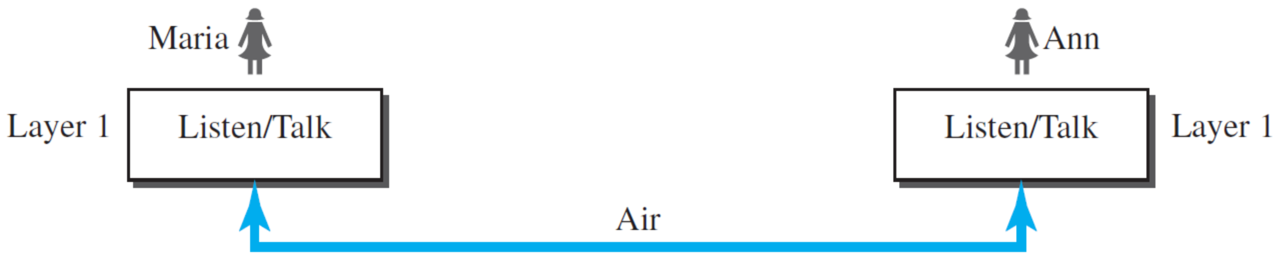
Chapter 2

Network Models

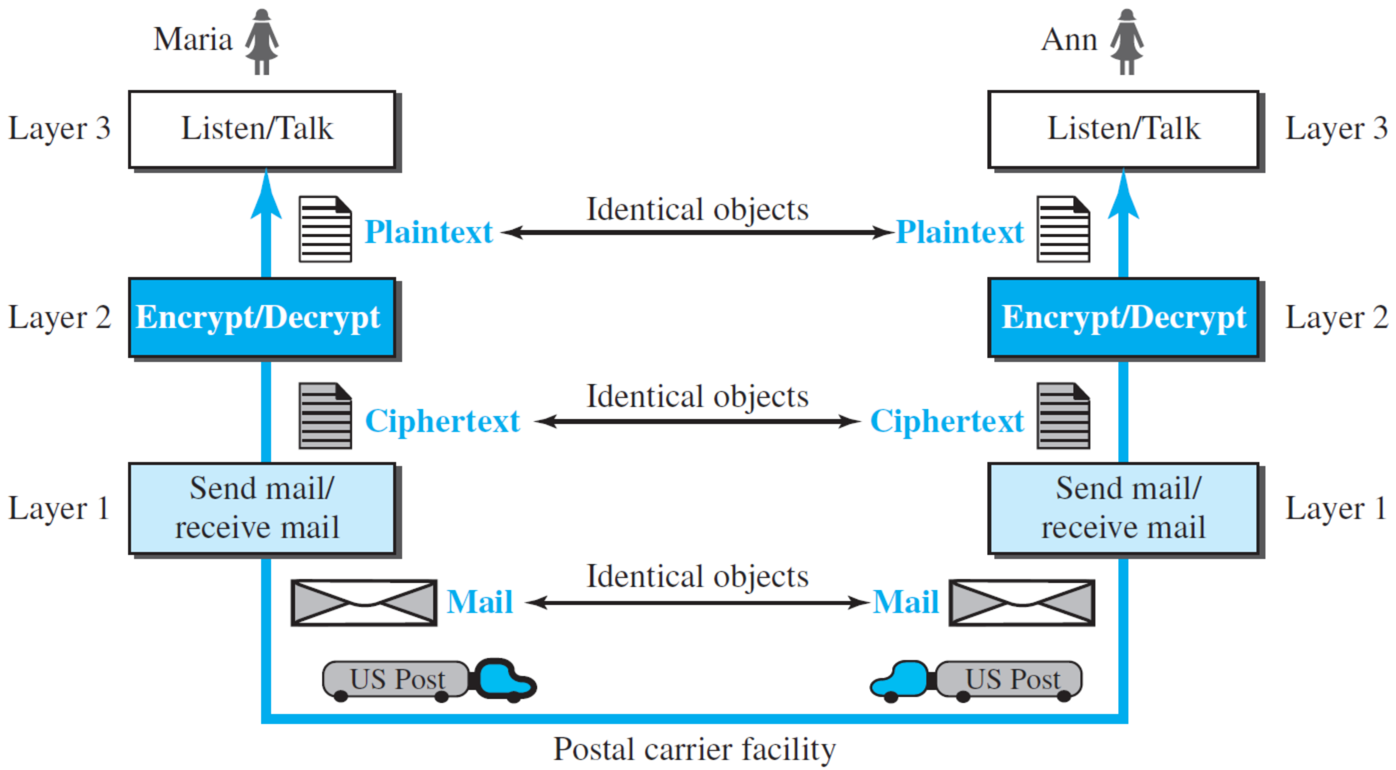
Protocol Layering

- 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**.

Sample Scenarios



A single-layer protocol



A three-layer protocol

Advantages of Protocol Layering

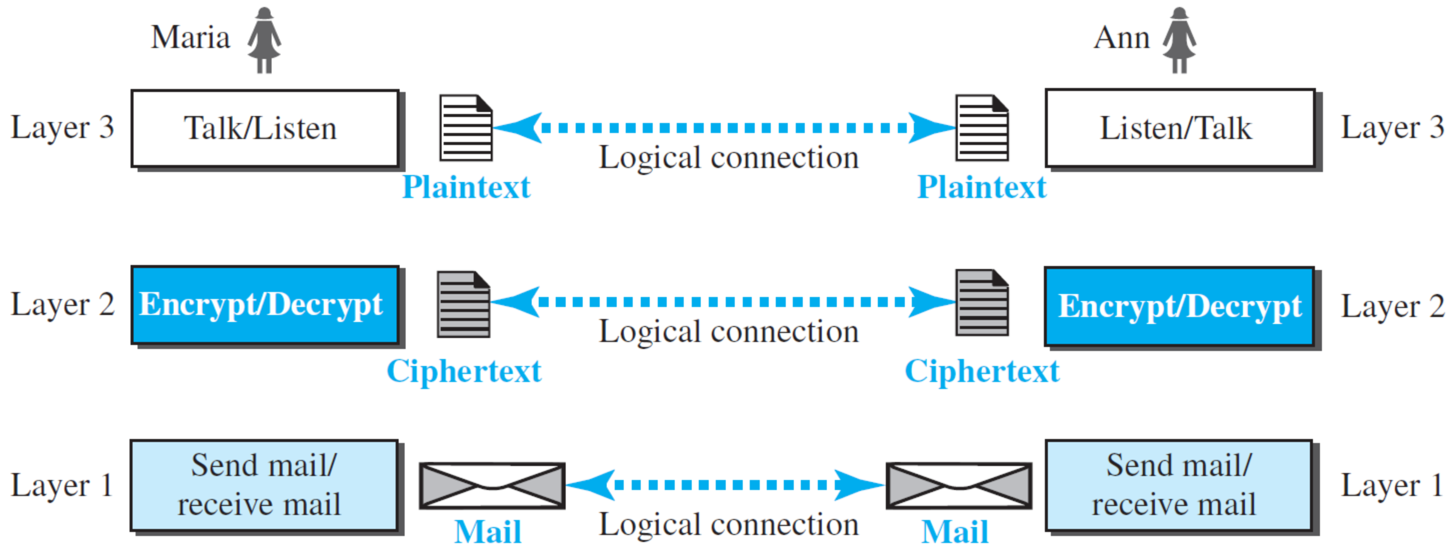
- Protocol layering enables us to divide a complex task into several smaller and simpler tasks.
- Protocol of one layer can be changed without touching the other layers
- Layering allows us to separate the services from the implementation. A layer needs to be able to receive a set of services from the lower layer and to give the services to the upper layer; we don’t care about how the layer is implemented.
- 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

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 ciphertext letter. The object under layer 1 at both sites should be a piece of mail.

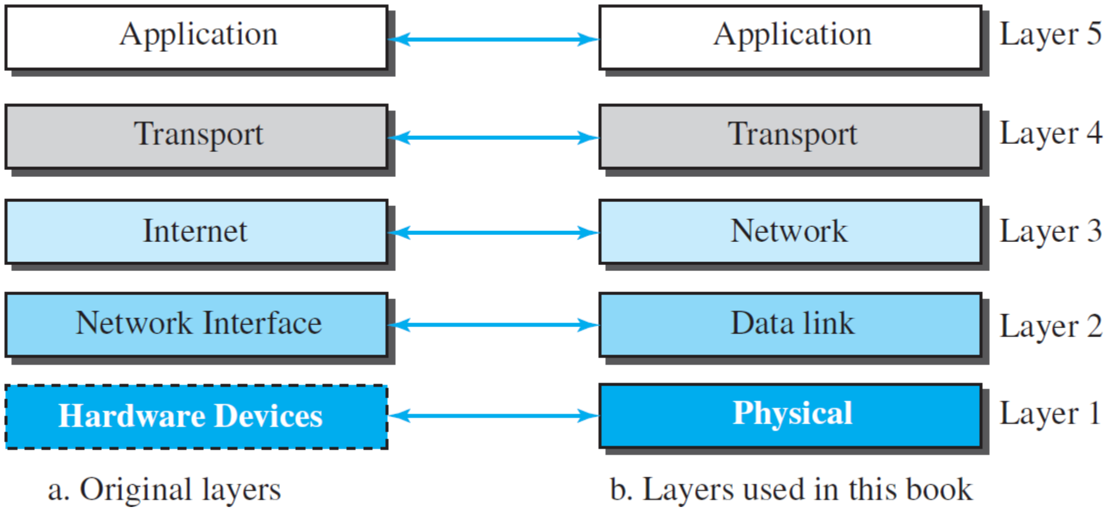
Logical Connections

- We can think about logical connection between each layer
- This means that we have layer-to-layer communication.
- Maria and Ann can think that there is a logical (imaginary) connection at 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 layering we encounter in data communication and networking.

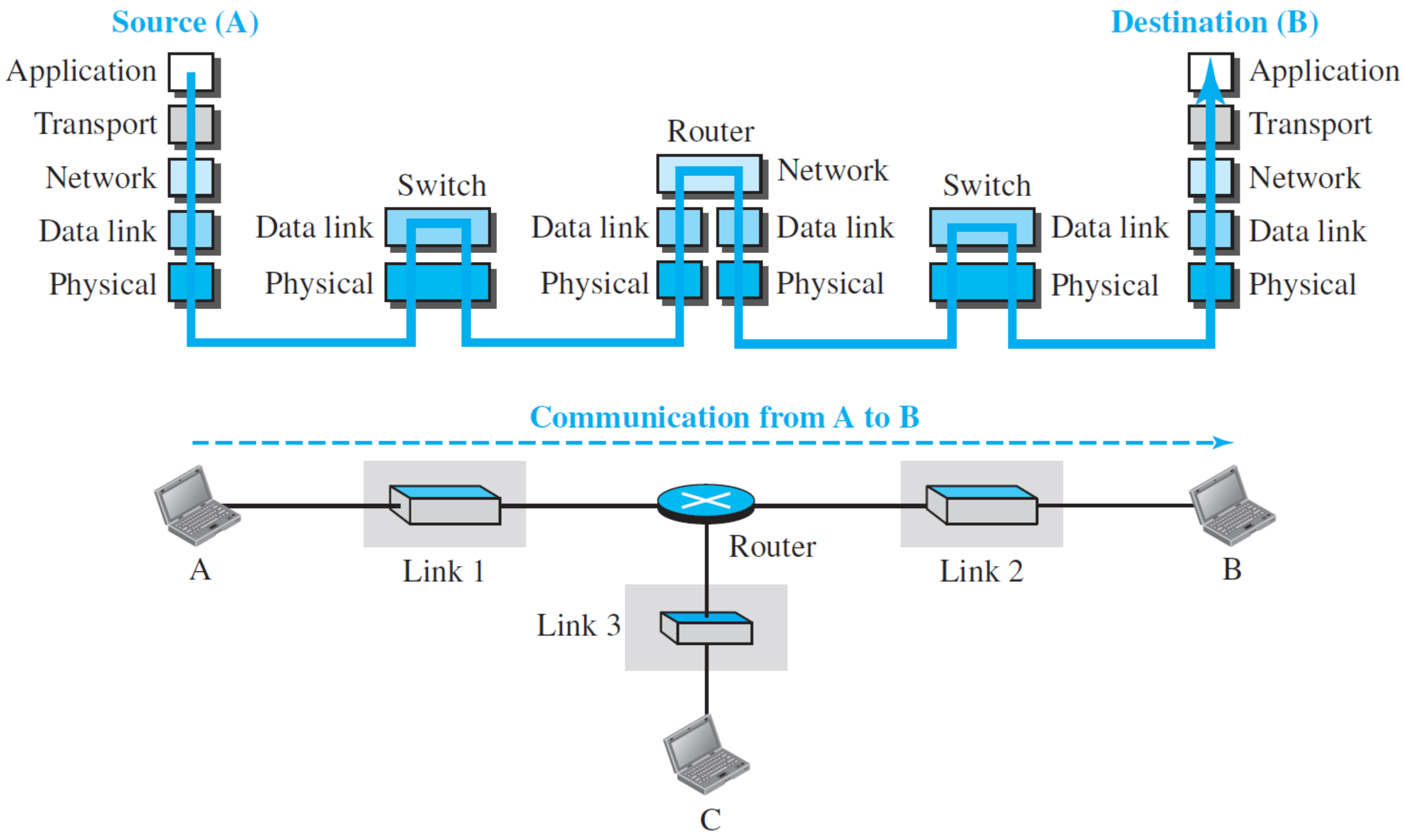


TCP/IP Protocol Suite

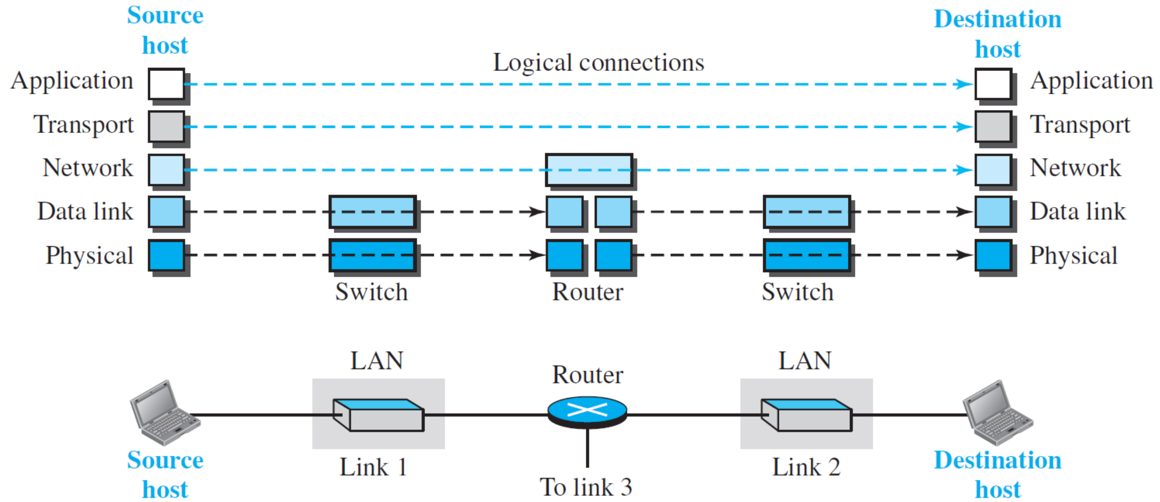
- **Protocol Suit** is a set of protocols organized in different layers
- **TCP/IP** (Transmission Control Protocol/Internet Protocol) is a protocol suite used in the Internet today.
- It is a hierarchical protocol made up of interactive modules, each of which provides a specific functionality.
- The 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 a five-layer model.



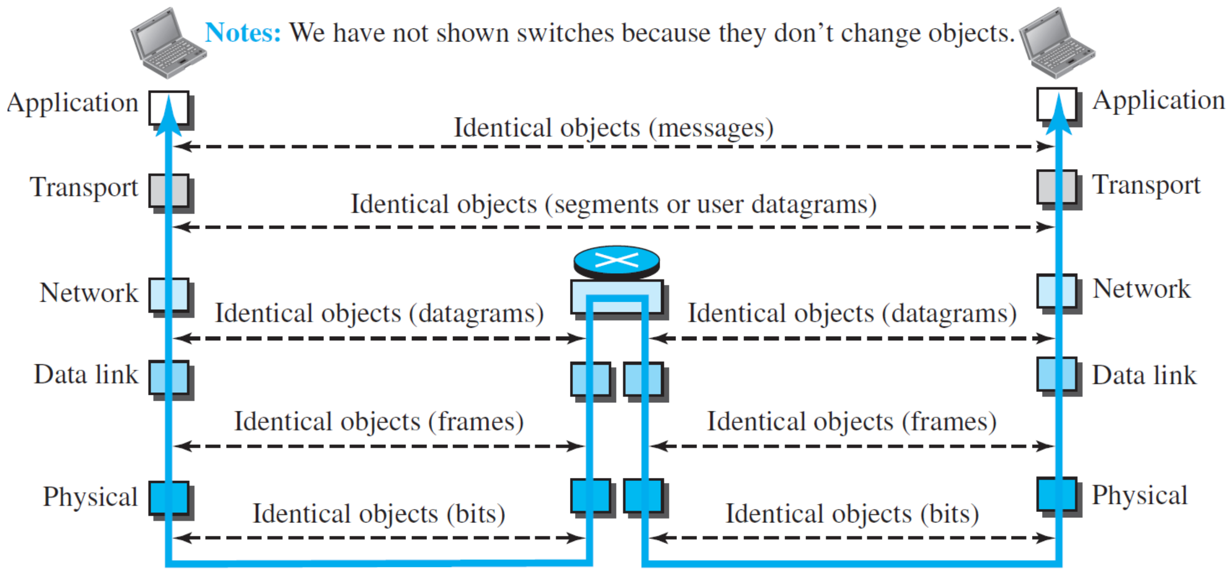
Layered Architecture



Layered Architecture



Logical connections between layers of the TCP/IP protocol suite



Identical objects in the TCP/IP protocol suite

Physical Layer

- **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 transformed 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.

Data-Link Layer

- 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 routers in the path are responsible for choosing the best route for each packet.
- Network layer is responsible for host-to-host communication and routing the packet through possible routes.
- 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 network 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.

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 have more than one protocol in the transport layer, which means that each application program can use the protocol that best matches its requirement
- The main protocol, Transmission Control Protocol (TCP), is a connection-oriented protocol that first establishes a logical connection between transport 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, error control, and congestion 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.
- UDP is a simple protocol that does not provide flow, error, or congestion control.

Application Layer

- 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.

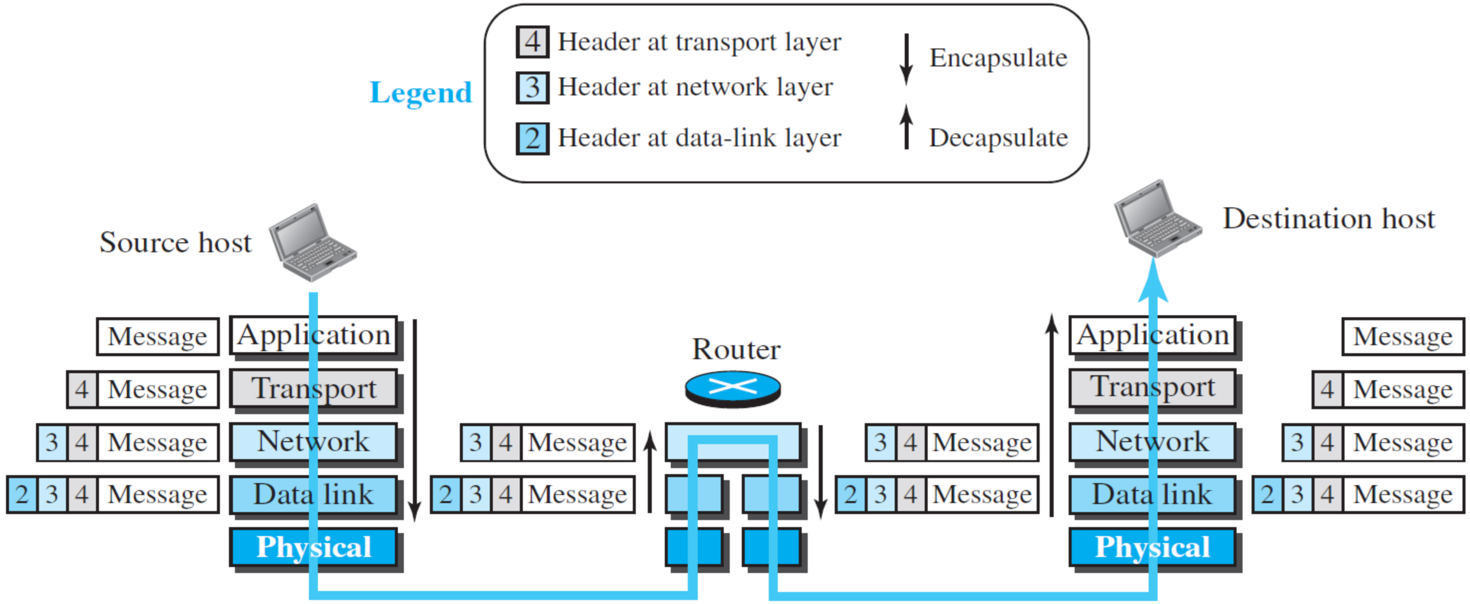
Encapsulation and Decapsulation

One of the important concepts in protocol layering in the Internet is encapsulation/decapsulation.

Encapsulation at the Source Host

Decapsulation and Encapsulation at the Router

Decapsulation at the Destination Host



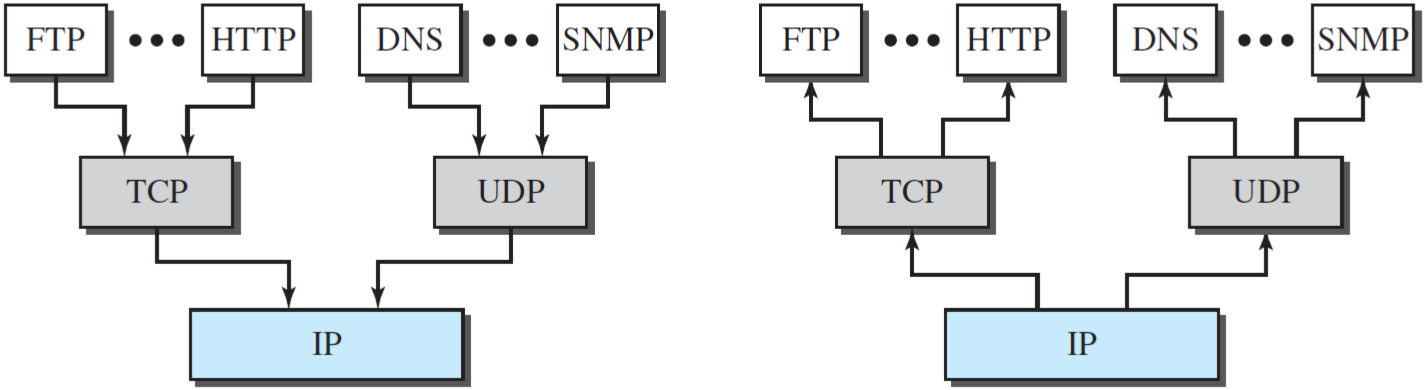
Addressing

- Any communication that involves two parties needs two addresses: source address and destination address.
- Although it looks as if we need five pairs of addresses, one pair per layer, we normally have only four because the physical layer does not need addresses; the unit of data exchange at the physical layer is a bit, which definitely cannot have an address.
- At the application layer, we normally use names such as *someorg.com*, or *somebody@cail.com*.
- At the transport layer, addresses are called port numbers, and these define the application-layer programs at the source and destination.
- At the network-layer, IP address - the addresses are global, with the whole Internet as the scope.
- The link-layer addresses, sometimes called MAC addresses, are locally defined addresses, each of which defines a specific host or router in a network (LAN or WAN).

Packet names	Layers	Addresses
Message	Application layer	Names
Segment / User datagram	Transport layer	Port numbers
Datagram	Network layer	Logical addresses
Frame	Data-link layer	Link-layer addresses
Bits	**Physical layer**	

Multiplexing and Demultiplexing

- Since the TCP/IP protocol suite uses several protocols at some layers, we can say that we have multiplexing at the source and demultiplexing at the destination.
- Multiplexing in this case means that a protocol at a layer can encapsulate a packet from several next-higher layer protocols (one at a time)
- Demultiplexing means that a protocol can decapsulate and deliver a packet to several next-higher layer protocols (one at a time).
- To be able to multiplex and demultiplex, a protocol needs to have a field in its header to identify to which protocol the encapsulated packets belong.
- At the transport layer, either UDP or TCP can accept a message from several application-layer protocols. At the network layer, IP can accept a segment from TCP or a user datagram from UDP. IP can also accept a packet from other protocols such as ICMP, IGMP, and so on.
- At the data-link layer, a frame may carry the payload coming from IP or other protocols such as ARP



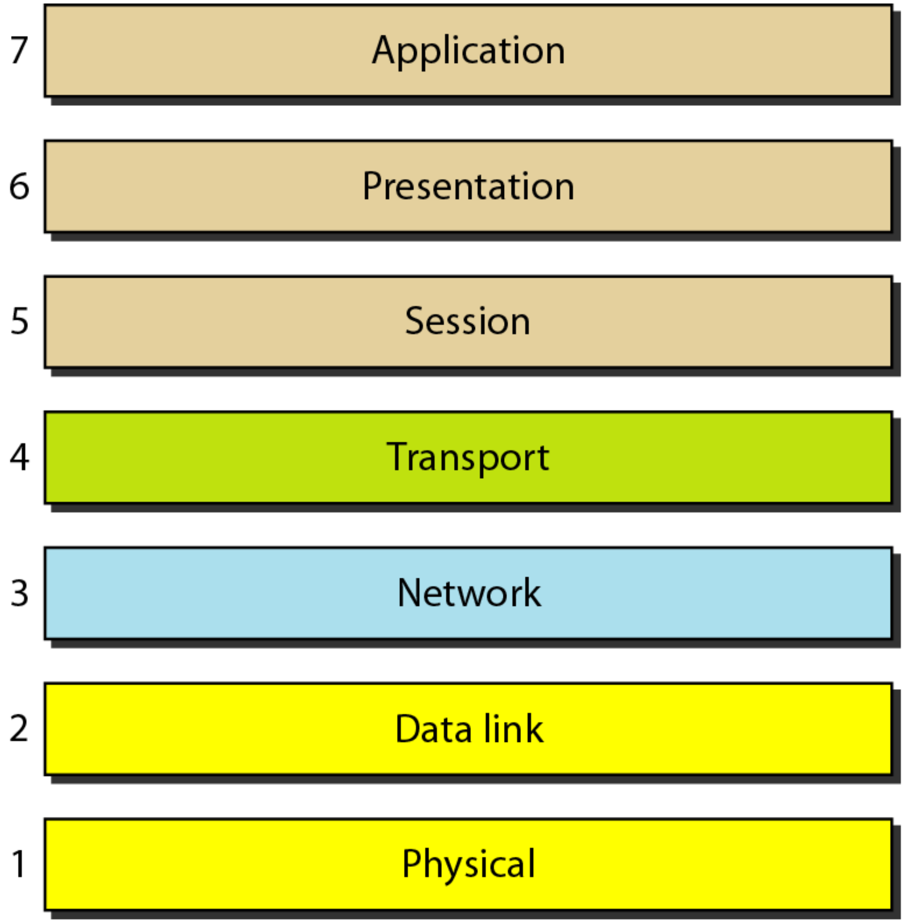
a. Multiplexing at source

b. Demultiplexing at destination

The OSI Model

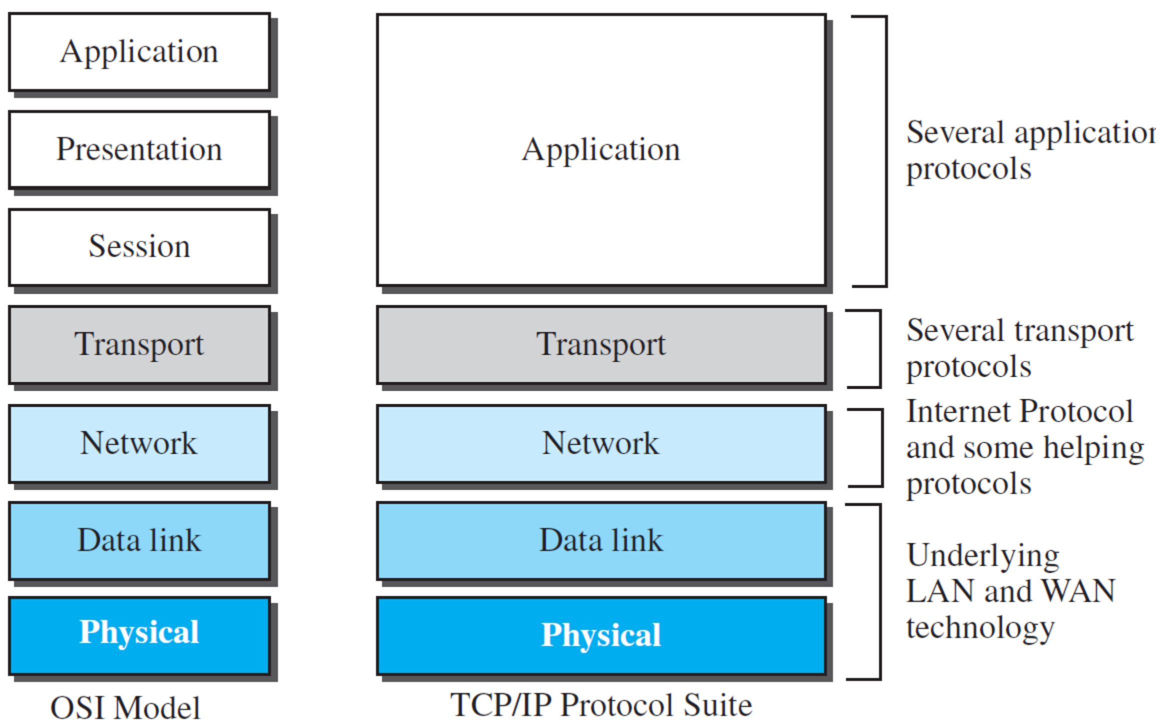
- When speaking of the Internet, everyone talks about the TCP/IP protocol suite, this suite is not the only suite of protocols defined.
- Established in 1947, the International Standards Organization (**ISO**) is a multinational body dedicated to worldwide agreement on international standards. An ISO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

Seven Layers of 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
- Two reasons for not adding session and presentation layers
- 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

- 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.
- 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.
- 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