Part Three: Wide Area Networks

Chapter 9

WAN Technology and Protocols

LEARNING OBJECTIVES

After reading this chapter, you should be able to:

- Define circuit switching and describe the key elements of circuit-switching networks.
- Define packet switching and describe the key elements of packet-switching technology.
- Discuss the relative merits of circuit switching and packet switching and analyze the circumstances for which each is most appropriate.
- Describe the features and characteristics of ATM networks.

Part Two describes how information can be encoded and transmitted over a communications link. We now turn to the broader discussion of networks, which can be used to interconnect many devices. The chapter begins with a general discussion of switched communications networks. The remainder of the chapter focuses on wide area networks and, in particular, on traditional approaches to wide area network design: circuit switching and packet switching.

Since the invention of the telephone, circuit switching has been the dominant technology for voice communications, and it has remained so well into the digital era. This chapter looks at the key characteristics of a circuit-switching network.

Around 1970, research began on a new form of architecture for long-distance digital data communications: packet switching. Although the technology of packet switching has evolved substantially since that time, it is remarkable that (1) the basic technology of packet switching is fundamentally the same today as it was in the early 1970s networks, and (2) packet switching remains one of the few effective technologies for long-distance data communications.

This chapter provides an overview of packet-switching technology. We will see, in this chapter and later in this part, that many of the advantages of packet switching (flexibility, resource sharing, robustness, responsiveness) come with a cost. The packet-switching network is a distributed collection of packet-switching nodes. Ideally, all packet-switching nodes would always know the state of the entire network. Unfortunately, because the nodes are distributed, there is a time delay between a change in status in one portion of the network and knowledge of that change elsewhere. Furthermore, there is overhead involved in communicating status information. As a result, a packet-switching network can never perform "perfectly," and elaborate algorithms are used to cope with the time delay and overhead penalties of network operation. These same issues will appear again when we discuss internetworking in Part Five.

9.1 SWITCHED COMMUNICATIONS NETWORKS

For transmission of data[1] beyond a local area, communication is typically achieved by transmitting data from source to destination through a network of intermediate switching nodes; this switched network design is typically used to implement LANs as well. The switching nodes are not concerned with the content of the data; rather, their purpose is to provide a switching facility that will move the data from node to node until they reach their destination. Figure 9.1 illustrates a simple network. The devices attached to the network may be referred to as *stations*. The stations may be computers, terminals, telephones, or other communicating devices. We refer to

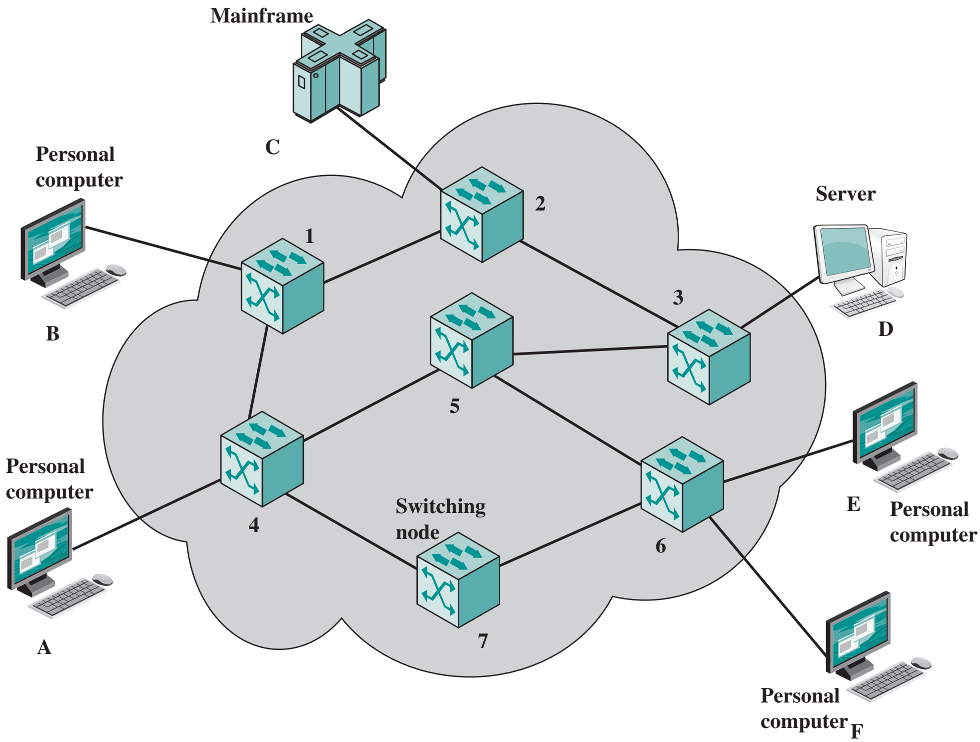


Figure 9.1 Simple Switching Network

[1]We use this term here in a very general sense, to include voice, image, and video, as well as ordinary data (e.g., numbers, text).

the switching devices whose purpose is to provide communication as *nodes*. Nodes are connected to one another in some topology by transmission links. Each station attaches to a node, and the collection of nodes is referred to as a *communications network*.

In a *switched communication network*, data entering the network from a station are routed to the destination by being switched from node to node.

EXAMPLE 9.1 In Figure 9.1, data from station A intended for station F are sent to node 4. They may then be routed via nodes 5 and 6 or nodes 7 and 6 to the destination. Several observations are in order:

1. Some nodes connect only to other nodes (e.g., 5 and 7). Their sole task is the internal (to the network) switching of data. Other nodes have one or more stations attached as well; in addition to their switching functions, such nodes accept data from and deliver data to the attached stations.
2. Node–station links are generally dedicated point-to-point links. Node–node links are usually multiplexed, using either frequency-division multiplexing (FDM) or time-division multiplexing (TDM).
3. Usually, the network is not fully connected; that is, there is not a direct link between every possible pair of nodes. However, it is always desirable to have more than one possible path through the network for each pair of stations. This enhances the reliability of the network.

Two different technologies are used in wide area switched networks: circuit switching and packet switching. These two technologies differ in the way the nodes switch information from one link to another on the way from source to destination.

9.2 CIRCUIT-SWITCHING NETWORKS

Communication via circuit switching implies that there is a dedicated communication path between two stations. That path is a connected sequence of links between network nodes. On each physical link, a logical channel is dedicated to the connection. Communication via circuit switching involves three phases, which can be explained with reference to Figure 9.1.

1. **Circuit establishment**. Before any signals can be transmitted, an end-to-end (station-to-station) circuit must be established. For example, station A sends a request to node 4 requesting a connection to station E. Typically, the link from A to 4 is a dedicated line, so that part of the connection already exists. Node 4 must find the next leg in a route leading to E. Based on routing information and measures of availability and perhaps cost, node 4 selects the link to node

5, allocates a free channel (using FDM or TDM) on that link, and sends a message requesting connection to E. So far, a dedicated path has been established from A through 4 to 5. Because a number of stations may attach to 4, it must be able to establish internal paths from multiple stations to multiple nodes. How this is done is discussed later in this section. The remainder of the process proceeds similarly. Node 5 allocates a channel to node 6 and internally ties that channel to the channel from node 4. Node 6 completes the connection to E. In completing the connection, a test is made to determine if E is busy or is prepared to accept the connection.

2. **Data transfer**. Data can now be transmitted from A through the network to E. The transmission may be analog or digital, depending on the nature of the network. As the carriers evolve to fully integrated digital networks, the use of digital (binary) transmission for both voice and data is becoming the dominant method. The path is A-4 link, internal switching through 4, 4-5 channel, internal switching through 5, 5-6 channel, internal switching through 6, 6-E link. Generally, the connection is full duplex.
3. **Circuit disconnect**. After some period of data transfer, the connection is terminated, usually by the action of one of the two stations. Signals must be propagated to nodes 4, 5, and 6 to deallocate the dedicated resources.

Note that the connection path is established before data transmission begins. Thus, channel capacity must be reserved between each pair of nodes in the path, and each node must have available internal switching capacity to handle the requested connection. The switches must have the intelligence to make these allocations and to devise a route through the network.

Circuit switching can be rather inefficient. Channel capacity is dedicated for the duration of a connection, even if no data are being transferred. For a voice connection, utilization may be rather high, but it still does not approach 100%. For a client/server or terminal-to-computer connection, the capacity may be idle during most of the time of the connection. In terms of performance, there is a delay prior to signal transfer for call establishment. However, once the circuit is established, the network is effectively transparent to the users. Information is transmitted at a fixed data rate with no delay other than the propagation delay through the transmission links. The delay at each node is negligible.

Circuit switching was developed to handle voice traffic but is now also used for data traffic. The best-known example of a circuit-switching network is the public telephone network (Figure 9.2). This is actually a collection of national networks interconnected to form the international service. Although originally designed and implemented to service analog telephone subscribers, it handles substantial data traffic via modem and is gradually being converted to a digital network. Another well-known application of circuit switching is the private branch exchange (PBX), used to interconnect telephones within a building or office. Circuit switching is also used in private networks. Typically, such a network is set up by a corporation or other large organization to interconnect its various sites. Such a network usually consists of PBX systems at each site interconnected by dedicated, leased lines obtained from one of the carriers, such as AT&T. A final common example of the application of circuit switching is the data switch. The data switch is similar to the

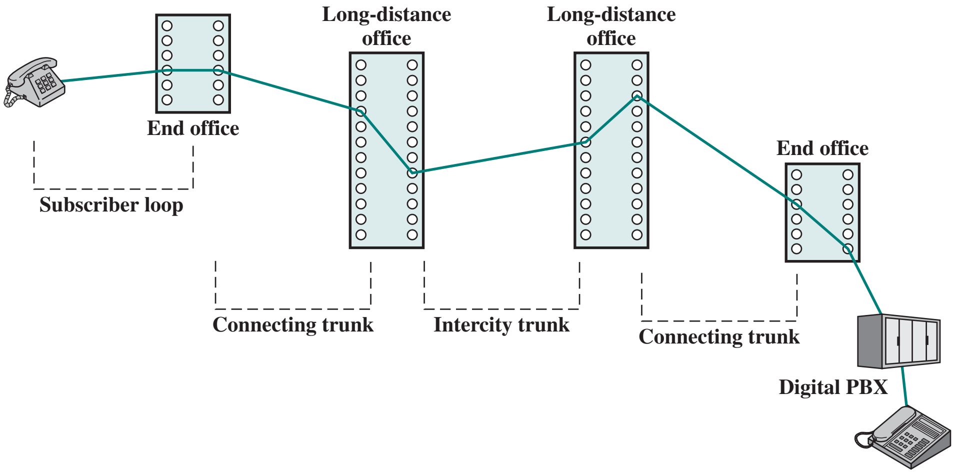


Figure 9.2 Example Connection Over a Public Circuit-Switching Network

PBX but is designed to interconnect digital data processing devices, such as terminals and computers.

A public telecommunications network can be described using four generic architectural components:

- **Subscribers**: The devices that attach to the network. It is still the case that most subscriber devices to public telecommunications networks are telephones, but the percentage of data traffic increases year by year.
- **Subscriber line**: The link between the subscriber and the network, also referred to as the **subscriber loop** or **local loop**. Almost all local loop connections use twisted-pair wire. The length of a local loop is typically in a range from a few kilometers to a few tens of kilometers.
- **Exchanges**: The switching centers in the network. A switching center that directly supports subscribers is known as an end office. Typically, an end office will support many thousands of subscribers in a localized area. There are over 19,000 end offices in the United States, so it is clearly impractical for each end office to have a direct link to each of the other end offices; this would require on the order of 2×10^8 links. Rather, intermediate switching nodes are used.
- **Trunks**: The branches between exchanges. Trunks carry multiple voice-frequency circuits using either FDM or synchronous TDM. We referred to these as carrier systems in Chapter 8.

Subscribers connect directly to an end office, which switches traffic between subscribers and between a subscriber and other exchanges. The other exchanges are responsible for routing and switching traffic between end offices. This distinction is shown in Figure 9.3. To connect two subscribers attached to the same end office, a circuit is set up between them in the same fashion as described before. If two subscribers connect to different end offices, a circuit between them consists of a chain of circuits through one or more intermediate offices. In the figure, a connection is established between lines a and b by simply setting up the connection through the end office. The

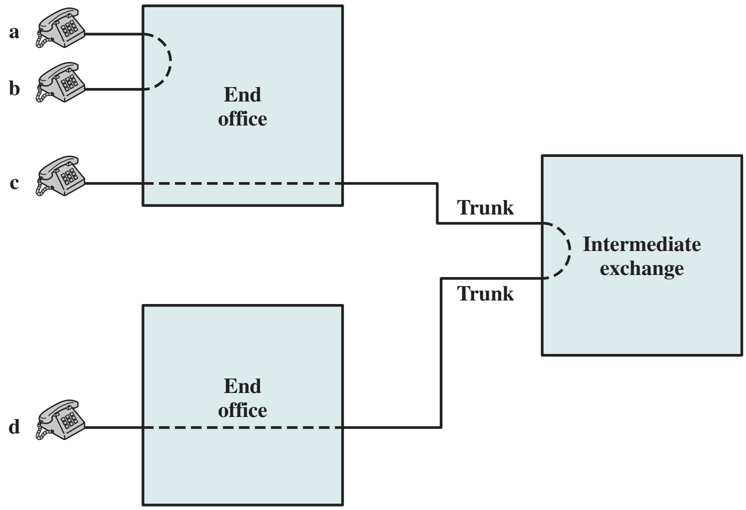


Figure 9.3 Circuit Establishment

connection between c and d is more complex. In c's end office, a connection is established between line c and one channel on a TDM trunk to the intermediate switch. In the intermediate switch, that channel is connected to a channel on a TDM trunk to d's end office. In that end office, the channel is connected to line d.

Circuit-switching technology has been driven by those applications that handle voice traffic. One of the key requirements for voice traffic is that there must be virtually no transmission delay and certainly no variation in delay. A constant signal transmission rate must be maintained, because transmission and reception occur at the same signal rate. These requirements are necessary to allow normal human conversation. Further, the quality of the received signal must be sufficiently high to provide, at a minimum, intelligibility.

Circuit switching achieved its widespread, dominant position because it is well suited to the analog transmission of voice signals. In today's digital world, its inefficiencies are more apparent. However, despite its inefficiencies, circuit switching will remain an attractive choice for both local area and wide area networking. One of its key strengths is that it is transparent. Once a circuit is established, it appears as a direct connection to the two attached stations; no special networking logic is needed at the station.

9.3 CIRCUIT-SWITCHING CONCEPTS

The technology of circuit switching is best approached by examining the operation of a single circuit-switching node. A network built around a single circuit-switching node consists of a collection of stations attached to a central switching unit. The central switch establishes a dedicated path between any two devices that wish to communicate. Figure 9.4 depicts the major elements of such a one-node network. The dotted lines inside the switch symbolize the connections that are currently active.

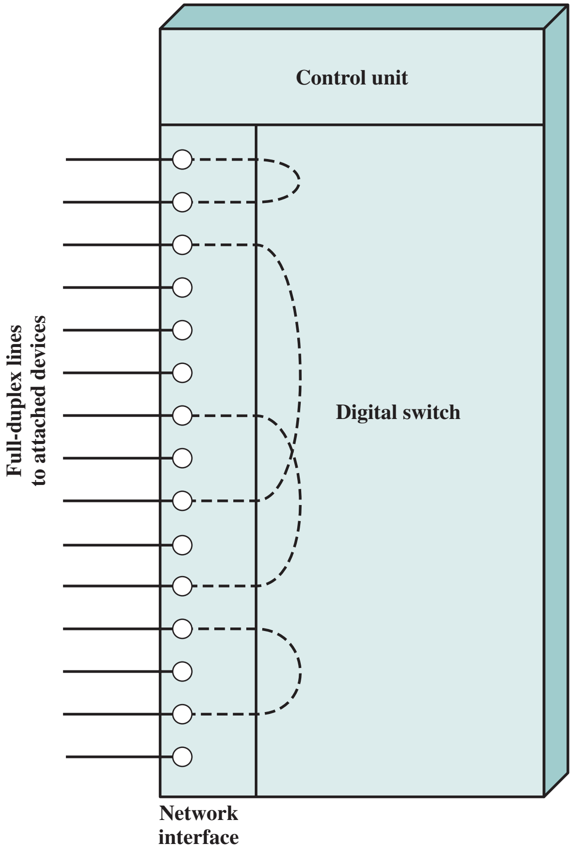


Figure 9.4 Elements of a Circuit-Switch Node

The heart of a modern system is a **digital switch**. The function of the digital switch is to provide a transparent signal path between any pair of attached devices. The path is transparent in that it appears to the attached pair of devices that there is a direct connection between them. Typically, the connection must allow full-duplex transmission.

The **network interface** element represents the functions and hardware needed to connect digital devices, such as data processing devices and digital telephones, to the network. Analog telephones can also be attached if the network interface contains the logic for converting to digital signals. Trunks to other digital switches carry TDM signals and provide the links for constructing multiple-node networks.

The **control unit** performs three general tasks. First, it establishes connections. This is generally done on demand, that is, at the request of an attached device. To establish the connection, the control unit must handle and acknowledge the request, determine if the intended destination is free, and construct a path through the switch. Second, the control unit must maintain the connection. Because the digital switch uses time-division principles, this may require ongoing manipulation of the switching elements. However, the bits of the communication are transferred transparently (from the point of view of the attached devices). Third, the control unit must tear down the connection, either in response to a request from one of the parties or for its own reasons.

An important characteristic of a circuit-switching device is whether it is blocking or nonblocking. Blocking occurs when the network is unable to connect two stations because all possible paths between them are already in use. A blocking network is one in which such blocking is possible. Hence a nonblocking network permits all stations to be connected (in pairs) at once and grants all possible connection requests as long as the called party is free. When a network is supporting only voice traffic, a blocking configuration is generally acceptable, because it is expected that most phone calls are of short duration and that therefore only a fraction of the telephones will be engaged at any time. However, when data processing devices are involved, these assumptions may be invalid. For example, for a data entry application, a terminal may be continuously connected to a computer for hours at a time. Hence, for data applications, there is a requirement for a nonblocking or "nearly nonblocking" (very low probability of blocking) configuration.

We turn now to an examination of the switching techniques internal to a single circuit-switching node.

Space Division Switching

Space division switching was originally developed for the analog environment and has been carried over into the digital realm. The fundamental principles are the same, whether the switch is used to carry analog or digital signals. As its name implies, a space division switch is one in which the signal paths are physically separate from one another (divided in space). Each connection requires the establishment of a physical path through the switch that is dedicated solely to the transfer of signals between the two endpoints. The basic building block of the switch is a metallic crosspoint or semiconductor gate that can be enabled and disabled by a control unit.

EXAMPLE 9.2 Figure 9.5 shows a simple **crossbar matrix** with 10 full-duplex I/O lines. The matrix has 10 inputs and 10 outputs; each station attaches to the matrix via one input and one output line. Interconnection is possible between any two lines by enabling the appropriate crosspoint. Note that a total of 100 crosspoints is required.

The crossbar switch has a number of limitations:

- The number of crosspoints grows with the square of the number of attached stations. This is costly for a large switch.
- The loss of a crosspoint prevents connection between the two devices whose lines intersect at that crosspoint.
- The crosspoints are inefficiently utilized; even when all of the attached devices are active, only a small fraction of the crosspoints are engaged.

To overcome these limitations, multiple-stage switches are employed.

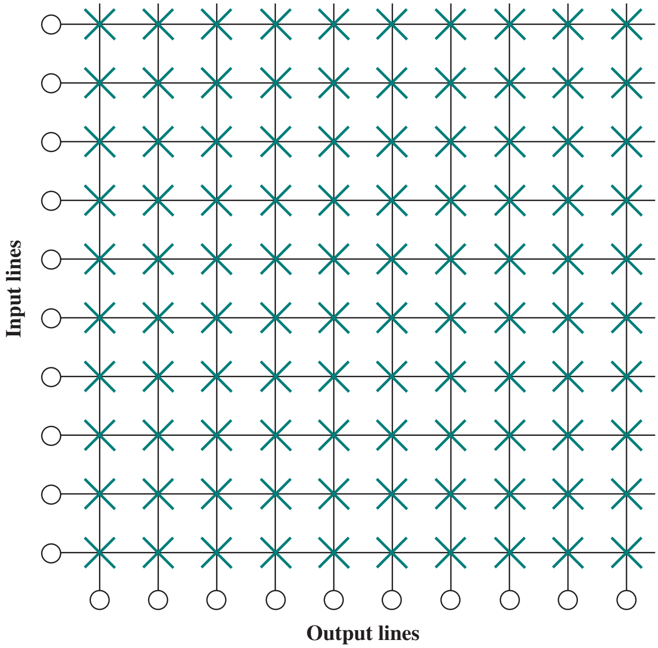


Figure 9.5 Space Division Switch

EXAMPLE 9.3 Figure 9.6 is an example of a three-stage switch.

A multiple-stage switch has two advantages over a single-stage crossbar matrix:

- The number of crosspoints is reduced, increasing crossbar utilization. In Examples 9.2 and 9.3, the total number of crosspoints for 10 stations is reduced from 100 to 48.
- There is more than one path through the network to connect two endpoints, increasing reliability.

Of course, a multistage network requires a more complex control scheme. To establish a path in a single-stage network, it is only necessary to enable a single gate. In a multistage network, a free path through the stages must be determined and the appropriate gates enabled.

A consideration with a multistage space division switch is that it may be blocking. It should be clear from Figure 9.5 that a single-stage crossbar matrix is nonblocking; that is, a path is always available to connect an input to an output. That this may not be the case with a multiple-stage switch can be seen in Figure 9.6. The heavier lines indicate the lines that are already in use. In this state, input line 10, for example, cannot be connected to output line 3, 4, or 5, even though all of these output lines are available. A multiple-stage switch can be made nonblocking

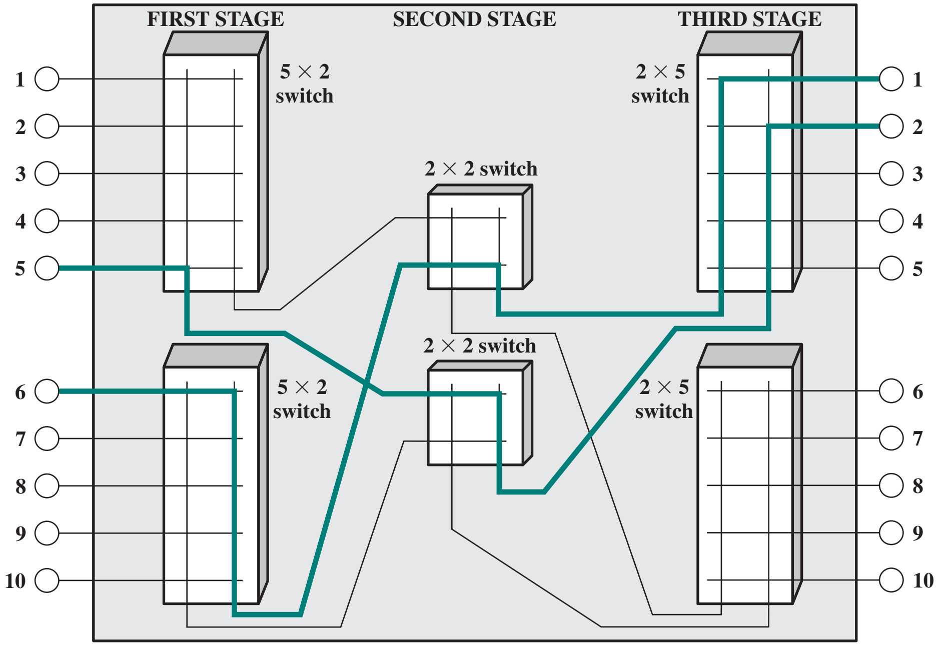


Figure 9.6 Three-Stage Space Division Switch

by increasing the number or size of the intermediate switches, but of course this increases the cost.

Time-Division Switching

The technology of switching has a long history, most of it covering an era when analog signal switching predominated. With the advent of digitized voice and synchronous TDM techniques, both voice and data can be transmitted via digital signals. This has led to a fundamental change in the design and technology of switching systems. Instead of the relatively dumb space division approach, modern digital systems rely on intelligent control of space and time-division elements.

Virtually all modern circuit switches use digital time-division techniques for establishing and maintaining "circuits." Time-division switching involves the partitioning of a lower-speed bit stream into pieces that share a higher-speed stream with other bit streams. The individual pieces, or slots, are manipulated by control logic to route data from input to output.

Time-Slot Interchange

The basic building block of many time-division switches is the time-slot interchange (TSI) mechanism. A TSI unit operates on a synchronous TDM stream of time slots, or channels, by interchanging pairs of slots to achieve a full-duplex operation. Figure 9.7a shows how the input line of device *I* is connected to the output line of device *J*, and vice versa.

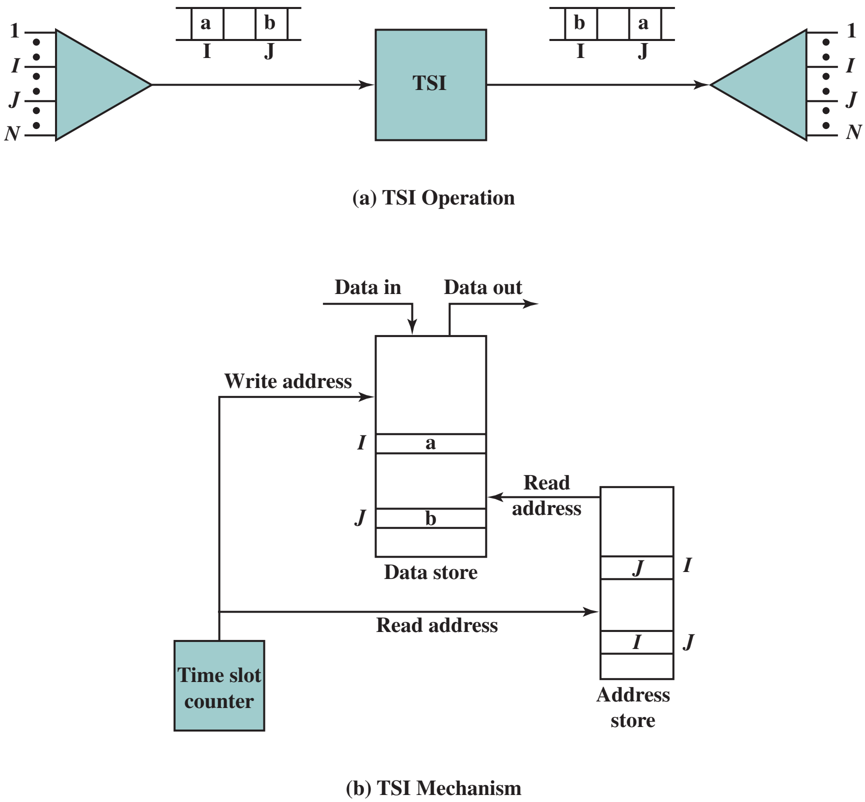


Figure 9.7 Time-Slot Interchange

The input lines of N devices are passed through a synchronous time-division multiplexer to produce a TDM stream with N slots. To allow the interchange of any two slots, to create a full-duplex connection, the incoming data in a slot must be stored until the data can be sent out on the correct channel in the next TDM frame cycle. Hence, the TSI introduces a delay and produces output slots in the desired order. The output stream of slots is then demultiplexed and routed to the appropriate output line. Because each channel is provided a time slot in each TDM frame, whether or not it transmits data, the size of the TSI unit must be chosen for the capacity of the line, not for the actual data rate.

Figure 9.7b shows a mechanism for implementing TSI. A random access data store whose width equals one time slot of data and whose length equals the number of slots in a frame is used. An incoming TDM frame is written sequentially, slot by slot, into the data store. An outgoing data frame is created by reading slots from the memory in an order dictated by an address store that reflects the existing connections. In the figure, the data in channels I and J are interchanged, creating a full-duplex connection between the corresponding stations.

TSI is a simple, effective way to switch TDM data. However, the size of such a switch, in terms of the number of connections, is limited by the amount of latency that can be tolerated. The greater the number of channels, the greater the average delay that each channel experiences.

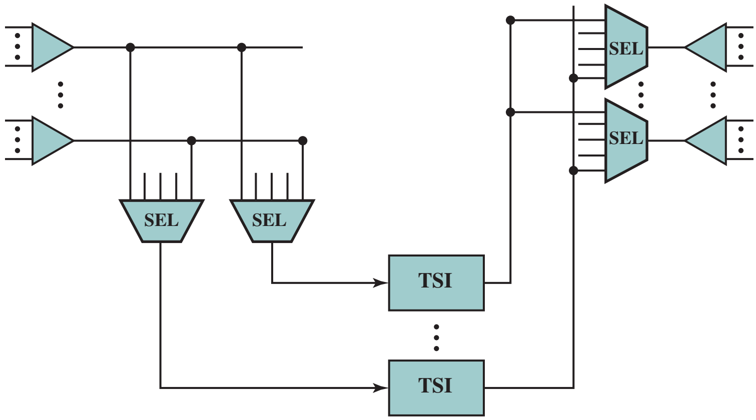


Figure 9.8 A Time-Multiplexed Switch

Time-Multiplexed Switching

To overcome the latency problems of TSI, contemporary time-division switches use multiple TSI units, each of which carries a portion of the total traffic. To connect two channels entering a single TSI unit, their time slots can be interchanged, as just described. However, to connect a channel on one TDM stream (going into one TSI) to a channel on another TDM stream (going into another TSI), some form of space division switching is needed. Naturally, we do not wish to switch all of the time slots from one TDM stream to another; we would like to do it one slot at a time. This technique is known as **time-multiplexed switching (TMS)**.

One means of implementing a TMS switch is the crossbar switch discussed earlier. This requires the crosspoints to be manipulated at each time slot. More commonly, the TMS switch is implemented using digital selectors. The selector (SEL) device selects an input line based on a channel assignment provided from a store controlled by a time-slot counter.

To reduce or eliminate blocking, multiple stage networks can be built by concatenating TMS (S) and TSI (T) stages. Systems are generally described by an enumeration of their stages from input to output using the symbols T and S. Figure 9.8 shows an example of a three-stage switch implemented with SEL units.

9.4 SOFTSWITCH ARCHITECTURE

The latest trend in the development of circuit-switching technology is generally referred to as the softswitch. In essence, a softswitch is a general-purpose computer running specialized software that turns it into a smart phone switch. Softswitches cost significantly less than traditional circuit switches and can provide more functionality. In particular, in addition to handling the traditional circuit-switching functions, a softswitch can convert a stream of digitized voice bits into packets. This opens up a number of options for transmission, including the increasingly popular voice over IP (Internet Protocol) approach.

In any telephone network switch, the most complex element is the software that controls call processing. This software performs call routing and implements call-processing logic for hundreds of custom-calling features. Typically, this software runs on a proprietary processor that is integrated with the physical circuit-switching hardware. A more flexible approach is to physically separate the call-processing function from the hardware-switching function. In softswitch terminology, the physical-switching function is performed by a **media gateway (MG)** and the call-processing logic resides in a **media gateway controller (MGC)**.

Figure 9.9 contrasts the architecture of a traditional telephone network circuit switch with the softswitch architecture. In the latter case, the MG and MGC are distinct entities and may be provided by different vendors. To facilitate interoperability, ITU-T has issued a standard for a media gateway control protocol between the MG and MGC: H.248.1 (*Gateway Control Protocol, Version 3*, 2005). RFC 2805 (*Media Gateway Control Protocol Architecture and Requirements*, 2000) provides an overview of media gateway concepts.

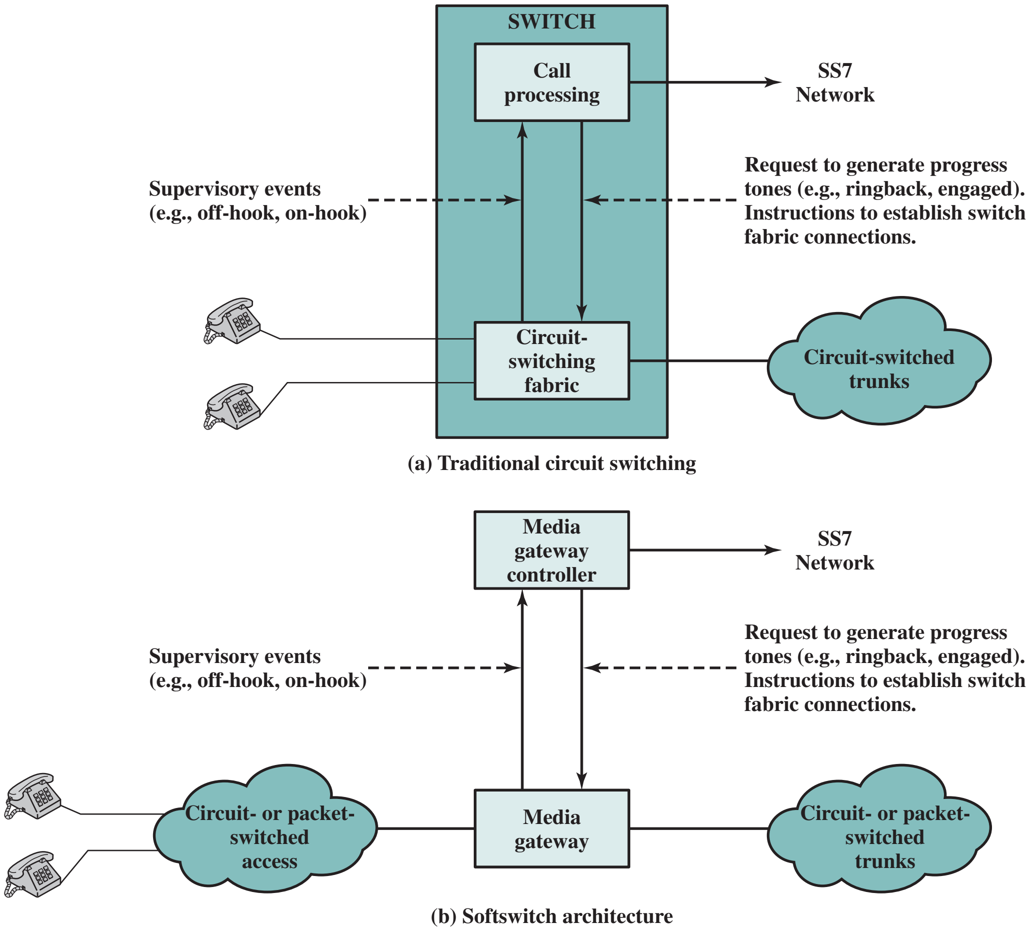


Figure 9.9 Comparison between Traditional Circuit Switching and Softswitch

9.5 PACKET-SWITCHING PRINCIPLES

The long-haul circuit-switching telecommunications network was originally designed to handle voice traffic, and the majority of traffic on these networks continues to be voice. A key characteristic of circuit-switching networks is that resources within the network are dedicated to a particular call. For voice connections, the resulting circuit will enjoy a high percentage of utilization because, most of the time, one party or the other is talking. However, as the circuit-switching network began to be used increasingly for data connections, two shortcomings became apparent:

- In a typical user/host data connection (e.g., personal computer user logged on to a database server), much of the time the line is idle. Thus, with data connections, a circuit-switching approach is inefficient.
- In a circuit-switching network, the connection provides for transmission at a constant data rate. Thus, each of the two devices that are connected must transmit and receive at the same data rate as the other. This limits the utility of the network in interconnecting a variety of host computers and workstations.

To understand how packet switching addresses these problems, let us briefly summarize packet-switching operation. Data are transmitted in short packets. A typical upper bound on packet length is 1000 octets (bytes). If a source has a longer message to send, the message is broken up into a series of packets (Figure 9.10). Each packet contains a portion (or all for a short message) of the user's data plus some control information. The control information, at a minimum, includes the information that the network requires to be able to route the packet through the network and deliver it to the intended destination. At each node en route, the packet is received, stored briefly, and passed on to the next node.

Let us return to Figure 9.1, but now assume that it depicts a simple packet-switching network. Consider a packet to be sent from station A to station E. The packet includes control information that indicates that the intended destination is E. The packet is sent from A to node 4. Node 4 stores the packet, determines the

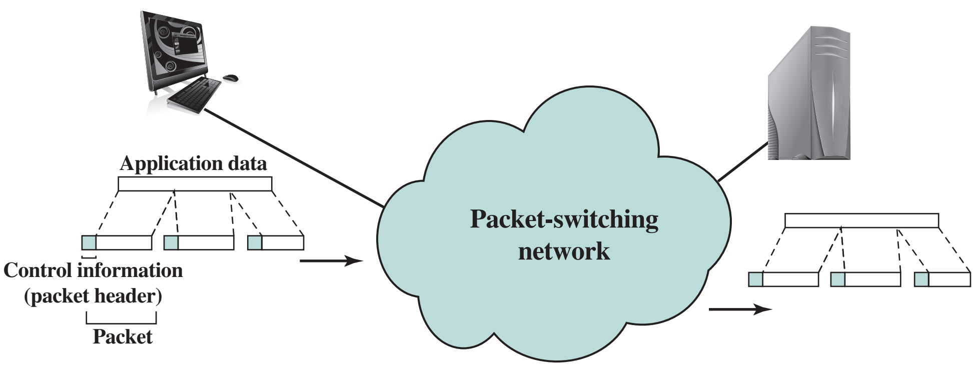


Figure 9.10 The Use of Packets

next leg of the route (say 5), and queues the packet to go out on that link (the 4-5 link). When the link is available, the packet is transmitted to node 5, which forwards the packet to node 6, and finally to E. This approach has a number of advantages over circuit switching:

- Line efficiency is greater, because a single node-to-node link can be dynamically shared by many packets over time. The packets are queued up and transmitted as rapidly as possible over the link. By contrast, with circuit switching, time on a node-to-node link is preallocated using synchronous time-division multiplexing. Much of the time, such a link may be idle because a portion of its time is dedicated to a connection that is idle.
- A packet-switching network can perform data-rate conversion. Two stations of different data rates can exchange packets because each connects to its node at its proper data rate.
- When traffic becomes heavy on a circuit-switching network, some calls are blocked; that is, the network refuses to accept additional connection requests until the load on the network decreases. On a packet-switching network, packets are still accepted, but delivery delay increases.
- Priorities can be used. If a node has a number of packets queued for transmission, it can transmit the higher-priority packets first. These packets will therefore experience less delay than lower-priority packets.

Switching Technique

If a station has a message to send through a packet-switching network that is of length greater than the maximum packet size, it breaks the message up into packets and sends these packets, one at a time, to the network. A question arises as to how the network will handle this stream of packets as it attempts to route them through the network and deliver them to the intended destination. Two approaches are used in contemporary networks: datagram and virtual circuit.

In the **datagram** approach, each packet is treated independently, with no reference to packets that have gone before. This approach is illustrated in Figure 9.11, which shows a time sequence of snapshots of the progress of three packets through the network. Each node chooses the next node on a packet's path, taking into account information received from neighboring nodes on traffic, line failures, and so on. So the packets, each with the same destination address, do not all follow the same route, and they may arrive out of sequence at the exit point. In this example, the exit node restores the packets to their original order before delivering them to the destination. In some datagram networks, it is up to the destination rather than the exit node to do the reordering. Also, it is possible for a packet to be destroyed in the network. For example, if a packet-switching node crashes momentarily, all of its queued packets may be lost. Again, it is up to either the exit node or the destination to detect the loss of a packet and decide how to recover it. In this technique, each packet, treated independently, is referred to as a datagram.

In the **virtual circuit** approach, a preplanned route is established before any packets are sent. Once the route is established, all the packets between a pair of communicating parties follow this same route through the network. This is illustrated in

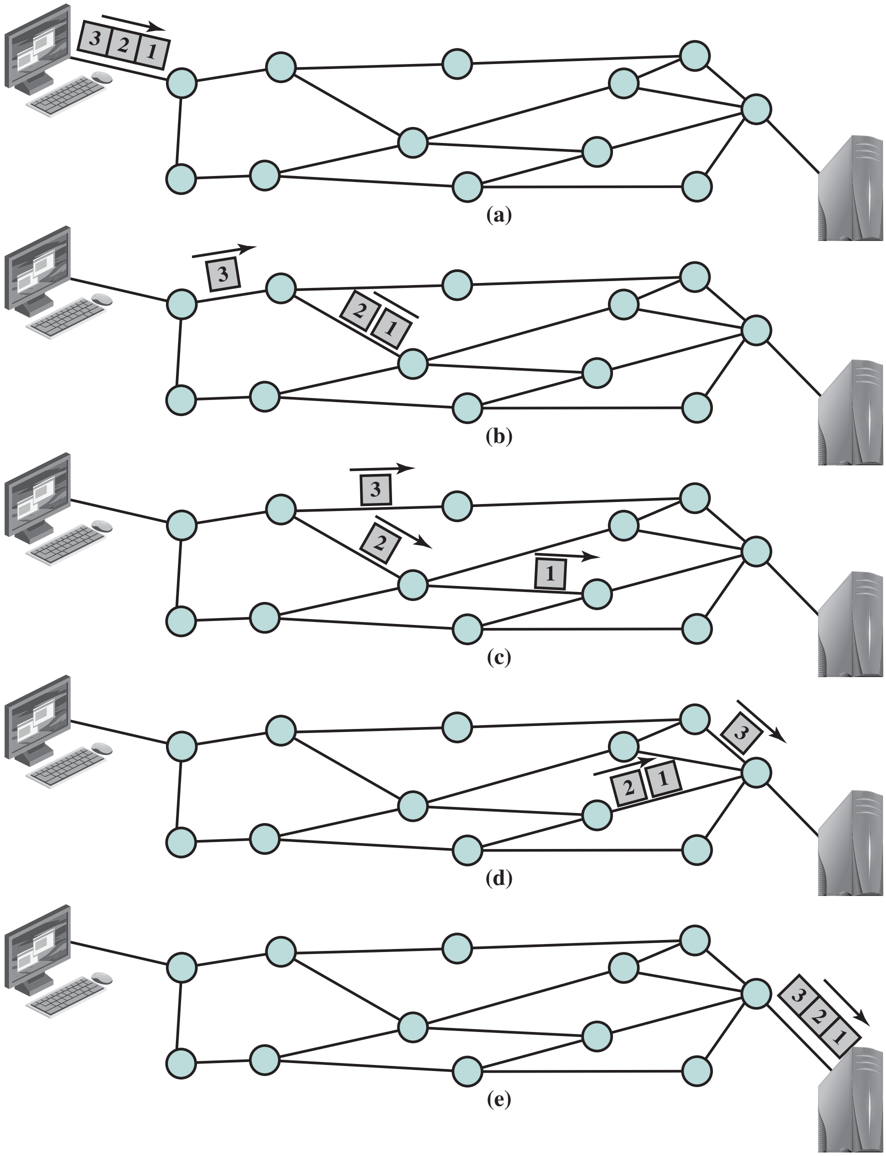


Figure 9.11 Packet Switching: Datagram Approach

Figure 9.12. Because the route is fixed for the duration of the logical connection, it is somewhat similar to a circuit in a circuit-switching network and is referred to as a virtual circuit. Each packet contains a virtual circuit identifier as well as data. Each node on the preestablished route knows where to direct such packets; no routing decisions are required. At any time, each station can have more than one virtual circuit to any other station and can have virtual circuits to more than one station.

So the main characteristic of the virtual circuit technique is that a route between stations is set up prior to data transfer. Note that this does not mean that this is a dedicated path, as in circuit switching. A transmitted packet is buffered at

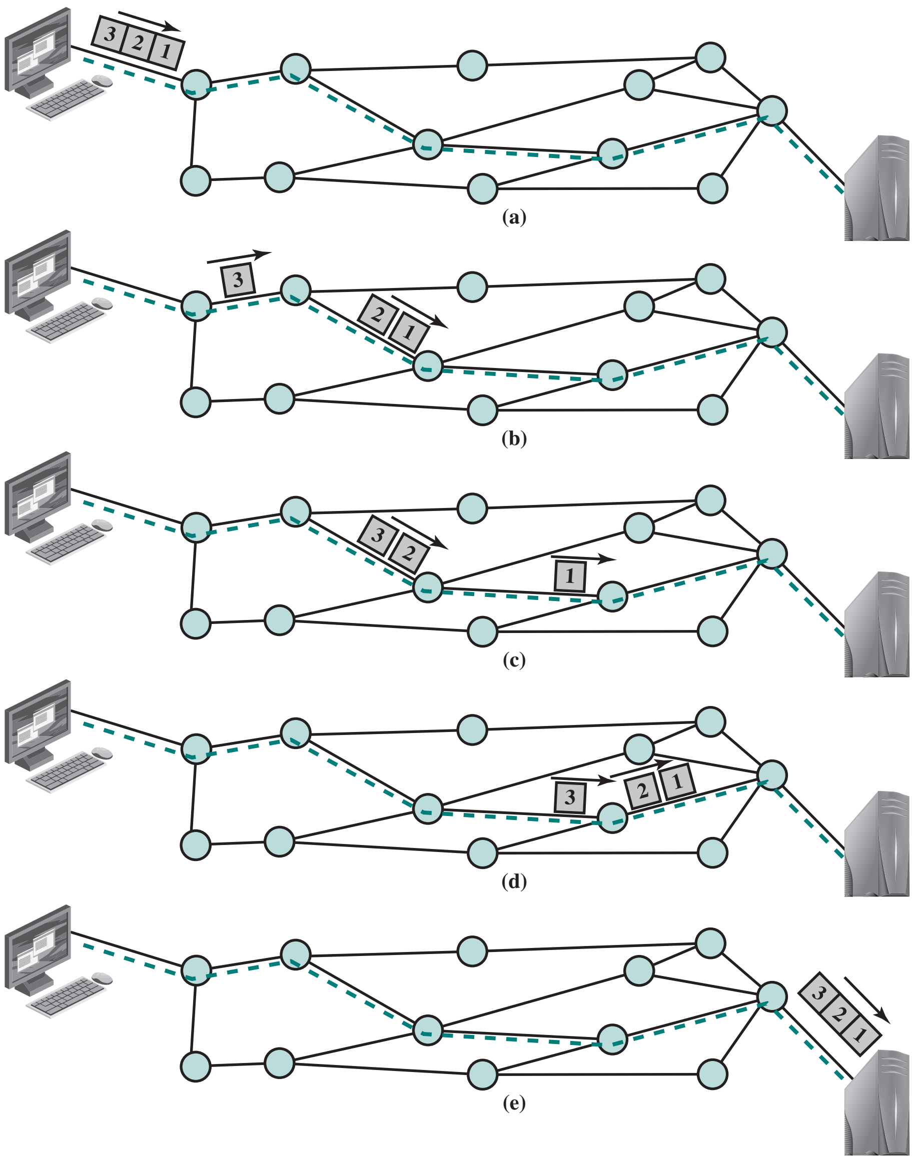


Figure 9.12 Packet Switching: Virtual-Circuit Approach

each node, and queued for output over a line, while other packets on other virtual circuits may share the use of the line. The difference from the datagram approach is that, with virtual circuits, the node need not make a routing decision for each packet. It is made only once for all packets using that virtual circuit.

If two stations wish to exchange data over an extended period of time, there are certain advantages to virtual circuits. First, the network may provide services related to the virtual circuit, including sequencing and error control. Sequencing refers to the fact that, because all packets follow the same route, they arrive in the original order. Error control is a service that assures not only that packets arrive in proper sequence, but also that all packets arrive correctly. For example, if a packet in a sequence from node 4 to node 6 fails to arrive at node 6, or arrives with an error, node 6 can request a retransmission of that packet from node 4. Another advantage is that packets should transit the network more rapidly with a virtual circuit; it is not necessary to make a routing decision for each packet at each node.

One advantage of the datagram approach is that the call setup phase is avoided. Thus, if a station wishes to send only one or a few packets, datagram delivery will be quicker. Another advantage of the datagram service is that, because it is more primitive, it is more flexible. For example, if congestion develops in one part of the network, incoming datagrams can be routed away from the congestion. With the use of virtual circuits, packets follow a predefined route, and thus it is more difficult for the network to adapt to congestion. A third advantage is that datagram delivery is inherently more reliable. With the use of virtual circuits, if a node fails, all virtual circuits that pass through that node are lost. With datagram delivery, if a node fails, subsequent packets may find an alternate route that bypasses that node. A datagram-style of operation is common in internetworks, discussed in Part Five.

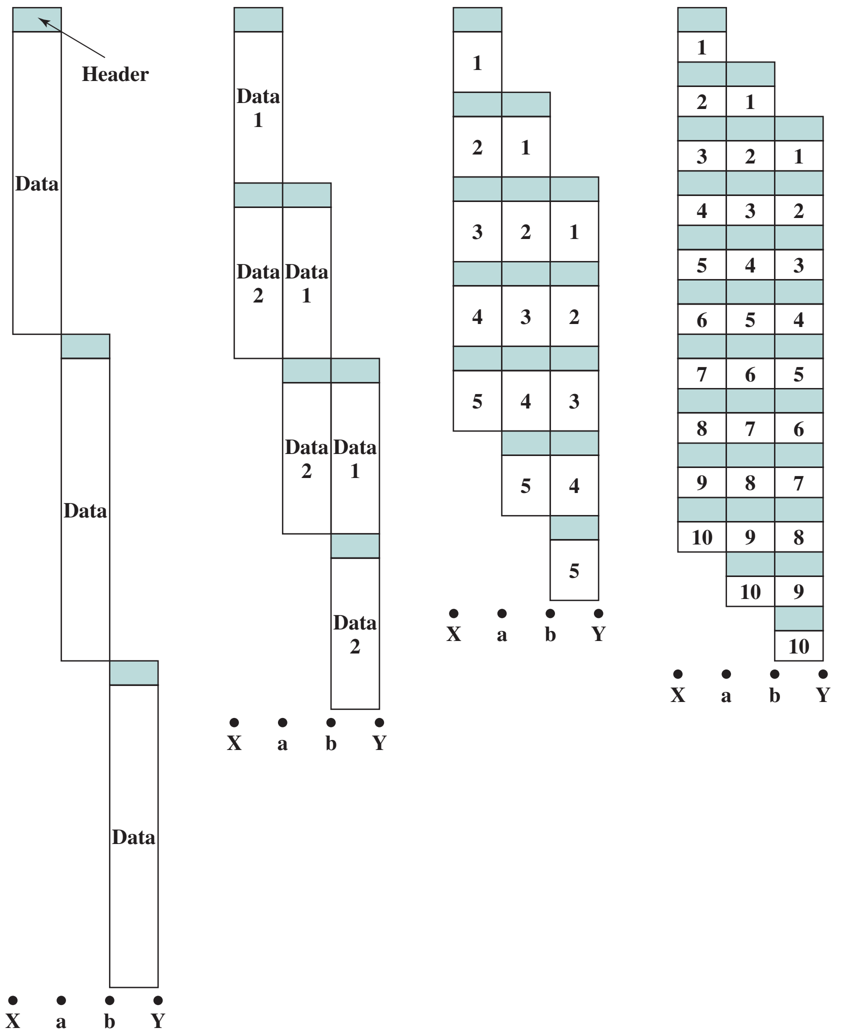
Packet Size

There is a significant relationship between packet size and transmission time, as shown in Figure 9.13. In this example, it is assumed that there is a virtual circuit from station X through nodes a and b to station Y. The message to be sent comprises 40 octets, and each packet contains 3 octets of control information, which is placed at the beginning of each packet and is referred to as a header. If the entire message is sent as a single packet of 43 octets (3 octets of header plus 40 octets of data), then the packet is first transmitted from station X to node a (Figure 9.13a). When the entire packet is received, it can then be transmitted from a to b. When the entire packet is received at node b, it is then transferred to station Y. Ignoring switching time, total transmission time is 129 octet-times (43 octets × 3 packet transmissions).

Suppose now that we break up the message into two packets, each containing 20 octets of the message and, of course, 3 octets each of header, or control information. In this case, node a can begin transmitting the first packet as soon as it has arrived from X, without waiting for the second packet. Because of this overlap in transmission, the total transmission time drops to 92 octet-times. By breaking the message up into five packets, each intermediate node can begin transmission even sooner and the savings in time is greater, with a total of 77 octet-times for transmission. However, this process of using more and smaller packets eventually results in increased, rather than reduced, delay as illustrated in Figure 9.13d. This is because each packet contains a fixed amount of header, and more packets mean more of these headers. Furthermore, the example does not show the processing and queuing delays at each node. These delays are also greater when more packets are handled for a single message. However, we shall see in the next chapter that an extremely small packet size (53 octets) can result in an efficient network design.

External Network Interface

One technical aspect of packet-switching networks remains to be examined: the interface between attached devices and the network. We have seen that a circuit-switching



(a) 1-packet message (b) 2-packet message (c) 5-packet message (d) 10-packet message

Figure 9.13 Effect of Packet Size on Transmission Time

network provides a transparent communications path for attached devices that makes it appear that the two communicating stations have a direct link. However, in the case of packet-switching networks, the attached stations must organize their data into packets for transmission. This requires a certain level of cooperation between the network and the attached stations. This cooperation is embodied in an interface standard. The standard used for traditional packet-switching networks is X.25, which is described in Appendix U. Another interface standard is frame relay, also discussed in Appendix U.

Typically, standards for packet-switching network interfaces define a virtual circuit service. This service enables any subscriber to the network to set up logical connections, called virtual circuits, to other subscribers. An example is shown in Figure 9.14 (compare Figure 9.1). In this example, station A has a virtual circuit

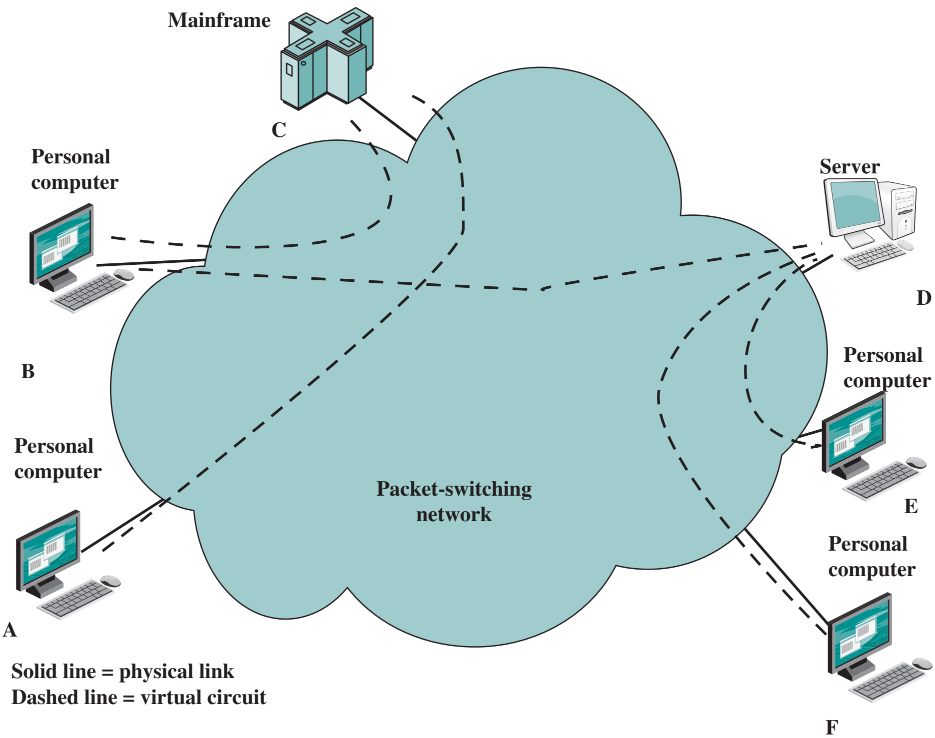


Figure 9.14 The Use of Virtual Circuits

connection to C; station B has two virtual circuits established, one to C and one to D; and stations E and F each have a virtual circuit connection to D.

In this context, the term *virtual circuit* refers to the logical connection between two stations through the network; this is perhaps best termed an **external virtual circuit**. Earlier, we used the term *virtual circuit* to refer to a specific preplanned route through the network between two stations; this could be called an **internal virtual circuit**. Typically, there is a one-to-one relationship between external and internal virtual circuits. However, it is also possible to employ an external virtual circuit service with a datagram-style network. What is important for an external virtual circuit is that there is a logical relationship, or logical channel, established between two stations, and all of the data associated with that logical channel are considered as part of a single stream of data between the two stations. For example, in Figure 9.14, station D keeps track of data packets arriving from three different workstations (B, E, F) on the basis of the virtual circuit number associated with each incoming packet.

Comparison of Circuit Switching and Packet Switching

Having looked at the internal operation of packet switching, we can now return to a comparison of this technique with circuit switching. We first look at the important issue of performance and then examine other characteristics.

PERFORMANCE A simple comparison of circuit switching and the two forms of packet switching is provided in Figure 9.15. The figure depicts the transmission of a

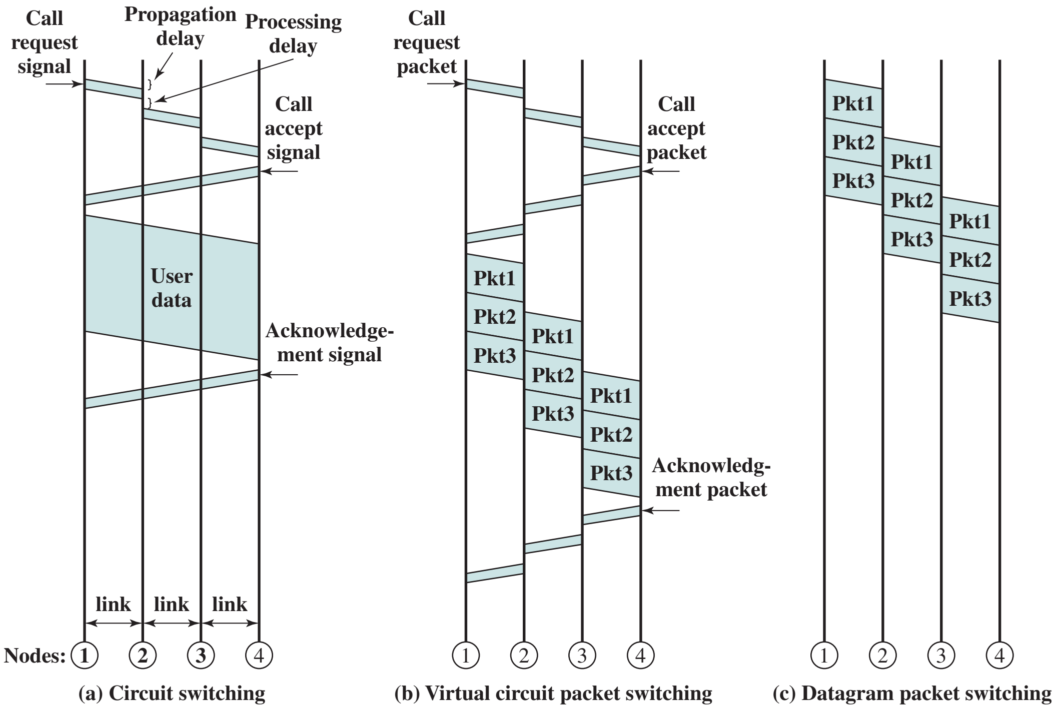


Figure 9.15 Event Timing for Circuit Switching and Packet Switching

message across four nodes, from a source station attached to node 1 to a destination station attached to node 4. In this figure, we are concerned with three types of delay:

- **Propagation delay:** The time it takes a signal to propagate from one node to the next. This time is generally negligible. The speed of electromagnetic signals through a wire medium, for example, is typically 2×10^8 m/s.
- **Transmission time:** The time it takes for a transmitter to send out a block of data. For example, it takes 1 s to transmit a 10,000-bit block of data onto a 10-kbps line.
- **Node delay:** The time it takes for a node to perform the necessary processing as it switches data.

For circuit switching, there is a certain amount of delay before the message can be sent. First, a Call Request signal is sent through the network, to set up a connection to the destination. If the destination station is not busy, a Call Accepted signal returns. Note that a processing delay is incurred at each node during the call request; this time is spent at each node setting up the route of the connection. On the return, this processing is not needed because the connection is already set up. After the connection is set up, the message is sent as a single block, with no noticeable delay at the switching nodes.

Virtual circuit packet switching appears quite similar to circuit switching. A virtual circuit is requested using a Call Request packet, which incurs a delay at each node. The virtual circuit is accepted with a Call Accept packet. In contrast to the circuit-switching case, the call acceptance also experiences node delays, even though the virtual circuit route is now established. The reason is that this packet is queued at each node and must wait its turn for transmission. Once the virtual circuit is established, the message is transmitted in packets. It should be clear that this phase of the operation can be no faster than circuit switching for comparable networks. This is because circuit switching is an essentially transparent process, providing a constant data rate across the network. Packet switching involves some delay at each node in the path. Worse, this delay is variable and will increase with increased load.

Datagram packet switching does not require a call setup. Thus, for short messages, it will be faster than virtual circuit packet switching and perhaps circuit switching. However, because each individual datagram is routed independently, the processing for each datagram at each node may be longer than for virtual circuit packets. Thus, for long messages, the virtual circuit technique may be superior.

Figure 9.15 is intended only to suggest what the relative performance of the techniques might be; actual performance depends on a host of factors, including the size of the network, its topology, the pattern of load, and the characteristics of typical exchanges.

Other Characteristics Besides performance, there are a number of other characteristics that may be considered in comparing the techniques we have been discussing. Table 9.1 summarizes the most important of these. Most of these characteristics have already been discussed. A few additional comments follow.

Table 9.1 Comparison of Communication Switching Techniques

Circuit Switching	**Datagram Packet Switching**	**Virtual Circuit Packet Switching**
Dedicated transmission path	No dedicated path	No dedicated path
Continuous transmission of data	Transmission of packets	Transmission of packets
Fast enough for interactive	Fast enough for interactive	Fast enough for interactive
Messages are not stored	Packets may be stored until delivered	Packets stored until delivered
The path is established for entire conversation	Route established for each packet	Route established for entire conversation
Call setup delay; negligible transmission delay	Packet transmission delay	Call setup delay; packet transmission delay
Busy signal if called party busy	Sender may be notified if packet not delivered	Sender notified of connection denial
Overload may block call setup; no delay for established calls	Overload increases packet delay	Overload may block call setup; increases packet delay
Electromechanical or computerized switching nodes	Small switching nodes	Small switching nodes
User responsible for message loss protection	Network may be responsible for individual packets	Network may be responsible for packet sequences
Usually no speed or code conversion	Speed and code conversion	Speed and code conversion
Fixed bandwidth	Dynamic use of bandwidth	Dynamic use of bandwidth
No overhead bits after call setup	Overhead bits in each packet	Overhead bits in each packet

As was mentioned, circuit switching is essentially a transparent service. Once a connection is established, a constant data rate is provided to the connected stations. This is not the case with packet switching, which typically introduces variable delay, so that data arrive in a choppy manner. Indeed, with datagram packet switching, data may arrive in a different order than they were transmitted.

An additional consequence of transparency is that there is no overhead required to accommodate circuit switching. Once a connection is established, the analog or digital data are passed through, as is, from source to destination. For packet switching, analog data must be converted to digital before transmission; in addition, each packet includes overhead bits, such as the destination address.

9.6 ASYNCHRONOUS TRANSFER MODE

Asynchronous transfer mode is a switching and multiplexing technology that employs small, fixed-length packets called **cells**. A fixed-size packet was chosen to ensure that the switching and multiplexing function could be carried out efficiently, with little delay variation. A small cell size was chosen primarily to support delay-intolerant interactive voice service with a small packetization delay. ATM is a connection-oriented packet-switching technology that was designed to provide the performance of a circuit-switching network and the flexibility and efficiency of a packet-switching network. A major thrust of the ATM standardization effort was to provide a powerful set of tools for supporting a rich QoS capability and a powerful traffic management capability. ATM was intended to provide a unified networking standard for both circuit-switched and packet-switched traffic, and to support data, voice, and video with appropriate QoS mechanisms. With ATM, the user can select the desired level of service and obtain guaranteed service quality. Internally, the ATM network makes reservations and preplans routes so that transmission allocation is based on priority and QoS characteristics.

ATM was intended to be a universal networking technology, with much of the switching and routing capability implemented in hardware, and with the ability to support IP-based networks and circuit-switched networks. It was also anticipated that ATM would be used to implement local area networks. ATM never achieved this comprehensive deployment. However, ATM remains an important technology. ATM is commonly used by telecommunications providers to implement wide area networks. Many DSL implementations use ATM over the basic DSL hardware for multiplexing and switching, and ATM is used as a backbone network technology in numerous IP networks and portions of the Internet.

A number of factors have led to this lesser role for ATM. IP, with its many associated protocols, provides an integrative technology that is more scalable and less complex than ATM. In addition, the need to use small fixed-sized cells to reduce jitter has disappeared as transport speeds have increased. The development of voice and video over IP protocols has provided an integration capability at the IP level.

Perhaps the most significant development related to the reduced role for ATM is the widespread acceptance of Multiprotocol Label Switching (MPLS). MPLS is a layer-2 connection-oriented packet-switching protocol that, as the name suggests, can provide a switching service for a variety of protocols and applications, including IP, voice, and video. We introduce MPLS in Chapter 23.

ATM Logical Connections

ATM is a packet-oriented transfer mode. It allows multiple logical connections to be multiplexed over a single physical interface. The information flow on each logical connection is organized into fixed-size packets called cells. Logical connections in ATM are referred to as **virtual channel connections (VCCs)**. A VCC is analogous to a virtual circuit; it is the basic unit of switching in an ATM network. A VCC is set up between two end users through the network, and a variable-rate, full-duplex flow of fixed-size cells is exchanged over the connection. VCCs are also used for user–network exchange (control signaling) and network–network exchange (network management and routing).

For ATM, a second sublayer of processing has been introduced that deals with the concept of virtual path (Figure 9.16). A **virtual path connection (VPC)** is a bundle of VCCs that have the same endpoints. Thus, all of the cells flowing over all of the VCCs in a single VPC are switched together.

The virtual path concept was developed in response to a trend in high-speed networking in which the control cost of the network is becoming an increasingly higher proportion of the overall network cost. The virtual path technique helps contain the control cost by grouping connections sharing common paths through the network into a single unit. Network management actions can then be applied to a small number of groups of connections instead of a large number of individual connections.

Several advantages can be listed for the use of virtual paths:

- **Simplified network architecture:** Network transport functions can be separated into those related to an individual logical connection (virtual channel) and those related to a group of logical connections (virtual path).
- **Increased network performance and reliability:** The network deals with fewer, aggregated entities.
- **Reduced processing and short connection setup time:** Much of the work is done when the virtual path is set up. By reserving capacity on a virtual path connection in anticipation of later call arrivals, new virtual channel connections can be established by executing simple control functions at the endpoints

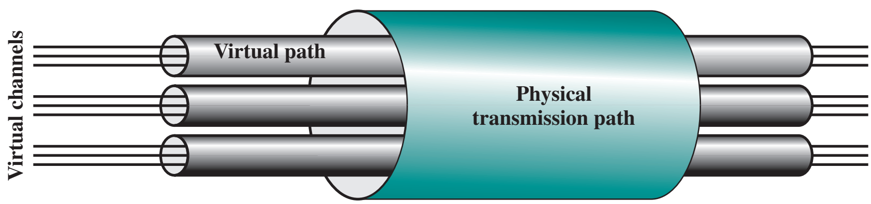


Figure 9.16 ATM Connection Relationships

of the virtual path connection; no call processing is required at transit nodes. Thus, the addition of new virtual channels to an existing virtual path involves minimal processing.

- **Enhanced network services:** The virtual path is used internal to the network but is also visible to the end user. Thus, the user may define closed user groups or closed networks of virtual channel bundles.

Virtual Path/Virtual Channel Characteristics ITU-T Recommendation I.150 lists the following as characteristics of virtual channel connections:

- **Quality of service (QoS):** A user of a VCC is provided with a QoS specified by parameters such as cell loss ratio (ratio of cells lost to cells transmitted) and cell delay variation.
- **Switched and semipermanent virtual channel connections:** A switched VCC is an on-demand connection, which requires a call control signaling for setup and tearing down. A semipermanent VCC is one that is of long duration and is set up by configuration or network management action.
- **Cell sequence integrity:** The sequence of transmitted cells within a VCC is preserved.
- **Traffic parameter negotiation and usage monitoring:** Traffic parameters can be negotiated between a user and the network for each VCC. The network monitors the input of cells to the VCC to ensure that the negotiated parameters are not violated.

The types of traffic parameters that can be negotiated include average rate, peak rate, burstiness, and peak duration. The network may need a number of strategies to deal with congestion and to manage existing and requested VCCs. At the crudest level, the network may simply deny new requests for VCCs to prevent congestion. Additionally, cells may be discarded if negotiated parameters are violated or if congestion becomes severe. In an extreme situation, existing connections might be terminated.

I.150 also lists characteristics of VPCs. The first four characteristics listed are identical to those for VCCs. That is, QoS; switched and semipermanent VPCs; cell sequence integrity; and traffic parameter negotiation and usage monitoring are all also characteristics of a VPC. There are a number of reasons for this duplication. First, this provides some flexibility in how the network service manages the requirements placed upon it. Second, the network must be concerned with the overall requirements for a VPC, and within a VPC may negotiate the establishment of virtual channels with given characteristics. Finally, once a VPC is set up, it is possible for the end users to negotiate the creation of new VCCs. The VPC characteristics impose a discipline on the choices that the end users may make.

In addition, a fifth characteristic is listed for VPCs:

- **Virtual channel identifier restriction within a VPC:** One or more virtual channel identifiers, or numbers, may not be available to the user of the VPC but may be reserved for network use. Examples include VCCs used for network management.

Control Signaling In ATM, a mechanism is needed for the establishment and release of VPCs and VCCs. The exchange of information involved in this process is referred to as control signaling and takes place on separate connections from those that are being managed.

For VCCs, I.150 specifies four methods for providing an establishment/release facility. One or a combination of these methods will be used in any particular network:

1. **Semipermanent VCCs** may be used for user-to-user exchange. In this case, no control signaling is required.
2. If there is no preestablished call control signaling channel, then one must be set up. For that purpose, a control signaling exchange must take place between the user and the network on some channel. Hence we need a permanent channel, probably of low data rate, that can be used to set up VCCs that can be used for call control. Such a channel is called a **meta-signaling channel**, as the channel is used to set up signaling channels.
3. The meta-signaling channel can be used to set up a VCC between the user and the network for call control signaling. This **user-to-network signaling virtual channel** can then be used to set up VCCs to carry user data.
4. The meta-signaling channel can also be used to set up a **user-to-user signaling virtual channel**. Such a channel must be set up within a preestablished VPC. It can then be used to allow the two end users, without network intervention, to establish and release user-to-user VCCs to carry user data.

For VPCs, three methods are defined in I.150:

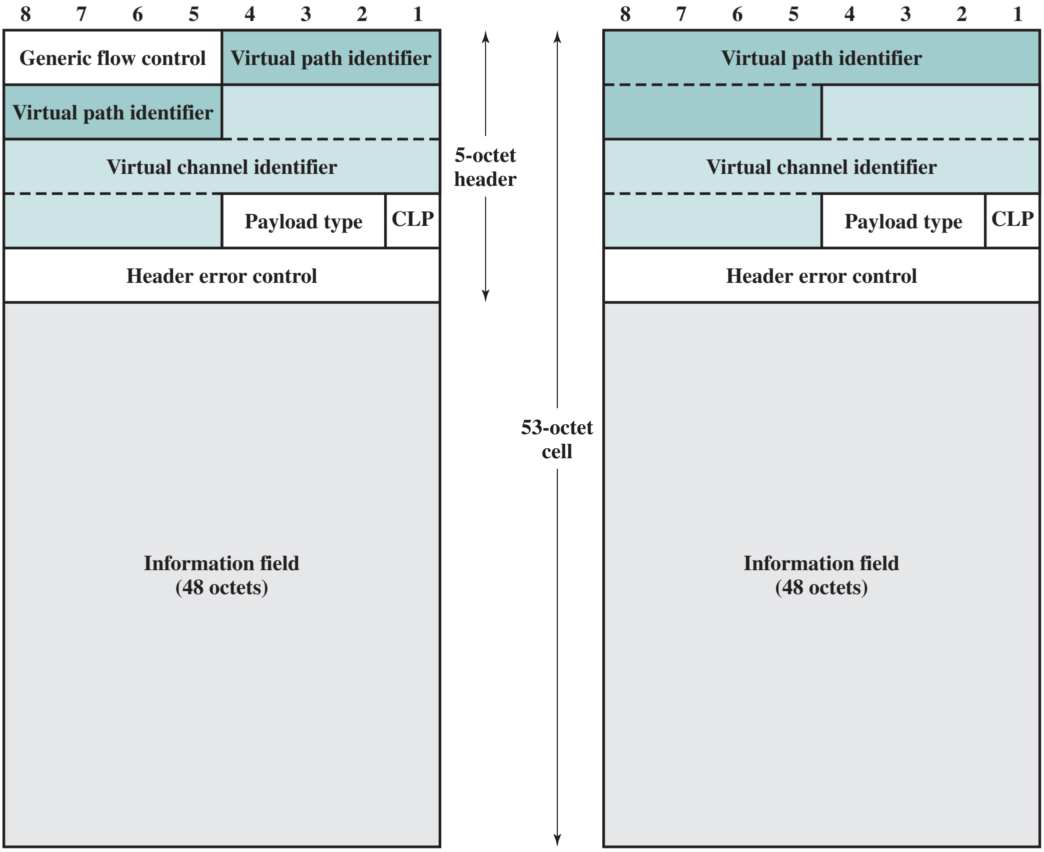
1. A VPC can be established on a **semipermanent** basis by prior agreement. In this case, no control signaling is required.
2. VPC establishment/release may be **customer controlled**. In this case, the customer uses a signaling VCC to request the VPC from the network.
3. VPC establishment/release may be **network controlled**. In this case, the network establishes a VPC for its own convenience. The path may be network-to-network, user-to-network, or user-to-user.

ATM Cells

ATM makes use of fixed-size cells, consisting of a 5-octet header and a 48-octet information field. There are several advantages to the use of small, fixed-size cells. First, the use of small cells may reduce queuing delay for a high-priority cell, because it waits less if it arrives slightly behind a lower-priority cell that has gained access to a resource (e.g., the transmitter). Second, fixed-size cells can be switched more efficiently, which is important for the very high data rates of ATM. With fixed-size cells, it is easier to implement the switching mechanism in hardware.

Figure 9.17a shows the cell header format at the user–network interface. Figure 9.17b shows the cell header format internal to the network.

The **Generic Flow Control (GFC)** field does not appear in the cell header internal to the network, but only at the user–network interface. Hence, it can be used for control of cell flow only at the local user–network interface. The field could



(a) User–network interface (b) Network–network interface

Figure 9.17 ATM Cell Format

be used to assist the customer in controlling the flow of traffic for different qualities of service. In any case, the GFC mechanism is used to alleviate short-term overload conditions in the network.

The **virtual path identifier (VPI)** constitutes a routing field for the network. It is 8 bits at the user–network interface and 12 bits at the network–network interface. The latter allows support for an expanded number of VPCs internal to the network, to include those supporting subscribers and those required for network management. The **virtual channel identifier (VCI)** is used for routing to and from the end user.

The **Payload Type (PT)** field indicates the type of information in the information field. Table 9.2 shows the interpretation of the PT bits. A value of 0 in the first bit indicates user information (i.e., information from the next higher layer). In this case, the second bit indicates whether congestion has been experienced; the third bit, known as the Service Data Unit (SDU) type bit, is a one-bit field that can be used to discriminate two types of ATM SDUs associated with a connection. The term *SDU* refers to the 48-octet payload of the cell. A value of 1 in the first bit of

Table 9.2 Payload Type (PT) Field Coding

PT Coding	Interpretation		
0 0 0	User data cell,	congestion not experienced,	SDU-type = 0
0 0 1	User data cell,	congestion not experienced,	SDU-type = 1
0 1 0	User data cell,	congestion experienced,	SDU-type = 0
0 1 1	User data cell,	congestion experienced,	SDU-type = 1
1 0 0	OAM segment associated cell		
1 0 1	OAM end-to-end associated cell		
1 1 0	Resource management cell		
1 1 1	Reserved for future function		

SDU = Service Data Unit

OAM = Operations, Administration, and Maintenance

the Payload Type field indicates that this cell carries network management or maintenance information. This indication allows the insertion of network-management cells onto a user's VCC without impacting the user's data. Thus, the PT field can provide inband control information.

The **Cell Loss Priority (CLP)** bit is used to provide guidance to the network in the event of congestion. A value of 0 indicates a cell of relatively higher priority, which should not be discarded unless no other alternative is available. A value of 1 indicates that this cell is subject to discard within the network. The user might employ this field so that extra cells (beyond the negotiated rate) may be inserted into the network, with a CLP of 1, and delivered to the destination if the network is not congested. The network may set this field to 1 for any data cell that is in violation of an agreement concerning traffic parameters between the user and the network. In this case, the switch that does the setting realizes that the cell exceeds the agreed traffic parameters but that the switch is capable of handling the cell. At a later point in the network, if congestion is encountered, this cell has been marked for discard in preference to cells that fall within agreed traffic limits.

The **Header Error Control (HEC)** field is an 8-bit error code that can be used to correct single-bit errors in the header and to detect double-bit errors. In the case of most existing data link layer protocols, such as LAPD and HDLC, the data field that serves as input to the error code calculation is in general much longer than the size of the resulting error code. This allows for error detection. In the case of ATM, there is also sufficient redundancy in the code to recover from certain error patterns.

9.7 RECOMMENDED READING

As befits its age, circuit switching has inspired a voluminous literature. Two good books on the subject are [BELL00] and [FREE04].

The literature on packet switching is enormous. [BERT92] is a good treatment of this subject. [ROBE78] is a classic paper on how packet-switching technology evolved. [BARA02] and [HEGG84] are also interesting. [IBM95] provides a detailed treatment of ATM technology.

BARA02 Baran, P. "The Beginnings of Packet Switching: Some Underlying Concepts." *IEEE Communications Magazine*, July 2002.

BELL00 Bellamy, J. *Digital Telephony*. New York: Wiley, 2000.

BERT92 Bertsekas, D., and Gallager, R. *Data Networks*. Englewood Cliffs, NJ: Prentice Hall, 1992.

FREE04 Freeman, R. *Telecommunication System Engineering*. New York: Wiley, 2004.

HEGG84 Heggestad, H. "An Overview of Packet Switching Communications." *IEEE Communications Magazine*, April 1984.

IBM95 IBM International Technical Support Organization. *Asynchronous Transfer Mode (ATM) Technical Overview*. IBM Redbook SG24-4625-00, 1995. www.redbooks.ibm.com

ROBE78 Roberts, L. "The Evolution of Packet Switching." *Proceedings of the IEEE*, November 1978.

[BERT92] is a good treatment of this subject.

9.8 KEY TERMS, REVIEW QUESTIONS, AND PROBLEMS

Key Terms

asynchronous transfer mode (ATM)	header error control (HEC)	time-division switching
cell	internal virtual circuit	time-multiplexed switching (TMS)
circuit switching	local loop	time-slot interchange (TSI)
circuit-switching network	media gateway controller (MGC)	trunk
crossbar matrix	packet switching	virtual channel connection (VCC)
datagram	softswitch	virtual circuit
digital switch	space division switching	virtual path connection (VPC)
exchange	subscriber	
external virtual circuit	subscriber line	
generic flow control (GFC)	subscriber loop	

Review Questions

9.1 Why is it useful to have more than one possible path through a network for each pair of stations?

9.2 What are the four generic architectural components of a public communications network? Define each term.

9.3 What is the principal application that has driven the design of circuit-switching networks?

9.4 What are the advantages of packet switching compared to circuit switching?

9.5 Explain the difference between datagram and virtual circuit operation.

9.6 What is the significance of packet size in a packet-switching network?

9.7 What types of delay are significant in assessing the performance of a packet-switching network?

9.8 What are the characteristics of a virtual channel connection?

9.9 What are the characteristics of a virtual path connection?

9.10 List and briefly explain the fields in an ATM cell.

Problems

9.1 Consider a simple telephone network consisting of two end offices and one intermediate switch with a 1-MHz full-duplex trunk between each end office and the intermediate switch. Assume a 4-kHz channel for each voice call. The average telephone is used to make four calls per 8-hour workday, with a mean call duration of six minutes. Ten percent of the calls are long distance. What is the maximum number of telephones an end office can support?

9.2
- **a.** If a crossbar matrix has n input lines and m output lines, how many crosspoints are required?
- **b.** How many crosspoints would be required if there were no distinction between input and output lines (i.e., if any line could be interconnected to any other line serviced by the crossbar)?
- **c.** Show the minimum configuration.

9.3 Consider a three-stage switch such as in Figure 9.6. Assume that there are a total of N input lines and N output lines for the overall three-stage switch. If n is the number of input lines to a stage 1 crossbar and the number of output lines to a stage 3 crossbar, then there are N/n stage 1 crossbars and N/n stage 3 crossbars. Assume each stage 1 crossbar has one output line going to each stage 2 crossbar, and each stage 2 crossbar has one output line going to each stage 3 crossbar. For such a configuration it can be shown that, for the switch to be nonblocking, the number of stage 2 crossbar matrices must equal $2n - 1$.
- **a.** What is the total number of crosspoints among all the crossbar switches?
- **b.** For a given value of N, the total number of crosspoints depends on the value of n. That is, the value depends on how many crossbars are used in the first stage to handle the total number of input lines. Assuming a large number of input lines to each crossbar (large value of n), what is the minimum number of crosspoints for a nonblocking configuration as a function of n?
- **c.** For a range of N from 10^2 to 10^6, plot the number of crosspoints for a single-stage $N \times N$ switch and an optimum three-stage crossbar switch.

9.4 Consider a TSI system with a TDM input of 8000 frames per second. The TSI requires one memory read and one memory write operation per slot. What is the maximum number of slots per frame that can be handled, as a function of the memory cycle time?

9.5 Consider a TDM system with 8 I/O lines, and connections 1-2, 3-7, and 5-8. Draw several frames of the input to the TSI unit and output from the TSI unit, indicating the movement of data from input time slots to output time slots.

9.6 Explain the flaw in the following reasoning: Packet switching requires control and address bits to be added to each packet. This introduces considerable overhead in packet switching. In circuit switching, a transparent circuit is established. No extra bits are needed. Therefore, there is no overhead in circuit switching. Because there is no overhead in circuit switching, line utilization must be more efficient than in packet switching.

9.7 Define the following parameters for a switching network:

N = number of hops between two given end systems

L = message length in bits

B = data rate, in bits per second (bps), on all links

P = fixed packet size, in bits

H = overhead (header) bits per packet
S = call setup time (circuit switching or virtual circuit) in seconds
D = propagation delay per hop in seconds

a. For $N = 4$, $L = 3200$, $B = 9600$, $P = 1024$, $H = 16$, $S = 0.2$, $D = 0.001$, compute the end-to-end delay for circuit switching, virtual circuit packet switching, and datagram packet switching. Assume that there are no acknowledgments. Ignore processing delay at the nodes.

b. Derive general expressions for the three techniques of part (a), taken two at a time (three expressions in all), showing the conditions under which the delays are equal.

9.8 What value of P, as a function of N, L, and H, results in minimum end-to-end delay on a datagram network? Assume that L is much larger than P, and D is zero.

9.9 Assuming no malfunction in any of the stations or nodes of a network, is it possible for a packet to be delivered to the wrong destination?

9.10 Although ATM does not include any end-to-end error detection and control functions on the user data, it is provided with a HEC field to detect and correct header errors. Let us consider the value of this feature. Suppose that the bit error rate of the transmission system is B. If errors are uniformly distributed, then the probability of an error in the header is

$$\frac{h}{h + i} \times B$$

and the probability of error in the data field is

$$\frac{i}{h + i} \times B$$

where h is the number of bits in the header and i is the number of bits in the data field.

a. Suppose that errors in the header are not detected and not corrected. In that case, a header error may result in a misrouting of the cell to the wrong destination; therefore, i bits will arrive at an incorrect destination, and i bits will not arrive at the correct destination. What is the overall bit error rate $B1$? Find an expression for the multiplication effect on the bit error rate: $M1 = B1/B$.

b. Now suppose that header errors are detected but not corrected. In that case, i bits will not arrive at the correct destination. What is the overall bit error rate $B2$? Find an expression for the multiplication effect on the bit error rate: $M2 = B2/B$.

c. Now suppose that header errors are detected and corrected. What is the overall bit error rate $B3$? Find an expression for the multiplication effect on the bit error rate: $M3 = B3/B$.

d. Plot $M1$, $M2$, and $M3$ as a function of header length, for $i = 48 \times 8 = 384$ bits. Comment on the results.

9.11 One key design decision for ATM was whether to use fixed- or variable-length cells. Let us consider this decision from the point of view of efficiency. We can define transmission efficiency as

$$N = \frac{\text{Number of information octets}}{\text{Number of information octets} + \text{Number of overhead octets}}$$

a. Consider the use of fixed-length packets. In this case the overhead consists of the header octets. Define

L = Data field size of the cell in octets
H = Header size of the cell in octets
X = Number of information octets to be transmitted as a single message

Derive an expression for N. *Hint:* The expression will need to use the operator $\lceil \cdot \rceil$, where $\lceil Y \rceil$ = the smallest integer greater than or equal to Y.

b. If cells have variable length, then overhead is determined by the header, plus the flags to delimit the cells or an additional length field in the header. Let Hv = additional overhead octets required to enable the use of variable-length cells. Derive an expression for N in terms of X, H, and Hv.

c. Let $L = 48$, $H = 5$, and $Hv = 2$. Plot N versus message size for fixed- and variable-length cells. Comment on the results.

9.12 Another key design decision for ATM is the size of the data field for fixed-size cells. Let us consider this decision from the point of view of efficiency and delay.

a. Assume that an extended transmission takes place, so that all cells are completely filled. Derive an expression for the efficiency N as a function of H and L.

b. Packetization delay is the delay introduced into a transmission stream by the need to buffer bits until an entire packet is filled before transmission. Derive an expression for this delay as a function of L and the data rate R of the source.

c. Common data rates for voice coding are 32 kbps and 64 kbps. Plot packetization delay as a function of L for these two data rates; use a left-hand y axis with a maximum value of 2 ms. On the same graph, plot transmission efficiency as a function of L; use a right-hand y-axis with a maximum value of 100%. Comment on the results.

9.13 Consider compressed video transmission in an ATM network. Suppose standard ATM cells must be transmitted through 5 switches. The data rate is 43 Mbps.

a. What is the transmission time for one cell through one switch?

b. Each switch may be transmitting a cell from other traffic all of which we assume to have lower (non-preemptive for the cell) priority. If the switch is busy transmitting a cell, our cell has to wait until the other cell completes transmission. If the switch is free our cell is transmitted immediately. What is the maximum time when a typical video cell arrives at the first switch (and possibly waits) until it is finished being transmitted by the fifth and last one? Assume that you can ignore propagation time, switching time, and everything else but the transmission time and the time spent waiting for another cell to clear a switch.

c. Now suppose we know that each switch is utilized 60% of the time with the other low-priority traffic. By this we mean that with probability 0.6 when we look at a switch it is busy. Suppose that if there is a cell being transmitted by a switch, the average delay spent waiting for a cell to finish transmission is one-half a cell transmission time. What is the average time from the input of the first switch to clearing the fifth?

d. However, the measure of most interest is not delay but jitter, which is the variability in the delay. Use parts (b) and (c) to calculate the maximum and average variability, respectively, in the delay.

In all cases assume that the various random events are independent of one another; for example, we ignore the burstiness typical of such traffic.

9.14 In order to support IP service over an ATM network, IP datagrams must first be segmented into a number of ATM cells before sending them over the ATM network. As ATM does not provide cell loss recovery, the loss of any of these cells will result in the loss of the entire IP packet. Given:

$$PC = \text{cell loss rate in the ATM network}$$
$$n = \text{number of cells required to transmit a single IP datagram}$$
$$PP = \text{IP-packet loss rate}$$

a. Derive an expression for PP, and comment on the resulting expression.

b. What ATM service would you use to get the best possible performance?