DATA COMMUNICATIONS

When we communicate, we are sharing information. This sharing can be local or remote. Between individuals, local communication usually occurs face to face, while remote communication takes place over distance. The term ***telecommunication,*** which includes telephony, telegraphy, and television, means communication at a distance (*tele* is Greek for "far"). The word ***data*** refers to information presented in whatever form is agreed upon by the parties creating and using the data.

Data communications are the exchange of data between two devices via some form of transmission medium such as a wire cable. For data communications to occur, the communicating devices must be part of a communication system made up of a combination of hardware (physical equipment) and software (programs). The effectiveness of a data communications system depends on four fundamental characteristics: delivery, accuracy, timeliness, and jitter.

1. **Delivery.** The system must deliver data to the correct destination. Data must be received by the intended device or user and only by that device or user.

2. **Accuracy.** The system must deliver the data accurately. Data that have been altered in transmission and left uncorrected are unusable.

3. **Timeliness.** The system must deliver data in a timely manner. Data delivered late are useless. In the case of video and audio, timely delivery means delivering data as they are produced, in the same order that they are produced, and without significant delay. This kind of delivery is called *real-time* transmission.

4. **Jitter.** Jitter refers to the variation in the packet arrival time. It is the uneven delay in the delivery of audio or video packets. For example, let us assume that video packets are sent every 30 ms. If some of the packets arrive with 30-ms delay and others with40-ms delay, an uneven quality in the video is the result

Components

A data communications system has five components (see Figure 1.1).

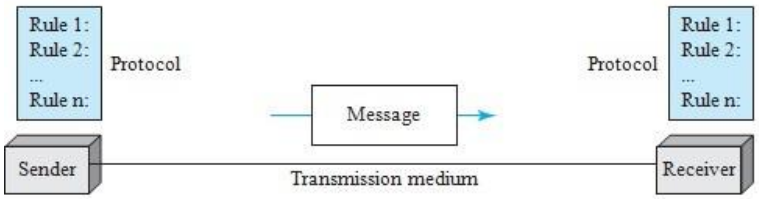


Figure 1.1 *Five components of data communication*

1. **Message.** The **message** is the information (data) to be communicated. Popularforms of information include text, numbers, pictures, audio, and video.

2. **Sender.** The **sender** is the device that sends the data message. It can be a computer, workstation, telephone handset, video camera, and so on.

3. **Receiver.** The **receiver** is the device that receives the message. It can be a computer, workstation, telephone handset, television, and so on.

4. **Transmission medium.** The **transmission medium** is the physical path by which a message travels from sender to receiver. Some examples of transmission media include twisted-pair wire, coaxial cable, fiber-optic cable, and radio waves.

5. **Protocol.** A protocol is a set of rules that govern data communications. It represents an agreement between the communicating devices. Without a protocol, two devices may be connected but not communicating, just as a person speaking French cannot be understood by a person who speaks only Japanese.

Data Representation

Information today comes in different forms such as text, numbers, images, audio, and video.

Text

In data communications, text is represented as a bit pattern, a sequence of bits (0s or 1s). Different sets of bit patterns have been designed to represent text symbols. Each set is called a **code,** and the process of representing symbols is called coding. Today, the prevalent coding system is called **Unicode,** which uses 32 bits to represent a symbol or character used in any language in the world. The **American Standard Code for Information Interchange (ASCII),** developed some decades ago in the United States, now constitutes the first 127 characters in Unicode and is also referred to as **Basic Latin.** Appendix A includes part of the Unicode.

Numbers

Numbers are also represented by bit patterns. However, a code such as ASCII is not usedto represent numbers; the number is directly converted to a binary number to simplify mathematical operations. Appendix B discusses several different numbering systems.

Images

Images are also represented by bit patterns. In its simplest form, an image is composed of a matrix of pixels (picture elements), where each pixel is a small dot. The size of the pixel depends on the resolution. For example, an image can be divided into 1000 pixelsor 10,000 pixels. In the second case, there is a better representation of the image (better resolution), but more memory is needed to store the image.

After an image is divided into pixels, each pixel is assigned a bit pattern. The size and the value of the pattern depend on the image. For an image made of only black- and-white dots (e.g., a chessboard), a 1-bit pattern is enough to represent a pixel.

If an image is not made of pure white and pure black pixels, we can increase the size of the bit pattern to include gray scale. For example, to show four levels of gray scale, we can use 2-bit patterns. A black pixel can be represented by 00, a dark gray pixel by 01, a light gray pixel by 10, and a white pixel by 11.

There are several methods to represent color images. One method is called RGB, so called because each color is made of a combination of three primary colors: red, green, and blue. The intensity of each color is measured, and a bit pattern is assigned to it. Another method is called **YCM,** in which a color is made of a combination of three other primary colors: *y*ellow, *c*yan, and *m*agenta.

Audio

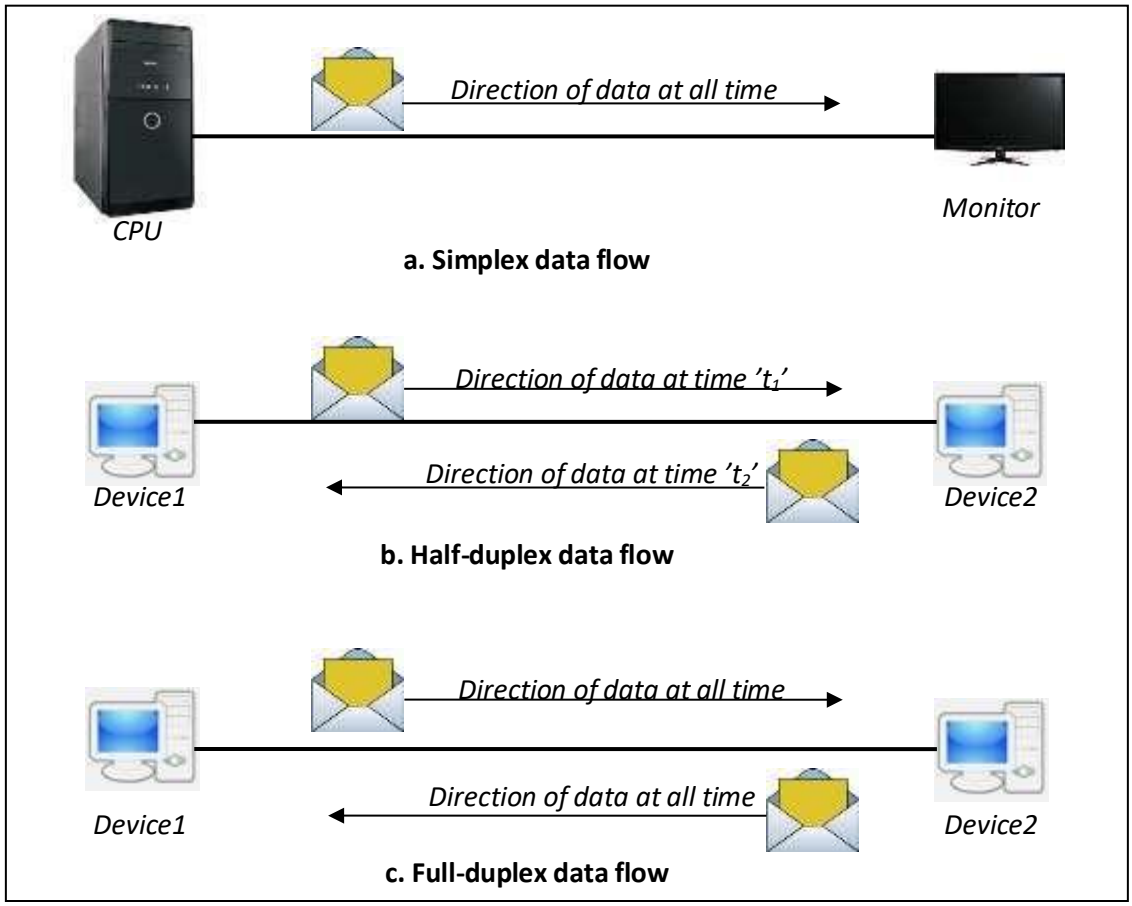
Audio refers to the recording or broadcasting of sound or music. Audio is by nature different from text, numbers, or images. It is continuous, not discrete. Even when we use a microphone to change voice or music to an electric signal, we create a continuoussignal.

Video

Video refers to the recording or broadcasting of a picture or movie. Video can either be produced as a continuous entity (e.g., by a TV camera), or it can be a combination of images, each a discrete entity, arranged to convey the idea of motion.

Data Flow

Communication between two devices can be simplex, half-duplex, or full-duplex as shown in Figure 1.2.



Simplex

In **simplex mode,** the communication is unidirectional, as on a one-way street. Only one of the two devices on a link can transmit; the other can only receive.

Keyboards and traditional monitors are examples of simplex devices. The key-board can only introduce input; the monitor can only accept output. The simplex modecan use the entire capacity of the channel to send data in one direction.

Half-Duplex

In **half-duplex mode,** each station can both transmit and receive, but not at the same time. When one device is sending, the other can only receive, and vice versa.

The half-duplex mode is like a one-lane road with traffic allowed in both directions. When cars are traveling in one direction, cars going the other way must wait. In a half-duplex transmission, the entire capacity of a channel is taken over by whichever ofthe two devices is transmitting at the time. Walkie-talkies and CB (citizens band) radios are both half-duplex systems.

The half-duplex mode is used in cases where there is no need for communication in both directions at the same time; the entire capacity of the channel can be utilized for each direction.

Full-Duplex

In **full-duplex mode** (also called *duplex*), both stations can transmit and receive simultaneously (see Figure 1.2c).

The full-duplex mode is like a two-way street with traffic flowing in both directions at the same time. In full-duplex mode, signals going in one direction share the

capacity of the link with signals going in the other direction. This sharing can occur in two ways: Either the link must contain two physically separate transmission paths, one for sending and the other for receiving; or the capacity of the channel is divided between signals traveling in both directions.

One common example of full-duplex communication is the telephone network. When two people are communicating by a telephone line, both can talk and listen at the same time.

The full-duplex mode is used when communication in both directions is required all the time. The capacity of the channel, however, must be divided between the two directions.

Networks

A network is set of interconnected devices (sometime referred as nodes) which are used to transmit data between them with agreed protocols. The networks are used to connect the people, machines, devices to share the data anywhere in the world. The devices can be computers, printers, mobile phones, servers which are capable of sending and receiving data. The data can be generated by a device.

There is considerable confusion in the literature between a computer network and a distributed system. The key distinction is that in a distributed system, a collection of independent computers appears to its users as a single coherent system. Usually, it has a single model or paradigm that it presents to the users. Often a layer of software on top of the operating system, called middleware, is responsible for implementing this model. A well-known example of a distributed system is the World Wide Web, in which everything looks like a document (Web page).

History of Network

A computer network is a digital telecommunications network which allows nodes to share resources. In computer networks, computing devices exchange data with each other using connections (data links) between nodes. These data links are established over cable media such as wires or optic cables, or wireless media such as Wi-Fi.

Computer networking as we know it today may be said to have gotten its start with the Arpanet development in the late 1960s and early 1970s. Prior to that time there were computer vendor "networks" designed primarily to connect terminals and remote job entry stations to a mainframe.

In 1940, George Sitbit used a teletype machine to send instructions for a problem set from his model at Dartmouth college to his complex number calculator in New York and received results back by the same means. In 1950's, early networks of communicating

Computers included the military radar system Semi-Automatic Ground Environment (SAGE)was started.

Later, in 1960s, the notion of networking between computers viewing each other as equal peers to achieve “resource sharing” was fundamental to the ARPA net design [1]. The other strong emphasis of the Arpanet work was its reliance on the then novel technique of packet switching to efficiently share communication resources among “bursty” users, instead of the more traditional message or circuit switching. The table 1.1 gives the time frame of the computer network growth from network to internet.

Year	Event
1961	The idea of ARPANET, one of the earliest computer networks, was proposed by Leonard Kleinrockin 1961, in his paper titled "Information Flow in Large Communication Nets."
1965	The term "packet" was coined by Donald Davies in 1965, to describe data sent between computers over a network.
1969	ARPANET was one of the first computer networks to use packet switching. Development of ARPANET started in 1966, and the first two nodes, UCLA and SRI (Standford Research Institute), were connected, officially starting ARPANET in 1969.
1969	The first RFC surfaced in April 1969, as a document to define and provide information about computer communications, network protocols, and procedures.
1969	The first network switch and IMP (Interface Message Processor) was sent to UCLA on August 29, 1969. It was used to send the first data transmission on ARPANET.
1969	The Internet was officially born, with the first data transmission being sent between UCLA and SRIon October 29, 1969, at 10:30 p.m.
1970	Steve Crocker and a team at UCLA released NCP (NetWare Core Protocol) in 1970. NCP is a file sharing protocol for use with NetWare.
1971	Ray Tomlinson sent the first e-mail in 1971.
1971	ALOHAnet, a UHF wireless packet network, is used in Hawaii to connect the islands together. Although it is not Wi-Fi, it helps lay the foundation for Wi-Fi.
1973	Ethernet is developed by Robert Metcalfe in 1973 while working at Xerox PARC.
1973	The first international network connection, called SATNET, is deployed in 1973 by ARPA.
1973	An experimental VoIP call was made in 1973, officially introducing VoIP technology and capabilities. However, the first software allowing users to make VoIP calls was not available until 1995.
1974	The first routers were used at Xerox in 1974. However, these first routers were not considered true IP routers.
1976	Ginny Strazisar developed the first true IP router, originally called a gateway, in 1976.
1978	Bob Kahn invented the TCP/IP protocol for networks and developed it, with help from Vint Cerf, in 1978.
1981	Internet protocol version 4, or IPv4, was officially defined in RFC 791 in 1981. IPv4 was the first major version of the Internet protocol.
1981	BITNET was created in 1981 as a network between IBM mainframe systems in the United States.
1981	CSNET (Computer Science Network) was developed by the U.S. National Science Foundation in 1981.
1983	ARPANET finished the transition to using TCP/IP in 1983.
1983	Paul Mockapetris and Jon Postel implement the first DNS in 1983.
1986	The NSFNET (National Science Foundation Network) came online in 1986. It was a backbone for ARPANET, before eventually replacing ARPANET in the early 1990's.
1986	BITNET II was created in 1986 to address bandwidth issues with the original BITNET.
1988	The first T1 backbone was added to ARPANET in 1988.

1988	WaveLAN network technology, the official precursor to Wi-Fi, was introduced to the market by AT&T, Lucent, and NCR in 1988.
1988	Details about network firewall technology was first published in 1988. The published paper discussed the first firewall, called a packet filter firewall, that was developed by Digital Equipment Corporation the same year.
1990	Kalpana, a U.S. network hardware company, developed and introduced the first network switch in 1990.
1996	IPv6 was introduced in 1996 as an improvement over IPv4, including a wider range of IP addresses, improved routing, and embedded encryption.
1997	The first version of the 802.11 standard for Wi-Fi is introduced in June 1997, providing transmission speeds up to 2 Mbps.
1999	The 802.11a standard for Wi-Fi was made official in 1999, designed to use the 5 GHz band and provide transmission speeds up to 25 Mbps.
1999	802.11b devices were available to the public starting mid-1999, providing transmission speeds up to11 Mbps.
1999	The WEP encryption protocol for Wi-Fi is introduced in September 1999, for use with 802.11b.
2003	802.11g devices were available to the public starting in January 2003, providing transmission speedsup to 20 Mbps.
2003	The WPA encryption protocol for Wi-Fi is introduced in 2003, for use with 802.11g.
2003	The WPA2 encryption protocol is introduced in 2004, as an improvement over and replacement for WPA. All Wi-Fi devices are required to be WPA2 certified by 2006.
2009	The 802.11n standard for Wi-Fi was made official in 2009. It provides higher transfer speeds over 802.11a and 802.11g, and it can operate on the 2.4 GHz and 5 GHz bandwidths.
2018	The Wi-Fi Alliance introduced WPA3 encryption for Wi-Fi in January 2018, which includes security enhancements over WPA2.

Table 1.1 Time period of Network growth

Uses of Computer Networks

The computer networks are used in different applications to meet the requirement of different people at different places in different time. The following are the uses of computer network.

a) Business Applications.

Many companies have a substantial number of computers. For example, a company may have separate computers to monitor production, keep track of inventories, and do the payroll. Initially, each of these computers may have worked in isolation from the others, but at some point, management may have decided to connect them to be able to extract and correlate information about the entire company.

1. **Resource sharing**: The main task of the connectivity of resources is resource sharing. For example, a high-volume networked printer may be installed instead of large collection of individual printers.
2. **Information Sharing** : large and medium-sized company and many small companies are vitally dependent on computerized information. This can be done by a simple client server model connected by network as illustrated in Fig.1.4.

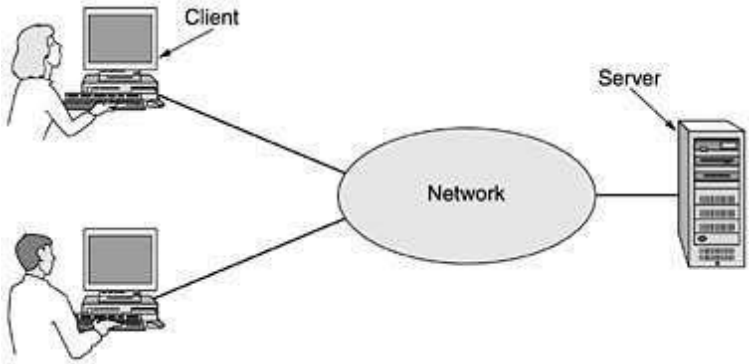


Figure 1.4 A network with two clients and one server

In client-server model in detail, two processes are involved, one on the client machine and one on the server machine. Communication takes the form of the client process sending a message over the network to the server process. The client process then waits for a reply message. When the server process gets the request, it performs the requested work or looks up the requested data and sends back a reply. These messagesare shown in Fig. 1.5.

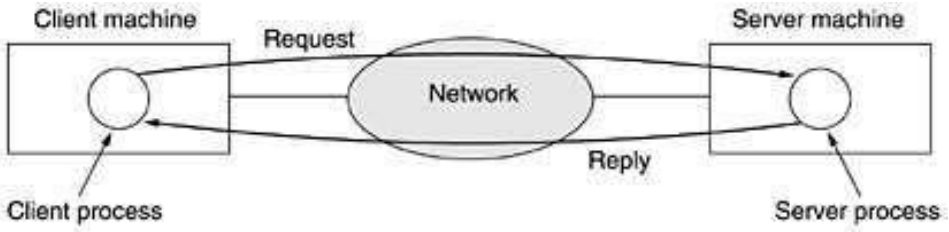


Figure 1.5 Client-server model involves requests and replies

3. **Connecting People** : another use of setting up a computer network has to do with people rather than information or even computers. It is achieved through Email, Video Conferencing.

4. **E-commerce** : many companies is doing business electronically with other companies, especially suppliers and customers, and doing business with consumers over the Internet.

b) Home Applications

The computer network provides better connectivity for home applications via desktop computers, laptops, iPads, iPhones. Some of the more popular uses of the Internet for home users are as follows:

1. Access to remote information.
2. Person-to-person communication (peer-to-peer).
    i. Peer-to-peer - there are no fixed clients and servers.
    ii. Audio and Video sharing
3. Interactive entertainment.
4. Electronic commerce.

Tag	Full name	Example
B2C	Business-to-consumer	Ordering books on-line
B2B	Business-to-business	Car manufacturer ordering tires from supplier
G2C	Government-to-consumer	Government distributing tax forms electronically
C2C	Consumer-to-consumer	Auctioning second-hand products on line
P2P	Peer-to-peer	File sharing

Table 1.2 Some forms of e-commerce

c) Mobile Users

As wireless technology becomes more widespread, numerous other applications are likely to emerge. Wireless networks are of great value to fleets of trucks, taxis, delivery vehicles, and repairpersons for keeping in contact with home. Wireless networks are also important to the military.

Although wireless networking and mobile computing are often related, they are not identical, as Table 1.3 shows. Here we see a distinction between fixed wireless and mobile wireless. Even notebook computers are sometimes wired. For example, if a traveller plugs a notebook computer into the telephone jack in a hotel room, he has mobility without a wirelessnetwork.

Wireless	Mobile	Applications
No	No	Desktop computers in offices
No	Yes	A notebook computer used in a hotel room
Yes	No	Networks in older, unwired buildings
Yes	Yes	Portable office; PDA for store inventory

Table 1.3 Combinations of wireless networks and mobile computing

Another area in which wireless could save money is utility meter reading. If electricity, gas, water, and other meters in people's homes were to report usage over a wireless network, there would be no need to send out meter readers.

d) Social issues

The widespread introduction of networking has introduced new social, ethical, and political problems. A popular feature of many networks are newsgroups or bulletin boards whereby people can exchange messages with like-minded individuals. As long as the subjects are restricted to technical topics or hobbies like gardening, not too many problems will arise.

The following are the issues in society due to the misbehave or misconduct of computer networks.

1. Network neutrality
2. Digital Millennium Copyright Act
3. Profiling users

4. Phishing

Network Criteria

A network must be able to meet a certain number of criteria. The most important of these are performance, reliability, and security.

Performance

Performance can be measured in many ways, including transit time and response time. Transit time is the amount of time required for a message to travel from one device to another. Response time is the elapsed time between an inquiry and a response. The performance of a network depends on a number of factors, including the number of users, the type of transmission medium, the capabilities of the connected hardware, and the efficiency of the software.

Performance is often evaluated by two networking metrics: **throughput** and **delay.** We often need more throughput and less delay. However, these two criteria are often contradictory. If we try to send more data to the network, we may increase throughput but we increase the delay because of traffic congestion in the network.

Reliability

In addition to accuracy of delivery, network **reliability** is measured by the frequency of failure, the time it takes a link to recover from a failure, and the network's robustness ina catastrophe.

Security

Network **security** issues include protecting data from unauthorized access, protecting data from damage and development, and implementing policies and procedures for recovery from breaches and data losses.

Type of Connection

A network is two or more devices connected through links. A link is a communications pathway that transfers data from one device to another. For visualization purposes, it is simplest to imagine any link as a line drawn between two points. For communication to occur, two devices must be connected in some way to the same link at the same time. There are two possible types of connections: point-to-point and multipoint.

Point-to-Point: A **point-to-point connection** provides a dedicated link between two devices. The entire capacity of the link is reserved for transmission between those two devices. Most point-to-point connections use an actual length of wire or cable to connect the two ends, but other options, such as microwave or satellite links, are also possible (see Figure 1.3a). When we change television channels by infrared remote control, we are establishing a point-to-point connection between the remote control and the television's control system.

Multipoint:

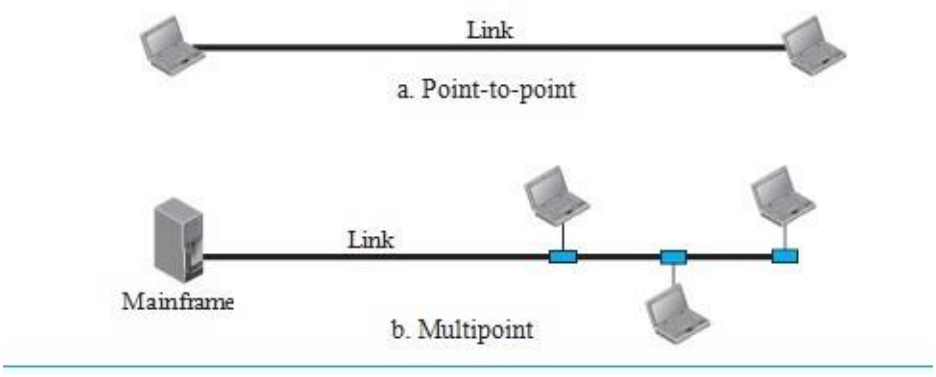


Figure 1.3 *Types of connections: point-to-point and multipoint*

A **multipoint** (also called **multidrop**) **connection** is one in which more than two specific devices share a single link (see Figure 1.3b). In a multipoint environment, the capacity of the channel is shared, either spatially or temporally. If several devices can use the link simultaneously, it is a *spatially shared*connection. If users must take turns, itis a *timeshared* connection.

NETWORK TYPES

After defining networks in the previous section and discussing their physical structures, we need to discuss different types of networks we encounter in the world today. The criteria of distinguishing one type of network from another is difficult and sometimes confusing. We use a few criteria such as size, geographical coverage, and ownership to make this distinction. After discussing two types of networks, LANs and WANs, we define switching, which is used to connect networks to form an internetwork (a network of networks).

Personal Area Networks

PANs (Personal Area Networks) let devices communicate over the range of a person. A common example is a wireless network that connects a computer with its peripherals. Almost every computer has an attached monitor, keyboard, mouse, and printer. Without using wireless, this connection must be done with cables.

- **Bluetooth:** some companies got together to design a short-range wireless network called *Bluetooth* to connect these components without wires.

The idea is that if your devices have Bluetooth, then you need no cables. You just put them down, turn them on, and they work together. For many people, this ease of operation is a big plus.

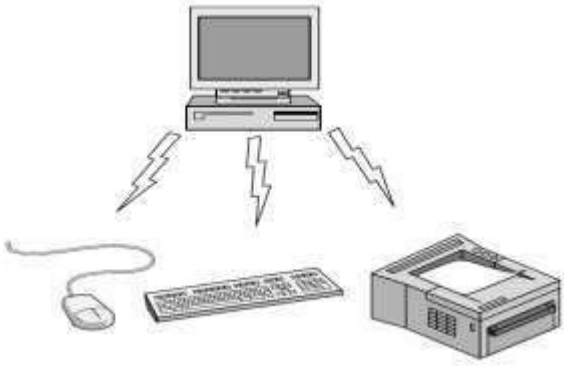


Figure 1.8 Bluetooth PAN configuration.

Bluetooth networks use the master-slave paradigm of Figure. 1.8. The system unit (the PC) is normally the master, talking to the mouse, keyboard, etc., as slaves. The master tells the slaves what addresses to use, when they can broadcast, how long they can transmit, what frequencies they can use, and so on.

Bluetooth can be used in other settings, too. It is often used to connect a headset to a mobile phone without cords and it can allow your digital music player A completely different kind of **PAN** is formed when an embedded medical device such as a pacemaker, insulin pump, or hearing aid talks to a user-operated remote control.

PANs can also be built with other technologies that communicate over short ranges, such as RFID o smartcards and library books

Local Area Network

A **local area network** (**LAN**) is usually privately owned and connects some hosts in a single office, building, or campus. Depending on the needs of an organization, a LAN can be as simple as two PCs and a printer in someone's home office, or it can extend throughout a company and include audio and video devices. Each host in a LAN has an identifier, an address, that uniquely defines the host in the LAN. A packet sent by a host to another host carries both the source host's and the destination host's addresses.

In the past, all hosts in a network were connected through a common cable, which meant that a packet sent from one host to another was received by all hosts. The intended recipient kept the packet; the others dropped the packet. Today, most LANs use a smart connecting switch, which is able to recognize the destination address of the packet and guide the packet to its destination without sending it to all other hosts. The switch alleviates the traffic in the LAN and allows more than one pair to communicate with each other at the same time if there is no common source and destination among them. Note that the above definition of a LAN does not define the minimum or maximum number of hosts in a LAN. Figure 1.8 shows a LAN using either a common cable or a switch.

When LANs were used in isolation (which is rare today), they were designed to allow resources to be shared between the hosts. As we will see shortly, LANs today are connectedto each other and to WANs (discussed next) to create communication at a widerlevel.

Metropolitan Area Networks

A **MAN** (**Metropolitan Area Network**) covers a city. The best-known examples of MANs are the cable television networks available in many cities. These systems grew from earlier community antenna systems used in areas with poor over-the-air television reception.

In those early systems, a large antenna was placed on top of a nearby hill and a signal was then piped to the subscribers' houses. At first, these were locally designed, ad hoc systems. Then companies began jumping into the business, getting contracts from local governments to wire up entire cities.

The next step was television programming and even entire channels designed for cable only. Often these channels were highly specialized, such as all news, all sports, all cooking, all gardening, and so on.

When the Internet began attracting a mass audience, the cable TV network operators began to realize that with some changes to the system, they could provide two-way Internet service in unused parts of the spectrum. At that point, the cable TV system began to morph from simply a way to distribute television to a metropolitan area network.

A MAN might look something like the system shown in Fig. 1.10. In this figure we see both television signals and Internet being fed into the centralized **cable head end** for subsequent distribution to people's homes.

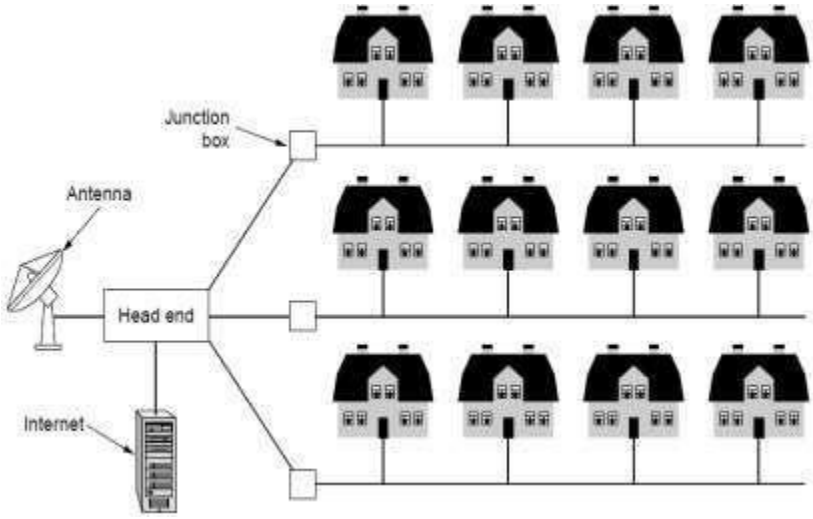


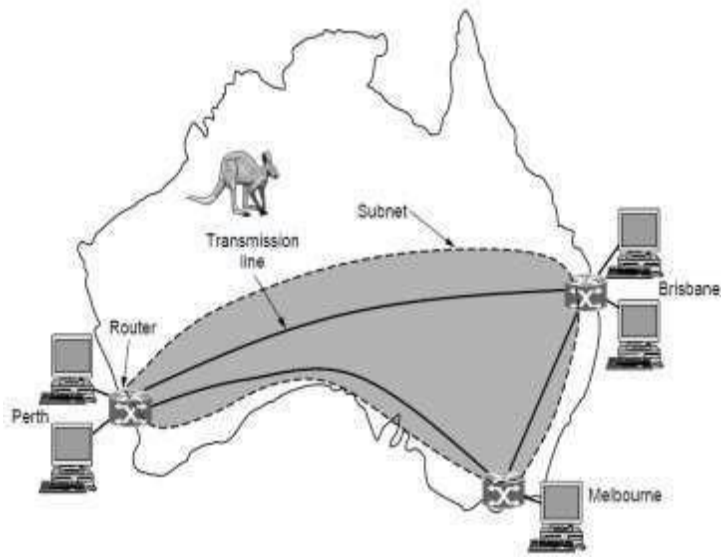
Figure 1.10 A metropolitan area network based on cable TV.

Recent developments in highspeed wireless Internet access have resulted in another MAN, which has been standardized as IEEE 802.16 and is popularly known as **WiMAX**.

Wide Area Networks

A **WAN** (**Wide Area Network**) spans a large geographical area, often a country or continent. We will begin our discussion with wired WANs, using the example of a company with branch offices in different cities.

The WAN in Fig.1.11 is a network that connects offices in Perth, Melbourne, and Brisbane. Each of these offices contains computers intended for running user (i.e., application) programs. We will follow traditional usage and call these machines **hosts**.



The rest of the network that connects these hosts is then called the **communication subnet**, or just **subnet** for short. The job of the subnet is to carry messages from host to host, just as the telephone system carries words (really just sounds) from speaker to listener.

PROTOCOL LAYERING

In data communication and networking, a protocol defines the rules that both the sender and receiver and all intermediate devices need to follow to be able to communicate effectively. When communication is simple, we may need only one simple protocol; when the communication is complex, we may need to divide the task between different layers, in which case we need a protocol at each layer, or **protocol layering.**

Scenarios

Let us develop two simple scenarios to better understand the need for protocol layering.

First Scenario

In the first scenario, communication is so simple that it can occur in only one layer. Assume Maria and Ann are neighbors with a lot of common ideas. Communication between Maria and Ann takes place in one layer, face to face, in the same language, asshown in Figure 1.16.

Figure 1.16 *A single-layer protocol*

Even in this simple scenario, we can see that a set of rules needs to be followed. First, Maria and Ann know that they should greet each other when they meet. Second, they know that they should confine their vocabulary to the level of their friendship. Third, each party knows that she should refrain from speaking when the other party is speaking. Fourth, each party knows that the conversation should be a dialog, not a monolog: both should have the opportunity to talk about the issue. Fifth, they should exchange some nice words when they leave.

We can see that the protocol used by Maria and Ann is different from the communication between a professor and the students in a lecture hall. The communication in the second case is mostly monolog; the professor talks most of the time unless a studenthas a question, a situation in which the protocol dictates that she should raise her hand and wait for permission to speak. In this case, the communication is normally very for-mal and limited to the subject being taught.

Second Scenario

In the second scenario, we assume that Ann is offered a higher-level position in her company, but needs to move to another branch located in a city very far from Maria. The two friends still want to continue their communication and exchange ideas because they have come up with an innovative project to start a new business when they both retire.

They decide to continue their conversation using regular mail through the post office. However, they do not want their ideas to be revealed by other people if the letters are intercepted. They agree on an encryption/decryption technique. The sender of the letter encrypts it to make it unreadable by an intruder; the receiver of the letter decrypts it to get the original letter., but for the moment we assume that Maria and Ann use one technique that makes it hard to decrypt the letter if one does not have the key for doing so. Now we can say that the communication between Maria and Ann takes place in three layers, as shown in Figure 1.17. We assume that Ann and Maria each have three machines (or robots) that can perform the task at each layer.

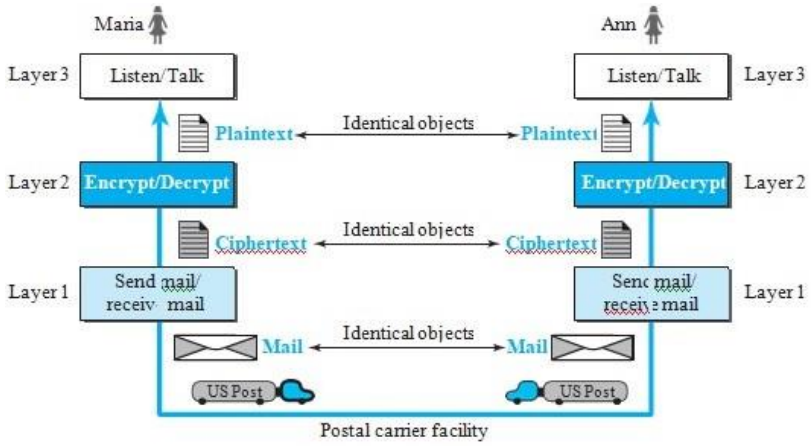


Figure 1.17 *A three-layer protocol*

Let us assume that Maria sends the first letter to Ann. Maria talks to the machine at the third layer as though the machine is Ann and is listening to her. The third layer machine listens to what Maria says and creates the plaintext (a letter in English), whichis passed to the second layer machine. The second layer machine takes the plaintext, encrypts it, and creates the cipher text, which is passed to the first layer machine. The first layer machine, presumably a robot, takes the cipher text, puts it in an envelope, adds the sender and receiver addresses, and mails it.

At Ann's side, the first layer machine picks up the letter from Ann's mail box, recognizing the letter from Maria by the sender address. The machine takes out the cipher- text from the envelope and delivers it to the second layer machine. The second layer machine decrypts the message, creates the plaintext, and passes the plaintext to the third-layer machine. The third layer machine takes the plaintext and reads it as though Maria is speaking.

Protocol layering enables us to divide a complex task into several smaller and simpler tasks. For example, in Figure 1.17, we could have used only one machine to do the job of all three machines. However, if Maria and Ann decide that the encryption/ decryption done by the machine is not enough to protect their secrecy, they would have to change the whole machine. In the present situation, they need to change only the second layer machine; the other two can remain the same. This is referred to as *modularity*. Modularity in this case means independent layers. A layer (module) can be defined as a black box with inputs and outputs, without concern about how inputs are changed to outputs. If two machines provide the same outputs when given the same inputs, they can replace each other. For example, Ann and Maria can buy the second layer machinefrom two different manufacturers. As long as the two machines create the same cipher- text from the same plaintext and vice versa, they do the job.

One of the advantages of protocol layering is that it allows us to separate the

services from the implementation. A layer needs to be able to receive a set of ser-

vices from the lower layer and to give the services to the upper layer; we don't care about how the layer is implemented. For example, Maria may decide not to buy the machine (robot) for the first layer; she can do the job herself. As long as Maria can do the tasks provided by the first layer, in both directions, the communication system works.

Another advantage of protocol layering, which cannot be seen in our simple examples but reveals itself when we discuss protocol layering in the Internet, is that communication does not always use only two end systems; there are intermediate systems that need only some layers, but not all layers. If we did not use protocol layering, we would have to make each intermediate system as complex as the end systems, which makes the whole system more expensive.

Is there any disadvantage to protocol layering? One can argue that having a single layer makes the job easier. There is no need for each layer to provide a service to the upper layer and give service to the lower layer. For example, Ann and Maria could find or build one machine that could do all three tasks. However, as mentioned above, if one day they found that their code was broken, each would have to replace the whole machine with a new one instead of just changing the machine in the second layer.

Principles of Protocol Layering Let us

discuss two principles of protocol layering.**First**

Principle

The first principle dictates that if we want bidirectional communication, we need to make each layer so that it is able to perform two opposite tasks, one in each direction. For example, the third layer task is to listen (in one direction) and *talk* (in the other direction). The second layer needs to be able to encrypt and decrypt. The first layer needs to send and receive mail.

Second Principle

The second principle that we need to follow in protocol layering is that the two objects under each layer at both sites should be identical. For example, the object under layer 3 at both sites should be a plaintext letter. The object under layer 2 at both sites should be a cipher text letter. The object under layer 1 at both sites should be a piece of mail.

Logical Connections

After following the above two principles, we can think about logical connection between each layer as shown in Figure 1.18. This means that we have layer-to-layer communication. Maria and Ann can think that there is a logical (imaginary) connectionat

each layer through which they can send the object created from that layer. We will see that the concept of logical connection will help us better understand the task of lay-ering we encounter in data communication and networking.

Figure 1.18 *Logical connection between peer layers*

TCP/IP PROTOCOL SUITE

Now that we know about the concept of protocol layering and the logical communication between layers in our second scenario, we can introduce the TCP/IP (Transmission Control Protocol/Internet Protocol). TCP/IP is a protocol suite (a set of protocols organized in different layers) used in the Internet today. It is a hierarchical protocol made upof interactive modules, each of which provides a specific functionality. The term *hierarchical* means that each upper level protocol is supported by the services provided by one or more lower level protocols. The original TCP/IP protocol suite was defined as four software layers built upon the hardware. Today, however, TCP/IP is thought of as afive-layer model. Figure 1.19. shows both configurations.

Layered Architecture

To show how the layers in the TCP/IP protocol suite are involved in communication between two hosts, we assume that we want to use the suite in a small internet made upof three LANs (links), each with a link-layer switch. We also assume that the links are connected by one router, as shown in Figure 1.20.

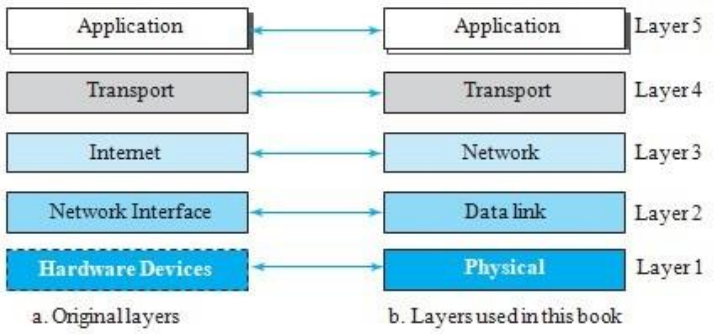


Figure 1.19 *Layers in the TCP/IP protocol suite*

Let us assume that computer A communicates with computer B. As the figure shows, we have five communicating devices in this communication: source host (computer A), the link-layer switch in link 1, the router, the link-layer switch in link 2, and the destination host (computer B). Each device is involved with a set of layers depending on the role of the device in the internet. The two hosts are involved in all fivelayers; the source host needs to create a message in the application layer and send it down the layers so that it is physically sent to the destination host. The destination host needs to receive the communication at the physical layer and then deliver itthrough theother layers to the application layer.

The router is involved in only three layers; there is no transport or application layer in a router as long as the router is used only for routing. Although a router is always involved in one network layer, it is involved in *n* combinations of link and physical layers in which *n* is the number of links the router is connected to. The reason is that each link may use its own data-link or physical protocol. For example, in the above figure, the router is involved in three links, but the message sent from source A to destination B is involved in two links. Each link may be using different link-layer and physical-layer protocols; the router needs to receive a packet from link 1 based on one pair of protocols and deliver it to link 2 based on another pair of protocols.

A link-layer switch in a link, however, is involved only in two layers, data-link and physical. Although each switch in the above figure has two different connections, the connections are in the same link, which uses only one set of protocols. This means that, unlike a router, a link-layer switch is involved only in one data-link and one physical layer.

Description of Each Layer

After understanding the concept of logical communication, we are ready to briefly discuss the duty of each layer.

Physical Layer

We can say that the physical layer is responsible for carrying individual bits in a frame across the link. Although the physical layer is the lowest level in the TCP/IP protocol suite, the communication between two devices at the physical layer is still a logical communication because there is another, hidden layer, the transmission media, under the physical layer. Two devices are connected by a transmission medium (cable or air). We need to know that the transmission medium does not carry bits; it carries electrical or optical signals. So the bits received in a frame from the data-link layer are trans- formed and sent through the transmission media, but we can think that the logical unit between two physical layers in two devices is a *bit*. There are several protocols that transform a bit to a signal. We discuss them in Part II when we discuss the physical layer and the transmission media.

Data-link Layer

We have seen that an internet is made up of several links (LANs and WANs) connected

by routers. There may be several overlapping sets of links that a datagram can travel from the host to the destination. The routers are responsible for choosing the *best* links. However, when the next link to travel is determined by the router, the data-link layer is responsible for taking the datagram and moving it across the link. The link can be a wired LAN with a link-layer switch, a wireless LAN, a wired WAN, or a wireless WAN. We can also have different protocols used with any link type. In each case, the data-link layer is responsible for moving the packet through the link.

TCP/IP does not define any specific protocol for the data-link layer. It supports all the standard and proprietary protocols. Any protocol that can take the datagram and carry it through the link suffices for the network layer. The data-link layer takes a datagram and encapsulates it in a packet called a *frame*.

Each link-layer protocol may provide a different service. Some link-layer protocols provide complete error detection and correction, some provide only error correction.

Network Layer

The network layer is responsible for creating a connection between the source computer and the destination computer. The communication at the network layer is host-to-host. However, since there can be several routers from the source to the destination, the routersin the path are responsible for choosing the best route for each packet. We can say that the network layer is responsible for host-to-host communication and routing the packet through possible routes. Again, we may ask ourselves why we need the network layer. We could have added the routing duty to the transport layer and dropped this layer. One reason, as we said before, is the separation of different tasks between different layers. The second reason is that the routers do not need the application and transport layers. Separating the tasks allows us to use fewer protocols on the routers.

The network layer in the Internet includes the main protocol, Internet Protocol (IP), that defines the format of the packet, called a datagram at the network layer. IP also defines the format and the structure of addresses used in this layer. IP is also responsible for routing a packet from its source to its destination, which is achieved by each router forwarding the datagram to the next router in its path.

IP is a connectionless protocol that provides no flow control, no error control, and no congestion control services. This means that if any of theses services is required for an application, the application should rely only on the transport-layer protocol. The net- work layer also includes unicast (one-to-one) and multicast (one-to-many) routing protocols. A routing protocol does not take part in routing (it is the responsibility of IP), but it creates forwarding tables for routers to help them in the routing process.

The network layer also has some auxiliary protocols that help IP in its delivery and routing tasks. The Internet Control Message Protocol (ICMP) helps IP to report some problems when routing a packet. The Internet Group Management Protocol (IGMP) is another protocol that helps IP in multitasking. The Dynamic Host Configuration Protocol (DHCP) helps IP to get the network-layer address for a host. The Address Resolution Protocol (ARP) is a protocol that helps IP to find the link-layer address of a host or a router when its network-layer address is given.

Transport Layer

The logical connection at the transport layer is also end-to-end. The transport layer at the source host gets the message from the application layer, encapsulates it in a transport- layer packet (called a *segment* or a *user datagram* in different protocols) and sends it, through the logical (imaginary) connection, to the transport layer at the destination host. In other words, the transport layer is responsible for giving services to the application layer: to get a message from an application program running on the source host and deliver it to the corresponding application program on the destination host. We may askwhy we need an end-to-end transport layer when we already have an end-to-end application layer. The reason is the separation of tasks and duties, which we discussed earlier. The transport layer should be independent of the application layer. In addition, we will see that we have more than one protocol in the transport layer, which means that each application program can use the protocol that best matches its requirement.

As we said, there are a few transport-layer protocols in the Internet, each designed for some specific task. The main protocol, Transmission Control Protocol (TCP), is a connection-oriented protocol that first establishes a logical connection between trans- port layers at two hosts before transferring data. It creates a logical pipe between two TCPs for transferring a stream of bytes. TCP provides flow control (matching the send- ing data rate of the source host with the receiving data rate of the destination host to prevent overwhelming the destination), error control (to guarantee that the segments arrive at the destination without error and resending the corrupted ones), and conges- tion control to reduce the loss of segments due to congestion in the network. The other common protocol, User Datagram Protocol (UDP), is a connectionless protocol that transmits user datagrams without first creating a logical connection. In UDP, each user datagram is an independent entity without being related to the previous or the next one(the meaning of the term *connectionless*). UDP is a simple protocol that does not pro- vide flow, error, or congestion control. Its simplicity, which means small overhead, is attractive to an application program that needs to send short messages and cannot afford the retransmission of the packets involved in TCP, when a packet is corrupted or lost. A new protocol, Stream Control Transmission Protocol (SCTP) is designed to respond to new applications that are emerging in the multimedia.

Application Layer

As Figure 2.6 shows, the logical connection between the two application layers is end- to-end. The two application layers exchange *messages* between each other as though there were a bridge between the two layers. However, we should know that the communication is done through all the layers.

Communication at the application layer is between two *processes* (two programs running at this layer). To communicate, a process sends a request to the other process

and receives a response. Process-to-process communication is the duty of the application layer. The application layer in the Internet includes many predefined protocols, but a user can also create a pair of processes to be run at the two hosts.

The Hypertext Transfer Protocol (HTTP) is a vehicle for accessing the World Wide Web (WWW). The Simple Mail Transfer Protocol (SMTP) is the main protocol used in electronic mail (e-mail) service. The File Transfer Protocol (FTP) is used for transferring files from one host to another. The Terminal Network (TELNET) and Secure Shell (SSH) are used for accessing a site remotely. The Simple Network Management Protocol (SNMP) is used by an administrator to manage the Internet at global and local levels. The Domain Name System (DNS) is used by other protocols to find the network-layer address of a computer. The Internet Group Management Protocol (IGMP) is used to collect membership in a group.

THE OSI MODEL

Although, when speaking of the Internet, everyone talks about the TCP/IP protocol suite, this suite is not the only suite of protocols defined. Established in 1947, the **International Organization for Standardization (ISO)** is a multinational body dedicated to worldwide agreement on international standards. Almost three-fourths of the countries in the world are represented in the ISO. An ISO standard that covers all aspects of network communications is the **Open Systems Interconnection (OSI) model.** It was first introduced in the late 1970s.

An *open system* is a set of protocols that allows any two different systems to communicate regardless of their underlying architecture. The purpose of the OSI model is to show how to facilitate communication between different systems without requiring changes to the logic of the underlying hardware and software. The OSI model is not a protocol; it is a model for understanding and designing a network architecture that is flexible, robust, and interoperable. The OSI model was intended to be the basis for the creation of the protocols in the OSI stack.

The OSI model is a layered framework for the design of network systems that allows communication between all types of computer systems. It consists of seven sep- arate but related layers, each of which defines a part of the process of moving informationacross a network (see Figure 1.26).

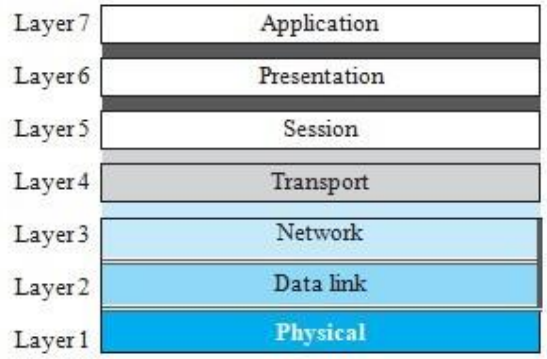


Figure 1.26 *The OSI model*

OSI versus TCP/IP

When we compare the two models, we find that two layers, session and presentation, are missing from the TCP/IP protocol suite. These two layers were not added to the TCP/IP protocol suite after the publication of the OSI model. The application layer in the suite is usually considered to be the combination of three layers in the OSI model, as shown in Figure 1.27.

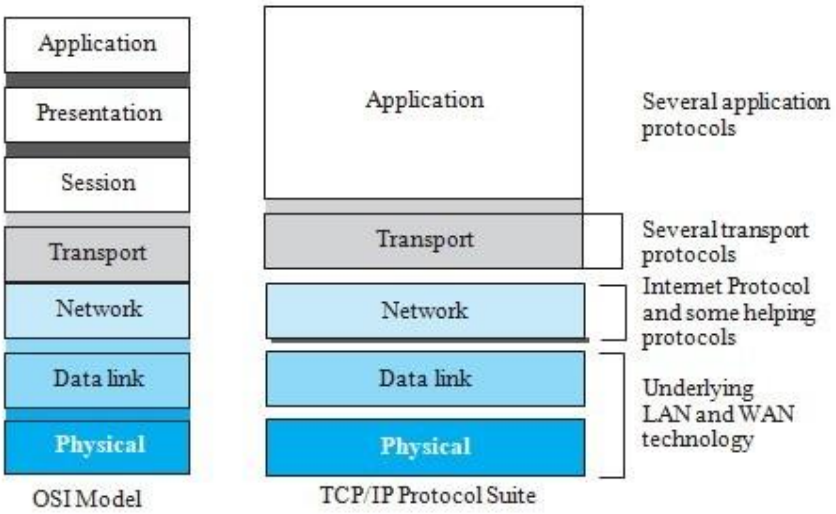


Figure 1.27 *TCP/IP and OSI model*

Two reasons were mentioned for this decision. First, TCP/IP has more than one transport-layer protocol. Some of the functionalities of the session layer are available in some of the transport-layer protocols. Second, the application layer is not only one piece of software. Many applications can be developed at this layer. If some of the functionalities mentioned in the session and presentation layers are needed for a particular application, they can be included in the development of that piece of software.

Lack of OSI Model's Success

The OSI model appeared after the TCP/IP protocol suite. Most experts were at first excited and thought that the TCP/IP protocol would be fully replaced by the OSI model. This did not happen for several reasons, but we describe only three, which are agreed upon by all experts in the field. First, OSI was completed when TCP/IP was fully in place and a lot of time and money had been spent on the suite; changing it would cost a lot. Second, some layers in the OSI model were never fully defined. For example, although the services provided by the presentation and the session layers were listed in the document, actual protocols for these two layers were not fully defined, nor were they fully described, and the corresponding software was not fully developed. Third, when OSI was implemented by an organization in a different application, it did not show a high enough level of performance to entice the Internet authority to switch from the TCP/IP protocol suite to the OSI model.