UNIT-01

→ **Data Communication:** Data communications are the exchange of data between two devices via some form of transmission medium such as a wire cable.
For data communication to occur, the communicating devices must be part of a communication system made up of a combination of hardware and software.

* The effectiveness of data communications system depends on four fundamental characteristics:

① **Delivery:** System must deliver data to the correct destination.

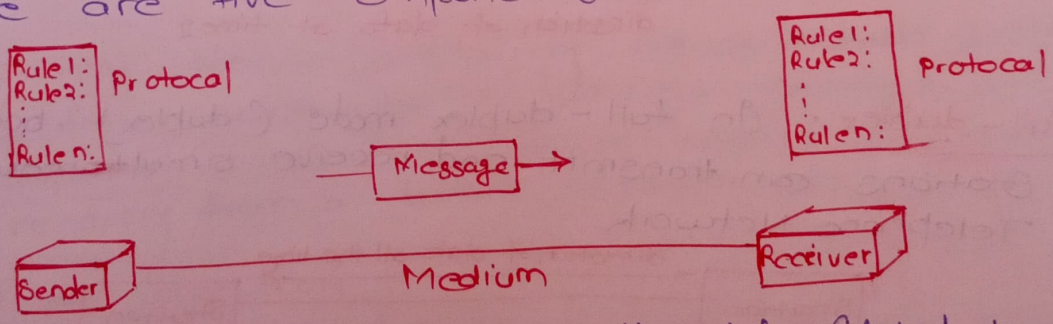
② **Accuracy:** The system must deliver data accurately (No altered or left uncorrected data in transmission)

③ **Timeliness:** The system must deliver data in timely manner. (eg) vedios audios must be delivered as they are produced, in the same order.

④ **Jitter:** Jitter refers to the variation in the packet amival time.
(eg) delay in sending vedio/audio packets.

→ **Component of Data Communication system:**

There are five components of data Communication:



① **Message:** The message is the info. (data) to be communicated. (eg) → text, numbers, pictures, audio, vedio etc.

② **Sender:** The sender is the device that sends data messages. (eg) Computer, workstation, telephone, handset, television etc

③ **Receiver:** The receive is the device that receives the messages.

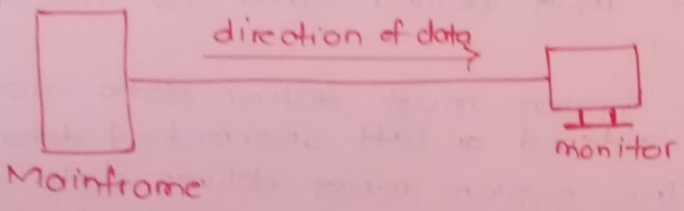
④ **Transmission medium:** The transmission medium is the physical path by which a message travels from sender to receiver. (eg) → twisted-pair wire, coaxial cable, fibre optic cable, radio-waves etc

(5) Protocal: a protocal is a set of rules that govern data communication. It represents an agreement between communicating devices.

→ Data Flow: Communication between two device can be simplex, half-duplex or full duplex.

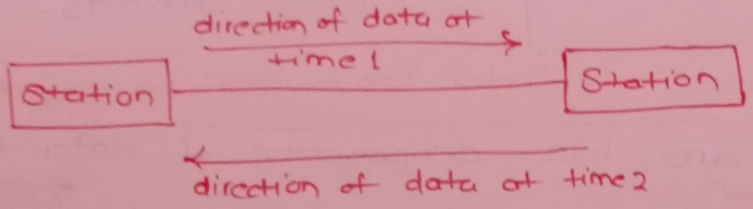
(1) Simplex: In simplex mode, the communication is unidirectional, as on a one-way street. Only one of the two devices on a link can transmit, the other can only receive.

eg) keyboards, traditional monitors.



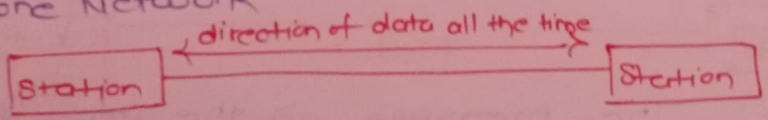
(2) Half-duplex: In half duplex mode, each station can both transmit but not at the same time. When one device is sending, the other can only receive and vice versa.

eg) Walkie-talkies, CB radios



(3) Full-duplex: In full-duplex mode (duplex), both stations can transmit and receive simultaneously.

eg) Telephone Network



→ Networks: A network is a set of devices (nodes) connected by communication links. any device capable of sending and/or receiving data generated by other nodes is a network

eg) Computer, printer etc

→ Distributed processing: tasks are devided among multiple computers

→ **Network Criteria**: A Network must be able to meet a certain number of criteria. These criteria are—

① **Performance**: Performance can be measured in many ways, including transit time and response time.
Transit Time: 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.

Performance depends on
- number of users
- type of transmission medium
- Capabilities of connected hardware
- efficiency of the software

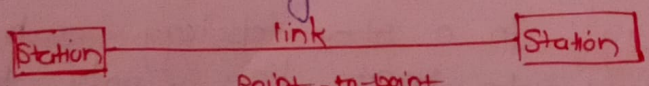
is evaluated by two networking metrics
- throughput ↑
- delay ↓

> Often Contradictory

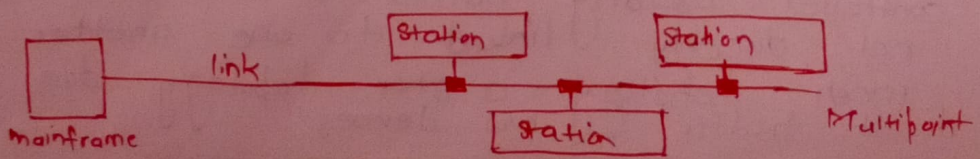
② **Reliability**: In addition to accuracy of delivery, network reliability is measured by the frequency of failure, the time it takes to recover from failure, the network's robustness in a catastrophe.

③ **Security**: Network security issues include protecting data from unauthorized access, damage and development and implementing policies and procedures for recovery from a breach or data loss.

→ **Type of Connection**
- **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.
- **Multipoint**: is one in which more than two specific devices share a single link.



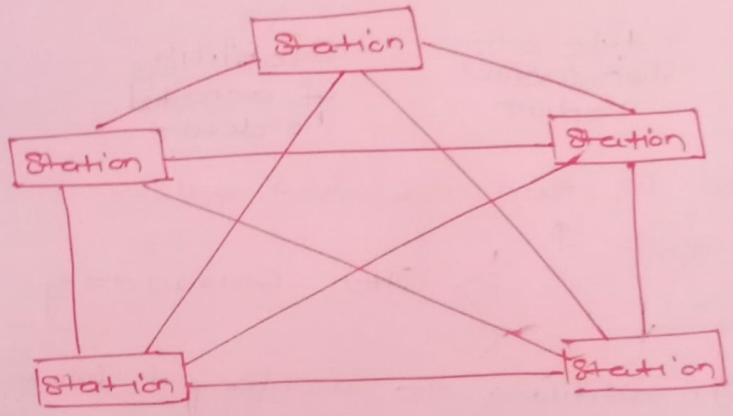
Point-to-point



Multipoint

→ **Physical Topology**: It refers to the way in which a network is laid out physically. Two or more devices connect to a link; two or more links form a topology.
(geometric representation of the relationship of all the links and linking devices)

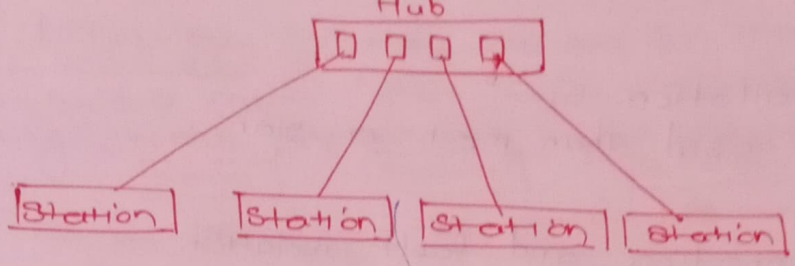
① **Mesh Topology**: In mesh topology, every device has a dedicated point to point link to every other devices. The term dedicated means that the link carries traffic only between the two devices it connects.
In a mesh topology, we need $n(n-1)/2$ duplex-mode links.



Advantages: ① Each connection carry its own data load, thus no traffic problems.
② It is robust - if one link becomes unusable, it does not incapacitate the entire system.
③ It is secure because every message travels a dedicated line.

Disadvantages: ① The main disadvantages of mesh are related to the cabling and the number of I/O ports required.
② Too much wiring
③ Expensive
eg telephone regional

② **Star Topology**: In a star topology, each device has a dedicated point-to-point link only to a central controller, usually called a hub. The devices are not directly linked to one another. Unlike a mesh topology, a star topology does not allow direct traffic b/w devices.



Advantages:

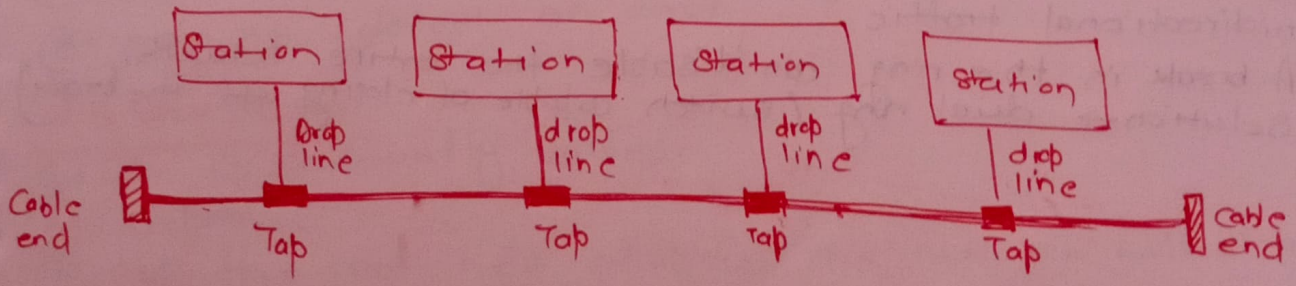
1. less expensive than a mesh topology.
2. easy to install and reconfigure (each device needs only one link and one I/O (port))
3. No direct traffic b/w devices.
4. less cabling
5. robustness
6. easy fault identification and fault isolation.

Disadvantages:

1. dependency of whole system on a single point → the hub. If hub goes down, the whole system is dead.
2. more cable than other topologies except mesh.

3. **Bus Topology**: A bus topology is multipoint. One long cable acts as a backbone to link all the devices in a network. Nodes are connected to the bus cable by drop lines and taps. A drop line is a connection running b/w the device and the main cable. A tap is a connector that create a contact with the metallic core of the cable.

As a signal travels along the backbone, some of its energy is transformed into heat. Therefore, it becomes weaker and weaker as it travels farther and farther. For this reason there is a limit on the no. of taps a bus can support and on the distance b/w taps.



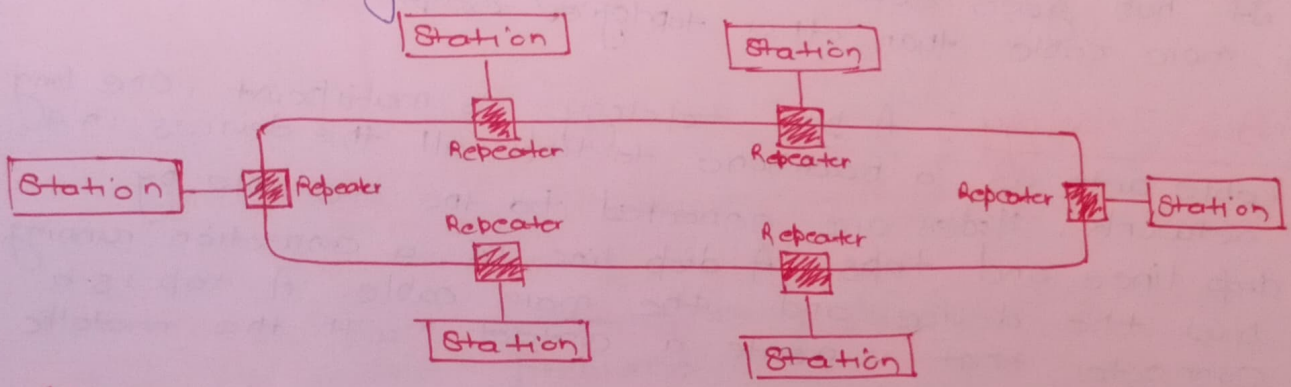
Advantages:

1. ease of installation.
2. less cabling used than mesh or star

disadvantages:

1. difficult reconnection and fault isolation.
2. difficulty in the installment of new devices.

4. **Ring Topology**: In ring topology each device has a dedicated point-to-point connection with only the two devices on either side of it. A signal is passed along the ring in one direction, from device to device, until it reaches its destination. Each device in the ring incorporates a repeater. When a device receives a signal intended for another device, its repeater regenerates the bits and passes them along.



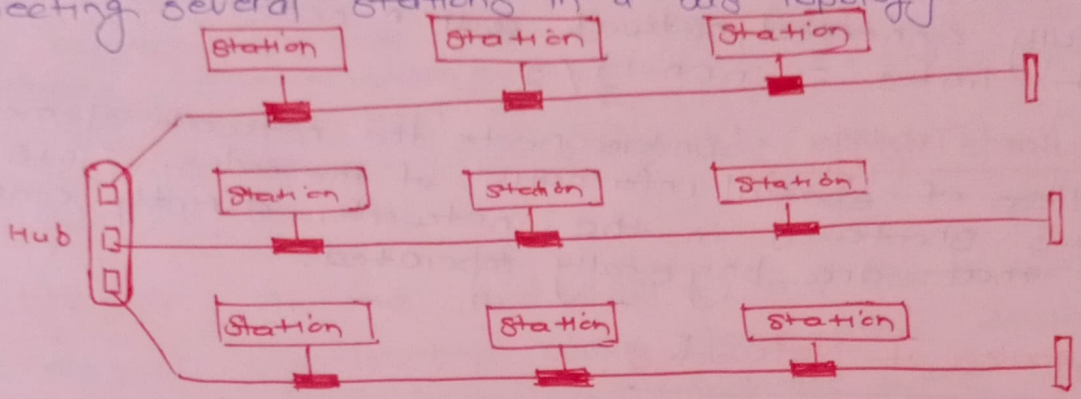
Advantages:

1. easy to install and reconfigure.
2. fault isolation is simplified.
3. If one device does not receive a signal within a specified period, it can issue an alarm. The alarm alerts the network operator to the problem and its location.

Disadvantages:

1. Unidirectional traffic
2. A break in the ring can disable the entire network. (Solution → dual ring / switch capable of closing off the break)

(5) **Hybrid Topology**: A network can be hybrid. for example, we can have a main star topology with each branch connecting several stations in a bus topology.

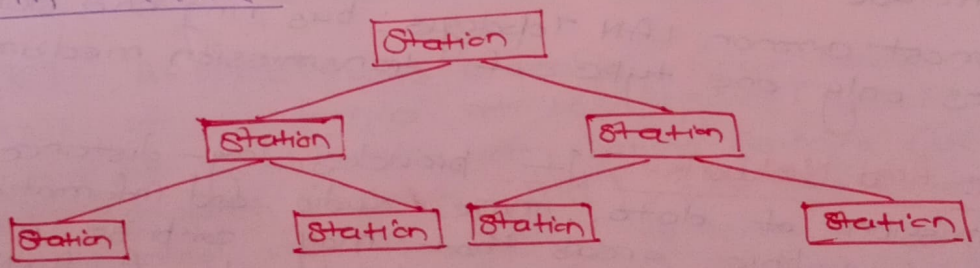


Advantages: (1) easy to troubleshoot and provides simple-error-detecting techniques.
(2) flexible network topology
(3) scalable

Disadvantages: (1) hard to design
(2) costly, as it involves more than one topology.

(6) **Tree Topology**: is a topology in which the nodes are connected heirarchically, with all the nodes connected to the topmost node or root node, also known as heirarchical topology. It has atleast three levels of heirarchy.

applied in — WAN



Advantages:
(1) easy to expand the network with more nodes.
(2) easy to maintain and manage
(3) easy to detect an error

disadvantages:
(1) It is profoundly cabled.
(2) expensive
(3) if the root node collapses, the network will also collapse.

* **Complete Topology**: A complete topology is a topology in which there is a direct link between all pairs of nodes.

On a fully connected network with n nodes
direct links = $n(n-1)/2$

* **Irregular Topology**: Interconnects the network elements regardless of spatial information of the nodes. This allows shortcuts in the network, directly connecting nodes that are physically seperated.

→ **Categories of Networks**:

LAN — less than 2 miles
WAN — worldwide
MAN — 10 miles
] generally

① **Local Area Network**: LAN is usually privately owned and links the devices in a single office, building, or campus. Depending on the needs of an organization and the type of technology used, 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 including audio and vedio peripherals.
currently LAN size is limited to a few kms
LAN's are designed for resources to be shared can include hardware, s/w or data.
The most common LAN Topologies: bus, ring and star
* uses only one type of transmission medium.

② **Wide Area Network**: It provides long-distance transmission of data, image, audio and information over large geographic areas that may comprise a country, a continent or even the whole world. A WAN can be as complex as the backbones that connect the internet or as simple as a dial up line that connects a home computer to the internet.

→ **Switched WAN**: Connects the end systems. (eg) router, ATM network

→ **Point-to-point**: Normally a line leased from a telephone or cable tv provider that connects a home computer or a small LAN to a internet provider.

(B) Metropolitan Area Network: A 'man' is a network with a size b/w a LAN and a WAN. It normally covers the area inside a town or a city. It is designed for customers who need a high-speed connectivity normally to the internet and have endpoints spread over a city or part of city.

(eg) Telephone Network that can provide a high speed DSL line, Cable TV network

→ ARPANET: In the mid-1960's, mainframe computers in research organizations were stand-alone devices. Computers from different manufacturers were unable to communicate with one another. The Advanced Research Projects Agency (ARPA) in the department of Defense (DOD) was interested in finding a way to connect computers so that the researchers they funded could share their findings, thereby reducing costs and eliminating duplication of effort.

In 1967, at an association for Computing Machinary (ACM) meeting, ARPA presented its ideas for ARPANET, a small network of connected computers. The idea was that each host computer (not necessarily from the same manufacturer) would be attached to a specialized computer, called an interface message processor (IMP). The IMPs, in turn, would be connected to one another. Each IMP had to be able to communicate with other IMPs as well as with its own attached host.

By 1969, ARPANET was a reality. Four nodes, at the University of California at Los Angeles (UCLA), the University of California at Santa Barbara (UCSB), Stanford Research Institute (SRI), and the University of Utah were connected via the IMPs to form a network. Software called Network Control Protocol (NCP) provided communication between the hosts.

* In 1972, Vint Cerf and Bob Kahn, both were part of core ARPANET group, collaborated on what they called the internetting project. Their paper on transmission control protocol (TCP) included concepts such as encapsulation, the datagram, and the functions of a gateway.

Shortly after, authorities made a decision to split TCP into two protocols: Transmission Control protocol (TCP) and ~~interconnecting~~ interworking Protocol (IP).

TCP: handles higher-level functions. (eg) segmentation, reassembly, error detection.

IP: handles datagram routing

The internetworking protocol became known as TCP/IP

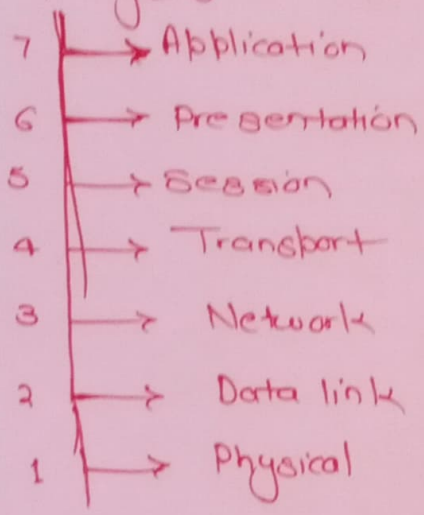
→ Internet: The internet is a structured, organized system. A network is a group of connected communicating devices such as computers and printers. An internet is two or more networks that can communicate with each other. The most notable internet is called the Internet, a collaboration of more than hundreds of thousands of interconnected networks. Private individuals as well as various organizations such as gat. agencies, schools, research facilities, corporations and libraries in more than 100 countries use the Internet. Yet this extraordinary communication system only came into existence in 1969.
Today most end users who want Internet connection use the services of Internet Serice ~~Protocol~~ providers (ISPs)

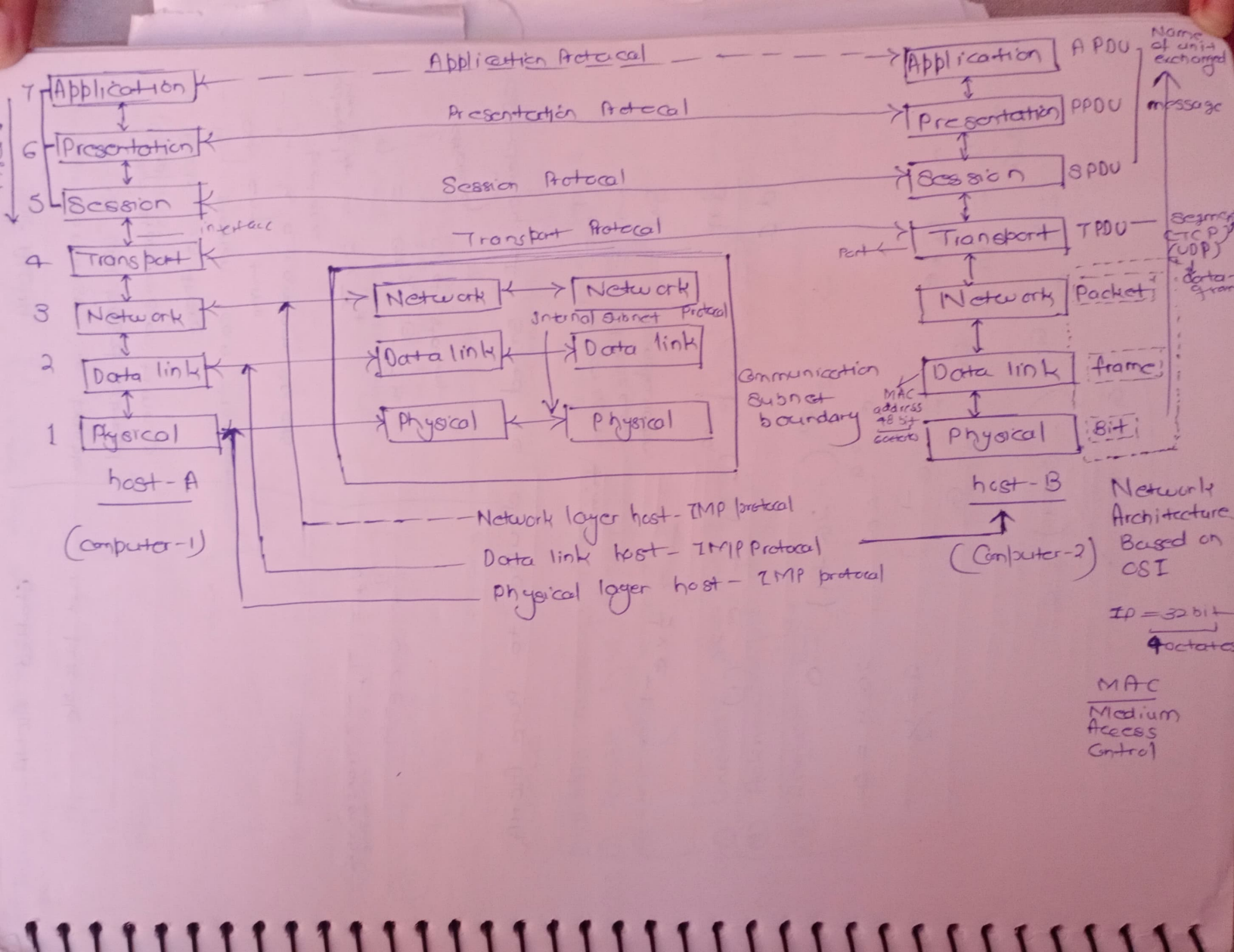
→ Protocals: On computer Networks, communication occurs between entities in different systems. An entity is anything capable of sending or receiving information. A protocal is a set of rules that gover data communications. A protocal defines what is communicated, how it is communicated, and when it is communicated.
Key elements
- → Syntax: The structure or format of the data, meaning the order in which they are presented.
- → Semantix: the meaning of each section of bits.
- → Timing: when data should be sent and how fast they can be sent.

→ Standards: Standards provide guidelines to manufacturers, venders, government agencies and other services providers to ensure the kind of intereconnectivity necessary in today's marketplace and in international communications.
- → De facto: Standards that have not been approved by any organized body but have been adopted as standards through widespread use. (By fact)
- → De Jure: (By Law) have been legislated by an officially recognized body.

→ **The OSI Model**: The OSI model is a layered framework for the design of network systems that allows communication b/w all types of computer systems. It was first introduced in the late 1970s by the International Standards Organization (ISO). It consists of seven seperate but related layers, each of which defines a part of the process of moving information across a network.

7 layers of the Open Systems Interconnection Model:




7 Application
6 Presentation
5 Session
interface
4 Transport
3 Network
2 Data link
1 Physical
host-A
(computer-1)
Application Protocal
Presentation Protocal
Session Protocal
Transport Protocal
Network
Network
Internal Subnet Protocal
Data link
Data link
Physical
Physical
Communication Subnet boundary
Network layer host-IMP protocal
Data link host-IMP Protocal
Physical layer host-IMP protocal
Application
Presentation
Session
Transport
Network
Data link
Physical
host-B
(Computer-2)
A PDU
PPDU
SPDU
TPDU
Port
Packet
frame
Bit
MAC address 48 bit
Name of unit exchanged
message
Segment (TCP) (UDP)
data gram
Network Architecture Based on OSI
IP = 32 bit
4 octate
MAC
Medium Access Control

Encapsulation

Figure 2.4 reveals another aspect of data communications in the OSI model: encapsulation. A packet (header and data) at level 7 is encapsulated in a packet at level 6. The whole packet at level 6 is encapsulated in a packet at level 5, and so on.

In other words, the data portion of a packet at level $N - 1$ carries the whole packet (data and header and maybe trailer) from level N. The concept is called *encapsulation;* level $N - 1$ is not aware of which part of the encapsulated packet is data and which part is the header or trailer. For level $N - 1$, the whole packet coming from level N is treated as one integral unit.

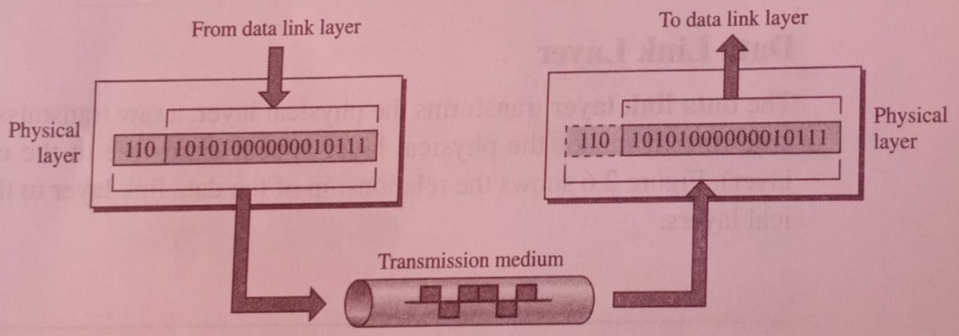
2.3 LAYERS IN THE OSI MODEL

In this section we briefly describe the functions of each layer in the OSI model.

Physical Layer

The **physical layer** coordinates the functions required to carry a bit stream over a physical medium. It deals with the mechanical and electrical specifications of the interface and transmission medium. It also defines the procedures and functions that physical devices and interfaces have to perform for transmission to occur. Figure 2.5 shows the position of the physical layer with respect to the transmission medium and the data link layer.

Figure 2.5 *Physical layer*



The physical layer is responsible for movements of individual bits from one hop (node) to the next.

The physical layer is also concerned with the following:

- **Physical characteristics of interfaces and medium.** The physical layer defines the characteristics of the interface between the devices and the transmission medium. It also defines the type of transmission medium.
- **Representation of bits.** The physical layer data consists of a stream of **bits** (sequence of 0s or 1s) with no interpretation. To be transmitted, bits must be

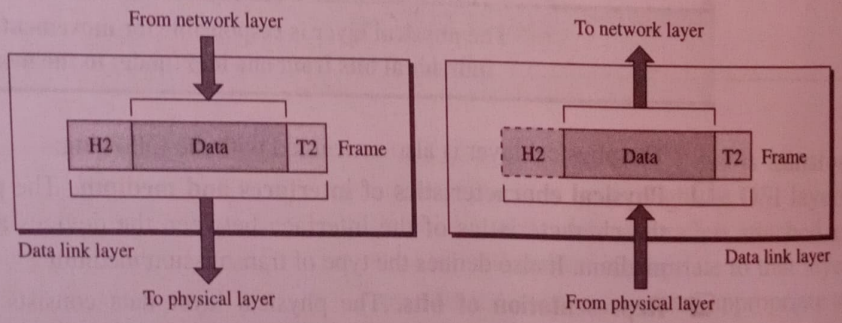
encoded into signals—electrical or optical. The physical layer defines the type of **encoding** (how 0s and 1s are changed to signals).

- **Data rate.** The **transmission rate**—the number of bits sent each second—is also defined by the physical layer. In other words, the physical layer defines the duration of a bit, which is how long it lasts.
- **Synchronization of bits.** The sender and receiver not only must use the same bit rate but also must be synchronized at the bit level. In other words, the sender and the receiver clocks must be synchronized.
- **Line configuration.** The physical layer is concerned with the connection of devices to the media. In a point-to-point configuration, two devices are connected through a dedicated link. In a multipoint configuration, a link is shared among several devices.
- **Physical topology.** The physical topology defines how devices are connected to make a network. Devices can be connected by using a mesh topology (every device is connected to every other device), a star topology (devices are connected through a central device), a ring topology (each device is connected to the next, forming a ring), a bus topology (every device is on a common link), or a hybrid topology (this is a combination of two or more topologies).
- **Transmission mode.** The physical layer also defines the direction of transmission between two devices: simplex, half-duplex, or full-duplex. In simplex mode, only one device can send; the other can only receive. The simplex mode is a one-way communication. In the half-duplex mode, two devices can send and receive, but not at the same time. In a full-duplex (or simply duplex) mode, two devices can send and receive at the same time.

Data Link Layer

The **data link layer** transforms the physical layer, a raw transmission facility, to a reliable link. It makes the physical layer appear error-free to the upper layer (network layer). Figure 2.6 shows the relationship of the data link layer to the network and physical layers.

Figure 2.6 *Data link layer*



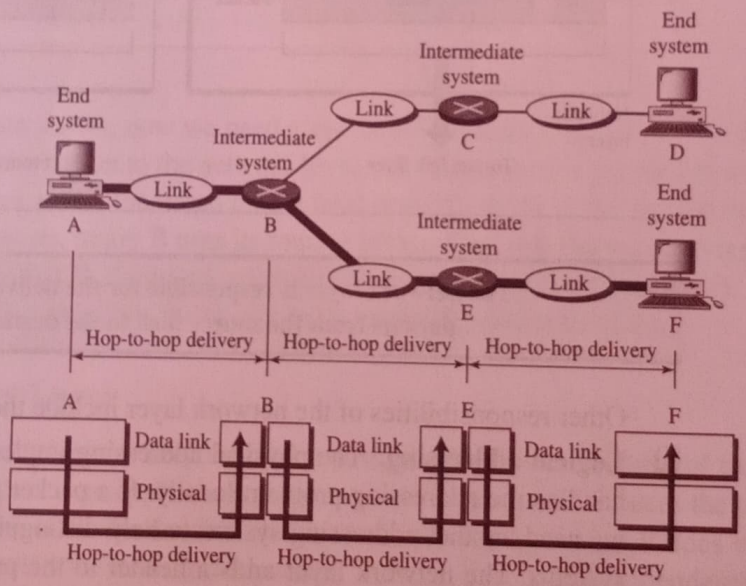
The data link layer is responsible for moving frames from one hop (node) to the next.

Other responsibilities of the data link layer include the following:

- **Framing.** The data link layer divides the stream of bits received from the network layer into manageable data units called **frames.**
- **Physical addressing.** If frames are to be distributed to different systems on the network, the data link layer adds a header to the frame to define the sender and/or receiver of the frame. If the frame is intended for a system outside the sender's network, the receiver address is the address of the device that connects the network to the next one.
- **Flow control.** If the rate at which the data are absorbed by the receiver is less than the rate at which data are produced in the sender, the data link layer imposes a flow control mechanism to avoid overwhelming the receiver.
- **Error control.** The data link layer adds reliability to the physical layer by adding mechanisms to detect and retransmit damaged or lost frames. It also uses a mechanism to recognize duplicate frames. Error control is normally achieved through a trailer added to the end of the frame.
- **Access control.** When two or more devices are connected to the same link, data link layer protocols are necessary to determine which device has control over the link at any given time.

Figure 2.7 illustrates **hop-to-hop (node-to-node) delivery** by the data link layer.

Figure 2.7 *Hop-to-hop delivery*



As the figure shows, communication at the data link layer occurs between two adjacent nodes. To send data from A to F, three partial deliveries are made. First, the data link layer at A sends a frame to the data link layer at B (a router). Second, the data

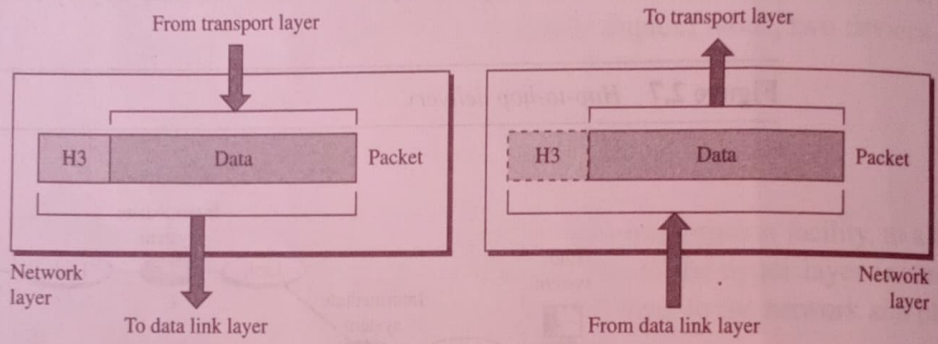
link layer at B sends a new frame to the data link layer at E. Finally, the data link layer at E sends a new frame to the data link layer at F. Note that the frames that are exchanged between the three nodes have different values in the headers. The frame from A to B has B as the destination address and A as the source address. The frame from B to E has E as the destination address and B as the source address. The frame from E to F has F as the destination address and E as the source address. The values of the trailers can also be different if error checking includes the header of the frame.

Network Layer

The **network layer** is responsible for the source-to-destination delivery of a packet, possibly across multiple networks (links). Whereas the data link layer oversees the delivery of the packet between two systems on the same network (links), the network layer ensures that each packet gets from its point of origin to its final destination.

If two systems are connected to the same link, there is usually no need for a network layer. However, if the two systems are attached to different networks (links) with connecting devices between the networks (links), there is often a need for the network layer to accomplish source-to-destination delivery. Figure 2.8 shows the relationship of the network layer to the data link and transport layers.

Figure 2.8 *Network layer*



The network layer is responsible for the delivery of individual packets from the source host to the destination host.

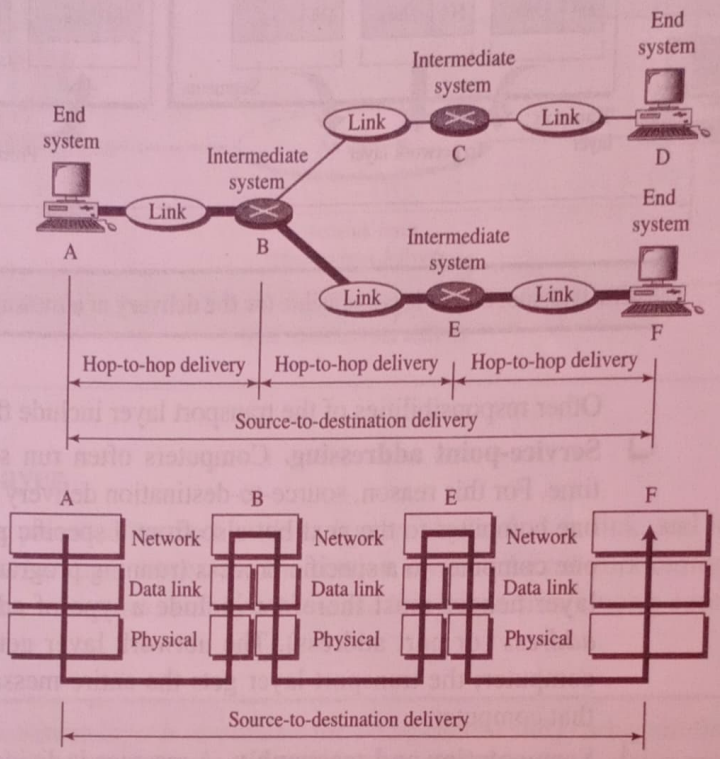
Other responsibilities of the network layer include the following:

- **Logical addressing.** The physical addressing implemented by the data link layer handles the addressing problem locally. If a packet passes the network boundary, we need another addressing system to help distinguish the source and destination systems. The network layer adds a header to the packet coming from the upper layer that, among other things, includes the logical addresses of the sender and receiver. We discuss logical addresses later in this chapter.
- **Routing.** When independent networks or links are connected to create *internetworks* (network of networks) or a large network, the connecting devices (called *routers*

or *switches*) route or switch the packets to their final destination. One of the functions of the network layer is to provide this mechanism.

Figure 2.9 illustrates end-to-end delivery by the network layer.

Figure 2.9 *Source-to-destination delivery*

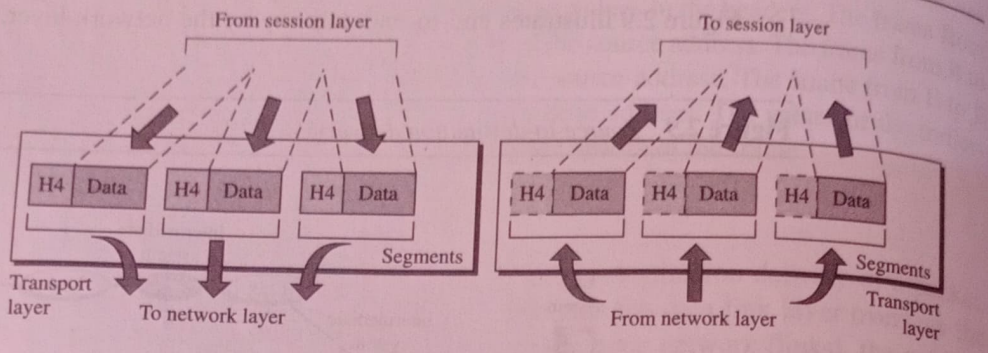


As the figure shows, now we need a source-to-destination delivery. The network layer at A sends the packet to the network layer at B. When the packet arrives at router B, the router makes a decision based on the final destination (F) of the packet. As we will see in later chapters, router B uses its routing table to find that the next hop is router E. The network layer at B, therefore, sends the packet to the network layer at E. The network layer at E, in turn, sends the packet to the network layer at F.

Transport Layer

The **transport layer** is responsible for **process-to-process delivery** of the entire message. A process is an application program running on a host. Whereas the network layer oversees **source-to-destination delivery** of individual packets, it does not recognize any relationship between those packets. It treats each one independently, as though each piece belonged to a separate message, whether or not it does. The transport layer, on the other hand, ensures that the whole message arrives intact and in order, overseeing both error control and flow control at the source-to-destination level. Figure 2.10 shows the relationship of the transport layer to the network and session layers.

Figure 2.10 *Transport layer*



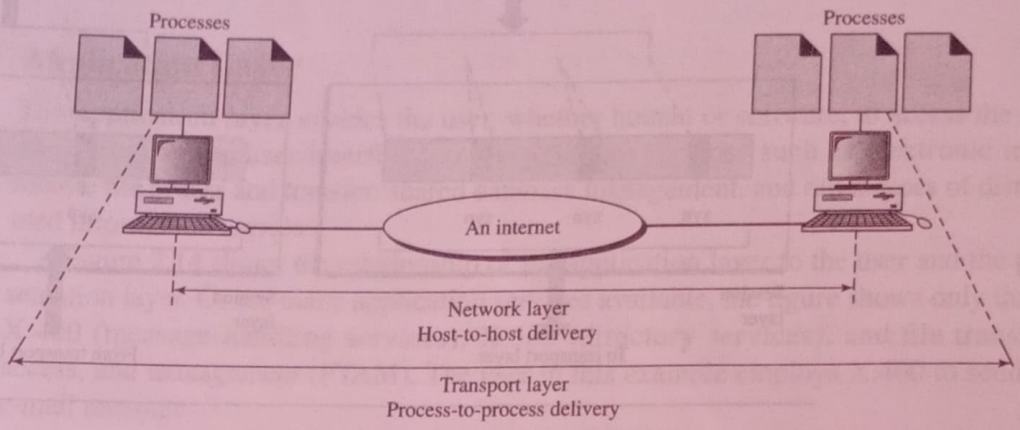
The transport layer is responsible for the delivery of a message from one process to another.

Other responsibilities of the transport layer include the following:

- **Service-point addressing.** Computers often run several programs at the same time. For this reason, source-to-destination delivery means delivery not only from one computer to the next but also from a specific process (running program) on one computer to a specific process (running program) on the other. The transport layer header must therefore include a type of address called a *service-point address* (or port address). The network layer gets each packet to the correct computer; the transport layer gets the entire message to the correct process on that computer.
- **Segmentation and reassembly.** A message is divided into transmittable segments, with each segment containing a sequence number. These numbers enable the transport layer to reassemble the message correctly upon arriving at the destination and to identify and replace packets that were lost in transmission.
- **Connection control.** The transport layer can be either connectionless or connection-oriented. A connectionless transport layer treats each segment as an independent packet and delivers it to the transport layer at the destination machine. A connection-oriented transport layer makes a connection with the transport layer at the destination machine first before delivering the packets. After all the data are transferred, the connection is terminated.
- **Flow control.** Like the data link layer, the transport layer is responsible for **flow control.** However, flow control at this layer is performed end to end rather than across a single link.
- **Error control.** Like the data link layer, the transport layer is responsible for **error control.** However, error control at this layer is performed process-to-process rather than across a single link. The sending transport layer makes sure that the entire message arrives at the receiving transport layer without **error** (damage, loss, or duplication). Error correction is usually achieved through retransmission.

Figure 2.11 illustrates process-to-process delivery by the transport layer.

Figure 2.11 *Reliable process-to-process delivery of a message*



Session Layer

The services provided by the first three layers (physical, data link, and network) are not sufficient for some processes. The **session layer** is the network *dialog controller*. It establishes, maintains, and synchronizes the interaction among communicating systems.

The session layer is responsible for dialog control and synchronization.

Specific responsibilities of the session layer include the following:

- **Dialog control.** The session layer allows two systems to enter into a dialog. It allows the communication between two processes to take place in either half-duplex (one way at a time) or full-duplex (two ways at a time) mode.
- **Synchronization.** The session layer allows a process to add checkpoints, or **synchronization points,** to a stream of data. For example, if a system is sending a file of 2000 pages, it is advisable to insert checkpoints after every 100 pages to ensure that each 100-page unit is received and acknowledged independently. In this case, if a crash happens during the transmission of page 523, the only pages that need to be resent after system recovery are pages 501 to 523. Pages previous to 501 need not be resent. Figure 2.12 illustrates the relationship of the session layer to the transport and presentation layers.

Presentation Layer

The **presentation layer** is concerned with the syntax and semantics of the information exchanged between two systems. Figure 2.13 shows the relationship between the presentation layer and the application and session layers.

Figure 2.12 *Session layer*

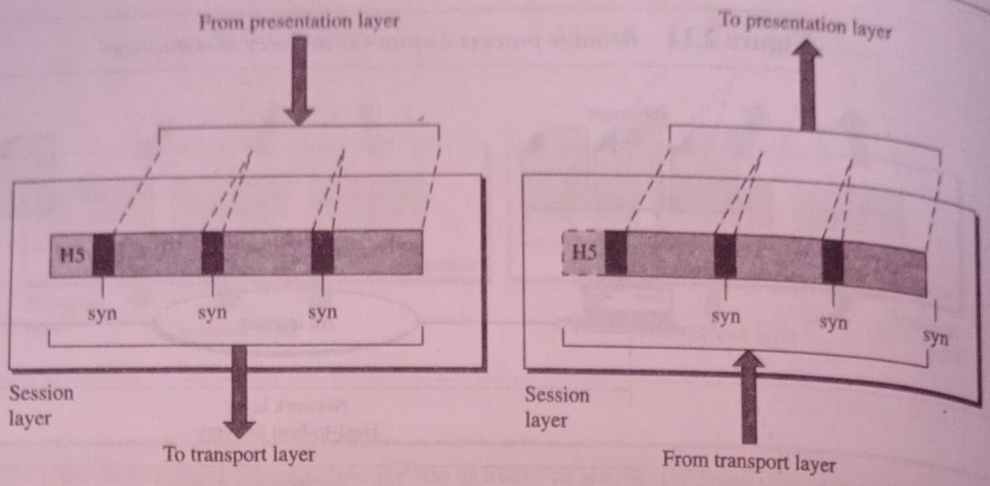
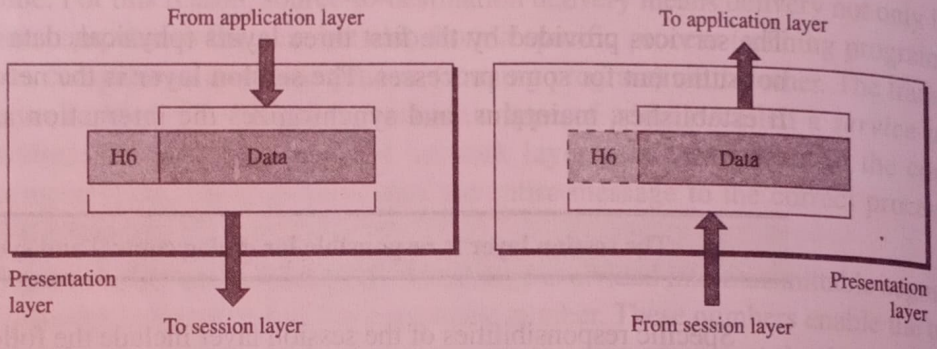


Figure 2.13 *Presentation layer*



The presentation layer is responsible for translation, compression, and encryption.

Specific responsibilities of the presentation layer include the following:

- **Translation.** The processes (running programs) in two systems are usually exchanging information in the form of character strings, numbers, and so on. The information must be changed to bit streams before being transmitted. Because different computers use different encoding systems, the presentation layer is responsible for interoperability between these different encoding methods. The presentation layer at the sender changes the information from its sender-dependent format into a common format. The presentation layer at the receiving machine changes the common format into its receiver-dependent format.
- **Encryption.** To carry sensitive information, a system must be able to ensure privacy. Encryption means that the sender transforms the original information to

another form and sends the resulting message out over the network. Decryption reverses the original process to transform the message back to its original form.

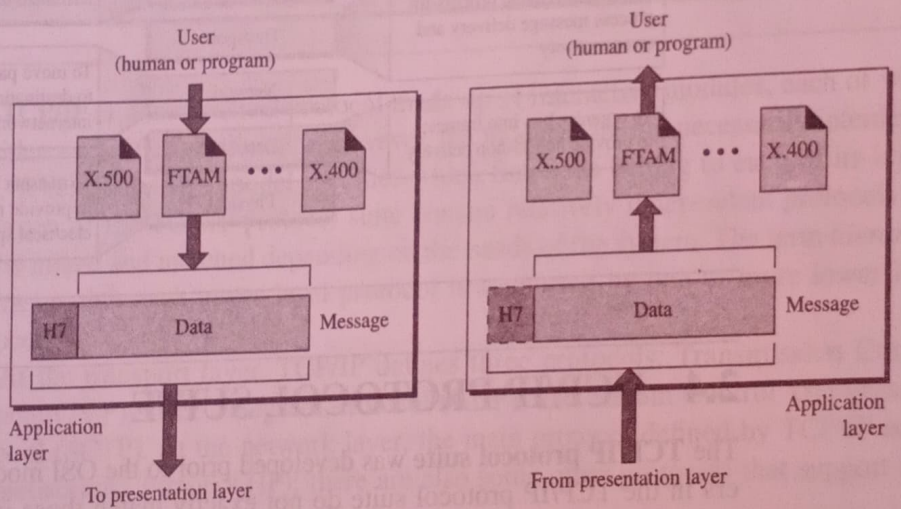
- **Compression.** Data compression reduces the number of bits contained in the information. Data compression becomes particularly important in the transmission of multimedia such as text, audio, and video.

Application Layer

The **application layer** enables the user, whether human or software, to access the network. It provides user interfaces and support for services such as electronic mail, remote file access and transfer, shared database management, and other types of distributed information services.

Figure 2.14 shows the relationship of the application layer to the user and the presentation layer. Of the many application services available, the figure shows only three: X.400 (message-handling services), X.500 (directory services), and file transfer, access, and management (FTAM). The user in this example employs X.400 to send an e-mail message.

Figure 2.14 *Application layer*



The application layer is responsible for providing services to the user.

Specific services provided by the application layer include the following:

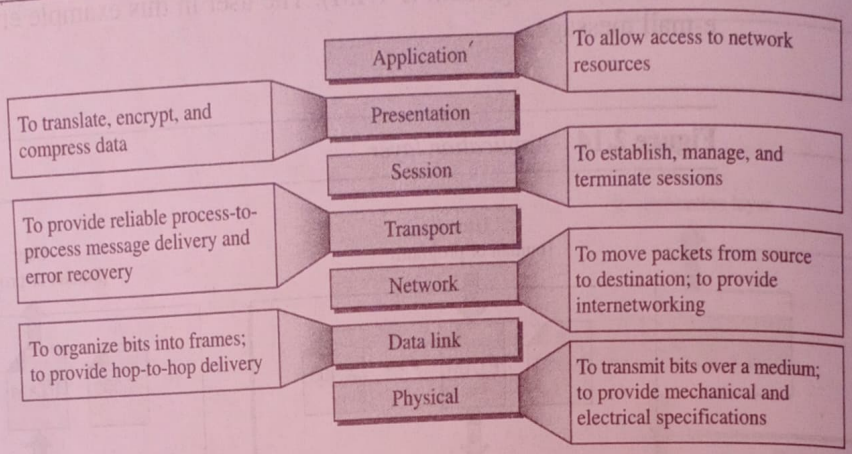
- **Network virtual terminal.** A network virtual terminal is a software version of a physical terminal, and it allows a user to log on to a remote host. To do so, the application creates a software emulation of a terminal at the remote host. The user's computer talks to the software terminal which, in turn, talks to the host, and vice versa. The remote host believes it is communicating with one of its own terminals and allows the user to log on.

- **File transfer, access, and management.** This application allows a user to access files in a remote host (to make changes or read data), to retrieve files from a remote computer for use in the local computer, and to manage or control files in a remote computer locally.
- **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 Layers

Figure 2.15 shows a summary of duties for each layer.

Figure 2.15 *Summary of layers*



2.4 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 exactly match those in the OSI model. The original TCP/IP protocol suite was defined as having four layers: host-to-network, internet, transport, and application. However, when TCP/IP is compared to OSI, we can say that the host-to-network layer is equivalent to the combination of the physical and data link layers. The internet layer is equivalent to the network layer, and the application layer is roughly doing the job of the session, presentation, and application layers with the transport layer in TCP/IP taking care of part of the duties of the session layer. So in this book, we assume that the TCP/IP protocol suite is made of five layers: physical, data link, network, transport, and application. The first four layers provide physical standards, network interfaces, internetworking, and transport functions that correspond to the first four layers of the OSI model. The three topmost layers in the OSI model, however, are represented in TCP/IP by a single layer called the *application layer* (see Figure 2.16).