

# Computer Network

Lecture taken by

Dharmendra Kumar

(Associate Professor)

United College of Engineering and Research, Prayagraj

# Syllabus

# Syllabus

## Unit-1:

**Introductory Concepts:** Goals and applications of networks, Categories of networks, Organization of the Internet, ISP, Network structure and architecture (layering principles, services, protocols and standards), The OSI reference model, TCP/IP protocol suite, Network devices and components.

**Physical Layer:** Network topology design, Types of connections, Transmission media, Signal transmission and encoding, Network performance and transmission impairments, Switching techniques and multiplexing.

# Syllabus

## Unit-2:

**Data Link Layer:** Framing, Error Detection and Correction, Flow control (Elementary Data Link Protocols, Sliding Window protocols). **Medium Access Control and Local Area Networks:** Channel allocation, Multiple access protocols, LAN standards, Link layer switches & bridges (learning bridge and spanning tree algorithms)

## Unit-3:

**Network Layer:** Point-to-point networks, Logical addressing, Basic internetworking (IP, CIDR, ARP, RARP, DHCP, ICMP), Routing, forwarding and delivery, Static and dynamic routing, Routing algorithms and protocols, Congestion control algorithms, IPv6.

# Syllabus

## Unit-4:

Transport Layer: Process-to-process delivery, Transport layer protocols (UDP and TCP), Multiplexing, Connection management, Flow control and retransmission, Window management, TCP Congestion control, Quality of service.

## Unit-5:

Application Layer: Domain Name System, World Wide Web and Hyper Text Transfer Protocol, Electronic mail, File Transfer Protocol, Remote login, Network management, Data compression, Cryptography – basic concepts.

# Books

1. Behrouz Forouzan, "Data Communication and Networking", McGraw Hill
2. Andrew Tanenbaum "Computer Networks", Prentice Hall.
3. William Stallings, "Data and Computer Communication", Pearson.
4. Kurose and Ross, "Computer Networking-A Top-Down Approach", Pearson.

# Course Outcome (CO's)

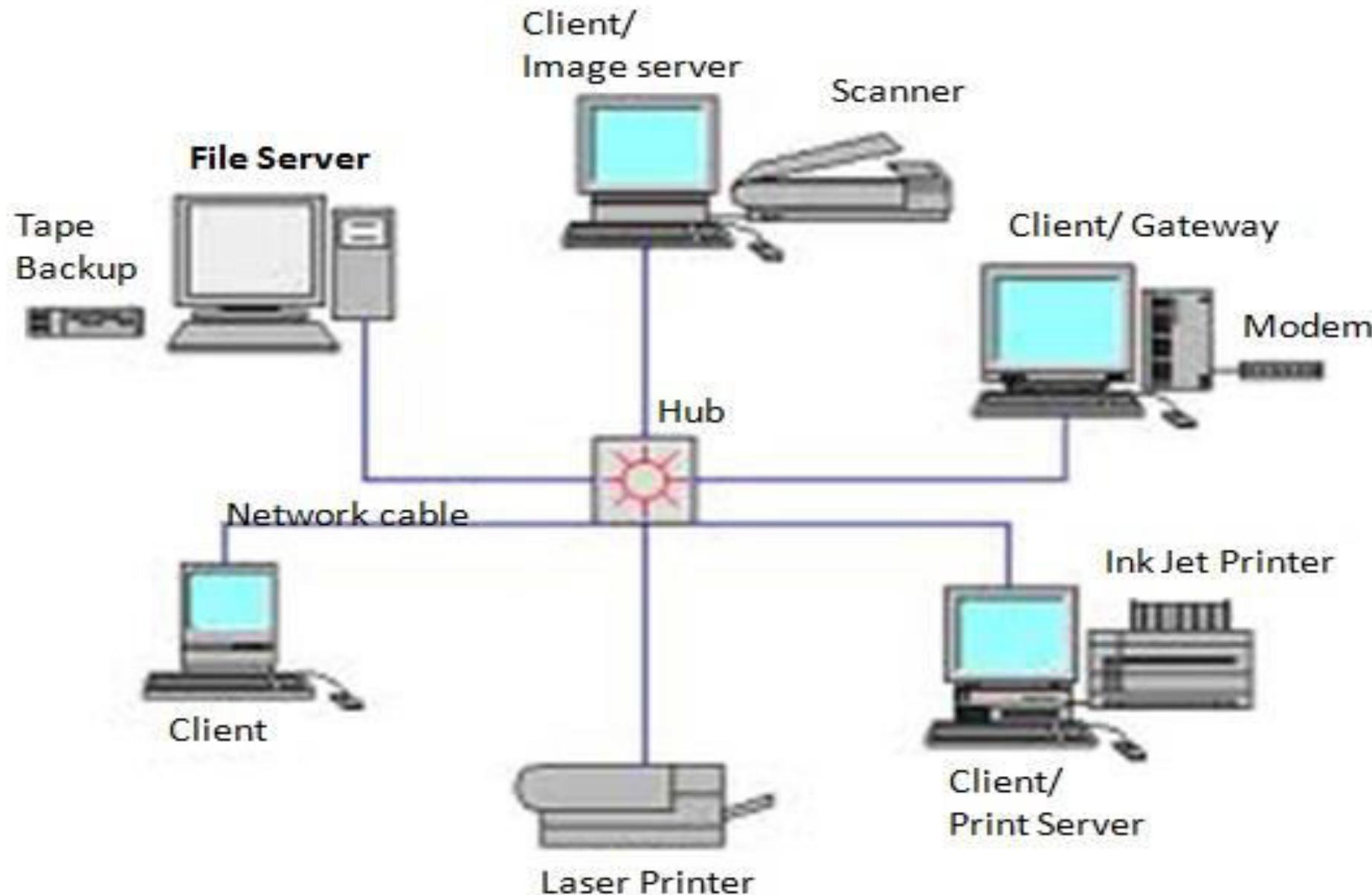
<b>CO1</b>	Explain basic concepts, OSI reference model, services and role of each layer of OSI model and TCP/IP, networks devices and transmission media, Analog and digital data transmission
<b>CO2</b>	Apply channel allocation, framing, error and flow control techniques.
<b>CO3</b>	Describe the functions of Network Layer i.e. Logical addressing, subnetting & Routing Mechanism.
<b>CO4</b>	Explain the different Transport Layer function i.e. Port addressing, Connection Management, Error control and Flow control mechanism.
<b>CO5</b>	Explain the different protocols used at application layer i.e. HTTP, SNMP, SMTP, FTP, TELNET and VPN

# **Unit-1**

# Computer Network

- ❖ A computer network is a set of devices (often referred to as nodes) connected by communication links. A node can be a computer, printer, or any other device capable of sending and/or receiving data generated by other nodes on the network.
- ❖ “Computer network” to mean a collection of autonomous computers interconnected by a single technology. Two computers are said to be interconnected if they are able to exchange information.

# Computer Network



# Computer Network

- ❖ Networks come in many sizes, shapes and forms. They are usually connected together to make larger networks.
- ❖ Internet being the most well-known example of a network of networks.

# Uses of Computer Network

## 1. Business Applications

- ❖ **Information sharing:** To distribute information throughout the company
- ❖ **Resource sharing:** Sharing physical resources such as printers, and tape backup systems
- ❖ **Client-Server model:** It is widely used and forms the basis of much network usage.
- ❖ **E-mail:** Employees generally use for a great deal of daily communication.
- ❖ **Voice over IP (VoIP):** Telephone calls between employees may be carried by the computer network instead of by the phone company. This technology is called IP telephony or Voice over IP (VoIP) when Internet technology is used.
- ❖ **Desktop sharing:** Remote workers see and interact with a graphical computer screen
- ❖ **E-commerce:** Doing business electronically, especially with customers and suppliers. It has grown rapidly in recent years.

# Uses of Computer Network

## 2. Home Applications

- ❖ Person-to-Person communication
- ❖ Electronic commerce
- ❖ Entertainment (game playing)

## 3. Mobile Users

- ❖ Text messaging or texting
- ❖ Smart phones,
- ❖ GPS (Global Positioning System)
- ❖ M-commerce
- ❖ NFC (Near Field Communication)

## 4. Social Issues

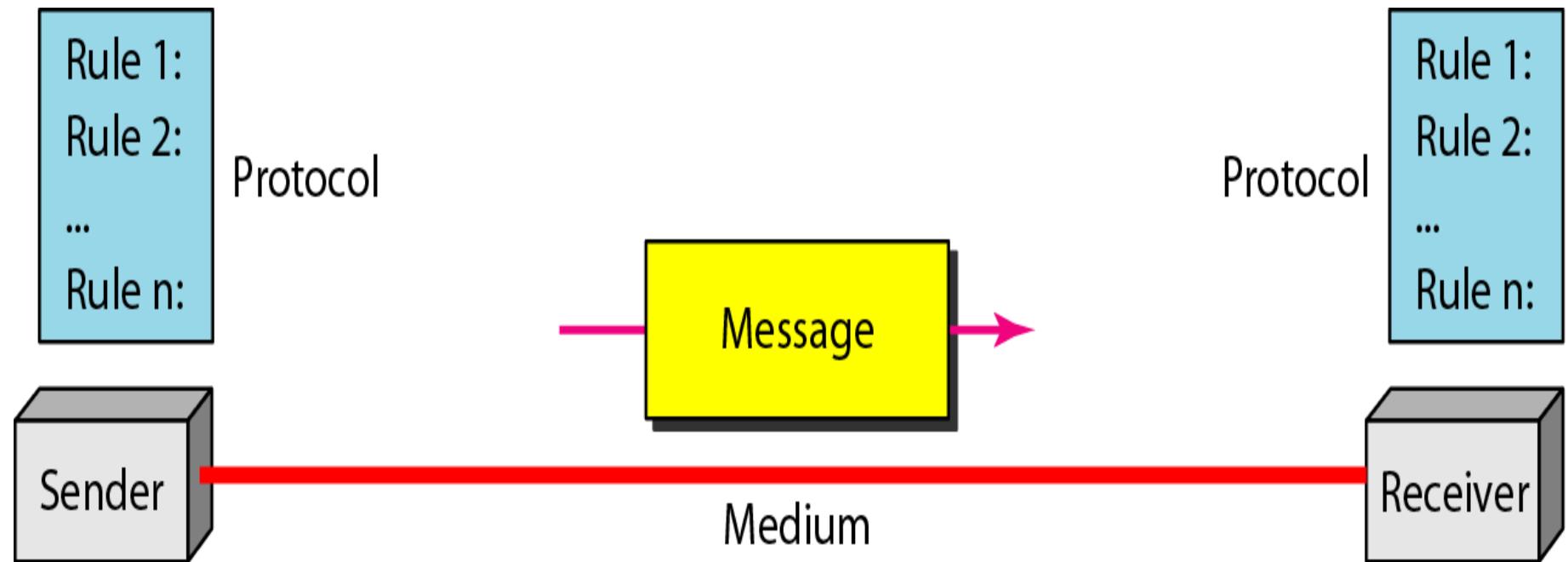
Social networks, message boards, content sharing sites, and a host of other applications allow people to share their views with like-minded individuals.

# Data Communication System

A data communication system has **five** components:-

1. **Message:** The message is the information (data) to be communicated. Popular forms of information include text, numbers, pictures, audio, and video.
2. **Sender:** The sender is the device that sends the data message. It can be a computer, workstation, telephone handset, video camera, and so on.
3. **Receiver:** The receiver is the device that receives the message. It can be a computer, workstation, telephone handset, television, and so on.
4. **Transmission medium:** The transmission medium is the physical path by which a message travels from sender to receiver. Some examples of transmission media include twisted-pair wire, coaxial cable, fiber-optic cable, and radio waves.
5. **Protocol:** A protocol is a set of rules that govern data communications. It represents an agreement between the communicating devices. Without a protocol, two devices may be connected but not communicating, just as a person speaking French cannot be understood by a person who speaks only Japanese.

# Data Communication System



# Data Representation

- ❖ Text
- ❖ Numbers
- ❖ Images
- ❖ Audio
- ❖ Video

# Effectiveness of a Data Communication System

The effectiveness of a data communications system depends on **four** fundamental characteristics: delivery, accuracy, timeliness, and jitter.

**1. 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.

**2. Accuracy:** The system must deliver the data accurately. Data that have been altered in transmission and left uncorrected are unusable.

# Effectiveness of a Data Communication System

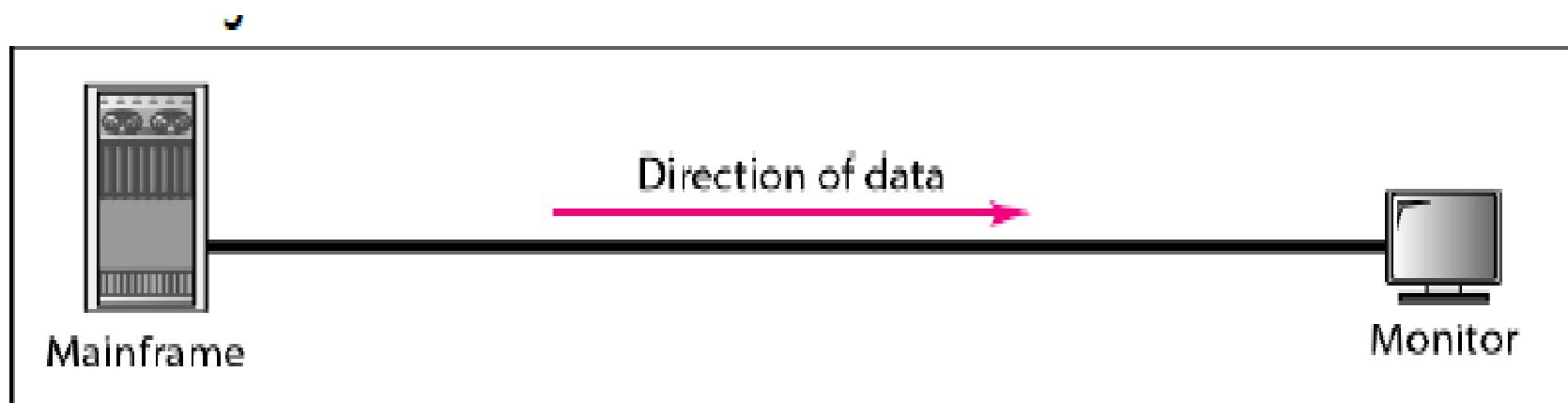
3. **Timeliness:** The system must deliver data in a timely manner. Data delivered late are useless. In the case of video and audio, timely delivery means delivering data as they are produced, in the same order that they are produced, and without significant delay. This kind of delivery is called real-time transmission.
4. **Jitter:** Jitter refers to the variation in the packet arrival time. It is the uneven delay in the delivery of audio or video packets. For example, let us assume that video packets are sent every 30 ms. If some of the packets arrive with 30 ms delay and others with 40 ms delay, then an uneven quality in the video is the result.

# Data Flow

Communication between two devices can be simplex, half-duplex, or full-duplex.

## Simplex Communication

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.

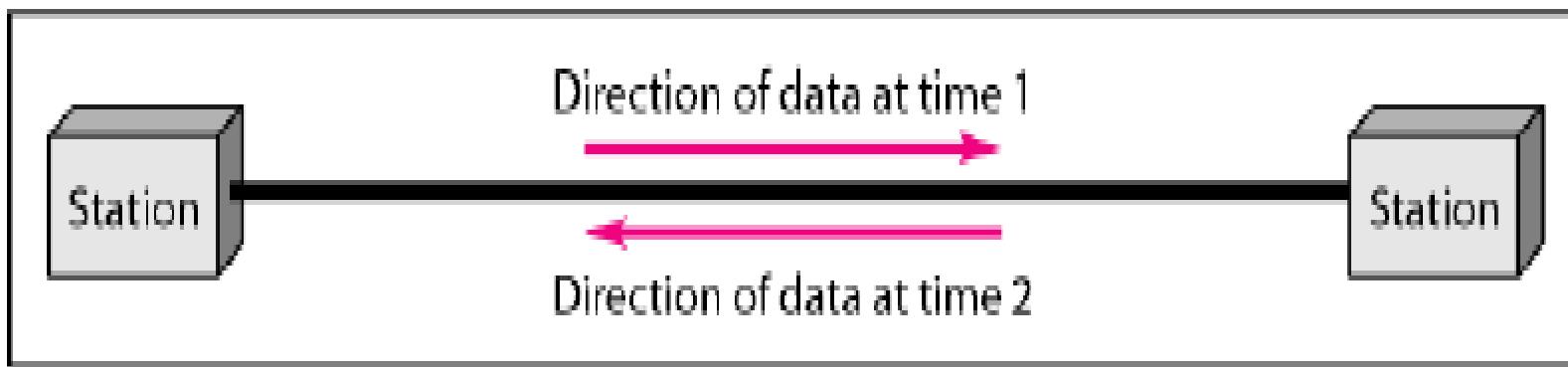


Keyboards and traditional monitors are examples of simplex devices.

# Data Flow

## Half-Duplex Communication

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

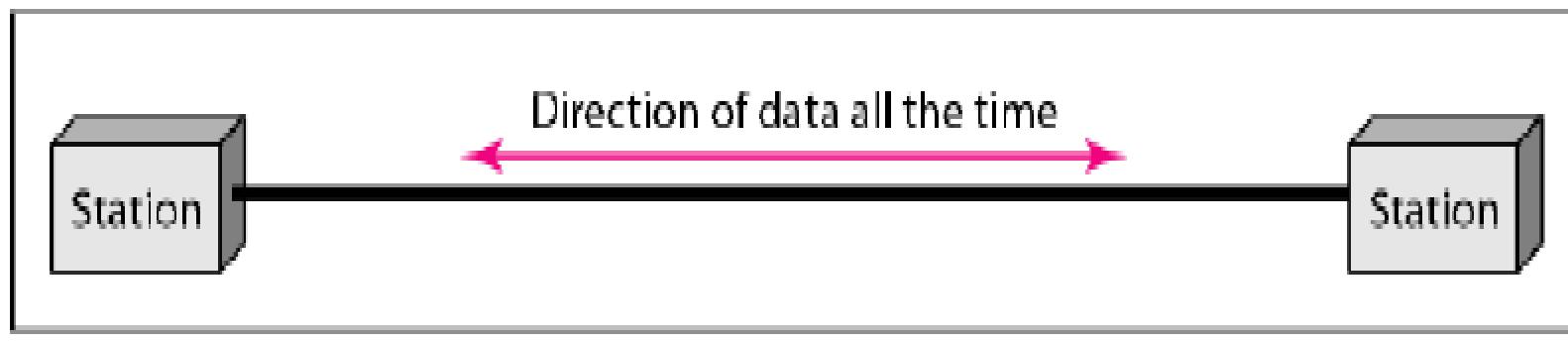


Walkie-talkies and CB (citizens band) radios are both half-duplex systems.

# Data Flow

## Full-Duplex Communication

In full-duplex, both stations can transmit and receive simultaneously. One common example of full-duplex communication is the telephone network. When two people are communicating by a telephone line, both can talk and listen at the same time. The full-duplex mode is used when communication in both directions is required all the time.



# Network Criteria

A network must be able to meet a certain number of criteria. The most important of these are performance, reliability, and security.

## Performance

- ❖ Performance can be measured in many ways, including transit time and response time.
  - **Transit time** is the amount of time required for a message to travel from one device to another.
  - **Response time** is the elapsed time between an inquiry and a response.
- ❖ The performance of a network depends on a number of factors, including the number of users, the type of transmission medium, the capabilities of the connected hardware, and the efficiency of the software.
- ❖ Performance is often evaluated by two networking metrics: **throughput and delay**. We often need more throughput and less delay.
- ❖ However, these two criteria are often contradictory. If we try to send more data to the network, we may increase throughput but we increase the delay because of traffic congestion in the network.

# Network Criteria

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

**Security:** Network security issues include protecting data from unauthorized access, protecting data from damage and development, and implementing policies and procedures for recovery from breaches and data losses.

# Type of Connection

A network is two or more devices connected through links. A link is a communications pathway that transfers data from one device to another. There are two possible types of connections: point-to-point and multipoint.

## Point-to-Point Connection

A point-to-point connection provides a dedicated link between two devices. The entire capacity of the link is reserved for transmission between those two devices. Most point-to-point connections use an actual length of wire or cable to connect the two ends, but other options, such as microwave or satellite links, are also possible.

When you change television channels by infrared remote control, you are establishing a point-to-point connection between the remote control and the television's control system.

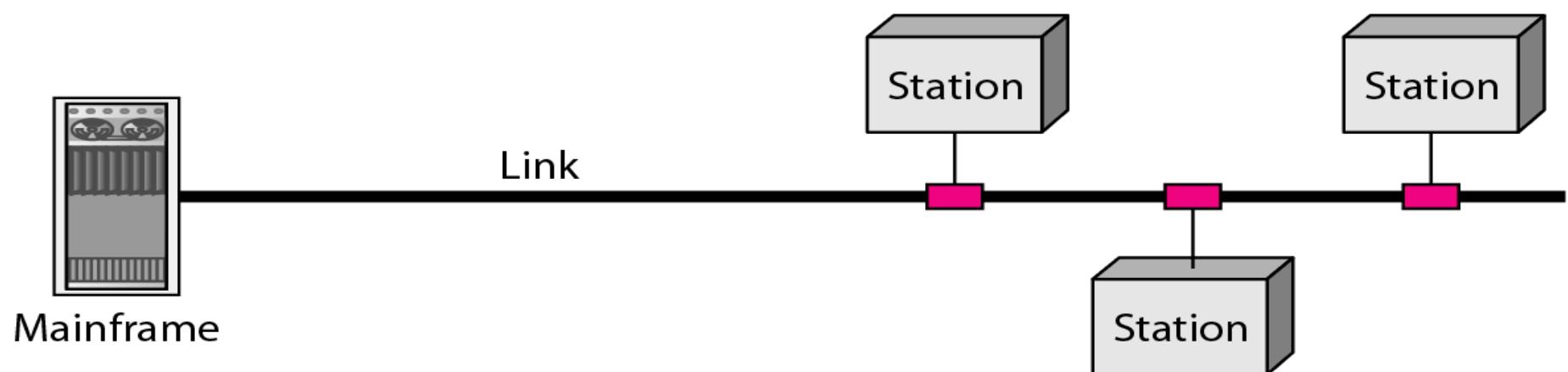


# Type of Connection

## Multipoint Connection

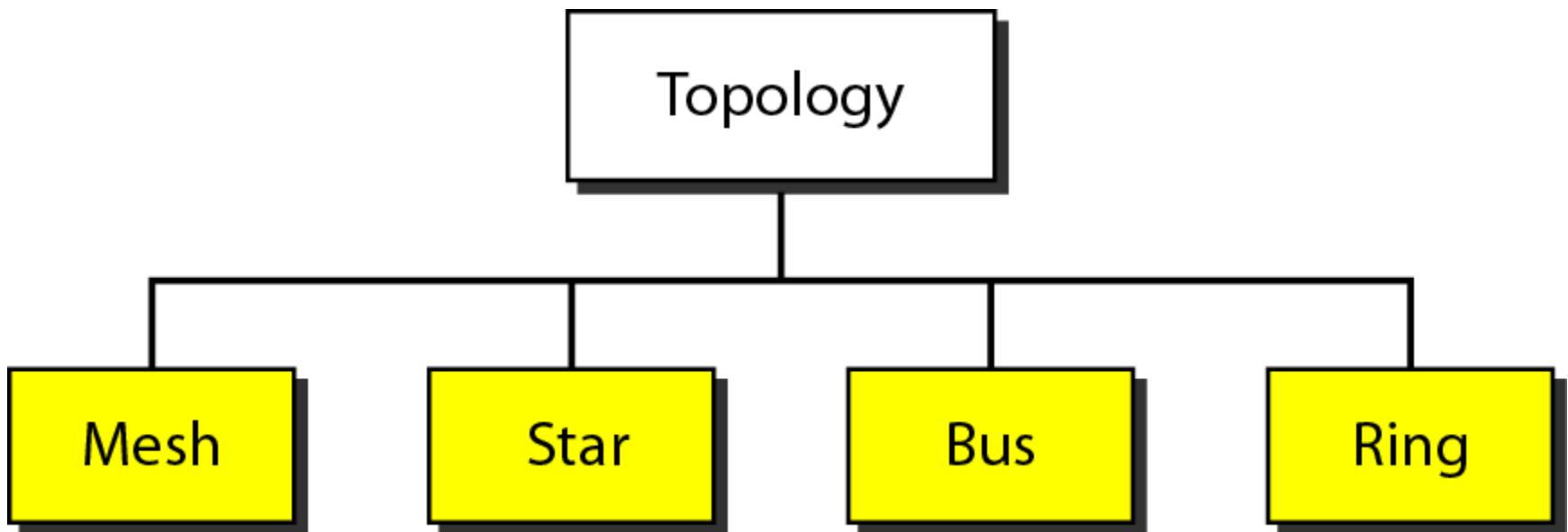
A multipoint (also called multi-drop) connection is one in which more than two specific devices share a single link.

In a multipoint environment, the capacity of the channel is shared, either spatially or temporally. If several devices can use the link simultaneously, it is a spatially shared connection. If users must take turns, it is a timeshared connection.



# Network Topology

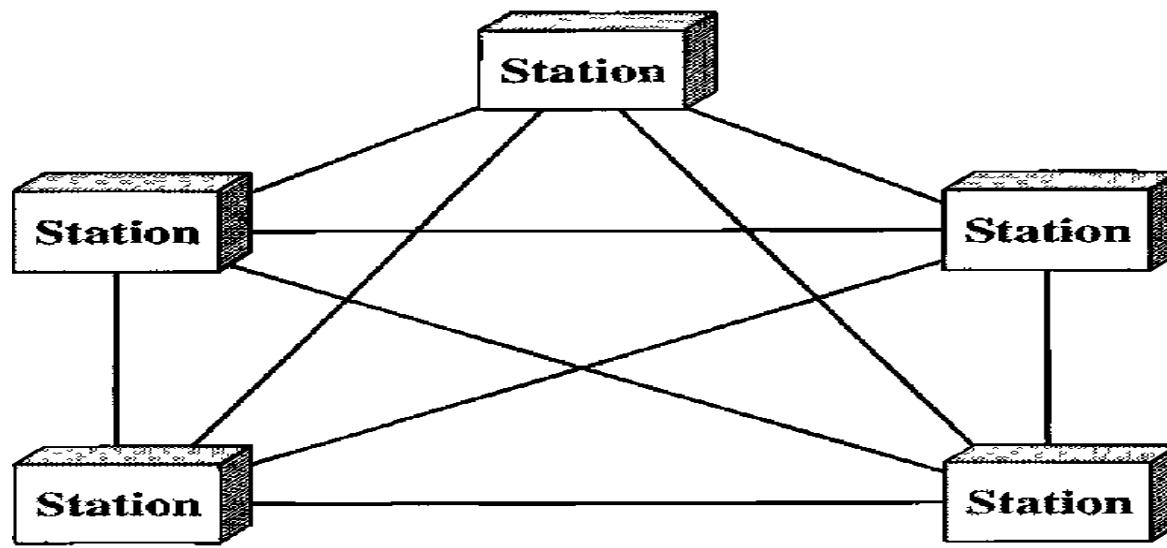
- ❖ The topology of a network is the geometric representation of the relationship of all the links and linking devices (usually called nodes) to one another.
- ❖ There are **four** basic topologies possible: mesh, star, bus, and ring.



# Mesh Topology

## Mesh Topology:

- ❖ In a mesh topology, every device has a dedicated point-to-point link to every other device. The term dedicated means that the link carries traffic only between the two devices it connects.
- ❖ The number of **physical links** in a fully connected mesh network with **n** nodes =  **$n(n-1)/2$** .



# Mesh Topology

## Advantages of a mesh topology

1. The use of dedicated links guarantees that each connection can carry its own data load, thus eliminating the traffic problems that can occur when links must be shared by multiple devices.
2. A mesh topology is robust. If one link becomes unusable, it does not incapacitate the entire system.
3. There is the advantage of privacy or security. When every message travels along a dedicated line, only the intended recipient sees it. Physical boundaries prevent other users from gaining access to messages.
4. Point-to-point links make fault identification and fault isolation easy. Traffic can be routed to avoid links with suspected problems. This facility enables the network manager to discover the precise location of the fault and aids in finding its cause and solution.

# Mesh Topology

## Disadvantages of a mesh topology

1. Since every device must be connected to every other device, therefore installation and reconnection are difficult.
2. Second, the sheer bulk of the wiring can be greater than the available space (in walls, ceilings, or floors) can accommodate.
3. Finally, the hardware required to connect each link (I/O ports and cable) can be prohibitively expensive.

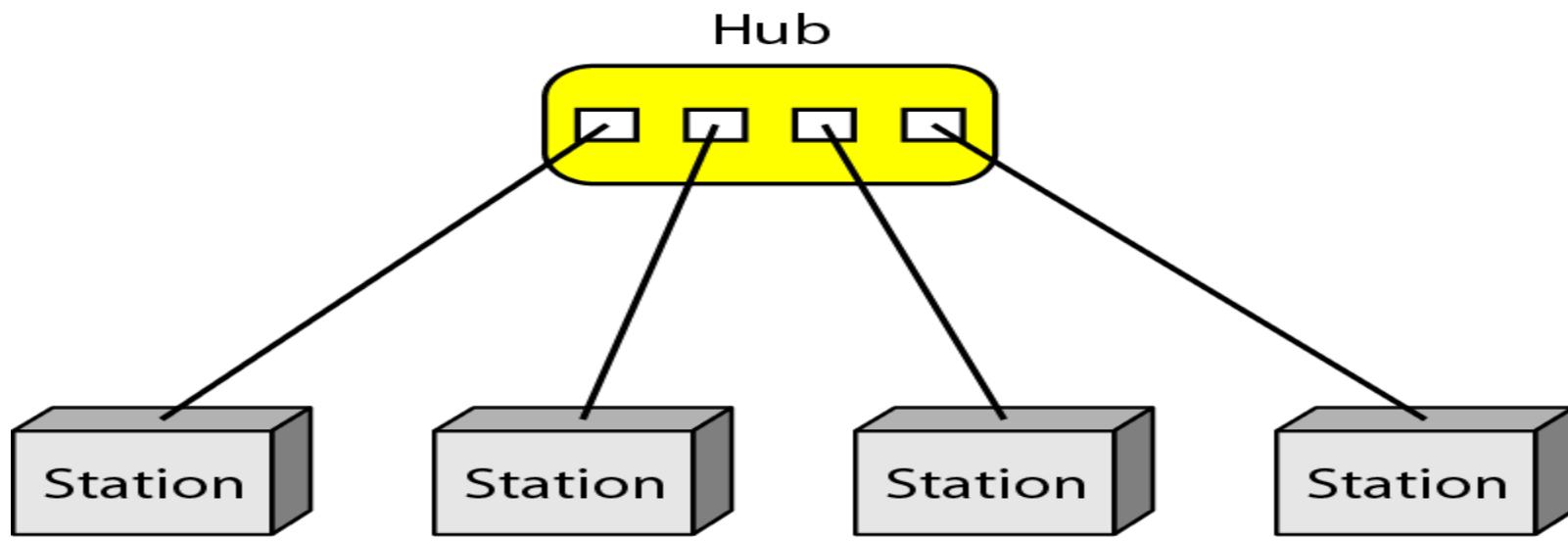
# Mesh Topology

## Uses of mesh topology

One practical example of a mesh topology is the connection of telephone regional offices in which each regional office needs to be connected to every other regional office.

# Star Topology

- ❖ In a star topology, each device has a dedicated point-to-point link only to a central controller, usually called a hub.
- ❖ The devices are not directly linked to one another.
- ❖ Unlike a mesh topology, a star topology does not allow direct traffic between devices.
- ❖ The controller acts as an exchange: If one device wants to send data to another, it sends the data to the controller, which then relays the data to the other connected device.



# Star Topology

## Advantages of Star topology

1. A star topology is less expensive than a mesh topology.
2. In a star, each device needs only one link and one I/O port to connect it to any number of others. This factor also makes it easy to install and reconfigure.
3. Other advantages include robustness. If one link fails, only that link is affected. All other links remain active. This factor also lends itself to easy fault identification and fault isolation. As long as the hub is working, it can be used to monitor link problems and bypass defective links.

# Star Topology

## Disadvantages of Star topology

A disadvantage of a star topology is the dependency of the whole topology on one single point, the hub. If the hub goes down, the whole system is dead.

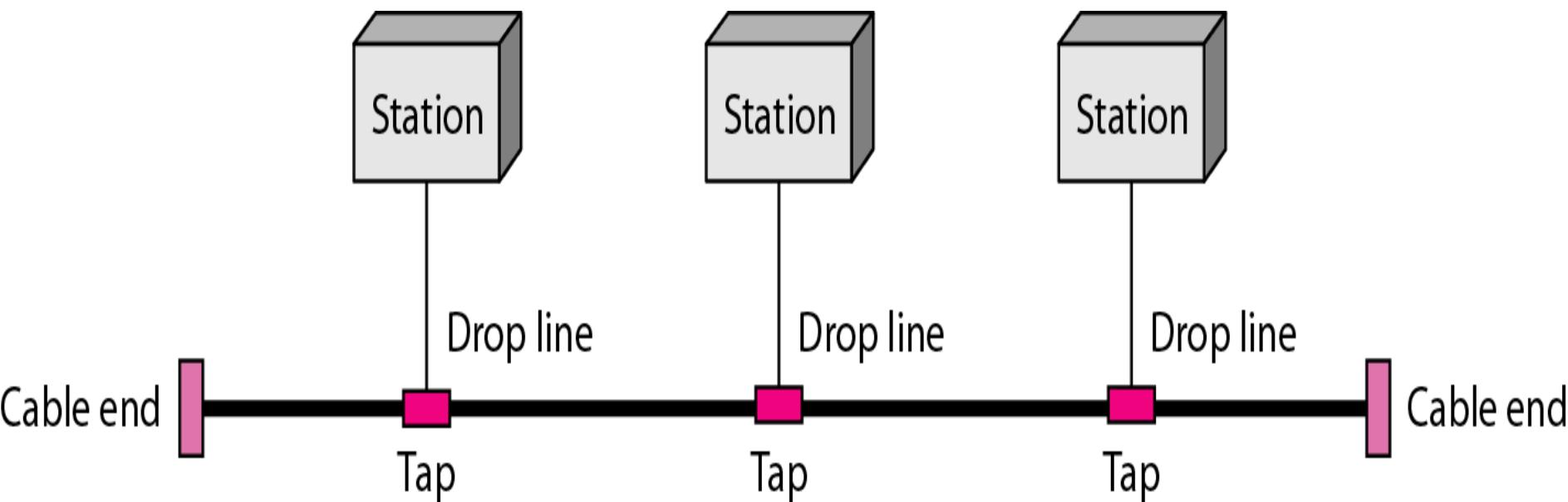
## Uses:

The star topology is used in local-area networks (LANs).

High-speed LANs often use a star topology with a central hub.

# Bus Topology

Bus topology uses multipoint connection link. One long cable acts as a **backbone** to link all the devices in a network.



# Bus Topology

- Nodes( systems) are connected to the backbone cable by **drop lines** and **taps**.
- A **drop line** is a connection running between the device and the main cable.
- A **tap** is a connector that either splices into the main cable or punctures the sheathing of a cable to create a contact with the metallic core.
- As a signal travels along the backbone, some of its energy is transformed into heat. Therefore, it becomes weaker and weaker as it travels farther and farther.
- For this reason there is a limit on the number of taps a bus can support and on the distance between those taps.

# Bus Topology

## Advantages of bus topology

### ❖ Ease of installation

Backbone cable can be laid along the most efficient path, then connected to the nodes by drop lines of various lengths.

- ❖ It works well when you have a small network.
- ❖ It's the easiest network topology for connecting computers or peripherals in a linear fashion.
- ❖ **A bus topology uses less cabling than mesh or star topologies.**

In a star, for example, four network devices in the same room require four lengths of cable reaching all the way to the hub. In a bus, this redundancy is eliminated. Only the backbone cable stretches through the entire facility. Each drop line has to reach only as far as the nearest point on the backbone.

# Bus Topology

## Disadvantages of bus topology

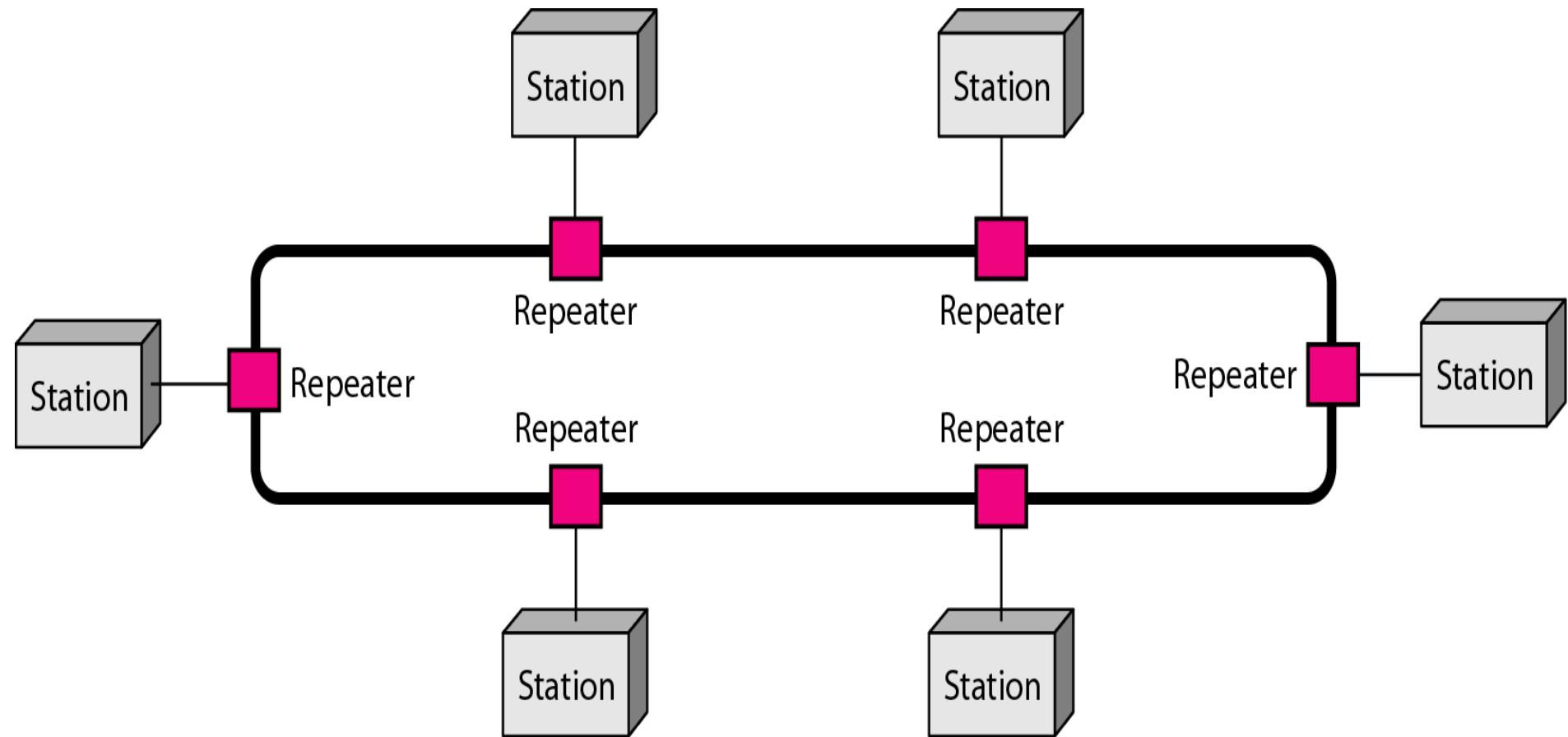
- ❖ It can be difficult to identify the problems if the whole network goes down.
- ❖ It can be hard to troubleshoot individual device issues.
- ❖ Bus topology is not great for large networks.
- ❖ Terminators are required for both ends of the main cable.
- ❖ Additional devices slow the network down.
- ❖ If a main cable is damaged, the network fails or splits into two.

# Bus Topology

## Uses of bus topology

- ❖ Bus topology was the one of the first topologies used in the design of early local area networks.
- ❖ Ethernet LANs can use a bus topology, but they are less popular now.

# Ring Topology



# Ring Topology

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

# Ring Topology

## Advantages of ring topology

- ❖ A ring topology is relatively easy to install and reconfigure.

Each device is linked to only its immediate neighbors (either physically or logically). To add or delete a device requires changing only two connections.

- ❖ Fault isolation is simplified.

Generally in a ring, a signal is circulating at all times. If one device does not receive a signal within a specified period, it can issue an alarm. The alarm alerts the network operator to the problem and its location.

# Ring Topology

## Disadvantages of ring topology

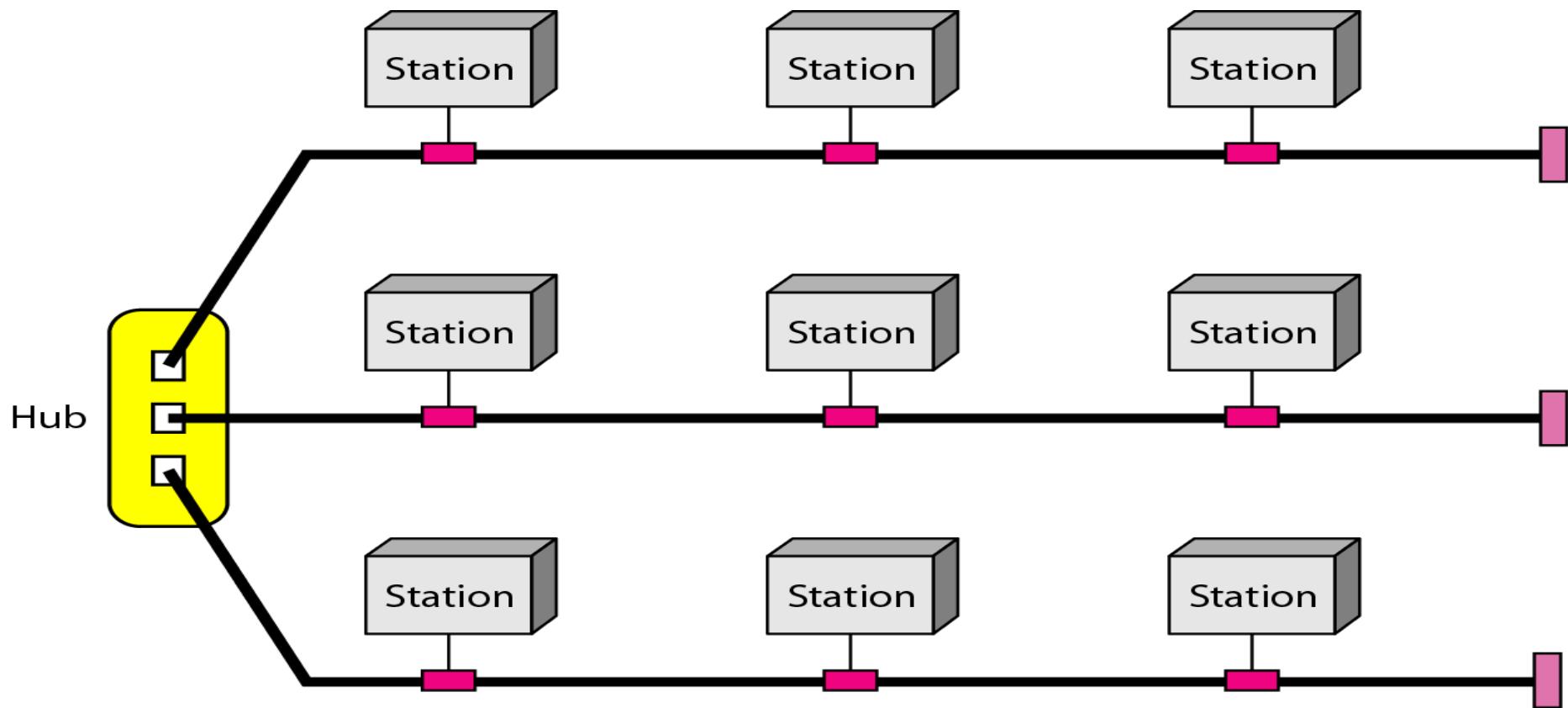
- ❖ Unidirectional traffic can be a disadvantage.
- ❖ In a simple ring, a break in the ring (such as a disabled station) can disable the entire network. This weakness can be solved by using a dual ring or a switch capable of closing off the break.

## Uses of ring topology

Ring topology was prevalent when IBM introduced its local-area network **Token Ring**. Today, the need for higher-speed LANs has made this topology less popular.

# Hybrid Topology

A network can be hybrid. For example, we can have a main star topology with each branch connecting several stations in a bus topology as shown in Figure:-



# Categories of Networks

Networks are classified based upon the size, the area it covers and its physical architecture. The three primary network categories are **LAN, WAN and MAN**. Each network differs in their characteristics such as distance, transmission speed, cables and cost.

# Local Area Network(LAN)

- ❖ A local area network (LAN) is usually privately owned and links the devices in a single office, building, or campus.
- ❖ Depending on the needs of an organization and the type of technology used, a LAN can be as simple as two PCs and a printer in someone's home office; or it can extend throughout a company.
- ❖ Currently, LAN size is limited to a few kilometers.

# Local Area Network(LAN)

- ❖ A local area network (LAN) is usually privately owned and links the devices in a single office, building, or campus.
- ❖ Depending on the needs of an organization and the type of technology used, a LAN can be as simple as two PCs and a printer in someone's home office; or it can extend throughout a company.
- ❖ Currently, LAN size is limited to a few kilometers.
- ❖ LANs are designed to allow resources to be shared between personal computers or workstations. The resources to be shared can include hardware (e.g., a printer), software(e.g., an application program), or data.

# Local Area Network(LAN)

- ❖ The most common LAN topologies are **bus**, **ring**, and **star**.
- ❖ Early LANs had data rates in the 4 to 16 megabits per second (Mbps) range. Today, however, speeds are normally 100 or 1000 Mbps.
- ❖ Wireless LANs are the newest evolution in LAN technology.

# **Wide Area Network(WAN)**

A wide area network (WAN) provides long-distance transmission of data, image, audio, and video information over large geographic areas that may comprise a country, a continent, or even the whole world.

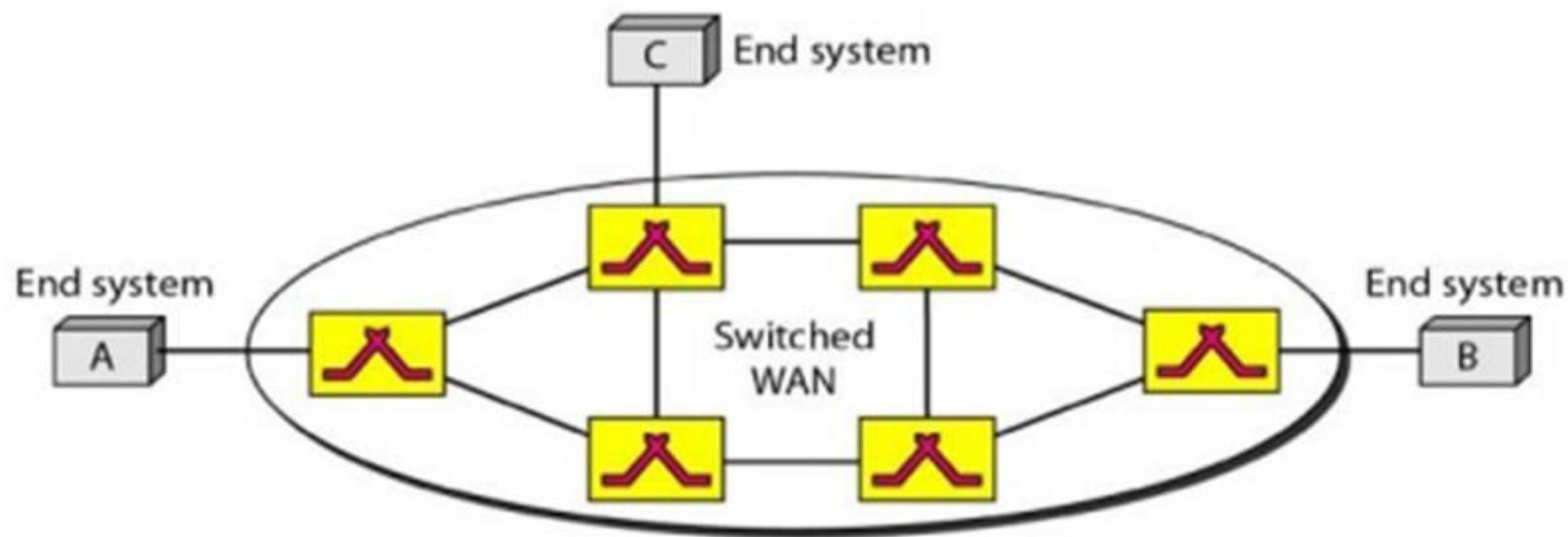
A WAN can be as complex as the backbones that connect the Internet or as simple as a dial-up line that connects a home computer to the Internet.

# Wide Area Network(WAN)

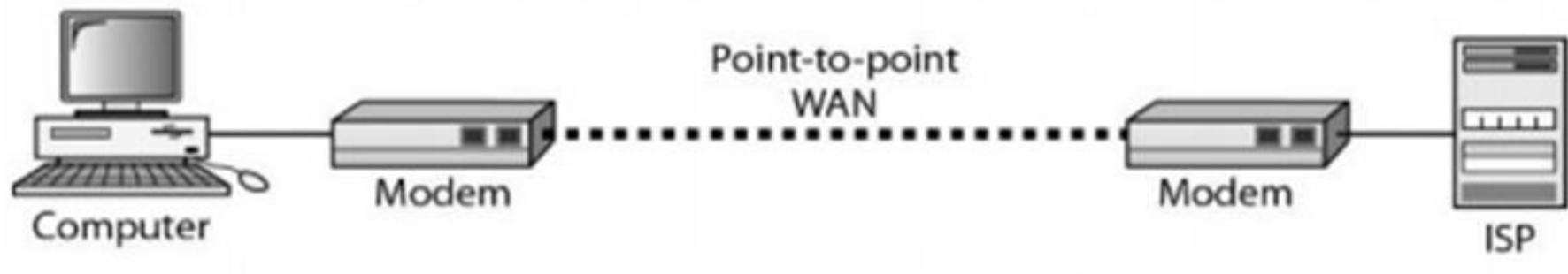
We normally refer to the first as a **switched WAN** and to the second as a **point-to-point WAN**.

- ❖ The **switched WAN** connects the end systems, which usually comprise a router (internetworking connecting device ) that connects to another LAN or WAN.
- ❖ The **point-to-point WAN** is normally a line leased from a telephone or cable TV provider that connects a home computer or a small LAN to an Internet service provider (ISP). This type of WAN is often used to provide Internet access.

# Wide Area Network(WAN)



a. Switched WAN



b. Point-to-point WAN

# Wide Area Network(WAN)

- ❖ An early example of a switched WAN is X.25, a network designed to provide connectivity between end users.
- ❖ X.25 is being gradually replaced by a high-speed, more efficient network called Frame Relay.
- ❖ A good example of a switched WAN is the asynchronous transfer mode (ATM) network, which is a network with fixed-size data unit packets called cells.
- ❖ Another example of WANs is the wireless WAN that is becoming more and more popular.

# Metropolitan Area Networks(MAN)

- ❖ A metropolitan area network (MAN) is a network with a size between a LAN and a WAN. It normally covers the area inside a town or a city. It is designed for customers who need a high-speed connectivity, normally to the Internet, and have endpoints spread over a city or part of city.
- ❖ A good example of a MAN is the part of the telephone company network that can provide a high-speed DSL line to the customer.
- ❖ Another example is the cable TV network that originally was designed for cable TV, but today can also be used for high-speed data connection to the Internet.

# Interconnection of Networks: Internetwork

An internetwork can be defined as two or more computer networks (typically Local Area Networks LAN) which are connected together, using Network Routers.

It is also called **internet**.

Each network in an Internetwork has its own Network Address, which is different from other networks in the Internetwork. Network Address is used to identify the networks inside an Internetwork.

Internetwork allows different users at different geographical locations of an organization to share data, resources and to communicate. Modern businesses cannot even function without Internetwork. Internet, Intranet and Extranet are different types of internetwork.

# **Internet, Intranet and Extranet**

## **Internet:**

Internet is a worldwide, publicly accessible computer network of interconnected computer networks (internetwork) that transmit data using the standard Internet Protocol (IP). Largest Internetwork in the world is Internet.

# Internet, Intranet and Extranet

## Intranet:

- ❖ An intranet is a private network that is contained within an enterprise.
- ❖ Typical intranet for a business organization consists of many interlinked local area networks (LAN) and use any Wide Area Network (WAN) technology for network connectivity.
- ❖ The main purpose of an intranet is to share company information and computing resources among employees.
- ❖ Intranet is a private Internetwork, which is usually created and maintained by a private organization.
- ❖ The content available inside Intranet are intended only for the members of that organization (usually employees of a company).

# Internet, Intranet and Extranet

## Extranet:

- ❖ An extranet can be viewed as part of a company's intranet that is extended to users outside the company like suppliers, vendors, partners, customers, or other business associates.
- ❖ Extranet is required for normal day-to-day business activities. For example, placing purchase order to registered vendors, billing & invoices, payments related activities, joint venture related activities, product brochures for partners, discounted price lists for partners etc.

# **Network Models**

# OSI Model

- ❖ Its full form is Open Systems Interconnection model.
- ❖ It was developed by International Standards Organization (ISO) organization in 1970.
- ❖ 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 show how to facilitate communication between different systems without requiring changes to the logic of the underlying hardware and software.
- ❖ The OSI model is not a protocol; it is a model for understanding and designing a network architecture that is flexible, robust, and interoperable.

# OSI Model

- ❖ The OSI model is a layered framework for the design of network systems that allows communication between all types of computer systems.
- ❖ It consists of seven separate but related layers, each of which defines a part of the process of moving information across a network.

# OSI Model

7

Application

6

Presentation

5

Session

4

Transport

3

Network

2

Data link

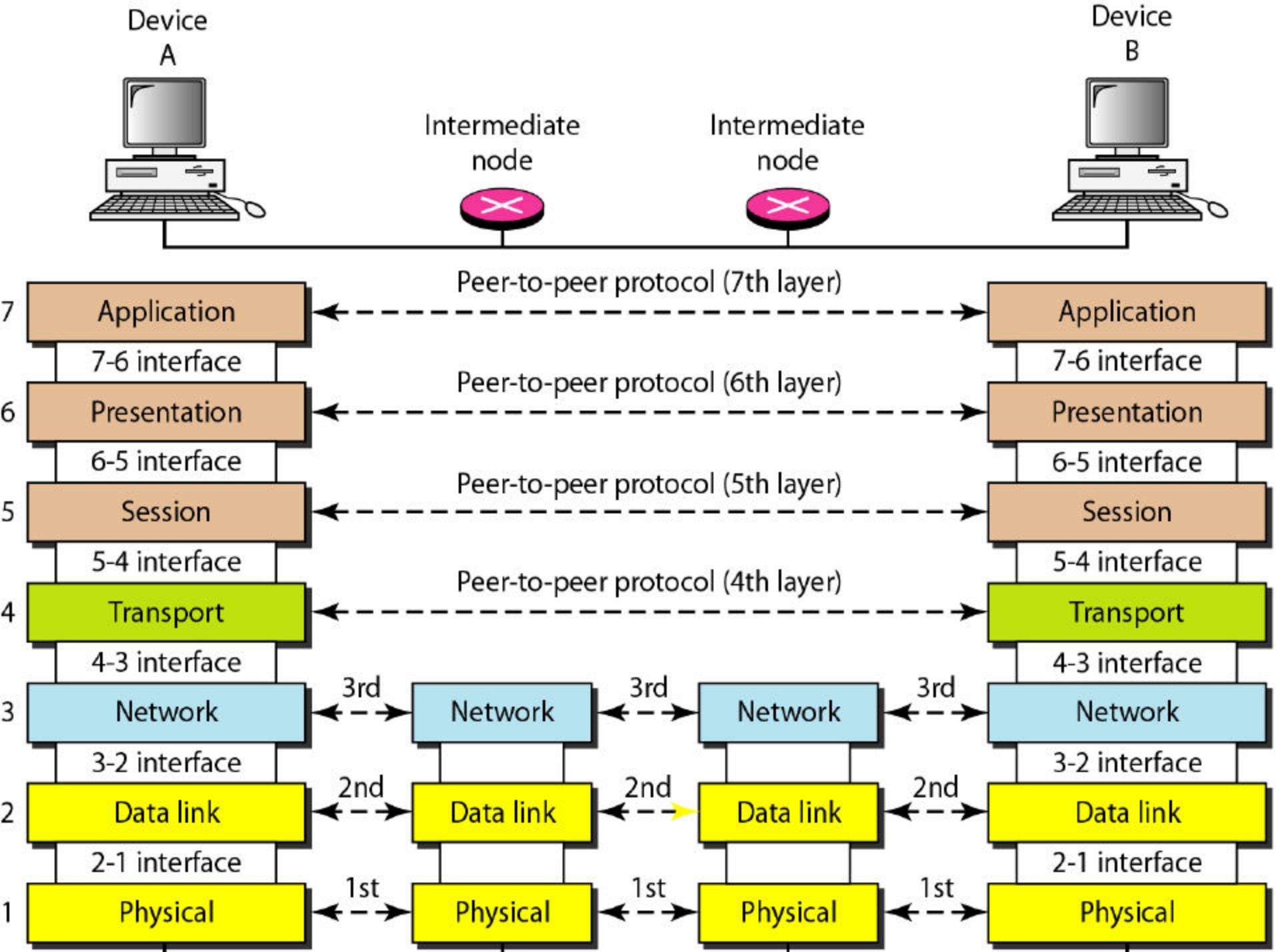
1

Physical

# OSI Model

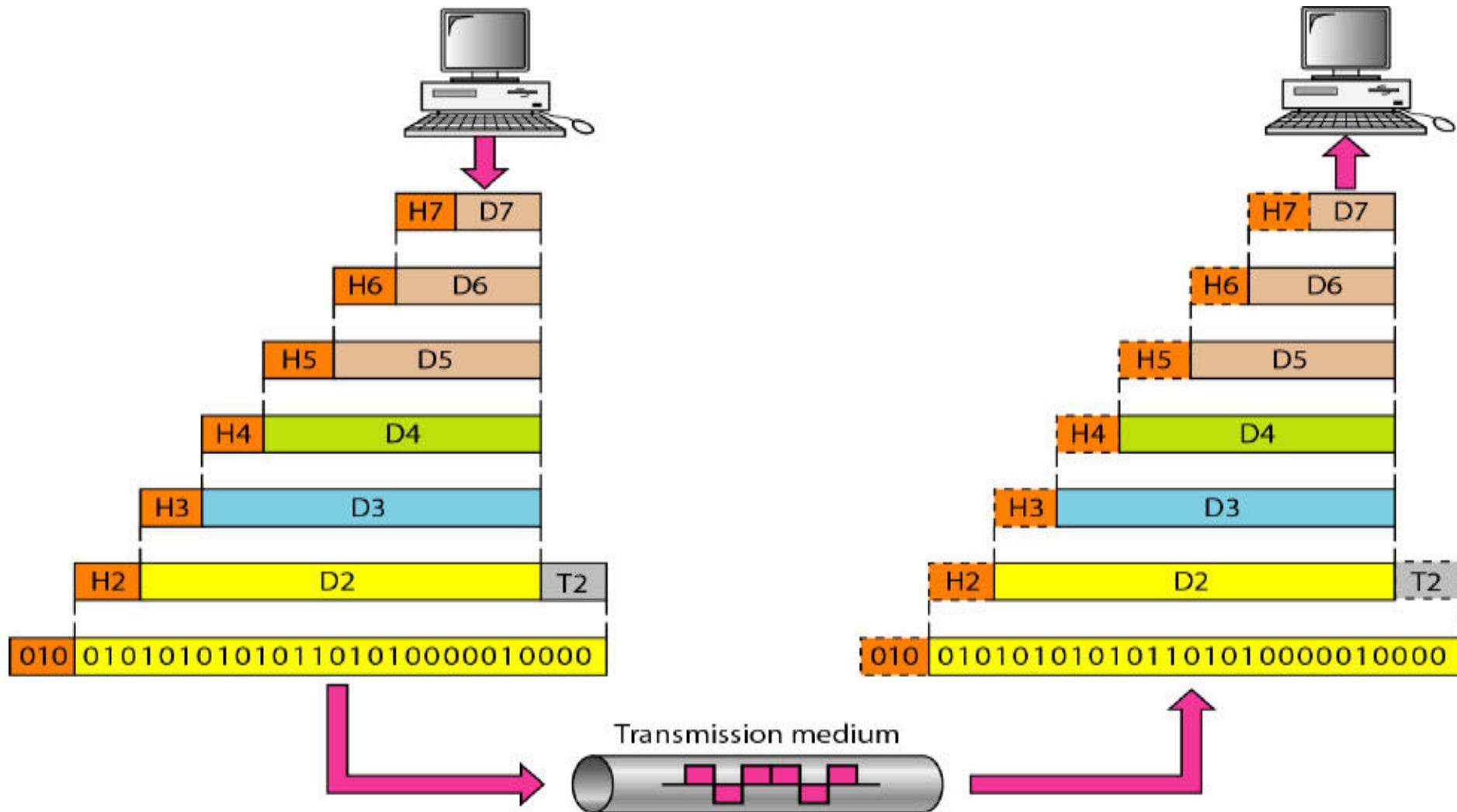
## Layered Architecture

Following figure shows the layers involved when a message is sent from device A to device B. As the message travels from A to B, it may pass through many intermediate nodes. These intermediate nodes usually involve only the first three layers of the OSI model.



# OSI Model

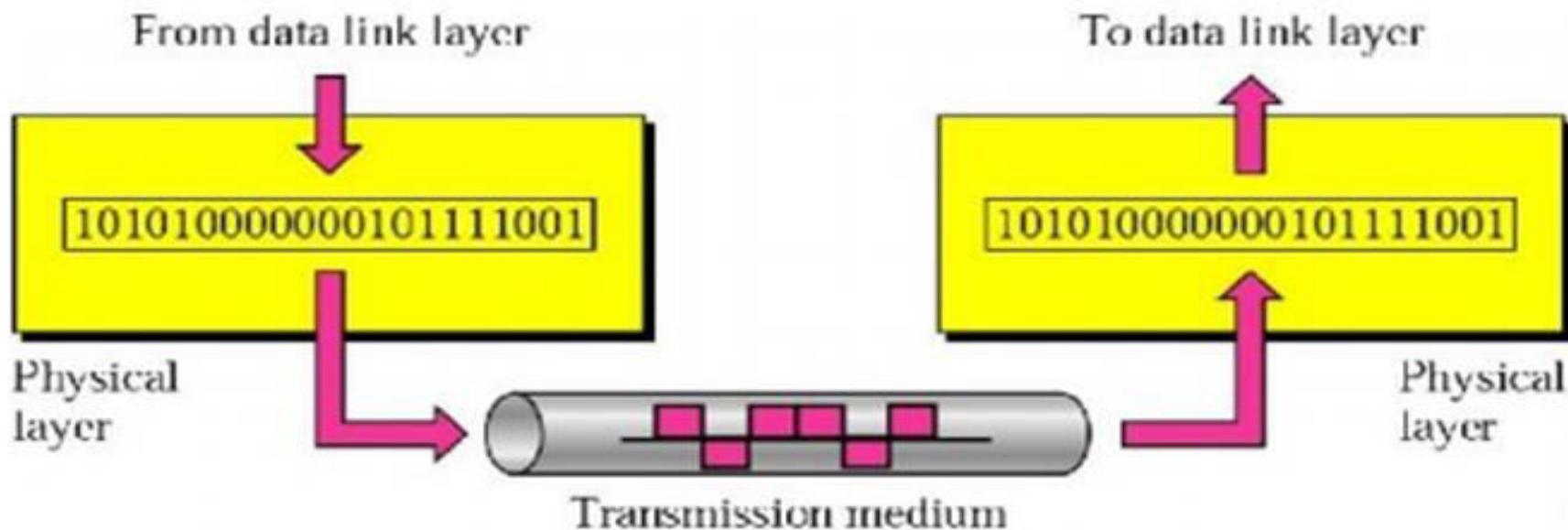
An exchange of messages using OSI model is shown in the following figure:-



# Functions of Physical Layer

The physical layer coordinates the functions required to carry a bit stream over a physical medium. It deals with the mechanical and electrical specifications of the interface and transmission medium.

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



# Functions of Physical Layer

The physical layer is also concerned with the following:

- **Physical characteristics of interfaces and medium:** The physical layer defines the characteristics of the interface between the devices and the transmission medium. It also defines the type of transmission medium.
- **Representation of bits:** The physical layer data consists of a stream of bits (sequence of 0s or 1s) with no interpretation. To be transmitted, bits must be encoded into signals--electrical or optical. The physical layer defines the type of encoding (how 0s and 1s are changed to signals).

# Functions of Physical Layer

- **Data rate:** The transmission rate-the number of bits sent each second-is also defined by the physical layer.
- **Synchronization of bits:** The sender and receiver not only must use the same bit rate but also must be synchronized at the bit level.
- **Line configuration:** The physical layer is concerned with the connection of devices to the media; it may be point-to-point configuration or multi-point.

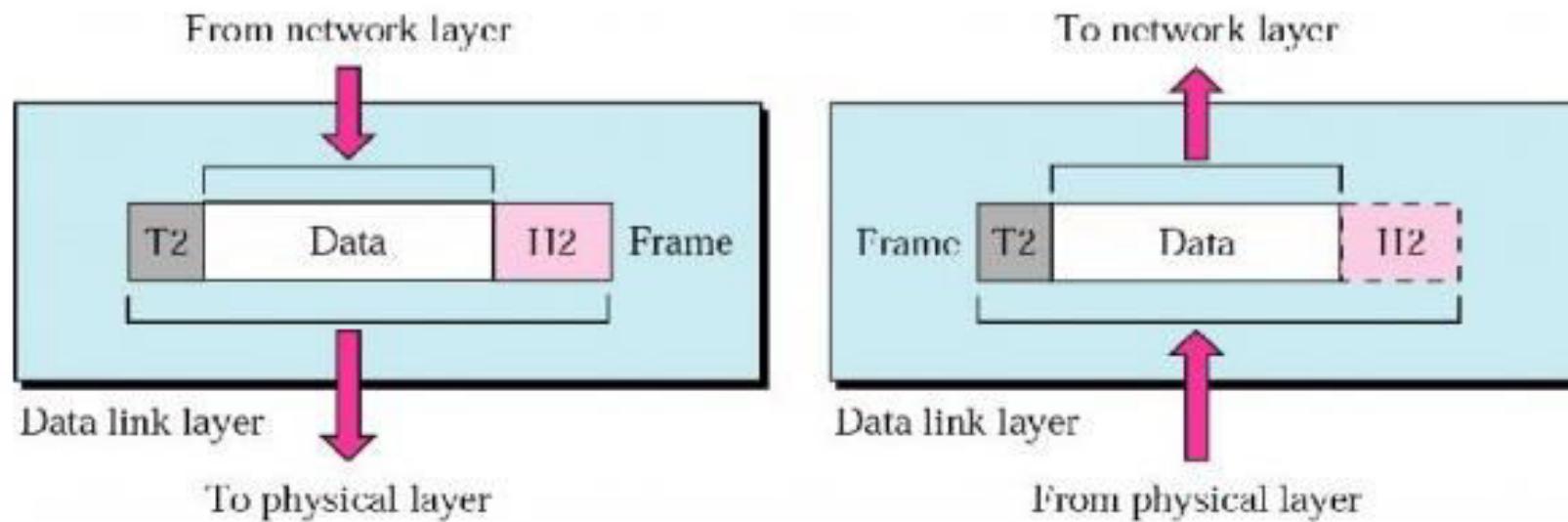
# Functions of Physical Layer

- **Physical topology:** The physical topology defines how devices are connected to make a network. Ex. Mesh, Star, Ring, Bus, or a hybrid topology.
- **Transmission mode:** The physical layer also defines the direction of transmission between two devices: simplex, half-duplex, or full-duplex.

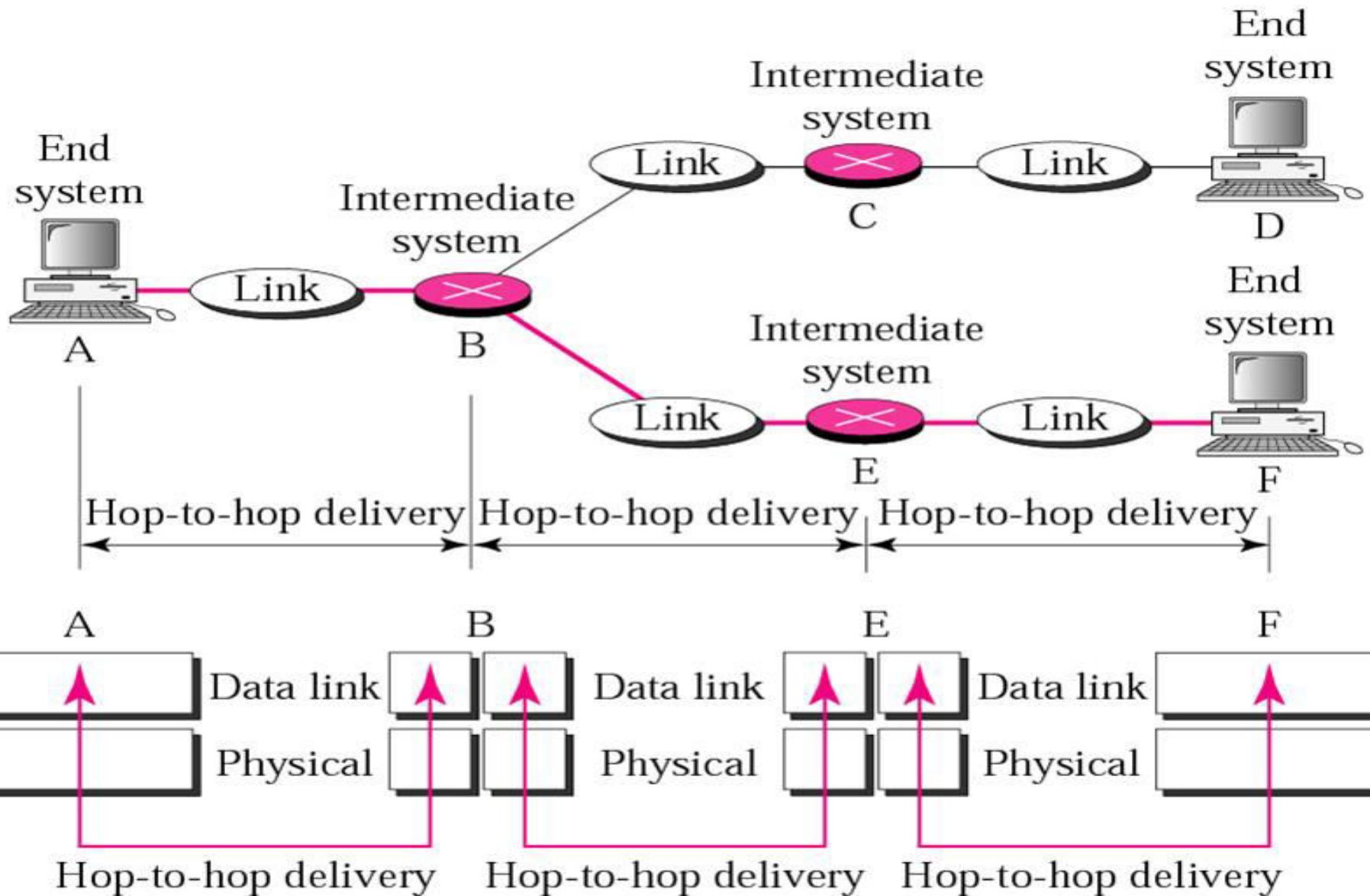
# Functions of Data Link Layer

The data link layer transforms the physical layer, a raw transmission facility, to a reliable link. It makes the physical layer appear error-free to the upper layer (network layer).

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



# Functions of Data Link Layer



# Functions of Data Link Layer

Other responsibilities of the data link layer include the following:

- 1. Framing:** The data link layer divides the stream of bits received from the network layer into manageable data units called frames.
- 2. Physical addressing:** If frames are to be distributed to different systems on the network , the data link layer adds a header to the frame to define the sender and/or receiver of the frame.
- 3. Flow control:** If the rate at which the data are absorbed by the receiver is less than the rate at which data are produced in the sender, the data link layer imposes a flow control mechanism to avoid overwhelming the receiver.

# Functions of Data Link Layer

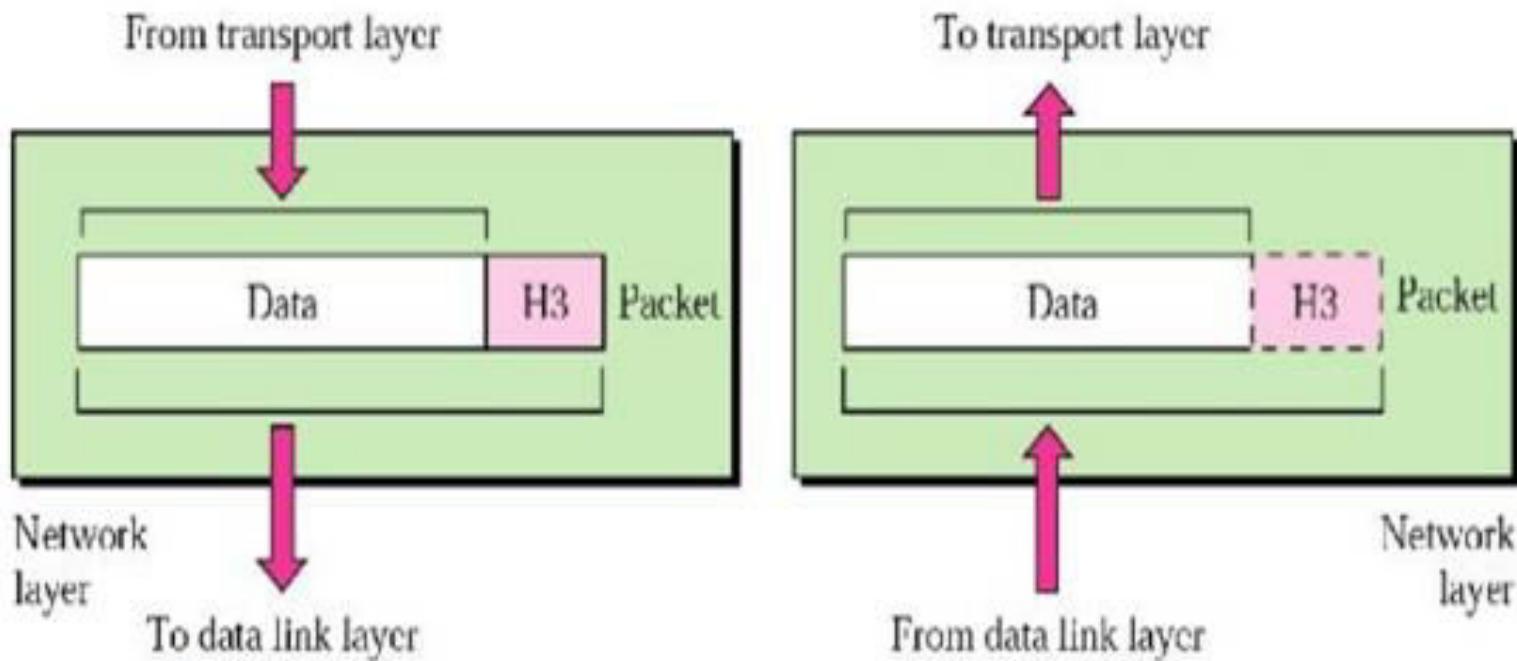
4. **Error control:** The data link layer adds reliability to the physical layer by adding mechanisms to detect and retransmit damaged or lost frames.
5. **Access control:** When two or more devices are connected to the same link, data link layer protocols are necessary to determine which device has control over the link at any given time.

# Functions of Network Layer

- The network layer is responsible for the source-to-destination delivery of a packet, possibly across multiple networks (links). Whereas the data link layer oversees the delivery of the packet between two systems on the same network (links), the network layer ensures that each packet gets from its point of origin to its final destination.
- If two systems are connected to the same link, there is usually no need for a network layer. However, if the two systems are attached to different networks (links) with connecting devices between the networks (links), there is often a need for the network layer to accomplish source-to-destination delivery.

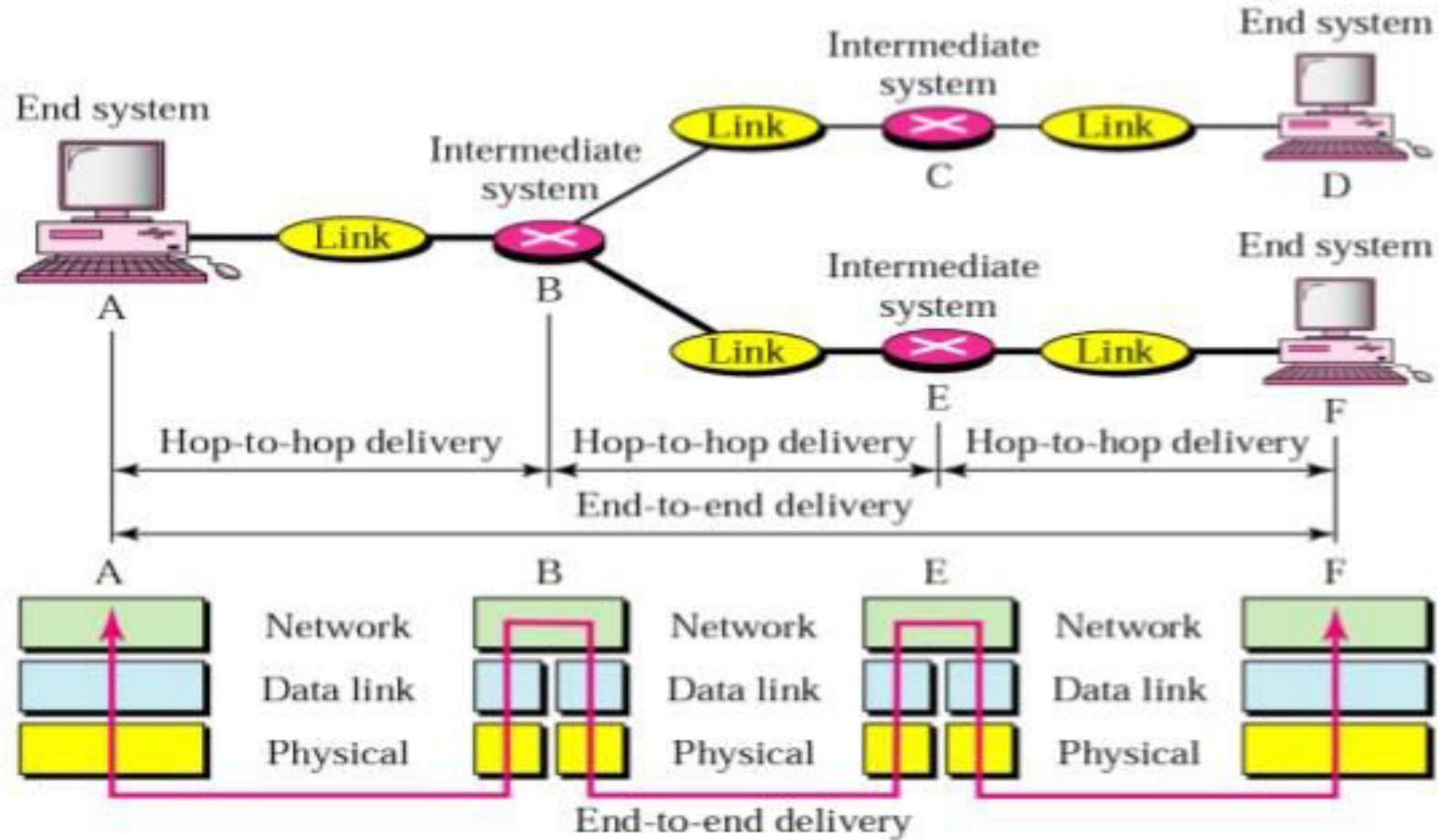
# Functions of Network Layer

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



# Functions of Network Layer

## Source to destination delivery



# Functions of Network Layer

Other responsibilities of the network layer include the following:

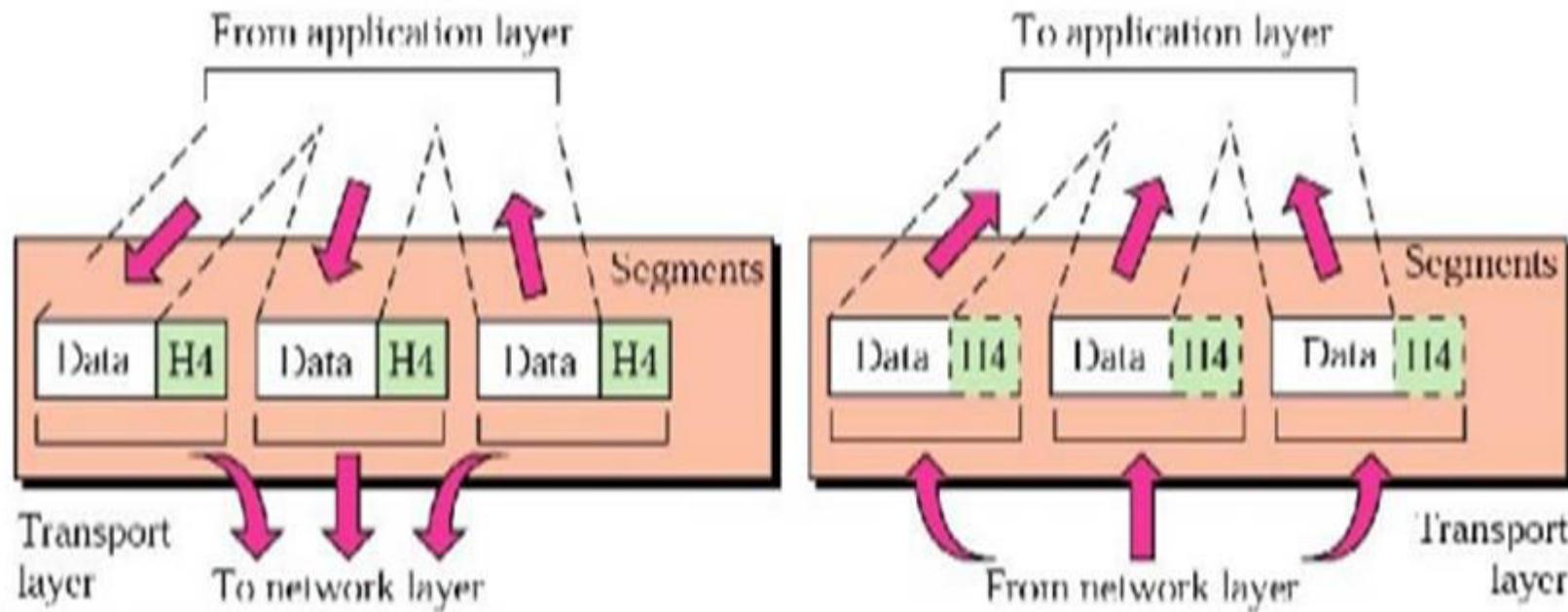
- **Logical addressing.** The physical addressing implemented by the data link layer handles the addressing problem locally. If a packet passes the network boundary, we need another addressing system to help distinguish the source and destination systems. The network layer adds a header to the packet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

# Functions of Transport Layer

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

# Functions of Transport Layer

Following figure shows the relationship of the transport layer to the network and session layers.



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

# Functions of Transport Layer

Other responsibilities of the transport layer include the following:

- a. **Service-point addressing:** Source-to-destination delivery means delivery not only from one computer to the next but also from a specific process (running program) on one computer to a specific process (running program) on the other. The transport layer header must therefore include a type of address called a service-point address (or port address).
  
- b. **Segmentation and reassembly:** A message is divided into transmittable segments, with each segment containing a sequence number. These numbers enable the transport layer to reassemble the message correctly upon arriving at the destination and to identify and replace packets that were lost in transmission.

# Functions of Transport Layer

- c. **Connection control:** The transport layer can be either connectionless or connection oriented. A connectionless transport layer treats each segment as an independent packet and delivers it to the transport layer at the destination machine. A connection oriented transport layer makes a connection with the transport layer at the destination machine first before delivering the packets. After all the data are transferred, the connection is terminated.
- d. **Flow control:** Like the data link layer, the transport layer is responsible for flow control. However, flow control at this layer is performed end to end rather than across a single link.

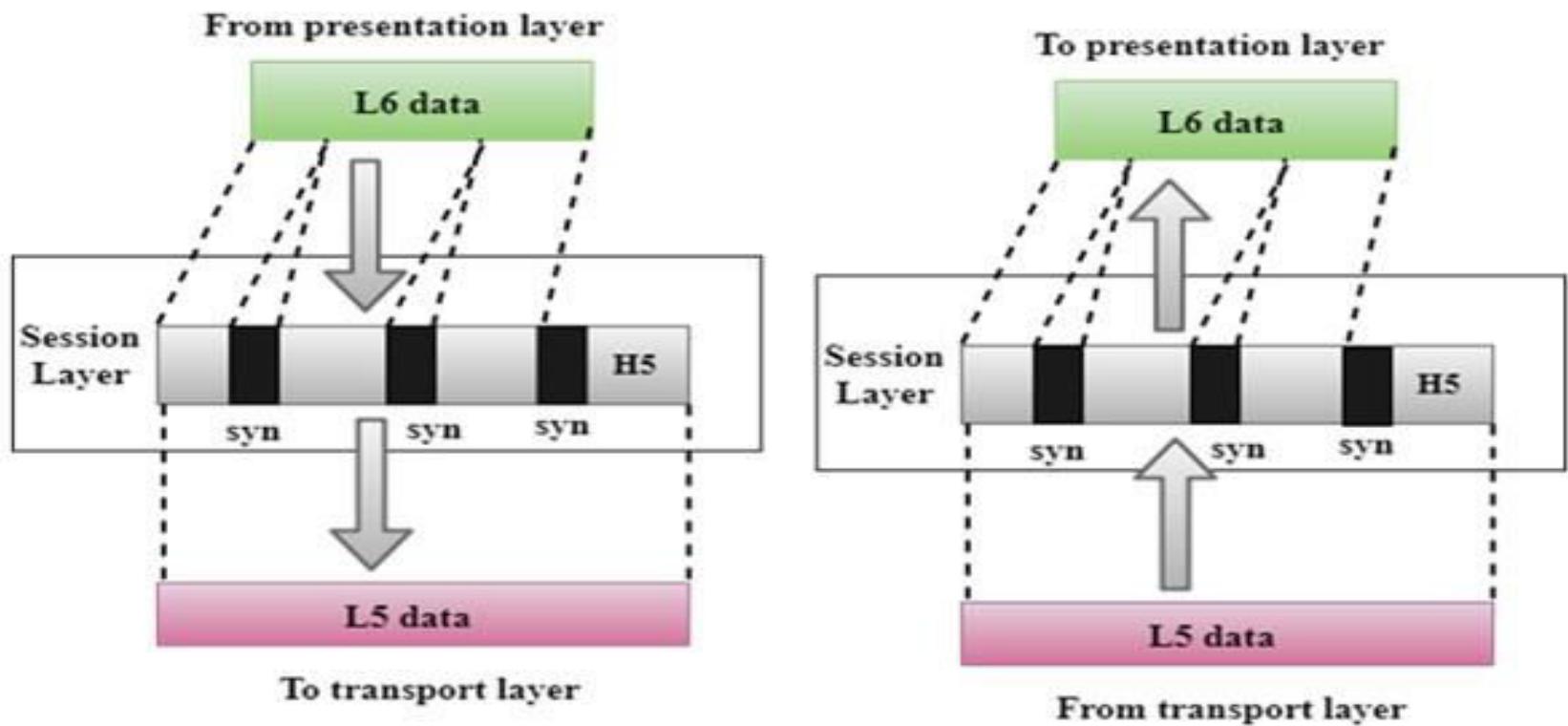
# Functions of Transport Layer

e. **Error control.** Like the data link layer, the transport layer is responsible for error control. However, error control at this layer is performed process-to-process rather than across a single link. The sending transport layer makes sure that the entire message arrives at the receiving transport layer without error (damage, loss, or duplication). Error correction is usually achieved through retransmission.

# Functions of Session Layer

The session layer is the network dialog controller. It establishes, maintains, and synchronizes the interaction among communicating systems.

**Note:** The session layer is responsible for dialog control and synchronization.



# Functions of Session Layer

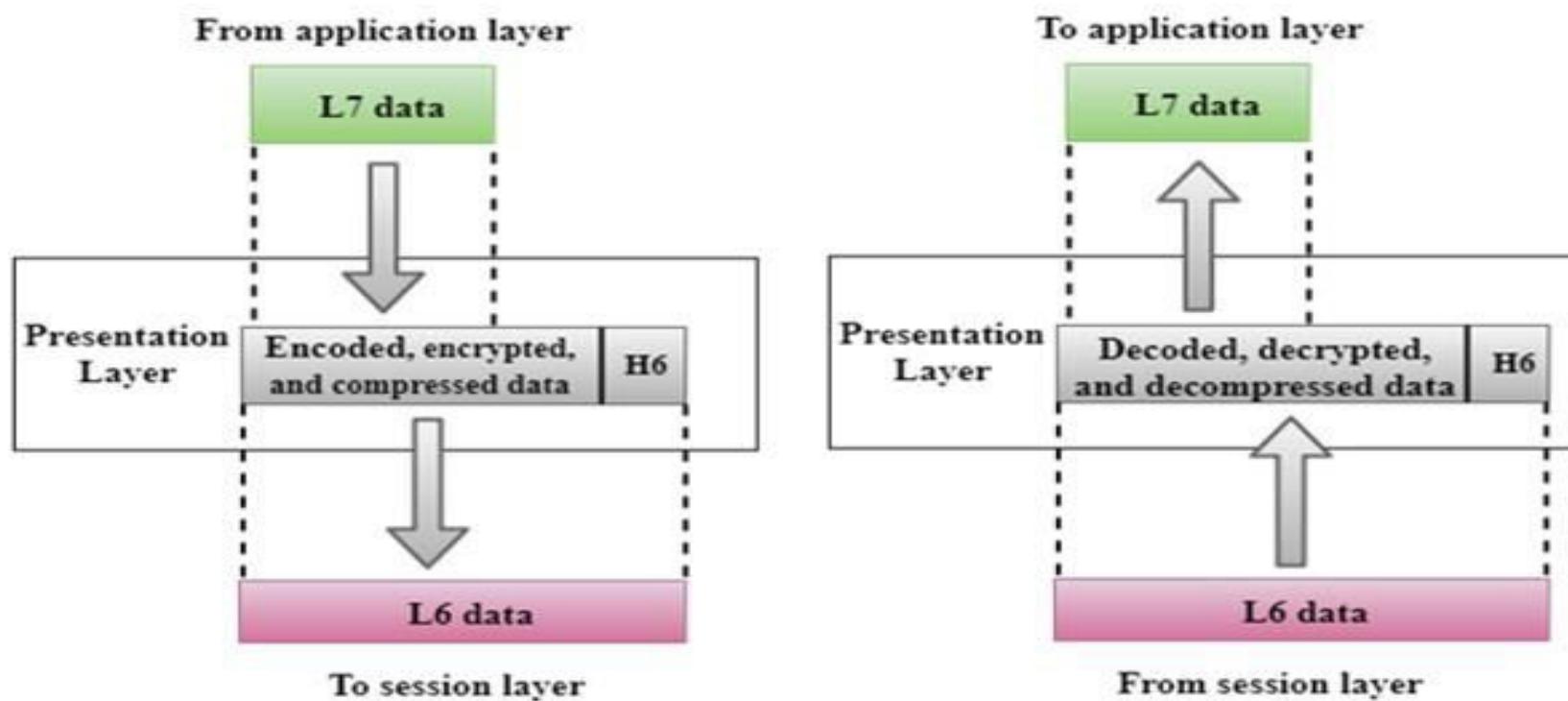
Specific responsibilities of the session layer include the following:

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

# Functions of Presentation Layer

The presentation layer is concerned with the syntax and semantics of the information exchanged between two systems.

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



# Functions of Presentation Layer

Specific responsibilities of the presentation layer include the following:

- a. **Translation.** The processes (running programs) in two systems are usually exchanging information in the form of character strings, numbers, and so on. The information must be changed to bit streams before being transmitted. The presentation layer at the sender changes the information from its sender-dependent format into a common format. The presentation layer at the receiving machine changes the common format into its receiver-dependent format.

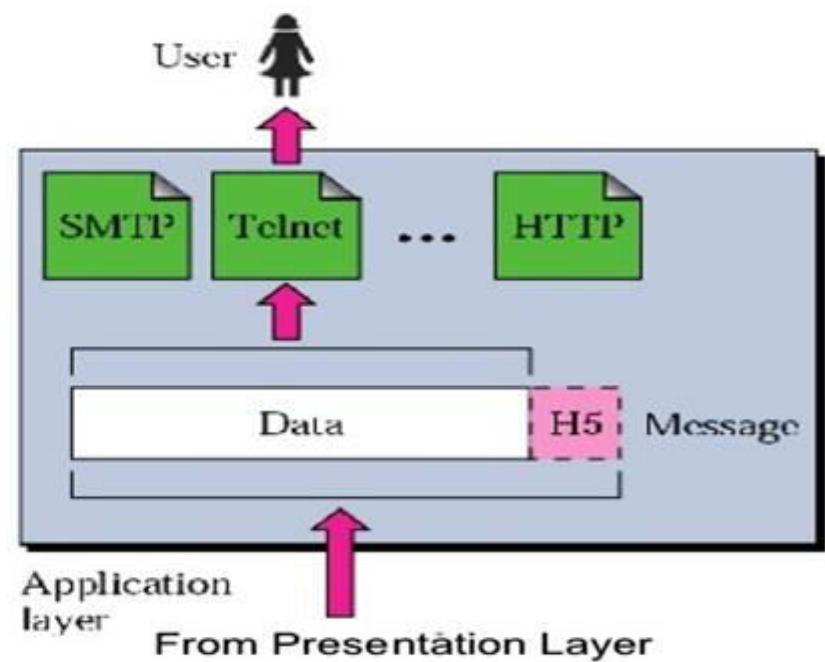
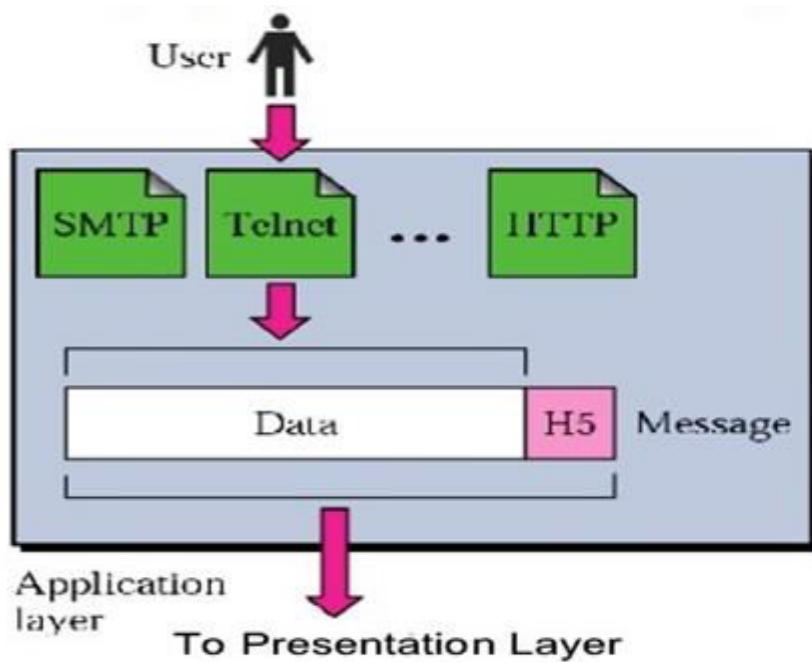
# Functions of Presentation Layer

- b. **Encryption.** A system must be able to ensure privacy. Encryption means that the sender transforms the original information to another form and sends the resulting message out over the network. Decryption reverses the original process to transform the message back to its original form.
  
- c. **Compression.** Data compression reduces the number of bits contained in the information. It is very important in the transmission of multimedia such as text, audio, and video.

# Functions of 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.

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



# Functions of Application Layer

Specific services provided by the application layer include the following:

- a. **Network virtual terminal.** It is a software version of a physical terminal, and it allows a user to log on to a remote host. To do so, the application creates a software emulation of a terminal at the remote host.
  
- b. **File transfer, access, and management.** This application allows a user to access files in a remote host, to retrieve files from a remote computer for use in the local computer, and to manage or control files in a remote computer locally.

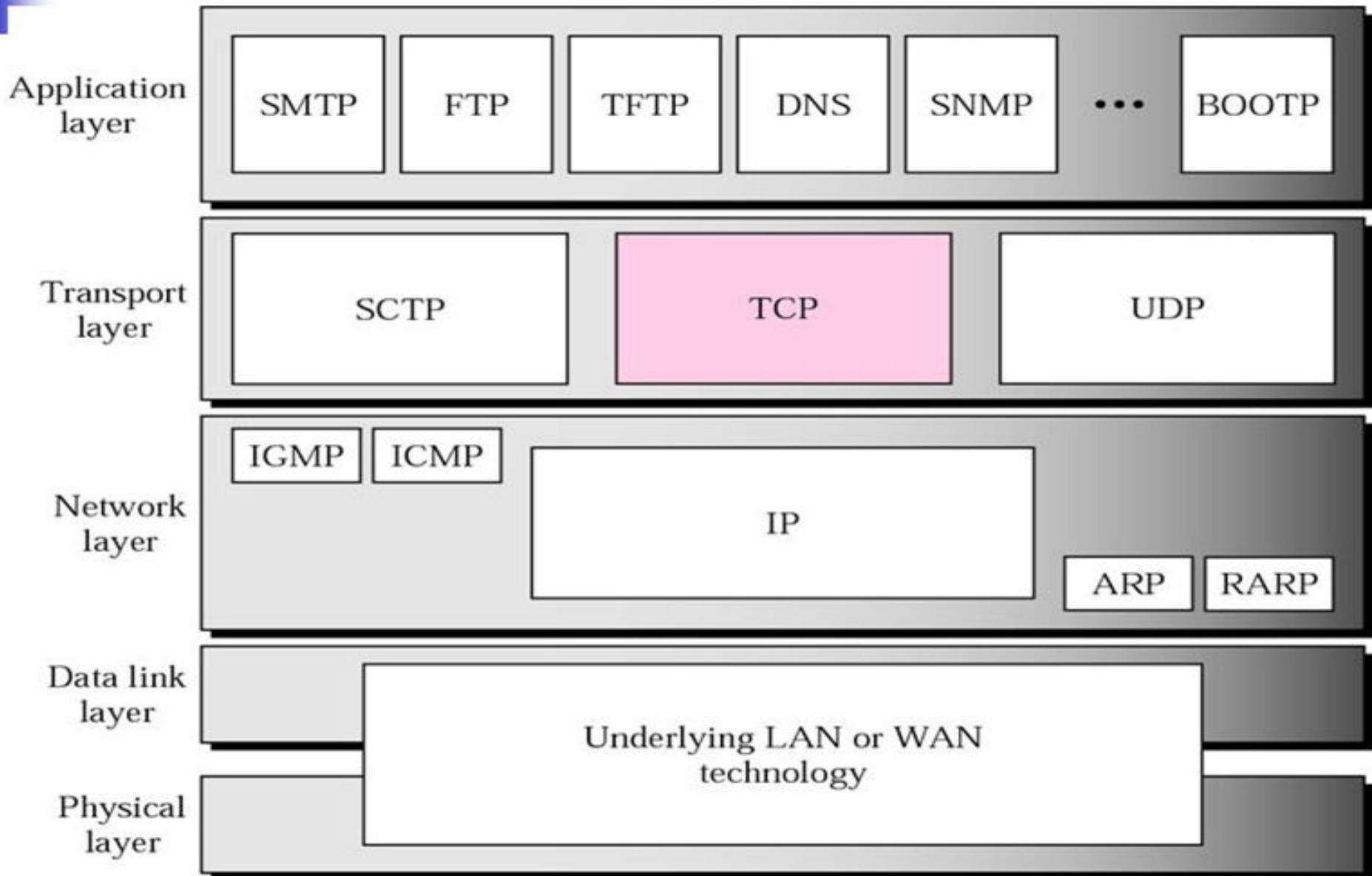
# Functions of Application Layer

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

# TCP/IP Protocol Suite

- The TCP/IP protocol suite was developed prior to the OSI model. Therefore, the layers in the TCP/IP protocol suite do not exactly match those in the OSI model.
- The original TCP/IP protocol suite was defined as having four layers: host-to-network, internet, transport, and application.
- However, when TCP/IP is compared to OSI, we can say that the host-to-network layer is equivalent to the combination of the physical and data link layers. The internet layer is equivalent to the network layer, and the application layer is roughly doing the job of the session, presentation, and application layers with the transport layer in TCP/IP taking care of part of the duties of the transport layer.

# TCP/IP Protocol Suite



# TCP/IP Protocol Suite

## 1. Physical and Data Link Layers:

At the physical and data link layers, TCP/IP does not define any specific protocol. It supports all the standard and proprietary protocols. A network in a TCP/IP internetwork can be a local-area network or a wide-area network.

## 2. Network Layer:

At the network layer (or, more accurately, the internetwork layer), TCP/IP supports the Internetworking Protocol. IP, in turn, uses four supporting protocols: ARP, RARP, ICMP, and IGMP.

### a. Internetworking Protocol (IP)

The Internetworking Protocol (IP) is the transmission mechanism used by the TCP/IP protocols. It is an unreliable and connectionless protocol a best-effort delivery service. The term **best effort** means that IP provides no error checking or tracking.

# TCP/IP Protocol Suite

## b. Address Resolution Protocol

The Address Resolution Protocol (ARP) is used to associate a logical address with a physical address. On a typical physical network, such as a LAN, each device on a link is identified by a physical or station address, usually imprinted on the network interface card (NIC). ARP is used to find the physical address of the node when its Internet address is known.

## c. Reverse Address Resolution Protocol

The Reverse Address Resolution Protocol (RARP) allows a host to discover its Internet address when it knows only its physical address. It is used when a computer is connected to a network for the first time or when a diskless computer is booted.

# TCP/IP Protocol Suite

## d. Internet Control Message Protocol

The Internet Control Message Protocol (ICMP) is a mechanism used by hosts and gateways to send notification of datagram problems back to the sender. ICMP sends query and error reporting messages.

## e. Internet Group Message Protocol

The Internet Group Message Protocol (IGMP) is used to facilitate the simultaneous transmission of a message to a group of recipients.

## 3. Transport Layer:

Traditionally the transport layer was represented in TCP/IP by two protocols: TCP and UDP. IP is a host-to-host protocol, meaning that it can deliver a packet from one physical device to another. UDP and TCP are transport level protocols responsible for delivery of a message from a process (running program) to another process. A new transport layer protocol, SCTP, has been devised to meet the needs of some newer applications.

# TCP/IP Protocol Suite

## a. User Datagram Protocol

The User Datagram Protocol (UDP) is the simpler of the two standard TCP/IP transport protocols. It is a process-to-process protocol that adds only port addresses, checksum error control, and length information to the data from the upper layer.

## b. Transmission Control Protocol

The Transmission Control Protocol (TCP) provides full transport-layer services to applications. TCP is a reliable stream transport protocol. The term stream, in this context, means connection-oriented: A connection must be established between both ends of a transmission before either can transmit data.

## c. Stream Control Transmission Protocol

The Stream Control Transmission Protocol (SCTP) provides support for newer applications such as voice over the Internet. It is a transport layer protocol that combines the best features of UDP and TCP.

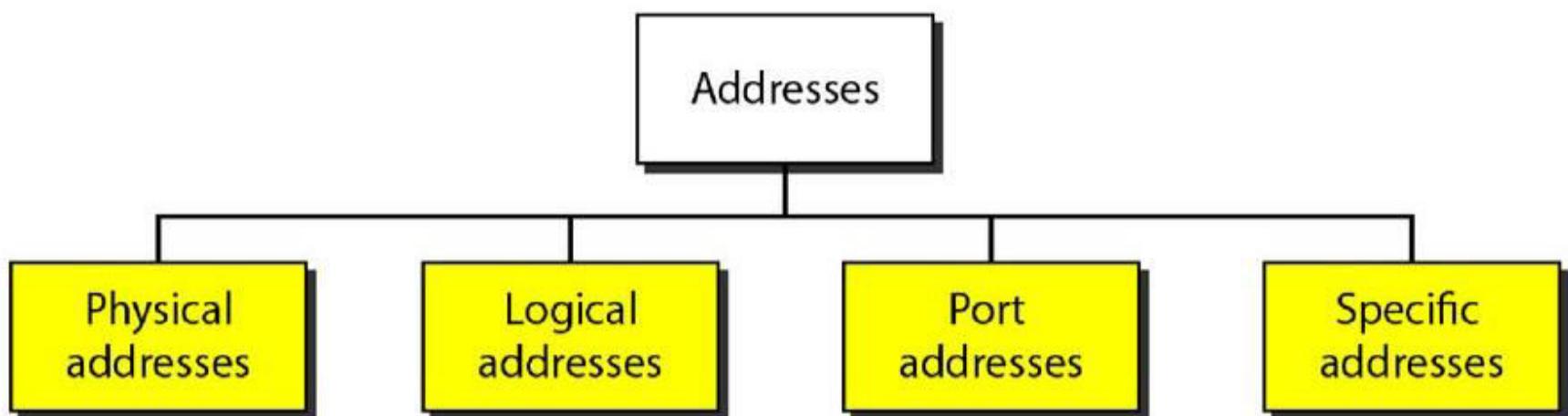
# TCP/IP Protocol Suite

## 4. Application Layer

The application layer in TCP/IP is equivalent to the combined session, presentation, and application layers in the OSI model. Many protocols are defined at this layer such as FTP, Telnet, SMTP, DNS, SNMP.

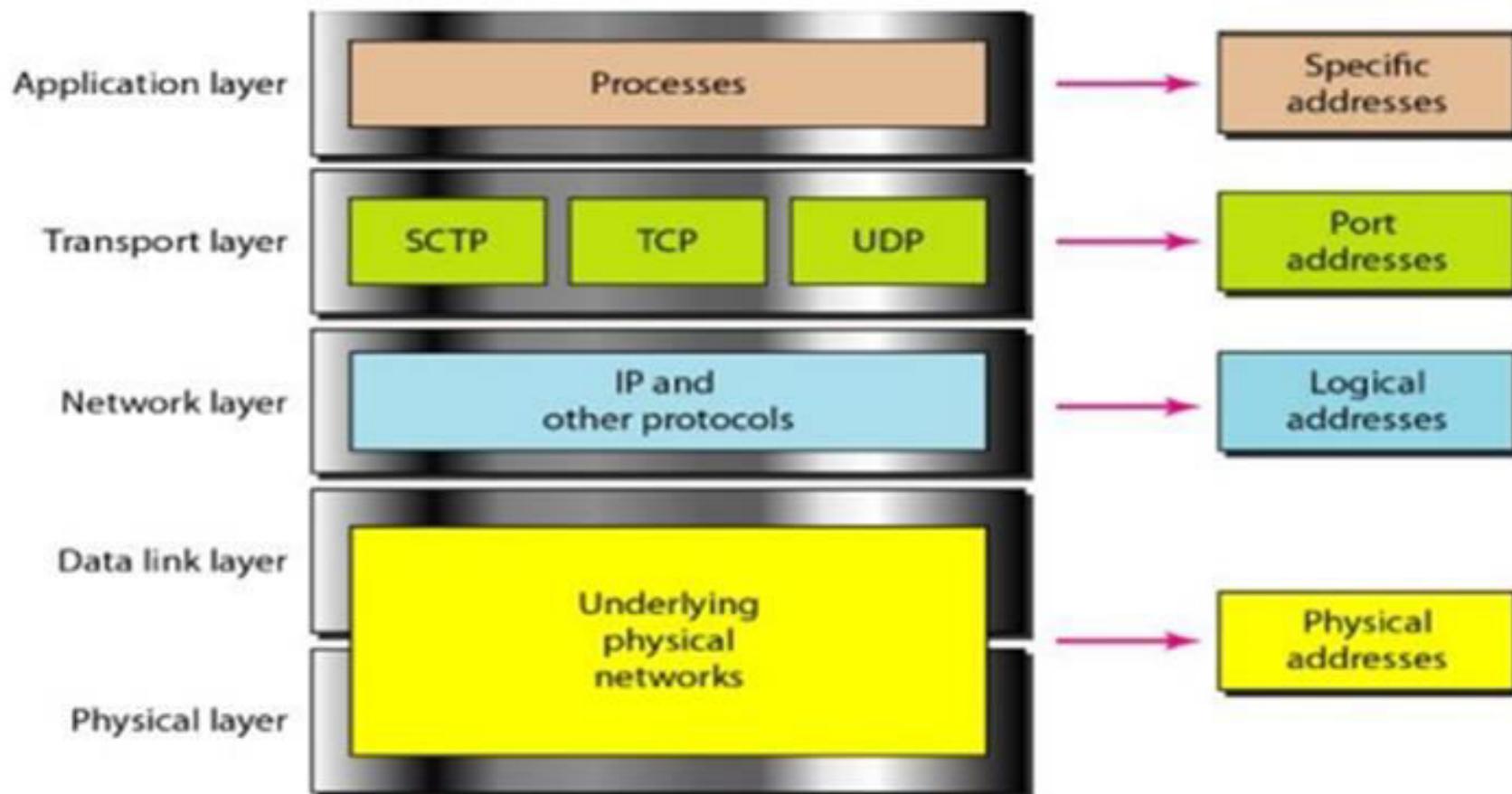
# ADDRESSING

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.



# ADDRESSING

Each address is related to a specific layer in the TCP/IP architecture.



# Computer Network Components

Some important network components are NIC, switch, cable, hub, router, and modem.

## NIC

- NIC stands for network interface card.
- NIC is a hardware component used to connect a computer with another computer onto a network
- It can support a transfer rate of 10, 100 to 1000 Mb/s.
- The MAC address or physical address is encoded on the network card chip which is assigned by the IEEE to identify a network card uniquely. The MAC address is stored in the PROM (Programmable read-only memory).

There are two types of NIC:

- (1) Wired NIC    (2) Wireless NIC

**Wired NIC:** The Wired NIC is present inside the motherboard. Cables and connectors are used with wired NIC to transfer data.

**Wireless NIC:** The wireless NIC contains the antenna to obtain the connection over the wireless network. For example, laptop computer contains the wireless NIC.

# Computer Network Components

## Hub

- ❖ A Hub is a hardware device that divides the network connection among multiple devices. When computer requests for some information from a network, it first sends the request to the Hub through cable. Hub will broadcast this request to the entire network. All the devices will check whether the request belongs to them or not. If not, the request will be dropped.
- ❖ The process used by the Hub consumes more bandwidth and limits the amount of communication. Nowadays, the use of hub is obsolete, and it is replaced by more advanced computer network components such as Switches, Routers.

# Computer Network Components

## Switch

A switch is a hardware device that connects multiple devices on a computer network. A Switch contains more advanced features than Hub. The Switch contains the updated table that decides where the data is transmitted or not. Switch delivers the message to the correct destination based on the physical address present in the incoming message. A Switch does not broadcast the message to the entire network like the Hub. It determines the device to whom the message is to be transmitted. Therefore, we can say that switch provides a direct connection between the source and destination. It increases the speed of the network.

# Computer Network Components

## Router

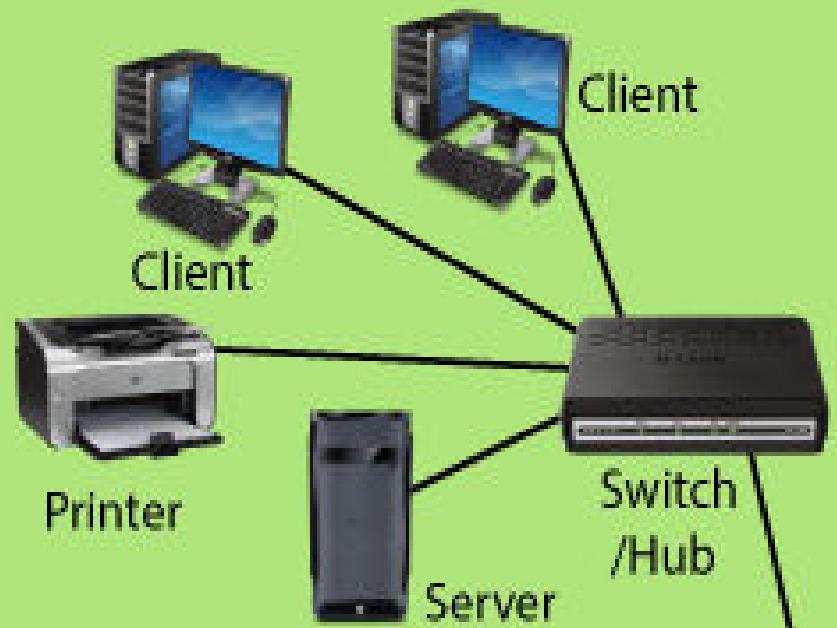
A router is a hardware device which is used to connect a LAN with an internet connection. It is used to receive, analyze and forward the incoming packets to another network.

A router works in a Layer 3 (Network layer) of the OSI Reference model.

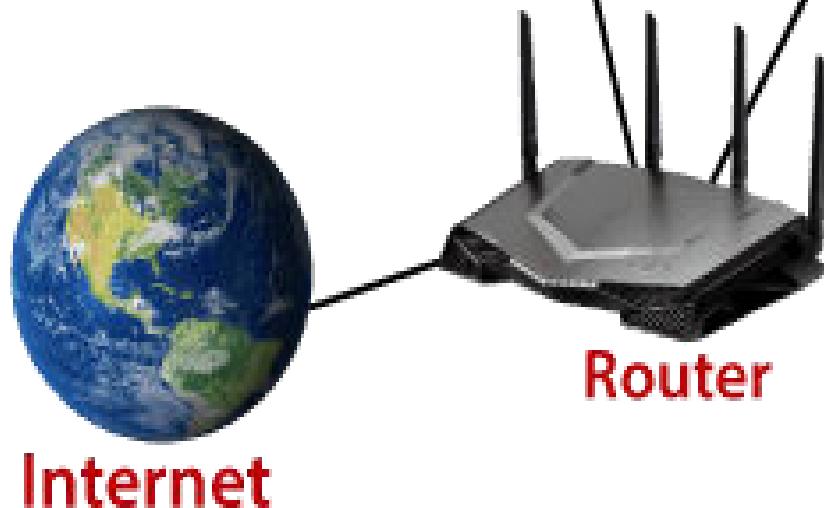
A router forwards the packet based on the information available in the routing table.

It determines the best path from the available paths for the transmission of the packet.

## LAN 1-Sales Department



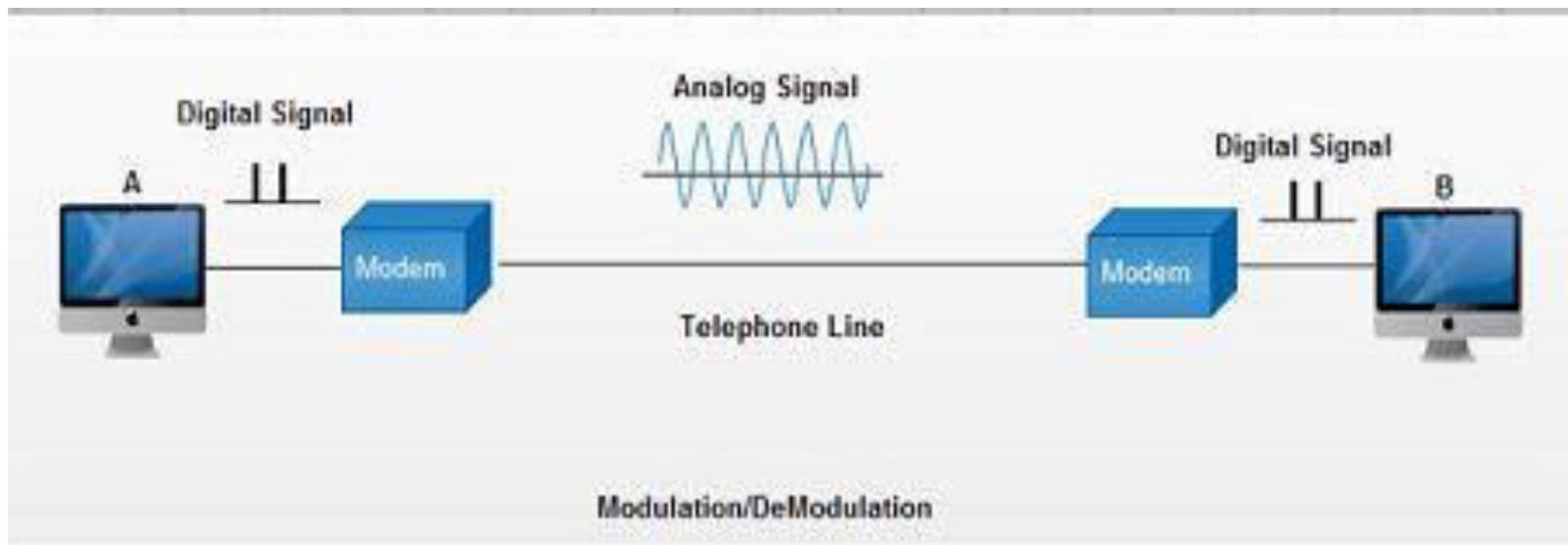
## LAN 2- Accounts Department



# Computer Network Components

## Modem

- ❖ A modem is a hardware device that allows the computer to connect to the internet over the existing telephone line.
- ❖ A modem is not integrated with the motherboard.
- ❖ It stands for Modulator/Demodulator. It converts the digital data into an analog signal over the telephone lines.

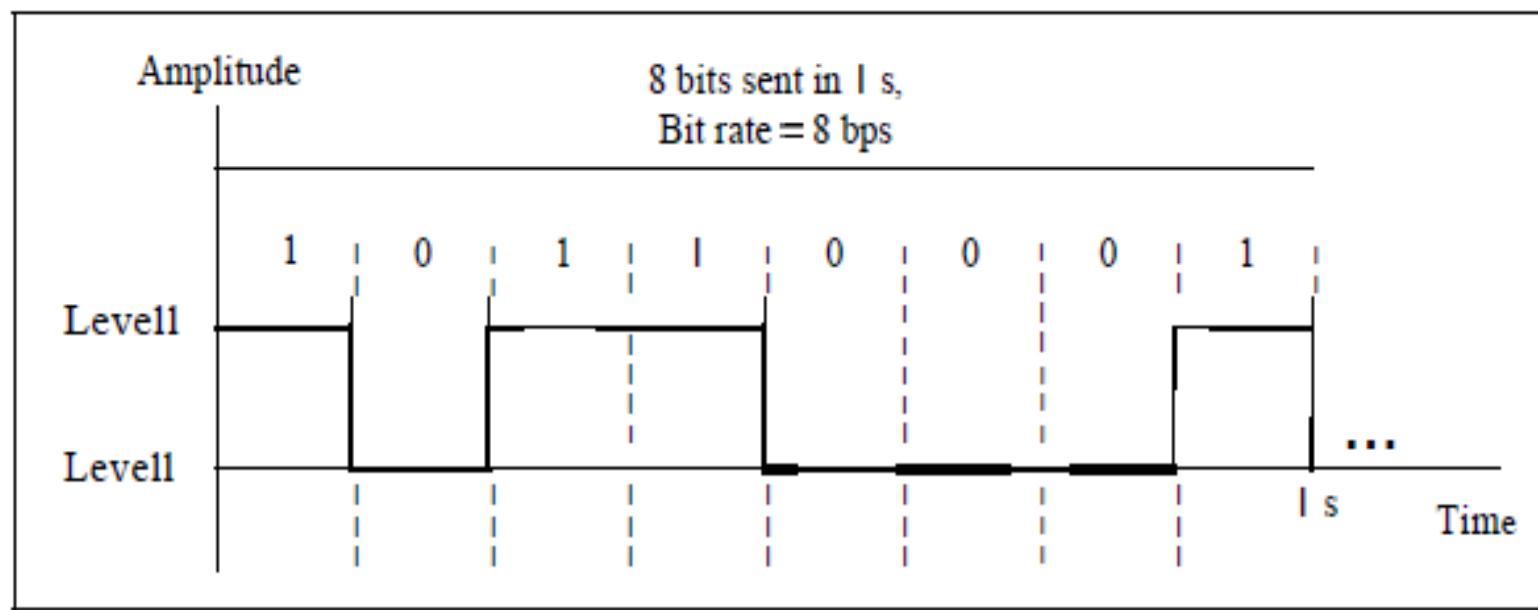


# Physical Layer

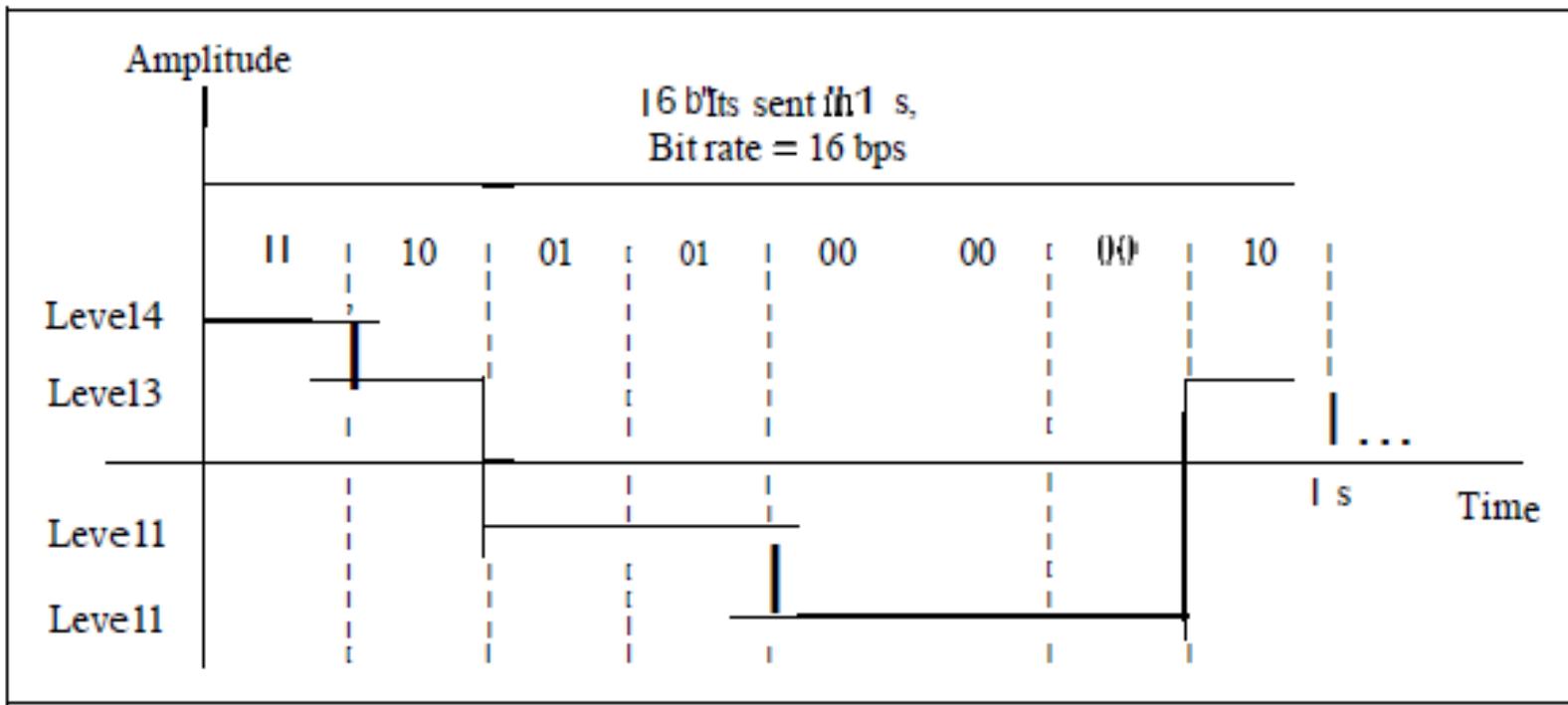
# Digital Signal

In addition to being represented by an analog signal, information can also be represented by a digital signal. For example, a 1 can be encoded as a positive voltage and a 0 as zero voltage.

A digital signal can have more than two levels. In this case, we can send more than 1 bit for each level. Following figure shows two signals, one with two levels and the other with four.



# Digital Signal



b. A digital signal with four levels

Note: If a signal has L levels, each level needs  $\log_2 L$  bits.

# Digital Signal

**Example:** A digital signal has eight levels. How many bits are needed per level?

**Solution:** Here, number of levels,  $L = 8$ .

Therefore, number of bits per level =  $\log_2 L = \log_2 8 = 3$

**Example:** A digital signal has nine levels. How many bits are needed per level?

**Solution:** Here, number of levels,  $L = 9$ .

Therefore, number of bits per level =  $\log_2 L = \log_2 9 = 3.17$  bits

However, this answer is not realistic. The number of bits sent per level needs to be an integer as well as a power of 2. For this example, 4 bits can represent one level.

# Digital Signal

## Bit rate

The bit rate is the number of bits sent in 1 second. It is expressed in bits per second (bps).

## Bit length

The bit length is the distance one bit occupies on the transmission medium.

**Bit length =propagation speed x bit duration**

# Digital Signal

## Baud rate

It is the rate at which a signal level changes over a given period of time.

**Baud rate = Bit rate / bits per signal level**

When binary bits are transmitted as an electrical signal with two levels 0 and 1, the bit rate and baud rate are the same.

**Example:**

Consider bit rate is 8 bps and number of signal levels is 4. find baud rate.

**Solution:** Number of bits required per signal level = 2

Therefore,    Baud rate =  $8/2 = 4$  bauds

# TRANSMISSION IMPAIRMENT

- Signals travel through transmission media, which are not perfect.
- The imperfection causes signal impairment. This means that the signal at the beginning of the medium is not the same as the signal at the end of the medium. What is sent is not what is received.
- Three causes of impairment are attenuation, distortion, and noise.

## Attenuation

Attenuation means a loss of energy. When a signal, simple or composite, travels through a medium, it loses some of its energy in overcoming the resistance of the medium. Some of the electrical energy in the signal is converted to heat. To compensate for this loss, amplifiers are used to amplify the signal.

# Attenuation

## Decibel

The decibel (dB) measures the relative strengths of two signals or one signal at two different points.

**Note:** The decibel is negative if a signal is attenuated and positive if a signal is amplified.

$$\text{dB} = 10 \log_{10} \frac{P_2}{P_1}$$

Variables  $P_1$  and  $P_2$  are the powers of a signal at points 1 and 2, respectively.

# Attenuation

**Example:** Suppose a signal travels through a transmission medium and its power is reduced to one-half. Find its attenuation.

**Solution:**

Clearly  $P_2 = P_1/2$ ,

Therefore ,

$$\begin{aligned}\text{Attenuation(dB)} &= 10 \log_{10} (P_2/P_1) = 10 \log_{10} (1/2) \\ &= 10 * (-0.3) = -3 \text{ dB}\end{aligned}$$

**Example:** A signal travels through an amplifier, and its power is increased 10 times. Find its attenuation.

**Solution:**

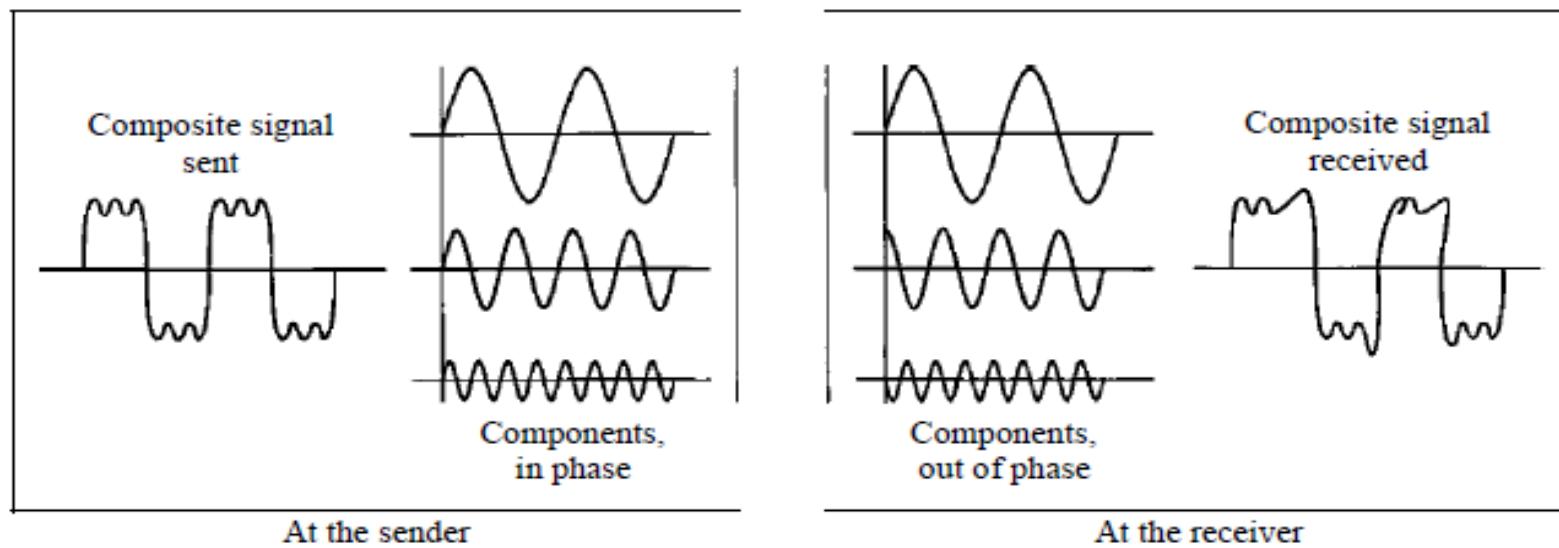
Clearly  $P_2 = 10P_1$ ,

Therefore ,

$$\begin{aligned}\text{Attenuation(dB)} &= 10 \log_{10} (P_2/P_1) \\ &= 10 * \log_{10}(10) \\ &= 10 \text{ dB}\end{aligned}$$

# Distortion

Distortion means that the signal changes its form or shape. Distortion can occur in a composite signal made of different frequencies. Each signal component has its own propagation speed (see the next section) through a medium and, therefore, its own delay in arriving at the final destination. Differences in delay may create a difference in phase if the delay is not exactly the same as the period duration. In other words, signal components at the receiver have phases different from what they had at the sender. The shape of the composite signal is therefore not the same. Following figure shows the effect of distortion on a composite signal.



# Noise

Noise is another cause of impairment. Several types of noise, such as thermal noise, induced noise, crosstalk, and impulse noise, may corrupt the signal.

**Thermal noise** is the random motion of electrons in a wire which creates an extra signal not originally sent by the transmitter.

**Induced noise** comes from sources such as motors and appliances. These devices act as a sending antenna, and the transmission medium acts as the receiving antenna.

**Crosstalk** is the effect of one wire on the other. One wire acts as a sending antenna and the other as the receiving antenna.

**Impulse noise** is a spike (a signal with high energy in a very short time) that comes from power lines, lightning, and so on.

# Noise

## Signal-to-Noise Ratio(SNR)

The signal-to-noise ratio is defined as

$$\text{SNR} = \frac{\text{average signal power}}{\text{average noise power}}$$

**Note:** A high SNR means the signal is less corrupted by noise; a low SNR means the signal is more corrupted by noise.

Because SNR is the ratio of two powers, it is often described in decibel units, SNR(dB), defined as

$$\text{SNR(dB)} = 10 \log_{10} \text{SNR}$$

**Example :**

The power of a signal is 10 mW and the power of the noise is 1  $\mu\text{W}$ ; what are the values of SNR and SNR(dB)?

**Example :** Find SNR and SNR(dB) for noiseless channel.

# Noiseless Channel: Nyquist Bit Rate

For a noiseless channel, the Nyquist bit rate formula defines the theoretical maximum bit rate.

$$\text{Bit Rate} = 2 \times B \times \log_2 L$$

In this formula, **B** is the bandwidth of the channel, **L** is the number of signal levels used to represent data, and **Bit Rate** is the bit rate in bits per second.

**Note:** Increasing the levels of a signal may reduce the reliability of the system.

**Example:** Consider a noiseless channel with a bandwidth of 3000 Hz transmitting a signal with two signal levels. Find the maximum bit rate.

**Solution:** Bit Rate =  $2 \times 3000 \times \log_2 2 = 6000 \text{ bps}$

# Noise

**Example:** We need to send 265 kbps over a noiseless channel with a bandwidth of 20 kHz. How many signal levels do we need?

**Solution:**

We can use the Nyquist formula as shown:

$$265,000 = 2 \times 20,000 \times \log_2 L$$

$$\log_2 L = 6.625$$

Therefore,  $L = 2^{6.625} = 98.7$  levels

Since this result is not a power of 2, we need to either increase the number of levels or reduce the bit rate. If we have 128 levels, the bit rate is 280 kbps. If we have 64 levels, the bit rate is 240 kbps.

# Noisy Channel: Shannon Capacity

In reality, we cannot have a noiseless channel; the channel is always noisy. In 1944, Claude Shannon introduced a formula, called the Shannon capacity, to determine the theoretical highest data rate for a noisy channel:

$$C = B \times \log_2 (1 + SNR)$$

In this formula, **B** is the bandwidth of the channel, **SNR** is the signal-to-noise ratio, and **C** is the capacity of the channel in bits per second.

**Note:** We cannot achieve a data rate higher than the capacity of the channel.

# Noisy Channel: Shannon Capacity

**Example:** A telephone line normally has a bandwidth of 3000 Hz (300 to 3300 Hz) assigned for data communications. The signal-to-noise ratio is usually 3162. Compute the capacity of this channel.

**Solution:**

$$\begin{aligned} C &= B \log_2 (1 + \text{SNR}) = 3000 \log_2 (1 + 3162) = 3000 \log_2 3163 \\ &= 3000 \times 11.62 = 34,860 \text{ bps} \end{aligned}$$

**Note:** This means that the highest bit rate for a telephone line is 34.860 kbps. If we want to send data faster than this, we can either increase the bandwidth of the line or improve the signal-to-noise ratio.

# Noisy Channel: Shannon Capacity

**Example:** The signal-to-noise ratio is often given in decibels. Assume that  $\text{SNR(dB)} = 36$  and the channel bandwidth is 2 MHz. Compute the theoretical channel capacity.

**Solution:**

We know that,  $\text{SNR(dB)} = 10 \log_{10} \text{SNR}$

Therefore,  $36 = 10 \log_{10} \text{SNR}$

$$\text{SNR} = 10^{3.6} = 3981$$

Now, Channel capacity  $C = B * \log_2(1+\text{SNR})$

$$= 2 * 10^6 \log_2(1+3981)$$

$$= 24 \text{ Mbps (approx.)}$$

**Note:** For practical purposes, when the SNR is very high, we can assume that  $\text{SNR} + 1$  is almost the same as  $\text{SNR}$ . In these cases, the theoretical channel capacity can be simplified to

$$C = B \times \text{SNR(dB)}/3$$

# Noisy Channel: Shannon Capacity

**Example:** We have a channel with a 1-MHz bandwidth. The SNR for this channel is 63. What are the appropriate bit rate and signal level?

**Solution:**

First, we use the Shannon formula to find the upper limit.

$$C = B * \log_2 (1 + \text{SNR}) = 10^6 * \log_2 (1 + 63) = 10^6 * \log_2 64 = 6 \text{ Mbps}$$

The Shannon formula gives us 6 Mbps, the upper limit. For better performance we choose something lower, 4 Mbps, for example. Then we use the Nyquist formula to find the number of signal levels.

$$\text{Bit rate} = 2 * B * \log_2 L$$

$$4 \times 10^6 = 2 \times 10^6 \times \log_2 L$$

Therefore,  $L = 4$

# Noisy Channel: Shannon Capacity

**Example:** If a binary signal is sent over a 3 kHz channel whose signal to noise ratio is 20dB. What is the maximum achievable data rate ?

**Solution:**

$$\text{SNR(dB)} = 10 * \log_{10} \text{SNR}$$

$$20 = 10 * \log_{10} \text{SNR}$$

$$\text{SNR} = 100$$

For maximum achievable data rate

$$\begin{aligned} C &= B * \log_2 (1 + \text{SNR}) = 3000 * \log_2 (1 + 100) = 3000 * \log_2 101 \\ &= 3000 * 6.658 = 19.974 \text{ kbps} \end{aligned}$$

$$\begin{aligned} \text{Nyquist Bit rate} &= 2 * B * \log_2 L = 2 * 3000 * \log_2 2 \\ &= 6000 \text{ bps} \end{aligned}$$

The bottleneck is therefore the Nyquist limit giving a maximum channel capacity of **6** kbps.

# PERFORMANCE

## Bandwidth

One characteristic that measures network performance is bandwidth. However, the term can be used in two different contexts with two different measuring values: **bandwidth in hertz** and **bandwidth in bits per second**.

### Bandwidth in Hertz

Bandwidth in hertz is the range of frequencies contained in a composite signal or the range of frequencies a channel can pass. For example, we can say the bandwidth of a subscriber telephone line is 4 kHz.

### Bandwidth in Bits per Seconds

The term bandwidth can also refer to the number of bits per second that a channel, a link, or even a network can transmit. For example, one can say the bandwidth of a Fast Ethernet network (or the links in this network) is a maximum of 100 Mbps. This means that this network can send 100 Mbps.

# PERFORMANCE

## Throughput

- The throughput is a measure of how fast we can actually send data through a network.
- For example, we may have a link with a bandwidth of 1 Mbps, but the devices connected to the end of the link may handle only 200 kbps. This means that we cannot send more than 200 kbps through this link.
- Imagine a highway designed to transmit 1000 cars per minute from one point to another. However, if there is congestion on the road, this figure may be reduced to 100 cars per minute. The bandwidth is 1000 cars per minute; the throughput is 100 cars per minute.

**Note:** The bandwidth is a potential measurement of a link; the throughput is an actual measurement of how fast we can send data.

# PERFORMANCE

## Example :

A network with bandwidth of 10 Mbps can pass only an average of 12,000 frames per minute with each frame carrying an average of 10,000 bits. What is the throughput of this network?

## Solution:

We can calculate the throughput as

$$\text{Throughput} = \frac{12,000 \times 10,000}{60} = 2 \text{ Mbps}$$

The throughput is almost one-fifth of the bandwidth in this case.

# PERFORMANCE

## Latency (Delay)

The latency or delay defines how long it takes for an entire message to completely arrive at the destination from the time the first bit is sent out from the source.

Latency is made of four components: propagation time, transmission time, queuing time and processing time.

$$\text{Latency} = \text{propagation time} + \text{transmission time} + \text{queuing time} \\ + \text{processing time}$$

## Propagation Time

Propagation time measures the time required for a bit to travel from the source to the destination.

$$\text{Propagation time} = \frac{\text{Distance}}{\text{Propagation speed}}$$

# PERFORMANCE

**Example:**

What is the propagation time if the distance between the two points is 12,000 km? Assume the propagation speed to be  $2.4 \times 10^8$ m/s in cable.

**Solution:**

We can calculate the propagation time as

$$\text{Propagation time} = \frac{12000 \times 1000}{2.4 \times 10^8} = 50 \text{ ms}$$

## Transmission time

The time required for transmission of a message depends on the size of the message and the bandwidth of the channel.

$$\text{Transmission time} = \frac{\text{Message size}}{\text{Bandwidth}}$$

# PERFORMANCE

Example:

What are the propagation time and the transmission time for a 2.5-kbyte message (an e-mail) if the bandwidth of the network is 1 Gbps? Assume that the distance between the sender and the receiver is 12,000 km and that light travels at  $2.4 \times 10^8$ m/s.

Solution:

We can calculate the propagation and transmission time as

$$\text{Propagation time} = \frac{12000 \times 1000}{2.4 \times 10^8} = 50 \text{ ms}$$

$$\text{Transmission time} = \frac{2.5 \times 1000 \times 8}{10^9} = 0.020 \text{ ms}$$

# PERFORMANCE

## Queuing Time

It is the time needed for each intermediate or end device to hold the message before it can be processed.

- The queuing time is not a fixed factor; it changes with the load imposed on the network.
- When there is heavy traffic on the network, the queuing time increases. An intermediate device, such as a router, queues the arrived messages and processes them one by one. If there are many messages, each message will have to wait.

## Processing time

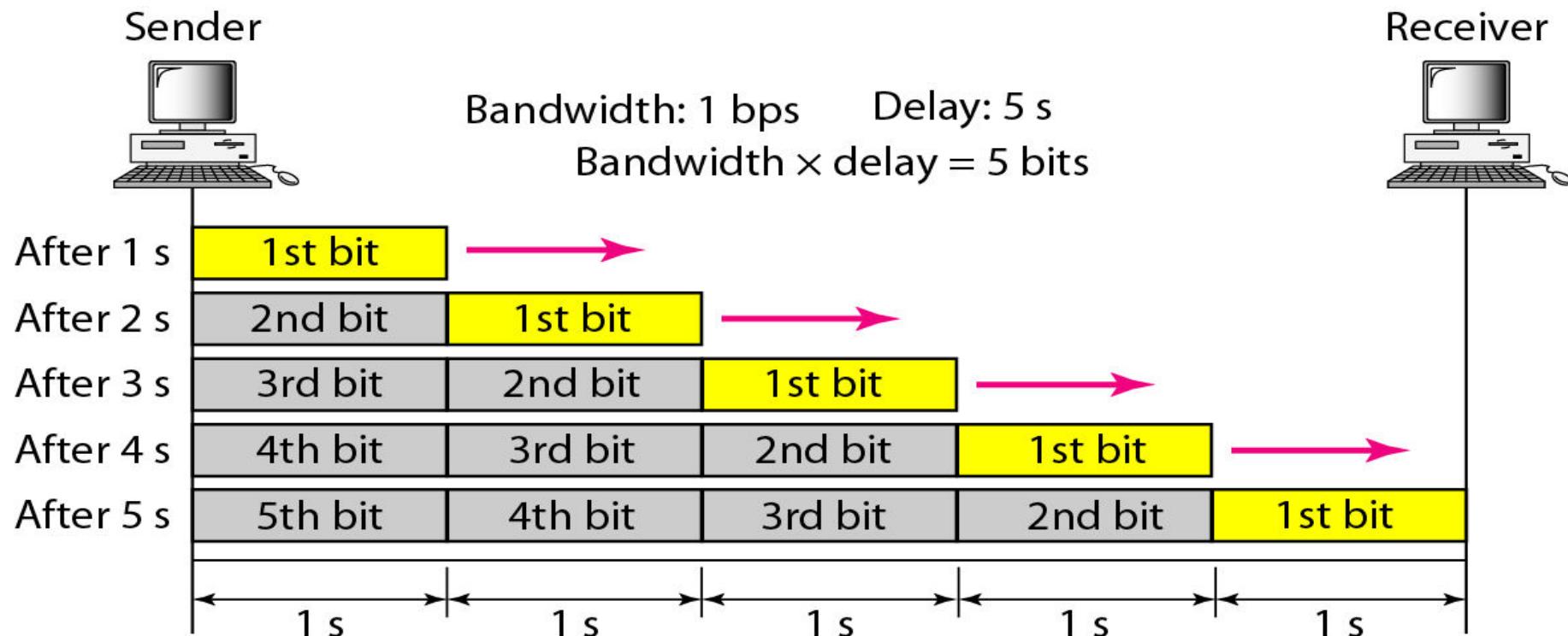
It is the time taken by intermediate or end devices to process the arrived message.

# PERFORMANCE

## Bandwidth-delay product

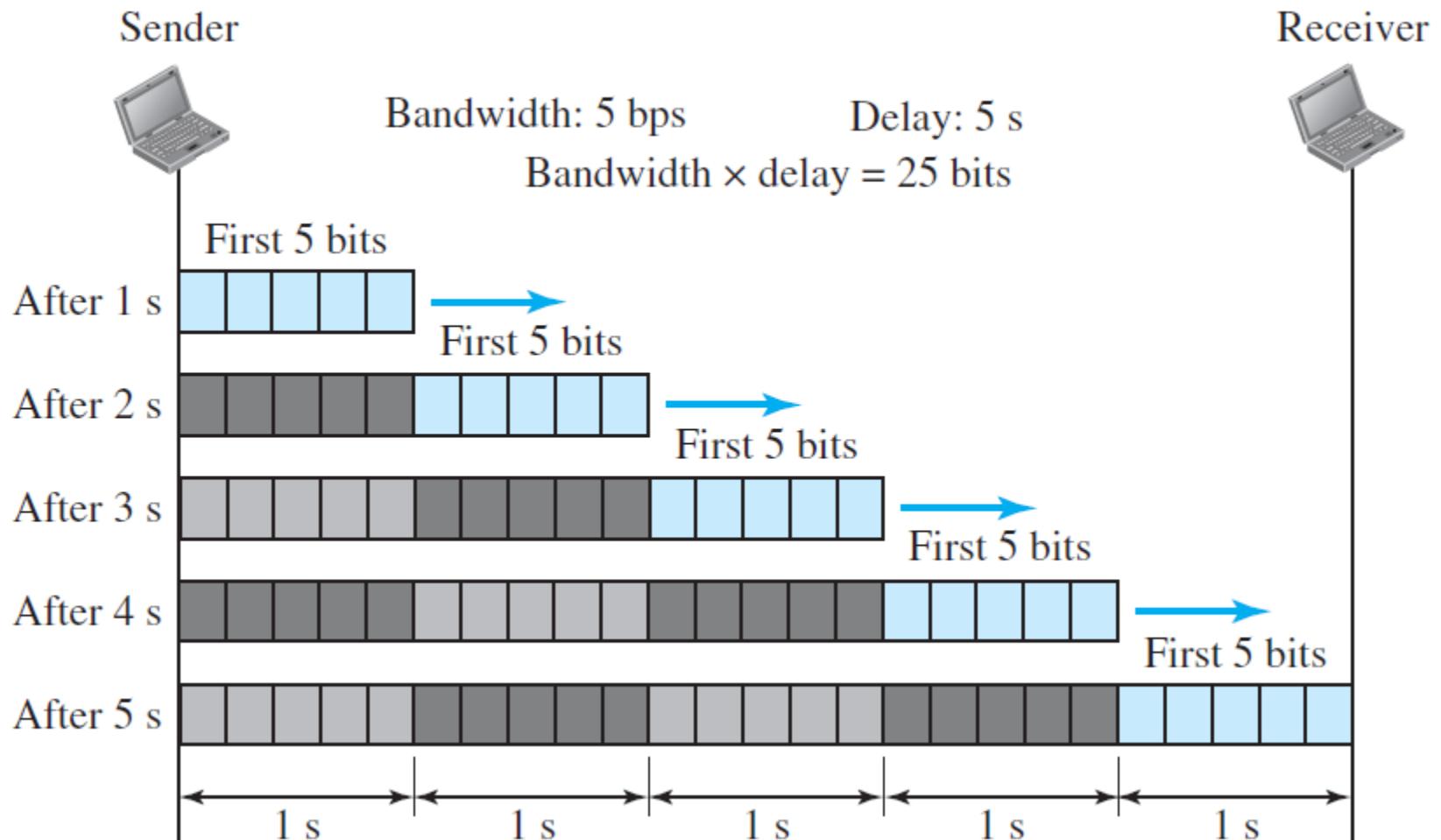
The bandwidth-delay product defines the number of bits that can fill the link.

$$\text{Bandwidth-delay product} = \text{Bandwidth} * \text{Delay}$$



# PERFORMANCE

## Bandwidth-delay product



# PERFORMANCE

**Question:** A device is sending out data at the rate of 1000 bps.

- a. How long does it take to send out 10 bits?
- b. How long does it take to send out a single character (8 bits)?
- c. How long does it take to send a file of 100,000 characters?

**Question:** A file contains 2 million bytes. How long does it take to download this file using a 56-Kbps channel? 1-Mbps channel?

**Question:** How many bits can fit on a link with a 2 ms delay if the bandwidth of the link is

- a. 1 Mbps?
- b. 10 Mbps?
- c. 100 Mbps?

# PERFORMANCE

**Question:** We are sending a 30 Mbits MP3 file from a source host to a destination host. All links in the path between source and destination have a transmission rate of 10 Mbps. Assume that the propagation speed is  $2 \times 10^8$  meters/sec, and the distance of each link is 10,000 km, the processing time of the router is 0.01 sec.

- a) Suppose there is only one link between source and destination. Find the latency?
- b) Suppose there is three links between source and destination. The router is found between each two links  
Find the latency?

# PERFORMANCE

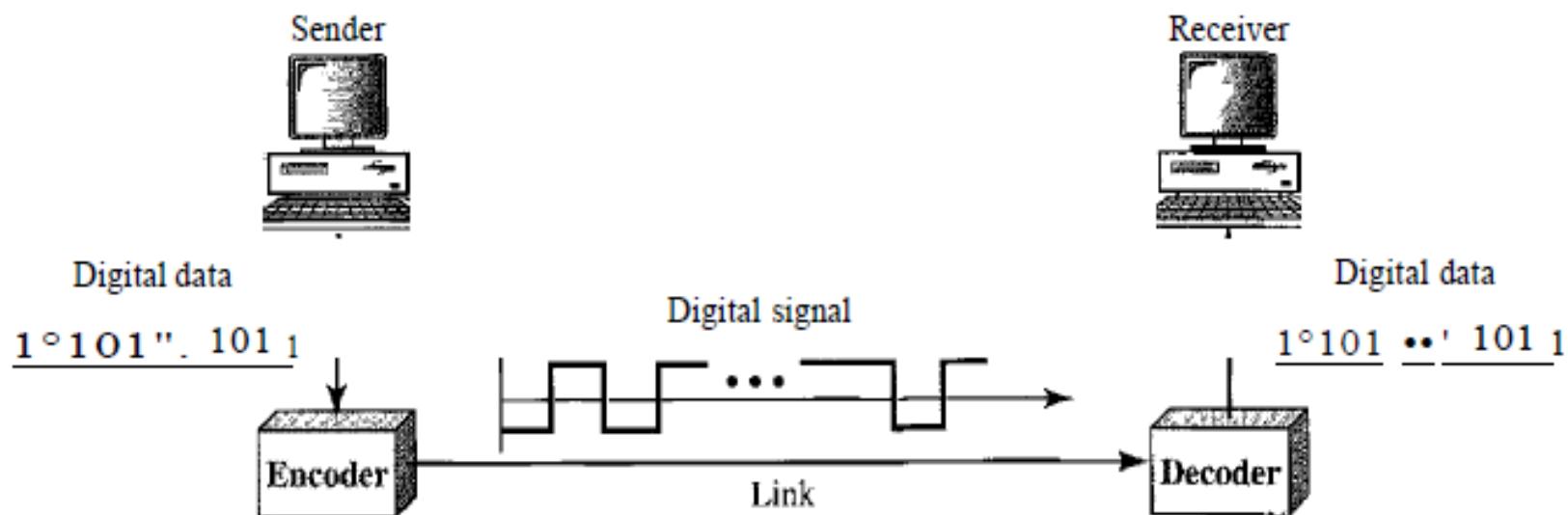
**Question:** What is the total delay (latency) for a frame of size 5 million bits that is being sent on a link with 10 routers each having a queuing time of  $2 \mu\text{s}$  and a processing time of  $1 \mu\text{s}$ . The length of the link is 2000 Km. The speed of light inside the link is  $2 \times 10^8 \text{ m/s}$ . The link has a bandwidth of 5 Mbps. Which component of the total delay is dominant? Which one is negligible?

# Encoding Schemes

## Line Coding Schemes:

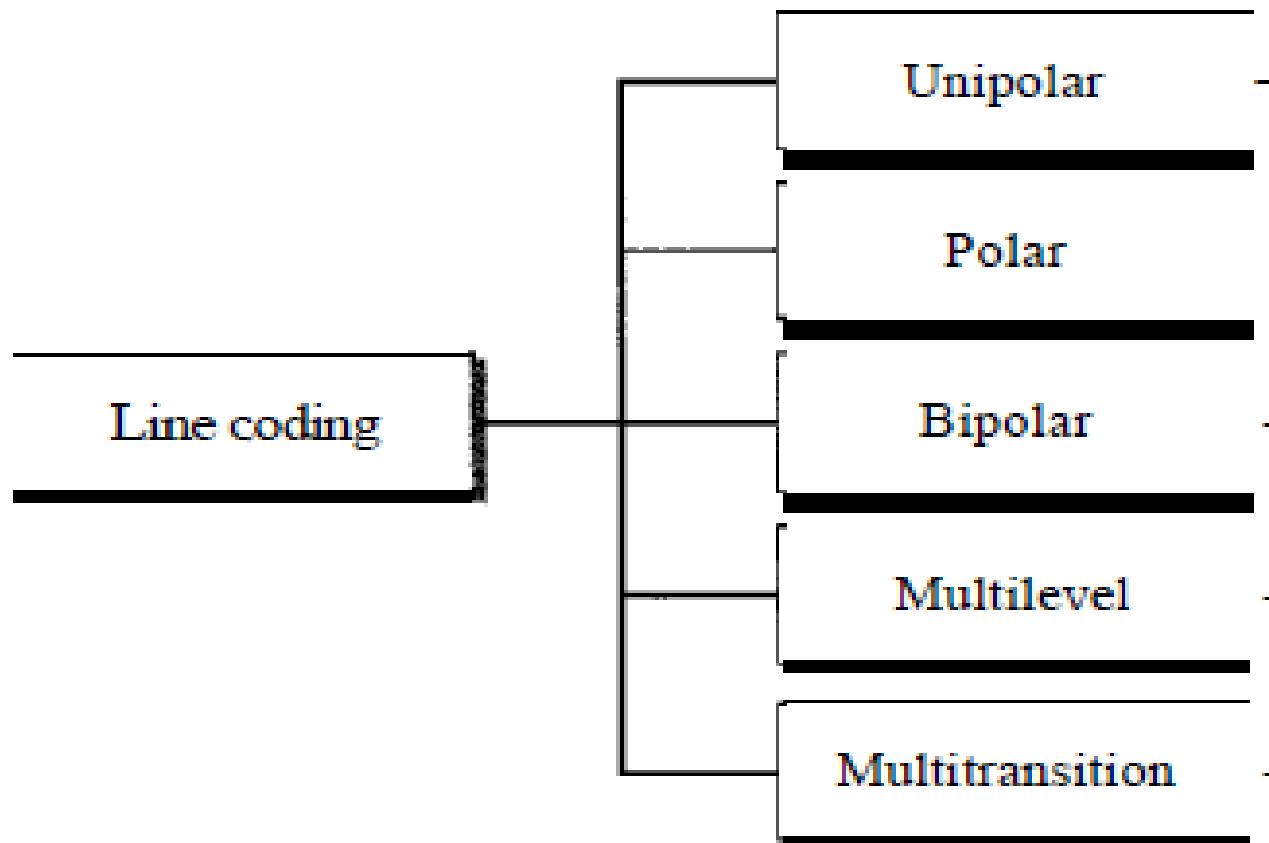
Line coding is the process of converting digital data to digital signals. We assume that data, in the form of text, numbers, graphical images, audio, or video, are stored in computer memory as sequences of bits.

Line coding converts a sequence of bits to a digital signal. At the sender, digital data are encoded into a digital signal; at the receiver, the digital data are recreated by decoding the digital signal. Following figure shows the process.



# Encoding Schemes

We can roughly divide line coding schemes into five broad categories, as the following:-



# Encoding Schemes

## Unipolar Scheme:

In a unipolar scheme, all the signal levels are on one side of the time axis, either above or below.

**NRZ (Non-Return-to-Zero):** Traditionally, a unipolar scheme was designed as a non-return-to-zero (NRZ) scheme in which the positive voltage defines bit 1 and the zero voltage defines bit 0. It is called NRZ because the signal does not return to zero at the middle of the bit. Following figure shows a unipolar NRZ scheme.



# Encoding Schemes

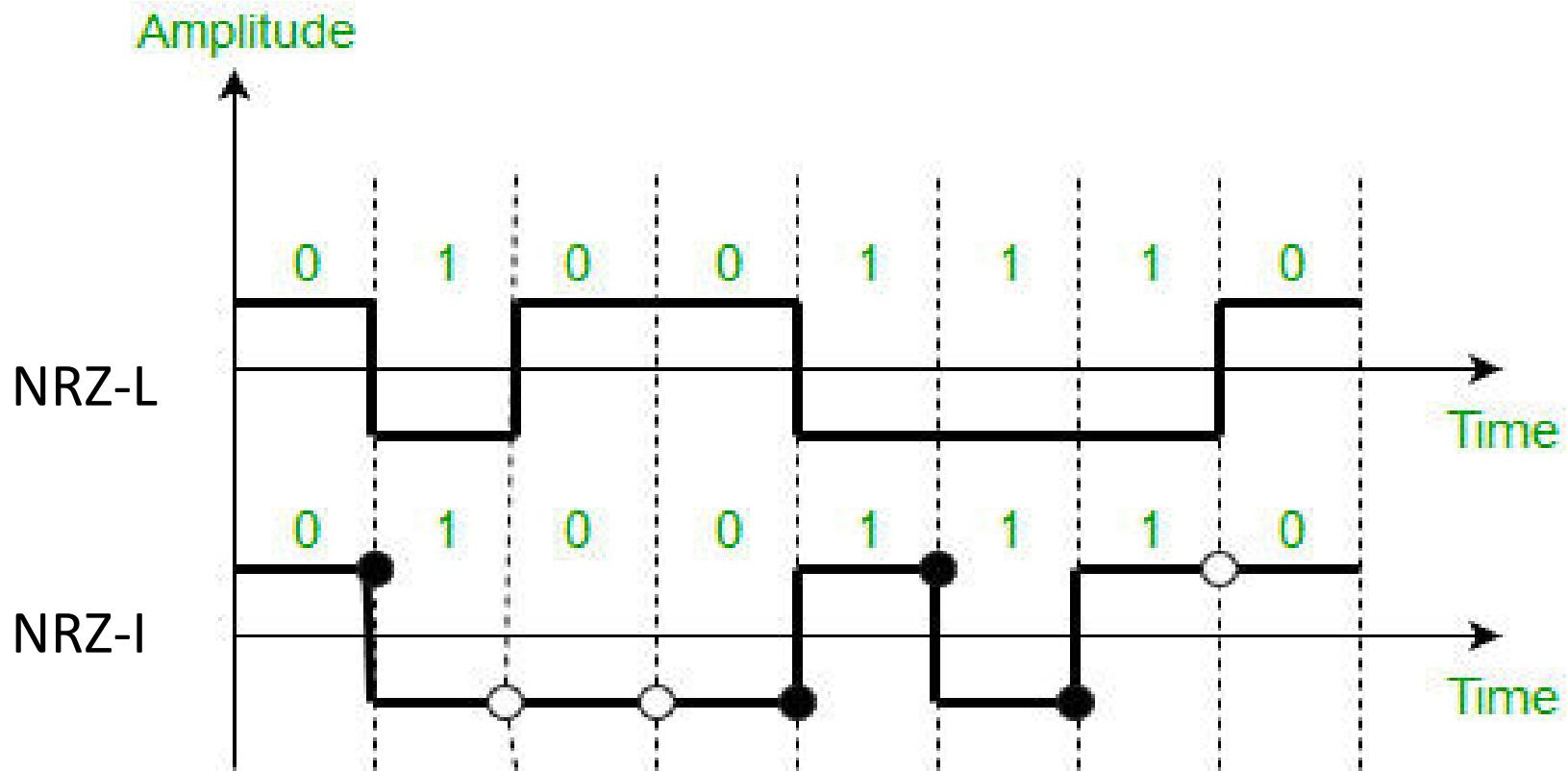
## Polar Schemes:

In polar schemes, the voltages are on the both sides of the time axis. For example, the voltage level for 0 can be positive and the voltage level for 1 can be negative.

## Non-Return-to-Zero (NRZ):

- In polar NRZ encoding, we use two levels of voltage amplitude. There are two versions of polar NRZ: NRZ-L and NRZ-I.
- In the first variation, NRZ-L (NRZ-Level), the level of the voltage determines the value of the bit.
- In the second variation, NRZ-I (NRZ-Invert), the change in the level of the voltage determines the value of the bit. If there is no change, the bit is 0; if there is a change, the bit is 1.

# Encoding Schemes

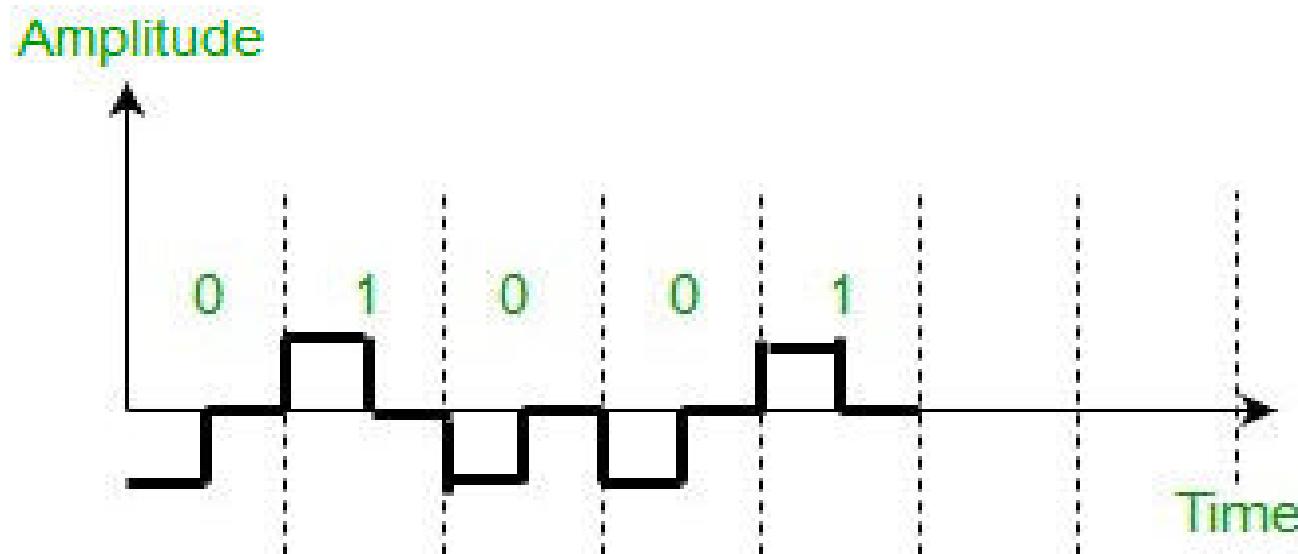


**Note:** In NRZ-L the level of the voltage determines the value of the bit. In NRZ-I the inversion or the lack of inversion determines the value of the bit.

# Encoding Schemes

## Return to Zero (RZ):

Return-to-zero (RZ) scheme uses three values: positive, negative, and zero. In RZ, the signal changes not between bits but during the bit. It remains there until the beginning of the next bit.

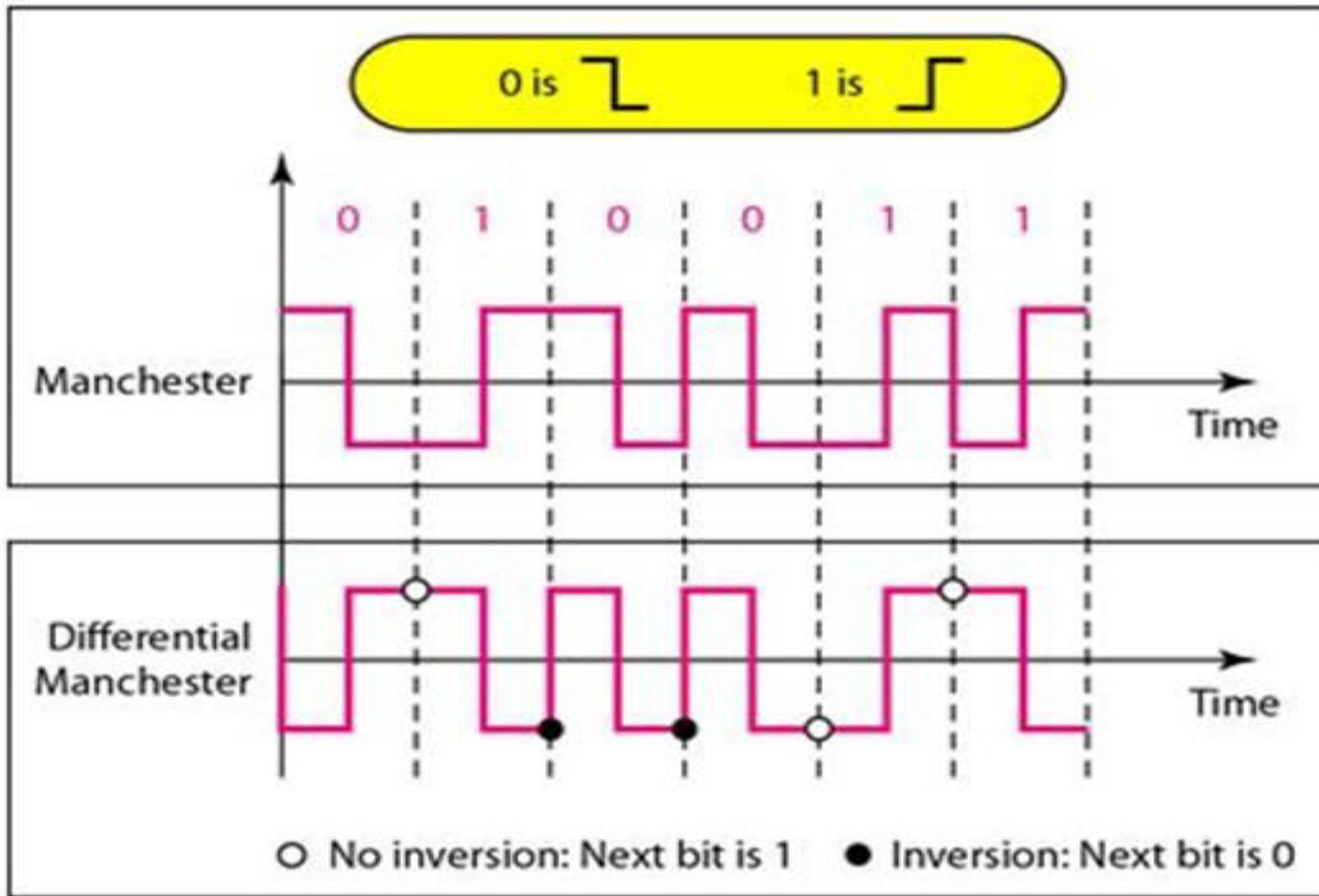


# Encoding Schemes

## Biphase: Manchester and Differential Manchester:

- ❖ The idea of RZ and the idea of NRZ-L are combined into the Manchester scheme. In Manchester encoding, the duration of the bit is divided into two halves. The voltage remains at one level during the first half and moves to the other level in the second half.
- ❖ Differential Manchester, on the other hand, combines the ideas of RZ and NRZ-I. There is always a transition at the middle of the bit, but the bit values are determined at the beginning of the bit. If the next bit is 0, there is a transition; if the next bit is 1, there is none.

# Encoding Schemes



# Encoding Schemes

**Question:** Sketch Manchester and differential Manchester encoding for the following bit stream:  
10111100010010011101

# Encoding Schemes

## Bipolar Schemes:

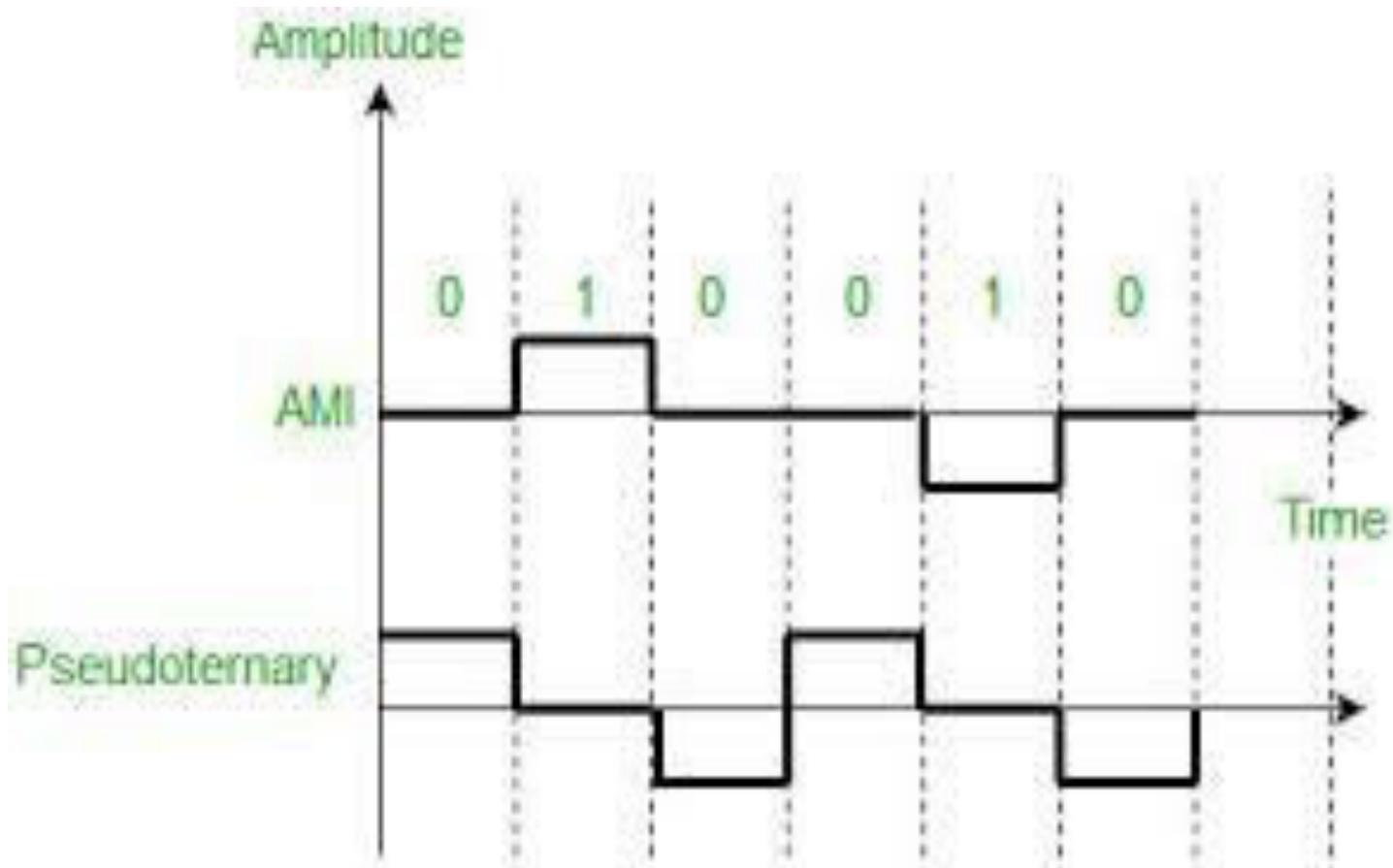
In bipolar encoding (sometimes called multilevel binary), there are three voltage levels: positive, negative, and zero. The voltage level for one data element is at zero, while the voltage level for the other element alternates between positive and negative.

## AMI and Pseudoternary:

There are two variations of bipolar encoding: AMI and pseudoternary.

- ❖ A common bipolar encoding scheme is called bipolar alternate mark inversion (AMI). In the term alternate mark inversion, the word mark comes from telegraphy and means 1. So AMI means alternate 1 inversion. A neutral zero voltage represents binary 0. Binary 1's are represented by alternating positive and negative voltages.
- ❖ A variation of AMI encoding is called pseudoternary in which the 1 bit is encoded as a zero voltage and the 0 bit is encoded as alternating positive and negative voltages.

# Encoding Schemes



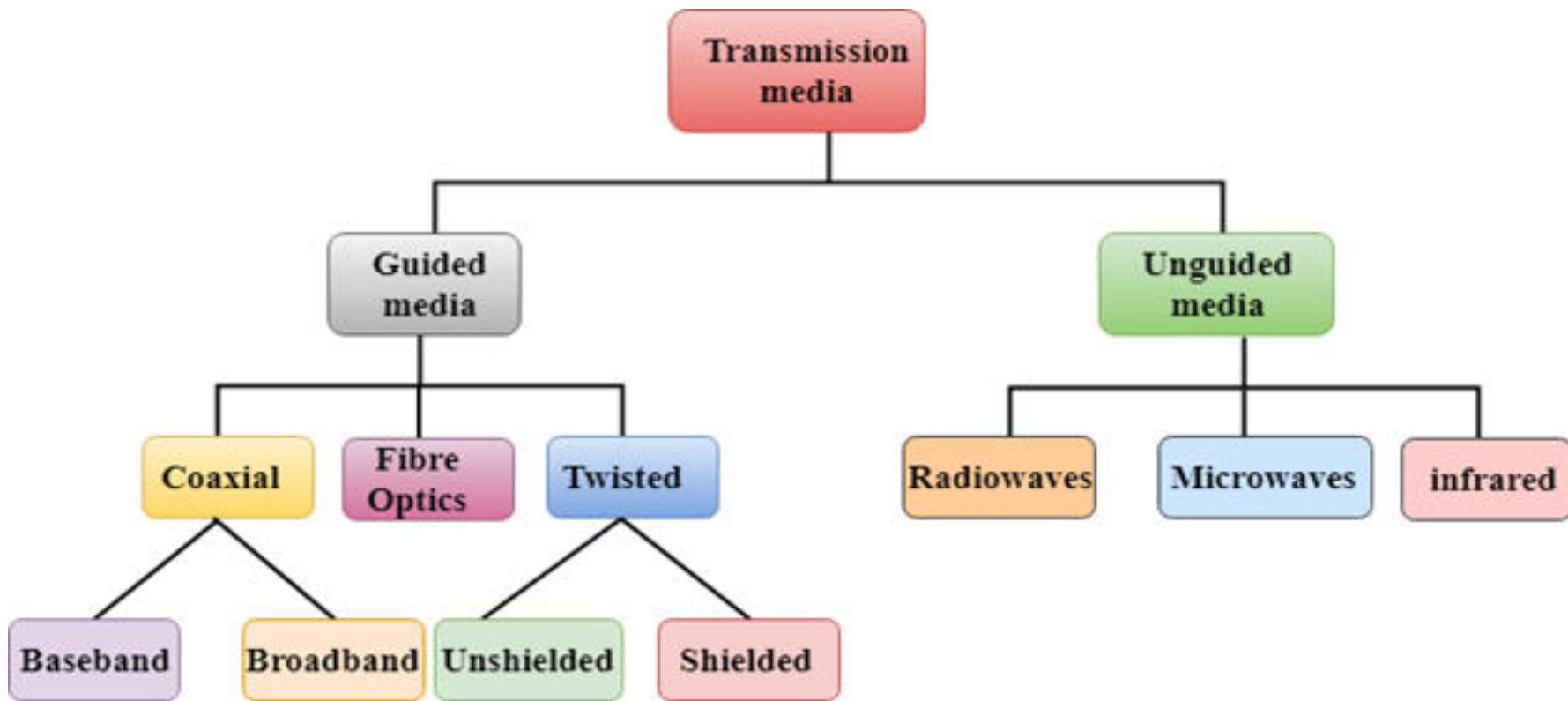
# Encoding Schemes

**Question:** Encode the data-stream 10011010 using the following encoding scheme:

- i. Unipolar
- ii. Bipolar NRZ-L
- iii. Bipolar NRZ-I
- iv. RZ
- v. Manchester
- vi. Differential Manchester
- vii. AMI

# Transmission Media

- ❖ A transmission medium can be broadly defined as anything that can carry information from a source to a destination.
- ❖ The transmission medium is usually free space, metallic cable, or fiber-optic cable.



# **GUIDED MEDIA**

Guided media, which are those that provide a conduit from one device to another, include twisted-pair cable, coaxial cable, and fiber-optic cable. A signal traveling along any of these media is directed and contained by the physical limits of the medium. Twisted-pair and coaxial cable use metallic (copper) conductors that accept and transport signals in the form of electric current. Optical fiber is a cable that accepts and transports signals in the form of light.

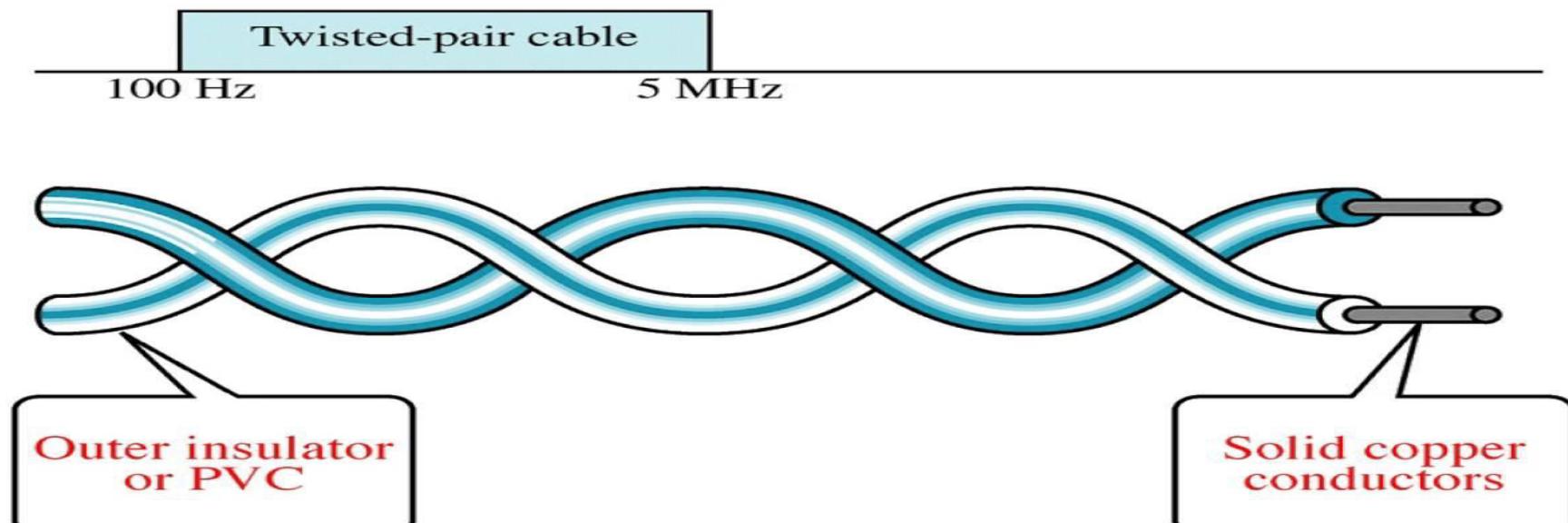
# GUIDED MEDIA

## Twisted-Pair Cable

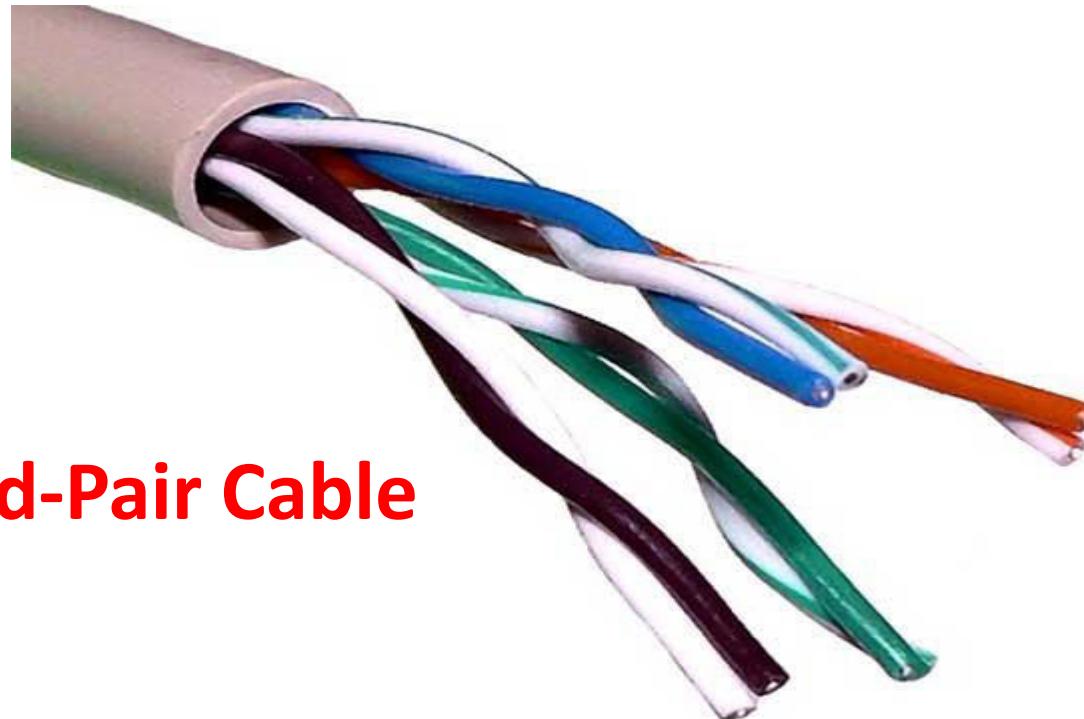
A twisted pair cable is made of two plastic insulated copper wires twisted together to form a single media. Out of these two wires, only one carries actual signal and another is used for ground reference.

The twists between wires are helpful in reducing noise (electromagnetic interference) and crosstalk.

The receiver uses the difference between the two.



# GUIDED MEDIA



**Twisted-Pair Cable**

# GUIDED MEDIA

There are two types of twisted pair cables:

- ❖ Unshielded Twisted Pair (UTP) Cable
- ❖ Shielded Twisted Pair (STP) Cable

## Unshielded Twisted Pair (UTP) Cable

The most common twisted-pair cable used in communications is referred to as unshielded twisted-pair (UTP).

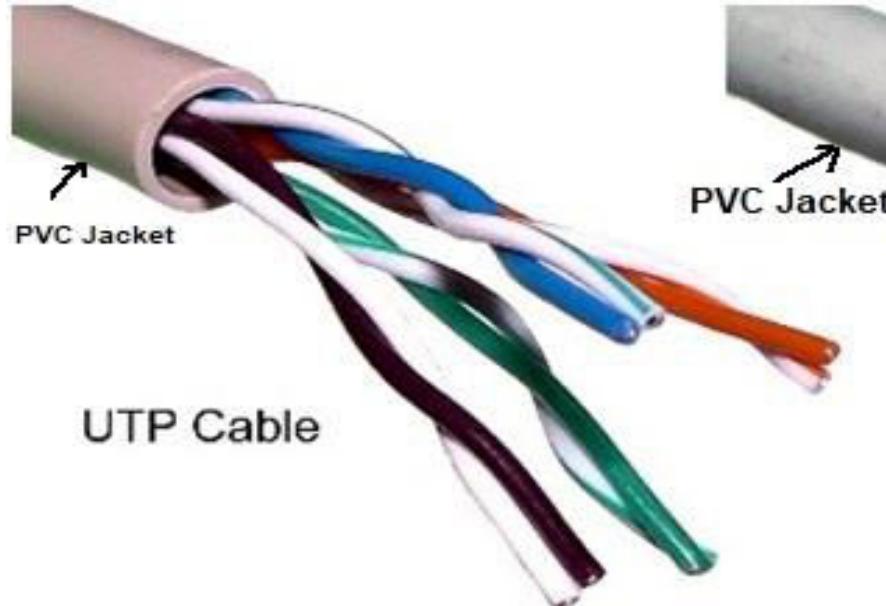
The Electronic Industries Association (EIA) has developed standards to classify unshielded twisted-pair cable into seven categories. Categories are determined by cable quality, with 1 as the lowest and 7 as the highest. Each EIA category is suitable for specific uses.

In computer networks, Cat-5, Cat-5e, and Cat-6 cables are mostly used. UTP cables are connected by RJ45 connectors.

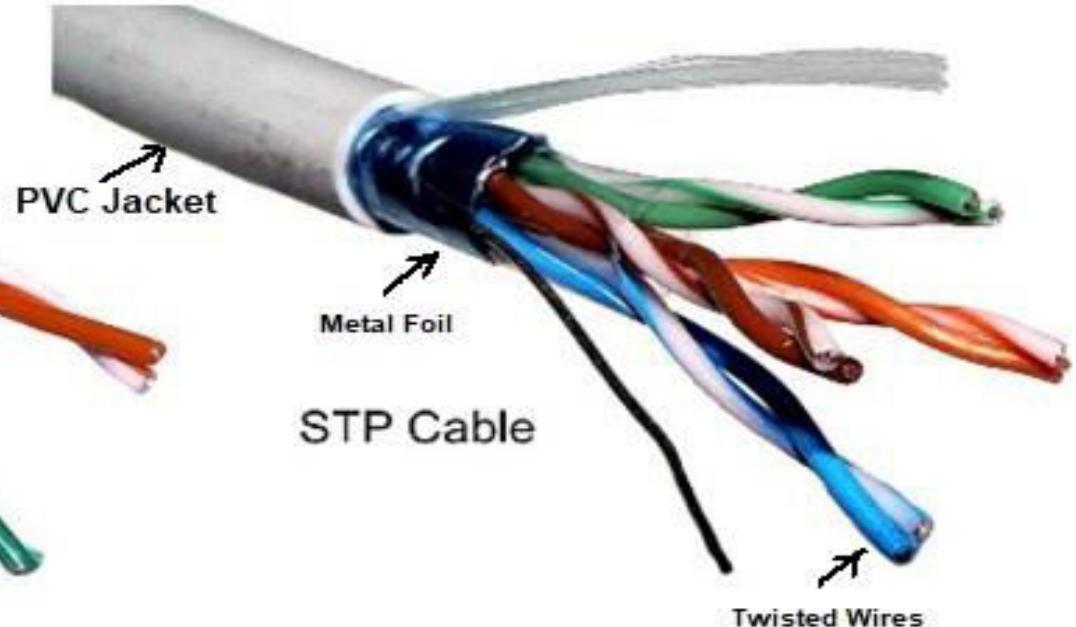
## Shielded Twisted Pair (STP) Cable

STP cable has a metal foil or braided-mesh covering that encases each pair of insulated conductors. Metal casing improves the quality of cable by preventing the penetration of noise or crosstalk.

# GUIDED MEDIA



UTP Cable

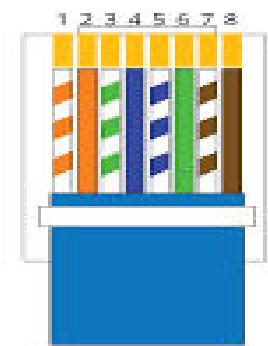
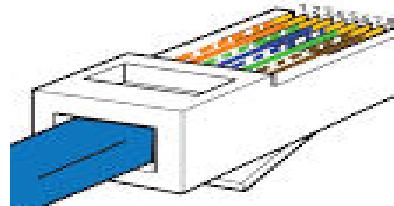


STP Cable



RJ45  
connector

**RJ45 Pinout**  
T-568B

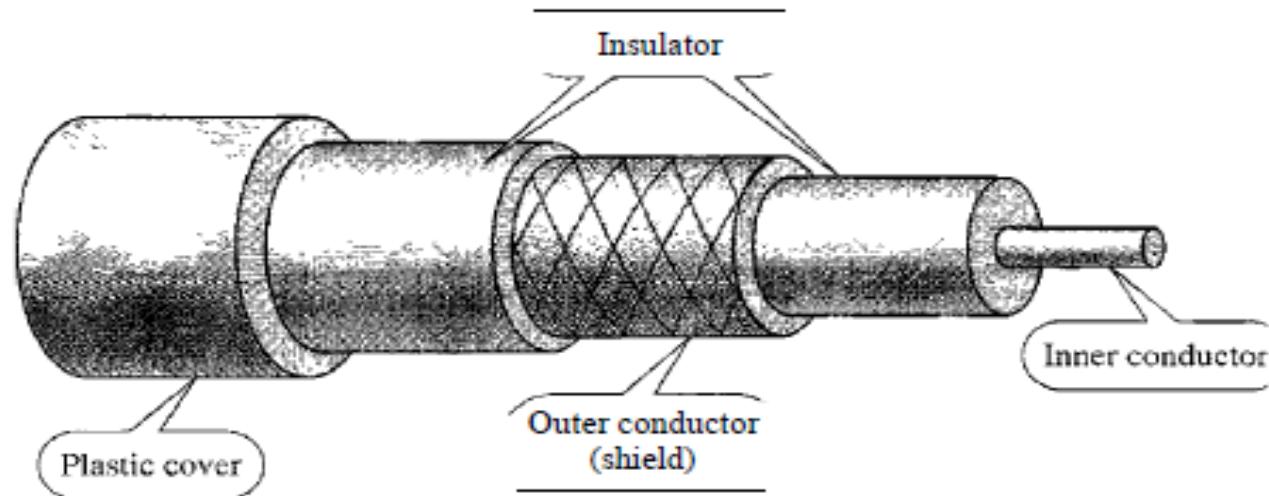


- |                 |                |
|-----------------|----------------|
| 1. White Orange | 5. White Blue  |
| 2. Orange       | 6. Green       |
| 3. White Green  | 7. White Brown |
| 4. Blue         | 8. Brown       |

# Guided Media

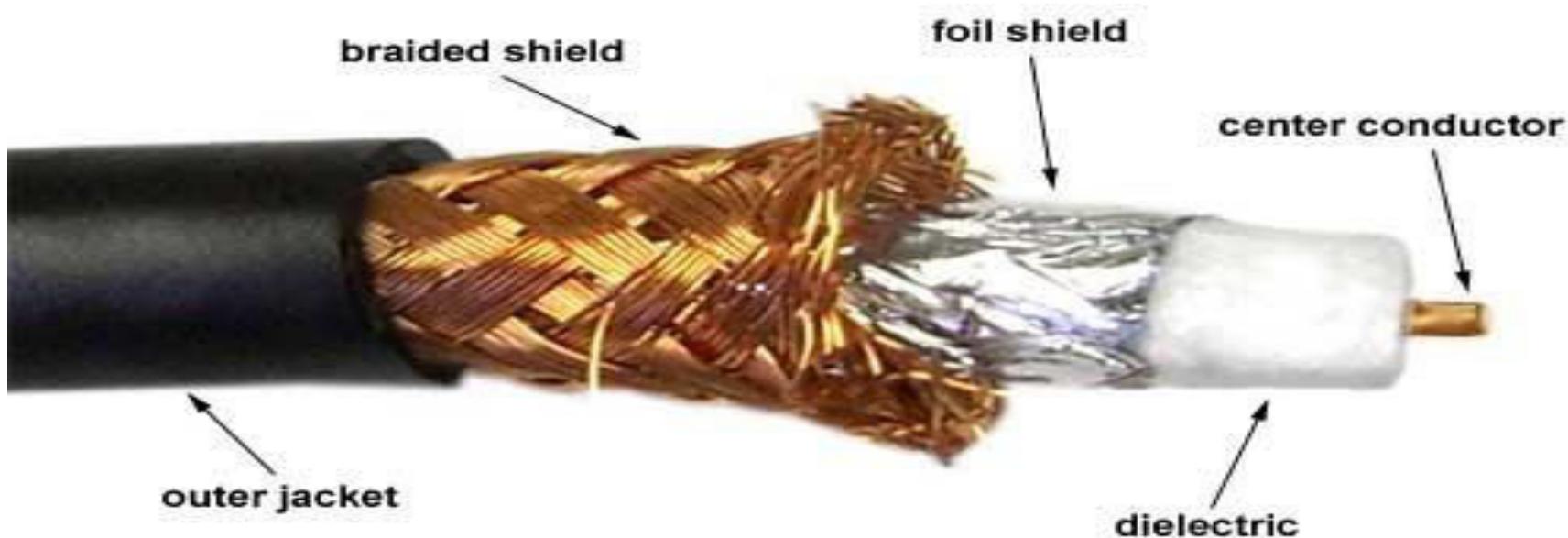
## Coaxial Cable

Coaxial cable (or coax) carries signals of higher frequency ranges than those in twisted pair cable Instead of having two wires. Coax has a central core conductor of solid or stranded wire (usually copper) enclosed in an insulating sheath, which is, in turn, encased in an outer conductor of metal foil, braid, or a combination of the two. The outer metallic wrapping serves both as a shield against noise and as the second conductor, which completes the circuit. This outer conductor is also enclosed in an insulating sheath, and the whole cable is protected by a plastic cover.



# Guided Media

## COAXIAL CABLE



# Guided Media

## Coaxial Cable Standards

Coaxial cables are categorized by their radio government (RG) ratings. Each RG number denotes a unique set of physical specifications, including the wire gauge of the inner conductor, the thickness and type of the inner insulator, the construction of the shield, and the size and type of the outer casing. Each cable defined by an RG rating is adapted for a specialized function.

Category	Use
RG-59	Cable TV
RG-58	Thin Ethernet
RG-11	Thick Ethernet

# Guided Media

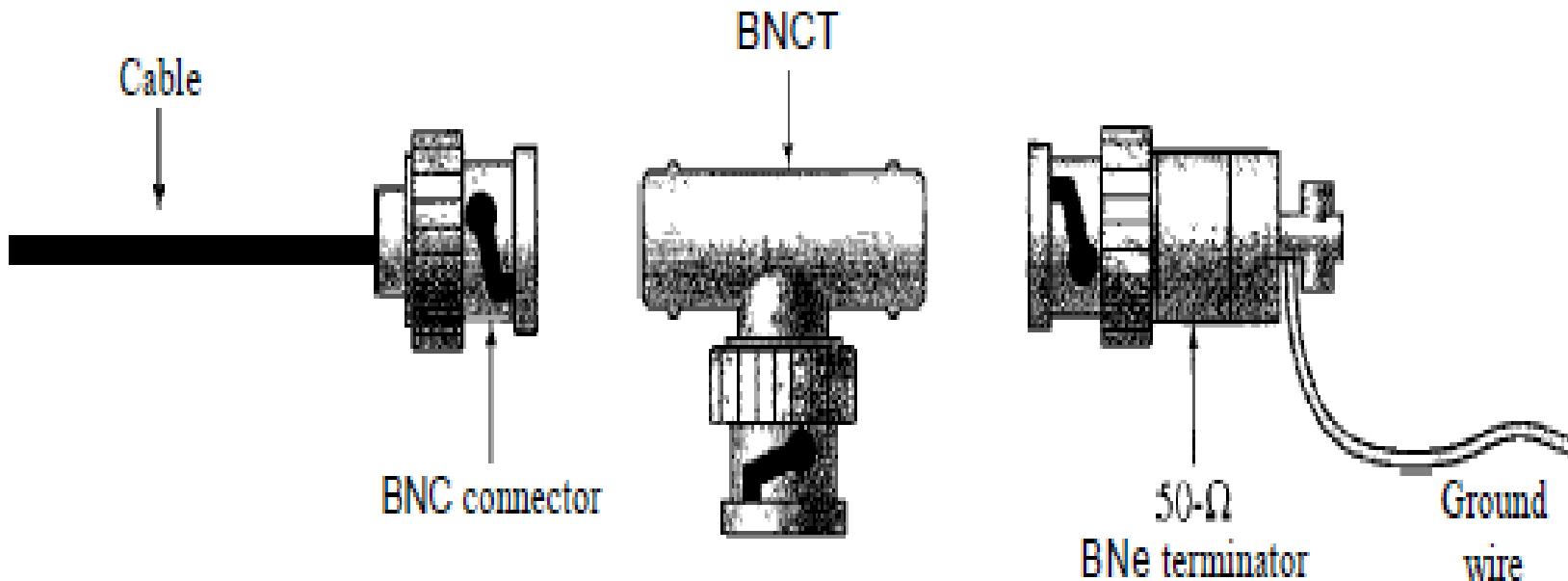
## Coaxial Cable Connectors

The most common type of connector used today is the Bayone-Neill-Conelman (BNC), connector. Three popular types of these connectors are the BNC connector, the BNC-T connector, and the BNC terminator.

- ❖ The BNC connector is used to connect the end of the cable to a device, such as a TV set.
- ❖ The BNC T connector is used in Ethernet networks to branch out to a connection to a computer or other device.
- ❖ The BNC terminator is used at the end of the cable to prevent the reflection of the signal.

# Guided Media

## Coaxial Cable Connectors

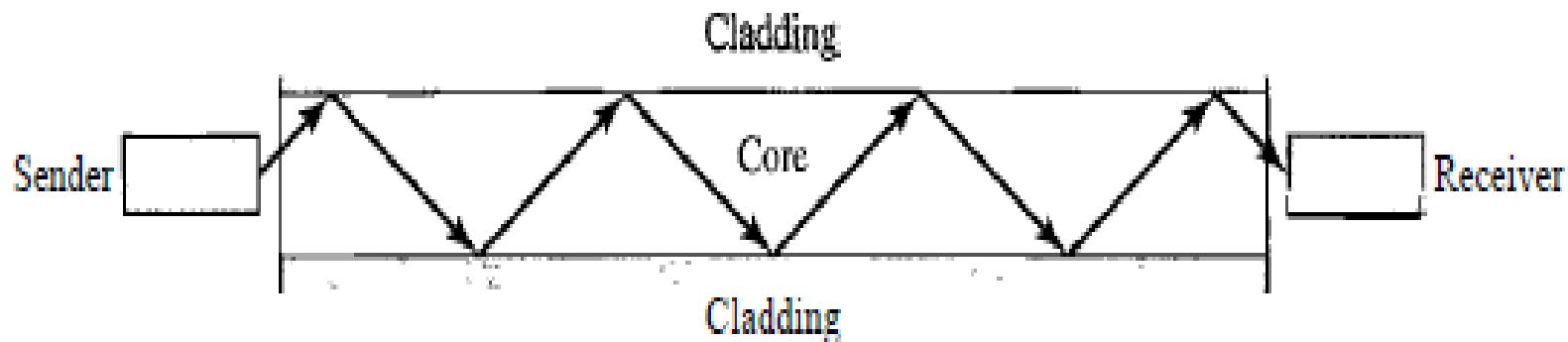


# Guided Media

## Fiber-Optic Cable

A fiber-optic cable is made of glass or plastic and transmits signals in the form of light.

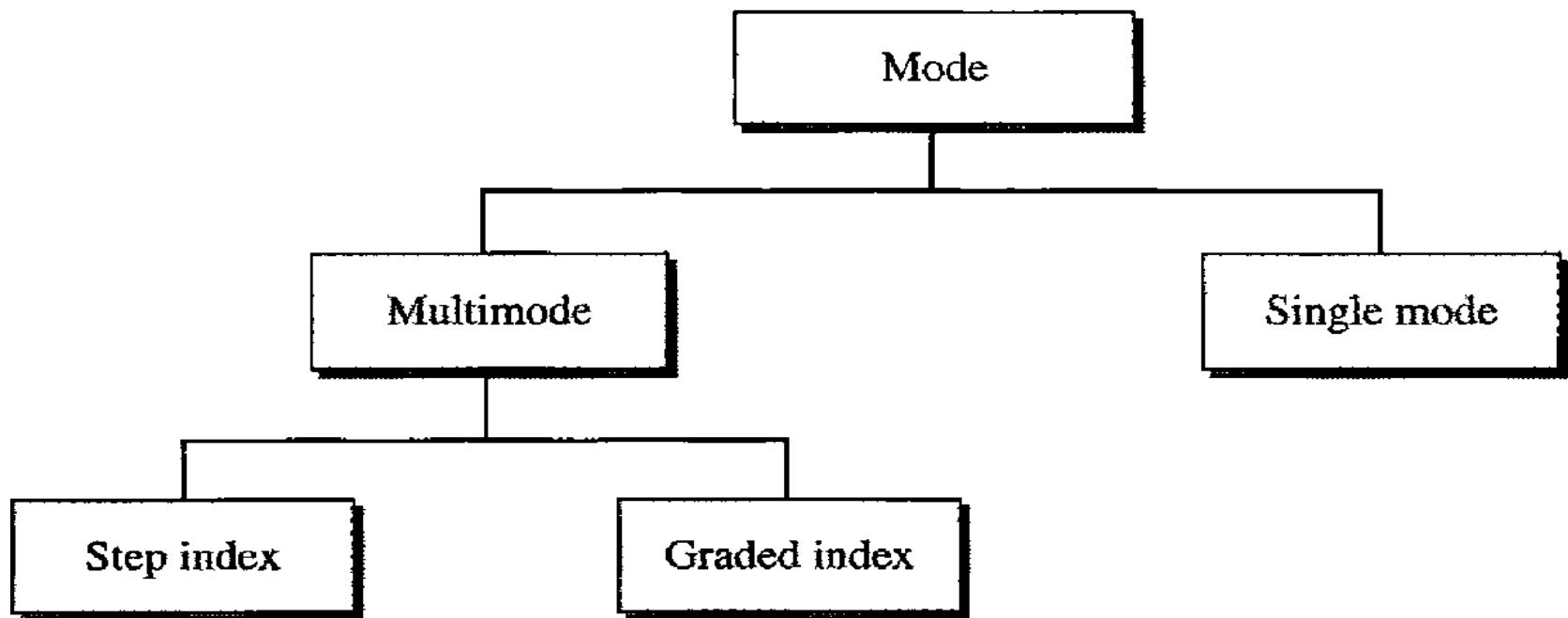
Optical fibers use reflection to guide light through a channel. A glass or plastic core is surrounded by a cladding of less dense glass or plastic. The difference in density of the two materials must be such that a beam of light moving through the core is reflected off the cladding instead of being refracted into it.



# Guided Media

## Propagation Modes

Current technology supports two modes (multimode and single mode) for propagating light along optical channels, each requiring fiber with different physical characteristics. Multimode can be implemented in two forms: step-index or graded-index.



# Guided Media

## Multimode

In this mode, multiple beams from a light source move through the core in different paths.

In **multimode step-index fiber**, the density of the core remains constant from the center to the edges. A beam of light moves through this constant density in a straight line until it reaches the interface of the core and the cladding. At the interface, there is an abrupt change due to a lower density; this alters the angle of the beam's motion. The term step index refers to the suddenness of this change, which contributes to the distortion of the signal as it passes through the fiber.

A **graded-index fiber** is one with varying densities. Density is highest at the center of the core and decreases gradually to its lowest at the edge.

# Guided Media

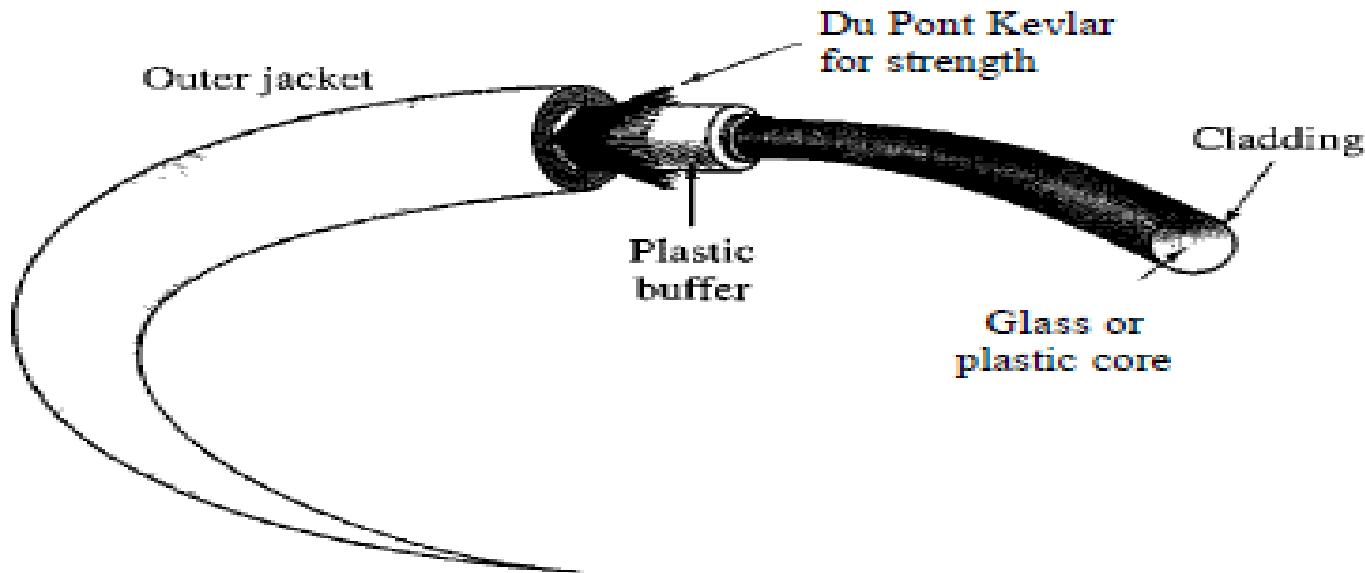
## Single-Mode

Single-mode uses step-index fiber and a highly focused source of light that limits beams to a small range of angles, all close to the horizontal.

# Guided Media

## Cable Composition

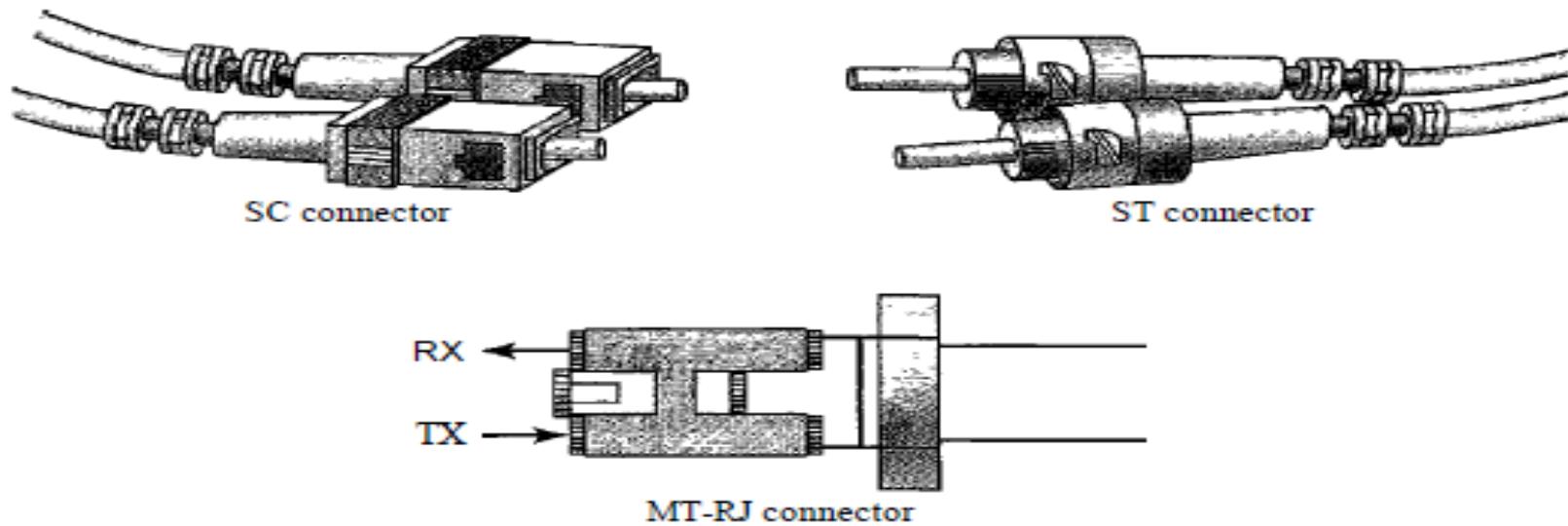
Following figure shows the composition of a typical fiber-optic cable. The outer jacket is made of either PVC or Teflon. Inside the jacket are Kevlar strands to strengthen the cable. Kevlar is a strong material used in the fabrication of bulletproof vests. Below the Kevlar is another plastic coating to cushion the fiber. The fiber is at the center of the cable, and it consists of cladding and core.



# Guided Media

## Fiber-Optic Cable Connectors

There are three types of connectors for fiber-optic cables, as shown in figure.



The subscriber channel (SC) connector is used for cable TV. It uses a push/pull locking system. The straight-tip (ST) connector is used for connecting cable to networking devices. It uses a bayonet locking system and is more reliable than SC. MT-RJ is a connector that is the same size as RJ45.

# Guided Media

## Advantages of Optical Fiber

Fiber-optic cable has several advantages over metallic cable (twisted pair or coaxial).

- ❖ **Higher bandwidth.** Fiber-optic cable can support dramatically higher bandwidths (and hence data rates) than either twisted-pair or coaxial cable.
- ❖ **Less signal attenuation.** Fiber-optic transmission distance is significantly greater than that of other guided media. A signal can run for 50 km without requiring regeneration. We need repeaters every 5 km for coaxial or twisted-pair cable.
- ❖ **Immunity to electromagnetic interference.** Electromagnetic noise cannot affect fiber-optic cables.
- ❖ **Resistance to corrosive materials.** Glass is more resistant to corrosive materials than copper.
- ❖ **Light weight.** Fiber-optic cables are much lighter than copper cables.
- ❖ **Greater immunity to tapping.** Fiber-optic cables are more immune to tapping than copper cables. Copper cables create antenna effects that can easily be tapped.

# Guided Media

## Disadvantages of Optical Fiber

There are some disadvantages in the use of optical fiber.

- ❖ **Installation and maintenance.** Fiber-optic cable is a relatively new technology. Its installation and maintenance require expertise that is not yet available everywhere.
- ❖ **Unidirectional light propagation.** Propagation of light is unidirectional. If we need bidirectional communication, two fibers are needed.
- ❖ **Cost.** The cable and the interfaces are relatively more expensive than those of other guided media. If the demand for bandwidth is not high, then the use of optical fiber cannot be justified.

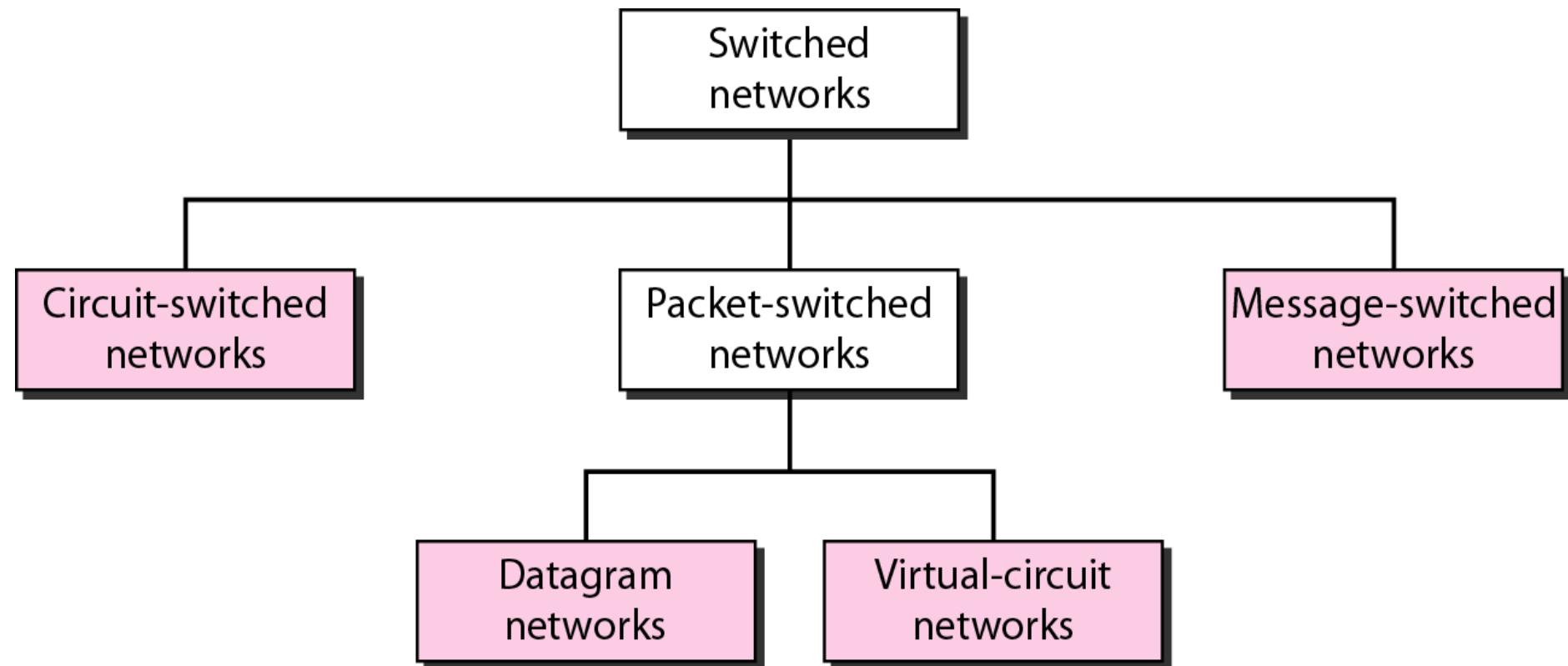
# Switching

- The technique of transferring the information from one computer network to another network is known as switching.
- Switching in a computer network is achieved by using **switches**. A **switch** is a small hardware device which is used to join multiple computers together with one local area network (LAN).
- Network switches operate at layer 2 (Data link layer) in the OSI model.
- In large networks, there can be multiple paths from sender to receiver. The switching technique will decide the best route for data transmission.
- Switching technique is used to connect the systems for making one-to-one communication.

# Switching

Switching techniques are classified in to the following three types:-

- Circuit Switching
- Packet Switching
- Message Switching

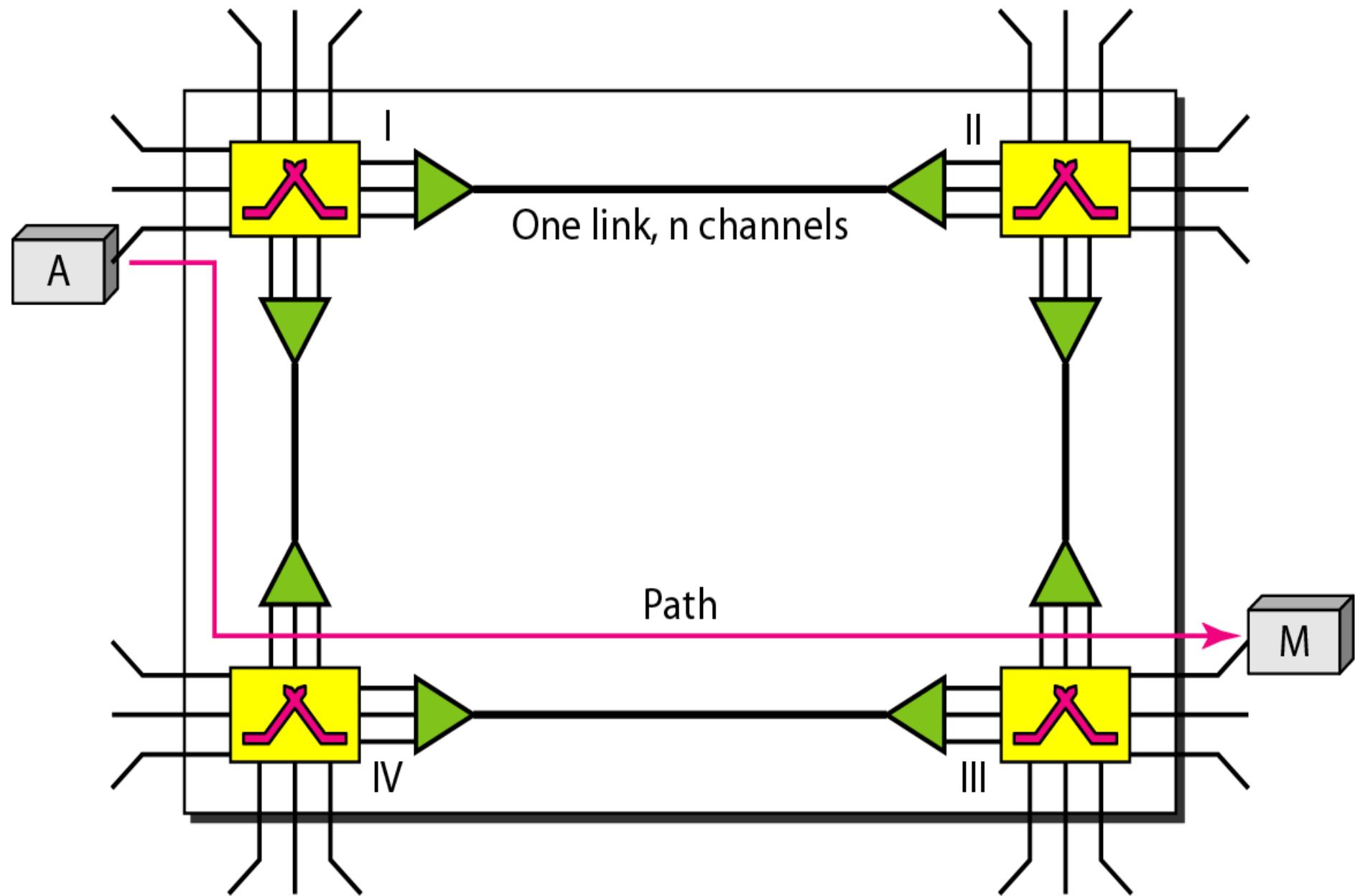


# Switching

## Circuit Switching

- Circuit switching is a switching technique that establishes a dedicated path between sender and receiver.
- In the Circuit Switching Technique, once the connection is established then the dedicated path will remain to exist until the connection is terminated.
- Circuit switching in a network operates in a similar way as the telephone works.
- A complete end-to-end path must exist before the communication takes place.
- In case of circuit switching technique, when any user wants to send the data, voice, video, a request signal is sent to the receiver then the receiver sends back the acknowledgment to ensure the availability of the dedicated path. After receiving the acknowledgment, dedicated path transfers the data.
- Circuit switching is used in public telephone network. It is used for voice transmission.
- Fixed data can be transferred at a time in circuit switching technology.

# Switching



# Switching

The actual communication in a circuit-switched network requires three phases: connection setup, data transfer, and connection teardown.

## Setup Phase

Before the two parties (or multiple parties in a conference call) can communicate, a dedicated circuit (combination of channels in links) needs to be established.

Connection setup means creating dedicated channels between the switches.

For example, in Figure, when system A needs to connect to system M, it sends a setup request that includes the address of system M, to switch I. Switch I finds a channel between itself and switch IV that can be dedicated for this purpose. Switch I then sends the request to switch IV, which finds a dedicated channel between itself and switch III. Switch III informs system M of system A's intention at this time.

In the next step to making a connection, an acknowledgment from system M needs to be sent in the opposite direction to system A. Only after system A receives this acknowledgment is the connection established.

# Switching

## Data Transfer Phase

After the establishment of the dedicated circuit (channels), the two parties can transfer data.

## Teardown Phase

When one of the parties needs to disconnect, a signal is sent to each switch to release the resources.

## Efficiency

Circuit-switched networks are not as efficient as the other two types of networks because resources are allocated during the entire duration of the connection. These resources are unavailable to other connections.

## Delay

The delay in this type of network is minimal. During data transfer the data are not delayed at each switch; the resources are allocated for the duration of the connection. The total delay is due to the time needed to create the connection, transfer data, and disconnect the circuit.

**Note:** Switching at the physical layer in the traditional telephone network uses the circuit-switching approach.

# Switching

## Packet Switching

The packet switching is a switching technique in which the message is divided into smaller pieces, and they are sent individually. The message splits into smaller pieces known as packets and packets are given a unique number to identify their order at the receiving end.

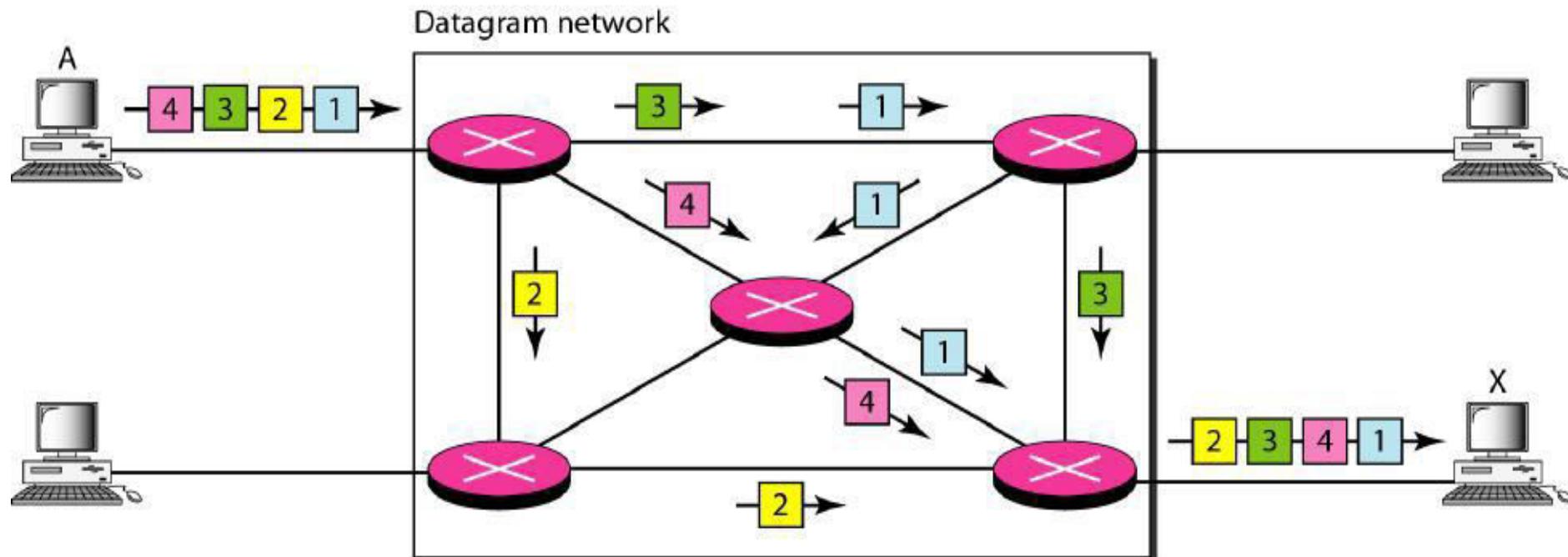
There are two types of packet switching.

1. Datagram Switching
2. Virtual Circuit Switching

# Switching

## Datagram Switching

- In a datagram network, each packet is treated independently of all others.
- Packets in this approach are referred to as datagrams.
- Datagram switching is normally done at the network layer.
- Figure shows how the datagram approach is used to deliver four packets from station A to station X. The switches in a datagram network are traditionally referred to as routers.



# Switching

A switch in a datagram network uses a routing table that is based on the destination address. The destination address in the header of a packet in a datagram network remains the same during the entire journey of the packet.

## Efficiency

The efficiency of a datagram network is better than that of a circuit-switched network; resources are allocated only when there are packets to be transferred.

## Delay

There may be greater delay in a datagram network than in a virtual-circuit network. Although there are no setup and teardown phases, each packet may experience a wait at a switch before it is forwarded. In addition, since not all packets in a message necessarily travel through the same switches, the delay is not uniform for the packets of a message.

**Note:** Switching in the Internet is done by using the datagram packet switching at the network layer.

# Switching

## Virtual Circuit Switching

A virtual-circuit network is a cross between a circuit-switched network and a datagram network. It has some characteristics of both.

1. As in a circuit-switched network, there are setup and teardown phases in addition to the data transfer phase.
2. Resources can be allocated during the setup phase, as in a circuit-switched network, or on demand, as in a datagram network.
3. As in a datagram network, data are packetized and each packet carries an address in the header.
4. As in a circuit-switched network, all packets follow the same path established during the connection.
5. A virtual-circuit network is normally implemented in the data link layer, while a circuit-switched network is implemented in the physical layer and a datagram network in the network layer.

# Switching

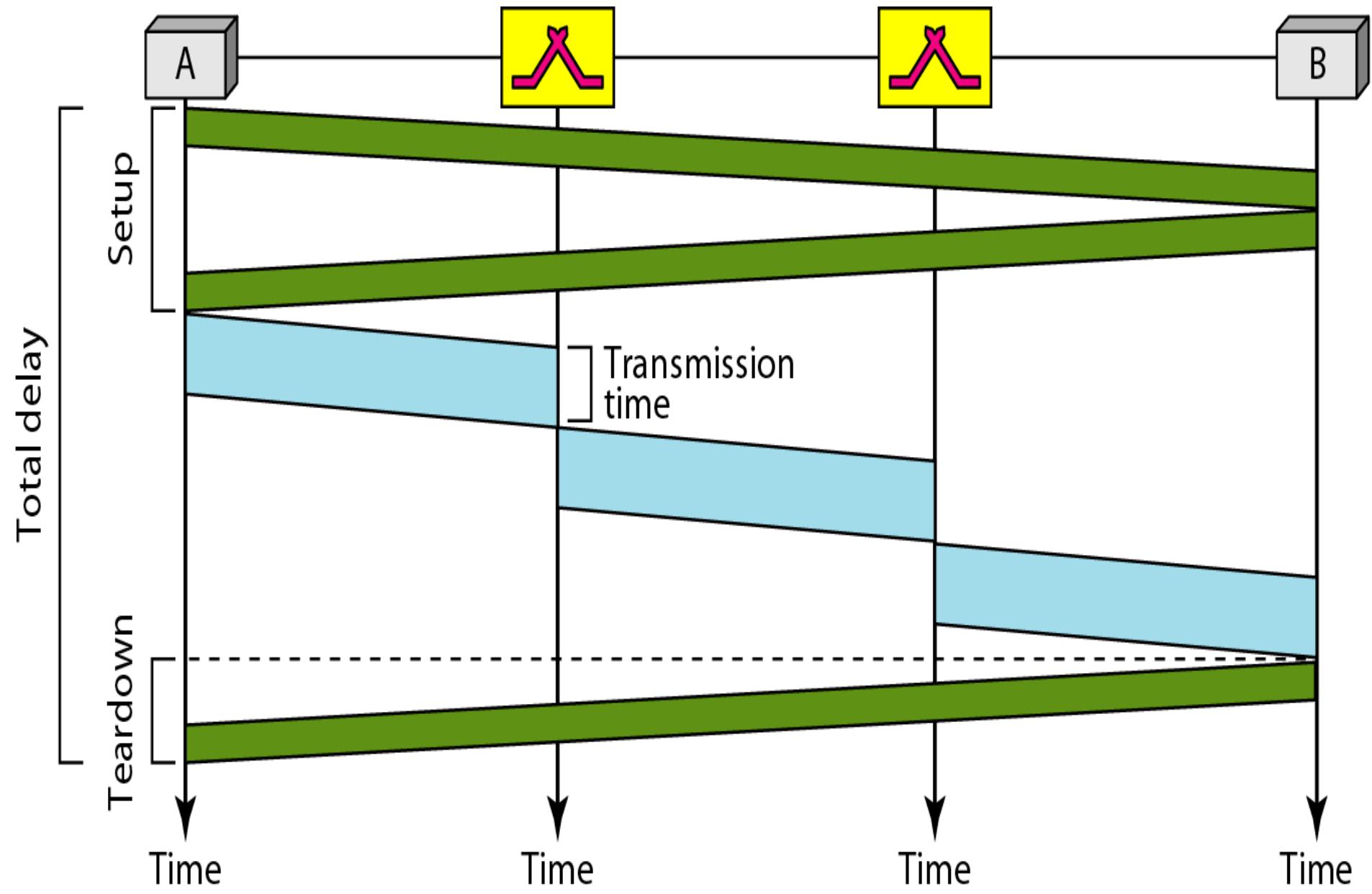
## Efficiency

In virtual-circuit switching, all packets belonging to the same source and destination travel the same path; but the packets may arrive at the destination with different delays if resource allocation is on demand.

## Delay

In a virtual-circuit network, there is a one-time delay for setup and a one-time delay for teardown. If resources are allocated during the setup phase, there is no wait time for individual packets. Figure shows the delay for a packet traveling through two switches in a virtual-circuit network.

# Switching

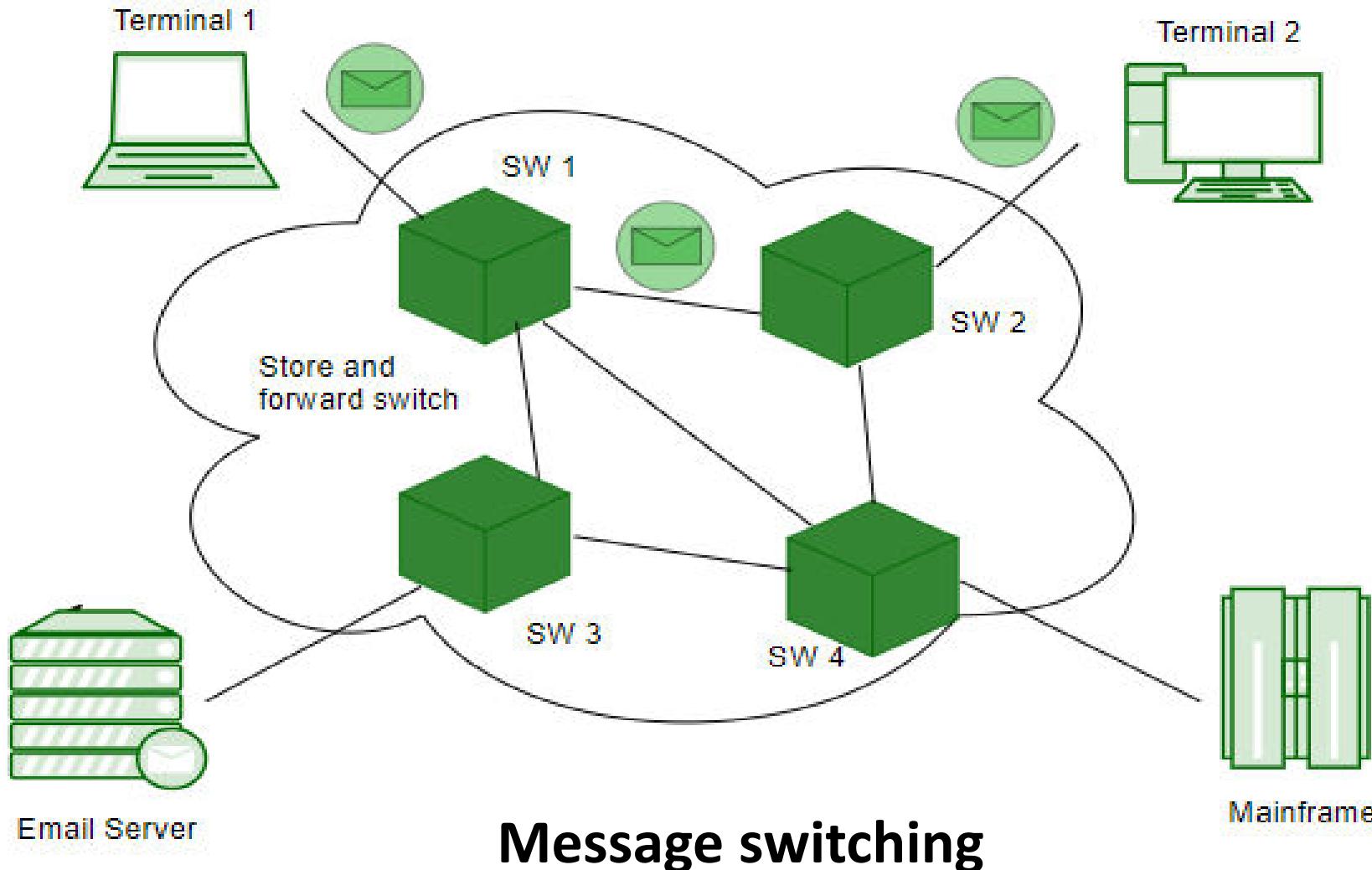


# Switching

## Message Switching

- Message Switching is a switching technique in which a message is transferred as a complete unit and routed through intermediate nodes at which it is stored and forwarded.
- In Message Switching technique, there is no establishment of a dedicated path between the sender and receiver.
- The destination address is appended to the message. Message Switching provides a dynamic routing as the message is routed through the intermediate nodes based on the information available in the message.
- Message switches are programmed in such a way so that they can provide the most efficient routes.
- Each and every node stores the entire message and then forward it to the next node. This type of network is known as **store and forward network**.
- Message switching treats each message as an independent entity.

# Switching



# Switching

## Advantages

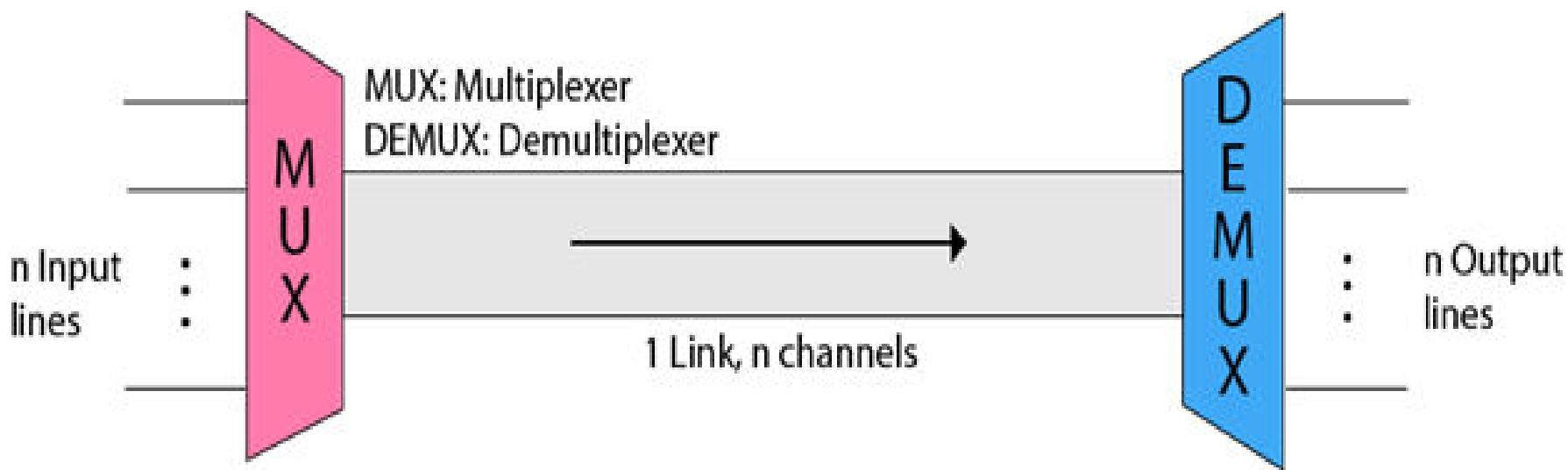
- Sharing of communication channels ensures better bandwidth usage.
- It reduces network congestion due to store and forward method. Any switching node can store the messages till the network is available.
- Broadcasting messages requires much less bandwidth than circuit switching.
- Messages of unlimited sizes can be sent.
- It does not have to deal with out of order packets or lost packets as in packet switching.

## Disadvantages

- In order to store many messages of unlimited sizes, each intermediate switching node requires large storage capacity.
- Store and forward method introduces delay at each switching node. This renders it unsuitable for real time applications.

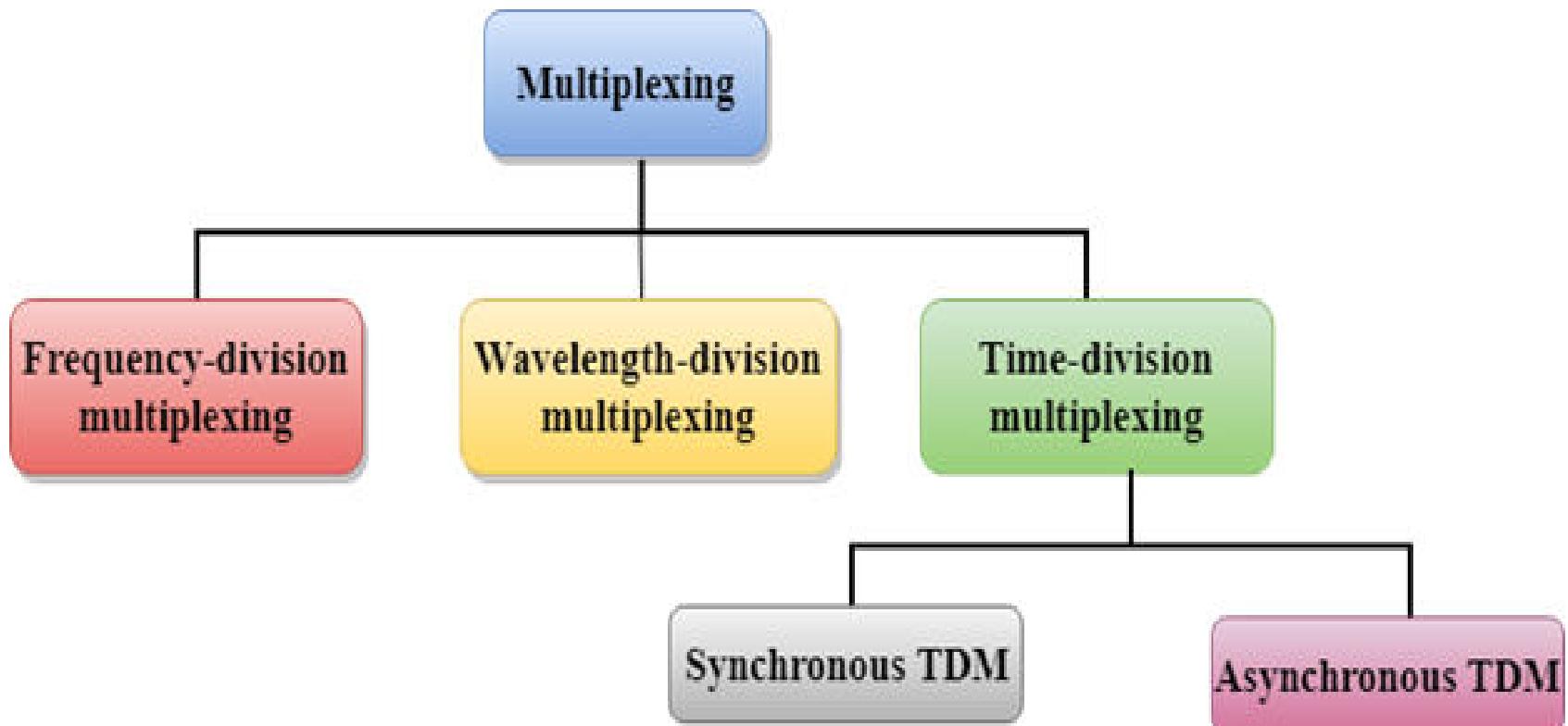
# Multiplexing

- Multiplexing is a technique used to combine and send the multiple data streams over a single medium.
- Multiplexing is achieved by using a device called Multiplexer (MUX) that combines  $n$  input lines to generate a single output line. Multiplexing follows many-to-one, i.e.,  $n$  input lines and one output line.
- Demultiplexing is achieved by using a device called Demultiplexer (DEMUX) available at the receiving end. DEMUX separates a signal into its component signals (one input and  $n$  outputs). Therefore, we can say that demultiplexing follows the one-to-many approach.



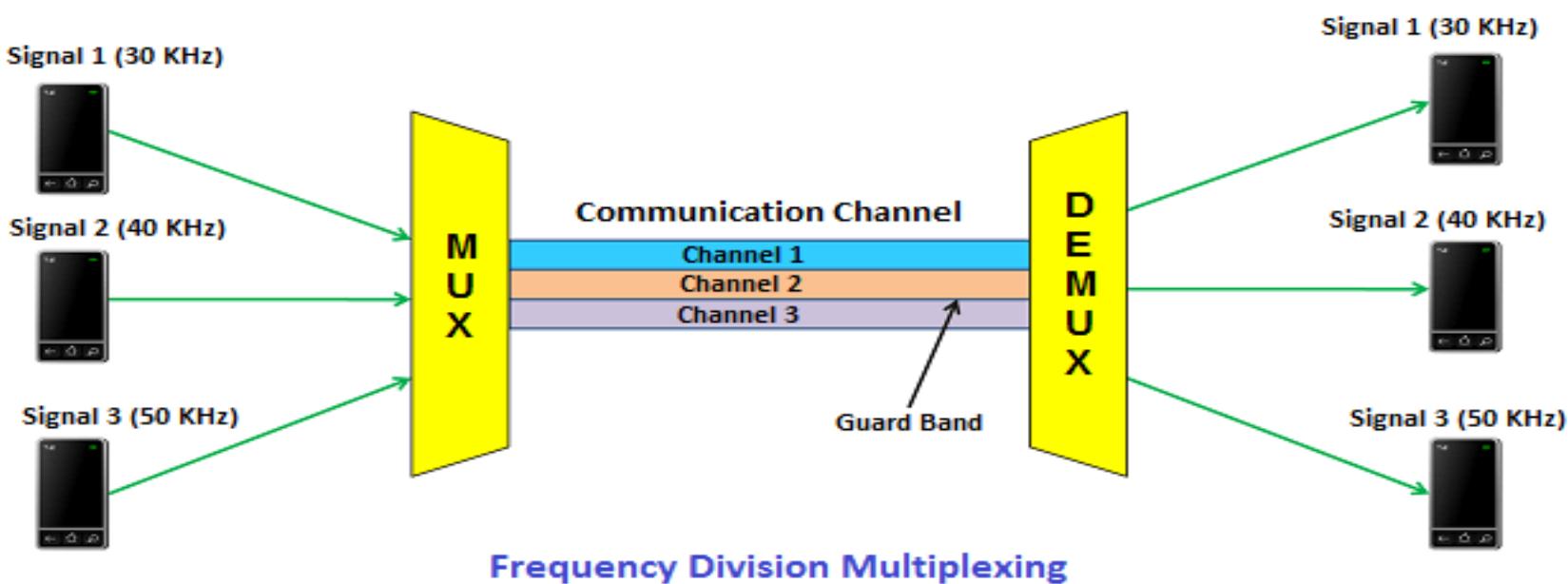
# Multiplexing

There are three basic multiplexing techniques: frequency-division multiplexing, wavelength-division multiplexing, and time-division multiplexing. The first two are techniques designed for **analog signals**, the third, for **digital signals**.



# Frequency-division Multiplexing (FDM)

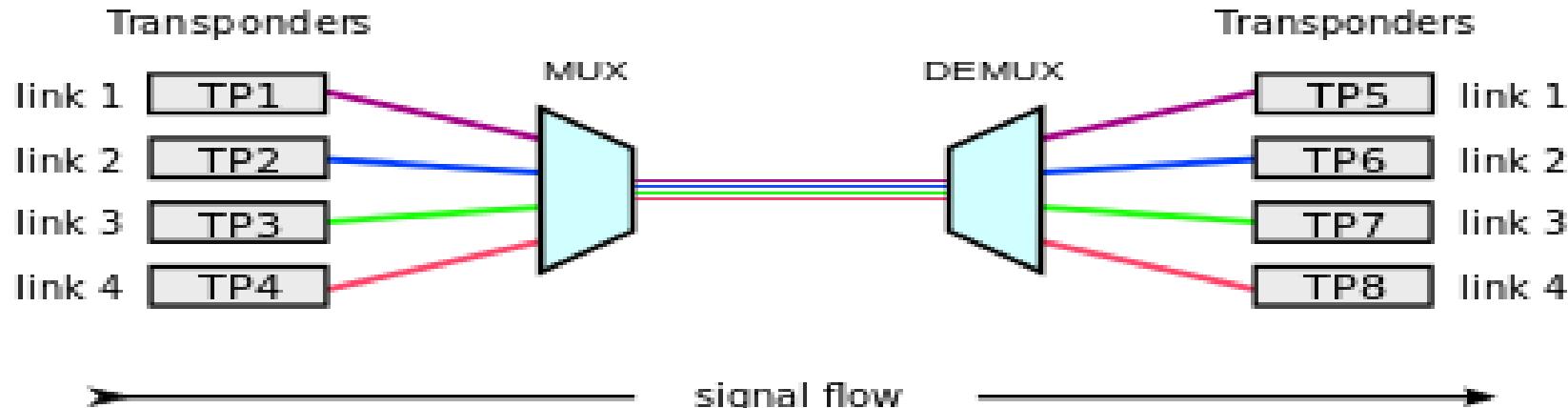
- Frequency Division Multiplexing is a technique in which the available bandwidth of a single transmission medium is subdivided into several channels.
- Frequency-division multiplexing (FDM) is an analog technique that can be applied when the bandwidth of a link (in hertz) is greater than the combined bandwidths of the signals to be transmitted.
- The main aim of the FDM is to subdivide the available bandwidth into different frequency channels and allocate them to different devices.
- FDM is mainly used in radio broadcasts and TV networks.



# Wavelength Division Multiplexing (WDM)

- Wavelength Division Multiplexing is same as FDM except that the optical signals are transmitted through the fiber optic cable.
- WDM is used on fiber optics to increase the capacity of a single fiber.
- It is used to utilize the high data rate capability of fiber optic cable.
- It is an analog multiplexing technique.
- Multiplexing and Demultiplexing can be achieved by using a prism.
- One application of WDM is the SONET network in which multiple optical fiber lines are multiplexed and demultiplexed.

## wavelength-division multiplexing (WDM)



# Time Division Multiplexing (TDM)

- It is a digital technique.
- In Frequency Division Multiplexing Technique, all signals operate at the same time with different frequency, but in case of Time Division Multiplexing technique, all signals operate at the same frequency with different time.
- In Time Division Multiplexing technique, the total time available in the channel is distributed among different users. Therefore, each user is allocated with different time interval known as a Time slot at which data is to be transmitted by the sender.
- A user takes control of the channel for a fixed amount of time.
- In Time Division Multiplexing technique, data is not transmitted simultaneously rather the data is transmitted one-by-one.
- In TDM, the signal is transmitted in the form of frames. Frames contain a cycle of time slots in which each frame contains one or more slots dedicated to each user.

# Time Division Multiplexing (TDM)

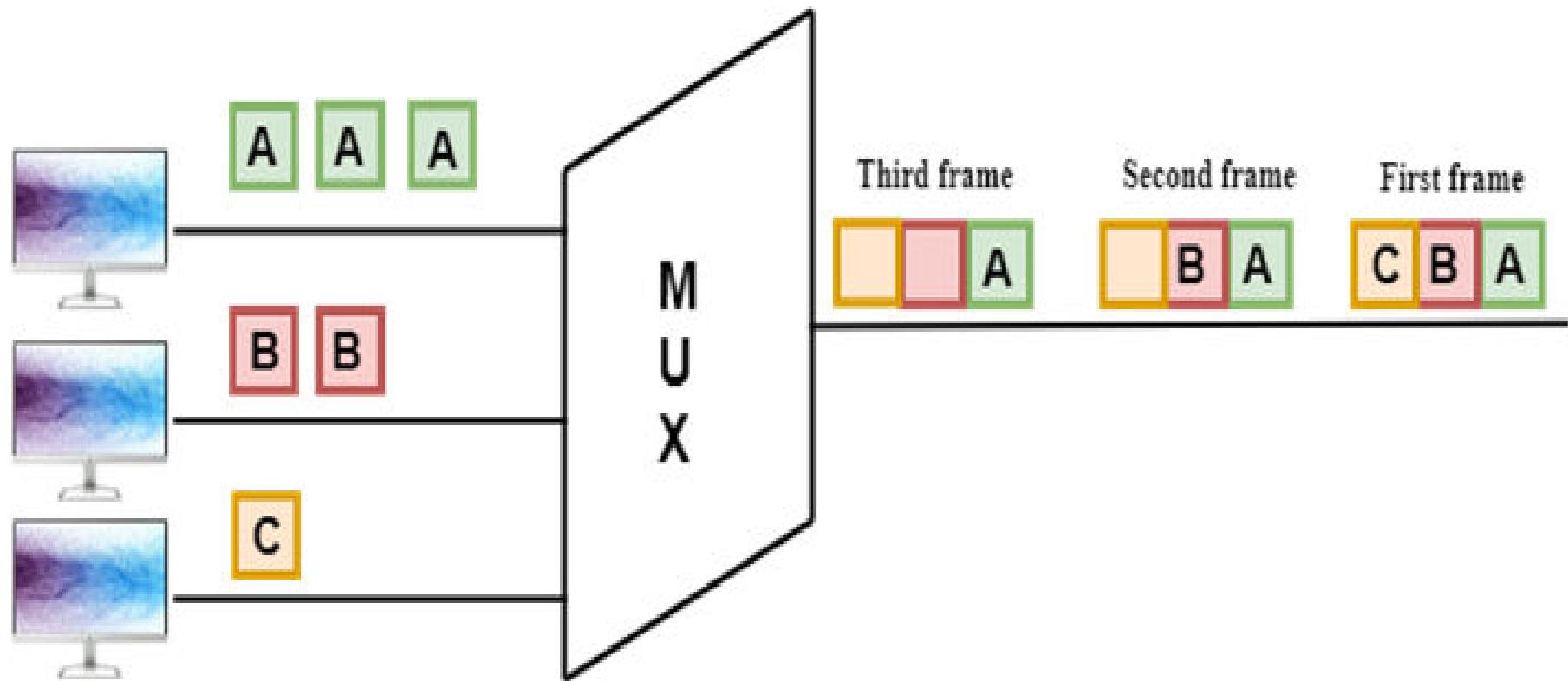
There are two types of TDM:

1. Synchronous TDM
2. Asynchronous TDM

## Synchronous TDM

- A Synchronous TDM is a technique in which time slot is pre-assigned to every device.
- In Synchronous TDM, each device is given some time slot irrespective of the fact that the device contains the data or not.
- If the device does not have any data, then the slot will remain empty.
- In Synchronous TDM, signals are sent in the form of frames. Time slots are organized in the form of frames. If a device does not have data for a particular time slot, then the empty slot will be transmitted.
- The most popular Synchronous TDM are T-1 multiplexing, ISDN multiplexing, and SONET multiplexing.
- If there are n devices, then there are n slots.

# Time Division Multiplexing (TDM)



# Multiplexing

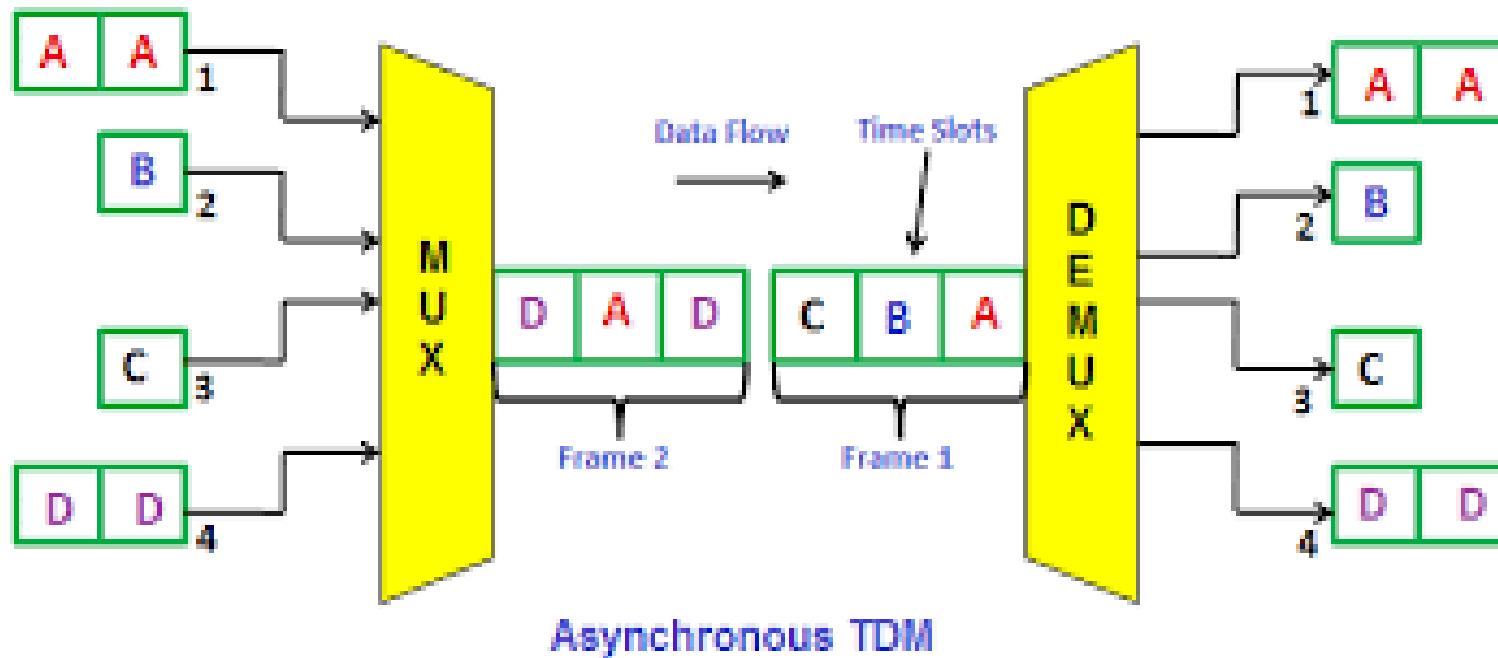
## Asynchronous TDM

- An asynchronous TDM is also known as Statistical TDM.
- An asynchronous TDM is a technique in which time slots are not fixed as in the case of Synchronous TDM. Time slots are allocated to only those devices which have the data to send. Therefore, we can say that Asynchronous Time Division multiplexor transmits only the data from active workstations.
- An asynchronous TDM technique dynamically allocates the time slots to the devices.
- In Asynchronous TDM, total speed of the input lines can be greater than the capacity of the channel.
- In Asynchronous TDM, each slot contains an address part that identifies the source of the data.



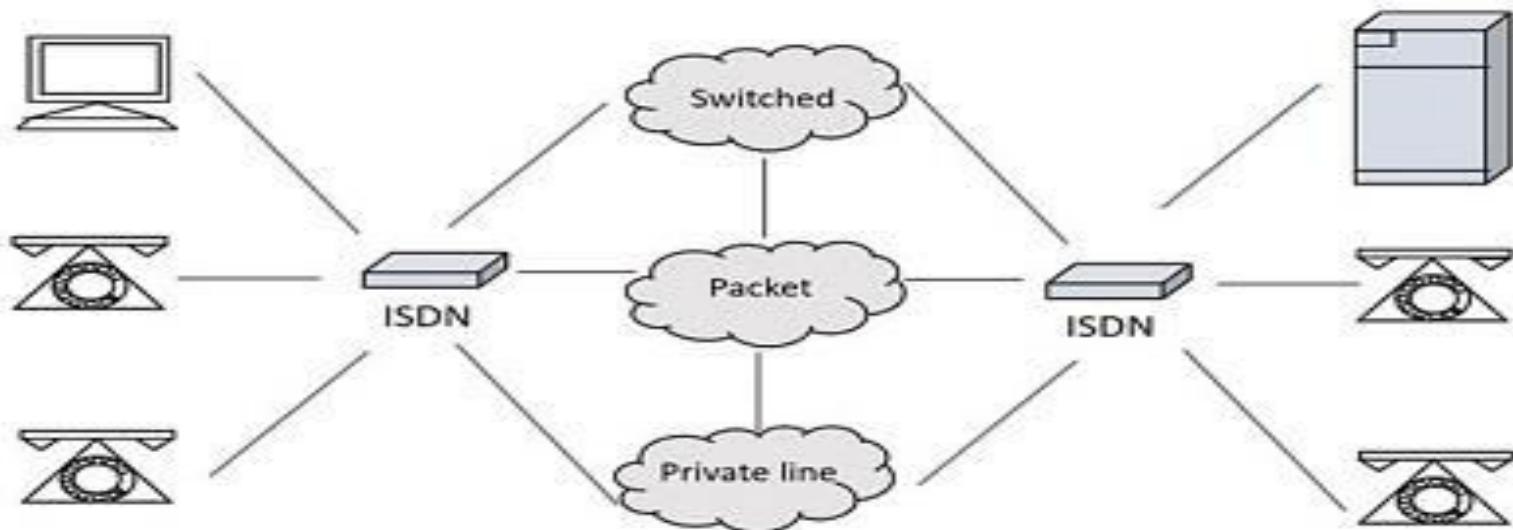
# Multiplexing

- In Synchronous TDM, if there are n sending devices, then there are n time slots. In Asynchronous TDM, if there are n sending devices, then there are m time slots where m is less than n ( $m < n$ ).
- The number of slots in a frame depends on the statistical analysis of the number of input lines.



# ISDN

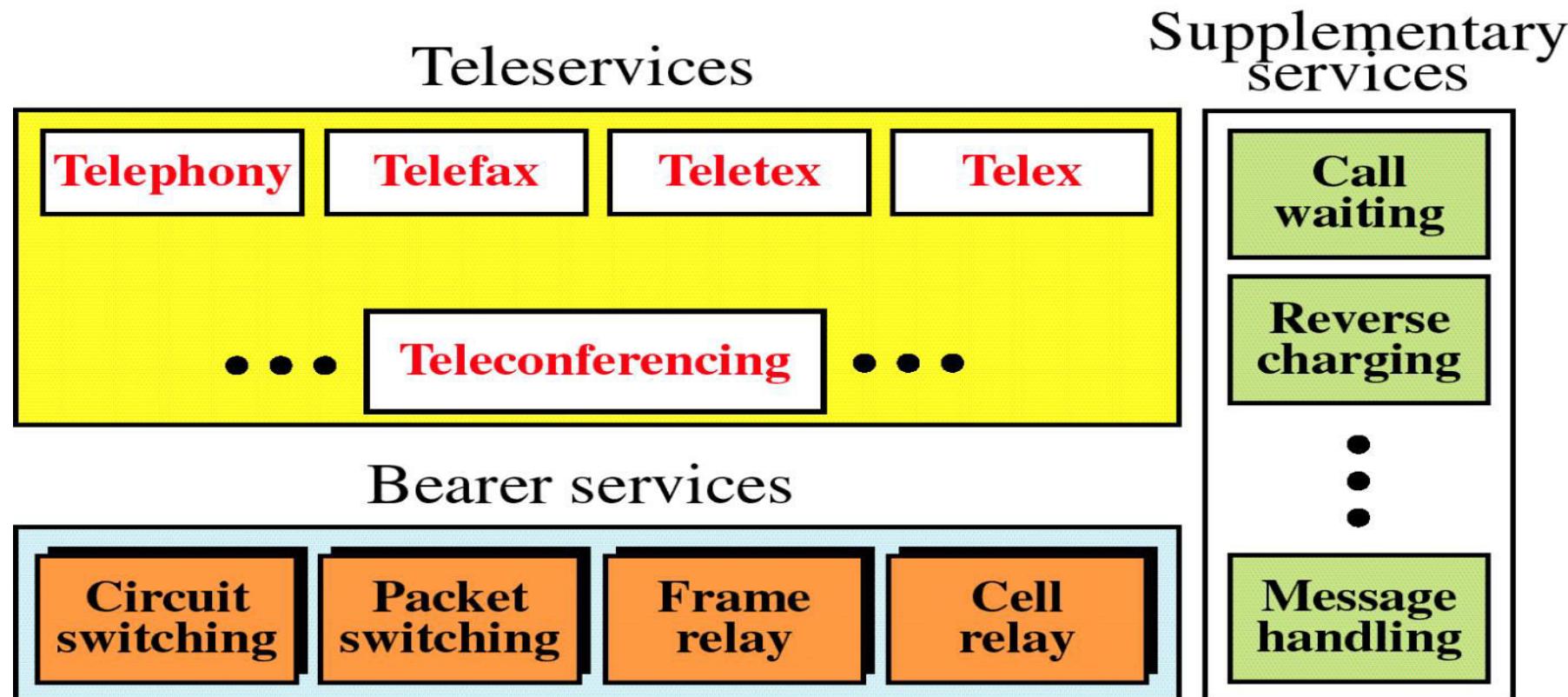
- Its full form is Integrated Services Digital Network.
- ISDN is a telephone network based infrastructure that allows the transmission of voice and data simultaneously at a high speed with greater efficiency.
- This is a circuit switched telephone network system, which also provides access to Packet switched networks.
- The model of a practical ISDN is as shown below:-



# ISDN

## ISDN Services

The purpose of the ISDN is to provide fully integrated digital services to users. These services fall into three categories: bearer services, teleservices and supplementary services.



# ISDN

## Bearer Services

These services provide the means to transfer information between users without the network manipulating the content of that information. The network does not need to process the information and therefore does not change the content. Bearer services belong to the first three layers of the OSI model. They can be provided using circuit-switched, packet-switched, frame-switched or cell switched networks.

## Teleservices

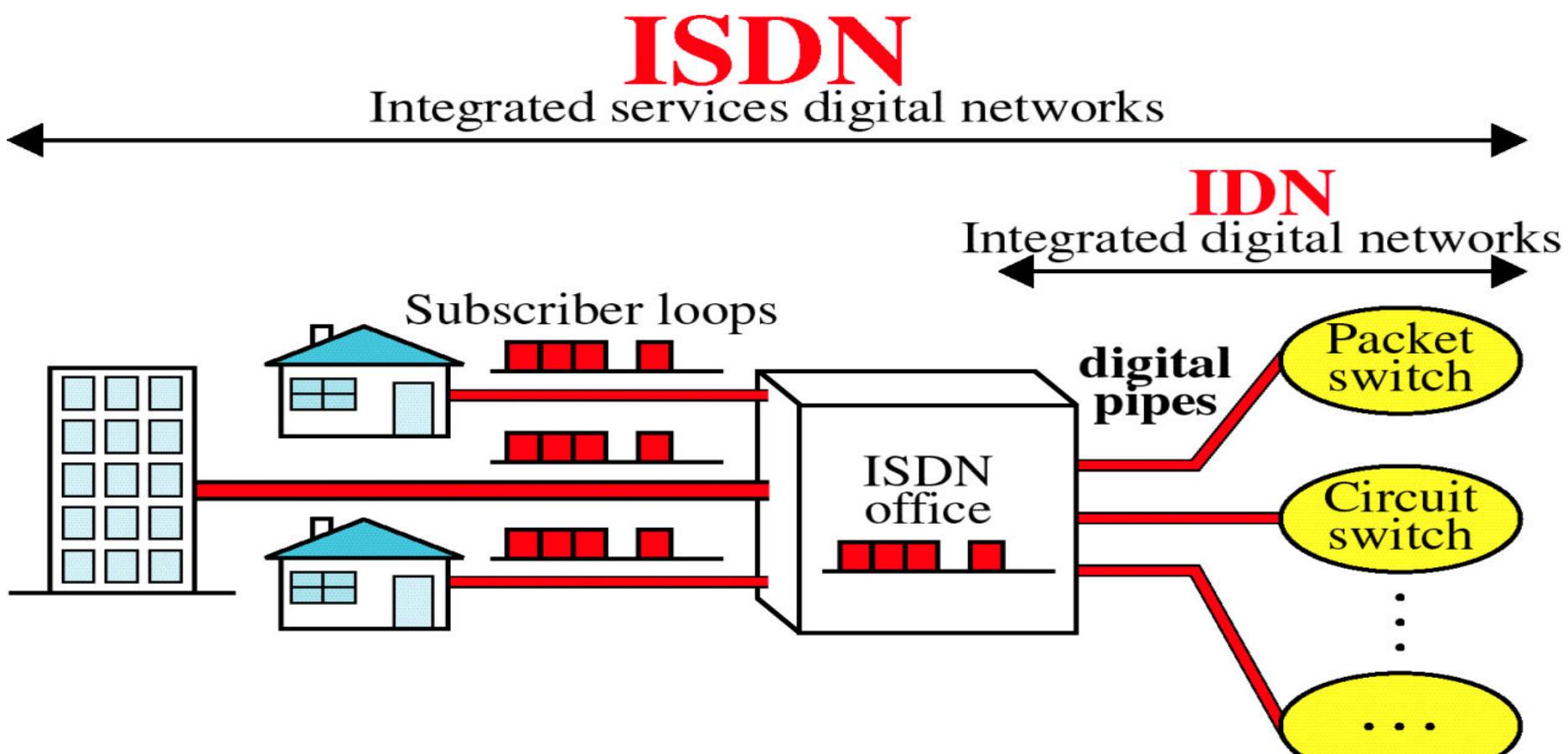
In teleservicing, the network may change or process the contents of the data. These services correspond to layers 4-7 of the OSI model. Teleservices include telephony, teletex, telefax, videotex, telex and teleconferencing.

## Supplementary Services

Supplementary services are those services that provide additional functionality to the bearer services and teleservices. These services are reverse charging, call waiting and message handling.

# ISDN

Following figure gives a conceptual view of the connection between users and an ISDN control office.



# ISDN

- Each user is linked to the central office through a digital pipe.
- Digital pipes between user and ISDN office are organized into multiple channels of different sizes. ISDN standard defines three channel types, each with different transmission rate: bearer channel, data channel and hybrid channels.

**B-Channel (Bearer Channel):** Bearer channel is defined at a rate of 64 kbps. It is the basic user channel and can carry any type of digital information in full duplex mode as long as the required information rate does not exceed 64 kbps.

**D channel (Data Channel):** Data channel can be either 16 or 64 kbps, depending on the needs of user. The primary function of D channel is to carry control signal for the B channel.

**H channel (Hybrid Channel):** Hybrid channels are available with data rates of 384 Kbps, 1536 Kbps or 1920 Kbps. These rates suit H channels for high data rate applications such as video, teleconferencing and so on.

# ISDN

## User Interfaces

Digital subscriber loops are of two types:

- (1) Basic rate interface (BRI)
- (2) Primary rate interface (PRI)

Each type is suited to a different level of customer needs. Both include one D channel and some number of either B or H channels.

# ISDN

## Basic rate interface

- The basic rate interface specifies a digital pipe consisting of two B channels and one 16 Kbps D channel.
- Two B channel of 64 Kbps each, plus one D channel of 16 Kbps, equal 144 Kbps. In addition, the BRI service itself requires 48 Kbps of operating overhead. Therefore, BRI requires a digital pipe of 192 Kbps.
- BRI is designed to meet the needs of residential and small-office customers.

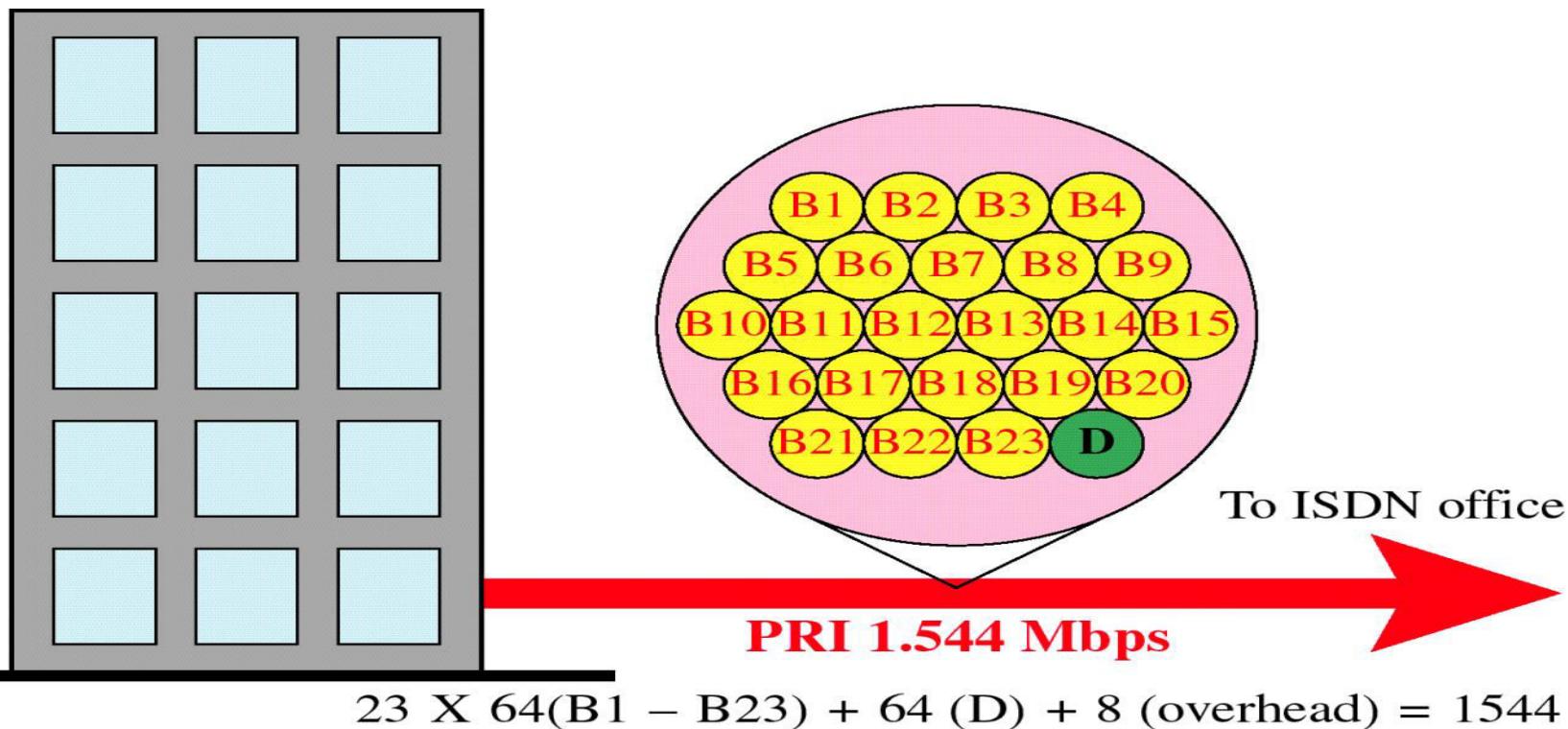


$$\begin{aligned} & 64(\text{B1}) + 64(\text{B2}) + 16(\text{D}) + 48 \text{ (overhead)} \\ & = 192 \text{ Kbps} \end{aligned}$$

# ISDN

## Primary rate interface

- The primary rate interface specifies a digital pipe with 23 B channels and one 64 Kbps D channel.
- Twenty three B channels of 64 Kbps each plus one D channel of 64 Kbps equals 1.536 Mbps. In addition, the PRI service itself uses 8 Kbps of overhead. Therefore, PRI requires a digital pipe of 1.544 Mbps.



# ISDN

## Broadband ISDN

- When ISDN was originally designed, data rates of 64 Kbps to 1.544 Mbps were sufficient to handle all existing transmission needs. But after sometimes, this rate is insufficient.
- To provide for the needs of next generation of technology, B-ISDN has been developed. The original ISDN is now known as narrow ISDN(N-ISDN). B-ISDN provides subscribers to the network with data rates in the range of 600 Mbps.

# Some questions

## Question-1:

A path in a digital circuit-switched network has a data rate of 1 Mbps. The exchange of 1000 bits is required for the setup and teardown phases. The distance between two parties is 5000 km. Answer the following questions if the propagation speed is  $2 \times 10^8$  m/s:

- a. What is the total delay if 1000 bits of data are exchanged during the data transfer phase?
- b. What is the total delay if 100,000 bits of data are exchanged during the data transfer phase?
- c. What is the total delay if 1,000,000 bits of data are exchanged during the data transfer phase?
- d. Find the delay per 1000 bits of data for each of the above cases and compare them. What can you infer?

# Some questions

**Question-1:** Five equal-size datagrams belonging to the same message leave for the destination one after another. However, they travel through different paths as shown in Table.

<i>Datagram</i>	<i>Path Length</i>	<i>Visited Switches</i>
1	3200Km	1,3,5
2	11,700 Km	1,2,5
3	12,200 Km	1,2,3,5
4	10,200 Km	1,4,5
5	10,700 Km	1,4,3,5

We assume that the delay for each switch (including waiting and processing) is 3, 10, 20, 7, and 20 ms respectively. Assuming that the propagation speed is  $2 \times 10^8$  m/s, find the order the datagrams arrive at the destination and the delay for each. Ignore any other delays in transmission.

# AKTU Examination Questions

1. What are header and trailers and how do they get added and removed?
2. What is the difference between network layer delivery and the transport layer delivery?
3. Define topology and explain the advantage and disadvantage of Bus, Star and Ring topologies.
4. What is OSI Model? Explain the functions, protocols and services of each layer?
5. Consider a noiseless channel with a bandwidth of 3000 Hz transmitting a signal with four signal levels. What is the maximum bit rate?

# AKTU Examination Questions

6. Encode the data-stream 10011010 using the following encoding scheme:
- i. Unipolar
  - ii. Bipolar NRZ-L
  - iii. Bipolar NRZ-I
  - iv. RZ
  - v. Manchester
  - vi. Differential Manchester
  - vii. AMI
7. Write four differences between circuit switching and packet switching.
8. Sketch Manchester and differential Manchester encoding for the following bit stream:  
10111100010010011101

# AKTU Examination Questions

9. What are the services of Transport Layer?
10. What are the major advantages of using optical fiber over twisted pair cable?
11. What do you mean by network architecture? What should be their design issues? Explain briefly.
12. Discuss different types of transmission media with their advantages and disadvantages.
13. Differentiate OSI and TCP/IP reference model. Which one is more popular and why?

# AKTU Examination Questions

14. Suppose a signal travels through a transmission medium then find:
  - i. The attenuation (loss of power) if the power is reduced to one half.
  - ii. The amplification (gain of power) if the power is Increased 10 times.
15. What do you mean by transmission impairment? Explain different types of transmission impairment.
16. What are the applications of Computer Networks?
17. List the advantages and disadvantages of ring topology.
18. If a binary signal is sent over a 3KHZ channel. Whose signal to noise ratio is 20db. What is the maximum achievable data rate?

# AKTU Examination Questions

19. Explain network topological design with necessary diagram and brief the advantages and disadvantages of various topologies.
20. Discuss the different physical layer transmission media.
21. Write about user access in ISDN.
22. List the advantages and disadvantages of star topology.
23. Explain functionalities of every layer in OSI reference model with neat block diagram.

# **Unit-2**

# Data Link Control

- The two main functions of the data link layer are **data link control and media access control**. The first, data link control, deals with the design of procedures for communication between two adjacent nodes: node-to-node communication. The second function of the data link layer is media access control, or how to share the link.
- Data link control functions include framing, flow and error control protocols that provide smooth and reliable transmission of frames between nodes.

# Data Link Layer

**Error Detection and Correction**

# Error Detection and Correction

## **Single bit error**

In single bit error, only 1 bit in the data unit has changed.

## **Burst error**

A burst error means that 2 or more bits in the data unit have changed.

## **Redundancy**

To detect or correct errors, we need to send extra bits with data.

## **Block Coding**

In block coding, we divide our message into blocks, each of  $k$  bits, called datawords. We add  $r$  redundant bits to each block to make the length  $n = k + r$ . The resulting  $n$ -bit blocks are called codewords.

# Error Detection and Correction

## Hamming Distance

- ❖ The Hamming distance between two words (of the same size) is the number of differences between the corresponding bits.
- ❖ The Hamming distance can easily be found if we apply the XOR operation on the two words and count the number of 1's in the result.

### Example

Find the Hamming distance between the following words:-

a = 10101110 and b = 01010100

### Minimum Hamming Distance

The minimum Hamming distance is the smallest Hamming distance between all possible pairs. We use  $d_{\min}$  to define the minimum Hamming distance in a coding scheme.

### Example

Find the minimum Hamming distance for the following set of words:-  
{ 00000, 10101, 01011, 11110 }.

# Error Detection and Correction

## Minimum Hamming Distance for Error Detection

To guarantee the detection of up to  $s$  errors in all cases, the minimum Hamming distance in a block code must be

$$d_{\min} = s + 1.$$

## Minimum Hamming Distance for Error Correction

To guarantee the correction of up to  $t$  errors in all cases, the minimum Hamming distance in a block code must be

$$d_{\min} = 2t + 1.$$

# Error Detection and Correction

## Simple Parity-Check Code

- ❖ In this code, a  $k$ -bit dataword is changed to an  $k+1$ -bit codeword.
- ❖ The extra bit, called the parity bit.
- ❖ It is selected to make the total number of 1's in the codeword even.

### Note:

A simple parity-check code is a single-bit error-detecting code in which  $n = k + 1$  with  $d_{\min} = 2$ .

# Error Detection and Correction

## Hamming Code

Hamming code is a set of error-correction codes that can be used to **detect and correct the errors** that can occur when the data is moved or stored from the sender to the receiver.

It is **technique developed by R.W. Hamming for error correction.**

### **Redundant bits –**

The number of redundant bits can be calculated using the following formula:

$$2^r \geq m+r+1$$

Where, r = number of redundant bits, and  
m = number of data bits

# Error Detection and Correction

## Algorithm of Hamming code

1. Write the bit positions starting from 1.
2. All the bit positions that are a power of 2 are marked as parity bits (1, 2, 4, 8, etc).
3. All the other bit positions are marked as data bits.
4. Each data bit is included in a unique set of parity bits, as determined its bit position in binary form.
  - Parity bit 1 covers all the bits positions whose binary representation includes a 1 in the least significant position (1, 3, 5, 7, 9, 11, etc).
  - Parity bit 2 covers all the bits positions whose binary representation includes a 1 in the second position from the least significant bit (2, 3, 6, 7, 10, 11, etc).
  - Parity bit 4 covers all the bits positions whose binary representation includes a 1 in the third position from the least significant bit (4–7, 12–15, 20–23, etc).
  - Parity bit 8 covers all the bits positions whose binary representation includes a 1 in the fourth position from the least significant bit bits (8–15, 24–31, 40–47, etc).

# Error Detection and Correction

- In general, each parity bit covers all bits where the bitwise AND of the parity position and the bit position is non-zero.
5. Set a parity bit to 1 if the total number of ones in the positions it checks is odd.
  6. Set a parity bit to 0 if the total number of ones in the positions it checks is even.

# Error Detection and Correction

**Ex.** Construct an even parity Hamming code word for a data word 1011001.

**Solution:**

**Step-1:** First we compute the number of redundant bits r.

Here, number of bits in the given data word (1011001), m = 7

Therefore, we compute r as following:-

$$2^r \geq m+r+1 \rightarrow 2^r \geq 7+r+1$$

Minimum value of r which satisfies above inequality = 4.

Therefore, r=4.

**Step-2:** Now, we compute the position of redundant bits in the codeword.

These redundancy bits are placed at positions that correspond to the power of 2. Therefore, the position these redundant bits will be 1, 2, 4 and 8.



# Error Detection and Correction

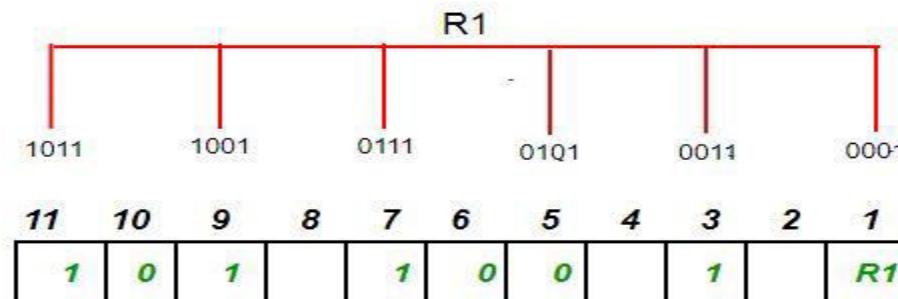
**Step-3:** Now, we compute the codeword.

Since the data to be transmitted is 1011001, therefore the bits will be placed as follows:

11	10	9	8	7	6	5	4	3	2	1
1	0	1	R8	1	0	0	R4	1	R2	R1

**Determining the Parity bits:**

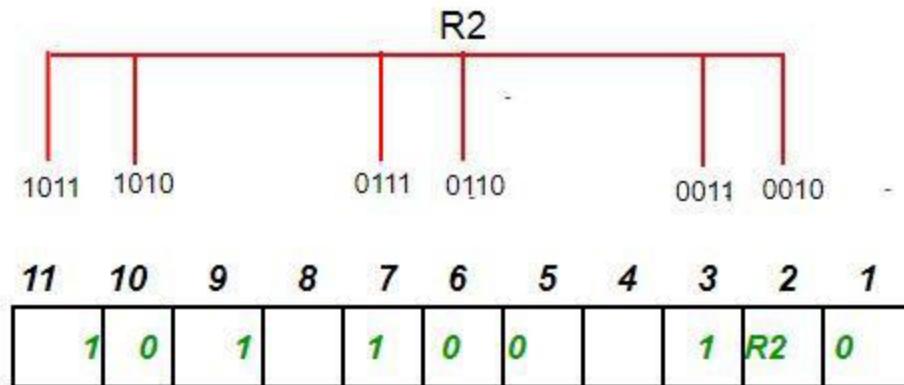
R1 bit is calculated using parity check at all the bits positions whose binary representation includes a 1 in the least significant position i.e. positions 1, 3, 5, 7, 9, 11.



To find the redundant bit R1, we check for even parity. Since the total number of 1's in all the bit positions corresponding to R1 is an even number the value of R1 (parity bit's value) = 0

# Error Detection and Correction

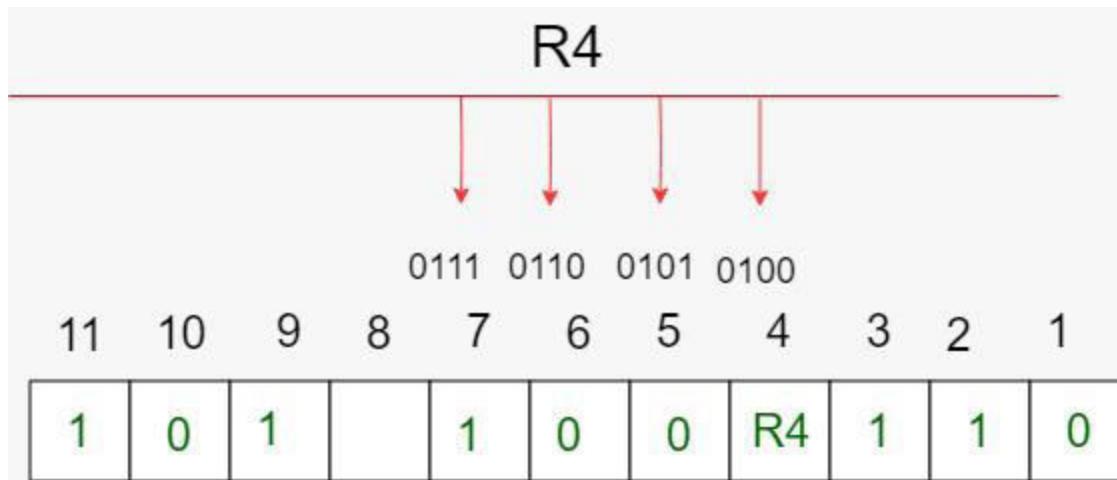
**R2 bit is calculated** using parity check at all the bits positions whose binary representation includes a 1 in the second position from the least significant bit. R2: bits 2,3,6,7,10,11.



To find the redundant bit R2, we check for even parity. Since the total number of 1's in all the bit positions corresponding to R2 is odd the value of R2 (parity bit's value)=1

# Error Detection and Correction

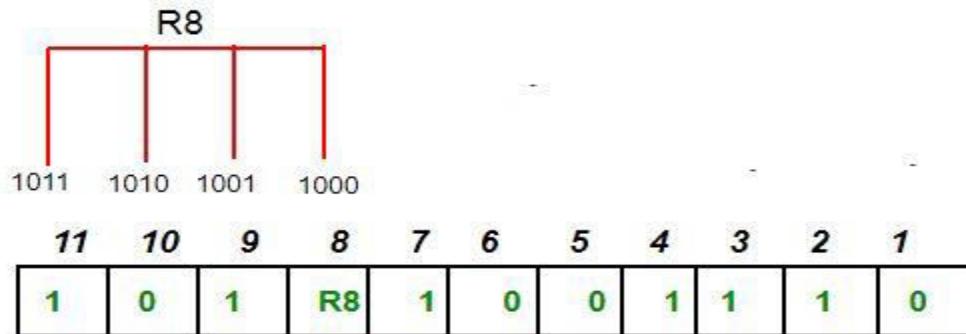
**R4 bit is calculated** using parity check at all the bits positions whose binary representation includes a 1 in the third position from the least significant bit. R4: bits 4, 5, 6, 7.



To find the redundant bit R4, we check for even parity. Since the total number of 1's in all the bit positions corresponding to R4 is odd the value of R4(parity bit's value) = 1.

# Error Detection and Correction

**R8 bit is calculated** using parity check at all the bits positions whose binary representation includes a 1 in the fourth position from the least significant bit. R8: bit 8, 9, 10, 11.

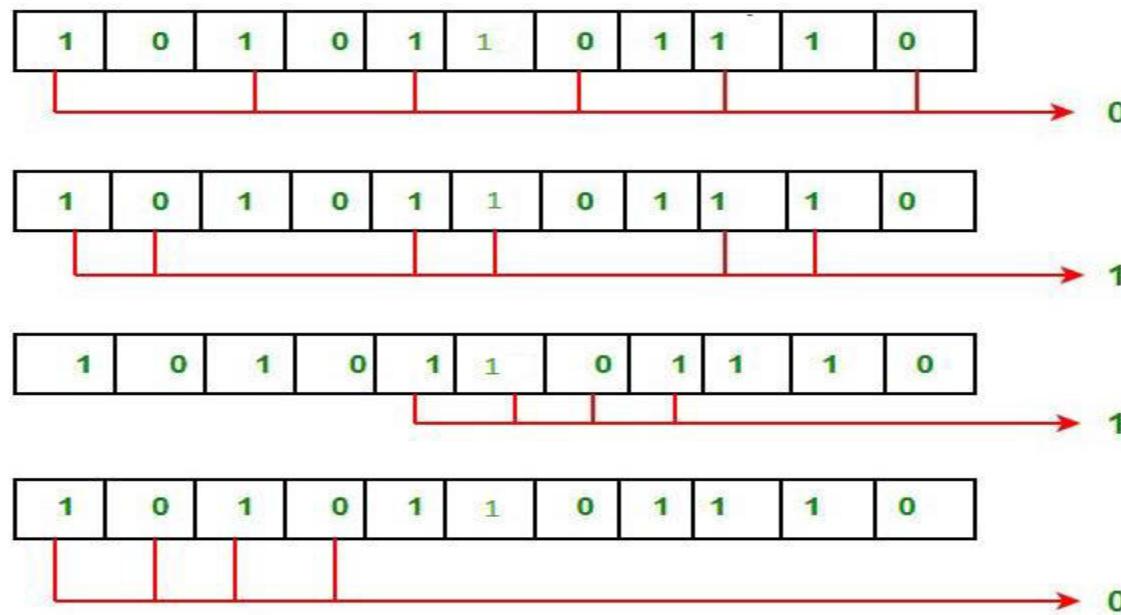


To find the redundant bit R8, we check for even parity. Since the total number of 1's in all the bit positions corresponding to R8 is an even number the value of R8(parity bit's value)= **0**. Thus, the data transferred is:

11	10	9	8	7	6	5	4	3	2	1
1	0	1	0	1	0	0	1	1	1	0

# Error Detection and Correction

**Error detection and correction:** Suppose in the above example the 6<sup>th</sup> bit is changed from 0 to 1 during data transmission, then it gives new parity values in the binary number:



The bits give the binary number **0110** whose decimal representation is **6**. Thus, bit 6 contains an error. **To correct the error** the 6<sup>th</sup> bit is changed from **1 to 0**.

# Error Detection and Correction

## Cyclic Redundancy Check (CRC)

- Suppose size of dataword is  $k$ -bits.
- This technique uses a divisor to find a codeword.
- Suppose size of divisor is  $m$ -bits.

### At sender end:

The codeword corresponding to dataword is found in the following way:-

1. First we find a word by augmenting  $m-1$  0's to the right end of the dataword.
2. Now, we divide this new word by the divisor and find a remainder.
3. Codeword is obtained by augmenting the remainder to the right end of dataword.

# Error Detection and Correction

## At receiver end:

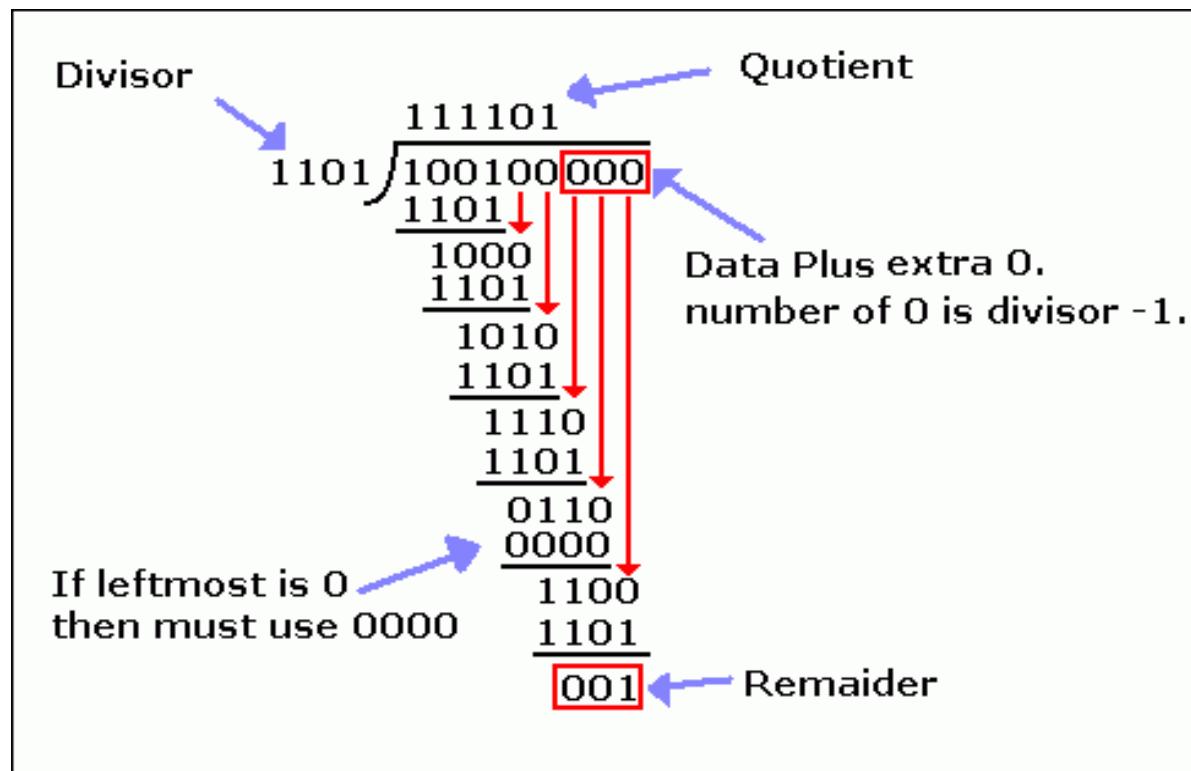
The dataword at the receiver end is found in the following way:-

1. First we divide the received codeword by divisor and find the remainder.
2. Remainder is called syndrome. If remainder is zero, then dataword will be accepted otherwise dataword will be rejected or discarded.
3. If remainder is zero, then dataword will be found by removing  $m-1$  least significant bits of received codeword.

# Error Detection and Correction

**Example:** If divisor is 1101, then find codeword corresponding to the dataword 100100.

**Solution:**



Therefore codeword = 100100001

# Error Detection and Correction

## Example:

- (1) If Codeword 100100001 is received at receiver end, then find syndrome.
- (2) If Codeword 100100101 is received at receiver end, then find syndrome.

## CRC in polynomial

- The divisor in CRC is normally called generator.
- We define the following terms:-

Dataword =  $d(x)$    Codeword =  $c(x)$    Generator =  $g(x)$

Syndrome =  $s(x)$    Error =  $e(x)$

# Error Detection and Correction

In a cyclic code,

1. If  $s(x) \neq 0$ , one or more bits is corrupted.
2. If  $s(x) = 0$ , then either
  - a. No bit is corrupted. or
  - b. Some bits are corrupted, but the decoder failed to detect them.

Received codeword =  $c(x) + e(x)$

The receiver divides the received codeword by  $g(x)$  to get the syndrome.

$$\frac{\text{Received codeword}}{g(x)} = \frac{c(x)}{g(x)} + \frac{e(x)}{g(x)}$$

$\frac{c(x)}{g(x)}$  does not have a remainder. So the syndrome is the remainder of  $\frac{e(x)}{g(x)}$ .

In a cyclic code, those  $e(x)$  errors that are divisible by  $g(x)$  are not caught.

# Error Detection and Correction

**Example:**

Let dataword  $d(x) = x^3 + 1$ , generator  $g(x) = x^3 + x + 1$ .

Find codeword.

**Solution:**

Augmented dataword  $= x^6 + x^3$

$$x^3 + x + 1 \quad | \quad x^6 + x^3 \quad (x^3 + x)$$

$$\underline{x^6 + x^4 + x^3}$$

$$x^4$$

$$\underline{x^4 + x^2 + x}$$

Remainder

$$\boxed{x^2 + x}$$

Therefore,  $c(x) = x^6 + x^3 + x^2 + x$

# Error Detection and Correction

## Single-Bit Error

If the generator has more than one term and the coefficient of  $x^0$  is 1, then all single bit errors can be caught.

### Example:

Which of the following  $g(x)$  values guarantees that a single-bit error is caught? For each case, what is the error that cannot be caught?

- (a)  $x + 1$
- (b)  $x^3$
- (c) 1

### Solution:

- (a) No  $x^i$  can be divisible by  $x + 1$ . In other words,  $x^i/(x + 1)$  always has a remainder. So the syndrome is nonzero. Any single-bit error can be caught.

# Error Detection and Correction

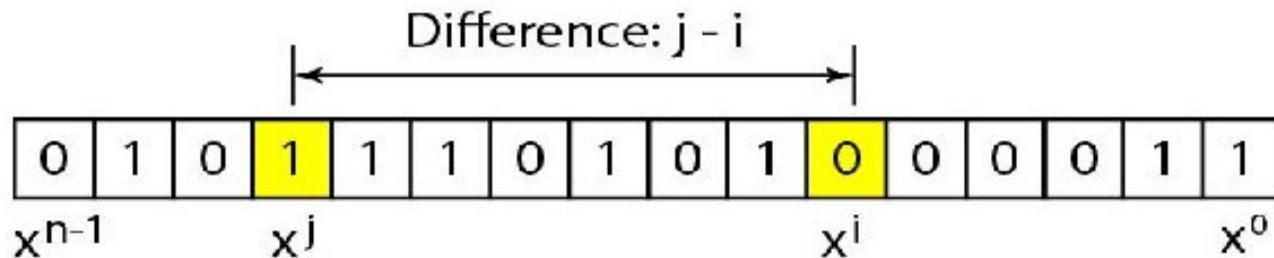
- (b) If  $i$  is equal to or greater than 3, then  $x^i$  is divisible by  $g(x)$ . The remainder of  $x^i/x^3$  is zero, and the receiver is fooled into believing that there is no error, although there might be one. Note that in this case, the corrupted bit must be in position 4 or above. All single-bit errors in positions 1 to 3 are caught.
- (c) For all values of  $i$ ,  $x^i$  is divisible by  $g(x)$ . No single-bit error can be caught. In addition, this  $g(x)$  is useless because it means the codeword is just the dataword augmented with  $n-k$  zeros.

# Error Detection and Correction

## Two Isolated Single-Bit Errors

$$e(x) = x^j + x^i$$

The values of  $i$  and  $j$  define the positions of the errors, and the difference  $j - i$  defines the distance between the two errors.



- ❖ If a generator cannot divide  $x^t + 1$  ( $t$  between 0 and  $n - 1$ ), then all isolated double errors can be detected.

# Error Detection and Correction

## Example:

Find the status of the following generators related to two isolated, single-bit errors.

- (a)  $x + 1$
- (b)  $x^4 + 1$
- (c)  $x^7 + x^6 + 1$
- (d)  $x^{15} + x^{14} + 1$

## Solution:

- (a) This is a very poor choice for a generator. Any two errors next to each other cannot be detected.
- (b) This generator cannot detect two errors that are four positions apart. The two errors can be anywhere, but if their distance is 4, they remain undetected.

# Error Detection and Correction

- (c) This is a good choice for this purpose.
- (d) This polynomial cannot divide any error of type  $x^t + 1$  if  $t$  is less than 32,768. This means that a codeword with two isolated errors that are next to each other or up to 32,768 bits apart can be detected by this generator.

## Odd Numbers of Errors

A generator that contains a factor of  $x + 1$  can detect all odd-numbered errors.

## Some standard generators

- (1) CRC-8 =  $x^8 + x^2 + x + 1$
- (2) (2) CRC-10 =  $x^{10} + x^9 + x^5 + x^4 + x^2 + 1$
- (3) CRC-16 =  $x^{16} + x^{12} + x^5 + 1$
- (4) CRC-32 =  $x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1$

# Error Detection and Correction

## CHECKSUM

- Checksum is an error detection method.
- The checksum is used in the Internet by several protocols.
- The checksum is based on the concept of redundancy.

## Internet Checksum

Internet uses 16-bit checksum. The sender calculates the checksum by following these steps.

### Sender site:

1. The message is divided into 16-bit words.
2. The value of the checksum word is set to 0.
3. All words including the checksum are added using one's complement addition.
4. The sum is complemented and becomes the checksum.
5. The checksum is sent with the data.

# Error Detection and Correction

The receiver uses the following steps for error detection.

## **Receiver site:**

1. The message (including checksum) is divided into 16-bit words.
2. All words are added using one's complement addition.
3. The sum is complemented and becomes the new checksum.
4. If the value of checksum is 0, the message is accepted; otherwise, it is rejected.

# Error Detection and Correction

**Example:** Calculate the checksum for a text of 8 characters ("Forouzan").

**Solution:**

I	0	1	3	Carries
4	6	6	F	(Fo)
7	2	6	F	(ro)
7	5	7	A	(luz)
6	1	6	E	(an)
0	0	0	0	Checksum (initial)
8	F	C	6	Sum (partial)
			1	
8	F	C	7	Sum
7	0	3	8	Checksum (to send)

a. Checksum at the sender site

I	0	1	3	Carries
4	6	6	F	IFo)
7	2	6	F	(ro)
7	5	7	A	(uz)
6	1	6	E	(an)
7	0	3	8	Checksum (received)
F	F	F	E	Sum (partial)
			1	
F	F	F	F	Sum
0	0	0	0	Checksum (new)

b. Checksum at the receiver site

# Error Detection and Correction

## Some questions:

1. What is the Hamming distance for each of the following codewords:
  - a. d (10000, 00000)
  - b. d (10101, 10000)
  - c. d (11111,11111)
  - d. d (000, 000)
2. Find the minimum Hamming distance for the following cases:
  - a. Detection of two errors.
  - b. Correction of two errors.
  - c. Detection of 3 errors or correction of 2 errors.
  - d. Detection of 6 errors or correction of 2 errors.

# Error Detection and Correction

3. Which of the following CRC generators guarantee the detection of a single bit error?
- $x^3 + x + 1$
  - $x^4 + x$
  - 1
  - $x^2 + 1$
4. Sender needs to send the four data items 0x3456, 0xABCC, 0x02BC, and 0xEEEE. Answer the following:
- Find the checksum at the sender site.
  - Find the checksum at the receiver site if there is no error.
  - Find the checksum at the receiver site if the second data item is changed to 0xABCE.
  - Find the checksum at the receiver site if the second data item is changed to 0xABCE and the third data item is changed to 0x02BA.

# **Data Link Control**

# Data Link Control

## Framing:

- Framing in the data link layer separates a message from one source to a destination, or from other messages to other destinations, by adding a sender address and a destination address. The destination address defines where the packet is to go; the sender address helps the recipient acknowledge the receipt.
- Frames can be of fixed size or of variable size.

## Fixed-Size Framing

In fixed-size framing, there is no need for defining the boundaries of the frames; the size itself can be used as a delimiter. An example of this type of framing is the ATM wide-area network, which uses frames of fixed size called cells.

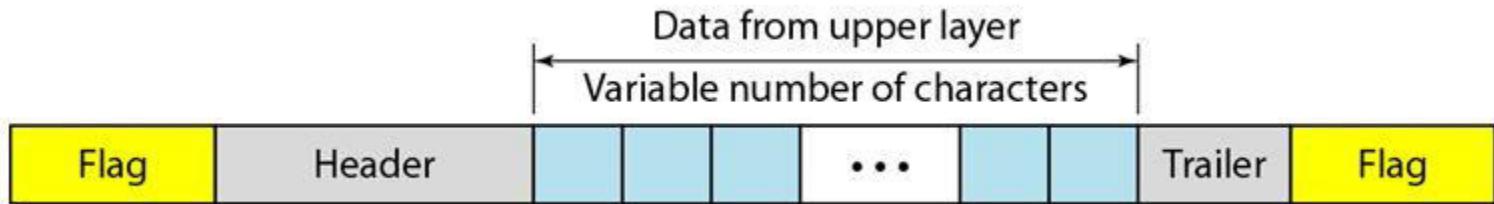
# Data Link Control

## Variable-Size Framing

In variable-size framing, we need a way to define the end of the frame and the beginning of the next. Historically, two approaches were used for this purpose: a character-oriented approach and a bit-oriented approach.

## Character-Oriented Protocols (byte oriented)

In a character-oriented protocol, data to be carried are 8-bit characters from a coding system such as ASCII. The header, which normally carries the source and destination addresses and other control information, and the trailer, which carries error detection or error correction redundant bits, are also multiples of 8 bits. To separate one frame from the next, an 8-bit (1-byte) flag is added at the beginning and the end of a frame. The flag, composed of protocol-dependent special characters, signals the start or end of a frame. Following figure shows the format of a frame in a character-oriented protocol.



# Character-Oriented Protocols (byte oriented)

Any pattern used for the flag could also be part of the information. If this happens, the receiver, when it encounters this pattern in the middle of the data, thinks it has reached the end of the frame. To fix this problem, a byte-stuffing strategy was added to character-oriented framing. In byte stuffing (or character stuffing), a special byte is added to the data section of the frame when there is a character with the same pattern as the flag. The data section is stuffed with an extra byte. This byte is usually called the escape character (ESC), which has a predefined bit pattern. Whenever the receiver encounters the ESC character, it removes it from the data section and treats the next character as data, not a delimiting flag.

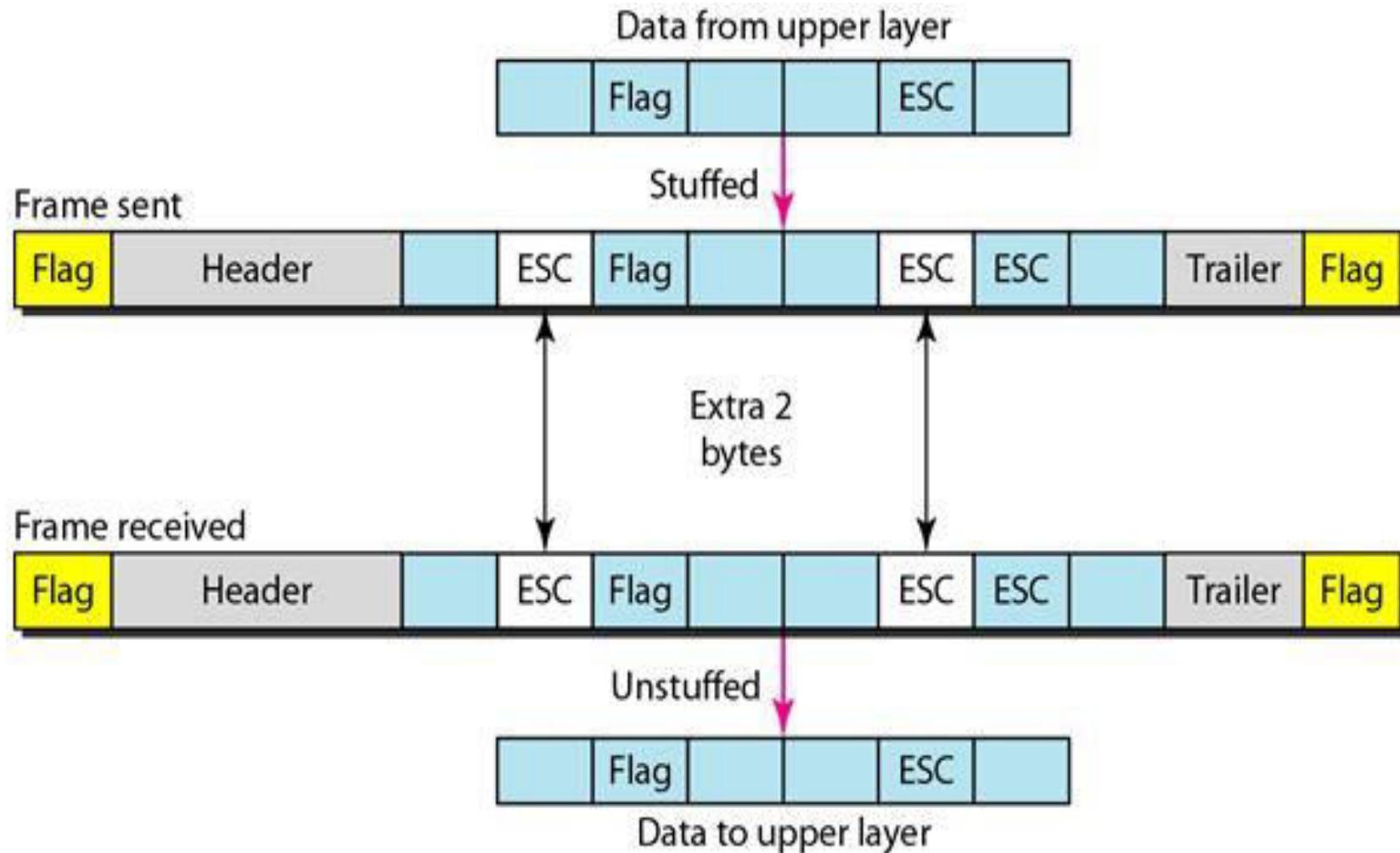
# Character-Oriented Protocols (byte oriented)

Byte stuffing by the escape character allows the presence of the flag in the data section of the frame, but it creates another problem. What happens if the text contains one or more escape characters followed by a flag? The receiver removes the escape character, but keeps the flag, which is incorrectly interpreted as the end of the frame. To solve this problem, the escape characters that are part of the text must also be marked by another escape character. In other words, if the escape character is part of the text, an extra one is added to show that the second one is part of the text.

**Note:** Byte stuffing is the process of adding 1 extra byte whenever there is a flag or escape character in the text.

# Character-Oriented Protocols (byte oriented)

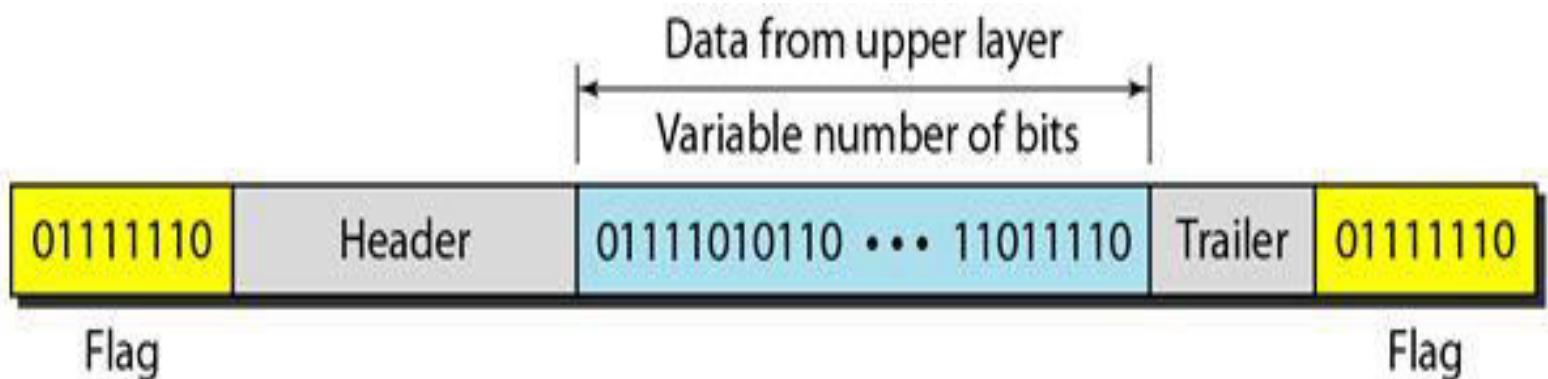
## Byte stuffing and unstuffing



# Bit-Oriented Protocols

## Bit-Oriented Protocols

In a bit-oriented protocol, the data section of a frame is a sequence of bits to be interpreted by the upper layer as text, graphic, audio, video, and so on. However, in addition to headers (and possible trailers), we still need a delimiter to separate one frame from the other. Most protocols use a special 8-bit pattern flag 01111110 as the delimiter to define the beginning and the end of the frame, as shown in following figure .

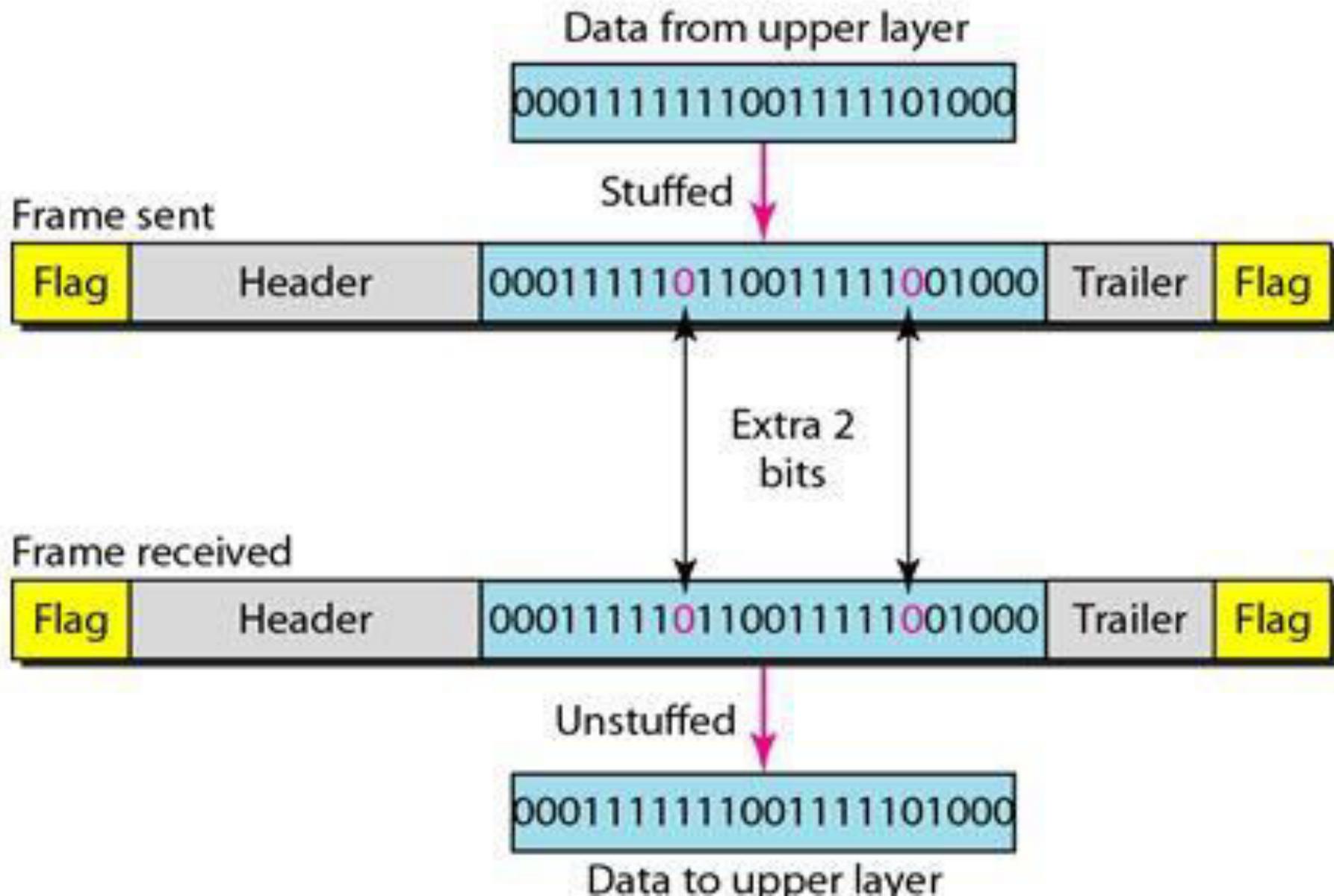


# Bit-Oriented Protocols

This flag can create the same type of problem we saw in the byte-oriented protocols. That is, if the flag pattern appears in the data, we need to somehow inform the receiver that this is not the end of the frame. We do this by stuffing 1 single bit (instead of 1 byte) to prevent the pattern from looking like a flag. The strategy is called **bit stuffing**. In bit stuffing, if a 0 and five consecutive 1 bits are encountered, an extra 0 is added. This extra stuffed bit is eventually removed from the data by the receiver. Note that the extra bit is added after one 0 followed by five 1's regardless of the value of the next bit. This guarantees that the flag field sequence does not inadvertently appear in the frame.

**Note:** Bit stuffing is the process of adding one extra 0 whenever five consecutive 1's follow a 0 in the data, so that the receiver does not mistake the pattern 0111110 for a flag.

# Bit-Oriented Protocols



# Bit-Oriented Protocols

Figure shows bit stuffing at the sender and bit removal at the receiver. Note that even if we have a 0 after five 1's, we still stuff a 0. The 0 will be removed by the receiver. If the flag like pattern 01111110 appears in the data, it will change to 011111010 (stuffed) and is not mistaken as a flag by the receiver. The real flag 01111110 is not stuffed by the sender and is recognized by the receiver.

# Flow and Error Control

The most important responsibilities of the data link layer are flow control and error control.

## Flow control

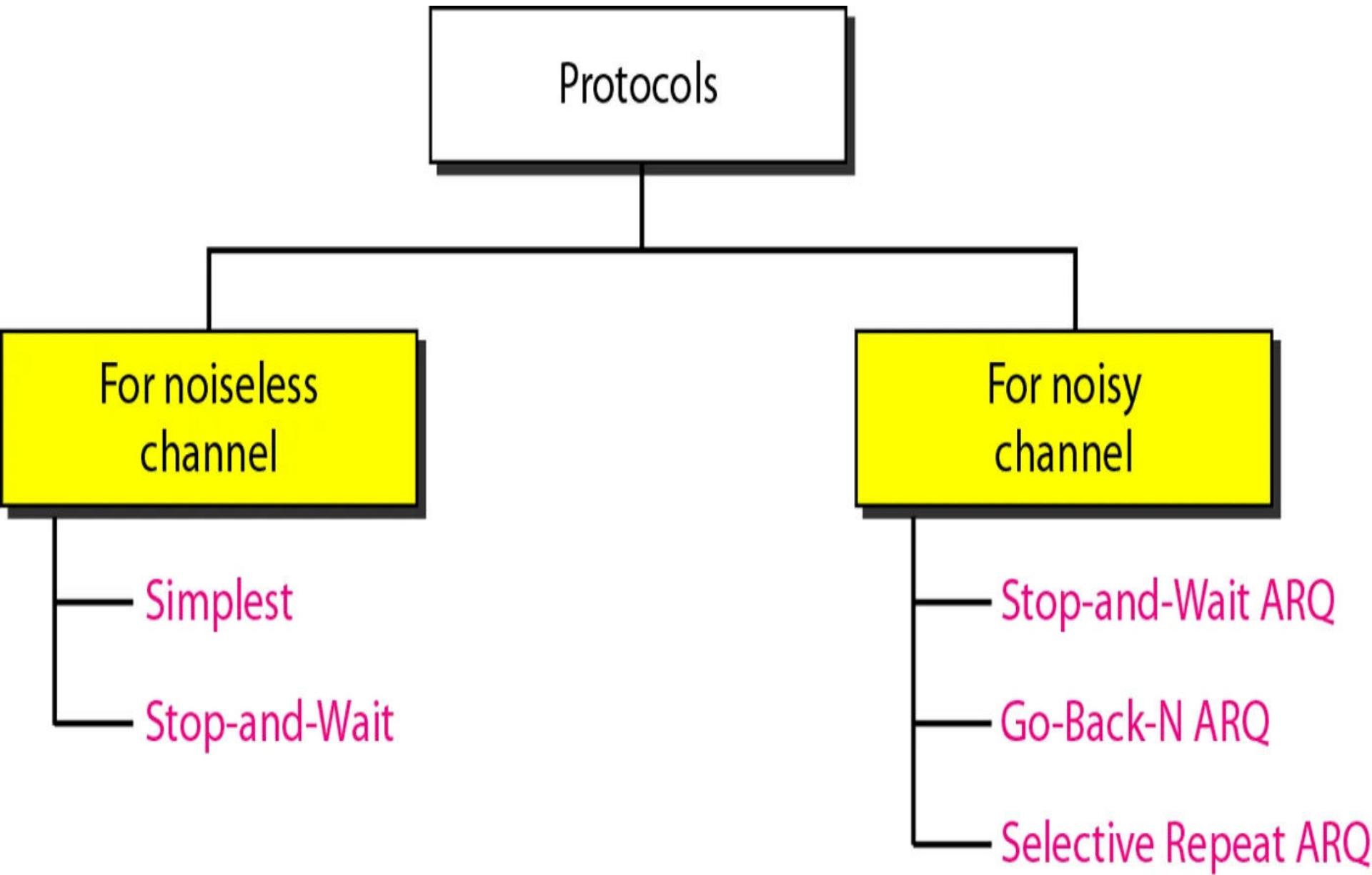
- Flow control coordinates the amount of data that can be sent before receiving an acknowledgment.
- In most protocols, flow control is a set of procedures that tells the sender how much data it can transmit before it must wait for an acknowledgment from the receiver.
- The flow of data must not be allowed to overwhelm the receiver. Any receiving device has a limited speed at which it can process incoming data and a limited amount of memory in which to store incoming data. The receiving device must be able to inform the sending device before those limits are reached and to request that the transmitting device send fewer frames or stop temporarily.

# Flow and Error Control

## Error control

- Error control is both error detection and error correction. It allows the receiver to inform the sender of any frames lost or damaged in transmission and coordinates the retransmission of those frames by the sender.
- Error control in the data link layer is often implemented simply: Any time an error is detected in an exchange, specified frames are retransmitted. This process is called automatic repeat request (ARQ).

# Data link control protocol



# Data link control protocol

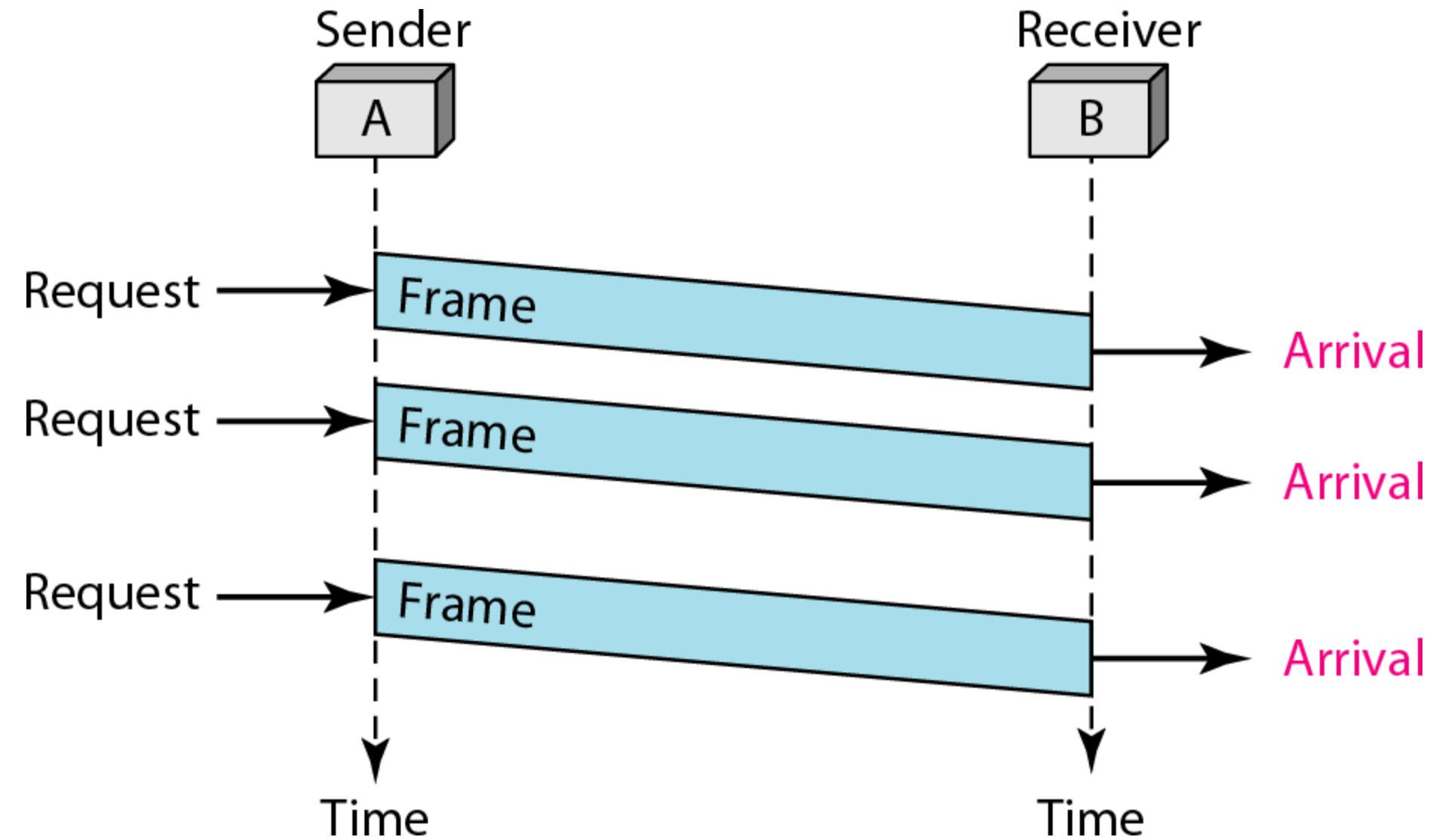
## Noiseless Channels

- Assume we have an ideal channel in which no frames are lost, duplicated, or corrupted.
- We have two protocols for this type of channel. The first is a protocol that does not use flow control; the second is the one that does.
- Neither has error control because we have assumed that the channel is a perfect noiseless channel.

# Simplest Protocol

- ❖ It is one that has no flow or error control.
- ❖ It is a unidirectional protocol in which data frames are traveling in only one direction-from the sender to receiver.
- ❖ We assume that the receiver can immediately handle any frame it receives with a processing time that is small enough to be negligible. The data link layer of the receiver immediately removes the header from the frame and hands the data packet to its network layer, which can also accept the packet immediately. In other words, the receiver can never be overwhelmed with incoming frames.

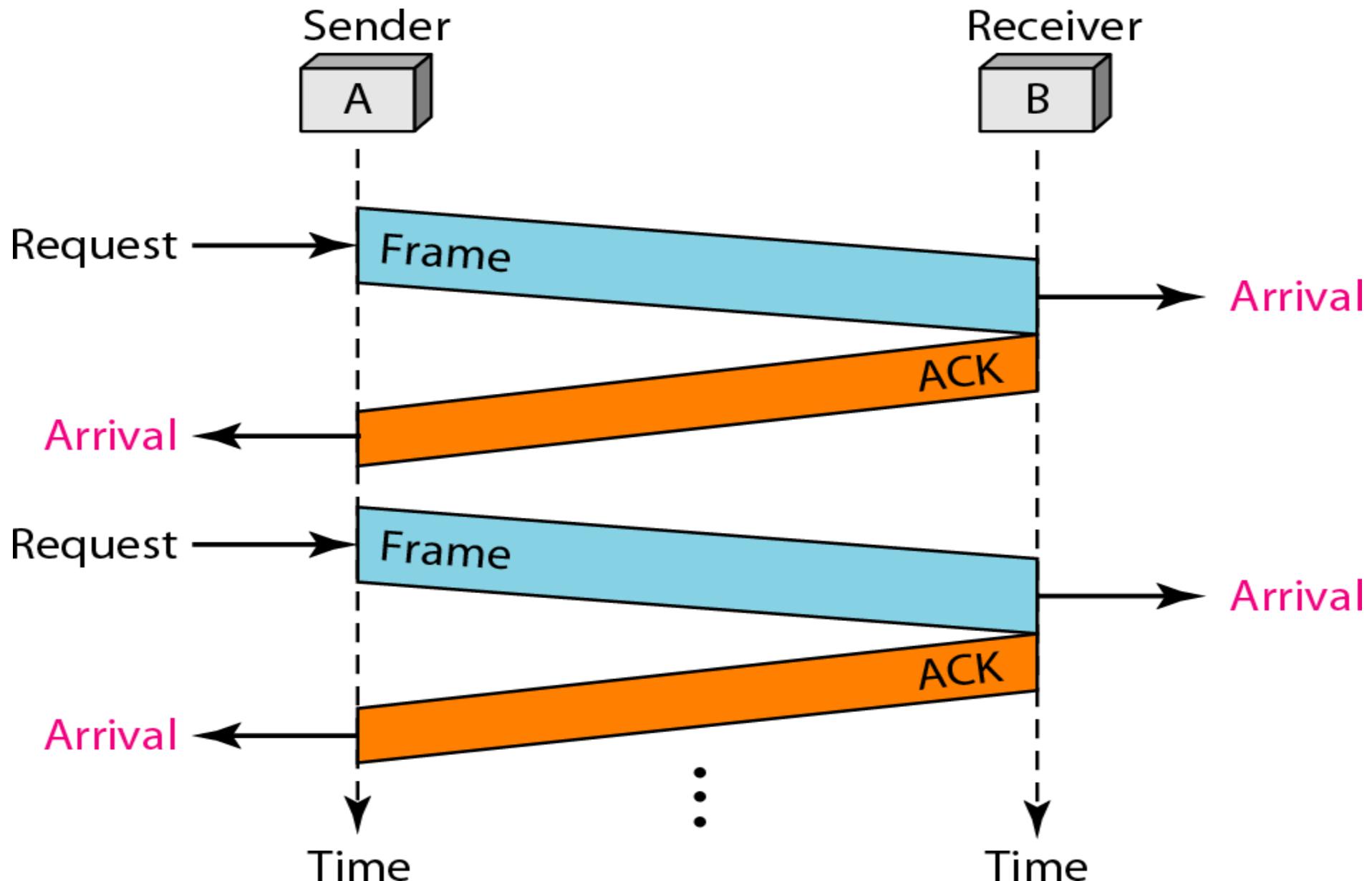
# Simplest Protocol



# Stop-and-Wait Protocol

- ❖ The sender sends one frame, stops until it receives confirmation from the receiver (okay to go ahead), and then sends the next frame.
- ❖ We still have unidirectional communication for data frames, but auxiliary ACK frames (simple tokens of acknowledgment) travel from the other direction.
- ❖ We add flow control to our previous protocol.

# Stop-and-Wait Protocol



# NOISY CHANNELS

## Stop-and-Wait Automatic Repeat Request (Stop-and-Wait ARQ)

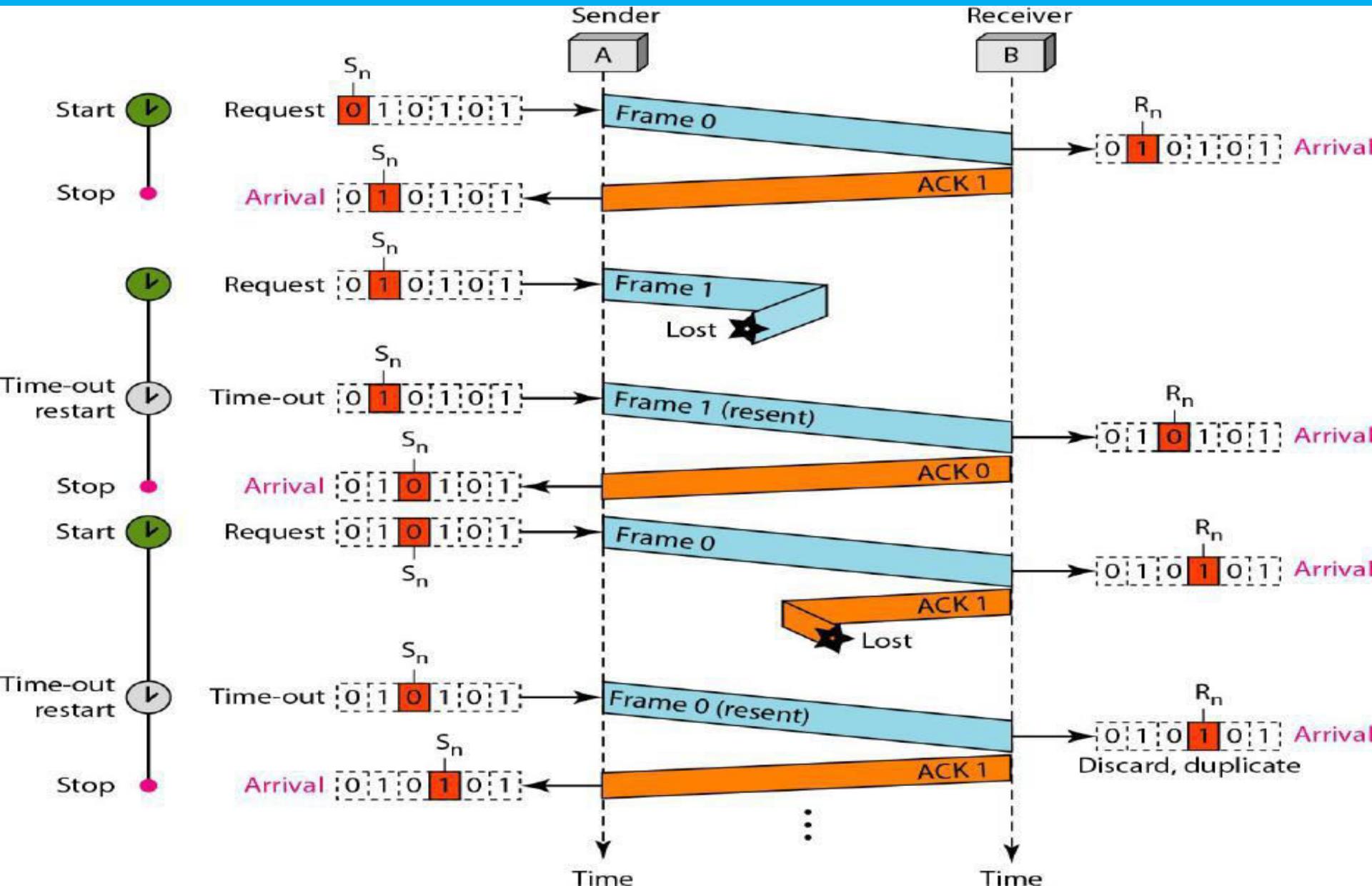
- ❖ This protocol adds a simple error control mechanism to the Stop-and-Wait Protocol.
- ❖ To detect and correct corrupted frames, we need to add redundancy bits to our data frame. When the frame arrives at the receiver site, it is checked and if it is corrupted, it is silently discarded.
- ❖ Lost frames are more difficult to handle than corrupted ones.
- ❖ To handle lost frames, this protocol uses **sequence number**.

# Stop-and-Wait ARQ

## Stop-and-Wait Automatic Repeat Request (Stop-and-Wait ARQ)

- ❖ When the receiver receives a data frame that is out of order, this means that frames were either lost or duplicated.
- ❖ The corrupted and lost frames need to be resent in this protocol.
- ❖ When the sender sends a frame, it keeps a copy of the sent frame. At the same time, it starts a timer. If the timer expires and there is no ACK for the sent frame, the frame is resent, the copy is held, and the timer is restarted.

# Stop-and-Wait ARQ



# Stop-and-Wait ARQ

- ❖ Since an ACK frame can also be corrupted and lost, it too needs redundancy bits and a sequence number.
- ❖ The ACK frame for this protocol has a sequence number field.
- ❖ In this protocol, the sender simply discards a corrupted ACK frame or ignores an out-of-order one.
- ❖ A field is added to the data frame to hold the sequence number of that frame.

# Stop-and-Wait ARQ

- ❖ If the number of bits used for sequence number is  $m$ , then range of sequence numbers is 0 to  $2^m-1$ .
- ❖ This protocol uses 1 bit for sequence number i.e.  $m=1$ .
- ❖ This protocol uses the following sequence numbers 0, 1, 0, 1, 0, 1, 0, 1, 0, 1.....
- ❖ This protocol uses the sliding window concept.
- ❖ The size of both sender and receiver window in this protocol is 1.

# Stop-and-Wait ARQ

## Efficiency

- ❖ The Stop-and-Wait ARQ discussed in the previous section is very inefficient if our channel is thick and long. By thick, we mean that our channel has a large bandwidth; by long, we mean the round-trip delay is long. The product of these two is called the bandwidth delay Product.
- ❖ The channel is always there. If we do not use it, we are inefficient. The bandwidth-delay product is a measure of the number of bits we can send out of our system while waiting for news from the receiver.

# Stop-and-Wait ARQ

## Example:

Assume that, in a Stop-and-Wait ARQ system, the bandwidth of the line is 1 Mbps, and 1 bit takes 20 ms to make a round trip. What is the bandwidth-delay product? If the system data frames are 1000 bits in length, what is the utilization percentage of the link?

# Stop-and-Wait ARQ

## Solution:

The bandwidth-delay product is

$$\begin{aligned} &= 1*10^6 * 20*10^{-3} = 20*10^3 \\ &= 20000 \text{ bits} \end{aligned}$$

- ❖ The system can send 20,000 bits during the time it takes for the data to go from the sender to the receiver and then back again. However, the system sends only 1000 bits. We can say that the link utilization is only 1000/20,000, or 5 percent.
- ❖ For this reason, for a link with a high bandwidth or long delay, the use of Stop-and-Wait ARQ wastes the capacity of the link.

# Stop-and-Wait ARQ

## Example:

What is the utilization percentage of the link in the previous example if we have a protocol that can send up to 15 frames before stopping and worrying about the acknowledgments?

## Solution:

The bandwidth-delay product is still 20,000 bits. The system can send up to 15 frames or 15,000 bits during a round trip. This means the utilization is  $15,000/20,000$ , or 75 percent.

**Note:** Of course, if there are damaged frames, the utilization percentage is much less because frames have to be resent.

# Go-Back-N Automatic Repeat Request

- ❖ To improve the efficiency of transmission (filling the pipe), multiple frames must be in transition while waiting for acknowledgment.
- ❖ In this protocol we can send several frames before receiving acknowledgments; we keep a copy of these frames until the acknowledgments arrive.
- ❖ In the Go-Back-N Protocol, the sequence numbers are modulo  $2^m$ , where  $m$  is the size of the sequence number field in bits.
- ❖ In this protocol, the sliding window is an abstract concept that defines the range of sequence numbers that is the concern of the sender and receiver.

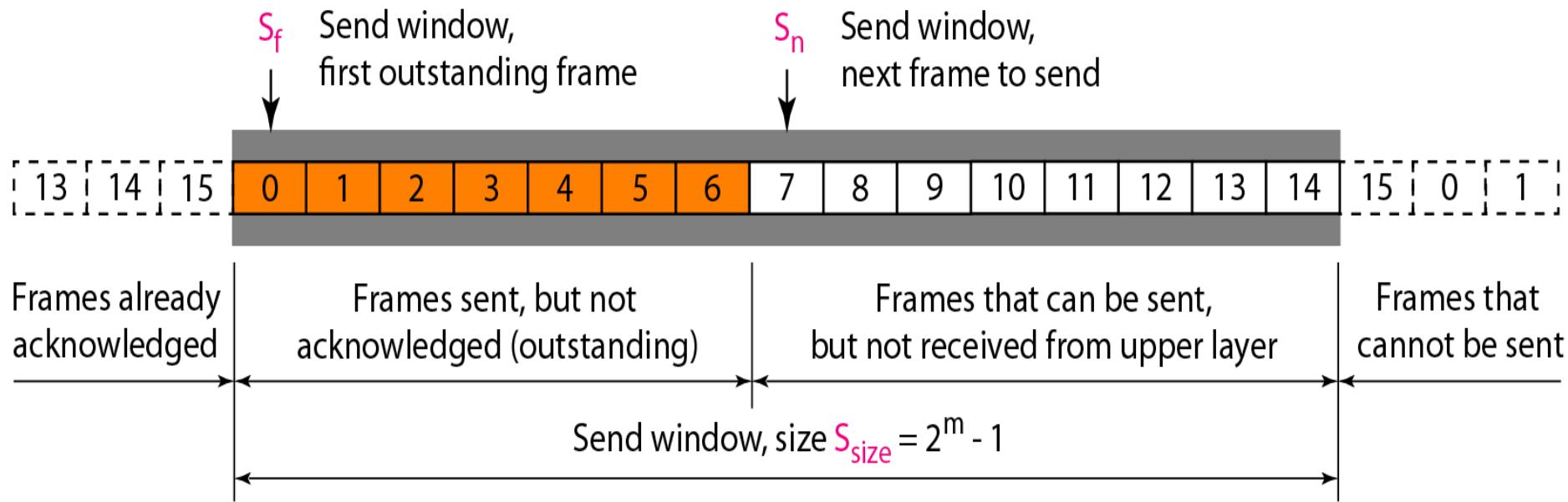
# Go-Back-N Automatic Repeat Request

- ❖ If  $m=4$  bits are used for sequence number, then the only sequence numbers are 0 through 15 inclusive. So the sequence numbers are

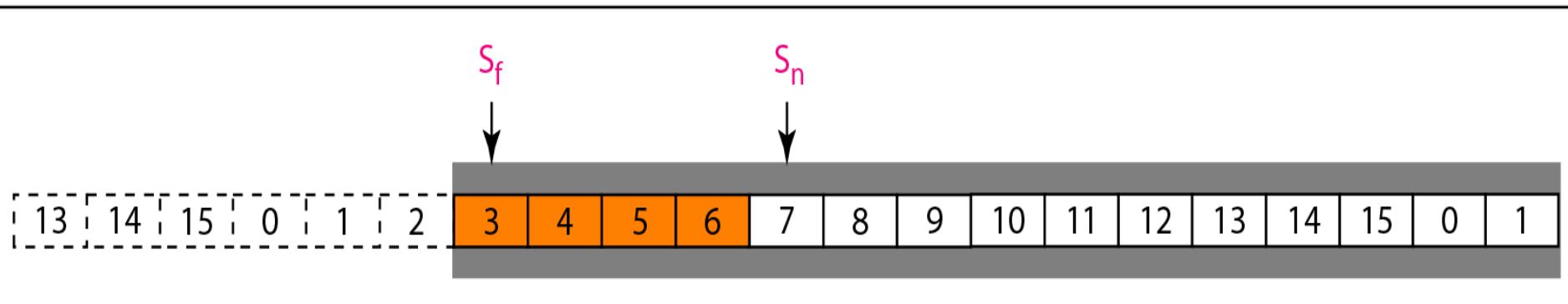
0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, ...

- ❖ The maximum size of the sender window is  $2^m-1$ .
- ❖ The size of the receiver window is 1.
- ❖ The sender window at any time divides the possible sequence numbers into four regions.

# Go-Back-N Automatic Repeat Request

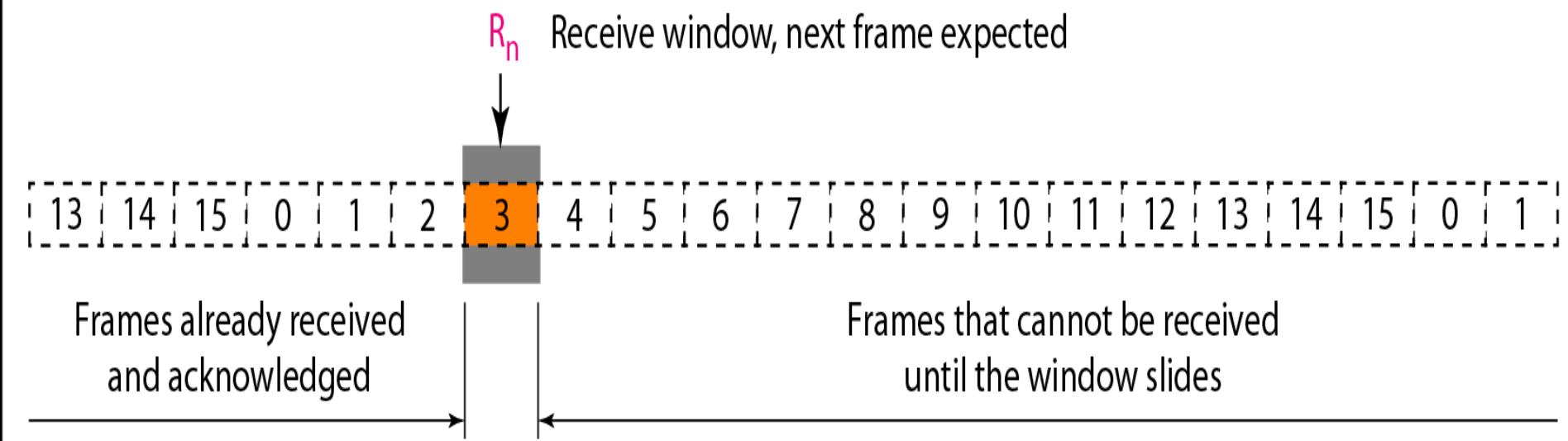


a. Send window before sliding

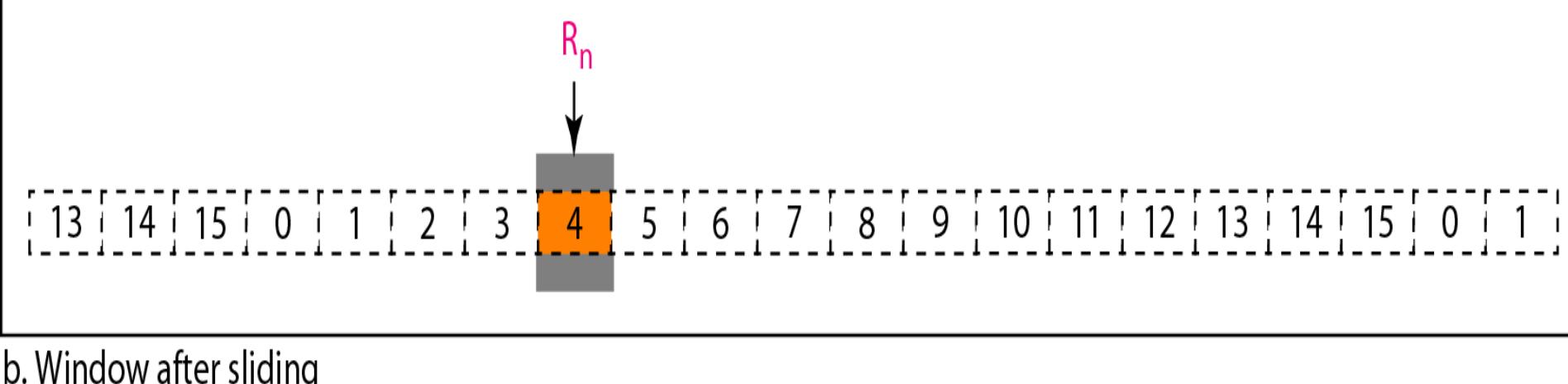


b. Send window after sliding

# Go-Back-N Automatic Repeat Request



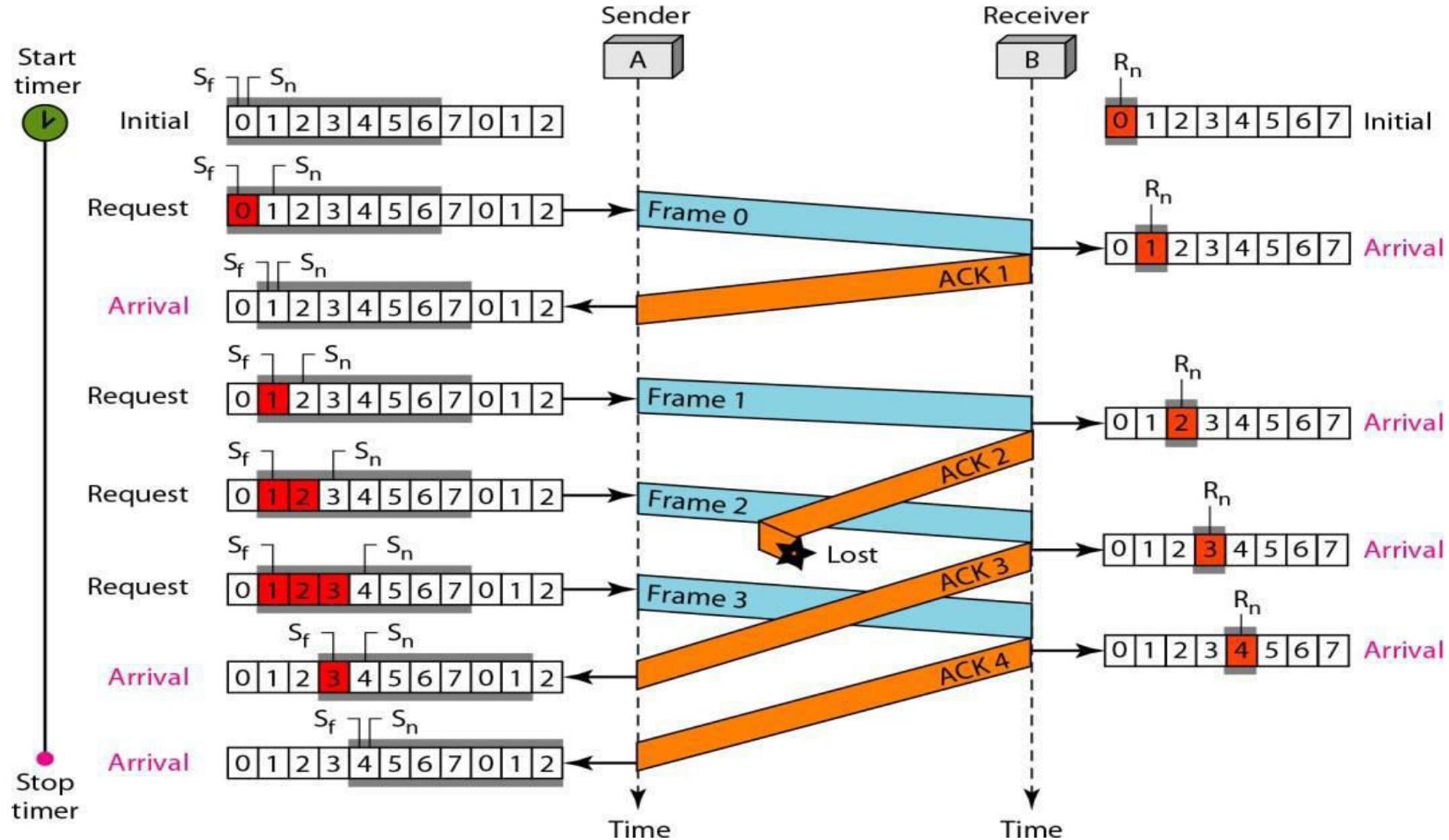
a. Receive window



b. Window after sliding

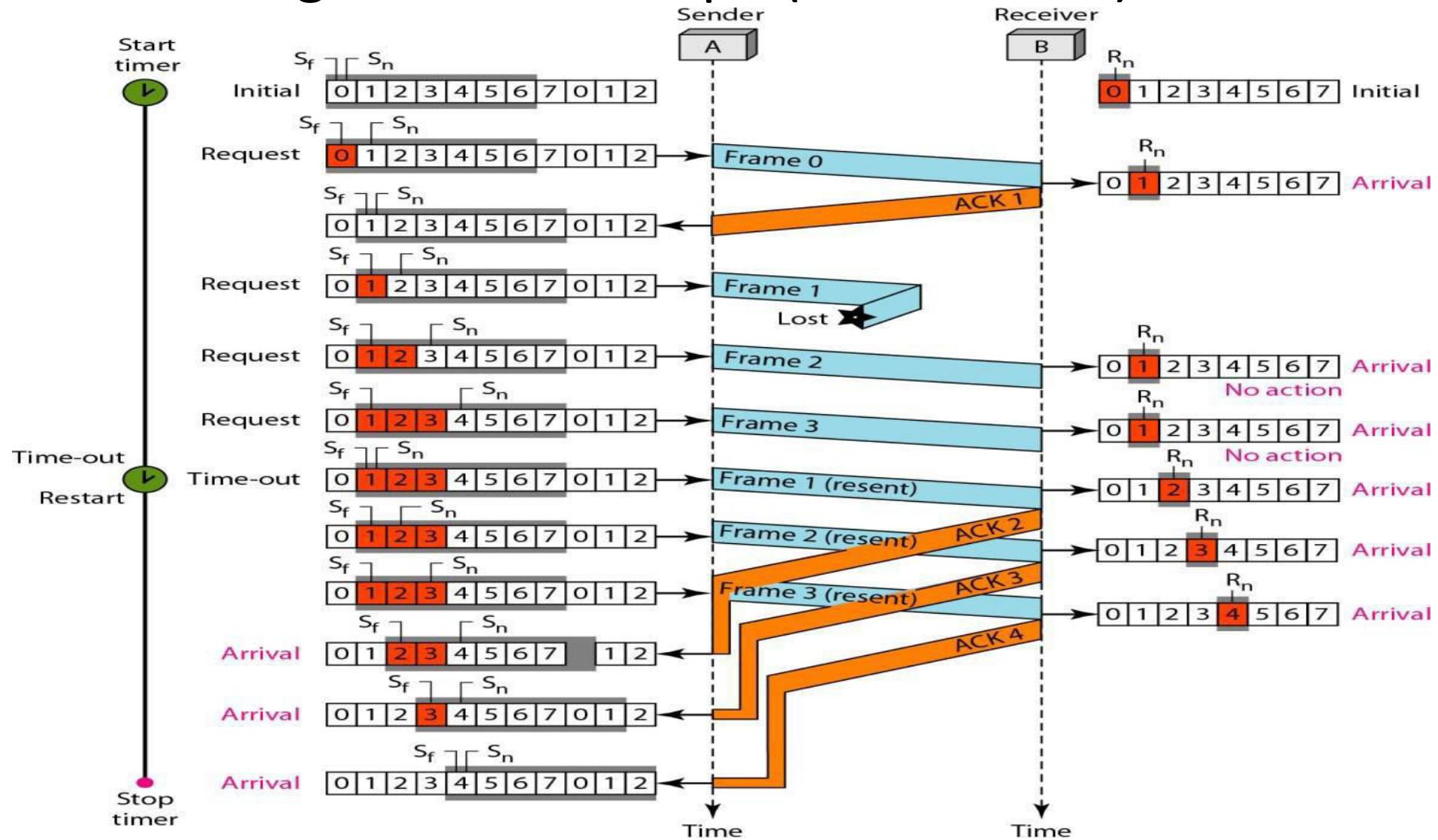
# Go-Back-N Automatic Repeat Request

❖ Below figure is an example(if ack lost)



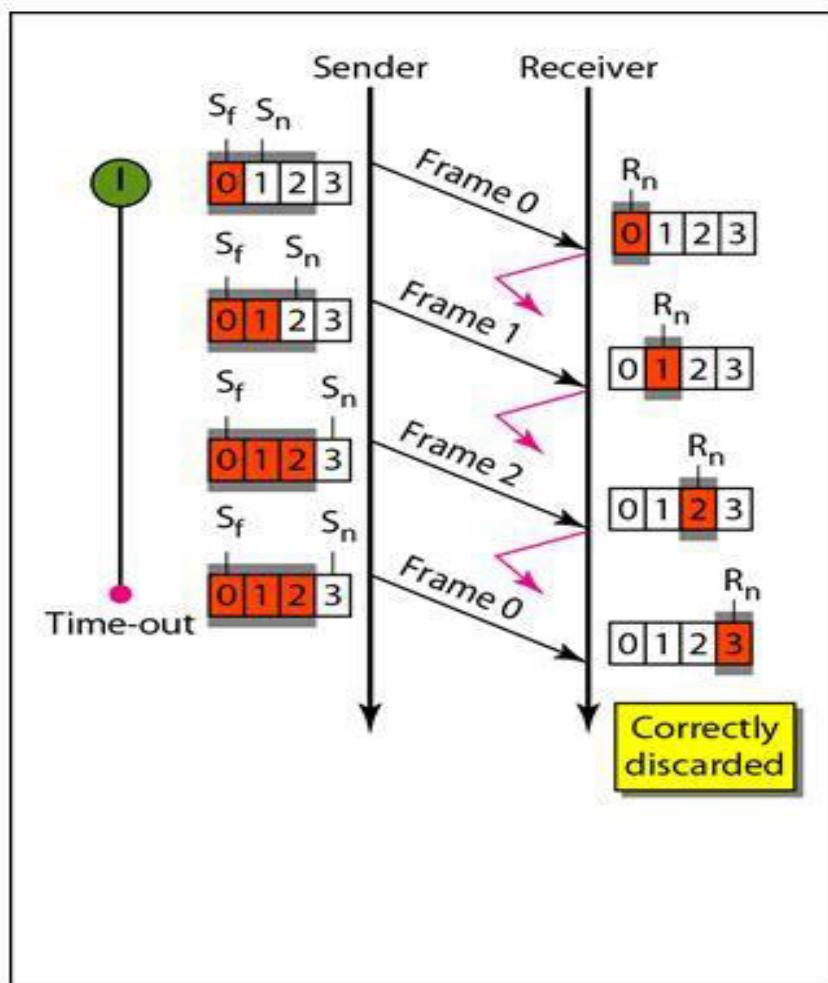
# Go-Back-N Automatic Repeat Request

❖ Below figure is an example(if frame lost)

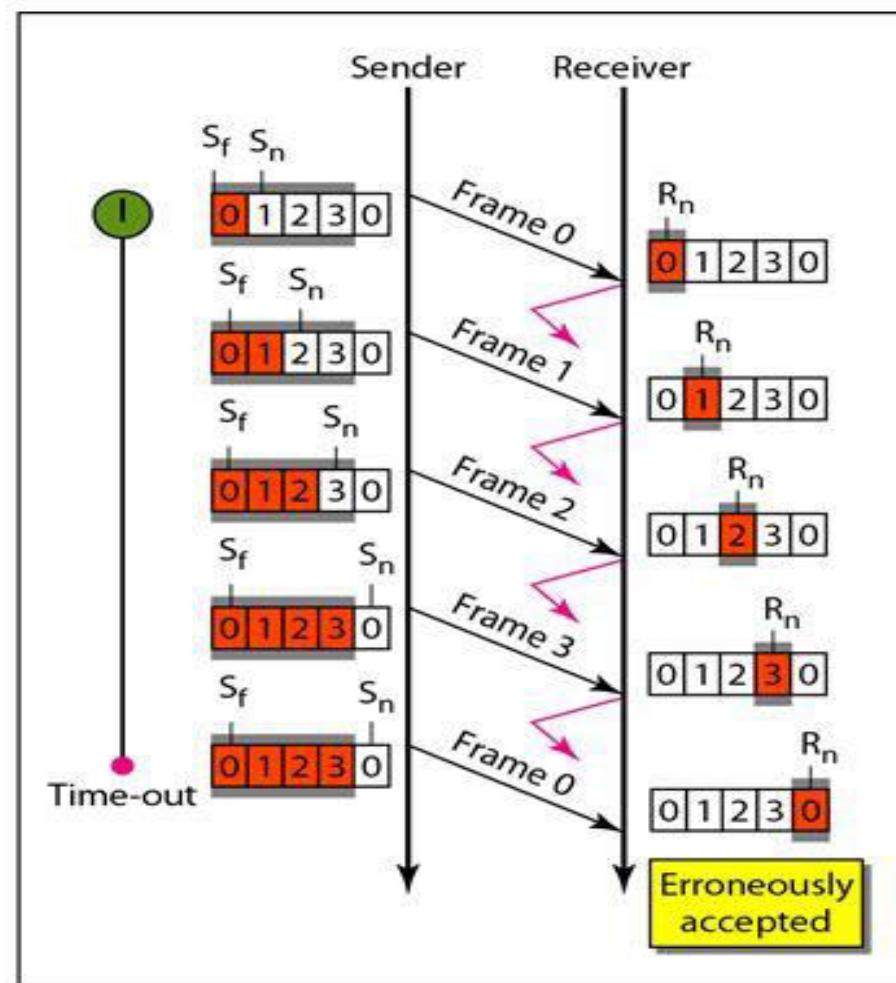


# Go-Back-N Automatic Repeat Request

**Note:** In Go-Back-N ARQ, the size of the sender window must be less than  $2^m$ ; the size of the receiver window is always 1.



a. Window size  $< 2^m$



b. Window size  $= 2^m$

# Go-Back-N Automatic Repeat Request

**Note:** Stop-and-Wait ARQ is a special case of Go-Back-N ARQ in which the size of the sender window is 1.

## Efficiency:

The efficiency of Go-Back-N ARQ ,

$$\text{Efficiency} = N/(1+2a),$$

Where N is the size of sender window and  $a = T_p/T_t$ .

Where  $T_p$  is propagation delay and  $T_t$  is the transmission delay

Also,  $T_t = D/B$ ;

and here D = data size and B = bandwidth

And  $T_p = d/v$ ,

here d = distance and v = propagation speed.

# Go-Back-N Automatic Repeat Request

Now to find the effective bandwidth (or throughput),

Effective bandwidth = efficiency \* bandwidth,

which means,

Effective bandwidth =  $(N/(1+2a)) * B$

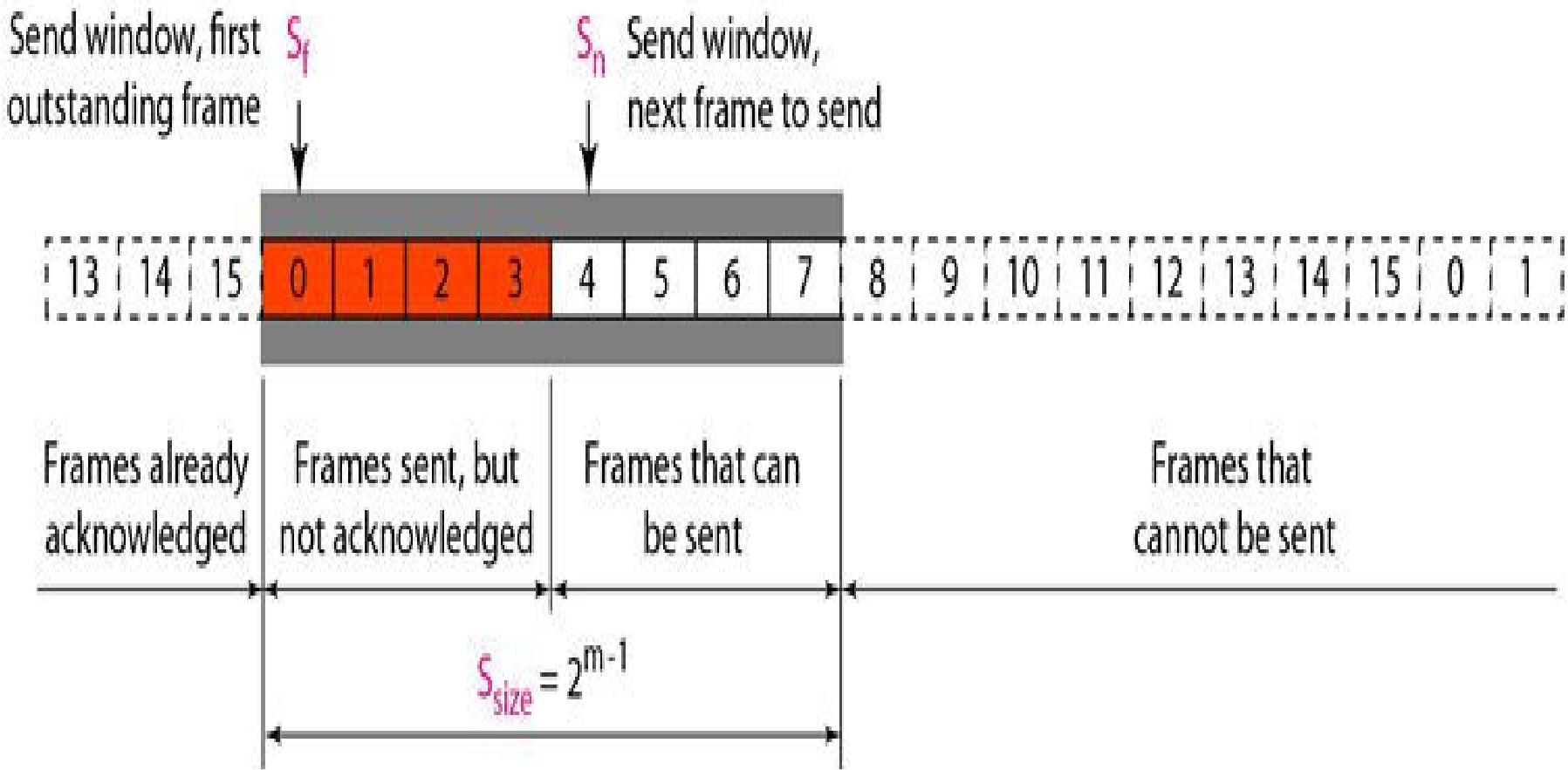
# Selective Repeat Automatic Repeat Request

- ❖ Go-Back-N ARQ protocol is inefficient for noisy links because in a noisy link a frame has a higher probability of damage, which means the resending of multiple frames. This resending uses up the bandwidth and slows down the transmission.
- ❖ For noisy links, there is another mechanism that does not resend N frames when just one frame is damaged; only the damaged frame is resent. This mechanism is called Selective Repeat ARQ.
- ❖ It is more efficient for noisy links, but the processing at the receiver is more complex.

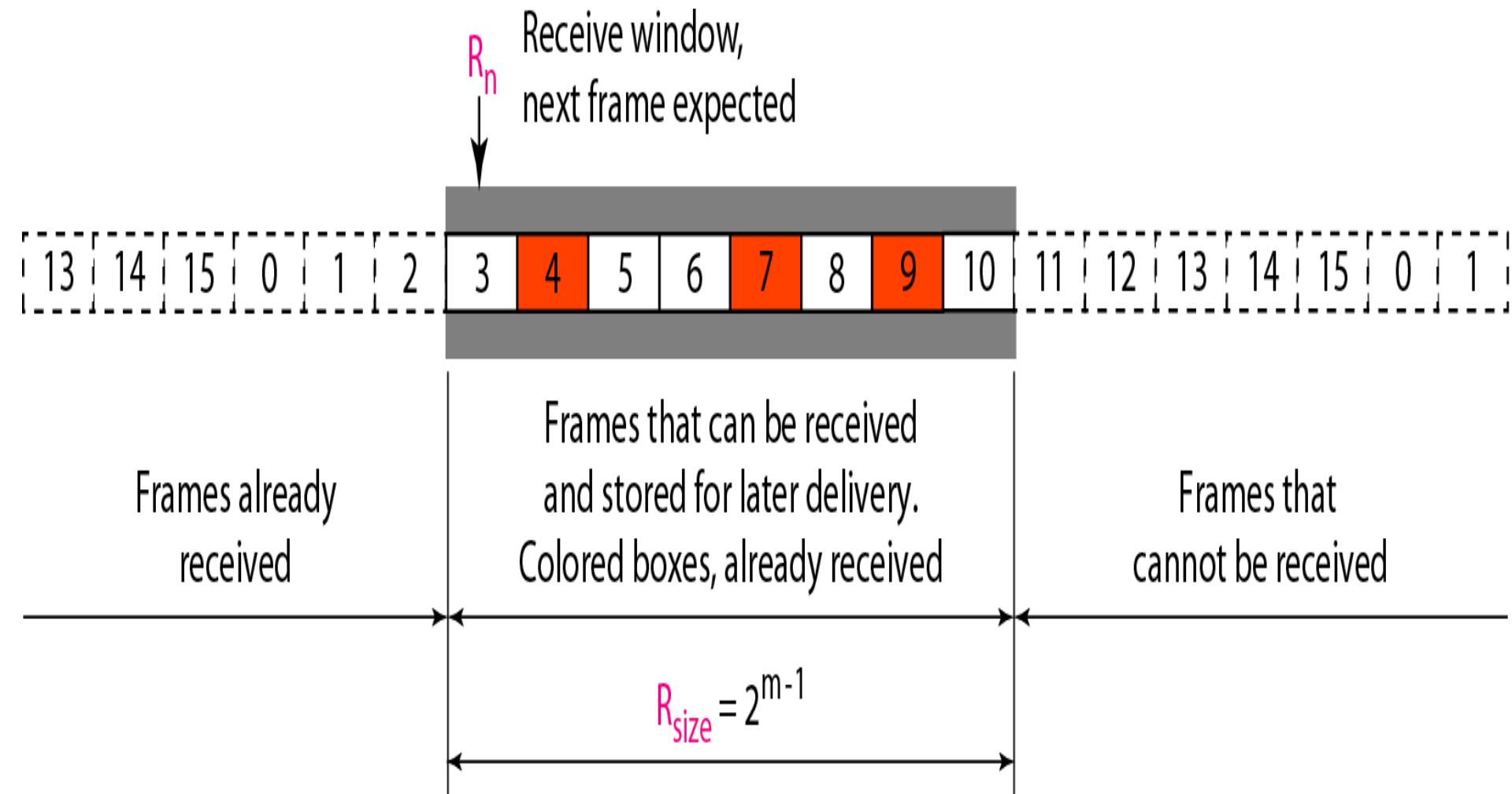
# Selective Repeat Automatic Repeat Request

The size of the sender window is  $2^{m-1}$ .

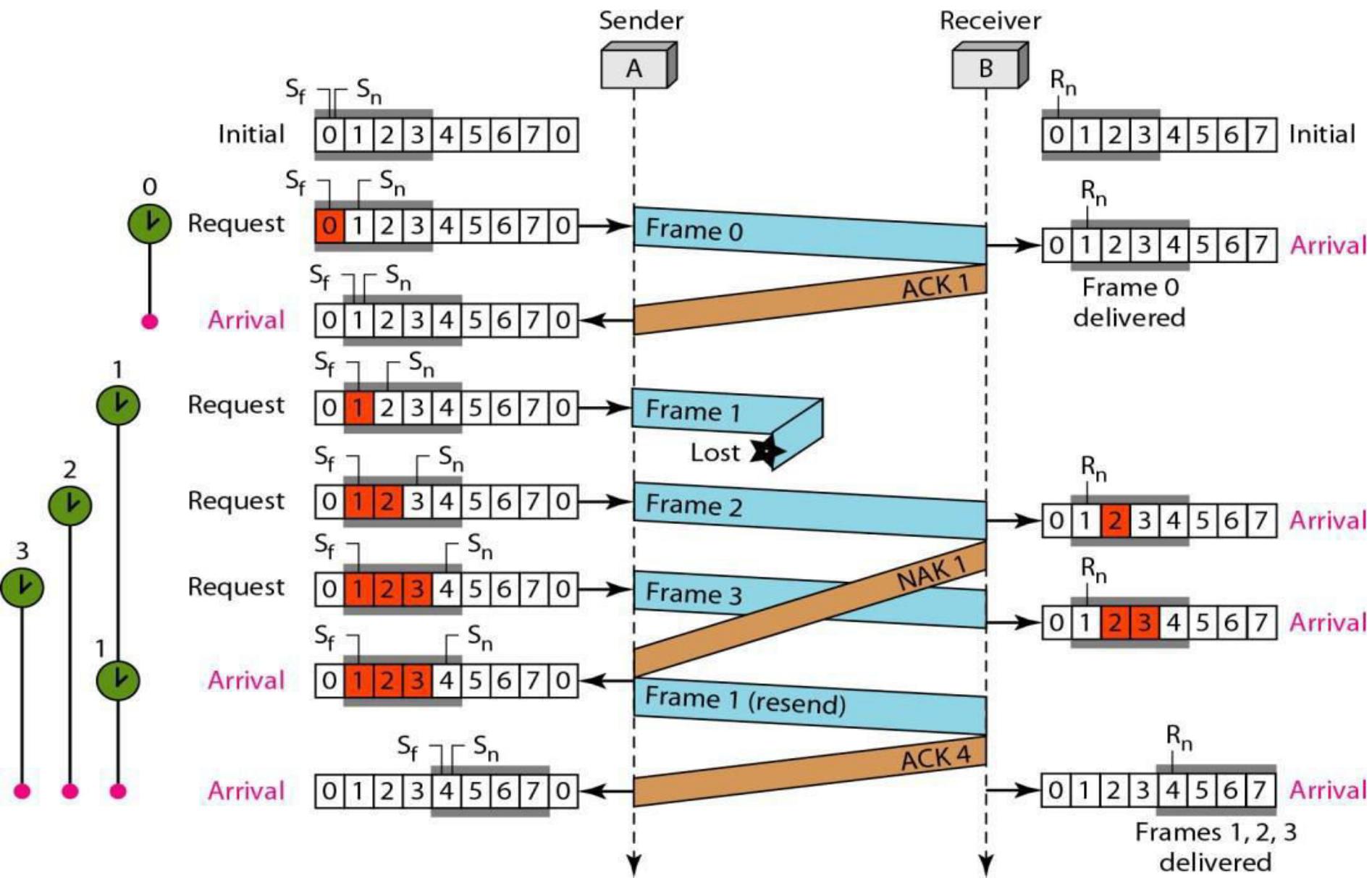
The size of the receiver window is  $2^{m-1}$ .



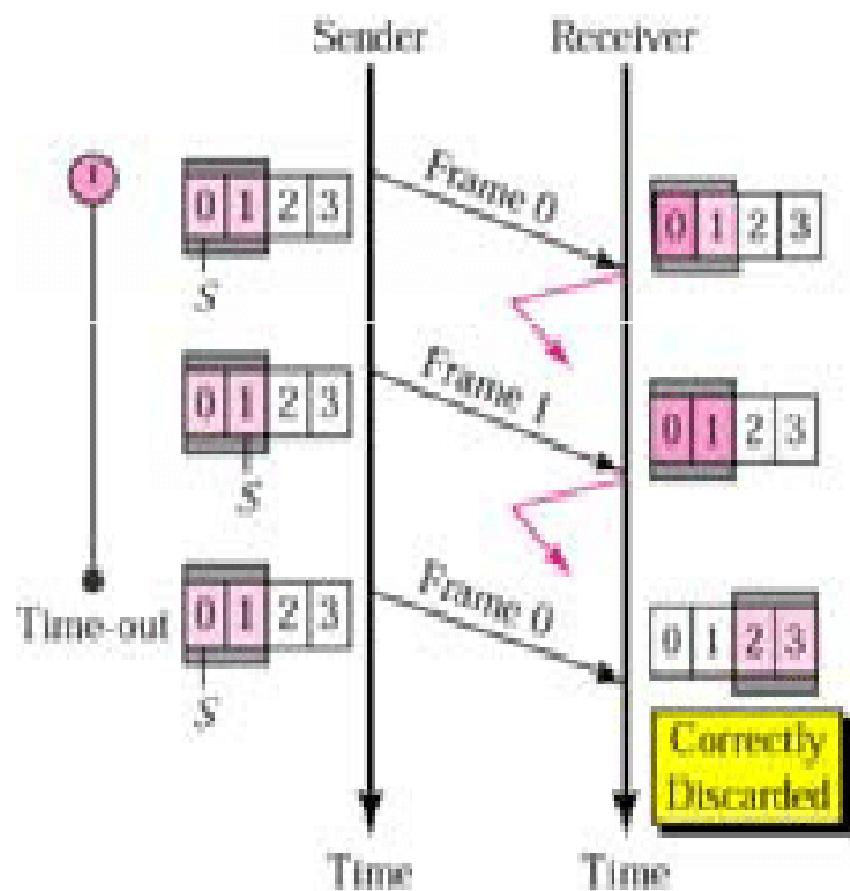
# Selective Repeat Automatic Repeat Request



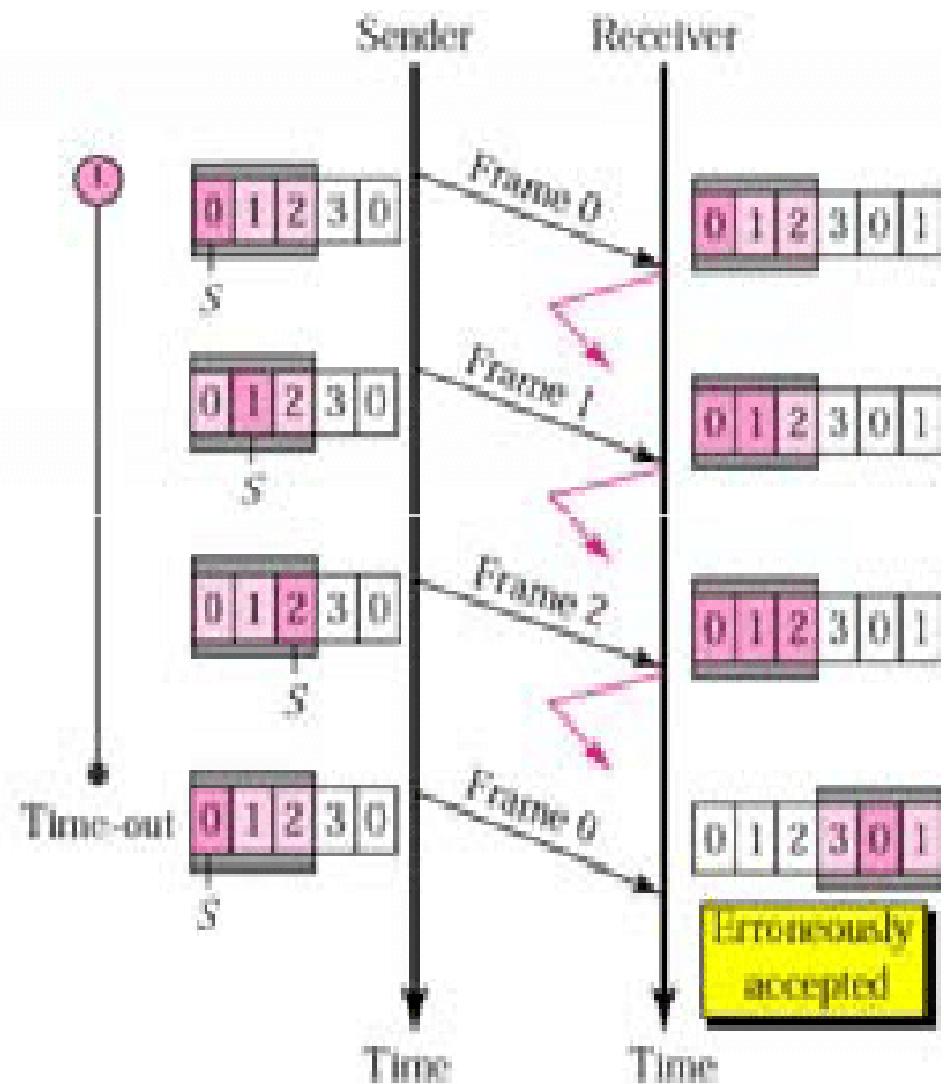
# Selective Repeat Automatic Repeat Request



# Selective Repeat Automatic Repeat Request



a. Window size =  $2^{m-1}$



b. Window size >  $2^{m-1}$

# Selective Repeat Automatic Repeat Request

## Efficiency:

The efficiency of selective repeat protocol is the same as of Go-Back-N ARQ protocol's efficiency.

$$\text{Efficiency} = N/(1+2a),$$

Where N is the size of sender window and  $a = T_p/T_t$ .

# Piggybacking

A technique called piggybacking is used to improve the efficiency of the bidirectional protocols. When a frame is carrying data from A to B, it can also carry control information about arrived (or lost) frames from B; when a frame is carrying data from B to A, it can also carry control information about the arrived (or lost) frames from A.

# Data Link Layer

## Exercise

1. A sender sends a series of packets to the same destination using 5-bit sequence numbers. If the sequence number starts with 0, what is the sequence number after sending 100 packets?
2. Using 5-bit sequence numbers, what is the maximum size of the sender and receiver windows for each of the following protocols?
  - a. Stop-and-Wait ARQ
  - b. Go-Back-N ARQ
  - c. Selective-Repeat ARQ

# Data Link Layer

3. A system uses the Stop-and-Wait ARQ Protocol. If each packet carries 1000 bits of data, how long does it take to send 1 million bits of data if the distance between the sender and receiver is 5000 Km and the propagation speed is  $2 \times 10^8$  m? Ignore transmission, waiting, and processing delays. We assume no data or control frame is lost or damaged.
4. Repeat Exercise 3 using the Go-back-N ARQ Protocol with a window size of 7. Ignore the overhead due to the header and trailer.
5. Repeat Exercise 3 using the Selective-Repeat ARQ Protocol with a window size of 4. Ignore the overhead due to the header and the trailer.

6. Consider a selective repeat sliding window protocol uses a frame size of 1KB to send data on a 15Mbps link with a one-way latency of 50 ms. To achieve a link utilization of 60%, find the minimum number of bits required to represent the sequence number field.

7. Consider the sliding window flow-control protocol operating between a sender and a receiver over a full-duplex free link. Assume the following:

- (i) The time taken for processing the data frame by the receiver is negligible.
- (ii) The time taken for processing the acknowledgement frame by the sender is negligible.
- (iii) The sender has infinite number of frames available for transmission
- (iv) The size of the data frame is 2,000 bits and the size of the acknowledgement frame is 10 bits.
- (v) The link data rate in each direction is 1 Mbps ( $10^6$  bits per second)
- (vi) One way propagation delay of the link is 100 milliseconds

The minimum value of the sender's window size in terms of the number of frames, (rounded to the nearest integer) needed to achieve a link utilization of 50% is \_\_\_\_\_.

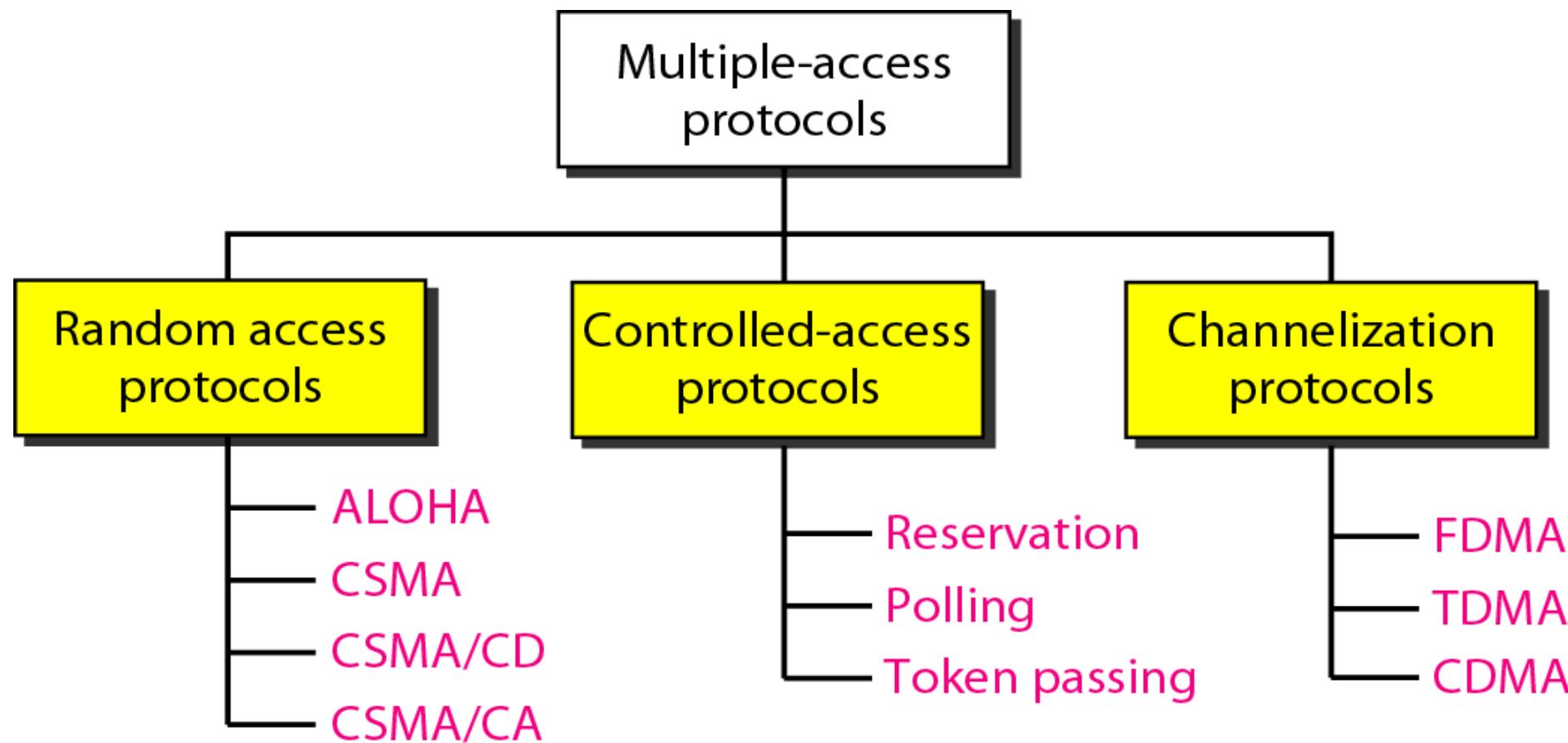
# **Media Access Control**

# Media Access Control

- ❖ Data link layer is considered as two sub-layers.
- ❖ The upper sub-layer is responsible for data link control. The upper sub-layer that is responsible for flow and error control is called the logical link control (LLC) layer.
- ❖ The lower sub-layer is responsible for resolving access to the shared media. The lower sub-layer that is mostly responsible for multiple access resolution is called the media access control (MAC) layer.

# Media Access Control

When nodes or stations are connected and use a common link, called a multipoint or broadcast link, we need a multiple-access protocol to coordinate access to the link.



# Random Access Protocol

- ❖ In random access or contention methods, no station is superior to another station and none is assigned the control over another.
- ❖ No station permits, or does not permit, another station to send.
- ❖ At each instance, a station that has data to send uses a procedure defined by the protocol to make a decision on whether or not to send. This decision depends on the state of the medium (idle or busy).
- ❖ If more than one station tries to send, there is an access conflict (collision) and the frames will be either destroyed or modified.

# Random Access Protocol

To avoid access conflict or to resolve it when it happens, each station follows a procedure that answers the following questions:

- ❖ When can the station access the medium?
- ❖ What can the station do if the medium is busy?
- ❖ How can the station determine the success or failure of the transmission?
- ❖ What can the station do if there is an access conflict?

# Random Access Protocol

## ALOHA Protocol

ALOHA was the earliest random access method. There are two types of ALOHA.

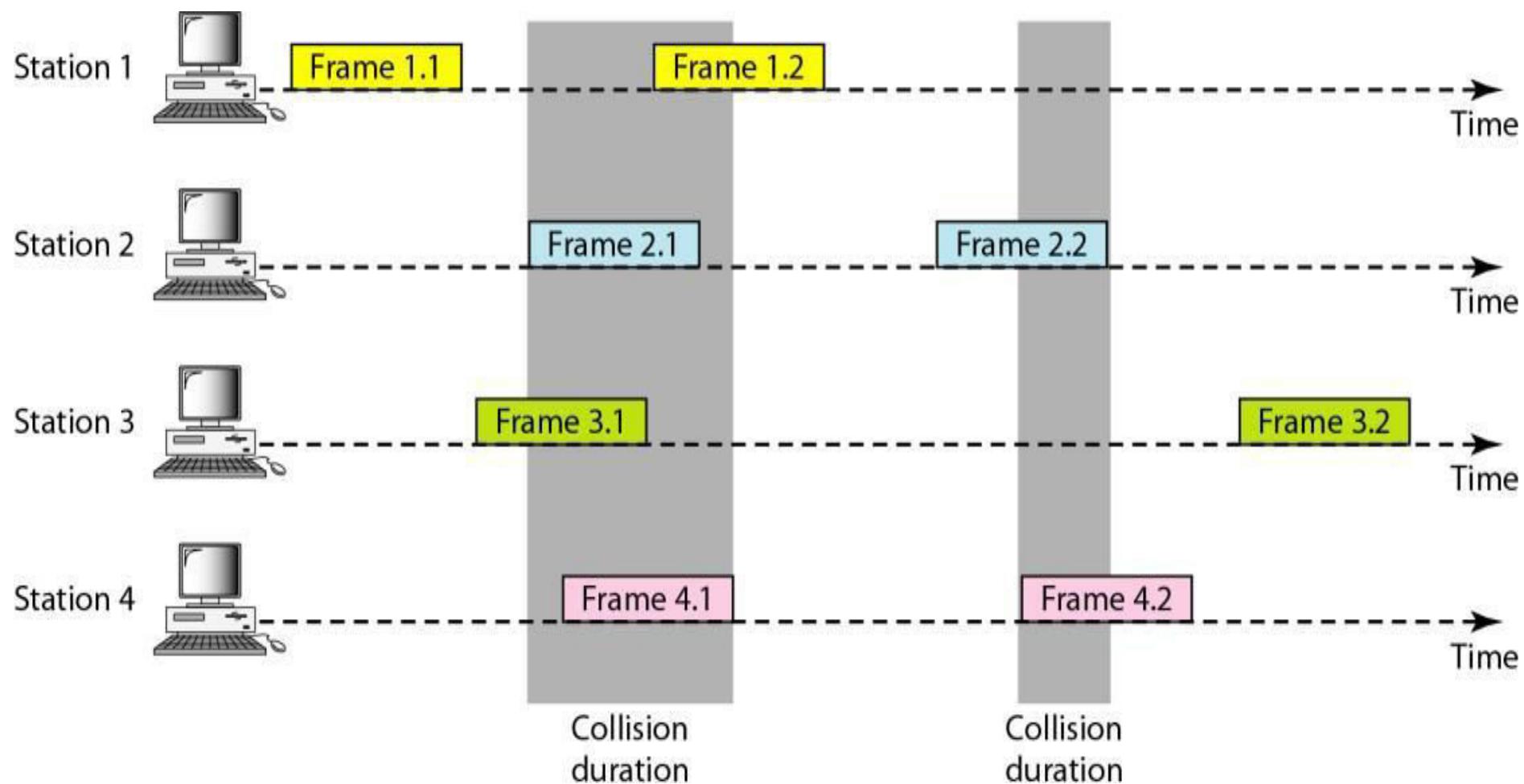
1. Pure ALOHA
2. Slotted ALOHA

## Pure ALOHA

- ❖ The original ALOHA protocol is called pure ALOHA. This is a simple protocol.
- ❖ Full form of ALOHA is **Additive Links On-line Hawaii Area.**
- ❖ In this protocol each station sends a frame whenever it has a frame to send. Since there is only one channel to share, there is the possibility of collision between frames from different stations.

# Pure ALOHA

Following figure shows an example of frame collisions in pure ALOHA.



# Pure ALOHA

- ❖ There are four stations that contend with one another for access to the shared channel.
- ❖ The figure shows that each station sends two frames; there are a total of eight frames on the shared medium. Some of these frames collide because multiple frames are in contention for the shared channel.
- ❖ Figure shows only two frames survive: frame 1.1 from station 1 and frame 3.2 from station 3.

# Pure ALOHA

- ❖ The pure ALOHA protocol relies on acknowledgments from the receiver. When a station sends a frame, it expects the receiver to send an acknowledgment. If the acknowledgment does not arrive after a time-out period, the station assumes that the frame (or the acknowledgment) has been destroyed and resends the frame.
- ❖ A collision involves two or more stations. If all these stations try to resend their frames after the time-out, the frames will collide again. Pure ALOHA dictates that when the time-out period passes, each station waits a random amount of time before resending its frame. The randomness will help avoid more collisions. We call this time the back-off time  $T_B$ .
- ❖ After a maximum number of retransmission attempts  $K_{max}$  a station must give up and try later.

# Pure ALOHA

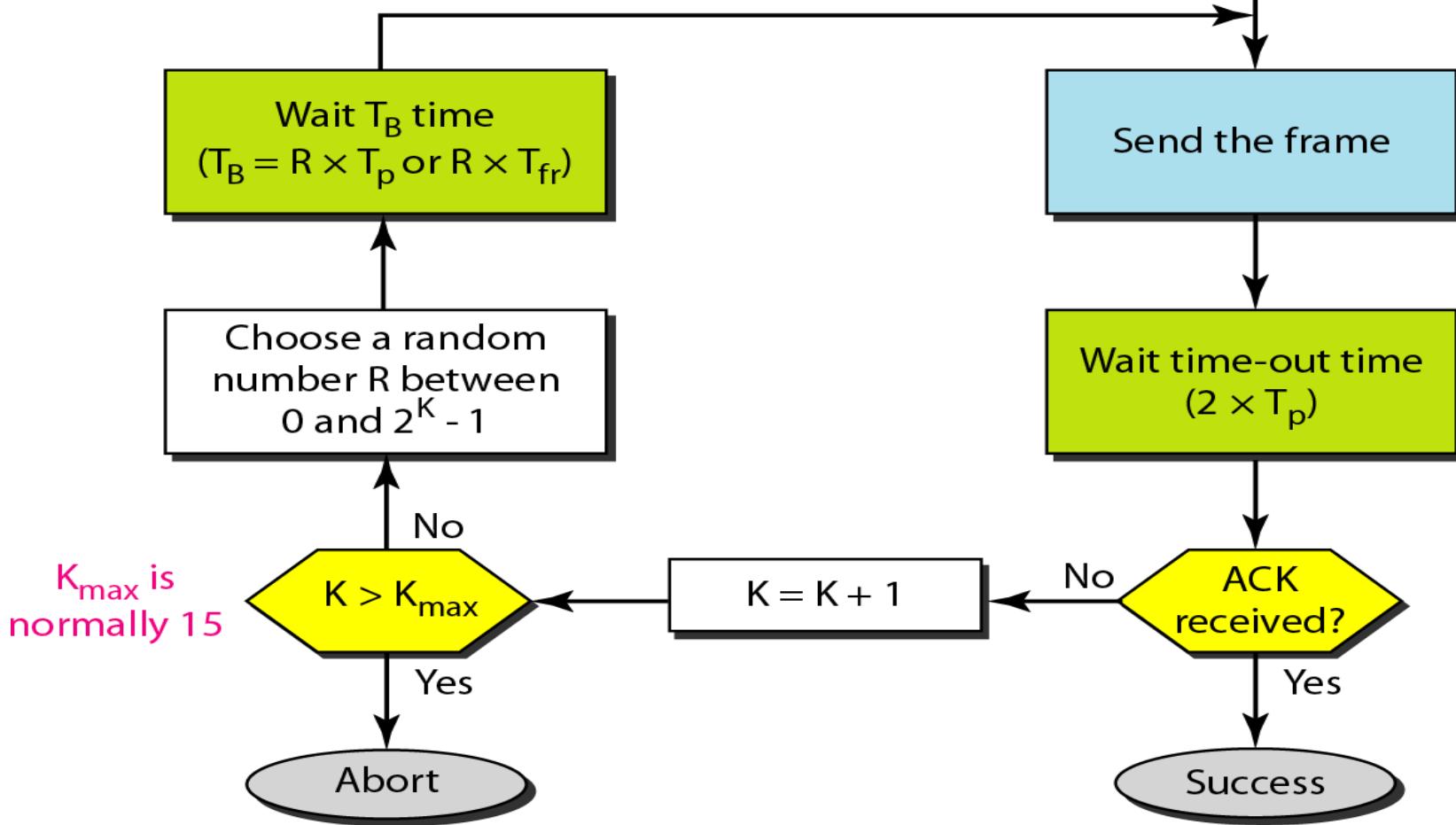
K: Number of attempts

$T_p$ : Maximum propagation time

$T_{fr}$ : Average transmission time for a frame

$T_B$ : Back-off time

Station has  
a frame to send



# Pure ALOHA

- ❖ The time-out period is equal to the maximum possible round-trip propagation delay i.e.

$$t_{\text{out}} = 2T_p$$

Where  $T_p$  is the propagation time between two most widely separated stations.

- ❖ The back-off time  $T_B$  is random value that normally depends on  $K$ , where  $K$  is the number of attempted unsuccessful transmissions.

- ❖  $T_B$  is calculated by **binary exponential back-off** algorithm.

According to this algorithm, for each retransmission, a multiplier in the range 0 to  $2^K - 1$  is randomly chosen and multiplied by  $T_p$  (maximum propagation time) or  $T_{\text{fr}}$  (the average time required to send out a frame) to find  $T_B$ .

- ❖ In this procedure, the range of the random numbers increases after each collision.
- ❖ The value of  $K_{\text{max}}$  is usually chosen as 15.

# Pure ALOHA

**Example:** The stations on a wireless ALOHA network are a maximum of 600 km apart. Signals propagate with speed at  $3 \times 10^8$  m/s. Find the value of  $T_B$  for different value of K.

**Solution:**

$$T_p = (600 \times 10^3) / (3 \times 10^8) = 2 \text{ ms.}$$

Now we can find the value of  $T_B$  for different values of K.

(a) For K = 1, the range is {0, 1}. The station needs to generate a random number with a value of 0 or 1. This means that  $T_B$  is either  $0 \times 2 = 0\text{ms}$  or  $1 \times 2 = 2\text{ms}$ , based on the outcome of the random variable.

(b) For K = 2, the range is {0, 1, 2, 3}. This means that  $T_B$  can be 0, 2, 4, or 6 ms, based on the outcome of the random variable.

(c) For K = 3, the range is to {0, 1, 2, 3, 4, 5, 6, 7}. This means that  $T_B$  can be 0, 2, 4, ..., 14 ms, based on the outcome of the random variable.

# Pure ALOHA

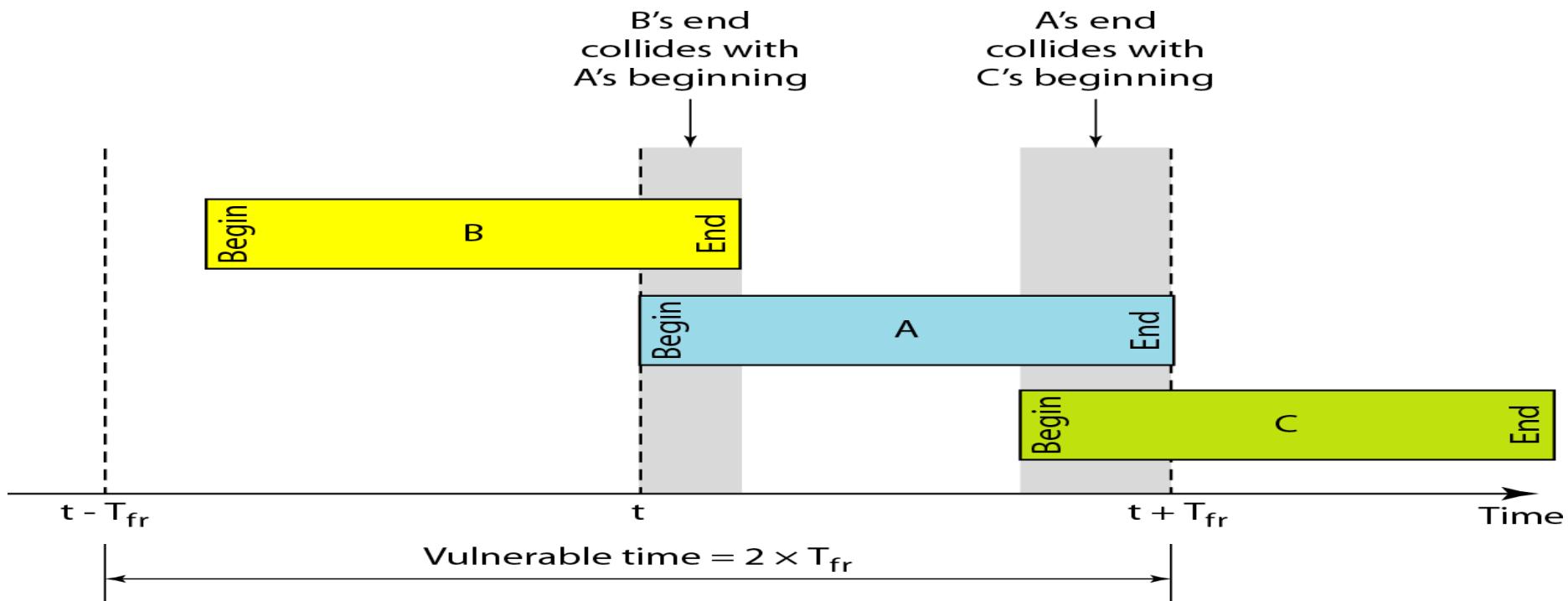
## Vulnerable time

**Vulnerable time is the length of time, in which there is a possibility of collision.**

We assume that the stations send fixed-length frames with each frame taking  $T_{fr}$  second to send.

This figure shows the vulnerable time for station A.

**Pure ALOHA vulnerable time =  $2 \times T_{fr}$**



# Pure ALOHA

## Throughput

Let  $G$  is the average number of frames generated by the system during one frame transmission time.

Average number of successful transmissions for pure ALOHA i.e. throughput,  $S = G \times e^{-2G}$ .

The maximum throughput

$$S_{\max} = 0.184, \text{ for } G = 1/2.$$

# Pure ALOHA

## Example:

A pure ALOHA network transmits 200-bit frames on a shared channel of 200 kbps. What is the requirement to make this frame collision-free?

## Example:

A pure ALOHA network transmits 200-bit frames on a shared channel of 200 kbps. What is the throughput if the system (all stations together) produces

- a. 1000 frames per second
- b. 500 frames per second
- c. 250 frames per second

# Pure ALOHA

**Solution:**

$$\begin{aligned}\text{Average frame transmission time } T_{fr} &= 200 \text{ bits}/200 \text{ kbps} \\ &= 1 \text{ ms.}\end{aligned}$$

Therefore, the vulnerable time =  $2 \times T_{fr} = 2 \text{ ms}$

This means no station should send later than 1 ms before this station starts transmission and no station should start sending during the 1ms period that this station is sending.

# Pure ALOHA

**Solution:**

Here,  $T_{fr} = 200/(200*10^{-3}) = 1 \text{ ms}$

(a) Throughput  $S = G * e^{-2G}$

Here,  $G$  is the average number of frames generated in frame transmission time.

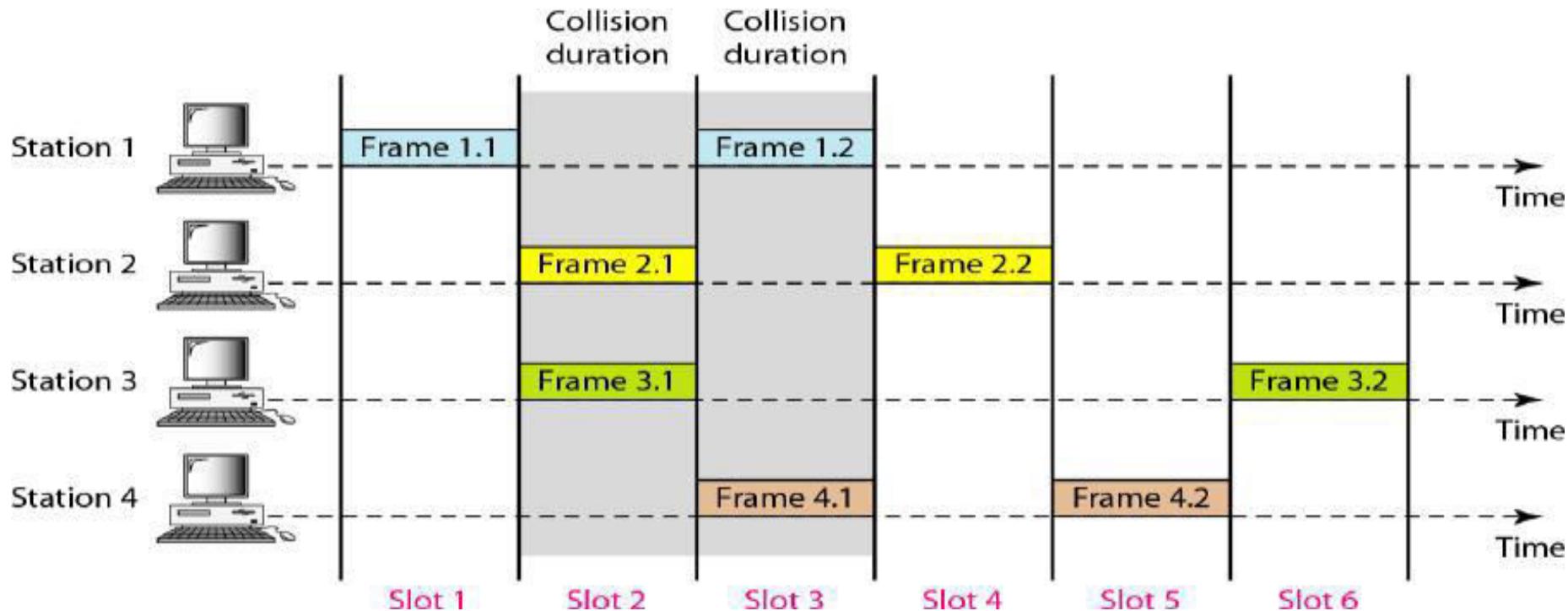
$G = 1000/1000 = 1 \text{ frame}$

$S = 1 * e^{-2*1} = e^{-2} = 0.135 \text{ (13.5 percent)}$

This means that the throughput is  $1000 \times 0.135 = 135 \text{ frames}$ . Only 135 frames out of 1000 will probably survive.

# Slotted ALOHA

- ❖ Slotted ALOHA was invented to improve the efficiency of pure ALOHA.
- ❖ In slotted ALOHA we divide the time into slots of  $T_{fr}$  and force the station to send only at the beginning of the time slot.
- ❖ Figure shows an example of frame collisions in slotted ALOHA.



# Slotted ALOHA

- ❖ Since a station is allowed to send only at the beginning of the synchronized time slot, if a station misses this moment, it must wait until the beginning of the next time slot.
- ❖ There is still the possibility of collision if two stations try to send at the beginning of the same time slot.
- ❖ The vulnerable time is now reduced to one-half, equal to  $T_{fr}$ .
- ❖ Throughput,  $S = G * e^{-G}$
- ❖ Maximum throughput  $S_{max} = 0.368$ , when  $G = 1$ .

# Slotted ALOHA

## Example:

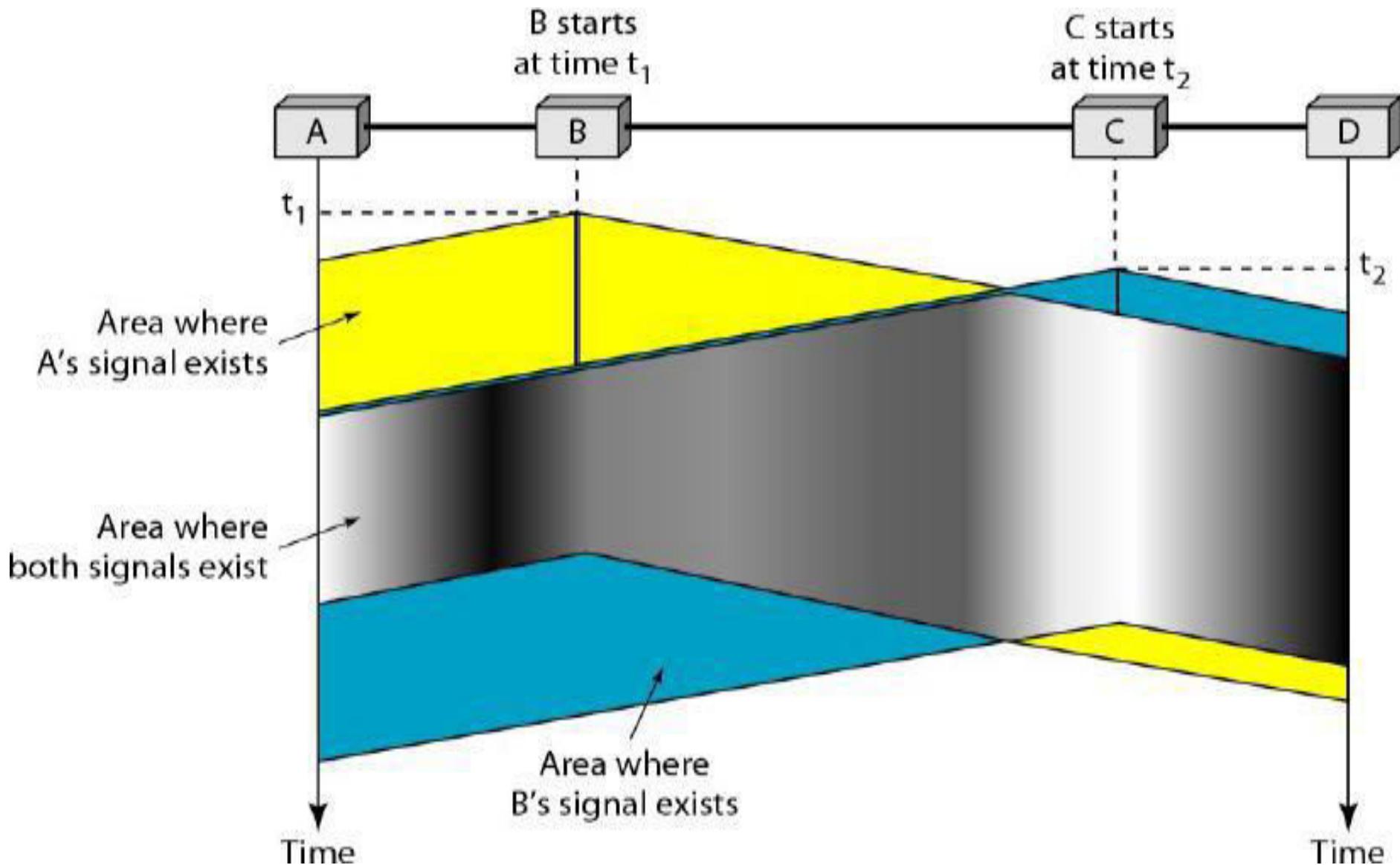
A slotted ALOHA network transmits 200-bit frames on a shared channel of 200 kbps. What is the throughput if the system (all stations together) produces

- a. 1000 frames per second
- b. 500 frames per second
- c. 250 frames per second

# Carrier Sense Multiple Access (CSMA)

- ❖ To minimize the chance of collision and, therefore, increase the performance, the CSMA method was developed.
- ❖ Carrier sense multiple access (CSMA) requires that each station first listen to the medium (or check the state of the medium) before sending. In other words, CSMA is based on the principle "sense before transmit" or "listen before talk."
- ❖ CSMA can reduce the possibility of collision, but it cannot eliminate it.
- ❖ The possibility of collision still exists because of propagation delay; when a station sends a frame, it still takes time for the first bit to reach every station and for every station to sense it.
- ❖ In other words, a station may sense the medium and find it idle, only because the first bit sent by another station has not yet been received.

# Carrier Sense Multiple Access (CSMA)



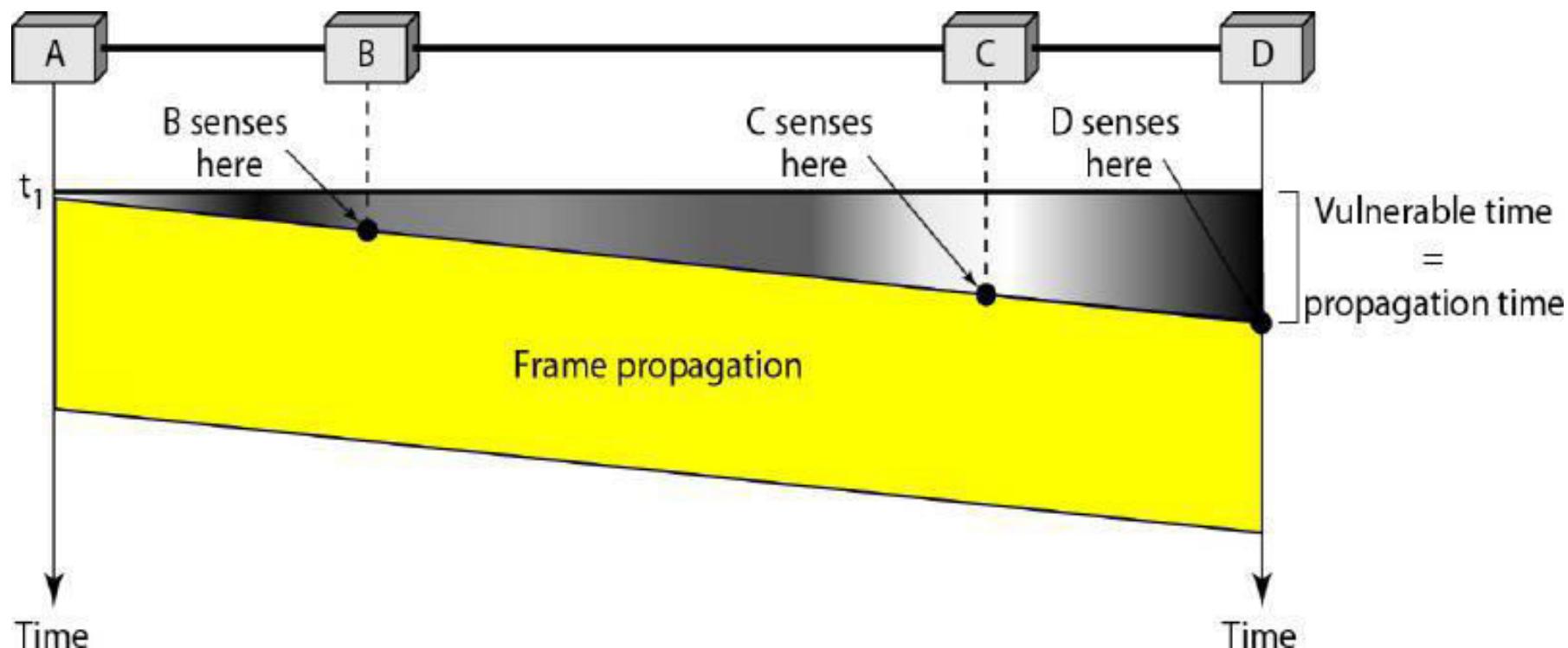
# Carrier Sense Multiple Access (CSMA)

At time  $t_1$  station B senses the medium and finds it idle, so it sends a frame. At time  $t_2$  ( $t_2 > t_1$ ) station C senses the medium and finds it idle because, at this time, the first bits from station B have not reached station C. Station C also sends a frame. The two signals collide and both frames are destroyed.

# Carrier Sense Multiple Access (CSMA)

## Vulnerable Time

- ❖ The vulnerable time for CSMA is the **propagation time  $T_p$** .
- ❖ This is the time needed for a signal to propagate from one end of the medium to the other.



# Carrier Sense Multiple Access (CSMA)

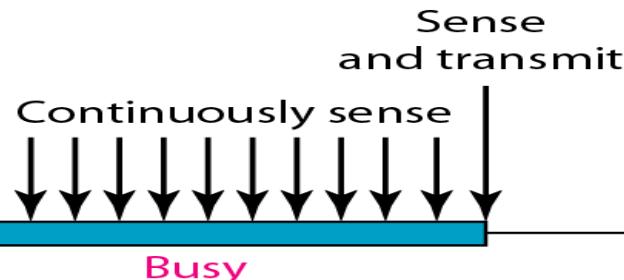
## Persistence Methods

- What should a station do if the channel is busy?
- What should a station do if the channel is idle?

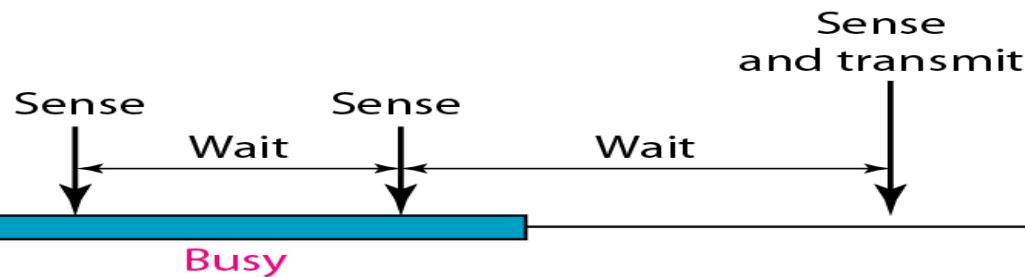
There are three methods to answer these questions:

1. 1-persistent method
2. Non-persistent method
3. p-persistent method

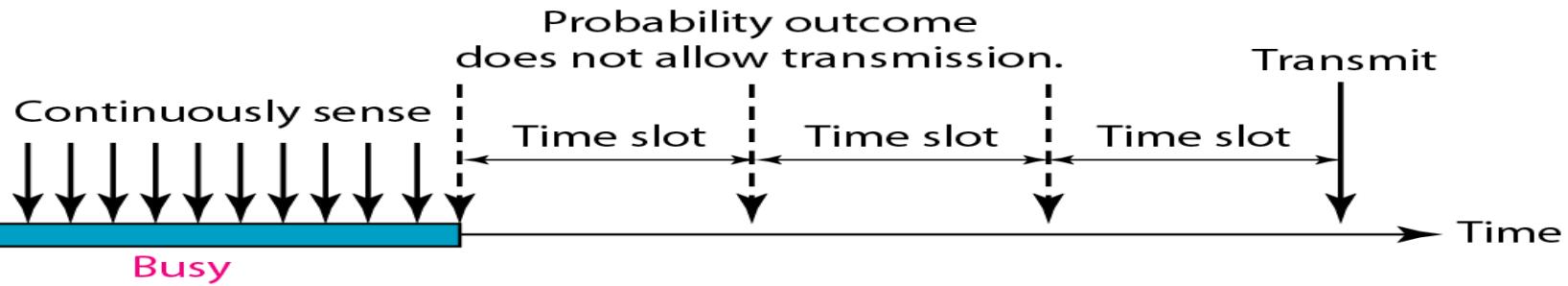
# Carrier Sense Multiple Access (CSMA)



a. 1-persistent



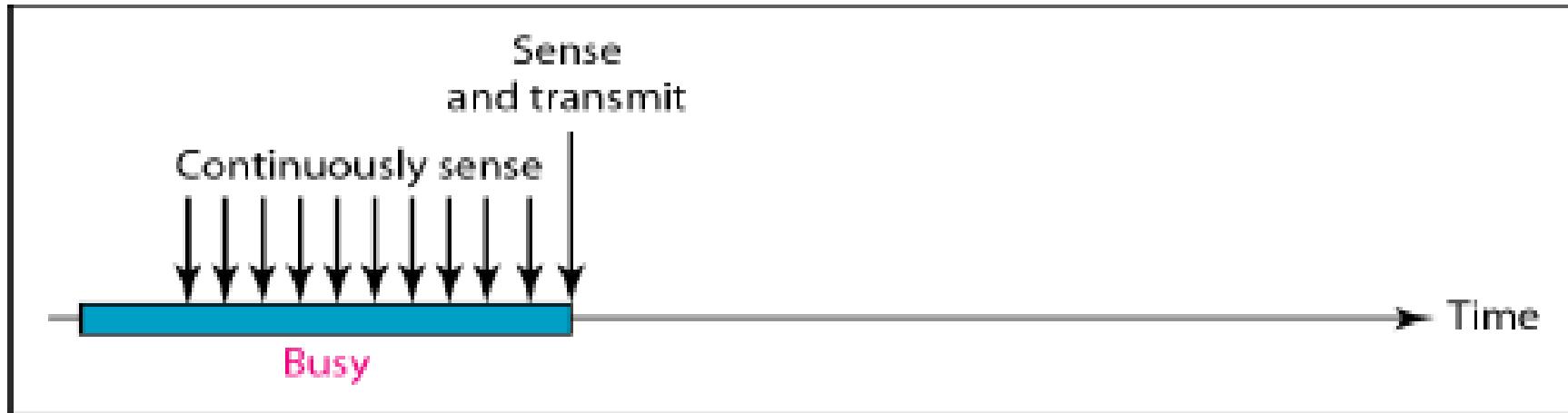
b. Nonpersistent



c. p-persistent

# Carrier Sense Multiple Access (CSMA)

## 1-Persistent

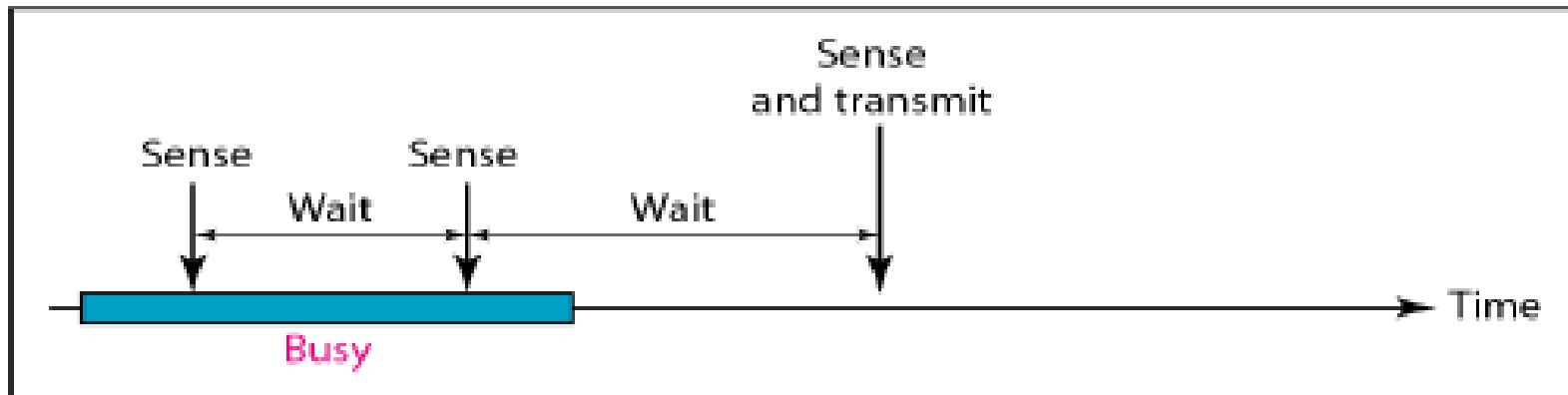


a. 1-persistent

- ❖ In this method, after the station finds the line idle, it sends its frame immediately (with probability 1).
- ❖ This method has the highest chance of collision because two or more stations may find the line idle and send their frames immediately.

# Carrier Sense Multiple Access (CSMA)

## Non-persistent

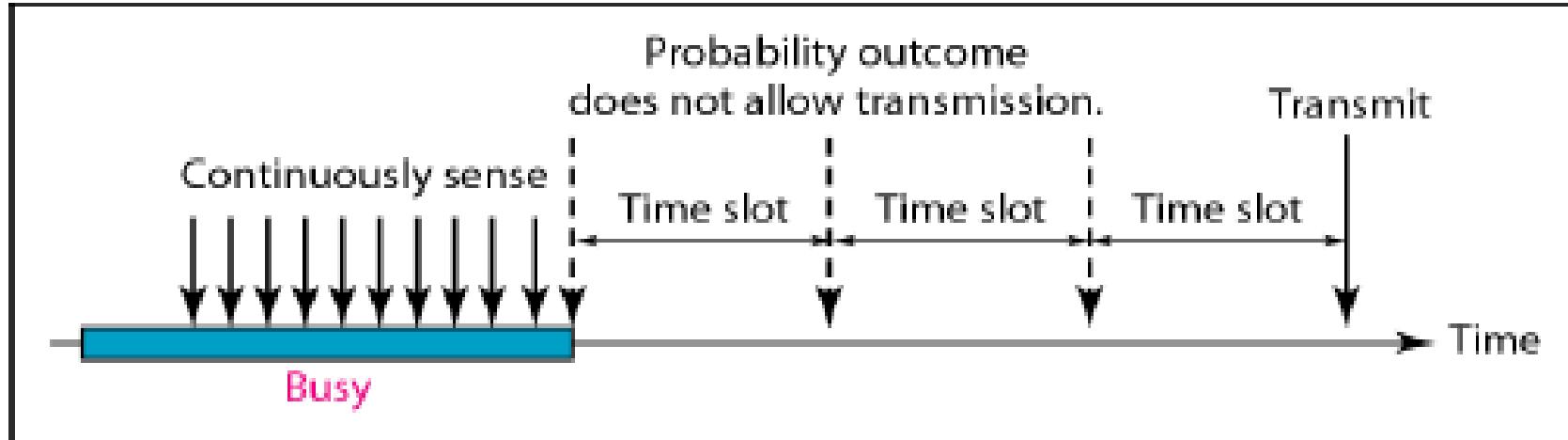


b. Nonpersistent

- ❖ In the non-persistent method, a station that has a frame to send senses the line. If the line is idle, it sends immediately. If the line is not idle, it waits a random amount of time and then senses the line again.
- ❖ The non-persistent approach reduces the chance of collision because it is unlikely that two or more stations will wait the same amount of time and retry to send simultaneously. However, this method reduces the efficiency of the network because the medium remains idle when there may be stations with frames to send.

# Carrier Sense Multiple Access (CSMA)

## p-Persistent



c. p-persistent

- ❖ The p-persistent method is used if the channel has time slots with a slot duration equal to or greater than the maximum propagation time.
- ❖ The p-persistent approach combines the advantages of the other two strategies.
- ❖ It reduces the chance of collision and improves efficiency.

# Carrier Sense Multiple Access (CSMA)

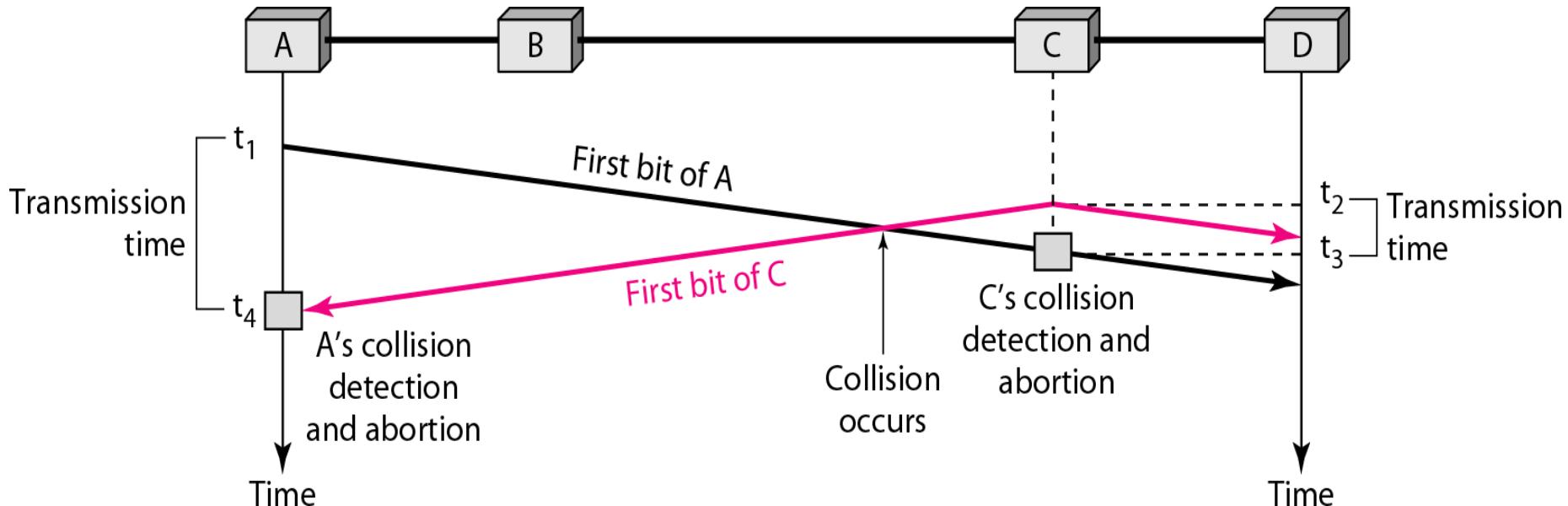
In this method, after the station finds the line idle it follows these steps:

1. With probability  $p$ , the station sends its frame.
2. With probability  $q = 1 - p$ , the station waits for the beginning of the next time slot and checks the line again.
  - a. If the line is idle, it goes to step 1.
  - b. If the line is busy, it acts as though a collision has occurred and uses the backoff procedure.

# **Carrier Sense Multiple Access with Collision Detection (CSMA/CD)**

- ❖ The CSMA method does not specify the procedure following a collision.
- ❖ CSMA/CD augments the algorithm to handle the collision.
- ❖ In this method, a station monitors the medium after it sends a frame to see if the transmission was successful. If so, the station is finished. If, however, there is a collision, the frame is sent again.

# Carrier Sense Multiple Access with Collision Detection (CSMA/CD)



- ❖ In this figure, stations A and C are involved in the collision.
- ❖ At time  $t_1$ , station A has executed its persistence procedure and starts sending the bits of its frame. At time  $t_2$ , station C has not yet sensed the first bit sent by A. Station C executes its persistence procedure and starts sending the bits in its frame, which propagate both to the left and to the right. The collision occurs sometime after time  $t_2$ . Station C detects a collision at time  $t_3$  when it receives the first bit of A's frame. Station C immediately aborts transmission. Station A detects collision at time  $t_4$  when it receives the first bit of C's frame; it also immediately aborts transmission.

# Carrier Sense Multiple Access with Collision Detection (CSMA/CD)

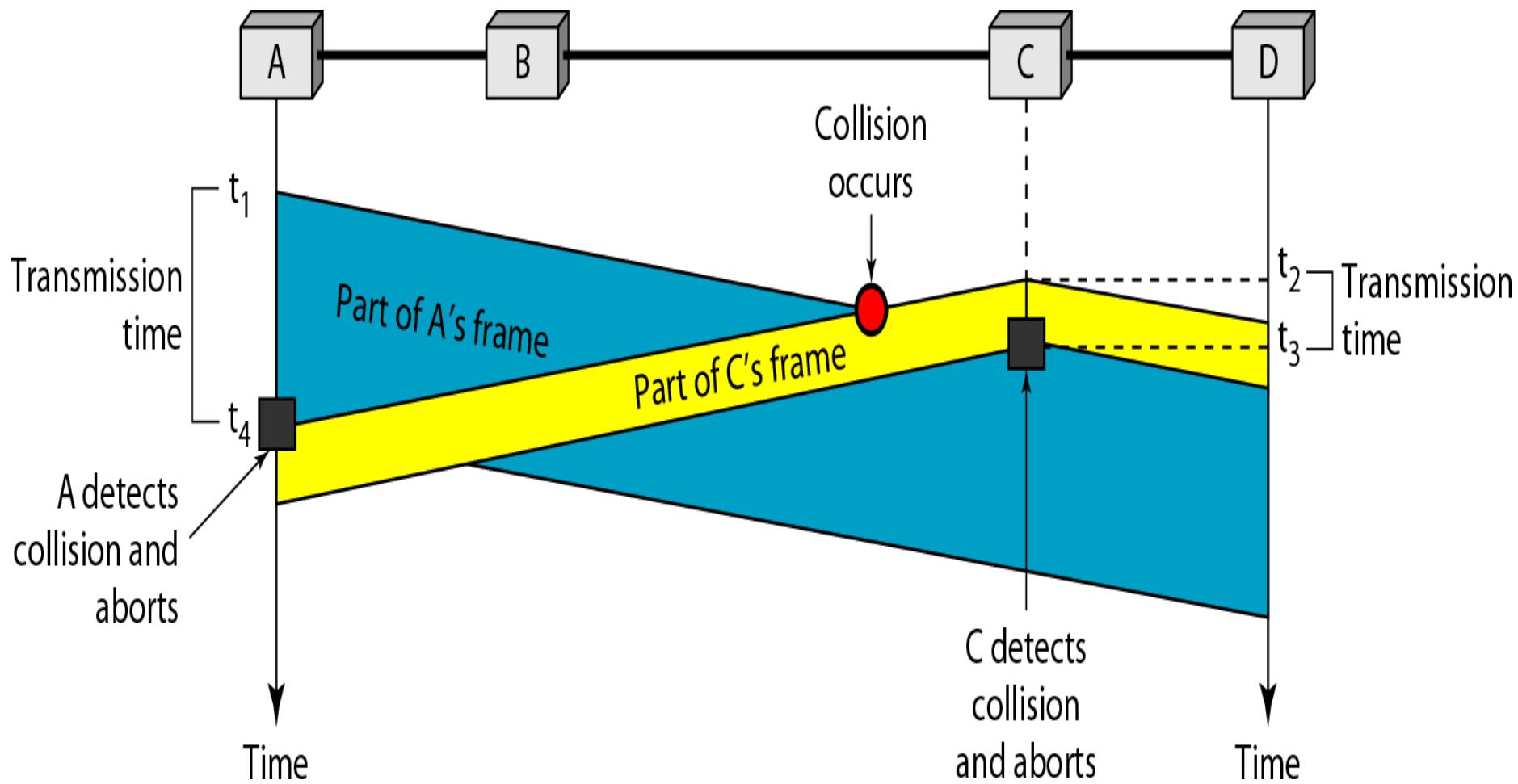
## Minimum Frame Size

- ❖ For CSMA/CD to work, we need a restriction on the frame size.
- ❖ Before sending the last bit of the frame, the sending station must detect a collision, if any, and abort the transmission.
- ❖ This is so because the station, once the entire frame is sent, does not keep a copy of the frame and does not monitor the line for collision detection.
- ❖ Therefore, the frame transmission time  $T_{fr}$  must be at least two times the maximum propagation time  $T_p$  i.e.

$$T_{fr} \geq 2T_p$$

# Carrier Sense Multiple Access with Collision Detection (CSMA/CD)

## Minimum Frame Size



# Carrier Sense Multiple Access with Collision Detection (CSMA/CD)

## Example:

A network using CSMA/CD has a bandwidth of 10 Mbps. If the maximum propagation time is 25.6  $\mu$ s, what is the minimum size of the frame?

## Solution:

Let the frame size is x.

In CSMA/CD, we know that  $T_{fr} \geq 2T_p$ ,

Therefore,  $x/(10*10^6) \geq 2*25.6*10^{-6}$

$$x \geq 2*10*25.6*10^6 * 10^{-6} = 512$$

Therefore, minimum size of the frame = 512 bits

# Carrier Sense Multiple Access with Collision Detection (CSMA/CD)

## Throughput

- ❖ The throughput of CSMA/CD is greater than that of pure or slotted ALOHA.
- ❖ The maximum throughput occurs at a different value of G and is based on the persistence method and the value of p in the p-persistent approach.
- ❖ For 1-persistent method the maximum throughput is around 50 percent when G=1.
- ❖ For non-persistent method, the maximum throughput can go up to 90 percent when G is between 3 and 8.

# Carrier Sense Multiple Access with Collision Detection (CSMA/CD)

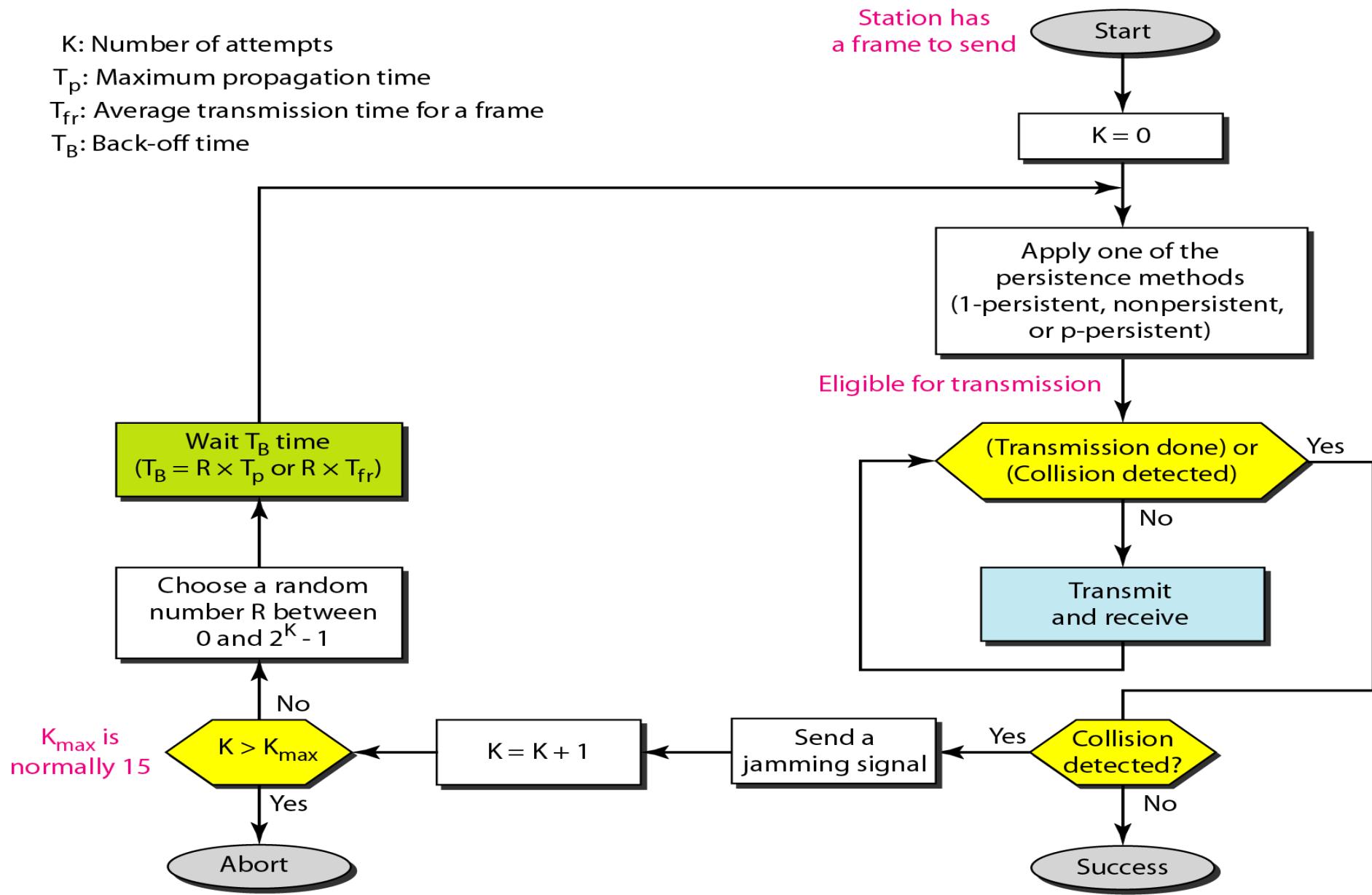
K: Number of attempts

$T_p$ : Maximum propagation time

$T_{fr}$ : Average transmission time for a frame

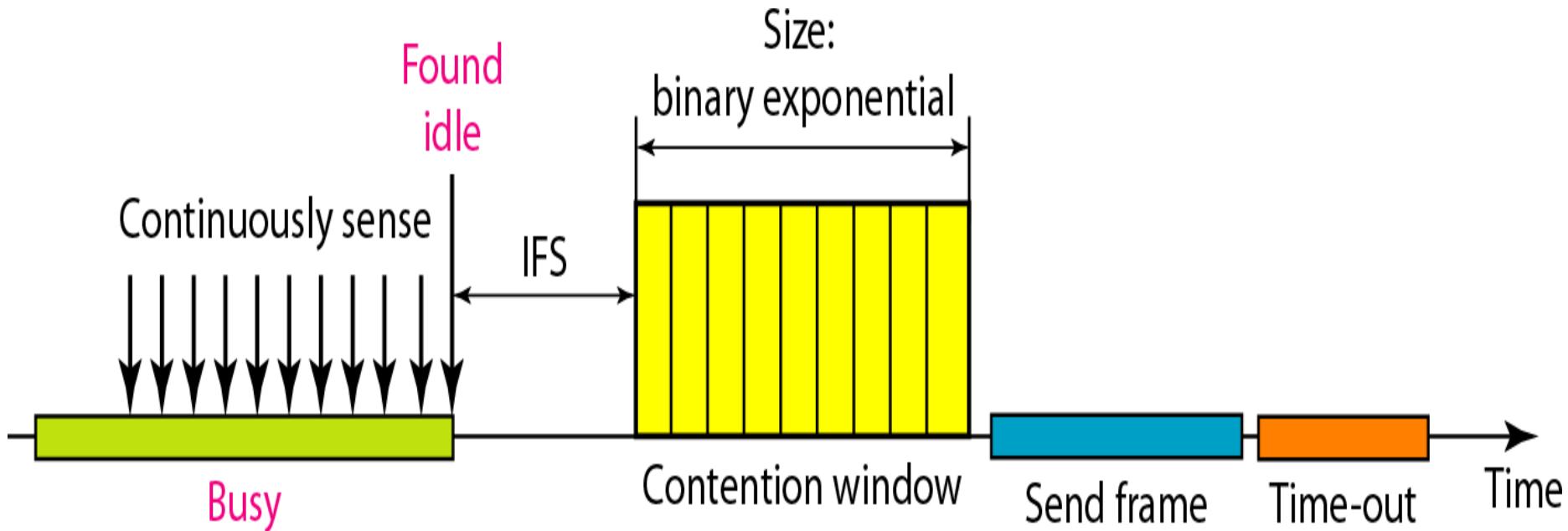
$T_B$ : Back-off time

Station has  
a frame to send



# Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA)

- ❖ CSMA/CA was invented for wireless network.
- ❖ Collisions are avoided through the use of CSMA/CA's three strategies: the interframe space, the contention window, and acknowledgments.



# **Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA)**

## **Interframe Space (IFS)**

When an idle channel is found, the station does not send immediately. It waits for a period of time called the interframe space or IFS.

If after the IFS time the channel is still idle, the station can send, but it still needs to wait a time equal to the contention time.

In CSMA/CA, the IFS variable can also be used to define the priority of stations or frames.

# Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA)

## Contention Window

- ❖ The contention window is an amount of time divided into slots. A station that is ready to send chooses a random number of slots as its wait time. The number of slots in the window changes according to the binary exponential back-off strategy.
- ❖ The station senses the channel after each time slot. However, if the station finds the channel busy, it does not restart the process; it just stops the timer and restarts it when the channel is sensed as idle.

# **Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA)**

## **Acknowledgment**

With all these precautions, there still may be a collision resulting in destroyed data.

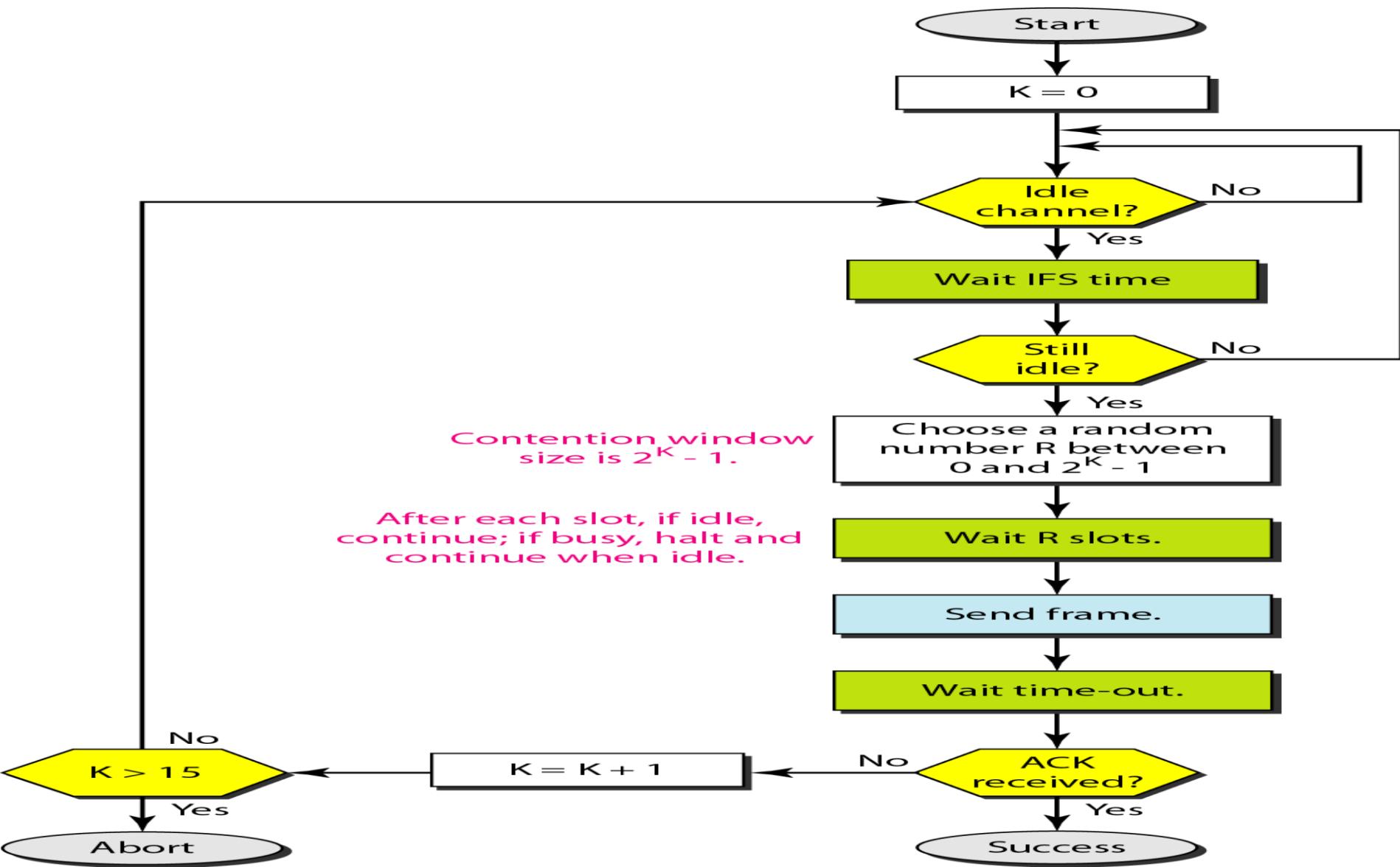
In addition, the data may be corrupted during the transmission. The positive acknowledgment and the time-out timer can help guarantee that the receiver has received the frame.

# Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA)

## Procedure

- ❖ The channel needs to be sensed before and after the IFS.
- ❖ The channel also needs to be sensed during the contention time.
- ❖ For each time slot of the contention window, the channel is sensed. If it is found idle, the timer continues; if the channel is found busy, the timer is stopped and continues after the timer becomes idle again.

# Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA)

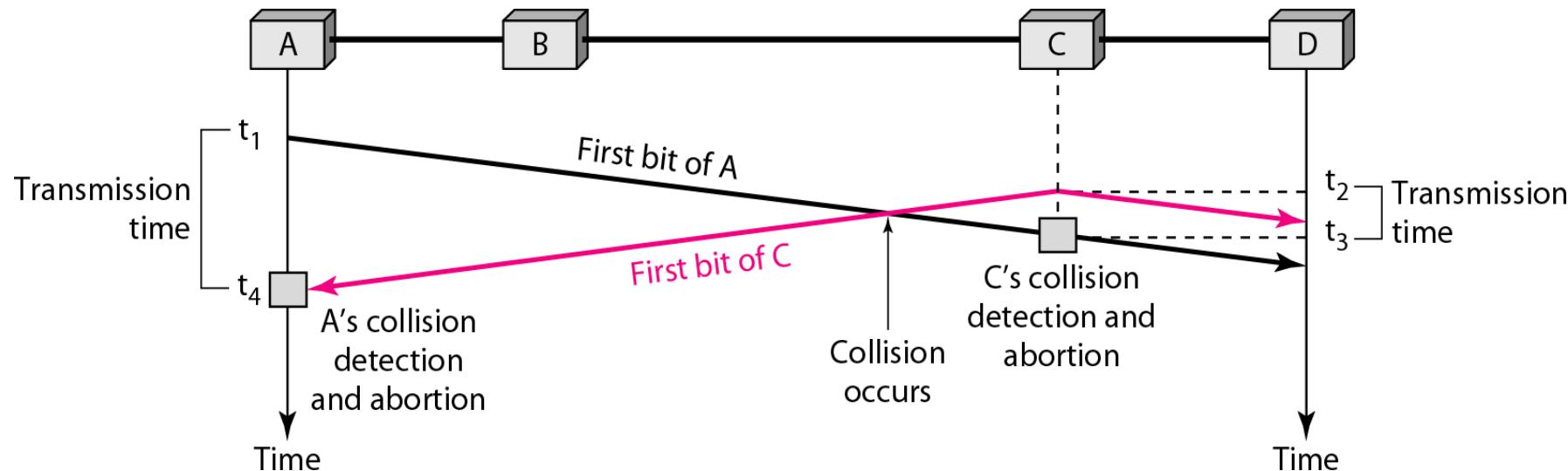


## Exercise

1. In a CSMA/CD network with a data rate of 10 Mbps, the minimum frame size is found to be 512 bits for the correct operation of the collision detection process. What should be the minimum frame size if we increase the data rate to 100 Mbps? To 1 Gbps? To 10 Gbps?
2. One hundred stations on a pure ALOHA network share a 1-Mbps channel. If frames are 1000 bits long, find the throughput if each station is sending 10 frames per second.

# Exercise

3.



The data rate is 10 Mbps, the distance between station A and C is 2000 m, and the propagation speed is  $2 \times 10^8$  m/s. Station A starts sending a long frame at time  $t_1 = 0$ ; station C starts sending a long frame at time  $t_2 = 3\mu s$ . The size of the frame is long enough to guarantee the detection of collision by both stations. Find:

- The time when station C hears the collision ( $t_3$ ).
- The time when station A hears the collision ( $t_4$ ).
- The number of bits station A has sent before detecting the collision.
- The number of bits station C has sent before detecting the collision.

# IEEE 802 STANDARDS

- ❖ IEEE 802 is a collection of networking standards that cover the physical and data-link layer specifications for technologies such as Ethernet and wireless. These specifications apply to local area networks (LAN) and metropolitan area networks (MAN).
- ❖ IEEE stands for Institute of Electrical and Electronics Engineers.

## Ethernet(802.3 standard)

It uses CSMA/CD protocol to access the medium.

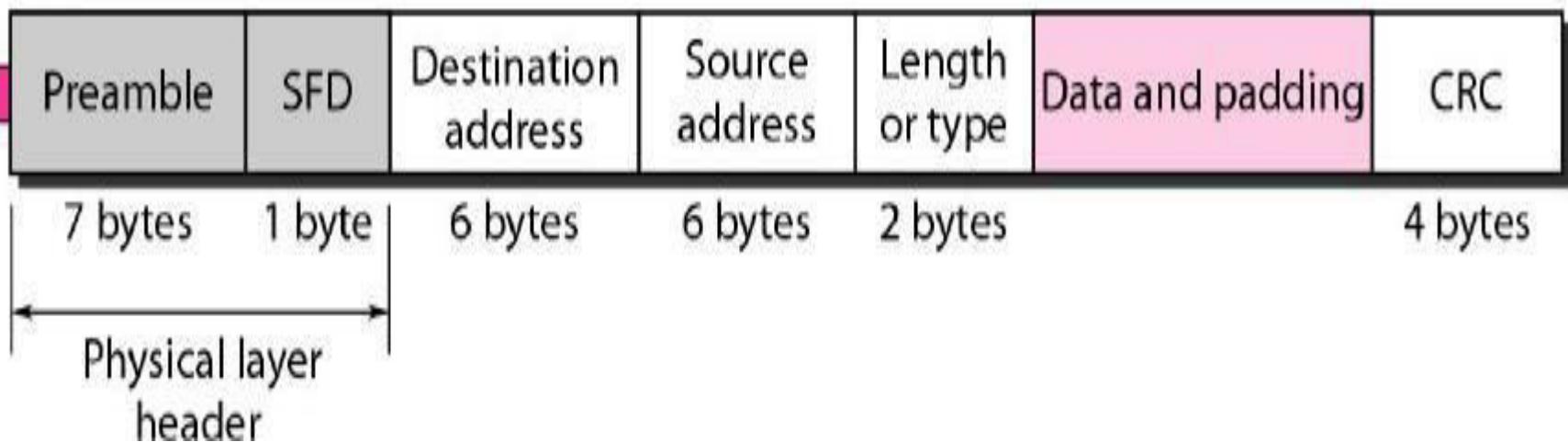
### Frame Format

The Ethernet frame contains seven fields: preamble, SFD, DA, SA, length or type of protocol data unit (PDU), upper-layer data, and the CRC.

# IEEE 802 STANDARDS

Preamble: 56 bits of alternating 1s and 0s.

SFD: Start frame delimiter, flag (10101011)



## Preamble

This is the first field. It contains 7-bytes of alternating 0's and 1's that alerts the receiving system to the coming frame and enables it to synchronize its input timing. The preamble is actually added at the physical layer and is not part of the frame.

# IEEE 802 STANDARDS

## **Start frame delimiter (SFD)**

The second field (1 byte: 10101011) signals the beginning of the frame. The SFD warns the station or stations that this is the last chance for synchronization. The last 2 bits is 11 and alerts the receiver that the next field is the destination address.

## **Destination address (DA)**

The DA field is 6 bytes and contains the physical address of the destination station or stations to receive the packet.

## **Source address (SA)**

The SA field is also 6 bytes and contains the physical address of the sender of the packet.

# IEEE 802 STANDARDS

## Length or type

This field is defined as a type field or length field. The original Ethernet used this field as the type field to define the upper-layer protocol using the MAC frame. The IEEE standard used it as the length field to define the number of bytes in the data field.

## Data

This field carries data encapsulated from the upper-layer protocols. It is a minimum of 46 and a maximum of 1500 bytes.

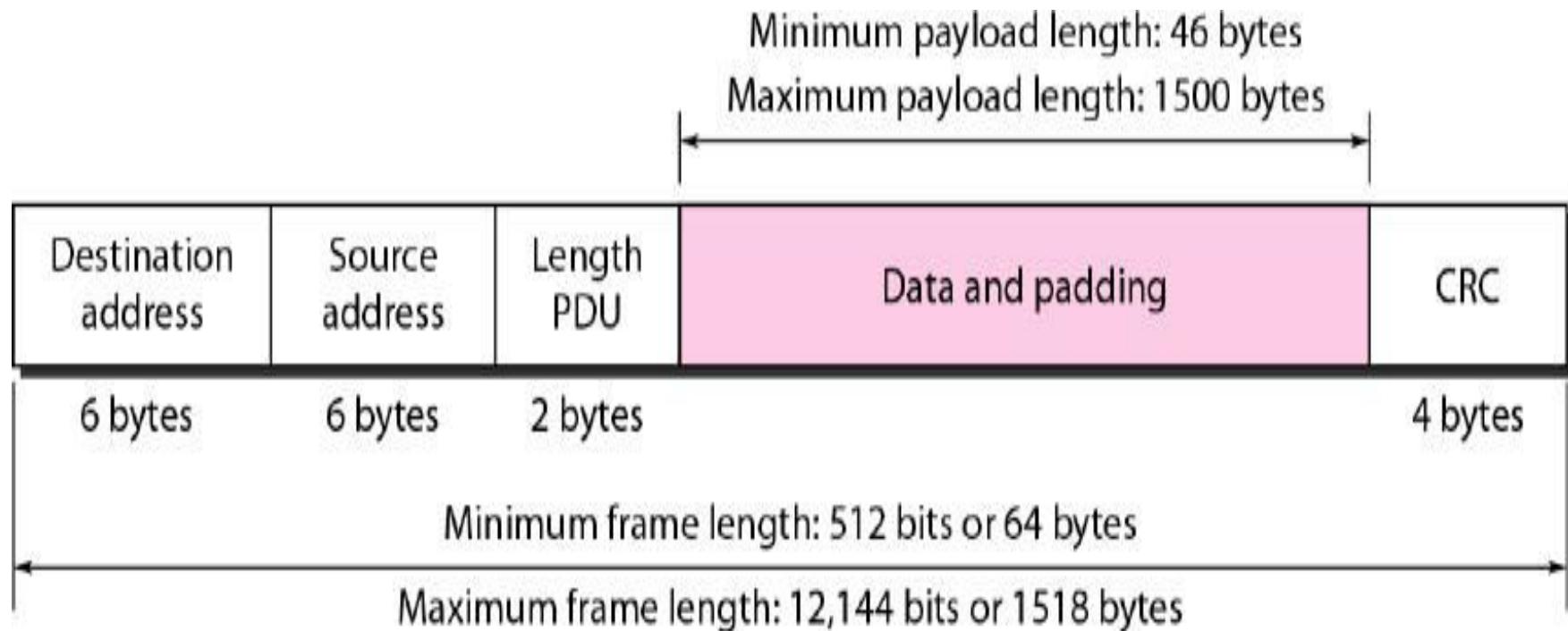
## CRC

The last field contains error detection information. It is of 4 bytes.

# IEEE 802 STANDARDS

## Frame Length

Ethernet has imposed restrictions on both the minimum and maximum lengths of a frame, as shown in below Figure:-



# IEEE 802 STANDARDS

## Addressing

Each station on an Ethernet network has its own network interface card (NIC). The NIC fits inside the station and provides the station with a 6-byte physical address. It is written in hexadecimal notation, with a colon between the bytes.

06:01:02:01:2C:4B

## Unicast, Multicast, and Broadcast Addresses

A source address is always a unicast address.

The destination address can be unicast, multicast, or broadcast.

If the least significant bit of the first byte in a destination address is 0, then the address is unicast; otherwise, it is multicast.

The broadcast address is a special case of the multicast address. In this case, the recipients are all the stations on the LAN. A broadcast destination address is forty eight 1's.

# IEEE 802 STANDARDS

## Example

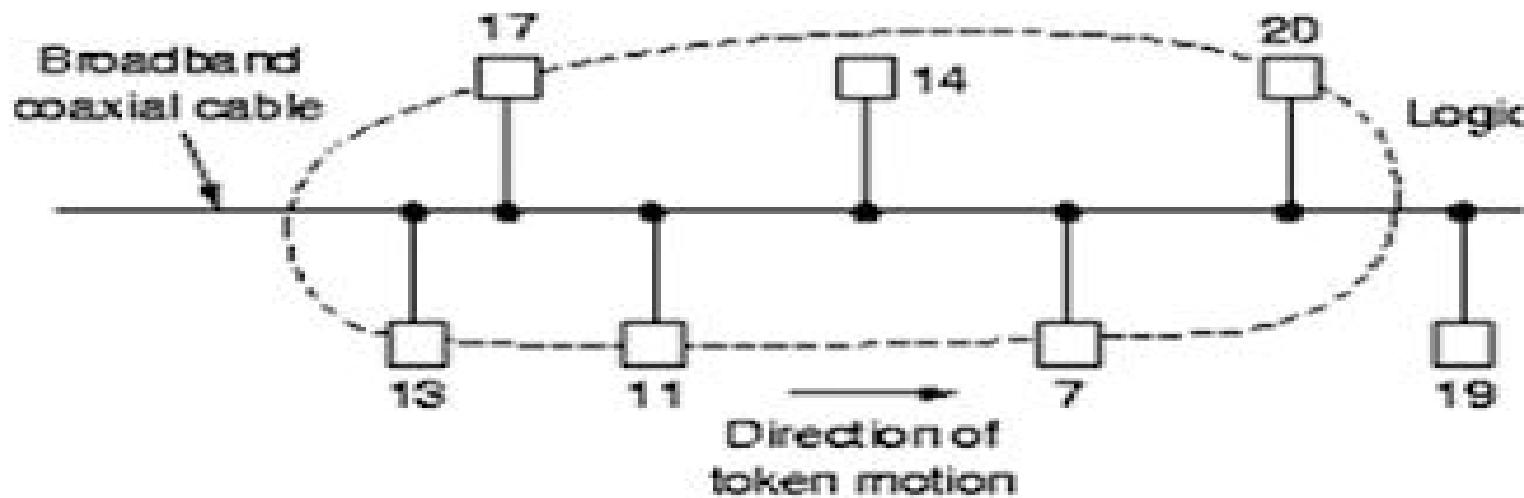
Define the type of the following destination addresses:

- a) 4A:30:10:21:10:1A
- b) 47:20:1B:2E:08:EE
- c) FF:FF:FF:FF:FF:FF

# IEEE 802 STANDARDS

## IEEE 802.4 (Token Bus)

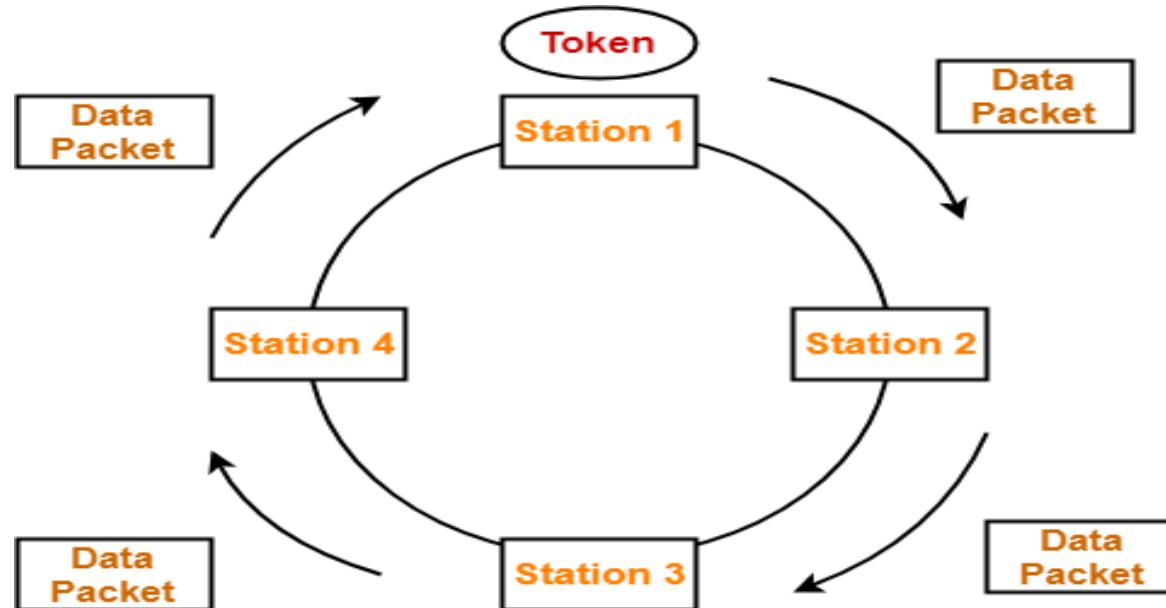
- ❖ IEEE specifications include physical layer and media access control sublayer for network that uses a bus topology and use token passing as the media access method.
- ❖ In this, all nodes are connected in a logical ring. It supports electrical(coaxial) and fiber optic cable.



# IEEE 802 STANDARDS

## IEEE 802.5 (Token Ring)

- ❖ A token ring network consists of a set of nodes connected in a ring. Data always flows in a particular direction around the ring.
- ❖ It uses token passing as the media access method.



Delayed Token Reinsertion Token Passing

# IEEE 802 STANDARDS

## Fiber Distributed Data Interface(FDDI)

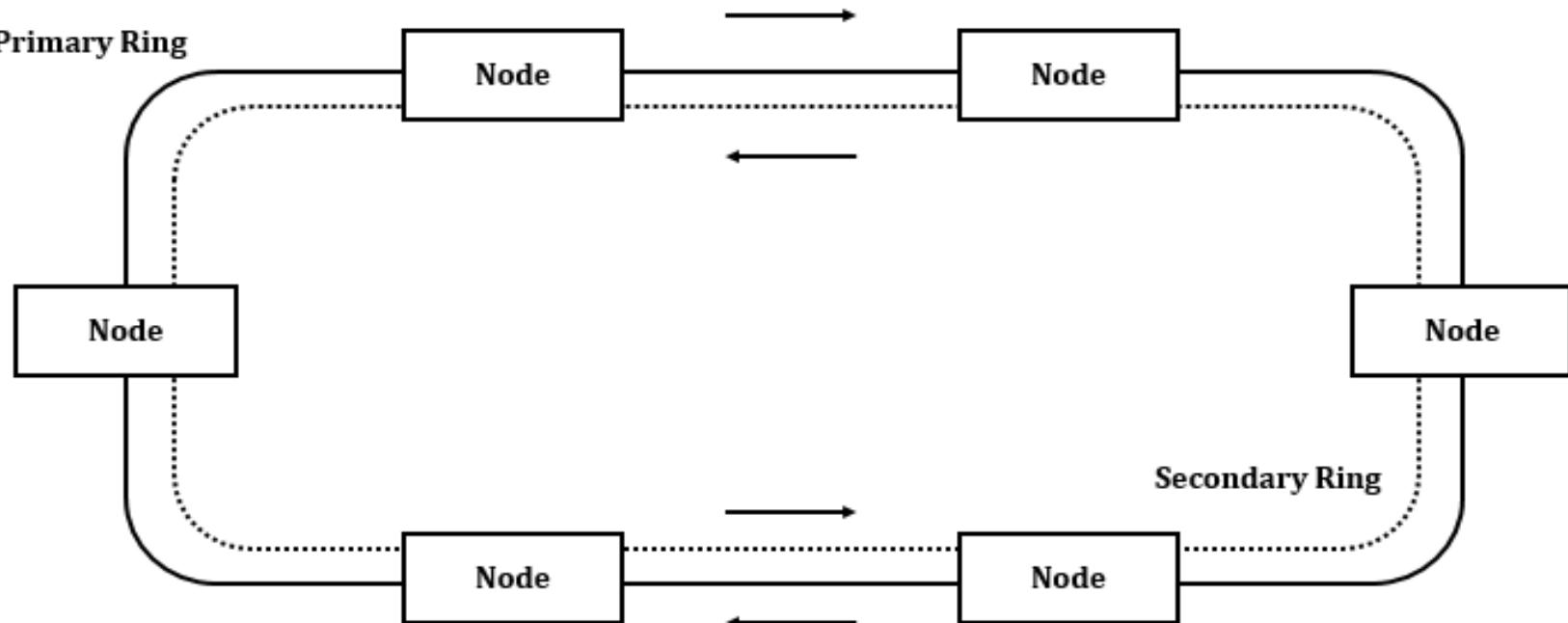
Fiber Distributed Data Interface (FDDI) is a set of ANSI and ISO standards for transmission of data in local area network (LAN) over fiber optic cables. It is applicable in large LANs that can extend up to 200 kilometers in diameter.

### Features

- FDDI uses optical fiber as its physical medium.
- It operates in the physical and medium access control (MAC layer) of the Open Systems Interconnection (OSI) network model.
- It provides high data rate of 100 Mbps and can support thousands of users.
- It is used in LANs up to 200 kilometers for long distance voice and multimedia communication.

# IEEE 802 STANDARDS

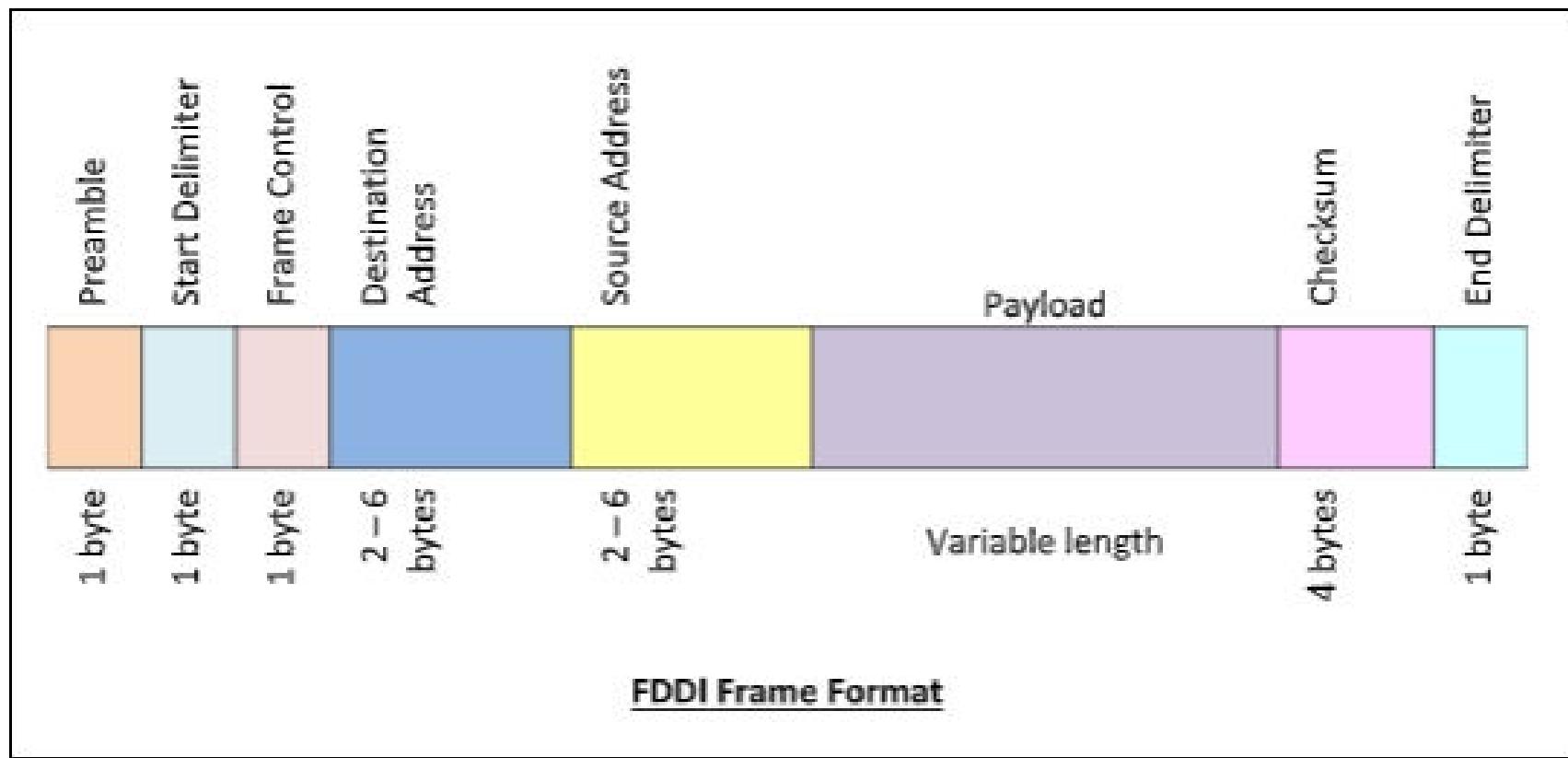
- It uses ring based token passing mechanism.
- It contains two token rings, a primary ring for data and token transmission and a secondary ring that provides backup if the primary ring fails.
- FDDI technology can also be used as a backbone for a wide area network (WAN).



# IEEE 802 STANDARDS

## Frame Format

The frame format of FDDI is similar to that of token bus as shown in the following diagram –



# **IEEE 802 STANDARDS**

# AKTU Examination Questions

1. A bit string 000111111001111101000 needs to be transmitted at the data link layer. What is the string actually transmitted after bit stuffing?
2. Explain the working of pure ALOHA and slotted ALOHA protocols. How slotted ALOHA improve the performance of pure ALOHA?
3. List different carrier sense protocols. How CSMA/CD protocol is different from other CSMA/CA protocol?
4. What is piggybacking?
5. Measurement of slotted ALOHA channel with infinite number of users such that the 10 percent of slots are idle.
  - (i) What is the channel load?
  - (ii) What is the throughput?

# AKTU Examination Questions

6. If a binary signal is sent over a 3KHZ channel. Whose signal to noise ratio is 20db. What is the maximum achievable data rate?
7. Discuss the issues in the data link layer and about its protocol on the basis of layering principle.
8. Discuss different carrier sense protocols. How are they different than collisions protocols?
9. Write short notes on following:
  - i. Stop and Wait ARQ
  - ii. Sliding Window Protocol
  - iii. Go Back N ARQ

# AKTU Examination Questions

10. An ALOHA network uses 9.2 kbps channel for sending message packets of 100 bits long size. Calculate the maximum throughput for pure ALOHA network.
11. What is the total delay (latency) for a frame size of 10 million bits that is being set up on link with 15 routers, each having queuing time of  $2\mu s$  and a processing time of  $1\mu s$ ? The length of link is 3000km. The speed of light inside the link is  $2 \times 10^8$  m/sec. The link has bandwidth of 6 Mbps.
12. What is hamming code? Explain its working with suitable example.
13. What are header and trailers and how do they get added and removed?

# AKTU Examination Questions

14. A large FDDI ring has 100 stations & a token rotation time of 40 msec. The token holding time is 10 msec. What is the maximum achievable efficiency of the ring?
15. A channel has a bit rate of 20 kbps. The stop and wait protocol with frame size 4500 bits is used. The delay for error detection and sending ACK by the receiver is 0.25 seconds because of a fault. Find the maximum efficiency of the channel if the destination is 30000 km away and the speed of the propagation of the signal is  $2.8 \times 10^8$  m/s. Find the decrease in efficiency due to the fault.
16. A slotted ALOHA network transmits 400-bit frames on a shared channel of 400 kbps. What is the throughput if the system (all stations together) produces –
  - (i) 1000 frames per second
  - (ii) 500 frames per second
  - (iii) 250 frames per second

# AKTU Examination Questions

17. Explain ARQ Error Control technique, in brief.
18. Compare ALOHA with slotted ALOHA.
19. State the requirements of CRC.
20. Discuss the issues in the data link layer and about its protocol on the basis of layering principle.
21. Consider the use of 10 K-bit size frames on a 10 Mbps satellite channel with 270 ms delay. What is the link utilization for stop-and-wait ARQ technique assuming  $P=10^{-3}$ ?
22. Brief about how line coding implemented in FDDI and describe its format.
23. Illustrate the performance issues for GO-BACK-N data link protocol.

# Computer Network

Lecture taken by

Dharmendra Kumar

(Associate Professor)

United College of Engineering and Research, Prayagraj

# Unit-3

# Logical Addressing

## IPv4 ADDRESSES

An IPv4 address is a 32-bit address that uniquely and universally defines the connection of a device (for example, a computer or a router) to the Internet.

**Note:** Two devices on the Internet can never have the same address at the same time.

**Note:** If a device operating at the network layer has  $m$  connections to the Internet, then it needs to have  $m$  addresses. Router is such a device that uses many addresses.

# Logical Addressing

## Address Space

- ❖ A protocol such as IPv4 that defines addresses has an address space. An address space is the total number of addresses used by the protocol. If a protocol uses N bits to define an address, the address space is  $2^N$  values.
- ❖ IPv4 uses 32-bit addresses, which means that the address space is  $2^{32}$  or 4,294,967,296 (more than 4 billion).

# Logical Addressing

## Notations

There are two prevalent notations to show an IPv4 address: binary notation and dotted decimal notation.

### Binary Notation

In binary notation, the IPv4 address is displayed as 32 bits. Each octet is often referred to as a byte.

The following is an example of an IPv4 address in binary notation:

**01110101 10010101 00011101 00000010**

### Dotted-Decimal Notation

The following is the dotted-decimal notation of the above address:

**117.149.29.2**

# Logical Addressing

## Example

Change the following IPv4 addresses from binary notation to dotted-decimal notation.

- a. 10000001 00001011 00001011 11101111
- b. 11000001 10000011 00011011 11111111

## Solution

- a. 129.11.11.239
- b. 193.131.27.255

# Logical Addressing

## Example

Change the following IPv4 addresses from dotted-decimal notation to binary notation.

- a. 111.56.45.78
- b. 221.34.7.82

## Solution

- a. 01101111 00111000 00101101 01001110
- b. 11011101 00100010 00000111 01010010

# Logical Addressing

## Example

Find the error, if any, in the following IPv4 addresses.

- a. 111.56.045.78
- b. 221.34.7.8.20
- c. 75.45.301.14
- d. 11100010.23.14.67

# Logical Addressing

## Classful Addressing

In classful addressing, the address space is divided into five classes: A, B, C, D, and E. Each class occupies some part of the address space.

	First byte	Second byte	Third byte	Fourth byte
Class A	0			
Class B	10			
Class C	110			
Class D	1110			
Class E	1111			

a. Binary notation

	First byte	Second byte	Third byte	Fourth byte
Class A	0–127			
Class B	128–191			
Class C	192–223			
Class D	224–239			
Class E	240–255			

b. Dotted-decimal notation

# Logical Addressing

## Example

Find the class of each address.

- (a) 00000001 00001011 00001011 11101111
- (b) 11000001 10000011 00011011 11111111
- (c) 14.23.120.8
- (d) 252.5.15.111

# Logical Addressing

## Classes and Blocks

One problem with classful addressing is that each class is divided into a fixed number of blocks with each block having a fixed size as shown in the following table:-

<i>Class</i>	<i>Number of Blocks</i>	<i>Block Size</i>	<i>Application</i>
A	128	16,777,216	Unicast
B	16,384	65,536	Unicast
C	2,097,152	256	Unicast
D	1	268,435,456	Multicast
E	1	268,435,456	Reserved

**Note:** In classful addressing, a large part of the available addresses were wasted.

# Logical Addressing

## Netid and Hostid

In classful addressing, an IP address in class A, B, or C is divided into netid and hostid. These parts are of varying lengths, depending on the class of the address.

- ❖ In class A, one byte defines the netid and three bytes define the hostid.
- ❖ In class B, two bytes define the netid and two bytes define the hostid.
- ❖ In class C, three bytes define the netid and one byte defines the hostid.

# Logical Addressing

## Mask

Mask is a 32-bit number made of contiguous 1's followed by contiguous 0's. The masks for classes A, B, and C are shown in the following table. The concept does not apply to classes D and E.

Class	Binary	Dotted-Decimal	CIDR
A	11111111 00000000 00000000 00000000	255.0.0.0	/8
B	11111111 11111111 00000000 00000000	255.255.0.0	/16
C	11111111 11111111 11111111 00000000	255.255.255.0	/24

The mask is used to find the netid and the hostid. For example, the mask for a class A address has eight 1s, which means the first 8 bits of any address in class A define the netid; the next 24 bits define the hostid.

# Logical Addressing

- ❖ The last column of Table shows the mask in the form /n where n can be 8, 16, or 24 in classful addressing.
- ❖ This notation is also called slash notation or Classless Interdomain Routing (CIDR) notation.
- ❖ This notation is used in classless addressing.

# Logical Addressing

## Subnetting

- ❖ If an organization was granted a large block in class A or B, it could divide the addresses into several contiguous groups and assign each group to smaller networks called subnets.
- ❖ Subnetting increases the number of 1's in the mask.

# Logical Addressing

## Supernetting

- ❖ In supernetting, an organization can combine several class C blocks to create a larger range of addresses. In other words, several networks are combined to create a super network or a supernet. An organization can apply for a set of class C blocks instead of just one.
- ❖ For example, an organization that needs 1000 addresses can be granted four contiguous class C blocks. The organization can then use these addresses to create one super network.
- ❖ Supernetting decreases the number of 1's in the mask. For example, if an organization is given four class C addresses, the mask changes from /24 to /22.

# Logical Addressing

## Classless Addressing

In this scheme, there are no classes, but the addresses are still granted in blocks.

### Address Blocks

- ❖ In classless addressing, when an entity, small or large, needs to be connected to the Internet, it is granted a block of addresses. The size of the block (the number of addresses) varies based on the nature and size of the entity.
- ❖ For example, a household may be given only two addresses; a large organization may be given thousands of addresses. An ISP may be given thousands or hundreds of thousands based on the number of customers it may serve.

# Logical Addressing

## Restriction

The Internet authorities impose three restrictions on classless address blocks:

1. The addresses in a block must be contiguous, one after another.
2. The number of addresses in a block must be a power of 2.
3. The first address must be evenly divisible by the number of addresses.

# Logical Addressing

## Example

Following figure shows a block of addresses, in both binary and dotted-decimal notation, granted to a small business that needs 16 addresses.

	Block	
First	205.16.37.32	
	205.16.37.33	
Last	205.16.37.47	

a. Decimal

Block	11001101 00010000 00100101 00100000	11001101 00010000 00100101 00100001	11001101 00010000 00100101 00101111	16 Addresses
				1

b. Binary

# Logical Addressing

## Mask

- ❖ A better way to define a block of addresses is to select any address in the block and the mask.
- ❖ In IPv4 addressing, a block of addresses can be defined as  
 $x.y.z.t/n$   
in which  $x.y.z.t$  defines one of the addresses and the  $/n$  defines the mask.
- ❖ The address and the  $/n$  notation completely define the whole block (the first address, the last address, and the number of addresses).

# Logical Addressing

## First Address

The first address in the block can be found by setting the  $32-n$  rightmost bits in the binary notation of the address to 0s.

## Last Address

The last address in the block can be found by setting the  $32-n$  rightmost bits in the binary notation of the address to 1s.

## Number of Addresses

The number of addresses in the block is the difference between the last and first address. It can easily be found using the formula  $2^{32-n}$ .

# Logical Addressing

## Example

A block of addresses is granted to a small organization. We know that one of the addresses is 205.16.37.39/28.

- (a) What is the first address in the block?
- (b) What is the last address in the block?
- (c) Find the number of addresses in this block.

# Logical Addressing

## Network Addresses

A very important concept in IP addressing is the network address.

The first address is called the network address and defines the organization itself to the rest of the world.

# Logical Addressing

## Hierarchy

IP addresses have levels of hierarchy.

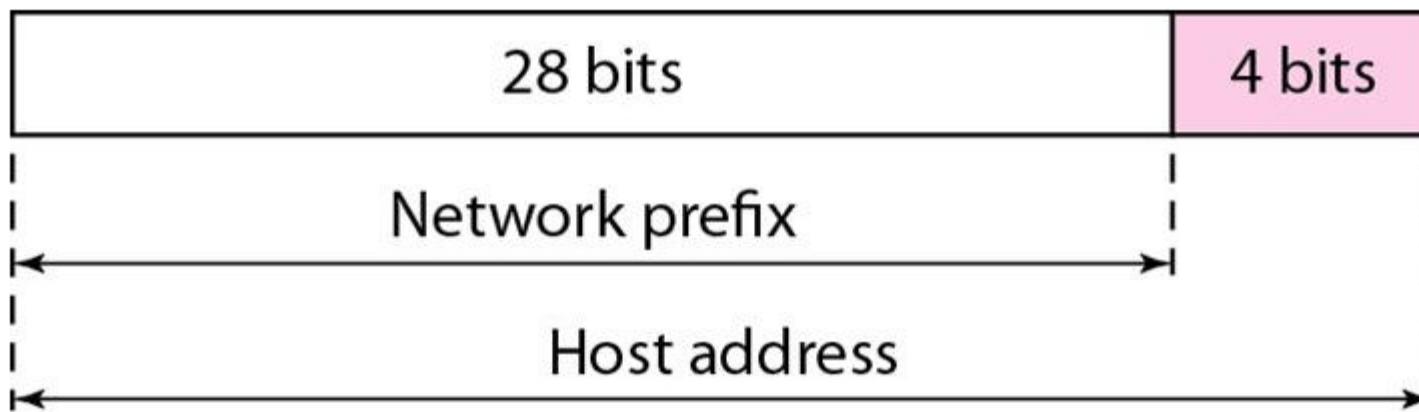
### Two-Level Hierarchy: No Subnetting

- ❖ An IP address can define only two levels of hierarchy when not subnetted. The  $n$  leftmost bits of the address  $x.y.z.t/n$  define the network (organization network); the  $32-n$  rightmost bits define the particular host (computer or router) to the network.
- ❖ The two common terms are prefix and suffix. The part of the address that defines the network is called the prefix; the part that defines the host is called the suffix.

# Logical Addressing

Following figure shows the hierarchical structure of an IPv4 address.

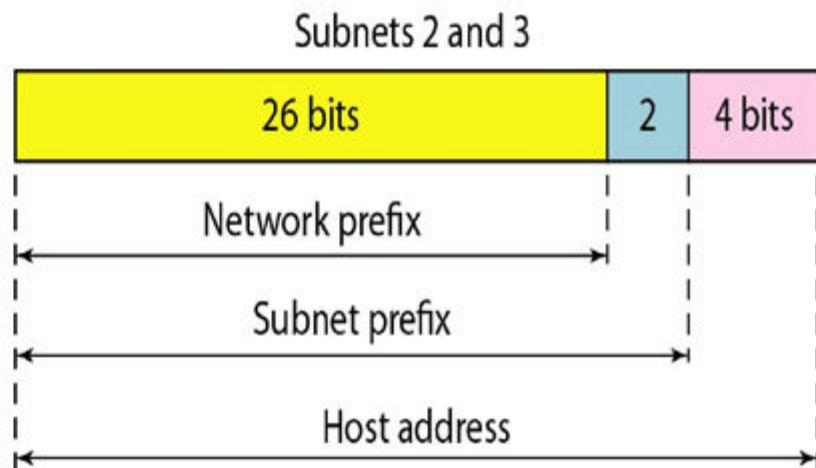
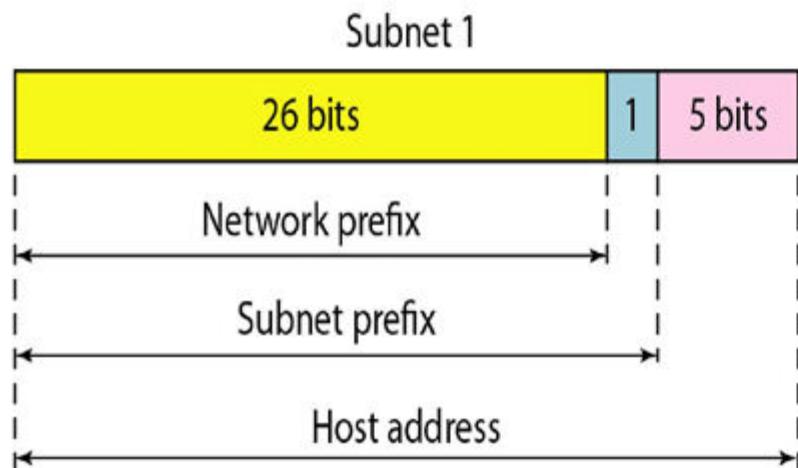
*Two levels of hierarchy in an IPv4 address*



# Logical Addressing

## Three-Levels of Hierarchy: Subnetting

An organization that is granted a large block of addresses may want to create clusters of networks (called subnets) and divide the addresses between the different subnets. The rest of the world still sees the organization as one entity; however, internally there are several subnets. The organization has its own mask; each subnet must also have its own.



# Logical Addressing

## Example:

Suppose an organization is given the block 17.12.40.0/26, which contains 64 addresses. The organization has three offices and needs to divide the addresses into three subblocks of 32, 16, and 16 addresses. Find the mask of these three offices.

## Solution:

- (a) Suppose the mask for the first office is  $n_1$ . Therefore,  
 $2^{32-n_1} = 32$  ,  $n_1 = 27$ .
- (b) Suppose the mask for the first office is  $n_2$ . Therefore,  
 $2^{32-n_2} = 16$  ,  $n_2 = 28$ .
- (c) Suppose the mask for the first office is  $n_3$ . Therefore,  
 $2^{32-n_3} = 16$  ,  $n_3 = 28$ .

# Logical Addressing

## Example

An ISP is granted a block of addresses starting with 190.100.0.0/16 (65,536 addresses). The ISP needs to distribute these addresses to three groups of customers as follows:

- (a) The first group has 64 customers; each needs 256 addresses.
- (b) The second group has 128 customers; each needs 128 addresses.
- (c) The third group has 128 customers; each needs 64 addresses.

Design the subblocks and find out how many addresses are still available after these allocations.

# Logical Addressing

## Solution:

### (a) Group 1

For this group, each customer needs 256 addresses. This means that 8 bits are needed to define each host. The prefix length is then  $32 - 8 = 24$ . The addresses are

1st Customer: 190.100.0.0/24 to 190.100.0.255/24

2nd Customer: 190.100.1.0/24 to 190.100.1.255/24

.....

.....

64th Customer: 190.100.63.0/24 to 190.100.63.255/24

Total =  $64 \times 256 = 16,384$

# Logical Addressing

**Solution:**

**(b) Group 2**

For this group, each customer needs 128 addresses. This means that 7 bits are needed to define each host. The prefix length is then  $32 - 7 = 25$ . The addresses are

1st Customer: 190.100.64.0/25 to 190.100.64.127/25

2nd Customer: 190.100.64.128/25 to 190.100.64.255/25

.....

.....

128th Customer: 190.100.127.128/25 to 190.100.127.255/25

Total =  $128 \times 128 = 16,384$

# Logical Addressing

**Solution:**

**(c) Group 3**

For this group, each customer needs 64 addresses. This means that 6 bits are needed to define each host. The prefix length is then  $32 - 6 = 26$ . The addresses are

1st Customer: 190.100.128.0/26 to 190.100.128.63/26

2nd Customer: 190.100.128.64/26 to 190.100.128.127/26

.....

.....

64th Customer: 190.100.159.192/26 to 190.100.159.255/26

Total =  $128 \times 64 = 8192$

Number of granted addresses to the ISP = 65536

Number of allocated addresses by the ISP =  $16,384 + 16,384 + 8192$   
= 40960

Number of available addresses =  $65536 - 40960$   
**= 24576**

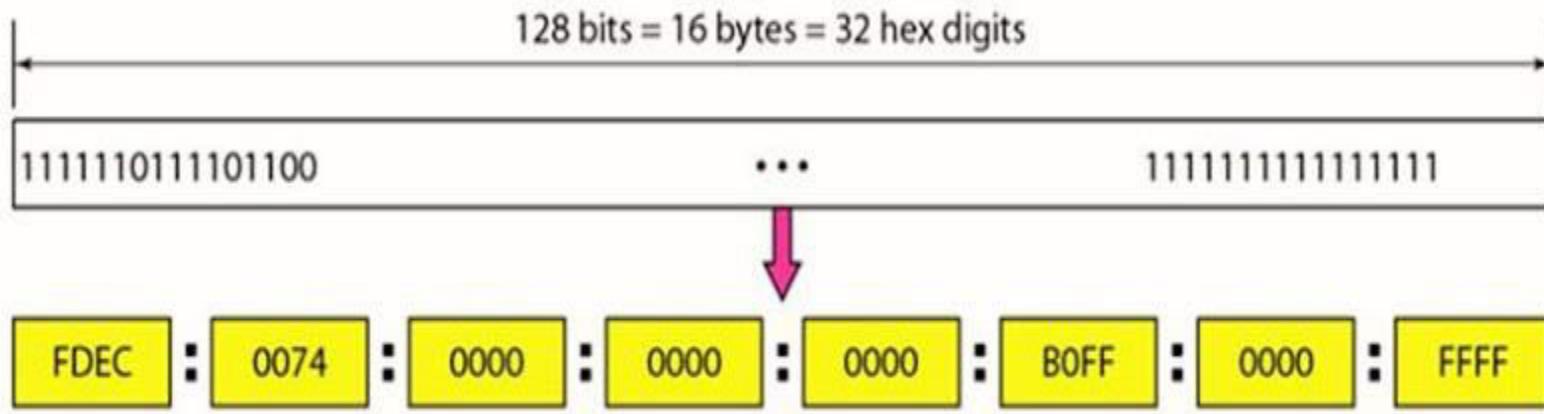
# Logical Addressing

## IPv6 ADDRESSES

An IPv6 address consists of 16 bytes (octets) i.e. it is 128 bits long.

### Hexadecimal Colon Notation

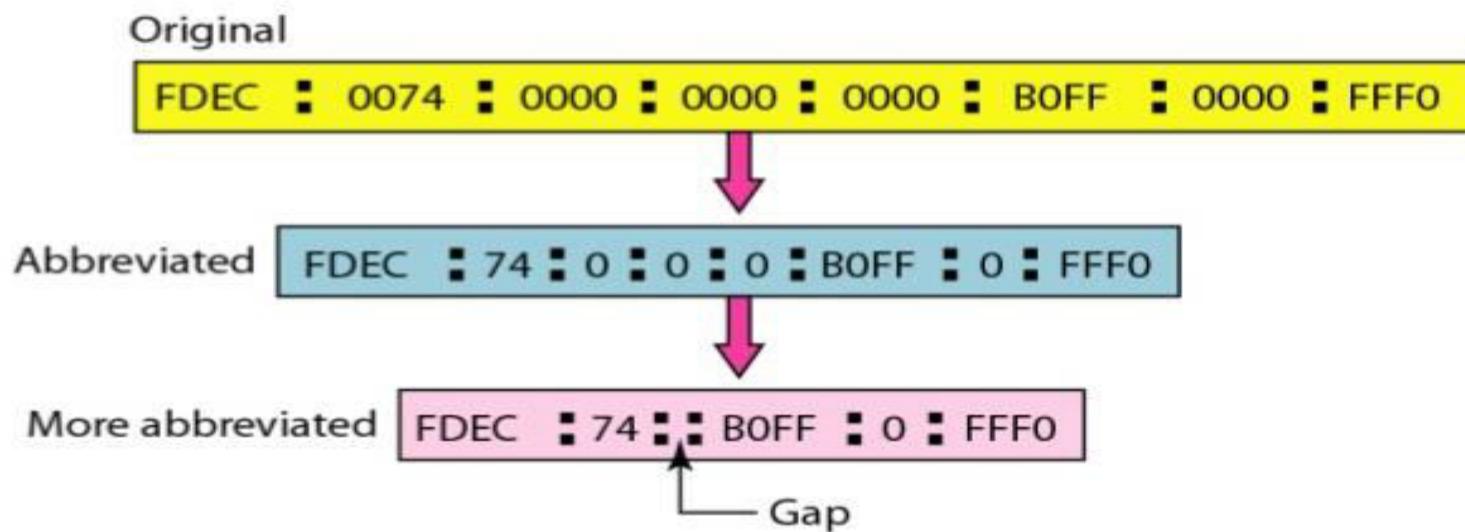
In this notation, 128 bits is divided into eight sections, each 2 bytes in length. Two bytes in hexadecimal notation requires four hexadecimal digits. Therefore, the address consists of 32 hexadecimal digits, with every four digits separated by a colon, as shown in figure:-



# Logical Addressing

## Abbreviation

- ❖ The leading zeros of a section can be omitted.
- ❖ If there are consecutive sections consisting of zeros only. We can remove the zeros altogether and replace them with a double colon. Note that this type of abbreviation is allowed only once per address.



# Logical Addressing

## Example

Expand the address 0:15::1:12:1213 to its original.

## Address Space

- ❖ IPv6 has a much larger address space;  $2^{128}$  addresses are available. The designers of IPv6 divided the address into several categories. A few leftmost bits, called the **type prefix**, in each address define its category.
- ❖ Following table shows the prefix for each type of address. The third column shows the fraction of each type of address relative to the whole address space.

# Logical Addressing

Type Prefix	Type	Fraction
00000000	Reserved	1/256
00000001	Unassigned	1/256
0000001	ISO network addresses	1/128
0000010	IPX (Novell) network addresses	1/128
0000011	Unassigned	1/128
00001	Unassigned	1/32
0001	Reserved	1/16
001	Reserved	1/8
010	Provider-based unicast addresses	1/8

# Logical Addressing

Type Prefix	Type	Fraction
011	Unassigned	1/8
100	Geographic-based unicast addresses	1/8
101	Unassigned	1/8
110	Unassigned	1/8
1110	Unassigned	1/16
11110	Unassigned	1/32
1111 10	Unassigned	1/64
1111 110	Unassigned	1/128
11111110 a	Unassigned	1/512
1111 111010	Link local addresses	1/1024
1111 1110 11	Site local addresses	1/1024
11111111	Multicast addresses	1/256

# Logical Addressing

## Exercise

1. Find the class of the following IP addresses.
  - a. 208.34.54.12
  - b. 238.34.2.1
  - c. 114.34.2.8
  - d. 129.14.6.8.
  
2. Find the class of the following IP addresses.
  - a. 11110111 11110011 10000111 11011101
  - b. 10101111 11000000 11110000 00011101
  - c. 11011111 10110000 00011111 01011101
  - d. 11101111 11110111 11000111 00011101

# Logical Addressing

3. Find the netid and the hostid of the following IP addresses.
  - a. 114.34.2.8
  - b. 132.56.8.6
  - c. 208.34.54.12
4. In a block of addresses, we know the IP address of one host is 25.34.12.56/16. What are the first address (network address) and the last address (limited broadcast address) in this block?
5. An organization is granted the block 16.0.0.0/8. The administrator wants to create 500 fixed-length subnets.
  - a. Find the subnet mask.
  - b. Find the number of addresses in each subnet.
  - c. Find the first and last addresses in subnet 1.
  - d. Find the first and last addresses in subnet 500

# Logical Addressing

6. Write the following masks in slash notation (/n).
- 255.255.255.0
  - 255.0.0.0
  - 255.255.224.0
  - 255.255.240.0
7. Find the range of addresses in the following blocks.
- 123.56.77.32/29
  - 200.17.21.128/27
  - 17.34.16.0/23
  - 180.34.64.64/30

# Logical Addressing

8. An ISP is granted a block of addresses starting with 150.80.0.0/16. The ISP wants to distribute these blocks to 2600 customers as follows.

- a. The first group has 200 medium-size businesses; each needs 128 addresses.
- b. The second group has 400 small businesses; each needs 16 addresses.
- c. The third group has 2000 households; each needs 4 addresses.

Design the subblocks and give the slash notation for each subblock. Find out how many addresses are still available after these allocations.

9. Show the shortest form of the following addresses.

- a. 2340: IABC:119A:A000:0000:0000:0000:0000
- b. 0000:00AA:0000:0000:0000:0000: 119A:A231
- c. 2340:0000:0000:0000:0000: 119A:A001:0000
- d. 0000:0000:0000:2340:0000:0000:0000:0000

# Logical Addressing

10. Show the original (unabbreviated) form of the following addresses.

- a. 0::0
- b. 0:AA::0
- c. 0: 1234::3
- d. 123::1:2

11. What is the type of each of the following addresses?

- a. FE80::12
- b. FEC0: :24A2
- c. FF02::0
- d. 0::01

12. What is the type of each of the following addresses?

- a. 0::0
- b. 0: :FFFF:0:0
- c. 582F:1234::2222
- d. 4821::14:22
- e. 54EF::A234:2

# **Internet Protocol**

# Internet Protocol

## Internet as a Datagram Network

- ❖ The Internet has chosen the datagram approach to switching in the network layer.
- ❖ It uses the universal addresses defined in the network layer to route packets from the source to the destination.

# Internet Protocol

## Internet as a Connectionless Network

- ❖ Delivery of a packet can be accomplished by using either a connection-oriented or a connectionless network service.
- ❖ In a **connection-oriented service**, the source first makes a connection with the destination before sending a packet. When the connection is established, a sequence of packets from the same source to the same destination can be sent one after another. In this case, there is a relationship between packets. They are sent on the same path in sequential order. A packet is logically connected to the packet traveling before it and to the packet traveling after it. When all packets of a message have been delivered, the connection is terminated.
- ❖ This type of service is used in a virtual-circuit approach. to packet switching such as in Frame Relay and ATM.

# Internet Protocol

In **connectionless service**, the network layer protocol treats each packet independently, with each packet having no relationship to any other packet. The packets in a message may or may not travel the same path to their destination.

This type of service is used in the datagram approach to packet switching. The **Internet** has chosen this type of service at the network layer.

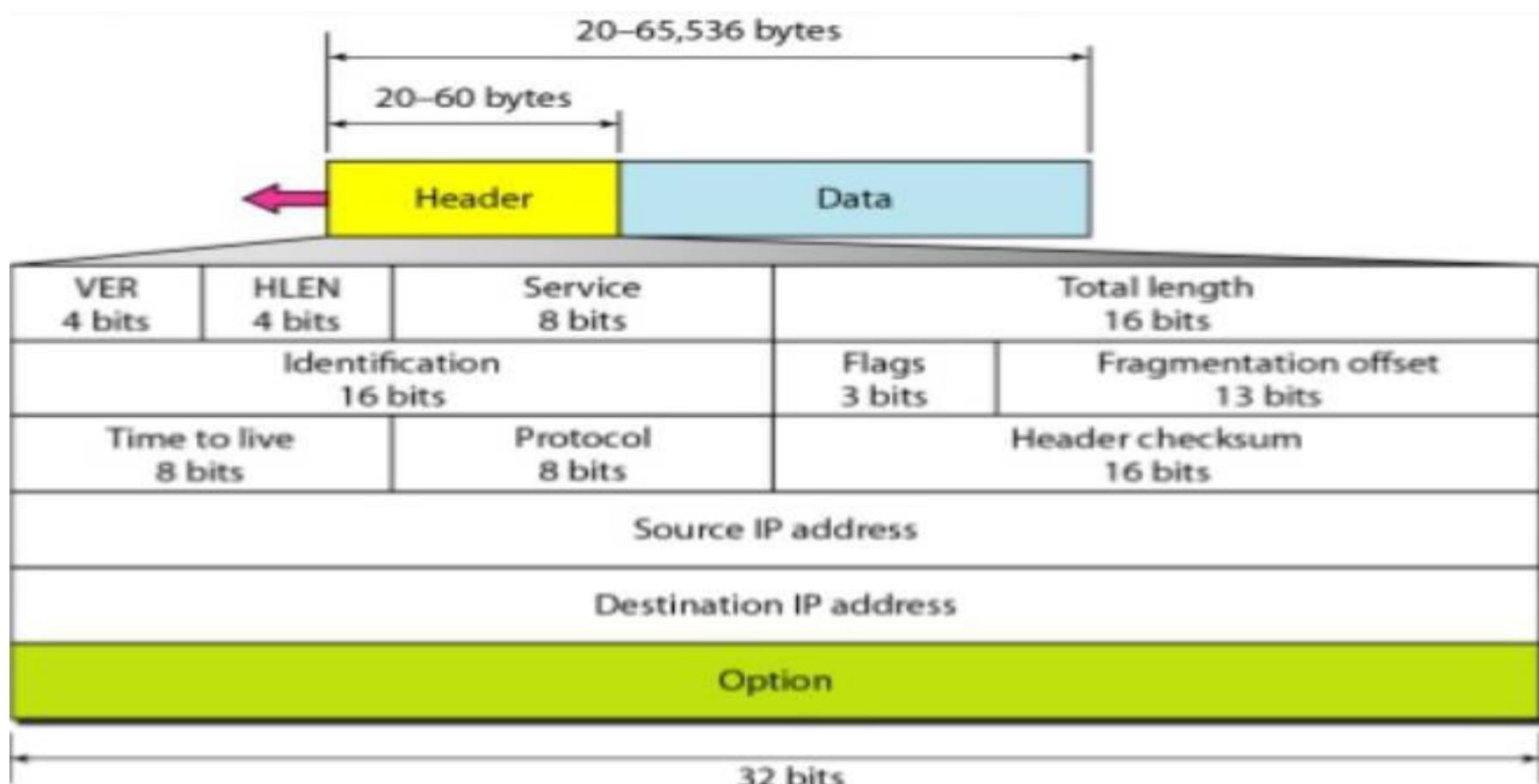
# IPv4

- ❖ The Internet Protocol version 4 (IPv4) is the delivery mechanism used by the TCP/IP protocols.
- ❖ IPv4 is an unreliable and connectionless datagram protocol-a best-effort delivery service.
- ❖ The term best-effort means that IPv4 provides no error control or flow control (except for error detection on the header).
- ❖ If reliability is important, IPv4 must be paired with a reliable protocol such as TCP.

# IPv4

## Datagram

Packets in the IPv4 layer are called datagrams. Following figure shows the IPv4 datagram format.



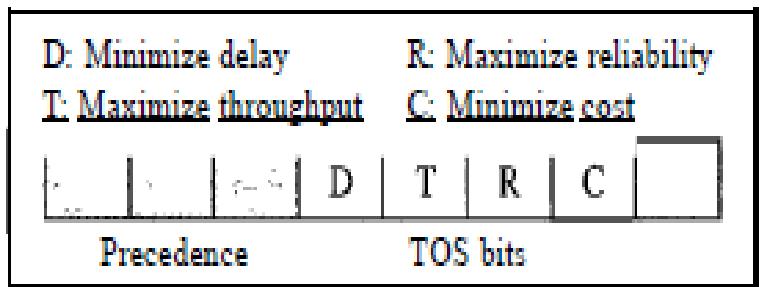
# IPv4

- ❖ A datagram is a variable-length packet consisting of two parts: header and data.
- ❖ The header is 20 to 60 bytes in length and contains information essential to routing and delivery.

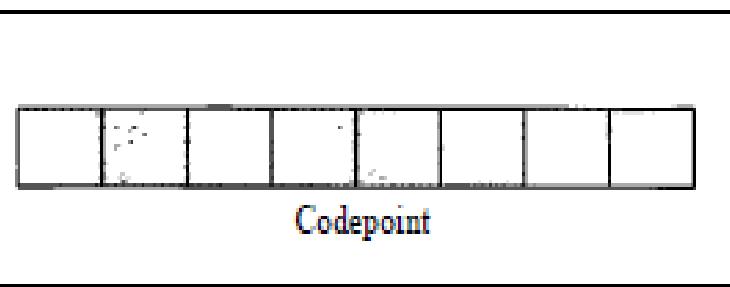
**Version (VER):** This 4-bit field defines the version of the IPv4 protocol.

**Header length (HLEN):** This 4-bit field defines the total length of the datagram header in 4-byte words.

**Services:** This field, previously called service type, is now called differentiated services.



Service type



Differentiated services

# IPv4

## Service Type

In this interpretation, the first 3 bits are called precedence bits. The next 4 bits are called type of service (TOS) bits, and the last bit is not used.

- a. Precedence is a 3-bit subfield ranging from 0 (000 in binary) to 7 (111 in binary). The precedence defines the priority of the datagram in issues such as congestion. If a router is congested and needs to discard some datagrams, those datagrams with lowest precedence are discarded first.
- b. TOS bits is a 4-bit subfield with each bit having a special meaning. Although a bit can be either 0 or 1, one and only one of the bits can have the value of 1 in each datagram.

# IPv4

## Total length:

- ❖ This is a 16-bit field that defines the total length (header plus data) of the IPv4 datagram in bytes.
- ❖ To find the length of the data coming from the upper layer, subtract the header length from the total length.
- ❖ The header length can be found by multiplying the value in the HLEN field by 4.

**Length of data = total length - header length**

**Identification:** This field is used in fragmentation.

**Flags:** This field is used in fragmentation

**Fragmentation offset:** This field is used in fragmentation

# IPv4

**Time to live:** A datagram has a limited lifetime in its travel through an internet.

- ❖ This field was originally designed to hold a timestamp, which was decremented by each visited router. The datagram was discarded when the value became zero.
- ❖ Today, this field is used mostly to control the maximum number of hops (routers) visited by the datagram. When a source host sends the datagram, it stores a number in this field. This value is approximately 2 times the maximum number of routes between any two hosts. Each router that processes the datagram decrements this number by 1. If this value, after being decremented, is zero, the router discards the datagram.

# IPv4

**Protocol:** This 8-bit field defines the higher-level protocol that uses the services of the IPv4 layer. An IPv4 datagram can encapsulate data from several higher-level protocols such as TCP, UDP, ICMP, and IGMP. This field specifies the final destination protocol to which the IPv4 datagram is delivered.

The value of this field for each higher-level protocol is shown in the following table:-

<i>Value</i>	<i>Protocol</i>
1	ICMP
2	IGMP
6	TCP
17	UDP
89	OSPF

# IPv4

**Checksum:** This field is used to detect error.

**Source address:** This 32-bit field defines the IPv4 address of the source. This field must remain unchanged during the time the IPv4 datagram travels from the source host to the destination host.

**Destination address:** This 32-bit field defines the IPv4 address of the destination. This field must remain unchanged during the time the IPv4 datagram travels from the source host to the destination host.

# IPv4

## Example

In an IPv4 packet, the value of HLEN is 1000 in binary. How many bytes of options are being carried by this packet?

## Solution

The HLEN value is 8, which means the total number of bytes in the header is  $8 \times 4$ , or 32 bytes. The first 20 bytes are the base header, the next 12 bytes are the options.

## Example

In an IPv4 packet, the value of HLEN is 5, and the value of the total length field is Ox0028. How many bytes of data are being carried by this packet?

## Solution

The HLEN value is 5, which means the total number of bytes in the header is  $5 \times 4$ , or 20 bytes (no options). The total length is 40 bytes, which means the packet is carrying 20 bytes of data (40- 20).

# IPv4

## Example

An IPv4 packet has arrived with the first few hexadecimal digits as shown.

Ox45000028000100000102 ...

How many hops can this packet travel before being dropped? The data belong to what upper-layer protocol?

## Solution

To find the time-to-live field, we skip 8 bytes (16 hexadecimal digits). The time-to-live field is the ninth byte, which is 01. This means the packet can travel only one hop. The protocol field is the next byte (02), which means that the upper-layer protocol is IGMP.

# IPv4

## Fragmentation

### Maximum Transfer Unit (MTU)

Each data link layer protocol has its own frame format in most protocols. One of the fields defined in the format is the maximum size of the data field. In other words, when a datagram is encapsulated in a frame, the total size of the datagram must be less than this maximum size, which is defined by the restrictions imposed by the hardware and software used in the network.

The value of the MTU depends on the physical network protocol. Following table shows the values for some protocols:-

<i>Protocol</i>	<i>MTU</i>
Hyperchannel	65,535
Token Ring (16 Mbps)	17,914
Token Ring (4 Mbps)	4,464
FDDI	4,352
Ethernet	1,500
X.25	576
PPP	296

# IPv4

- ❖ If we divide the datagram into fragments to make it possible to pass through the networks, then this process is called the **fragmentation**.
- ❖ When a datagram is fragmented, each fragment has its own header with most of the fields repeated, but with some changed. A fragmented datagram may itself be fragmented if it encounters a network with an even smaller MTU. In other words, a datagram can be fragmented several times before it reaches the final destination.

# IPv4

## Fields Related to Fragmentation

The fields that are related to fragmentation and reassembly of an IPv4 datagram are the identification, flags, and fragmentation offset fields.

### Identification:

- ❖ This 16-bit field identifies a datagram originating from the source host. The combination of the identification and source IPv4 address must uniquely define a datagram as it leaves the source host.
- ❖ When a datagram is fragmented, the value in the identification field is copied to all fragments. In other words, all fragments have the same identification number, the same as the original datagram.
- ❖ The identification number helps the destination in reassembling the datagram. It knows that all fragments having the same identification value must be assembled into one datagram.

# IPv4

## Flags:

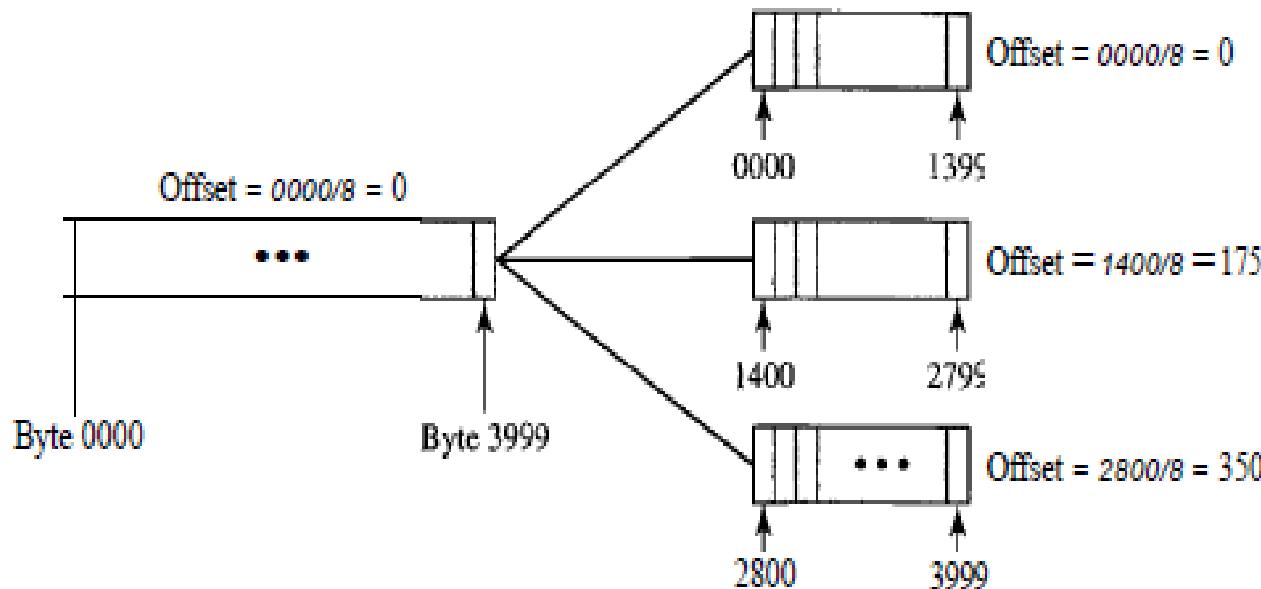
- ❖ This is a 3-bit field.
- ❖ The first bit is reserved.
- ❖ The second bit is called the do not fragment bit. If its value is 1, the machine must not fragment the datagram. If it cannot pass the datagram through any available physical network, it discards the datagram and sends an ICMP error message to the source host. If its value is 0, the datagram can be fragmented if necessary.
- ❖ The third bit is called the more fragment bit. If its value is 1, it means the datagram is not the last fragment; there are more fragments after this one. If its value is 0, it means this is the last or only fragment.



# IPv4

**Fragmentation offset:** This 13-bit field shows the relative position of this fragment with respect to the whole datagram. It is the offset of the data in the original datagram measured in units of 8 bytes.

**Example:** Following figure shows a datagram with a data size of 4000 bytes fragmented into three fragments.

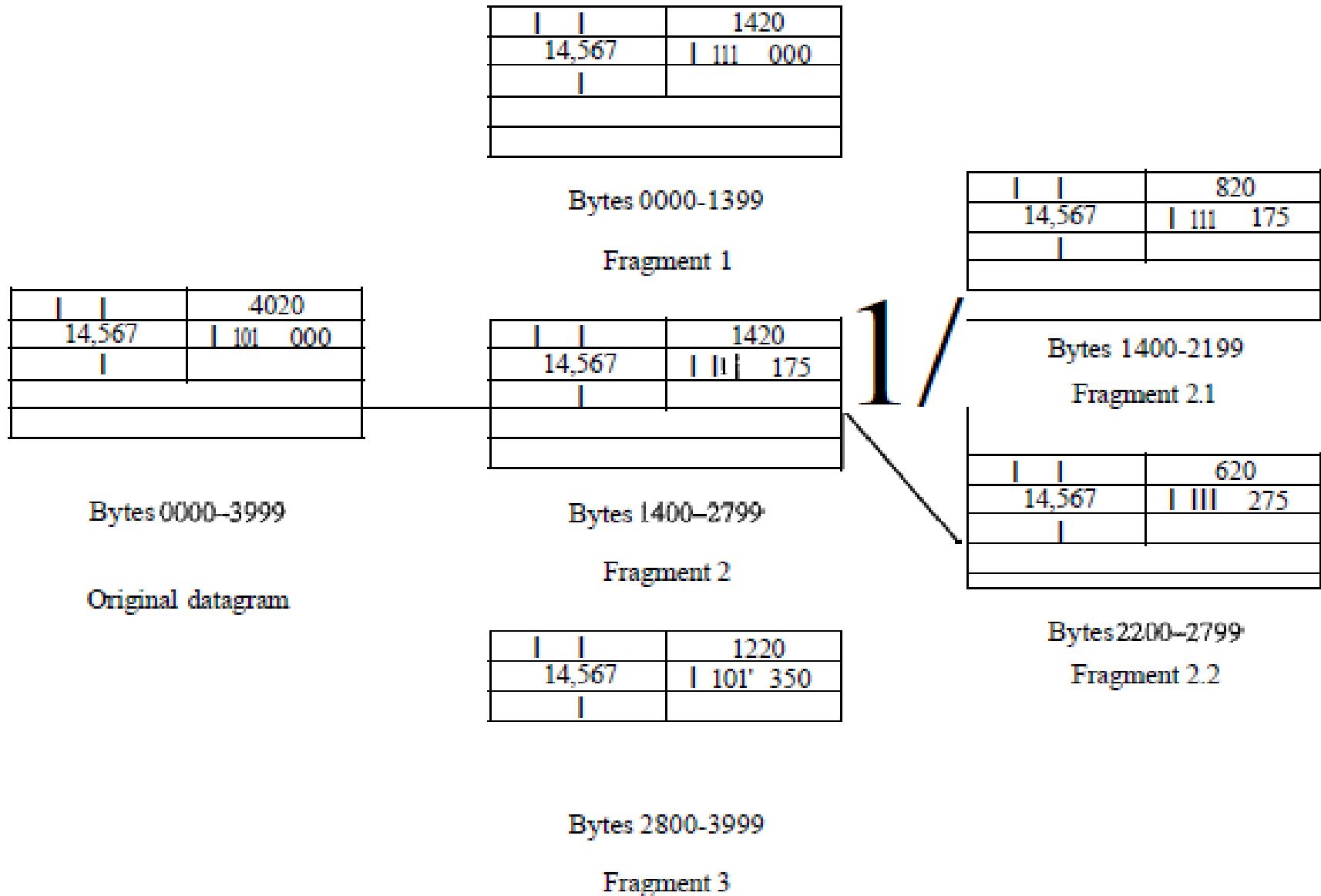


# IPv4

The bytes in the original datagram are numbered 0 to 3999.

- ❖ The first fragment carries bytes 0 to 1399. The offset for this datagram is  $0/8 = 0$ .
- ❖ The second fragment carries bytes 1400 to 2799; the offset value for this fragment is  $1400/8 = 175$ .
- ❖ Finally, the third fragment carries bytes 2800 to 3999. The offset value for this fragment is  $2800/8 = 350$ .

# IPv4



# IPv4

If each fragment follows a different path and arrives out of order, the final destination host can reassemble the original datagram from the fragments received (if none of them is lost) by using the following strategy:

1. The first fragment has an offset field value of zero.
2. Divide the length of the first fragment by 8. The second fragment has an offset value equal to that result.
3. Divide the total length of the first and second fragments by 8. The third fragment has an offset value equal to that result.
4. Continue the process. The last fragment has a more bit value of 0.

# IPv4

## Example

A packet has arrived with an M bit value of 0. Is this the first fragment, the last fragment, or a middle fragment? Do we know if the packet was fragmented?

## Solution

If the M bit is 0, it means that there are no more fragments; the fragment is the last one. However, we cannot say if the original packet was fragmented or not. A non fragmented packet is considered the last fragment.

## Example

A packet has arrived with an M bit value of 1. Is this the first fragment, the last fragment, or a middle fragment? Do we know if the packet was fragmented?

## Solution

If the M bit is 1, it means that there is at least one more fragment. This fragment can be the first one or a middle one, but not the last one. We don't know if it is the first one or a middle one; we need more information (the value of the fragmentation offset).

# IPv4

## Example

A packet has arrived with an M bit value of 1 and a fragmentation offset value of 0. Is this the first fragment, the last fragment, or a middle fragment?

## Solution

Because the M bit is 1, it is either the first fragment or a middle one. Because the offset value is 0, it is the first fragment.

## Example

A packet has arrived in which the offset value is 100. What is the number of the first byte? Do we know the number of the last byte?

## Solution

To find the number of the first byte, we multiply the offset value by 8. This means that the first byte number is 800. We cannot determine the number of the last byte unless we know the length of the data.

# IPv4

## Example

A packet has arrived in which the offset value is 100, the value of HLEN is 5, and the value of the total length field is 100. What are the numbers of the first byte and the last byte?

## Solution

The first byte number is  $100 \times 8 = 800$ . The total length is 100 bytes, and the header length is 20 bytes ( $5 \times 4$ ), which means that there are 80 bytes in this datagram. If the first byte number is 800, the last byte number must be 879.

# IPv6

## IPv6:

IPv4 has **some deficiencies** that make it unsuitable for the fast-growing Internet.

- ❖ Despite all short-term solutions, such as subnetting, classless addressing, and NAT, address depletion is still a long-term problem in the Internet.
- ❖ The Internet must accommodate real-time audio and video transmission. This type of transmission requires minimum delay strategies and reservation of resources not provided in the IPv4 design.
- ❖ The Internet must accommodate encryption and authentication of data for some applications. No encryption or authentication is provided by IPv4.

To overcome these deficiencies, IPv6 (Internetworking Protocol, version 6), also known as IPng (Internetworking Protocol, next generation), was proposed.

# IPv6

## Advantages of IPv6 over IPv4

- ❖ **Larger address space:** An IPv6 address is 128 bits long. Compared with the 32-bit address of IPv4, this is a huge ( $2^{96}$ ) increase in the address space.
- ❖ **Better header format:** IPv6 uses a new header format in which options are separated from the base header and inserted, when needed, between the base header and the upper-layer data. This simplifies and speeds up the routing process because most of the options do not need to be checked by routers.
- ❖ **New options:** IPv6 has new options to allow for additional functionalities.
- ❖ **Allowance for extension:** IPv6 is designed to allow the extension of the protocol if required by new technologies or applications.

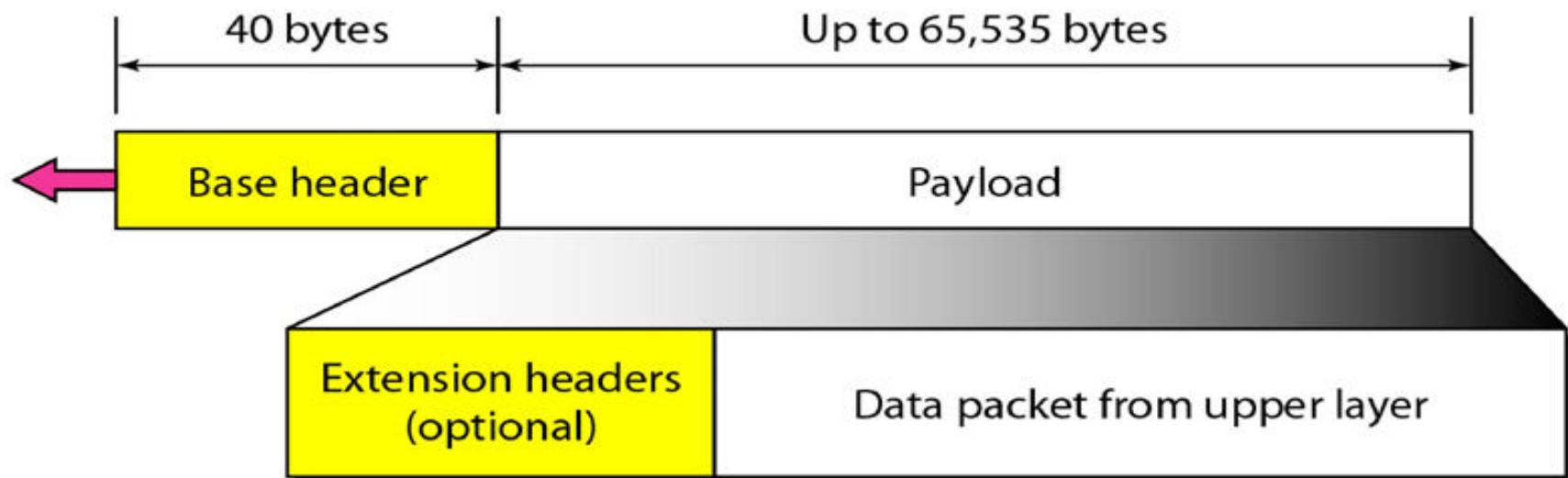
# IPv6

- ❖ Support for resource allocation. In IPv6, the type-of-service field has been removed, but a mechanism (called Flow label) has been added to enable the source to request special handling of the packet. This mechanism can be used to support traffic such as real-time audio and video.
- ❖ Support for more security. The encryption and authentication options in IPv6 provide confidentiality and integrity of the packet.

# IPv6

## Packet Format

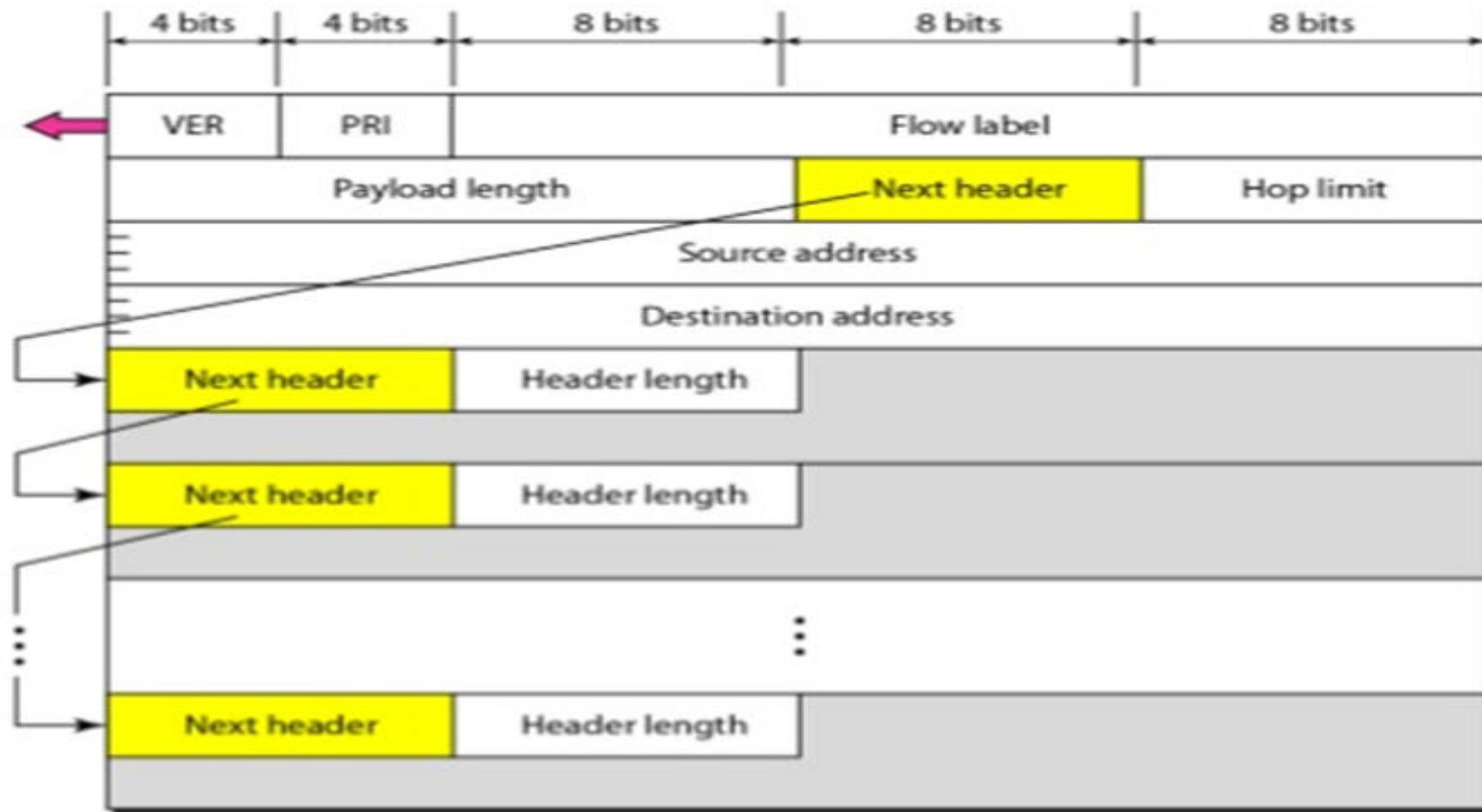
Each packet is composed of a mandatory base header followed by the payload. The payload consists of two parts: optional extension headers and data from an upper layer. The base header occupies 40 bytes, whereas the extension headers and data from the upper layer contain up to 65,535 bytes of information. It is shown in the following figure:-



**IPv6 datagram header and payload**

# IPv6

## Format of an IPv6 datagram



# IPv6

## Base Header

The fields in the base header are as the following:-

**Version:** This 4-bit field defines the version number of the IP. For IPv6, the value is 6.

**Priority:** The 4-bit priority field defines the priority of the packet with respect to traffic congestion.

**Flow label:** The flow label is a 3-byte (24-bit) field that is designed to provide special handling for a particular flow of data.

**Payload length:** The 2-byte payload length field defines the length of the IP datagram excluding the base header.

# IPv6

**Next header:** The next header is an 8-bit field defining the header that follows the base header in the datagram. The next header is either one of the optional extension headers used by IP or the header of an encapsulated packet such as UDP or TCP. Each extension header also contains this field. Note that this field in version 4 is called the protocol.

**Hop limit:** This 8-bit hop limit field serves the same purpose as the TTL field in IPv4.

**Source address:** The source address field is a 16-byte (128-bit) Internet address that identifies the original source of the datagram.

**Destination address:** The destination address field is a 16-byte (128-bit) Internet address that usually identifies the final destination of the datagram.

# Exercise

1. An IPv4 datagram has arrived with the following information in the header (in hexadecimal):

0x45 00 00 54 00 03 58 50 20 06 00 00 7C 4E 03 02 B4 0E 0F 02

- a. Is the packet corrupted?
- b. Are there any options?
- c. Is the packet fragmented?
- d. What is the size of the data?
- e. How many more routers can the packet travel to?
- f. What is the identification number of the packet?
- g. What is the type of service?

2. In an IPv4 datagram, the M bit is 0, the value of HLEN is 5, the value of total length is 200, and the offset value is 200. What is the number of the first byte and number of the last byte in this datagram? Is this the last fragment, the first fragment, or a middle fragment?

# **Delivery, Forwarding, and Routing**

# Delivery, Forwarding, and Routing

- ❖ Delivery refers to the way a packet is handled by the underlying networks under the control of the network layer.
- ❖ Forwarding refers to the way a packet is delivered to the next station.
- ❖ Routing refers to the way routing tables are created to help in forwarding. Routing protocols are used to continuously update the routing tables that are consulted for forwarding and routing.

# Delivery, Forwarding, and Routing

## Delivery

The delivery of a packet to its final destination is accomplished by using two different methods of delivery, direct and indirect.

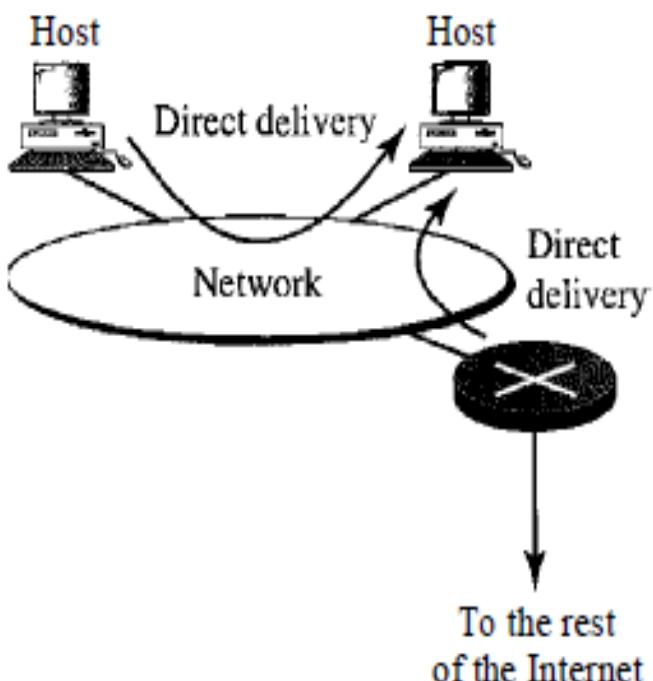
### Direct Delivery

- ❖ In a direct delivery, the final destination of the packet is a host connected to the same physical network as the deliverer.
- ❖ Direct delivery occurs when the source and destination of the packet are located on the same physical network or when the delivery is between the last router and the destination host.

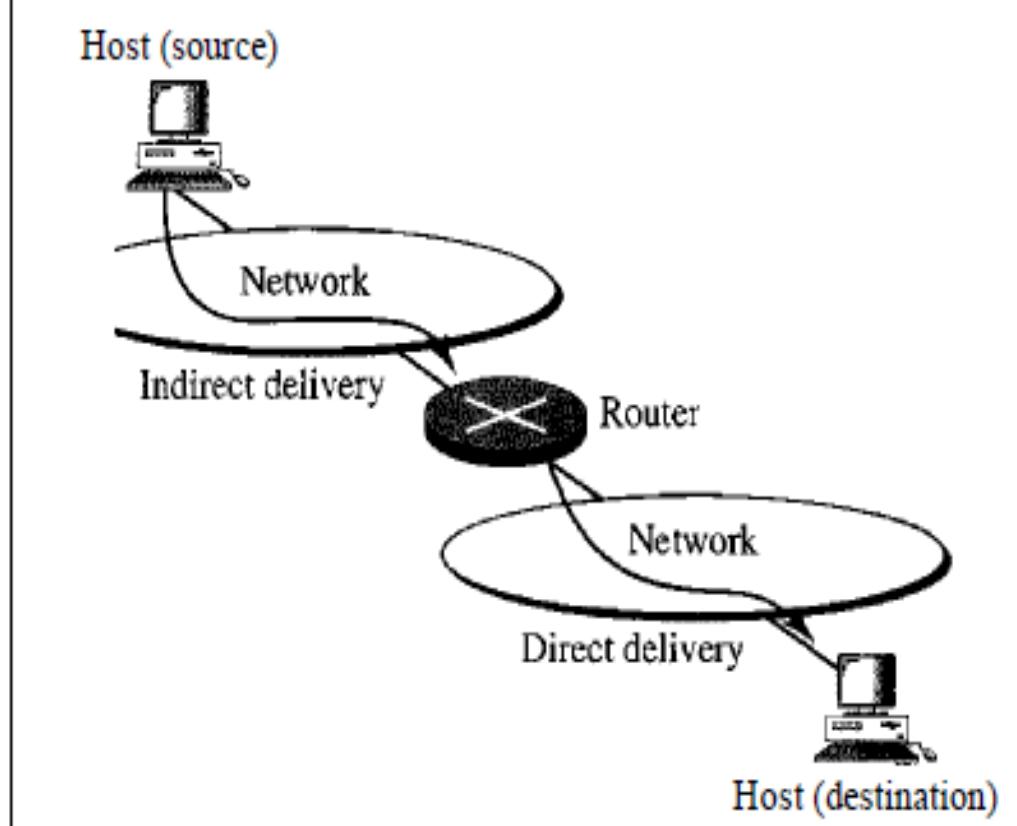
### Indirect Delivery

- ❖ If the destination host is not on the same network as the deliverer, the packet is delivered indirectly.
- ❖ In an indirect delivery, the packet goes from router to router until it reaches the one connected to the same physical network as its final destination.

# Delivery, Forwarding, and Routing



a. Direct delivery



b. Indirect and direct delivery

- Note:** (1) A delivery always involves one direct delivery but zero or more indirect deliveries.
- (2) The last delivery is always a direct delivery.

# Delivery, Forwarding, and Routing

## Forwarding

Forwarding means to place the packet in its route to its destination. Forwarding requires a host or a router to have a routing table. When a host has a packet to send or when a router has received a packet to be forwarded, it looks at this table to find the route to the final destination.

There are following methods of forwarding:-

# Delivery, Forwarding, and Routing

## Next-Hop Method Versus Route Method

One technique to reduce the contents of a routing table is called the next-hop method. In this technique, the routing table holds only the address of the next hop instead of information about the complete route (route method).

### a. Routing tables based on route

Destination	Route
HostB	R1, R2, host B

Destination	Route
HostB	R2, host B

Destination	Route
HostB	HostB

### Routing table for host A

### Routing table for R1

### Routing table for R2

### b. Routing tables based on next hop

Destination	Next hop
Host B	R1

Destination	Next hop
HostB	R2

Destination	Next hop
Host B	

Host A



Network

R1

HostB



Network

R2

# Delivery, Forwarding, and Routing

## Network-Specific Method Versus Host-Specific Method

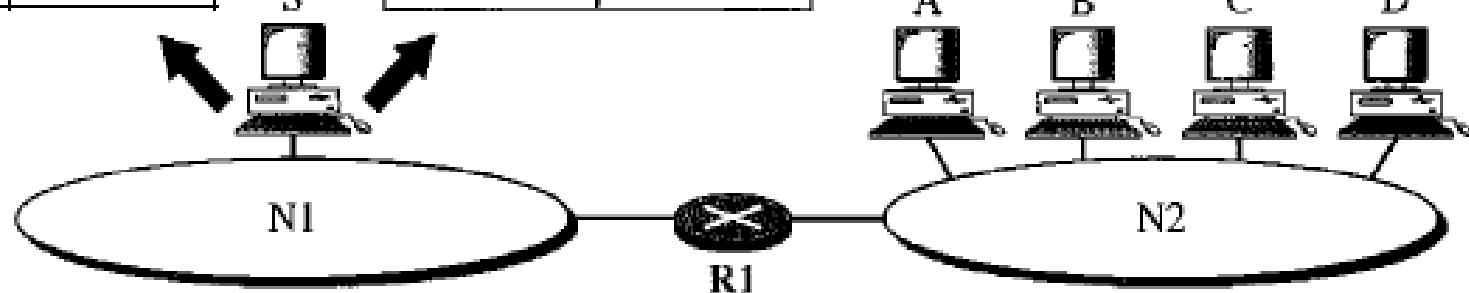
A second technique to reduce the routing table and simplify the searching process is called the network-specific method. Here, instead of having an entry for every destination host connected to the same physical network (host-specific method), we have only one entry that defines the address of the destination network itself. In other words, we treat all hosts connected to the same network as one single entity.

Routing table for host S based  
on host-specific method

Destination	Next hop
A	R1
B	R1
C	R1
D	R1

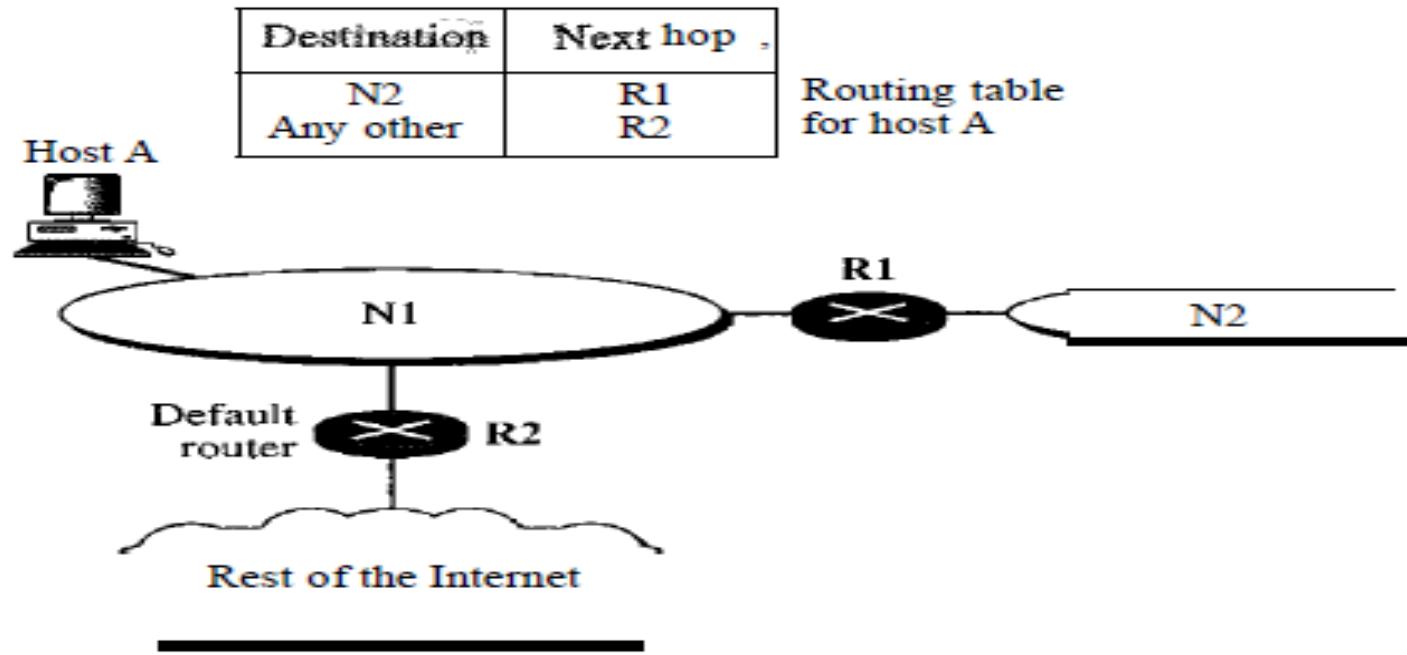
Routing table for host S based  
on network-specific method

Destination	Next hop
N2	R1



# Delivery, Forwarding, and Routing

## Default Method



In this figure, host A is connected to a network with two routers. Router R1 routes the packets to hosts connected to network N2. However, for the rest of the Internet, router R2 is used. So instead of listing all networks in the entire Internet, host A can just have one entry called the **default** (normally defined as network address).

# Delivery, Forwarding, and Routing

## Structure of Routing Table

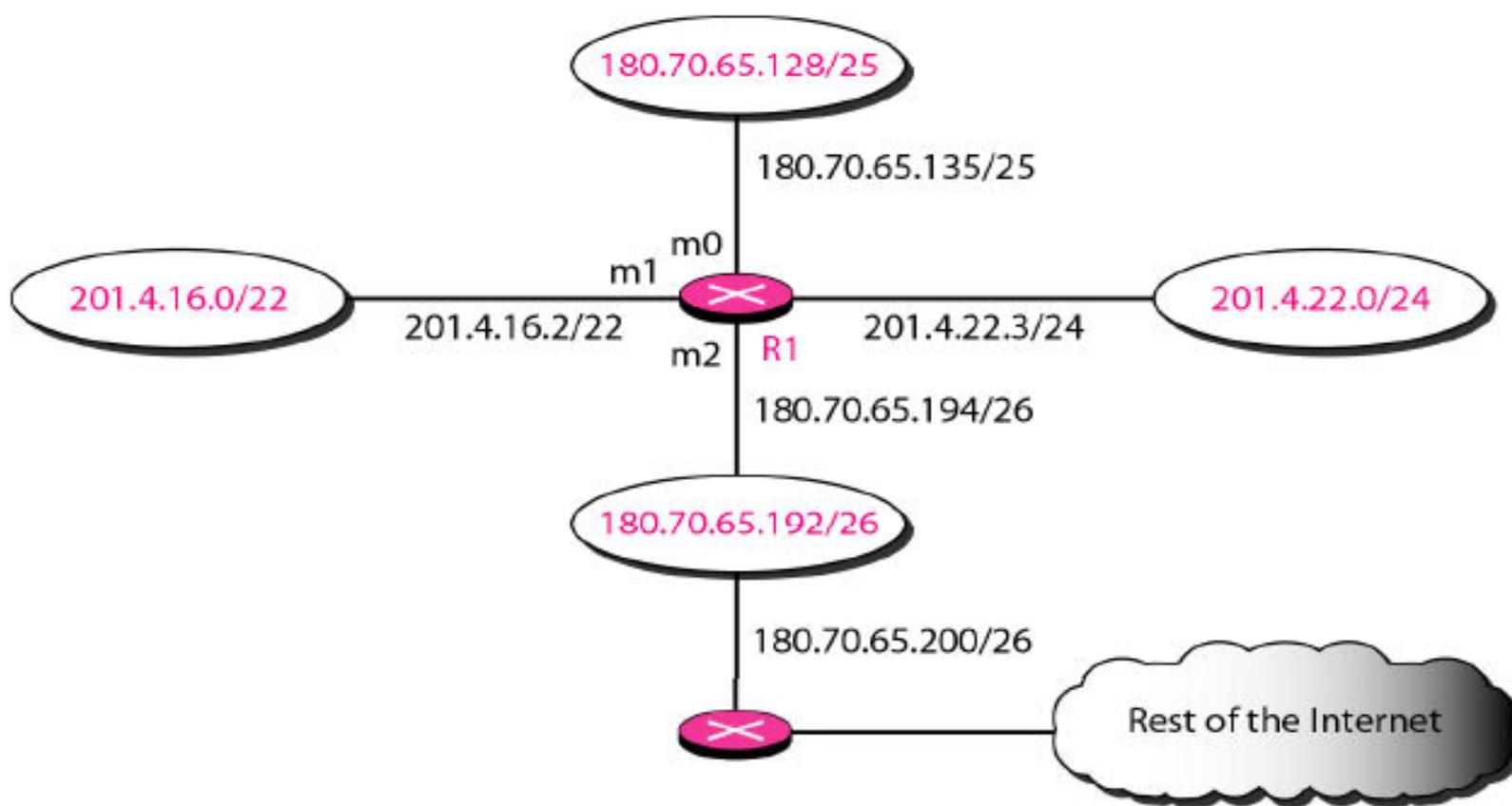
In classless addressing, we need at least four columns in a routing table.

Mask <i>(In)</i>	Network address	Next-hop address	Interface

# Delivery, Forwarding, and Routing

## Example

Make a routing table for router R1, using the configuration in shown in following figure:-



# Delivery, Forwarding, and Routing

**Solution:** Routing table corresponding to router R1 is shown in the following figure:-

<i>Mask</i>	<i>Network Address</i>	<i>Next Hop</i>	<i>Interface</i>
/26	180.70.65.192	—	m2
/25	180.70.65.128	—	m0
/24	201.4.22.0	—	m3
/22	201.4.16.0	....	m1
Any	Any	180.70.65.200	m2

# Delivery, Forwarding, and Routing

## Example

Show the forwarding process if a packet arrives at R1 in previous example with the destination address 180.70.65.140.

## Solution

The router performs the following steps:

1. The first mask (/26) is applied to the destination address. The result is 180.70.65.128, which does not match the corresponding network address.
2. The second mask (/25) is applied to the destination address. The result is 180.70.65.128, which matches the corresponding network address. Therefore, router forwards the packet at the interface number m0.

# Delivery, Forwarding, and Routing

## Example

Show the forwarding process if a packet arrives at R1 in previous example with the destination address 201.4.22.35.

## Solution

The router performs the following steps:

1. The first mask (/26) is applied to the destination address. The result is 201.4.22.0, which does not match the corresponding network address (row 1).
2. The second mask (/25) is applied to the destination address. The result is 201.4.22.0, which does not match the corresponding network address (row 2).
3. The third mask (/24) is applied to the destination address. The result is 201.4.22.0, which matches the corresponding network address. The destination address of the packet and the interface number m3 are passed to ARP.

# Delivery, Forwarding, and Routing

## Example

Show the forwarding process if a packet arrives at R1 in previous example with the destination address 18.24.32.78.

## Solution

This time all masks are applied, one by one, to the destination address, but no matching network address is found. When it reaches the end of the table, the module gives the next-hop address 180.70.65.200 and interface number m2 to ARP. This is probably an outgoing package that needs to be sent, via the default router, to someplace else in the Internet.

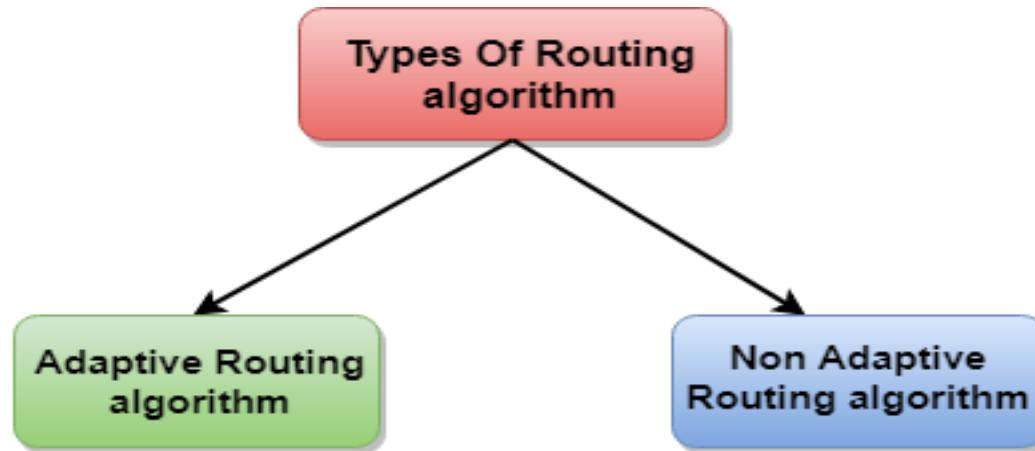
# Delivery, Forwarding, and Routing

## Routing algorithm

- ❖ The routing protocol is a routing algorithm that provides the best path from the source to the destination. The best path is the path that has the "least-cost path" from source to the destination.
- ❖ Routing is the process of forwarding the packets from source to the destination but the best route to send the packets is determined by the routing algorithm.

# Delivery, Forwarding, and Routing

## Classification of a Routing algorithm



### Adaptive Routing algorithm

- ❖ An adaptive routing algorithm is also known as dynamic routing algorithm.
- ❖ This algorithm makes the routing decisions based on the topology and network traffic.
- ❖ The main parameters related to this algorithm are hop count, distance and estimated transit time.

# Delivery, Forwarding, and Routing

## Non-Adaptive Routing algorithm

- ❖ Non-Adaptive routing algorithm is also known as a static routing algorithm.
- ❖ Non-Adaptive routing algorithms do not take the routing decision based on the network topology or network traffic.

The Non-Adaptive Routing algorithm is of two types:

**Flooding:** In case of flooding, every incoming packet is sent to all the outgoing links except the one from it has been reached. The disadvantage of flooding is that node may contain several copies of a particular packet.

**Random walks:** In case of random walks, a packet sent by the node to one of its neighbors randomly. An advantage of using random walks is that it uses the alternative routes very efficiently.

# Delivery, Forwarding, and Routing

## Unicast Routing Protocols

- ❖ A routing table can be either static or dynamic. A static table is one with manual entries.
- ❖ A dynamic table, on the other hand, is one that is updated automatically when there is a change somewhere in the internet.
- ❖ Today, an internet needs dynamic routing tables.
- ❖ A routing protocol is a combination of rules and procedures that lets routers in the internet inform each other of changes. It allows routers to share whatever they know about the internet or their neighborhood.

# Delivery, Forwarding, and Routing

## Intra domain and Inter domain Routing

- ❖ An internet is divided into autonomous systems.
- ❖ An autonomous system (AS) is a group of networks and routers under the authority of a single administration.
- ❖ Routing inside an autonomous system is referred to as intra domain routing.
- ❖ Routing between autonomous systems is referred to as inter domain routing.
- ❖ Each autonomous system can choose one or more intra domain routing protocols to handle routing inside the autonomous system. However, only one inter domain routing protocol handles routing between autonomous systems.

# Delivery, Forwarding, and Routing

- ❖ We are going to discuss two intra domain routing protocols: distance vector and link state and one inter domain routing protocol: Path vector.
- ❖ Routing Information Protocol (RIP) is an implementation of the distance vector protocol.
- ❖ Open Shortest Path First (OSPF) is an implementation of the link state protocol.
- ❖ Border Gateway Protocol (BGP) is an implementation of the path vector protocol.

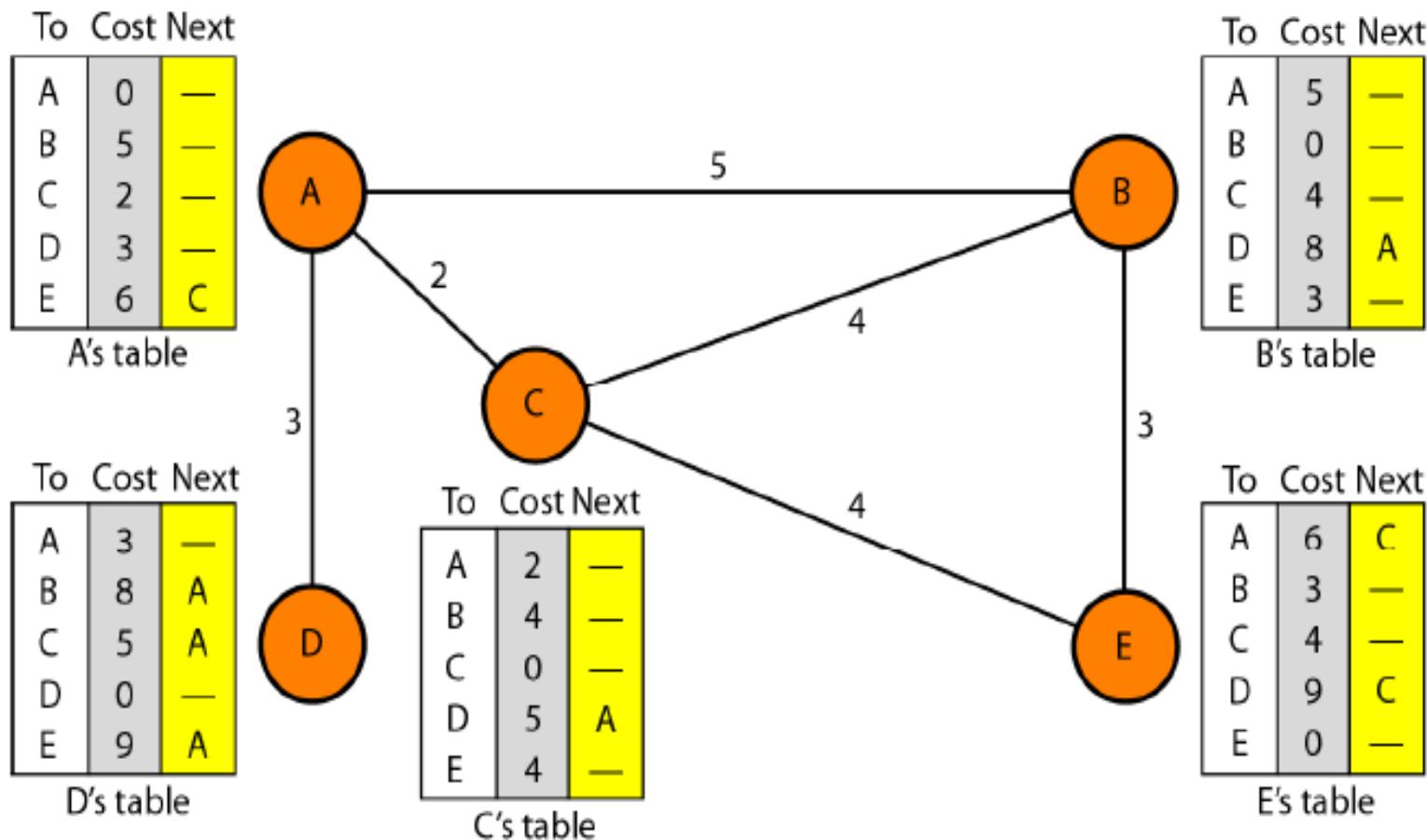
# Delivery, Forwarding, and Routing

## Distance Vector Routing

In distance vector routing, the least-cost route between any two nodes is the route with minimum distance. In this protocol, as the name implies, each node maintains a vector (table) of minimum distances to every node. The table at each node also guides the packets to the desired node by showing the next hop in the route (next-hop routing).

# Delivery, Forwarding, and Routing

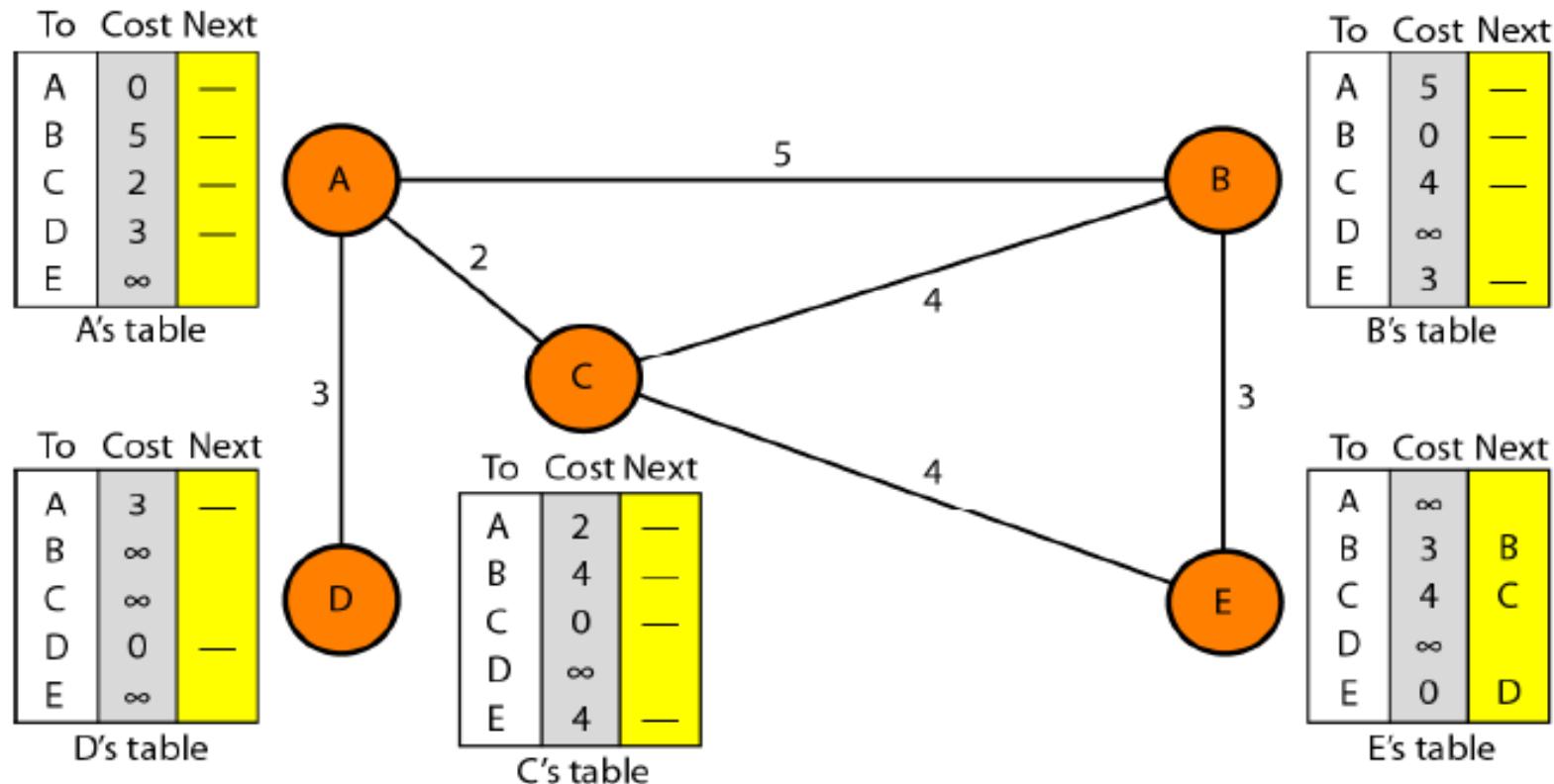
Consider a system of five nodes with their corresponding tables.



# Delivery, Forwarding, and Routing

## Procedure to compute routing table

### Initialization



# Delivery, Forwarding, and Routing

## Sharing

In distance vector routing, each node shares its routing table with its immediate neighbors periodically and when there is a change.

When the neighbor receives a table, third column needs to be replaced with the sender's name. If any of the rows can be used, the next node is the sender of the table. A node therefore can send only the first two columns of its table to any neighbor.

In other words, sharing here means sharing only the first two columns.

# Delivery, Forwarding, and Routing

## Updating

When a node receives a two-column table from a neighbor, it needs to update its routing table. Updating takes three steps:

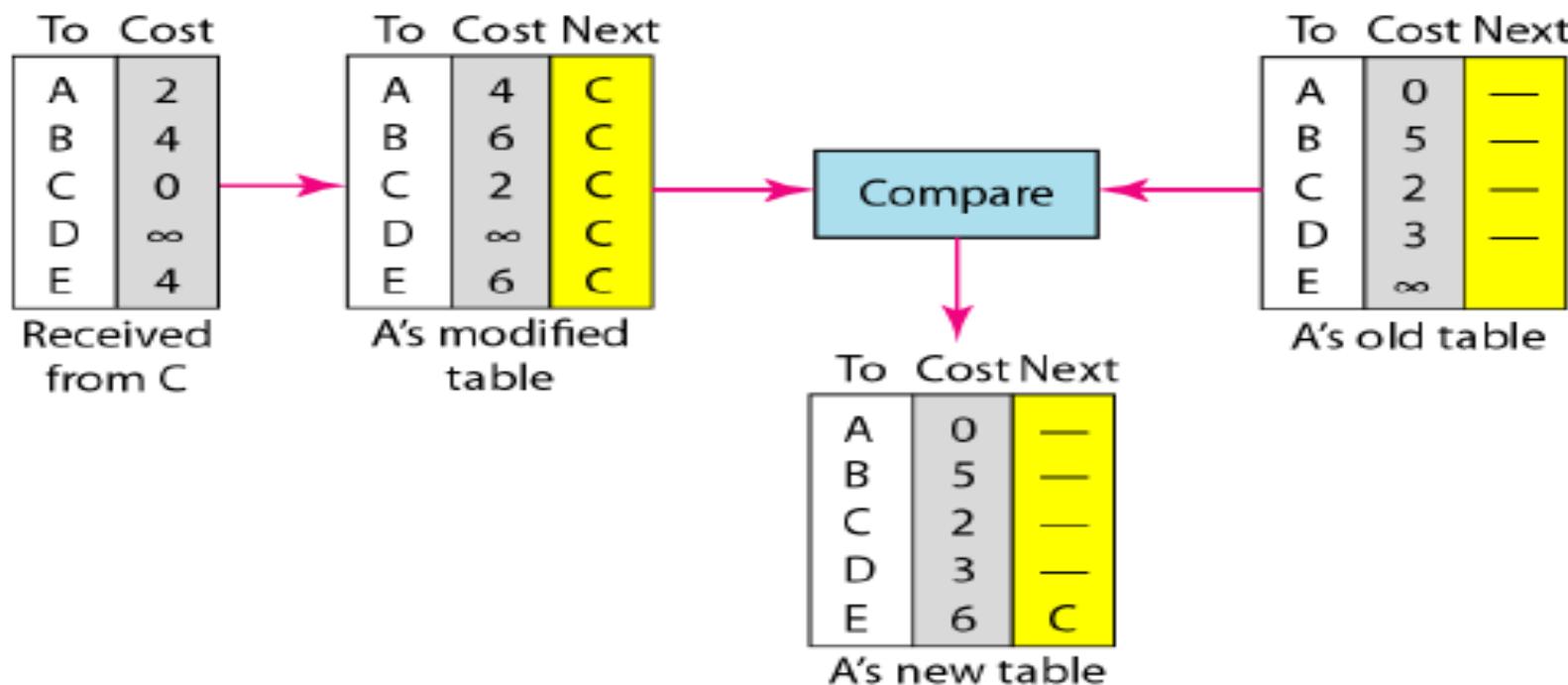
1. The receiving node needs to add the cost between itself and the sending node to each value in the second column.
2. The receiving node needs to add the name of the sending node to each row as the third column if the receiving node uses information from any row. The sending node is the next node in the route.

# Delivery, Forwarding, and Routing

3. The receiving node needs to compare each row of its old table with the corresponding row of the modified version of the received table.
  - a. If the next-node entry is different, the receiving node chooses the row with the smaller cost. If there is a tie, the old one is kept.
  - b. If the next-node entry is the same, the receiving node chooses the new row.

# Delivery, Forwarding, and Routing

Following figure shows how node A updates its routing table after receiving the partial table from node C.



# Delivery, Forwarding, and Routing

## When to Share

The question now is, When does a node send its partial routing table (only two columns) to all its immediate neighbors? The table is sent both periodically and when there is a change in the table.

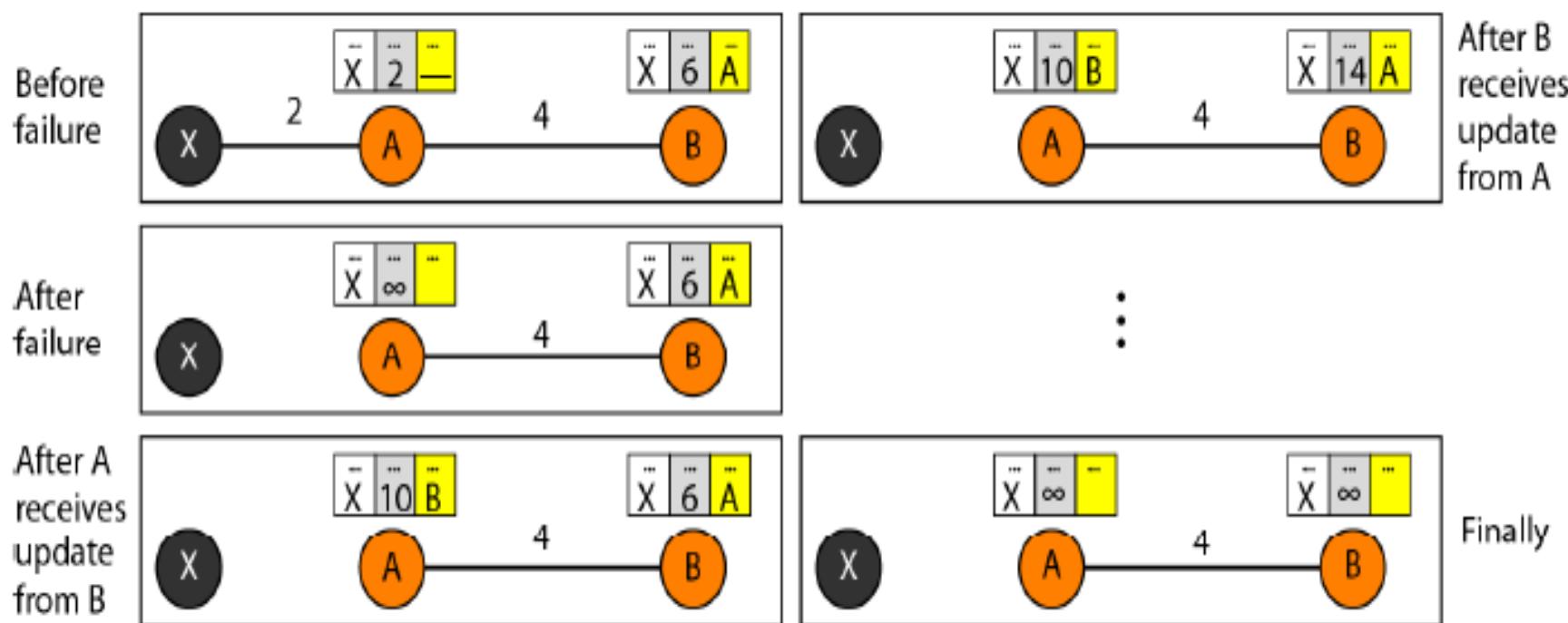
**Periodic Update:** A node sends its routing table, normally every 30 s, in a periodic update. The period depends on the protocol that is using distance vector routing.

**Triggered Update:** A node sends its two-column routing table to its neighbors anytime there is a change in its routing table. This is called a triggered update.

# Delivery, Forwarding, and Routing

## Two-Node Loop Instability (Count to infinity problem)

A problem with distance vector routing is instability, which means that a network using this protocol can become unstable. To understand the problem, we consider the following figure:-



# Delivery, Forwarding, and Routing

- In this figure, at the beginning, both nodes A and B know how to reach node X. But suddenly, the link between A and X fails. Node A changes its table.
- If A can send its table to B immediately, everything is fine.
- However, the system becomes unstable if B sends its routing table to A before receiving A's routing table. Node A receives the update and, assuming that B has found a way to reach X, immediately updates its routing table.
- Based on the triggered update strategy, A sends its new update to B. Now B thinks that something has been changed around A and updates its routing table. The cost of reaching X increases gradually until it reaches infinity. At this moment, both A and B know that X cannot be reached. However, during this time the system is not stable. Node A thinks that the route to X is via B; node B thinks that the route to X is via A. If A receives a packet destined for X, it goes to B and then comes back to A. Similarly, if B receives a packet destined for X, it goes to A and comes back to B. Packets bounce between A and B, creating a two-node loop problem.

# Delivery, Forwarding, and Routing

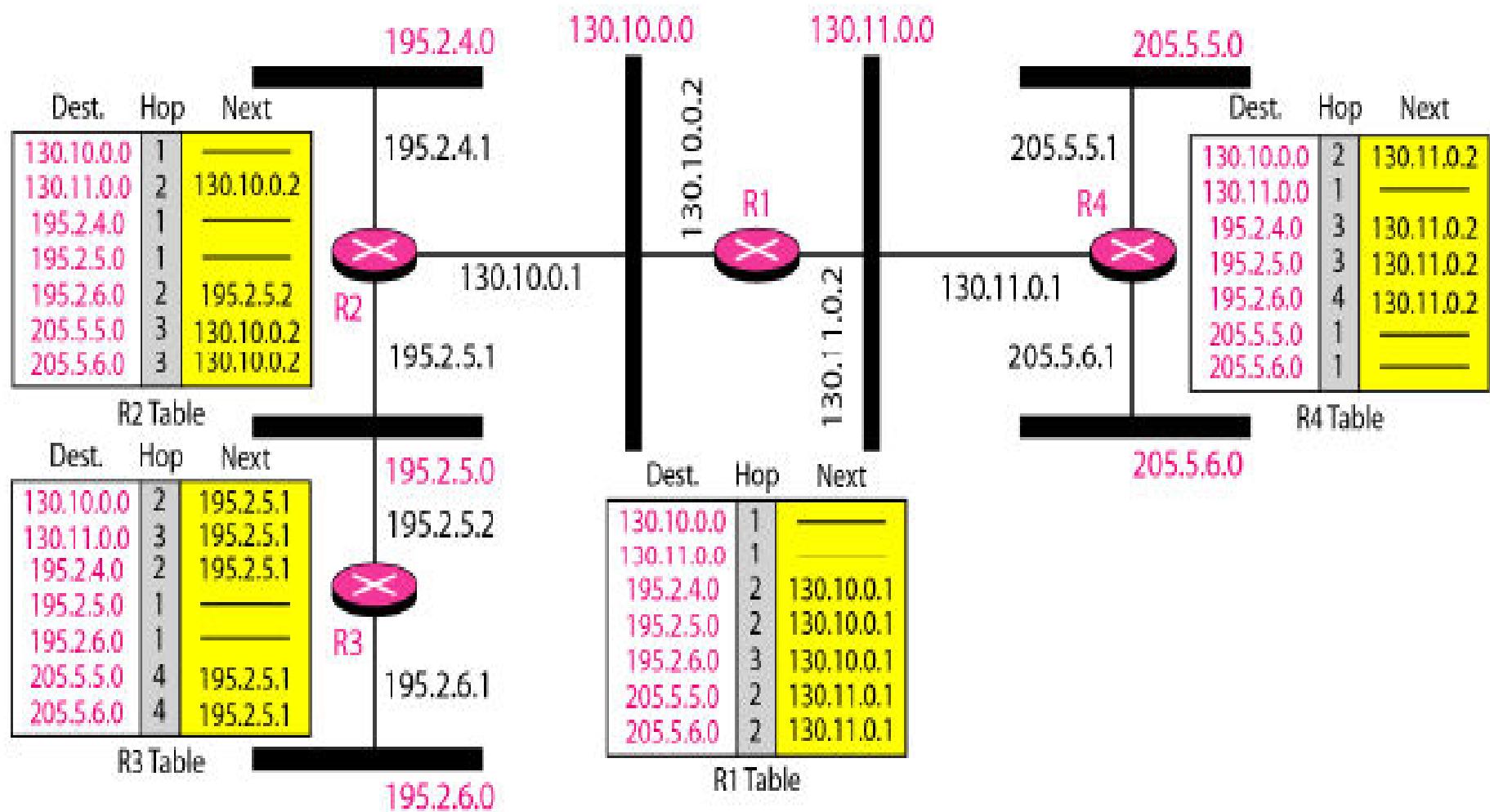
## The Routing Information Protocol (RIP)

It is an intra domain routing protocol used inside an autonomous system. It is a very simple protocol based on distance vector routing. RIP implements distance vector routing directly with some considerations:

1. In an autonomous system, we are dealing with routers and networks (links). The routers have routing tables; networks do not.
2. The destination in a routing table is a network, which means the first column defines a network address.
3. The metric used by RIP is very simple; the distance is defined as the number of links (networks) to reach the destination. For this reason, the metric in RIP is called a hop count.
4. Infinity is defined as 16, which means that any route in an autonomous system using RIP cannot have more than 15 hops.
5. The next-node column defines the address of the router to which the packet is to be sent to reach its destination.

# Delivery, Forwarding, and Routing

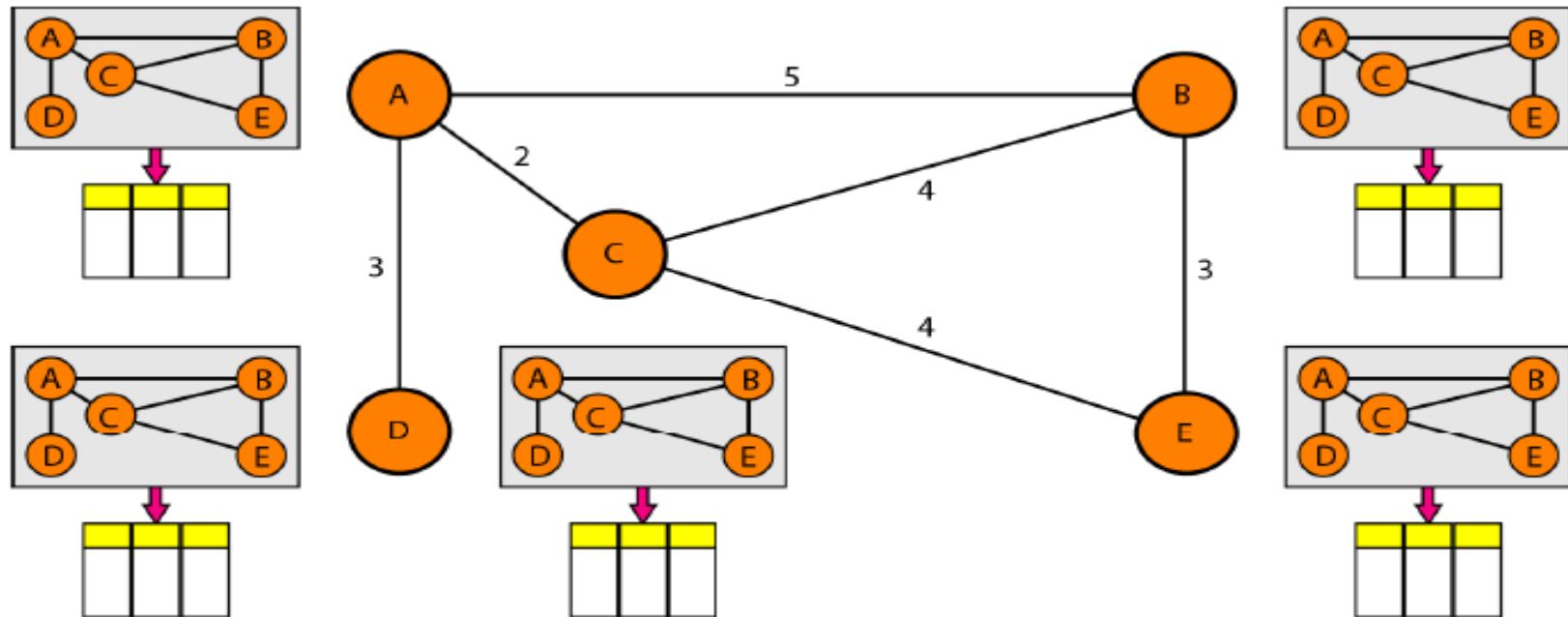
Following figure shows an autonomous system with seven networks and four routers. The table of each router is also shown.



# Delivery, Forwarding, and Routing

## Link State Routing

In link state routing, each node in the domain has the entire topology of the domain i.e. the list of nodes and links, how they are connected including the type, cost (metric), and condition of the links (up or down). The node can use Dijkstra's algorithm to build a routing table.



# Delivery, Forwarding, and Routing

- ❖ Each node uses the same topology to create a routing table, but the routing table for each node is unique because the calculations are based on different interpretations of the topology.
- ❖ The topology must be dynamic, representing the latest state of each node and each link. If there are changes in any point in the network (a link is down, for example), the topology must be updated for each node.

# Delivery, Forwarding, and Routing

## Building Routing Tables

In link state routing, four sets of actions are required to ensure that each node has the routing table showing the least-cost node to every other node.

1. Creation of the states of the links by each node, called the link state packet (LSP).
2. Dissemination of LSPs to every other router, called flooding, in an efficient and reliable way.
3. Formation of a shortest path tree for each node.
4. Calculation of a routing table based on the shortest path tree.

# Delivery, Forwarding, and Routing

## Creation of Link State Packet (LSP)

A link state packet can carry a large amount of information. For the moment, however, we assume that it carries a minimum amount of data: the node identity, the list of links, a sequence number, and age. The first two, node identity and the list of links, are needed to make the topology. The third, sequence number, facilitates flooding and distinguishes new LSPs from old ones. The fourth, age, prevents old LSPs from remaining in the domain for a long time. LSPs are generated on two occasions:

1. When there is a change in the topology of the domain.
2. On a periodic basis.

# Delivery, Forwarding, and Routing

## Flooding of LSPs

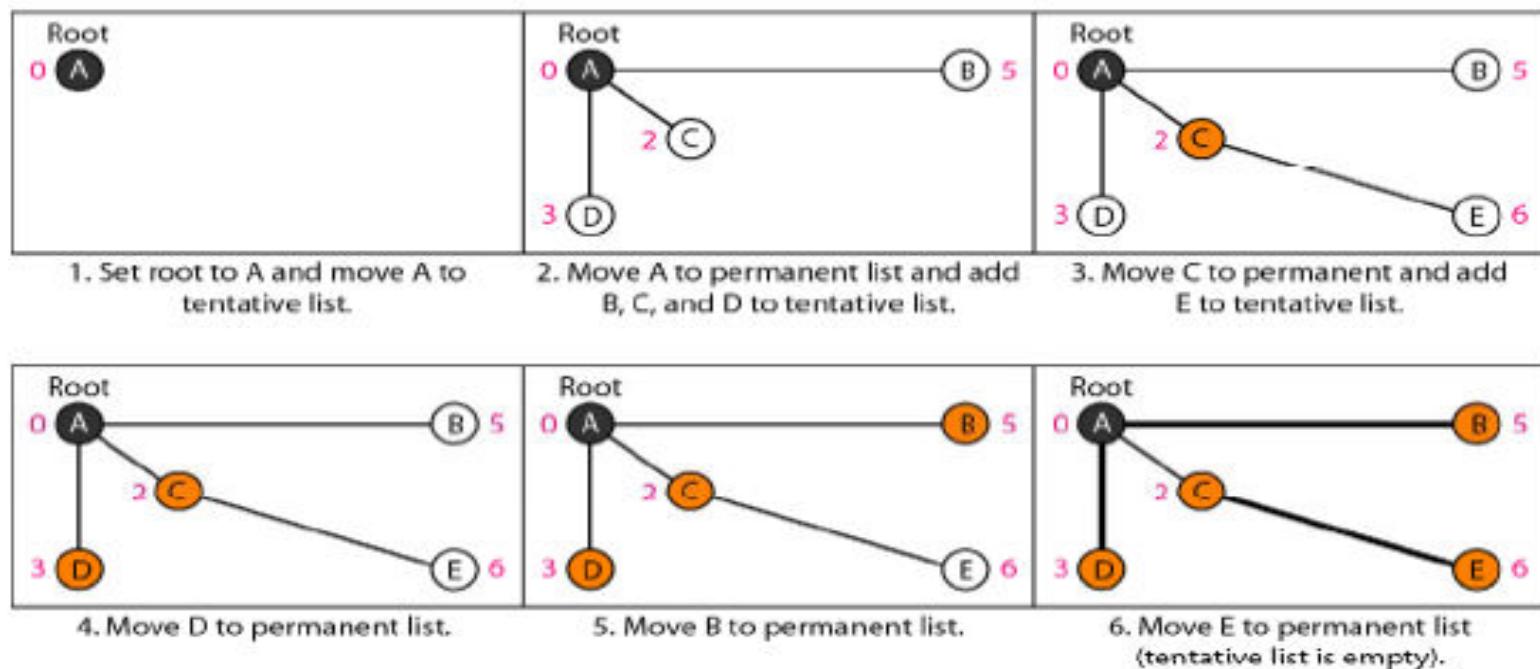
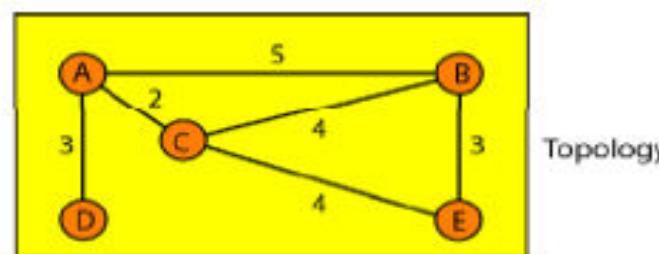
After a node has prepared an LSP, it must be disseminated to all other nodes, not only to its neighbors. The process is called flooding and based on the following:

1. The creating node sends a copy of the LSP out of each interface.
2. A node that receives an LSP compares it with the copy it may already have. If the newly arrived LSP is older than the one it has, it discards the LSP. If it is newer, the node does the following:
  - a. It discards the old LSP and keeps the new one.
  - b. It sends a copy of it out of each interface except the one from which the packet arrived. This guarantees that flooding stops somewhere in the domain (where a node has only one interface).

# Delivery, Forwarding, and Routing

## Formation of Shortest Path Tree:

After receiving all LSPs, each node will have a copy of the whole topology. Using Dijkstra algorithm, we create shortest path tree at each node.



# Delivery, Forwarding, and Routing

## Calculation of Routing Table from Shortest Path Tree

Each node uses the shortest path tree protocol to construct its routing table. The routing table shows the cost of reaching each node from the root. Following table shows the routing table for node A.

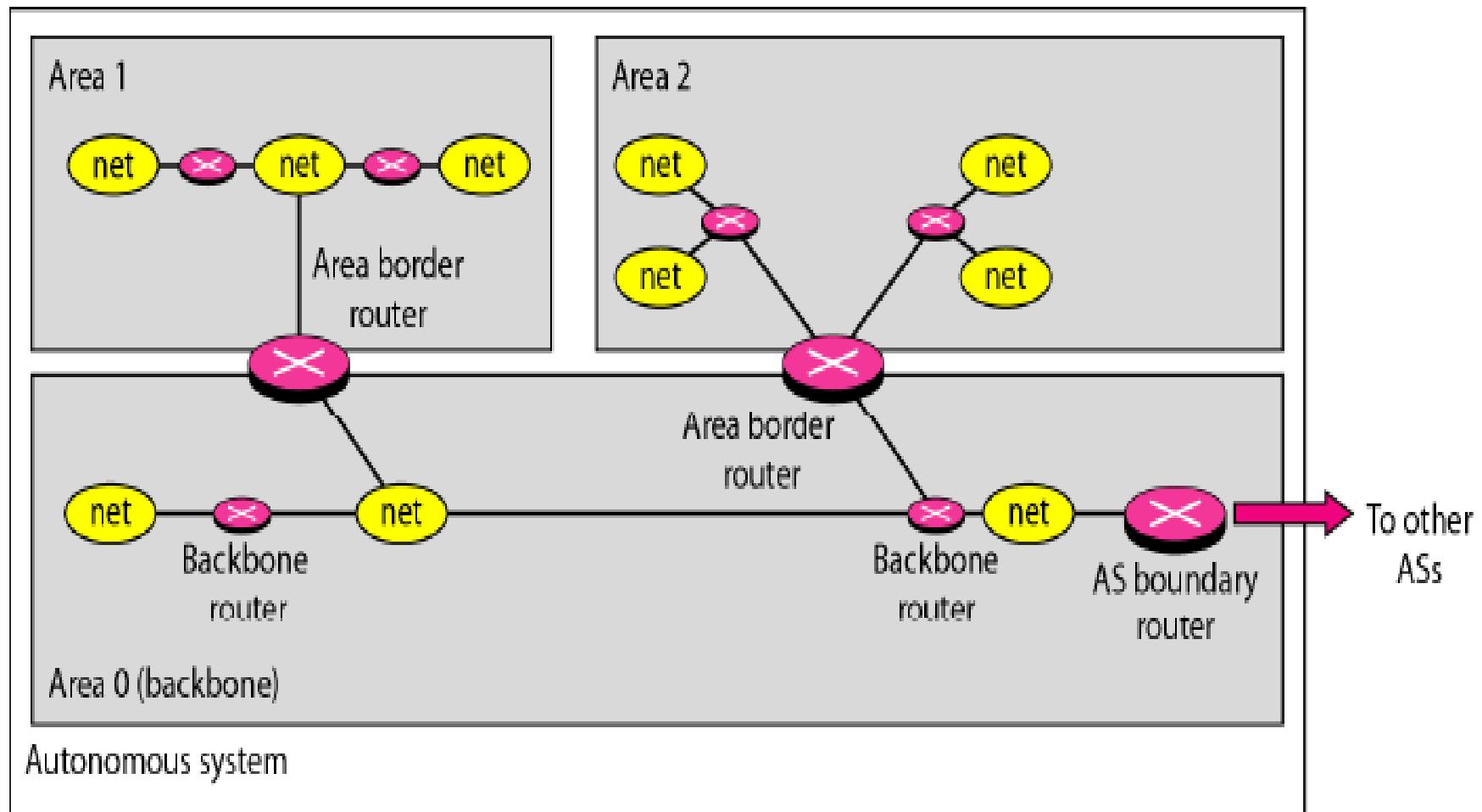
<i>Node</i>	<i>Cost</i>	<i>Next Router</i>
A	0	—
B	5	—
C	2	—
D	3	—
E	6	C

# Delivery, Forwarding, and Routing

## **Open Shortest Path First(OSPF) Routing Protocol**

- ❖ The Open Shortest Path First or OSPF protocol is an intra domain routing protocol based on link state routing. Its domain is also an autonomous system.
  
- ❖ To handle routing efficiently and in a timely manner, OSPF divides an autonomous system into areas. An area is a collection of networks, hosts, and routers all contained within an autonomous system. All networks inside an area must be connected.

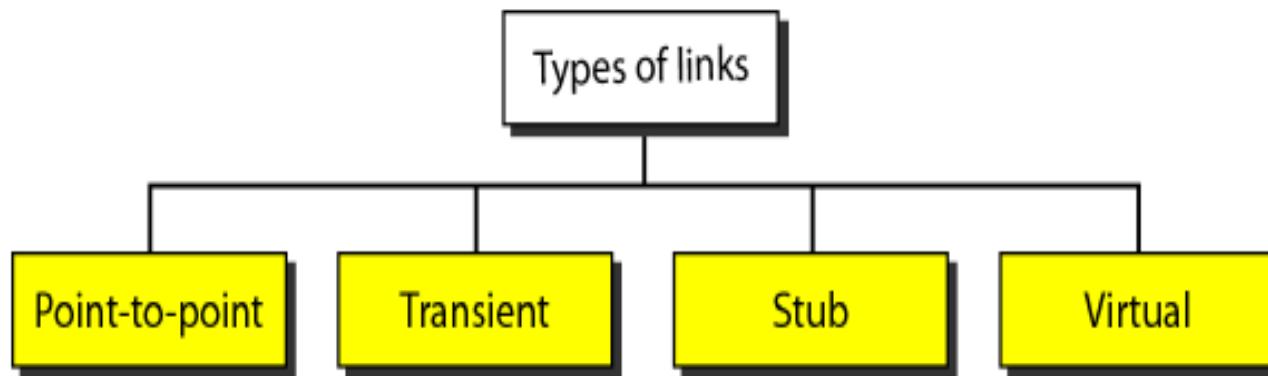
# Delivery, Forwarding, and Routing



# Delivery, Forwarding, and Routing

**Metric:** The OSPF protocol allows the administrator to assign a cost, called the metric, to each route. The metric can be based on a type of service (minimum delay, maximum throughput, and so on).

**Types of Links:** In OSPF terminology, a connection is called a link. Four types of links have been defined: point-to-point, transient, stub, and virtual.



# Delivery, Forwarding, and Routing

## **Point-to-Point link**

A point-to-point link connects two routers without any other host or router in between. There is no need to assign a network address to this type of link.

## **Transient link**

A transient link is a network with several routers attached to it. The data can enter through any of the routers and leave through any router. All LANs and some WANs with two or more routers are of this type. In this case, each router has many neighbors.

## **Stub link**

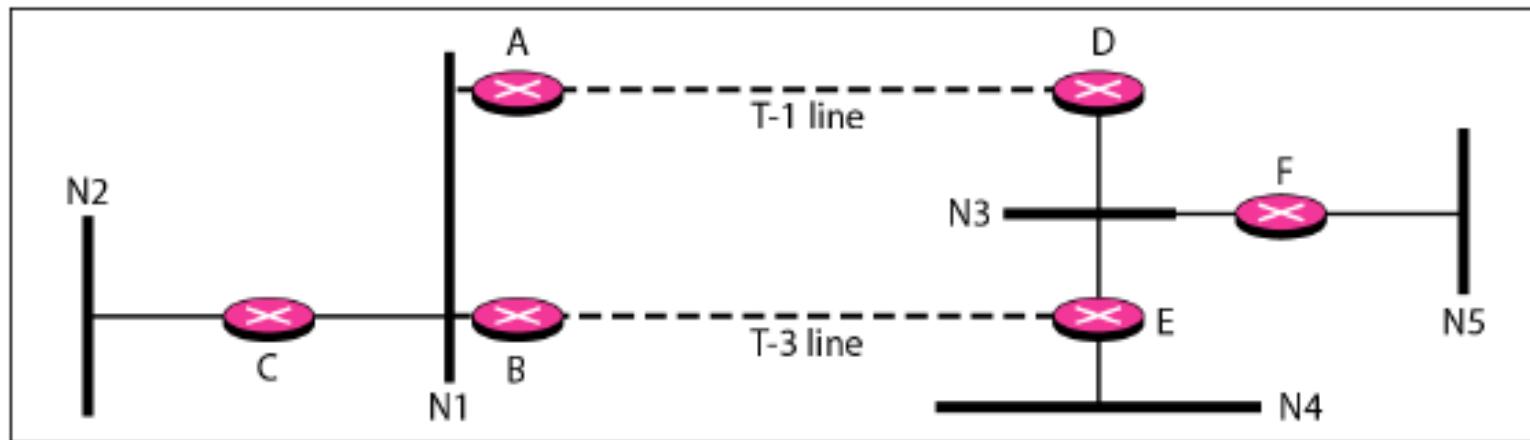
A stub link is a network that is connected to only one router. The data packets enter the network through this single router and leave the network through this same router.

## **Virtual link**

When the link between two routers is broken, the administration may create a virtual link between them, using a longer path that probably goes through several routers.

# Delivery, Forwarding, and Routing

## Example of an AS and its graphical representation in OSPF



a. Autonomous system



b. Graphical representation

# AKTU Examination Questions

1. If a class B network on the Internet has a subnet mask of 255.255.248.0, what is the maximum number of hosts per subnet?
2. What is count-to-infinity problem?
3. What is time-to-live or packet lifetime?
4. What is unicast routing? Discuss unicast routing protocols.
5. Write advantages of Next-generation IPV6 over IPV4.
6. The IP network 200.198.160.0 is using subnet mask 255.255.255.224. Design the subnets.

# AKTU Examination Questions

7. Write two use of subnet mask.
8. Convert the IPv4 address whose hexadecimal representation is C22F15B2 to dotted decimal notation. What is the class of this address?
9. What do you mean by adaptive and non-adaptive routing algorithm? Discuss Distance Vector Routing including count to infinity problem.
10. Sketch the IP header neatly and explain the functions of each field. What are the deficiencies of IPV4 over IPV6?
11. An organization is granted a block 211.17.180.0 /24. The administrator wants to create 32 subnets.
  - i) Find the subnet mask.
  - ii) Find the number of addresses in each subnet.
  - iii) Find the first & last address in subnet 1.
  - iv) Find the first & last address in subnet 32.

# AKTU Examination Questions

12. Given the IP address 180.25.21.172 and the subnet mask 255.255.192.0, what is the subnet address?
13. What is IP addressing? How it is classified? How is subnet addressing is performed?
14. What is unicast routing? Discuss unicast routing protocols.
15. With the given IP-address, how will you extract its net-id and host-id?
16. Describe the problem of count to infinity associated with distance vector routing technique.
17. Given the IP address 180.2 5.21 .172 and the subnet mask 255.255.192.A, what is the subnet address?

# AKTU Examination Questions

18. What is the net mask of the gateway interface in a subnetwork where maximum of 25 hosts exist and IP address of one of the hosts is 192.168.1.1 ?
19. Define routing. In what way it is different from switching?
20. What is unicast routing? Discuss unicast routing protocols.
21. Find the class of each address
  - (a) 140.213.10.80
  - (b) 52.15.150.11
22. What is the type of the following address?
  - (a) 4F::A234:2
  - (b) 52F::1234:2222

# Computer Network

Lecture taken by  
Dharmendra Kumar  
(Associate Professor)

United College of Engineering and Research, Prayagraj

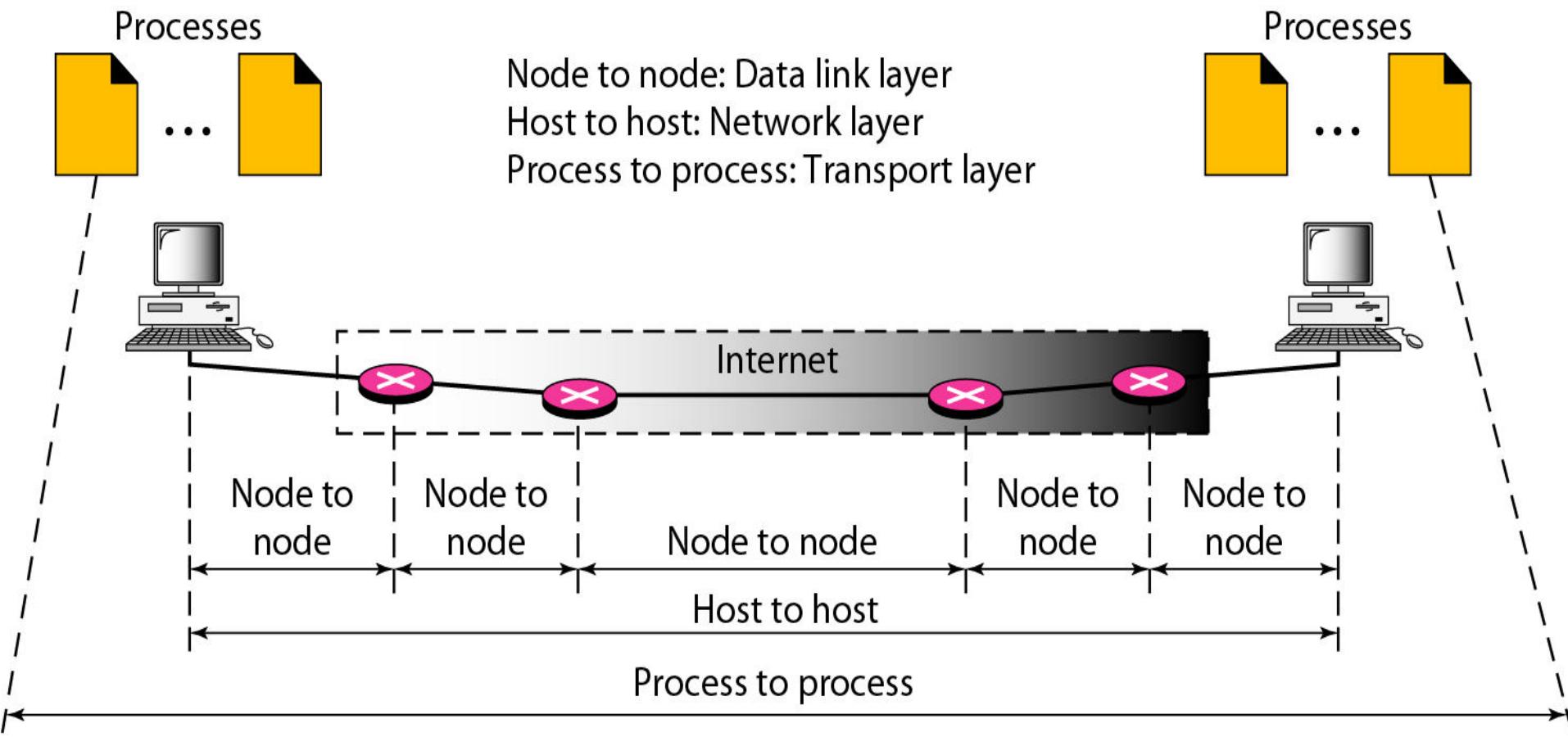
# **Unit-4**

# **Transport Layer**

# **Process-to-Process Delivery**

# Process-to-Process Delivery

The transport layer is responsible for process-to-process delivery—the delivery of a packet, part of a message, from one process to another.



# Process-to-Process Delivery

## Client/Server Paradigm

- ❖ A process on the local host, called a client, needs services from a process usually on the remote host, called a server.
- ❖ Both processes (client and server) have the same name. For example, to get the day and time from a remote machine, we need a Daytime client process running on the local host and a Daytime server process running on a remote machine.

# Process-to-Process Delivery

## Addressing

- ❖ At the transport layer, we need a transport layer address, called a **port number**, to choose among multiple processes running on the destination host. The destination port number is needed for delivery; the source port number is needed for the reply.
- ❖ In the Internet model, the port numbers are 16-bit integers between 0 and 65,535.

# Process-to-Process Delivery

## IANA Ranges

The IANA (Internet Assigned Number Authority) has divided the port numbers into three ranges: well known, registered, and dynamic (or private).

**Well-known ports:** The ports ranging from 0 to 1023 are assigned and controlled by IANA. These are the well-known ports.

**Registered ports:** The ports ranging from 1024 to 49,151 are not assigned or controlled by IANA. They can only be registered with IANA to prevent duplication.

**Dynamic ports:** The ports ranging from 49,152 to 65,535 are neither controlled nor registered. They can be used by any process. These are the ephemeral ports.

# Process-to-Process Delivery

## Socket Addresses

Process-to-process delivery needs two identifiers, IP address and the port number, at each end to make a connection. The combination of an IP address and a port number is called a socket address.

The client socket address defines the client process uniquely just as the server socket address defines the server process uniquely.



# Process-to-Process Delivery

## Connectionless Versus Connection-Oriented Service

A transport layer protocol can either be connectionless or connection-oriented.

### Connectionless Service

In a connectionless service, the packets are sent from one party to another with no need for connection establishment or connection release. The packets are not numbered; they may be delayed or lost or may arrive out of sequence. There is no acknowledgment.

### Connection-Oriented Service

In a connection-oriented service, a connection is first established between the sender and the receiver. Data are transferred. At the end, the connection is released.

# Process-to-Process Delivery

## Reliable Versus Unreliable

- ❖ The transport layer service can be reliable or unreliable. If the application layer program needs reliability, we use a reliable transport layer protocol by implementing flow and error control at the transport layer.
- ❖ On the other hand, if the application program does not need reliability because it uses its own flow and error control mechanism or it needs fast service or the nature of the service does not demand flow and error control (real-time applications), then an unreliable protocol can be used.

# Process-to-Process Delivery

- ❖ In the Internet, there are three different transport layer protocols, UDP, TCP and SCTP.
- ❖ UDP is connectionless and unreliable;
- ❖ TCP and SCTP are connection-oriented and reliable.

# Process-to-Process Delivery

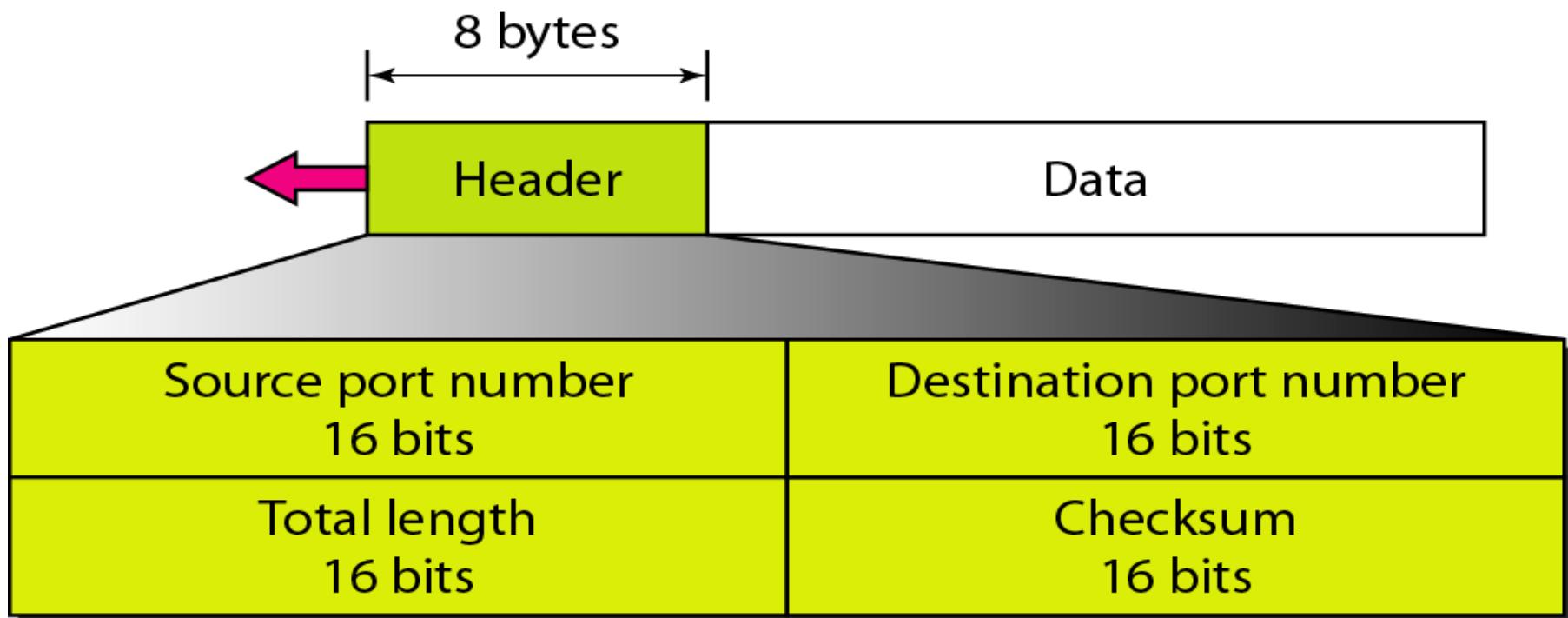
## USER DATAGRAM PROTOCOL (UDP)

- ❖ The User Datagram Protocol (UDP) is called a connectionless, unreliable transport protocol. It does not add anything to the services of IP except to provide process-to-process communication instead of host-to-host communication. Also, it performs very limited error checking.
- ❖ UDP is a very simple protocol using a minimum of overhead. If a process wants to send a small message and does not care much about reliability, it can use UDP. Sending a small message by using UDP takes much less interaction between the sender and receiver than using TCP or SCTP.

# Process-to-Process Delivery

## User Datagram

UDP packets, called user datagrams, have a fixed-size header of 8 bytes. Following figure shows the format of a user datagram.



# Process-to-Process Delivery

All the fields are explained as following:-

## Source port number:

This is the port number used by the process running on the source host. It is 16 bits long, which means that the port number can range from 0 to 65,535. If the source host is the client (a client sending a request), the port number, in most cases, is an **ephemeral port number** requested by the process and chosen by the UDP software running on the source host. If the source host is the server (a server sending a response), the port number, in most cases, is a **well-known port number**.

# Process-to-Process Delivery

**Destination port number:** This is the port number used by the process running on the destination host. It is also 16 bits long. If the destination host is the server (a client sending a request), the port number, in most cases, is a well-known port number. If the destination host is the client (a server sending a response), the port number, in most cases, is an ephemeral port number. In this case, the server copies the ephemeral port number it has received in the request packet.

**Total length:** This is a 16-bit field that defines the total length of the user datagram, header plus data. The 16 bits can define a total length of 0 to 65,535 bytes.

UDP datagram length = IP datagram length – IP header length

**Checksum:** This field is used to detect errors over the entire user datagram (header plus data).

# Process-to-Process Delivery

## UDP Operation

### Connectionless Services

- ❖ UDP provides a connectionless service. This means that each user datagram sent by UDP is an independent datagram.
- ❖ There is no relationship between the different user datagrams even if they are coming from the same source process and going to the same destination program.
- ❖ The user datagrams are not numbered.
- ❖ Also, there is no connection establishment and no connection termination, as is the case for TCP. This means that each user datagram can travel on a different path.
- ❖ Only those processes sending short messages should use UDP.

# Process-to-Process Delivery

## Flow and Error Control

- ❖ UDP is a very simple, unreliable transport protocol. There is no flow control and hence no window mechanism. The receiver may overflow with incoming messages.
- ❖ There is no error control mechanism in UDP except for the checksum. This means that the sender does not know if a message has been lost or duplicated. When the receiver detects an error through the checksum, the user datagram is silently discarded.

# Process-to-Process Delivery

## Use of UDP

The following are some uses of the UDP protocol:

- ❖ UDP is suitable for a process that requires simple request-response communication with little concern for flow and error control.
- ❖ UDP is suitable for a process with internal flow and error control mechanisms. For example, the Trivial File Transfer Protocol (TFTP) process includes flow and error control. It can easily use UDP.
- ❖ UDP is a suitable transport protocol for multicasting. Multicasting capability is embedded in the UDP software but not in the TCP software.
- ❖ UDP is used for management processes such as SNMP.
- ❖ UDP is used for some route updating protocols such as Routing Information Protocol(RIP).

# Process-to-Process Delivery

## Transmission Control Protocol(TCP)

- ❖ Unlike UDP, TCP is a connection oriented protocol; it creates a virtual connection between two TCPs to send data. In addition, TCP uses flow and error control mechanisms at the transport level.
- ❖ TCP is called a connection-oriented, reliable transport protocol. It adds connection-oriented and reliability features to the services of IP.

# Process-to-Process Delivery

## TCP Services

Following are the services offered by TCP to the processes at the application layer.

### Process-to-Process Communication

Like UDP, TCP provides process-to-process communication using port numbers.

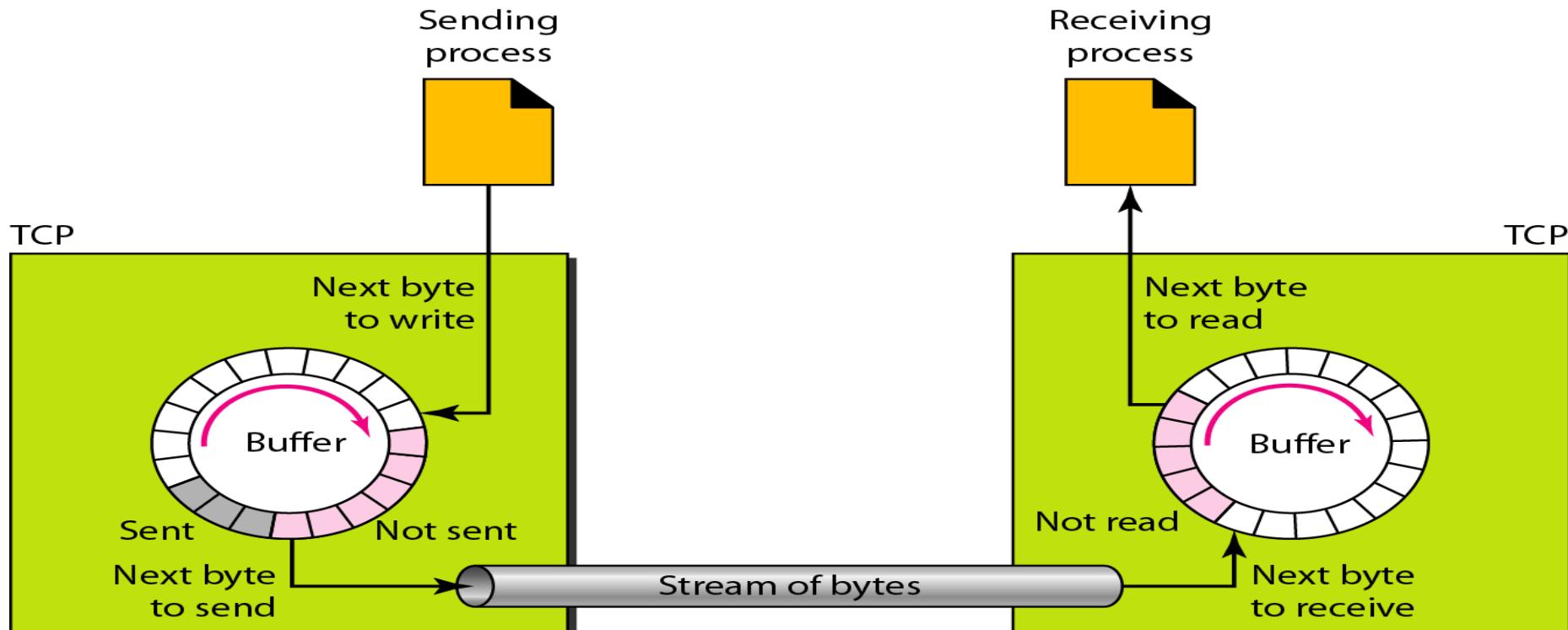
### Stream Delivery Service

- ❖ TCP, unlike UDP, is a stream-oriented protocol.
- ❖ TCP allows the sending process to deliver data as a stream of bytes and allows the receiving process to obtain data as a stream of bytes.
- ❖ TCP creates an environment in which the two processes seem to be connected by an imaginary "tube" that carries their data across the Internet.
- ❖ The sending process produces (writes to) the stream of bytes, and the receiving process consumes (reads from) them.

# Process-to-Process Delivery

## Sending and Receiving Buffers

Because the sending and the receiving processes may not write or read data at the same speed, TCP needs buffers for storage. There are two buffers, the sending buffer and the receiving buffer, one for each direction.



# Process-to-Process Delivery

- ❖ At the sending site, the buffer has three types of chambers. The white section contains empty chambers that can be filled by the sending process (producer). The gray area holds bytes that have been sent but not yet acknowledged. TCP keeps these bytes in the buffer until it receives an acknowledgment. The colored area contains bytes to be sent by the sending TCP. After the bytes in the gray chambers are acknowledged, the chambers are recycled and available for use by the sending process.
- ❖ The operation of the buffer at the receiver site is simpler. The circular buffer is divided into two areas (shown as white and colored). The white area contains empty chambers to be filled by bytes received from the network. The colored sections contain received bytes that can be read by the receiving process. When a byte is read by the receiving process, the chamber is recycled and added to the pool of empty chambers.

# Process-to-Process Delivery

## Segments

At the transport layer, TCP groups a number of bytes together into a packet called a segment. TCP adds a header to each segment (for control purposes) and delivers the segment to the IP layer for transmission. The segments are encapsulated in IP datagrams and transmitted.

## Full-Duplex Communication

TCP offers full-duplex service, in which data can flow in both directions at the same time. Each TCP then has a sending and receiving buffer, and segments move in both directions.

# Process-to-Process Delivery

## Connection-Oriented Service

TCP, unlike UDP, is a connection-oriented protocol.

## Reliable Service

TCP is a reliable transport protocol. It uses an acknowledgment mechanism to check the safe and sound arrival of data.

# Process-to-Process Delivery

## TCP Features

To provide the services mentioned in the previous section, TCP has several features.

### Numbering System

To keeps track of the segments being transmitted or received, TCP uses two fields called the sequence number and the acknowledgment number. These two fields refer to the byte number and not the segment number.

### Byte Number

TCP numbers all data bytes that are transmitted in a connection. Numbering is independent in each direction. When TCP receives bytes of data from a process, it stores them in the sending buffer and numbers them. The numbering does not necessarily start from 0. Instead, TCP generates a random number between 0 and  $2^{32} - 1$  for the number of the first byte. For example, if the random number happens to be 1057 and the total data to be sent are 6000 bytes, the bytes are numbered from 1057 to 7056.

# Process-to-Process Delivery

## Sequence Number

After the bytes have been numbered, TCP assigns a sequence number to each segment that is being sent. The sequence number for each segment is the number of the first byte carried in that segment.

## Acknowledgment Number

The acknowledgment number defines the number of the next byte that the party expects to receive. The acknowledgment number is cumulative.

# Process-to-Process Delivery

## Flow Control

TCP, unlike UDP, provides flow control. The receiver of the data controls the amount of data that are to be sent by the sender. This is done to prevent the receiver from being overwhelmed with data. The numbering system allows TCP to use a byte-oriented flow control.

## Error Control

To provide reliable service, TCP implements an error control mechanism.

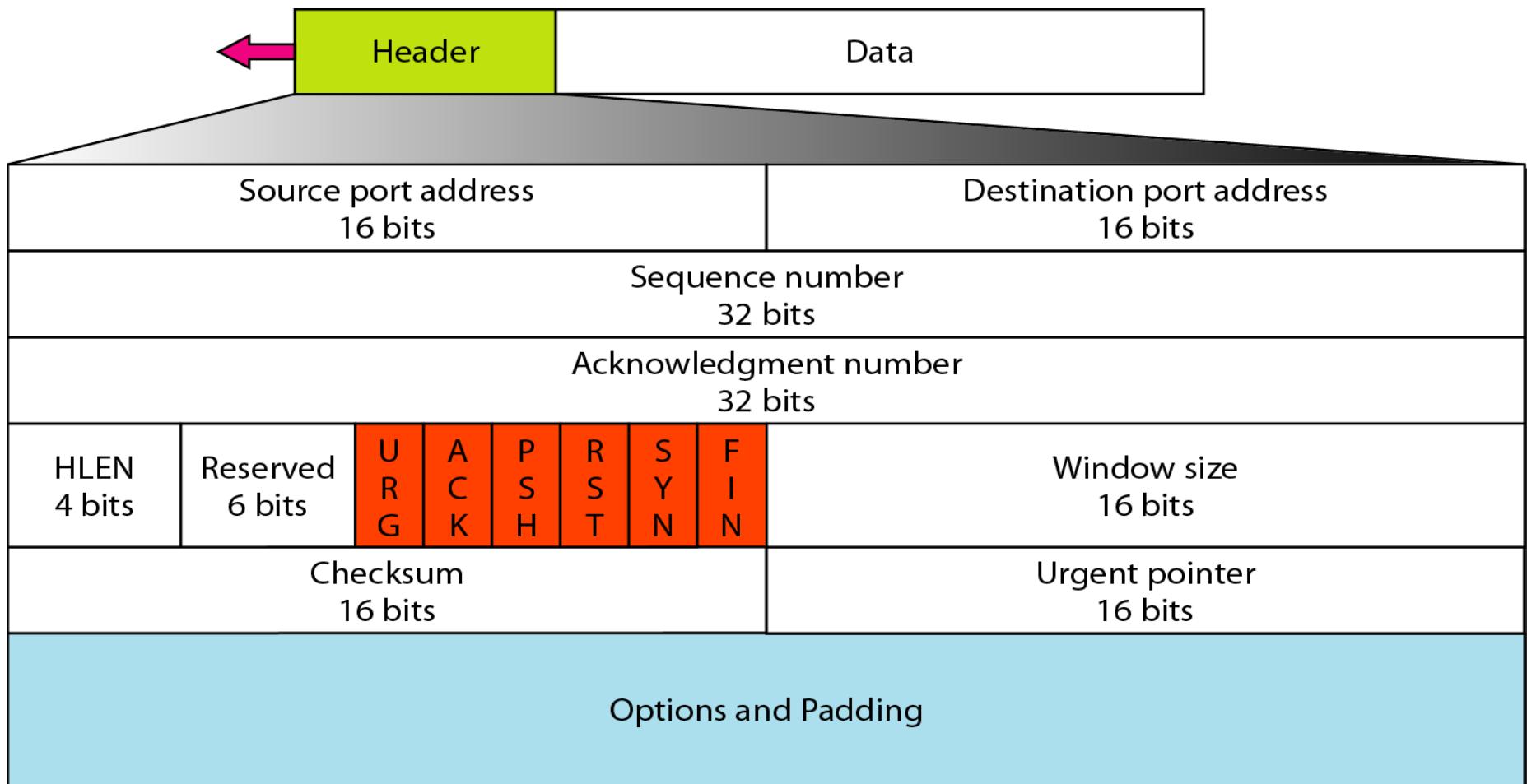
## Congestion Control

TCP, unlike UDP, takes into account congestion in the network. The amount of data sent by a sender is not only controlled by the receiver (flow control), but is also determined by the level of congestion in the network.

# Process-to-Process Delivery

## TCP segment format

The format of a segment is shown in the following figure:-



# Process-to-Process Delivery

The segment consists of a 20- to 60-byte header, followed by data from the application program. The header is 20 bytes if there are no options and up to 60 bytes if it contains options.

**Source port address:** This is a 16-bit field that defines the port number of the application program in the host that is sending the segment.

**Destination port address:** This is a 16-bit field that defines the port number of the application program in the host that is receiving the segment.

**Sequence number:** This 32-bit field defines the number assigned to the first byte of data contained in this segment.

**Acknowledgment number:** This 32-bit field defines the byte number that the receiver of the segment is expecting to receive from the other party. If the receiver of the segment has successfully received byte number  $x$  from the other party, it defines  $x + 1$  as the acknowledgment number. Acknowledgment and data can be piggybacked together.

# Process-to-Process Delivery

**Header length:** This 4-bit field indicates the number of 4-byte words in the TCP header. The length of the header can be between 20 and 60 bytes.

**Reserved:** This is a 6-bit field reserved for future use.

**Control:** This field defines 6 different control bits or flags as shown in figure. One or more of these bits can be set at a time.

URG: Urgent pointer is valid

ACK: Acknowledgment is valid

PSH: Request for push

RST: Reset the connection

SYN: Synchronize sequence numbers

FIN: Terminate the connection

URG	ACK	PSH	RST	SYN	FIN
-----	-----	-----	-----	-----	-----

# Process-to-Process Delivery

These bits enable flow control, connection establishment and termination, connection abortion, and the mode of data transfer in TCP.

<i>Flag</i>	<i>Description</i>
URG	The value of the urgent pointer field is valid.
ACK	The value of the acknowledgment field is valid.
PSH	Push the data.
RST	Reset the connection.
SYN	Synchronize sequence numbers during connection.
FIN	Terminate the connection.

# Process-to-Process Delivery

**Window size:** This field defines the size of the window, in bytes, that the other party must maintain. Note that the length of this field is 16 bits, which means that the maximum size of the window is 65,535 bytes. This value is normally referred to as the receiving window (rwnd) and is determined by the receiver. The sender must obey the dictation of the receiver in this case.

**Checksum:** This 16-bit field contains the checksum.

**Urgent pointer:** This 16-bit field, which is valid only if the urgent flag is set, is used when the segment contains urgent data. It defines the number that must be added to the sequence number to obtain the number of the last urgent byte in the data section of the segment.

**Options:** There can be up to 40 bytes of optional information in the TCP header.

# Process-to-Process Delivery

## TCP Connection

In TCP, connection-oriented transmission requires three phases: connection establishment, data transfer, and connection termination.

## Connection Establishment

TCP transmits data in full-duplex mode. When two TCPs in two machines are connected, they are able to send segments to each other simultaneously. This implies that each party must initialize communication and get approval from the other party before any data are transferred.

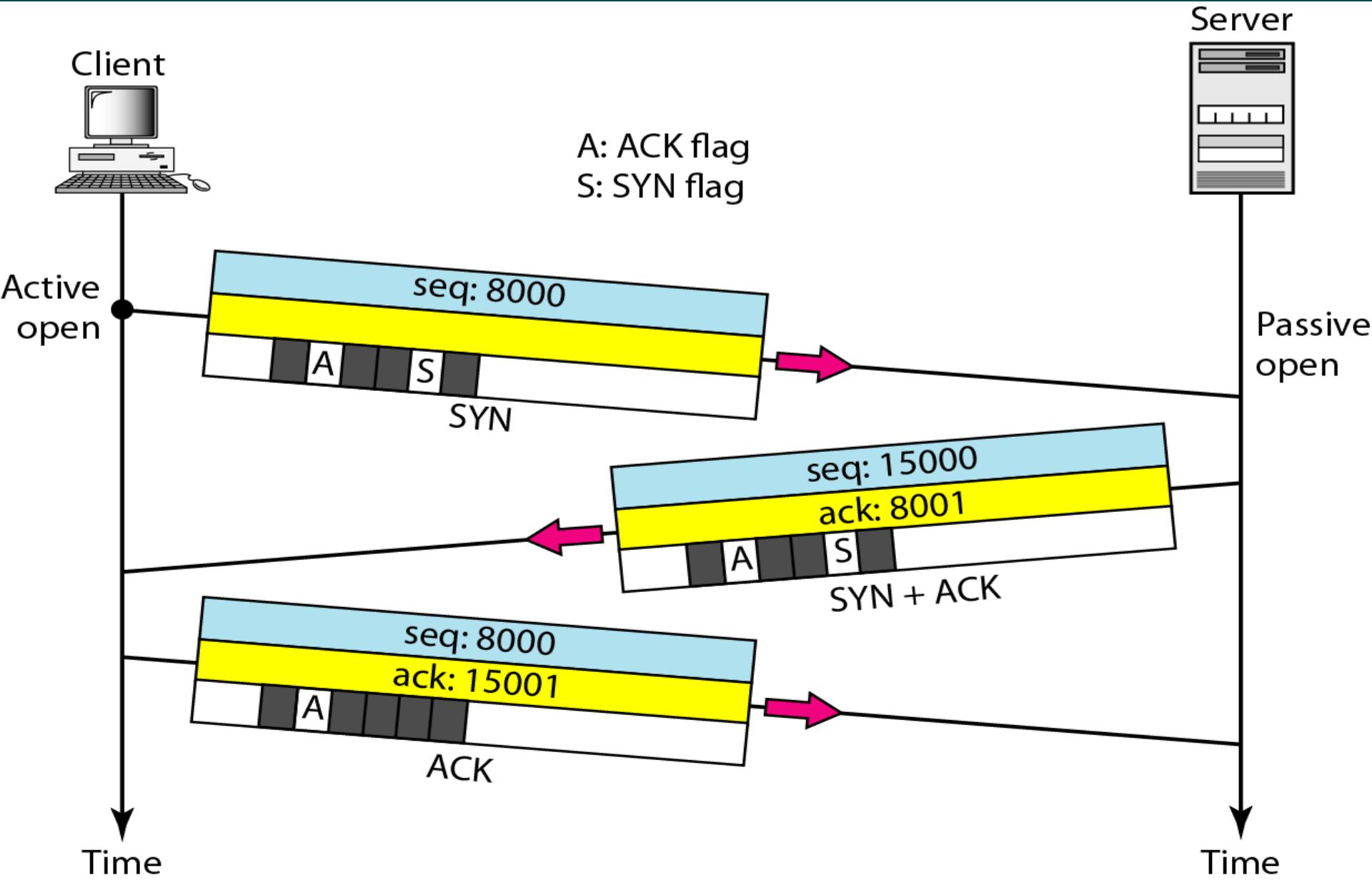
# Process-to-Process Delivery

## Three-Way Handshaking

The connection establishment in TCP is called three way handshaking. In our example, an application program, called the client, wants to make a connection with another application program, called the server, using TCP as the transport layer protocol.

The process starts with the server. The server program tells its TCP that it is ready to accept a connection. This is called a request for a passive open. Although the server TCP is ready to accept any connection from any machine in the world, it cannot make the connection itself.

# Process-to-Process Delivery



# Process-to-Process Delivery

The three steps in this phase are as follows:-

- (1) The client sends the first segment, a SYN segment, in which only the SYN flag is set. This segment is for synchronization of sequence numbers. It consumes one sequence number. When the data transfer starts, the sequence number is incremented by 1. We can say that the SYN segment carries no real data, but we can think of it as containing 1 imaginary byte.
- (2) The server sends the second segment, a SYN +ACK segment, with 2 flag bits set: SYN and ACK. This segment has a dual purpose. It is a SYN segment for communication in the other direction and serves as the acknowledgment for the SYN segment. It consumes one sequence number.

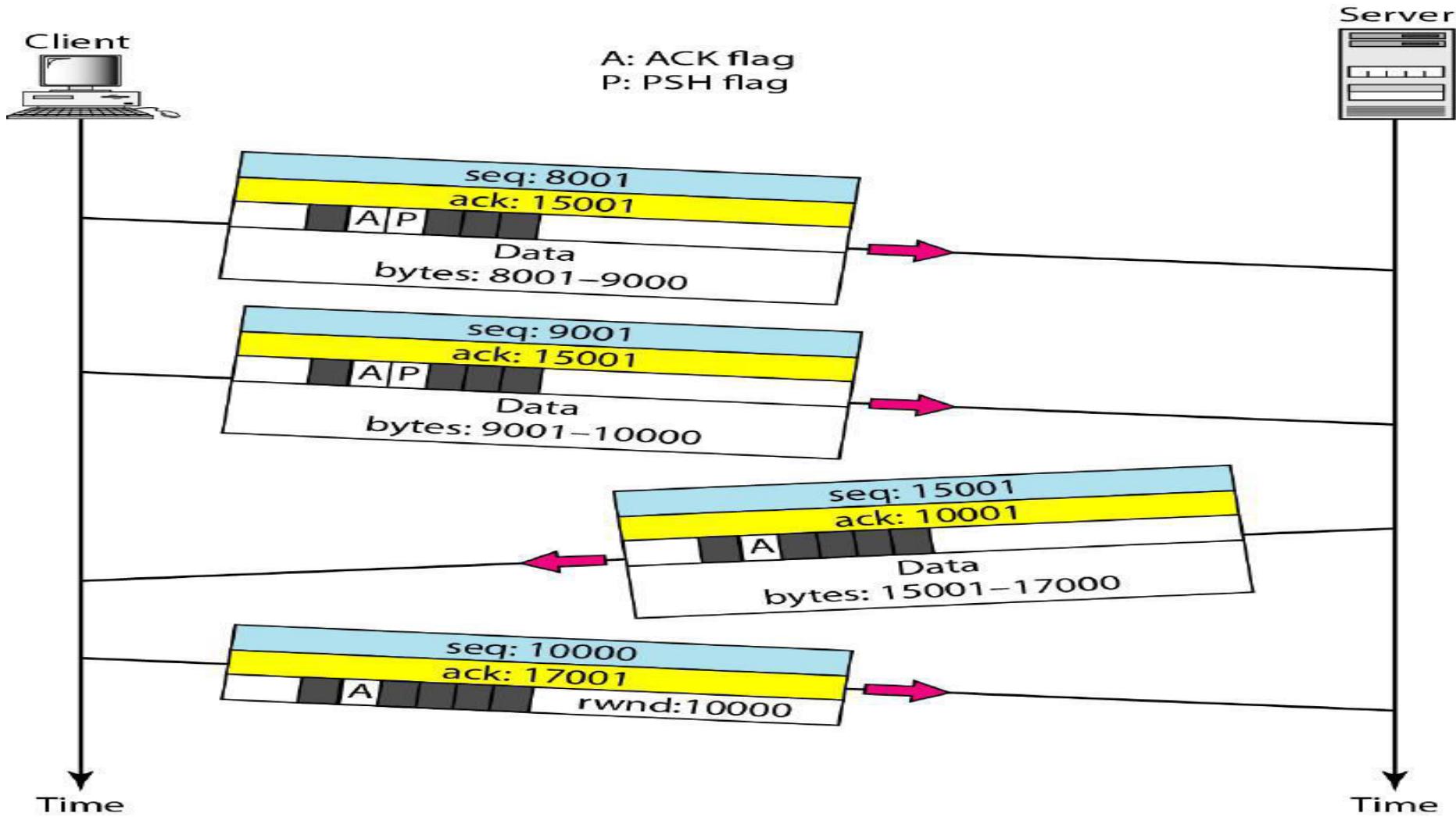
# Process-to-Process Delivery

(3) The client sends the third segment. This is just an ACK segment. It acknowledges the receipt of the second segment with the ACK flag and acknowledgment number field. Note that the sequence number in this segment is the same as the one in the SYN segment; the ACK segment does not consume any sequence numbers.

# Process-to-Process Delivery

## Data Transfer

This process is shown in the following figure:-



# Process-to-Process Delivery

In this figure, after connection is established , the client sends 2000 bytes of data in two segments. The server then sends 2000 bytes in one segment. The client sends one more segment. The first three segments carry both data and acknowledgment, but the last segment carries only an acknowledgment because there are no more data to be sent. The data segments sent by the client have the PSH (push) flag set so that the server TCP knows to deliver data to the server process as soon as they are received. The segment from the server, on the other hand, does not set the push flag.

## **Pushing Data**

The application program at the sending site can request a push operation. This means that the sending TCP must not wait for the window to be filled. It must create a segment and send it immediately. The sending TCP must also set the push bit (PSH) to let the receiving TCP know that the segment includes data that must be delivered to the receiving application program as soon as possible and not to wait for more data to come.

# Process-to-Process Delivery

## Urgent Data

- ❖ URG bit is set to send urgent data.
- ❖ The sending application program tells the sending TCP that the piece of data is urgent. The sending TCP creates a segment and inserts the urgent data at the beginning of the segment. The rest of the segment can contain normal data from the buffer. The urgent pointer field in the header defines the end of the urgent data and the start of normal data.
- ❖ When the receiving TCP receives a segment with the URG bit set, it extracts the urgent data from the segment, using the value of the urgent pointer, and delivers them, out of order, to the receiving application program.

# Process-to-Process Delivery

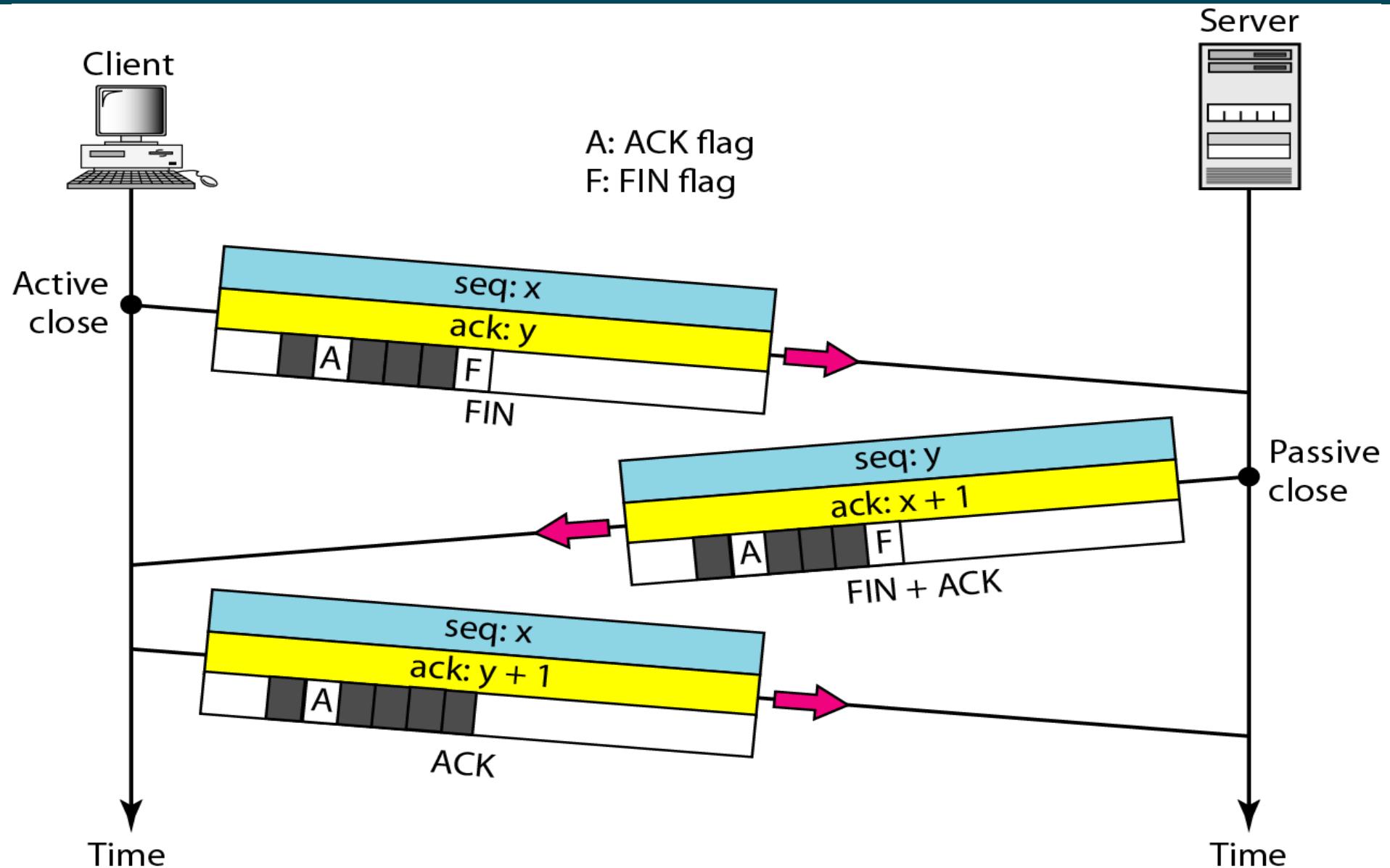
## Connection Termination

Any of the two parties involved in exchanging data (client or server) can close the connection, although it is usually initiated by the client. There are two methods to close the connection as three-way handshaking and four-way handshaking with a half-close option.

## Three-Way Handshaking

Most implementations today allow three-way handshaking for connection termination. It is shown in the following figure:-

# Process-to-Process Delivery



# Process-to-Process Delivery

1. In a normal situation, the client TCP, after receiving a close command from the client process, sends the first segment, a FIN segment in which the FIN flag is set. Note that a FIN segment can include the last chunk of data sent by the client, or it can be just a control segment as shown in Figure 23.20. If it is only a control segment, it consumes only one sequence number.
2. The server TCP, after receiving the FIN segment, informs its process of the situation and sends the second segment, a FIN +ACK segment, to confirm the receipt of the FIN segment from the client and at the same time to announce the closing of the connection in the other direction. This segment can also contain the last chunk of data from the server. If it does not carry data, it consumes only one sequence number.

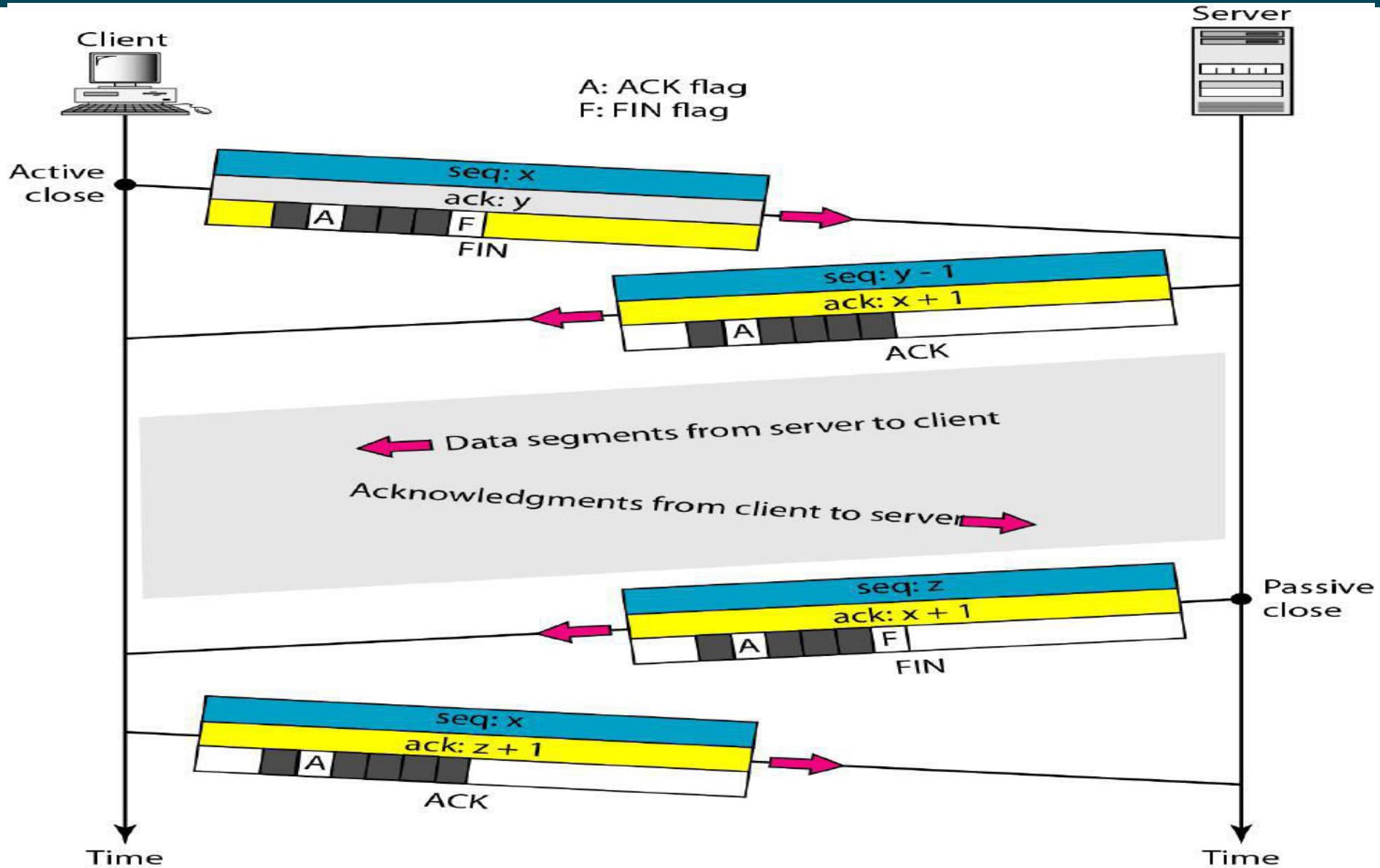
# Process-to-Process Delivery

3. The client TCP sends the last segment, an ACK segment, to confirm the receipt of the FIN segment from the TCP server. This segment contains the acknowledgment number, which is 1 plus the sequence number received in the FIN segment from the server. This segment cannot carry data and consumes no sequence numbers.

## Half-Close

In TCP, one end can stop sending data while still receiving data. This is called a half-close. Although either end can issue a half-close, it is normally initiated by the client.

# Process-to-Process Delivery



# Process-to-Process Delivery

This figure shows an example of a half-close. The client half-closes the connection by sending a FIN segment. The server accepts the half-close by sending the ACK segment. The data transfer from the client to the server stops. The server, however, can still send data. When the server has sent all the processed data, it sends a FIN segment, which is acknowledged by an ACK from the client.

After half-closing of the connection, data can travel from the server to the client and acknowledgments can travel from the client to the server. The client cannot send any more data to the server. Note the sequence numbers we have used. The second segment (ACK) consumes no sequence number. Although the client has received sequence number  $y - 1$  and is expecting  $y$ , the server sequence number is still  $y - 1$ . When the connection finally closes, the sequence number of the last ACK segment is still  $x$ , because no sequence numbers are consumed during data transfer in that direction.

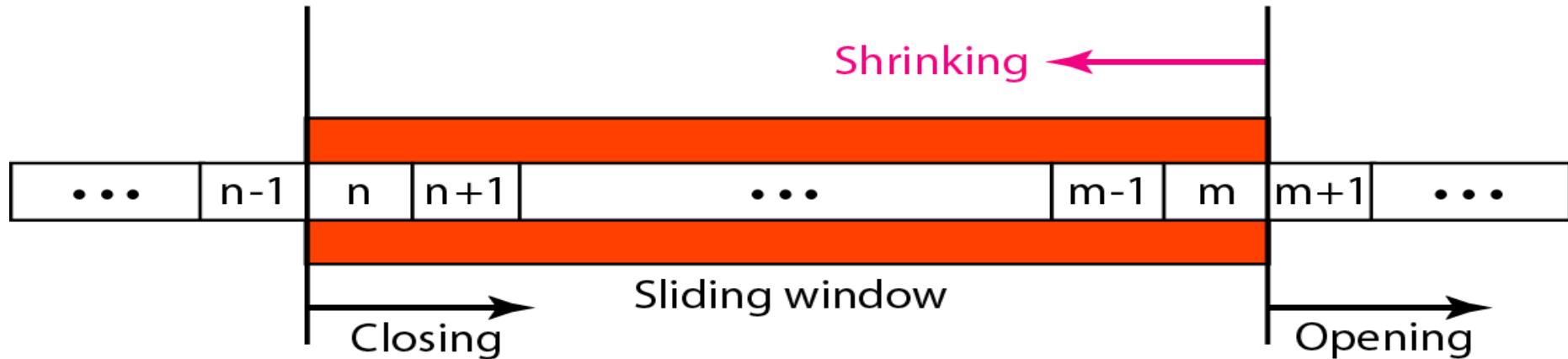
# Process-to-Process Delivery

## Flow Control or TCP Sliding Window

- ❖ TCP uses a sliding window, to handle flow control. The sliding window protocol used by TCP, however, is something between the Go-Back-N and Selective Repeat sliding window.
- ❖ The sliding window protocol in TCP looks like the Go-Back-N protocol because it does not use NAKs; it looks like Selective Repeat because the receiver holds the out-of-order segments until the missing ones arrive.
- ❖ There are two big differences between this sliding window and the one we used at the data link layer.
  - ❖ The sliding window of TCP is byte-oriented; the one we discussed in the data link layer is frame-oriented.
  - ❖ The TCP's sliding window is of variable size; the one we discussed in the data link layer was of fixed size.

# Process-to-Process Delivery

Window size = minimum (rwnd, cwnd)



The window is opened, closed, or shrunk. These three activities, are in the control of the receiver (and depend on congestion in the network), not the sender.

# Process-to-Process Delivery

The sender must obey the commands of the receiver in this matter.

**Opening** a window means moving the right wall to the right. This allows more new bytes in the buffer that are eligible for sending.

**Closing** the window means moving the left wall to the right. This means that some bytes have been acknowledged and the sender need not worry about them anymore.

**Shrinking** the window means moving the right wall to the left.

# Process-to-Process Delivery

The size of the window at one end is determined by the lesser of two values: receiver window (rwnd) or congestion window (cwnd).

The **receiver window** is the value advertised by the opposite end in a segment containing acknowledgment. It is the number of bytes the other end can accept before its buffer overflows and data are discarded.

The **congestion window** is a value determined by the network to avoid congestion

# Process-to-Process Delivery

## Silly Window Syndrome

Silly Window Syndrome is a problem that arises due to the poor implementation of TCP flow control.

It degrades the TCP performance and makes the data transmission extremely inefficient.

It causes the sender window size to shrink to a silly value.

The window size shrinks to such an extent where the data being transmitted is smaller than TCP Header.

# Process-to-Process Delivery

## Causes-

The problem arises due to following causes-

1. Sender transmitting data in small segments repeatedly
2. Receiver accepting only few bytes at a time repeatedly

## Cause-01: Sender Transmitting Data In Small Segments Repeatedly-

- Consider application generates one byte of data to send at a time.
- The poor implementation of TCP causes the sender to send each byte of data in an individual TCP segment.

This problem is solved using Nagle's Algorithm.

## Nagle's Algorithm-

# Process-to-Process Delivery

## Nagle's Algorithm-

Nagle's algorithm suggests-

- Sender should send only the first byte on receiving one byte data from the application.
- Sender should buffer all the rest bytes until the outstanding byte gets acknowledged.
- In other words, sender should wait for 1 RTT.
- After receiving the acknowledgement, sender should send the buffered data in one TCP segment.
- Then, sender should buffer the data again until the previously sent data gets acknowledged.

# Process-to-Process Delivery

## Cause-02: Receiver Accepting Only Few Bytes Repeatedly-

- Consider the receiver continues to be unable to process all the incoming data.
- In such a case, its window size becomes smaller and smaller.
- A stage arrives when it repeatedly sends the window size of 1 byte to the sender.

This problem is solved using Clark's Solution.

## Clark's Solution-

Clark's solution suggests-

- Receiver should not send a window update for 1 byte.
- Receiver should wait until it has a decent amount of space available.
- Receiver should then advertise that window size to the sender.

# Process-to-Process Delivery

## Error Control

TCP is a reliable transport layer protocol. This means that an application program that delivers a stream of data to TCP relies on TCP to deliver the entire stream to the application program on the other end in order, without error, and without any part lost or duplicated.

TCP provides reliability using error control. Error control includes mechanisms for detecting corrupted segments, lost segments, out-of-order segments, and duplicated segments. Error control also includes a mechanism for correcting errors after they are detected. Error detection and correction in TCP is achieved through the use of three simple tools: **checksum, acknowledgment, and time-out**.

# Process-to-Process Delivery

## Checksum

Each segment includes a checksum field which is used to check for a corrupted segment. If the segment is corrupted, it is discarded by the destination TCP and is considered as lost. TCP uses a 16-bit checksum that is mandatory in every segment.

## Acknowledgment

TCP uses acknowledgments to confirm the receipt of data segments. Control segments that carry no data but consume a sequence number are also acknowledged. ACK segments are never acknowledged.

ACK segments do not consume sequence numbers and are not acknowledged.

# Process-to-Process Delivery

## Retransmission

The heart of the error control mechanism is the retransmission of segments. When a segment is corrupted, lost, or delayed, it is retransmitted.

In modern implementations, a segment is retransmitted on two occasions: when a retransmission timer expires or when the sender receives three duplicate ACKs.

## Out-of-Order Segments

- When a segment is delayed, lost, or discarded, the segments following that segment arrive out of order.
- Originally, TCP was designed to discard all out-of-order segments, resulting in the retransmission of the missing segment and the following segments.
- Most implementations today do not discard the out-of-order segments. They store them temporarily and flag them as out-of-order segments until the missing segment arrives. TCP guarantees that data are delivered to the process in order.

# Process-to-Process Delivery

## Exercise

1. The following is a dump of a UDP header in hexadecimal format.

0632000D00 1CE217

- a. What is the source port number?
- b. What is the destination port number?
- c. What is the total length of the user datagram?
- d. What is the length of the data?
- e. Is the packet directed from a client to a server or vice versa?
- f. What is the client process?

# Process-to-Process Delivery

2. The following is a dump of a TCP header in hexadecimal format.

05320017 00000001 00000000 500207FF 00000000

- a. What is the source port number?
- b. What is the destination port number?
- c. What is the sequence number?
- d. What is the acknowledgment number?
- e. What is the length of the header?
- f. What is the type of the segment?
- g. What is the window size?

# Process-to-Process Delivery

3. If WAN link is 2 Mbps and RTT between source and destination is 300 msec, what would be the optimal TCP window size needed to fully utilize the line?
  - a) 60,000 bits
  - b) 75,000 bytes
  - c) 75,000 bits
  - d) 60,000 bytes
4. Suppose host A is sending a large file to host B over a TCP connection. The two end hosts are 10 msec apart (20 msec RTT) connected by a 1 Gbps link. Assume that they are using a packet size of 1000 bytes to transmit the file. For simplicity, ignore ack packets. At least how big would the window size (in packets) have to be for the channel utilization to be greater than 80%?
  - a) 1000
  - b) 1500
  - c) 2000
  - d) 2500

# Process-to-Process Delivery

5. A TCP machine is sending windows of 65535 B over a 1 Gbps channel that has a 10 msec one way delay.
  - a) What is the maximum throughput achievable?
  - b) What is the line efficiency?
6. Consider the three-way handshake mechanism followed during TCP connection establishment hosts P and Q. Let X and Y be two random 32-bit starting sequence numbers chosen by P and Q respectively. Suppose P sends a TCP connection request message to Q with a TCP segment having SYN bit = 1, SEQ number = X, and ACK bit = 0. Suppose Q accepts the connection request. Which one of the following choices represents the information present in the TCP segment header that is sent by Q to P?
  - (A) SYN bit = 0, SEQ number = X + 1, ACK bit = 0, ACK number = Y, FIN bit = 1
  - (B) SYN bit = 1, SEQ number = Y, ACK bit = 1, ACK number = X + 1, FIN bit = 0
  - (C) SYN bit = 1, SEQ number = Y, ACK bit = 1, ACK number = X, FIN bit = 0
  - (D) SYN bit = 1, SEQ number = X + 1, ACK bit = 0, ACK number = Y, FIN bit = 0

# Process-to-Process Delivery

5. Consider a TCP connection in a state where there are no outstanding ACKs. The sender sends two segments back to back. The sequence numbers of the first and second segments are 230 and 290 respectively. The first segment was lost but the second segment was received correctly by the receiver. Let X be the amount of data carried in the first segment (in bytes) and Y be the ACK number sent by the receiver. The values of X and Y are-
- a) 60 and 290
  - b) 230 and 291
  - c) 60 and 231
  - d) 60 and 230

# **Congestion Control and Quality of Service**

# Congestion Control and Quality of Service

- ❖ Congestion control and quality of service are two issues so closely bound together that improving one means improving the other and ignoring one usually means ignoring the other.
- ❖ Most techniques to prevent or eliminate congestion also improve the quality of service in a network.

# Congestion Control and Quality of Service

## CONGESTION

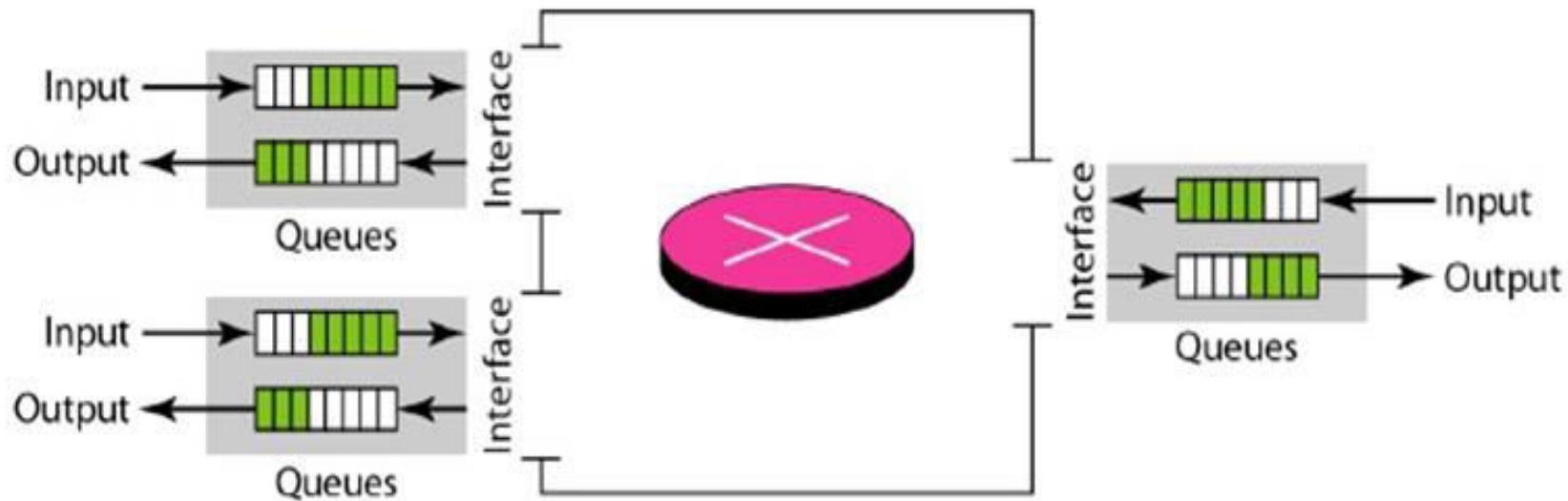
Congestion in a network may occur if the load on the network (the number of packets sent to the network) is greater than the capacity of the network (the number of packets a network can handle).

Congestion control refers to the mechanisms and techniques to control the congestion and keep the load below the capacity.

Congestion happens in any system that involves waiting. For example, congestion happens on a freeway because any abnormality in the flow, such as an accident during rush hour, creates blockage.

# Congestion Control and Quality of Service

Congestion in a network or internetwork occurs because routers and switches have queues-buffers that hold the packets before and after processing. A router, for example, has an input queue and an output queue for each interface. When a packet arrives at the incoming interface, it undergoes three steps before departing, as shown in following figure .



# Congestion Control and Quality of Service

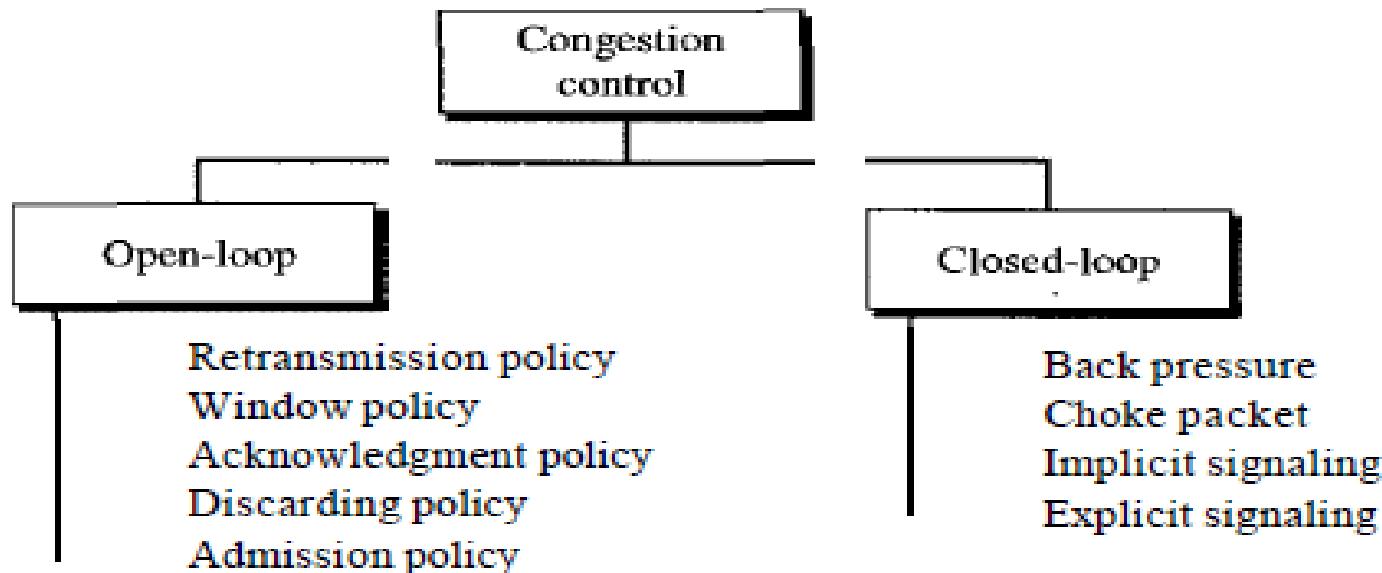
1. The packet is put at the end of the input queue while waiting to be checked.
2. The processing module of the router removes the packet from the input queue once it reaches the front of the queue and uses its routing table and the destination address to find the route.
3. The packet is put in the appropriate output queue and waits its turn to be sent.

We need to be aware of two issues. First, if the rate of packet arrival is higher than the packet processing rate, the input queues become longer and longer. Second, if the packet departure rate is less than the packet processing rate, the output queues become longer and longer.

# Congestion Control and Quality of Service

## CONGESTION CONTROL

- ❖ Congestion control refers to techniques and mechanisms that can either prevent congestion, before it happens, or remove congestion, after it has happened.
- ❖ In general, we can divide congestion control mechanisms into two broad categories: open-loop congestion control (prevention) and closed-loop congestion control (removal) as shown in following figure:



# Congestion Control and Quality of Service

## Open-Loop Congestion Control

In open-loop congestion control, policies are applied to prevent congestion before it happens. In these mechanisms, congestion control is handled by either the source or the destination.

## Retransmission Policy

- Retransmission is sometimes unavoidable.
- If the sender feels that a sent packet is lost or corrupted, the packet needs to be retransmitted.
- Retransmission in general may increase congestion in the network. However, a good retransmission policy can prevent congestion. The retransmission policy and the retransmission timers must be designed to optimize efficiency and at the same time prevent congestion.
- For example, the retransmission policy used by TCP is designed to prevent or alleviate congestion.

# Congestion Control and Quality of Service

## Window Policy

The type of window at the sender may also affect congestion. The Selective Repeat window is better than the Go-Back-N window for congestion control. In the Go-Back-N window, when the timer for a packet times out, several packets may be resent, although some may have arrived safe and sound at the receiver. This duplication may make the congestion worse. The Selective Repeat window, on the other hand, tries to send the specific packets that have been lost or corrupted.

## Acknowledgment Policy

The acknowledgment policy imposed by the receiver may also affect congestion. If the receiver does not acknowledge every packet it receives, it may slow down the sender and help prevent congestion.

Several approaches are used in this case. A receiver may send an acknowledgment only if it has a packet to be sent or a special timer expires. A receiver may decide to acknowledge only  $N$  packets at a time. We need to know that the acknowledgments are also part of the load in a network. Sending fewer acknowledgments means imposing less load on the network.

# Congestion Control and Quality of Service

## Discarding Policy

A good discarding policy by the routers may prevent congestion and at the same time may not harm the integrity of the transmission. For example, in audio transmission, if the policy is to discard less sensitive packets when congestion is likely to happen, the quality of sound is still preserved and congestion is prevented or alleviated.

## Admission Policy

An admission policy, which is a quality-of-service mechanism, can also prevent congestion in virtual-circuit networks. Switches in a flow first check the resource requirement of a flow before admitting it to the network. A router can deny establishing a virtual circuit connection if there is congestion in the network or if there is a possibility of future congestion.

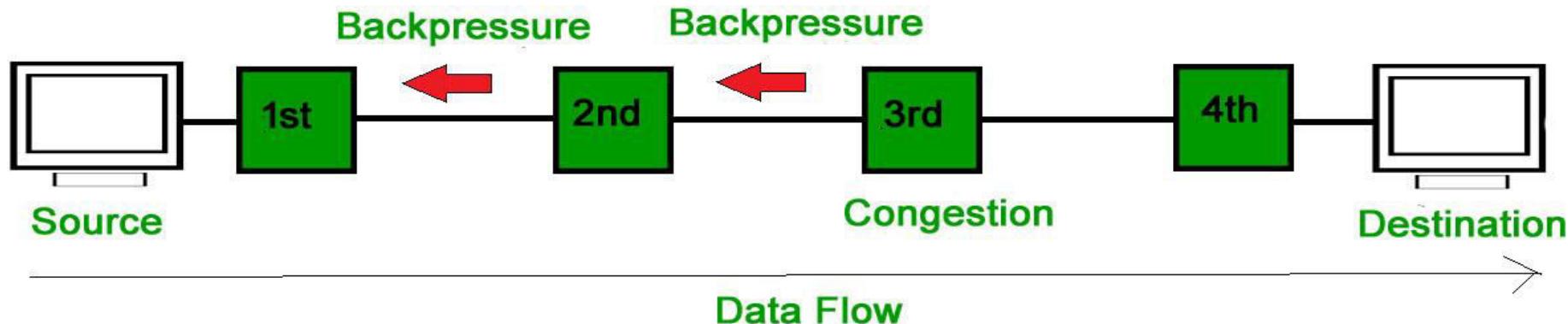
# Congestion Control and Quality of Service

## Closed-Loop Congestion Control

Closed-loop congestion control mechanisms try to alleviate congestion after it happens. Several mechanisms have been used by different protocols.

### Backpressure

Backpressure is a node-to-node congestion control that starts with a node and propagates, in the opposite direction of data flow, to the source. The backpressure technique can be applied only to virtual circuit networks, in which each node knows the upstream node from which a flow of data is coming. Following figure shows the idea of backpressure.



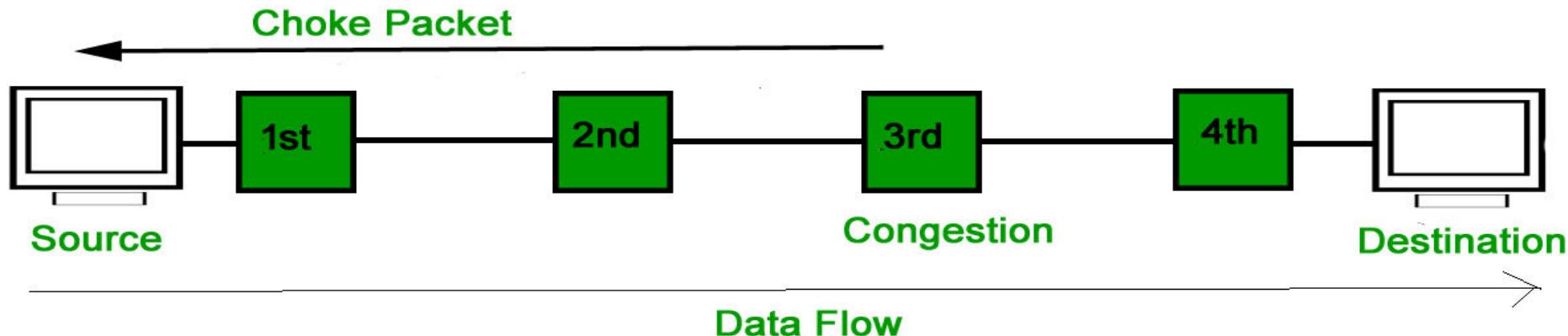
# Congestion Control and Quality of Service

## Choke Packet

A choke packet is a packet sent by a node to the source to inform it of congestion.

In backpressure, the warning is from one node to its upstream node, although the warning may eventually reach the source station.

In the choke packet method, the warning is from the router, which has encountered congestion, to the source station directly. The intermediate nodes through which the packet has traveled are not warned. An example of this type of control in ICMP.



# Congestion Control and Quality of Service

## Implicit Signaling

In implicit signaling, there is no communication between the congested node or nodes and the source. The source guesses that there is a congestion somewhere in the network from other symptoms. For example, when a source sends several packets and there is no acknowledgment for a while, one assumption is that the network is congested. The delay in receiving an acknowledgment is interpreted as congestion in the network; the source should slow down. It is used in the TCP congestion control technique.

## Explicit Signaling

The node that experiences congestion can explicitly send a signal to the source or destination. The explicit signaling method, however, is different from the choke packet method. In the choke packet method, a separate packet is used for this purpose; in the explicit signaling method, the signal is included in the packets that carry data. Explicit signaling, used in Frame Relay congestion control, can occur in either the forward or the backward direction.

# Congestion Control and Quality of Service

## Backward Signaling

A bit can be set in a packet moving in the direction opposite to the congestion. This bit can warn the source that there is congestion and that it needs to slow down to avoid the discarding of packets.

## Forward Signaling

A bit can be set in a packet moving in the direction of the congestion. This bit can warn the destination that there is congestion. The receiver in this case can use policies, such as slowing down the acknowledgments, to alleviate the congestion.

# Congestion Control and Quality of Service

## Congestion Control in TCP

### Congestion Window

The sender's window size is determined not only by the receiver but also by congestion in the network.

The sender has two pieces of information: the receiver advertised window size(rwnd) and the congestion window size(cwnd). The actual size of the window is the minimum of these two i.e.

Actual window size = minimum (rwnd, cwnd)

### Congestion Policy

TCP's general policy for handling congestion is based on three phases: slow start, congestion avoidance, and congestion detection.

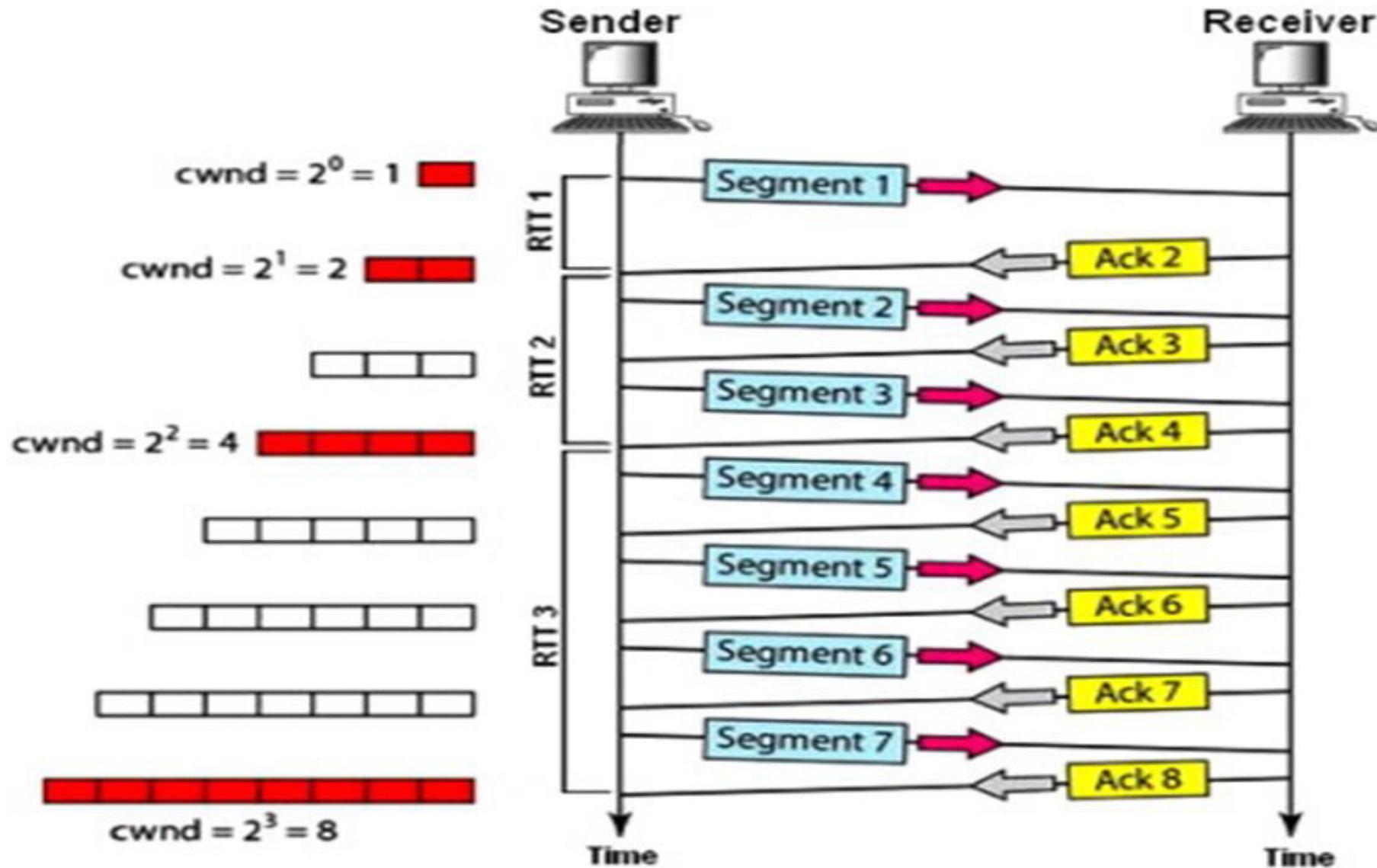
# Congestion Control and Quality of Service

## **Slow Start: Exponential Increase:**

This algorithm is based on the idea that the size of the congestion window (cwnd) starts with one maximum segment size (MSS). The size of the window increases one MSS each time an acknowledgment is received. As the name implies, the window starts slowly, but grows exponentially. It is shown in the following figure:-

In this figure, we have assumed that rwnd is much higher than cwnd, so that the sender window size always equals cwnd. We have assumed that each segment is acknowledged individually.

# Congestion Control and Quality of Service



# Congestion Control and Quality of Service

The sender starts with  $cwnd = 1$  MSS. This means that the sender can send only one segment. After receipt of the acknowledgment for segment 1, the size of the congestion window is increased by 1, which means that  $cwnd$  is now 2. Now two more segments can be sent. When each acknowledgment is received, the size of the window is increased by 1 MSS. When all seven segments are acknowledged,  $cwnd = 8$ .

# Congestion Control and Quality of Service

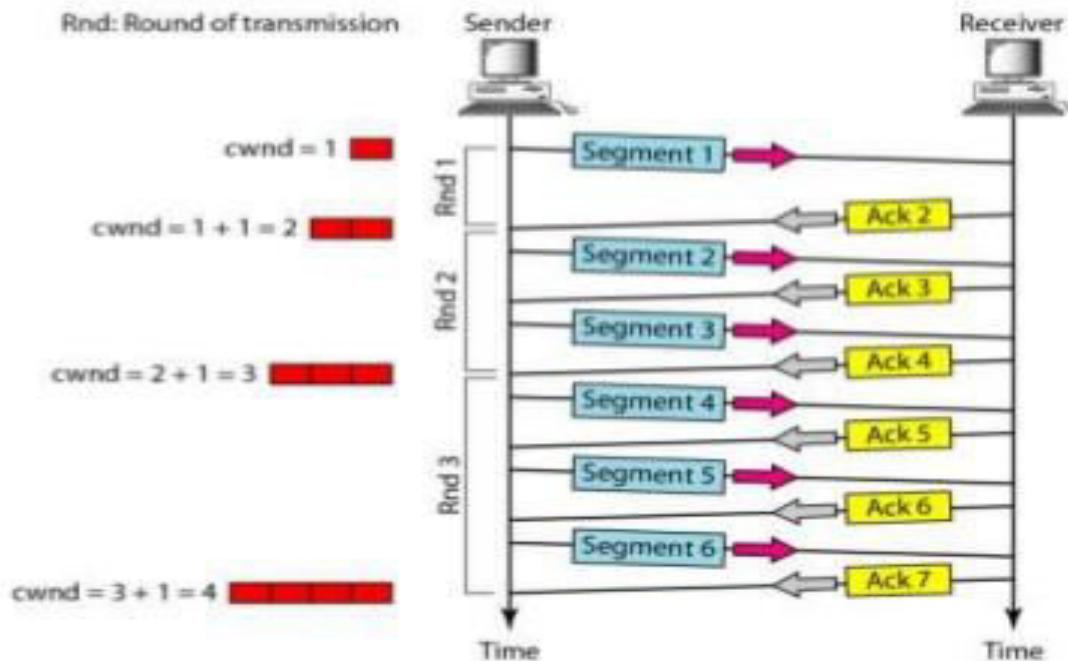
## Congestion Avoidance: Additive Increase

TCP defines another algorithm called congestion avoidance, which undergoes an additive increase instead of an exponential one. When the size of the congestion window reaches the slow-start threshold, the slow-start phase stops and the additive phase begins. In this algorithm, each time the whole window of segments is acknowledged (one round), the size of the congestion window is increased by 1. It is shown in the following figure:-

# Congestion Control and Quality of Service

## CONGESTION CONTROL IN TCP

*Congestion avoidance, additive increase:*



# Congestion Control and Quality of Service

## Congestion Detection: Multiplicative Decrease

If congestion occurs, the congestion window size must be decreased. The only way the sender can guess that congestion has occurred is by the need to retransmit a segment. However, retransmission can occur in one of two cases: when a timer times out or when three ACKs are received. In both cases, the size of the threshold is dropped to one-half, a multiplicative decrease. Most TCP implementations have two reactions:

1. If a time-out occurs, there is a stronger possibility of congestion; a segment has probably been dropped in the network, and there is no news about the sent segments.

In this case TCP reacts strongly:

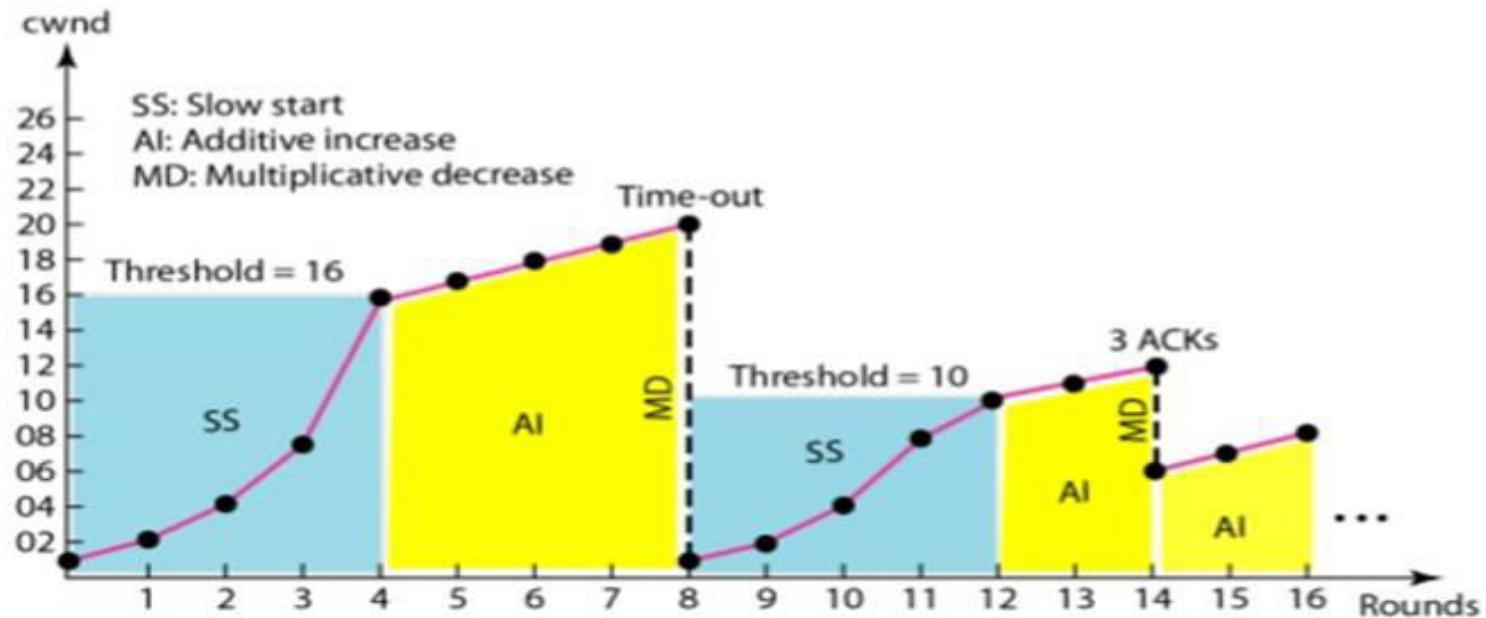
- a. It sets the value of the threshold to one-half of the current window size.
- b. It sets cwnd to the size of one segment.
- c. It starts the slow-start phase again.

# Congestion Control and Quality of Service

2. If three ACKs are received, there is a weaker possibility of congestion; a segment may have been dropped, but some segments after that may have arrived safely since three ACKs are received. This is called fast transmission and fast recovery. In this case, TCP has a weaker reaction:
- a. It sets the value of the threshold to one-half of the current window size.
  - b. It sets cwnd to the value of the threshold (some implementations add three segment sizes to the threshold).
  - c. It starts the congestion avoidance phase.

# Congestion Control and Quality of Service

Example:



In this figure, the time-out occurs when the window size is 20. At this moment, the multiplicative decrease procedure takes over and reduces the threshold to one-half of the previous window size. The previous window size was 20 when the time-out happened so the new threshold is now 10.

# Congestion Control and Quality of Service

TCP moves to slow start again and starts with a window size of 1, and TCP moves to additive increase when the new threshold is reached. When the window size is 12, a three-ACKs event happens. The multiplicative decrease procedure takes over again. The threshold is set to 6 and TCP goes to the additive increase phase this time. It remains in this phase until another time-out or another three ACKs happen.

# Congestion Control and Quality of Service

## QUALITY OF SERVICE

We can informally define quality of service as something a flow seeks to attain.

## Flow Characteristics

There are four types of characteristics attributed to a flow: reliability, delay, jitter, and bandwidth.

## Reliability

Reliability is a characteristic that a flow needs. Lack of reliability means losing a packet or acknowledgment, which entails retransmission. However, the sensitivity of application programs to reliability is not the same. For example, it is more important that electronic mail, file transfer, and Internet access have reliable transmissions than telephony or audio conferencing.

# Congestion Control and Quality of Service

## Delay

Source-to-destination delay is another flow characteristic. Again applications can tolerate delay in different degrees. In this case, telephony, audio conferencing, video conferencing, and remote log-in need minimum delay, while delay in file transfer or e-mail is less important.

## Jitter

Jitter is the variation in delay for packets belonging to the same flow. For example, if four packets depart at times 0, 1, 2, 3 and arrive at 20, 21, 22, 23, all have the same delay, 20 units of time. On the other hand, if the above four packets arrive at 21, 23, 21, and 28, they will have different delays: 21, 22, 19, and 24.

For applications such as audio and video, the first case is completely acceptable; the second case is not.

## Bandwidth

Different applications need different bandwidths. In video conferencing we need to send millions of bits per second to refresh a color screen while the total number of bits in an e-mail may not reach even a million.

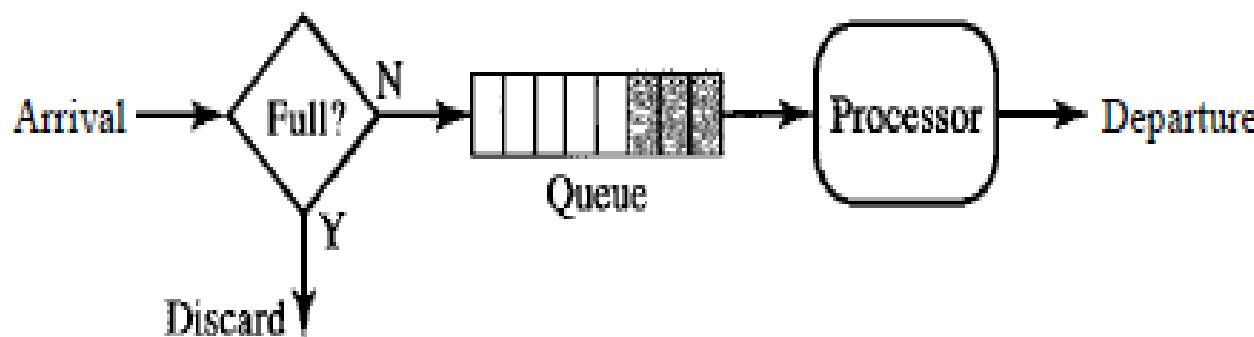
# Congestion Control and Quality of Service

## TECHNIQUES TO IMPROVE QoS

There are four common methods to improve the quality of service: scheduling, traffic shaping, admission control, and resource reservation.

### FIFO Queuing

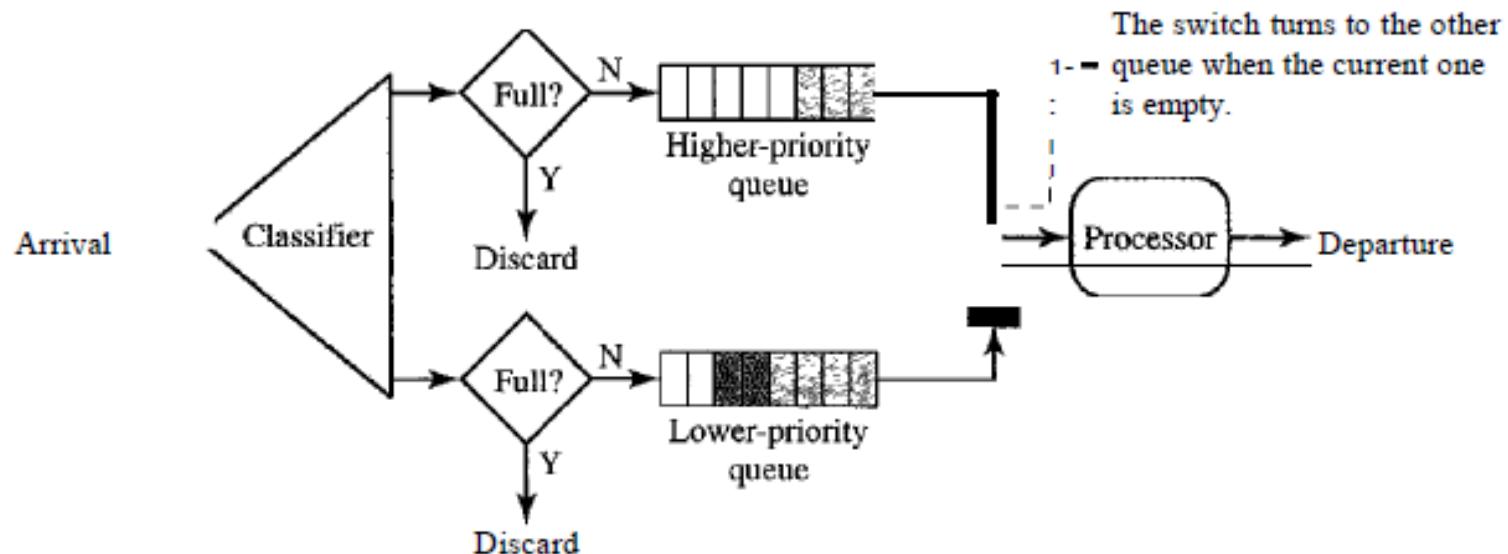
In first-in, first-out (FIFO) queuing, packets wait in a buffer (queue) until the node(router or switch) is ready to process them. If the average arrival rate is higher than the average processing rate, the queue will fill up and new packets will be discarded.



# Congestion Control and Quality of Service

## Priority Queuing

In priority queuing, packets are first assigned to a priority class. Each priority class has its own queue. The packets in the highest-priority queue are processed first. Packets in the lowest-priority queue are processed last. Note that the system does not stop serving a queue until it is empty. Following figure shows priority queuing with two priority levels.



A priority queue can provide better QoS than the FIFO queue because higher priority traffic, such as multimedia, can reach the destination with less delay.

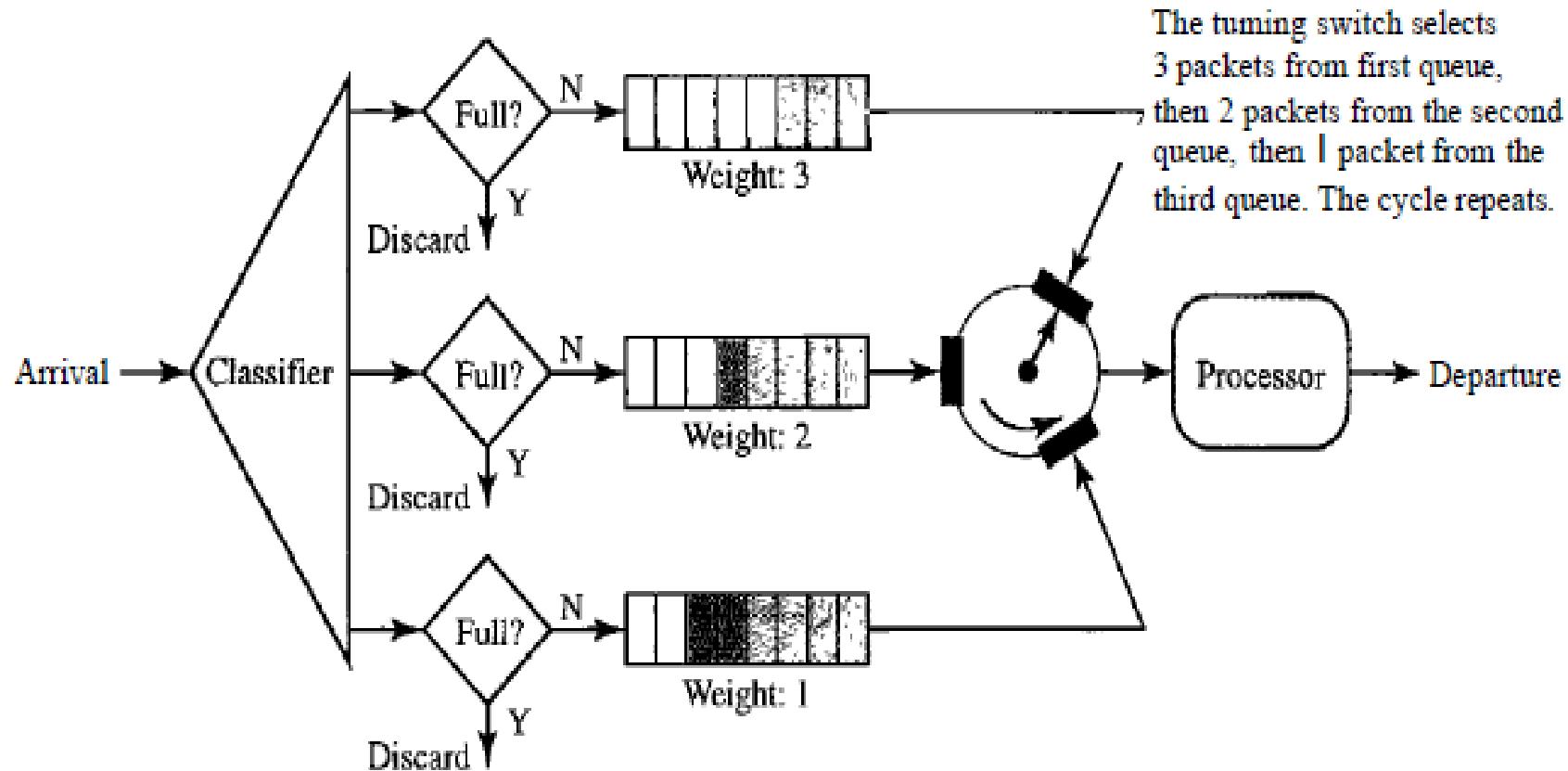
# Congestion Control and Quality of Service

## Weighted Fair Queuing

A better scheduling method is weighted fair queuing. In this technique, the packets are still assigned to different classes and admitted to different queues. The queues, however, are weighted based on the priority of the queues; higher priority means a higher weight.

The system processes packets in each queue in a round-robin fashion with the number of packets selected from each queue based on the corresponding weight. For example, if the weights are 3, 2, and 1, three packets are processed from the first queue, two from the second queue, and one from the third queue.

# Congestion Control and Quality of Service



# Congestion Control and Quality of Service

## Traffic Shaping

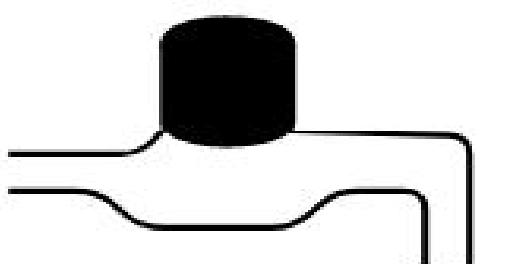
Traffic shaping is a mechanism to control the amount and the rate of the traffic sent to the network. Two techniques can shape traffic: leaky bucket and token bucket.

### Leaky Bucket

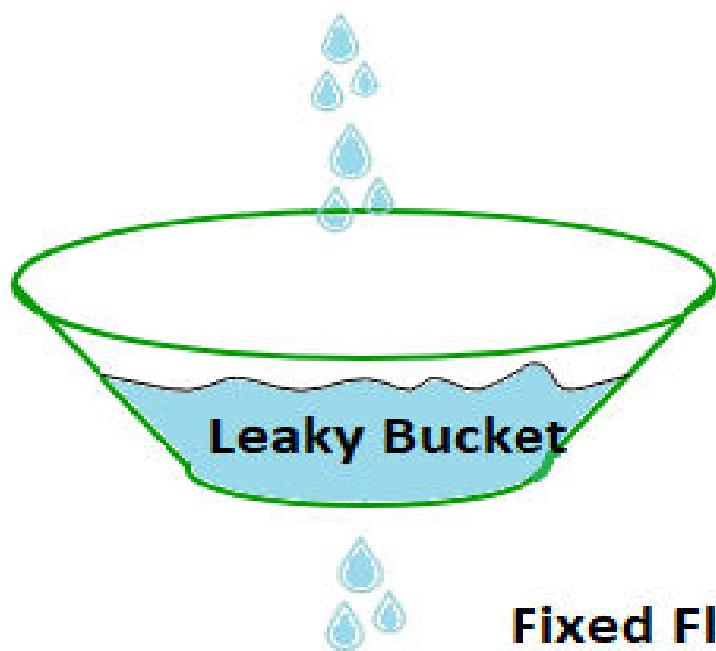
If a bucket has a small hole at the bottom, the water leaks from the bucket at a constant rate as long as there is water in the bucket. The rate at which the water leaks does not depend on the rate at which the water is input to the bucket unless the bucket is empty. The input rate can vary, but the output rate remains constant.

Similarly, in networking, leaky bucket technique can smooth out bursty traffic. Bursty chunks are stored in the bucket and sent out at an average rate. Following figure shows a leaky bucket and its effect.

# Congestion Control and Quality of Service

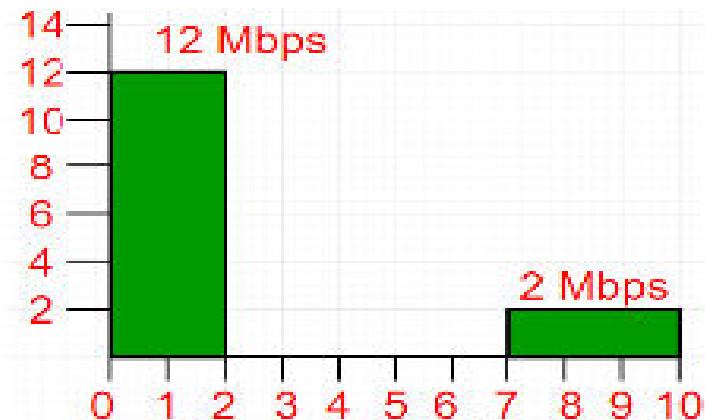


Bursty Flow

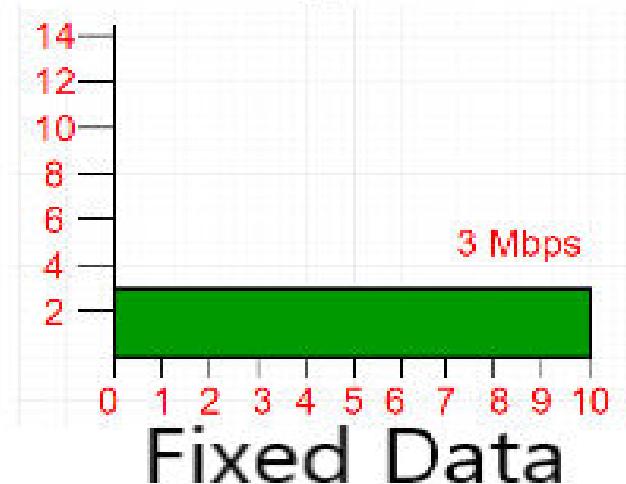


Leaky Bucket

Fixed Flow



Bursty Data



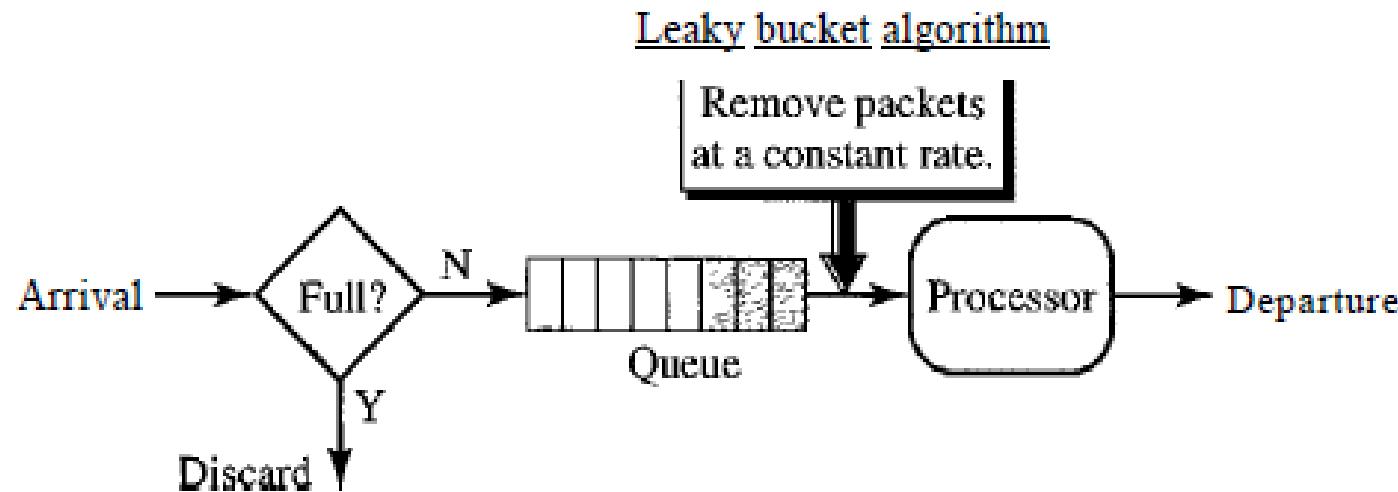
Fixed Data

# Congestion Control and Quality of Service

In this figure, the host sends a burst of data at a rate of 12 Mbps for 2s, for a total of 24 Mbits of data. The host is silent for 5s and then sends data at a rate of 2 Mbps for 3s, for a total of 6 Mbits of data. In all, the host has sent 30 Mbits of data in 10s. The leaky bucket smooths the traffic by sending out data at a rate of 3 Mbps during the same 10s.

**Note:** Leaky bucket may prevent congestion.

## Leaky bucket implementation



# Congestion Control and Quality of Service

In above figure, FIFO queue holds the packets. If the traffic consists of fixed-size packets, then the process removes a fixed number of packets from the queue at each tick of the clock. If the traffic consists of variable-length packets, the fixed output rate must be based on the number of bytes or bits.

The following is an algorithm for variable-length packets:

1. Initialize a counter to  $n$  at the tick of the clock.
2. If  $n$  is greater than the size of the packet, send the packet and decrement the counter by the packet size. Repeat this step until  $n$  is smaller than the packet size.
3. Reset the counter and go to step 1.

# Congestion Control and Quality of Service

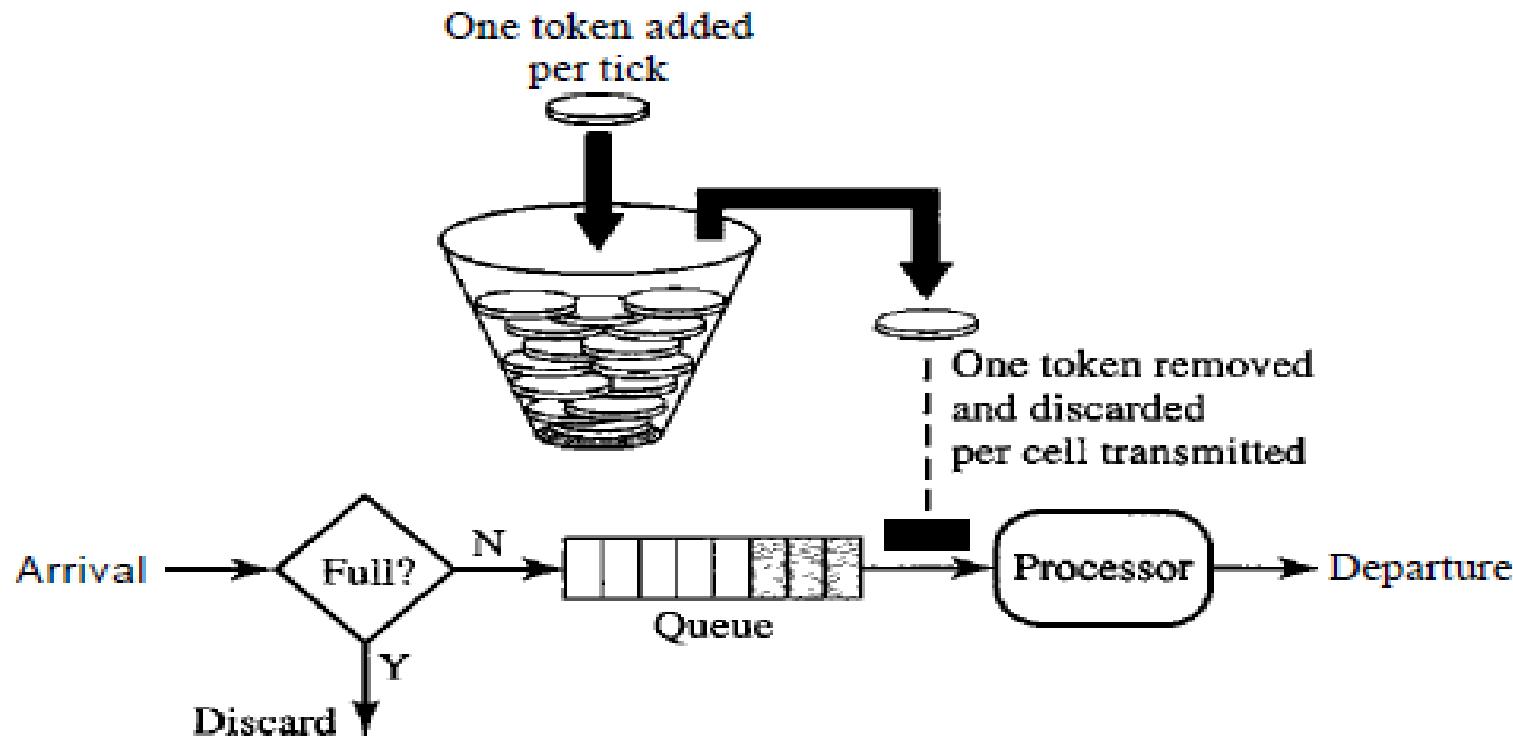
## Token Bucket

The leaky bucket is very restrictive. It does not credit an idle host. For example, if a host is not sending for a while, its bucket becomes empty. Now if the host has bursty data, the leaky bucket allows only an average rate. The time when the host was idle is not taken into account.

On the other hand, the token bucket algorithm allows idle hosts to accumulate credit for the future in the form of tokens. For each tick of the clock, the system sends  $n$  tokens to the bucket. The system removes one token for every cell (or byte) of data sent. For example, if  $n$  is 100 and the host is idle for 100 ticks, the bucket collects 10,000 tokens. Now the host can consume all these tokens in one tick with 10,000 cells, or the host takes 1000 ticks with 10 cells per tick. In other words, the host can send bursty data as long as the bucket is not empty.

# Congestion Control and Quality of Service

It is shown in the following figure:-



The token bucket can easily be implemented with a counter. The token is initialized to zero. Each time a token is added, the counter is incremented by 1. Each time a unit of data is sent, the counter is decremented by 1. When the counter is zero, the host cannot send data.

# Congestion Control and Quality of Service

# AKTU Examination Questions

1. What are the services of Transport Layer?
2. Discuss TCP window management in detail. Also explain silly window syndrome and their solution.
3. What is Congestion? Differentiate between congestion control and flow control with example. Also discuss congestion prevention policies.
4. Provide few reasons for congestion in a network.
5. How does transport layer perform duplication control?
6. What is congestion? Briefly describe the techniques that prevent congestion.
7. Enumerate on TCP header and working of TCP and differentiate TCP and UDP with frame format.

# Process-to-Process Delivery

8. Enumerate how the transport layer ensures that the complete message arrives at the destination and in the proper order.
9. Explain the three way handshaking protocol to establish the transport level connection.
10. Explain about the TCP header and working of TCP protocol and differentiate between TCP and UDP with frame format.
11. The following is the dump of a TCP header in hexa decimal format:  
05320017 00000001 00000000 500207FF 00000000
  - (i) What is the sequence number?
  - (ii) What is the destination port number?
  - (iii) What is the acknowledgment number?
  - (iv) What is the window size?
12. What do you understand by Quality of service, parameters? List various Quality of service parameters.

# Process-to-Process Delivery

13. How does transport layer perform duplication control?
14. Enumerate on TCP header and working of TCP and differentiate TCP and UDP with frame format.
15. Explain the three way handshaking protocol to establish the transport level connection.
16. Enumerate how the transport layer ensure that the complete message arrives at the destination and in the proper order.

# **Computer Network**

**Lecture taken by**

**Dharmendra Kumar  
(Associate Professor)**

**United College of Engineering and Research,  
Prayagraj**

# **UNIT-5**

# **Domain Name System(DNS)**

# Domain Name System(DNS)

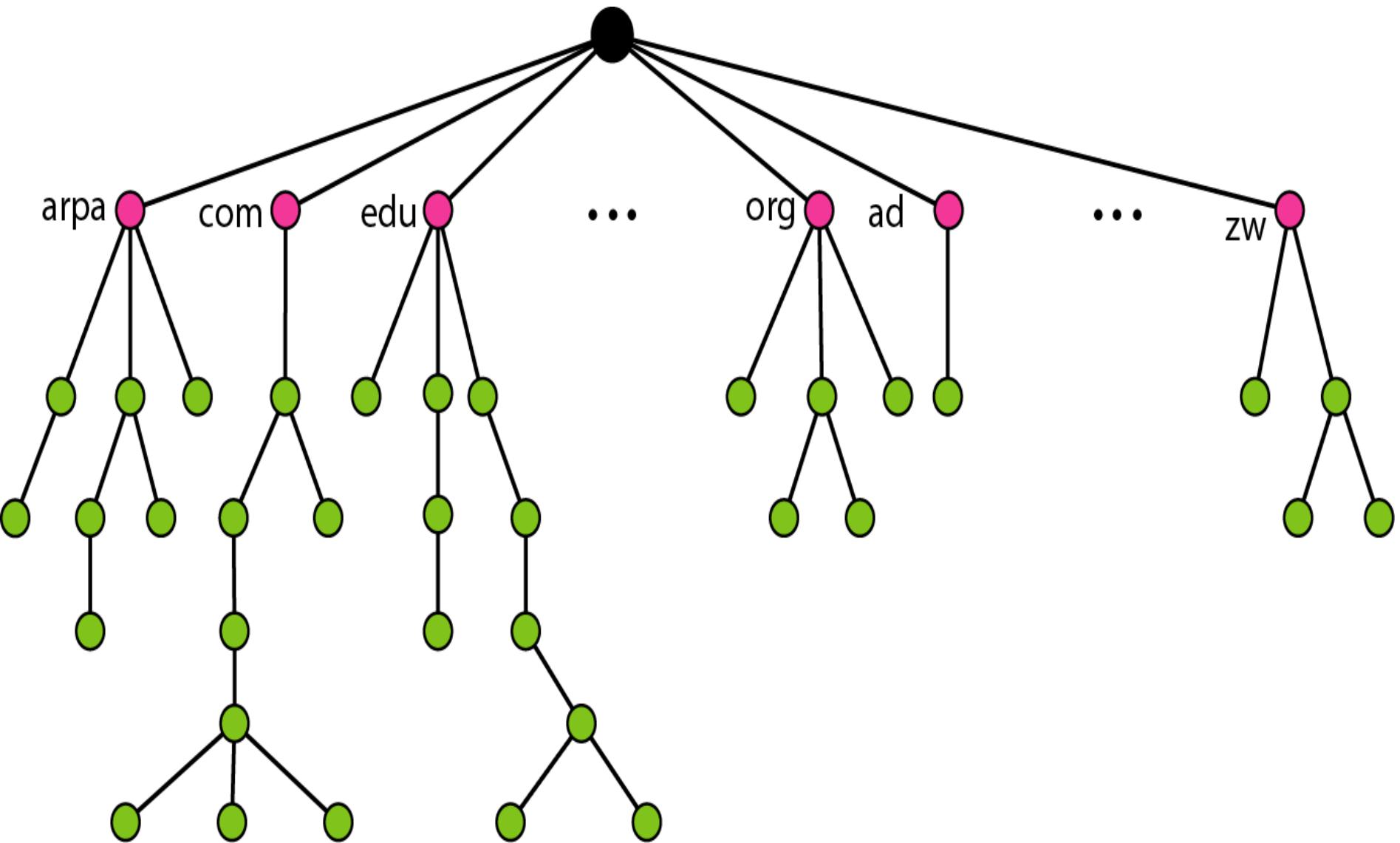
- Domain Name System is a client/server based application layer protocol.
- It translates a domain name (eg. nec.edu.np) into an IP address (eg. 202.37.94.177).
- The DNS has a distributed database that resides on multiple machines on the Internet.

# Domain Name System(DNS)

## DOMAIN NAME SPACE

- Domain name space is designed in the form of hierarchical name space.
- In this design, the names are defined in an inverted-tree structure with the root at the top. The tree can have only 128 levels: level 0 (root) to level 127.
- It is shown in the following figure:-

# Domain Name System(DNS)



# Domain Name System(DNS)

## Label

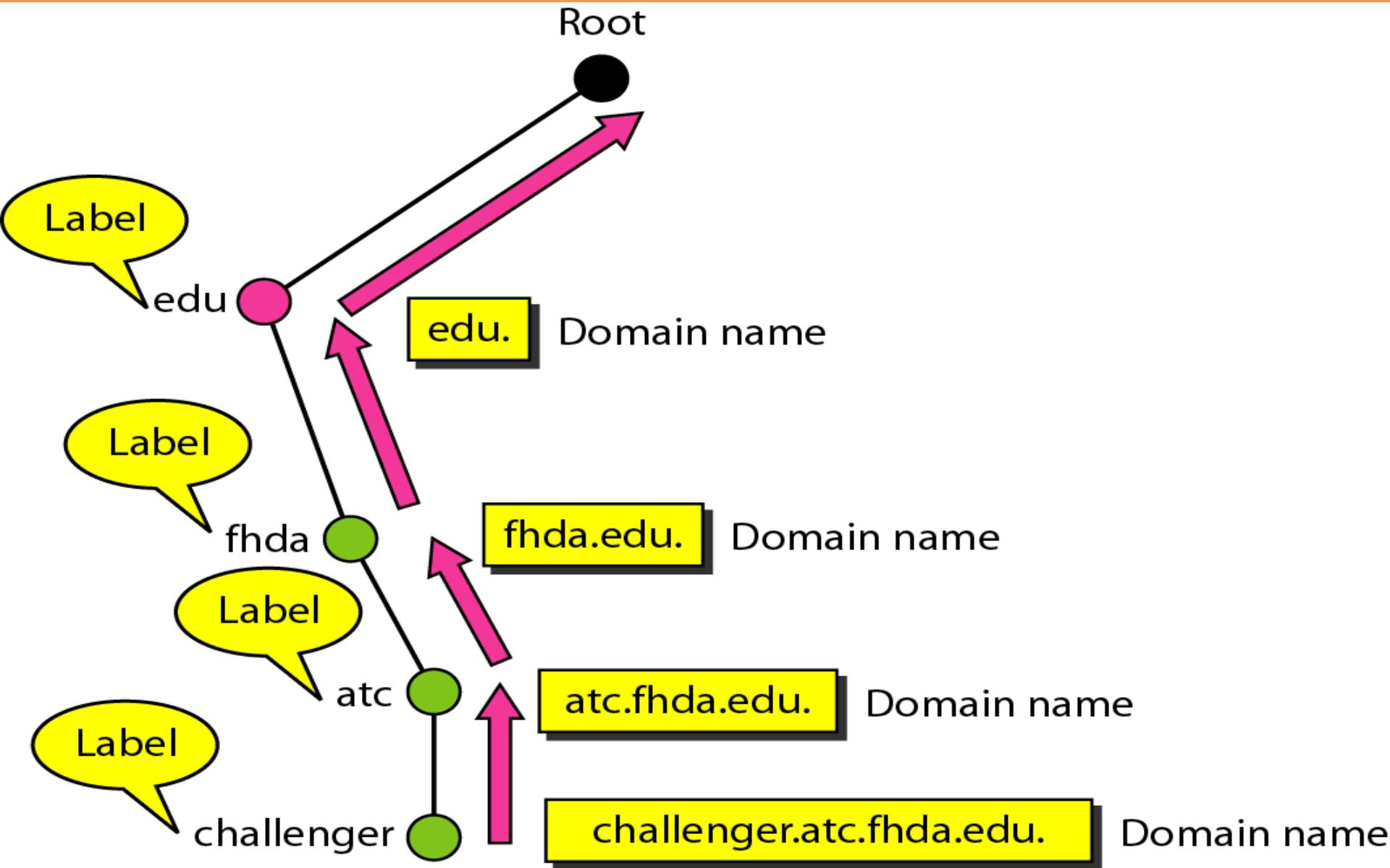
- Each node in the tree has a label, which is a string with a maximum of 63 characters.
- The root label is a null string (empty string).
- DNS requires that children of a node (nodes that branch from the same node) have different labels, which guarantees the uniqueness of the domain names.

# Domain Name System(DNS)

## Domain Name

- Each node in the tree has a domain name.
- A full domain name is a sequence of labels separated by dots (.). The domain names are always read from the node up to the root.
- The last label is the label of the root (null). This means that a full domain name always ends in a null label, which means the last character is a dot because the null string is nothing.
- Following figure shows some domain names.

# Domain Name System(DNS)



# Domain Name System(DNS)

## Fully Qualified Domain Name

If a label is terminated by a null string, it is called a fully qualified domain name (FQDN).

An FQDN is a domain name that contains the full name of a host.

It contains all labels, from the most specific to the most general, that uniquely define the name of the host.

**Example:** **challenger.ate.tbda.edu.**

# Domain Name System(DNS)

## Partially Qualified Domain Name

- If a label is not terminated by a null string, it is called a partially qualified domain name (PQDN).
- A PQDN starts from a node, but it does not reach the root. It is used when the name to be resolved belongs to the same site as the client. Here the resolver can supply the missing part, called the suffix, to create an FQDN. For example, if a user at the jhda.edu. site wants to get the IP address of the challenger computer, he or she can define the partial name  
**challenger**
- The DNS client adds the suffix **atc.jhda.edu.** before passing the address to the DNS server.

# Domain Name System(DNS)

FQDN

challenger.atc.fhda.edu.  
cs.hmme.com.  
www.funny.int.

PQDN

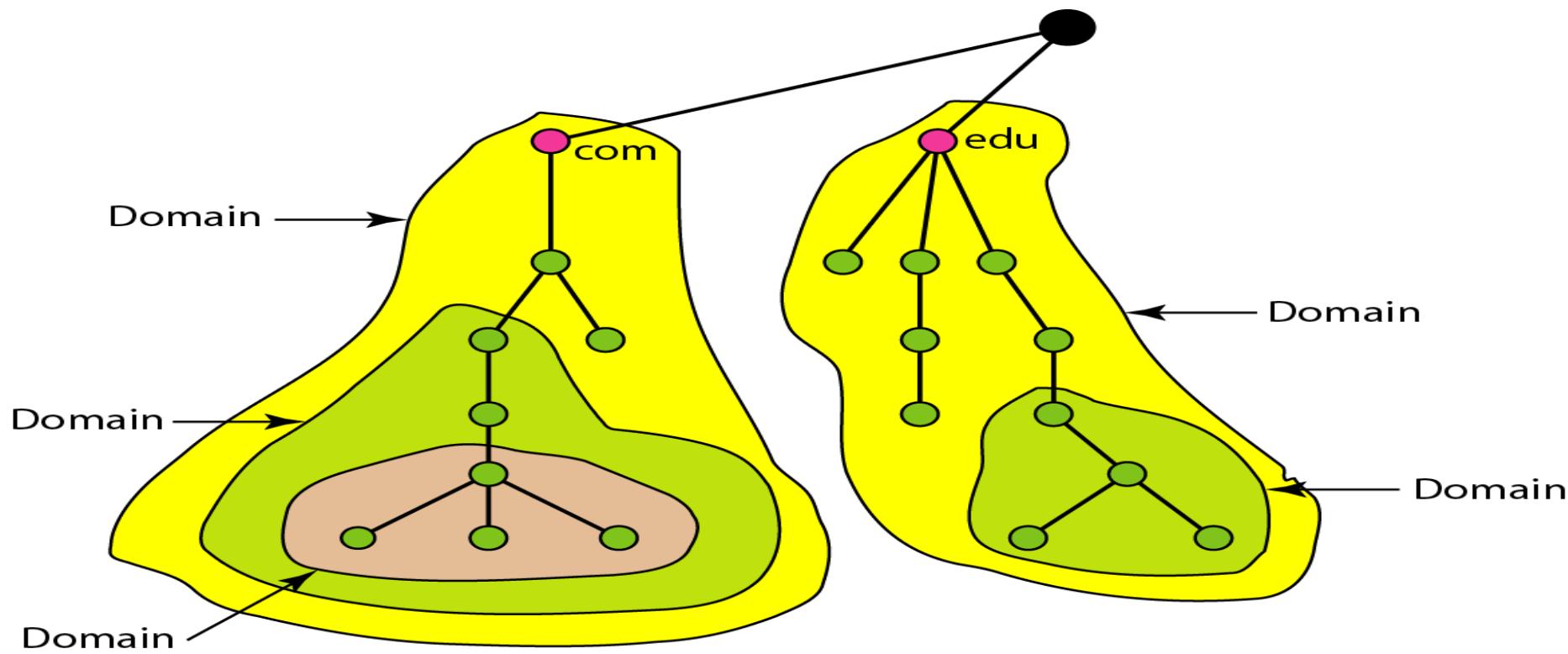
challenger.atc.fhda.edu  
cs.hmme  
www

# Domain Name System(DNS)

## Domain

A domain is a sub-tree of the domain name space. The name of the domain is the domain name of the node at the top of the sub-tree. Following figure shows some domains.

**Note:** A domain may itself be divided into domains.



# Domain Name System(DNS)

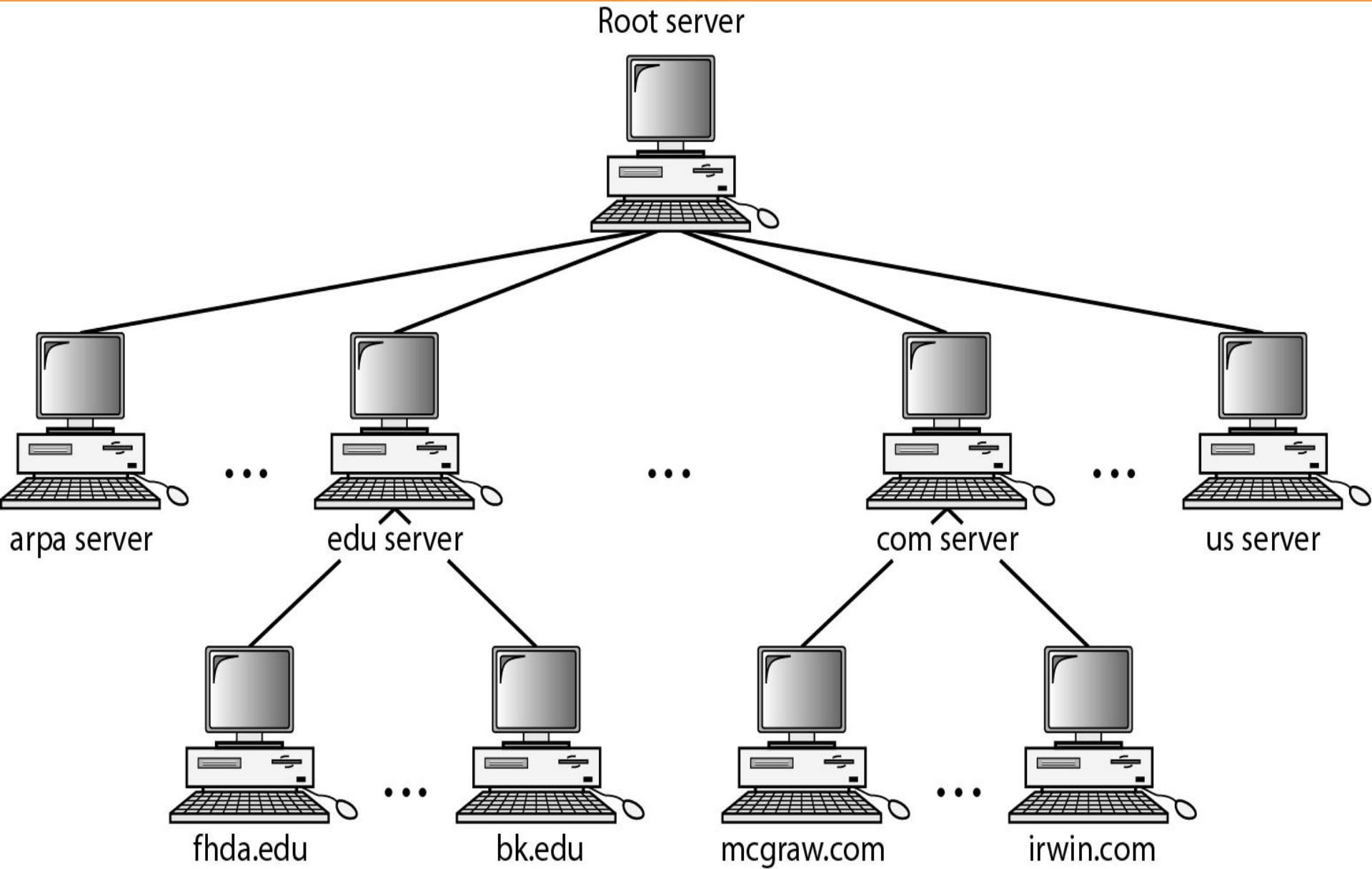
## DISTRIBUTION OF NAME SPACE

- It is very inefficient and also unreliable to have just one computer store all domain name space.
- It is inefficient because responding to requests from all over the world places a heavy load on the system.
- It is unreliable because any failure makes the data inaccessible.

## Hierarchy of Name Servers

- The solution to above problems is to distribute the information among many computers called DNS servers.
- One way to do this is to divide the whole space into many domains based on the first level.
- Because a domain created in this way could be very large, DNS allows domains to be divided further into smaller domains (subdomains).
- Each server can be responsible (authoritative) for either a large or a small domain. In other words, we have a hierarchy of servers in the same way that we have a hierarchy of names.

# Domain Name System(DNS)



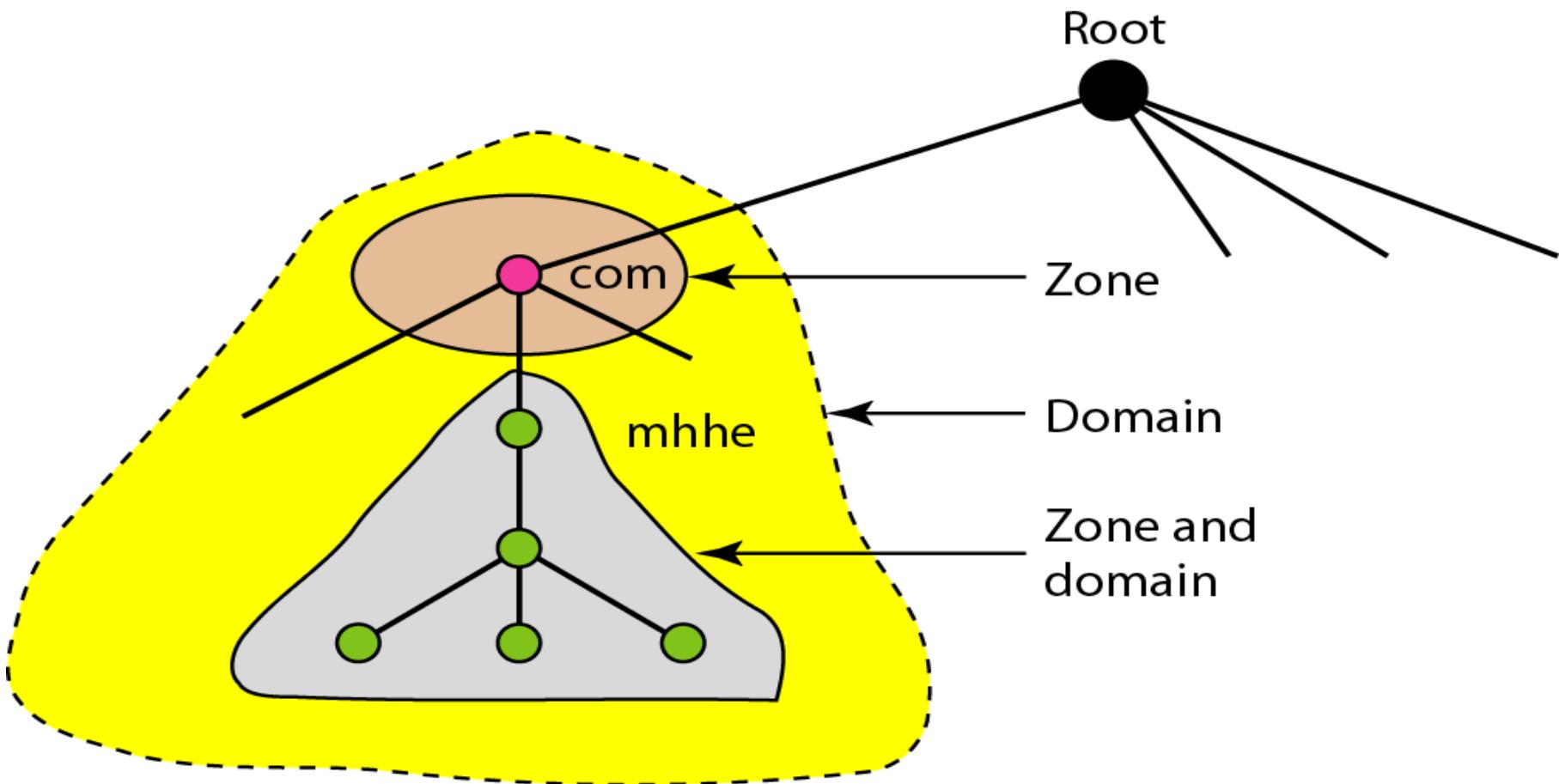
# Domain Name System(DNS)

## Zone

- Since the complete domain name hierarchy can not be stored on a single server, so it is divided among many servers. What a server is responsible for or has authority over is called a zone.
- We can define a zone as a contiguous part of the entire tree.
- If a server accepts responsibility for a domain and does not divide the domain into smaller domains, the domain and the zone refer to the same thing.
- The server makes a database called a zone file and keeps all the information for every node under that domain.
- If a server divides its domain into subdomains and delegates part of its authority to other servers, then domain and zone refer to different things.

# Domain Name System(DNS)

- The information about the nodes in the subdomains is stored in the servers at the lower levels, with the original server keeping some sort of reference to these lower-level servers.



# Domain Name System(DNS)

## Root Server

- A root server is a server whose zone consists of the whole tree.
- A root server usually does not store any information about domains but delegates its authority to other servers, keeping references to those servers.
- There are several root servers, each covering the whole domain name space.
- The servers are distributed all around the world.

# Domain Name System(DNS)

## Primary and Secondary Servers

- DNS defines two types of servers: **primary and secondary**. A primary server is a server that stores a file about the zone for which it is an authority. It is responsible for creating, maintaining, and updating the zone file. It stores the zone file on a local disk.
- A secondary server is a server that transfers the complete information about a zone from another server (primary or secondary) and stores the file on its local disk. The secondary server neither creates nor updates the zone files. If updating is required, it must be done by the primary server, which sends the updated version to the secondary.
- The primary and secondary servers are both authoritative for the zones they serve.

# Domain Name System(DNS)

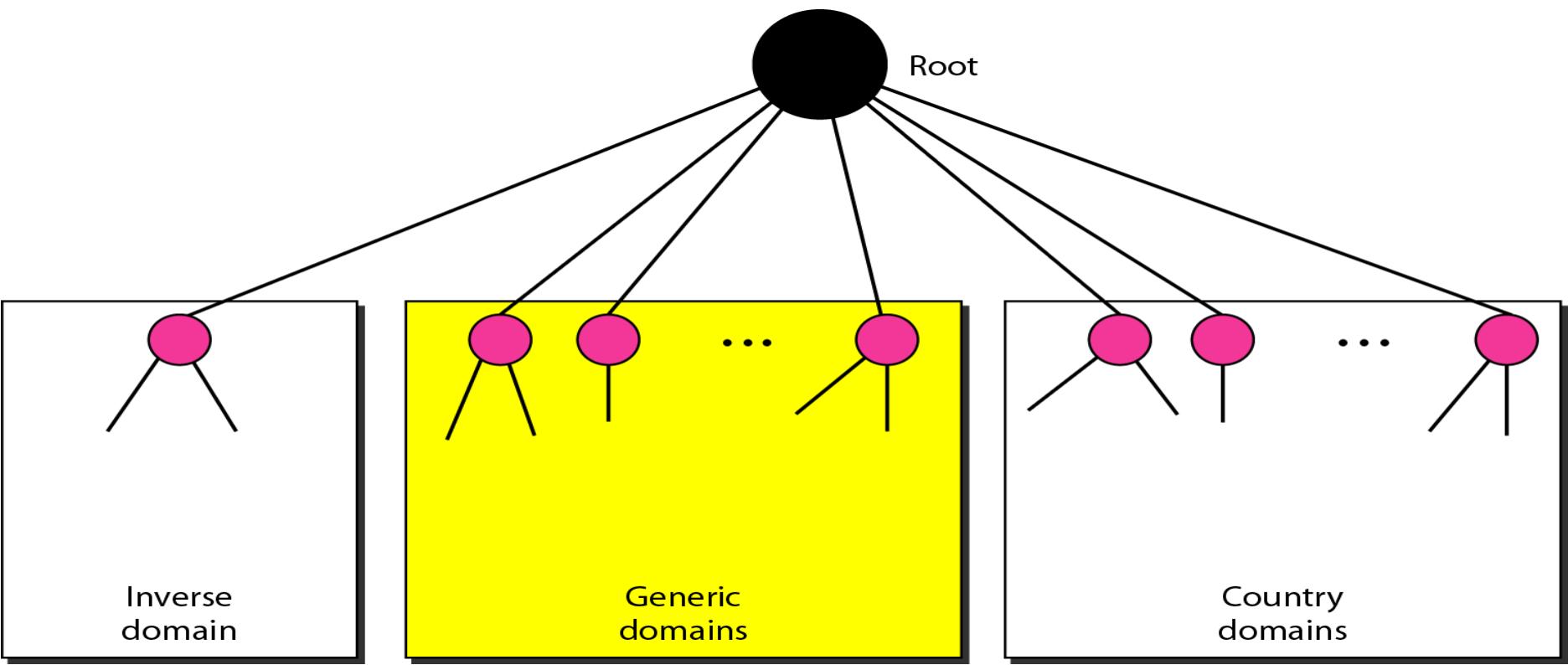
**Note:** A server can be a primary server for a specific zone and a secondary server for another zone.

- A primary server loads all information from the disk file; the secondary server loads all information from the primary server.
- When the secondary downloads information from the primary, it is called **zone transfer**.

# Domain Name System(DNS)

## DNS IN THE INTERNET

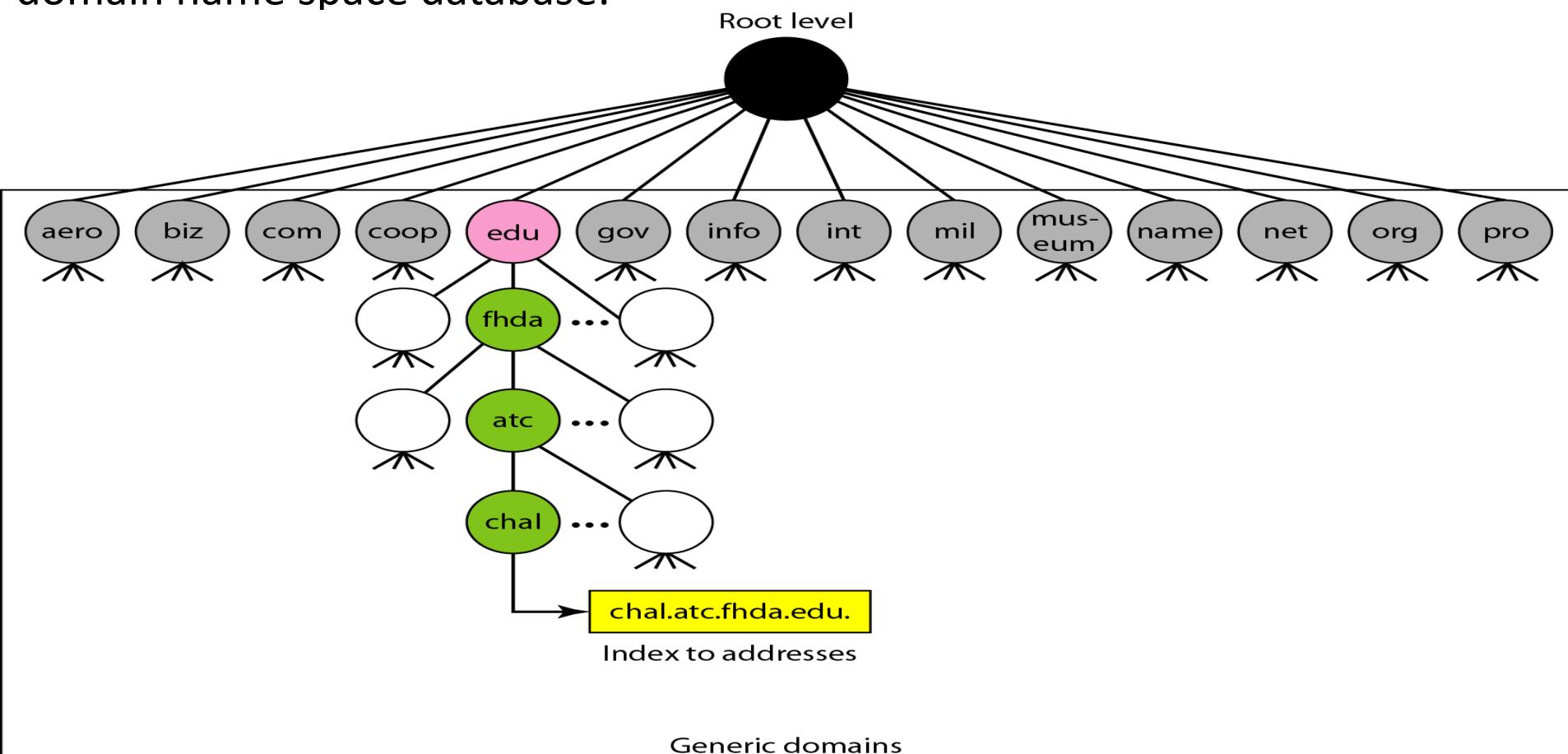
- DNS is a protocol that can be used in different platforms.
- In the Internet, the domain name space (tree) is divided into three different sections: generic domains, country domains, and the inverse domain.



# Domain Name System(DNS)

## Generic Domains

The generic domains define registered hosts according to their generic behavior. Each node in the tree defines a domain, which is an index to the domain name space database.



# Domain Name System(DNS)

- In the tree, the first level in the generic domains section allows 14 possible labels. These labels describe the organization types.

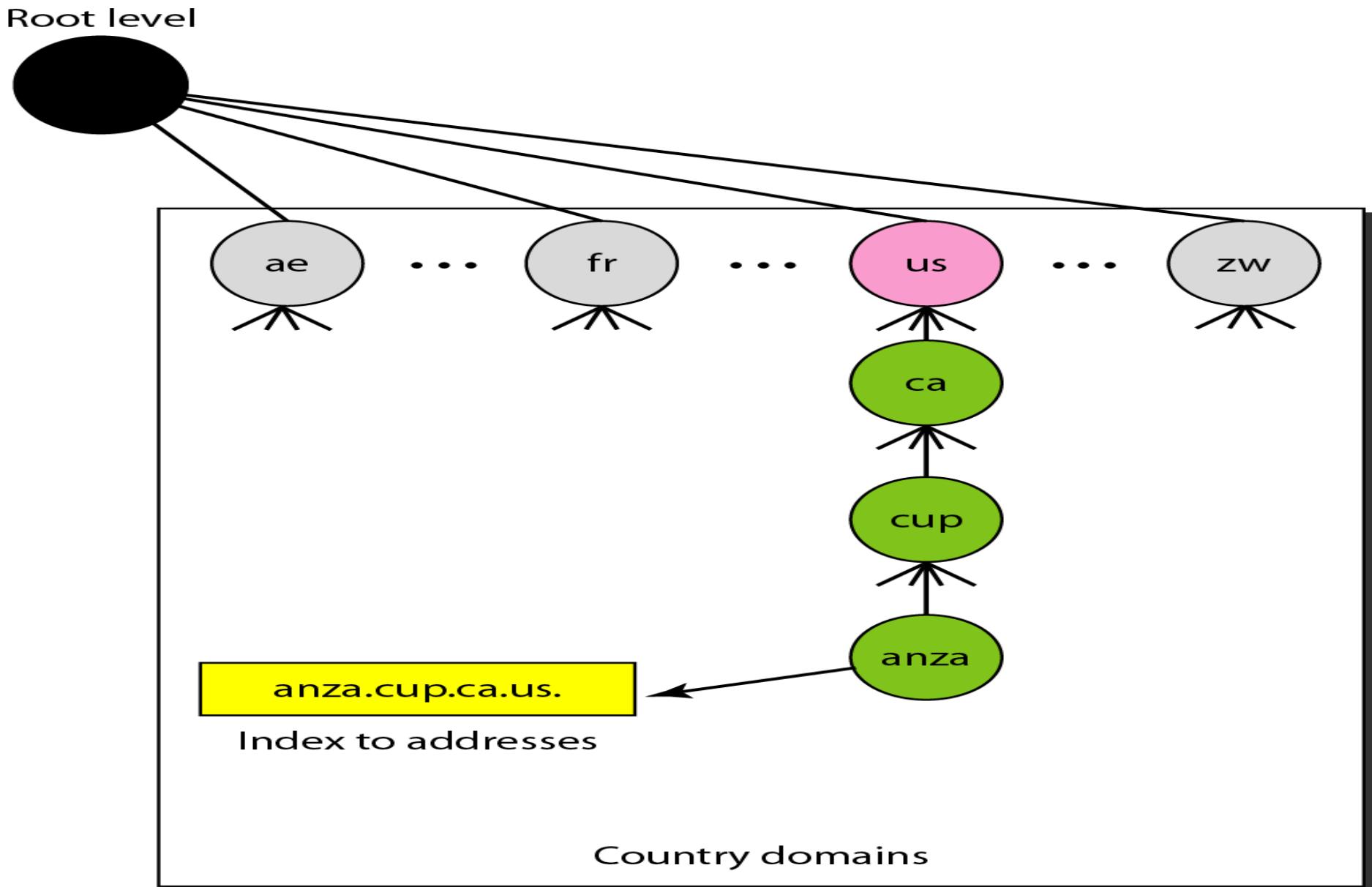
<i>Label</i>	<i>Description</i>
<b>aero</b>	Airlines and aerospace companies
<b>biz</b>	Businesses or firms (similar to “com”)
<b>com</b>	Commercial organizations
<b>coop</b>	Cooperative business organizations
<b>edu</b>	Educational institutions
<b>gov</b>	Government institutions
<b>info</b>	Information service providers
<b>int</b>	International organizations
<b>mil</b>	Military groups
<b>museum</b>	Museums and other nonprofit organizations
<b>name</b>	Personal names (individuals)
<b>net</b>	Network support centers
<b>org</b>	Nonprofit organizations
<b>pro</b>	Professional individual organizations

# Domain Name System(DNS)

## Country Domains

- The country domains section uses two-character country abbreviations (e.g., us for United States). Second labels can be organizational, or they can be more specific, national designations.
- Following figure shows the country domains section.
- The address **anza.cup.ca.us** can be translated to De Anza College in Cupertino, California, in the United States.

# Domain Name System(DNS)

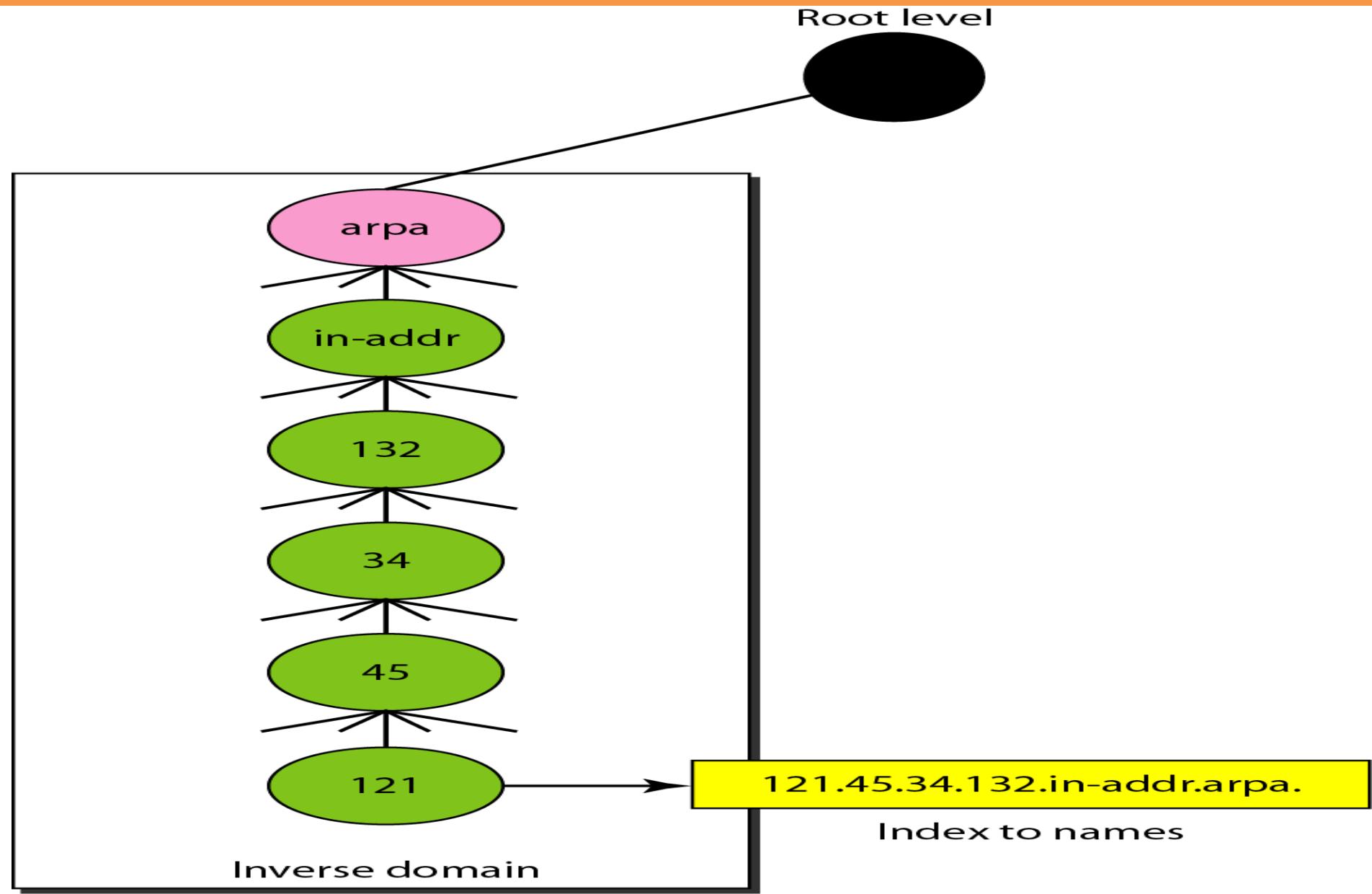


# Domain Name System(DNS)

## Inverse Domain

- The inverse domain is used to map an address to a name.
- This may happen, for example, when a server has received a request from a client to do a task.
- The server asks its resolver to send a query to the DNS server to map an address to a name to determine if the client is on the authorized list. This type of query is called an inverse or pointer (PTR) query.
- To handle a pointer query, the inverse domain is added to the domain name space with the first-level node called **arpa**. The second level is also one single node named **in-addr** (for inverse address). The rest of the domain defines IP addresses.

# Domain Name System(DNS)



# Domain Name System(DNS)

## RESOLUTION

Mapping a name to an address or an address to a name is called name-address resolution.

## Resolver

- DNS is designed as a client/server application.
- A host that needs to map an address to a name or a name to an address calls a DNS client called a resolver.
- The resolver accesses the closest DNS server with a mapping request. If the server has the information, it satisfies the resolver; otherwise, it either refers the resolver to other servers or asks other servers to provide the information.
- After the resolver receives the mapping, it interprets the response to see if it is a real resolution or an error, and finally delivers the result to the process that requested it.

# Domain Name System(DNS)

## Mapping Names to Addresses

- The resolver gives a domain name to the server and asks for the corresponding address. In this case, the server checks the generic domains or the country domains to find the mapping.
- If the domain name is from the generic domains section, the resolver receives a domain name such as "chal.atc.jhda.edu.". The query is sent by the resolver to the local DNS server for resolution. If the local server cannot resolve the query, it either refers the resolver to other servers or asks other servers directly.
- If the domain name is from the country domains section, the resolver receives a domain name such as "ch.jhda.cu.ca.us.". The procedure is the same.

# Domain Name System(DNS)

## Mapping Addresses to Names

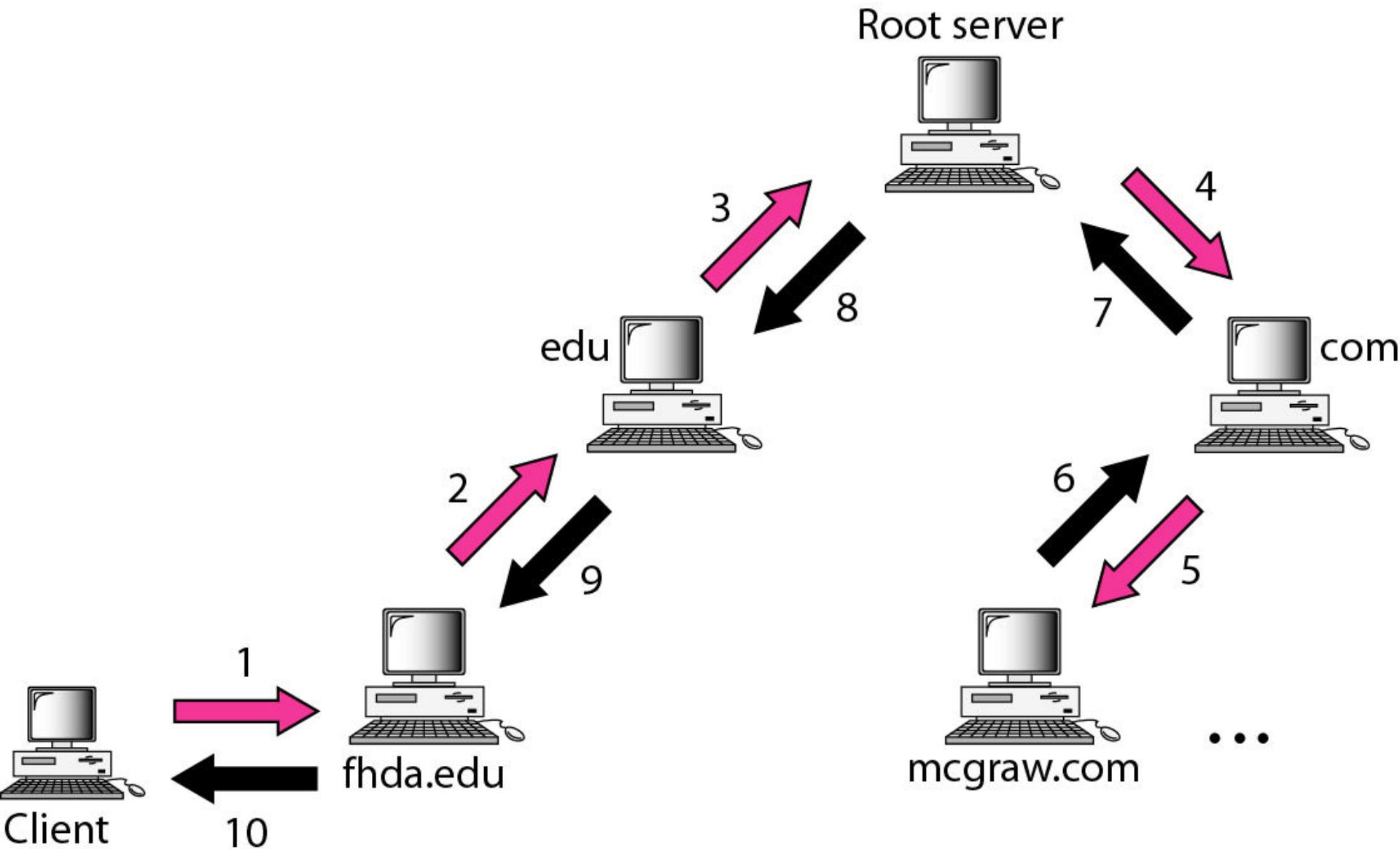
- A client send an IP address to a server to be mapped to a domain name. This is called a PTR query.
- To answer queries of this kind, DNS uses the inverse domain. However, in the request, the IP address is reversed and the two labels in-addr and arpa are appended to create a domain acceptable by the inverse domain section.
- For example, if the resolver receives the IP address 132.34.45.121, the resolver first inverts the address and then adds the two labels before sending. The domain name sent is "121.45.34.132.in-addr.arpa." which is received by the local DNS and resolved.

# Domain Name System(DNS)

## Recursive Resolution

- The client (resolver) can ask for a recursive answer from a name server. This means that the resolver expects the server to supply the final answer.
- If the server is the authority for the domain name, it checks its database and responds.
- If the server is not the authority, it sends the request to another server (the parent usually) and waits for the response. If the parent is the authority, it responds; otherwise, it sends the query to yet another server. When the query is finally resolved, the response travels back until it finally reaches the requesting client. This is called recursive resolution and is shown in the following figure:-

# Domain Name System(DNS)

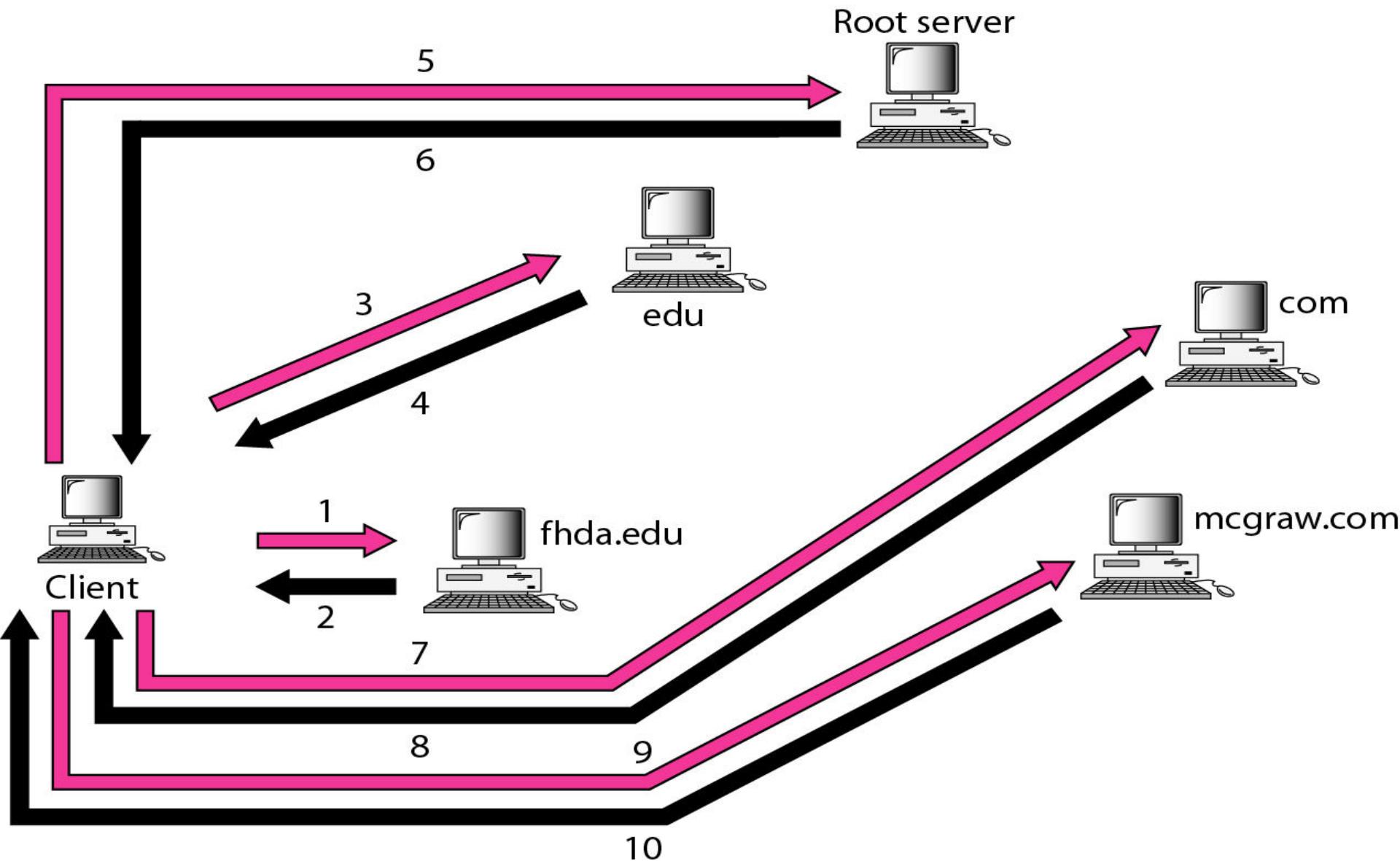


# Domain Name System(DNS)

## Iterative Resolution

- In this approach, if the server is an authority for the name, it sends the answer. If it is not, it returns (to the client) the IP address of the server that it thinks can resolve the query. The client is responsible for repeating the query to this second server.
- If the newly addressed server can resolve the problem, it answers the query with the IP address; otherwise, it returns the IP address of a new server to the client.
- Now the client must repeat the query to the third server. This process is called iterative resolution because the client repeats the same query to multiple servers.
- In following figure, the client queries four servers before it gets an answer from the **mcgraw.com** server.

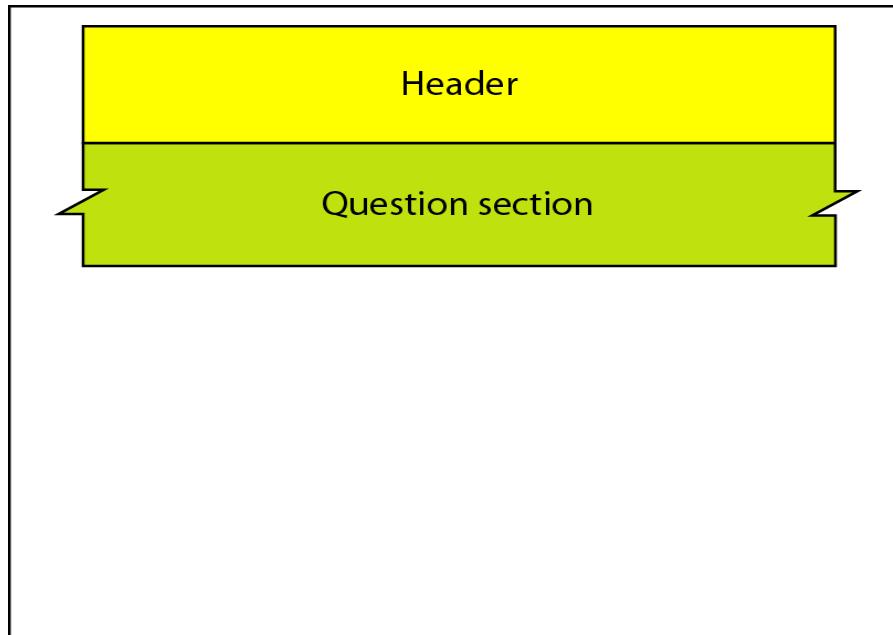
# Domain Name System(DNS)



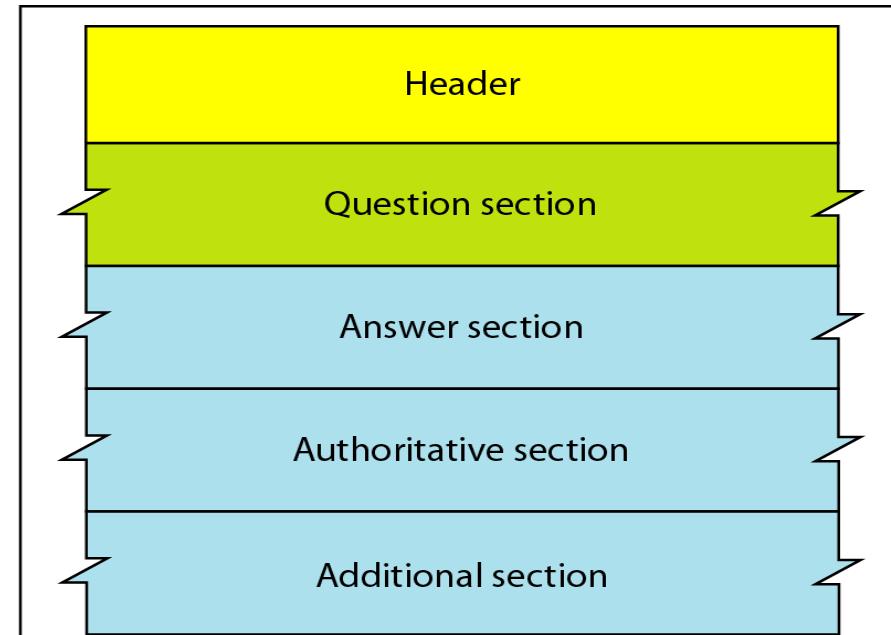
# Domain Name System(DNS)

## DNS MESSAGES

- DNS has two types of messages: **query and response**.
- Both types have the same format. The query message consists of a header and question records; the response message consists of a header, question records, answer records, authoritative records, and additional records.



a. Query



b. Response

# Domain Name System(DNS)

## Header

Both query and response messages have the same header format with some fields set to zero for the query messages. The header is 12 bytes, and its format is shown in the following figure:-

Identification	Flags
Number of question records	Number of answer records (all 0s in query message)
Number of authoritative records (all 0s in query message)	Number of additional records (all 0s in query message)

# Domain Name System(DNS)

- The **identification subfield** is used by the client to match the response with the query. The client uses a different identification number each time it sends a query. The server duplicates this number in the corresponding response.
- The **flags subfield** is a collection of subfields that define the type of the message, the type of answer requested, the type of desired resolution (recursive or iterative), and so on.
- The **number of question records subfield** contains the number of queries in the question section of the message.

# Domain Name System(DNS)

- The **number of answer records subfield** contains the number of answer records in the answer section of the response message. Its value is zero in the query message.
- The **number of authoritative records subfield** contains the number of authoritative records in the authoritative section of a response message. Its value is zero in the query message.
- Finally, the **number of additional records subfield** contains the number additional records in the additional section of a response message. Its value is zero in the query message.

# Domain Name System(DNS)

## Question Section

This is a section consisting of one or more question records. It is present on both query and response messages.

## Answer Section

This is a section consisting of one or more resource records. It is present only on response messages. This section includes the answer from the server to the client (resolver).

## Authoritative Section

This is a section consisting of one or more resource records. It is present only on response messages. This section gives information (domain name) about one or more authoritative servers for the query.

# Domain Name System(DNS)

## Additional Information Section

This is a section consisting of one or more resource records. It is present only on response messages. This section provides additional information that may help the resolver.

For example, a server may give the domain name of an authoritative server to the resolver in the authoritative section, and include the IP address of the same authoritative server in the additional information section.

# Domain Name System(DNS)

## TYPES OF RECORDS

Two types of records are used in DNS.

### **Question Record**

- The question records are used in the question section of the query and response messages.
- A question record is used by the client to get information from a server. This contains the domain name.

### **Resource Record**

- The resource records are used in the answer, authoritative, and additional information sections of the response message.
- Each domain name (each node on the tree) is associated with a record called the resource record. The server database consists of resource records. Resource records are also what is returned by the server to the client.

# **Remote Logging, Electronic Mail, and File Transfer**

# REMOTE LOGGING

- Remote Login is a process in which user can login into remote site i.e. computer use services that are available on the remote computer.
- After logging on, a user can use the services available on the remote computer and transfer the results back to the local computer.

## TELNET

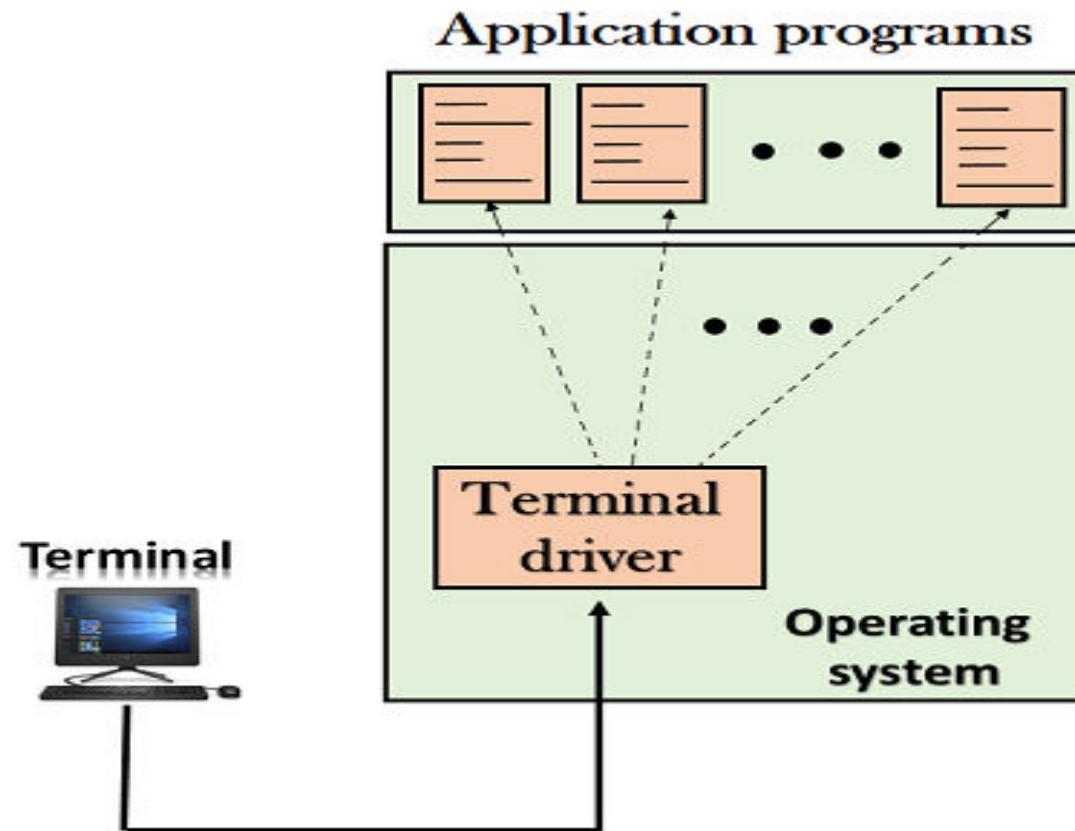
- It is a client/server application program.
- TELNET is an abbreviation for TErminal NETwork.
- It is the standard TCP/IP protocol for virtual terminal service as proposed by the International Organization for Standards (ISO).
- TELNET enables the establishment of a connection to a remote system in such a way that the local terminal appears to be a terminal at the remote system.

# TELNET

## Types of login

There are two types of login: local login and remote login.

## Local login

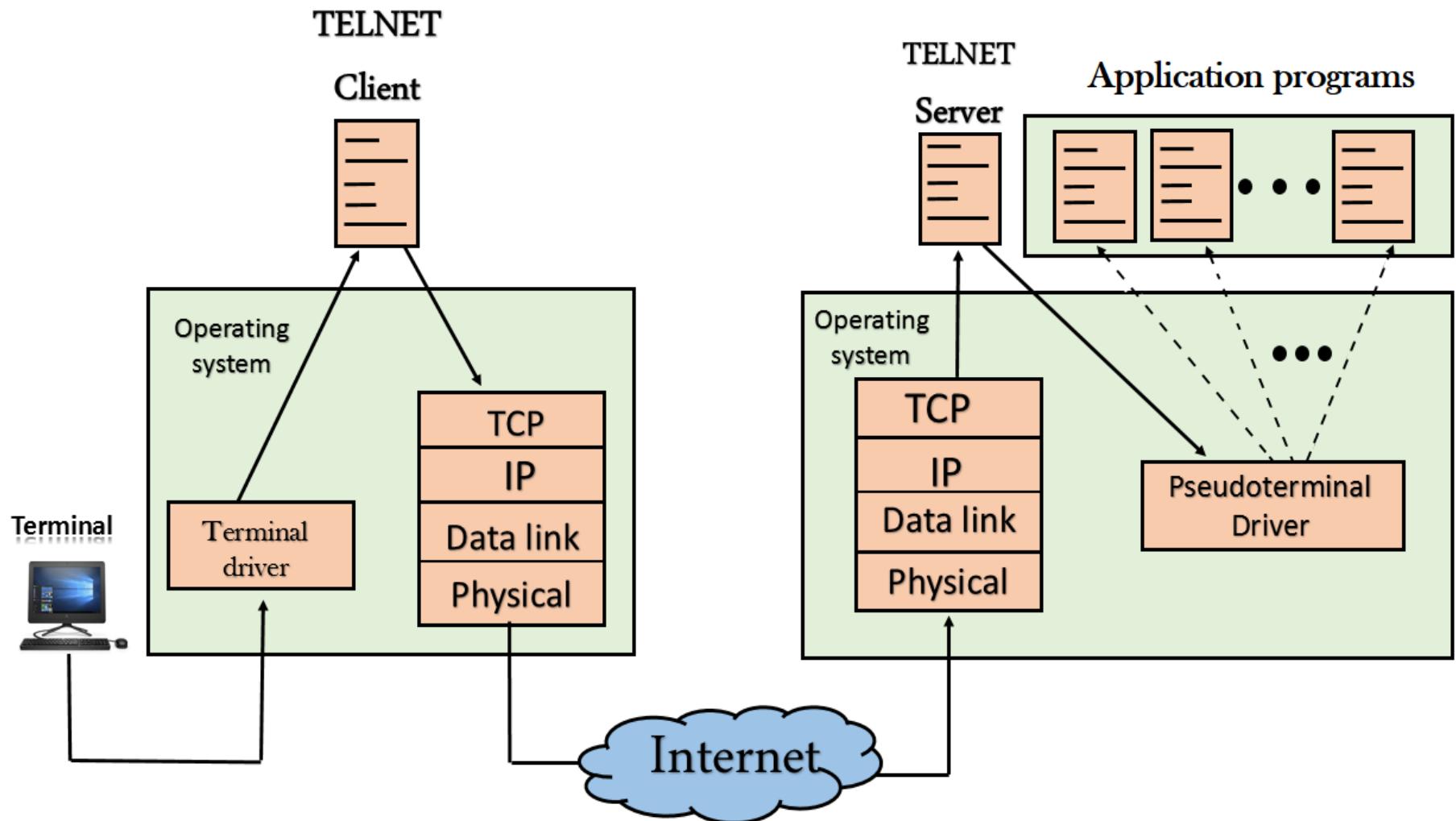


# Local Login(continue)

When a user logs into a local timesharing system, it is called local login. As a user types at a terminal or at a workstation running a terminal emulator, the keystrokes are accepted by the terminal driver. The terminal driver passes the characters to the operating system. The operating system, in turn, interprets the combination of characters and invokes the desired application program or utility.

# Remote login

## Remote login



# Remote login(continue)

- When a user wants to access an application program or utility located on a remote machine, he performs remote login.
- Here, the TELNET client and server programs come into use.
- The user sends the keystrokes to the terminal driver, where the local operating system accepts the characters but does not interpret them. The characters are sent to the TELNET client, which transforms the characters to a universal character set called **network virtual terminal** (NVT) characters and delivers them to the local TCP/IP protocol stack.

# Remote login(continue)

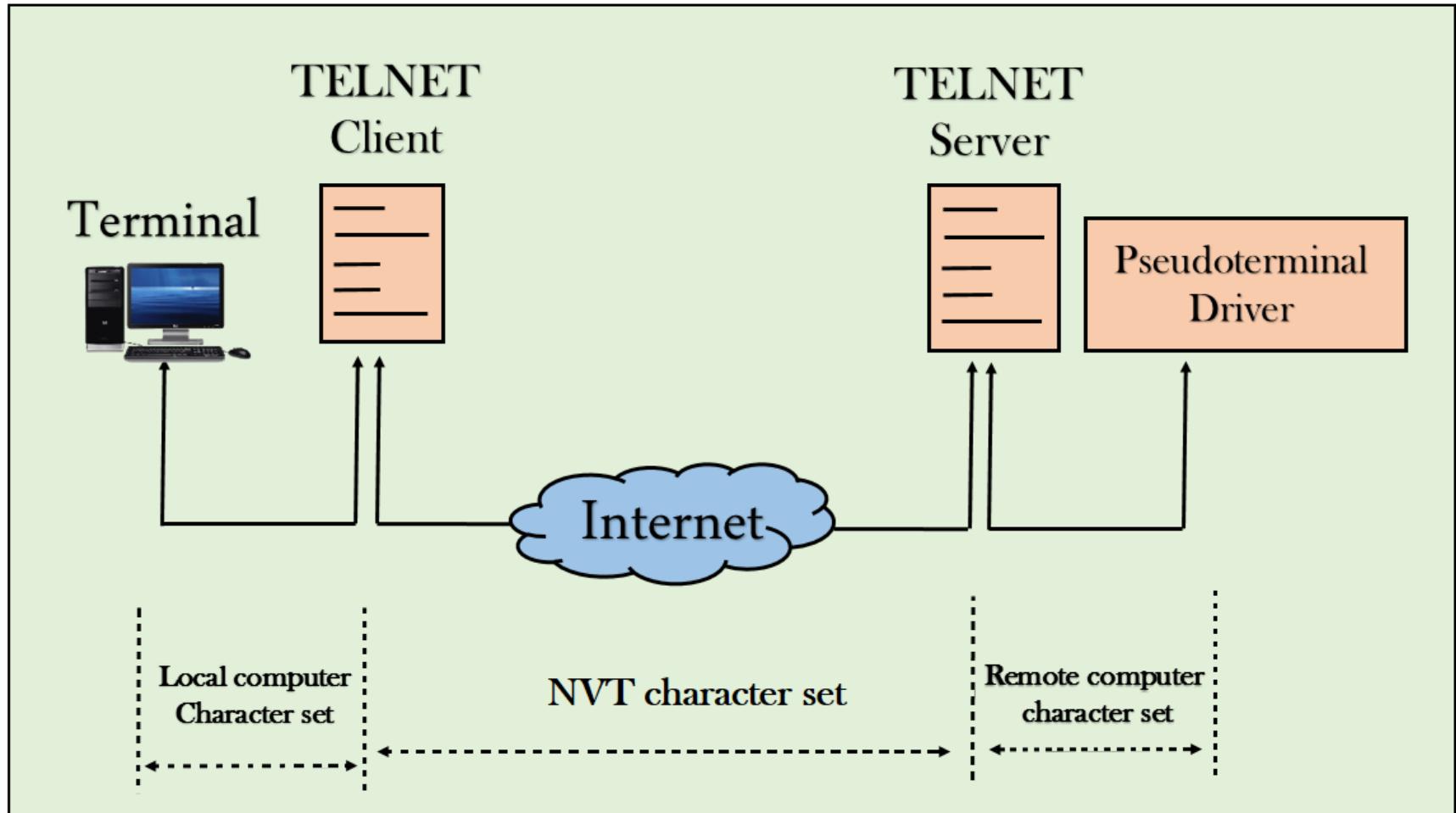
- The commands or text, in NVT form, travel through the Internet and arrive at the TCP/IP stack at the remote machine.
- Here, the characters are delivered to the operating system and passed to the TELNET server, which changes the characters to the corresponding characters understandable by the remote computer. However, the characters cannot be passed directly to the operating system because the remote operating system is not designed to receive characters from a TELNET server: It is designed to receive characters from a terminal driver.
- The solution is to add a piece of software called a pseudoterminal driver which pretends that the characters are coming from a terminal. The operating system then passes the characters to the appropriate application program.

# Network Virtual Terminal(NVT)

- The mechanism to access a remote computer is complex. This is so because every computer and its operating system accept a special combination of characters as tokens.
- For example, the end-of-file token in a computer running the DOS operating system is Ctrl+z, while the UNIX operating system recognizes Ctrl+d.
- TELNET solves this problem by defining a universal interface called the network virtual terminal (NVT) character set.
- Via this interface, the client TELNET translates characters (data or commands) that come from the local terminal into NVT form and delivers them to the network. The server TELNET, on the other hand, translates data and commands from NVT form into the form acceptable by the remote computer.

# Network Virtual Terminal(NVT)

It is illustrated in the following figure:-



# Network Virtual Terminal(NVT)

## NVT Character Set

- NVT uses two sets of characters, one for data and the other for control. Both are 8-bit bytes.
- For data, NVT is an 8-bit character set in which the 7 lowest-order bits are the same as ASCII and the highest-order bit is 0.
- To send control characters between computers (from client to server or vice versa), NVT uses an 8-bit character set in which the highest-order bit is set to 1.

# Telnet

- TELNET uses only one TCP connection. The server uses the well-known port 23, and the client uses an ephemeral port.
- The same connection is used for sending both data and control characters.
- TELNET accomplishes this by embedding the control characters in the data stream. However, to distinguish data from control characters, each sequence of control characters is preceded by a special control character called ***interpret as control (IAC)***.

# ELECTRONIC MAIL

- E-mail is defined as the transmission of messages on the Internet.
- It is one of the most commonly used features over communications networks that may contain text, files, images, or other attachments.
- Email messages are conveyed through email servers; it uses multiple protocols within the TCP/IP suite.
- For example, SMTP is a protocol, stands for simple mail transfer protocol and used to send messages whereas other protocols IMAP or POP are used to retrieve messages from a mail server.

# ELECTRONIC MAIL

## User Agent

- The first component of an electronic mail system is the user agent (VA). It provides service to the user to make the process of sending and receiving a message easier.
- A user agent is a software package (program) that composes, reads, replies to, and forwards messages. It also handles mailboxes.

## User Agent Types

There are two types of user agents: **command-driven** and **GUI-based**

# ELECTRONIC MAIL

## Sending Mail

To send mail, the user, through the UA, creates mail that looks very similar to postal mail. It has an envelope and a message.



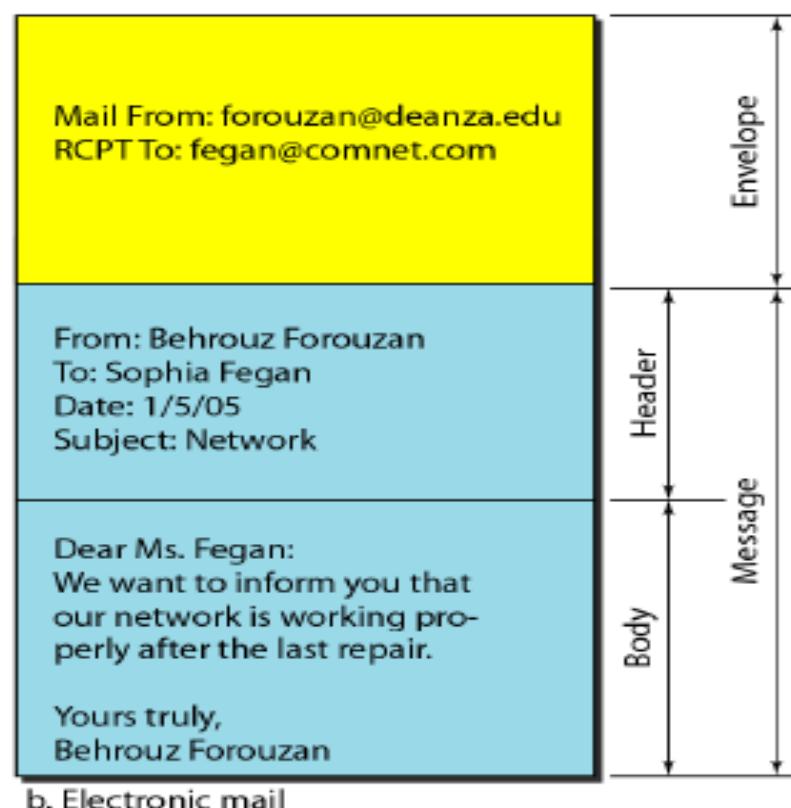
a. Postal mail

Sophia Fegan  
Com-Net  
Cupertino, CA 95014  
Jan. 5, 2005

Subject: Network

Dear Ms. Fegan:  
We want to inform you that  
our network is working pro-  
perly after the last repair.

Yours truly,  
Behrouz Forouzan



# ELECTRONIC MAIL

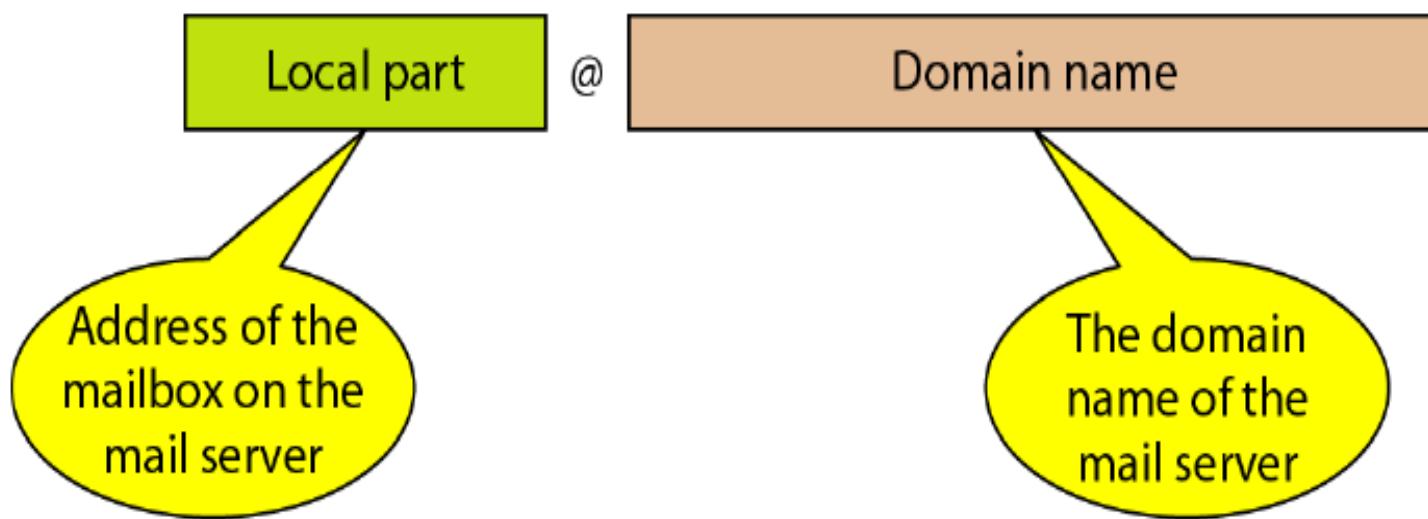
## Receiving Mail

- The user agent is triggered by the user (or a timer). If a user has mail, the UA informs the user with a notice.
- If the user is ready to read the mail, a list is displayed in which each line contains a summary of the information about a particular message in the mailbox.
- The summary usually includes the sender mail address, the subject, and the time the mail was sent or received. The user can select any of the messages and display its contents on the screen.

# ELECTRONIC MAIL

## Addresses

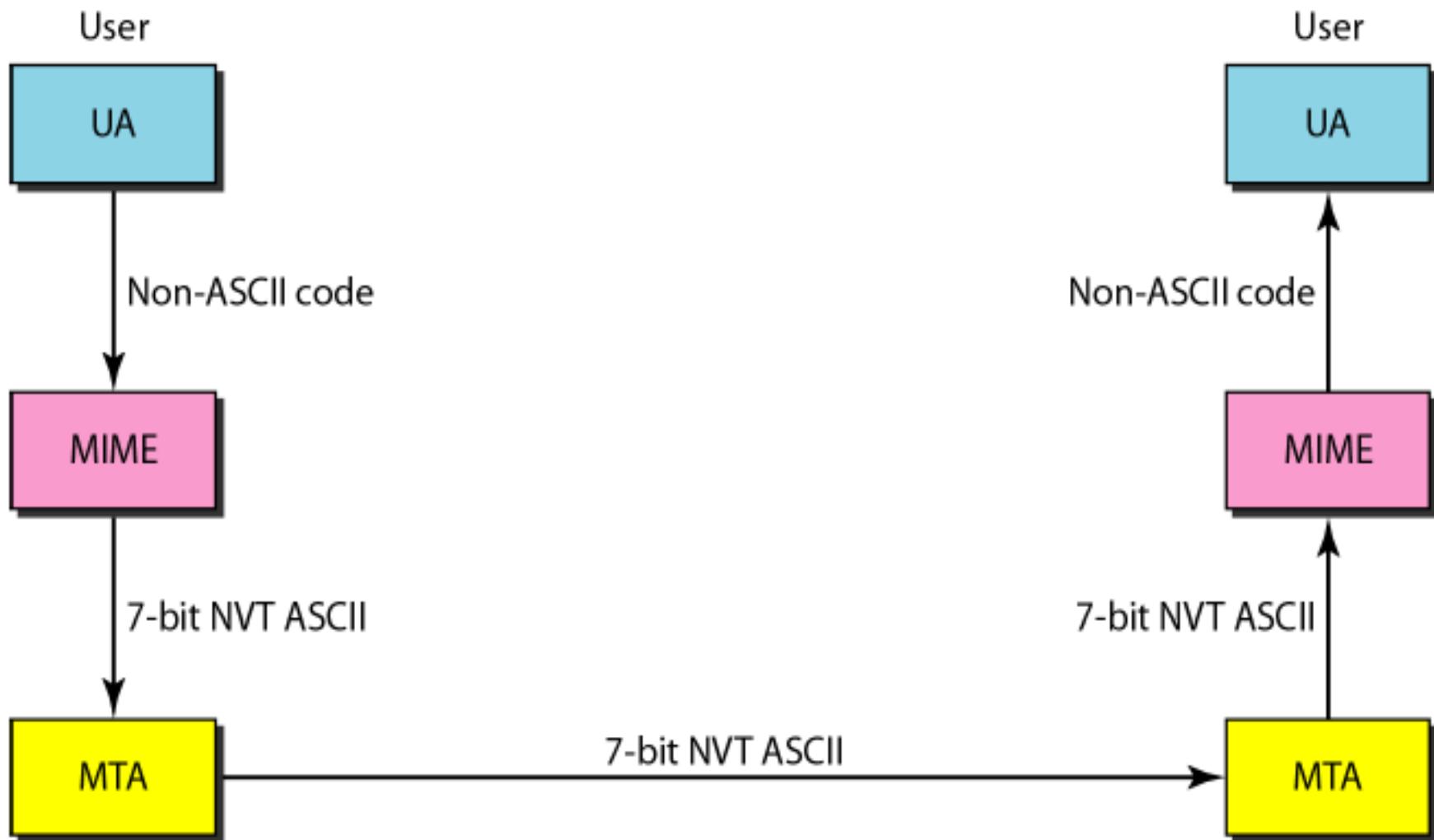
To deliver mail, a mail handling system must use an addressing system with unique addresses. In the Internet, the address consists of two parts: a local part and a domain name, separated by an @ sign.



# MIME

- Multipurpose Internet Mail Extensions (MIME) is a supplementary protocol that allows non-ASCII data to be sent through e-mail.
- MIME transforms non-ASCII data at the sender site to NVT ASCII data and delivers them to the client MTA to be sent through the Internet. The message at the receiving side is transformed back to the original data.

# MIME



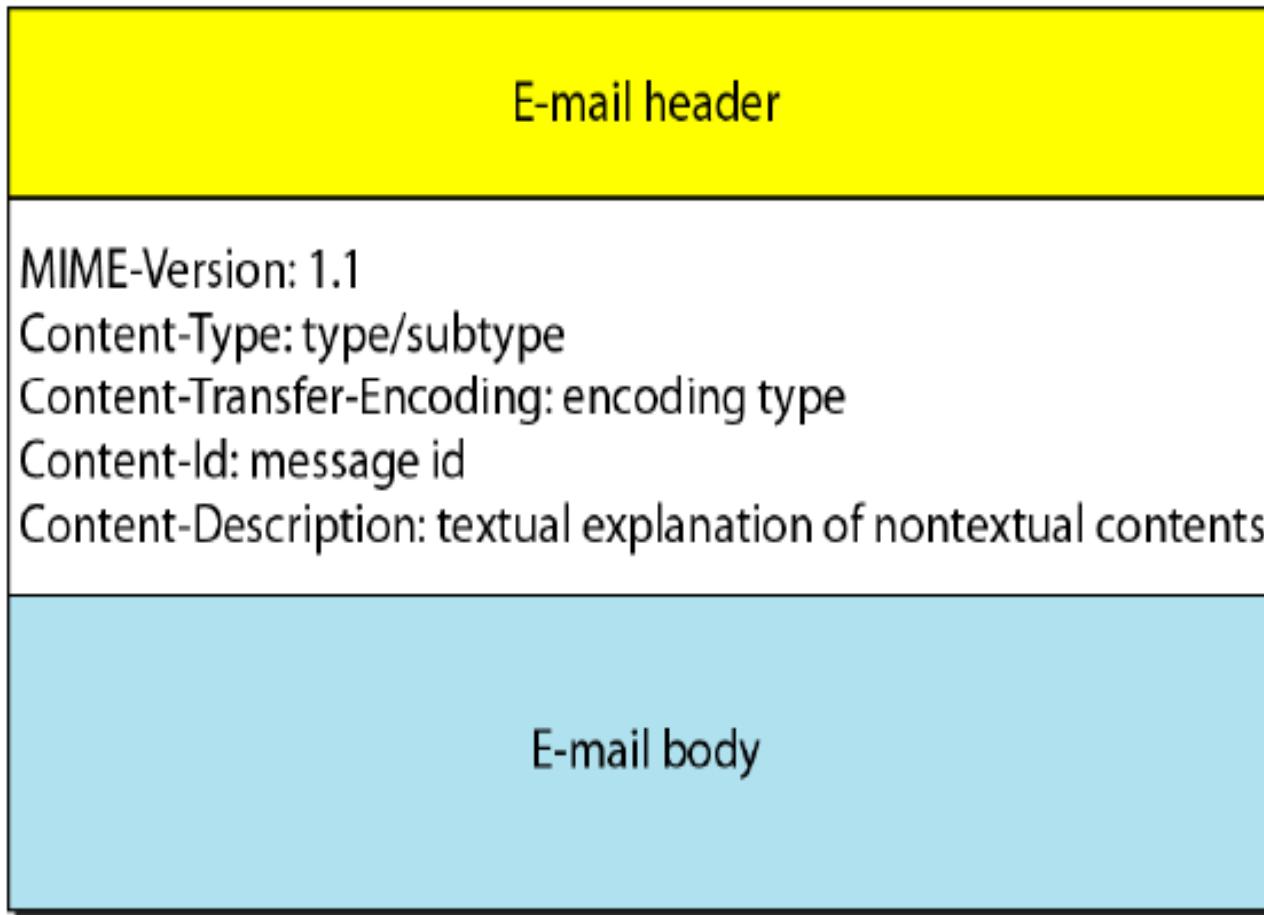
# MIME

MIME defines five headers that can be added to the original e-mail header section to define the transformation parameters:

1. MIME-Version
2. Content-Type
3. Content-Transfer-Encoding
4. Content-Id
5. Content-Description

# MIME

Following figure shows the MIME headers:-



MIME headers

# MIME

## MIME-Version

This header defines the version of MIME used. The current version is 1.1.

## Content-Type

This header defines the type of data used in the body of the message. The content type and the content subtype are separated by a slash. Depending on the subtype, the header may contain other parameters.

**Content-Type: <type/subtype; parameters> .**

MIME allows seven different types of data.

# MIME

## Content-type

Type	Subtype	Description
Text	Plain	Unformatted
	HTML	HTML format (see Chapter 27)
Multipart	Mixed	Body contains ordered parts of different data types
	Parallel	Same as above, but no order
	Digest	Similar to mixed subtypes, but the default is message/RFC822
	Alternative	Parts are different versions of the same message
Message	RFC822	Body is an encapsulated message
	Partial	Body is a fragment of a bigger message
	External-Body	Body is a reference to another message
Image	JPEG	Image is in JPEG format
	GIF	Image is in GIF format
Video	MPEG	Video is in MPEG format
Audio	Basic	Single-channel encoding of voice at 8 kHz
Application	PostScript	Adobe PostScript
	Octet-stream	General binary data (8-bit bytes)

# MIME

## Content-Transfer-Encoding

This header defines the method used to encode the messages into 0's and 1's for transport:

### **Content-Transfer-Encoding: <type>**

The five types of encoding methods are listed in the following table:-

Type	Description
7-bit	NVT ASCII characters and short lines
8-bit	Non-ASCII characters and short lines
Binary	Non-ASCII characters with unlimited-length lines
Base-64	6-bit blocks of data encoded into 8-bit ASCII characters
Quoted-printable	Non-ASCII characters encoded as an equals sign followed by an ASCII code

# MIME

## Content-Id

This header uniquely identifies the whole message in a multiple-message environment.

**Content-Id: id=<content-id>**

## Content-Description

This header defines whether the body is image, audio, or video.

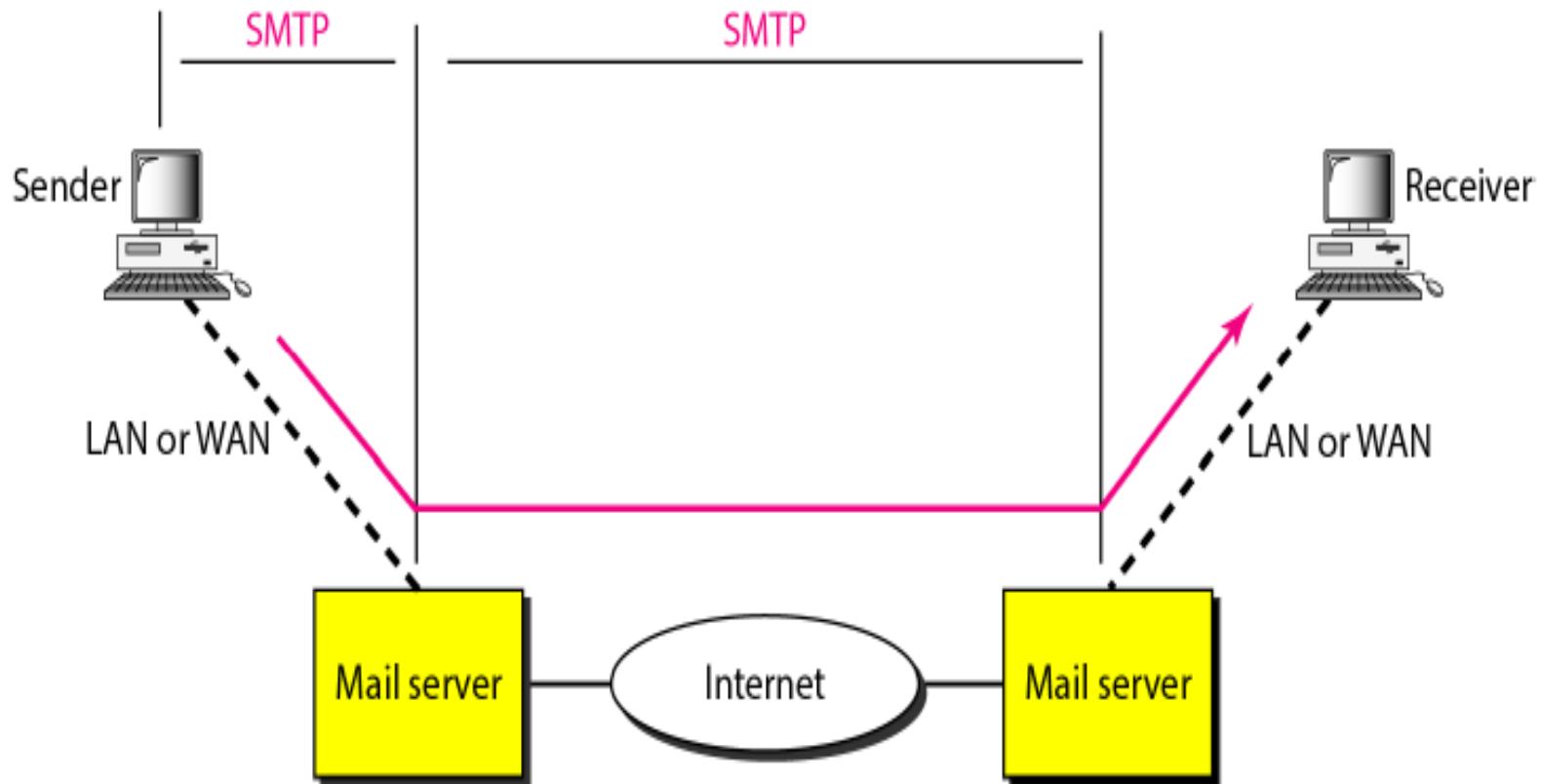
**Content-Description: <description>**

# Message Transfer Agent: SMTP

- SMTP is a message transfer agent.
- The actual mail transfer is done through message transfer agents.
- To send mail, a system must have the client MTA, and to receive mail, a system must have a server MTA.
- The protocol that defines the MTA client and server in the Internet is called the Simple Mail Transfer Protocol (SMTP).

# SMTP

Following figure shows the range of the SMTP protocol

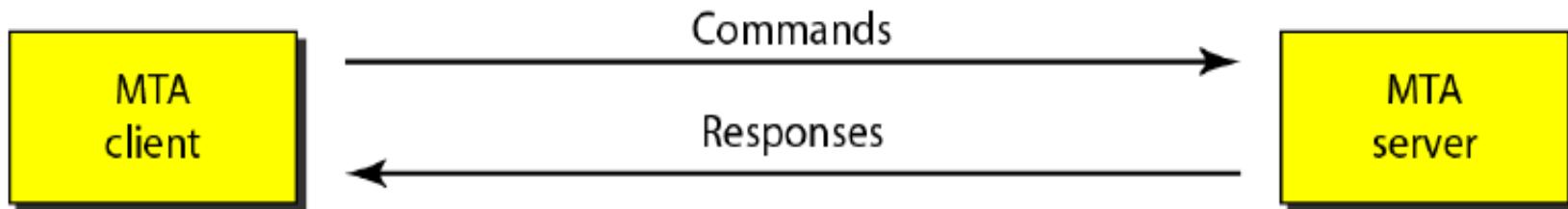


# SMTP

- SMTP is used two times, between the sender and the sender's mail server and between the two mail servers.

## Commands and Responses

SMTP uses commands and responses to transfer messages between an MTA client and an MTA server.



- Each command or reply is terminated by a two-character (carriage return and line feed) end-of-line token.

# SMTP

## Commands

Commands are sent from the client to the server. The format of a command is shown in following figure:-

**Keyword:** argument(s)

- It consists of a keyword followed by zero or more arguments.
- SMTP defines 14 commands. The first five are mandatory; every implementation must support these five commands. The next three are often used and highly recommended. The last six are seldom used.

# SMTP

## Commands

<i>Keyword</i>	<i>Argument(s)</i>
HELO	Sender's host name
MAIL FROM	Sender of the message
RCPT TO	Intended recipient of the message
DATA	Body of the mail
QUIT	
RSET	
VRFY	Name of recipient to be verified
NOOP	
TURN	
EXPN	Mailing list to be expanded
HELP	Command name
SEND FROM	Intended recipient of the message
SMOL FROM	Intended recipient of the message
SMAL FROM	Intended recipient of the message

# SMTP

## Responses

Responses are sent from the server to the client. A response is a three digit code that may be followed by additional textual information. Following table lists some of the responses:-

<i>Code</i>	<i>Description</i>
<b>Positive Completion Reply</b>	
<b>211</b>	System status or help reply
<b>214</b>	Help message
<b>220</b>	Service ready
<b>221</b>	Service closing transmission channel
<b>250</b>	Request command completed
<b>251</b>	User not local; the message will be forwarded
<b>Positive Intermediate Reply</b>	
<b>354</b>	Start mail input
<b>Transient Negative Completion Reply</b>	
<b>421</b>	Service not available
<b>450</b>	Mailbox not available
<b>451</b>	Command aborted: local error
<b>452</b>	Command aborted: insufficient storage

# SMTP

## Responses(continue)

<i>Code</i>	<i>Description</i>
<b>Permanent Negative Completion Reply</b>	
<b>500</b>	Syntax error; unrecognized command
<b>501</b>	Syntax error in parameters or arguments
<b>502</b>	Command not implemented
<b>503</b>	Bad sequence of commands
<b>504</b>	Command temporarily not implemented
<b>550</b>	Command is not executed; mailbox unavailable
<b>551</b>	User not local
<b>552</b>	Requested action aborted; exceeded storage location
<b>553</b>	Requested action not taken; mailbox name not allowed
<b>554</b>	Transaction failed

- As the table shows, responses are divided into four categories. The leftmost digit of the code (2, 3, 4, and 5) defines the category.

# SMTP

## Mail Transfer Phases

The process of transferring a mail message occurs in three phases: connection establishment, mail transfer, and connection termination.

## Example

Let us see how we can directly use SMTP to send an e-mail and simulate the commands and responses we described in this section. We use TELNET to log into port 25 (well known port for SMTP). We then use the commands directly to send an e-mail. In this example, `forouzanb@adelphia.net` is sending an e-mail to himself. The first few lines show TELNET trying to connect to the Adelphia mail server.

After connection, we can type the SMTP commands and then receive the responses, as shown below. We have shown the commands in black and the responses in color. Note that we have added, for clarification, some comment lines, designated by the "=" signs. These lines are not part of the e-mail procedure.

# SMTP

```
$ telnet mail.adelphia.net 25
```

```
Trying 68.168.78.100...
```

```
Connected to mail.adelphia.net (68.168.78.100).
```

```
===== Connection Establishment =====
```

```
220 mta13.adelphia.net SMTP server ready Fri, 6 Aug 2004 ...
```

```
HELO mail.adelphia.net
```

```
250 mta13.adelphia.net
```

# SMTP

===== Mail Transfer =====

**MAIL FROM:** forouzanb@adelphia.net

**250 Sender <forouzanb@adelphia.net> Ok**

**RCPT TO:** forouzanb@adelphia.net

**250 Recipient <forouzanb@adelphia.net> Ok**

**DATA**

**354 Ok Send data ending with <CRLF>.<CRLF>**

**From:** Forouzan

**TO:** Forouzan

This is a test message  
to show SMTP in action.

.

===== Connection Termination =====

**250 Message received: adelphia.net@mail.adelphia.net**

**QUIT**

**221 mta13.adelphia.net SMTP server closing connection**

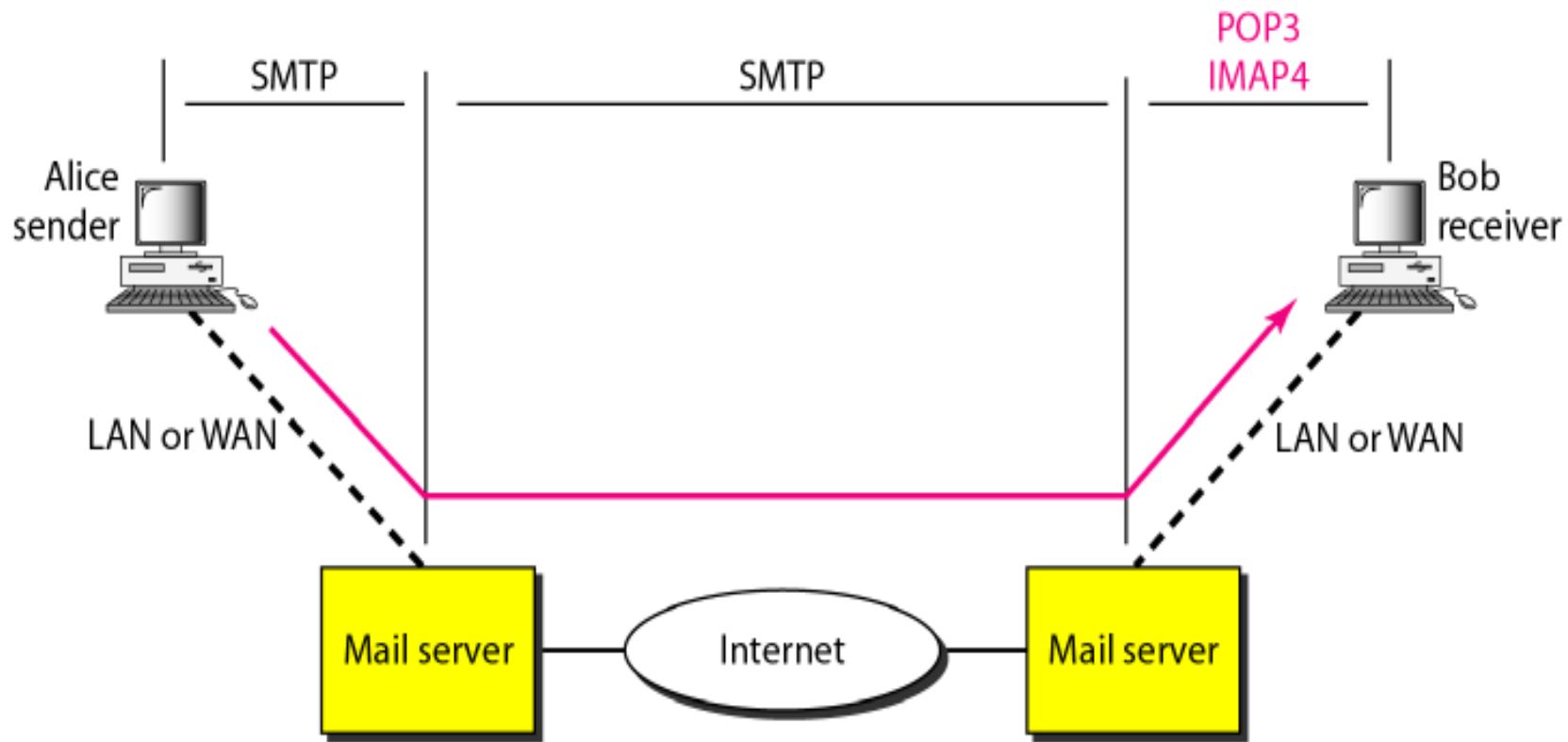
**Connection closed by foreign host.**

# Message Access Agent: POP and IMAP

- The first and the second stages of mail delivery use SMTP. However, SMTP is not involved in the third stage because SMTP is a push protocol; it pushes the message from the client to the server. In other words, the direction of the bulk: data (messages) is from the client to the server.
- On the other hand, the third stage needs a pull protocol; the client must pull messages from the server. The direction of the bulk data is from the server to the client. The third stage uses a message access agent.

# Message Access Agent: POP and IMAP

Currently two message access protocols are available: Post Office Protocol, version 3 (POP3) and Internet Mail Access Protocol, version 4 (IMAP4). Following figure shows the position of these two protocols in the most common situation.

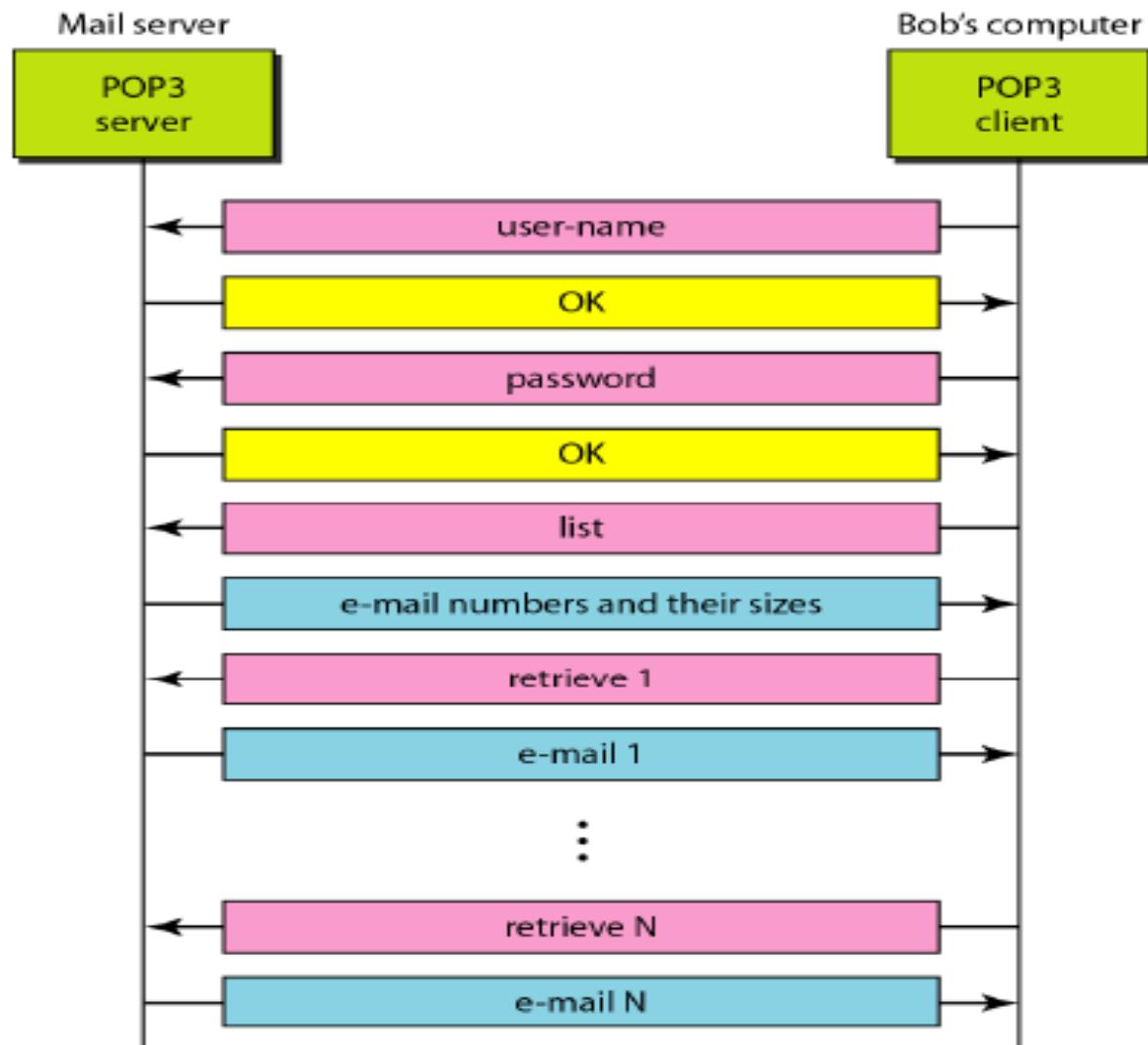


# POP3

- Post Office Protocol, version 3 (POP3) is simple and limited in functionality. The client POP3 software is installed on the recipient computer; the server POP3 software is installed on the mail server.
- Mail access starts with the client when the user needs to download e-mail from the mailbox on the mail server. The client opens a connection to the server on TCP port 110. It then sends its user name and password to access the mailbox. The user can then list and retrieve the mail messages, one by one.

# POP3

Following figure shows an example of downloading using POP3



# POP3

- POP3 has two modes: the delete mode and the keep mode.
- In the delete mode, the mail is deleted from the mailbox after each retrieval.
- In the keep mode, the mail remains in the mailbox after retrieval.
- The delete mode is normally used when the user is working at her permanent computer and can save and organize the received mail after reading or replying.
- The keep mode is normally used when the user accesses her mail away from her primary computer (e.g., a laptop). The mail is read but kept in the system for later retrieval and organizing.

# POP3

## Deficiencies:

- It does not allow the user to organize her mail on the server.
- The user cannot have different folders on the server.
- In addition, POP3 does not allow the user to partially check the contents of the mail before downloading.

# IMAP4

- Another mail access protocol is Internet Mail Access Protocol, version 4 (IMAP4). IMAP4 is similar to POP3, but it has more features. IMAP4 is more powerful and more complex.
- IMAP4 provides the following extra functions:
  - A user can check the e-mail header prior to downloading.
  - A user can search the contents of the e-mail for a specific string of characters prior to downloading.

# **IMAP4**

- A user can partially download e-mail. This is especially useful if bandwidth is limited and the e-mail contains multimedia with high bandwidth requirements.
- A user can create, delete, or rename mailboxes on the mail server.
- A user can create a hierarchy of mailboxes in a folder for e-mail storage.

# Web-Based Mail

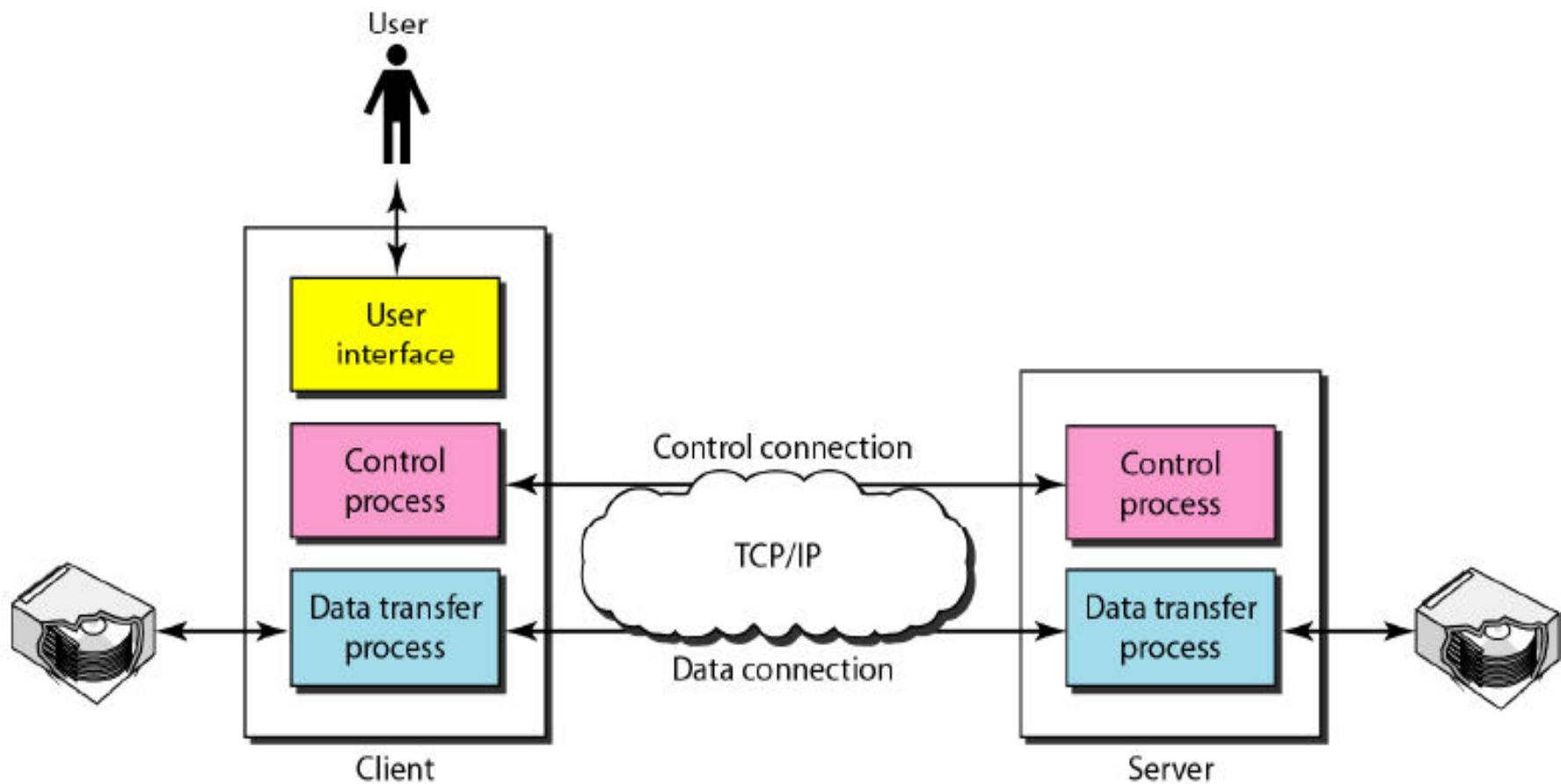
- E-mail is such a common application that some websites today provide this service to anyone who accesses the site.
- Suppose the mail is transferred from Alice to Bob.
  - Mail transfer from Alice's browser to her mail server is done through HTTP.
  - The transfer of the message from the sending mail server to the receiving mail server is still through SMTP.
  - Finally, the message from the receiving server (the Web server) to Bob's browser is done through HTTP.
- In the last phase, instead of POP3 or IMAP4, HTTP is normally used.

# File Transfer Protocol (FTP)

- File Transfer Protocol (FTP) is the standard mechanism provided by TCP/IP for copying a file from one host to another.
- FTP differs from other client/server applications in that it establishes two connections between the hosts. One connection is used for data transfer, the other for control information (commands and responses). Separation of commands and data transfer makes FTP more efficient.
- The control connection uses very simple rules of communication. We need to transfer only a line of command or a line of response at a time.
- The data connection, on the other hand, needs more complex rules due to the variety of data types transferred.
- However, the difference in complexity is at the FTP level, not TCP.
- For TCP, both connections are treated the same.

# File Transfer Protocol (FTP)

- FTP uses two well-known TCP ports: Port **21** is used for the control connection, and port **20** is used for the data connection.
- Following figure shows the basic model of FTP:-



# File Transfer Protocol (FTP)

- The client has three components: user interface, client control process, and the client data transfer process.
- The server has two components: the server control process and the server data transfer process.
- The control connection is made between the control processes. The data connection is made between the data transfer processes.

# File Transfer Protocol (FTP)

- The control connection remains connected during the entire interactive FTP session.
- The data connection is opened and then closed for each file transferred. It opens each time commands that involve transferring files are used, and it closes when the file is transferred.
- In other words, when a user starts an FTP session, the control connection opens. While the control connection is open, the data connection can be opened and closed multiple times if several files are transferred.

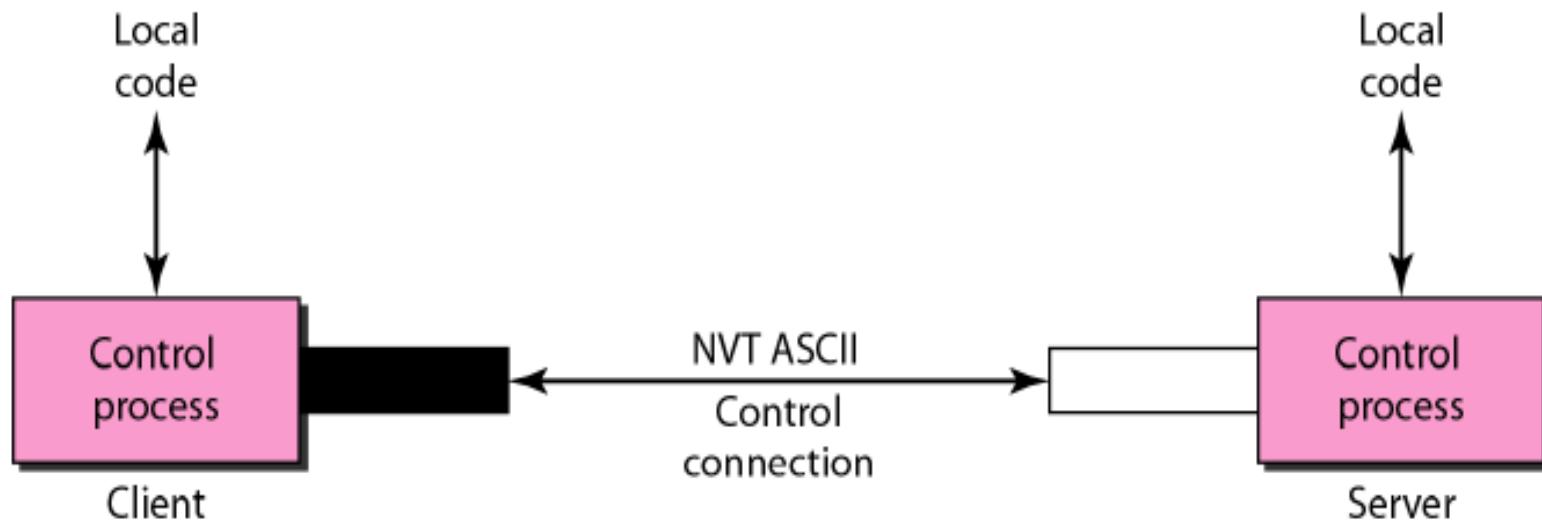
# File Transfer Protocol (FTP)

## Communication over Control Connection

- FTP uses the same approach as SMTP to communicate across the control connection.
- It uses the 7-bit ASCII character set.
- Communication is achieved through commands and responses.
- This simple method is adequate for the control connection because we send one command (or response) at a time.

# File Transfer Protocol (FTP)

- Each command or response is only one short line, so we need not worry about file format or file structure.
- Each line is terminated with a two-character (carriage return and line feed) end-of-line token.



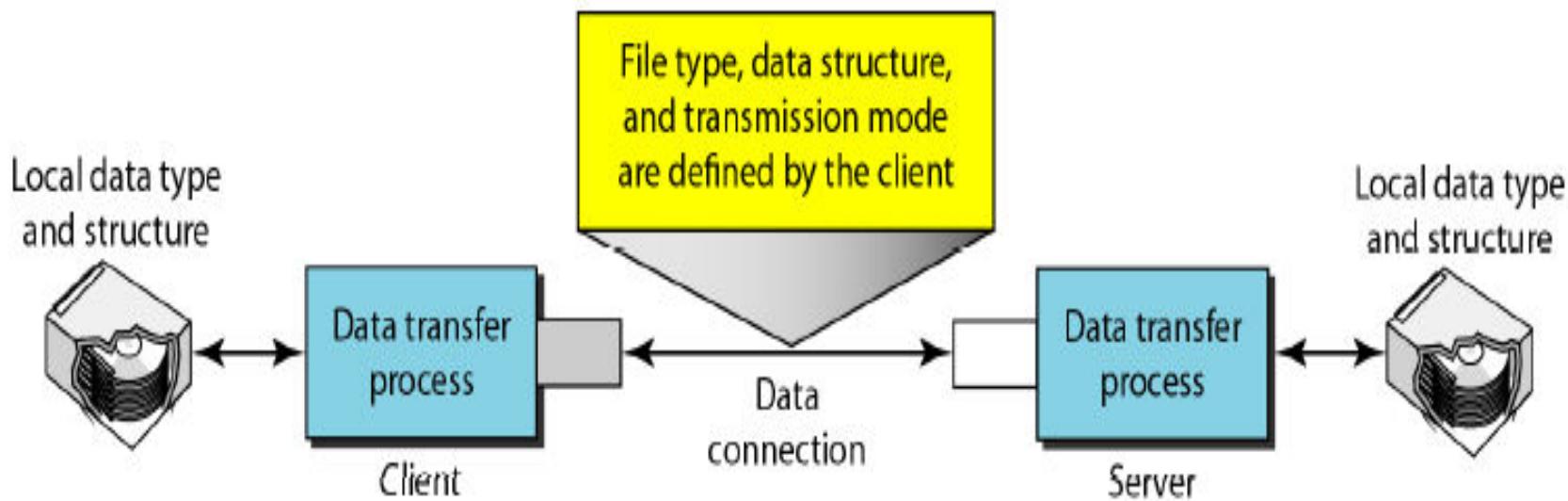
# File Transfer Protocol (FTP)

## Communication over Data Connection

- File transfer occurs over the data connection under the control of the commands sent over the control connection.
- File transfer in FTP means one of three things:
  - A file is to be copied from the server to the client. This is called retrieving a file. It is done under the supervision of the **RETR** command.
  - A file is to be copied from the client to the server. This is called storing a file. It is done under the supervision of the **STOR** command.
  - A list of directory or file names is to be sent from the server to the client. This is done under the supervision of the **LIST** command.

# File Transfer Protocol (FTP)

- The client must define the type of file to be transferred, the structure of the data, and the transmission mode.



# File Transfer Protocol (FTP)

## File Type

FTP can transfer one of the following file types across the data connection: an ASCII file, EBCDIC file, or image file.

- The ASCII file is the default format for transferring text files. Each character is encoded using 7-bit ASCII. The sender transforms the file from its own representation into ASCII characters, and the receiver transforms the ASCII characters to its own representation.
- If one or both ends of the connection use EBCDIC encoding (the file format used by IBM), the file can be transferred using EBCDIC encoding.
- The image file is the default format for transferring binary files. The file is sent as continuous streams of bits without any interpretation or encoding. This is mostly used to transfer binary files such as compiled programs.

# File Transfer Protocol (FTP)

## Data Structure

FTP can transfer a file across the data connection by using one of the following interpretations about the structure of the data: file structure, record structure, and page structure.

- In the **file structure** format, the file is a continuous stream of bytes.
- In the **record structure**, the file is divided into records. This can be used only with text files.
- In the **page structure**, the file is divided into pages, with each page having a page number and a page header. The pages can be stored and accessed randomly or sequentially.

# File Transfer Protocol (FTP)

## Transmission Mode

FTP can transfer a file across the data connection by using one of the following three transmission modes: stream mode, block mode, and compressed mode.

- The **stream mode** is the default mode. Data are delivered from FTP to TCP as a continuous stream of bytes. TCP is responsible for chopping data into segments of appropriate size. If the data are simply a stream of bytes (file structure), no end-of-file is needed. End-of-file in this case is the closing of the data connection by the sender. If the data are divided into records (record structure), each record will have a 1-byte end-of-record (EOR) character and the end of the file will have a 1-byte end-of-file (EOF) character.

# File Transfer Protocol (FTP)

## Transmission Mode(continue)

- In **block mode**, data can be delivered from FTP to TCP in blocks. In this case, each block is preceded by a 3-byte header. The first byte is called the block descriptor; the next 2 bytes define the size of the block in bytes.
- In **the compressed mode**, if the file is big, the data can be compressed. The compression method normally used is run-length encoding. In this method, consecutive appearances of a data unit are replaced by one occurrence and the number of repetitions.

# File Transfer Protocol (FTP)

## Example

The following shows an actual FTP session for retrieving a list of items in a directory. The colored lines show the responses from the server control connection; the black lines show the commands sent by the client. The lines in white with a black background show data transfer.

# File Transfer Protocol (FTP)

```
$ ftp voyager.deanza.fhda.edu
```

```
Connected to voyager.deanza.fhda.edu.
```

```
220 (vsFTPd 1.2.1)
```

```
530 Please login with USER and PASS.
```

```
Name (voyager.deanza.fhda.edu:forouzan): forouzan
```

```
331 Please specify the password.
```

```
Password:
```

```
230 Login successful.
```

```
Remote system type is UNIX.
```

```
Using binary mode to transfer files.
```

```
ftp> ls reports
```

```
227 Entering Passive Mode (153,18,17,11,238,169)
```

```
150 Here comes the directory listing.
```

drwxr-xr-x	2	3027	411	4096	Sep 24	2002	business
drwxr-xr-x	2	3027	411	4096	Sep 24	2002	personal
drwxr-xr-x	2	3027	411	4096	Sep 24	2002	school

```
226 Directory send OK.
```

```
ftp> quit
```

```
221 Goodbye.
```

# File Transfer Protocol (FTP)

1. After the control connection is created, the FIP server sends the 220 (service ready) response on the control connection.
2. The client sends its name.
3. The server responds with 331 (user name is OK, password is required).
4. The client sends the password (not shown).
5. The server responds with 230 (user log-in is OK).
6. The client sends the list command (ls reports) to find the list of files on the directory named reports.

# File Transfer Protocol (FTP)

7. Now, the server responds with 150 and opens the data connection.
8. The server then sends the list of the files or directories (as a file) on the data connection. When the whole list (file) is sent, the server responds with 226 (closing data connection) over the control connection.
9. The client now has two choices. It can use the QUIT command to request the closing of the control connection, or it can send another command to start another activity (and eventually open another data connection). In our example, the client sends a QUIT command.
10. After receiving the QUIT command, the server responds with 221 (service closing) and then closes the control connection.

# **File Transfer Protocol (FTP)**



# AKTU Examination Questions

1. Mention the use of HTTP.
2. List out few email gateways.
3. Elaborate about TELNET and its working procedure.
4. How does FTP work? Differentiate between passive and active FTP.
5. Explain the SNMP protocols in detail.
6. How is TFTP different from FTP?
7. What three functions can SNMP perform to manage network devices?
8. How is the BOOTP different from DHCP?
9. What is the purpose of the Domain Name System? Discuss the three main divisions of the domain name space.
10. Write short notes on any two: (i) SMTP (ii) TELNET (iii) HTTP

# AKTU Examination Questions

11. Elaborate about TELNET and its working procedure.
12. Write short notes on any two of the following:
  - i. DNS in the internet
  - ii. Voice Over IP
  - iii. File Transfer Protocol
13. Explain the SNMP protocols in detail.
14. What do you mean by DNS?
15. The symbols & their frequencies are given below

Symbol	A	B	C	D	E	F	G	H
Frequency	20	28	16	15	15	10	4	2

Construct Huffman codes.

# AKTU Examination Questions

16. Encrypt “EXTRANETPLANETSOURCE” using a transposition cipher with the following key: 3 5 2 1 4
17. Explain the following:
- (i) Telnet
  - (ii) FTP
  - (iii) SNMP
  - (iv) HTTP
  - (v) MIME
18. How does DNS perform data name resolution?  
What are the different types of name servers?  
Mention the DNS message format for query and reply messages.

# AKTU Examination Questions

19. Write short notes on any three of the following:

- (i) DNS in the Internet
- (ii) Voice Over IP
- (iii) SNMP
- (iv) Electronic mail
- (v) File Transfer Protocol