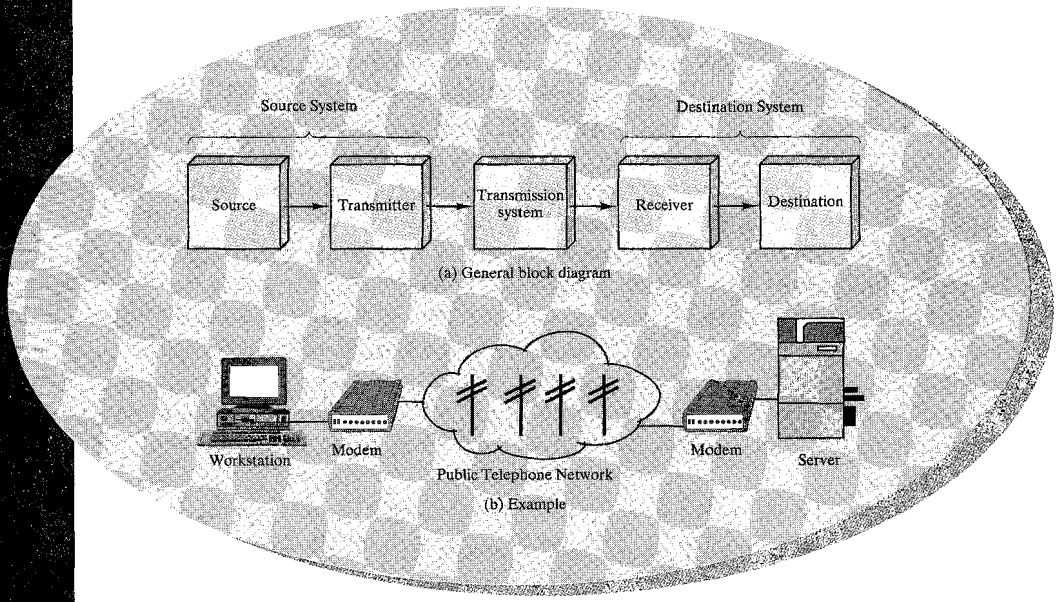
CHAPTER 1

INTRODUCTION



The 1970s and 1980s saw a merger of the fields of computer science and data communications that profoundly changed the technology, products, and companies of the now-combined computer-communications industry. Although the consequences of this revolutionary merger are still being worked out, it is safe to say that the revolution has occurred, and any investigation of the field of data communications must be made within this new context.

The computer-communications revolution has produced several remarkable facts:

- There is no fundamental difference between data processing (computers) and data communications (transmission and switching equipment).
- There are no fundamental differences among data, voice, and video communications.
- The lines between single-processor computer, multi-processor computer, local network, metropolitan network, and long-haul network have blurred.

One effect of these trends has been a growing overlap of the computer and communications industries, from component fabrication to system integration. Another result is the development of integrated systems that transmit and process all types of data and information. Both the technology and the technical-standards organizations are driving toward a single public system that integrates all communications and makes virtually all data and information sources around the world easily and uniformly accessible.

It is the ambitious purpose of this book to provide a unified view of the broad field of data and computer communications. The organization of the book reflects an attempt to break this massive subject into comprehensible parts and to build, piece by piece, a survey of the state of the art. This introductory chapter begins with a general model of communications. Then, a brief discussion introduces each of the four major parts of this book. Next, the all-important role of standards is introduced. Finally, a brief outline of the rest of the book is provided.

1.1 A COMMUNICATIONS MODEL

We begin our study with a simple model of communications, illustrated by the block diagram in Figure 1.1a.

The fundamental purpose of a communications system is the exchange of data between two parties. Figure 1.1b presents one particular example, which is the communication between a workstation and a server over a public telephone network. Another example is the exchange of voice signals between two telephones over the same network. The key elements of the model are

- **Source.** This device generates the data to be transmitted; examples are telephones and personal computers.

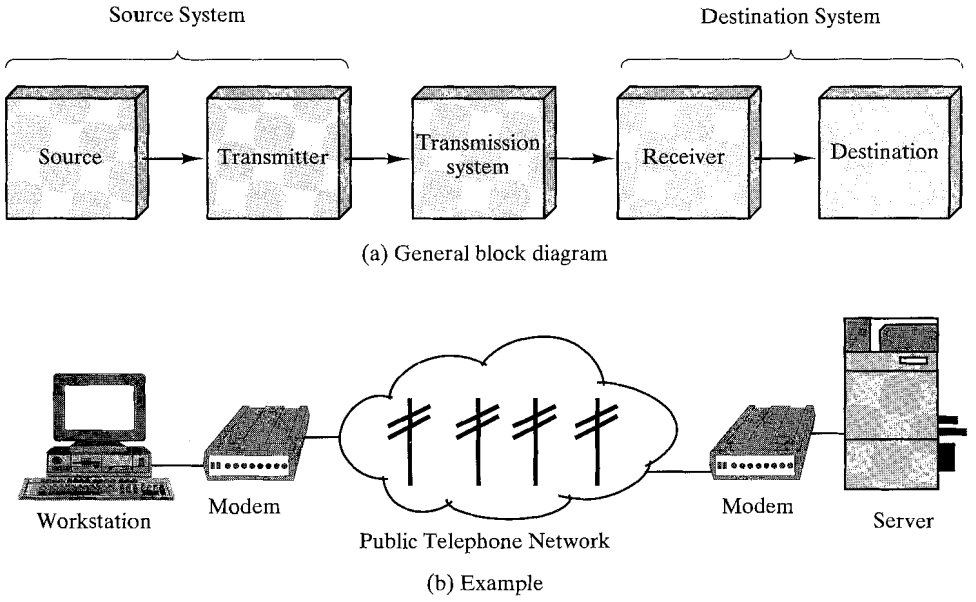


FIGURE 1.1 Simplified communications model.

- **Transmitter.** Usually, the data generated by a source system are not transmitted directly in the form in which they were generated. Rather, a transmitter transforms and encodes the information in such a way as to produce electromagnetic signals that can be transmitted across some sort of transmission system. For example, a modem takes a digital bit stream from an attached device such as a personal computer and transforms that bit stream into an analog signal that can be handled by the telephone network.
- **Transmission System.** This can be a single transmission line or a complex network connecting source and destination.
- **Receiver.** The receiver accepts the signal from the transmission system and converts it into a form that can be handled by the destination device. For example, a modem will accept an analog signal coming from a network or transmission line and convert it into a digital bit stream.
- **Destination.** Takes the incoming data from the receiver.

This simple narrative conceals a wealth of technical complexity. To get some idea of the scope of this complexity, Table 1.1 lists some of the key tasks that must be performed in a data communications system. The list is somewhat arbitrary: Elements could be added; items on the list could be merged; and some items represent several tasks that are performed at different "levels" of the system. However, the list as it stands is suggestive of the scope of this book.

TABLE 1.1 Communications tasks.

Transmission system utilization	Addressing
Interfacing	Routing
Signal generation	Recovery
Synchronization	Message formatting
Exchange management	Security
Error detection and correction	Network management
Flow control	

The first item, **transmission system utilization**, refers to the need to make efficient use of transmission facilities that are typically shared among a number of communicating devices. Various techniques (referred to as multiplexing) are used to allocate the total capacity of a transmission medium among a number of users. Congestion control techniques may be required to assure that the system is not overwhelmed by excessive demand for transmission services.

In order to communicate, a device must **interface** with the transmission system. All the forms of communication discussed in this book depend, at bottom, on the use of electromagnetic signals propagated over a transmission medium. Thus, once an interface is established, **signal generation** is required for communication. The properties of the signal, such as form and intensity, must be such that they are (1) capable of being propagated through the transmission system, and (2) interpretable as data at the receiver.

Not only must the signals be generated to conform to the requirements of the transmission system and receiver, but there must be some form of **synchronization** between transmitter and receiver. The receiver must be able to determine when a signal begins to arrive and when it ends. It must also know the duration of each signal element.

Beyond the basic matter of deciding on the nature and timing of signals, there are a variety of requirements for communication between two parties that might be collected under the term **exchange management**. If data are to be exchanged in both directions over a period of time, the two parties must cooperate. For example, for two parties to engage in a telephone conversation, one party must dial the number of the other, causing signals to be generated that result in the ringing of the called phone. The called party completes a connection by lifting the receiver. For data processing devices, more will be needed than simply establishing a connection; certain conventions must be decided upon. These conventions may include whether both devices may transmit simultaneously or must take turns, the amount of data to be sent at one time, the format of the data, and what to do if certain contingencies, such as an error, arise.

The next two items might have been included under exchange management, but they are important enough to list separately. In all communications systems, there is a potential for error; transmitted signals are distorted to some extent before reaching their destination. **Error detection and correction** are required in circumstances where errors cannot be tolerated; this is usually the case with data process-

ing systems. For example, in transferring a file from one computer to another, it is simply not acceptable for the contents of the file to be accidentally altered. **Flow control** is required to assure that the source does not overwhelm the destination by sending data faster than they can be processed and absorbed.

Next, we mention the related but distinct concepts of **addressing** and **routing**. When a transmission facility is shared by more than two devices, a source system must somehow indicate the identity of the intended destination. The transmission system must assure that the destination system, and only that system, receives the data. Further, the transmission system may itself be a network through which various paths may be taken. A specific route through this network must be chosen.

Recovery is a concept distinct from that of error correction. Recovery techniques are needed in situations in which an information exchange, such as a data base transaction or file transfer, is interrupted due to a fault somewhere in the system. The objective is either to be able to resume activity at the point of interruption or at least to restore the state of the systems involved to the condition prior to the beginning of the exchange.

Message formatting has to do with an agreement between two parties as to the form of the data to be exchanged or transmitted. For example, both sides must use the same binary code for characters.

Frequently, it is important to provide some measure of **security** in a data communications system. The sender of data may wish to be assured that only the intended party actually receives the data; and the receiver of data may wish to be assured that the received data have not been altered in transit and that the data have actually come from the purported sender.

Finally, a data communications facility is a complex system that cannot create or run itself. **Network management** capabilities are needed to configure the system, monitor its status, react to failures and overloads, and plan intelligently for future growth.

Thus we have gone from the simple idea of data communication between source and destination to a rather formidable list of data communications tasks. In this book, we further elaborate this list of tasks to describe and encompass the entire set of activities that can be classified under data and computer communications.

1.2 DATA COMMUNICATIONS

This book is organized into four parts. The first part deals with the most fundamental aspects of the communications function, focusing on the transmission of signals in a reliable and efficient manner. For want of a better name, we have given Part I the title "Data Communications," although that term arguably encompasses some or even all of the topics of Parts II, III, and IV.

To get some flavor for the focus of Part I, Figure 1.2 provides a new perspective on the communications model of Figure 1.1a. Let us trace through the details of this figure using electronic mail as an example.

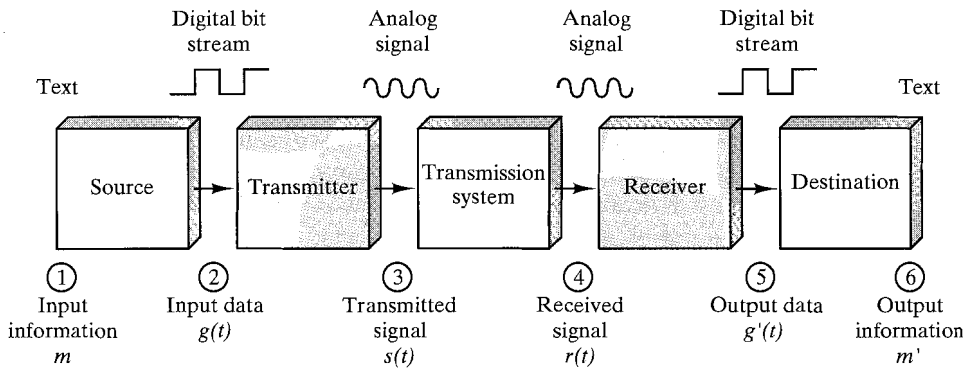


FIGURE 1.2 Simplified data communications model.

Consider that the input device and transmitter are components of a personal computer. The user of the PC wishes to send a message to another user—for example, "The meeting scheduled for March 25 is canceled" (m). The user activates the electronic mail package on the PC and enters the message via the keyboard (input device). The character string is briefly buffered in main memory. We can view it as a sequence of bits (g) in memory. The personal computer is connected to some transmission medium, such as a local network or a telephone line, by an I/O device (transmitter), such as a local network transceiver or a modem. The input data are transferred to the transmitter as a sequence of voltage shifts [$g(t)$] representing bits on some communications bus or cable. The transmitter is connected directly to the medium and converts the incoming stream [$g(t)$] into a signal [$s(t)$] suitable for transmission. Specific alternatives to this procedure will be described in Chapter 4.

The transmitted signal $s(t)$ presented to the medium is subject to a number of impairments, discussed in Chapter 2, before it reaches the receiver. Thus, the received signal $r(t)$ may differ to some degree from $s(t)$. The receiver will attempt to estimate the original $s(t)$, based on $r(t)$ and its knowledge of the medium, producing a sequence of bits $g'(t)$. These bits are sent to the output personal computer, where they are briefly buffered in memory as a block of bits (g). In many cases, the destination system will attempt to determine if an error has occurred and, if so, will cooperate with the source system to eventually obtain a complete, error-free block of data. These data are then presented to the user via an output device, such as a printer or a screen. The message (m'), as viewed by the user, will usually be an exact copy of the original message (m).

Now consider a telephone conversation. In this case, the input to the telephone is a message (m) in the form of sound waves. The sound waves are converted by the telephone into electrical signals of the same frequency. These signals are transmitted without modification over the telephone line. Hence, the input signal $g(t)$ and the transmitted signal $s(t)$ are identical. The signal $s(t)$ will suffer some distortion over the medium, so that $r(t)$ will not be identical to $s(t)$. Nevertheless, the signal $r(t)$ is converted back into a sound wave with no attempt at correction or

improvement of signal quality. Thus m' is not an exact replica of m. However, the received sound message is generally comprehensible to the listener.

The discussion so far does not touch on other key aspects of data communications, including data-link control techniques for controlling the flow of data and detecting and correcting errors, and multiplexing techniques for transmission efficiency. All of these topics are explored in Part I.

1.3 DATA COMMUNICATIONS NETWORKING

In its simplest form, data communication takes place between two devices that are directly connected by some form of point-to-point transmission medium. Often, however, it is impractical for two devices to be directly, point-to-point connected. This is so for one (or both) of the following contingencies:

- The devices are very far apart. It would be inordinately expensive, for example, to string a dedicated link between two devices thousands of miles apart.
- There is a set of devices, each of which may require a link to many of the others at various times. Examples are all of the telephones in the world and all of the terminals and computers owned by a single organization. Except for the case of a very few devices, it is impractical to provide a dedicated wire between each pair of devices.

The solution to this problem is to attach each device to a communications network. Figure 1.3 relates this area to the communications model of Figure 1.1a and also suggests the two major categories into which communications networks are traditionally classified: wide-area networks (WANs) and local-area networks (LANs). The distinction between the two, both in terms of technology and application, has become somewhat blurred in recent years, but it remains a useful way of organizing the discussion.

Wide-Area Networks

Wide-area networks have been traditionally been considered to be those that cover a large geographical area, require the crossing of public right-of-ways, and rely at least in part on circuits provided by a common carrier. Typically, a WAN consists of a number of interconnected switching nodes. A transmission from any one device is routed through these internal nodes to the specified destination device. These nodes (including the boundary 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.

Traditionally, WANs have been implemented using one of two technologies: circuit switching and packet switching. More recently, frame relay and ATM networks have assumed major roles.

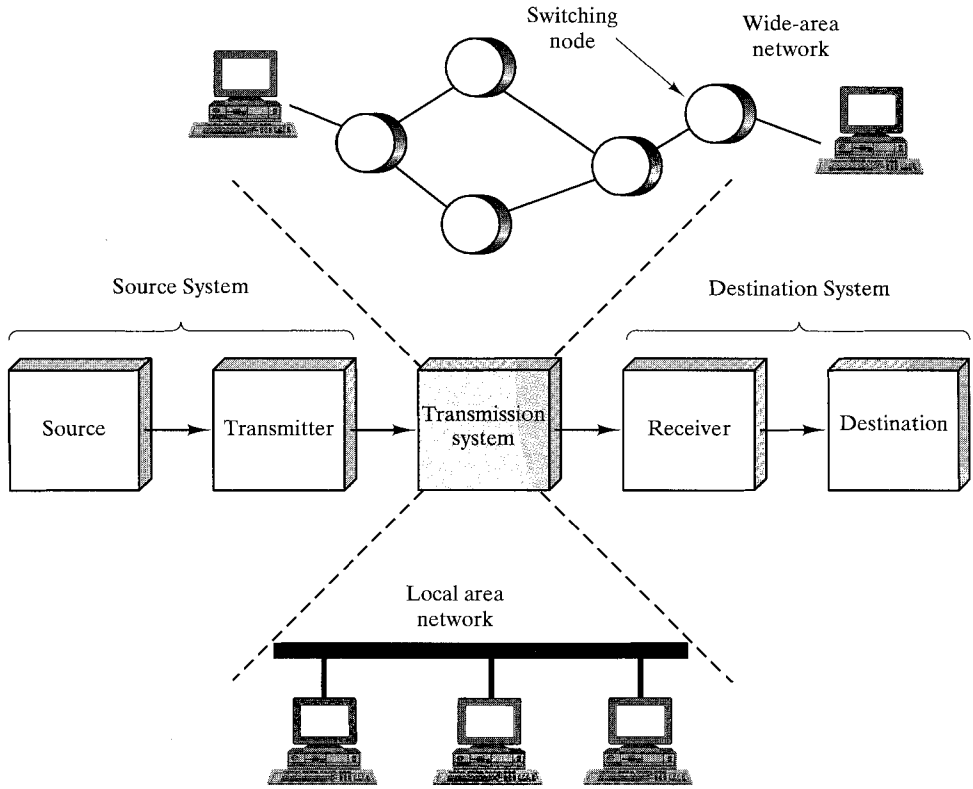


FIGURE 1.3 Simplified network models.

Circuit Switching

In a circuit-switched network, a dedicated communications path is established between two stations through the nodes of the network. That path is a connected sequence of physical links between nodes. On each link, a logical channel is dedicated to the connection. Data generated by the source station are transmitted along the dedicated path as rapidly as possible. At each node, incoming data are routed or switched to the appropriate outgoing channel without delay. The most common example of circuit switching is the telephone network.

Packet Switching

A quite different approach is used in a packet-switched network. In this case, it is not necessary to dedicate transmission capacity along a path through the network. Rather, data are sent out in a sequence of small chunks, called packets. Each packet is passed through the network from node to node along some path leading from source to destination. At each node, the entire packet is received, stored briefly, and then transmitted to the next node. Packet-switched networks are commonly used for terminal-to-computer and computer-to-computer communications.

Frame Relay

Packet switching was developed at a time when digital long-distance transmission facilities exhibited a relatively high error rate compared to today's facilities. As a result, there is a considerable amount of overhead built into packet-switched schemes to compensate for errors. The overhead includes additional bits added to each packet to introduce redundancy and additional processing at the end stations and the intermediate switching nodes to detect and recover from errors.

With modern high-speed telecommunications systems, this overhead is unnecessary and counterproductive. It is unnecessary because the rate of errors has been dramatically lowered and any remaining errors can easily be caught in the end systems by logic that operates above the level of the packet-switching logic; it is counterproductive because the overhead involved soaks up a significant fraction of the high capacity provided by the network.

Frame relay was developed to take advantage of these high data rates and low error rates. Whereas the original packet-switching networks were designed with a data rate to the end user of about 64 kbps, frame relay networks are designed to operate efficiently at user data rates of up to 2 Mbps. The key to achieving these high data rates is to strip out most of the overhead involved with error control.

ATM

Asynchronous transfer mode (ATM), sometimes referred to as cell relay, is a culmination of all of the developments in circuit switching and packet switching over the past 25 years.

ATM can be viewed as an evolution from frame relay. The most obvious difference between frame relay and ATM is that frame relay uses variable-length packets, called frames, and ATM uses fixed-length packets, called cells. As with frame relay, ATM provides little overhead for error control, depending on the inherent reliability of the transmission system and on higher layers of logic in the end systems to catch and correct errors. By using a fixed-packet length, the processing overhead is reduced even further for ATM compared to frame relay. The result is that ATM is designed to work in the range of 10s and 100s of Mbps, compared to the 2-Mbps target of frame relay.

ATM can also be viewed as an evolution from circuit switching. With circuit-switching, only fixed-data-rate circuits are available to the end system. ATM allows the definition of multiple virtual channels with data rates that are dynamically defined at the time the virtual channel is created. By using full, fixed-size cells, ATM is so efficient that it can offer a constant-data-rate channel even though it is using a packet-switching technique. Thus, ATM extends circuit switching to allow multiple channels with the data rate on each channel dynamically set on demand.

ISDN and Broadband ISDN

Merging and evolving communications and computing technologies, coupled with increasing demands for efficient and timely collection, processing, and dissemination of information, are leading to the development of integrated systems that

transmit and process all types of data. A significant outgrowth of these trends is the integrated services digital network (ISDN).

The ISDN is intended to be a worldwide public telecommunications network to replace existing public telecommunications networks and deliver a wide variety of services. The ISDN is defined by the standardization of user interfaces and implemented as a set of digital switches and paths supporting a broad range of traffic types and providing value-added processing services. In practice, there are multiple networks, implemented within national boundaries, but, from the user's point of view, there is intended to be a single, uniformly accessible, worldwide network.

Despite the fact that ISDN has yet to achieve the universal deployment hoped for, it is already in its second generation. The first generation, sometimes referred to as **narrowband ISDN**, is based on the use of a 64-kbps channel as the basic unit of switching and has a circuit-switching orientation. The major technical contribution of the narrowband ISDN effort has been frame relay. The second generation, referred to as **broadband ISDN**, supports very high data rates (100s of Mbps) and has a packet-switching orientation. The major technical contribution of the broadband ISDN effort has been asynchronous transfer mode (ATM), also known as cell relay.

Local Area Networks

As with wide-area networks, a local-area network is a communications network that interconnects a variety of devices and provides a means for information exchange among those devices. There are several key distinctions between LANs and WANs:

1. The scope of the LAN is small, typically a single building or a cluster of buildings. This difference in geographic scope leads to different technical solutions, as we shall see.
2. It is usually the case that the LAN is owned by the same organization that owns the attached devices. For WANs, this is less often the case, or at least a significant fraction of the network assets are not owned. This has two implications. First, care must be taken in the choice of LAN, as there may be a substantial capital investment (compared to dial-up or leased charges for wide-area networks) for both purchase and maintenance. Second, the network management responsibility for a local network falls solely on the user.
3. The internal data rates of LANs are typically much greater than those of wide-area networks.

Traditionally, LANs make use of a broadcast network approach rather than a switching approach. With a broadcast communication network, there are no intermediate switching nodes. At each station, there is a transmitter/receiver that communicates over a medium shared by other stations. A transmission from any one station is broadcast to and received by all other stations. A simple example of this is a CB radio system, in which all users tuned to the same channel may communicate. We will be concerned with networks used to link computers, workstations, and

other digital devices. In the latter case, data are usually transmitted in packets. Because the medium is shared, only one station at a time can transmit a packet.

More recently, examples of switched LANs have appeared. The two most prominent examples are ATM LANs, which simply use an ATM network in a local area, and Fibre Channel. We will examine these LANs, as well as the more common broadcast LANs, in Part III.

1.4 PROTOCOLS AND PROTOCOL ARCHITECTURE

When computers, terminals, and/or other data processing devices exchange data, the scope of concern is much broader than the concerns we have discussed in Sections 1.2 and 1.3. Consider, for example, the transfer of a file between two computers. There must be a data path between the two computers, either directly or via a communication network. But more is needed. Typical tasks to be performed are

1. The source system must either activate the direct data communication path or inform the communication network of the identity of the desired destination system.
2. The source system must ascertain that the destination system is prepared to receive data.
3. The file transfer application on the source system must ascertain that the file management program on the destination system is prepared to accept and store the file for this particular user.
4. If the file formats used on the two systems are incompatible, one or the other system must perform a format translation function.

It is clear that there must be a high degree of cooperation between the two computer systems. The exchange of information between computers for the purpose of cooperative action is generally referred to as *computer communications*. Similarly, when two or more computers are interconnected via a communication network, the set of computer stations is referred to as a *computer network*. Because a similar level of cooperation is required between a user at a terminal and one at a computer, these terms are often used when some of the communicating entities are terminals.

In discussing computer communications and computer networks, two concepts are paramount:

- Protocols
- Computer-communications architecture, or protocol architecture

A protocol is used for communication between entities in different systems. The terms "entity" and "system" are used in a very general sense. Examples of

entities are user application programs, file transfer packages, data-base management systems, electronic mail facilities, and terminals. Examples of systems are computers, terminals, and remote sensors. Note that in some cases the entity and the system in which it resides are coextensive (e.g., terminals). In general, an entity is anything capable of sending or receiving information, and a system is a physically distinct object that contains one or more entities. For two entities to communicate successfully, they must "speak the same language." What is communicated, how it is communicated, and when it is communicated must conform to some mutually acceptable conventions between the entities involved. The conventions are referred to as a protocol, which may be defined as a set of rules governing the exchange of data between two entities. The key elements of a protocol are

- **Syntax.** Includes such things as data format and signal levels.
- **Semantics.** Includes control information for coordination and error handling.
- **Timing.** Includes speed matching and sequencing.

Having introduced the concept of a protocol, we can now introduce the concept of a protocol architecture. It is clear that there must be a high degree of cooperation between the two computers. Instead of implementing the logic for this as a single module, the task is broken up into subtasks, each of which is implemented separately. As an example, Figure 1.4 suggests the way in which a file transfer facility could be implemented. Three modules are used. Tasks 3 and 4 in the preceding list could be performed by a file transfer module. The two modules on the two systems exchange files and commands. However, rather than requiring the file transfer module to handle the details of actually transferring data and commands, the file transfer modules each rely on a communications service module. This module is responsible for making sure that the file transfer commands and data are reliably exchanged between systems. Among other things, this module would perform task 2. Now, the nature of the exchange between systems is independent of the nature of the network that interconnects them. Therefore, rather than building details of the network interface into the communications service module, it makes sense to have a third module, a network access module, that performs task 1 by interacting with the network.

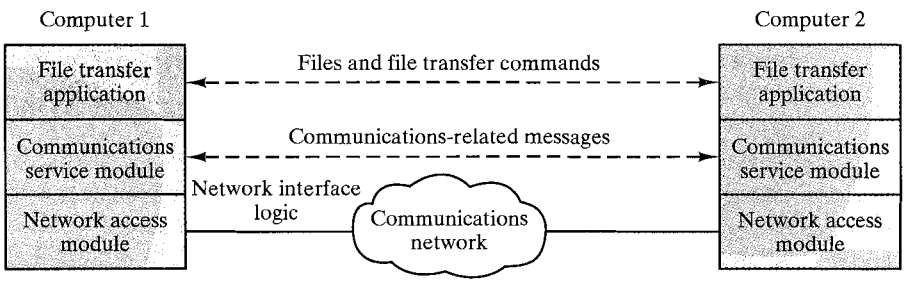


FIGURE 1.4 A simplified architecture for file transfer.

Let us try to summarize the motivation for the three modules in Figure 1.4. The file transfer module contains all of the logic that is unique to the file transfer application, such as transmitting passwords, file commands, and file records. There is a need to transmit these files and commands reliably. However, the same sorts of reliability requirements are relevant to a variety of applications (e.g., electronic mail, document transfer). Therefore, these requirements are met by a separate communications service module that can be used by a variety of applications. The communications service module is concerned with assuring that the two computer systems are active and ready for data transfer and for keeping track of the data that are being exchanged to assure delivery. However, these tasks are independent of the type of network that is being used. Therefore, the logic for actually dealing with the network is separated out into a separate network access module. That way, if the network to be used is changed, only the network access module is affected.

Thus, instead of a single module for performing communications, there is a structured set of modules that implements the communications function. That structure is referred to as a protocol architecture. In the remainder of this section, we generalize the preceding example to present a simplified protocol architecture. Following that, we look at more complex, real-world examples: TCP/IP and OSI.

A Three-Layer Model

In very general terms, communications can be said to involve three agents: applications, computers, and networks. One example of an application is a file transfer operation. These applications execute on computers that can often support multiple simultaneous applications. Computers are connected to networks, and the data to be exchanged are transferred by the network from one computer to another. Thus, the transfer of data from one application to another involves first getting the data to the computer in which the application resides and then getting it to the intended application within the computer.

With these concepts in mind, it appears natural to organize the communication task into three relatively independent layers:

- Network access layer
- Transport layer
- Application layer

The **network access layer** is concerned with the exchange of data between a computer and the network to which it is attached. The sending computer must provide the network with the address of the destination computer, so that the network may route the data to the appropriate destination. The sending computer may wish to invoke certain services, such as priority, that might be provided by the network. The specific software used at this layer depends on the type of network to be used; different standards have been developed for circuit switching, packet switching, local area networks, and others. Thus, it makes sense to separate those functions having to do with network access into a separate layer. By doing this, the remainder of the communications software, above the network access layer, need not be

concerned with the specifics of the network to be used. The same higher-layer software should function properly regardless of the particular network to which the computer is attached.

Regardless of the nature of the applications that are exchanging data, there is usually a requirement that data be exchanged reliably. That is, we would like to be assured that all of the data arrive at the destination application and that the data arrive in the same order in which they were sent. As we shall see, the mechanisms for providing reliability are essentially independent of the nature of the applications. Thus, it makes sense to collect those mechanisms in a common layer shared by all applications; this is referred to as the **transport layer**.

Finally, the **application layer** contains the logic needed to support the various user applications. For each different type of application, such as file transfer, a separate module is needed that is peculiar to that application.

Figures 1.5 and 1.6 illustrate this simple architecture. Figure 1.5 shows three computers connected to a network. Each computer contains software at the network access and transport layers and software at the application layer for one or more applications. For successful communication, every entity in the overall system must have a unique address. Actually, two levels of addressing are needed. Each computer on the network must have a unique network address; this allows the network to deliver data to the proper computer. Each application on a computer must have an address that is unique within that computer; this allows the transport layer to support multiple applications at each computer. These latter addresses are

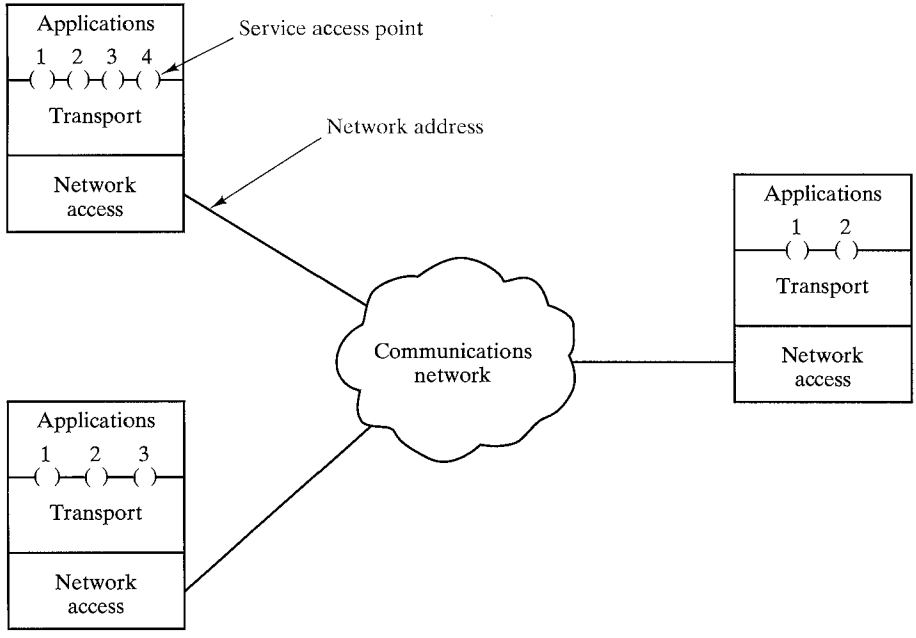


FIGURE 1.5 Protocol architectures and networks.

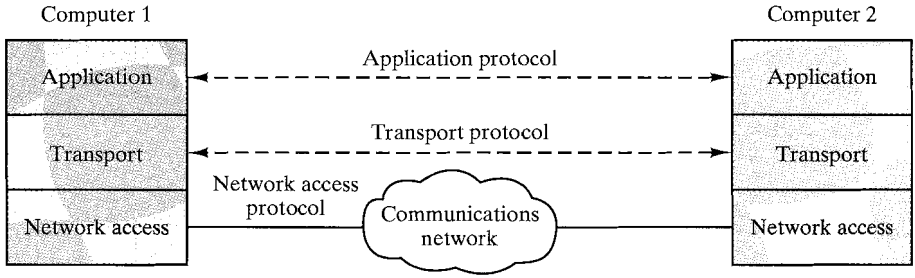


FIGURE 1.6 Protocols in a simplified architecture.

known as service access points (SAPs), connoting that each application is individually accessing the services of the transport layer.

Figure 1.6 indicates the way in which modules at the same level on different computers communicate with each other: by means of a protocol. A protocol is the set of rules or conventions governing the ways in which two entities cooperate to exchange data. A protocol specification details the control functions that may be performed, the formats and control codes used to communicate those functions, and the procedures that the two entities must follow.

Let us trace a simple operation. Suppose that an application, associated with SAP 1 at computer A, wishes to send a message to another application, associated with SAP 2 at computer B. The application at A hands the message over to its transport layer with instructions to send it to SAP 2 on computer B. The transport layer hands the message over to the network access layer, which instructs the network to send the message to computer B. Note that the network need not be told the identity of the destination service access point. All that it needs to know is that the data are intended for computer B.

To control this operation, control information, as well as user data, must be transmitted, as suggested in Figure 1.7. Let us say that the sending application generates a block of data and passes this to the transport layer. The transport layer may break this block into two smaller pieces to make it more manageable. To each of these pieces the transport layer appends a transport header, containing protocol control information. The combination of data from the next higher layer and control information is known as a protocol data unit (PDU); in this case, it is referred to as a transport protocol data unit. The header in each transport PDU contains control information to be used by the peer transport protocol at computer B. Examples of items that may be stored in this header include

- **Destination SAP.** When the destination transport layer receives the transport protocol data unit, it must know to whom the data are to be delivered.
- **Sequence number.** Because the transport protocol is sending a sequence of protocol data units, it numbers them sequentially so that if they arrive out of order, the destination transport entity may reorder them.

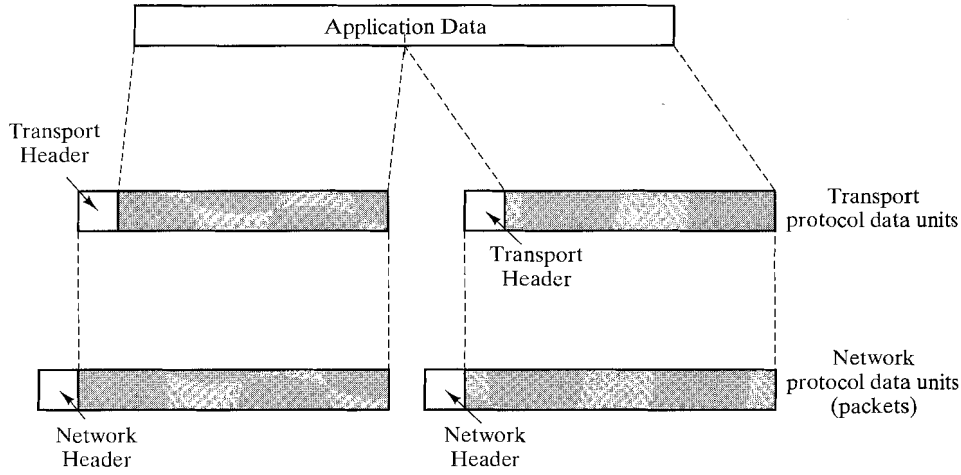


FIGURE 1.7 Protocol data units.

- **Error-detection code.** The sending transport entity may include a code that is a function of the contents of the remainder of the PDU. The receiving transport protocol performs the same calculation and compares the result with the incoming code. A discrepancy results if there has been some error in transmission. In that case, the receiver can discard the PDU and take corrective action.

The next step is for the transport layer to hand each protocol data unit over to the network layer, with instructions to transmit it to the destination computer. To satisfy this request, the network access protocol must present the data to the network with a request for transmission. As before, this operation requires the use of control information. In this case, the network access protocol appends a network access header to the data it receives from the transport layer, creating a network-access PDU. Examples of the items that may be stored in the header include

- **Destination computer address.** The network must know to whom (which computer on the network) the data are to be delivered.
- **Facilities requests.** The network access protocol might want the network to make use of certain facilities, such as priority.

Figure 1.8 puts all of these concepts together, showing the interaction between modules to transfer one block of data. Let us say that the file transfer module in computer X is transferring a file one record at a time to computer Y. Each record is handed over to the transport layer module. We can picture this action as being in the form of a command or procedure call. The arguments of this procedure call include the destination computer address, the destination service access point, and

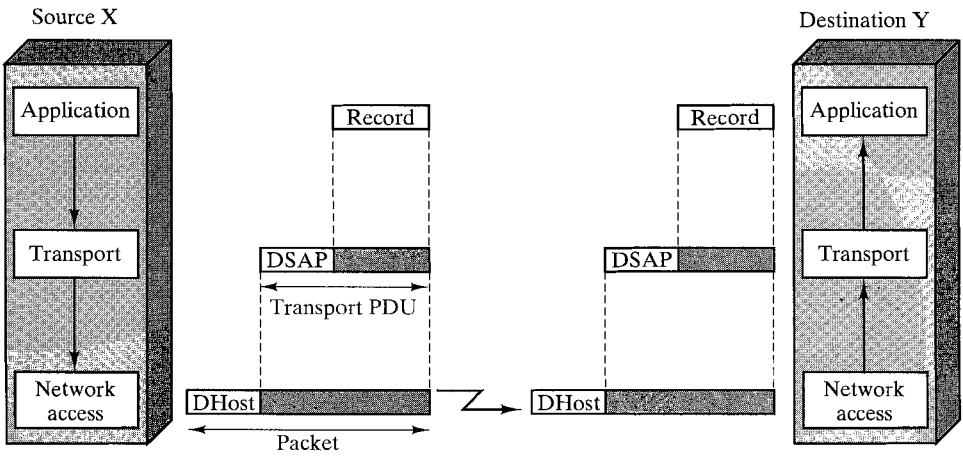


FIGURE 1.8 Operation of a protocol architecture.

the record. The transport layer appends the destination service access point and other control information to the record to create a transport PDU. This is then handed down to the network access layer by another procedure call. In this case, the arguments for the command are the destination computer address and the transport protocol data unit. The network access layer uses this information to construct a network PDU. The transport protocol data unit is the data field of the network PDU, and the network PDU header includes information concerning the source and destination computer addresses. Note that the transport header is not "visible" at the network access layer; the network access layer is not concerned with the contents of the transport PDU.

The network accepts the network PDU from X and delivers it to Y. The network access module in Y receives the PDU, strips off the header, and transfers the enclosed transport PDU to X's transport layer module. The transport layer examines the transport protocol data unit header and, on the basis of the SAP field in the header, delivers the enclosed record to the appropriate application, in this case the file transfer module in Y.

The TCP/IP Protocol Architecture

Two protocol architectures have served as the basis for the development of interoperable communications standards: the TCP/IP protocol suite and the OSI reference model. TCP/IP is the most widely used interoperable architecture, and OSI has become the standard model for classifying communications functions. In the remainder of this section, we provide a brief overview of the two architectures; the topic is explored more fully in Chapter 15.

TCP/IP is a result of protocol research and development conducted on the experimental packet-switched network, ARPANET, funded by the Defense Advanced Research Projects Agency (DARPA), and is generally referred to as the

TCP/IP protocol suite. This protocol suite consists of a large collection of protocols that have been issued as Internet standards by the Internet Architecture Board (IAB).

There is no official TCP/IP protocol model as there is in the case of OSI. However, based on the protocol standards that have been developed, we can organize the communication task for TCP/IP into five relatively independent layers:

- Application layer
- Host-to-host, or transport layer
- Internet layer
- Network access layer
- Physical layer

The **physical layer** covers the physical interface between a data transmission device (e.g., workstation, computer) and a transmission medium or network. This layer is concerned with specifying the characteristics of the transmission medium, the nature of the signals, the data rate, and related matters.

The **network access layer** is concerned with the exchange of data between an end system and the network to which it is attached. The sending computer must provide the network with the address of the destination computer, so that the network may route the data to the appropriate destination. The sending computer may wish to invoke certain services, such as priority, that might be provided by the network. The specific software used at this layer depends on the type of network to be used; different standards have been developed for circuit-switching, packet-switching (e.g., X.25), local area networks (e.g., Ethernet), and others. Thus, it makes sense to separate those functions having to do with network access into a separate layer. By doing this, the remainder of the communications software, above the network access layer, need not be concerned about the specifics of the network to be used. The same higher-layer software should function properly regardless of the particular network to which the computer is attached.

The network access layer is concerned with access to and routing data across a network for two end systems attached to the same network. In those cases where two devices are attached to different networks, procedures are needed to allow data to traverse multiple interconnected networks. This is the function of the **internet layer**. The internet protocol (IP) is used at this layer to provide the routing function across multiple networks. This protocol is implemented not only in the end systems but also in routers. A router is a processor that connects two networks and whose primary function is to relay data from one network to the other on its route from the source to the destination end system.

Regardless of the nature of the applications that are exchanging data, there is usually a requirement that data be exchanged reliably. That is, we would like to be assured that all of the data arrive at the destination application and that the data arrive in the same order in which they were sent. As we shall see, the mechanisms for providing reliability are essentially independent of the nature of the applica-

tions. Thus, it makes sense to collect those mechanisms in a common layer shared by all applications; this is referred to as the **host-to-host layer**, or **transport layer**. The transmission control protocol (TCP) is the most commonly-used protocol to provide this functionality.

Finally, the **application layer** contains the logic needed to support the various user applications. For each different type of application, such as file transfer, a separate module is needed that is peculiar to that application.

Figure 1.9 shows how the TCP/IP protocols are implemented in end systems and relates this description to the communications model of Figure 1.1a. Note that the physical and network access layers provide interaction between the end system and the network, whereas the transport and application layers are what is known as end-to-end protocols; they support interaction between two end systems. The internet layer has the flavor of both. At this layer, the end system communicates routing information to the network but also must provide some common functions between the two end systems; these will be explored in Chapters 15 and 16.

The OSI Model

The open systems interconnection (OSI) model was developed by the International Organization for Standardization (ISO) as a model for a computer communications architecture and as a framework for developing protocol standards. It consists of seven layers:

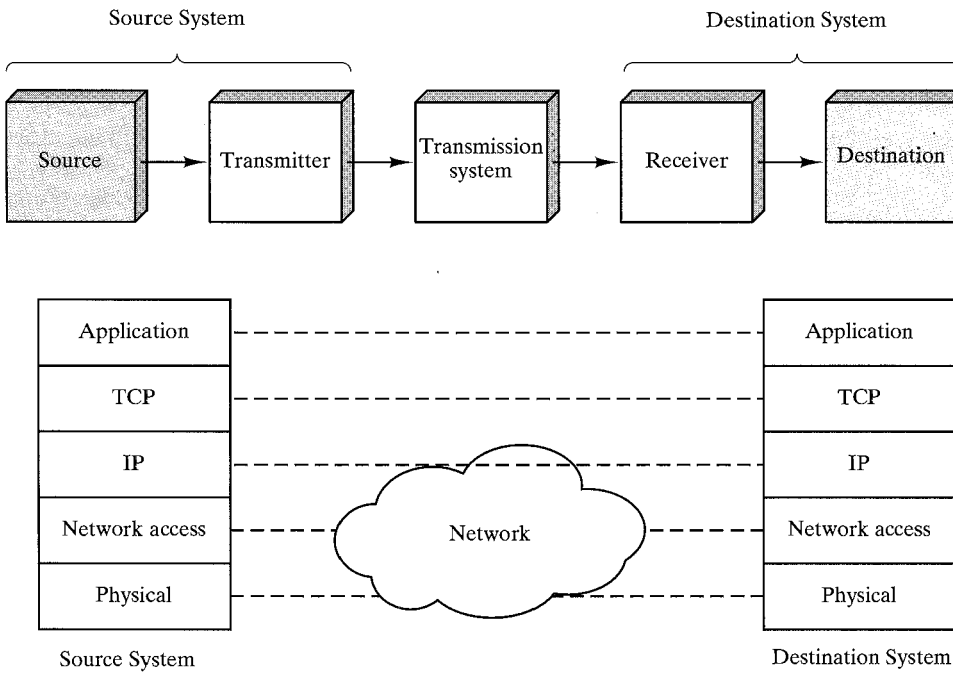


FIGURE 1.9 Protocol architecture model.

- Application
- Presentation
- Session
- Transport
- Network
- Data Link
- Physical

Figure 1.10 illustrates the OSI model and provides a brief definition of the functions performed at each layer. The intent of the OSI model is that protocols be developed to perform the functions of each layer.

The designers of OSI assumed that this model and the protocols developed within this model would come to dominate computer communications, eventually replacing proprietary protocol implementations and rival multivendor models such as TCP/IP. This has not happened. Although many useful protocols have been developed in the context of OSI, the overall seven-layer model has not flourished. Instead, it is the TCP/IP architecture that has come to dominate. Thus, our emphasis in this book will be on TCP/IP.

Application Provides access to the OSI environment for users and also provides distributed information services.
Presentation Provides independence to the application processes from differences in data representation (syntax).
Session Provides the control structure for communication between applications; establishes, manages, and terminates connections (sessions) between cooperating applications.
Transport Provides reliable, transparent transfer of data between end points; provides end-to-end error recovery and flow control.
Network Provides upper layers with independence from the data transmission and switching technologies used to connect systems; responsible for establishing, maintaining, and terminating connections.
Data Link Provides for the reliable transfer of information across the physical link; sends blocks of data (frames) with the necessary synchronization, error control, and flow control.
Physical Concerned with transmission of unstructured bit stream over physical medium; deals with the mechanical, electrical, functional, and procedural characteristics to access the physical medium.

FIGURE 1.10 The OSI layers.

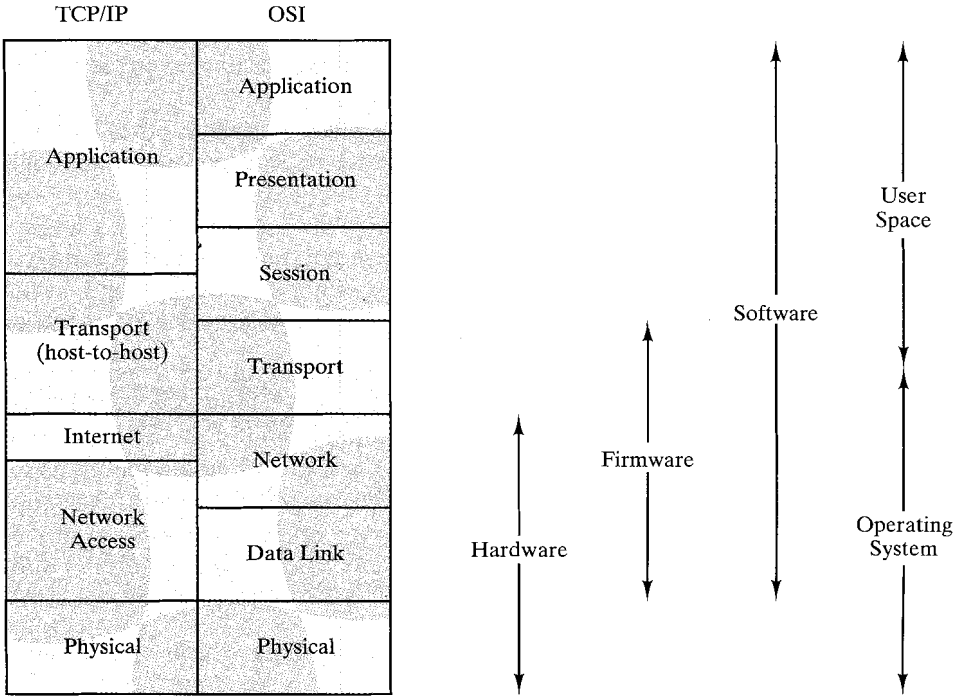


FIGURE 1.11 Protocol architectures.

Figure 1.11 illustrates the layers of the TCP/IP and OSI architectures, showing roughly the correspondence in functionality between the two. The figure also suggests common means of implementing the various layers.

1.5 STANDARDS

It has long been accepted in the communications industry that standards are required to govern the physical, electrical, and procedural characteristics of communication equipment. In the past, this view has not been embraced by the computer industry. Whereas communication-equipment vendors recognize that their equipment will generally interface to and communicate with other vendors' equipment, computer vendors have traditionally attempted to lock their customers into proprietary equipment; the proliferation of computers and distributed processing has made that an untenable position. Computers from different vendors must communicate with each other and, with the ongoing evolution of protocol standards, customers will no longer accept special-purpose protocol-conversion software development. The result is that standards now permeate all of the areas of technology discussed in this book.

Throughout the book we will describe the most important standards that are in use or that are being developed for various aspects of data and computer communications. Appendix 1A looks at the key organizations involved with the development of standards.

There are a number of advantages and disadvantages to the standards-making process. We list here the most striking ones. The principal advantages of standards are the following:

- A standard assures that there will be a large market for a particular piece of equipment or software. This encourages mass production and, in some cases, the use of large-scale-integration (LSI) or very-large-scale-integration (VLSI) techniques, resulting in lower costs.
- A standard allows products from multiple vendors to communicate, giving the purchaser more flexibility in equipment selection and use.

The principal disadvantages are these:

- A standard tends to freeze the technology. By the time a standard is developed, subjected to review and compromise, and promulgated, more efficient techniques are possible.
- There are multiple standards for the same thing. This is not a disadvantage of standards per se, but of the current way things are done. Fortunately, in recent years the various standards-making organizations have begun to cooperate more closely. Nevertheless, there are still areas where multiple conflicting standards exist.

1.6 OUTLINE OF THE BOOK

This chapter, of course, serves as an introduction to the entire book. A brief synopsis of the remaining chapters follows.

Data Transmission

The principles of data transmission underlie all of the concepts and techniques presented in this book. To understand the need for encoding, multiplexing, switching, error control, and so on, the reader must understand the behavior of data signals propagated through a transmission medium. Chapter 2 provides an understanding of the distinction between digital and analog data and digital and analog transmission. Concepts of attenuation and noise are also examined.

Transmission Media

Transmission media can be classified as either guided or wireless. The most commonly-used guided transmission media are twisted pair, coaxial cable, and optical

fiber. Wireless techniques include terrestrial and satellite microwave, broadcast radio, and infrared. Chapter 3 covers all of these topics.

Data Encoding

Data come in both analog (continuous) and digital (discrete) form. For transmission, input data must be encoded as an electrical signal that is tailored to the characteristics of the transmission medium. Both analog and digital data can be represented by either analog or digital signals; each of the four cases is discussed in Chapter 4. This chapter also covers spread-spectrum techniques.

The Data Communications Interface

In Chapter 5 the emphasis shifts from data transmission to data communications. For two devices linked by a transmission medium to exchange digital data, a high degree of cooperation is required. Typically, data are transmitted one bit at a time over the medium. The timing (rate, duration, spacing) of these bits must be the same for transmitter and receiver. Two common communication techniques—asynchronous and synchronous—are explored. This chapter also looks at transmission line interfaces. Typically, digital data devices do not attach to and signal across a transmission medium directly. Rather, this process is mediated through a standardized interface.

Data Link Control

True cooperative exchange of digital data between two devices requires some form of data link control. Chapter 6 examines the fundamental techniques common to all data link control protocols including flow control and error detection and correction, and then examines the most commonly used protocol, HDLC.

Multiplexing

Transmission facilities are, by and large, expensive. It is often the case that two communication stations will not utilize the full capacity of a data link. For efficiency, it should be possible to share that capacity. The generic term for such sharing is multiplexing.

Chapter 7 concentrates on the three most common types of multiplexing techniques. The first, frequency-division multiplexing (FDM), is the most widespread and is familiar to anyone who has ever used a radio or television set. The second is a particular case of time-division multiplexing (TDM), often known as synchronous TDM. This is commonly used for multiplexing digitized voice streams. The third type is another form of TDM that is more complex but potentially more efficient than synchronous TDM; it is referred to as statistical or asynchronous TDM.

Circuit Switching

Any treatment of the technology and architecture of circuit-switched networks must of necessity focus on the internal operation of a single switch. This is in con-

trast to packet-switched networks, which are best explained by the collective behavior of the set of switches that make up a network. Thus, Chapter 8 begins by examining digital-switching concepts, including space- and time-division switching. Then, the concepts of a multinode circuit-switched network are discussed; here, we are primarily concerned with the topics of routing and control signaling.

Packet Switching

There are two main technical problems associated with a packet-switched network, and each is examined in Chapter 9:

- **Routing.** Because the source and destination stations are not directly connected, the network must route each packet, from node to node, through the network.
- **Congestion control.** The amount of traffic entering and transiting the network must be regulated for efficient, stable, and fair performance.

The key design issues in both of these areas are presented and analyzed; the discussion is supported by examples from specific networks. In addition, a key packet-switching interface standard, X.25, is described.

Frame Relay

Chapter 10 examines the most important innovation to come out of the work on ISDN: frame relay. Frame relay provides a more efficient means of supporting packet switching than X.25 and is enjoying widespread use, not only in ISDN but in other networking contexts. This chapter looks at the data-transfer protocol and call-control protocol for frame relay and also looks at the related data link control protocol, LAPF.

A critical component for frame relay is congestion control. The chapter explains the nature of congestion in frame relay networks and both the importance and difficulty of controlling congestion. The chapter then describes a range of congestion control techniques that have been specified for use in frame relay networks.

Asynchronous Transfer Mode (ATM)

Chapter 11 focuses on the transmission technology that is the foundation of broadband ISDN: asynchronous transfer mode (ATM). As with frame relay, ATM is finding widespread application beyond its use as part of broadband. This chapter begins with a description of the ATM protocol and format. Then the physical layer issues relating to the transmission of ATM cells and the ATM Adaptation Layer (AAL) are discussed.

Again, as with frame relay, congestion control is a vital component of ATM. This area, referred to as ATM traffic and congestion control, is one of the most complex aspects of ATM and is the subject of intensive ongoing research. This chapter surveys those techniques that have been accepted as having broad utility in

ATM environments.

LAN Technology

The essential technology underlying all forms of local area networks comprises topology, transmission medium, and medium access control technique. Chapter 12 examines the first two of these elements. Four topologies are in common use: bus, tree, ring, star. The most common transmission media for local networking are twisted pair (unshielded and shielded), coaxial cable (baseband and broadband), and optical fiber. These topologies and transmission media are discussed, and the most promising combinations are described.

LAN Systems

Chapter 13 looks in detail at the topologies, transmission media, and MAC protocols of the most important LAN systems in current use; all of these have been defined in standards documents. The discussion opens with what might be called traditional LANs, which typically operate at data rates of up to 10 Mbps and which have been in use for over a decade. These include Ethernet and related LANs and two token-passing schemes, token ring and FDDI (fiber distributed data interface). Then, more recent high-speed LAN systems are examined, including ATM LANs. Finally, the chapter looks at wireless LANs.

Bridges

The increasing deployment of LANs has led to an increased need to interconnect LANs with each other and with wide-area networks. Chapter 14 focuses on a key device used in interconnection LANs: the bridges. Bridge operation involves two types of protocols: protocols for forwarding packets and protocols for exchanging routing information.

This chapter also returns to the topic of ATM LANs to look at the important concept of ATM LAN emulation, which relates to connecting other types of LANs to ATM networks.

Protocols and Architecture

Chapter 15 introduces the subject of protocol architecture and motivates the need for a layered architecture with protocols defined at each layer. The concept of protocol is defined, and the important features of protocols are discussed.

The two most important communications architectures are introduced in this chapter. The open systems interconnection (OSI) model is described in some detail. Next, the TCP/IP model is examined. Although the OSI model is almost universally accepted as the framework for discourse in this area, it is the TCP/IP protocol suite that is the basis for most commercially available interoperable products.

Internetworking

With the proliferation of networks, internetworking facilities have become essential components of network design. Chapter 16 begins with an examination of the requirements for an internetworking facility and the various design approaches that can be taken to satisfy those requirements. The remainder of the chapter explores the use of routers for internetworking. The internet protocol (IP) and the new IPv6, also known as IPng, are examined. Various routing protocols are also described, including the widely used OSPF and BGP.

Transport Protocols

The transport protocol is the keystone of the whole concept of a computer communications architecture. It can also be one of the most complex of protocols. Chapter 17 examines in detail transport protocol mechanisms and then introduces two important examples: TCP and UDP.

Network Security

Network security has become increasingly important with the growth in the number and importance of networks. Chapter 18 provides a survey of security techniques and services. The chapter begins with a look at encryption techniques for insuring privacy, which include the use of conventional and public-key encryption. Then, the area of authentication and digital signatures is explored. The two most important encryption algorithms, DES and RSA, are examined, as well as MD5, a one-way hash function important in a number of security applications.

Distributed Applications

The purpose of a communications architecture is to support distributed applications. Chapter 19 examines three of the most important of these applications; in each case, general principles are discussed and are followed by a specific example. The applications discussed are network management, world-wide web (WWW) exchanges, and electronic mail. The corresponding examples are SNMPv2, HTTP, and SMTP/MIME. Before getting to these examples, the chapter opens with an examination of Abstract Syntax Notation One (ASN.1), which is the standardized language for defining distributed applications.

ISDN and Broadband ISDN

The integrated-services digital network (ISDN) is a projected worldwide public telecommunications network that is designed to service a variety of user needs. Broadband ISDN is an enhancement of ISDN that can support very high data rates. Appendix A looks at the architecture, design principles, and standards for ISDN and broadband ISDN.

1A APPENDIX

STANDARDS ORGANIZATIONS

THROUGHOUT THIS BOOK, we describe the most important standards in use or being developed for various aspects of data and computer communications. Various organizations have been involved in the development or promotion of these standards. This appendix provides a brief description of the most important (in the current context) of these organizations:

- IETF
- ISO
- ITU-T

Internet Standards and the IETF

Many of the protocols that make up the TCP/IP protocol suite have been standardized or are in the process of standardization. By universal agreement, an organization known as the Internet Architecture Board (IAB) is responsible for the development and publication of these standards, which are published in a series of documents called Requests for Comments (RFCs).

This section provides a brief description of the way in which standards for the TCP/IP protocol suite are developed.

The Internet and Internet Standards

The Internet is a large collection of interconnected networks, all of which use the TCP/IP protocol suite. The Internet began with the development of ARPANET and the subsequent support by the Defense Advanced Research Projects Agency (DARPA) for the development of additional networks to support military users and government contractors.

The IAB is the coordinating committee for Internet design, engineering, and management. Areas covered include the operation of the Internet itself and the standardization of protocols used by end systems on the Internet for interoperability. The IAB has two principle subsidiary task forces:

- Internet Engineering Task Force (IETF)
- Internet Research Task Force (IRTF)

The actual work of these task forces is carried out by working groups. Membership in a working group is voluntary; any interested party may participate.

It is the IETF that is responsible for publishing the RFCs. The RFCs are the working notes of the Internet research and development community. A document in this series may be on essentially any topic related to computer communications, and may be anything from a meeting report to the specification of a standard.

The final decision of which RFCs become Internet standards is made by the IAB, on the recommendation of the IETF. To become a standard, a specification must meet the following criteria:

- Be stable and well-understood
- Be technically competent
- Have multiple, independent, and interoperable implementations with operational experience

- Enjoy significant public support
- Be recognizably useful in some or all parts of the Internet

The key difference between these criteria and those used for international standards is the emphasis here on operational experience.

The Standardization Process

Figure 1.12 shows the series of steps, called the *standards track*, that a specification goes through to become a standard. The steps involve increasing amounts of scrutiny and testing. At each step, the IETF must make a recommendation for advancement of the protocol, and the IAB must ratify it.

The white boxes in the diagram represent temporary states, which should be occupied for the minimum practical time. However, a document must remain a proposed standard for at least six months and a draft standard for at least four months to allow time for review and comment. The gray boxes represent long-term states that may be occupied for years.

A protocol or other specification that is not considered ready for standardization may be published as an experimental RFC. After further work, the specification may be resubmitted. If the specification is generally stable, has resolved known design choices, is believed to be well-understood, has received significant community review, and appears to enjoy enough community interest to be considered valuable, then the RFC will be designated a proposed standard.

For a specification to be advanced to draft-standard status, there must be at least two independent and interoperable implementations from which adequate operational experience has been obtained.

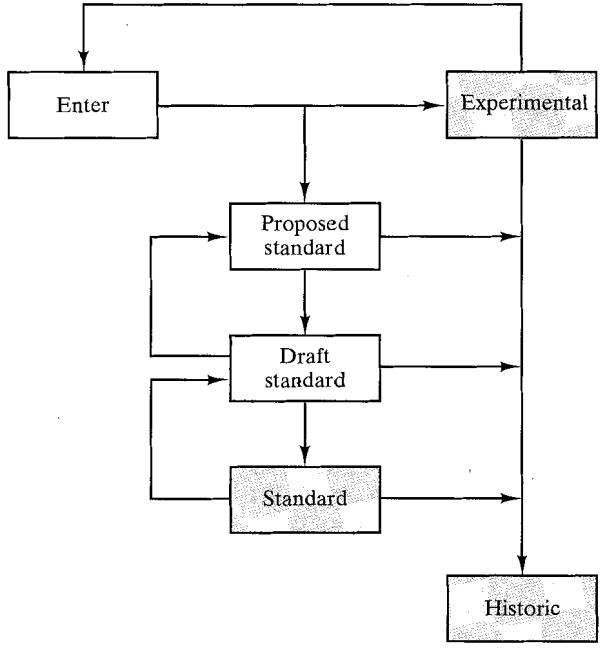


FIGURE 1.12 Standards track diagram.

After significant implementation and operational experience has been obtained, a specification may be elevated to standard. At this point, the specification is assigned an STD number as well as an RFC number.

Finally, when a protocol becomes obsolete, it is assigned to the historic state.

The International Organization for Standardization (ISO)

ISO is an international agency for the development of standards on a wide range of subjects. It is a voluntary, nontreaty organization whose members are designated standards bodies of participating nations, plus nonvoting observer organizations. Although ISO is not a governmental body, more than 70 percent of ISO member bodies are governmental standards institutions or organizations incorporated by public law. Most of the remainder have close links with the public administrations in their own countries. The United States member body is the American National Standards Institute.

ISO was founded in 1946 and has issued more than 5000 standards in a broad range of areas. Its purpose is to promote the development of standardization and related activities to facilitate international exchange of goods and services and to develop cooperation in the sphere of intellectual, scientific, technological, and economic activity. Standards have been issued to cover everything from screw threads to solar energy. One important area of standardization deals with the open systems interconnection (OSI) communications architecture and the standards at each layer of the OSI architecture.

In the areas of interest in this book, ISO standards are actually developed in a joint effort with another standards body, the International Electrotechnical Commission (IEC). IEC is primarily concerned with electrical and electronic engineering standards. In the area of information technology, the interests of the two groups overlap, with IEC emphasizing hardware and ISO focusing on software. In 1987, the two groups formed the Joint Technical Committee 1 (JTC 1). This committee has the responsibility of developing the documents that ultimately become ISO (and IEC) standards in the area of information technology.

The development of an ISO standard from first proposal to actual publication of the standard follows a seven-step process. The objective is to ensure that the final result is acceptable to as many countries as possible. The steps are briefly described here. (Time limits are the minimum time in which voting could be accomplished, and amendments require extended time.)

1. A new work item is assigned to the appropriate technical committee, and within that technical committee, to the appropriate working group. The working group prepares the technical specifications for the proposed standard and publishes these as a draft proposal (DP). The DP is circulated among interested members for balloting and technical comment. At least three months are allowed, and there may be iterations. When there is substantial agreement, the DP is sent to the administrative arm of ISO, known as the Central Secretariat.
2. The DP is registered at the Central Secretariat within two months of its final approval by the technical committee.
3. The Central Secretariat edits the document to ensure conformity with ISO practices; no technical changes are made. The edited document is then issued as a draft international standard (DIS).
4. The DIS is circulated for a six-month balloting period. For approval, the DIS must receive a majority approval by the technical committee members and 75 percent approval of all voting members. Revisions may occur to resolve any negative vote. If more than two negative votes remain, it is unlikely that the DIS will be published as a final standard.
5. The approved, possibly revised, DIS is returned within three months to the Central Secretariat for submission to the ISO Council, which acts as the board of directors of ISO.

6. The DIS is accepted by the Council as an international standard (IS).
7. The IS is published by ISO.

As can be seen, the process of issuing a standard is a slow one. Certainly, it would be desirable to issue standards as quickly as the technical details can be worked out, but ISO must ensure that the standard will receive widespread support.

ITU Telecommunications Standardization Sector

The ITU Telecommunications Standardization Sector (ITU-T) is a permanent organ of the International Telecommunication Union (ITU), which is itself a United Nations specialized agency. Hence, the members of ITU-T are governments. The U.S. representation is housed in the Department of State. The charter of the ITU is that it "is responsible for studying technical, operating, and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis." Its primary objective is to standardize, to the extent necessary, techniques and operations in telecommunications to achieve end-to-end compatibility of international telecommunication connections, regardless of the countries of origin and destination.

The ITU-T was created on March 1, 1993, as one consequence of a reform process within the ITU. It replaces the International Telegraph and Telephone Consultative Committee (CCITT), which had essentially the same charter and objectives as the new ITU-T.

ITU-T is organized into 15 study groups that prepare Recommendations:

1. Service Description
2. Network Operation
3. Tariff and Accounting Principles
4. Network Maintenance
5. Protection Against Electromagnetic Environment Effects
6. Outside Plant
7. Data Network and Open Systems Communications
8. Terminal Equipment and Protocols for Telematic Services
9. Television and Sound Transmission
10. Languages for Telecommunication Applications
11. Switching and Signalling
12. End-to-End Transmission Performance
13. General Network Aspects
14. Modems and Transmission Techniques for Data, Telegraph, and Telematic Services
15. Transmission Systems and Equipment

Work within ITU-T is conducted in four-year cycles. Every four years, a World Telecommunications Standardization Conference is held. The work program for the next four years is established at the assembly in the form of questions submitted by the various study groups, based on requests made to the study groups by their members. The conference assesses the questions, reviews the scope of the study groups, creates new or abolishes existing study groups, and allocates questions to these groups.

Based on these questions, each study group prepares draft Recommendations. A draft Recommendation may be submitted to the next conference, four years hence, for approval. Increasingly, however, Recommendations are approved when they are ready, without having to wait for the end of the four-year Study Period. This accelerated procedure was adopted after the study period that ended in 1988. Thus, 1988 was the last time that a large batch of documents was published at one time as a set of Recommendations.

1B APPENDIX

INTERNET RESOURCES

THERE ARE A number of resources available on the Internet for keeping up with developments in this field.

USENET Newsgroups

A number of USENET newsgroups are devoted to some aspect of data communications and networking. As with virtually all USENET groups, there is a high noise-to-signal ratio, but it is worth experimenting to see if any meet your needs. Here is a sample:

- comp.dcom.lans, comp.dcom.lans.misc: General discussions of LANs.
- comp.std.wireless: General discussion of wireless networks, including wireless LANs.
- comp.security.misc: Computer security and encryption.
- comp.dcom.cell-relay: Covers ATM and ATM LANs.
- comp.dcom.frame-relay: Covers frame-relay networks.
- comp.dcom.net-management: Discussion of network-management applications, protocols, and standards.
- comp.protocols.tcp-ip: The TCP/IP protocol suite.

Web Sites for This Book

A special web page has been set up for this book at http://www.shore.net/~ws/DCC5e.html The site includes the following:

- Links to other web sites, including the sites listed in this book, provide a gateway to relevant resources on the web.
- Links to papers and reports available via the Internet provide additional, up-to-date material for study.
- We also hope to include links to home pages for courses based on the book; these pages may be useful to other instructors in providing ideas about how to structure the course.
- Additional problems, exercises, and other activities for classroom use are also planned.

As soon as any typos or other errors are discovered, an errata list for this book will be available at http://www.shore.net/~ws/welcome.html. The file will be updated as needed. Please email any errors that you spot to ws@shore.net. Errata sheets for other books are at the same web site, as well as discount ordering information for the books.

Other Web Sites

There are numerous web sites that provide some sort of information related to the topics of this book. Here is a sample:

- http://www.soc.hawaii.edu/con/com-resources.html: Information and links to resources about data communications and networking.
- http://www.internic.net/ds/dspg01.html: Maintains archives that relate to the Internet and IETF activities. Includes keyword-indexed library of RFCs and draft documents as well as many other documents related to the Internet and related protocols.

- http://www.ronin.com/SBA: Links to over 1500 hardware and software vendors who currently have WWW sites, as well as a list of thousands of computer and networking companies in a Phone Directory.
- http://liinwww.ira.uka.de/bibliography/index.html: The Computer Science Bibliography Collection, a collection of hundreds of bibliographies with hundreds of thousands of references.

In subsequent chapters, pointers to more specific web sites can be found in the "Recommended Reading" section.