

# **COMPUTER NETWORKS**

## **MODULE I**

---

Jincy J Fernandez

Asst. Professor- CSE

RSET

# Topics

---

- Network Software
- The OSI reference model
- Services in the OSI model
- TCP/ IP reference model

# Network Software- Protocol Hierarchies

---

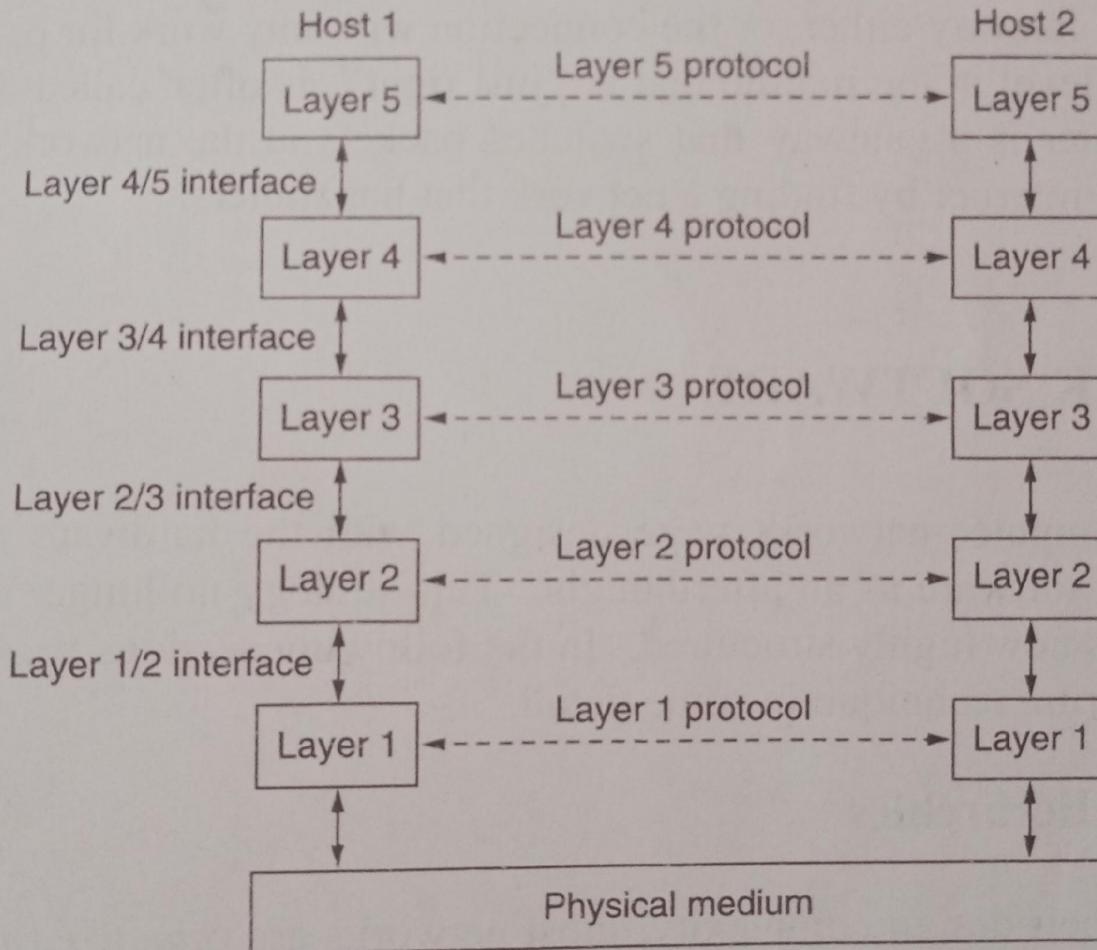
- Layered approach: stack of layers or levels.
- The number of layers, name of the layer, functions of each layer differ from network to network.
- Offer services to the higher layers.
- Protocol: agreement between the communicating parties on how the communication is to proceed.
- Data and control information are passed to the lower layers.

# Network Software- Protocol Hierarchies

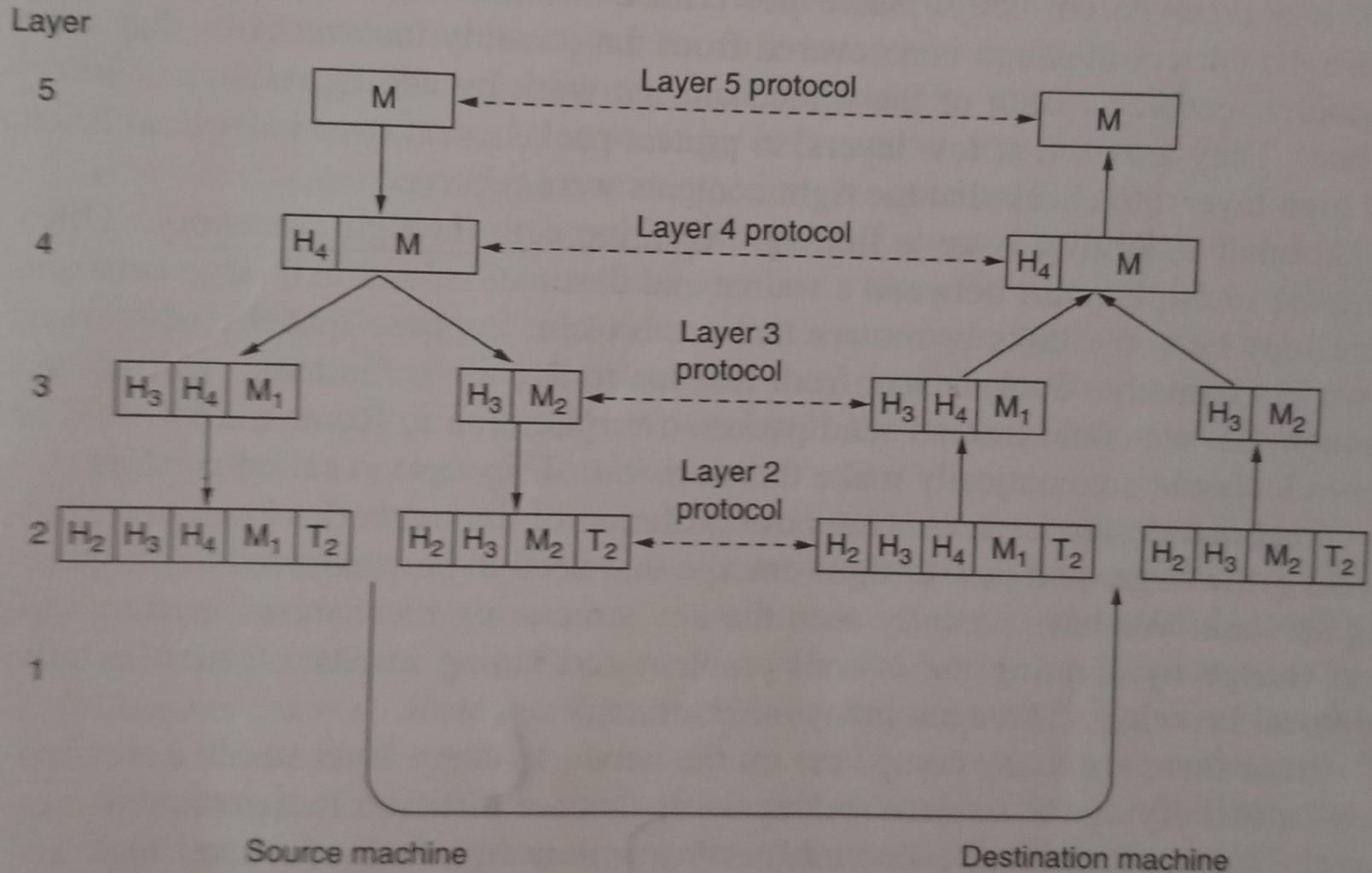
---

- **Interfaces:** the primitive operations and services the lower layers make available to the upper layers
- **Network architecture:** a set of layers and protocols.
- **Protocol stack:** a list of protocols used by a certain system, one protocol per layer.

# Network Software- Protocol Hierarchies



# Network Software- Protocol Hierarchies



# Network Software- Design issues for the layers

## 1. Reliability:

- Design an error free network.
- Error detection and error correction mechanisms.
- Routing

## 2. Addressing

- Identify the sender's and receiver's address.
- Scalability issue.

# Network Software- Design issues for the layers

---

## 3. Resource Allocation:

- Allocate the resources available.
- Chances of congestion

## 4. Security

- Secure the network against threats
- Ensure confidentiality and authentication.

# Network Software- Types of Services

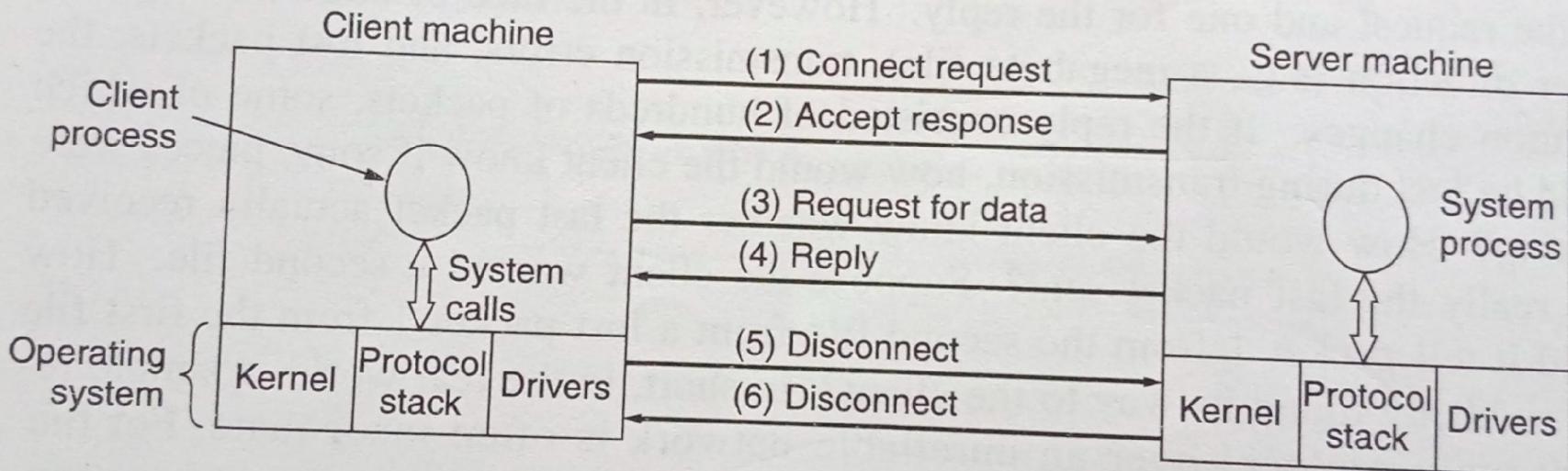
Connection Oriented service	Connectionless Service
Connection-oriented service is related to the telephone system.	Connectionless service is related to the postal system.
Three phase process: a) Connection establishment b) Data transfer c) Termination	Single phase process: Data transfer
Reliable and secure	No guarantee of reliability and security
Packets follow the same route	Packets does not follow the same route
Requires authentication	Does not require authentication
E.g. TCP (Transmission Control Protocol)	UDP (User Datagram Protocol)

# Network Software- Service Primitives

---

- Service = set of primitives provided by one layer to layer above.
- Service defines what layer can do (but not how it does it).
- Primitives for connection oriented service are different from those of connectionless service.

# Network Software- Service Primitives



# Network Software- Relationship of Service to protocols

---

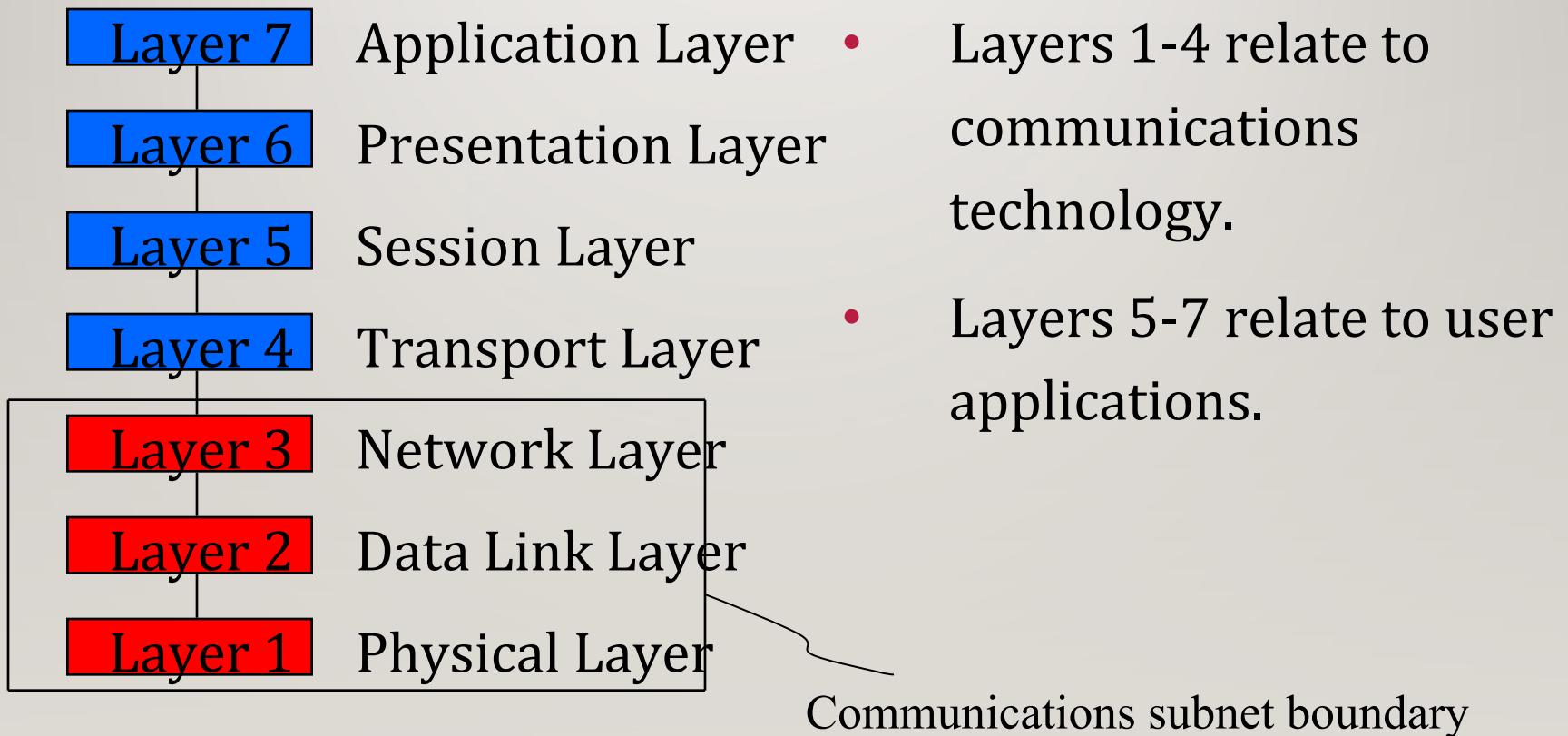
- Service = set of primitives provided by one layer to layer above.
- Service defines what layer can do (but not how it does it).
- Protocol defines set of rules governing the format and meaning of the packets.

# OSI Reference Model

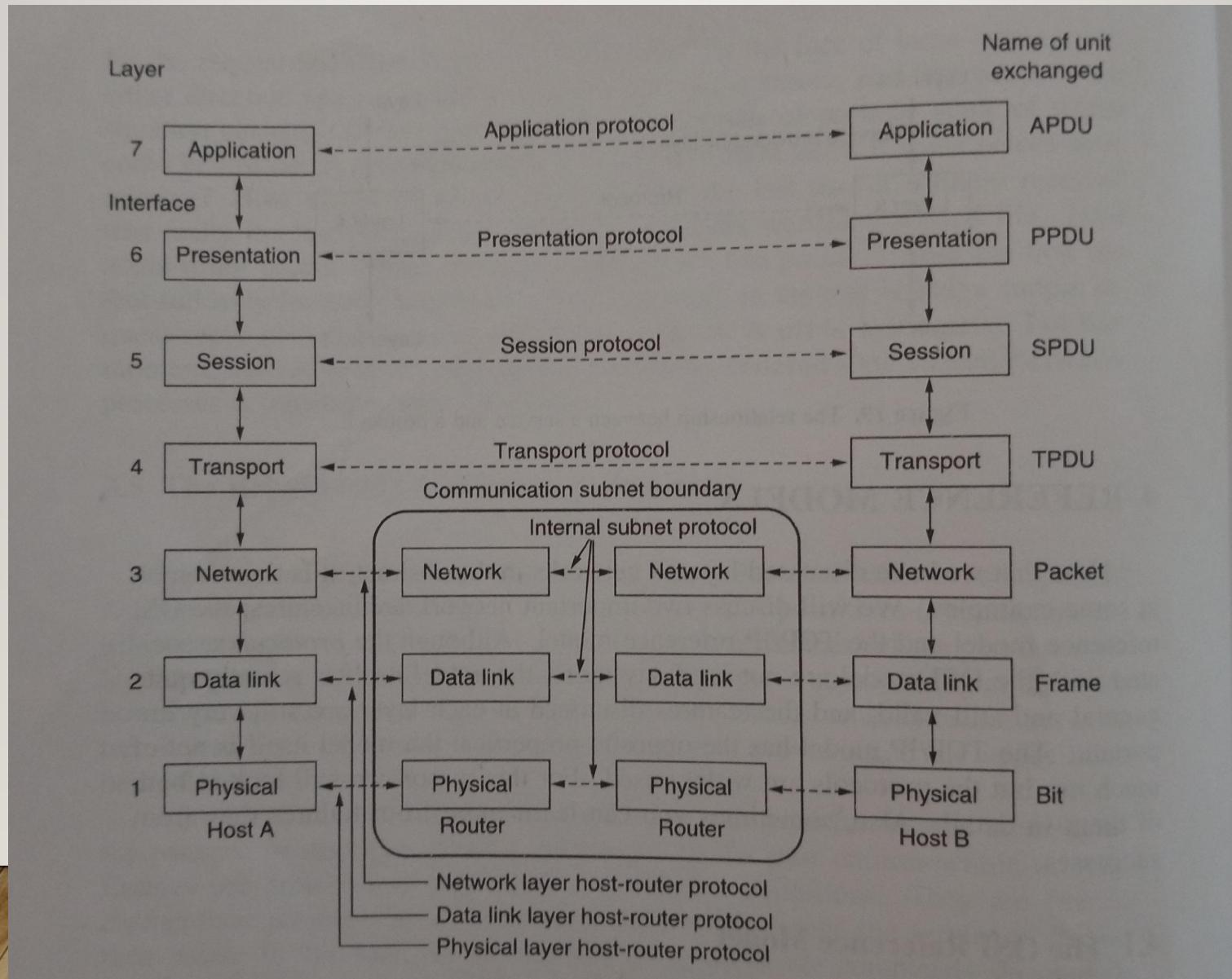
---

- OSI Reference Model - internationally standardised network architecture.
- **OSI = *Open Systems Interconnection*:** deals with *open systems*, i.e. systems open for communications with other systems.
- Developed by International Standards Organization.
- Model has 7 layers.
- Each layer has specific functionality to perform.

# 7-Layer OSI Model



# 7-Layer OSI Model



# Layer 1: Physical Layer

---

- Responsible for actual physical connection between the devices.
- Transmits bits from one computer to another.
- Regulates the transmission of a stream of bits over a physical medium.
- Defines how the cable is attached to the network adapter and what transmission technique is used to send data over the cable.

# Layer 1: Physical Layer

---

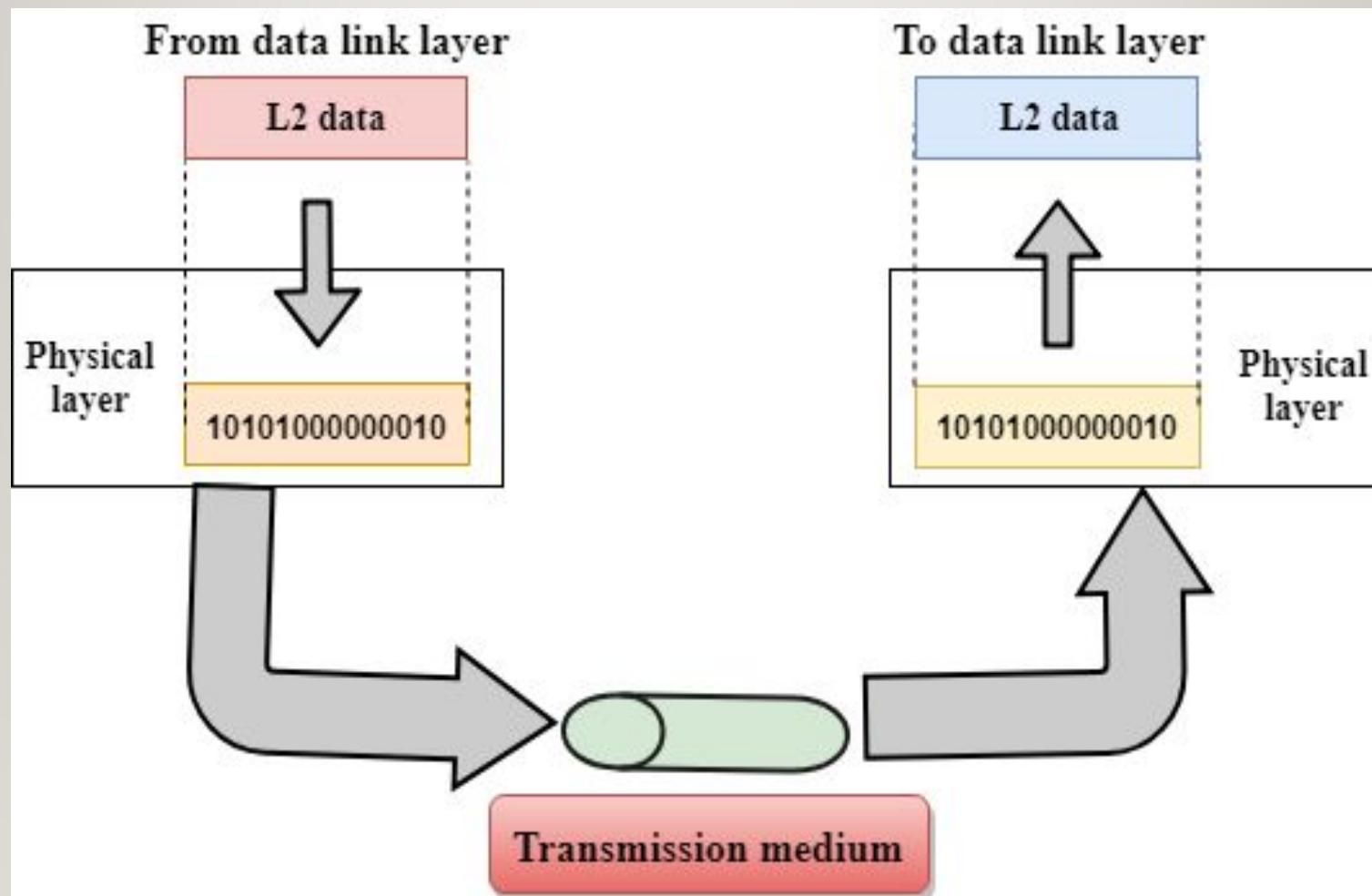
- Deals with issues like
  - The definition of 0 and 1, e.g. how many volts represents a 1, and how long a bit lasts?
  - Whether the channel is simplex or duplex?
  - How many pins a connector has, and what the function of each pin is?

# Layer 1: Physical Layer- Functions

---

- Bit Synchronization:
  - Use of clock that controls both sender and receiver.
- Bit rate control
  - Number of bits sent per second
- Physical topologies:
  - The way in which different nodes are arranged in a network.
- Transmission mode:
  - Simplex / Half Duplex / Duplex

# Layer 1: Physical Layer- Functions



# Layer 2: Data Link Layer- Functions

---

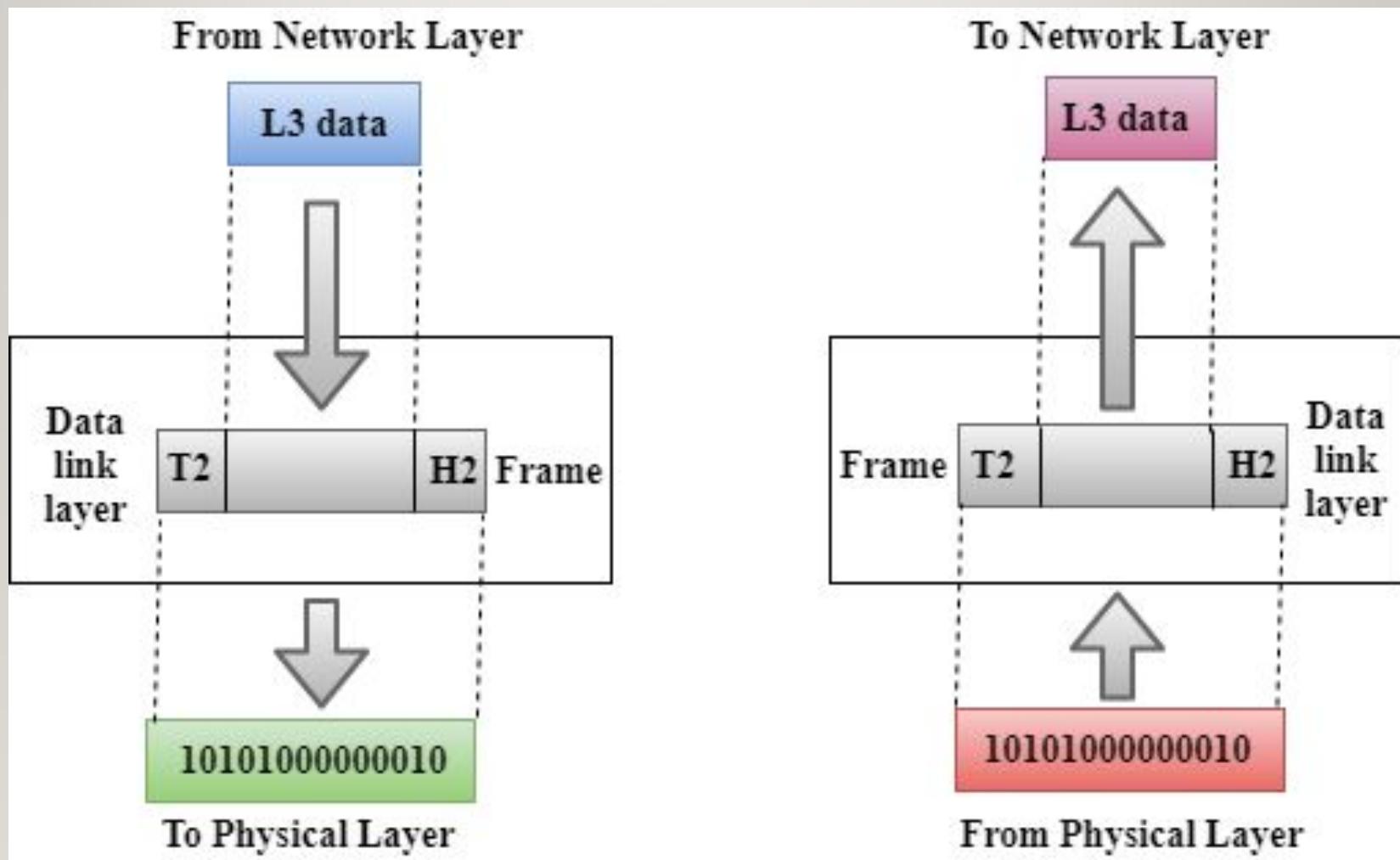
- **Framing:**
  - Transmit a set of bits that are meaningful to the receiver.
  - Attach special bit patterns to the beginning and end of the frame.
- **Physical addressing:**
  - Add the physical address of the sender and receiver in the header of each frame.

# Layer 2: Data Link Layer- Functions

---

- Error Control:
  - Mechanism of error control: detect errors and retransmit damaged or lost frames.
- Flow Control:
  - Data rate must be constant at both the sides.
  - Coordinates the amount of data that can be sent before receiving acknowledgement.

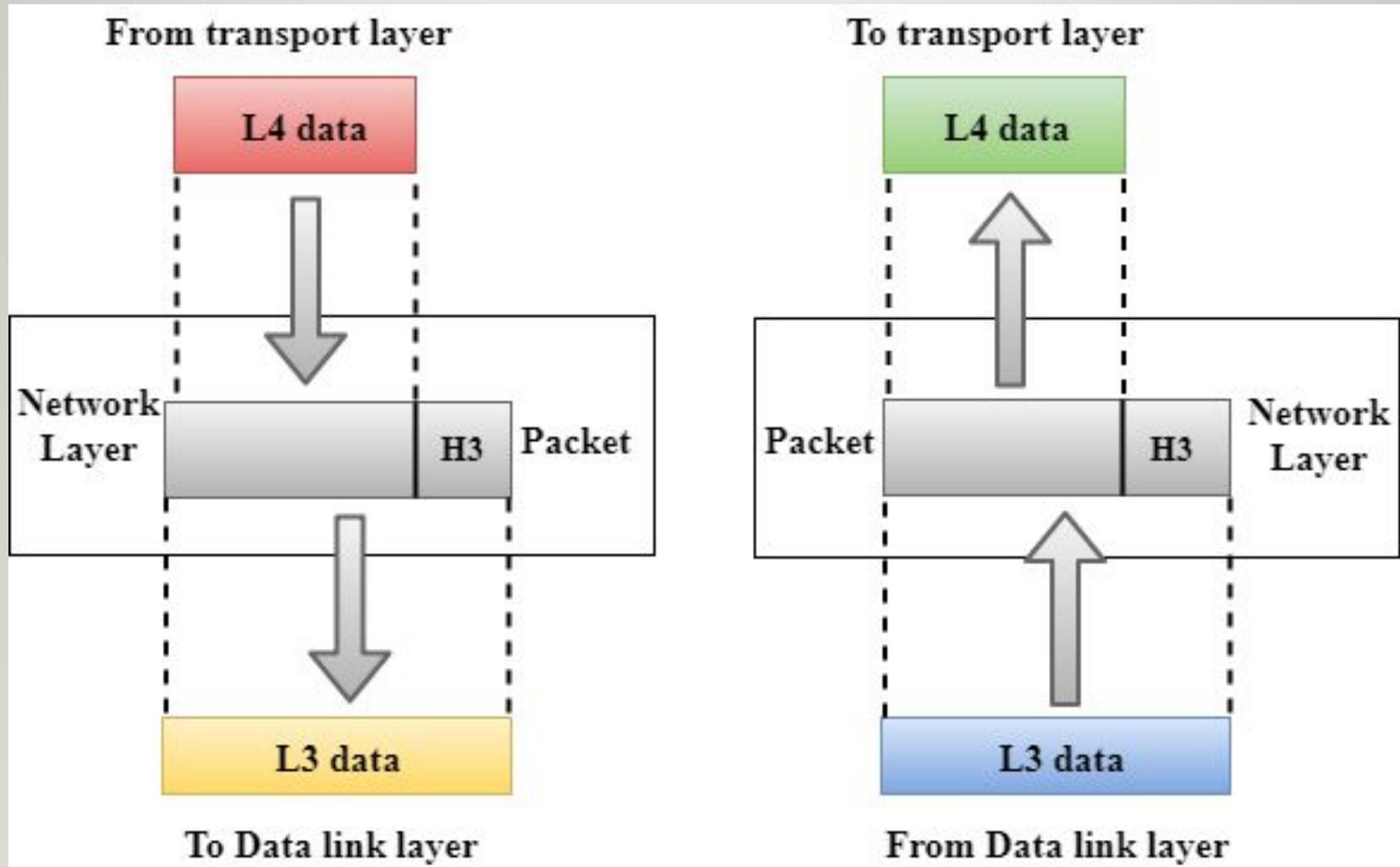
# Layer 2: Data Link Layer- Functions



# Layer 3: Network Layer

- Track the location of the devices on the network.
- Transmission of data from one host to the other located in different networks.
- Packet routing: selection of the shortest path to transmit the packet.
- Logical Addressing: The IP address of the sender and receiver are placed in the header.
- Manages traffic problems and controlling the congestion of data packets.

# Layer 3: Network Layer- Functions



# Layer 4: Transport Layer

---

- Heart of the OSI model.
- Manages transmission packets
  - Repackages long messages into small packets for transmission.
  - Reassembles packets in correct order to get the original message.
- Handles error recognition and recovery.
  - Provides acknowledgement of successful data transmission.
  - Resends missing packets.

# Layer 4: Transport Layer-Functions

---

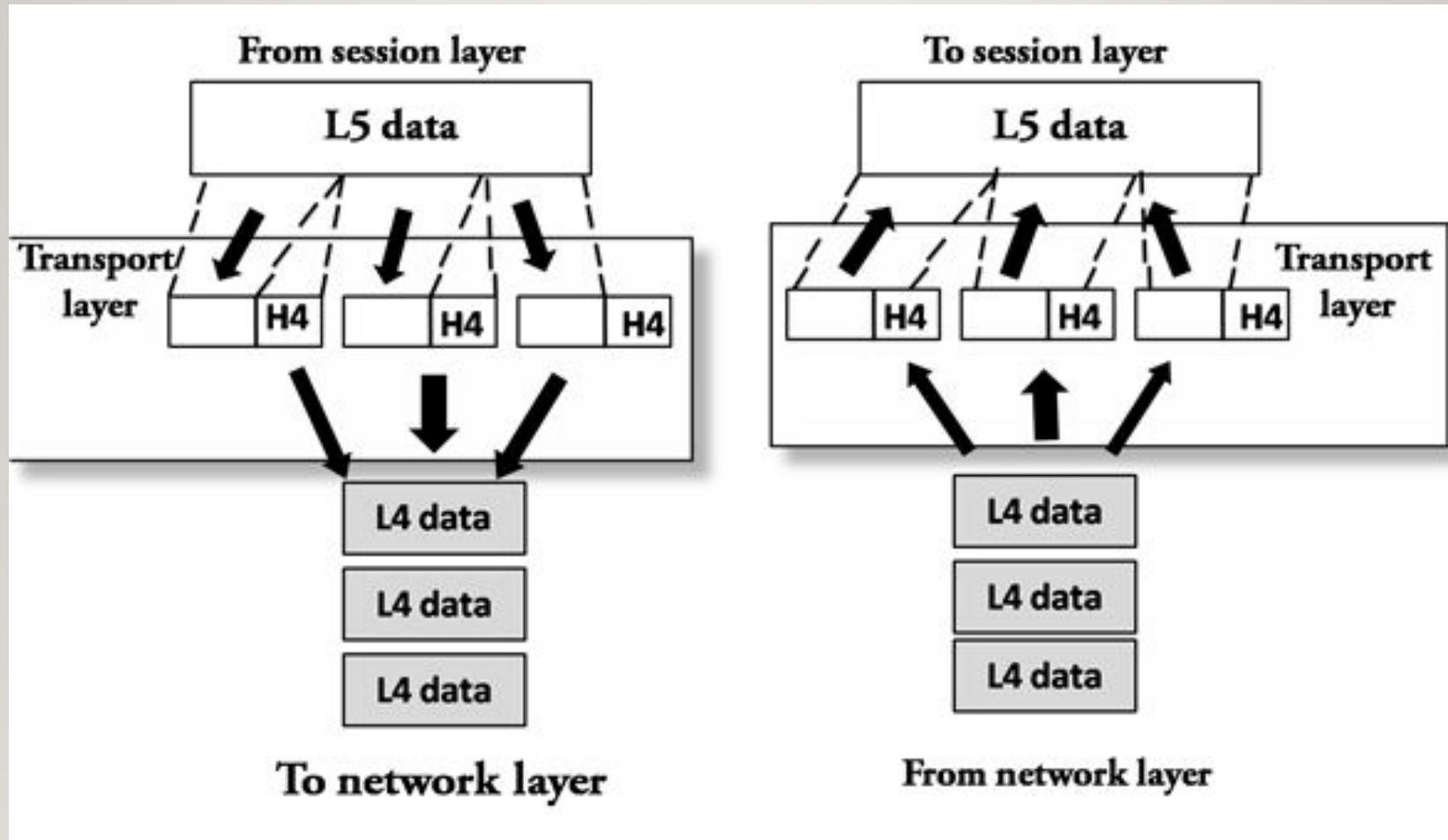
- Segmentation and Reassembly:
  - Accepts the message from the upper layer and breaks the message into smaller units.
  - Each segment has a header.
- Service point addressing:
  - Includes service point address or port address.
- Flow and error control

# Services provided by transport layer

---

- Connection oriented service
  - Three phase process: connection establishment, data transfer, disconnection.
  - Reliable and secure.
  - Acknowledgement by the receiver.
- Connectionless service
  - One phase process: data transfer.
  - No acknowledgement receipt of a packet.

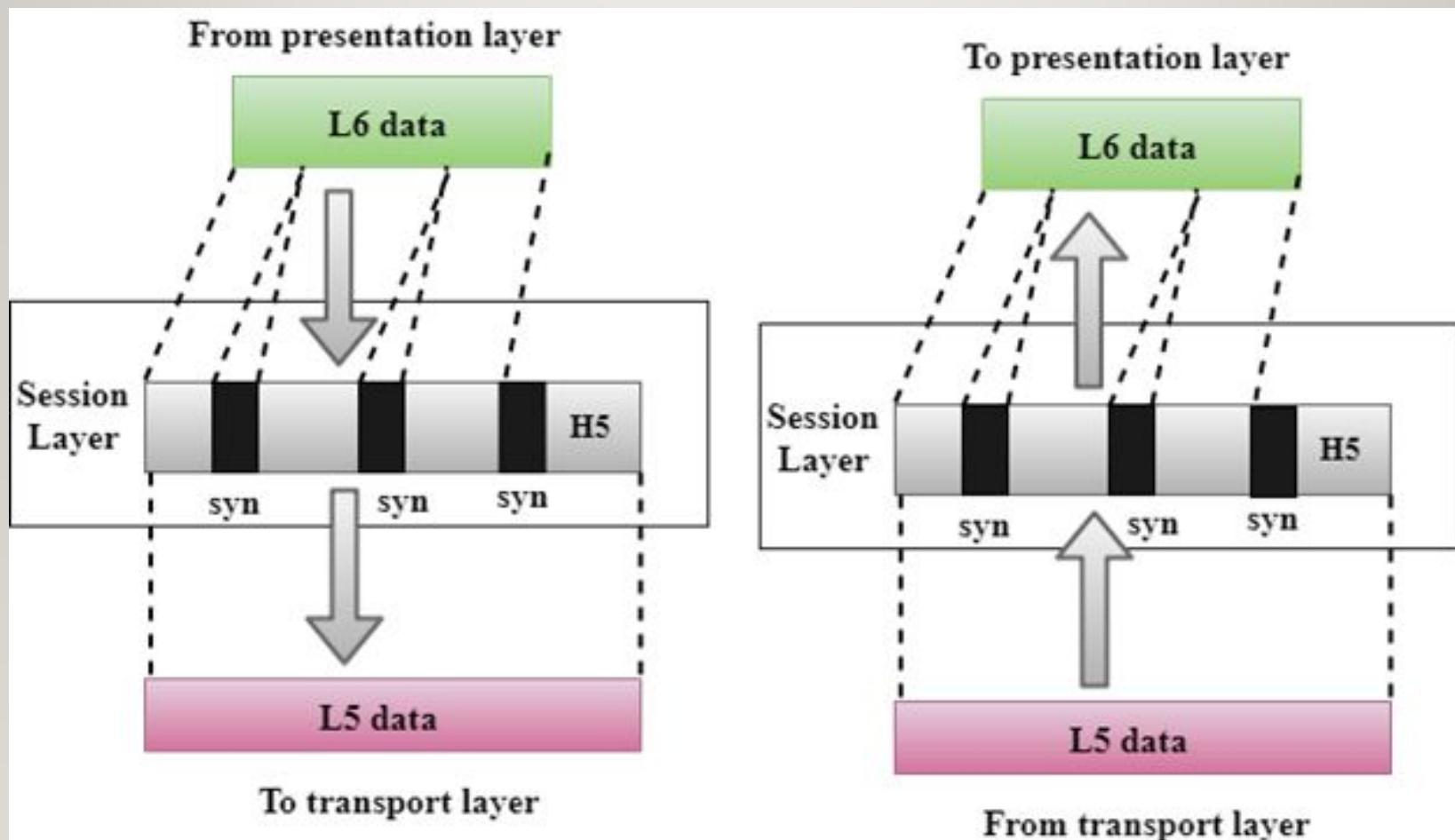
# Layer 4: Transport Layer



# Layer 5: Session Layer

- Allows two applications on different computers to establish, use, and end a session.
- Establishes dialog control
  - Keep track of whose turn it is to transmit.
- Performs synchronization
  - Add check points for long transmissions to identify the error and subsequent recovery.
  - Retransmission starts from the check point.

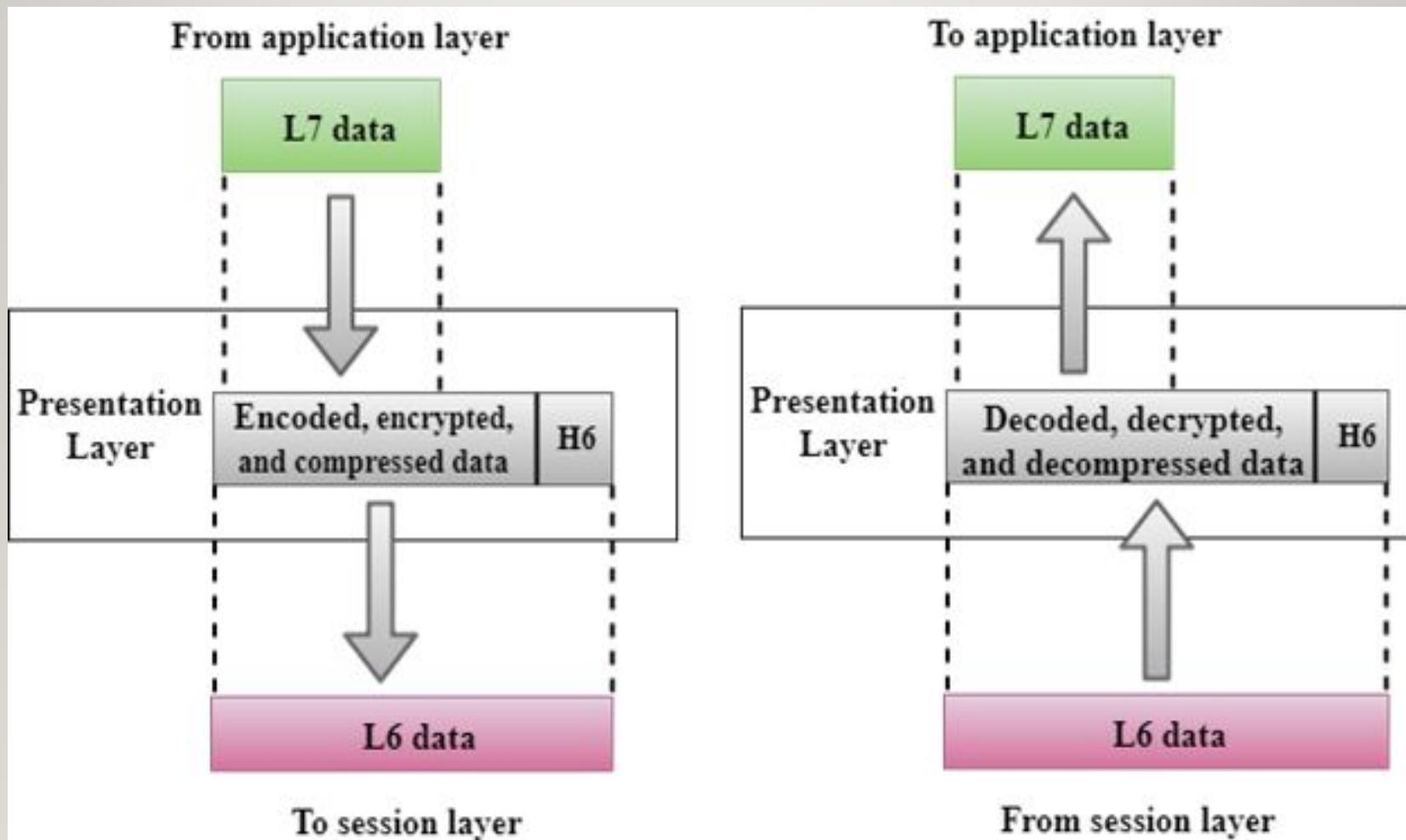
# Layer 5: Session Layer



# Layer 6: Presentation Layer

- Also known as translation layer.
- Translation
  - Translates different data representations from the Application layer into uniform standard format.
- Providing services for secure efficient data transmission
  - Encryption to maintain privacy.
- Compression
  - Reduce the number of bits to be transmitted.

# Layer 6: Presentation Layer



# Layer 7: Application Layer

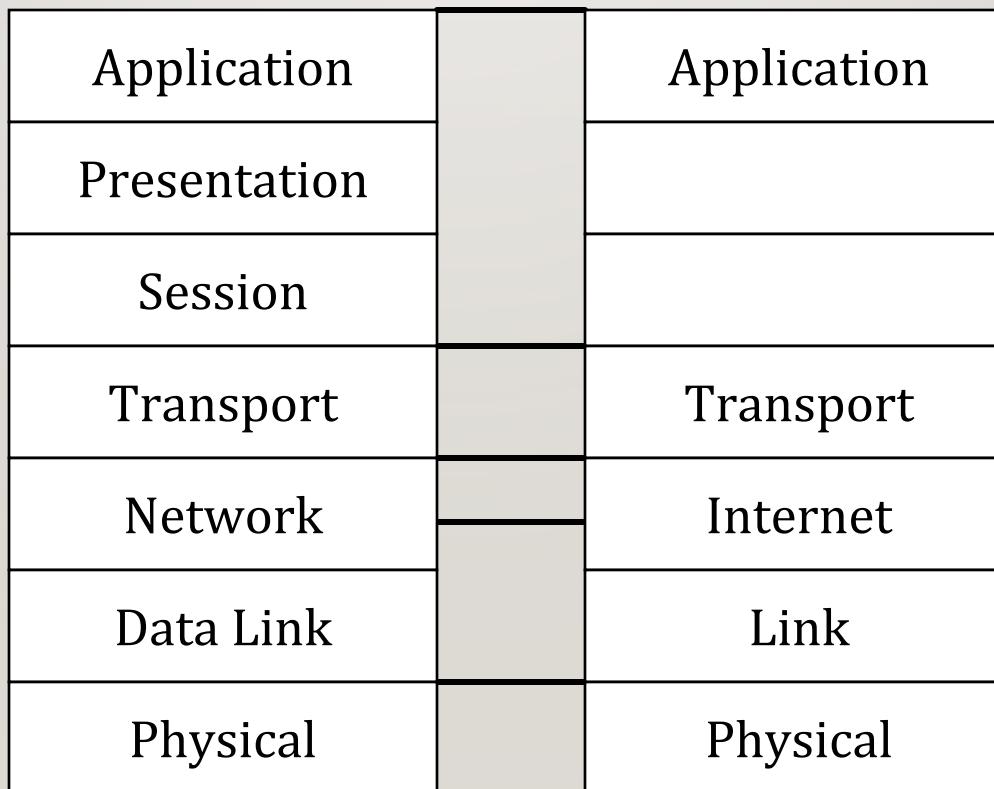
---

- Window for users and application processes to access the network service.
- File transfer, access and management:
  - Allows user to access files in a remote computer.
- Mail service
  - Facility for email forwarding and storage.

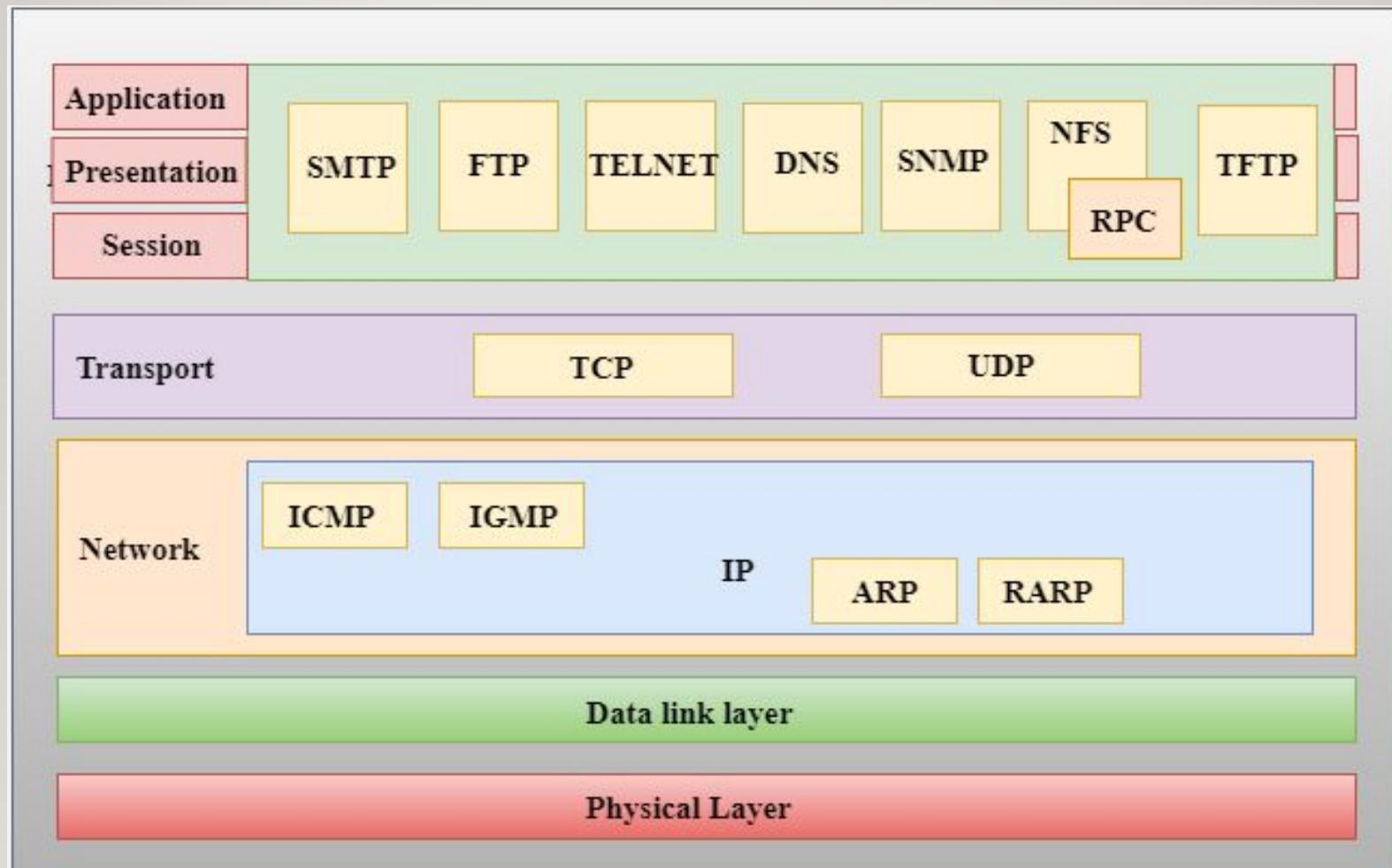
---

# TCP/IP Reference Model

# TCP/ IP Reference Model



# TCP/ IP Reference Model



# Comparison between OSI and TCP / IP Reference Model

---

OSI	TCP / IP
OSI represents <b>Open System Interconnection.</b>	TCP/IP model represents the Transmission Control Protocol / Internet Protocol.
OSI is a generic, protocol independent standard.	TCP/IP model depends on standard protocols about which the computer network has created
The OSI model was developed first, and then protocols were created to fit the network architecture's needs.	The protocols were created first and then built the TCP/IP model.

# Comparison between OSI and TCP / IP Reference Model

---

OSI	TCP / IP
It provides both connection and connectionless oriented transmission in the network layer; however, only connection-oriented transmission in the transport layer.	It provides connectionless transmission in the network layer and supports connecting and connectionless-oriented transmission in the transport layer.
The smallest size of the OSI header is 5 bytes.	The smallest size of the TCP/IP header is 20 bytes.
7 layered architecture	5- layered architecture

# **COMPUTER NETWORKS**

## **MODULE I**

---

Jincy J Fernandez

Asst. Professor- CSE

RSET

# Topics

---

- Physical Layer
  - Modes of Communication
  - Physical Topologies
  - Signal Encoding
  - Repeaters and Hub
  - Transmission media
  - Performance indicators

# Physical Layer-Data and Signals

- Data:

---
- Analog: continuous information
- Digital: discrete information
- Signal:
  - Analog: infinite levels of intensity over a period of time.
  - Digital: limited number of defined values.
  - Periodic: completes a pattern within a measurable time frame (period) and it repeats.
  - Non periodic: changes without a pattern.

# Physical Layer-Data and Signals

---

- Cycle:
  - Completion of one full pattern.
- Period:
  - Amount of time, a signal needs to complete one cycle.
- Frequency:
  - The number of periods in 1 second.
  - $f = \frac{1}{T}$
  - Expressed in Hertz.

# Physical Layer-Data and Signals

---

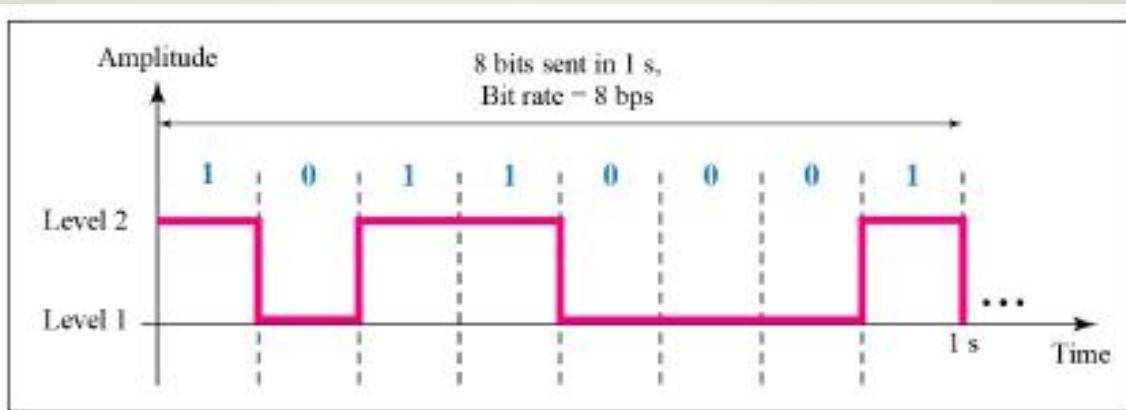
- Q1. The power at home has a frequency of 60Hz.  
Determine the period of the wave.
- Q2. The period of a signal is 100ms. What is its frequency in KHz?

# Physical Layer-Sampling & Quantization

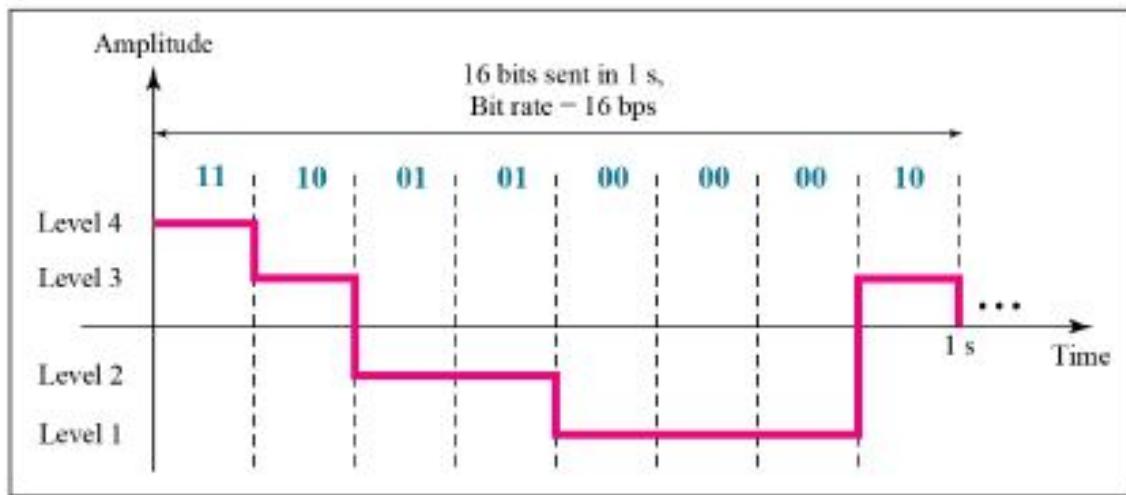
---

- For converting analog signal to digital.
- **Sampling:**
  - Process of recording an analog signal at regular discrete moments of time.
- **Quantization**
  - the process of mapping continuous amplitude (analog) signal into discrete amplitude (digital) signal

# Physical Layer-Digital Signals



a. A digital signal with two levels



b. A digital signal with four levels

# Physical Layer-Digital Signals

---

- Cycle:
  - Completion of one full pattern.
- Period:
  - Amount of time, a signal needs to complete one cycle.
- Frequency:
  - The number of periods in 1 second.
  - $f = \frac{1}{T}$
  - Expressed in Hertz.

# Physical Layer-Digital Signals

- **Bit Rate:**

---

- The number of bits sent in 1 sec.
- Expressed as bits per second (bps).
- Q1. What is the required bit rate of the channel to download text documents at the rate 100 pages per sec?
  - Assume a page is an average of 25 lines with 80 characters in each line. Assume 8 bits per character.
  - Ans:  $1600000 \text{ bps} = 1.6 \text{ Mbps}$

# Physical Layer-Digital Signals

- 
- Q2. A digitized voice channel is made by digitizing a 4-kHz bandwidth analog voice signal. Sample the signal at twice the highest frequency (two samples per Hertz). Assume that each sample requires 8 bits. What is the required bit rate in Kbps?
    - Ans:  $64000\text{bps} = 64\text{Kbps}$

# Physical Layer-Digital Signals

---

- Q2. A video signal transmission system transmits 625 frames per second. Each frame consists of 200x200 pixel grid with 64 intensity levels per pixel. Find the data rate in bps?
  - Ans: 150Mbps
  - n<sup>o</sup> no. of pixels ( $200 \times 200$ )
  - No. of bits for intensity levels, ( $\log_2$  levels= 6)
  - Rate of transmission of frames (625frames).

# Physical Layer-Digital Signals

---

- Cycle:
  - Completion of one full pattern.
- Period:
  - Amount of time, a signal needs to complete one cycle.
- Frequency:
  - The number of periods in 1 second.
  - $f = \frac{1}{T}$
  - Expressed in Hertz.

# Physical Layer-Digital Signals

---

- Cycle:
  - Completion of one full pattern.
- Period:
  - Amount of time, a signal needs to complete one cycle.
- Frequency:
  - The number of periods in 1 second.
  - $f = \frac{1}{T}$
  - Expressed in Hertz.

# Physical Layer-Digital Signals

---

- Q3. What is the bit rate of
  - A) a signal in which 1 bit lasts 0.001 s ?
    - Ans:  $1000 \text{ bps} = 1 \text{ Kbps}$
  - B) a signal in which 1 bit lasts 2ms?
    - Ans: 500bps
  - C) a signal in which 10 bits lasts  $20\mu\text{s}$ ?
    - Ans: 500 Kbps

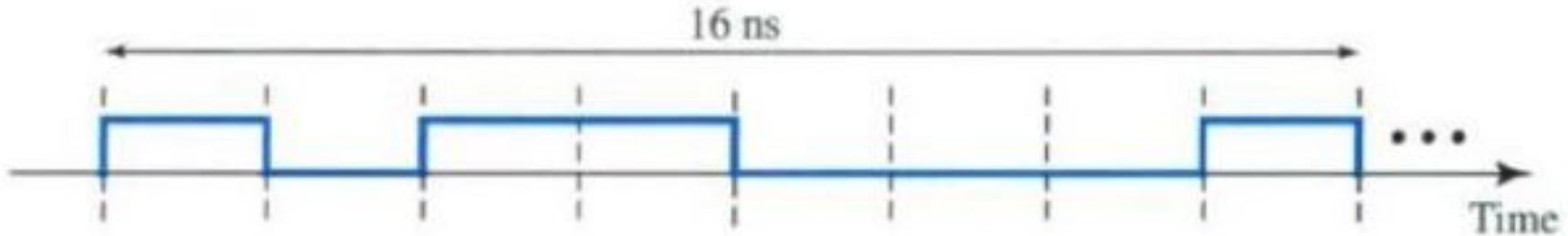
# Physical Layer-Digital Signals

---

- Q4. A device is sending out data at the rate of 1000 bps.
  - A) How long does it take to send out 10 bits?
    - Ans: 0.01s
  - B) How long does it take to send out a single character (8 bits)?
    - Ans: 8ms

# Physical Layer-Digital Signals

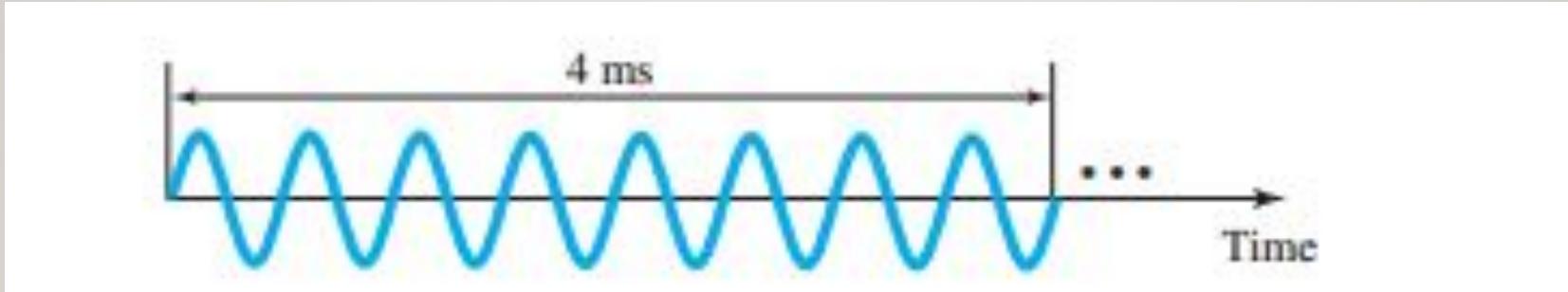
- Q5. What is the bit rate for the signal?



- Ans: 500 Mbps

# Physical Layer-Digital Signals

- Q5. What is the frequency of the signal?



- Ans: 2 KHz

# Physical Layer-Digital Signals

- 
- Q5. A file contains 2 million bytes. How long does it take to download this file using a 56-Kbps channel?
  - Ans: 286 sec

# Transmission Modes

---

- **Simplex:**
  - Unidirectional data transfer.
  - Use the entire capacity of the channel to send data in one direction.
  - E.g. Keyboards, traditional monitors
- **Half-Duplex:**
  - Bidirectional data transfer; not simultaneously.
  - E.g. Walkie-talkie
  - Channel capacity = bandwidth \* propagation delay

# Transmission Modes

---

- **Duplex:**

- Bidirectional data transfer; simultaneous transmission possible.
- E.g. Telephone networks
- Channel capacity=  $2 * \text{bandwidth} * \text{propagation delay}$

# Physical Topologies

---

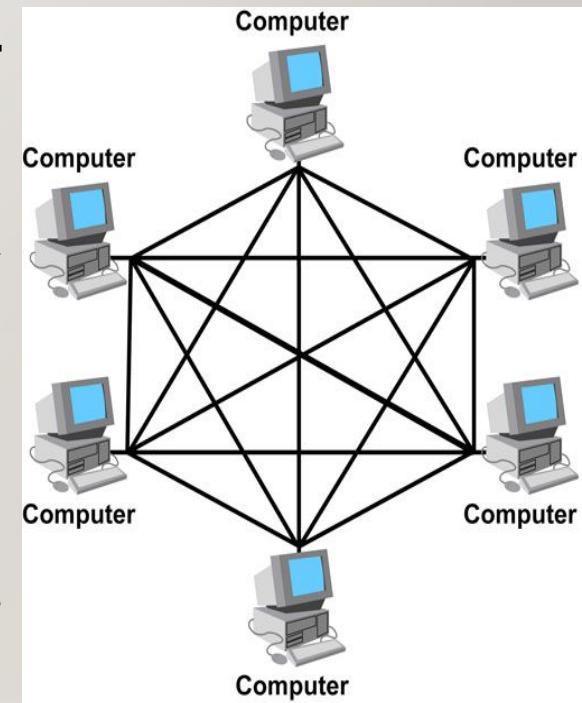
- The way in which a network is laid out physically.
- 2 or more links form a topology.
- Define the layout, virtual shape or structure of network.
- Four types.

# Physical Topologies-Types

---

## I. Mesh

- Dedicated point to point link to every other device.
- No. of physical links in a fully connected mesh =  $n*(n-1)$ .
- No. of physical links (if duplex) =  $n*(n-1)/2$ .
- E.g. connection of a telephone regional offices in which every regional office needs to be connected to every other regional offices.



# Physical Topologies-Types

---

## Mesh Topology- Advantages

- Use of dedicated links eliminates traffic problems.
- Robust
- Advantage of privacy or security
- The network can be expanded without disruption to current uses.
- Point to point links make fault identification and fault isolation easy.

# Physical Topologies-Types

---

## Mesh Topology – Disadvantages

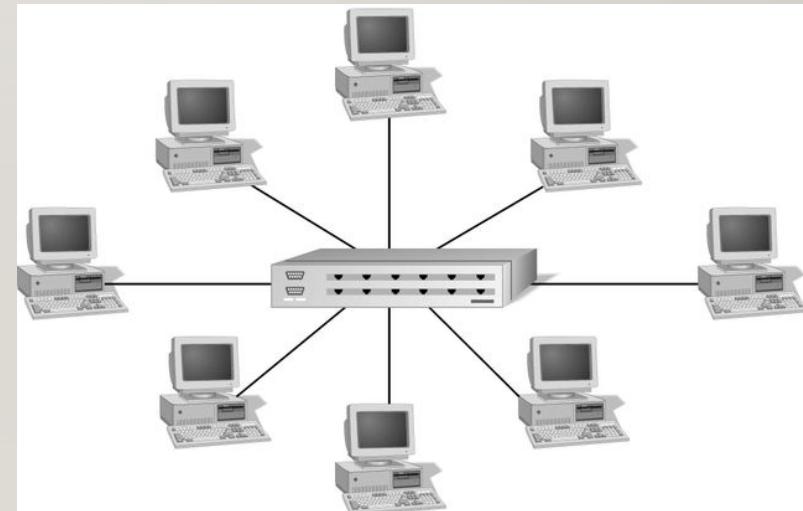
- Requires more cable than the other LAN topologies
- Complicated implementation
- Expensive Hardware.

# Physical Topologies-Types

---

## 2. Star

- Each device has a dedicated point-to-point link only to a central controller, called a hub.
- The devices are not directly linked to one another.
- No direct traffic between devices.
- The controller acts as an exchange.



# Physical Topologies-Types

---

## Star Topology - Advantages

- Less expensive than a mesh topology.
- Robust.
- Easy fault identification and fault isolation.
- Easy to install and reconfigure.

## Star Topology - Disadvantages

- Dependency of the whole topology on the central hub.
- More cabling is required than ring and bus topology.

# Physical Topologies-Types

---

## 3. Bus Topology

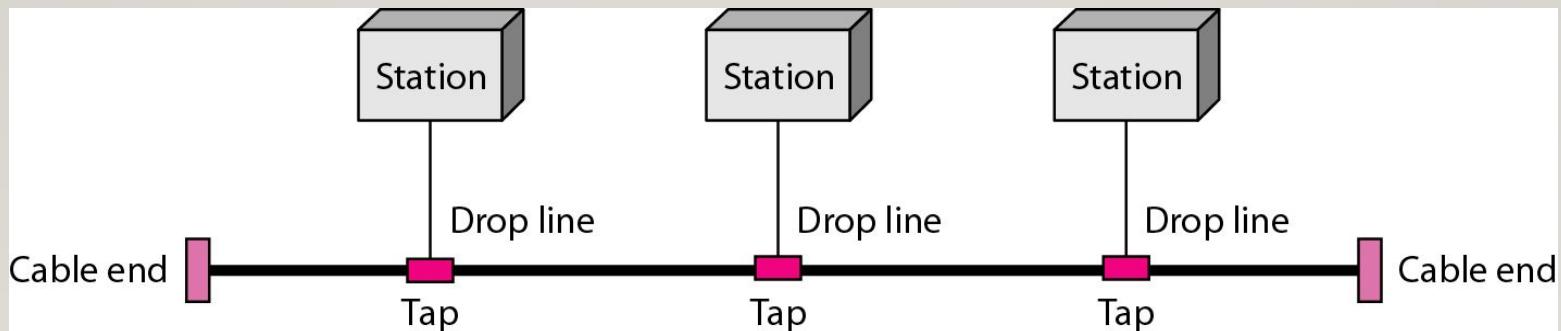
- Multipoint connection.
- One long cable acts as a backbone to link all the devices in a network.
- Nodes are connected to the bus cable by drop lines and taps.
- A drop line is a connection running between the device and the main cable.

# Physical Topologies-Types

---

## 3. Bus Topology

- 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.



# Physical Topologies-Types

---

## Bus Topology- Advantages

- Works well for small networks.
- Ease of installation.
- Requires less cabling than mesh or star topologies.

## Bus Topology – Disadvantages

- Fault or break in the bus cable stops all transmission.
- Difficult to add new devices.

# Physical Topologies-Types

---

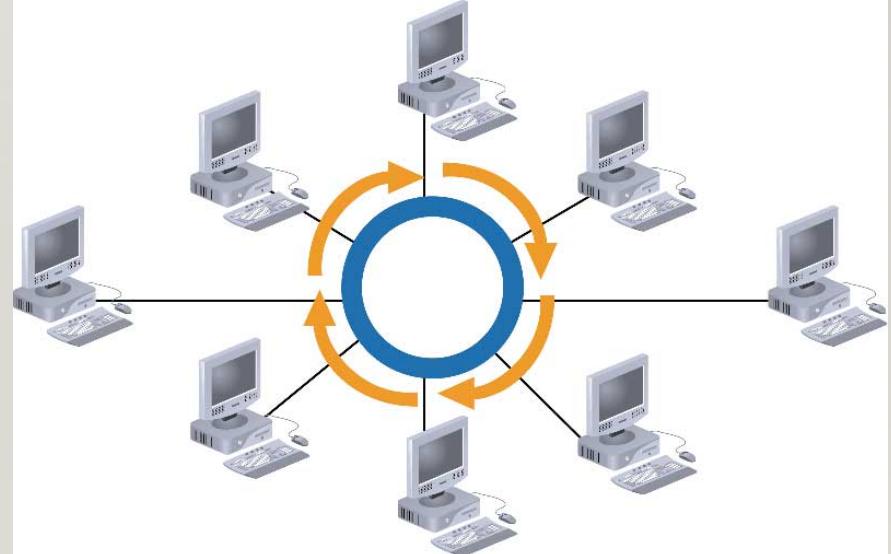
## 4. Ring Topology

- 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.

# Physical Topologies-Types

## 4. Ring Topology

- When a device receives a signal intended for another device, its repeater regenerates the bits and passes them along.



# Physical Topologies-Types

---

## Ring Topology- Advantages

- Cable faults are easily located, making troubleshooting easier.
- Ring networks are moderately easy to install

## Ring Topology – Disadvantages

- Unidirectional traffic. A single break in the cable can disrupt the entire network.
- Expansion to the network can cause network disruption.

# Physical Topologies-Types

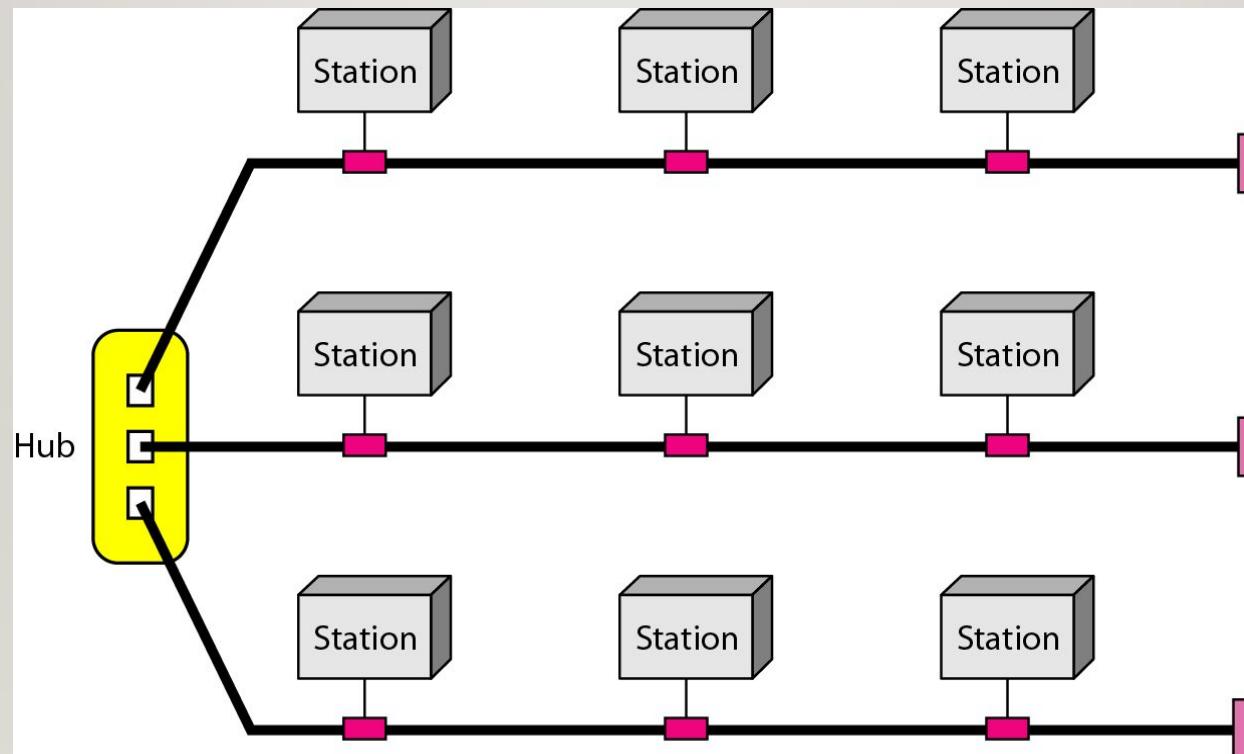
---

## Hybrid Topology

- One example of Hybrid Topology is Tree topology
- Tree topology is a combination of Bus and Star topology.
- It consists of groups of star-configured workstations connected to a linear bus backbone cable.
- If the backbone line breaks, the entire segment goes down
- An example of this network could be cable TV technology

# Physical Topologies-Types

## Hybrid Topology



# COMPUTER NETWORKS

## MODULE I

---

Jincy J Fernandez

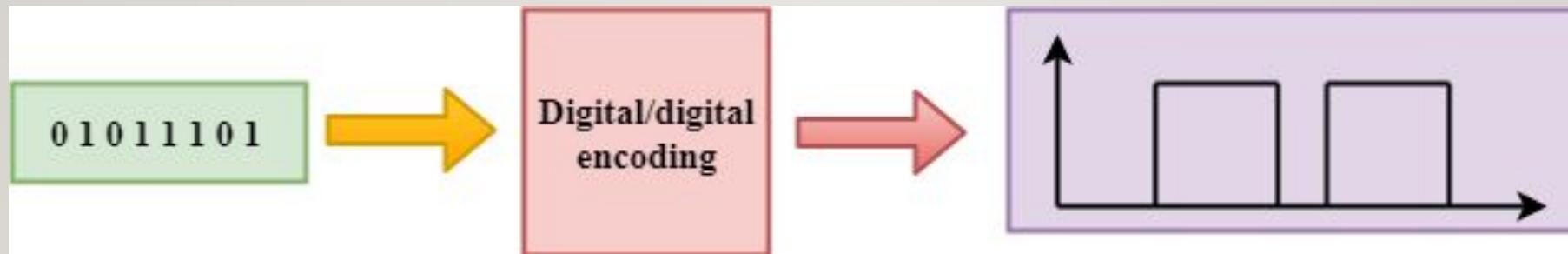
Asst. Professor- CSE

RSET

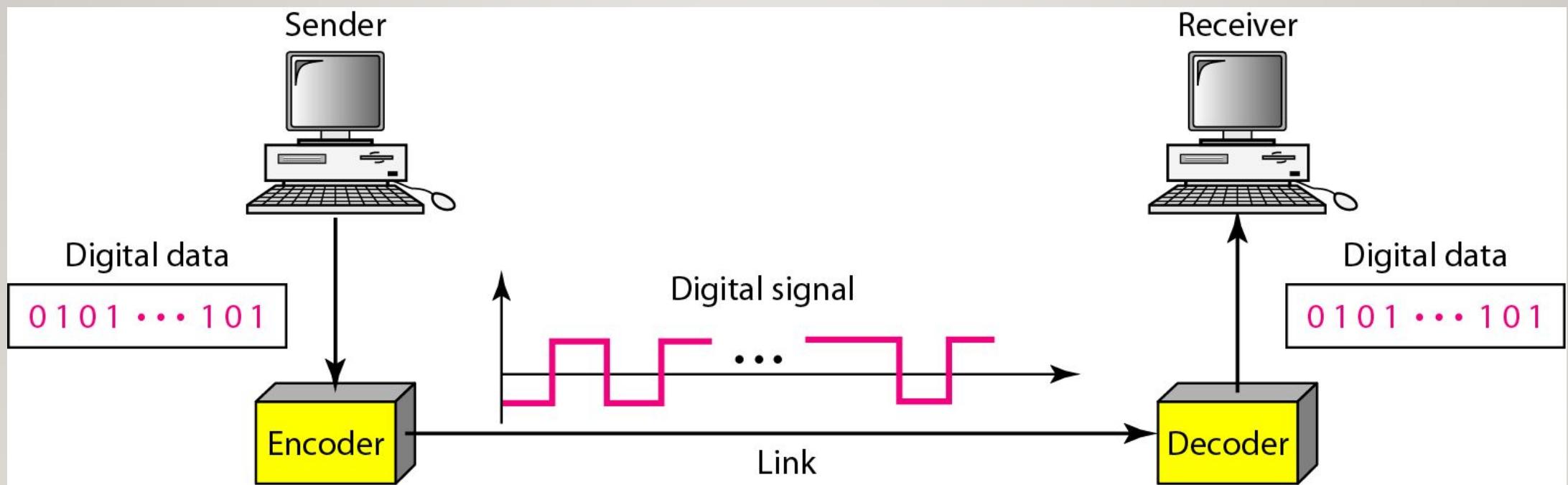
# Digital to Digital Conversion

---

- Digital-to-digital encoding is the representation of digital information by a digital signal



# Digital to Digital Conversion



# Data symbols to signals

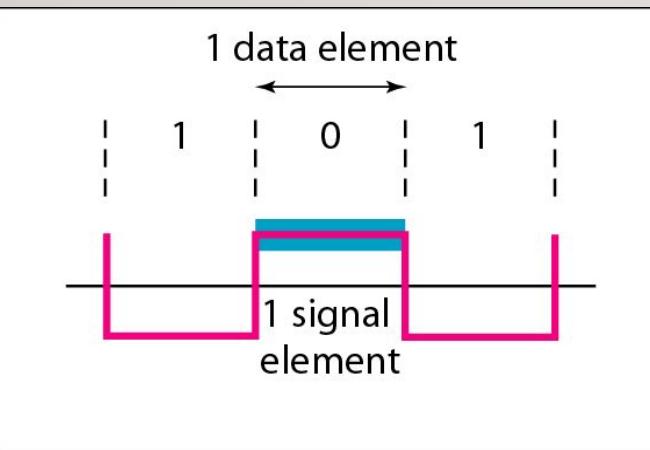
---

- A data symbol (or element) can consist of a number of data bits:
  - 1, 0 or
  - 11, 10, 01, .....
- A data symbol can be coded into a single signal element or multiple signal elements
  - 1 -> +V, 0 -> -V
  - 1 -> +V and -V, 0 -> -V and +V
- The ratio 'r' is the number of data elements carried by a signal element.

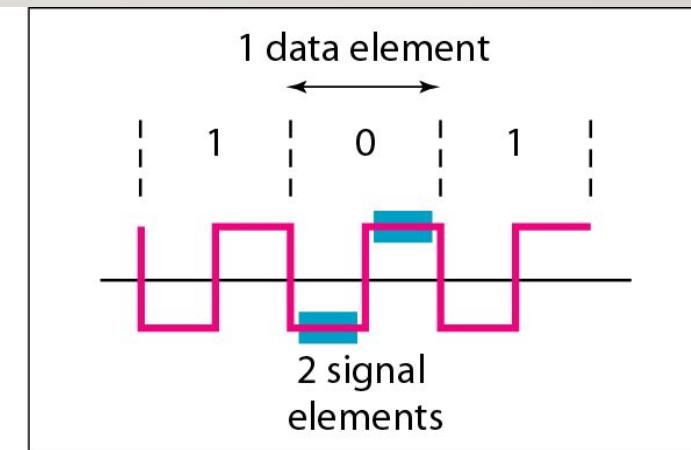
# Data Rate, Signal RateZ

---

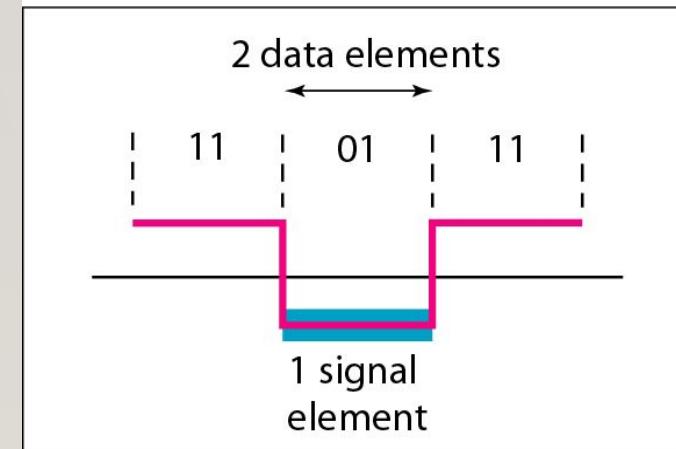
- The **data rate** defines the number of bits sent per sec - bps.
- It is often referred to the bit rate.
- The **signal rate** is the number of signal elements sent in a second and is measured in bauds.
- It is also referred to as the modulation rate (baud rate).
- Goal is to **increase the data rate whilst reducing the baud rate**.



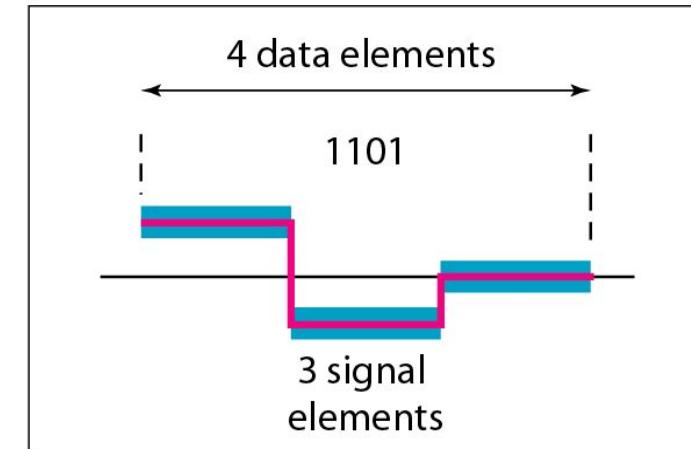
a. One data element per one signal element ( $r = 1$ )



b. One data element per two signal elements ( $r = \frac{1}{2}$ )



c. Two data elements per one signal element ( $r = 2$ )



d. Four data elements per three signal elements ( $r = \frac{4}{3}$ )

# Baud Rate

---

- The baud or signal rate can be expressed as:

$$S = c \times N \times 1/r \text{ bauds}$$

where N is data rate

c is the case factor (worst, best & avg)

r is the ratio between data element & signal element

# Questions

---

- A signal is carrying data in which one data element is encoded as one signal element ( $r = 1$ ). If the bit rate is 100 Kbps, what is the average value of the baud rate if  $c$  is between 0 and 1?
  - Assume  $c=1/2$ .

$$S = c \times N \times \frac{1}{r} = \frac{1}{2} \times 100,000 \times \frac{1}{1} = 50,000 = 50 \text{ baud}$$

# Signal Encoding Schemes

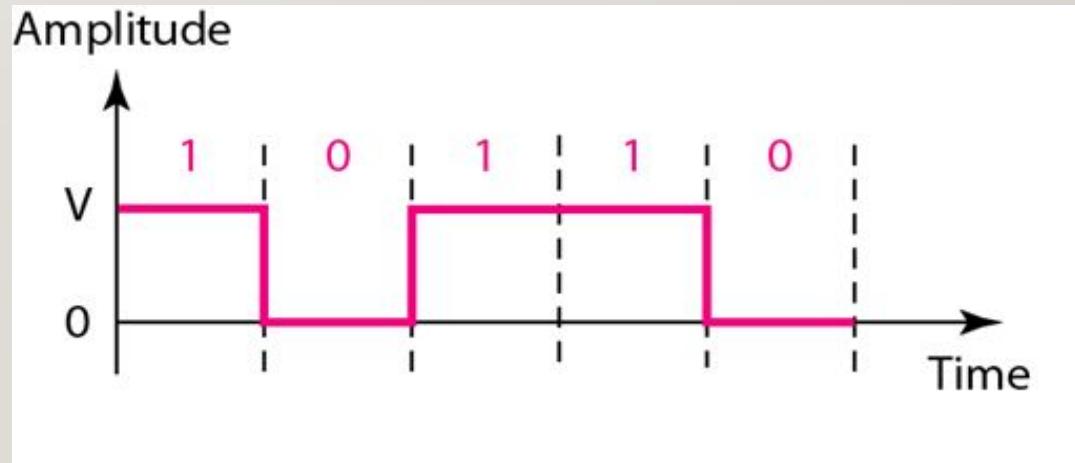
---

- Unipolar
  - '1' is considered as a high voltage and '0' is considered as a zero voltage.
- Polar
  - uses two voltage levels: one is positive, and another is negative.
- Bipolar
  - represents three voltage levels: positive, negative, and zero.
  - zero level represents binary 0, and binary 1 is represented by alternating positive and negative voltages.

# Unipolar- NRZ

---

- Non-Return-to-Zero (NRZ).
- 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.



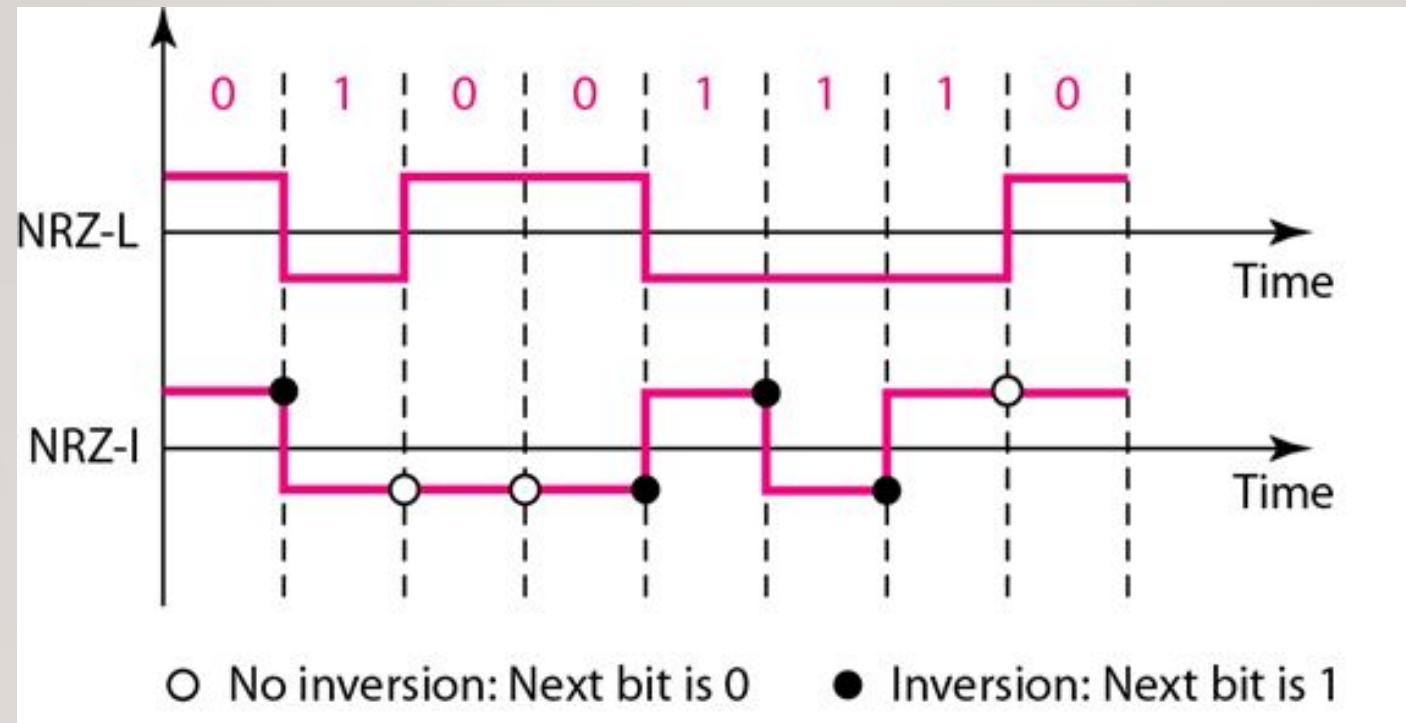
# Polar- NRZ

---

- The voltages are on both sides of the time axis.
- Polar NRZ scheme can be implemented with two voltages.
- E. g. +V for 0 and -V for 1.
- There are two versions:
  - NRZ - Level (**NRZ-L**) - positive voltage for one symbol and negative for the other.
  - NRZ - Inversion (**NRZ-I**) - the change or lack of change in polarity determines the value of a symbol. E. g. a “1” symbol inverts the polarity a “0” does not.

# Polar- NRZ

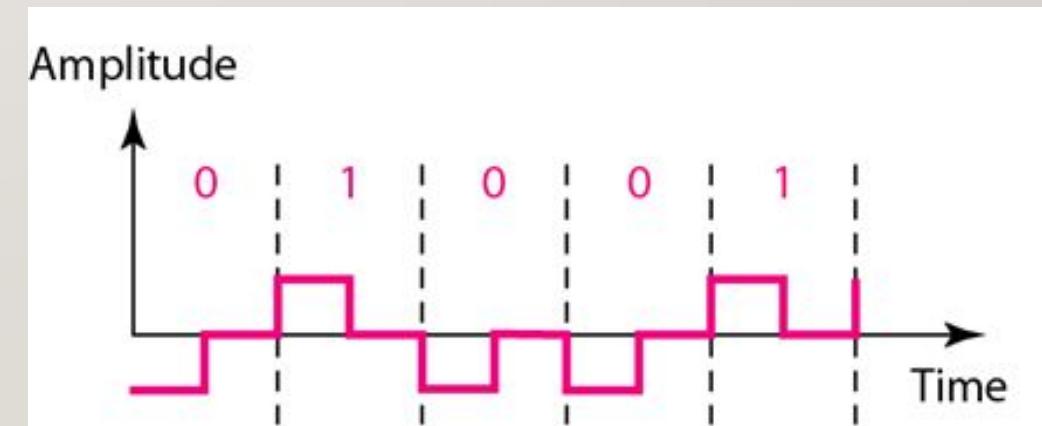
---



# Polar- RZ

---

- The Return to Zero (RZ) scheme uses three voltage values. +, 0, -.
- Each symbol has a transition in the middle. Either from high to zero (bit 1) or from low to zero (bit 0).
- This scheme has more signal transitions (two per symbol) and therefore requires a wider bandwidth.

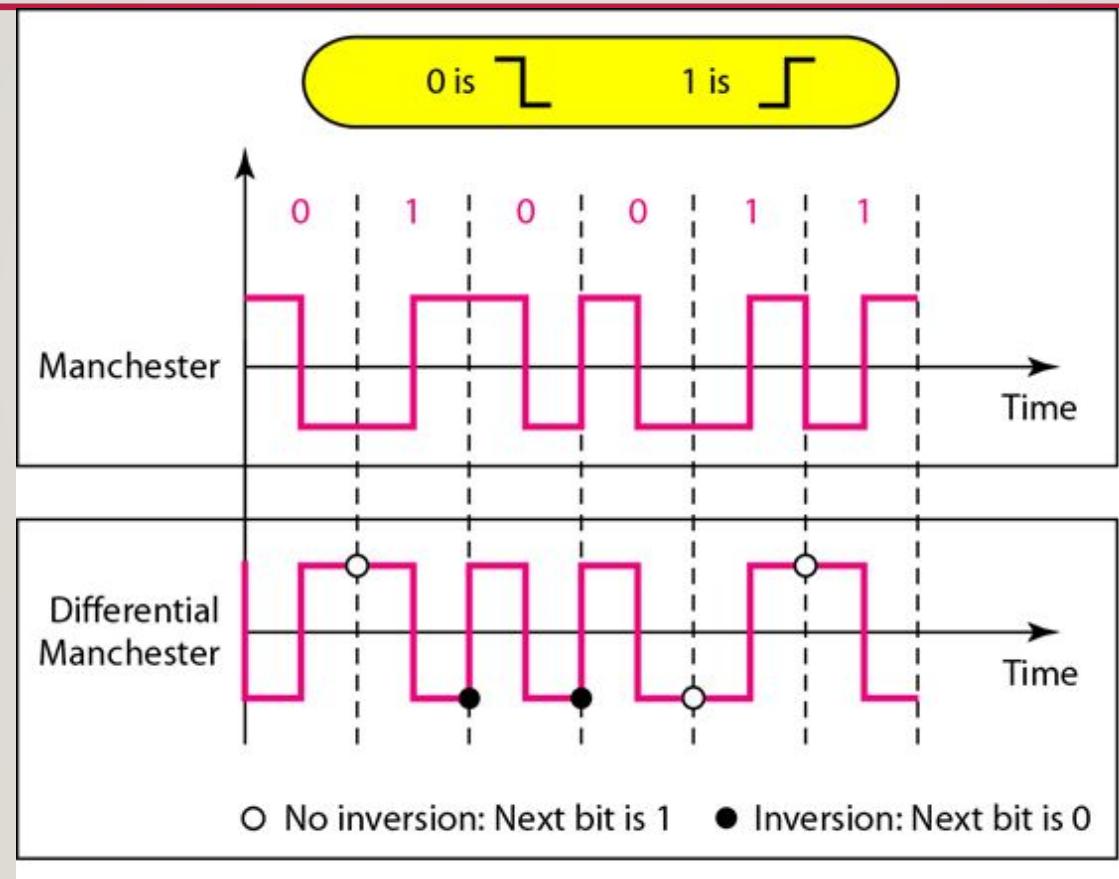


# Polar- Biphase

---

- **Manchester Encoding**
  - consists of combining the NRZ-L and RZ schemes.
  - Every symbol has a level transition in the middle: from high to low or low to high.
  - Uses only two voltage levels.
- **Differential Manchester Encoding**
  - consists of combining the NRZ-I and RZ schemes.
  - Always a transition at the middle of the bit.
  - Bit values are determined at the beginning of the bit.

# Polar- Biphase



# Bipolar

---

- **Alternate Mark Inversion (AMI)**

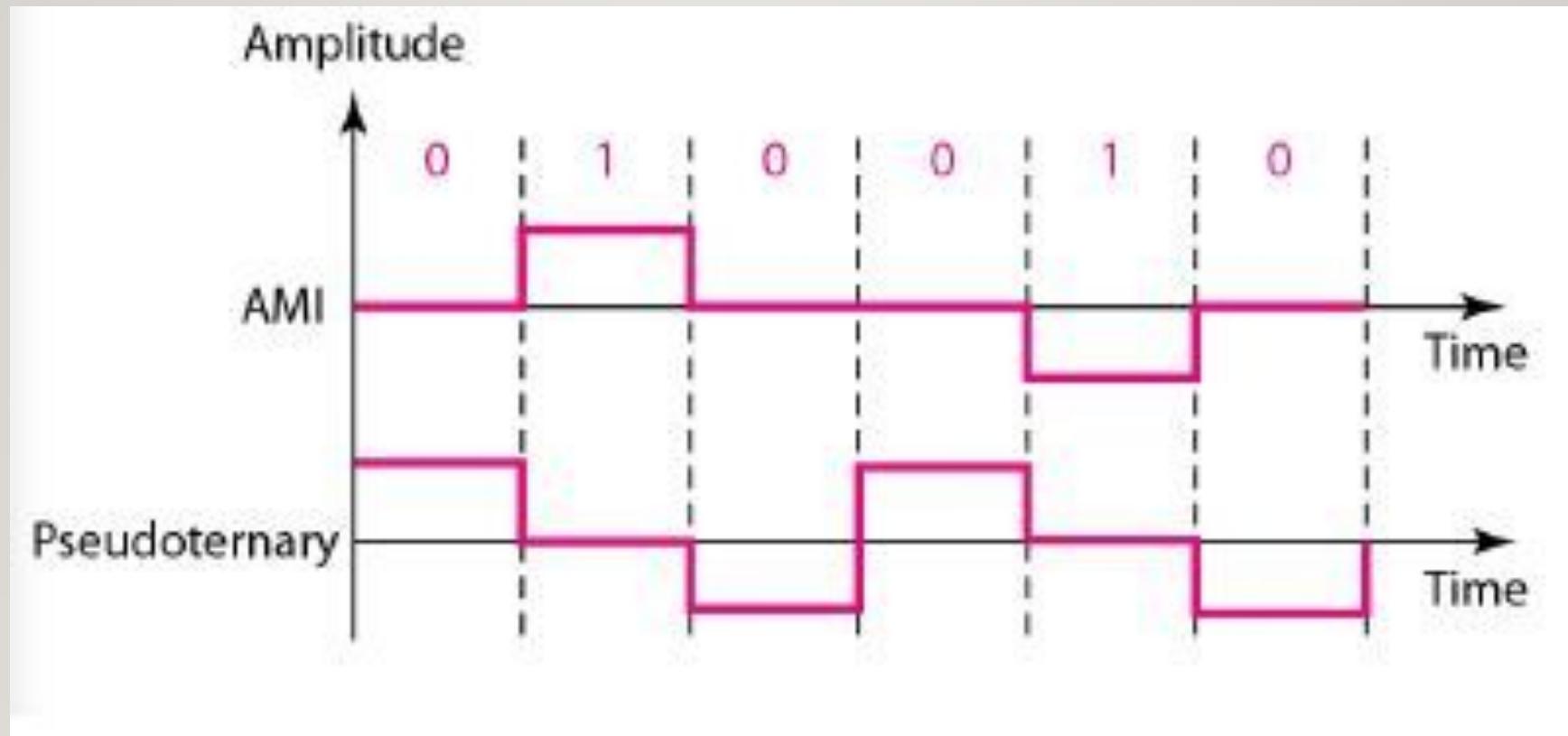
- Mark means '1'.
- Zero voltage represents bit 0.
- Binary 1's are represented by alternating positive and negative voltages.

- **Pseudoternary**

- Zero voltage represents bit 1.
- Binary 0's are represented by alternating positive and negative voltages.

# Bipolar

---



# Questions

---

Q1. Sketch the waveform in Manchester and Differential Manchester encoding for the bitstream 11000110010.

Q2. Sketch the waveform in Manchester and Differential Manchester encoding for the bitstream 10100111001.

# Performance Indicators

---

- How good is the performance of the network?
- Measure of service quality of a network.
- Some general performance indicators are:
  - Bandwidth
  - Throughput
  - Latency or delay
  - Bandwidth-delay product
  - Jitter

# Bandwidth

---

- Measure of data that can be transmitted in a fixed measure of time.
- For digital devices: bits per second (bps): no. of bits a network can transmit in a second.
- E. g: bandwidth of a Fast Ethernet network is a maximum of 100 Mbps.
- For analog devices: cycles per second or Hertz (Hz).
- An increase in bandwidth in hertz means an increase in bandwidth in bits per second.

# Throughput

---

- Number of messages successfully transmitted per unit time.
- Measured in bits per second (bps).
- The bandwidth is a potential measurement of a link; the throughput is an actual measurement of how fast we can send data.
- A link may have a bandwidth of ' $B$ ' bps, but ' $T$ ' bps can be sent through this link with  $T < B$ .
- Throughput may be affected by
  - Hindrance of the underlying physical medium.
  - Available processing power of the system components.

# Throughput

---

Q1. A network with bandwidth of 10Mbps can pass only an average of 12000 frames per minute where each frame carries an average of 10000 bits. What will be the throughput for this network?

Ans: 2Mbps

# Latency (Delay)

---

- Total time taken for a complete message to arrive at the destination.
- It starts with the time when first bit is sent out from the sender and ends with the time when the last bit of the message is delivered.
- Low latency networks and High latency networks.
- Measured in milliseconds.

**Latency = propagation time + transmission time + queuing time +  
processing delay**

# Latency- Propagation Time

---

- Time required for a bit to travel from source to destination.
- $$\text{Propagation Time} = \frac{\text{Distance}}{\text{Propagation speed}}$$
- Q1. What is the propagation time when the distance between two points is 12000km? Assume the propagation speed to be  $2.4 \times 10^8$  m/s.

Ans: 50ms

# Latency- Transmission Time

---

- The time required for transmission of a message depends on the size of the message and the bandwidth of the channel.
  - $Transmission\ Time = \frac{Message\ Size}{Bandwidth}$
  - Q1. What will be the transmission time for a 2.5KB message when the bandwidth of the network is 1Gbps?

Ans: 0.0204ms

# Latency- Queuing Time

---

- The time needed for each intermediate or end device to hold the message before it can be processed.
- The packet will be in a queue.
- The more the traffic, the more likely a packet is stuck in the queue.

# Latency- Processing Delay

---

- Delay based on how long it takes the router to figure out where to send the packet.
- As soon as the router finds it out, it will queue the packet for transmission.

# Latency

---

- Q1. 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  $\mu$ s and a processing time of 1  $\mu$ s. The length of the link is 2000 Km. The speed of light inside the link is  $2 \times 10^8$  m/s. The link has a bandwidth of 5 Mbps.

Ans: 1.0100003 s

# Bandwidth Delay Product

---

- Signifies how many bits the sender can send before the first bit reaches the receiver.
- $\text{Bandwidth Delay Product} = \text{Total available bandwidth (bps)} * \text{Round trip time(sec)}$
- RTT is the sum of the time taken for a signal to be transmitted from the sender to the receiver and the time taken for its acknowledgement to reach the sender from the receiver.
- Unit is bits or bytes.

# Bandwidth Delay Product

---

Q1. Consider that link capacity of a channel is 512Kbps and round-trip time is 1000ms.

Ans: 64000 bytes

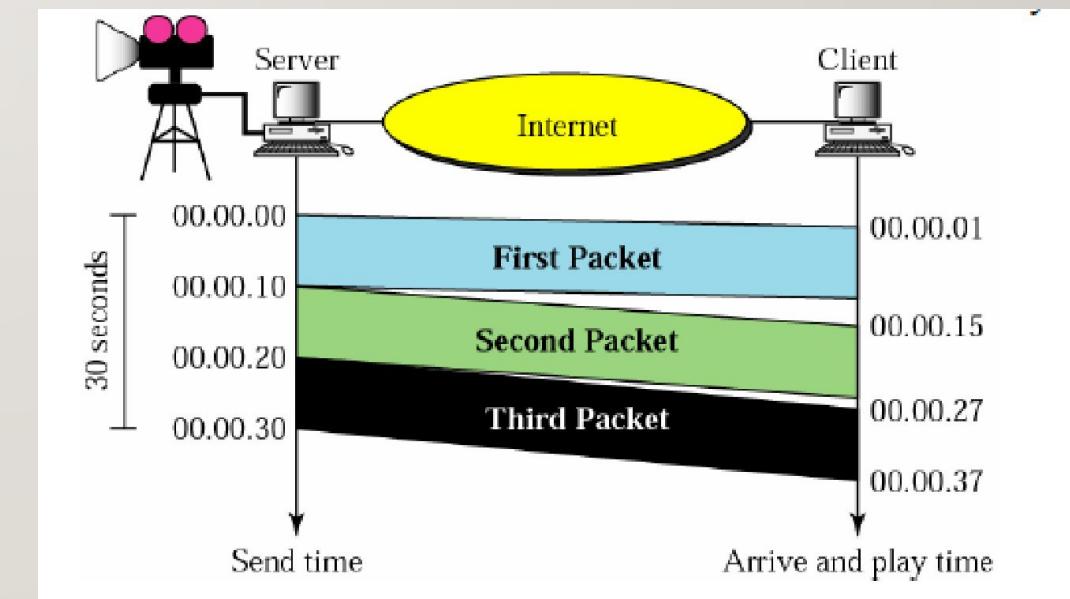
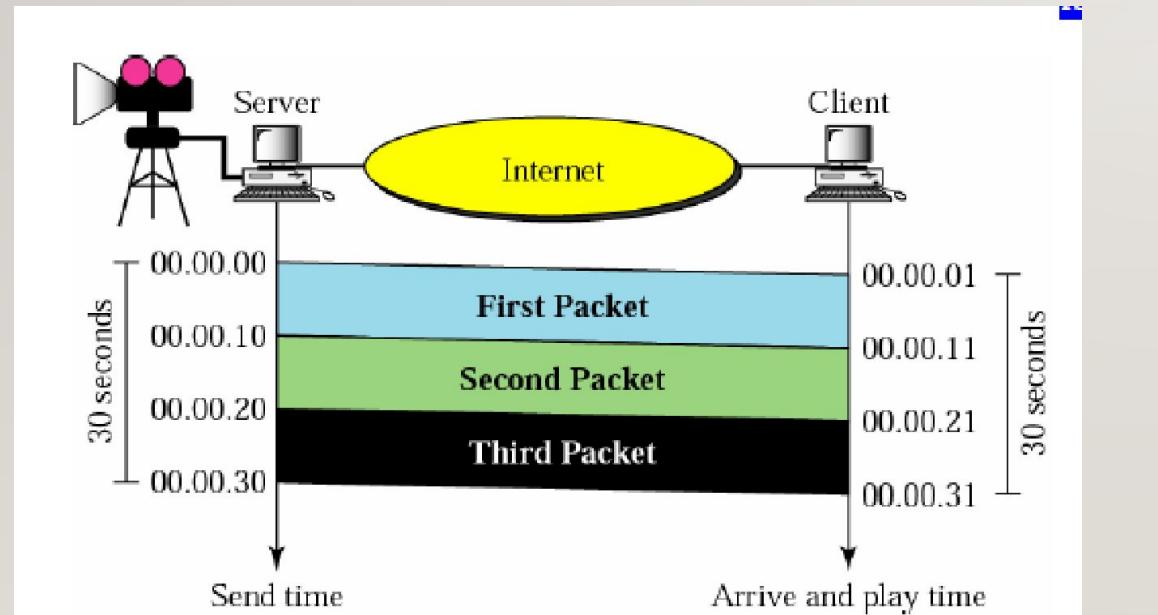
# Jitter

---

- Jitter is any deviation in the signal pulses in a high-frequency digital signal.
- Jitter can cause a display monitor to flicker, introduce undesired effects in audio signals, and lead to loss of transmitted data between network devices.
- degrade the quality of communications.
- It is a problem if different packets of data encounter different delays.

# Jitter

- Send a digitized and packetized video.
- Assume, three packets in total; each holds 10 seconds of video information, 1 sec delay for each packet to reach the destination.



# COMPUTER NETWORKS

## MODULE I

---

Jincy J Fernandez

Asst. Professor- CSE

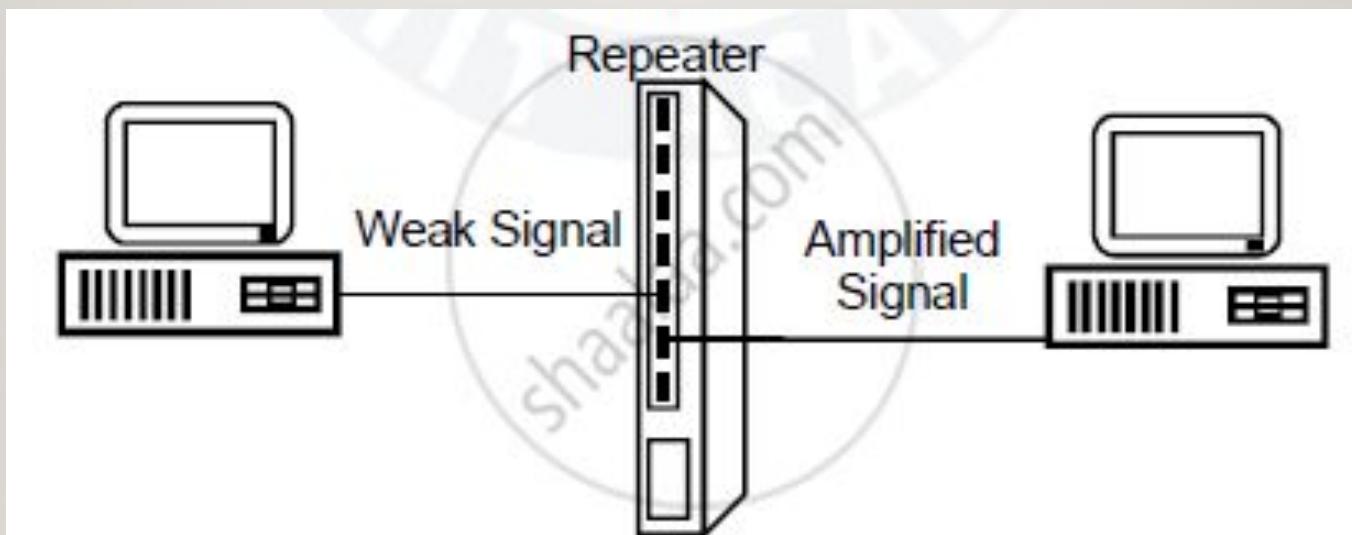
RSET

# Physical Layer Devices- Repeater\*

- 
- A repeater is a device that operates only in the physical layer.
  - Signals that carry information within a network can travel a fixed distance before attenuation endangers the integrity of the data.
  - A repeater receives a signal and, before it becomes too weak or corrupted, regenerates the original bit pattern.
  - The repeater then sends the refreshed signal.

# Physical Layer Devices- Repeater\*

---



# Physical Layer Devices- Hub

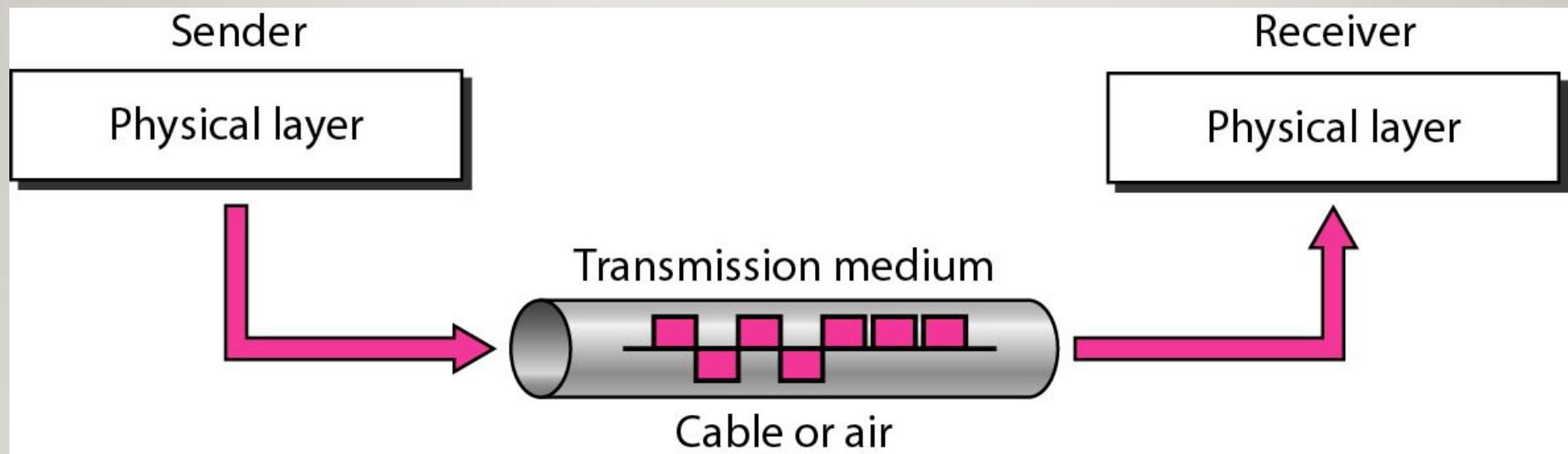
---

- A hub is just a connector.
- It connects the wires coming from different branches.
- In a star-topology, a passive hub is just a point where the signals coming from different stations collide; the hub is the collision point.



# Transmission Medium & Physical Layer

---



# Transmission Media

- 
- Communicating channel to carry information.
  - Copper based network? bits in the form of electromagnetic signals.
  - Fiber based network, bits in the light pulses.
  - The quality of the data transmission is determined by the characteristics of the medium and signal.

# Transmission Media- Factors to be considered

## 1. Bandwidth

---

- Bandwidth and data rate are directly proportional.

## 2. Transmission Impairment

- Received signal  $\neq$  transmitted signal.

1. Attenuation: loss of energy (signal strength decreases with increasing distance)

2. Distortion: change in the shape of the signal.

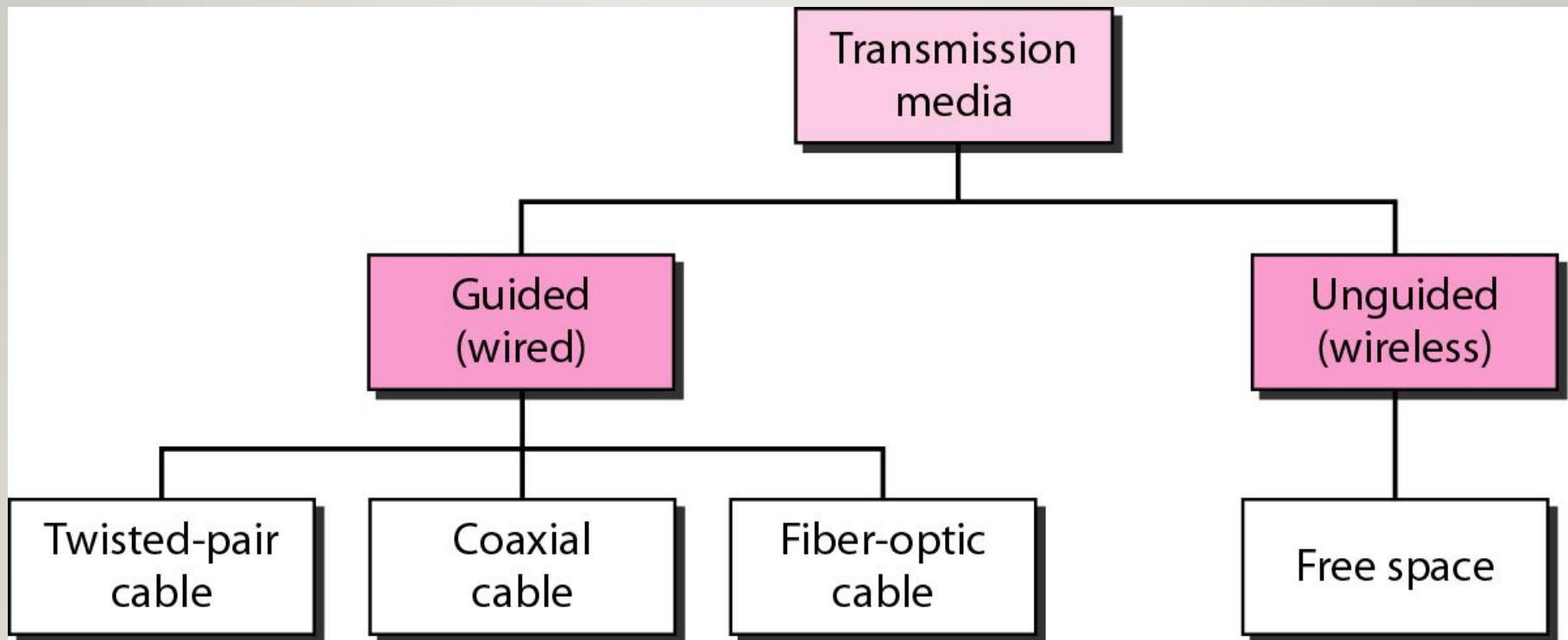
3. Noise: unwanted external signal gets added to the transmitted signal.

## 4. Interference

- Process of disrupting a signal by adding unwanted signals.

# Classes of Transmission Medium

---



# Guided Media

---

- Provide a channel from one device to another.
- Medium characteristics are important.
- Main features: high speed, covers small distances.
- Include twisted-pair cable, coaxial cable, and fiber-optic cable.

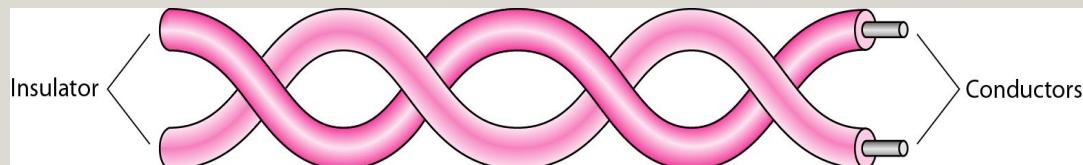
# Guided Media

- 
- 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

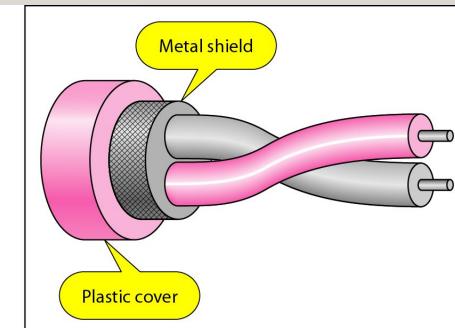
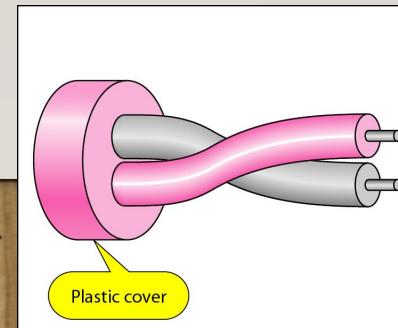
- A twisted pair consists of two conductors, each with its own plastic insulation, twisted together.
- One of the wires is used to carry signals to the receiver, and the other is used only as a ground reference.
- In addition to the signal sent by the sender on one of the wires, interference (noise) and crosstalk may affect both wires and create unwanted signals.



# Guided Media

## • Twisted Pair- UTP and STP

- The most common twisted-pair cable: unshielded twisted-pair (UTP).
- IBM has also produced a version of twisted-pair cable for its use called shielded twisted-pair (STP).
- STP cable has a metal foil or braided mesh covering that encases each pair of insulated conductors.



# Guided Media

---

## • Twisted Pair- UTP and STP

- Unshielded twisted-pair (UTP):
  - Low cost
  - Simple to install
  - High speed
  - Used in telephonic applications.

# Guided Media

---

## • Twisted Pair- UTP and STP

- Shielded twisted-pair (STP):
  - Removes crosstalk.
  - Good speed.
  - Expensive.
  - Difficult to manufacture and install.
  - Used in voice and data channels of telephone lines.

# Guided Media

---

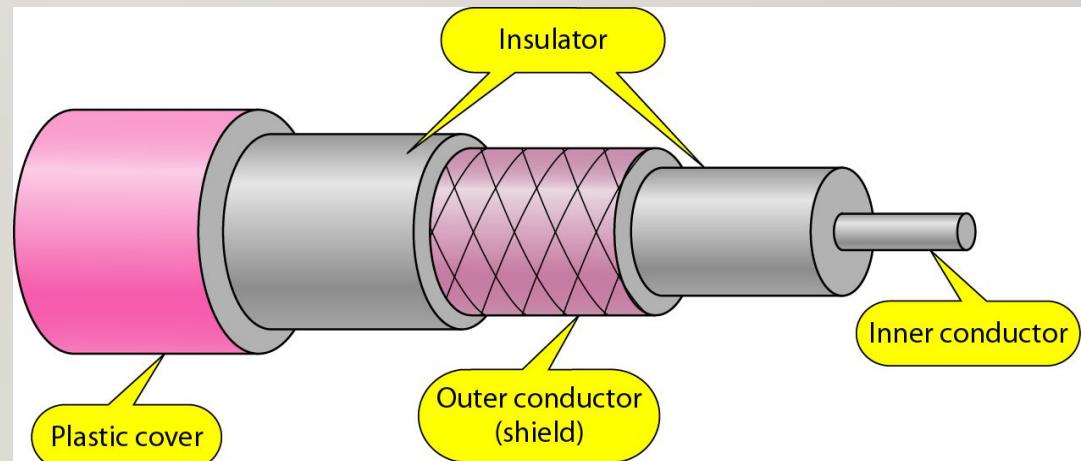
## • Coaxial Cable

- Coaxial cable (or *coax*) carries signals of higher frequency ranges than those in twisted pair cable.
- Coax has a **central core conductor** of solid or stranded wire 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.

# Guided Media

## • Coaxial Cable

- This outer conductor is also enclosed in an insulating sheath, and the whole cable is protected by a **plastic cover**.
- Uses – Ethernet LANs, cable TV



# Guided Media

---

## • **Coaxial Cable**

- High bandwidth
- Good noise immunity
- Low cost
- Simple to install.
- Failure of cable can disturb whole network.

# Guided Media

---

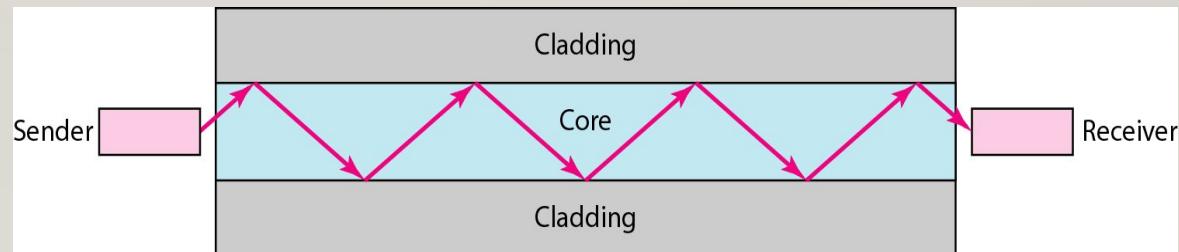
## • **Fiber Optics Cable**

- 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 cladding causes light to be confined to the core of the fibre.

# Guided Media

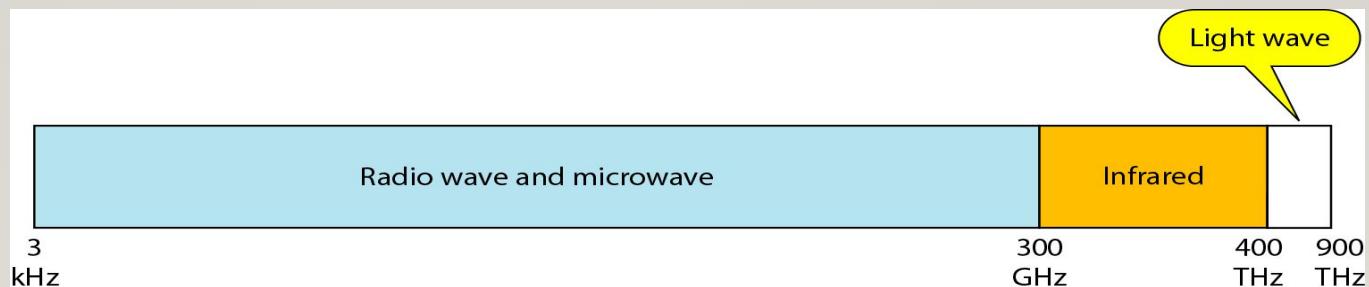
## • Fiber Optics Cable

- Light weight.
- Increased bandwidth.
- Less signal attenuation
- Industry standard for high-speed networking.
- Installation and maintenance is difficult.
- Cost



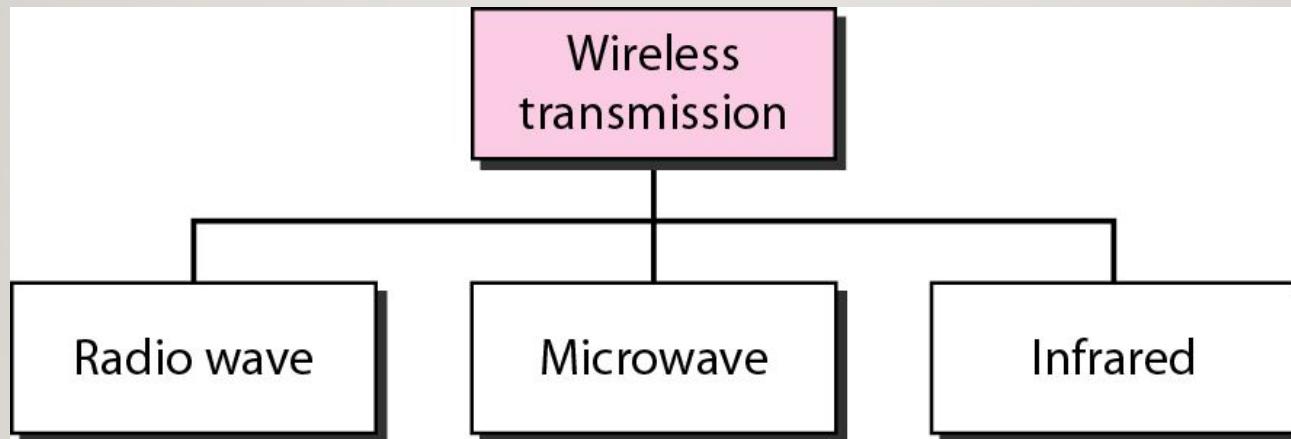
# Unguided Media

- Unguided media transport electromagnetic waves without using a physical conductor.
- This type of communication is often referred to as **wireless communication**.
- Signals are normally broadcast through free space and thus are available to anyone who has a device capable of receiving them.



# Classes of Wireless transmission

---



# Wireless Transmission Waves

---

## I. Radio waves:

- used for multicast communications, such as radio and television, and paging systems.
- can penetrate through walls.
- Frequency ranges from 3KHz to 1 GHz.
- Used in AM and FM radios

# Wireless Transmission Waves

## 2. Microwaves:

---

- used for unicast communication such as cellular telephones, satellite networks, and wireless LANs.
- Frequency ranges from 1GHz to 300GHz.
- Distance covered is directly proportional to antenna's height.

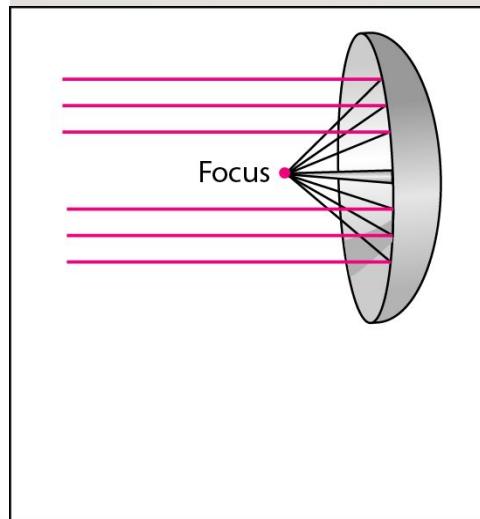
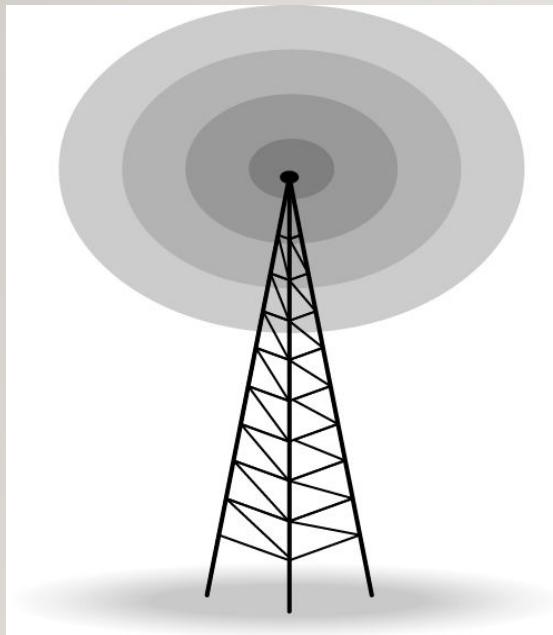
# Wireless Transmission Waves

## 3. Infrared signals:

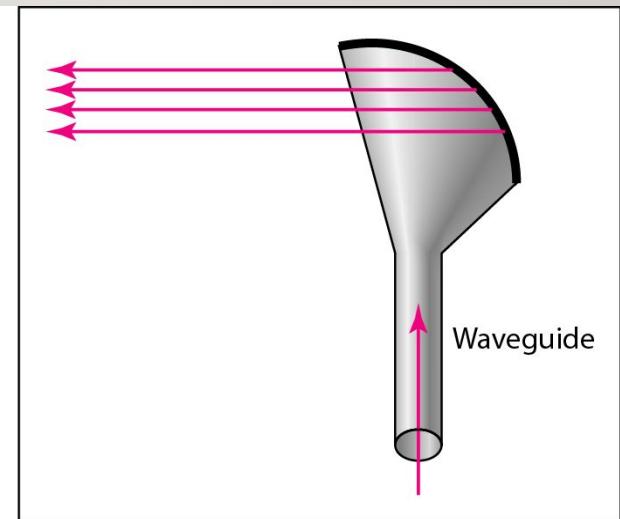
- can be used for short-range communication in a closed area (TV remote operation).
- Cannot go through obstacles.
- High bandwidth, high data rate, minimum interference.
- Unreliable outside the building because the sun rays will interfere with the infrared waves.
- Frequency ranges from 300 GHz to 400 THz.

# ANTENNAS

---



a. Dish antenna



b. Horn antenna

# COMPUTER NETWORKS

## MODULE 2

---

MS. JINCY J FERNANADEZ

ASST PROF, CSE

RSET

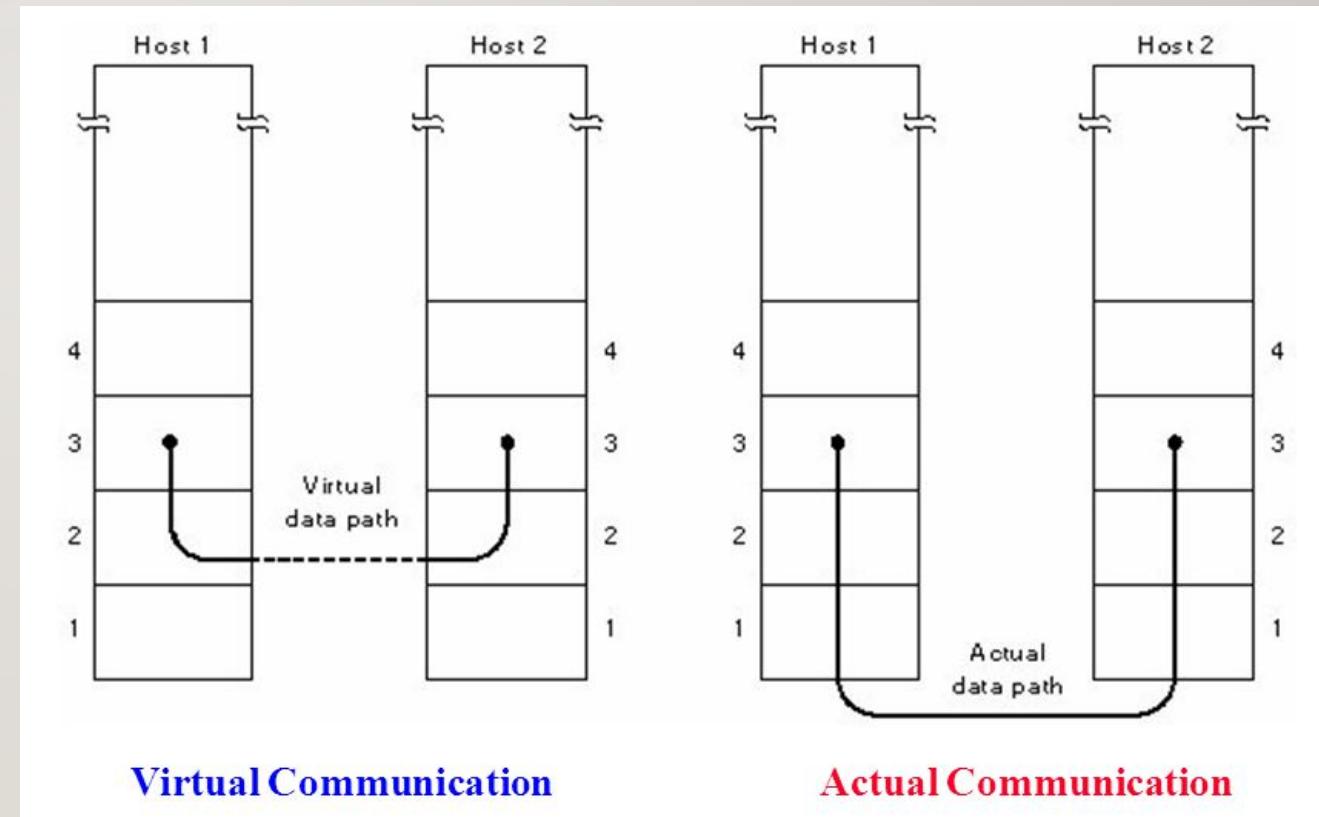
# Data link layer- Design Issues

---

- Provide a well-defined **service interface** to the network layer.
- Deal with **errors** in transmission of frames.
- Regulate the **flow** of data so that slow receivers are not overwhelmed by fast senders.
- **Framing** of data.
- Detect and correct errors in frame data.

# 1. Service to Network layer

- Main service is transferring data from network layer on the source machine to network layer on the destination machine.



# Service to Network layer

---

- **Unacknowledged connectionless service**
  - Source machine send independent frames to the destination machine without any acknowledgement policy.
  - No logical connection established or released.
  - If a frame is lost, no attempt to detect loss or recover it.
  - Used in LANs.

# Service to Network layer

---

- Acknowledged connectionless service
  - No logical connections used here also.
  - Each frame sent is individually acknowledged.
  - In case of a time-out, data will be sent again.
  - Used in unreliable channels- wireless systems.

# Service to Network layer

---

- Acknowledged connection-oriented service
  - Source and destination establish a connection before transferring data.
  - Release connection after transfer of data.
  - Each frame is numbered.
  - Guaranteed delivery of frames.
  - Ordered delivery of frames.
  - Used in WAN subnets containing routers connected by point to point leased telephone lines.

## 2. Error Control

---

- Need to ensure frames are delivered to the destination network layer correctly in proper order.
- Use **acknowledgements**: Positive or negative from receiver.
- Chance of lost messages. So, no positive or negative ACK will come from receiver.
- Use **timers**. Start a timer at sender when a frame is transmitted.
- Timer is set to expire after an interval long enough for frame to reach destination, get processed and ACK to propagate back to sender.
- If frame or ACK is lost. Then timer goes off. Then frame is retransmitted again.
- Chances of receiving same frame multiple times in time delays.
- Use **sequence numbers** for outgoing frames.

## 3. Flow Control

---

- Senders can be running on a fast computer and the receivers on a slow computer.
- Sender pumps out frames at a high rate and receiver gets overwhelmed.
- Even if transmission is error free destination will be unable to handle the frames and they get dropped off.
- Two solutions:
  - Feedback based flow control
  - Rate based flow control

# Flow Control Techniques

---

- Feedback based control
- Receiver sends back information to the sender giving it permission to send more data.
- Depends on current status of receiver.
- Rate based control
- Protocol has a built-in mechanism to limit the rate at which sender may transmit data.
- No feedbacks given.

## 4. Framing

---

- Data link layer encapsulates data into **frames** for transmission.
- Each frame has a frame header, payload field for holding the packets, a frame trailer and flags.



# Frames

---

- A frame has the following parts:
  - Frame Header – It contains the source, and the destination addresses of the frame.
  - Payload field – It contains the message to be delivered.
  - Trailer – It contains the error detection and error correction bits.
  - Flag – It marks the beginning and end of the frame.
- Types of frame:
  - Fixed size frames
  - Variable size frames

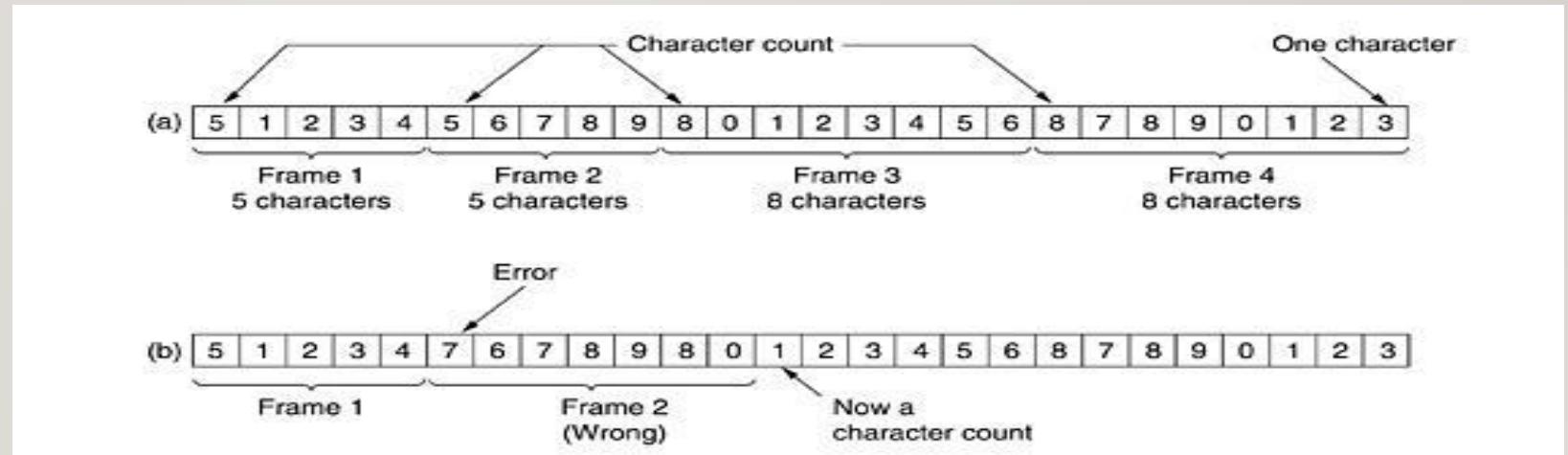
# Methods for Framing

---

- Byte count
- Flag bytes with byte stuffing
- Starting and ending flags with bit stuffing

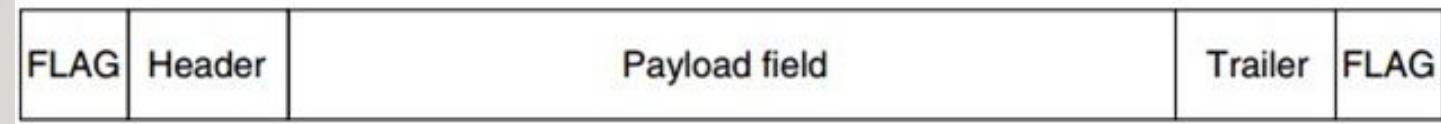
# 1. Byte Count

- Use a field in the header to specify number of bytes in the frame.
- Count can get garbled by a transmission error: unable to locate the correct start of the next frame.
- Asking for retransmission to sender is of no use as well.
- Rarely used.



## 2. Flag Bytes

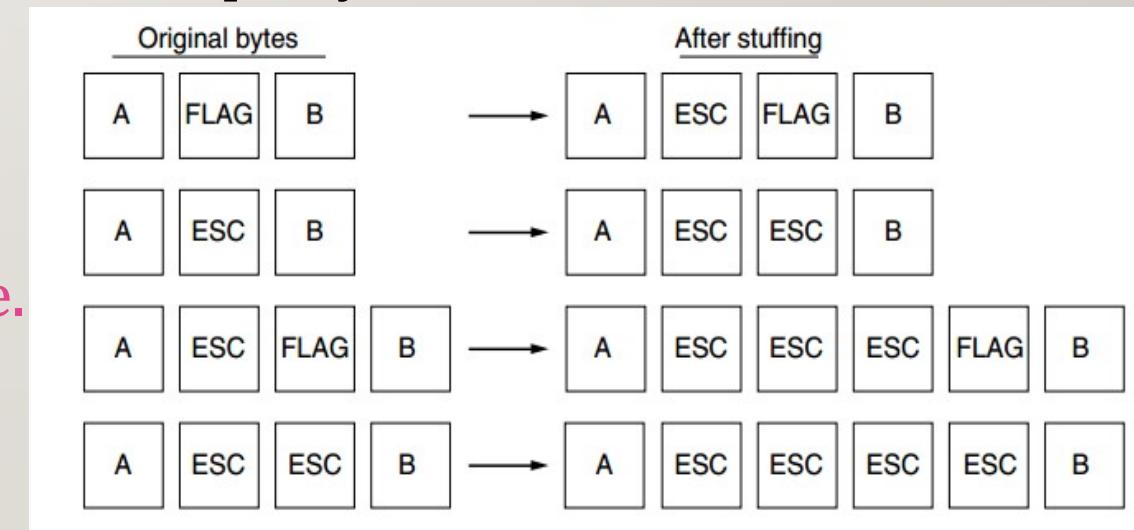
- Each frame starts and ends with special bytes- FLAG byte.



- Two consecutive flag bytes indicate end of one frame and start of next one.
- If receiver loses synchronization it can search for flag byte to get the end of the current frame and start of the next frame.
- Flag byte bit pattern may occur in the data.
- This interferes with framing.

# Flag Bytes with byte stuffing (Character oriented framing)

- Sender's data link layer inserts a special **escape byte(ESC)** just before each accidental flag byte in the data.
- Data link layer on receiving end removes the escape byte before the data is given to network layer.
- If escape byte occurs in the data??
- It is also stuffed with an **extra escape byte**.



# Flag Bytes with bit stuffing (Bit oriented framing)

- Each frame begins and ends with a special pattern 01111110 (flag pattern).
- Whenever there are 5 consecutive 1s in the data, sender automatically stuffs a 0 in the outgoing bit stream. It will be removed at destination data link layer.
- If user data contains the flag pattern 01111110 then it is transmitted as 011111010.

(a) 01101111111111111110010

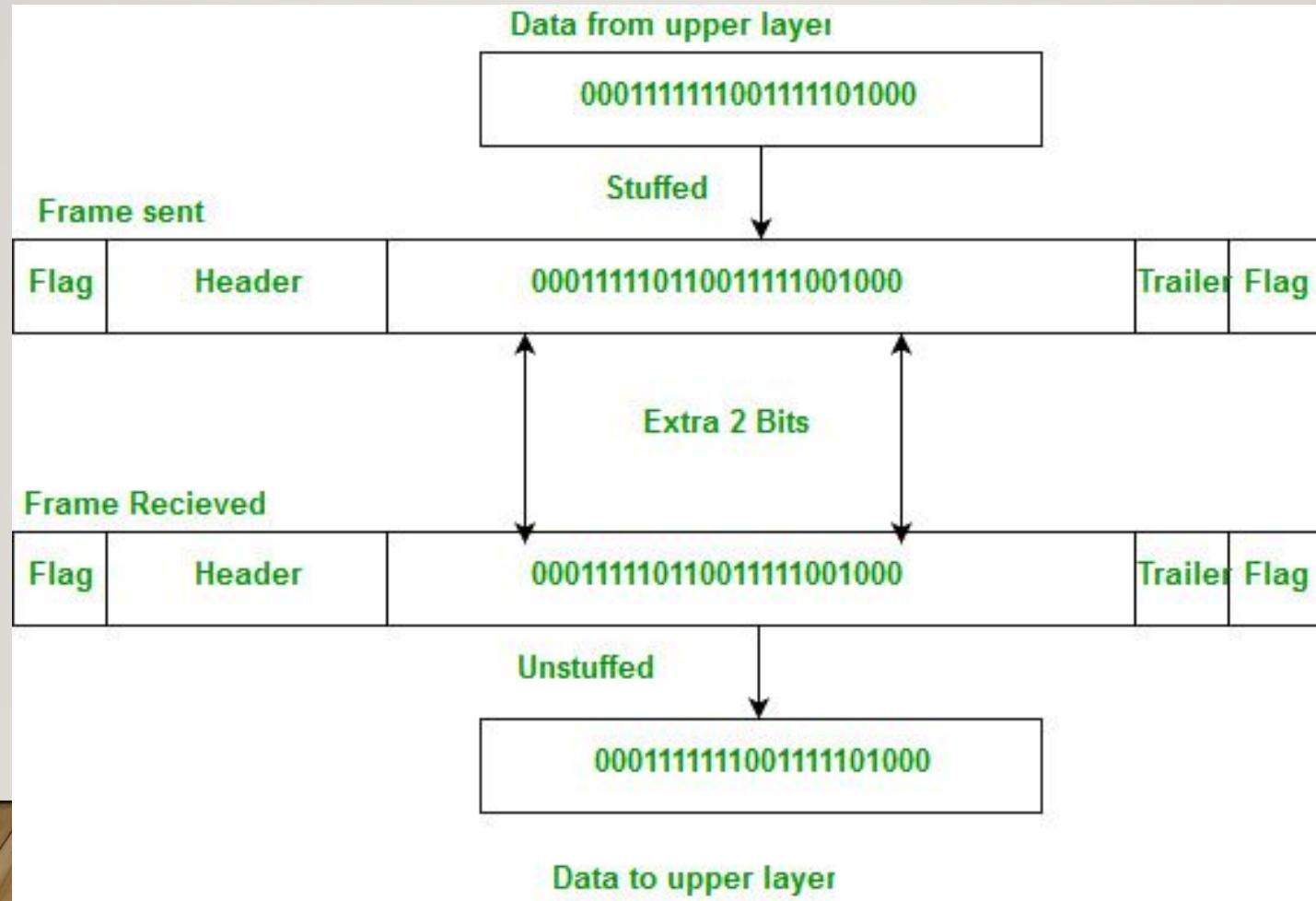
(b) 011011110111110111111010010

↑  
Stuffed bits

(c) 0110111111111111111110010

# Flag Bytes with bit stuffing

## (Bit oriented framing)



## 5. Error Detection and Correction

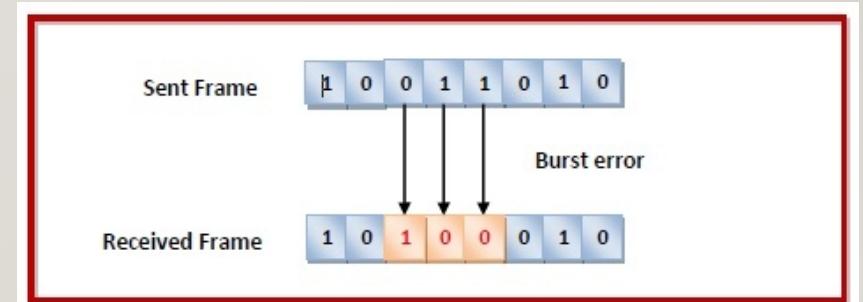
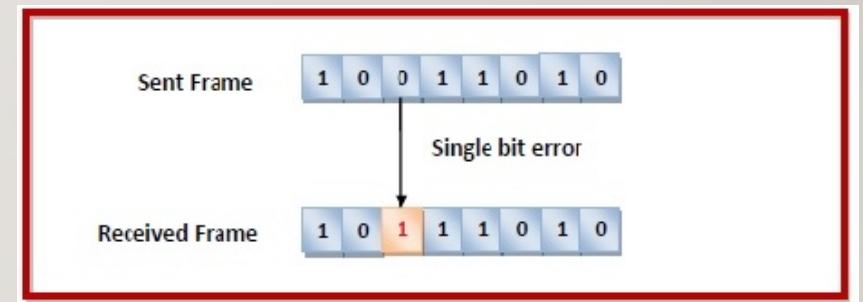
---

- Unpredictable changes during transfer of data due to interference.
- Noise usually occurs as bursts rather than independent, single bit errors.
- Detecting and correcting errors requires redundancy → sending additional information along with the data.
- Redundancy is achieved through various coding schemes.

# Error Detection and Correction

---

- There are two types of errors:
- Single bit error:
  - Only one bit has corrupted.
- Burst error:
  - More than one bits have corrupted.



# Error Detection and Correction

---

- There are two ways to control errors:
- Error Detecting Codes:
  - Check for whether error occurred or not.
  - The number of error bits and the type of error does not matter.
- Error Correcting Codes:
  - Need to know the exact number of bits that are corrupted and their location in the message.

# Coding Scheme

---

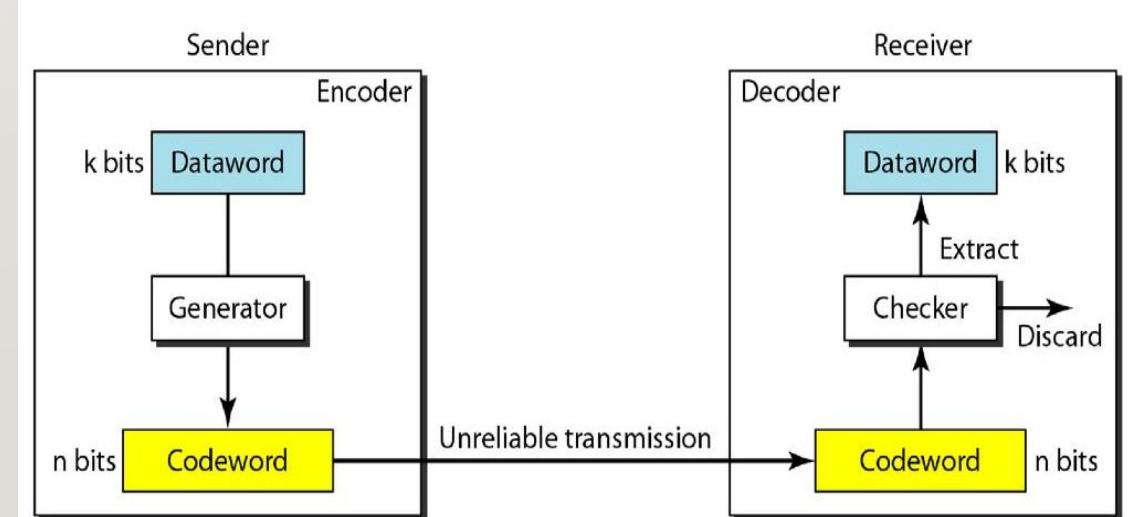
- The sender adds redundant bits that creates a relationship between the redundant bits and the actual data. The receiver checks the relationships between the two sets of bits to detect errors.
- Important factors of coding scheme:
  - The ratio of redundant bits to the data bits
  - Robustness of the coding process
- Two types of Coding
  - Block Coding: information bits are immediately followed by parity bits.
  - Convolution Coding: information bits are not followed by parity bits instead spread along the sequence.

# Coding Scheme

---

- **Block Coding:**

- Divide the message into blocks, each of 'm' bits called data words.
- Add 'r' redundant (check or parity) bits to each block ( $n=m+r$ ).  $\rightarrow$  codewords.
- With 'm' bits,  $2^m$  data words are possible.
- One to one coding.
- Code rate= fraction of the codeword that carries information that is not redundant.
- Code rate=  $\frac{m}{r}$



# Coding Scheme

---

- Block Code:
  - Systematic Code:
    - 'm' data bits are sent directly, along with check bits.
  - Linear code:
    - 'r' check bits are computed as a linear function of the 'm' data bits.
    - XOR is a popular function.

# Error Correcting Codes- Block codes

---

## • Hamming Distance

- Between two words is the number of distances between the corresponding bits.
- The number of bit positions in which two codewords differ.
- The Hamming distance  $d(000, 011)$  is 2

$000 \oplus 011$  is 011 (two 1s)

- The Hamming distance  $d(10101, 11110)$  is 3

$10101 \oplus 11110$  is 01011 (three 1s)

# Error Correcting Codes- Block codes

## • Hamming Distance

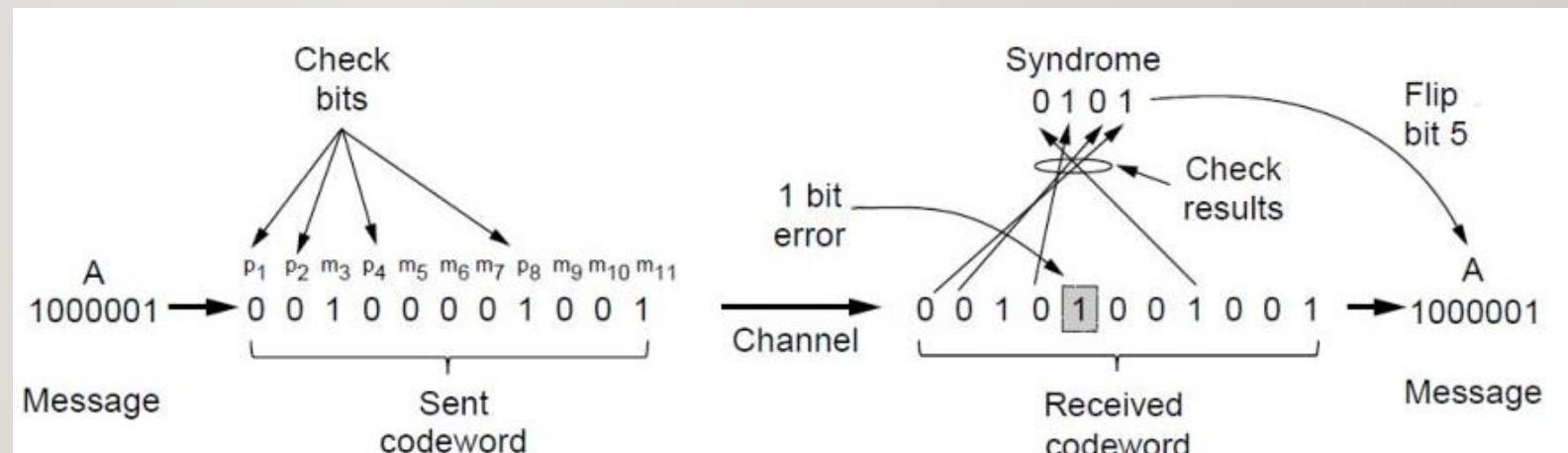
- The bits of the codeword are numbered consecutively.
- The bits that are powers of 2 ( $1, 2, 4, 8, \dots$ ) are check bits (parity bits) and rest are filled up with 'm' data bits.
- $(n,m)$  hamming code.

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

# Error Correcting Codes- Block codes

## • Hamming Distance

- The bits of the codeword are numbered consecutively.
- The bits that are powers of 2 ( $1, 2, 4, 8, \dots$ ) are check bits and rest are filled up with 'm' data bits.
- $(n, r)$  hamming code.



Example of an (11, 7) Hamming code  
correcting a single-bit error.

# Error Correcting Codes- Block codes

- 
- If the 7-bit hamming word received by a receiver is 1011011. Assuming the even parity, state whether the received code word is correct or wrong. If wrong, locate the bit having error and do the correction.

# Error Detecting Codes

---

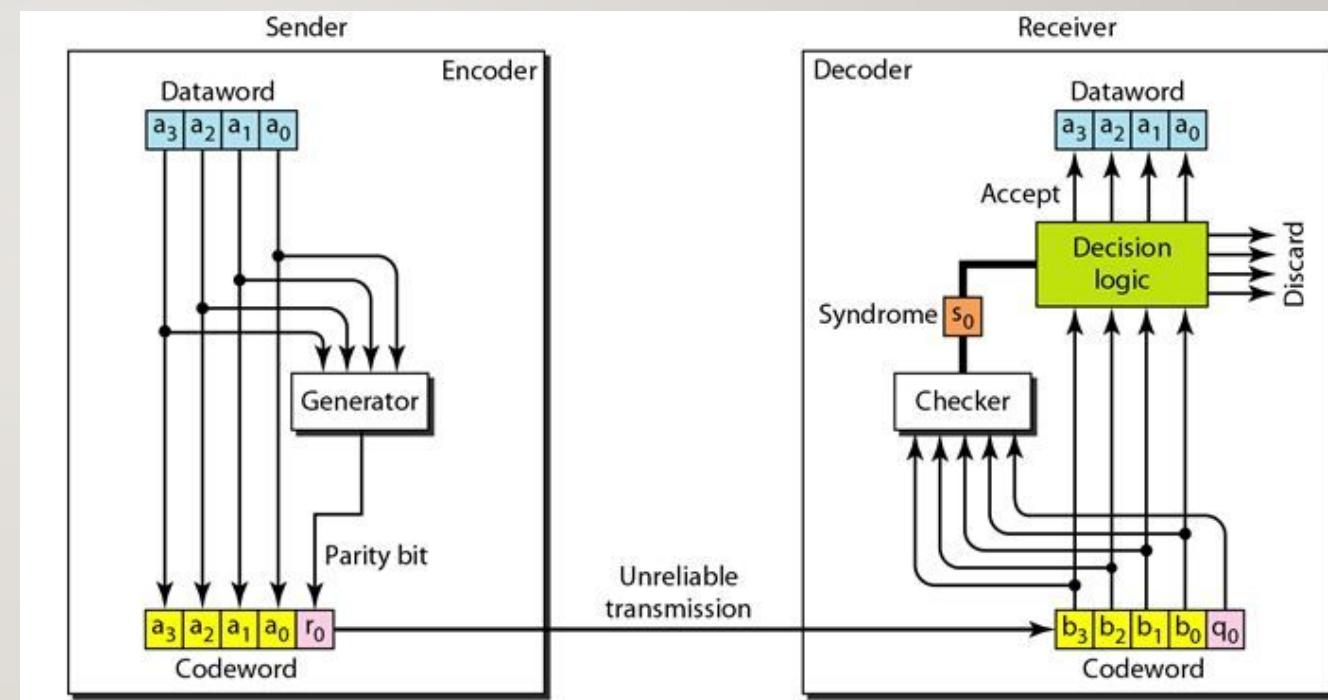
## • Minimum Hamming Distance

- In a set of codewords, the minimum hamming distance is the smallest hamming distance between all possible pairs of codewords.
- To guarantee the detection of up to 'n' errors, the minimum hamming distance must be  $d_{min} = n + 1$ .

# Error Detecting Codes

## • Parity

- Sender adds parity bit.
- Assume the coding scheme follows even parity.
- If the data word contains odd number of 1's, then parity bit=1.
- If the data word contains even number of 1's, then parity bit=0.
- Detect only single bit errors.



# Error Detecting Codes-Checksum

---

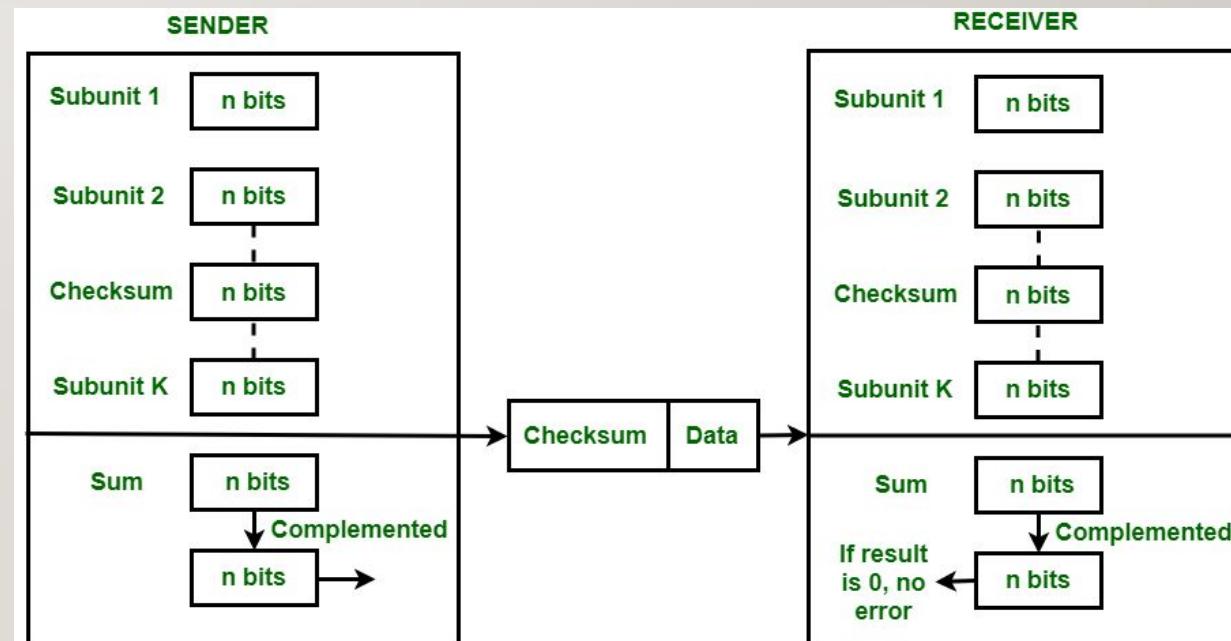
- A **checksum** is a small-sized block of data derived from another block of digital data for the purpose of detecting errors.
- Error detecting technique that can be applied to message of any length.
- Sender side ↴ checksum creation
- Receiver side ↴ checksum validation

# Error Detecting Codes-Checksum

---

- Steps

- Break the original message into ‘k’ number of blocks with ‘n’ bits in each block.
- Sum all the ‘k’ data blocks.
- Add the carry to the sum, if any.
- Checksum= 1’s complement of the sum obtained.

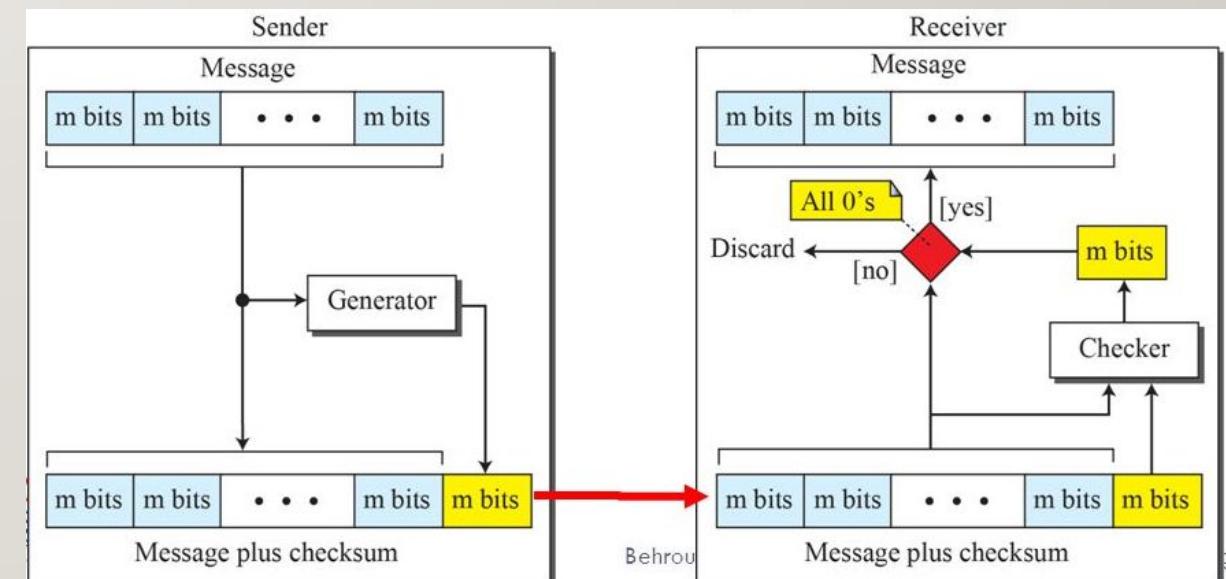


# Error Detecting Codes-Checksum

---

- **Steps: Sender Side**

- Break the original message into ‘k’ number of blocks with ‘m’ bits in each block.
- Sum all the ‘k’ data blocks.
- Add the carry to the sum, if any.
- Checksum= 1’s complement of the sum obtained.

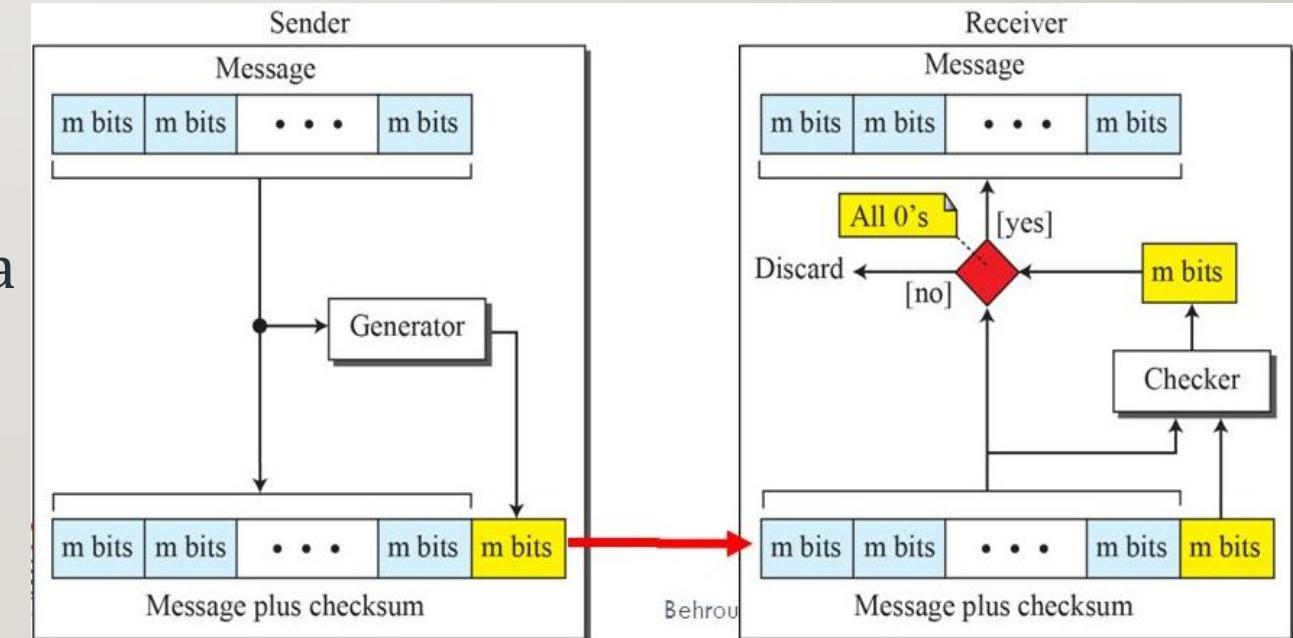


# Error Detecting Codes-Checksum

---

- Steps: Receiver Side

- All received segments are added using 1's complement arithmetic to get the sum.
- Complement the sum.
- If the result is zero, the received data accepted; otherwise discarded.



# Error Detecting Codes-Checksum

- Input: 1100110010101010

1111000011000011

Sender's End	Receiver's End
Frame 1: 11001100	Frame 1: 11001100
Frame 2: + 10101010	Frame 2: + 10101010
Partial Sum: 1 01110110	Partial Sum: 1 01110110
+ 1	+ 1
01110111	01110111
Frame 3: + 11110000	Frame 3: + 11110000
Partial Sum: 1 01100111	Partial Sum: 1 01100111
+ 1	+ 1
01101000	01101000
Frame 4: + 11000011	Frame 4: + 11000011
Partial Sum: 1 00101011	Partial Sum: 1 00101011
+ 1	+ 1
00101100	00101100
Sum: 00101100	Checksum: 11010011
Checksum: 11010011	Sum: 11111111
	Complement: 00000000
	Hence accept frames.

# Error Detecting Codes-Checksum

---

- Compute the checksum value of 10010011 10010011 10011000 01001101 of 16 bit segment.
- Ans: 110101000011110

# Error Detecting Codes-CRC

---

- Stands for Cyclic Redundancy Check (Polynomial code).
- Cyclic code: if a codeword is cyclically shifted (rotated), the result is another codeword.
- Polynomial code: treat bit strings as representations of polynomials with coefficients 0 and 1 only.
- k- bit frame  $\rightarrow$  polynomial with 'k' terms, ranging from  $x^{k-1}$  to  $x^0$ .
- E. g. 110001  $\rightarrow 1x^5 + 1x^4 + 0x^3 + 0x^2 + 0x^1 + 1x^0$

# Error Detecting Codes-CRC

---

- Uses Modulo 2 arithmetic
  - No carry for addition or borrows for subtraction.
  - Addition and subtraction are identical to XOR.
  - CRC uses **Generator Polynomial,  $G(x)$**  which is agreed by both sender and receiver side.
  - An example generator polynomial is of the form like  $x^3 + x + 1$ , represents key 1011.
  - Both high and low order bits of the generator polynomial must be 1.

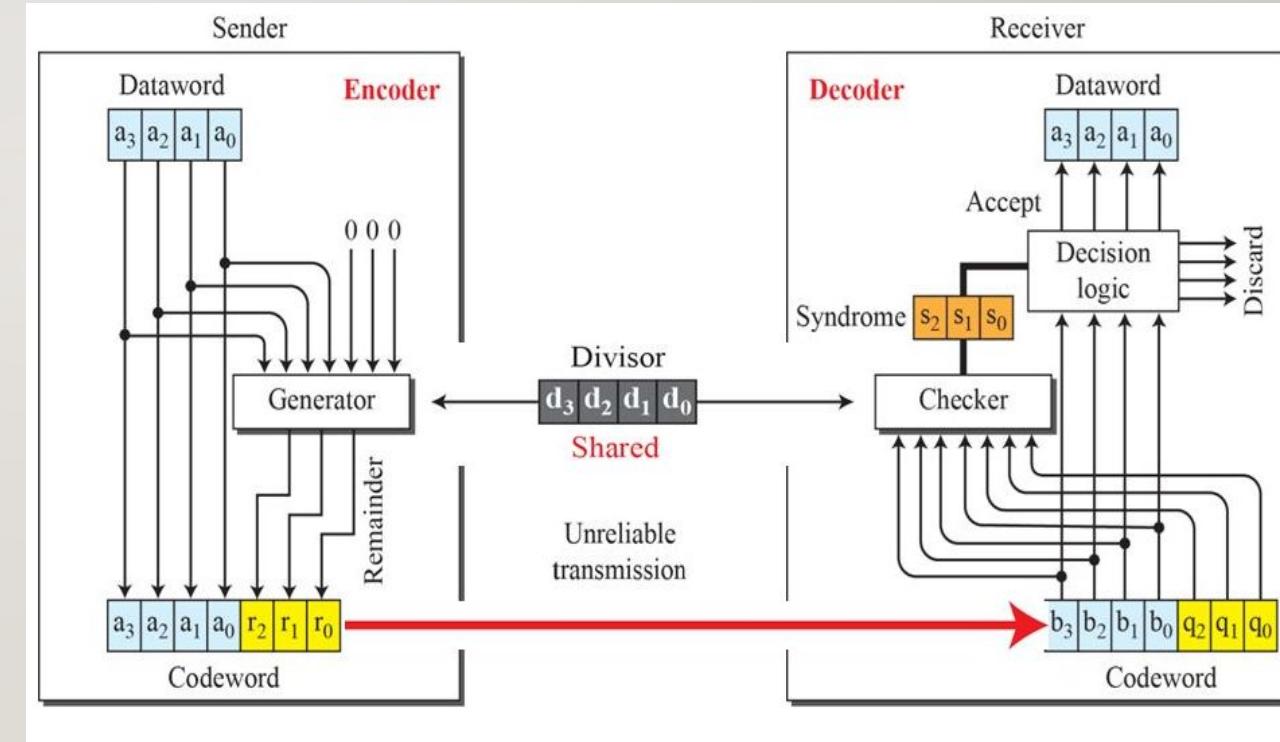
# Error Detecting Codes-CRC

---

- To compute CRC for some frame with ‘m’ bits, the frame must be longer than the generator polynomial.
- Idea is to append a CRC to the end of the frame such that polynomial represented by the frame is divisible by  $G(x)$ .
- Receiver tries dividing it by  $G(x)$ . If there is any remainder  $\Rightarrow$  transmission error.
- Let ‘r’ be the degree of  $G(x)$ . Append ‘r’ zero bits to the low order end of the frame so that it contains  $m+r$  bits.

# Error Detecting Codes-CRC

- Unlike checksum scheme, which is based on addition, CRC is based on binary division.
- Divisor is known to the sender and receiver in advance.
- Perform modulo-2 division.
- Discard the quotient.
- Append the remainder ( $r_2r_1r_0$ ) to the data word to form the code word.

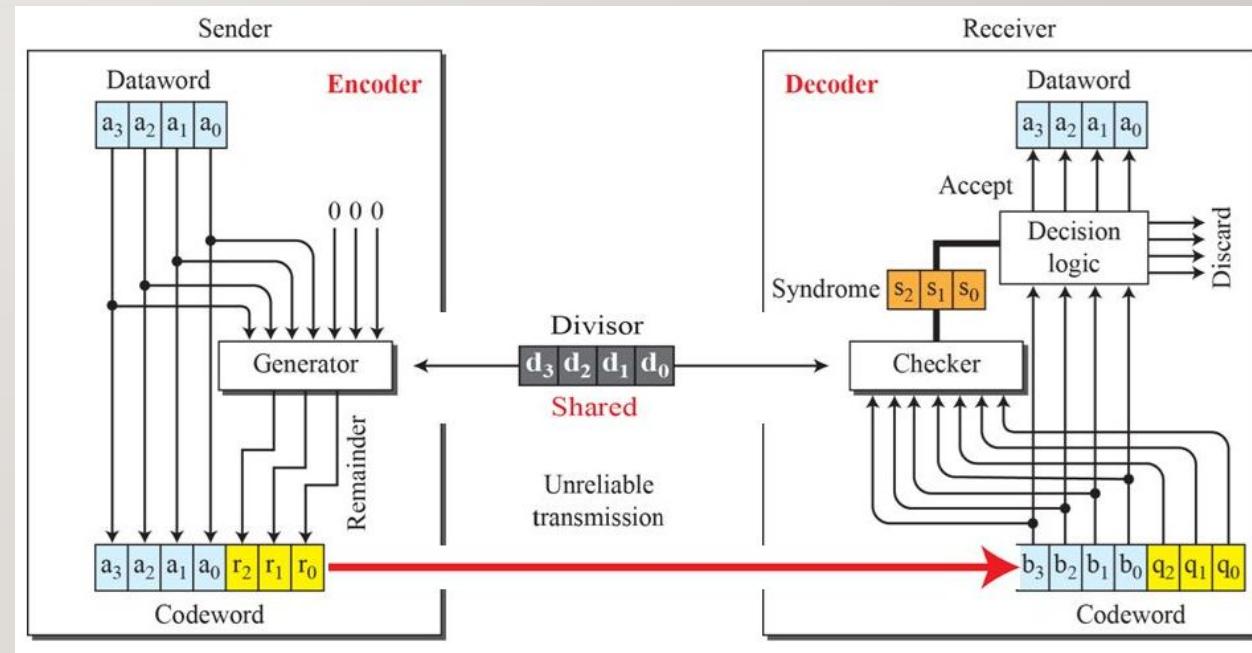


# Error Detecting Codes-CRC

---

- Decoder

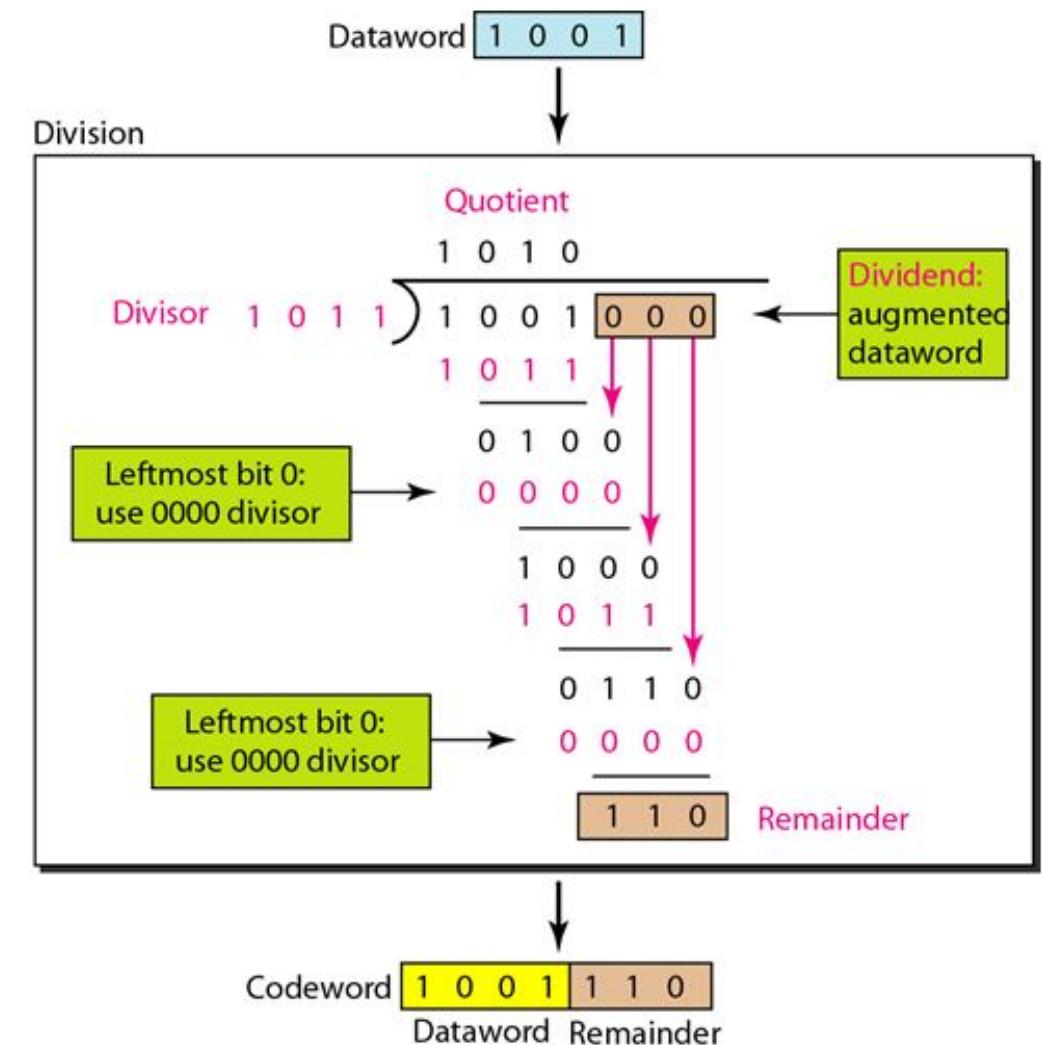
- n- bits are fed to the checker.
- The remainder produced by the checker is a syndrome of n-k bits and fed to the decision logic analyzer.
- Accept if the syndrome bits are all zeros



# Error Detecting Codes-CRC

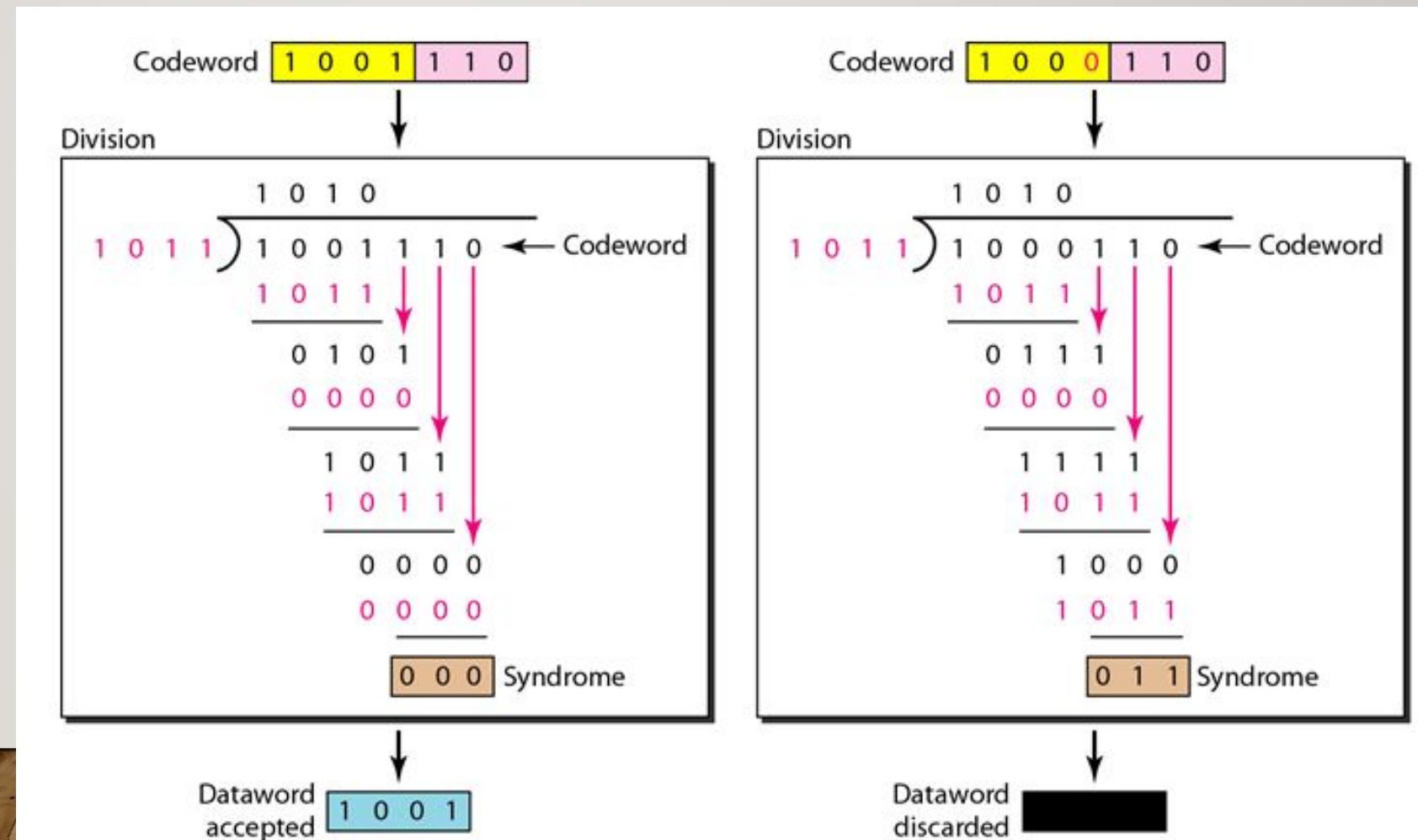
Encoder:

- Modulo 2 Arithmetic:



# Error Detecting Codes-CRC

## Decoder



# Error Detecting Codes-CRC

---

- Q1. Station A wants to send a dataword 1101011111 to station B using CRC .The generator polynomial agreed by both A and B is  $x^4 + x^1 + 1$ . Find the transmitted codeword?
- Ans: 11010111110010

# THANK YOU!!!

---

# COMPUTER NETWORKS

## MODULE 2

---

MS. JINCY J FERNANADEZ

ASST PROF, CSE

RSET

# Introduction to DLL Protocols

---

- A link level protocol that wants to deliver frames reliably must recover the lost frames.
- Two fundamental mechanisms:
  - Acknowledgements
  - Time Outs

# Acknowledgement

---

- An acknowledgement (ACK for short) is a small control frame that a protocol sends back to its peer saying that it has received the earlier frame.
- A control frame is a frame with header only (no data).
- The receipt of an acknowledgement indicates to the sender of the original frame that its data frame was successfully delivered.

# Timeout

---

- If the sender does not receive an *acknowledgment* after a reasonable amount of time, then it retransmits the original frame.
- The action of waiting a reasonable amount of time is called a *timeout*.
- The general strategy of using *acknowledgements* and *timeouts* to implement reliable delivery is called Automatic Repeat reQuest (ARQ).

# Protocols in ARQ

---

- Stop and Wait Protocol
- Sliding Window Protocol
  - Go Back N
  - Selective Repeat

# Stop and Wait Protocol

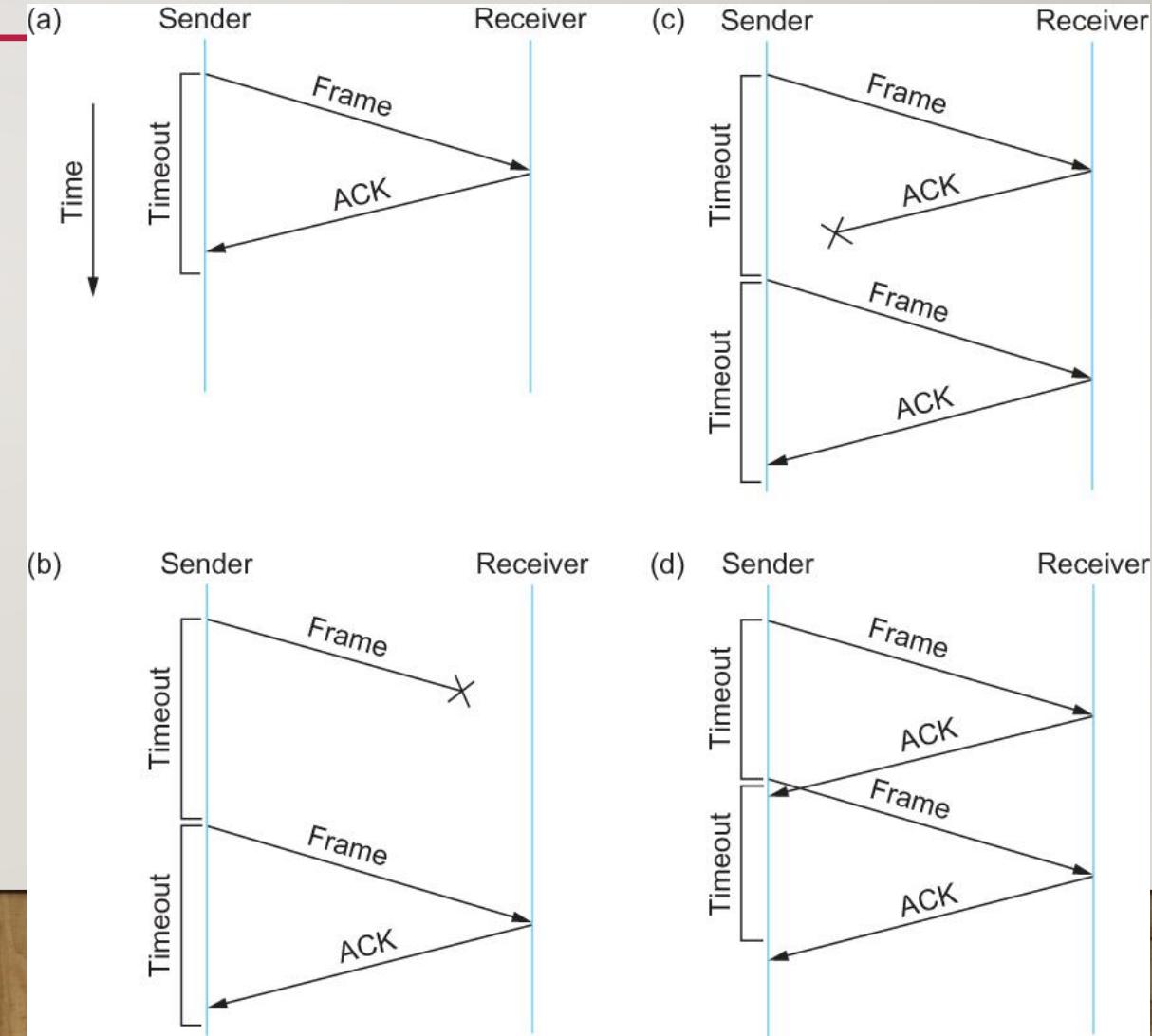
---

- Idea of stop-and-wait protocol is straightforward.
- After transmitting one frame, the sender waits for an acknowledgement before transmitting the next frame.
- If the acknowledgement does not arrive after a certain period of time, the sender times out and retransmits the original frame.

# Stop and Wait Protocol

Timeline showing four different scenarios for the stop-and-wait algorithm.

- (a) The ACK is received before the timer expires;
- (b) the original frame is lost;
- (c) The ACK is lost;
- (d) the timeout fires too soon



# Stop and Wait Protocol

- If the acknowledgment is lost or delayed in arriving:
  - The sender times out and retransmits the original frame, but the receiver will think that it is the next frame since it has correctly received and acknowledged the first frame.
  - As a result, duplicate copies of frames will be delivered.
- How to solve this??
  - Use 1 bit **sequence number** (0 or 1).
  - When the sender retransmits frame 0, the receiver can determine that it is seeing a second copy of frame 0 rather than the first copy of frame 1 and therefore can ignore it (the receiver still acknowledges it, in case the first acknowledgement was lost).

# Stop and Wait Protocol

---

- Sender's Window size (  $W_s$ ) = Receiver's Window size (  $W_r$ )= 1.
- Sequence number= [0,1]

# Stop and Wait Protocol- Features

---

- The sending device keeps a copy of the sent frame transmitted until it receives an acknowledgment (ACK).
- The sender starts a timer when it sends a frame. If an ACK is not received within an allocated time period, the sender resends it.
- Both frames and acknowledgment (ACK) are numbered alternately 0 and 1( two sequence number only).
- This numbering allows for identification of frames in case of duplicate transmission.
- The acknowledgment number defines the number of next expected frame. (frame 0 received ACK 1 is sent).

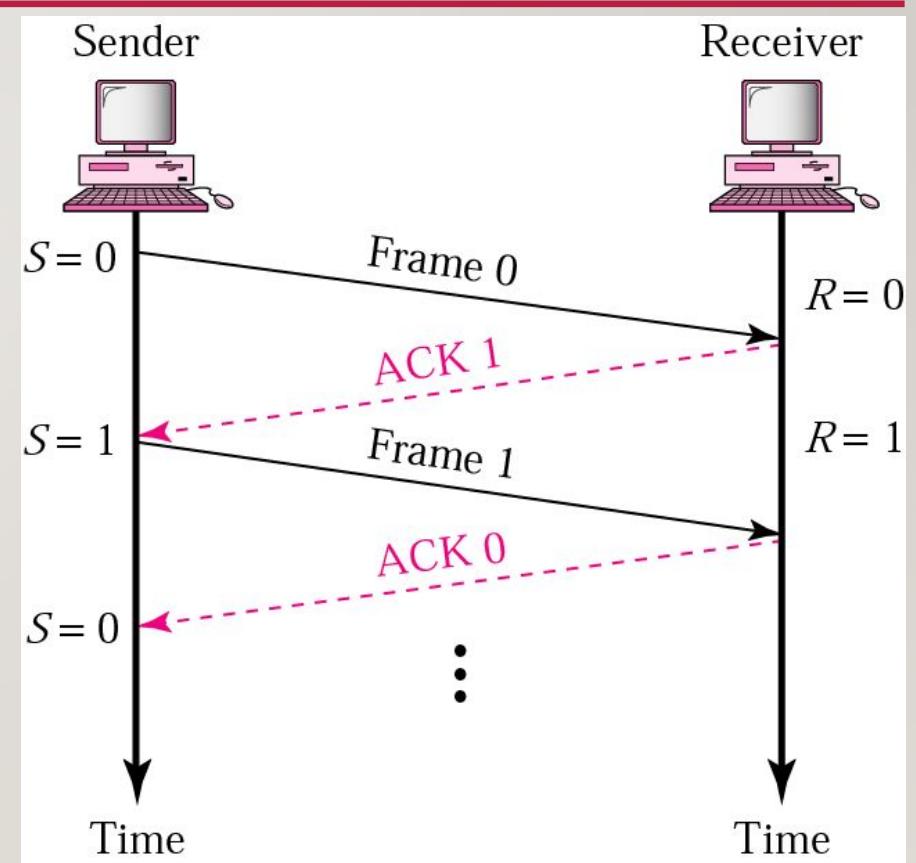
# Stop and Wait Protocol- Features

---

- A damage or lost frame treated by the same manner by the receiver.
- If the receiver detects an error in the received frame or receives a frame out of order, it simply discards the frame.
- The receiver send only positive ACK for frames received safe; it is silent about the frames damage or lost.
- The sender has a control variable holds the number of most recently sent frame S(0 or 1). The receiver has control variable R, that holds the number of the next frame expected (0 or 1).

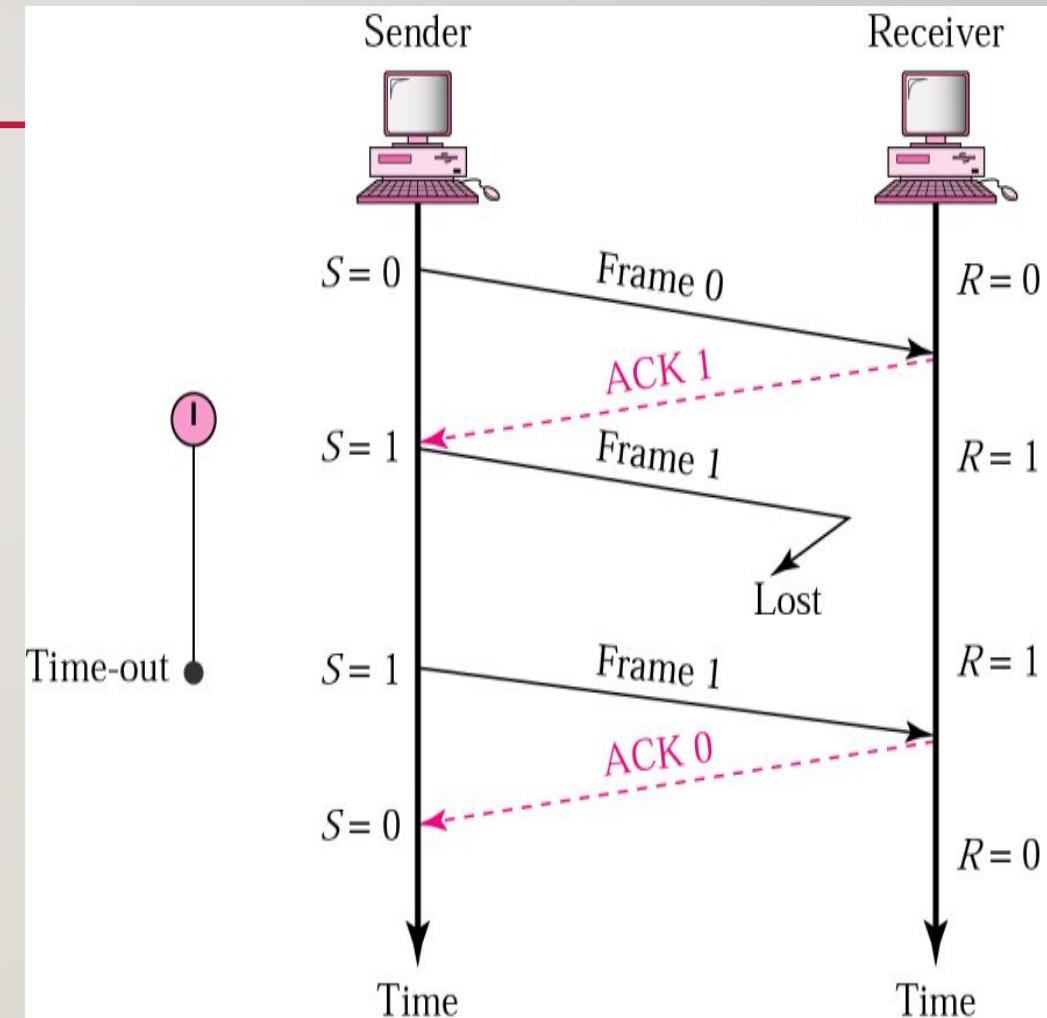
# Stop and Wait Protocol- Normal operation

- The sender will not send the next frame until it is sure that the current one is received.
- Sequence number is necessary to check for duplicated frames .



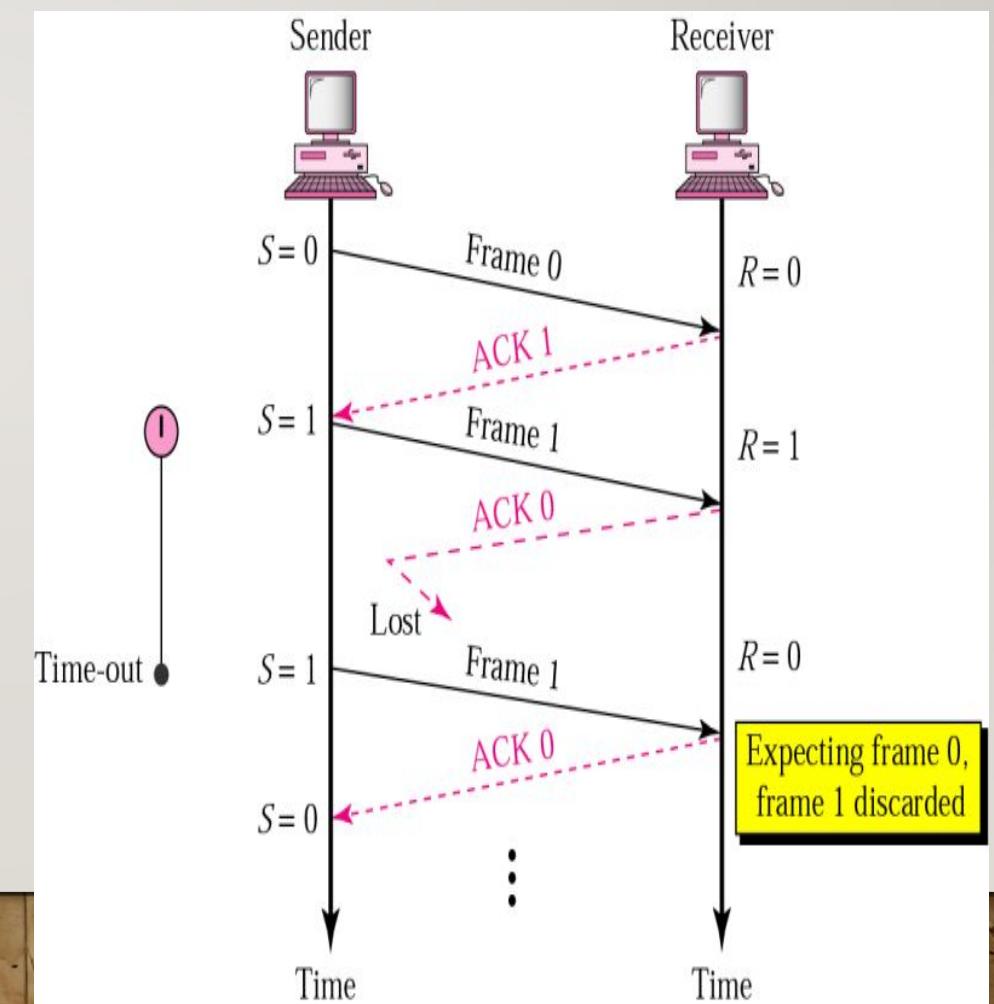
# Stop and Wait Protocol- Lost or Damaged frame

- A damage or lost frame treated by the same manner by the receiver.
- No ACK when frame is corrupted or duplicate.



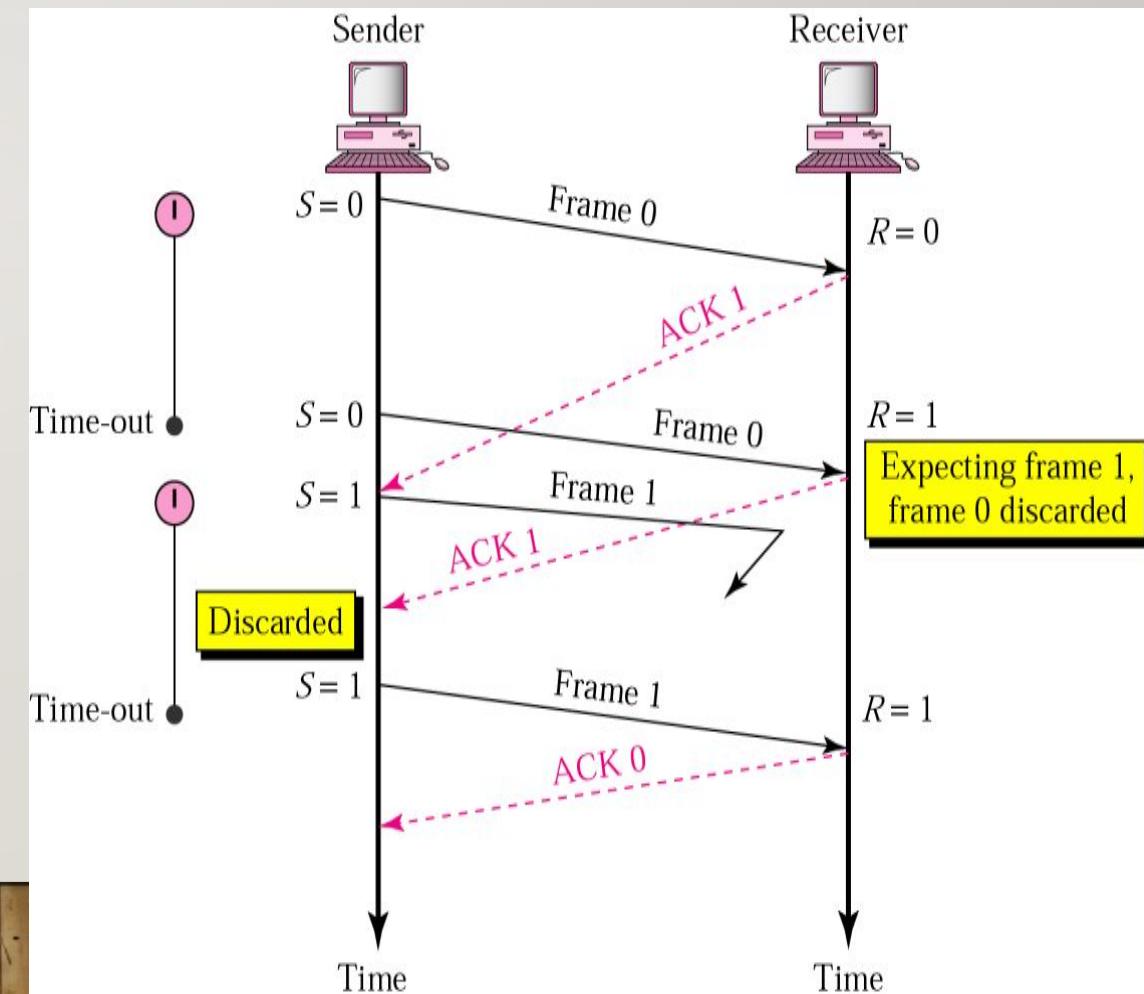
# Stop and Wait Protocol- Lost ACK

- Importance of frame numbering.
- Prevents retaining duplicate frames.



# Stop and Wait Protocol- Delayed ACK & Lost frame

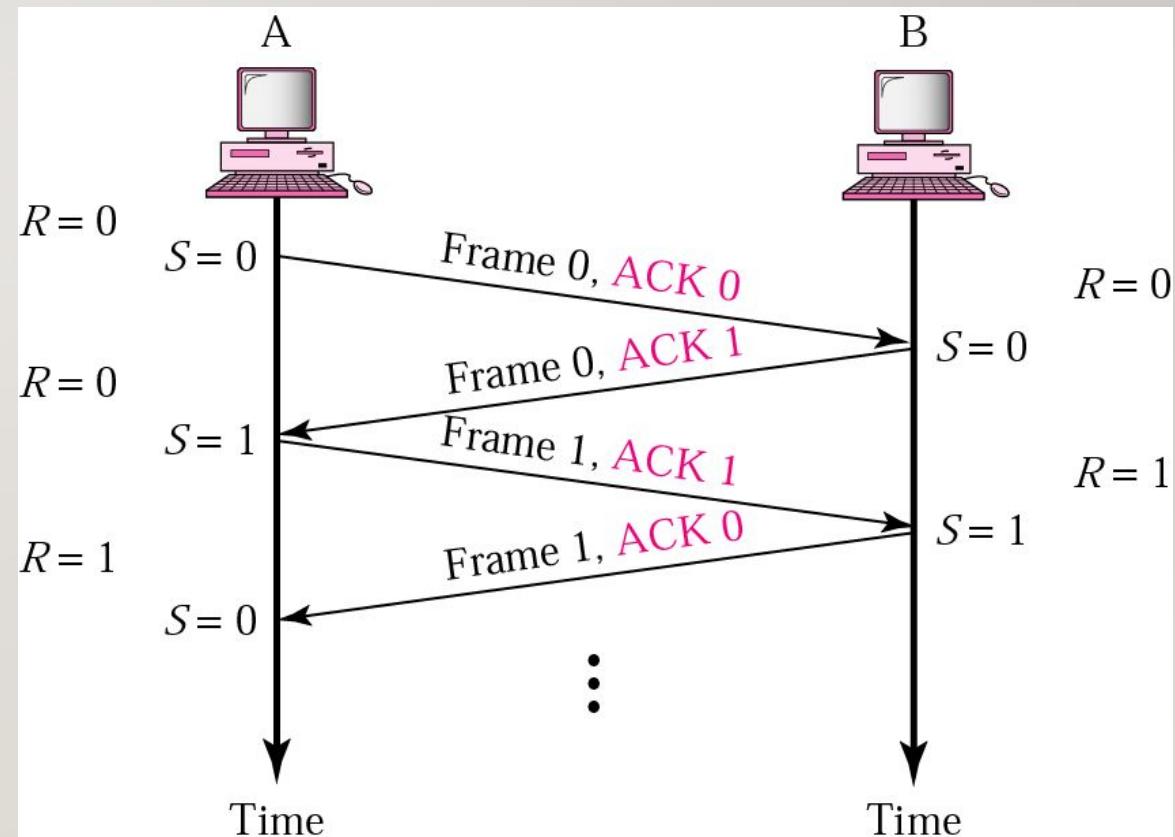
- Importance of frame numbering.
- Numbered acknowledgments are needed if an acknowledgment is delayed, and the next frame is lost.



# Piggybacking

---

- Method to combine a data frame with an acknowledgment.
- Used in bidirectional cases.
- It can save bandwidth because data frame and an ACK frame can be combined into just one frame.



# Stop and Wait- Disadvantage

---

- After each frame sent the host must wait for an ACK.
  - ❖ Inefficient use of bandwidth.
- To improve efficiency ACK should be sent after multiple frames only.
- Alternatives: Sliding Window protocol.

# Sliding Window Protocols

- For sending multiple frames at a time, thus improves efficiency of the transmission.
- No. of frames to be sent at a time is based on window size.
- **Outstanding frames:** frames sent but not acknowledged.
- Can send up to  $W$  frames and keep a copy of outstanding frames until the ACKs arrive.
- Each frame is numbered? **sequence number**.
- Sequence number is stored in the header of the frame.
- If the header has ' $m$ ' bits for sequence number, it ranges from 0 to  $2^m - 1$ .
- If  $m = 3$ , sequence ranges from 0 to 7.

# Sliding Window Protocols

---

- Sliding window is used to hold the **unacknowledged outstanding frames**.
- 2 bidirectional sliding window protocols with (max sending & receiving window size) are:
  - 1-bit sliding window (1,1)
  - Go back N ARQ(>1,1)
  - Selective Repeat ARQ(>1,>1)
- They differ in efficiency, complexity and buffer requirements.

# 1-bit Sliding Window Protocol

---

- Window size=1.
- Uses stop and wait since the sender transmits a frame and waits for its acknowledgement before sending the next one.

# Go back N Sliding Window Protocol

---

- Uses the concept of protocol pipelining: multiple frames can be transmitted before receiving acknowledgement for the first frame.
- Finite number of frames and frames are numbered sequentially.
- The number of frames that can be send depends on sender window size.
- If acknowledgement of a frame is not received within the time interval, all frames in the current window are retransmitted.

# Go back N Sliding Window Protocol

---

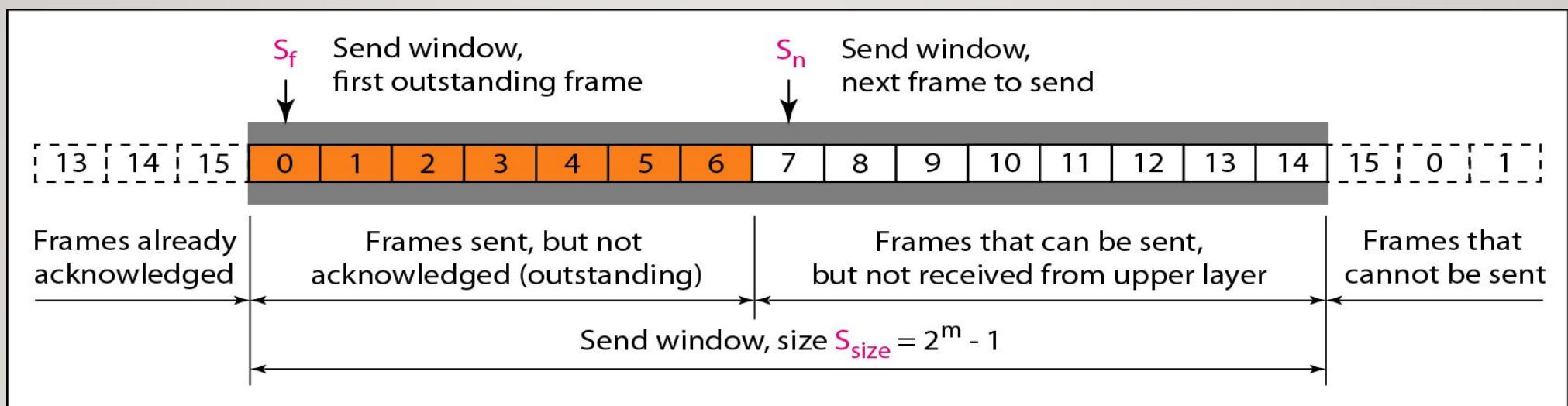
- Uses cumulative acknowledgement technique
- ‘N’ → Sender’s window size ( $W_s = N$ ).
- ‘N’ is the number of frames that can be sent at a time before receiving acknowledgement.
- Receiver’s window size is always 1 ( $W_R = 1$ ).

# Go back N Sliding Window Protocol

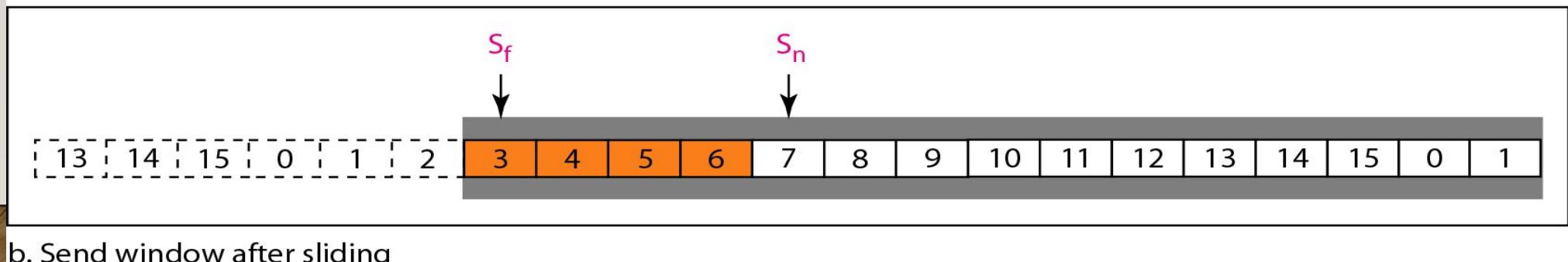
---

- Size of the sending window determines the sequence numbers of the frames.
- E. g. if the sending window size=4, then the sequence numbers will be 0, 1, 2, 3, 0, 1, 2, 3 and so on.
- Let 'm' represents number of bits to represent sequence number, the range of sequence numbers assigned are 0 to  $2^m - 1$ .

# Go back N Sliding Window Protocol



a. Send window before sliding



b. Send window after sliding

# Go back N Sliding Window Protocol- Features

---

- One timer for the first outstanding frame.
- The receiver sends a positive ACK if a frame has arrived safe and in order.
- If a frame is damaged or out of order ,the receiver is silent and will discard all subsequent frames.
- When the timer of an unacknowledged frame at the sender site is expired , the sender goes back and resend all frames , beginning with the one with expired timer.

# Selective Repeat Sliding Window Protocol

---

- Go back N protocol works well when errors are rare.
- Wastes a lot of bandwidth on retransmitted frames.
- Resent only the damaged frame while the correct frames are received and buffered.
- Receiver keeps track of the sequence numbers, buffers the frames in memory and send NACK for only frames which is missing or damaged.
- Both sender and receiver maintain a window of outstanding and acceptable sequence numbers, respectively.

# Selective Repeat Sliding Window Protocol

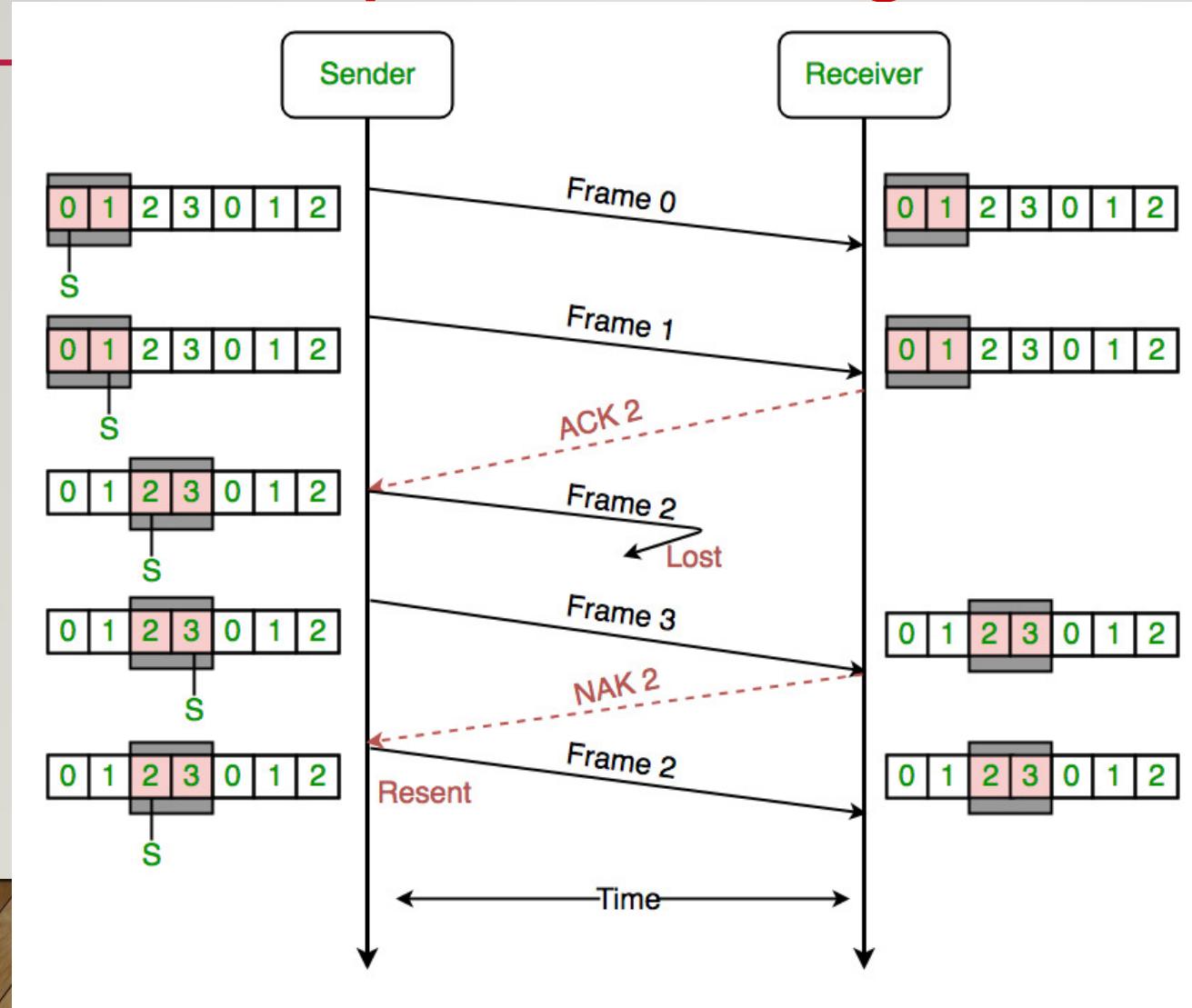
- Seq number =  $2^m$ , where 'm' is the number of bits to represent sequence number.
- E. g. : Let m=2, seq num=[0,1,2,3]
- Sender's Window size (  $W_s$ ) = Receiver's Window size (  $W_r$ )=  $2^{m-1}$ .
- Window size should be less than or equal to half the sequence number in Selective Repeat protocol ( $W_s \leq \frac{2^m}{2} \leq 2^{m-1}$ ).
- Receiver must be able to accept packets out of order.
- Since receiver must release packets to higher layer in order, the receiver must be able to buffer some packets.

# Selective Repeat Sliding Window Protocol

---

- When a frame arrives , its sequence number is checked to see if it falls within the window.
- If so, and if it has not already been received, it is accepted and stored.
- No. of retransmissions < that in Go back N ARQ
- Sender will retransmit only the packets for which NACK is received.
- It is more efficient for noisy link, but the processing at the receiver is more complex.

# Selective Repeat Sliding Window Protocol

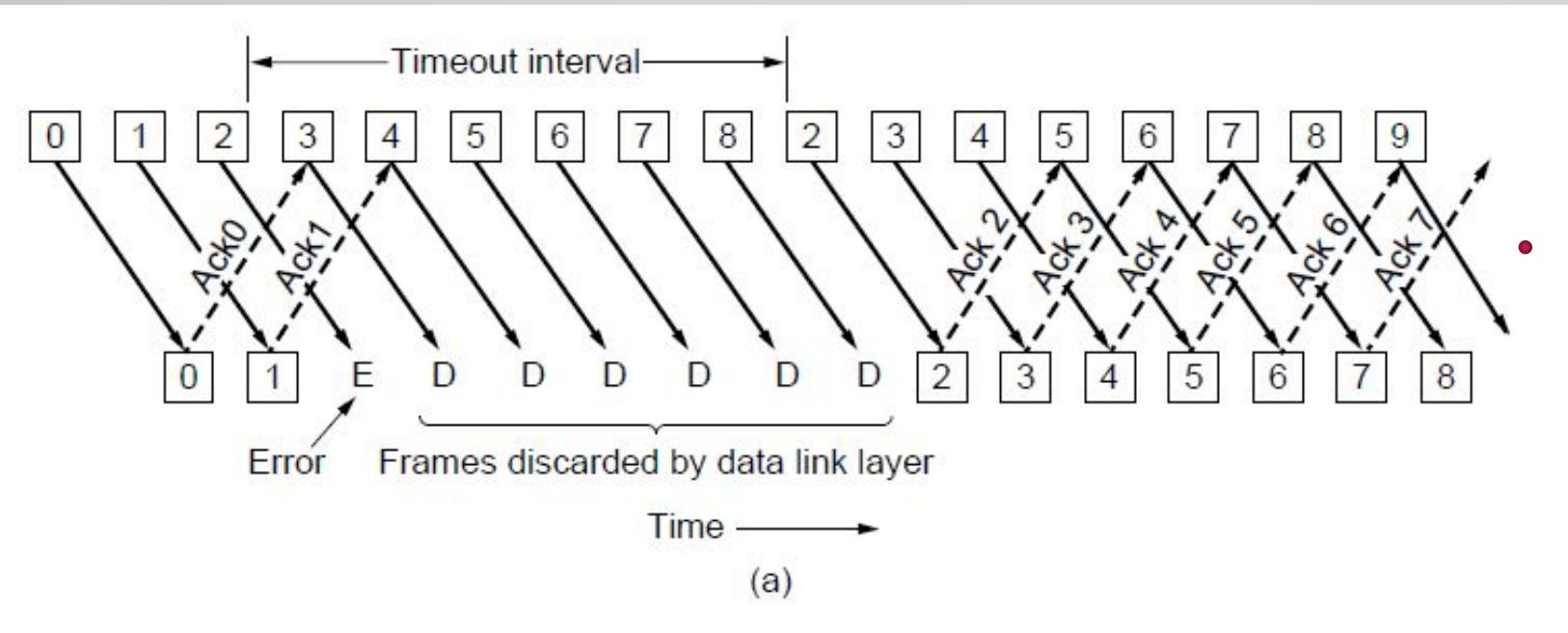


## GO BACK N

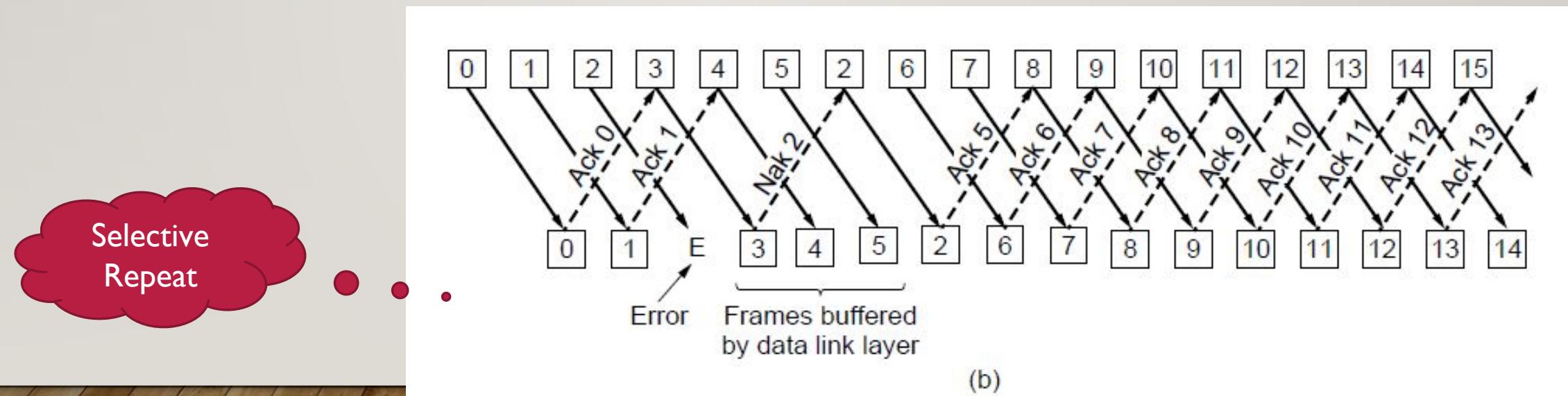
- Receiver in case of error discards all subsequent frames.
- Sends no ACK for discarded frames.
- Receiver window size =1.
- Data link refuses to accept any frame except the next one it must give to the network layer.
- Eventually sender will time out and retransmit all unacknowledged frames in order starting with damaged or lost one.
- Waste of bandwidth.

## SELECTIVE REPEAT

- Receiver in case of error discards bad frame, buffers all good frames.
- Sends a (NAK) when it detects an error.
- Receiver window size  $>1$ .
- Buffers remaining frames received.
- When sender times out, only the oldest unacknowledged frame is retransmitted as others are already buffered.
- Better use of bandwidth.



(a)



(b)

# Sliding Window Protocol

---

- Q1. Using 5 bit sequence numbers what is the maximum size of sender and receiver window for the following protocols: a) stop and wait ARQ (b) go back n ARQ (c) selective repeat ARQ

- (a)  $W_s = 1 \quad W_R = 1$
- (b) The maximum sender window size in go-back n is  $(2^n)-1$ , n is the sequence number.

$$W_S = 31, \quad W_R = 1$$

- (c) The maximum window size is  $(2^n)/2$

$$W_S = W_R = 16$$

# Sliding Window Protocol

- Q2. 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?

With 5 bits, no of sequence no. possible is 32 ie (0-31)

For frames 1-32 seq no 0-31

For frame 33-64 seq no 0-31

For frame 65-96 seq no 0-31

Now 97 98 99 100

0 1 2 3

So after sending 100 packets sequence no is 4

# Sliding Window Protocol

---

- Q3. Station A needs to send a message consisting of 9 packets to station B using a sliding window (window size=3) and go back N strategy. All packets are ready and immediately available for transmission. If every 5<sup>th</sup> packet that A transmits gets lost (but no ACKs from B ever get lost), then what is the number of packets that A transmit to B?
- Ans: 16

# Sliding Window Protocol

---

- Q4. Host A wants to send 10 frames to host B. The hosts agreed to go with Go back 4. How many frames are transmitted by Host A if every 6<sup>th</sup> frame that is transmitted by host A is either corrupted or lost?
- Ans: 17

# Sliding Window Protocol

- Q5. In SR protocol, suppose frames through 0 to 4 have been transmitted. Now imagine that 0 times out, 5(a new frame) is transmitted, 1 times out, 2times out and 6 (another new frame) is transmitted. At this point, what will be the outstanding packets in sender's window?
  - Ans: 3 4 0 5 1 2 6

# Sliding Window Protocol

---

- Q4. Host A wants to send 10 frames to host B. The hosts agreed to go with SR protocol. How many frames are transmitted by Host A if every 6<sup>th</sup> frame that is transmitted by host A is either corrupted or lost?
- Ans: 11

# THANK YOU!!!

---

# **COMPUTER NETWORKS**

## **MODULE 2**

---

**MS. JINCY J FERNANDEZ**

**ASST. PROF, CSE**

**RSET**

# HDLC(High Level Data link Control) Protocol

---

- **Bit – oriented** protocol: views the frame as a collection of bits.
- IBM developed SDLC (Synchronous Data Link Control) protocol as a bit oriented protocol.
- ISO standardized SDLC to HDLC.
- Basis for many other DLL protocols.
- Used for communication over point – to – point and multipoint links.
- Implements **Stop-and-Wait** protocol.
- Supports error and flow control.

# HDLC(High Level Data link Control) Protocol

---

- Types of stations:
  - Primary station: can send commands.
  - Secondary station: can only respond.
  - Combined: can both send and give response.

# HDLC(High Level Data link Control) Protocol

	Flag	Header	Body	CRC	Flag
bits	8	16		16	8

- **Flag:** synchronization pattern

⇒ 01111110

⇒ start and end of the frame.

⇒ also transmitted during any times that the link is idle so that sender and receiver can keep their clocks synchronized.

# HDLC(High Level Data link Control) Protocol

Flag	Header	Body	CRC	Flag
bits	8	16	16	8

- **Header:** address field and control field.
  - Address field: address of the secondary station.
  - Control field: to identify the types of HDLC frames; used for flow and error control.
- **Body:** Payload (varying size)
- **CRC:** Cyclic Redundancy Check: error detection

# HDLC Protocol- Types of frames

## I. I- Frames: Information Frame

❑ carry user data from the network layer.

❑ flow and error control information (piggybacking)

## 2. S-Frames: Supervisory Frame

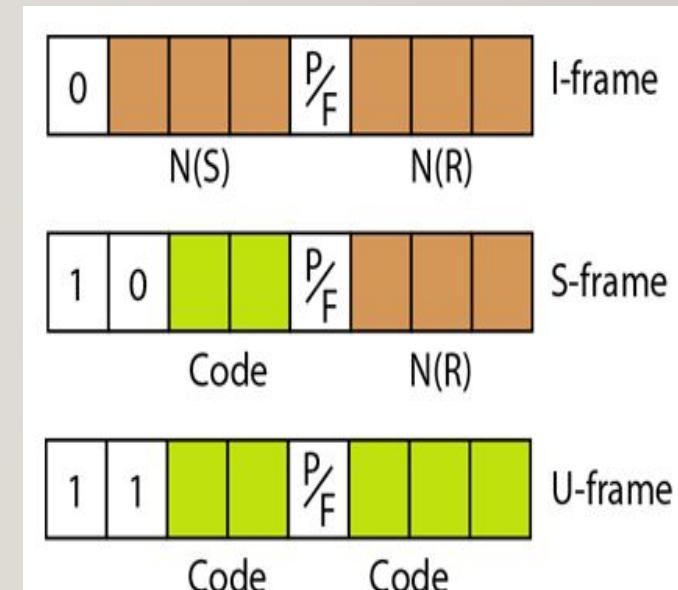
❑ flow and error control information (without piggybacking)

❑ do not have information fields.

## 3. U- Frames: Un-numbered Frame

❑ no user data

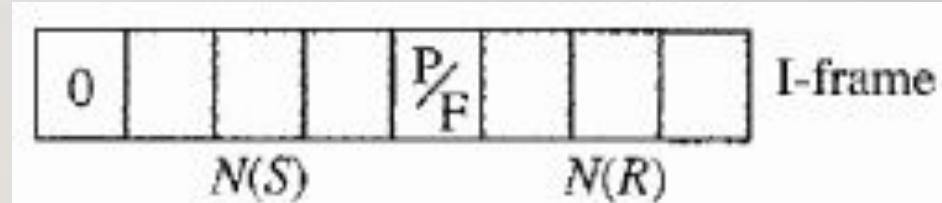
❑ system management information



# Control Field format for I- frames

---

- First bit – defines the **type** (0 for I-frame).
- Next 3 bits  $N(S)$  – **sequence number** of the frame.
- Last 3 bits  $N(R)$  – **acknowledgement number** when piggybacking is used.
- P/F bit – meaningful only when set to 1.

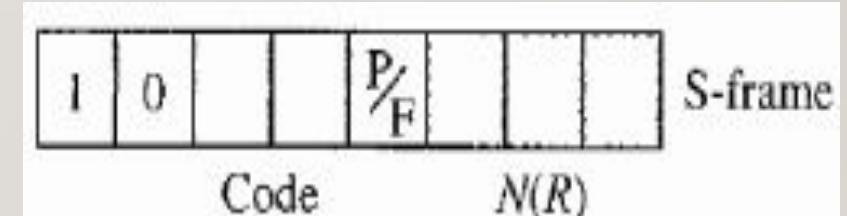


- Poll/ final
- Poll when the frame is sent by a primary station to a secondary station.
- Final when the frame is sent by a secondary station to a primary station.

# Control Field format for S- frames

---

- Used for flow and error control when piggybacking is impossible or inappropriate.
- First 2 bits **10** for S-frame.
- Last 3 bits  $N(R)$  - acknowledgment number or negative acknowledgment
- 3<sup>rd</sup> and 4<sup>th</sup> bits (code) – type of S-frame
  - 4 types of frames:

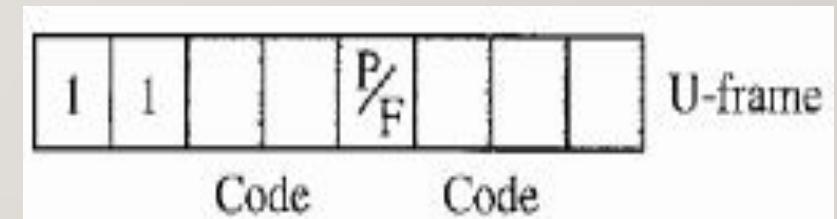


Code Bits	Type	Purpose
00	Receive Ready (RR)	<p>Acknowledges the receipt of a safe and sound frame or a group of frames</p> <p>N(R) field contains acknowledgment number</p>
10	Receive not Ready (RNR)	<p>Acknowledges the receipt of a safe and sound frame or a group of frames and announces that the receiver is busy and cannot receive more frames (acts as a congestion control mechanism)</p> <p>N(R) field contains acknowledgment number</p>
01	Reject (REJ)	<p>NAK frame used in Go-Back-N ARQ</p> <p>Informs the sender before the sender time expires that the last frame is lost or damaged</p> <p>N(R) field contains negative acknowledgment number</p>
11	Selective Reject (SREJ)	<p>NAK frame used in Selective Repeat ARQ</p> <p>N(R) field contains negative acknowledgment number</p>

# Control Field format for U- frames

---

- Used to exchange session management and control information between connected devices.
- Information field used for system management information.
- First 2 bits **11** for U-frame.
- U-frame codes divided into 2 sections
  - 2 bit prefix before the P/F bit
  - 3 bit prefix after the P/F bit
- 32 different types of U-frames.



# HDLC Protocol- Transfer modes

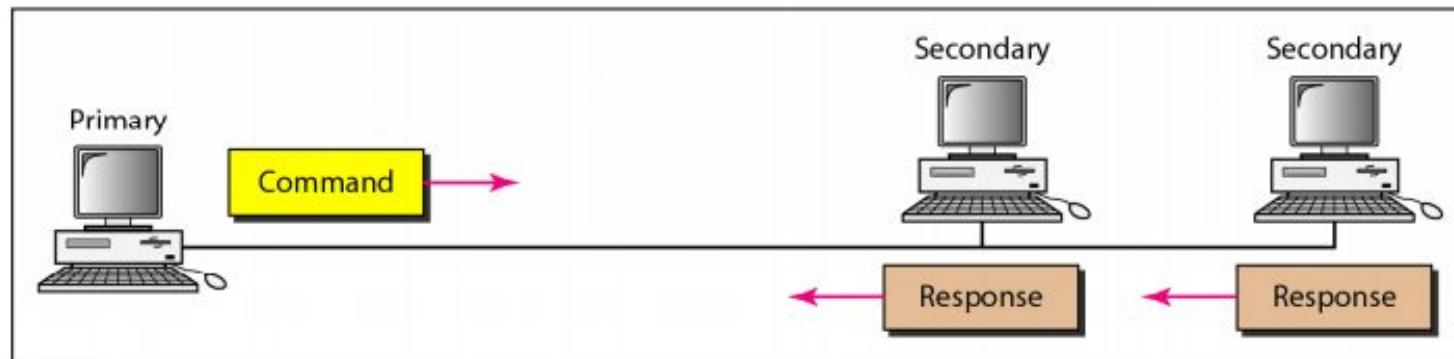
---

- Provides two transfer modes:
  - Normal Response Mode (NRM)
    - Unbalanced station configuration.
    - One primary station and multiple secondary stations.
    - Primary stations can send commands.
    - Secondary stations can only respond to commands.
    - Used for point-to-point and multipoint links.
  - Asynchronous Balanced Mode (ABM)

# NRM



a. Point-to-point

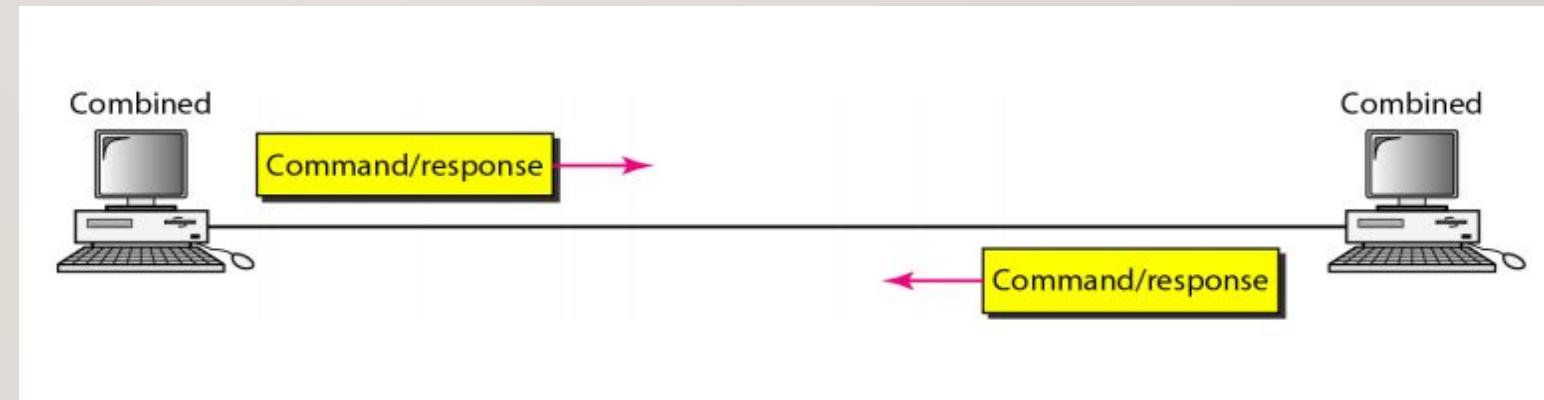


b. Multipoint

# HDLC Protocol- Transfer modes

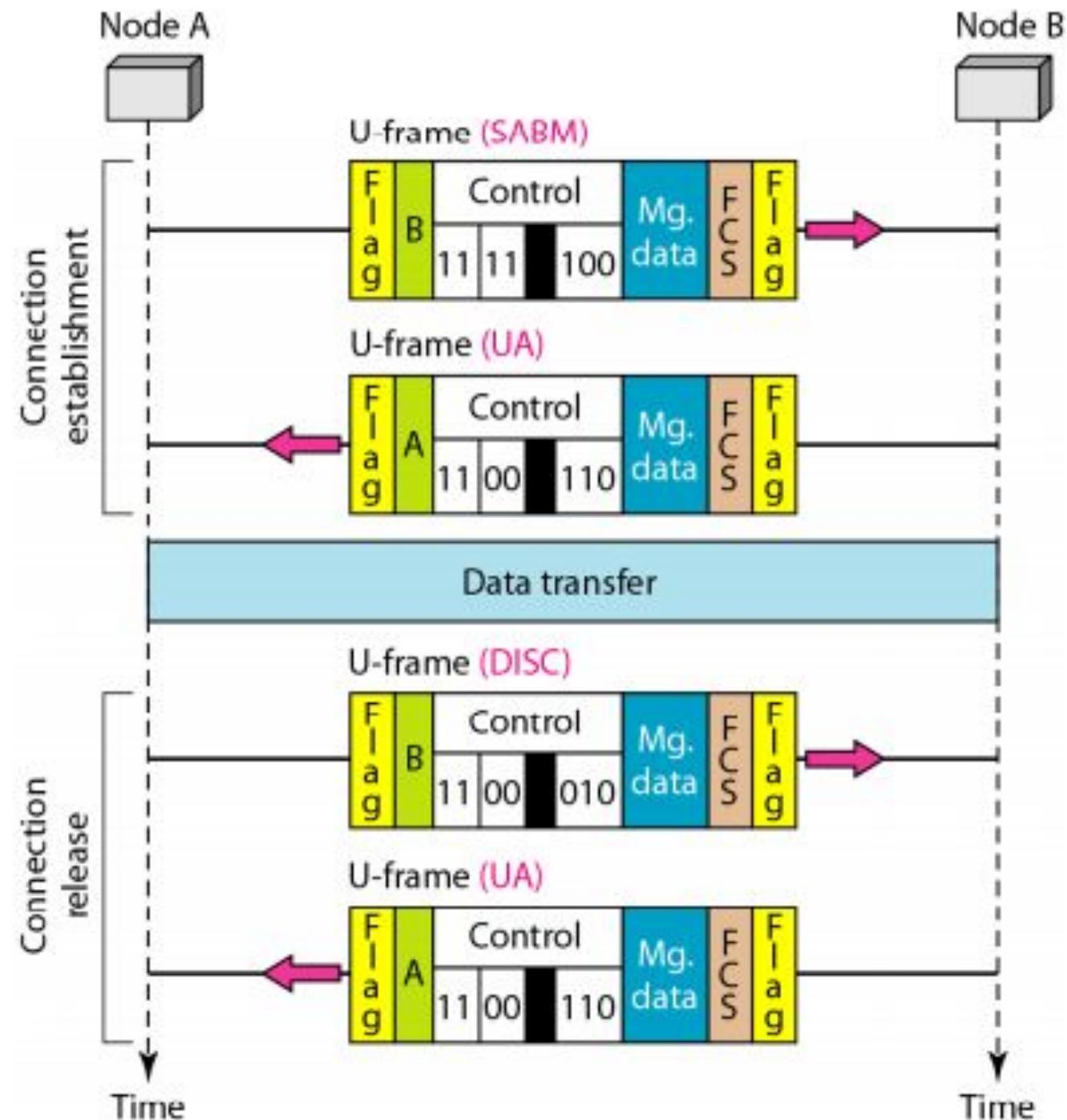
---

- Provides two transfer modes:
  - Asynchronous Balanced Mode (ABM)
    - Link is point-to-point in nature.
    - Each station can function as primary and secondary stations (peers).
    - Commonly used mode.

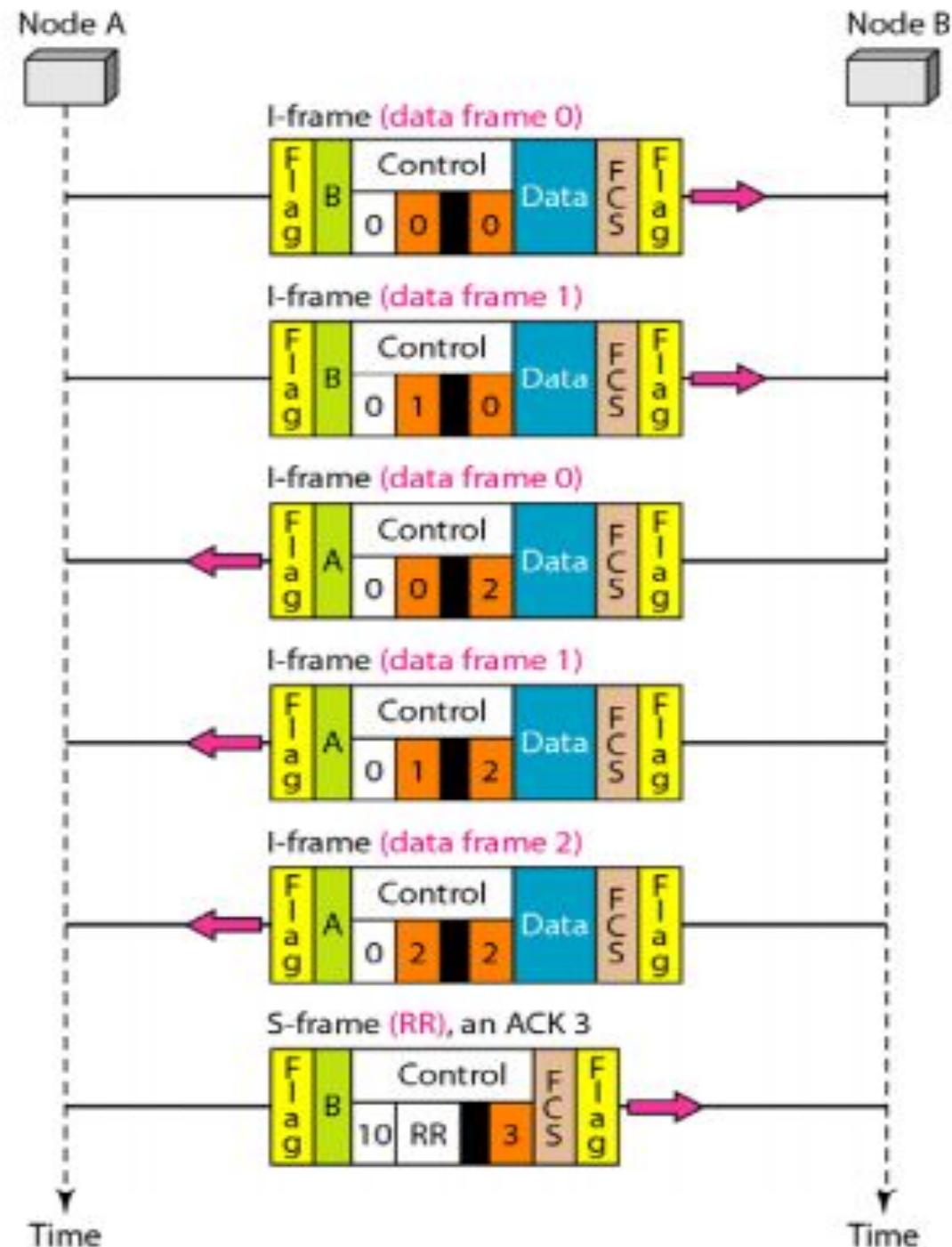


<i>Code</i>	<i>Command</i>	<i>Response</i>	<i>Meaning</i>
<b>00 001</b>	SNRM		Set normal response mode
<b>11 011</b>	SNRME		Set normal response mode, extended
<b>11 100</b>	SABM	<b>DM</b>	Set asynchronous balanced mode or <b>disconnect mode</b>
<b>11 110</b>	SABME		Set asynchronous balanced mode, extended
<b>00 000</b>	UI	<b>UI</b>	Unnumbered information
<b>00 110</b>		<b>UA</b>	<b>Unnumbered acknowledgment</b>
<b>00 010</b>	DISC	<b>RD</b>	Disconnect or <b>request disconnect</b>
<b>10 000</b>	SIM	<b>RIM</b>	Set initialization mode or <b>request information mode</b>
<b>00 100</b>	UP		Unnumbered poll
<b>11 001</b>	RSET		Reset
<b>11 101</b>	XID	<b>XID</b>	Exchange ID
<b>10 001</b>	FRMR	<b>FRMR</b>	Frame reject

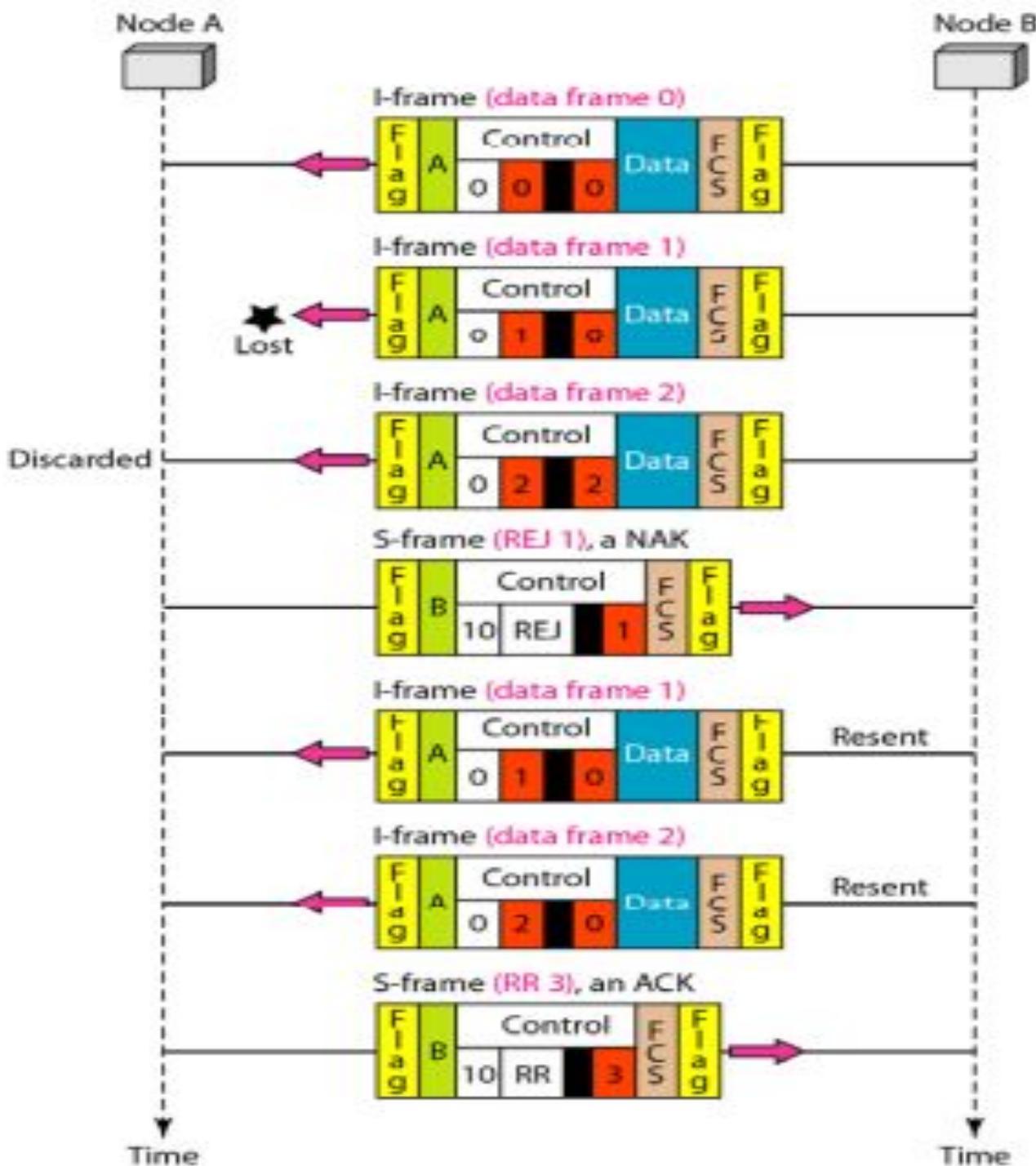
## EXAMPLE 1



## EXAMPLE 2



## EXAMPLE 3



# THANK YOU!!!

---

# **COMPUTER NETWORKS**

## **MODULE 2.4**

---

**MS. JINCY J FERNANDEZ**

**ASST. PROF, CSE**

**RSET**

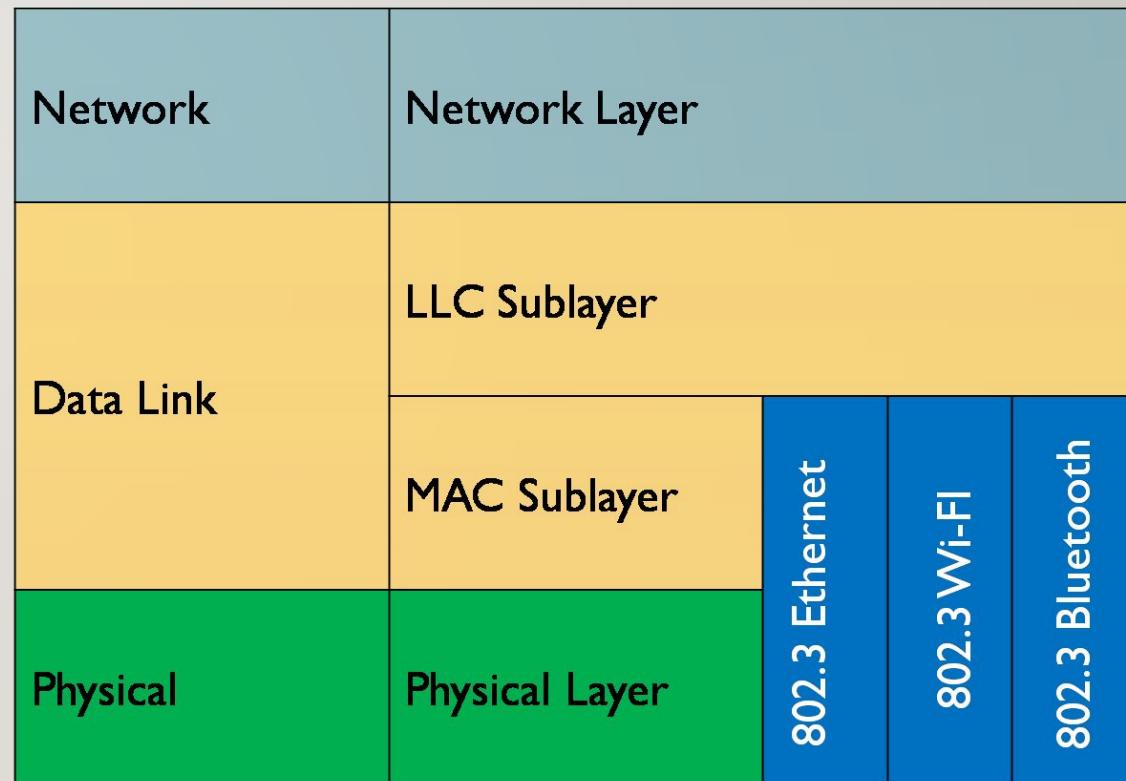
# Sublayers of Data link layer

- Logical Link Control sublayer (LLC)

- Takes data from the network layer
- Add control information to the data packet to ensure flow control.

- Medium Access Control sublayer (MAC)

- Lowest sublayer of Data Link layer
- Two responsibilities
  - Data encapsulation
  - Medium Access Control



# Sublayers of Data link layer

---

- Medium Access Control sublayer (MAC)
  - Data encapsulation
    - Frame assembly before transmission and frame disassembly upon reception of a frame.
    - Adds the header and trailer information to the packet to be transmitted to network layer
  - Medium Access Control
    - Handles access to a shared medium.

# Medium Access Control Sublayer

- 
- Network Links can be divided into:
    - Point-to-point connections
    - Broadcast channels(Multi-access channels or random-access channels)
  - In a broadcast network, the key issue is how to determine who gets to use the channel when there is competition.
  - The protocols used to determine who goes next on a multicast channel belongs to a sublayer of the DLL – **Medium Access Control sublayer**.
  - LANs use multi-access channels.

# Channel Allocation Problem

---

- How to allocate a single broadcast channel among competing users.
- Channel: portion of the wireless spectrum or a single wire or optical fiber to which multiple nodes are connected.
- Two types:
  - Static Channel Allocation
  - Dynamic Channel Allocation

# Static Channel Allocation

---

- Capacity of the channel is split among multiple competing users:
  - Frequency Division Multiplexing (FDM).
  - Time Division Multiplexing (TDM).
- FDM
  - With N users, bandwidth is divided into N equal sized portions, with each user being assigned one portion → no interference among users.
  - Simple and efficient if there is only small and constant number of users.
  - E. g. FM radio station.
- TDM
  - The time domain is divided into several recurrent *time slots* of fixed length, one for each sub-channel.

# Static Channel Allocation

---

- Problem with FDM and TDM if number of users is too large.
- If the spectrum is cut up into N regions:
  - Fewer than N users are currently interested in communicating then,
    - A part or large piece of valuable spectrum will be wasted.
  - More than N users want to communicate then,
    - Some of them will be denied permission due to lack of bandwidth.

# Dynamic Channel Allocation

---

- Channel allocation is done dynamically.
- Allocation is done **based on the current demands** of the users.
- Protocols that solve the channel allocation problem dynamically are called Multiple Access Protocols.

# Assumptions

---

- **Independent Traffic.** The model consists of  $N$  independent **stations/terminals**. Once a frame is generated, the station is blocked and does nothing until the frame has been successfully transmitted.
- **Single Channel Assumption.** A single channel is available for all communication. All stations can transmit on it, and all can receive from it.
- **Observable Collisions.** If two frames are transmitted simultaneously, they overlap in time and the resulting signal is garbled. This event is called a **collision**. All stations can detect collisions. A collided frame must be transmitted again later.

# Assumptions

---

- **Continuous or Slotted Time:**
  - **Continuous Time:** Frame transmission can begin at any instant. There is no master clock dividing time into discrete intervals.
  - **Slotted Time.** Time is divided into discrete intervals (slots). Frame transmissions always begin at the start of a slot. A slot may contain 0, 1, or more frames, corresponding to an idle slot, a successful transmission, or a collision, respectively.

# Assumptions

---

- **Carrier Sense or No Carrier Sense:**

- **Carrier Sense.** Stations can tell if the channel is in use before trying to use it. If the channel is sensed as busy, no station will attempt to use it until it goes idle.
- **No Carrier Sense.** Stations cannot sense the channel before trying to use it. They just go ahead and transmit. Only later can they determine whether the transmission was successful.

# Multiple Access Protocols

## • Contention Protocols

- Resolves collision after it occurs.
- Executes a collision resolution protocol after each collision.
  - ❖ ALOHA
  - ❖ Carrier Sense Multiple Access (CSMA)
  - ❖ Carrier Sense Multiple Access with Collision Detection (CSMA/CD)

## • Collision Free Protocols

- Ensures that a collision never occurs.
  - ❖ Bit –Map Protocol
  - ❖ Token passing
  - ❖ Binary Countdown

# Aloha

---

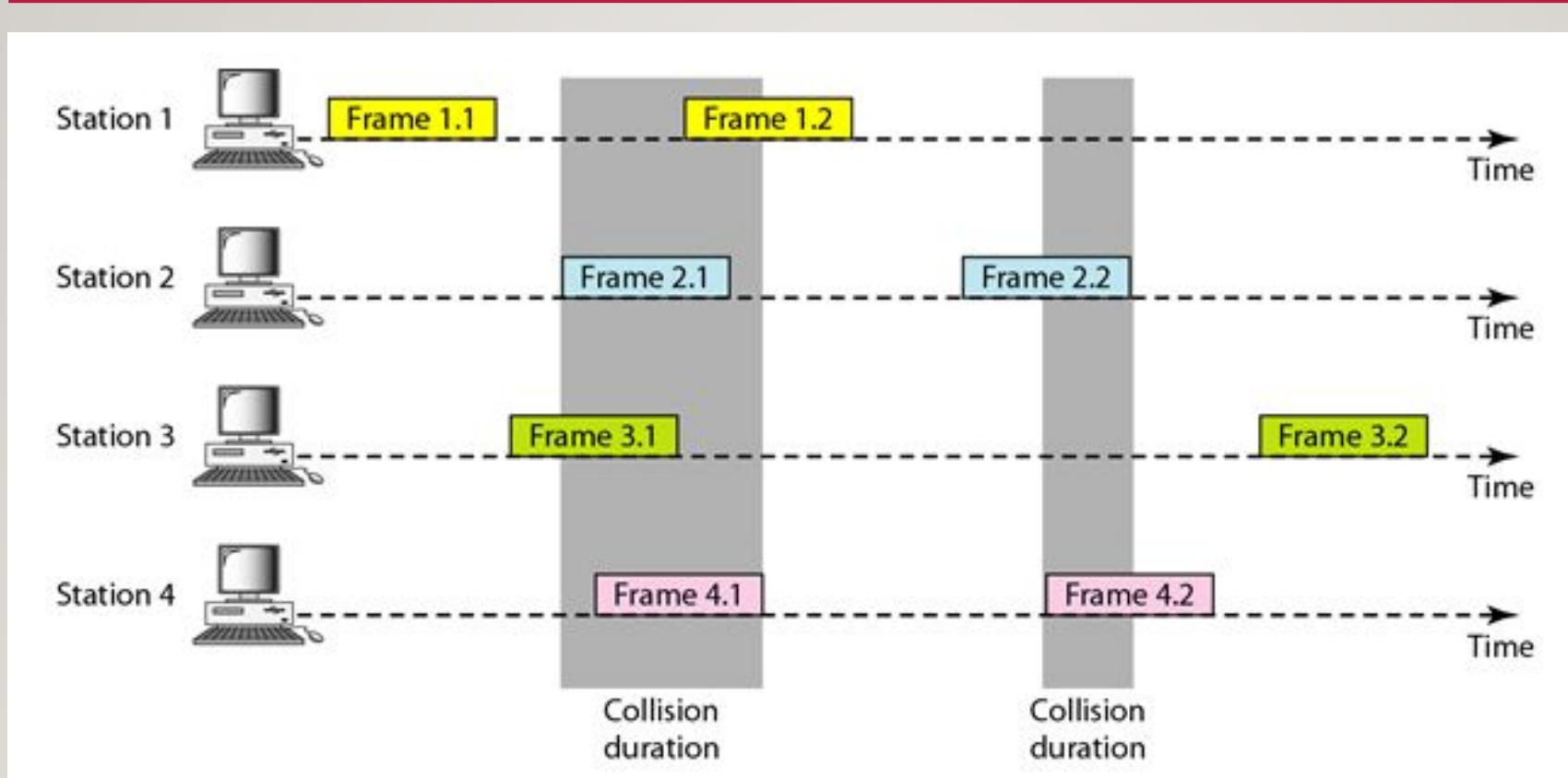
- Developed by Norman Abramson in 1970.
- Earliest random-access protocol : any station can send data at any time.
- It was designed for a radio (wireless) LAN, but it can be used on any shared medium.
- Two versions of the protocol:
  - Pure ALOHA
  - Slotted ALOHA

# Pure Aloha

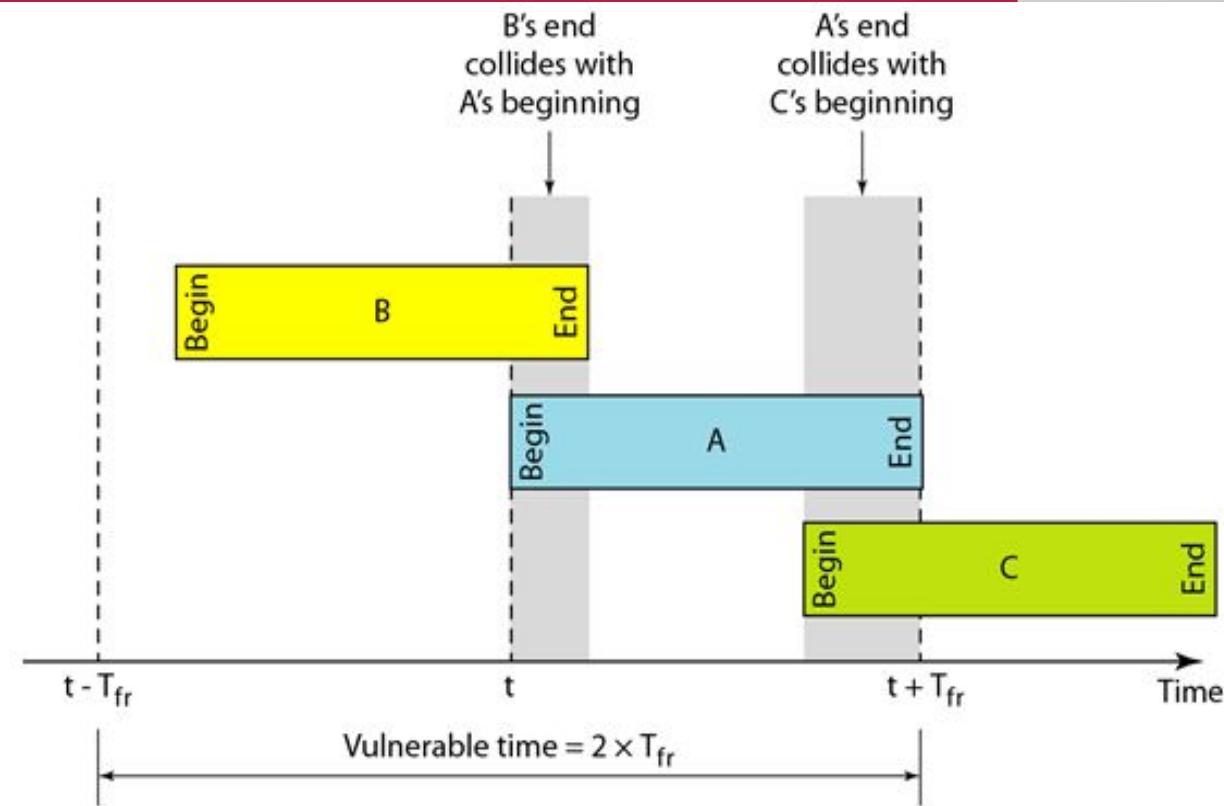
---

- Each user is free to transmit whenever they have data to be sent.
- There are possibilities of collision and colliding frames are destroyed.
- Sender finds whether transmission was successful or has experienced a collision by listening to the channel. (feedback system).
- If the frame is destroyed, **the sender waits a random amount of time and sends it again.**
- Waiting time must be random.
- Such systems are called contention systems.

# Example



# Vulnerable Time for a frame



# Maximum Throughput Of Pure Aloha

---

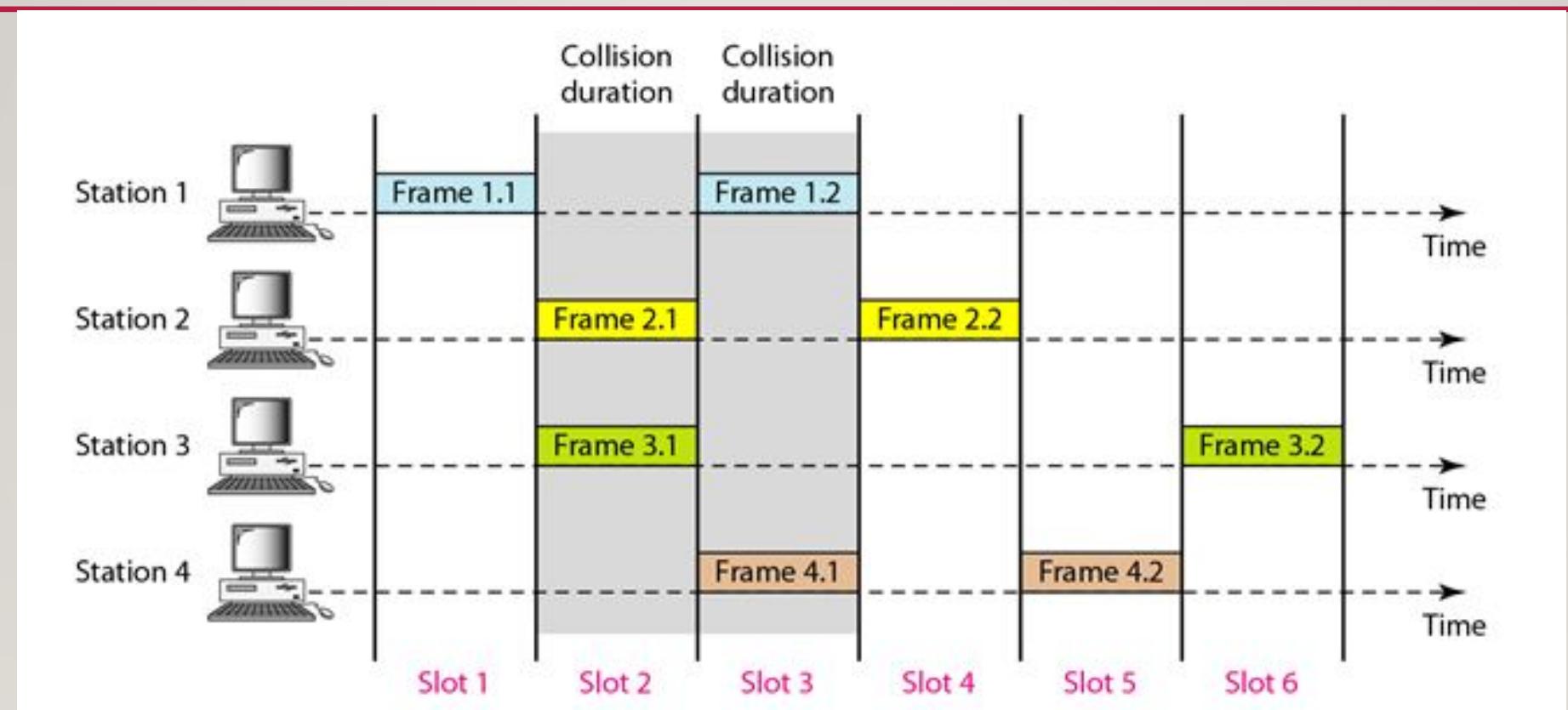
- Let  $G \rightarrow$  average number of frames generated during one frame transmission time.
- Throughput,  $S = G \times e^{-2G}$

# Slotted Aloha

---

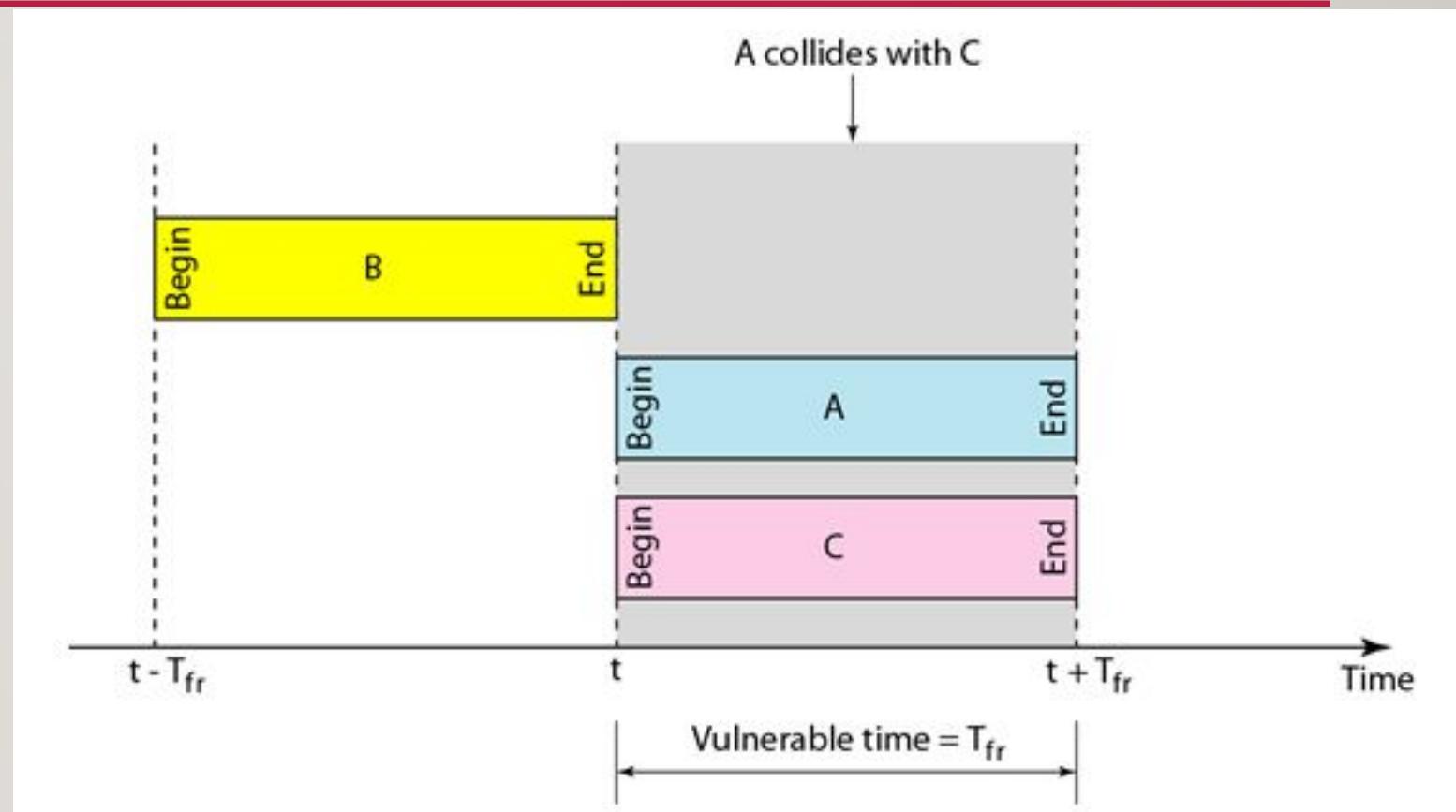
- Capacity of ALOHA doubled - Roberts in 1972.
- Time divided into discrete intervals called **slots**; each interval corresponds to one frame.
- Users agree on slot boundaries – synchronization needed using clocks.
- A station is not permitted to send whenever it is ready with the data.
- Have to wait for the beginning of the next slot.
- So, this is a discrete ALOHA, previous one was continuous ALOHA.
- Still the possibility of collision if two stations try to send at the beginning of the same slot.

# Example



# Maximum Throughput Of Slotted Aloha

- Throughput,  $S = G \times e^{-G}$

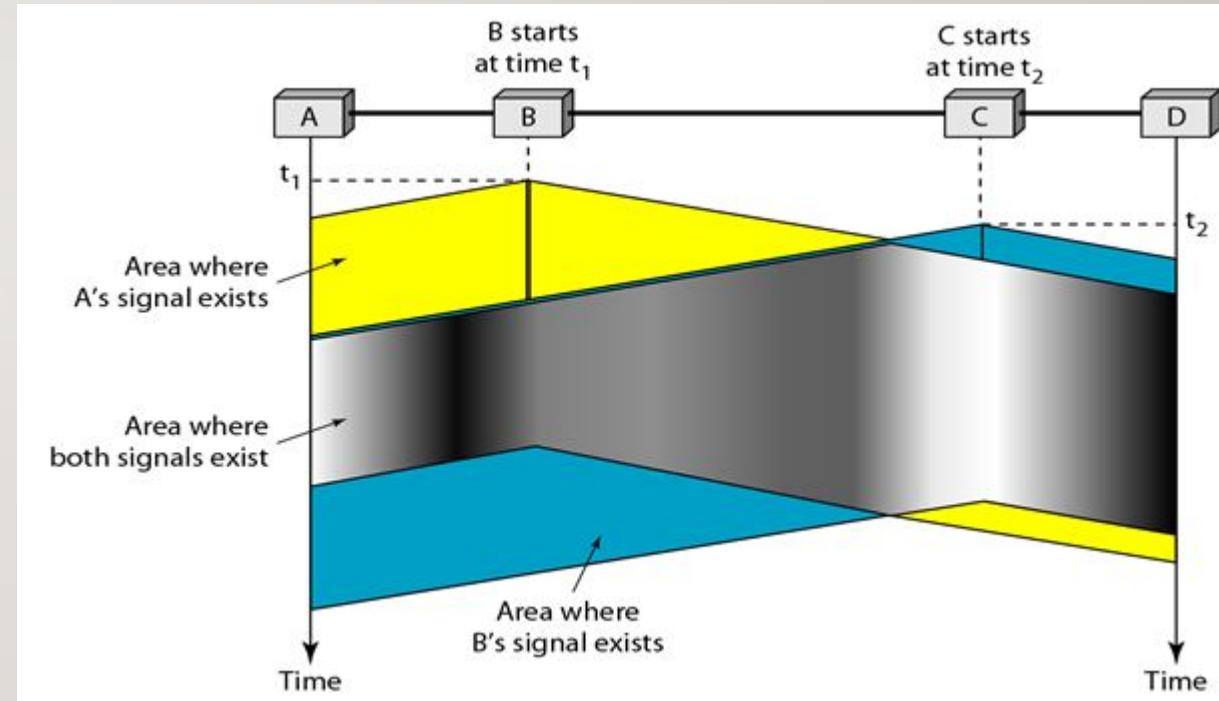


# Carrier Sense Multiple Access Protocols

- 
- Protocols in which stations listen for a carrier (transmission) and act accordingly are called **Carrier Sense Multiple Access (CSMA)** protocols.
  - Minimize the chance of collision and increase the performance.
  - CSMA is based on the principle “sense before transmit” or “listen before talk”.
  - 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 destination and for every station to sense it.

# Carrier Sense Multiple Access Protocols

- At time ‘t1’, station B senses the medium and finds it idle, so it sends frame.
- At time ‘t2’ ( $t_2 > t_1$ ), station C senses the medium and finds it idle because first bit of B is not reached, so C also sends frame.
- Two signals collided and both frames are destroyed.



# Carrier Sense Multiple Access Protocols

---

- Two versions:
  - Persistent and Non-persistent Protocols
    - What should a station do if the channel is busy?
    - What should a station do if the channel is idle?
      - 1-persistent CSMA
      - p-persistent CSMA
      - Non-persistent CSMA
  - CSMA with Collision Detection (CSMA/CD)

# 1-persistent CSMA

---

- When a station has data to send, it first listens to the channel.
- If the channel is busy, the station waits until it becomes idle. (continuously checking).
- If the channel is idle, **the stations immediately transmits data with a probability of 1.**
- If a collision occurs, station waits a random amount of time and starts all over again.
- Propagation delay is an issue here. Longer the delay worse the performance of the protocol.
- Better performance than ALOHA.

# Non-persistent CSMA

---

- A station senses the channel when it wants to send a frame. If idle, send the frame. If the channel is already in use, **the station does not continually sense it.**
- Instead, it **waits a random period of time and then repeats the algorithm.**
- This algorithm leads to better channel utilization but longer delays than 1-persistent CSMA.

# P-persistent CSMA

---

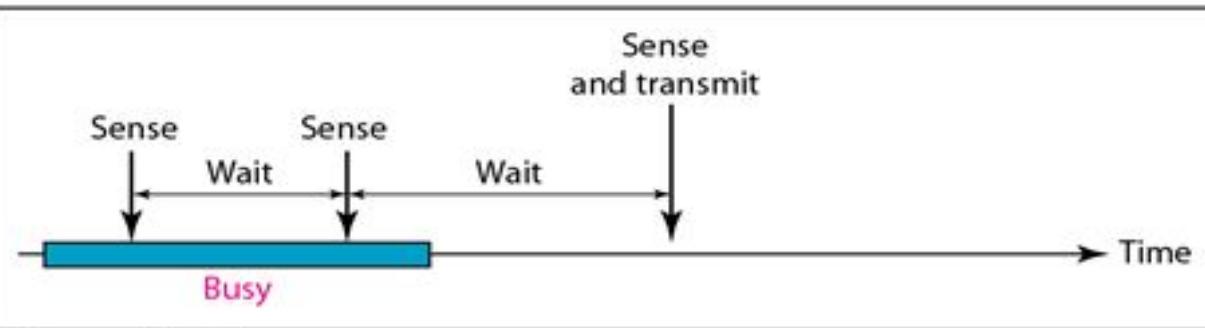
- It applies to slotted channels.
- When a station becomes ready to send, it senses the channel.
- If it is idle, **it transmits with a probability  $p$ . (*only in its available slot*).**
- It **defers with a probability  $q = 1 - p$**  until the next slot.
- If that slot is also idle, it either transmits or defers again, with probabilities  $p$  and  $q$ .
- This process is repeated.
- So, station has to check for channel and time slot.

# Example

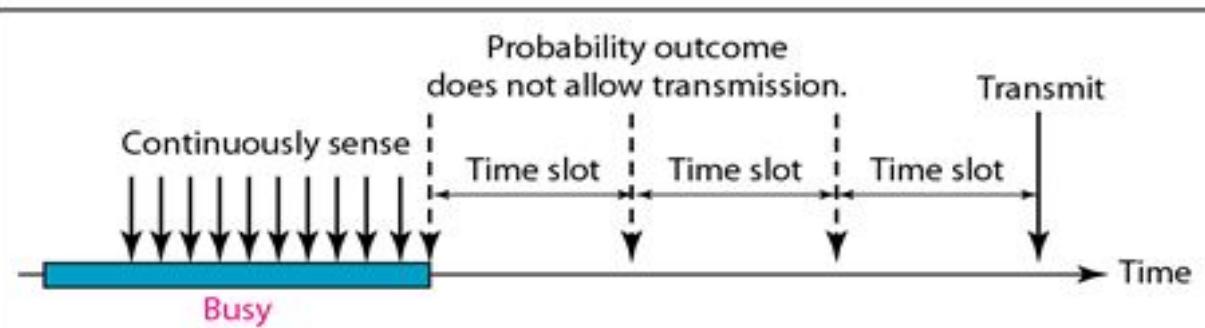
---



a. 1-persistent



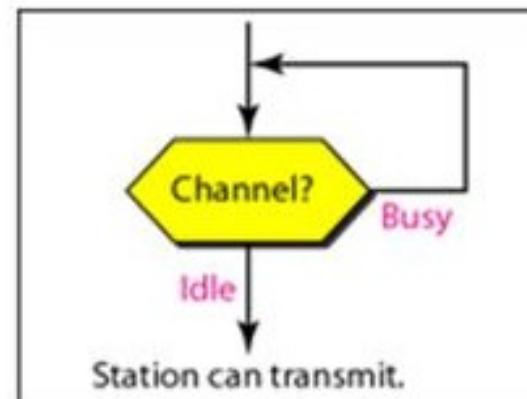
b. Nonpersistent



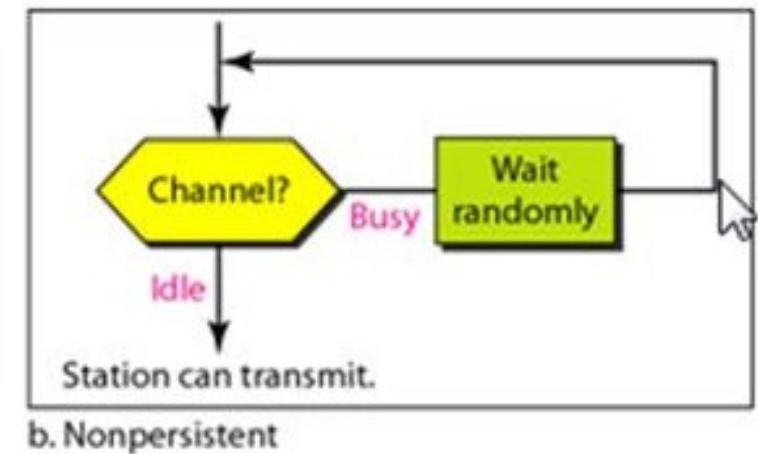
c. p-persistent

# Flow Diagram

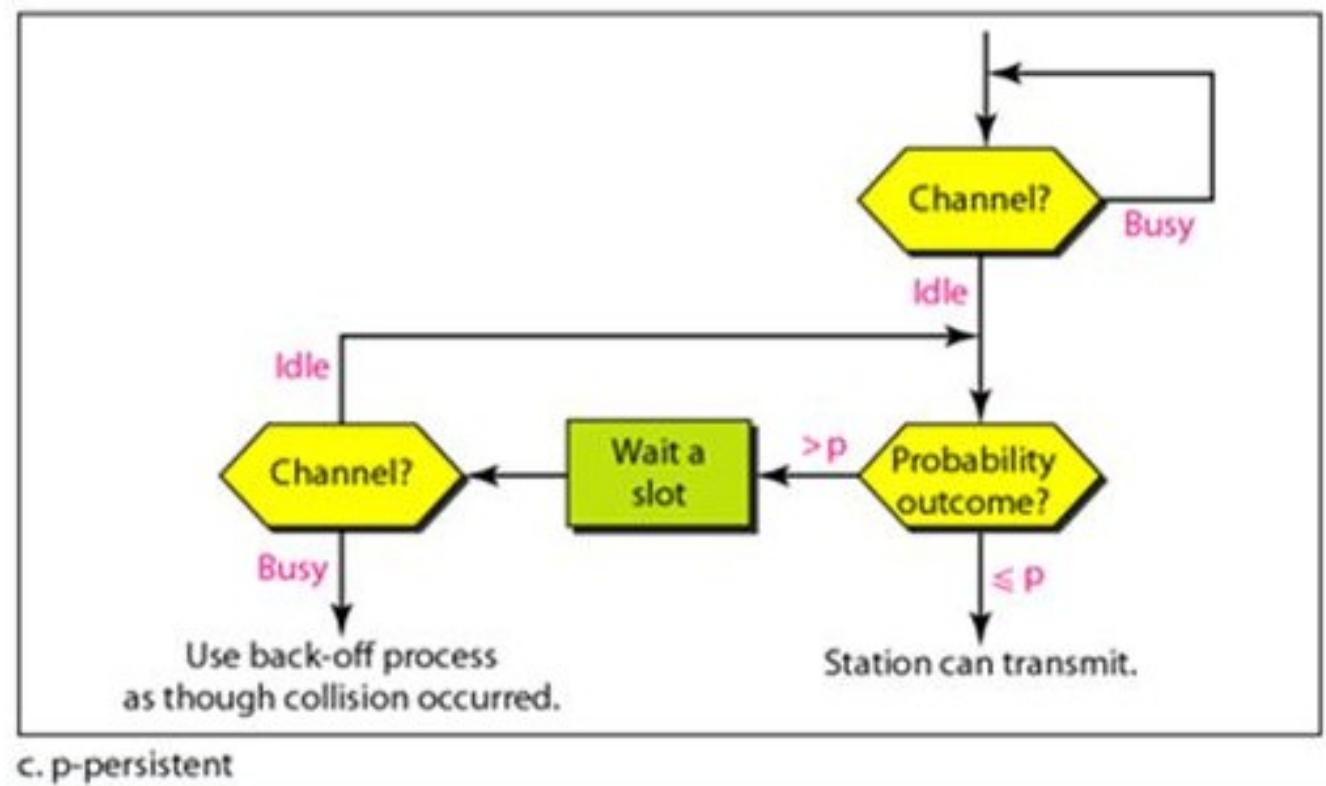
---



a. 1-persistent



b. Nonpersistent



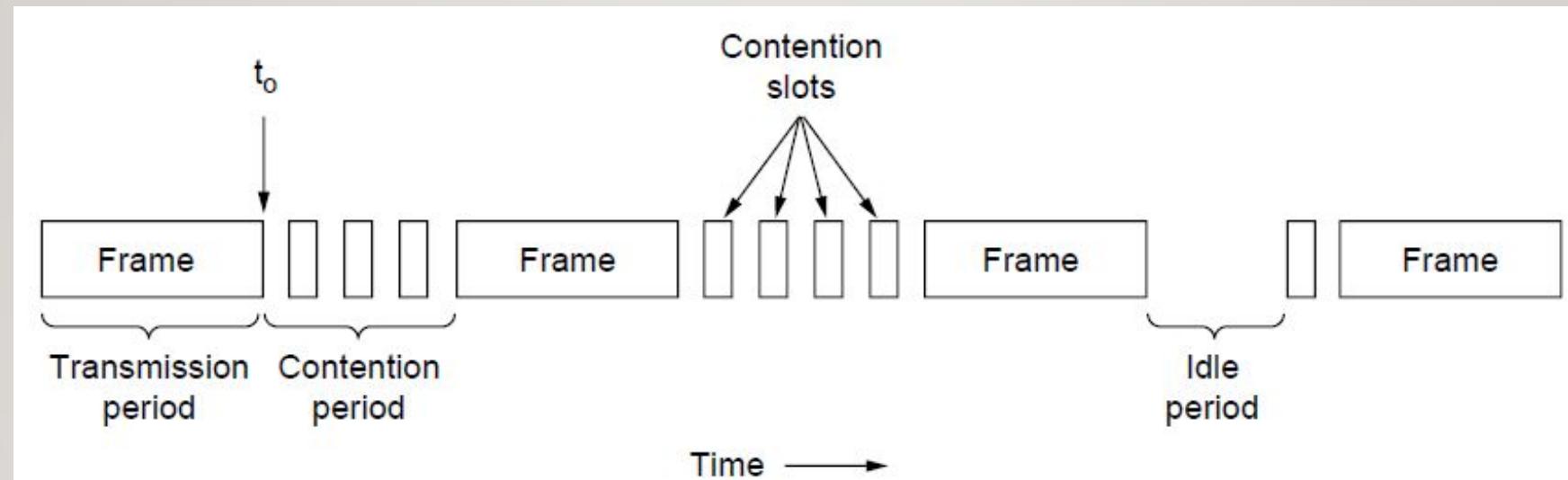
c. p-persistent

# CSMA With Collision Detection

---

- Protocols that abort transmissions as soon as they detect collisions? **CSMA with Collision Detection** (CSMA/CD).
- Quickly terminating damaged frames save time and bandwidth.
- This protocol is a basis of classical Ethernet LAN.
- Uses power or pulse width of the received signals to detect collision: power or pulse width of the transmitted is better than that of received signal.
- No ACK used; It checks for the successful and unsuccessful transmissions through collision signals.

# CSMA With Collision Detection



- The model for CSMA/CD consists of alternating contention and transmission periods, with idle periods occurring when all the stations are quiet.
- Contention period is the minimum time a host must wait to make sure that no other host is transmitting.

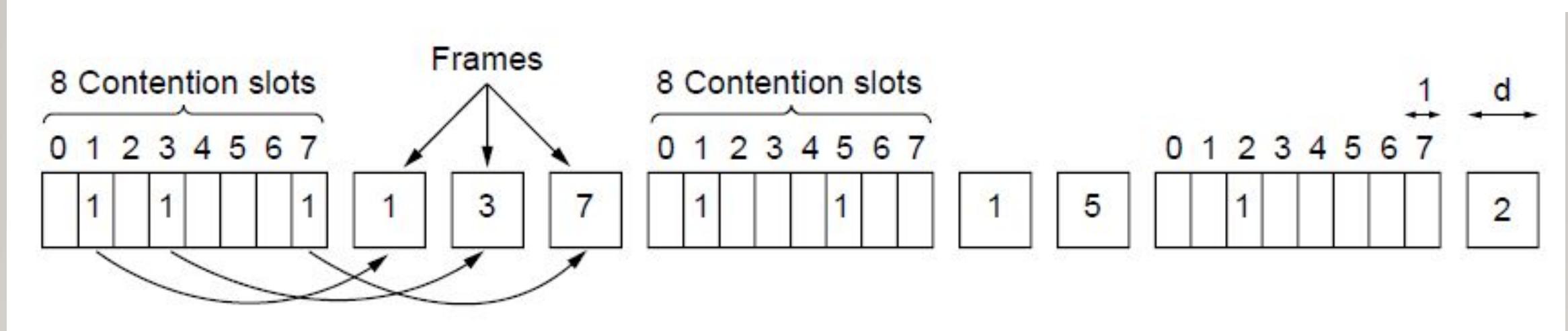
# Collision – Free Protocols

---

- Collisions may occur during contention period in CSMA/CD.
- Also, it is not universally applicable.
- So go for protocols that ensure collisions don't occur at all.
- Assumptions:
  - N – Stations each with a unique address 0- (N-1).
  - Propagation delay is assumed to be negligible.
  - Some stations may be inactive throughout.
- Basic question-Which station gets the channel (e.g., the right to transmit) after a successful transmission?

# Bit-map Protocol

- 
- Called reservation protocol.
  - Each contention period (reservation frame) consists of N slots.
  - If station 0 has a frame to send, it transmits a 1 bit during the zeroth slot.
  - No other station transmits during this slot.
  - **Station j may announce that it has a frame to send by inserting a 1 bit into slot j.**
  - After all N slots have passed by, each station has complete knowledge of which stations wish to transmit.
  - The stations then begin transmitting frames in numerical order.



### The basic bit-map protocol

Since everyone agrees on who goes next, there will never be any collisions.

# Token Passing

- A small message called **token** is passed from one station to the next in a predefined order.
- Token represents permission to send.
- If a station has a frame queued for transmission when it gets the token, it can send that frame before it passes the token to the next station.
- If it has no queued frame, it simply passes the token..  
*After sending a frame a station must wait for all N stations to pass the token to its neighbor and all N-1 stations to send a frame, if they have one, before it gets the next chance.*

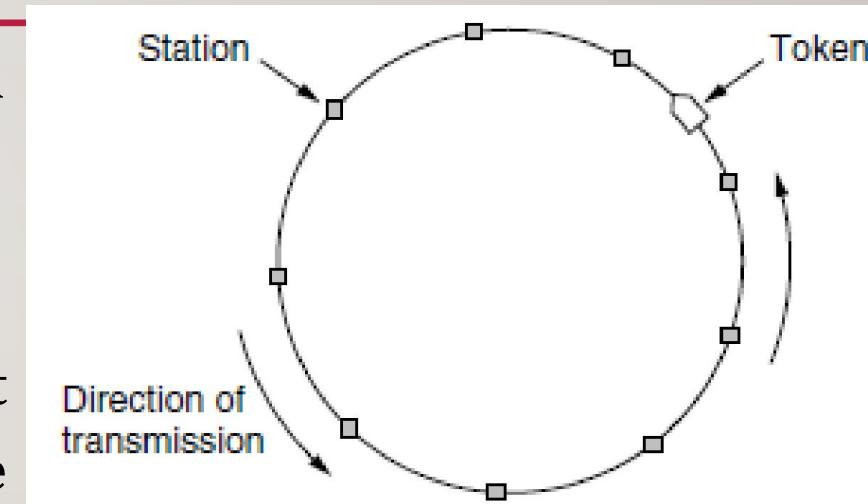


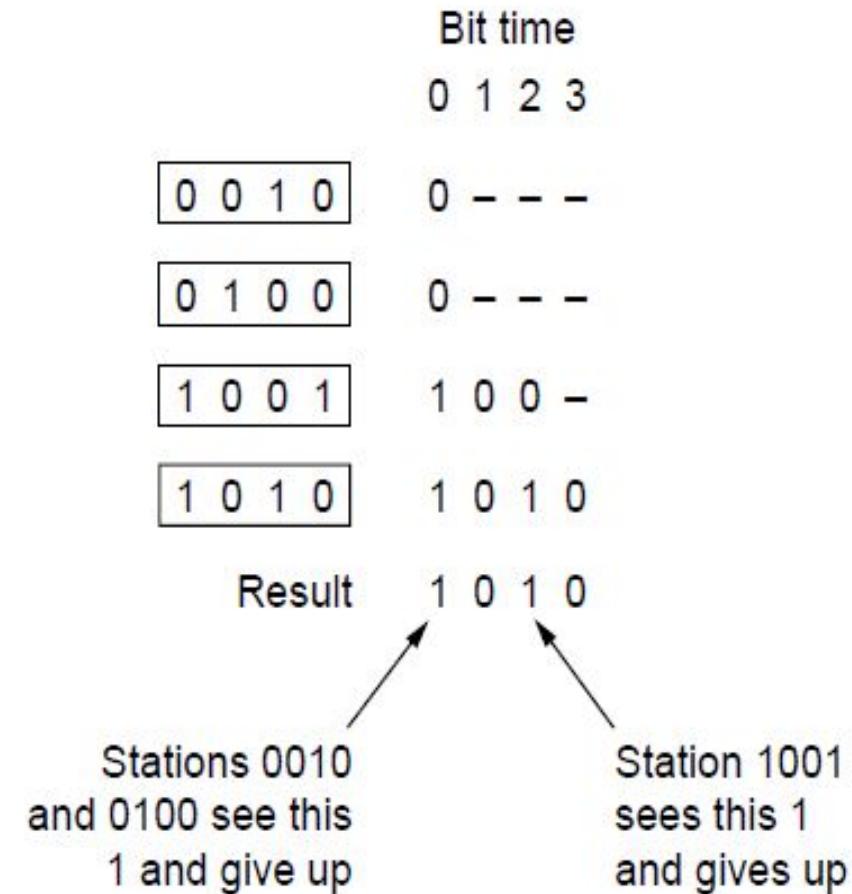
Figure 4-7. Token ring.

# Binary Countdown Protocol

- Issue with bitmap protocol is its overhead. Does not scale well for large networks.
- Issue with token passing is starvation.
- Better solution- go for binary station addresses and priorities.
- A station that wants to use the channel broadcasts its address (**assumed to be the same length**) as a binary bit string starting with the high-order bit.
- The bits in each address position from different stations are BOOLEAN OR-ed together by the channel when they are sent at the same time.
- **Arbitration rule:** As soon as a station sees that a high-ordered bit position that is 0 in its address has been overwritten with 1 it gives up.

# Binary Countdown Protocol

- If stations 0010,0100,1001,1010 are trying to get the channel, in the first bit time, stations transmits 0, 0, 1, 1. Boolean OR results in 1.
- So, stations 0010, 0100 gives up and wait.
- Stations 1001 and 1010 continue and check next bit and so on.
- It has the property that higher numbered stations have a higher priority.



# Performance Measures

---

- Two main measures:
  - Delay
  - Channel efficiency
- Under conditions of light load:
  - **Contention protocols** preferable – low delay only.
- Under conditions of high load:
  - **Collision free protocols** preferable – better channel efficiency.
- Combine these two protocols into a hybrid one- **limited contention protocols**.

# Wireless LAN Protocols

---

- Each radio transmitter has some fixed range.
- Its range is represented by an ideal circular coverage region.
- Within that region station can sense and receive the station's transmission.
- Using CSMA for channel access is possible.
- Carrier Sense with Collision Avoidance
- Listen for other transmissions and transmit only if no one else is doing so.

**THANK YOU!!!**

---

# Pure Aloha

---

- Q1. A pure ALOHA network transmits 200-bit frames on a shared channel of 200Kbps. What is the throughput if the system produces 1000 frames per second?
  - Frame transmission time = $200/200\text{Kbps} = 1\text{ms}$
  - 1000 frames per second= 1 frame per milliseconds, G=1.
  - S=

Ans: 135

# **COMPUTER NETWORKS**

## **MODULE 3.I**

---

**MS. JINCY J FERNANDEZ**

**ASST. PROF, CSE**

**RSET**

# Network Layer- Introduction

---

- Provides services to the transport layer and receives services from the data link layer.
- Get packets from the source all the way to the destination.
- This may require making many hops at intermediate routers along the way.

# Network Layer- Introduction

---

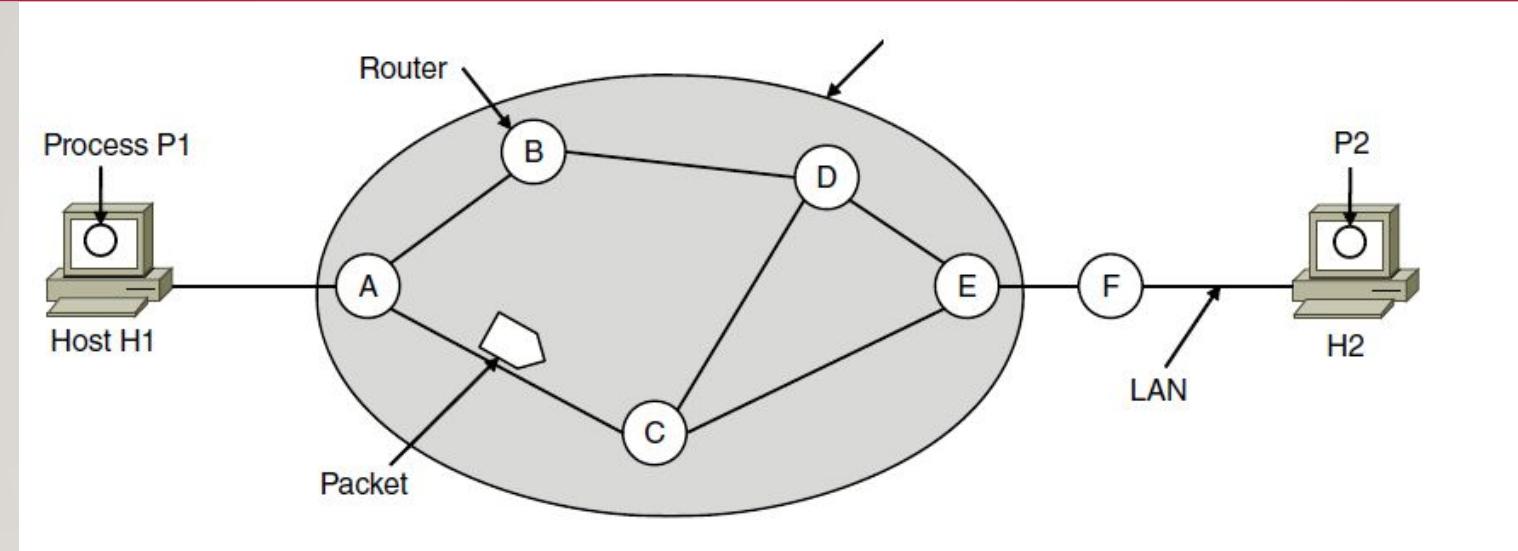
- Duties of Network layer:
  - Awareness about topology of the network and choose appropriate paths through it.
  - Choose routes to avoid overloading some of the communication lines and routers while leaving others idle.
  - Deal with cases of source and destination on different networks.

# Network Layer- Design Issues

---

- Store-and-forward packet switching
- Services provided to transport layer
- Implementation of connectionless service
- Implementation of connection-oriented service

# Store-and-forward Packet Switching



- A host with a packet to send transmits it to the nearest router.
- The packet is **stored in the router** until it has fully arrived, checksum is verified.
- Then the packet is **forwarded to the next router along the path** until it reaches the destination host.

# Services Provided To Transport Layer

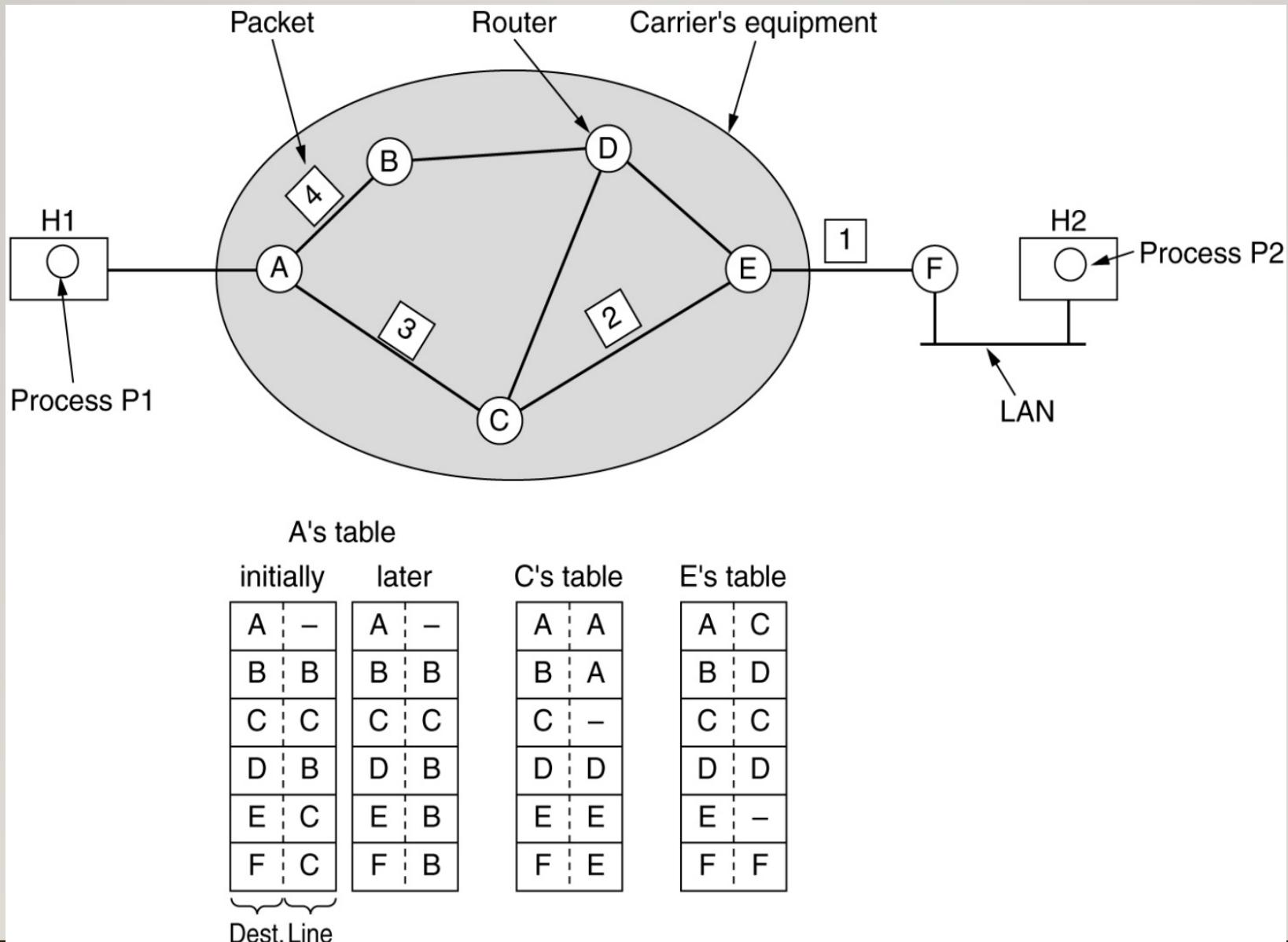
---

- Services should be designed to achieve the following goals:
  - Services independent of router technology.
  - Transport layer should be shielded from number, type, topology of routers.
  - Network addresses available to transport layer should use a uniform numbering plan.
- Connectionless Service (**Internet**)
- Connection – oriented Service (**ATM Networks**)

# Implementation Of Connectionless Service

---

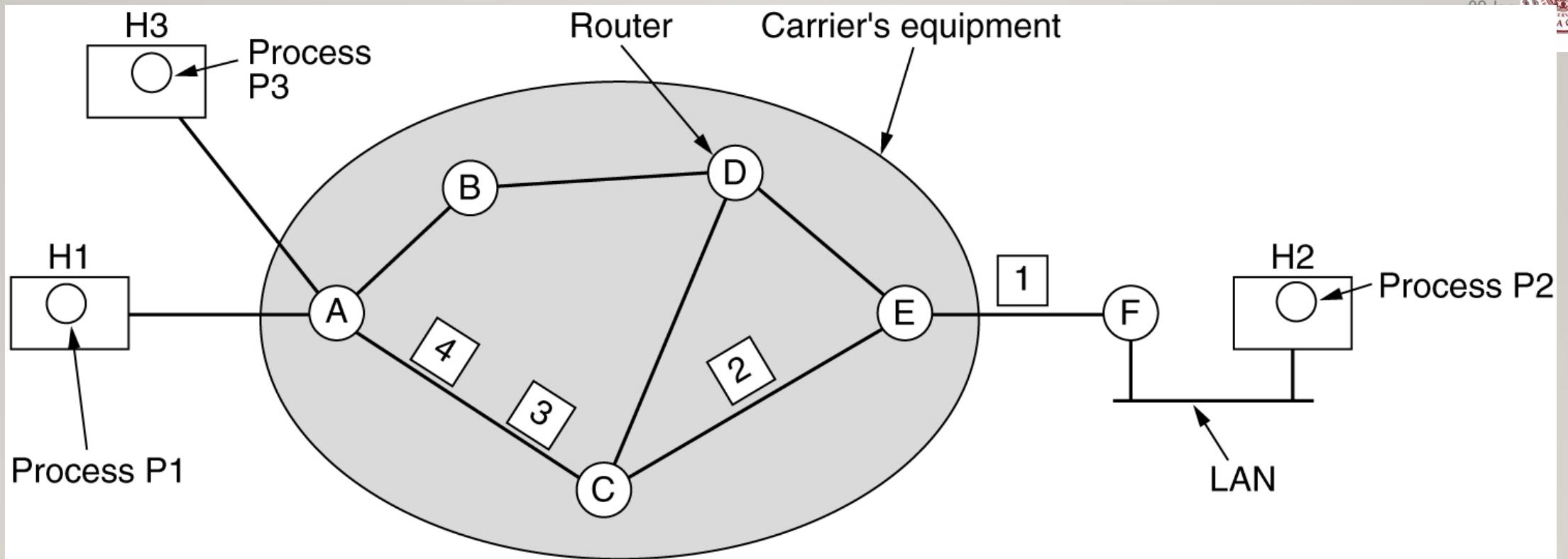
- Packets are injected into the network individually and **routed independently** of each other.
  - No advance setup is required.
- Packets are called **datagrams** and the network is called a **datagram network**.
- H1 sends a message to H2.
- Message broken down into 4 packets.
- Packets are routed based on the entry in the routing table.
  - **Routing Algorithm** is the algorithm that **manages the tables and makes the routing decisions**.



# Implementation of Connection-oriented Service

---

- Before sending a message, a connection is established (virtual circuit).
- A route from the source machine to the destination machine is chosen as part of the connection setup and stored in tables inside the routers.
- This route is used for all traffic flowing over the connection.
- When the connection is released, the virtual circuit is also terminated.
- Each packet will carry an identifier telling which virtual circuit it belongs to.
- Subnet is called a virtual circuit subnet.
- **Label switching:** Routers replacing connection identifiers in outgoing packets.



If a packet with ID 1 comes in from H1, it is to be given to router C and given the ID 1

A's table

H1   1	C   1
H3   1	C   2

In      Out

C's table

A   1	E   1
A   2	E   2

E's table

C   1	F   1
C   2	F   2

# Comparison Of Datagram Network And Virtual Circuit

---

<b>Issue</b>	<b>Datagram subnet</b>	<b>Virtual-circuit subnet</b>
Circuit setup	Not needed	Required
Addressing	Each packet contains the full source and destination address	Each packet contains a short VC number
State information	Routers do not hold state information about connections	Each VC requires router table space per connection
Routing	Each packet is routed independently	Route chosen when VC is set up; all packets follow it
Effect of router failures	None, except for packets lost during the crash	All VCs that passed through the failed router are terminated
Quality of service	Difficult	Easy if enough resources can be allocated in advance for each VC
Congestion control	Difficult	Easy if enough resources can be allocated in advance for each VC

# Routing Algorithms

- **Routing algorithms:** part of the network layer software responsible for deciding which output line an incoming packet should be transmitted on.
- Construction and maintenance of routing table is done by the **routing algorithm**.
- **Forwarding** - Decides to which output line an incoming packet should be transmitted.
- **Routing-** Decision is made based on the routing table.
  - If **datagrams** are used, the **decision is made anew for every arriving data packet**.
  - If **virtual circuits** are used, the **decision is made when the new virtual circuit is set up – all the data packets follow the established route – called session routing**.

# Properties Desirable in a Routing Algorithm

---

- Correctness
- Simplicity
- Robustness: able to cope with changes in the topology and traffic without requiring all jobs in all hosts to be aborted.
- Stability
- Fairness
- Efficiency: improve the overall network throughput.

# Routing Algorithms

---

- **Non-adaptive Routing Algorithms**

- **Static:** Current topology and traffic not considered for making routing decisions.
- The choice of the route is computed in advance, offline, and downloaded to the routers when the network is booted.

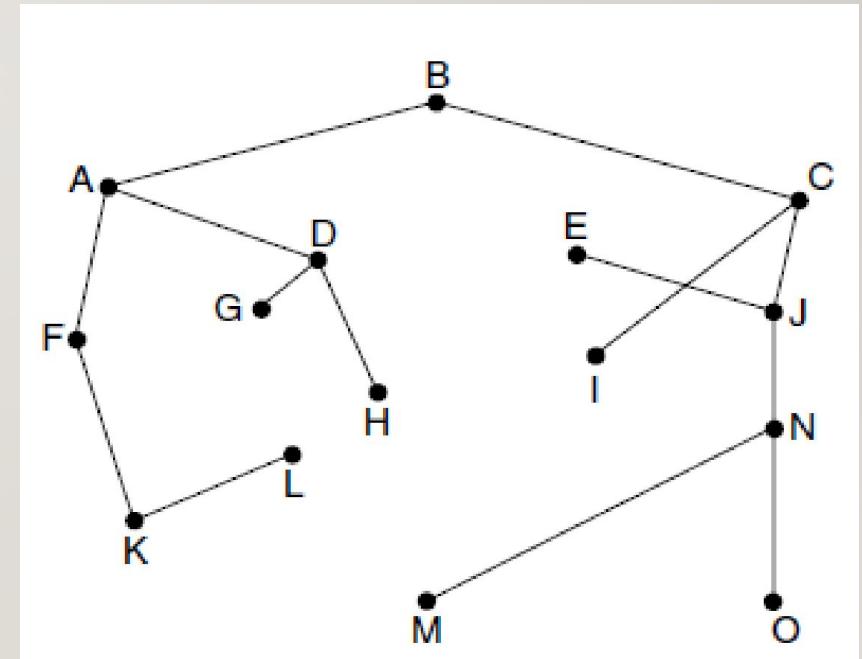
- **Adaptive Routing Algorithms**

- **Dynamic:** Routing decisions changes to reflect changes in the topology and in the traffic.
- Differ in where they get their information, when they change the routes and what metric is used for optimization.

# Optimality Principle

---

- It states that if router  $J$  is on the optimal path from router  $I$  to router  $K$ , then the optimal path from  $J$  to  $K$  also falls along the same route.
- Set of optimal routes from all sources to a given destination form a tree rooted at the destination.
- Such a tree is called a **sink tree**.
- Here distance metric is the number of hops.



# Shortest Path Algorithm

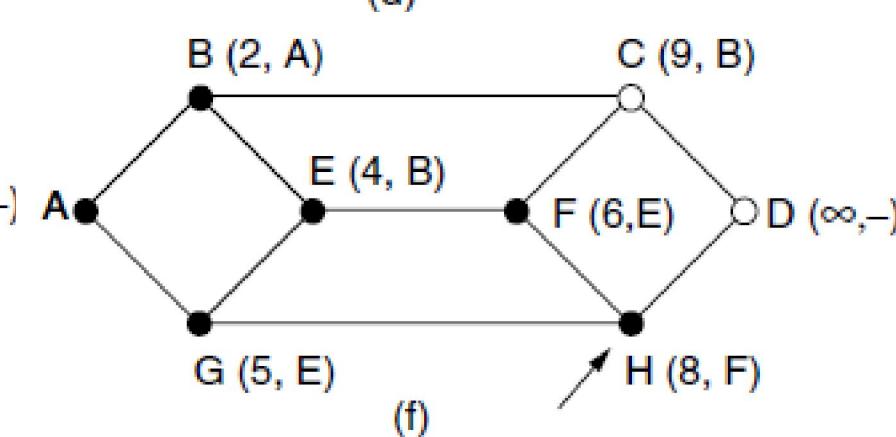
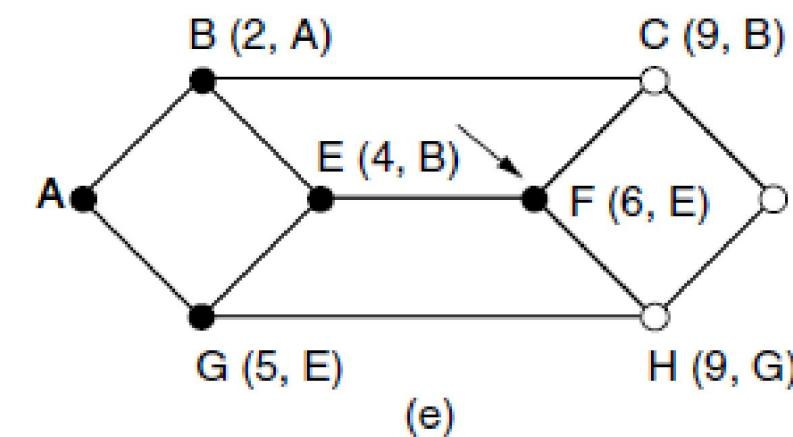
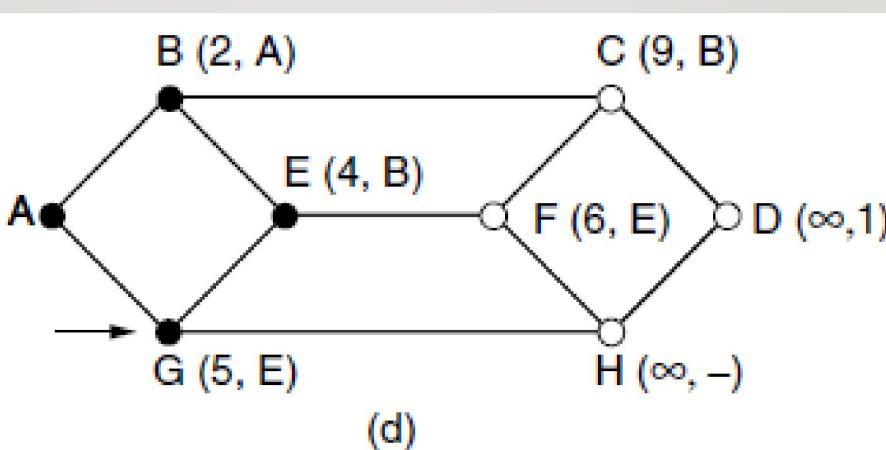
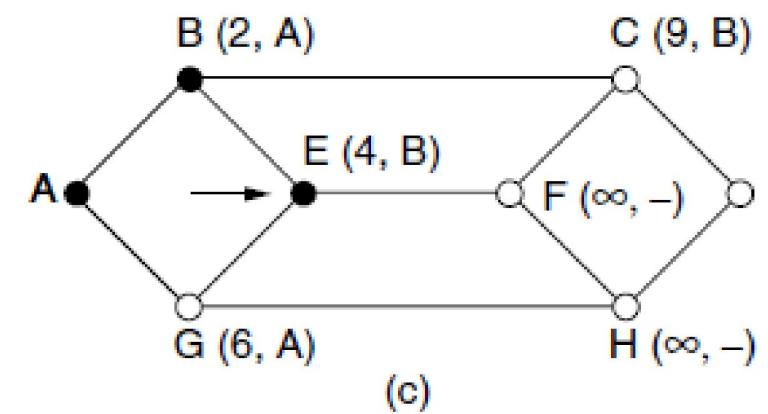
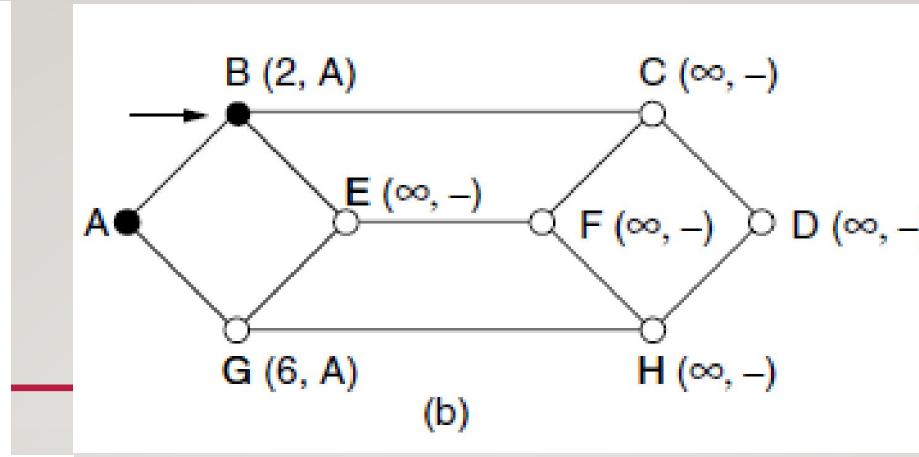
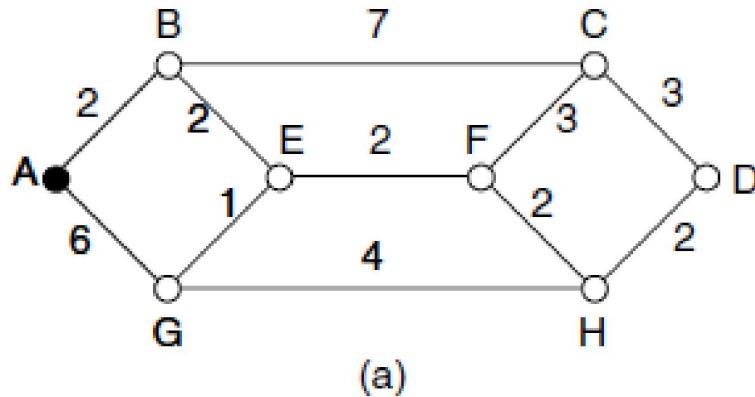
---

- Simple and easy to understand.
- Network represented as a graph.
- Nodes represent routers.
- Edge represents a communication link.
- Given a pair of routers, the algorithm finds the shortest path between the routers.
- The metric for the shortest path can **number of hops, distance, mean delay, average bandwidth, communication cost** etc.

# Shortest Path Algorithm- Dijkstra's Algorithm

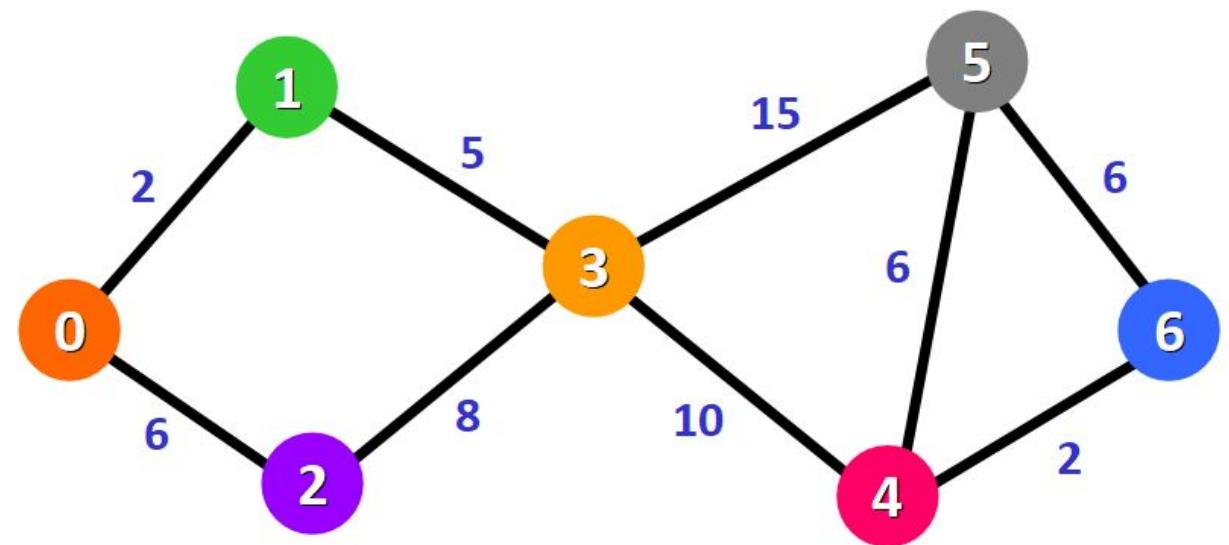
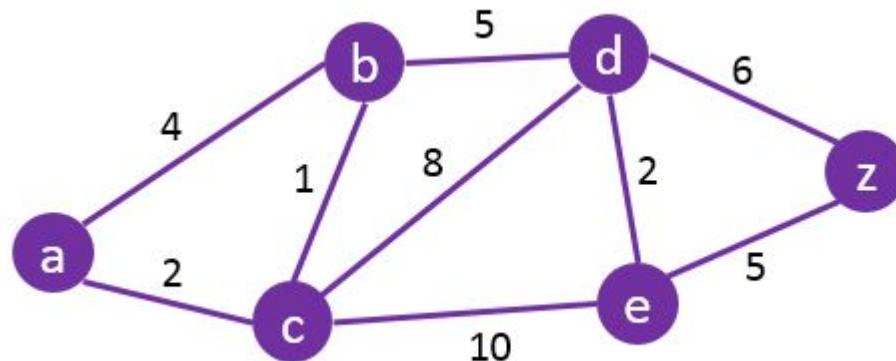
---

- Dijkstra's algorithm used – finds the shortest path between a source and all destinations in the network.
- Initially all the nodes are labelled with infinity and all the labels are tentative. Source node is set.
- As the algorithm proceeds, the nodes are labelled with its distance from the source along the best known path.
- When the label represents the shortest possible path from the source, it is made permanent.



- Make node A permanent
- Relabel the nodes adjacent to A with the distance to A
- Examine all the tentatively labeled nodes and make the one with the smallest label permanent

# Questions..



# Algorithm

---

- An array of distances **dist[ ]** of size  $|V|$  (number of nodes), where **dist [s] = 0** and **dist[u] =  $\infty$**  (infinity), where 's' represents the source vertex and 'u' represents a node in the graph except s.
- An array, **Q**, containing all nodes in the graph. When the algorithm runs into completion, **Q** will become empty.
- An initially empty set, **S**, to add the visited node. When the algorithm runs into completion, **S** will contain all the nodes in the graph.

# Algorithm

---

- Repeat while **Q** is not empty –
  - Remove from **Q**, the node, ‘**u**’ having the smallest **dist[u]** and which is not in **S**. In the first run, **dist[s]** is removed.
  - Add ‘**u**’ to **S**, marking **u** as visited.
  - For each node ‘**v**’ which is adjacent to **u**, update **dist[v]** as –
    - If (**dist[u]** + weight of edge **u-v**) < **dist[v]**, Then
      - Update **dist[v] = dist[u] + weight of edge u-v**
  - The array **dist[ ]** contains the shortest path from **s** to every other node.

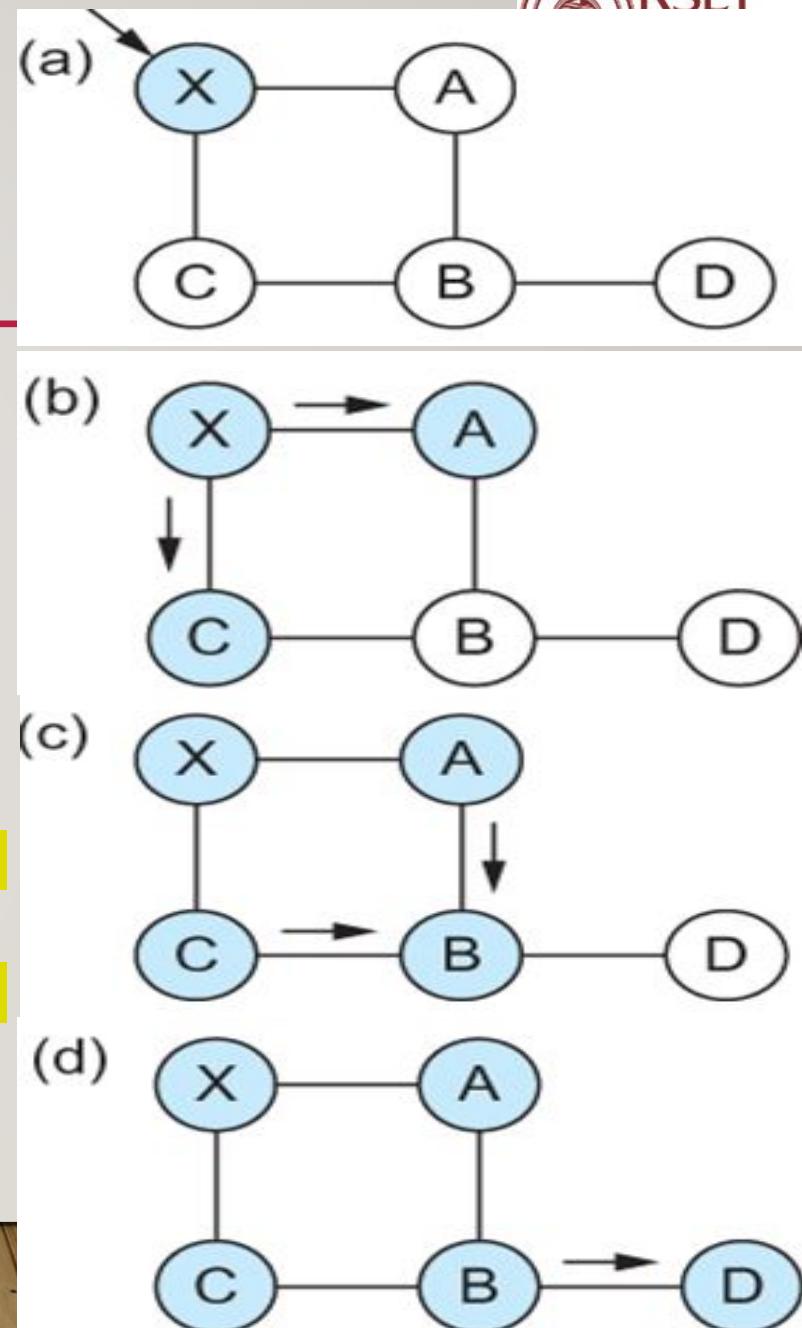
# Disadvantages

---

- It does a blind search, so wastes a lot of time while processing.
- It can't handle negative edges.
- It leads to the acyclic graph and most often cannot obtain the right shortest path.
- Need to keep track of vertices that have been visited.

# Flooding

- Non adaptive routing algorithm.
- Every incoming packet is sent out on every outgoing line except the one it arrived on.
- **Disadvantage:**
  - Duplicate Packets.
- **Solution:**
  - Source router put a sequence number in each packet it receives from its hosts.
  - Each router maintains a list per source router telling which sequence numbers originating at that source have already been seen.
  - If an incoming packet is on the list, it is not flooded.



# Flooding - Disadvantages

---

- Flooding tends to create an infinite number of **duplicate data packets**, unless some measures are adopted to damp packet generation.
- It is wasteful if a **single destination** needs the packet, since it delivers the data packet to all nodes irrespective of the destination.
- The network may be **clogged** with unwanted and duplicate data packets. This may hamper delivery of other data packets.

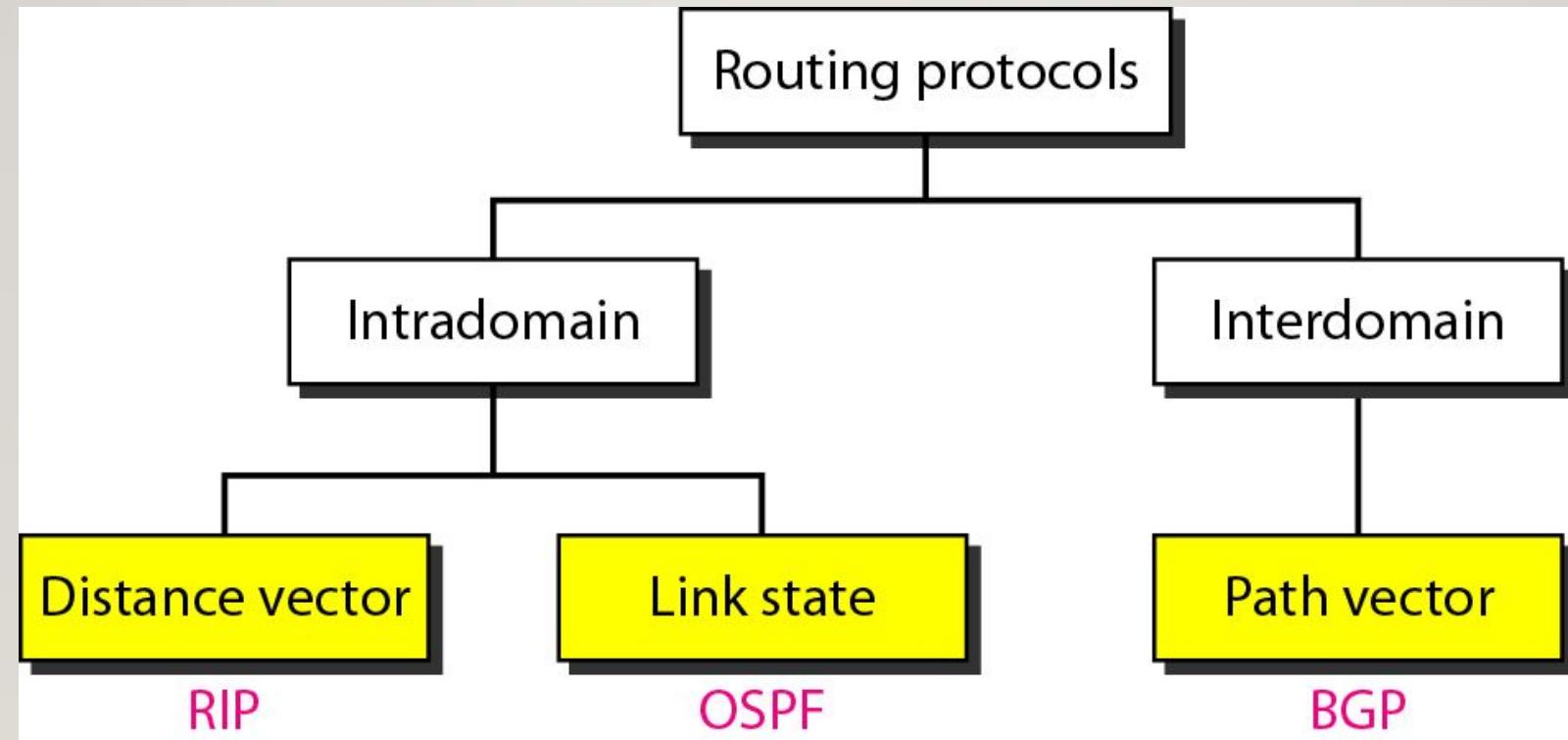
# Inter and Intra Domain Routing

---

- Internet is divided into autonomous systems (AS).
- An AS is a group of networks and routers under the authority of a single administration.
- Routing inside an AS is called intra-domain routing.
- Routing between AS is called inter-domain routing.

# Routing Protocols

---



# Distance Vector Routing

---

- Dynamic (Adaptive) routing algorithm.
- Also known as **Bellman – Ford routing algorithm** or **Ford-Fulkerson algorithm**.
- It is used in Internet under the name **Routing Information Protocol (RIP)**.
- Each router maintains a **routing table** indexed by, and containing one entry for each router in the network.

# Distance Vector Routing

---

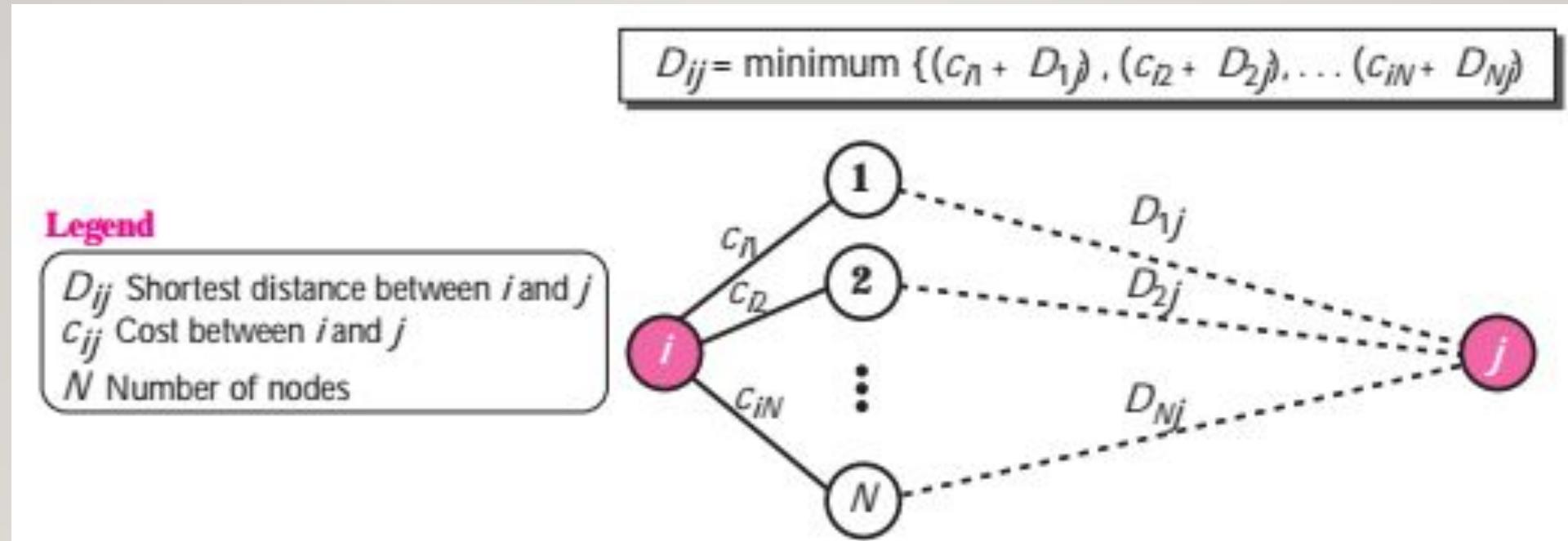
- Routing table for a router keeps three pieces of information:
  - destination router.
  - the distance/cost (The distance might be measured as the number of hops or using another metric).
  - the next hop (preferred outgoing line).

# Distance Vector Routing- Working

---

- Every router discovers the identity of the immediate neighbours and knows the **distance** to each of its neighbours.
- Each router constructs a list containing the distances to all other nodes and distributes this to all its immediate neighbours at regular intervals.
- A router receives this vector from all its neighbours and **updates** its table.
- After a router has updated its routing table, it should send the result to its neighbours so that they can also update their routing table.

# Distance Vector Routing-Bellman Ford Algorithm



# Distance Vector Routing-Shortest Distance Table

---

- Create a shortest distance table for each node:
  - Shortest distance and cost between a node and itself is initialized to 0.
  - Shortest distance between a node and any other node is set to infinity.
  - Cost between a node and any other node will be available if connected, else infinity.
- Run the algorithm.
- Cost- hop count

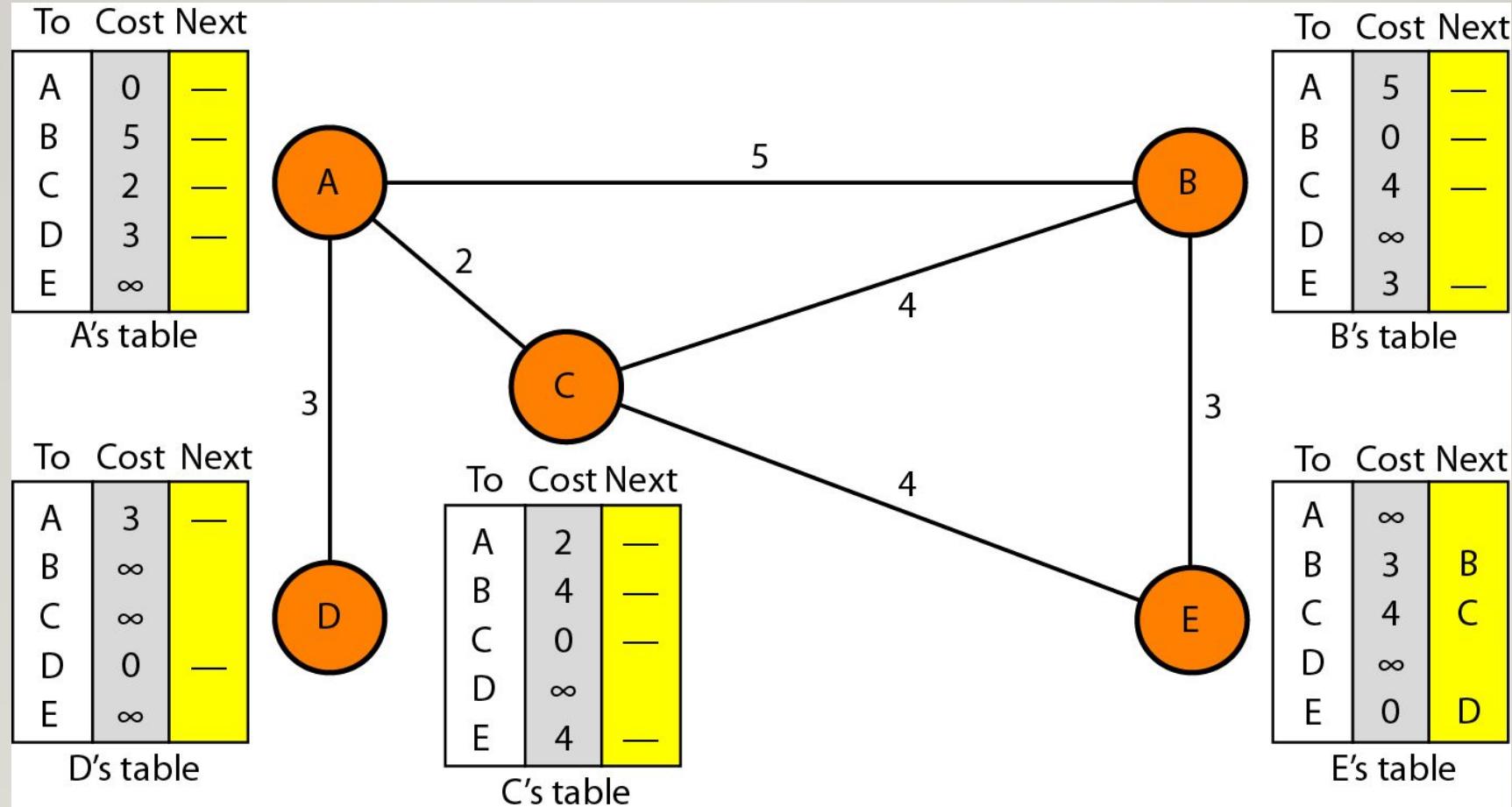
```
Bellman_Ford ( )  
{  
    // Initialization  
    for (i = 1 to N; for j = 1 to N)  
    {  
        if (i == j) Dij = 0 cij = 0  
        else Dij = ∞ cij = cost between i and j  
    }  
    // Updating  
    repeat  
    {  
        for (i = 1 to N; for j = 1 to N)  
        {  
            Dij ← minimum [(ci1 + D1j) . . . (ciN + DNj)]  
        } // end for  
    } until (there was no change in previous iteration)  
} // end Bellman-Ford
```

# Distance Vector Routing- Steps

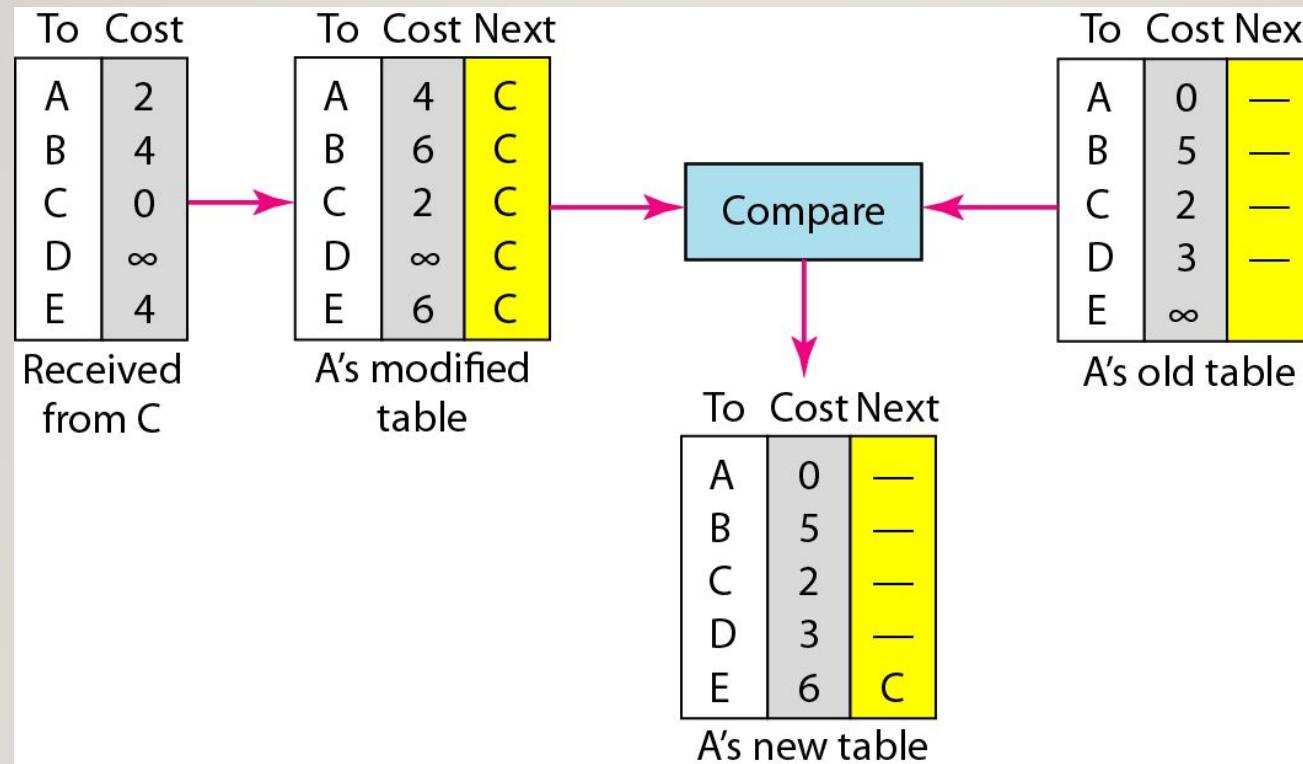
---

- Initialization
  - Each node knows distance between itself and immediate neighbour only. Others are marked as infinity.
- Sharing
  - Nodes share their routing table with immediate neighbours periodically when there is a change.
- Updating
  - Update the infinity values using information obtained.

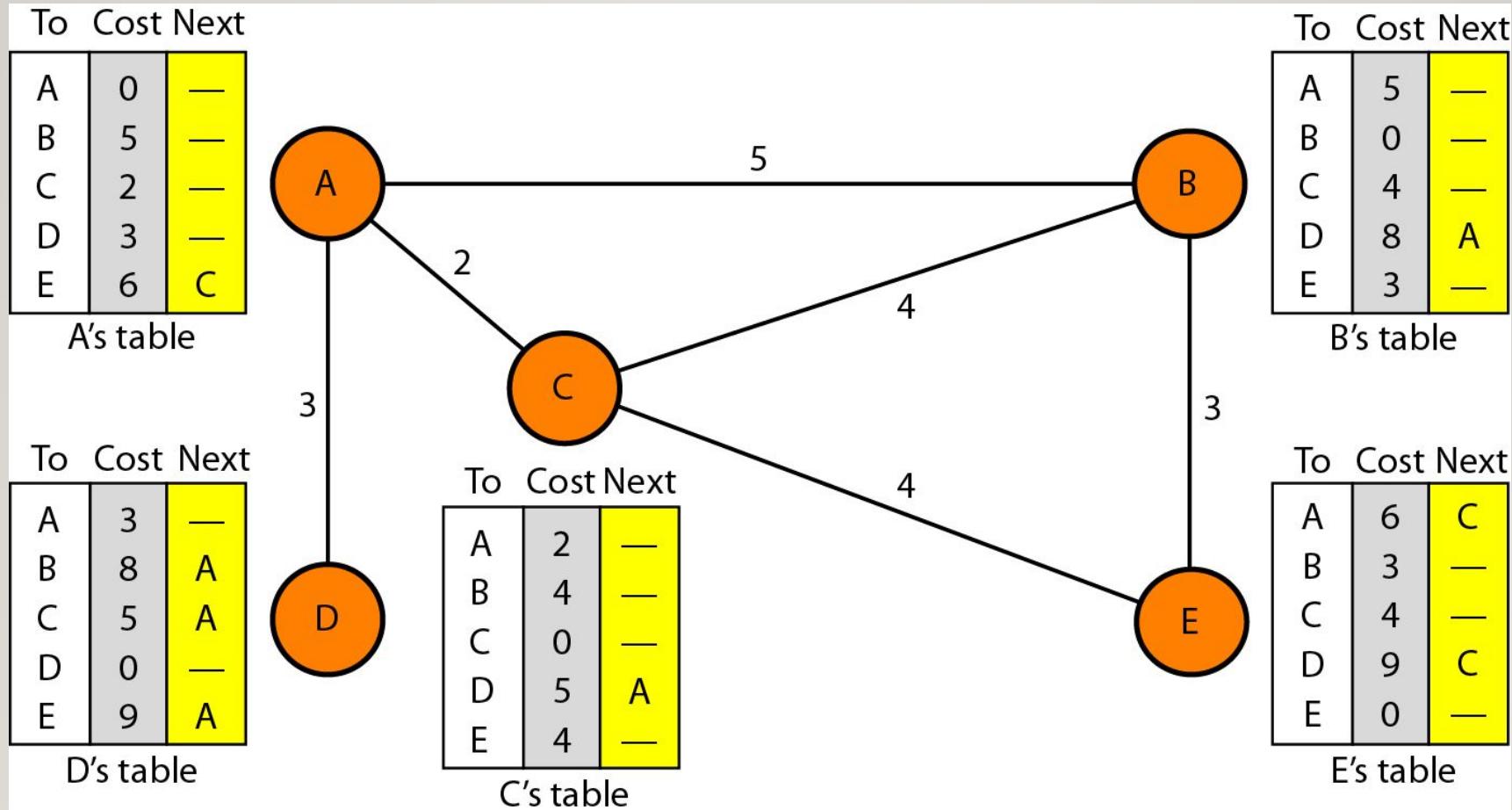
## Example : Step1: Initialization of tables in distance vector routing



## Step 2: Updating in distance vector routing



## Distance vector routing tables



# Updating The Routing Table

---

- If the next hop entry is different:
  - If new value is smaller than cost in table, take new one.
  - If there is a tie, the old one is kept.
  - If entry not in table, then add to the table.

# When To Share the Routing 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 routing table to its neighbors anytime there is a change in its routing table.

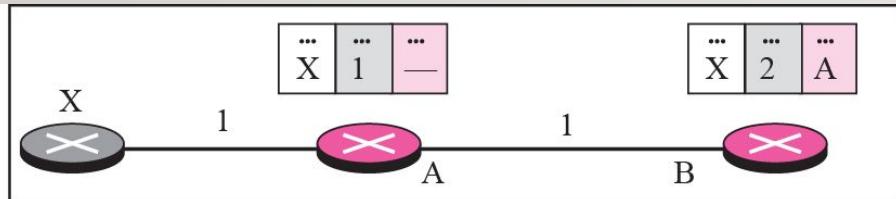
# Limitations of DVR- Count To Infinity Problem.

---

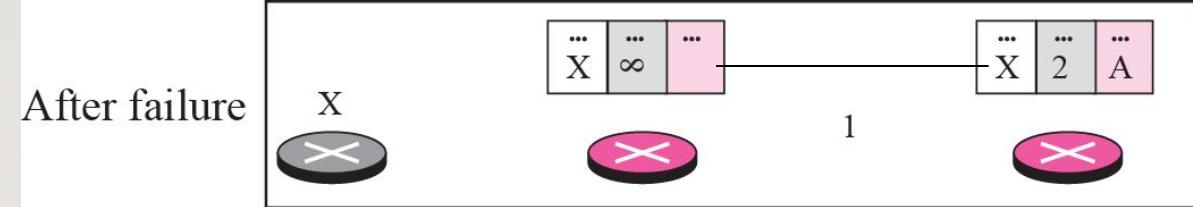
- The settling of routes to best paths across the network is called **Convergence**.
- It converges to the correct route, but it may do so slowly.

# Count To Infinity Problem

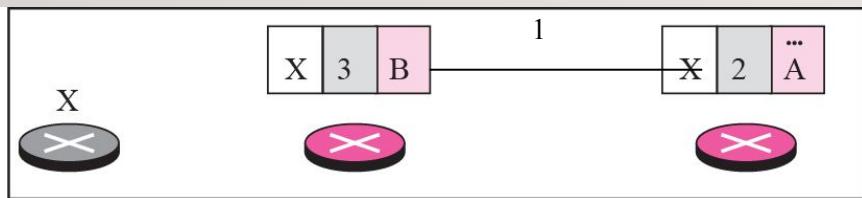
Before failure



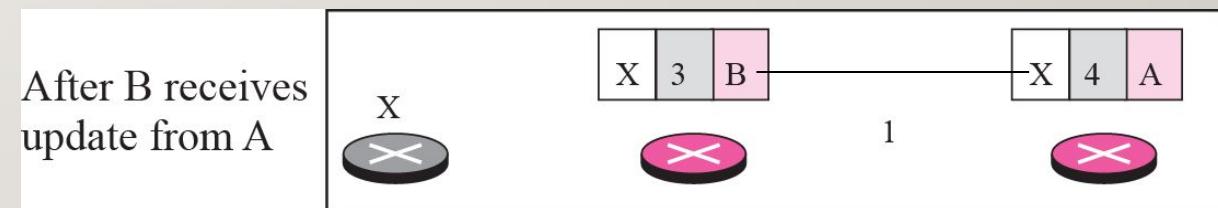
After failure



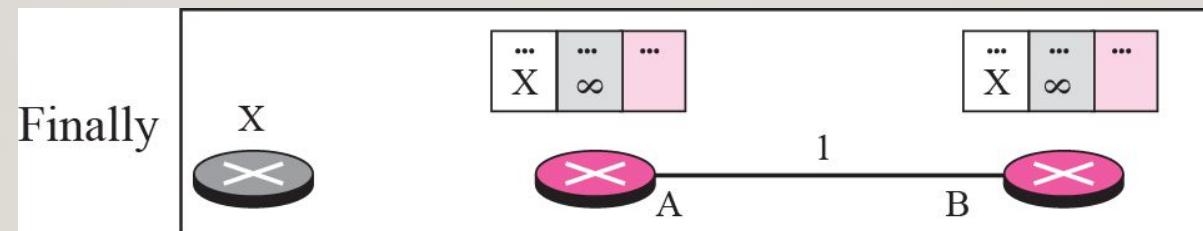
After A receives update from B



After B receives update from A



Finally



# SOLUTION

---

- **Defining infinity**
  - Most implementations define 16 as infinity. So DVR cannot be used in large systems. Maximum 15 hops only allowed.
- **Split Horizon**
  - Instead of flooding the table through each interface, each node sends only part of its table through each interface. So even if some connections are mistaken to be true, confusions wont be created.

# COMPARISON

---

Parameter	Bellman-Ford	Dijkstra
Overheads	More	Less
Scalability	Less	More
Quality of the best available route	Less	More
Negative edges	Yes	No
Delay	More	Less

# ~~Link State Routing~~

- If each node in the domain has the information on entire topology of the domain- the list of nodes, links, how they are connected, type, cost and condition of links:
  - Then Dijkstra's algorithm can be used to build a routing table.
- But the topology is dynamic, it represents the latest situation of each node and link. If there are any changes in the network, topology must be updated.
- For this a partial knowledge of the topology is maintained at every node- **Link state knowledge**.

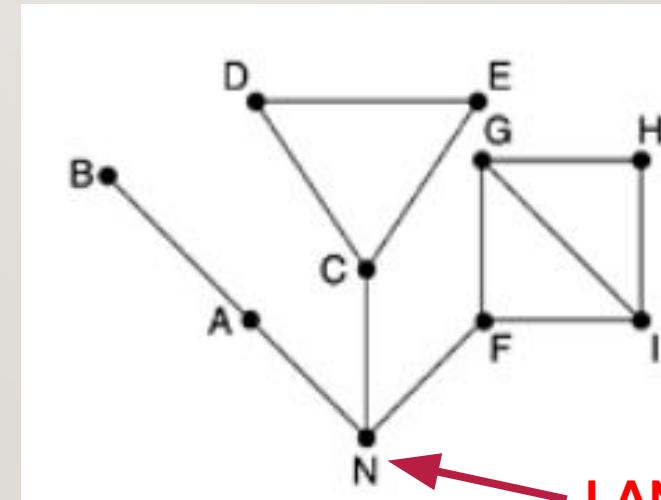
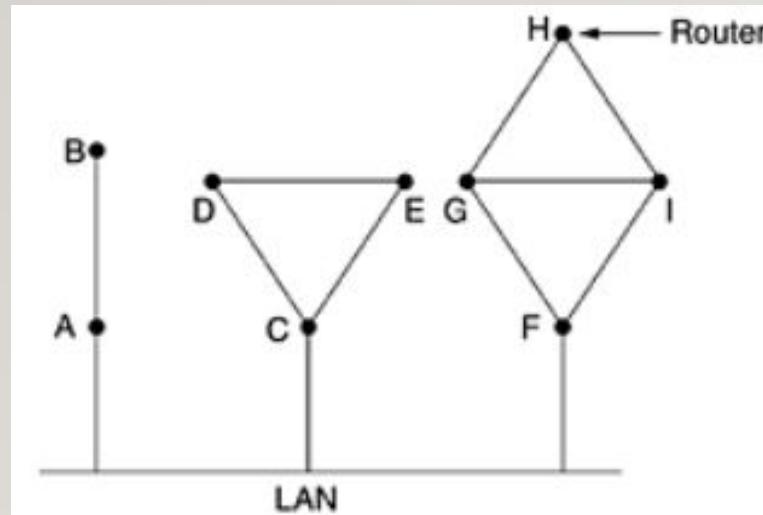
# ~~Steps In Link State Routing~~

---

- Each router must:
  1. Discover its neighbours and learn their network addresses.
  2. Measure the delay or cost to each of its neighbours.
  3. Construct a packet telling all information just learned.
  4. Send this packet to all other routers.
  5. Compute the shortest path to every other router.

# ~~Learning About Neighbours~~

- On booting, the router will send a **HELLO** packet on each of its links.
- The router on the other end will send back a reply which contains its network address.



**LAN represented as a node**

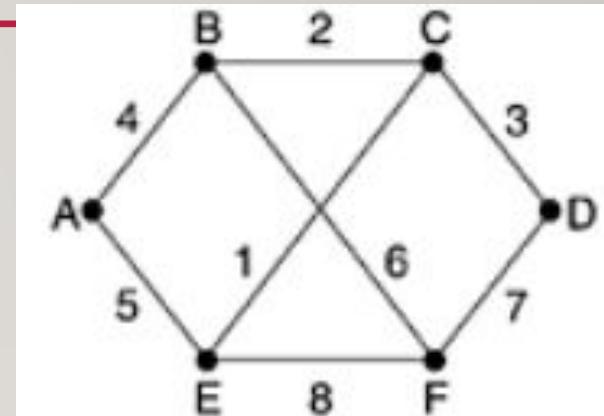
# ~~Measuring The Link Cost~~

---

- LSR algorithm requires each link to have a distance or cost metric for finding shortest paths.
- The Cost to reach neighbours can be set automatically or configured by the network operator.
- The most common way to determine this cost is by sending **ECHO** packets and measuring the round trip time and dividing it by two.

# ~~Building Link State Packets~~

- Router builds a packet that contains:
    - Identity of the sender.
    - Sequence number.
    - Age- bound the maximum lifetime of a link state packet in the network.
    - List of neighbours and the cost to each.



Link	State	Packets			
A	B	C	D	E	F
Seq.	Seq.	Seq.	Seq.	Seq.	Seq.
Age	Age	Age	Age	Age	Age
B 4	A 4	B 2	C 3	A 5	B 6
E 5	C 2	D 3	F 7	C 1	D 7
	F 6	E 1		F 8	E 8

# ~~Building Link State Packets~~

---

- When to build the packet?
  - Build them periodically **at regular intervals**.
  - Build them **when some significant event occurs**, such as a line or neighbour going down or coming back up again.

# ~~Distributing The Link State Packets~~

- **Flooding** is used to disseminate this information.
- A node will sent its LSP out on all its directly connected links.
- Each node that receives the LSP from some node will check if it already has a copy of the LSP.
  - If not it will store it and will sent it out on all its links.
  - If it already has a copy it will compare the SEQNO.
    - If the new LSP has a larger SEQNO; it is more recent – so this LSP is stored and is sent out on all its links.
    - If the new LSP has a smaller (or equal) SEQNO it is discarded.

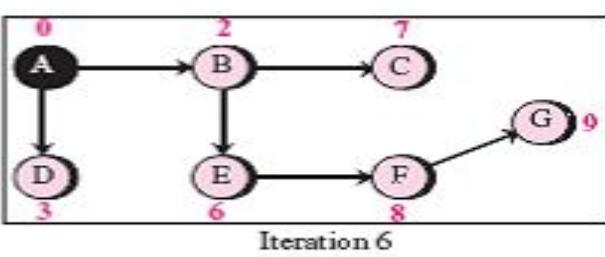
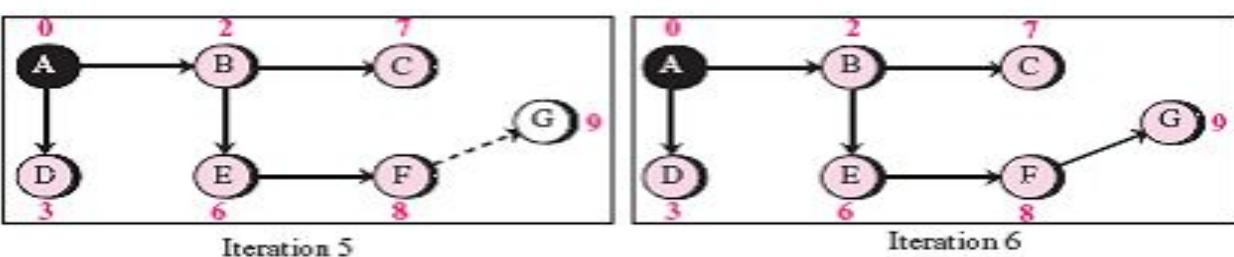
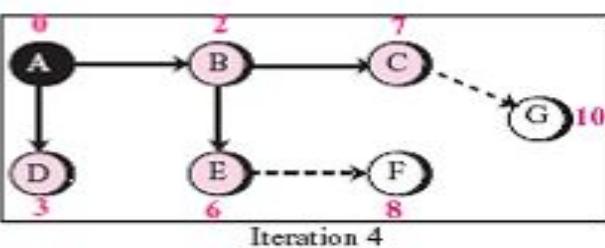
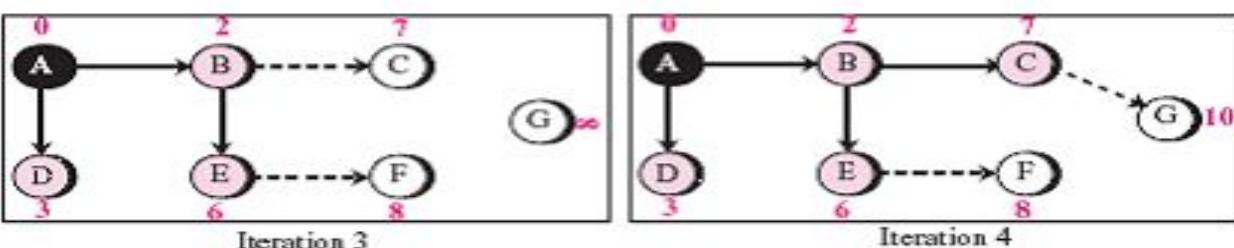
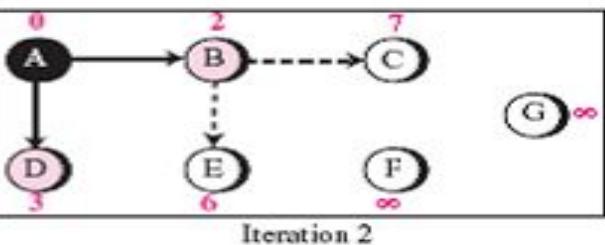
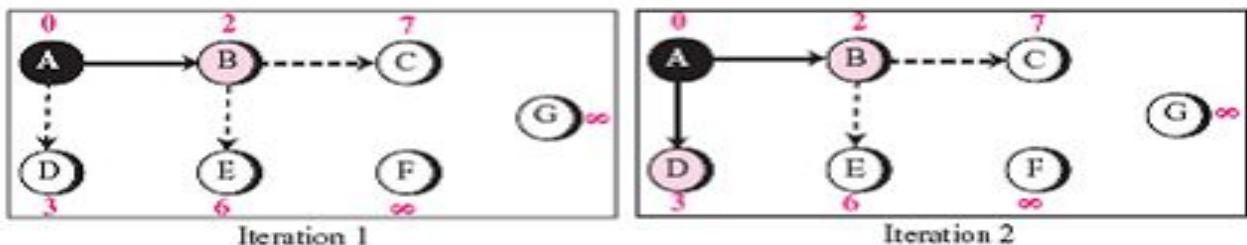
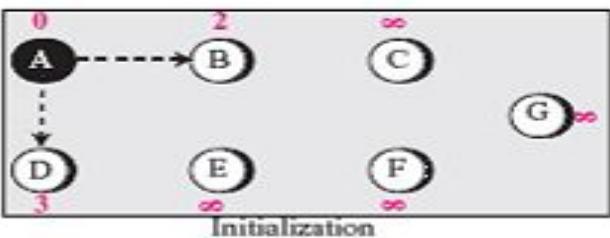
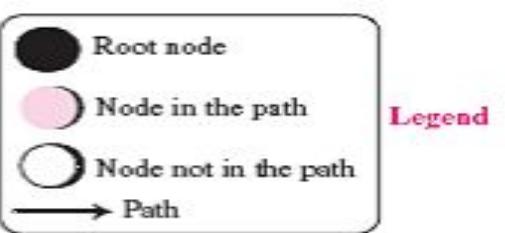
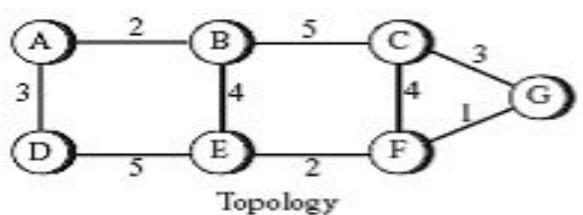
# ~~Distributing The Link State Packets~~

- Problems with comparing sequence number:
  - If a router crashes, it will lose track of its sequence number. If it starts again at 0, the next packet it sends will be rejected as a duplicate.
- Solution: Use the **age** field
  - When a router generates a LSP, it sets its lifetime (usually measured in seconds) in the *age* field
  - Decrement age once per second.
  - When age hits zero, the information from that router is discarded.
  - Helps to make sure that no packet live for an indefinite period of time.
  - LSPs from a failed router does not remain.

# ~~Computing The New Routes~~

---

- When a router gets the full set of LSPs, it constructs the routing table.
  - Use **Dijkstra algorithm** is used to find the shortest path to all destinations.
  - Then construct the routing table.



Routing table for node A

Destination	Cost	Next router
A	0	-
B	2	-
C	7	B
D	3	-
E	6	B
F	8	B
G	9	B

# ~~Limitations~~

---

- They require more memory and processor power than distance vector protocols. This makes it expensive to use for organizations with small budgets and legacy hardware.
- They require strict hierarchical network design, so that a network can be broken into smaller areas to reduce the size of the topology tables.
- They require an administrator who understands the protocols well.
- They flood the network with LSPs during the initial discovery process. This process can significantly decrease the capability of the network to transport data. It can noticeably degrade the network performance.

<b>Link State routing algorithm</b>	<b>Distance Vector routing algorithm</b>
The network topology and all the link costs are the input to this algorithm.	The input to the algorithm is all the associated costs with the current node to all its neighbors.
It computes the least-cost path from source to destination with a complete knowledge on the network.	It computes the least-cost path in an iterative and distributed manner.
The shortest path is calculated using dijkstras algorithm.	The shortest cost path is calculated using Bellman Ford algorithm.
Open Shortest Path First (OSPF) is an example of link state routing algorithm.	Routing Information Protocol (RIP) is an example of distance vector routing algorithm.

# THANK YOU!!!

---

# **COMPUTER NETWORKS**

---

**MS. JINCY J. FERNANDEZ**

**ASST. PROF, CSE**

**RSET**

# Multicast Routing

---

- Some applications require widely separated processes to work together in groups.
- It may be required for 1 process to send messages to all other members of the group.
- Broadcasting and unicasting may not be good in these cases.
- So go for **multicasting**: sending messages to well-defined groups that are numerically large in size, but small compared to the network as a whole.
- Routing algorithm for performing multicasting is called multicast routing.
- A single router in a network can be part of two or more different groups.
- Multicasting requires group management.

# Multicast Routing

---

- There should be provision to create, destroy groups, join and leave groups.
- Main concern of the routing algorithm is to identify which routers are members of a group.
- Hosts can then inform their routers about changes in the network.
- Each group is identified by a multicast address and that routers know the groups to which they belong.
- To do multicast routing, each router computes a **spanning tree** covering all other routers.

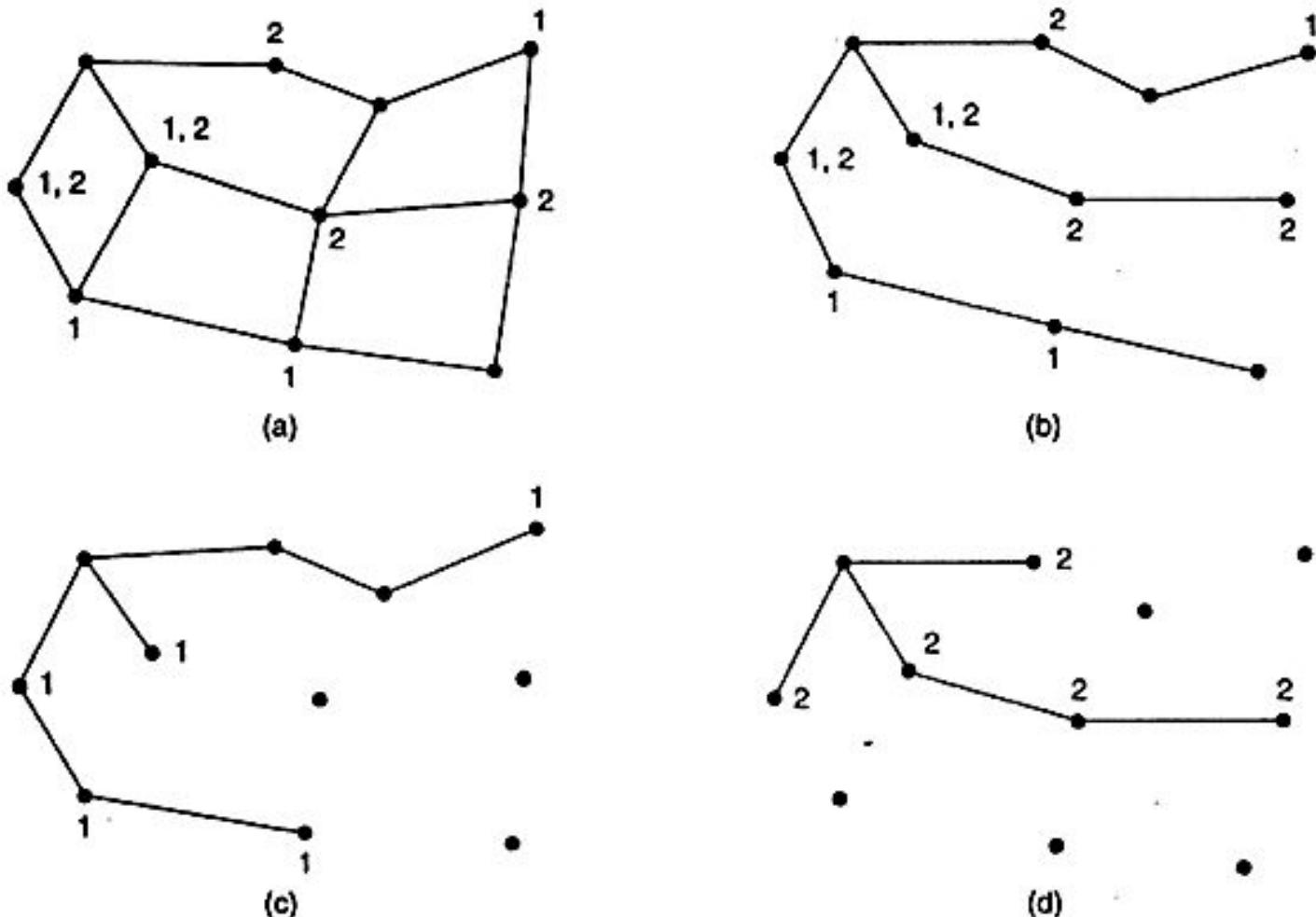


Fig. 1. (a) A subnet. (b) A spanning tree for the leftmost router. (c) A multi-cast tree for group 1. (d) A multi-cast tree for group 2.

# Working

---

- When a process sends a multicast packet to a group, the first router examines its spanning tree and prunes it, removing all lines that do not lead to hosts that are members of the group.
- In previous example, Fig. 1(c) shows the pruned spanning tree for group 1. Similarly, Fig. 1(d) shows the pruned spanning tree for group 2.
- Multicast packets are forwarded only along the appropriate spanning tree.

# How To Prune?- Multicast Distance Vector

---

- Extension of Unicast routing.
- Uses the source-based tree approach to multicasting.
- Each router that receives a multicast packet to be forwarded implicitly creates a source-based multicast tree in 3 steps:
  - Router uses an algorithm called **Reverse Path Forwarding (RPF)** to simulate creating part of the optimal source-based tree between source and itself.
  - Router uses an algorithm called **Reverse Path Broadcasting (RPB)** to create a broadcast spanning tree whose root is the router itself and whose leaves are all networks in the internet.
  - Router uses an algorithm called **Reverse Path Multicasting (RPM)** to create a multicast tree by cutting some branches of the tree that end in networks with no member in the group.

# How To Prune?-Link State Routing

---

- The simplest one can be used if **link state routing** is used, and each router is aware of the complete subnet topology, including which hosts belong to which groups.
- Then the spanning tree can be pruned by starting at the end of each path and working toward the root removing all routers that do not belong to the group in question.

# How To Prune?- Multicast Distance Vector

---

- With **distance vector routing**, a different pruning strategy can be followed. The basic algorithm is reverse path forwarding. However, whenever a router with no hosts interested in a particular group and no connections to other routers receives a multicast message for that group, it responds with a PRUNE message, telling the sender not to send it any more multicasts for that group. When a router with no group members among its own hosts has received such messages on all its lines, it, too, can respond with a PRUNE message. In this way, the subnet is recursively pruned.

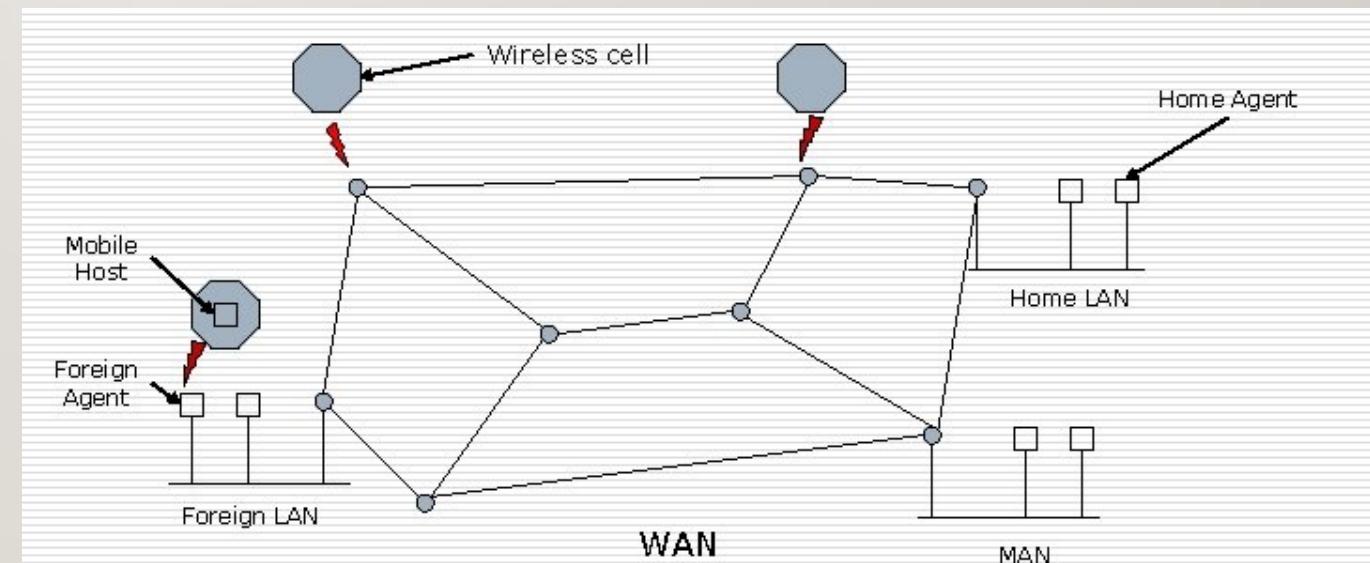
# Types of Multicast Routing Protocol

---

- 1) Multicast Distance Vector Routing Protocol (DVMRP)
- 2) Multicast Link State (MOSPF)
- 3) Protocol Independent Multicast (PIM)

# Routing For Mobile Hosts

- People use portable computers and devices nowadays.
- Routing of packets to such devices is also necessary.
- **Mobile Hosts:** Hosts that move from one network to another.



# Routing For Mobile Hosts

---

- Two addresses for a mobile host:
  - **Home address:** a permanent address that associates the hosts with its **home network** (home location).
  - **Care of address:** a temporary address; it changes when a host moves from one network to another. It is associated with the **foreign network** (the network to which the host moves).
- The role of routing is to make it possible to send packets to mobile hosts using their fixed home addresses and have the packets efficiently reach them wherever they may be.

# Mobile Host Routing

---

- World is divided geographically into areas.
- Home location will have a **home agent** (a router attached to the home network of the mobile host).
- Home agent acts on behalf of the mobile host when a remote host sends a packet to the mobile host.
- Each area has a **foreign agent** that keep track of mobile hosts visiting an area.
- Once the home agent knows where the mobile host is currently located, it can forward packets so that they are delivered to the mobile host.

# Mobile Host Routing- Working- 3 phases

---

## I. Agent Discovery

- Mobile host must learn the address of the home agent before it leaves the network.
- Mobile host must discover a foreign agent after it has moved to a foreign network.
- Discovery involves two types of messages:
  - **Agent Advertisement:** router advertise its presence on the network if it acts as an agent.
  - **Agent Solicitation:** When a mobile host has moved to a new network and has not received agent advertisement, it can initiate agent solicitation.

# Mobile Host Routing- Working- 3 phases

---

## 2. Registration

- After a mobile host has moved to a foreign network and discovered the foreign agent, it must register.
- Aspects of Registration
  - Mobile hosts registers with the foreign agent and home agent.
  - Mobile host must renew the registration if it has expired.
  - Mobile host must cancel its registration when it returns home.
- Mobile host uses a registration request and registration reply to register with the foreign agent and the home agent.

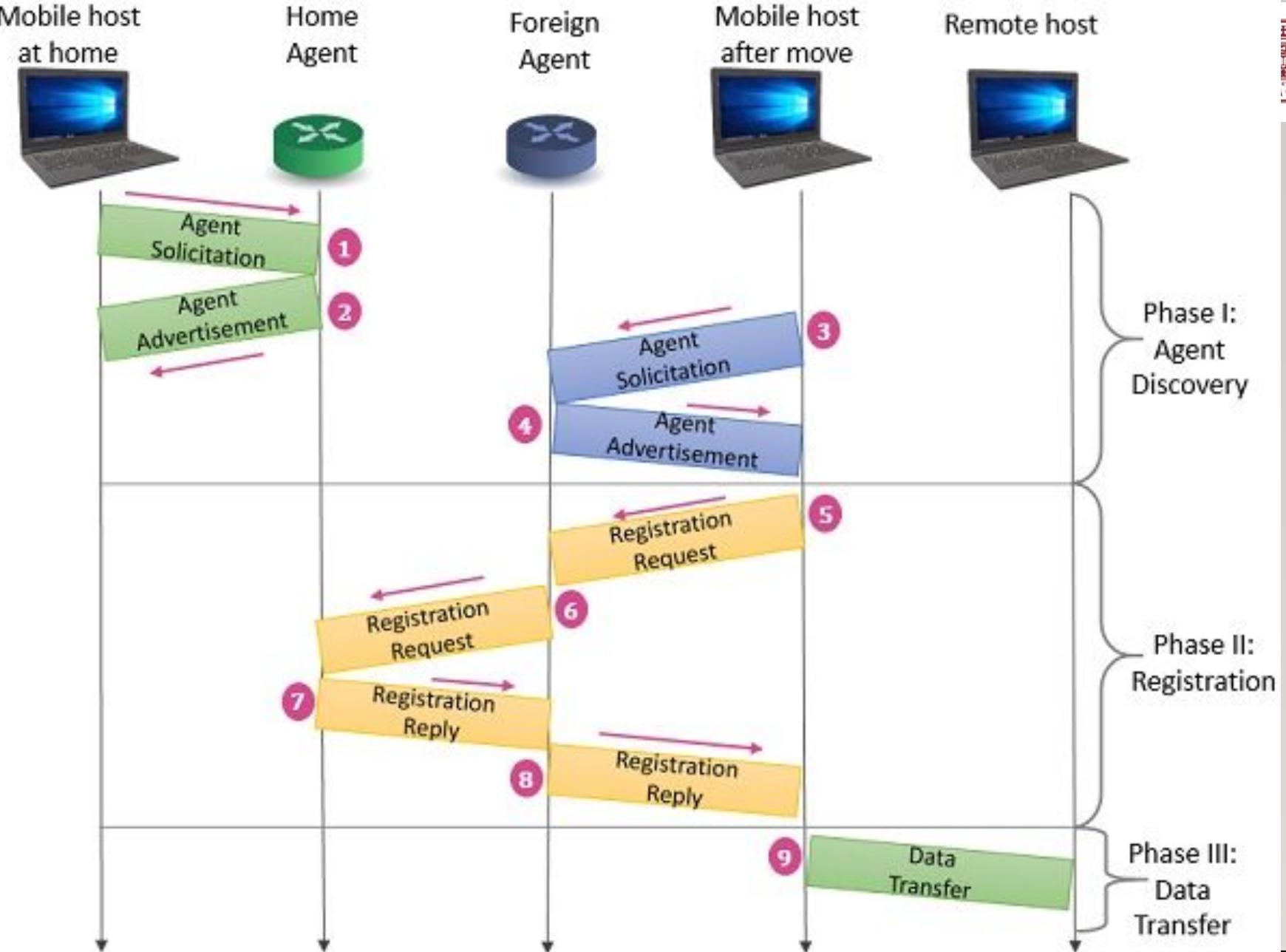
# Mobile Host Routing- Working- 3 phases

## 2. Registration :

- Request and Reply
  - Send registration request from the mobile host to the foreign agent to register its care of address and to announce home address and home agent address.
  - Foreign agent relays the message to the home agent.
  - Home agent knows the address of the foreign agent.
  - A registration reply is sent from the home agent to the foreign agent and then relayed to the mobile host.
  - The reply confirms or denies the registration request.

# Mobil

---

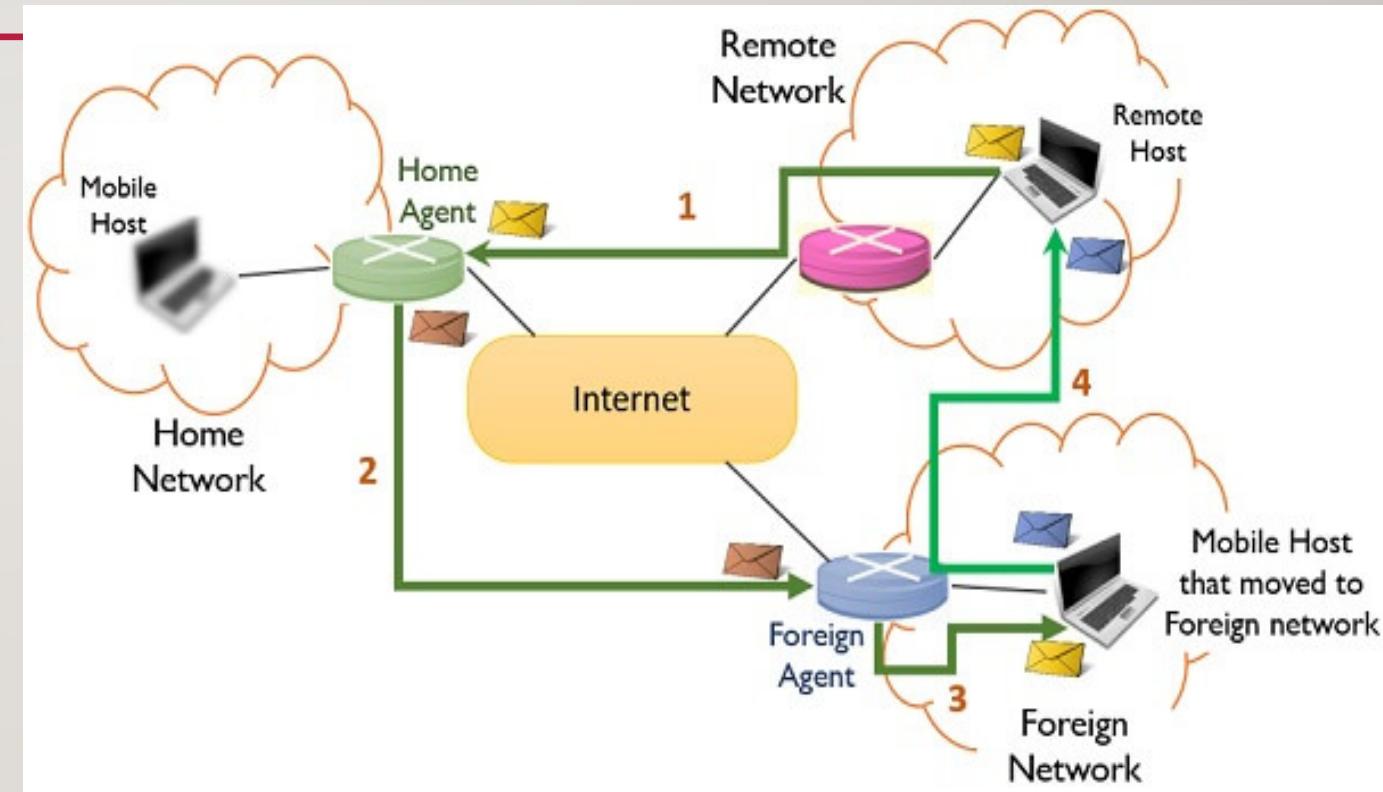


Communication Between Mobile Host and Remote Host

# Mobile Host Routing- Working- 3 phases

## 3. Data Transfer

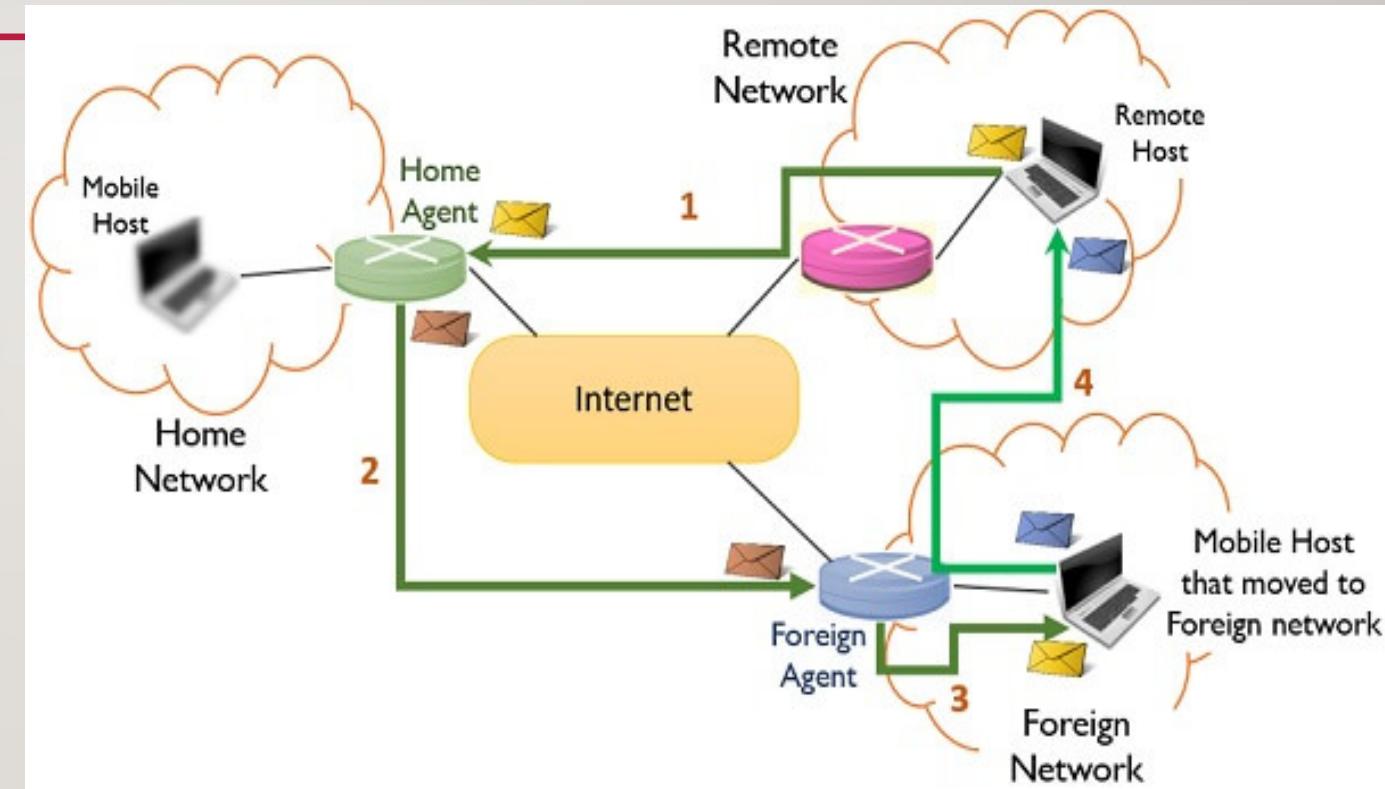
- A mobile host can communicate with the remote host.
- Remote host sends to home agent.
  - Home agent then wraps or encapsulates the packet with a new header and sends this bundle to the foreign agent- This mechanism is called **tunneling**.



# Mobile Host Routing- Working- 3 phases

## 3. Data Transfer

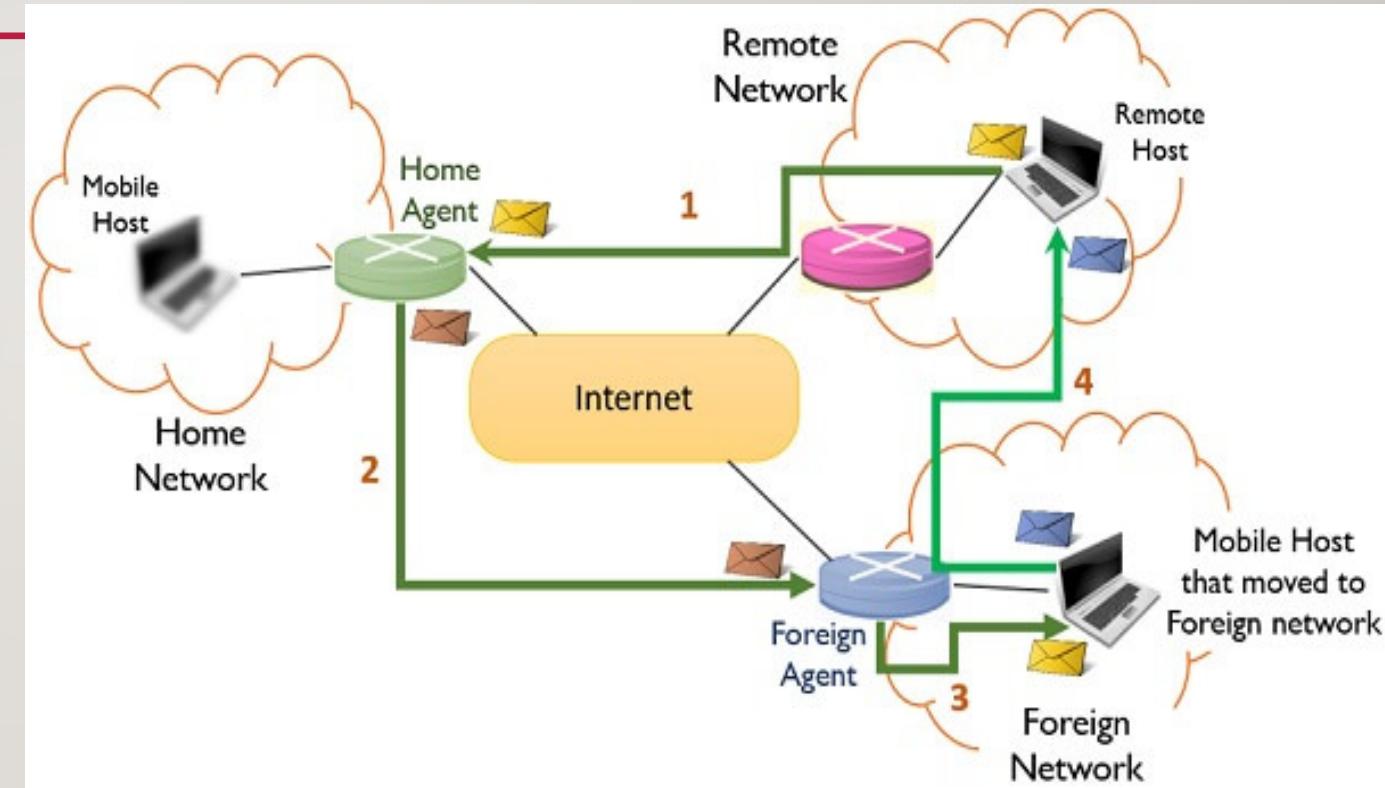
- c) When foreign agent receives the packet, it takes the original packet. Foreign agent consults a registry table to find the care of address of the mobile host. Send the packet to the care of address.



# Mobile Host Routing- Working- 3 phases

## 3. Data Transfer

- d) Normal data transfer if mobile host wants to send to a remote host.



# THANK YOU!!!

---

# COMPUTER NETWORKS

---

MS. JINCY J FERNANDEZ

ASST. PROF, CSE

RSET

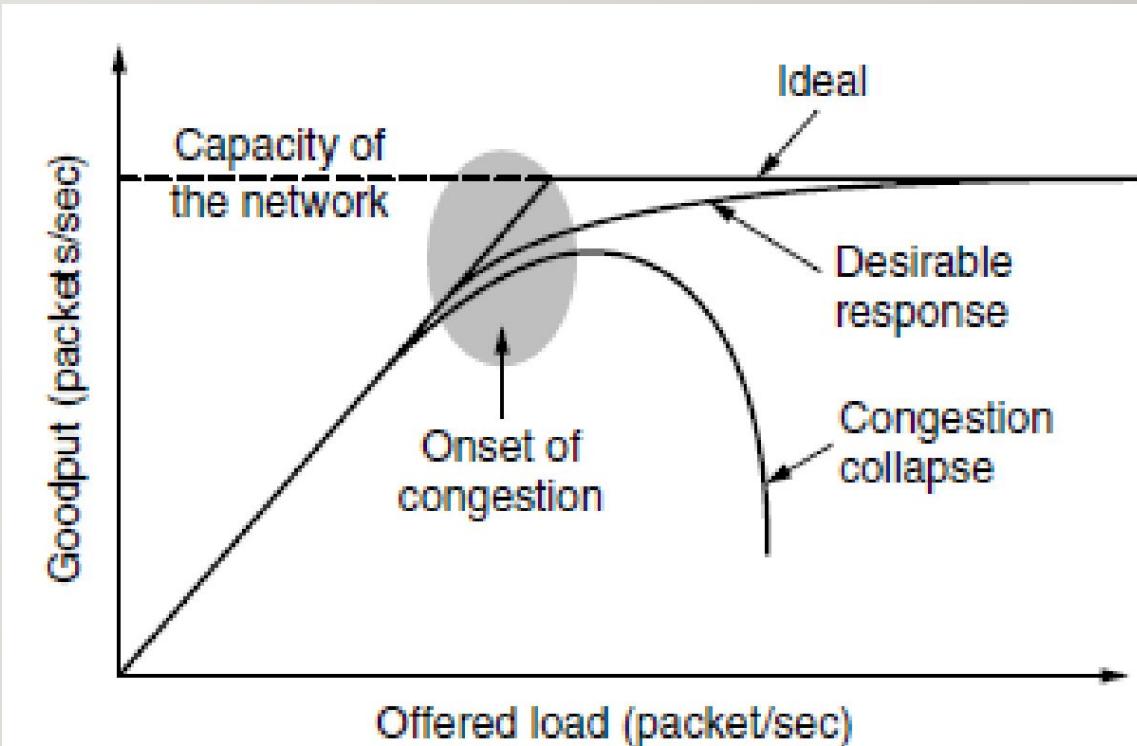
# Congestion Control Algorithms

---

- **Congestion:** Too many packets in the network cause packet delay and loss that degrades the performance.
- The network and transport layers share the responsibility for handling congestion.
- The most effective way to control congestion is to reduce the load that the transport layer is placing on the network.

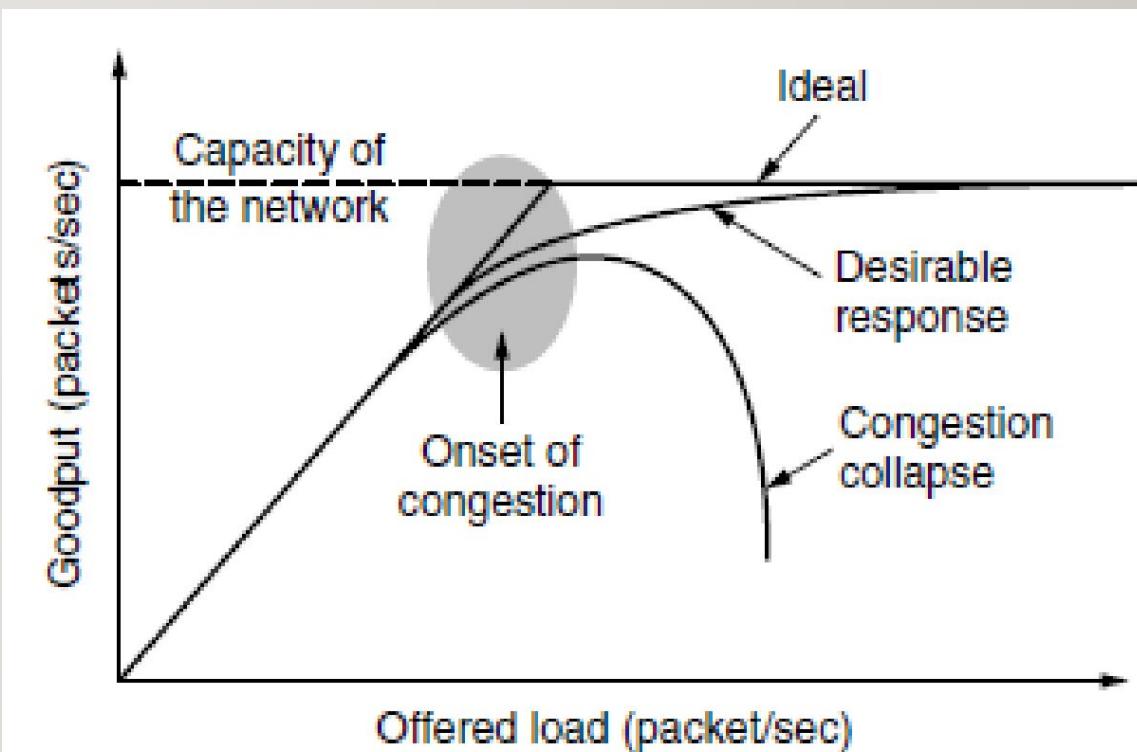
# Congestion

- When the number of packets the hosts sent is within the carrying capacity, the number of packets delivered will be proportional to the number of packets sent.
- As the offered load approaches the carrying capacity, bursts of traffic fills up the buffers inside the routers & some packets are lost.
- The lost packets consume some of the capacity, so the number of delivered packets falls below the ideal curve.



# Congestion

- **Goodput:**
  - The rate at which *useful* packets are delivered by the network.
  - Duplicate copies of the same packet delivered is not considered.
- **Congestion Collapse:**
  - Performance rapidly drops as the offered load increases beyond the capacity.



# Congestion

---

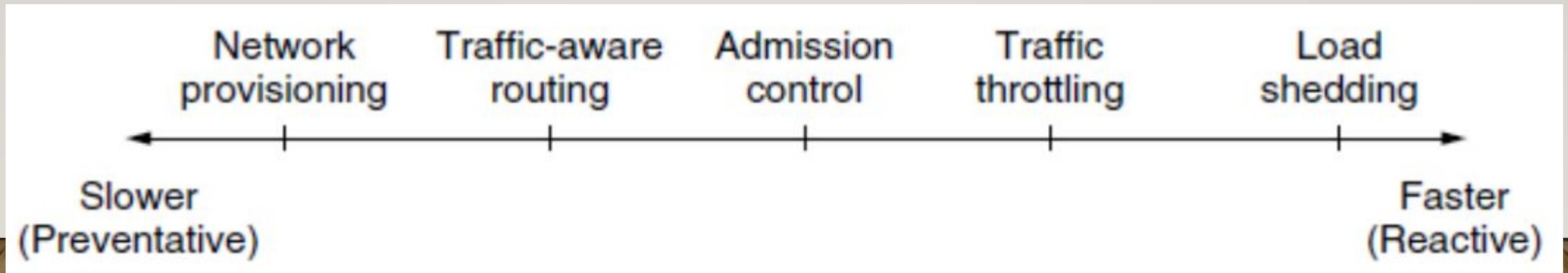
- Design networks that:
  - Avoid congestion where possible.
  - Do not suffer from congestion collapse if they do become congested.

# Reasons for Congestion

- Streams of packets arriving from multiple input lines require a common output line a queue will build up.
- Insufficient memory or buffer space to accumulate packets.
- Slow processing speed of machines.
- Low bandwidth of channels.

# How To Avoid Congestion?

- Presence of congestion means that the load is greater than the resources in the network can handle.
- Solutions:
  - Increase the resources.
  - Decrease the load.



# Approaches To Congestion Control

---

- Provisioning
  - Upgrading links and routers that are regularly heavily utilized.
    - ❖ Turning on spare routers.
    - ❖ Enabling lines that are normally used only as backups.
    - ❖ Purchasing bandwidth on the open market.
- Traffic-aware Routing
  - Routes may be changed to shift traffic away from heavily used paths by changing shortest path weights.

# Approaches To Congestion Control

---

- Admission Control
  - Decrease the load.
  - Used in virtual circuit networks.
  - Do not add a new virtual circuit unless the network can carry the added traffic without becoming congested.

# Approaches To Congestion Control

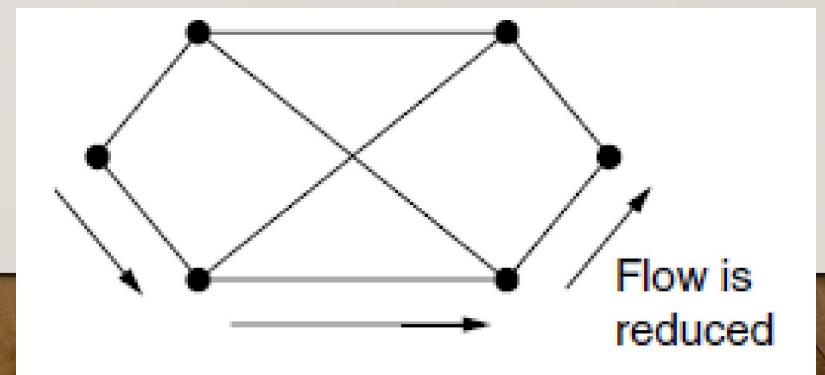
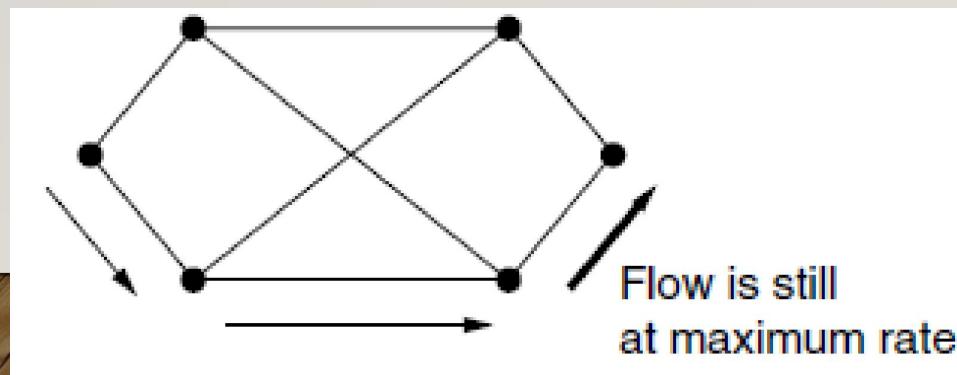
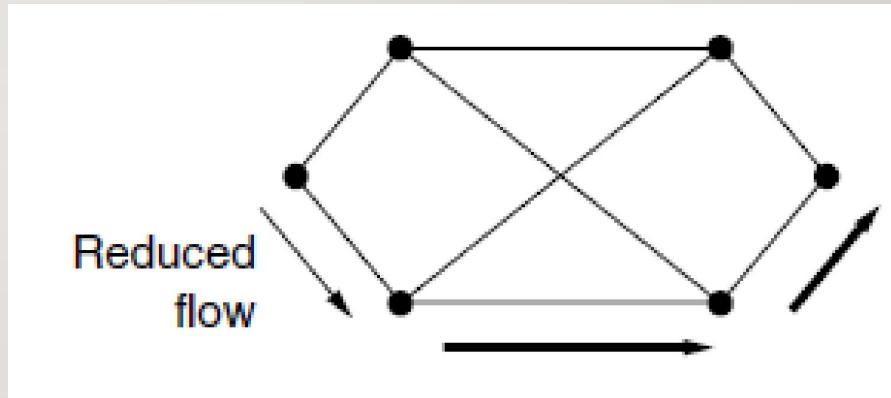
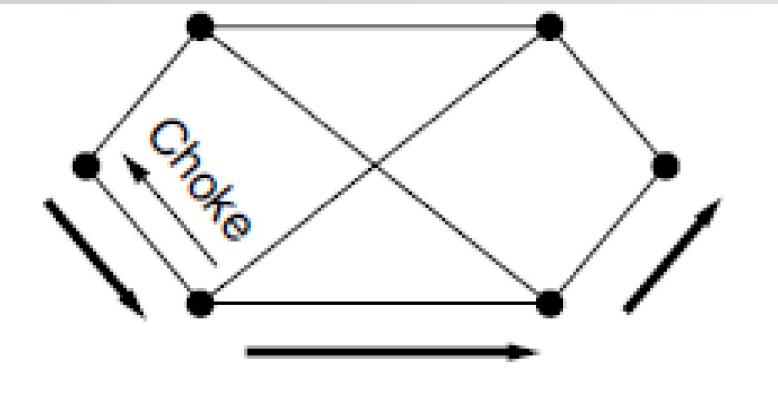
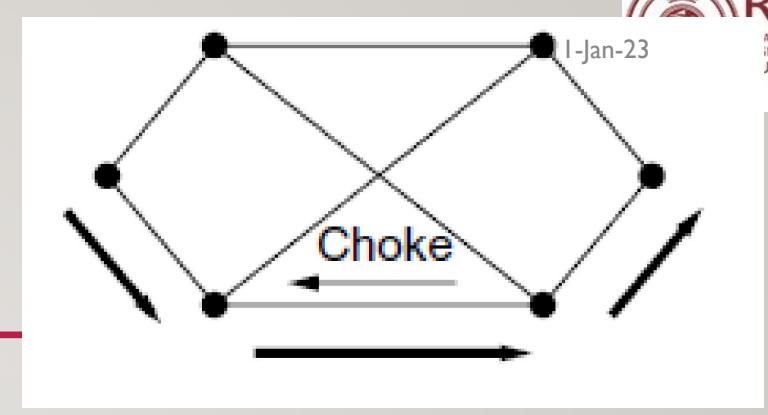
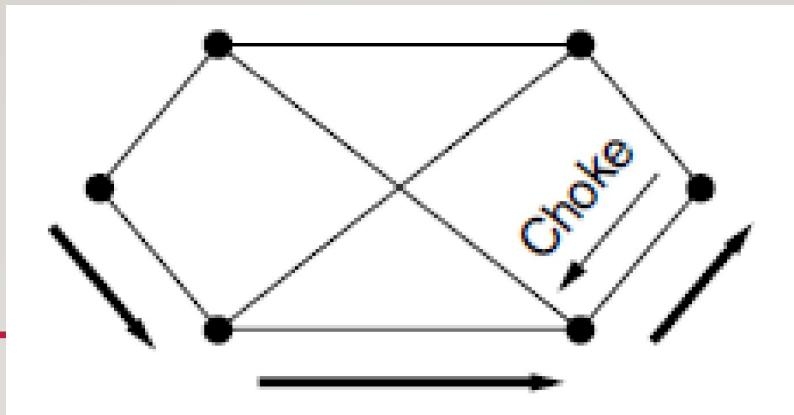
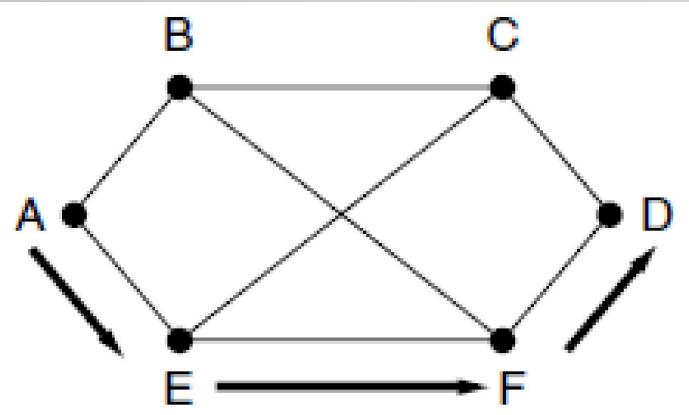
---

- Traffic throttling
  - Senders adjust their transmissions to send as much traffic as the network can readily deliver.
  - In this setting the network aim is to operate just before the onset of congestion.
  - Approach should consider:
    - Routers must determine when congestion is approaching. ↗ continuous monitoring of the resources.
    - Routers must deliver timely feedback to the senders that are causing the congestion.

# Approaches To Congestion Control

---

- Traffic throttling
  - Feedback Mechanisms
    - Router must identify the appropriate senders
  - a) **Choke Packets:** direct way to notify a sender of congestion.
    - A choke packet is a packet sent by a router to the source to inform it of congestion.
    - Router selects a congested packet and sends a choke packet back to the source host.

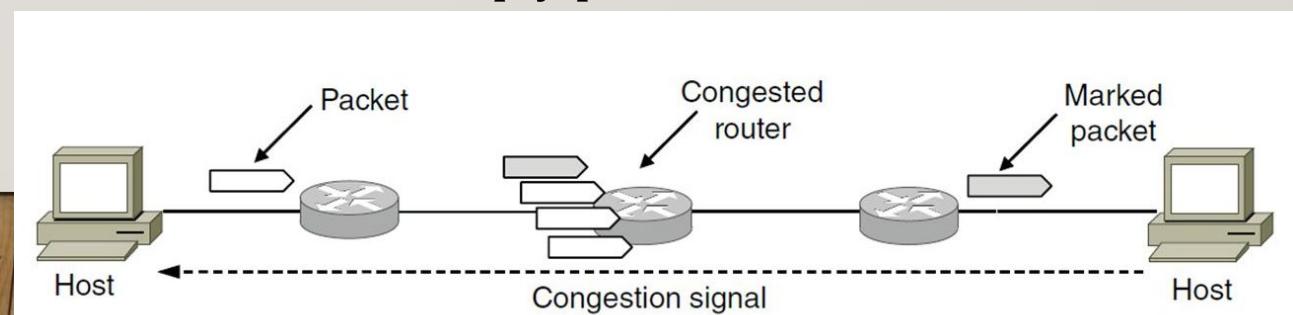


# Approaches To Congestion Control

- Traffic throttling
  - Feedback Mechanisms

## b) Explicit congestion Notification:

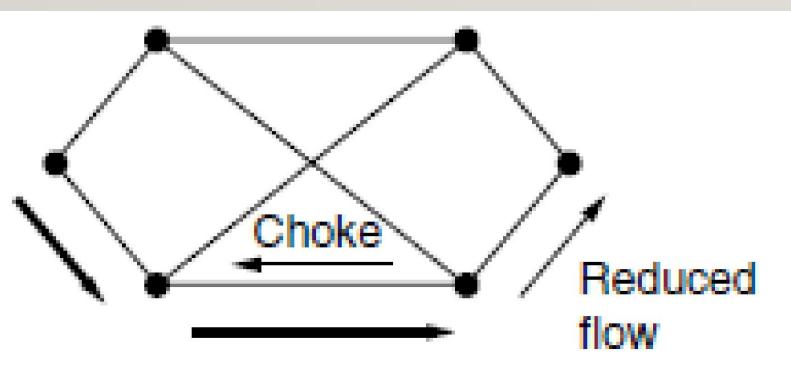
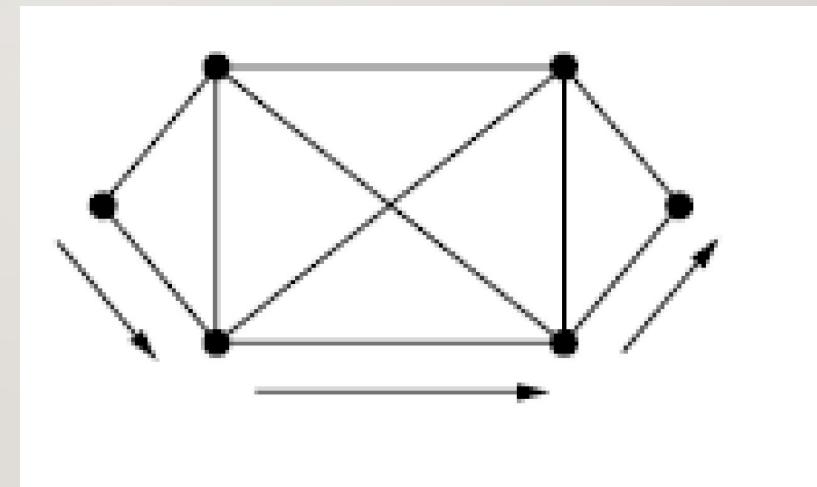
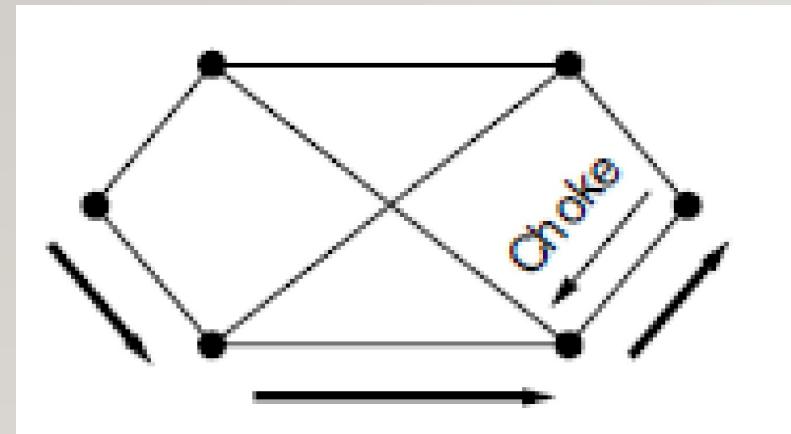
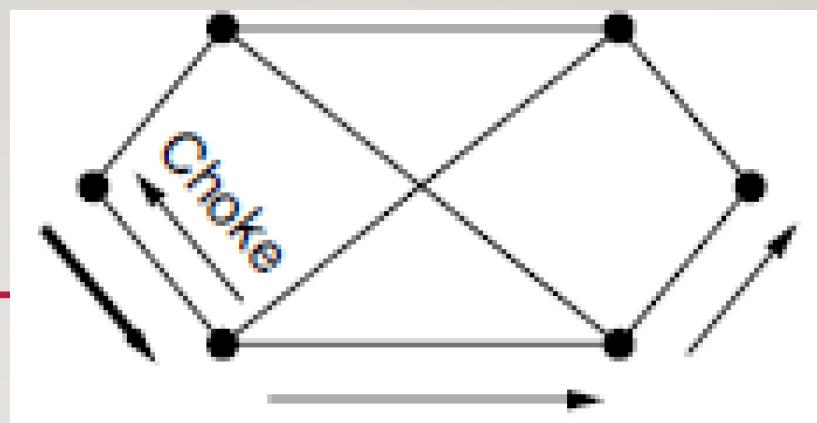
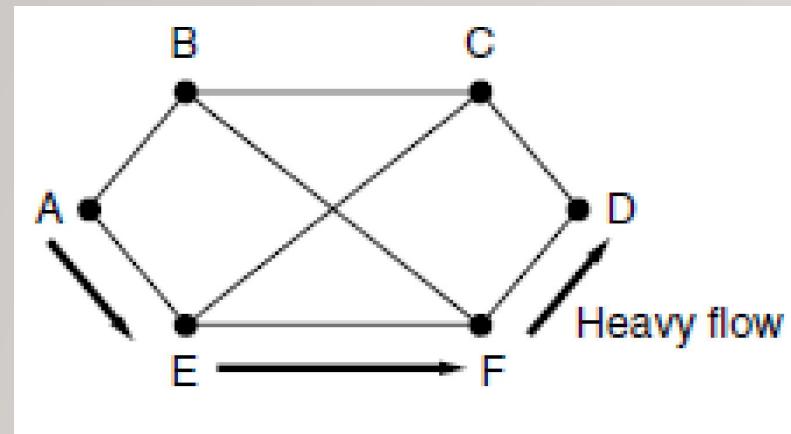
- Instead of generating additional packets to warn of congestion, a router can tag any packet it forwards (by setting a bit in the header field) to signal that it is experiencing congestion.
- When the network delivers the packet, the destination can note that there is congestion and inform the sender when it sends a reply packet.



# Approaches To Congestion Control

---

- Traffic throttling
  - Feedback Mechanisms
    - c) Hop by Hop backpressure:
      - Backpressure is a node-to-node congestion control technique that propagate in the opposite direction of data flow.
      - Choke packets take effect at every hop it passes through.
      - Provide quick relief at the point of congestion at the price of using up more buffers upstream.



# Approaches To Congestion Control

---

- Load Shedding

- Discard packets that the network cannot deliver.
- Question is which packets to drop?
- A good policy for choosing which packets to discard can help to prevent congestion collapse.
  - To implement an intelligent discard policy, applications must mark their packets into priority classes.
  - When packets have to be discarded, routers first drops packets from the lowest priority class.

# THANK YOU!!!

---

# COMPUTER NETWORKS

---

MS. JINCY J FERNANDEZ

ASST. PROF, CSE

RSET

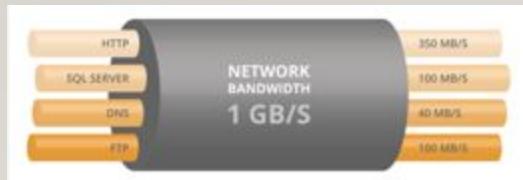
# Quality of Service- Introduction

---

- Strong performance guarantees are required from networks.
- Need to ensure **quality of service** in networks.
- Issues to deal??
  - What applications need from the network?
  - How to regulate traffic that enters the network?
  - How to reserve resources at routers to guarantee performance?
  - Whether the network can safely accept more traffic?

# QUALITY OF SERVICE

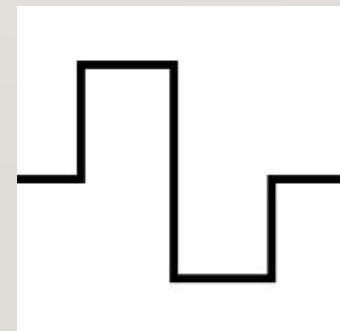
- Flow is a stream of packets from the source to the destination.
- Quality of Service (QoS) refers to the **capability of a network to provide better service** to selected network traffic/flow.
- The Quality of Service (QoS) a flow requires is characterized by four parameters:



**Bandwidth**



**Delay**



**Jitter**



**Reliability**

**h**

# Applications and their QOS Requirements

---

Application	Reliability	Delay	Jitter	Bandwidth
E-mail	High	Low	Low	Low
File transfer	High	Low	Low	Medium
Web access	High	Medium	Low	Medium
Remote login	High	Medium	Medium	Low
Audio on demand	Low	Low	High	Medium
Video on demand	Low	Low	High	High
Telephony	Low	High	High	Low
Videoconferencing	Low	High	High	High

# Techniques For Achieving Good QOS

---

- Overprovisioning
- Traffic Shaping
  - Leaky Bucket
  - Token Bucket
- Resource Reservation
- Proportional Routing
- Packet Scheduling
- Integrated Services
  - RSVP
- Differentiated Services
  - Expedited Forwarding
  - Assured Forwarding

# Overprovisioning

---

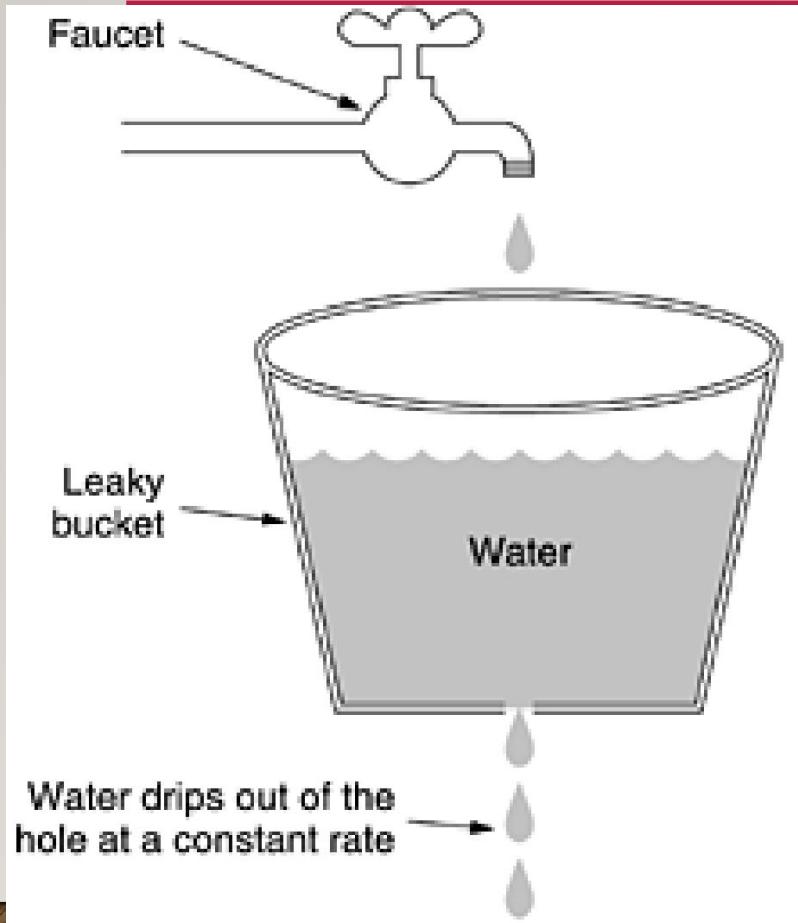
- Build a network with enough capacity for whatever traffic will be thrown at it.
- Provide **so much router capacity, buffer space and bandwidth** so that data can fly through easily.
- **Expensive in nature.**

# Traffic Shaping

---

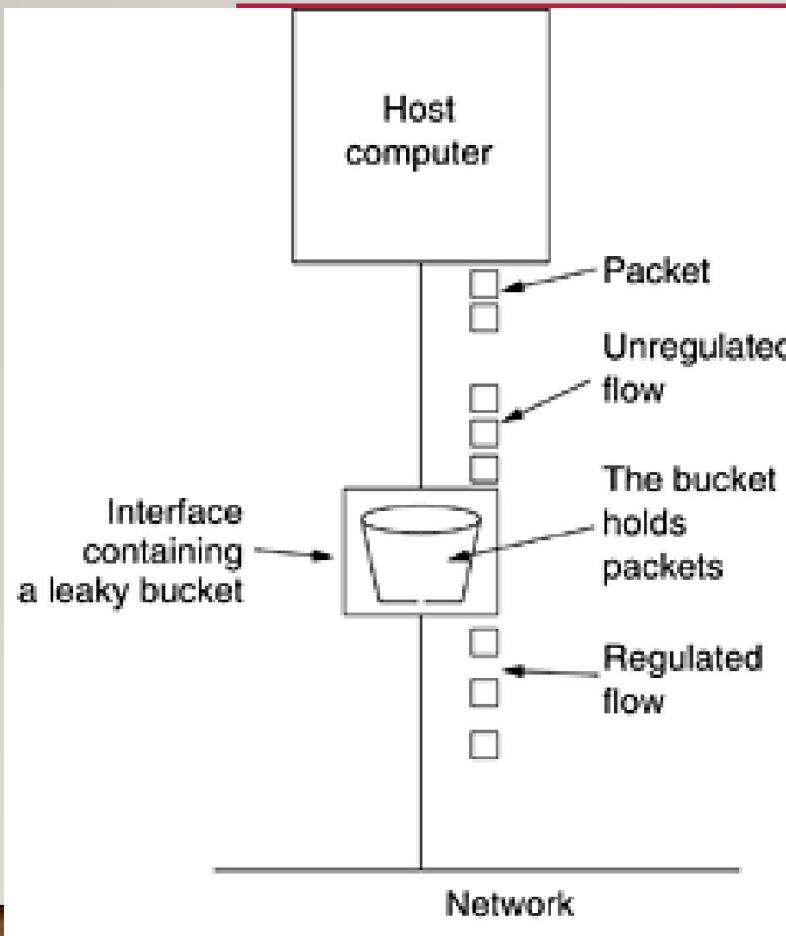
- Regulating the average rate of data transmission.
- When a connection is set up, the user and the subnet agree on a certain traffic pattern called **Service Level Agreement**.
- Carrier should monitor the traffic to ensure that the customer is following the agreement.
- Monitoring a traffic flow is called **Traffic Policing**.
- Shaping and Policing is important for real time data such as audio and video.

# Traffic Shaping- Leaky Bucket Algorithm



- No matter the rate at which water enters the bucket, the **outflow is at a constant rate,  $\rho$** , when there is water in the bucket.
- Once the bucket is full, any additional water entering it spills over the sides and is lost.
- Enforces a rigid output rate how much ever input is coming.

# Traffic Shaping- Leaky Bucket Algorithm



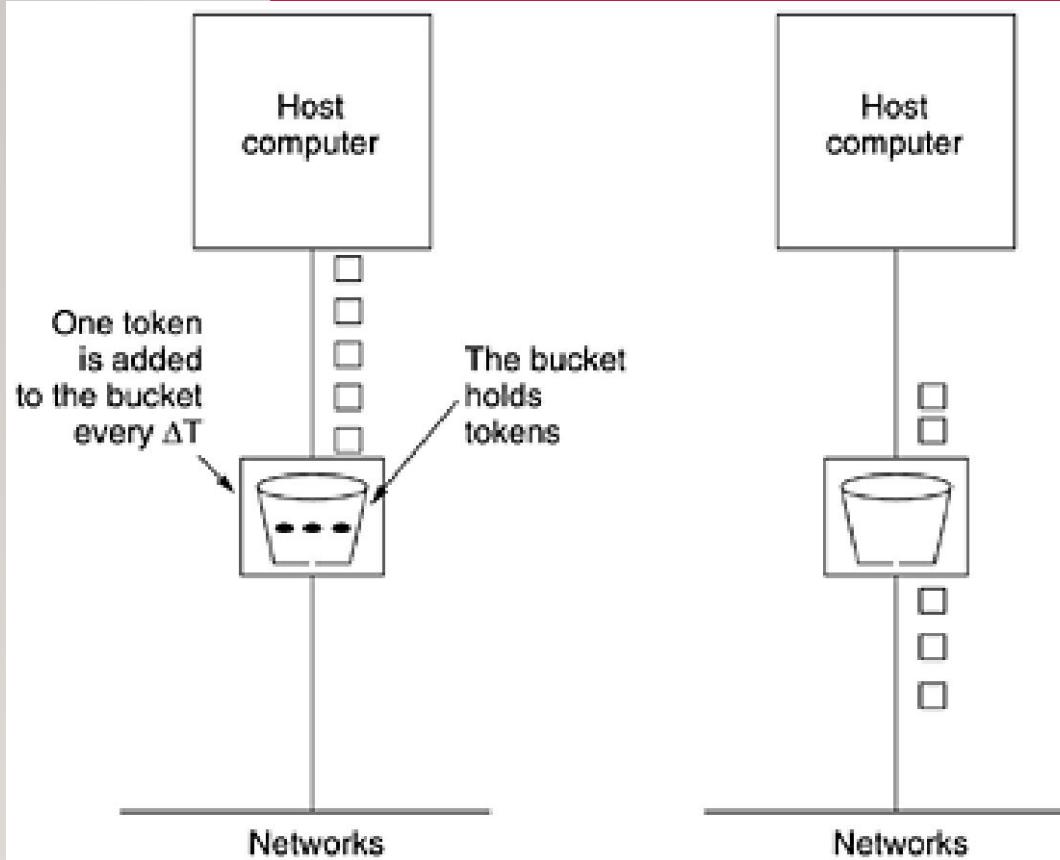
- Each host is connected to the network by an interface containing a leaky bucket (finite internal queue).
- To send a packet into the network, it must be possible to put more packets into the bucket.
- If a packet arrives when the bucket is full, the packet must either be queued until enough packets leak out to hold it or be discarded.
- This technique was proposed by Turner (1986) and is called the **leaky bucket algorithm**.
- Not good for bursty traffic.

# Traffic Shaping- Leaky Bucket Algorithm

---

- The host is allowed to put **one packet per clock tick** onto the network.
- An uneven flow of packets from the user processes inside the host **is turned into an even flow of packets onto the network**, smoothing out bursts and greatly reducing the chances of congestion.
- Works well when the packets are all the same size.
- When the packets are of **variable-sized**, it is often better to **allow a fixed number of bytes per tick**, rather than just one packet.
- **Byte counting leaky bucket algorithm.**

# Traffic Shaping- Token Bucket Algorithm



- This allows output rate to vary depending on size of burst.
- The leaky **bucket** holds tokens.
- The tokens are generated by a clock at the rate of one token every  $\Delta T$  sec.
- For a **packet** to be transmitted, it must capture and **destroy** one token.
- A minor variant is possible, in which **each token** represents the right to send not one packet, but  $k$  bytes.

# Comparison

<b>TOKEN BUCKET</b>	<b>LEAKY BUCKET</b>
Token dependent.	Token independent.
If bucket is full token are discarded, but not the packet.	If bucket is full packet or data is discarded.
Packets can only transmitted when there are enough token	Packets are transmitted continuously.
It allows large bursts to be sent faster rate after that constant rate	It sends the packet at constant rate
It saves token to send large bursts.	It does not save token.

# Resource Reservation

---

- Once a specific route for a flow is established, it becomes possible to reserve resources along that route to make sure the needed capacity is available.
- Resources that can be reserved are:
  - Bandwidth
  - Buffer space
  - CPU cycles

# Proportional Routing

---

- Most routing algorithms finds the best path for each destination and send all traffic to that destination over the best path.
- A different approach to provide a higher quality of service is to **split the traffic for each destination over multiple paths.**
- Divide the traffic equally or in proportion to the capacity of the outgoing links.

# Packet Scheduling

---

- If a router is handling multiple flows, there is a danger that **one flow will take over too much of its capacity and starve all the other flows.**
- A good scheduling technique treats the different flows in a fair and appropriate manner.
- **FIFO queuing algorithm**
  - Packets wait in a buffer (queue) until the node is ready to process them.

# Packet Scheduling

---

- Priority queuing algorithm
  - Packets are assigned to priority classes.
  - The packets in the highest priority queue are processed first.
  - The packets in the highest priority queue are processed last.
  - If there is a continuous flow in a high priority queue, the packets in the low priority queues will never have a chance to be processed? **Starvation.**

# Packet Scheduling

---

- Weighted Fair algorithm
  - Packets are assigned to different classes and admitted to different queues.
  - The queues are weighted based on the priority of the queues; higher priority means a higher weight.
  - Process the packets in each queue in a round robin fashion with the number of packets selected from each queue based on the corresponding weight.
  - Starvation is avoided.

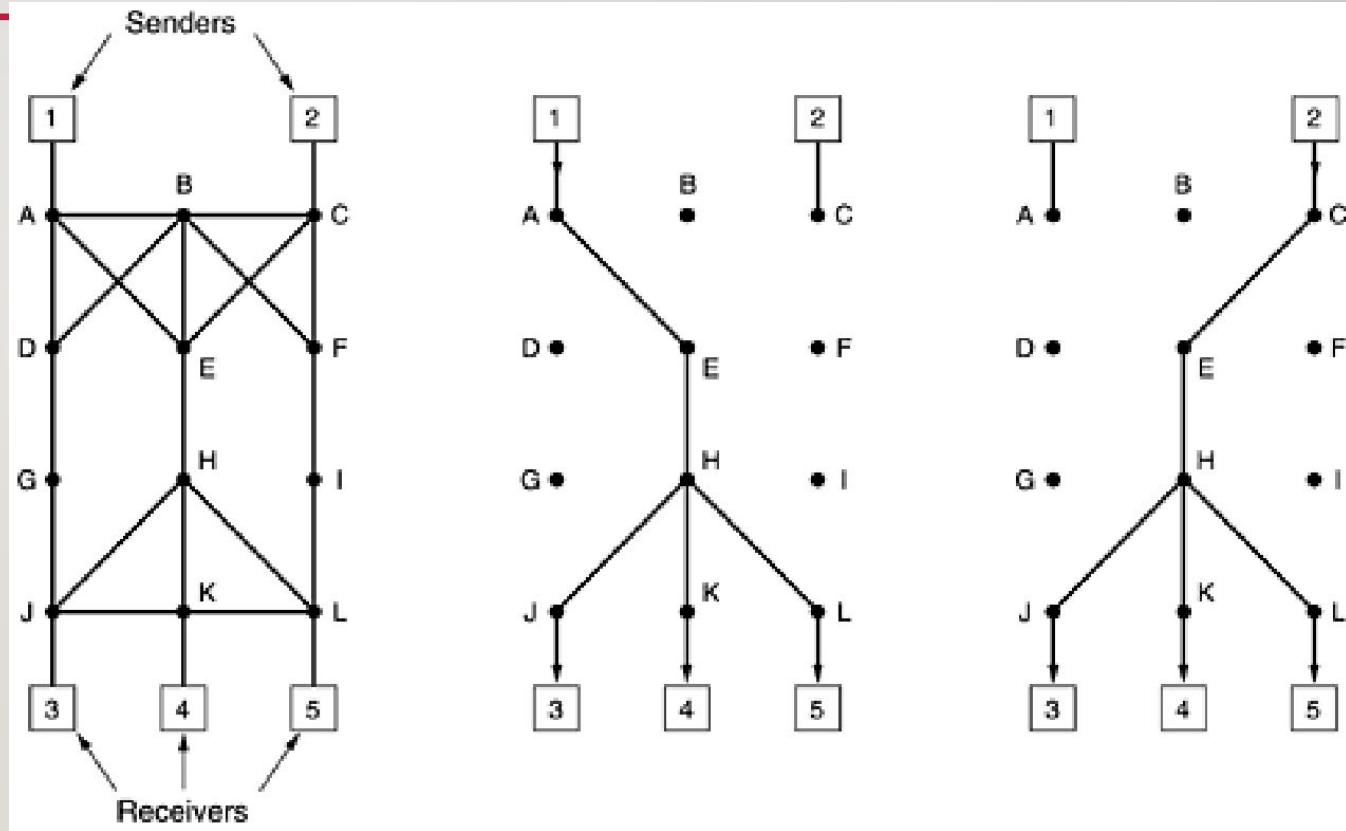
# INTEGRATED SERVICES

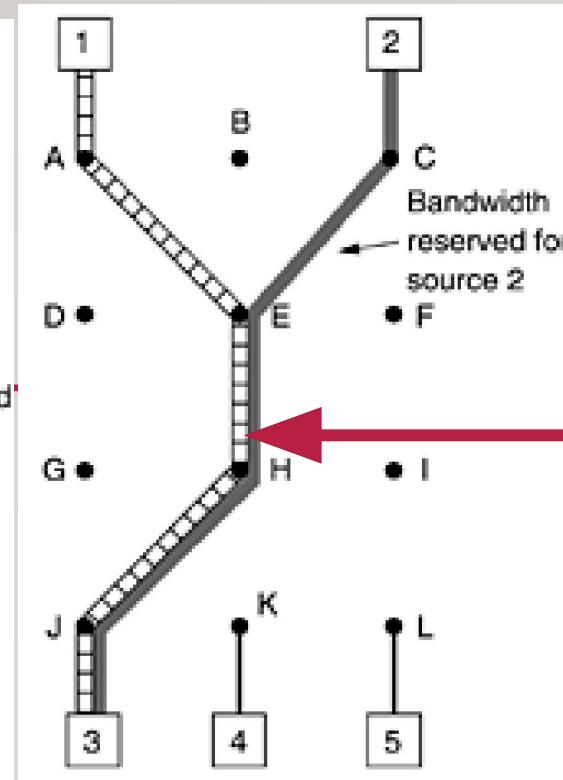
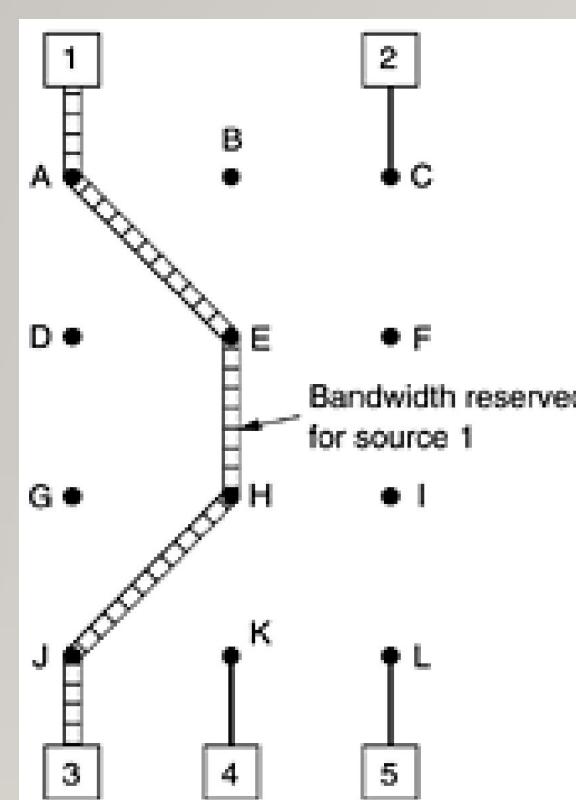
---

- Architecture for streaming multimedia.
- Flow Based Algorithms or Integrated Services.
- Advance setup required to establish each flow.
- For unicast and multicast applications.
- Resource reSerVation Protocol (RSVP) is the IETF architecture for the integrated services architecture.
  - Allows multiple senders to transmit to multiple groups of receivers.
  - Permits individual receivers to switch channels freely.
  - Optimizes bandwidth eliminating congestion.

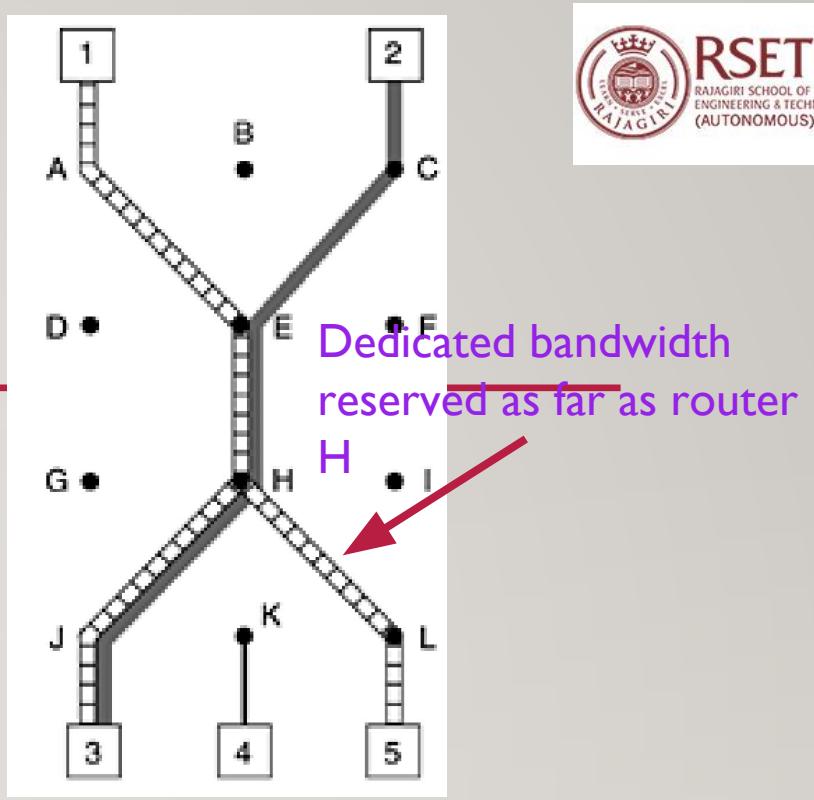
# RSVP

- Each group is assigned a group address.
- To send to a group, the sender put the group's address in its packets.
- The standard multicast routing algorithm builds a spanning tree covering all group members.





Two separate channels are needed from host 3 to router E, because two independent streams are being transmitted



- Receivers can send a **reservation message** to the sender
- The message is forwarded using the reverse path forwarding algorithm.
- At each hop, the router notes the reservation and reserves the necessary bandwidth.

# DIFFERENTIATED SERVICES

---

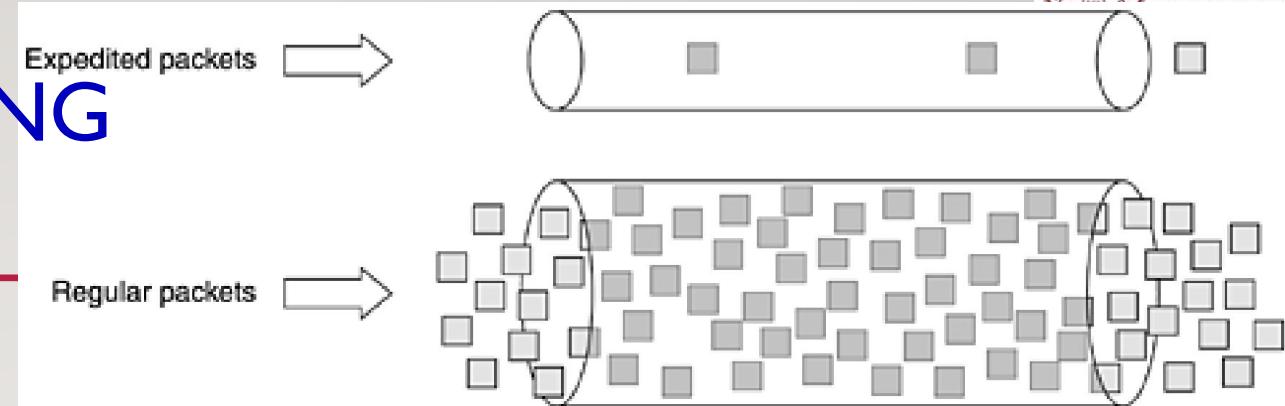
- Architecture for **class based QoS** and not flow based.
- Differentiated Services (DS) can be offered by a **set of routers** forming an **administrative domain** eg: ISP.
- The administration defines a set of **service classes** with **corresponding forwarding rules**.
- If a customer signs up for a DS, the customer packets may carry a **Type of Service** field in them.
- Better services may be provided to some classes compared to others.
- Traffic within the same class need to follow shaping. (leaky or token bucket).

# ADVANTAGES

---

- No advance setup needed.
- No resource reservation required.
- No time consuming end to end negotiation for each flow.
  
- So easy to implement compared to integrated services.

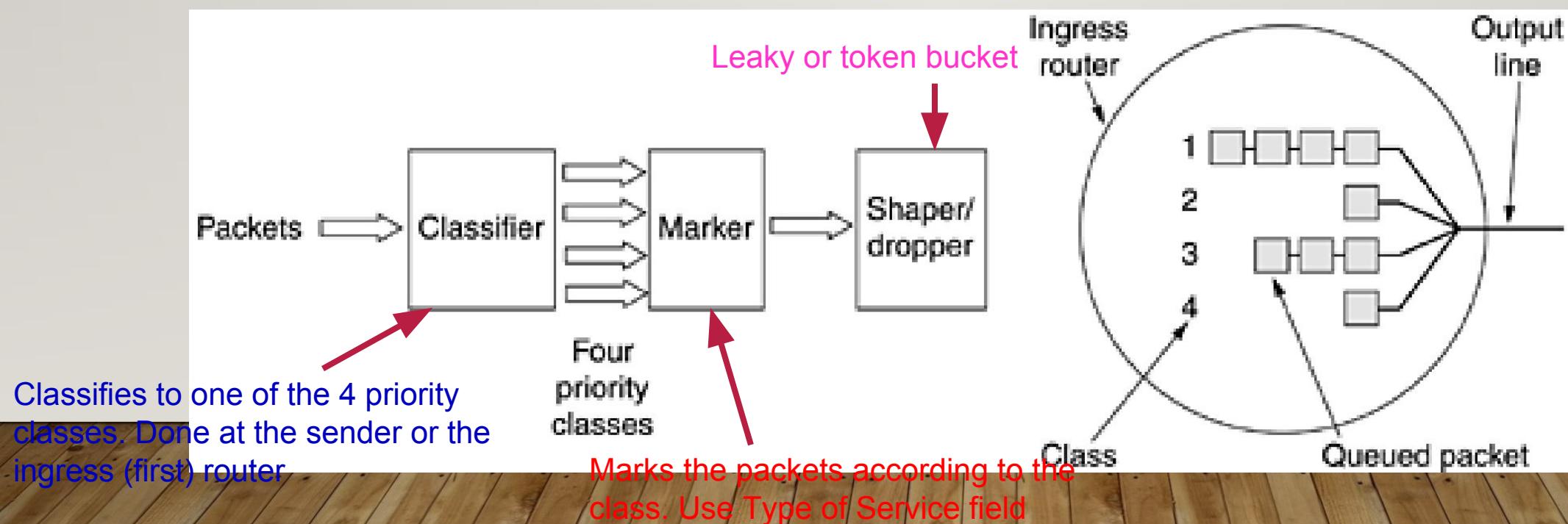
# EXPEDITED FORWARDING



- Two classes of services are available:
  - Regular
  - Expedited
- Majority of the traffic (90%) is regular.
- Small fraction of the packets are expedited (10%).
- Expedited packets transit the subnet as though no other packets are present.
- 20% of the bandwidth may be dedicated for expedited traffic.
- Only 1 physical line is present (implemented as 2 queues).

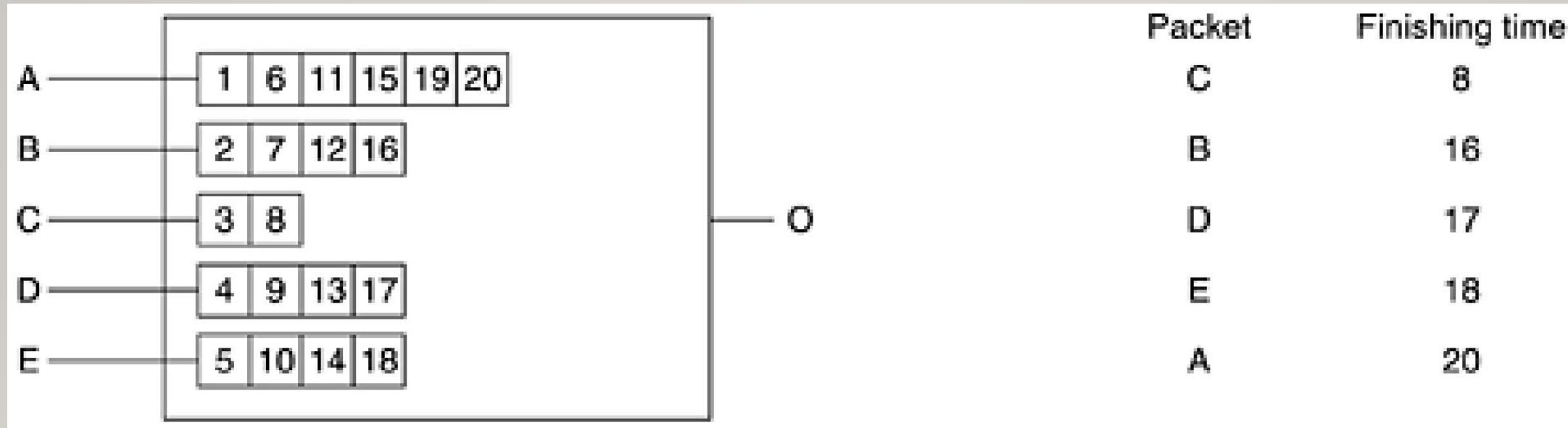
# ASSURED FORWARDING

- Four priority classes – each with its own resources.
- Three discard policies (low, medium, high).
- Together 12 service classes.



**THANK YOU!!!**

---



- The algorithm gives more bandwidth to hosts that use large packets than to hosts that use small packets.
  - Simulate Byte-by-byte round robin, instead of a packet-by-packet round robin.
  - Packets are sorted in order of finishing and sent in that order.

# COMPUTER NETWORKS

## MODULE 4.I

---

MR. SANDY JOSEPH

ASST. PROF, CSE

RSET

# Network Layer in the Internet -10 principles

---

1. Make sure it works: Do not finalize the design or standard until multiple prototypes have successfully communicated with each other.
2. Keep it simple: use the simplest solution.
3. Make clear choices: Choose one out of several ways of doing the same thing.
4. Exploit modularity: use of protocol stacks, each of its layers is independent of all the other ones.
5. Expect heterogeneity: possibility of different types of hardware, transmission facilities and applications can occur on large network. Network design must be simple, general and flexible.

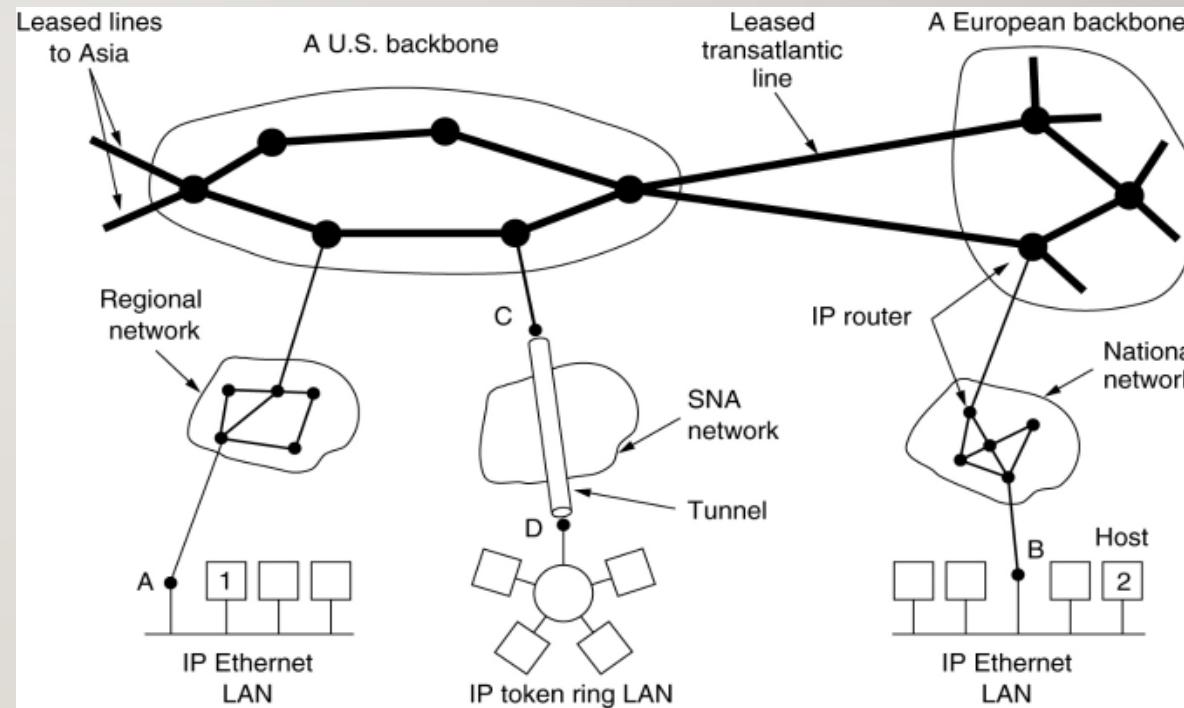
# Network Layer in the Internet -10 principles

---

6. Avoid static options and parameters: if parameters are unavoidable, it is best to have the sender and receiver negotiate a value rather than defining fixed choices.
7. Look for a good design; it need not be perfect.
8. Be strict when sending and tolerant when receiving: send only packets that comply with the standards, but expect packets that may not be fully conformant.
9. Think about scalability: on networks with billions of users, load must be spread as evenly as possible over the available resources.
10. Consider performance and cost: if a network has poor performance or outrageous costs, no one will use it.

# Internet- Collection Of Subnets

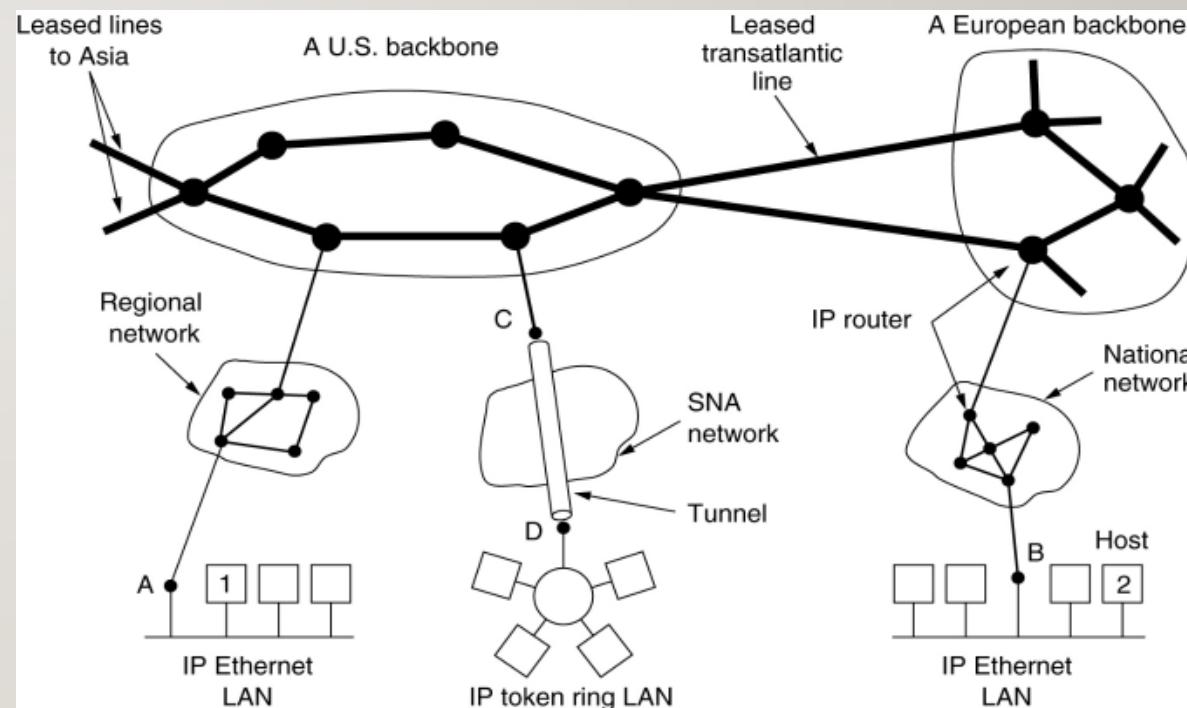
- Internet? collection of networks or ASes (Autonomous Systems) that are interconnected.
- An autonomous system (AS) is a **very large network or group of networks with a single routing policy**.
- Each AS is assigned a unique ASN, which is a number that identifies the AS.
- ASNs, are unique 16-bit numbers between 1 and 65534 or 32-bit numbers between 131072 and 4294967294.



# Internet- Collection Of Subnets

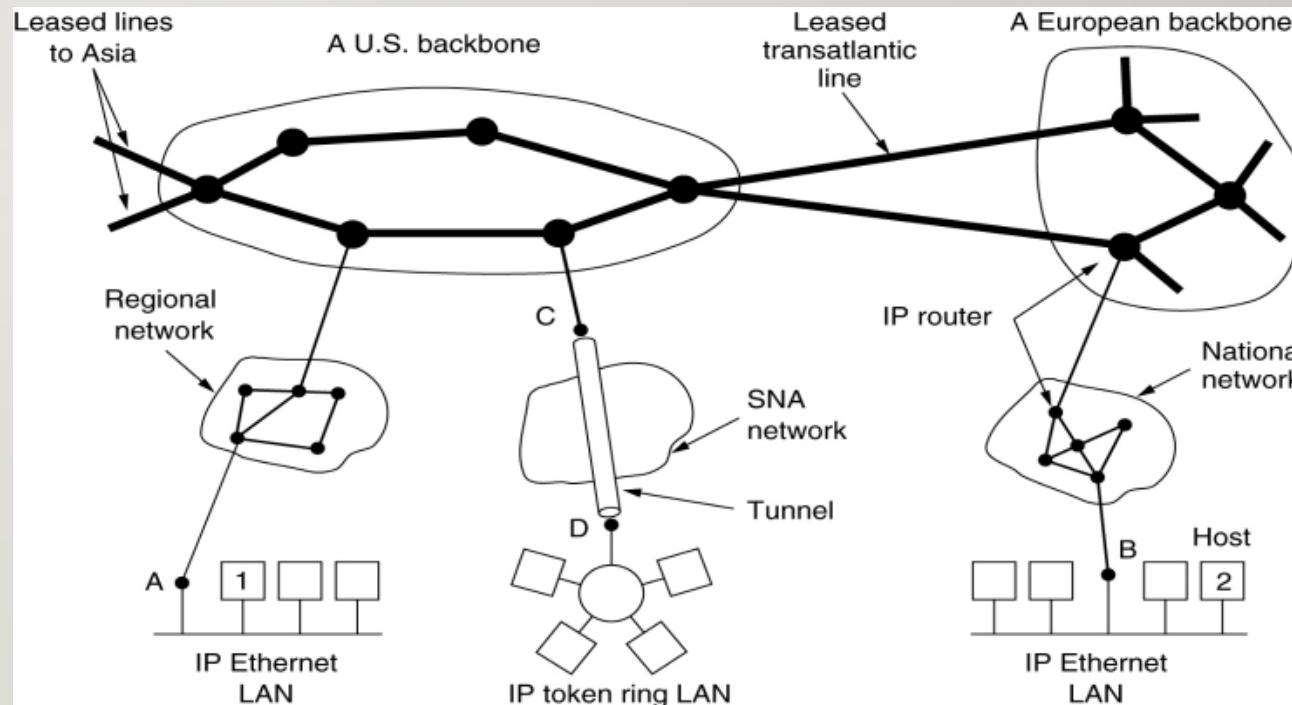
---

- ASNs are only required for external communications with inter-network.
- Internal routers and computers within an AS may not need to know that AS's number, since they are only communicating with devices within that AS.



# Internet- Collection Of Subnets

- **ISP (Internet Service Provider)** provide internet access to homes and businesses, data centers, and regional networks.
- An **Internet service provider (ISP)** is an organization that provides services for accessing, using, or participating on the internet.
- **IP (Internet Protocol)** ? The glue that holds the whole Internet together.



# IP-Internet Protocol

---

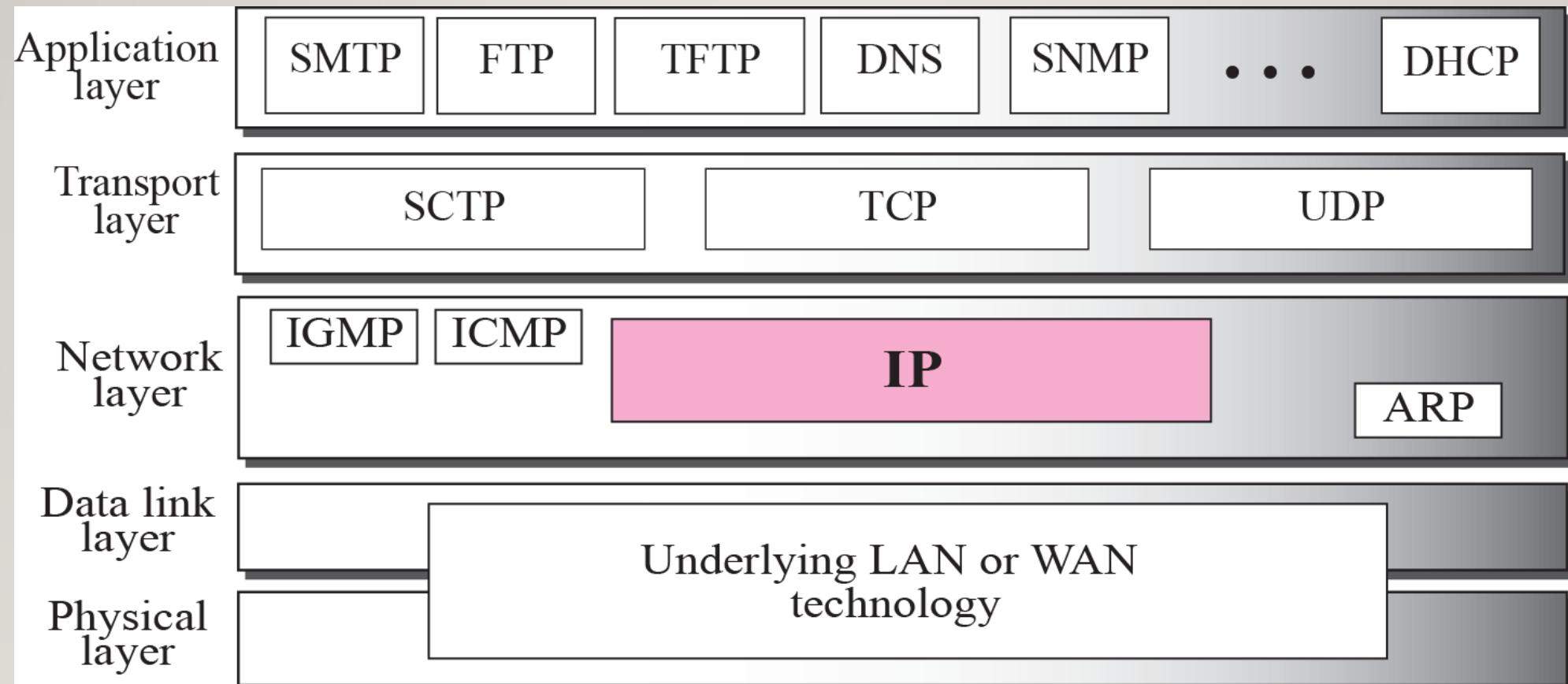
- The principal communications **protocol** in the Internet protocol suite for relaying datagrams across network boundaries.
- Its **routing** function enables internetworking, and essentially establishes the Internet.
- IP has the task of **delivering** packets from the source host to the destination host solely based on the IP addresses in the packet headers.
- IP defines **packet structures** that encapsulate the data to be delivered.

# IP-Internet Protocol

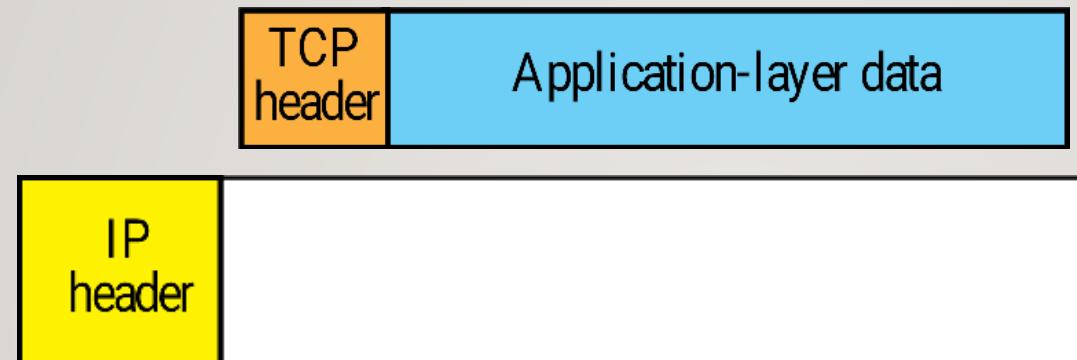
---

- It also defines **addressing methods** that are used to label the datagram with source and destination information.
  - Unicast Addressing : data is sent only to one destined host; 32-bit destination address.
  - Broadcast Addressing: packet is addressed to all the hosts in a network segment; Destination Address field contains a special broadcast address, i.e. 255.255.255.255.
  - Multicast Addressing: packet is addressed to a group of hosts in a network segment; Destination Address contains a special address which starts with 224.x.x.x.

# Position of IP in the TCP/IP Suite



# Encapsulation



- Packets in the network (internet) layer are called **datagrams**.
- A datagram is a variable-length packet consisting of two parts: **header and data**.
- The header is **20 to 60 bytes** in length.

# IPV4 Address

---

- An IPV4 address is a **32-bit address** that uniquely and universally defines the connection of a device to the internet.
- IP address is the address of the connection, not the host or router.
- **Address Space:** the total number of addresses used by the protocol.
- The **address space** of IPV4 is  $2^{32} = 4,294,967,296$  (more than 4 billion)

# IPV4 Address

---

- Notation:
  - **Binary notation** : represented as 32 bits; with space between octets; 10000000 11010000 00000010 10010111
  - **Dotted decimal notation**: each of the four bytes in decimal (0 to 255); each octet is separated by dot; 128.208.2.151.
  - **Hexadecimal notation**: in 8 hexadecimal digits; 80 0B 03 1F.

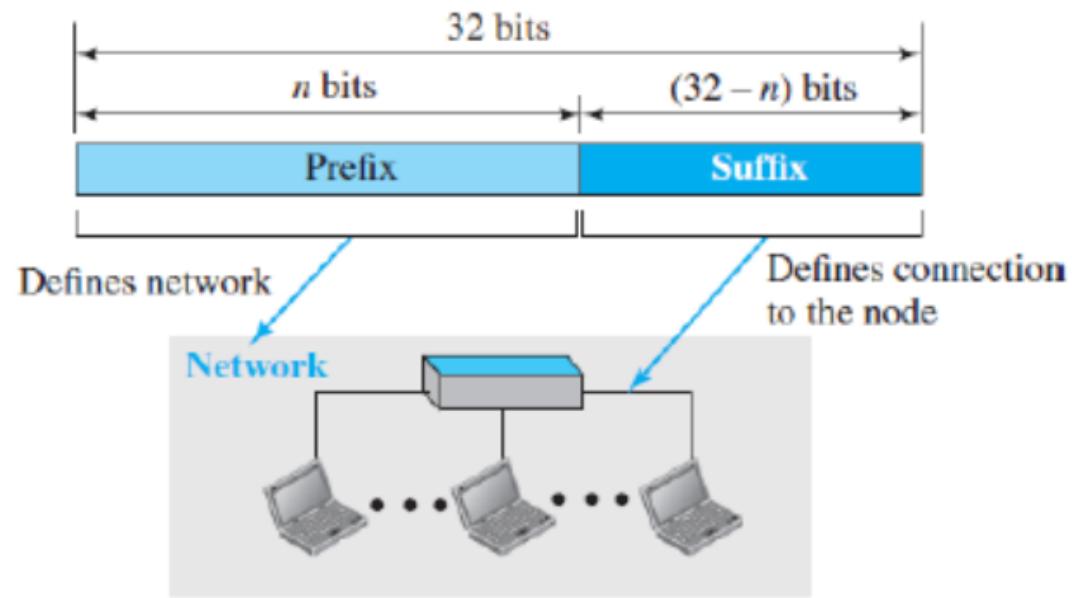
# IPV4 Address

---

- **Static IP Address:**
  - IP Address that once assigned to a network element always remains the same.
  - They are configured manually.
- **Dynamic IP Address:**
  - Dynamic IP Address is a temporarily assigned IP Address to a network element.
  - It can be assigned to a different device if it is not in use.
  - DHCP assigns dynamic IP addresses.

# IPV4 Address- Hierarchy in Addressing

- Hierarchical addressing system.
- Prefix: first part of the address defines the network.
- Suffix: defines the host (connection of a device to the internet).
- Prefix: defines the network; can be fixed length (Classful addressing), or variable length (Classless addressing).



# IPV4 Address- Classful Addressing

---

- Whole address space is divided into 5 classes (A, B, C, D, E)

Class	HOB	NET ID Bits	Host ID Bits	No of Networks	Host Per Network	Start Address	End Address
Class A	0	8	24	$2^7=128$	$2^{24}=16,777,216$	0.0.0.0	127.255.255.255
Class B	10	16	16	$2^{14}=16,384$	$2^{16}=65,536$	128.0.0.0	191.255.255.255
Class C	110	24	8	$2^{21}=2,097,152$	$2^8=256$	192.0.0.0	223.255.255.255
Class D	1110	-	-	-	-	224.0.0.0	239.255.255.255
Class E	1111	-	-	-	-	240.0.0.0	255.255.255.255

# IPV4 Address- Subnetting

---

- IP addresses are the identity of the device in the network.
- **Subnet Mask**
  - Subnet Mask helps to extract the Network ID and the Host from an IP Address
  - Separate 32-bit pattern to define the network and host portion of an address.
  - To know who are the neighbors in the network.
  - Classes A, B, C are accompanied with default subnet mask.

# IPV4 Address- Subnetting

---

- **Subnet Mask**
  - Does not contain network or host portion of an IPV4 address; tells where to look for these portions in a given IPV4 address.
  - 255 in octet means network portion; remaining octets are host portion.
  - A higher-class subnet mask can be used for lower class.
  - If the IP address in binary is ANDed with its Subnet Mask, the result yields the Network address.

# IPV4 Address- Subnetting

---

- Question 1: Check whether the following addresses belong to the same network or not.
  - 10.10.10.1 and 10.10.20.16 (subnet mask: 255.0.0.0)
  - 10.10.10.1 and 10.10.20.16 (subnet mask: 255.255.255.0)
  - 172.16.200.1 and 172.16.165.2 (subnet mask: 255.255.0.0)
  - 172.16.200.1 and 172.16.165.2 (subnet mask: 255.255.255.0)

# IPV4 Address- Subnetting

---

- Question 2: How many bits are allocated for Network ID and Host ID in 23.192.157.234 address?
- Question 3: What is the network ID of the IP Address 230.100.123.70?

# IPV4 Address- Subnetting

---

- **Advantages**
  - Subnetting reduces broadcast volume and hence reduces network traffic.
  - The network security may easily be utilized amongst sub-networks instead of using it on the entire network.
  - Subnetworks are simple to handle and maintain.
- **Disadvantages**
  - require a qualified administrator to perform the subnetting process.
  - subnetting process is quite expensive.

# IPV4 Address- Address Depletion

---

- Disadvantage of classful addressing.
- Means inefficient address use.
- Class A? Maximum 128 organizations with maximum 16777216 nodes per network.
- Only few organizations were this much large.

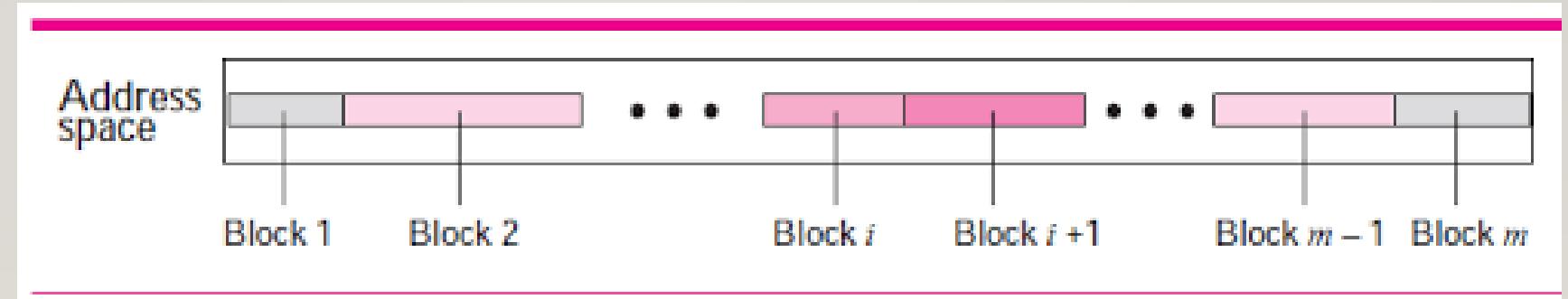
# IPV4 Address- Classless Addressing

---

- Formal Name is Classless Inter-Domain Routing (CIDR).
- With the growth of internet, large address space is needed. Allow service providers to allocate IPV4 addresses on any address bit boundary instead of classes A, B, C.
- Number of addresses in a block needs to be a power of 2.
- Classless addressing is possible with the help of subnetting.

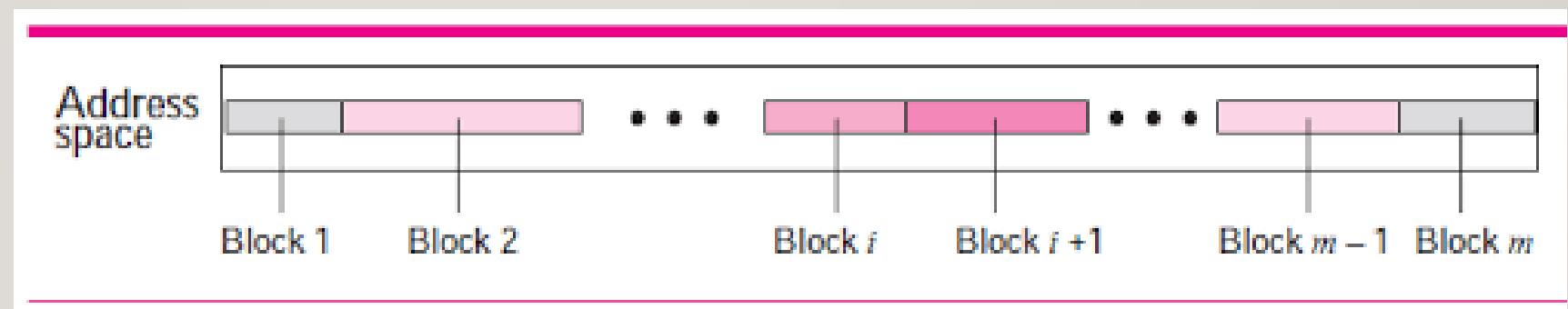
# IPV4 Address- Classless Addressing

- Whole address space is divided into variable length blocks.
  - Prefix: defines the network; variable length; ranges from 0 to 32.
  - Suffix: defines the host.



# IPV4 Address- Classless Addressing

- Whole address space is divided into variable length blocks.
  - Prefix: defines the network; variable length; ranges from 0 to 32.
  - Suffix: defines the host.
- How to find prefix length?
  - Slash Notation



# IPv4 Address- Classless Addressing

---

- **Slash Notation (/n)**
  - Slash notation is a compact way to write an IPv4 subnet mask.
  - **Format:** write the IP address, a forward slash (/), and the subnet mask number.
  - To find the subnet mask number:
    - Convert the decimal representation of the subnet mask to a binary.
    - Count each “1” in the subnet mask. The total is the subnet mask number.
  - E.g. 192.168.42.23 with a subnet mask of 255.255.255.0 ↳ 192.168.42.23/24

# IPV4 Address- Classless Addressing

---

- An address in classless addressing does not define the block or network to which the address belongs to with out knowing prefix length.

# IPV4 Address- Classless Addressing – Extracting information from an address

---

- Number of addresses in a block,  $N = 2^{32-n}$
- To find first address, keep the 'n' leftmost bits and set the (32-n) rightmost bits all to 0s.
- To find last address, keep the 'n' leftmost bits and set the (32-n) rightmost bits all to 1s.

# IPV4 Address- Classless Addressing – Extracting information from an address

---

- Q1. A classless address is given as 167.199.170.82/27. Find the number of addresses, first address, and last address?
  - Number of addresses= 32
  - First address=10100111 11000111 10101010 01000000 (167.199.170.64/27)
  - Last address= 10100111 11000111 10101010 01011111 (167.199.170.95/27)

# IPV4 Address- Subnetting

---

- Number of valid hosts in a subnet

$$2^{32 - \text{network bits}} - 2$$

# IPV4 Address-Subnetting

---

- Qn.1 How many usable IP addresses can we get from a /19 subnet?  
Ans: 8190
- Qn. 2. How many usable IP addresses can we get from a /24 subnet?  
Ans: 254
- Qn. 3. how many usable IP addresses are there in 172.16.23.0 255.255.240.0?  
Ans: 255.255.240.0 is /20  
4094

# IPV4 Address-Subnetting

- 
- Qn.4 What is the minimum subnet size we need to accommodate 20 hosts?  
Ans: minimum number of host bits required is 5 bits  
/27
  - Qn. 5. What is the minimum subnet size we need to accommodate 127 hosts?  
Ans: /24
  - Qn. 6. What is the maximum number of IP addresses that can be assigned to hosts on a local subnet that uses the 255.255.255.224 subnet mask?  
Ans: 30

# IPV4 Address- Questions

- 
- Q7. Write the IP address 222.1.1.20 mask 255.255.255.192 in CIDR notation.  
Ans: 222.1.1.20/26
  - Q8. You have been allocated a class C network address of 211.1.1.0 and are using the default subnet mask of 255.255.255.0 how may hosts can you have?  
Ans:  $2^8 - 2 = 254$  hosts
  - Q9. You have an interface on a router with the IP address of 192.168.192.10/29. Including the router interface, how many hosts can have IP addresses on the LAN attached to the router interface?  
Ans: 8 hosts

# IPV4 Address- Questions

---

- Q10. An address space has a total of 1024 addresses. How many bits are needed to represent an address?  
Ans: 10
- Q11. 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?  
first address: 25.34.0.0  
Last address: 25.34.255.255
- Q12. An address space uses the three symbols 0, 1, and 2 to represent addresses. If each address is made of 10 symbols, how many addresses are available in this system?  
Ans:  $3^{10} = 59049$

# IPV4 Addresses –Subnet Id and Number of Subnets

---

- The subnet mask defines the size of the network.
- For more subnets, “borrow” bits from the host part.
- $n_{sub} = n + \log_2(S)$

$n_{sub}$  is the length of the subnet id  
S is the number of subnets

# IPV4 Address- Questions

Qn. 13. Find the subnet mask in each case:

a. 1024 subnets in class A

$$\begin{aligned} n_{\text{sub}} &= n + \log S \\ &= 8 + \log 1024 = 18 \end{aligned}$$

Subnet mask is  
255.255.192.0

b. 256 subnets in class B

$$n_{\text{sub}} = 16 + \log 256 = 24$$

Subnet mask is 255.255.255.0

c. 4 subnets in class C

$$n_{\text{sub}} = 24 + \log 4 = 26$$

Subnet mask is 255.255.255.192

# IPV4 Address- Questions

---

- Q14. An organization is granted the block 211.17.180.0/24. The administrator wants to create 32 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 32.
- Ans: a. Mask: /29 (24 + 5)
- b.  $2^{(32-29)} = 8$  Addresses per subnet
- c. First address in subnet 1: **211 . 17 . 180 . 0/29** and last address in Subnet 1 is **211 . 17 . 180 . 7/29**
- d. First address in subnet 32: **211 . 17 . 180 . 248/29** and last address in Subnet 32 is **211 . 17 . 180 . 255/29**

# IPV4 Address- Questions

---

- Q15. Find the range of addresses in the following blocks.
- a. 123.56.77.32/29
- b. 200.17.21.128/27
- c. 17.34.16.0/23
- d. 180.34.64.64/30
- Ans:
  - a. From: 123 . 56 . 77 . 32  
0 . 0 . 0 . 7
  - To: 123 . 56 . 77 . 39
  - b. From: 200 . 17 . 21 . 128  
0 . 0 . 0 . 31
  - To: 200 . 17 . 21 . 159
  - c. From: 17 . 34 . 16 . 0  
0 . 0 . 1 . 255
  - To: 17 . 34 . 17 . 255
  - d. From: 180 . 34 . 64 . 64  
0 . 0 . 0 . 3
  - To: 180 . 34 . 64 . 67

# IPV4 Address- What Subnet an Address Belongs to?

- Steps:
  - Determine how many network bits are in use.
  - Determine the maximum number of bits in the octet in which the subnet is
  - Determine the subnet block size by subtracting the network bits from the answer in step 2 above and raising to the power of 2.
  - To find the subnet to which the address belongs, start at 0 (in whatever octet the subnet is) and increase by the block size.

# IPV4 Address- What Subnet an Address Belongs to?

---

Qn.1. On what subnet does the 156.67.154.75/28 IP address belong

- No. of network bits =28.
- /28 is in the fourth octet and the maximum number of bits from the first to fourth octet is 32 bits.
- Subnet block size :  $2^{(32-28)} = 16$  subnet blocks.
- To find the subnet: 156.67.154.0/28      156.67.154.16/28  
  
156.67.154.32/28    156.67.154.48/28  
156.67.154.64/28    156.67.154.80/28  
156.67.154.96/28, etc
- 156.67.154.75 belongs to 156.67.154.64

# IPV4 Address- What Subnet an Address Belongs to?

---

Qn.2. What subnet does the address 77.81.23.45/19 belong?

- No. of network bits =19.
- /19 is in the third octet and the maximum number of bits from the first to third octet is 24 bits.
- Subnet block size :  $2^{(24-19)} = 32$  subnet blocks.
- To find the subnet: 77.81.0.0/19      77.81.32.0/19  
  
77.81.64.0/19, etc.
- Since .23 is greater than .0 and less than .32, then the 77.81.23.45/19 IP address belongs in the 77.81.0.0/19 subnet.

# IPV4 Address- What Subnet an Address Belongs to?

---

Qn.3. Do the following addresses belong on the same subnet?

10.21.45.137/13 and 10.23.156.198/13?

- Ans: same network

# IPV4 Address

---

Qn. 4. 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.
- 
- a- Mask = /17
  - b- 32768
  - c- 16.0.0.0 , 16.0.127.255
  - d- 16.249.128.0 , 16.249.255.255

# IPV4 Address- Subnetting

---

- **Steps:**
  - Identify the class of the IP address and note the default subnet mask.
  - Convert the default subnet mask into binary
  - Note the number of hosts required per subnet and find the subnet generator (SG) and octet position.
  - Generate the new subnet mask.
  - Use the SG and generate the network ranges (subnets) in the appropriate octet position.

# IPV4 Address- Subnetting

---

- Question 1: Subnet the IP address 216.21.5.0 into 30 hosts in each subnet.
  1. Class C – Default subnet mask ↗ 255.255.255.0
  2. Binary ↗ 11111111 11111111 11111111 00000000
  3. **No. of hosts per subnet** =30 (11110 in binary); 5 bits. Reserve 5 bits from the right (remaining bits will be 1's).  
11111111 11111111 11111111 11100000 (1<sup>st</sup> one from right side =32)  
**Subnet Generator (SG)** = 32 ; **Octet Position** (of SG)= 4.
  4. **New Subnet Mask** = 11111111 11111111 11111111 11100000 or 255.255.255.224 or /27

# IPV4 Address- Subnetting

---

- Question 1: Subnet the IP address 216.21.5.0 into 30 hosts in each subnet.

## 5. Subnets:

- 216.21.5.0 - 216.21.5.31
- 216.21.5.32 - 216.21.5.63
- 216.21.5.64 - 216.21.5.95
- 216.21.5.96 - 216.21.5.127

and so on

# IPV4 Address- Subnetting

- Question 2: Subnet the IP address 196.10.20.0 into 52 hosts in each subnet.
  1. Class C – Default subnet mask ↗ 255.255.255.0
  2. Binary ↗ 11111111 11111111 11111111 00000000
  3. **No. of hosts per subnet** =52 (110100 in binary); 6 bits. Reserve 6 bits from the right (remaining bits will be 1's).  
11111111 11111111 11111111 11000000 (1<sup>st</sup> one from right side =64)  
**Subnet Generator (SG)** = 64 ; **Octet Position** (of SG)= 4.
  4. **New Subnet Mask** = 11111111 11111111 11111111 11000000 or 255.255.255.192 or /26

# IPV4 Address- Subnetting

---

- Question 2: Subnet the IP address 196.10.20.0 into 52 hosts in each subnet.

## 5. Subnets:

196.10.20.0 - 196.10.20.63

196.10.20.64 - 196.10.20.127

196.10.20.128 - 196.10.20.191

196.10.20.192 - 196.10.20.255

and so on

# IPV4 Address- Subnetting

- Question 3: Subnet the IP address 150.15.0.0 into 500 hosts in each subnet.
  1. Class B – Default subnet mask ↗ 255.255.0.0
  2. Binary ↗ 11111111 11111111 00000000 00000000
  3. **No. of hosts per subnet** =500 (111110100 in binary); 9 bits. Reserve 9 bits from the right (remaining bits will be 1's).  
11111111 11111111 11111110 00000000 (1<sup>st</sup> one from right side =2)  
**Subnet Generator (SG)** = 2 ; **Octet Position** (of SG)= 3.
  4. **New Subnet Mask** = 11111111 11111111 11111110 00000000 or 255.255.254.0 or /23

# IPV4 Address- Subnetting

---

- Question 3: Subnet the IP address 150.15.0.0 into 500 hosts in each subnet.

## 5. Subnets:

150.15.0.0 - 150.15.1.255

150.15.2.0 - 150.15.3.255

150.15.4.0 - 150.15.5.255

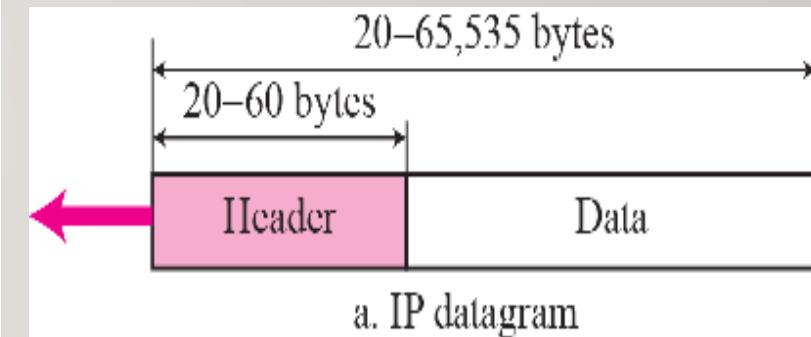
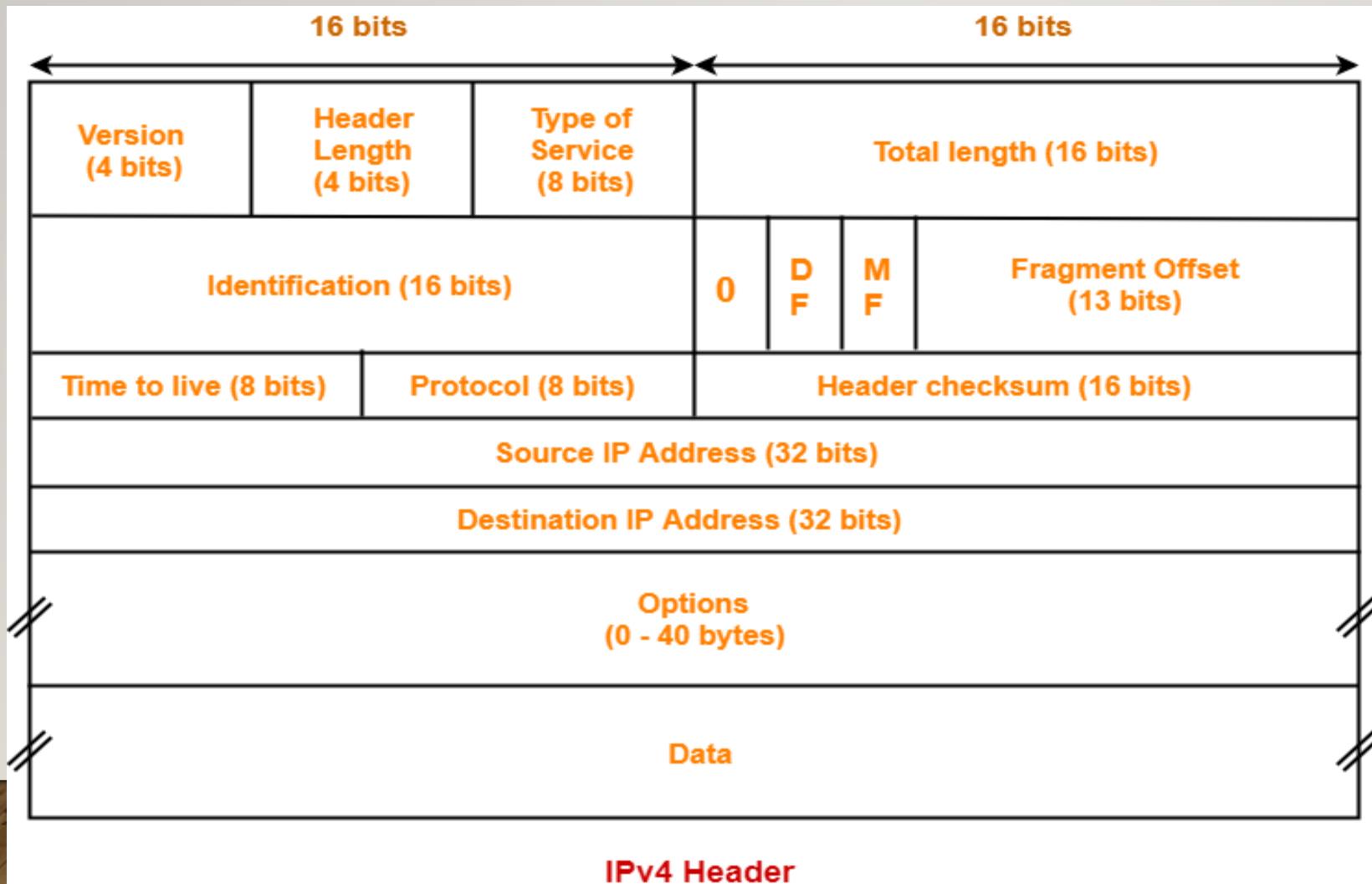
and so on

# IPV4 Header

---

- Provides a Connectionless, datagram service
- Datagram = Header + Data
- Header  $\approx$  20 – 60 bytes.
- Data  $\approx$  0-65515 bytes
- Size of Datagram  $\approx$  65535 bytes.

# IPv4 Header Format

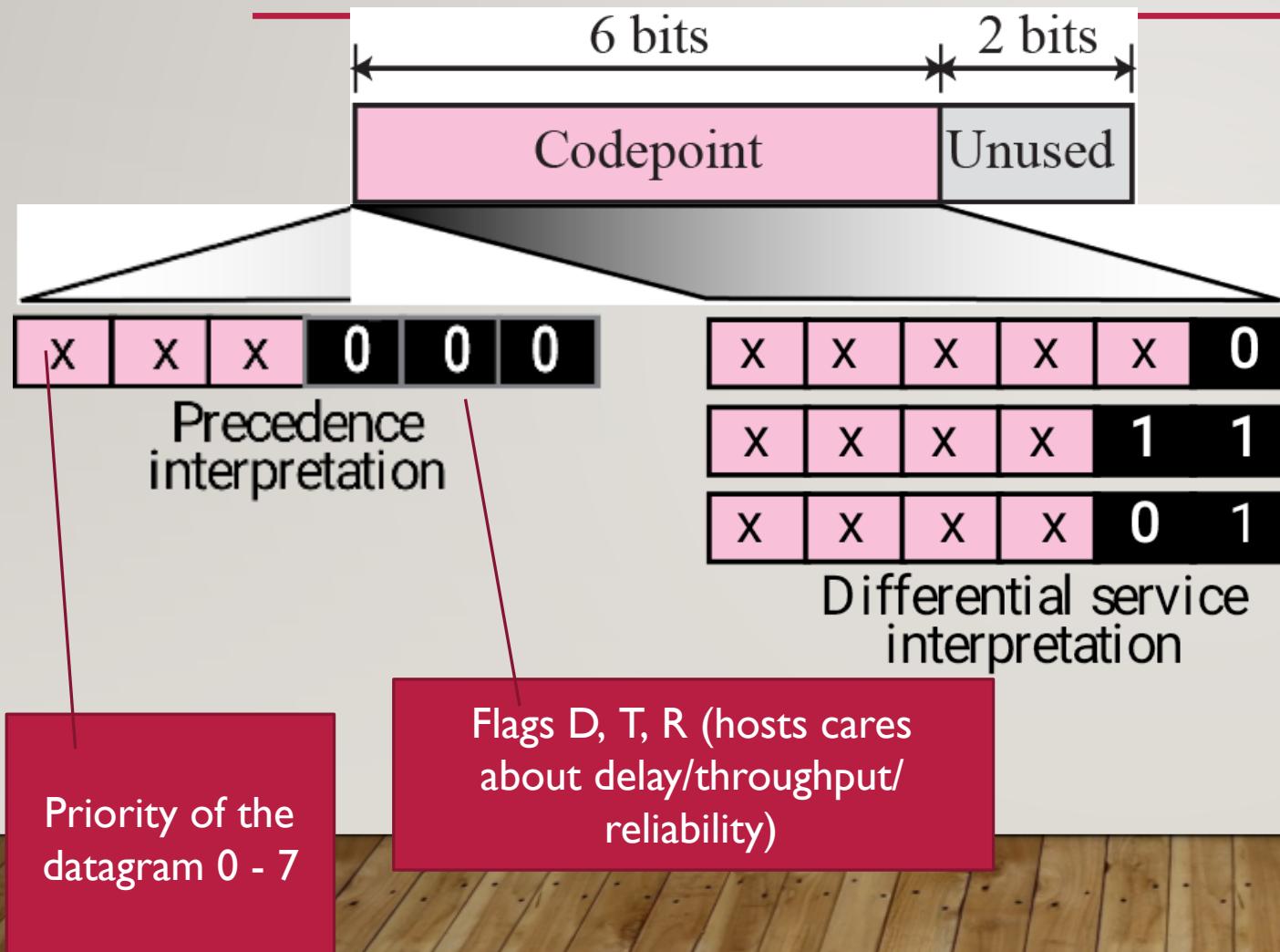


# IPV4 Header Format

---

- **Version (4 bits):** version of IP protocol – IPv4 or IPv6.
- **Header length (4 bits):** total length of the datagram header, in 4-byte words.
  - If header length is **20 bytes**, the value of the field is **5**
- **Service Type (8 bits):** defines a set of differentiated services.
  - First 3 bits? defines priority of the packets.
  - Next 3 bits? defines whether a host cares about delay, throughput and reliability.
  - Next 2 bits ? defines congestion notification information.

# IPV4 Header Format



**Table 7.1** Values for codepoints

Category	Codepoint	Assigning Authority
1	XXXXX0	Internet
2	XXXX11	Local
3	XXXX01	Temporary or experimental

# IPV4 Header Format

---

- **Total Length (16 bits):** defines the total length of the IP datagram in bytes.
  - Length of data = total length – header length
  - Total length is limited to 65,535 bytes (since the field length is 16 bits).
- **Identification (16 bits):** allow the destination to determine which packet a newly arrived fragment belongs to. All the fragments of a packet contains same Identification value.

# IPV4 Header Format

---

- **Flags (3 bits):**
  - First bit  $\oplus$  0 (unused bit).
  - Second bit  $\oplus$  DF  $\oplus$  **Don't Fragment**: order to the routers not to fragment the packet.
  - Third bit  $\oplus$  MF  $\oplus$  **More Fragments**: All fragments except the last one have this bit set. It is needed to know when all fragments of a datagram have arrived.

# IPV4 Header Format

---

- **Fragmentation offset (13 bits):** where in the current packet this fragment belongs. All fragments except the last one in a datagram must be a multiple of 8 bytes. With 13 bits, there is a maximum of 8192 fragments per datagram.
- **Time to live (8 bits):** counter used to limit packet lifetimes. Maximum lifetime of 255 sec.
  - Holds a timestamp (approximately twice the maximum number of routers between any two hosts) which is **decremented by each visited router**.
  - Datagram is **discarded when the value becomes zero**.

# Fragmentation

---

- Maximum size of IP datagram is 65,535 bytes.
  - But the data link layer protocol generally imposes a limit that is much smaller.
- For example:
  - Ethernet frames have a maximum payload of 1500 bytes.
  - IP datagrams encapsulated in Ethernet frame cannot be longer than 1500 bytes.
- The limit on the maximum IP datagram size, imposed by the data link protocol is called **Maximum Transmission Unit (MTU)**.

# Fragmentation

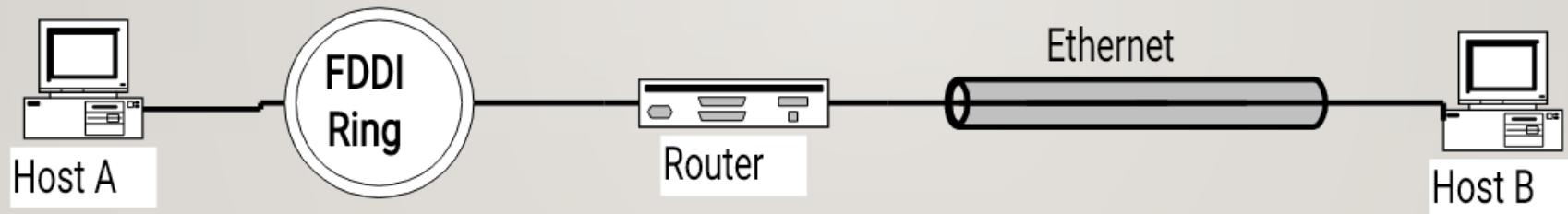
---

- MTUs for various data link layers protocols:
  - Ethernet : 1500
  - FDDI : 4352
  - PPP : 296
  - 802.3 : 1492
  - ATM AAL5 : 9180
  - 802.5 : 4464

# Fragmentation

---

- What if the size of an IP datagram exceeds the MTU?
- What if the route contains networks with different MTUs?



- IP datagram has to be divided to make it possible to pass through these networks – Fragmentation.

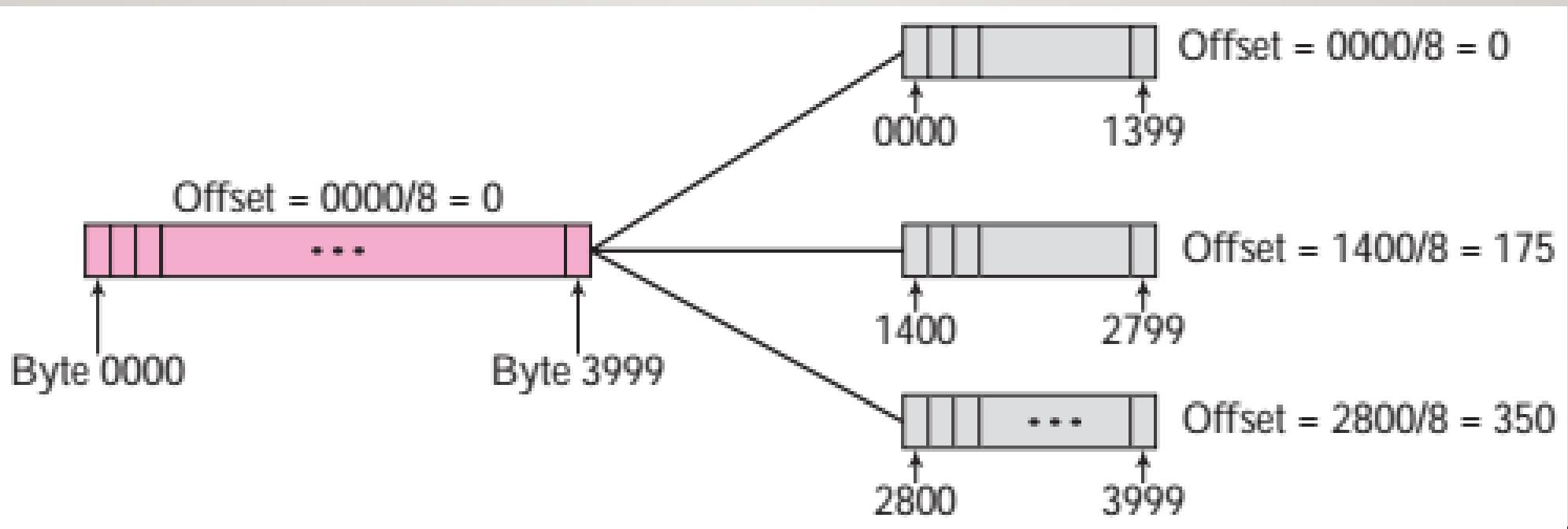
# Fragmentation

---

- Fragmentation can be done at the sender or at intermediate routers.
- The same datagram can be fragmented several times.
- Reassembly of original datagram is only done at destination host.

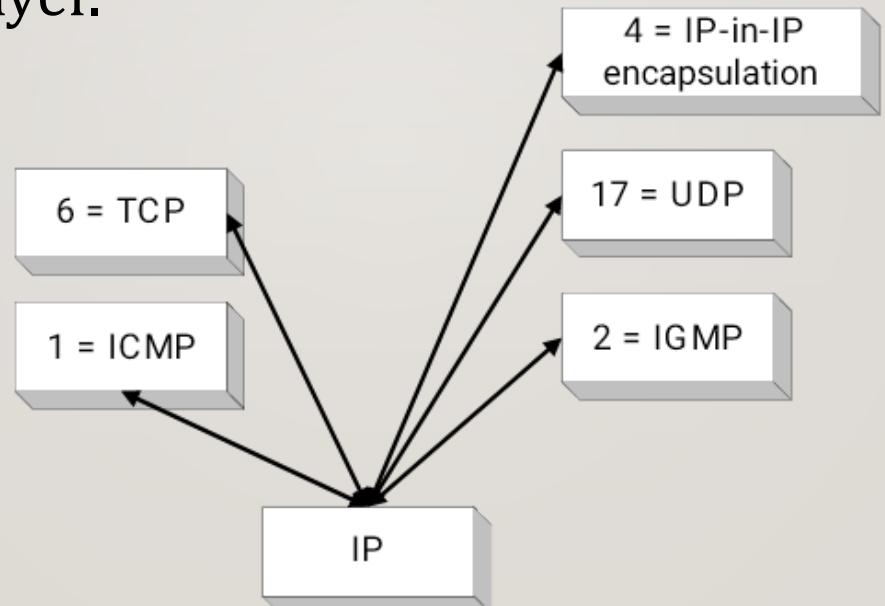
# Fragmentation

- **Fragmentation offset (13 bits):** Offset of the payload of the current fragment in the original datagram



# IPV4 Header Format

- **Protocol (8 bits):** defines the higher-level protocol that uses the services of the IP layer.



# IPV4 Header Format

---

- **Header checksum (16 bits)**
  - IP is not a reliable protocol; it does not check whether the payload is corrupted during transmission.
  - Used to verify the header. It is recomputed at every router.
- **Source Address (32 bits)**
  - Indicate the IP address of the source machine.
- **Destination Address (32 bits)**
  - Indicate the IP address of the destination machine.

# IPV4 Header Format

---

- **Options + Padding (0 to more)**

- Options can be used for network testing and debugging.
- It provides an escape to allow subsequent versions of the protocol to include information not present in the original design.

Option	Description
Security	Specifies how secret the datagram is
Strict source routing	Gives the complete path to be followed
Loose source routing	Gives a list of routers not to be missed
Record route	Makes each router append its IP address
Timestamp	Makes each router append its address and timestamp

# Questions

---

- Q1. An IPV4 packet has arrived with the first 8 bits as  $(01000010)_2$  . The receiver discards the packet. Why?

[Ans: invalid header length]

- Q2. In an IPV4 packet, the value of HLEN is  $(1000)_2$ . How many bytes of options are being carried by this packet?

[Ans: 12 bytes]

- Q3. In an IPV4 packet, the value of HLEN is 5 and the value of the total length field is  $(0028)_{16}$ . How many bytes of data are being carried by this packet?

[Ans: Total length= 40 bytes, Data size = 20 bytes]

# Questions

---

- Q4. A packet has arrived with an MF bit value of 0. Is this the first fragment, the last fragment, or a middle fragment?

[Ans: Last fragment]

- Q5. A packet has arrived with an MF bit value of 1 and a fragmentation offset value of 0. Is this the first fragment, the last fragment, or a middle fragment?

[Ans: First fragment]

# Questions

---

- Q6. A packet has arrived in which the offset value is 100. What is the number of the first byte?

[Ans: 800]

- Q7. 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 ?

[Ans: first byte number= 800,

80 bytes in the datagram, last byte number =879]

# Private Address

---

- IP addresses are publicly registered with the **Network Information Center (NIC)** to avoid address conflicts.
- Devices that need to be publicly identified, must have a globally unique IP address and are assigned a public IP address.
- Devices that do not require public access (network printer) may be assigned a private IP address.
- For organizations to freely assign private IP addresses, NIC has reserved certain address blocks for private use.

Class	Range of Private Address
Class A	10.0.0.0 to 10.255.255.255
Class B	172.16.0.0 to 172.31.255.255

# Special Addresses

# THANK YOU!!!

---

# COMPUTER NETWORKS

## MODULE 4.2

---

MS. JINCY J. FERNANDEZ

ASST. PROF, CSE

RSET

# WHY ICMP?

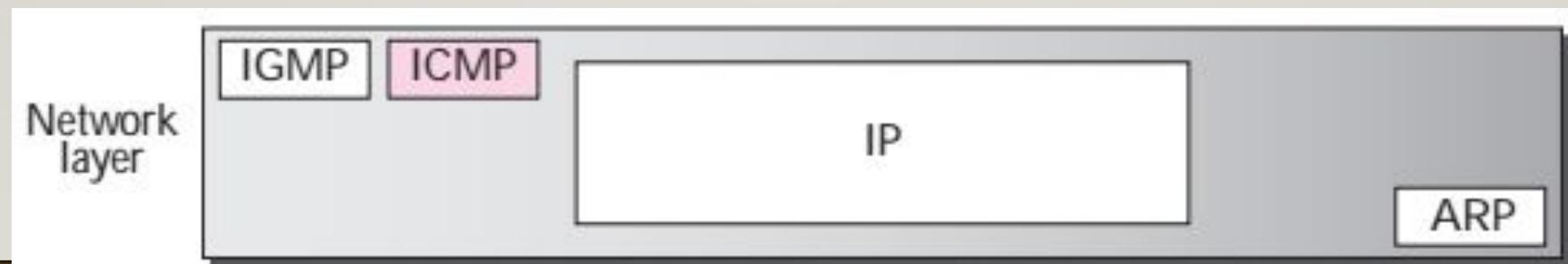
---

- The IP protocol has no error-reporting or error correcting mechanism.
- What happens if a router must discard a datagram because it cannot find a router to the final destination, or because the time-to-live field has a zero value?
- What happens if the final destination host must discard all fragments of a datagram because it has not received all fragments within a predetermined time limit?
- IP protocol has no built-in mechanism to notify the original host.

# Introduction to ICMP

---

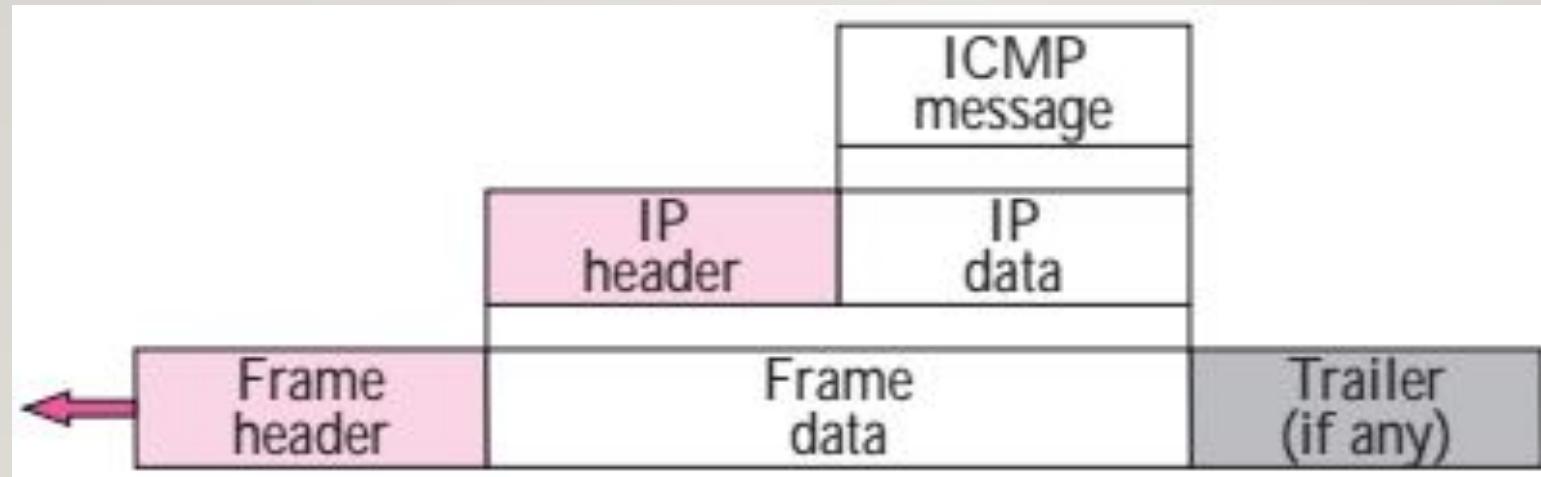
- Internet Control Message Protocol (ICMP)
- Companion to the IP protocol.
- Designed to compensate for the deficiencies of the IP protocol:
  - No error reporting or error correcting mechanism.
  - No mechanism for host and management queries.



# Features

---

- ICMP messages are encapsulated within IP datagrams.
- Value in the protocol field is set to 1 for an ICMP message.

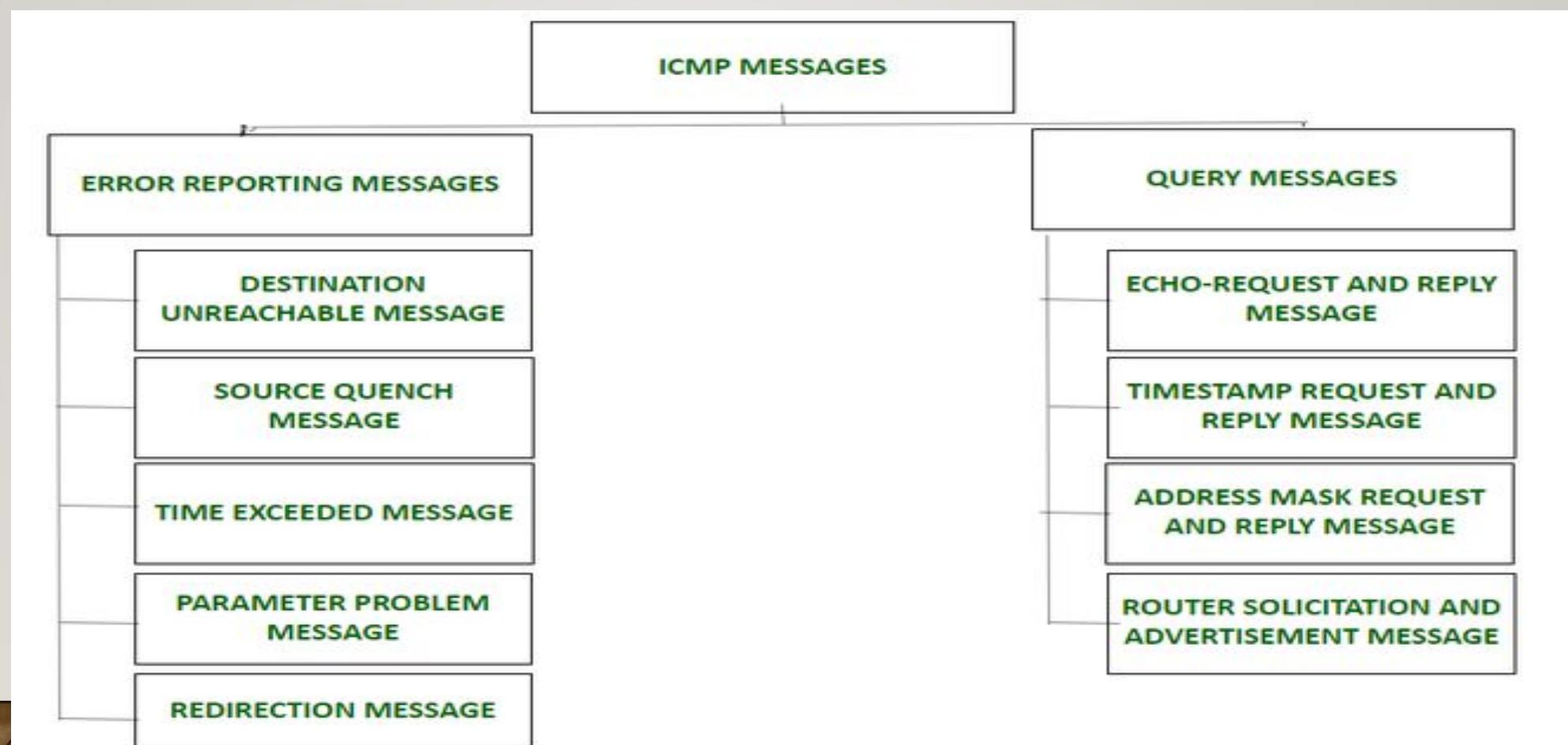


# ICMP Messages

---

- Divided into two categories:
- Error reporting messages
  - Reports problems that a router or a host (destination) may encounter when it processes an IP packet.
- Query messages
  - Occurs in pairs.
  - Helps a host or a network manager get specific information from a router or another host.

# ICMP Messages



# Message Format

- 8-byte header and a variable size data section.

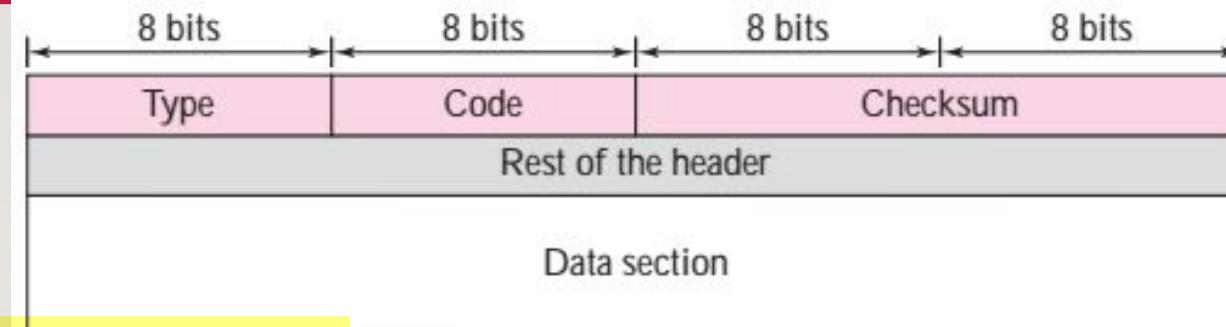
- **ICMP Type:** defines the type of the message.

- **Code field:** specifies the reason for the particular message type.

- **Checksum field:** for error control.

- **Rest of the header:** is specific for each message type.

- **Data section:** in error messages carries information for finding the original packet that had the error. In query messages, the data section carries extra information based on the type of the query.



# Message Format

Category	Type	Message
Error Reporting messages	3	Destination Unreachable
	4	Source Quench
	11	Time Exceeded
	12	Parameter Problem
	5	Redirection
Query Messages	8 or 0	Echo Request or Reply
	13 or 14	Timestamp Request or Reply
	17 or 18	Address Mask Request & Reply
	9 or 10	Router-solicitation & Advertisement

## Message Type- Destination Unreachable

---

- When a router cannot route a datagram or a host cannot deliver a datagram, the datagram is discarded and the router or the host sends a destination-unreachable message back to the source host that initiated the datagram.

# Message Type- Source Quench

---

- Is a request to decrease the traffic rate.
- A source-quench message informs the source that a datagram has been discarded due to congestion in a router or the destination host.
- The source must slow down the sending of datagrams until the congestion is relieved.
- Take the source IP address from the discarded packet and informs the source by sending a source quench message.

# Message Type- Time Exceeded

---

- Each datagram contains a field called *time to live*. When a datagram visits a router, the value of this field is decremented by 1.
- Whenever a router decrements a datagram with a time-to-live value to zero, it discards the datagram and sends a time-exceeded message to the original source.
- When the final destination does not receive all of the fragments in a set time, it discards the received fragments and sends a time-exceeded message to the original source.

# Message Type- Time Exceeded

---

- Code 0 - by routers to show that the value of the time-to-live field is zero.
- Code 1 - by the destination host to show that not all of the fragments have arrived within a set time.

Type: 11	Code: 0 or 1	Checksum
Unused (All 0s)		
Part of the received IP datagram including IP header plus the first 8 bytes of datagram data		

# Message Type- Parameter Problem

---

- If a router or the destination host discovers an ambiguous or missing value in any field of the datagram, it discards the datagram and sends a parameter-problem message back to the source.
- Can be created by a router or a destination host.
- E.g. checksum error

# Message Type- Redirection

- Used in the building of routing tables of hosts.
- Used when the source uses a wrong router to send out its message. The router informs the source that it needs to change its default router in the future.
- For efficiency, hosts do not take part in the routing update process because there are many more hosts in an internet than routers.
- Updating the routing tables of hosts dynamically produces unacceptable traffic. Hence the hosts usually use static routing with help of routers.

Type: 5	Code: 0 to 3	Checksum
IP address of the target router		
Part of the received IP datagram including IP header plus the first 8 bytes of datagram data		

## Message Type- Echo Request And Echo Reply

---

- An echo-request message can be sent by a host or router.
- An echo-reply message is sent by the host or router that receives an echo-request message.
- Echo-request and echo-reply messages can be used by network managers to check the operation of the IP protocol.
- Echo-request and echo-reply messages can test the reachability of a host.
- This is usually done by invoking the ping command.
- It is used to ping a message to another host that “Are you alive”.

# Message Type- Echo Request and Echo Reply

---

Type 8: Echo request

Type 0: Echo reply

Type: 8 or 0	Code: 0	Checksum
Identifier		Sequence number
Optional data Sent by the request message; repeated by the reply message		

# Message Type- Time Stamp Request and Reply

---

- Timestamp-request and timestamp-reply messages can be used to calculate the round-trip time between two devices or to check whether the clocks in two devices are synchronized.
- The timestamp request message sends a 32-bit number  $\tau$  defines the time the message is sent.
- The timestamp reply resends that number and two new 32-bit numbers representing the time the request was received and the time the response was sent.
- The sender can calculate the one way and round-trip time.

# Message Type- Address Mask Request & Reply

---

- A host may know its IP address, but it may not know the corresponding mask.
- To obtain its mask, a host sends an address-mask-request message to a router on the LAN. If the host knows the address of the router, it sends the request directly to the router. If it does not know, it broadcasts the message.
- The router receiving the address-mask-request message responds with an address-mask-reply message, providing the necessary mask for the host.

# Message Type- Router-solicitation & Advertisement

- A host can broadcast (or multicast) a router-solicitation message.
- The router or routers that receive the solicitation message broadcast their routing information using the router-advertisement message.
- Router Advertisement message is sent by a router on the local area network to announce its IP address as available for routing.
- A router can also periodically send router-advertisement messages even if no host has solicited.
- When a router sends out an advertisement, it announces not only its own presence but also the presence of all routers on the network of which it is aware.

**THANK YOU!!!**

---

# COMPUTER NETWORKS

## MODULE 4.3

---

MS. JINCY J FERNANDEZ

ASST. PROF, CSE

RSET

# Logical Addresses

---

- An Internet is a combination of physical networks.
- The physical networks are connected together by routers.
- The hosts and routers are recognized at the network level by their logical addresses.
- Logical address is unique globally.
- **Logical addresses are the IP addresses – 32 bits long.**

# Physical Addresses

---

- At the physical level the hosts are recognized by their physical addresses.
  - Physical address is the local address.
- 
- **48-bit MAC address in the Ethernet**
  - Delivery of a packet to a host or a router requires two levels of addressing:  
logical and physical.

# Issues with IP address

---

- IP addresses are not sufficient for sending packets.
- Data link layer NICs do not understand IP addresses.
- Every NIC equipped with 48-bit Ethernet address.
- NICs send and receive frames based on 48-bit Ethernet address.
- Need to map a logical address to its corresponding physical address and vice versa.
- Mapping is of two types:
  - Static Mapping
  - Dynamic Mapping

# Static Mapping

- 
- Table which associates a logical address to a physical address is stored in each machine.
  - Limitations
    - I. A machine could change its NIC, resulting in a new physical address.
    - 2. In some LANs, the physical address changes every time the computer is turned on.
    - 3. A mobile computer can move from one physical network to another, resulting in a change in its physical address.
  - Static mapping table has to be updated periodically.
  - It can affect network performance.

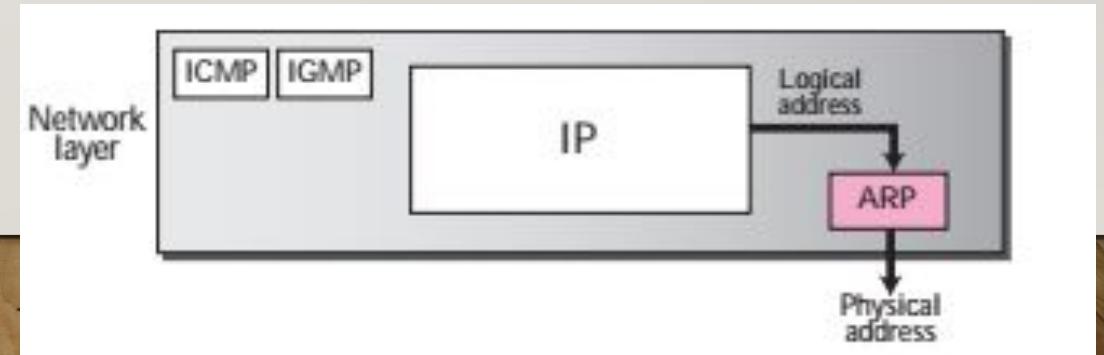
# Dynamic Mapping

---

- Each time a machine knows the logical address of another machine it uses a protocol to find the physical address and vice versa.
- Two protocols are designed to perform dynamic mapping are:
  - **Address Resolution Protocol (ARP)**
    - ARP maps a logical address to a physical address
  - **Reverse Address Resolution Protocol (RARP)**
    - RARP maps a physical address to a logical address

# Address Resolution Protocol (ARP)

- IP datagram has to be encapsulated in a frame and passed to the physical network.
- Sender needs the physical address of the receiver but it knows only the logical address of the receiver.
- ARP accepts a logical address from the IP protocol, maps the address to the corresponding physical address and pass it to the data link layer.



# ARP Process

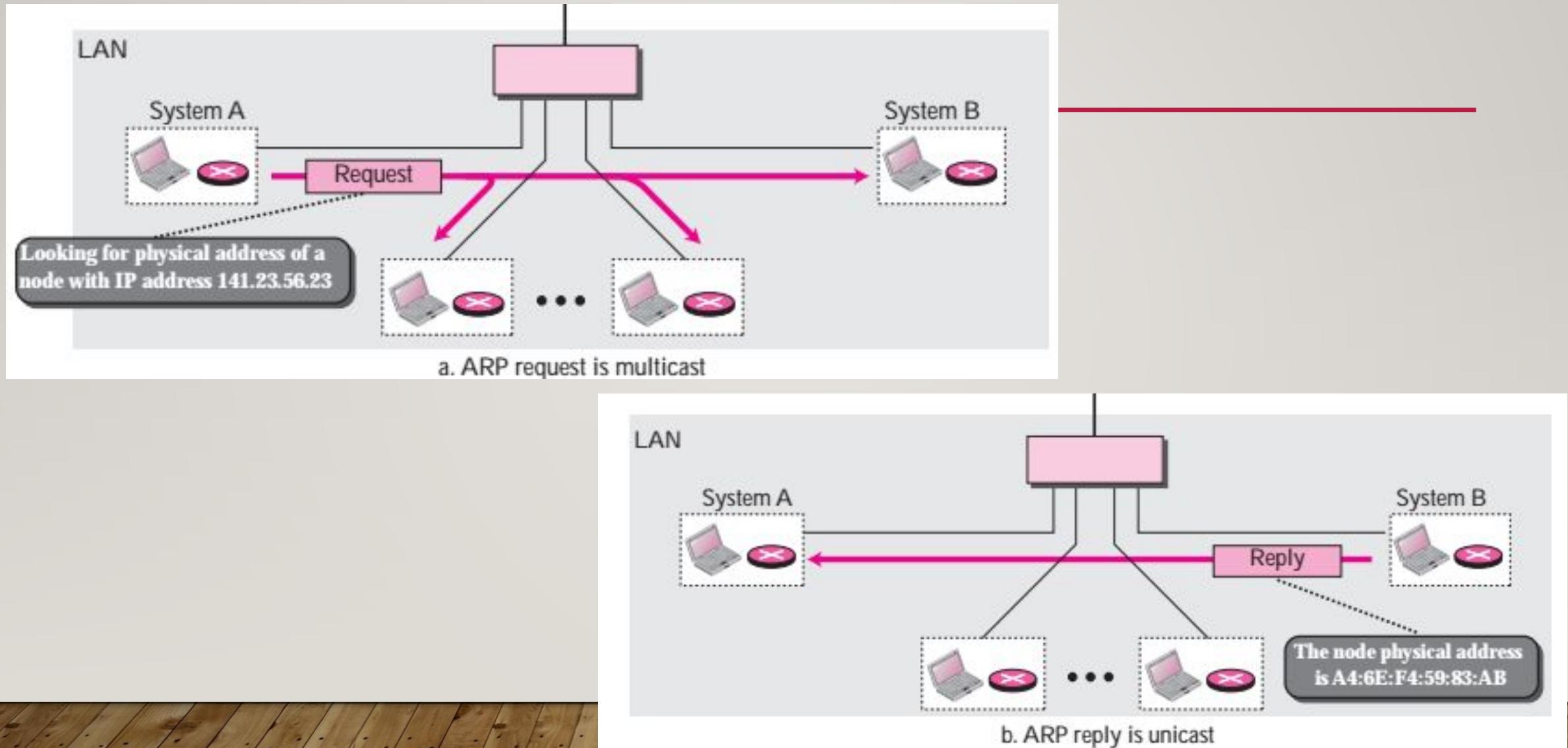
---

- Anytime a host, or a router, needs to find the physical address of another host it sends an **ARP request packet**.
- The packet includes the physical and IP addresses of the sender and the IP address of the receiver.
- Broadcast ARP request.

# Arp Process

---

- Every host or router on the network receives and processes the ARP request packet.
- Only the intended recipient recognizes its IP address and sends back an **ARP response packet.**
- The response packet contains the recipient's IP and physical addresses.
- The packet is unicast directly to the inquirer using the physical address received in the query packet.



# ARP Packet Format

Hardware Type	Protocol Type	
Hardware length	Protocol length	Operation Request 1, Reply 2
Sender hardware address (For example, 6 bytes for Ethernet)		
Sender protocol address (For example, 4 bytes for IP)		
Target hardware address (For example, 6 bytes for Ethernet) (It is not filled in a request)		
Target protocol address (For example, 4 bytes for IP)		

# ARP Packet Format

- **Hardware type**
  - This is a 16-bit field defining the type of the network on which ARP is running.
  - Each LAN has been assigned an integer based on its type – e.g. Ethernet is given the type 1.
- **Protocol type**
  - This is a 16-bit field defining the protocol.
  - For example, the value of this field for the IPv4 protocol is  $(0800)_{16}$ .
- **Hardware length**
  - This is an 8-bit field defining the length of the physical address in bytes.
  - For example, for Ethernet the value is 6.

# ARP Packet Format

---

- **Protocol length**
  - This is an 8-bit field defining the length of the logical address in bytes.
  - For example, for the IPv4 protocol the value is 4.
- **Operation**
  - This is a 16-bit field defining the type of packet.
  - Two packet types are defined - ARP request (1) and ARP reply (2).
- **Sender hardware address**
  - This is a variable-length field defining the physical address of the sender.
  - For example, for Ethernet this field is 6 bytes long.

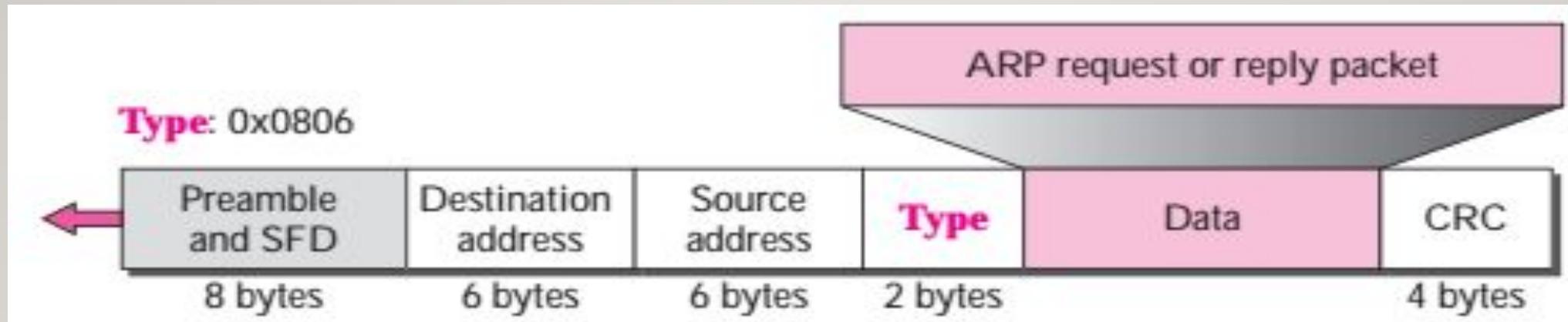
# ARP Packet Format

---

- Sender protocol address
  - This is a variable-length field defining the logical (for example, IP) address of the sender.
- Target hardware address
  - This is a variable-length field defining the physical address of the target.
  - In an ARP request message, this field is all 0s because the sender does not know the physical address of the target.
- Target protocol address
  - This is a variable-length field defining the logical address of the target.

# Encapsulation of an ARP Packet

---



- An ARP packet is encapsulated in an Ethernet frame

# Steps in an ARP Process

---

1. The sender knows the IP address of the target.
2. IP asks ARP to create an ARP request message, filling in :
  - The sender physical address, the sender IP address, and the target IP address
  - The target physical address field is filled with 0s
3. The message is passed to the data link layer where it is encapsulated in a frame using the physical address of the sender as the source address and the physical broadcast address as the destination address

# Steps in an ARP Process

---

4. Every host or router receives the frame, because the frame contains a broadcast destination address
  - All machines except the one targeted drop the packet
  - The target machine recognizes the IP address
  
5. The target machine replies with an ARP reply message that contains its physical address
  - The message is unicast

# Steps in an ARP Process

---

6. The sender receives the reply message
  - It now knows the physical address of the target machine
7. The IP datagram, which carries data for the target machine, is now encapsulated in a frame and is unicast to the destination

# RARP

- 
- Given an Ethernet address, what is the corresponding IP address?
  - Use RARP- Reverse ARP.
  - This protocol allows a newly booted workstation to broadcast its Ethernet address.
  - RARP server sees this request, looks up the Ethernet address in its configuration files, and sends back the corresponding IP address

# Disadvantage

---

- RARP server is needed on all networks.
- RARP can provide only IP address, no information on subnet mask, IP address of router, IP address of file server etc.
- Broadcast messages will be blocked by certain routers.
- So go for an alternative Protocol:
  - **BOOTP**

# BOOTP

- 
- It is a client-server protocol designed to overcome deficiencies of RARP protocol.
  - It can run anywhere in the Internet.
  - BOOTP uses UDP messages, which are forwarded over routers.
  - Provides a diskless workstation with all information like:
    - IP address of the file server
    - IP address of the default router
    - Subnet mask

# Disadvantage of BOOTP

---

- It is a static configuration protocol.
- Requires manual configuration of tables mapping IP address to Ethernet address.
- Administrator has to assign an IP address and enter mapping of (Ethernet Address, IP Address) into the BOOTP configuration tables.
- So go for an extended version named as **DHCP**.
- **Dynamic Host Configuration Protocol**

# Dynamic Host Configuration Protocol(DHCP)

---

- Large organizations or ISP receive block of addresses from ICANN(Internet Corporation for Assigned Names and Numbers)
- Small organizations receive block of addresses from ISP.
- Network administrator manually assign the IP addresses to the hosts or routers? tedious and error prone.

# Dynamic Host Configuration Protocol(DHCP)

---

- Address assignment in an organization can be done automatically using DHCP.
- Dynamic Host Configuration Protocol (DHCP) is a client/server protocol that automatically provides an Internet Protocol (IP) host with its IP address and other related configuration information such as the subnet mask and default gateway.
- With DHCP, every network must have a DHCP server that is responsible for configuration.

# DHCP - Working

---

- A newly booted machine broadcasts a **DHCP DISCOVER** packet. It must reach the DHCP server.
- Router will be configured to receive DHCP broadcasts and relay them to the DHCP server.
- For this, the router should be aware of the IP address of the DHCP server.
- When the server receives the packet, it allocates a free IP address and sends it to the host in a **DHCP OFFER** packet

# DHCP

---

- Issue with automatic assignment is how long an IP address should be allocated?
  - If a host leaves a network and does not return its IP address to the DHCP server, the address will be permanently lost.
- **Leasing:** technique of assigning an IP address for a fixed period of time.
- Just before the lease expires, the host must ask for a DHCP renewal.

# ADVANTAGES OF DHCP

---

- **Reliable IP address configuration.**

- DHCP minimizes configuration errors caused by manual IP address configuration, such as typographical errors, or address conflicts caused by the assignment of an IP address to more than one computer at the same time.

- **Reduced network administration.**

- Centralized and automated TCP/IP configuration.

# Configuration

---

- **Automatic allocation**—The DHCP server assigns a permanent IP address to a client from its IP Pools. On the firewall, a Lease specified as Unlimited means the allocation is permanent.
- **Dynamic allocation**—The DHCP server assigns a reusable IP address from IP Pools of addresses to a client for a maximum period of time, known as a lease. This method of address allocation is useful when the customer has a limited number of IP addresses; they can be assigned to clients who need only temporary access to the network.
- **Static allocation**—The network administrator chooses the IP address to assign to the client and the DHCP server sends it to the client. A static DHCP allocation is permanent. The DHCP assignment remains in place even if the client logs off, reboots, has a power outage, etc.

# THANK YOU!!!

---

# **COMPUTER NETWORKS**

## **MODULE 4.4**

---

**MS. JINCY J FERNANDEZ**

**ASST. PROFESSOR**

**RSET**

# Terminologies

---

- The Internet is divided into hierarchical domains called Autonomous Systems (ASs).
- Autonomous System is a set of networks which share common routing policies, e.g. AT&T.
  - A large corporation that manages its own network and has full control over it, is an autonomous system.
  - A local ISP that provides services to local customers is an autonomous system.

# Autonomous System

---

- Autonomous Systems are divided into three categories:

## 1. Stub AS

- A stub AS has **only one connection to another AS**.
- A stub AS **is either a source or a sink**.
- The hosts in the AS can send data traffic to other AS and can receive data coming from hosts in other AS.
- **Data traffic cannot pass** through a stub AS.
- An example of a stub AS is a **small corporation or a small local ISP**.

# Autonomous System

---

## 2. Multihomed AS

- Has **more than one connection to other AS**.
- It is still **only a source or sink for data traffic** - can receive data from more than one AS and can send data to more than one AS.
- There is **no transient traffic** - does not allow data coming from one AS and going to another AS to pass through.
- An example is a large corporation that is connected to more than one **regional or national AS that does not allow transient traffic**.

# Autonomous System

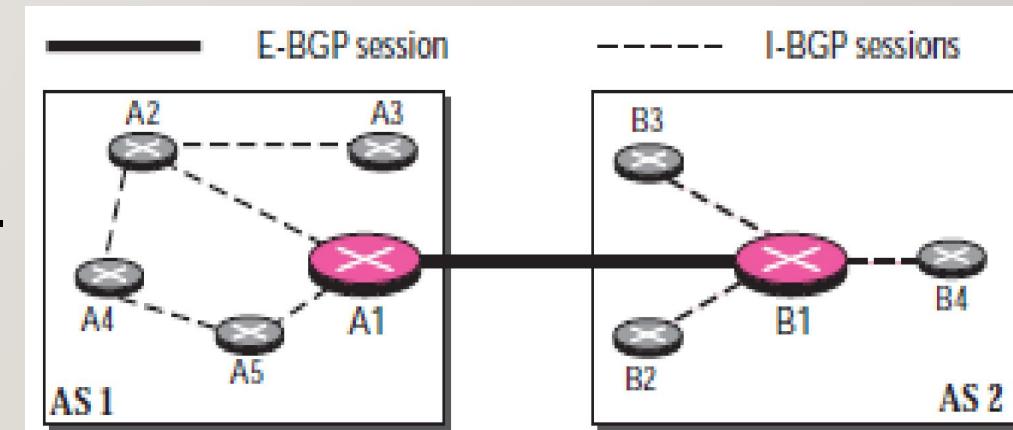
---

## 3. Transit AS

- A transit AS is a multihomed AS that also allows transient traffic
- Examples of transit ASs are national and international ISPs.

# Gateway Protocols

- Interior Gateway Protocols are routing protocols within an Autonomous System, e.g. RIP, OSPF
- Exterior Gateway Protocols are routing protocols used between Autonomous Systems, e.g. BGP
  - E-BGP session
    - To exchange info between two different AS.
  - I-BGP session
    - To exchange info within the same AS.



Collect info using I-BGP and exchange using E-BGP.

# BGP- Border Gateway Protocol

- Interdomain protocol (exterior gateway protocol).
- Distance vector protocol for routing between different Autonomous Systems.
- BGP uses a TCP connection to exchange information between peers.
- Policy based:
  - A corporate AS might be unwilling to carry transit packets originating in a foreign AS and ending in a different foreign AS, even if its own AS is on the shortest path between the two foreign AS.
  - On the other hand, it might be willing to carry transit traffic for its neighbors, or even for specific other AS that paid it for this service.

# BGP

---

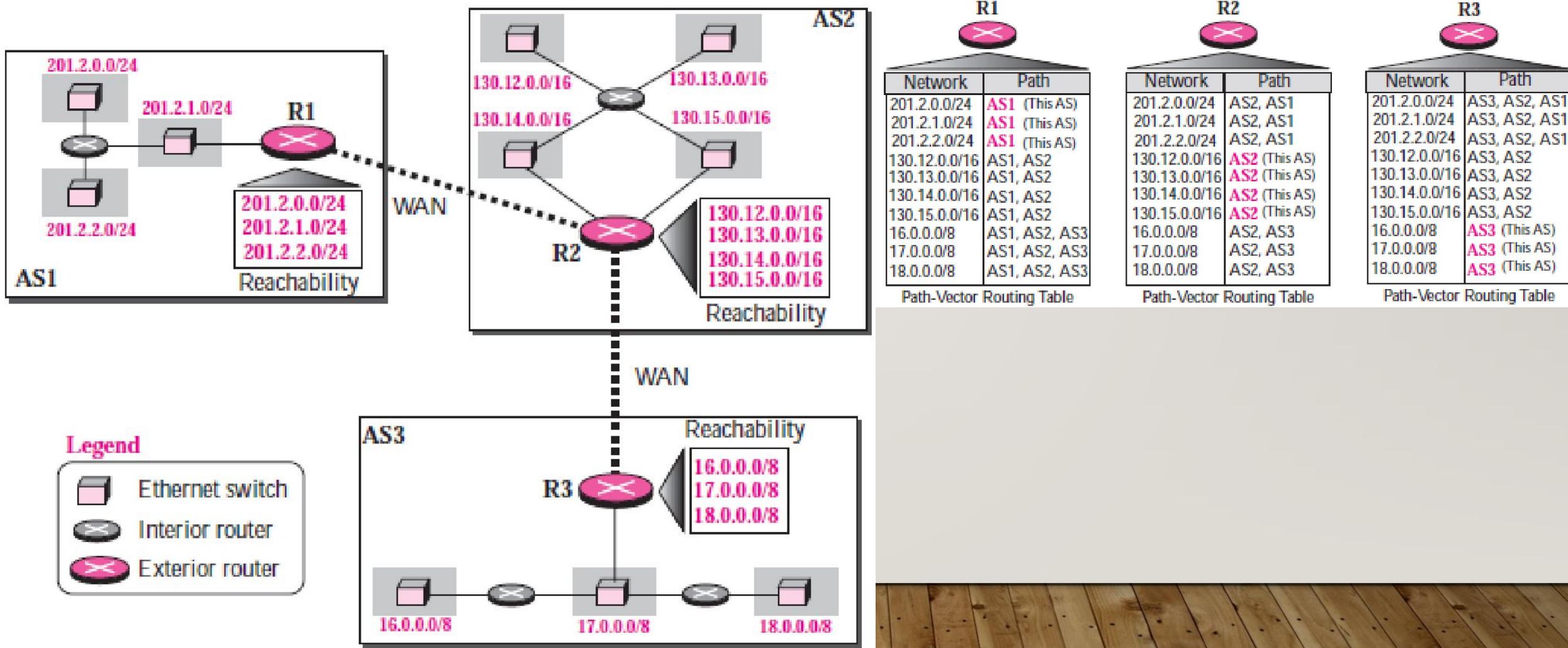
- Routing policy is implemented by deciding what traffic can flow over which of the links between AS.
- One common policy is that a customer ISP pays another provider ISP to deliver packets to any other destination on the Internet and receive packets sent from any other destination.
- The customer ISP is said to buy **transit service** from the provider ISP.

# BGP

---

- Each BGP router keeps track of the exact path to reach the destination not just the cost.
- Instead of periodically giving each neighbor its estimated cost to each destination, each BGP router tells its neighbors the path it is using to reach each destination.
- BGP solves the count to infinity problem as whole path is known.

# BGP USES PATH VECTOR ROUTING



# EXAMPLES

- Consider the routing table of F.
- After all path information come from the neighbors F examines them to get the best.
- It discards paths from I and E, as they pass through F itself.
- Choice remains between B and G.
- Every BGP router contains a module to examine routes and score them.
- The scoring will consider policy violations as well.

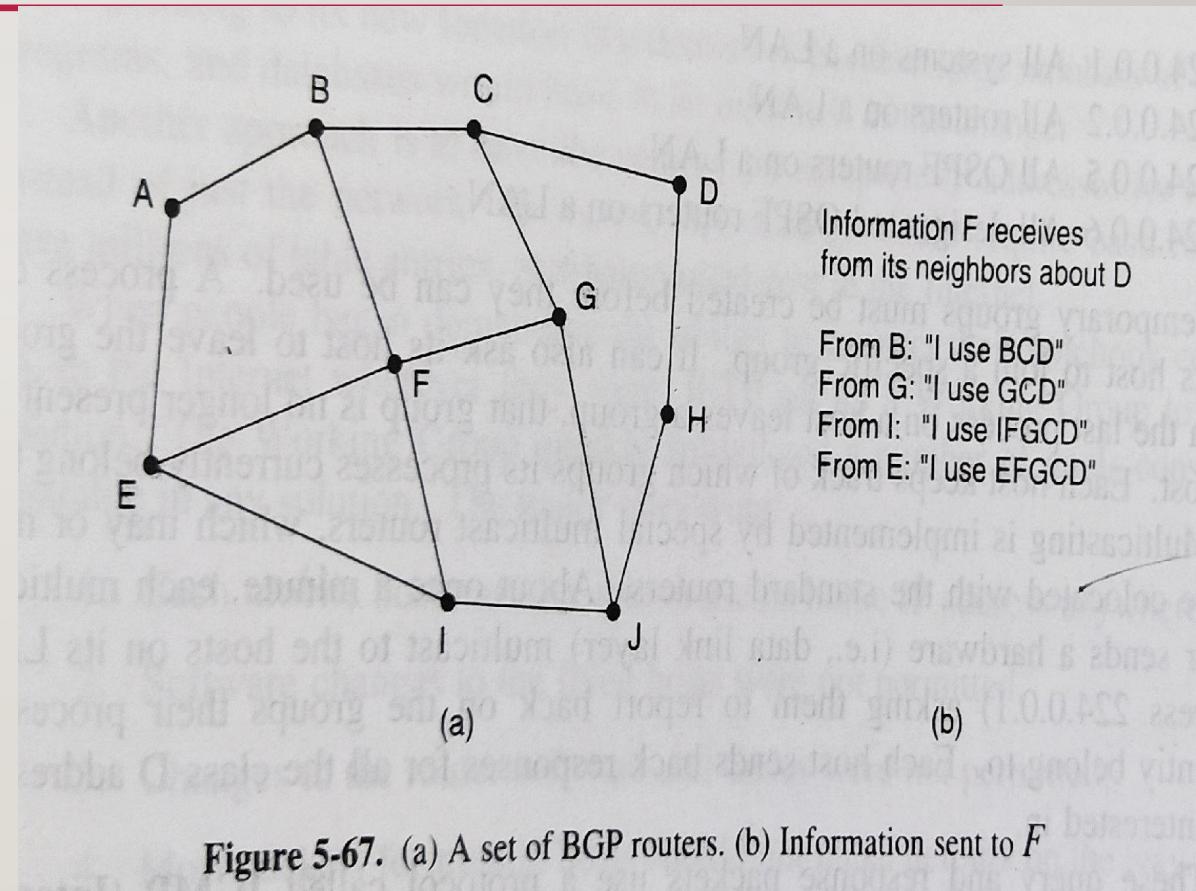


Figure 5-67. (a) A set of BGP routers. (b) Information sent to F

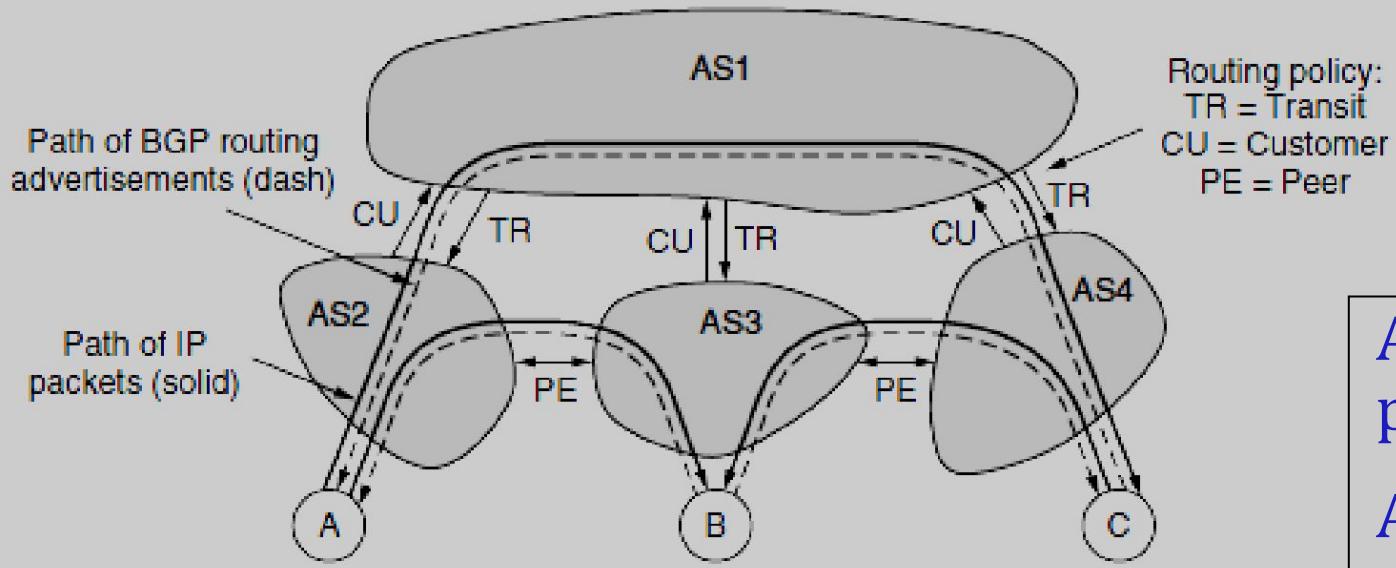


Figure 5.67 Routine policies between four autonomous systems

- **Transit**
- AS2, AS3 and AS4 are customers of AS1 and buy transit service from AS1.
- Suppose source A wants to send packets to C.
- AS4 advertises C as a destination to its transit provider AS1.
- AS1 now advertises a route to C to its other customers.

AS2 now knows that it can send a packet to C via AS1(paid).

A - AS2 - AS1 - AS4 - C

## Peering

AS2 and AS3 can send traffic directly to each other for free.

Two AS send routing advertisements to each other for addresses that reside in their network.

A - AS2 - AS3 - B

But a packet from A to C should be send through the transit.

# **COMPUTER NETWORKS**

## **MODULE 4.5**

---

MS. JINCY J FERNANDEZ

ASST. PROFESSOR

RSET

# INTERNET MULTICASTING

---

- Normal IP Communication: between one sender and one receiver.
- However, for some applications, it is useful for a process to able to send to large number of receivers simultaneously.
  - Updating distributed systems.
- IP supports multicasting using class D address.
- Each class D identifies a group of hosts.

# INTERNET MULTICASTING

---

- When a process sends a packet to a class D address, best attempt is made to deliver it to all the members of the group addressed. But no guarantee.
- Two kinds of group addresses are supported: Temporary and Permanent.
- **Permanent:**
  - Always there and does not need to setup.
  - Each permanent group has a permanent group address.
  - E.g. 224.0.0.1 ↗ all systems on a LAN  
224.0.0.2 ↗ all routers on a LAN  
224.0.0.5 ↗ all OSPF routers on a LAN

# INTERNET MULTICASTING

---

- **Temporary:**
  - Must be created before they can be used
  - A process can ask its host to join or leave the group.
  - Each group keep tracks of which groups its process currently belongs to.
  - About once a minute, each multicast router sends a query packet to all the hosts on its LAN asking them to report back on the groups to which they currently belong.
  - Each host sends back responses for all the class D addresses it is interested in.
  - These query and response packets use a protocol called IGMP (**Internet Group Management Protocol**)

# IPV6

---

- Main reason for migration from IPV4 to IPV6 is the small size of the address space in IPV4.
  - Uses 128 bit addresses.

# IPV6

---

- Representations
  - Binary notation: 128 bits
  - Colon hexadecimal notation: divides the address into eight sections, each made of four hexadecimal digits separated by colons;  
FEF6:BA98:7654:3210:ADEF:BBFF:2322:FF00
  - CIDR notation: IPV6 uses hierarchical addressing; FDEC::BBFF:0:FFFF/60

# IPV6

---

- Abbreviation:

- Abbreviate the address if many of the digits are zeros.
- 0074:74; 000F:F; 0000:0

- Zero compression:

- Applied to colon hex notation if there are consecutive sections of zeros only.
- Remove all the zeros and replace them with a double semicolon.
- FDEC:0:0:0:BBFF:0:FFFF → FDEC::BBFF:0:FFFF
- zero compression with a **double colon can be applied only once.**

# IPV6

---

- Address Space

- $2^{128}$  addresses
- No address depletion
- Expand the address 0:16::1:12:1214

Ans: 0000:0016:0000:0000:0001:0012:1214

## ~~IPV6- ADDRESS SPACE ALLOCATION~~

- Like IPV4, the address space of IPV6 is divided into several blocks of varying size and each block is allocated for a special purpose.

<i>Block prefix</i>	<i>CIDR</i>	<i>Block assignment</i>
0000 0000	0000::/8	Special addresses
001	2000::/3	Global unicast
1111 110	FC00::/7	Unique local unicast
1111 1110 10	FE80::/10	Link local addresses
1111 1111	FF00::/8	Multicast addresses

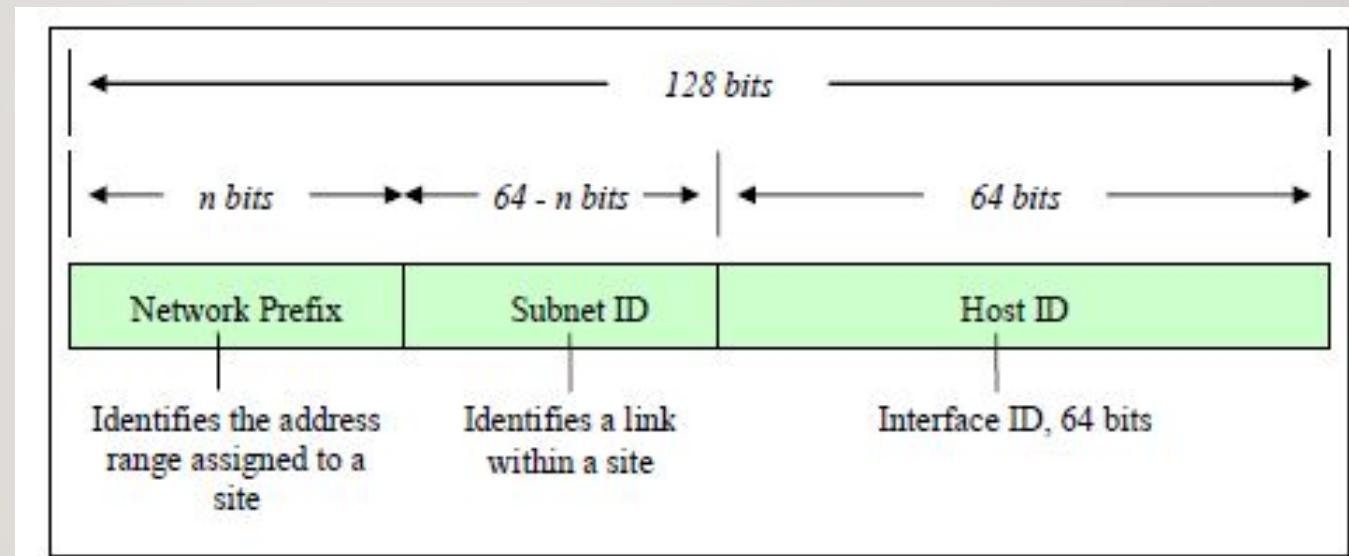
## ~~IPV6- ADDRESS SPACE ALLOCATION~~

---

- Global unicast addresses:
  - For one-to-one communication between two hosts in the Internet.
  - Size of this block is  $2^{125}$  bits.
  - Address in this block is divided into global routing prefix, subnet identifier and interface identifier.
    - **Global routing prefix:** to route the packet through the internet to the organization site.
      - First 3 bits are fixed (001), rest of the 45 bits can define up to  $2^{45}$  sites.
    - **Subnet identifier:** define a subnet in an organization; can have up to  $2^{16} = 65536$  subnets.
    - **Interface identifier:** similar to host id in IPV4.

# ~~IPV6 ADDRESS SPACE ALLOCATION~~

- Global unicast addresses:



## ~~IPV6- ADDRESS SPACE ALLOCATION~~

---

- Other assigned blocks
  - IPV6 uses two large blocks for private addressing (unique local unicast and link local block) and one large block for multicasting.

## ~~IPV6- ADDRESS SPACE ALLOCATION~~

---

- During the transition from IPV4 to IPV6, hosts can use their IPV4 addresses embedded in IPV6 addresses.
- Two formats for this:
  - **Compatible address:** address of 96 zero bits followed by 32 IPV4 address.
  - **Mapped address:** used when a computer already migrated to IPV6 wants to send a message to another computer using IPV6.

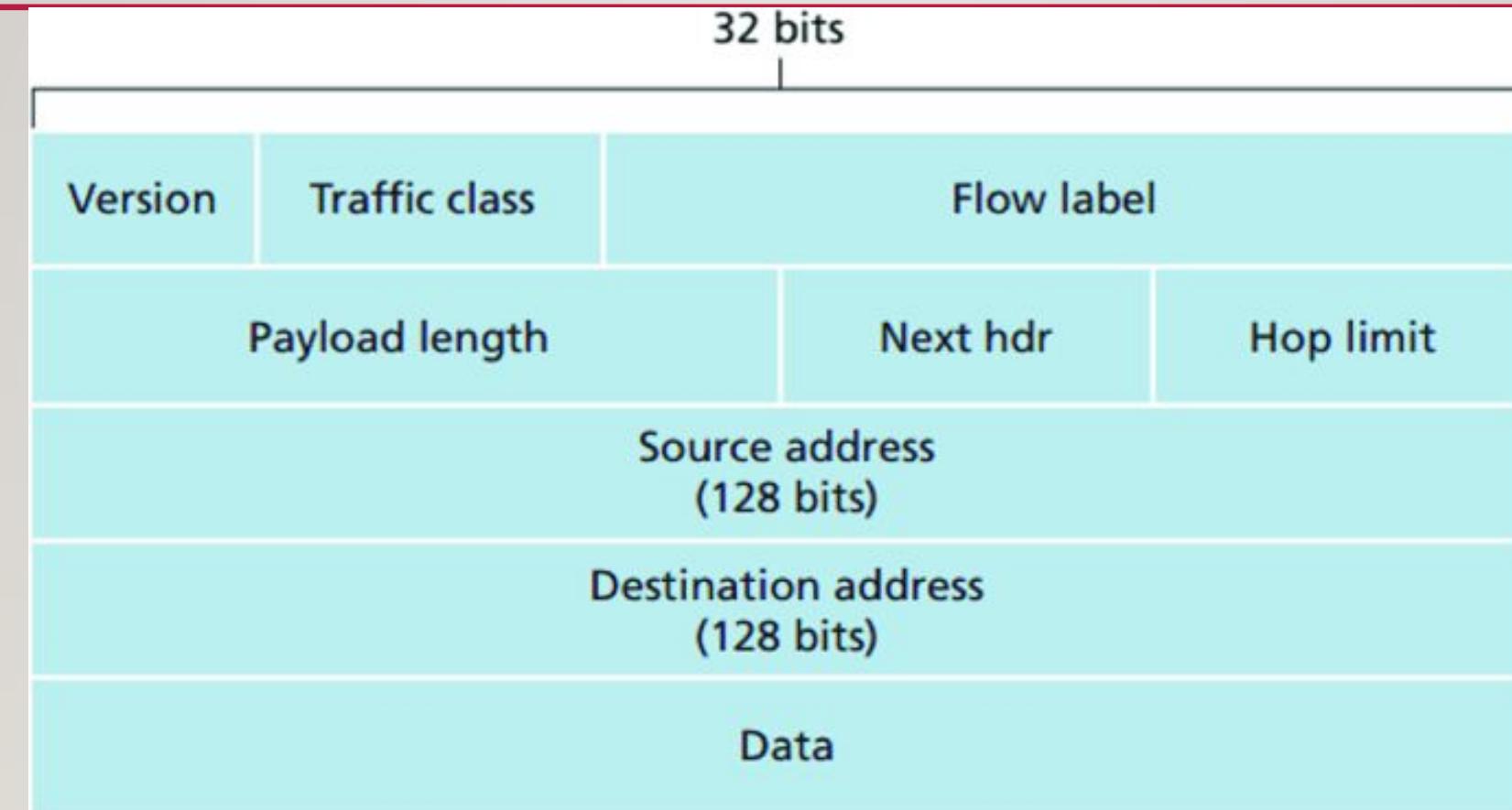
# IPV6- DATAGRAM FORMAT

---

- Changes in IPV6 protocol:

- **Better header format:** Options are separated from the base header and inserted (when needed) between the base header and data.
- **New options:** for additional functionalities.
- **Allowance for extension:** designed to allow the extension of the protocol if required by new technologies or applications.
- **Support for resource allocation:** instead of type of service field, traffic class and flow label fields have been added to enable the source to request special handling of the packet.
- **Support for more security:** Encryption and authentication options in IPV6 provide confidentiality and integrity of the packet.

# IPV6- DATAGRAM FORMAT



# IPV6- DATAGRAM FORMAT

---

- Version: 4- bit field with value 6.
- Traffic Class: 8- bit field; distinguish different payloads with different delivery requirements. It replaces the type of service field in IPv4 (**PRIORITY**)
- Flow label: 20-bit field; handles flow of data.
- Payload length: 16- bit field; defines the length of the IP datagram excluding the header.
  - Length of the base header is fixed (40bytes)

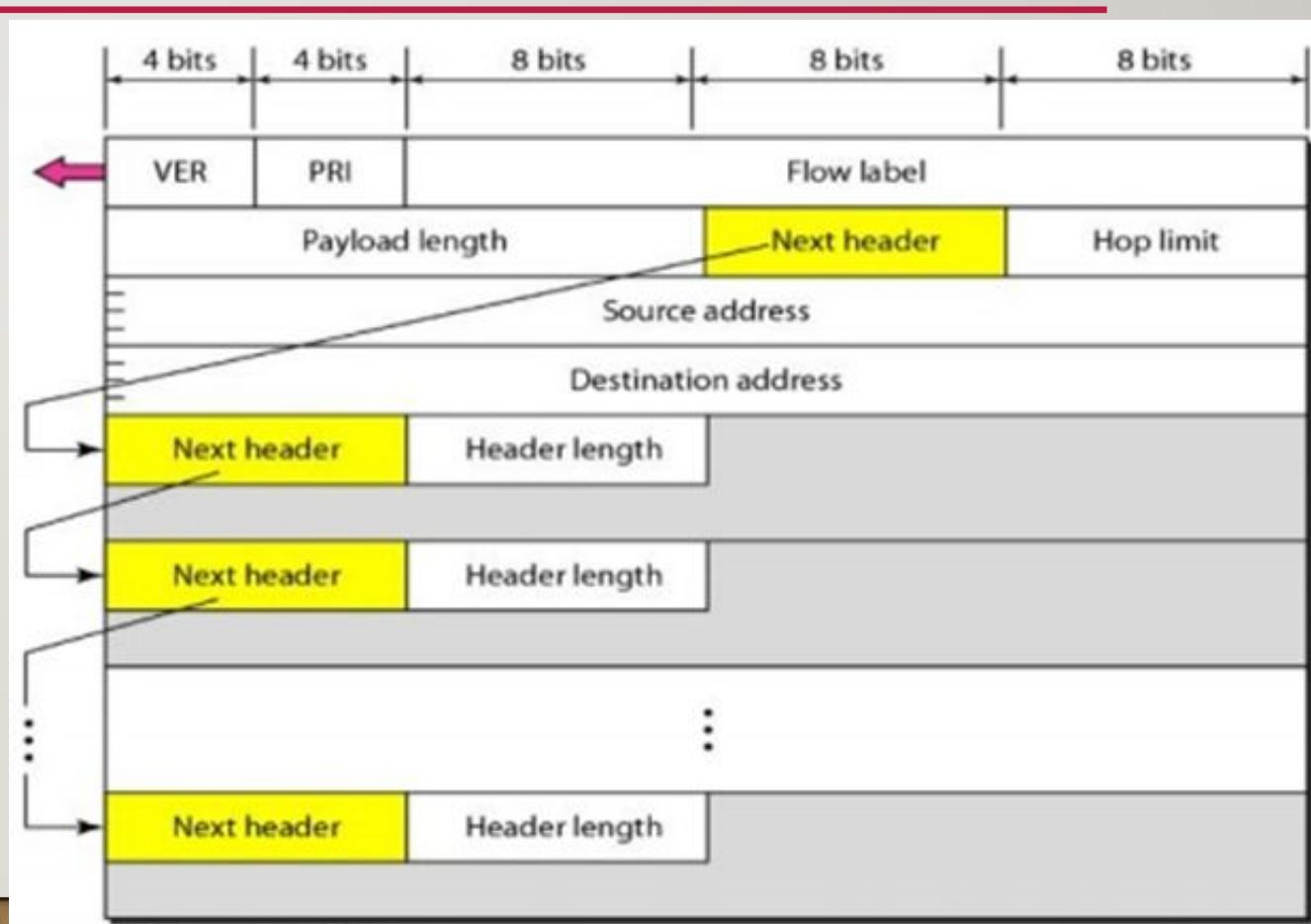
# IPV6- DATAGRAM FORMAT

---

- Next header: 8-bit field defining the type of the first extension header (if present) or the type of the data that follows the base header in the datagram; similar to protocol field in IPV4.
- Hop limit: 8-bit field; same as TTL field in IPV4.
- Source and Destination addresses: 16- byte; defines the source and destination of the datagram.
- Payload: has a different format and meaning.

# IPV6- DATAGRAM FORMAT

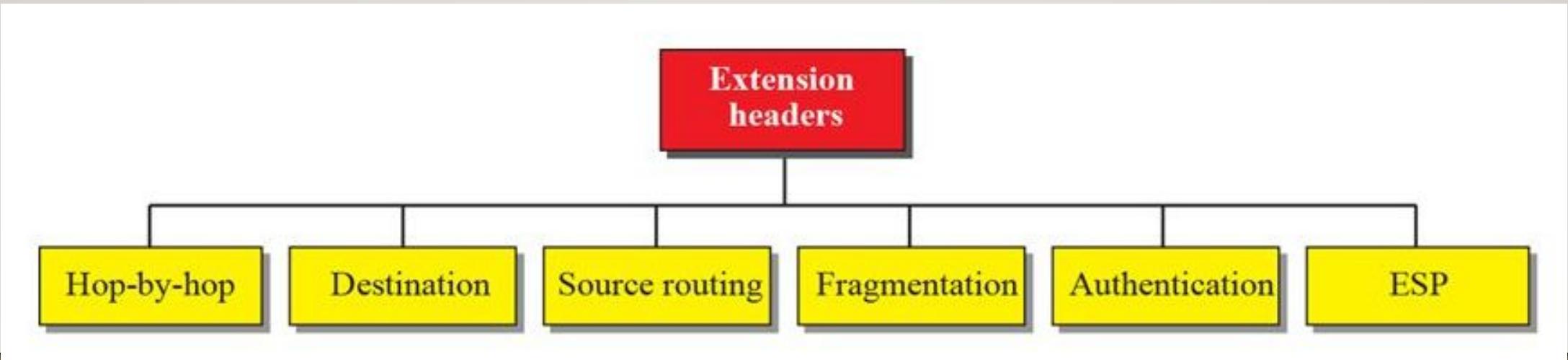
- Payload: has a different format and meaning.
- Payload means a combination of zero or more extension headers followed by the data from other protocols.
- Each extension header has two mandatory fields: next header and length followed by information related to the particular option.



# IPV6- DATAGRAM FORMAT

- Extension header:

- IPV6 packet<sup>2</sup> base header and some extension headers.
- The length of the base header is fixed at 40 bytes.
- Maximum 6 extension headers.



# IPV6- DATAGRAM FORMAT

---

- Extension header:
  - Hop by hop option: used when source needs to pass information to all routers visited by the datagram.
  - Destination option: used when the source needs to pass information to the destination only. Intermediate routers are not permitted access to this information.
  - Source Routing: combines the concepts of the strict source route and loose source route options in IPV4.

# IPV6- DATAGRAM FORMAT

---

- Extension header:
  - Fragmentation: IPV6 datagrams can be fragmented only by the source, not by the routers. Reassembly takes place at the destination. The fragmentation of the packets at the router is not allowed to speed up the processing of packets in the router.
  - Authentication: it validates the message sender and ensures the integrity of data.
  - ESP(Encrypted Security Payload): provides confidentiality and guards against eavesdropping.

# **COMPUTER NETWORKS**

## **MODULE 5.I**

---

**MS. JINCY J FERNANDEZ**

**ASST. PROF, CSE**

**RSET**

# MODULE 5

---

Transport Layer

# INTRODUCTION

---

- Responsible for the delivery of a message from one process to another.
- A process is an application program running on a host.
- The transport layer header include port numbers
- Transport layer protocol can be connectionless or connection oriented
- A message is normally divided into transmittable segments.

# TRANSPORT LAYER SERVICES

---

- Services provided to the upper layers.
- Ultimate goal of transport layer is to provide efficient, reliable and cost-effective services to its users which are processes running on the application layer.
- For this we have a transport entity. It is located in the OS kernel or is a separate user process or on the NIC.

# TRANSPORT LAYER SERVICES

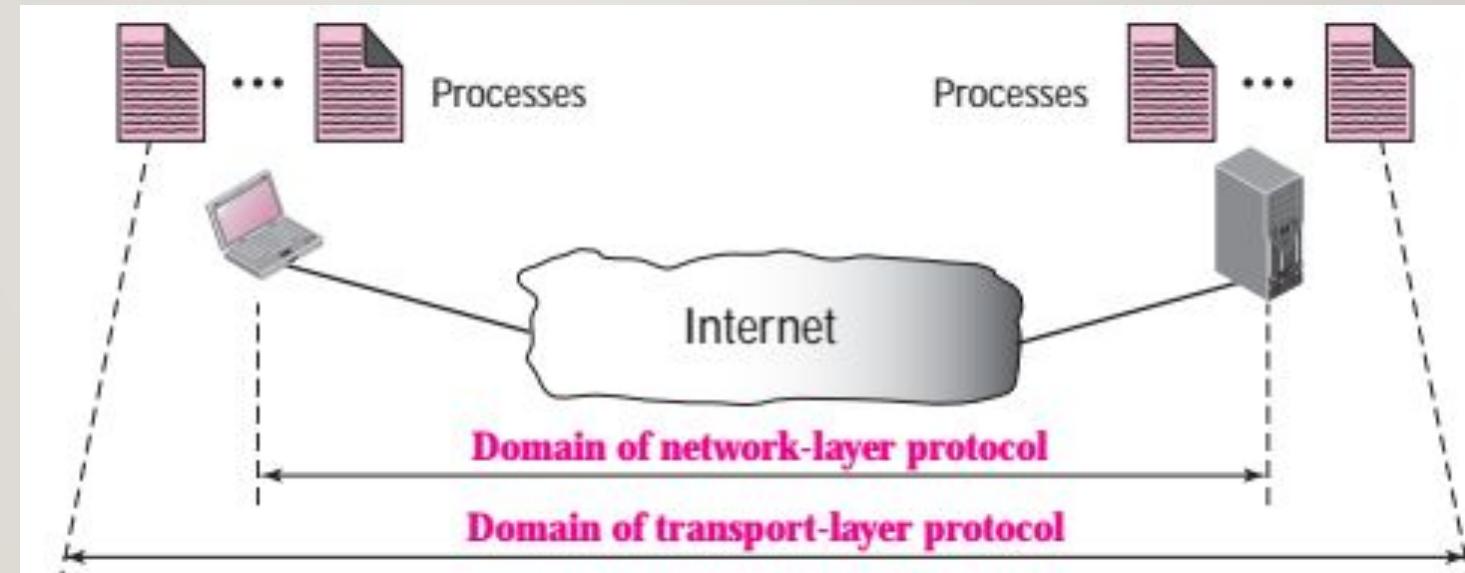
---

- ❑ Process-to-Process Communication
- ❑ Addressing: Port Numbers
- ❑ Encapsulation and Decapsulation
- ❑ Multiplexing and Demultiplexing
- ❑ Flow Control
- ❑ Error Control
- ❑ Congestion Control
- ❑ Connectionless and Connection-Oriented Services

# PROCESS – TO – PROCESS COMMUNICATION

---

- Process is an application layer entity – running program.
- Transport layer is responsible for delivering the message to the appropriate process.



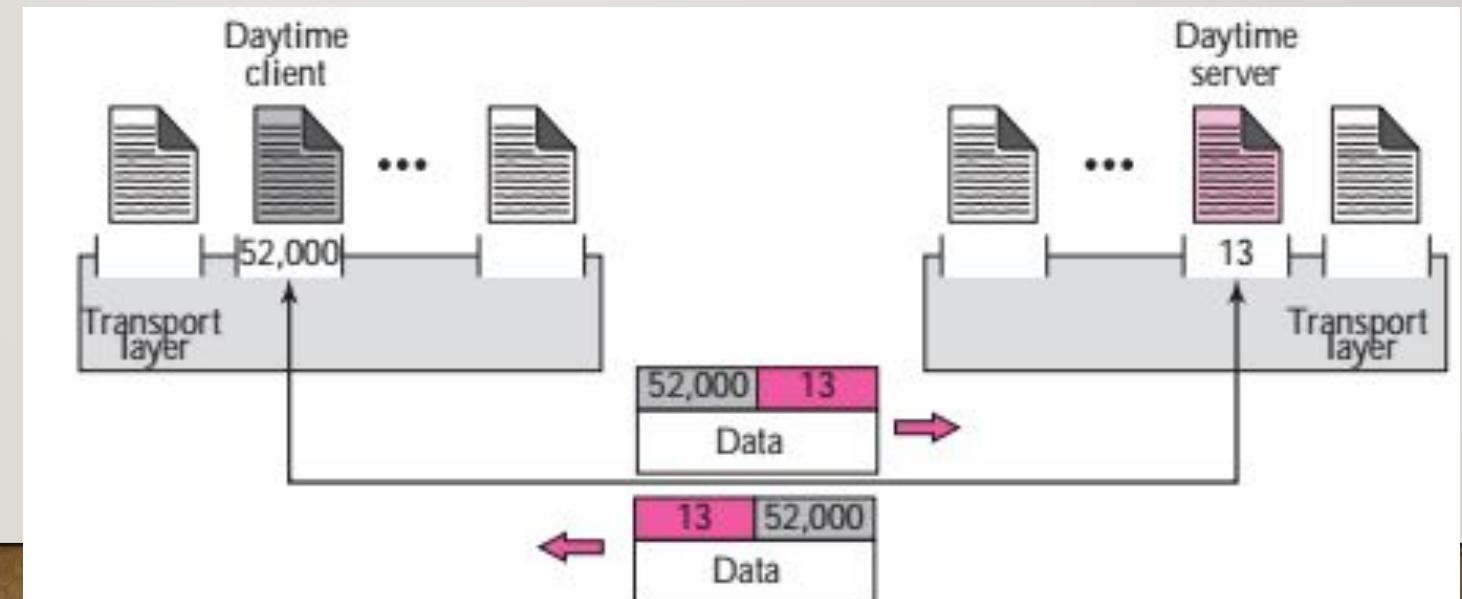
# ADDRESSING: PORT NUMBERS

---

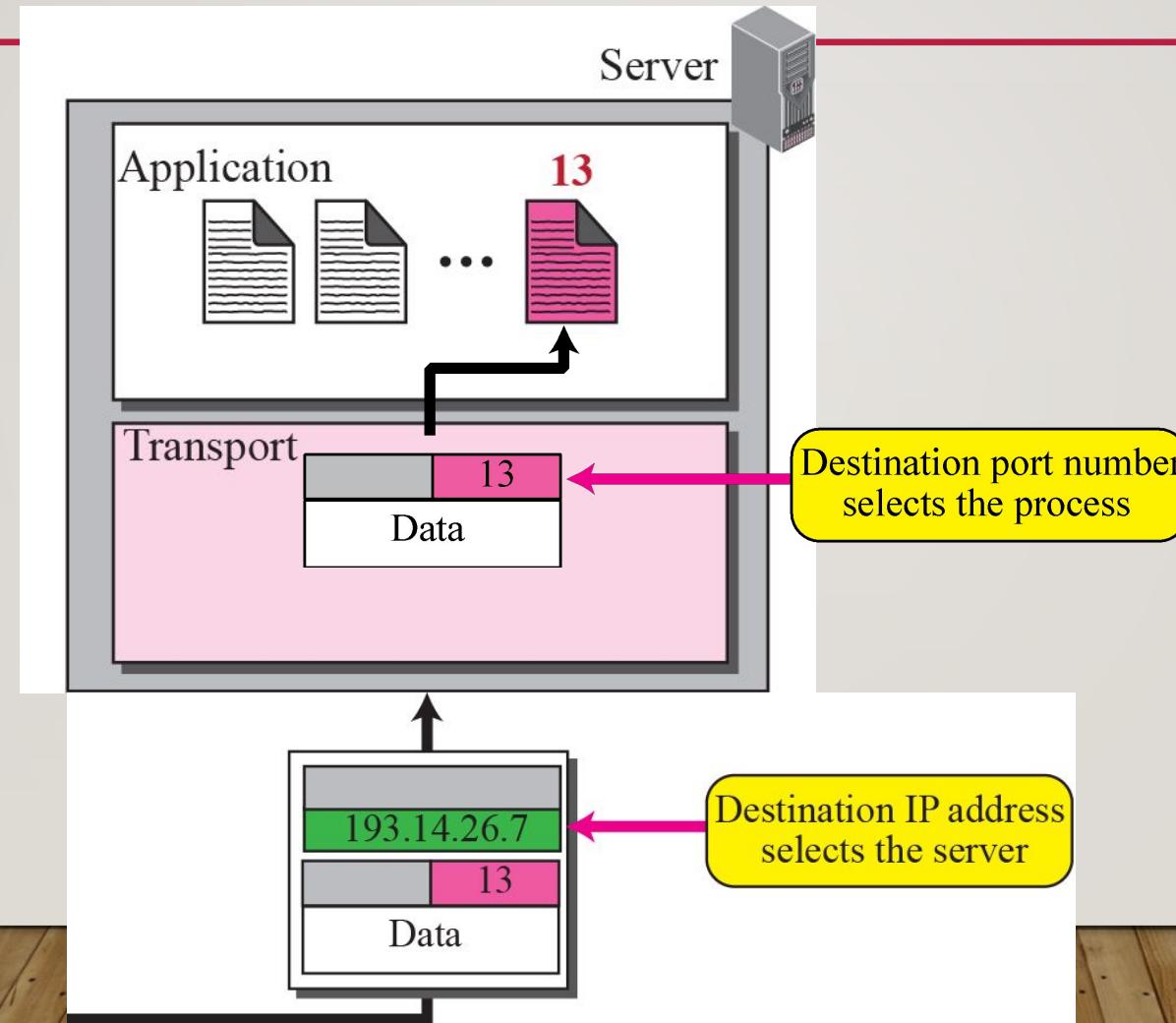
- ❑ Process-to-process communication usually follows the client-server paradigm.
- ❑ Hosts are identified by their IP addresses.
- ❑ Processes are defined by ports.
- ❑ Ports are identified by port numbers.
- ❑ In TCP/IP the **port numbers** are identifiers between **0 – 65,535**.

# ADDRESSING-PORT NUMBERS

- ❑ Clients use **ephemeral** port numbers: short lived.
- ❑ Servers use **well-known** port numbers.
- ❑ Every client process knows the well known port number of the corresponding server process.

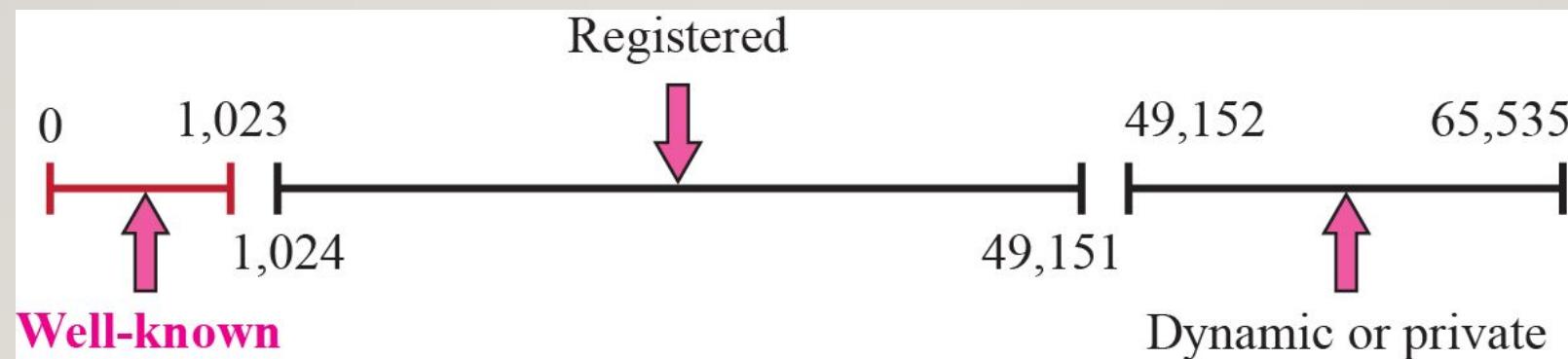


# ADDRESSING- IP ADDRESSES VERSUS PORT NUMBERS



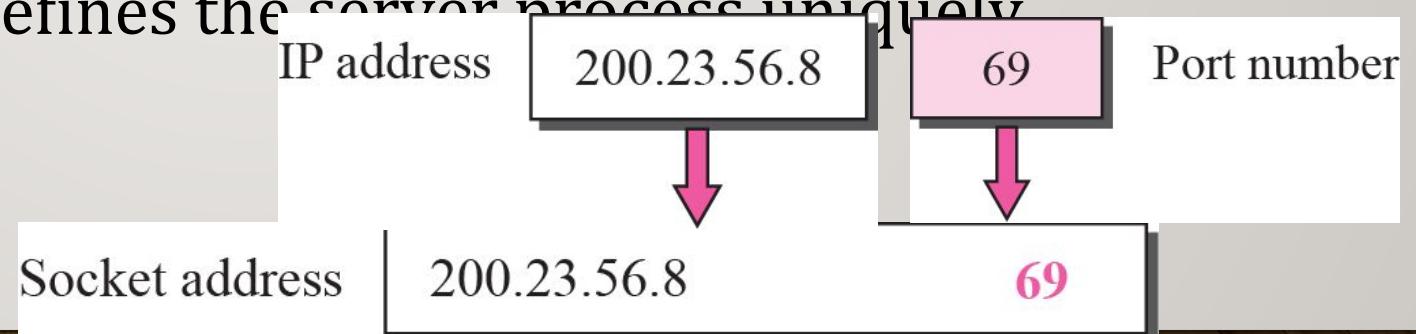
# ADDRESSING- ICANN RANGES

- Well known ports; controlled by ICANN. Eg: DNS-53, HTTP-80 etc.
- Registered ports.: not controlled by ICANN, but registered with ICANN. Eg: used by companies.
- Dynamic ports: neither controlled or registered with ICANN; can be used as temporary or private port numbers. Eg: assigned to normal users.

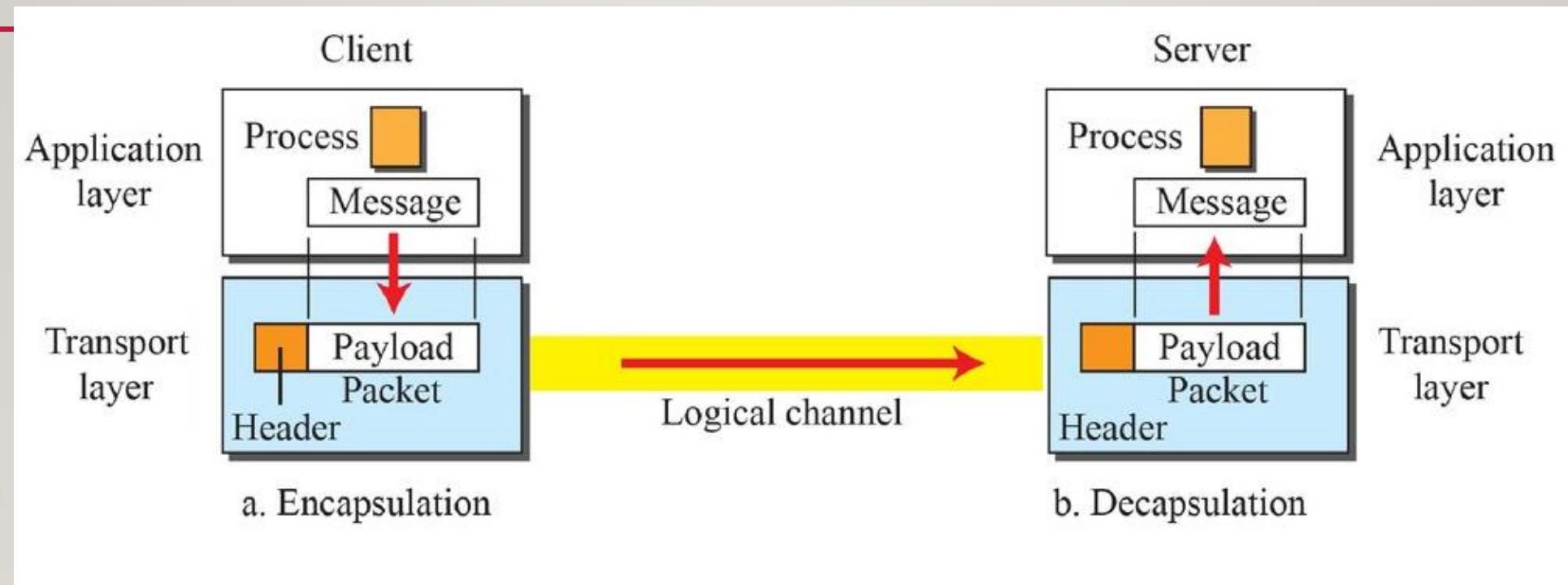


# ADDRESSING- SOCKET ADDRESS

- Transport layer in TCP suite needs both IP address and the port number to make a connection.
- The combination of IP address and port number is called a **socket address**.
- The client socket address defines the client process uniquely.
- The server socket address defines the server process uniquely.



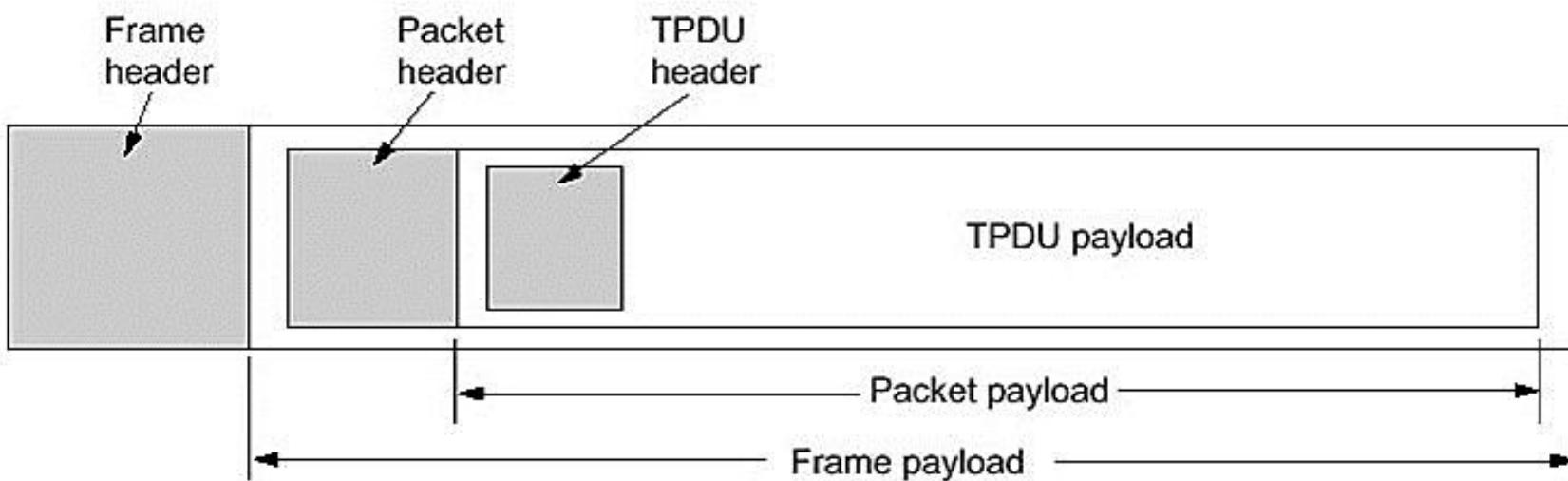
# ENCAPSULATION AND DECAPSULATION



# TPDU

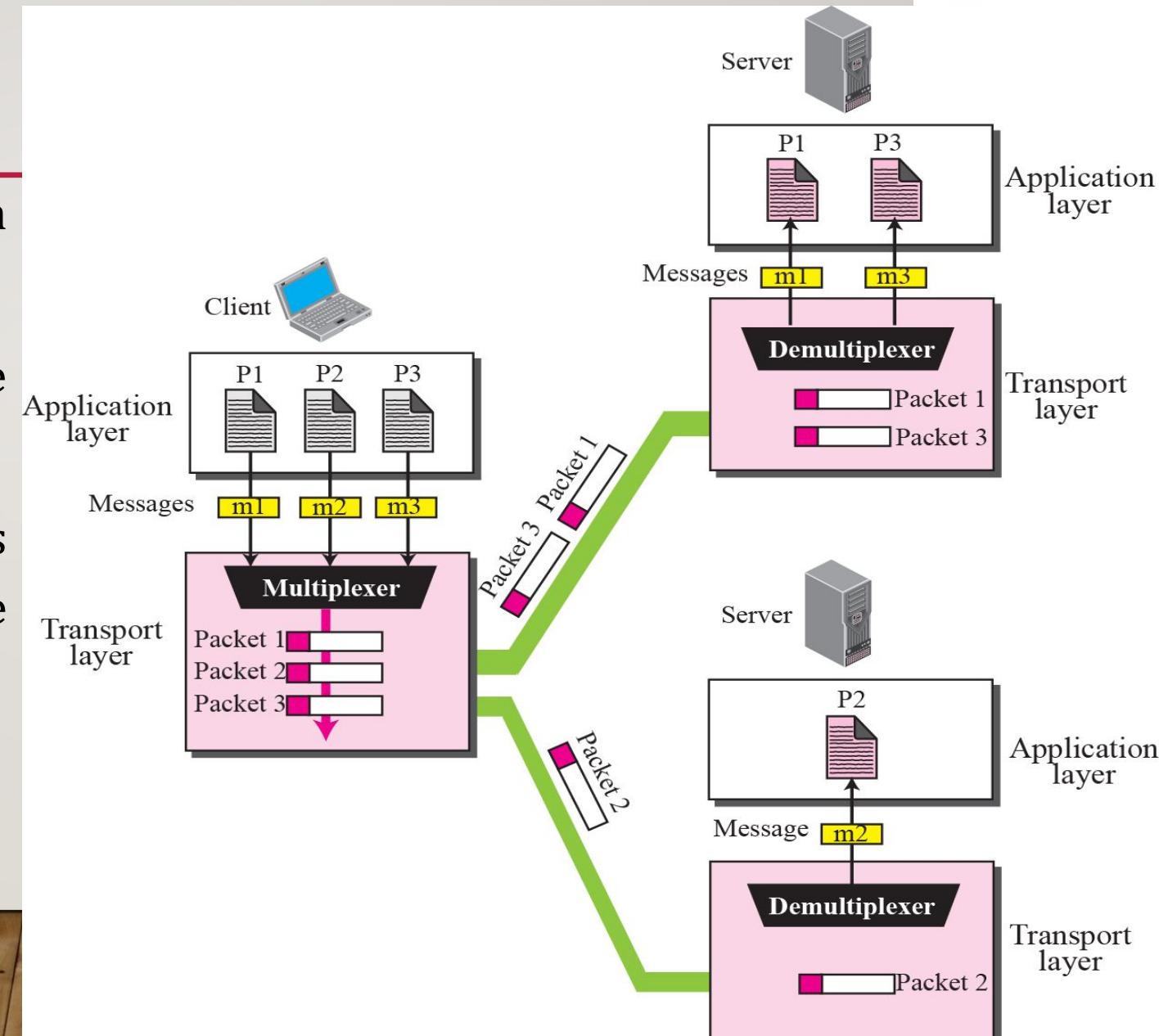
---

- Transport protocol data unit- TPDU



# MULTIPLEXING AND DEMULTIPLEXING

- When an entity accepts items from more than one source ↳ multiplexing.
- When an entity delivers items to more than one source ↳ demultiplexing.
- Transport layer at the source performs multiplexing; transport layer at the destination performs demultiplexing.



# FLOW CONTROL

---

- When the producer **produces items faster** than what can be consumed by the consumer, the **consumer will have to drop some packets**.
- **Pushing and Pulling.**
- **Pushing:** If the sender delivers items whenever they are produced without a prior request from the consumer.
- **Pulling:** If the producer delivers the items after the consumer has requested them.

# ERROR CONTROL

---

- In the internet, network layer is unreliable, so it is essential to make the transport layer reliable.
- Detect and discard **corrupted packets**
- Keep track of **lost and discarded packets** and resend them
- Recognize **duplicate packets** and discard them
- Buffering **out-of-order packets** until the missing packets arrive
  - Sequence numbers
  - Acknowledgements
  - Time-out

# CONGESTION CONTROL

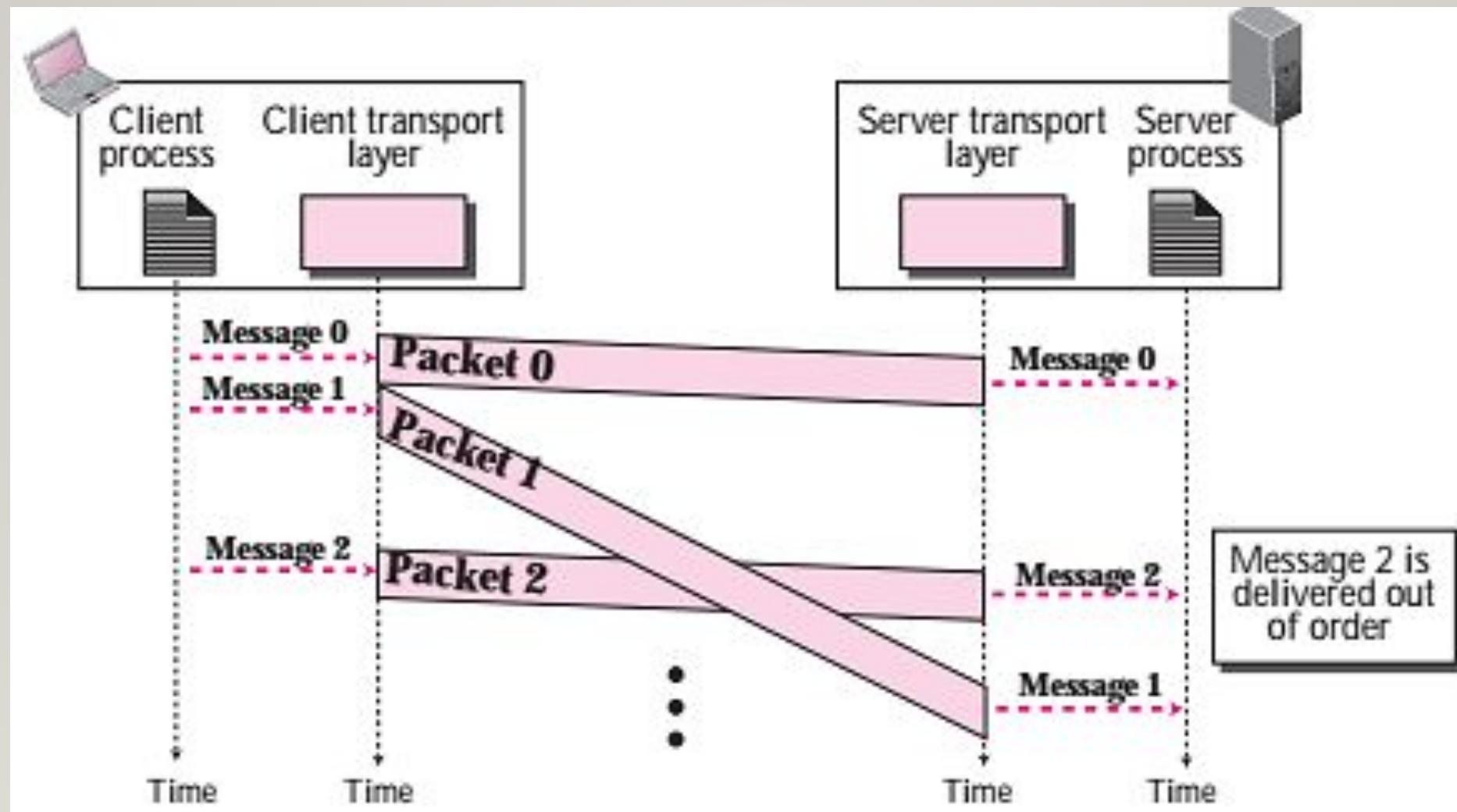
---

- Congestion occurs 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
- Open-loop congestion control
  - Policies are applied to **prevent congestion** before it happens
- Closed-loop congestion control
  - Try to **alleviate congestion** after it happens

# CONNECTIONLESS SERVICE

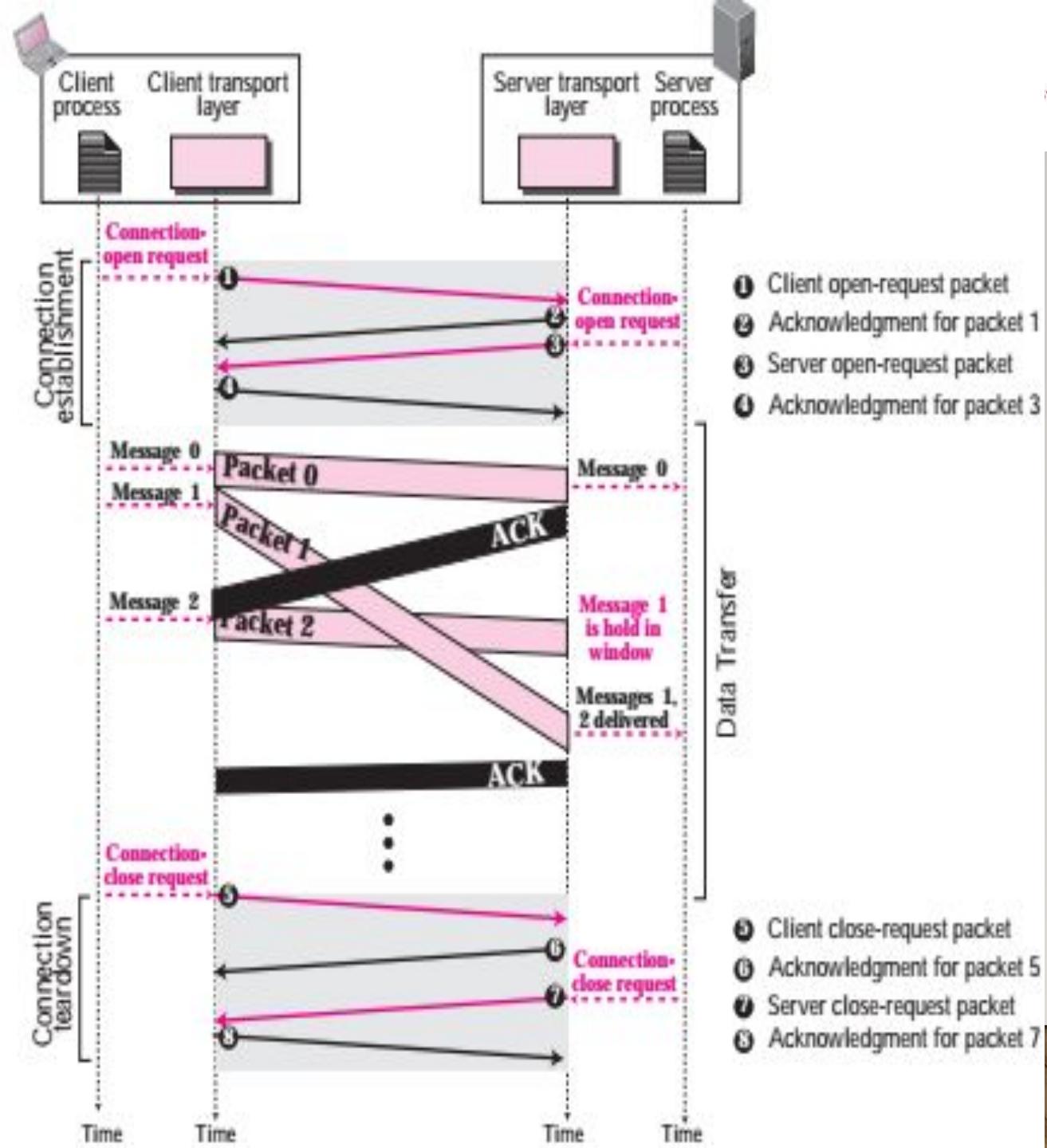
---

- The source process (application program) divides its message into chunks of data
- Chunks are delivered to the connectionless transport protocol one by one
- The transport layer treats each chunk as a single unit without any relation between the chunks
- **No dependency between the packets** at the transport layer
- The packets **may arrive out of order at the destination** and will be delivered out of order to the server process



# CONNECTION-ORIENTED SERVICE

- Connection established
- Data transferred
- Connection teared down



# THANK YOU!!!

---

# COMPUTER NETWORKS

## MODULE 5.2

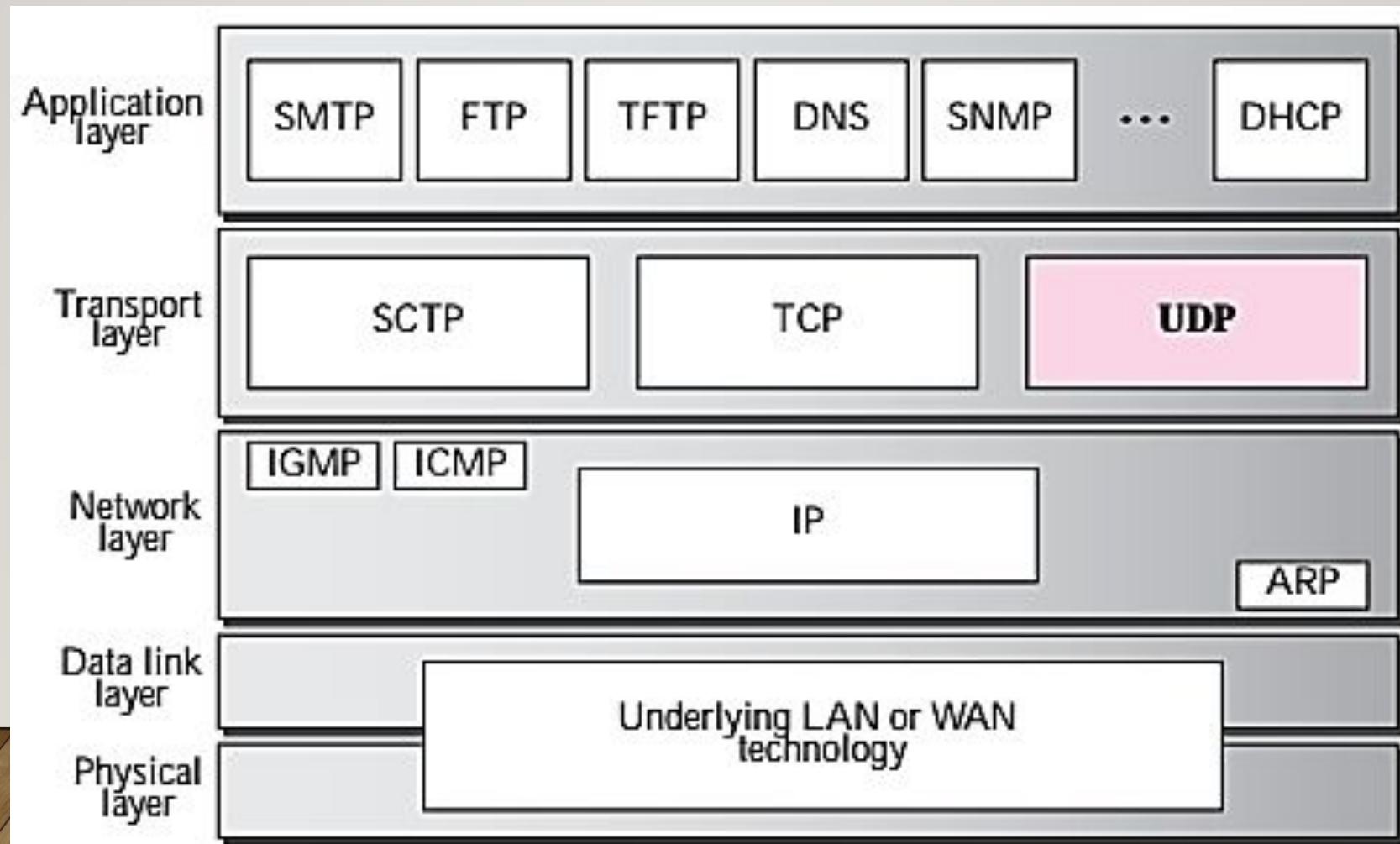
---

MS. JINCY J FERNANDEZ

ASST. PROF, CSE

RSET

# TRANSPORT LAYER PROTOCOLS



# UDP

---

- User Datagram Protocol.
- UDP is a **connectionless unreliable** protocol in transport layer.
- Provides process-to-process communication **using port numbers**.
- **Does not provide flow control, congestion control and does not use acknowledgements.**
- Error control provided to some extent.
- If UDP detects an error in a received packet, it silently drops it.

# WHY UDP?

---

- Simple protocol.
- Minimum overhead.
- A process can use UDP to send a **small message without giving importance to reliability**.
- Sending small messages using UDP requires only less interaction between sender and receiver than using TCP.

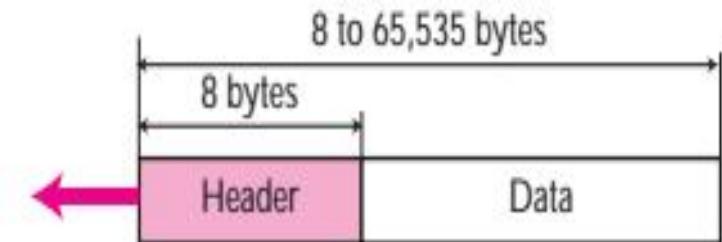
# WELL KNOWN PORTS USED WITH UDP AND TCP

<i>Port</i>	<i>Protocol</i>	<i>UDP</i>	<i>TCP</i>	<i>Description</i>
7	Echo	√		Echoes back a received datagram
9	Discard	√		Discards any datagram that is received
11	Users	√	√	Active users
13	Daytime	√	√	Returns the date and the time
17	Quote	√	√	Returns a quote of the day
19	Chargen	√	√	Returns a string of characters
20, 21	FTP		√	File Transfer Protocol
23	TELNET		√	Terminal Network
25	SMTP		√	Simple Mail Transfer Protocol
53	DNS	√	√	Domain Name Service
67	DHCP	√	√	Dynamic Host Configuration Protocol
69	TFTP	√		Trivial File Transfer Protocol
80	HTTP		√	Hypertext Transfer Protocol
111	RPC	√	√	Remote Procedure Call
123	NTP	√	√	Network Time Protocol
161, 162	SNMP		√	Simple Network Management Protocol

# USER DATAGRAM

---

- UDP packets are called **user datagram**.
- It has a fixed **size header of 8 bytes**.
- Source port number- used by process running on source host.
- Destination port number- used by process running on destination host.
- Total length- defines the total length of the user datagram header plus data.
- Checksum- used to detect errors over the entire user datagram.



a. UDP user datagram



b. Header format

# USER DATAGRAM

---

- Qn. Let the content of a UDP header in hexadecimal format is CB84000D001C001C.
  - 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?

# UDP SERVICES

---

## 1. Process-to-process Communication

- Done using **sockets** – combination of IP addresses & port numbers

## 2. Connectionless Service

- User datagram sent by UDP should be an independent datagram.
- User datagrams are not numbered.
- No connection establishment and no connection termination.
- Each request must be small enough to fit into one user datagram.
- Only processes sending short messages, messages less than **65,507 bytes (65,535 minus 8 bytes for the UDP header and minus 20 bytes for the IP header)** can use UDP.

# UDP SERVICES

---

## 3. No Flow Control

- No window mechanisms.
- Receiver may overflow with incoming packets.

## 4. No Congestion Control

- UDP assumes that packets sent are small and sporadic and won't create any congestion.

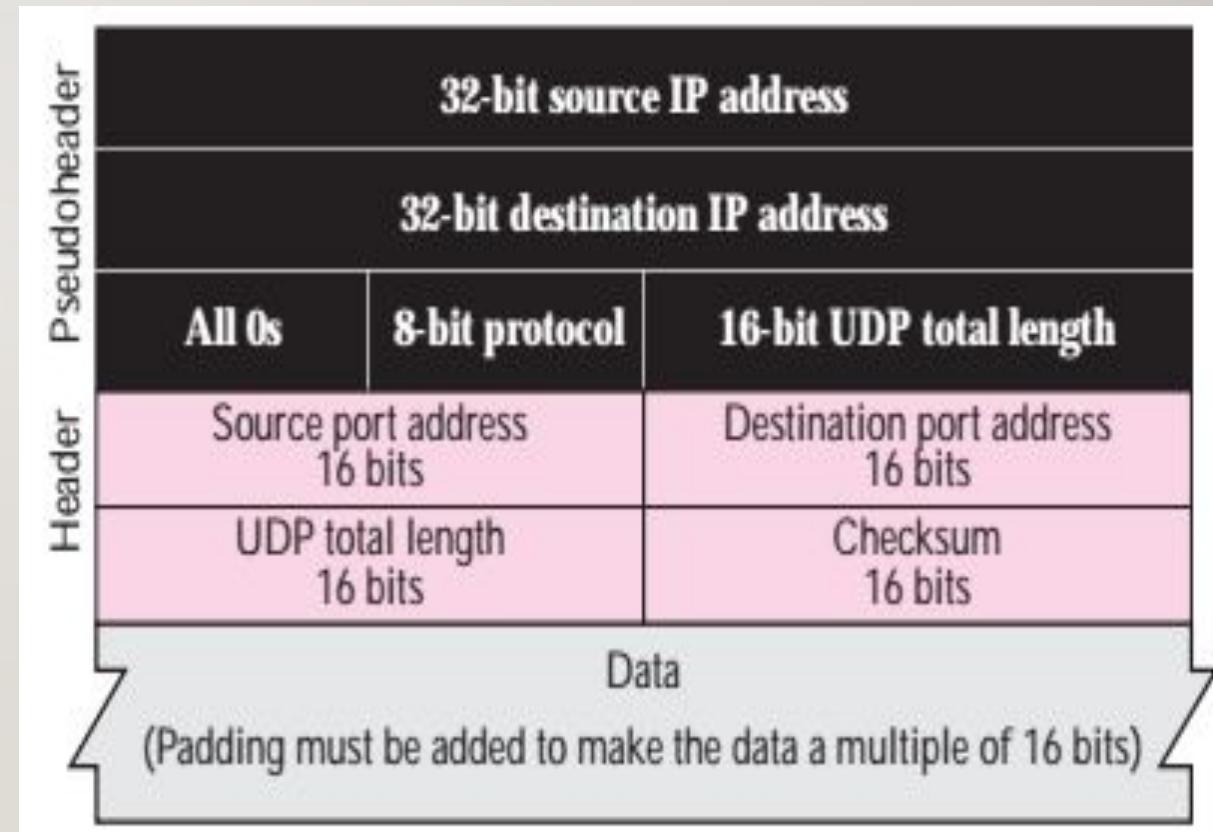
## 5. Error Control – only through checksum

- 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.

# ERROR CONTROL-CHECKSUM

---

- UDP checksum calculation includes three sections: a pseudo header, the UDP header, and the data coming from the application layer.
- To calculate the checksum a pseudoheader (part of the IP header) is prepended to actual header.
- Calculate checksum on data, header & pseudoheader and insert.
- The sender of a UDP packet can choose **not to calculate** the checksum. Then insert all 0s.



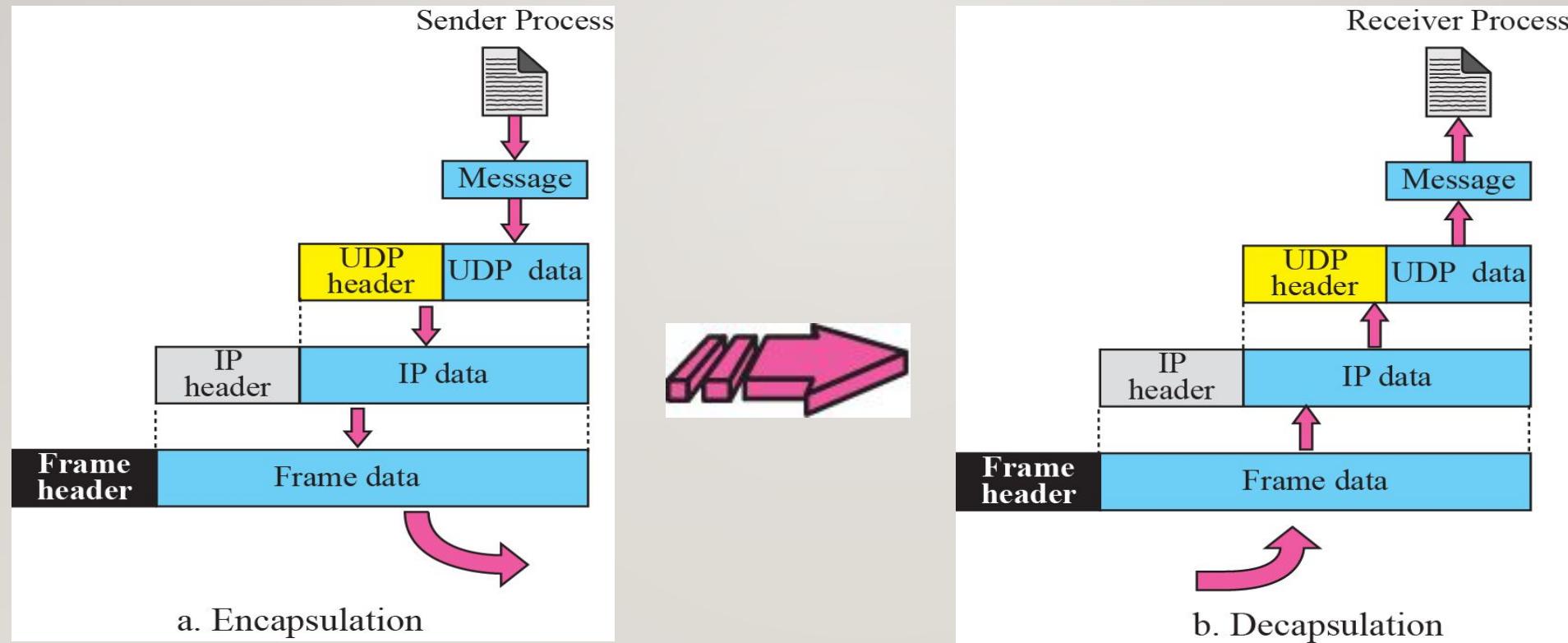
# CHECKSUM CALCULATION OF A UDP DATAGRAM

<b>153.18.8.105</b>			
<b>171.2.14.10</b>			
All 0s	17	15	
1087		13	
15		All 0s	
T	E	S	T
I	N	G	Pad

10011001	00010010	→ 153.18
00001000	01101001	→ 8.105
10101011	00000010	→ 171.2
00001110	00001010	→ 14.10
00000000	00010001	→ 0 and 17
00000000	00001111	→ 15
00000100	00111111	→ 1087
00000000	00001101	→ 13
00000000	00001111	→ 15
00000000	00000000	→ 0 (checksum)
01010100	01000101	→ T and E
01010011	01010100	→ S and T
01001001	01001110	→ I and N
01000111	00000000	→ G and O (padding)
<b>10010110 11101011</b>		→ Sum
<b>01101001 00010100</b>		→ Checksum

# UDP SERVICES

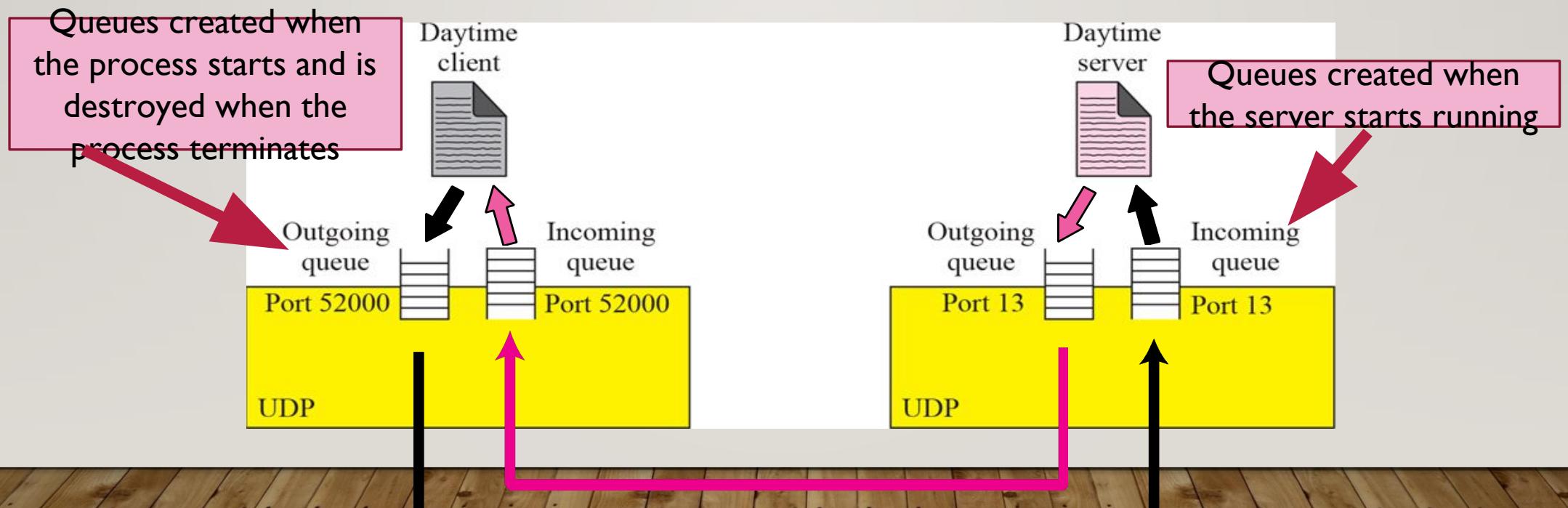
## 6. Encapsulation and Decapsulation



# UDP SERVICES

## 7. Queuing

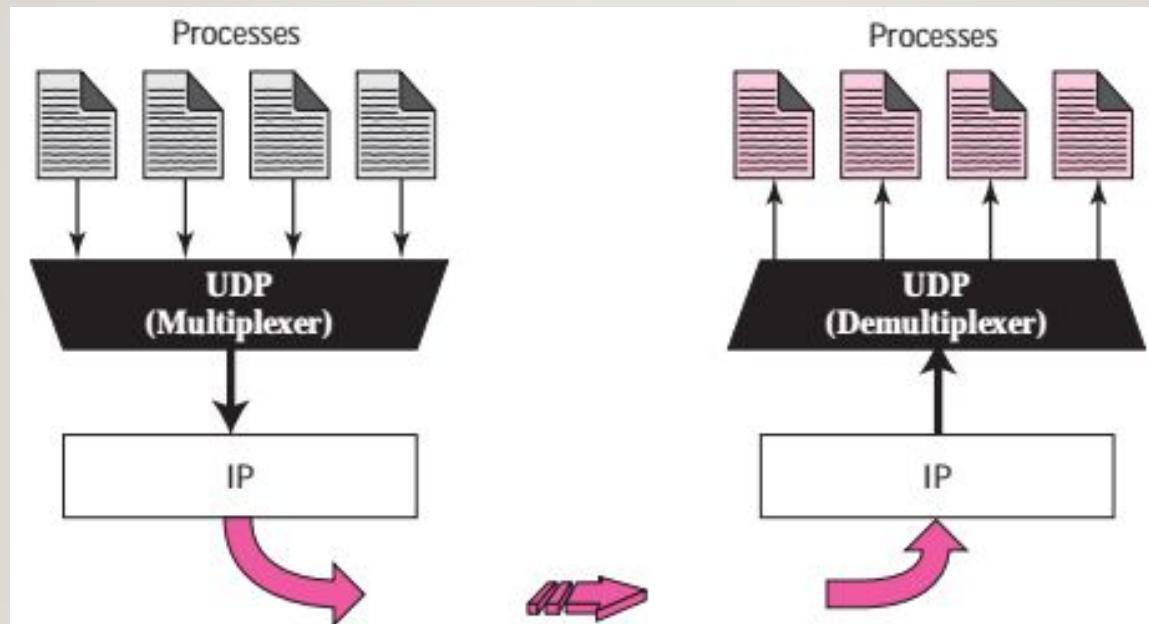
- Queues are associated with ports.



# UDP SERVICES

---

## 8. Multiplexing and Demultiplexing



# APPLICATIONS

---

- Suitable for multicasting applications.
- Used for management processes such as SNMP.
- Used for route updating protocols like RIP.
- Used for real time applications.
- Used in applications with internal flow and error control mechanisms like TFTP.

# TCP

---

- Transmission Control Protocol.
- Connection oriented reliable protocol of transport layer.
- Ensures stream delivery service.
- Provides all most all transport layer services.

# TCP SERVICES

---

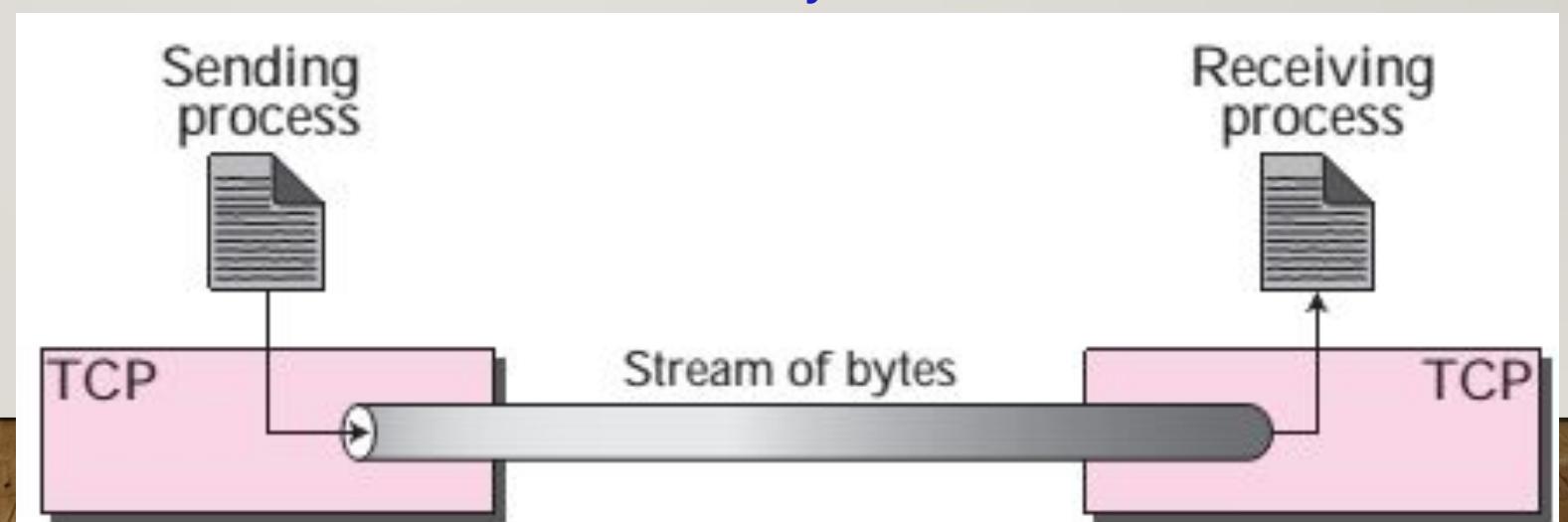
## 1. Process-to-process Communication

- Done using **sockets** – combination of IP addresses & port numbers

# TCP SERVICES

## 2. Stream Delivery Service:

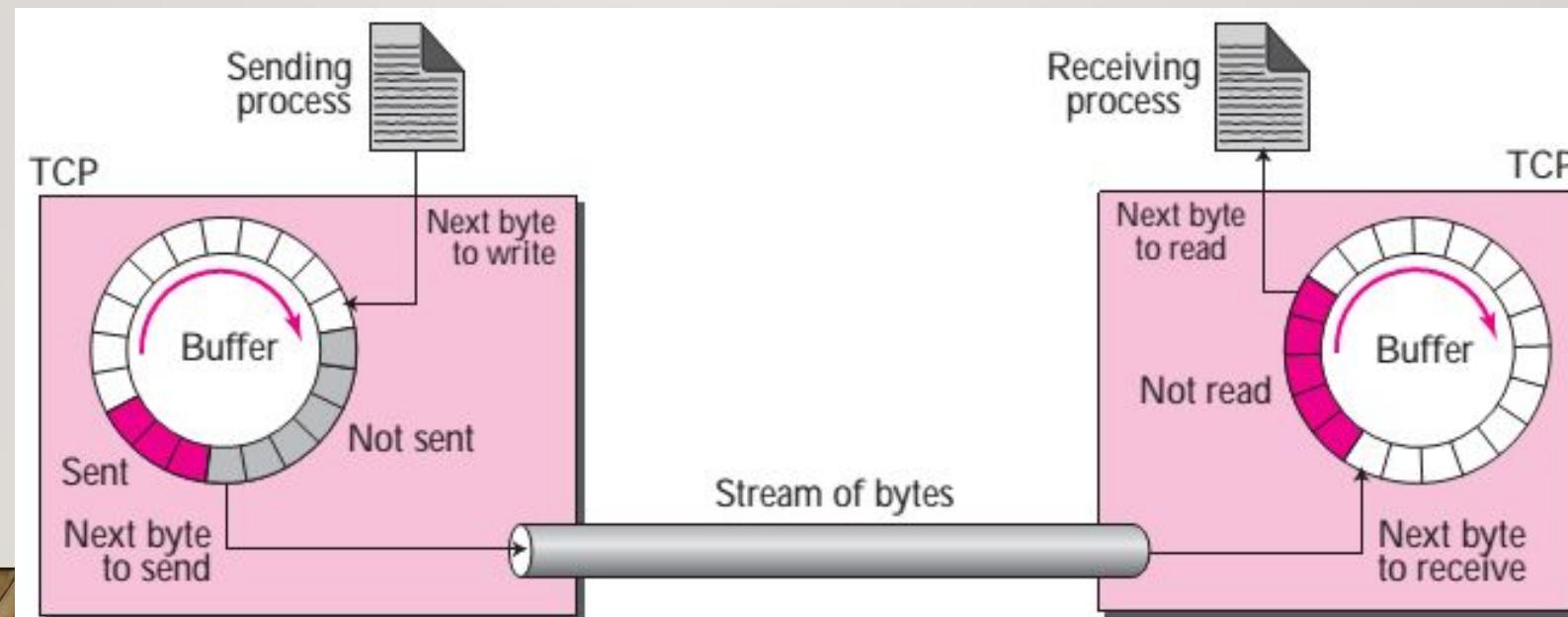
- UDP had messages with pre-defined boundaries.
- TCP is a stream-oriented protocol.
- TCP allows the sending process to deliver data as a stream of bytes and allows the receiving process to obtain data as a stream of bytes.



# TCP SERVICES

### 3. Sending and Receiving Buffers:

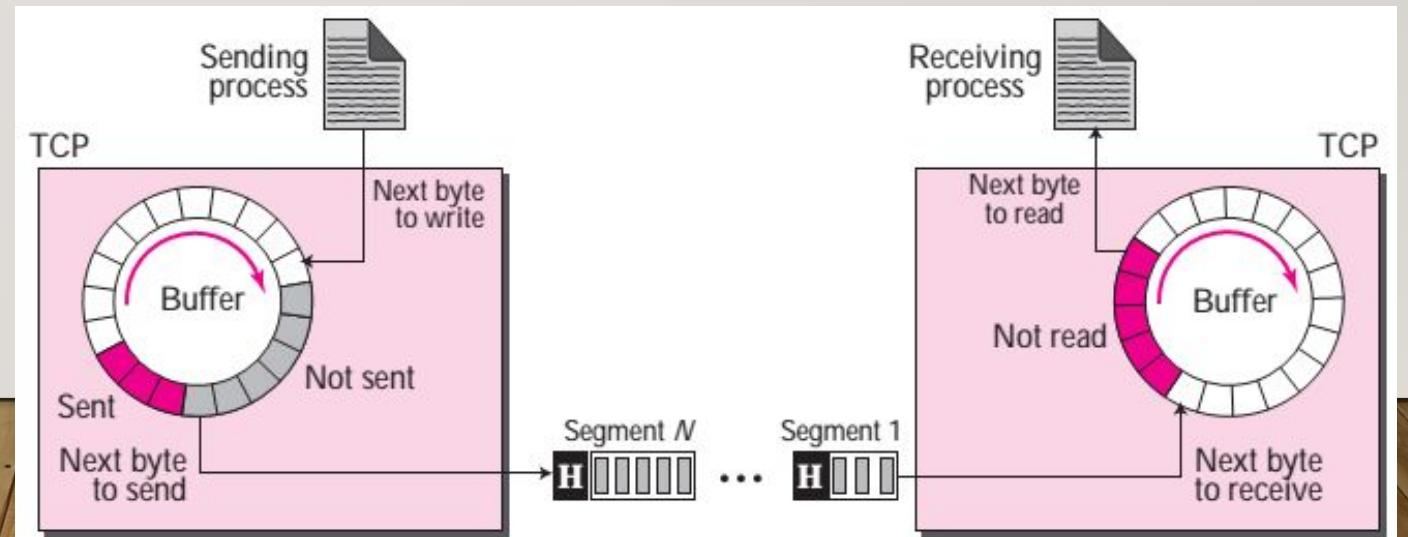
- Since the sending and receiving process may not operate at the same rate, buffers are used for storage.



# TCP SERVICES

## 4. Segments:

- TCP groups bytes together into packets called **segments**.
- TCP adds a header to each segment and delivers the segment to the network layer for transmission.
- Segments are encapsulated in an IP datagram and transmitted.



# TCP SERVICES

## 5. Full Duplex Communication

---

- Segments move in both directions.

## 6. Connection Oriented Service

- TCPs at the sender and the receiver establishes a virtual connection between them before sending the data and terminates the connection after sending the data.

## 7. Multiplexing and Demultiplexing

- Sender performs multiplexing and receiver performs demultiplexing.

## 8. Reliable Service

- Acknowledgements used to ensure safe arrival of data.

# TCP FEATURES

---

- Numbering system
  - Byte Number
  - Sequence Number
  - Acknowledgement Number

# TCP FEATURES

---

## • Byte Number

- TCP numbers all data bytes that are transmitted in a connection.
- When TCP receives bytes of data from a process, TCP stores them in the sending buffer and numbers them.
- TCP chooses an arbitrary number between 0 and  $2^{32} - 1$  for the number of the first byte.
- E.g. if byte number = 1057 and total data to be sent is 6000 bytes, then the bytes are numbered from 1057 to 7056.
- Used for flow control and error control.

# TCP FEATURES

---

- Sequence Number

- TCP assigns a sequence number to each segment.
- The sequence number for each segment is the number of the first byte of data carried in that segment.

<b>Segment 1</b>	→	<b>Sequence Number:</b>	<b>10,001</b>	<b>Range:</b>	<b>10,001</b>	to	<b>11,000</b>
<b>Segment 2</b>	→	<b>Sequence Number:</b>	<b>11,001</b>	<b>Range:</b>	<b>11,001</b>	to	<b>12,000</b>
<b>Segment 3</b>	→	<b>Sequence Number:</b>	<b>12,001</b>	<b>Range:</b>	<b>12,001</b>	to	<b>13,000</b>
<b>Segment 4</b>	→	<b>Sequence Number:</b>	<b>13,001</b>	<b>Range:</b>	<b>13,001</b>	to	<b>14,000</b>
<b>Segment 5</b>	→	<b>Sequence Number:</b>	<b>14,001</b>	<b>Range:</b>	<b>14,001</b>	to	<b>15,000</b>

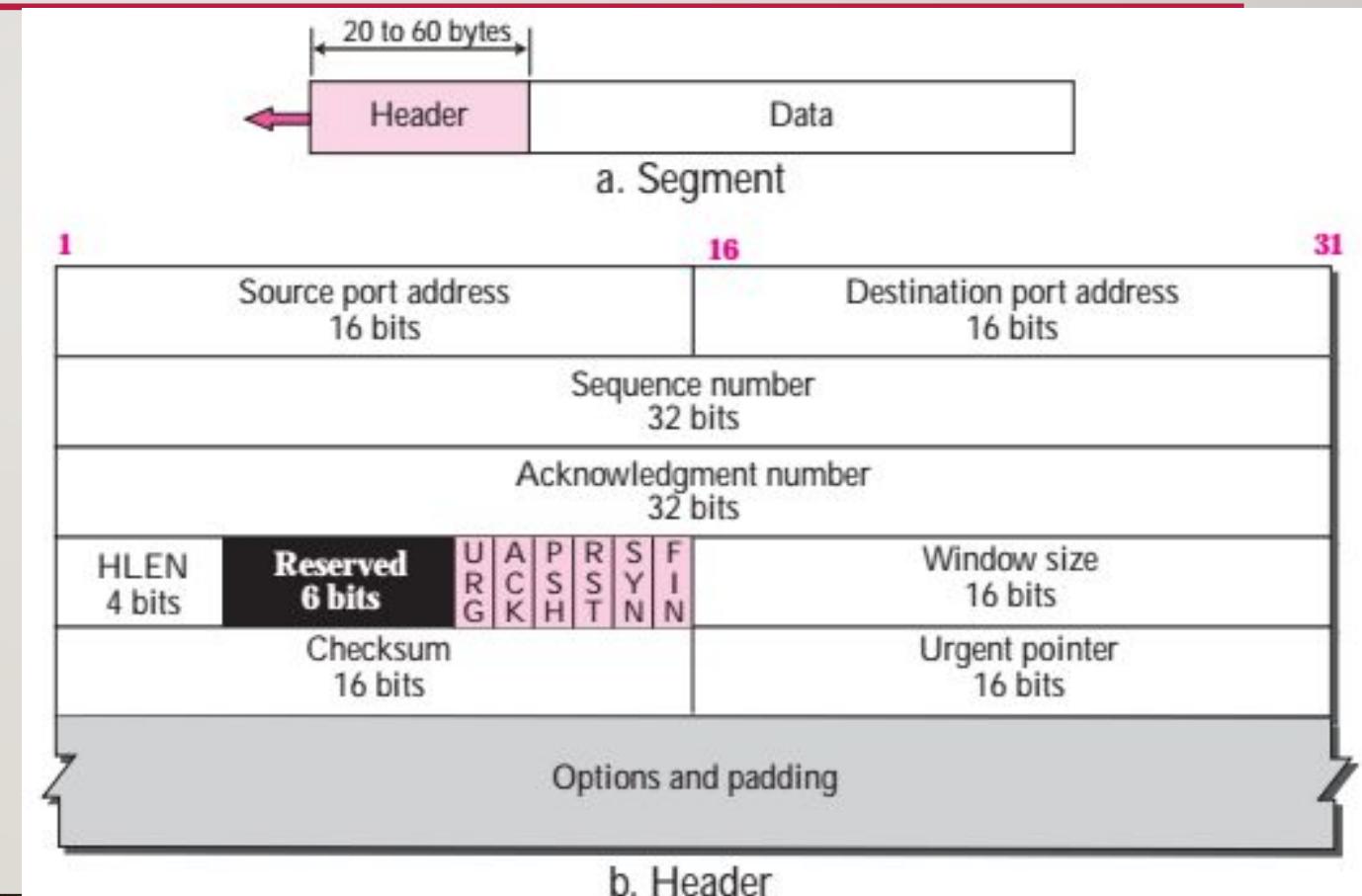
# TCP FEATURES

---

- Acknowledgment number
  - Acknowledgment number is used by a party to confirm the bytes it has received.
  - The acknowledgment number defines the number of the next byte that the party expects to receive.
  - The acknowledgment number is cumulative.
  - If a party uses 5,641 as an acknowledgment number, it has received all bytes from the beginning up to 5,640.

# TCP SEGMENT FORMAT

- A packet in TCP is called a segment.
  - Consists of header and data.



# FIELDS IN THE SEGMENT HEADER FORMAT

---

- Source port number
  - 16-bit field that defines the port number of the host that is sending the segment.
- Destination port number
  - 16-bit field that defines the port number of the host that is receiving the segment.
- Sequence number
  - 32-bit field defines the number assigned to the first byte of data contained in this segment.
  - During connection establishment each party uses a random number generator to create an initial sequence number (ISN).The ISN is usually different in each direction.

# FIELDS IN THE SEGMENT HEADER FORMAT

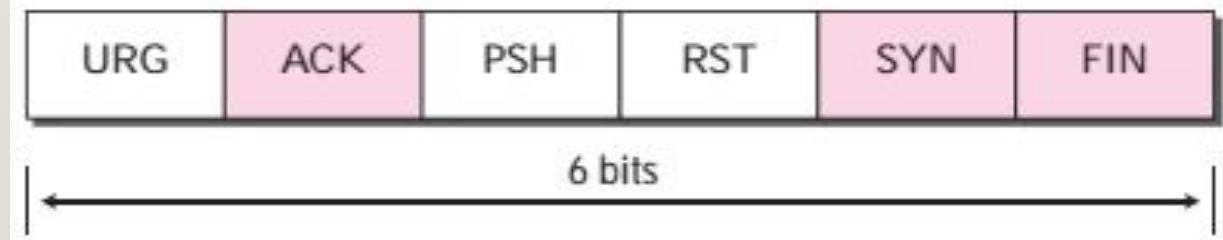
- **Acknowledgement number**
  - 32-bit field defines the byte number that the receiver of the segment is expecting to receive from the other party.
  - If the receiver has successfully received byte number  $x$ , it returns  $x + 1$  as the acknowledgment number .
  - Acknowledgment and data can be piggybacked together.
- **Header length**
  - 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**
  - Reserved for future use.

# FLAGS

- URG- it is set to 1 if urgent pointer is in use.
- ACK- it is set to 1 to indicate that the segment contains an acknowledgement.
- PSH- it is set to 1 to indicate pushed data.
- RST- it is set to 1 to indicate a connection is being reset, rejected or refused.
- SYN- it is set to 1 to indicate that a new connection is to be established.
- FIN- it is set to 1 to indicate that a connection is been released.

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



# FIELDS IN THE SEGMENT HEADER FORMAT

- **Window size**
  - Defines the window size of the sending TCP in bytes.
  - The length of this field is 16 bits, so the maximum size of the window is 65,535 bytes.
  - This value is referred to as the receiving window and is determined by the receiver.
- **Urgent pointer**
  - 16-bit field, which is valid only if the urgent flag is set. It is used when the segment contains urgent data.
  - It defines a value that must be added to the sequence number to obtain the number of the last urgent byte in the data section of the segment.

# FIELDS IN THE SEGMENT HEADER FORMAT

---

- Checksum
  - Calculation is same as that in UDP.
- Options
  - There can be up to 40 bytes of optional information.

# PHASES IN A TCP CONNECTION

---

- TCP is a connection oriented protocol.
- It establishes a virtual path between the source and the destination.
- All the segments are sent along this virtual path.
- Transmission is done in three phases:
  - Connection Establishment
  - Data Transfer
  - Connection Termination

# CONNECTION ESTABLISHMENT

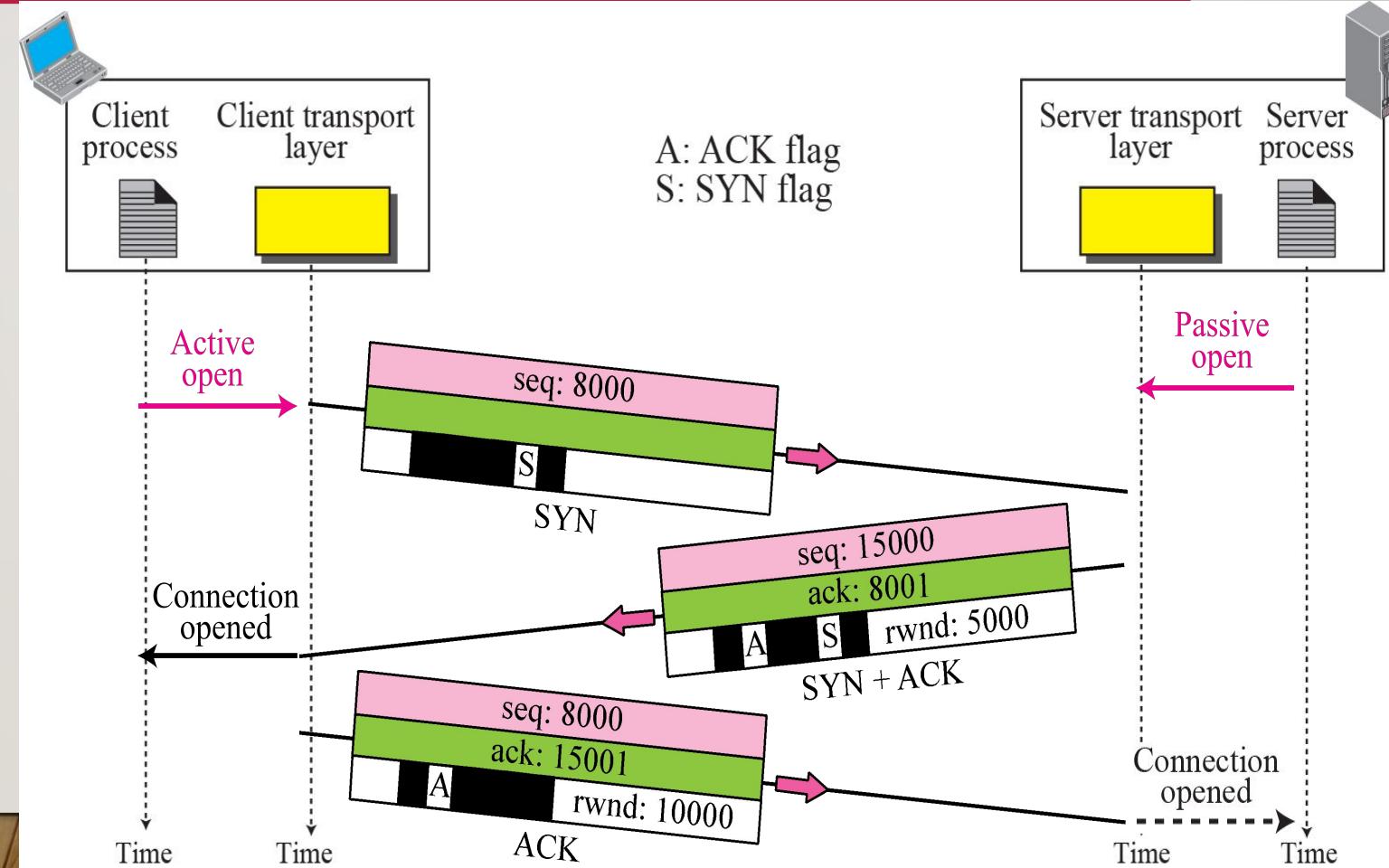
---

- TCP transmits data in full-duplex mode.
- Each party must **initialize communication** and get approval from the other party before data is transferred.
- Connection establishment in TCP is called **three-way handshaking**.

# THREE-WAY HANDSHAKING

- Step 1:

- Client sends SYN segment? for synchronization of sequence numbers.
- Random number as the first sequence number (ISN) and send it to server.
- This segment does not contain any acknowledgement number.
- A SYN segment cannot carry data, but it consumes one sequence number.



# THREE-WAY HANDSHAKING

---

- Step 2:
  - Server sends the second segment, a SYN+ACK segment with two flag bits set.
  - Uses this segment to initialize a sequence number for numbering the byte sent from the server to the client.
  - Server also acknowledge the receipt of SYN segment from the client by setting the ACK flag.
  - Needs to define the receive window, rwnd.
  - A SYN+ACK segment cannot carry data.

# THREE-WAY HANDSHAKING

---

- Step 3:
  - Client sends the third segment? an ACK segment.
  - An ACK segment does not consume any sequence number if it does not carry data.
  - Some implementations allow this segment to carry first chunk of data from the client.

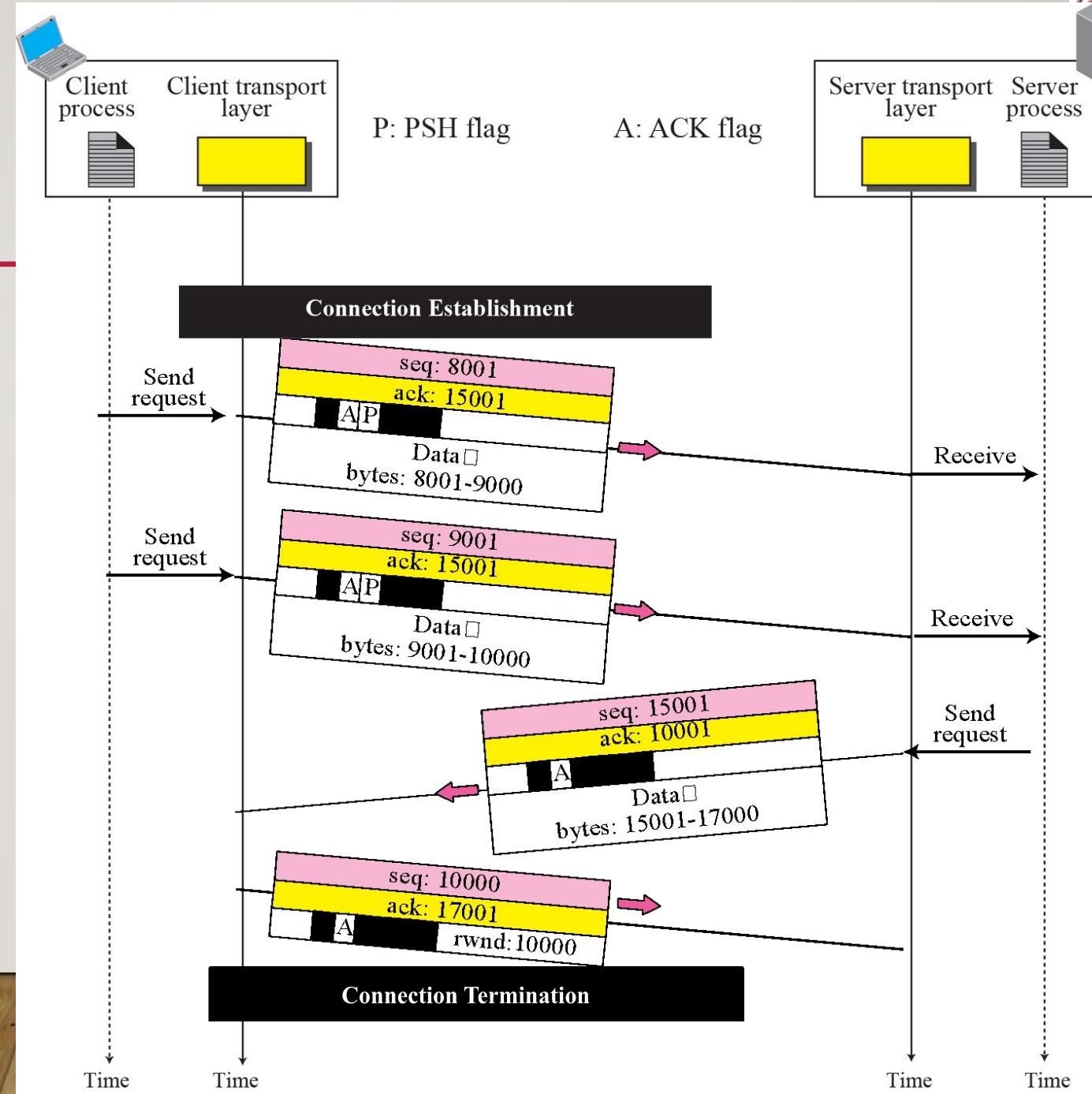
## THREE-WAY HANDSHAKING- SYN FLOODING ATTACK

---

- Connection establishment procedure in TCP is susceptible to a serious security problem called SYN flooding attack.
- When one or more malicious attackers send a large number of SYN segments to a server pretending that each of them is coming from a different client by faking the source IP addresses in the datagrams.
- The server, assuming that the clients are issuing an active open, allocates the necessary resources.
- TCP server then sends the SYN+ACK segments to the fake clients, which are lost.
- Server eventually runs out of resources and may be unable to accept connection requests from valid clients.

# DATA TRANSFER

- After connection is established, bidirectional data transfer can take place.



# DATA TRANSFER- URGENT DATA

---

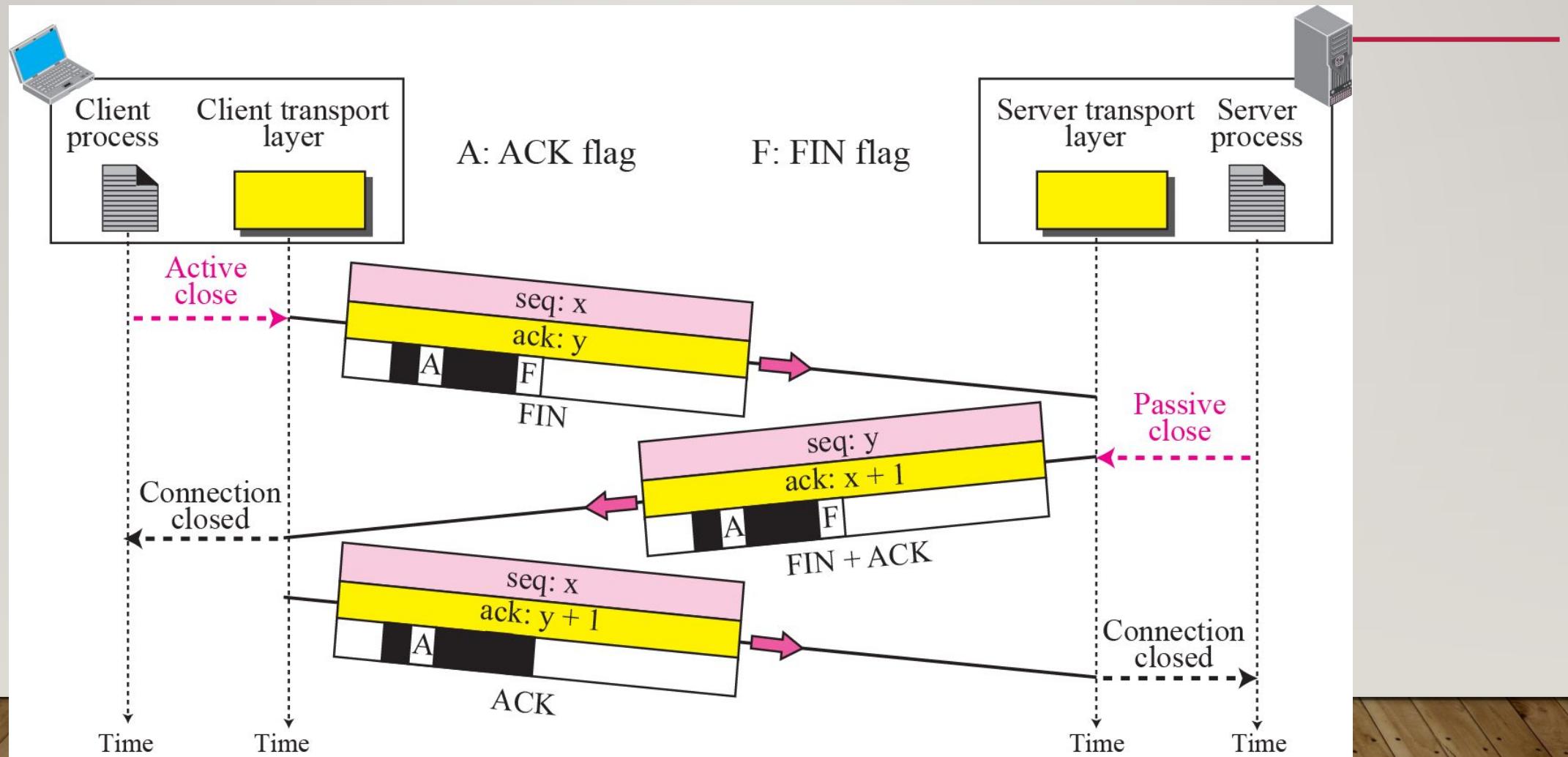
- Used when an application program needs to send urgent bytes.-> send a segment with URG bit set.
- Urgent data is inserted at the beginning of the segment.
- Rest of the segment can contain normal data from the buffer.
- Urgent pointer field in the header defines the end of the urgent data (lasts byte of the urgent data)

# CONNECTION TERMINATION

---

- Any of the two parties can close a connection
  
- Three-way handshaking
  - Both ends agree and stop sending data.
  
- Four-way handshaking with a half-close option
  - One end stop sending data, but still is able to receive data

# THREE-WAY HANDSHAKING- TERMINATION



# THREE-WAY HANDSHAKING- TERMINATION

---

- Step 1:
  - Client TCP sends the first segment, a FIN segment in which FIN flag is set.
  - A FIN segment can include the last chunk of data sent by the client or it can be just a control segment.
  - FIN segment consumes one sequence number if it does not carry data.

# THREE-WAY HANDSHAKING- TERMINATION

---

- Step 2:
  - Server TCP, after receiving the FIN segment 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.
  - FIN+ACK segment consumes one sequence number if it does not carry data.

# THREE-WAY HANDSHAKING- TERMINATION

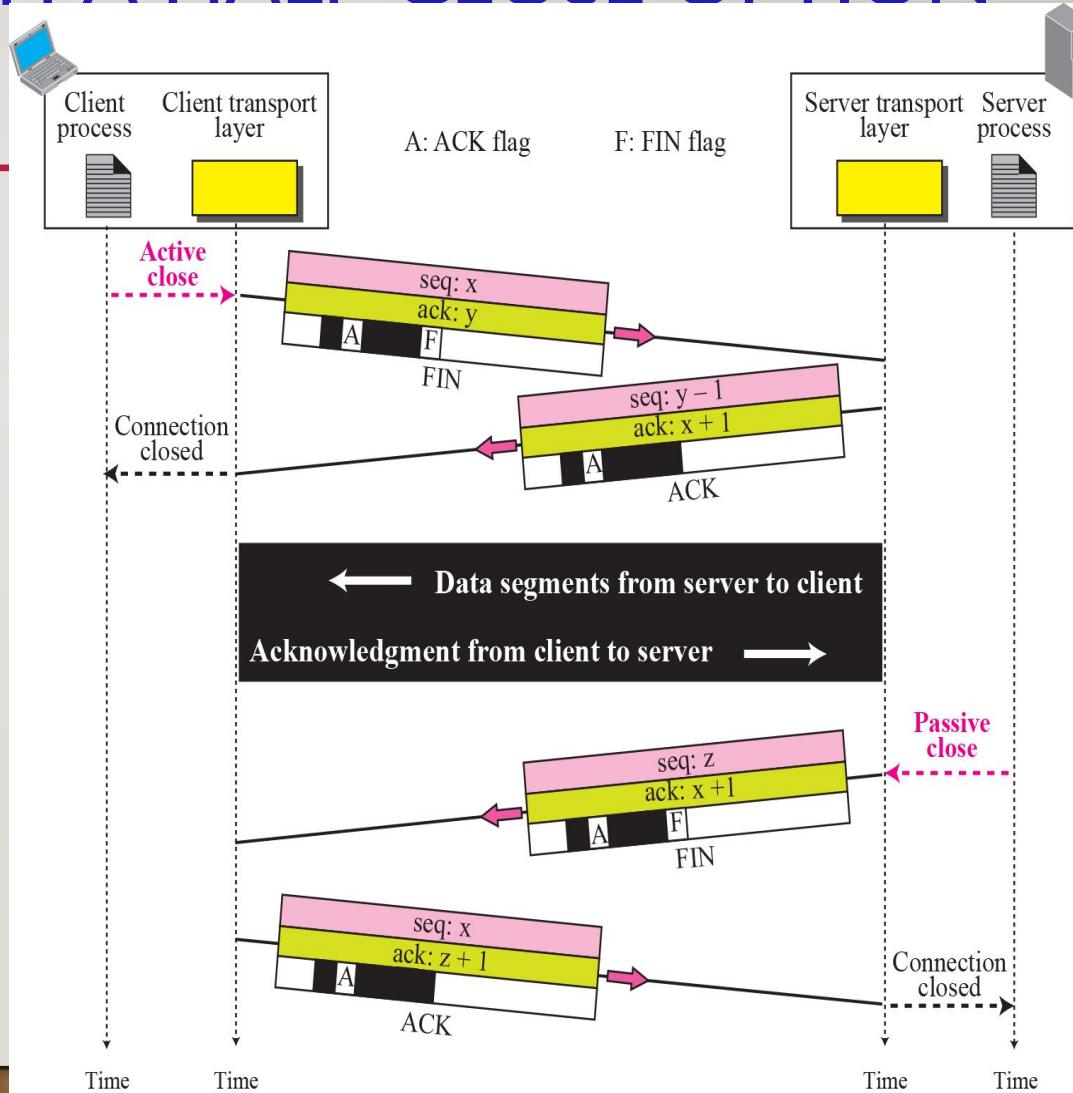
---

- Step 3:
  - 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 acknowledgement number.
  - This segment does not carry data and consumes no sequence numbers.

# FOUR-WAY HANDSHAKING WITH A HALF-CLOSE OPTION

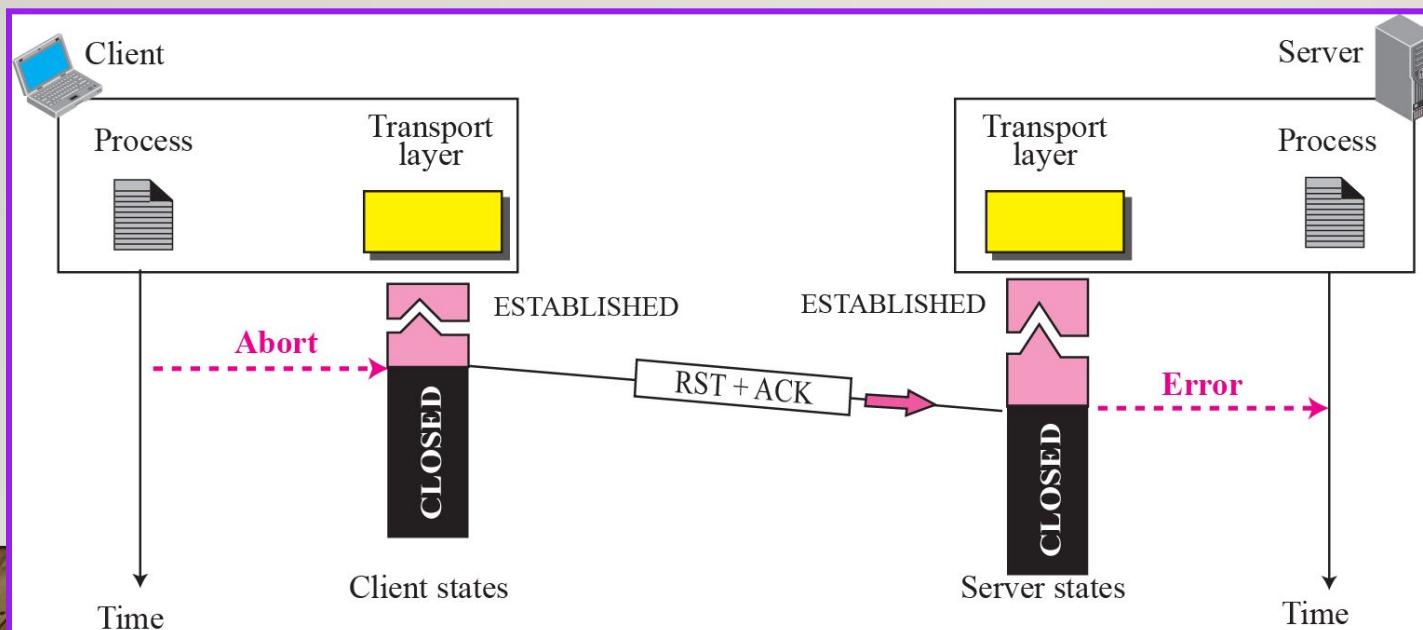
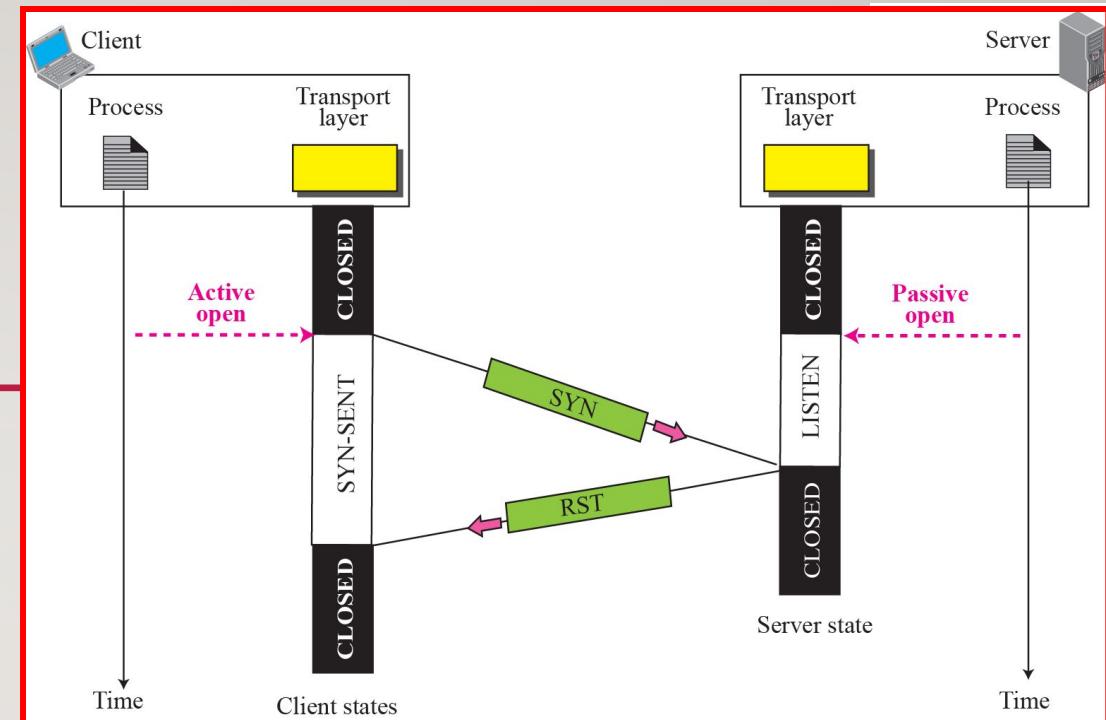
- Half Close:

- One end can stop sending data while still receiving data.
- Either the server or client can issue a half close request.
- E.g. client half closes the connection by ending a FIN segment.
- Server accepts the half close by sending the ACK segment.
- When server has sent all the data, it sends a FIN segment, which is acknowledged



# CONNECTION RESET

- TCP at one end may
  - Deny a connection request
  - Abort an existing connection
  - or terminate an idle

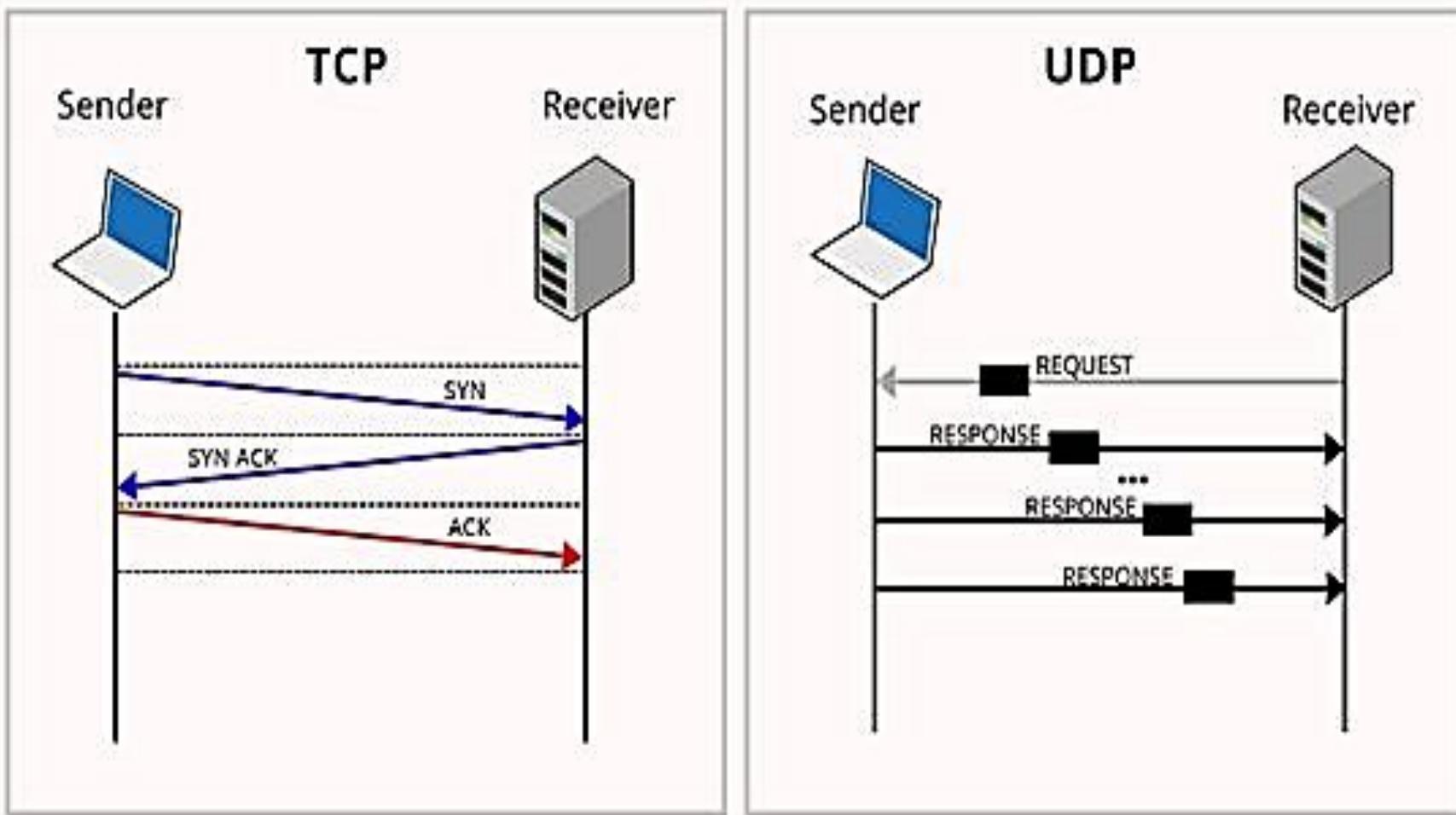


# APPLICATIONS

---

- Bootstrap Protocol
- Dynamic Host Configuration Protocol (DHCP)
- Domain Name System (DNS)
- E-mail
- File Transfer Protocol (FTP)
- IP filtering and network address translation
- Secure Shell (SSH) applications such as PuTTY
- Remote Desktop (RDP) applications such as Microsoft Remote Desktop Client
- Secure File Transfer Protocol (SFTP)
- Secure Copy Protocol (SCP) applications such as WinSCP

# TCP Vs UDP Communication



TCP	UDP
Secure	Unsecure
Connection-Oriented	Connectionless
Slow	Fast
Guaranteed transmission	No Guarantee
Used by critical applications	Used by real-time applications
Packet reorder mechanism	No reorder mechanism
Flow control	No flow control
Error Checking	No Error Checking
20 Bytes Header	8 Bytes Header
Acknowledgement Mechanism	No Acknowledgement
Three-way handshake (SYN, SYN-ACK, ACK)	No handshake
DNS, HTTP, HTTPS, FTP, SMTP, Telnet, SNMP	DNS, DHCP, TFTP, SNMP, RIP, VOIP

**THANK YOU!!!**

---

# COMPUTER NETWORKS

## MODULE 5.3

---

MS. JINCY J FERNANDEZ

ASST. PROF, CSE

RSET

# TCP RETRANSMISSION POLICY

- 
- Retransmission ↗ resending the packets over the network that have been either lost or damaged to provide reliable communication.
  - Retransmission means the data packets have been lost, which leads to a lack of acknowledgment. This lack of acknowledgment triggers a timer to timeout.
  - **The sender sets the timeout period for an ACK. The timeout period can be of two types:**
    - **Too short:** If the timeout period is too short, then the retransmissions will be wasted.
    - **Too long:** If the timeout period is too long, then there will be an excessive delay when the packet is lost.

# TCP RETRANSMISSION POLICY

---

- In modern implementations, a segment is retransmitted on two occasions: when a retransmission timer expires or when the sender receives three duplicate ACKs.
- No retransmission occurs for segments that do not consume sequence numbers.
- No retransmission timer is set for an ACK segment.

# TCP RETRANSMISSION POLICY

---

- Retransmission After RTO:
  - TCP maintains one retransmission time-out (RTO) timer for all outstanding (sent, but not acknowledged) segments.
  - When the timer matures, the earliest outstanding segment is retransmitted.
  - no time-out timer is set for a segment that carries only an Acknowledgment
  - The value of RTO is dynamic in TCP and is updated based on the round-trip time (RTT) of segments. Round trip time is the time required for the packet to travel from the source to the destination and then come back again.
  - The RTT can vary depending upon the network's characteristics, i.e., if the network is congested, it means that the RTT is very high.

# TCP RETRANSMISSION POLICY

---

- Retransmission After Three Duplicate ACK Segments:
  - Retransmission of a segment using RTO is sufficient if the value of RTO is not very large.
  - Sometimes, one segment is lost, and the receiver receives so many out-of-order segments that they cannot be saved (limited buffer size).
  - Most implementations today follow the three-duplicate-ACKs rule and retransmit the missing segment immediately.

# TCP RETRANSMISSION POLICY

---

- 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.
  - The out-of-order segments are not delivered to the process. TCP guarantees that data are delivered to the process in order..

# TCP CONGESTION CONTROL

---

- Congestion occurs if the transport entities on many machines send too many packets into the network too quickly.
- Degrade the performance of the network as packets are delayed and lost.
- Controlling congestion is the combined responsibility of the network and transport layers.
- Congestion occurs at routers; so it is detected at the network layer.
- Congestion is caused by traffic sent into the network by the transport layer.
- Effective way to control congestion is for the transport layers to send packets into the network more slowly

# TCP CONGESTION CONTROL

---

- TCP is an end-to-end protocol that uses the service of IP.
- The congestion in the router is in the IP territory and should be taken care of by IP.
- TCP uses a **congestion window (cwnd)**, and a congestion policy that avoid congestion
- The size of cwnd is controlled by the congestion situation in the network.
- cwnd and rwnd together define the size of the send window in TCP.
- cwnd is related to the congestion in the middle.
- rwnd is related to the congestion at the end.
- Actual size of the send window= minimum(cwnd, rwnd)

# TCP CONGESTION CONTROL- CONGESTION DETECTION

---

- TCP sender uses the occurrence of two events as signs of congestion in the network: **time out** and receiving **three duplicate ACKs**.
- **Timeout:** If a TCP sender does not receive an ACK for a segment or a group of segments before the timeout occurs, it assumes that corresponding segment or group of segments are lost and the loss is due to congestion.
- **Three duplicate ACKs:** When a TCP receiver sends a duplicate ACK, it is the sign that a segment has been delayed, but sending three duplicate ACKs is the sign of a missing segment, which can be due to congestion in the network.

# TCP CONGESTION CONTROL- CONGESTION POLICIES

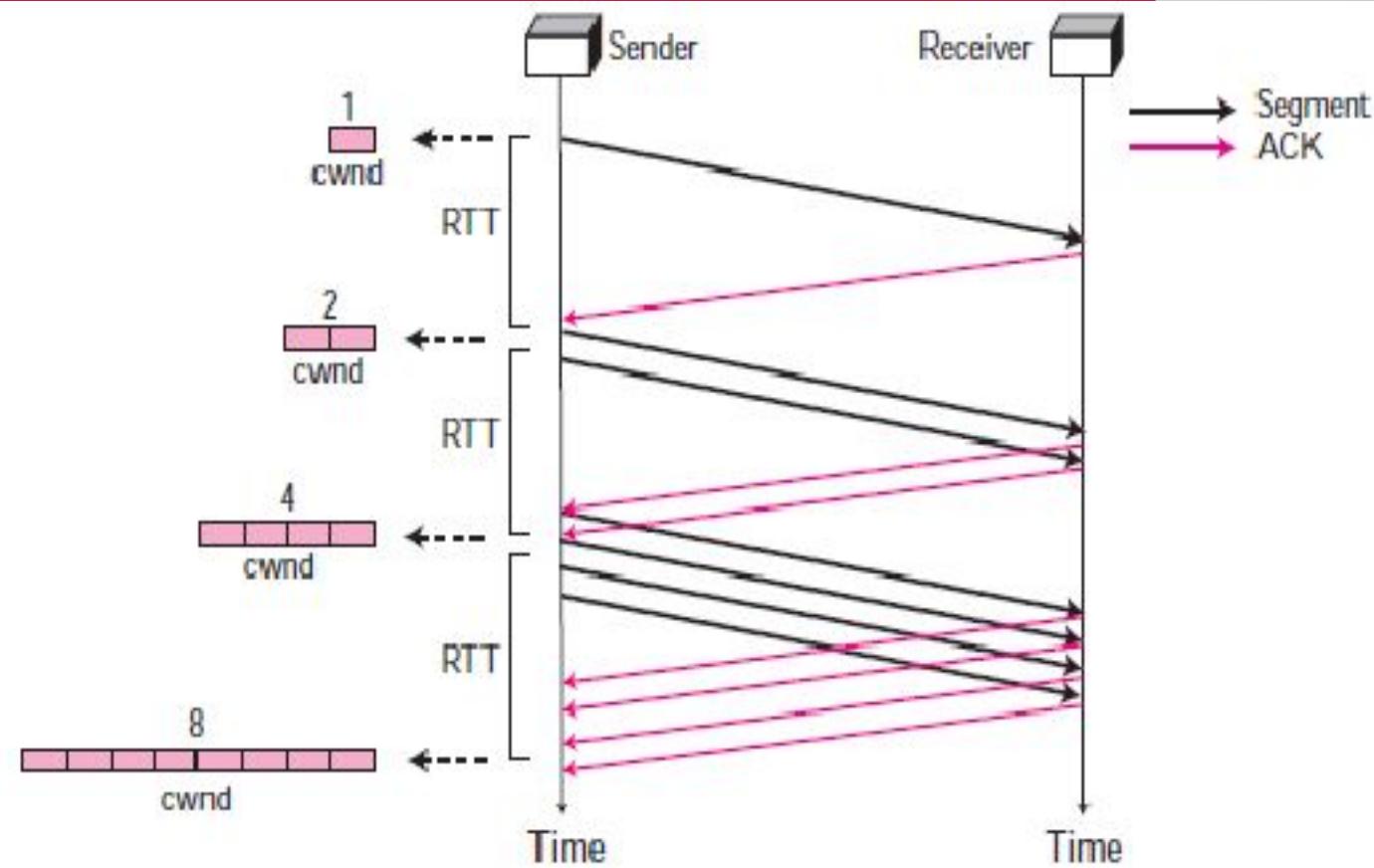
---

- Slow start: Exponential Increase
  - In this phase, after every RTT the congestion window size increments exponentially.
  - The algorithm starts slowly but grows exponentially.
  - Assume that each segment is of same size and carries MSS bytes.

Start	
After 1 RTT	
After 2 RTT	
After 3 RTT	

# TCP CONGESTION CONTROL- CONGESTION POLICIES

- Slow start: Exponential Increase



# TCP CONGESTION CONTROL- CONGESTION POLICIES

---

- Slow start: Exponential Increase
  - A slow start cannot continue indefinitely; use a threshold to stop.
  - Sender keeps track of the threshold.
  - When the size of the window in bytes reaches the threshold, slow start stops.
  - The size of the congestion window increases exponentially until it reaches the threshold.
  - Threshold = Maximum number of TCP segments that receiver window can accommodate / 2  
$$= (\text{Receiver window size} / \text{Maximum Segment Size}) / 2$$

# TCP CONGESTION CONTROL- CONGESTION POLICIES

---

- Congestion avoidance: Additive Increase
  - With slow start algorithm, the size of the congestion window increases exponentially.
  - To avoid congestion before it happens, slow down this exponential growth.
  - The size of the congestion window increases additively instead of exponentially.
  - When the size of the congestion window reaches the slow start threshold, slow start phase stops and additive phase begins.

# TCP CONGESTION CONTROL- CONGESTION POLICIES

---

- Congestion avoidance: Additive Increase

Initially cwnd = i

After 1 RTT, cwnd = i+1

2 RTT, cwnd = i+2

3 RTT, cwnd = i+3

Start	cwnd=1
After 1 RTT	cwnd=cwnd+1= 1+1 =2
After 2 RTT	cwnd= cwnd+1 =2+1=3
After 3 RTT	cwnd= cwnd+1=3+1=4

# TCP CONGESTION CONTROL- CONGESTION POLICIES

---

- Congestion Detection Phase : multiplicative decrement
  - If congestion occurs, the congestion window size is decreased.
  - If a time-out occurs
    - 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.
  - If three ACKs are received,
    - It sets the value of the threshold to one-half of the current window size.
    - It sets *cwnd* to the value of the threshold
    - It starts the congestion avoidance phase.

**THANK YOU!!!**

---

# **COMPUTER NETWORKS**

## **MODULE 5.4**

---

MS. JINCY J FERNANDEZ

ASST. PROF, CSE

RSET

# 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, file access and transfer, access to system resources, surfing the world wide web, and network management.
- The application layer is responsible for providing services to the user.

# SERVICES OF APPLICATION LAYER

---

- **Network Virtual terminal:**

- An application layer allows a user to log on to a remote host. To do so, the application creates a software emulation of a terminal at the remote host. The user's computer talks to the software terminal, which in turn, talks to the host. The remote host thinks that it is communicating with one of its own terminals, so it allows the user to log on.

- **File Transfer, Access, and Management (FTAM):**

- An application allows a user to access files in a remote computer, to retrieve files from a computer and to manage files in a remote computer.

# SERVICES OF APPLICATION LAYER

---

- **Addressing:**
  - To obtain communication between client and server, there is a need for addressing. When a client made a request to the server, the request contains the server address and its own address. The server response to the client request, the request contains the destination address, i.e., client address. To achieve this kind of addressing, DNS is used.
- **Mail Services:**
  - An application layer provides Email forwarding and storage.
- **Directory Services:**
  - An application contains a distributed database that provides access for global information about various objects and services.

# FILE TRANSFER PROTOCOL

---

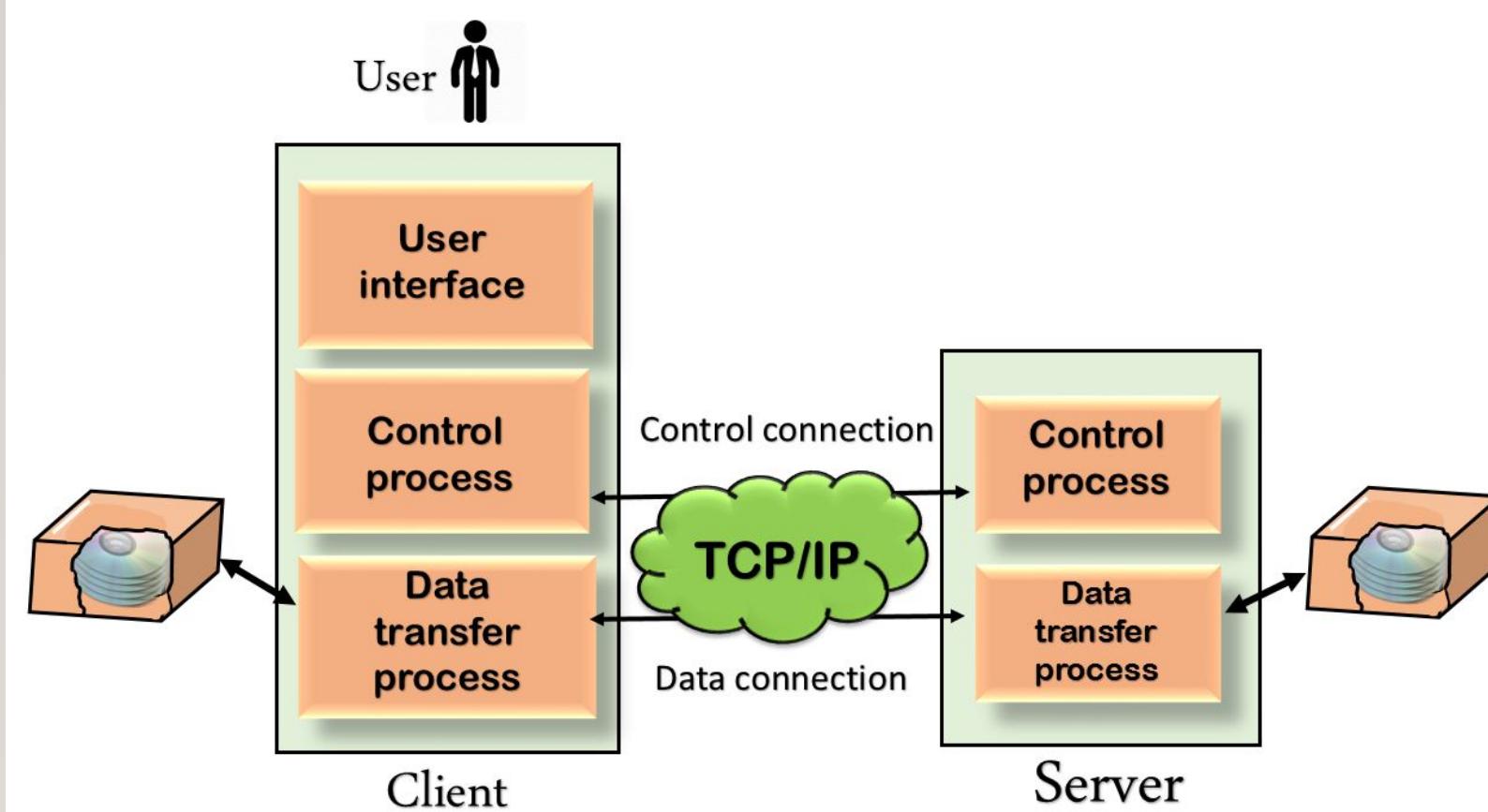
- Transferring files from one computer to another is one of the most common tasks expected from a networking or internetworking environment.
- File Transfer Protocol (FTP) is the standard mechanism provided by *TCP/IP* for copying a file from one host to another.
- Transferring files from one system to another seems simple and straightforward.
- Problems exist
  - Two systems may use different file name conventions.
  - Two systems may have different ways to represent text and data.
  - Two systems may have different directory structures.

# FILE TRANSFER PROTOCOL

---

- FTP differs from other client/server applications in that it establishes two connections between the hosts.
  - One connection is used for data transfer.
  - Other for control information (commands and responses).
- The control connection uses very simple rules of communication.
- The data connection needs more complex rules due to the variety of data types transferred.
- FTP uses two well-known TCP ports: Port 21 is used for the control connection, and port 20 is used for the data connection.

# FILE TRANSFER PROTOCOL



# FILE TRANSFER PROTOCOL

- The client has three components:
  - User interface, Client control process, Client data transfer process.
- The server has two components: server control process and server data transfer process.
- The control connection is made between the control processes.
- The data connection is made between the data transfer processes.
- The control connection remains connected during the entire interactive FTP session.
- The data connection is opened and then closed for each file transferred.
- 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

---

- Communication is achieved through commands and responses.
- Each command or response is only one short line, no need to worry about file format or file structure..
- A file is to be copied from the server to the client. This is called *retrieving*. 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*. 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. FTP treats a list of directory or file names as a file. It is sent over the data connection.

# FILE TRANSFER PROTOCOL

Command	Argument(s)	Description
<b>ABOR</b>		Abort the previous command
<b>CDUP</b>		Change to parent directory
<b>CWD</b>	Directory name	Change to another directory
<b>DELETE</b>	File name	Delete a file
<b>LIST</b>	Directory name	List subdirectories or files
<b>MKD</b>	Directory name	Create a new directory
<b>PASS</b>	User password	Password
<b>PASV</b>		Server chooses a port
<b>PORT</b>	port identifier	Client chooses a port
<b>PWD</b>		Display name of current directory
<b>QUIT</b>		Log out of the system
<b>RETR</b>	File name(s)	Retrieve files; files are transferred from server to client
<b>RMD</b>	Directory name	Delete a directory
<b>RNFR</b>	File name (old)	Identify a file to be renamed
<b>RNTO</b>	File name (new)	Rename the file
<b>STOR</b>	File name(s)	Store files; file(s) are transferred from client to server
<b>STRU</b>	<b>F</b> , <b>R</b> , or <b>P</b>	Define data organization ( <b>F</b> : file, <b>R</b> : record, or <b>P</b> : page)
<b>TYPE</b>	<b>A</b> , <b>E</b> , or <b>I</b>	Default file type ( <b>A</b> : ASCII, <b>E</b> : EBCDIC, <b>I</b> : image)
<b>USER</b>	User ID	User information
<b>MODE</b>	<b>S</b> , <b>B</b> , or <b>C</b>	Define transmission mode ( <b>S</b> : stream, <b>B</b> : block, or <b>C</b> : compressed)

# FILE TRANSFER PROTOCOL

---

- 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.
  - EBCDIC developed by IBM
  - The image file is the default format for transferring binary files.

# FILE TRANSFER PROTOCOL

---

- 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.

**THANK YOU!!!**

---