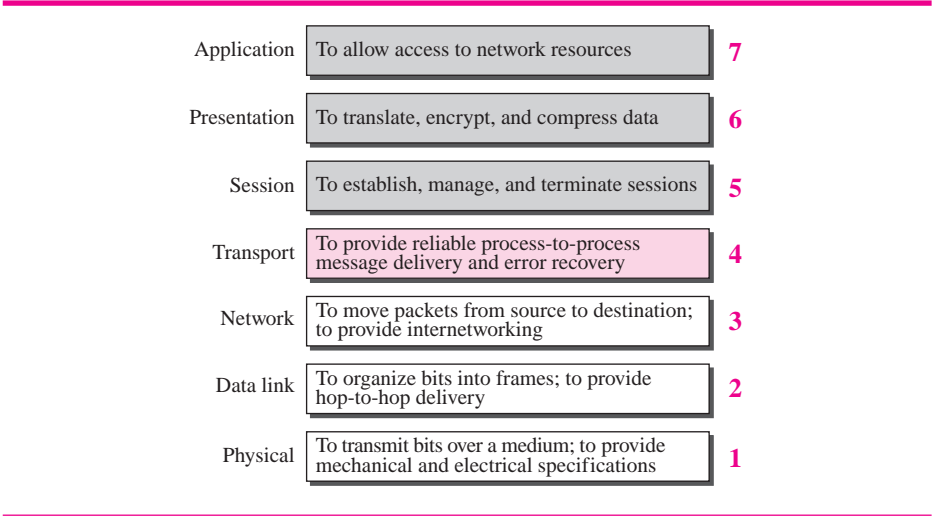


- ❑ **E-mail services.** This application provides the basis for e-mail forwarding and storage.
- ❑ **Directory services.** This application provides distributed database sources and access for global information about various objects and services.

Summary of OSI Layers

Figure 2.6 shows a summary of duties for each layer. In the next section, we describe how some of these duties are mixed and spread into five categories in the TCP/IP protocol suite.

Figure 2.6 Summary of OSI layers

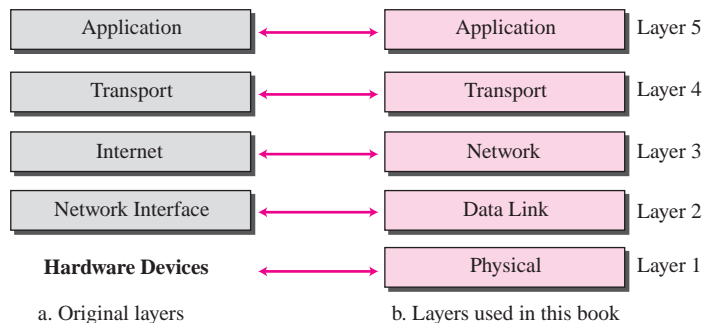
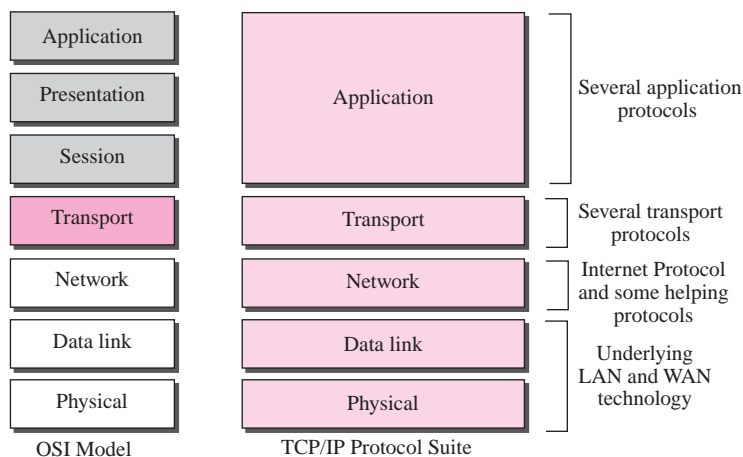


2.3 TCP/IP PROTOCOL SUITE

The **TCP/IP protocol suite** was developed prior to the OSI model. Therefore, the layers in the TCP/IP protocol suite do not match exactly with those in the OSI model. The original TCP/IP protocol suite was defined as four software layers built upon the hardware. Today, however, TCP/IP is thought of as a five-layer model with the layers named similarly to the ones in the OSI model. Figure 2.7 shows both configurations.

Comparison between OSI and TCP/IP Protocol Suite

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 2.8.

Figure 2.7 Layers in the TCP/IP Protocol Suite**Figure 2.8** TCP/IP and OSI model

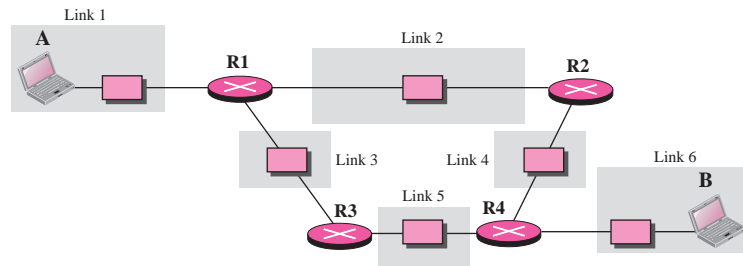
Two reasons were mentioned for this decision. First, TCP/IP has more than one transport-layer protocol. Some of the functionalities of the session layer are available in some of the transport layer protocols. Second, the application layer is not only one piece of software. Many applications can be developed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

TCP/IP is a hierarchical protocol made up of interactive modules, each of which provides a specific functionality, but the modules are not necessarily interdependent. Whereas the OSI model specifies which functions belong to each of its layers, the layers of the TCP/IP protocol suite contain relatively independent protocols that can be mixed and matched, depending on the needs of the system. The term *hierarchical* means that each upper level protocol is supported by one or more lower level protocols.

Layers in the TCP/IP Protocol Suite

In this section, we briefly discuss the purpose of each layer in the TCP/IP protocol suite. When we study the purpose of each layer, it is easier to think of a private *internet*, instead of the global Internet. We assume that we want to use the TCP/IP suite in a small, private internet. Such an internet is made up of several small networks, which we call links. A **link** is a network that allows a set of computers to communicate with each other. For example, if all computers in an office are wired together, the connection makes a link. If several computers belonging to a private company are connected via a satellite channel, the connection is a link. A link, as we discussed in Chapter 3, can be a LAN (local area network) serving a small area or a WAN (wide area network) serving a larger area. We also assume that different links are connected together by devices called *routers* or *switches* that route the data to reach their final destinations. Figure 2.9 shows our imaginary internet that is used to show the purpose of each layer. We have six links and four routers (R1 to R4). We have shown only two computers, A and B.

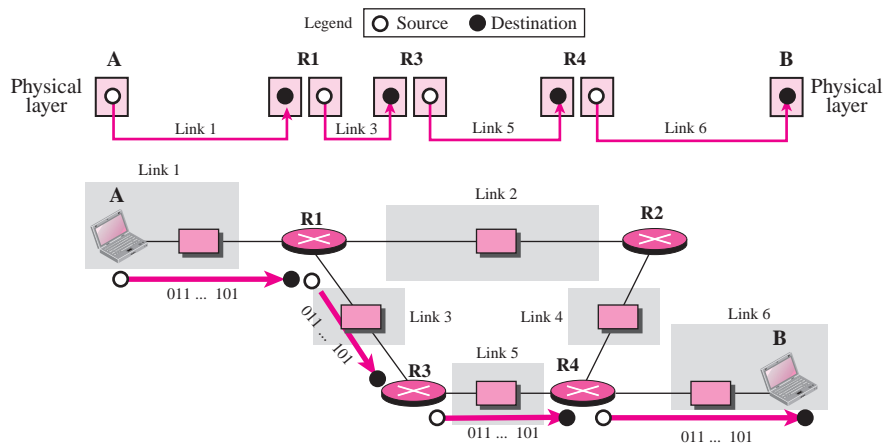
Figure 2.9 A private internet



Physical Layer

TCP/IP does not define any specific protocol for the physical layer. It supports all of the standard and proprietary protocols. At this level, the communication is between two hops or nodes, either a computer or router. The unit of communication is a single bit. When the connection is established between the two nodes, a stream of bits is flowing between them. The physical layer, however, treats each bit individually. Figure 2.10 shows the communication between nodes. We are assuming that at this moment the two computers have discovered that the most efficient way to communicate with each other is via routers R1, R3, and R4. How this decision is made is the subject of some future chapters.

Note that if a node is connected to n links, it needs n physical-layer protocols, one for each link. The reason is that different links may use different physical-layer protocols. The figure, however, shows only physical layers involved in the communication. Each computer involves with only one link; each router involves with only two links. As Figure 2.10 shows, the journey of bits between computer A and computer B is made of four independent short trips. Computer A sends each bit to router R1 in the format of the protocol used by link 1. Router 1 sends each bit to router R3 in the format dictated by the protocol used by link 3. And so on. Router R1 has two three physical layers (two are shown in our scenario). The layer connected to link 1 receives bits according to the format of the protocol

Figure 2.10 *Communication at the physical layer*

used by link 1; the layer connected to link 3 sends bits according to the format of the protocol used by link 3. It is the same situation with the other two routers involved in the communication.

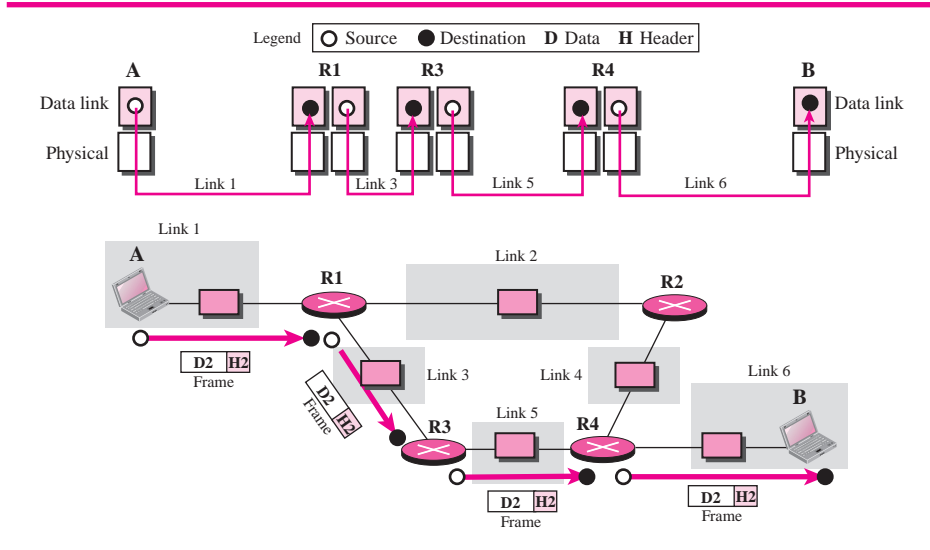
The unit of communication at the physical layer is a bit.

The responsibility of the physical layer, in addition to delivery of bits, matches with what mentioned for the physical layer of the OSI model, but it mostly depends on the underlying technologies that provide links. We see in the next chapter that they are, for example, many protocols for the physical layer of LANs or WANs.

Data Link Layer

TCP/IP does not define any specific protocol for the data link layer either. It supports all of the standard and proprietary protocols. At this level, the communication is also between two hops or nodes. The unit of communication however, is a packet called a **frame**. A frame is a packet that encapsulates the data received from the network layer with an added header and sometimes a trailer. The head, among other communication information, includes the source and destination of frame. The destination address is needed to define the right recipient of the frame because many nodes may have been connected to the link. The source address is needed for possible response or acknowledgment as may be required by some protocols. Figure 2.11 shows the communication at the data link layer.

Note that the frame that is travelling between computer A and router R1 may be different from the one travelling between router R1 and R3. When the frame is received by router R1, this router passes the frame to the data link layer protocol shown at the left. The frame is opened, the data are removed. The data are then passed to the data

Figure 2.11 *Communication at the data link layer*

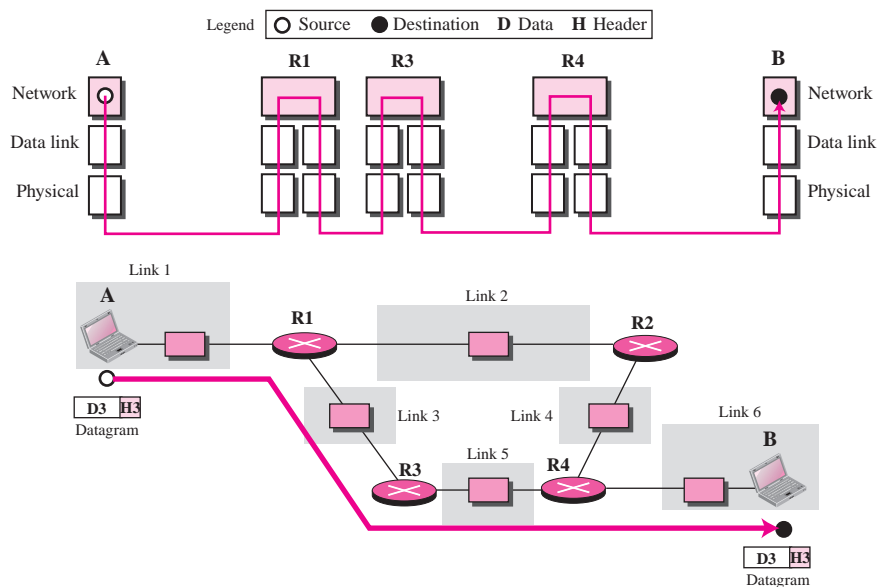
link layer protocol shown at the right to create a new frame to be sent to the router R3. The reason is that the two links, link 1 and link 3, may be using different protocols and require frames of different formats. Note also that the figure does not show the physical movement of frames; the physical movement happens only at the physical layer. The two nodes communicate logically at the data link layer, not physically. In other words, the data link layer at router R1 only *thinks* that a frame has been sent directly from the data link layer at computer A. What is sent from A to R1 is a stream of bits from one physical layer to another. Since a frame at A is transformed to a stream of bits, and the bits at R1 are transformed to a frame, it gives this impression to the two data link layer that a frame has been exchanged.

The unit of communication at the data link layer is a frame.

Network Layer

At the network layer (or, more accurately, the internetwork layer), TCP/IP supports the Internet Protocol (IP). The **Internet Protocol (IP)** is the transmission mechanism used by the TCP/IP protocols. IP transports data in packets called **datagrams**, each of which is transported separately. Datagrams can travel along different routes and can arrive out of sequence or be duplicated. IP does not keep track of the routes and has no facility for reordering datagrams once they arrive at their destination. Figure 2.12 shows the communication at the network layer.

Note that there is a main difference between the communication at the network layer and the communication at data link or physical layers. Communication at the

Figure 2.12 *Communication at the network layer*

network layer is end to end while the communication at the other two layers are node to node. The datagram started at computer A is the one that reaches computer B. The network layers of the routers can inspect the source and destination of the packet for finding the best route, but they are not allowed to change the contents of the packet. Of course, the communication is logical, not physical. Although the network layer of computer A and B *think* that they are sending and receiving datagrams, the actual communication again is done at the physical level.

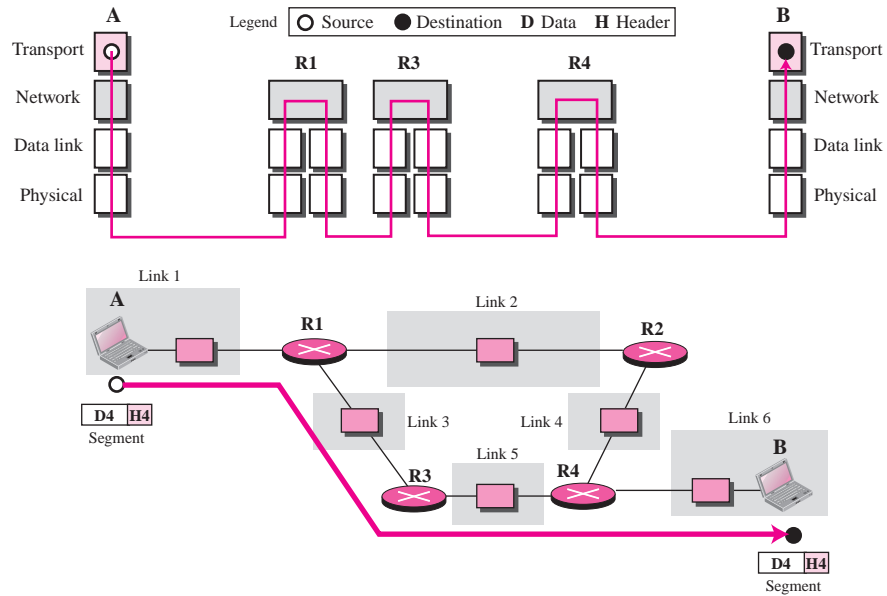
The unit of communication at the network layer is a datagram.

Transport Layer

There is a main difference between the transport layer and the network layer. Although all nodes in a network need to have the network layer, only the two end computers need to have the transport layer. The network layer is responsible for sending individual datagrams from computer A to computer B; the transport layer is responsible for delivering the whole message, which is called a segment, a user datagram, or a packet, from A to B. A segment may consist of a few or tens of datagrams. The segments need to be broken into datagrams and each datagram has to be delivered to the network layer for transmission. Since the Internet defines a different route for each datagram, the datagrams may arrive out of order and may be lost. The transport layer at computer B needs to wait until all of these datagrams to arrive, assemble

them and make a segment out of them. Figure 2.13 shows the communication at the transport layer.

Figure 2.13 *Communication at the transport layer*



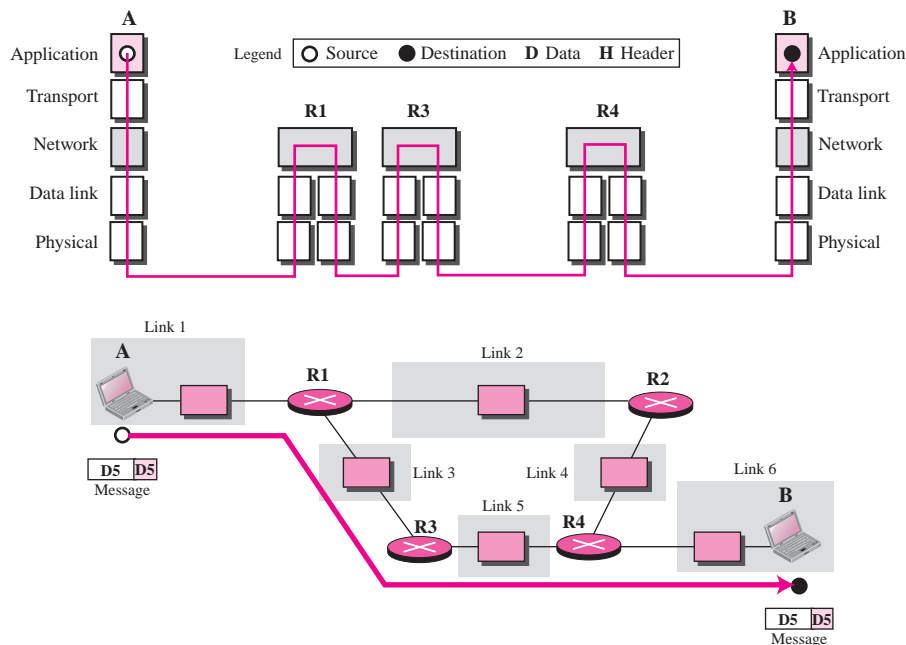
Again, we should know that the two transport layers only think that they are communicating with each other using a segment; the communication is done through the physical layer and the exchange of bits.

Traditionally, the transport layer was represented in the TCP/IP suite by two protocols: **User Datagram Protocol (UDP)** and **Transmission Control Protocol (TCP)**. A new protocol called **Stream Control Transmission Protocol (SCTP)** has been introduced in the last few years.

The unit of communication at the transport layer is a segment, user datagram, or a packet, depending on the specific protocol used in this layer.

Application Layer

The application layer in TCP/IP is equivalent to the combined session, presentation, and application layers in the OSI model. The application layer allows a user to access the services of our private internet or the global Internet. Many protocols are defined at this layer to provide services such as electronic mail, file transfer, accessing the World Wide Web, and so on. We cover most of the standard protocols in later chapters. Figure 2.14 shows the communication at the application layer.

Figure 2.14 *Communication at the application layer*

Note that the communication at the application layer, like the one at the transport layer, is end to end. A message generated at computer A is sent to computer B without being changed during the transmission.

The unit of communication at the application layer is a message.

2.4 ADDRESSING

Four levels of addresses are used in an internet employing the TCP/IP protocols: **physical address**, **logical address**, **port address**, and **application-specific address**. Each address is related to a one layer in the TCP/IP architecture, as shown in Figure 2.15.

Physical Addresses

The physical address, also known as the link address, is the address of a node as defined by its LAN or WAN. It is included in the frame used by the data link layer. It is the lowest-level address. The physical addresses have authority over the link (LAN or WAN). The size and format of these addresses vary depending on the network. For example, Ethernet uses a 6-byte (48-bit) physical address that is imprinted on the network interface card (NIC). LocalTalk (Apple), however, has a 1-byte dynamic address that changes each time the station comes up.