Ministry of Higher Education and Scientific

Research Al-Mustaqbal University College

Department of Technical Computer Engineering

# Lecture Number: 10, 11

# Computer Networks 3rd Stage

Lecturer: Dr. Hussein Ali Ameen

hussein_awadh@uomus.edu.iq

2022-2023

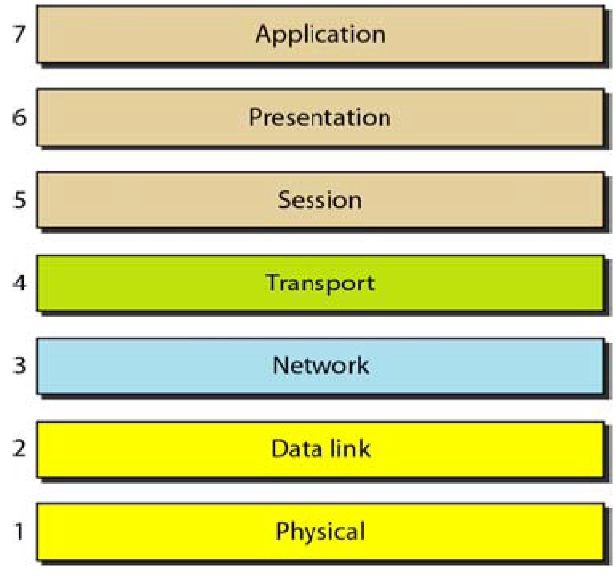
# Chapter Two

# The OSI Model

The International Standards Organization **(ISO)** is a multinational body dedicated to worldwide agreement on international standards (Established in 1947). An ISO standard that covers all aspects of network communications is the **O**pen **S**ystems **I**nterconnection **(OSI)** model. 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 open communication between different systems without requiring, changes to the logic of the underlying hardware and software. The OSI is not a protocol, it is a model for understanding and designing a network architecture.

## 2.1- The Model

The Open Systems Interconnection model is a layered framework for the design of network system that allows for communication across all types of computer systems. It consists of seven separate but related layers, each of which defines a segment of the process of moving information across a network:



### 2.1.1- OSI Layered Architecture:

Each layer defines a family of functions distinct from those of the other layers. By defining and localizing functionality in this fashion, the designers created an architecture that is both comprehensive and flexible. Figure (2.1) shows the layers involved when a message sent from device **A** to device **B**. As the message travels from A to B, it may pass through many intermediate nodes. The intermediate nodes usually involve only the first three layers of the OSI model.

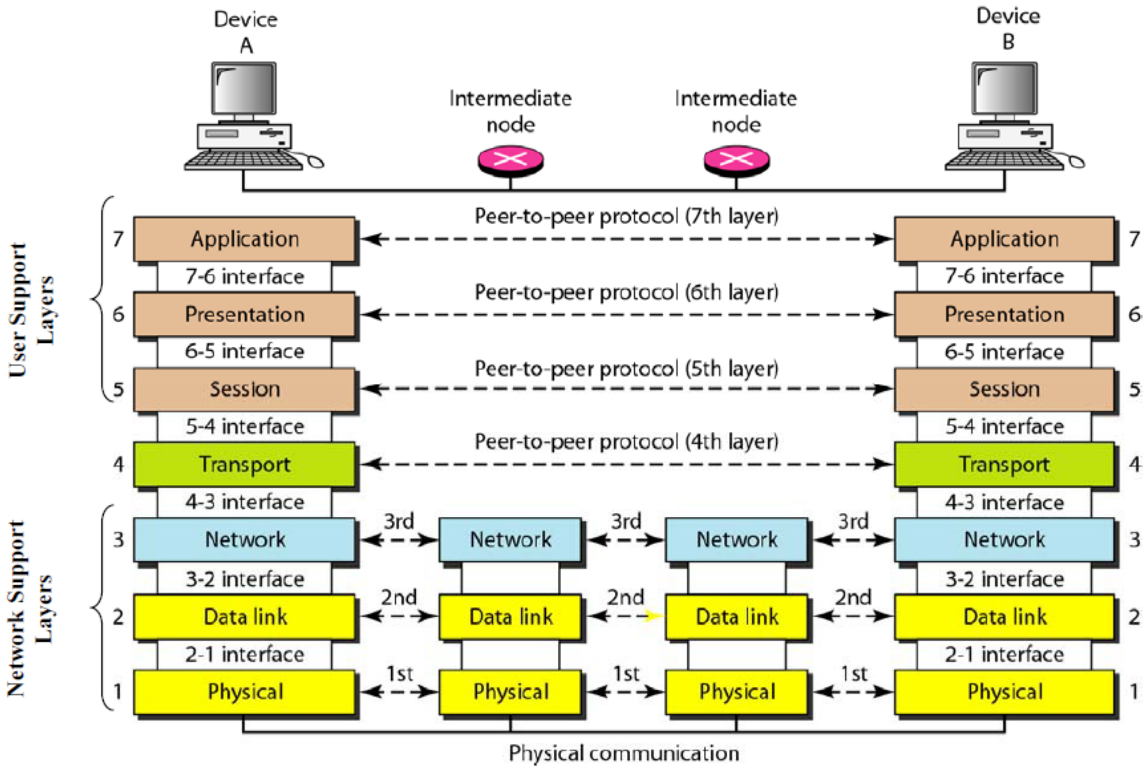


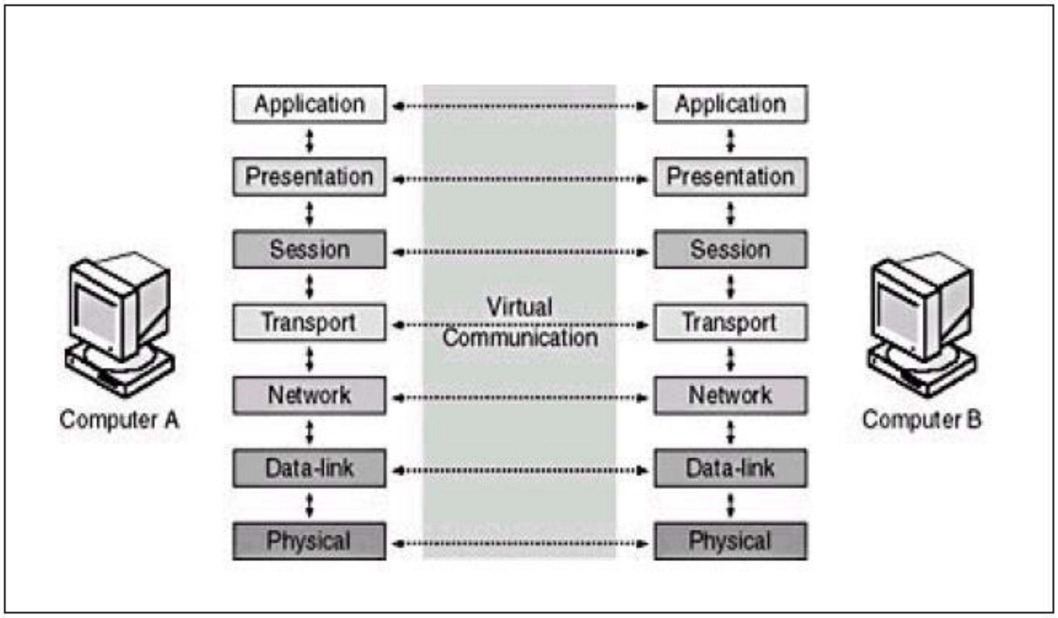

***Figure (2.1) shows the layers involved when a message sent from device A to device B.***

### 2.1.2- Interfaces between Layers

Within a single machine, the passing of the data and network information down through the layers of the sending machine and back up through the layers of the receiving machine is made possible by an interface between each pair of adjacent layers. Each interface defines what information and services a layer must provide for the layer above it. Layer 3, for example, uses the services provided by layer 2

### 2.1.3- OSI Peer-to-Peer Processes

Between machines, layer x on one machine communicates with layer x on another machine. This communication is governed by an agreed upon series of rules and conventions called **protocol.** The processes on each machine that communicate at a given layer called **peer-to-peer processes**. Each layer in the sending machine **adds** its own information to the message it receives from the layer just above it, and **passes** the whole package to the layer just below it. This information added in the form of **headers** or **trailers** (control data appended to the beginning or end of a data parcel).




### 2.1.4- Encapsulation and Protocol data unit (PDU)

As application data is passed down the protocol stack on its way to be transmitted across the network media, various protocols add information to it at each level. This is commonly known as the encapsulation process.

The form that a piece of data takes at any layer is called a protocol data unit (PDU). During encapsulation, each succeeding layer encapsulates the PDU that it receives from the layer above in accordance with the protocol being used. At each stage of the process, a PDU has a different name to reflect its new functions.

- **Data -** The general term for the PDU used at the application layer
- **Segment -** Transport layer PDU
- **Packet -** Network layer PDU
- **Frame -** Data Link layer PDU
- **Bits -** A PDU used when physically transmitting data over the medium

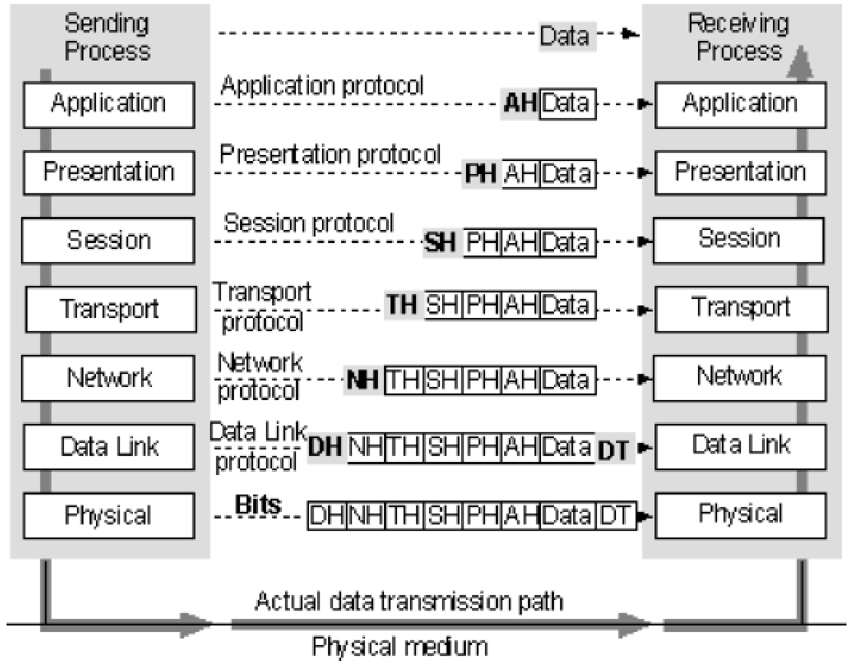



***Figure (2.2): The encapsulation process***

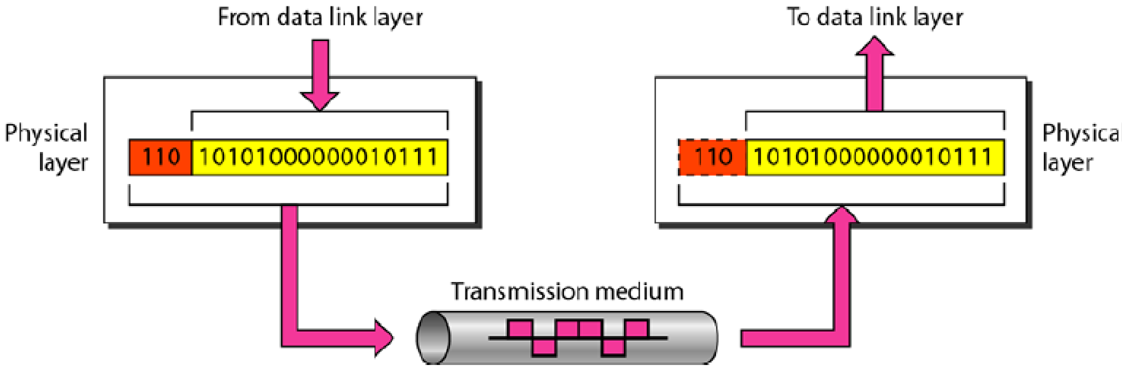
Figure (2.2), demonstrates the encapsulation process. The process starts at layer 7 (the application layer), then moves from layer to layer in descending, sequential order. At each layer, a header, or possibly a trailer, can be added to the data unit. Commonly, the trailer is added only at layer 2. When the formatted data unit passes through the physical layer (layer 1), it is changed into an electromagnetic signal and transported along a physical link.

Upon reaching its destination, the signal passes into layer 1 and is transformed back into digital form. The data units then move back up through the OSI layers. As each block of data reaches the next higher layer, the headers and trailers attached to it at the corresponding sending layer are removed, and actions appropriate to that layer are taken. By the time it reaches layer 7, the message is again in a form appropriate to the application and is made available to the recipient.

## 2.2- Functions of the Layers

### 2.2.1- Physical Layer

The physical layer coordinates the functions required to transmit a bit streams over a physical medium. It **deals with** the mechanical and electrical specifications of the primary connections, such as cables, connectors, and signalling options that physically link two nodes on a network. This first layer receives a data unit from the second layer and puts it into a format capable of being carried by a communications link. It oversees the changing of a bit streams into electromagnetic signals, and their transmission onto and across a medium.

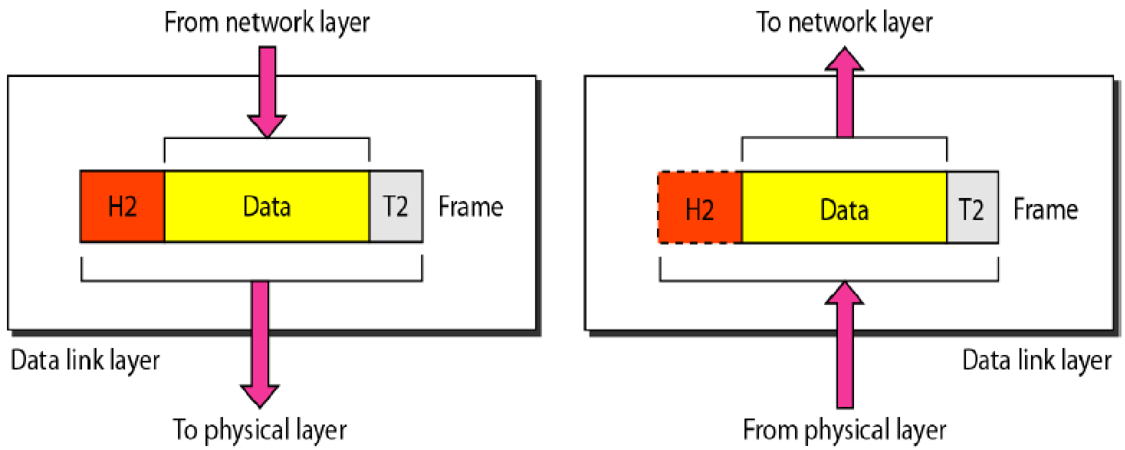



**<u>This task requires a number of considerations:</u>**

- **Line configuration**: How can two or more devices are linked physically?
Are transmission lines to be shared or limited to use between two devices?
Is the line available or not?
- **Data transmission mode:** It does transmission flow one-way or both ways between two connected devices. Or does it alternate?
- **Topology:** How network devices arranged? Do they pass data directly to each other or through an intermediary? And by what paths?
- **Signals:** What type of signals is useful for transmitting information?
- **Encoding:** How are bits (0s and 1s) to be represented by available signalling systems? How data are represented by signals?
- **Interface**: What information must be shared between two closely linked devices to enable and facilitate communication? What is the most efficient way to communicate that information?
- **Medium**: What is the physical environment for the transmission of data?
- **Multiplexing:** Using a single physical line to carry data between many devices at the same time.

## 2.2.2- Data link Layer

The data link layer is **responsible for** delivering data units (groups of bits) from one station to the next without errors. It accepts a data unit from the third layer and **adds** meaningful bits to the beginning (header) and end (trailer) that contain addresses and of control information. A data unit with this additional information called a **Frame**.

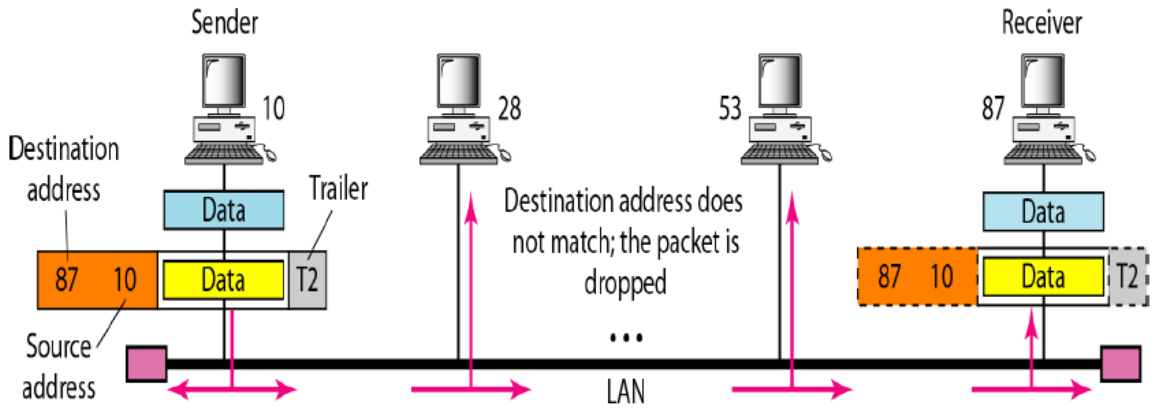



**The responsibilities of the data link layer include the following:**

- **Node-to-node delivery:** The data link layer is responsible for node-to-node delivery.
- **Physical Addressing:** Headers and trailers added at this layer include the physical address of the most recent node and the next intended node.
- **Access control:** When two or more devices connected to the same link, the data-link layer protocols are necessary to determine which device has control over the line at any given time.
- **Flow control:** To avoid overwhelming the receiver, the data link layer regulates the amount of data that can be transmitted at one time. It adds identifying numbers to enable the receiving node to control the ordering of the frames.
- **Error handling:** Data link layer protocols provide for data recovery, usually by having the entire frame retransmitted.

**Example:**

In Figure below a node with physical address 10 sends a frame to a node with physical address 87. The two nodes connected by a link. At the data link level, this frame contains physical (link) addresses in the header. These are the only addresses needed. The rest of the header contains other information needed at this level. The trailer usually contains extra bits needed for error detection.



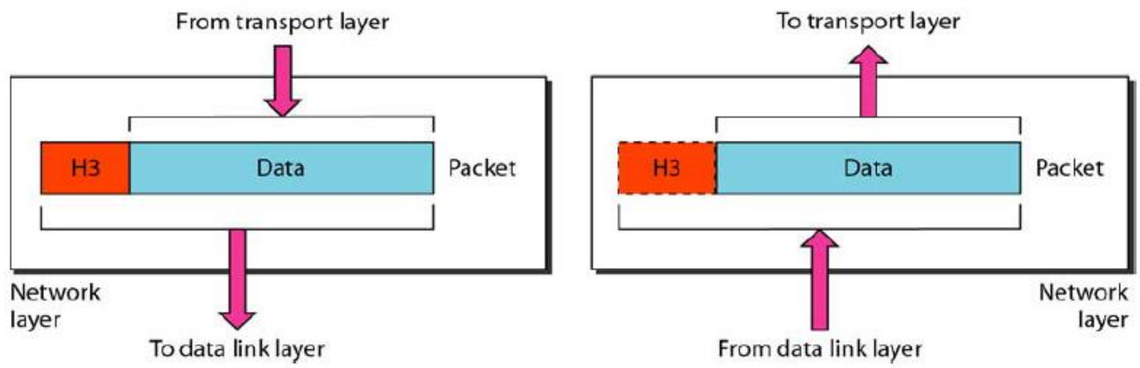

## 2.2.3- Network Layer

The network layer is **responsible for** the source-to-destination delivery of a packet across multiple network links. Whereas the data link layer oversees node-to-node delivery, the network layer **ensures** that each **packet** gets from its point of origin to its final destination successfully and efficiently.
To make such end-to-end delivery possible, the network layer **provides** two related services: Addressing and routing**.**

**Addressing end devices -** In the same way that a phone has a unique telephone number, end devices must be configured with a unique IP address for identification on the network. An end device with a configured IP address is referred to as a host.
**Routing** means selecting the best path for sending a packet from one point to another, when more than one path is available.

Routing requires the addition of a header that includes, among other, information, the source and destination addresses of the packet. These addresses are different from the physical (node) addresses included in the data link header; it is known as a **logical address**.

Physical addresses are of current to next node only; they are changes as a frame move from one node to another. Whereas the logical addresses are those for the original source to the final destination, and do not changed during packet transmission.

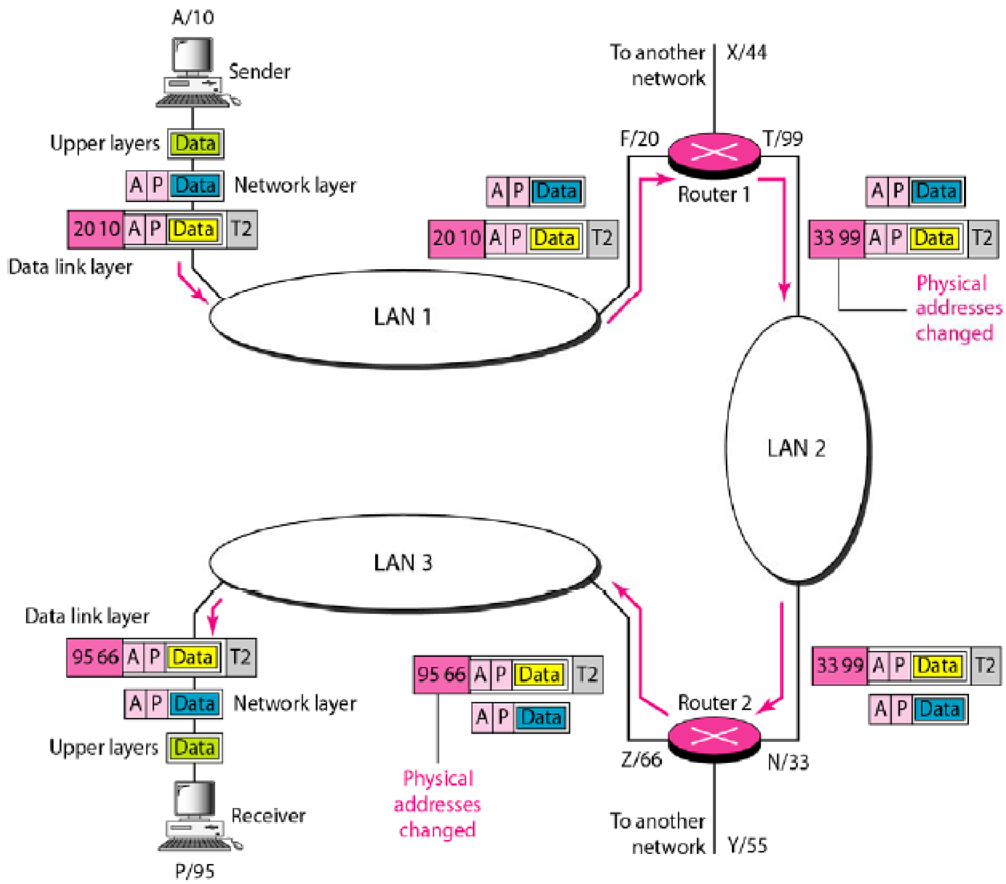



***Specific responsibilities of the network layer include the following:***

- **Source-to-destination delivery:** Moving a packet (best effort) from its point of origin to its intended destination across multiple network links.,
- **Logical addressing:** Inclusion of the source and destination addresses in the header.
- **Routing:** Deciding which of multiple paths a packet should take.
- **Address transformation:** Interpreting logical addresses to find their physical equivalents.

**Example**

Now imagine that in Figure below we want to send data from a node with network address **A**, physical address 10, located on one local area network, to a node with a network address **P**, physical address 95, located on another local area network. Because the two devices are located on different networks, we cannot use link addresses only. What we need here are universal addresses that can pass through the boundaries of local area networks. The network (logical) addresses have this characteristic. The packet at the network layer contains the logical addresses, which remain the same from the original source to the destination (A and P. respectively, in the figure). They will not change when we go from network to network. However, the physical addresses will change when the packet moves from one network to another. The box with the R is a router (intemetwork device), which we will discuss later.

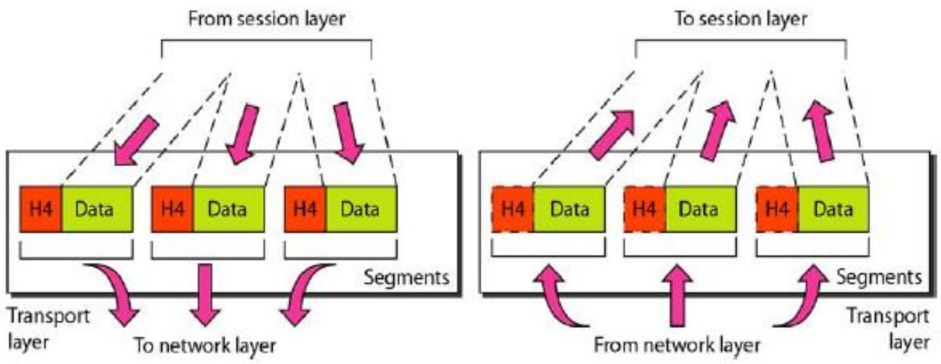



## 2.2.4- Transport Layer

The transport layer is **responsible for** source -to-destination (end-to-end) delivery of the entire message. Whereas the network layer oversees end-to-end 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.

Computers often run several programs at the same time. For this reason source-todestination delivery means delivery not only from one computer to the next but also from a specific application on one computer to a specific application on the other. The transport layer header must therefore include a type of address called a service point address (also called a **port address** or socket address). The network layer gets each packet to the correct computer; the transport layer gets the entire message to the correct application on that computer.

As the transport layer receives the message from session layer it divides the message into a transmittable segments, indicating in the header the sequence of the segments so that it can be reassembled upon receipt at the destination.

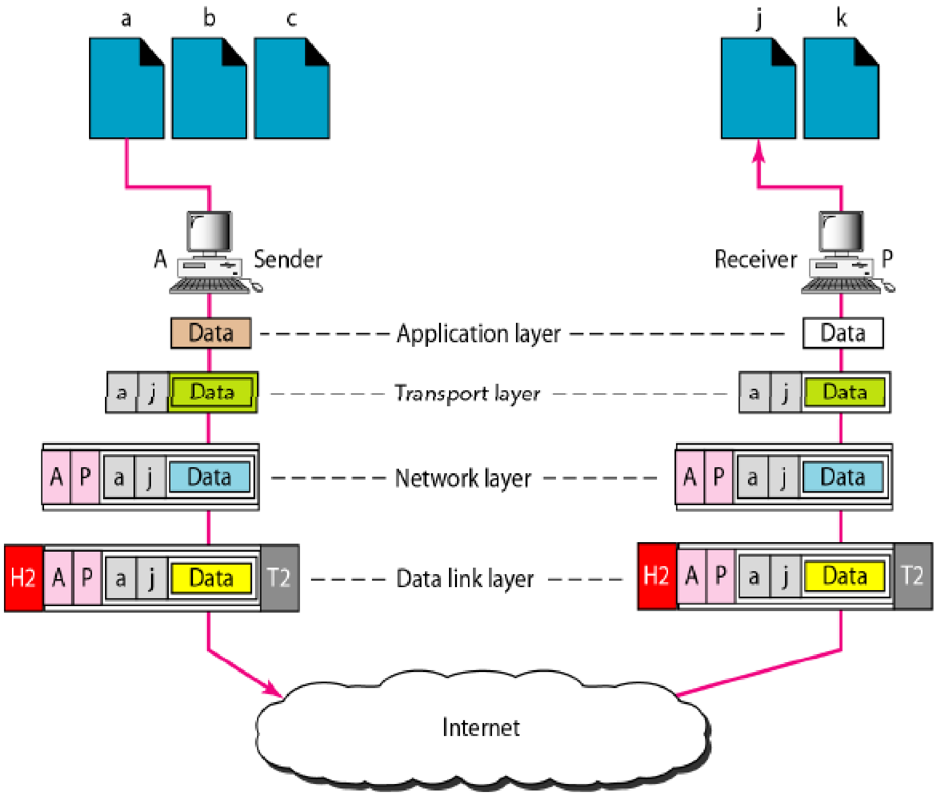



**Specific responsibilities of the transport layer include the following:**

- **End-to-end message delivery:** Overseeing the transmission and arrival of all packets of a message at the destination point.
- **Service-point (port) addressing:** Guaranteeing delivery of a message to the appropriate application on **a** computer running multiple applications.
- **Segmentation and reassembly:** Dividing a message into transmittable segments and marking each segment with a sequence number. These numbers enable the transport layer to reassemble the message correctly at the destination and to identify and replace packets lost in transmission.
- **Connection control:** Deciding whether or not to send all packets by a single path.

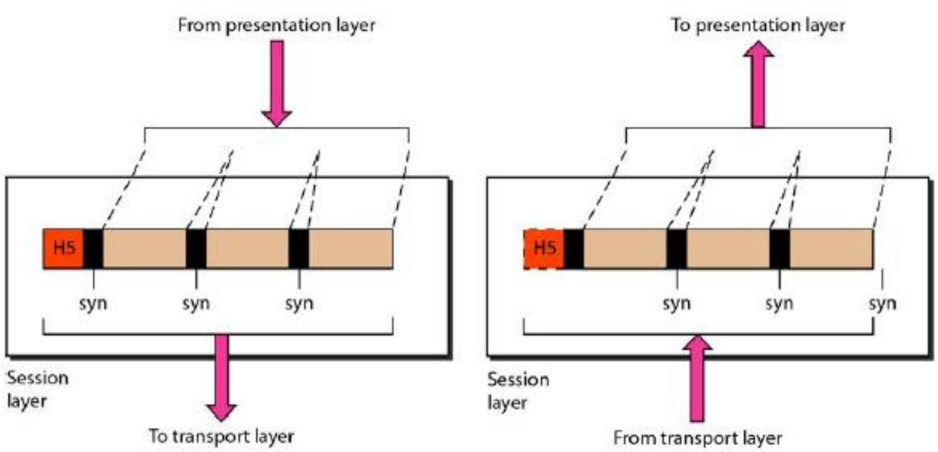
**Example:**

Figure below shows an example of a transport layer. Data coming from the upper layers have Service-point (port) addresses (a) and (j) ( a is the address of the sending application, and j is the address of the receiving application). Then in the network layer, network addresses (A and P) are added to each packet. The packets may travel on different paths and arrive at the destination either in order or out of order. The packets are delivered to the destination transport layer, which is responsible for removing the network layer headers and the delivery to the upper layers.




## 2.2.5- Session Layer

The session layer is the network **dialog controller**. It establishes, maintains, and synchronizes the interaction between' communicating devices. It also **ensures** that each session closes appropriately rather than shutting down abruptly and leaving the user hanging.



**Specific responsibilities of the session layer include the following:**

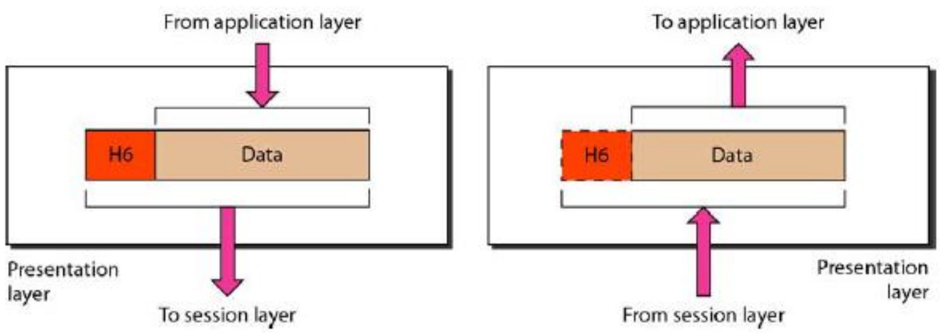
- **Session management:** Dividing a session into sub sessions by the introduction of checkpoints and separating long messages into shorter units called dialog units appropriate for transmission.
- **Synchronization:** Deciding in what order to pass the dialog units to the transport layer and where in the transmission to require confirmation from the receiver.
- **Dialog control:** Deciding who sends, and when.
- **Graceful close:** Ensuring that the exchange has been completed appropriately before the session closes.

**Example**

A computer needs to update a huge file (e.g. a database). The session layer subdivides the task into different dialog units.

## 2.2.6- Presentation Layer

The presentation layer **ensures** interoperability among communicating devices. Functions at this layer make it possible for two computers to communicate even if their internal representations of data differ (e.g., when one device uses one type of code and the other uses another). It **provide** necessary translation of different control codes and character sets, graphics characters, and so on to allow both devices to understand the same transmission the same way.



**Specific responsibilities of the presentation layer include following:**

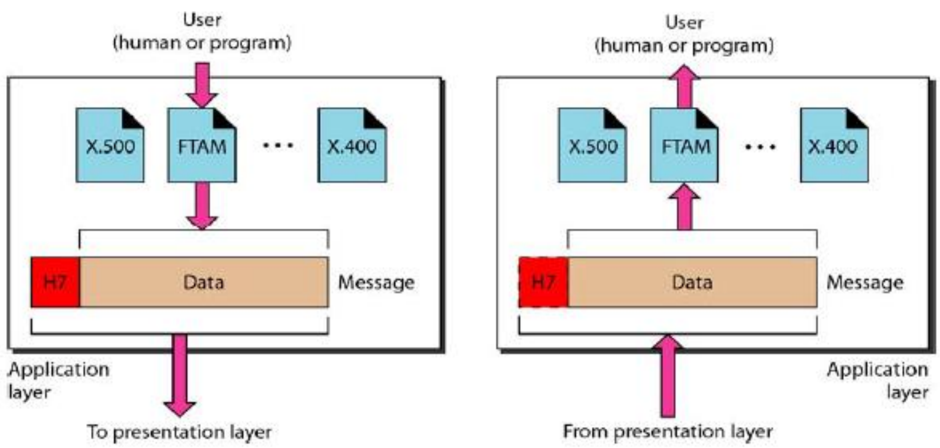
- **Translation:** Changing the format of a message from that used by the sender into one mutually acceptable for transmission. Then, at the destination, changing that format into the one understood by the receiver.
- **Encryption:** Encryption and decryption of data for security purposes.
- **Compression:** Compressing and decompressing data to make transmission more efficient.
- **Security:** Validating passwords and log-in codes.

**Example**

The sending station uses an encryption algorithm to protect the data from eavesdropping. The encrypted data are decrypted at the destination presentation layer before being delivered to the application layer.

## 2.2.7- 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.




**Specific services provided by the application layer include the following:**

- **Web Services :** Allows accessing web content in the form of text, images and multimedia using the HTTP protocol
- **File access, transfer, and management:** Allows a user at a remote computer to access files in another host (to make changes or read data); to retrieve files from remote computer for use in the local computer; and to

manage or control files in remote computer at that computer.

- **Mail services:** Provides the basis for electronic mail forwarding and storage.
- **Directory services:** Provides distributed database sources and access for global information about various objects.

**Example**

A user in Beijing, China, wants to send a large proprietary data file to a station in Los Gatos, California. An application service such as FTAM (file transfer and access management) can do the job.

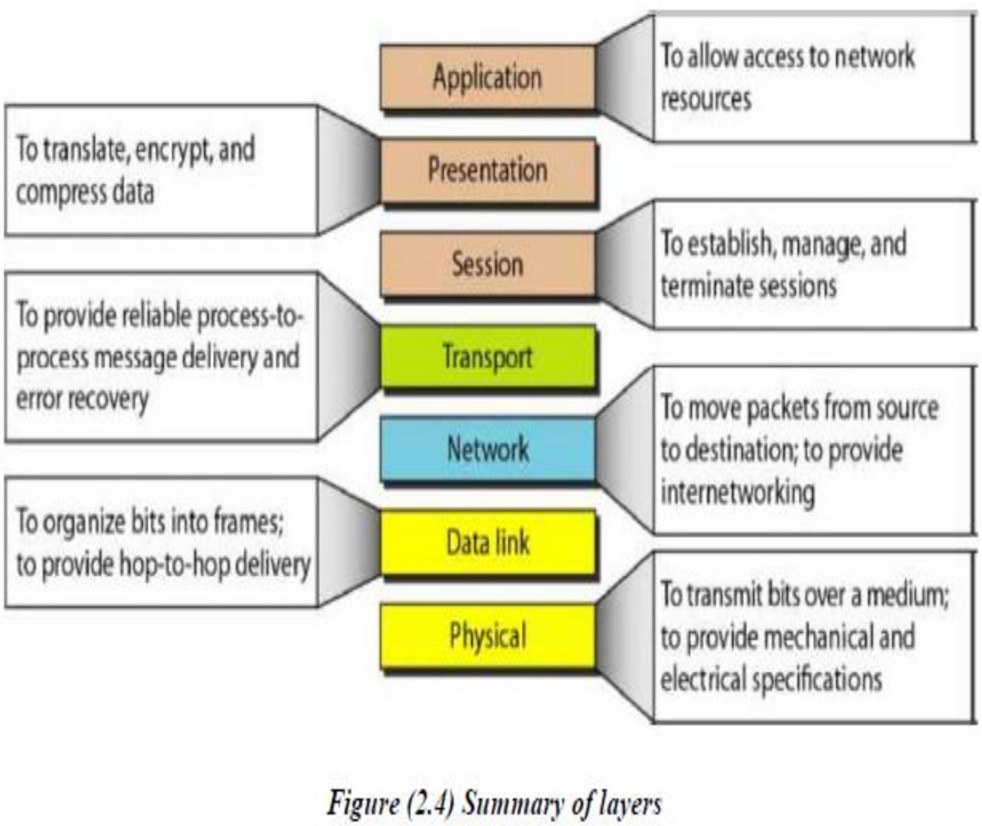
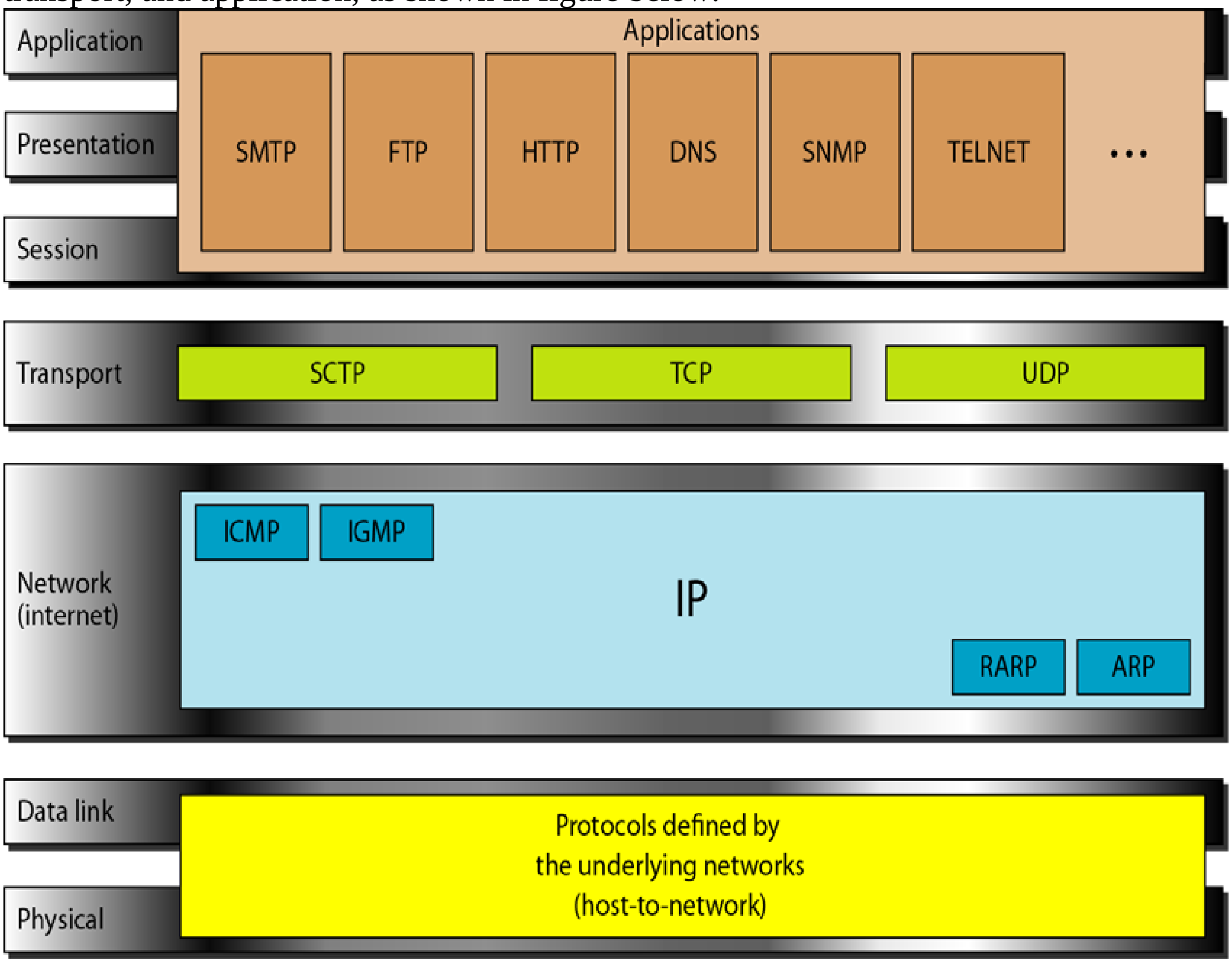



*Figure (2.4) Summary of layers*

## 2.3- TCP/IP Protocol Suite

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 TCP/IP protocol suite is made of five layers: physical, data link, network, transport, and application, as shown in figure below.



***<u>The TCP/IP protocol stack is the de-facto standard in networking.</u>***

- It is an alternative for the OSI 7-layer model, which has never really been implemented in practice.
- TCP/IP is an open standard and is the protocol used over the Internet.
- It can be found in most modern day operating systems

***<u>The main differences between TCP/IP and the OSI 7-layer model are:</u>***

- Number of layers

- TCP/IP defines only 4 or 5 layers.

- TCP/IP protocol model while OSI reference model
- In TCP/IP application layer equal to application ,presentation and session layers in OSI

- Functions performed at a given layer

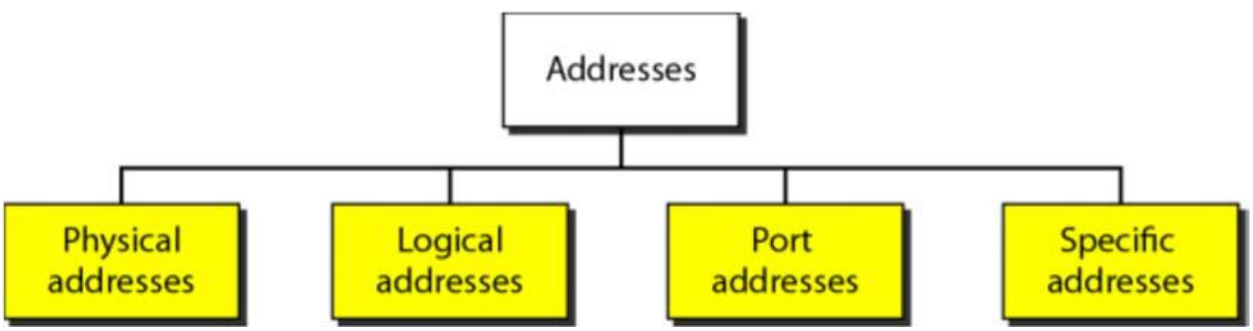
- In the OSI model each layer performs specific functions
- In TCP/IP different protocols may be defined within a layer, each performing different functions. What is common about a set of protocols at the same layer is that they share the same set of support protocols at the next lower layer.

- Interface between adjacent layers

- In the OSI model, a protocol at a given layer may be substituted by a new one without impacting on adjacent layers.
- In TCP/IP the strict use of all layers is not mandated

**2.4- Addressing**

Four levels of addresses are used in an internet employing the TCP/IP protocols: physical (link) addresses, logical (IP) addresses, port addresses, and specific addresses (see Figure 16).



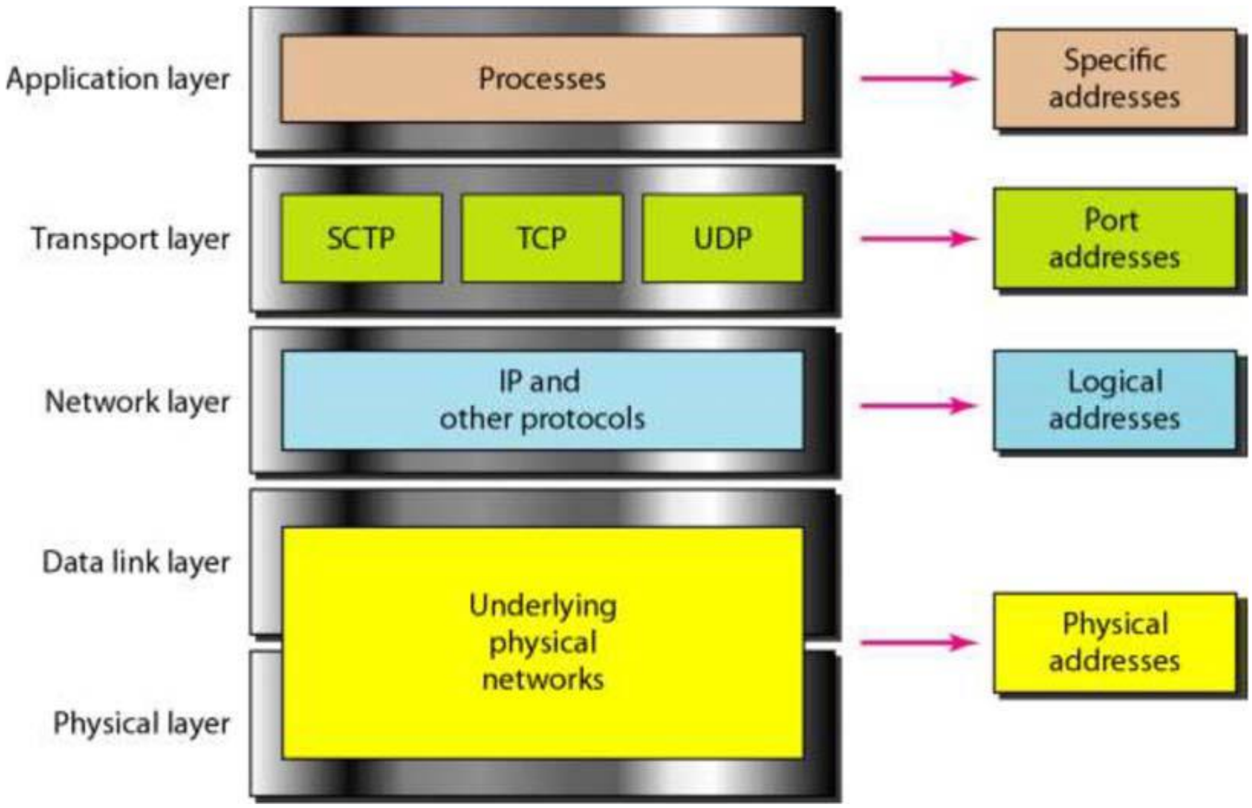



**Figure (2.6): Relationship of layers and addresses in TCP/IP**

### 2.4.1-Physical Addresses

The physical address, also known as the link address or mac address ,it is the address used in local network it has a 6-byte (48-bit) physical address that is imprinted on the network interface card(NIC) example 07:01:02:01 :2C:4B A 6-byte (12 hexadecimal digits) physical address

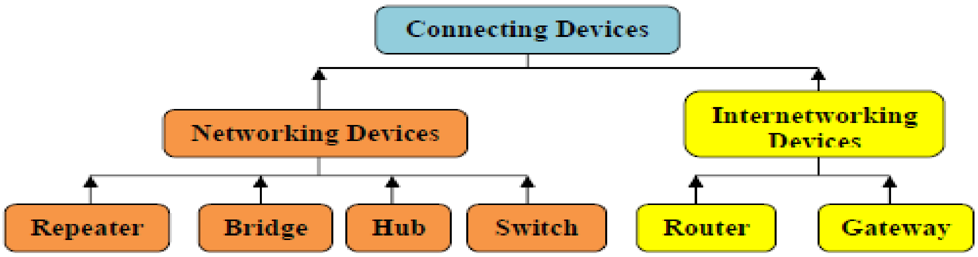
### 2.4.2-Logical Addresses (IP)

Logical addresses are necessary for network to network communication A logical address in the Internet is currently a 32-bit address that can uniquely define a host connected to the Internet. No two publicly addressed and visible hosts on the Internet can have the same IP address.

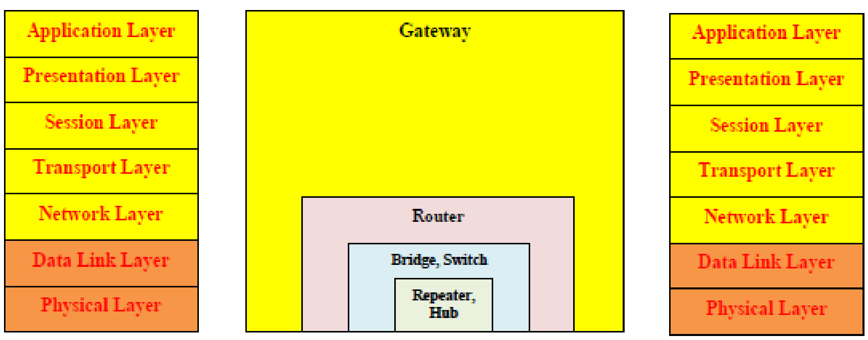
**2.4.3- Port Addresses**
It is implemented in transport layer as process to process delivering

## 2.5- Networking and Internetworks

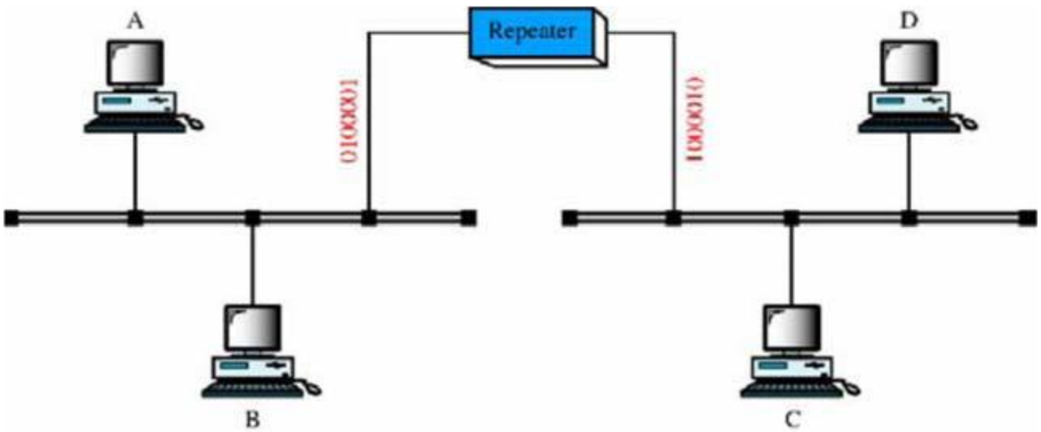
To connect a network or multiple networks together there are different types of connecting devices as illustrated in figure below:




Each of these devices operates at different layer of the OSI model as shown in figure below:




**2.5.1- Repeater:**
Devices used to extend the network cable length beyond the limit of the specified cable. The purpose of a repeater is to regenerate and retime network signals at the bit level to allow them to travel a longer distance on the media. Repeaters are classified as Layer 1 devices in the OSI model, because they act only on the bit level and look at no other information.

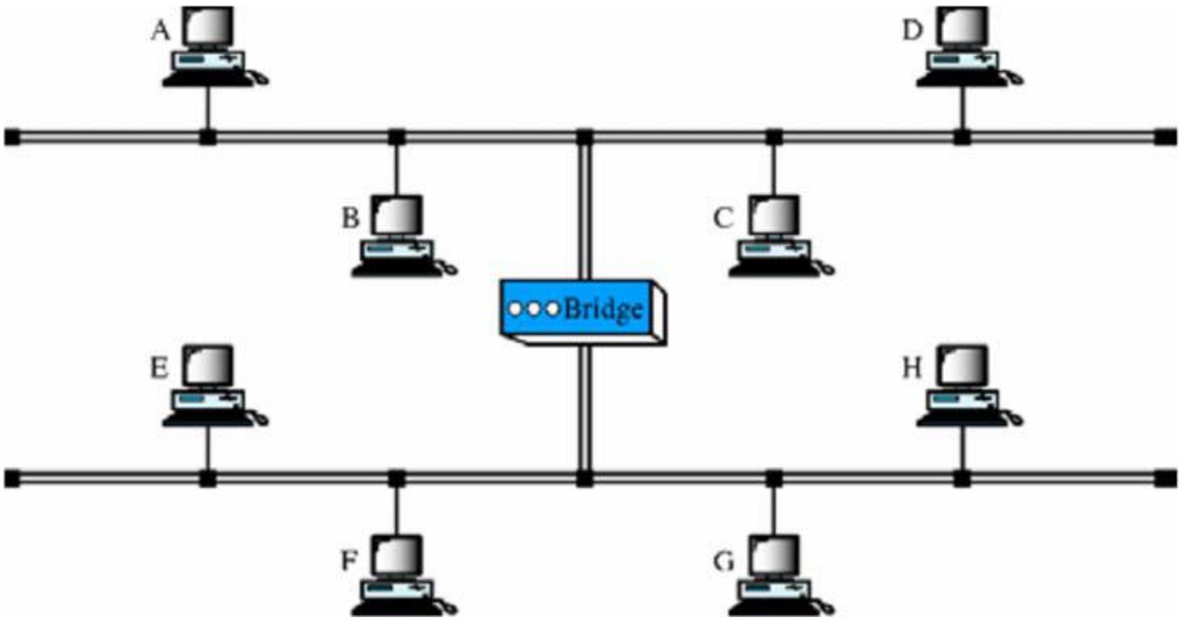



**2.5.2- Hubs**

The purpose of a hub is to regenerate and retime network signals. This is done at the bit level to a large number of hosts (e.g. 4, 8, or even 24) using a process known as concentration. You will notice that this definition is very similar to the repeaters, which is why a hub is also known as a multi-port repeater. The difference is the number of cables that connect to the device. Two reasons for using hubs are to create a central connection point for the wiring media, and increase the reliability of the network. The reliability of the network is increased by allowing any single cable to fail without disrupting the entire network. This differs from the bus topology where having one cable fail will disrupt the entire network. Hubs are considered Layer 1 devices because they only regenerate the signal and broadcast it out all of their ports (network connections).

**2.5.3- Bridges**

A bridge is a Layer 2 device designed to connect two LAN segments. The purpose of a bridge is to filter traffic on a LAN, to keep local traffic local, yet allow connectivity to other parts (segments) of the LAN for traffic that has been directed there. You may wonder, then, how the bridge knows which traffic is local and which is not. The answer is the same one that the postal service uses when asked how it knows which mail is local. It looks at the local address. Every networking device has a unique MAC address on the NIC, the bridge keeps track of which MAC addresses are on each side of the bridge and makes its decisions based on this MAC address list.




**2.5.4 Switches**

A switch is a Layer 2 device just as a bridge is. In fact a switch is called a multiport bridge, just like a hub is called a multi-port repeater. The difference between the hub and switch is that switches make decisions based on MAC addresses and hubs don't make decisions at all. Because of the decisions that switches make, they make a LAN much more efficient. They do this by "switching" data only out the port to which the proper host is connected. In contrast, a hub will send the data out all of its ports so that all of the hosts have to see and process (accept or reject) all of the data

**2.5.5- Routers**

A router is a Layer 3 device. Working at Layer 3 allows the router to make decisions based on groups of network addresses as opposed to individual Layer 2 MAC addresses. Routers can also connect different Layer 2 technologies, such as Ethernet, Token-ring, and FDDI. However, because of their ability to route packets based on Layer 3 information, routers have become the backbone of the Internet, running the IP protocol. The purpose of a router is to examine incoming packets (Layer 3 data), choose the best path for them through the network, and then switch them to the proper outgoing port. A router is a type of internetworking device that passes data packets between networks, based on Layer 3 addresses. A router has the ability to make intelligent decisions regarding the best path for delivery of data on the network.

**2.5.6- Gateway**

Gateways are multi-purpose connection devices. They are able to convert the format of data in one computing environment to a format that is usable in another computer environment (for example, AppleTalk and DECnet).

Gateways are devices that link different network types and protocols. For example, gateways translate different electronic mail protocols and convey email across the Internet. Gateways can operate at all layers of the OSI model. Gateways are available as stand-alone devices or in the form of a network station functioning as a gateway server. In both cases, the gateway requires appropriate network adapters LAN, WAN or both—and appropriate software. It is the software which is responsible for translating the messages from the received format to the format understood by the destination system.