

# UNIT -I

# PROTOCOLS AND STANDARDS

## Protocol

A protocol is a set of rules that govern data communications. A protocol defines what is communicated, how it is communicated, and when it is communicated. The key elements of a protocol are **syntax, semantics, and timing**

### Syntax

The term *syntax* refers to the **structure or format of the data**, meaning the order in which they are presented

For example, a simple protocol might expect the first 8 bits of data to be the address of the sender, the second 8 bits to be the address of the receiver, and the rest of the stream to be the message itself.

### Semantics

The word *semantics* refers to the **meaning of each section of bits**.

For example, does an address identify the route to be taken or the final destination of the message?

### Timing

The term *timing* refers to two characteristics: **when data should be sent** and **how fast** they can be sent.

For example, if a sender produces data at 100 Mbps but the receiver can process data at only 1 Mbps, the transmission will overload the receiver and some data will be lost.

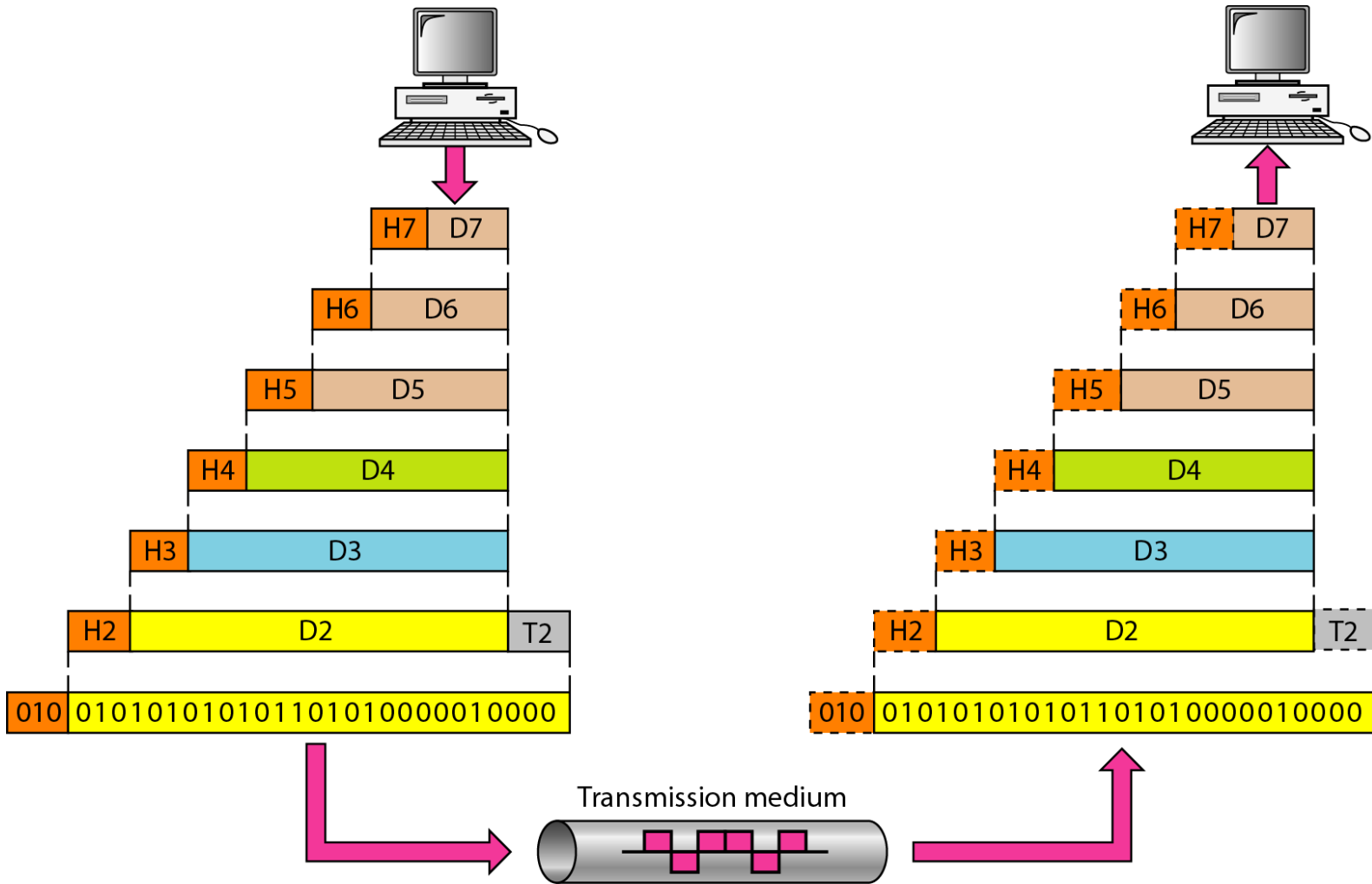
## *Interfaces Between Layers*

The passing of the data and network information down through the layers of the sending device and back up through the layers of the receiving device is made possible by an **interface** between each pair of adjacent layers.

Each interface defines the **information** and **services** a layer must provide for the layer above it. Well-defined interfaces and layer functions provide modularity to a network.

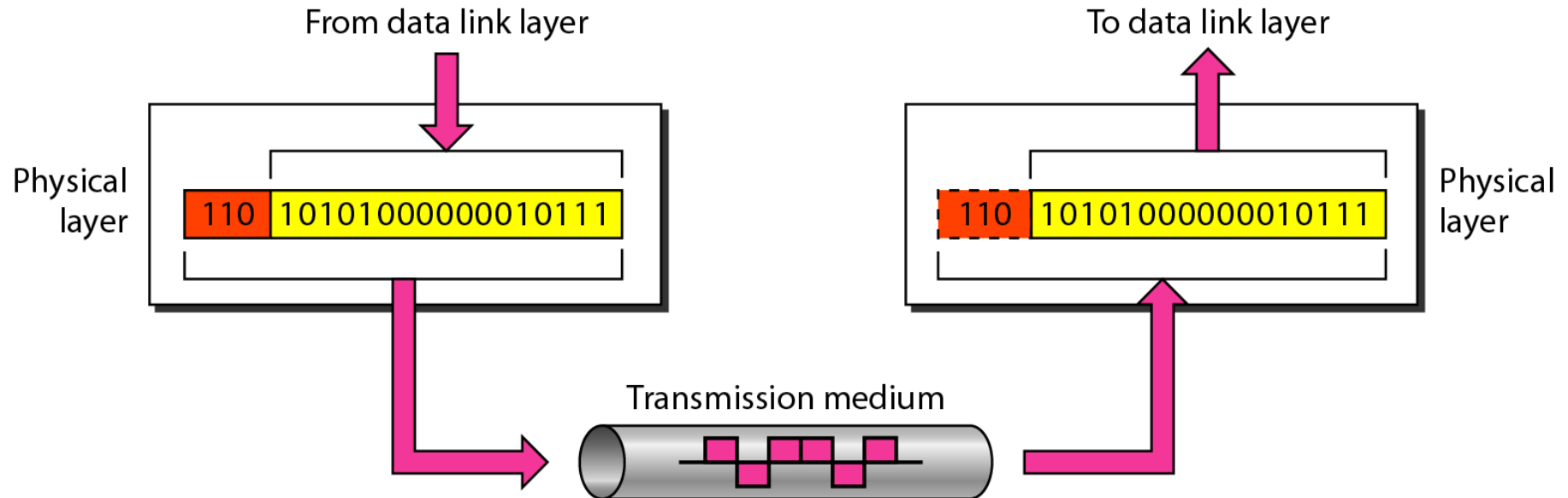
As long as a layer provides the expected services to the layer above it, the specific implementation of its functions can be modified or replaced without requiring changes to the surrounding layers

# An exchange using the OSI model



# LAYERS IN THE OSI MODEL

## 1. Physical layer



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

## *Physical layer*

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

It also defines the procedures and functions that physical devices and interfaces have to perform for transmission to Occur

The physical layer is also concerned with the following:

### **•Physical characteristics of interfaces and medium**

The physical layer defines the characteristics of the **interface** between the devices and the **transmission medium**. It also defines the type of transmission medium.

## •Physical topology

The physical topology defines how devices are connected to make a network.

Devices can be connected by using a **mesh** topology, **star** topology, **bus** topology, **ring** topology or **hybrid** topology

## •Transmission mode

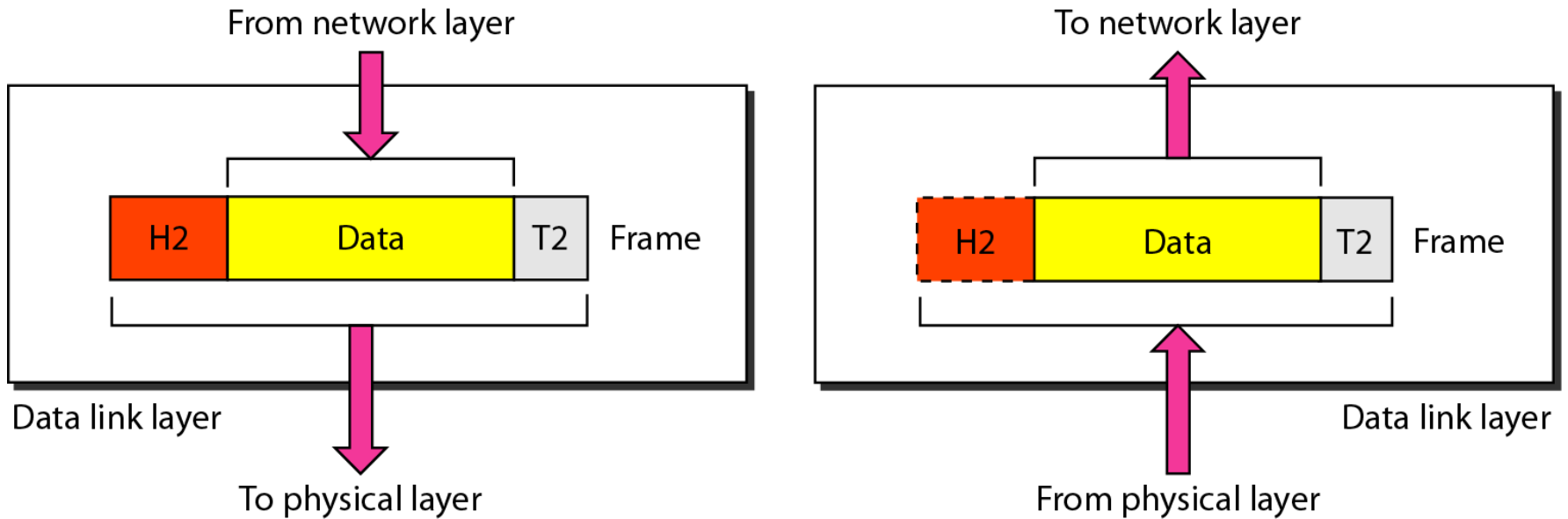
Physical layer defines direction of transmission between to systems

**Simplex mode** : only one device can send and other can only receive. It is one way communication.

**Half duplex mode**: Two devices can send and receive but not at the same time

**Full duplex mode**: Two devices can send and receive at the same time

## 2. Data link layer



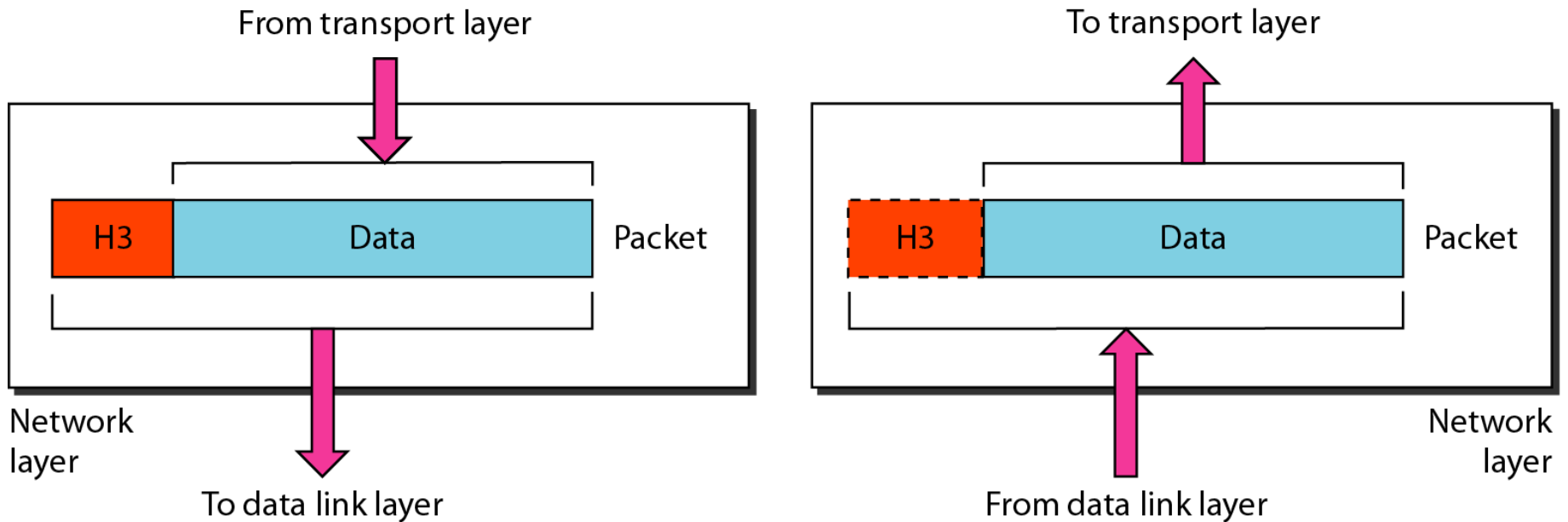
The data link layer is responsible for moving frames from one hop (node) to the next.



## Access control

When **two or more devices** are connected to the same link, data link layer protocols are necessary to determine **which device has control** over the link at any given time.

### 3. Network layer



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

## Network Layer

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

Whereas the data link layer oversees the delivery of the packet between two systems on the same network (links), the network layer ensures that each packet gets from its point of origin to its final destination.

If the two systems are attached to different networks (links) with connecting devices between the networks (links), there is often a need for the network layer to accomplish source-to-destination delivery

Other responsibilities of the network layer include the following:

### **Logical addressing.**

The **physical addressing** implemented by the data link layer handles the addressing problem **locally**.

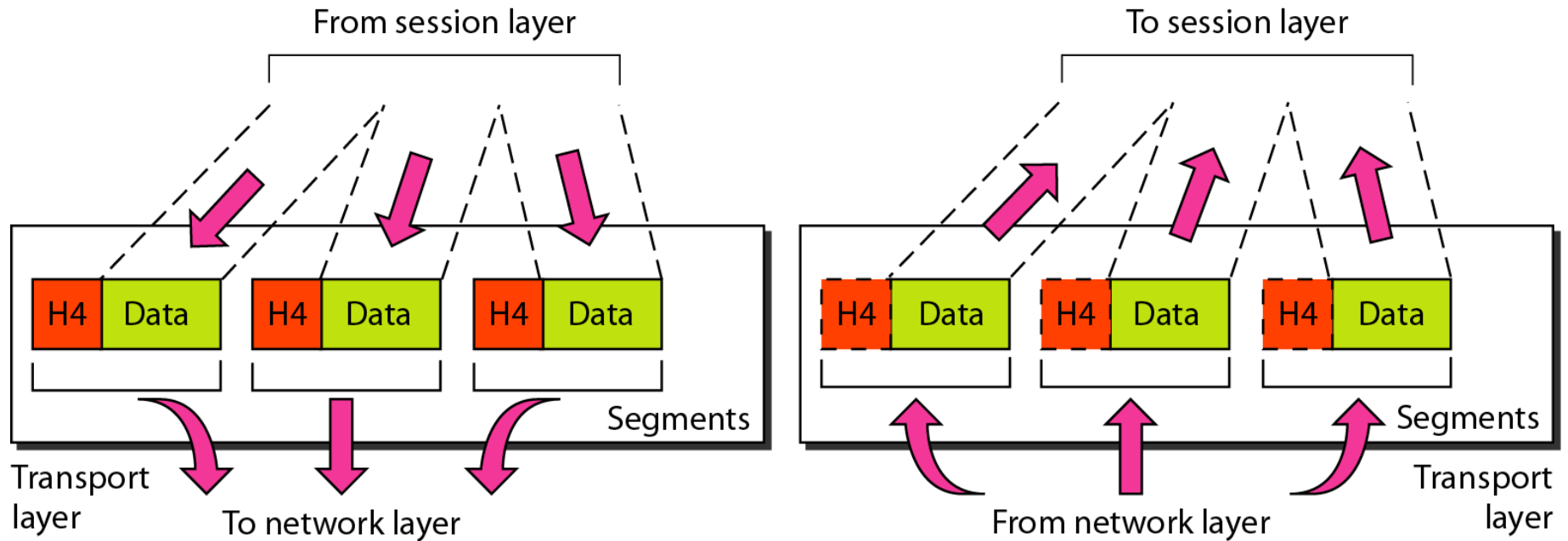
If a packet passes the network boundary, we need another addressing system to help distinguish the source and destination systems.

The network layer **adds a header** to the packet coming from the upper layer that, among other things, includes the logical addresses of the sender and receiver.

### **Routing**

When independent networks or links are connected to create *internetworks*(network of networks) or a large network, the connecting devices (called *routers or switches*) *route or switch the packets to their final destination*. *One of the functions* of the network layer is to provide this mechanism

## 4 .Transport layer



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

## Transport Layer

A process is an application program running on a host

The transport layer, on the other hand, ensures that the whole message arrives intact and in order, overseeing both error control and flow control at the source-to-destination level.

Other responsibilities of the transport layer include the following:

### **Service-point addressing**

Computers often run several programs at the same time. For this reason, source to destination delivery means delivery not only from one computer to the next but also from a **specific process** (running program) **on one computer to a specific process** (running program) **on the other**

The transport layer header must therefore include a type of address called a *service-point address* (or *port address*).

The network layer gets each packet to the correct computer; the transport layer gets the entire message to the correct process on that computer.

### **Segmentation and reassembly**

A message is divided into transmittable **segments**, with each segment containing a **sequence number**

These numbers enable the transport layer to **reassemble** the message correctly upon arriving at the destination and to identify and replace packets that were **lost** in transmission

### **Connection control**

The transport layer can be either connectionless or connection oriented.

A **connectionless** transport layer treats each segment as an independent packet and delivers it to the transport layer at the destination machine.

A **connection oriented** transport layer makes a connection with the transport layer at the destination machine first before delivering the packets. After all the data are transferred ,the connection is terminated.

## Flow control

Like the data link layer, the transport layer is responsible for flow control. However, flow control at this layer is performed **end to end** rather than across a **single link**.

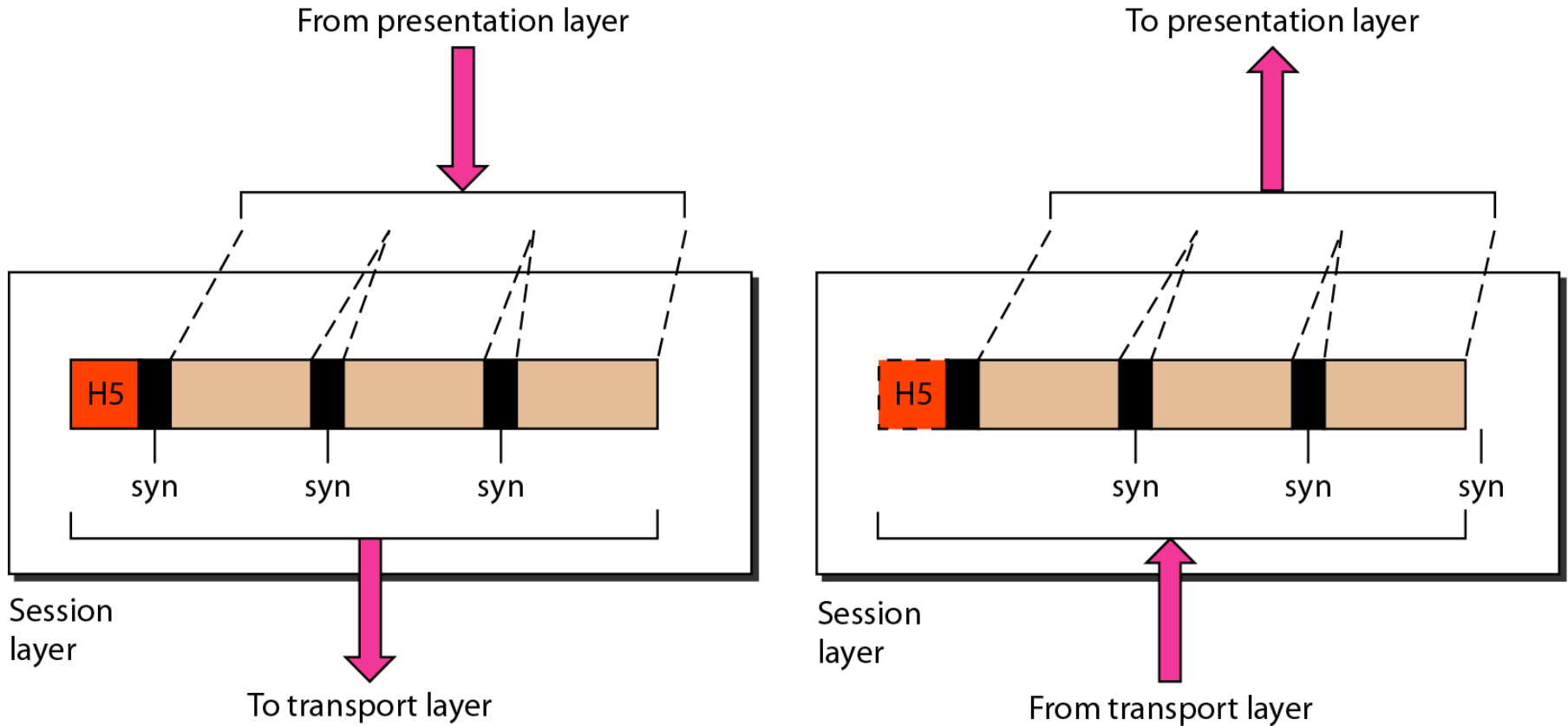
## Error control

Like the data link layer, the transport layer is responsible for error control. However, error control at this layer is performed **process-to process** rather than **across a single link**.

The sending transport layer makes sure that the entire message arrives at the receiving transport layer without error (damage, loss, or duplication). Error correction is usually achieved through retransmission.

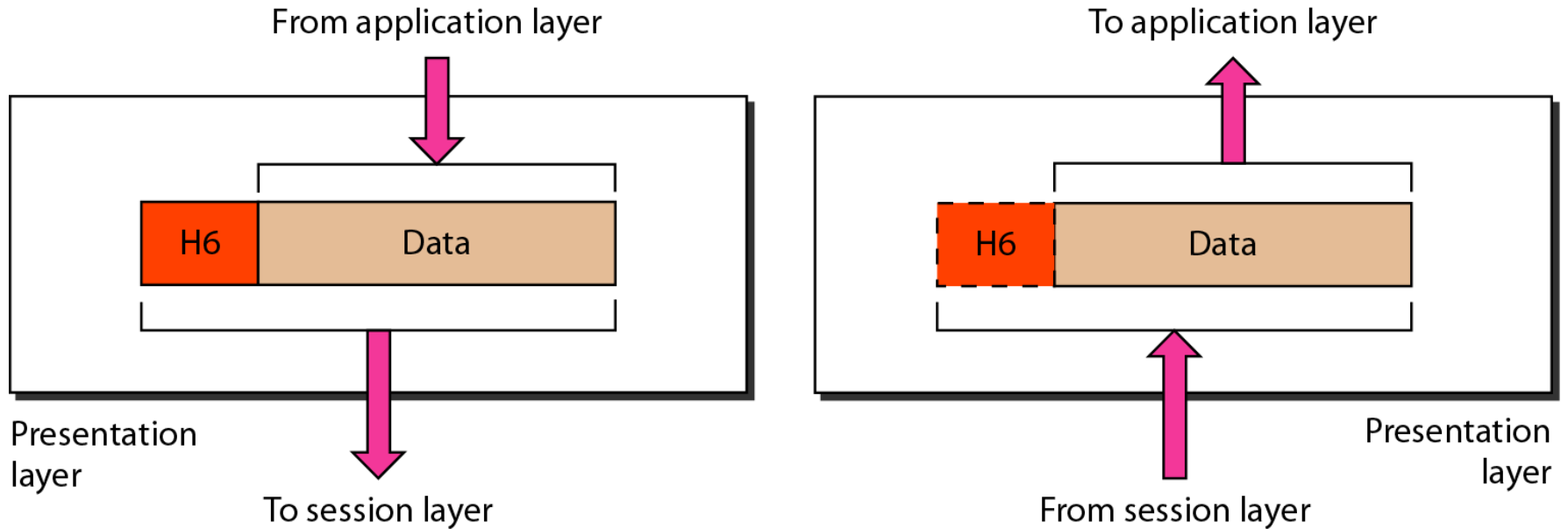


## 5. Session layer



The session layer is responsible for dialog control and synchronization.

## 6. Presentation layer



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

## Presentation Layer

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

Specific responsibilities of the presentation layer include the following:

### **Translation**

The processes (running programs) in two systems are usually exchanging **information** in the form of **character strings, numbers, and so on**. The information must be changed to bit streams before being transmitted.

Because different computers use different encoding systems, the presentation layer is responsible for **interoperability** between these different encoding methods.

The presentation layer at the sender changes the information from its **sender-dependent format** into a **common format**.

The presentation layer at the receiving machine changes the **common format** into its **receiver-dependent format**

### **Encryption**

To carry sensitive information, a system must be able to ensure **privacy**.

Encryption means that the sender transforms the original information to another form and sends the resulting message out over the network

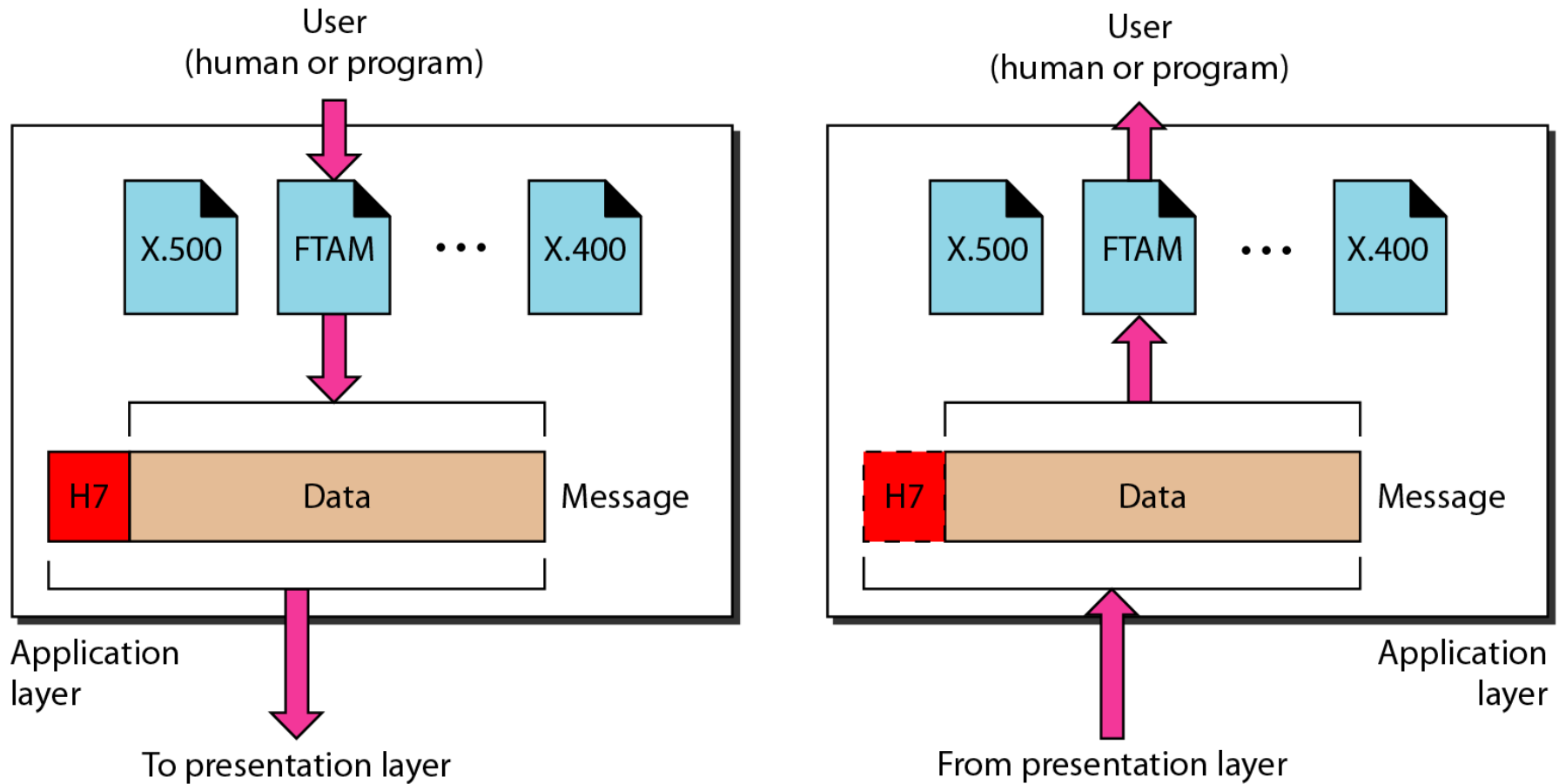
Decryption reverses the original process to transform the message back to its original form

### **Compression.**

Data compression reduces the **number of bits** contained in the information.

Data compression becomes particularly important in the transmission of multimedia such as **text, audio, and video**

## 7. Application layer



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

## Application Layer

The application layer enables the user, whether human or software, to access the network.

It provides user interfaces and support for services such as electronic mail, remote file access and transfer, shared database management, and other types of distributed information services.

Figure 2.14 shows the relationship of the application layer to the user and the presentation layer.

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

Specific services provided by the application layer include the following

### **Network virtual terminal**

A network virtual terminal is a software version of a physical terminal, and it allows a user to **log on to a remote host**.

To do so, the application creates a software emulation of a terminal at the remote host. The user's computer talks to the software terminal which, in turn, talks to the host, and vice versa.

The remote host believes it is communicating with one of its own terminals and allows the user to log on.

### **File transfer, access, and management**

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

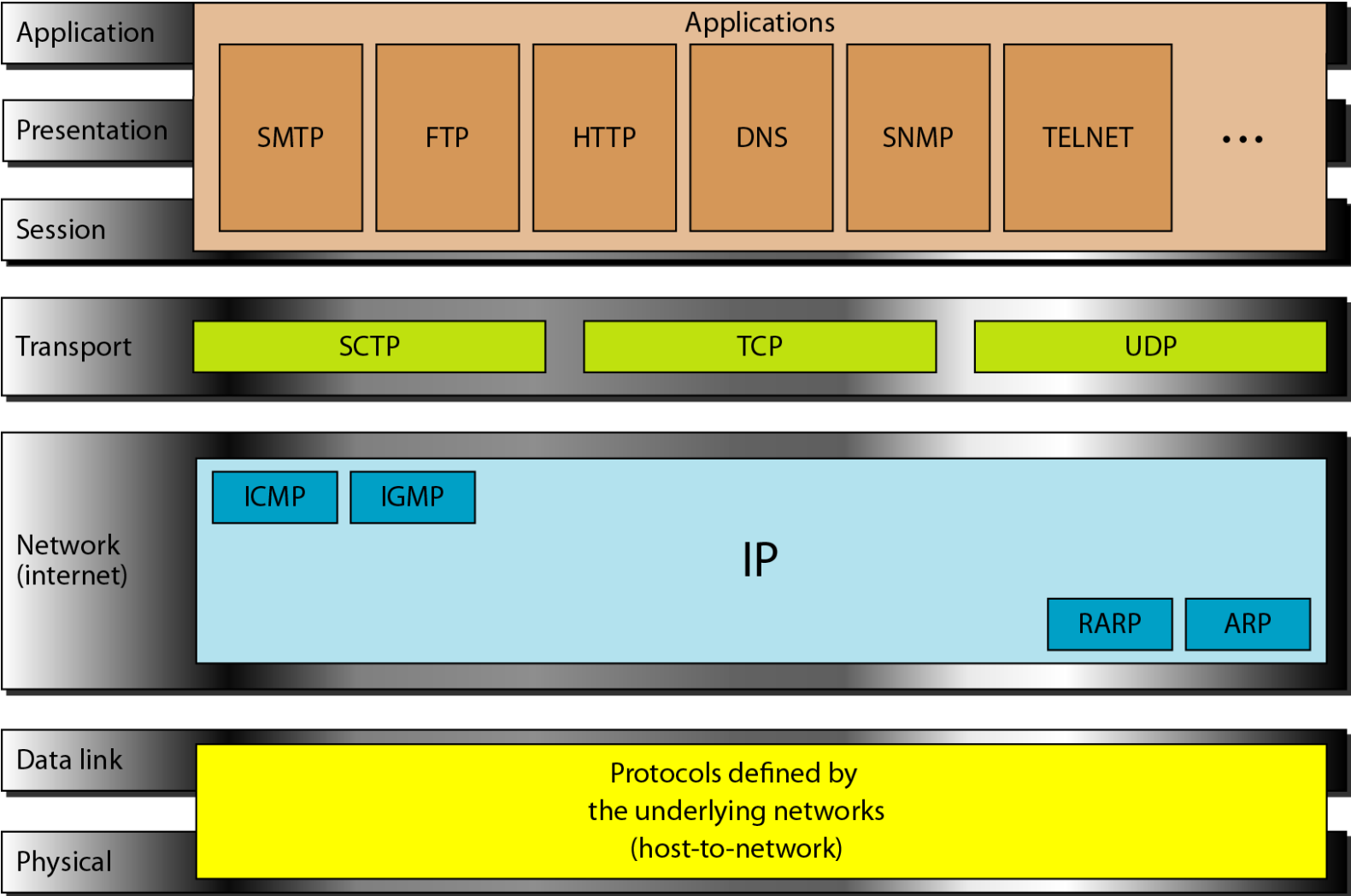
### **Mail services**

This application provides the basis for e-mail forwarding and storage.

### **Directory services**

This application provides distributed database sources and access for global information about various objects and services.

# TCP/IP model





# TCP/IP PROTOCOL SUITE

The original TCP/IP protocol suite was defined as having four layers:

- 1 Host-to-Network Layer
- 2 Internet Layer
- 3 Transport Layer
- 4 Application Layer

TCPIIP protocol suite is made of five layers:

physical, data link, network, transport, and application.

The first four layers provide physical standards, network interfaces, internetworking, and transport functions that correspond to the first four layers of the OSI model.

## Transport Layer

Traditionally the transport layer was represented in *TCP/IP* by *two protocols: TCP and UDP*. IP is a host-to-host protocol, meaning that it can deliver a packet from one physical device to another.

UDP and TCP are transport level protocols responsible for delivery of a message from a **process** (running program) **to another process**.

A new transport layer protocol, SCTP, has been devised to meet the needs of some newer applications.

### *User Datagram Protocol(UDP)*

The User Datagram Protocol (UDP) is the simpler of the two standard TCP/IP transport protocols.

It is a process-to-process protocol that adds only **port addresses, checksum error control, and length information** to the data from the upper layer

### *Transmission Control Protocol(TCP)*

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

At the sending end of each transmission, TCP divides a stream of data into smaller units called *segments*. *Each segment includes a sequence number for reordering after receipt*, together with an acknowledgment number for the segments received.

Segments are carried across the internet inside of IP datagrams. At the receiving end, TCP collects each datagram as it comes in and reorders the transmission based on sequence numbers.

# THE INTERNET

## A Brief History

- A network is a group of connected communicating devices such as computers and printers.
- In the **mid-1960s**, mainframe computers in research organizations were standalone devices.
- The **Advanced Research Projects Agency (ARPA)** in the **Department of Defense (DoD)** was interested in finding a way to connect computers so that the researchers they funded could share their findings, thereby reducing costs and eliminating duplication of effort
- In 1967**, at an **Association for Computing Machinery (ACM)** meeting, **ARPA** presented its ideas for **ARPANET**, a small network of connected computers.
- In 1972**, Vint Cerf and Bob Kahn, both of whom were part of the core ARPANET group, collaborated on what they called the *Internetting Project*. *Cerf and Kahn's landmark 1973* paper outlined the protocols to achieve **end-to-end delivery of packets**. This paper on **Transmission Control Protocol (TCP)** included concepts such as encapsulation, the datagram, and the functions of a gateway.

•Shortly thereafter, authorities made a decision to **split TCP into two protocols: Transmission Control Protocol (TCP) and Internetworking Protocol (IP)**.. The internetworking protocol became known as TCP/IP

## **The Internet Today**

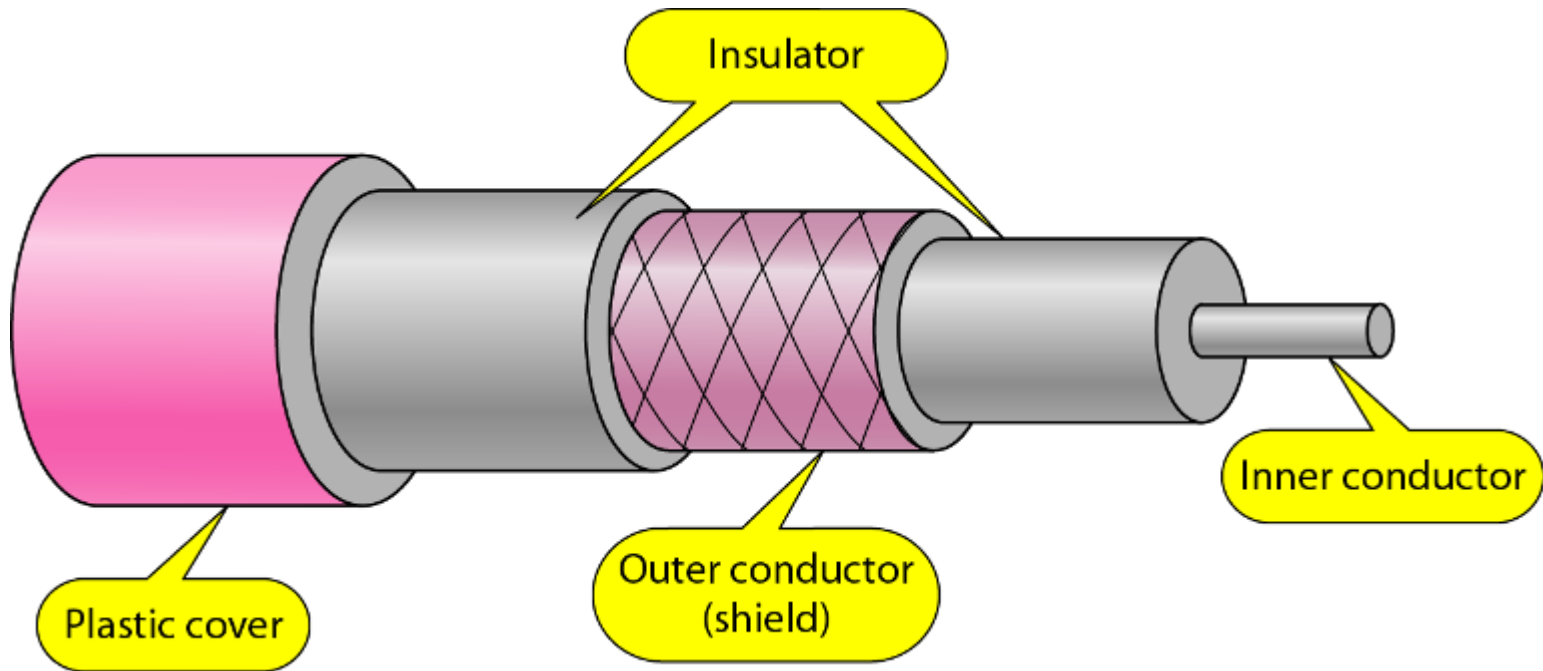
The Internet has come a long way since the 1960s. The Internet today is not a simple hierarchical structure. It is made up of many wide- and local-area networks joined by connecting devices and switching stations.

Today most end users who want Internet connection use the services of **Internet service providers (ISPs)**.

There **are**

- international service providers,**
- national service providers,**
- regional service providers**
- local service providers.**

# Coaxial Cable



# Coaxial Cable

- **coax has a central core conductor of solid or stranded wire (usually copper) enclosed in an insulating sheath,**
- **which is, in turn, encased in an outer conductor of metal foil, braid, or a combination of the two**
- **The outer metallic wrapping serves both as a shield against noise and as the second conductor, which completes the circuit.**
- **This outer conductor is also enclosed in an insulating sheath, and the whole cable is protected by a plastic cover**

# Categories of Coaxial Cables

<i>Category</i>	<i>Impedance</i>	<i>Use</i>
RG-59	75 $\Omega$	Cable TV
RG-58	50 $\Omega$	Thin Ethernet
RG-11	50 $\Omega$	Thick Ethernet

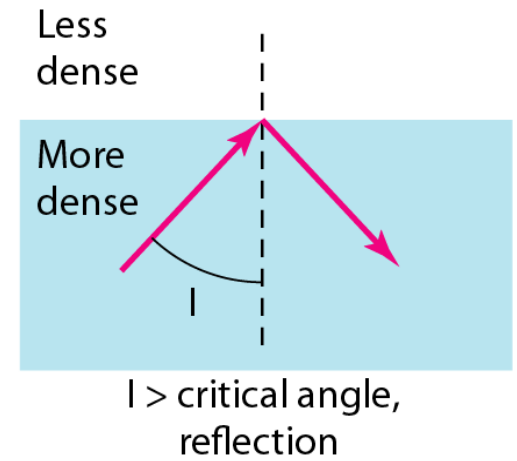
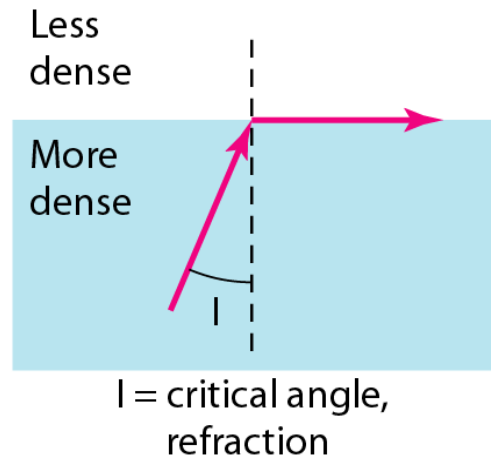
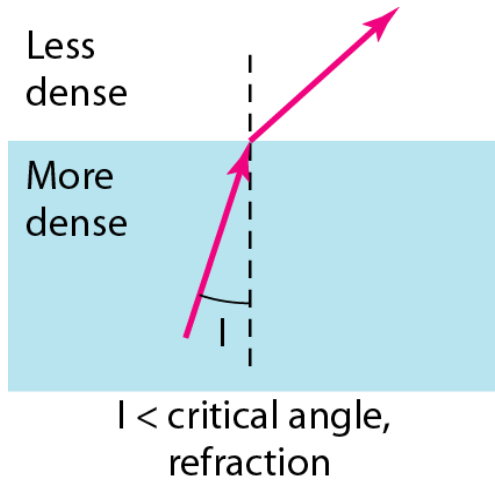
**RG – Radio Government**

# *Applications*

- Coaxial cable was widely used in **analog telephone networks**
- In the **traditional cable TV network**, the entire network used coaxial cable
- Another common application of coaxial cable is in traditional **Ethernet LANs**



## Fiber optics: *Bending of light ray*



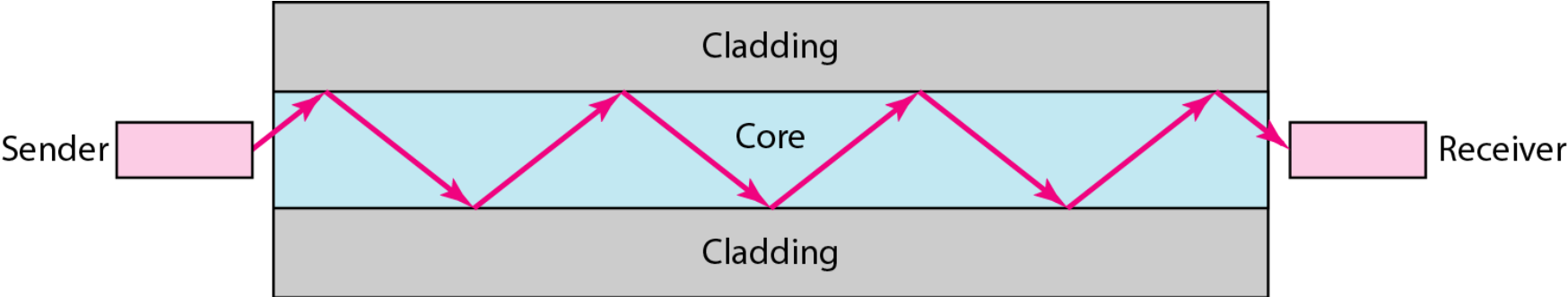
# Fiber-Optic Cable

- A fiber-optic cable is made of **glass or plastic** and transmits signals in the form of **light**
- Light travels in a straight line as long as it is moving through a single uniform substance
- if the angle of incidence  $I$  is **less than** the critical angle, the ray **refracts** and moves **closer to the surface**
- If the angle of incidence is **equal to** the critical angle, the light **bends along the interface**
- If the angle is **greater than the critical angle**, the ray reflects and travels again in the **denser substance**.

---

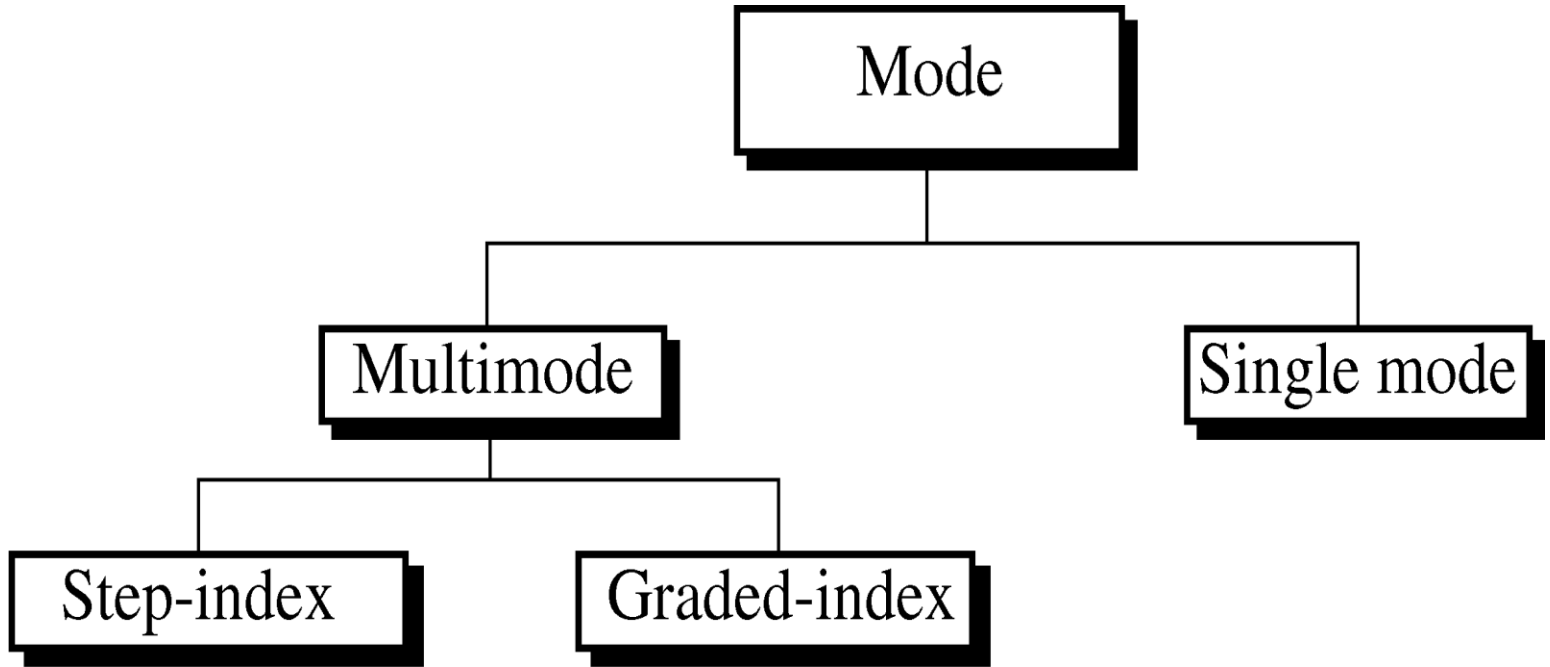
*Optical fiber*

---

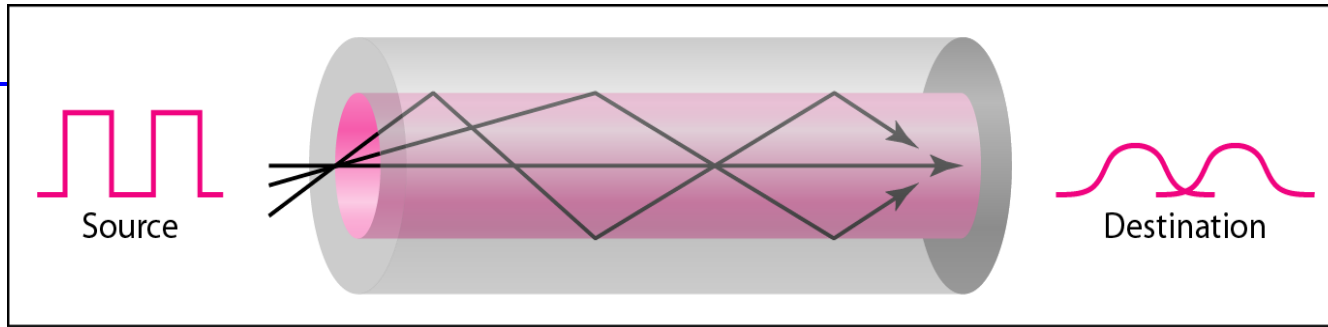


# *Optical fiber*

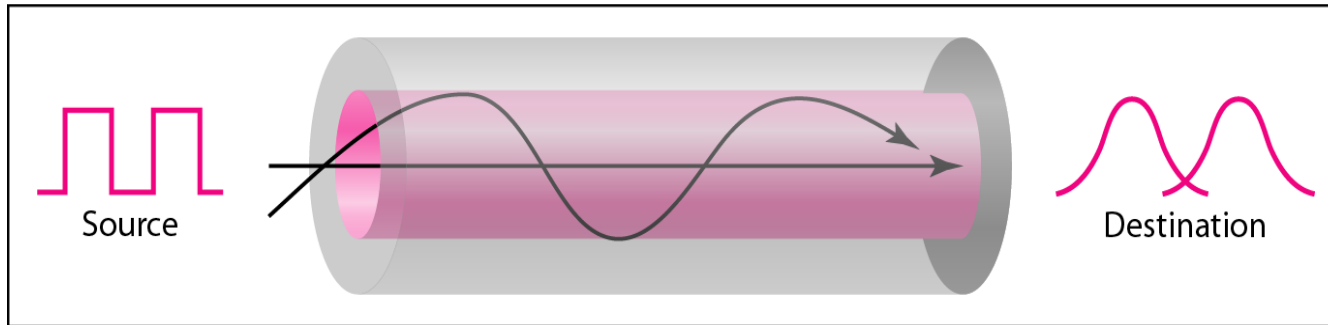
- Optical fibers use reflection to guide light through a channel
- . A glass or plastic core is surrounded by a cladding of less dense glass or plastic.
- The difference in density of the two materials must be such that
- a beam of light moving through the **core is reflected off the cladding instead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



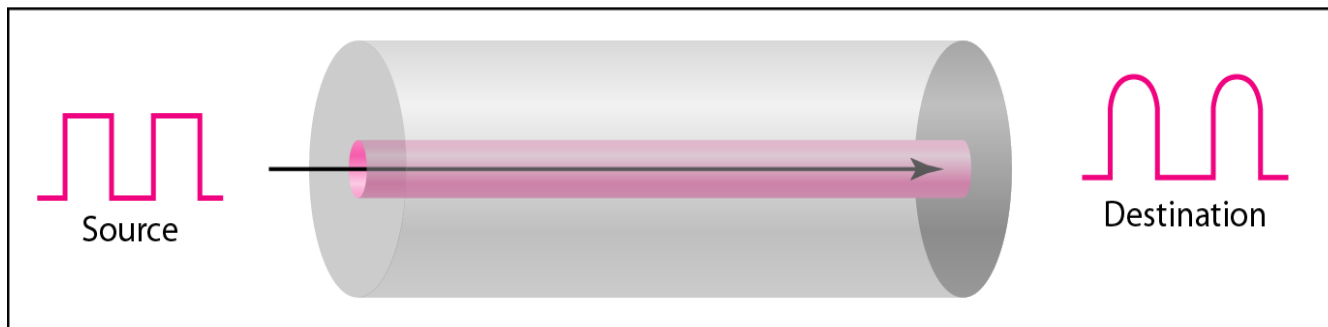
# Modes



a. Multimode, step index



b. Multimode, graded index



c. Single mode

# Multimode

- Multimode is so named because multiple beams from a light source move through the core in different paths
- In **multimode step-index fiber**, the density of the core remains constant from the center to the edges.
- A beam of light moves through this **constant density** in a straight line until it reaches the interface of the **core and the cladding**
- *step index refers to the suddenness of this change, which contributes to the distortion* of the signal as it passes through the **fiber**.

# Multimode

- **multimode graded-index fiber, decreases this distortion of the signal through the cable**
- A graded-index fiber, therefore, is one with **varying densities. Density is highest at the center of the core and decreases gradually to its lowest at the edge.**



# Single-Mode

- **Single-mode uses step-index fiber and a highly focused source of light**
- that limits beams to a small range of angles, all close to the horizontal
- propagation of **different beams is almost identical, and delays are negligible.**
- All the beams arrive at the destination "together" and can be recombined with little distortion to the signal

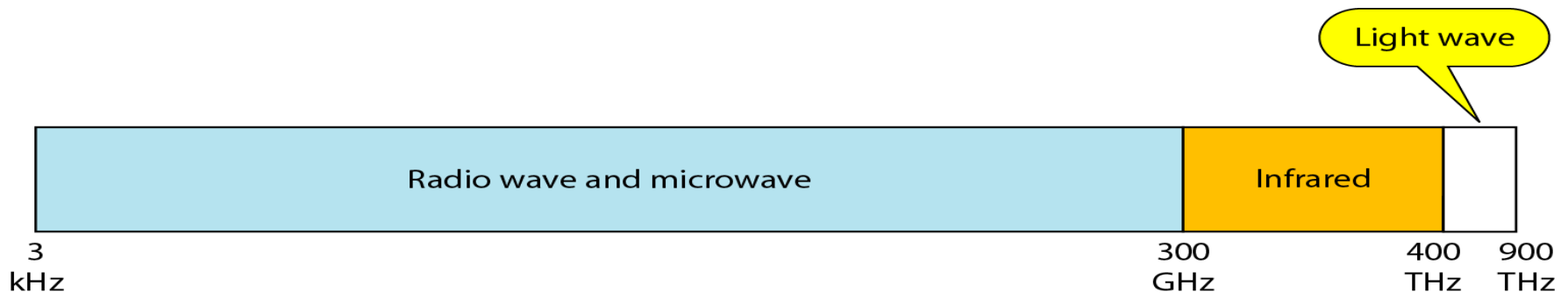
# *Applications*

- Fiber-optic cable is often found in **backbone networks**
- cable TV companies use a combination of **optical fiber and coaxial cable**, thus creating a hybrid network
- Local-area networks such as 100Base-FX network (Fast Ethernet) and 1000Base-X also use fiber-optic cable

# UNGUIDED MEDIA: WIRELESS

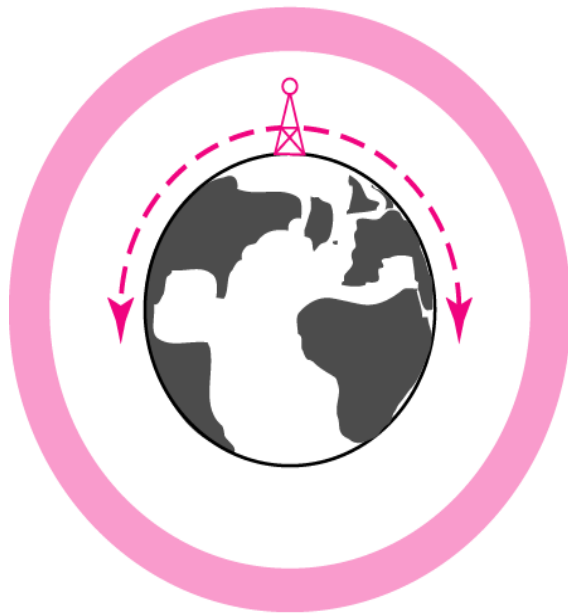
- Unguided media transport electromagnetic waves without using a physical conductor
- This type of communication is often referred to as wireless communication
- Signals are normally broadcast through free space

## Electromagnetic Spectrum



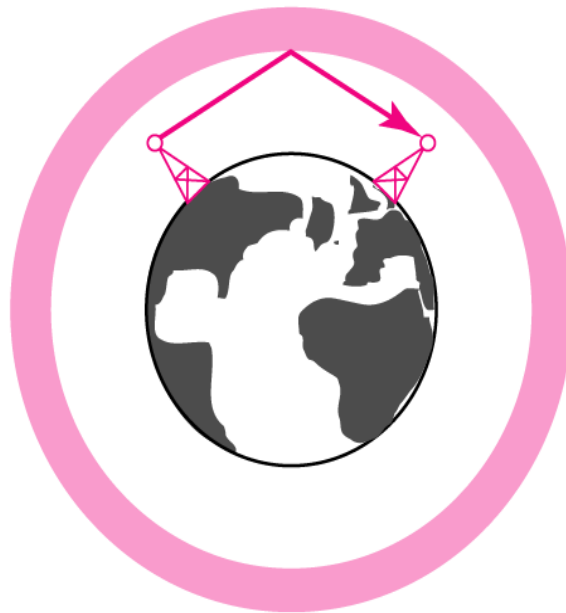
# Propagation Methods

Ionosphere



Ground propagation  
(below 2 MHz)

Ionosphere



Sky propagation  
(2–30 MHz)

Ionosphere



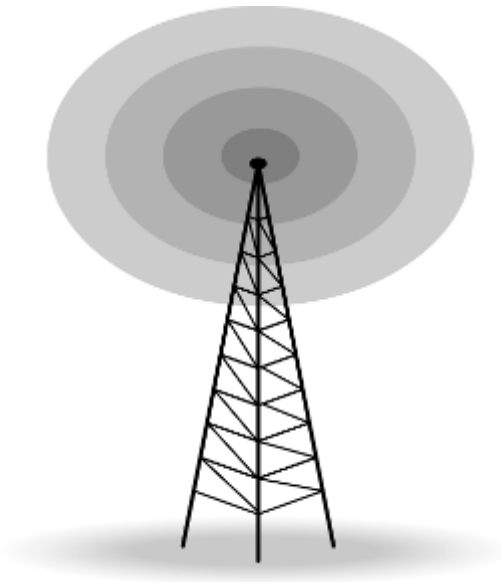
Line-of-sight propagation  
(above 30 MHz)

Unguided signals can travel from the source to destination in several ways: ground propagation, sky propagation, and line-of-sight propagation

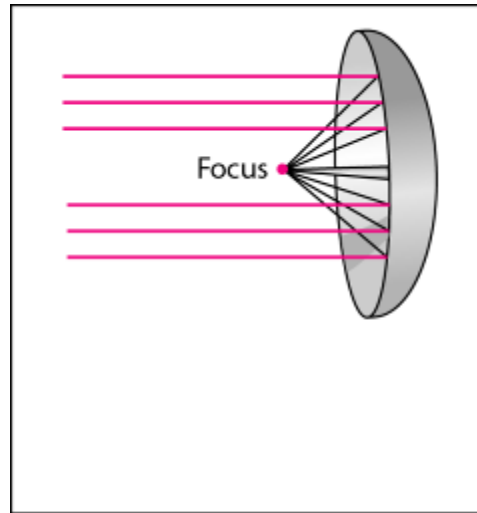
# Propagation Methods

- In **ground propagation**, radio waves travel through the **lowest portion of the atmosphere**, hugging the earth
- In **sky propagation**, higher-frequency radio waves radiate upward into the **ionosphere** where they are reflected back to earth.
- In **line-of-sight propagation**, very high-frequency signals are transmitted in straight lines directly from antenna to antenna. Antennas must be directional, facing each other,

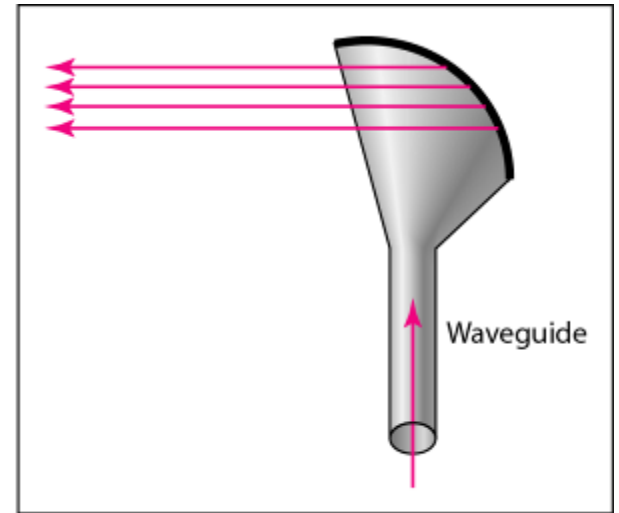
# Antennas



Omnidirectional Antenna



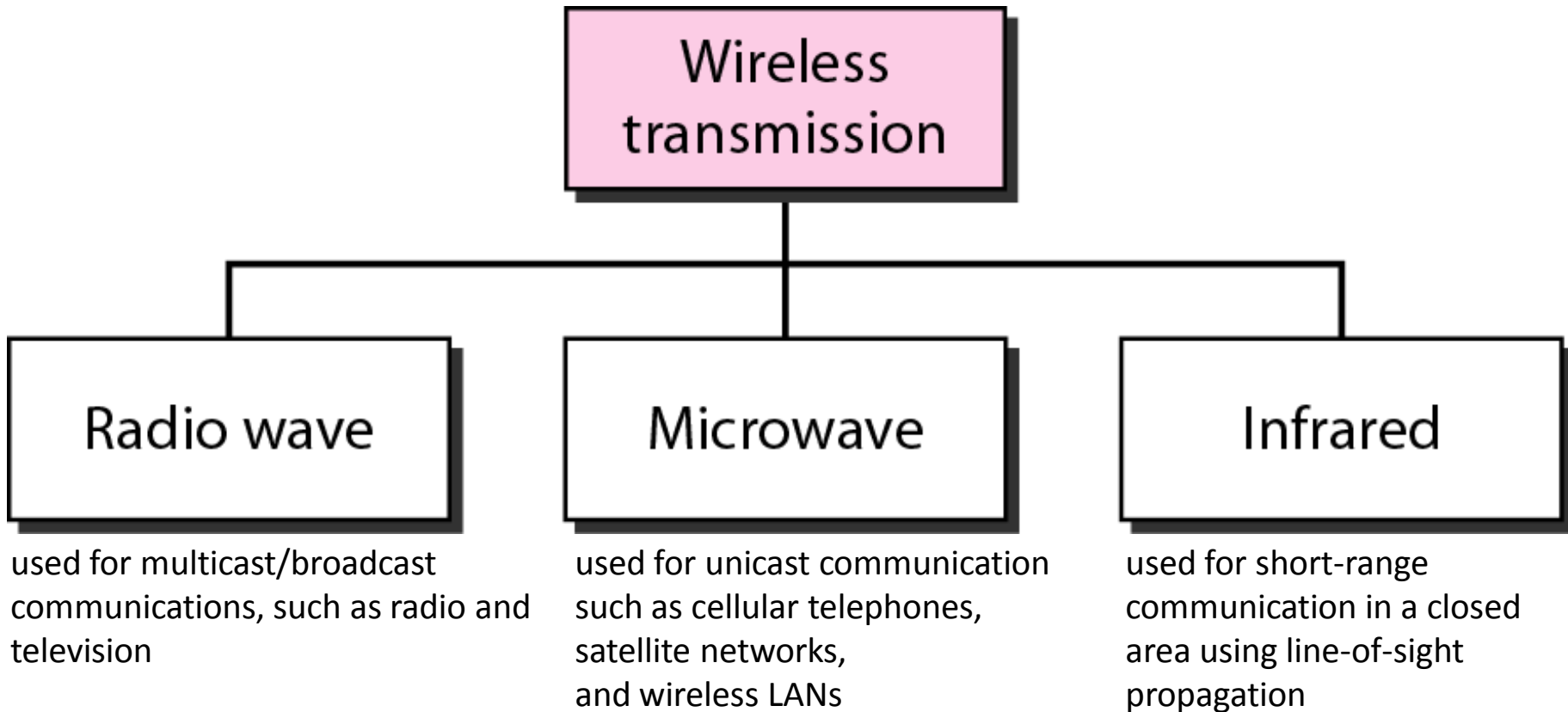
a. Dish antenna



b. Horn antenna

Unidirectional Antennas

# Wireless Transmission Waves



# Radio Waves

- Electromagnetic waves ranging in frequencies between 3 kHz and 1 GHz are called radio waves.
- Radio waves, for the most part, are omni directional.
- When an antenna transmits radio waves, they are propagated in all directions
- The radio waves transmitted by one antenna are susceptible to interference by another antenna that may send signals using the same frequency
- Radio waves, particularly those of low and medium frequencies, **can penetrate walls.**



## *Applications*

- AM and FM radio, television, maritime radio, cordless phones, and paging are examples of multicasting
- Radio waves are used for multicast communications, such as radio and television, and paging systems.

# Microwaves

- Electromagnetic waves having frequencies between 1 and 300 GHz are called microwaves
- Microwaves are unidirectional.
- Sending and receiving antennas need to be aligned
- Microwave propagation is line-of-sight.
- Very high-frequency microwaves cannot penetrate walls.

# *Applications*

- Microwaves, due to their unidirectional properties, are very useful
- when unicast (one-to-one) communication is needed
- Microwaves are used for unicast communication such as cellular telephones
- satellite networks, and wireless LANs.

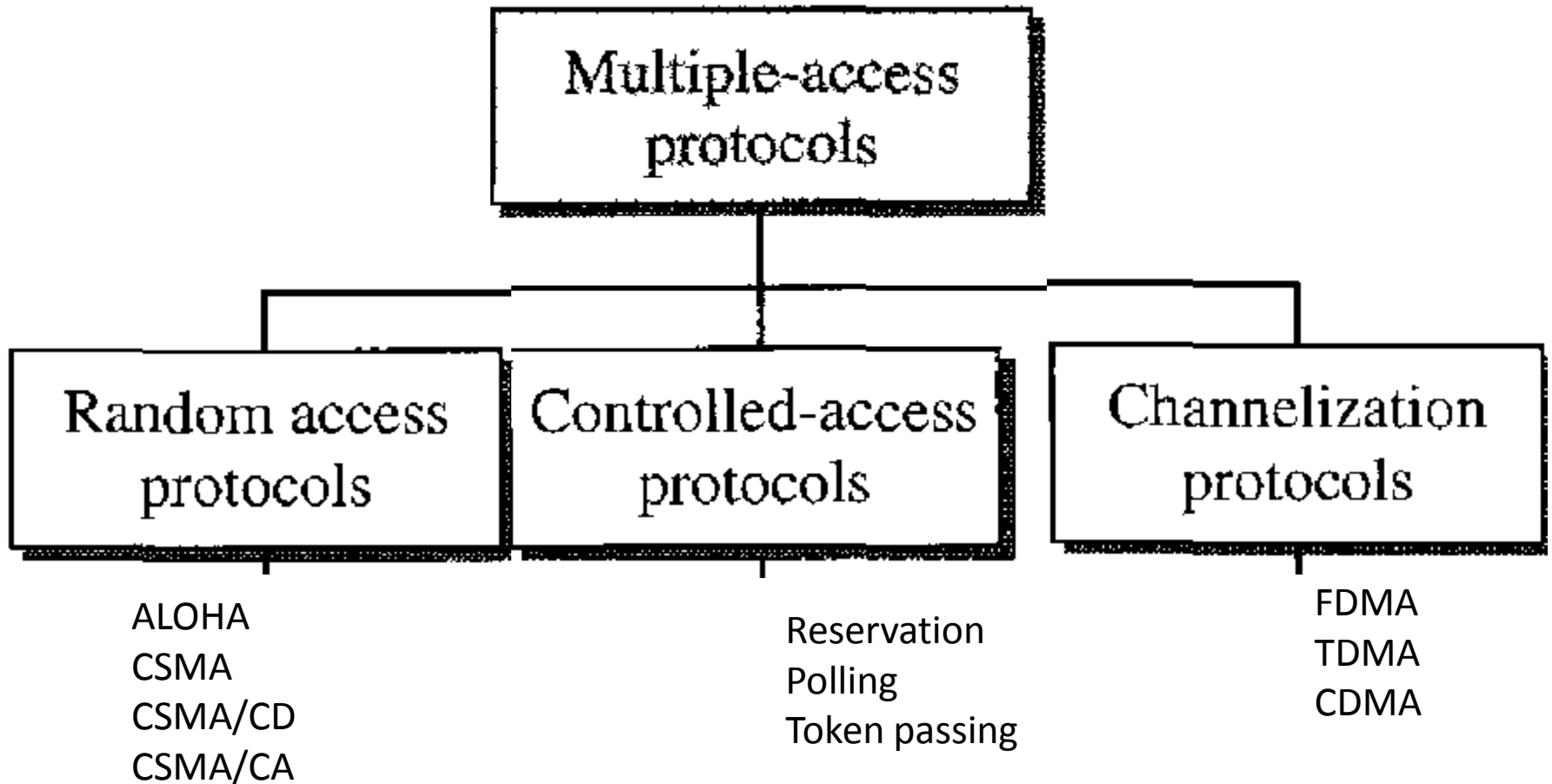
# Infrared

- Infrared waves, with frequencies from 300 GHz to 400 THz can be used for short-range communication
- Infrared waves, having high frequencies, cannot penetrate walls
- *Applications*
- Infrared signals can be used for short-range communication in a closed area using line-of-sight propagation.
- A wireless keyboard to communicate with a PC

# UNIT -II

*Multiple  
Access*

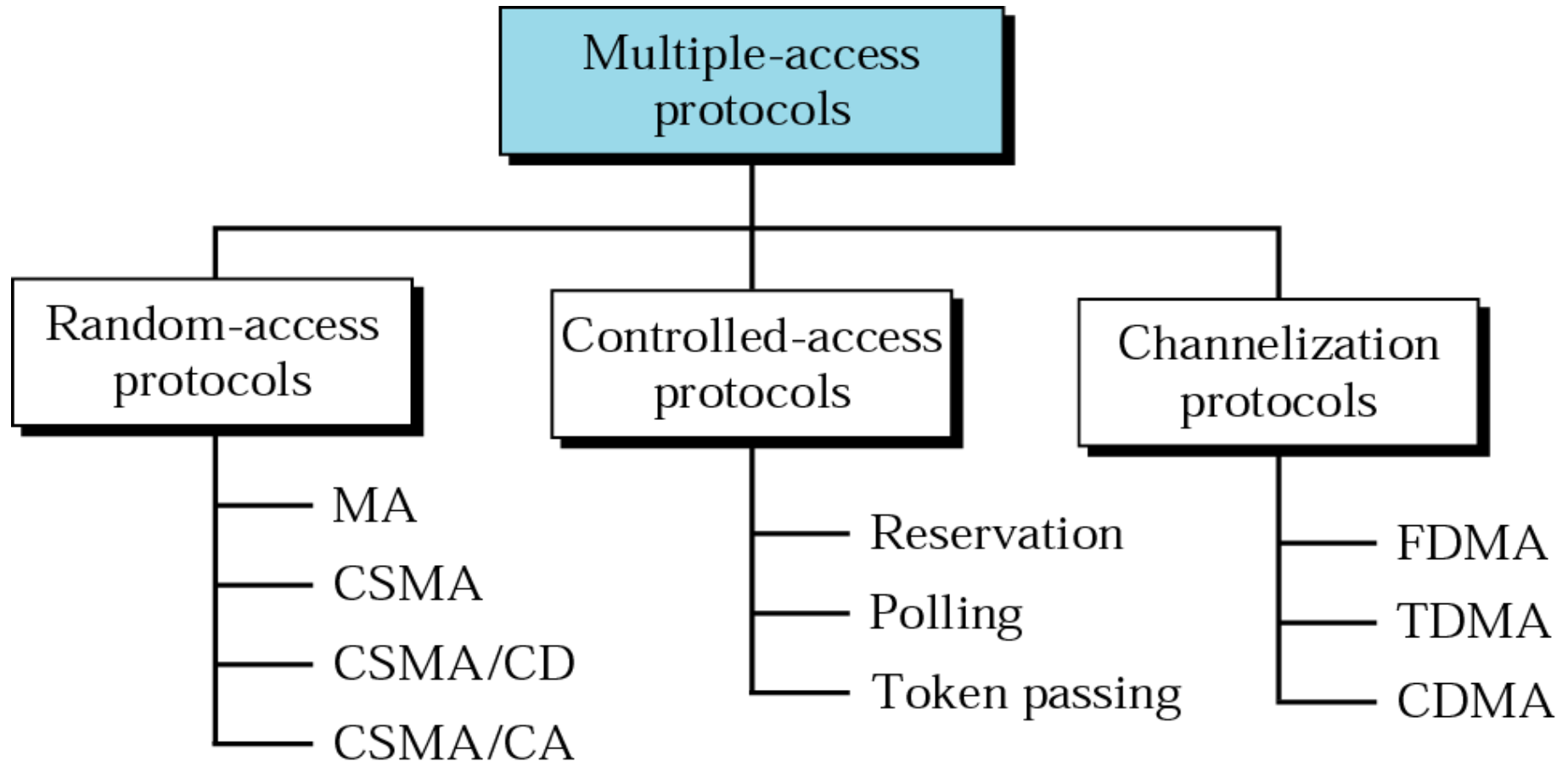
*Taxonomy of multiple-access protocols discussed  
in this chapter*



# MULTIPLE ACCESS

- When nodes or stations are connected and use a common link, called a multipoint or broadcast link, we need a multiple-access protocol to coordinate access to the link.
- The problem of controlling the access to the medium is similar to the rules of speaking in an assembly.
- The procedures guarantee that the right to speak is upheld and ensure that :
  - Two people do not speak at the same time
  - do not interrupt each other
  - do not monopolize the discussion, and so on.





# RANDOM ACCESS

- In random access or contention methods, no station is superior to another station and none is assigned the control over another.
- No station permits, or does not permit, another station to send.
- At each instance, a station that has data to send uses a procedure defined by the protocol to make a decision on whether or not to send.
- This decision depends on the state of the medium (idle or busy)

# Two features give this method its name

- First, there is no scheduled time for a station to transmit.
- Transmission is random among the stations. That is why these methods are called *random access*.
- Second, no rules specify which station should send next.
- Stations compete with one another to access the medium.
- That is why these methods are also called *contention methods*.

# RANDOM ACCESS

- In a random access method, each station has the right to the medium without being controlled by any other station.
- However, if more than one station tries to send, there is an access conflict-collision-and the frames will be either destroyed or modified.

# RANDOM ACCESS

- To avoid access conflict or to resolve it when it happens, each station follows a procedure that answers the following questions:
  - When can the station access the medium?
  - What can the station do if the medium is busy?
  - How can the station determine the success or failure of the transmission?
  - What can the station do if there is an access conflict?

# Random Access

*MA*

The random-access methods  
have evolved as

*CSMA*

*CSMA/CD*

*CSMA/CA*

# Random Access

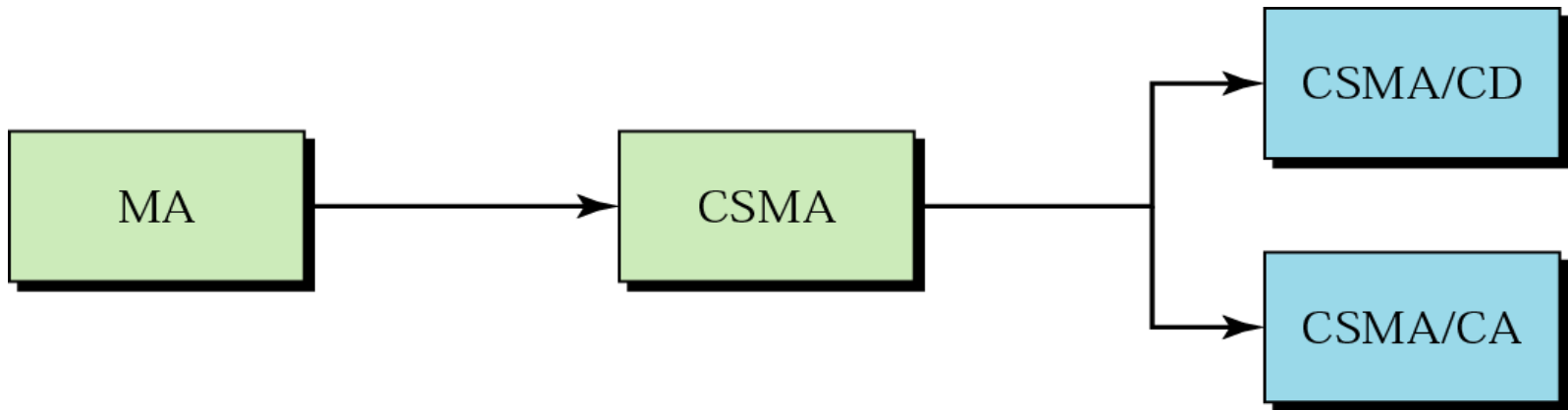
- The random access methods have evolved from a very interesting protocol known as **ALOHA**, which used a very simple procedure called **multiple access (MA)**.
- The method was improved with the addition of a procedure that forces the station to sense the medium before transmitting.
- This was called **carrier sense multiple access(CSMA)**.

# Random Access

- This method later evolved into two parallel methods:
  - carrier sense multiple access with collision detection (CSMA/CD)
  - carrier sense multiple access with collision avoidance (CSMA/CA).
- *CSMA/CD tells the station what to do when a collision is detected.*
- *CSMA/CA tries to avoid the collision.*



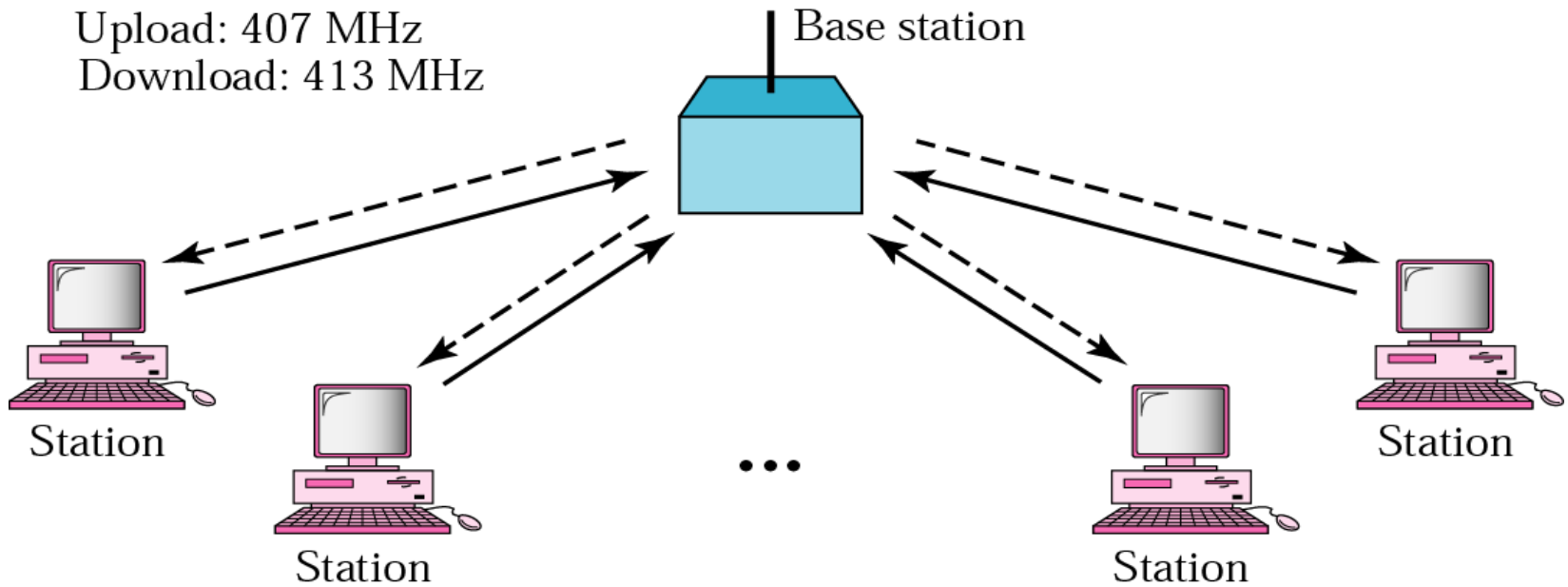
# Evolution of random-access methods



# Pure aloha:

- ALOHA was developed in 1970. it was designed for a radio LAN, but it cannot be used on any shared medium.
- Pure aloha: original aloha
- Idea is that each station sends frame whenever it has frame to send.
- Only one channel to share there is possibility of collision between frames from different stations.
- Pure aloha relies on acknowledgment from receiver. if acknowledgment does not arrive after timeout period station resends the frame.
- If all stations try to resend their frames after timeout, frames collide again. pure aloha dictates that when timeout period passes, each station waits for random amount of time before resending the **frame. this time is called back off time**.

# ALOHA network



# Procedure for ALOHA protocol

$K$ : Number of attempts

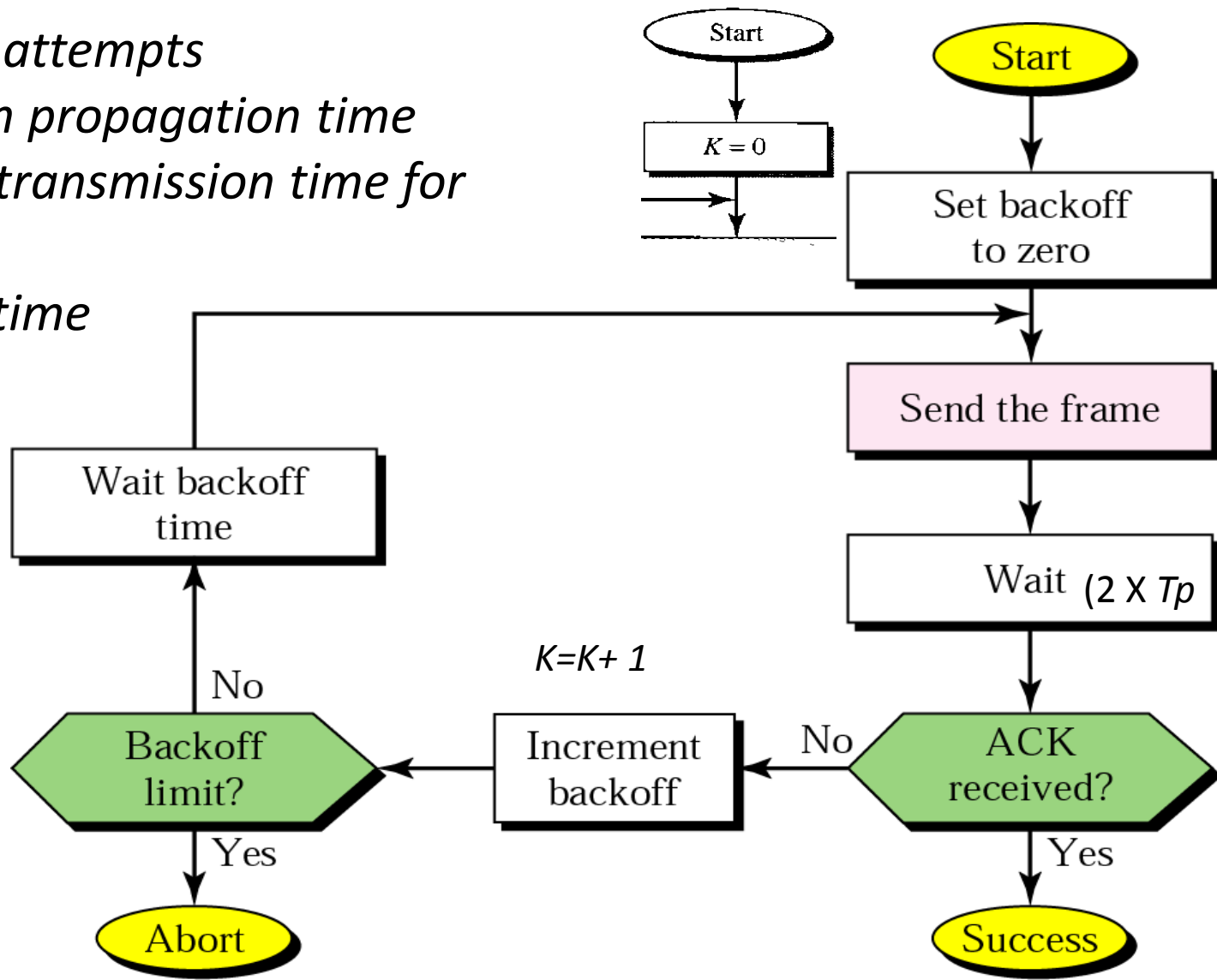
$T_p$ : Maximum propagation time

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

$TB$ : Back-off time

$(T_a = R \times T_p \text{ or } R \times T_{fr})$

$K_{max}$ , is normally 15

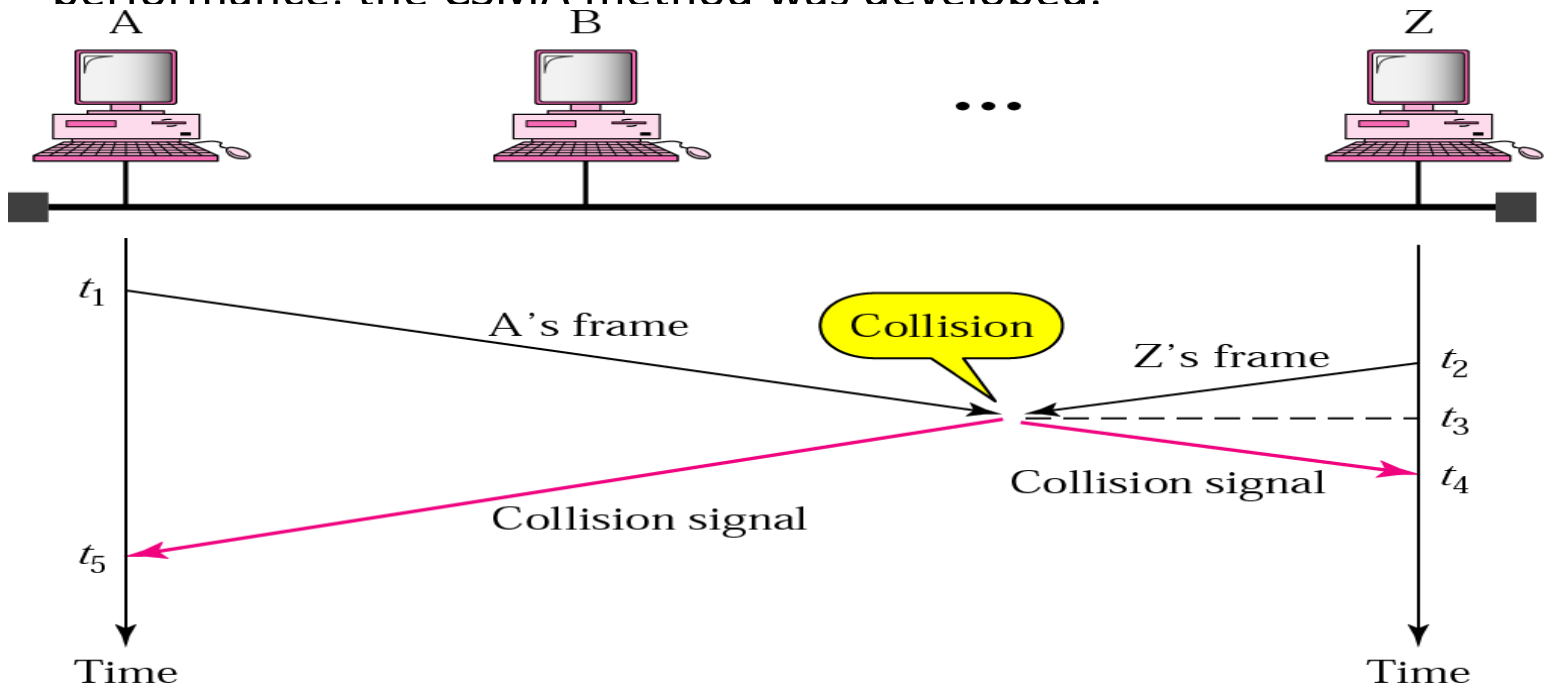


# Slotted aloha

- Slotted aloha : was invented to improve the efficiency of pure aloha.
- here we divide the time into slots  $t_{fr}$  s and force the station to send only at the beginning of time slot.
- Here if station misses to send at beginning of time slot it has to wait till next time slot.
- But 2 or more stations try to send at same time slot, collision occurs.

# CSMA(carrier sense multiple access)

- In CSMA every station must first listen to the medium before sending.
- **principle:** sense before transmit or listen before talk.
- It can reduce the possibility of collision but cannot avoid it because of propagation delay.
- To minimize the chance of collision and, therefore, increase the performance. the CSMA method was developed.



Collision in CSMA

# CSMA(carrier sense multiple access)

- The possibility of collision still exists because of propagation delay; when a station sends a frame, it still takes time (although very short) for the first bit to reach every station and for every station to sense it
- a station may sense the medium and find it idle, only because the first bit sent by another station has not yet been received.
- *At time  $t_1$  station A senses the medium and finds it idle, so it sends a frame.*
- Station Z at right end of the medium senses the medium *At time  $t_2$  ( $t_2 > t_1$ )' and finds it idle because, at this time, the first bits from station A have not reached station Z, Station Z also sends a frame.*
- The two signals collide and both frames are destroyed.



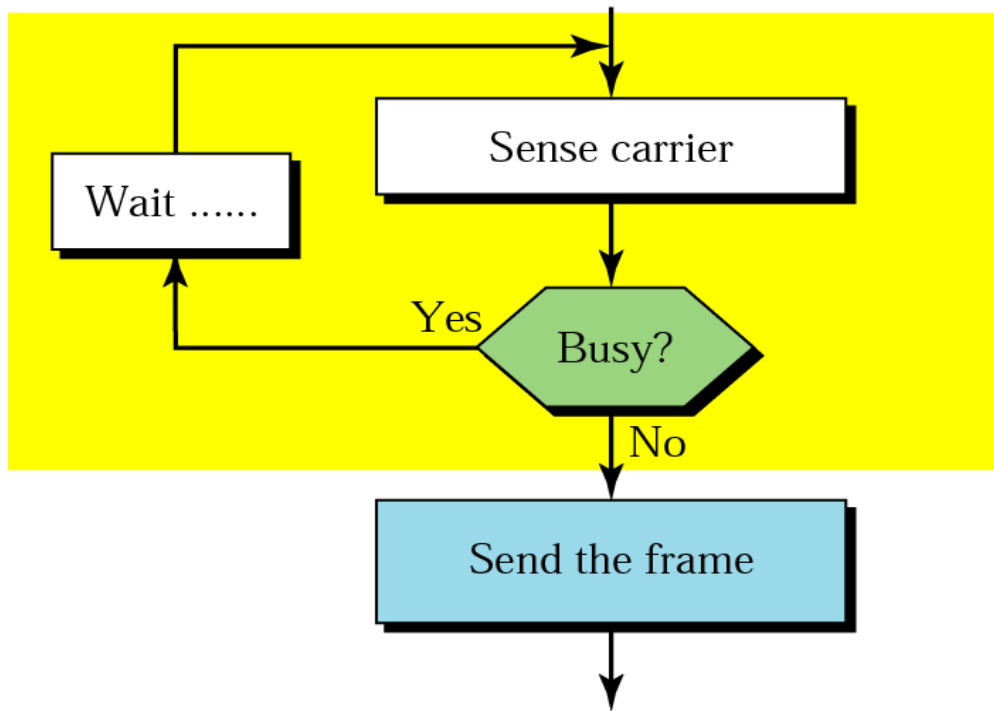
# Persistence methods

- **What should a stn do if channel is busy?**
- **What should a stn do if channel is idle?**
- Non persistent: stn senses the line. if line is not idle, it waits random amount of time and then senses again.
- 1-persistent: after stn finds line idle, it sends its frame immediately with probability 1.
- P-persistent: after stn finds line idle then
  - 1) with probability  $p$  stn sends frame.
  - 2) with probability  $q=1-p$ , stn waits for beginning of next time slot and checks again.
  - A) if line is idle goto step 1
    - b) If line is busy, it acts as though collision has occurred and uses backoff procedure.

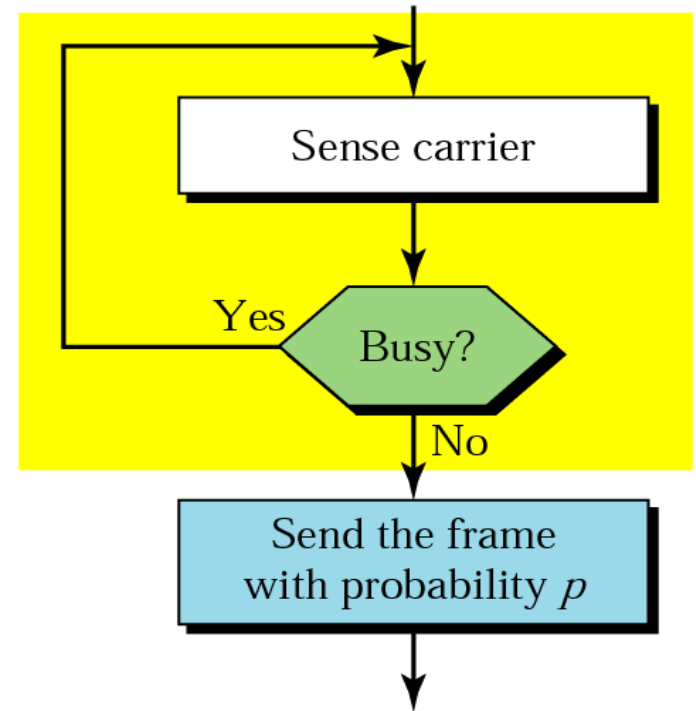


# Persistence strategies

Nonpersistent strategy



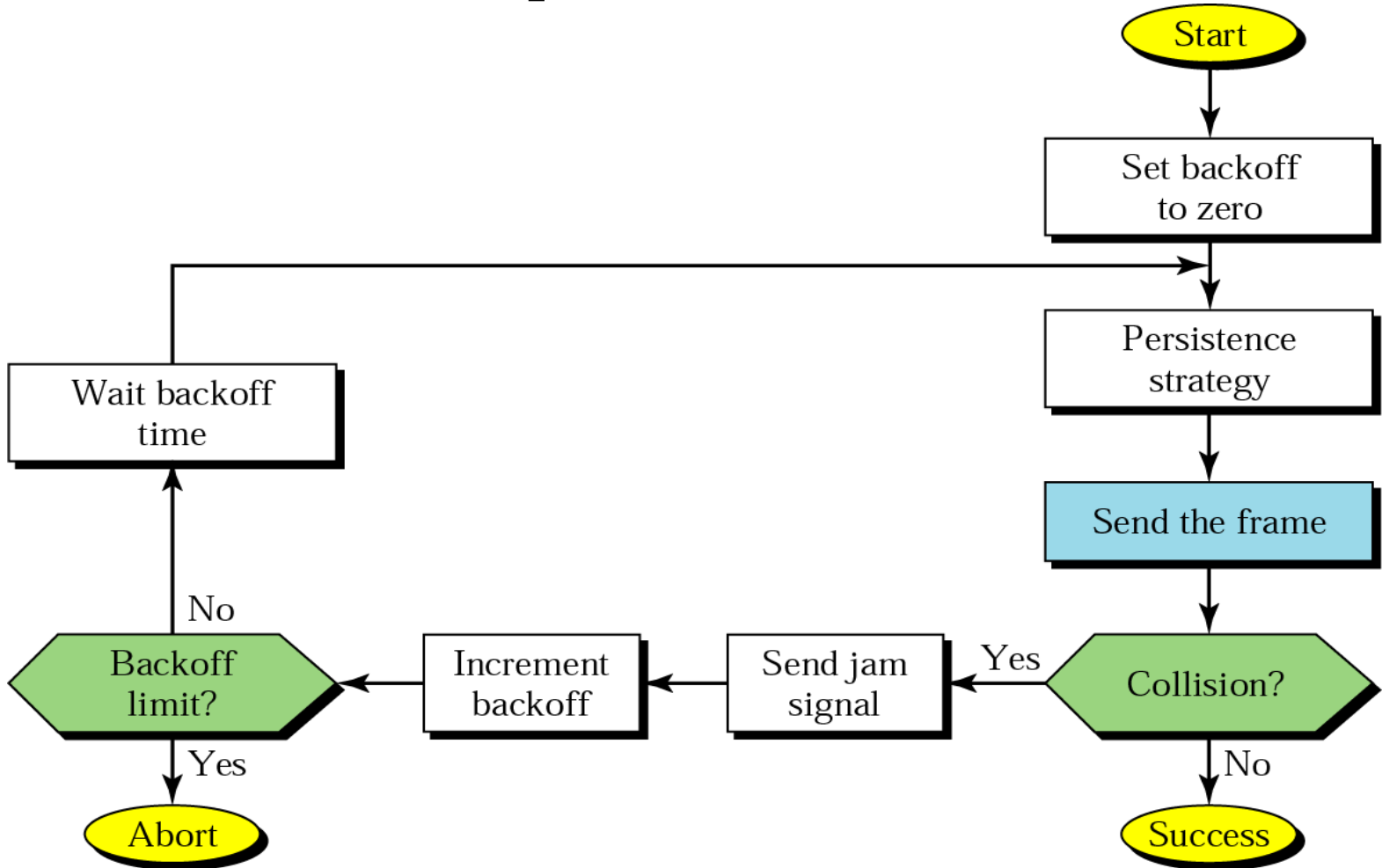
Persistent strategy



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

- The CSMA method does not specify the procedure following a collision. Carrier sense multiple access with collision detection (CSMA/CD) augments the algorithm to handle the collision.
- Csma/cd: In this mthd a stn monitors the medium after it sends the frame to see if the ftransmission was sucessful.
- if so,it is finished, otherwise frame is sent again.
- To reduce the probabiliy of collision the second time, the station waits—it needs to back-off
- It is reasonable that the station waits a little the first time, more if a collision occurs again, much more if it happens a third time, and so on.
- the station waits an amount of time between 0 and  $2^{n-1} \times \text{max\_propagation\_time}$  where n is no of attempted transmission.

# CSMA/CD procedure

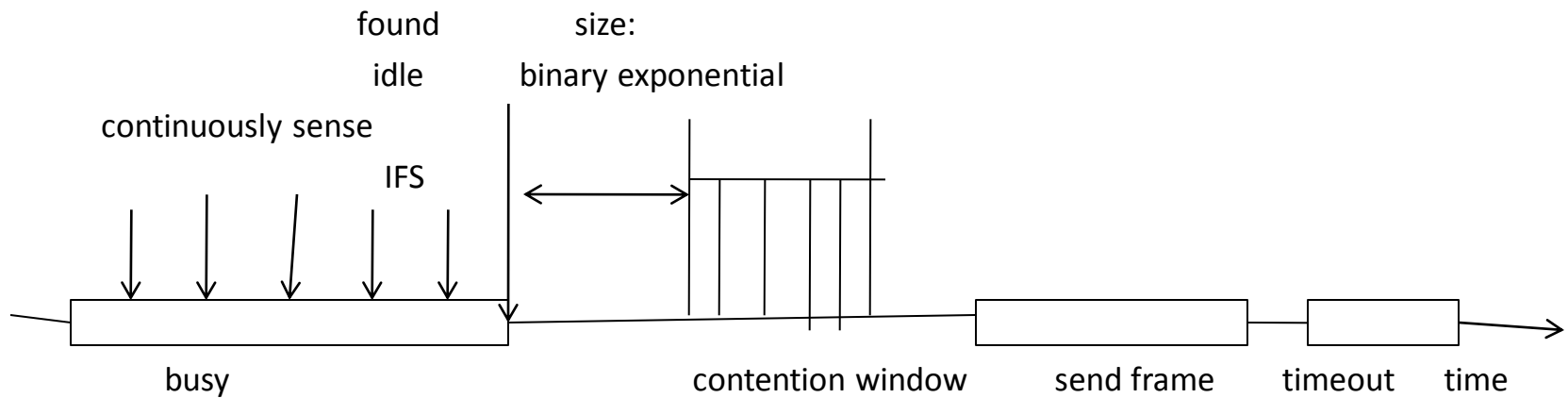


# CSMA/CA

- Collisions are avoided through the use of CSMA/CA's three strategies:
  - The inter-frame space or gap (IFS or IFG)
  - The contention window
  - Acknowledgments

# CSMA/CA

- Timing in csma/ca:



## CSMA/CA- Inter-frame Space (IFS)

- When an idle channel is found, the station does not send immediately. It waits for a period of time called the inter-frame space or IFS.
- Even though the channel may appear idle when it is sensed, a distant station may have already started transmitting.
- The distant station's signal has not yet reached this station.
- The IFS time allows the front of the transmitted signal by the distant station to reach this station
- If after the IFS time the channel is still idle, the station can send.

# CSMA/CA contention window

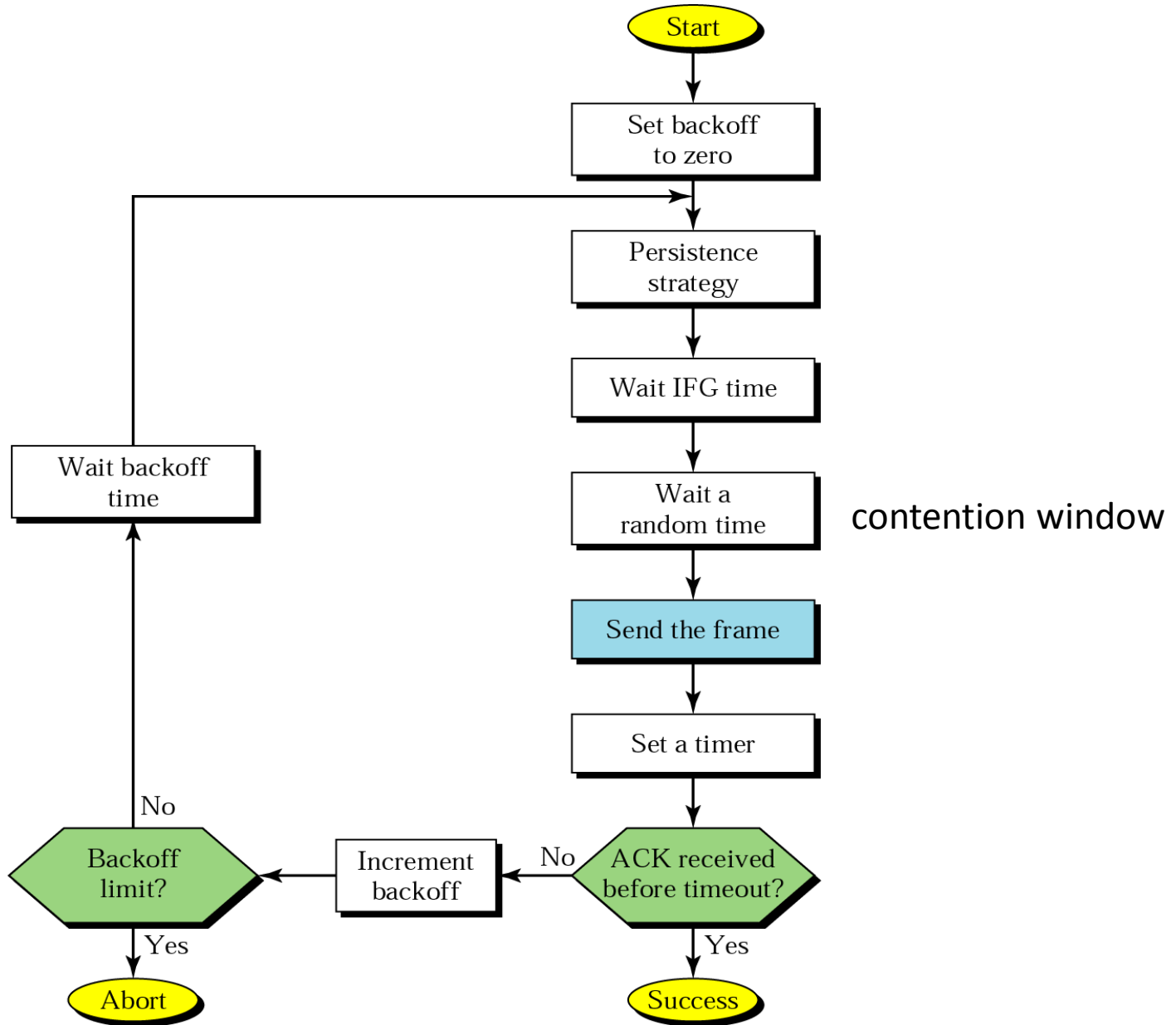
- The contention window is an amount of time divided into slots.
- A station that is ready to send chooses a random number of slots as its wait time.
- The number of slots in the window changes according to the binary exponential back-off strategy.
- This means that it is set to one slot the first time and then doubles each time the station cannot detect an idle channel after the IFS time.

# *Acknowledgment*

- With all these precautions, there still may be a collision resulting in destroyed data.
- In addition, the data may be corrupted during the transmission.
- The positive acknowledgment and the time-out timer can help guarantee that the receiver has received the frame.



# CSMA/CA procedure



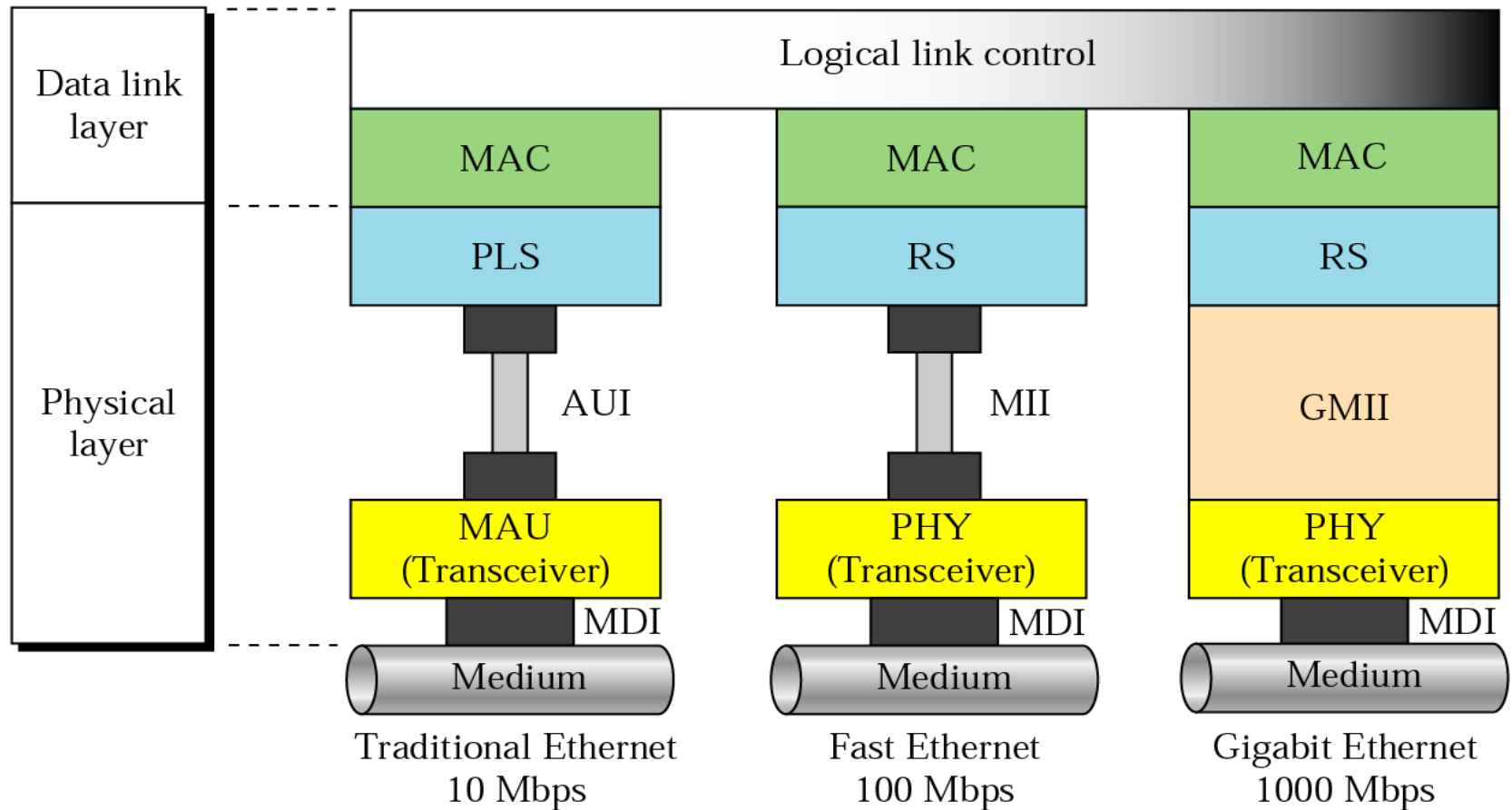
*Local Area  
Networks:  
Ethernet*

# Three generations of Ethernet

AUI: Attachment Unit Interface  
 MAC: Media Access Control  
 MAU: Medium Attachment Unit

MDI: Medium-Dependent Interface  
 MII: Medium-Independent Interface  
 GMII: Gigabit Medium-Independent Interface

PHY: Physical Layer Entity  
 PLS: Physical Layer Signaling  
 RS: Reconciliation Signaling



# Traditional Ethernet

*MAC Sublayer*

*Physical Layer*

*Physical Layer Implementation*

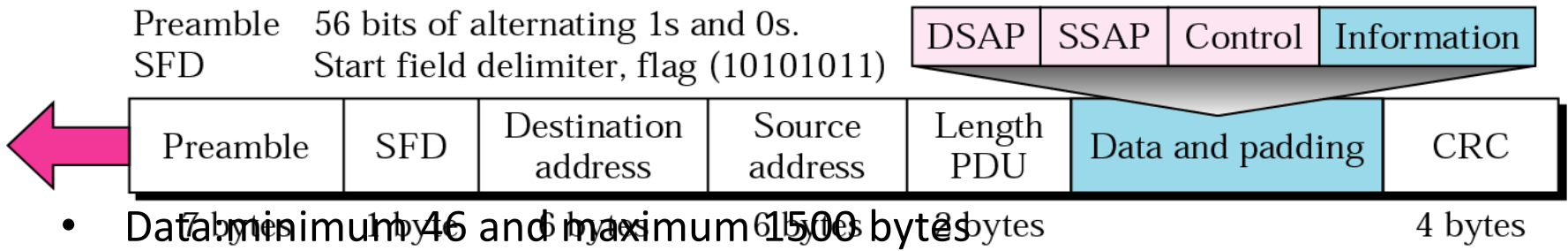
*Bridged Ethernet*

*Switched Ethernet*

*Full-Duplex Ethernet*

# 802.3 MAC frame

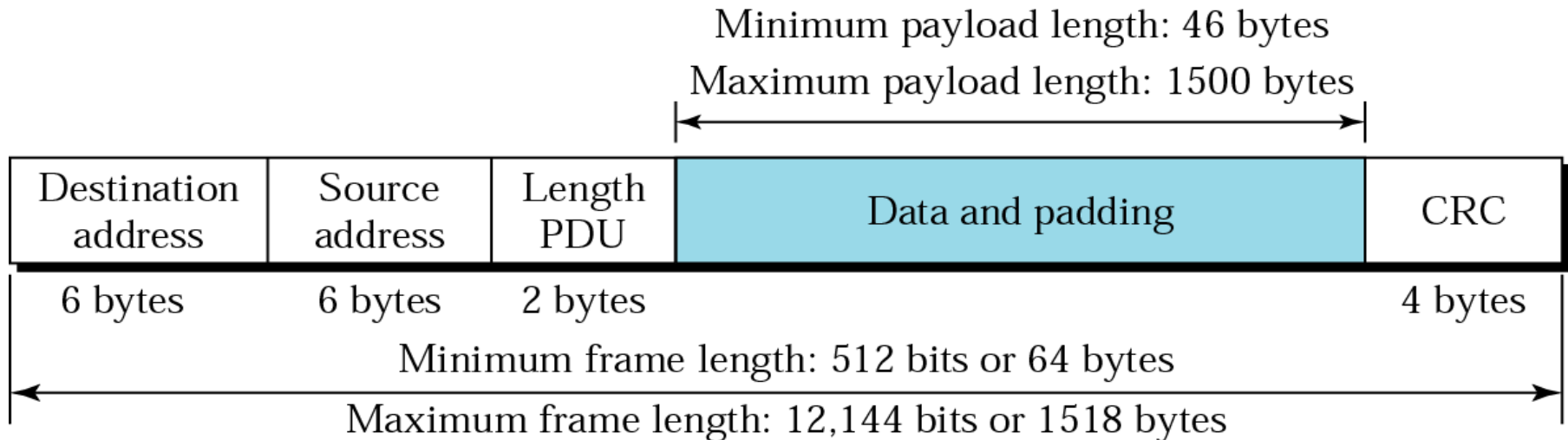
- Preamble: it alerts the receiver to coming frame and enables synchronization. it is added at physical layer
- Sfd: signals beginning of frame. sfd warns stns that this is last chance for sync. the last bits 11 alerts the recvr that next field is destination address.



- Data: minimum 46 and maximum 1500 bytes
- Crc: crc-32 is used
- Dsap: destination service access point.

# Minimum and maximum length

- Min length restriction is req for correct opn of csma/cd.of 64 bytes,header and trailer length is 18 bytes.so data 46 bytes
- Max length is 1518,of which header and trailer length is 18 bytes .so data 1500 bytes

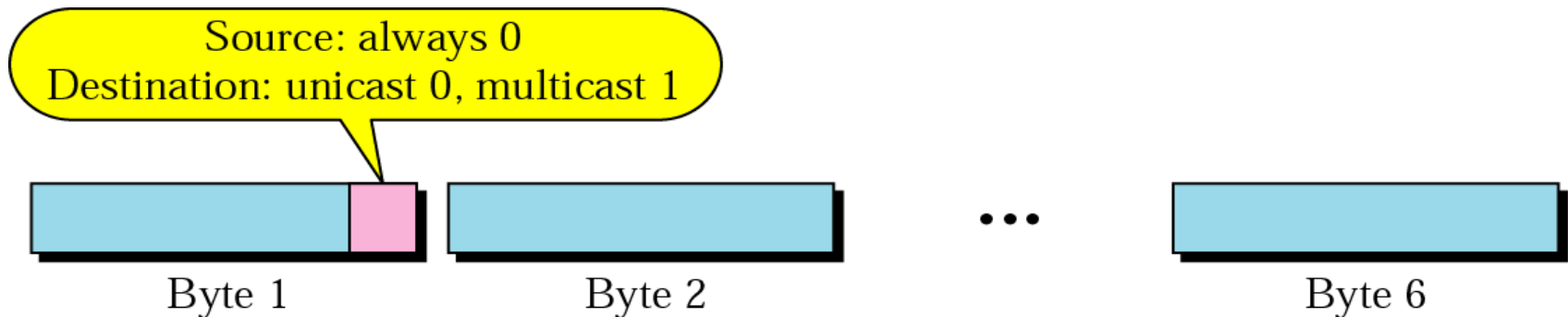


- Max restriction bcoz memory was expensive and it prevents 1 stn from monopolizing shared medium,blocking other stns that have data to send.

# Ethernet addresses in hexadecimal notation

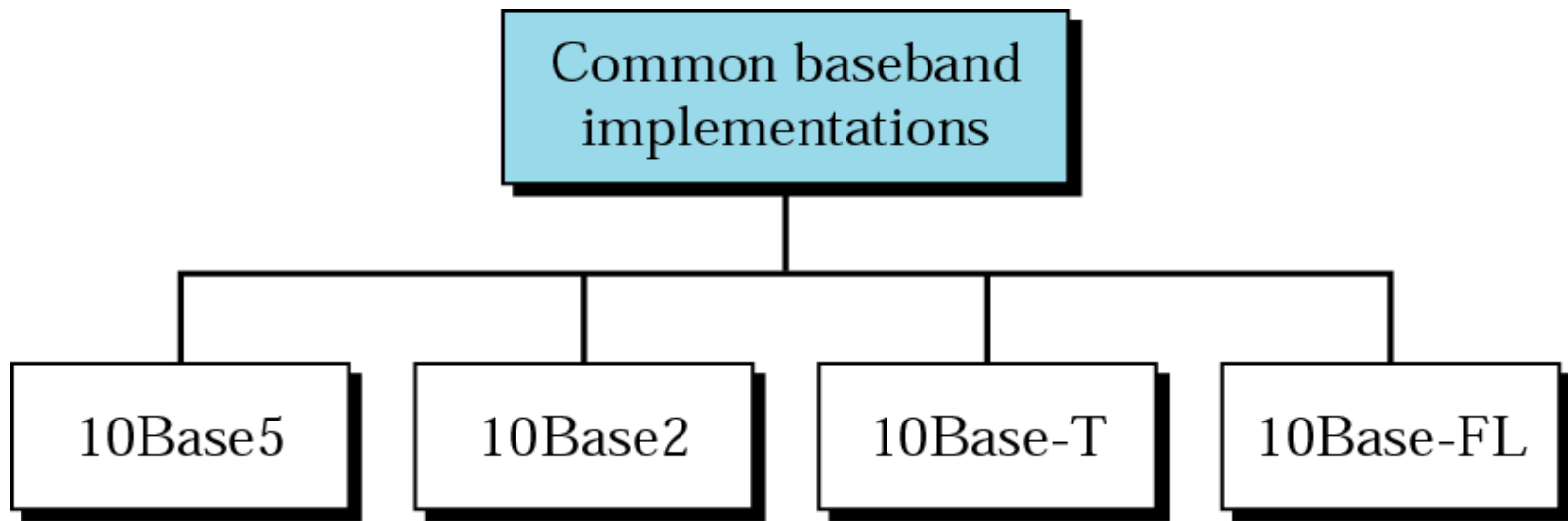
- Each stn on ethernet n/w has n/w interface card.(NIC).nic fits inside the stn and provides the stn with 6 byte physical address.
- In unicast dest address defines only 1 receipient,multicast defines group of adreses,broadcast are all stns on lan.its address is 48 1s.

**06-01-02-01-2C-4B**



# Categories of traditional Ethernet

- All standard implementations use digital signaling at 10 mbps. at sender data are converted to digital signal using manchester scheme



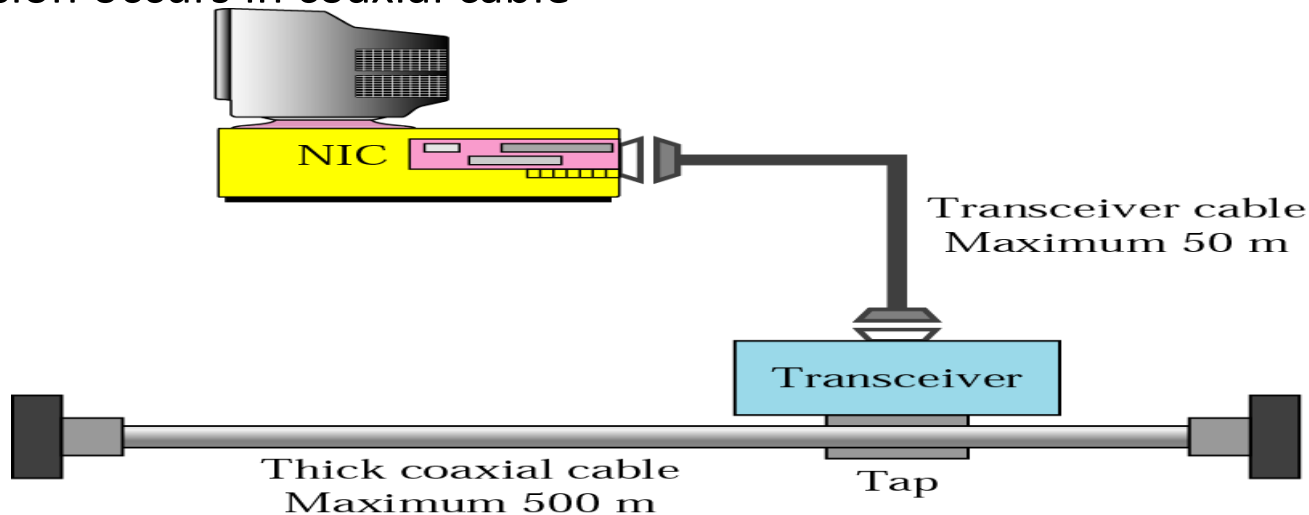
- Bus,thick coaxial      bus,thin coaxial      star,utp      star,fiber



# Connection of a station to the medium using 10Base5

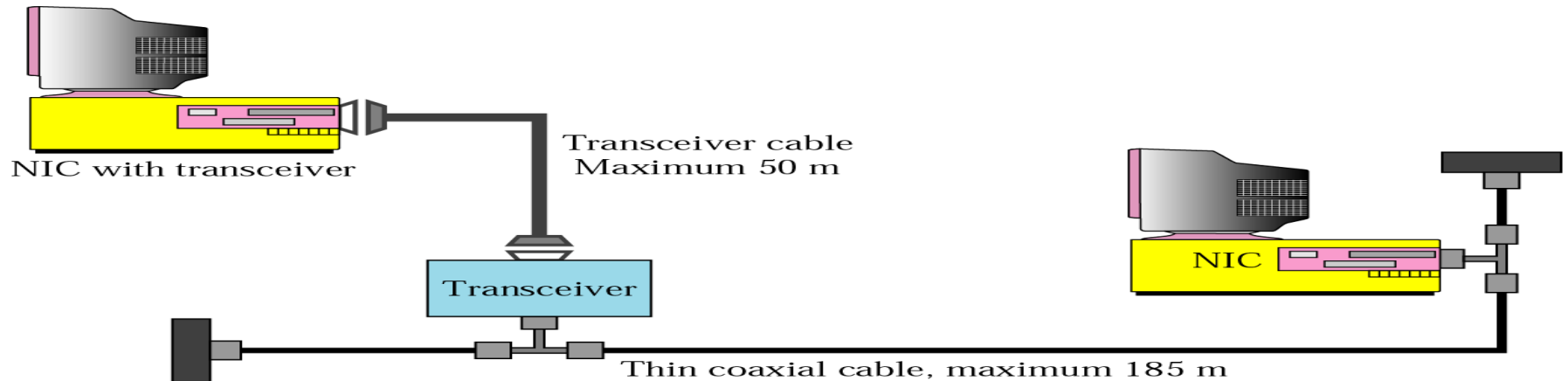
## Thick ethernet

- First implementation is called 10base5,thick ethernet or thicknet.uses bus topology with external transceiver (transmitter/receiver) connected via a tap to thick coaxial cable. Can't bend
- 10 base 5(10 Mbps baseband 500m)
- Transceiver is responsible for transmitting,receiving and detecting collisions.  
. collision occurs in coaxial cable



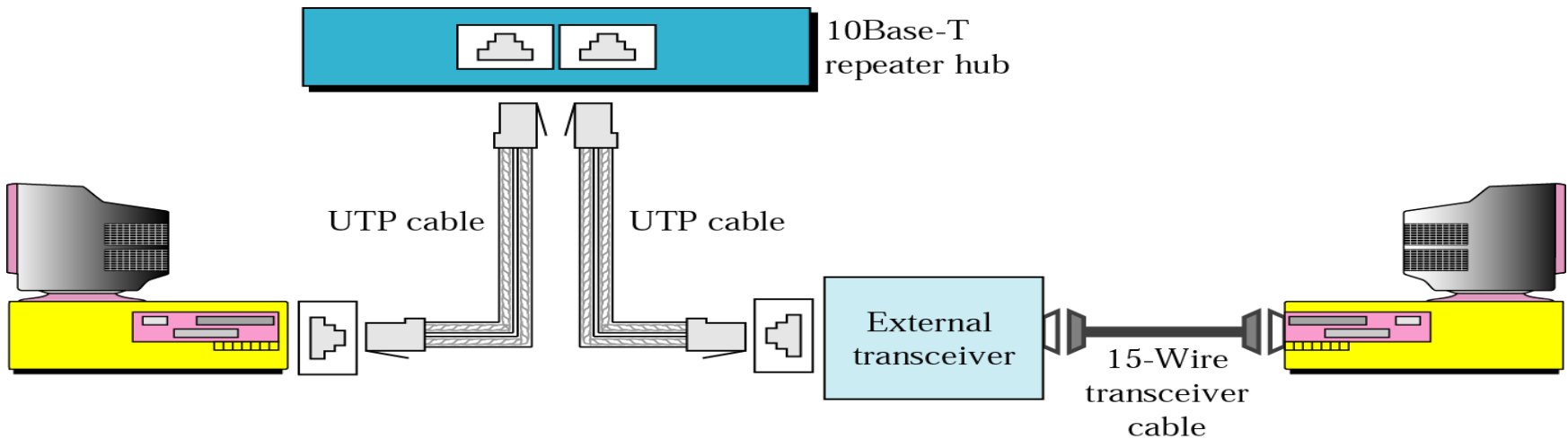
# Connection of stations to the medium using 10Base2 Thin Ethernet

- Second implementation is called 10base2, thin ethernet or cheapernet. uses bus topology with thinner and flexible cable. can bend.
- Transceiver is part of n/w interface card, installed inside stn.
- 10 base 2(10 Mbps baseband 185m)
- Installation is simpler. collision occurs in coaxial cable



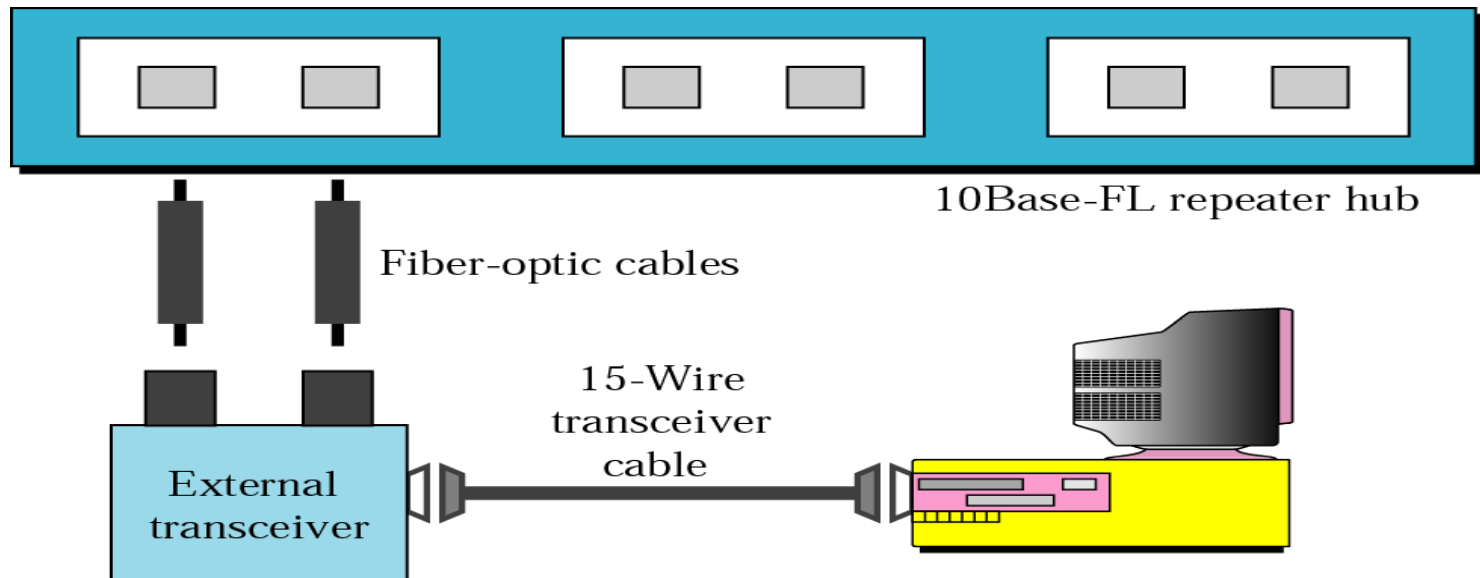
# Connection of stations to the medium using 10Base-T Twisted pair Ethernet

- Third implementation is called 10baseT, Twisted pair ethernet.uses physical star topology .stns are connected to a hub via 2 pairs of twisted cable.collisions occur in hub.max length is 100m to minimise effect of attenuation.
- 10 base T(10 Mbps baseband Twisted pair)



# Connection of stations to the medium using 10Base-FL Fiber Ethernet

- Uses star topology. stns are connected to a hub via 2 pairs of fiber optic cable.upto 2000 m
- 10 base T(10 Mbps baseband fiber optic pair)



# Fast Ethernet

*MAC Sublayer*

*Physical Layer*

*Physical Layer Implementation*

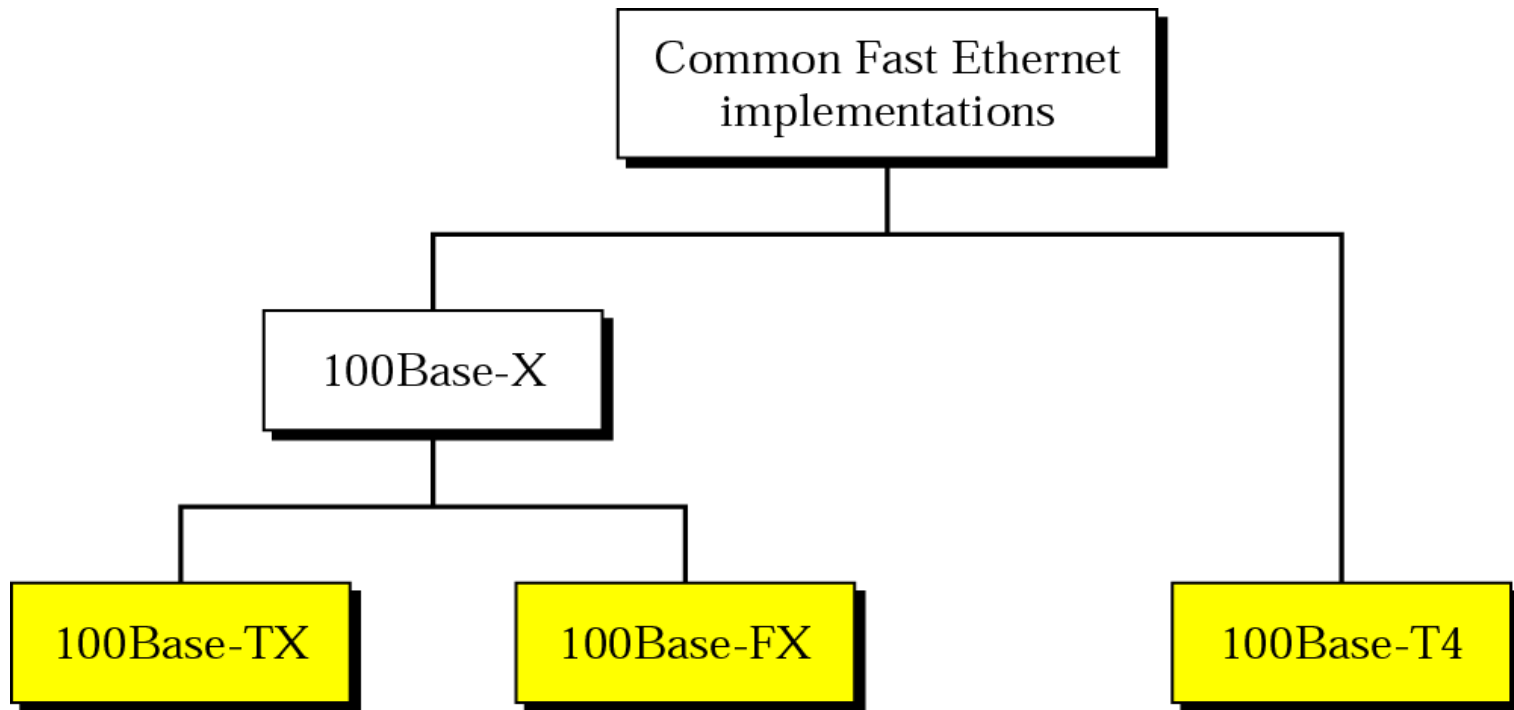


# Fast Ethernet physical layer

## Fast Ethernet

- It is designed to compete with lan protocols such as fddi or fiber channel.it is backward compatible with stnd Ethernet.
- **Goals:** 1)upgrade the datarate to 100mbps
- 2) compatible with stnd Ethernet.
- 3)keep same 48 bit address.
- 4)keep same frame format.
- 5)keep same min and max lengths.
- **Mac sublayer**
- Keep mac layer untouched,but drop bus topology and consider star topology-half & full duplex.in half duplex,stns r connected via hub and acess mthd is csma/cd.in full duplex,connection is made via switch with buffers at each port no need of acess mthd as csma/cd.
- **Autonegotiation:** it allows a stn or a hub a range of capabilities.it allows 2 devices to negotiate the mode r data rate of opn.

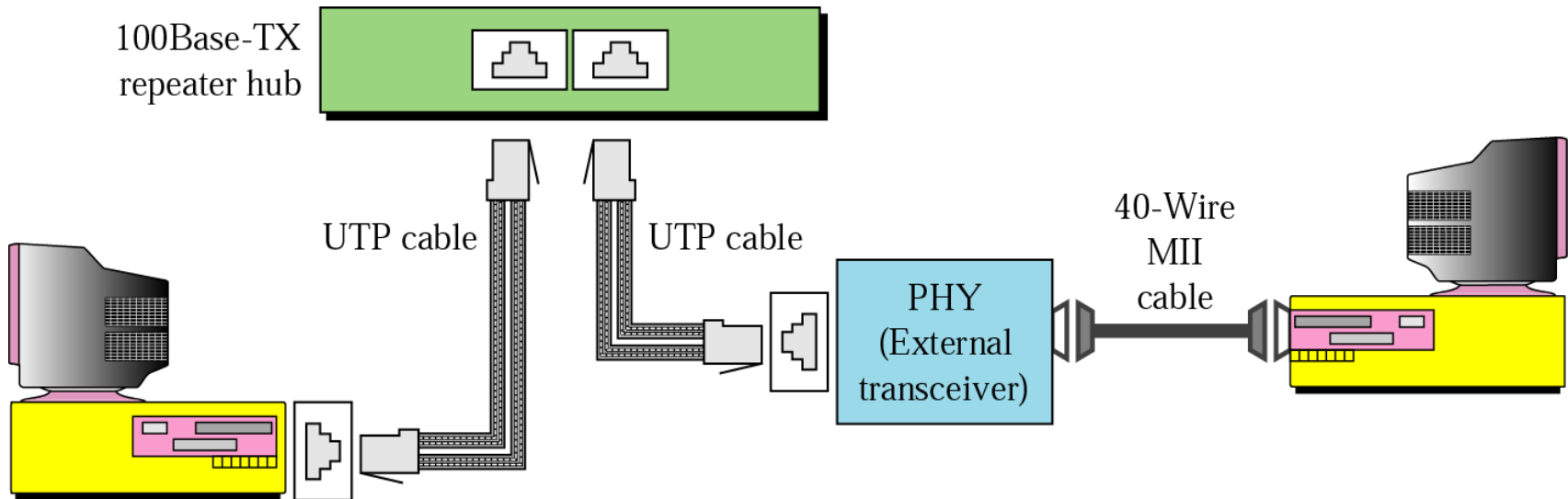
# Fast Ethernet implementations



- 2 wires category 5 UTP 2 wires, fiber 4 wires category 3 UTP

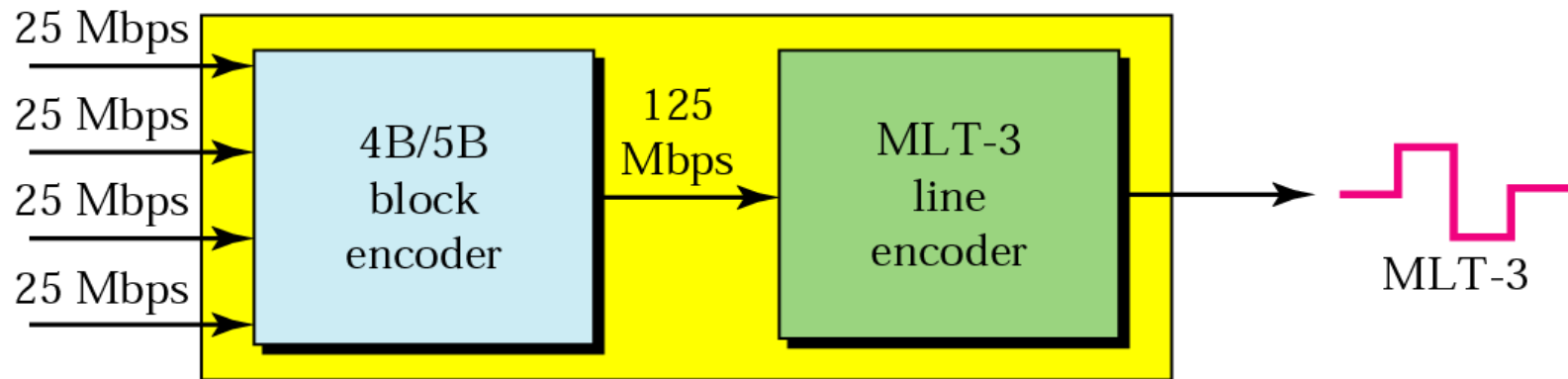
# 100Base-TX implementation

- It uses 2 pairs of twisted pair cables.
- It uses ml3 scheme since it has good performance. since ml3 is not self synchronous, 4B/5B encoding is used for bit synchronization by preventing long sequence of 0's and 1's. this creates a data rate of 125 mbps, which is fed into ml3 for encoding



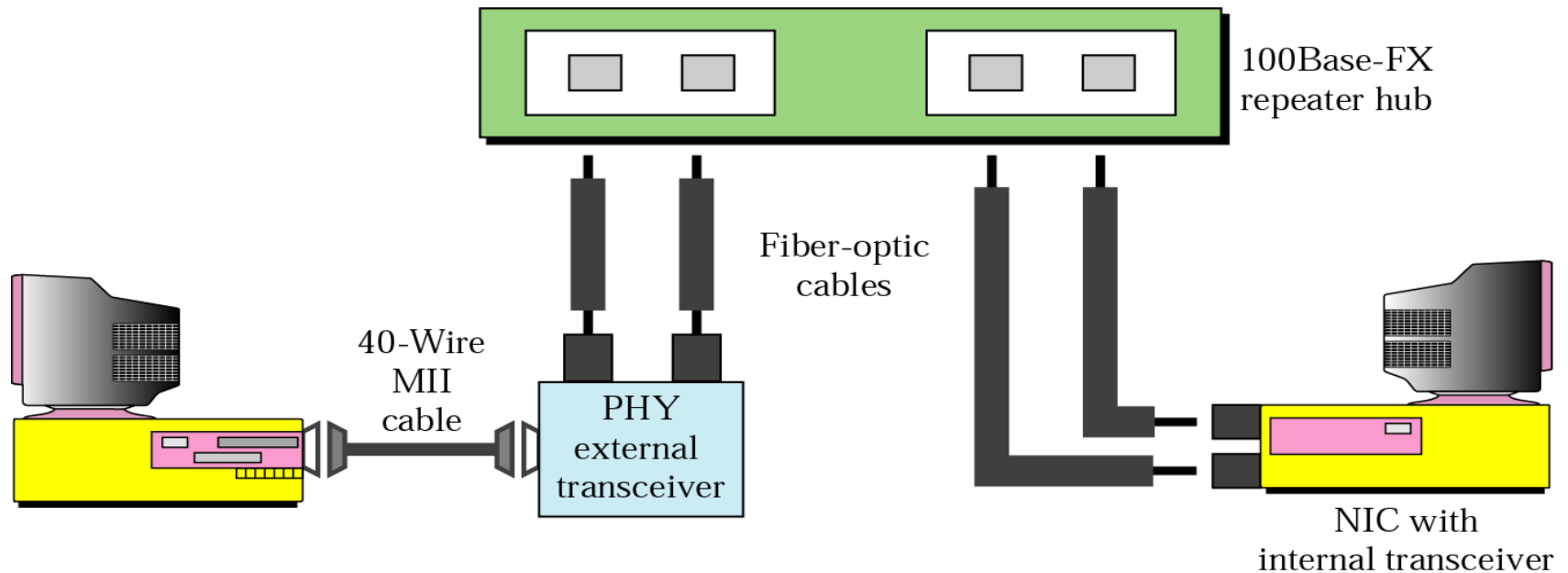


# Encoding and decoding in 100Base-TX

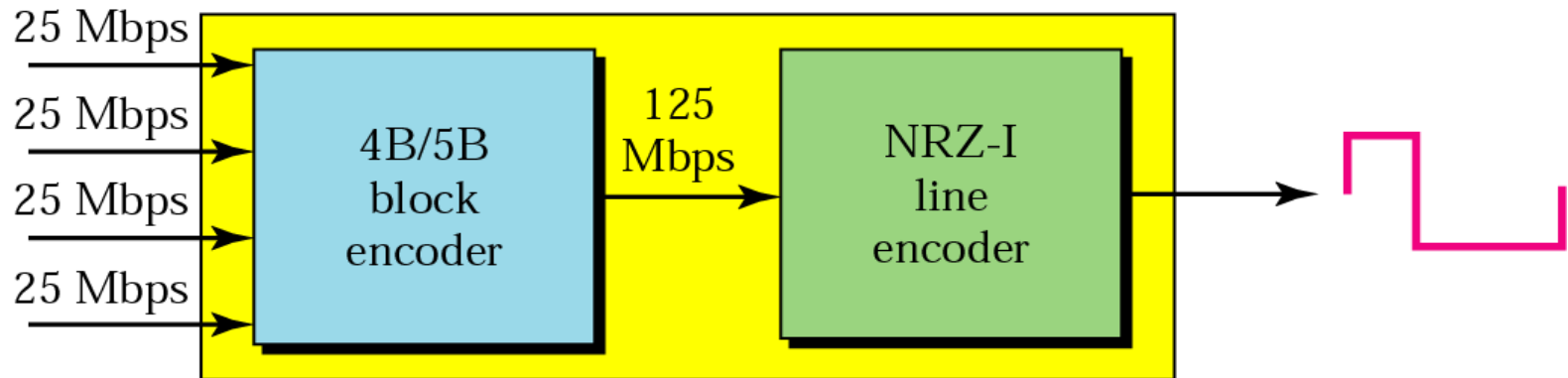


# 100Base-FX implementation

- It uses 2 pairs of fiber optic cables. fiber optic cables can easily handle high bandwidth requirements by using encoding schemes.
- Nrz-I has bit synchronization prblm for long seq of 0's.designers use 4b/5b block encoding.

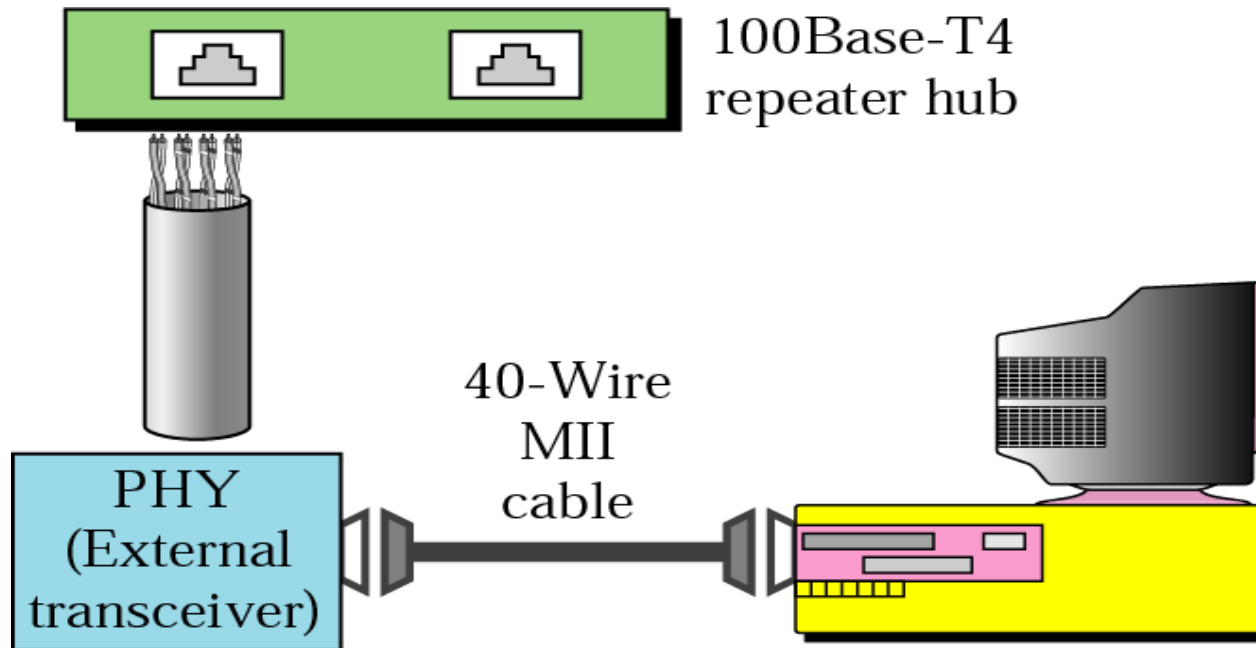


# Encoding and decoding in 100Base-FX

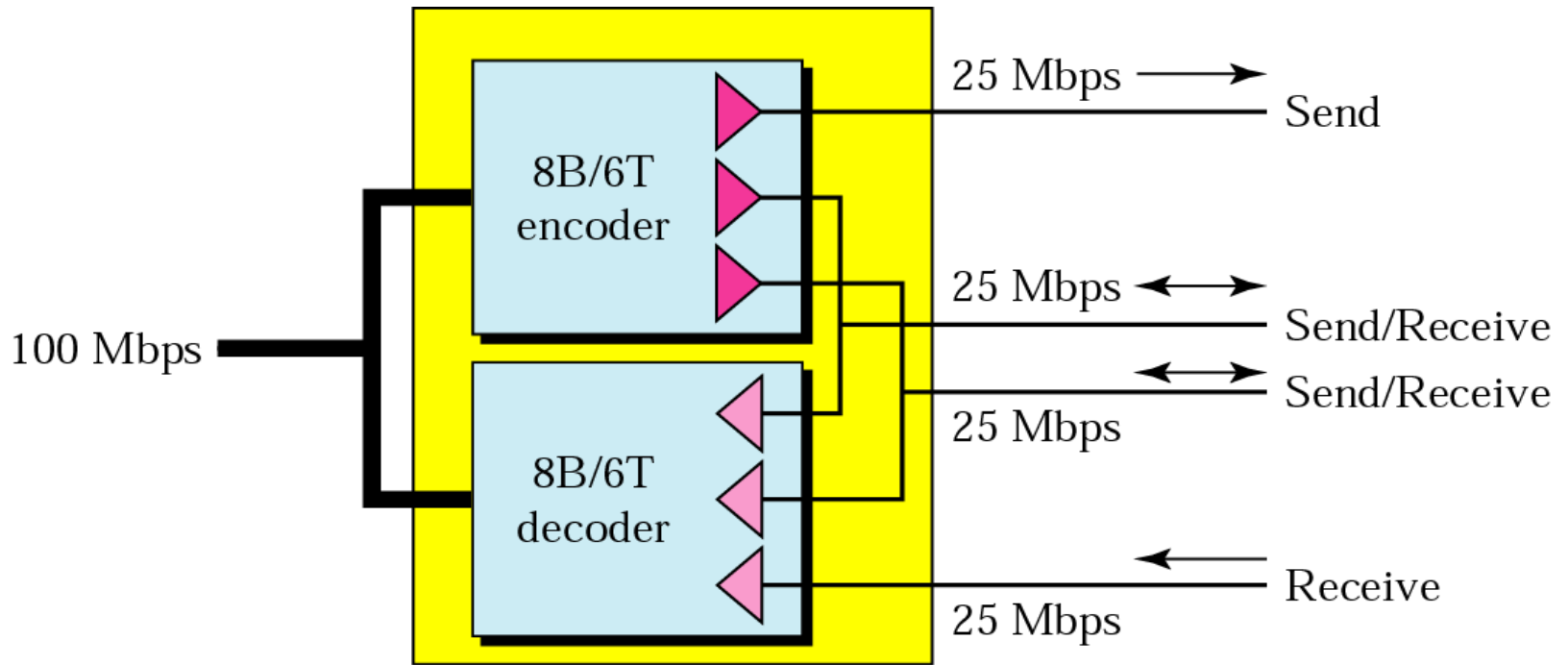


# 100Base-T4 implementation

- Uses category 3 or higher utp. encoding and decoding is complicated. uses 4 pairs of utp for transmitting.



# Using four wires in 100Base-T4



# Gigabit Ethernet

*MAC Sublayer*

*Physical Layer*

*Physical Layer Implementation*

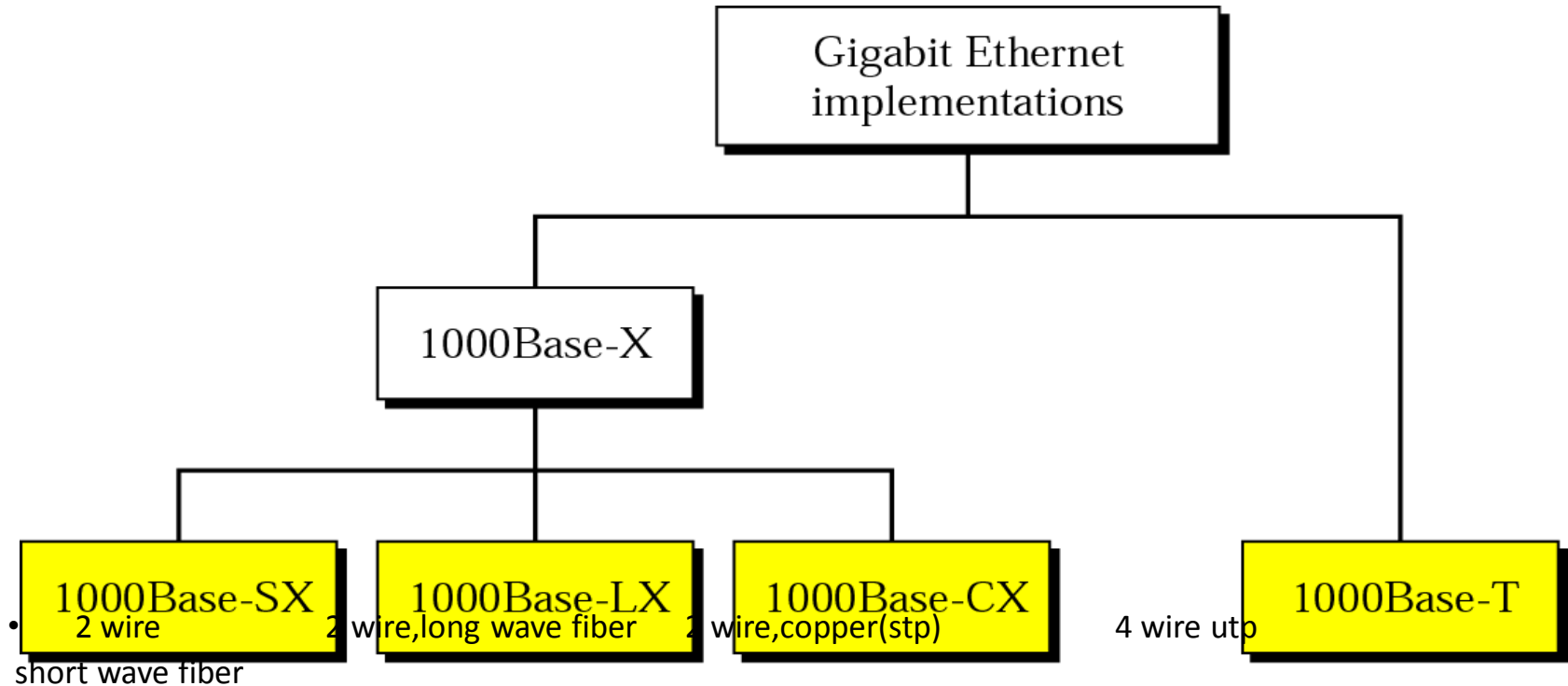


# Physical layer in Gigabit Ethernet

- Need for even higher data rate resulted in design of gigabit ethernet(1000mbps)
- **Goals:** 1)upgrade the data rate to 1gbps.
- 2) compatible with standard ethernet.
- 3)keep same 48 bit address.
- 4)keep same frame format.
- 5)keep same min and max lengths.
- 6)to support autonegotiation.
- **Mac sublayer:** it is untouched.uses half & full duplex.
- In full duplex there is central switch connected to all computers.each switch has buffers for each input port in which data is stored until they are transmitted.no collision in this mode.csma/cd not used.
- In half duplex,a switch is replaced by hub,which acts as a common cable in which collision might occur.uses csma/cd.

# Gigabit Ethernet implementations

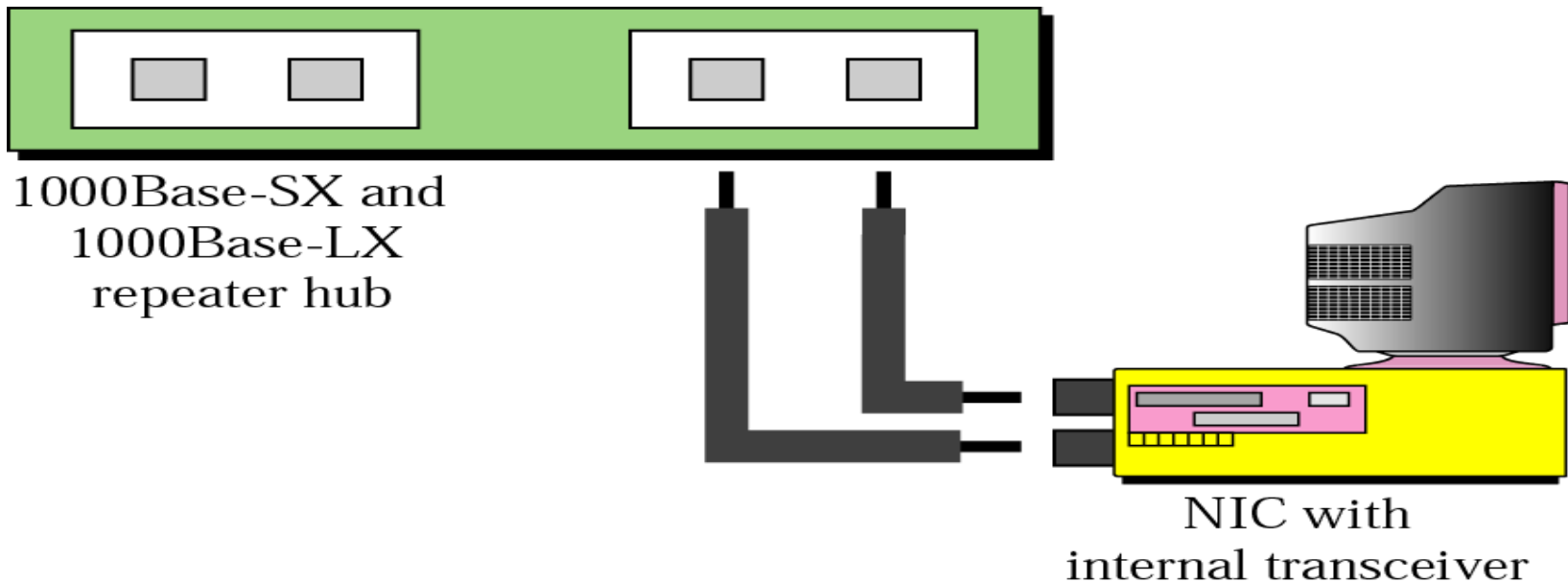
- In traditional approach we keep minimum length of frame as 512 bits. so slot time in gigabit ethernet is  $512 \text{ bits} \times 1/1000 \text{ micro secs}$ . reduced slot time means collision is detected 100 times earlier.



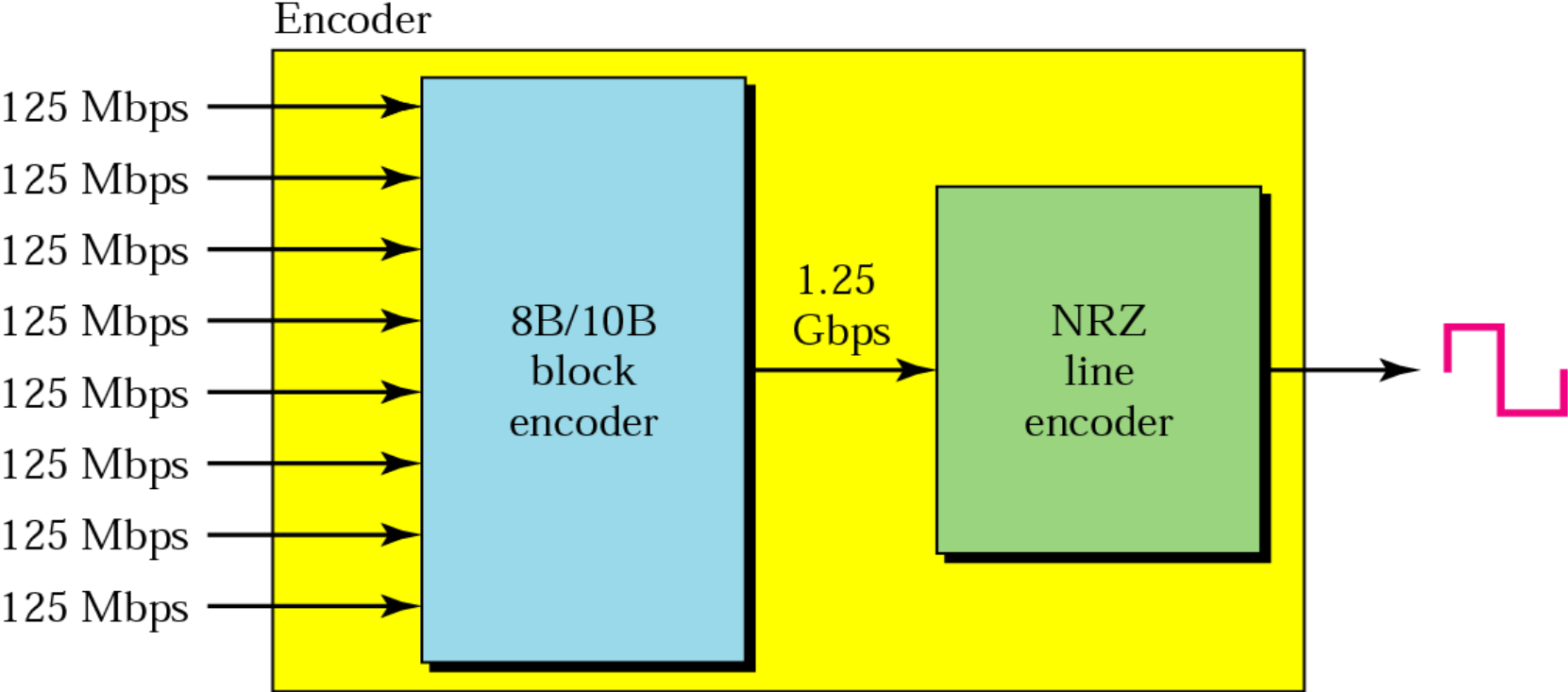


# 1000Base-X implementation

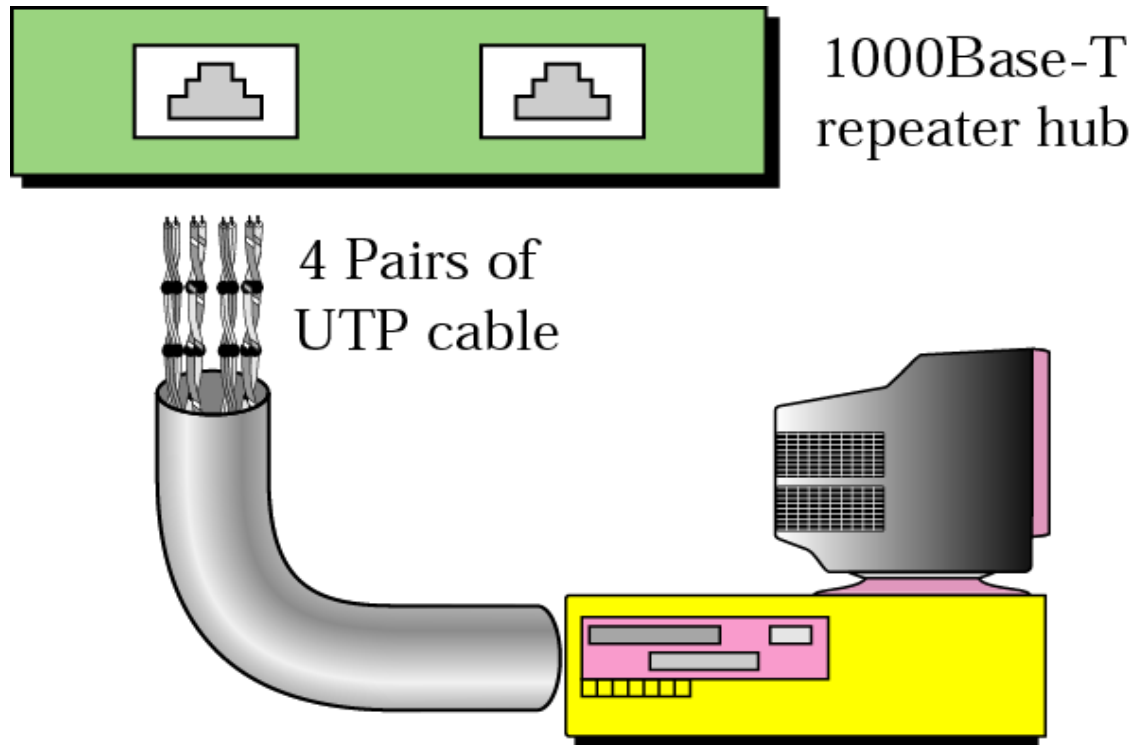
- 1000 base-T was designed in response to those users who had already installed this wiring for other purposes such as fast ethernet.
- Gigabit ethernet cannot use manchester encoding scheme bcoz it involves very high bandwidth. 8b/10b block encoding is used.



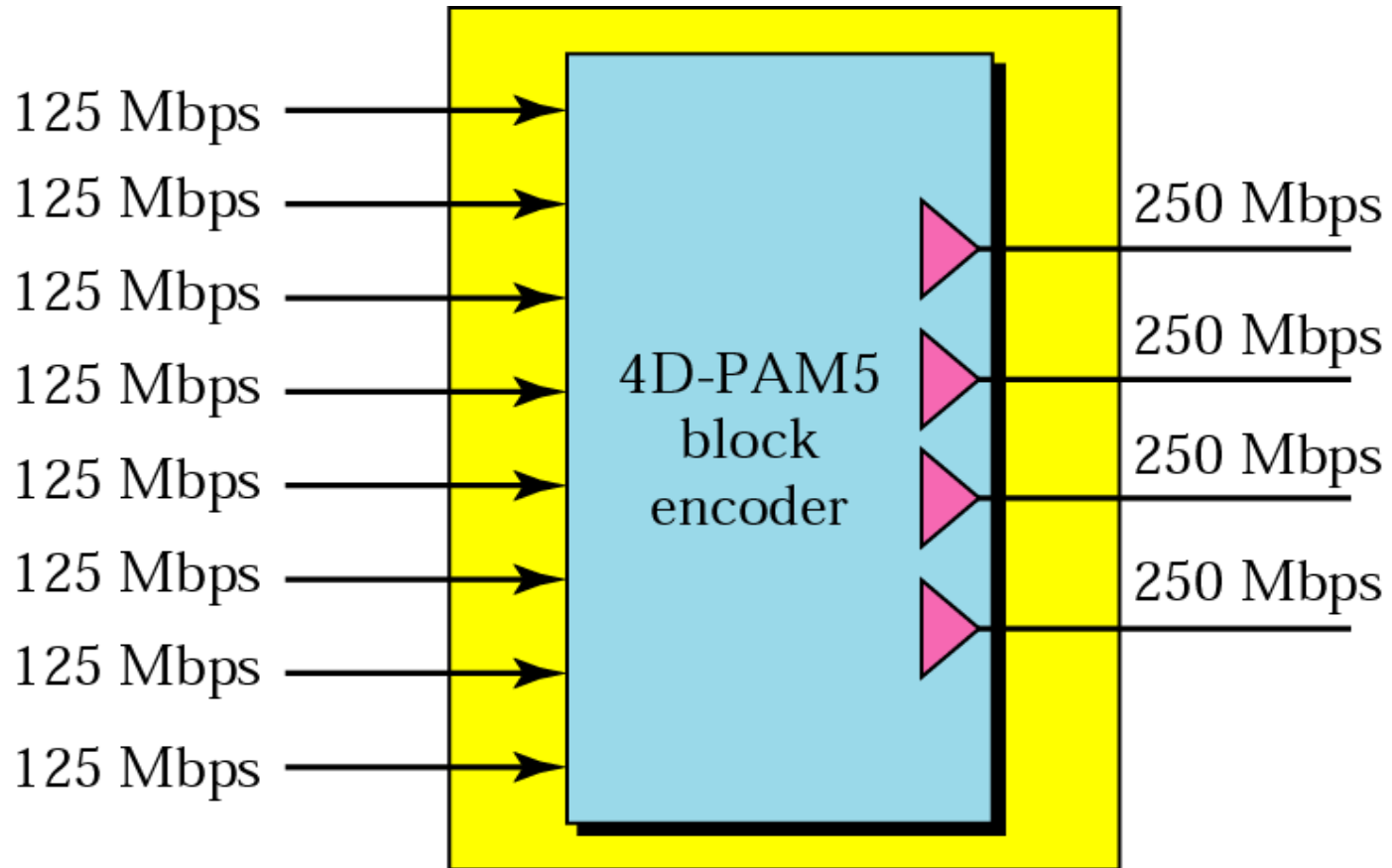
# Encoding in 1000Base-X



# 1000Base-T implementation



# Encoding in 1000Base-T



# Connecting devices

- LANs do not normally operate **in isolation**. **They are connected to one another or to the Internet.**
- To connect LANs, or segments of LANs, we use connecting devices.
- Connecting devices can operate **in different layers of the Internet model.**

# Connecting devices

*Repeaters*

*Hubs*

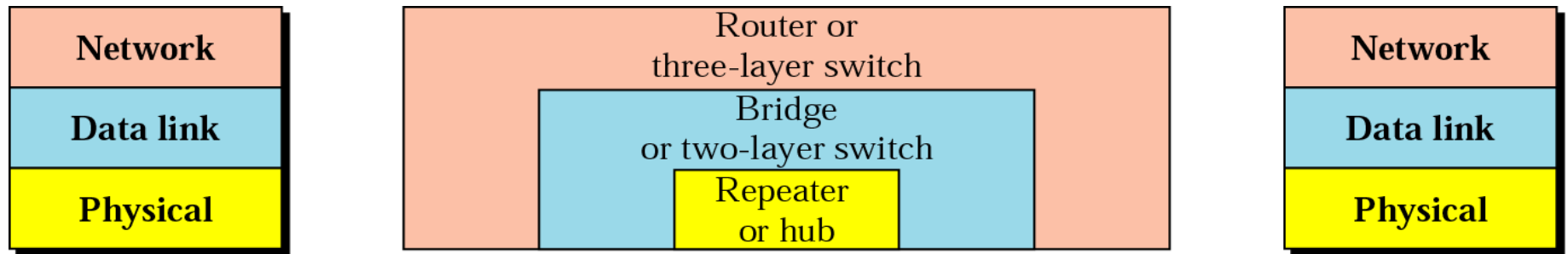
*Bridges*

*Two-Layer Switches*

# The five categories contain devices which can be defined as

- 1.Those which operate below the physical layer such as a passive hub.
- 2.Those which operate at the physical layer (a repeater or an active hub).
- 3.Those which operate at the physical and data link layers (a bridge or a two-layer switch).
- 4. Those which operate at the physical, data link, and network layers (a router or a three-layer switch).
- 5. Those which can operate at all five layers (a gateway).

# Connecting devices





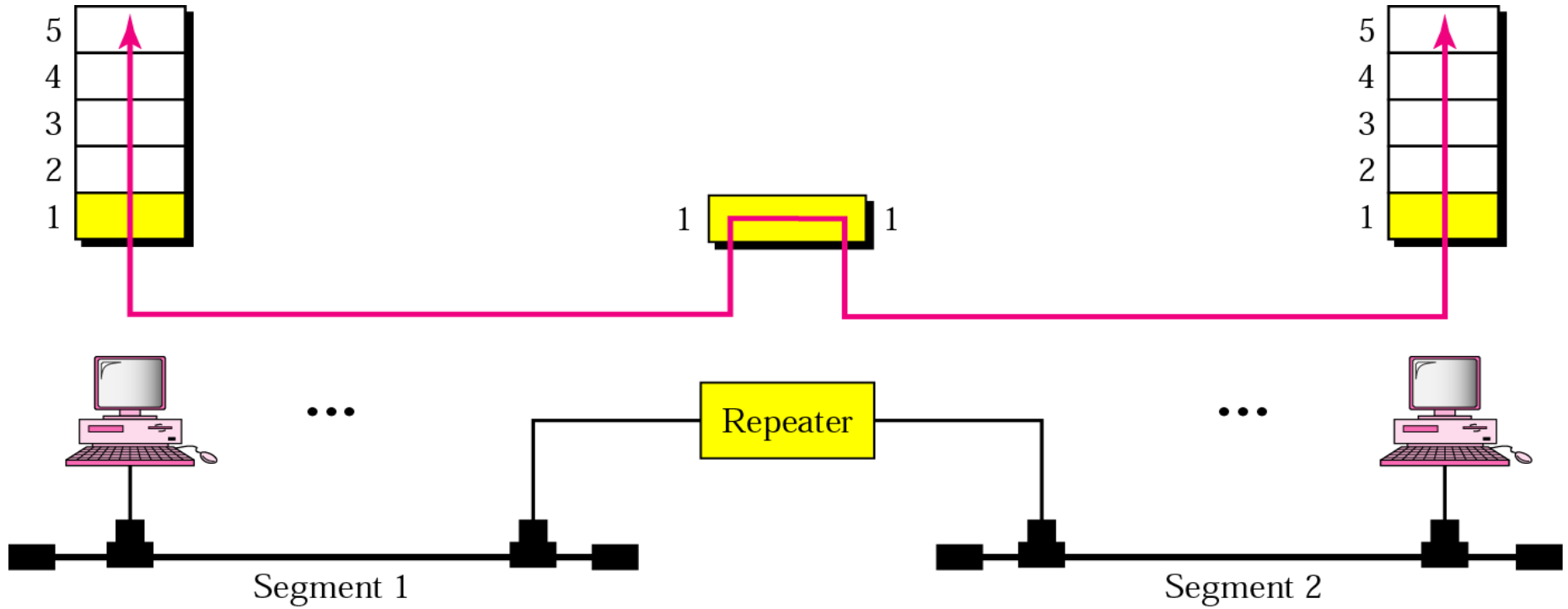
# Passive Hubs

- A passive hub is just a connector. It connects the wires coming from different branches.
- The hub is the collision point where the signals coming from different stations collide.
- This type of a hub is part of the media
- Its location in the Internet model is below the physical layer.

# Repeaters

- 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 it becomes too weak or corrupted.
- A repeater receives a signal and, regenerates the original bit pattern.
- The repeater then sends the refreshed signal.
- A repeater can extend the physical length of a LAN.

# Repeater





Note:

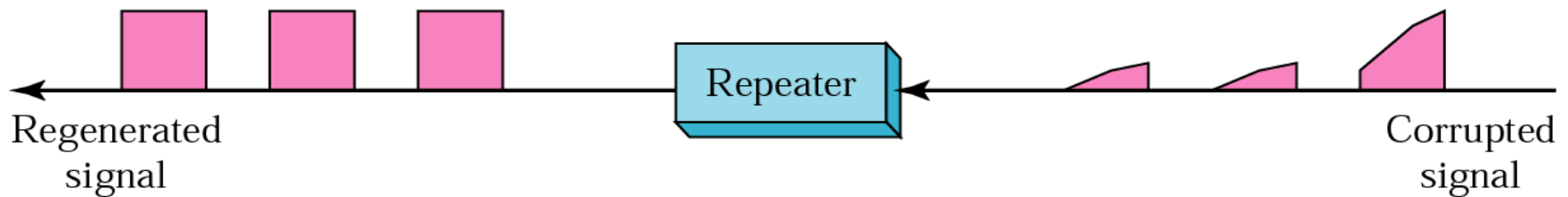
*A repeater connects segments of a LAN.*

*A repeater forwards every frame; it has no filtering capability.*

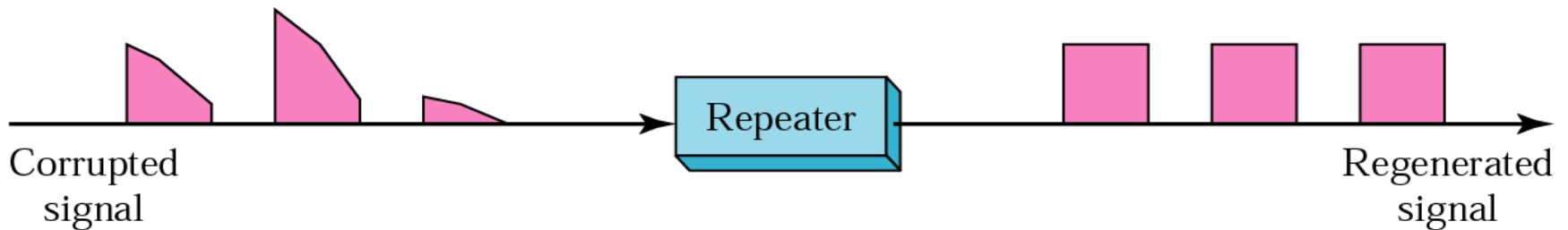
*A repeater is a regenerator, not an amplifier.*

# Function of a repeater

- Location of repeater on a link is vital.
- A repeater must be placed so that a signal reaches it before any noise changes meaning of bit.



a. Right-to-left transmission.

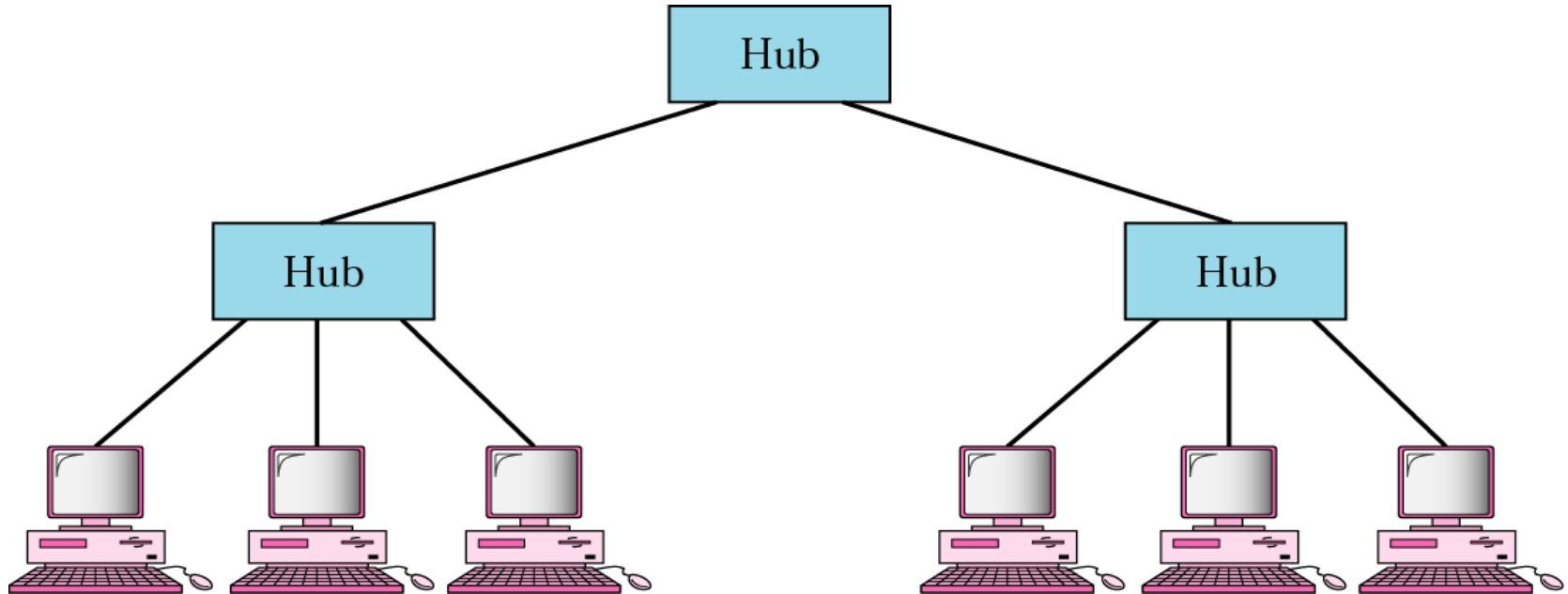


b. Left-to-right transmission.

# Active Hubs

- An active hub is actually a multipart repeater.
- It is normally used to create connections between stations in a physical star topology.
- Hubs can also be used to create multiple levels of hierarchy
- The hierarchical use of hubs removes the length limitation of 10Base-T (100 m).

# Active Hubs



# Bridges

- A bridge operates in both the physical and the data link layer.
- As a physical layer device, it regenerates the signal it receives.
- As a data link layer device, the bridge can check the physical (MAC) addresses (source and destination) contained in the frame.

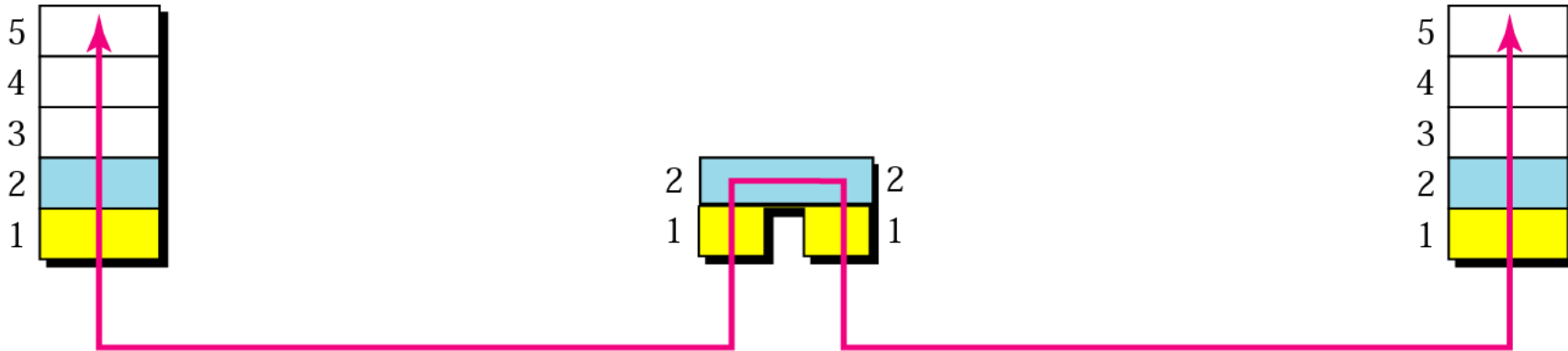




*A bridge operates in both physical and dll. A bridge has a table used in filtering decisions.*

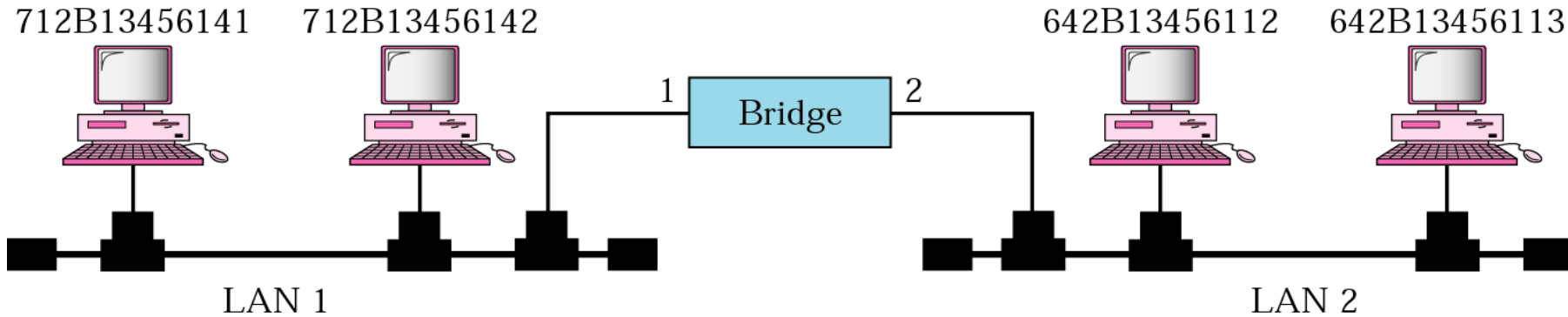
*A bridge does not change the physical (MAC) addresses in a frame.*

# Bridge



Address	Port
712B13456141	1
712B13456142	1
642B13456112	2
642B13456113	2

Bridge Table



# Switches

- switch can mean two different things.
  - **A two-layer switch**
  - **A three-layer switch**
- **A three-layer switch ->**
  - **is used at the network layer; it is a kind of router.**
- **The two-layer switch performs at the physical and data link layers.**

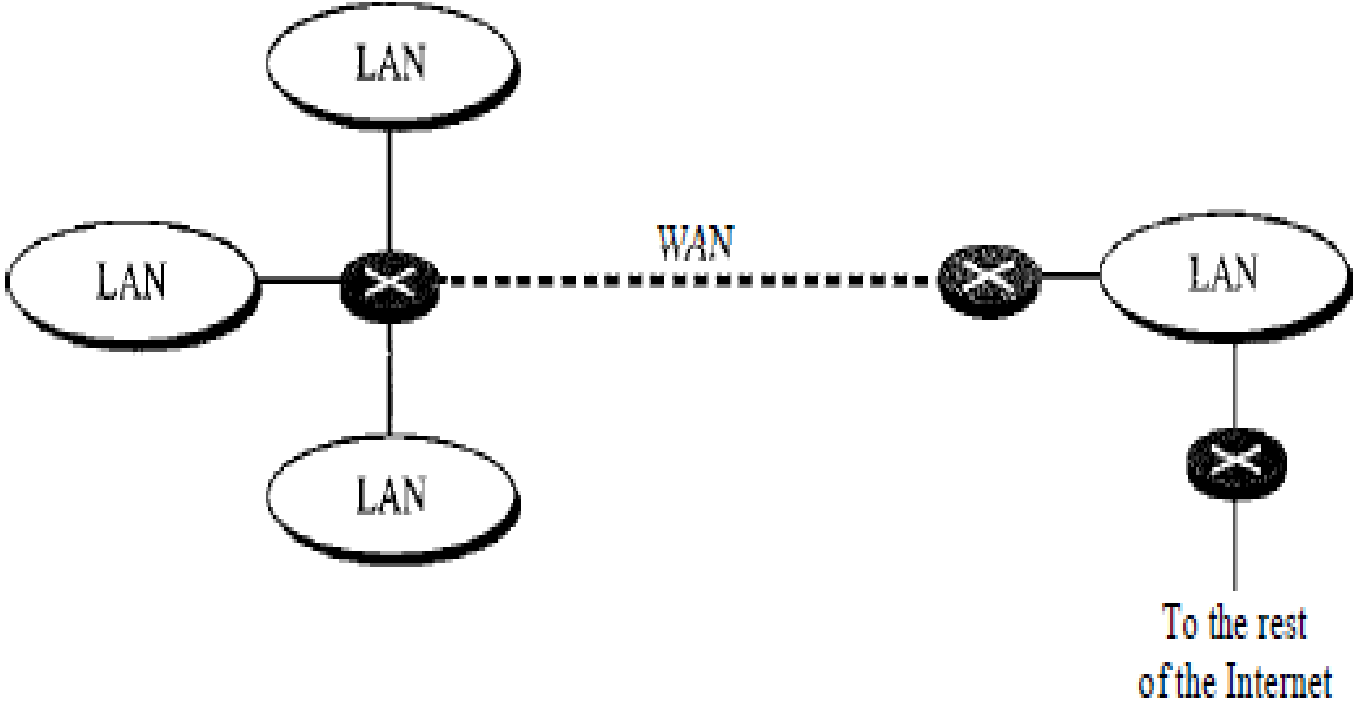
# A two –layer switches

- A two-layer switch is a bridge, a bridge with many ports and a design that allows better (faster) performance.
- makes a filtering decision based on the MAC address of the frame it received
- It can have a buffer to hold the frames for processing.
- It can have a switching factor that forwards the frames faster. Some new two-layer switches, called ***cut-through switches***, have been designed to forward the frame as soon as they check the MAC addresses in the header of the frame.

# Routers

- A router is a three-layer device that routes packets based on their logical addresses (host-to-host addressing).
- A router normally connects LANs and WANs in the Internet and has a routing table that is used for making decisions about the route.

# Routers



# A three –layer switches

- A three-layer switch is a router, but a faster and more sophisticated.
- The switching fabric in a three-layer switch allows faster table lookup and forwarding

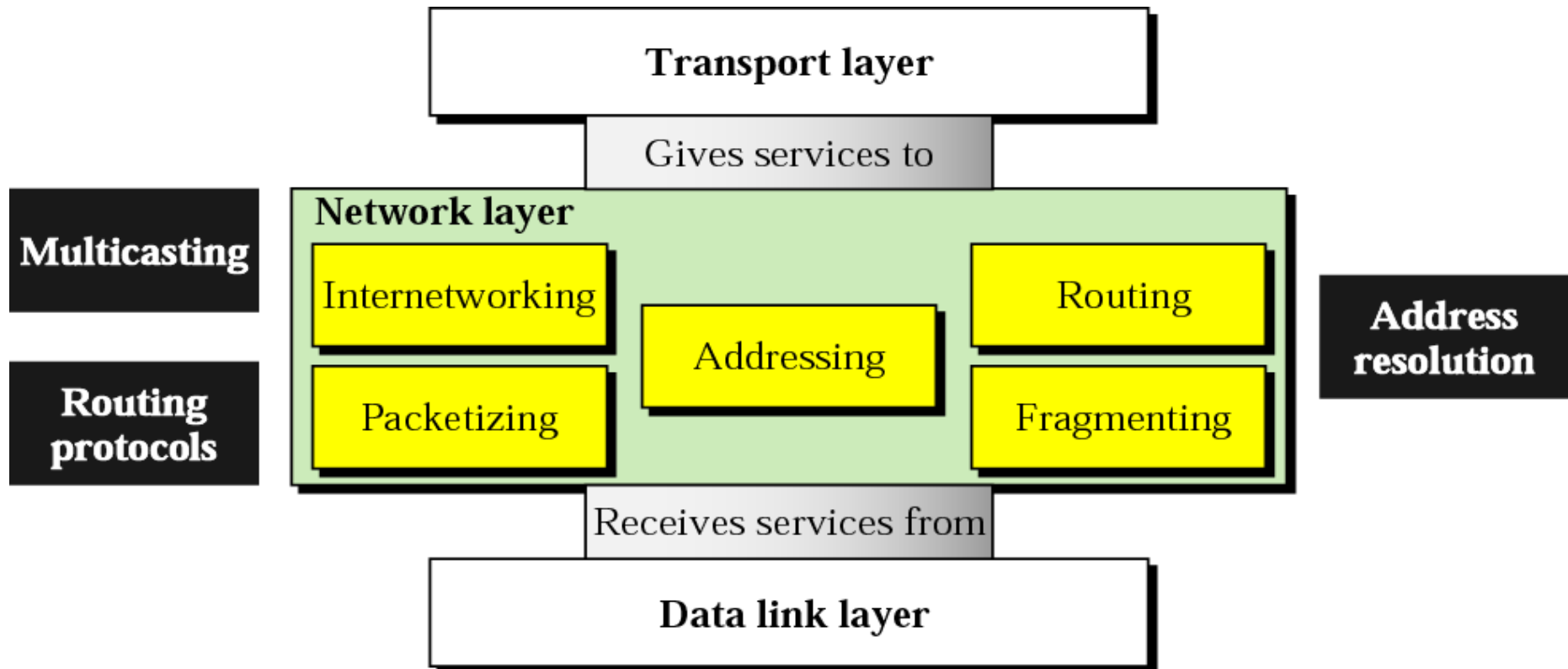
# Gateway

- A gateway is normally a computer that operates in all five layers of the Internet or seven layers of OSI model.
- A gateway takes –
  - an application message
  - reads it
  - interprets it.
- This means that it can be used as a connecting device between two internetworks that use different models.

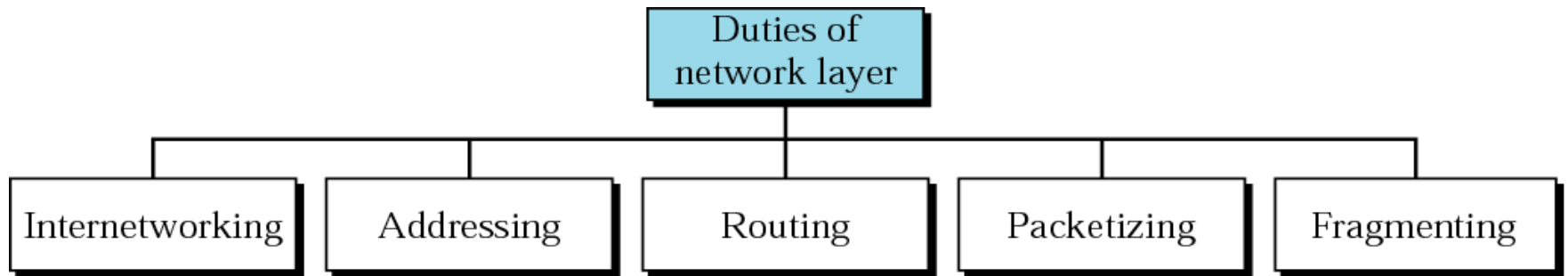


# UNIT -III

# Position of network layer



# Network layer duties



# RIP

- The routing information protocol (RIP) is an interior routing protocol used inside an autonomous system.
- It is a very simple protocol based on distance vector routing which uses the Bellman-Ford alg for calculating the routing tables.

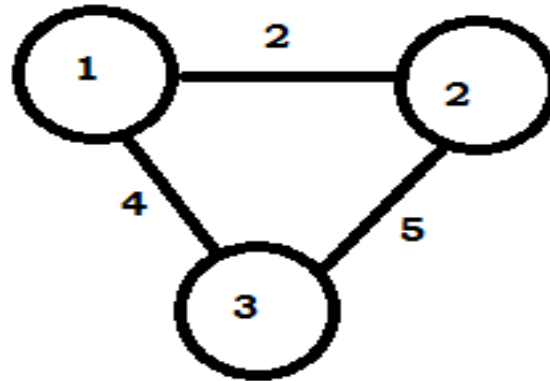
# Distance vector routing

- In distance vector routing each router periodically shares its knowledge about the entire internet with its neighbors.
- In distance vector routing, the least-cost route between any two nodes is the route with minimum distance.
- In this protocol, as the name implies, each node maintains a vector (table) of minimum distances to every node.
- The table at each node also guides the packets to the desired node by showing the next stop in the route (next-hop routing).
- We can think of nodes as the cities in an area and the lines as the roads connecting them.
- A table can show a tourist the minimum distance between cities.

# Distance vector routing alg

Node	Cost
1	0
2	2
3	4

**Routing Table**



Node	Cost
1	2
2	0
3	5

**Routing Table**

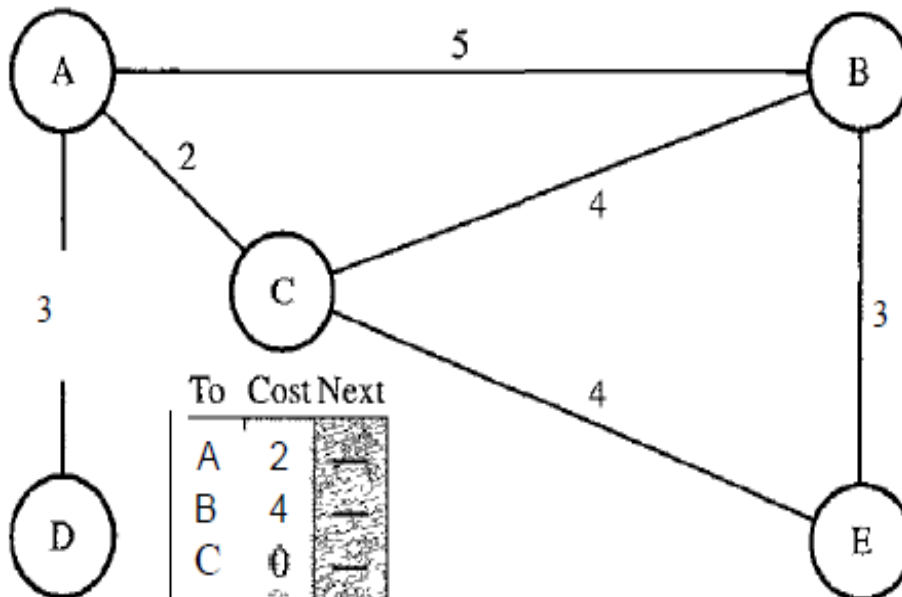
Node	Cost
1	4
2	5
3	0

**Routing Table**

# example

To	Cost	Next
A	0	—
B	5	—
C	2	—
D	3	—
E	6	C

A's table



To	Cost	Next
A	5	—
B	0	—
C	4	—
D	8	A
E	3	—

B's table

To	Cost	Next
A	3	—
B	8	A
C	5	A
D	0	—
E	9	A

D's table

To	Cost	Next
A	2	—
B	4	—
C	0	—
D	5	A
E	4	—

e's table

To	Cost	Next
A	—	—
B	—	—
C	—	—
D	—	C
E	—	—

E's table

*Congestion Control  
and  
Quality of Service*

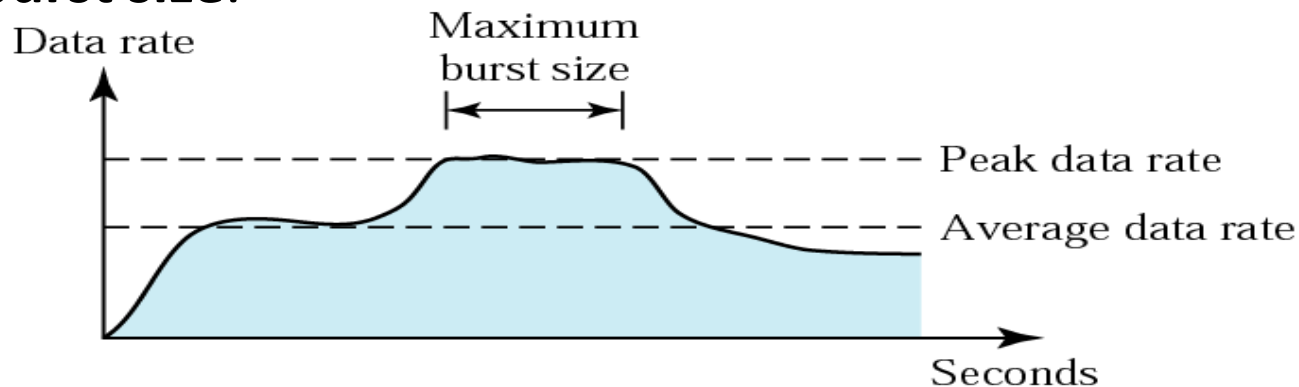


# Data Traffic

*Traffic Descriptor*

*Traffic Profiles*

- In **cc** control we try to avoid traffic congestion. in **qos** we try to create an appropriate envt for traffic.
- **Traffic descriptor** r qualitative values that represent data flow.
- **Avg bit rate** is no. of bits sent during period of time divided by no. of secs.
- **Peak data rate** defines max data rate of traffic.
- Max length of time traffic is generated at peak rate is **max burst size**.



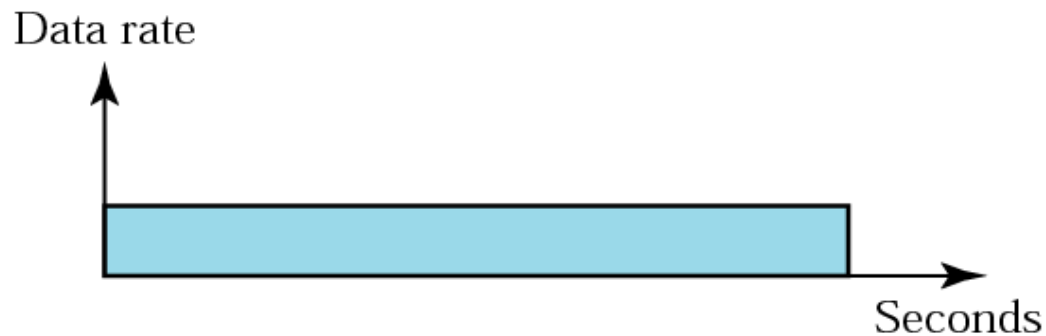
# Traffic profiles

- a data flow can have one of the following traffic profiles:
  - constant bit rate,
  - variable bit rate,
  - or bursty

# Constant-bit-rate traffic

**A constant-bit-rate (CBR),** or a fixed-rate, traffic model has a data rate that does not change.

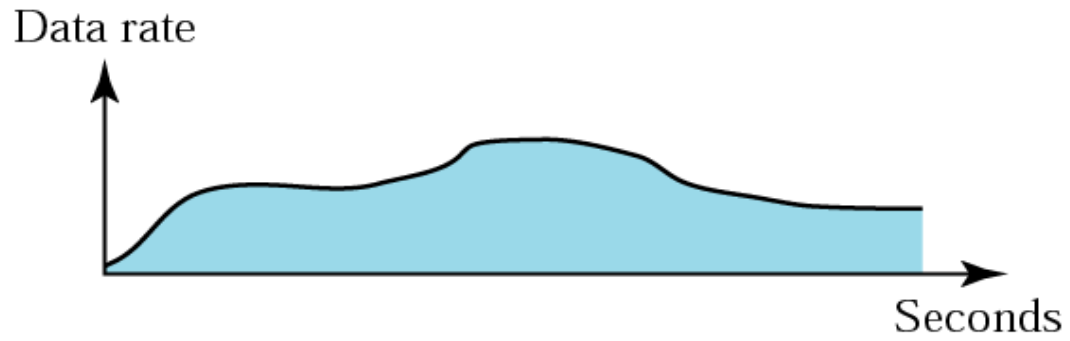
In this type of flow, the average data rate and the peak data rate are the same.



# Variable-bit-rate (VBR)

- In the variable-bit-rate (VBR) category, the rate of the data flow changes in time, with the changes smooth instead of sudden and sharp.
- In this type of flow, the average data rate and the peak data rate are different.
- The maximum burst size is usually a small value.

# Variable-bit-rate traffic



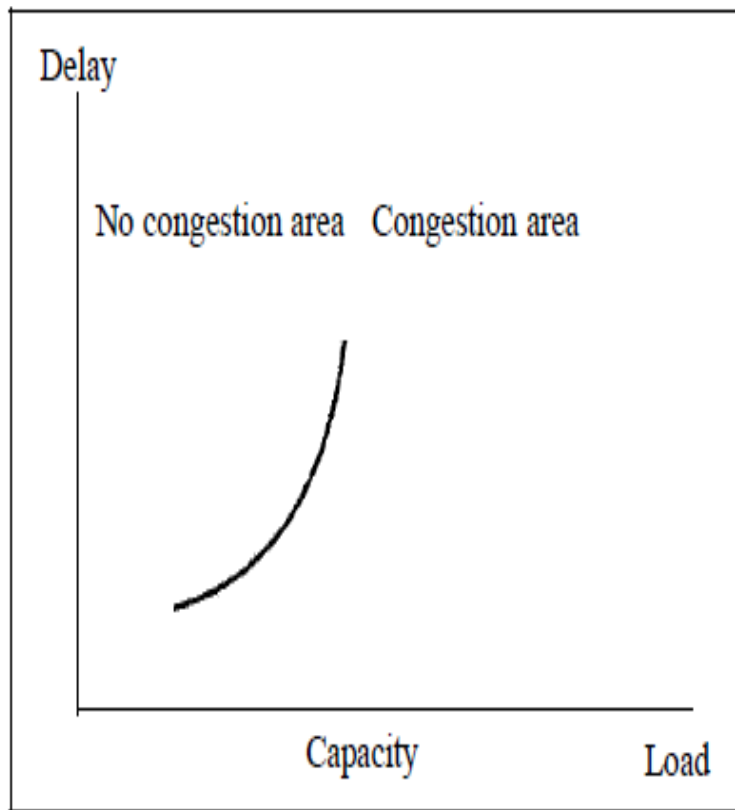
# Bursty data

- In the **bursty data category**, the data rate **changes suddenly in a very short time**.
- **It** may jump from zero, for example, to 1 Mbps in a few microseconds and vice versa.
- The average bit rate and the peak bit rate are very different values in this type of flow.
- The maximum burst size is significant. This is the most difficult type of traffic for a network to handle because the profile is very unpredictable.

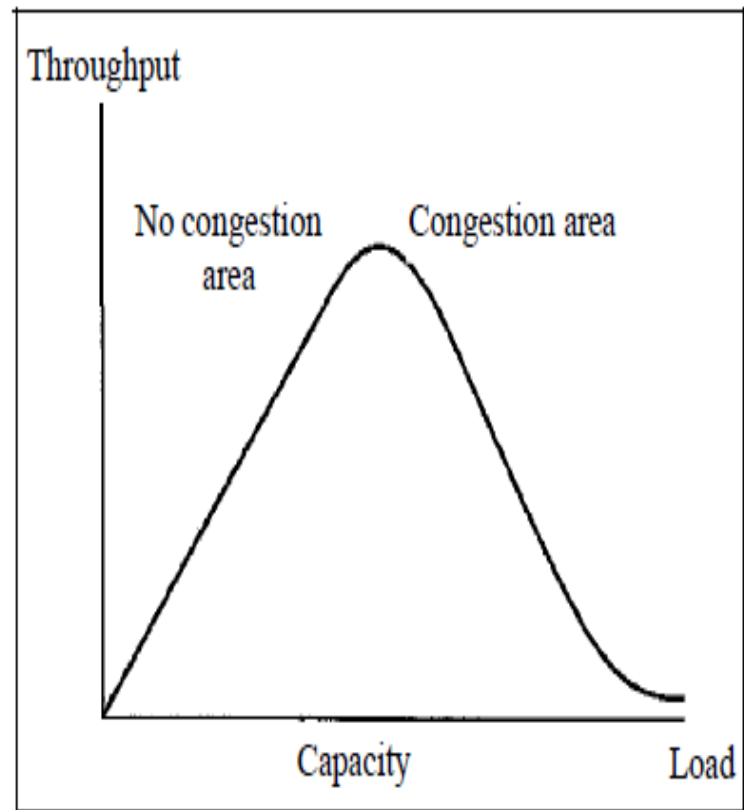
- In bursty traffic, data rate changes suddenly in a very short time.







a. Delay as a function of load



b. Throughput as a function of load

# Congestion

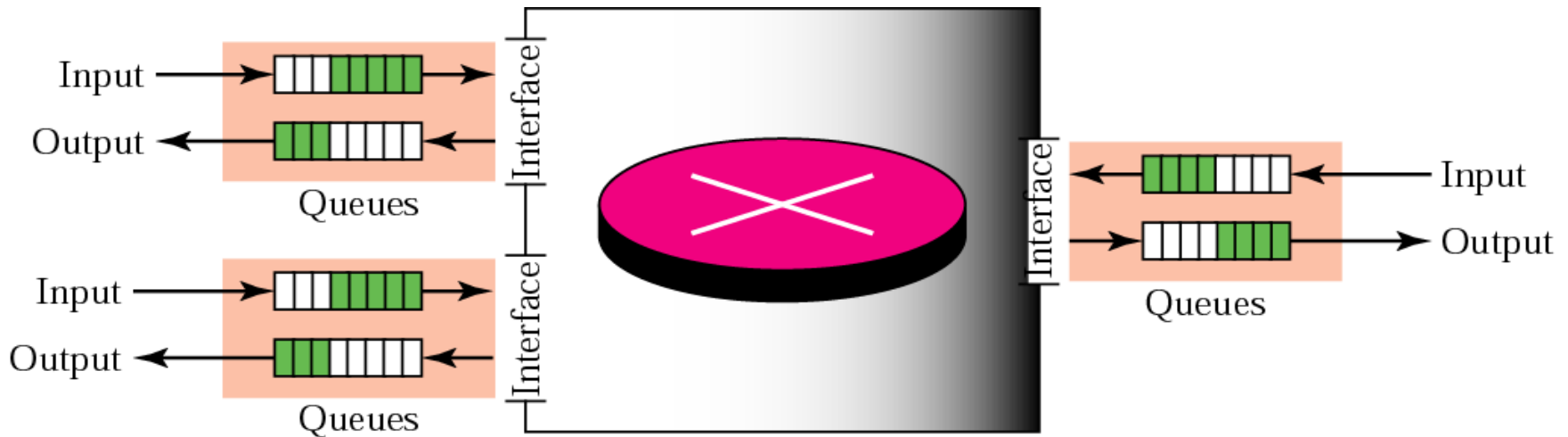
- Congestion in a network may occur if the **load on the network**
  - The number of packets sent to the network-is greater than the *capacity of the network*.
  - **capacity** : *The number of packets a network can handle.*
- **Congestion control** refers to the mechanisms and techniques to control the congestion and keep the load below the capacity.

# Congestion contd...

- Congestion in a network or internetwork occurs because routers and switches have queues-buffers that hold the packets before and after processing.
- A router, for example, has an input queue and an output queue for each interface.
- When a packet arrives at the incoming interface, it undergoes three steps before departing,

# 3 steps

- 1. The packet is put at the end of the input queue while waiting to be checked.
- 2. The processing module of the router removes the packet from the input queue once it reaches the front of the queue and uses its routing table and the destination address to find the route.
- 3. The packet is put in the appropriate output queue and waits its turn to be sent.



# Congestion

## *Network Performance*

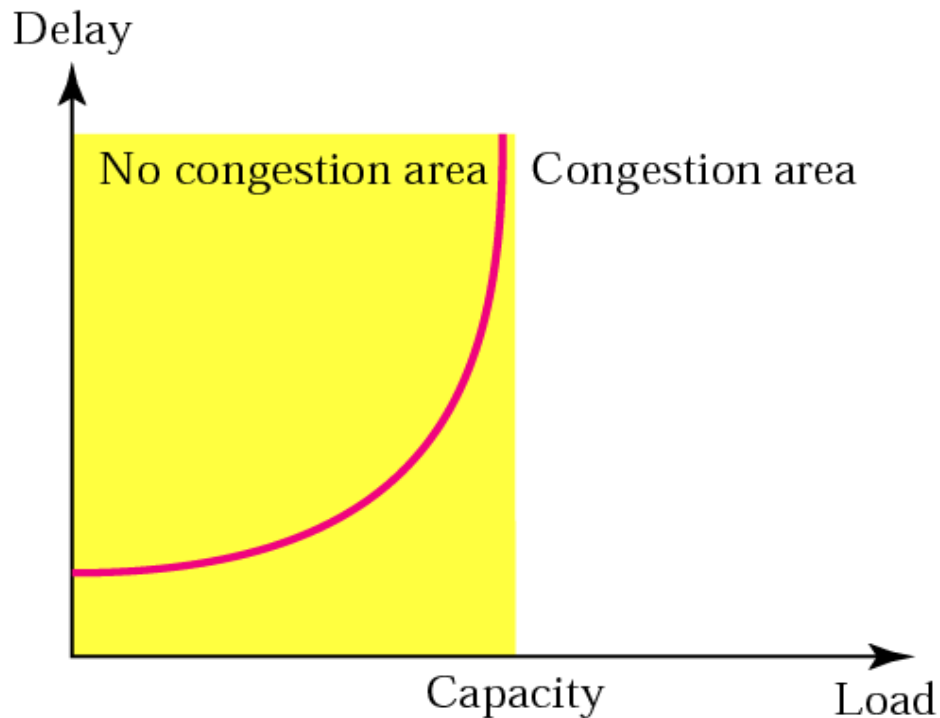
Congestion control involves two factors that measure the performance of a network:

>*delay*

>*throughput*

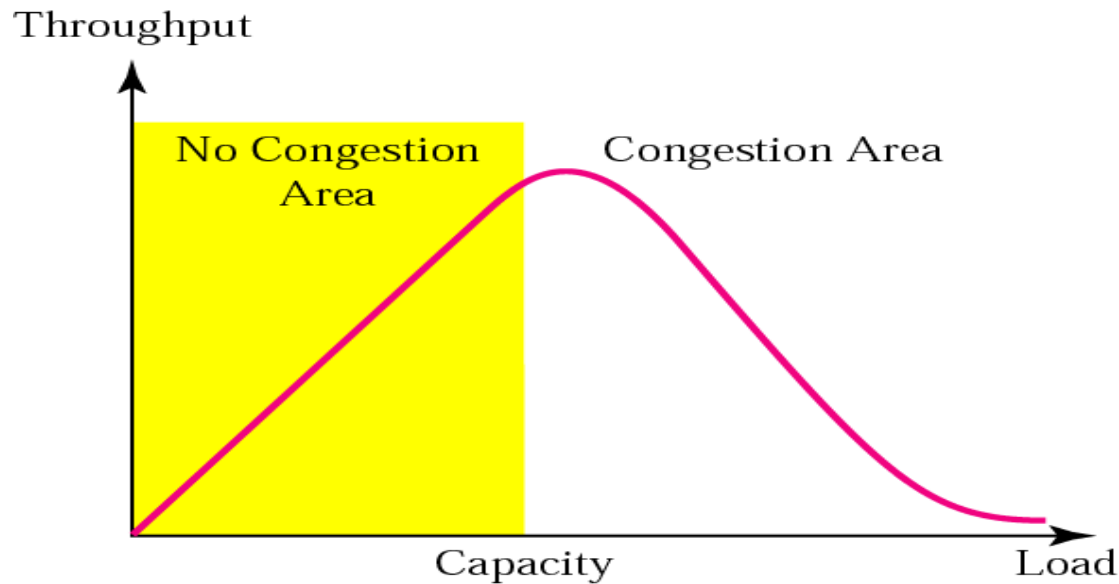
The number of packets passing through the network in a unit of time is called as ***Network Performance***

- CC involves 2 factors that measure the performance of n/w -delay & throughput.



## Throughput versus network load

- Throughput is no. of packets passing through n/w in unit of time.



# Congestion Control

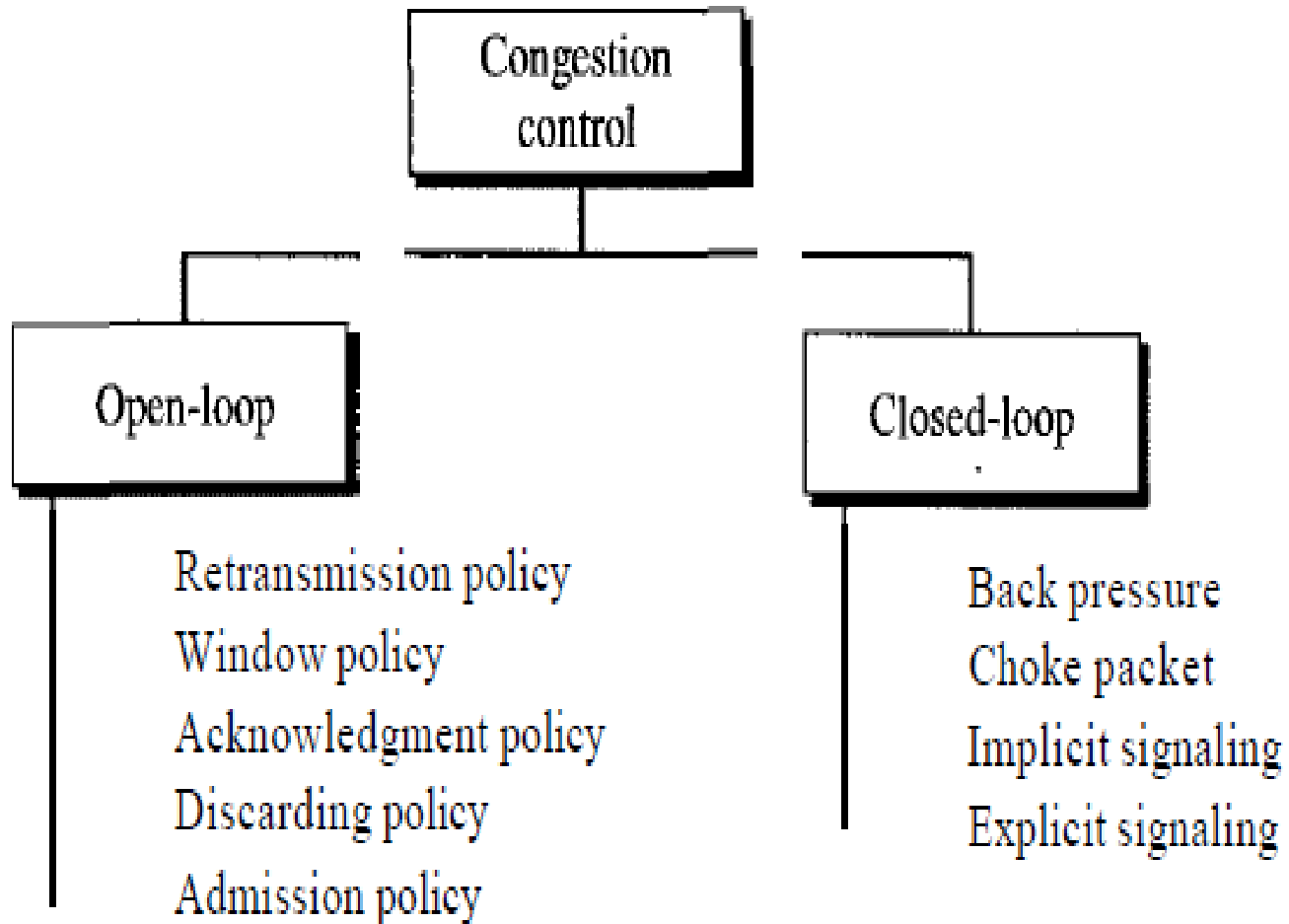
*Open Loop*

*Closed Loop*



# Congestion control

- Congestion control refers to techniques and mechanisms that can either prevent congestion, before it happens, or remove congestion, after it has happened.
- In general, we can divide congestion control mechanisms into two broad categories:
  - **open-loop congestion control (prevention)**
  - **and closed-loop congestion control (removal)**



# Open-loop Congestion Control

- In open-loop congestion control, policies are applied to prevent congestion before it happens.
- In these mechanisms, congestion control is handled by either the source or the destination.

# *Retransmission Policy*

- If the sender feels that a sent packet is lost or corrupted, the packet needs to be retransmitted.
- Retransmission in general may increase congestion in the network.

# *Window Policy*

- The type of window at the sender may also affect congestion.
- The Selective Repeat window is better than the Go-Back-N window for congestion control.
- In the *Go-Back-N* window, when the timer for a packet times out, several packets may be resent
- The Selective Repeat window, on the other hand, tries to send the specific packets that have been lost or corrupted.

## ***Acknowledgment Policy***

- The acknowledgment policy imposed by the receiver may also affect congestion.
- If the receiver does not acknowledge every packet it receives, it may slow down the sender and help prevent congestion.

## ***Discarding Policy***

- A good discarding policy by the routers may prevent congestion and at the same time may not harm the integrity of the transmission.

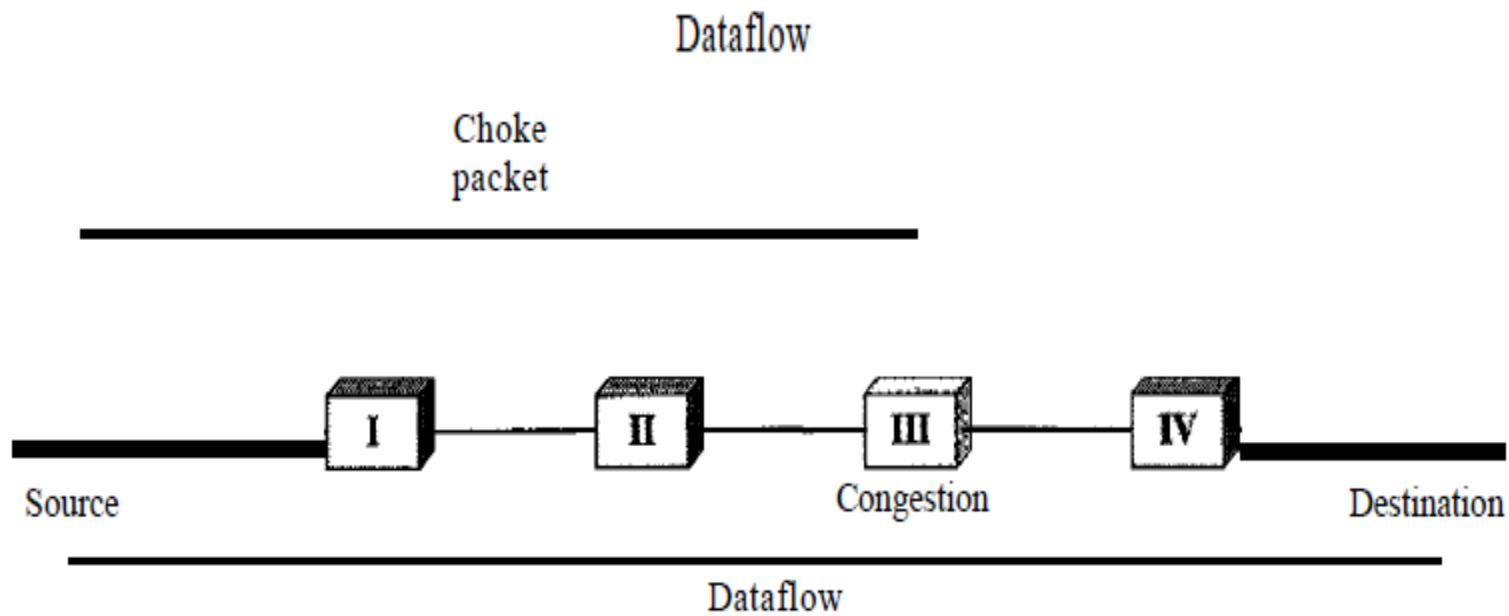
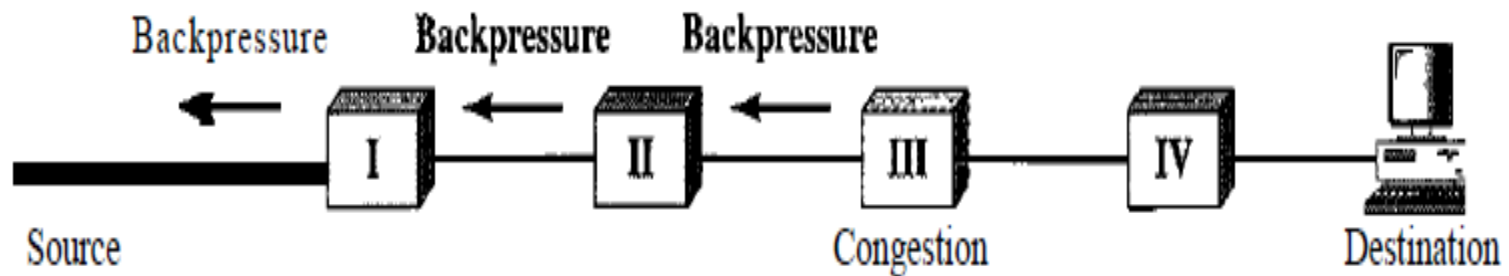
# ***Admission Policy***

- An admission policy, which is a quality-of-service mechanism, can also prevent congestion in virtual-circuit networks.
- Switches in a flow first check the resource requirement of a flow before admitting it to the network.
- A router can deny establishing a virtual circuit connection if there is congestion in the network or if there is a possibility of future congestion.

# Closed loop

- Closed-loop congestion control mechanisms try to alleviate congestion after it happens.
- **back pressure** ->in which congested node stops recvng data frm immediate upstream
- **Choke packet**:is a packet sent by a node to source to inform it of congestion.
- **Implicit signaling**:no communication btw source and congested nodes.source guesses that there is congestion somewhere in n/w.
- **Explicit signaling**:source that experiences congestion can explicitly send a signal to source r destn.





# Two Examples

*Congestion Control in TCP*

*Congestion Control in Frame Relay*



*TCP assumes that the cause of a lost segment is due to congestion in the network.*



*If the cause of the lost segment is congestion, retransmission of the segment does not remove the cause—it aggravates it.*

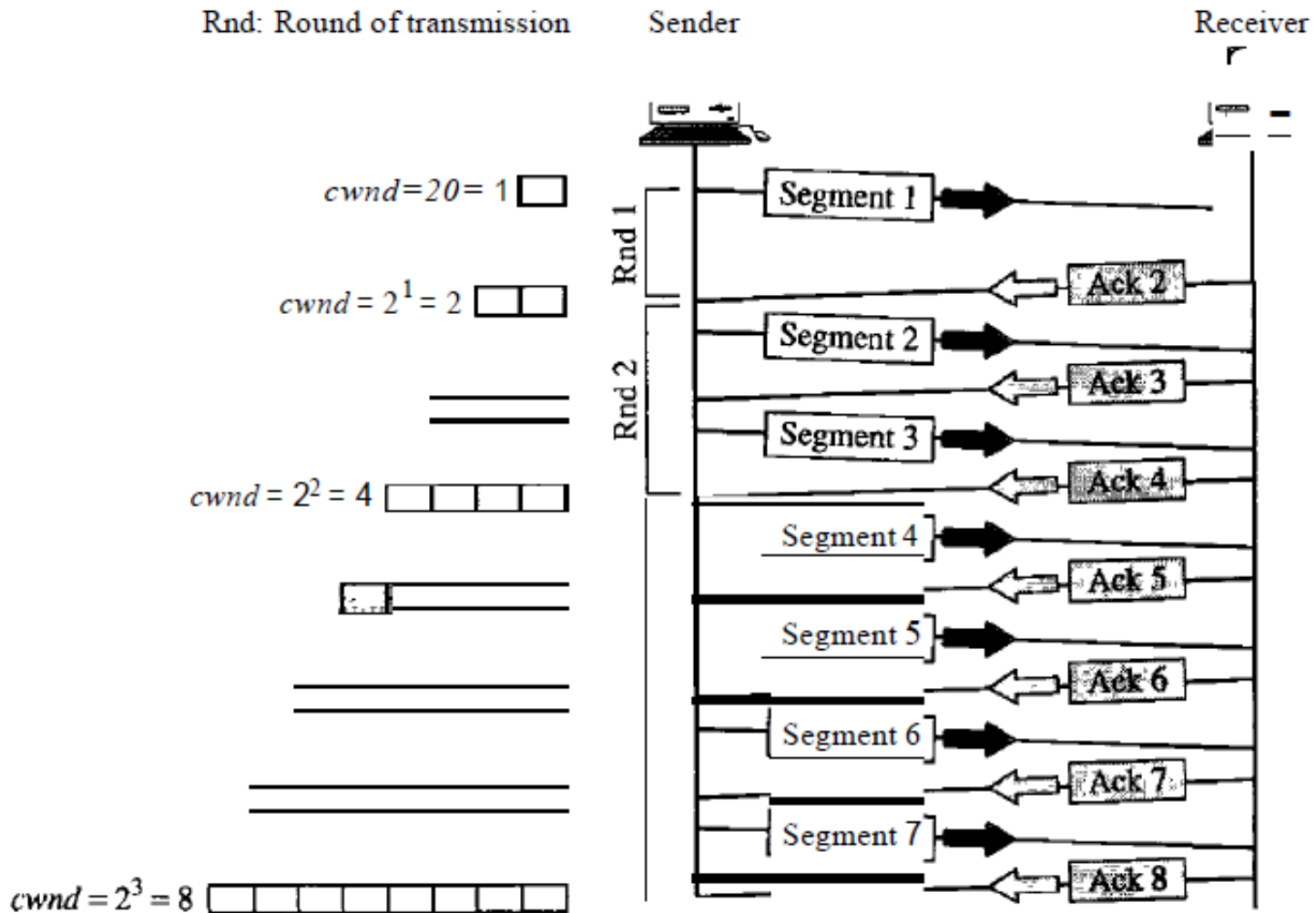
# Congestion Window

- The sender's window size is determined not only by the receiver but also by congestion in the network.
- The sender has two pieces of information:
  - The receiver-advertised window size and
  - The congestion window size.
- The actual size of the window is the minimum of these two.
- **Actual window size = minimum (rwnd, cwnd)**

# Congestion Avoidance

- **In tcp it uses 2 policies**
  - slow start and additive increase
  - multiplicative decrease
- **1)slow start:exponential increase:** alg is based on idea that size of congestion window starts with 1 max segment size(MSS).
- MSS is determined during connection establishment.
- size of window increases 1 MSS each time ack is sent.
- window starts slowly and grows exponentially.
- **additive increase:**to avoid congestion before it happens,one must slow down its exponential growth.
- when size of congestion window reaches slow start threshold,slow start phase stops and additive phase begins.
- In this algorithm, each time the whole window of segments is acknowledged (one round), the size of the congestion window is increased by 1
- **2)multiplicative decrease:**if congestion occurs,congestion window size must be decreased.size of threshold must be dropped to one half.

# Slow start, exponential increase

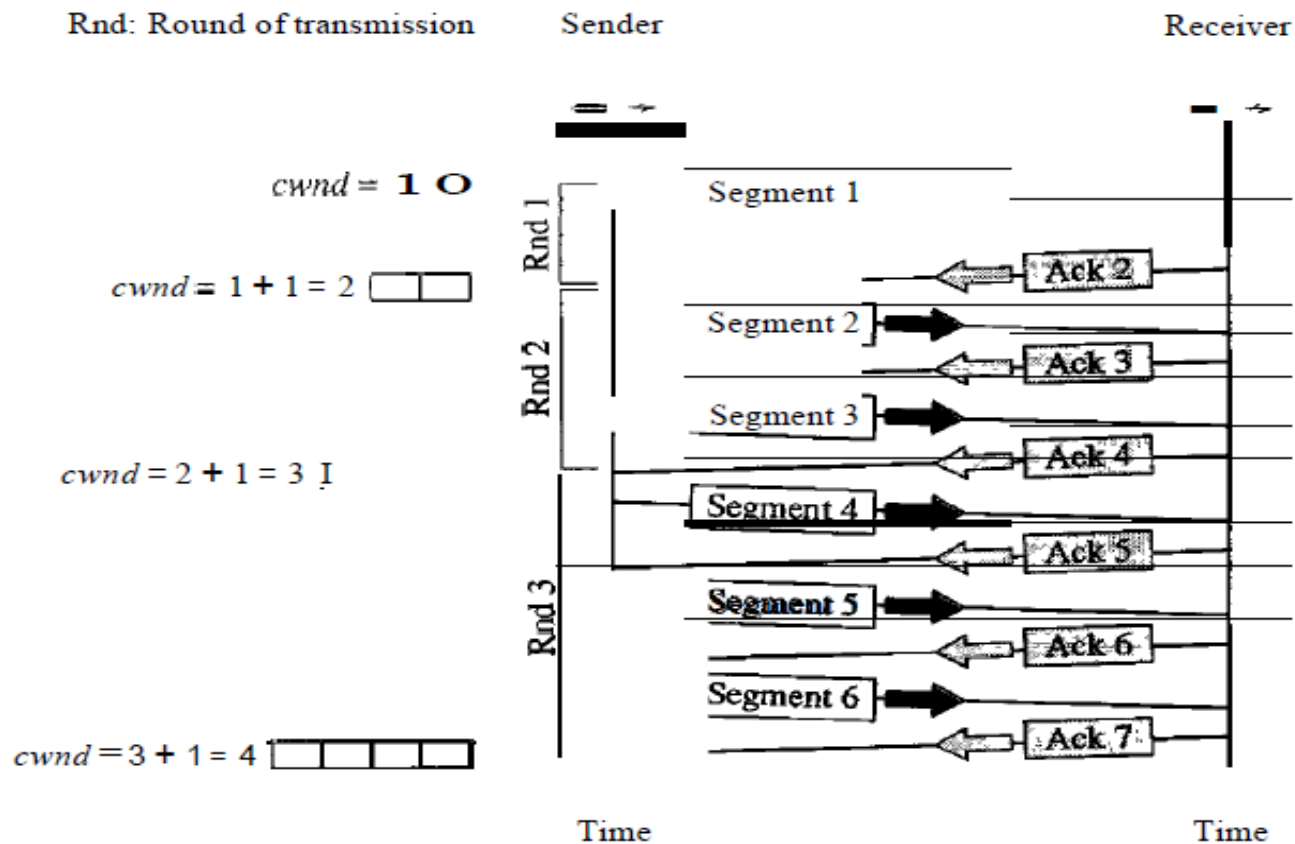


If we look at the size of *cwnd* in terms of rounds we find that the rate is exponential as shown below:

- Start .....  $cwnd=1$
- After round 1 .....  $cwnd=2^1 = 2$
- After round 2 .....  $cwnd=2^2 = 4$
- After round 3 .....  $cwnd=2^3 = 8$



# Congestion avoidance, additive increase



this case, after the sender has received acknowledgments for a complete window

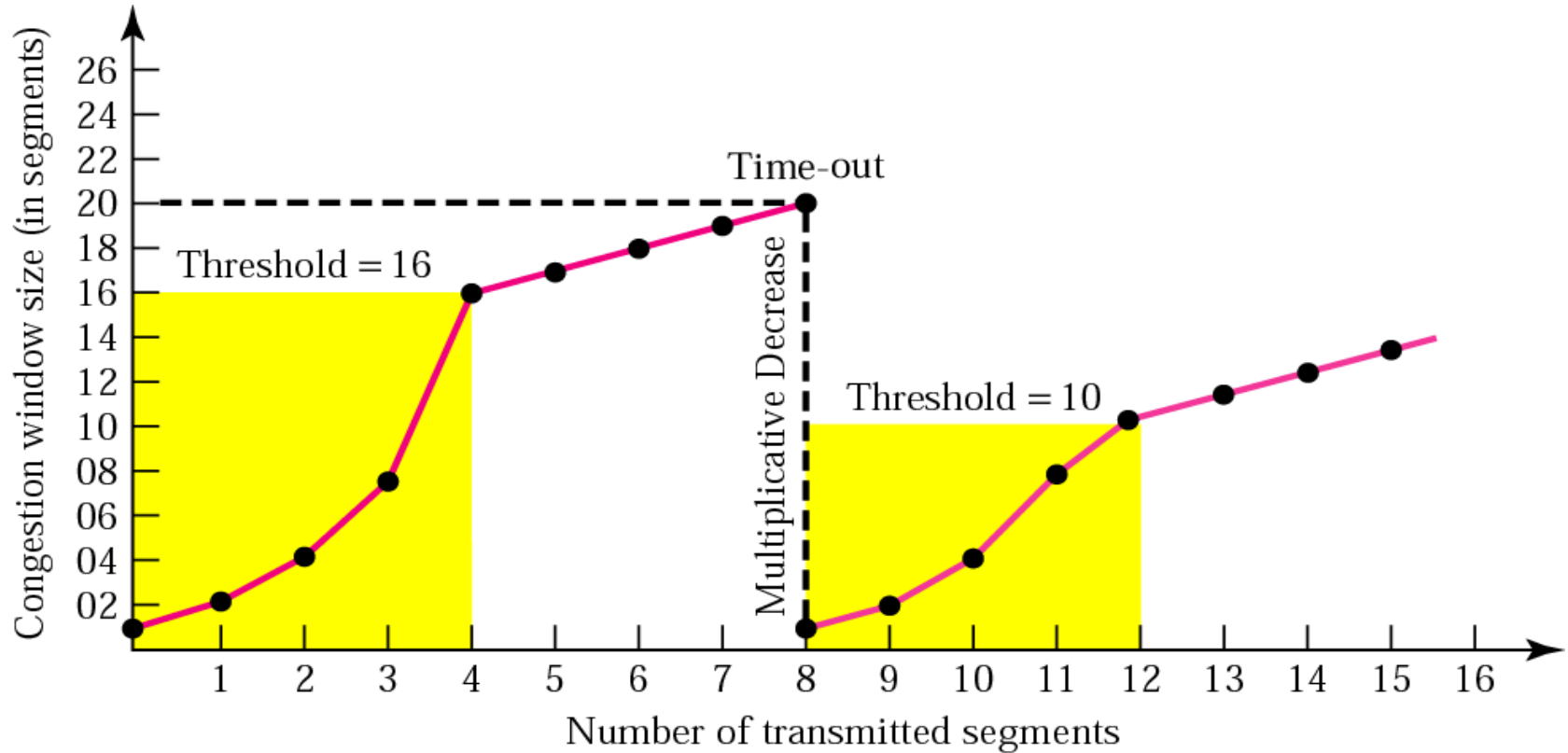
If we look at the size of *cwnd* in terms of rounds, we find that the rate is additive as shown below:

- Start
- After round 1
- After round 2
- After round 3
- .....
- $cwnd=1$
- $cwnd=1+1=2$
- $cwnd=2+1=3$
- $cwnd=3+1=4$

# Multiplicative Decrease

- If congestion occurs, the congestion window size must be decreased.
- The only way the sender can guess that congestion has occurred is by the need to retransmit a segment.
- The strategy says if a time-out occurs, the threshold must be set to one-half of the last congestion window size, and the congestion window size from 1 again.

# Multiplicative decrease



# Congestion Control in frame relay

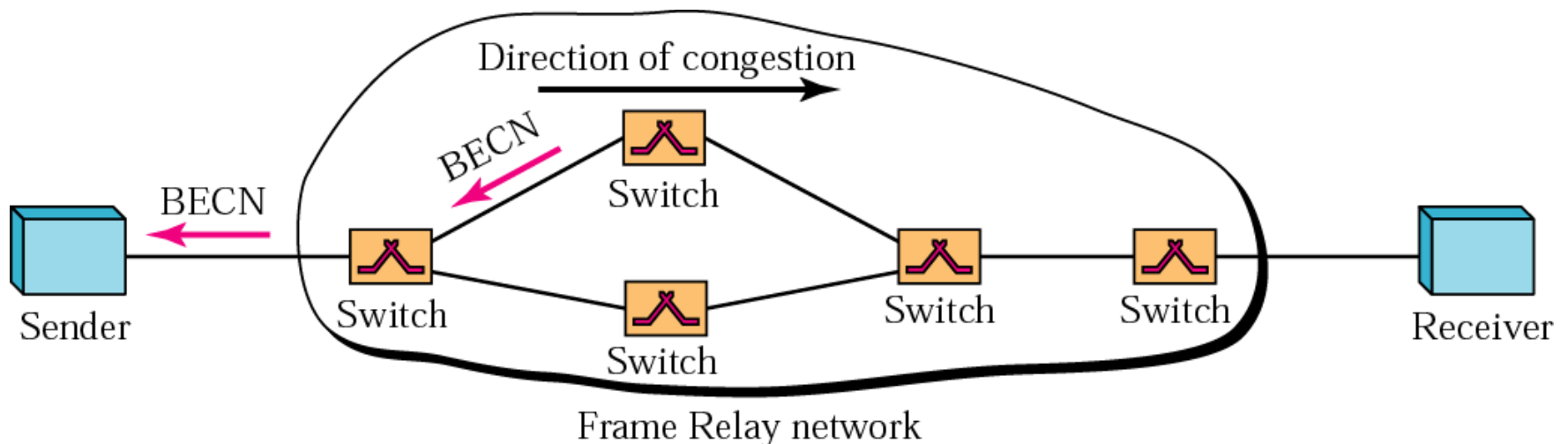
- Congestion in a Frame Relay network decreases throughput and increases delay.
- A high throughput and low delay are the main goals of the Frame Relay protocol.
- Frame Relay does not have flow control.

## **Congestion Avoidance**

- For congestion avoidance, the Frame Relay protocol uses 2 bits in the frame to explicitly warn the source and the destination of the presence of congestion.
- **BECN**
- **FECN**

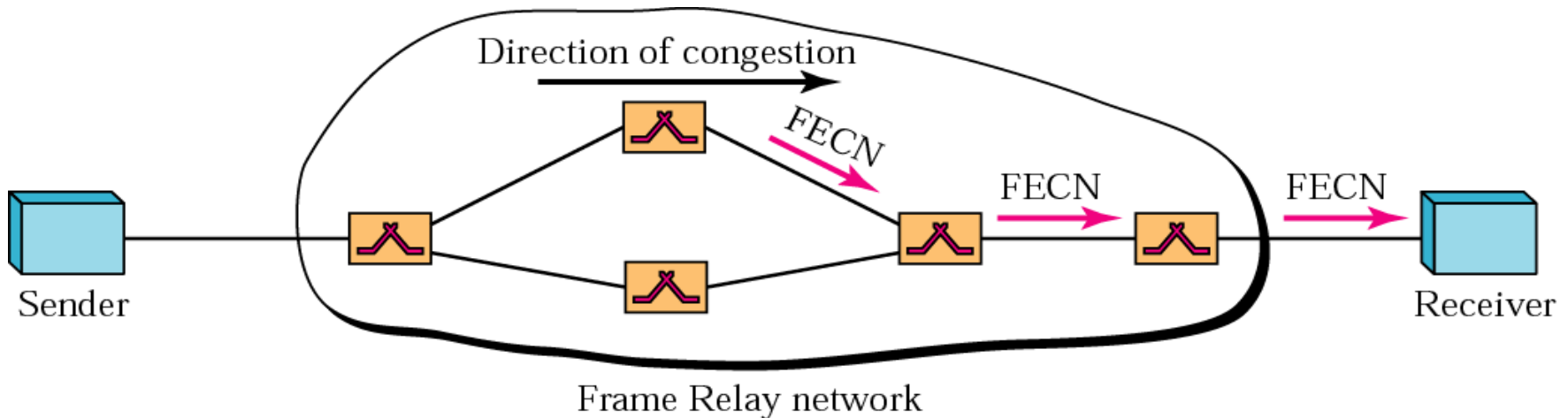
## Backward explicit congestion notification(BECCN):

- A bit warns sender of congestion in n/w.
- switch uses response frames from receiver or switch can use predefined connection(DLCL) to send special frames for this specific purpose.
- The sender can respond to this warning by simply reducing the data rate.

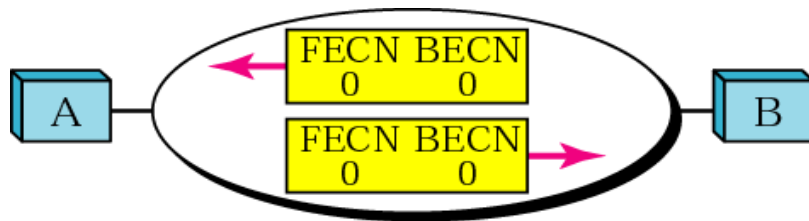


# Forward explicit congestion notification(FECN)

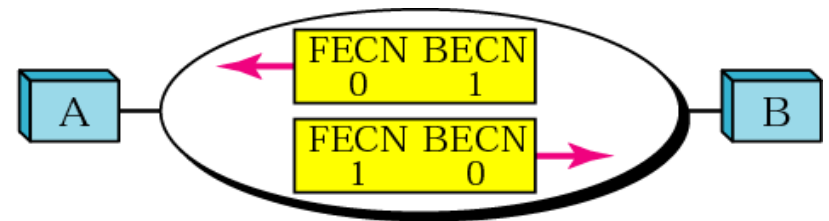
- A Bit is used to warn receiver of congestion in n/w.
- The receiver can respond to this warning by simply can delay the acknowledge thus forcing the sender to slow down rate.



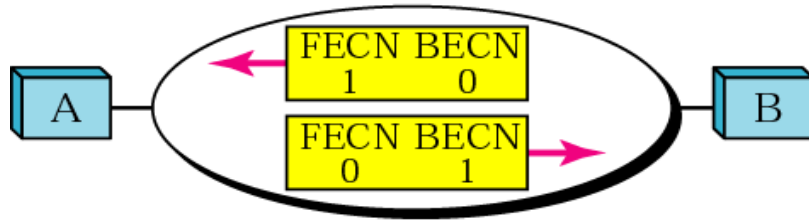
# Four cases of congestion



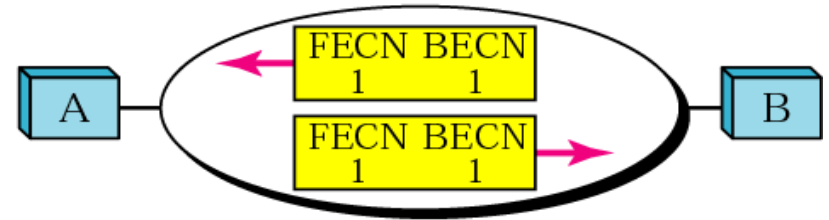
a. No congestion



b. Congestion in the direction A-B



c. Congestion in the direction B-A



d. Congestion in both directions



# Quality of Service

*Flow Characteristics*

*Flow Classes*

# QUALITY OF SERVICE

- four types of characteristics are attributed to a flow:
  - Reliability
  - delay,
  - Jitter
  - bandwidth

# Reliability

- Reliability is a characteristic that a flow needs.
- Lack of reliability means losing a packet or acknowledgment, which entails retransmission.
- However, the sensitivity of application programs to reliability is not the same.
- For example, it is more important that electronic mail, file transfer, and Internet access have reliable transmissions than telephony or audio conferencing

# delay

- Source-to-destination delay is another flow characteristic.
- Again applications can tolerate delay in different degrees.
- In this case, telephony, audio conferencing, video conferencing, and remote log-in need minimum delay, while delay in file transfer or e-mail is less important

# *Jitter*

- Jitter is the variation in delay for packets belonging to the same flow.
- For example, if four packets depart at times 0, 1, 2, 3 and arrive at 20, 21, 22, 23, all have the same delay, 20 units of time.
- On the other hand, if the above four packets arrive at 21, 23, 21, and 28, they will have different delays: 21, 22, 19, and 24.

# *Bandwidth*

- Different applications need different bandwidths.
- For In video conferencing need more bandwidth
- in an e-mail it may need less bandwidth

# Techniques to Improve QoS

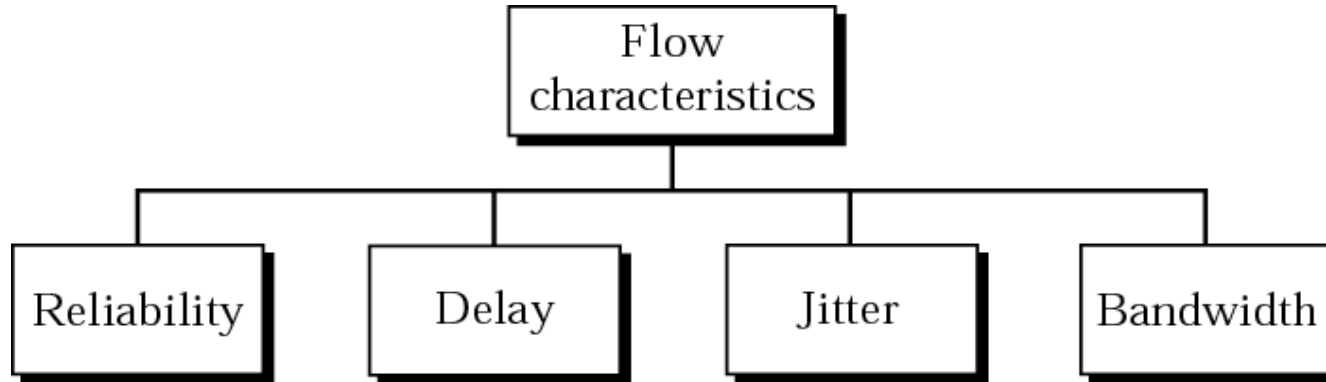
*Scheduling*

*Traffic Shaping*

*Resource Reservation*

*Admission Control*

# Flow characteristics



- **reliability** Lack of reliability means losing a packet or acknowledgment, which entails retransmission.
- **delay** Applications can tolerate delay in different degrees.
- In this case, telephony, audio conferencing, video conferencing, and remote log-in need minimum delay.
- while delay in file transfer or e-mail is less important.



# Jitter

- **Jitter** is the variation in delay for packets belonging to the same flow.
- For example, if four packets depart at times 0, 1, 2, 3 and arrive at 20, 21, 22, 23, all have the same delay, 20 units of time.
- On the other hand, if the above four packets arrive at 21, 23, 21, and 28, they will have different delays: 21, 22, 19, and 24.
- For applications such as audio and video, the first case is completely acceptable; the second case is not.
- For these applications, it does not matter if the packets arrive with a short or long delay as long as the delay is the same for all packets. For this application, the second case is not acceptable.
- Jitter is defined as the variation in the packet delay. High jitter means the difference between delays is large; low jitter means the variation is small.

# *Bandwidth*

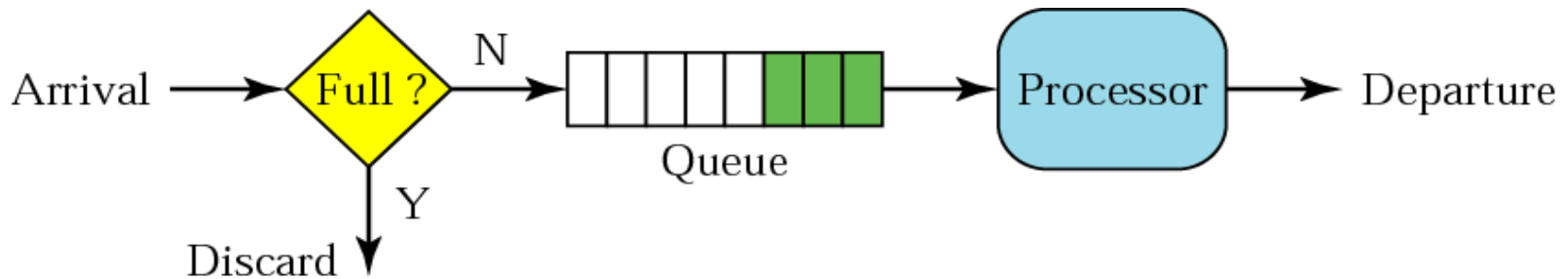
- Different applications need different bandwidths.
- In video conferencing we need to send millions of bits per second to refresh a color screen so it needs more bandwidth
- while the total number of bits in an e-mail may not reach even a million so it needs less bandwidth

# TECHNIQUES TO IMPROVE QoS

- four common methods to improve QoS
  - scheduling,
  - traffic shaping,
  - admission control,
  - resource reservation.

# FIFO queue

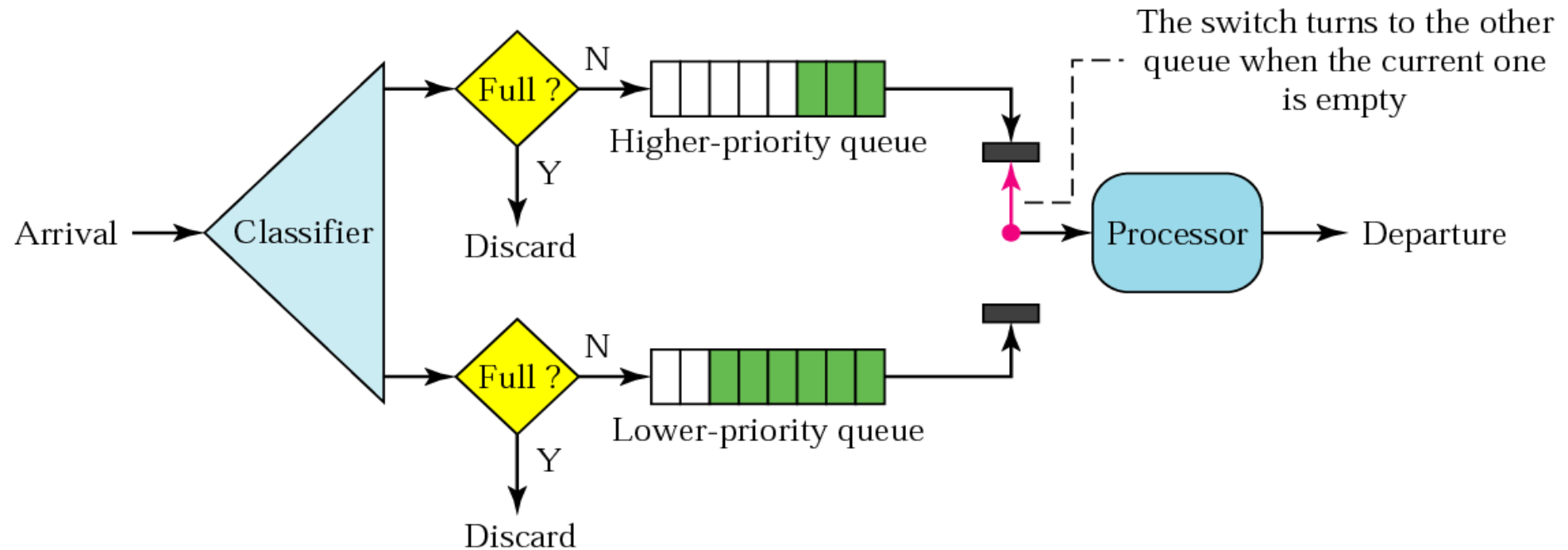
- Several scheduling techniques are designed to improve QoS.
- FIFO queing:in FIFO queue,packets wait in a buffer until node is ready to process them



# Priority Queuing

- In priority queuing, packets are first assigned to a priority class.
- Each priority class has its own queue.
- The packets in the highest-priority queue are processed first.
- Packets in the lowest-priority queue are processed last.

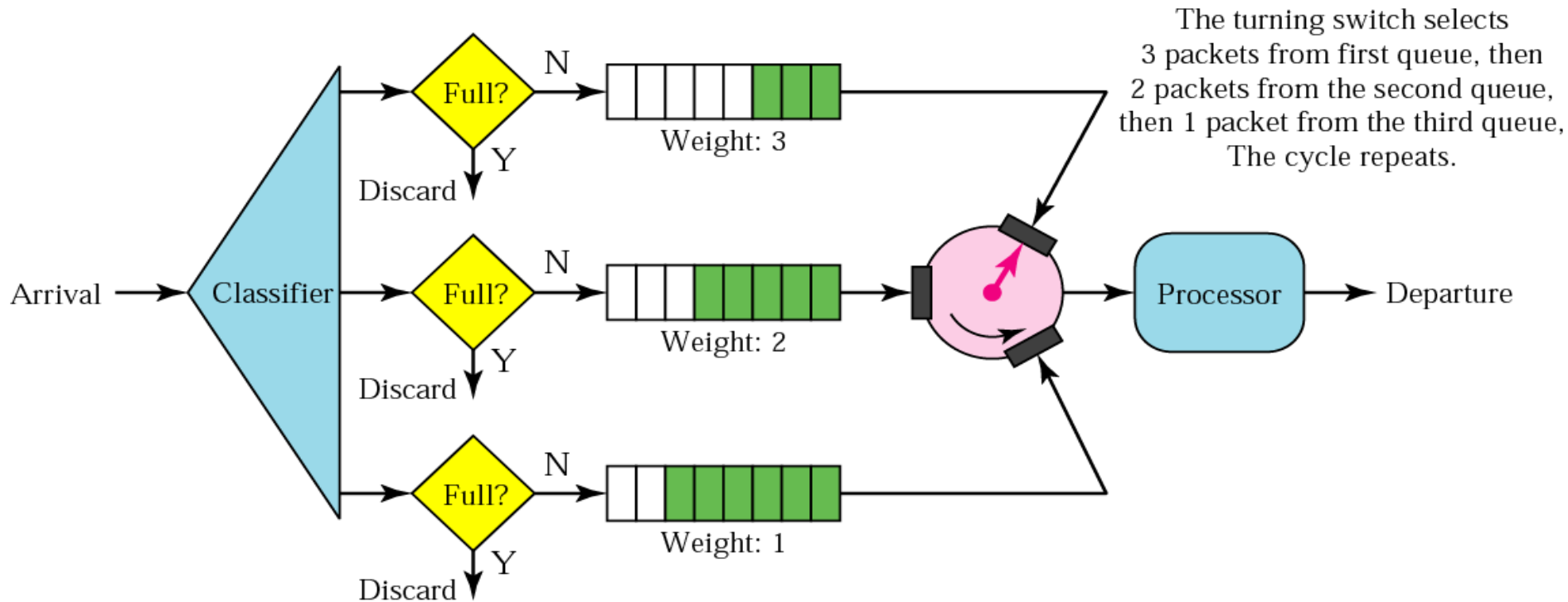
# Priority queuing



# Weighted fair Queuing

- In this technique, the packets are still 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.
- The system processes packets in each queue in a round-robin fashion with the number of packets selected from each queue based on the corresponding weight.
- For example, if the weights are 3, 2, and 1, three packets are processed from the first queue, two from the second queue, and one from the third queue.

# Weighted fair queuing

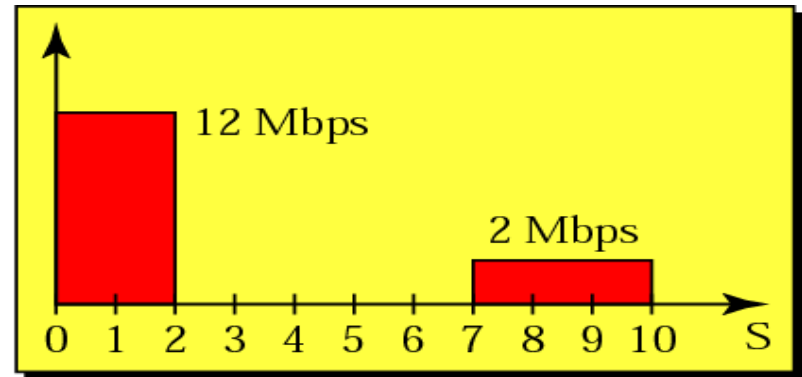
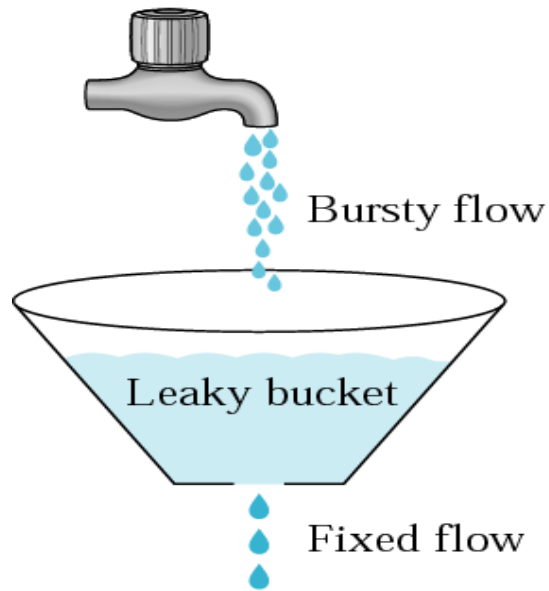




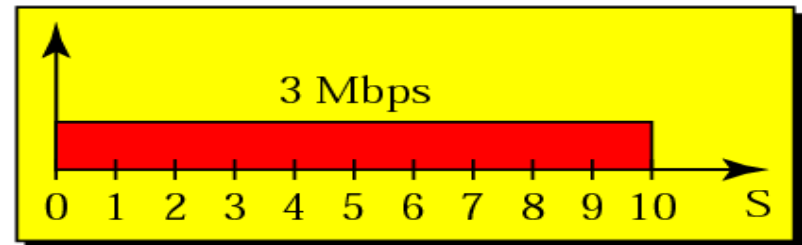
# Traffic Shaping

- Traffic shaping is a mechanism to control the amount and the rate of the traffic sent to the network.
- Two techniques can shape traffic:
  - leaky bucket and
  - token bucket.

# Leaky bucket



Bursty data



Fixed-rate data

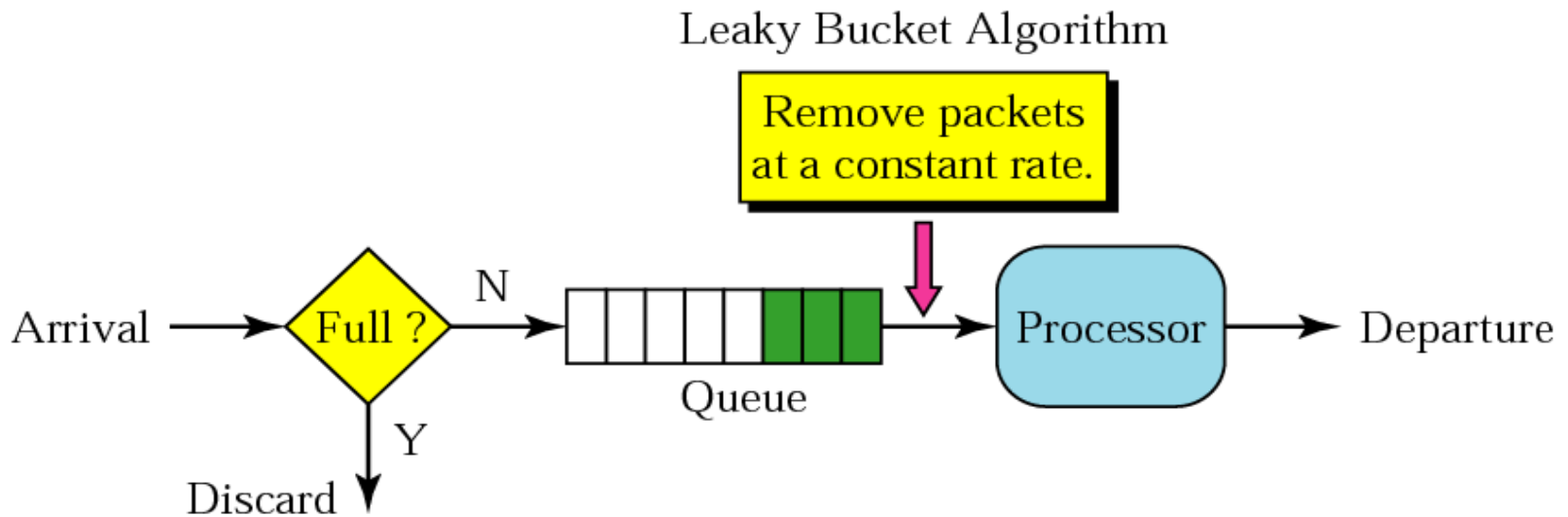
In all, the host has sent 30 Mbits of data in 10s.

The leaky bucket smooths the traffic by sending out data at a rate of 3 Mbps during the same 10 s.

# leaky bucket implementation

- A FIFO queue holds the packets. If the traffic consists of **fixed-size packets** (e.g., cells in ATM networks), the process removes a fixed number of packets from the queue at each tick of the clock.
- If the traffic consists of **variable-length** packets, the fixed output rate must be based on the number of bytes or bits.
- **The following is an algorithm for variable-length packets:**
  1. Initialize a counter to ***n*** *at the tick of the clock.*
  2. If ***n*** *is greater than the size of the packet, send the packet and decrement the counter by the packet size. Repeat this step until ***n*** is smaller than the packet size.*
  3. Reset the counter and go to step 1.

# Leaky bucket implementation



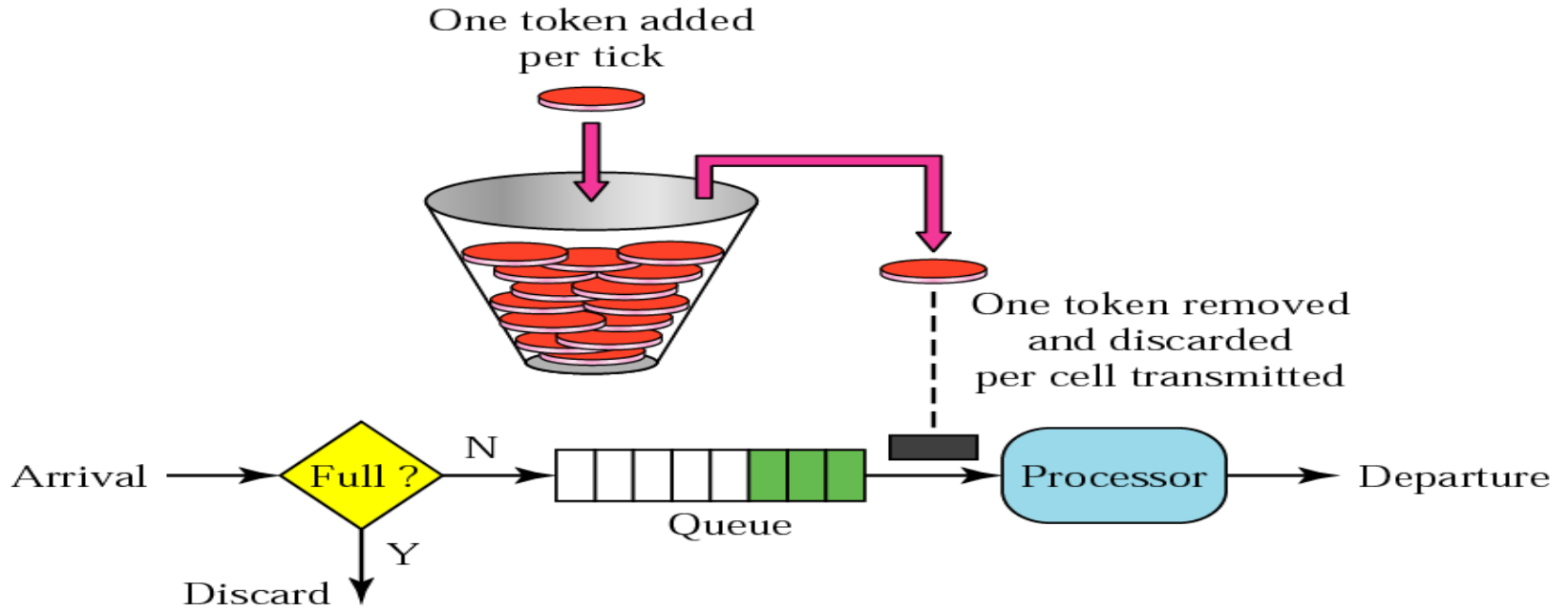


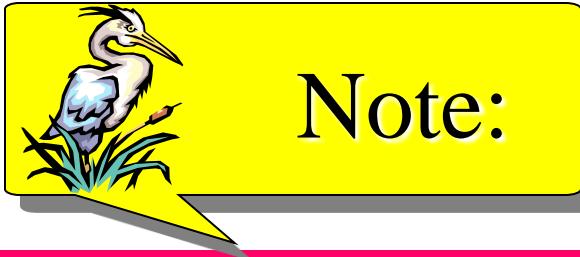
*A leaky bucket algorithm shapes bursty traffic into fixed-rate traffic by averaging the data rate. It may drop the packets if the bucket is full.*

# Token bucket

- Leaky bucket is very restrictive.
- Doesn't credit idle host.
- token bucket alg allows idle hosts to accumulate credit for future in the form of tokens.
- For each tick of clock, system sends  $n$  tokens to bucket.
- System removes 1 token for every cell of data sent.
- **Implemented** by using counter-initialized to 0 & incremented by 1 each time token added
- Each time a unit of data is sent, the counter is decremented by 1.
- When the counter is zero, the host cannot send data.

# Token bucket





Note:

*The token bucket allows bursty traffic  
at a regulated maximum rate.*



# Integrated Services

*Signaling*

*Flow Specification*

*Admission*

*Service Classes*

*RSVP*



*Integrated Services is a flow-based QoS model designed for IP.*

*Which means that a user needs to create a flow, a kind of virtual circuit, from the source to the destination and inform all routers of the resource requirements*

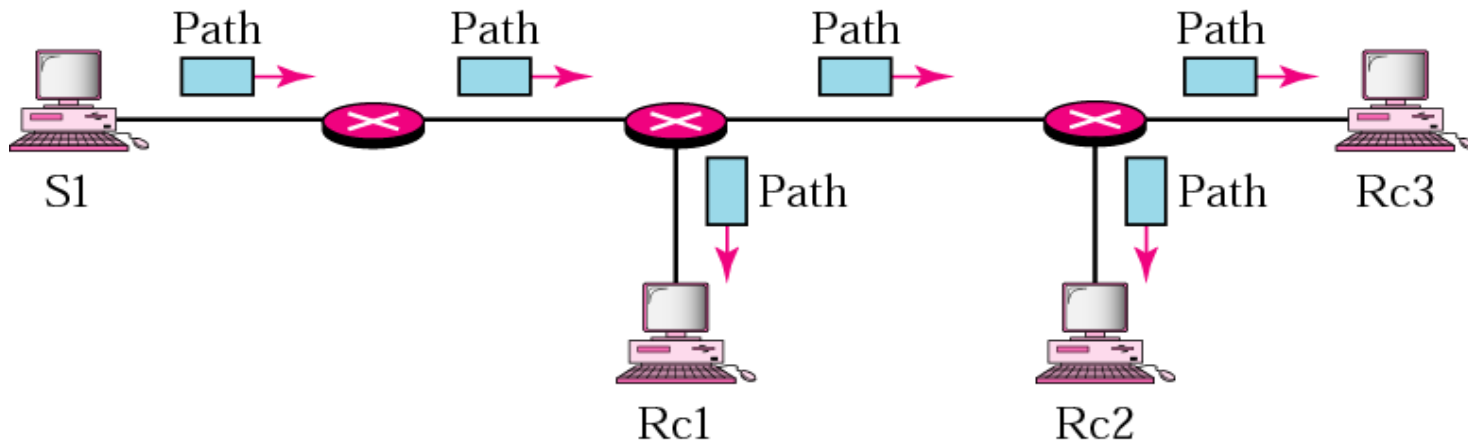
# Integrated Services

- How can we implement a flow based model over connectionless protocol?
- Solution is signaling protocol to run over IP that provides signaling mechanism for making reservation.
- This protocol resource reservation protocol.
- **Flow specification** has 2 parts-
  - rspec(resource specification) → buffer, bandwidth
  - tspec(traffic specification) → traffic characterization
- **Admission**: After router rcvs flow specn frm appn it decides to admit or deny service.
- **Service classes** : **guaranteed service** is designed for realtime traffic that needs minimum end to end delay
- **Controlled load service access**: designed for appns that can accept some delays, but they are sensitive to overloaded n/w.ex:file transfer,email,internet access.

# RSVP

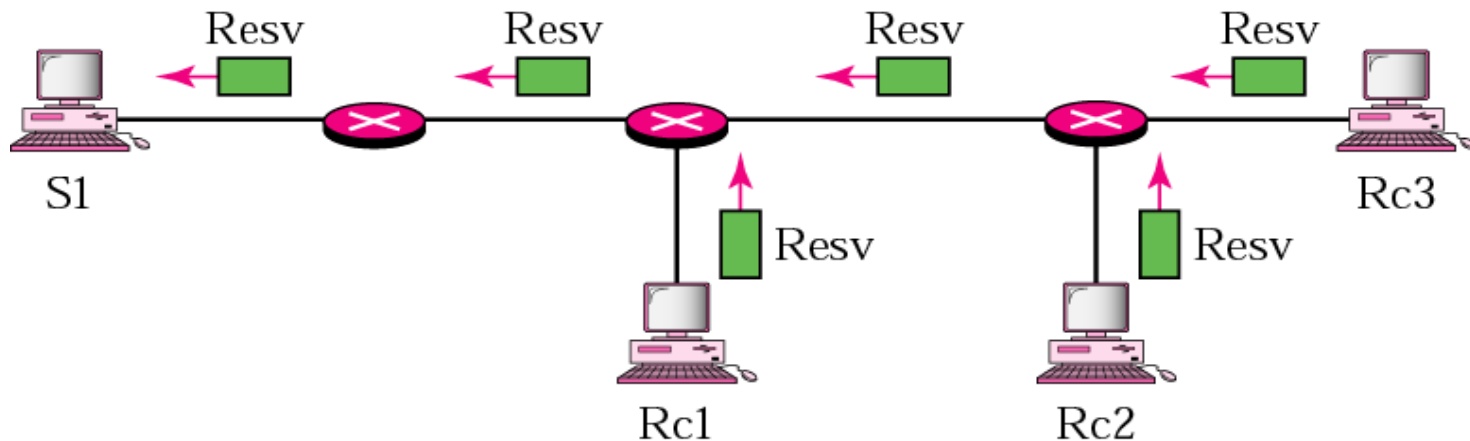
- RSVP is a signaling protocol to help IP to create a flow and consequently make resource reservation.
- Multicast Trees
  - designed for multicasting
  - also used for unicasting because unicasting is just a special case of multicasting with only one member in the multicast group
- *Receiver-Based Reservation:*
- Rsvp messages :path &rsvp
- Path msgs:rcvrs in flow make reservation in rsvp.but they don't know the path traveled by packets bfore reservation is made.
- path is needed for reservation .to solve it rsvp uses path msgs

# RSVP



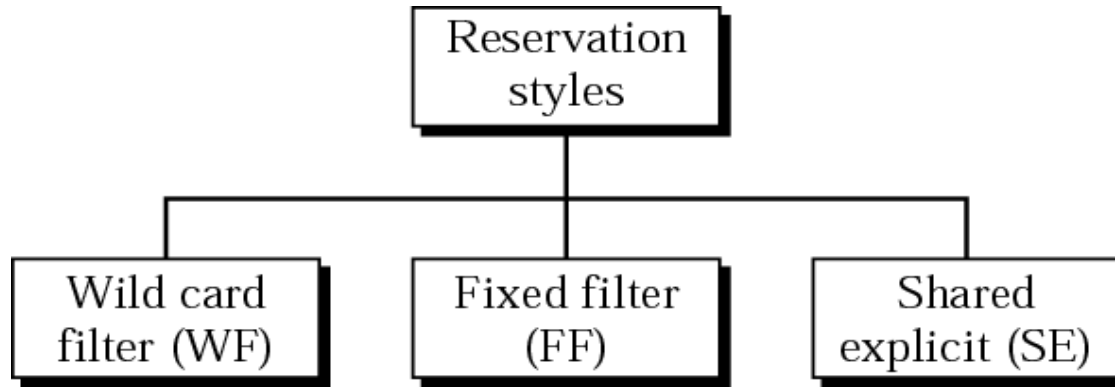
# Resv messages

- When rcvr rceives path msg,it sends recv msg which trvels toward sender and makes rr on the routers that support rsvp.



## Reservation styles

When there is more than one flow the router needs to make reservation to accommodate all of them



- In wf, router creates single reservation for all senders. reservation is based on larger request.
- In ff, router creates distinct reservation for flow.
- In se, router creates single reservation which can be shared by set of flows.

# Differentiated Services

An Alternative to  
Integrated Services





*Differentiated Services is a class-based QoS model designed for IP.*

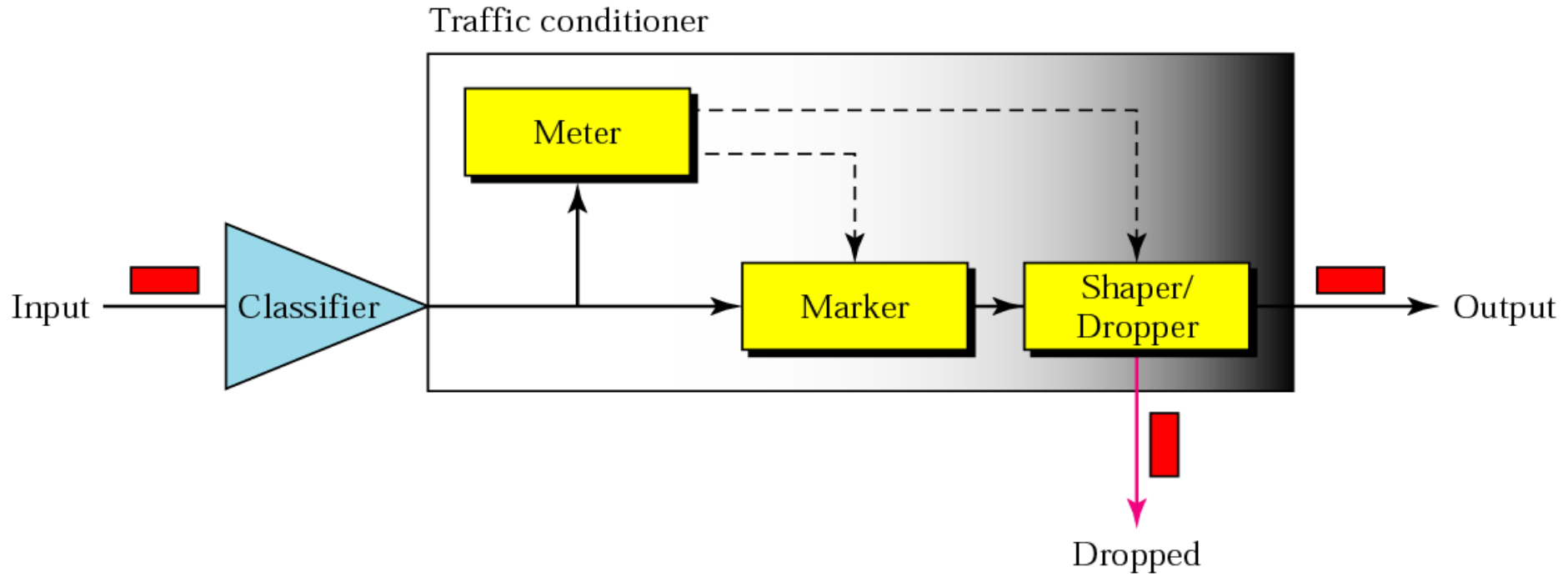
# DS field

- In diffserv each packet contains DS field whose value is set at the boundary of n/w by host or first router designated as boundary router.



- Differentiated services core point (DSCP) defines per hop behaviour.
- cu-currently unused.
- To implement diffserv DS node uses traffic conditioners such as
  - Meters
  - Markers
  - shapers.
- Meters check to see if incmg flow matches negotiated traffic profile
- Marker can mark a packet that is using best effort delivery.
- Shaper uses infn recvd from meter to reshape the traffic
- Dropper works as a shaper with no buffer ,discards the packets.

# Traffic conditioner



# QoS in Switched Networks

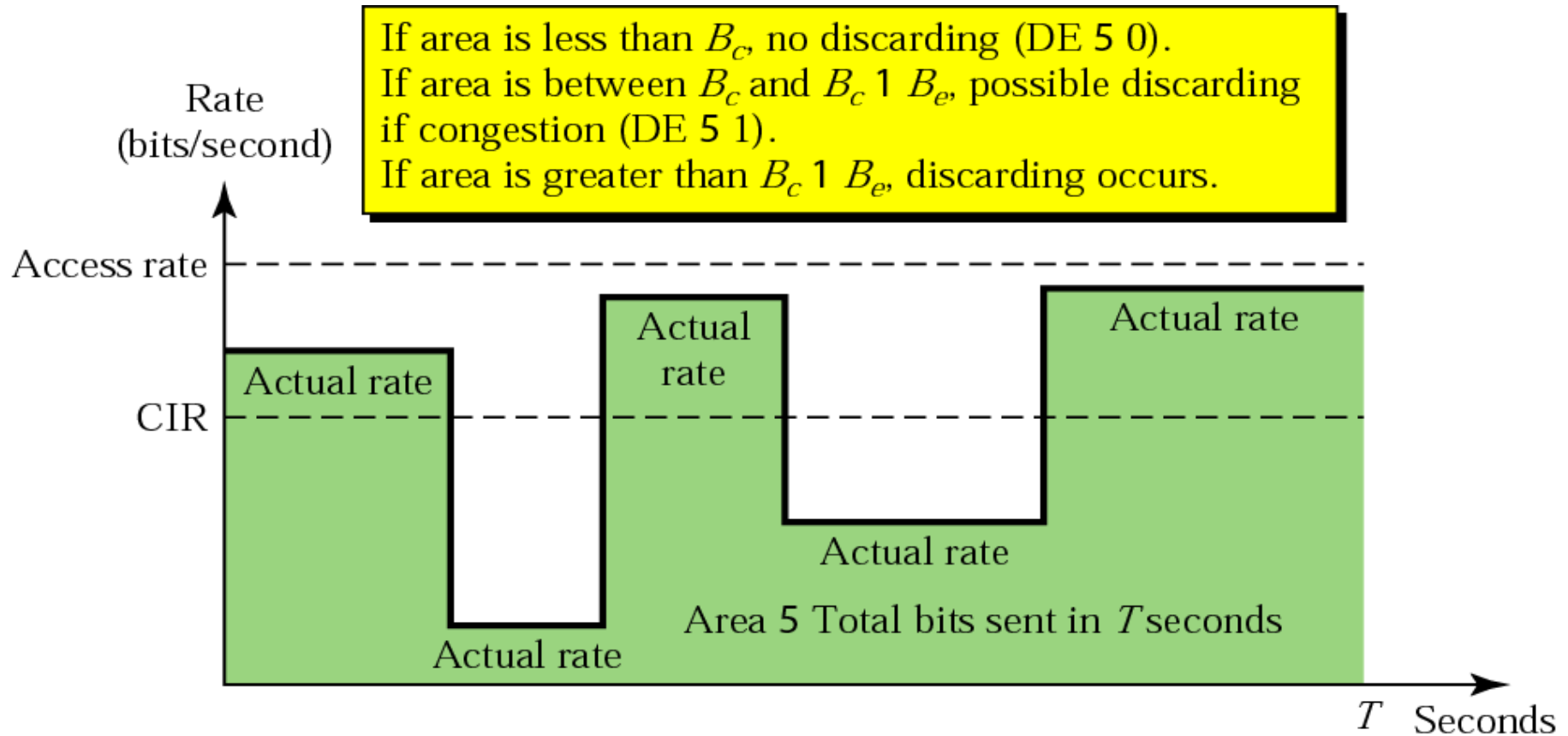
*QoS in Frame Relay*

*QoS in ATM*

# QoS IN SWITCHED NETWORKS

- **QoS in Frame Relay**
- Four different attributes to control traffic have been devised in Frame Relay:
  - access rate,
  - committed burst size  $B_c$
  - *committed information rate (CIR)*,
  - *excess burst size  $B_e$*
- For every connection access rate which depends on the bw of channel is defined.
- Committed burst size  $B_c$ : maximum no. of bits in predefined time that n/w is committed to transfer without discarding any frame r setting de bit.
- Committed info rate(CIR) defines avg rate in bits per sec.
  - $CIR = B_c / T$
- Excess burst size  $B_e$ : max no. of bits in excess of  $b_c$  that a user can send during a predefined time.

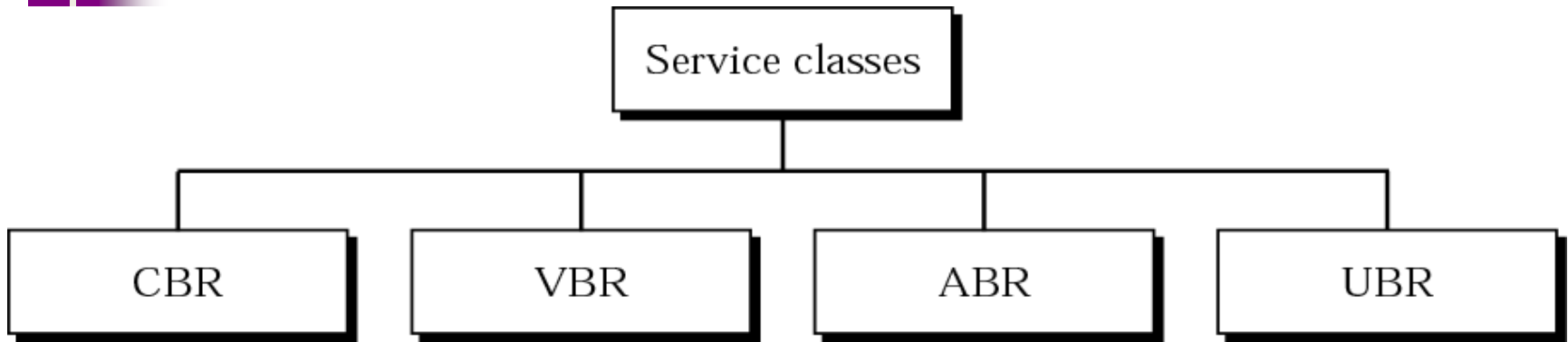
# User rate in relation to $B_c$ and $B_c + B_e$



# QoS in ATM

- The QoS in ATM is based on the
  - class,
  - user-related attributes,
  - and network-related attributes

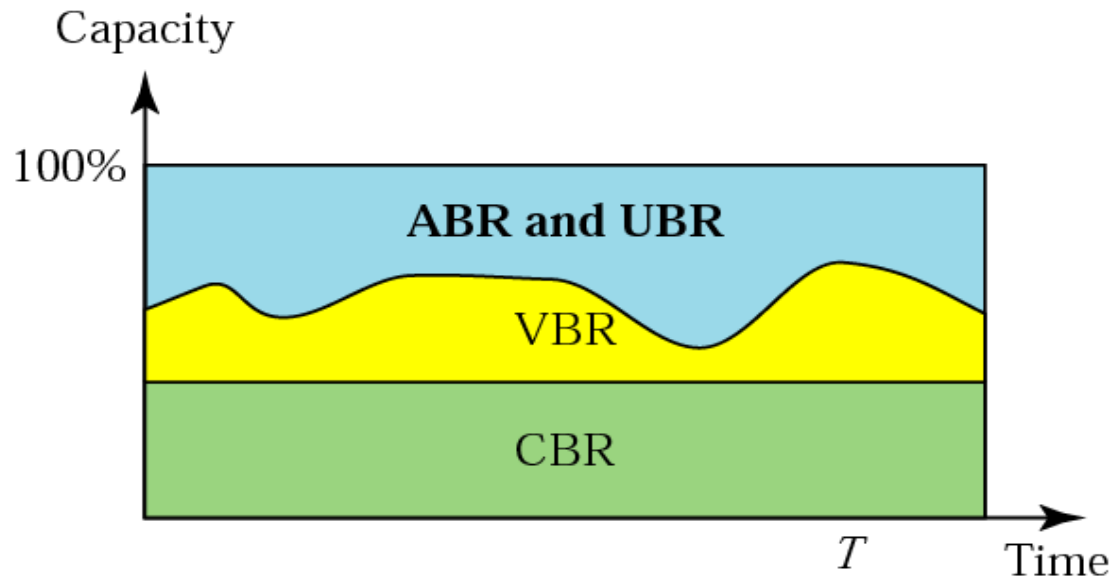
## ATM:Service classes



- CBR designed for customers who need real time audio, video services.
- VBR class is divided into two subclasses: real-time (VBR-RT) and non-real-time (VBR-NRT).
- VBR-RT is designed for those users who need real-time services
- VBR-NRT is designed for those users who do not need real-time services
- ABR (available bit rate) delivers cells at minimum rate.
- Unspecified bit rate (UBR) is a best effort delivery service that does not guarantee anything



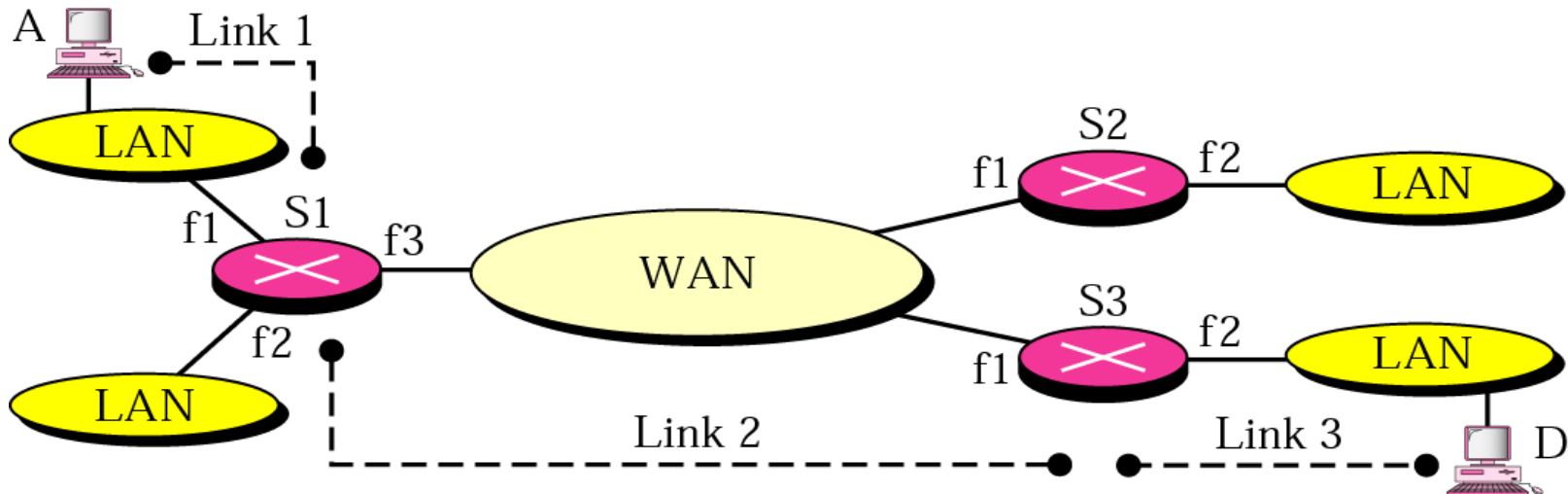
# Relationship of service classes to the total capacity



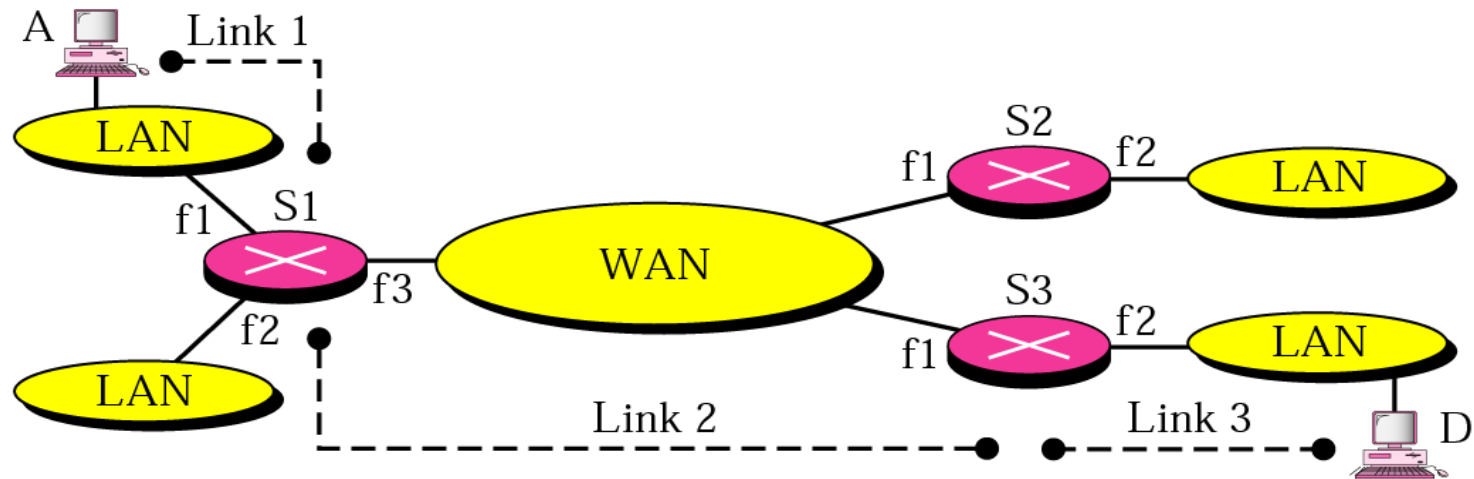
# **UNIT -IV**

# Internetwork

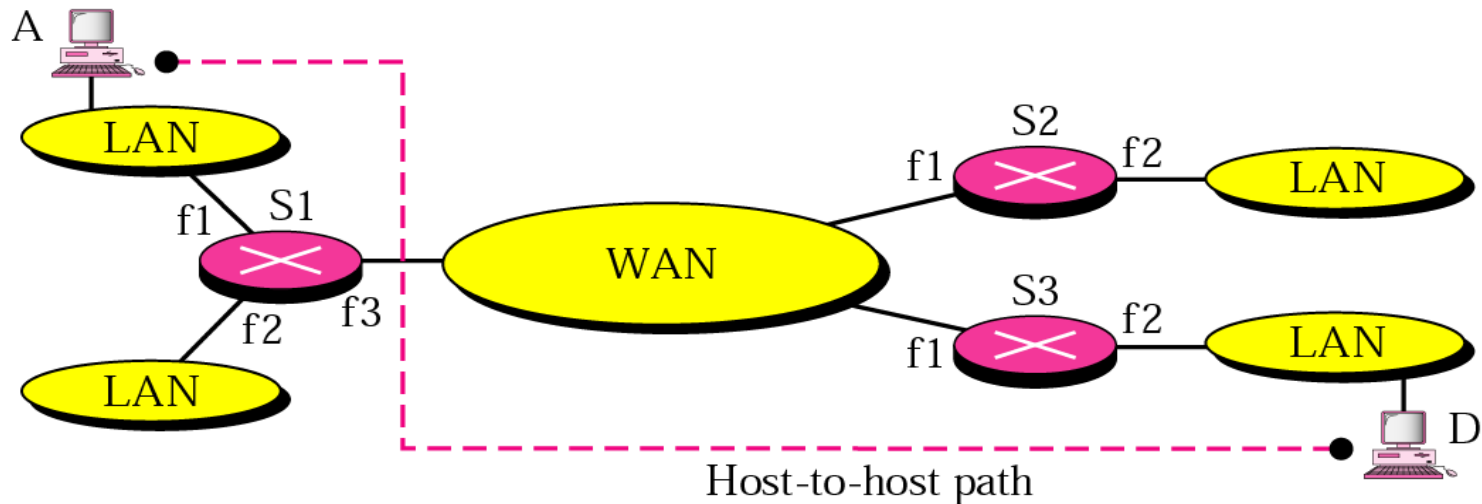
- Here Intern/w is made up of 4 lans and 1 wan.to solve prblm of delivery through several links,n/w layer was designed for host to host delivery and routing packets.



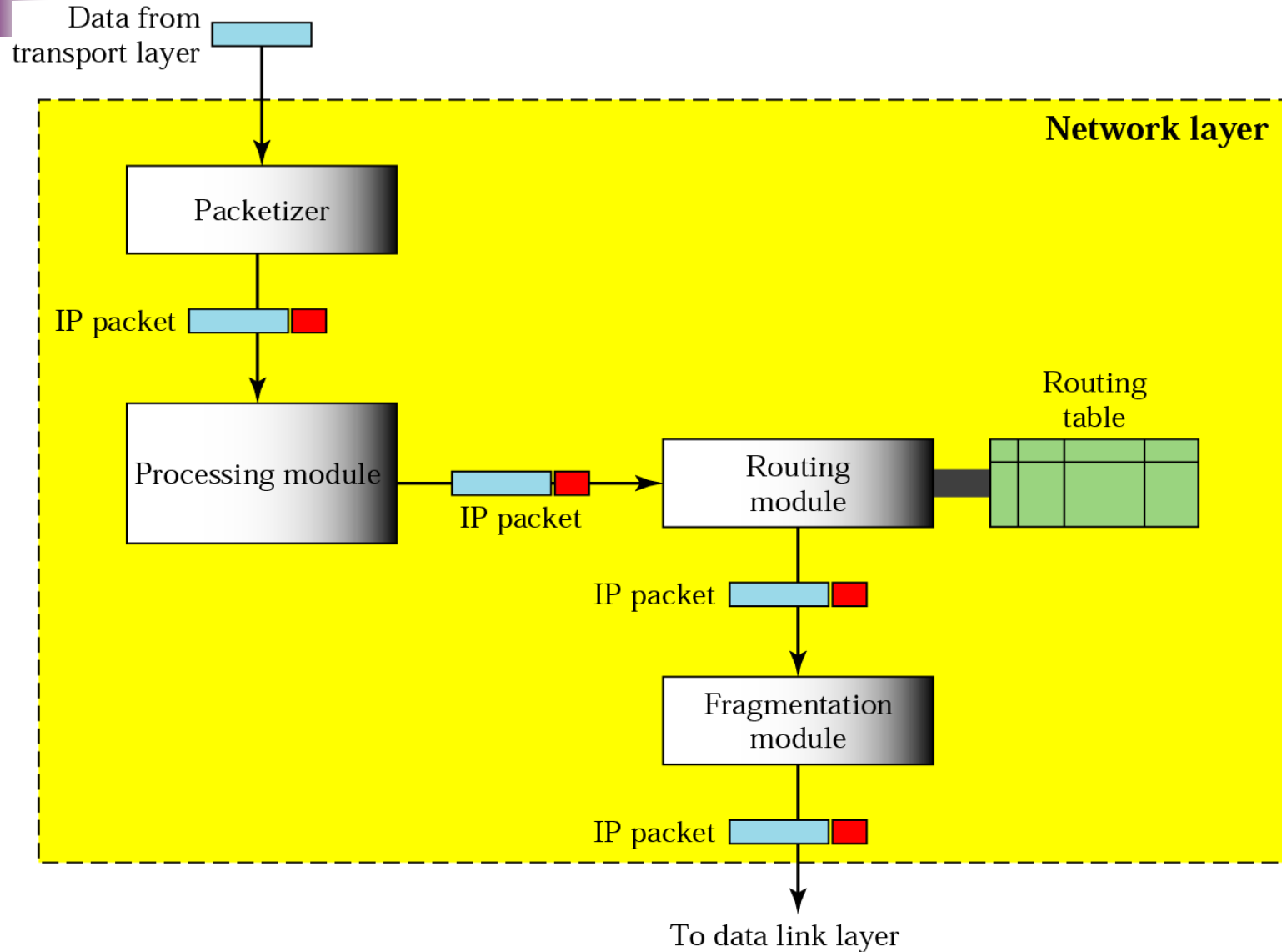
# Links in an internetwork



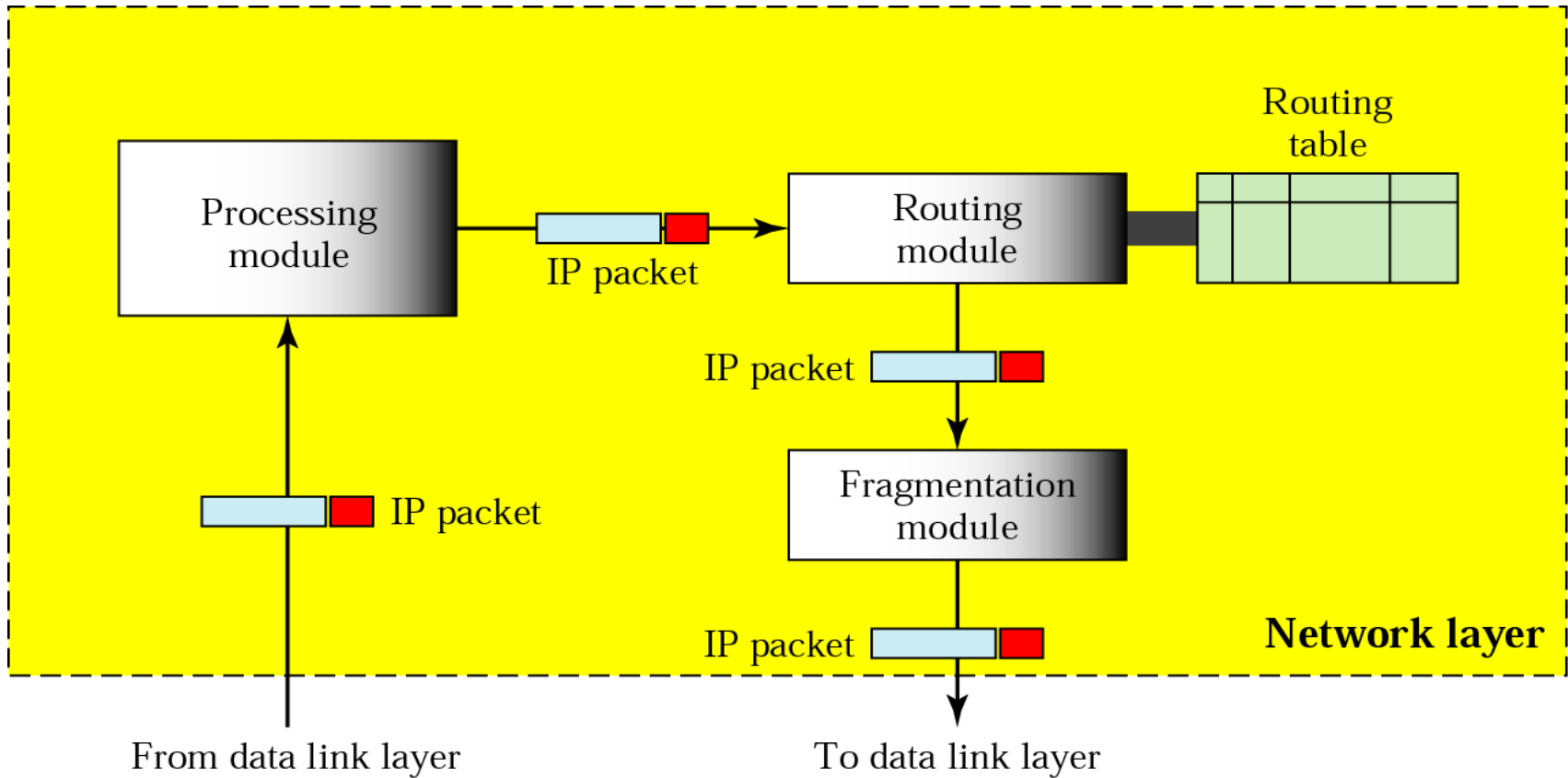
# Network layer in an internetwork



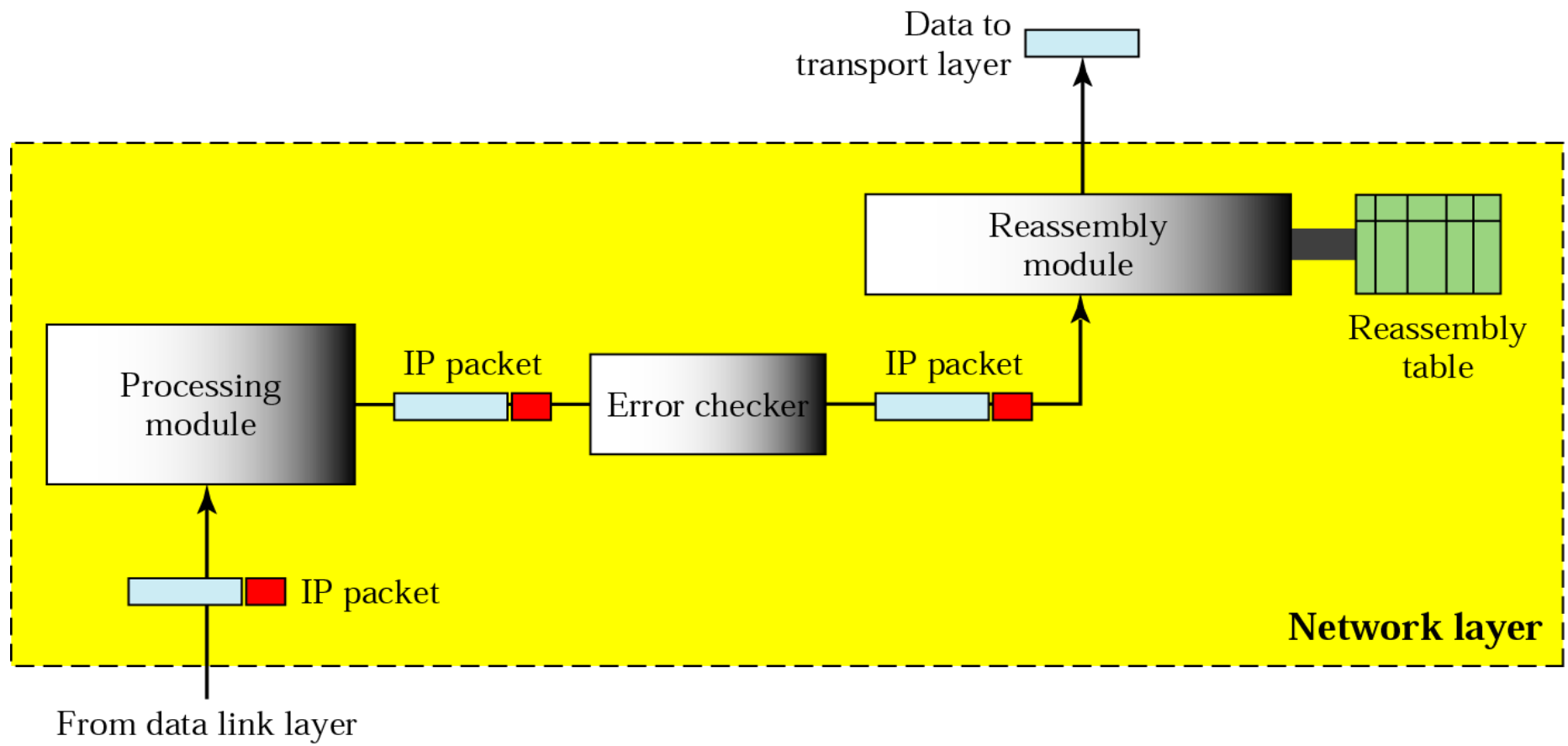
# Network layer at the source



# Network layer at a router



# Network layer at the destination





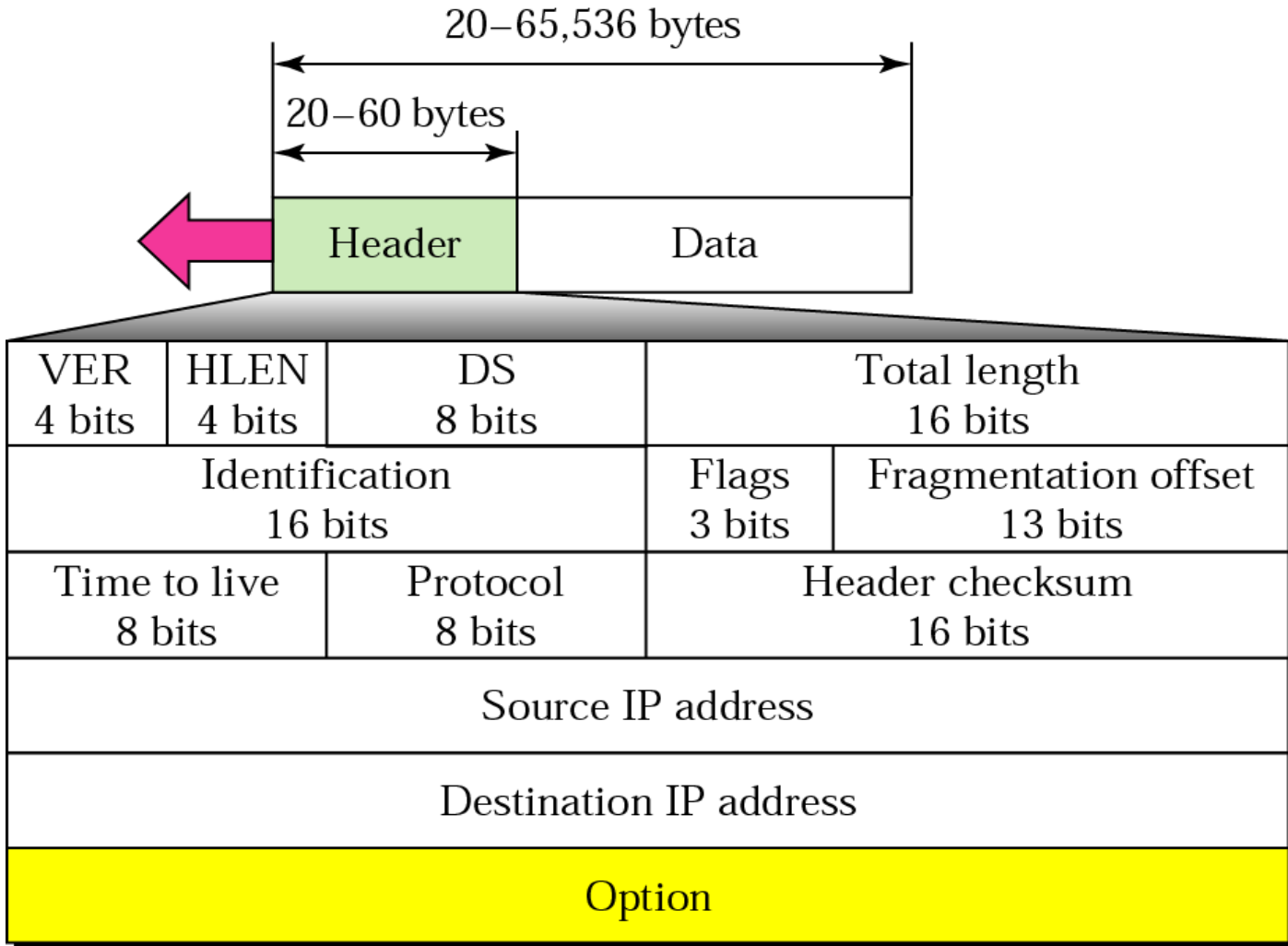
# IPv4

- The Internet Protocol version 4 (IPv4) is the delivery mechanism used by the TCP/IP protocols.
- IPv4 is an unreliable and connectionless datagram protocol—a best-effort delivery service.
- The term *best-effort* means that IPv4 provides no error control or flow control (except for error detection on the header).
- IPv4 assumes the unreliability of the underlying layers and does its best to get a transmission through to its destination, but with no guarantees.
- If reliability is important, IPv4 must be paired with a reliable protocol such as TCP.

# Datagram

- Packets in the IPv4 layer are called datagram's.
- A datagram is a variable-length packet consisting of two parts: header and data.
- The header is 20 to 60 bytes in length and contains information essential to routing and delivery.
- It is customary in *TCP/IP* to show the header in *4-byte sections*.

# IP datagram



# *A brief*

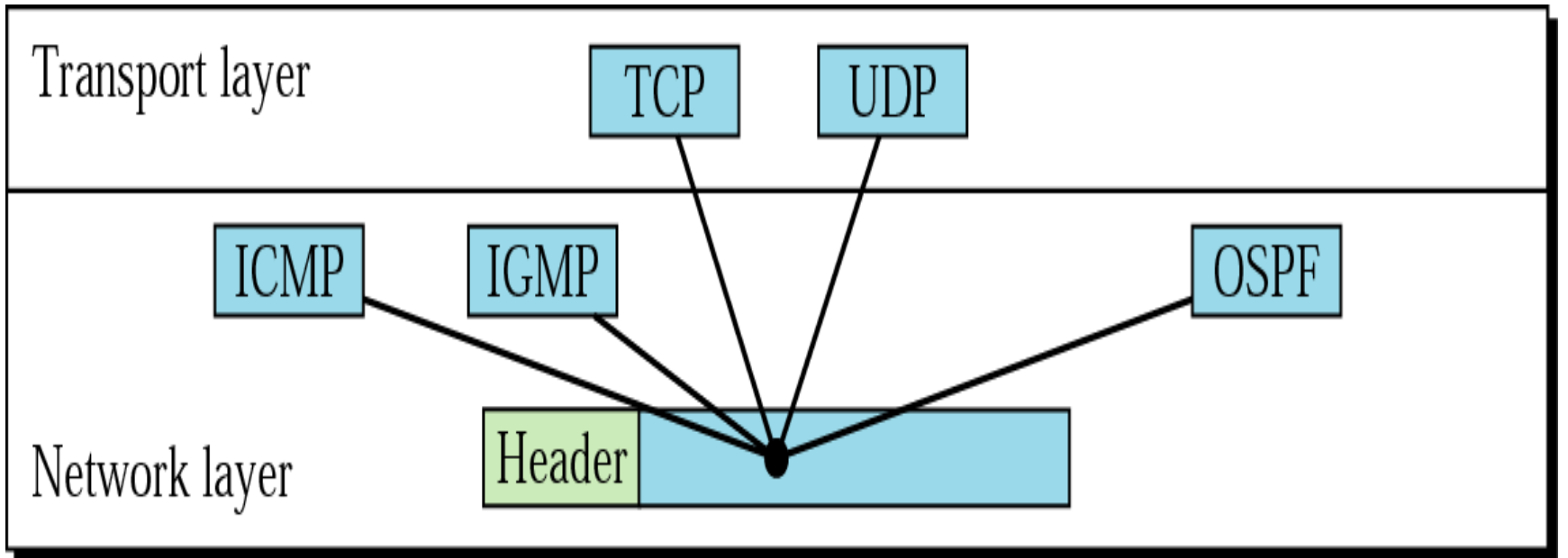
description of each field is in order.

- **Version (VER).** This 4-bit field defines the version of the IPv4 protocol.
- **Header length (HLEN).** This 4-bit field defines the total length of the datagram header in 4-byte words.
- This field is needed because the length of the header is variable (between 20 and 60 bytes).
- **Differentiated services:**
- This field defines the class of the datagram for quality-of – service purposes.
- **Total length:** this field defines the total length(header plus data) of the IP datagram in bytes.
- **Identification, flag, and offset:**

# *A brief*

## description of each field is in order.

- **Time to live:** A datagram has a limited lifetime in its travel through an internet.
- This field was originally designed to hold a timestamp, which was decremented by each visited router.
- The datagram was discarded when the value became zero.
- This field is used mostly to control the maximum number of hops (routers) visited by the datagram.
- When a source host sends the datagram, it stores a number in this field. This value is approximately 2 times the maximum number of routes between any two hosts.
- Each router that processes the datagram decrements this number by 1. If this value, after being decremented, is zero, the router discards the datagram.
- **Protocol.** This 8-bit field defines the higher-level protocol that uses the services of the IPv4 layer. An IPv4 datagram can encapsulate data from several higher-level protocols such as TCP, UDP, ICMP, and IGMP.



# Fragmentation

- A datagram can travel through different networks. Each router de-capsulates the IPv4 datagram from the frame it receives, processes it, and then encapsulates it in another frame.
- The format and size of the received frame depend on the protocol used by the physical network through which the frame has just traveled.
- The format and size of the sent frame depend on the protocol used by the physical network through which the frame is going to travel.
- For example, if a router connects a LAN to a WAN, it receives a frame in the LAN format and sends a frame in the WAN format.

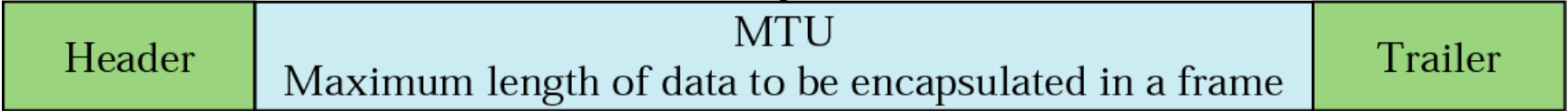
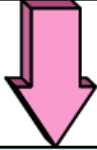
# *Maximum Transfer Unit (MTU)*

- Each data link layer protocol has its own frame format in most protocols.
- One of the fields defined in the format is the maximum size of the data field.
- In other words, when a datagram is encapsulated in a frame, the total size of the datagram must be less than this maximum size



The value of the MTU depends on the physical network protocol.

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



Frame

- To make the IPv4 protocol independent of the physical network, the designers decided to make the maximum length of the IPv4 datagram equal to **65,535 bytes**.
- When a datagram is fragmented, each fragment has its own header with most of the fields repeated, but with some changed.
- A fragmented datagram may itself be fragmented if it encounters a network with an even smaller MTU In IPv4, a datagram can be fragmented by the source host or any router in the path
- The reassembly of the datagram, however, is done only by the destination host because each fragment becomes an independent datagram.

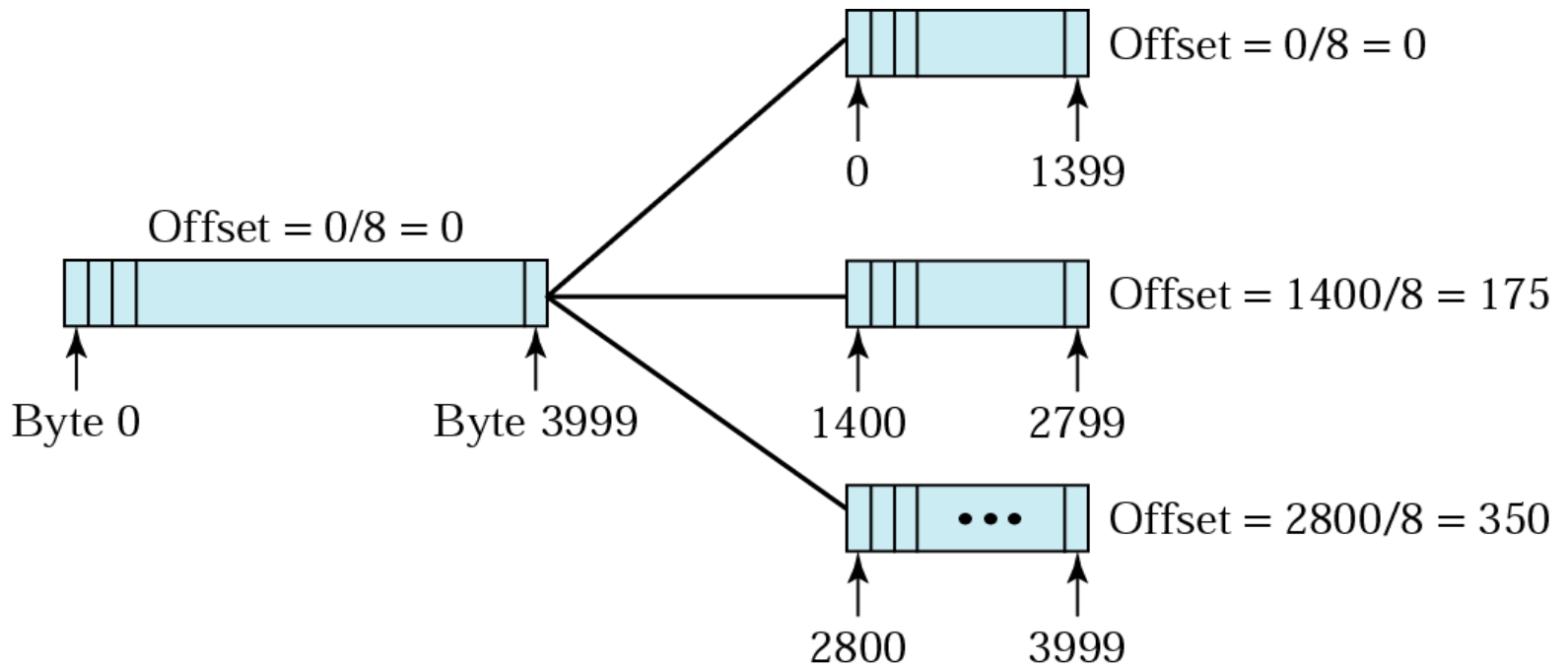
- Whereas the fragmented datagram can travel through different routes, and we can never control or guarantee which route a fragmented datagram may take, all the fragments belonging to the same datagram should finally arrive at the destination host.
- So it is logical to do the reassembly at the final destination.

- When a datagram is fragmented, required parts of the header must be copied by all fragments.
- The host or router that fragments a datagram must change the values of three fields:
  - Flags
  - fragmentation
  - Offset
  - Total length.

- **Identification.** This 16-bit field identifies a datagram originating from the source host.
  - The combination of the identification and source IPv4 address must uniquely define a datagram as it leaves the source host.
- **Flags.** This is a 3-bit field.
  - The first bit is reserved.
  - The second bit is called the ***do not fragment bit***. *If its value is 1, the machine must not fragment the datagram.*
  - *If it cannot pass the datagram through any available physical network, it discards the datagram and sends an ICMP error message to the source host.*
- If its value is 0, the datagram can be fragmented if necessary.
  - The third bit is called the *more fragment bit*. *If its value is 1, it means the datagram is not the last fragment;*
  - there are more fragments after this one.
  - If its value is 0, it means this is the last or only fragment

- Figure shows a datagram with a data size of 4000 bytes fragmented into three fragments.
- The bytes in the original datagram are numbered 0 to 3999.
- **The first fragment** carries bytes 0 to 1399.
- The **offset** for this datagram is  $0/8 = 0$ .
  
- The **second fragment** carries bytes 1400 to 2799; **the offset** value for this fragment is  $1400/8 = 175$ .
  
- Finally, **the third fragment** carries bytes 2800 to 3999. **The offset** value for this fragment is  $2800/8 = 350$ .
- Remember that the value of the offset is measured in units of 8 bytes.
- This is done because the length of the offset field is only 13 bits and cannot represent a sequence of bytes greater than 8191.
- This forces hosts or routers that fragment datagram's to choose a fragment size so that the first byte number is divisible by 8.

# Fragmentation example





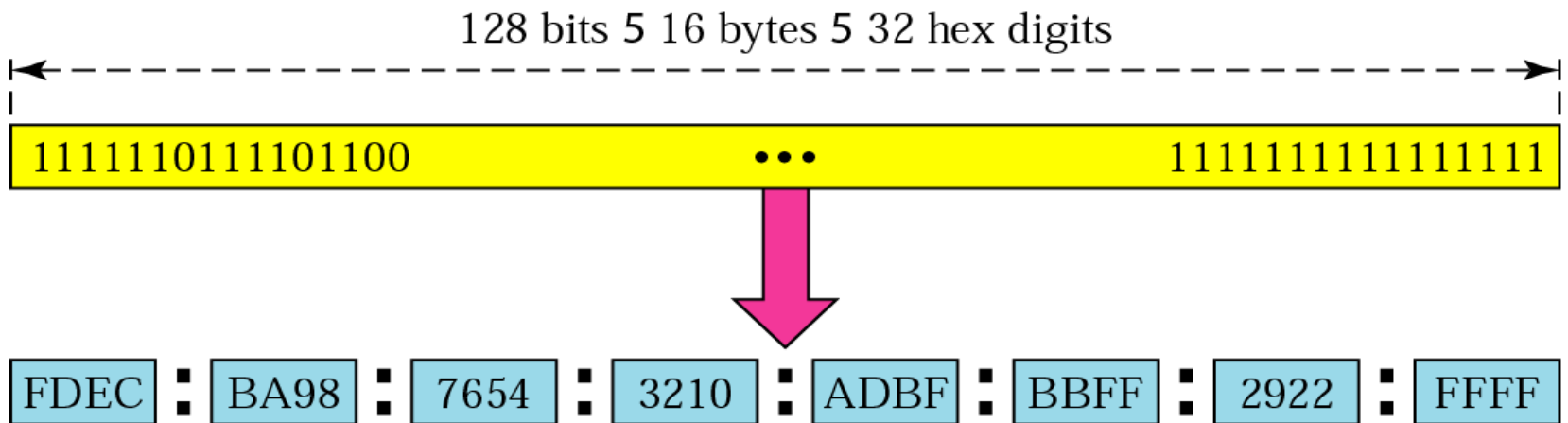
# IPV4 deficiencies:

- IPV4 has a two-level address structure categorized into five classes. The use of address space is inefficient.
- The internet must accommodate real-time audio and video transmission. This type of transmission requires minimum delay strategies and reservation of resources not provided in the ipv4 design.
- The internet must accommodate encryption and authentication of data for some application. No security mechanism was provided by IPv4

# IPv6

- Internetworking protocol, next generation(IPng) was

# IPv6 address



Unabbreviated

FDEC ■ BA98 ■ 0074 ■ 3210 ■ 000F ■ BBFF ■ 0000 ■ FFFF



FDEC ■ BA98 ■ 74 ■ 3210 ■ F ■ BBFF ■ 0 ■ FFFF

Abbreviated

# Abbreviated address with consecutive zeros

Abbreviated

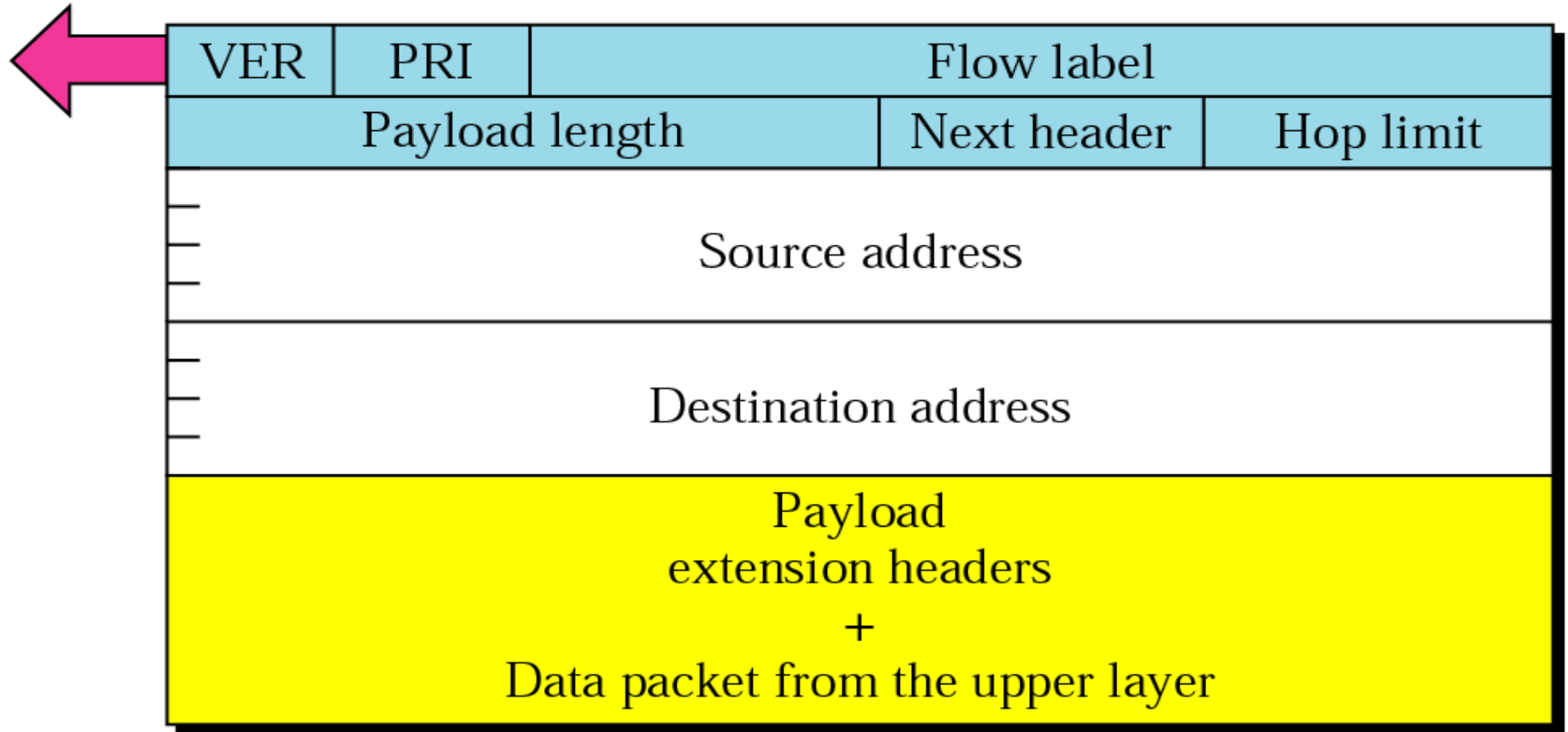
FDEC ■ 0 ■ 0 ■ 0 ■ 0 ■ BBFF ■ 0 ■ FFFF



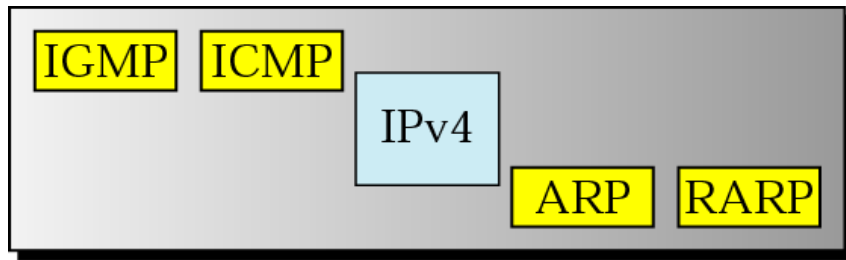
FDEC ■ ■ BBFF ■ 0 ■ FFFF

More Abbreviated

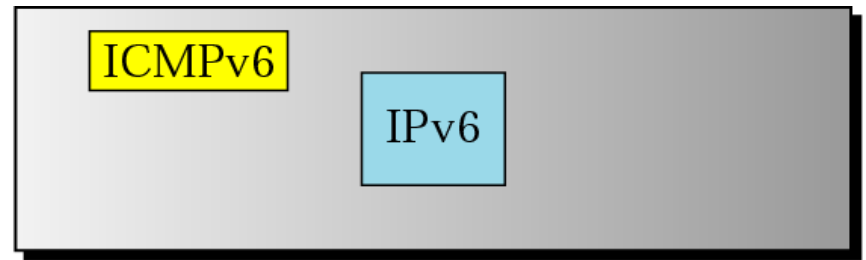
# Format of an IPv6 datagram



# Comparison of network layers in version 4 and version 6

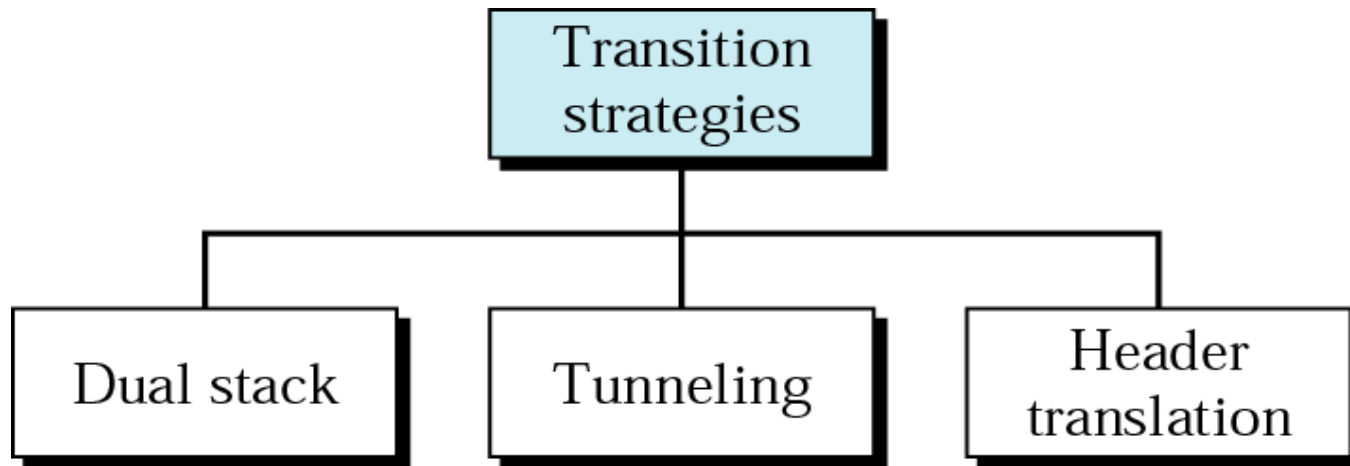


Network layer in version 4



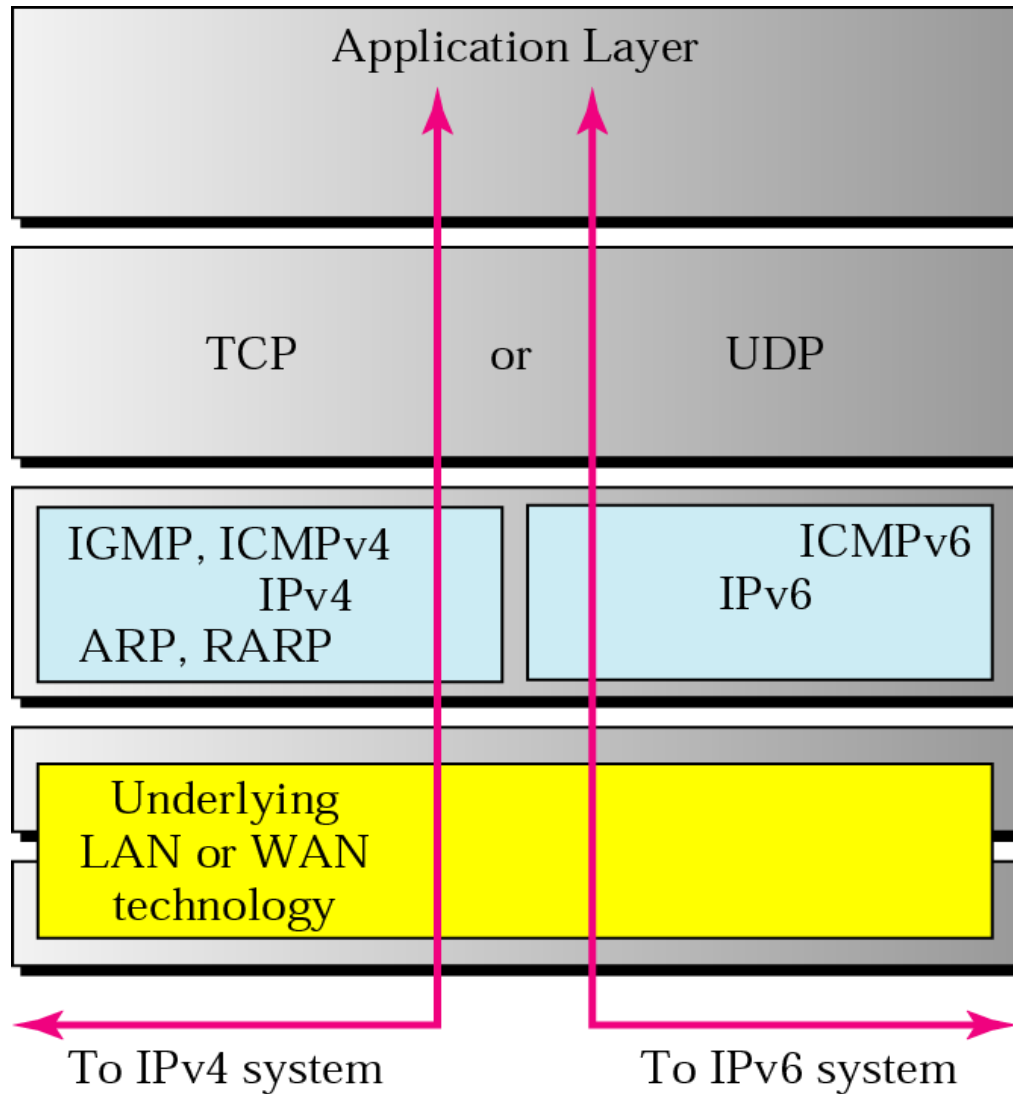
Network layer in version 6

## Three transition strategies

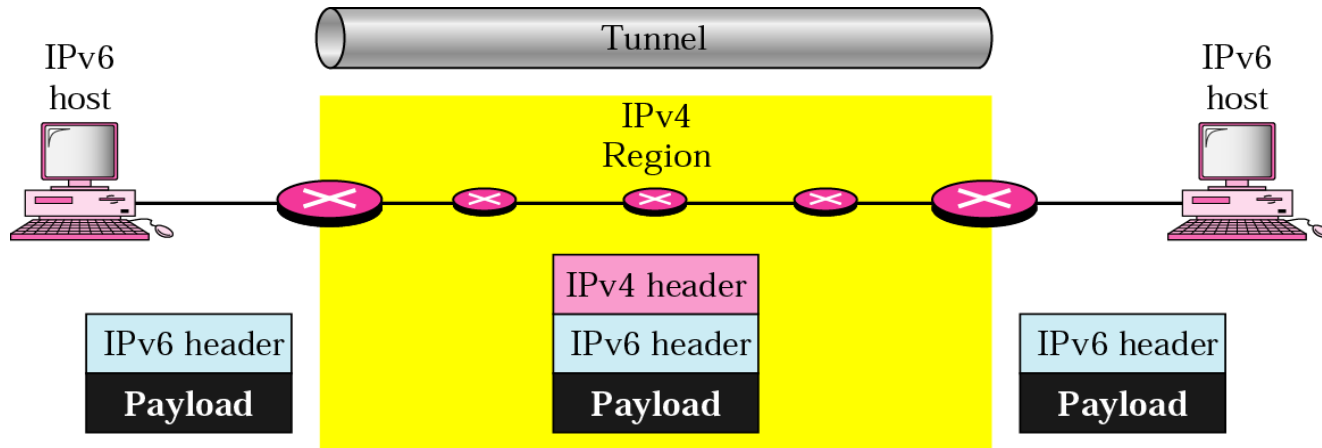




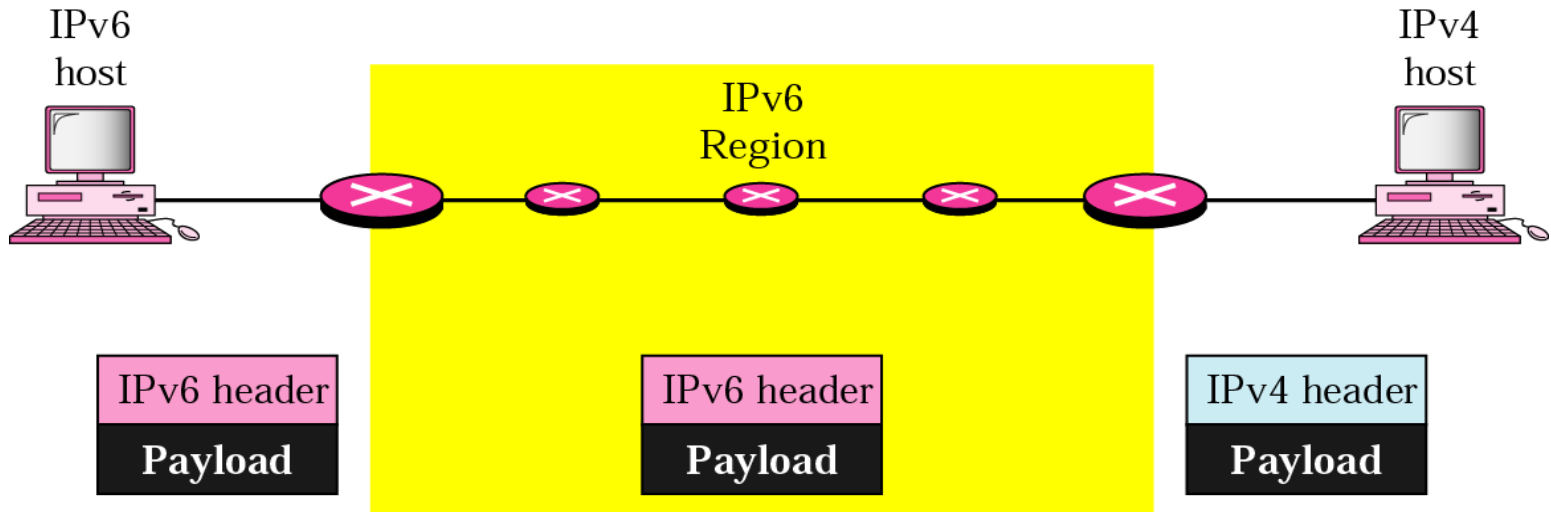
# Three transition strategies



# Tunneling



# Header translation



# ARP-INTRODUCTION

- An internet is made of a combination of physical networks connected by internetworking devices such as routers.
- A packet starting from a source host may pass through several different physical networks before finally reaching the destination host.
- The hosts and routers are recognized at the network level by their logical (IP) addresses.
- A physical address is a local address. It must be unique locally, but is not necessarily unique universally.
- It is called a *physical address* because it is usually (but not always) implemented in hardware.
- An example of a physical address is the 48-bit MAC address in the Ethernet protocol, which is imprinted on the NIC installed in the host or router

# ARP

- The physical address and the logical address are two different identifiers.
- A packet at a network layer such as IP may pass through different physical networks such as Ethernet and LocalTalk (Apple)
- This means that delivery of a packet to a host or a router requires two levels of addressing:
  - logical and
  - physical.

# MAPPING

- We need to be able to map a logical address to its corresponding physical address and vice versa.
- To implement these changes, a static mapping table must be updated periodically. This overhead could affect network performance.
- These can be done by using either
  - static or
  - dynamic mapping.

# STATIC Mapping

- Static mapping involves in the creation of a table that associates a logical address with a physical address.
- This table is stored in each machine on the network. Each machine that knows, the IP address of another machine but not its physical address
- This has some limitations because physical addresses may change in the following ways:
  - 1. A machine could change its NIC, resulting in a new physical address.
  - 2. In some LANs, such as LocalTalk, 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
- To implement these changes, a static mapping table must be updated periodically. This overhead could affect network performance.

# Mapping Logical to Physical Address: ARP

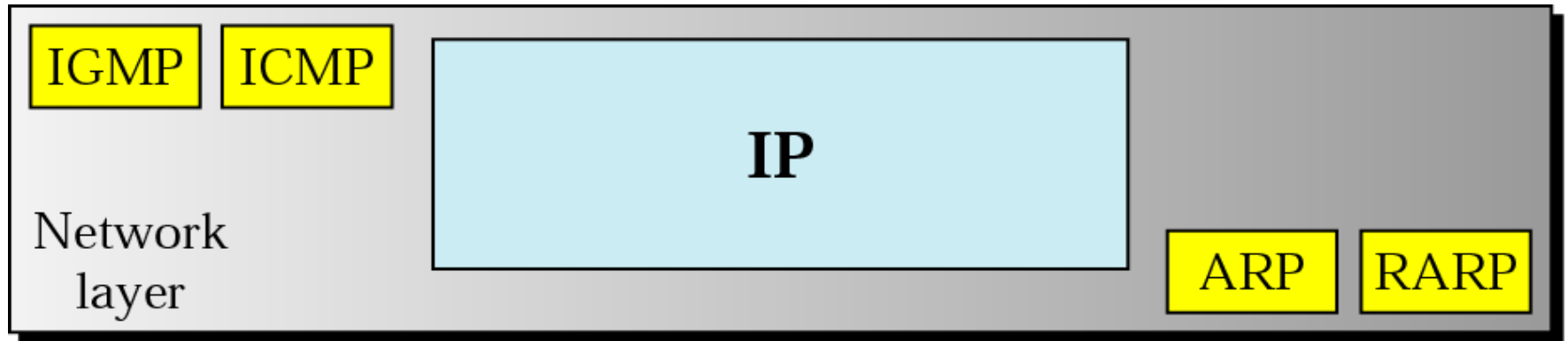
- In dynamic mapping each time a machine knows one of the two addresses (logical or physical), it can use a protocol to find the other one.
- Two protocols have been designed to perform dynamic mapping:
  - Address Resolution Protocol(ARP)
  - Reverse Address Resolution(RARP)



# ARP

- It maps an IP address to a MAC address
- **RARP** maps a MAC address to an IP address
- ARP associates an IP address on a link is identified by a physical or station address that is usually
- Anytime a host or a router has an IP datagram to send to another host or router, it has the logical (IP) address of the receiver imprinted on the NIC
- This means that the sender needs the physical address of the receiver.
- The host or the router sends an ARP query packet. The packet includes the physical and IP addresses of the sender and the IP address of the receiver.
- Because the sender does not know the physical address of the receiver, the query is broadcast over the network
- Every host or router on the network receives and processes the ARP query packet, but 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.

# Protocols at network layer



# ARP

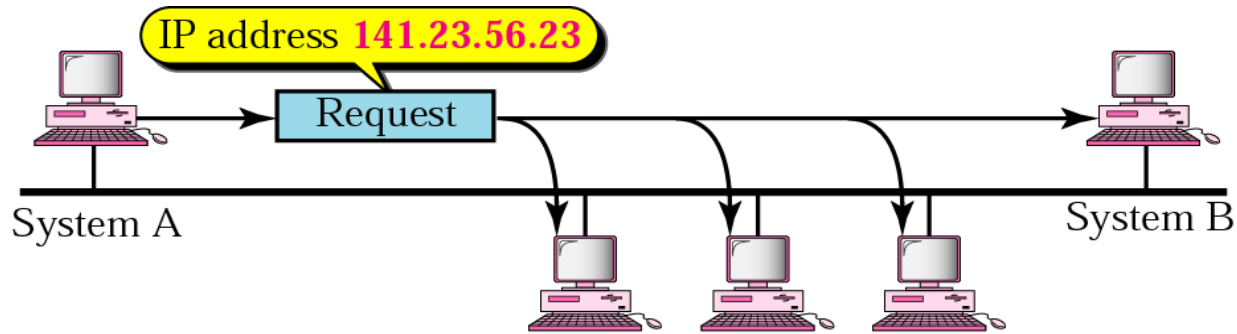
*Mapping*

*Packet Format*

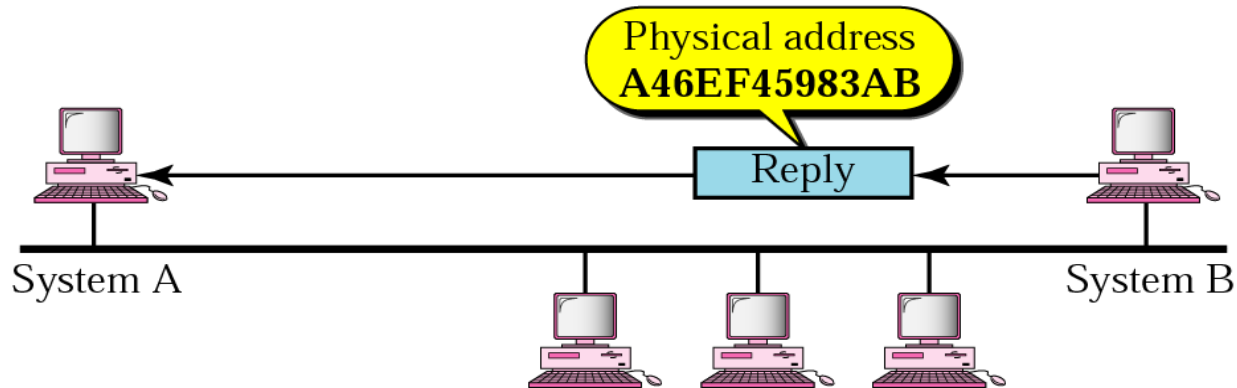
*Encapsulation*

*Operation*

# ARP operation



a. ARP request is broadcast



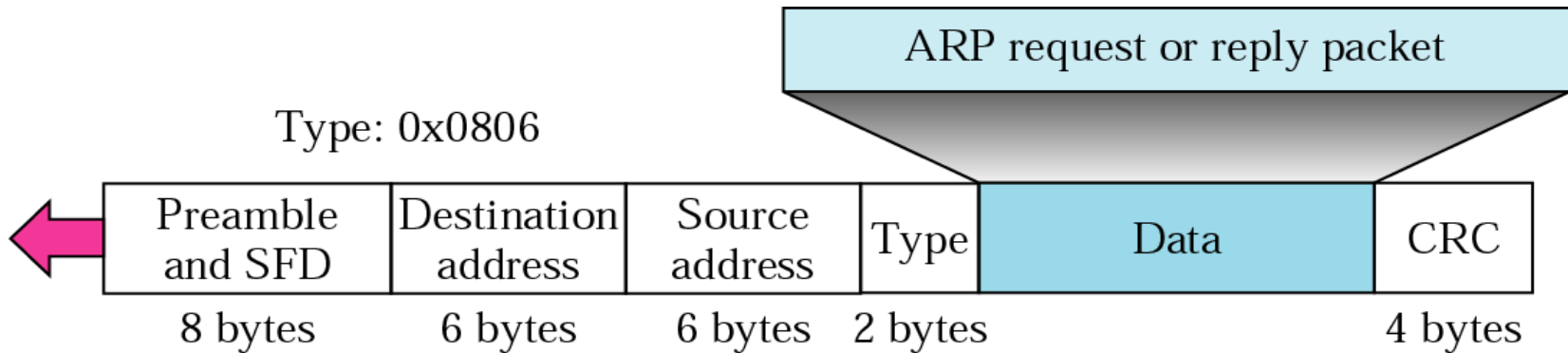
b. ARP reply is unicast

# ARP packet

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)		

# Encapsulation of ARP packet

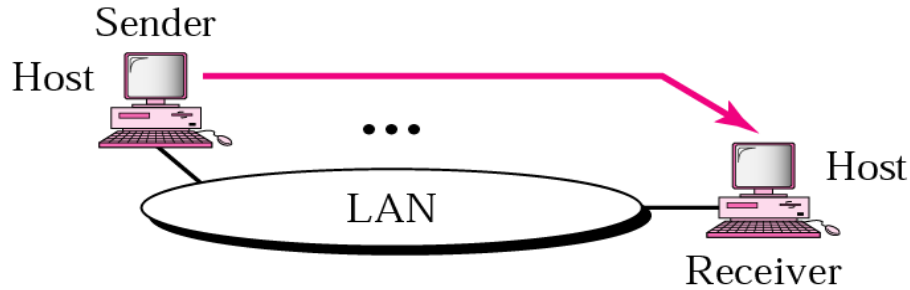
An ARP packet is encapsulated directly into a data link frame.



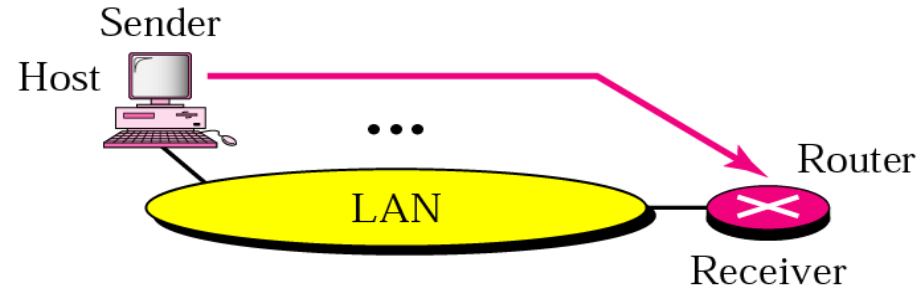
# Operation

- 1.sndr knows ipaddress of target.
- 2.ip asks **ARP** to create **ARP** request msg,filling sndr physical addr,sender **IP** addr,target **IP** add.
- 3.msg is passed to dll where it is encapsulated in a frame by using physical address of sndr at source address and physical broadcast address at dest address.
- 4.every host or router receives the frame.bcoz frame contains broadcast dest address,all stns remove msg and pass it to arp.all machines except one targeted drop the packet.target machine recgnizes ip address.
- 5.target machine replies with **ARP** reply msg that contains its physical address.msg is unicast.
- 6.sndr rcvs reply msg.it now kows physical address of target machine.
- 7.IP addr which carries data for target machine is now encapsulated in a frame and is unicast to dest.

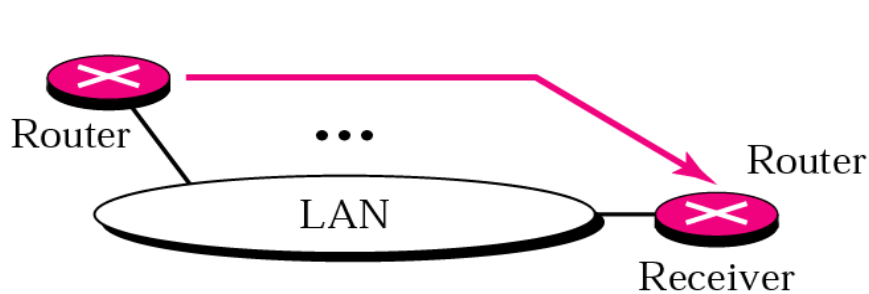
# Four cases using ARP



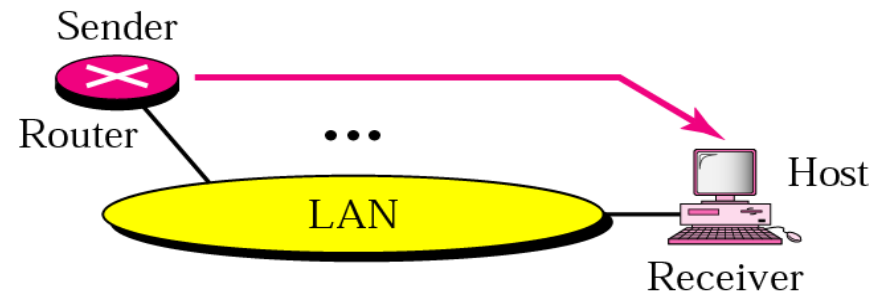
Case 1. A host has a packet to send to another host on the same network.



Case 2. A host wants to send a packet to another host on another network. It must first be delivered to the appropriate router.



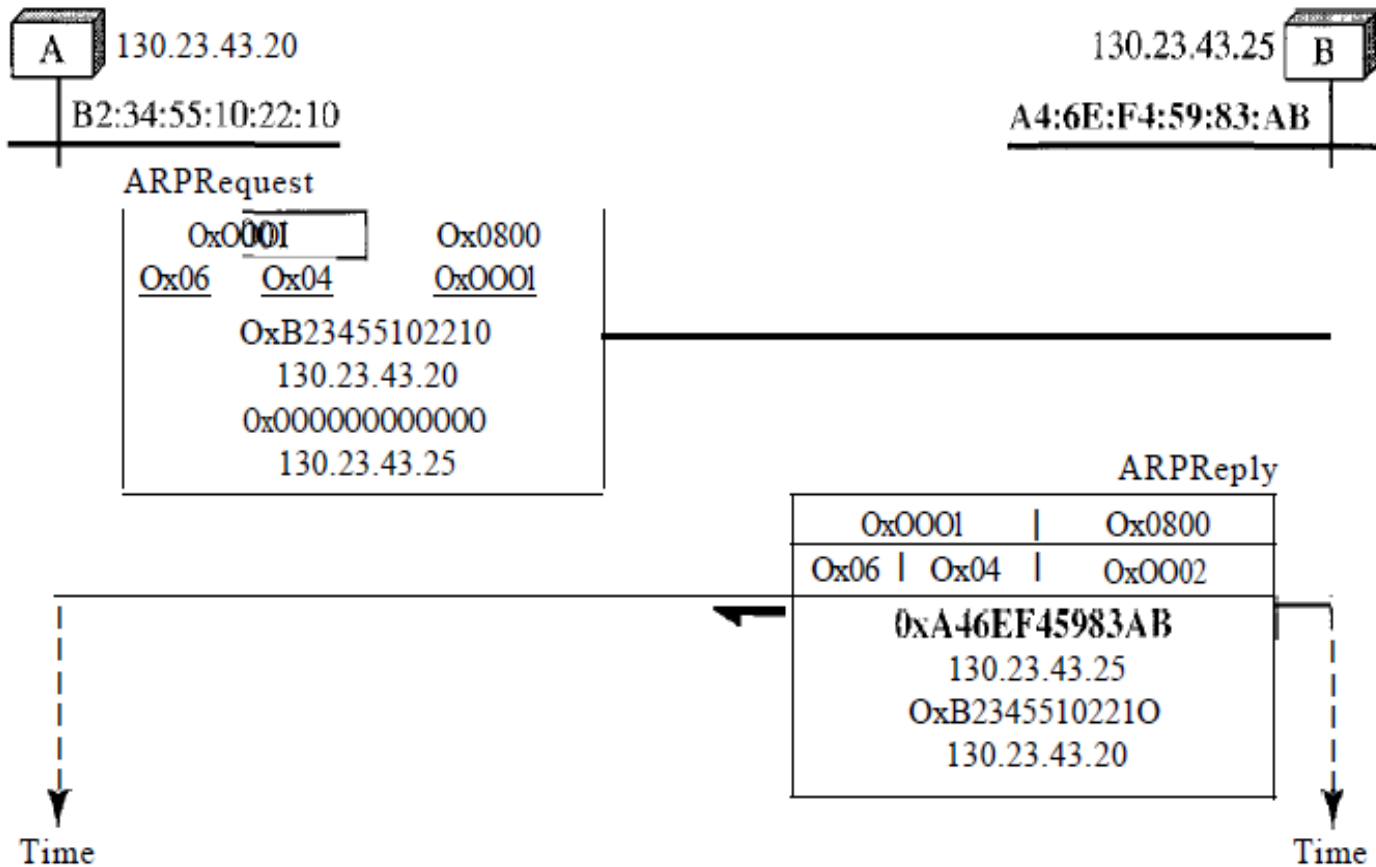
Case 3. A router receives a packet to be sent to a host on another network. It must first be delivered to the appropriate router.



Case 4. A router receives a packet to be sent to a host on the same network.



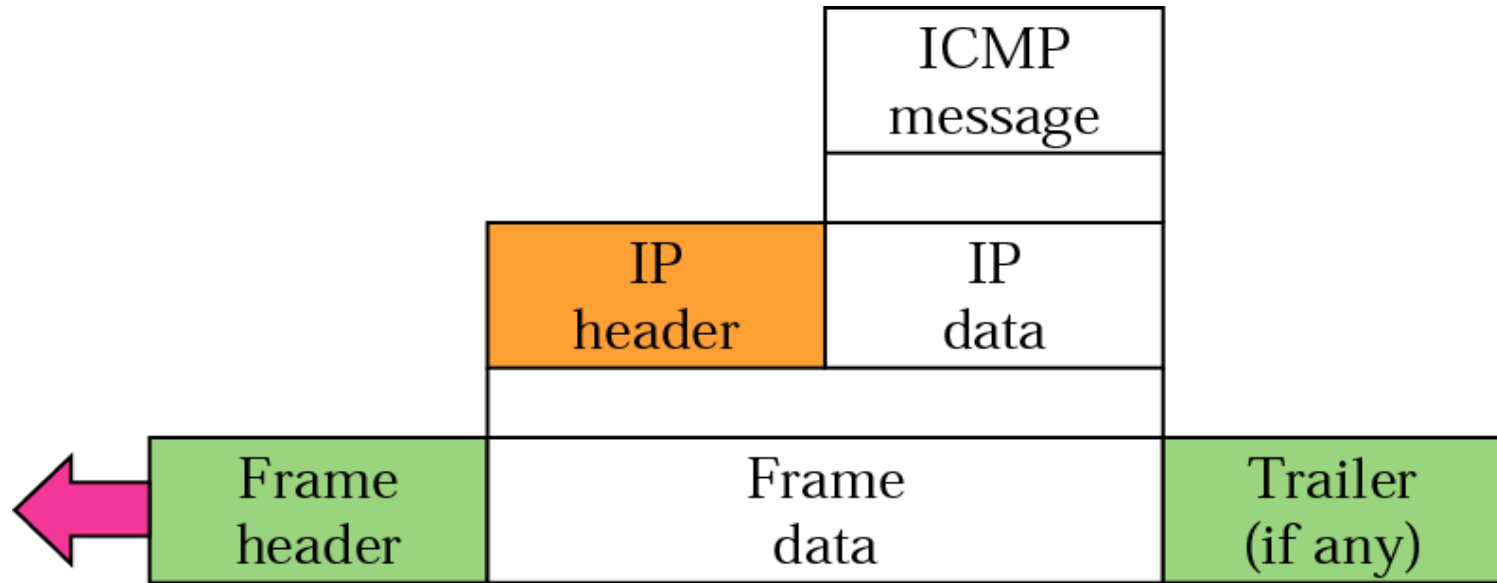
- ***Example***
- A host with IP address 130.23.43.20 and physical address B2:34:55:10:22:10 has a packet to send to another host with IP address 130.23.43.25 and physical address A4:6E:F4:59:83:AB (which is unknown to the first host). The two hosts are on the same Ethernet network. Show the ARP request and reply packets encapsulated in Ethernet frames.
- **Solution**
- Figure shows the ARP request and reply packets. Note that the ARP data field in this case is 28 bytes, and that the individual addresses do not fit in the 4-byte boundary.
- That is why we do not show the regular 4-byte boundaries for these addresses.

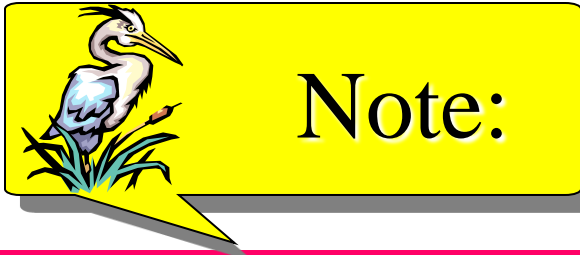


# ICMP

- The IP protocol also lacks a mechanism for host and management queries.
- A host sometimes needs to determine if a router or another host is alive.
- And sometimes a network administrator needs information from another host or router.
- The Internet Control Message Protocol (ICMP) has been designed to compensate for the above two deficiencies.
- It is a companion to the IP protocol
- The value of the protocol field in the IP datagram is 1 to indicate that the IP data are an ICMP message.

# ICMP encapsulation





*ICMP always reports error messages to the original source.*

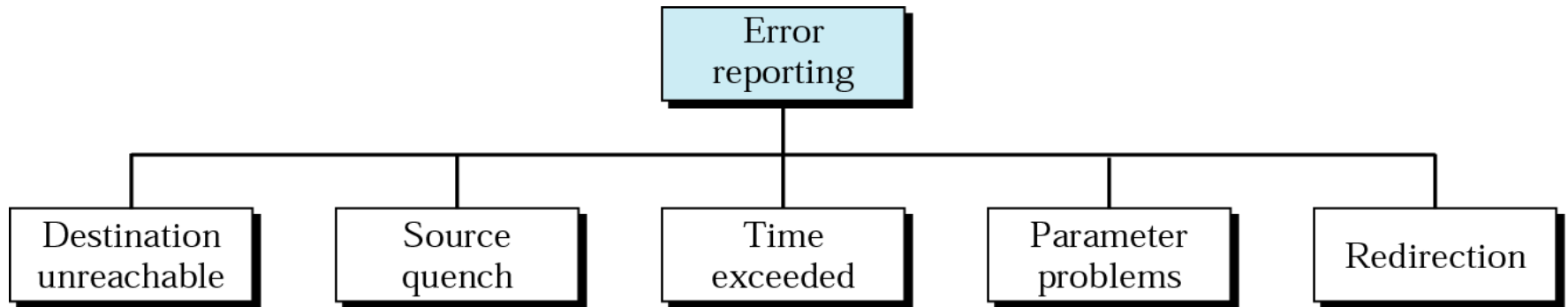
# ICMP

*Types of Messages*

*Error Reporting*

*Query*

# Error-reporting messages



# *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.
- Note that destination-unreachable messages can be created by either a router or the destination host.



# source-quench

- When a router or host discards a datagram due to congestion, it sends a source-quench message to the sender of the datagram.
- This message has two purposes.
- **First**, it informs the source that the datagram has been discarded.
- **Second**, it warns the source that there is congestion somewhere in the path and that the source should slow down (quench) the sending process.

# *Time Exceeded*

- When the time-to-live value reaches 0, after decrementing, the router discards the datagram.
- **First:** when the datagram is discarded, a time-exceeded message must be sent by the router to the original source.
- **Second,** a time-exceeded message is also generated when not all fragments that make up a message arrive at the destination host within a certain time limit.

# *Parameter Problem*

- Any ambiguity in the header part of a datagram can create serious problems as the datagram travels through the Internet.
- 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

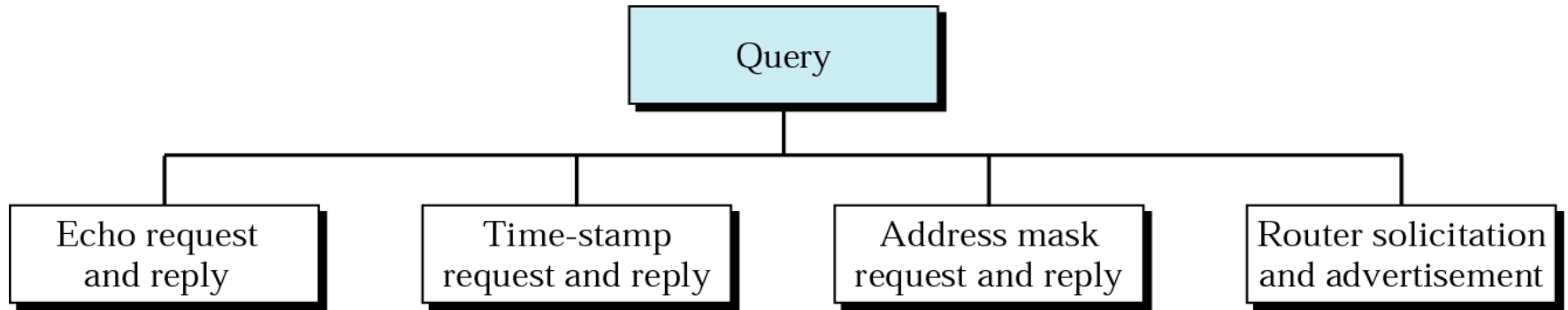
# redirection message

- If Any host may send a datagram, which is destined for another network, to the wrong router.
- In this case, the router that receives the datagram will forward the datagram to the correct router.
- However, to update the routing table of the host, it sends a redirection message to the host



*There is no flow control or congestion control mechanism in IP.*

# Query messages



# *Echo Request and Reply*

- The echo-request and echo-reply messages are designed for diagnostic purposes.
- Network managers and users utilize this pair of messages to identify network problems.
- The combination of echo-request and echo-reply messages determine whether two systems (hosts or routers) can communicate with each other.

# *Timestamp Request and Reply*

- Two machines (hosts or routers) can use the timestamp request and timestamp reply messages to determine the round-trip time needed for an IP datagram to travel between them.
- It can also be used to synchronize the clocks in two machines.



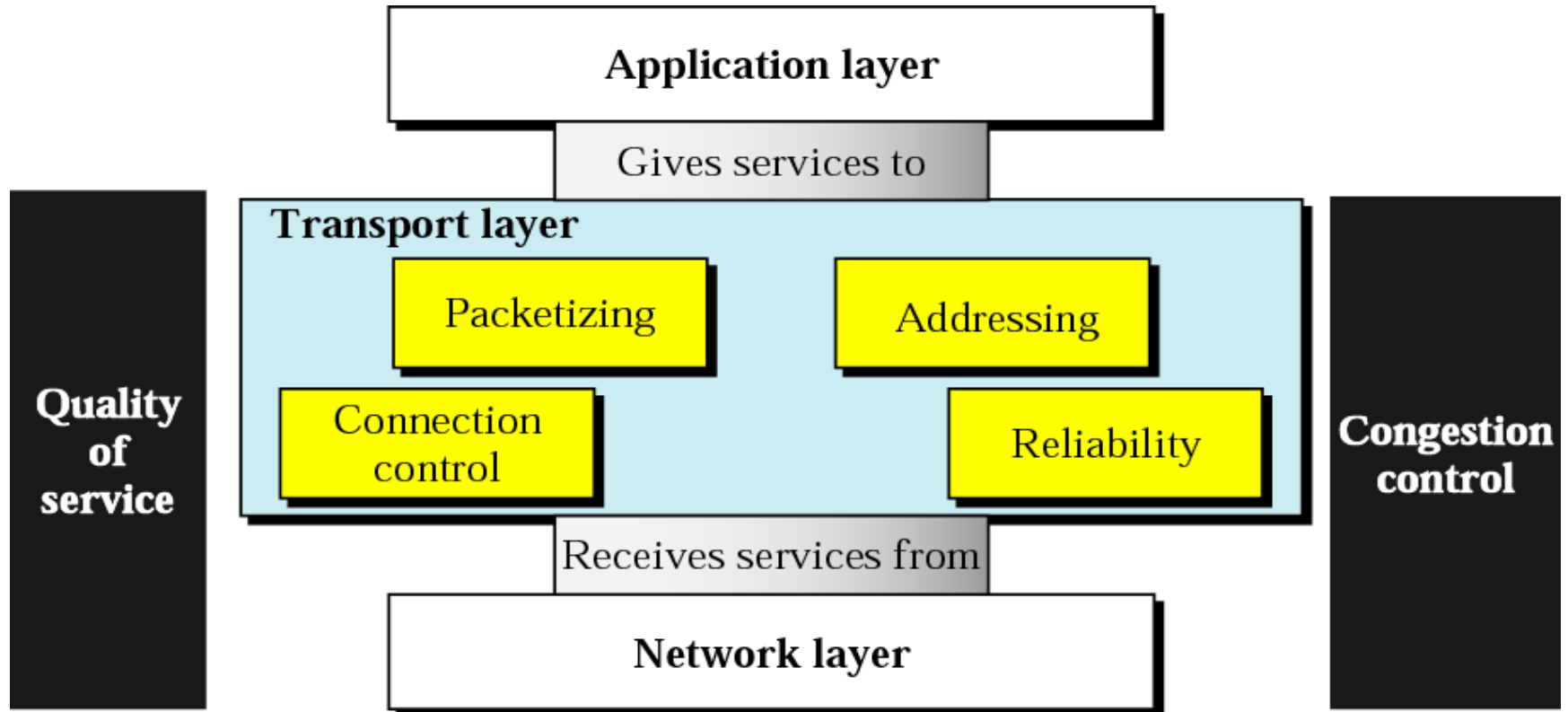
# *Address-Mask Request and Reply*

- A host may know its IP address, but it may not know the corresponding mask.
- For example, a host may know its IP address as 159.31.17.24, but it may not know that the corresponding mask is /16.
- To obtain its mask, a host sends an address-mask-request message to a router on the LAN.

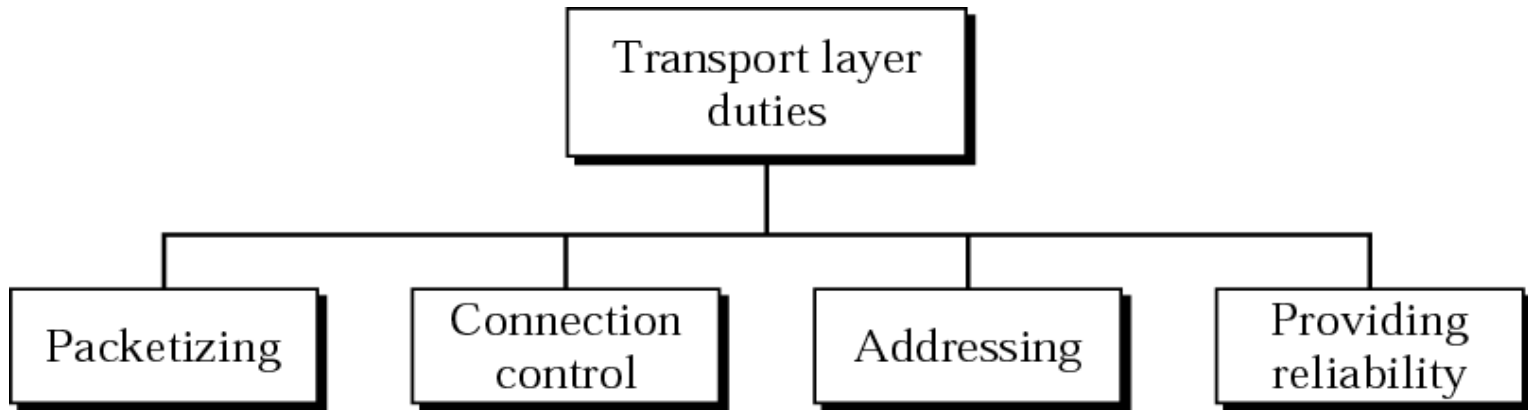
# *Router Solicitation and Advertisement*

- A host that wants to send data to a host on another network needs to know the address of routers connected to its own network.
- Also, the host must know if the routers are alive and functioning.
- The router-solicitation and router-advertisement messages can help in this situation.
- 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.

# Position of transport layer



# Transport layer duties



- The data link layer is responsible for delivery of frames between two neighboring nodes over a link. This is called *node-to-node delivery*.
- *The network layer is responsible for delivery of datagrams between two hosts. This is called host-to-host delivery.*
- *Communication* on the Internet is not defined as the exchange of data between two nodes or between two hosts.
- Real communication takes place between two processes (application programs). We need process-to-process delivery.
- However, at any moment, several processes may be running on the source host and several on the destination host.
- To complete the delivery, we need a mechanism to deliver data from one of these processes running on the source host to the corresponding process running on the destination host.

# Process-to-Process Delivery

*Client-Server Paradigm*

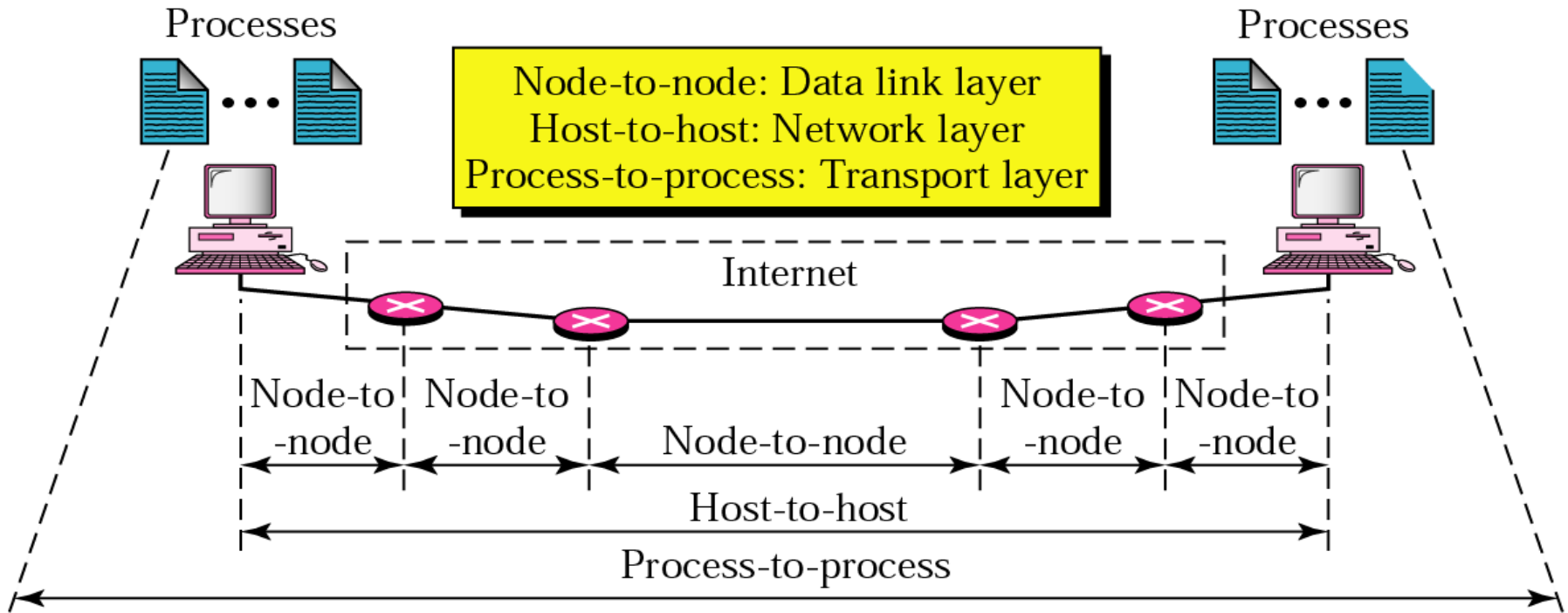
*Addressing*

*Multiplexing and Demultiplexing*

*Connectionless/Connection-Oriented*

*Reliable/Unreliable*

# Types of data deliveries

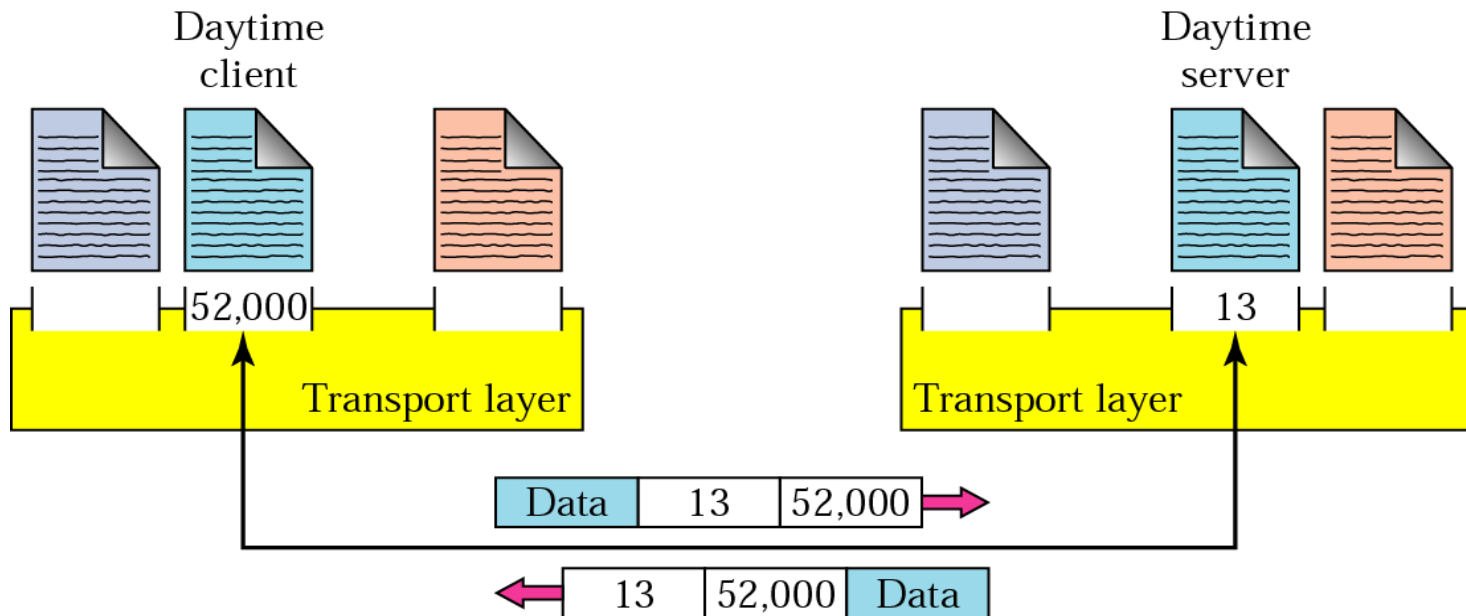


# Client/Server Paradigm

- A process on the local host, called a client, needs services from a process usually on the remote host, called a server.
- Both processes (client and server) have the same name.
- For example, to get the day and time from a remote machine, we need a Daytime client process running on the local host and a Daytime server process running on a remote machine.



# Port numbers



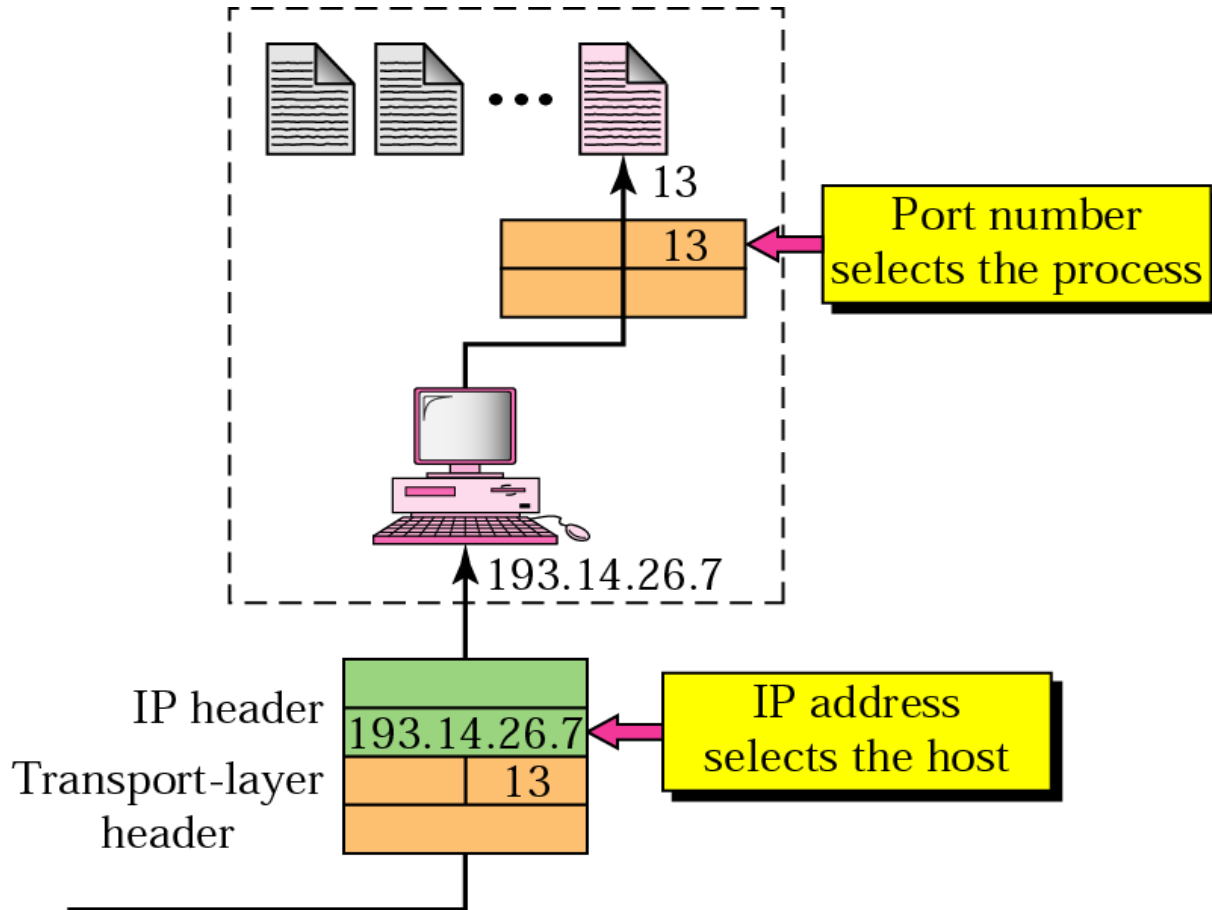
For communication,  
we must define the following:

- 1. Local host
- 2. Local process
- 3. Remote host
- 4. Remote process

# addressing

- A frame in the data link layer needs a destination MAC address for delivery and a source address for the next node's reply.
- At the network layer, we need an IP address to choose one host among millions. A datagram in the network layer needs a destination IP address for delivery and a source IP address for the destination's reply.
- At the transport layer, we need a transport layer address, called a port number, to choose among multiple processes running on the destination host.
- The destination port number is needed for delivery; the source port number is needed for the reply.
- In the Internet model, the port numbers are 16-bit integers between 0 and 65,535. The client program defines itself with a port number, chosen randomly by the transport layer software running on the client host. This is the ephemeral port number.
- The Internet has decided to use universal port numbers for servers; these are called well-known port numbers

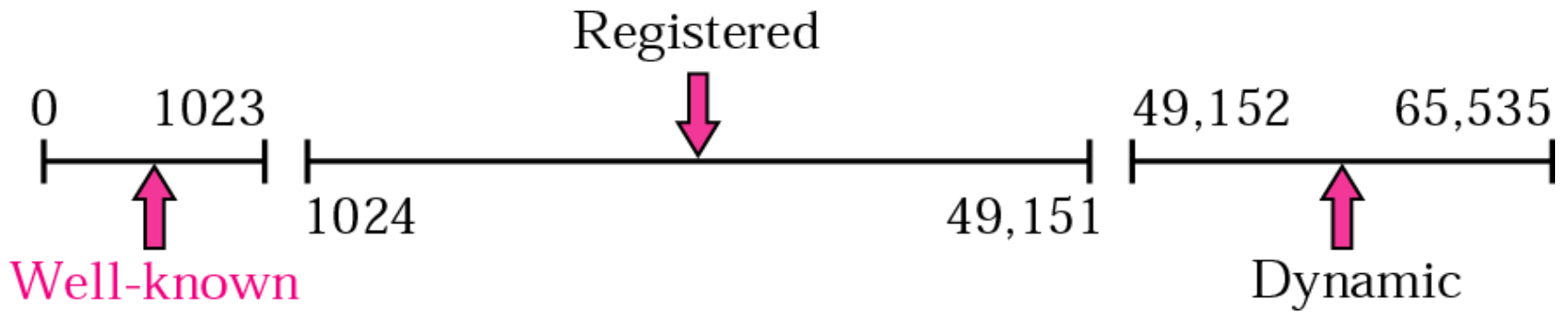
# IP addresses versus port numbers



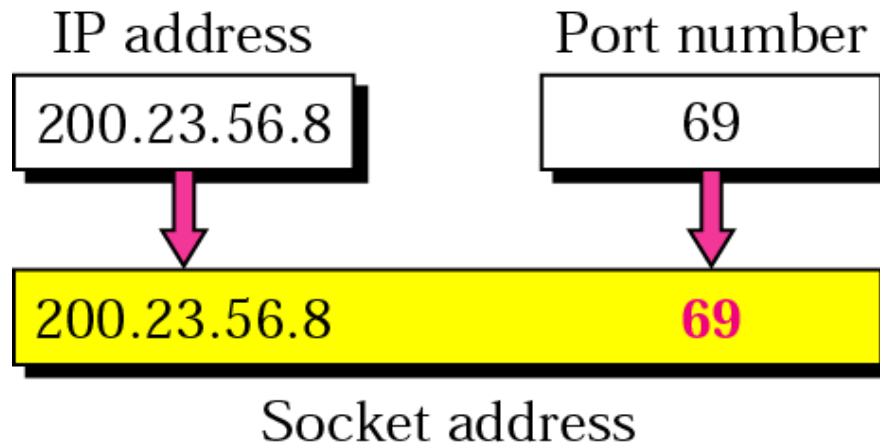
# *IANA Ranges*

- The IANA (Internet Assigned Number Authority) has divided the port numbers into three ranges:
  - well known,
  - registered,
  - dynamic (or private)
- Well-known ports. The ports ranging from 0 to 1023 are assigned and controlled by IANA. These are the **well-known ports**.
- Registered ports. The ports ranging from 1024 to 49,151 are not assigned or controlled by IANA. They can only be registered with IANA to prevent duplication.
- Dynamic ports. The ports ranging from 49,152 to 65,535 are neither controlled nor registered. They can be used by any process. These are the **ephemeral ports**.

# IANA ranges



# Socket address



# Socket Addresses

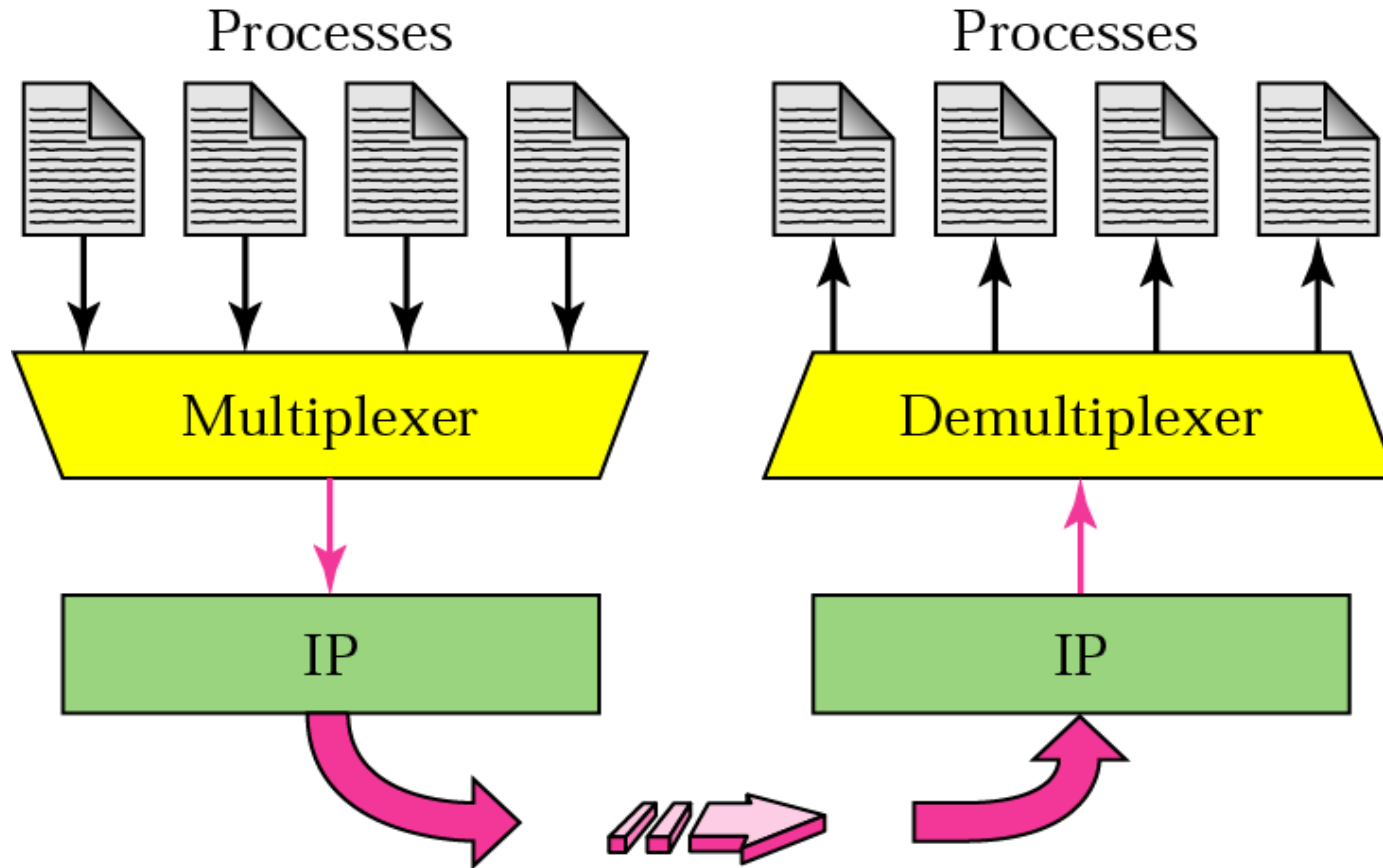
- Process-to-process delivery needs two identifiers, IP address and the port number, at each end to make a connection.
- The combination of an IP address and a port number is called a **socket address**.
- The client socket address defines the client process uniquely
- And the server socket address defines the server process uniquely
- A transport layer protocol needs a pair of socket addresses:
  - the client socket address
  - and the server socket address.
- These four pieces of information are part of the IP header and the transport layer protocol header.
- The IP header contains the IP addresses; the UDP or TCP header contains the port numbers.



# *Multiplexing*

- At the sender site, there may be several processes that need to send packets.
- However, there is only one transport layer protocol at any time.
- This is a many-to-one relationship and requires multiplexing.
- The protocol accepts messages from different processes, differentiated by their assigned port numbers.
- After adding the header, the transport layer passes the packet to the network layer.

# Multiplexing and demultiplexing



# *Demultiplexing*

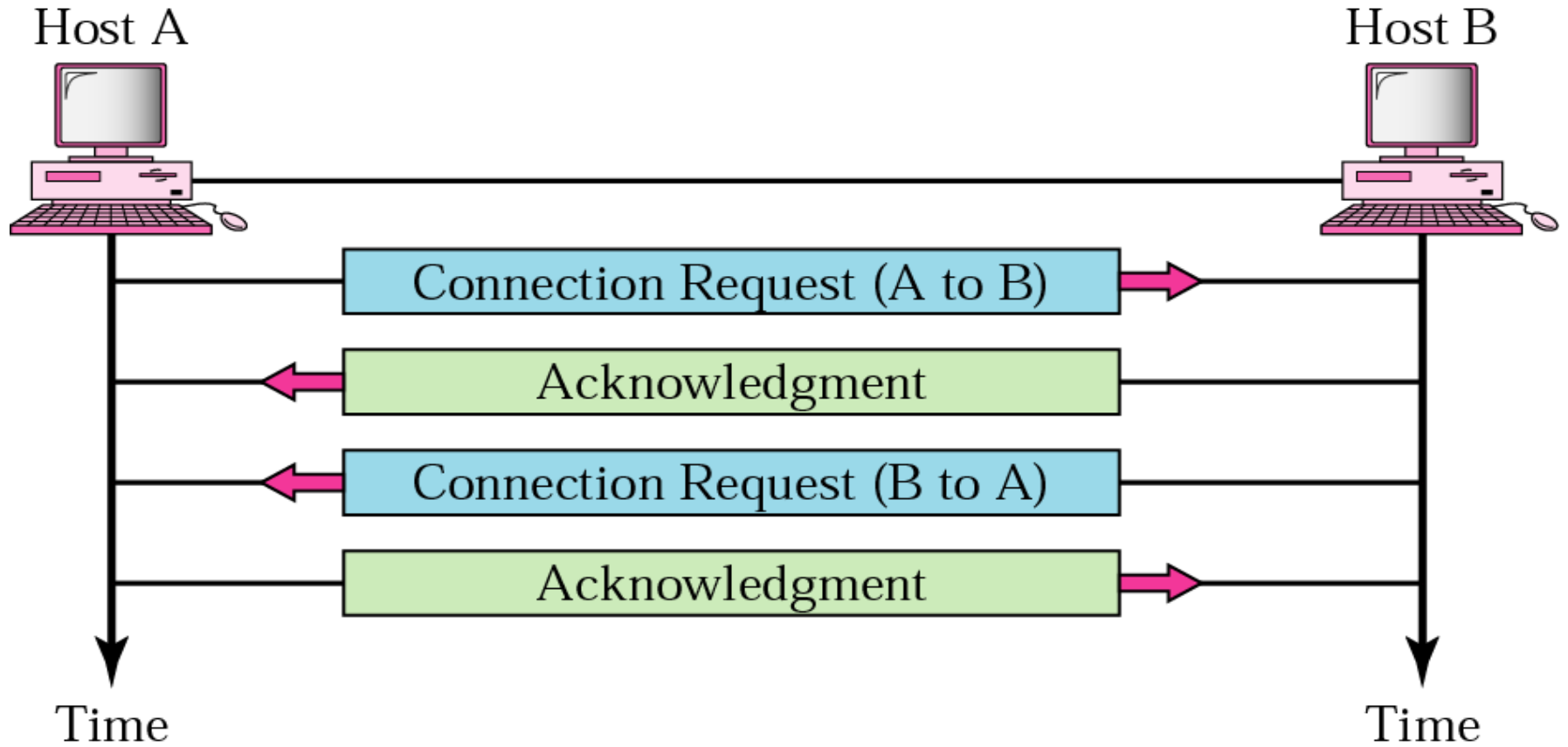
- At the receiver site, the relationship is one-to-many and requires demultiplexing.
- The transport layer receives datagrams from the network layer.
- After error checking and dropping of the header, the transport layer delivers each message to the appropriate process based on the port number.

# Connectionless Versus Connection-Oriented Service

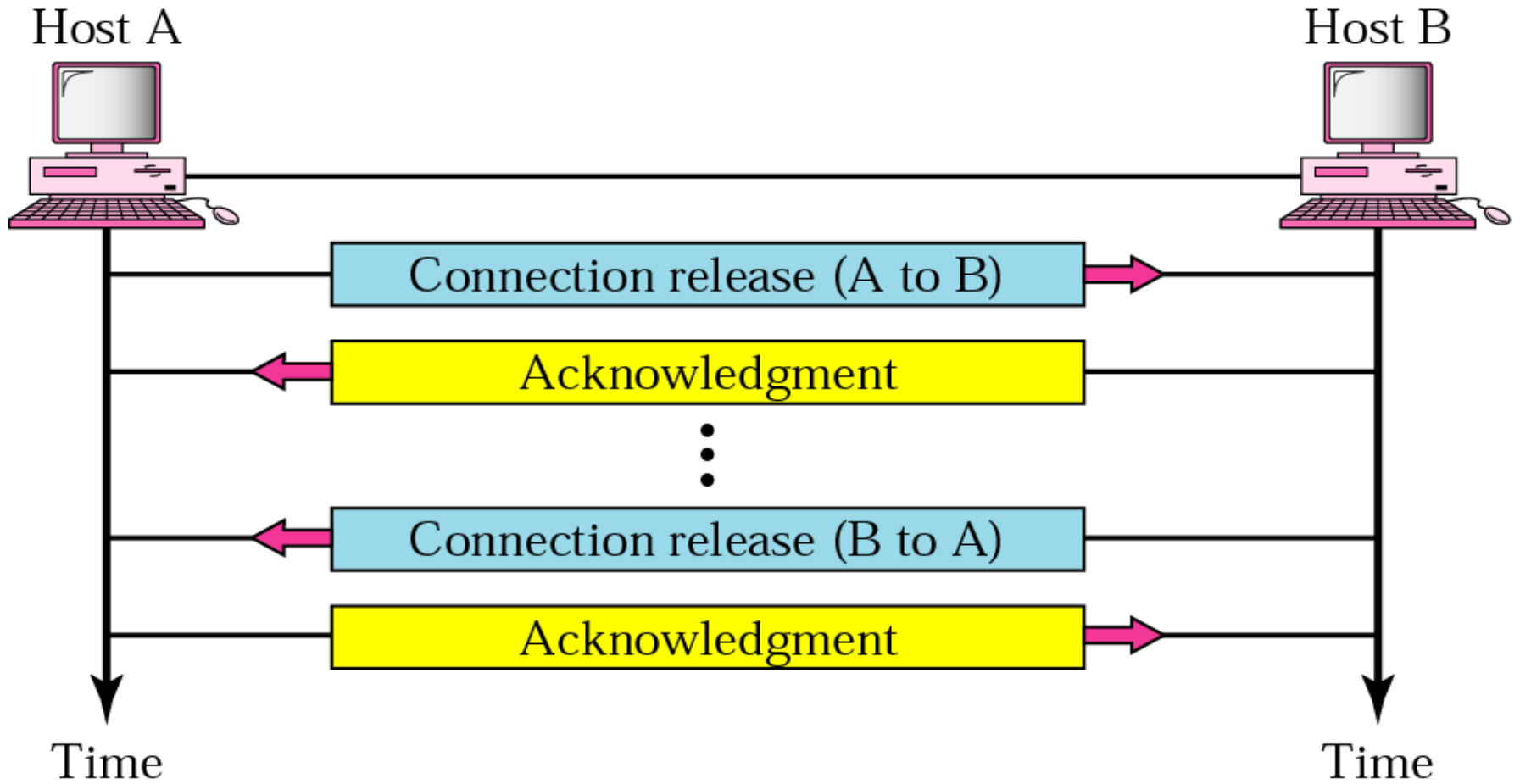
A transport layer protocol can either be connectionless or connection-oriented.

- ***Connectionless Service***
- In a connectionless service, the packets are sent from one party to another with no need for connection establishment or connection release.
- The packets are not numbered; they may be delayed or lost or may arrive out of sequence.
- There is no acknowledgment either.

# Connection establishment



# Connection termination

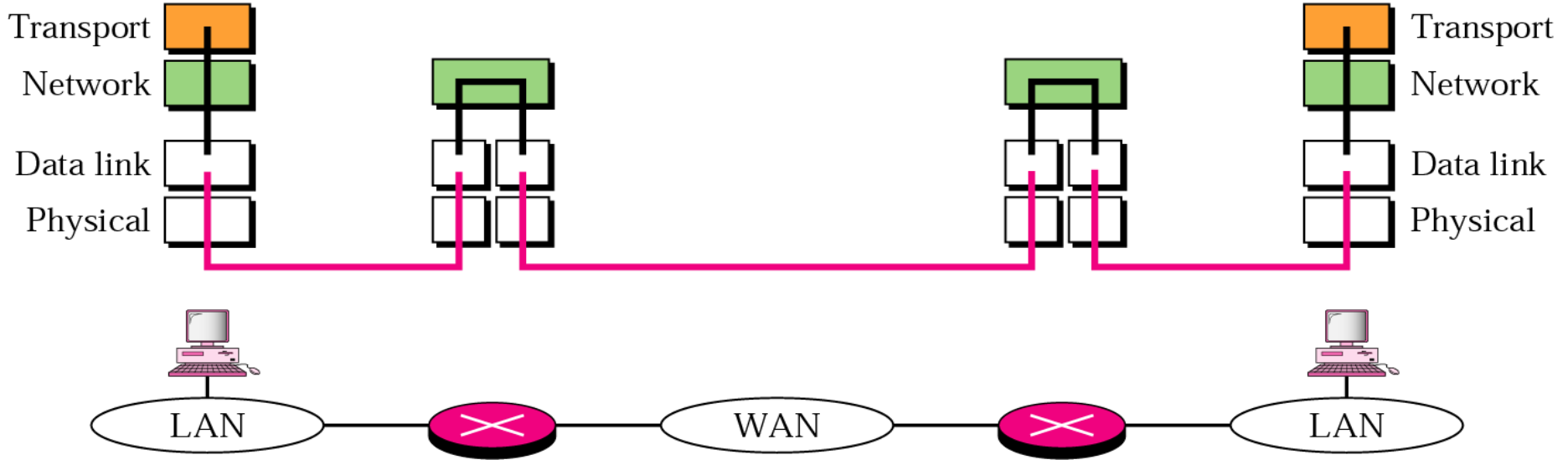


## **Connection-Oriented *Service***

- In a connection-oriented service, a connection is first established between the sender and the receiver.
- Data are transferred. At the end, the connection is released.

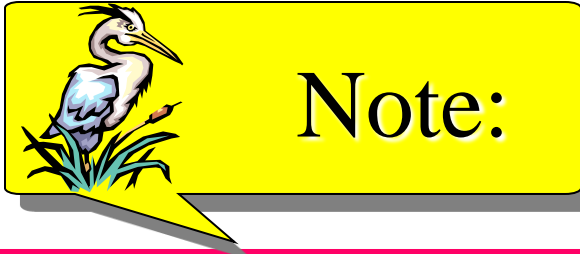
# Error control

— Error is checked in these paths by the data link layer  
— Error is not checked in these paths by the data link layer





# **UNIT -V**

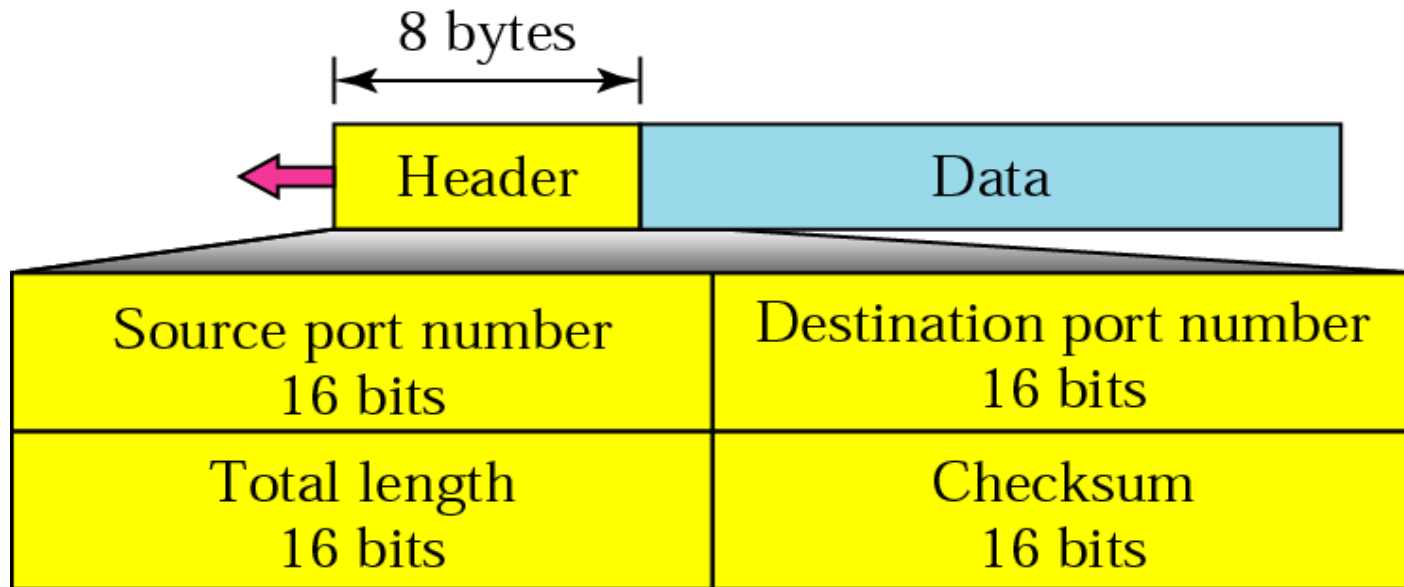


*UDP is a connectionless, unreliable protocol that has no flow and error control. It uses port numbers to multiplex data from the application layer.*

## *Well-known ports used by UDP*

<i>Port</i>	<i>Protocol</i>	<i>Description</i>
<b>7</b>	<b>Echo</b>	<b>Echoes a received datagram back to the sender</b>
<b>9</b>	<b>Discard</b>	<b>Discards any datagram that is received</b>
<b>11</b>	<b>Users</b>	<b>Active users</b>
<b>13</b>	<b>Daytime</b>	<b>Returns the date and the time</b>
<b>17</b>	<b>Quote</b>	<b>Returns a quote of the day</b>
<b>19</b>	<b>Chargen</b>	<b>Returns a string of characters</b>
<b>53</b>	<b>Nameserver</b>	<b>Domain Name Service</b>
<b>67</b>	<b>Boots</b>	<b>Server port to download bootstrap information</b>
<b>68</b>	<b>Bootpc</b>	<b>Client port to download bootstrap information</b>
<b>69</b>	<b>TFTP</b>	<b>Trivial File Transfer Protocol</b>
<b>111</b>	<b>RPC</b>	<b>Remote Procedure Call</b>
<b>123</b>	<b>NTP</b>	<b>Network Time Protocol</b>
<b>161</b>	<b>SNMP</b>	<b>Simple Network Management Protocol</b>
<b>162</b>	<b>SNMP</b>	<b>Simple Network Management Protocol (trap)</b>

# User datagram format



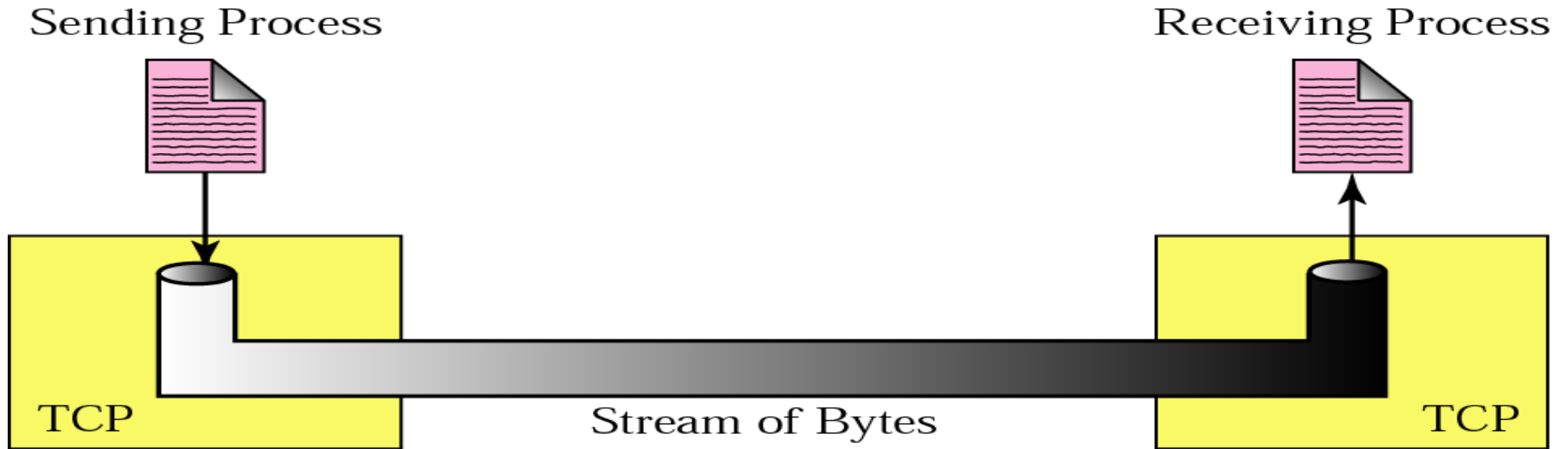


*The calculation of checksum and its inclusion in the user datagram are optional.*



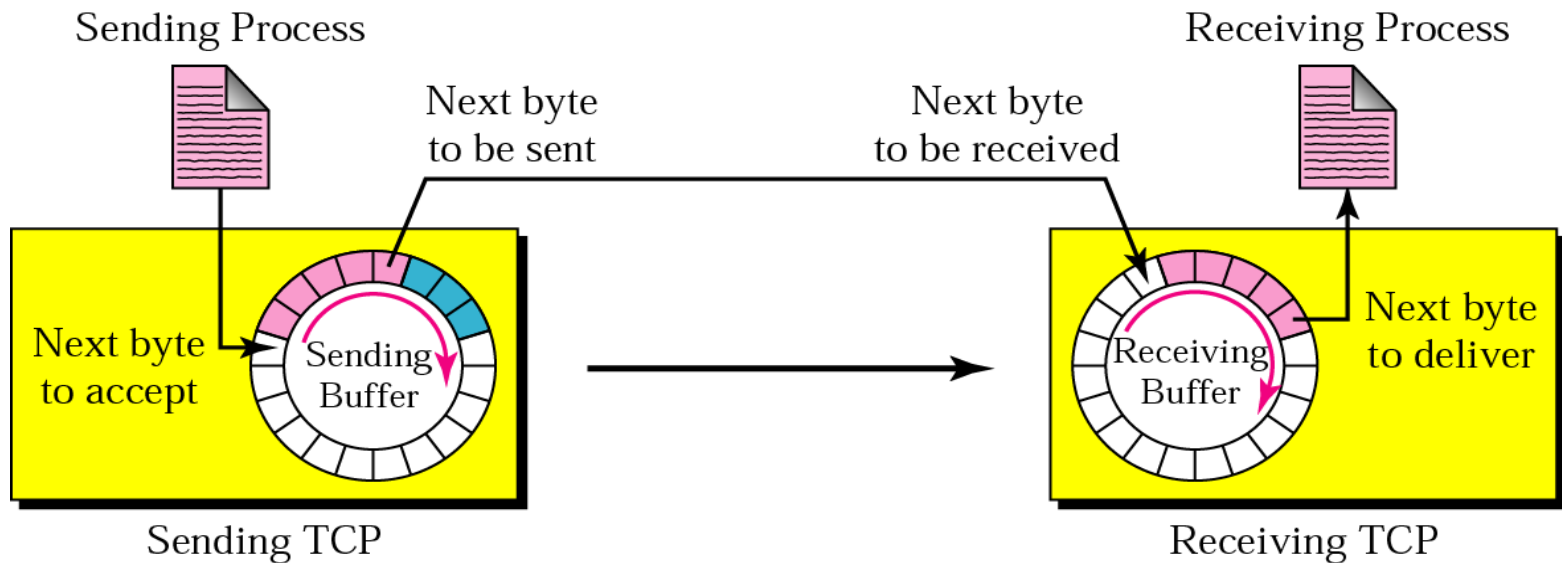
*UDP is a convenient transport-layer protocol for applications that provide flow and error control. It is also used by multimedia applications.*

## Stream delivery



- In udp process snds msgs with predefined boundaries to udp for delivery, but tcp allows sndng process to deliver data as a stream of bytes and allows revng process to obtain data as stream of bytes.

- Bcoz sndng & rcvng process may not write r read data at same speed ,tcp needs buffers for storage.

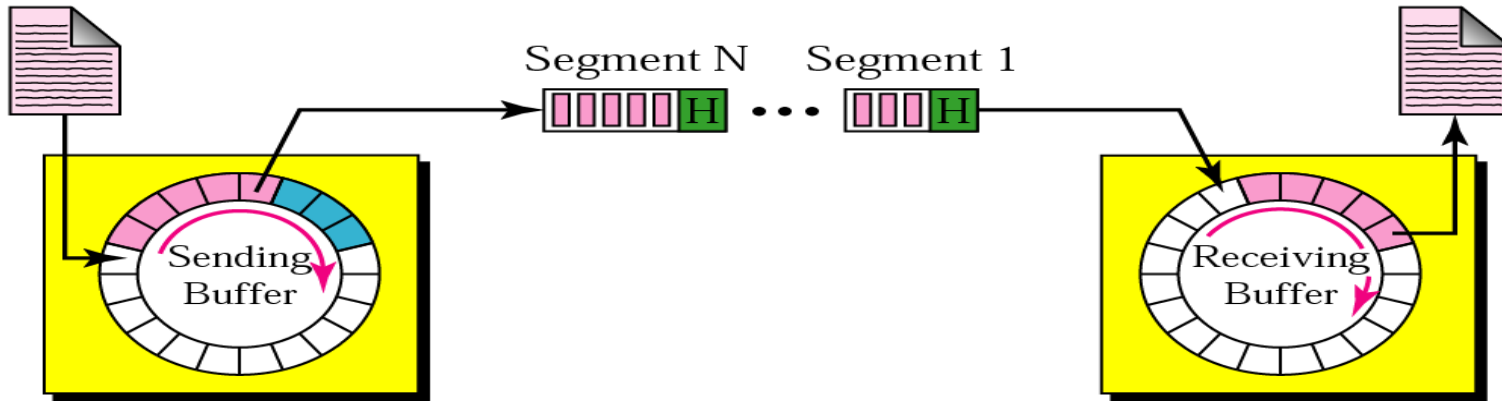




# TCP segments

Sending Process

Receiving Process



- At TL, TCP groups no. of bytes together into packet and adds header to each segment and delivers segment to ip layer for transmission. segments are encapsulated in ip datagram.
- Tcp supports full duplex, connection oriented, reliable service.

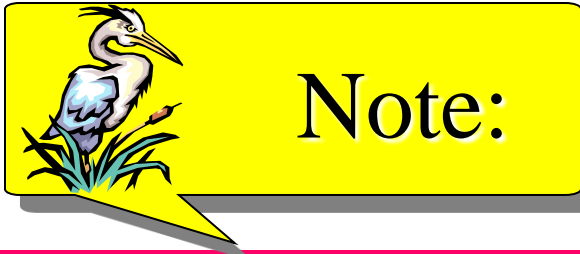
## *Example 1*

Imagine a TCP connection is transferring a file of 6000 bytes. The first byte is numbered 10010. What are the sequence numbers for each segment if data are sent in five segments with the first four segments carrying 1000 bytes and the last segment carrying 2000 bytes?

## *Solution*

The following shows the sequence number for each segment:

- Segment 1 ==> sequence number: 10,010 (range: 10,010 to 11,009)
- Segment 2 ==> sequence number: 11,010 (range: 11,010 to 12,009)
- Segment 3 ==> sequence number: 12,010 (range: 12,010 to 13,009)
- Segment 4 ==> sequence number: 13,010 (range: 13,010 to 14,009)
- Segment 5 ==> sequence number: 14,010 (range: 14,010 to 16,009)



*The bytes of data being transferred in each connection are numbered by TCP. The numbering starts with a randomly generated number.*



*The value of the sequence number field in a segment defines the number of the first data byte contained in that segment.*

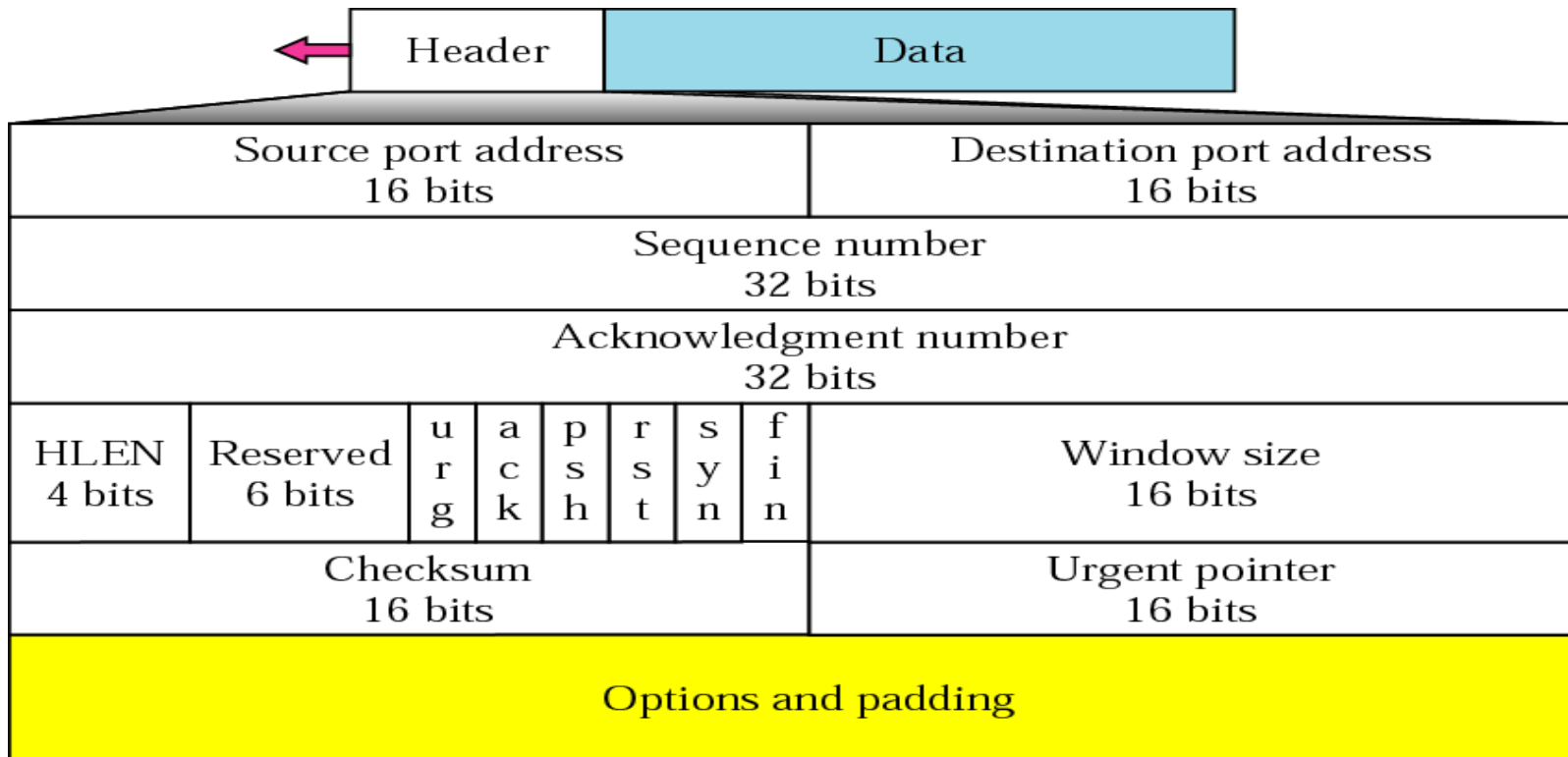


*The value of the acknowledgment field in a segment defines the number of the next byte a party expects to receive.*

*The acknowledgment number is cumulative.*

# TCP segment format

- Segment consists of 20-60 byte header followed by data from appn prgm.



## Control field

URG: Urgent pointer is valid  
ACK: Acknowledgment is valid  
PSH: Request for push

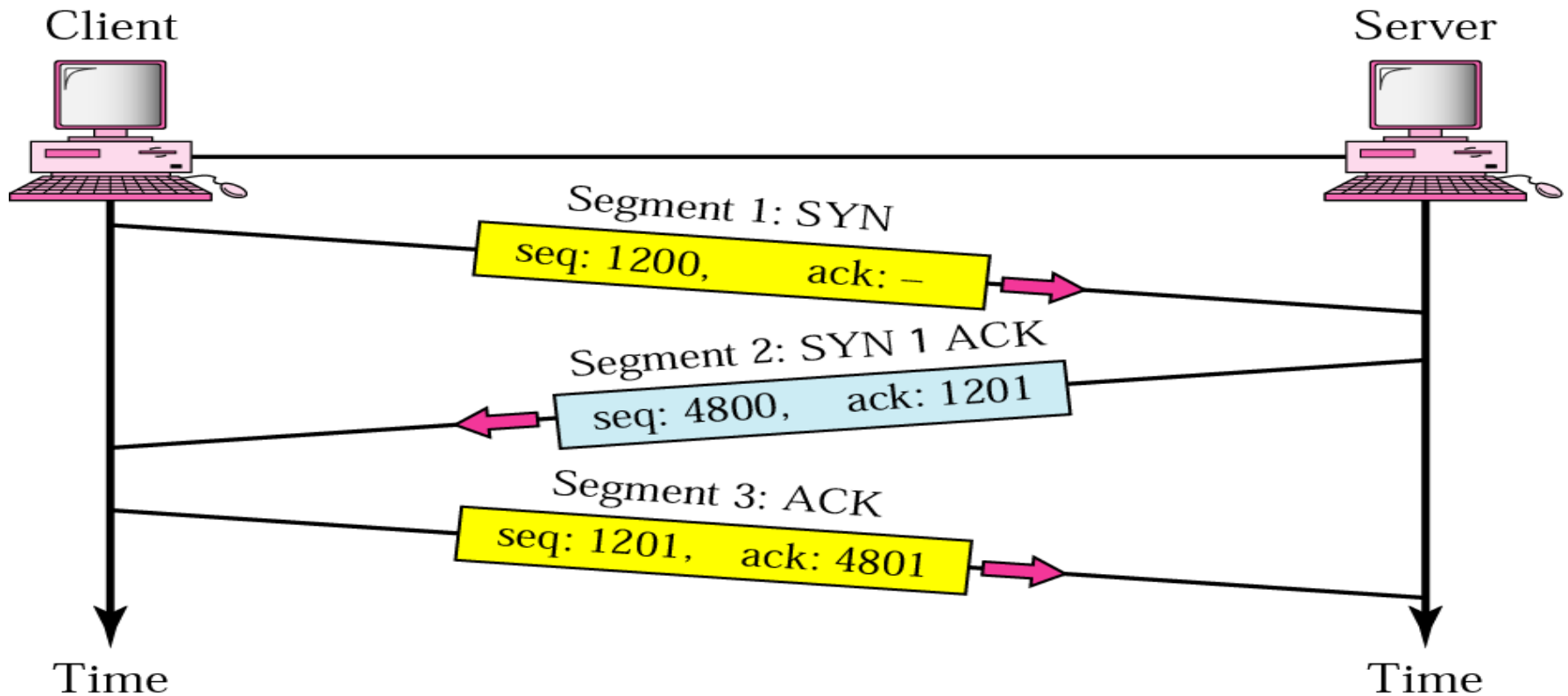
RST: Reset the connection  
SYN: Synchronize sequence numbers  
FIN: Terminate the connection

URG	ACK	PSH	RST	SYN	FIN
-----	-----	-----	-----	-----	-----

Flag	Description
<b>URG</b>	The value of the urgent pointer field is valid.
<b>ACK</b>	The value of the acknowledgment field is valid.
<b>PSH</b>	Push the data.
<b>RST</b>	The connection must be reset.
<b>SYN</b>	Synchronize sequence numbers during connection.
<b>FIN</b>	Terminate the connection.

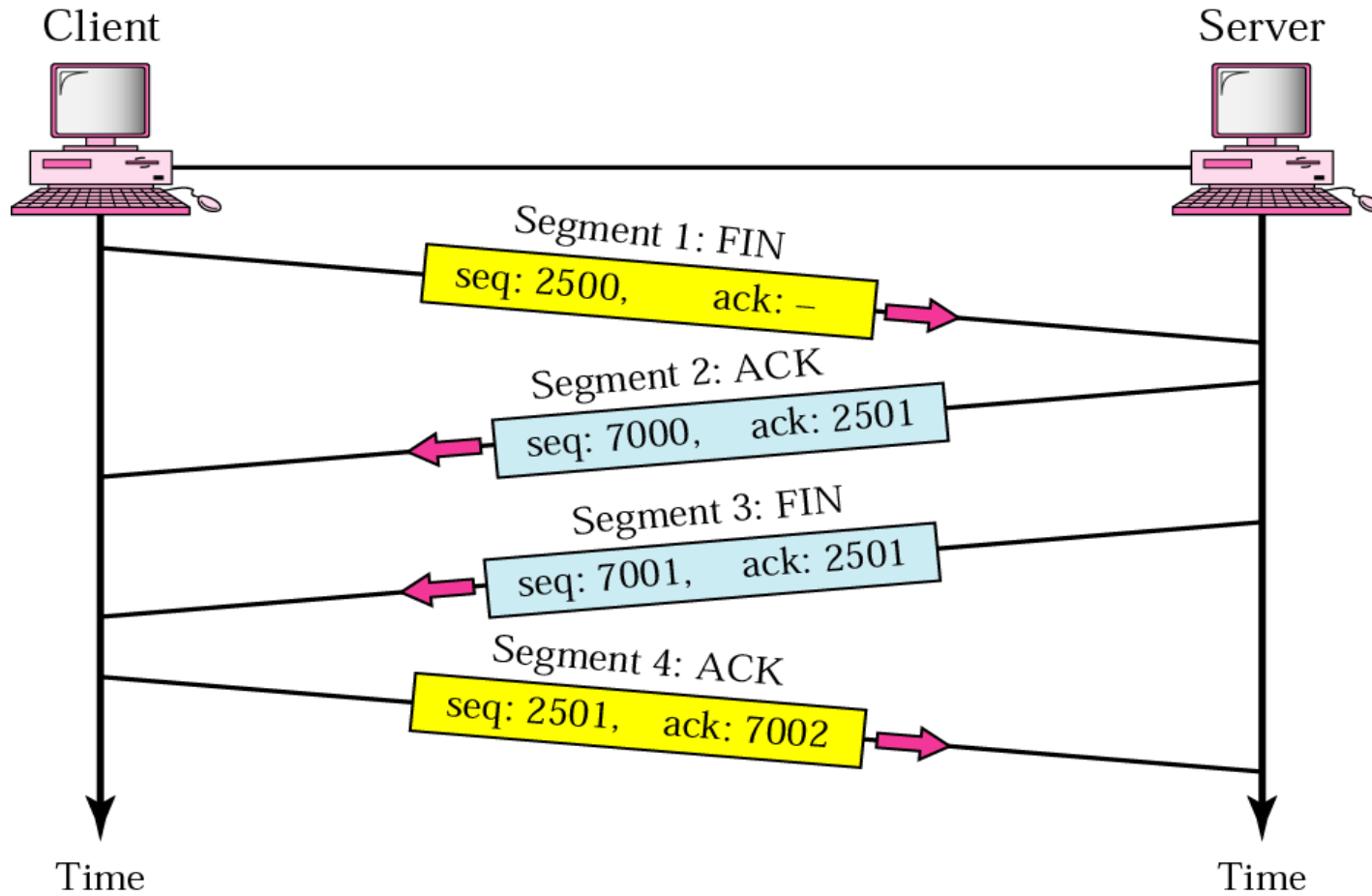
## Three-step connection establishment

- In tcp connected oriented transmission requires 3 phases-connection establishment,data transfer,connection termination.





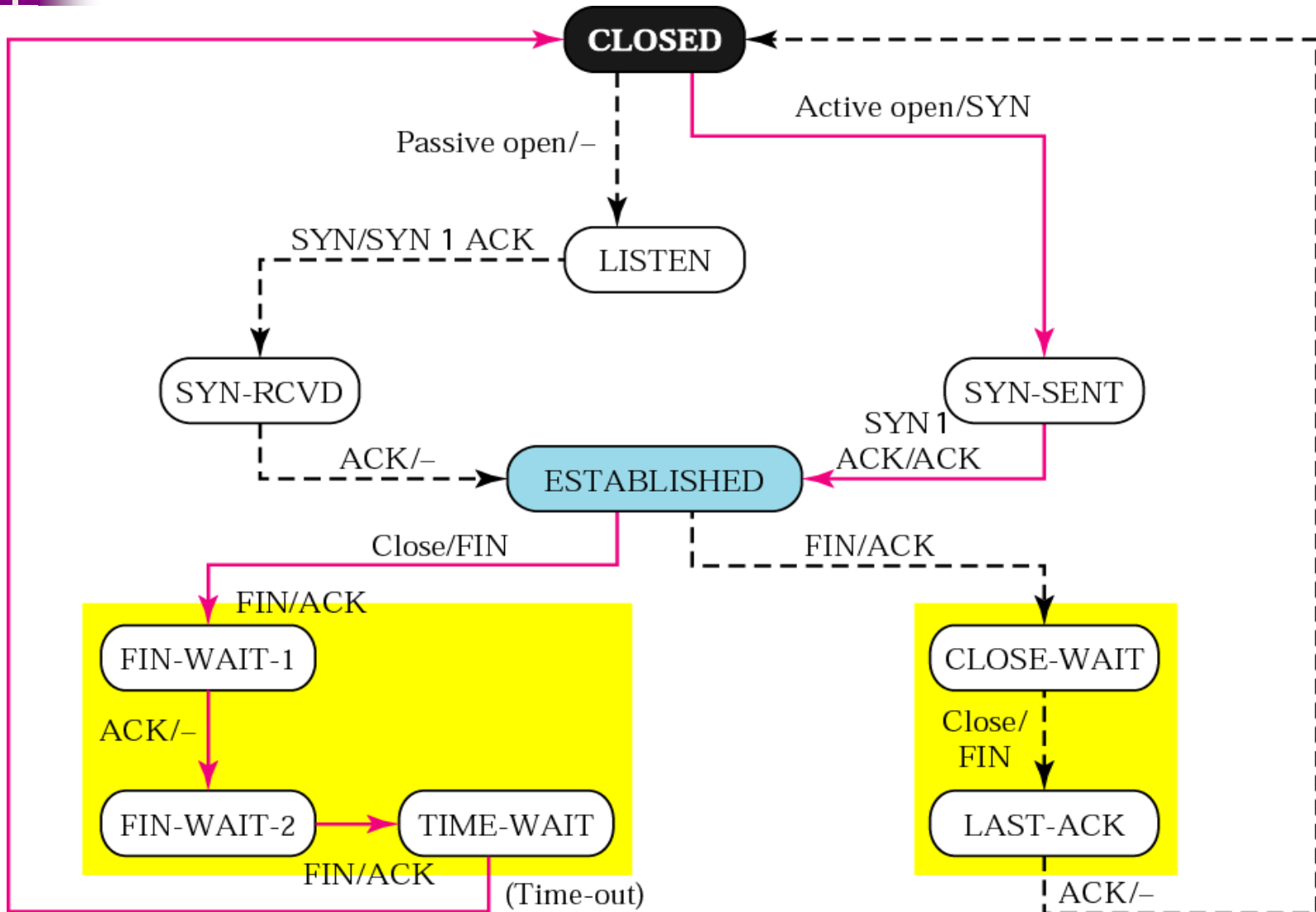
# Four-step connection termination

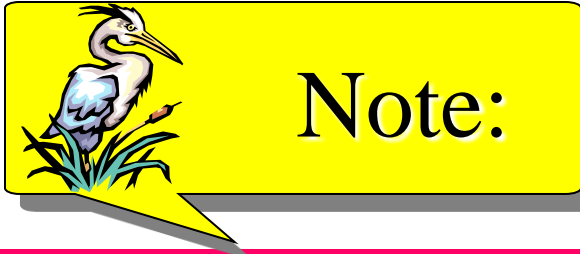


## *States for TCP*

<b>State</b>	<b>Description</b>
<b>CLOSED</b>	There is no connection.
<b>LISTEN</b>	The server is waiting for calls from the client.
<b>SYN-SENT</b>	A connection request is sent; waiting for acknowledgment.
<b>SYN-RCVD</b>	A connection request is received.
<b>ESTABLISHED</b>	Connection is established.
<b>FIN-WAIT-1</b>	The application has requested the closing of the connection.
<b>FIN-WAIT-2</b>	The other side has accepted the closing of the connection.
<b>TIME-WAIT</b>	Waiting for retransmitted segments to die.
<b>CLOSE-WAIT</b>	The server is waiting for the application to close.
<b>LAST-ACK</b>	The server is waiting for the last acknowledgment.

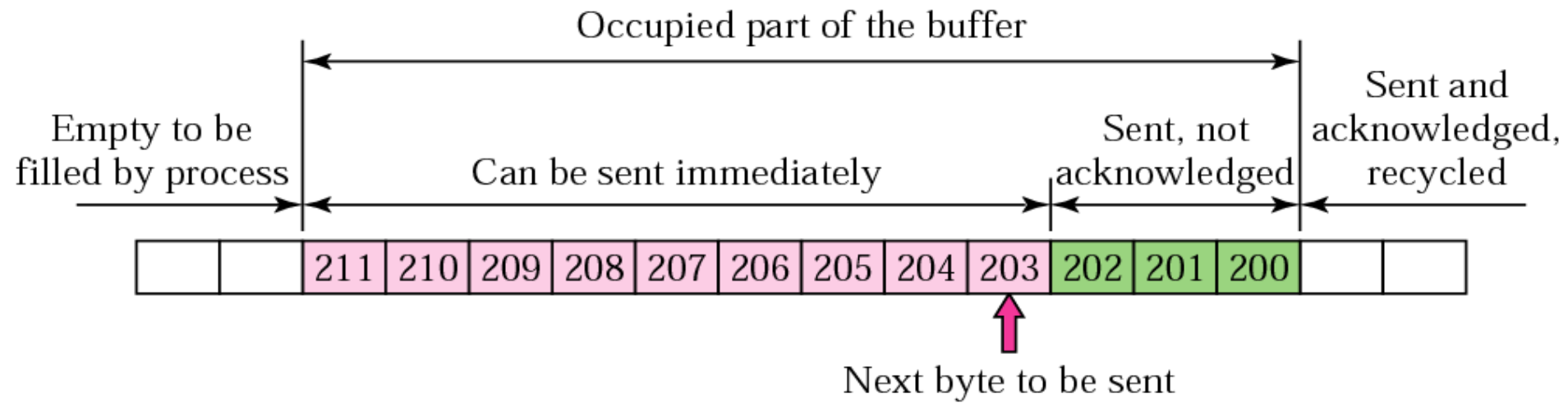
# State transition diagram



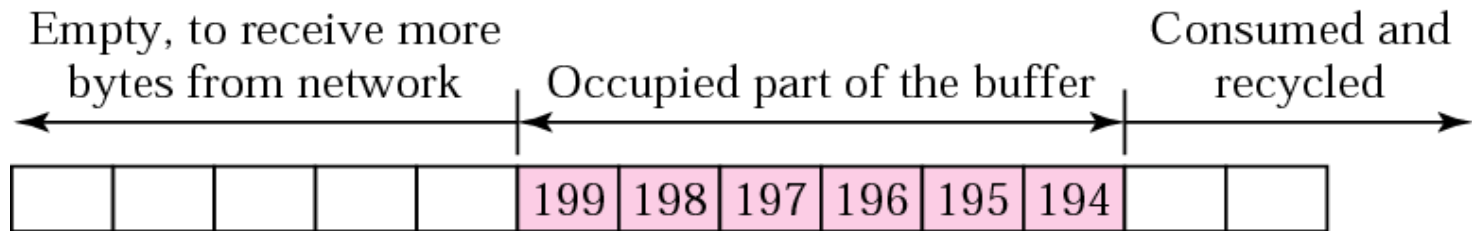


*A sliding window is used to make transmission more efficient as well as to control the flow of data so that the destination does not become overwhelmed with data. TCP's sliding windows are byte-oriented.*

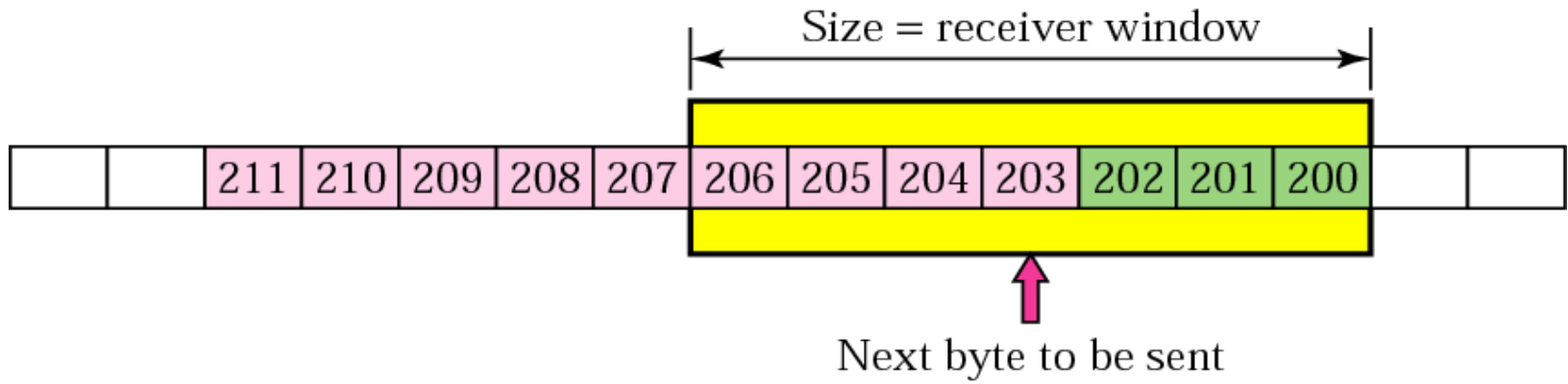
# Sender buffer



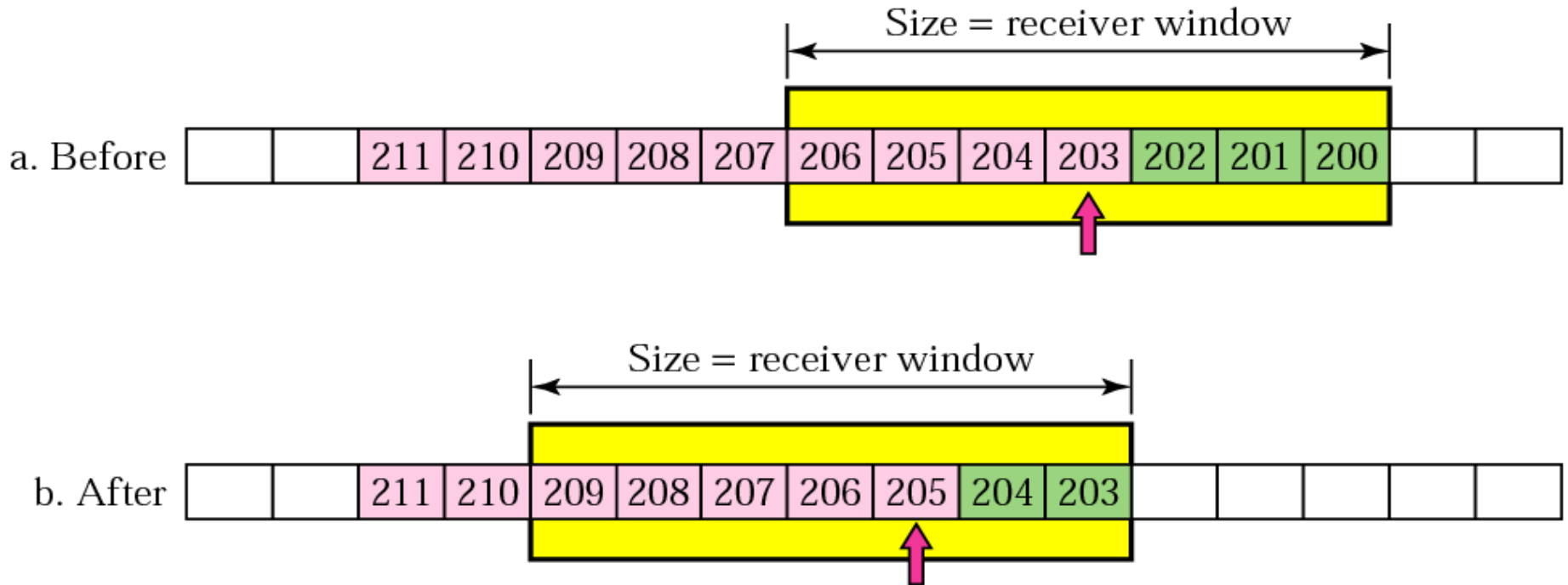
# Receiver window



# Sender buffer and sender window

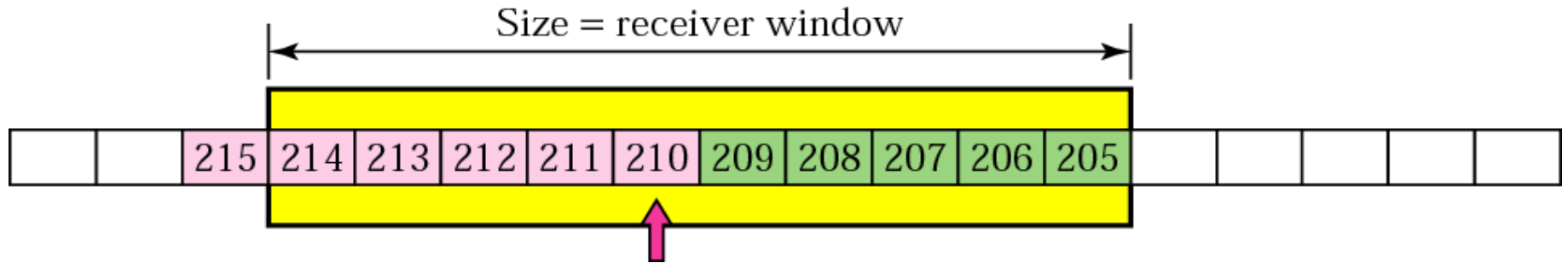


# Sliding the sender window

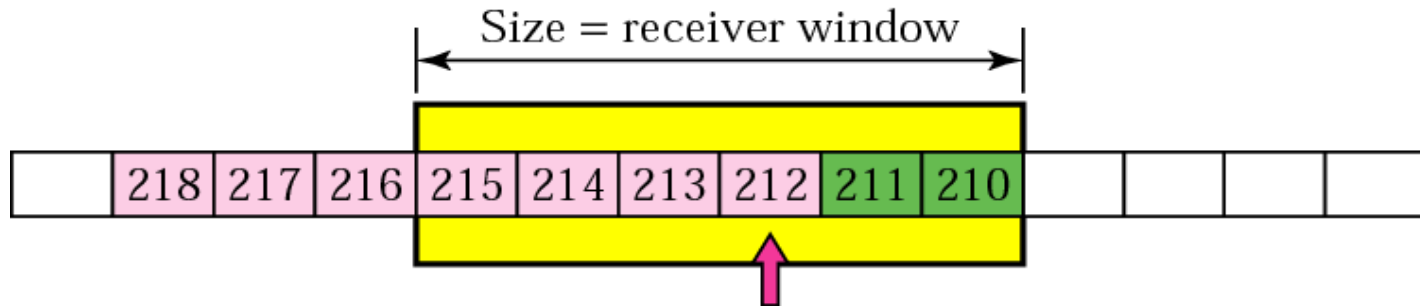


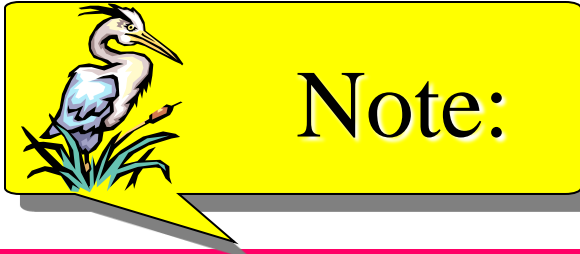


# Expanding the sender window



# Shrinking the sender window





Note:

*In TCP, the sender window size is totally controlled by the receiver window value (the number of empty locations in the receiver buffer).*

*However, the actual window size can be smaller if there is congestion in the network.*



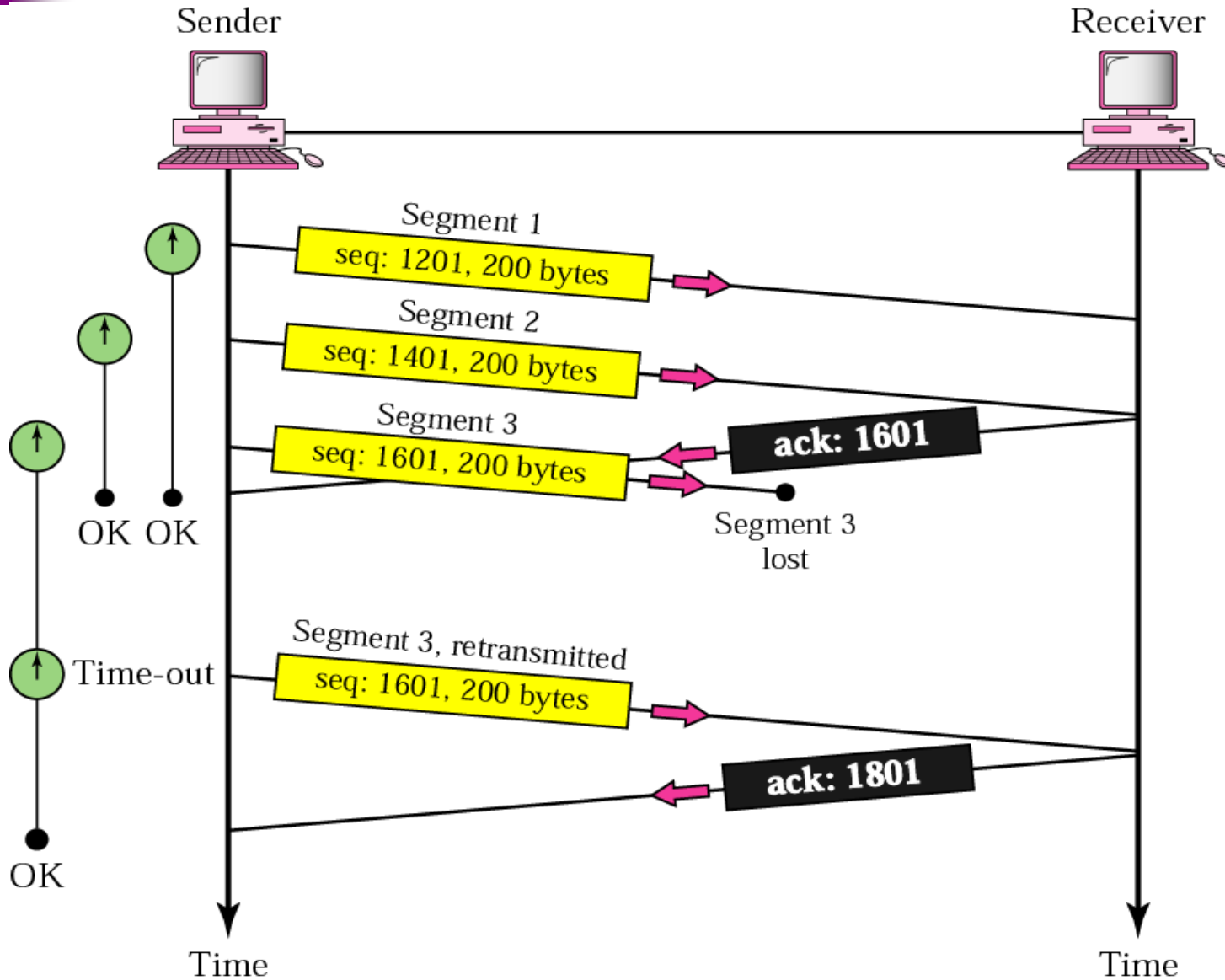
Note:

## *Some points about TCP's sliding windows:*

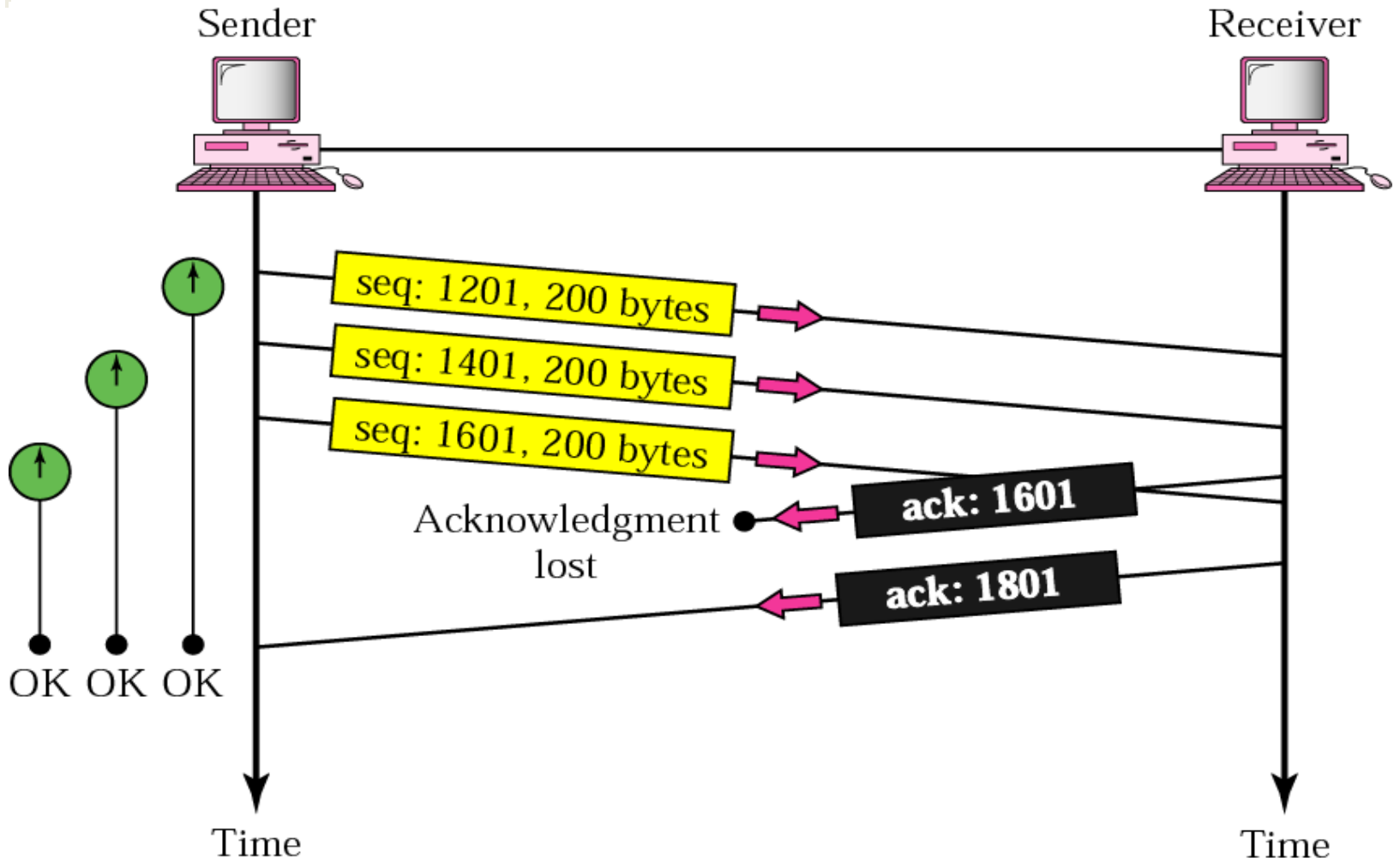
- The source does not have to send a full window's worth of data.
- The size of the window can be increased or decreased by the destination.

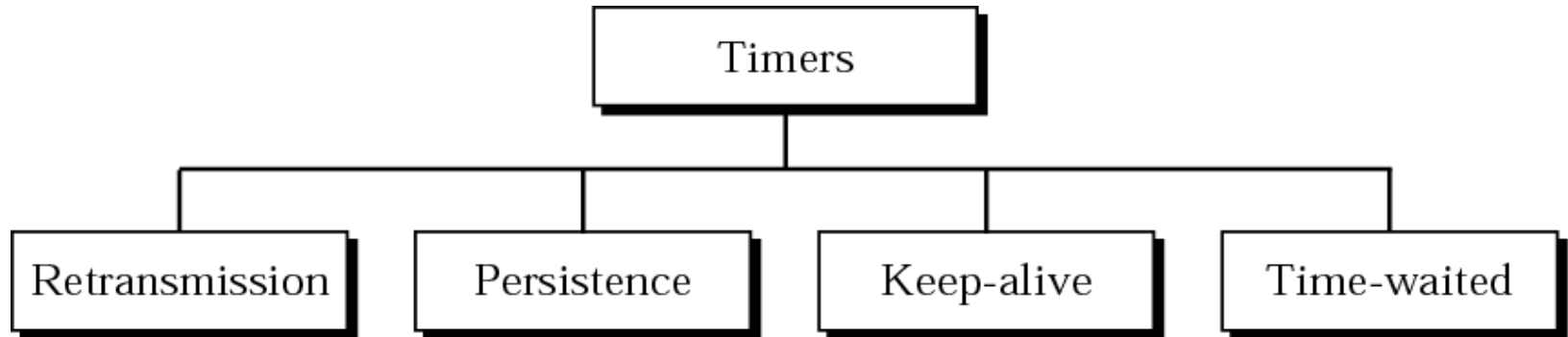


# Lost segment



# Lost acknowledgment



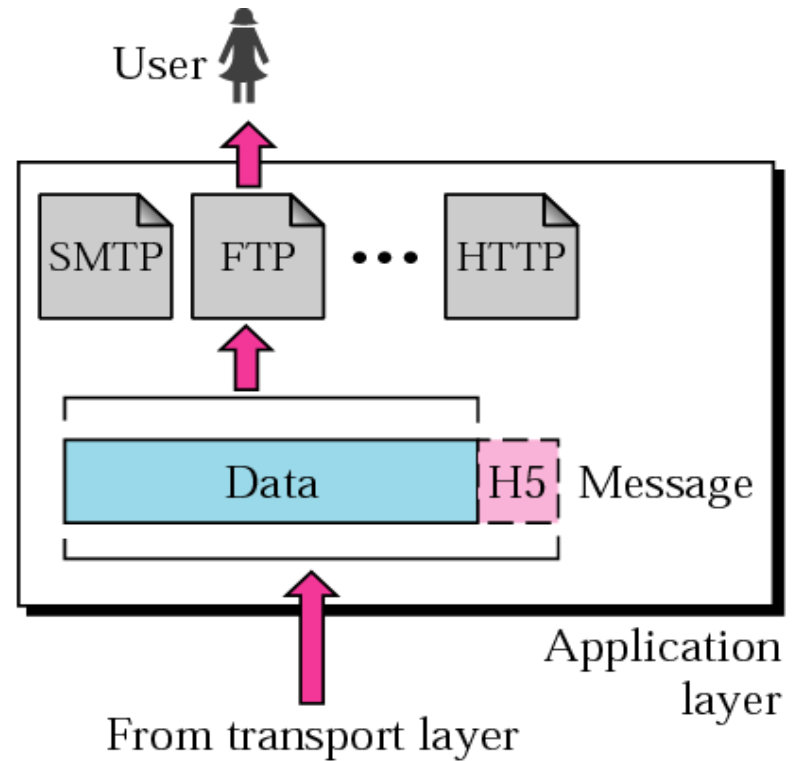
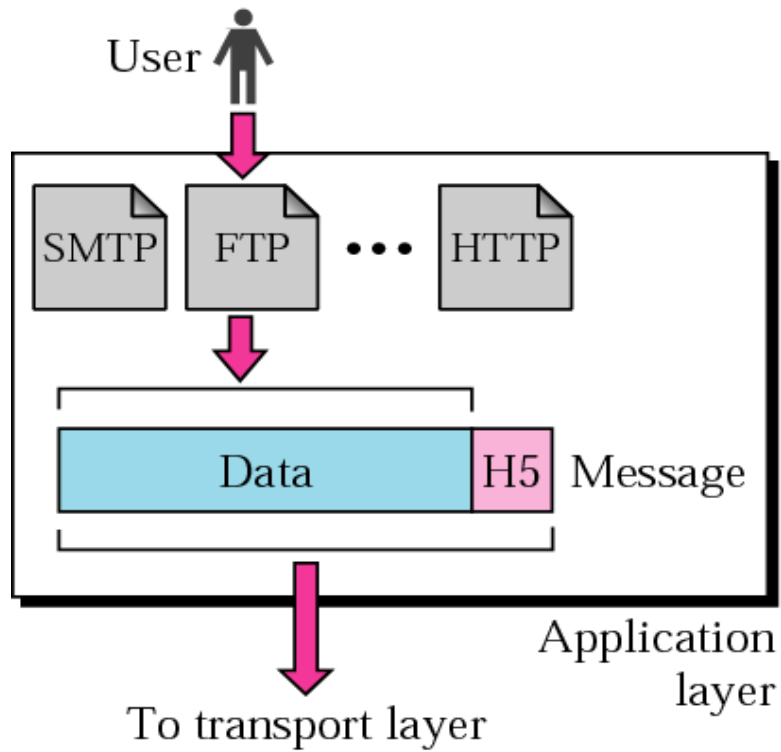


# Objectives

- 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,
  - network management.



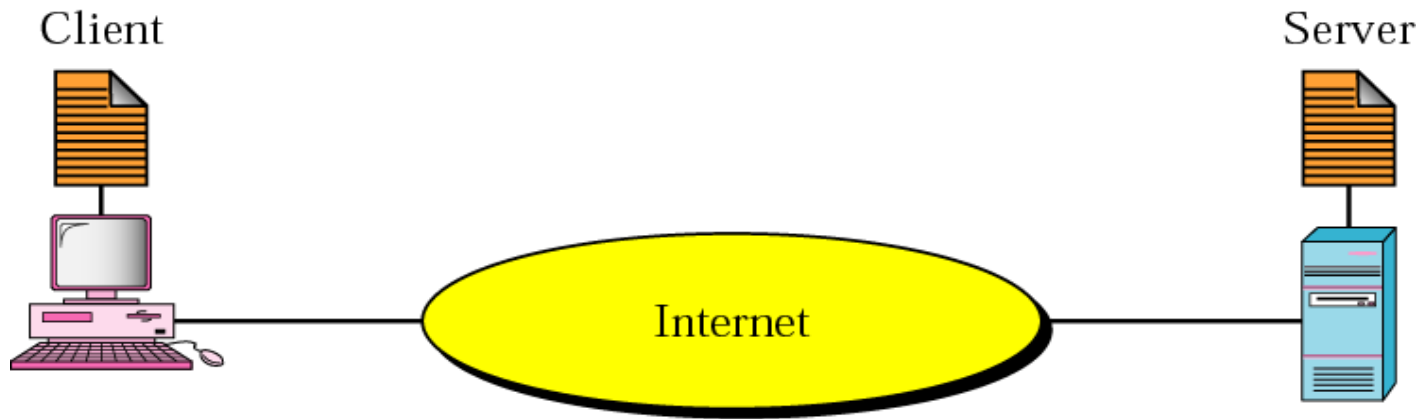
# Network layer duties



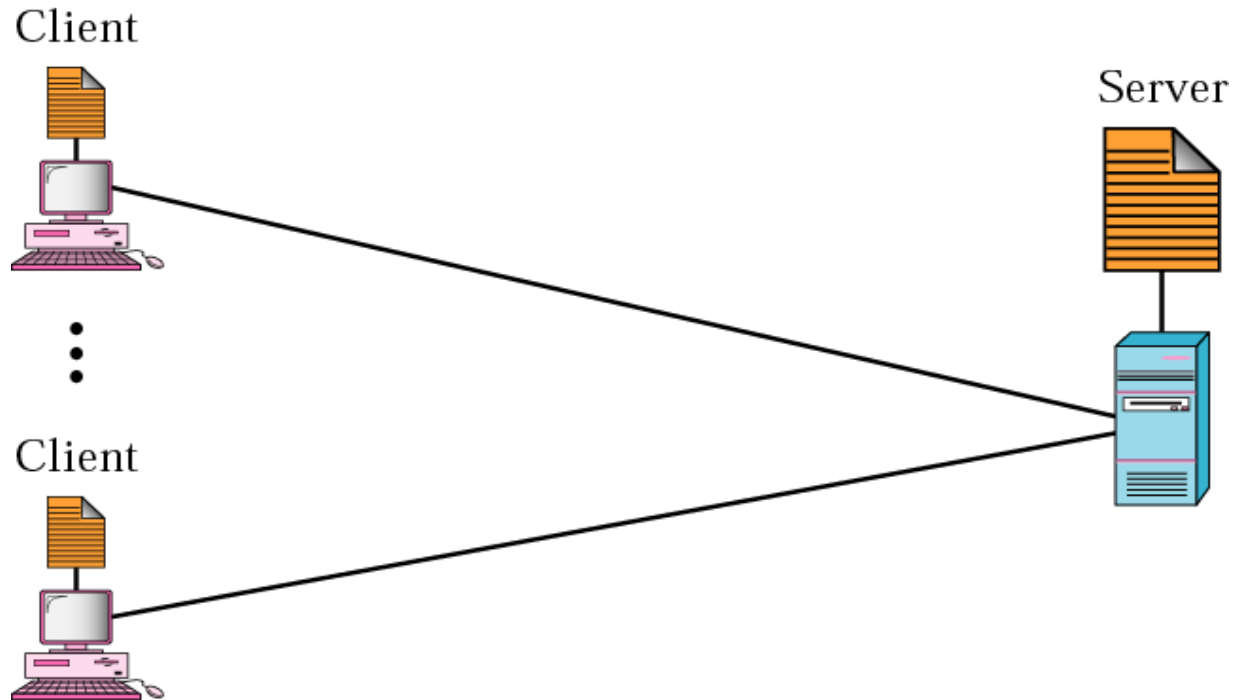
# Client-Server Model

- There are several ways that a computer can ask for the services of another computer the most common is the client-server model.
- *Relationship*
  - The purpose of a network is to provide services to users
  - a user at local site wants to receive a service from a computer at a remote site.
  - Two computers connected by an internet, must each run a program one to provide a service and the other to request a service.

# Client-server model



# Client-server relationship



# Client-Server Model

- **Client** application program running on the **local machine** requests a service from another application program – server – running on the **remote** machine.
- Commonly server provides service to any client, not a particular client
- Generally, a **client** application program that requests a service should run **only when** it is **needed**.
- A **server** program providing service should run **all the time**, as it does not know when its services will be needed.

# Client-Server Model

- A **client opens** the communication channel using IP address of the remote host and the port address of the specific server program running on the host – **Active open**.
- Request-response may be repeated several times, the process is **finite**.
- The client closes the communication channel with an **Active close**.

# Client-Server Model

- A server program opens its door for incoming requests from clients but never initiates a service unless explicitly requested – **Passive open**.
- A **server** program is **infinite** – runs unless a problem occurs.
- Clients can be run on a machine either
  - **iteratively**  
or
  - **concurrently**.
- **Concurrency in client:**
- **concurrency:** two or more clients can run at the same time on a machine – current trend,
- **alternatively:** one client must start, run, and terminate before another client may start (**iterative**).

# Client-Server Model

- **Concurrency in servers:**
- **An iterative server** can process only one request at a time
- a **concurrent server** can process many requests at the same time – **share** its time.



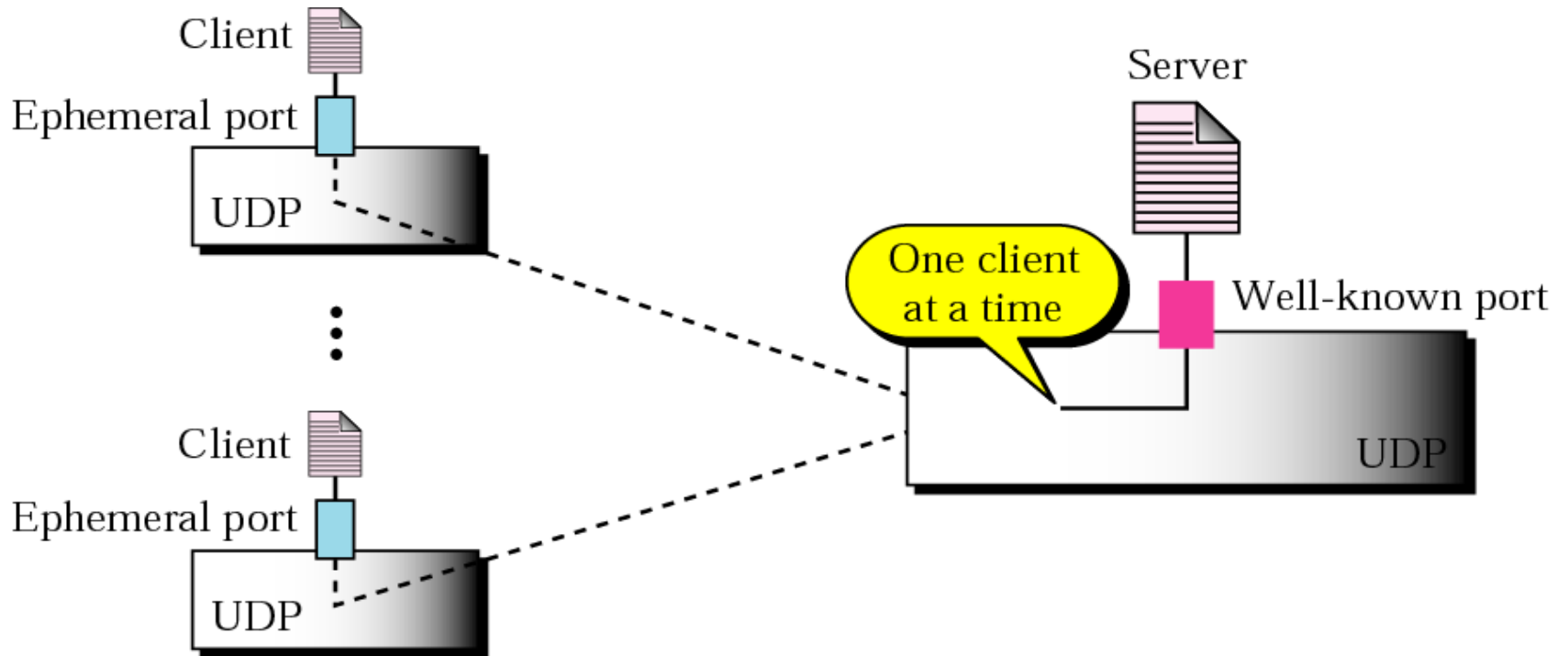
# Client-Server Model

- The servers use either UDP, a connectionless transport layer protocol, or TCP a connection-oriented transport protocol.
- There are four types of servers:
  - Connectionless iterative server
  - Connectionless concurrent server
  - Connection-oriented iterative server
  - Connection-oriented concurrent server

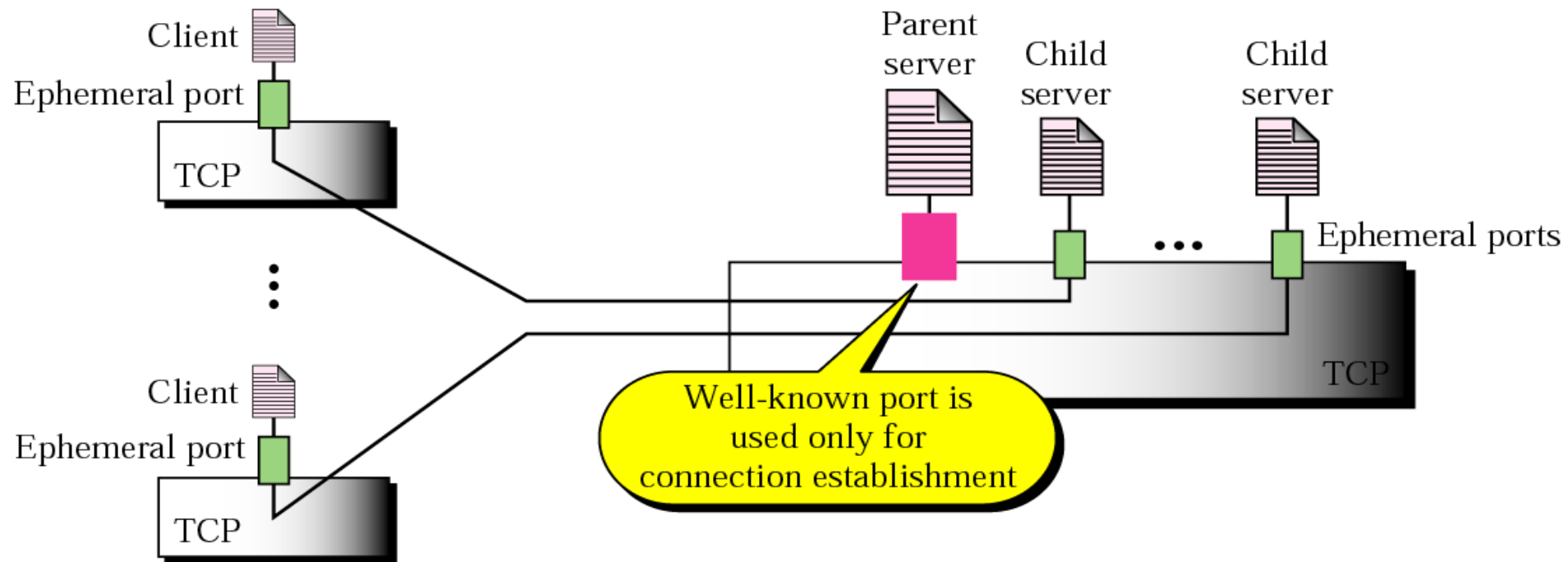
# Client-Server Model

- **Connectionless iterative server**: the ones that use **UDP** are iterative, server uses one single port, arriving packets wait in line,
- **Connection oriented concurrent server**: the ones that use **TCP** are normally concurrent,

# Connectionless iterative server



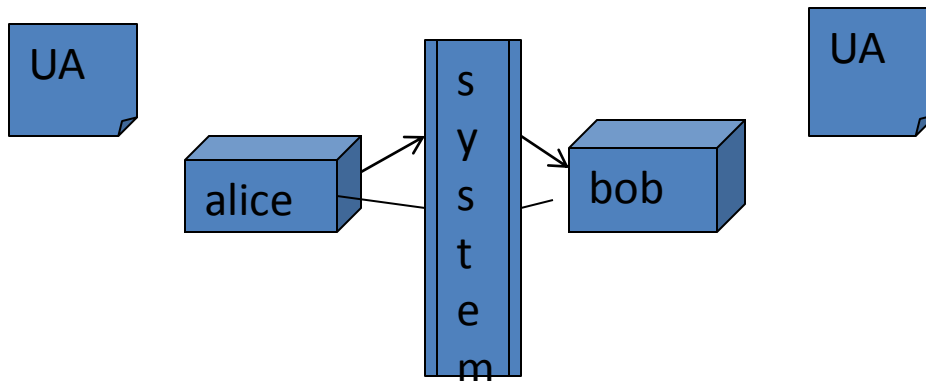
# Connection-oriented concurrent server



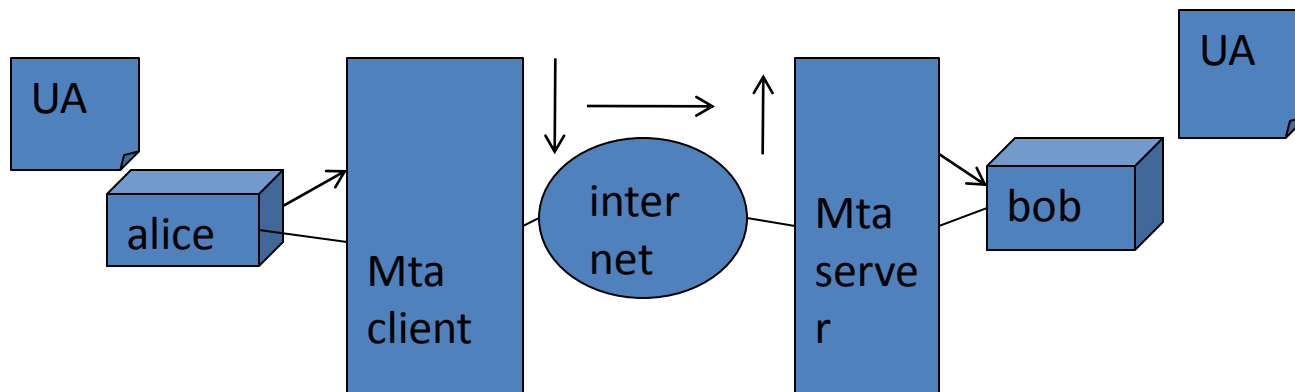
# Client-Server Model

- => well-known port is **free** to receive requests for additional connections
- A **program** is code (in UNIX) and defines all the variables and actions to be performed on those variables.
- A **process** is an instance of a program.
- An OS creates a process when executing a program, several **processes** can be created from one program running **concurrently**.

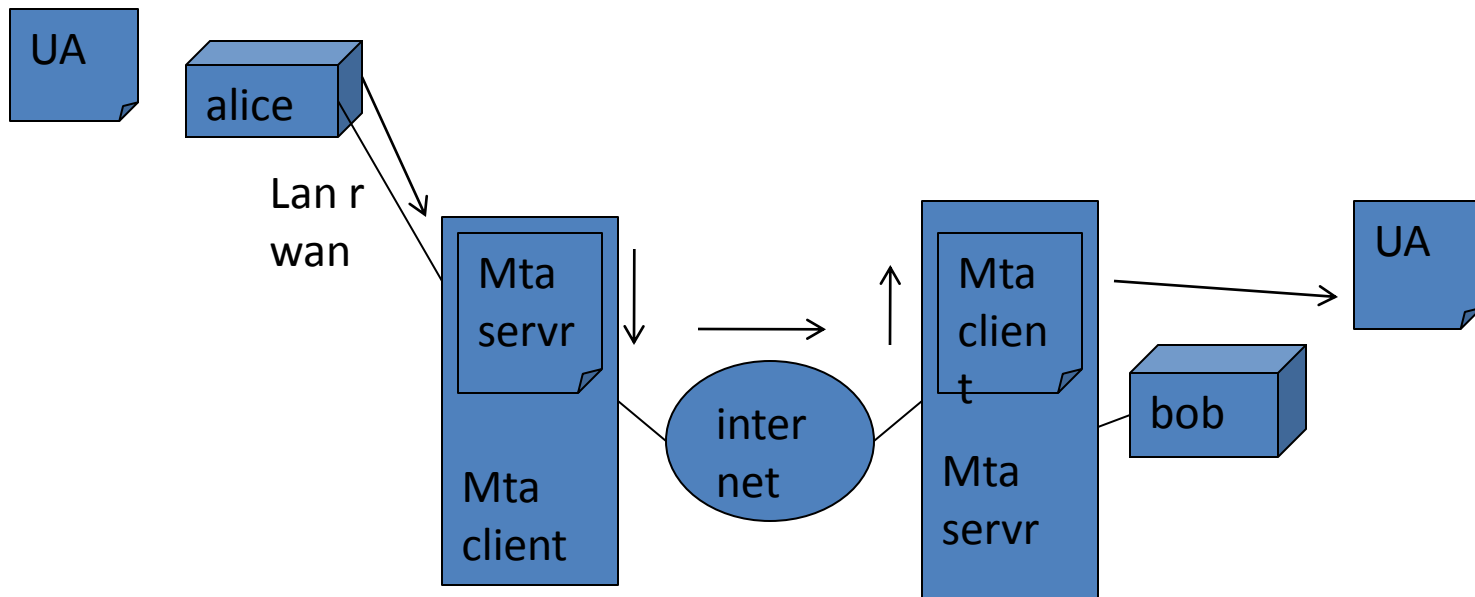
- Email:
- Four scenarios:
- First: sndr & rcvr of email r users on same sys.they r directly connected to shared sys.admin creates 1 mailbox for each user.when alice wants to snd msg to bob alice runs a useragent prgm to prepare msg & store in bob's mailbox



- second: sndr & rcvr of email r users on different sys.msg to b sent to internet.here v need useragents & message agents.they r directly connected to shared.when alice wants to snd msg to bob alice runs a useragent prgm to send her msg to sys at her own site..

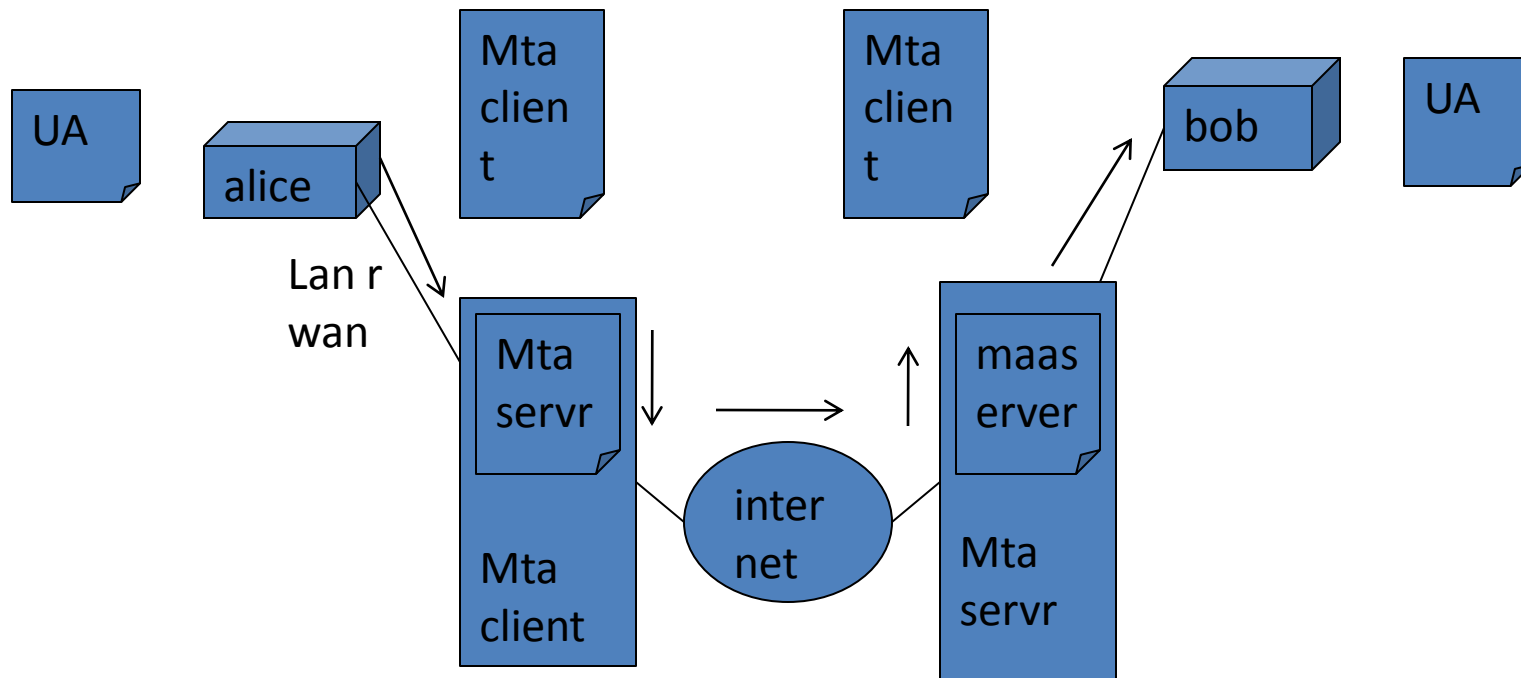


- third: same as second but bob is directly connected to his system and alice is separated from her sys. alice is connected to sys via lan or wan. msg to b sent to internet. here alice still needs user agents to prepare msg and send it on lan. this is done by pair of msg transfer agents. When alice has msg to send, she calls ua which in turn calls mta client. mta client establishes connection with mta server. you need 2 ua's 2 ma's.





- fourth: bob is also connected to his mail server by lan r wan system .after msg has arrived at bob's mail server bob needs to retrieve it.here v need another set of client server agents which v call msg access agents(MAA).bob uses maa client to retrieve his msgs.



# ELECTRONIC MAIL

- There are **two** popular applications for exchanging information.
  - **Electronic mail** exchanges information between people.
  - **File transfer** exchanges files between computers.

# ELECTRONIC MAIL

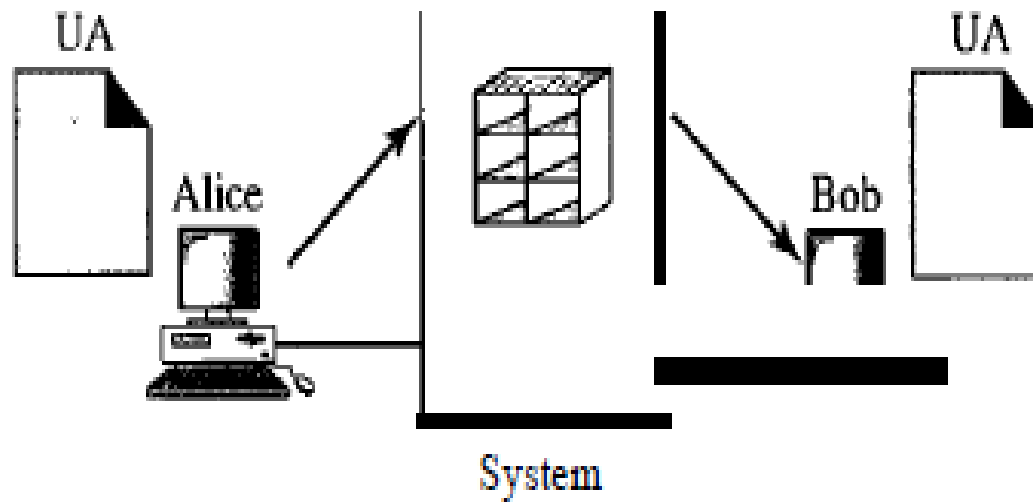
- One of the most popular Internet services is electronic mail (e-mail).
- Electronic mail used for sending a single message that includes text, voice, video or graphic to one or more recipients.

# Architecture

- To explain the architecture of e-mail, we give four scenarios
- *First Scenario*
  - In the first scenario, the sender and the receiver of the e-mail are users (or application programs) on the same system;
  - they are directly connected to a shared system.
  - When Alice, a user, needs to send a message to Bob, another user, Alice runs a *user agent (UA) program to prepare the message* and store it in Bob's mailbox.
  - Bob can retrieve and read the contents of his mailbox at his convenience, using a user agent.

# *First Scenario*

*UA*: user agent



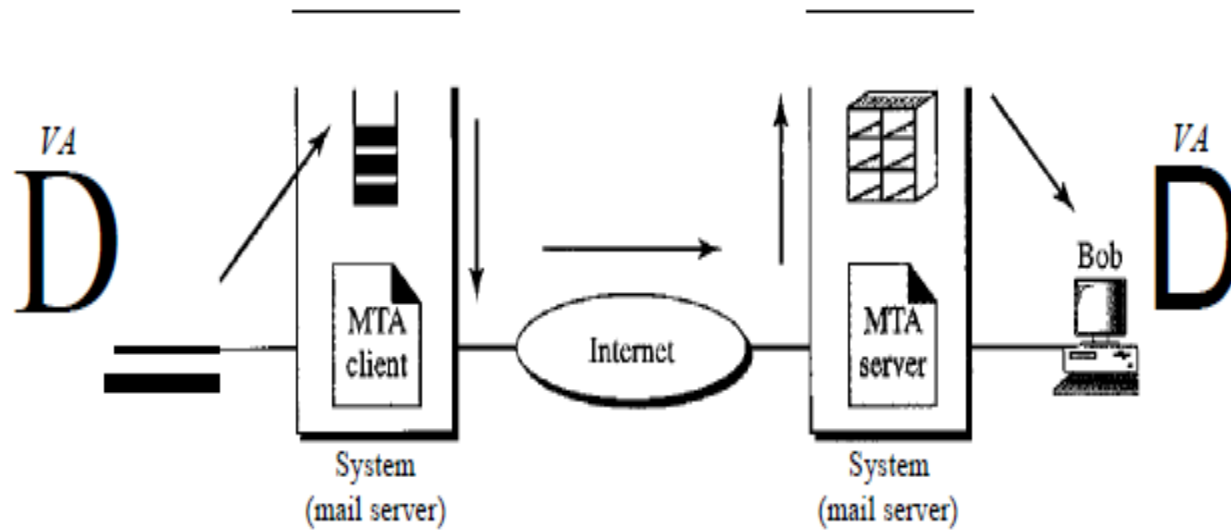
# *Second Scenario*

- In the second scenario, the sender and the receiver of the e-mail are users (or application programs) on two different systems.
- The message needs to be sent over the Internet.
- Here we need user agents (VAs) and message transfer agents (MTAs)
- **When the sender and the receiver of an e-mail are on different systems, we need two VAs and a pair of MTAs (client and server).**

# Second Scenario

*VA*: user agent

MTA: message transfer agent

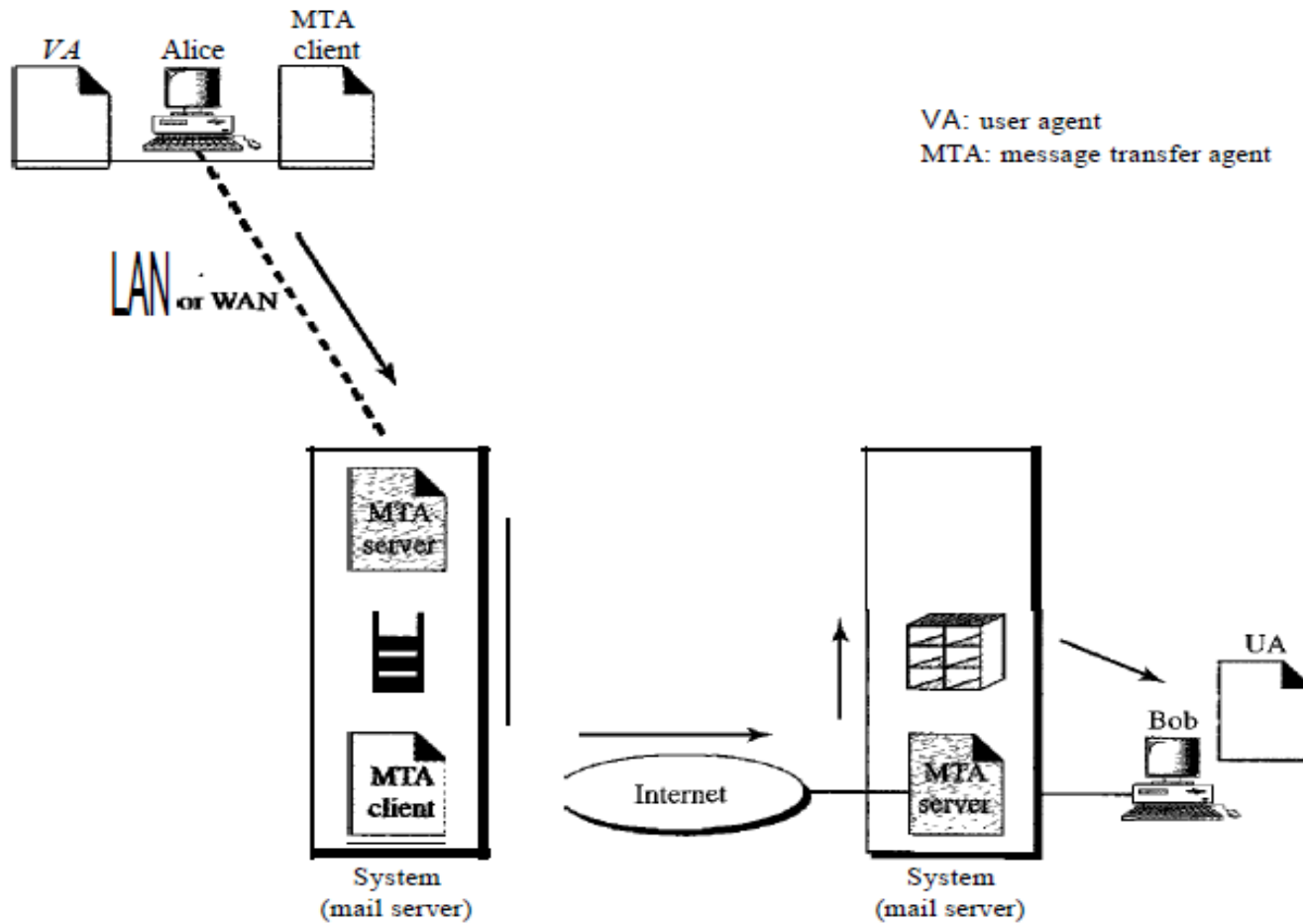


# *Third Scenario*

- In the third scenario, Bob is directly connected to his system.
- Alice, however, is separated from her system.
- Either Alice is connected to the system via
  - a point-to-point WAN,
  - such as a dial-up modem,
  - a DSL, or a cable modem;
- or she is connected to a LAN in an organization that uses one mail server for handling e-mails.



# Third Scenario

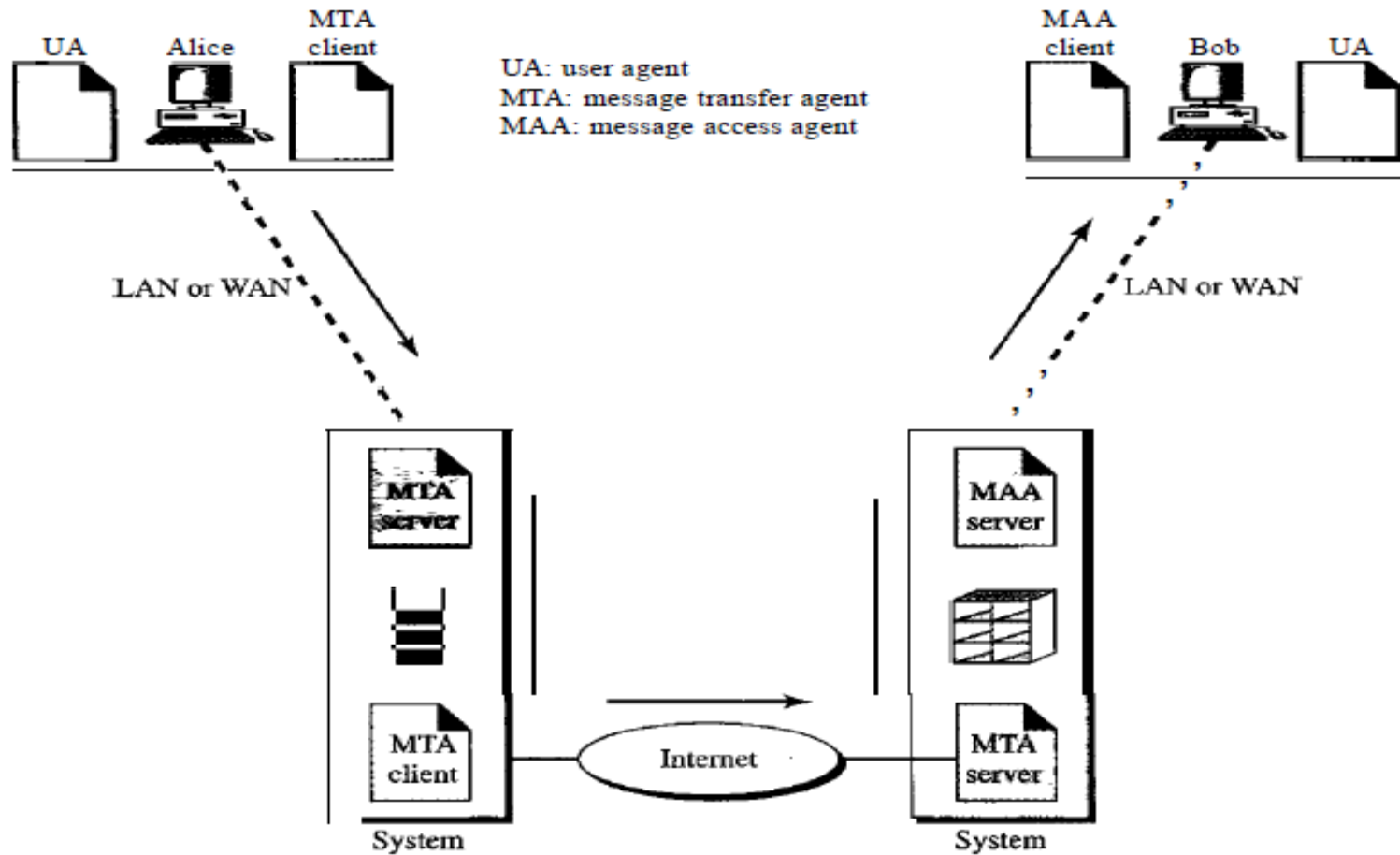


## *Fourth Scenario*

- In the fourth and most common scenario, both Alice and Bob are connected to his mail server by a
- WAN or a LAN. After the message has arrived at Bob's mail server, Bob needs to retrieve it.
- Here, we need another set of client/server agents, which we call message access agents (MAAs).
- Bob uses an MAA client to retrieve his messages.
- The client sends a request to the MAA server, which is running all the time, and requests the transfer of the messages.

- When both sender and receiver are connected to the mail server via a LAN or a WAN, we need two VAs, two pairs of MTAs (client and server), and a pair of MAAs (client and server).
- This is the most common situation today.

# Fourth Scenario



# Sending Mail

- To send mail the user creates mail that looks very similar to postal mail
- It has envelope and a message.
- **En elope:**
  - the envelope usually contains the sender address, the receiver address and other information
- **Message:**
  - Contains header and body.
  - The header of the message define the sender, the receiver, the subject of the message
  - The body of the message contains actual information to be read by the recipient.

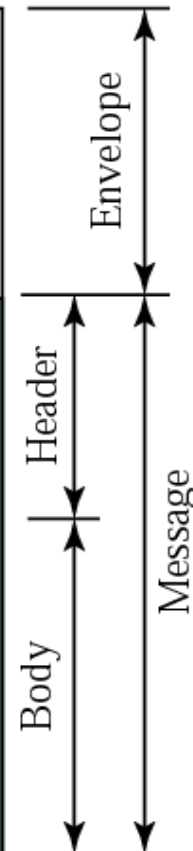
# Format of an email

Behrouz Forouzan  
De Anza College  
Cupertino, CA 96014  
Sophia Fegan  
Com-Net  
Cupertino, CA 95014

Sophia Fegan  
Com-Net  
Cupertino, CA 95014  
Jan. 5, 2003  
Subject: Network  
Dear Mrs. Fegan:  
We want to inform you that our network is working properly after the last repair.  
Yours truly,  
Behrouz Forouzan

Mail From: forouzan@deanza.edu  
RCPT To: fegan@comnet.com

From: Behrouz Forouzan  
To: Sophia Fegan  
Date: 1/5/03  
Subject: Network  
Dear Mrs. Fegan:  
We want to inform you that our network is working properly after the last repair.  
Yours truly,  
Behrouz Forouzan

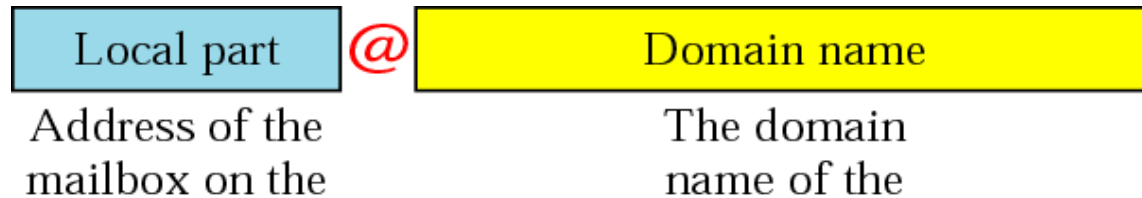


# Receiving Mail

- The receiving system periodically checks mailboxes.
- If a user has mail, it informs the user with a notice.
- If the user is ready to read the mail, a list is displayed in which each line contains a summary of the information about a particular message in the mailbox
- The summary usually includes the sender mail address, the subject and the time the mail was sent or received.
- The user can select any of the messages and display its contents on the screen.

# Email address

- To deliver mail, a mail handling system must use an addressing system with unique address.



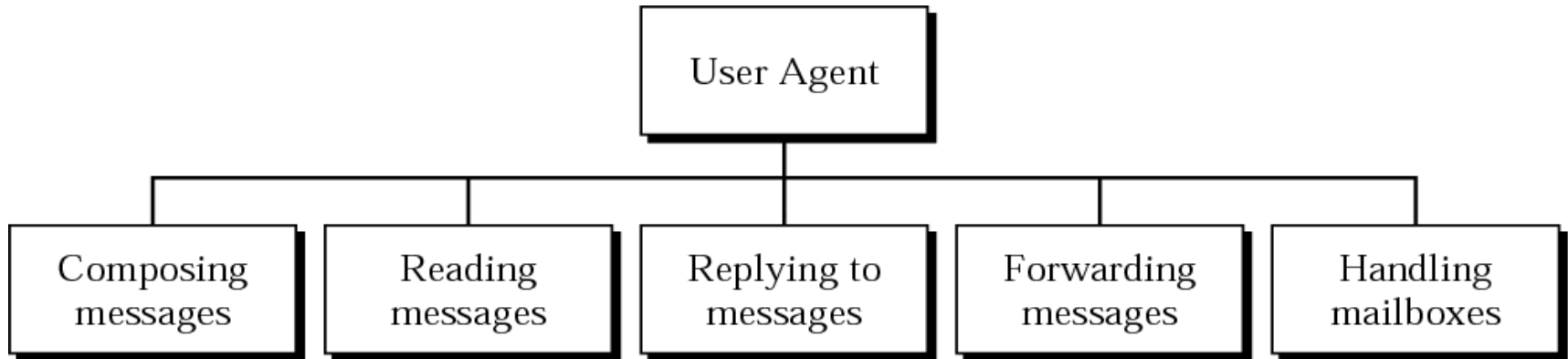
- Local part defines the name of a special file, called the user mailbox, where all the mail received are stored for retrieval by the user agent.



# User Agent

- The first component of an electronic mail system is the user agent (*UA*).
- *It provides* service to the user to make the process of sending and receiving a message easier.
- A user agent is a software package (program) that
  - composes,
  - reads,
  - replies to,
  - and forwards messages
- It also handles mailboxes

# User agent



# MIME

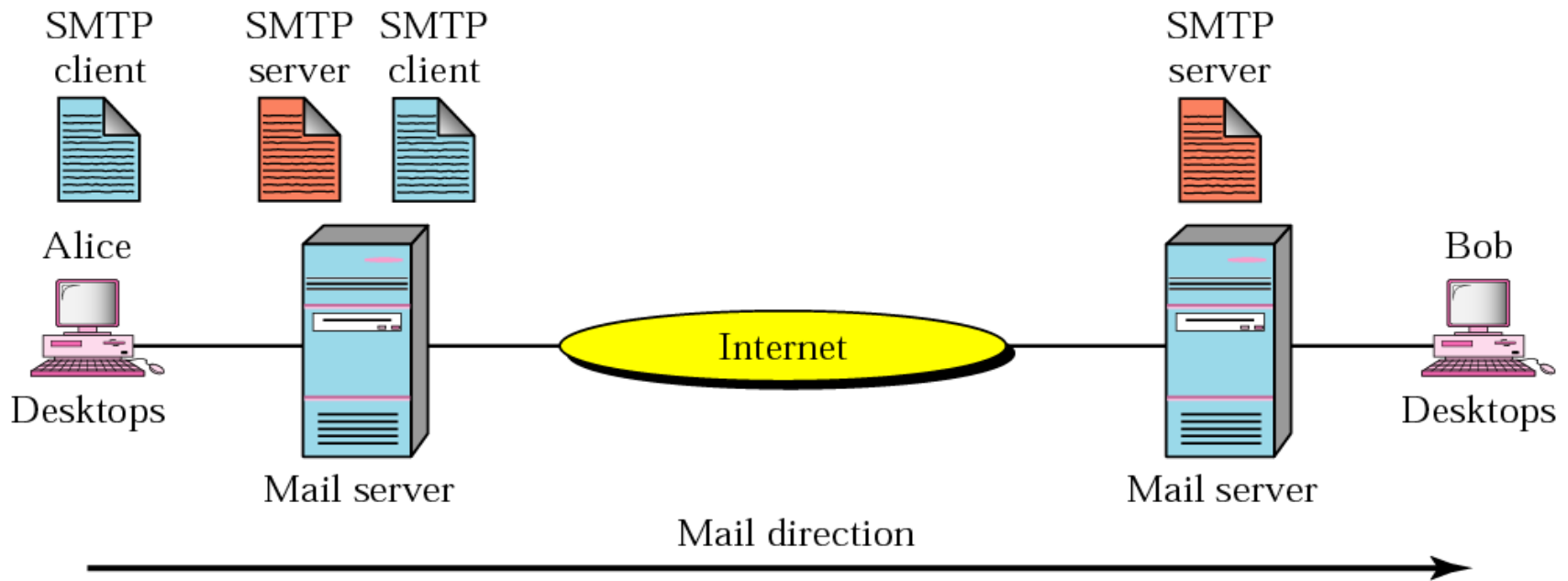
- It can send messages only in 7-bit ASCII format.
- In other words, it has some limitations.
- For example, it cannot be used for languages that are not supported by 7-bit ASCII characters (such as French, German, Hebrew, Russian, Chinese, and Japanese).
- Also, it cannot be used to send binary files or video or audio data.
- **Multipurpose Internet Mail Extensions (MIME)** is a supplementary protocol that allows non-ASCII data to be sent through e-mail.
- MIME transforms non-ASCII data at the sender site to ASCII data and delivers them to the client MTA to be sent through the Internet.
- The message at the receiving side is transformed back to the original data.
- We can think of MIME as a set of software functions that transforms non-ASCII data (stream of bits) to ASCII data and vice versa,

# MIME

- MIME defines five headers that can be added to the original e-mail header section to define the transformation parameters:
  - 1. MIME-Version
  - 2. Content-Type
  - 3. Content-Transfer-Encoding
  - 4. Content-Id
  - 5. Content-Description

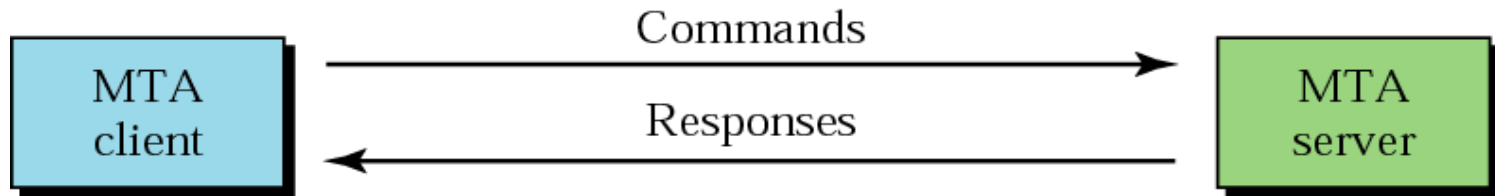
# MTA client and server

- Actual mail transfer is done through message transfer agents.
- To send a mail, a sys must have client MTA and to rcv mail,a sys must have a server MTA.



# Commands and responses

- SMTP simply defines how cmds and responses must be sent back and forth btw MTA client and MTA server.

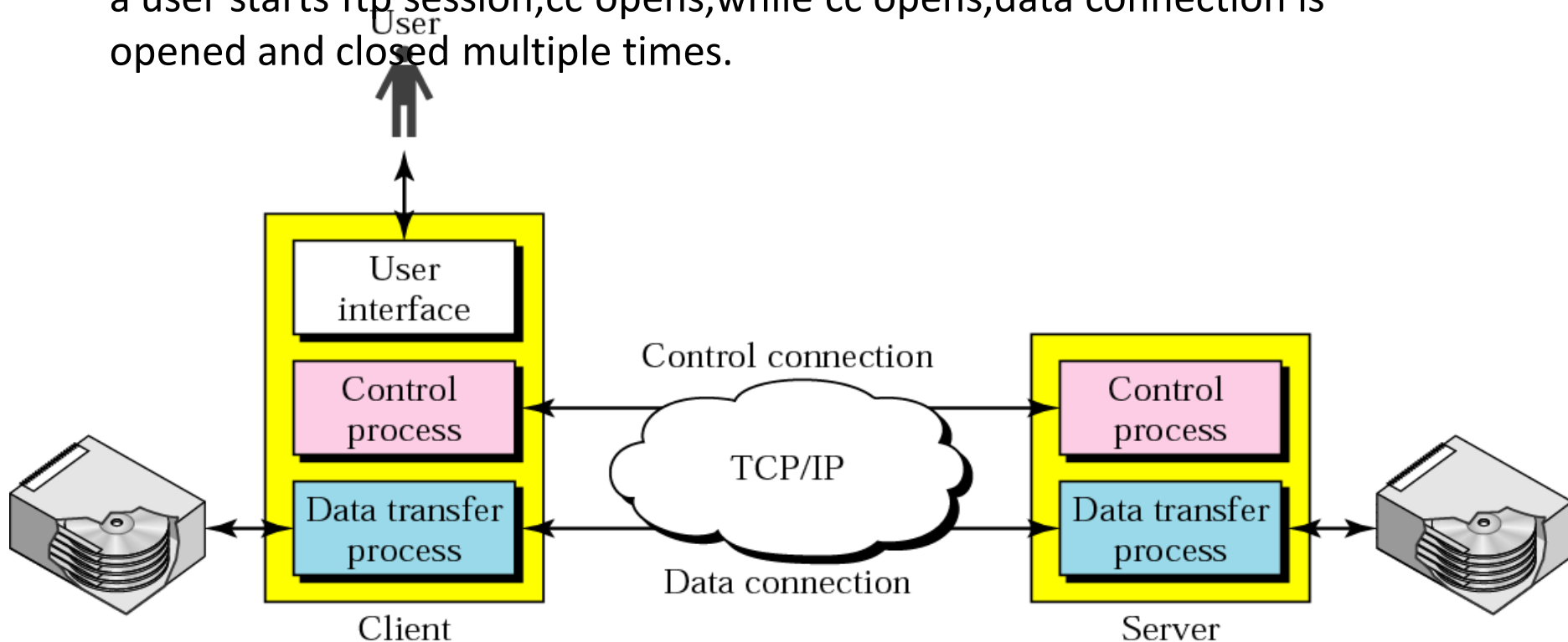


# FTP

- File Transfer Protocol (FTP) is the standard mechanism provided by *TCP/IP* for copying a file from one host to another.
- Although transferring files from one system to another seems simple and straightforward, some problems must be dealt with first.
- For example, 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. All these problems have been solved by FTP

# FTP CONNECTIONS

- **Control connection** remains connected during entire interactive ftp session.
- **data connection** is opened and then closed for each file transferred. when a user starts ftp session, cc opens, while cc opens, data connection is opened and closed multiple times.

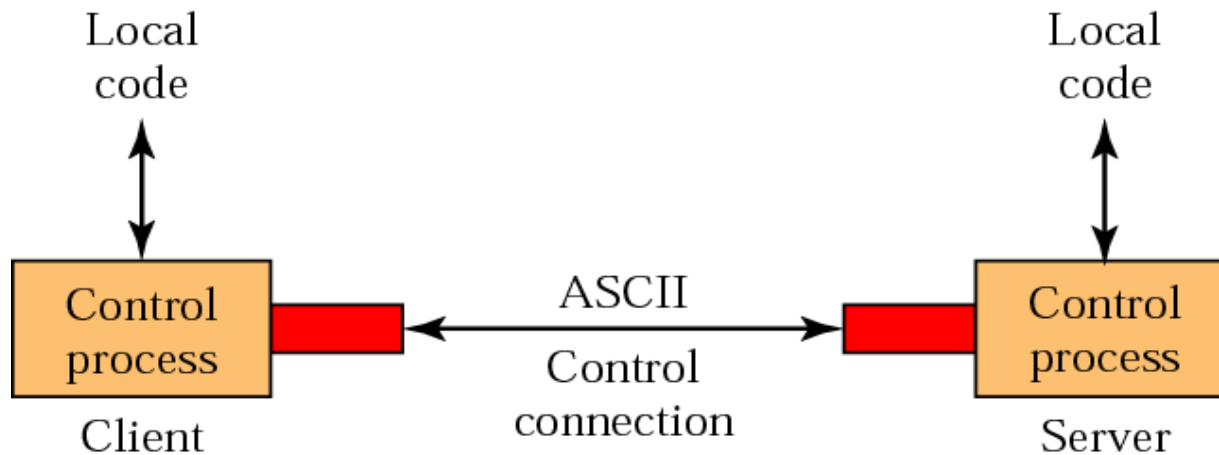




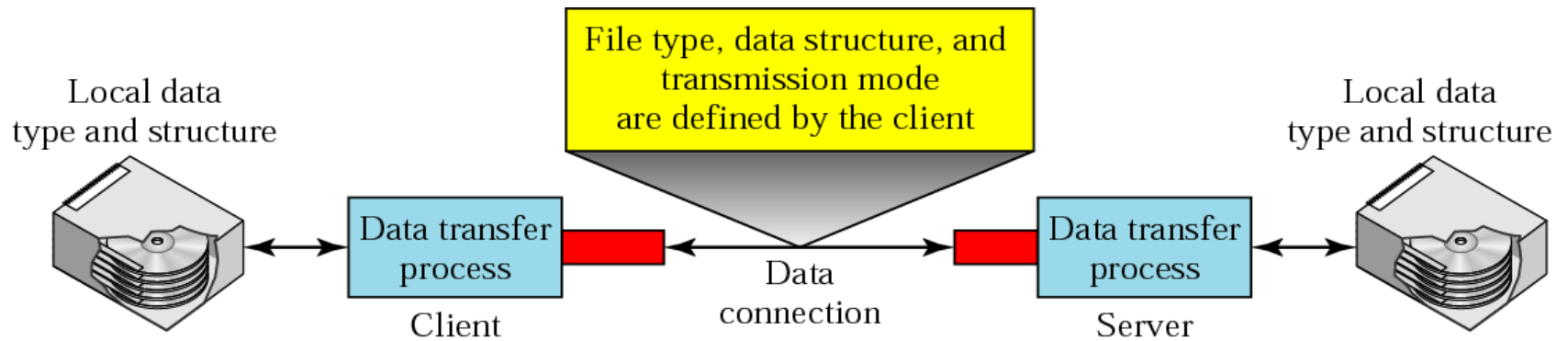
# Communication

- The FTP client and server which run on different computers, must communicate with each other.
- These two computers may use diff operating systems, diff character sets, diff file structures and diff file formats
- FTP must make this compatible.

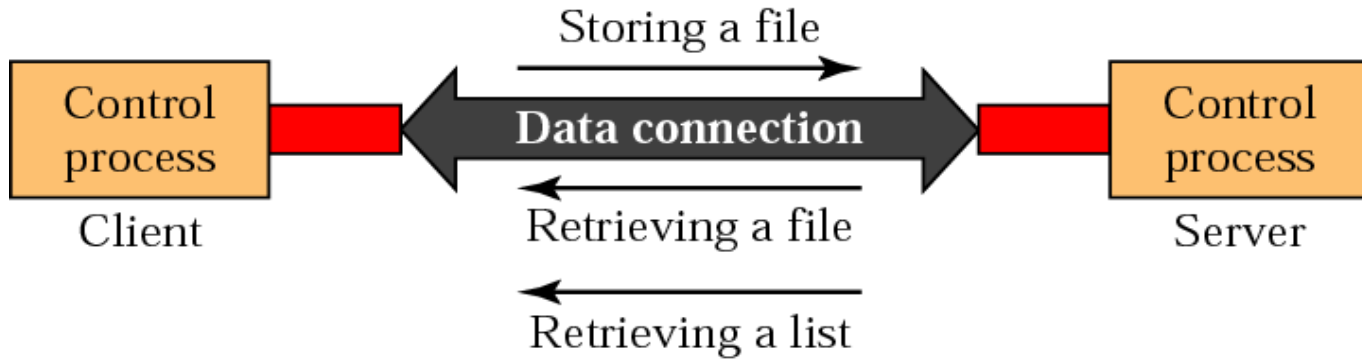
# Communication over control connection



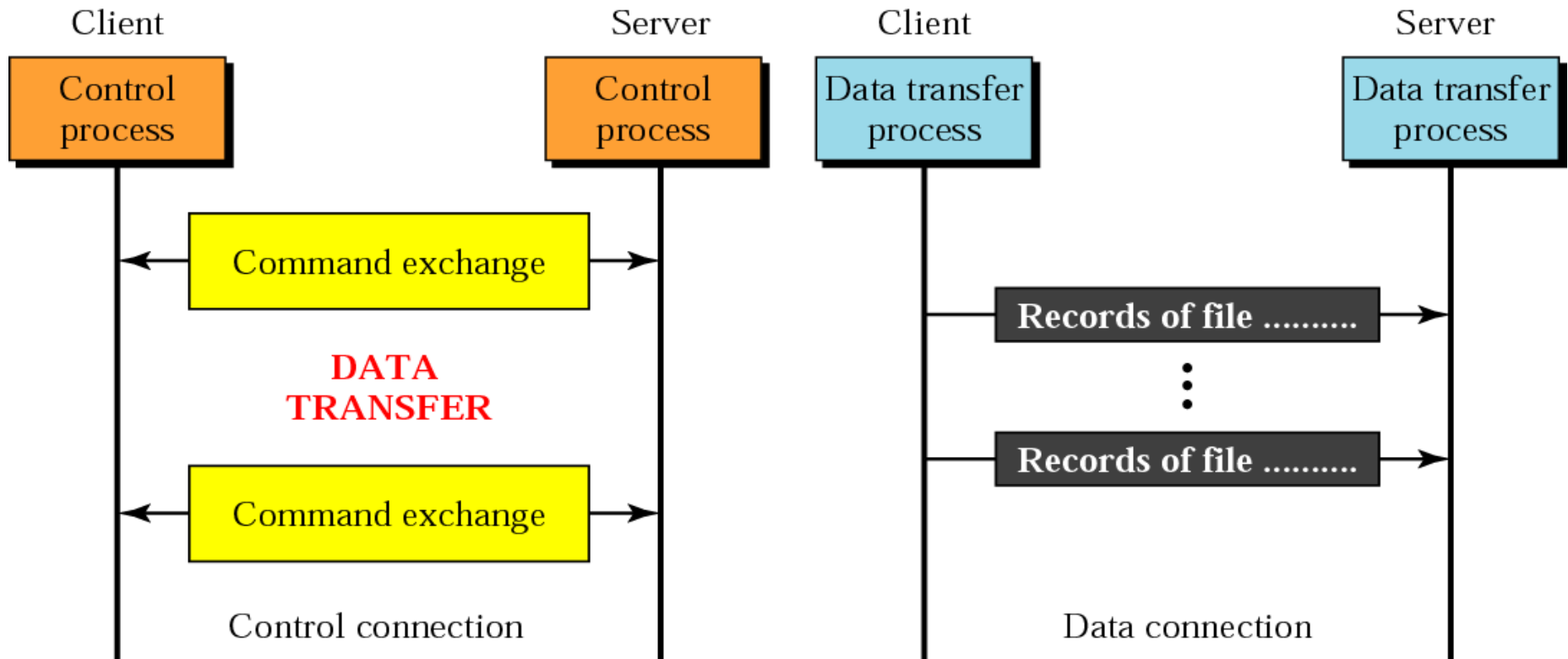
# Using the data connection



# File transfer



# Example 1

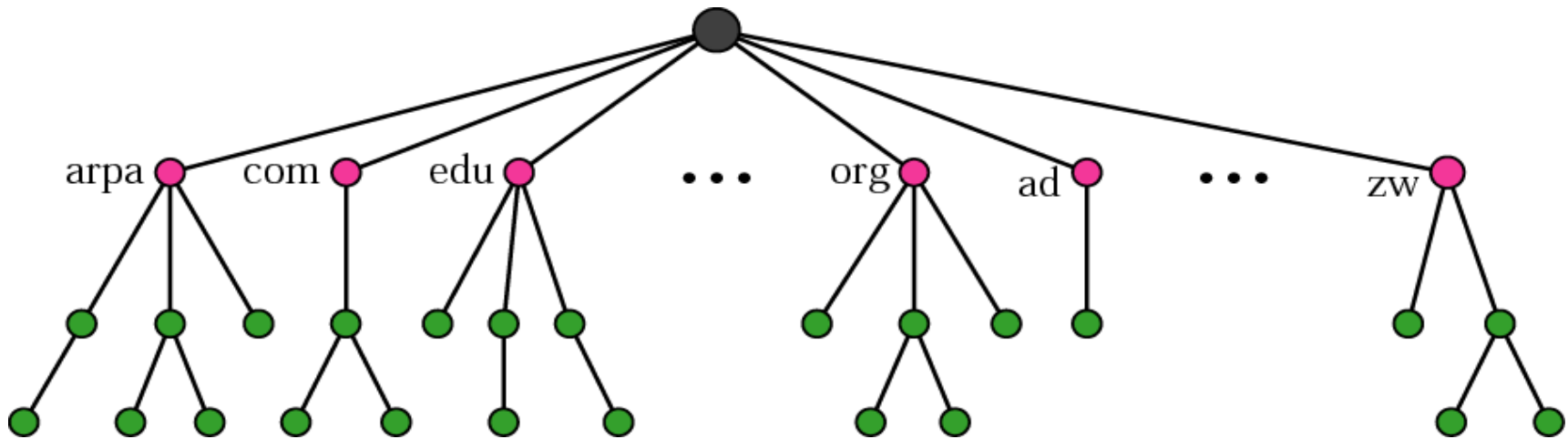


# NAME SPACE

- DNS is supporting program that is used by other programs such as email.
- To be unambiguous, the names assigned to machine must be carefully selected from a name space.
- *Flat Name Space: a name is assigned to an address.*
- *a name in this space is sequence of chars without structure.*
- *main disadv it cannot b used in large sys such as internet.*
- **Hierarchical Name Space:**
- *Each name is made of several names first part can*
- *define nature of orgn ,second part can define depts in orgn .here authority to assign and control name spaces can b decentralized.*

# Domain name space

- To have a hierarchical name space a domain name space was designed.
- Names are defined using inverted tree structure with root at top.
- Tree can have from 0 to 127 levels.
- Each node in tree has label which is string with max of 63 chars.
- Each node in tree has domain name .





# FQDN and PQDN

- If domain name is terminated by null string, it is called fully qualified domain name.
- If domain name is not terminated by null string, it is called partially qualified domain name

FQDN

challenger.atc.fhda.edu.  
cs.hmme.com.  
www.funny.int.

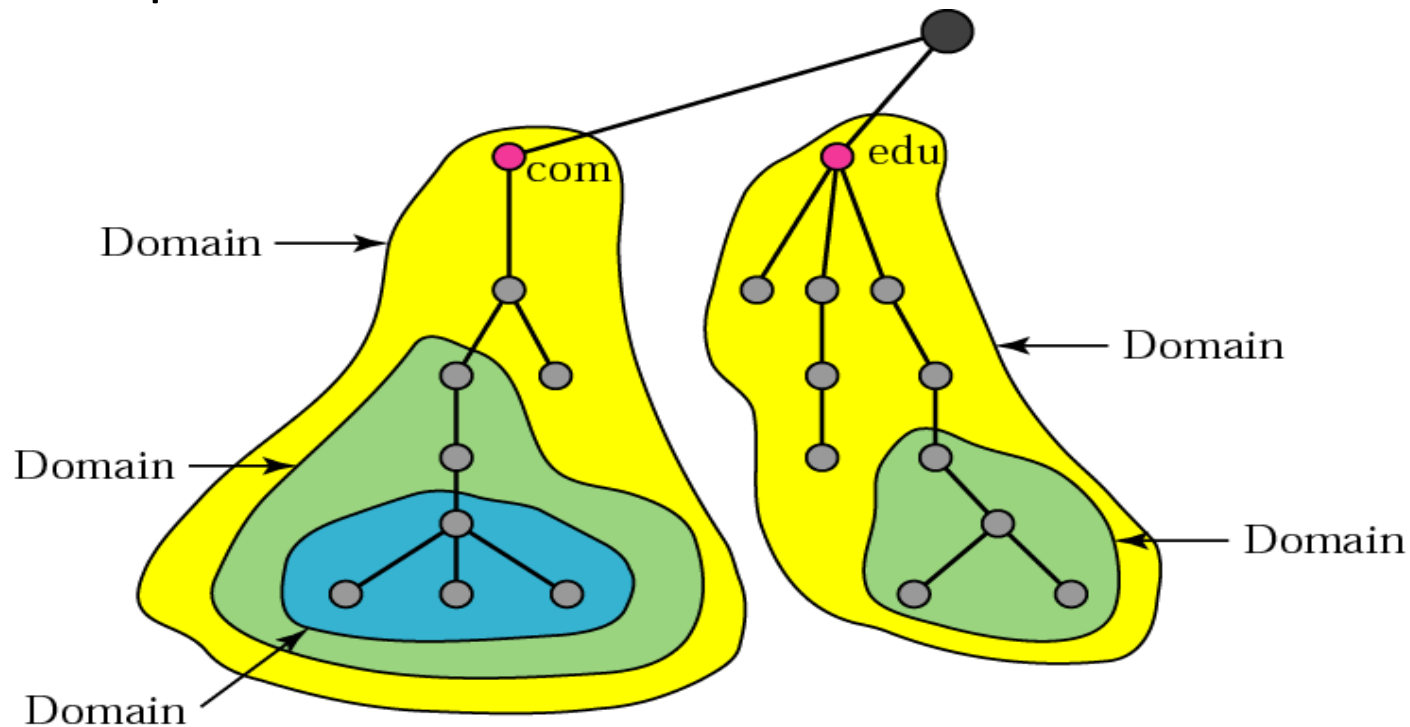
PQDN

challenger.atc.fhda.edu  
cs.hmme  
www



# Domains

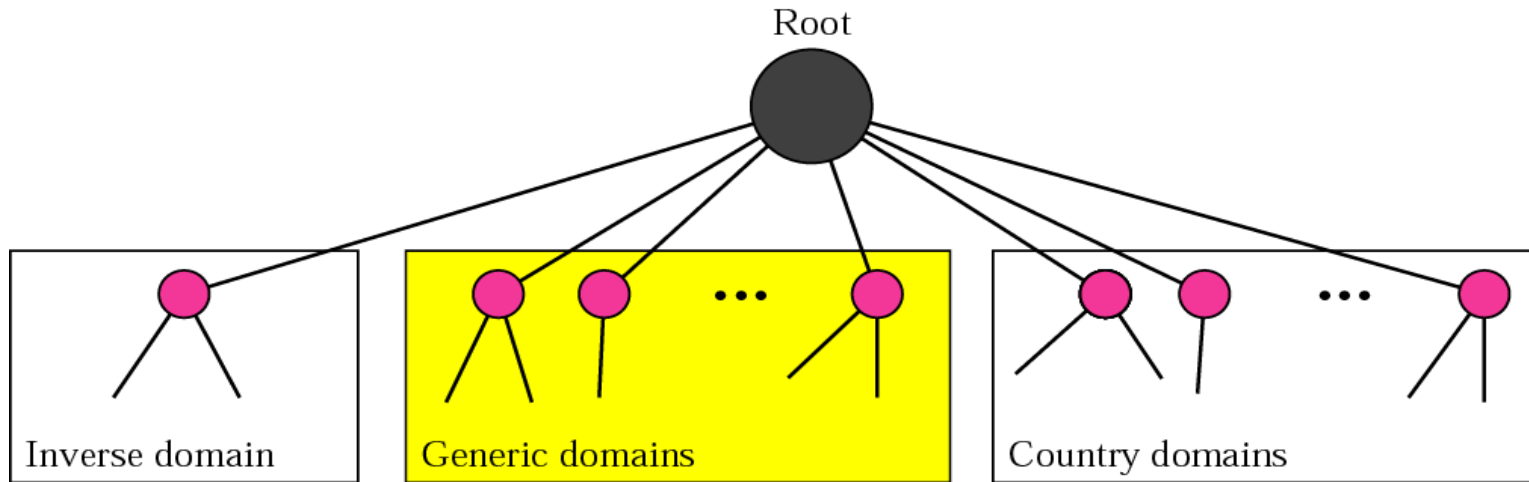
- A domain is a sub tree of domain name space.
- The name of the domain is the domain name of the node at the top of the sub tree.



# DNS IN THE INTERNET

- DNS is a protocol that can be used in different platforms.
- In the Internet, the domain name space (tree) is divided into three different sections:
  - generic domains,
  - country domains,
  - the inverse domain

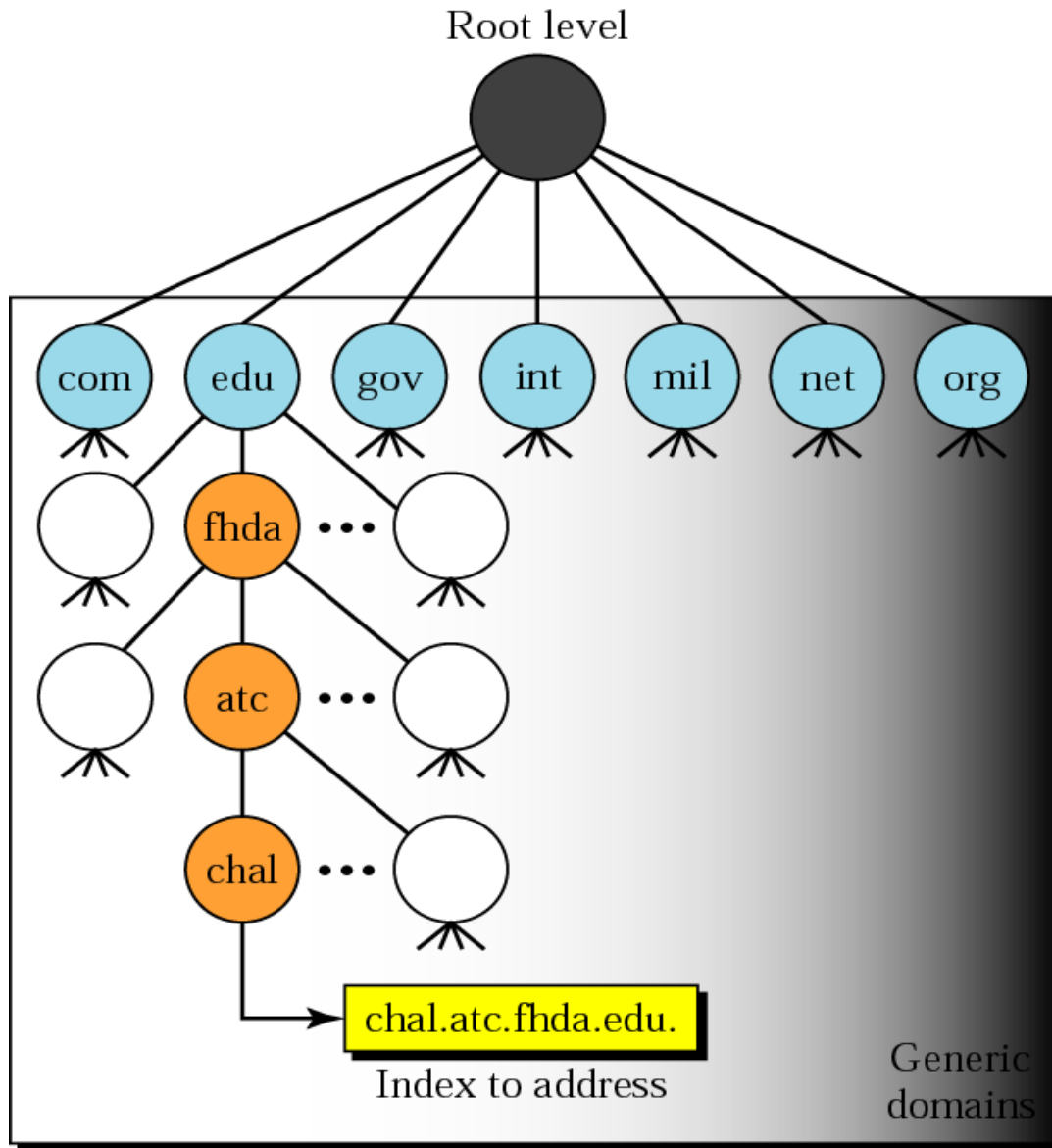
# DNS in the Internet



# generic domains

- The **generic domains define registered hosts according to their generic behavior.**
- Each node in the tree defines a domain, which is an index to the domain name space database

# Generic domains



# country domains

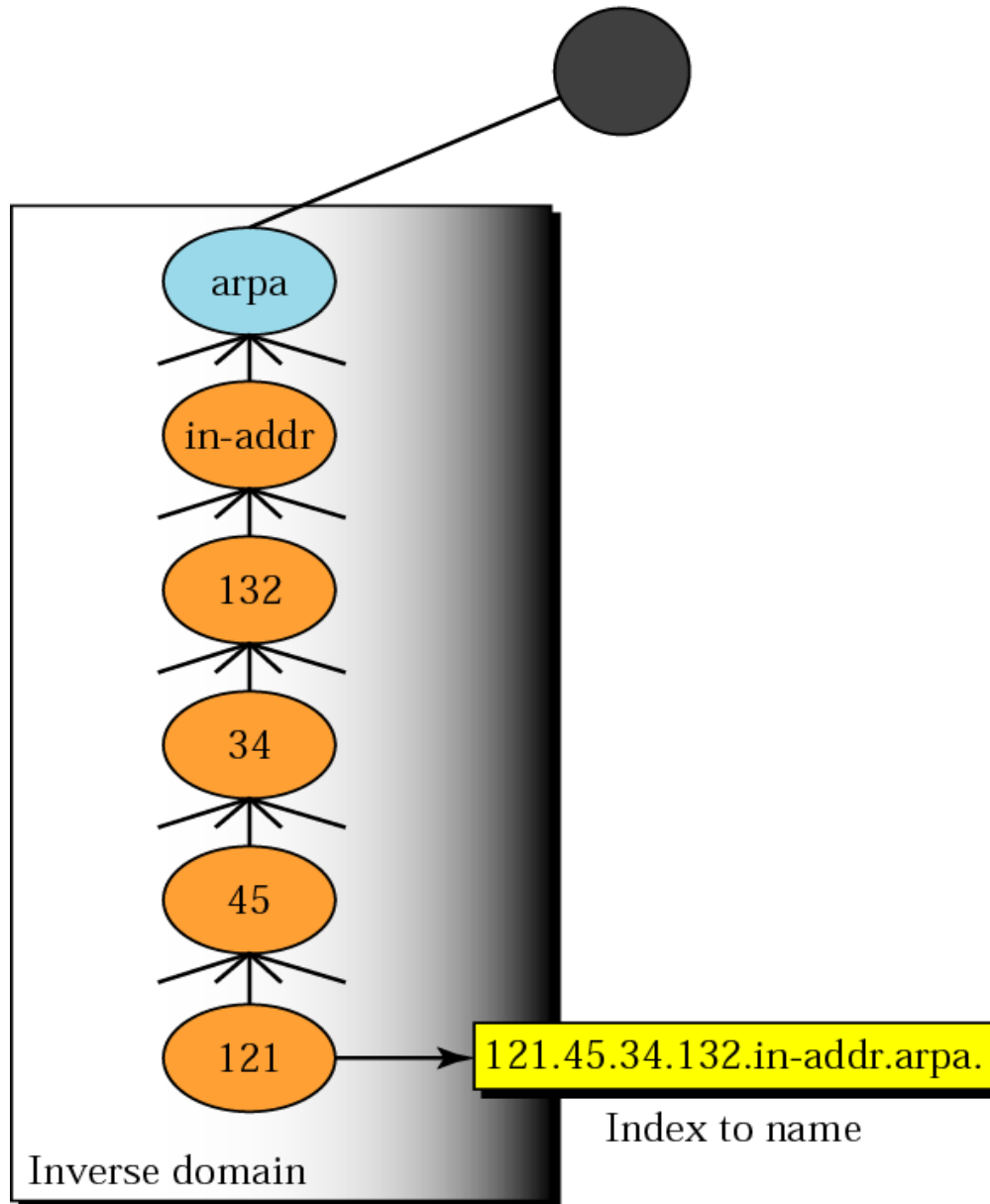
- The country domains section uses two-character country abbreviations (e.g., us for United States).
- Second labels can be organizational, or they can be more specific, national designations.
- The United States, for example, uses state abbreviations as a subdivision of us (e.g., ca.us.).
- The address *anza.cup.ca.us* can be translated to De Anza College in Cupertino, California, in the United States.

# inverse domain

- The inverse domain is used to map an address to a name.
- This may happen, for example, when a server has received a request from a client to do a task.

# Inverse domain

Root level





# RESOLUTION

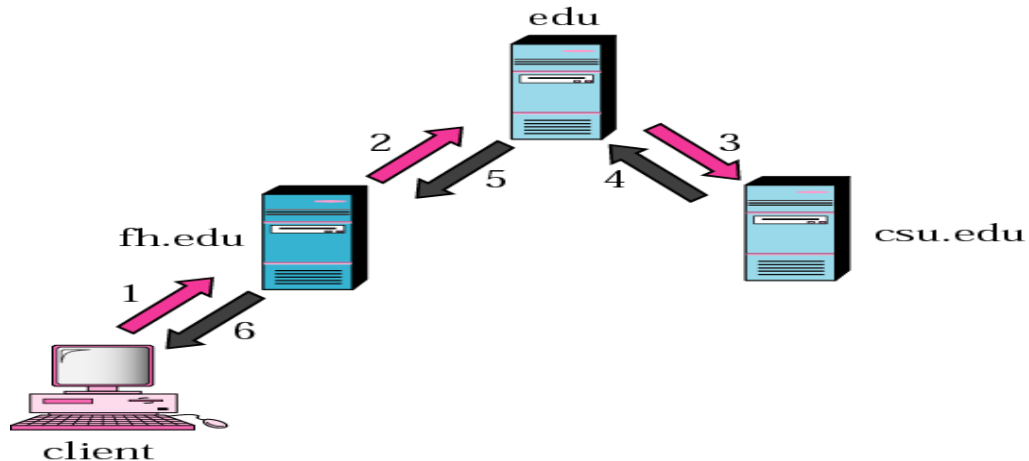
- Mapping a name to an address or an address to a name is called *name-address resolution*.

## Resolver

- DNS is designed as a client/server application. A host that needs to map an address to a name or a name to an address calls a DNS client called a resolver.
- The resolver accesses the closest DNS server with a mapping request.
- If the server has the information, it satisfies the resolver; otherwise, it either refers the resolver to other servers to provide the information.
- After the resolver receives the mapping, and delivers the result to the process that requested it.

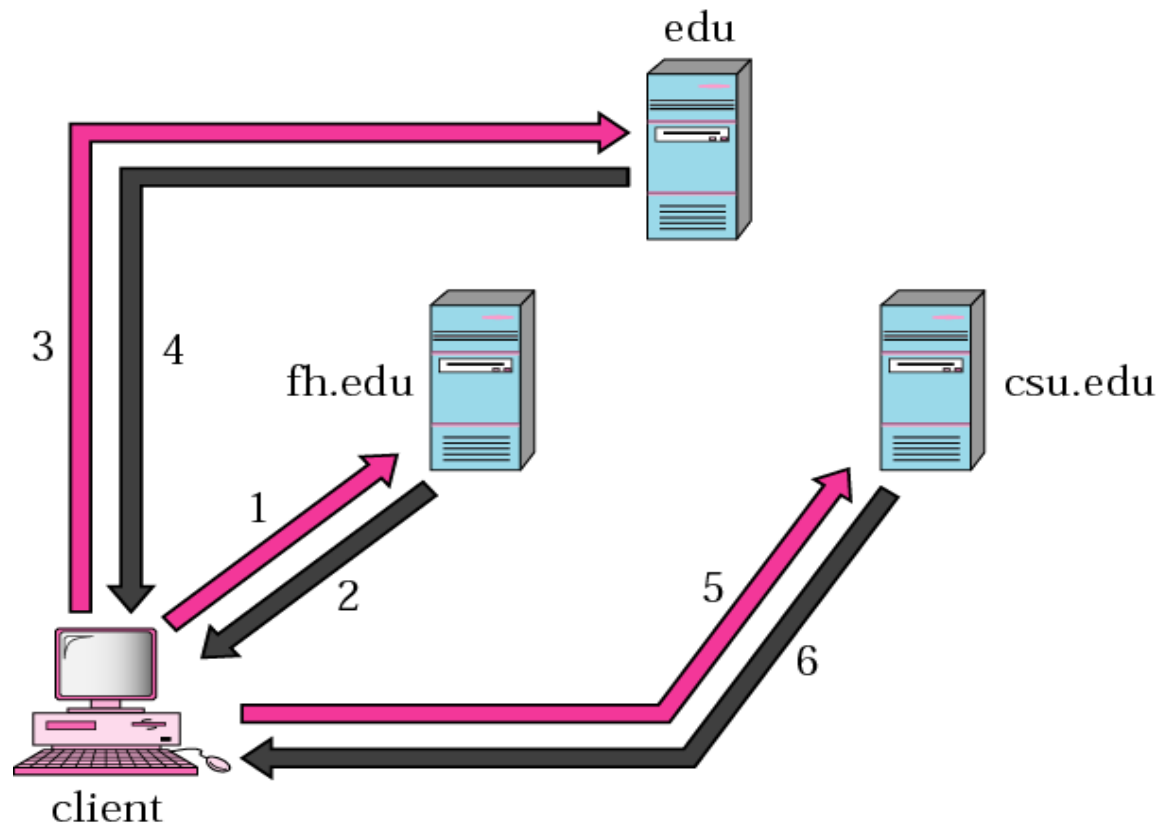
# Recursive resolution

- Dns is designed as a client server appn.a host that needs to map an address to a name or a name to an address calls a dns client called a resolver.resolver accesses the closest dns server with a mapping req.if the server has infn ,it satisfies resolver otherwise it asks resolver to refer other server r to give infn frm other servers.
- Mapping names to addresses
- Mapping addresses to names.
- Recursive resolution:resolver expects server to supply final answer.



# Iterative resolution

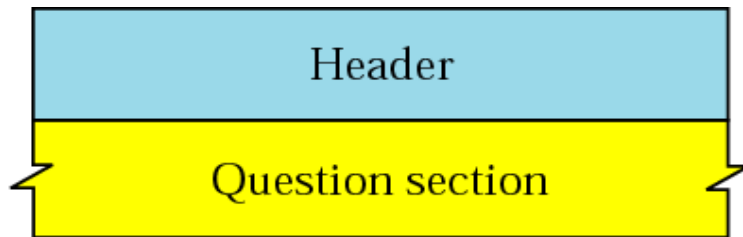
- Client repeats same query to multiple servers.



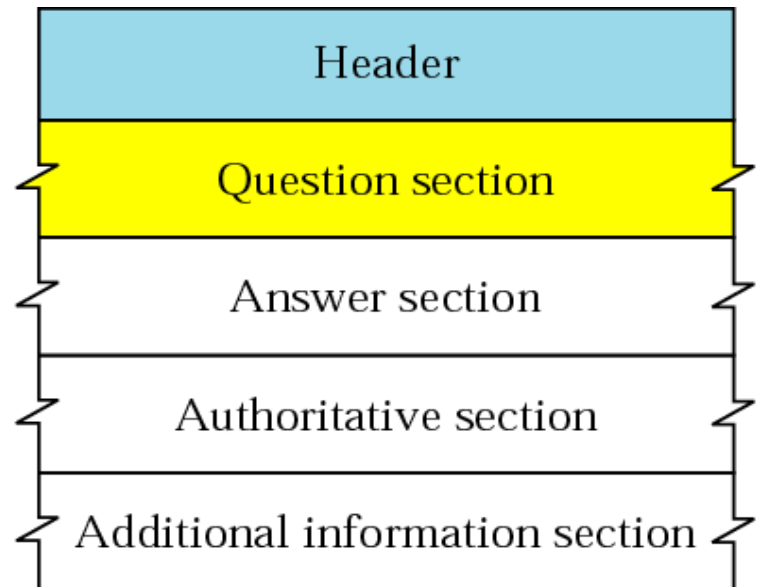
# Query and response messages

## Dns messages

- Identification subfield is used by client to match response with query.



a. Query



b. Response

# Header format

- Id field is used by client to match response with query. flags field define type of msg, type of answer requested, type of desired resolution. no. of question records contains no. of queries in question section of msg.

