This is the first part of Chapter 10 from the second edition of :

Networks, Routers and Transputers:

Function, Performance and applications

Edited by: M.D. May, P.W. Thompson, and P.H. Welch

IOS Press
Van Diemenstraat 94
1013 CN Amsterdam
Netherlands

IOS Press, Inc.
P.O. Box 10558
Burke, VA 22009–0558
U.S.A.

IOS Press/Lavis Marketing
73 Lime Walk
Headington
Oxford OX3 7AD
England

Kaigai Publications, Ltd.
21 Kanda Tsukasa–Cho 2–Chome
Chiyoda–Ka
Tokyo 101
Japan

This chapter was written by C. Barnaby and N. Richards.

10 A Generic Architecture for ATM Systems

10.1 Introduction

The rapid growth in the use of personal computers and high–performance workstations over the last ten years has fueled an enormous expansion in the data communications market. The desire to connect computers together to share information, common databases and applications led to the development of Local Area Networks and the emergence of ***distributed computing.*** At the same time, the geographical limitations of LANs and the desire to provide corporate–wide networks stimulated the development towards faster, more reliable telecommunications networks for LAN interconnection, with the need to support data as well as traditional voice traffic. The resulting increase in the use of digital technology and complex protocols has resulted in the need for enormous computing capability within the telecommunications network itself, with the consequent emergence of the concept of the ***Intelligent Network***. With new, higher bandwidth applications such as video and multimedia on the horizon and user pressure for better, more seamless connection between computer networks, this convergence of computing and communications systems looks set to accelerate during the nineties.

A key step in this convergence is the development by the CCITT of standards for the ***Broadband Integrated Services Digital Network (B–ISDN)***. B–ISDN seeks to provide a common infrastructure on which a wide variety of voice, data and video services can be provided, thereby eliminating (hopefully) the final barriers between the world of computer networks and the world of telecommunications. The technological basis for B–ISDN chosen by the CCITT is the ***Asynchronous Transfer Mode (ATM)***, a fast–packet switching technique using small, self–routing packets called ***cells***.

The single most important element which has driven the development of both distributed computing and the intelligent network is the microprocessor. Indeed, as systems such as telecommunications networks have come to look more like distributed computers, so microprocessor architectures which support distributed multi–processing have come to look like communications networks. A message–passing computer architecture, such as that of the transputer, shares much in common with a packet switching system and thus provides a natural architecture from which to build communication systems. The communications architecture of the latest generation transputer, the T9000, shares much in common with ATM and is thus a natural choice for the implementation of ATM systems.

In this Chapter we describe the application of the transputer, in particular the serial links and packet routing capabilities of the communications architecture, to the design of ATM switching systems. We discuss their use in public switching systems and present a generic architecture for the implementation of private ATM switches and internetworking applications. We look at terminal adaption requirements and develop some ideas for interfacing transputers, routers and serial links to ATM networks. Finally, we consider various aspects of the performance of this architecture.

10.2 An Introduction to Asynchronous Transfer Mode

10.2.1 Background

Current communications systems split roughly into two basic categories:–

a) The existing telephone network, a ***Wide Area Network (WAN)***, predominantly designed around the requirements to transmit voice traffic around the globe

b) Existing ***Local Area Networks (LANs)***, designed to transmit digital data between computers over relatively short distances

As the idea of distributed computing and corporate–wide networks has gained acceptance, so has the desire to connect computers (predominantly PC's and workstations) across larger and larger distances. Unfortunately, seamless transmission of data from computer to computer across the globe using either of the existing types of networks is severely limited by the constraints inherent in each system:–

a) The telephone network is optimized for low–bandwidth, low latency point–to–point voice traffic (this traffic is relatively insensitive to noise and data errors)

b) Local area networks are optimized for high bandwidth computer data (which is not generally sensitive to latency, but is intolerant of data errors and usually uses some form of shared medium)

In summary, the telephone network is unreliable and too slow and LANs can't carry voice easily and don't go far enough. This split has led to communications networks developing from two directions over the past decade or so; one trying to make the telephone network faster and the other to make LANs go further.

Attempts to make the telephone network faster and more useful to data communications has resulted in a plethora of communications techniques and standards to transmit data between otherwise isolated computers. First came analogue modems (maximum 19kbits/s), then digital networks like X.25 (generally 64kbits/s), and latterly higher bandwidth access via basic/primary rate ISDN, frame relay, etc. However, the fastest access rates in common use are still no more than 1.5 – 2 Mbits/s, compared with 10–16 Mbits/s on LANs such as ethernet and token ring. Of more concern has been the need to use 'heavyweight' protocols to protect computer data as it travels over the existing, relatively unreliable, telephone network. The processing overhead of these protocols has a significant impact on the useable bandwidth available.

Progress on extending LANs has resulted in the development of ***Metropolitan Area Networks (MANs)*** designed to offer high bandwidth connections between computers over an area the size of, say, a reasonable city. An example is the ***Fibre Distributed Data Interface (FDDI)***, which can offer 100 Mbits/s connection over several kilometres. FDDI, however, is still a shared medium, is relatively expensive, requires new fibre cabling (although copper standards for short distances have been developed) and still requires expensive internetworking equipment to connect to WANs. In addition it cannot support voice traffic very easily. Another standard, ***IEEE 802.6***, shows greater promise in the longer term since it is designed to be 'media independent' and also to integrate more easily with WANs.

However, the situation has become exacerbated in recent years with the arrival of higher and higher bandwidth users (large CAD design databases, for example) and the expected growth of ***multimedia***, with its requirement to support voice, video and computer data applications (multimedia applications are described in more detail in Chapter 11 of this book). So, into the picture comes the CCITT with its efforts to provide the basis for the ***Broadband–ISDN***, a telecommunications infrastructure capable of supporting any type of traffic anywhere across the globe. The

CCITT has based this infrastructure on ***Asynchronous Transfer Mode (ATM)*** technology, which is described in the next section.

10.2.2 Basic ATM Concepts

ATM Cells

ATM is based on the concept of a universal ***cell*** (a very small packet) 53 bytes in length, of which the first 5 bytes are used for a routing header and the remaining 48 bytes are for carrying data. Each ATM cell is a self–contained entity which can be routed individually through each switching node in the network from source to destination. This cell has no awareness of the type of data it is carrying and can be considered to be a universal carrier of data, a sort of communications ‘truck’ (or ‘lorry’, for those of us in the UK) into which you can put voice, video, data, etc. The term ‘asynchronous’ is used since no clocking or timing relationship is maintained between the ATM cells.

DATA FIELD	HEADER
48 bytes	5 bytes

Figure 10.1 ATM Cell

The CCITT Recommendations for the public networks have so far defined ATM to run at a nominal 155 Mbits/s to fit in with the *Synchronous* (framed) bit rates used in the transmission systems between exchanges. In these systems, the ATM cells are packed in like bricks into a two–dimensional frame for transport to the next switch (described later). In reality the bit rate available for the ATM cells is about 149 Mbits/s once the framing overhead has been allowed for. It is expected that a 622 Mbits/s standard will follow (4 x 155 Mbit/s plus some extra overhead) with eventual data rates up to 2.4 Gbits/s being anticipated.

The situation for private networks is not yet clear, since the standards have not yet been set. 155 Mbits/s seems likely, but since ATM cells can be transmitted either framed (synchronously) or unframed (asynchronously) lower data rates (< 155 Mbits/s) for unframed cells may also be adopted. It is important to remember that this is the point–to–point bandwidth available to each connection, not the bandwidth of the network as a whole, which is the case of conventional shared–medium LANs/MANs like ethernet and FDDI.

ATM Connections

Any user who wishes to gain access to an ATM network must first establish a connection with the local switch. In the diagram below, our subscriber picks up a (very sophisticated) ATM telephone in order to send data across the network. During call set–up, the user negotiates with the network for the call and service characteristics desired. For example, the number dialled, bandwidth and service quality (error rates, etc.) required may be sent to the local switch. This is important, since different types of traffic require different performance from the network and the user will be charged accordingly. The local switch then negotiates with all the other switches necessary to connect to the desired destination. Assuming the connection is possible and that the user requested bandwidth and quality of service can be supported, the local switch confirms the connection to the user and allocates an ATM cell routing header from those available. If the requirements are not met and a lower standard of service is offered, it is up to the user to either accept this or terminate the call. Otherwise, the user equipment can now start sending data into the network using ATM cells and the routing header specified by the switch.

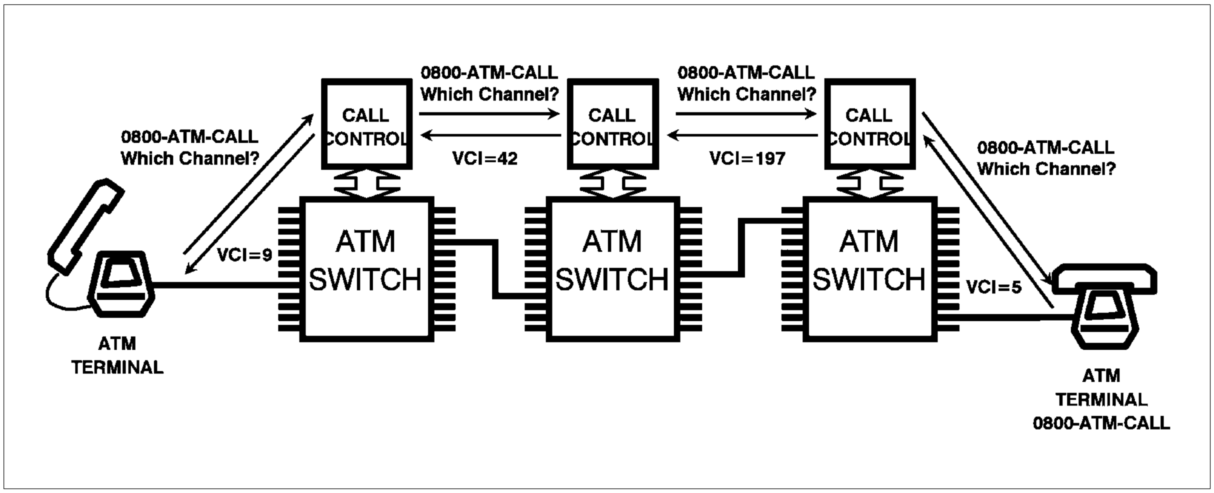

Figure 10.2 ATM call connections

Cell Header Policing

During the call set–up, the user negotiates with the network for certain service characteristics such as bandwidth. This may be specified in terms of the peak and average bandwidth required from the network (other parameters are under discussion). Since the user will be charged (on the public network) for his/her use of the system, and this charge will be dependent on the bandwidth negotiated, it is clearly necessary to monitor the actual use made to ensure nobody is cheating. It is proposed that this be done by monitoring the instantaneous and average bandwidth (or any other parameters) used by each cell on the network. This is referred to as ***Cell Header Policing*** and is done on a cell–by–cell basis on input by the network interface (ATM line card) at each ATM switch. Various algorithms have been proposed to perform this bandwidth policing, the most common of which is the ***Leaky Bucket*** algorithm. Depending on the type of service negotiated, transgressors of the negotiated policing limits may either be charged more (according to their use) or find their cells being discarded if they threaten the quality–of–service of other users.

Another important aspect of header policing arises due to the nature of ATM itself. On entering each ATM switch, each ATM cell is routed asynchronously (hence the name) from input to the appropriate output across the ATM switching fabric. Since cells may suffer delay in crossing this fabric due to internal traffic congestion, they may arrive at the output in 'clusters', resulting in a larger instantaneous bandwidth through no fault of the user (this is analogous to the behavior of buses in cities...). In extremis, if no flow control is provided across the switch fabric, cells may arrive at the output out of order. It would clearly be unreasonable to charge the user more or, worse, start discarding cells because of this behavior, so it is therefore necessary for the ATM switch itself to re–time the cells on output to the next switching node in order to meet the original user requirements. There is, therefore, a requirement to use header policing on output, as well as on input, and the system must ensure that cell order is maintained across the switch.

Cell Header Translation

The route that an ATM cell takes through the B–ISDN *network* is determined by the routing values in the cell header. Only a very limited routing 'space' is provided for each ATM cell since the header is only 5 bytes long and the bit–fields available are necessarily limited. To overcome this, the routing value is re–used (re–mapped) at each ATM switching point in the B–ISDN network. That is, the routing value only applies locally to one switching node and changes as the cell progresses through the network from one switching node to another. This constant re–mapping of the cell header is called ***Cell Header Translation*** and is performed when the cell is received by the ATM switch. Cell header translation is performed on a cell–by–cell basis by the network interface, or ATM 'line card', and with ATM operating at 155 or 620 Mbits/s, this requires either very fast processing, custom hardware, or preferably an intelligent combination of the two.

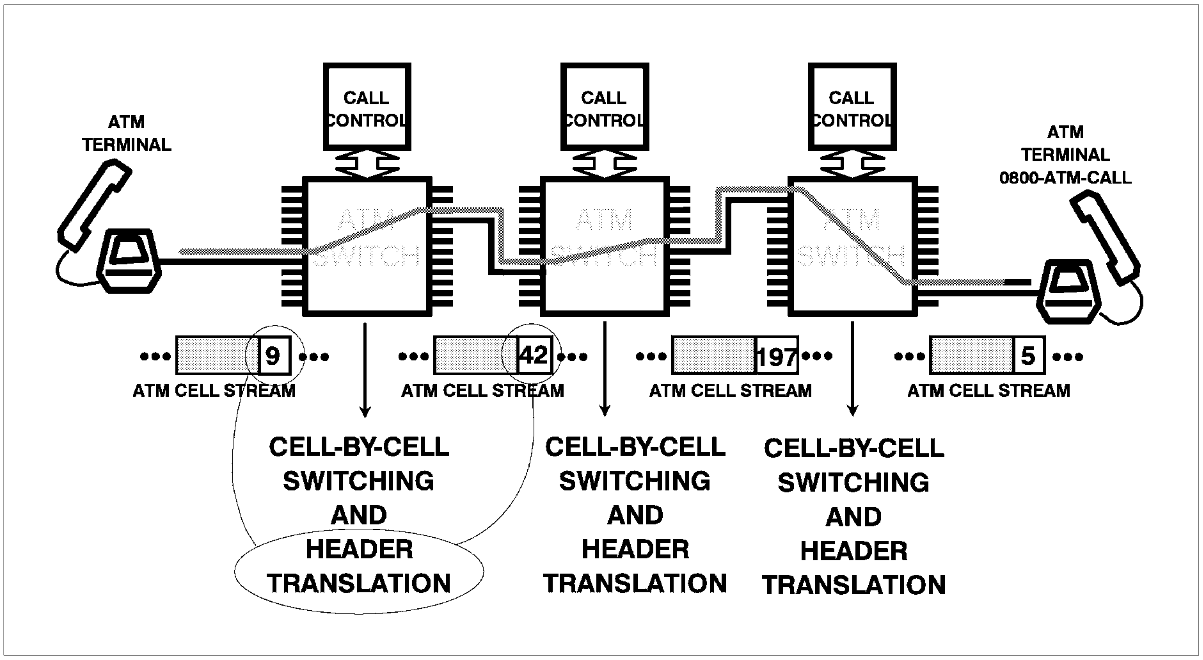

Figure 10.3 ATM Cell Header Translation

Within the ATM switch itself, routing decisions from network input to network output across the internal switching fabric also need to be made on a cell–by–cell basis. It may be necessary to perform another translation of the ATM cell header, to an internal format for routing purposes within the ATM fabric itself.

10.2.3 ATM Protocols and Standards

Having explained the basic principles it is now worth considering a few of the details. A good place to start is the CCITT Recommendations which apply to ATM. These are part of the I.xxx series of Recommendations which form the standards for ISDN networks.

ATM Protocol Reference Model

Like all good protocols, the ATM standard is defined as a series of layers. There are 3 basic layers which, from the top down, are:–

AAL: The **‘*ATM Adaption Layer*’** defines various ‘mapping’ mechanisms from existing protocols (ISDN, voice, video, LAN data, etc.) onto ATM and vice versa.

ATM: This defines the ***ATM cell***, routing techniques and error mechanisms

PHY: This is the ***Physical*** layer and defines media (for example fibre/copper, connectors, etc.), bit timings, framing standards, etc.

In addition, the ATM standards describe Management and Control functions for each of the layers, such as call set–up and maintenance functions within the network. These layers constitute the ATM ***Protocol Reference Model (PRM)*** and are shown pictorially in Figure 10.4. The details of each layer are shown in Figure 10.5.

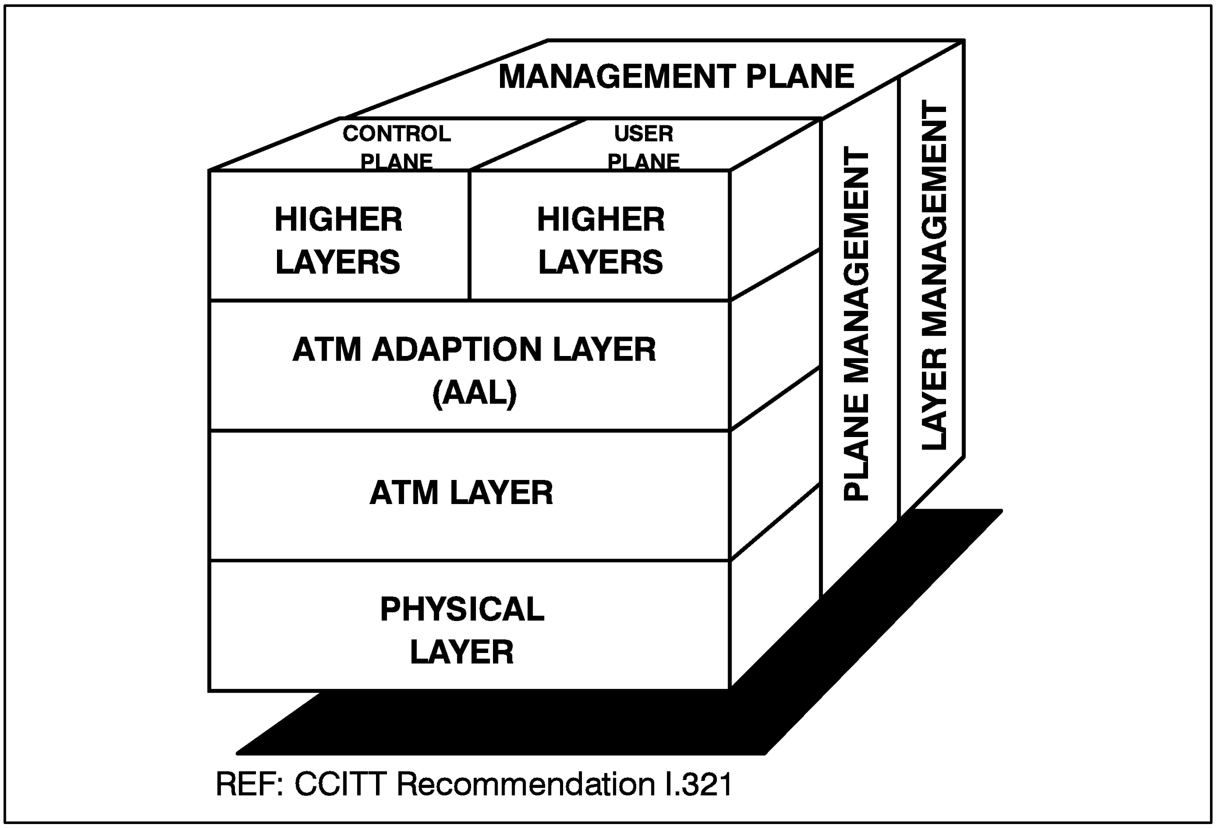

Figure 10.4 ATM Protocol Reference Model (PRM) [1]

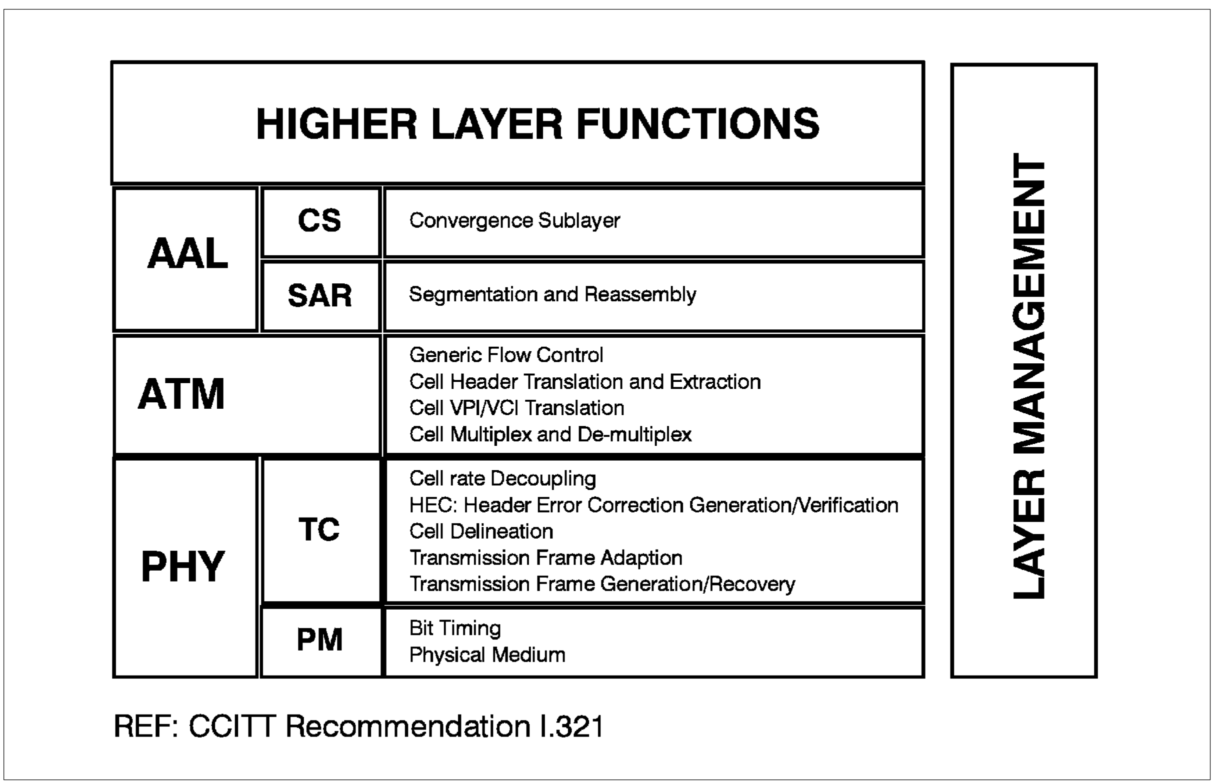

Figure 10.5 ATM PRM Layer Functions [1]

It is important to point out that many of the details in the ATM standards are still not yet finalized, particularly many of the management functions. However, a simplified diagram showing what all 3 layers do is given below and this may be referred to in the discussion of each layer in the following sections.

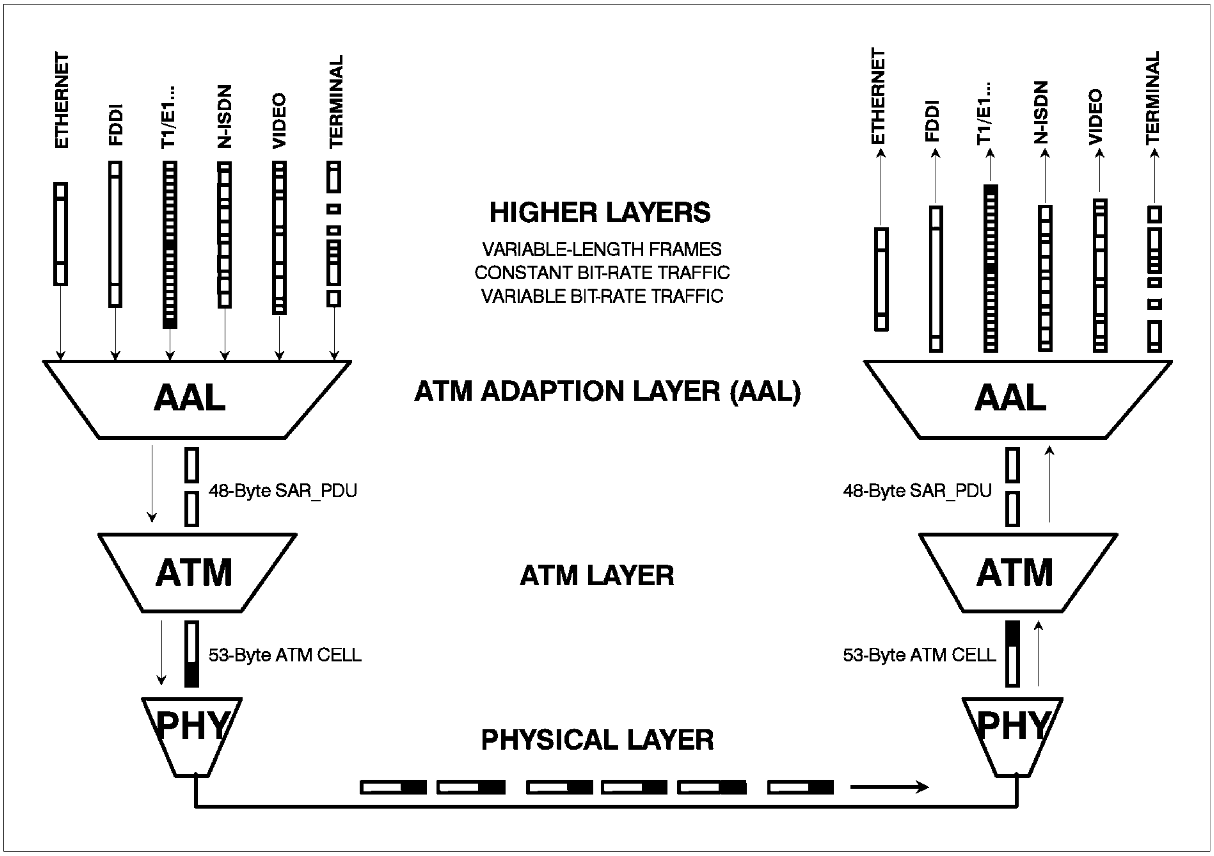

Figure 10.6   ATM Summary

The AAL Layer

The 'ATM Adaption Layer' is responsible for mapping other protocols onto the ATM cell format for transmission and switching. Examples of this would be to carry data traffic (in the form of ethernet, token ring or FDDI frames), voice traffic (64 kbit/s ISDN, for instance) or video traffic. Of necessity, the AAL layer comes in several varieties to suit the nature of the protocols being mapped. Data traffic is typically 'bursty' in nature and needs to be handled on a frame–by–frame basis. Voice traffic is referred to as 'constant bit–rate' traffic, that is, it is a constant flow of bits with no pause. Video traffic is referred to as 'variable bit–rate', since video coding algorithms typically generate an output which varies in bit–rate according to the contents of the picture being transmitted. The AAL layer provides functions to map all of these different types of traffic onto a flow of ATM cells. shown in the previous diagram.

There are four types of AAL specified in the CCITT standards, denoted as AAL1 to AAL4. Recently, a proposal for a fifth, AAL5, has been made with a view to providing a 'lightweight' AAL for frame (packet) based computer data (currently provided by AAL3). In each case, the AAL layer is responsible for ***Segmentation*** of the outgoing data, whatever it is, into small chunks of 48 bytes which then form the data field of the ATM cell. This 48–byte field will also contain overheads, such as CRC and payload type information which depend on which type of AAL is in use. For example, the actual user data field in AAL3 is only 44 bytes, with 2 bytes of header and 2 bytes of trailer added by the AAL to form the 48–byte field. Incoming data received from the ATM layer undergoes ***Reassembly*** by the AAL to provide an appropriate output stream, i.e. it undergoes the reverse of the segmentation process. An example is given in Figure 10.7, showing the AAL3 operation.

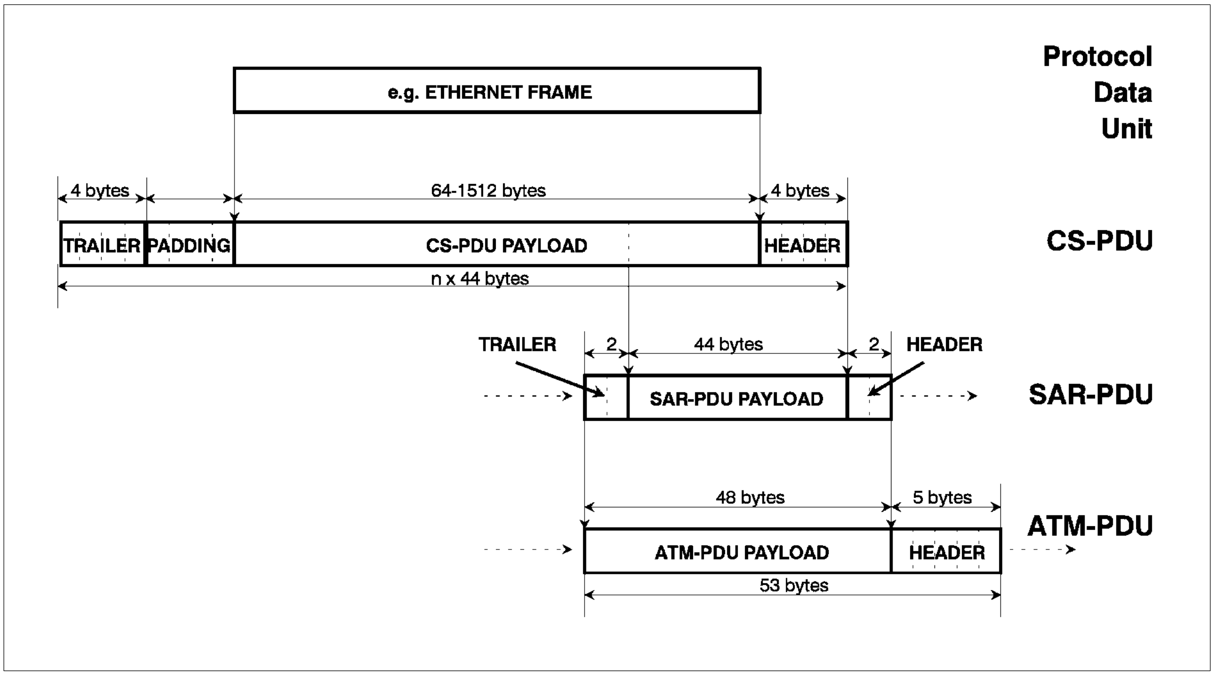

Figure 10.7 AAL3 Example

The use of each layer of the ATM protocol standard is illustrated in a simple form in Figure 10.8. ATM and PHY layer protocols are implemented everywhere in our simple network, but an AAL is only invoked at the termination points of the ATM network; that is, an AAL function is needed at:–

- the endpoints of the network (the user terminals)
- points where the ATM network meets another type of network (connecting to an ethernet network, for example)
- certain control nodes within the ATM network itself (passing signalling, management and control information between the control processors in the ATM exchanges, for instance).

There is insufficient space here to cover the AAL layer in detail so the reader is referred to the many papers on the subject for more detailed information, for example in [1] and [2]

The AAL layer is not needed as part of the switching function of an ATM network; this is handled entirely by the ATM layer.

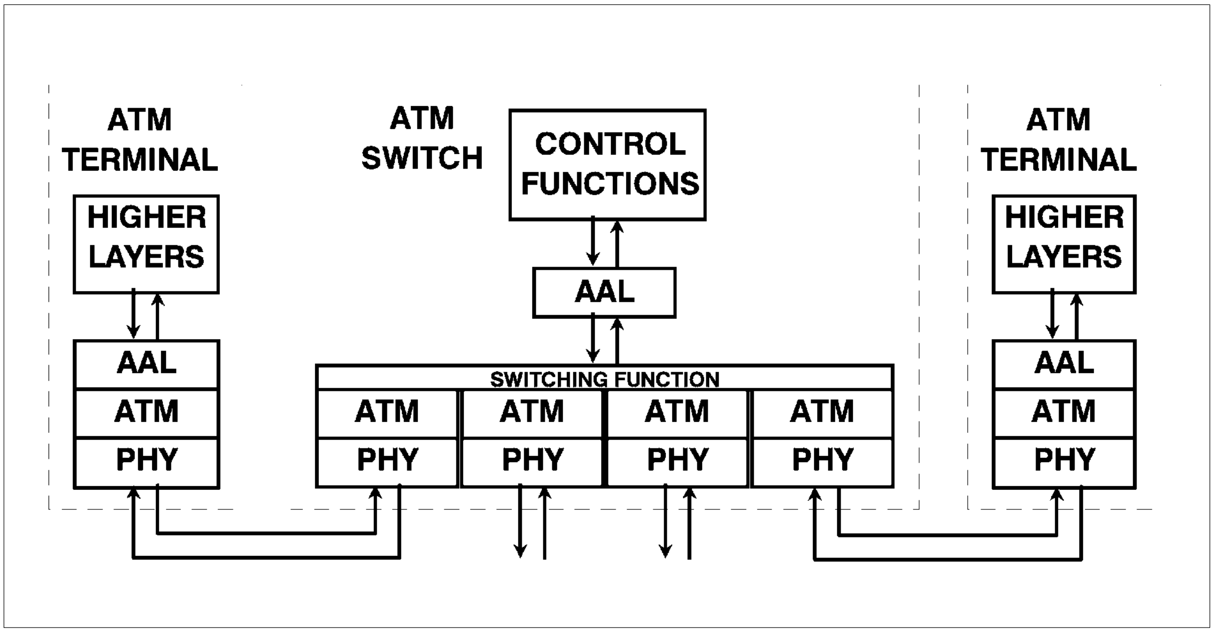

Figure 10.8 PRM Illustration in a simple Network

ATM Layer

There are two versions of the ATM cell format, one for the ***User–Network Interface (UNI)*** and another for the ***Network Node Interface (NNI)***. The basic structure of the ATM cell is shown in Figure 10.9.

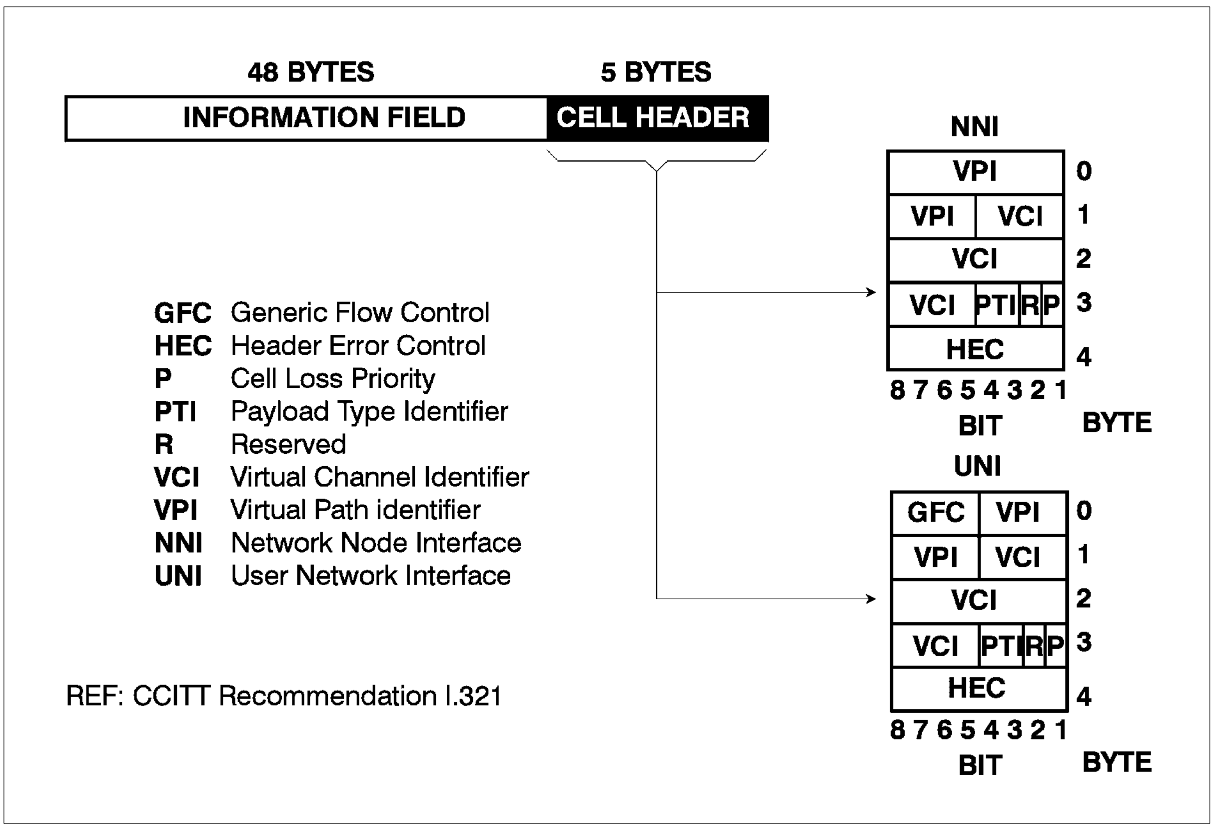

Figure 10.9 ATM Cell Structure

The cell header contains routing information, control bits and error detection features. Two methods of routing are provided; one is via the ***'Virtual Channel Identifier'*** and the other the ***'Virtual Path Identifier'*** (VCI and VPI respectively).