



Computer Network Fundamentals

Lecture 1:

Introduction to Computer Networks

**lecturer : Dr. Aladdin Abbas
Alsharifi**

Network Definition

- **Network:** can be defined as two or more computers connected together in such a way that they can share information and resources.
 - The purpose of a network is to share information and resources.
 - The resources may be:
data, file, folder, printer, an Internet connection, applications, or anything else that exists on a computer



Network Definition

- **Data Network:** is a network that allows computers to exchange data.
 - The simplest data network is two PCs connected through a cable.
 - Most data networks connect many devices.



Network Definition

- **Computer Network** can be defined as a collection of devices that can store and manipulate electronic data, interconnected in such a way that network users can store, retrieve, and share information.
- **Internetwork** is a collection of individual networks connected by networking devices and function as a single large network.
 - The public Internet is the most common example which it is a single network that connects millions of computers.



Advantages of networking

- 1. Connectivity and Communication**
- 2. Data Sharing**
- 3. Hardware Sharing**
- 4. Internet Access**
- 5. Internet Access Sharing**
- 6. Data Security and Management**

Network Characteristics

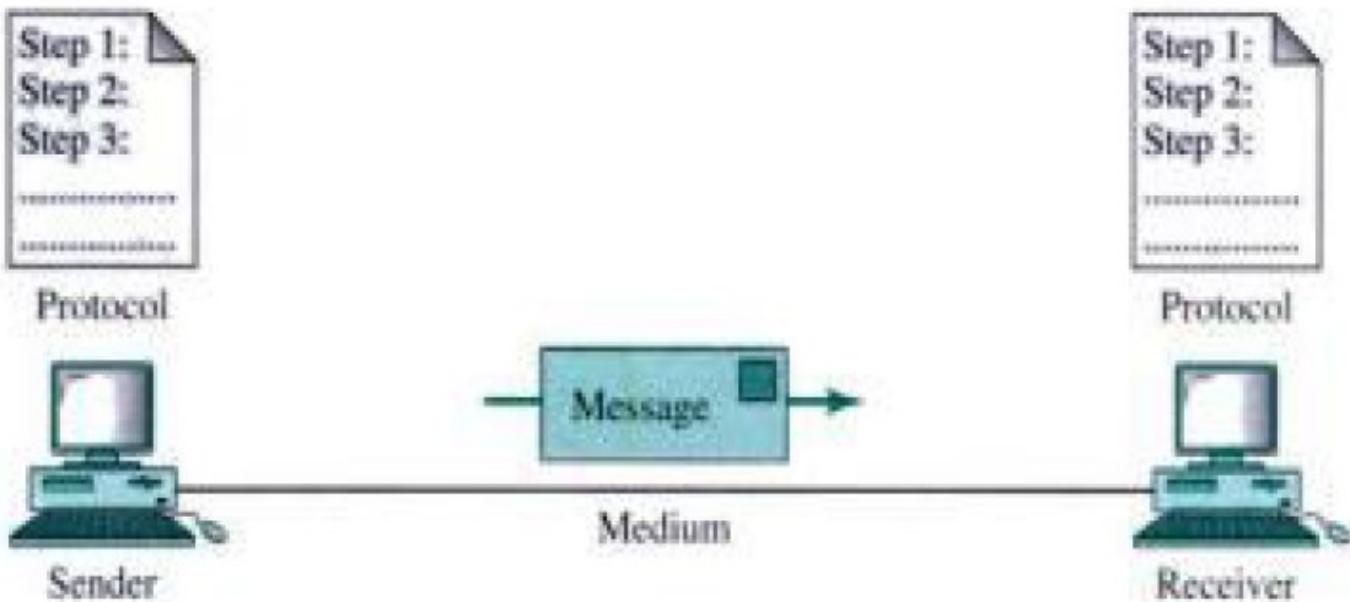
- **Delivery:** The system must deliver data to the correct destination. Data must be received by the intended device or user and only by that device or user.
- **Accuracy:** The system must deliver the data accurately. Data that have been altered in transmission and left uncorrected are unusable.
- **Timeliness:** The system must deliver data in a timely manner. Data delivered late are useless. In the case of video and audio, timely delivery means delivering data as they are produced, in the same order that they are produced, and without significant delay. This kind of delivery is called real-time transmission.

Elements of a Network

- **All networks have the following four basic elements:**
 - **Rules or agreements:** Rules or agreements (protocols) govern how the messages are sent, directed, received, and interpreted.
 - **Messages:** The messages or units of information travel from one device to another.
 - **Medium:** A medium is a means of interconnecting these devices, that is, a medium can transport the messages from one device to another.
 - **Devices:** Devices on the network exchange messages with each other.

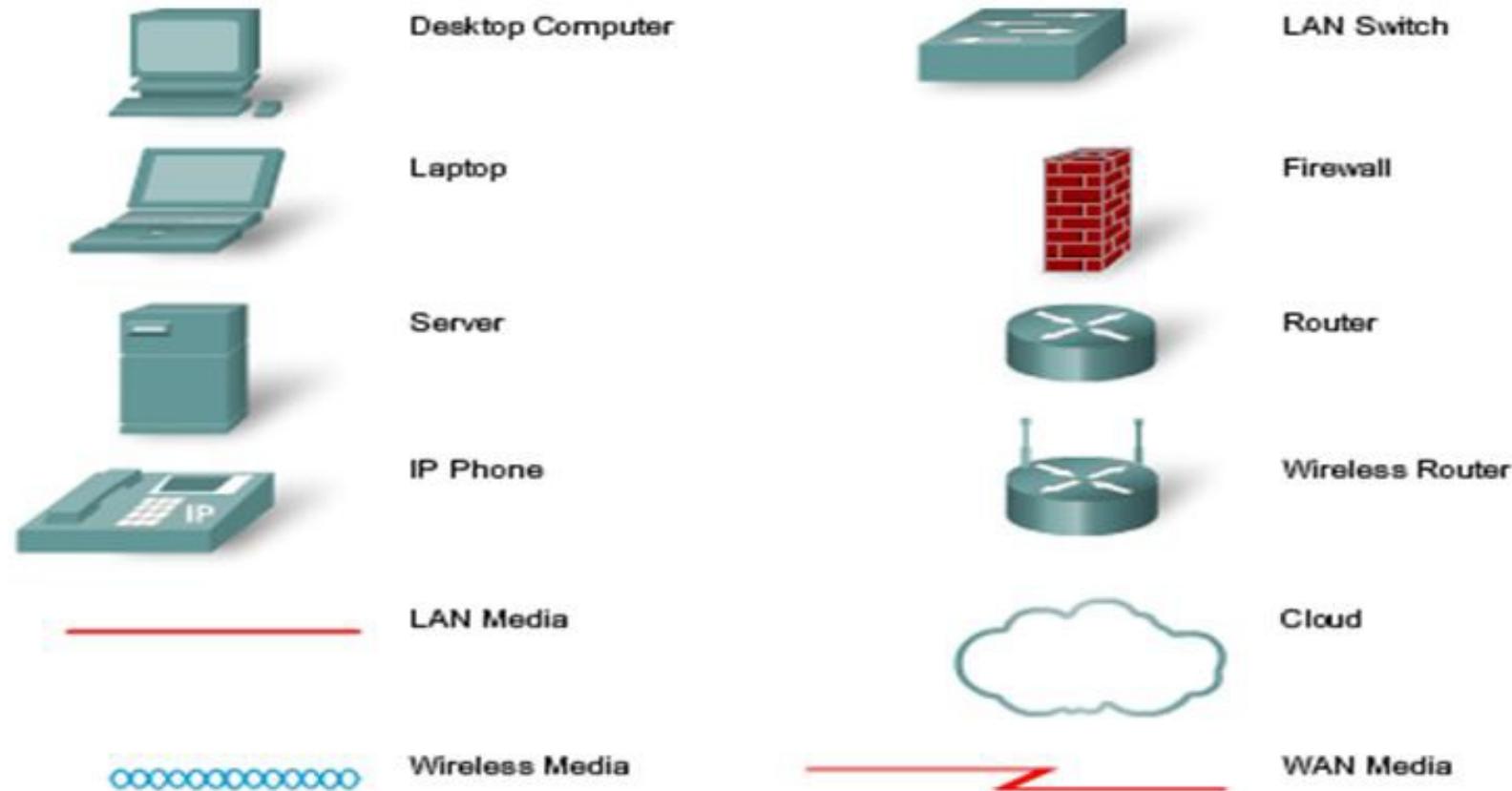
Figure 1-1 depicts a small network featuring rules, messages, a medium, and two devices.

Elements of a Network



Elements of a Network

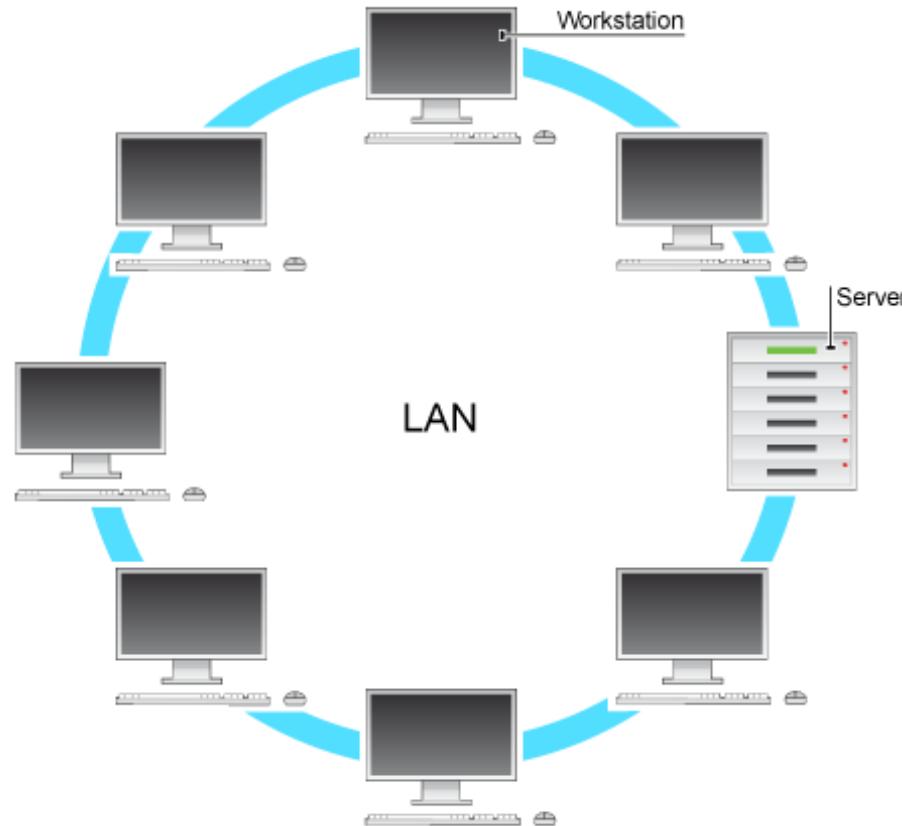
Common Data Network Symbols



Local Area Networks (LANs):

- **(LAN)** is a computer network that covers a **small geographic area**, like a home, office, or group of buildings, to exchange files and messages and to access shared resources such as printers and disk storage.
 - The simplest LAN consist of two computers connecting through a cable in a home office.

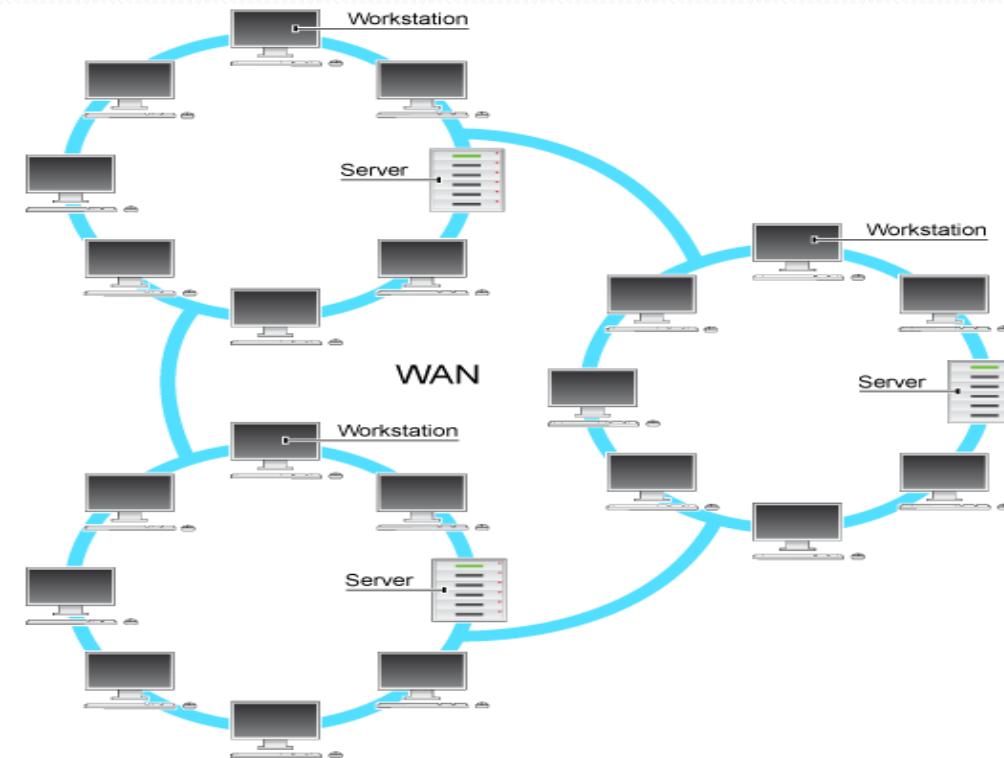
Network Classifications (LAN)



Network Classifications(WAN)

Wide Area Networks (WANs)

- WAN is a computer network that covers a **large geographic area** (i.e., any network whose communications links cross metropolitan, regional, or national boundaries). network that uses routers and public communications links.



Network Classifications(WAN)

The largest and most well-known example of a WAN is the **Internet**.

-WANs are used to connect LANs and other types of networks together, so that users and computers in one location can communicate with users and computers in other locations

Network Classifications(MAN)

Metropolitan Area Network (MAN):

- (MAN) is a network that interconnects users with computer resources in a geographic area or region larger than that covered by even a large local area network (LAN) but smaller than the area covered by a wide area network (WAN). The term is applied to the interconnection of networks in **a city** into a single larger network (which may then also offer efficient connection to a wide area network).

Data Flow

- **Communication** between two devices can be *Simplex, Half-Duplex, or Full-Duplex:*

- ❖ **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. For example, **Keyboards**.

- ❖ **Half -Duplex**

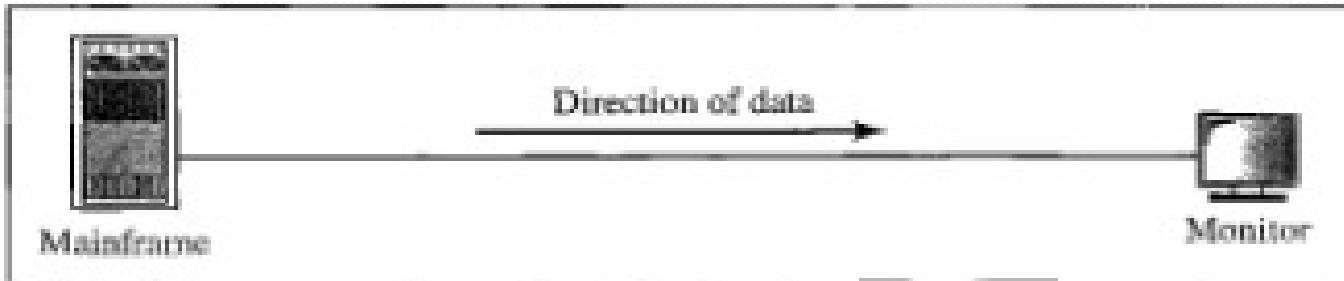
With half-duplex, communications happen in both directions, but in only one direction at a time. When two computers communicate using half-duplex, one computer sends a signal and the other receives; then, at some point, they switch sending and receiving roles. For example, push-to-talk technology (**walkie-talkie**).

Data Flow (Cont.)

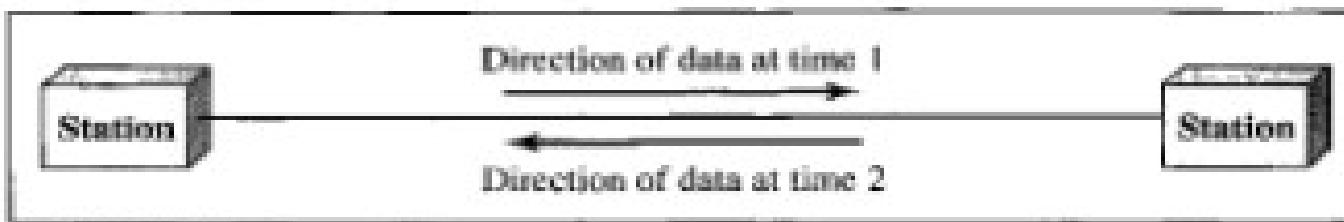
❖ Full-duplex

Full-duplex allows communication in both directions simultaneously. Both stations can send and receive signals at the same time. Full-duplex communications are similar to a **telephone call**, in which both people can talk simultaneously.

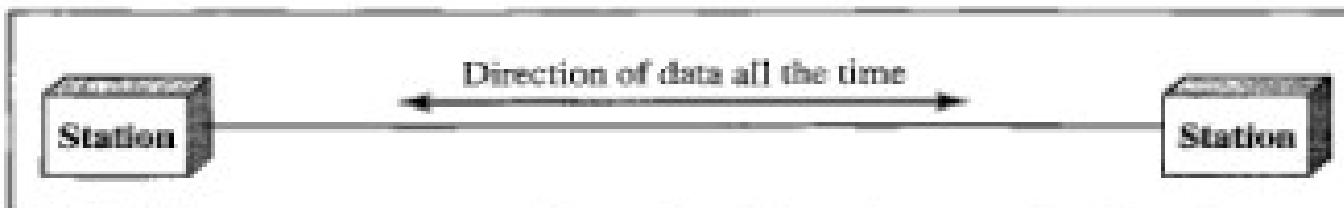
Data Flow



a. Simplex



b. Half-duplex



c. Full-duplex

Server, Workstation, and Client role in networking

● Server

A core component of the network, It **provides resources** to the clients on the network (“serves” them, in other words). Servers are typically powerful computers that run the software that controls and maintains the network. This software is known as the network operating system.

- A server computer provides a link to the resources necessary to perform any task.
- for example **Print Server** (Controls and manages one or more printers for the network).

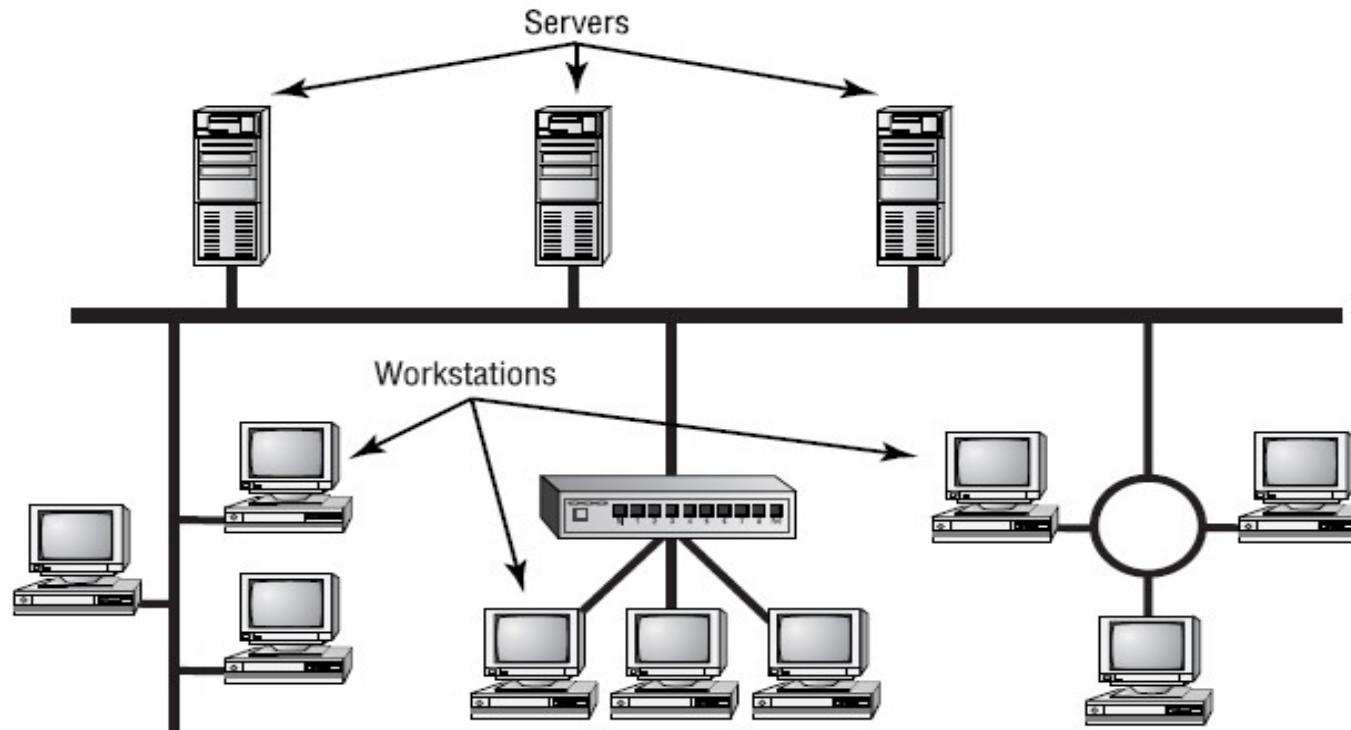
● Client

A *client* is any network entity that can **request resources** from the network. Client computers also depends primarily on the central server for processing activities.

Server, Workstation, and Client role in networking (Cont.)

- ***workstation***

Normally refers to any computer that is connected to the network and used by an individual to do work.



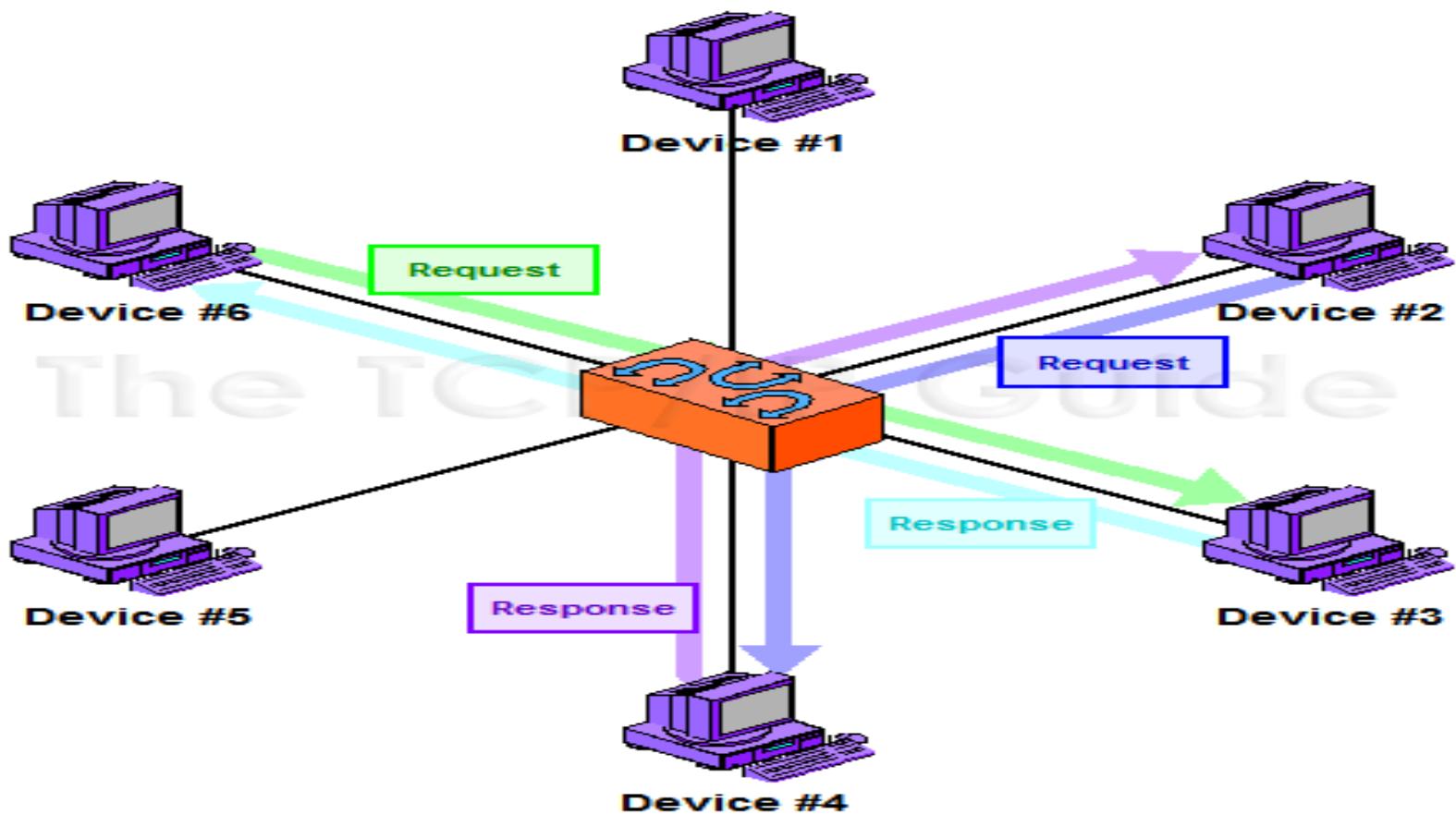
Network Architecture

- As discussed previously, the purpose of networking is to share resources, but we don't know how this is accomplished?
 - This depends on the **architecture of the network operating system software**. The two most common network types are **peer-to-peer** and **client/server**.

➤ Peer-to-Peer Architecture

In peer-to-peer networks, the connected computers have no centralized authority. From an authority viewpoint, all of these computers are equal. Each computer in a peer-to-peer network can ***be both a client that requests resources and a server that provides resources.*** There is no assigned role for any particular device, and each of the devices usually runs similar software.

Peer-to-Peer Architecture

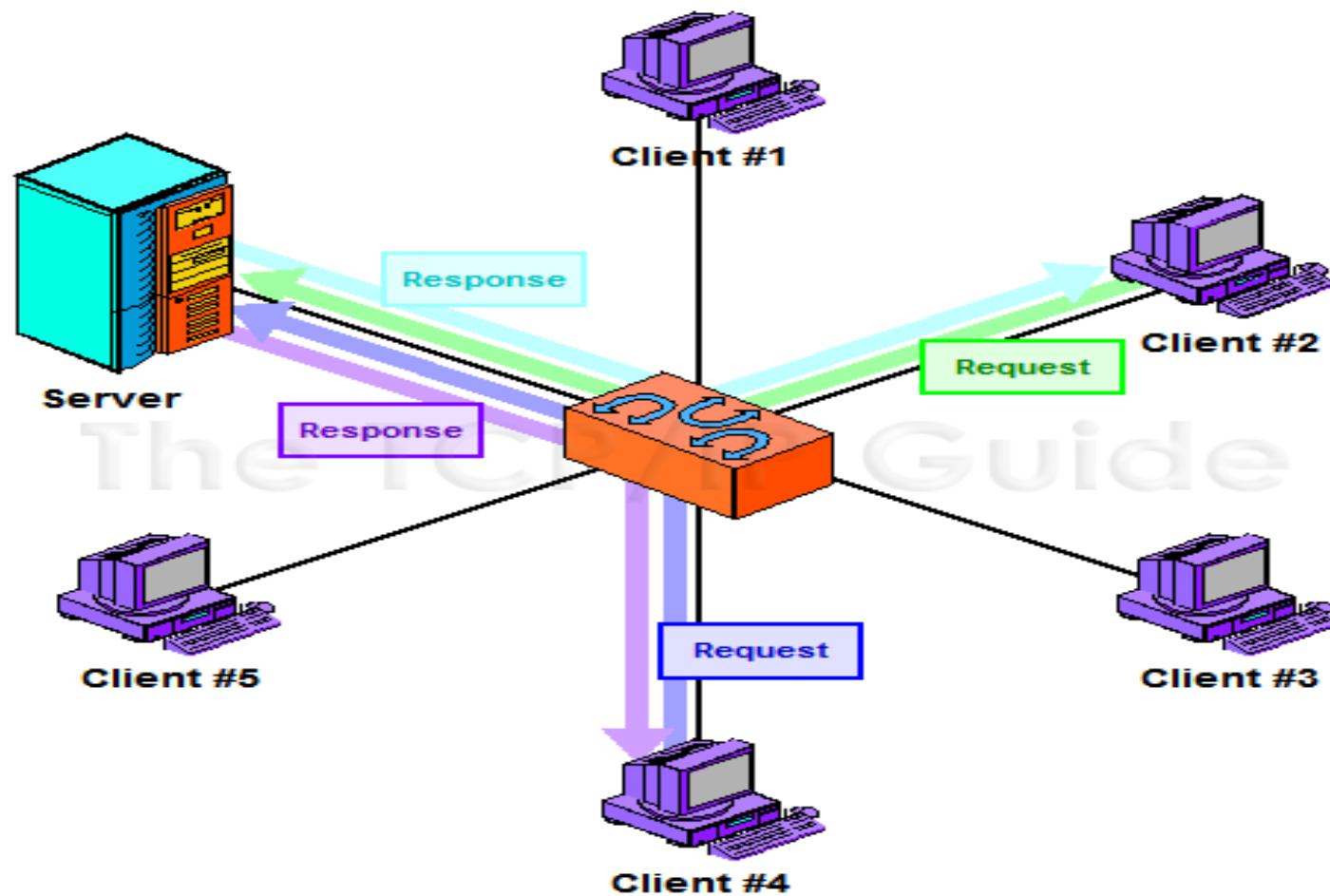


Client/Server Architecture

➤ Client/Server Architecture

In this design, a client/server network uses a network operating system designed to manage the entire network from a **centralized point**, which is **the server**. **Clients** make requests of the server, and the server responds with the information or access to a resource.

Client/Server Architecture(Cont.)



Network Topology

- **A topology** is basically a map of a network. The physical topology of a network describes the layout of the cables and workstations and the location of all network components. Topologies can be either physical or logical.

--*Physical topologies*

The term physical topology 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.

--*Logical topologies* describe how the network messages travel.

- The cables or connections in a physical topology are often referred to as **network media** (or **physical media**).

Network Topology

- There are four famous type of topology:
 - **Bus** (can be both logical and physical)
 - **Star** (physical only)
 - **Ring** (can be both logical and physical)
 - **Mesh** (can be both logical and physical)

Network Topology (BUS)

● Bus Topology

A bus is the simplest physical topology. It consists of a **single cable** that runs to every workstation.

**This topology uses the least amount of cabling, but also covers the shortest amount of distance.

**When communicating on a network that uses a bus topology, all computers see the data on the wire.

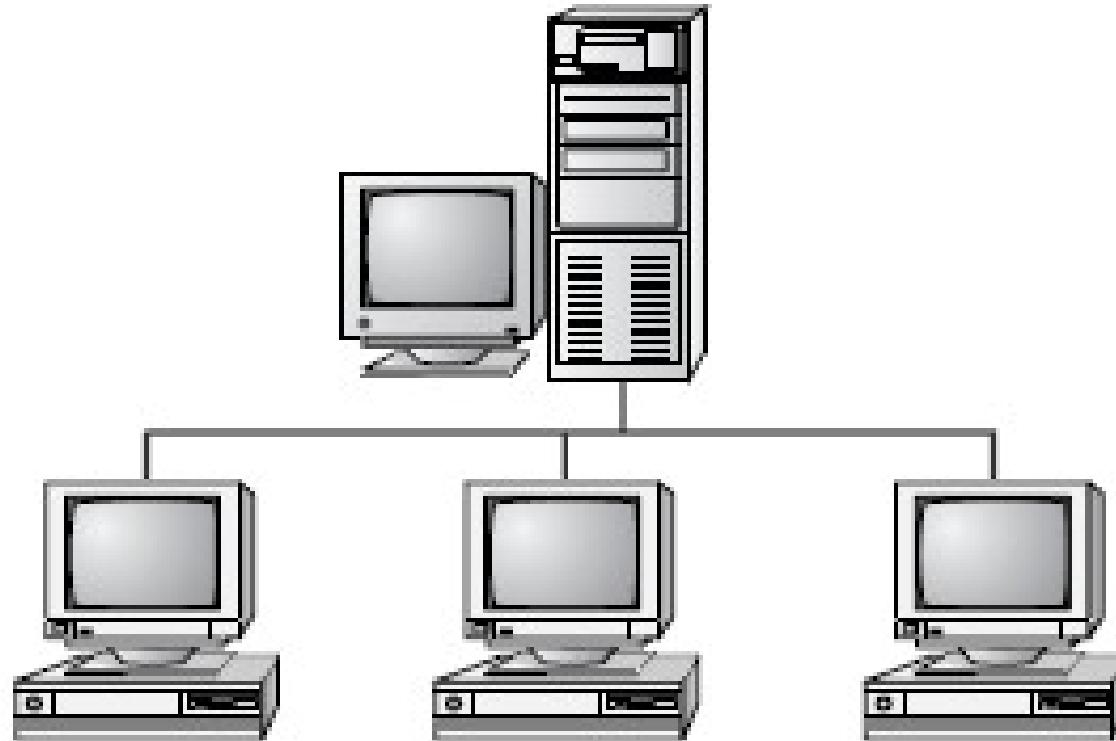
**With a logical bus topology, messages pass through the trunk, and each workstation checks to see if the message is addressed to itself. If the address of the message matches the workstation's address, the network adapter copies the message to the card's on-board memory.

Network Topology (BUS)

- **A bus topology has the following characteristics:**
 1. Is simple to install.
 2. Is relatively inexpensive.
 3. Uses less cable than other topologies.

- **The following characteristics describe the downside of a bus topology:**
 1. Is difficult to move and change.
 2. Has little fault tolerance (a single fault can bring down the entire network).
 3. Is difficult to troubleshoot.

Network Topology (BUS)



Network Topology (Star)

• **Star Topology**

A physical star topology branches each network device off a central device called a *hub*, making it very easy to add a new workstation.

--Star topologies are easy to install. A cable is run from each workstation to the hub. The hub is placed in a central location in the office.

--Star topologies are more expensive to install than bus networks, because there are several more cables that need to be installed, plus the cost of the hubs that are needed.

Network Topology (Star)

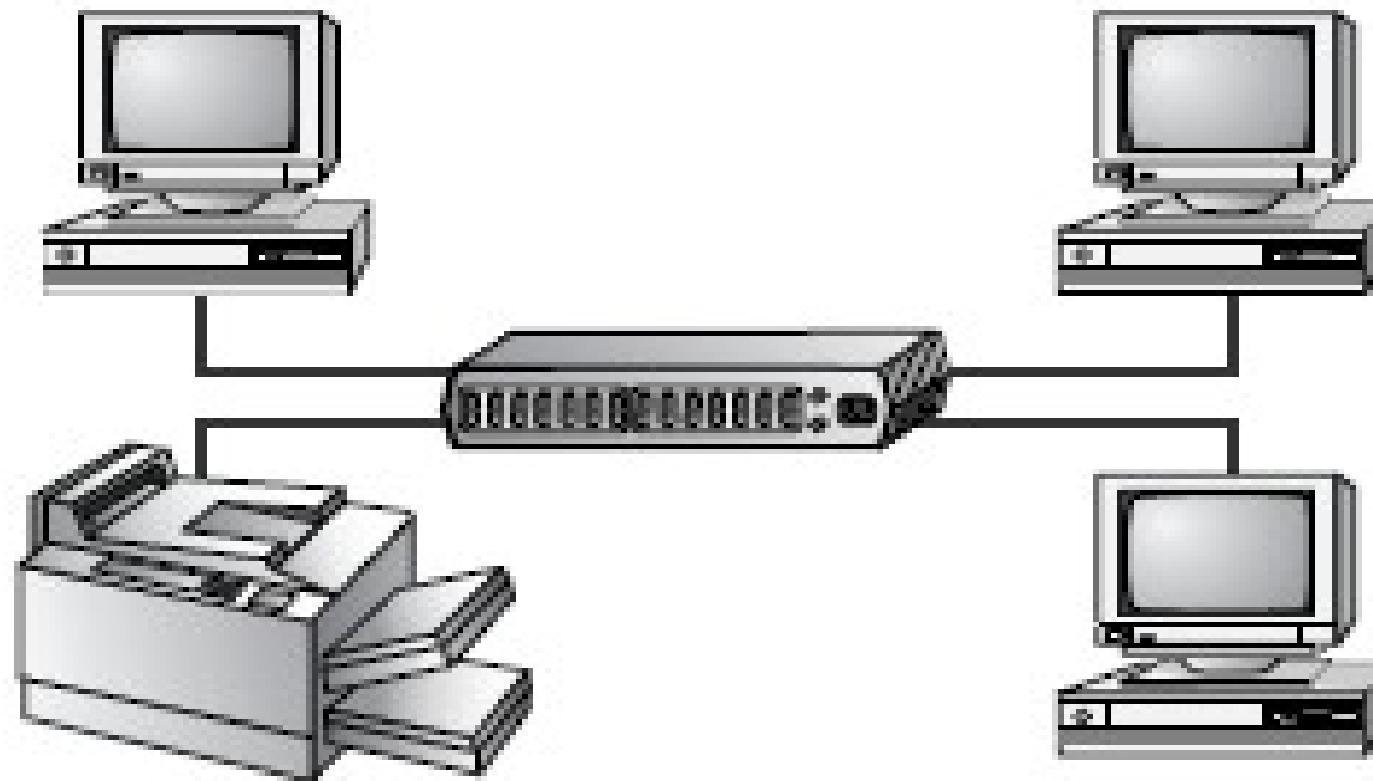
➤ The star topology has advantages:

1. New stations can be added easily and quickly.
2. A single cable failure won't bring down the entire network.
3. It is relatively easy to troubleshoot.

➤ The disadvantages of a star topology include the following:

1. Total installation cost can be higher because of the larger number of cables, but prices are constantly becoming more and more competitive.
2. It has a single point of failure (the hub, or other central device).

Network Topology (Star)



Network Topology (Ring)

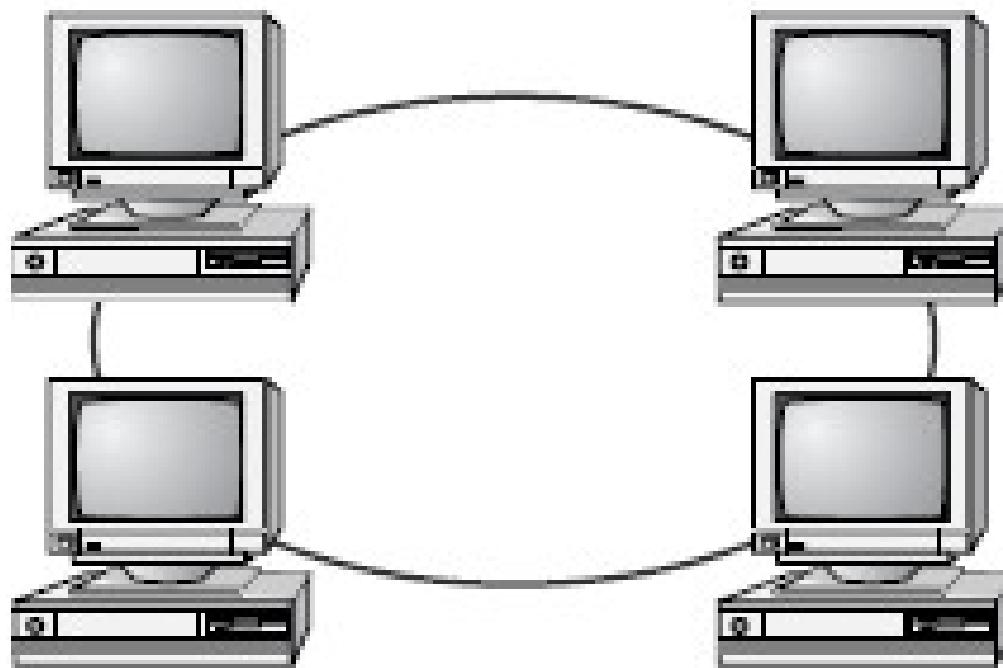
● Ring Topology

In this topology, computer is connected directly to two other computers in the network. Data moves down a one-way path from one computer to another. Each entity participating in the ring reads a message, then regenerates it and hands it to its neighbor on a different network cable.

➤ A ring topology has the following characteristics:

1. Expensive, because multiple cables are needed for each workstation.
2. The ring makes it difficult to add or remove new computers.
3. Difficult to reconfigure(the ring topology network will go down if one entity is removed from the ring.).
4. Not fault tolerant. A single cable fault can bring down the entire network.
5. The physical ring topology is seldom used.

Network Topology (Ring)



Network Topology(Mesh)

- The *mesh topology* is the simplest **logical topology** in terms of data flow, but it is the most complex in terms of physical design.

In this **physical topology**, each device is connected to every other device. This topology is rarely found in **LANs**, mainly because of the complexity of the cabling.

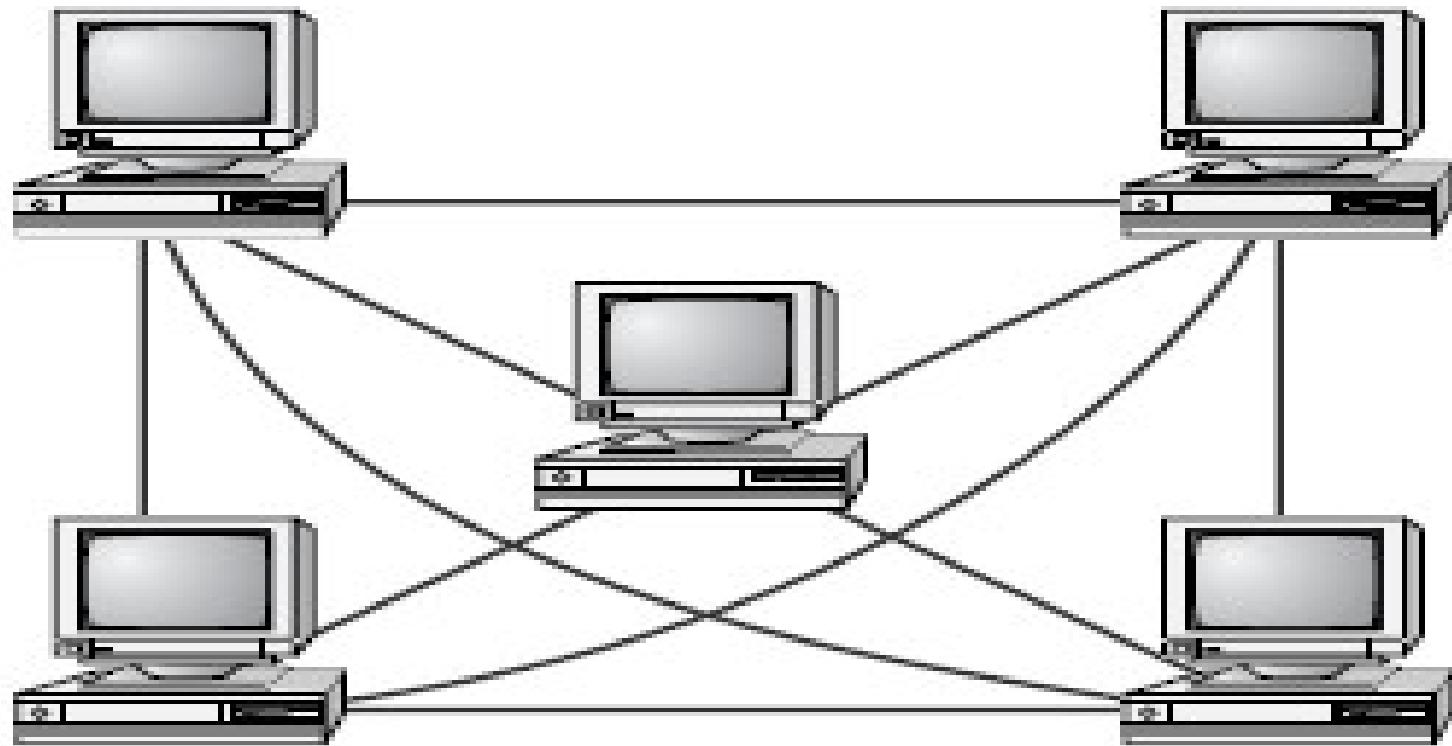
If there are n computers, there will be $(n * (n-1)) / 2$ cables in the network. For example, if you have five computers in a mesh network, it will use $5 * (5 - 1) / 2$, which equals 10 cables. This complexity is compounded when you add another workstation.

For example, your five-computer, 10-cable network will jump to 15 cables just by adding one more computer. Imagine how the person doing the cabling would feel if you told them you had to cable 50 computers in a mesh network —they'd have to come up with $50 \times (50 - 1) \div 2 = 1225$ cables!

Network Topology (Mesh)

- Because of its **design**, the **physical mesh topology** is very expensive to install and maintain.
- Cables must be run from each device to every other device. The advantage you gain from it is its **high fault tolerance**.
- With a logical mesh topology, however, there will always be a way of getting the data from source to destination. It may not be able to take the direct route, but it can take an alternate, indirect route. It is for this reason that the mesh topology is still found in **WANs** to connect multiple sites across WAN links. It uses devices called *routers* to search multiple routes through the mesh and determine the best path. However, the mesh topology does become inefficient with five or more entities.

Network Topology (Mesh)



Thank You

Introduction to Networks

Lecture 2:

Network Connectivity Devices

lecturer : Dr. Aladdin Abbas



تعديل من خلال WPS Office

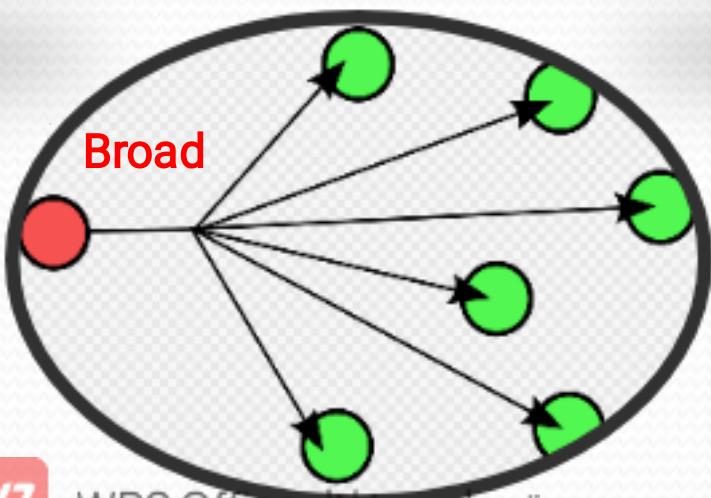
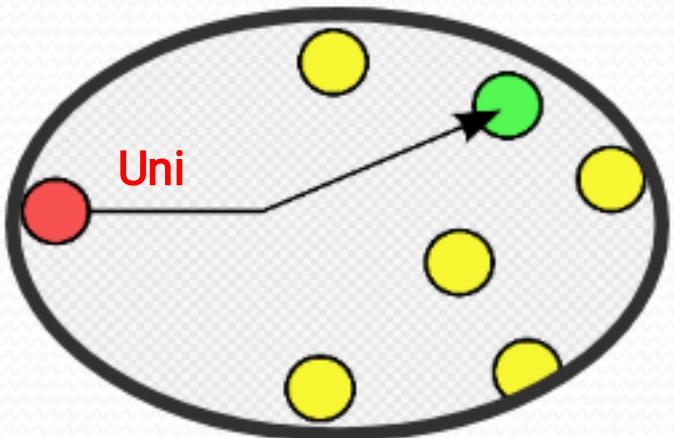
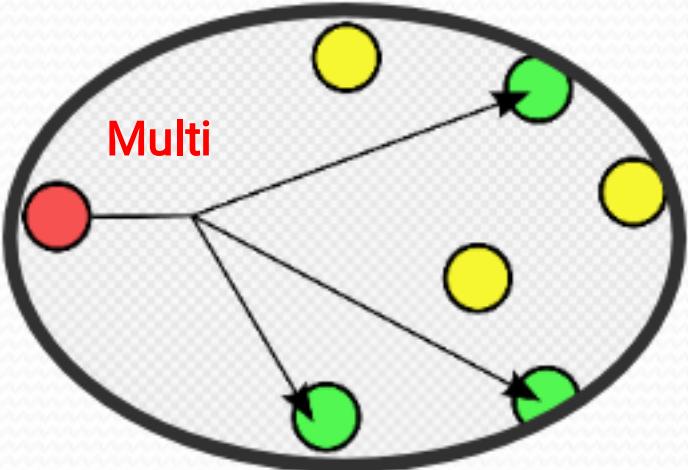
Network Devices- Terminology

- Some terminology related to the operation of network devices is:

- **Domain** : is a specific part of a network.
- **Bandwidth** : is the amount of data that can be carried across a network in a given time period.
- **Unicast data**: is data meant for a specific device.
- **Broadcast data** : is data meant for all devices; a special broadcast address indicates this.
- **Multicast data**: is data destined for a specific group of devices; a special address indicates this.
- **A bandwidth domain**, known as a **collision domain** for Ethernet LANs, includes all devices that share the same bandwidth.
- **A broadcast domain** includes all devices that receive each other's' broadcasts (and multicasts)



Network Devices



تعديل من حلول WPS Office

Network Devices

- Network devices are the devices that interconnect networks. Because these devices connect network entities, they are known as **connectivity devices**. These devices include:

- ❖ Hub
- ❖ Switch
- ❖ Router



تعديل من خلال WPS Office

Network Devices - Hub

HUB

A typical Ethernet LAN uses unshielded twisted-pair (UTP) cables with RJ-45 connectors.

- Because these cables have only two ends, an intermediary device is needed to connect more than two computers. That device is **a hub**. Hubs are devices used to link several computers together.
- A hub works at Layer 1 and connects **multiple devices** so that they are logically all on one LAN.
- **NOTE:** The physical connection point on a network device—a hub, switch, or router—is called an ***interface*** or a ***port***.

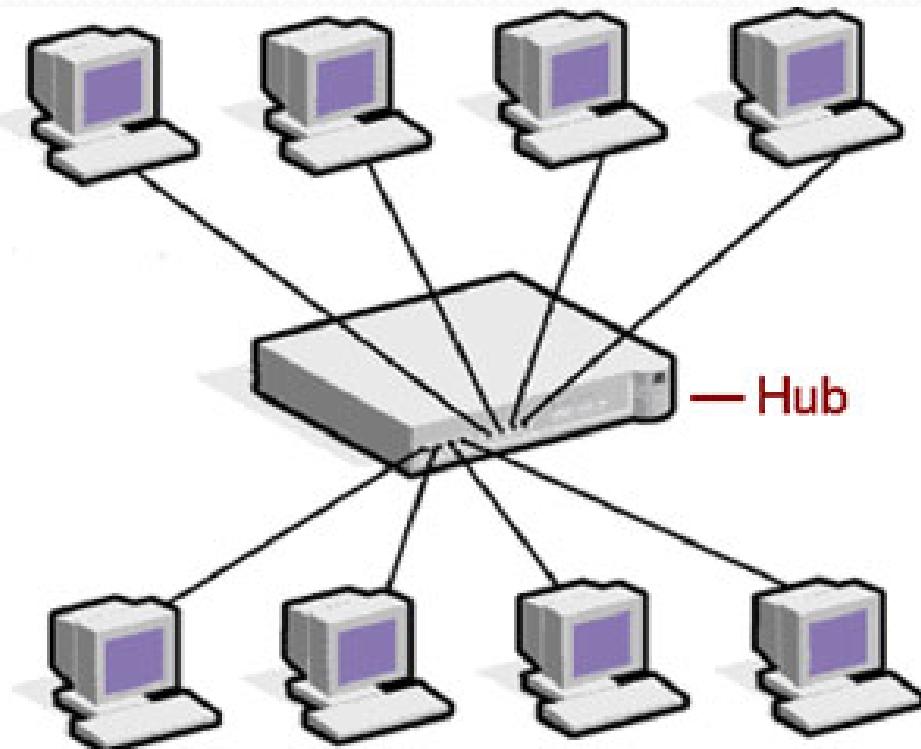


Network Devices - Hub

- A hub has no intelligence—it sends all data received on any port to all the other ports. So, devices connected through a hub receive everything that the other devices send, whether or not it was meant for them. This process called **broadcasting**).
- All devices connected to a hub are in one collision domain and one broadcast domain.
- Note: A hub just repeats all the data received on any port to all the other ports; thus, **hubs are also known as *repeaters***.



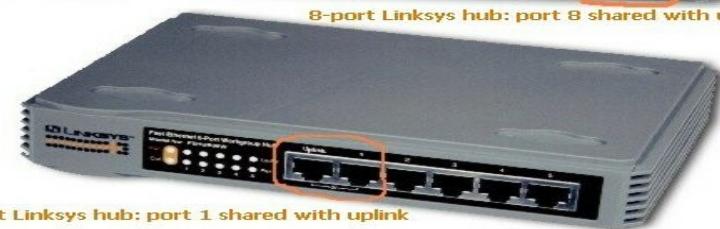
Network Devices - Hub



16-port Linksys hub: port 16 shared with uplink



8-port Linksys hub: port 8 shared with uplink



5-port Linksys hub: port 1 shared with uplink



8-port SMC Hub: port 8 shared with uplink



تعديل من خلال
WPS Office

Network Devices- Switch

- **LAN** switches are Layer 2 devices and have some intelligence—they send data to a port only if the data needs to go there.
- A device connected to a switch port does not receive any of the information addressed to devices on other ports. Therefore, the main advantage of using a switch instead of a hub is that the traffic received by a device is reduced because only frames addressed to a specific device are forwarded to the port on which that device is connected.



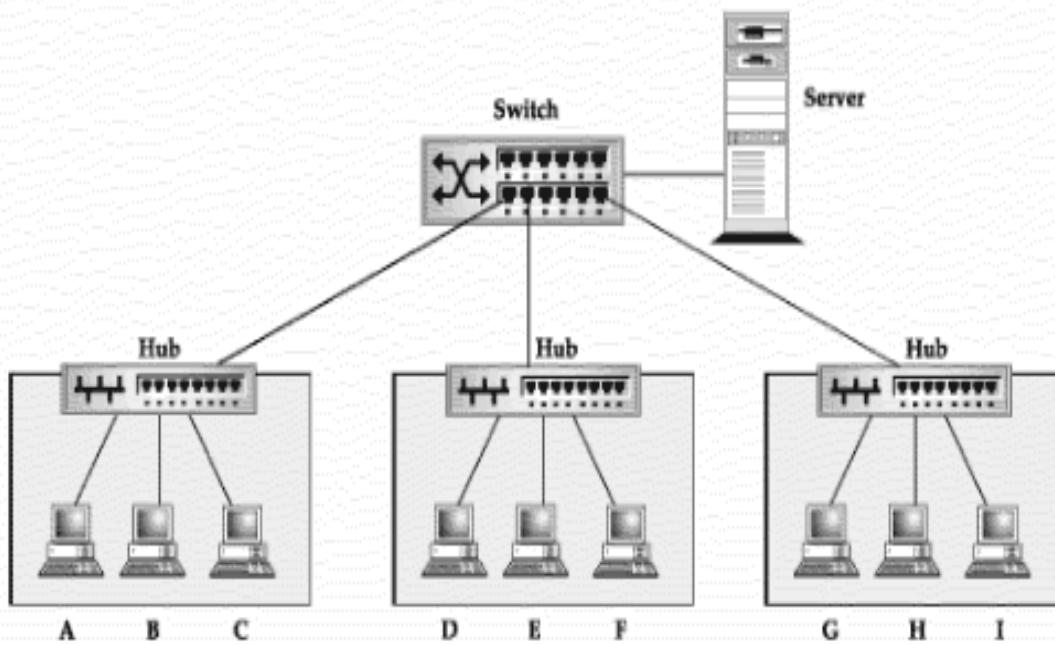
Network Devices- Switch

Switches read the source and destination MAC addresses in the frames and therefore can keep track of who is where, and who is talking to whom, and send data only where it needs to go.

- If the switch receives a frame whose destination address indicates that it is a broadcast (information meant for everyone) or multicast (information meant for a group), by default it sends the frame out all ports (except for the one on which it was received).
- All devices connected to one switch port are in the same collision domain, but devices connected to different ports are in different collision domains. By default, all devices connected to a switch are in the same broadcast domain.



Network Devices- Switch



تعديل من خلال WPS Office

Network Devices - Router

- A *router* goes one step further than a switch. It is a Layer 3 device that has much more intelligence than a hub or switch.
- By using logical Layer 3 addresses, routers allow devices on different LANs to communicate with each other and with distant devices—for example, those connected through the Internet or through a WAN.
- The logical Layer 3 addresses is the TCP/IP's IP addresses..
- The router reads the source and destination logical addresses in the packets and therefore keeps track of who is where, and who is talking to whom, and sends data only where it needs to go.



Network Devices - Router

- All devices connected to one router port are in the same collision domain, but devices connected to different ports are in different collision domains.
- Routers block broadcasts (destined for *all* networks) and multicasts by default; routers forward only *unicast* packets (destined for a specific device) and packets of a special type called *directed broadcasts*.



تعديل من خلال WPS Office

Network Devices- Router

- Routers work at the OSI model **network layer**. The main functions of a router are first to determine the best path that each packet should take to get to its destination and second to send the packet on its way.
- Sending the packet out the appropriate interface, along the best path, is also called ***switching the packet*** because the packet is encapsulated in a new frame, with the appropriate framing information.
- Routers are normally used to connect one LAN to another. Typically, when a WAN is set up, there will be at least two routers used.

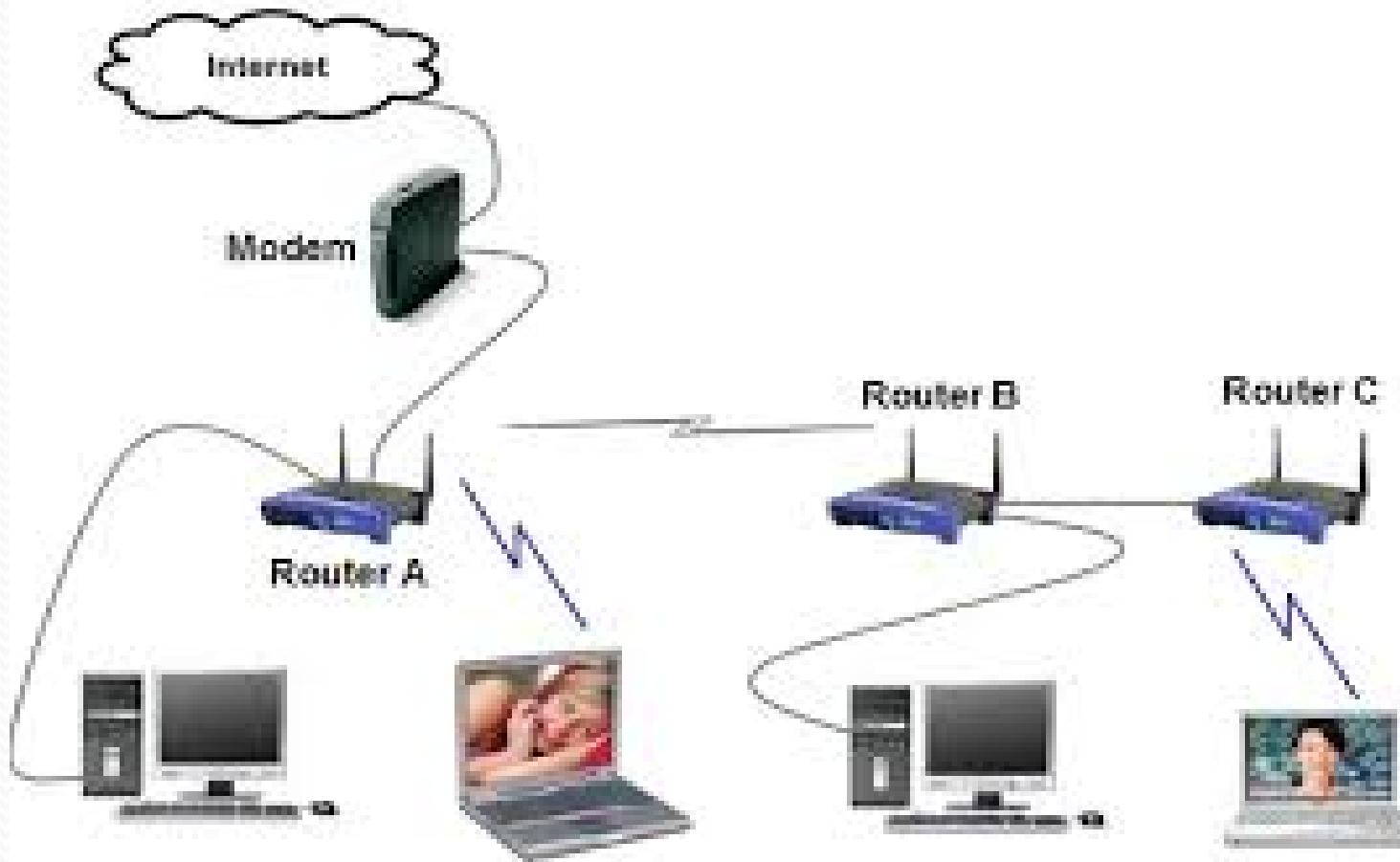


Network Devices- Router



تعديل من خلال WPS Office

Network Devices- Router



تعديل من خلال WPS Office

Thank You



تعديل من خلال WPS Office

Introduction to Computer Networks

Lecture 3:

OSI Model

Asst Prof. : Dr.Suad Abdulelah Alasadi

OSI Model

- What is OSI model???

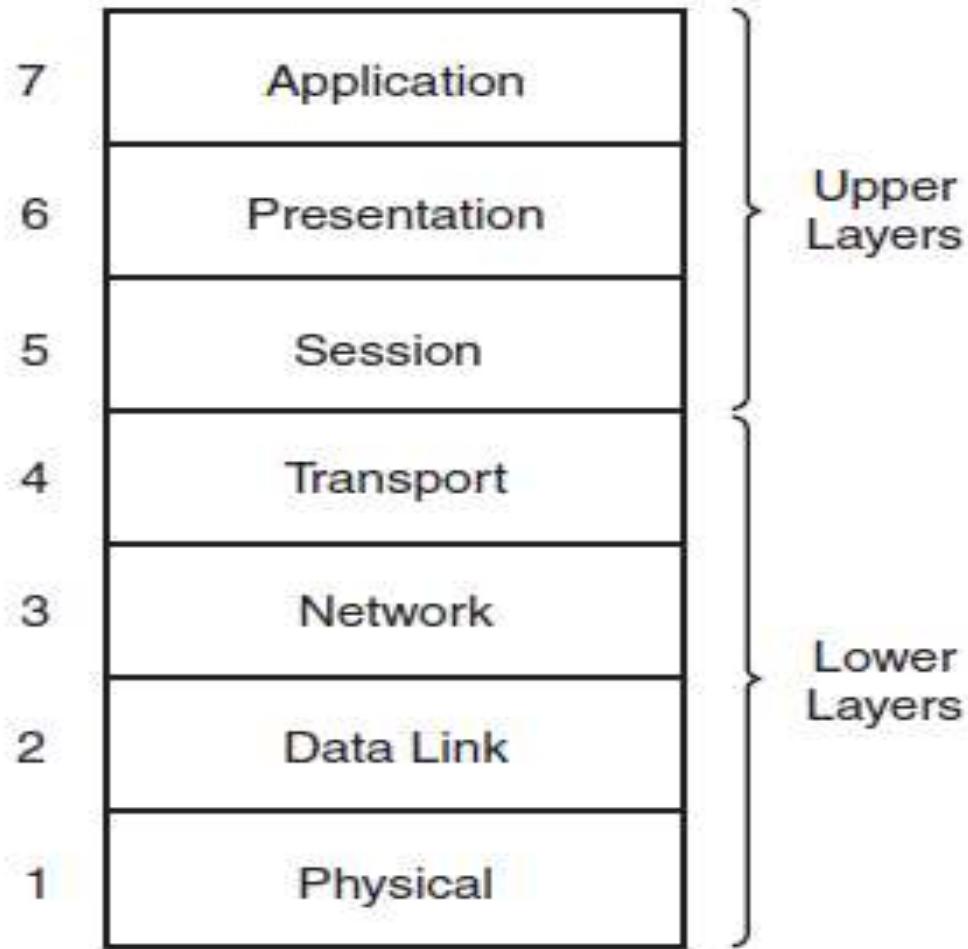
To understand what is the OSI model and why it use, first we will take the following example:

- imagine that you are in Baghdad and you want to send an e-mail to your friend in Lebanon . Successfully sending and receiving e-mail involves doing many things, including the following:
 - You must type the message in your e-mail application.
 - You must address the message in your e-mail application.
 - You must click the **Send** button in your e-mail application to start sending the message.
 - You must use the correct type of connections and wires to connect your PC to your local network.
 - Your PC must put the data on the wire.
 - Your PC must be able to connect to the Internet, and you must provide any necessary login information.
 - Network devices must find the best path through the Internet so that the e-mail is received by the right person.

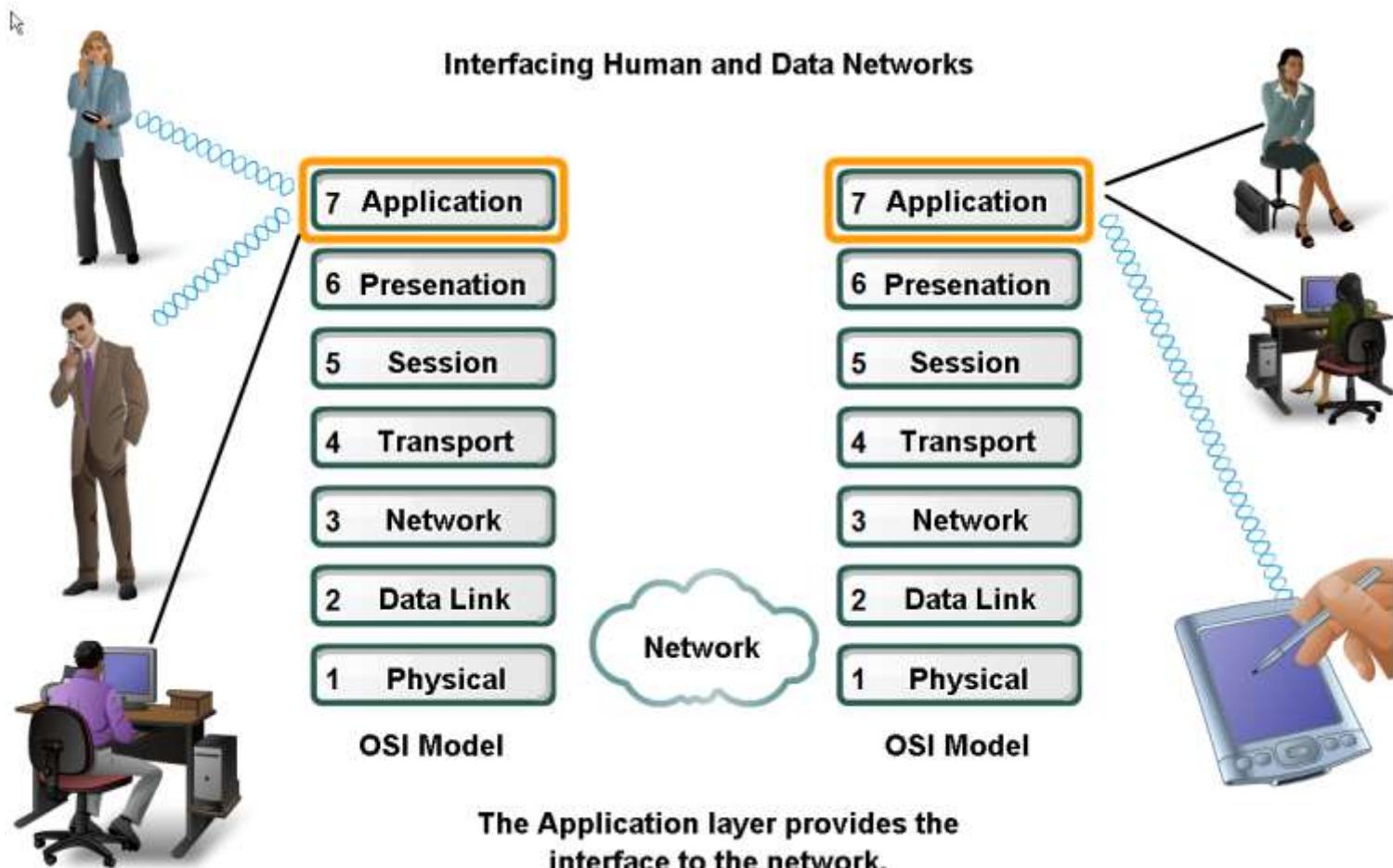
OSI Model

- OSI is stand for (**Open System Interconnection**) Reference Model.
- The International Standards Organizations (ISO) committee created a list of all the network functions required for sending data and divided them into seven categories. This model is the *OSI seven layer* model. The OSI seven-layer model was released in 1984.
- *Each of the Seven Layers of the OSI Model Represents Functions Required for Communication.*

OSI Model



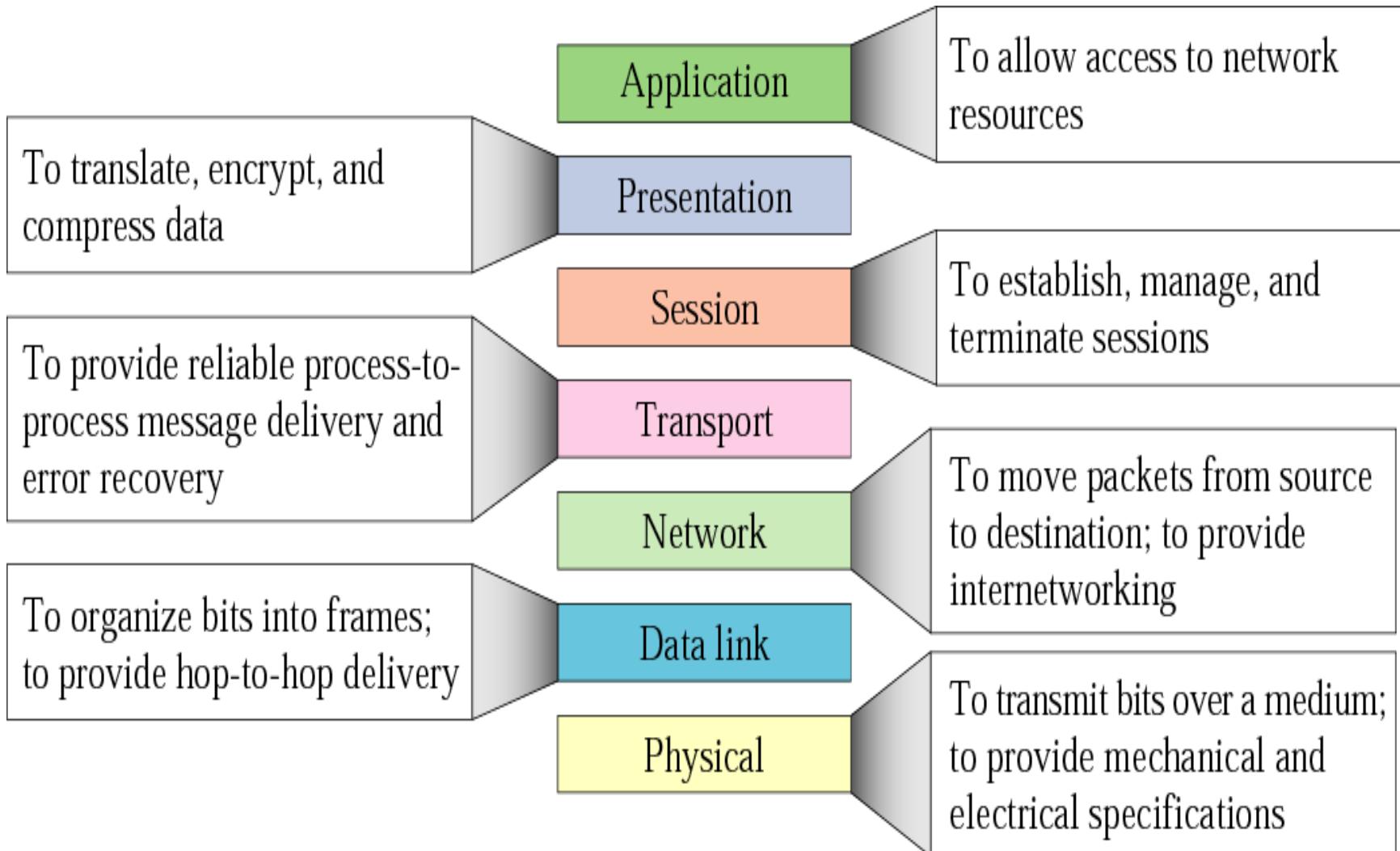
OSI Model



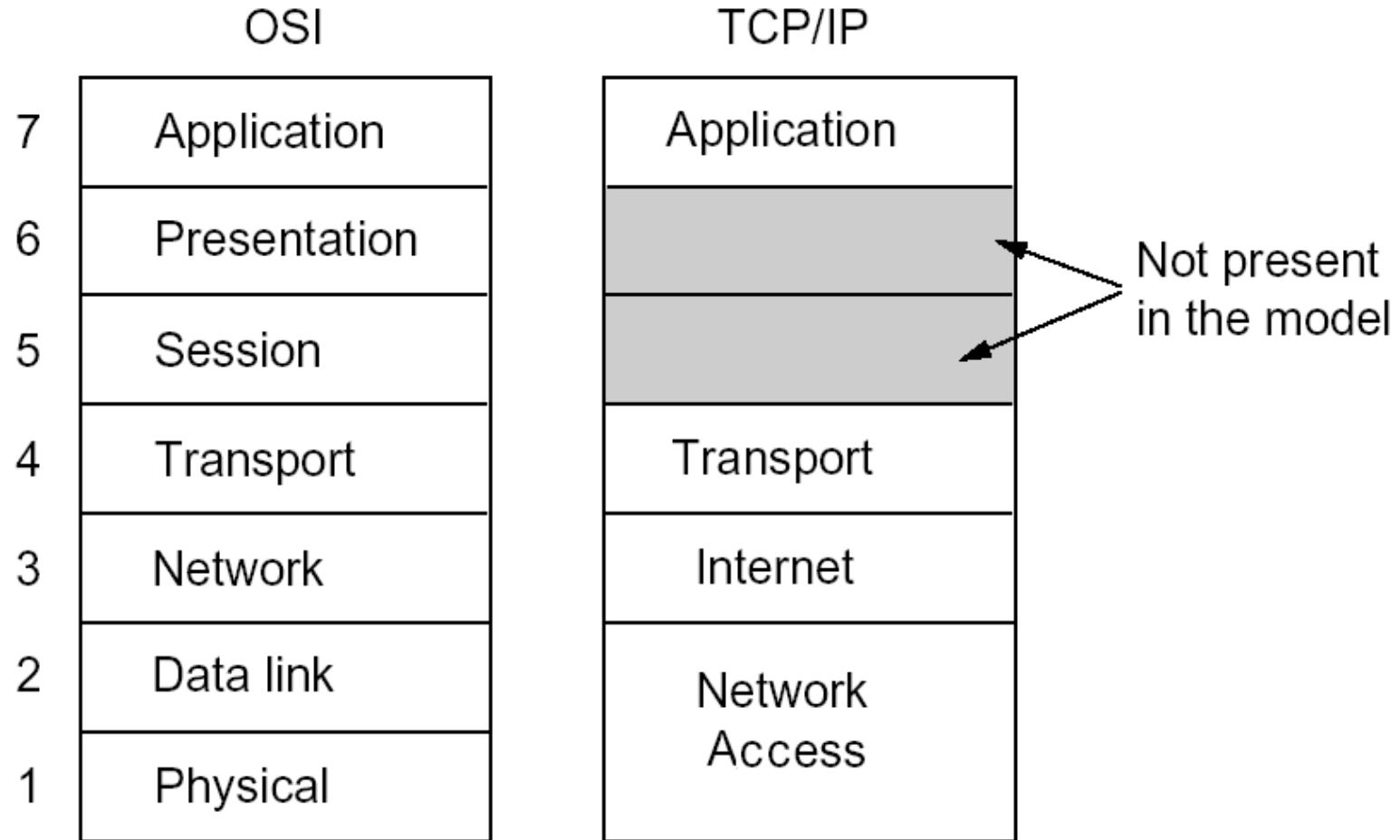
OSI Model

- The OSI model **represents everything** that must happen to send data. The important thing to remember is that the OSI model **does not specify *how* these things are to be done**, just **what needs to be done**. Different protocols can implement these functions differently. For example, the open-standard Internet Protocol (IP) and Novell's Internetwork Packet Exchange (IPX) protocol are different implementations of the network layer.

OSI Model

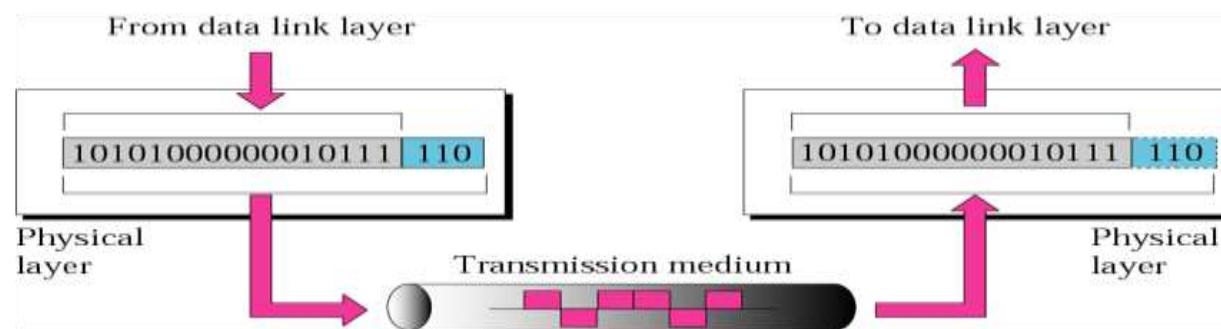


OSI and TCP/IP



Physical Layer

- Layer 1, Physical layer defines **specifications** such as the electrical and mechanical conditions necessary for activating, maintaining, and deactivating the physical link between devices.
- **Specifications include voltage levels, maximum cable lengths, connector types, and maximum data rates.**
- The physical layer is concerned with the binary transmission of data. This binary data is represented as **bits** (which is short for *binary digits*). A bit has a single binary value, either 0 or 1.



Data Link Layer

- Layer 2, defines the format of data that is to be transmitted across the physical network. It indicates how the physical medium is accessed, including physical addressing, error handling, and flow control.
- The data link layer sends **frames** of data.
- A *frame* is a defined set of data that includes **addressing** and **control information** and is transmitted between network devices. A frame can contain a **header field** (in front of the data) and **a trailer field** (after the data); these two fields are said to “frame” the data.
- Has two sub-layers:
 - Logical Link Control (LLC):
 - ✓ Allows multiple network layer protocols to communicate .
 - ✓ Monitoring and controlling the connection.
 - Media Access Control (MAC): uniquely physical address identify device on network.
- The **MAC** sublayer specifies the *physical MAC address that uniquely identifies a device on a network*. Each frame that is sent specifies a destination MAC address; only the device with that MAC address should receive and process the frame. Each frame also includes the MAC address of the frame’s source.

Network Layer

- Layer 3, Network layer is responsible for routing, which allows data to be properly forwarded across a logical internetwork (consisting of multiple physical networks).
- The Network layer sends datagram (**Packets**) of data
 - **Internet Protocol (IP) addresses** (*Logical* network addresses) as opposed to physical MAC addresses are specified at Layer 2.
 - Layer 3 protocols include routed and routing protocols. The routing protocols determine the best path that should be used to forward the routed data through the internetwork to its destination.

Network Layer (Cont.)

- A *datagram* is a defined set of data that includes addressing and control information and is routed between the data's source and destination.
- If a datagram needs to be sent across a network that can handle only a certain amount of data at a time, the datagram can be fragmented into multiple packets and then reassembled at the destination. Therefore, a *datagram* is a unit of data, whereas a *packet* is what physically goes on the network. If no fragmentation is required, a packet is a datagram; the two terms are often used interchangeably.

Transport Layer

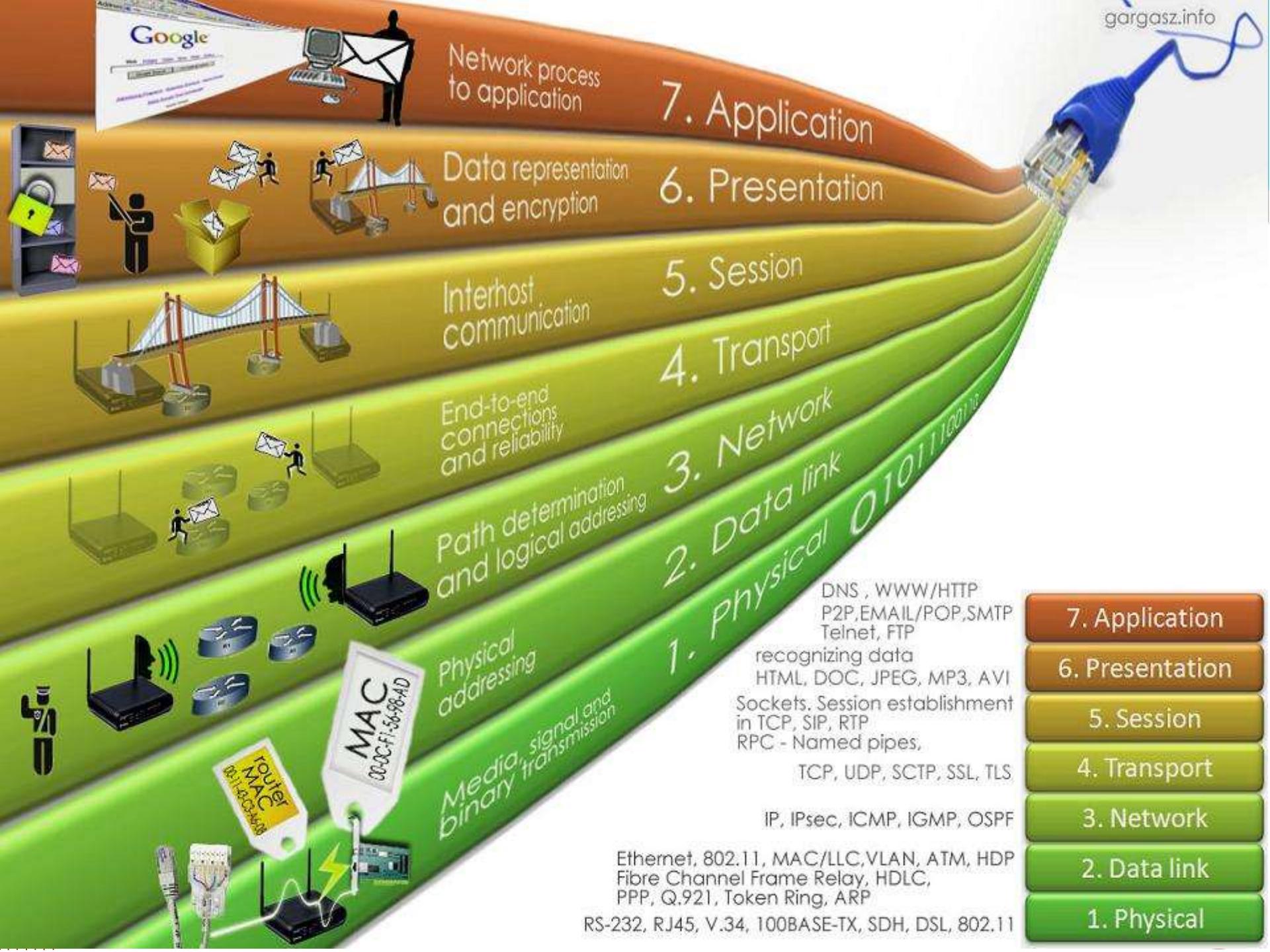
- Layer 4, the transport layer, is concerned with **end-to-end connections between the source and the destination.**
- The transport layer provides network **services to the upper layers.**
- The Transport layer sends **Segments** of data
- A *segment* is a defined set of data that includes **control information** and is sent between the transport layers of the sender and receiver of the data.

Transport Layer(Con.)

- Connection-oriented reliable transport.
- transport establishes a logical connection and uses sequence numbers to ensure that all data is received at the destination.
- Connectionless best-effort transport.
- transport just sends the data and relies on upper-layer error detection mechanisms to report and correct problems.
- Reliable transport has more overhead than best-effort transport

The upper layers

- The three upper layers represent the data that must be transmitted from the source to the destination; the network typically neither knows nor cares about the contents of these layers.
- **The Session layer, Layer 5**, is responsible for establishing, maintaining, and terminating communication sessions between applications running on different hosts.
- **The Presentation layer, Layer 6**, specifies the format, data structure, coding, compression, and other ways of representing the data to ensure that information sent from one host's application layer can be read by the destination host.
- **The Application layer, Layer 7**, is the closest to the end, it provides a means for the user to access information on the network through an application. This layer is the main interface for users to interact with the application and therefore the network.



Communication Among OSI Layers

- In OSI model terms, information is exchanged between peer OSI layers—the **application layer** on your computer is communicating with the application layer on your friend’s computer. However, to accomplish this, the e-mail must go **through all the other layers on you computer**.
- for example, it must have the **correct network layer address**, be put in the correct frame type, and so on.
- The e-mail must then go over the network, and then **go back through all the layers** on your friend’s computer, until it finally arrives at your friend’s e-mail application.
- **The direction of moving** in the sender device is from **top to down** while in the **destination device** is from **down to up**.

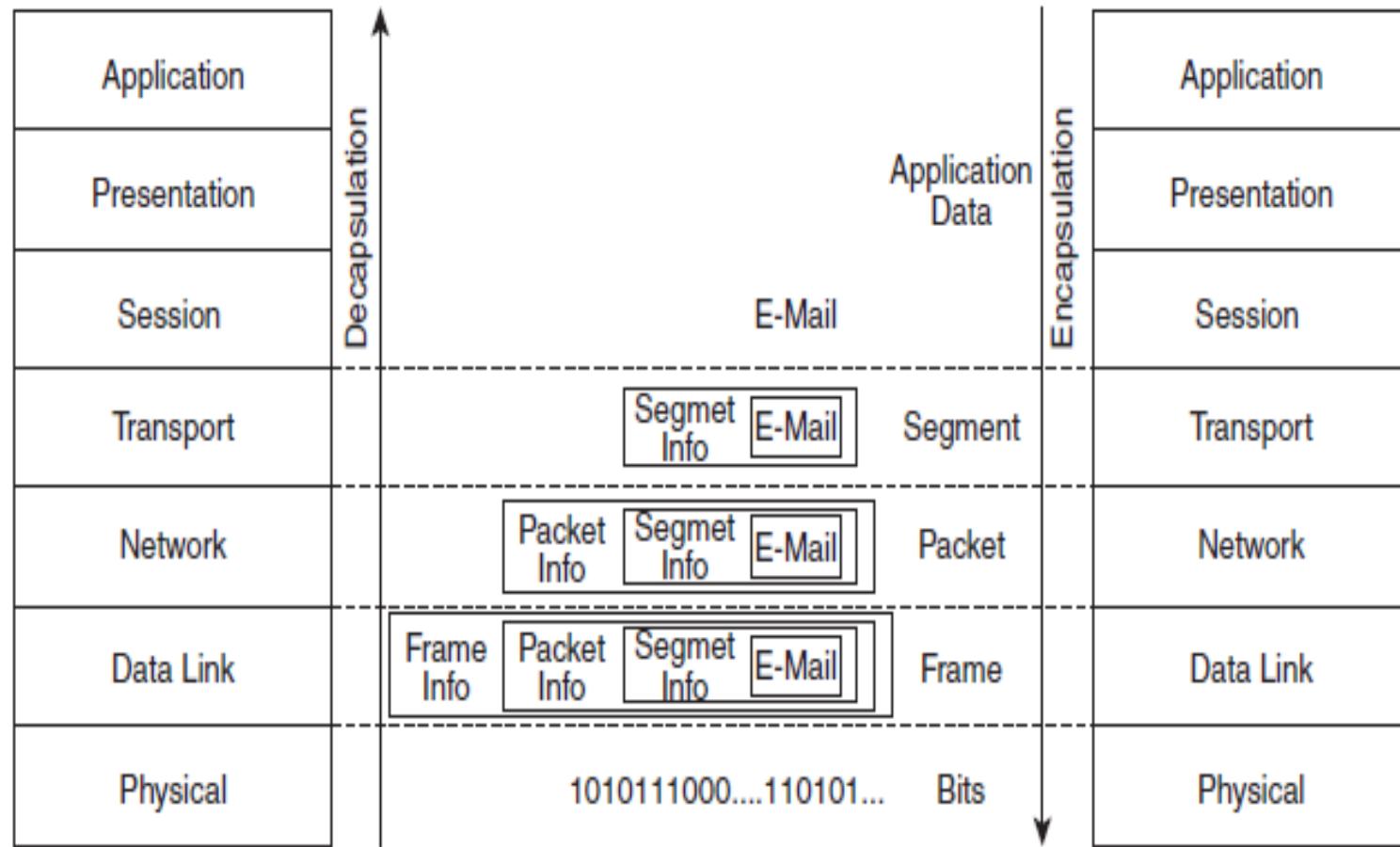
Communication Among OSI Layers(Cont.)

- Control information from each layer is **added** to the e-mail data before it passes to **lower layers**; this control information is necessary to allow the data to go through the network properly. Thus, the data at each layer is *encapsulated in* the information appropriate for that layer.
 - At **Layer 4**, the data is encapsulated in a **segment**.
 - At **Layer 3**, this segment is encapsulated in a **packet**.
 - At **Layer 2**, this packet is encapsulated in a **frame**.
 - Finally, at **Layer 1**, the frame is sent out on the **wire** (or air, if wireless is used) in bits.

Communication Among OSI Layers(Cont.)

- When data is received at the other end of the network, this additional information is analyzed and then removed as the data is passed to the higher layers toward the application layer. In other words, the data is decapsulation (*unencapsulated*).

Communication Among OSI Layers(Cont.)



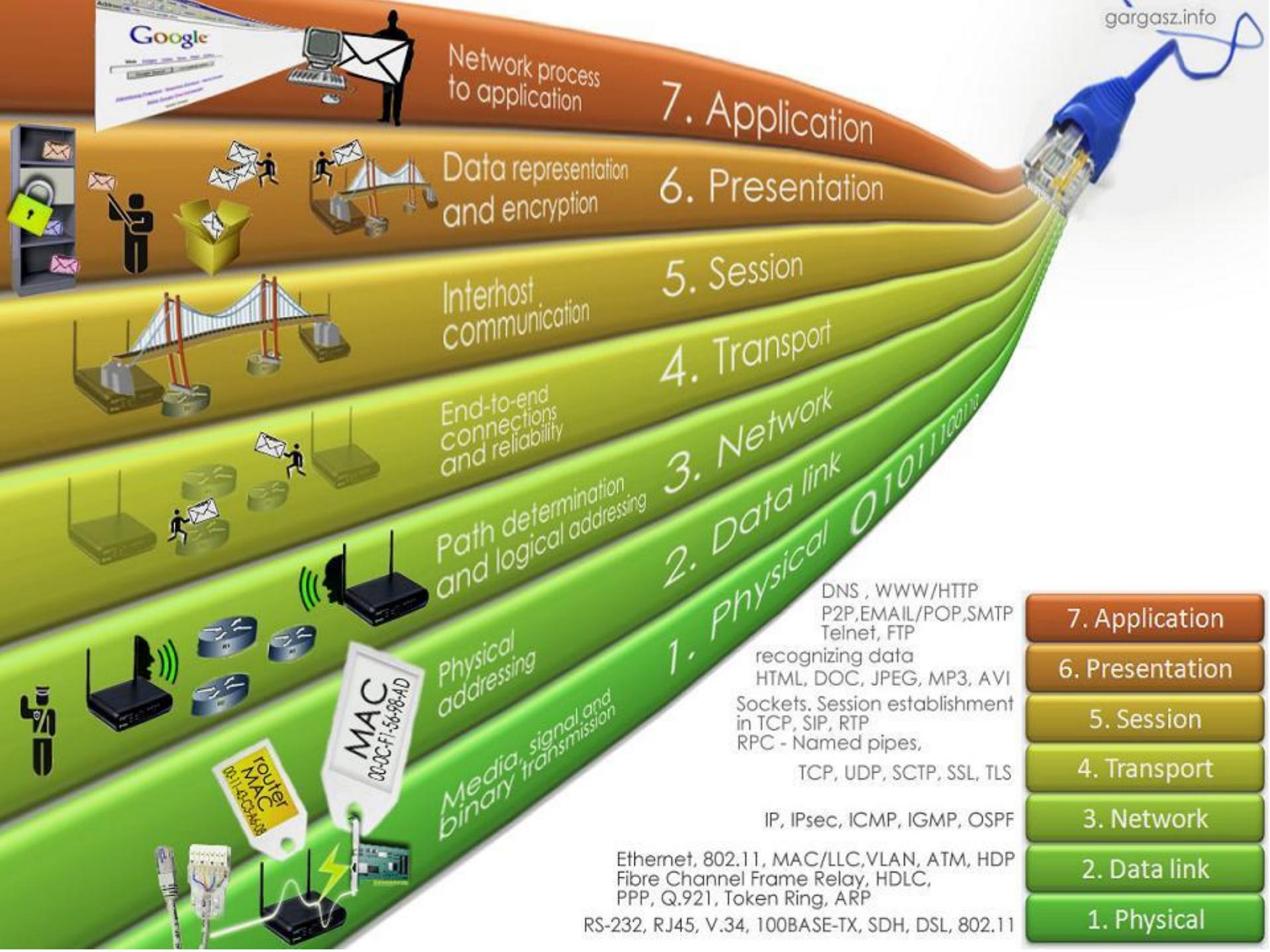
Thank You

Introduction to Networks

Lecture 4:

OSI Model Layers Protocols

Lecturer :Dr. Suad A. Alasadi

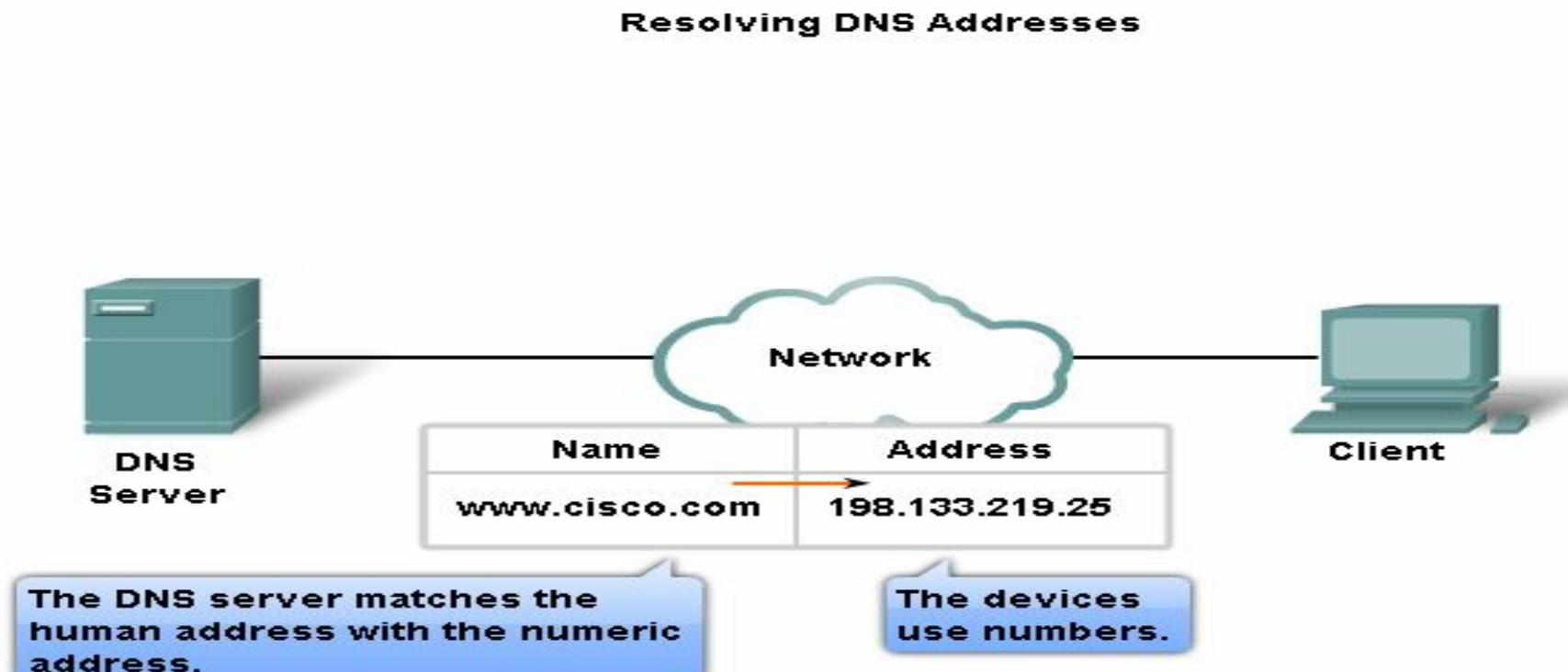


Protocols

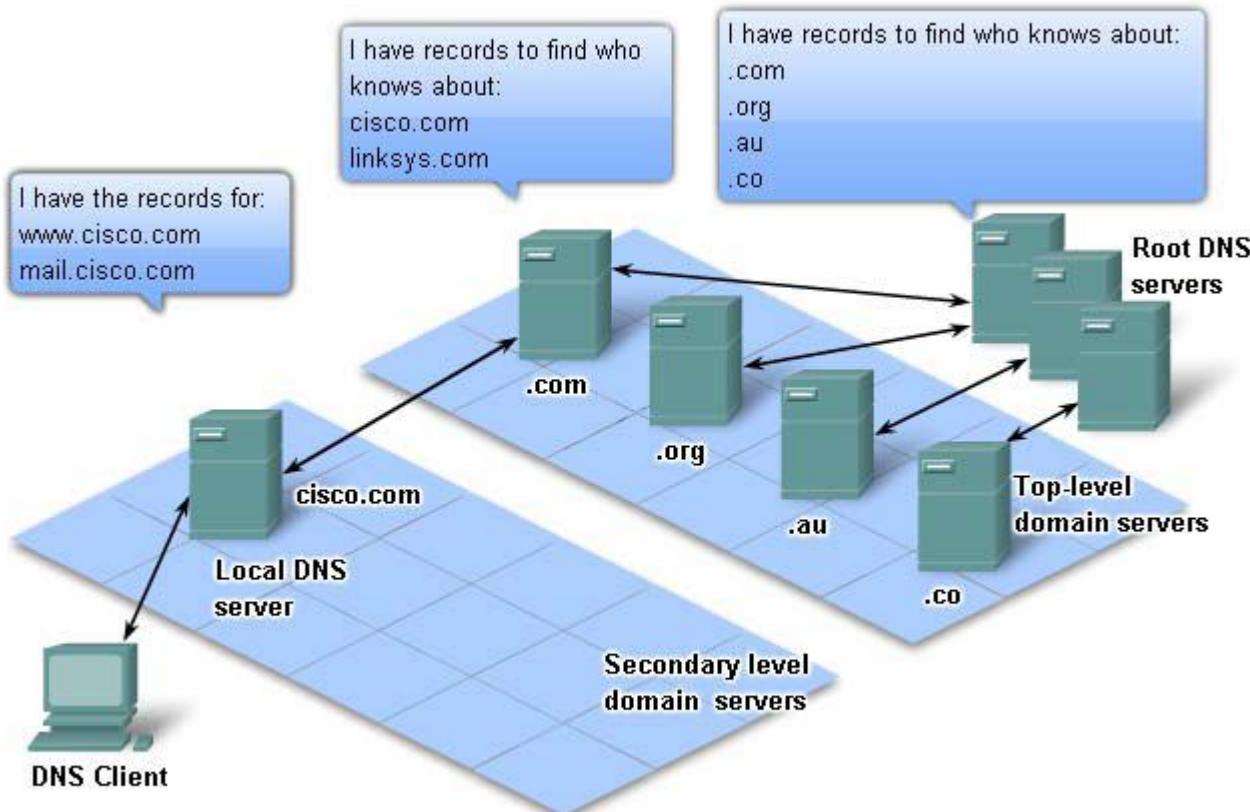
- A *protocol* is a set of rules. The OSI model provides a framework for the *communication protocols* used between computers. Just as we need rules of the road—for example, so that we know that a red light means stop and a green light means go—computers also need to agree on a set of rules to successfully communicate.
- Two computers must use **the same protocol** to communicate. Computers that try to use different protocols would be analogous to speaking in Italian to someone who understands only English—it would not work.
- Many *protocol suites* define various protocols that correspond to the functions defined in the seven OSI layers, **including routed protocols**, a selection of routing protocols, applications, and so forth. Protocol suites are also known as *protocol stacks*.
- The most widely used network protocol suite today is the **TCP/IP suite**.

Application layer Protocols

- Application layer examples include:
- Domain Name Service (DNS) :DNS protocol is used to resolve Internet names to IP addresses.



Application layer protocols



A hierarchy of DNS servers contains the resource records that match names with addresses.

Application layer protocols

- Application layer examples include:
- **Dynamic Host Configuration Protocol (DHCP)** :Enables devices on a network to obtain IP addresses and other information from a DHCP server. DHCP allows a host to obtain an IP address dynamically when it connects to the network.
- **File Transfer Protocol (FTP)** : FTP was developed to allow for file transfers between a client and a server . (FTP) Protocol is used for interactive file transfer between systems.
- **Simple Mail Transfer Protocol (SMTP)** : is used for the transfer of mail messages and attachments.
- **Terminal Emulation Protocol (Telnet)**: is used to provide remote access to servers and networking devices.

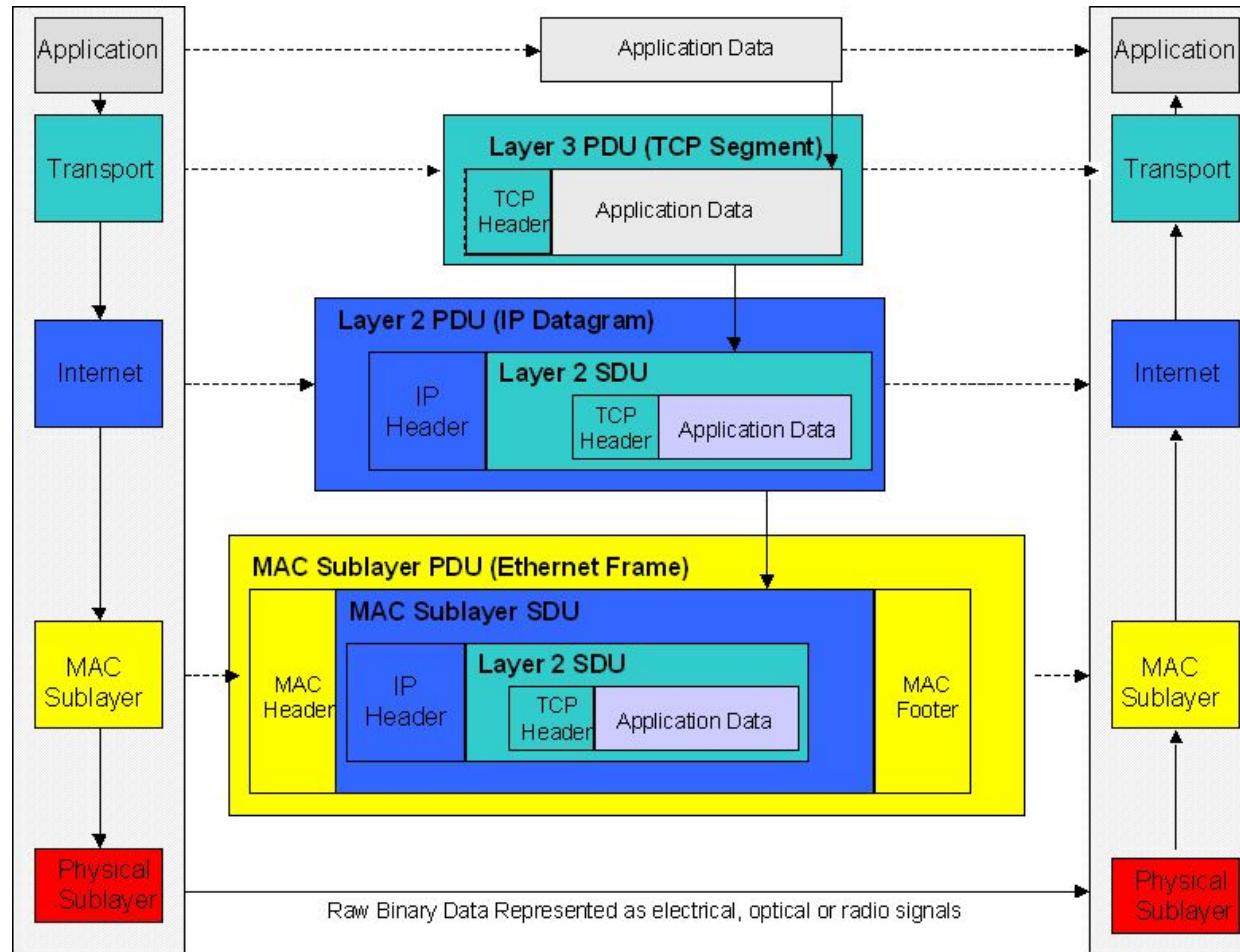
Application layer protocols

- **Application layer examples include:**
- **Hypertext Transfer Protocol (HTTP)** : The (HTTP) is one of the protocols in the TCP/IP suite, was originally developed to publish and retrieve HTML pages and is now used for distributed, collaborative information systems. HTTP is used across the WWW for data transfer and is one of the most used application protocols. (HTTP) is used to transfer files that make up the Web pages of the World Wide Web.
- **Uniform Resource Locator (URL)** : When a web address (or URL) is typed into a web browser, the web browser establishes a connection to the web service running on the server using the HTTP protocol. URLs (or Uniform Resource Locator) and URIs (Uniform Resource Identifier) are the names most people associate with web addresses.

Application layer protocols

- **Application layer examples include:**
- Post Office Protocol (POP).
- Internet Message Access Protocol (IMAP).
- Internet Relay Chat (IRC).
- Simple Network Management Protocol (SNMP)

Transport Layer protocols



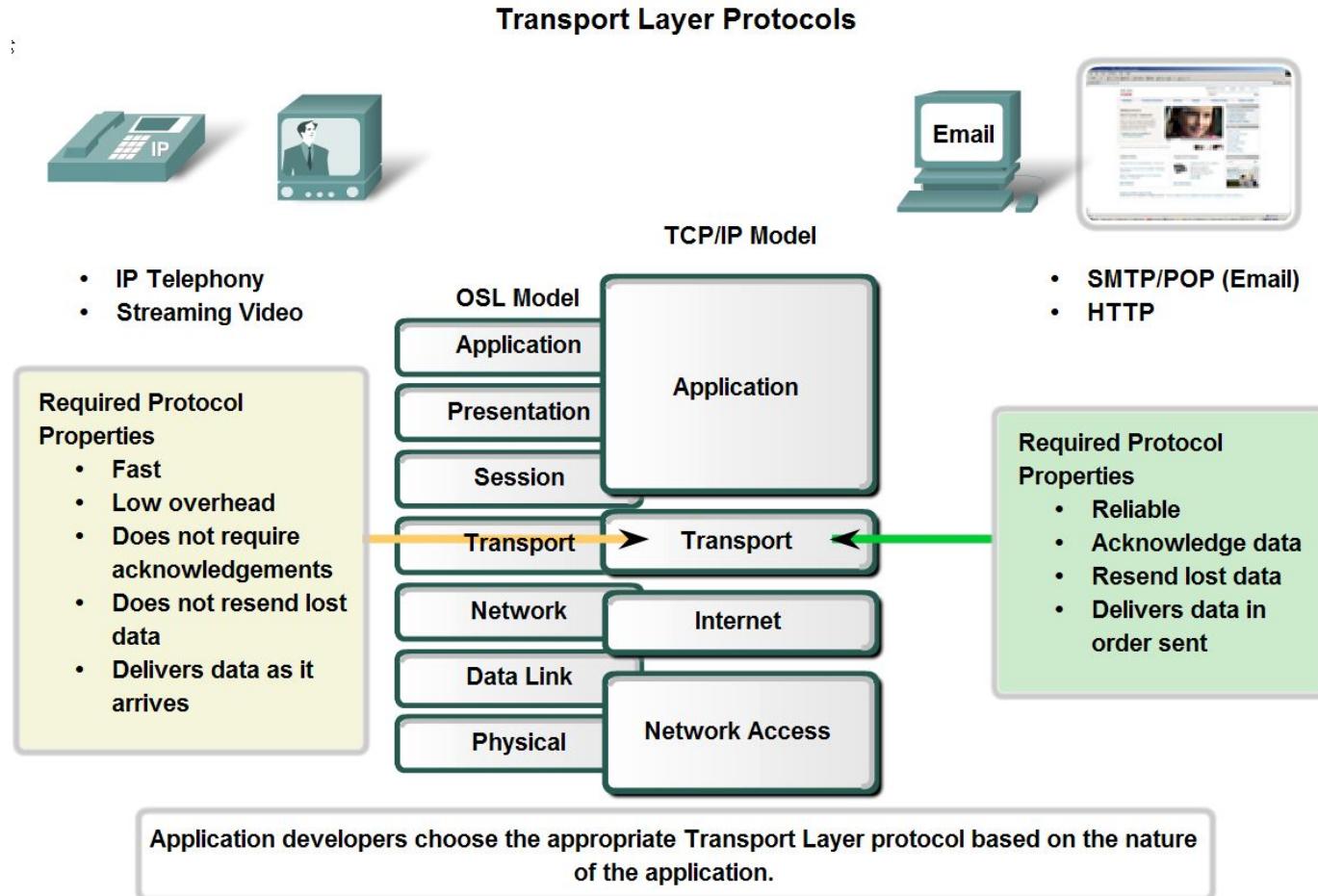
Transport Layer **protocols**

- The Transport layer provides for the **segmentation of data** and the **control necessary** to reassemble these pieces into the various communication streams. Its primary responsibilities to accomplish this are:
 - Segmenting data and managing each piece.
 - Reassembling the segments into streams of application data
 - Identifying the different applications.

Transport Layer protocols

UDP Protocol

TCP Protocol



Transport Layer Header



TCP and UDP Headers

TCP SEGMENT & HEADER FIELDS

Bit 0	Bit 15 Bit 16	Bit 31
Source Port (16)		Destination Port (16)
Sequence Number (32)		
Acknowledgement Number (32)		
Header Length (4) Reserved (6) Code Bits (6)	Window (16)	
Checksum (16)	Urgent (16)	
Options (0 or 32 if any)		
APPLICATION LAYER DATA SEGMENT (Size varies)		

↑
20 Bytes
↓

UDP SEGMENT & HEADER FIELDS

Bit (0)	Bit (15) Bit (16)	Bit (31)
Source Port (16)		Destination Port (16)
Length (16)		Checksum (16)
APPLICATION LAYER DATA SEGMENT (Size varies)		

↑
8 Bytes
↓

TCP Connection Establishment and Termination

- Three way handshake
- Step 1
- A TCP client begins the three-way handshake by sending a segment with the SYN (Synchronize Sequence Number) control flag set, indicating an initial value in the sequence number field in the header. This initial value for the sequence number, known as the Initial Sequence Number (ISN), is **randomly chosen** and is used to begin tracking the flow of data from the client to the server for this session. The ISN in the header of each segment is increased by one for each byte of data sent from the client to the server as the data conversation continues

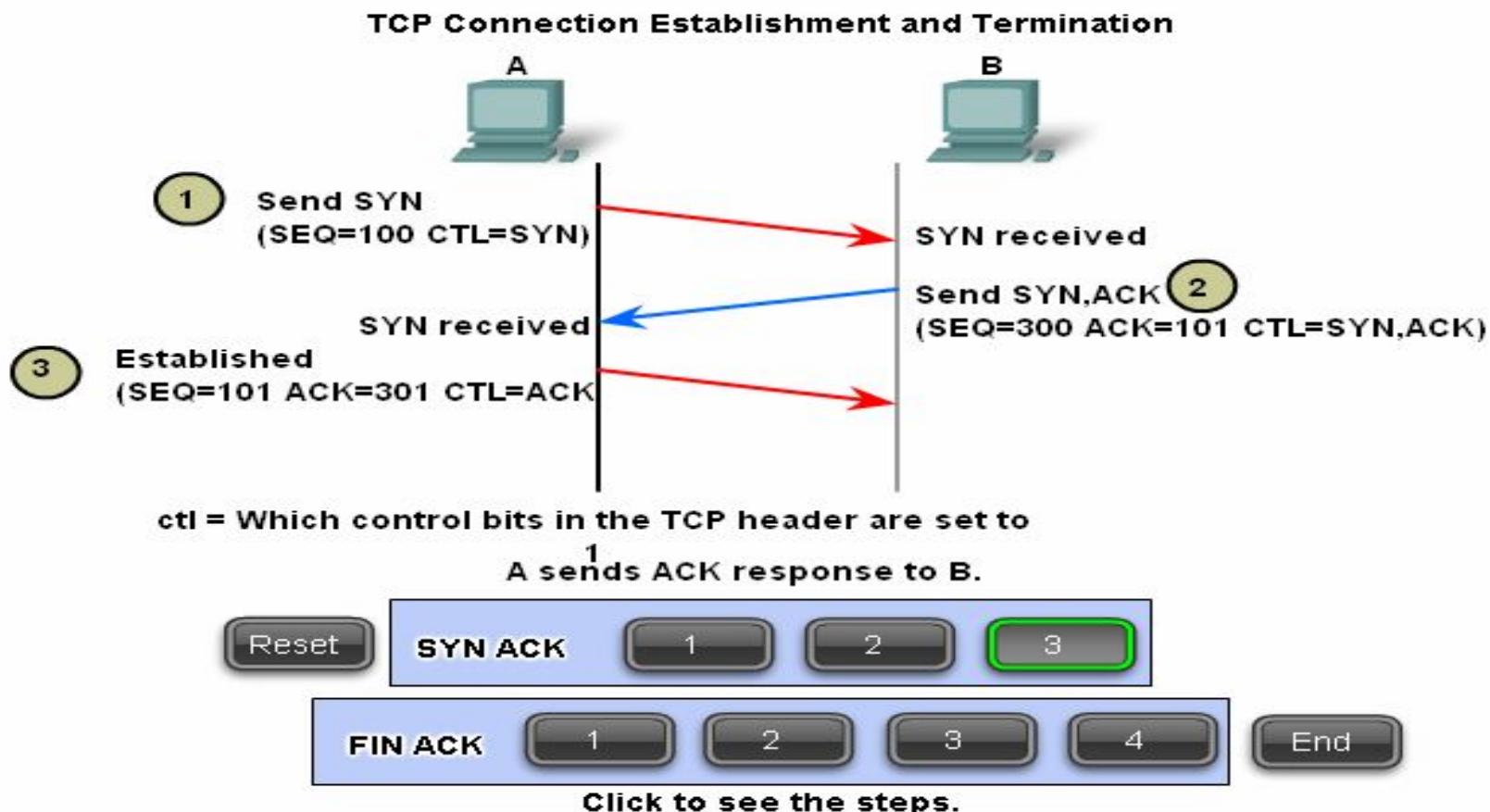
TCP Connection Establishment and Termination

- **Step 2**
- The TCP server needs to acknowledge the receipt of the SYN segment from the client to establish the session from the client to the server. To do so, the server sends a segment back to the client with the ACK flag set indicating that the Acknowledgment number is significant. With this flag set in the segment, the client recognizes this as an acknowledgement that the server received the SYN from the TCP client.
- The value of the acknowledgment number field is equal to the client initial sequence number plus 1.

TCP Connection Establishment and Termination

- **Step 3**
- Finally, the TCP client responds with a segment containing an ACK that is the response to the TCP SYN sent by the server. There is no user data in this segment.

TCP Connection Establishment and Termination



The Transport Layer applications port numbers

- **Transport Layer Role and Services**



(The Transport Layer(Cont.

Port Addressing

The Internet Assigned Numbers Authority (IANA) assigns port numbers. IANA is a standards body that is responsible for assigning various addressing standards.

There are different types of port numbers:

1- Well Known Ports (Numbers 0 to 1023) - These numbers are reserved for services and applications.

2- Registered Ports (Numbers 1024 to 49151) - These port numbers are assigned to user processes or applications. These processes are primarily individual applications that a user has chosen to install rather than common applications that would receive a Well Known Port. When not used for a server resource, these ports may also be used dynamically selected by a client as its source port.

3- Dynamic or Private Ports (Numbers 49152 to 65535) - Also known as Ephemeral Ports, these are usually assigned dynamically to client applications when initiating a connection.

The Transport Layer(Cont.)

Port Numbers	
Port Number Range	Port Group
0 to 1023	Well Known (Contact) Ports
1024 to 49151	Registered Ports
49152 to 65535	Private and/or Dynamic Ports

Registered TCP Ports:

1863	MSN Messenger
8008	Alternate HTTP
8080	Alternate HTTP

Well Known TCP Ports

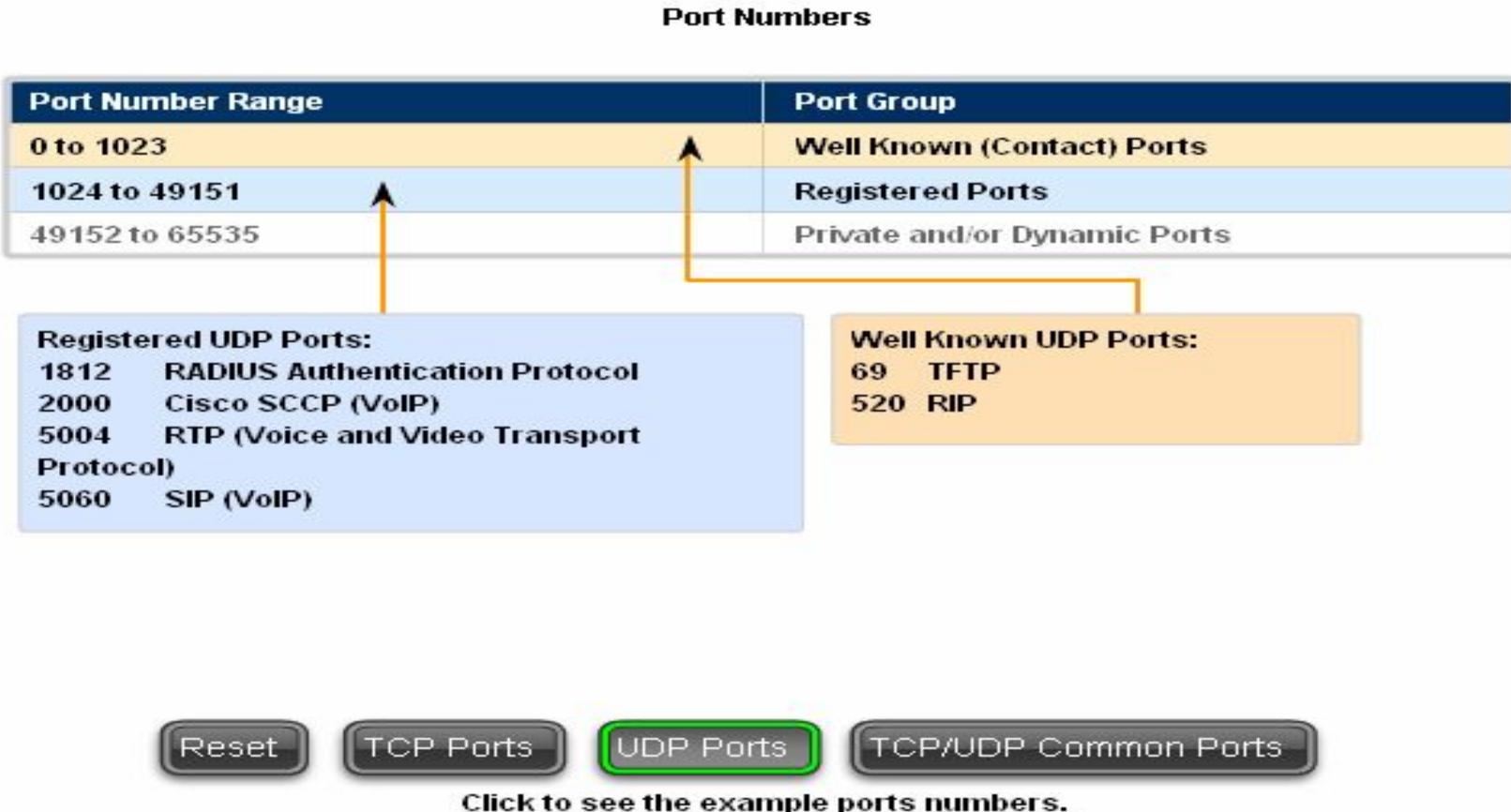
21	FTP
23	Telnet
25	SMTP
80	HTTP
110	POP3
194	Internet Relay Chat (IRC)
443	Secure HTTP (HTTPS)

Buttons:

Reset **TCP Ports** UDP Ports TCP/UDP Common Ports

Click to see the example ports numbers.

The Transport Layer(Cont.)



The Transport Layer(Cont.)

Port Numbers

Port Number Range	Port Group
0 to 1023	Well Known (Contact) Ports
1024 to 49151	Registered Ports
49152 to 65535	Private and/or Dynamic Ports

Registered TCP/UDP Common Ports:

1433	MS SQL
2948	WAP (MMS)

Well Known TCP/UDP Common Ports:

53	DNS
161	SNMP
531	AOL Instant Messenger, IRC

Click to see the example ports numbers.

Reset **TCP Ports** **UDP Ports** **TCP/UDP Common Ports**

The Transport Layer(Cont.)

- **Domain Name Service (DNS):** TCP/UDP port 53
- **HTTP:** TCP port 80
- **Simple Mail Transfer Protocol (SMTP):** TCP port 25
- **Post Office Protocol (POP):** UDP port 110
- **Telnet:** TCP port 23
- **DHCP:** UDP port 67
- **FTP:** TCP ports 20 and 21

Thank You

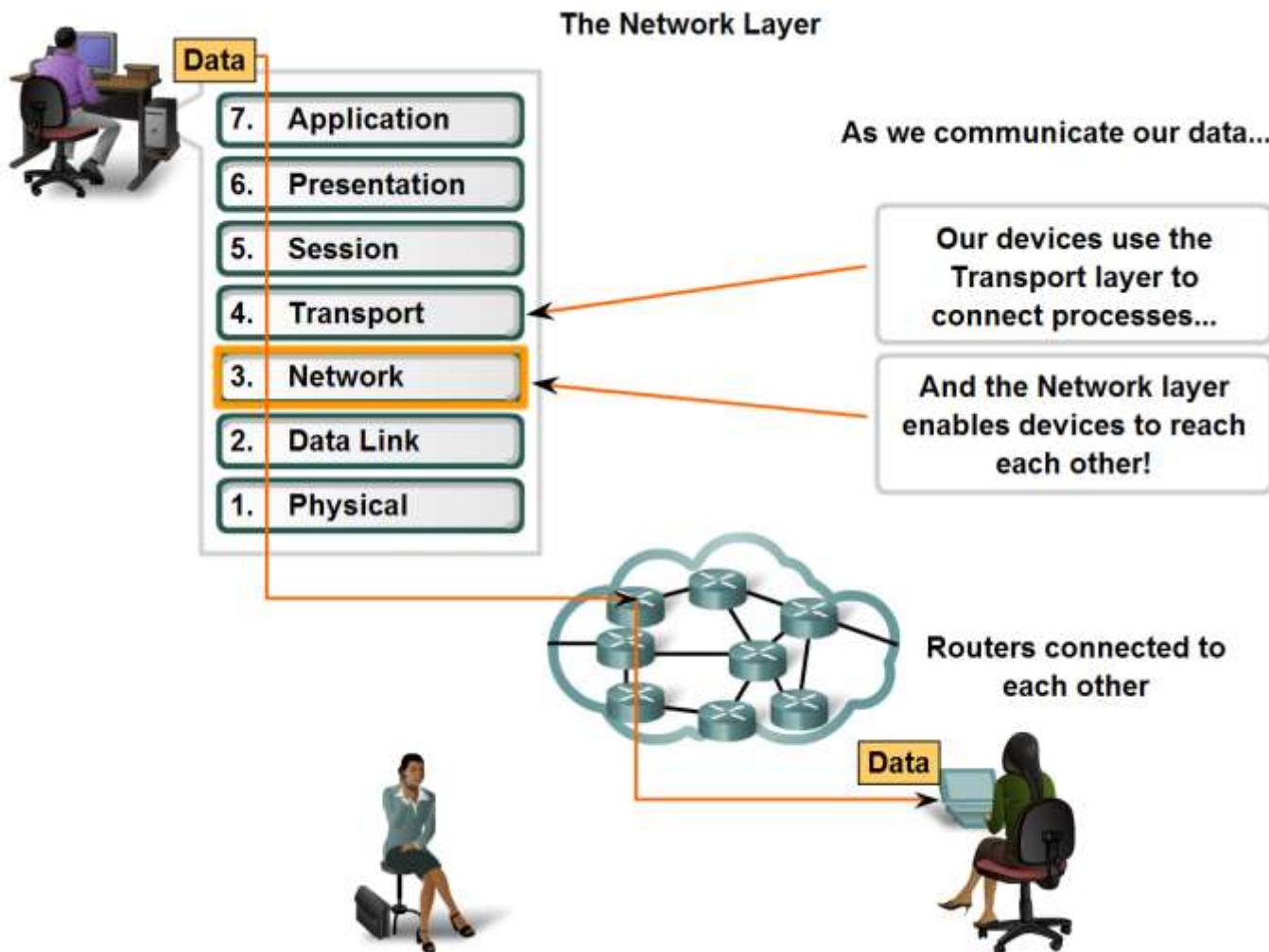
Computer Network Fundamentals

Lecture 5:

OSI Model Network layer protocols

Assist Prof. Dr. Aladdin Abbas Abdulhassan

Network Layer Protocols and Internet Protocol (IP)



The Network Layer

Layer 3 uses four basic processes:

- Addressing
- Encapsulation
- Routing
- Decapsulation

The Network Layer (Cont.)

1- Addressing

- First, the Network layer must provide a mechanism for addressing these end devices. If individual pieces of data are to be directed to an end device, that device must have a unique address.
- In an IPv4 network, when this address is added to a device, the device is then referred to as a host .

The Network Layer (Cont.)

2- Encapsulation

Second, the Network layer must provide encapsulation.

- During the encapsulation process, Layer 3 receives the Layer 4 data and adds a Layer 3 header, or label, to create the Layer 3 data. When referring to the Network layer, we call this data a packet.
- Layer 4 data +Layer 3 header, or label = packet
- When a packet is created, the *header must contain*, among other information, the destination address. The Layer 3 header also contains the source address.

The Network Layer (Cont.)

3- Routing

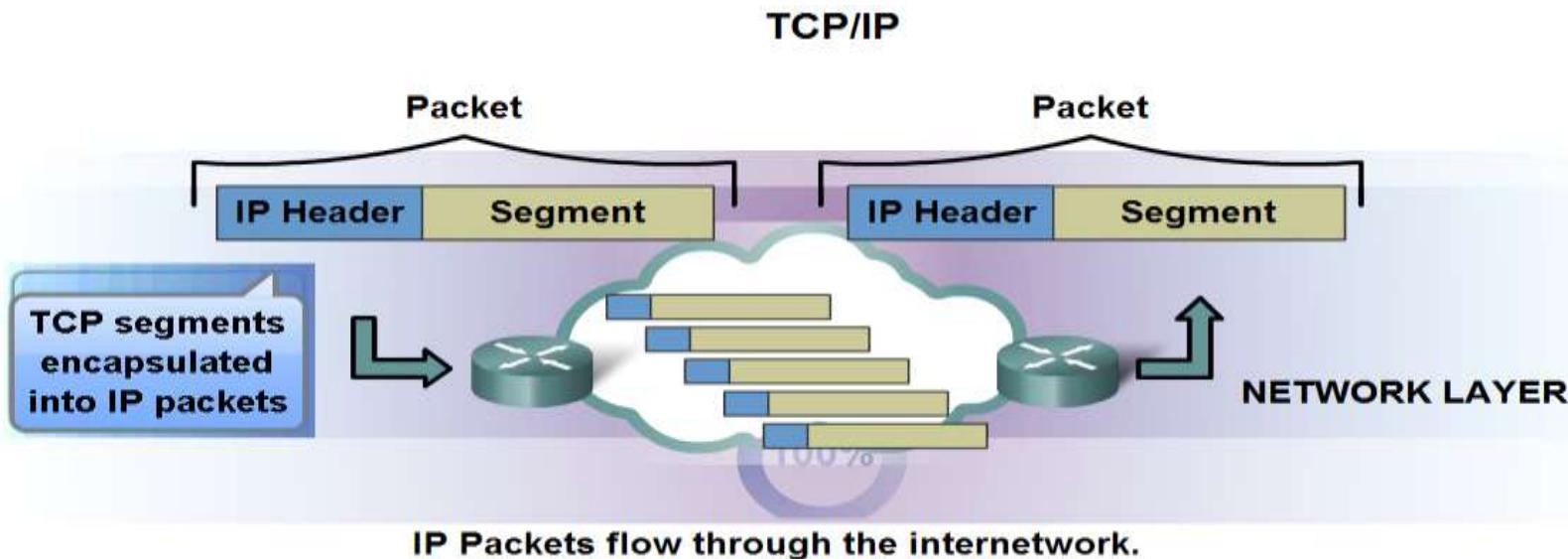
Next, the Network layer must provide services to “direct these packets to their destination host. The source and destination hosts are *not always connected to the same network*. In fact, the packet might have to travel through many different networks. Along the way, each packet must be guided through the network to reach its final destination. Intermediary devices that connect the networks are called routers. The *role of the router* is to ***select paths for and direct packets toward their destination***. This process is known as **routing**.

Network Layer Protocols(Cont.)

Protocols implemented at the Network layer include:

- Internet Protocol version 4 (IPv4)
- Internet Protocol version 6 (IPv6)
- Novell Internetwork Packet Exchange (IPX)
- AppleTalk
- Connectionless Network Service (CLNS/DECNet)

IPv4 protocol Basic characteristics

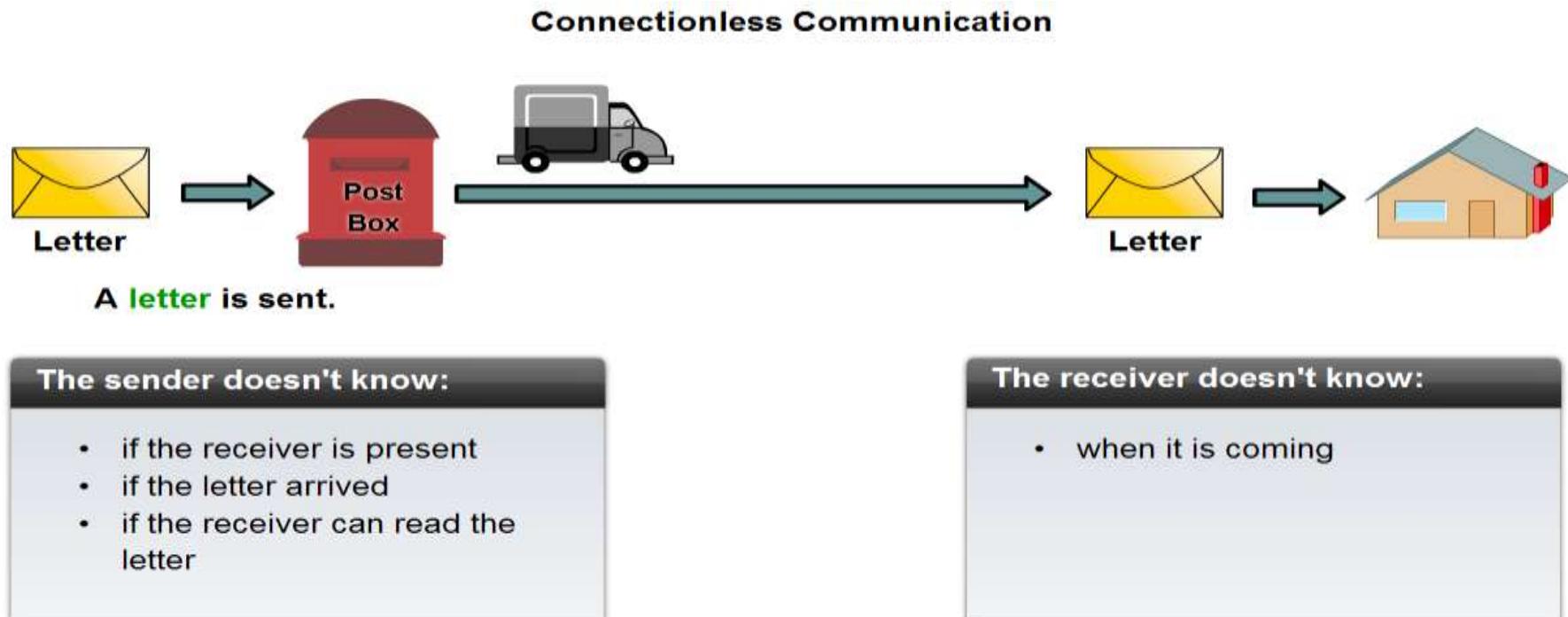


- Connectionless - No connection is established before sending data packets.
- Best Effort (unreliable) - No overhead is used to guarantee packet delivery.
- Media Independent - Operates independently of the medium carrying the data.

Internet Protocol IPv4 was designed as a protocol with low overhead. It *provides* only the functions that are necessary to deliver a packet from a source to a destination over an interconnected system of networks.

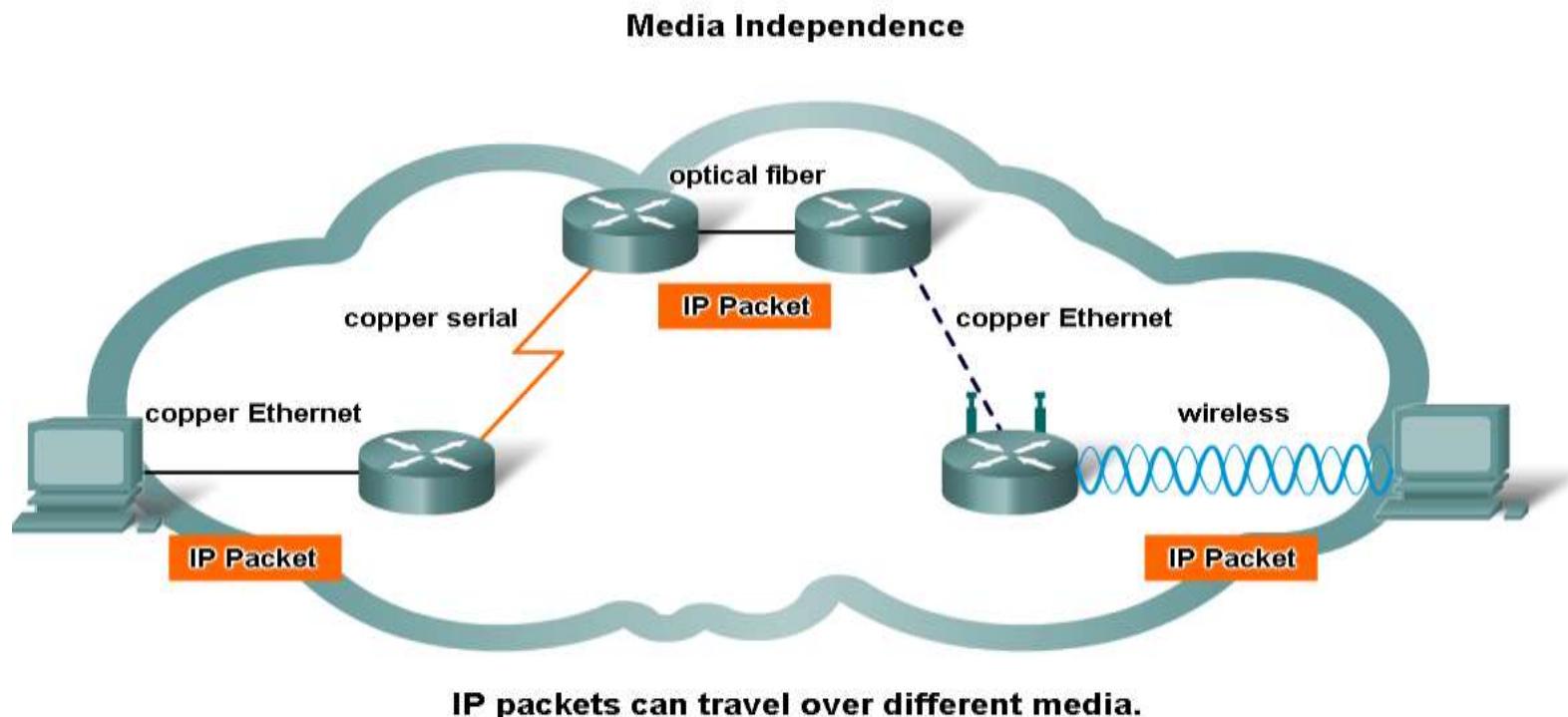
Internet Protocol (IP):Example

- Describe the implications for the use of the IP protocol as it is connectionless and Unreliable protocol



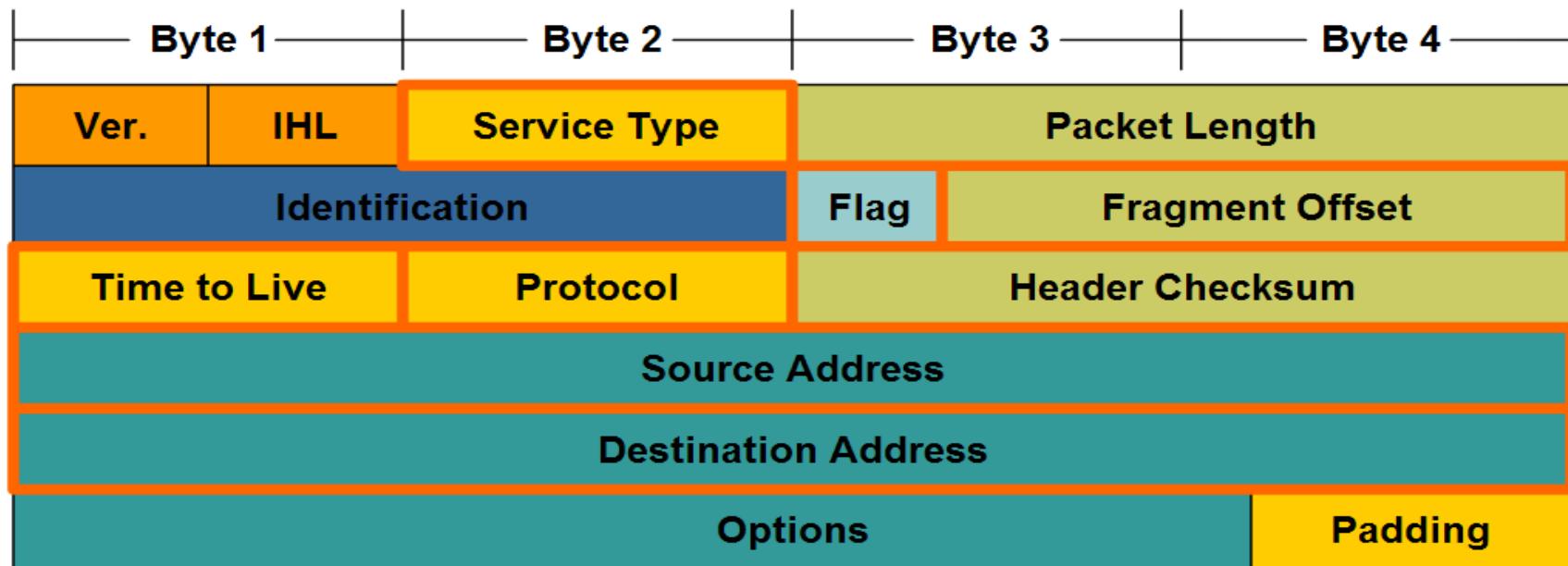
Internet Protocol (IP):Example

Use of the IP as it is media independent



Internet Protocol (IP) Header fields

IPv4 Packet Header Fields



Internet Protocol (IP) header fields

IP Destination Address

The IP Destination Address field contains a 32-bit binary value that represents the packet destination Network layer host address.

IP Source Address

The IP Source Address field contains a 32-bit binary value that represents the packet source Network layer host address.

Time to Live (TTL)

The 8-bit TTL field describes the maximum hops the packet can take before it is considered “lost”.

Version

Indicates IP version 4 or 6.

Addressing

- In the network, there are three types of addressing:
 1. Physical Address
 2. Logical Address
 3. Port Addressing

Physical Addresses

- **Physical Addresses**

it is the Media Access Control (MAC)address that assign to the device in the data link layer of OSI model.

- Each frame contains source MAC address and destination MAC address.
- When a network interface card is manufactured, it is assigned an address—called a *burned-in address (BIA)*—that doesn't change when the network card is installed in a device and is moved from one network to another.
- The MAC address concerned with the person not where live, and this address doesn't change when the person move from one place to another.

Physical Addresses (Con.)

- MAC Address

The **BIA** is a **48-bit value**. The upper 24 bits are an Organizational Unique Identifier (OUI) representing the vendor that makes the device. The lower 24 bits are a unique value for that OUI, typically the device's serial number.

Logical Address

- **Logical Address**

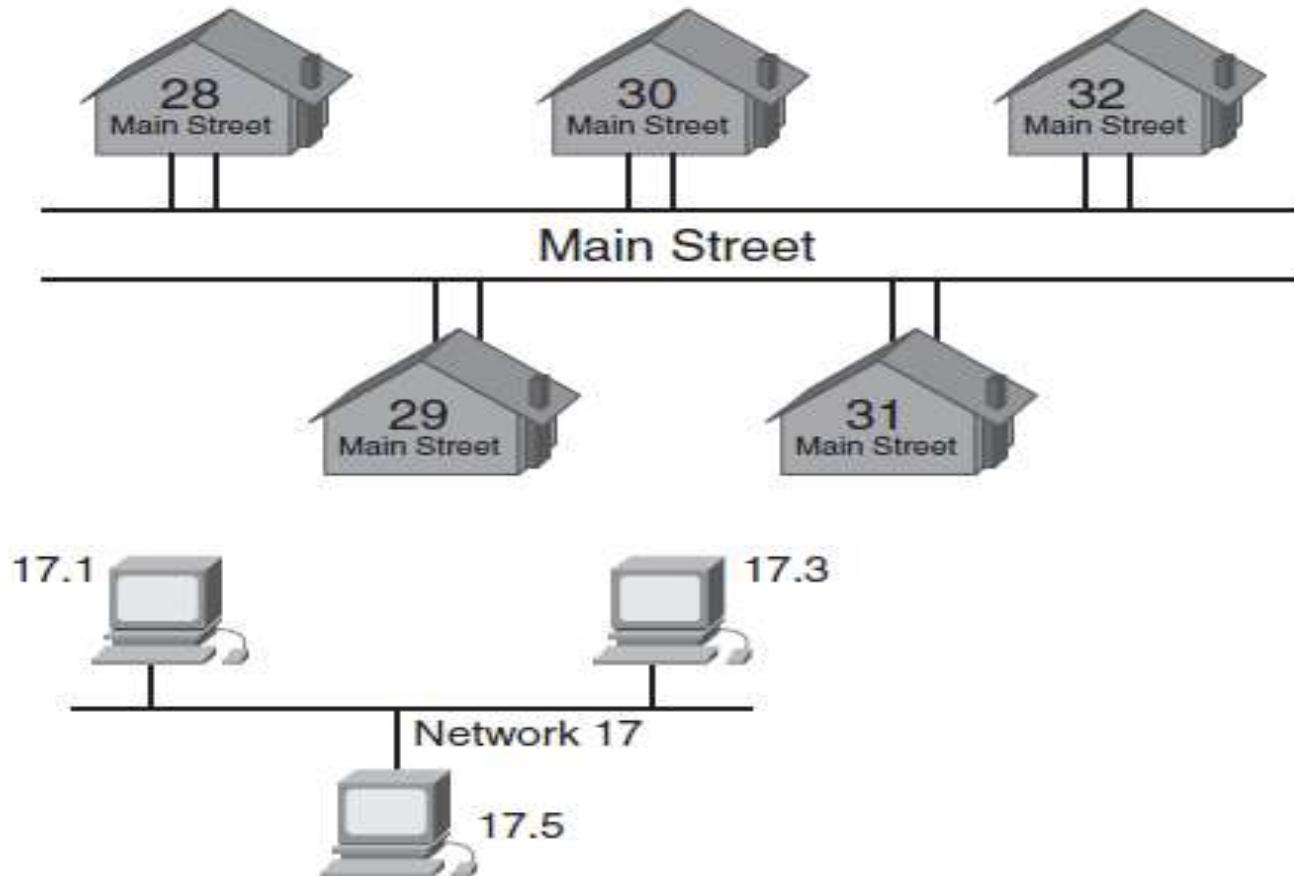
it is the Internet Protocol (**IP**) address that assign to the device in the **network** layer of OSI model.

- When you send a letter to someone, you have to know that **person's postal address**. Because every postal address in the world is **unique**, you can potentially send a letter to anyone in the world.
- Postal addresses are logical and hierarchical—for example, they include the country, province/ state, street, and building/house number.

Logical Address(Con.)

- Network layer addresses are also logical and hierarchical, and they are either defined **statically by an administrator or obtained automatically from a server.**
- They have 32 bits and divided into two main parts: the **network that the device is on**(16 bit) (similar to the street, city, province, and so on) and the **device number** on that network (16 bits)(similar to the building number).

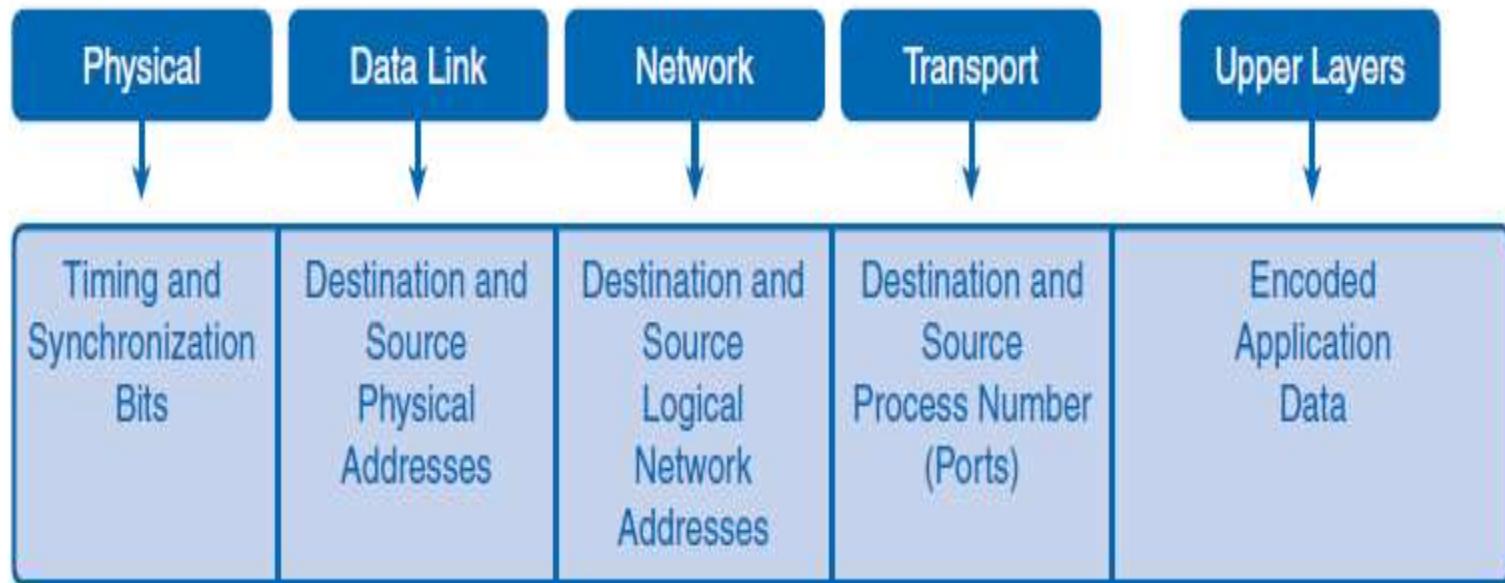
Logical Address(Con.)



Logical Address(Con.)

- The top portion of Figure illustrates Main Street with various houses. All these houses have one portion of their address in common—Main Street—and one portion that is unique—their house number.
- The lower portion of Figure illustrates a network, 17, with various PCs on it. All these PCs have one portion of their address in common—17—and one part that is unique—their device number. Devices on the same logical network must share the same network portion of their address and have different device portions.

The information added at each layer



Thank You



IPv4 header

Version	IHL	Type of Service	Total Length			
Identification			Flags	Fragment Offset		
Time to Live	Protocol		Header Checksum			
Source Address						
Destination Address						
Options				Padding		

IPv4 Header Format

0				1				2				3																									
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31						
Version		IHL		DSCP				ECN				Total Length																									
Identification								Flags				Fragment Offset																									
Time To Live				Protocol				Header Checksum																													
Source IP Address								Destination IP Address																													
Options (if IHL > 5)																																					

IPv4 Header Format

The IPv4 packet header consists of 14 fields, of which 13 are required. The 14th field is optional and aptly named: options. The most significant bits are considered to come first (MSB 0 bit numbering). The most significant bit is numbered 0, so the version field is actually found in the four most significant bits of the first byte, for example.

- **Version** - The first header field in an IP packet is the four-bit version field. For IPv4, this is always equal to 4, for IPv6 is equal to 6.



- **Internet Header Length (IHL)**

The Internet Header Length (IHL) field has 4 bits, which is the number of 32-bit words. Since an IPv4 header may contain a variable number of options, ***this field specifies the size of the header (this also coincides with the offset to the data).*** The minimum value for this field is 5, which indicates a length of 5×32 bits = 160 bits = 20 bytes. As a 4-bit field, the maximum value is 15 words (15×32 bits, or 480 bits = 60 bytes).

- **Type of Service (TOS)** - Defines delay, throughput and reliability requirement of the IP packet.

Differentiated Services Code Point (DSCP)

Originally defined as the Type of service (ToS) field. This field is now defined by [RFC 2474](#) (updated by [RFC 3168](#) and [RFC 3260](#)) for **Differentiated services** (DiffServ). New technologies are emerging that require real-time data streaming and therefore make use of the DSCP field. An example is [Voice over IP](#) (VoIP), which is used for interactive data voice exchange.

Explicit Congestion Notification (ECN)

This field is defined in [RFC 3168](#) and allows end-to-end notification of **network congestion** without dropping packets. ECN is an optional feature that is only used when both endpoints support it and are willing to use it. It is only effective when supported by the underlying network.

- **Total Length-** Indicates the entire packet size, including header and data, in bytes.

This 16-bit field defines the entire **packet size in bytes**, including header and data. The minimum size is 20 bytes (header without data) and the maximum is 65,535 bytes. All hosts are required to be able to reassemble datagrams of size up to 576 bytes, but most modern hosts handle much larger packets. Sometimes links impose further restrictions on the packet size, in which case datagrams must be fragmented. Fragmentation in IPv4 is handled in either the host or in routers.

- **Identification-** Used for uniquely identifying fragments of an original IP datagram.



This field is an identification field and is primarily used for uniquely identifying the group of fragments of a single IP datagram. Some experimental work has suggested using the ID field for other purposes, such as for adding packet-tracing information to help trace datagrams with spoofed source addresses.

- **Flags-** Used to control or identify fragments. Specifies whether fragmentation should occur.

A three-bit field follows and is used to control or identify fragments. They are (in order, from most significant to least significant):

- bit 0: Reserved; must be zero.
- bit 1: Don't Fragment (DF)
- bit 2: More Fragments (MF)

If the DF flag is set, and fragmentation is required to route the packet, then the packet is dropped. This can be used when sending packets to a host that does not have sufficient resources to handle fragmentation. It can also be used for [Path MTU Discovery](#), either automatically by the host IP software, or manually using diagnostic tools such as [ping](#) or [traceroute](#). For unfragmented packets, the MF flag is cleared. For fragmented packets, all fragments except the last have the MF flag set. The last fragment has a non-zero Fragment Offset field, differentiating it from an unfragmented packet.

- **Fragment Offset-** Allows a receiver to determine the place of a particular fragment in the original IP datagram, The fragment offset field is measured in units of eight-byte blocks. It is 13 bits long and specifies the offset of a particular fragment relative to the beginning of the original unfragmented IP datagram. The first fragment has an offset of zero. This allows a maximum offset of $(2^{13} - 1) \times 8 = 65,528$ bytes, which would exceed the maximum IP packet length of 65,535 bytes with the header length included ($65,528 + 20 = 65,548$ bytes).



- **Time To Live (TTL)**- The maximum number of hops a packet can traverse between the source and destination hosts. Each packet switch (or router) that a packet crosses decrements the TTL field by one. When the TTL field becomes zero, the packet is no longer forwarded by a packet switch and is discarded. This mechanism prevents packets from being trapped in endless routing loops, clogging up a network.

An eight-bit time to live field helps prevent datagrams from persisting (e.g. going in circles) on an internet. This field limits a datagram's lifetime. It is specified in seconds, but time intervals less than 1 second are rounded up to 1. In practice, the field has become a hop count—when the datagram arrives at a router, the router decrements the TTL field by one. When the TTL field hits zero, the router discards the packet and typically sends an ICMP Time Exceeded message to the sender. The program traceroute uses these ICMP Time Exceeded messages to print the routers used by packets to go from the source to the destination.
- **Protocol**—identifies the protocol used in the data portion of the IP packet. It is used by the IP module at the receive end to pass the data to the correct transport layer module (For example, TCP or UDP). For example, 6 for TCP, and 17 for UDP. Also, it represents the protocol for the same layer. For example, value 1 represents ICMP, 2 for IGMP.

This field defines the protocol used in the data portion of the IP datagram. The [Internet Assigned Numbers Authority](#) maintains a [list of IP protocol numbers](#) which was originally defined in [RFC 790](#).
- **Header Checksum**- used for error checking of the header. At each hop, the checksum of the header is compared to the value of this field. If a header checksum is found to be mismatched, the



packet is discarded. Because the TTL field is decremented on each hop; the checksum must be recomputed and inserted into the IP packet.

The 16-bit **checksum** field is used for error-checking of the header. When a packet arrives at a router, the router calculates the checksum of the header and compares it to the checksum field. If the values do not match, the router discards the packet. Errors in the data field must be handled by the encapsulated protocol. Both **UDP** and **TCP** have checksum fields.

When a packet arrives at a router, the router decreases the TTL field. Consequently, the router must calculate a new checksum. **RFC 791** defines the checksum calculation:

- **Source Address**—IP address of the sender of the packet. This may not be the actual address of the sender if NAT is used.
- **Destination address**—IP address of the receiver of the IP packet.
- **Options**-Additional header fields (called options) may follow the destination address field, but these are not often used.

The options field is not often used. Note that the value in the IHL field must include enough extra 32-bit words to hold all the options (plus any padding needed to ensure that the header contains an integer number of 32-bit words). The list of options may be terminated with an EOL (**End of Options List**, 0x00) option; this is only necessary if the end of the options would not otherwise coincide with the end of the header. The possible options that can be put in the header are as follows:

- Note: If the header length is greater than 5 (i.e., it is from 6 to 15) it means that the options field is present and must be considered.
- Note: Copied, Option Class, and Option Number are sometimes referred to as a single eight-bit field, the *Option Type*.

The following two options are discouraged because they create security concerns: Loose Source and Record Route (LSRR) and Strict Source and



Record Route (SSRR). Many routers block packets containing these options.^[13]

- **Data** -The data portion of the packet is not included in the packet checksum. Its contents are interpreted based on the value of the Protocol header field.



IPv4 Network layer Classfull and Classless Addressing

This talk will cover the basics of IP addressing and subnetting.

Topics covered will include:

- What is an IP Address?
- What are Classes?
- What is a Network Address?
- What are Subnet Masks and Subnet Addresses?
- How are Subnet Masks defined and used?
- How can all this be applied?
- What is CIDR?

The Network Layer (layer 3)

The network layer is responsible for the source-to-destination delivery of a packet, possibly across multiple networks (links). It 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.

In other word The Network layer is responsible for routing the packet based on its logical address. It also fragments and reassembles packets if necessary.

Internet Protocol (IP) Addresses (Logical Address)

What is IP Addressing?

Every host on a network needs to have a unique address, similar to you needing a unique address for your house. With this unique address, it is possible to send data from host to host. Every packet contains addressing information in the header, and the IP address in the header is used to route packets. If several people on your street had the same address, the post office would have a difficult time



sorting mail. For a similar reason, IP addresses are unique on each network. IP addressing is simply configuring each host with a valid IP address. For access to the Internet, a host must have an IP address that identifies not only the host address (like a house number) but also identifies the network address (like a street number). An administrator needs to be aware of proper addressing techniques so that the hosts on the network will function correctly.

An IP address is a logical identifier for a computer or device on a network. The key feature of IP addresses is that they can be routed across networks.

The format of an IP address is a 32-bit numeric address written as four numbers separated by periods, sometimes referred to as a dotted-quad. The range of each number can be from 0 to 255.

For example, 2.165.12.230 would be a valid IP address. The four numbers in an IP address are used to identify a particular network and a host within that network.

Protocols looks at IP addresses in ***binary*** form, but as humans, we prefer to see IP addresses in ***decimal*** form. Because the protocol is seeing only binary, working with IP addresses makes more sense when you also look at the IP addresses in binary. To do so, you need to understand the two numbering systems (***binary and decimal***) and be able to convert from one to another.

There are many versions of IP the TCP/IP protocol: ***IP version 4*** and ***IP version 6***. ***IP version 6*** is much more complicated than ***IP version 4*** and is much newer. First, we will be working with ***IP version 4*** which is the address format of the four digits separated by full-stops.

IPv4 uses 32-bit addresses, which means that the address space is 2^{32} or 4,294,967,296 (more than 4 billion). This means that, theoretically, if there were no restrictions, more than 4 billion devices could be connected to the Internet.



There are two types of addresses in the network layer of IPv4

- 1. Classful Address**
- 2. Classless Address**

1. Classful Address

IPv4 addresses Classes

IP addresses are divided into five classes:

- Class A: Large networks.
- Class B: Medium-sized networks.
- Class C: Small networks with less than 256 devices.
- Class D: Multicasting.
- Class E: Reserved.

An IP address has two parts :

Network Address & Host Address (also known as local or node)

1. Class A Addresses

The designers of the IP address scheme said that the first bit of the first byte in a Class A network address must always be off, or 0. Thus a Class A address must be between 0 and 127 inclusive.

Consider the following network address: 0xxxxxxxx



If all the other 7 bits turned off, then turned on sequentially, the class A range of network addresses will be found:

00000000 = 0

01111111 = 127

So, a Class A network is defined in the first octet between 0 and 127, and it can't be less or more. In a Class A network address, the first byte is assigned to the network address and the three remaining bytes are used for the node addresses. The Class A format is:

NETWORK.HOST.HOST.HOST

For example, in the IP address 49.22.102.70, the 49 is the network address, and 22.102.70 is the host address. Every machine on this particular network would have the distinctive network address of 49.

Class A network addresses are 1 byte long, with the first bit of that byte reserved and the seven remaining bits available for manipulation (addressing). As a result, the maximum number of Class A networks that can be created is 128. Because each of the seven bit positions can either be a 0 or a 1, thus 2⁷ or 128.

To complicate matters further, the network address of all 0s (00000000) is reserved to designate the default route. Additionally, the address 127, which is reserved for diagnostics, can't be used either, which means that you can really only use the numbers 1 to 126 to designate Class A network addresses. This means the actual number of usable Class A network addresses is 128 minus 2, or 126.

Each Class A address has three bytes (24-bit positions) for the host address of a machine. This means there are 2²⁴ or 16,777,216 - unique combinations and, therefore, precisely that many possible unique node addresses for each Class A network. Because node addresses with the two patterns of all 0s and all 1s are reserved, the actual maximum usable number of nodes for a Class A network is 2²⁴-2.



minus 2, which equals 16.777.214. Either way, that's a huge number of hosts on a network segment of class A.

2. Class B Addresses

In a Class B network, the first bit of the first byte must always be turned on, but the second bit must always be turned off (**10**). If all the other 6 bits turned off and then turned on, the range of Class B network will be found:

$$10000000 = 128$$

$$10111111 = 191$$

This means that a Class B network is defined when the first byte is configured from 128 to 191.

In a Class B network address, the first 2 bytes are assigned to the 'network address', and the remaining 2 bytes are used for 'node addresses'. The format is:

NETWORK.NETWORK.HOST.HOST

For example, in the IP address 172.16.30.56, the network address portion is 172.16, and the node address portion is 30.56 .

With a network address being 2 bytes (8 bits each), there would be 216 unique combinations. But the Internet Protocol designers decided that all Class B network addresses should start with the binary digit 1, then 0. This leaves 14 bit positions to manipulate, and therefore 16.384 (that is, 214) unique Class B network addresses.

A Class B address uses two bytes for node addresses. This is 216 minus the two reserved patterns (all 0s and all 1s), for a total of 65.534 possible node addresses for each Class B network.

3. Class C Addresses



For Class C networks, the first two bits of the first octet always turned on, but the third bit can never be on (110). Following the same process as the previous classes, convert from binary to decimal to find the range. Here's the range for a Class C network:

$$11000000 = 192$$

$$11011111 = 223$$

So, if you see an IP address that starts at 192 and goes to 223, you'll know it is a Class C IP address.

The first 3 bytes of a Class C network address are dedicated to the network portion of the address, with only one measly byte remaining for the node address. The format is:

NETWORK.NETWORK.NETWORK.HOST

Using the example IP address 192.168.100.102, the network address is 192.168.100, and the node address is 102.

In a Class C network address, the first three bit positions are always the binary 110. The calculation is such: 3 bytes, or 24 bits, minus 3 reserved positions, leaves 21 positions. Hence, there are 221, or 2.097.152, possible Class C networks. Each unique Class C network has 1 byte to use for 'node addresses'. This leads to 28 or 256, minus the two reserved patterns of all 0s and all 1s, for a total of 254 node addresses for each Class C network.

To summarize the rules and equations for Class A, B, and C addressing:

A host ID cannot be all 0s.

A host ID cannot be all 1s.

To determine the number of networks that can be created, use the formula 2^N , where N is the number of bits in the network portion of the address.



To determine the number of hosts that can be created, use the formula $2^N - 2$, where N is the number of bits in the host portion of the address.

4. Class D Addresses

The Class D address space is a radical departure from the first three classes. Unlike the previous three classes (A, B and C classes), this class does not adhere to the convention of dividing the bit string into network and host address subcomponents. This makes it rather difficult to use in uniquely identifying networked hosts in an internetwork. Quite simply, it cannot define a network address. This is by design; this uniquely flat address space is not used for endpoint addressing. Instead, ***it serves as a code that lets a host send single stream of IP packets to numerous destination machines simultaneously.***

In other words, these addresses are called ***multicast*** addresses, and they are invalid for any workstation or host to use. ***[Multicast: A communication between a single sender and multiple receivers on a network. No one host can have this address, but several can receive data by listening to it.]***

The purpose of a multicast address is *to enable a server somewhere to send data to a Class D address* that no one host has so that several hosts can listen to that address at the same time. When you are watching TV on the Internet or listening to the radio on the Internet, your computer is listening to a Class D address. No server is sending data directly to your workstation; instead, a server is sending data to the multicast address. Any host can use software to listen for data at that address, and many hosts can be listening at once.

The first 4 bits of a Class D address must be **1110**. Binary mathematics dictates that, given the location of this 0, the lowest address in this class can be 11100000, or $128 + 64 + 32 = 224$. The highest mathematically possible address, given this constraint in the first octet, is 11101111, or $128 + 64 + 32 + 8 + 4 + 2 + 1 = 255$.



$1 = 239$. Thus, Class D multicast group addresses are from 224.0.0.0 to 239.255.255.255 .

5. Class E Addresses

The last class of addresses is Class E. Class E addresses range from 240 to 255 in the first octet, and the 5 leftmost bits are **11110**. Class E addresses are reserved addresses and are invalid host addresses. They are used for experimental purposes by the IETF [Internet Engineering Task Force. *A governing body of the Internet*].

IP Address Classes

Address Class	1st octet range (decimal)	1st octet bits (green bits do not change)	Network(N) and Host(H) parts of address	Default subnet mask (decimal and binary)	Number of possible networks and hosts per network
A	1-127**	00000000- 01111111	N.H.H.H	255.0.0.0	128 nets (2^7) 16,777,214 hosts per net (2^{24-2})
B	128-191	10000000- 10111111	N.N.H.H	255.255.0.0	16,384 nets (2^{14}) 65,534 hosts per net (2^{16-2})
C	192-223	11000000- 11011111	N.N.N.H	255.255.255.0	2,097,150 nets (2^{21}) 254 hosts per net (2^{8-2})
D	224-239	11100000- 11101111	NA (multicast)		
E	240-255	11110000- 11111111	NA (experimental)		

** All zeros (0) and all ones (1) are invalid hosts addresses.



Class	Identifiers	Range	Network Bits	Networks Available	Host bits	Hosts Available
A	0/	1 through 126	8 [7 bits (first byte)]	126	24 bits (last three bytes)	16,777,214
B	10/	128 through 191	16 [14 bits (first two bytes)]	16,384	16 bits (last two bytes)	65,534
C	110/	192 through 223	24 [21 bits (first three bytes)]	2,079,152	8 bits (last byte)	254
D	1110/	224 through 239	Ranges from 224.0.0.0 through 239.255.255.255 →	(268,435,456)		
E	11110/	240 through 255	Reserved →	(268,435,456)		

All addresses in IPv4 → 4,294,967,296

Addresses in class A → 2,113,928,964 Addresses in class B → 1,073,709,056 Addresses in class C → 528,104,608

All addresses are placed in a particular class based on the decimal values of their first octets. In the first octet, an IP address can start with a decimal value between 1 and 255. The system of class addresses has been set up to help ensure assignment of unique IP addresses.

Protocols implemented at the Network layer include:

- Internet Protocol version 4 (IPv4)
- Internet Protocol version 6 (IPv6)
- Novell Internetwork Packet Exchange (IPX)
- AppleTalk
- Connectionless Network Service (CLNS/DECNet)

Notation

There are three common notations to show an IPv4 address:

- Binary Notation (Base 2),
- Dotted-Decimal Notation (Base 256), and
- Hexadecimal Notation (Base 16).



The most prevalent, however, is base 256.

- **Binary Notation:** In binary notation, the IPv4 address is displayed as 32 bits. Each octet is often referred to as a byte. So it is common to hear an IPv4 address referred to as a 32-bit address or a 4-byte address. The following is an example of an IPv4 address in binary notation:

01110101 10010101 00011101 00000010

- **Dotted-Decimal Notation:** To make the IPv4 address more compact and easier to read, Internet addresses are usually written in decimal form with a decimal point (dot) separating the bytes. The following is the Dotted-Decimal notation of the above address: 117.149.29.2
- **Hexadecimal Notation:** We sometimes see an IPv4 address in **hexadecimal notation**. Each hexadecimal digit is equivalent to four bits. This means that a 32-bit address has 8 hexadecimal digits. This notation is often used in network programming.

Within the address range of each IPv4 network, we have three types of addresses:

Network address - The address by which we refer to the network.

Broadcast address - A special address used to send data to all hosts in the network.

Host addresses - The addresses assigned to the end devices in the network.

Each network has an Internet address. Each network also must know the address of every other network with which it communicates.

After the network is identified, the specific host or node must be specified. A unique host address for the particular network is added to the end of the IP address.

Network Masks



A network mask helps you know which portion of the address identifies the network and which portion of the address identifies the node. Class A, B, and C networks have default masks, also known as natural masks, as shown here:

Class A: 255.0.0.0

Class B: 255.255.0.0

Class C: 255.255.255.0

Private Addresses

The private address blocks are:

10.0.0.0 to 10.255.255.255 (10.0.0.0 /8) for class A

172.16.0.0 to 172.31.255.255 (172.16.0.0 /12) for class B

192.168.0.0 to 192.168.255.255 (192.168.0.0 /16) For class C

2. Classless Addressing

To overcome address depletion and give more organizations access to the Internet, classless addressing was designed and implemented. In this scheme, there are no classes, but the addresses are still granted in blocks.

Classless Inter-Domain Routing (**CIDR**) is an IP addressing scheme that was developed after the class system of A, B, C, D, and E [*uses a slash followed by a number to highlight the network portion of an address instead of using a subnet mask*].

The traditional class system considers an IP address as four octets with the network portion of the address highlighted by a subnet mask. The standard network portion is the first octet, the first two octets, or the first three octets. CIDR addressing still represents IP addresses in the traditional dotted decimal notation, but highlights the network portion with a slash followed by a number. For example:

192.168.3.15/26



172.21.165.1/19

The number after the slash is the number of bits that represent the network portion of the IP address. *CIDR was developed to increase the efficiency of address allocation and to alleviate overloaded Internet routers.*



Lecture 8:

IPv4 Addressing

Within the address range of each IPv4 network, we have three types of addresses:

Network address - The address by which we refer to the network.

Broadcast address - A special address used to send data to all hosts in the network.

Host addresses - The addresses assigned to the end devices in the network.

Each network has an Internet address. Each network also must know the address of every other network with which it communicates.

After the network is identified, the specific host or node must be specified. A unique host address for the particular network is added to the end of the IP address.

Network Masks

A network mask helps you know which portion of the address identifies the network and which portion of the address identifies the node. Class A, B, and C networks have default masks, also known as natural masks, as shown here:

Class A: 255.0.0.0

Class B: 255.255.0.0

Class C: 255.255.255.0

Private Addresses

The private address blocks are:

10.0.0.0 to 10.255.255.255 (10.0.0.0 /8) for class A

172.16.0.0 to 172.31.255.255 (172.16.0.0 /12) for class B

192.168.0.0 to 192.168.255.255 (192.168.0.0 /16) For class C



1. Classless Addressing

To overcome address depletion and give more organizations access to the Internet, classless addressing was designed and implemented. In this scheme, there are no classes, but the addresses are still granted in blocks.

Classless Inter-Domain Routing (CIDR) is an IP addressing scheme that was developed after the class system of A, B, C, D, and E [*uses a slash followed by a number to highlight the network portion of an address instead of using a subnet mask*].

The traditional class system considers an IP address as four octets with the network portion of the address highlighted by a subnet mask. The standard network portion is the first octet, the first two octets, or the first three octets. CIDR addressing still represents IP addresses in the traditional dotted decimal notation, but highlights the network portion with a slash followed by a number. For example:

192.168.3.15/26

172.21.165.1/19

The number after the slash is the number of bits that represent the network portion of the IP address. **CIDR was developed to increase the efficiency of address allocation and to alleviate overloaded Internet routers.**

Computer Network Fundamentals

Lecture9: **Prefix and Subnet Mask**

Assistant Prof Dr. Suad Abdulelah Alasadi

IP Address

- An IP address is a 32-bit, two-level hierarchical number. It is uniquely defined by a network layer address.
- It is hierarchical because the first portion of the address represents the **network**, and the second portion of the address represents the **node (or host)**.
- The 32 bits are grouped into four octets, with 8 bits per octet. The value of each octet ranges from 0 to 255 decimal, or 00000000 to 11111111 binary. IP addresses are usually written in dotted decimal notation, which means that each octet is written in decimal notation and dots are placed between the octets.

Network Prefixes

- When an IPv4 network address is expressed, you add a prefix length to the network address.
- This prefix length is the number of bits in the address that gives the network portion. This prefix length is written in slash format. That is a forward slash (/) followed by the number of network bits.

Network Prefixes(Con.)

- For example, in **172.16.4.0 /24**, the **/24** is the prefix length.

This tells you that the first **24** bits are the **network address**. The remaining **8 bits**, the last octet, are the **host portion**.

Network Prefixes (Con.)

- Depending on **the number of hosts** on the network, the prefix assigned can be different. Having a different prefix number changes the host range and broadcast address for each network.
- Notice that the network addresses in **Table 1** remain the same, but the host range and the broadcast address are different for the different prefix lengths. You can also see that the number of hosts that can be addressed on the network changes as well.

Network Prefixes (Con.)

Table 1: Using Different Prefixes for the 172.16.4.0

Network Network	Address	Host Range	Broadcast Address
172.16.4.0 /24	172.16.4.0	172.16.4.1–172.16.4.254	172.16.4.255
172.16.4.0 /25	172.16.4.0	172.16.4.1–172.16.4.126	172.16.4.127
172.16.4.0 /26	172.16.4.0	172.16.4.1–172.16.4.62	172.16.4.63
172.16.4.0 /27	172.16.4.0	172.16.4.1–172.16.4.30	172.16.4.31

Subnet Mask

- “How do the **network devices** know how many bits are the network portion and how many bits are the host portion?”
- The answer to this question is the **subnet mask**.
- **The subnet mask is used to Define the Network and Host Portions of the Address.**

Prefix and subnet

- The **prefix and the subnet mask** are different ways of representing the same information: the **network portion** of an address.
- The **prefix length** tells you the number of bits in the address that are the network portion in a way that is easier to communicate to humans.
- The **subnet mask** is used in data networks to define this network portion for the devices.

Subnet Mask(con.)

- The **subnet mask is** a **32-bit value** used with the IPv4 address that specifies the **network portion** of the address to the network devices.
- The subnet mask uses 1s and 0s to indicate which bits of the IPv4 address are network bits and which bits are hosts bits.
- The subnet mask is expressed in the same dotted decimal format as the IPv4 address.

Subnet Mask(con.)

- For example A /24 prefix represents a **subnet mask** of **255.255.255.0**
- **(11111111.11111111.11111111.00000000).**
- The first three octets, the higher-order 24 bits, are all 1s. The remaining low-order bits of the subnet mask are 0s, indicating the host address within the network.

Subnet Mask(con.)

- For example, examine the host **172.16.4.35/27** shown in Table 2.

	Dotted Decimal				Binary Octets			
Host	172	16	4	35	10101100	00010000	00000100	00100011
Mask	255	255	255	224	11111111	11111111	11111111	11100000
Network	172	16	4	32	10101100	00010000	00000100	00100000

Subnet Mask(con.)

Mask (Decimal)	Mask (Binary)	Network Bits	Host Bits
0	00000000	0	8
128	10000000	1	7
192	11000000	2	6
224	11100000	3	5
240	11110000	4	4
248	11111000	5	3
252	11111100	6	2
254	11111110	7	1
255	11111111	8	0

