join buffered cells in competition for available outputs. Because

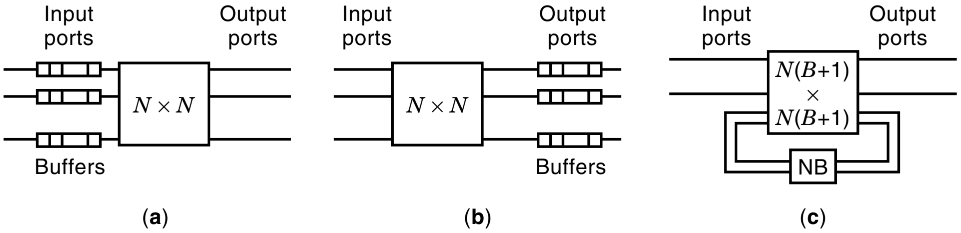

Figure 13. Nonblocking buffered switches.

more cells are available to select from, it is possible that fewer output ports will be idle when using the shared buffer scheme. Thus, the shared buffer switch can achieve high throughput. However, one drawback is that cells may be delivered out of sequence because cells that arrived more recently may win over buffered cells during contention (19). Another drawback is the increase in the number of input and output ports internal to the switch. The Starlite switch with trap by Bellcore is an example of the shared buffer switch architecture (16). Other examples of shared buffer switches include Cisco's Lightstream 1010 switches, IBM's Prizma switches, Hitachi's 5001 switches, and Lucent's ATM cell switches.

CONTINUING RESEARCH IN ATM NETWORKS

ATM is continuously evolving, and its attractive ability to support broadband integrated services with strict quality of service guarantees has motivated the integration of ATM and existing widely deployed networks. Recent additions to ATM research and technology include, but are not limited to, seamless integration with existing LANs [e.g., LAN emulation (20)], efficient support for traditional Internet IP networking [e.g., IP over ATM (21), IP switching (22)], and further development of flow and congestion control algorithms to support existing data services [e.g., ABR flow control (12)]. Research on topics related to ATM networks is currently proceeding and will undoubtedly continue to proceed as the technology matures.

BIBLIOGRAPHY

1. *CCITT Recommendation I-Series.* Geneva: International Telephone and Telegraph Consultative Committee.
2. J. B. Kim, T. Suda, and M. Yoshimura, International standardization of B-ISDN, *Comput. Networks ISDN Syst.,* **27**: 1994.
3. *CCITT Recommendation G-Series.* Geneva: International Telephone and Telegraph Consultative Committee.
4. *ATM Forum Technical Specifications [Online].* Available www: www.atmforum.com
5. *Report of ANSI T1S1.5/91-292,* Simple and Efficient Adaptation Layer (SEAL), August 1991.
6. *Report of ANSI T1S1.5/91-449,* AAL5—A New High Speed Data Transfer, November 1991.
7. *CCITT Recommendation Q-Series.* Geneva: International Telephone and Telegraph Consultative Committee.
8. J. Bae and T. Suda, Survey of traffic control schemes and protocols in ATM networks, *Proc. IEEE,* **79**: 1991.
9. B. J. Vickers et al., Congestion control and resource management in diverse ATM environments, *IECEJ J.,* **J76-B-I** (11): 1993.
10. J. S. Turner, New directions in communications (or which way to the information age?), *IEEE Commun. Mag.,* **25** (10): 1986.
11. G. Gallassi, G. Rigolio, and L. Fratta, ATM: Bandwidth assignment and bandwidth enforcement policies. *Proc. GLOBECOM'89.*
12. ATM Forum, ATM Forum Traffic management specification version 4.0, *af-tm-0056.000,* April 1996, Mountain View, CA: ATM Forum.
13. Quantum Flow Control version 2.0, *Flow Control Consortium, FCC-SPEC-95-1,* [Online], July 1995. http://www.qfc.org
14. Y. Oie et al., Survey of switching techniques in high-speed networks and their performance, *Int. J. Satellite Commun.,* **9**: 285–303, 1991.
15. M. De Prycker and M. De Somer, Performance of a service independent switching network with distributed control, *IEEE J. Select. Areas Commun.,* **5**: 1293–1301, 1987.
16. A. Huang and S. Knauer, Starlite: A wideband digital switch. *Proc. IEEE GLOBECOM'84,* 1984.
17. K. Y. Eng, A photonic knockout switch for high-speed packet networks, *IEEE J. Select. Areas Commun.,* **6**: 1107–1116, 1988.
18. Y. S. Yeh, M. G. Hluchyj, and A. S. Acampora, The knockout switch: A simple, modular architecture for high-performance packet switching, *IEEE J. Select. Areas Commun.,* **5**: 1274–1283, 1987.
19. J. Y. Hui and E. Arthurs, A broadband packet switch for integrated transport, *IEEE J. Select. Areas Commun.,* **5**: 1264–1273, 1987.
20. ATM Forum, LAN emulation over ATM version 1.0. *AF-LANE-0021,* 1995, Mountain View, CA: ATM Forum.
21. IETF, IP over ATM: A framework document, *RFC-1932,* 1996.
22. Ipsilon Corporation, IP switching: The intelligence of routing, *The Performance of Switching* [Online]. Available www: www.ipsilon.com

Tatsuya Suda
University of California, Irvine

ATC. See Air traffic control.

ATM. See Statistical multiplexing.

ATM NETWORKS. See Asynchronous transfer mode networks.

ATM NETWORKS, VIDEO ON. See Video on ATM networks.

ATMOSPHERICS. See Whistlers.

ATTENUATION. See Refraction and attenuation in the troposphere.