SRI CHANDRASEKHARENDRA SARASWATHI VISWA MAHAVIDHYALAYA

(Deemed to be university u/s 3 of UGC act 1956)
(Accredited with "A" by NAAC)
Enathur, Kanchipuram – 631561. Tamilnadu
www.kanchiuniv.ac.in

CS602 – COMPUTER NETWORKS

Name of the Faculty : Dr. N Kumaran

Assistant Professor, Dept. of CSE

E-Mail : nkumaran@kanchiuniv.ac.in

- COURSE : Computer Networks
- PROGRAM : Computer Science and Engineering
- DEGREE : B.E, VI Sem

*

- **PRE-REQUISITE**
- 1. Basics of Computer.
- 2. Digital Circuits.

- **OBJECTIVES**
- To develop an understanding of modern network architectures from a design and performance perspective.
- To introduce the student to the major concepts involved in wide-area networks (WANs), local area networks (LANs) and Wireless LANs (WLANs).
- To provide an opportunity to do network programming
- To provide a WLAN measurement idea.

COURSE OUTCOMES

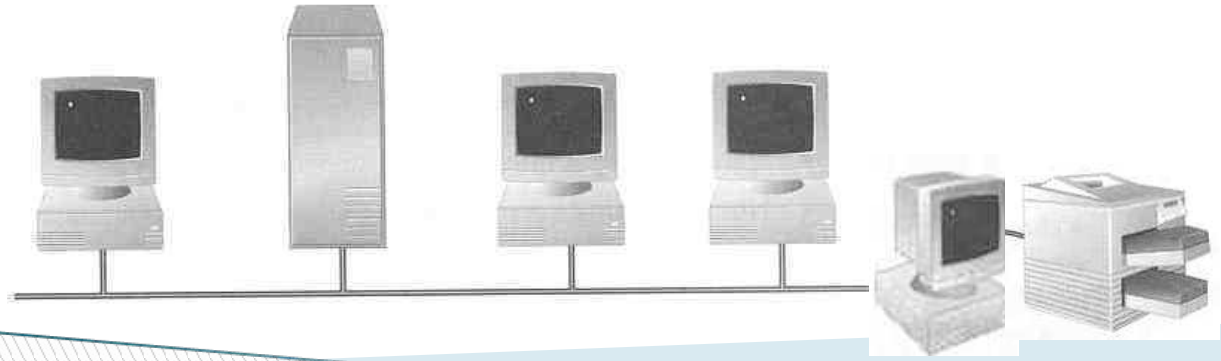
- Draw the functional block diagram of wide-area networks (WANs), local area networks (LANs) and Wireless LANs (WLANs) describe the function of each block.
- Explain the functions of the different layer of the OSI Protocol.
- For a given requirement (small scale) of wide-area networks (WANs), local area networks (LANs) and Wireless LANs (WLANs) design it based on the market available component
- For a given problem related TCP/IP protocol developed the network programming.
- Configure DNS DDNS, TELNET, EMAIL, File Transfer Protocol (FTP), WWW, HTTP, SNMP, Bluetooth,

Outline

- Introduction to Network
- Data communication Components: Representation of data and its flow Networks
- Various Connection Topology,
- Protocols and Standards
- Transmission Media
- Types of Networks
- OSI Reference Model
- Software Defined Network

What is a Network?

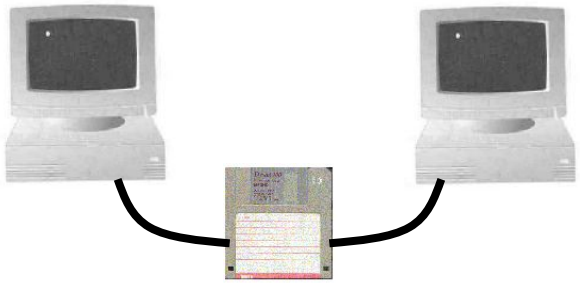
A network consists of 2 or more computers connected together, and they can communicate and share resources (e.g. information)



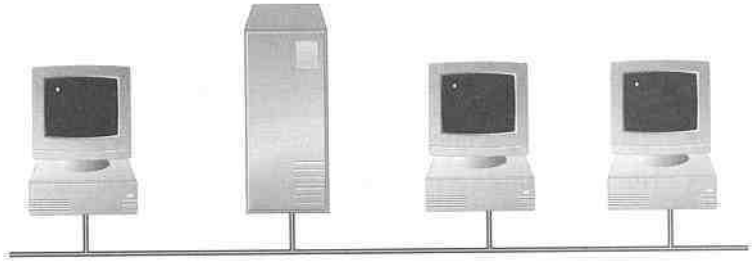
Why Networking?

- **Sharing information — i.e. data communication**

- **Do you prefer these?**



- **Or this?**

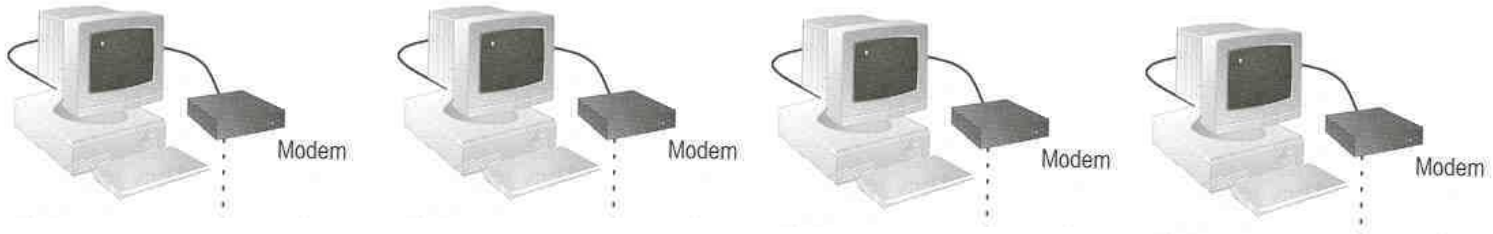


- **Sharing hardware or software**

- **E.g. print document**

Centralize administration and support

- **E.g. Internet-based, so everyone can access the same administrative or support application from their PCs**



USES OF COMPUTER NETWORKS

- **Business Applications**
- to distribute information throughout the company (**resource sharing).** sharing physical resources such as printers, and tape backup systems, is sharing information
- **client-server model**. It is widely used and forms the basis of much network usage.
- **communication medium** among employees. **email** (**electronic mail**), which employees generally use for a great deal of daily communication.
- Telephone calls between employees may be carried by the computer network instead of by the phone company. This technology is called **IP telephony** or **Voice over IP** (**VoIP**) when Internet technology is used.
- **Desktop sharing** lets remote workers see and interact with a graphical computer screen doing business electronically, especially with customers and suppliers.
- This new model is called **e-commerce** (**electronic commerce**) and it has grown rapidly in recent years.

Contd.,

- **Home Applications**
- **peer-to-peer** communication
- person-to-person communication
- electronic commerce
- entertainment. (game playing,)
- **Mobile Users**
- Text messaging or texting
- Smart phones,
- GPS (Global Positioning System)
- m-commerce
- NFC (Near Field Communication)

A data communications system has five components

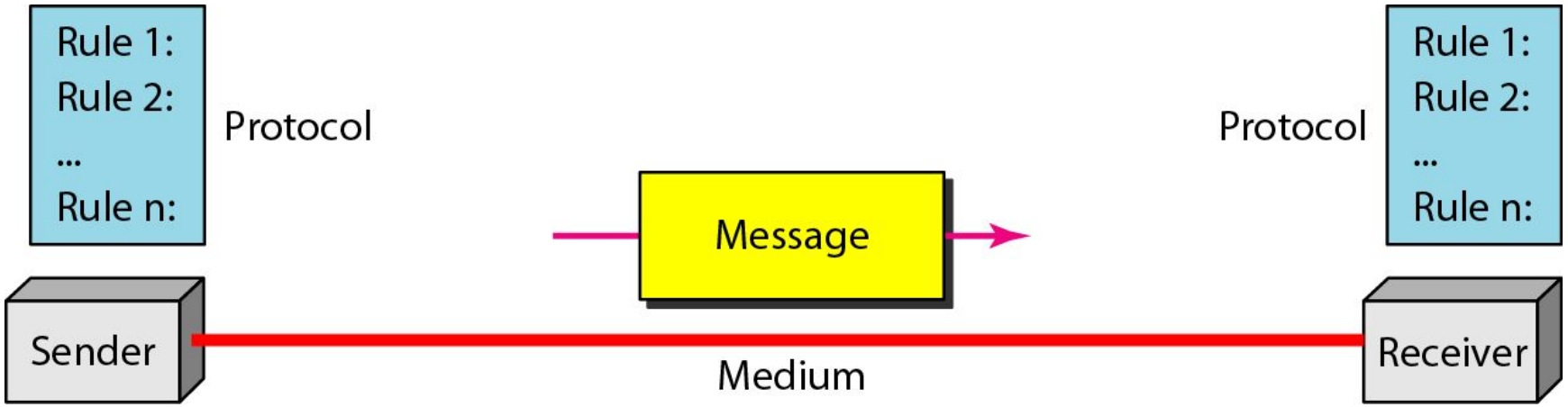
- I. **Message**. The message is the information (data) to be communicated. Popular forms of information include text, numbers, pictures, audio, and video.

- **2. Sender**. The sender is the device that sends the data message. It can be a computer, workstation, telephone handset, video camera, and so on.

- **3. Receiver.** The receiver is the device that receives the message. It can be a computer, workstation, telephone handset, television, and so on.

- **4. Transmission medium**. The transmission medium is the physical path by which a message travels from sender to receiver. Some examples of transmission media include twisted-pair wire, coaxial cable, fiber-optic cable, and radio waves.

5. Protocol. A protocol is a set of rules that govern data communications. It represents an agreement between the communicating devices. Without a protocol, two devices may be connected but not communicating, just as a person speaking French cannot be understood by a person who speaks only Japanese.



Data Flow

- Communication between two devices can be
  - Simplex
  - Full-Duplex
  - Half-Duplex
- Simplex: a simplex mechanism can only transfer data in a single direction
  - **Real Time Example:-** It is analogous to broadcast of radio or télévision
  - Keyboards and traditional monitors are examples of simplex devices.
- Full-Duplex: allows transmission in two directions simultaneously
  - **Real Time Example:-** It is analogous to a voice telephone conversation
    - In which a participant can speak even if they are able to hear background music at the other end

Simplex, Half-Duplex, and Full-Duplex Transmission

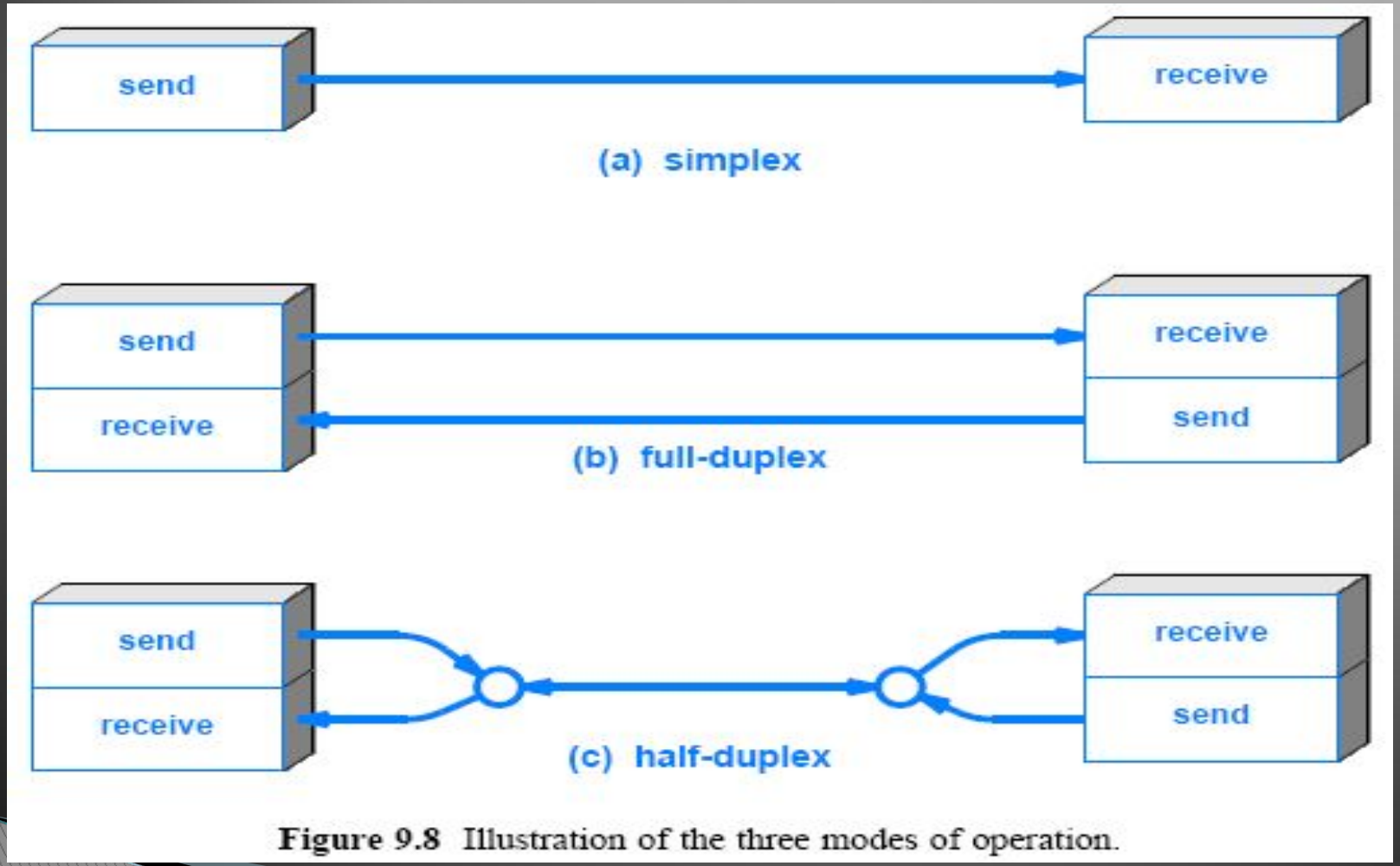


Figure 9.8 Illustration of the three modes of operation.

- Half-Duplex: A half-duplex mechanism involves a shared transmission medium
  - The shared medium can be used for communication in each direction
  - But the communication cannot proceed simultaneously
  - **Real Time Example:-** It is analogous to using walkie-talkies where only one side can transmit at a time
- An additional mechanism is needed at each end of a half-duplex communication that coordinates transmission
  - to insure that only one side transmits at a given time

Kinds of Networks?

- **Depending on one's perspective, we can classify networks in different ways**
  - **Based on transmission media: Wired (UTP, coaxial cables, fiber-optic cables) and Wireless**
  - **Based on network size: LAN and WAN (and MAN)**
  - **Based on management method: Peer-to-peer and Client/Server**
  - **Based on topology (connectivity): Bus, Star, Ring …**
    - :
    - :

LAN and WAN

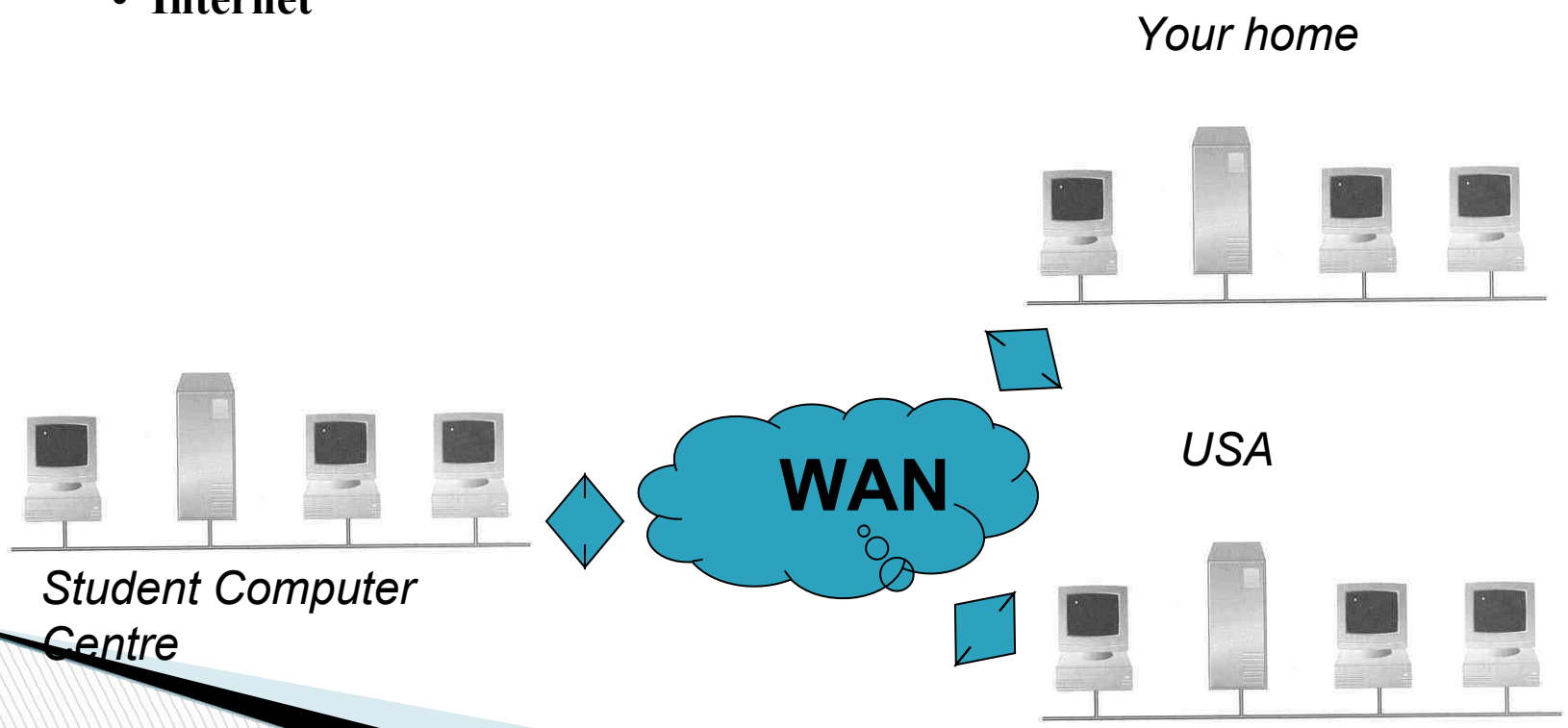
- **Local Area Network (LAN)**

  - small network, short distance
  - A room, a floor, a building
  - Limited by no. of computers and distance covered
  - Usually one kind of technology throughout the LAN
  - Serve a department within an organization

  - **Examples:**
    - Network inside the Student Computer Room
    - Network inside our department
    - Network inside your home

• Wide Area Network (WAN)

- **A network that uses long-range telecommunication links to connect 2 or more MANs/computers housed in different places far apart.**
  - **Towns, states, countries**
- **Examples:**
  - **Network of our Campus**
  - **Internet**



- **Example of WAN: Broadband Cable Network**

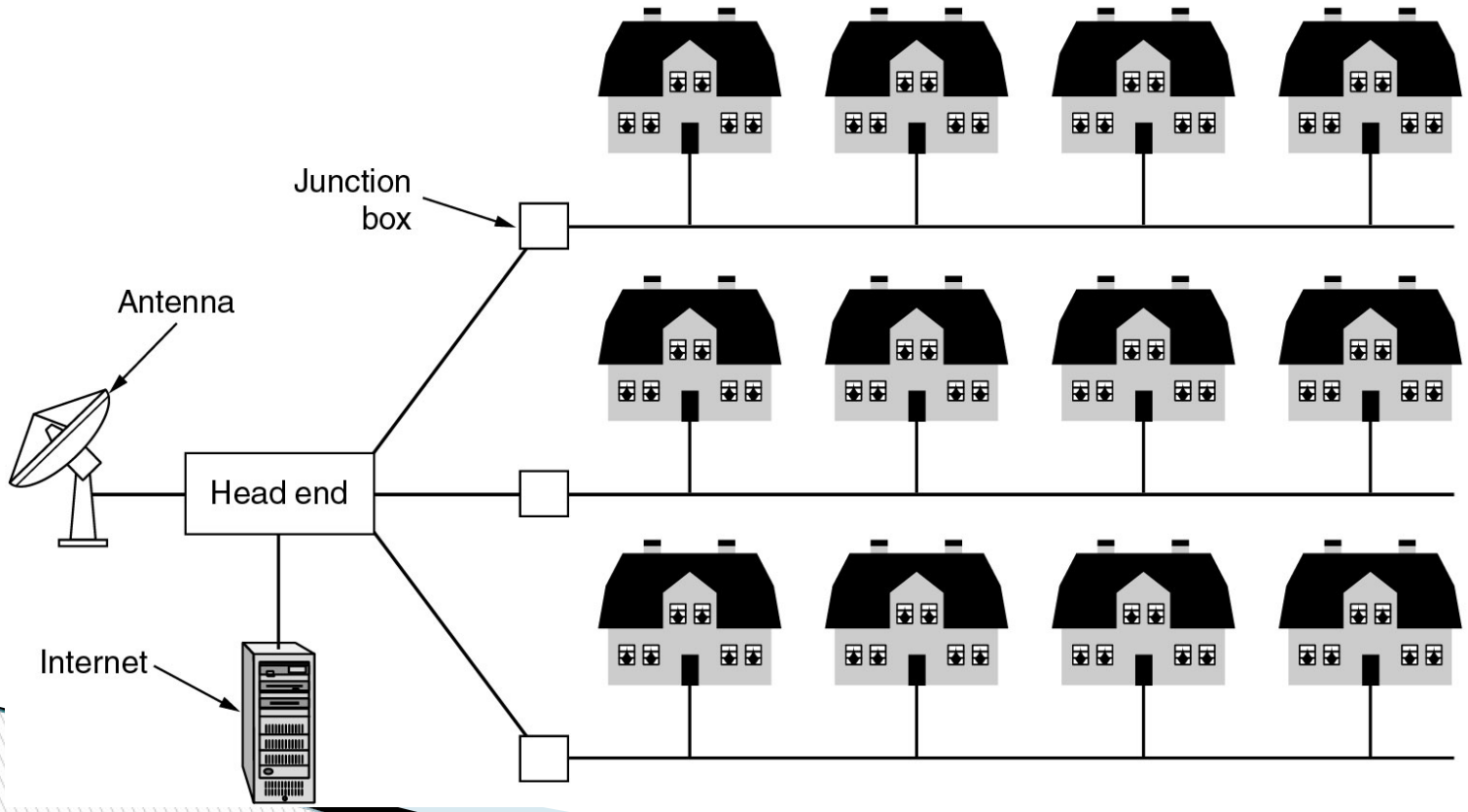
  - **Cable TV services have been extensively developed in most modern cities**
  - **Cable TV companies try to make use of their coaxial cable installed (that are supposed to carry TV signals) to deliver broadband data services**
  - **Many cable network wiring has been replaced with hybrid fiber-coax (HFC) — i.e. use of fiber-optic cable to connect to the subscribers' buildings, and then the original coaxial cable to connect to each household**

Metropolitan area network

- A *Metropolitan Area Network* (MAN) is a network that is utilized across multiple buildings
- Commonly used in school campuses or large
- companies with multiple buildings
- Is larger than a LAN, but smaller than a WAN
- Is also used to mean the interconnection of several LANs by bridging them together. This sort of network is also referred to as a *campus network*

Metropolitan Area Networks

- A metropolitan area network based on cable TV.



Peer-to-Peer Networks

- Peer-to-peer network is also called workgroup
- No hierarchy among computers ⇒ all are equal
- No administrator responsible for the network

- **Advantages of peer-to-peer networks:**
  - **Low cost**
  - **Simple to configure**
  - **User has full accessibility of the computer**

- **Disadvantages of peer-to-peer networks:**
  - **May have duplication in resources**
  - **Difficult to uphold security policy**

- **Where peer-to-peer network is appropriate:**
  - **10 or less users**
  - **No specialized services required**
  - **Security is not an issue**
  - **Only limited growth in the foreseeable future**

Clients and Servers

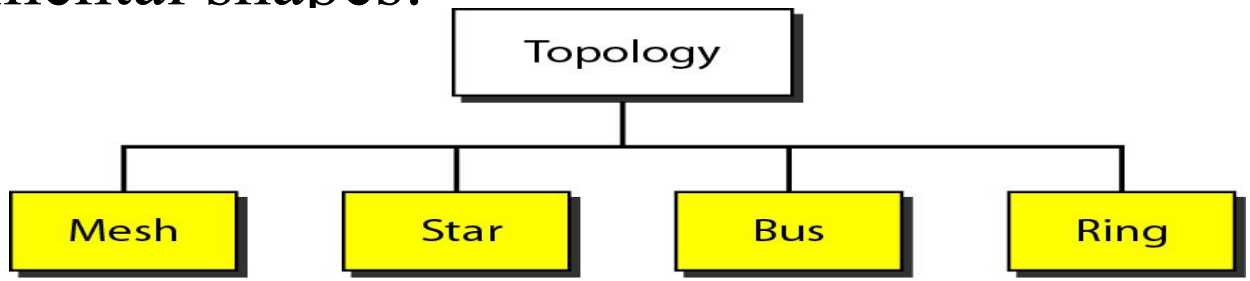
- Network Clients (Workstation)
  - Computers that request network resources or services
- Network Servers
  - Computers that manage and provide network resources and services to clients
    - Usually have more processing power, memory and hard disk space than clients
    - Run Network Operating System that can manage not only data, but also users, groups, security, and applications on the network
    - Servers often have a more stringent requirement on its performance and reliability

- **Advantages of client/server networks**
  - **Facilitate resource sharing – centrally administrate and control**
  - **Facilitate system backup and improve fault tolerance**
  - **Enhance security – only administrator can have access to Server**
  - **Support more users – difficult to achieve with peer-to-peer networks**

- **Disadvantages of client/server networks**
  - **High cost for Servers**
  - **Need expert to configure the network**
  - **Introduce a single point of failure to the system**

Simple Physical Topologies

- The term physical topology refers to the way in which a network is laid out physically.
- Three fundamental shapes:
  - Bus
  - Ring
  - Star

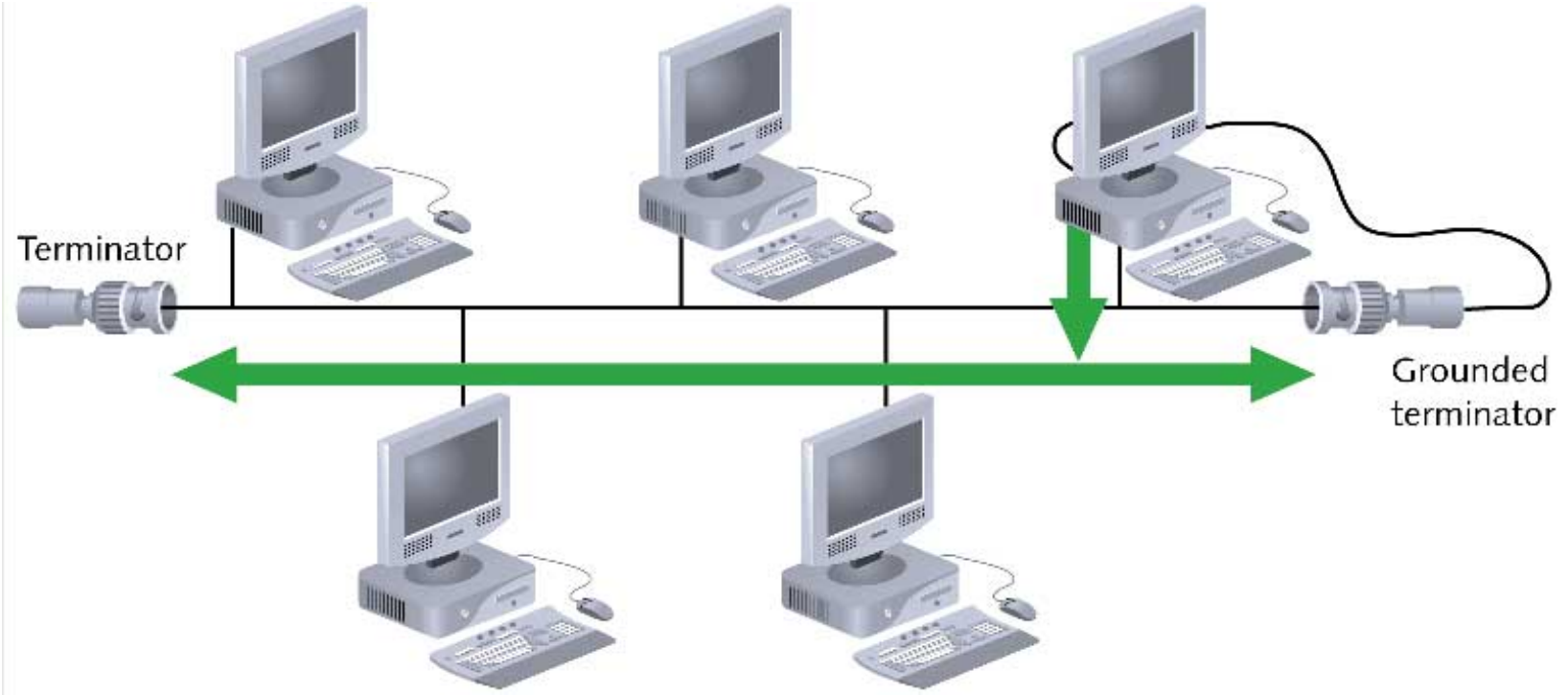


- May create hybrid topologies
- Topology integral to type of network, cabling infrastructure, and transmission media used

Bus

- A **Bus topology** consists of a single cable—called a **bus**— connecting all nodes on a network without intervening connectivity devices
- Single cable connects all network nodes without intervening connectivity devices
- Devices share responsibility for getting data from one point to another
- Terminators stop signals after reaching end of wire
  - Prevent signal bounce
- Inexpensive, not very scalable
- Difficult to troubleshoot, not fault-tolerant

Bus (continued)



Advantages of Bus Topology

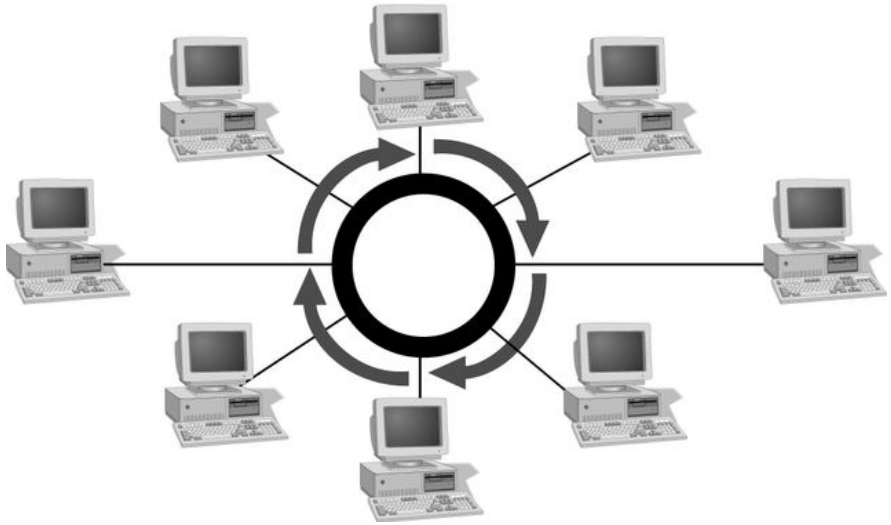
- Works well for small networks
- Relatively inexpensive to implement
- Easy to add to it

Disadvantages of Bus Topology

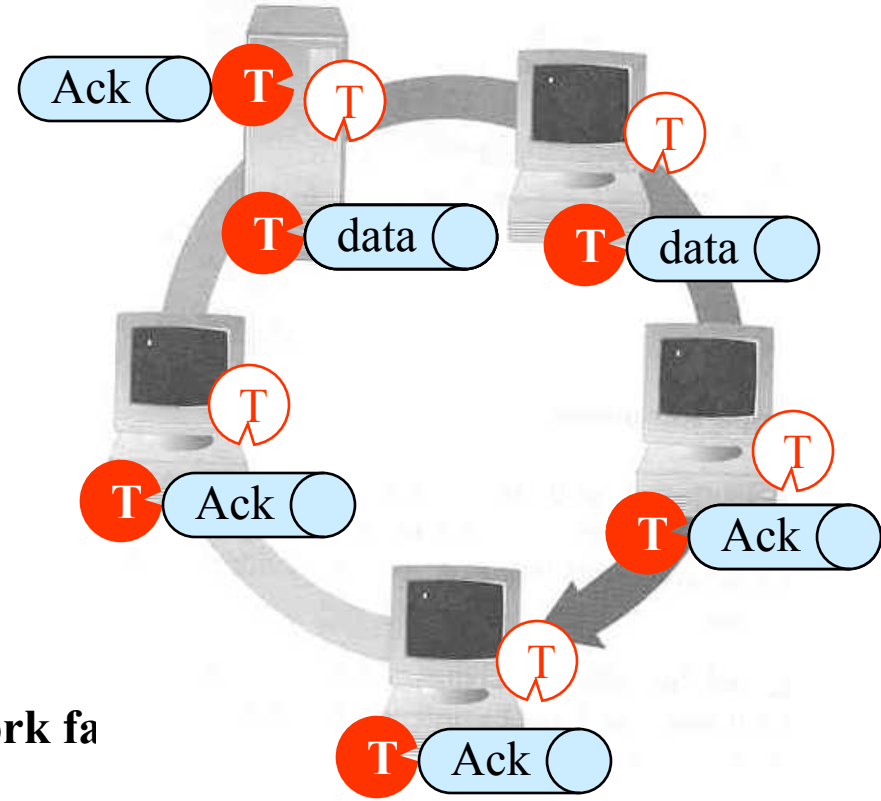
- Management costs can be high
- Potential for congestion with network traffic

Simple Physical Topologies

- Ring topology
  - Each node is connected to the two nearest nodes so the entire network forms a circle
  - One method for passing data on ring networks is **token passing**
- Active topology
  - Each workstation transmits data



- **Ring Topology**
  - Every computer serves as a repeater to boost signals
  - Typical way to send data:
    - Token passing
      - only the computer who gets the token can send data
  - Disadvantages
    - Difficult to add computers
    - More expensive
    - If one computer fails, whole network fa



Advantages of Ring Topology

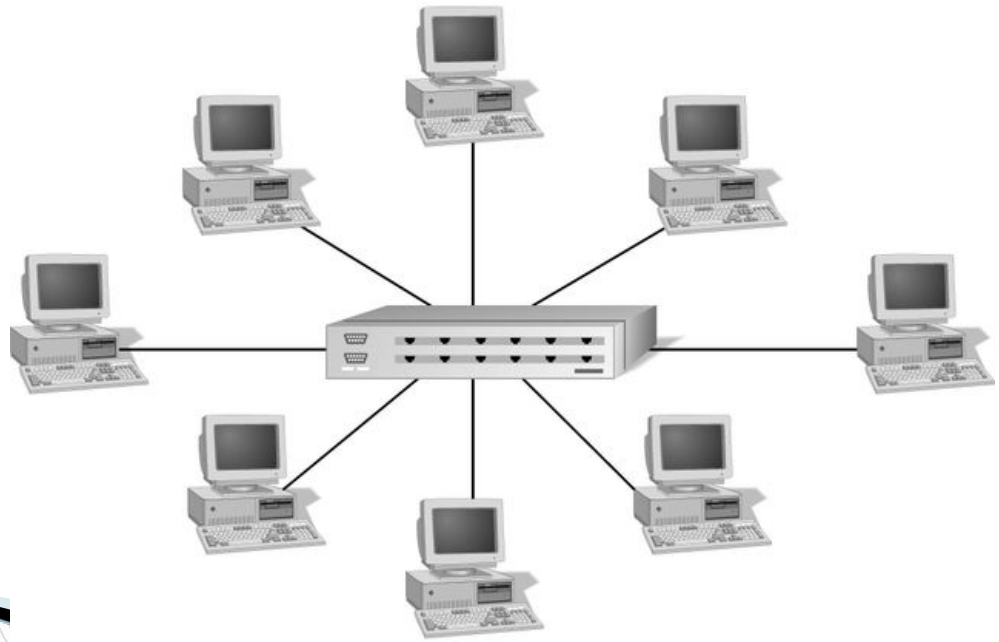
- Easier to manage; easier to locate a defective node or cable problem
- Well-suited for transmitting signals over long distances on a LAN
- Handles high-volume network traffic
- Enables reliable communication

Disadvantages of Ring Topology

- Expensive
- Requires more cable and network equipment at the start
- Not used as widely as bus topology
  - Fewer equipment options
  - Fewer options for expansion to high-speed communication

Simple Physical Topologies

- Star topology
  - Every node on the network is connected through a central device



Star (continued)

- Any single cable connects only two devices
  - Cabling problems affect two nodes at most
- Requires more cabling than ring or bus networks
  - More fault-tolerant
- Easily moved, isolated, or interconnected with other networks
  - Scalable
- Supports max of 1024 addressable nodes on logical network

Advantages of Star Topology

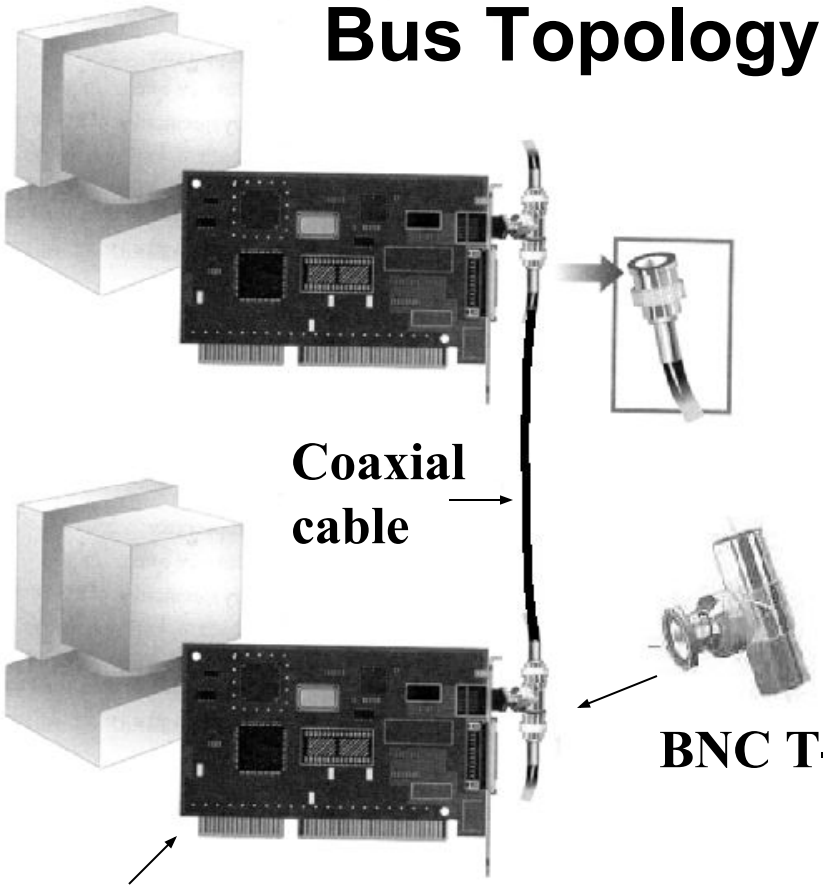
- Good option for modern networks
- Low startup costs
- Easy to manage
- Offers opportunities for expansion
- Most popular topology in use; wide variety of equipment available

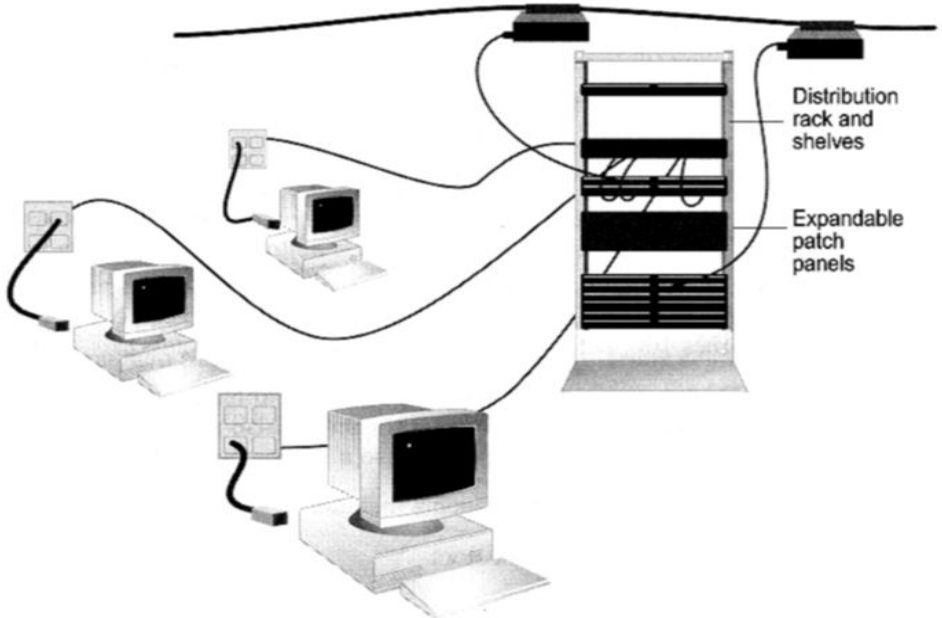
Disadvantages of Star Topology

- Hub is a single point of failure
- Requires more cable than the bus

How to construct a network with Bus / Star Topology?

Bus Topology





Star Topology

Other types

- **WLAN (Wireless LAN)**
- A LAN that uses high frequency radio waves for communication. Provides short range connectivity with high speed data transmission.
- **PAN (Personal Area Network)**

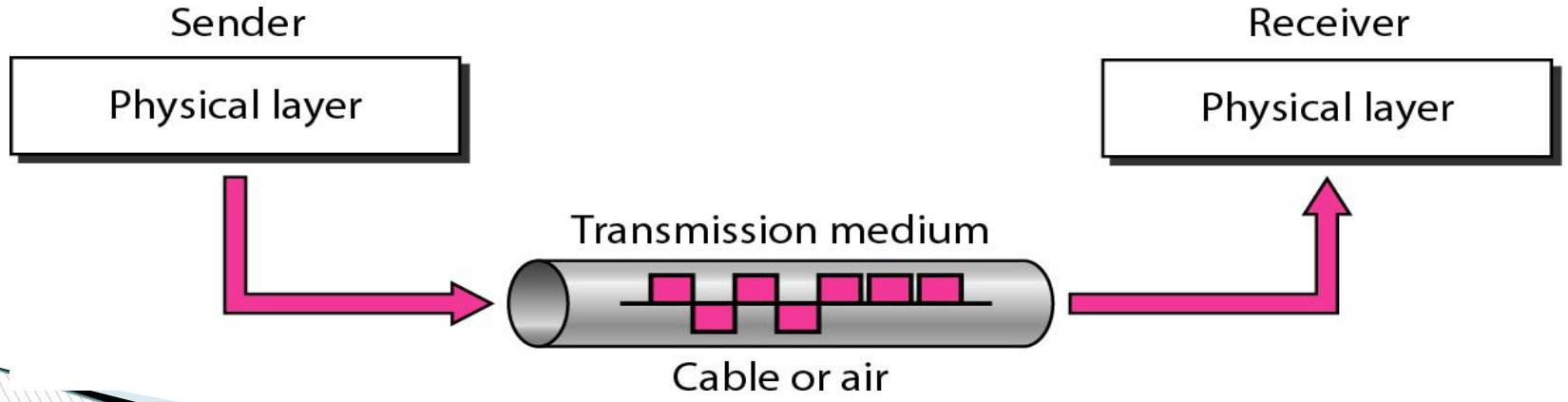
  Network organized by the individual user for its personal use.
- **SAN (Storage Area Network)**

  Connects servers to data storage devices via fiber- optic cables.

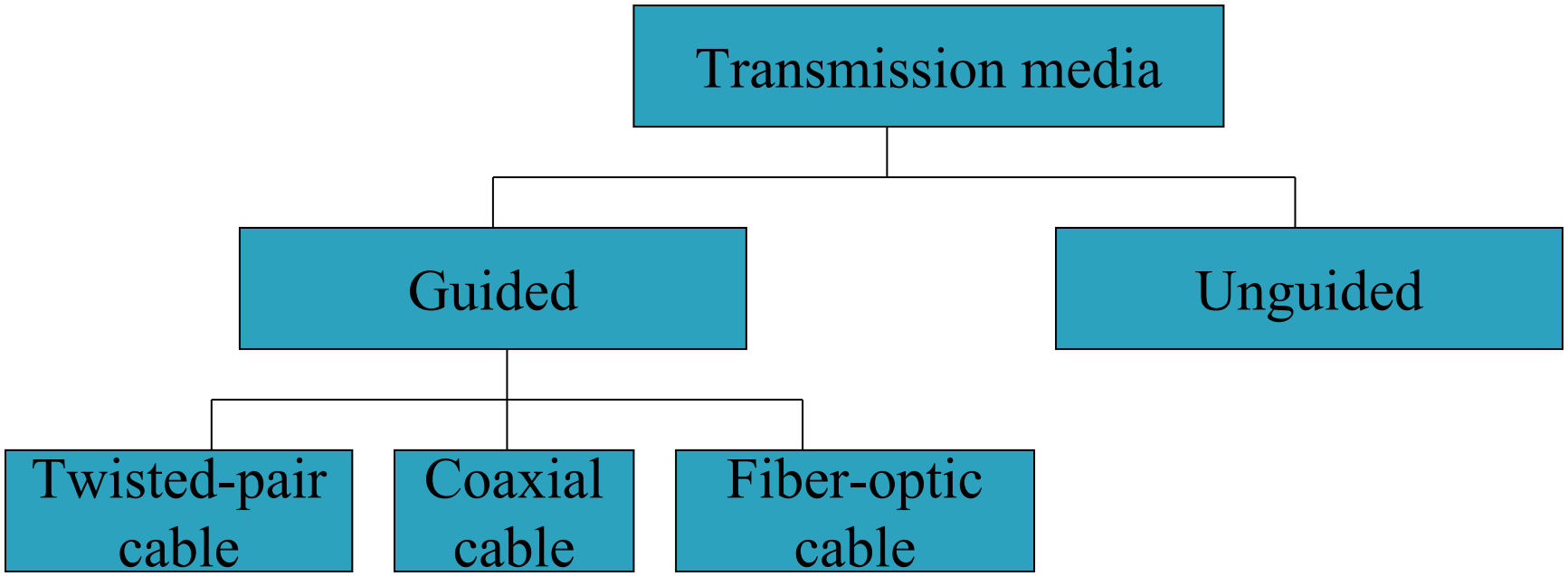
  Eg:-Used for daily backup of organization or a mirror copy

Transmission Media

- Conducted or guided media
  - use a conductor such as a wire or a fiber optic cable to move the signal from sender to receiver
- Wireless or unguided media
  - use radio waves of different frequencies and do not need a wire or cable conductor to transmit signals



Transmission Media

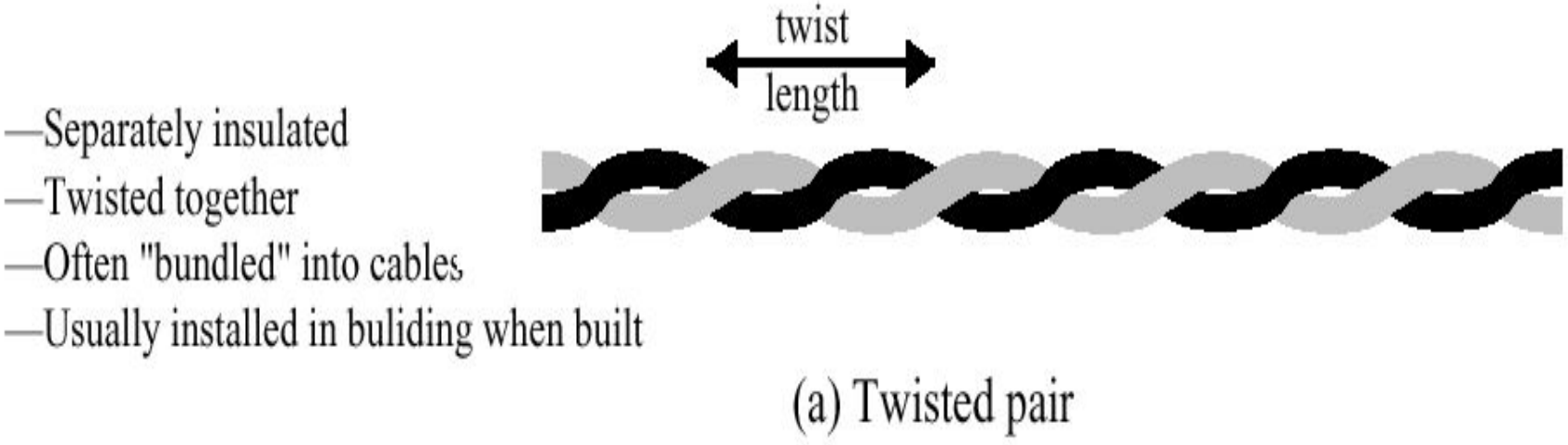


Wired - Twisted Pair

- The oldest, least expensive, and most commonly used media
- Pair of insulated wires twisted together to reduce susceptibility to interference : ex) capacitive coupling, crosstalk
- Skin effect at high frequency
- Up to 250 kHz analog and few Mbps digital signaling (for long-distance point-to-point signaling)
- Need repeater every 2-3 km (digital), and amplifier every 5-6 km (analog)

Twisted Pair

- Consists of two insulated copper wires arranged in a regular spiral pattern to minimize the electromagnetic interference between adjacent pairs
- Often used at customer facilities and also over distances to carry voice as well as data communications
- Low frequency transmission medium
- Telephone (subscriber loop: between house and local exchange)
- High-speed (10 - 100 Mbps) LAN :
  - token ring, fast - Ethernet



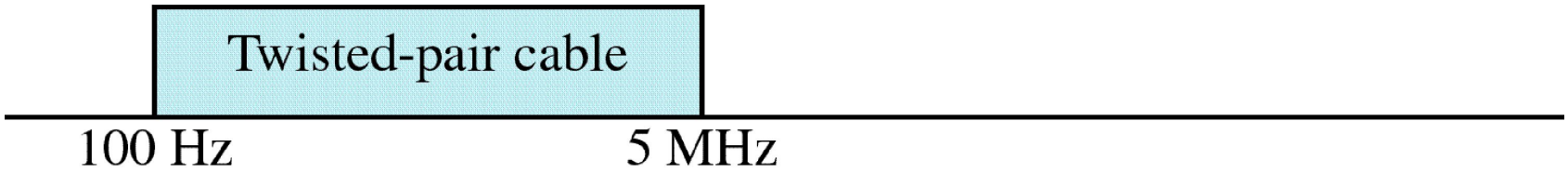
Types of Twisted Pair

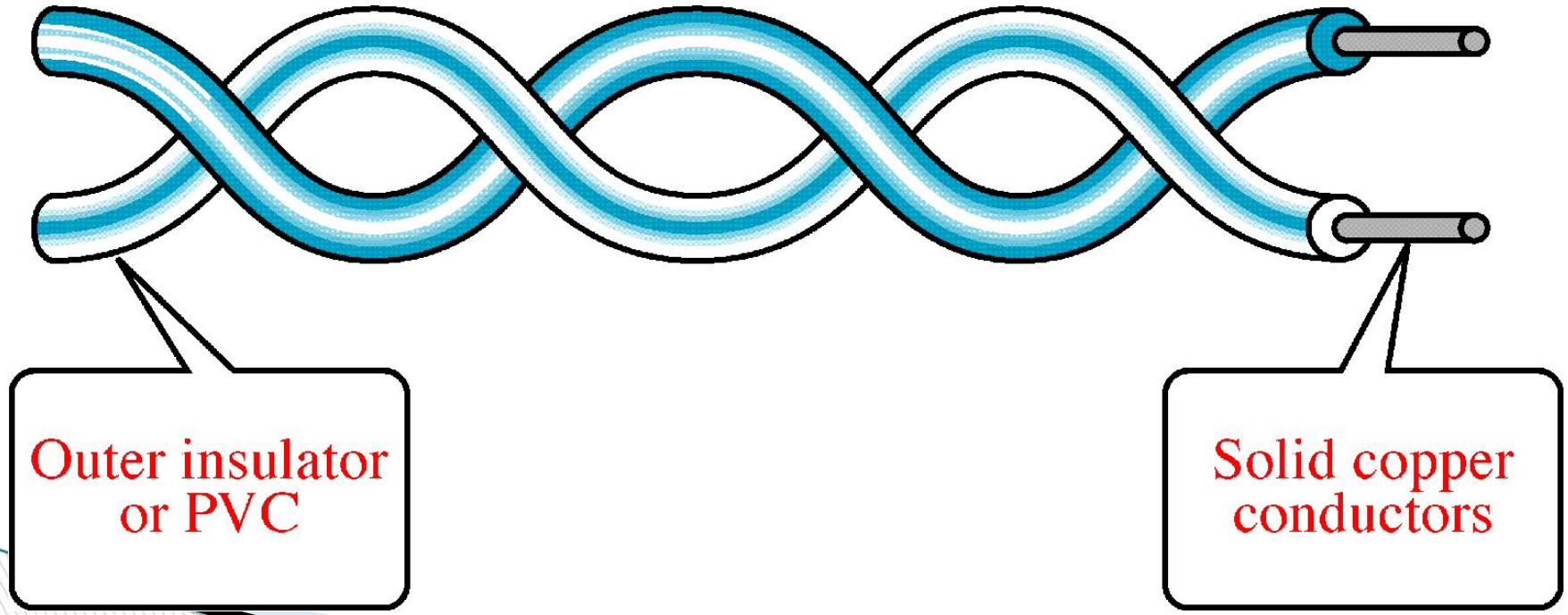
- STP (shielded twisted pair)
  - the pair is wrapped with metallic foil or braid to insulate the pair from electromagnetic interference
- UTP (unshielded twisted pair)
  - each wire is insulated with plastic wrap, but the pair is encased in an outer covering

Twisted Pair Advantages

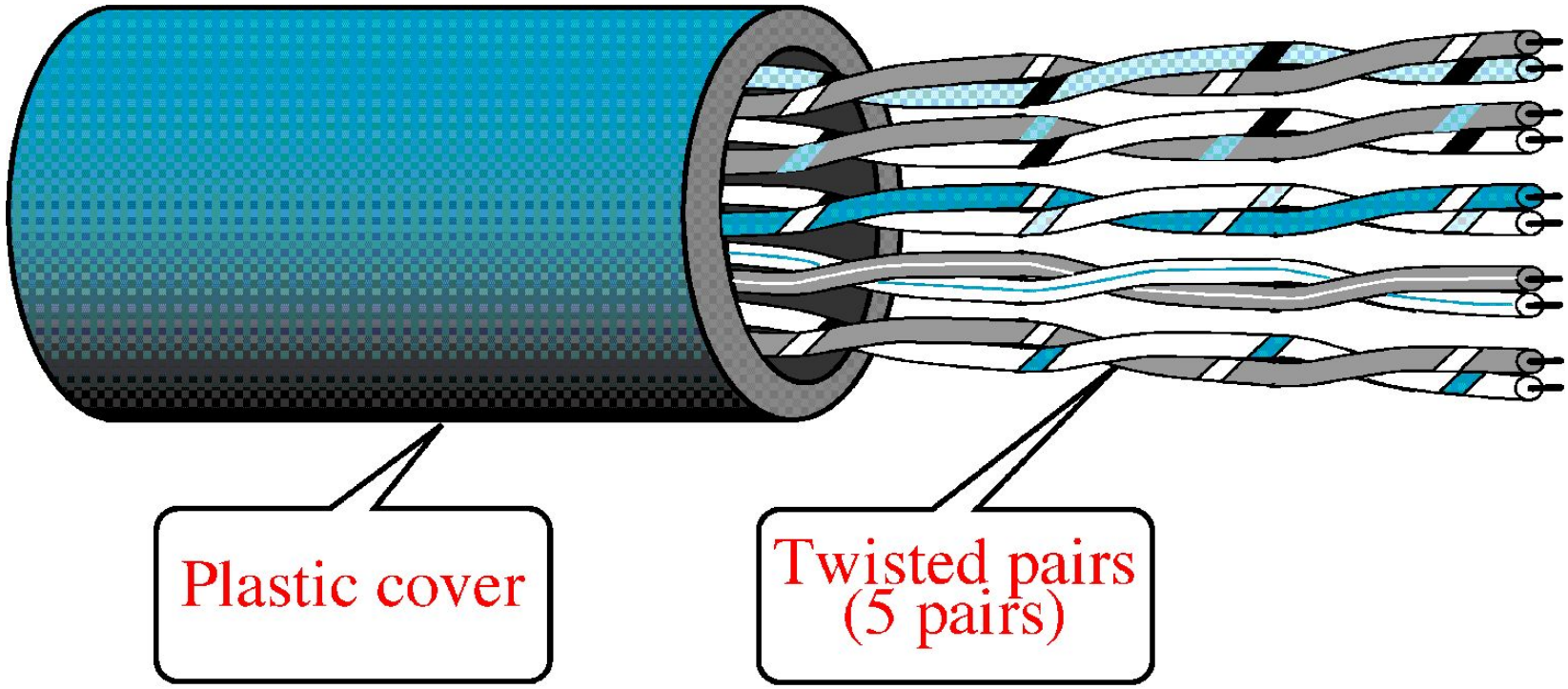
- Advantages
  - Inexpensive and readily available
  - Flexible and light weight
  - Easy to work with and install
- Disvantages
  - Susceptibility to interference and noise
  - Attenuation problem
    - For analog, repeaters needed every 5-6km
    - For digital, repeaters needed every 2-3km
  - Relatively low bandwidth (MHz)

Twisted-Pair Cable

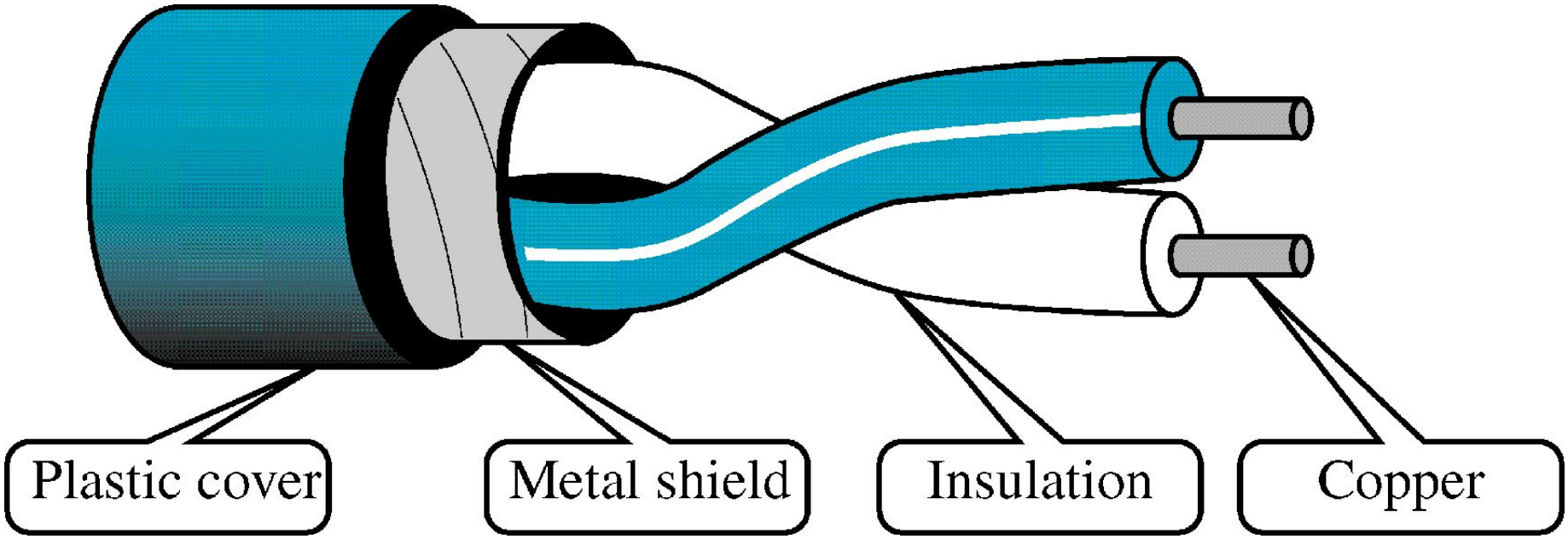




Unshielded Twisted-Pair Cable

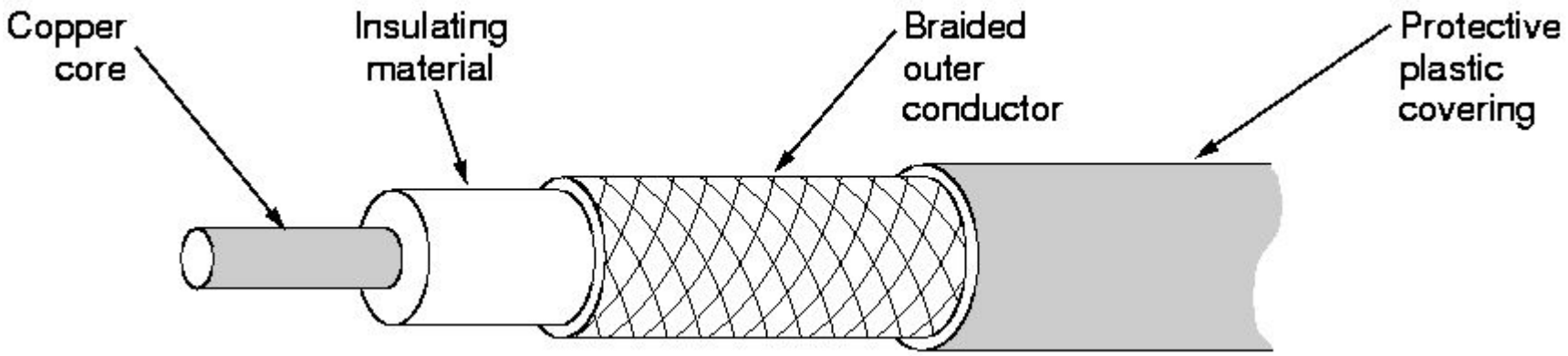


Shielded Twisted-Pair Cable



Coaxial cable

- Signal and ground wire
  - Solid center conductor running coaxially inside a solid (usually braided) outer circular conductor.
  - Center conductor is shielded from external interference signals.



Properties of coaxial cable

- Better shielding allows for longer cables and higher transfer rates.
- 100 m cables
  - 1 to 2 Gbps feasible (modulation used)
  - 10 Mbps typical
- Higher bandwidth
  - 400 to 600Mhz
  - up to 10,800 voice conversations
- Can be tapped easily: Station easily added (pros and cons)
- Much less susceptible to interference than twisted pair

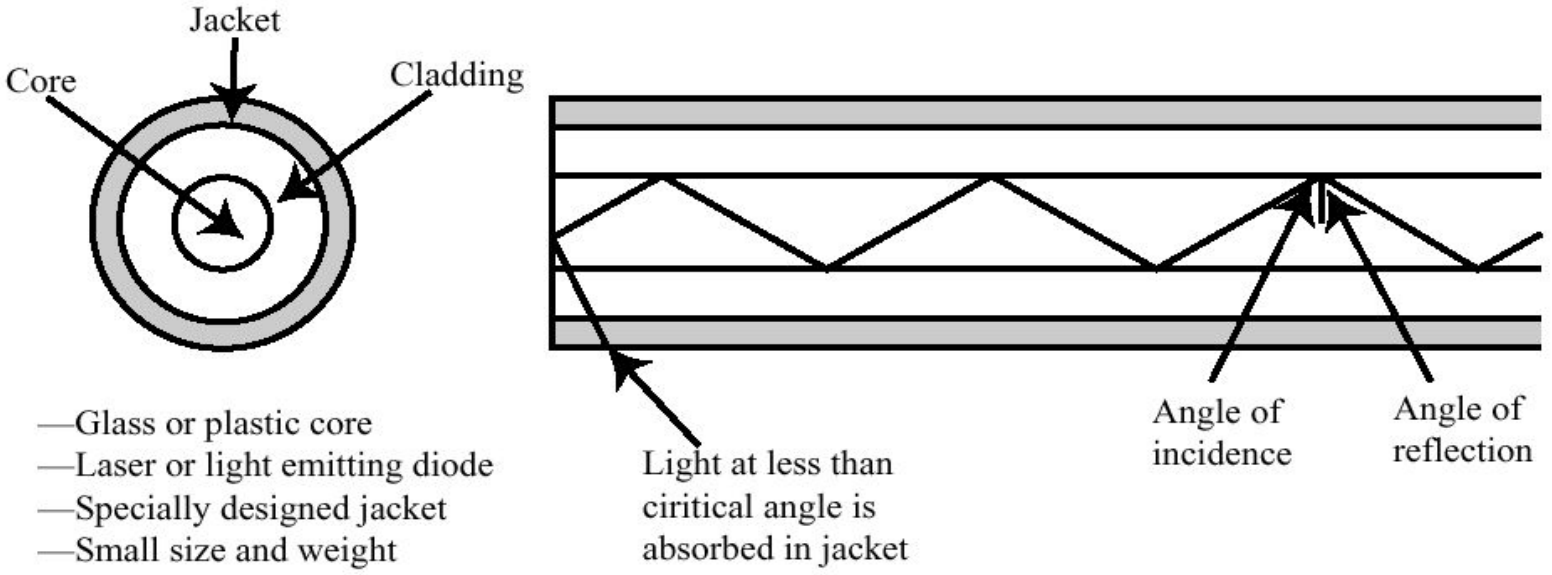
- High attenuation rate makes it expensive over long distance
- Bulky

Wired Transmission Media

- Optical Fiber
  - Flexible, thin (few to few hundred μm), very pure glass/plastic fiber capable of conducting optical rays
  - Extremely high bandwidth : capable of ≥ 2 Gbps
  - Very high noise immunity, resistant to electromagnetic interference
  - Does not radiate energy/cause interference
  - Very light
  - Need repeaters only 10's or 100 km apart
  - Very difficult to tap : Better security but multipoint not easy
  - Require a light source with injection laser diode (ILD) or light-emitting diodes (LED)

Wired Transmission Media

- Optical Fiber (Cont'd)
  - Need optical-electrical interface (more expensive than electrical interface)

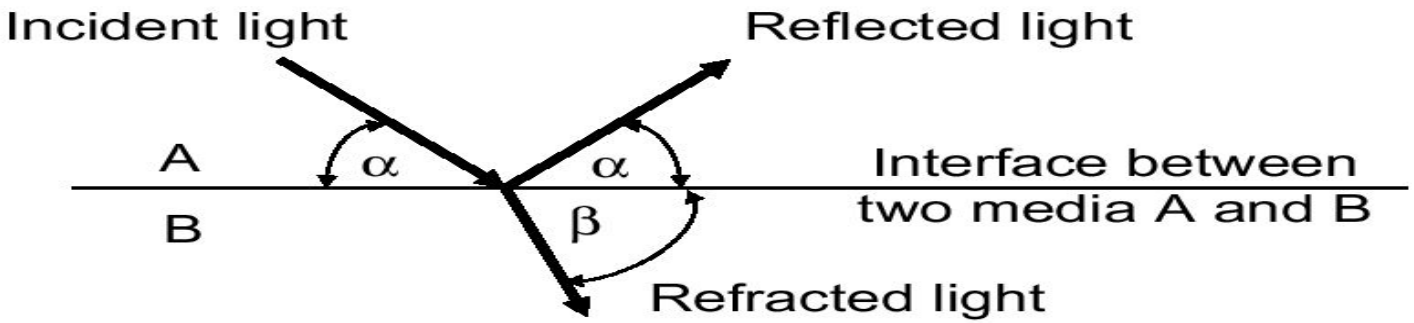


(c) Optical Fiber

Wired Transmission Media

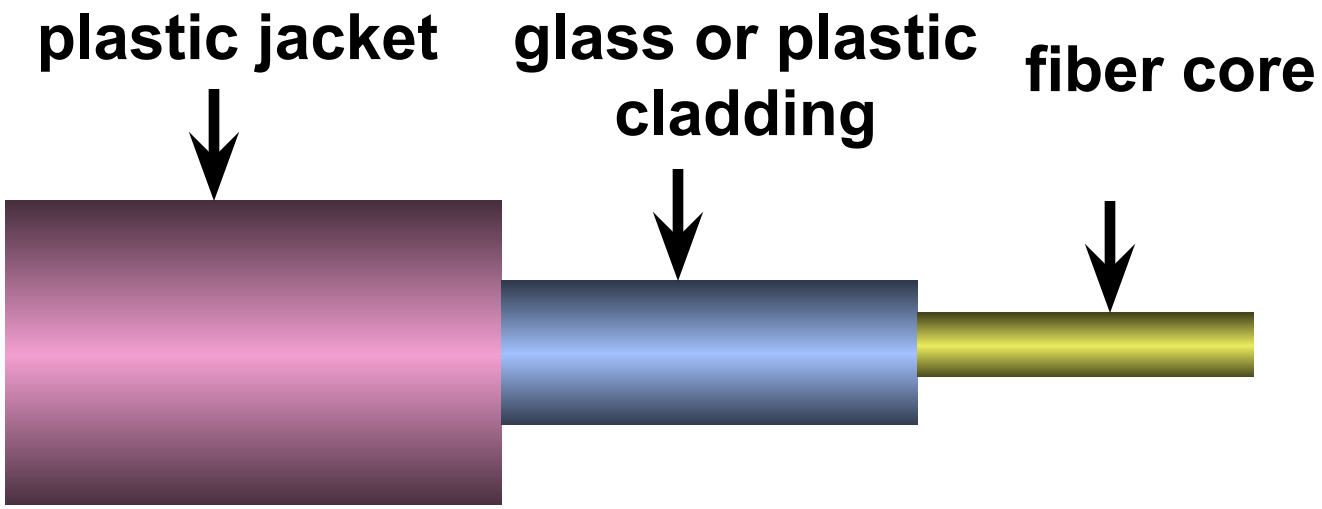
Optical Fiber

- Principle of optical fiber transmission: Based on the principle of total internal reflection



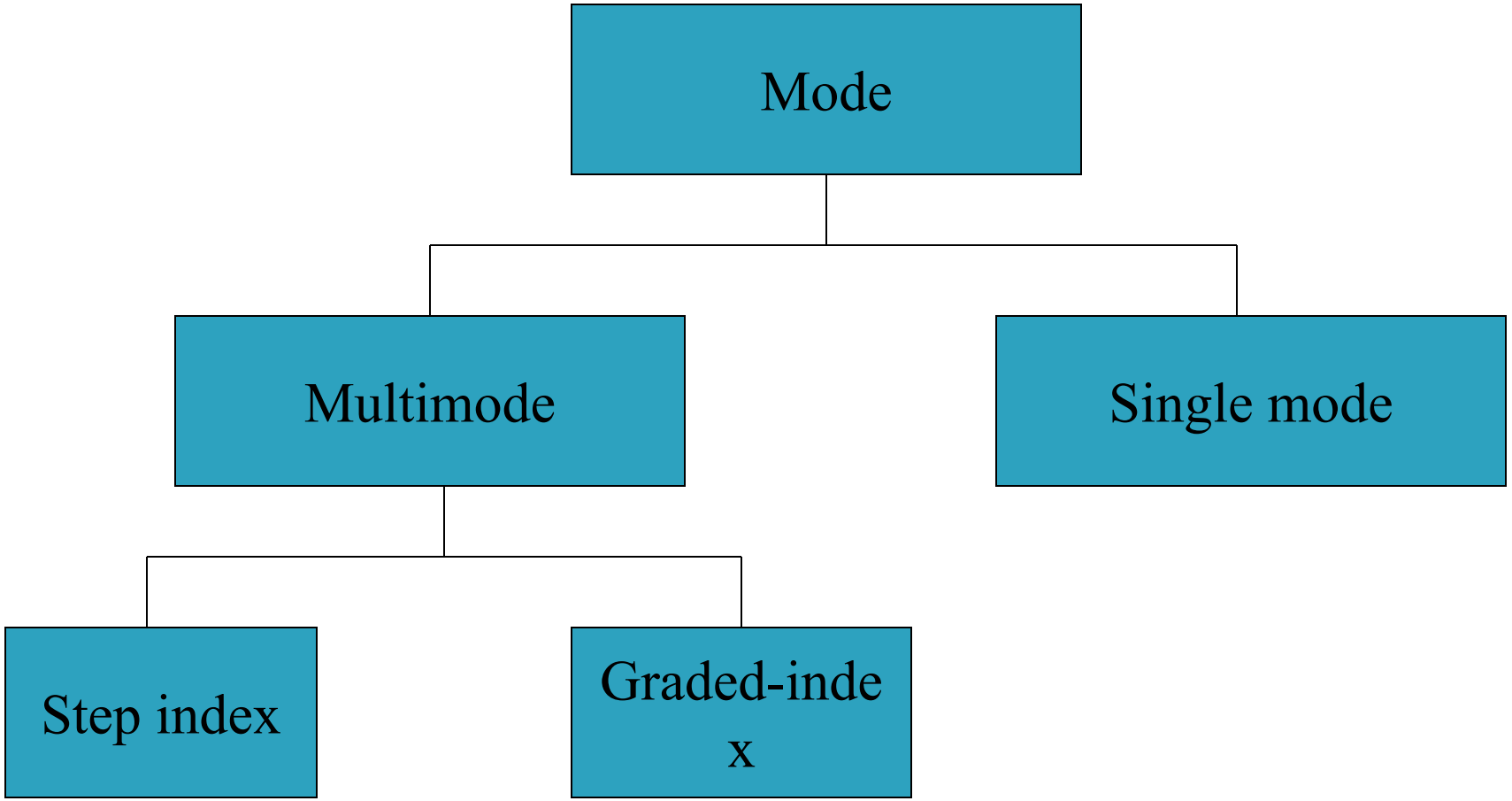
Fiber Optic Layers

- consists of three concentric sections



Modes of fiber

- Fiber consists of two parts: the *glass core* and *glass cladding* with a lower refractive index.
- Light propagates in 1 of 3 ways depending on the type and width of the core material.
  - **Multimode stepped index fiber**
    - Both core and cladding have different but uniform refractive index.
    - Relies on total internal reflection; Wide pulse width.
  - **Multimode graded index fiber**
    - Core has variable refractive index (light bends as it moves away from core).
    - Narrow pulse width resulting in higher bit rate.
  - **Singlemode fiber (> 100 Mbs)**
    - Width of core diameter equal to a single wavelenth.

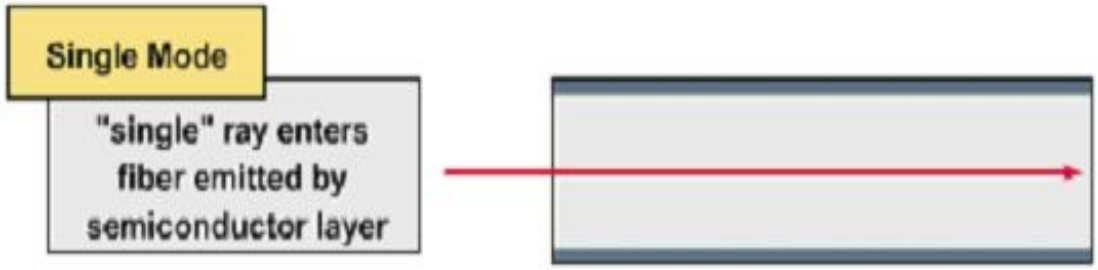

Mode
Multimode
Single mode
Step index
Graded-inde
x

Fiber Optic Types

- multimode step-index fiber
  - the reflective walls of the fiber move the light pulses to the receiver
- multimode graded-index fiber
  - acts to refract the light toward the center of the fiber by variations in the density
- single mode fiber
  - the light is guided down the center of an extremely narrow core

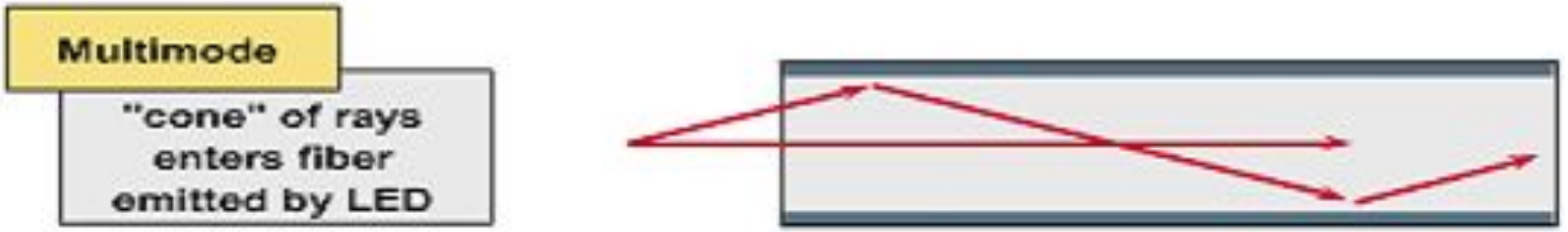
Types of optical fiber

- Modes, bundles of light rays enter the fiber at a particular angle
- Single-mode
  - Also known as mono-mode
  - Only one mode propagates through fiber
  - Higher bandwidth than multi-mode
  - Longer cable runs than multi-mode
  - Lasers generate light signals
  - Used for inter-building connectivity

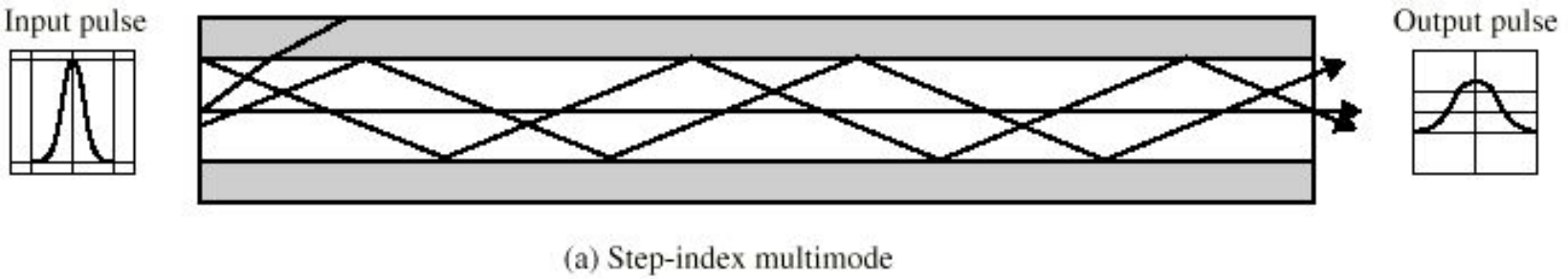


Types of optical fiber

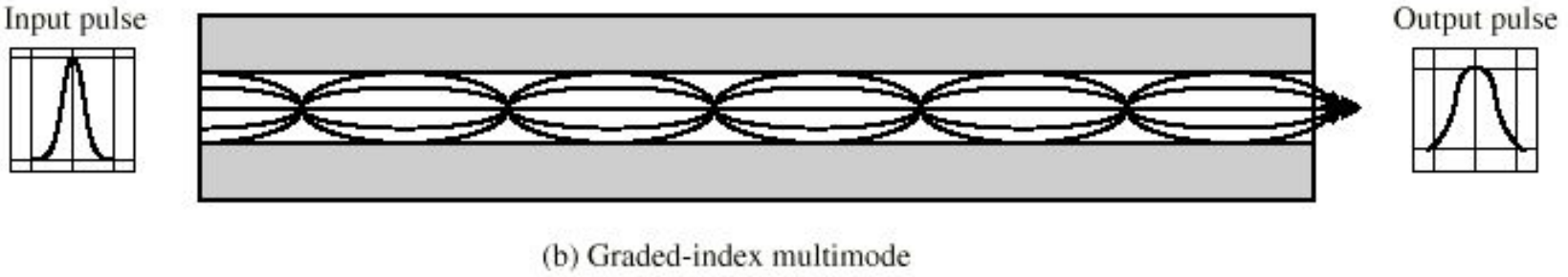
- Multi-mode
  - Multiple modes propagate through fiber
  - Different angles mean different distances to travel
    - Transmissions arrive at different times
    - Modal dispersion
  - LEDs as light source
  - Used for intra-building connectivity



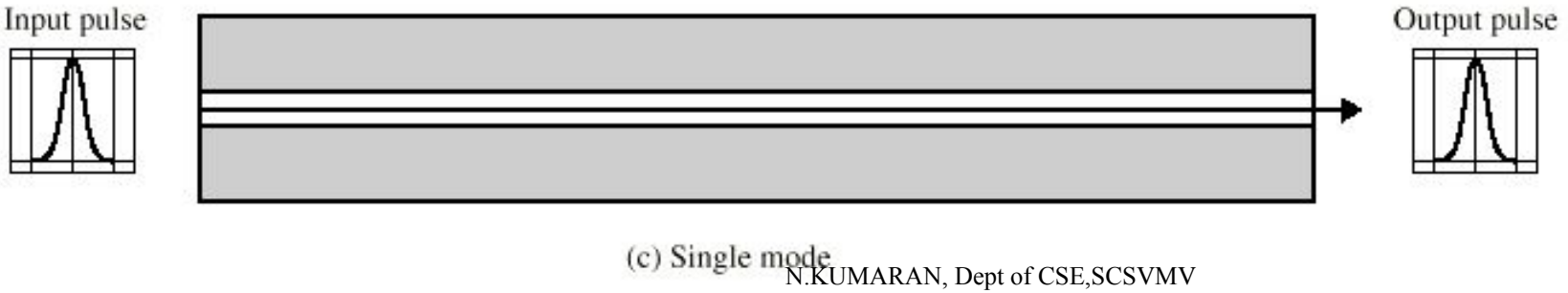
Optical Fiber Transmission Mode



(a) Step-index multimode



(b) Graded-index multimode



(c) Single mode

Fiber Optic

Advantages

- greater capacity (bandwidth Gbps)
- smaller size and lighter weight
- lower attenuation
- immunity to environmental interference
- highly secure due to tap difficulty and lack of signal radiation

Disvantages

- expensive over short distance
- requires highly skilled installers
- adding additional nodes is difficult

Wireless (Unguided Media) Transmission

- transmission and reception are achieved by means of an antenna
- directional
  - transmitting antenna puts out focused beam
  - transmitter and receiver must be aligned
- omnidirectional
  - signal spreads out in all directions
  - can be received by many antennas

Wireless Transmission

Infrared

- For short-range communication
- Remote controls for TVs, VCRs and stereos
- IRD port
- Indoor wireless LANs
- Do not pass through solid walls
- Better security and no interference (with a similar system in adjacent rooms)
- No government license is needed
- Cannot be used outdoors

Infrared

- Transceivers must be within line of sight of each other (directly or via reflection)
- Unlike microwaves, infrared does not penetrate walls
- Fairly low bandwidth (4 Mbps).
- Uses wavelengths between microwave and visible light.
- Uses transmitters/receivers (transceivers) that modulate noncoherent infrared light.
- No frequency allocation issue since not regulated.
- Uses include local building connections, wireless LANs, and new wireless peripherals.

Wireless Transmission

Terrestrial Microwave

- Parabolic dish
- Focused beam
- Line of sight
- Long haul telecommunications
- Higher frequencies give higher data rates

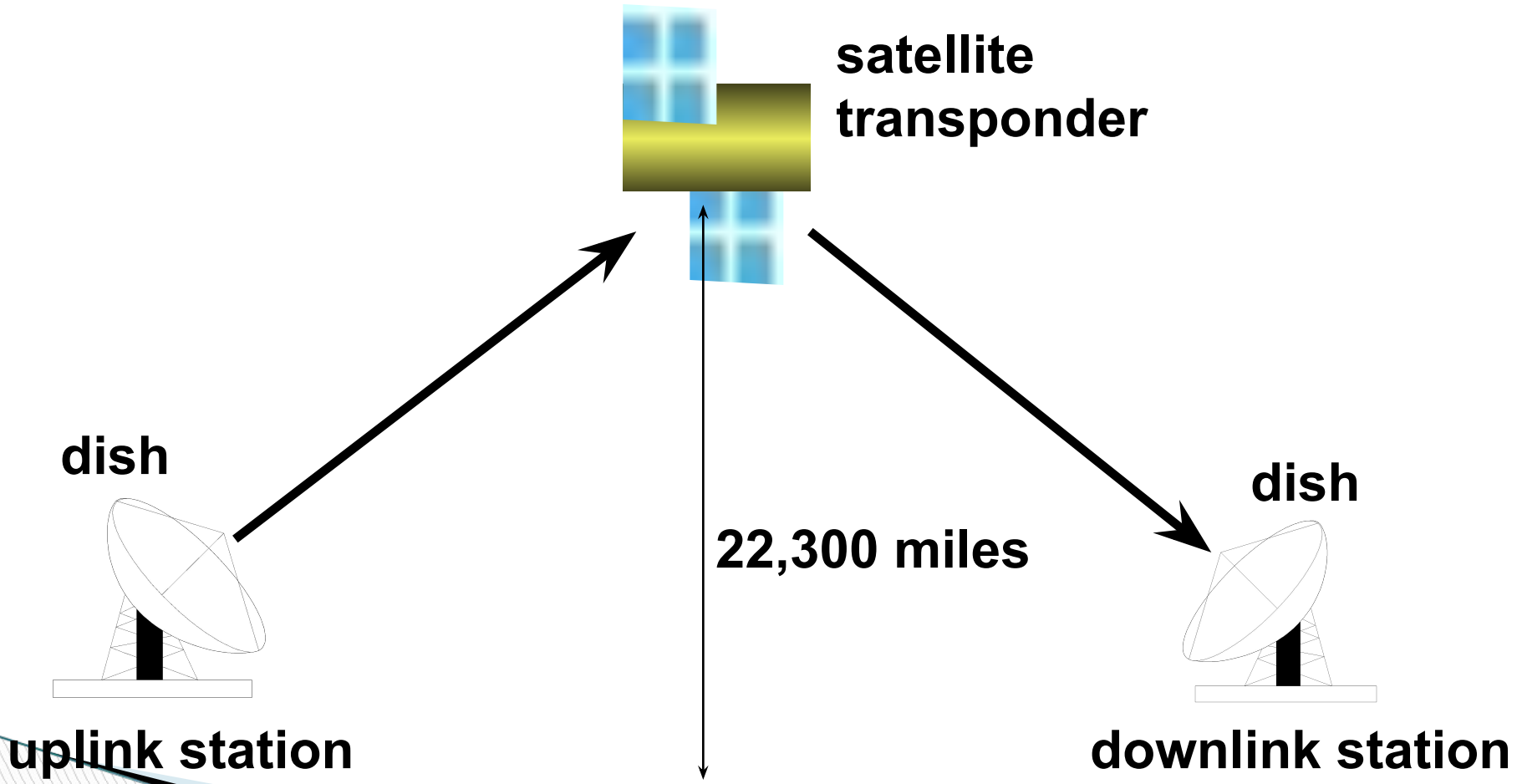
Terrestrial Microwave

- used for long-distance telephone service
- uses radio frequency spectrum, from 2 to 40 Ghz
- parabolic dish transmitter, mounted high
- used by common carriers as well as private networks
- requires unobstructed line of sight between source and receiver
- curvature of the earth requires stations (repeaters) ~30 miles apart

Radio Transmission

- Radio waves
  - Easy to generate, travel long distances, and penetrate buildings easily.
  - Omnidirectional.
  - Low frequencies
    - Pass through obstacles well,
    - Quick power drop off (e.g. 1/r3 in air).
  - High frequencies
    - Travel in straight lines and bounce off obstacles.
    - Absorbed by rain.
  - Subject to electrical interference

Satellite Transmission Process



Satellite Transmission Applications

- television distribution
  - a network provides programming from a central location
  - direct broadcast satellite (DBS)
- long-distance telephone transmission
  - high-usage international trunks
- private business networks

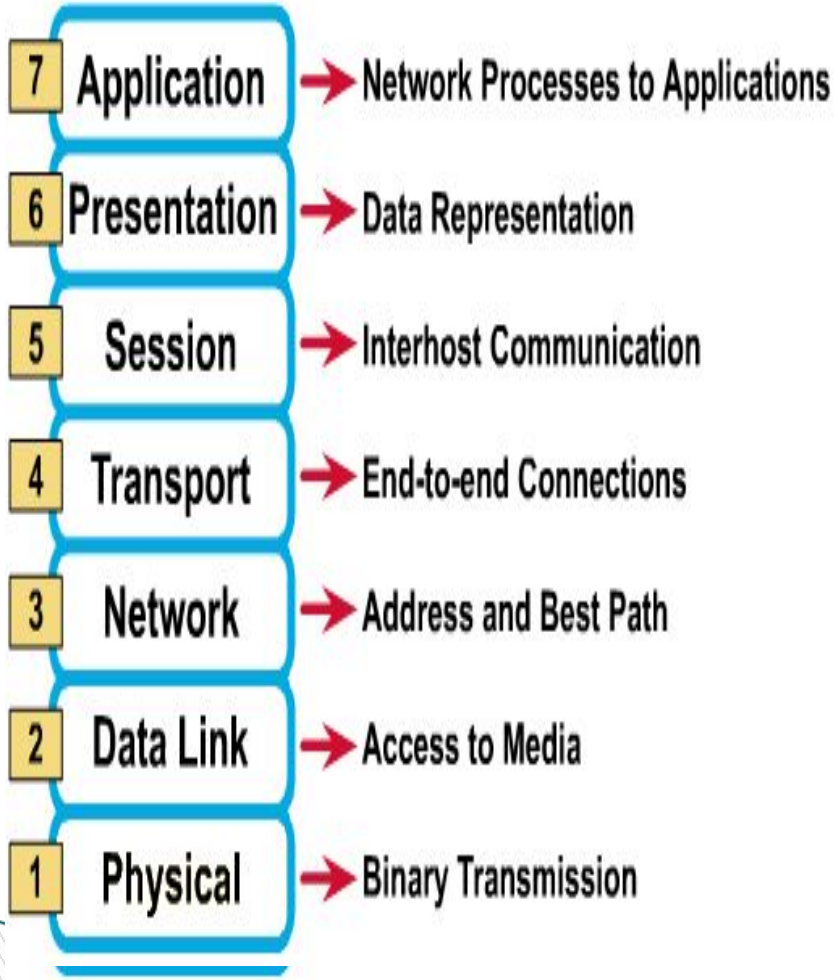
OSI Reference Model

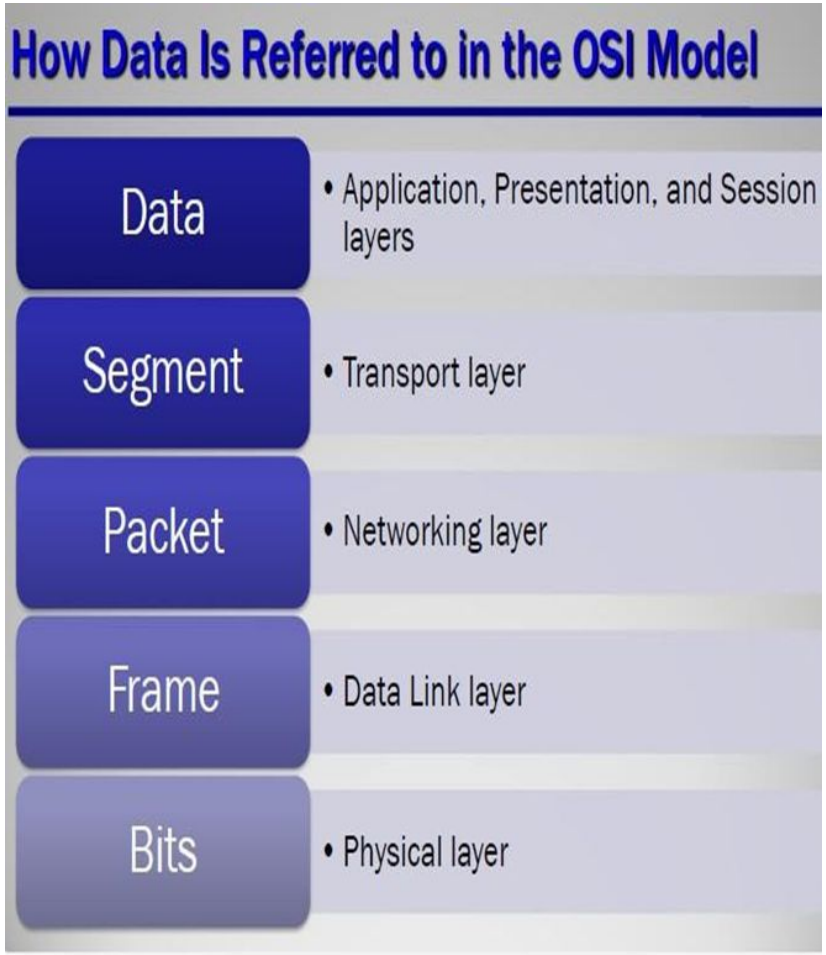
- Specific Functional Objectives On Completion of this OSI Reference Model
- The students will be able to:
- State the requirement for layered approach
- Explain the basic concept of layering in the network model •
- Define entities protocols in networking context •
- Describe ISO's OSI Reference Model •
- Explain information flow in OSI references Model.
- Explain functions of the seven layers of OSI Model

OSI Reference Model

- The OSI model is now considered the primary Architectural model for inter-computer communications.
- The OSI reference model divides the problem of moving information between computers over a network medium into SEVEN smaller and more manageable problems .
- This separation into smaller more manageable functions is known as layering.

OSI Reference Model: 7 Layers





OSI: A Layered Network Model

- The process of breaking up the functions or tasks of networking into layers reduces complexity.
- Each layer provides a service to the layer above it in the protocol specification.
-
- The lower 4 layers (transport, network, data link and physical —Layers 4, 3, 2, and 1) are concerned with the flow of data from end to end through the network.
- The upper four layers of the OSI model (application, presentation and session—Layers 7, 6 and 5) are orientated more toward services to the applications.
- Data is Encapsulated with the necessary protocol information as it moves down the layers before network transit.

Physical Layer

- Provides physical interface for transmission of information.
- Representation of bits
- Date Rate or transmission rate, the number of bits transfer per second
- Line Configuration- point-to-point or multipoint configuration
- Physical topology
- Synchronization of bits-synchronize the bit level otherwise the sender and receiver clock must be synchronize
- Defines rules by which bits are passed from one system to another on a physical communication medium.
- Covers all - mechanical, electrical, functional and procedural - aspects for physical communication.

Data Link Layer

- Data link layer attempts to provide reliable communication over the physical layer interface.
- breaks the outgoing data into frames and reassemble the received frames.
- Create and detect frame boundaries.
- Handle errors by implementing an acknowledgement and retransmission scheme.
- Implement flow control.
- Physical address or MAC
- Supports simplex, half-duplex or full-duplex communication.

Network Layer

- Implements routing of frames (packets) through the network.

- Defines the most optimum path the packet should take from the source to the destination
- Defines logical addressing so that any endpoint can be identified.
- Handles congestion in the network.
- Facilitates interconnection between heterogeneous networks (Internetworking).
- The network layer also defines how to fragment a packet into smaller packets to accommodate different media.

Transport Layer

- Purpose of this layer is to provide a reliable mechanism for the exchange of data between two processes in different computers.

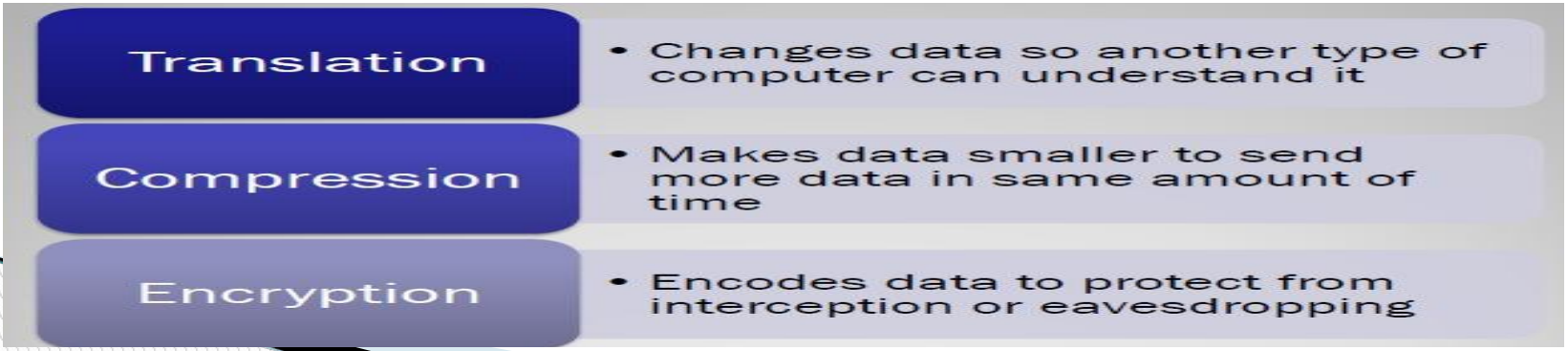
- Ensures that the data units are delivered error free.
- Ensures that data units are delivered in sequence(segmentation and reassembling).
- Ensures that there is no loss or duplication of data units.
- Provides connectionless or connection oriented service.
- Provides for the connection management.
- Multiplex multiple connection over a single channel.

Session Layer

- Session layer provides mechanism for controlling the dialogue between the two end systems. It defines how to start, control and end conversations (called sessions) between applications.
- This layer requests for a logical connection to be established on an end-user's request.
- Any necessary log-on or password validation is also handled by this layer.
- Session layer is also responsible for terminating the connection.
- This layer provides services like dialogue discipline which can be full duplex or half duplex.
- Session layer can also provide check-pointing mechanism such that if a failure of some sort occurs between checkpoints, all data can be retransmitted from the last checkpoint.

Presentation Layer

- It concerns with the syntax and semantics of the information exchanged between two systems
- Presentation layer defines the format in which the data is to be exchanged between the two communicating entities.
- Handles data compression and decompression
- Data encryption and decryption (cryptography).

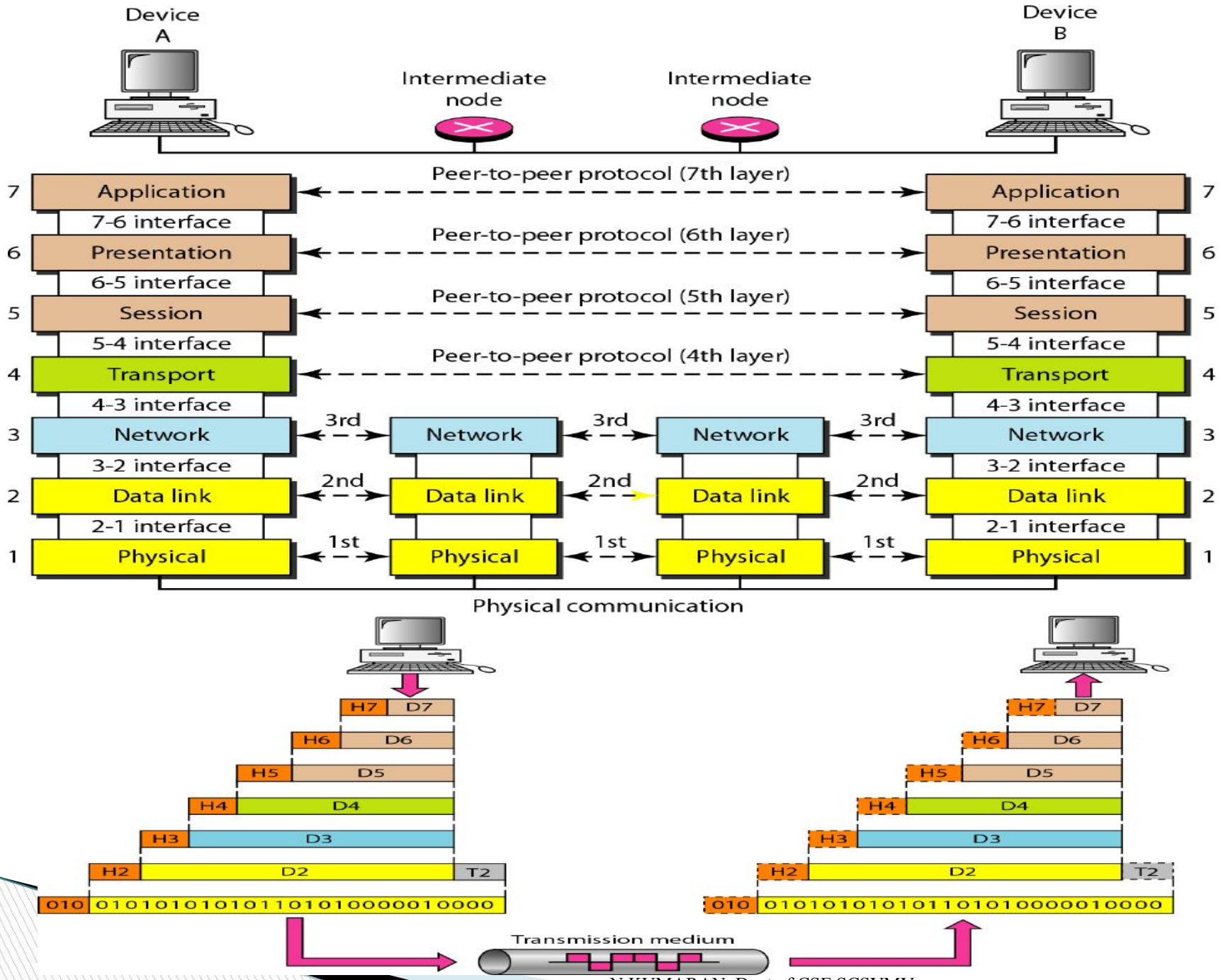


Application Layer

1. Application layer interacts with application programs and is the highest level of OSI model.

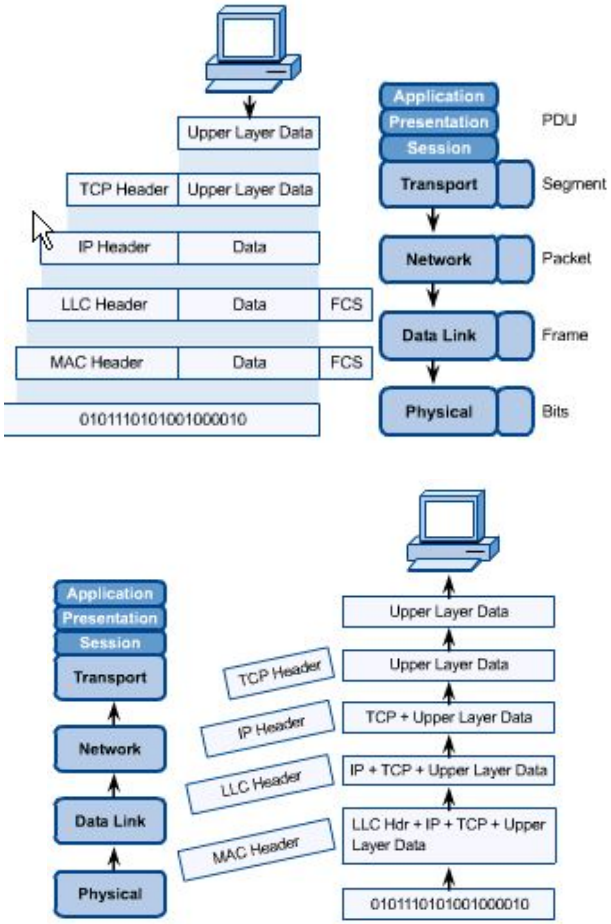
2. Application layer contains management functions to support distributed applications.

3. Examples of application layer are applications such as file transfer, electronic mail, remote login etc.

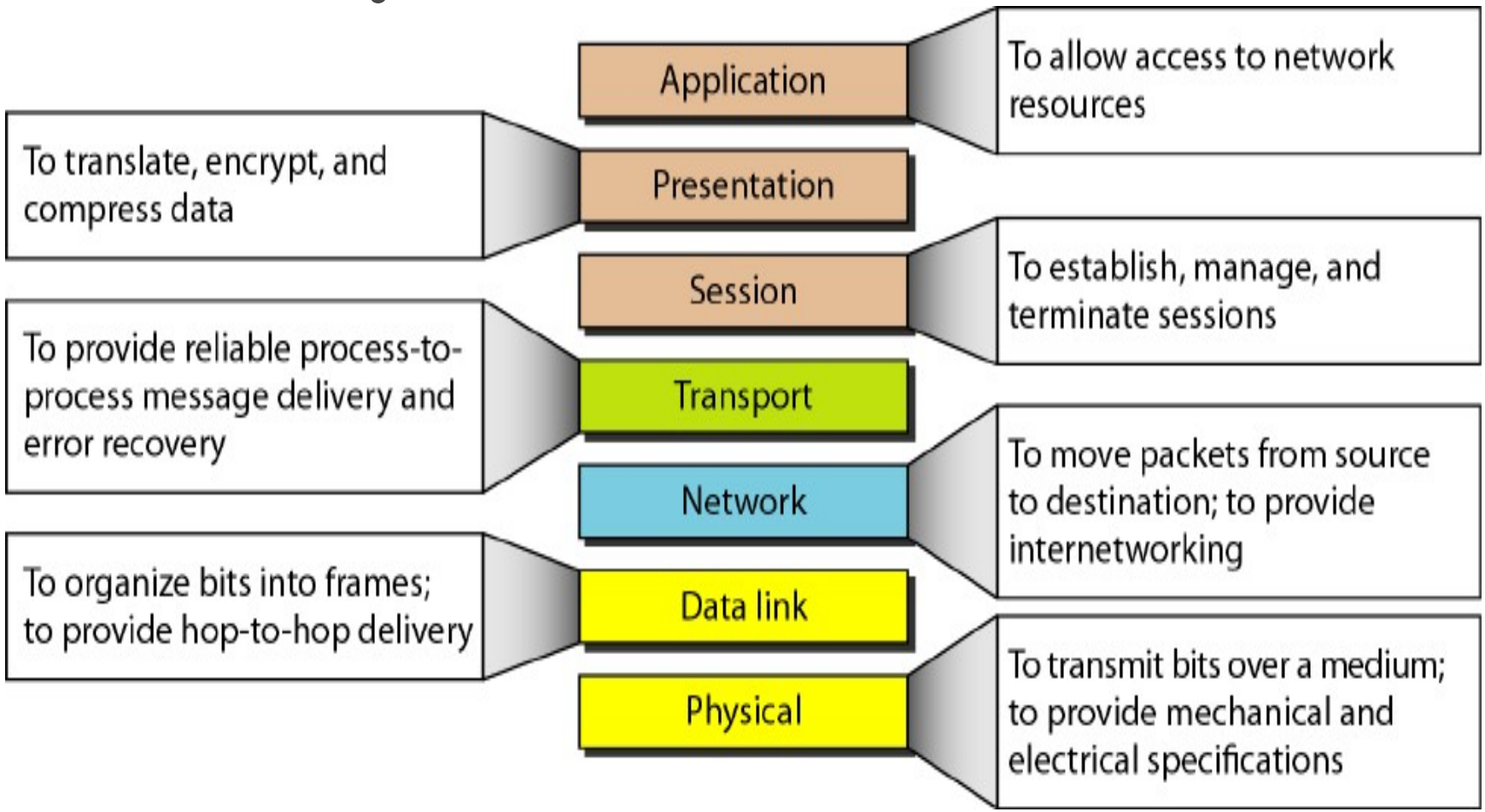

Device A
Device B
Intermediate node
Intermediate node
Peer-to-peer protocol (7th layer)
Peer-to-peer protocol (6th layer)
Peer-to-peer protocol (5th layer)
Peer-to-peer protocol (4th layer)
7 Application
7-6 interface
6 Presentation
6-5 interface
5 Session
5-4 interface
4 Transport
4-3 interface
3 Network
3-2 interface
2 Data link
2-1 interface
1 Physical
3rd
2nd
1st
Physical communication
H7 D7
H6 D6
H5 D5
H4 D4
H3 D3
H2 D2 T2
010 0101010101011010100000010000
Transmission medium

OSI in Action

- A message begins at the top application layer and moves down the OSI layers to the bottom physical layer.
- As the message descends, each successive OSI model layer adds a header to it.
- A header is layer-specific information that basically explains what functions the layer carried out.
- Conversely, at the receiving end, headers are striped from the message as it travels up the corresponding layers.



Summary:


Application
To allow access to network resources
To translate, encrypt, and compress data
Presentation
Session
To establish, manage, and terminate sessions
To provide reliable process-to-process message delivery and error recovery
Transport
Network
To move packets from source to destination; to provide internetworking
To organize bits into frames; to provide hop-to-hop delivery
Data link
Physical
To transmit bits over a medium; to provide mechanical and electrical specifications

Key Points to remember layer order

Please Don't Take Silly People Advice[1-7 Bottom to Top]

All People Seem To Need Domino Pizza[7 – 1 Top to Bottom]

Animated Video link

https://www.youtube.com/watch?v=nFnLPGk8WjA

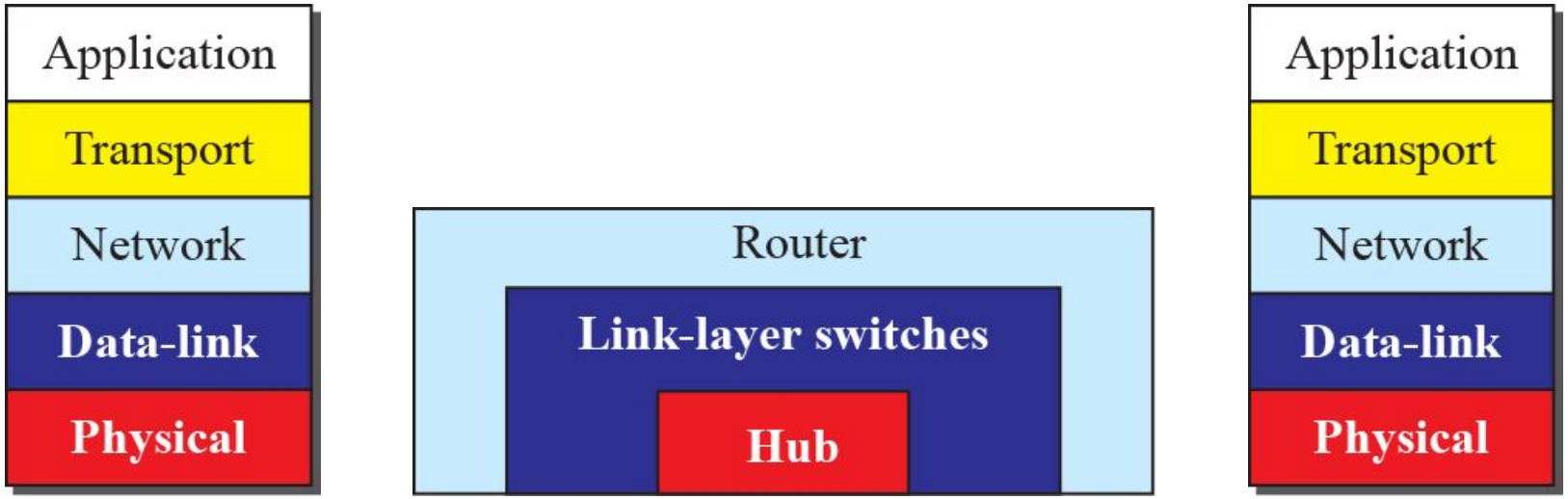
Review questions

1. Why it is necessary to have layering in a network?
2. What are the key benefits of layered network?
3. What do you mean by OSI?
4. What are the seven layers of ISO's OSI model?
5. Briefly write functionalities of different OSI layers?
6. How two adjacent layers communicate in a layered network? (or What do you mean by Service Access Point?)
7. What are the key functions of data link layer?
8. What do you mean by Protocol?

Networking Devices

Hosts and networks do not normally operate in isolation. We use connecting devices to connect hosts together to make a network or to connect networks together to make an internet. Connecting devices can operate in different layers of the Internet model. We discuss three kinds of connecting devices:

1. Hubs
2. link-layer switches
3. Routers
4. Repeater
5. Bridge



We use connecting devices:

- to connect hosts together to make a network
- to connect networks together to make an internet.

Today, connecting devices can operate in different layers of the Internet model:

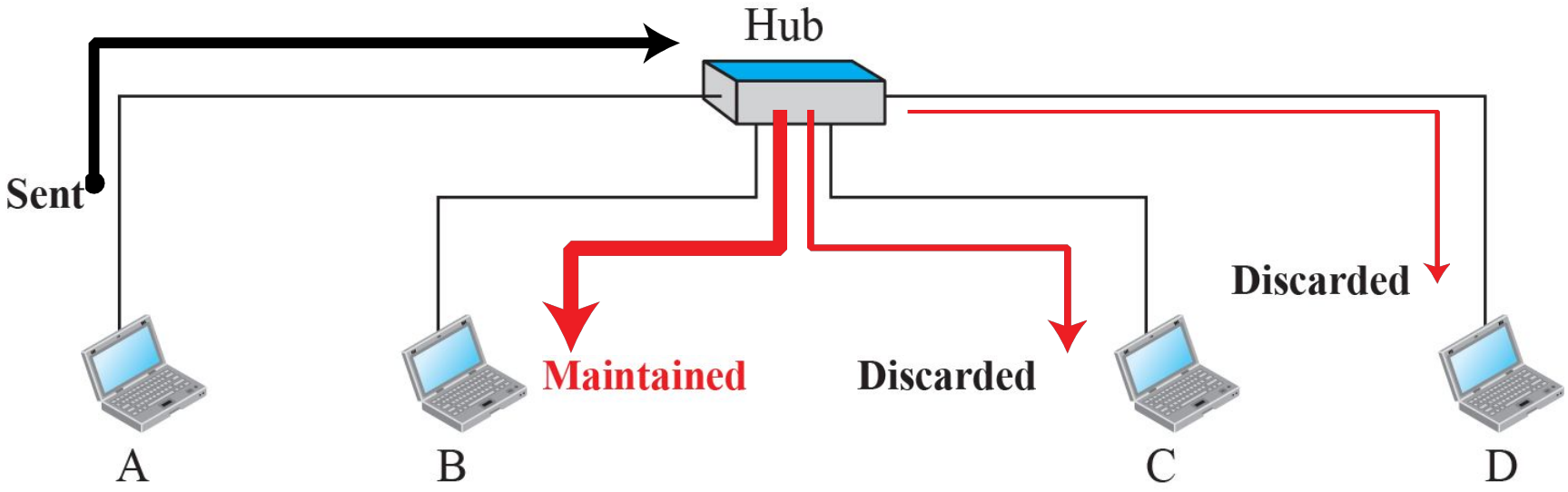
- Hubs: operate in the first layer of the Internet model.
- Link-layer switches: operate in the first two layers.
- Routers: operate in the first three layers.

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

The hub does not have a link-layer address and they do not check the link-layer address of the received frame. (no filtering capability)

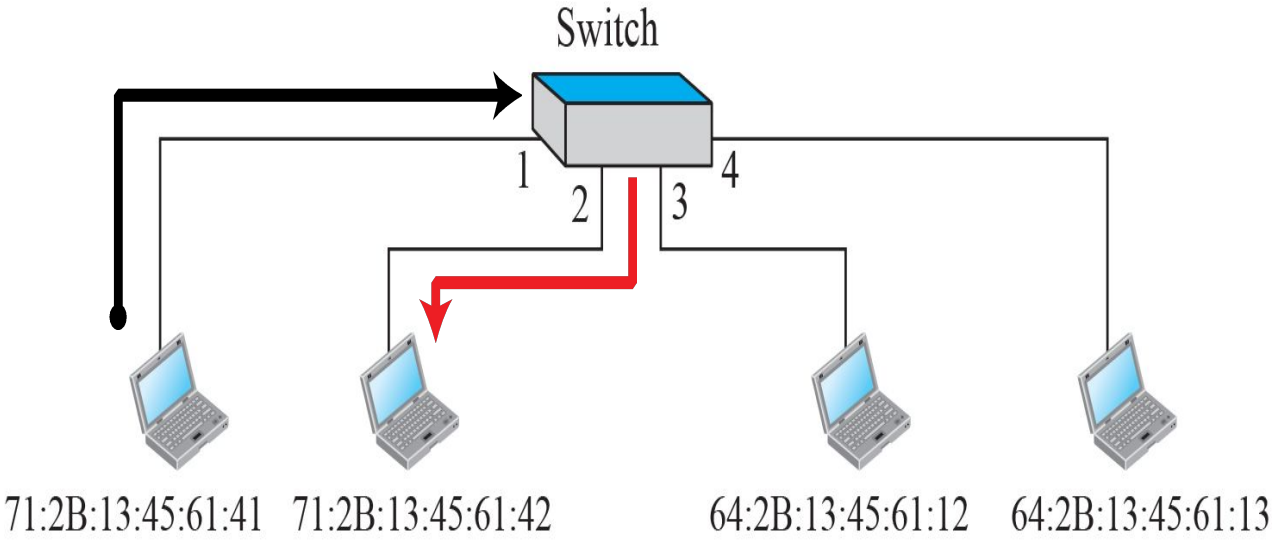
when a packet from station A to station B arrives at the hub, the signal is regenerated to remove any possible corrupting noise. Then the hub forwards the packet from all outgoing ports except the sender port.

(broadcast: all station gets it, but only station B keeps it)



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

A link-layer switch has a table used in filtering decisions.

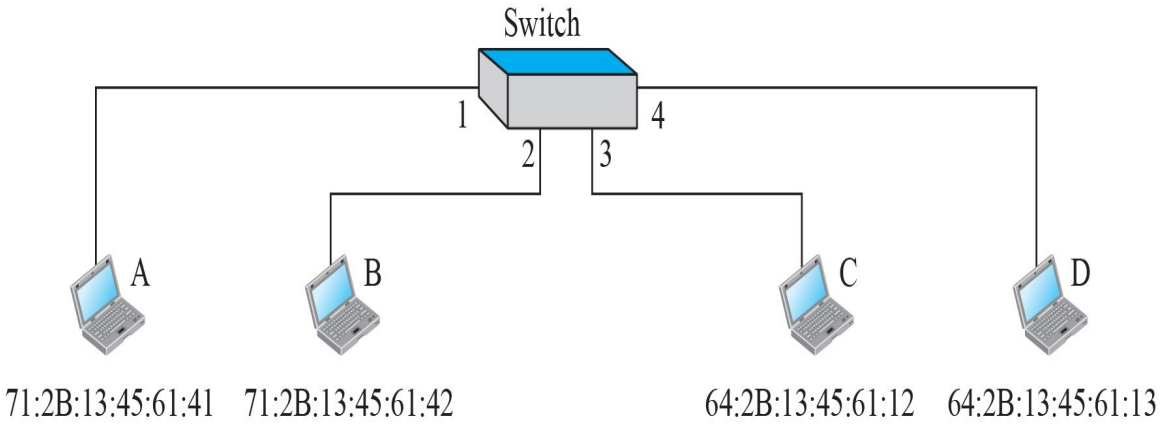


Switching table

Address	Port
71:2B:13:45:61:41	1
71:2B:13:45:61:42	2
64:2B:13:45:61:12	3
64:2B:13:45:61:13	4

We have a LAN with four stations that are connected to a link-layer switch. If a frame destined for station 71:2B:13:45:61:42 arrives at port 1, the link-layer switch consults its table to find the departing port. According to its table, frames for 71:2B:13:45:61:42 should be sent out only through port 2; therefore, there is no need for forwarding the frame through other ports.

The earliest switches had switching tables that were static: manually enter each table entry during switch setup
A better solution is a dynamic table that maps addresses to ports automatically. So we need a switch that gradually learns from the frames' movements (transparent switch).



When station A sends a frame to station D, The frame goes out from all three ports; the frame floods the network. However, by looking at the source address, the switch learns that station A must be connected to port 1. The switch adds this entry to its table. The table has its first entry now.

When station D sends a frame to station B, the switch has no entry for B, so it floods the network again. However, it adds one more entry to the table related to station D.

The learning process continues until the table has information about every port, but it may take a long time. For example, if a station does not send out a frame (a rare situation), the station will never have an entry in the table.

Address	Port

a. Original

Address	Port
71:2B:13:45:61:41	1

b. After A sends a frame to D

Address	Port
71:2B:13:45:61:41	1
64:2B:13:45:61:13	4

c. After D sends a frame to B

Address	Port
71:2B:13:45:61:41	1
64:2B:13:45:61:13	4
71:2B:13:45:61:42	2

d. After B sends a frame to A

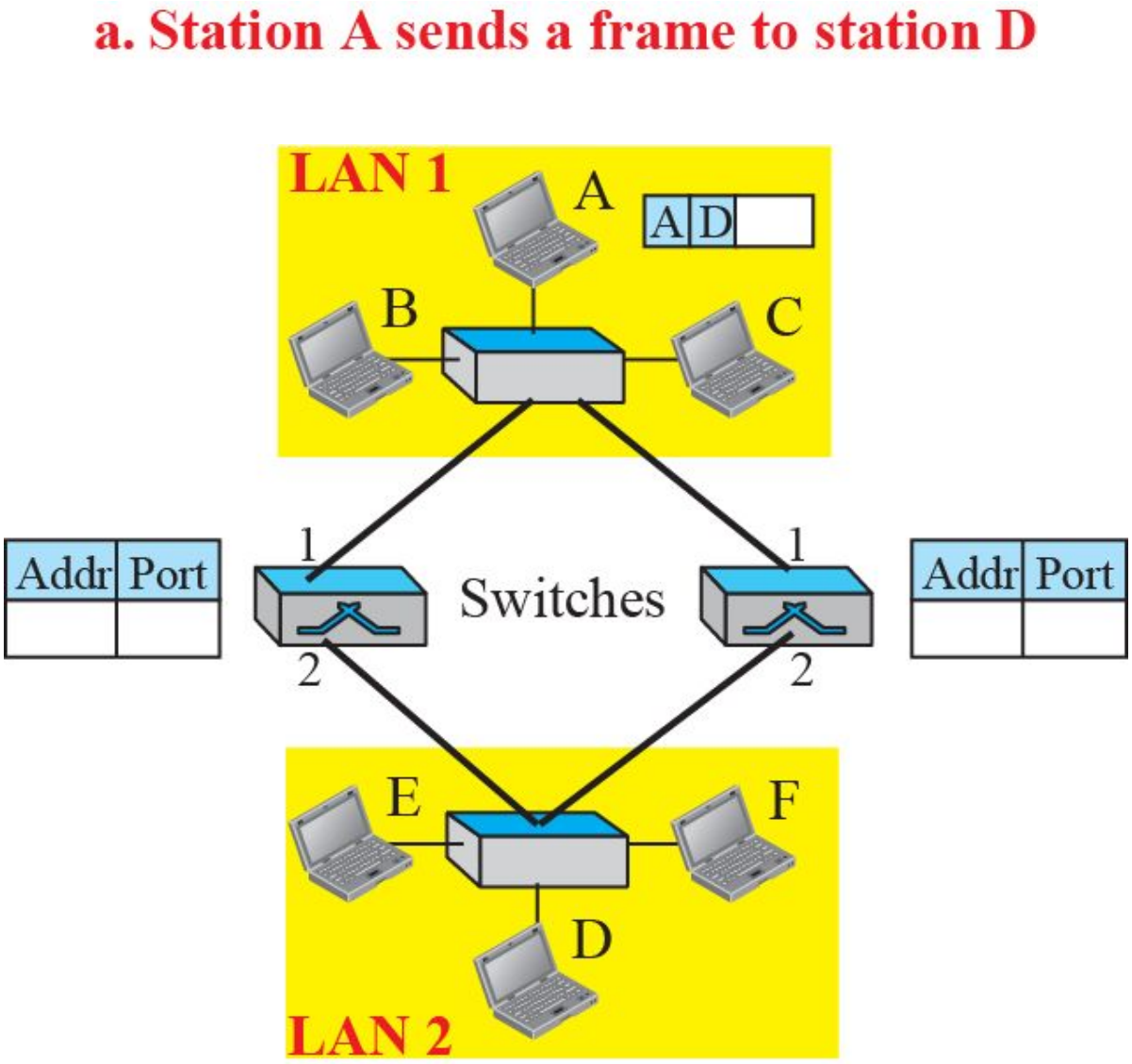
Address	Port
71:2B:13:45:61:41	1
64:2B:13:45:61:13	4
71:2B:13:45:61:42	2
64:2B:13:45:61:12	3

e. After C sends a frame to D

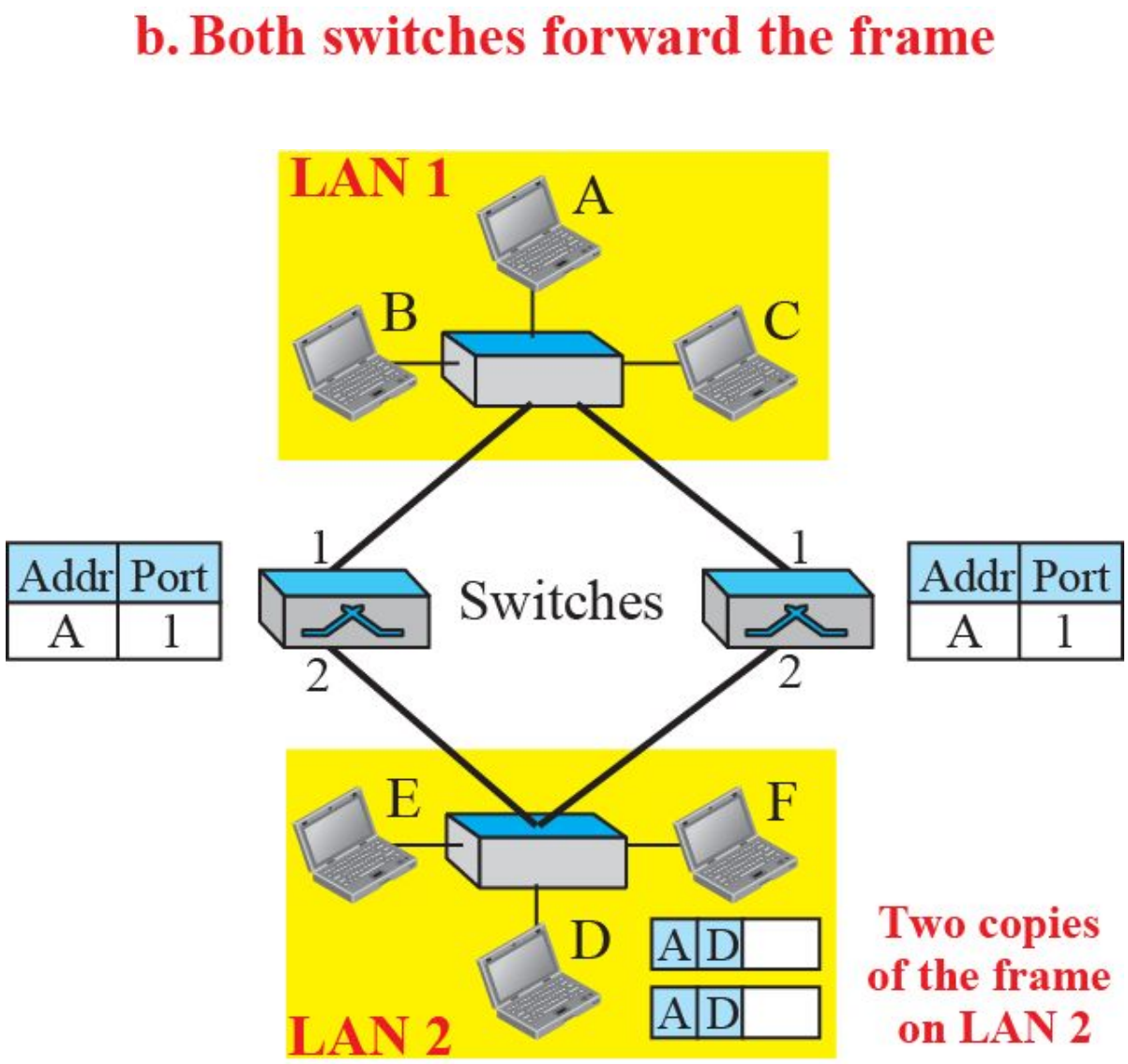
Gradual building of Table

Loop problem in a learning switch (Part a)

Station A sends a frame to station D. The tables of both switches are empty. Both forward the frame and update their tables based on the source address A.



Loop problem in a learning switch (Part b)



Now there are two copies of the frame on LAN 2:

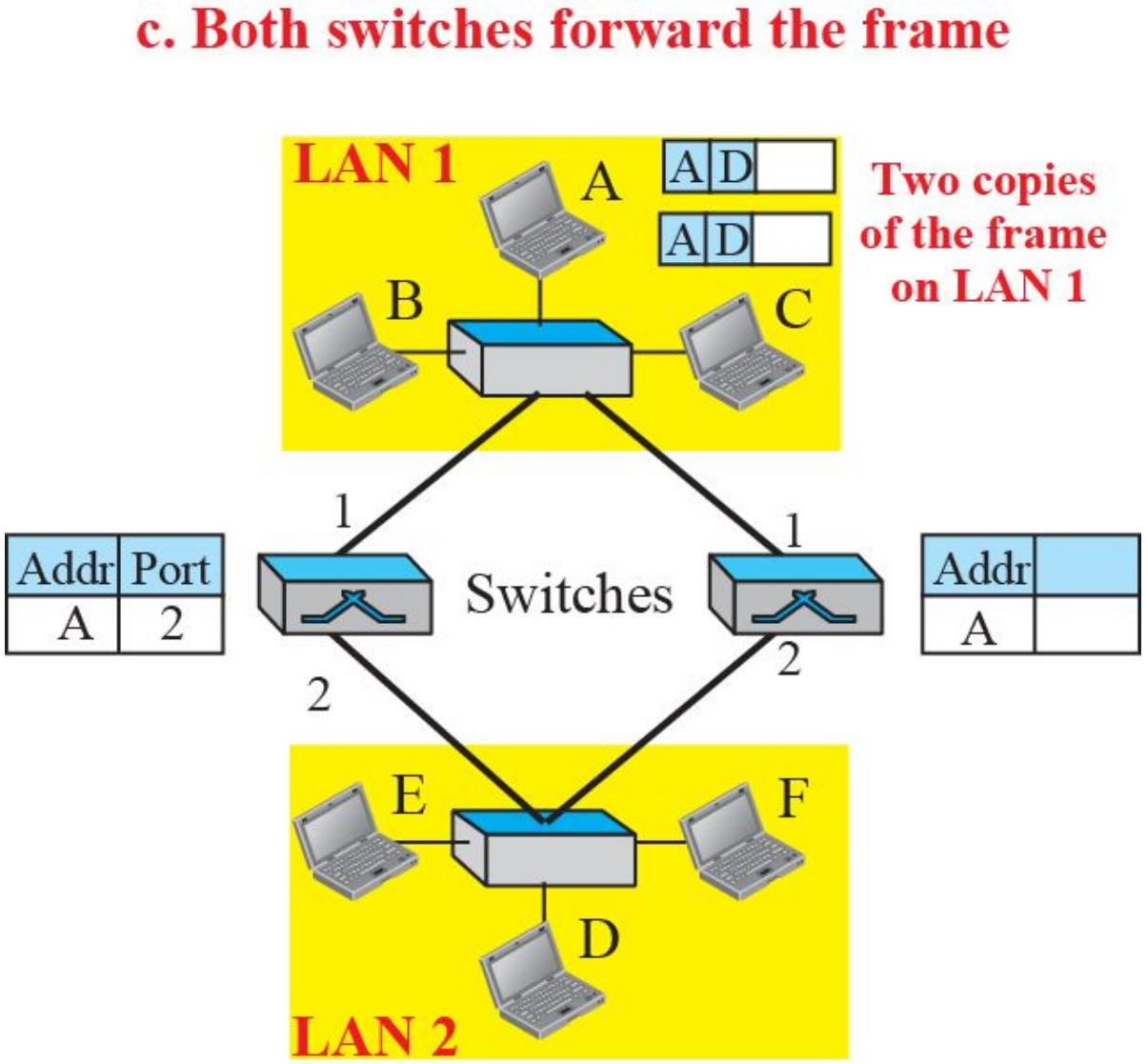
1. The copy sent out by the left switch is received by the right switch, which does not have any information about the destination address D; it forwards the frame.

2. The copy sent out by the right switch is received by the left switch and is sent out for lack of information about D.

The tables of both switches are updated, but still there is no information for destination D.

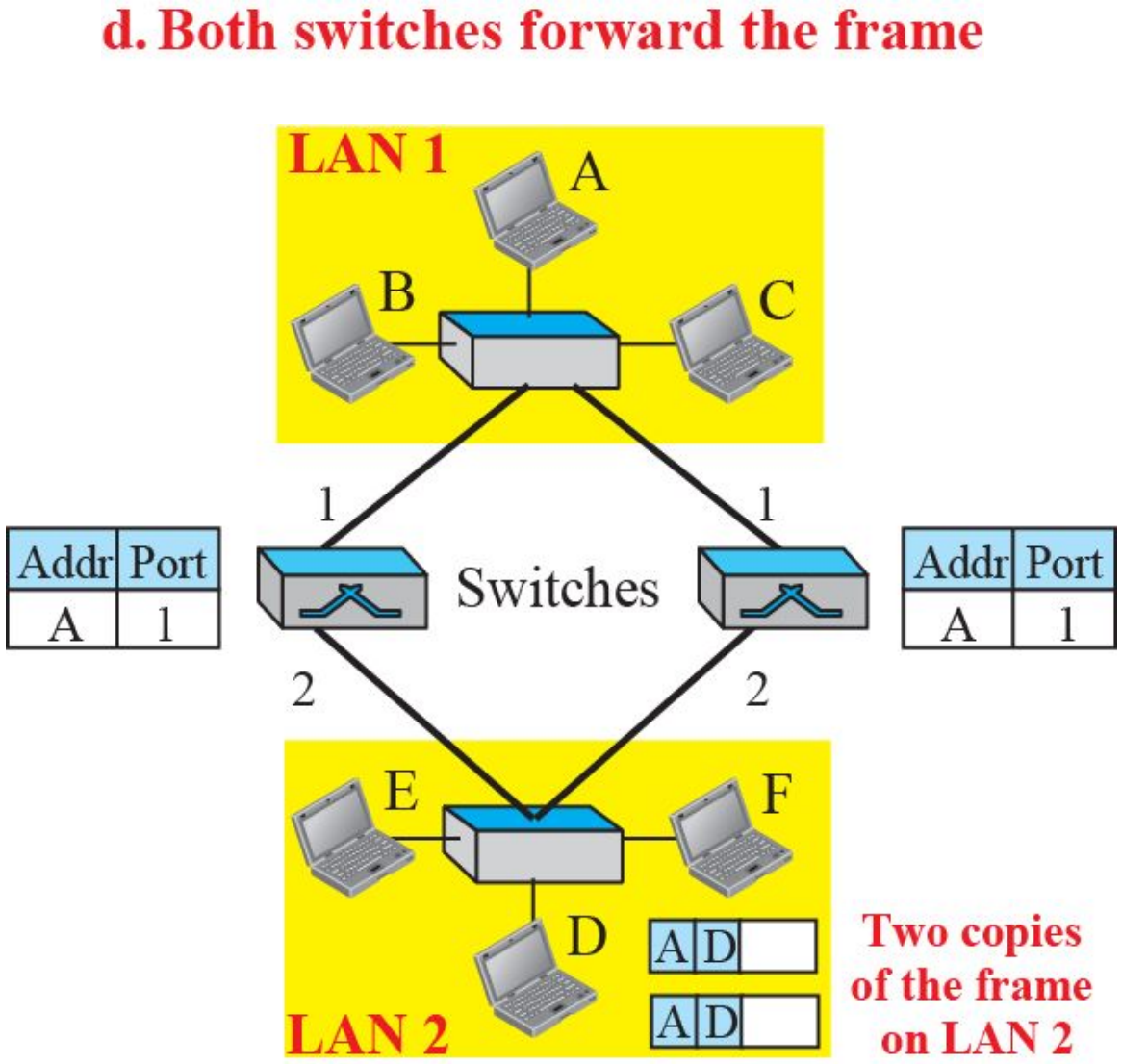
Loop problem in a learning switch (Part c)

Now there are two copies of the frame on LAN 1. Step 2 is repeated, and both copies are sent to LAN2.

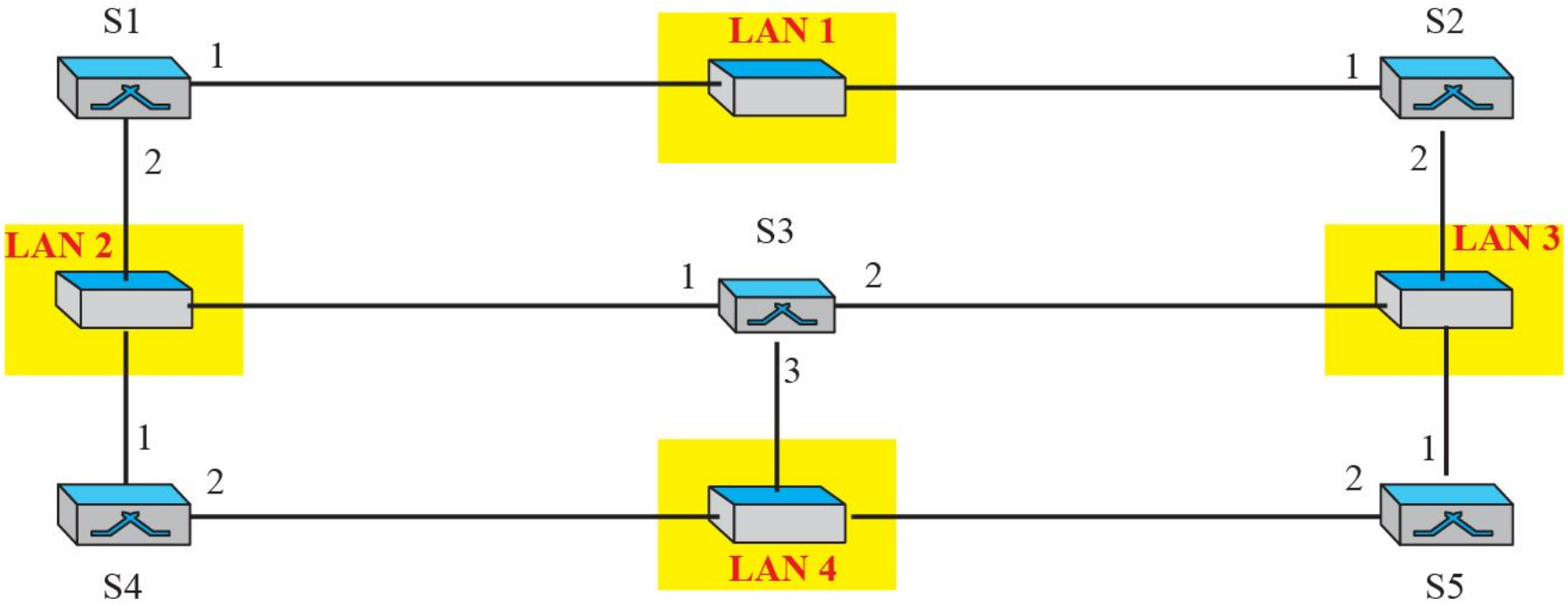


Loop problem in a learning switch (part d)

The process continues on and on.



A system of connected LANs and its graph (Part a)



a. Actual system

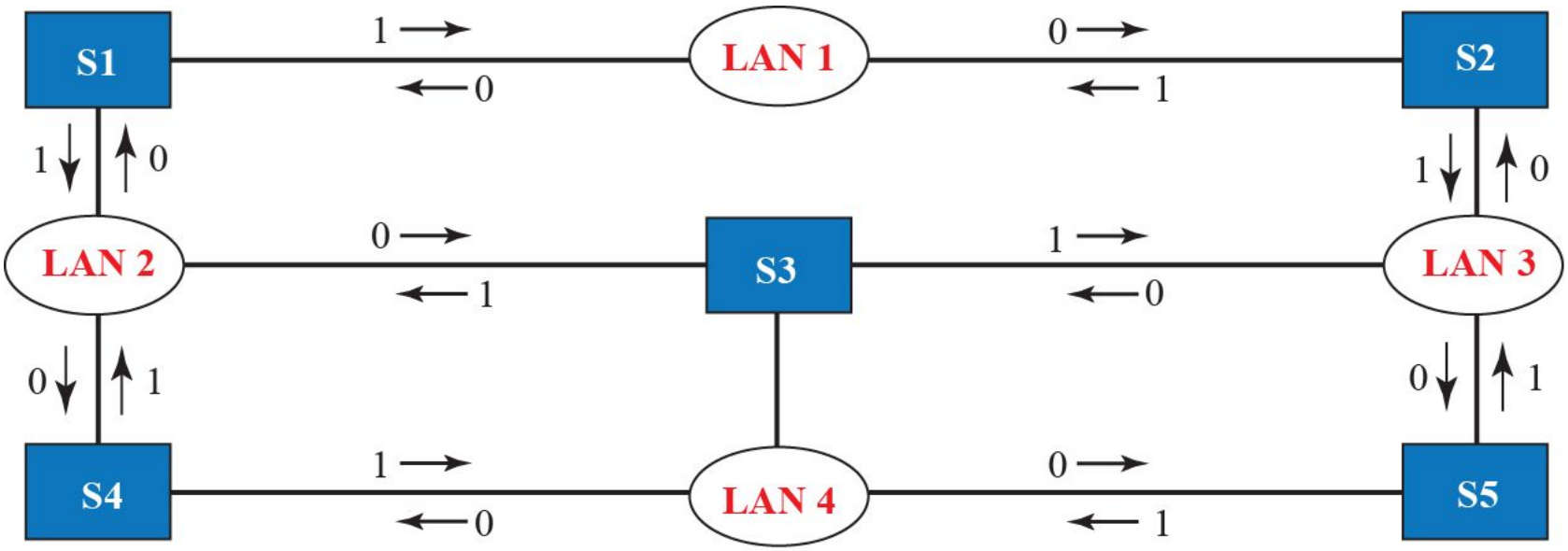
To solve the looping problem, the IEEE specification requires that switches use the Spanning Tree Algorithm.

A spanning tree is a graph in which there is no loop.

this means creating a topology in which each LAN can be reached from any other LAN through one path only (no loop).

Figure 17.6 shows a system with four LANs and five switches

A system of connected LANs and its graph (Part b)

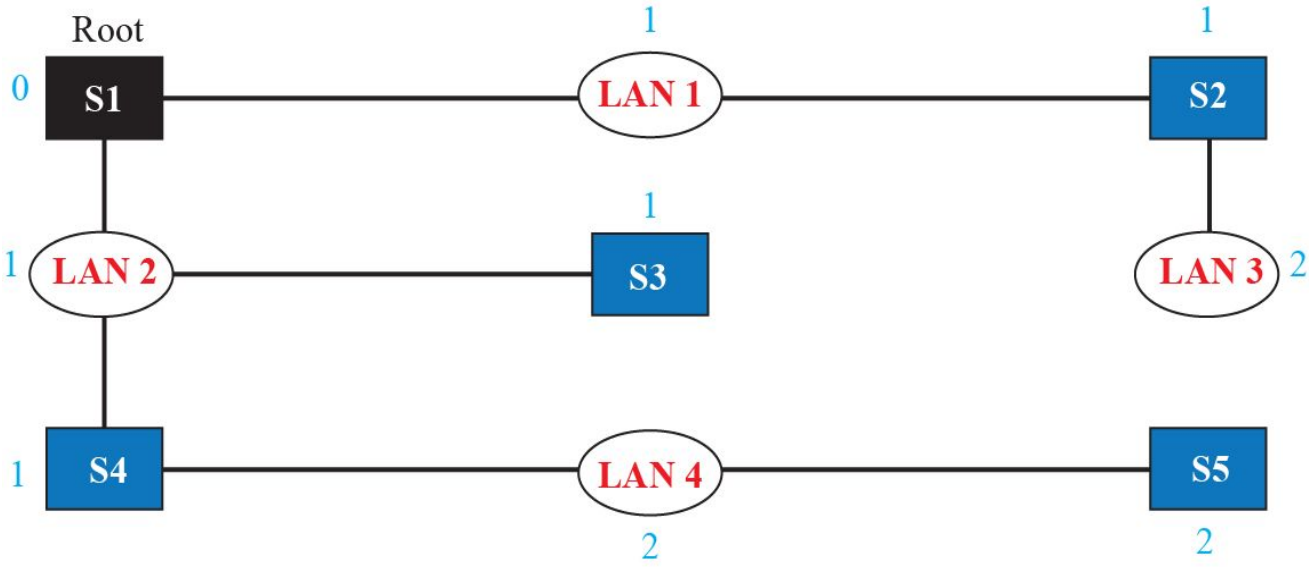


b. Graph representation with cost assigned to each arc

The hop count is normally 1 from a switch to the LAN and 0 in the reverse direction.

Finding the shortest path and the spanning tree for a switch.

In the spanning tree system, there is only one path from any LAN to any other LAN (No loops)



The process for finding the spanning tree:

1. Every switch has a built-in ID (normally the serial number, which is unique). Each switch broadcasts this ID so that all switches know which one has the smallest ID. The switch with the smallest ID is selected as the root switch (root of the tree). We assume that switch S1 has the smallest ID (root switch).

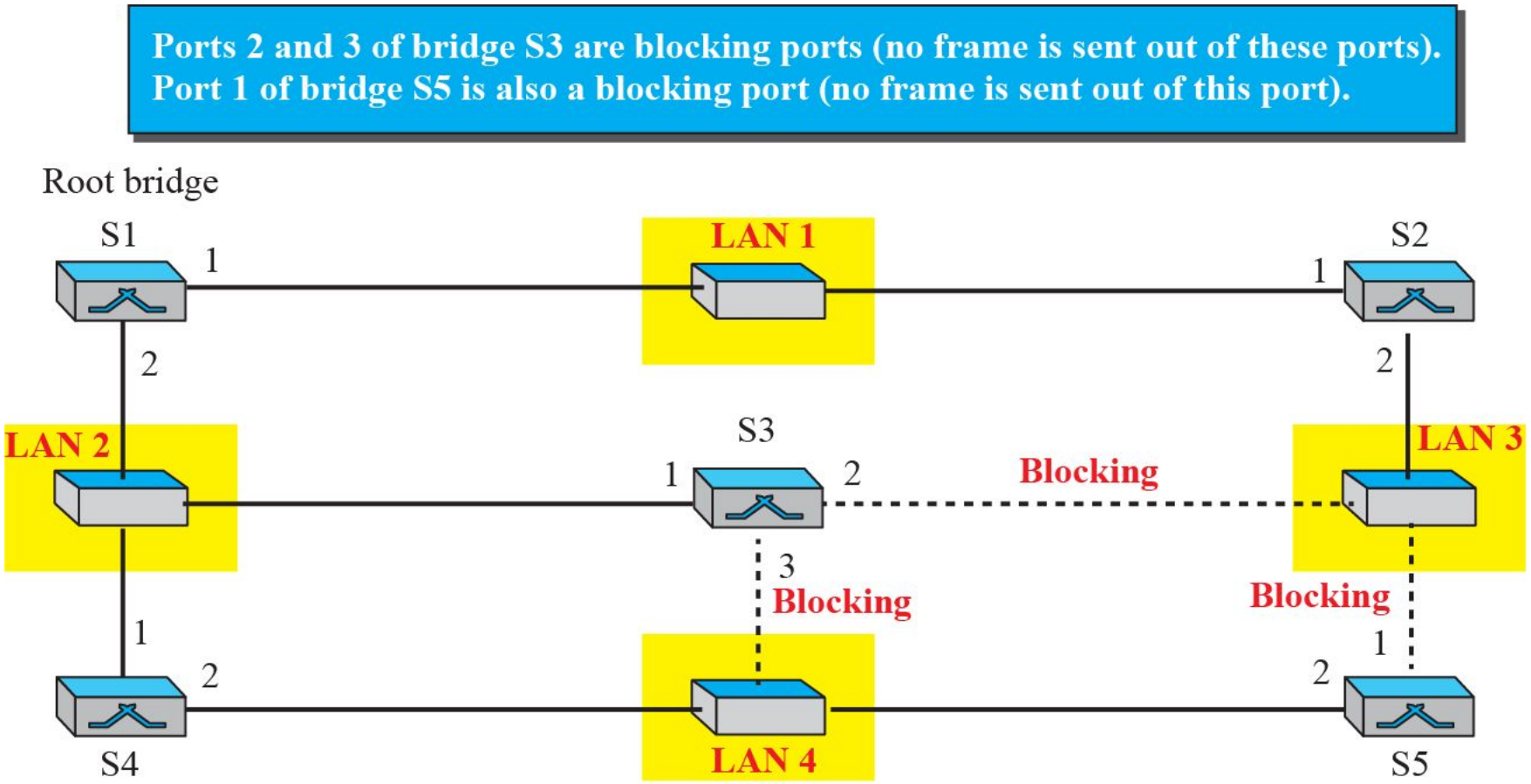
2. The algorithm tries to find the shortest path (a path with the shortest cost) from the root switch to every other switch or LAN. The shortest path can be found by examining the total cost from the root switch to the destination.

3. The combination of the shortest paths creates the shortest tree.

4. We mark the ports that are part of spanning tree, and the ports that are not part of it, the blocking ports, which block the frames received by the switch.

Forwarding and blocking ports after using spanning tree algorithm

The figure shows the logical systems of LANs with forwarding ports (solid lines) and blocking ports (broken lines).



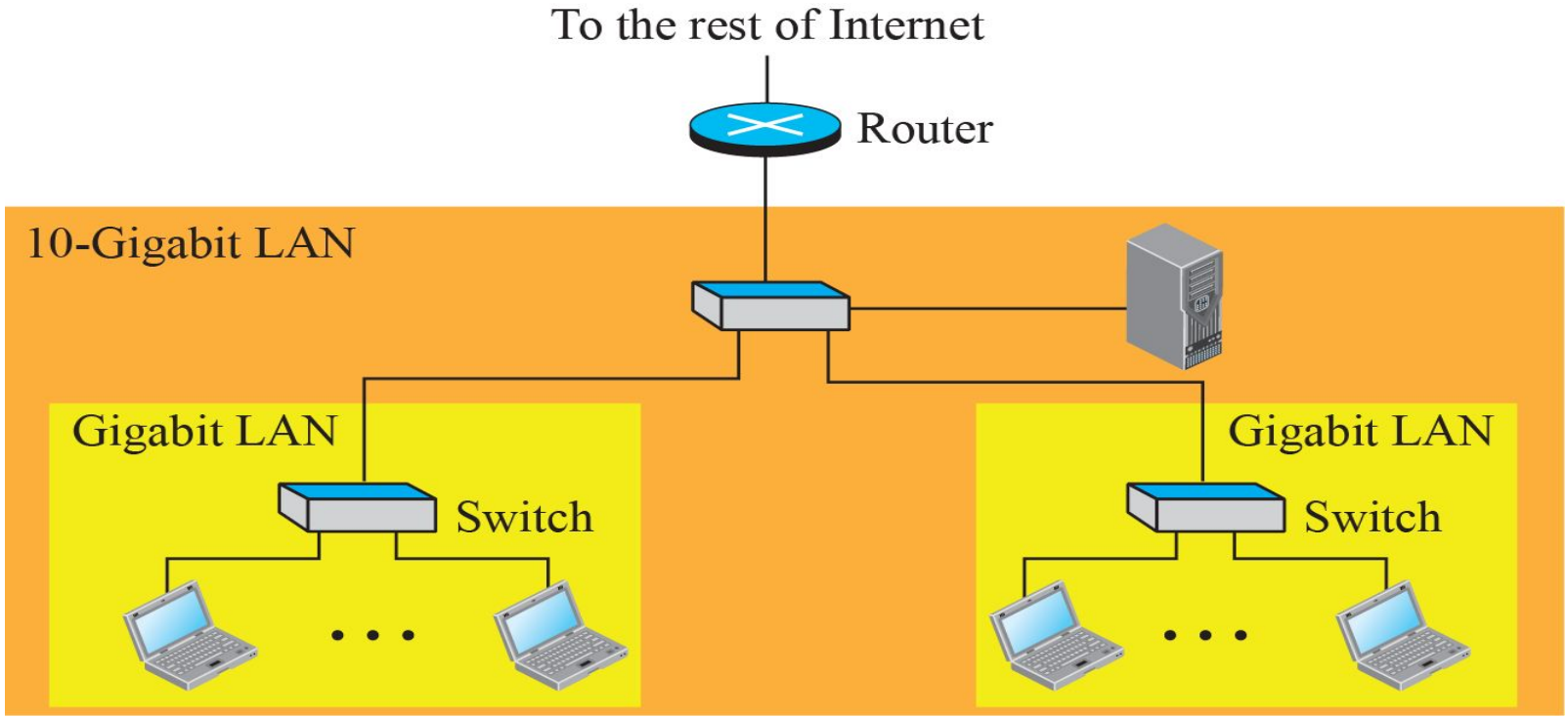
Routers

We will discuss routers later but in this section, we mention routers to compare them with a two-layer switch and a hub.

A router is a three-layer device; it operates in the physical, data-link, and network layers. There are three major differences between a router and a repeater or a switch:

1. ***A router has a physical and logical (IP) address for each of its interfaces.***
2. ***A router acts only on those packets in which the link-layer destination address matches the address of the interface at which the packet arrives.***
3. ***A router changes the source and distention link-layer addresses of the packet (source and destination MAC addresses) when it forwards the packet.***

Routing example



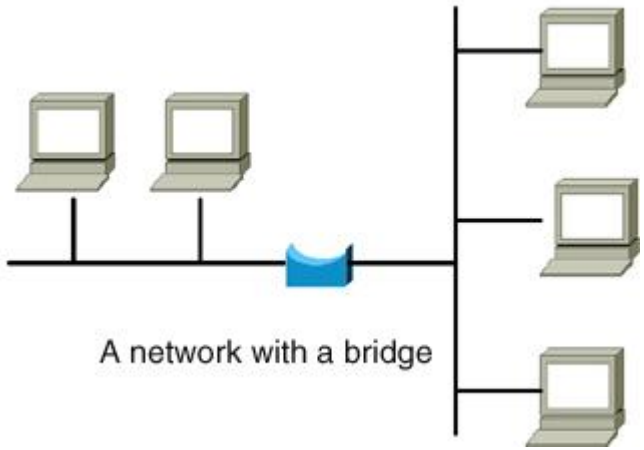
Assume an organization has two separate buildings with a Gigabit Ethernet LAN installed in each building. The organization uses switches in each LAN. The two LANs can be connected to form a larger LAN using 10 Gigabit Ethernet technology that speeds up the connection to the Ethernet and the connection to the organization server. A router then can connect the whole system to the Internet.

Repeater

- A repeater receives a signal, regenerates it, and passes it on.
- It can regenerate and retime network signals at the bit level to allow them to travel a longer distance on the media.
- It operates at Physical Layer of OSI
- The Four Repeater Rule for 10-Mbps Ethernet should be used as a standard when extending LAN segments.
- This rule states that no more than four repeaters can be used between hosts on a LAN.
- This rule is used to limit latency added to frame travel by each repeater.

Bridge

- Bridges are used to logically separate network segments within the same network.
- They operate at the OSI data link layer (Layer 2) and are independent of higher-layer protocols.
- The function of the bridge is to make intelligent decisions about whether or not to pass signals on to the next segment of a network.
- When a bridge receives a frame on the network, the destination MAC address is looked up in the bridge table to determine whether to filter, flood, or copy the frame onto another segment
- Broadcast Packets are forwarded



A network with a bridge

SOFTWARE DEFINED NETWORK

- LIMITATIONS OF EXISTING NETWORKS

- Difficult to perform real world experiments on large scale production networks.

- Research stagnation-huge costly equipment to be procured and networks to be setup by each team for research •

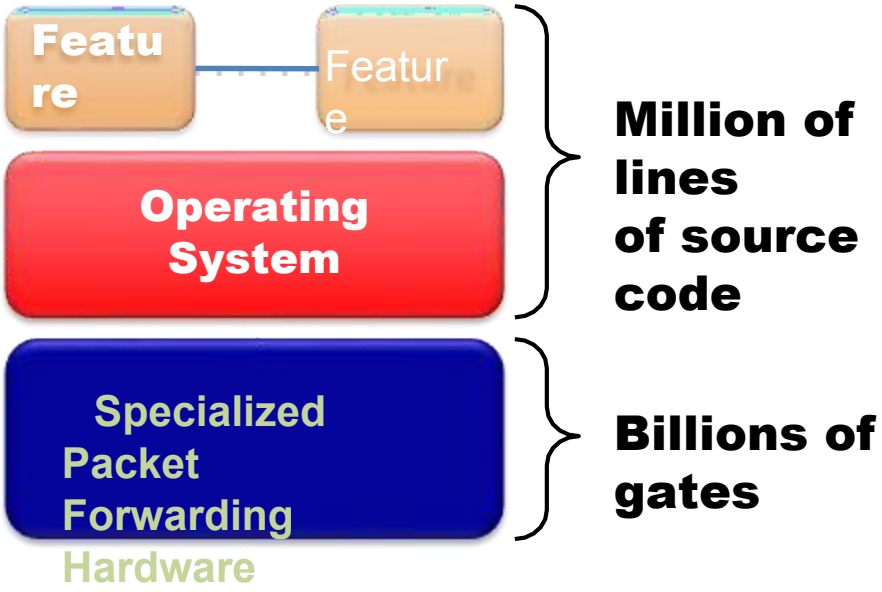
- Networks have remained the same for many years

- Rate of innovation in networks is slower as protocols are defined in isolation-lack of high level abstraction.

- Closed systems

- Hard to collaborate meaningfully due to lack of standard open interfaces.
- Vendors starting to open-up but not meaningfully.
- Innovation is limited to vendor/vendor partners •
- Huge barriers for new ideas in networking.

Limitations of Current Networks

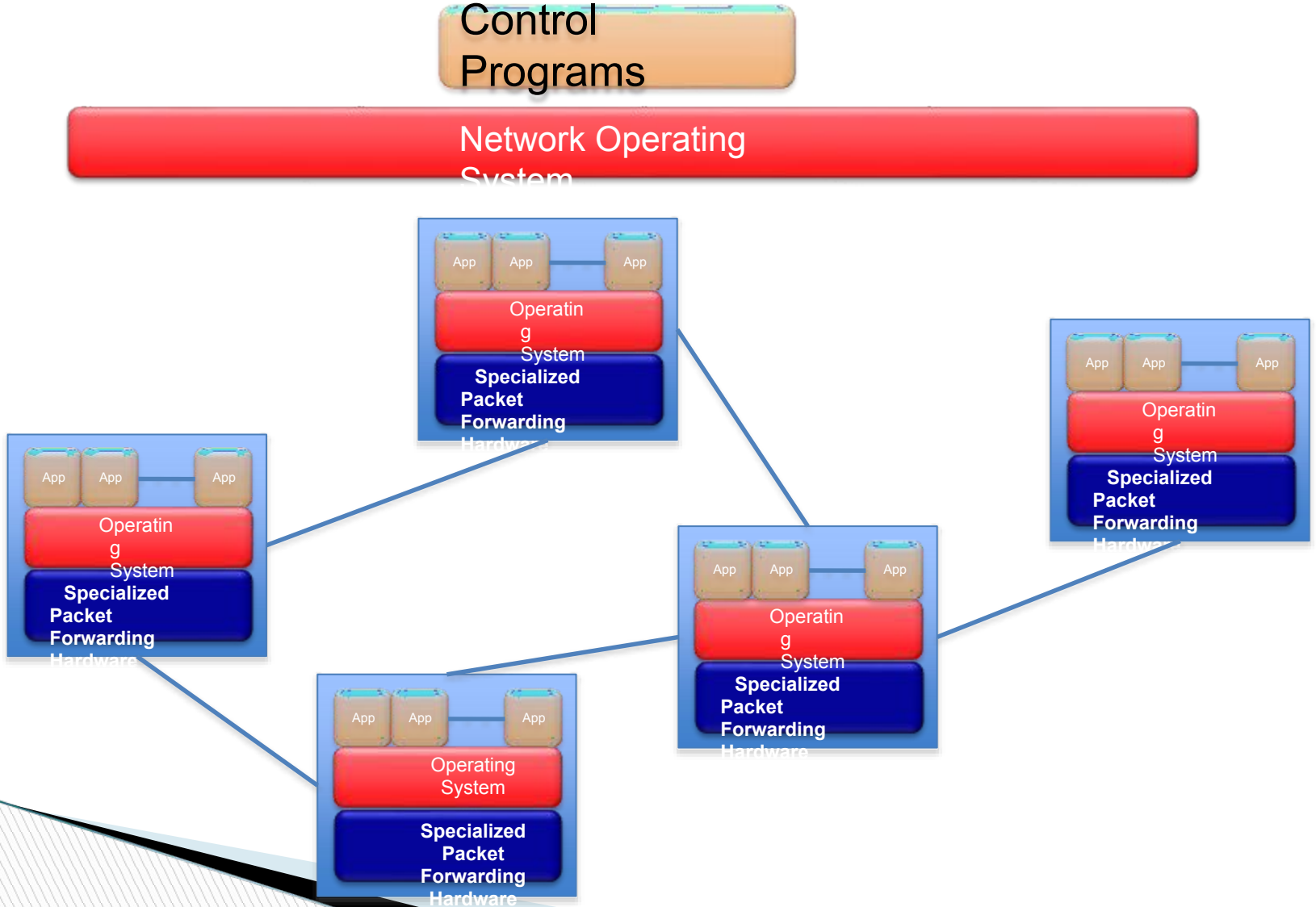


Many complex functions baked into infrastructure

OSPF, BGP, multicast, differentiated services, Traffic Engineering, NAT, firewalls, …

Cannot dynamically change according to network conditions

Idea: An OS for Networks



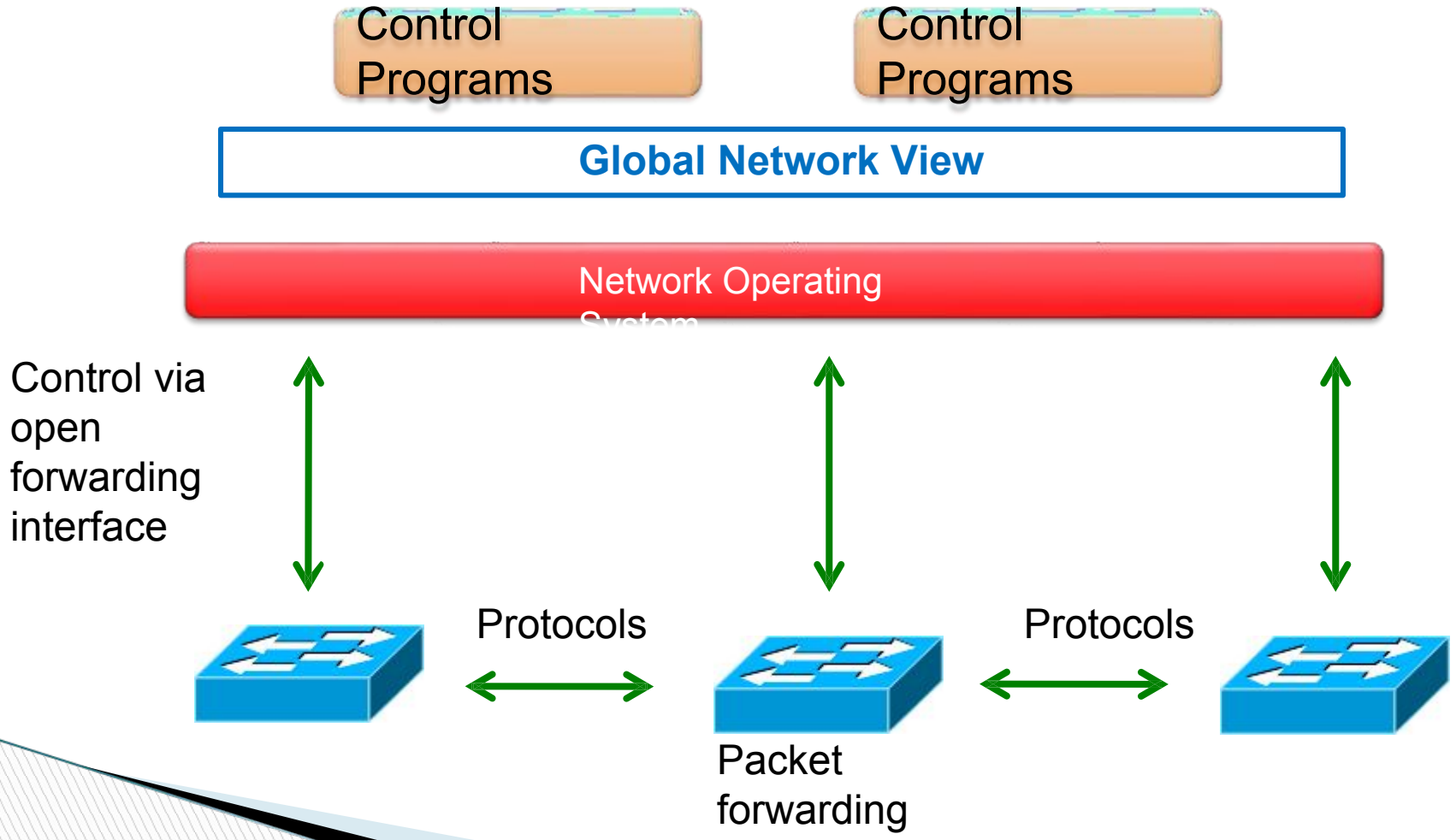
SOFTWARE DEFINED NETWORK

- **Data Plane**: processing and delivery of packets Based on state in routers and endpoints E.g., IP, TCP, Ethernet, etc.

- **Control Plane**: establishing the state in routers Determines how and where packets are forwarded Routing, traffic engineering, firewall state, …

- Separate control plane and data plane entities
- Have programmable data planes—maintain, control and program data plane from a central entity i.e. control plane software called controller.
- An architecture to control not just a networking device but an entire network.

Software-Defined Networking (SDN)

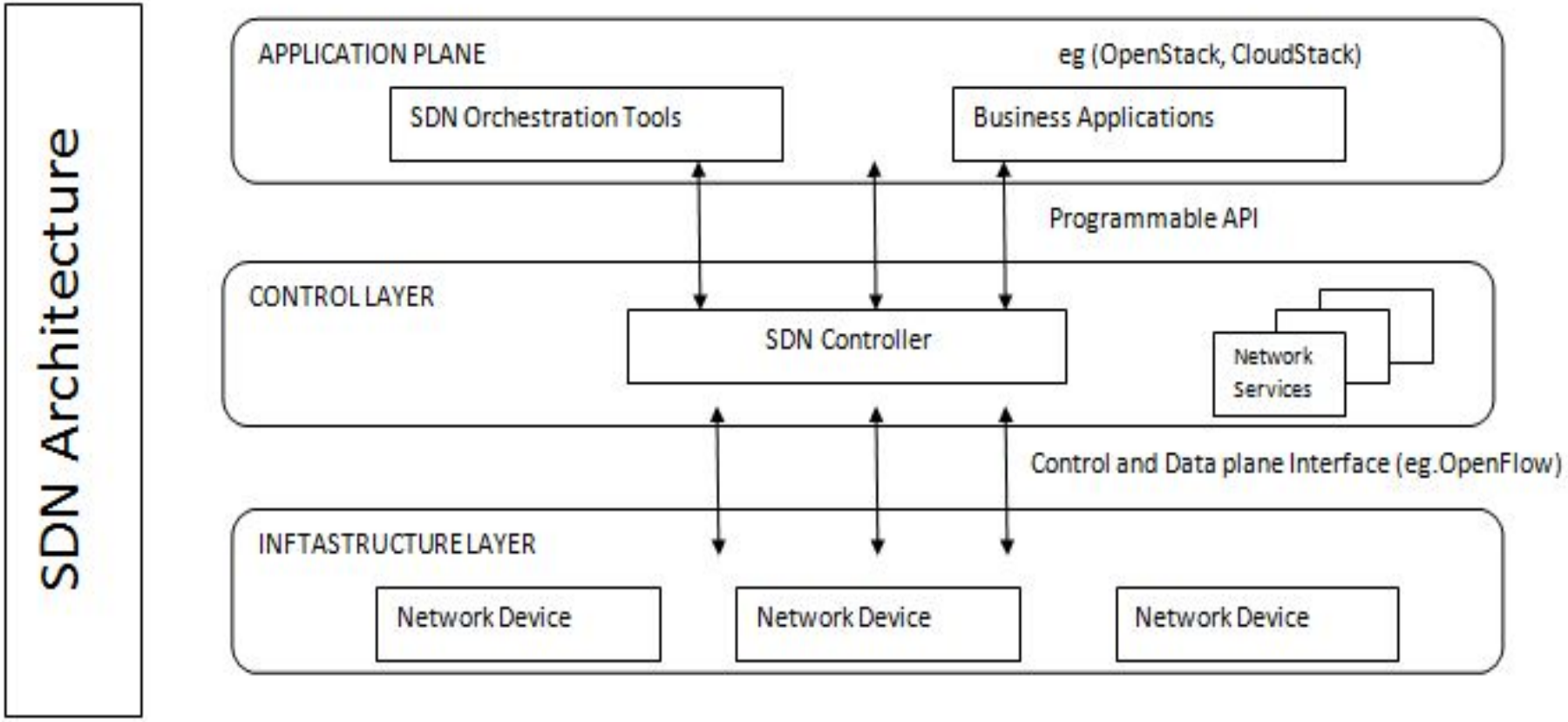


NEED FOR SDN

- Facilitate innovation in network.
- Layered architecture with standard Open interfaces.
- Experiment and research using non-bulky, non-expensive equipment.
- More accessibility since software can be easily developed by more vendors.
- More flexibility with programmability.
- Ease of customization and integration with other software applications
- Program a network vs. configure a network

ARCHITECTURE OF SDN

In the SDN architecture, the control and data planes are decoupled, network intelligence and state centralized, and the underlying network infrastructure is abstracted from the applications.



SDN LAYERS

- Infrastructure layer: it is the foundation layer consists of both physical and virtual network devices such as switches and routers. All the network devices will implement OpenFlow protocol to implement traffic forwarding rules.

- Control layer: This layer consists of a centralized control plane that is decoupled from the physical infrastructure to provide centralized global view to entire network. The layer will use OpenFlow protocol to communicate with below layer i.e. infrastructure layer.

- Application layer: it consists of network services, application and orchestration tools that are used to interact with control layer. It provide an open interface to communicate with other layers in the architecture.

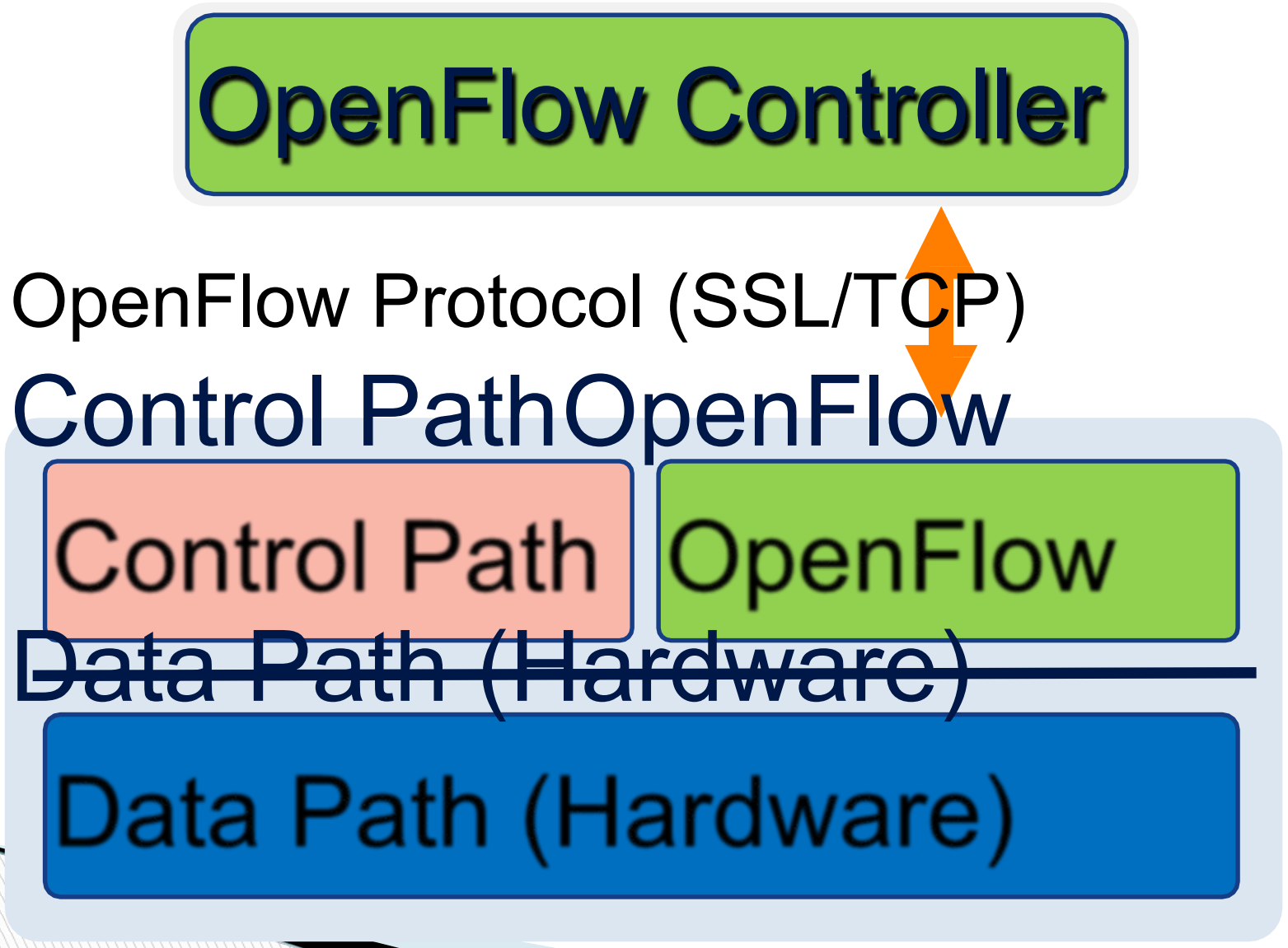
OPENFLOW PROTOCOL

- OPENFLOW is an open API that provides a standard interface for programming the data plane switches. It is a protocol for remotely controlling the forwarding table of a switch or router and is one element of SDN.

- It is implemented on Ethernet switches to allow the forwarding plane i.e. data plane to be managed by a controller present on control plain in SDN architecture. OpenFlow based controllers will discover and maintain an inventory of all the links in the network and then will create and store all possible paths in entire network.

- OpenFlow protocol can instruct switches and routers to direct the traffic by providing software-based access to flow tables that can be used to quickly change the network layout and traffic flows as per users requirements.

OpenFlow


OpenFlow Controller
OpenFlow Protocol (SSL/TCP)
Control PathOpenFlow
Control Path
OpenFlow
Data Path (Hardware)
Data Path (Hardware)

OPENFLOW SWITCH AND CONTROLLER

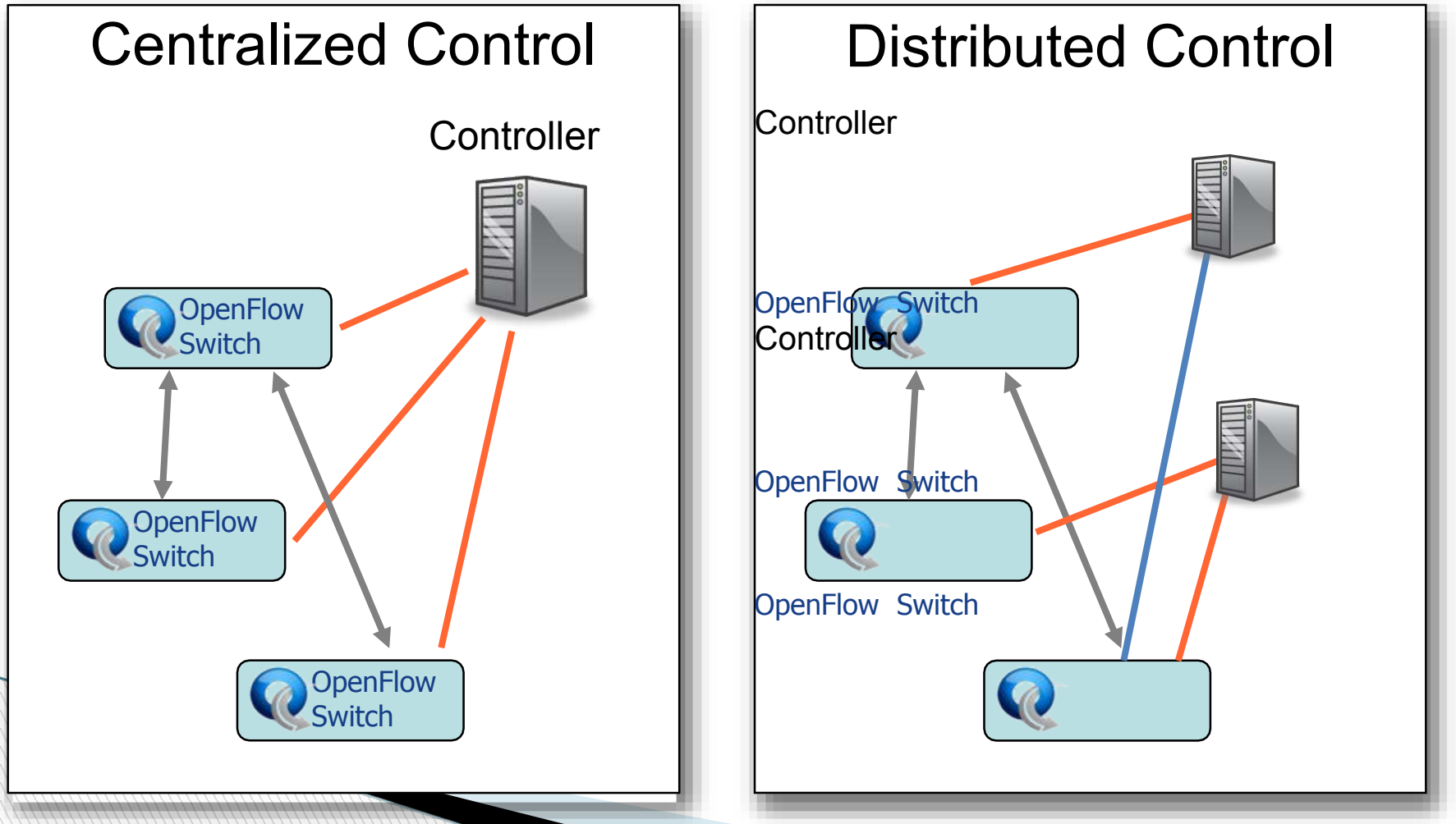
- An OpenFlow Switch contain one or more flow tables that implement packet lookups and forwarding, and an OpenFlow channel to link to an external controller .The switch interconnects with the controller and the controller directs the switch using the OpenFlow protocol.

- The controller can delete, add or update flow entries in flow tables existing in the switch, both reactively i.e. in response to packets or proactively, using the OpenFlow protocol.

- Controller make this decision based on policies set by administrator or depending on the conditions of the network and the decision it makes is forwarded to flow table entries of all the switches in the network.

Centralized/Distributed Control

- **“Onix: A Distributed Control Platform for Large-scale Production Networks”**



CURRENT STATUS of SDN

- Google built hardware and software based on the OpenFlow protocol

- VMware purchased Nicira for $1.26 billion in 2012

- IBM, HP, NEC, Cisco and Juniper also are offering SDNs that may incorporate OpenFlow, but also have other elements that are specific to that vendor and their gear.

CONCLUSIONS and FUTURE SCOPE

- In future, networking will rely more on software to pick up the pace the innovations in networks.

- SDN can transform today's static networks into more flexible, programmable platforms to provide scalability to support large data centers. It will also provide virtualization that is needed to support automated, dynamic and secure cloud environment.

- Mostly implementations of newly proposed systems, frameworks, or applications

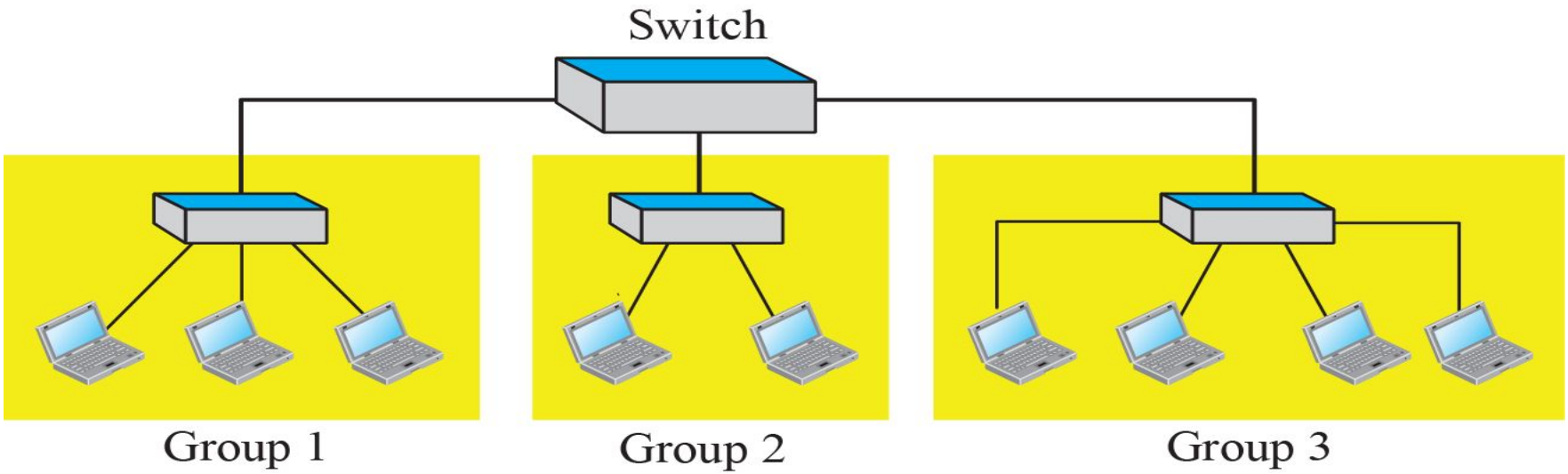
VIRTUAL LANS

A station is considered part of a LAN if it physically belongs to that LAN. The criterion of membership is geographic.

What happens if we need a virtual connection between two stations belonging to two different physical LANs?

We can roughly define a virtual local area network (VLAN) as a local area network configured by software, not by physical wiring.

A switch connecting three LANs



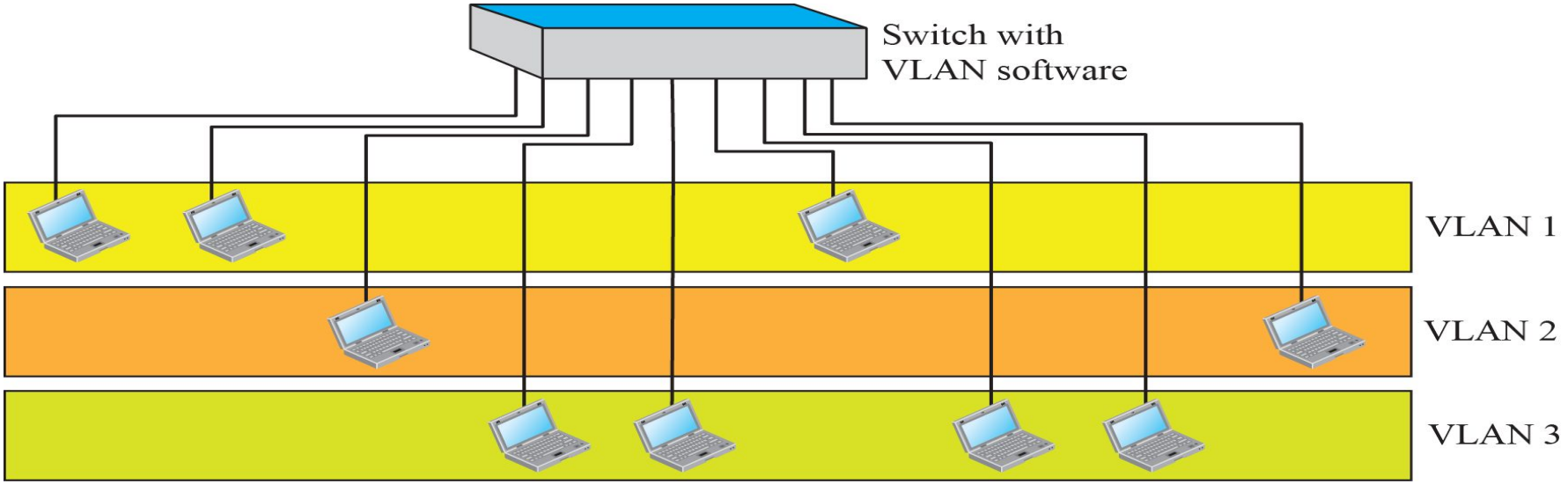
The figure shows a switched LAN in an engineering firm in which nine stations are grouped into three LANs that are connected by a switch.

what would happen if the administrators needed to move engineers from their groups?

- ***The LAN configuration would need to be changed.***
- ***The network technician must rewire.***

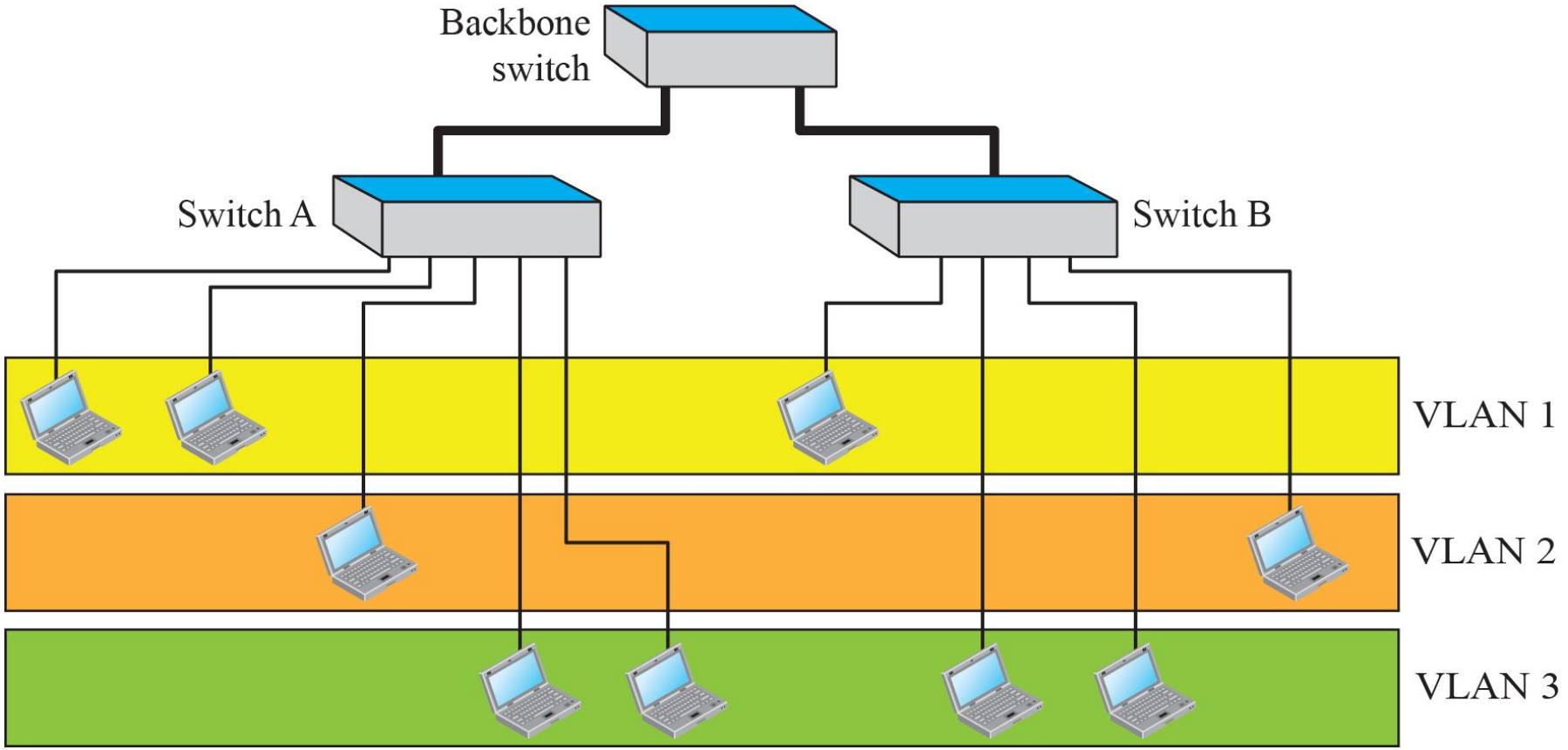
So...In a switched LAN, changes in the work group mean physical changes in the network configuration.

A switch using VLAN software



The whole idea of VLAN technology is to divide a LAN into logical segments, instead of physical.
A LAN can be divided into several logical LANs, called VLANs.
Each VLAN is a work group in the organization.
The group membership in VLANs is defined by software, not hardware.
Any station can be logically moved to another VLAN.
All members belonging to a VLAN can receive broadcast messages sent to that particular VLAN
(if a station moves from VLAN 1 to VLAN 2, it only receives broadcast messages sent to VLAN 2, and there is no need to change the physical configuration.)

Two switches in a backbone using VLAN software



VLAN technology even allows the grouping of stations connected to different switches in a VLAN.

Membership

What characteristic can be used to group stations in a VLAN? Vendors use different characteristics such as:

Interface Numbers: use switch interface numbers as a membership characteristic.

For example, the administrator can define that stations connecting to ports 1, 2, 3, and 7 belong to VLAN 1, stations connecting to ports 4, 10, and 12 belong to VLAN 2, and so on.

MAC Addresses: use the 48-bit MAC address as a membership characteristic.

For example, the administrator can stipulate that stations having MAC addresses E2:13:42:A1:23:34 and F2:A1:23:BC:D3:41 belong to VLAN 1.

IP Addresses: use the 32-bit IP address as a membership characteristic.

For example, the administrator can stipulate that stations having IP addresses 181.34.23.67, 181.34.23.72, 181.34.23.98, and 181.34.23.112 belong to VLAN 1.

Multicast IP Addresses: use the multicast IP address as a membership characteristic. Multicasting at the IP layer is now translated to multicasting at the data- link layer.(ch21)

Combination: allows all these characteristics to be combined. The administrator can choose one or more characteristics when installing the software. In addition, the software can be reconfigured to change the settings.

Configuration

How are the stations grouped into different VLANs? Stations are configured in one of three ways:

Manual Configuration: the network administrator uses the VLAN software to manually assign the stations into different VLANs at setup (logical not a physical configuration). The term manually here means that the administrator types the port numbers, the IP addresses, or other characteristics, using the VLAN software.

Automatic Configuration: the stations are automatically connected or disconnected from a VLAN using criteria defined by the administrator. For example, the administrator can define the project number as the criterion for being a member of a group. When a user changes projects, he or she automatically migrates to a new VLAN.

Semiautomatic Configuration: A semiautomatic configuration is somewhere between a manual configuration and an automatic configuration. Usually, the initializing is done manually, with migrations done automatically.

Communication between Switches

In a multi-switched backbone, each switch must know not only which station belongs to which VLAN, but also the membership of stations connected to other switches. For example, in Figure 17.12, switch A must know the membership status of stations connected to switch B, and switch B must know the same about switch A. Three methods have been devised for this purpose:

Table Maintenance: when a station sends a broadcast frame to its group members, the switch creates an entry in a table and records station membership. The switches send their tables to one another periodically for updating.

Frame Tagging: when a frame is traveling between switches, an extra header is added to the MAC frame to define the destination VLAN.

Time-Division Multiplexing (TDM): The connection (trunk) between switches is divided into time-shared channels. For example, if the total number of VLANs in a backbone is five, each trunk is divided into five channels. The traffic destined for VLAN 1 travels in channel 1, and so on.

Wireless Sensor Networks

- Introduction
- Wireless Sensor Networks Applications
- Factors Influencing Sensor Network Design
- Sensor Node Components
- Sensor Networks Communication Architecture
- Sensor Network Protocols
- Sensor Networks Operating Systems
- Sensor Networks Simulators
- Conclusion

Introduction

- sensor
  - A transducer
  - converts physical phenomenon e.g. heat, light, motion, vibration, and sound into electrical signals
- sensor node
  - basic unit in sensor network
  - contains on-board sensors, processor, memory, transceiver, and power supply
- sensor network
  - consists of a large number of sensor nodes
  - nodes deployed either inside or very close to the sensed phenomenon

Wireless Sensor Networks Applications

Military Applications

- Monitoring friendly forces, equipment, and ammunition
- Battlefield surveillance
- Reconnaissance of opposing forces and terrain
- Targeting
- Battle damage assessment
- Nuclear, biological, and chemical attack detection

Environmental Applications

- Forest fire detection
- Bio-complexity mapping of environment
- Flood detection
- Precision Agriculture
- Air and water pollution

Health Applications

- Telemonitoring of human physiological data
- Tracking and monitoring doctors and patients inside a hospital
- Drug administration in hospitals

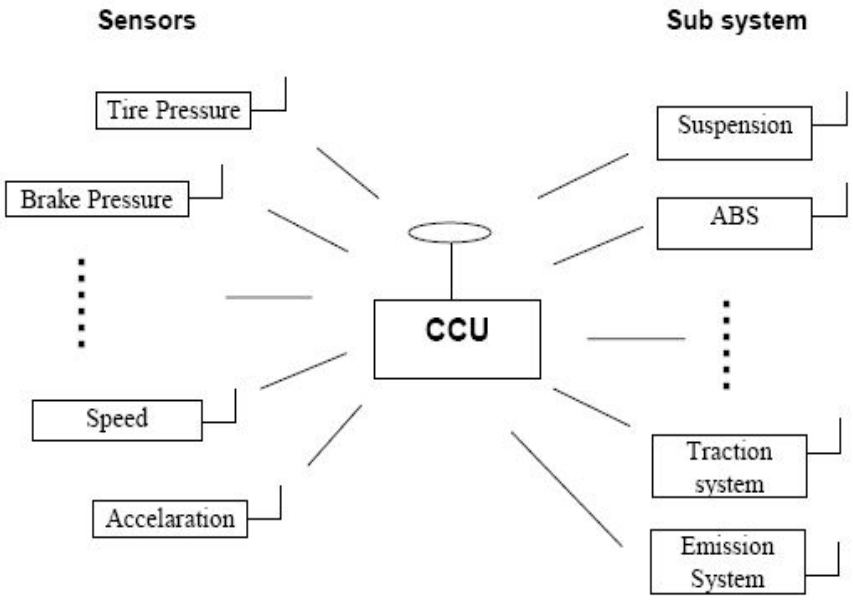
Home and Office Applications

- Home and office automation
- Smart environment

Automotive Applications

- Reduces wiring effects
- Measurements in chambers and rotating parts
- Remote technical inspections
- Conditions monitoring e.g. at a bearing

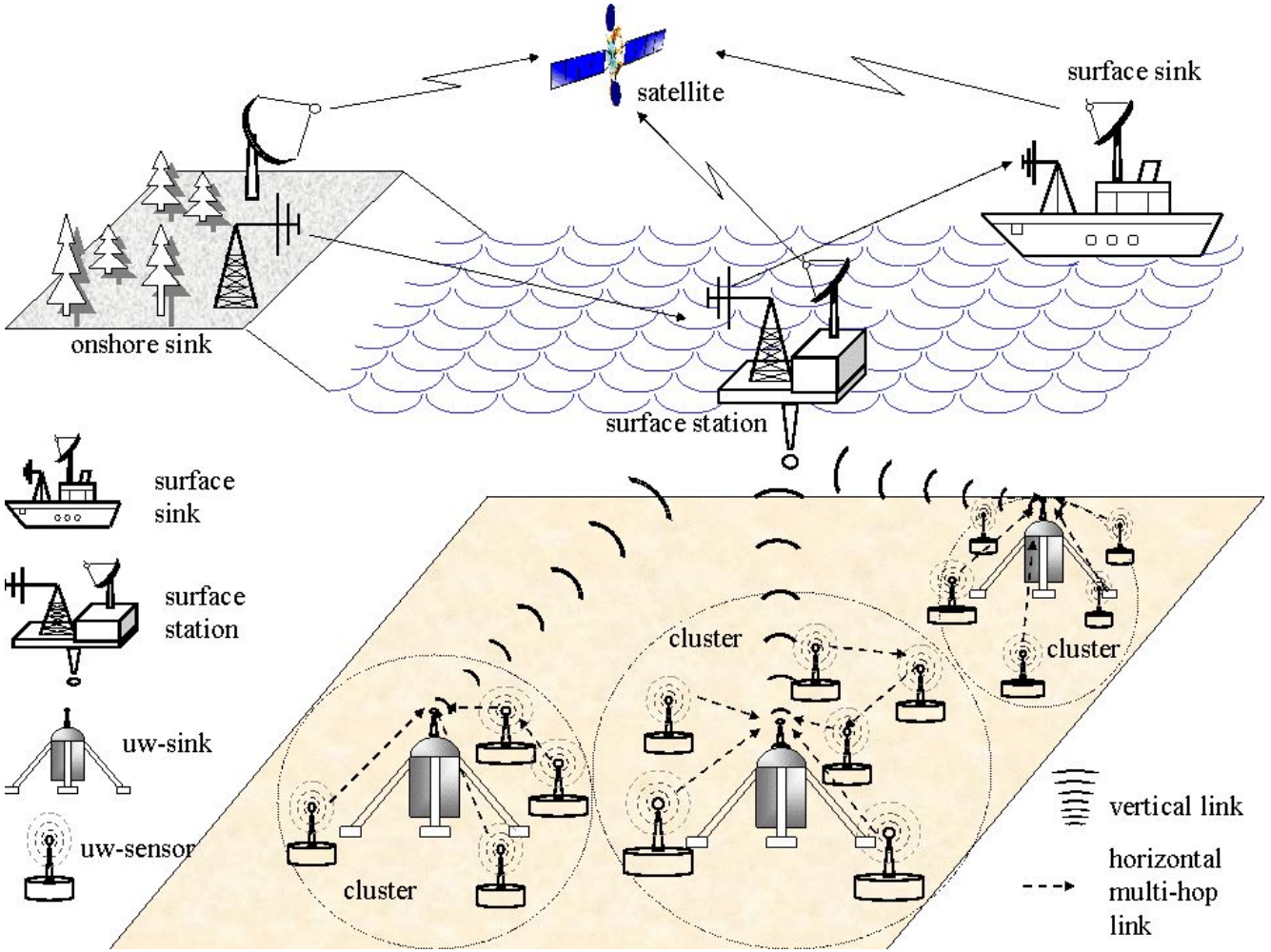
Automotive Applications



Other Commercial Applications

- Environmental control in office buildings (estimated energy savings $55 billion per year!)
- Interactive museums
- Detecting and monitoring car thefts
- Managing inventory control
- Vehicle tracking and detection

Underwater Acoustic Sensor Networks ref. Georgia Institute of Technology



Factors Influencing WSN Design

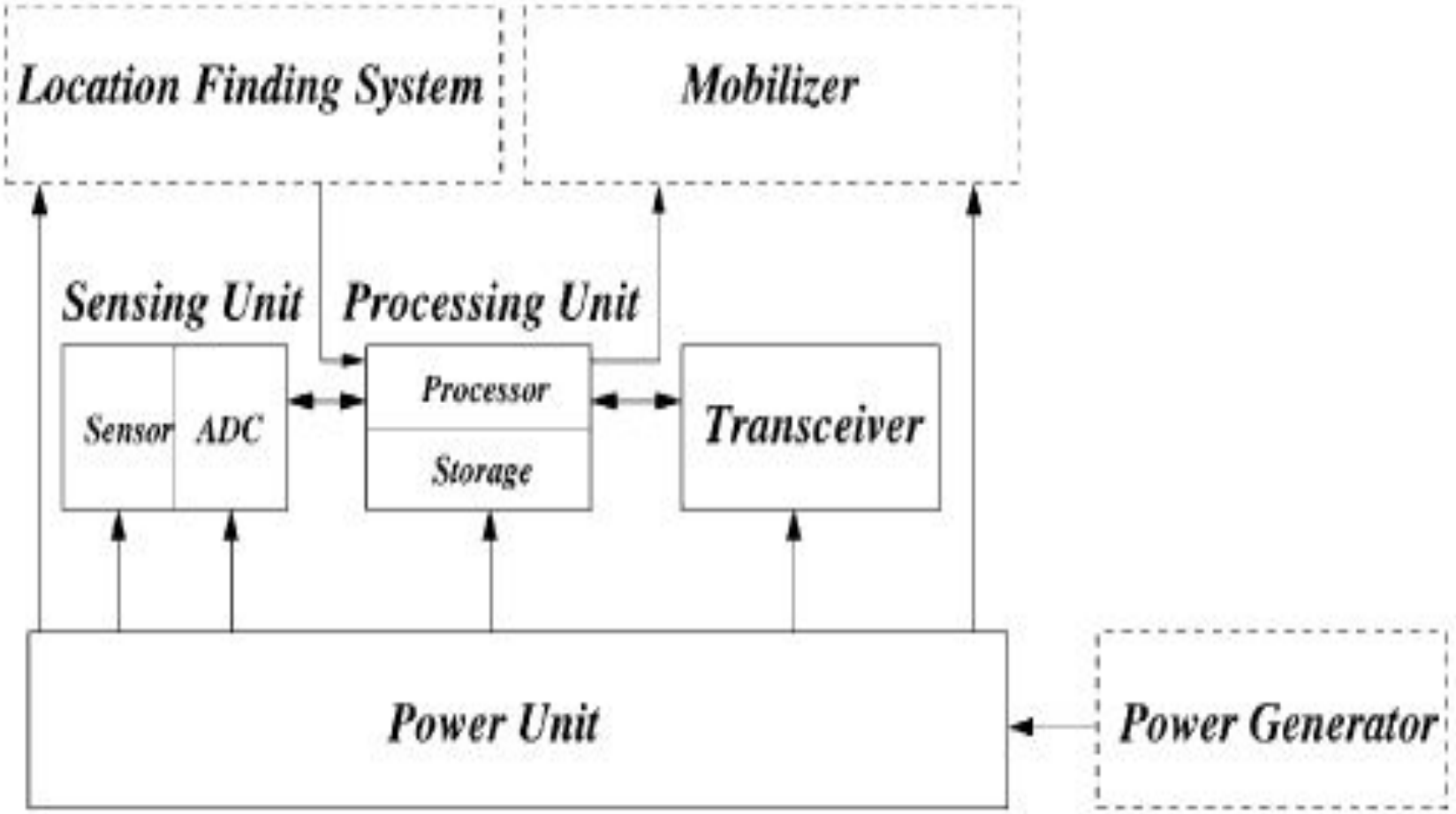
- Fault tolerance
- Scalability
- Production costs
- Hardware constraints
- Sensor network topology
- Environment
- Transmission media
- Power Consumption
  - Sensing
  - Communication
  - Data processing

Sensor Nodes

- Sensor Nodes

Crossbow Sensor Node

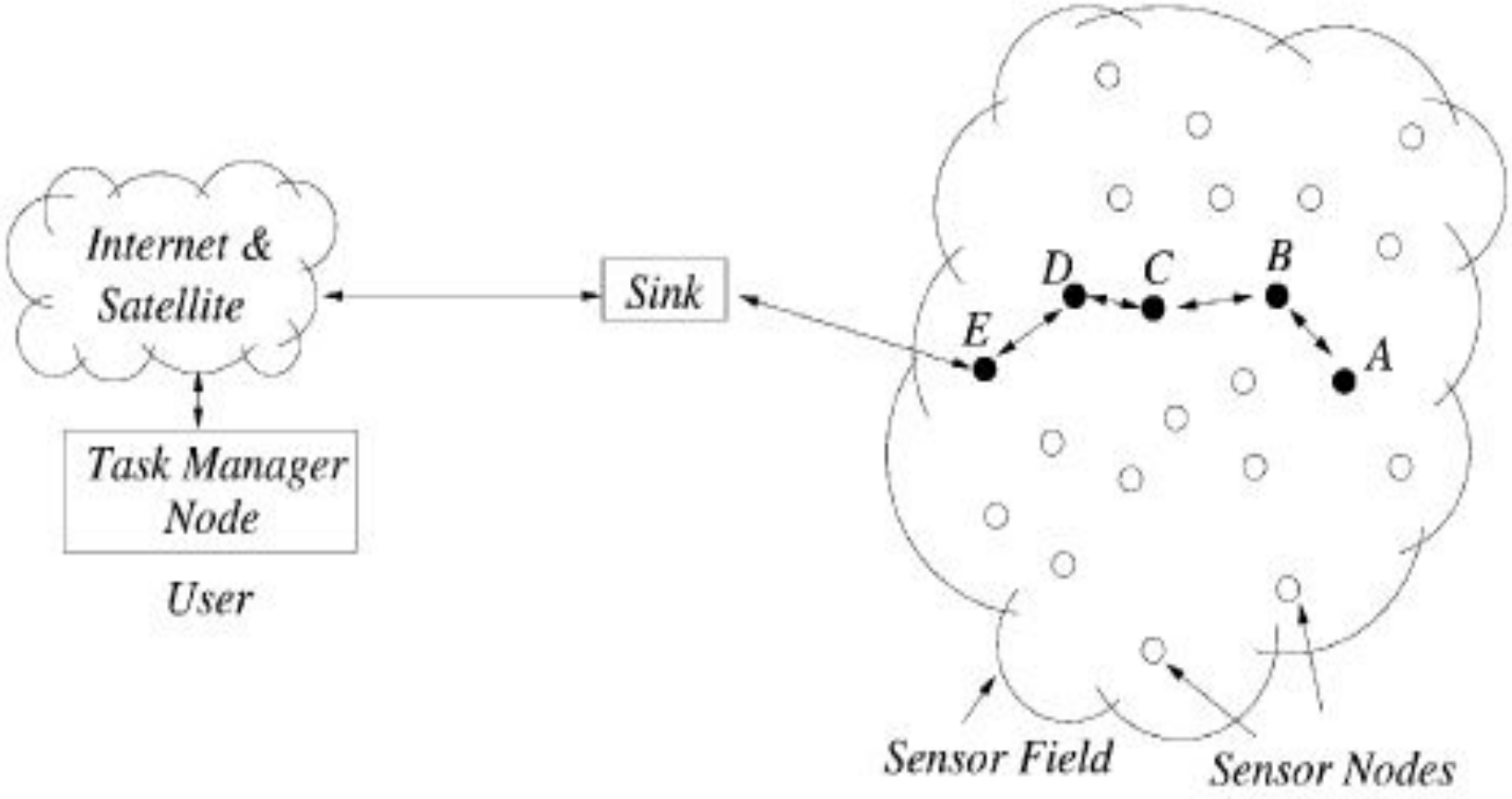
Sensor Node Components



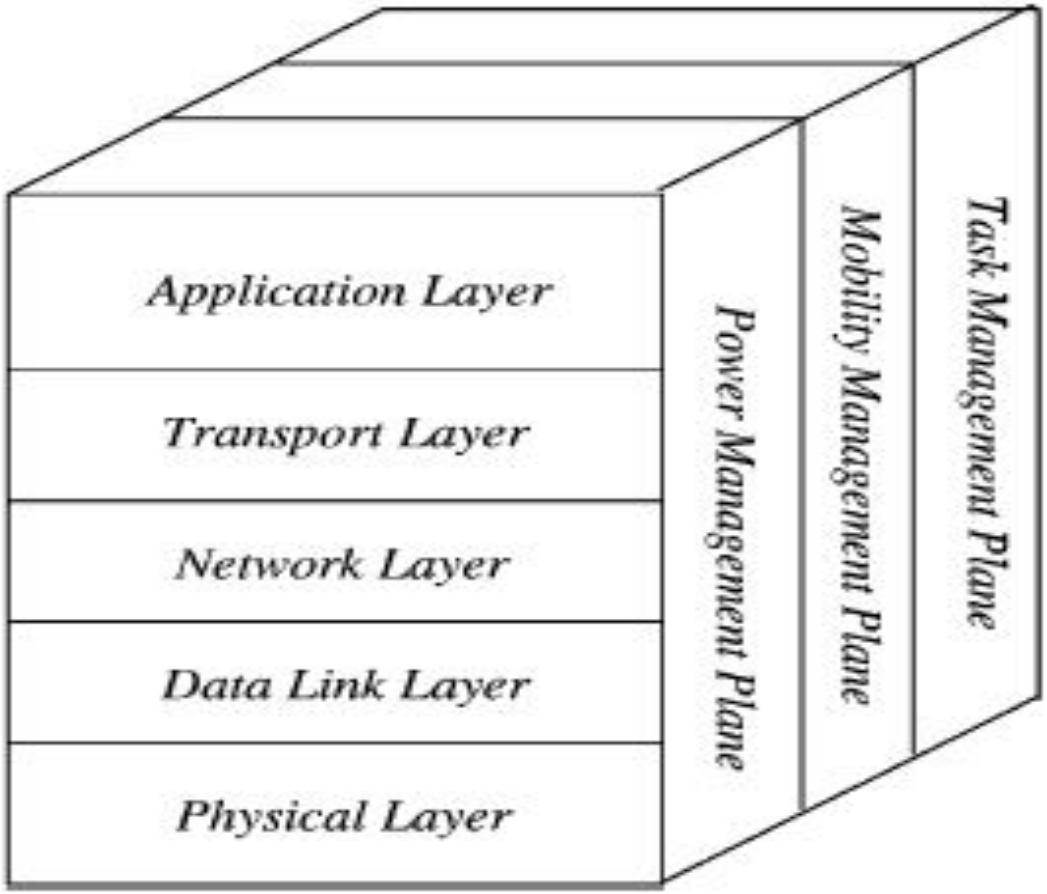
Sensor Node Components

- Sensing Unit
- Processing Unit
- Transceiver Unit
- Power Unit
- Location Finding System (optional)
- Power Generator (optional)
- Mobilizer (optional)

WSN Communication Architecture



WSN Protocol Stack



WSN Protocols

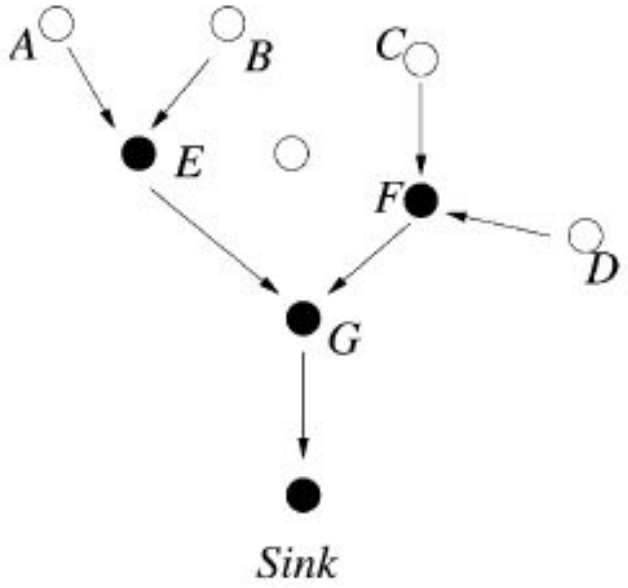
- Sensor management protocol
  - Provides software operations needed to perform administrative tasks e.g. moving sensor nodes, turning them on an off
- Sensor query and data dissemination protocol
  - Provides user applications with interfaces to issue queries and respond to queries
  - Sensor query and tasking language (SQTL)
- Directed diffusion
- Sensor MAC (S-MAC)
- IEEE 802.15.4

Data-Centric Routing

- Interest dissemination is performed to assign sensing tasks to sensor nodes
  - Sinks broadcast the interest
  - Sensor nodes broadcast an advertisement for available data
- Requires attribute-based naming
  - Users are more interested in querying the attribute of the phenomenon, rather than querying an individual node
  - E.g. the sensor nodes in the area where temperature is greater than 75 F

Data Aggregation in WSNs

- Data coming from multiple sensor nodes are aggregated if they are about the same attribute of the phenomenon when they reach the same routing node on the way back to the sink
  - Solves implosion and overlap problem
  - Energy efficient

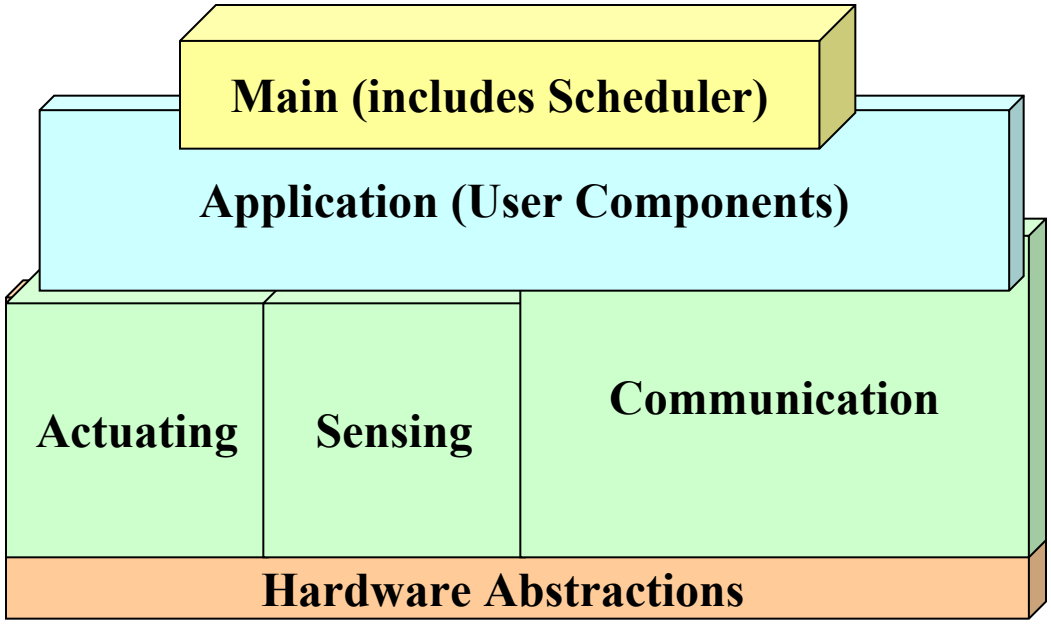


WSN Operating Systems

- TinyOS
- Contiki
- MANTIS
- BTnut
- SOS
- Nano-RK

TinyOS

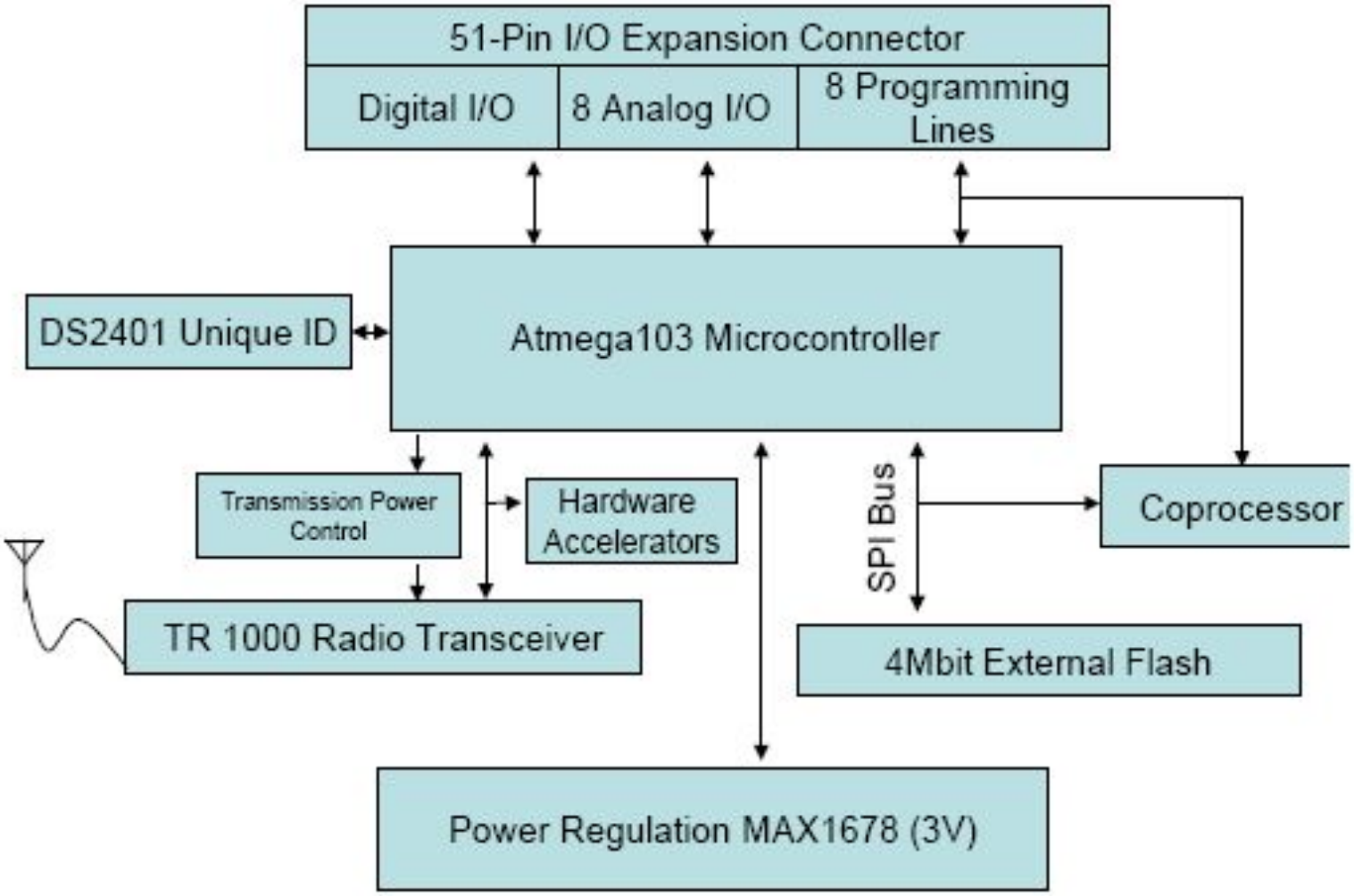
- Event-driven programming model instead of multithreading
- TinyOS and its programs written in nesC



TinyOS Characersitics

- Small memory footprint
  - non-preemptable FIFO task scheduling
- Power Efficient
  - Puts microcontroller to sleep
  - Puts radio to sleep
- Concurrency-Intensive Operations
  - Event-driven architecture
  - Efficient Interrupts and event handling
- No Real-time guarantees

MICA Sensor Mote



MICA Mote Specifications

- 4 MHz ATMEGA103L Microprocessor
- 128 KB of Flash Program Memory
- 4KB RAM
- 10 bit Analog to Digital Converter (ADC)
- 3 Hardware Timers
- Serial Peripheral Interface (SPI) bus
- External UART
- A coprocessor AT90LS2343 (to handle wireless reprogramming)
- DS2401 silicon serial number (provides unique ID to nodes)
- RF Monolithics TR1000 transceiver
- External 4Mbit Atmel AT45DB041B Serial Flash Chip (for persistent data storage)
- Maxim1678 DC-DC Converter (provides a constant 3.0 V supply)

Smart Dust Mote Specifications

- 4 MHz Atmel AVR 8535 Microprocessor
- 8 KB Instruction Flash Memory
- 512 Bytes RAM
- 512 Bytes EEPROM
- Total Stored Energy approx. 1 Joule
- TinyOS Operating System (OS) with 3500 bytes OS code space and 4500 bytes available code space

WSN Development Platforms

- Crossbow
- Dust Networks
- Sensoria Corporation
- Ember Corporation
- Worldsens

WSN Simulators

- NS-2
- GloMoSim
- OPNET
- SensorSim
- J-Sim
- OMNeT++
- Sidh
- SENS

WSN Emulators

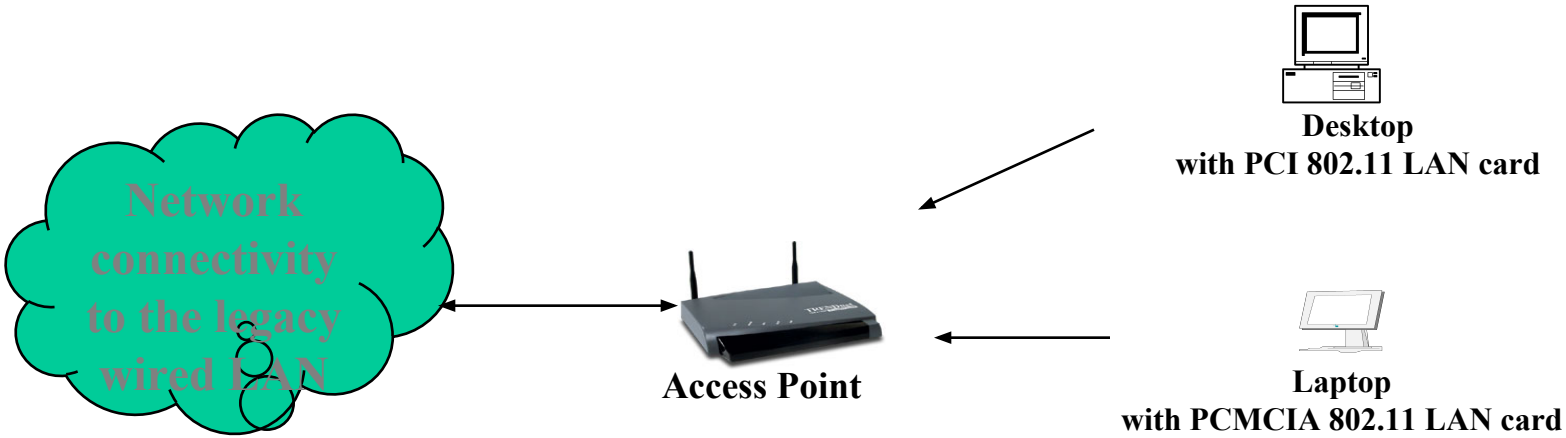
- TOSSIM
- ATEMU
- Avrora
- EmStar

Conclusion

- WSNs possible today due to technological advancement in various domains
- Envisioned to become an essential part of our lives
- Design Constraints need to be satisfied for realization of sensor networks
- Tremendous research efforts being made in different layers of WSNs protocol stack

802.11 Wireless LAN

- **Provides network connectivity over wireless media**
- **An Access Point (AP) is installed to act as Bridge between Wireless and Wired Network**
- **The AP is connected to wired network and is equipped with antennae to provide wireless connectivity**



- **Range (Distance between Access Point and WLAN client) depends on structural hindrances and RF gain of the antenna at the Access Point**
- **To service larger areas, multiple APs may be installed with a 20-30% overlap**
- **A client is always associated with one AP and when the client moves closer to another AP, it associates with the new AP (Hand-Off)**
- **Three flavors:**
  - 802.11b
  - 802.11a
  - 802.11g

- Issued in four stages
- First part in 1997
  - IEEE 802.11
  - Includes MAC layer and three physical layer specifications
  - Two in 2.4-GHz band and one infrared
  - All operating at 1 and 2 Mbps
- Two additional parts in 1999
  - IEEE 802.11a
    - 5-GHz band up to 54 Mbps
  - IEEE 802.11b
    - 2.4-GHz band at 5.5 and 11 Mbps
- Most recent in 2002
  - IEEE 802.g extends IEEE 802.11b to higher data rat

- Three physical media
- Direct-sequence spread spectrum
  - 2.4 GHz ISM band at 1 Mbps and 2 Mbps
  - Up to seven channels, each 1 Mbps or 2 Mbps, can be used
  - Depends on bandwidth allocated by various national regulations
    - 13 in most European countries
    - One in Japan
  - Each channel bandwidth 5 MHz
  - Encoding scheme DBPSK for 1-Mbps and DQPSK for 2-Mbps

- Frequency-hopping spread spectrum
  - 2.4 GHz ISM band at 1 Mbps and 2 Mbps
  - Uses multiple channels
  - Signal hopping from one channel to another based on a pseudonoise sequence
  - 1-MHz channels are used
  - 23 channels in Japan
  - 70 in USA
- Hopping scheme adjustable
  - E.g. Minimum hop rate forUSA is 2.5 hops per second
  - Minimum hop distance 6 MHz in North America and most of Europe and 5 MHz in Japan
- Two-level Gaussian FSK modulation for 1-Mbps
  - Bits encoded as deviations from current carrier frequency
- For 2 Mbps, four-level GFSK used
  - Four different devia

- Omnidirectional
- Range up to 20 m
- 1 Mbps used 16-PPM (pulse position modulation)
  - Each group of 4 data bits mapped into one of 16-PPM symbols
  - Each symbol a string of 16 bits
  - Each 16-bit string consists of fifteen 0s and one binary 1
- For 2-Mbps, each group of 2 data bits is mapped into one of four 4-bit sequences
  - Each sequence consists of three 0s and one binary 1
  - Intensity modulation
    - Presence of signal corresponds to 1

802.11a

- 5-GHz band
- Uses orthogonal frequency division multiplexing (OFDM)
  - Not spread spectrum
- Also called multicarrier modulation
- Multiple carrier signals at different frequencies
- Some bits on each channel
  - Similar to FDM but all subchannels dedicated to single source
- Data rates 6, 9, 12, 18, 24, 36, 48, and 54 Mbps
- Up to 52 subcarriers modulated using BPSK, QPSK, 16-QAM, or 64-QAM
  - Depending on rate
  - Subcarrier frequency spacing 0.3125 MHz
  - Convolutional code at rate of 1/2, 2/3, or 3/4 provides forward error correction

802.11b

- Extension of 802.11 DS-SS scheme
- 5.5 and 11 Mbps
- Chipping rate 11 MHz
  - Same as original DS-SS scheme
  - Same occupied bandwidth
  - Complementary code keying (CCK) modulation to achieve higher data rate in same bandwidth at same chipping rate
  - CCK modulation complex
    - Overview on next slide
  - Input data treated in blocks of 8 bits at 1.375 MHz
    - 8 bits/symbol × 1.375 MHz = 11 Mbps
    - Six of these bits mapped into one of 64 code sequences
    - Output of map

802.11g

- Higher-speed extension to 802.11b
- Combines physical layer encoding techniques used in 802.11a and 802.11b to provide service at a variety of data rates

Thank you

SRI CHANDRASEKHARENDRA SARASWATHI VISWA MAHAVIDHYALAYA

(Deemed to be university u/s 3 of UGC act 1956)
(Accredited with "A" by NAAC)
Enathur, Kanchipuram – 631561. Tamilnadu
www.kanchiuniv.ac.in

Computer Networks
Unit II- Data Link Layer

Name of the Faculty : Dr. N Kumaran

Assistant Professor, Dept. of CSE

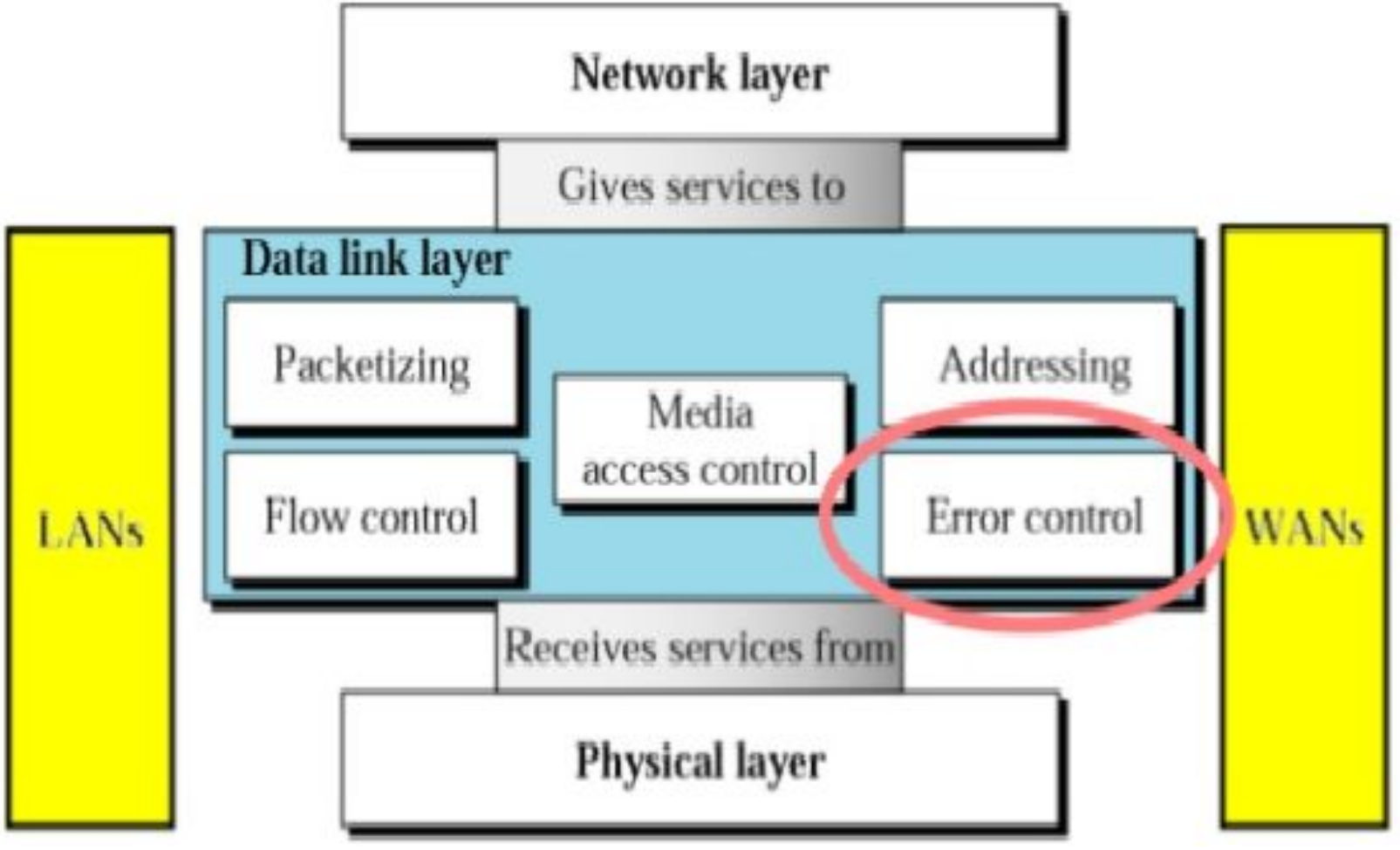
E-Mail : nkumaran@kanchiuniv.ac.in

Outline

- Data Link Layer
- Error Detection and Correction
- **Types of Errors**
- **Detection**
- **Correction**

*

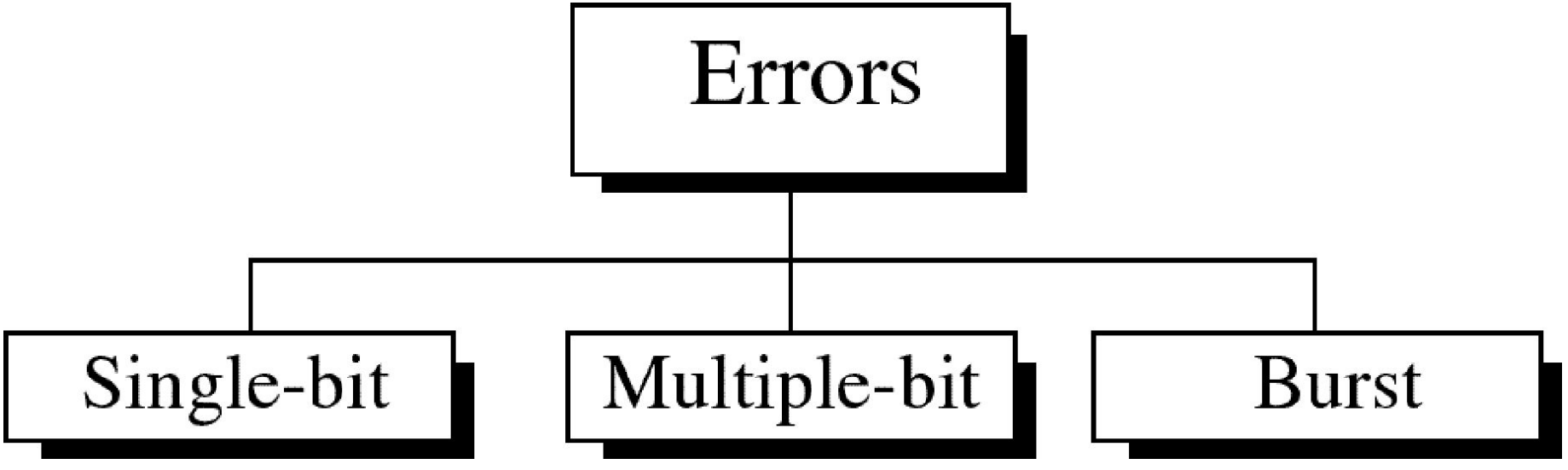
Data Link Layer



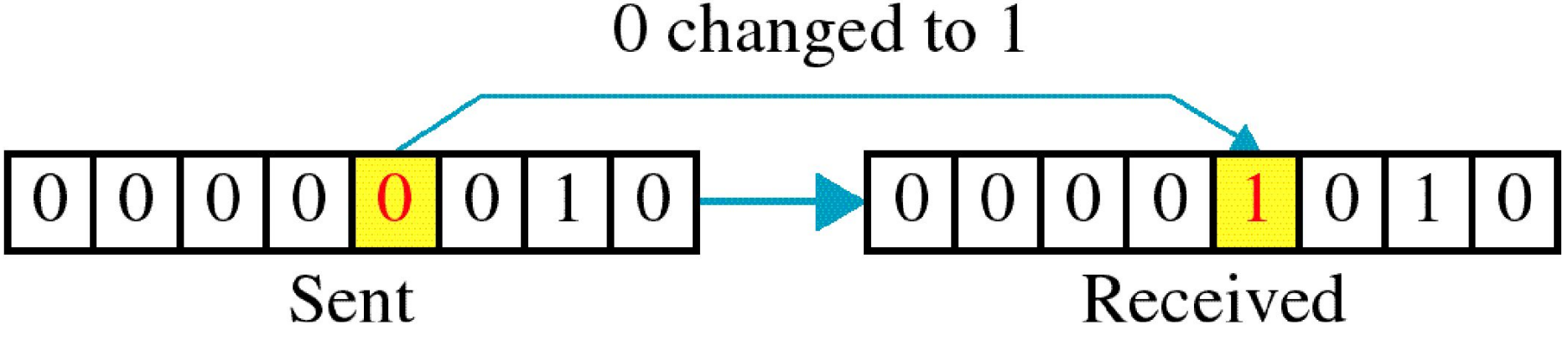
Basic concepts

- Networks must be able to transfer data from one device to another with complete accuracy.
- Data can be corrupted during transmission.
- For reliable communication, errors must be detected and corrected.
- **Error detection and correction** are implemented either at the **data link layer** or the **transport layer** of the OSI model.

Types of Errors



Single-bit error

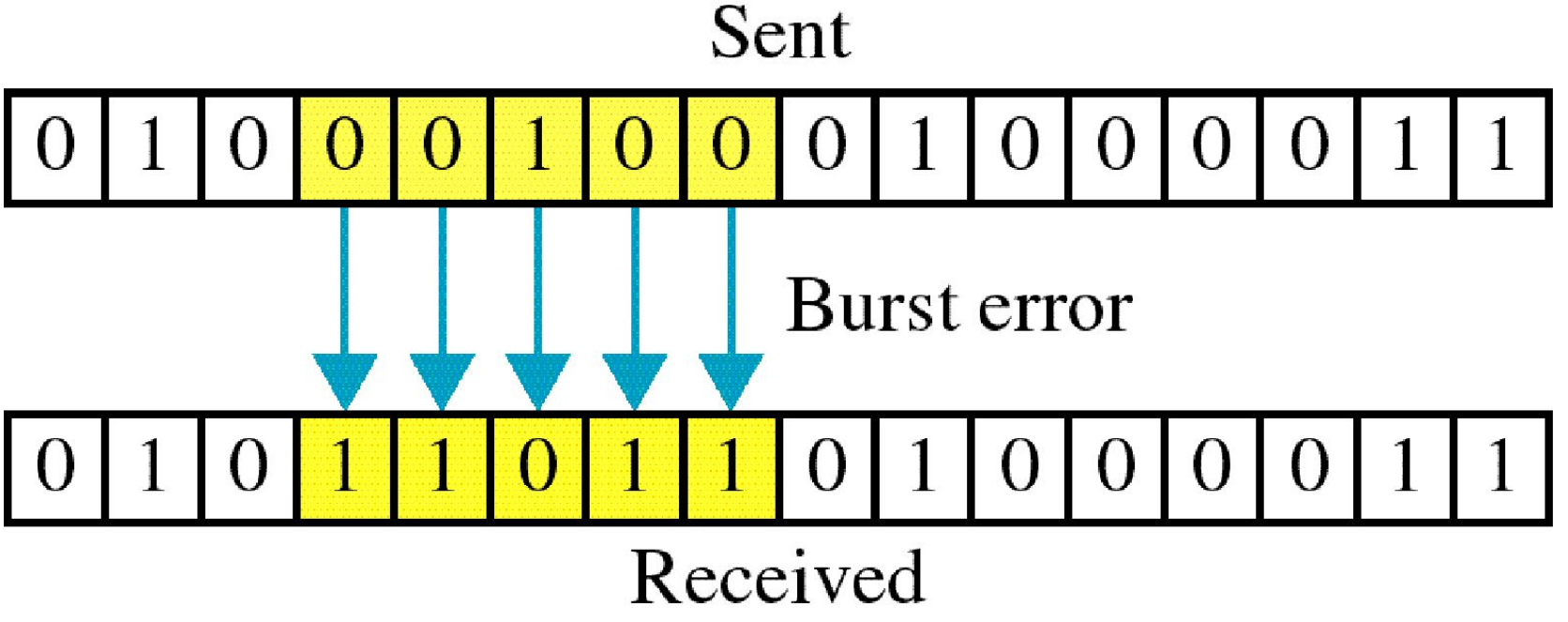


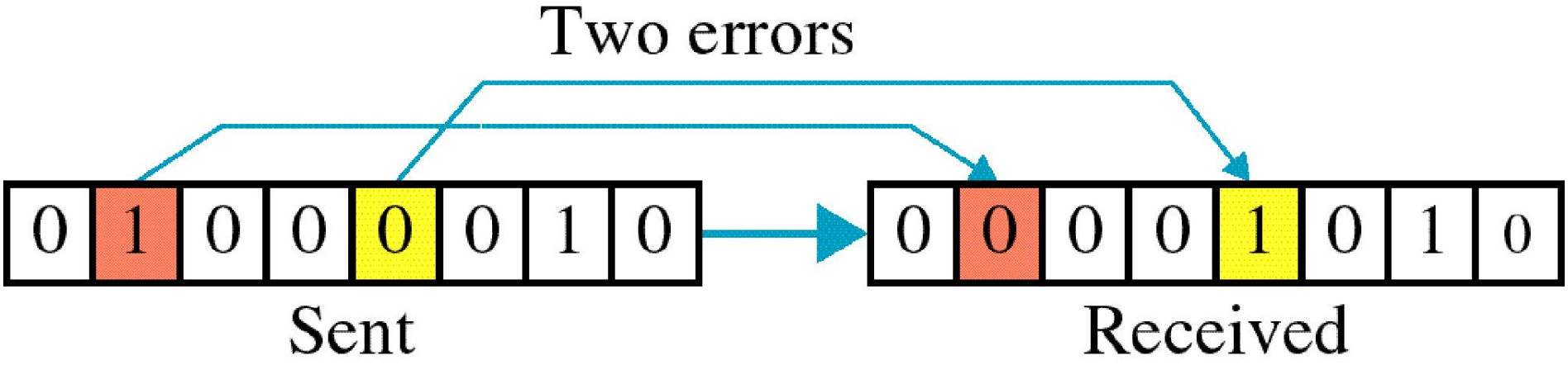
Single bit errors are the **least likely** type of errors in serial data transmission because the noise must have a very short duration which is very rare. However this kind of errors can happen in parallel transmission.

Example:

- ★ If data is sent at 1Mbps then each bit lasts only 1/1,000,000 sec. or 1 μs.
- ★ For a single-bit error to occur, the noise must have a duration of only 1 μs, which is very rare.

Burst error


Sent
0 1 0 0 0 1 0 0 0 1 0 0 0 0 1 1
Burst error
0 1 0 1 1 0 1 1 0 1 0 0 0 0 1 1
Received


Two errors
0 1 0 0 0 0 1 0
Sent
0 0 0 0 1 0 1 0
Received

The term **burst error** means that two or more bits in the data unit have changed from 1 to 0 or from 0 to 1.

Burst errors does not necessarily **mean that the errors occur in consecutive bits**, the length of the burst is measured from the first corrupted bit to the last corrupted bit. Some bits in between may not have been corrupted.

★ **Burst error is most likely to happen in serial transmission** since the duration of noise is normally longer than the duration of a bit.

★ The number of bits affected depends on the data rate and duration of noise.

Example:

- If data is sent at rate = 1Kbps then a noise of 1/100 sec can affect 10 bits.(1/100*1000)

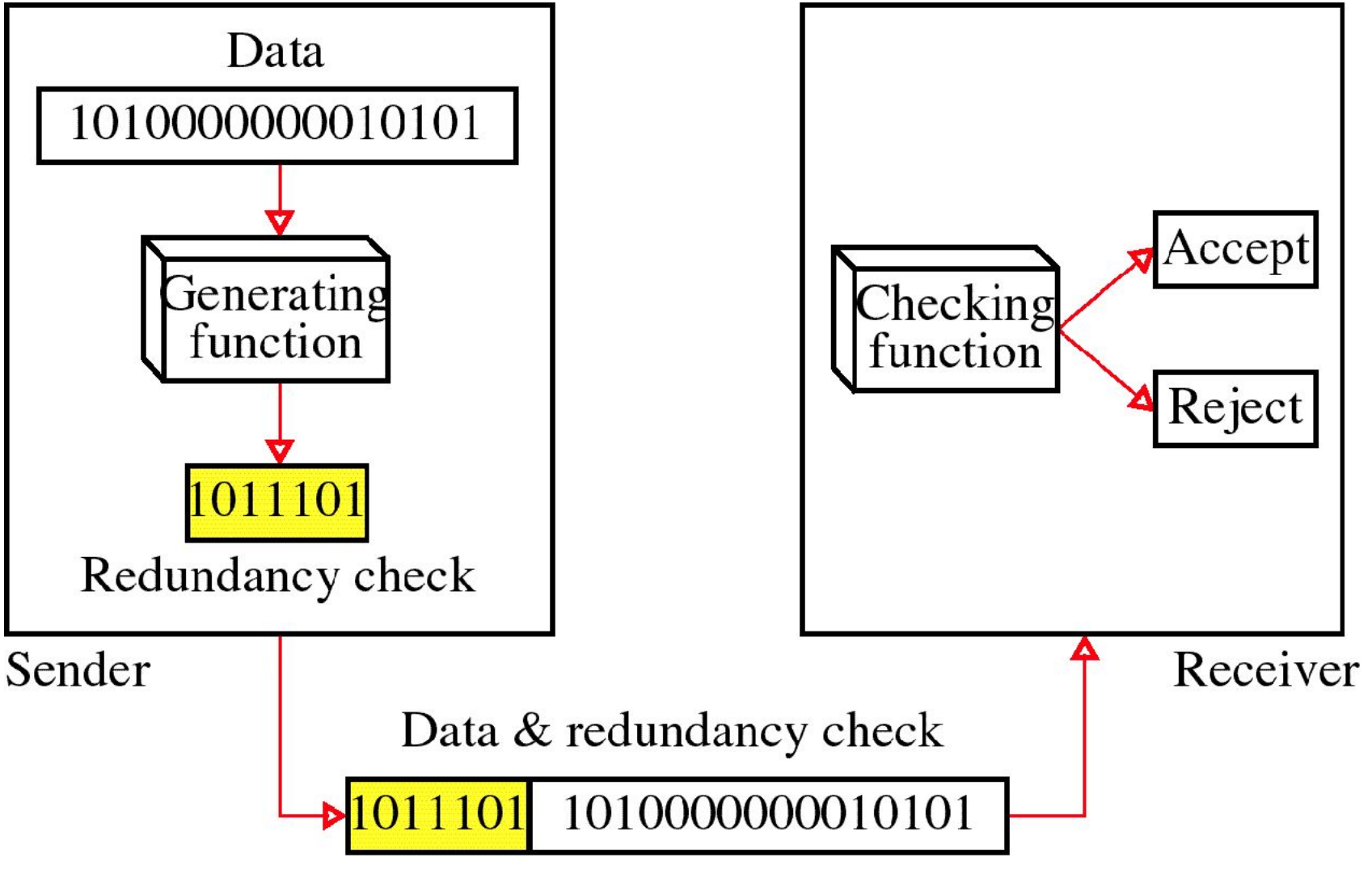
- If same data is sent at rate = 1Mbps then a noise of 1/100 sec can affect 10,000 bits.($1/100*10^6$)

Error detection

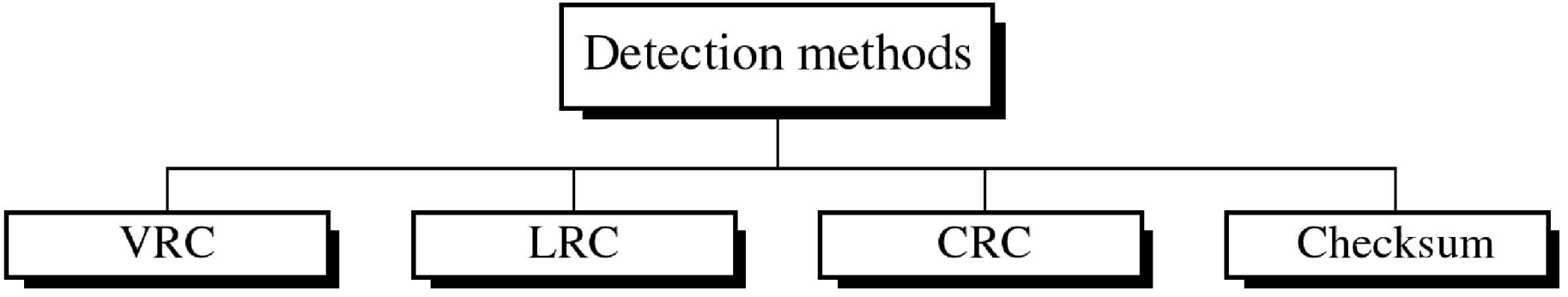
Error detection means to decide whether the received data is correct or not without having a copy of the original message.

Error detection **uses the concept of redundancy**, **which means** adding extra bits for detecting errors at the destination.

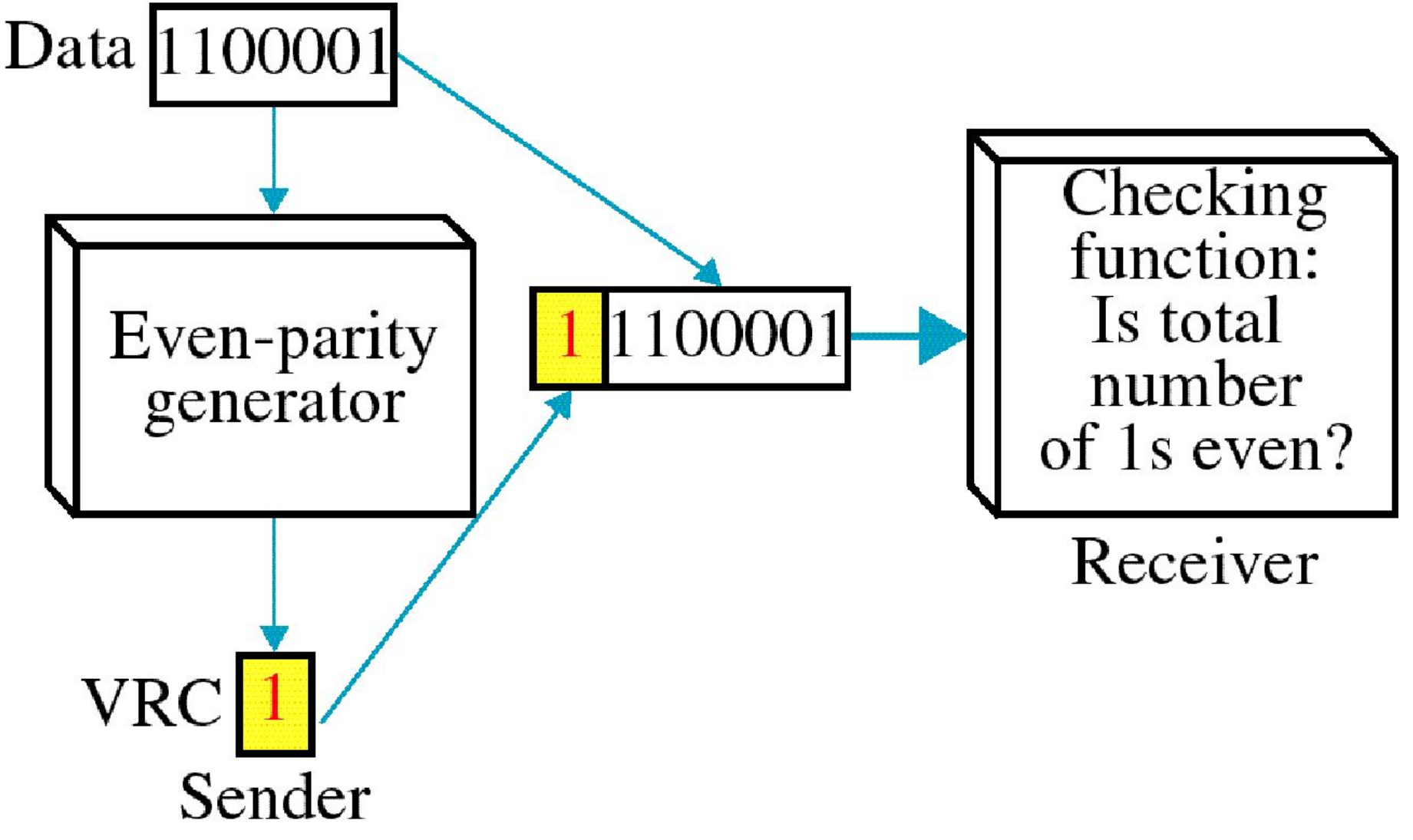
Redundancy


Data
1010000000010101
Generating function
1011101
Redundancy check
Sender
Checking function
Accept
Reject
Receiver
Data & redundancy check
1011101
1010000000010101

Four types of redundancy checks are used in data communications



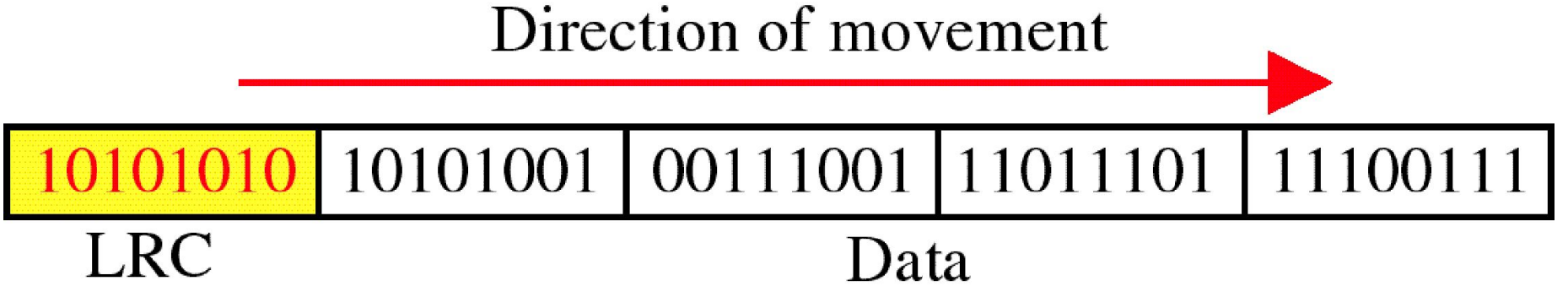
Vertical Redundancy Check
VRC



Performance

- It can detect single bit error
- It can detect burst errors only if the total number of errors is odd.

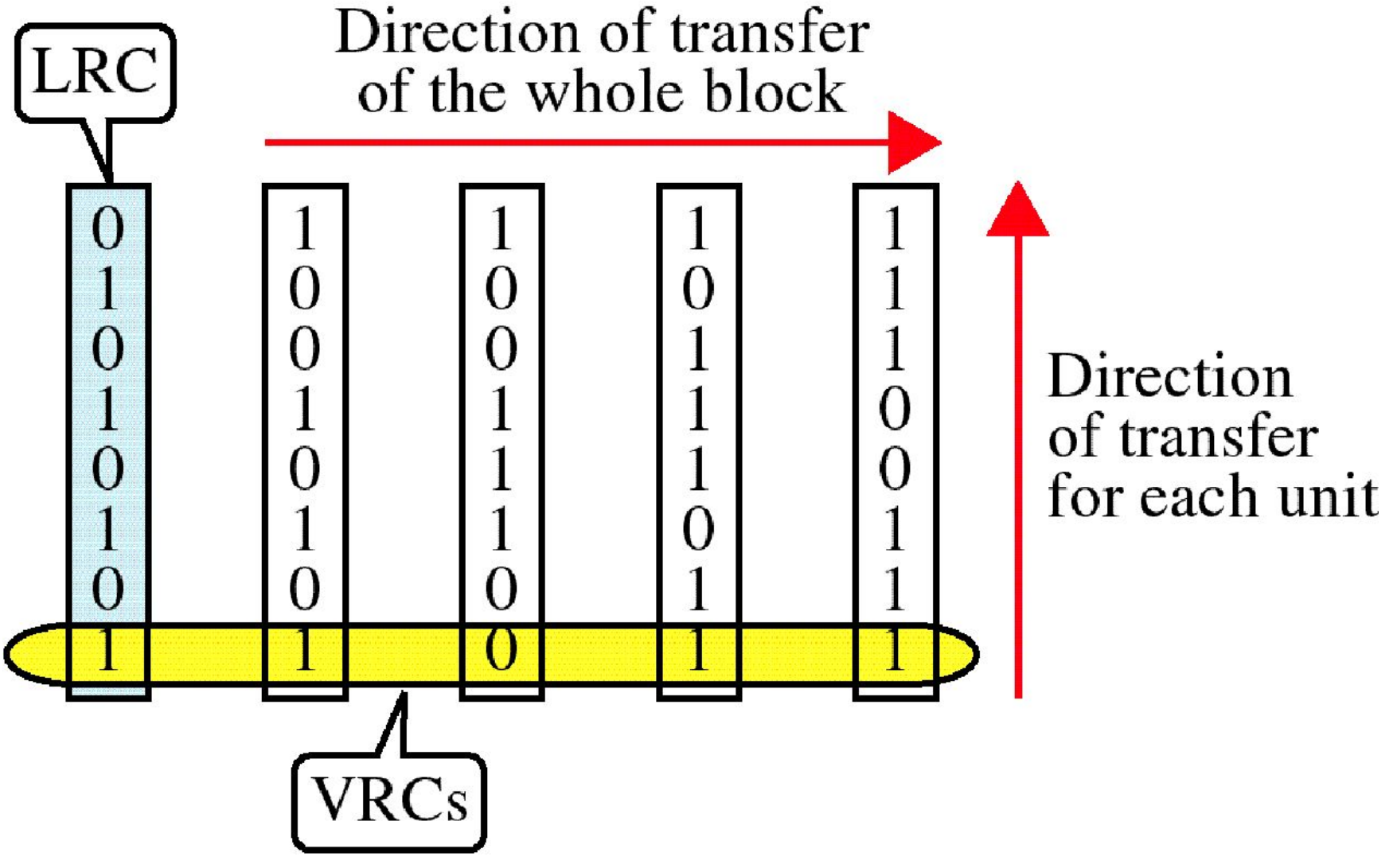
Longitudinal Redundancy Check
LRC



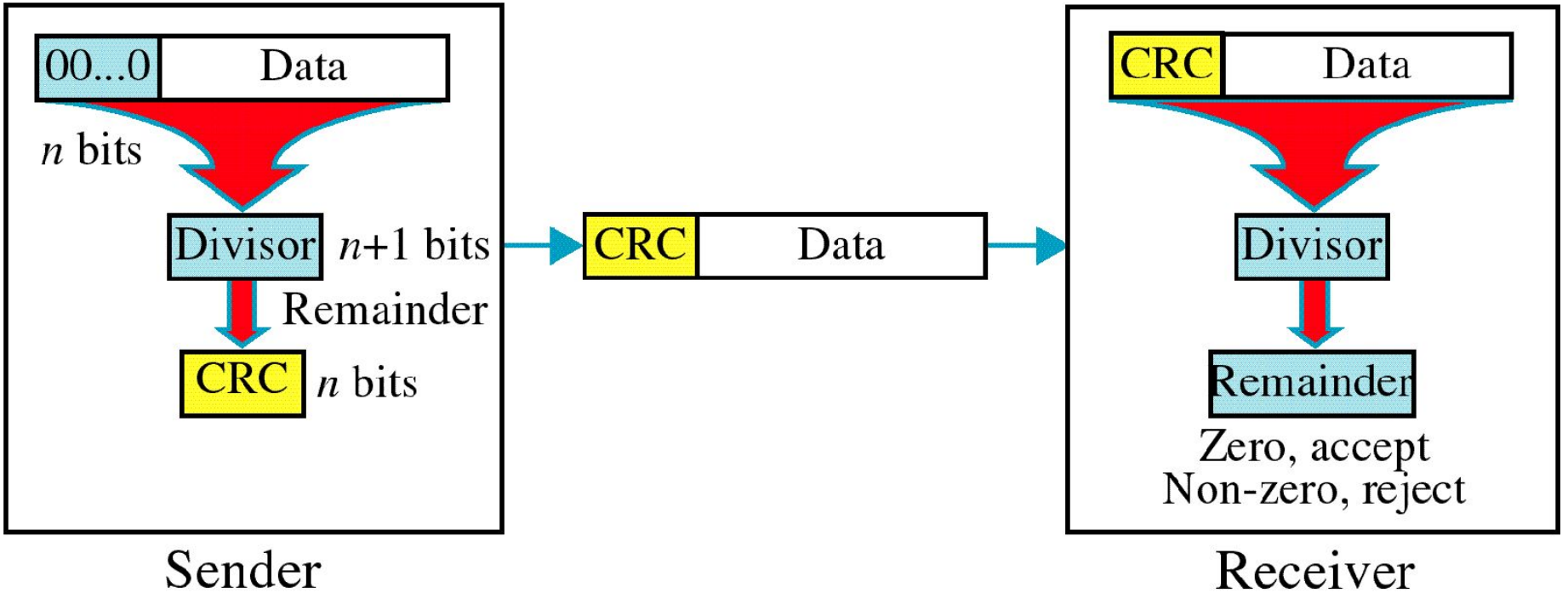
Performance

- LCR increases the likelihood of detecting burst errors.
- If two bits in one data units are damaged and two bits in exactly the same positions in another data unit are also damaged, the LRC checker will not detect an error.

VRC and LRC


LRC
Direction of transfer
of the whole block
0 1 0 1 0 1 0 1
1 0 0 1 0 1 0 1
1 0 0 1 1 1 1 0
1 0 1 1 1 0 1 1
1 1 1 0 0 1 1 1
Direction
of transfer
for each unit
VRCs

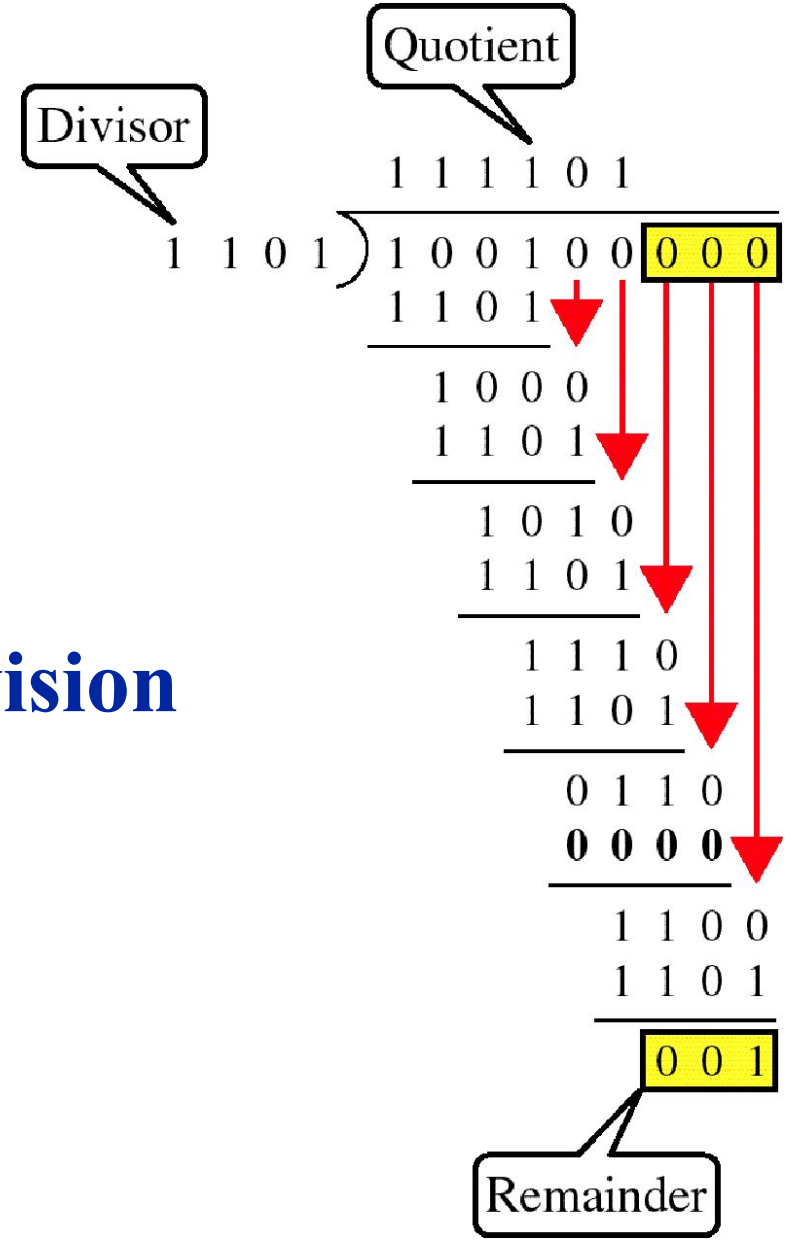
Cyclic Redundancy Check
CRC



Cyclic Redundancy Check

- Given a k-bit frame or message, the transmitter generates an n-bit sequence, known as a *frame check sequence (FCS)*, so that the resulting frame, consisting of (k+n) bits, is exactly divisible by some predetermined number.
- The receiver then divides the incoming frame by the same number and, if there is no remainder, assumes that there was no error.

Binary Division



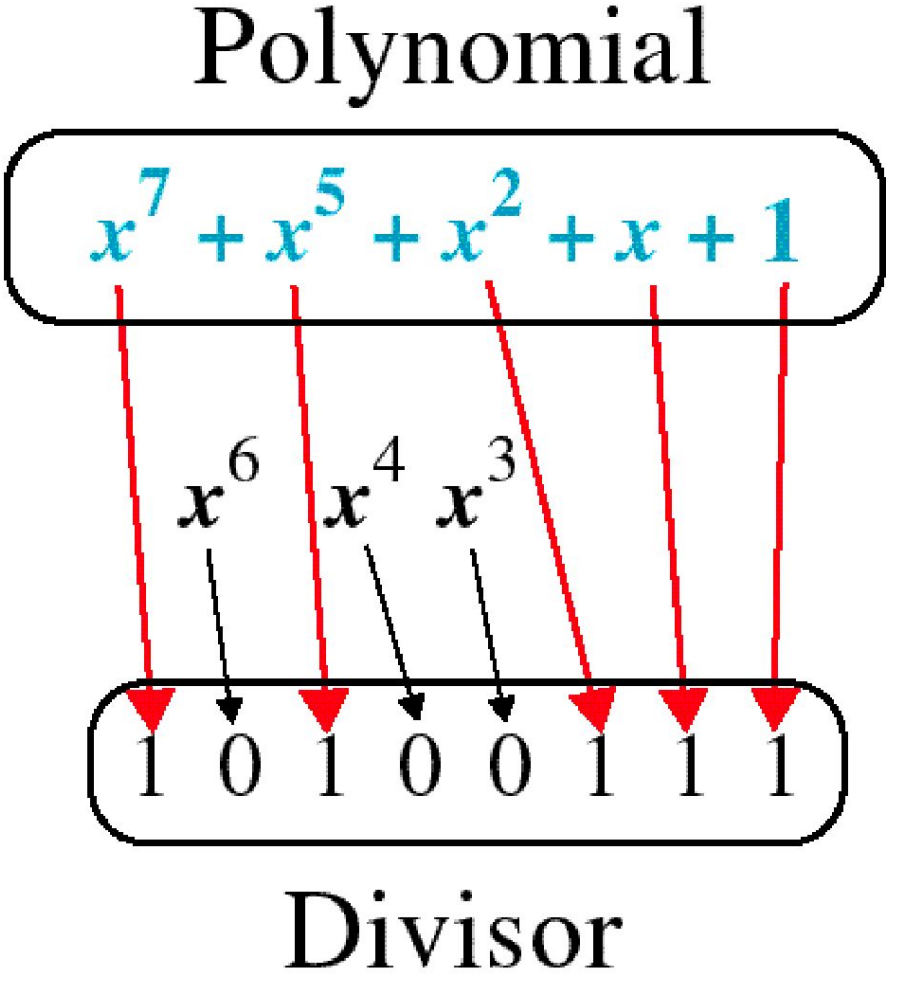
1101 divisor

100100001
dividend

Polynomial

$$x^7 + x^5 + x^2 + x + 1$$

Polynomial and Divisor



Standard Polynomials

CRC-12

$$x^{12} + x^{11} + x^3 + x + 1$$

CRC-16

$$x^{16} + x^{15} + x^2 + 1$$

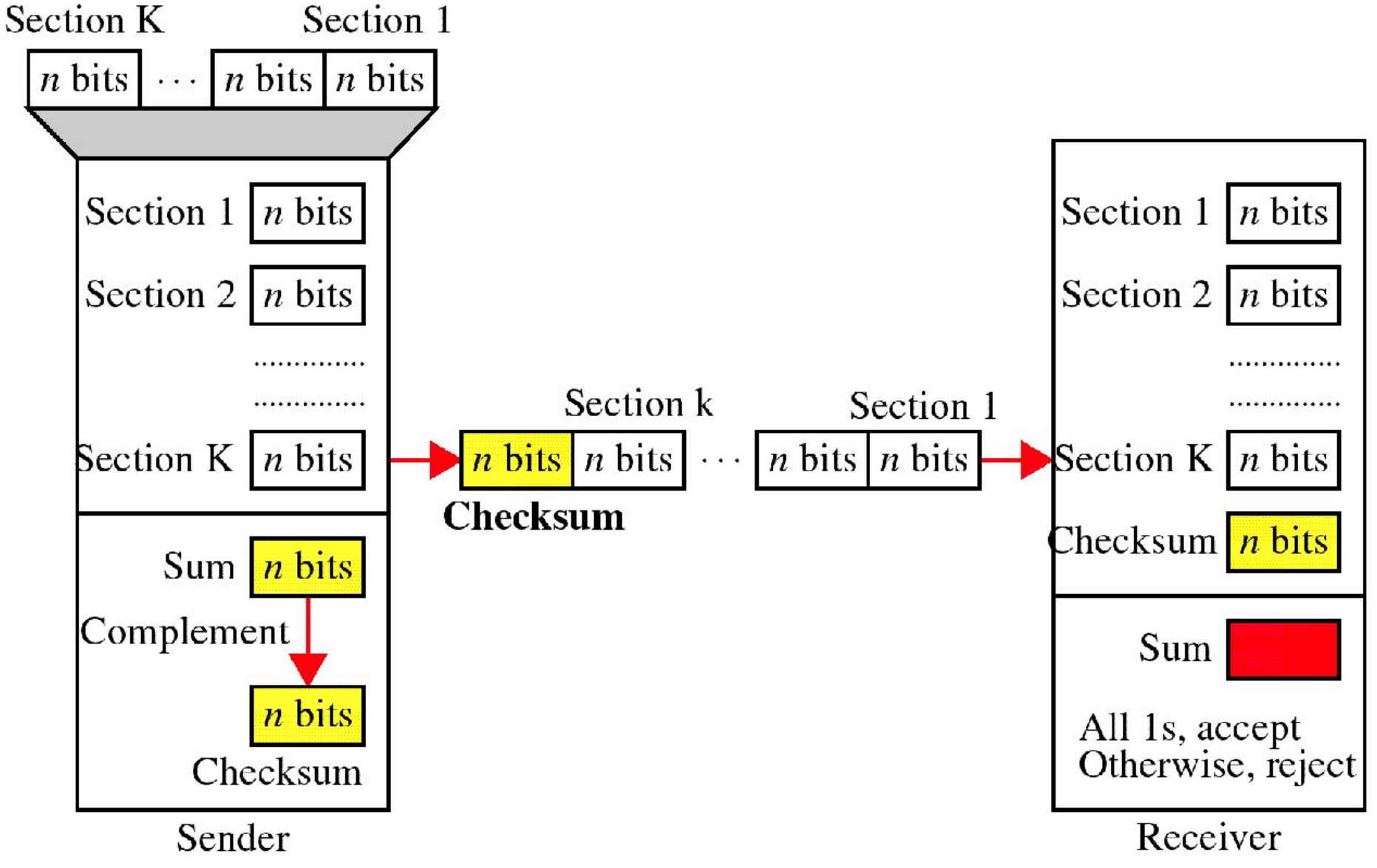
CRC-ITU

$$x^{16} + x^{12} + x^5 + 1$$

CRC-32

$$x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1$$

Checksum


Section K
Section 1
n bits
n bits
n bits
Section 1
n bits
Section 2
n bits
Section K
n bits
Sum
n bits
Complement
n bits
Checksum
Sender
Section k
Section 1
n bits
n bits
n bits
n bits
Checksum
Section 1
n bits
Section 2
n bits
Section K
n bits
Checksum
n bits
Sum
All 1s, accept
Otherwise, reject
Receiver

At the sender

- The unit is divided into *k* sections, each of *n* bits.
- All sections are added together using one's complement to get the sum.
- The sum is complemented and becomes the checksum.
- The checksum is sent with the data

At the receiver

- The unit is divided into *k* sections, each of *n* bits.
- All sections are added together using one's complement to get the sum.
- The sum is complemented.
- If the result is zero, the data are accepted: otherwise, they are rejected.

Performance

- The checksum detects all errors involving an odd number of bits.
- It detects most errors involving an even number of bits.
- If one or more bits of a segment are damaged and the corresponding bit or bits of opposite value in a second segment are also damaged, the sums of those columns will not change and the receiver will not detect a problem.

Error Correction

It can be handled in two ways:

1) receiver can have the sender retransmit the entire data unit.
2) The receiver can use an error-correcting code, which automatically corrects certain errors.

Single-bit error correction

To correct an error, the receiver reverses the value of the altered bit. To do so, it must know which bit is in error.

Number of redundancy bits needed

- Let data bits = m
- Redundancy bits = r

$\therefore$ Total message sent = $m+r$

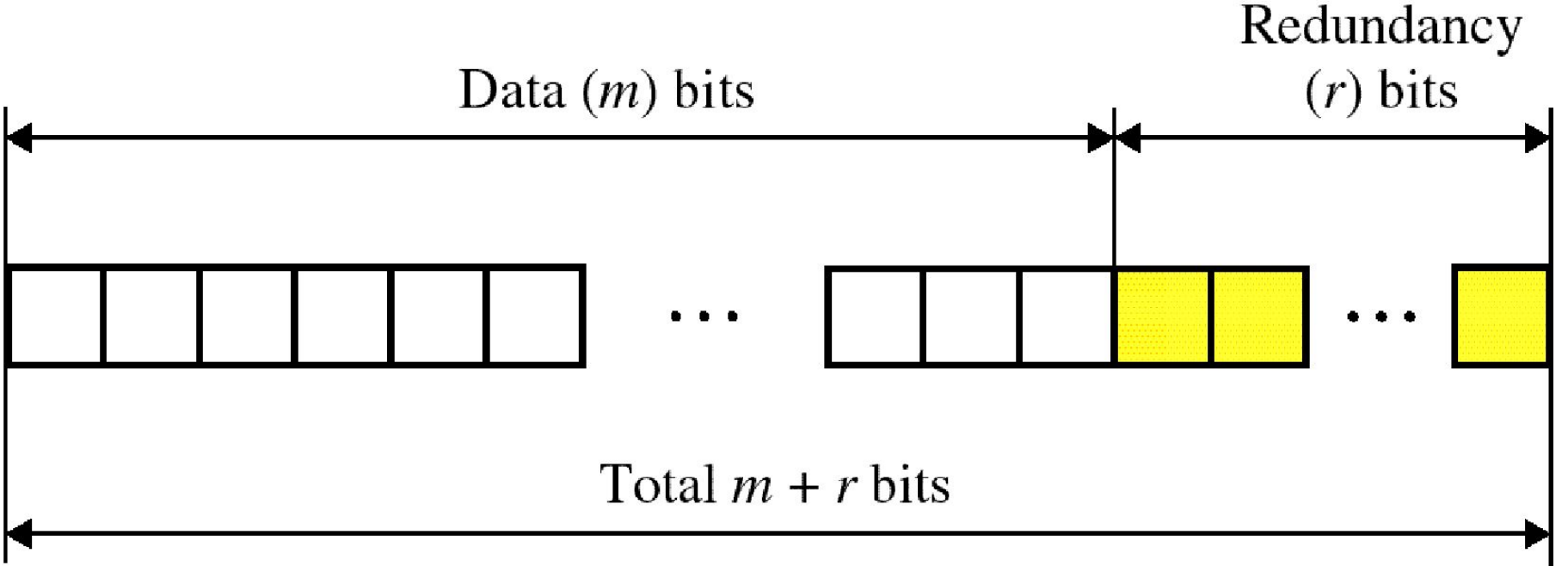
The value of r must satisfy the following relation:

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

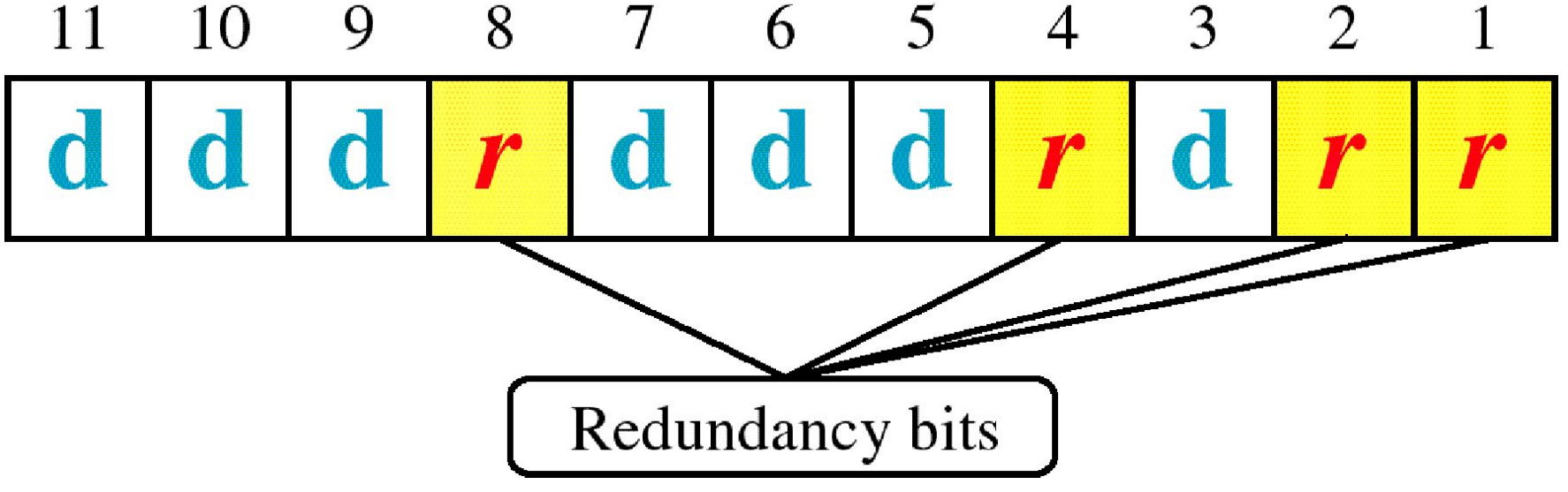
Number of Redundant Bits

Number of data bits k	Number of redundancy bits r	Total bits $k + r$
1	2	3
2	3	5
3	3	6
4	3	7
5	4	9
6	4	10
7	4	11

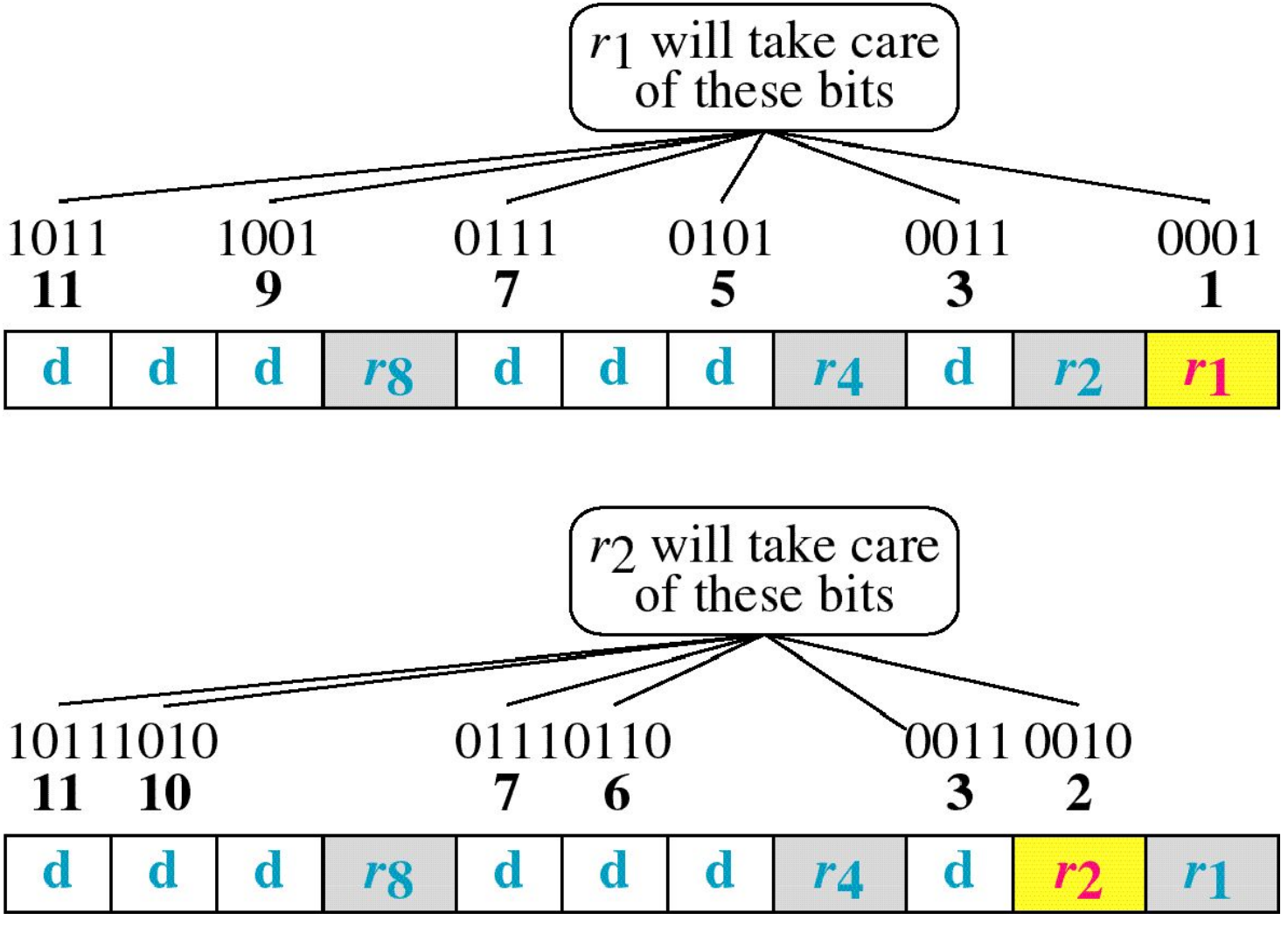
Error Correction



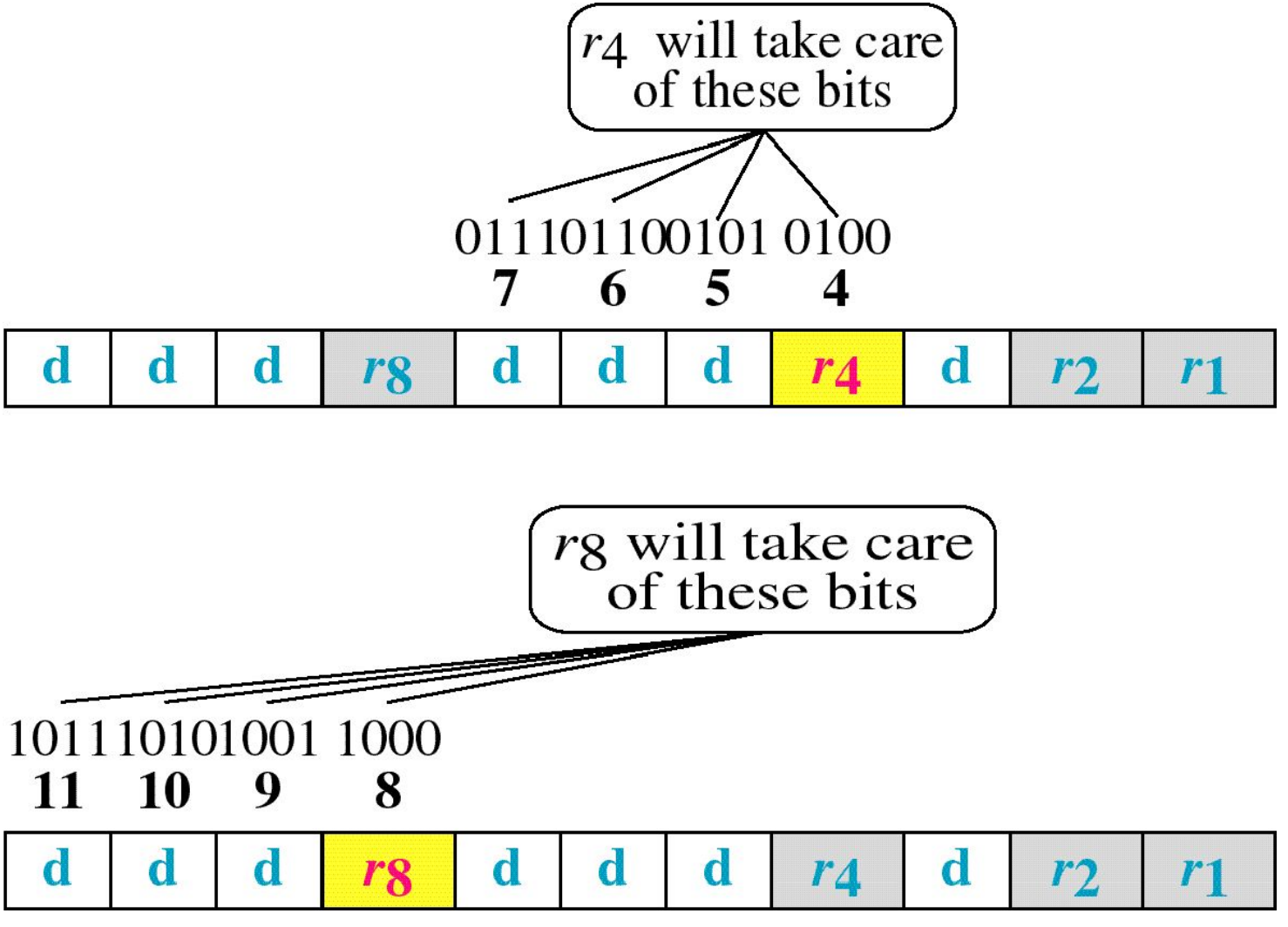
Hamming Code



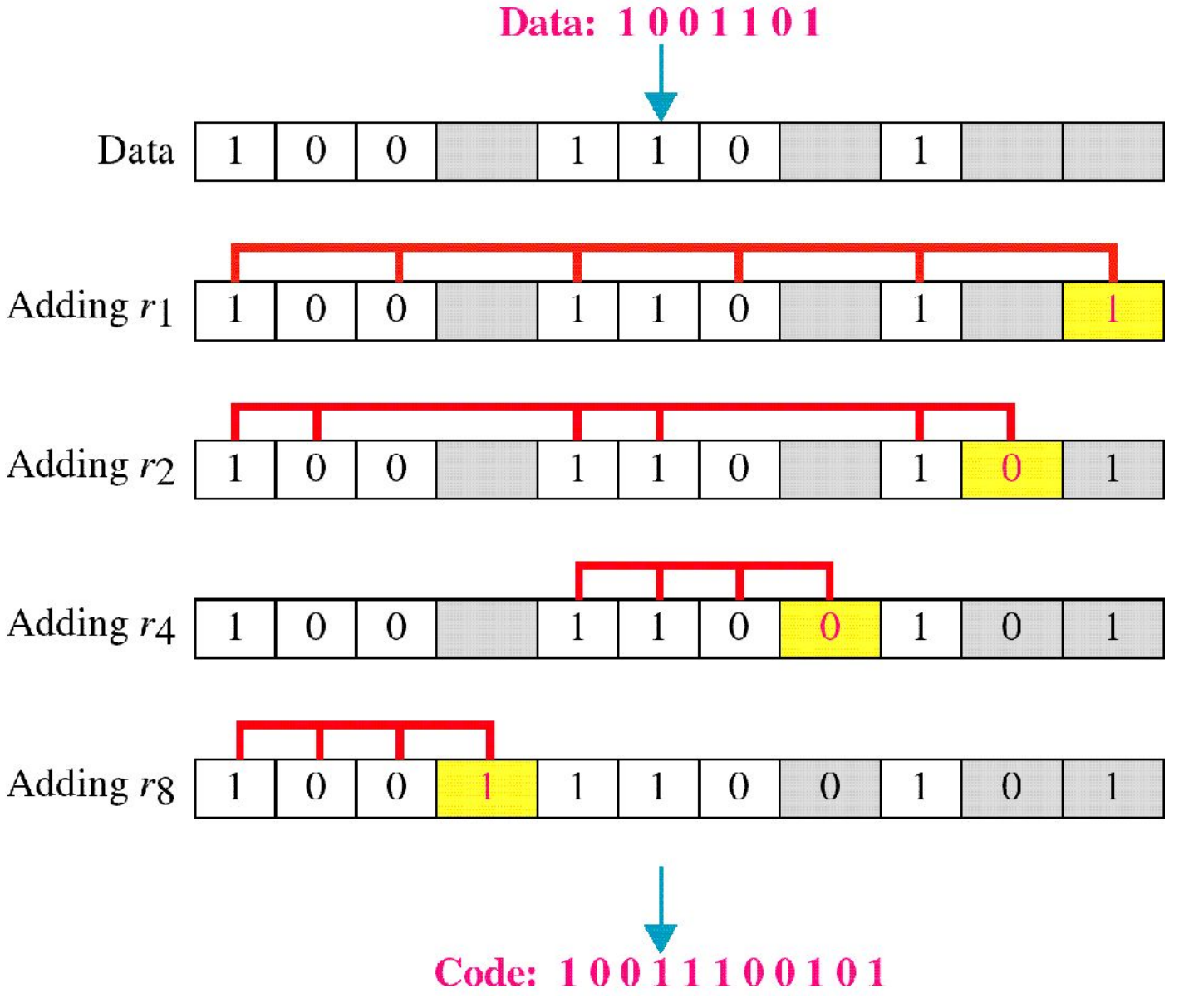
Hamming Code



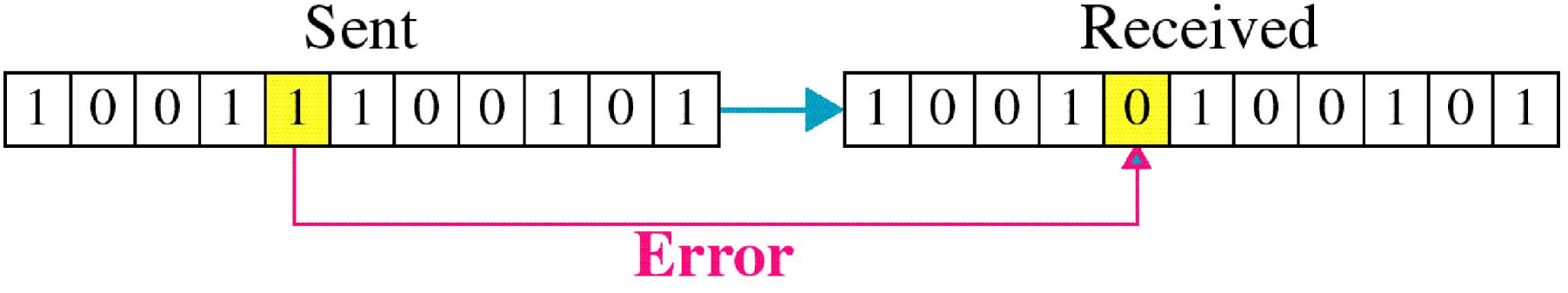
Hamming Code



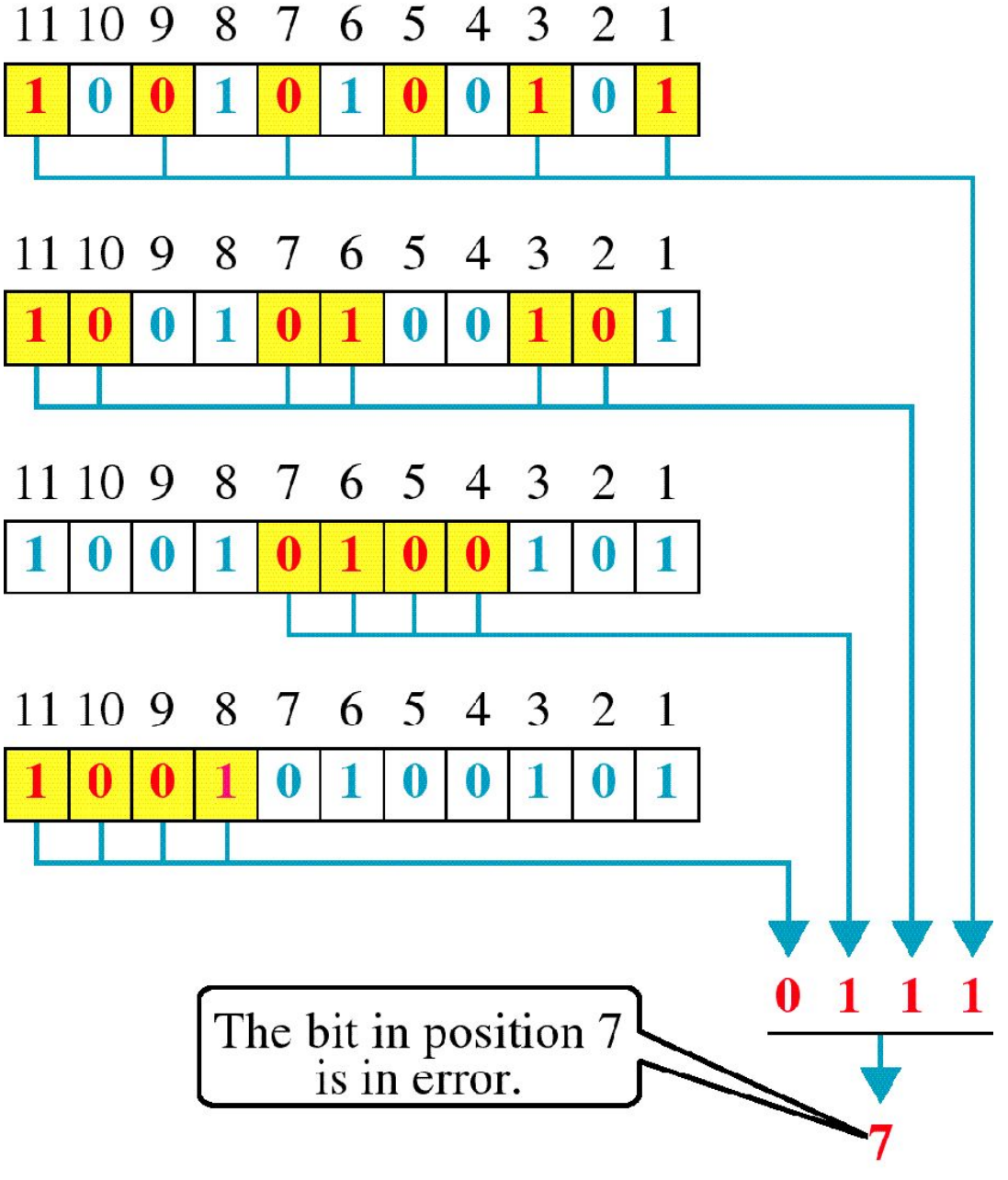
Example of Hamming Code



Single-bit error



Error Detection



Flow and Error Control

Flow Control

- Flow control coordinates the amount of data that can be sent before receiving acknowledgement
- It is one of the most important functions of data link layer.
- Flow control is a set of procedures that tells the sender how much data it can transmit before it must wait for an acknowledgement from the receiver.
- Receiver has a limited speed at which it can process incoming data and a limited amount of memory in which to store incoming data.
- Receiver must inform the sender before the limits are reached and request that the transmitter to send fewer frames or stop temporarily.
- Since the rate of processing is often slower than the rate of transmission, receiver has a block of memory (buffer) for storing incoming data until they are processed.

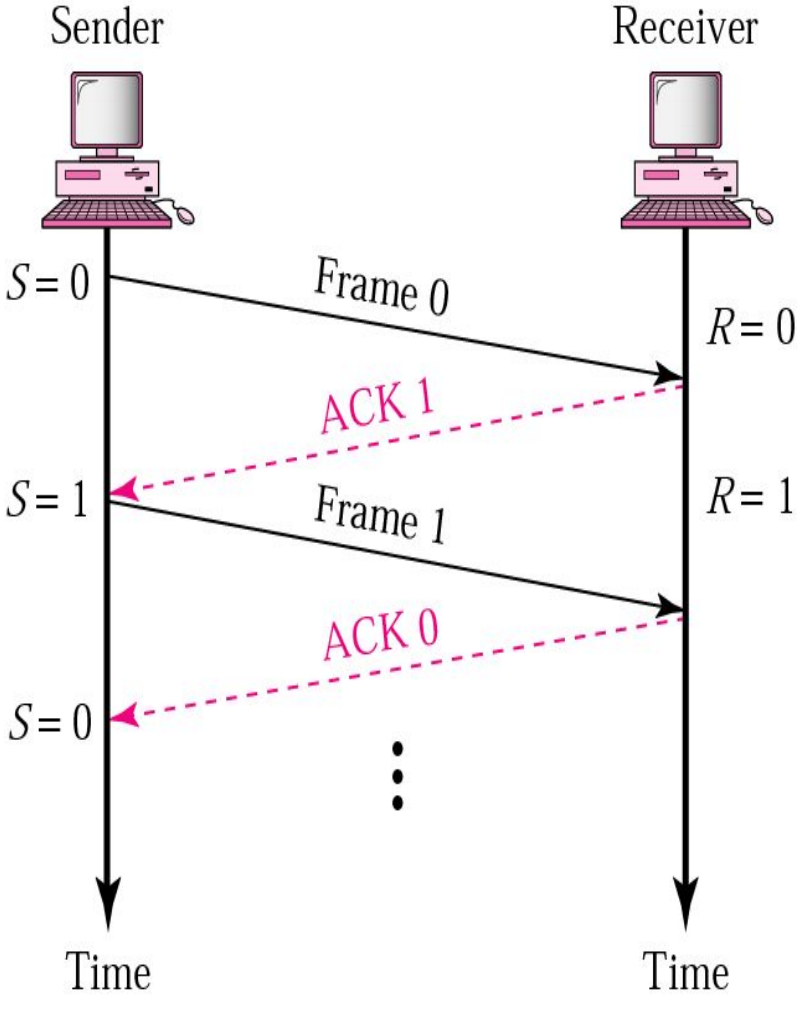
Error Control

- Error control includes both error detection and error correction.
- It allows the receiver to inform the sender if a frame is lost or damaged during transmission and coordinates the retransmission of those frames by the sender.
- Error control in the data link layer is based on automatic repeat request (ARQ). Whenever an error is detected, specified frames are retransmitted.

Error and Flow Control Mechanisms

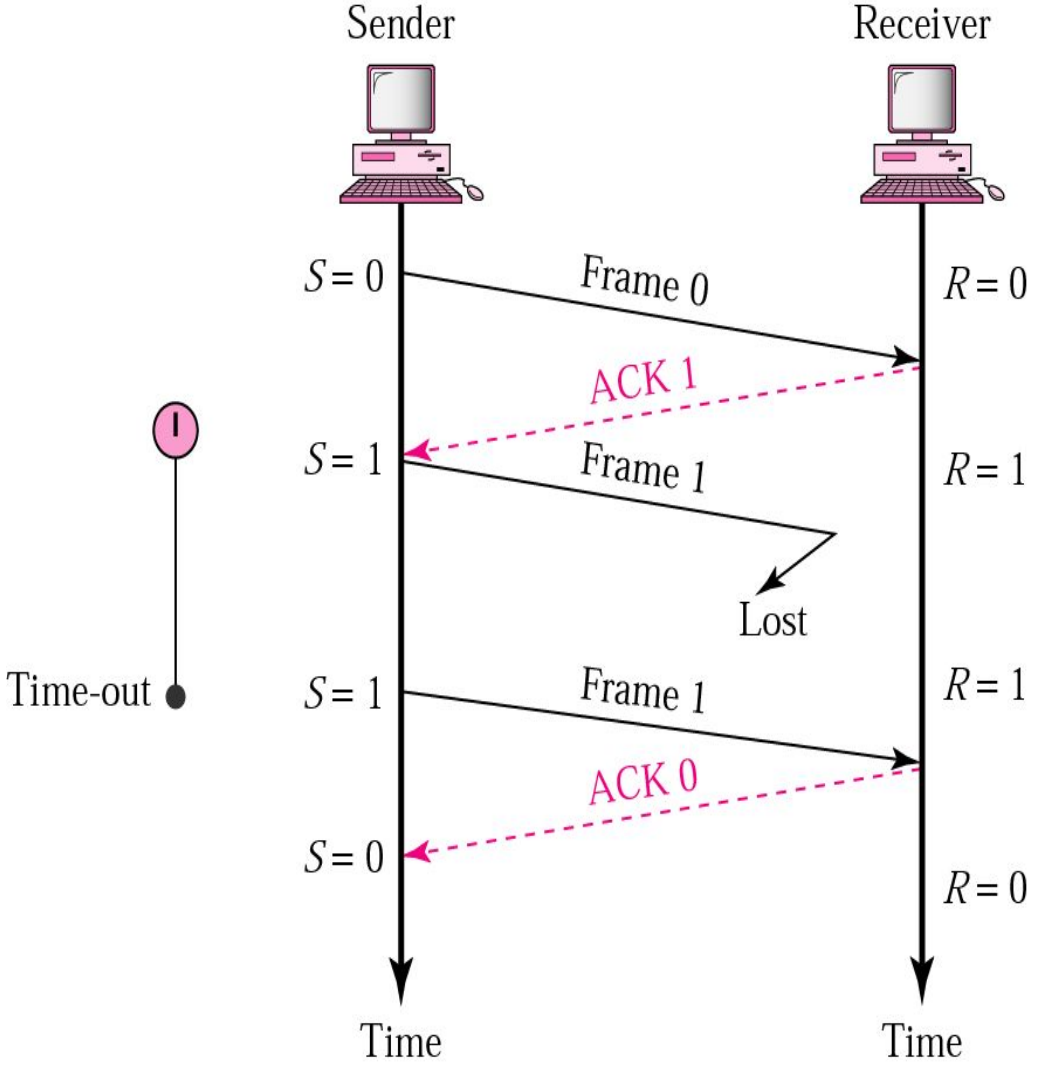
- Stop-and-Wait
- Go-Back-N ARQ
- Selective-Repeat ARQ

Stop-and-Wait



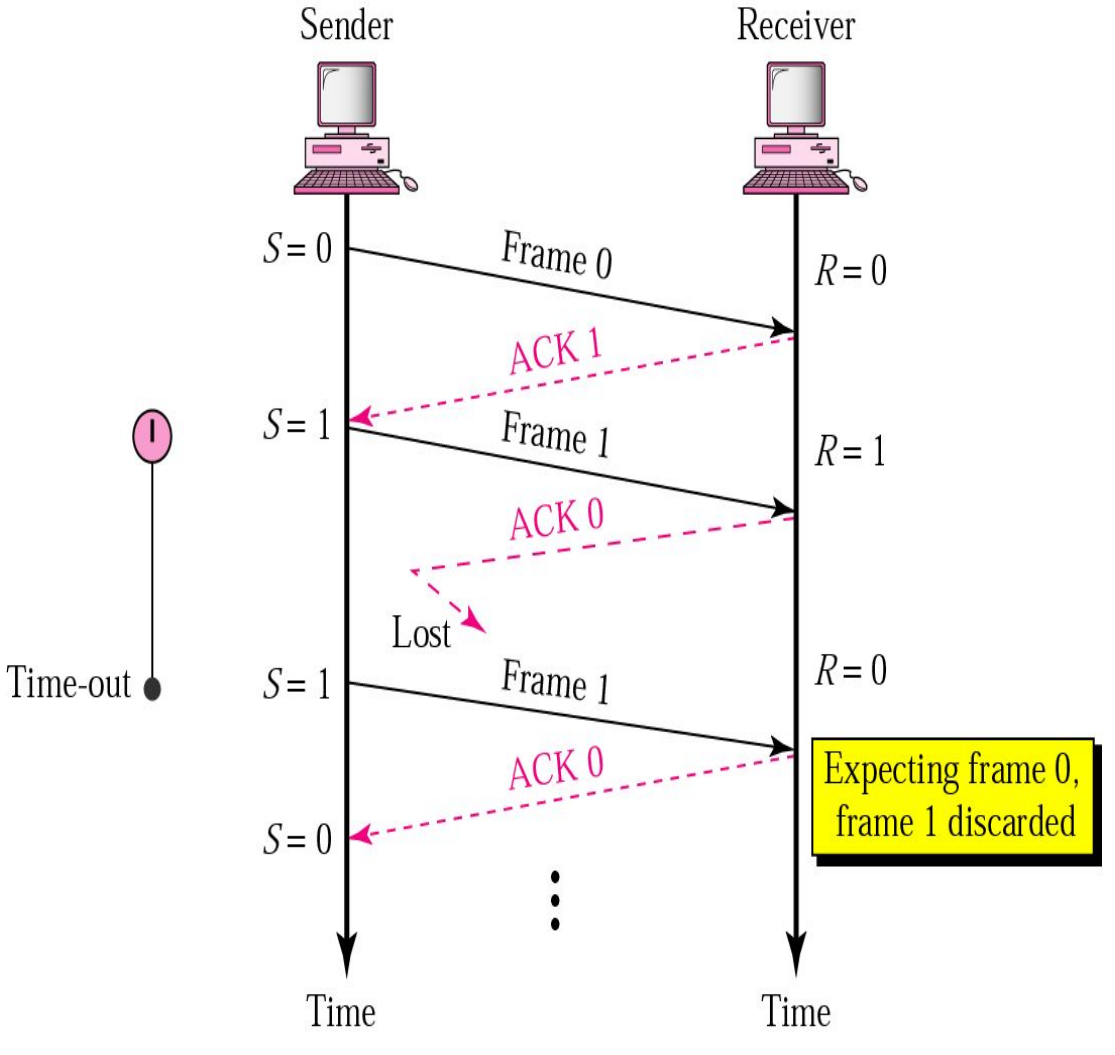
- Sender keeps a copy of the last frame until it receives an acknowledgement.
- For identification, both data frames and acknowledgements (ACK) frames are numbered alternatively 0 and 1.
- Sender has a control variable (S) that holds the number of the recently sent frame. (0 or 1)
- Receiver has a control variable ® that holds the number of the next frame expected (0 or 1).
- Sender starts a timer when it sends a frame. If an ACK is not received within a allocated time period, the sender assumes that the frame was lost or damaged and resends it
- Receiver send only positive ACK if the frame is intact.
- ACK number always defines the number of the next expected frame

Stop-and-Wait ARQ, lost ACK frame



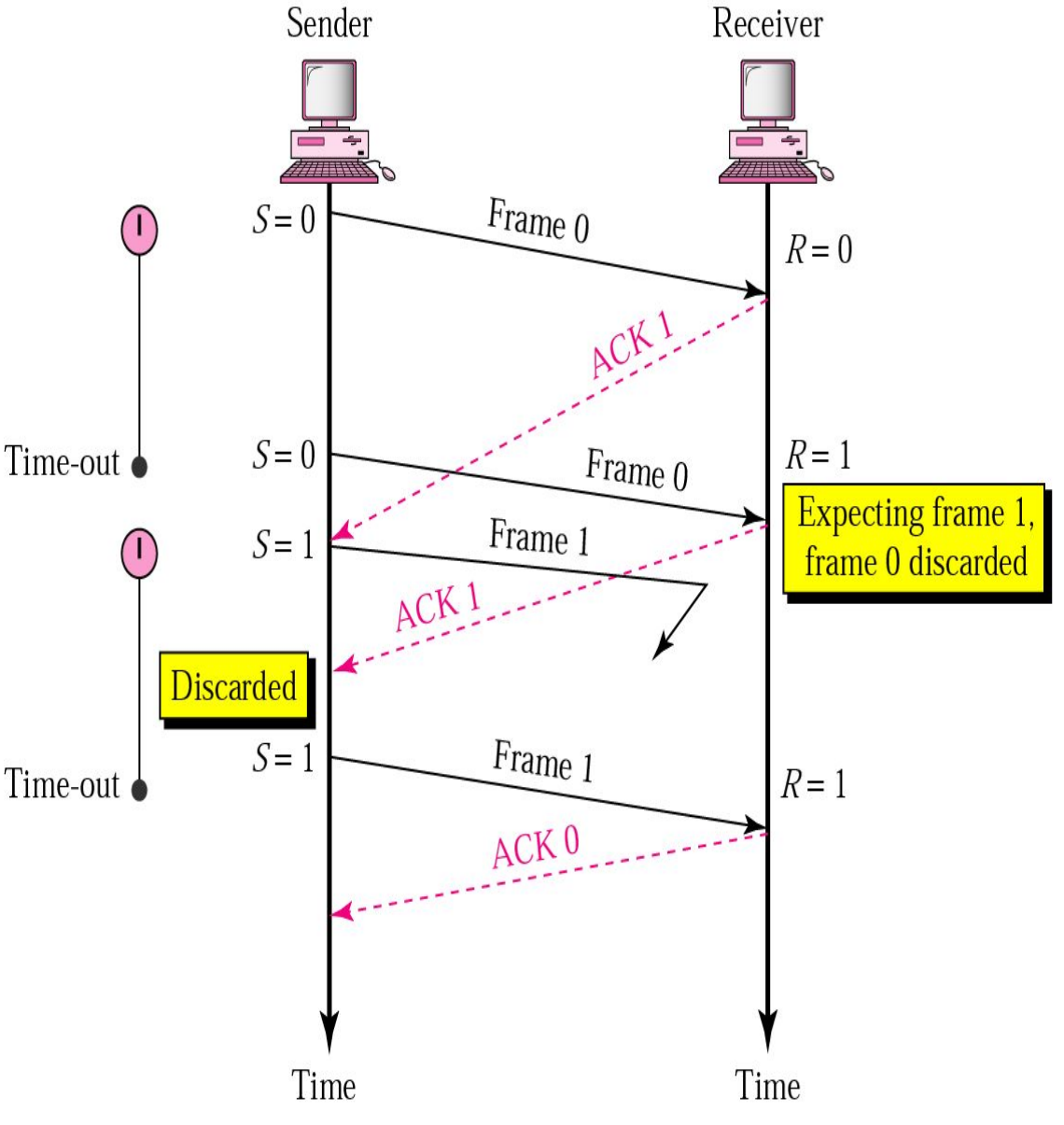
- When a receiver receives a damaged frame, it discards it and keeps its value of R.
- After the timer at the sender expires, another copy of frame 1 is sent.

Stop-and-Wait, lost ACK frame



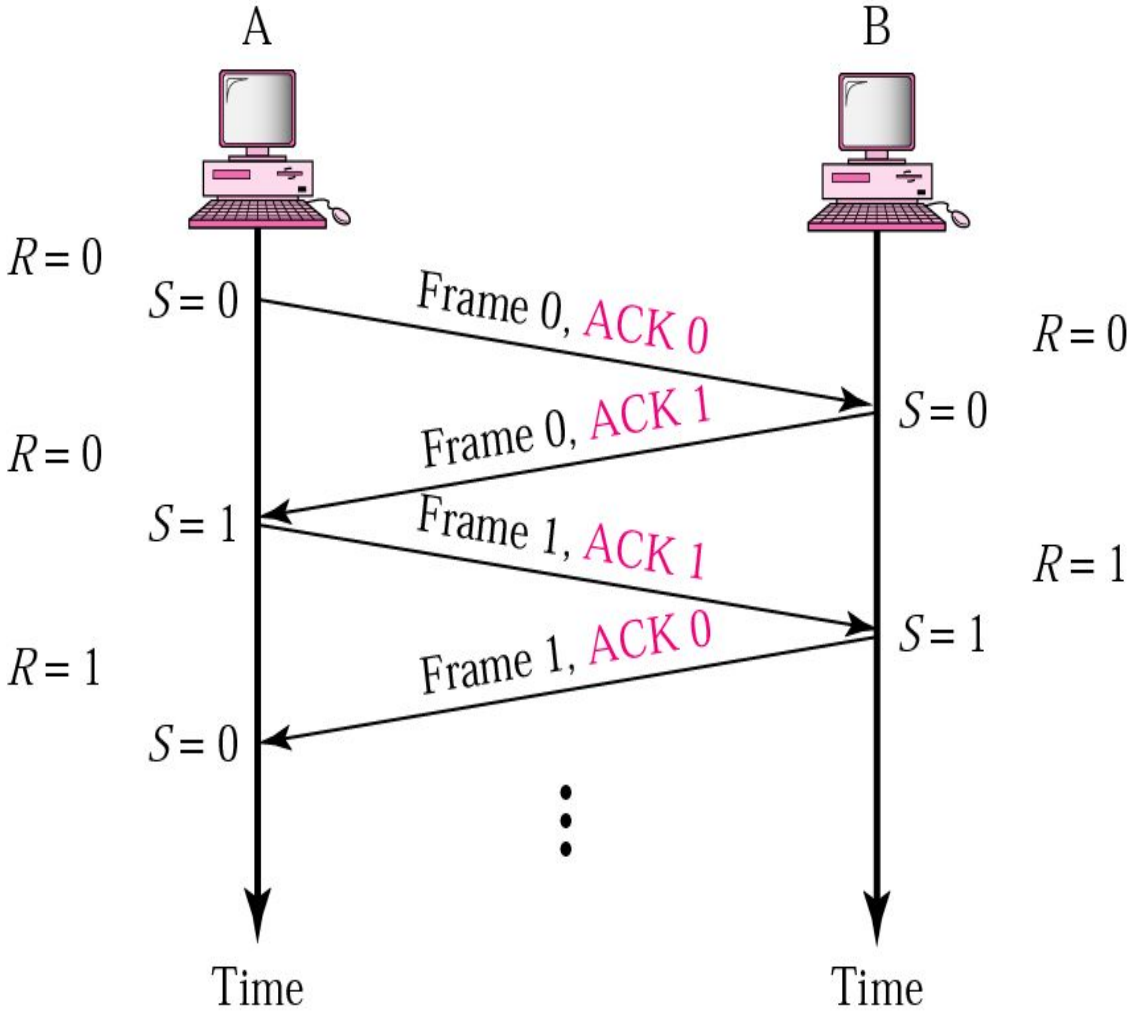
- If the sender receives a damaged ACK, it discards it.
- When the timer of the sender expires, the sender retransmits frame 1.
- Receiver has already received frame 1 and expecting to receive frame 0 (R=0). Therefore it discards the second copy of frame 1.

Stop-and-Wait, delayed ACK frame



- The ACK can be delayed at the receiver or due to some problem
- It is received after the timer for frame 0 has expired.
- Sender retransmitted a copy of frame 0. However, R =1 means receiver expects to see frame 1. Receiver discards the duplicate frame 0.
- Sender receives 2 ACKs, it discards the second ACK.

Piggybacking



- A method to combine a data frame with ACK.
- Station A and B both have data to send.
- Instead of sending separately, station A sends a data frame that includes an ACK.
- Station B does the same thing.
- Piggybacking saves bandwidth.

Disadvantage of Stop-and-Wait

- In stop-and-wait, at any point in time, there is only one frame that is sent and waiting to be acknowledged.
- This is not a good use of transmission medium.
- To improve efficiency, multiple frames should be in transition while waiting for ACK.
- Two protocol use the above concept,
  - **Go-Back-N ARQ**
  - **Selective Repeat ARQ**

Go-Back-N ARQ

- We can send up to W frames before worrying about ACKs.
- We keep a copy of these frames until the ACKs arrive.
- This procedure requires additional features to be added to Stop-and-Wait ARQ.

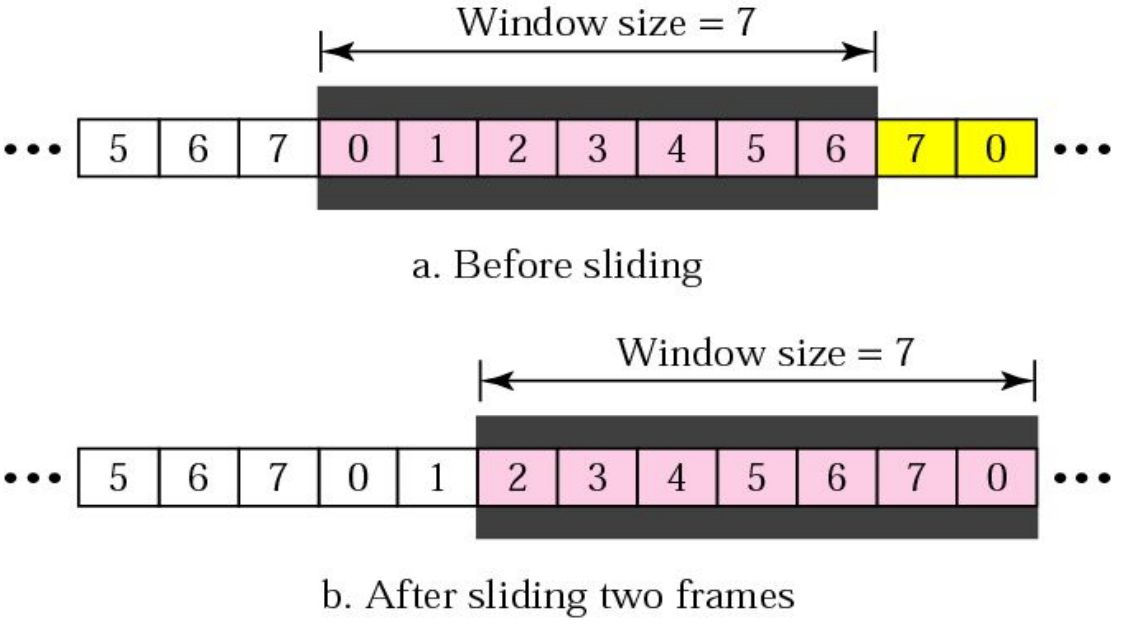
Sequence Numbers

- Frames from a sender are numbered sequentially.
- We need to set a limit since we need to include the sequence number of each frame in the header.
- If the header of the frame allows m bits for sequence number, the sequence numbers range from 0 to $2^m - 1$. for m = 3, sequence numbers are: 1, 2, 3, 4, 5, 6, 7.
- We can repeat the sequence number.
- Sequence numbers are:

  0, 1, 2, 3, 4, 5, 6, 7, 0, 1, 2, 3, 4, 5, 6, 7, 0, 1, …

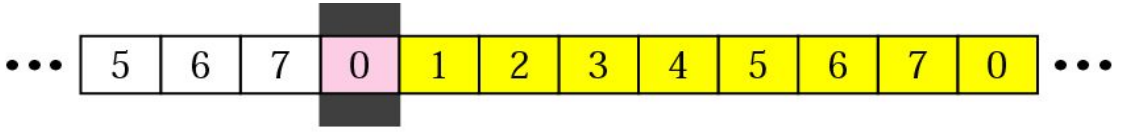
Sender Sliding Window

- At the sending site, to hold the outstanding frames until they are acknowledged, we use the concept of a window.
- The size of the window is at most 2^m -1 where m is the number of bits for the sequence number.
- Size of the window can be variable, e.g. TCP.
- The window slides to include new unsent frames when the correct ACKs are received

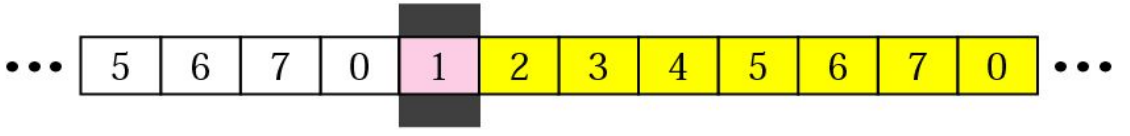


Receiver Sliding Window

- Size of the window at the receiving site is always 1 in this protocol.
- Receiver is always looking for a specific frame to arrive in a specific order.
- Any frame arriving out of order is discarded and needs to be resent.
- Receiver window slides as shown in fig. Receiver is waiting for frame 0 in part a.



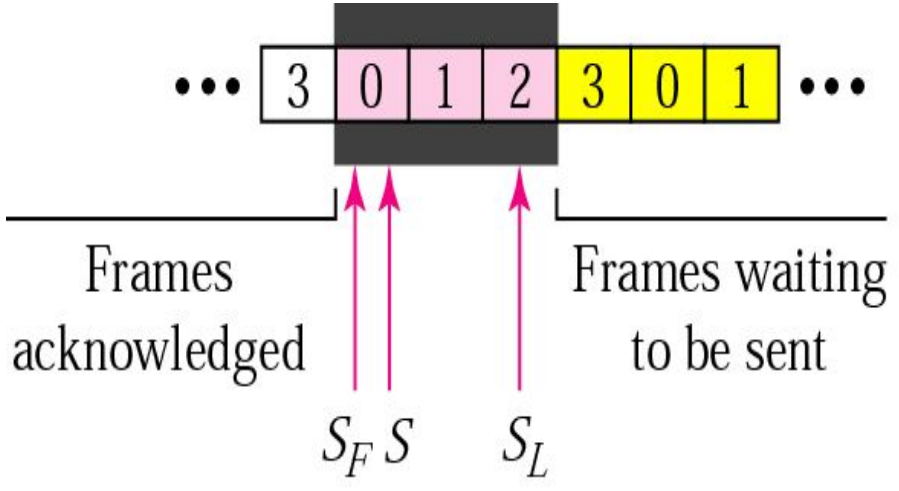
a. Before sliding



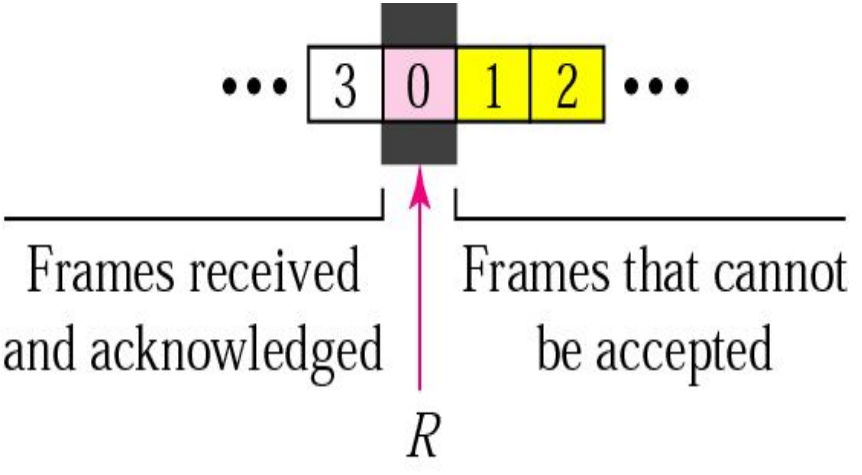
b. After sliding

Control Variables

- Sender has 3 variables: S, S_F, and S_L
- S holds the sequence number of recently sent frame
- S_F holds the sequence number of the first frame
- S_L holds the sequence number of the last frame
- Receiver only has the one variable, R, that holds the sequence number of the frame it expects to receive. If the seq. no. is the same as the value of R, the frame is accepted, otherwise rejected.



a. Sender window



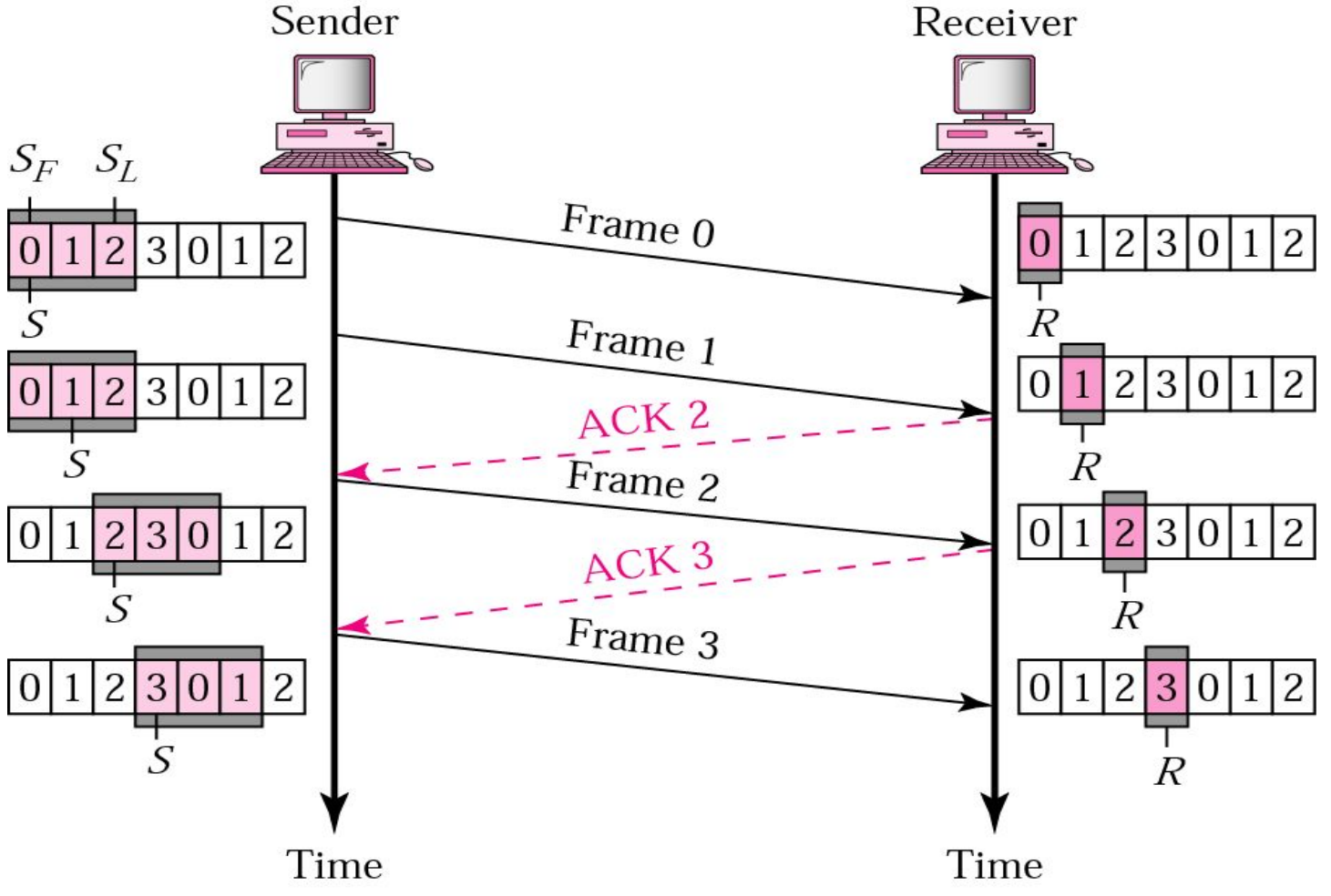
b. Receiver window

Acknowledgement

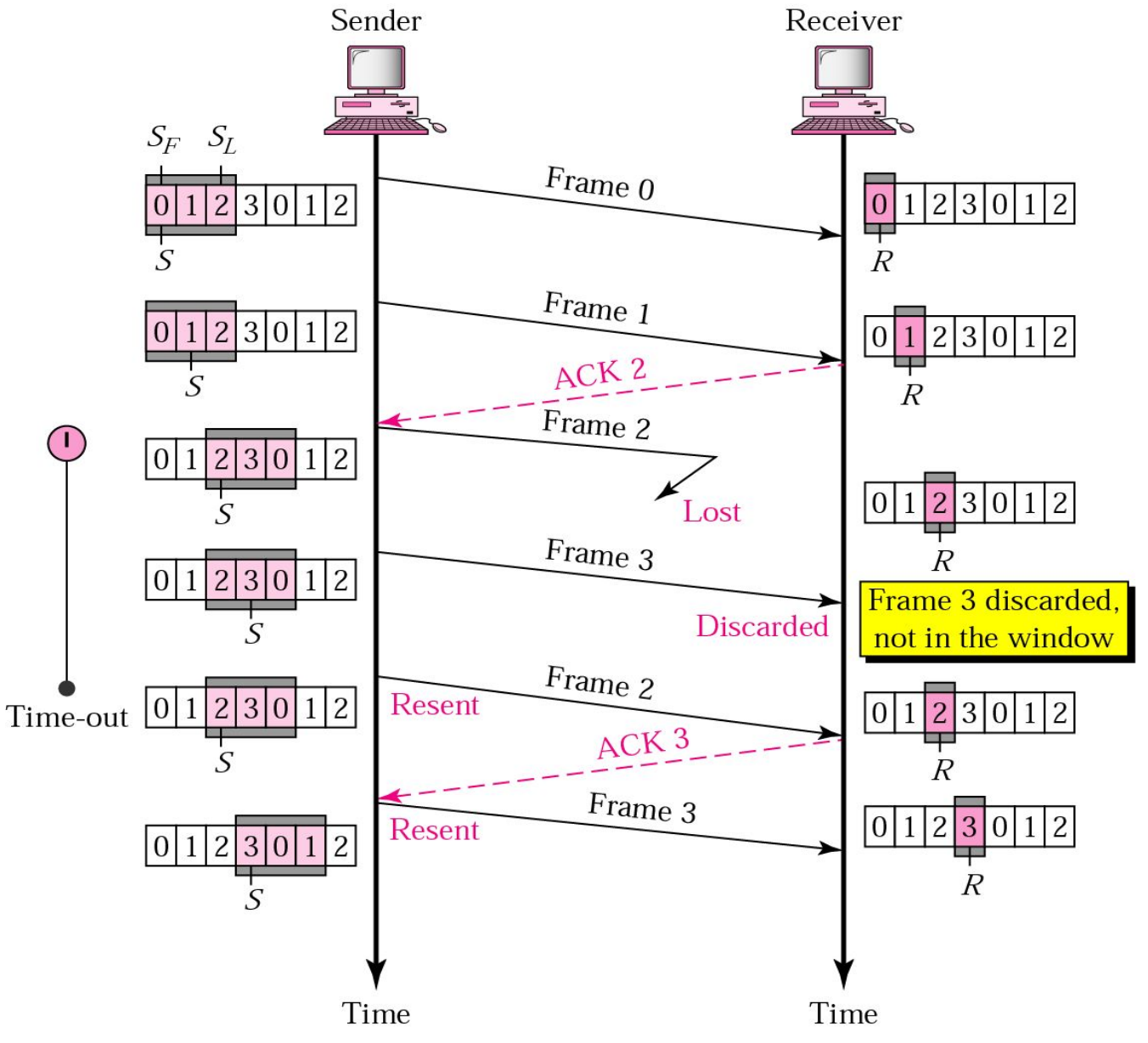
- Receiver sends positive ACK if a frame arrived safe and in order.
- If the frames are damaged/out of order, receiver is silent and discard all subsequent frames until it receives the one it is expecting.
- The silence of the receiver causes the timer of the unacknowledged frame to expire.
- Then the sender resends all frames, beginning with the one with the expired timer.
- For example, suppose the sender has sent frame 6, but the timer for frame 3 expires (i.e. frame 3 has not been acknowledged), then the sender goes back and sends frames 3, 4, 5, 6 again. Thus it is called Go-Back-N-ARQ
- The receiver does not have to acknowledge each frame received, it can send one cumulative ACK for several frames.

Go-Back-N ARQ, normal operation

- The sender keeps track of the outstanding frames and updates the variables and windows as the ACKs arrive.



Go-Back-N ARQ, lost frame



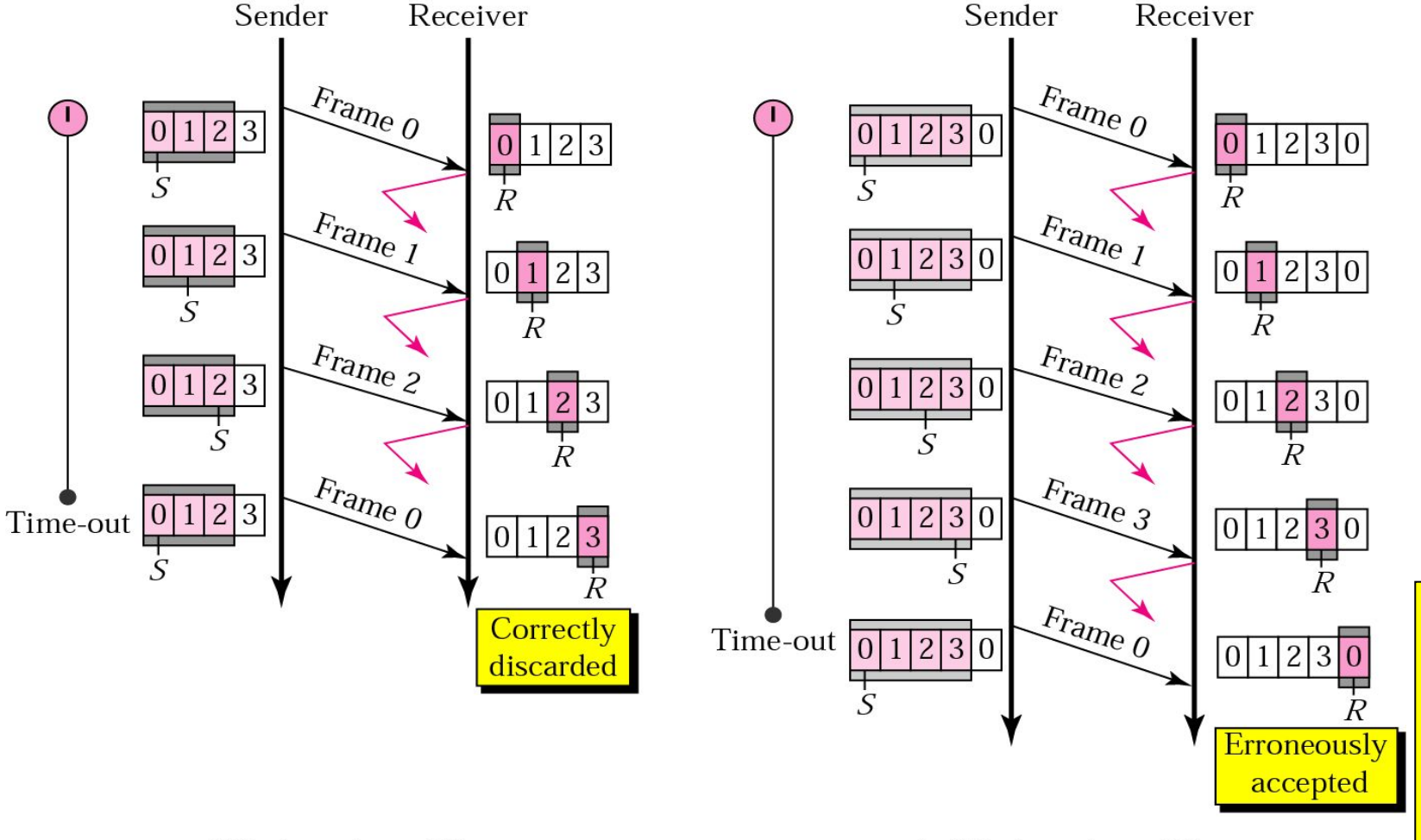
- Frame 2 is lost
- When the receiver receives frame 3, it discards frame 3 as it is expecting frame 2 (according to window).
- After the timer for frame 2 expires at the sender site, the sender sends frame 2 and 3. (go back to 2)

Go-Back-N ARQ, damaged/lost/delayed ACK

- If an ACK is damaged/lost, we can have two situations:
- If the next ACK arrives before the expiration of any timer, there is no need for retransmission of frames because ACKs are cumulative in this protocol.
- If ACK1, ACK2, and ACk3 are lost, ACK4 covers them if it arrives before the timer expires.
- If ACK4 arrives after time-out, the last frame and all the frames after that are resent.
- Receiver never resends an ACK.
- A delayed ACK also triggers the resending of frames

Go-Back-N ARQ, sender window size

- Size of the sender window must be less than 2^m. Size of the receiver is always 1. If m = 2, window size = $2^m - 1 = 3$.
- Fig compares a window size of 3 and 4.

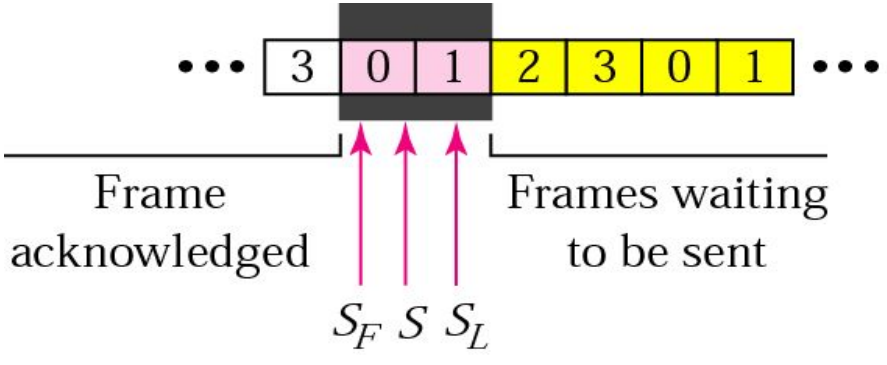


a. Window size < 2^m

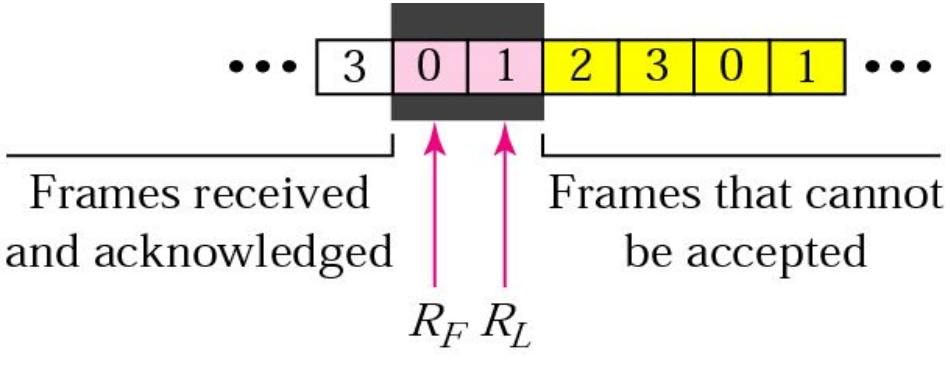
b. Window size = 2^m

Selective Repeat ARQ, sender and receiver windows

- Go-Back-N ARQ simplifies the process at the receiver site. Receiver only keeps track of only one variable, and there is no need to buffer out-of-order frames, they are simply discarded.
- However, Go-Back-N ARQ protocol is inefficient for noisy link. It bandwidth inefficient and slows down the transmission.
- In Selective Repeat ARQ, only the damaged frame is resent. More bandwidth efficient but more complex processing at receiver.
- It defines a negative ACK (NAK) to report the sequence number of a damaged frame before the timer expires.

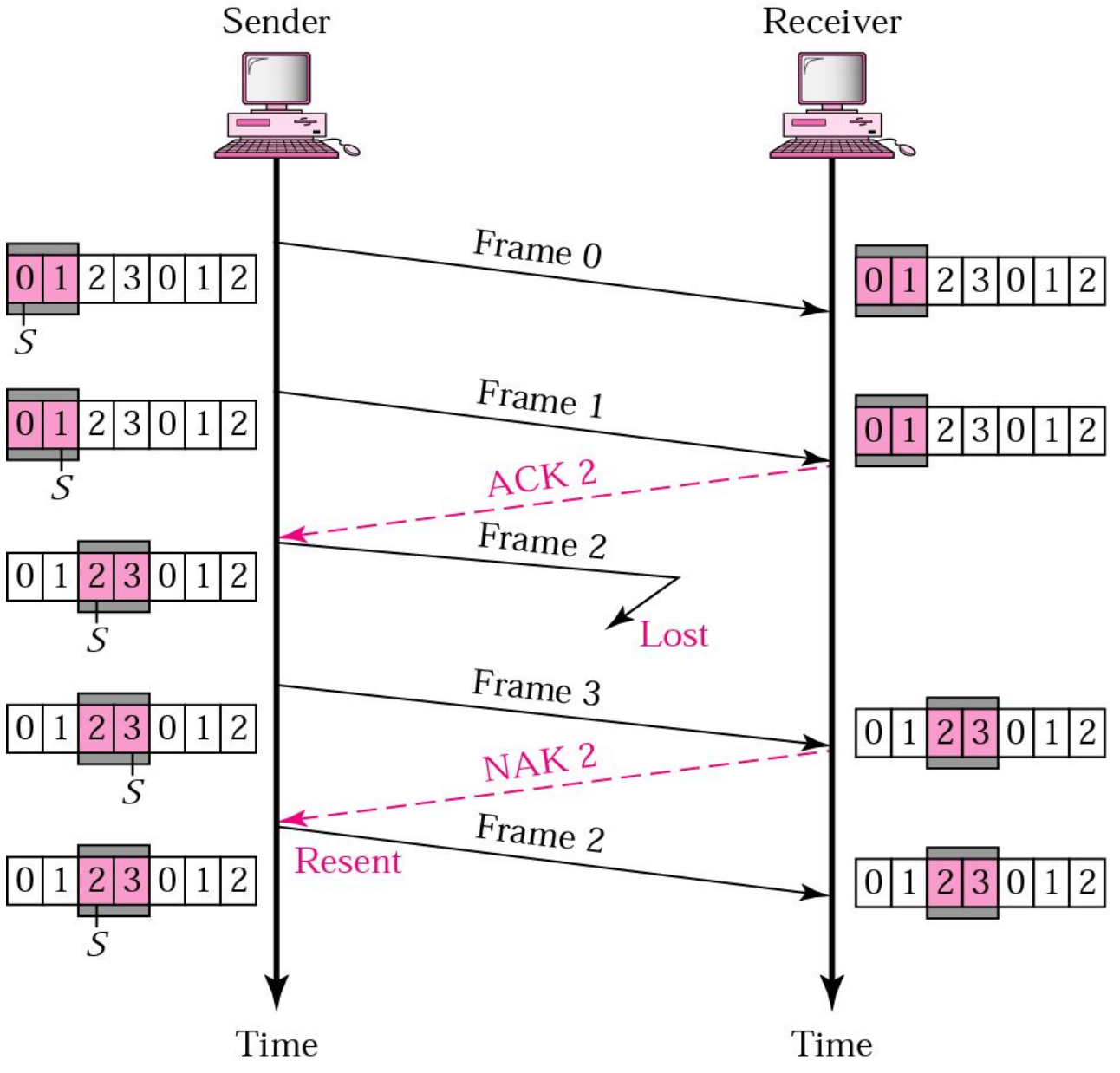


a. Sender window



b. Receiver window

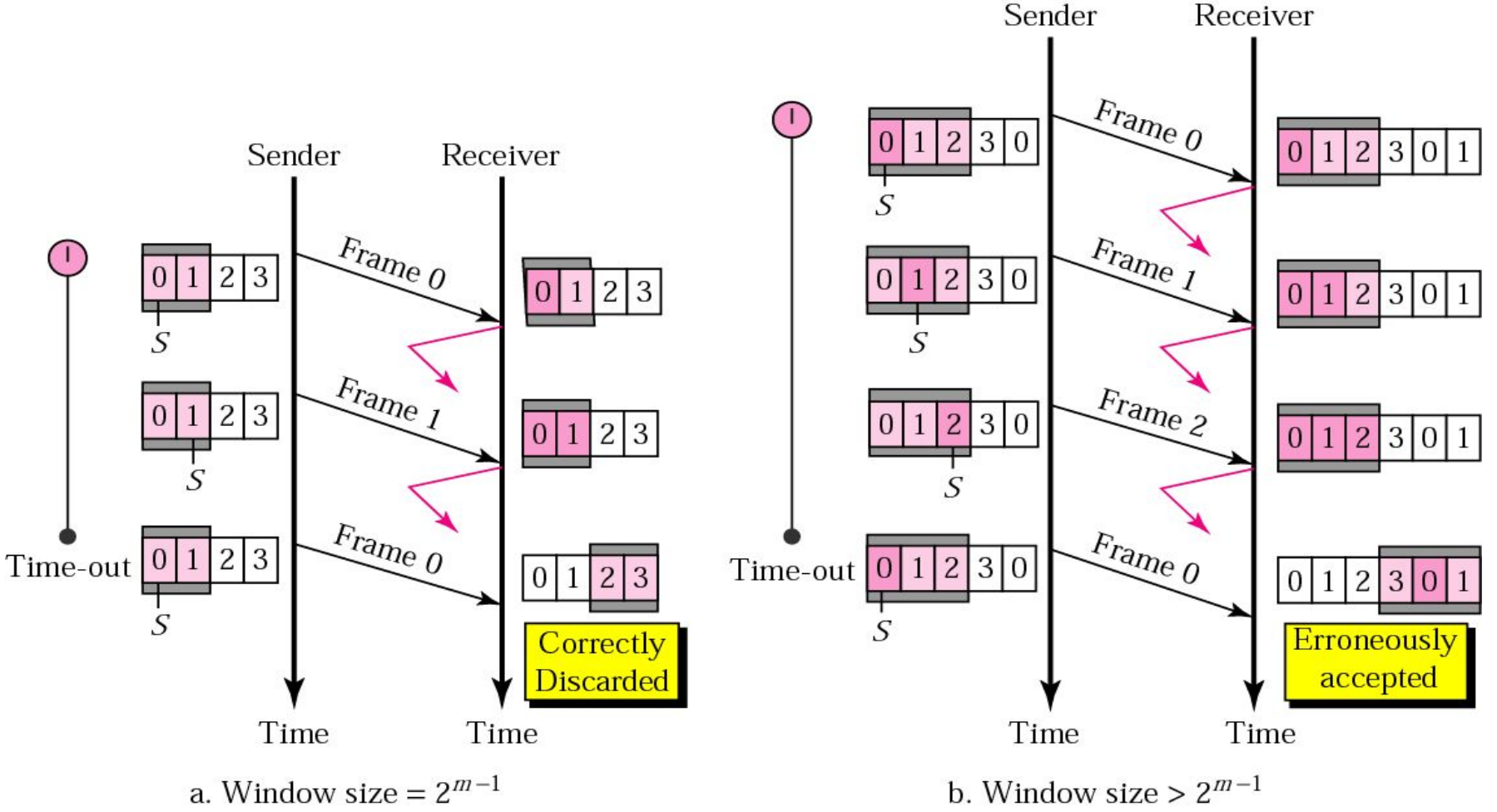
Selective Repeat ARQ, lost frame



- Frames 0 and 1 are accepted when received because they are in the range specified by the receiver window. Same for frame 3.
- Receiver sends a NAK2 to show that frame 2 has not been received and then sender resends only frame 2 and it is accepted as it is in the range of the window.

Selective Repeat ARQ, sender window size

- Size of the sender and receiver windows must be at most one-half of 2^m. If m = 2, window size should be $2^m/2 = 2$. Fig compares a window size of 2 with a window size of 3. Window size is 3 and all ACKs are lost, sender sends duplicate of frame 0, window of the receiver expect to receive frame 0 (part of the window), so accepts frame 0, as the 1st frame of the next cycle – an **error**.



- Random Access
- Multiple access protocols
- Pure ALOHA, Slotted ALOHA
- CSMA/CD
- CDMA/CA.

Random Access MAC Protocols

- When node has packet to send
  - Transmit at full channel data rate
  - No *a priori* coordination among nodes
- Two or more transmitting nodes ⇒ collision
  - Data lost
- Random access MAC protocol specifies:
  - How to detect collisions
  - How to recover from collisions
- Examples
  - ALOHA and Slotted ALOHA
  - CSMA, CSMA/CD, CSMA/CA (wireless)

Key Ideas of Random Access

- Carrier sense
  - *Listen before speaking, and don't interrupt*
  - Checking if someone else is already sending data
  - … and waiting till the other node is done
- Collision detection
  - *If someone else starts talking at the same time, stop*
  - Realizing when two nodes are transmitting at once
  - …by detecting that the data on the wire is garbled
- Randomness
  - *Don't start talking again right away*
  - Waiting for a random time before trying again

Carrier Sensing Multiple Access (CSMA)

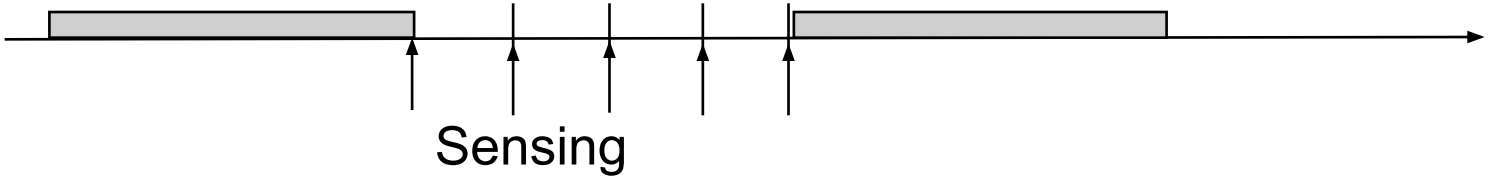
- A station senses the channel before it starts transmission
  - If busy, either wait or schedule backoff (different options)
  - If idle, start transmission
  - Vulnerable period is reduced to t_{prop} (due to *channel capture* effect)
  - Collisions in ALOHA or Slotted ALOHA involve 2 or 1 frame transmission times X
  - If t_{prop} >X (or if a>1), no gain compared to ALOHA or slotted ALOHA

Station A begins transmission at $t = 0$

Station A captures channel at $t = t_{prop}$

CSMA Options

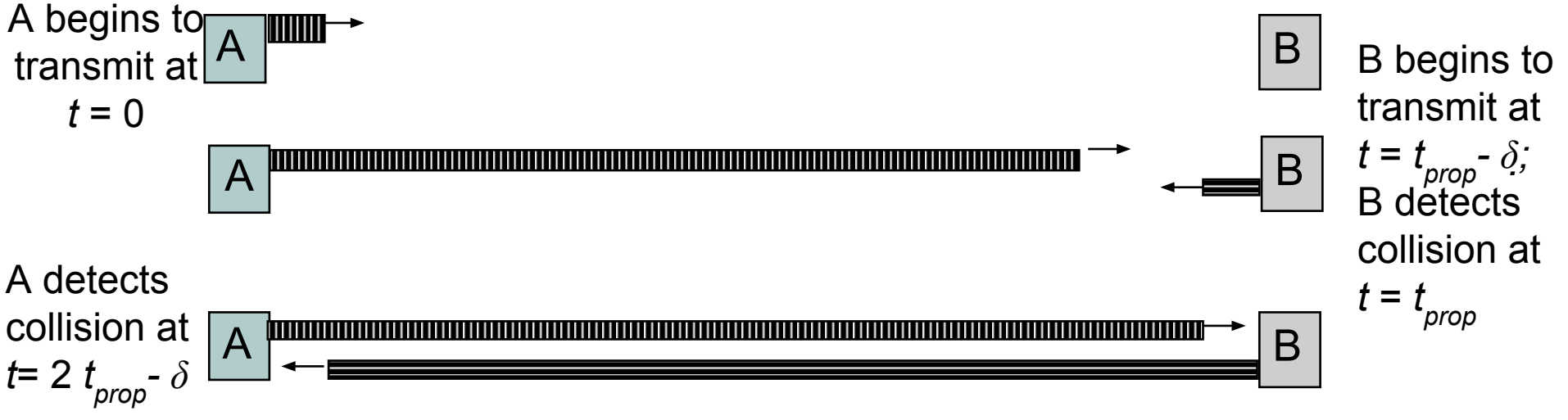
- Transmitter behavior when busy channel is sensed
  - 1-persistent CSMA (most greedy)
    - Start transmission as soon as the channel becomes idle
    - Low delay and high collision rates
  - Non-persistent CSMA (least greedy)
    - Wait a backoff period, then sense carrier again
    - High delay and low collision rates
  - p-persistent CSMA (adjustable greedy)
    - Wait till channel becomes idle, transmit with prob. p; or wait another t_{prop} & re-sense with probability 1-*p*
    - Delay and collisions rates can be balanced



CSMA with Collision Detection (CSMA/CD)

- Monitor for collisions & abort transmission
  - Stations with frames to send, first do carrier sensing
  - After beginning transmissions, stations continue listening to the medium to detect collisions
  - If collisions detected, all stations involved stop transmission, reschedule random backoff times, and try again at scheduled times
- In CSMA collisions result in wastage of X seconds spent transmitting an entire frame
- CSMA-CD reduces wastage to time to detect collision and abort transmission

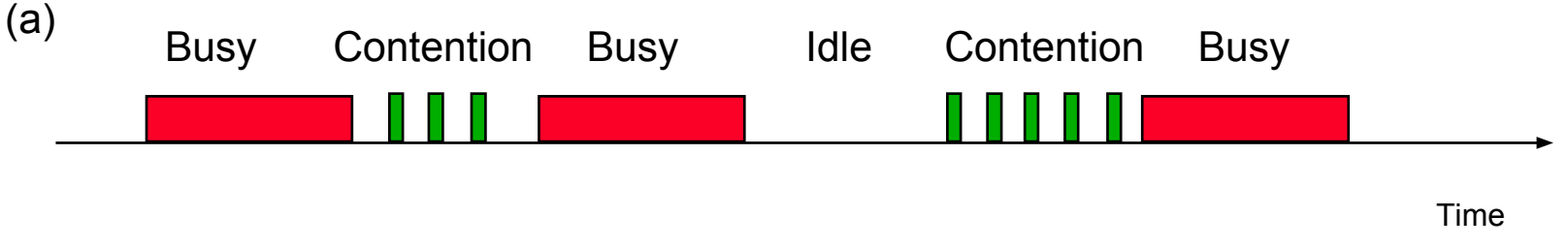
CSMA/CD reaction time



It takes 2 t_{prop} to find out if channel has been captured

CSMA-CD Model

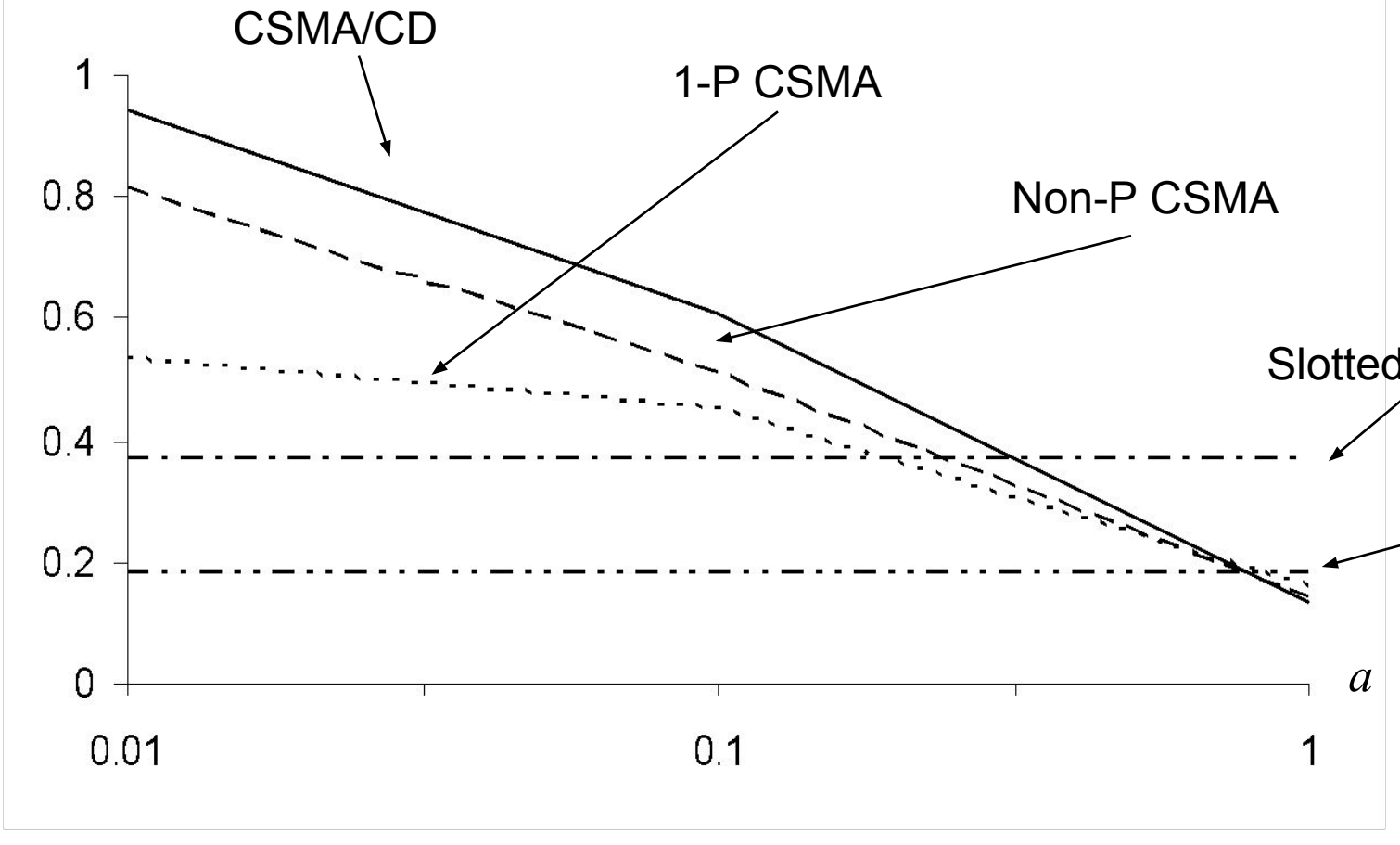
- Assumptions
  - Collisions can be detected and resolved in $2t_{prop}$
  - Time slotted in $2t_{prop}$ slots during contention periods
  - Assume n busy stations, and each may transmit with probability p in each contention time slot
  - Once the contention period is over (a station successfully occupies the channel), it takes X seconds for a frame to be transmitted
  - It takes t_{prop} before the next contention period starts.



CSMA-CD Application: Ethernet

- First Ethernet LAN standard used CSMA-CD
  - 1-persistent Carrier Sensing
  - R = 10 Mbps
  - t_{prop} = 51.2 microseconds
    - 512 bits = 64 byte slot
    - accommodates 2.5 km + 4 repeaters
  - Truncated Binary Exponential Backoff
    - After the nth collision, select backoff from {0, 1,…, 2^k – 1}, where k=min(n, 10)

Throughput for Random Access MACs



- For small *a:* CSMA-CD has best throughput
- For larger *a:* Aloha & slotted Aloha better throughput

Carrier Sensing and Priority Transmission

- Certain applications require faster response than others, e.g. ACK messages
- Impose different interframe times
  - High priority traffic sense channel for time τ_1
  - Low priority traffic sense channel for time $\tau_2 > \tau_1$
  - High priority traffic, if present, seizes channel first
- This priority mechanism is used in IEEE 802.11 wireless LAN

CSMA (Carrier Sense Multiple Access)

- CSMA: listen before transmit
  - If channel sensed idle: transmit entire frame
  - If channel sensed busy, defer transmission
- Human analogy: don't interrupt others!
- Does this eliminate all collisions?
  - No, because of nonzero propagation delay

CSMA Collisions

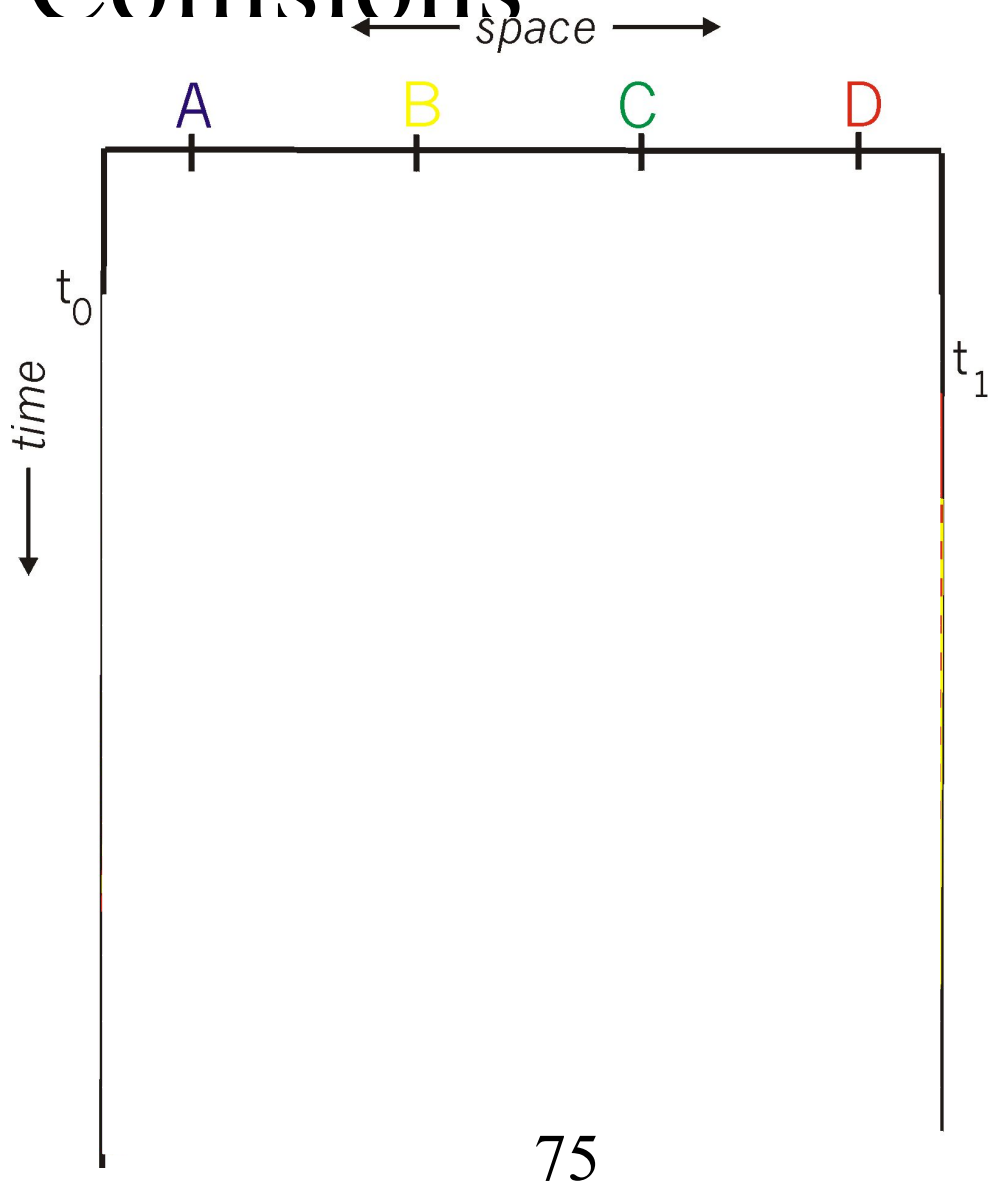
Propagation delay: two nodes may not hear each other's before sending.

Would slots hurt or help?

CSMA reduces but does not eliminate collisions

Biggest remaining problem?

Collisions still take full slot! How do you fix that?



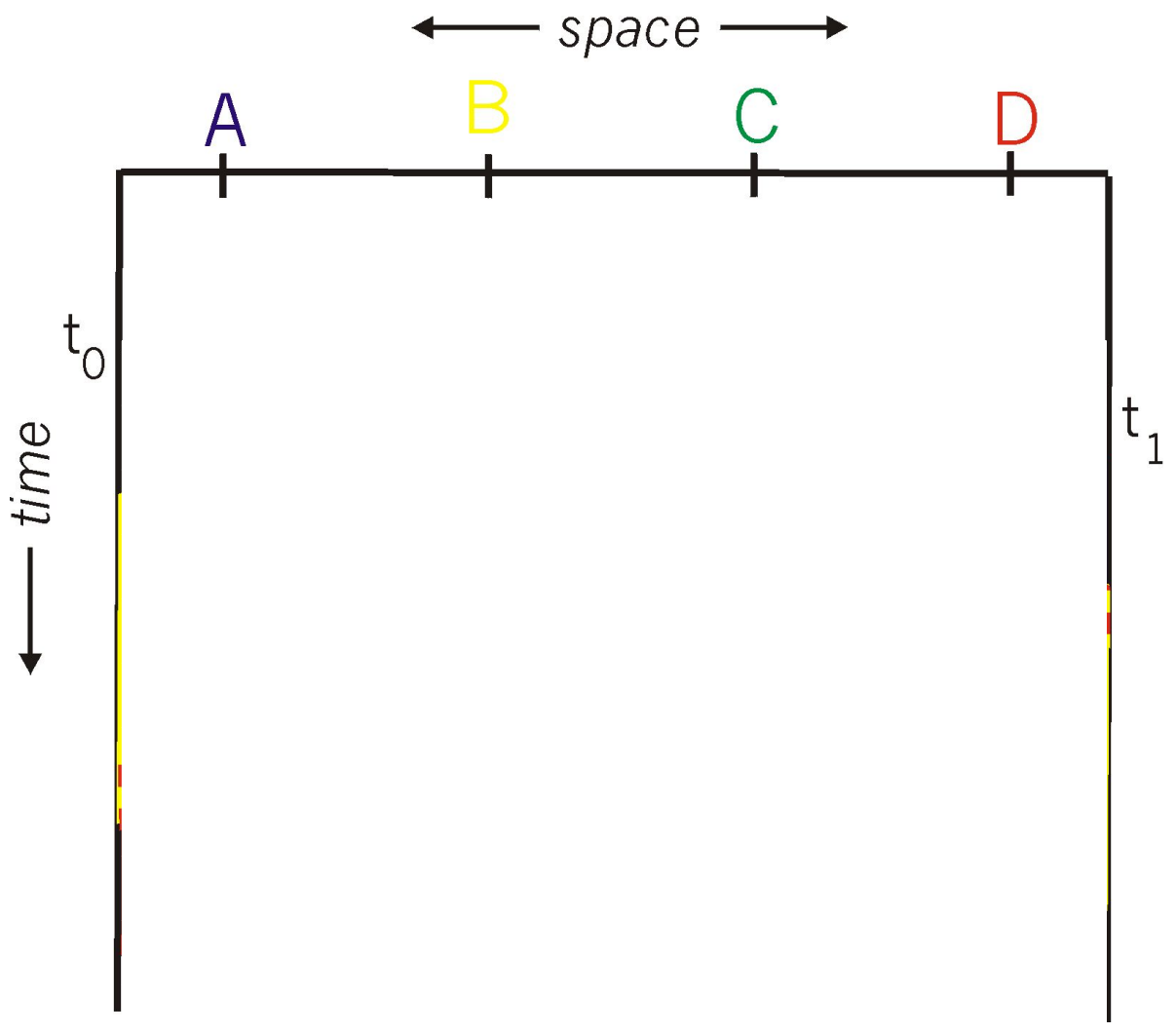
CSMA/CD (Collision Detection)

- CSMA/CD: carrier sensing, deferral as in CSMA
  - **Collisions detected within short time**
  - Colliding transmissions aborted, reducing wastage
- Collision detection easy in wired LANs:
  - Compare transmitted, received signals
- Collision detection difficult in wireless LANs:
  - Reception shut off while transmitting (well, perhaps not)
  - Not perfect broadcast (limited range) so collisions local
  - Leads to use of *collision avoidance* instead (later)

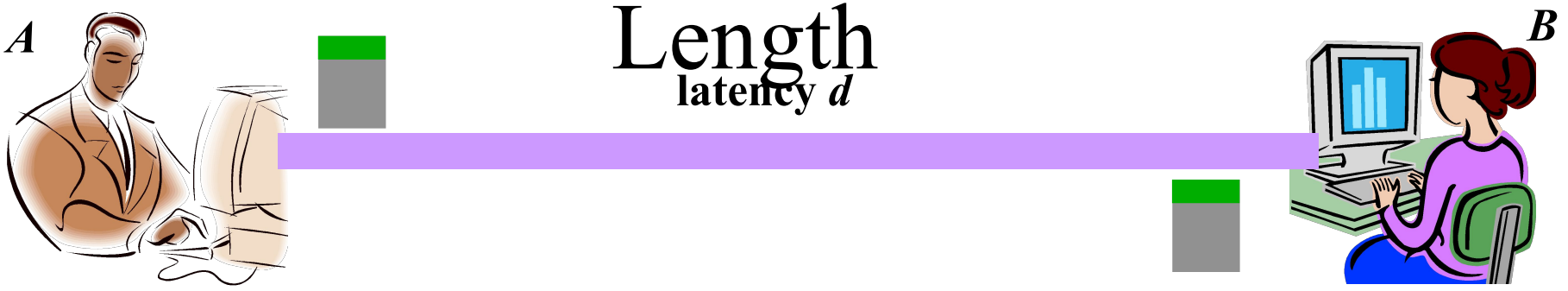
CSMA/CD Collision Detection

B and D can tell that collision occurred.

Note: for this to work, need restrictions on minimum frame size and maximum distance. Why?



Limits on CSMA/CD Network Length



- Latency depends on physical length of link
  - Time to propagate a packet from one end to the other
- Suppose *A* sends a packet at time ***t***
  - And *B* sees an idle line at a time just before ***t***+***d***
  - … so *B* happily starts transmitting a packet
- *B* detects a collision, and sends jamming signal
  - But *A* can't see collision until ***t***+***2d***

Performance of CSMA/CD

- Time wasted in collisions
  - Proportional to distance d
- Time spend transmitting a packet
  - Packet length p divided by bandwidth b
- Rough estimate for efficiency (K some constant)

$$E \sim \frac{\frac{p}{b}}{\frac{p}{b} + Kd}$$

- Note:
  - For large packets, small distances, E ~ 1
  - As bandwidth increases, E decreases
  - That is why high-speed LANs are all switched

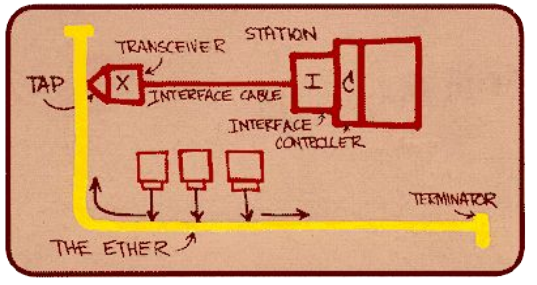
Benefits of Ethernet

- Easy to administer and maintain
- Inexpensive
- Increasingly higher speed
- Evolvable!

Evolution of Ethernet

- Changed everything except the frame format
  - From single coaxial cable to hub-based star
  - From shared media to switches
  - From electrical signaling to optical
- Lesson #1
  - The right interface can accommodate many changes
  - Implementation is hidden behind interface
- Lesson #2
  - Really hard to displace the dominant technology
  - Slight performance improvements are not enough

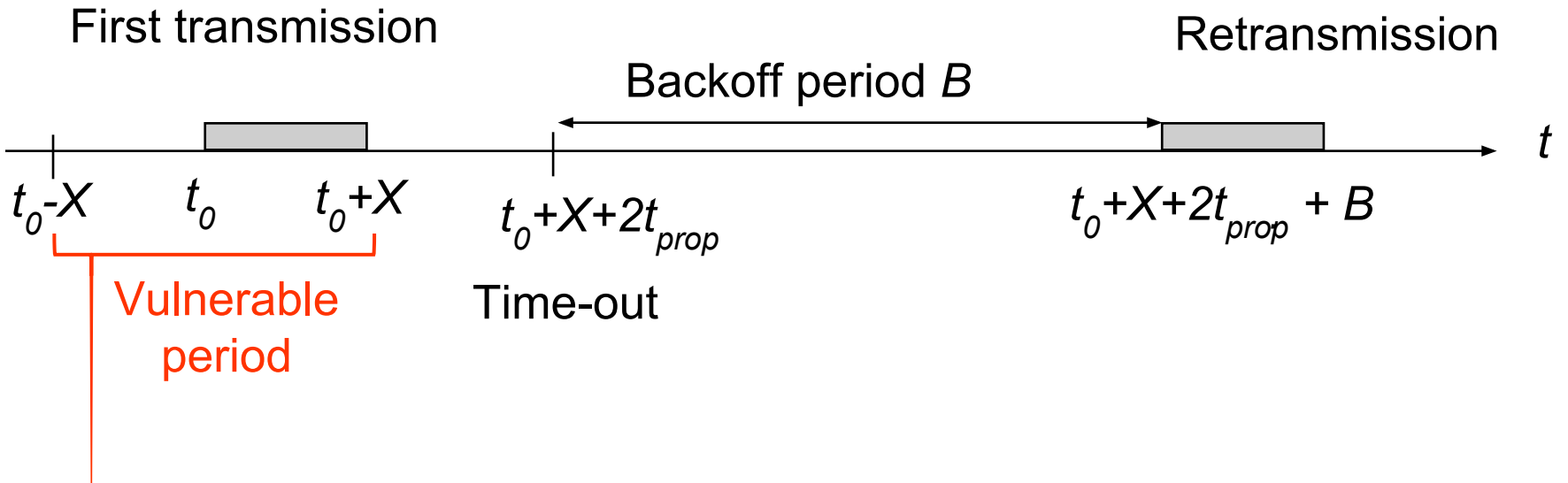
Ethernet: CSMA/CD Protocol



- **Carrier sense**: wait for link to be idle
- **Collision detection**: listen while transmitting
  - No collision: transmission is complete
  - Collision: abort transmission & send **jam** signal
- **Random access**: binary exponential back-off
  - After collision, wait a random time before trying again
  - After m^{th} collision, choose K randomly from $\{0, \ldots, 2^m-1\}$
  - … and wait for K*512 bit times before trying again
    - Using min packet size as "slot"
    - **If transmission occurring when ready to send, wait until end of transmission (CSMA)**

ALOHA

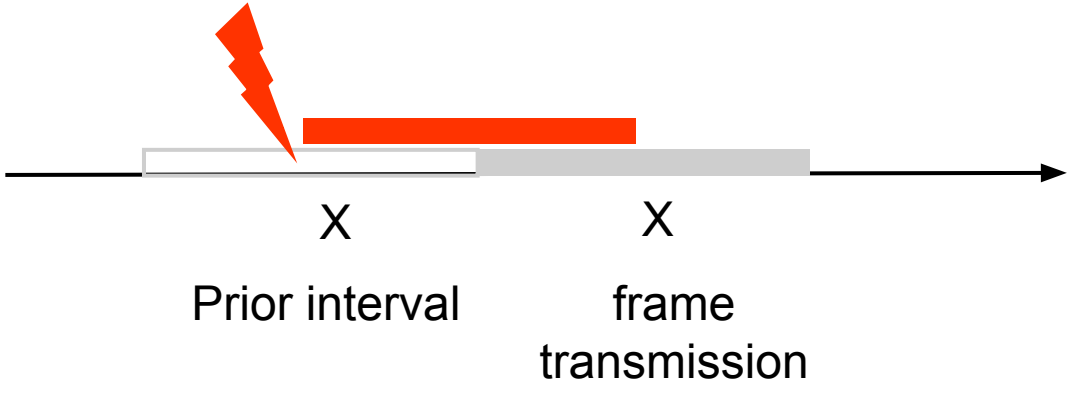
- Wireless link to provide data transfer between main campus & remote campuses of University of Hawaii
- Simplest solution: just do it
  - A station transmits whenever it has data to transmit
  - If more than one frames are transmitted, they interfere with each other (collide) and are lost
  - If ACK not received within timeout, then a station picks random backoff time (to avoid repeated collision)
  - Station retransmits frame after backoff time



ALOHA Model

- Definitions and assumptions
  - X: frame transmission time (assumed to be constant)
  - S: throughput (average # of successful frame transmissions per X seconds)
  - G: load (average # of transmission attempts per X sec.)
  - $P_{success}$: probability a frame transmission is successful

$$S = GP_{success}$$



- Any transmission that begins during vulnerable period leads to collision
- Success if no arrivals during 2X seconds

Abramson's Assumption

- *What is probability of no arrivals in vulnerable period?*
- Abramson assumption: The backoff algorithm spreads the retransmissions so that frame transmissions, new and repeated, are equally likely to occur at any instant
- This implies that the number of frames transmitted in a time interval has a Poisson distribution

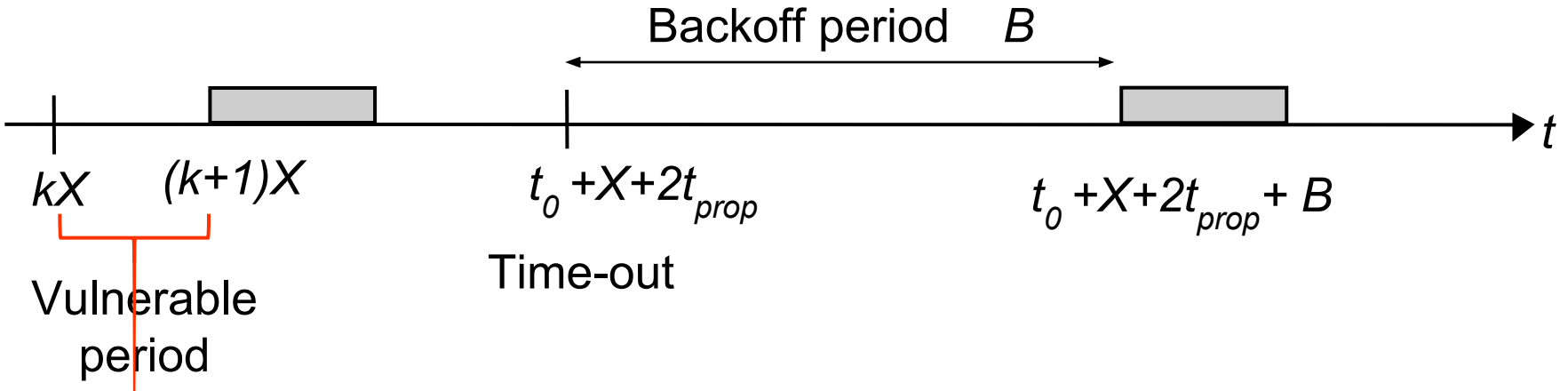
$$P[k \text{ arrivals in } t \text{ seconds}] = \frac{(\lambda t)^k}{k!} e^{-\lambda t}$$

$$P_{success} = P[0 \text{ arrivals in } 2\text{X seconds}]$$

$$= \frac{(2\lambda X)^0}{0!} e^{-2\lambda X} = e^{-2G}$$

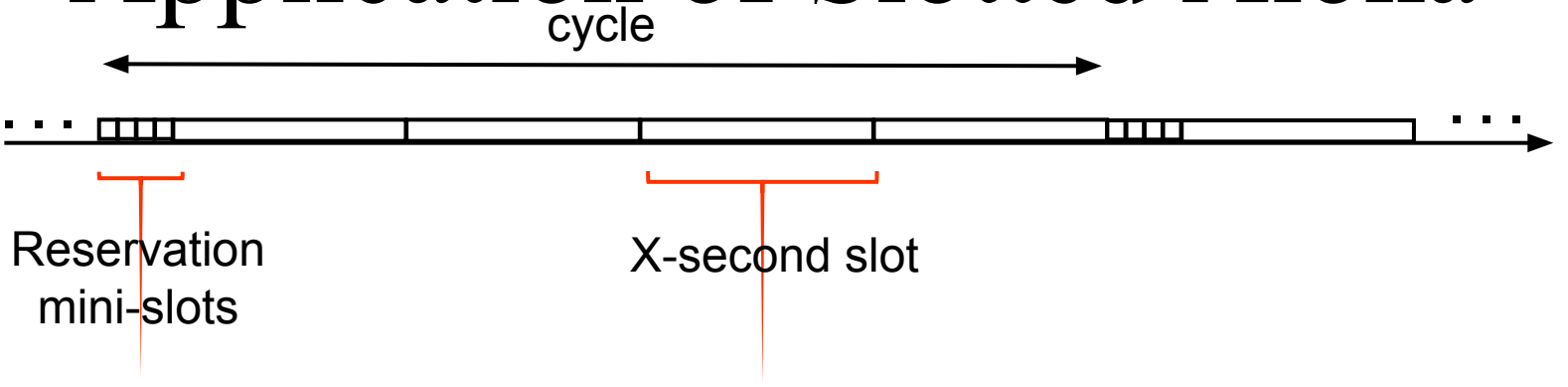
Slotted ALOHA

- Time is slotted in X seconds slots
- Stations synchronized to frame times
- Stations transmit frames in first slot after frame arrival
- Backoff intervals in multiples of slots



Only frames that arrive during prior X seconds collide

Application of Slotted Aloha



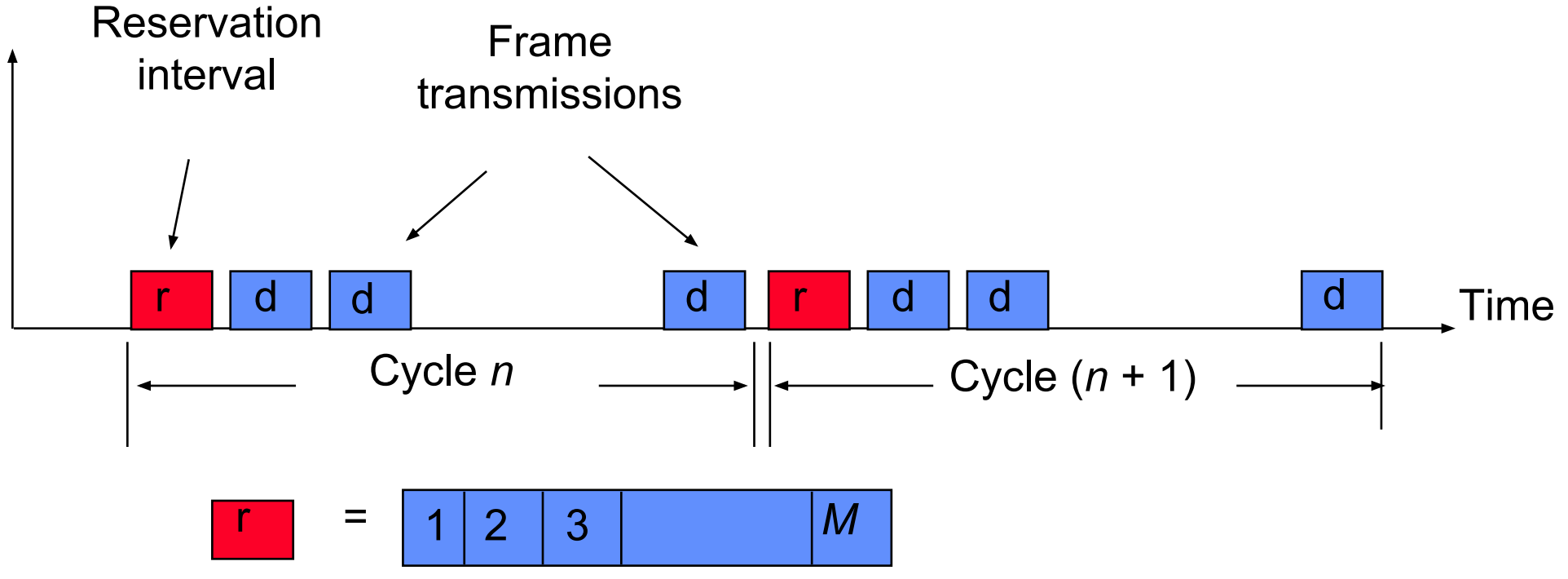
- Reservation protocol allows a large number of stations with infrequent traffic to reserve slots to transmit their frames in future cycles
- Each cycle has mini-slots allocated for making reservations
- Stations use slotted Aloha during mini-slots to request slots

- Random Access in wired medium

Scheduling for Medium Access Control

- Schedule frame transmissions to avoid collision in shared medium
  - ✔ More efficient channel utilization
  - ✔ Less variability in delays
  - ✔ Can provide fairness to stations
  - ✘ Increased computational or procedural complexity
- Two main approaches
  - Reservation
  - Polling

Reservation Systems



- Transmissions organized into cycles
- Cycle: reservation interval + frame transmissions
- The reservation intervals has a minislot for each station to request reservations for frame transmissions

Reservation System Options

- Centralized or distributed system
  - *Centralized systems*: A central controller listens to reservation information, decides order of transmission, issues grants
  - *Distributed systems*: Each station determines its slot for transmission from the reservation information
- Single or Multiple Frames
  - *Single frame reservation*: Only one frame transmission can be reserved within a reservation cycle
  - *Multiple frame reservation*: More than one frame transmission can be reserved within a frame
- Channelized or Random Access Reservations
  - *Channelized (typically TDMA) reservation*: Reservation messages from different stations are multiplexed without any risk of collision
  - *Random access reservation*: Each station transmits its reservation message randomly until the message goes through

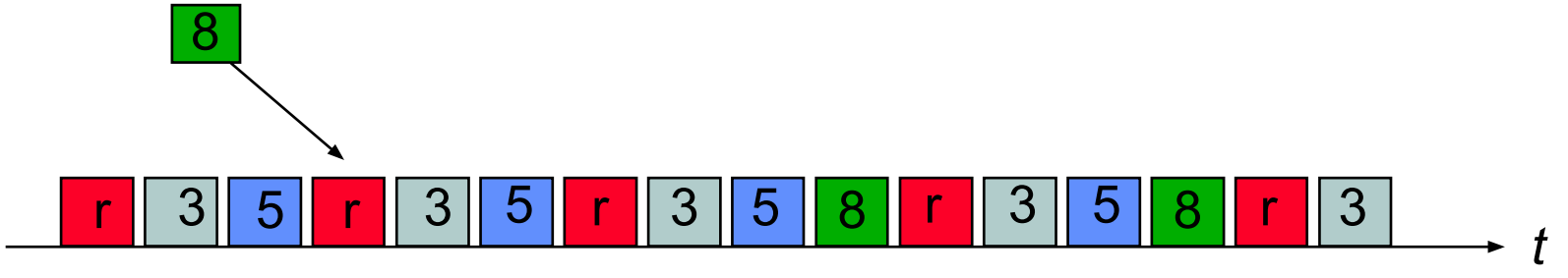
Example

- Initially stations 3 & 5 have reservations to transmit frames

(a)

- Station 8 becomes active and makes reservation
- Cycle now also includes frame transmissions from station 8

(b)



Random Access Reservation Systems

- *Large number of light traffic stations*
  - Dedicating a minislot to each station is inefficient
- Slotted ALOHA reservation scheme
  - Stations use slotted Aloha on reservation minislots
  - On average, each reservation takes at least *e* minislot attempts
  - Effective time required for the reservation is 2.71*vX*

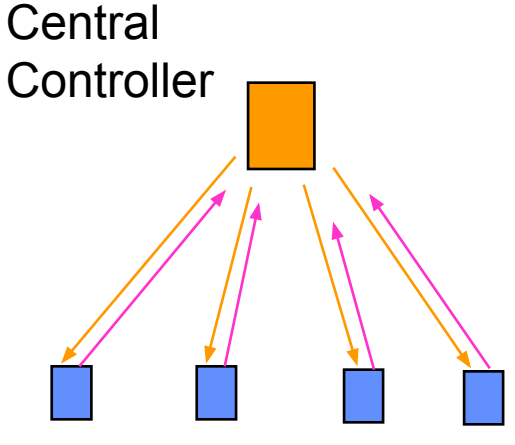
$$\rho_{\max} = \frac{X}{X(1+ev)} = \frac{1}{1+2.71\mathrm{v}}$$

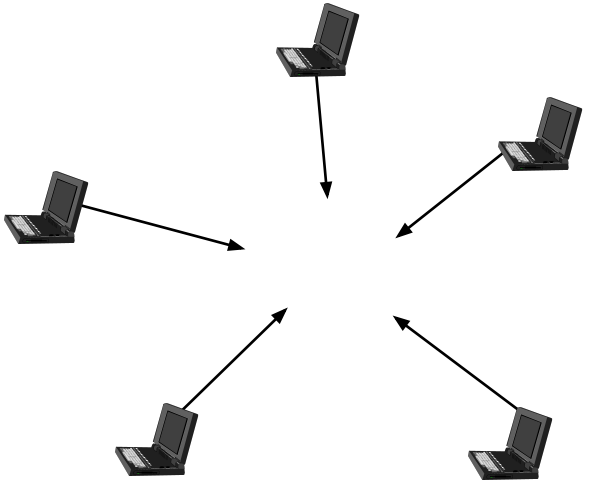
Example: GPRS

- General Packet Radio Service
  - Packet data service in GSM cellular radio
  - GPRS devices, e.g. cellphones or laptops, send packet data over radio and then to Internet
  - Slotted Aloha MAC used for reservations
  - Single & multi-slot reservations supported

Polling Systems

- *Centralized polling systems*: A central controller transmits polling messages to stations according to a certain order
- *Distributed polling systems*: A permit for frame transmission is passed from station to station according to a certain order
- A signaling procedure exists for setting up order





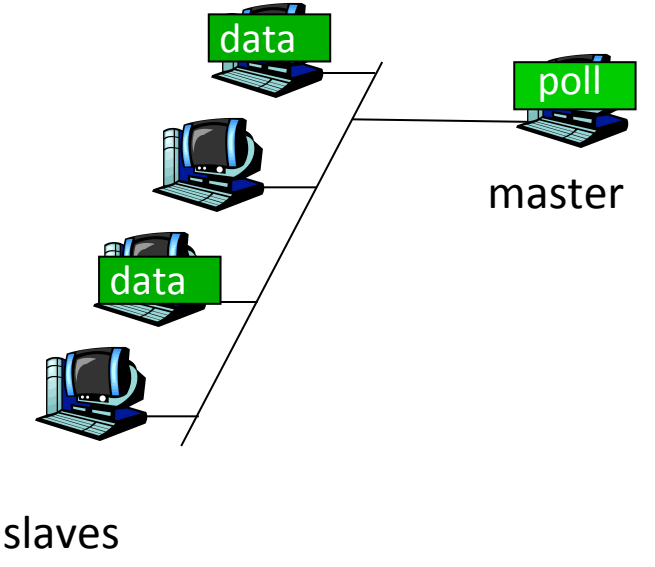
Polling System Options

- Service Limits: How much is a station allowed to transmit per poll?
  - *Exhaustive*: until station's data buffer is empty (including new frame arrivals)
  - *Gated*: all data in buffer when poll arrives
  - *Frame-Limited*: one frame per poll
  - *Time-Limited*: up to some maximum time
- Priority mechanisms
  - More bandwidth & lower delay for stations that appear multiple times in the polling list
  - Issue polls for stations with message of priority k or higher

“Taking Turns” MAC protocols

Polling:

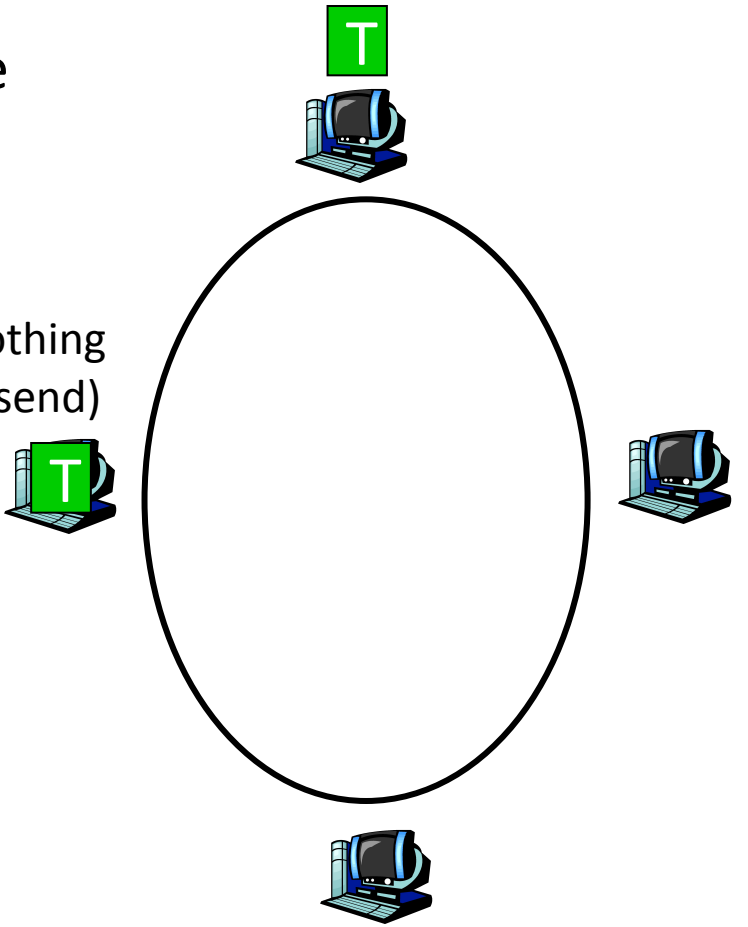
- master node “invites” slave nodes to transmit in turn
- typically used with “dumb” slave devices
- concerns:
  - polling overhead
  - latency
  - single point of failure (master)



"Taking Turns" MAC protocols

Token passing:

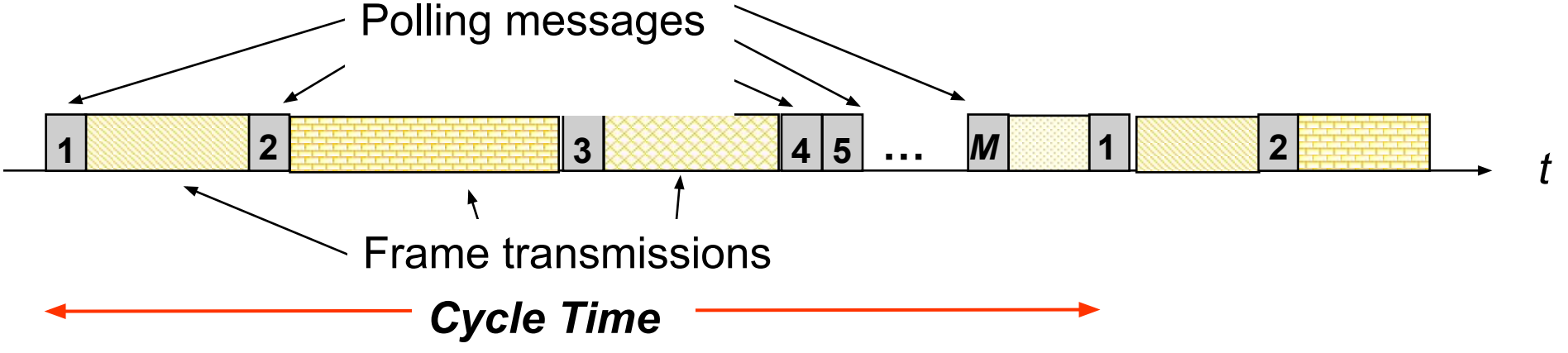
- control **token** passed from one node to next sequentially.
- token message
- concerns:
  - token overhead
  - latency
  - single point of failure (token)
- concerns fixed in part by a slotted ring (many simultaneous *tokens)*



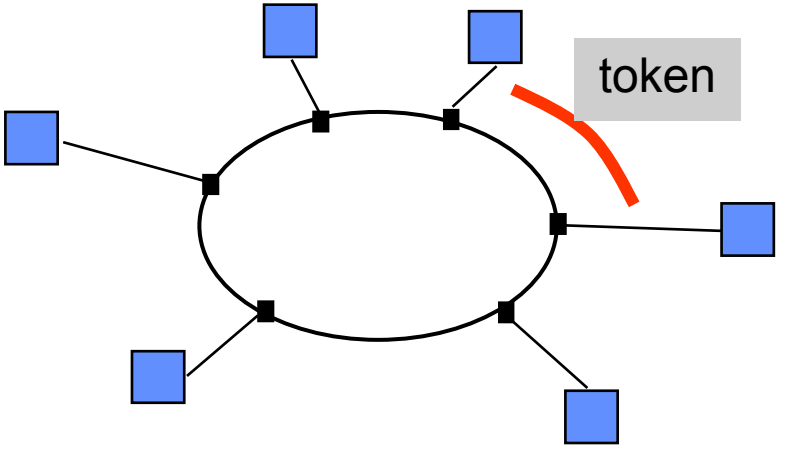
Cambridge students – this is YOUR heritage
Cambridge RING, Cambridge Fast RING,
Cambridge Backbone RING, these things gave us

Walk Time & Cycle Time

- Assume polling order is round robin
- Time is "wasted" in polling stations
  - Time to prepare & send polling message
  - Time for station to respond
- *Walk time:* from when a station completes transmission to when next station begins transmission
- *Cycle time* is between consecutive polls of a station
- Overhead/cycle = total walk time/cycle time

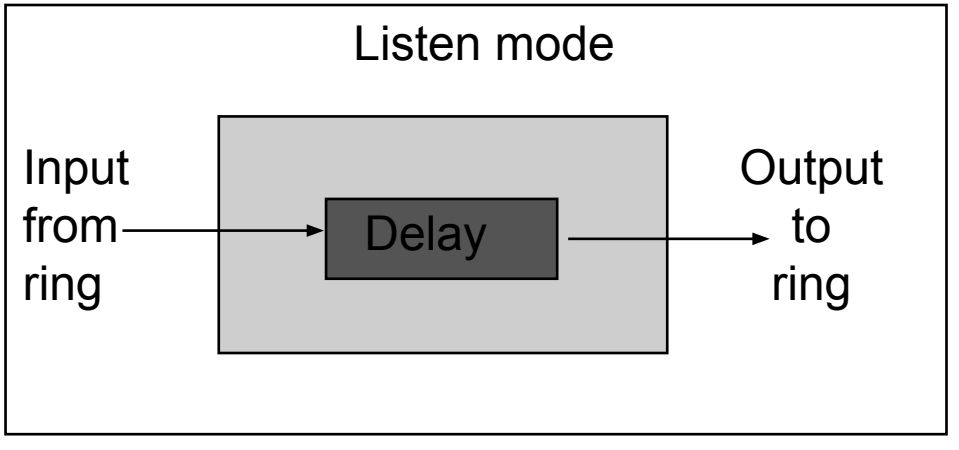


Application: Token-Passing Rings

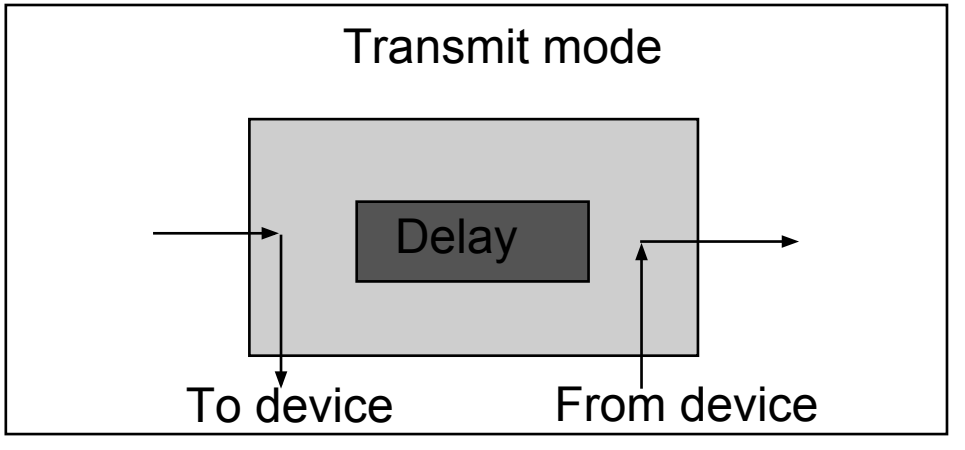


Free Token = Poll

Frame Delimiter is Token
Free = 01111110
Busy = 01111111



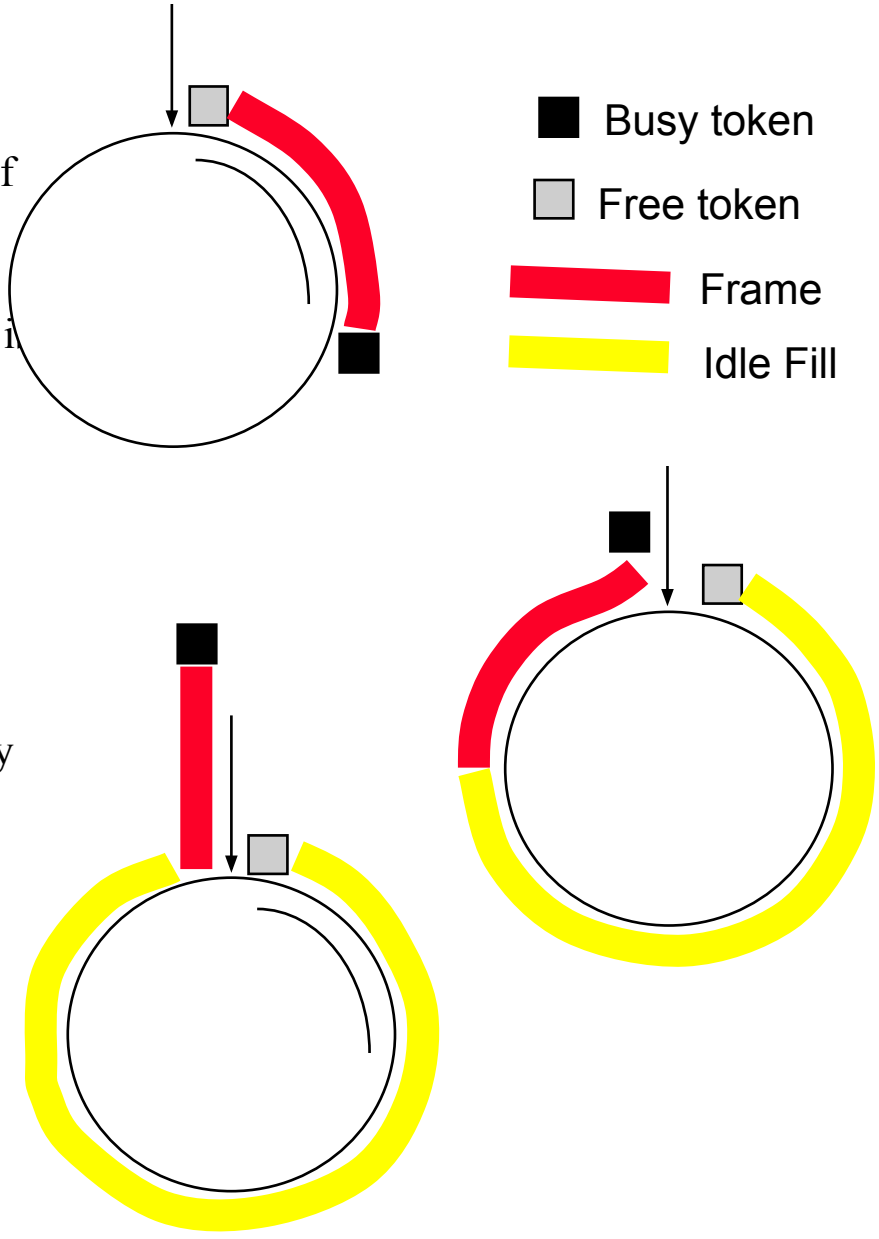
Ready station looks for free token
Flips bit to change free token to busy



Ready station inserts its frames
Reinserts free token when done

Methods of Token Reinsertion

- Multi-token operation
  - Free token transmitted immediately after last bit of data frame
- Single-token operation
  - Free token inserted after last bit of the busy token i received back
  - Transmission time at least ring latency
  - If frame transmission time is longer than ring latency, equivalent to multi-token operation
- Single-Frame operation
  - Free token inserted after transmitting station has received last bit of its frame
  - Equivalent to attaching trailer equal to ring latency



Application Examples

- Single-frame reinsertion
  - IEEE 802.5 Token Ring LAN @ 4 Mbps
- Single token reinsertion
  - IBM Token Ring @ 4 Mbps
- Multitoken reinsertion
  - IEEE 802.5 and IBM Ring LANs @ 16 Mbps
  - FDDI Ring @ 50 Mbps
- All of these LANs incorporate token priority mechanisms

Comparison of MAC approaches

- Aloha & Slotted Aloha
  - Simple & quick transfer at very low load
  - Accommodates a large number of low-traffic bursty users
  - Highly variable delay at moderate loads
  - Efficiency does not depend on *a*
- CSMA-CD
  - Quick transfer and high efficiency for low delay-bandwidth product
  - Can accommodate a large number of bursty users
  - Variable and unpredictable delay

Comparison of MAC approaches

- Reservation
  - On-demand transmission of bursty or steady streams
  - Accommodates large number of low-traffic users with slotted Aloha reservations
  - Can incorporate QoS
  - Handles large delay-bandwidth product via delayed grants
- Polling
  - Generalization of time-division multiplexing
  - Provides fairness through regular access opportunities
  - Can provide bounds on access delay
  - Performance deteriorates with large delay-bandwidth product

- Random access in wireless

802.11 Architecture

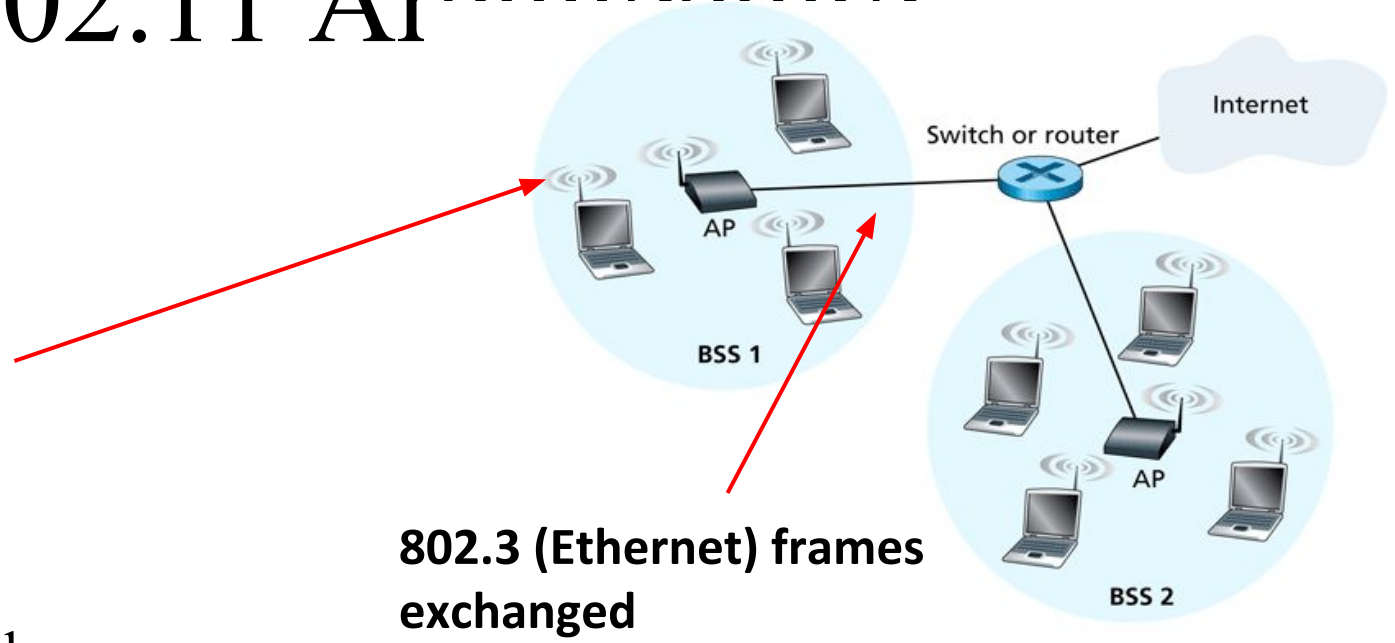


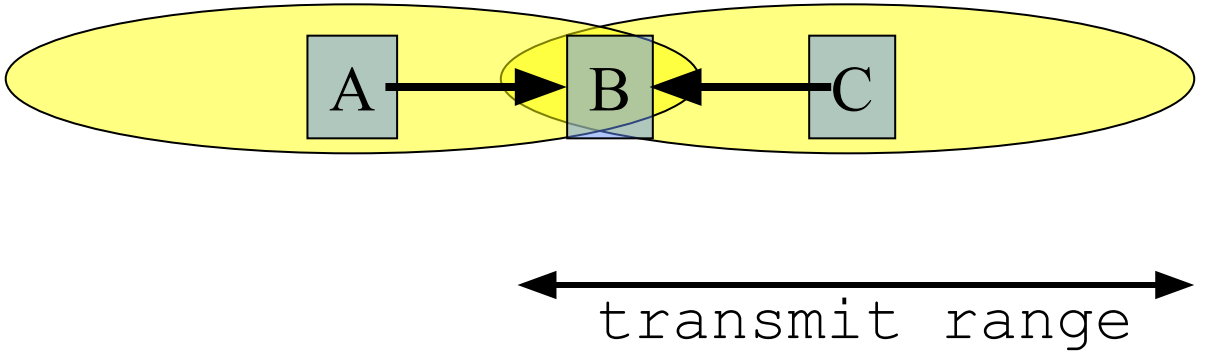
Figure 6.7 ♦ IEEE 802.11 LAN architecture

- Designed for limited area
- AP's (Access Points) set to specific channel
- Broadcast beacon messages with SSID (Service Set Identifier) and MAC Address periodically
- Hosts scan all the channels to discover the AP's
  - Host associates with AP

Wireless Multiple Access Technique?

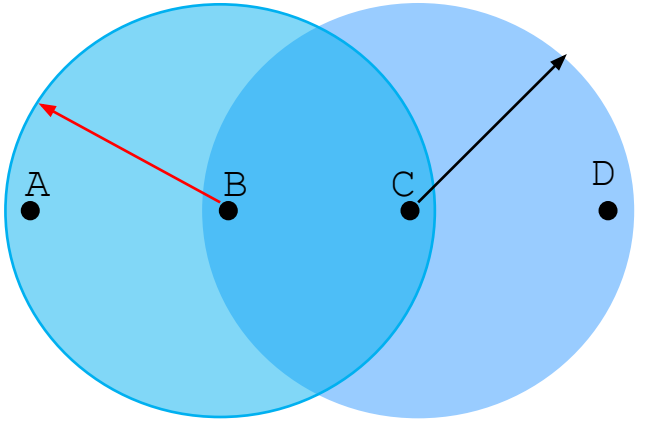
- Carrier Sense?
  - Sender can listen before sending
  - What does that tell the sender?
- Collision Detection?
  - Where do collisions occur?
  - How can you detect them?

Hidden Terminals



- A and C can both send to B but can't hear each other
  - A is a *hidden terminal* for C and vice versa
- Carrier Sense will be ineffective

Exposed Terminals



- Exposed node: B sends a packet to A; C hears this and decides not to send a packet to D (despite the fact that this will not cause interference)!
- Carrier sense would prevent a successful transmission.

Key Points

- No concept of a global collision
  - Different receivers hear different signals
  - Different senders reach different receivers

- Collisions are at receiver, not sender
  - Only care if receiver can hear the sender clearly
  - It does not matter if sender can hear someone else
  - As long as that signal does not interfere with receiver

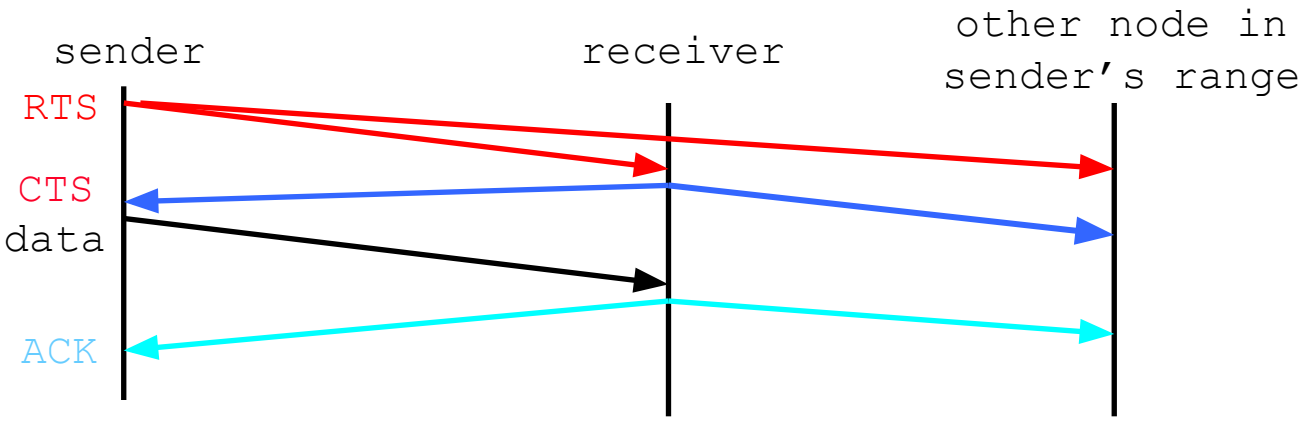
- Goal of protocol:
  - Detect if receiver can hear sender
  - Tell senders who might interfere with receiver to shut up

Basic Collision Avoidance

- Since can't detect collisions, we try to *avoid* them
- Carrier sense:
  - When medium busy, choose random interval
  - Wait that many **idle** timeslots to pass before sending

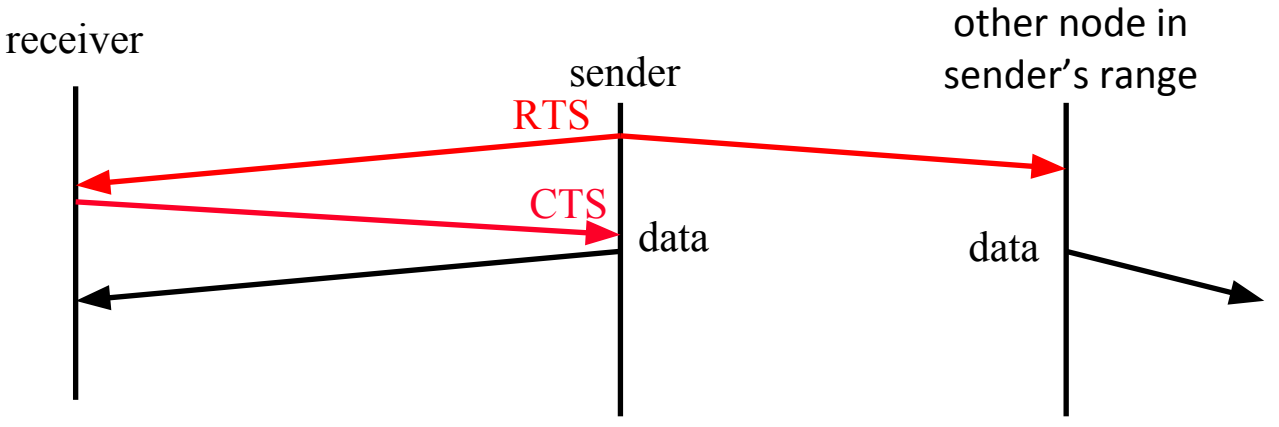
- When a collision is inferred, retransmit with binary exponential backoff (like Ethernet)
  - Use ACK from receiver to infer "no collision"
  - Use exponential backoff to adapt contention window

CSMA/CA -MA with Collision Avoidance



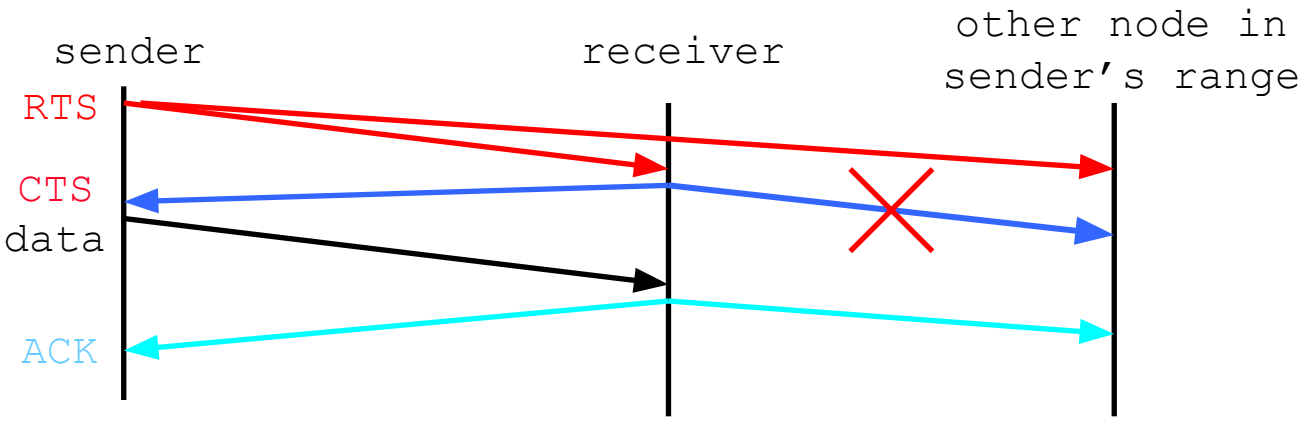
- Before every data transmission
  - Sender sends a Request to Send (RTS) frame containing the length of the transmission
  - Receiver respond with a Clear to Send (CTS) frame
  - Sender sends data
  - Receiver sends an ACK; now another sender can send data
- When sender doesn't get a CTS back, it assumes collision

CSMA/CA, con't



- If other nodes hear RTS, but not CTS: send
  - Presumably, destination for first sender is out of node's range …

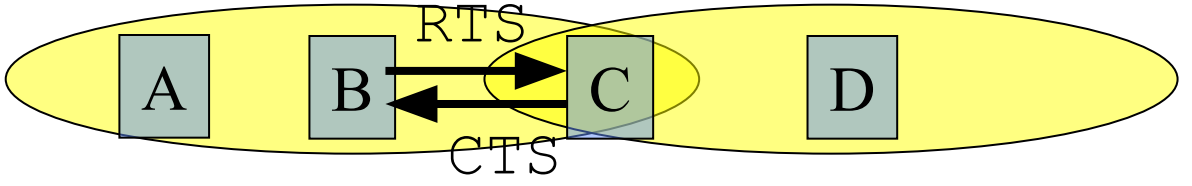
CSMA/CA, con't



- If other nodes hear RTS, but not CTS: send
  - Presumably, destination for first sender is out of node's range …
  - … Can cause problems when a CTS is lost
- When you hear a CTS, you keep quiet until scheduled transmission is over (hear ACK)

RTS / CTS Protocols (CSMA/CA)

B sends to C

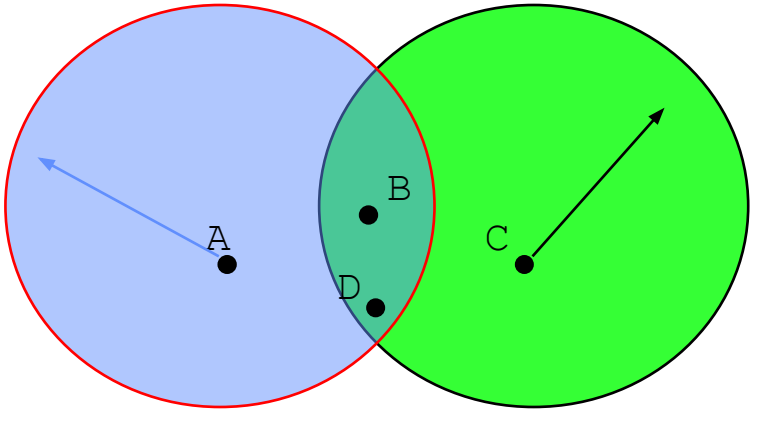


Overcome hidden terminal problems with contention-free protocol

1. B sends to C Request To Send (RTS)
2. A hears RTS and defers (to allow C to answer)
3. C replies to B with Clear To Send (CTS)
4. D hears CTS and defers to allow the data
5. B sends to C

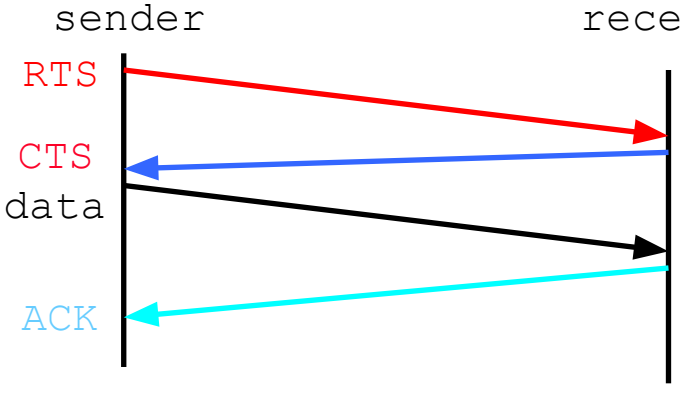
Preventing Collisions Altogether

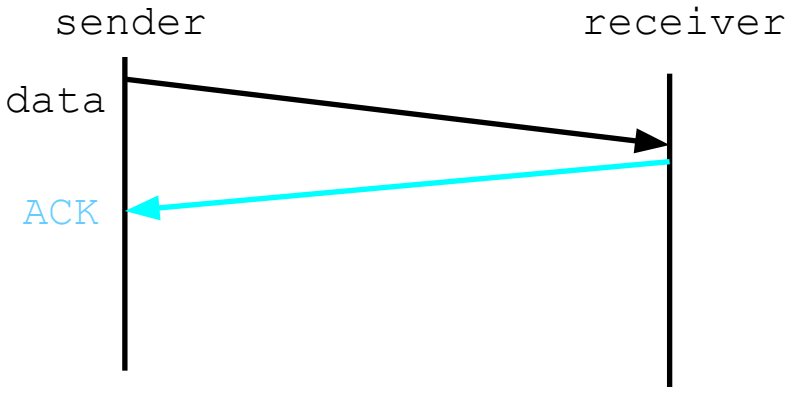
- Frequency Spectrum partitioned into several channels
  - Nodes within interference range can use separate channels



  - Now A and C can send without any interference!
- Most cards have only 1 transceiver
  - **Not Full Duplex: Cannot send and receive at the same time**
  - Aggregate Network throughput doubles

CSMA/CA and RTS/CTS





RTS/CTS

- helps with hidden terminal
- good for high-traffic Access Points
- often turned on/off dynamically

Without RTS/CTS

- lower latency -> faster!
- reduces wasted b/w

  if the *Pr(collision)* is low
- good for when net is small and not *weird*

  eg no hidden/exposed terminals

CSMA/CD vs CSMA/CA (without RTS/CTS)

CD Collision Detect

wired – listen and talk

1. Listen for others
2. Busy? goto 1.
3. Send message (and listen)
4. Collision?
    a. JAM
    b. increase your BEB
    c. sleep
    d. goto 1.

CA Collision Avoidance

wireless – talk OR listen

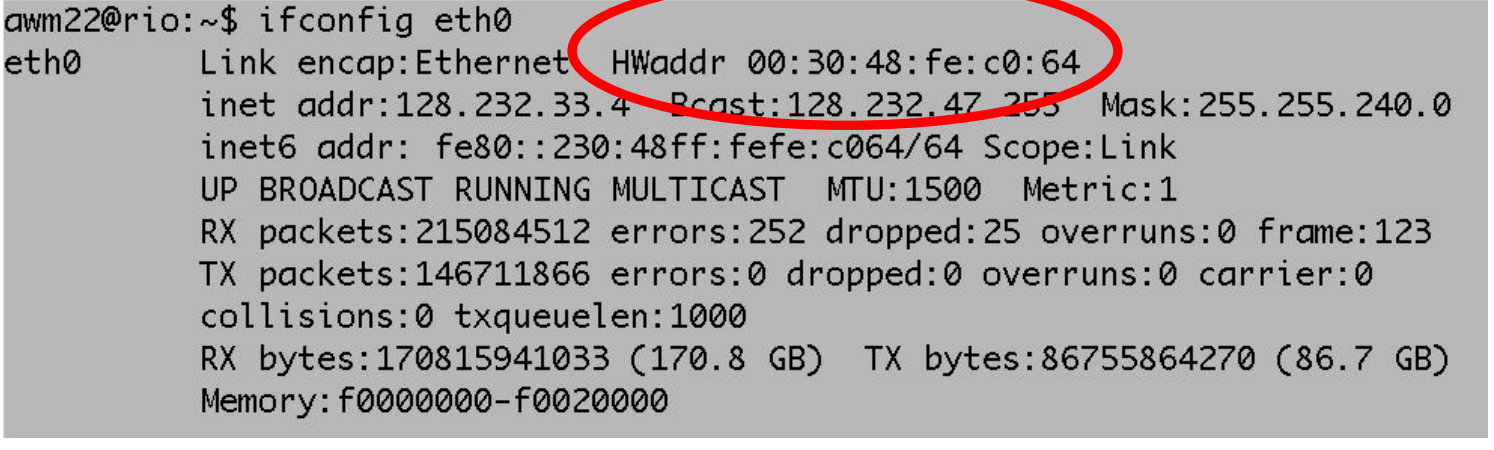
1. Listen for others
2. Busy?
    a. increase your BEB
    b. sleep
    c. goto 1.
3. Send message
4. Wait for ACK (*MAC ACK*)
5. Got No ACK from MAC?
    a. increase your BEB
    b. sleep
    c. goto 1.

Summary of MAC protocols

- *channel partitioning,* by time, frequency or code
  - Time Division, Frequency Division
- *random access* (dynamic),
  - ALOHA, S-ALOHA, CSMA, CSMA/CD
  - carrier sensing: easy in some technologies (wire), hard in others (wireless)
  - CSMA/CD used in Ethernet
  - CSMA/CA used in 802.11
- *taking turns*
  - polling from central site, token passing
  - Bluetooth, FDDI, IBM Token Ring

MAC Addresses

- MAC (or LAN or physical or Ethernet) address:
  - function: *get frame from one interface to another physically-connected interface (same network)*
  - 48 bit MAC address (for most LANs)
    - *burned* in NIC ROM, nowadays usually software



Unit III- Network Layer

Dr. N Kumaran
Assistant Professor/CSE
SCSVMV University

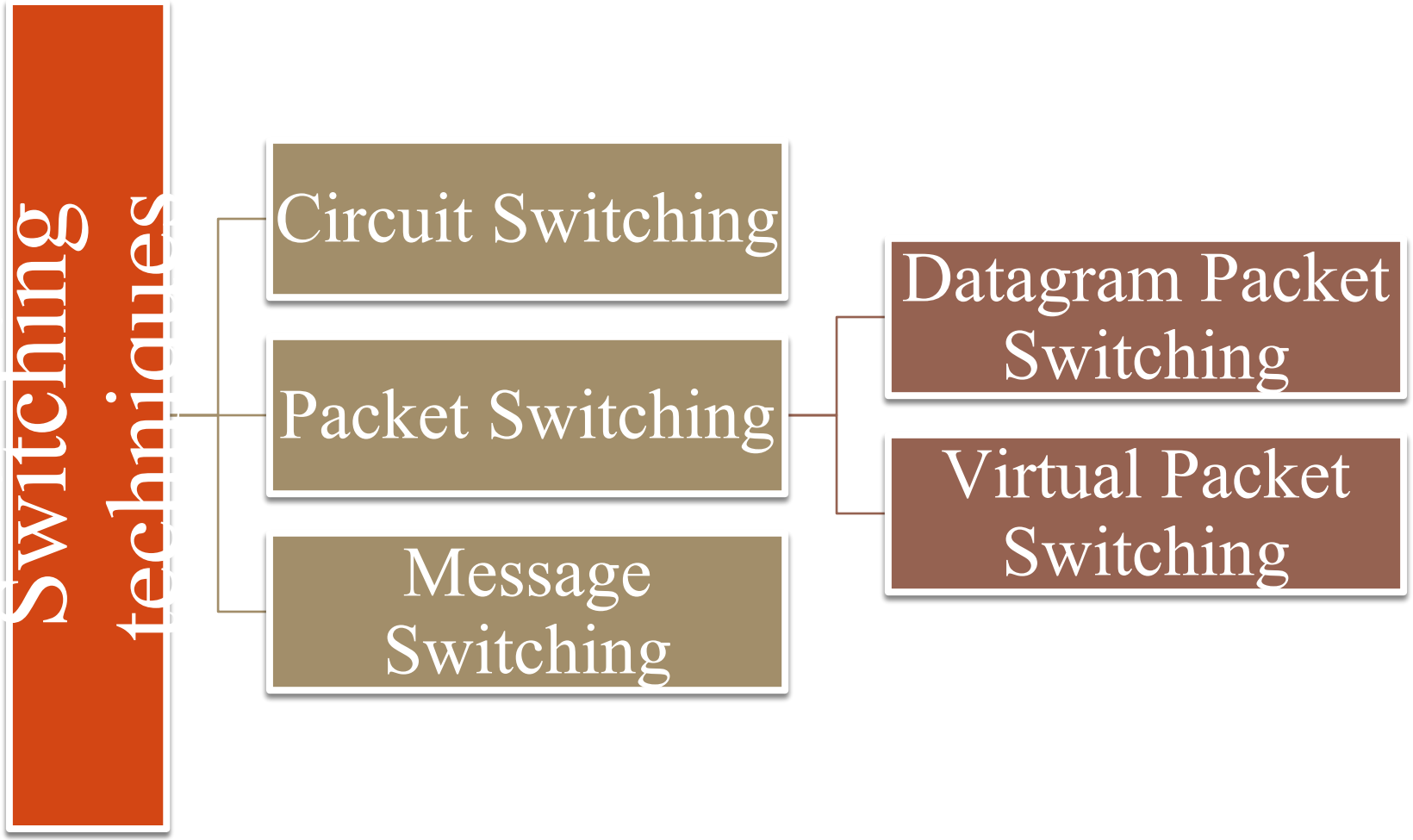
Network Layer Design Isues

- Store-and-Forward Packet Switching
- Services Provided to the Transport Layer
- Implementation of Connectionless Service
- Implementation of Connection-Oriented Service
- Comparison of Virtual-Circuit and Datagram Subnets

Outline

- **Switching techniques,**

Switching techniques



Circuit Switching

- When two nodes communicate with each other over a dedicated communication path, it is called circuit switching.
- There 'is a need of pre-specified route from which data will travels and no other data is permitted.
- In circuit switching, to transfer the data, circuit must be established so that the data transfer can take place.
- Circuits can be permanent or temporary. Applications which use circuit switching may have to go through **three phases**:
  - Establish a circuit
  - Transfer the data
  - Disconnect the circuit

Circuit Switching

- Circuit switching was designed for voice applications.
- Telephone is the best suitable example of circuit switching. Before a user can make a call, a virtual path between caller and callee is established over the

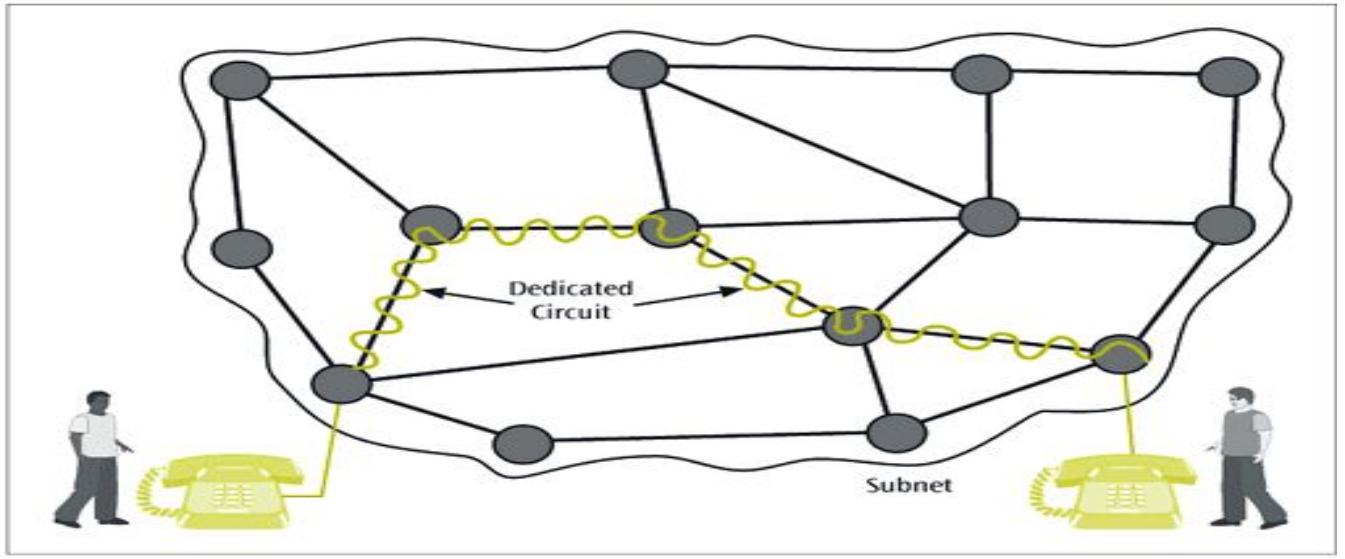
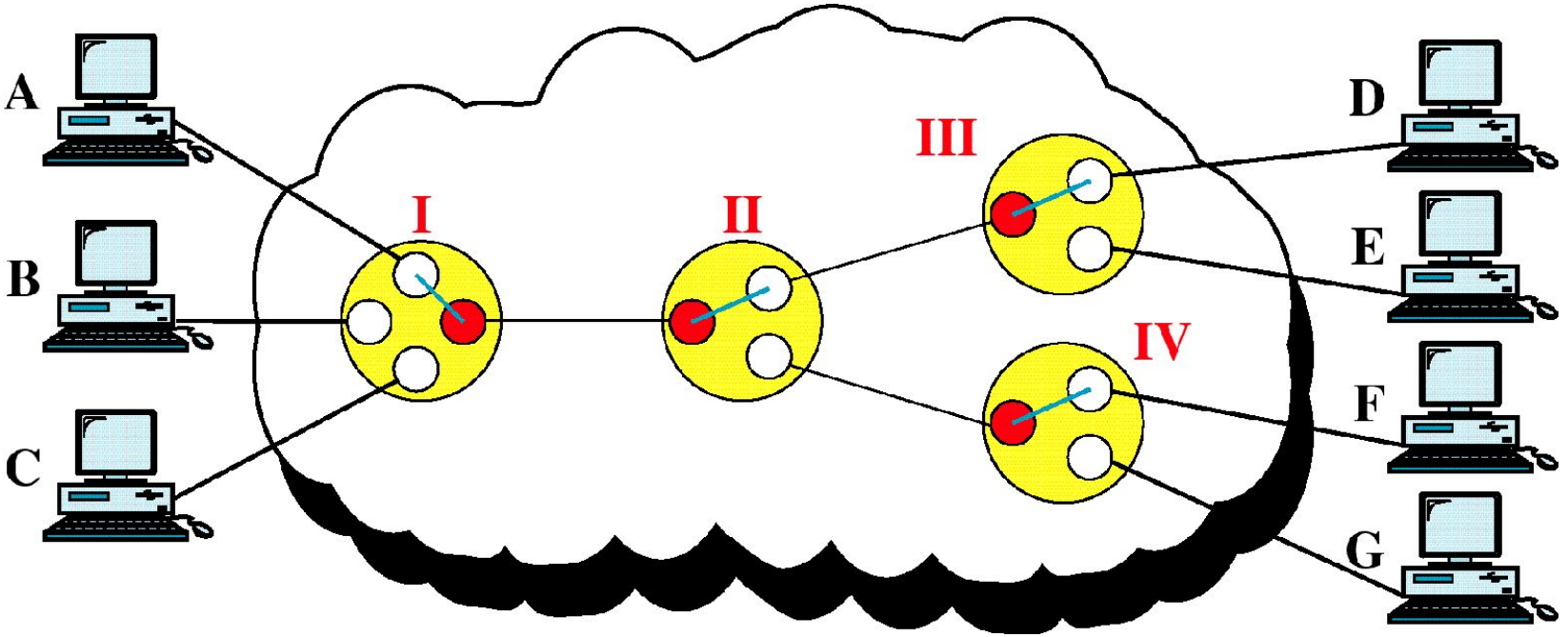


Figure 10-6
Two people carrying on a telephone conversation using a circuit-switched network

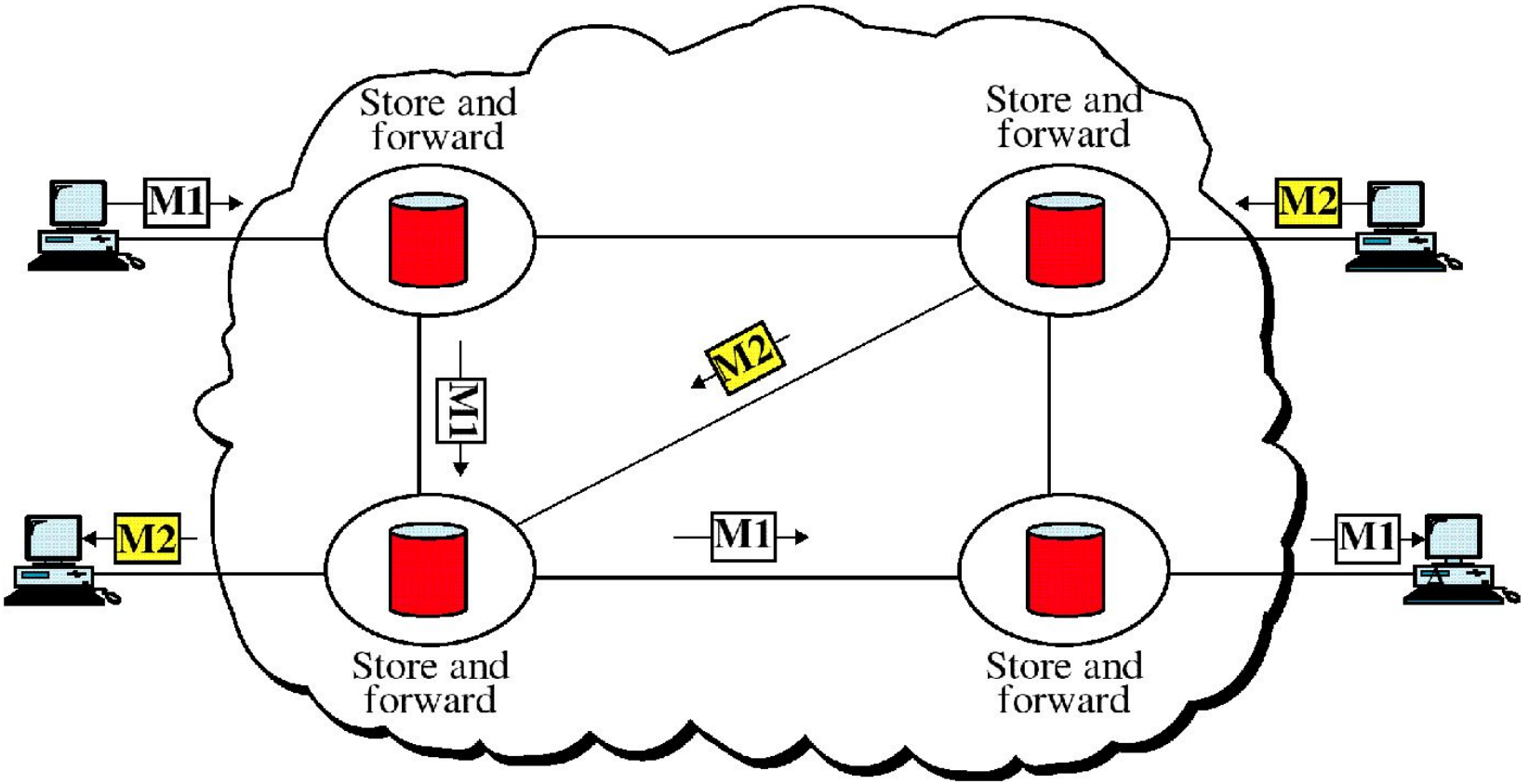
Circuit Switched Networks



Message Switching

- This technique was somewhere in middle of circuit switching and packet switching. In message switching, the whole message is treated as a data unit and is switching / transferred in its entirety.
- A switch working on message switching, first receives the whole message and buffers it until there are resources available to transfer it to the next hop.
- If the next hop is not having enough resource to accommodate large size message, the message is stored and switch waits.

Message Switching



Message Switching drawbacks

- Every switch in transit path needs enough storage to accommodate entire message.
- Because of store-and-forward technique and waits included until resources are available, message switching is very slow.
- Message switching was not a solution for streaming media and real-time applications.

Packet Switching

- Shortcomings of message switching gave birth to an idea of packet switching.
- The entire message is broken down into smaller chunks called packets.
- The switching information is added in the header of each packet and transmitted independently.
- It is easier for intermediate networking devices to store small size packets and they do not take much resources either on carrier path or in the internal memory of switches.

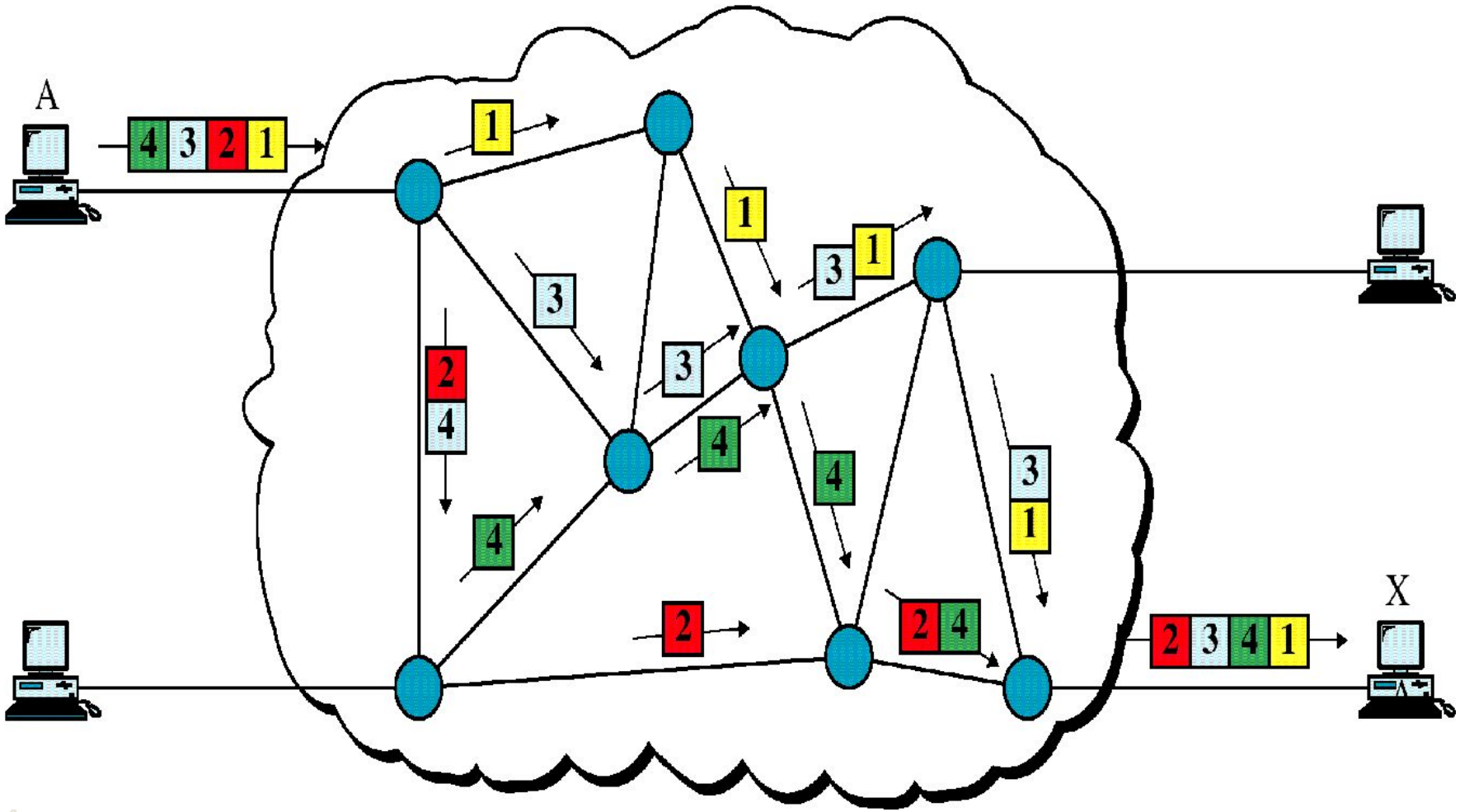
Packet Switching Technique

- A station breaks long message into packets
- Packets are sent out to the network sequentially, one at a time
- How will the network handle this stream of packets as it attempts to route them through the network and deliver them to the intended destination?
  - Two approaches
    - **Datagram** approach
    - **Virtual circuit** approach

Datagram

- Each packet is treated independently, with no reference to packets that have gone before.
  - Each node chooses the next node on a packet's path.
- Packets can take any possible route.
- Packets may arrive at the receiver out of order.
- Packets may go missing.
- It is up to the receiver to re-order packets and recover from missing packets.
- Example: Internet

Datagram Approach



Virtual Circuit

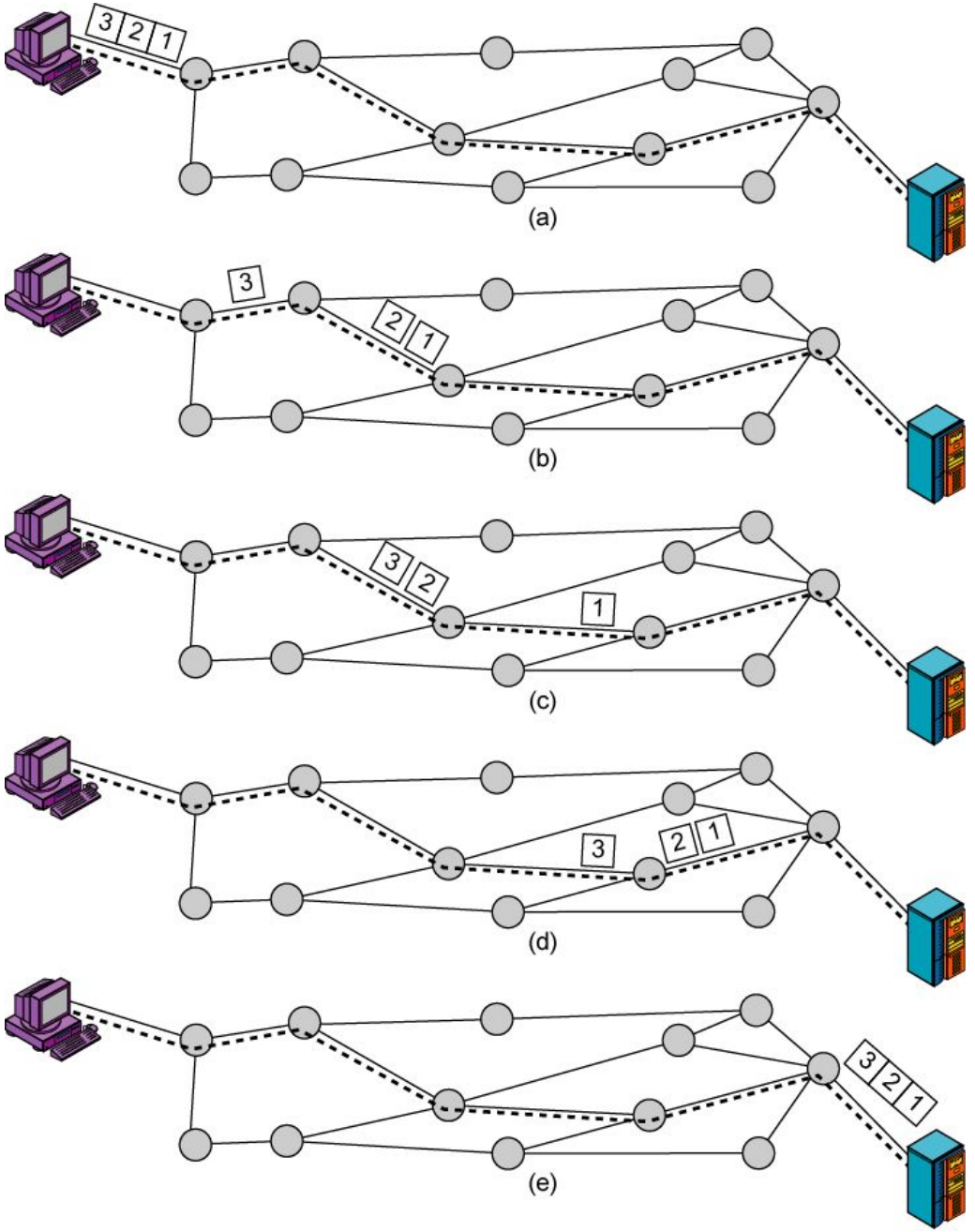
- In virtual circuit, a preplanned route is established before any packets are sent, then all packets follow the same route.
- Each packet contains a **virtual circuit identifier** instead of destination address, and each node on the preestablished route knows where to forward such packets.
  - The node need not make a routing decision for each packet.
- Example: X.25, Frame Relay, ATM

Virtual Circuit Approach

A route between stations is set up prior to data transfer.

All the data packets then follow the same route.

But there is no dedicated resources reserved for the virtual circuit! Packets need to be stored-and-forwarded.



Comparison of Virtual-Circuit and Datagram Subnets

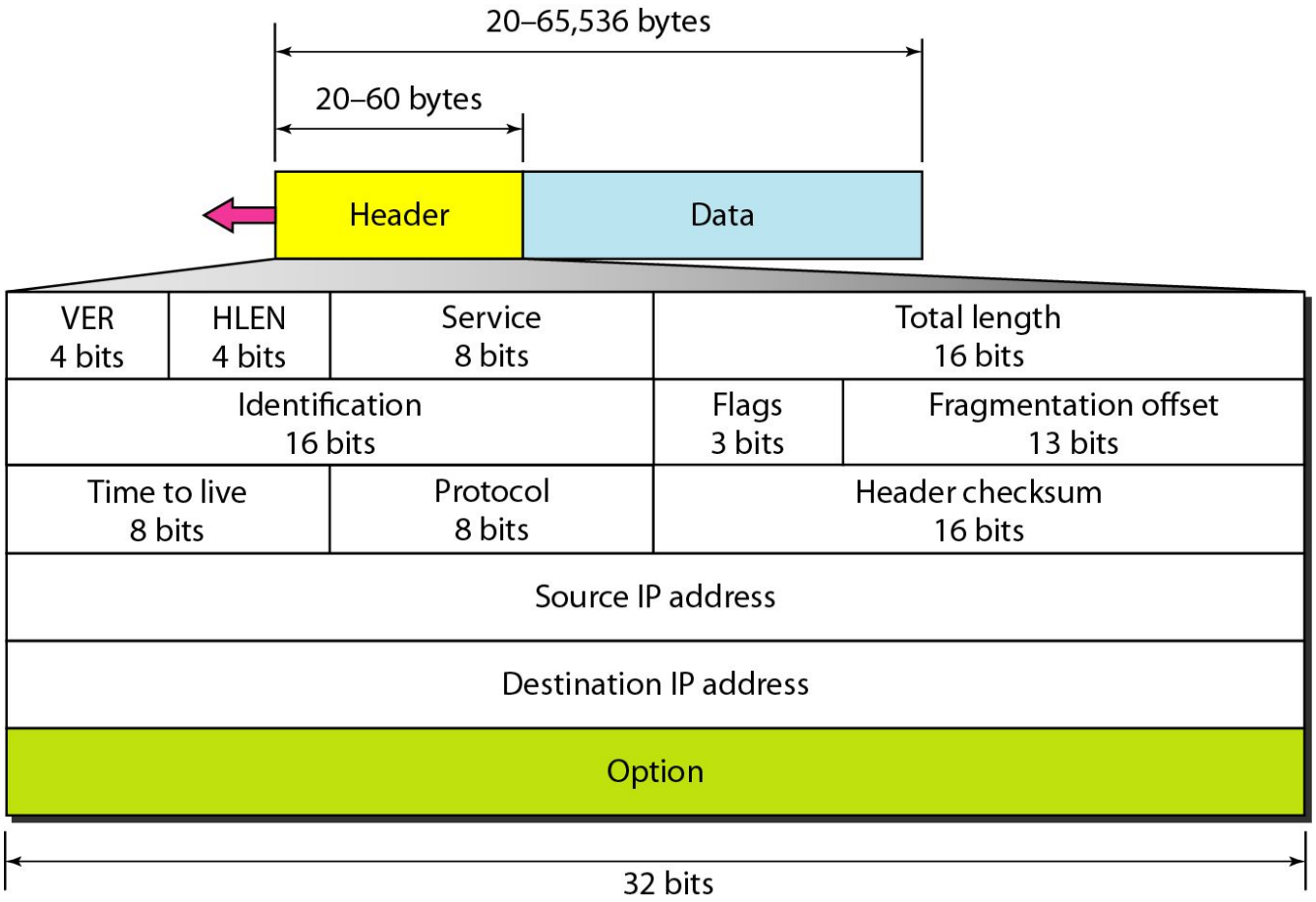
Issue	Datagram subnet	Virtual-circuit subnet
Circuit setup	Not needed	Required
Addressing	Each packet contains the full source and destination address	Each packet contains a short VC number
State information	Routers do not hold state information about connections	Each VC requires router table space per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

Circuit Switching	Datagram Packet Switching	Virtual Circuit Packet Switching
Dedicated transmission path	No dedicated path	No dedicated path
Continuous transmission of data	Transmission of packets	Transmission of packets
Fast enough for interactive	Fast enough for interactive	Fast enough for interactive
Messages are not stored	Packets may be stored until delivered	Packets stored until delivered
The path is established for entire conversation	Route established for each packet	Route established for entire conversation
Call setup delay; negligible transmission delay	Packet transmission delay	Call setup delay; packet transmission delay
Busy signal if called party busy	Sender may be notified if packet not delivered	Sender notified of connection denial
Overload may block call setup; no delay for established calls	Overload increases packet delay	Overload may block call setup; increases packet delay
Electromechanical or computerized switching nodes	Small switching nodes	Small switching nodes
User responsible for message loss protection	Network may be responsible for individual packets	Network may be responsible for packet sequences
Usually no speed or code conversion	Speed and code conversion	Speed and code conversion
Fixed bandwidth	Dynamic use of bandwidth	Dynamic use of bandwidth
No overhead bits after call setup	Overhead bits in each packet	Overhead bits in each packet

Outline

- Switching techniques,
- **IP Protocol,**

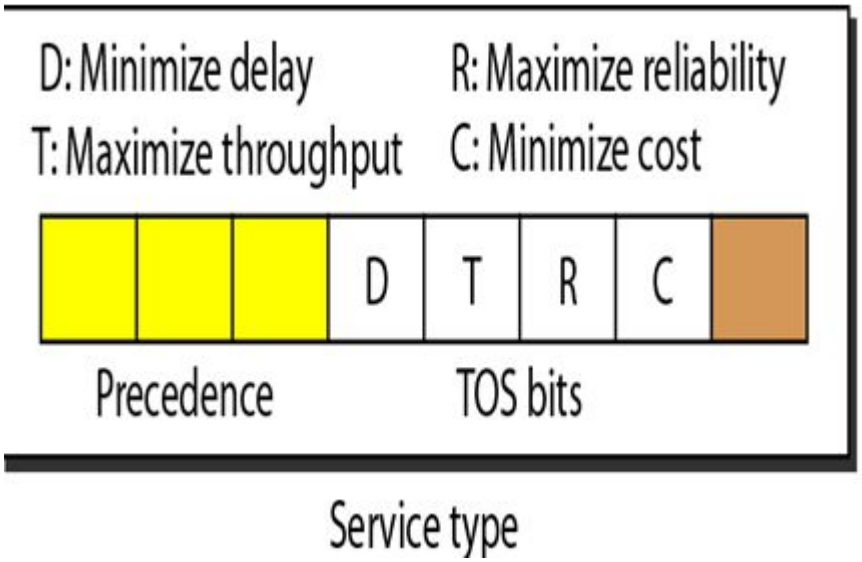
IPv4 datagram format (IPV4 Header)



IPv4 datagram format (IPV4 Header)

- **Version: IP Version**
  - 4 for IPv4
  - 6 for IPv6
- **HLen: Header Length**
  - 32-bit words (typically 5)
- **TOS: Type of Service**
  - Priority information
- **Identifier, flags, fragment offset** □ used primarily for fragmentation
- **Time to live**
  - Must be decremented at each router
  - Packets with TTL=0 are thrown away
  - Ensure packets exit the network
- **Protocol**
  - Demultiplexing to higher layer protocols
  - TCP = 6, ICMP = 1, UDP = 17…
- **Header checksum**
  - Ensures some degree of header integrity
  - Relatively weak – only 16 bits
- **Options**
  - E.g. Source routing, record route, etc.
  - Performance issues at routers
    - Poorly supported or not at all
- **Source Address**
  - 32-bit IP address of sender
- **Destination Address**
  - 32-bit IP address of destination

Service type field in IPV4



Protocol values

Value	*Protocol*
1	ICMP
2	IGMP
6	TCP
17	UDP
89	OSPF

Some of the IPv4 options.

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

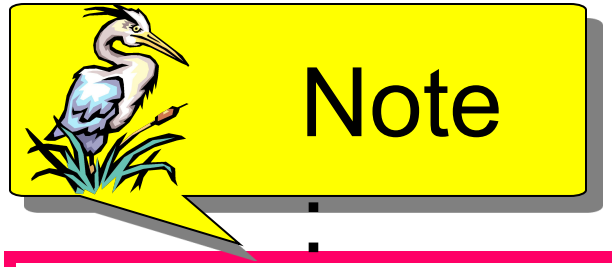
Outline

- Switching techniques,
- IP Protocol,
- **IPv4 and IPv6 addressing schemes,**

IPv4 Addressing- Introduction

An IP address is a 32-bit address that uniquely and universally defines the connection of a host or a router to the Internet.

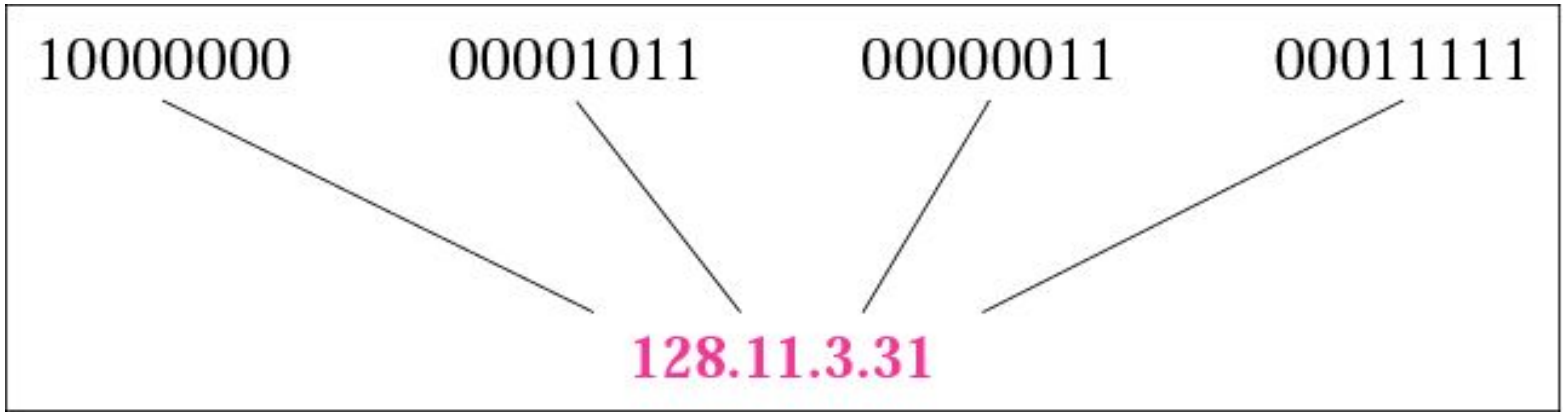
IP addresses are unique.



An IP address is a 32-bit address.

The address space of IPv4 is 2^{32} or 4,294,967,296.

Dotted-decimal and Binary equivalent notation



Example 1

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

a. 10000001 00001011 00001011 11101111
b. 11000001 10000011 00011011 11111111
c. 11100111 11011011 10001011 01101111
d. 11111001 10011011 11111011 00001111

Solution

We replace each group of 8 bits with its equivalent decimal number and add dots for separation:

a. 129.11.11.239 *b. 193.131.27.255*
c. 231.219.139.111 *d. 249.155.251.15*

Example 2

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

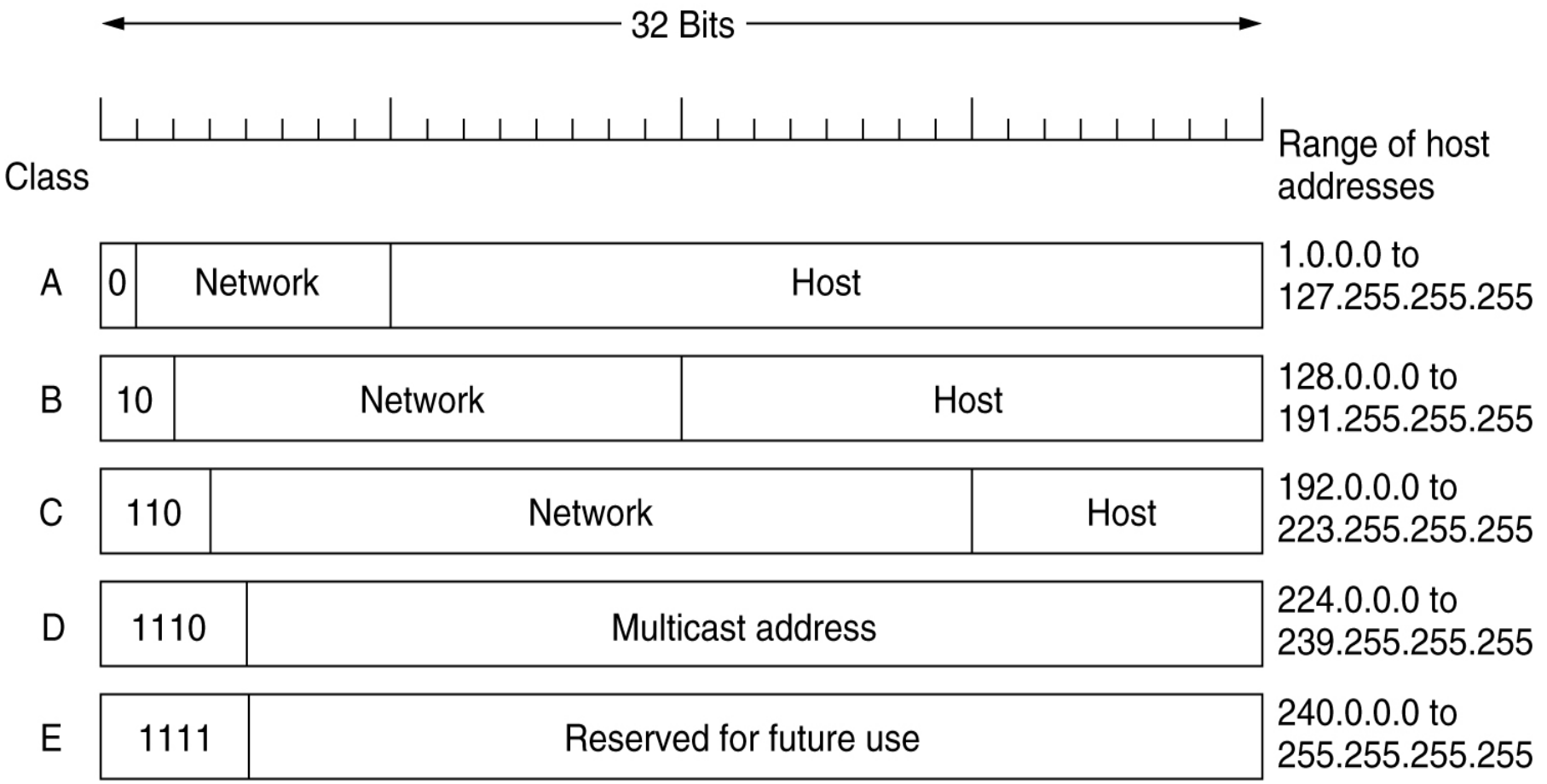
a. 111.56.45.78
b. 221.34.7.82
c. 241.8.56.12
d. 75.45.34.78

Solution

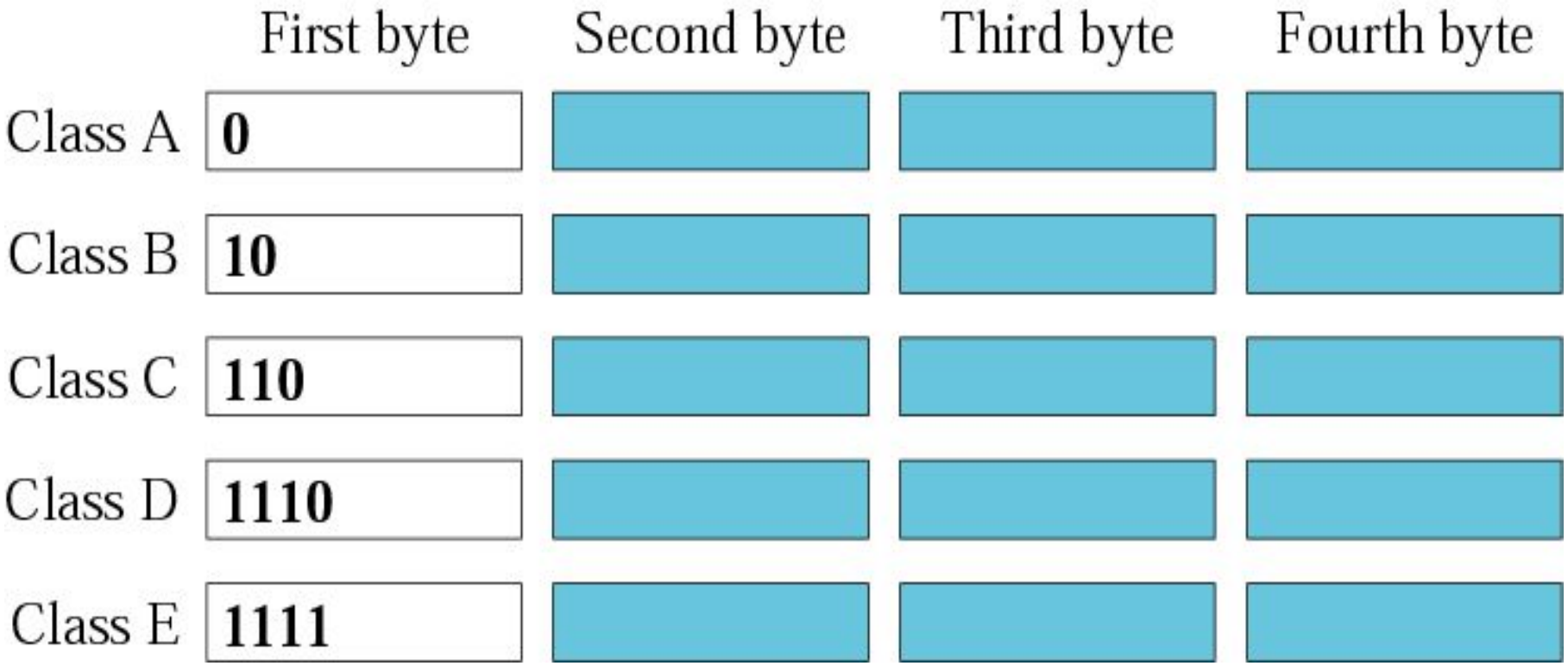
We replace each decimal number with its binary equivalent:

a. 01101111 00111000 00101101 01001110
b. 11011101 00100010 00000111 01010010
c. 11110001 00001000 00111000 00001100
d. 01001011 00101101 00100010 01001110

IP Addresses formats and ranges.



Finding the class in binary notation



Example

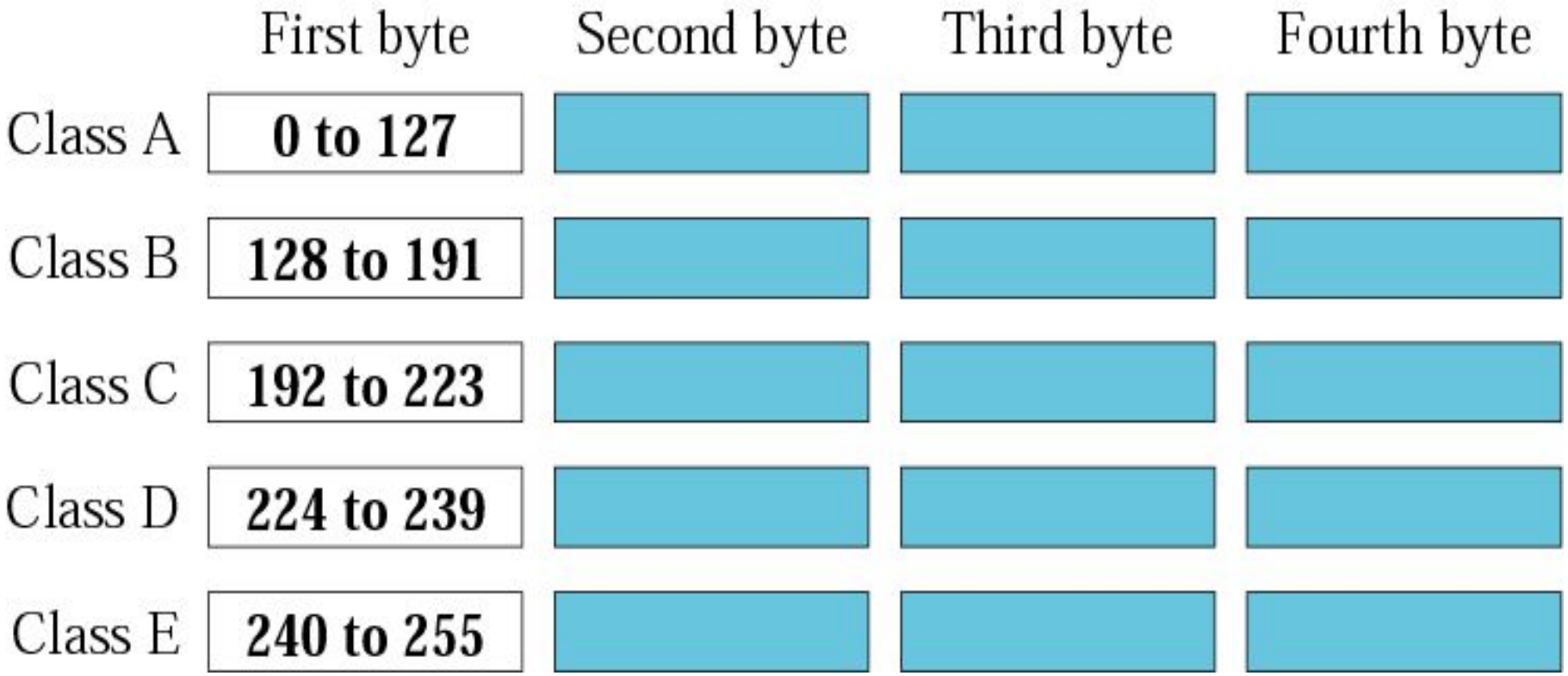
Find the class of each address:

a. 00000001 00001011 00001011 11101111
b. 11000001 10000011 00011011 11111111
c. 10100111 11011011 10001011 01101111
d. 11110011 10011011 11111011 00001111

Solution

a. The first bit is 0. This is a class A address.
b. The first 2 bits are 1; the third bit is 0. This is a class C address.
c. The first bit is 0; the second bit is 1. This is a class B address.
d. The first 4 bits are 1s. This is a class E address..

Finding the class in decimal notation



Example

Find the class of each address:

a. 227.12.14.87 *b.193.14.56.22* *c.14.23.120.8*

d. 252.5.15.111 *e.134.11.78.56*

Solution

a. The first byte is 227 (between 224 and 239); the class is D.

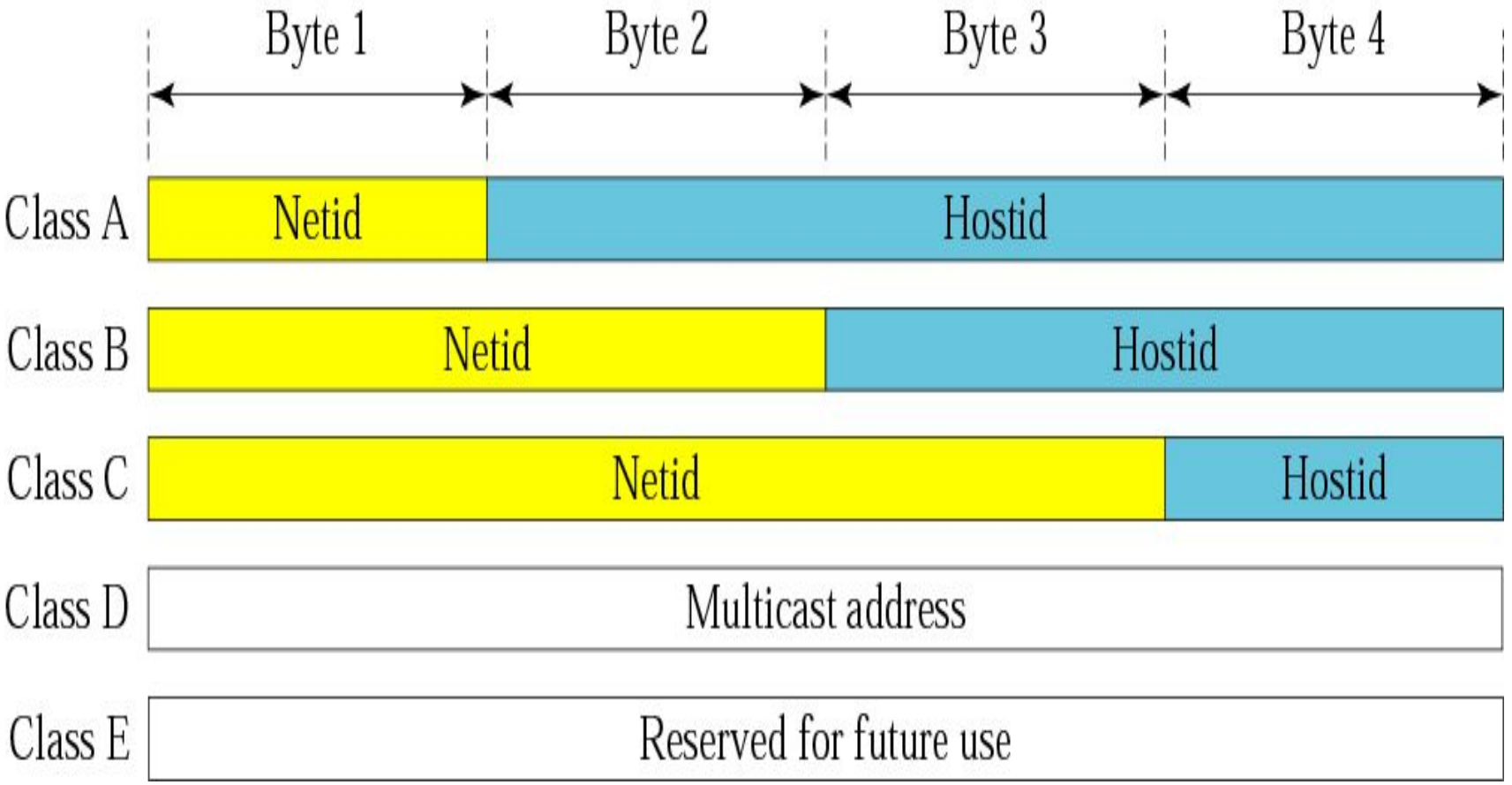
b. The first byte is 193 (between 192 and 223); the class is C.

c. The first byte is 14 (between 0 and 127); the class is A.

d. The first byte is 252 (between 240 and 255); the class is E.

e. The first byte is 134 (between 128 and 191); the class is B.

Netid and hostid



Example

Given the network address 17.0.0.0, find the class, the block, and the range of the addresses.

Solution

The class is A because the first byte is between 0 and 127.

The block has a netid of 17.

The addresses range from 17.0.0.0 to 17.255.255.255.

Example

Given the network address 132.21.0.0, find the class, the block, and the range of the addresses.

Solution

The class is B because the first byte is between 128 and 191.

The block has a netid of 132.21.

The addresses range from 132.21.0.0 to 132.21.255.255.

Example

Given the network address 220.34.76.0, find the class, the block, and the range of the addresses.

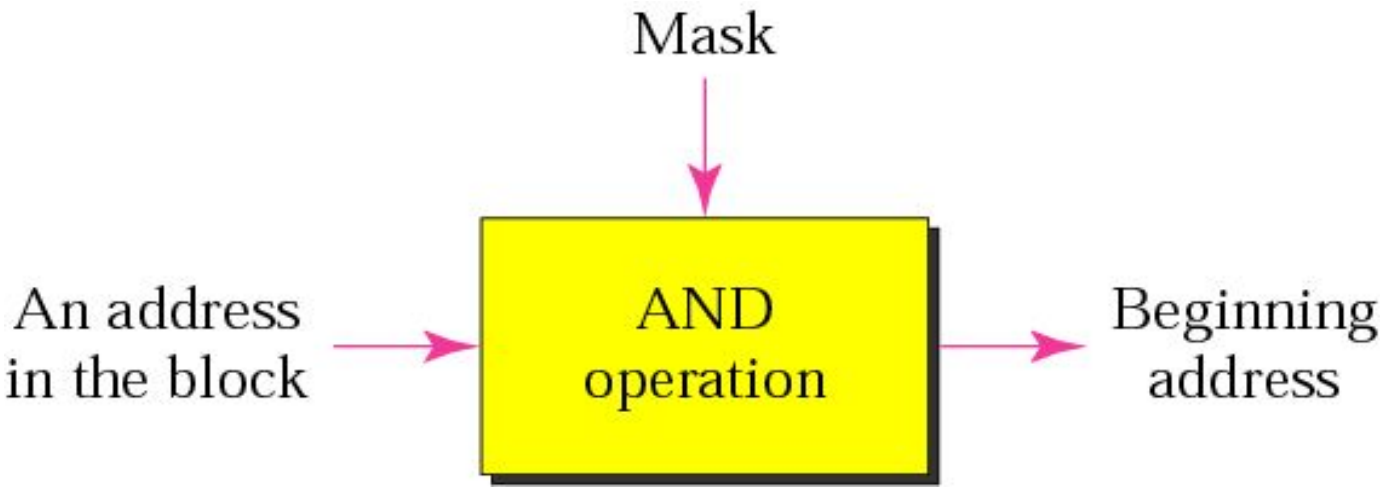
Solution

The class is C because the first byte is between 192 and 223.

The block has a netid of 220.34.76.

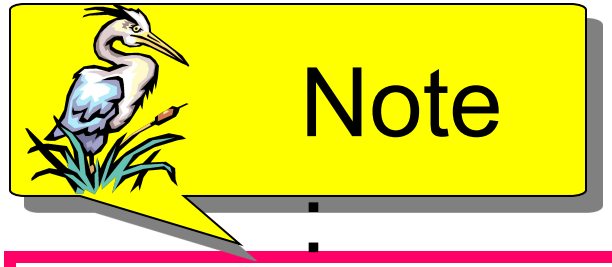
The addresses range from 220.34.76.0 to 220.34.76.255.

Masking concept



Default masks

Class	*Mask in binary*	*Mask in dotted-decimal*
A	**11111111** 00000000 00000000 00000000	**255.**0.0.0
B	**11111111 11111111** 00000000 00000000	**255.255.**0.0
C	**11111111 111111111 11111111** 00000000	**255.255.255.**0



The network address is the beginning address of each block. It can be found by applying the default mask to any of the addresses in the block (including itself). It retains the netid of the block and sets the hostid to zero.

Example

Given the address 23.56.7.91, find the beginning address (network address).

Solution

The default mask is 255.0.0.0,

which means that only the first byte is preserved

and the other 3 bytes are set to 0s.

The network address is 23.0.0.0.

Example

Given the address 132.6.17.85, find the beginning address (network address).

Solution

The default mask is 255.255.0.0,

which means that the first 2 bytes are preserved

and the other 2 bytes are set to 0s.

The network address is 132.6.0.0.

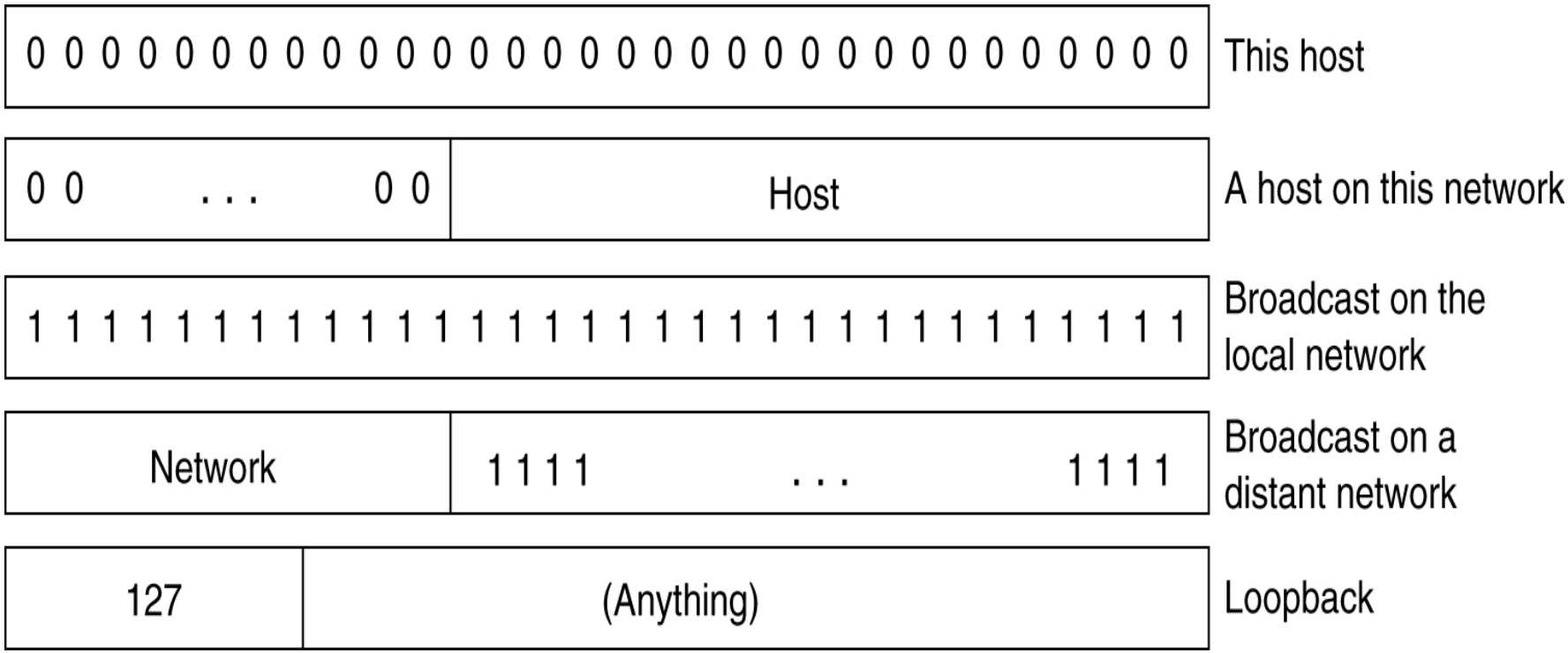
Example

Given the address 201.180.56.5, find the beginning address (network address).

Solution

The default mask is 255.255.255.0, which means that the first 3 bytes are preserved and the last byte is set to 0. The network address is 201.180.56.0.

Special IP addresses

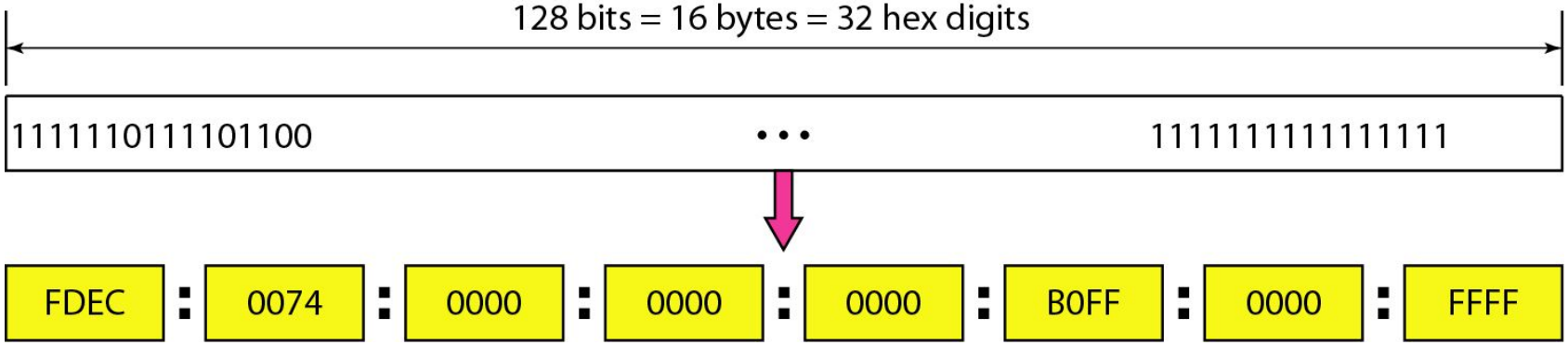


IPv6 ADDRESSES

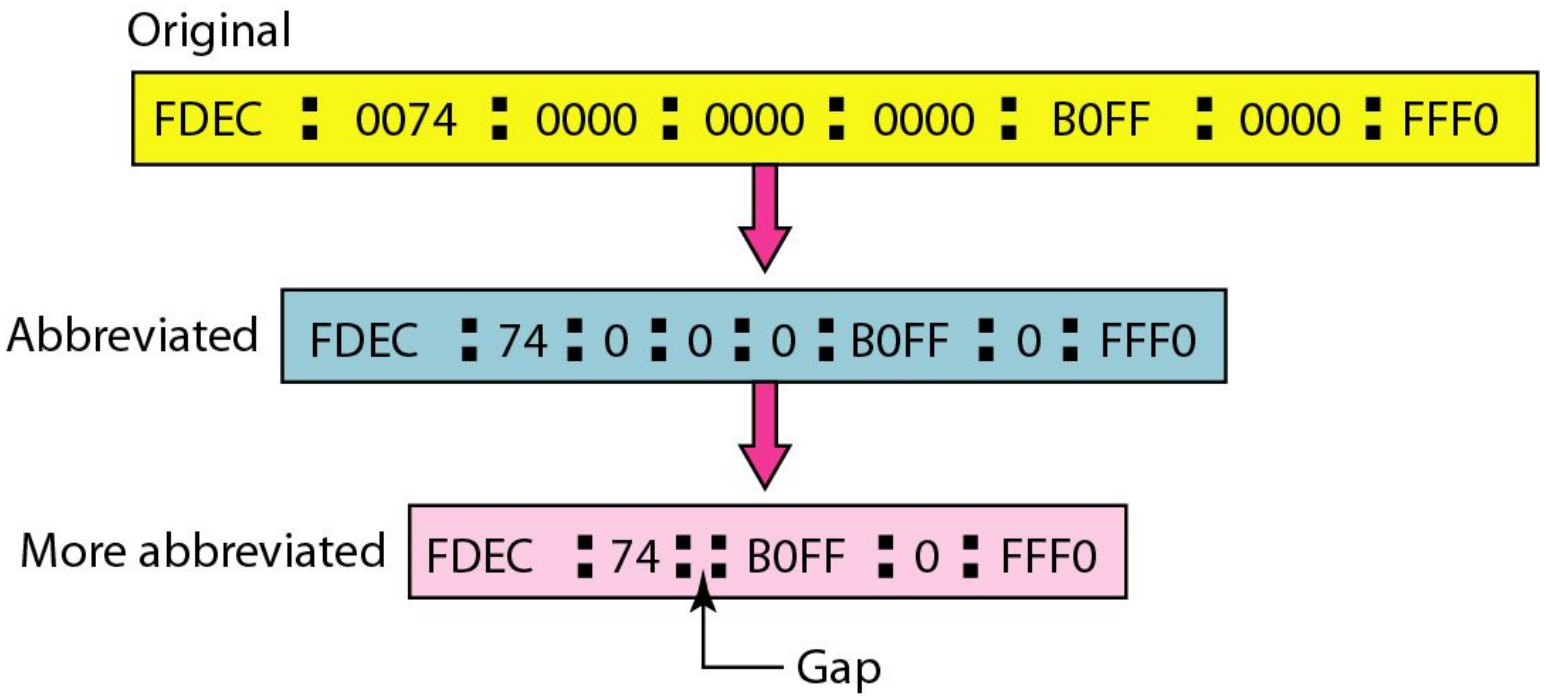
Despite all short-term solutions, address depletion is still a long-term problem for the Internet. This and other problems in the IP protocol itself have been the motivation for IPv6.

An IPv6 address is 128 bits long.

IPv6 address in binary and hexadecimal colon notation



Abbreviated IPv6 addresses



IPv6 Colon Hexadecimal Notation

- 128 bit number expressed as dotted decimal

  104.230.140.100.255.255.255.255.0.0.17.128.150.10.255.255 becomes

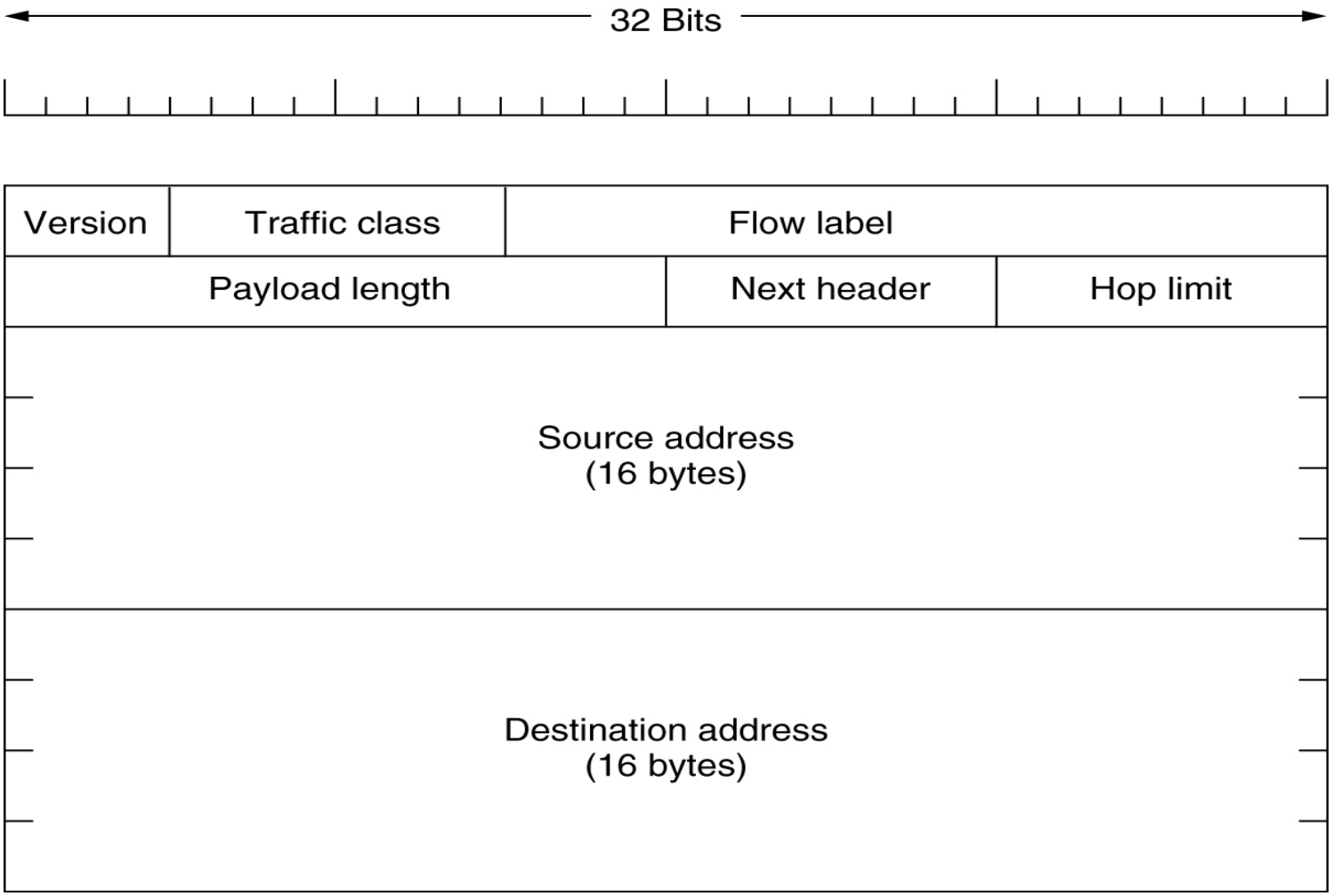
  68E6:8C64:FFFF:FFFF:0:1180:96A:FFFF

- **Hex notation allows zero compression**
  - A string of repeated zeros is replaced with a pair of colons
  - FF05:0:0:0:0:0:0:B3 becomes FF05::B3
  - **Can be applied only once in any address**

Basic IPv6 Address Types

- **Unicast** – Destination address specifies a single computer. Route datagram along shortest path.
- **Anycast** – Destination is a set of computers, possibly at different locations, that all share a single address. Route datagram along shortest path and deliver to exactly one member of the group (i.e. closest member)
- **Multicast** - Destination is a set of computers, possibly at different locations. One copy of the datagram will be delivered to each member of the group using hardware multicast or broadcast if viable.

The Main IPv6 Header



The IPv6 fixed header (required).

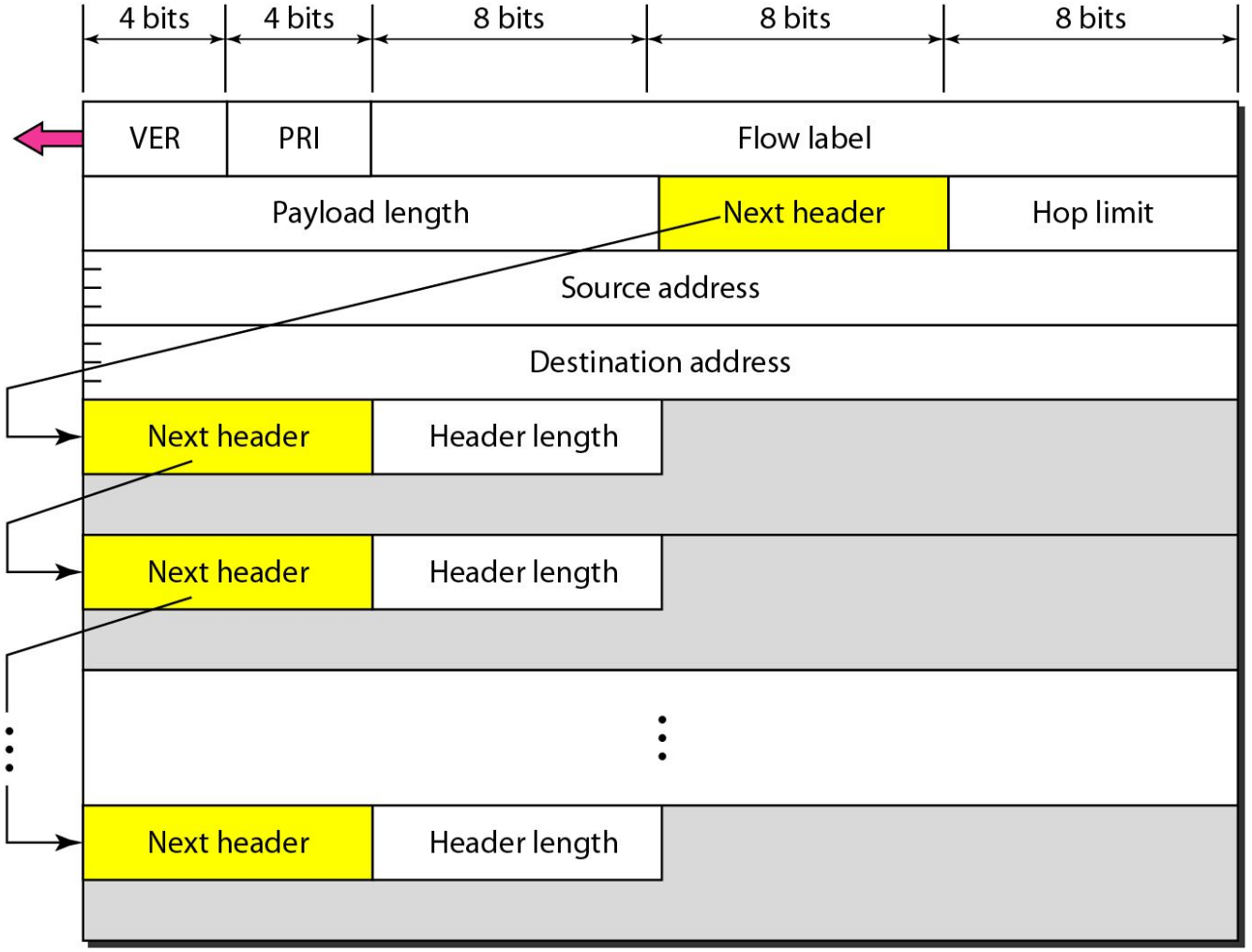
IPV6 Header Description

- **Version** (4-bits): It represents the version of Internet Protocol
- **Traffic Class** (8-bits): These 8 bits are divided into two parts. The most significant 6 bits are used for Type of Service & The least significant 2 bits are used for Explicit Congestion Notification (ECN).
- **Flow Label** (20-bits): This label is used to maintain the sequential flow of the packets belonging to a communication. **Payload Length** (16-bits): This field is used to tell the routers how much information a particular packet contains in its payload.

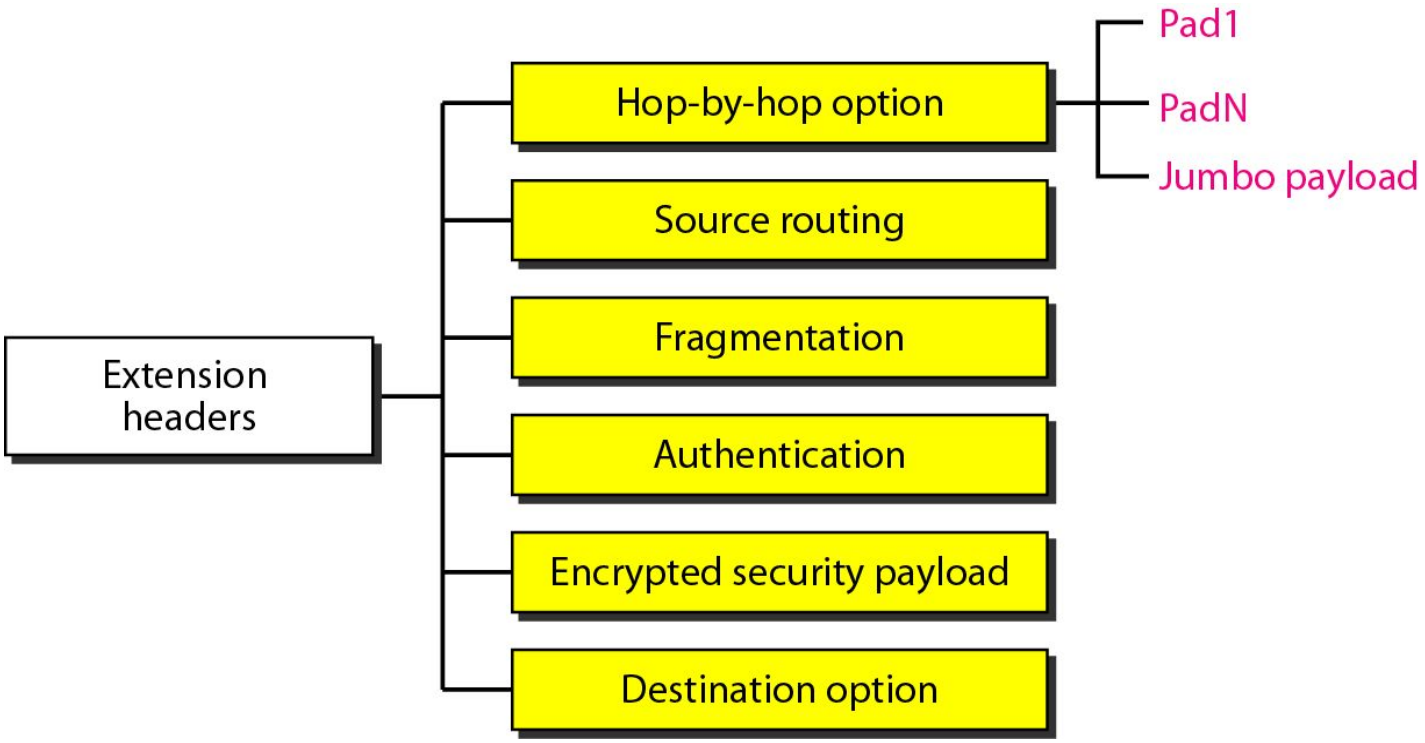
IPV6 Header Description

- **Next Header** (8-bits): This field is used to indicate either the type of Extension Header.
- **Hop Limit** (8-bits): This field is used to stop packet to loop in the network infinitely. The value of Hop Limit field is decremented by 1 as it passes a link (router/hop). When the field reaches 0 the packet is discarded.
- **Source Address** (128-bits): This field indicates the address of originator of the packet.
- **Destination Address** (128-bits): This field provides the address of intended recipient of the packet.

Format of an IPv6 datagram



Extension header types



Advantages of IPv6 over IPv4(Ipv4 v/s Ipv6)

Feature	IPv4	IPv6
Source and destination address	32 bits	128 bits
Address Format	Dotted Decimal	Hexadecimal Notation
No of Address	2^32	2^128
IPSec	Optional	required
Payload ID for QoS in the header	No identification	Using Flow label field
Fragmentation	Both router and the sending hosts	Only supported at the sending hosts
Header checksum	included	Not included
Resolve IP address to a link layer address	broadcast ARP request	Multicast Neighbor Solicitation message

Advantages of IPv6 over IPv4 (Ipv4 v/s Ipv6) (2)

Feature	IPv4	IPv6
Determine the address of the best default gateway	ICMP Router Discovery(optional)	ICMPv6 Router Solicitation and Router Advertisement (required)
Send traffic to all nodes on a subnet	Broadcast	Link-local scope all-nodes multicast address
Configure address	Manually or DHCP	Autoconfiguration
Manage local subnet group membership	(IGMP)	Multicast Listener Discovery (MLD)

Outline

- Switching techniques,
- IP Protocol,
- IPv4 and IPv6 addressing schemes,
- **Subnetting,**
- **NAT, CIDR,**

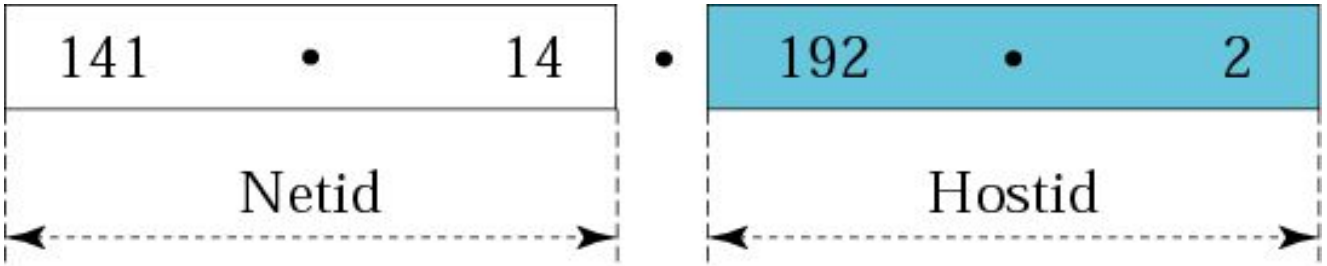
IPv4 Addressing- **Subnetting**

The problems associated with classful addressing is that the network addresses available for assignment to organizations are close to depletion.

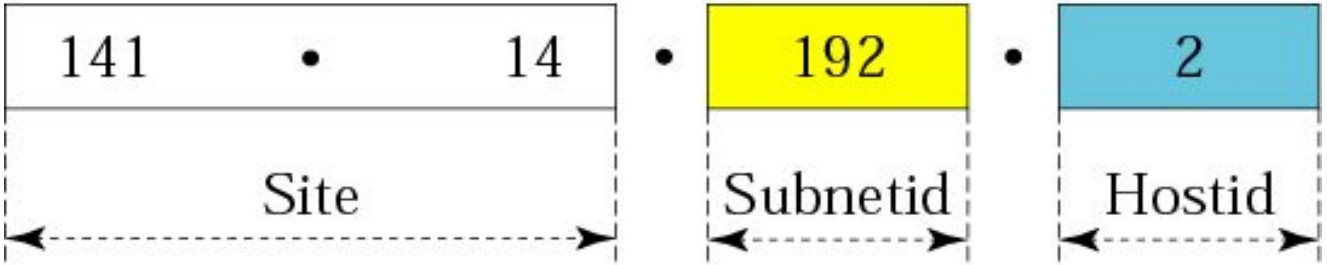
This is coupled with the ever-increasing demand for addresses from organizations that want connection to the Internet.

In this section we briefly discuss two solutions: subnetting and supernetting.

Addresses in a network with and without subnetting

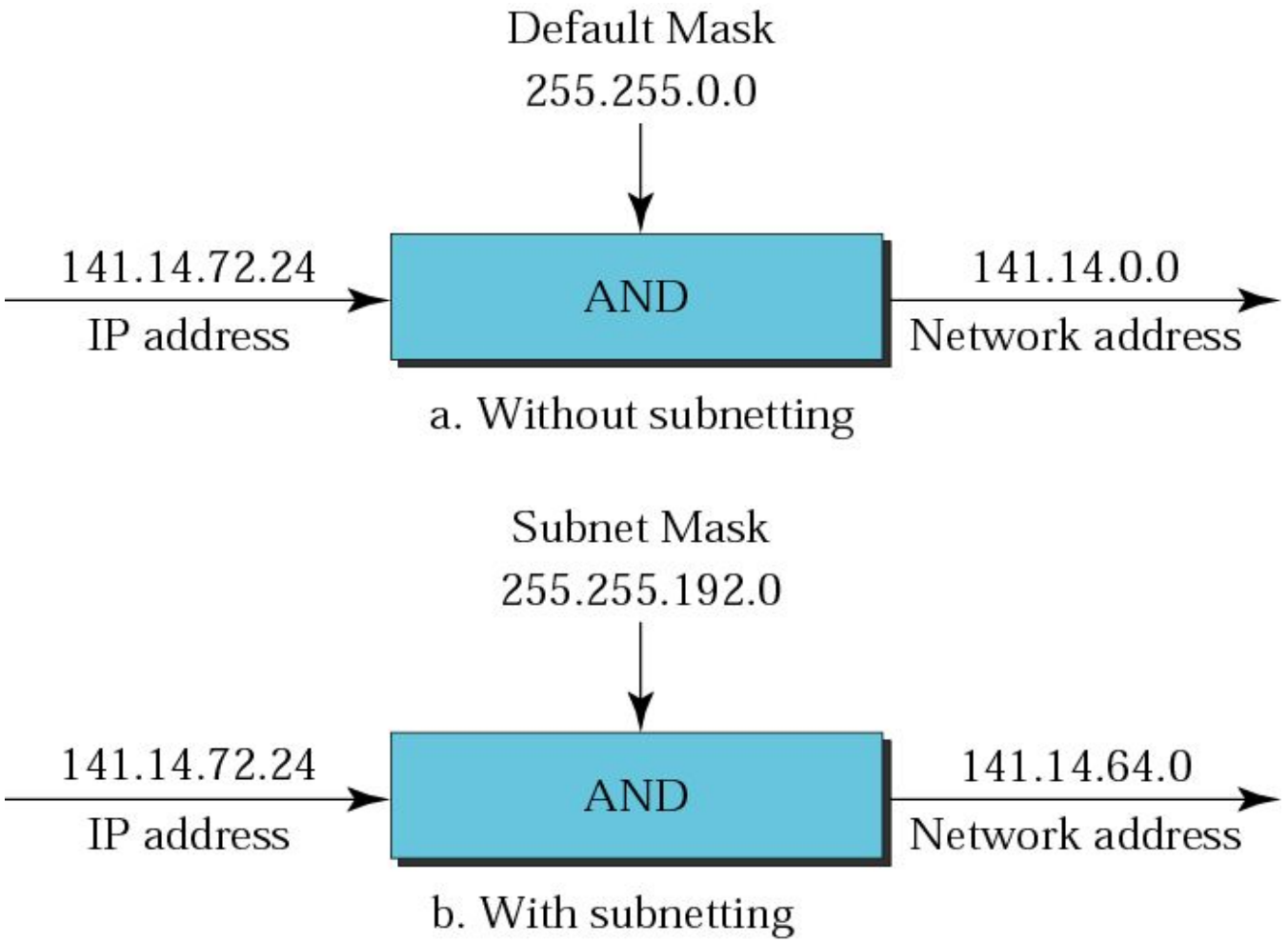


a. Without subnetting

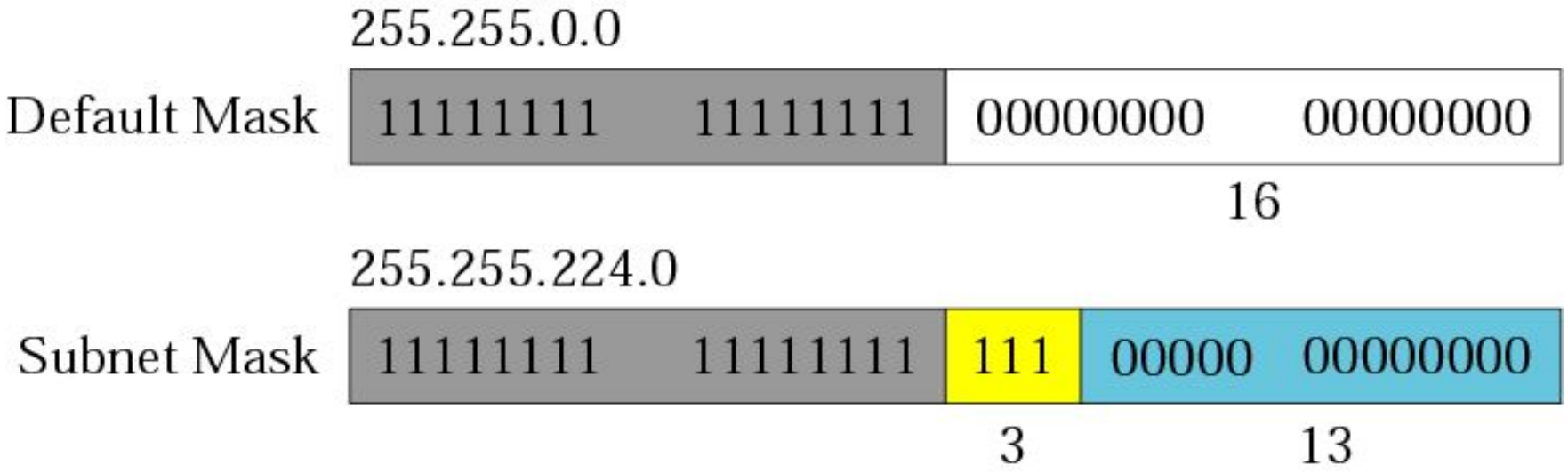


b. With subnetting

Default mask and subnet mask



Comparison of a default mask and a subnet mask



For more examples refer

- https://www.kirkwood.edu/pdf/uploaded/569/ip_addressing_&_subnetting_workbook.pdf
- http://www.routeralley.com/guides/ipv4.pdf

PPTS from NPTEL

- http://www.facweb.iitkgp.ernet.in/~isg/INTERNET/SLIDES/Lecture-06.pdf

Example

What is the subnetwork address if the destination address is 200.45.34.56 and the subnet mask is 255.255.240.0?

Solution

We apply the AND operation on the address and the subnet mask.

Address	➡ *11001000 00101101 00100010 00111000*
Subnet Mask	➡ *11111111 11111111 11110000 00000000*
Subnetwork Address	➡ *11001000 00101101 00100000 00000000.*

Style -1 ***Subnetting when given a required number of networks***

Example 1: A service provider has given you the Class C network range 209.50.1.0. Your company must break the network into 20 separate subnets.

Solution

Step 1) **Determine the number of subnets and convert to binary**

-In this example, the binary representation of 20 = 00010100.

Step 2) **Reserve required bits in subnet mask and find incremental value**

- The binary value of 20 subnets tells us that we need at least 5 network bits to satisfy this requirement

Example 1 continued

- Our original subnet mask is 255.255.255.0 (Class C subnet) - The full binary representation of the subnet mask is as follows:
255.255.255.0 = 11111111.11111111.11111111.00000000

- We must “convert” 5 of the client bits (0) to network bits (1) in order to satisfy the requirements:
New Mask = 11111111.11111111.11111111.11111000

-If we convert the mask back to decimal, we now have the subnet mask that will be used on all the new
networks – 255.255.255.248 –

New subnet mask 255.255.255.248

Our increment bit is the last possible network bit, converted back to a binary number:

New Mask = 11111111.11111111.11111111.1111(1)000 –

bit with the parenthesis is your increment bit.

If you convert this bit to a decimal number, it becomes the number “8” that is every subnet is having 8 addresses allotted to it (from 0 to 7, then 8 to 15 etc)

Step 3) **Use increment to find network ranges**

You can now fill in your end ranges, which is the last possible IP address before you start the next range
209.50.1.0 – 209.50.1.7
209.50.1.8 – 209.50.1.15
209.50.1.16 – 209.50.1.23 …etc

You can then assign these ranges to your networks!

Remember the first and last address from each range (network / broadcast IP) are unusable

Style 2- ***Subnetting when given a required number of clients***

Example 1: A service provider has given you the Class C network range 209.50.1.0. Your company must break the network into as many subnets as possible as long as there are *at least 50 clients per network.*

Solution

Step 1) **Determine the number of clients and convert to binary**

-In this example, the binary representation of 50 = 00110010

-Step 2) **Reserve required bits in subnet mask and find incremental value**

- The binary value of 50 clients tells us that we need at least 6 client bits to satisfy this requirement

Example 2 continued

- Our original subnet mask is 255.255.255.0 (Class C subnet) - The full binary representation of the subnet mask is as follows:
255.255.255.0 = 11111111.11111111.11111111.00000000

-We must ensure 6 of the client bits (0) *remain client bits (save the clients!) in order to satisfy the* requirements. All other bits can become network bits:
-New Mask = 11111111.11111111.11111111.11 000000
- **note the 6 client bits that we have saved**

-If we convert the mask back to decimal, we now have the subnet mask that will be used on all the new networks –
255.255.255.192

Example 2 continued

New subnet Mask - 255.255.255.192

Our increment bit is the last possible network bit, converted back to a binary number:

New Mask = 11111111.11111111.11111111.1(1)000000

– bit with the parenthesis is your increment bit.

If you convert this bit to a decimal number, it becomes the number “64” (i.e from 0 to 63, 64 to 127 etc)

Step 3) **Use increment to find network ranges**

209.50.1.0 – 209.50.1.63
209.50.1.64 – 209.50.1.127
209.50.1.128 – 209.50.1.191
209.50.1.192 – 209.50.1.255

You can then assign these ranges to your networks!

Remember the first and last address from each range (network / broadcast IP) are unusable

Style 3 - **Given an IP address & Subnet Mask, find original network range**

Example - You are given the following IP address and subnet mask: 192.168.1.58 255.255.255.240 Identify the original range of addresses (the subnet) that this IP address belongs to

Solution

Break the subnet mask back into binary

255.255.255.240 = 11111111.11111111.11111111.11110000

-As before, the last possible network bit is your increment.

-In this case, the increment is 16

-

-Use this increment to find the network ranges until you pass the given IP address:

192.168.1.0

192.168.1.16

192.168.1.32

192.168.1.48

192.168.1.64 (*passed given IP address 192.168.1.58)*

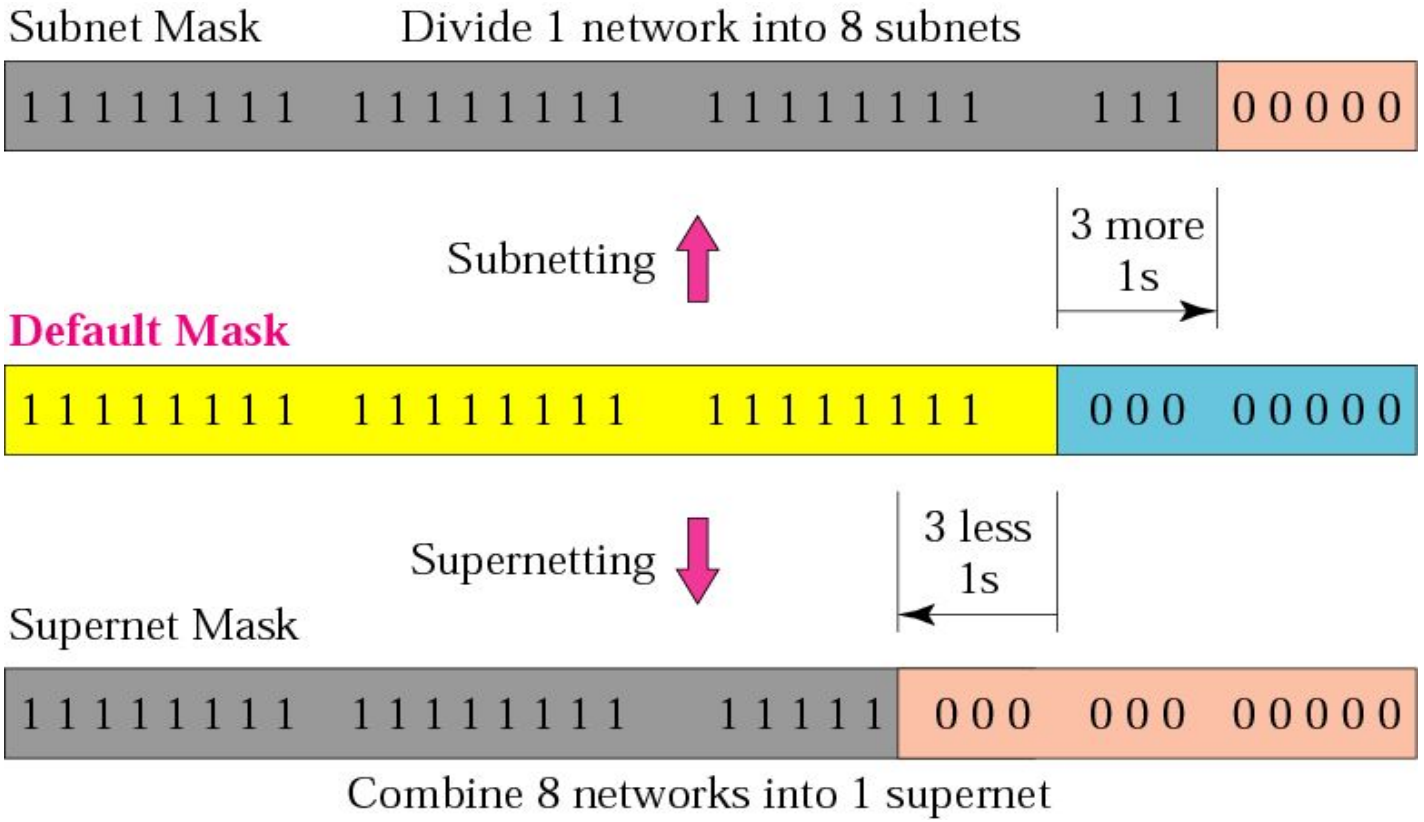
- Now, fill in the end ranges to find the answer to the scenario:

192.168.1.0 – 192.168.1.15
192.168.1.16 – 192.168.1.31
192.168.1.32 – 192.168.1.47
192.168.1.48 – 192.168.1.63
(*IP address 192.168.1.58 belongs to this range*)

In subnetting, we need the first address of the subnet and the subnet mask to define the range of addresses.

In supernetting, we need the first address of the supernet and the supernet mask to define the range of addresses.

Comparison of subnet, default, and supernet masks



IP Addresses:
Classless Addressing (CIDR- Classless Inter domain Routing)

Classless Addressing

It uses slash notation with IP Address

Example: 142.4.7.3/27

Here /27 means from total 31bit address first 27 bits are for Network and remaining i.e. 32-27=5 bits are for host

Prefix lengths

/n	Mask	/n	Mask	/n	Mask	/n	Mask
/1	128.0.0.0	/9	255.128.0.0	/17	255.255.128.0	/25	255.255.255.128
/2	192.0.0.0	/10	255.192.0.0	/18	255.255.192.0	/26	255.255.255.192
/3	224.0.0.0	/11	255.224.0.0	/19	255.255.224.0	/27	255.255.255.224
/4	240.0.0.0	/12	255.240.0.0	/20	255.255.240.0	/28	255.255.255.240
/5	248.0.0.0	/13	255.248.0.0	/21	255.255.248.0	/29	255.255.255.248
/6	252.0.0.0	/14	255.252.0.0	/22	255.255.252.0	/30	255.255.255.252
/7	254.0.0.0	/15	255.254.0.0	/23	255.255.254.0	/31	255.255.255.254
/8	255.0.0.0	/16	255.255.0.0	/24	255.255.255.0	/32	255.255.255.255

The addresses in color are the default masks for classes A, B, and C. Thus, classful addressing is a special case of classless addressing.

Example - Find first address

What is the first address in the block if one of the addresses is 167.199.170.82/27?

Example - Find first address

What is the first address(network address) in the block if one of the addresses is 167.199.170.82/27?

Solution

The prefix length is 27, which means that we must keep the first 27 bits as is and change the remaining bits (5) to 0s. The following shows the process:

Address in binary: 10100111 11000111 10101010 01010010
Keep the left 27 bits: 10100111 11000111 10101010 01000000
Result in CIDR notation: 167.199.170.64/27

A block of addresses is granted to a small organization. We know that one of the addresses is 205.16.37.39/28. What is the first address in the block?

Solution

The binary representation of the given address is

11001101 00010000 00100101 00100111

If we set 32−28 = 4 rightmost bits to 0, we get

***11001101 00010000 00100101 0010*0000**

or

205.16.37.32.

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

Example

Find the last address for the block in 205.16.37.39/28

Solution

The binary representation of the given address is

11001101 00010000 00100101 00100111

If we set 32 − 28=4 rightmost bits to 1, we get

11001101 00010000 00100101 00101111

or

205.16.37.47

The number of addresses in the block can be found by using the formula 2^{32-n}.

Example

Find the number of addresses in 205.16.37.39/28

Solution

The value of n is 28, which means that number of addresses is 2^{32-28} *or 16.*

Example

Another way to find the first address, the last address, and the number of addresses is to represent the mask as a 32-bit binary (or 8-digit hexadecimal) number. This is particularly useful when we are writing a program to find these pieces of information. In Example 19.5 the /28 can be represented as

11111111 11111111 11111111 11110000

(twenty-eight 1s and four 0s).

Find

a. The first address
b. The last address
c. The number of addresses.

Example

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

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

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

Example

Solution

.

Group 1

For this group, each customer needs 256 addresses. This means that 8 (log2 256) bits are needed to define each host. The prefix length is then 32 − 8 = 24. The addresses are

1st Customer:	*190.100.0.0/24*	*190.100.0.255/24*
2nd Customer:	*190.100.1.0/24*	*190.100.1.255/24*
. . .		
64th Customer:	*190.100.63.0/24*	*190.100.63.255/24*
Total = 64 × 256 = 16,384		

Example 19.10

Group 2

For this group, each customer needs 128 addresses. This means that 7 (log2 128) bits are needed to define each host. The prefix length is then 32 − 7 = 25. The addresses are

1st Customer:	*190.100.64.0/25*	*190.100.64.127/25*
2nd Customer:	*190.100.64.128/25*	*190.100.64.255/25*
. . .		
128th Customer:	*190.100.127.128/25*	*190.100.127.255/25*
Total = 128 × 128 = 16,384		

Example

Group 3

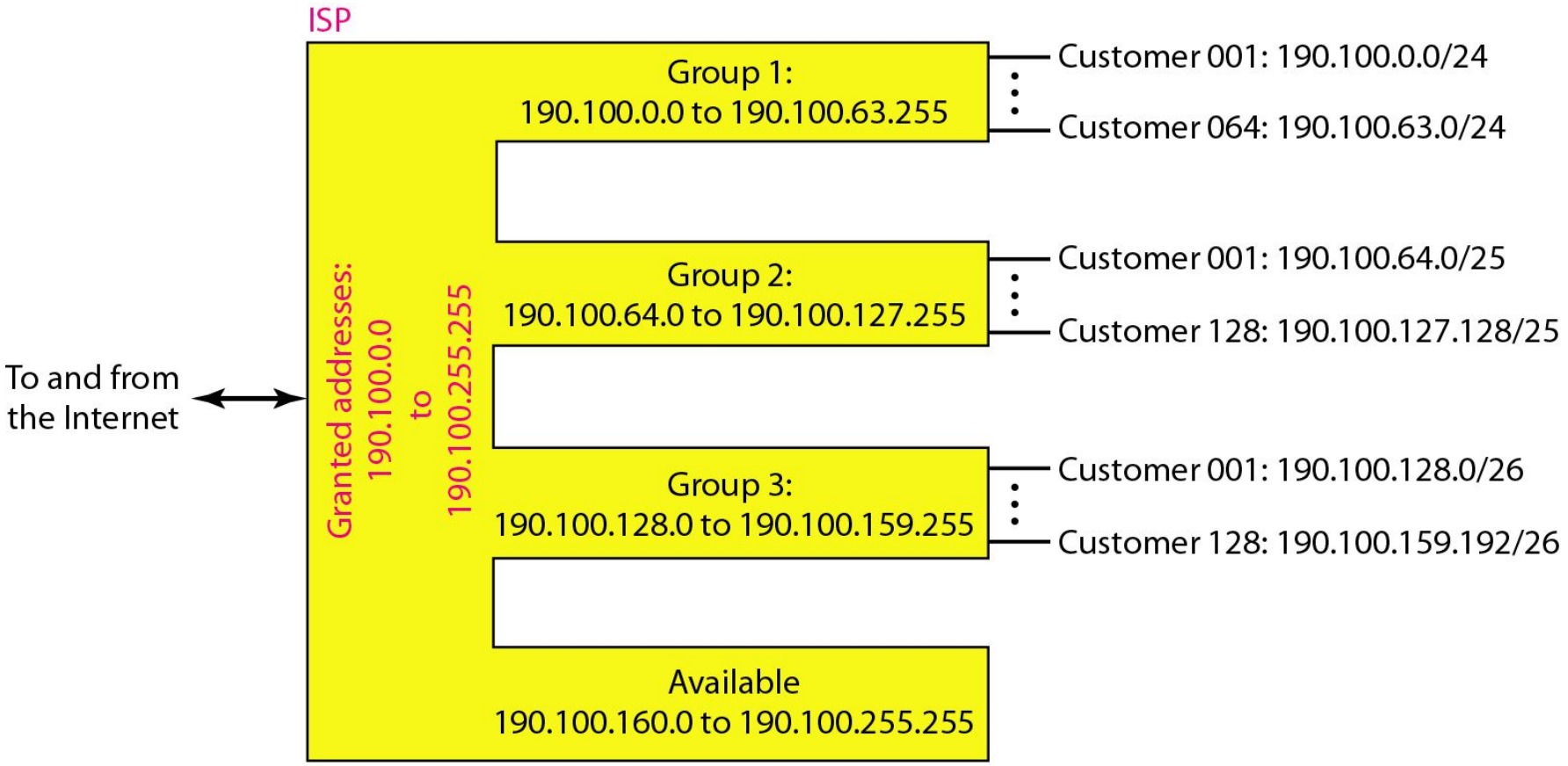
For this group, each customer needs 64 addresses. This means that 6 ($log_2 64$) bits are needed to each host. The prefix length is then 32 − 6 = 26. The addresses are

1st Customer:	*190.100.128.0/26*	*190.100.128.63/26*
2nd Customer:	*190.100.128.64/26*	*190.100.128.127/26*
. . .		
128th Customer:	*190.100.159.192/26*	*190.100.159.255/26*

Total = 128 × 64 = 8192

Number of granted addresses to the ISP: 65,536
Number of allocated addresses by the ISP: 40,960
Number of available addresses: 24,576

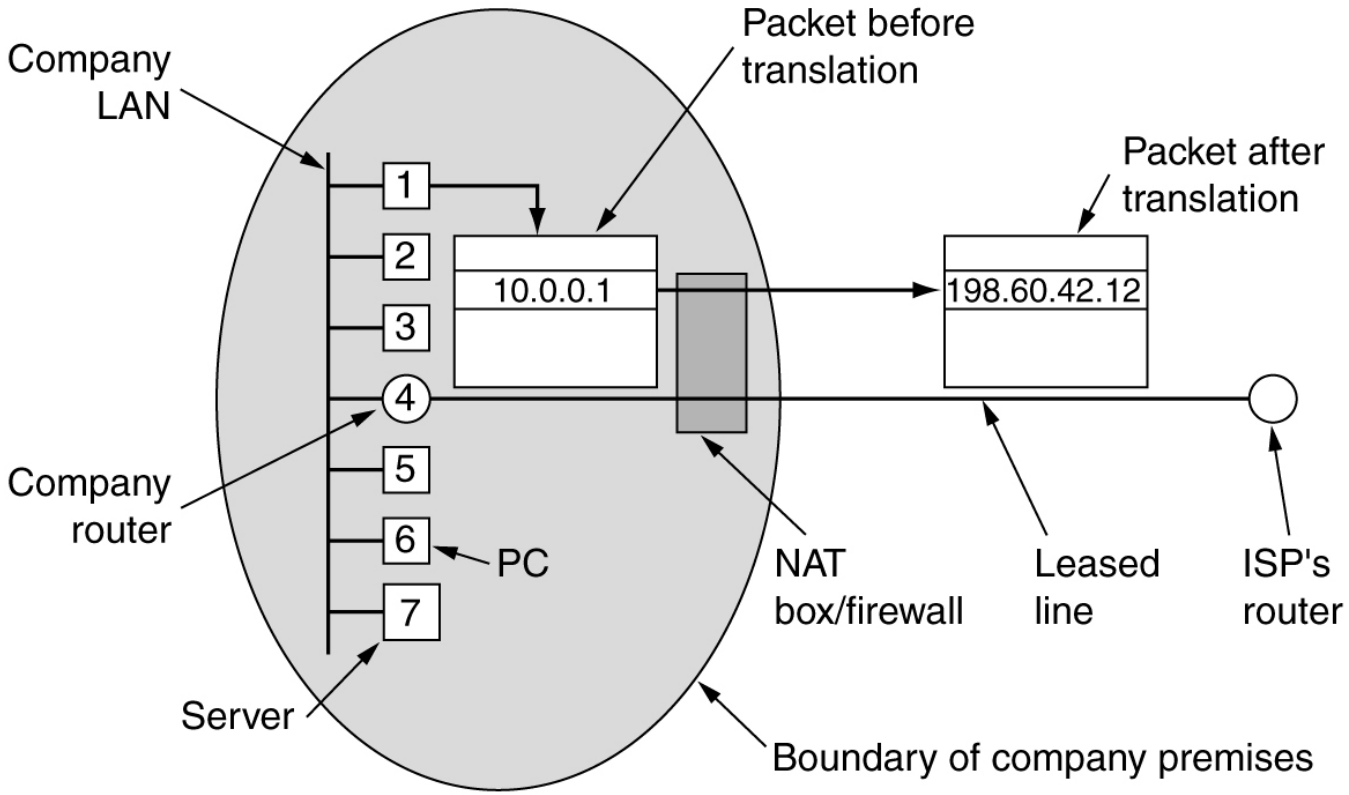
An example of address allocation and distribution by an ISP



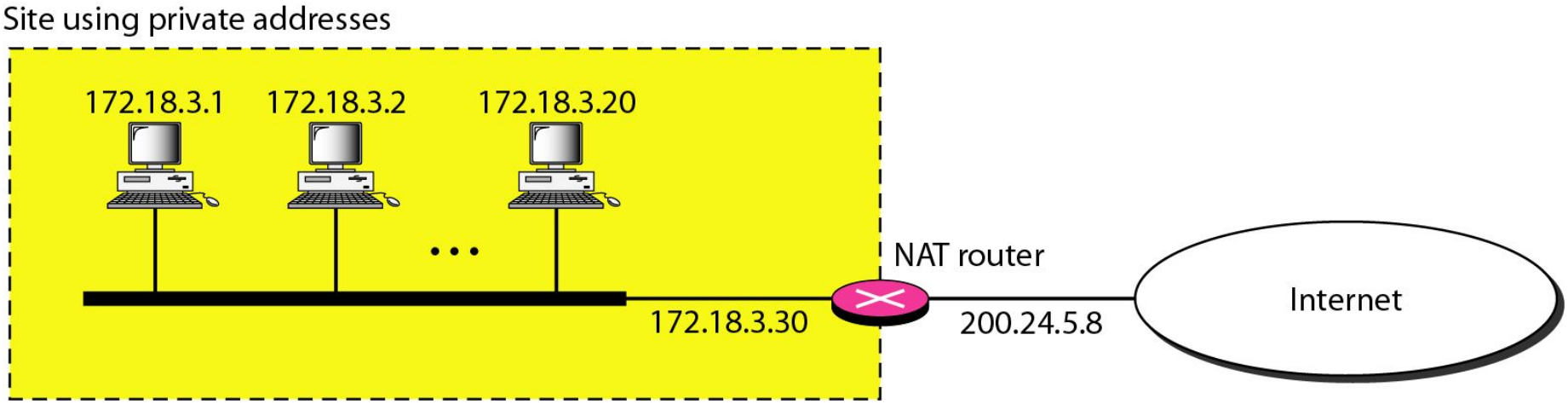
Addresses for private networks

Class	*Netids*	*Blocks*
A	10.0.0	1
B	172.16 to 172.31	16
C	192.168.0 to 192.168.255	256

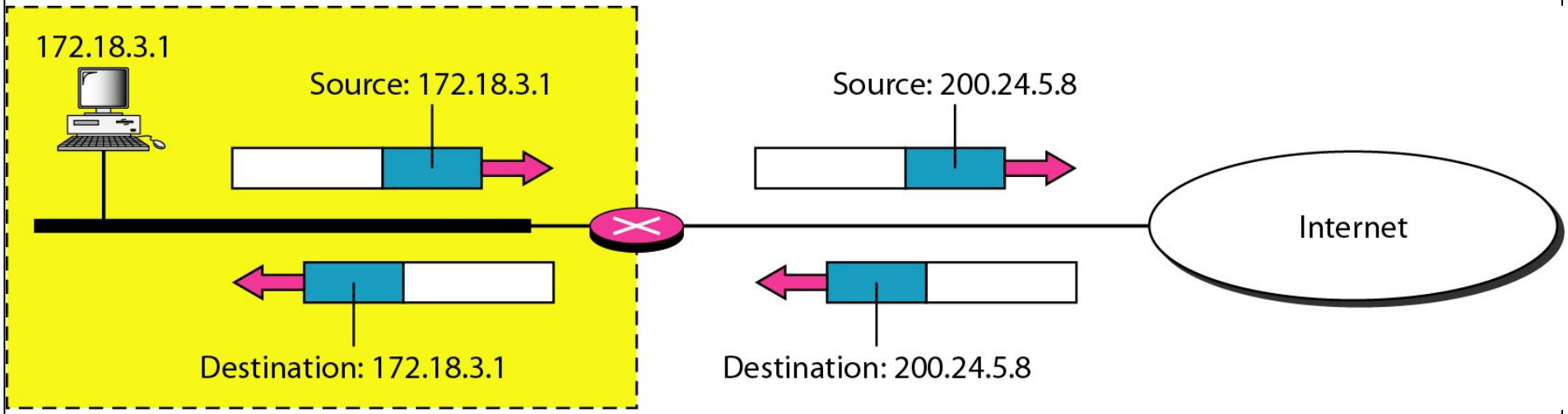
NAT – Network Address Translation



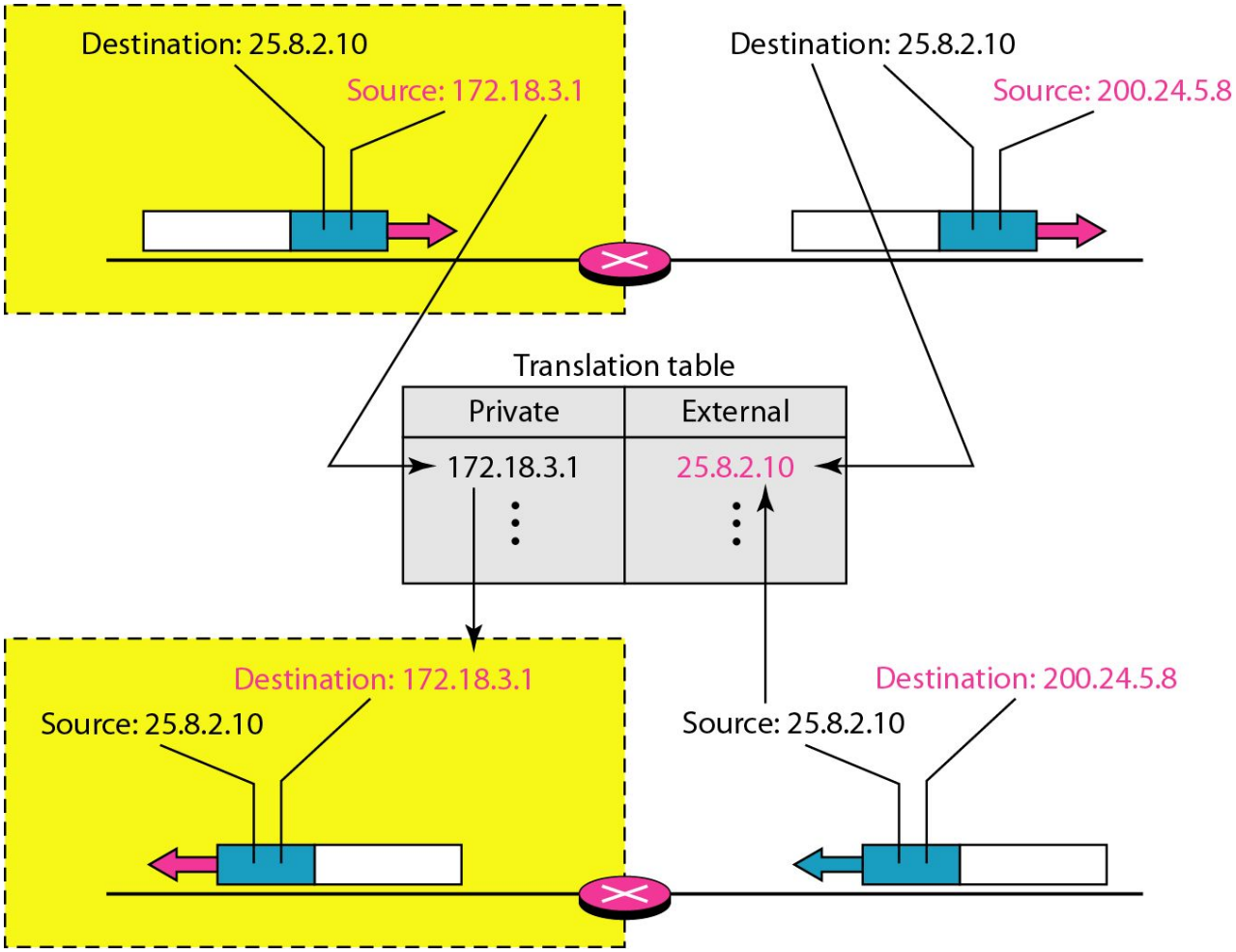
A NAT implementation



Addresses in a NAT



NAT address translation



Outline

- Switching techniques,
- IP Protocol,
- IPv4 and IPv6 addressing schemes,
- Subnetting,
- NAT, CIDR,
- **ICMP,**

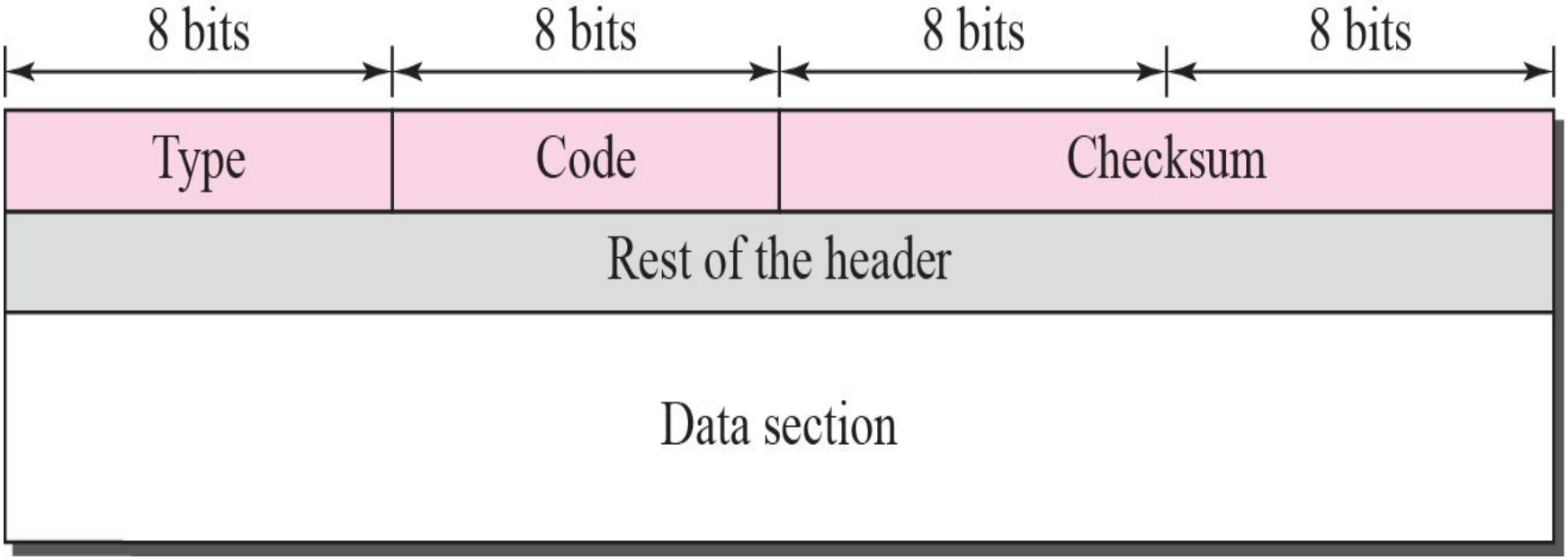
ICMP V4 -Introduction

- The IP protocol has no error-reporting or error correcting mechanism.
- What happens if something goes wrong? What happens if a router must discard a datagram because it cannot find a router to the final destination, or
- Because the time-to-live field has a zero value?
- These are examples of situations where an error has occurred and the IP protocol has no built-in mechanism to notify the original host.
- **The solution is ICMP protocol**

ICMP V4 -MESSAGES

- ICMP messages are divided into two broad categories:
  1. **error-reporting messages**
  2. **query messages.**
- The error-reporting messages report problems that a router or a host (destination) may encounter when it processes an IP packet.
- The query messages, help a host or a network manager get specific information from a router or another host. Also, hosts can discover and learn about routers on their network and routers can help a node redirect its messages.

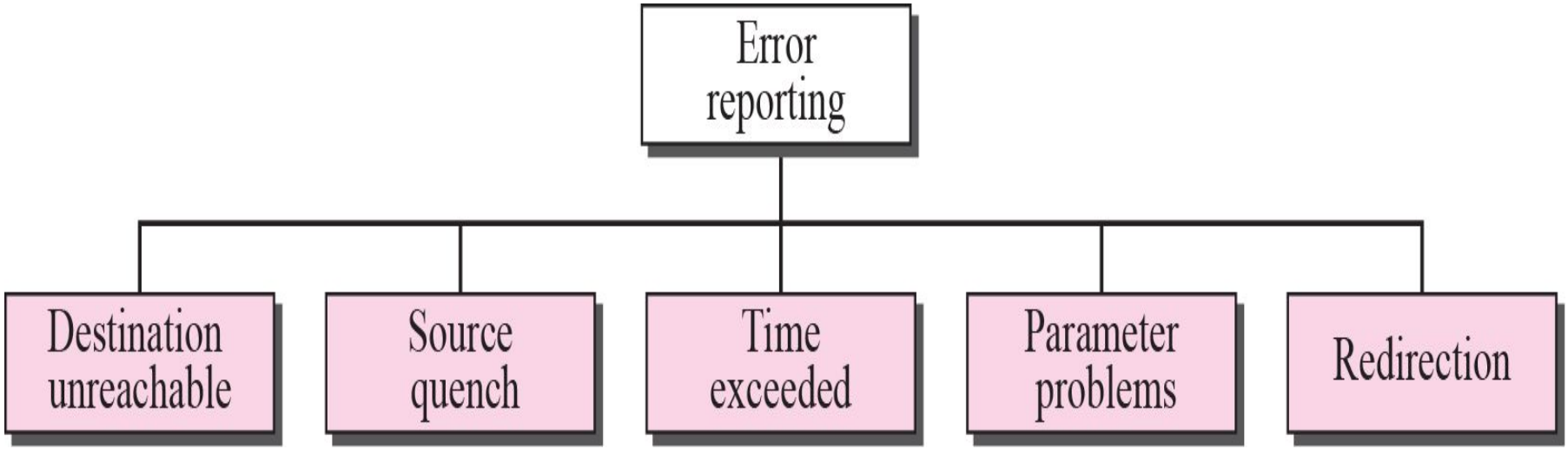
General format of ICMP messages or ICMP header



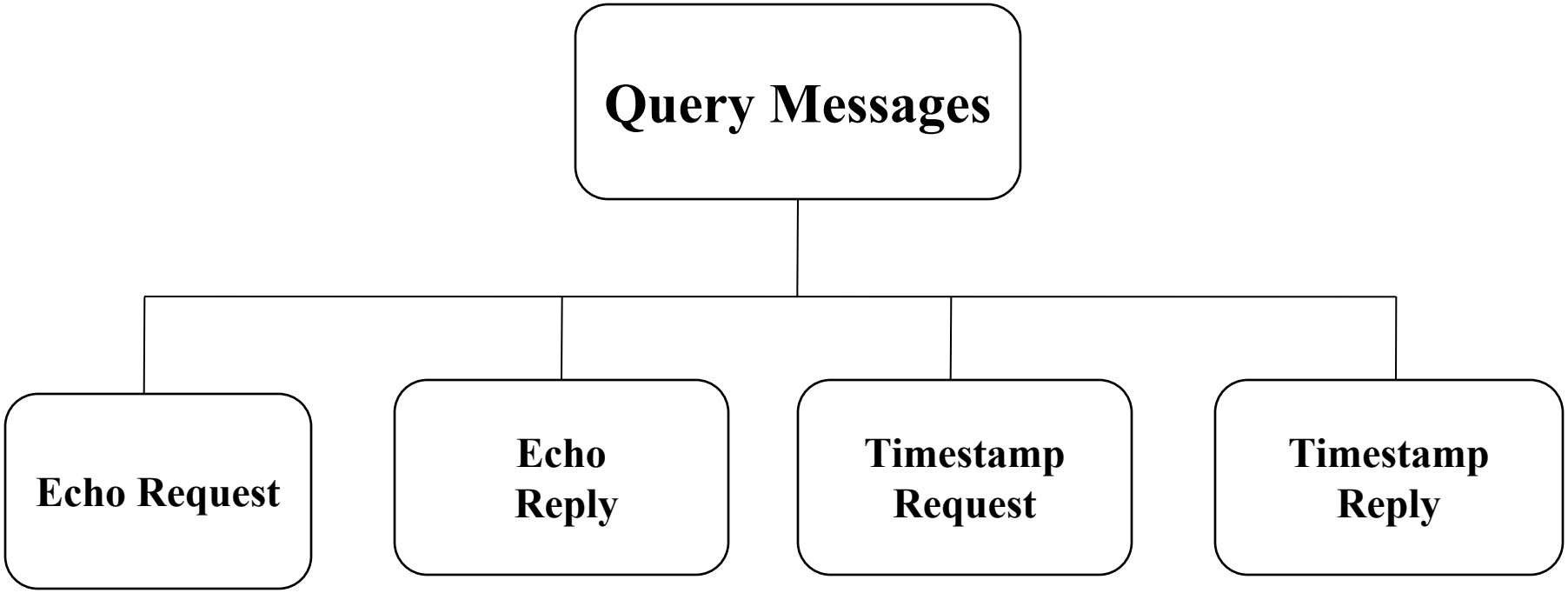
Basic ICMP Header

- Headers are 32 bits in length; all contain same three fields
  - type - 8 bit message type code
    - Thirteen message type are defined
  - code - 8 bit;
    - indicating why message is being sent
  - checksum - standard internet checksum
    - for purpose of calculation the checksum field is set to zero

Error-reporting messages



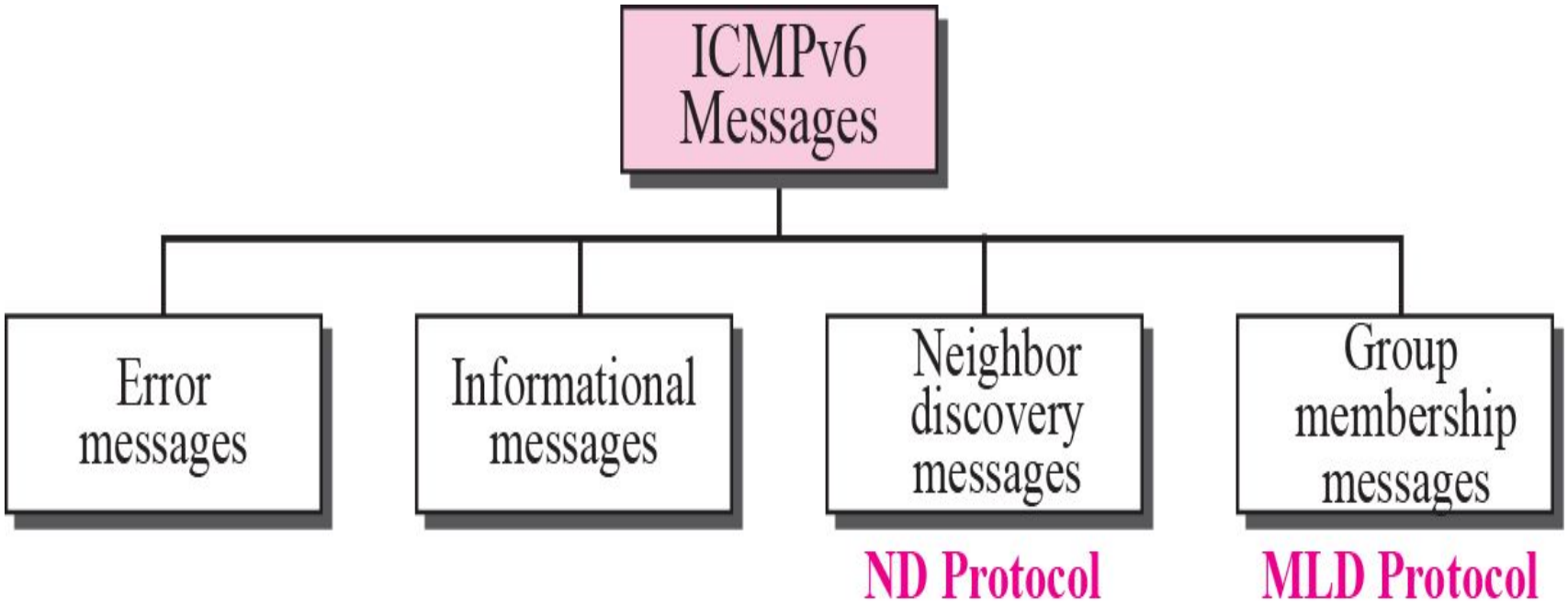
Query Messages



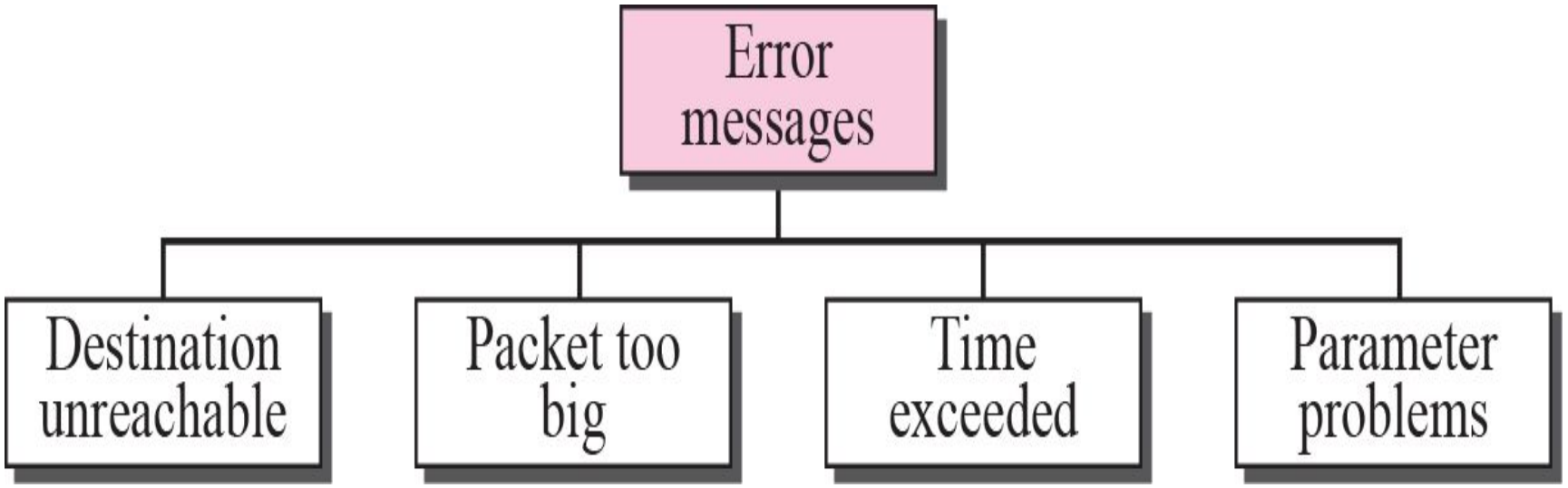
ICMP V6- INTRODUCTION

- Another protocol that has been modified in **version 6** of the TCP/IP protocol suite is ICMP.
- This new version, Internet Control Message Protocol version 6 (ICMPv6), follows the same strategy and purposes of version 4.
- ICMPv6, however, is more complicated than ICMPv4: some protocols that were independent in version 4 are now part of ICMPv6 and
- some new messages have been added to make it more useful.

Taxonomy of ICMPv6 messages



Error-reporting messages



Informational Messages

- Two of the ICMPv6 messages can be categorized as informational messages: echo request and echo reply messages.
- The echo request and echo response messages are designed to check if two devices in the Internet can communicate with each other.
- A host or router can send an echo request message to another host; the receiving computer or router can reply using the echo response message.

Neighbor-Discovery Messages

- The most important issue is the definition of two new protocols that clearly define the functionality of these group messages:
  1. Neighbor-Discovery (ND) protocol
  2. Inverse-Neighbor-Discovery (IND) protocol.
- These two protocols are used by nodes (hosts or routers) on the same link (network).

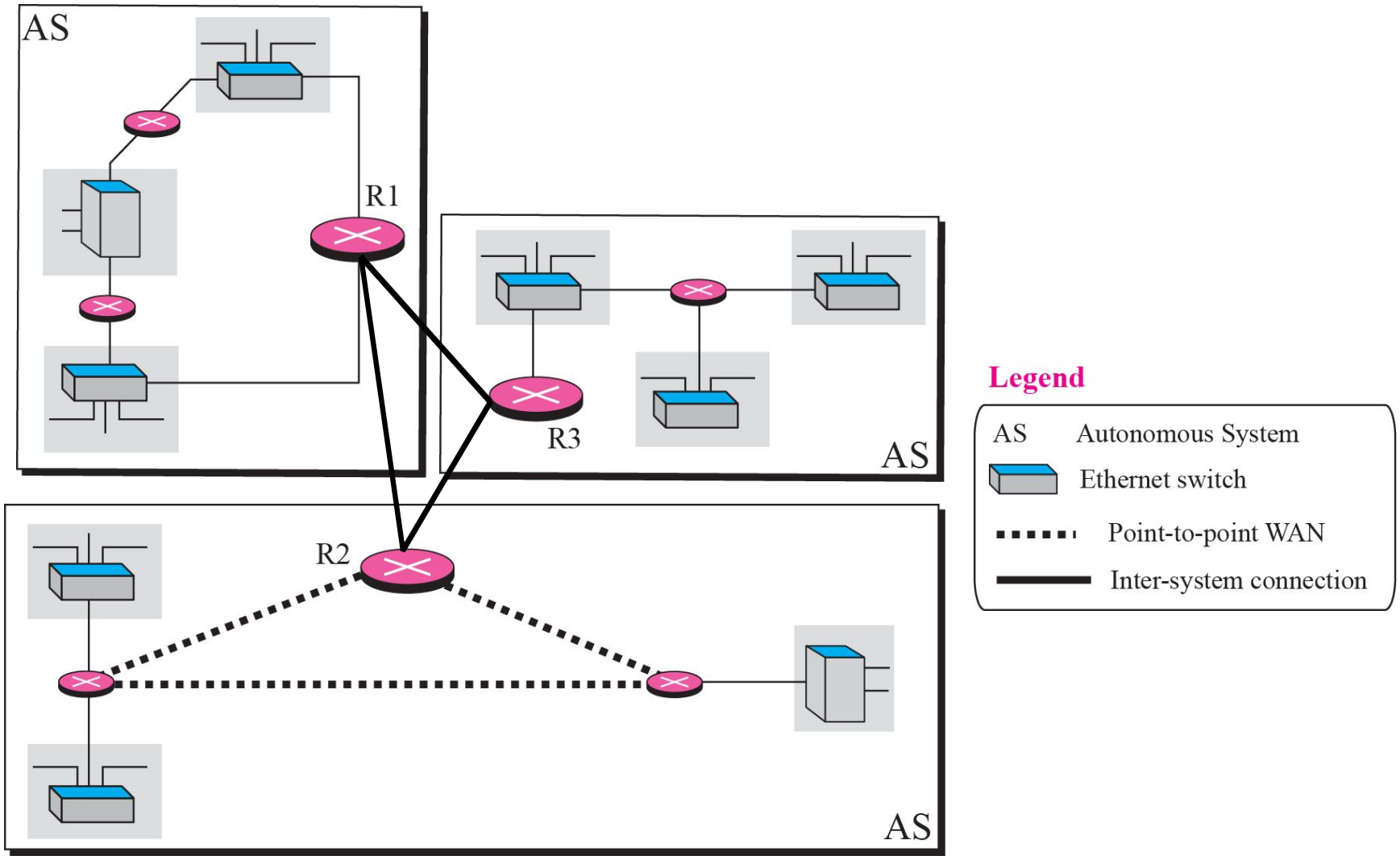
Outline

- Switching techniques,
- IP Protocol,
- IPv4 and IPv6 addressing schemes,
- Subnetting,
- NAT, CIDR,
- ICMP,
- **Routing Protocols: Distance Vector, Link State, Path Vector.**
- **Routing in Internet: RIP ,OSPF, BGP,**

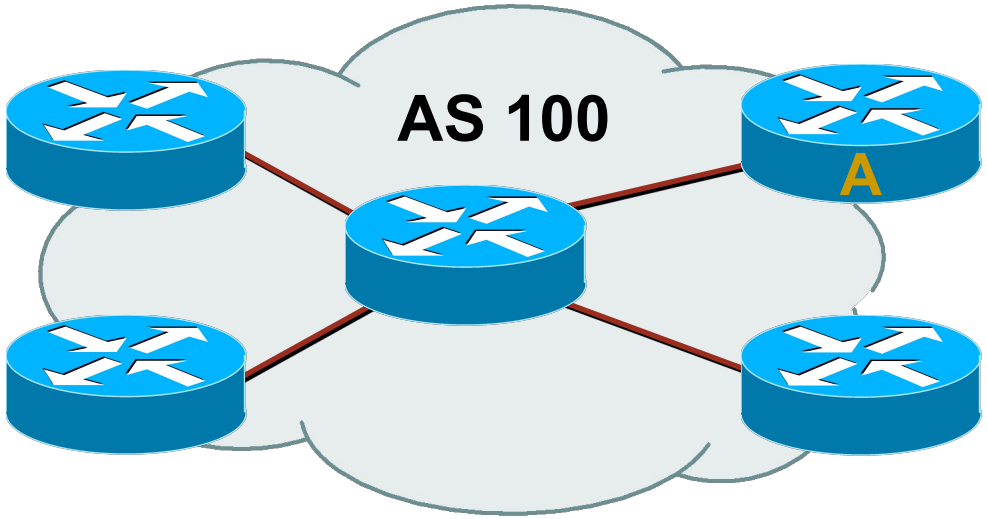
INTER-AND INTRA-DOMAIN ROUTING

- Today, an internet can be so large that one routing protocol cannot handle the task of updating the routing tables of all routers.
- For this reason, an internet is divided into autonomous systems.
- An autonomous system (AS) is a group of networks and routers under the authority of a single administration.
- Routing inside an autonomous system is called intra-domain routing.
- Routing between autonomous systems is called inter-domain routing

Figure ***Autonomous systems***

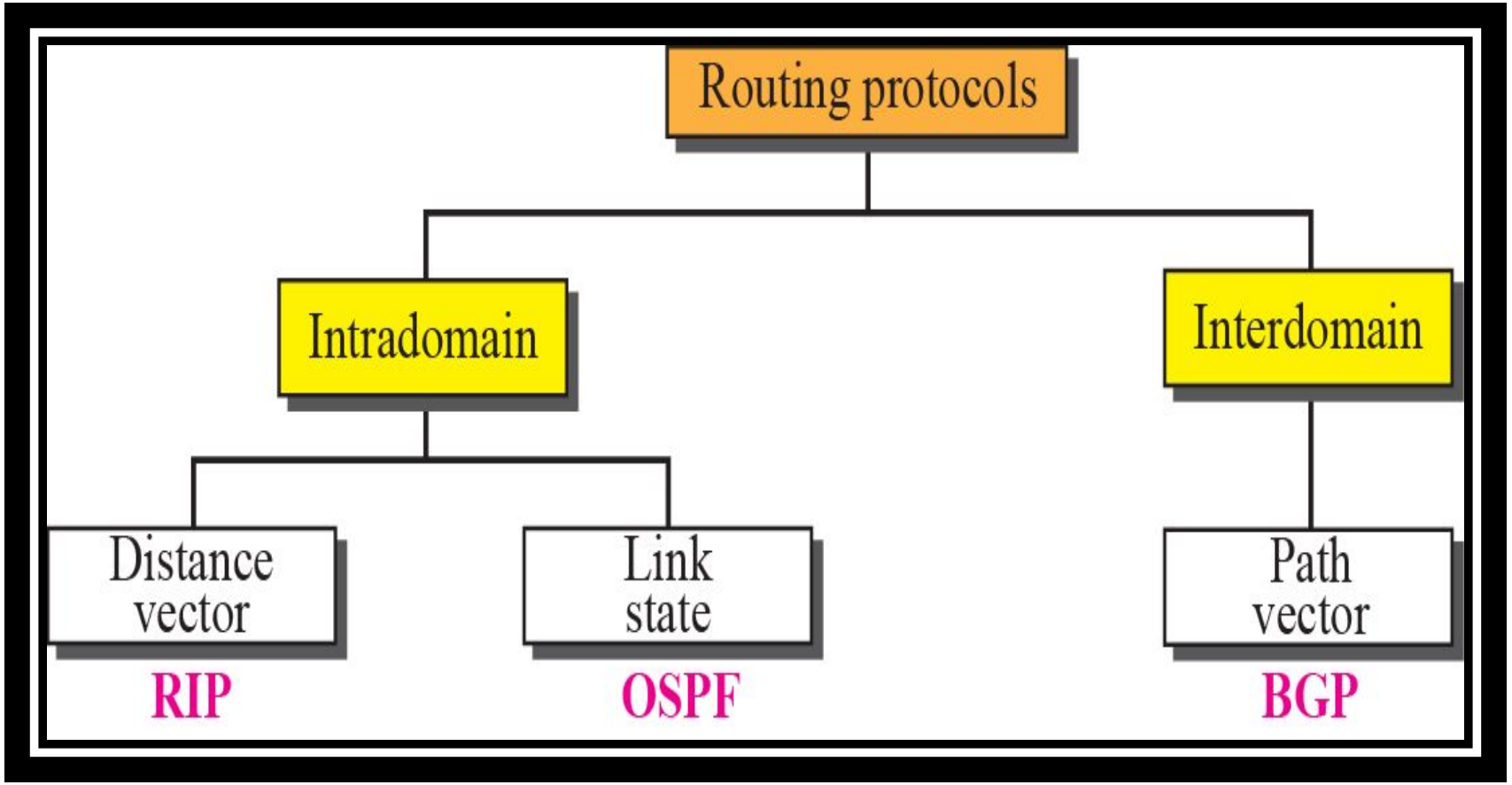


Autonomous System (AS)

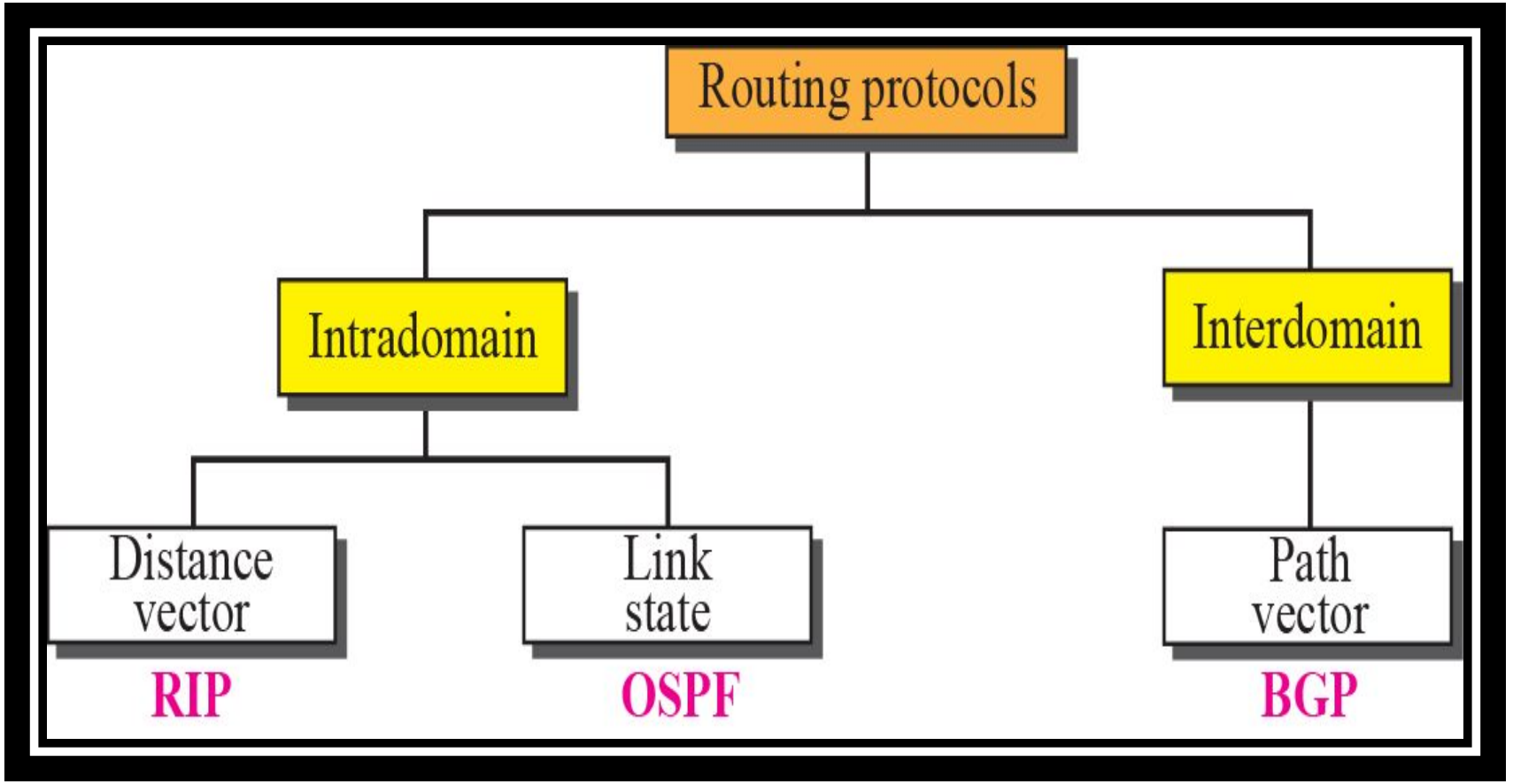


- Collection of networks with same policy
- Single routing protocol
- Usually under single administrative control

Popular routing protocols



Popular routing protocols



DISTANCE VECTOR ROUTING

Today, an internet can be so large that one routing protocol cannot handle the task of updating the routing tables of all routers. For this reason, an internet is divided into autonomous systems.

An autonomous system (AS) is a group of networks and routers under the authority of a single administration.

Routing inside an autonomous system is called intra-domain routing.

Distance Vector Routing Working

- No node has complete information about the costs of all network links
- Gradual calculation of path by exchanging information with neighbors
- Each node constructs a one-dimensional array containing the “distances” or “costs” to all other nodes (as it relates to its knowledge) and distributes it to its immediate neighbors.
- Key thing -- each node knows the cost of links to its neighbors.
- If no link exists between two nodes, the cost of a direct link between the nodes is “infinity”.

Distance Vector Routing

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

Example:

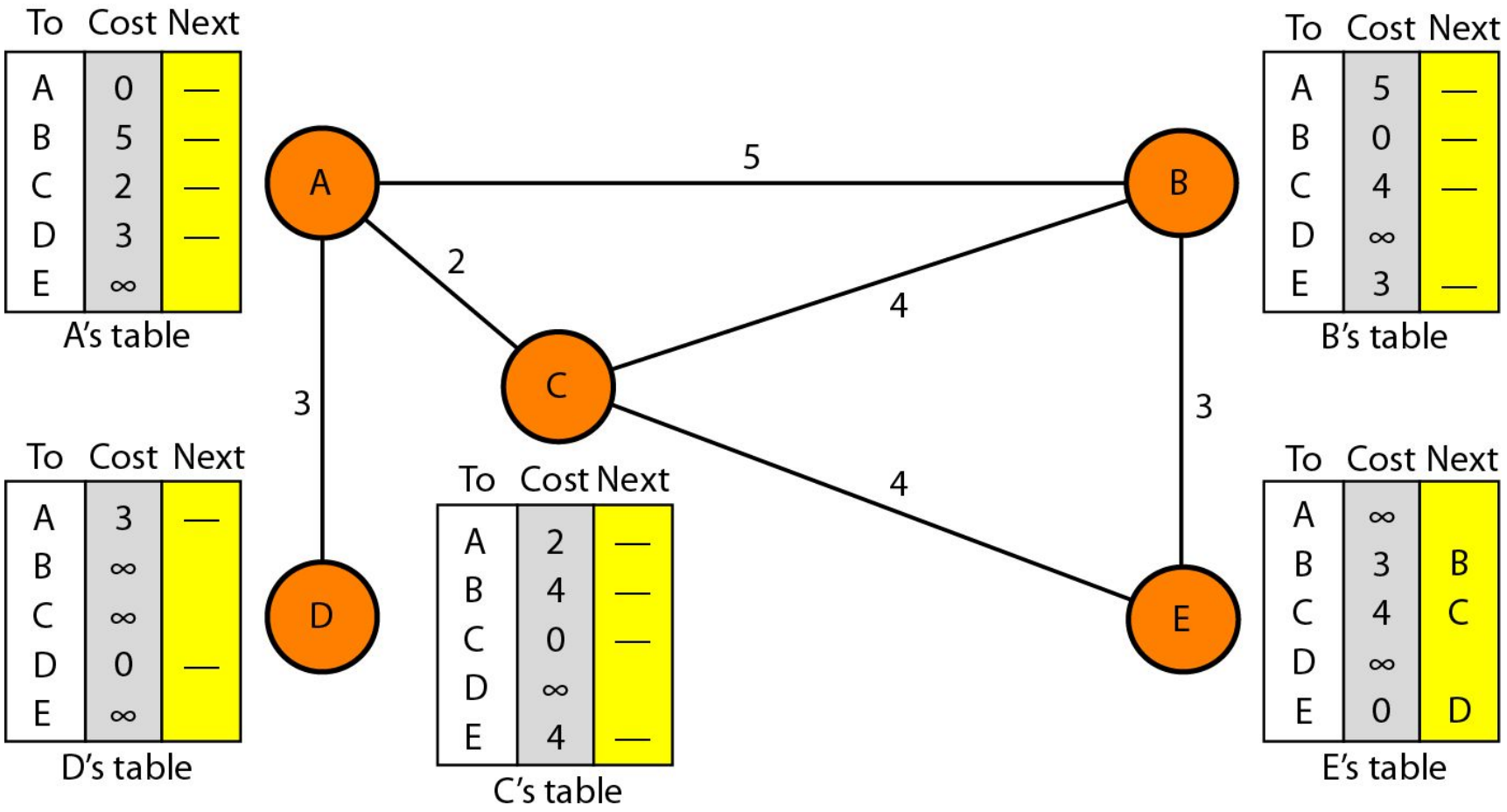
Assume each node as the cities.

Lines as the roads connecting them.

Distance Vector Routing-Initialization

1
2
1

Distance Vector Routing-Initialization



1
2
2

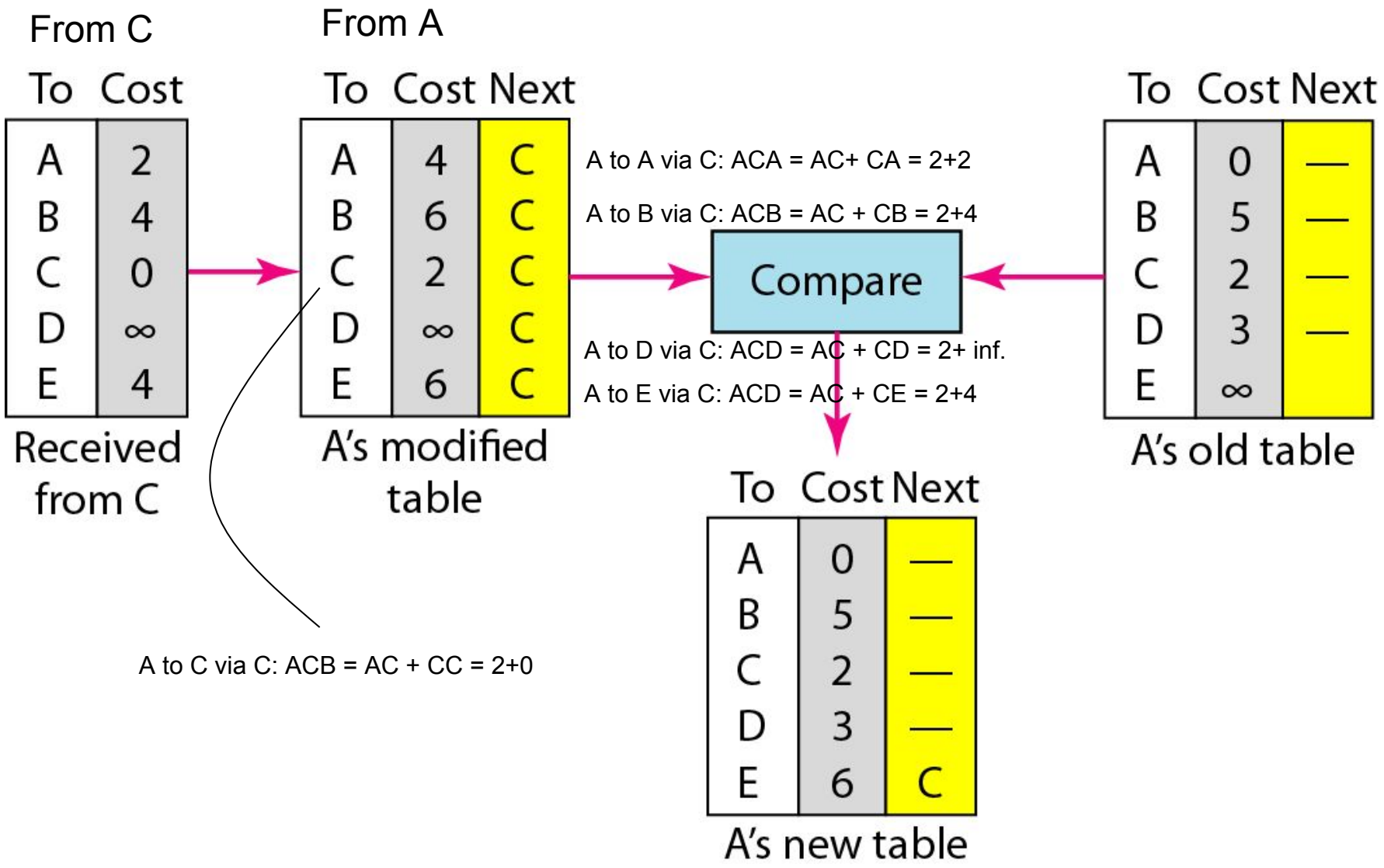
Distance Vector Routing-Sharing

- Idea is to share the information between neighbors.
- The node A does not know the distance about E, but node C does.
- If node C share it routing table with A, node A can also know how to reach node E.
- On the other hand, node C does not know how to reach node D, but node A does.
- If node A share its routing table with C, then node C can also know how to reach node D.
- Node A and C are immediate neighbors, can improve their routing tables if they help each other.

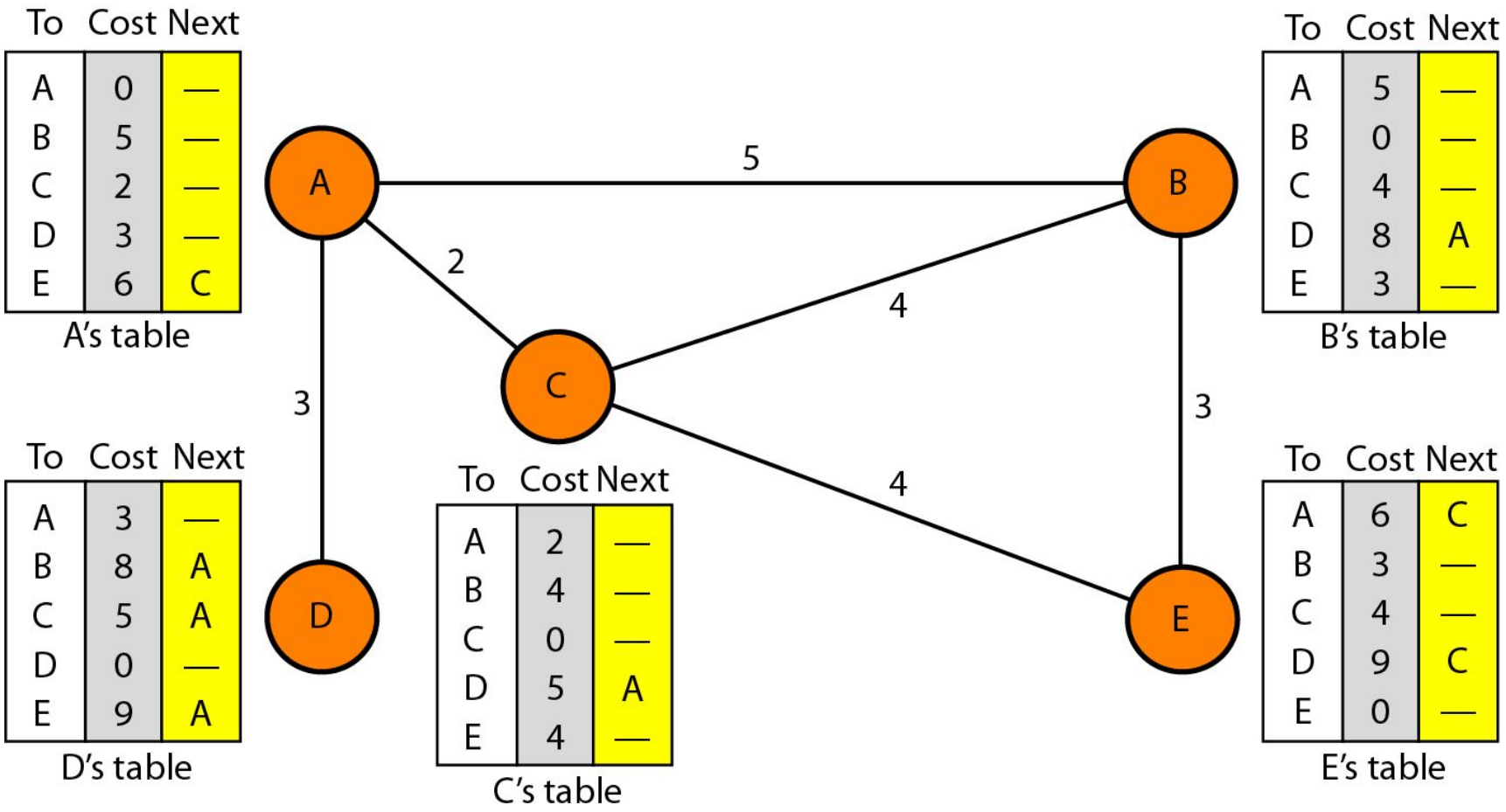
Distance Vector Routing-Sharing

- How much of the table must be shared with each neighbor?
- The third column of the table(next hop) is not useful for the neighbor.
- When the neighbor receives a table, this column needs to be replaced with the sender's name.
- If any of the rows can be used, the next node column filled with sender of the table.
- Therefore, a node can send only the first two column of its table to any neighbor.

Updating in distance vector routing example: C to A



Final Distance vector routing tables



When to Share Routing table with neighbors

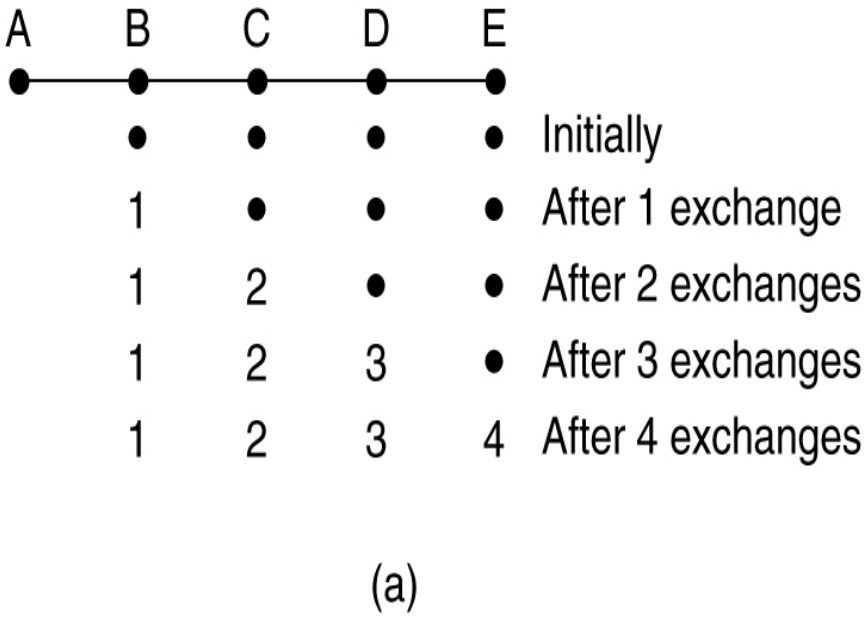
Periodic Update

- A node sends its routing table, normally 30 seconds, in a periodic update

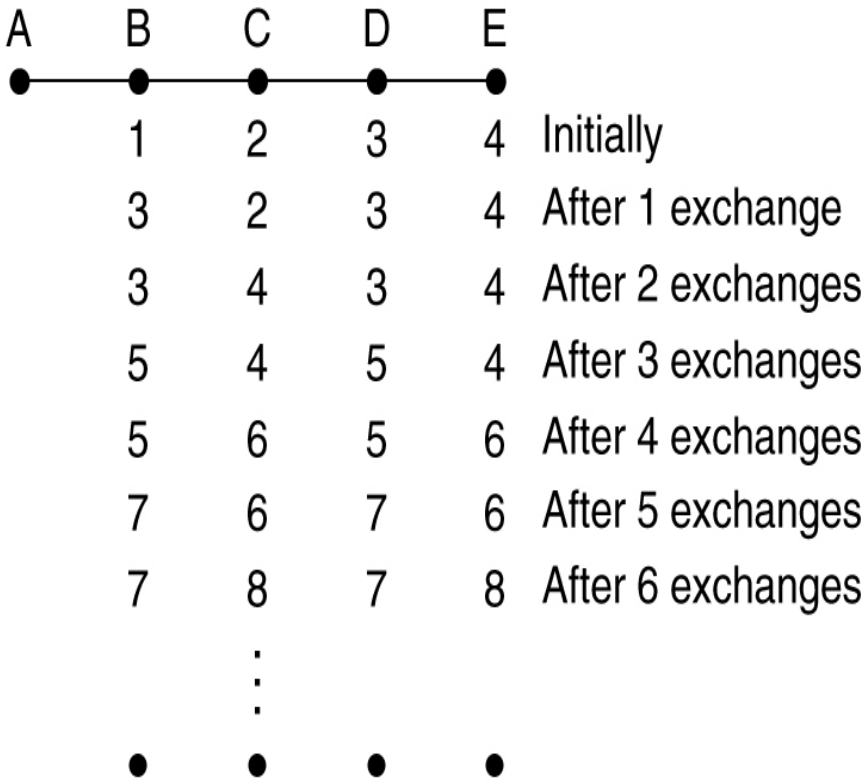
Triggered Update

- A node sends its routing table to its neighbors any time when there is a change in its routing table
  - 1. After updating its routing table, or
  - 2. Detects some failure in the neighboring links

Distance Vector Routing –
The count-to-infinity problem.



(a)



(b)

Distance Vector Routing –
The count-to-infinity problem.

To see the problem clearly, imagine a subnet connected like A–B–C–D–E, and let the metric between the routers be "number of jumps(Hops)".

Now suppose that A is taken offline.

In the vector-update-process B notices that the route to A, which was distance 1, is down – B does not receive the vector update from A.

Distance Vector Routing –
The count-to-infinity problem cont….

The problem is, B also gets an update from C, and C is still not aware of the fact that A is down – so it tells B that A is only two jumps from C (C to B to A), which is false.

Since B doesn't know that the path from C to A is through itself (B), it updates its table with the new value "B to A = 2 + 1".

Later on, B forwards the update to C and due to the fact that A is reachable through B (From C point of view), C decides to update its table to "C to A = 3 + 1".

This slowly propagates through the network until it reaches to infinity (hop 16)

RIP- Routing Information Protocol

The Routing Information Protocol (RIP) is an **intra-domain** (interior) routing protocol used inside an autonomous system.

It is a very simple protocol **based on distance vector routing.**

In the Internet, goal of routers is to learn how to forward packets to various networks.

Routing Information Protocol (RIP)

RIP treats all network equals; the cost of passing thru a network is the same: one hop count per network.

Each router/node maintains a vector (table) of minimum distances to every node.

The hop-count is the number of networks that a packet encounters to reach its destination. Path costs are based on number of hops.

In distance vector routing, each **router periodically shares its table** with its neighbour.

Each router keeps a routing table that has one entry for each destination network . The entry consists of **Destination Network Address, Hop-Count and Next-Router.**

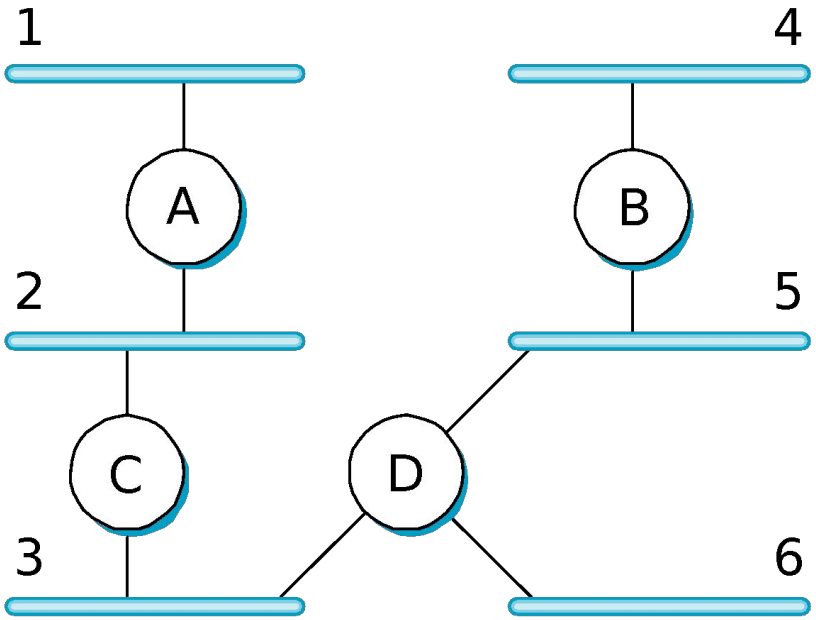
An Example of RIP

Routers advertise the cost of reaching networks.

In this example, C's update to A would indicate that C can reach Networks 2 and 3 with cost 0,

Networks 5 and 6 with cost 1

and Network 4 with cost 2.



https://www.youtube.com/watch?v=ud7qWRBirsk

RIP messages

Request

- A request message is sent by a router that has just come up

Response

- A response can be within 30s or when there is a change in the routing table

RIP Timers

Periodic timer

- Routing tables are exchanged every 30 seconds using the RIP

Expiration timer

- If a router does not hear from its neighbor once every 180 seconds, the neighbor is deemed unreachable.

LINK STATE ROUTING

Link-state routers exchange messages to allow each router to learn the entire network topology.

Based on this learned topology, each router is then able to compute its routing table by using a shortest path computation [Dijkstra1959].

Link State Routing Algorithm Steps

Discover its **neighbors**, learn their network address.

Measure the **delay or cost** to each of its neighbors.

Construct a packet telling all it has just learned.

Send this packet to all other routers.

Compute the **shortest path** to every other router.

Measure the delay or cost

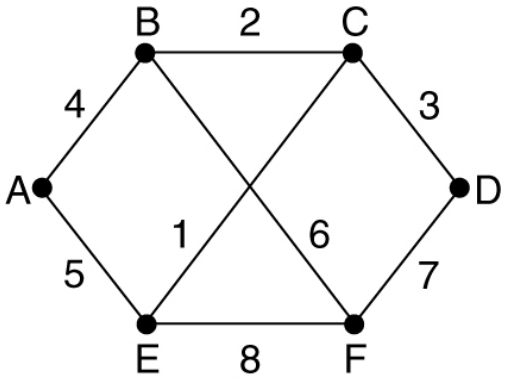
Echo packets are used to measure the line cost

Calculate total time used to echo packet

t = Arrival time – Departure time

Then t/2 gives cost(time) of line

Construct Link State packet



(a)

Link State Packets

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

(b)

(a) A subnet. (b) The link state packets for this subnet.

Send this packet to all other routers and compute the shortest path

Flood the LSP in subnet and then by using Shortest path algorithm (**Dijkstra's algorithm**) compute the shortest path for each router

Distance Vector Routing Vs Link State Routing (DVR Vs LSR)

Distance Vector Routing	Link State Routing
used in small networks	used in larger networks
it has a limited number of hops.	it has unlimited number of hops
high convergence time	convergence time is low.
periodically advertise updates	only new changes in a network.
It has loop problem	No loop problem
Updates are broadcasted	Updates are multicasted
advertises only the directly connected routers and full routing tables,	advertise the updates, and flood the advertisement.
Eg. **RIP ,IGRP , BGP** .	Eg. : **OSPF , IS-IS**

OSPF- Open Shortest Path First

The Open Shortest Path First (OSPF) protocol is an intra-domain routing protocol based on link state routing.

Its domain is also an autonomous system.

Support variety of distance metrics

Dynamic algorithm that adapted to changes in the topology automatically and quickly

https://www.youtube.com/watch?v=UYB_qia_73g

OSPF- Open Shortest Path First

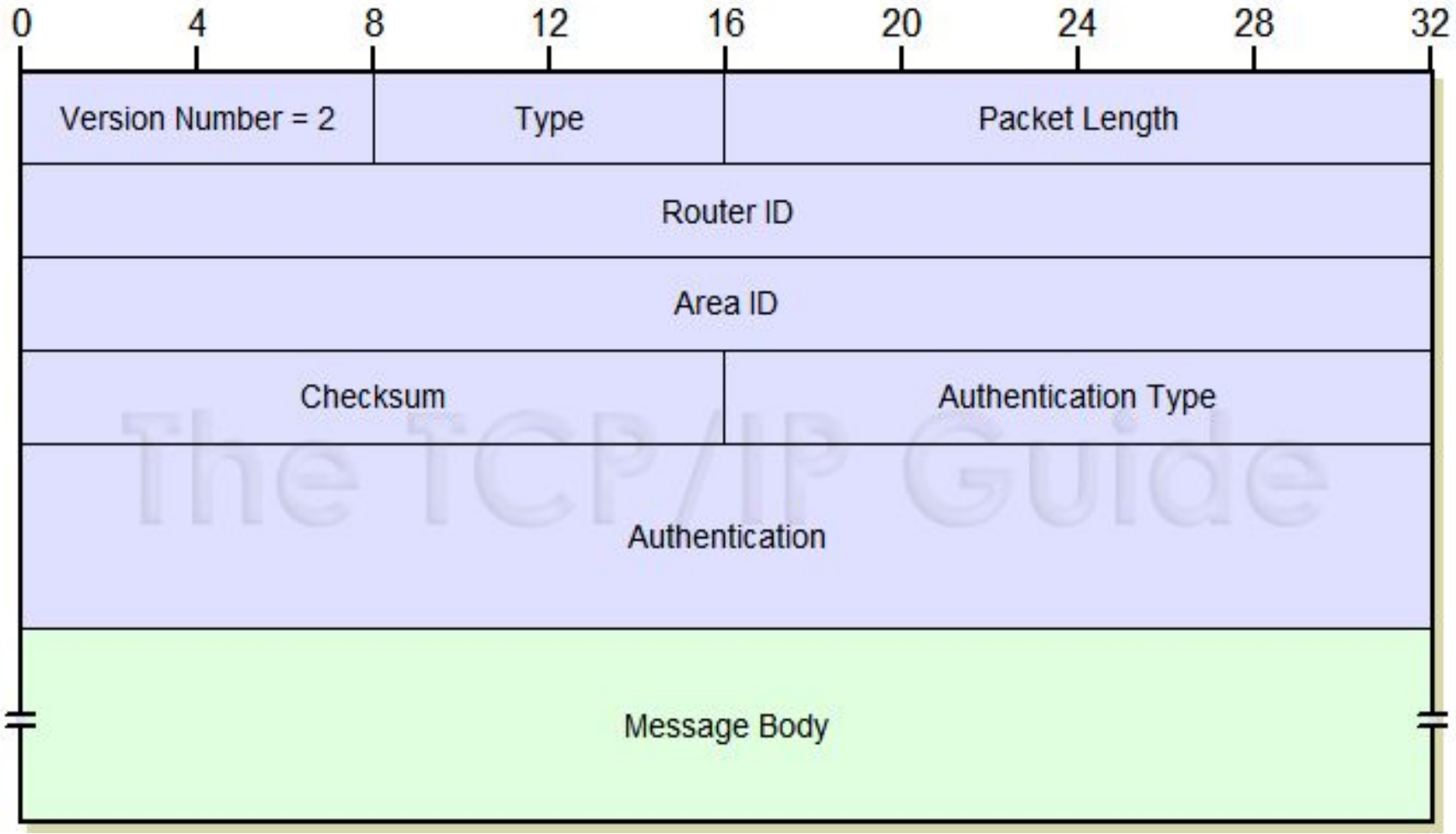
Support routing based on type of service

Do load balancing, splitting the load over multiple lines

Prevent spoofing ie better security provision

Provision for dealing with routers that were connected to the internet via a tunnel

OSPF Header Format



OSPF- Open Shortest Path First

- OSPF divides AS into **areas.**

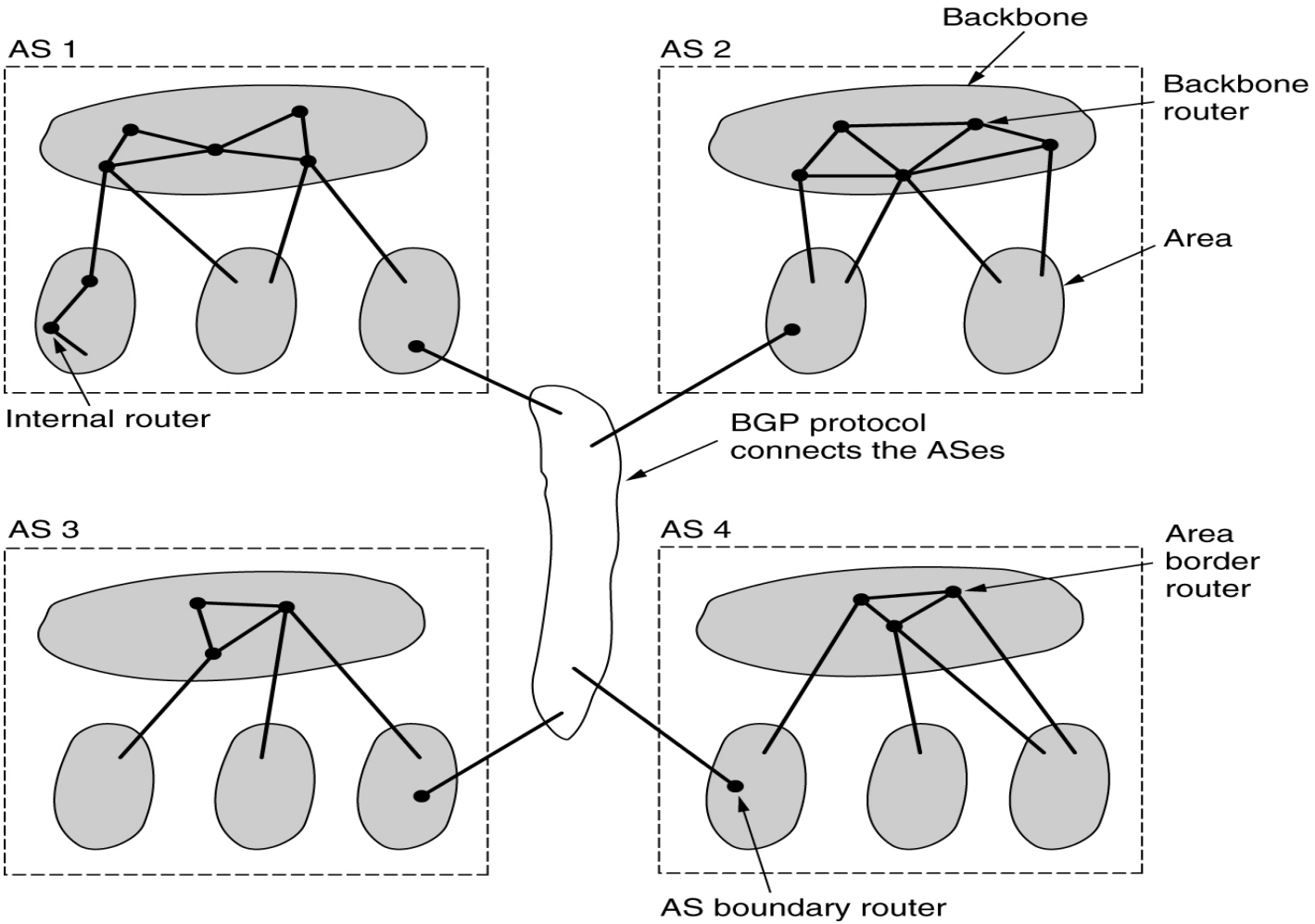
Every AS has a backbone area called **area 0**

All areas are connected to backbone areas

- **OSPF has four classes of router**

1. Internal routers -wholly within on area
2. Area border routers -connect two or more areas
3. Backbone routers -On the backbone area
4. AS boundary routers -Talk to other routers in other AS

OSPF- Open Shortest Path First



The relation between ASes, backbones, and areas in OSPF.

OSPF - WORKING

When a router starts,it first initializes the routing protocol

It then uses the OSPF's handshaking **Hello Protocol** to learn about each other.

The routers exchange information describing their knowledge of the routing domain.This information is called database description and is placed in **LSA** messages.

OSPF - WORKING

Using the above LSA messages the receiving router knows if its LSD is consistent with its peer's databases.If all is consistent the neighbor is now defined as fully adjacent.

A router periodically advertises its state (link state) to detect dead routers in a timely fashion.

From this database each router calculates a shortest path tree with itself the root.

This shortest path tree in turn yields a routing table for the protocol.

OSPF- Routing protocol packets

Hello packet: It is used to discover and maintain neighbor relationships.

Data Description packet and Link State Request packets: They are used in forming adjacencies.

Link State Update and Link State Acknowledgment packets:Used for reliable update mechanisms.

OSPF- Databases

Neighbor Database: Initial table displaying neighbors learned through Hello packets.

Link State Database: Similar in all routers. Formed after each router floods its neighbor database.

Routing table: Each router builds this table by using SPF technology. It gives the shortest path to all the routers in the AS.

BGP- Border Gateway Protocol

Border Gateway Protocol (BGP) is an interdomain routing protocol using path vector routing.

The Border Gateway Protocol makes routing decisions based on paths, network policies or rule-sets configured by a network administrator,

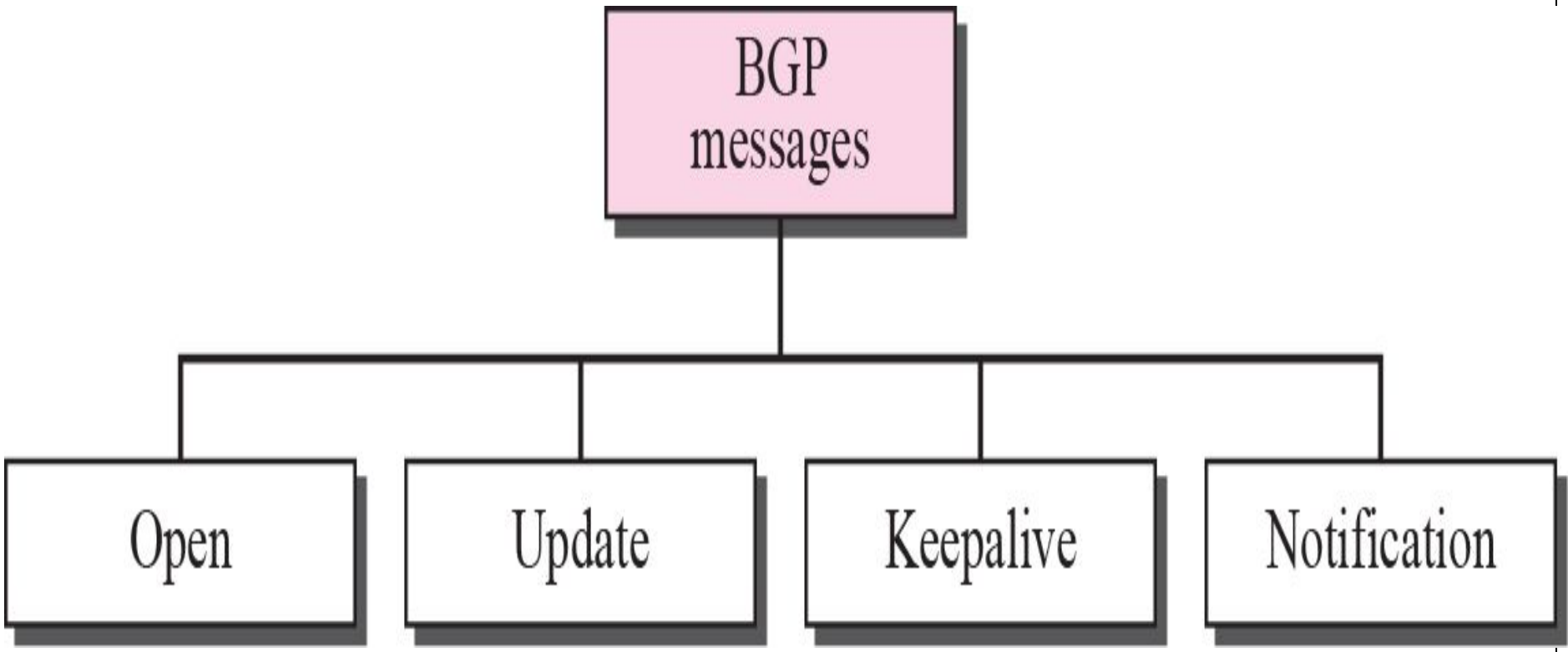
BGP- Border Gateway Protocol

When BGP runs between two peers in the same autonomous system (AS), it is referred to as ***Internal BGP* (*iBGP* or *Interior Border Gateway Protocol*).**

When it runs between different autonomous systems, it is called ***External BGP* (*EBGP* or *Exterior Border Gateway Protocol*).**

Routers on the boundary of one AS exchanging information with another AS are called *border* or *edge routers. BGP uses the services of TCP on port 179.*

Types of BGP messages



BGP Messages

Open

- Announces AS ID
- Determines hold timer – interval between keep_alive or update messages, zero interval implies no keep_alive

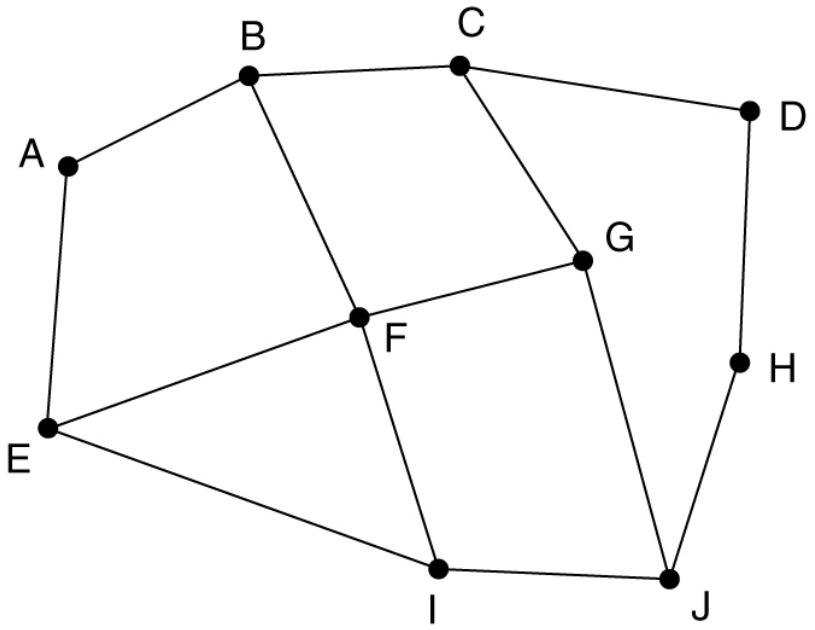
Keep_alive

- Sent periodically to peers to ensure connectivity.

Notification

- Used for error notification
- TCP connection is closed *immediately* after notification

BGP – Example



(a)

Information F receives from its neighbors about D

From B: "I use BCD"
From G: "I use GCD"
From I: "I use IFGCD"
From E: "I use EFGCD"

(b)

(a) A set of BGP routers. (b) Information sent to F.

Path Attributes

ORIGIN

- The source of the routing information (RIP, OSPF, etc)

AS_PATH

- The list of ASs through which the destination can be reached

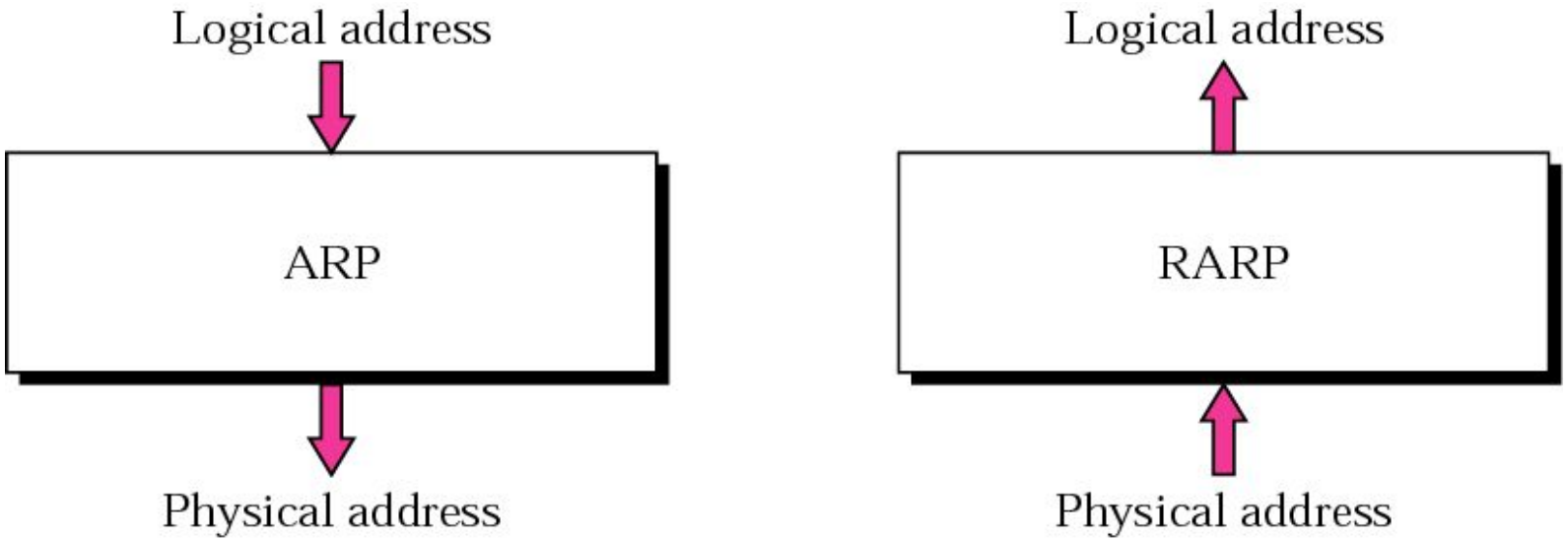
NEXT-HOP

- The next router to which the data packet should be sent

Outline

- Switching techniques,
- IP Protocol,
- IPv4 and IPv6 addressing schemes,
- Subnetting,
- NAT, CIDR,
- ICMP,
- Routing Protocols: Distance Vector, Link State, Path Vector.
- Routing in Internet: RIP ,OSPF, BGP,
- **ARP and RARP**

ARP and RARP


Logical address
ARP
Physical address
Logical address
RARP
Physical address

ARP (Address Resolution Protocol)

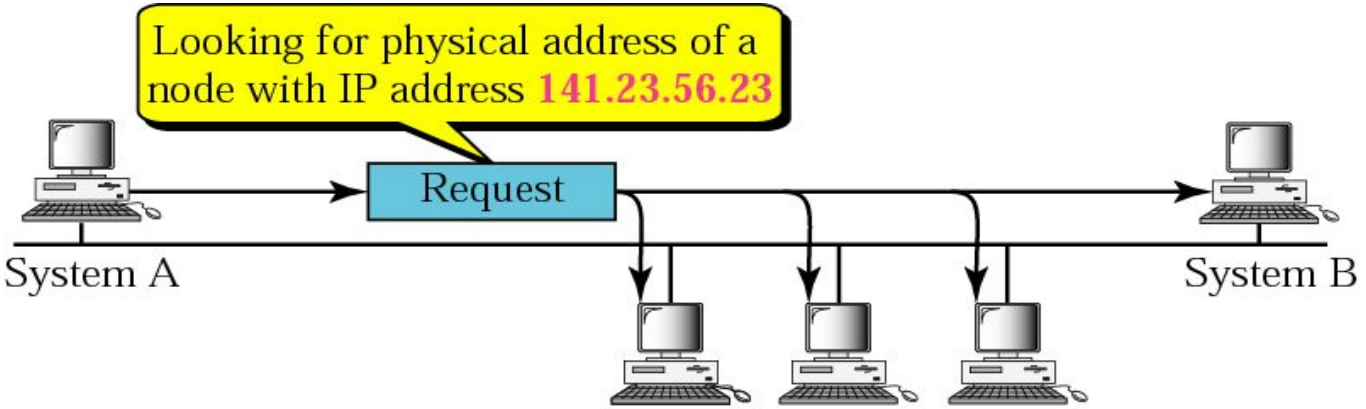
ARP associates an IP address with its physical address. On a typical physical network, such as a LAN, each device on a link is identified by a physical or station address that is usually imprinted on the NIC.

The delivery of a packet to a host or a router requires two levels of addressing: logical and physical.

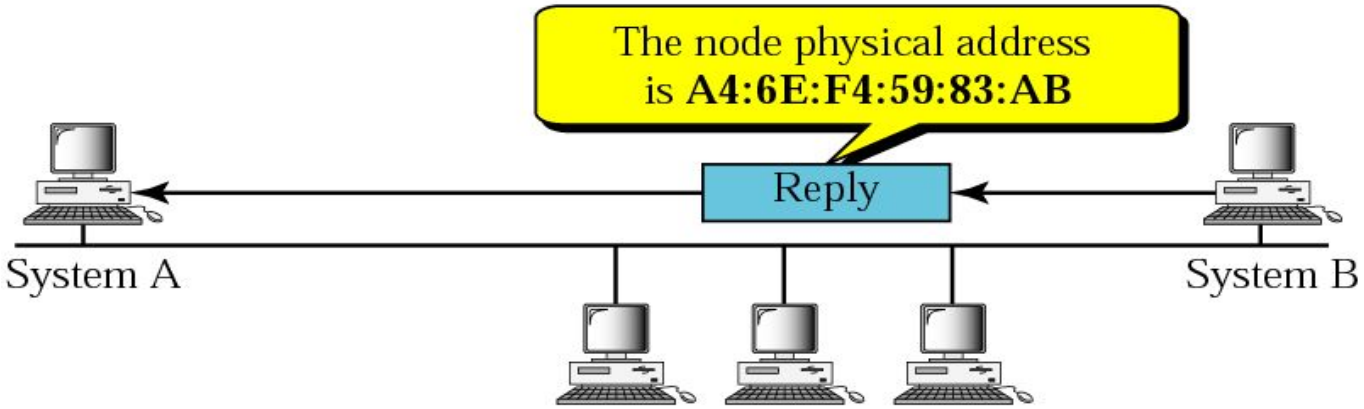
We need to be able to map a logical address to its corresponding physical address and vice versa. These can be done using either static or dynamic mapping.

Logical address to physical address translation can be done statically (not practical) or dynamically (with ARP).

ARP operation

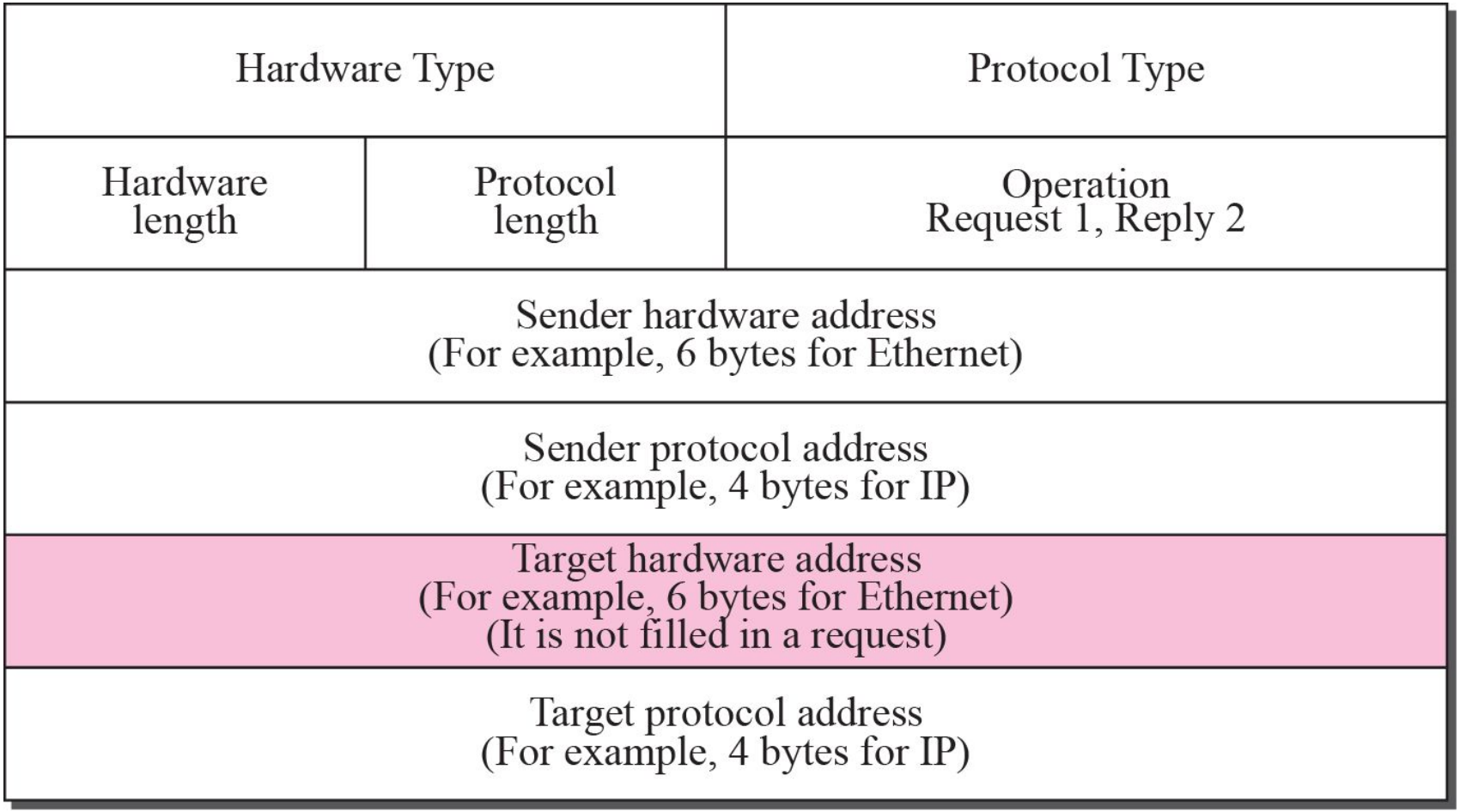


a. ARP request is broadcast



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)

Note

An ARP request is broadcast; an ARP reply is unicast.

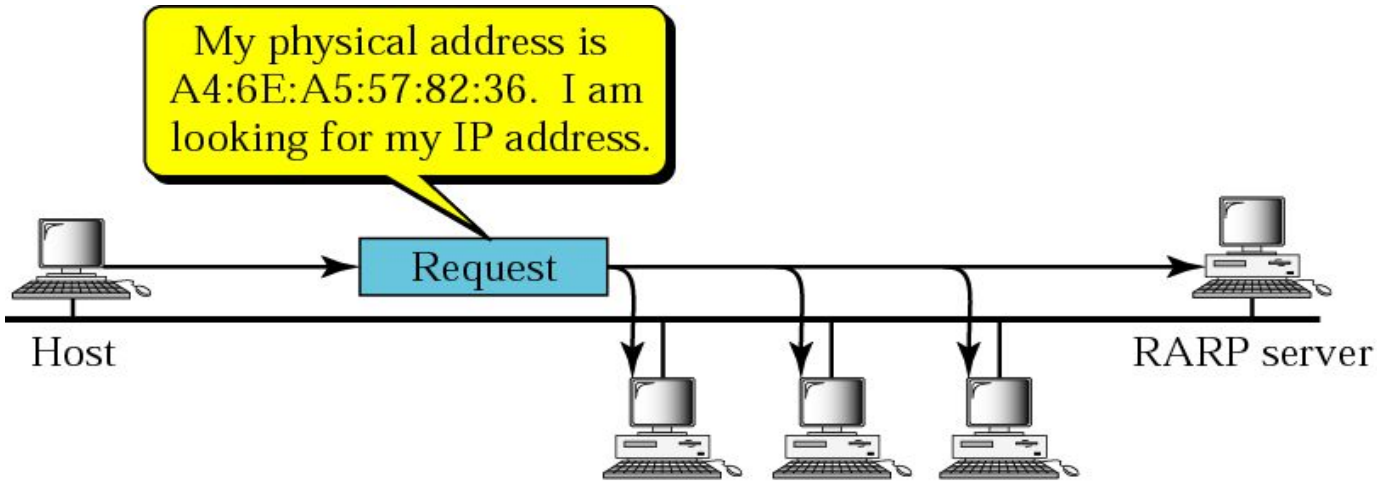
RARP (Reverse Address resolution Protoco

RARP finds the logical address for a machine that only knows its physical address. RARP requests are broadcast, RARP replies are unicast.

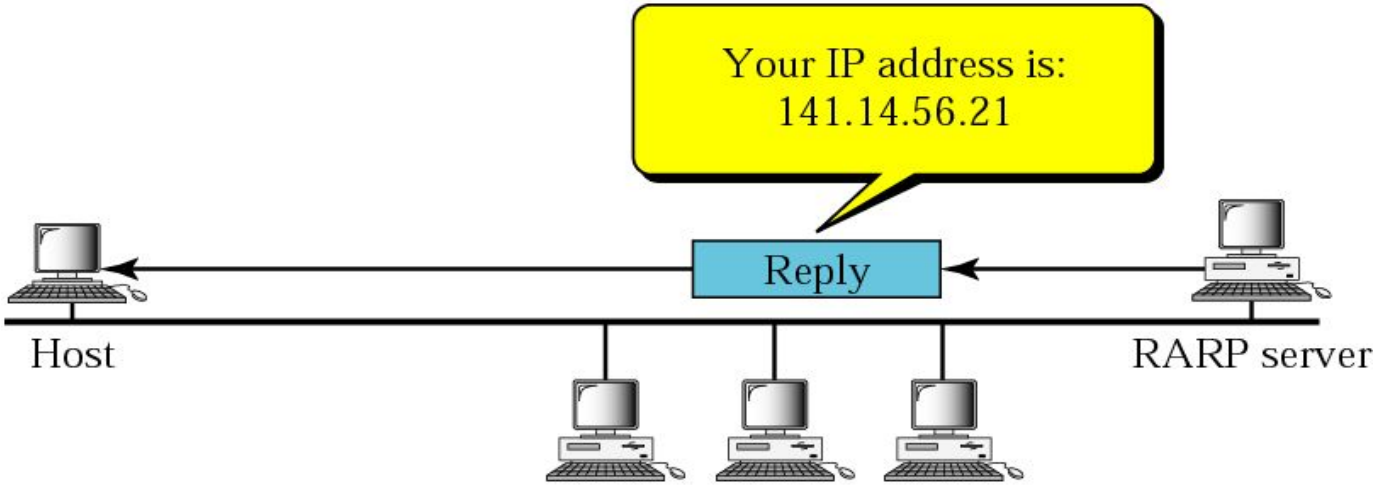
This if often encountered on thin-client workstations. No disk, so when machine is booted, it needs to know its IP address (don't want to burn the IP address into the ROM).

If a thin-client workstation needs to know its IP address, it probably also needs to know its subnet mask, router address, DNS address, etc.So we need something more than RARP. BOOTP, and now DHCP have replaced RARP.

RARP operation

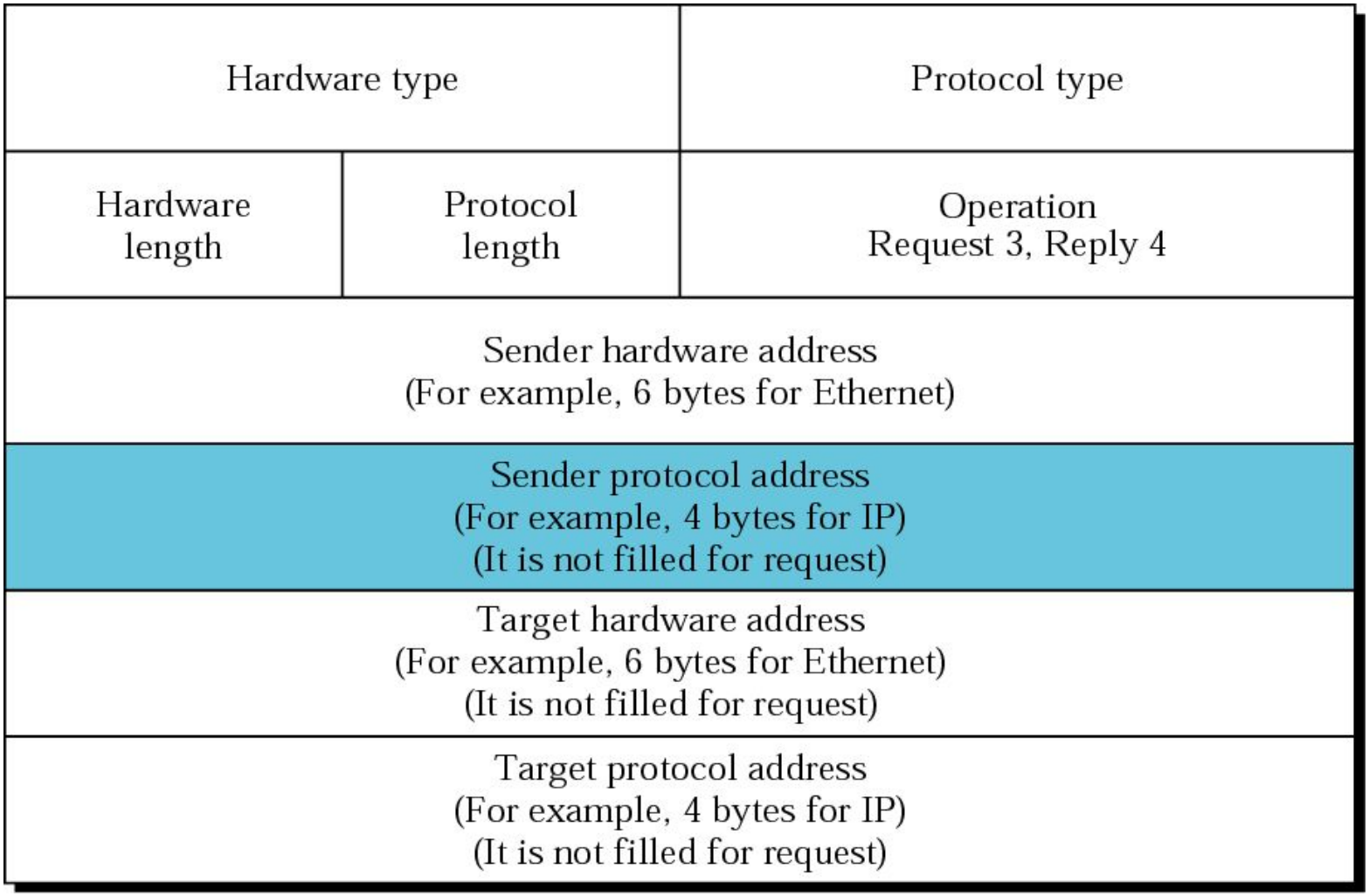


a. RARP request is broadcast



b. RARP reply is unicast

RARP packet

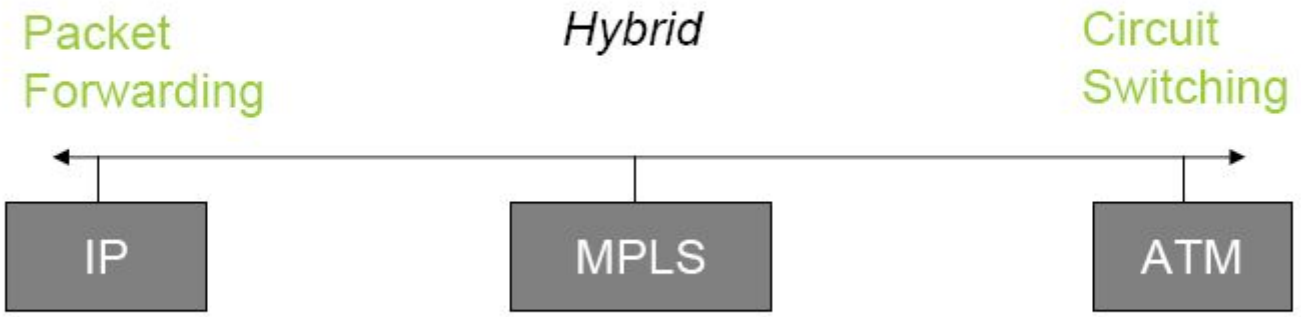

Hardware type
Protocol type
Hardware length
Protocol length
Operation
Request 3, Reply 4
Sender hardware address
(For example, 6 bytes for Ethernet)
Sender protocol address
(For example, 4 bytes for IP)
(It is not filled for request)
Target hardware address
(For example, 6 bytes for Ethernet)
(It is not filled for request)
Target protocol address
(For example, 4 bytes for IP)
(It is not filled for request)

Outline

- Switching techniques,
- IP Protocol,
- IPv4 and IPv6 addressing schemes,
- Subnetting,
- NAT, CIDR,
- ICMP,
- Routing Protocols: Distance Vector, Link State, Path Vector,
- Routing in Internet: RIP ,OSPF, BGP,
- ARP and RARP
- MPLS,

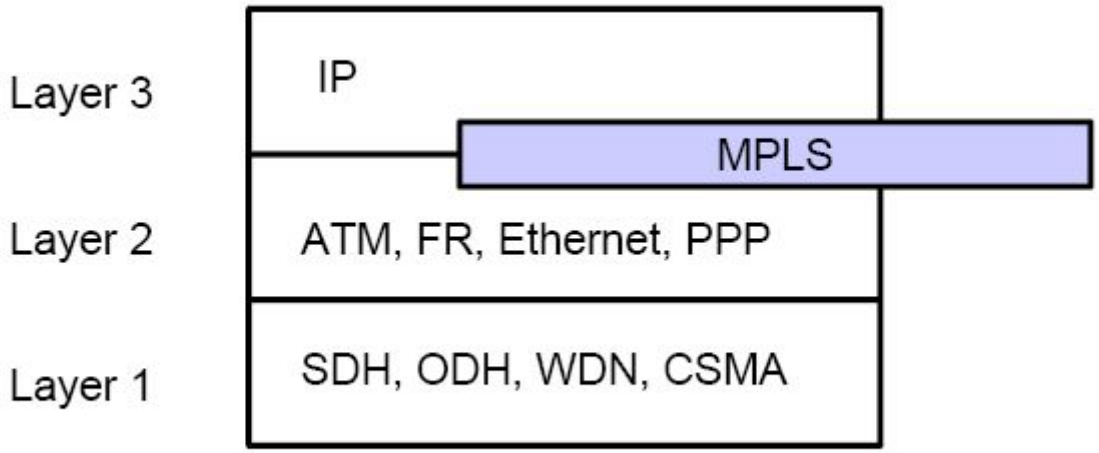
Motivation

- Idea: Combine the forwarding algorithm used in ATM with IP.



MPLS Basics

- Multi Protocol Label Switching is arranged between Layer 2 and Layer 3

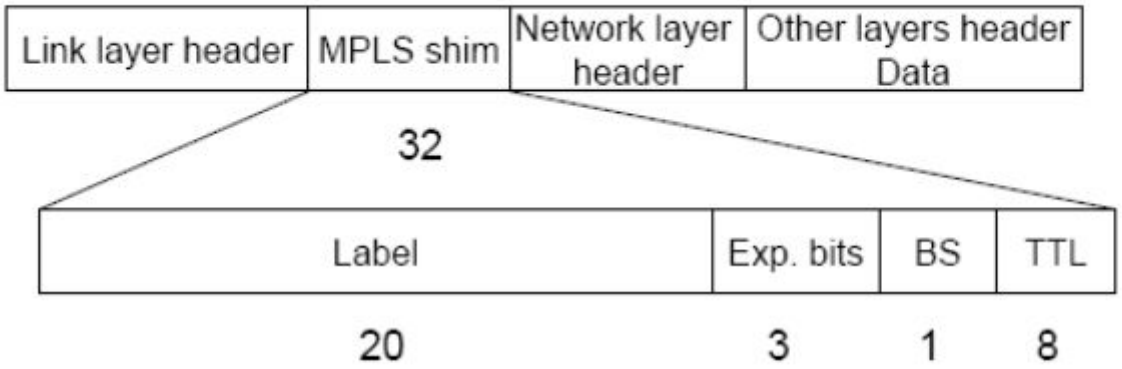


MPLS Basics (cont.)

- MPLS Characteristics
  - Mechanisms to manage traffic flows of various granularities (*Flow Management*)
  - Is independent of Layer-2 and Layer-3 protocols
  - Maps IP-addresses to fixed length labels
  - Supports ATM, Frame-Relay and Ethernet

Label

- Generic label format



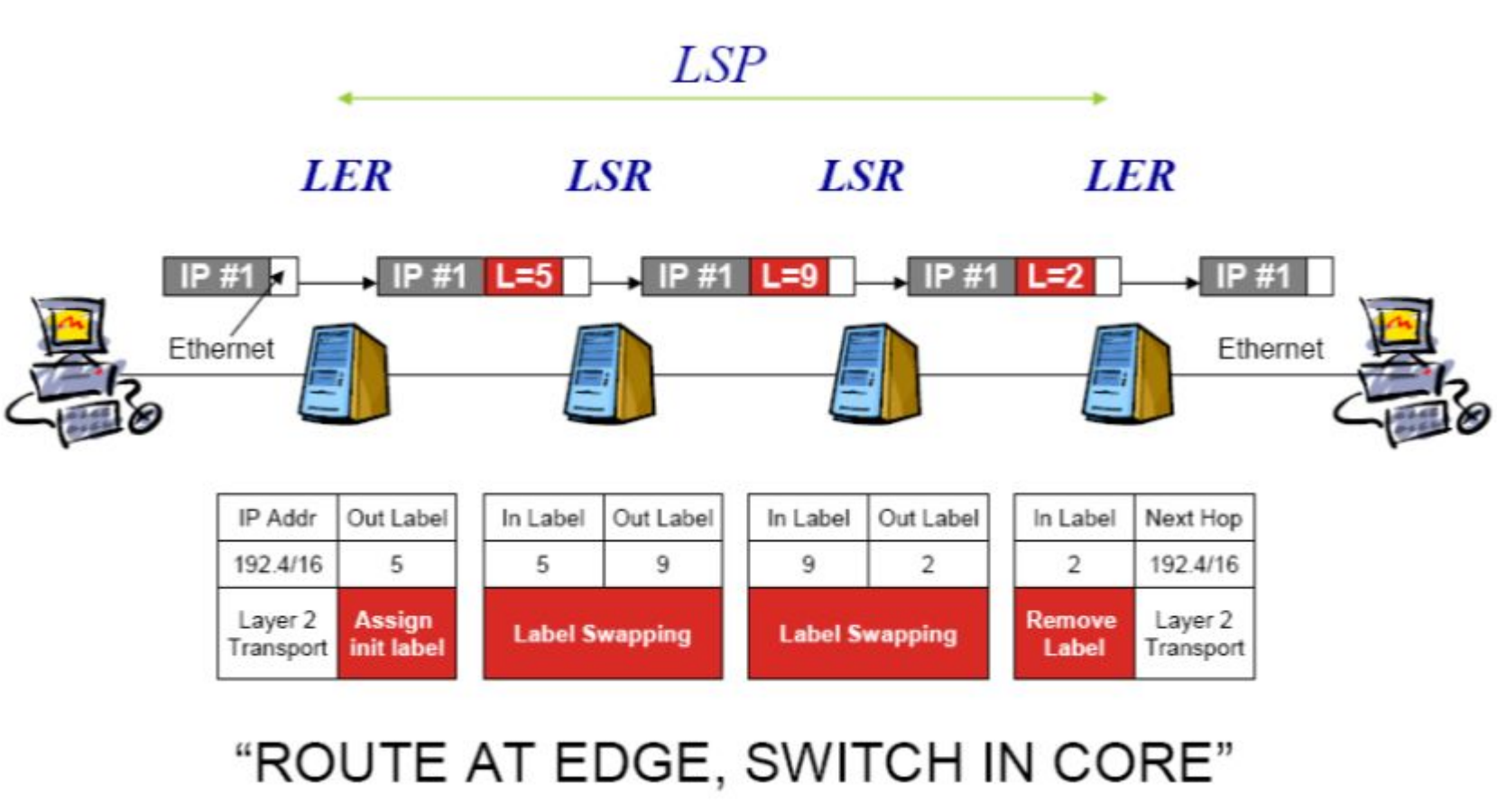
Exp.bits: Experimental Bits, often used for Class of Service
BS: Bottom of Stack bit, is set if no label follows
TTL: Time To Leave, used in the same way like in IP

MPLS ROuters

- **Label Edge Router - LER**
  - Resides at the edge of an MPLS network and assigns and removes the labels from the packets.
  - Support multiple ports connected to dissimilar networks (such as frame relay, ATM, and Ethernet).

- **Label Switching Router – LSR**
  - Is a high speed router in the core on an MPLS network.
  - ATM switches can be used as LSRs without changing their hardware. Label switching is equivalent to VP/VC switching.

Positions of LERs & LSRs



"ROUTE AT EDGE, SWITCH IN CORE"

MPLS Advantages & Disadvantages

- Advantages
  - Improves packet-forwarding performance in the network
  - Supports QoS and CoS for service differentiation
  - Supports network scalability
  - Integrates IP and ATM in the network
  - Builds interoperable networks
- Disad.
  - An additional layer is added
  - The router has to understand MPLS

Outline

- Switching techniques,
- IP Protocol,
- IPv4 and IPv6 addressing schemes,
- Subnetting,
- NAT, CIDR,
- ICMP,
- Routing Protocols: Distance Vector, Link State, Path Vector,
- Routing in Internet: RIP ,OSPF, BGP,
- Congestion control and QoS,
- MPLS,
- **Mobile IP,**

Mobile IP

- **Developed as a means for transparently dealing with problems of mobile users**
- Enables hosts to stay connected to the Internet regardless of their location and without changing IP addresses
- Requires no changes to software of non-mobile hosts/routers
- **Requires addition of some infrastructure**
- Has no geographical limitations
- Requires no modifications to IP addresses
- Supports security
- IETF standardization process is still underway

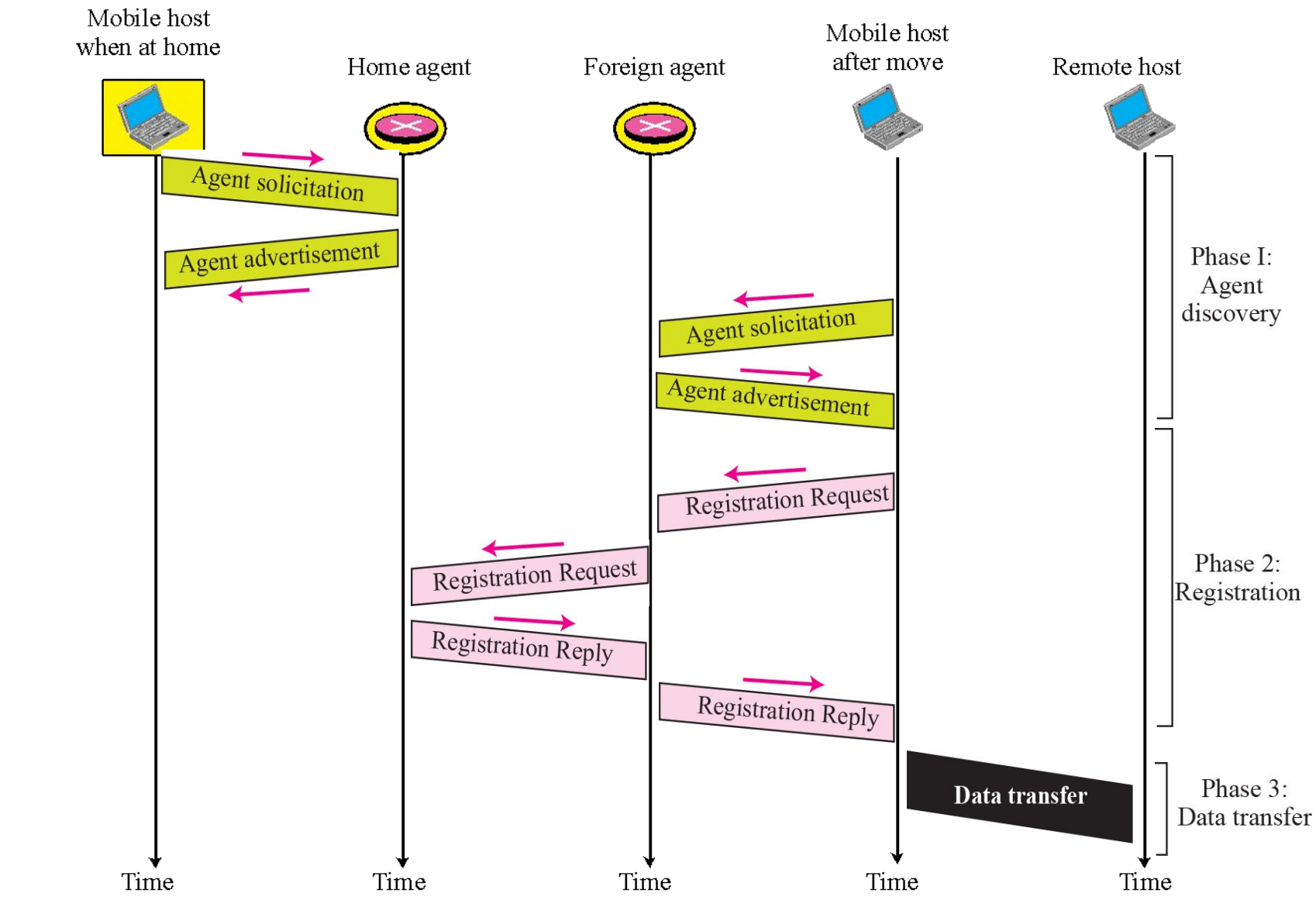
Mobile IP Entities

- Mobile Node (MN)
  - The entity that moves from network to network
  - Assigned a permanent IP called its *home address* to which other hosts send packets regardless of MN's location
- Home Agent (HA)
  - Router with additional functionality
  - Located on home network of MN
  - Mobility binding of MN's IP with its *Care of Address* (COA)
  - Forwards packets to appropriate network when MN is away – uses encapsulation

Mobile IP Entities contd.

- Foreign Agent (FA)
  - Another router with enhanced functionality
  - Used to send/receive data between MN and HA
  - Advertises itself periodically
- Care-of-address (COA)
  - Address which identifies MN's current location
  - Sent by FA to HA when MN attaches
  - Usually the IP address of the FA
- Correspondent Node (CN)
  - End host to which MN is corresponding (eg. a web server)

Remote host and mobile host configuration



Mobile IP Support Services

- Agent Discovery
  - HA's and FA's broadcast their presence on each network to which they are attached
  - MN's listen for advertisement and then initiate registration
- Registration
  - When MN is away, it registers its COA with its HA, via FA
  - Registration control messages sent via UDP to well known port
- Encapsulation/decapsulation – just like standard IP only with COA

Mobile IP Operation

- A MN listens for agent advertisement and then initiates registration
  - If responding agent is the HA, then mobile IP is not necessary
- After receiving the registration request from a MN, the HA acknowledges and registration is complete
  - Registration happens as often as MN changes networks
- HA intercepts all packets destined for MN
  - This is simple unless sending application is on or near the same network as the MN
  - HA masquerades as MN
  - There is a specific lifetime for service before a MN must re-register
  - There is also a de-registration process with HA if an MN returns home

Outline

- Switching techniques,
- IP Protocol,
- IPv4 and IPv6 addressing schemes,
- Subnetting,
- NAT, CIDR,
- ICMP,
- Routing Protocols: Distance Vector, Link State, Path Vector.
- Routing in Internet: RIP ,OSPF, BGP,
- Congestion control and QoS,
- MPLS,
- Mobile IP,
- **Routing in MANET : AODV, DSR**

Unicast Routing Protocols

- Many protocols have been proposed

- Some specifically invented for MANET
- Others adapted from protocols for wired networks

- No single protocol works well in all environments
  - some attempts made to develop adaptive/hybrid protocols

- Standardization efforts in IETF
  - MANET, MobileIP working groups
  - **http://www.ietf.org**

Routing Protocols

Proactive protocol

- Traditional distributed shortest-path protocols
- Maintain routes between every host pair at all times
- Based on periodic updates; High routing overhead
- Example: DSDV (destination sequenced distance vector)

Reactive protocols

- Determine route if and when needed
- Source initiates route discovery
- Example: DSR (dynamic source routing)

Hybrid protocols

- Adaptive; Combination of proactive and reactive
- Example : ZRP (zone routing protocol)

Protocol Trade-offs

Proactive protocols

- Always maintain routes
- Little or no delay for route determination
- Consume bandwidth to keep routes up-to-date
- Maintain routes which may never be used

Reactive protocols

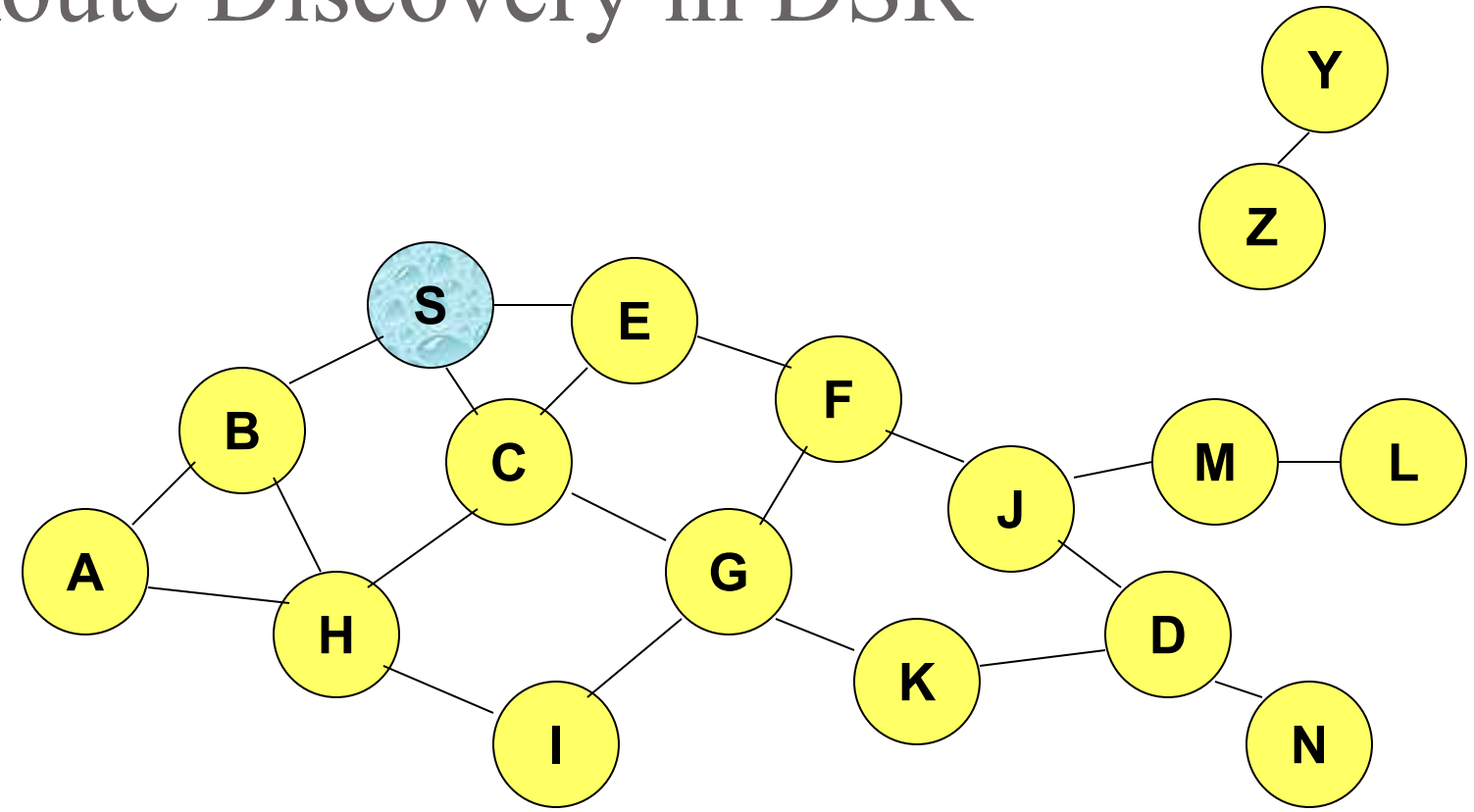
- Lower overhead since routes are determined on demand
- Significant delay in route determination
- Employ flooding (global search)
- Control traffic may be bursty

Which approach achieves a better trade-off depends on the traffic and mobility patterns

Dynamic Source Routing (**DSR**)

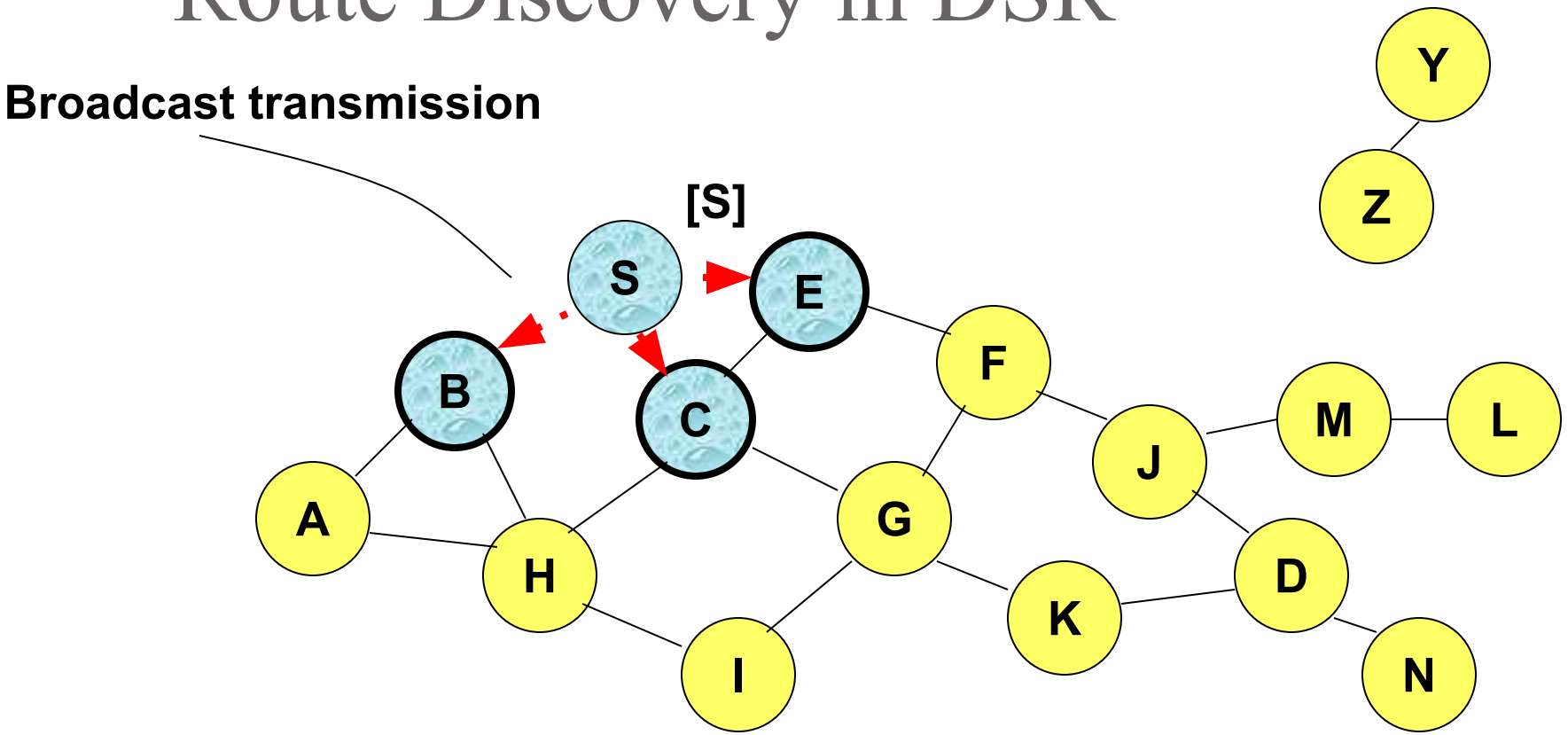
- When node S wants to send a packet to node D, but does not know a route to D, node S initiates a route discovery
- Source node S floods Route Request (RREQ)
- Each node *appends own identifier* when forwarding RREQ

Route Discovery in DSR



Represents a node that has received RREQ for D from S

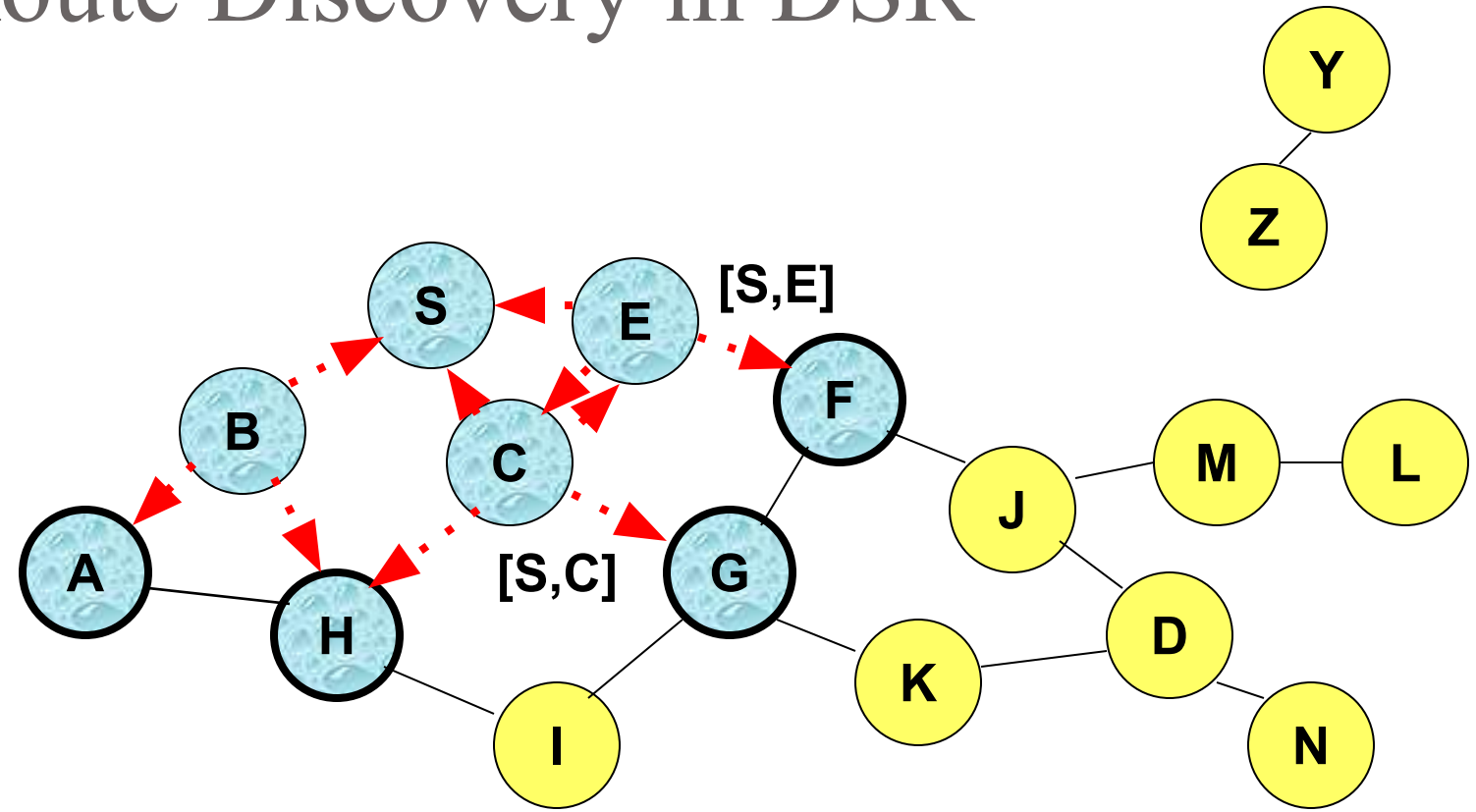
Route Discovery in DSR



Represents transmission of RREQ

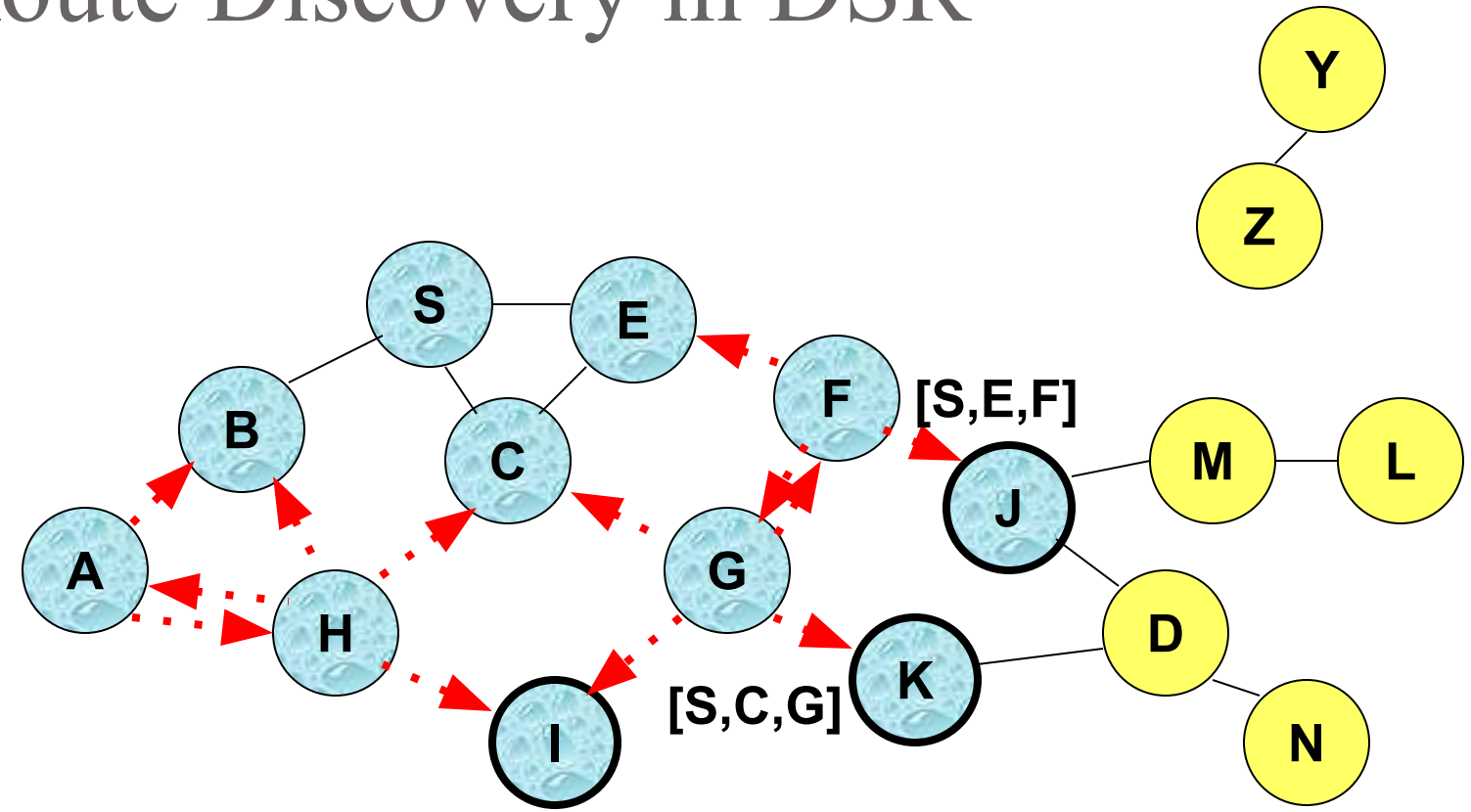
[X,Y] Represents list of identifiers appended to RREQ

Route Discovery in DSR



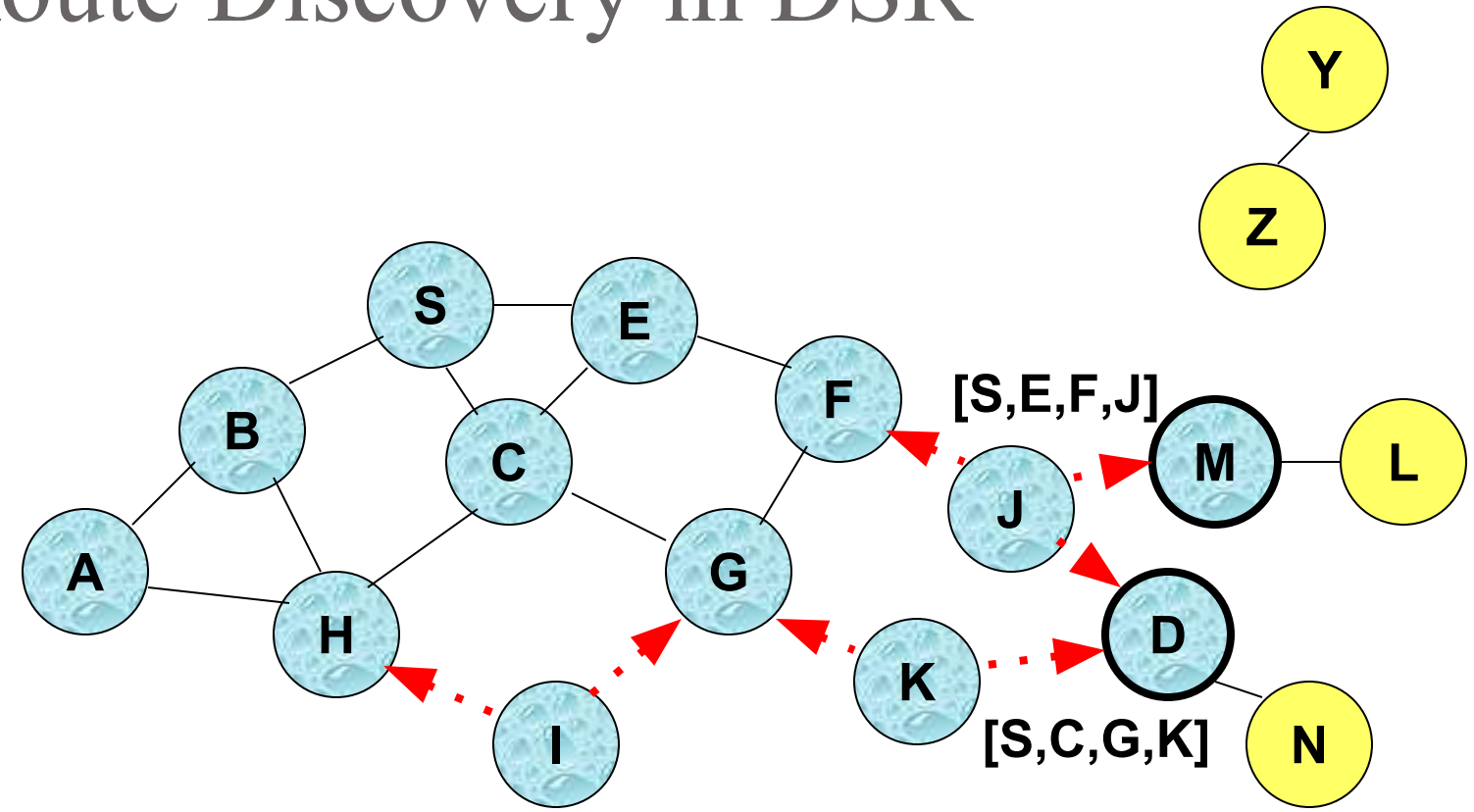
- **Node H receives packet RREQ from two neighbors:**
  potential for collision

Route Discovery in DSR



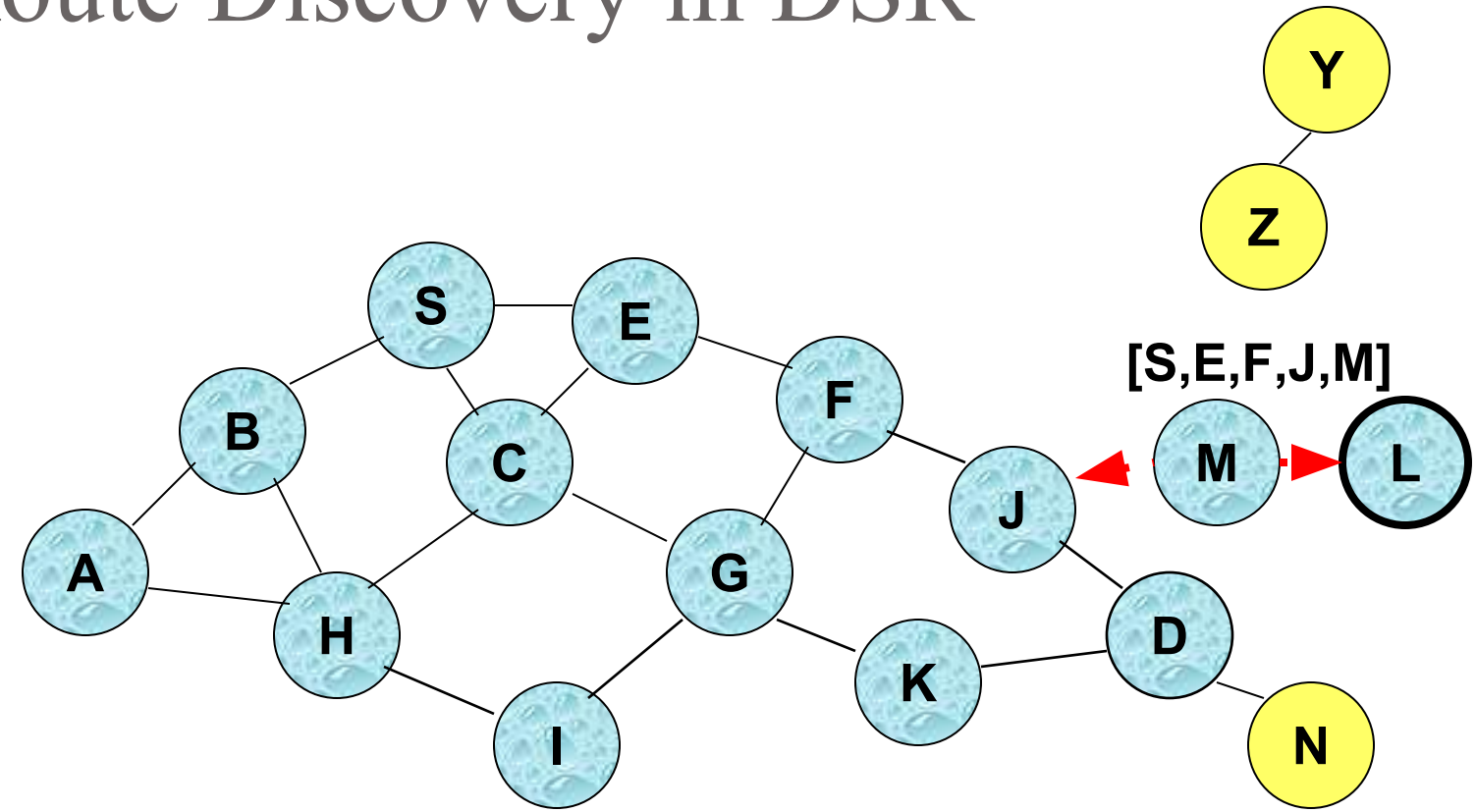
- Node C receives RREQ from G and H, but does not forward it again, because node C has already forwarded RREQ once

Route Discovery in DSR



- Nodes J and K both broadcast RREQ to node D
- Since nodes J and K are hidden from each other, their transmissions may collide

Route Discovery in DSR

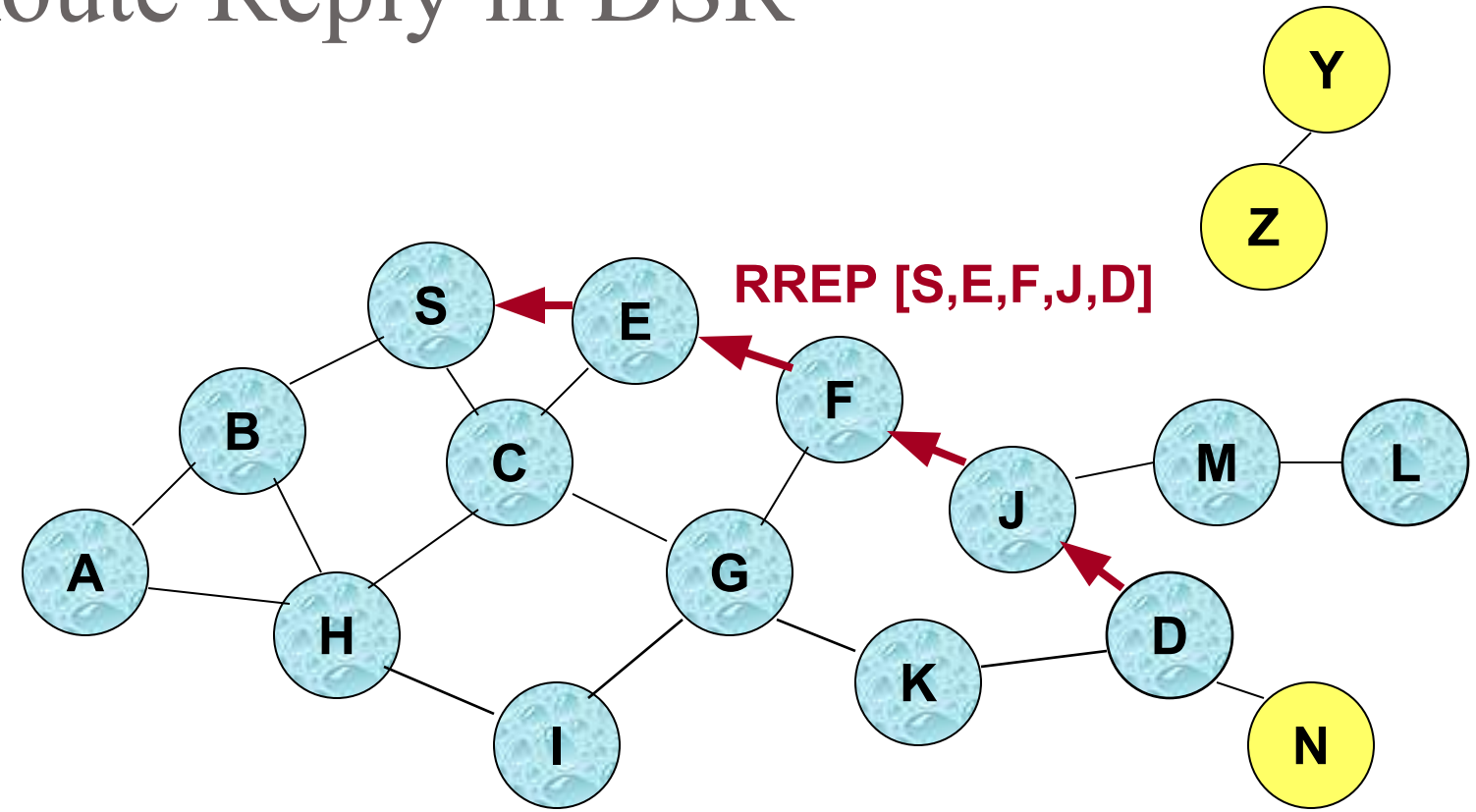


- Node D **does not forward** RREQ, because node D is the **intended target** of the route discovery

Route Discovery in DSR

- Destination D on receiving the first RREQ, sends a Route Reply (RREP)
- RREP is sent on a route obtained by reversing the route appended to received RREQ
- RREP includes the route from S to D on which RREQ was received by node D

Route Reply in DSR

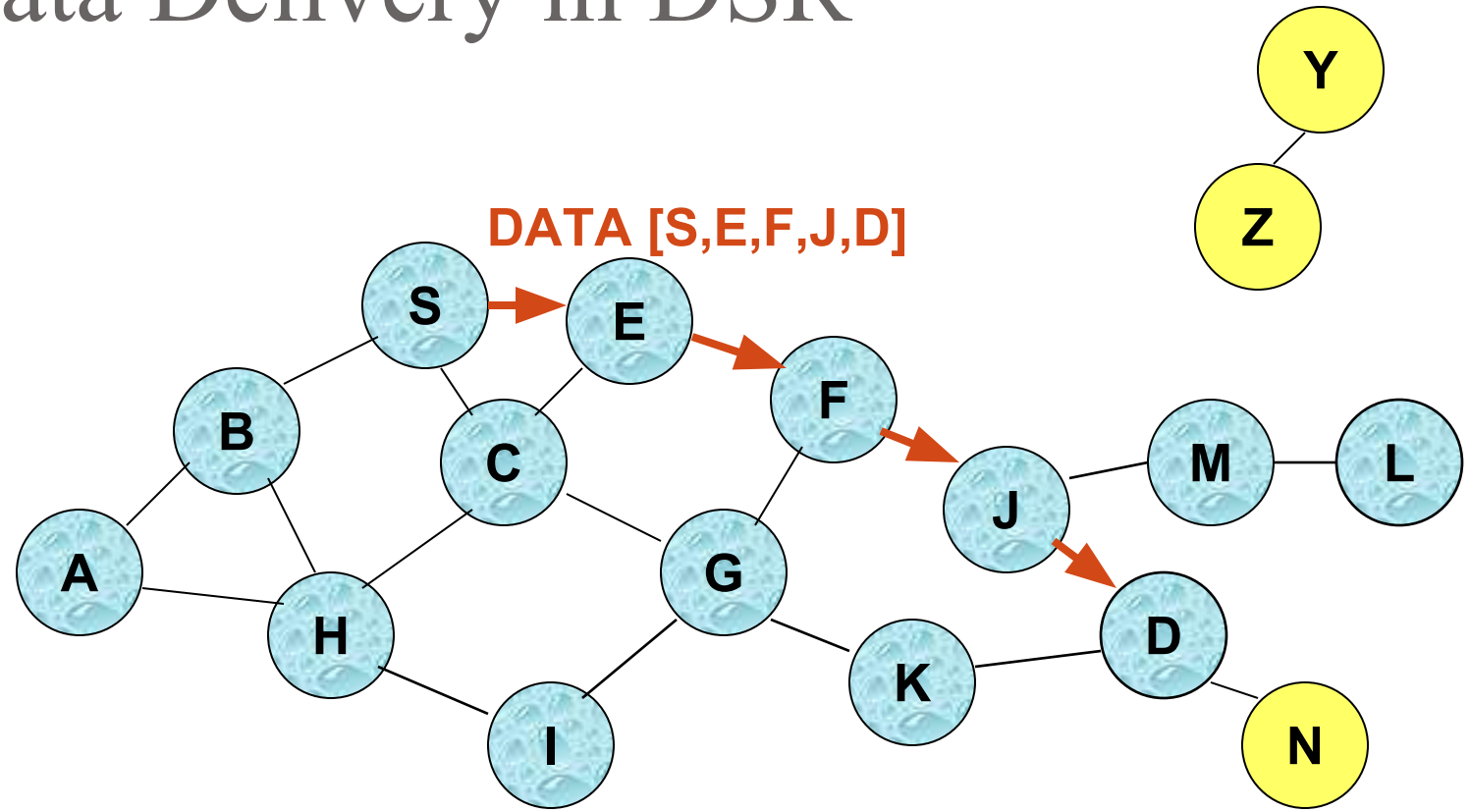


Represents RREP control message

Dynamic Source Routing (DSR)

- Node S on receiving RREP, caches the route included in the RREP
- When node S sends a data packet to D, the entire route is included in the packet header
  - hence the name source routing
- Intermediate nodes use the source route included in a packet to determine to whom a packet should be forwarded

Data Delivery in DSR



Packet header size grows with route length

DSR Optimization: Route Caching

- Each node caches a new route it learns by *any means*
- When node S finds route [S,E,F,J,D] to node D, node S also learns route [S,E,F] to node F
- When node K receives Route Request [S,C,G] destined for node, node K learns route [K,G,C,S] to node S
- When node F forwards Route Reply RREP [S,E,F,J,D], node F learns route [F,J,D] to node D
- When node E forwards Data [S,E,F,J,D] it learns route [E,F,J,D] to node D
- A node may also learn a route when it overhears Data
- Problem: Stale caches may increase overheads

Dynamic Source Routing: Advantages

- Routes maintained only between nodes who need to communicate
  - reduces overhead of route maintenance
- Route caching can further reduce route discovery overhead
- A single route discovery may yield many routes to the destination, due to intermediate nodes replying from local caches

Dynamic Source Routing: Disadvantages

- Packet header size grows with route length due to source routing
- Flood of route requests may potentially reach all nodes in the network
- Potential collisions between route requests propagated by neighboring nodes
  - insertion of random delays before forwarding RREQ
- Increased contention if too many route replies come back due to nodes replying using their local cache
  - Route Reply *Storm* problem
- Stale caches will lead to increased overhead

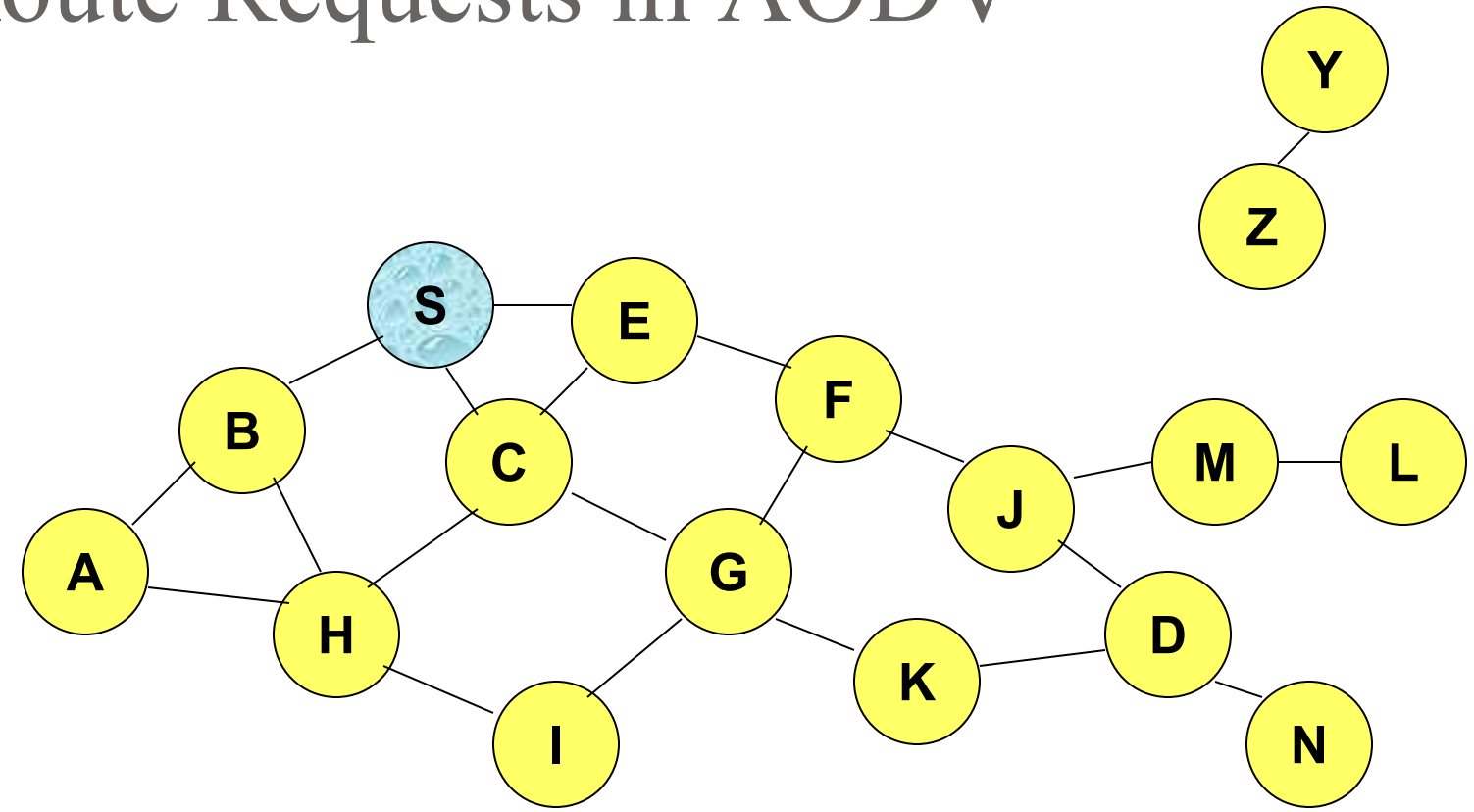
Ad Hoc On-Demand Distance Vector Routing (**AODV**)

- DSR includes source routes in packet headers
- Resulting large headers can sometimes degrade performance
  - particularly when data contents of a packet are small
- AODV attempts to improve on DSR by maintaining routing tables at the nodes, so that data packets do not have to contain routes
- AODV retains the desirable feature of DSR that routes are maintained only between nodes which need to communicate

AODV

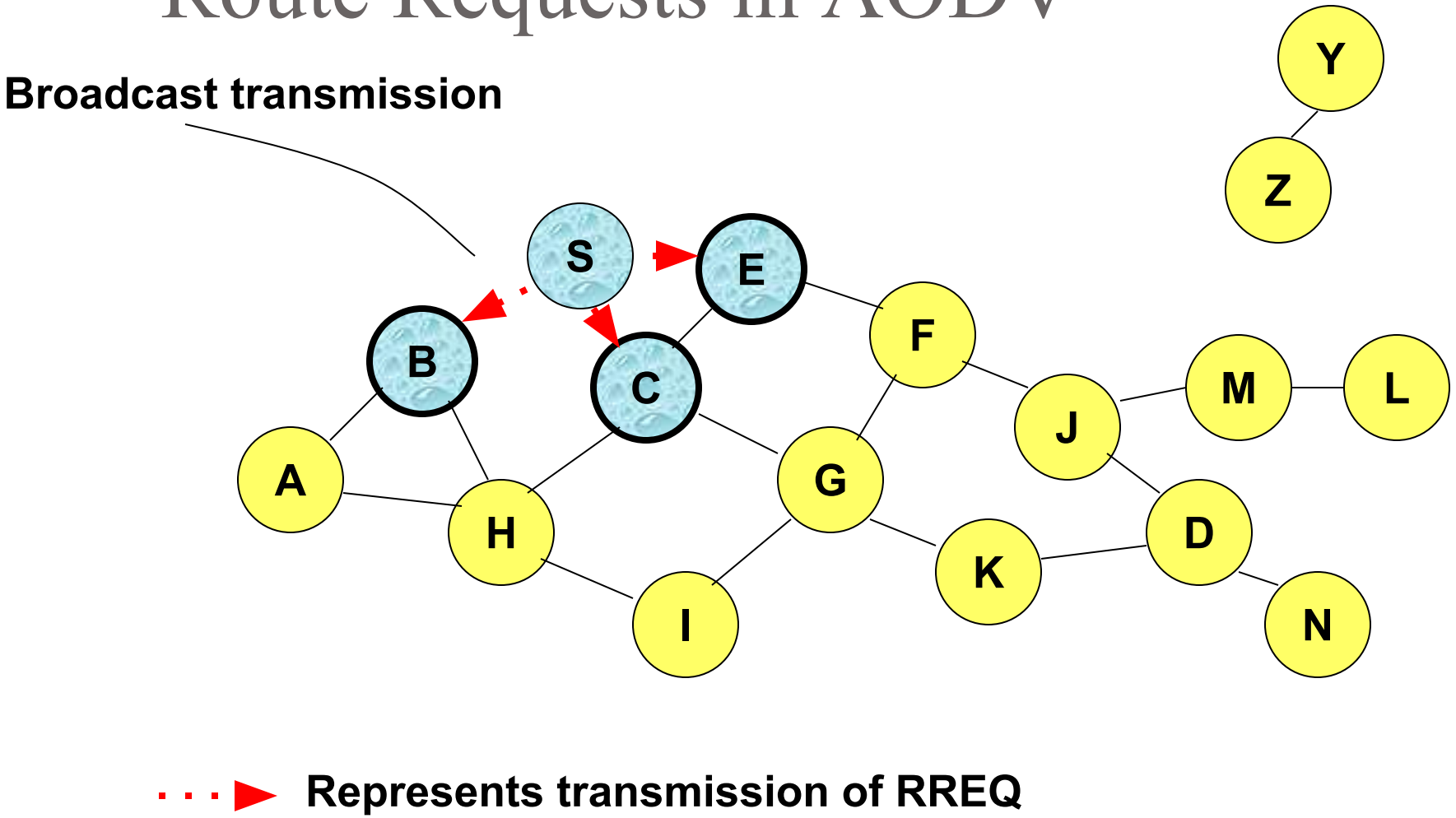
- Route Requests (RREQ) are forwarded in a manner similar to DSR
- When a node re-broadcasts a Route Request, it sets up a reverse path pointing towards the source
  - AODV assumes symmetric (bi-directional) links
- When the intended destination receives a Route Request, it replies by sending a Route Reply (RREP)
- Route Reply travels along the reverse path set-up when Route Request is forwarded

Route Requests in AODV

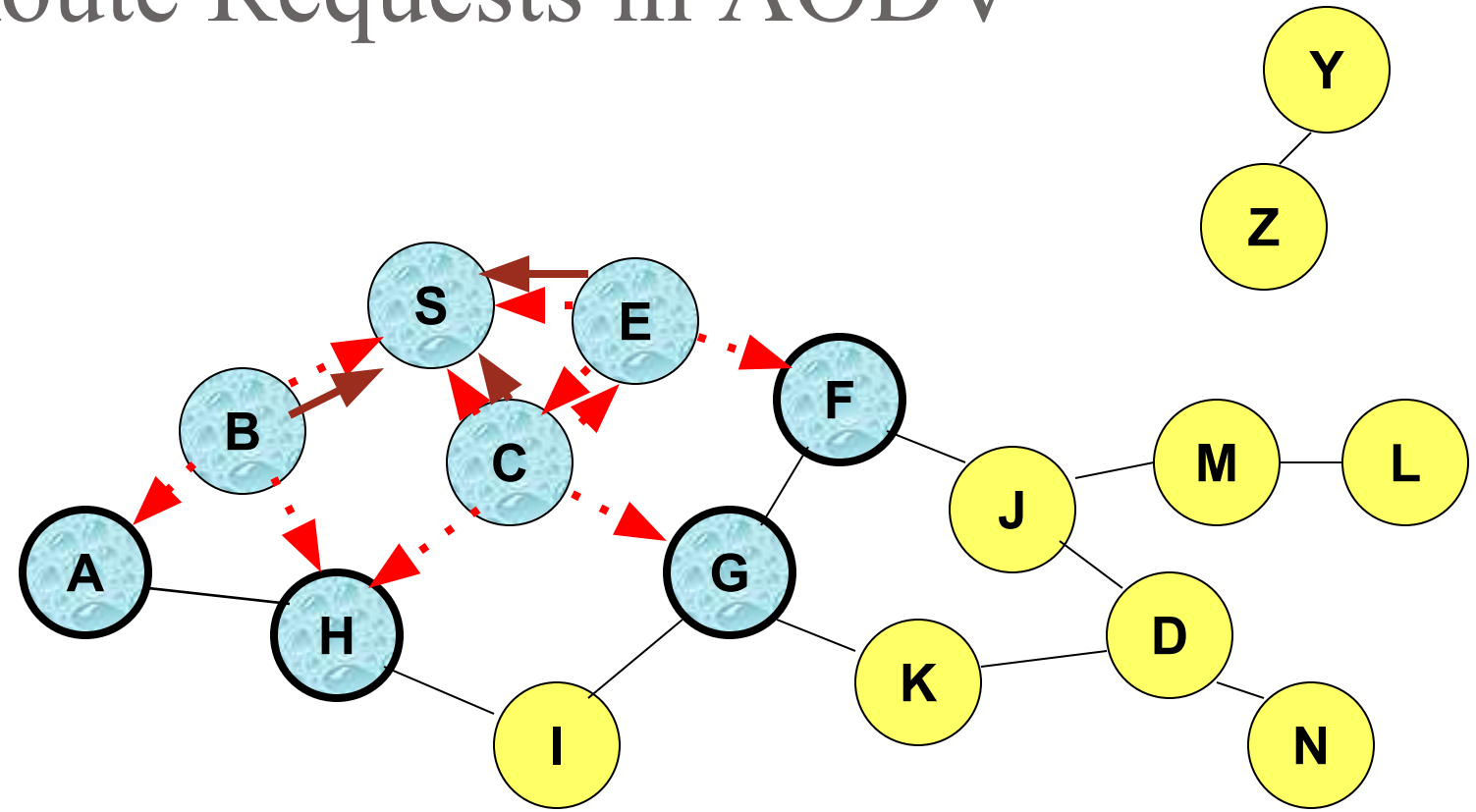


Represents a node that has received RREQ for D from S

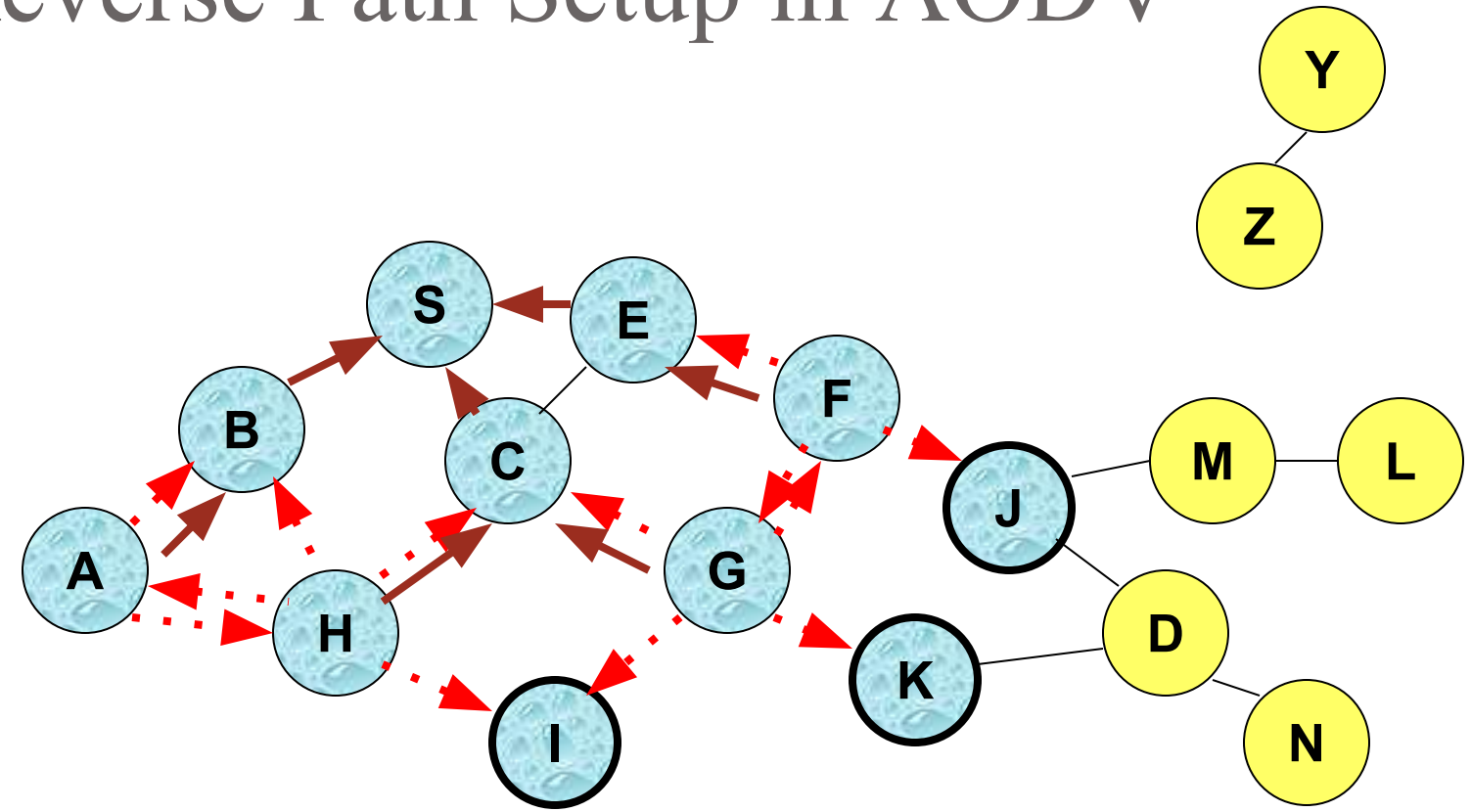
Route Requests in AODV



Route Requests in AODV

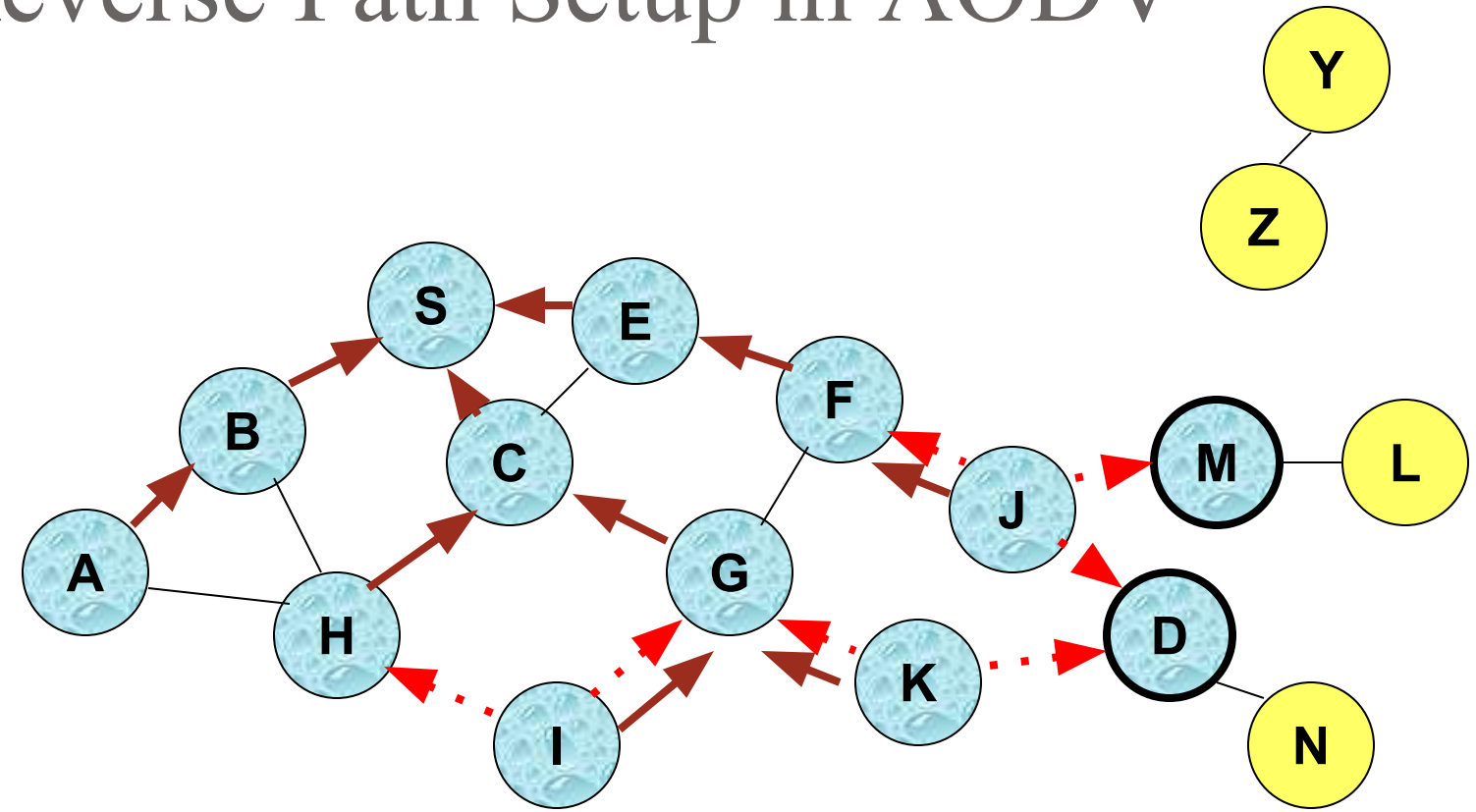


Reverse Path Setup in AODV

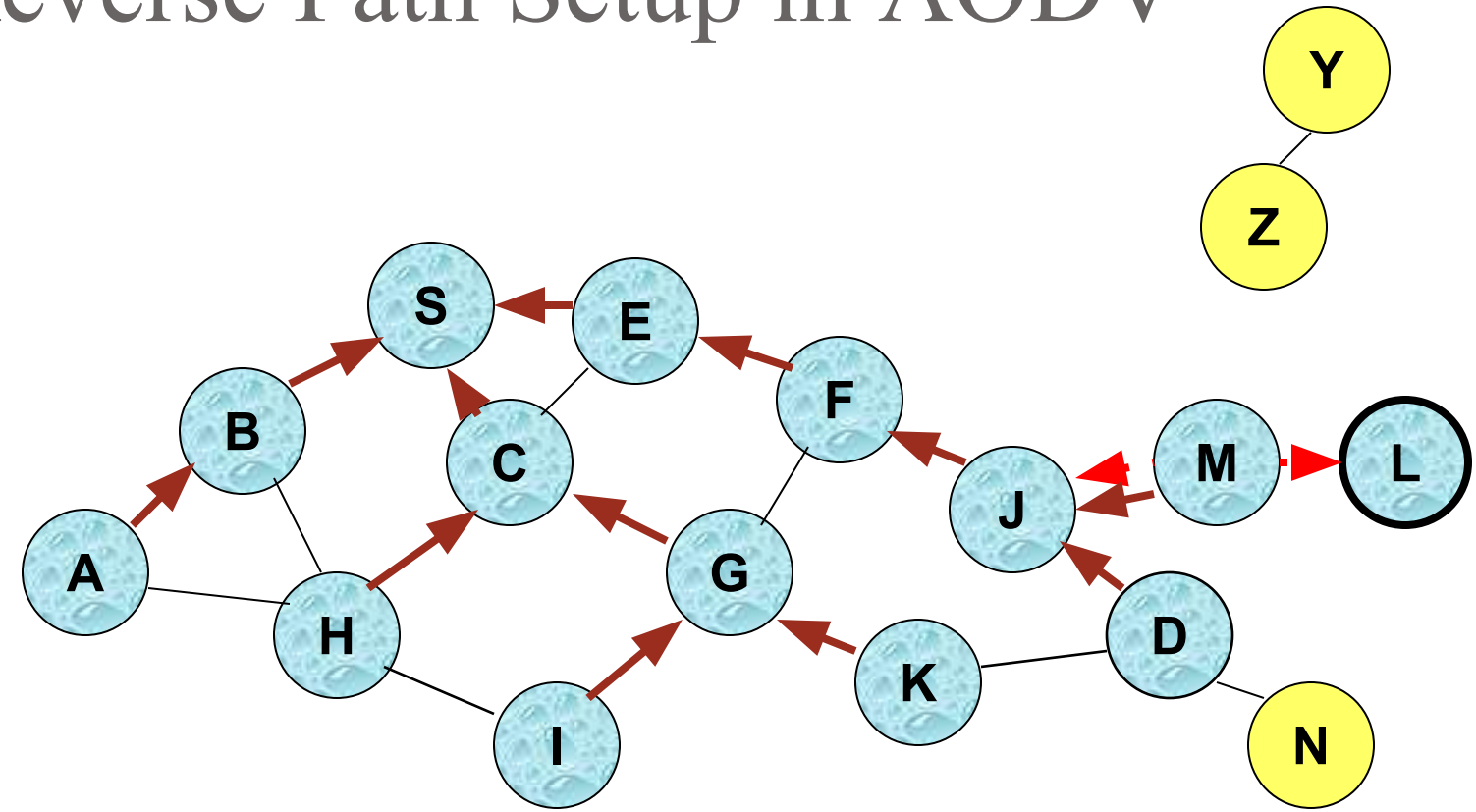


- Node C receives RREQ from G and H, but does not forward it again, because node C has already forwarded RREQ once

Reverse Path Setup in AODV

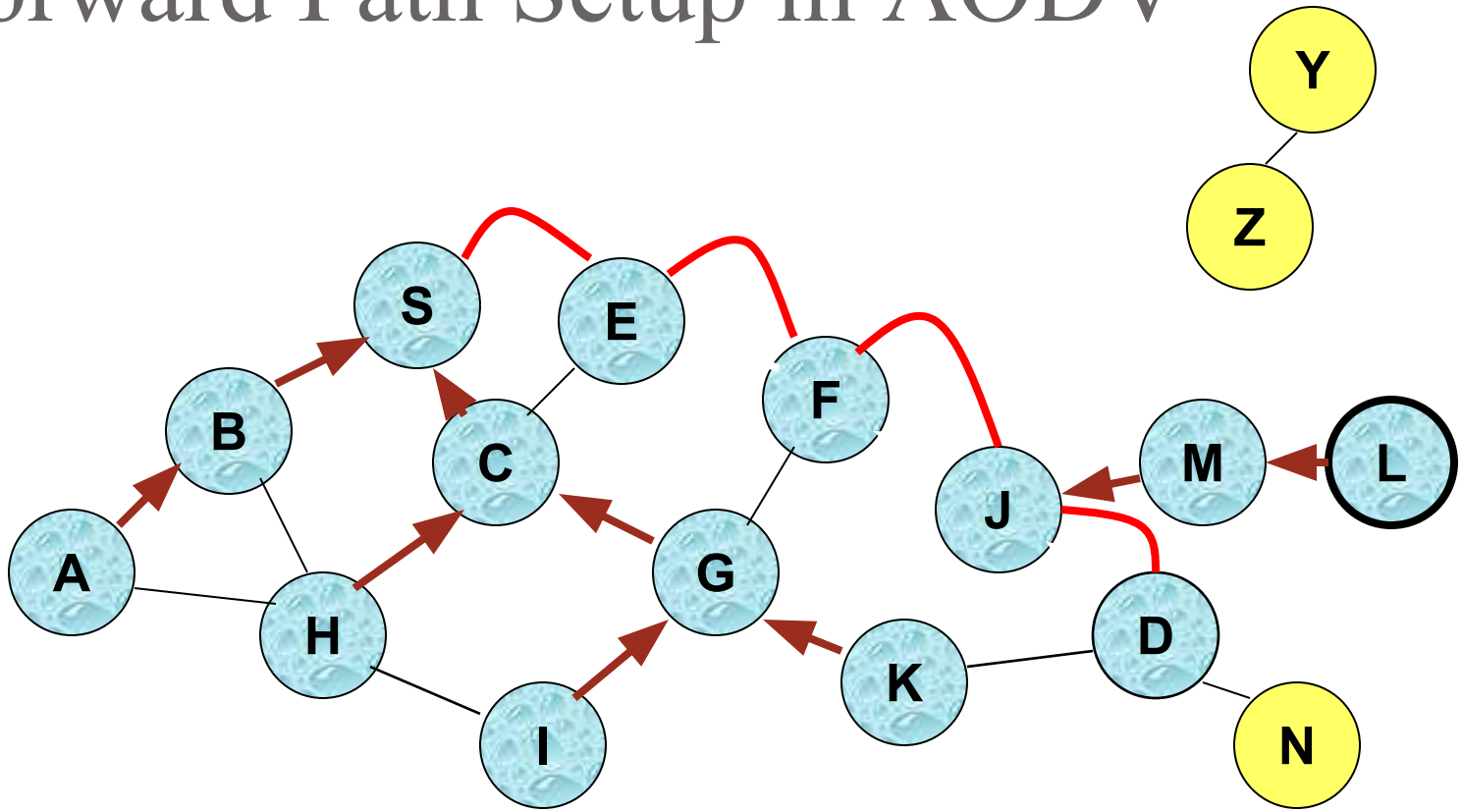


Reverse Path Setup in AODV



- Node D **does not forward** RREQ, because node D is the **intended target** of the RREQ

Forward Path Setup in AODV



Forward links are setup when RREP travels along the reverse path

Represents a link on the forward path

Route Request and Route Reply

- Route Request (RREQ) includes the last known sequence number for the destination
- An intermediate node may also send a Route Reply (RREP) provided that it knows a more recent path than the one previously known to sender
- Intermediate nodes that forward the RREP, also record the next hop to destination
- A routing table entry maintaining a reverse path is purged after a timeout interval
- A routing table entry maintaining a forward path is purged if *not used* for a *active_route_timeout* interval

Link Failure

- A neighbor of node X is considered active for a routing table entry if the neighbor sent a packet within *active_route_timeout* interval which was forwarded using that entry
- Neighboring nodes periodically exchange hello message
- When the next hop link in a routing table entry breaks, all active neighbors are informed
- Link failures are propagated by means of Route Error (RERR) messages, which also update destination sequence numbers

Route Error

- When node X is unable to forward packet P (from node S to node D) on link (X,Y), it generates a RERR message
- Node X increments the destination sequence number for D cached at node X
- The incremented sequence number *N* is included in the RERR
- When node S receives the RERR, it initiates a new route discovery for D using destination sequence number at least as large as *N*
- When node D receives the route request with destination sequence number N, node D will set its sequence number to N, unless it is already larger than N

AODV: Summary

- Routes need not be included in packet headers
- Nodes maintain routing tables containing entries only for routes that are in active use
- At most one next-hop per destination maintained at each node
  - DSR may maintain several routes for a single destination
- Sequence numbers are used to avoid old/broken routes
- Sequence numbers prevent formation of routing loops
- Unused routes expire even if topology does not change

References

- ece626web.groups.et.byu.net/Lectures/
- http://www.mhhe.com/engcs/compsci/forouzan/dcn/student/olc/powerpoints13.mhtml
- http://highered.mheducation.com/sites/0072460601/student_view0/chapter5/powerpoint_slides.html
- http://www.mhhe.com/engcs/compsci/forouzan/dcn/
- https://www.slideshare.net/tameemyousaf/switching-techniques
- https://www.tutorialspoint.com/data_communication_computer_network/physical_layer_switching.htm
- https://www.slideshare.net/vipinsahu/mpls-multiprotocol-label-switching
- *www.it.iitb.ac.in/~sri/talks/manet.ppt*

SRI CHANDRASEKHARENDRA SARASWATHI VISWA MAHAVIDHYALAYA
(Deemed to be university u/s 3 of UGC act 1956)
(Accredited with "A" by NAAC)
Enathur, Kanchipuram – 631561. Tamilnadu
www.kanchiuniv.ac.in

Unit V Application layer

Name of the Faculty : R.RADHIKA
Assistant Professor, Dept. of CSE
E-Mail : rradhika@kanchiuniv.ac.in

OBJECTIVE

1. To understand the concept of computer network and Application layer.
2. To understand how Internet Protocol suite and OSI model use the application layer.
3. Discuss various protocols in Application layer.

COURSE OUTCOME

1. Recognize the technological trends of application protocol layer.
2. Evaluate the challenges in building networks and solutions to those.
3. Configure DNS DDNS, TELNET, EMAIL, File Transfer Protocol (FTP),
- WWW,HTTP, SNMP, Bluetooth, Firewalls using open source available software and tools.

INTRODUCTION

- In computer networking, the layered concept of networking was developed to accommodate changes in technology. Each layer of a specific network model may be responsible for a different function of the network. Each layer will pass information up and down to the next subsequent layer as data is processed.
- The **application layer** is the top-most layer in the OSI Model and is used for establishing process-to-process communication and user services in a network. It's the interface between user applications and the underlying network. Whether you open a web page in a browser or read an email, you are interacting with the application layer of the network. In short, it's a layer which involves human interaction with applications and software to connect users together across the globe.
- A protocol is a set of rules used to communicate between systems in a network. Although the application layer is the medium through which you are able to communicate with other users, a set of protocols are required to assist with this communication. For example, if you have to open a web page, you need the HTTP or HTTPS protocols. Similarly, you would require POP3 or IMAP and SMTP for sending and receiving emails..

REAL TIME EXAMPLE

- ***WEB BROWSER***
- A web browser such as Internet Explorer or Netscape provides the means for your computer to contact a web server and download several files that go together to produce a single web page.

- You can request a web page by typing in a web address (a URL) or by clicking a link in an open web page. The web browser is an **APPLICATION.** The web browser *application* gives you the means to select a web server, contact the server and request a web page. The web browser handles the process of finding the web server (the remote computer that has the web page you want stored on it) , requesting the desired web page and displaying all the files contained within the web page.

Let us take a look at the various types of protocols with their uses

Application Type	Application-layer protocol	Transport Protocol
Electronic mail	Send: Simple Mail Transfer Protocol SMTP [RFC 821]	TCP 25
	Receive: Post Office Protocol v3 POP3 [RCF 1939]	TCP 110
Remote terminal access	Telnet [RFC 854]	TCP 23
World Wide Web (WWW)	HyperText Transfer Protocol 1.1 HTTP 1.1 [RFC 2068]	TCP 80
File Transfer	File Transfer Protocol FTP [RFC 959]	TCP 21
	Trivial File Transfer Protocol TFTP [RFC 1350]	UDP 69
Remote file server	NFS [McKusik 1996]	UDP or TCP
Streaming multimedia	Proprietary (e.g., Real Networks)	UDP or TCP
Internet telephony	Proprietary (e.g., Vocaltec)	Usually UDP

ELECTRONIC MAIL

Electronic mail or E-mail as it is popularly called is a system that allows a person or a group to electronically communicate with each other through a network. Presently people can now receive and send e-mail to:

- Nearly any country in the world.
- One of millions of computer users.
- Many users at once.
- Computer programs.

The first e-map systems consisted of file transfer protocols, with the convention that the first line of each message contained the recipient address.

Some of the complaints at that time were

1. Sending a message to a group of people was inconvenient.
2. Messages had no internal structure, making computer processing difficult.
3. The sender never knew if a message arrived or not.
4. It is difficult to forward the mails
5. It is not possible to create and send messages containing a mixer of text, drawing facsimile and voice

After a decade of competition, email systems based on RFC822 are widely used, where all the above problems are solved.

BASIC FUNCTION

- Email systems support five basic functions, which are:

1. **Composition** is a process for creating the messages and answers. This can be done by text editor, outside the mailer, the system will provide assistance in addressing and numerous header fields attached to each message. For eample:when answering a message, the e mail system can extract the originator's address from the incoming e-mail and automatically insert it into the address space in reply.
2. **Transfer** refers to moving of messages from the source to the recipent. In some cases, connection establishment is needed with the destination, outputting the

- message and releasing the connection. The e-mail system should do automatically this.

3. **Reporting** is used to indicate the originator what happened to the message i.e.,
- confirmation of the message delivery. Was it delivers successfully? Was it rejected? Was it lost? Did errors occur?

3. **Displaying** It refers to read the incoming e-mail by the person. Sometimes conversion is required or a special viewer must be invoked.

4. **Disposition** It concerns what the recipient does with the message after receiving it. The possibilities are

(a) Throwing it away before reading
(b) Throwing it away after reading.
(c) Saving it and so on. It is also possible to forward them or process them in other ways.

In addition to these basic services, most of e-mail systems provide a large variety of advanced features such as

(a) It allows to create a mailbox to store incoming e-mail.
(b) It allows to have a mailing list, to which the e-mail messages have to send.
(c) Carbon copies, high priority email, secret email, registered email etc.

THE USER AGENT

The user agent is a program that allows users to read reply to, forward, save and compose messages. User agents for electronic mail are sometimes called mail readers. Some user agents have menu or icon driven interface that requires a mouse, some other requires only 1 character command from keyboard.

Sending e-mail: To send an email message the user must provide

message

(a) destination address and

(b) priority or security levels (options).

Message can be produced with a free standing text editor, a word processing program or by using a text editor built into the user agents. The format of an e-mail message is similar to that of a conventional letter.

There are two main parts: Header and body.

The header contains out name and address, the name and address of the person it's being sent to, the name and address of the person who is being sent a copy, the date of the message and the subject when we receive an e-mail from someone, the header tells us where it came from, what it is about, how it was sent and when.

The body is the place where we write the contents of what we want to

communicate. The message sent should be simple and direct. Body is entirely for human recipient.

The designation address must be in a format that the user agent can deal with. The basic form of e-mail address is

User name @host name.subdomain.domain.

The text before the sign @(pronounced “at”) specifies the user name of the individual, the text after the @ sign indicates how the computer system can locate that individual’s mailboxes.

For example mvs@cs.colorado.edu

Here cs is a sub domain of Colorado is a sub domain of edu. The edu specifies the top-level domain name.

The number of periods (pronounced as dots) varies from e-mail address. Reading e-mail: On connecting to the net, the first thing a user usually does is check his mail, it’s like checking the mailbox when we go home

MESSAGE FORMATS

The e-mail message format was defined in RFC 822.There are two types: ASCII e-mail and multimedia extensions.

ASCII e-mails using RFC 822: The e-mail message consists of a primitive

envelope, some number of header fields, a blank like and then message body.

Each header field consists of a single line of ASCII text containing the field name, a colon, and a value of RFC.

The lists of header fields related to message transport are

A recipient's address or "To"

A sender's address or "From"

A subject.

The email header may additionally contain.

A List of "CC": This is a list of e-mail or 'carbon copies' addresses to whom a copy of the message is to be delivered. Multiple e-mail addresses in the "CC"

fields are separated by a comma.

A List of “BCC”: This is same as “CC” except that this is a carbon copy. The list of recipients is not visible to the person who receives this message

Attached: This is a convenient method to share both data and programs. These files may be attached or enclosed with an e-mail message.

Signature: It contains sender’s full name and address or whatever information the sender wishes to send. Instead of creating a message from the scratch, we

may choose to reply or forward the messages.

Replying: When we reply a message, the sender’s address is automatically put in the “To” header and subject of the original message is reduced proceeded by Re, for the reply.

Forwarding: When we forward a message, the subject of the original message

is reused, with prefix “FW”. We must specify the e-mail address of the recipient of the forward message.

Redirecting: Some e-mail programs allow to redirect messages. It is similar to forwarding a message, except that the message retains the original sender in the form header and adds a notation that the message comes through you.

SMTP (Simple Mail Transfer Protocol):

One of the most popular network service is electronic mail (e-mail).
The TCP/IP protocol that supports electronic mail on the Internet is called Simple Mail Transfer Protocol (SMTP).

SMTP transfers messages from senders' mail servers to the recipients' mail servers using TCP connections.

Users based on e-mail addresses.
SMTP provides services for mail exchange between users on the same or different computers.
Following the client/server model:
SMTP has two sides: a client side which executes on a sender's mail server, and server side which executes on recipient's mail server.

Both the client and server sides of SMTP run on every mail server.
When a mail server sends mail (to other mail servers), it acts as an SMTP client.
When a mail server receives mail (from other mail servers) it acts as an SMTP server

Multipurpose Internet Mail Extensions (MIME):

- It is an extension of SMTP that allows the transfer of multimedia messages.
- If binary data is included in a message MIME headers are used to inform the receiving mail agent:

- The MIME defines five new message headers
- MIME-Version: It tells the use agent receiving the message that it is dealing with a MIME message, and which version of MIME it uses.
- Content-Description: It tells what is there in the message, this header helps the recipient whether it is worth decoding and reading the message.
- Content-Transfer Encoding: It tells how the body is wrapped for transmission through a network that may object to most characters other than letters, numbers and punctuation mark
- Content-Type: It specifies the nature of the message body. Seven types are defined in RFC 1521, each of which has one or more sub types. The type and sub type are separated by a slash. The sub type must be given explicitly in the header, no defaults are provided

- Example:

Text plain, html, rich text

Image GIF,JPEG

Audio au, basic

- **POP (Post Office Protocol):**
  - POP is also called as POP3 protocol.
  - This is a protocol used by a mail server in conjunction with SMTP to receive and holds mail for hosts.
  - POP3 mail server receives e-mails and filters them into the appropriate user folders. When a user connects to the mail server to retrieve his mail, the messages are downloaded from mail server to the user's hard disk.

POP_3 progress through three phases.

- **Authorization:** The user agent sends a user name and a password to authenticate the user downloading the mail.
- **Transaction:** The user agent receives messages. In this phase the user agent can also mark messages for deletion, remove deletion marks, and obtain mail statistics.
- **Update:** During the third phase, update occurs after the client has issued the quit command, ending the POP_3 session. This time the mail server deletes the messages that were marked for deletion.

TELNET (Terminal Network):

- TELNET is client-server application that allows a user to log onto remote machine and lets the user to access any application program on a remote computer.
- TELNET uses the NVT (Network Virtual Terminal) system to encode characters on the local system.
- On the server (remote) machine, NVT decodes the characters to a form acceptable to the remote machine.
- TELNET is a protocol that provides a general, bi-directional, eight-bit byte oriented communications facility.
- Many application protocols are built upon the TELNET protocol
- Telnet services are used on PORT 23

FTP (File Transfer Protocol):

- FTP is the standard mechanism provided by TCP/IP for copying a file from one host to another.
- FTP differs form other client-server applications because it establishes 2 connections between hosts.
- Two connections are: Data Connection and Control Connection.
- Data Connection uses PORT 20 for the purpose and control connection uses PORT 21 for the purpose.
- FTP is built on a client-server architecture and uses separate control and data connections between the client and the server.
- One connection is used for data transfer, the other for control information (commands and responses).
- It transfer data reliably and efficiently.

HTTP (Hypertext Transfer Protocol):

- This is a protocol used mainly to access data on the World Wide Web (www).
- The Hypertext Transfer Protocol (HTTP) the Web's main application-layer protocol although current browsers can access other types of servers
- A repository of information spread all over the world and linked together.
- The HTIP protocol transfer data in the form of plain text, hyper text, audio, video and so on.
- HTTP utilizes TCP connections to send client requests and server replies.
- it is a synchronous protocol which works by making both persistent and non persistent connections

Domain Name System (DNS):

- To identify an entity, TCP/IP protocol uses the IP address which uniquely identifies the connection of a host to the Internet.
- DNS is a hierarchical system, based on a distributed database, that uses a hierarchy of Name Servers to resolve Internet host names into the corresponding IP addresses required for packet routing by issuing a DNS query to a name server.
- However, people refer to use names instead of address. Therefore, we need a system that can map a name to an address and conversely an address to name.
- In TCP/IP, this is the domain name system.
- DNS in the Internet: DNS is protocol that can be used in different platforms. Domain name space is divided into three categories.
- Generic Domain: The generic domain defines registered hosts according, to their

- Generic behaviour. Each node in the tree defines a domain which is an index to the domain name space database.
- Country Domain: The country domain section follows the same format as the generic domain but uses 2 characters country abbreviations (e.g., US for United States) in place of 3 characters.
- Inverse Domain: The inverse domain is used to map an address to a name

BLUTOOTH

Introduction:-Bluetooth wireless technology is a *short-range radio technology*, which is developed for Personal Area Network (PAN). Bluetooth is a standard developed by a group of electronics manufacturers that allows any sort of electronic equipment -- from computers and cell phones to keyboards and headphones -- to make its own connections, without wires, cables or any direct action from a user.

- It is an ad hoc type network operable over a small area such as a room. Bluetooth wireless technology makes it possible to transmit signals over short distances between telephones, computers and other devices and thereby simplify communication and synchronization between devices. It is a global standard that:
  - Eliminates wires and cables between both stationary and mobile devices
  - Facilitates both data and voice communication
  - Offers the possibility of ad hoc networks and delivers the ultimate synchronicity between all your personal devices
- Bluetooth is a dynamic standard where devices can automatically find each other, establish connections, and discover what they can do for each other on an ad hoc basis.

- Bluetooth is intended to be a standard that works at two levels:
  - It provides agreement at the physical level -- Bluetooth is a radio-frequency standard.
  - It also provides agreement at the next level up, where products have to agree on when bits are sent, how many will be sent at a time and how the parties in a conversation can be sure that the message received is the same as the message sent.

It is conceived initially by Ericsson, before being adopted by a myriad of other companies, Bluetooth is a standard for a **small, cheap radio chip to be plugged into computers, printers, mobile phones, etc**.

A Bluetooth chip is designed to replace cables by taking the information normally carried by the cable, and transmitting it at a special frequency to a receiver Bluetooth chip, which will then give the information received to the computer, phone whatever.

- **Topology**
- There are two types of topology for Bluetooth – Piconet, Scatternet. The Piconet is a small ad hoc network of devices (normally 8 stations) as shown in Fig. 5.8.1. It has the following features:

- One is called **Master** and the others are called **Slaves**
- All slave stations synchronizes their clocks with the master
- Possible communication - One-to-one or one-to-many
- There may be one station in *parked state*
- Each piconet has a **unique hopping pattern/ID**
- Each **master** can connect to **7 simultaneous** or **200+ inactive (parked) slaves** per piconet

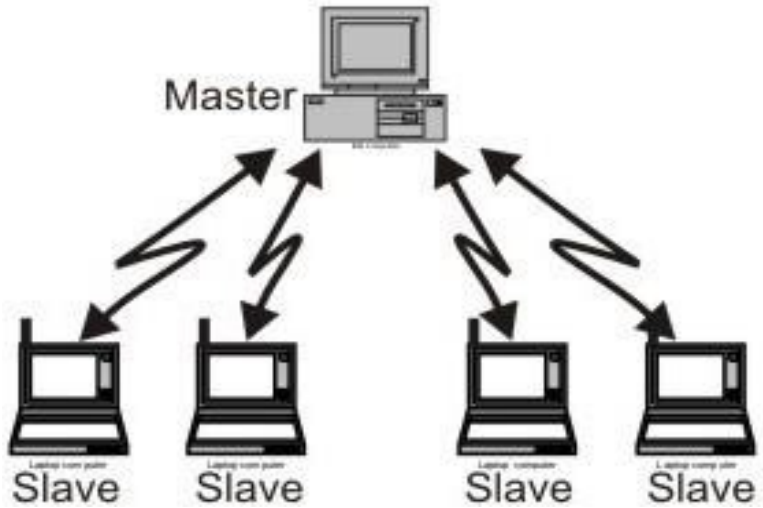


Figure : Piconet topology of Bluetooth

- By making one slave as master of another Piconet, Scatter net is formed by combining several Piconets as shown in Fig. 5.8.2. Key features of the scatter net topology are mentioned below:

  - A **Scatter net** is the **linking** of multiple **co-located piconets**
- through the sharing of common master or slave devices.
  - A device can be both a **master** and a **slave.**
  - Radios are **symmetric** (same radio can be master or slave).
  - **High capacity system**, each piconet has maximum capacity (720 Kbps)

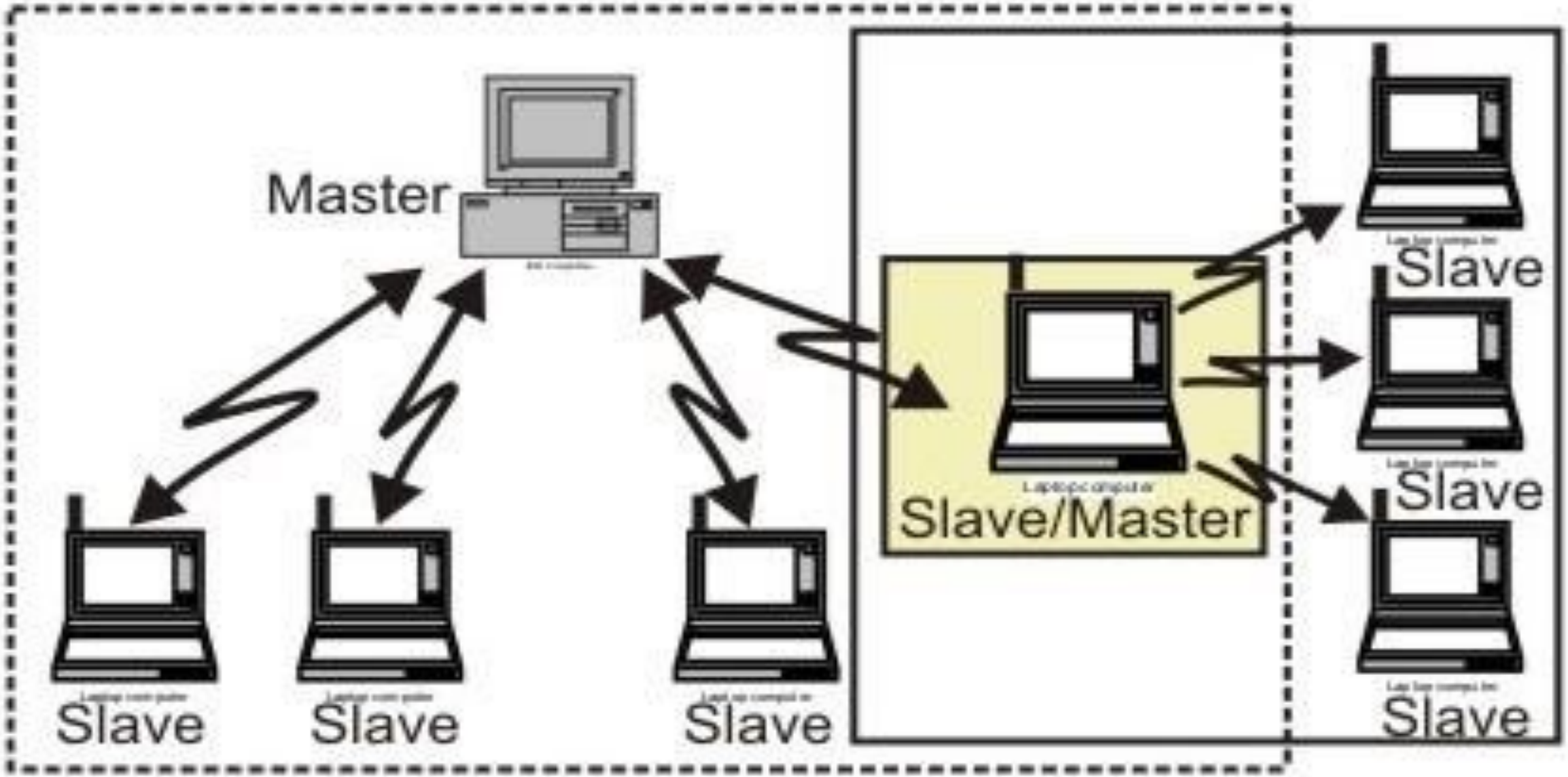


Figure: Scatternet topology

Bluetooth Architecture

- The Bluetooth architecture, showing all the major layers in the Bluetooth system, are depicted in the Fig. 5.8.3. The layers below can be considered to be different hurdles in an obstacle course. This is because all the layers function one after the other. One layer comes into play only after the data has been throug

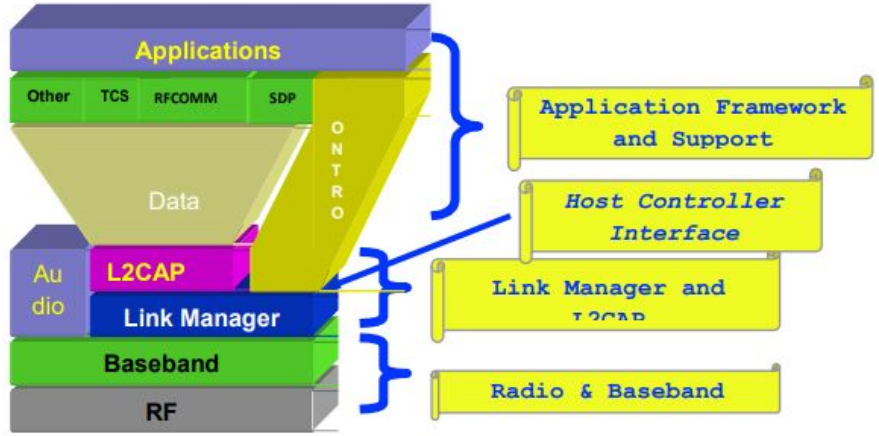


Figure : The Bluetooth architecture

- **Radio:** The Radio layer defines the requirements for a Bluetooth transceiver operating in the 2.4 GHz ISM band.
- **Baseband**: The Baseband layer describes the specification of the Bluetooth Link Controller (LC), which carries out the baseband protocols and other low- level link routines. It specifies Piconet/Channel definition, “Low-level” packet definition, Channel sharing
- LMP: The Link Manager Protocol (LMP) is used by the Link Managers (on either side) for link set-up and control.
- **HCI:** The Host Controller Interface (HCI) provides a command interface to the Baseband Link Controller and Link Manager, and access to hardware status and control registers.

- **L2CAP:** Logical Link Control and Adaptation Protocol (L2CAP) supports higher level protocol multiplexing, packet segmentation and reassembly, and the conveying of quality of service information.
- **RFCOMM:** The RFCOMM protocol provides emulation of serial ports over the L2CAP protocol. The protocol is based on the ETSI standard TS 07.10.
- **SDP:** The Service Discovery Protocol (SDP) provides a means for applications to discover, which services are provided by or available through a Bluetooth device. It also allows applications to determine the characteristics of those available services.

- Now we shall be study each layer in detail (in next few sections) so that we come to know the function of each layer.

CASE STUDY

1. Asked to present an email architectural and implementation solution for a small sized enterprise.

2. Case study for developing a website and hosting it on the web

MCQ TYPE QUESTION

1. The translates internet domain and host names to IP address.
    1. Domain name system
    2. Routing information protocol
    3. Network time protocol
    4. Internet relay chat
2. Application layer protocol defines
    1. Types of messages exchanged
    2. Message format, syntax and semantics
    3. Rules for when and how processes send and respond to messages
    4. All of the mentioned

3. When displaying a web page, the application layer uses the
   1. HTTP protocol
   2. FTP protocol
   3. SMTP protocol
   4. TCP protocol

4. When the mail server sends mail to other mail servers it becomes
   5. SMTP server
   6. SMTP client
   7. Peer
   8. Master

6. Expansion of SMTP is
   a) Simple Message Transfer Protocol
   d) Simple Mail Transmission Protocol
   e) Simple Message Transmission Protocol

SRI CHANDRASEKHARENDRA SARASWATHI VISWA MAHAVIDHYALAYA

(Deemed to be university u/s 3 of UGC act 1956)

(Accredited with "A" by NAAC)

Enathur, Kanchipuram – 631561. Tamilnadu

www.kanchiuniv.ac.in

CS602 – COMPUTER NETWORKS

Name of the Faculty : Dr. D.THAMARAISELVI

Assistant Professor, Dept. of CSE

E-Mail : dthamaraiselvi@kanchiuniv.ac.in

COURSE : TRANSPORT LAYER PROTOCOLS

PROGRAM : Computer Science and Engineering

DEGREE : B.E., VI Sem

PRE-REQUISIT

1. Basics of Computer.
2. Digital Circuits.

OBJECTIVES

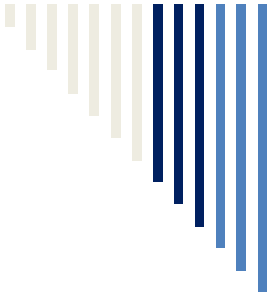
- To introduce TCP as a protocol that provides reliable stream delivery service.
- To define the format of a TCP segment and its fields.
- To show how TCP provides a connection-oriented service, and show the segments exchanged during connection establishment and connection termination phases.
- To discuss the state transition diagram for TCP and discuss some scenarios.
- To introduce windows in TCP that are used for flow and error control.

Course Outcomes

- Explain the functions of the Transport layer.
- Explain the working of online video streaming protocols.
- Explain different types of protocols used in real time applications.

OUTLINE

- Functions of transport layer
- Transport layer protocols
- TCP
- UDP
- STCP



TRANSPORT LAYER

TRANSPORT LAYER

- *The transport layer is responsible for process-to-process delivery—the delivery of a packet, part of a message, from one process to another. Two processes communicate in a client/server relationship.*

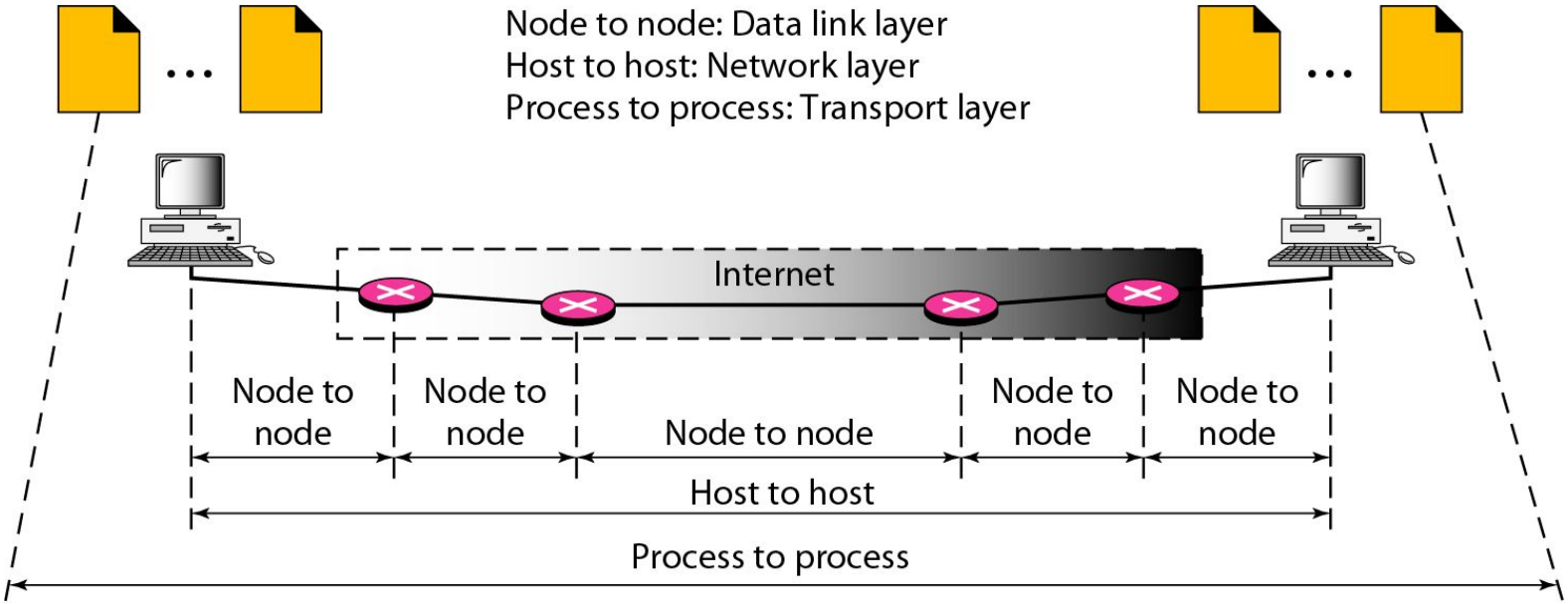
TRANSPORT LAYER SERVICES

- ✔Process-to-Process Communication
- ✔ Stream Delivery Service
- ✔ Full-Duplex Communication
- ✔ Multiplexing and Demultiplexing
- ✔ Connection-Oriented Service
- ✔ Reliable Service

INTRODUCTION

- ***Responsibilities of Transport Layer***
  - to create a process-to-process communication
    using port numbers in case of UDP
  - to provide a flow-and-error control mechanism at the transport level
    - But, no flow control mechanism and no acknowledgment for received packets in UDP
    - If UDP detects an error in the received packets, it silently drops it.
  - to provide a connection mechanism for the processes
    - sending streams of data to the transport layer by process
    - making the connection, chopping the stream into transportable units, numbering them and sending them one by one

TYPES OF DELIVERIES

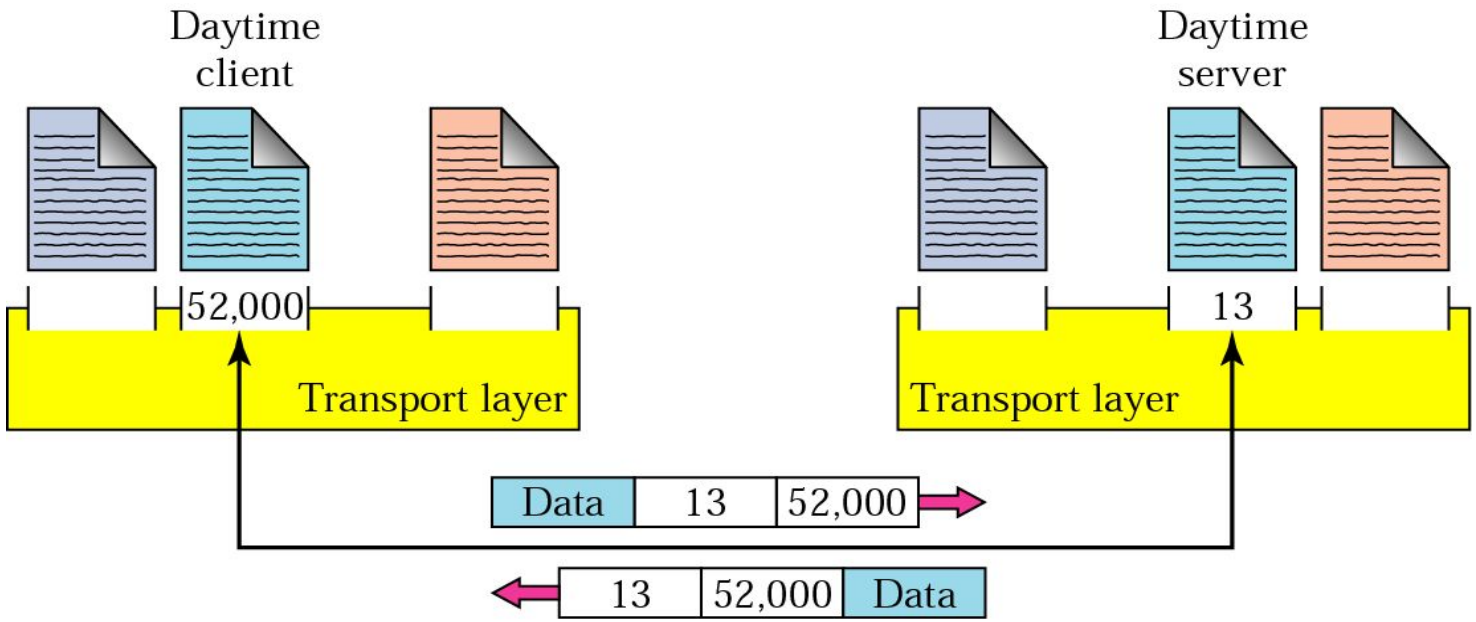


- Data link layer is responsible for delivery of frames between two neighboring nodes over a link ☐ Node-to-node delivery
- Network layer is responsible for delivery of datagrams between two hosts ☐ host-to-host delivery
- Real communication takes place between two processes (application programs). So, we need process-to-process delivery
- Transport layer is responsible for process-to-process delivery, the delivery of a packet, part of a message, from one process to another.

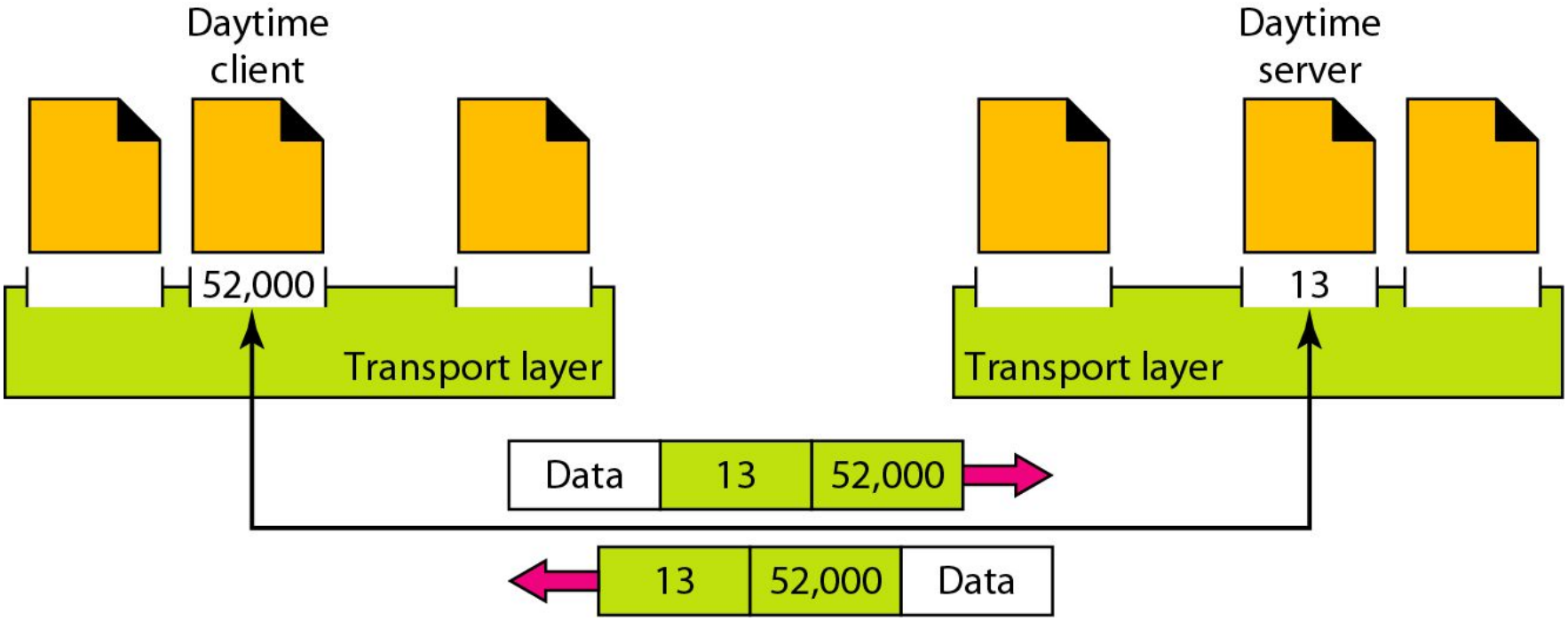
Transport Layer Addressing

Addresses

- Data link layer MAC address
- Network layer IP address
- Transport layer *Port number* (choose among multiple processes running on destination host)



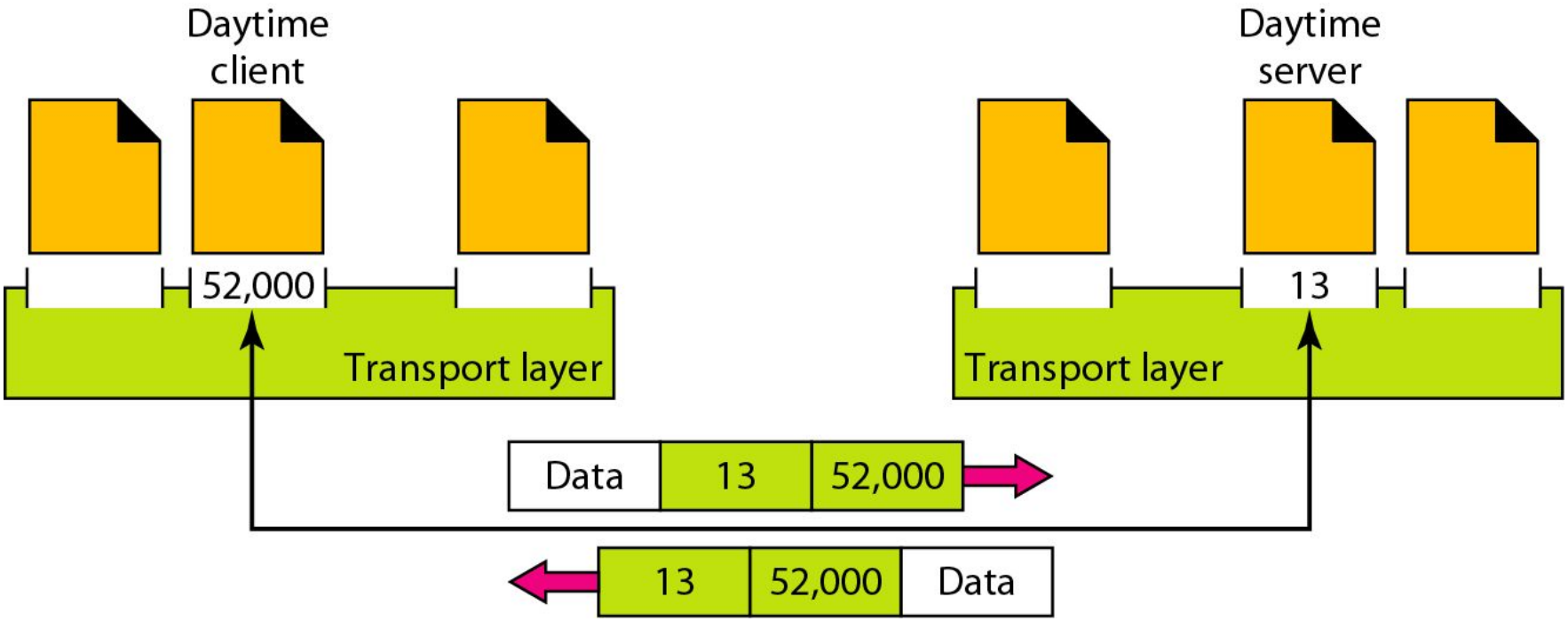
Port numbers



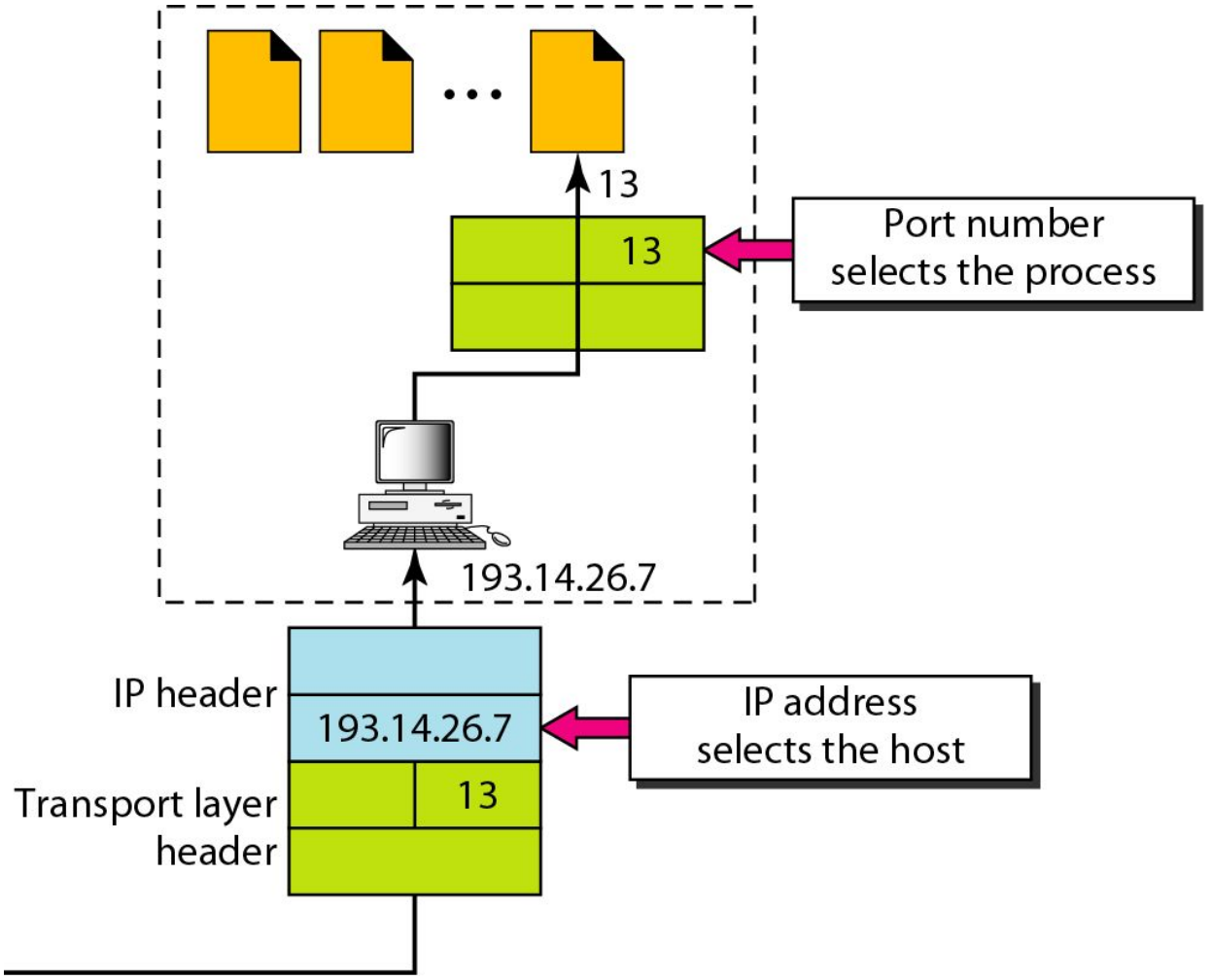
Port Numbers

- Port numbers are 16-bit integers (0 – 65,535)
  - Servers use *well know ports*, 0-1023 are privileged
  - Clients use *ephemeral* (short-lived) ports
- *Internet Assigned Numbers Authority* (IANA) maintains a list of port number assignment
  - Well-known ports (0-1023) – controlled and assigned by IANA
  - Registered ports (1024-49151) – IANA registers and lists use of ports as a convenience (49151 is ¾ of 65536)
  - Dynamic ports (49152-65535) – ephemeral ports
- For well-known port numbers, see /etc/services on a UNIX or Linux machine

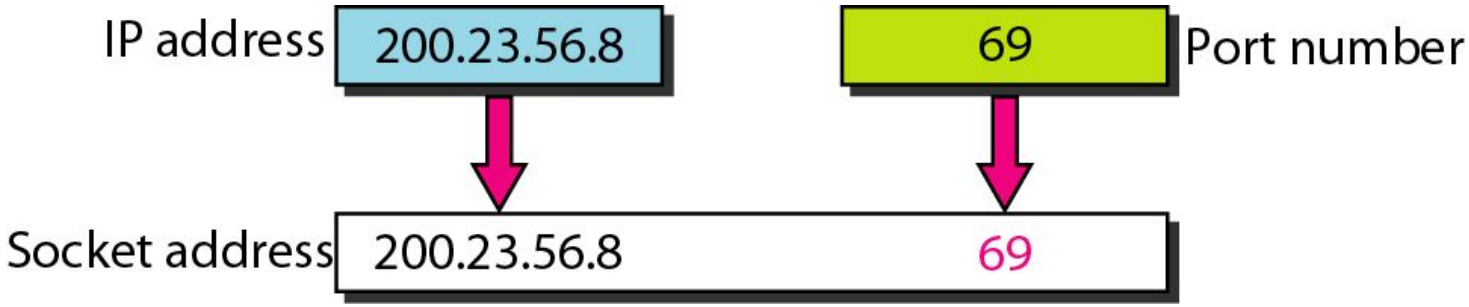
Port Numbers



IP addresses versus port numbers

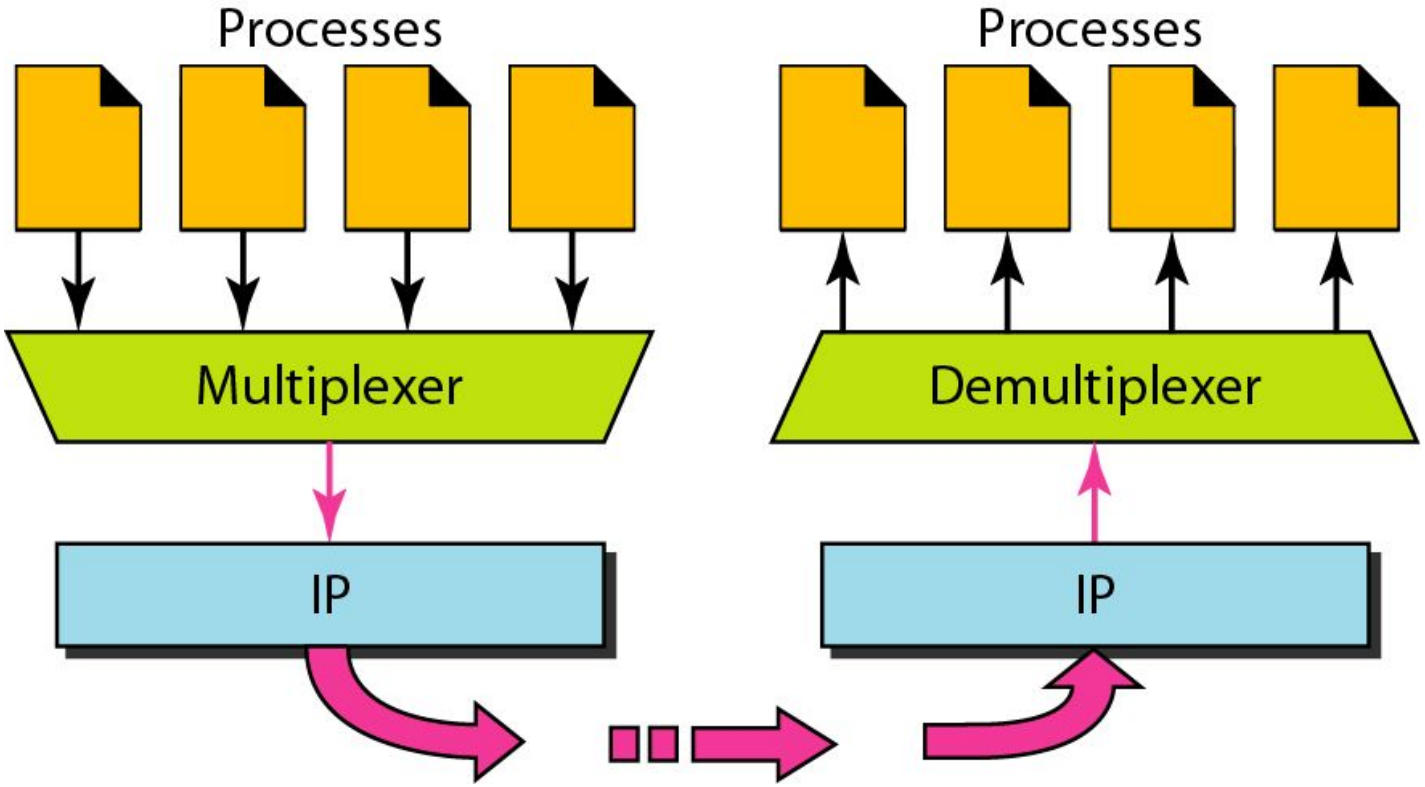


Sockets



- Combination of an IP address and a port number is called a socket address.
- Client socket address defines the client process uniquely just as the server socket address defines the server process uniquely.
- Transport layer protocol needs a pair of socket addresses: the client socket address and server socket address.

Multiplexing and demultiplexing



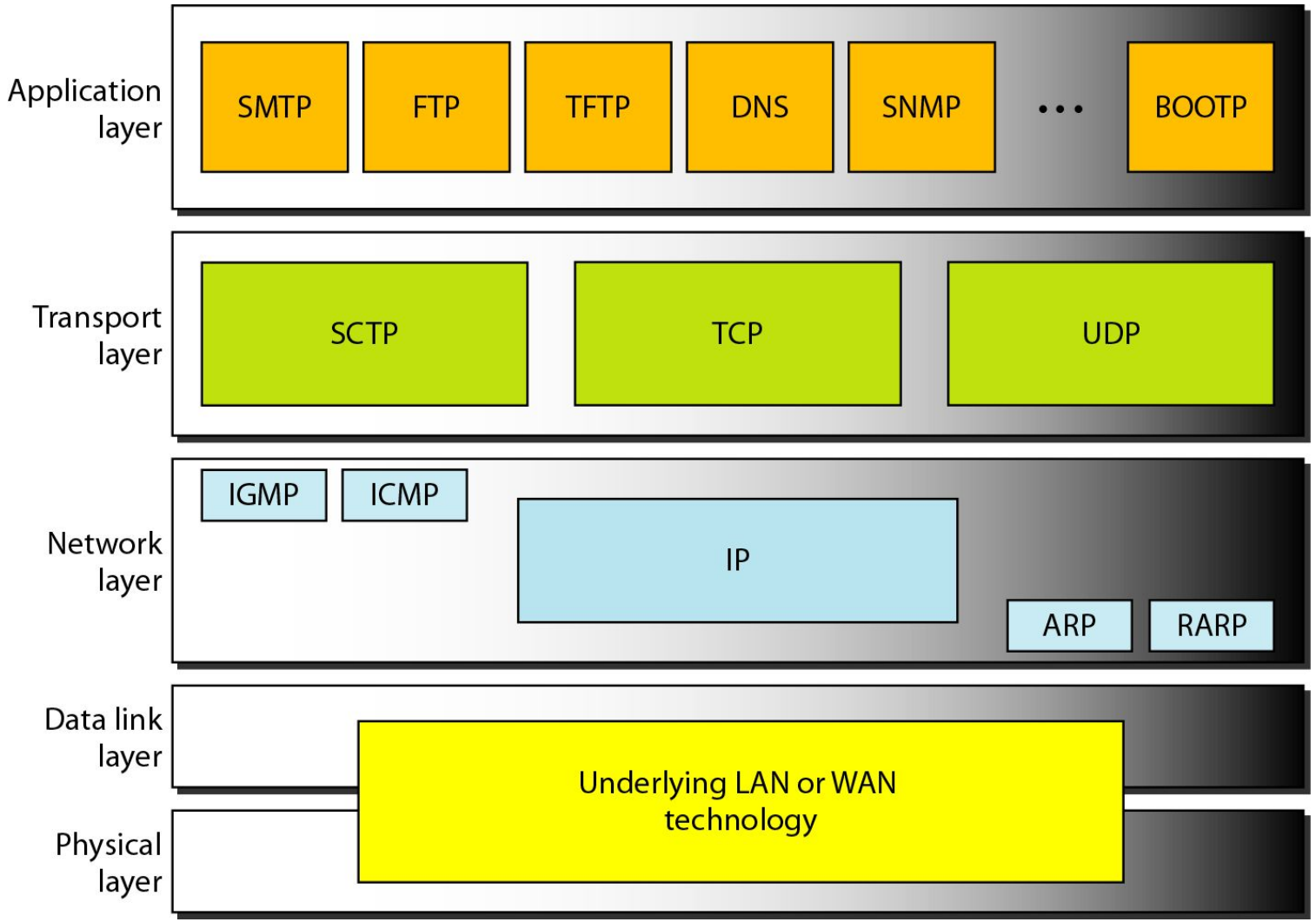
Socket Addressing

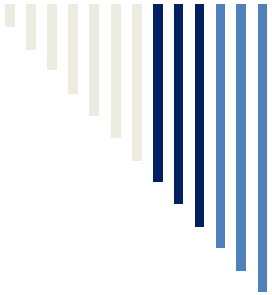
- Process-to-process delivery needs *two* identifiers
  - IP address and Port number
  - Combination of IP address and port number is called a socket address (a socket is a communication endpoint)
  - Client socket address uniquely identifies client process
  - Server socket address uniquely identifies server process
- Transport-layer protocol needs a *pair* of socket addresses
  - Client socket address
  - Server socket address
  - For example, socket pair for a TCP connection is a 4-tuple
    - ✔Local IP address, local port, and
    - ✔foreign IP address, foreign port

Connection Oriented and Connectionless

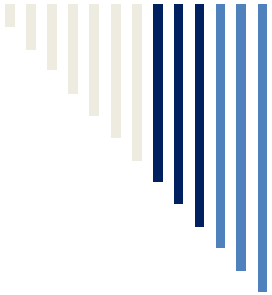
- Connectionless service
  - Packets are sent from one party to another with no need for connection establishment or connection release.
  - Packets are not numbered; they may be delayed, lost, or arrive out of sequence.
  - No acknowledgement.
  - UDP [User Datagram Protocol]
- Connection-oriented service
  - Connection is first established between the sender and the receiver
  - Data are transferred
  - At the end, the connection is released.
  - TCP [Transmission Control Protocol]

Position of UDP and TCP in TCP/IP suite





Transport layer protocols



TCP

TCP

TCP is a connection-oriented protocol; it creates a virtual connection between two TCPs to send data. In addition, TCP uses flow and error control mechanisms at the transport level.

Topics discussed in this section:

TCP Services
TCP Features
Segment
A TCP Connection
Flow Control
Error Control

Table 23.2 *Well-known ports used by TCP*

Port	*Protocol*	*Description*
7	Echo	Echoes a received datagram back to the sender
9	Discard	Discards any datagram that is received
11	Users	Active users
13	Daytime	Returns the date and the time
17	Quote	Returns a quote of the day
19	Chargen	Returns a string of characters
20	FTP, Data	File Transfer Protocol (data connection)
21	FTP, Control	File Transfer Protocol (control connection)
23	TELNET	Terminal Network
25	SMTP	Simple Mail Transfer Protocol
53	DNS	Domain Name Server
67	BOOTP	Bootstrap Protocol
79	Finger	Finger
80	HTTP	Hypertext Transfer Protocol
111	RPC	Remote Procedure Call

Stream Delivery

- Stream Data Service (stream transport layer service)
  - The sending TCP

    1) accepts a stream of characters from sending application program

    2) creates packets called *segments*, of appropriate size extracted from the stream

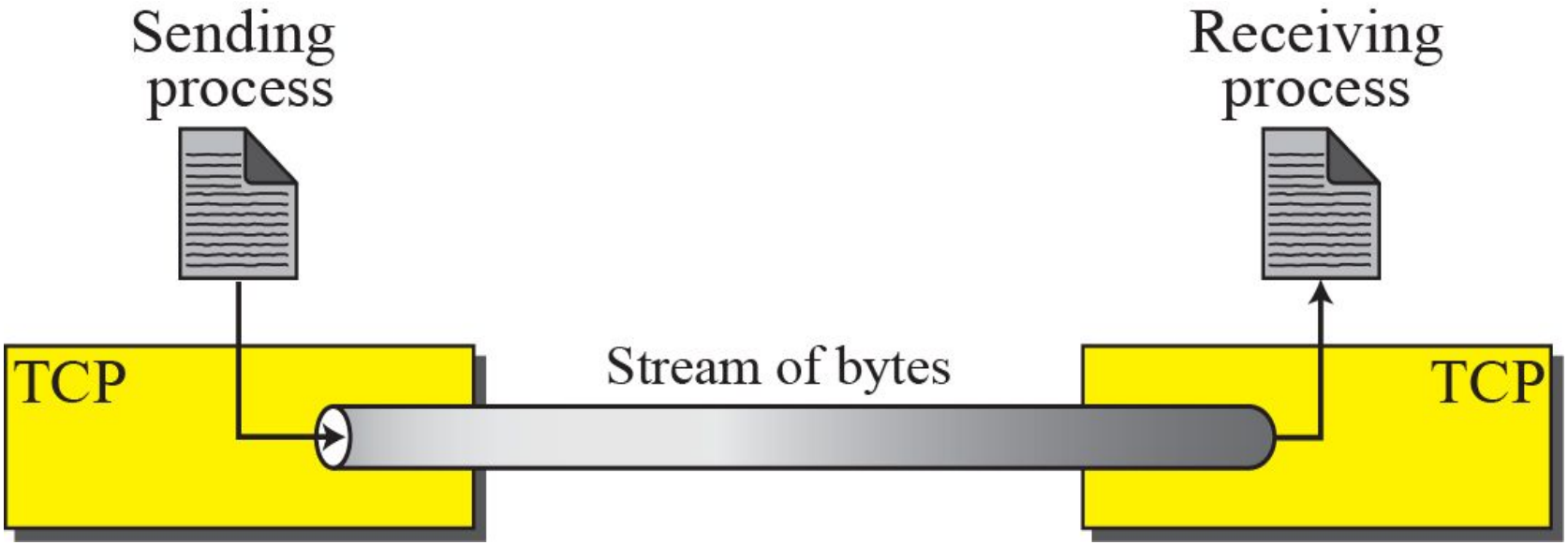
    3) sends segments across the network
  - The receiving TCP

    1) receives segments, extracts data from segments

    2) orders segments if they have arrived out of order

    3) delivers segments as a stream of characters to the receiving application program

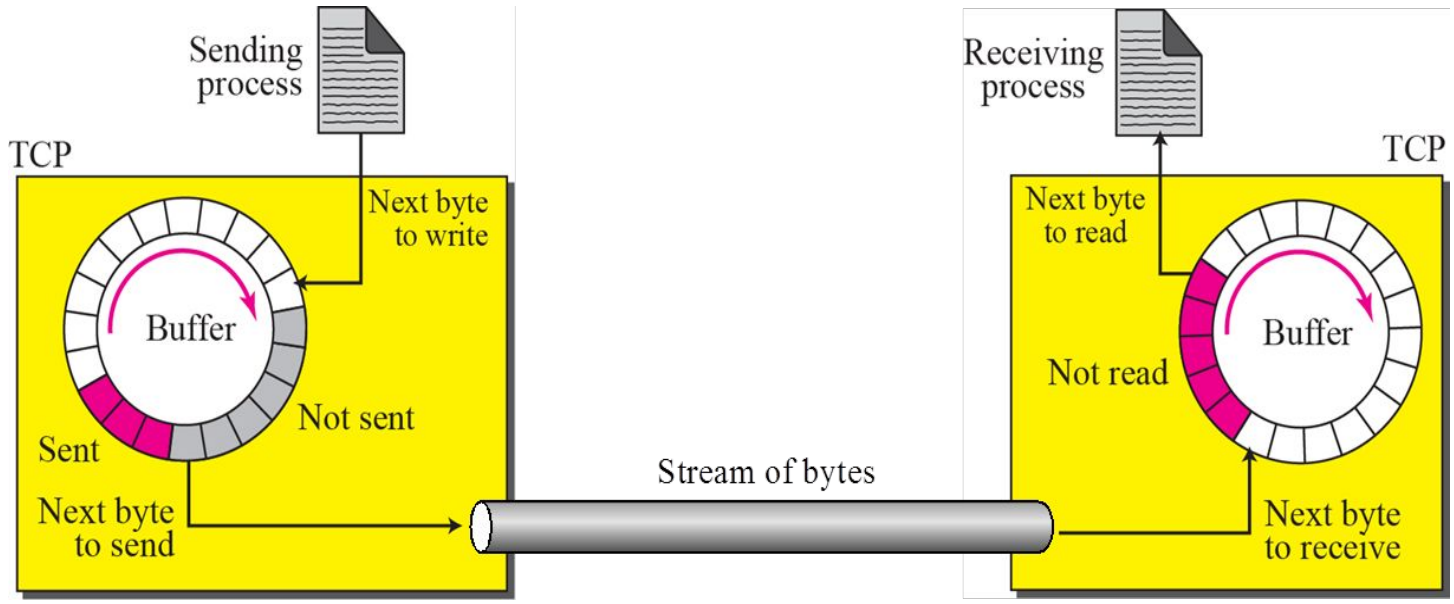
Stream Delivery (cont'd)



Sending and receiving buffers

- For stream delivery,
  - the sending and receiving TCPs use buffers
    - the sending TCP uses sending buffer to store the data coming from the sending application program.
      - the sending application program *writes* data to the buffer of the sending TCP
    - the receiving TCP receives the segments and stores them in a receiving buffer
      - the receiving application program uses the read operation to read the data from the receiving buffer.
      - Since the rate of reading can be slower than the rate of receiving, the data is kept in the buffer until the receiving application reads it completely.

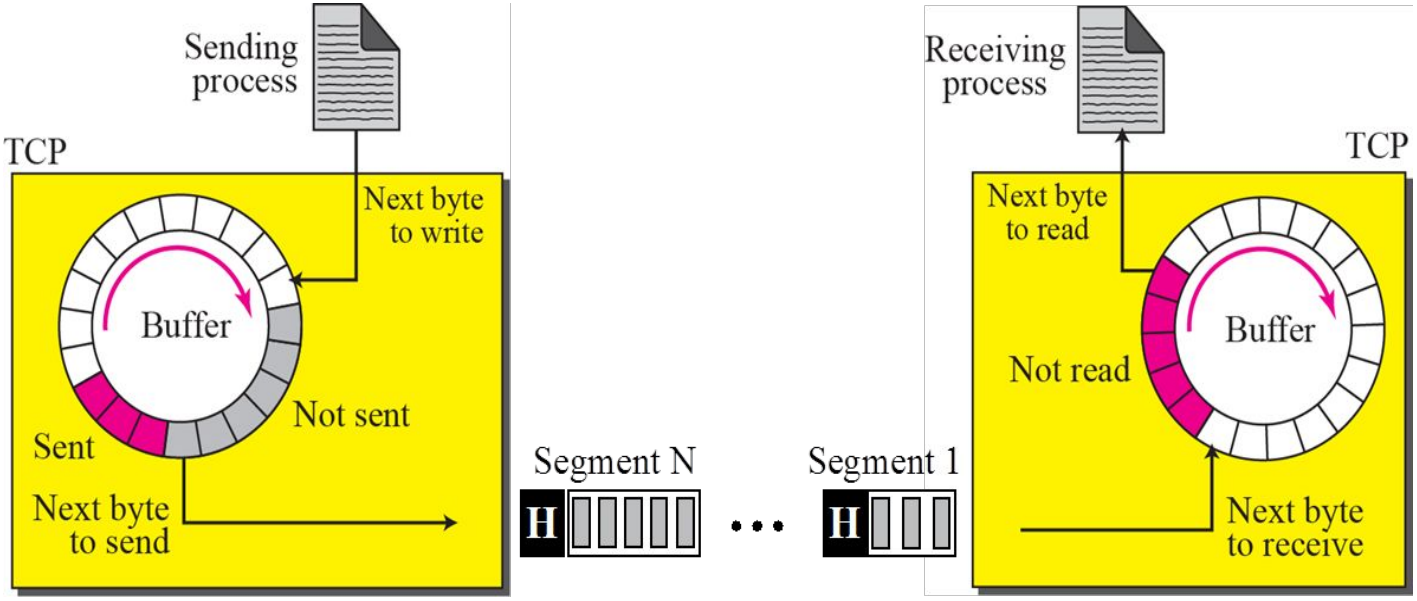
Sending and receiving buffers (cont'd)



Segments

- The IP layer, as a service provider for TCP, needs to send data in packets, not as a stream of bytes
- At the transport layer, TCP groups a number of bytes together into a packet called a segment.
  - TCP adds a header to each segment and delivers the segment to the IP layer for transmission

TCP Segments



TCP Features

- Numbering System
- Flow Control
- Error Control
- Congestion Control

Numbering System

- Byte numbers
  - All data bytes being transferred in each connection are numbered by TCP.
  - The numbering starts with a randomly generated number.
  - Number range for first byte : $0 \sim 2^{32} - 1$
    - If random number is 1,057 and total number 6,000bytes, the bytes are numbered from 1,057 to 7,056
  - Byte numbering is used for flow and error control.

Numbering System (cont'd)

- Sequence number
  - After the bytes have been numbered, TCP assigns a sequence number to each segment that is being sent.
  - Sequence number for each segment is number of the first byte carried in that segment.

Example

Suppose a TCP connection is transferring a file of 5,000 bytes. The first byte is numbered 10,001. What are the sequence numbers for each segment if data are sent in five segments, each carrying 1,000 bytes?

Solution

The following shows the sequence number for

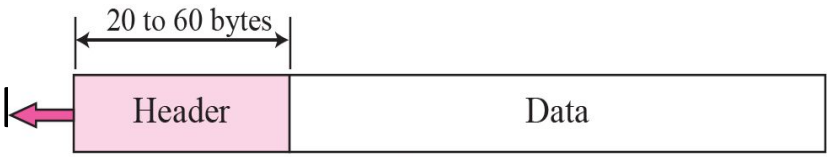
Segment 1	→	**Sequence Number:**	**10,001**	**Range:**	**10,001**	to	**11,000**
Segment 2	→	**Sequence Number:**	**11,001**	**Range:**	**11,001**	to	**12,000**
Segment 3	→	**Sequence Number:**	**12,001**	**Range:**	**12,001**	to	**13,000**
Segment 4	→	**Sequence Number:**	**13,001**	**Range:**	**13,001**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

Numbering System (cont'd)

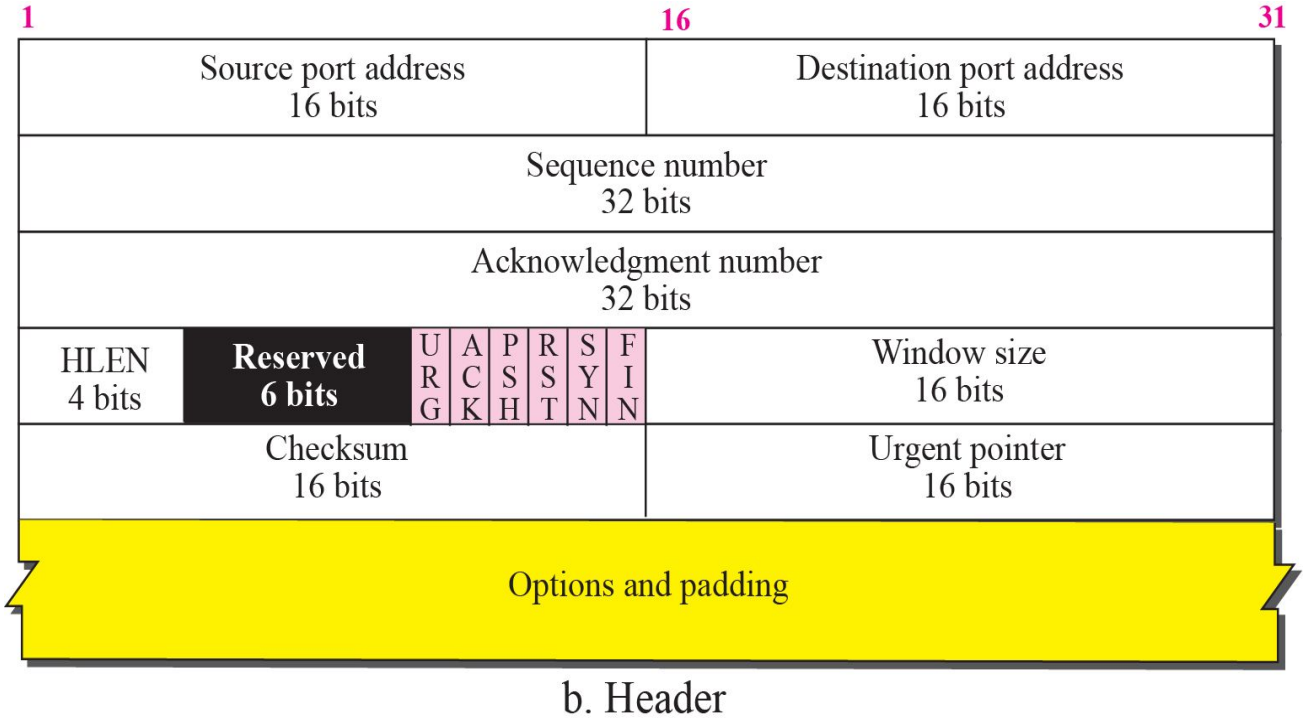
- Acknowledgment Number
  - The value of the acknowledgment field in a segment defines the number of the next byte a party expects to receives. The acknowledgment number is cumulative.

Segment

- TCP Segment



a. Segment



b. Header

Segment (cont'd)

- Source port address
  - defining the port number of application program in the host that is sending the segment
- Destination port address
  - defining the port number of application program in the host that is receiving the segment
- Sequence number
  - defining the number assigned to the first byte of data contained in this segment
  - during the connection establishment, each party uses a random number generator to create an *initial sequence number (ISN)*

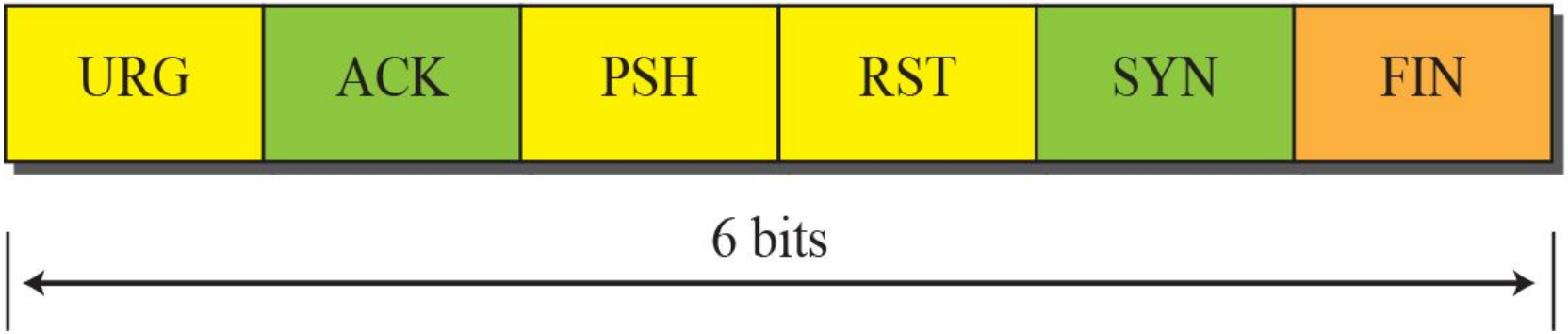
Segment (cont'd)

- Acknowledgment number
  - If the source of the segment has successfully received byte number x from the other party, it defines x+1 as the acknowledgment number
- Header length
  - Indicating the number of 4-byte words in the TCP header
    - the value between 5 and 15 (20 and 60 bytes)
- Reserved
  - For future use

Segment (cont'd)

- Control
  - Enabling flow control, connection establishment and termination, and mode of data transfer in TCP

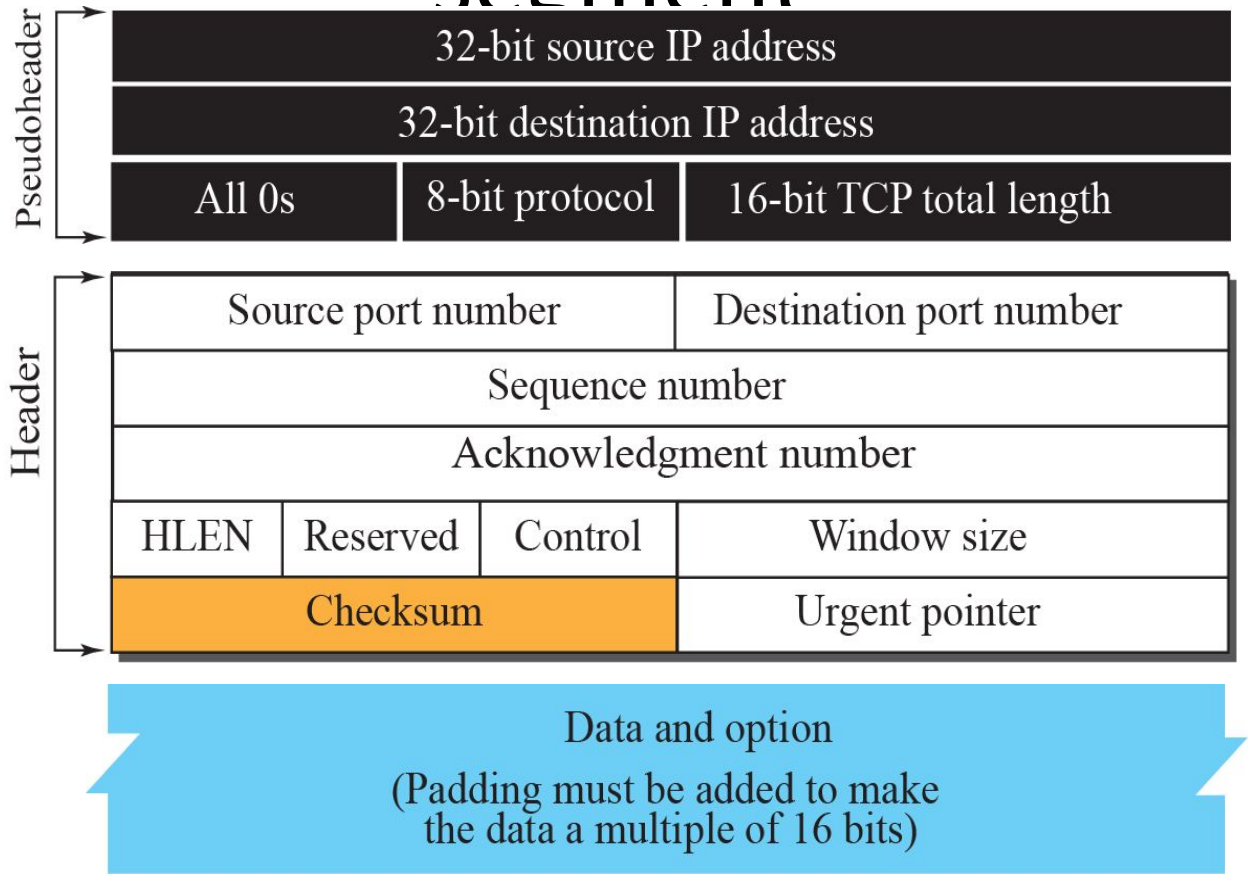
URG: Urgent pointer is valid
ACK: Acknowledgment is valid
PSH: Request for push
RST: Reset the connection
SYN: Synchronize sequence numbers
FIN: Terminate the connection



Segment (cont'd)

- Window size
  - defining the size of the window, in bytes, that the other party must maintain.
  - maximum size of window : 65,535 bytes
- Checksum : picture in next page
- Urgent pointer
  - used when the segment contains urgent data
  - defining the number that must be added to the sequence number to obtain the number of the last urgent byte in the data section of the segment
- Options : 40 bytes

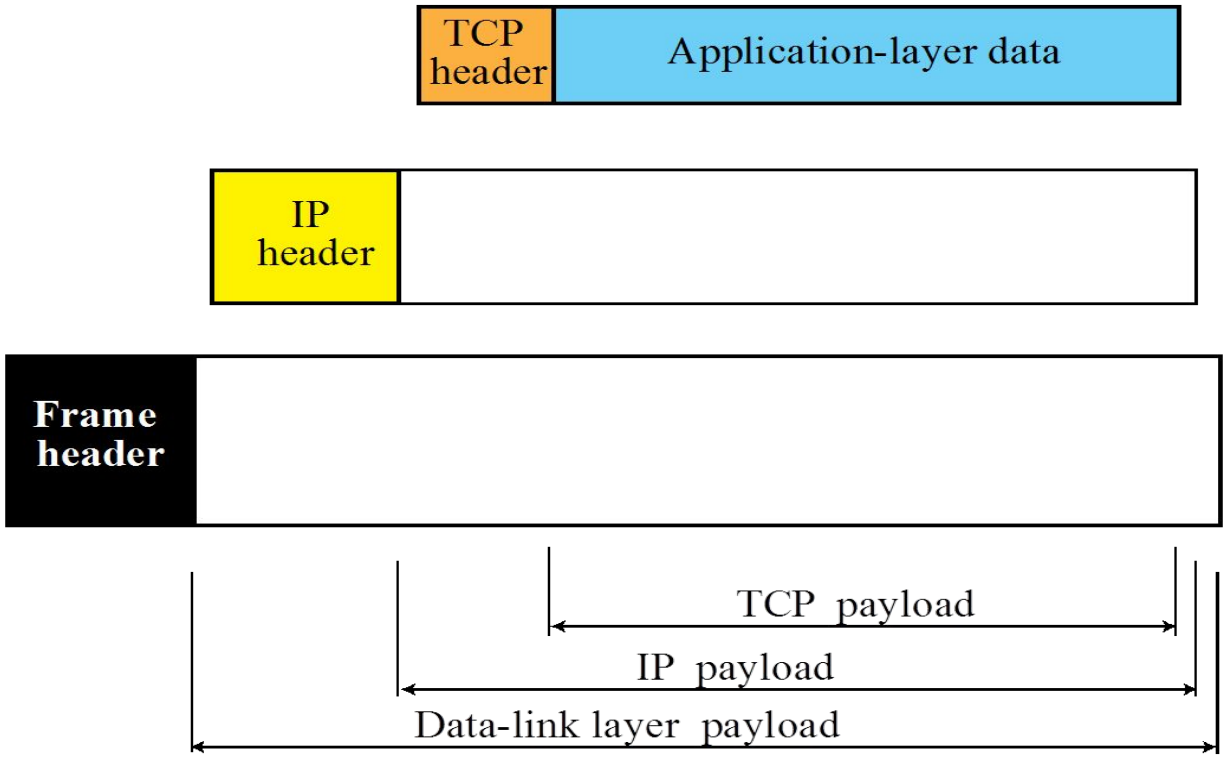
Pseudo header added to the TCP segment



- The use of the checksum in TCP is mandatory.

Encapsulation

- A TCP segments is encapsulated in an IP datagram



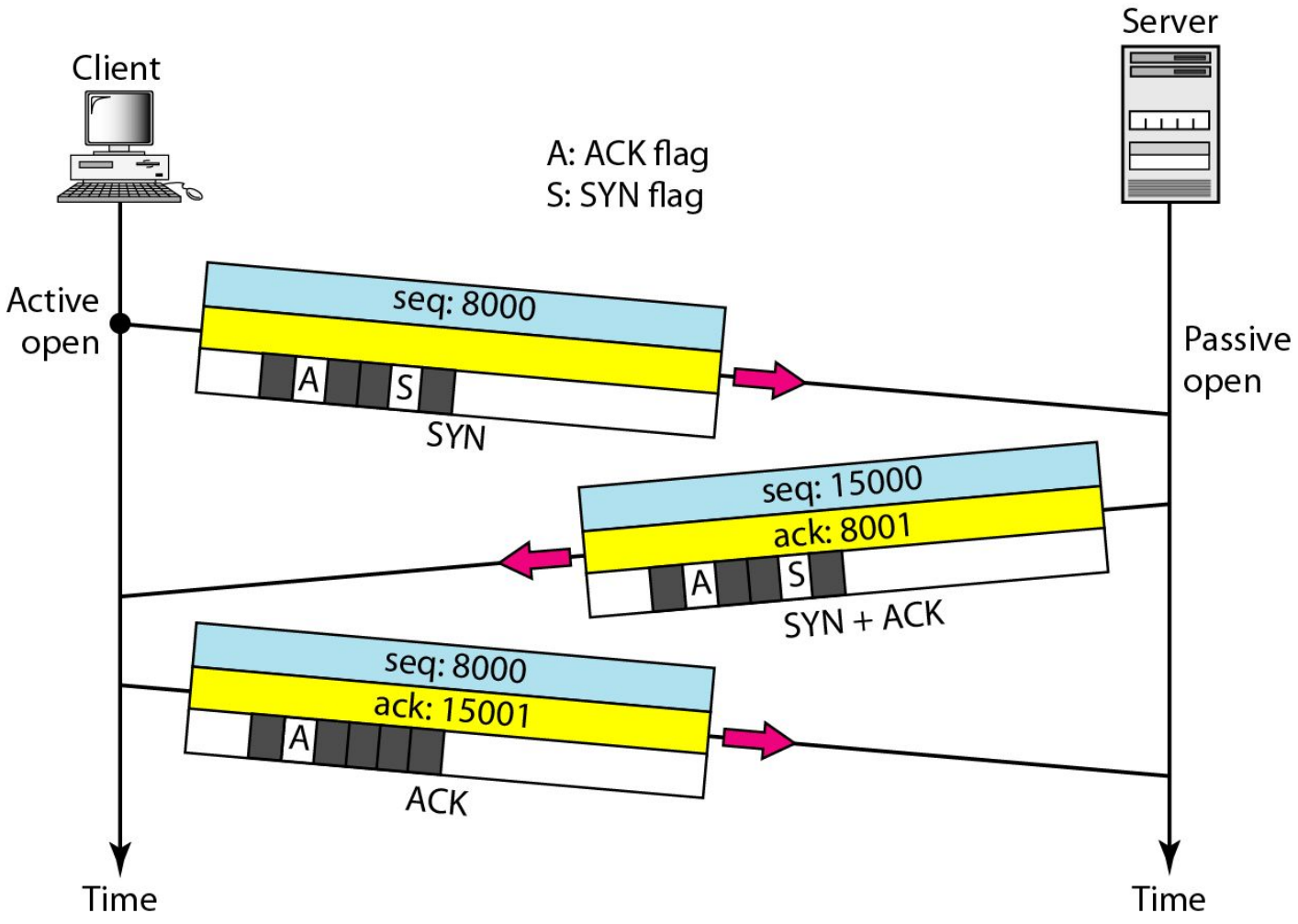
TCP Connection

- TCP is connection-oriented
  - Establishes a virtual path between the source and destination
  - TCP connection is virtual, not physical

- TCP uses the services of IP to deliver individual segments to the receiver, but it controls the connection itself

- If a segment is lost or corrupted, it is retransmitted

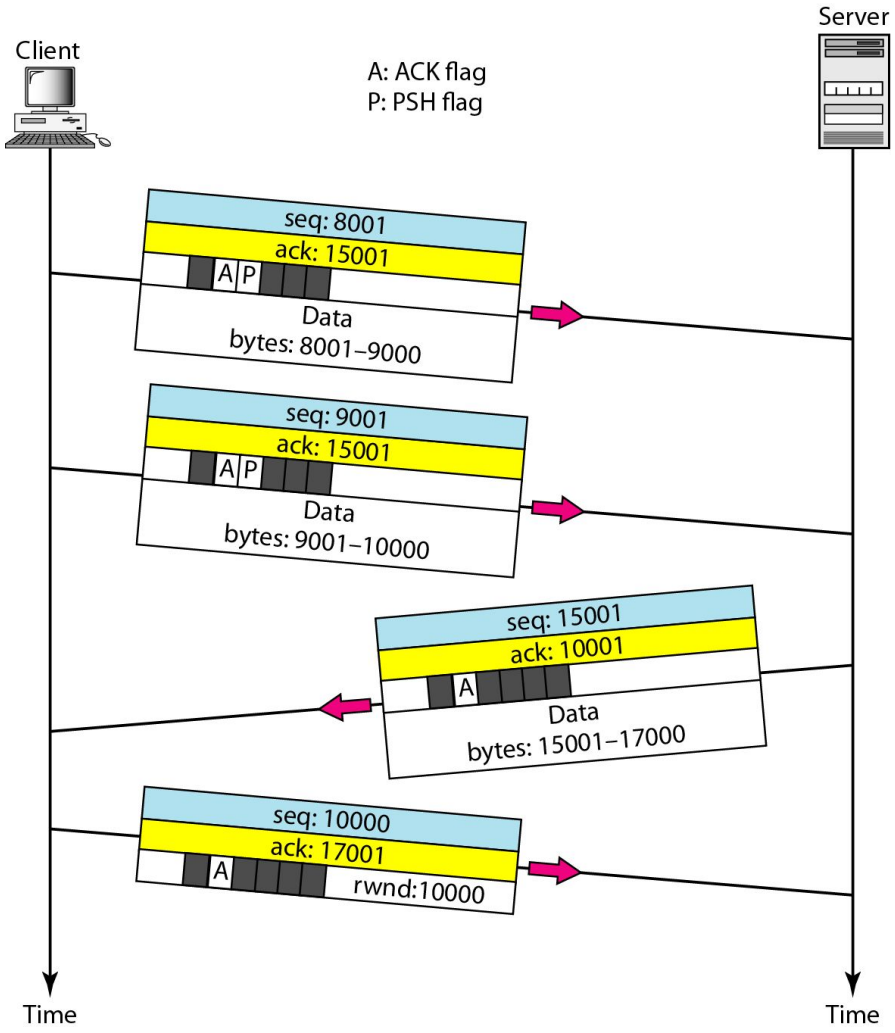
Connection establishment using three-way handshaking



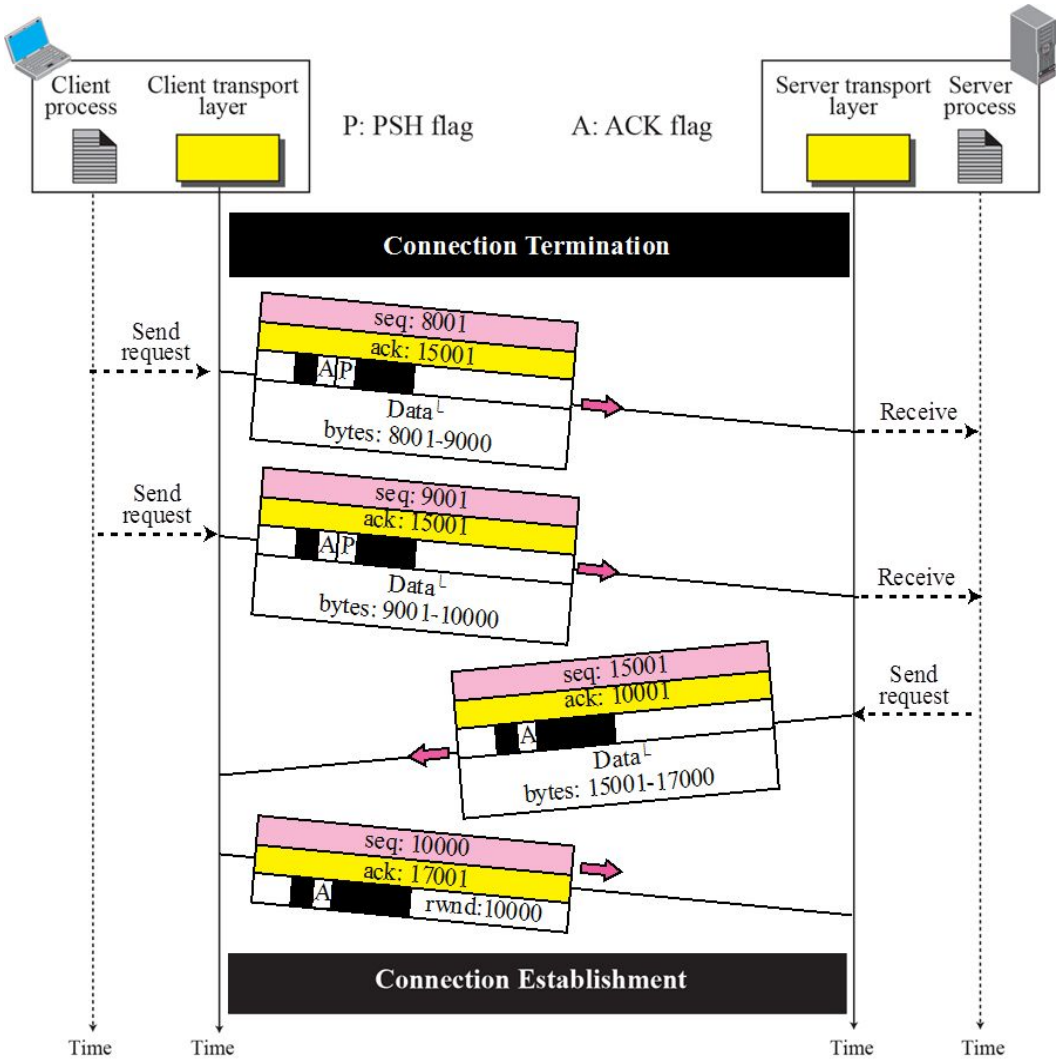
Connection Establishment using Three-way Handshake (cont'd)

- A SYN segment cannot carry data, but it consumes one sequence number.
- A SYN + ACK segment cannot carry data, but does consume one sequence number.
- An ACK segment, if carrying no data, consumes no sequence number.

Data transfer



Data Transfer



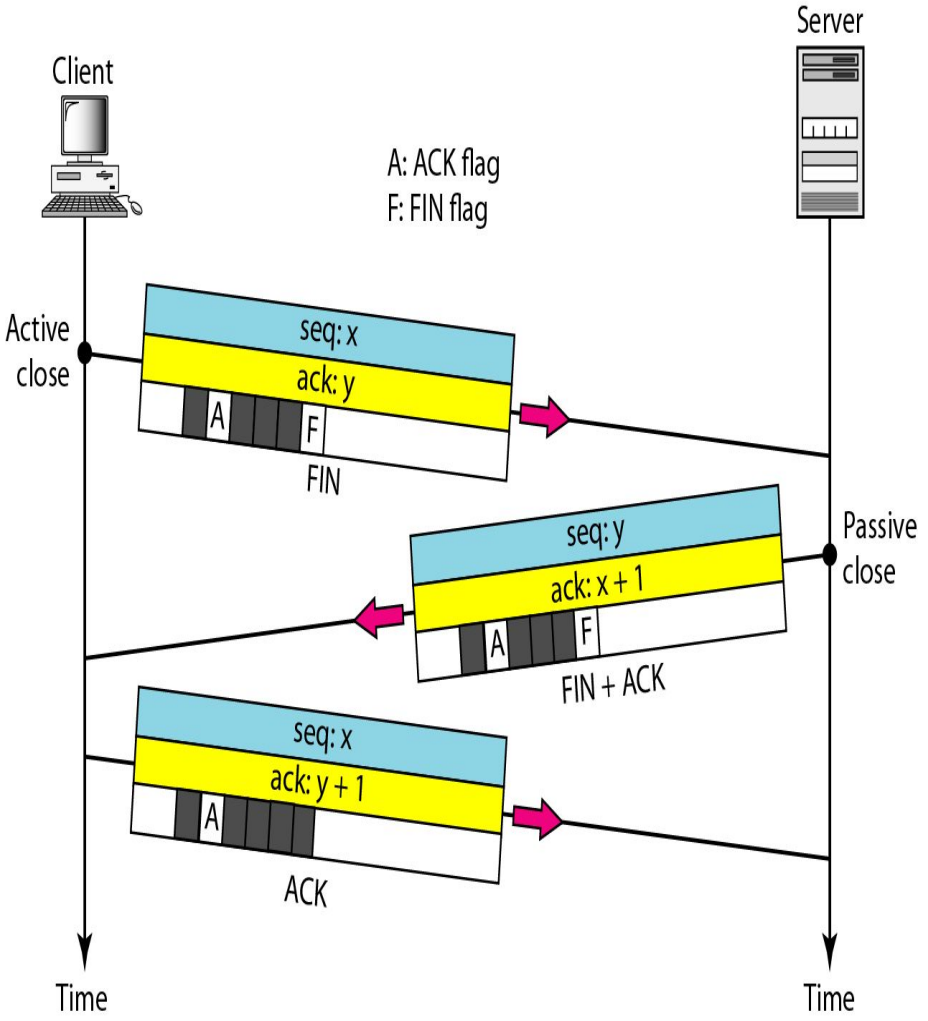
Data Transfer (cont'd)

- Urgent data
  - To send urgent data
  - Use of URG bit set by sending TCP
  - Receiving TCP extracts the urgent data from the segment using urgent pointer

Connection termination using three-way handshaking

The FIN segment consumes one sequence number if it does not carry data

The FIN + ACK segment consumes one sequence number if it does not carry data



Flow Control

- In TCP, the sender window size is totally controlled by the receiver window value. However, the actual window size can be smaller if there is congestion in the network.
- *Some Points about TCP's Sliding Windows:*
  - The size of the window is the lesser of rwnd and cwnd
  - The source does not have to send a full window's worth of data.
  - The window can be opened or closed by the receiver, but should not be shrunk.
  - The destination can send an acknowledgment at any time as long as it does not result in a shrinking window.
  - The receiver can temporarily shut down the window; the sender, however, can always send a segment of one byte after the window is shut down.
    - To prevent deadlock by proving

SLIDING Windows in TCP

- TCP uses two Windows
  - Send window and receive window
- This means four windows for a bidirectional communication
  - To make simple, we make an assumption that communication is only unidirectional
  - The bidirectional communication can be inferred using two unidirectional communications with piggybacking

SLIDING Windows in TCP

- 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.

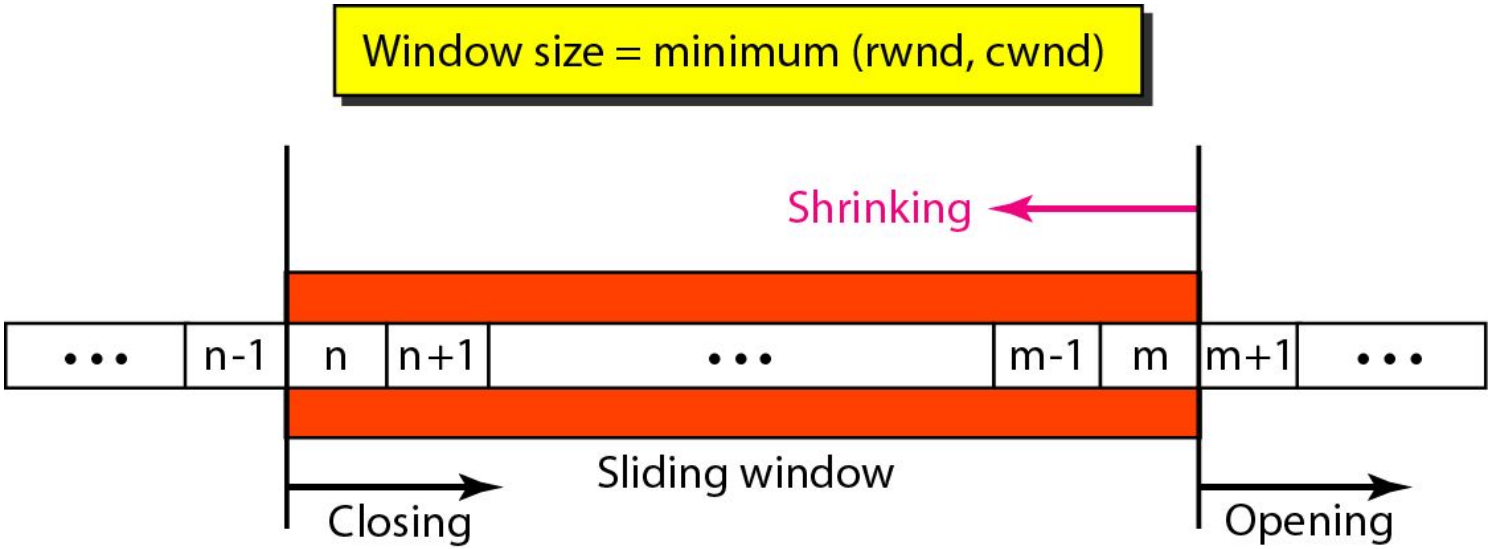
TCP sliding windows are byte-oriented.

SLIDING Windows in TCP

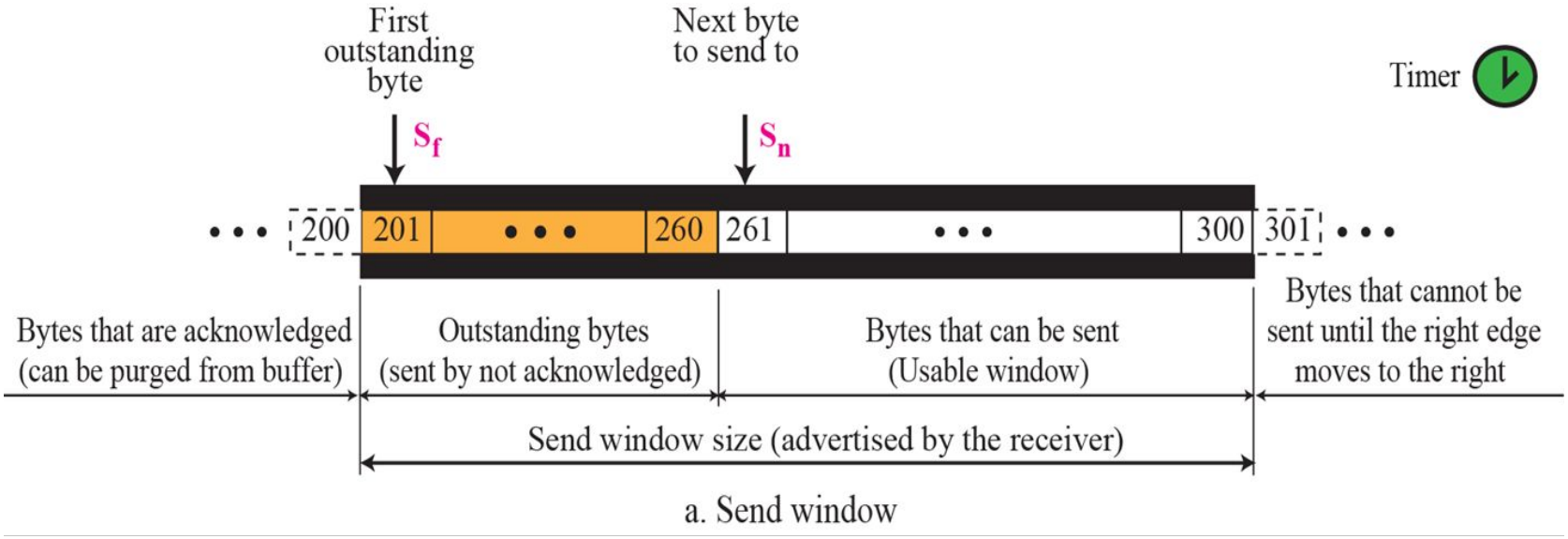
TCP sliding windows:

- ❑ The size of the window is the lesser of rwnd and cwnd.
- ❑ The source does not have to send a full window's worth of data.
- ❑ The window can be opened or closed by the receiver, but should not be shrunk.
- ❑ The destination can send an acknowledgment at any time as long as it does not result in a shrinking window.
- ❑ The receiver can temporarily shut down the window; the sender, however, can always send a segment of 1 byte after the window is shut down.

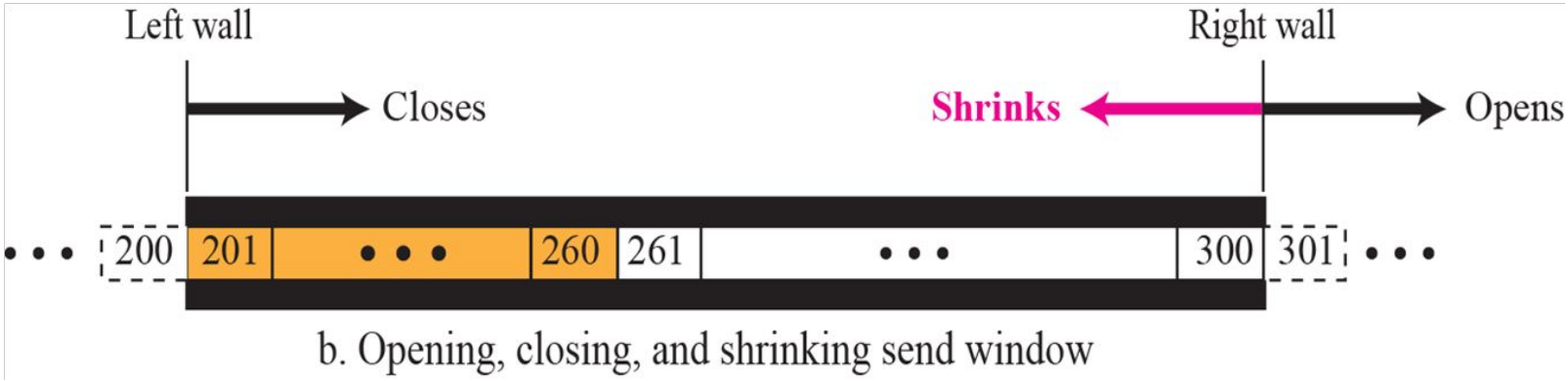
SLIDING WINDOW



Send Window in TCP

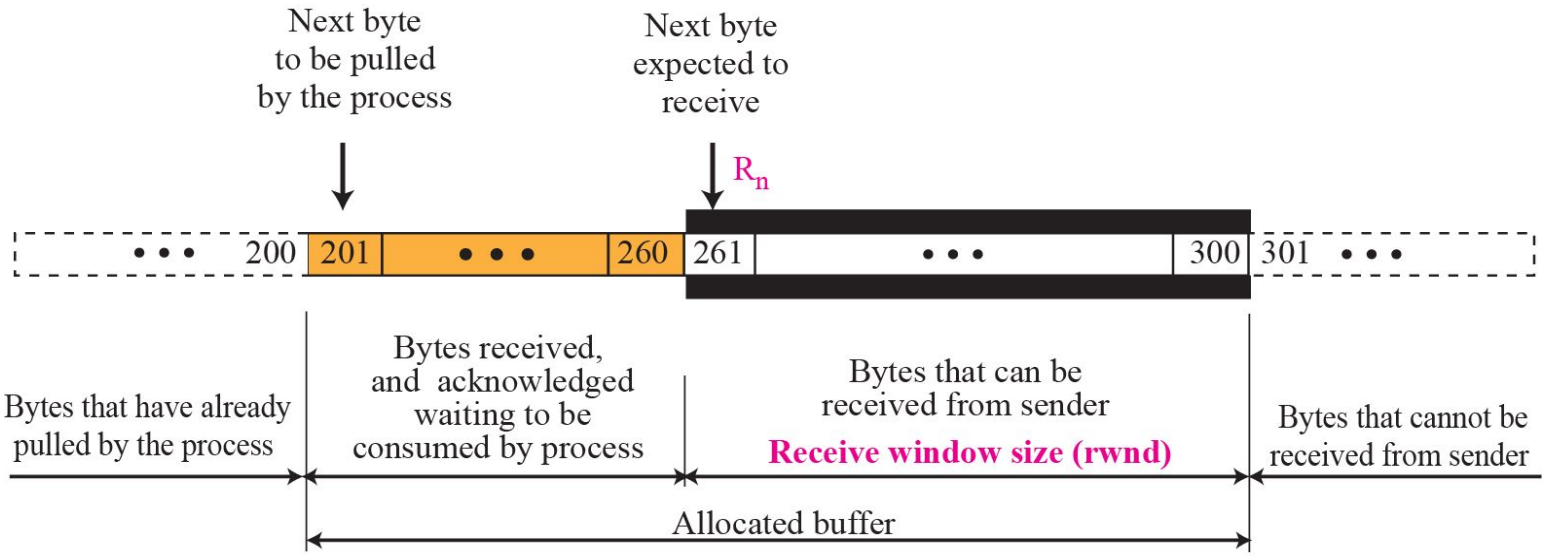


a. Send window

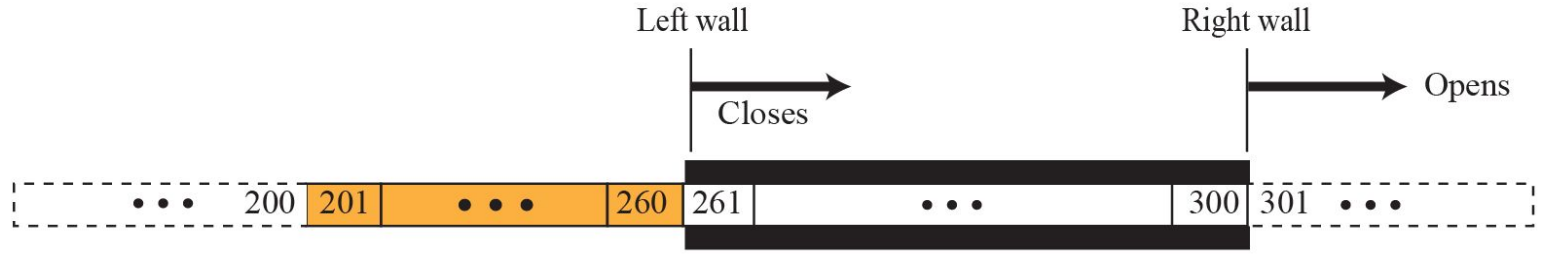


b. Opening, closing, and shrinking send window

Receive Window in TCP



a. Receive window and allocated buffer



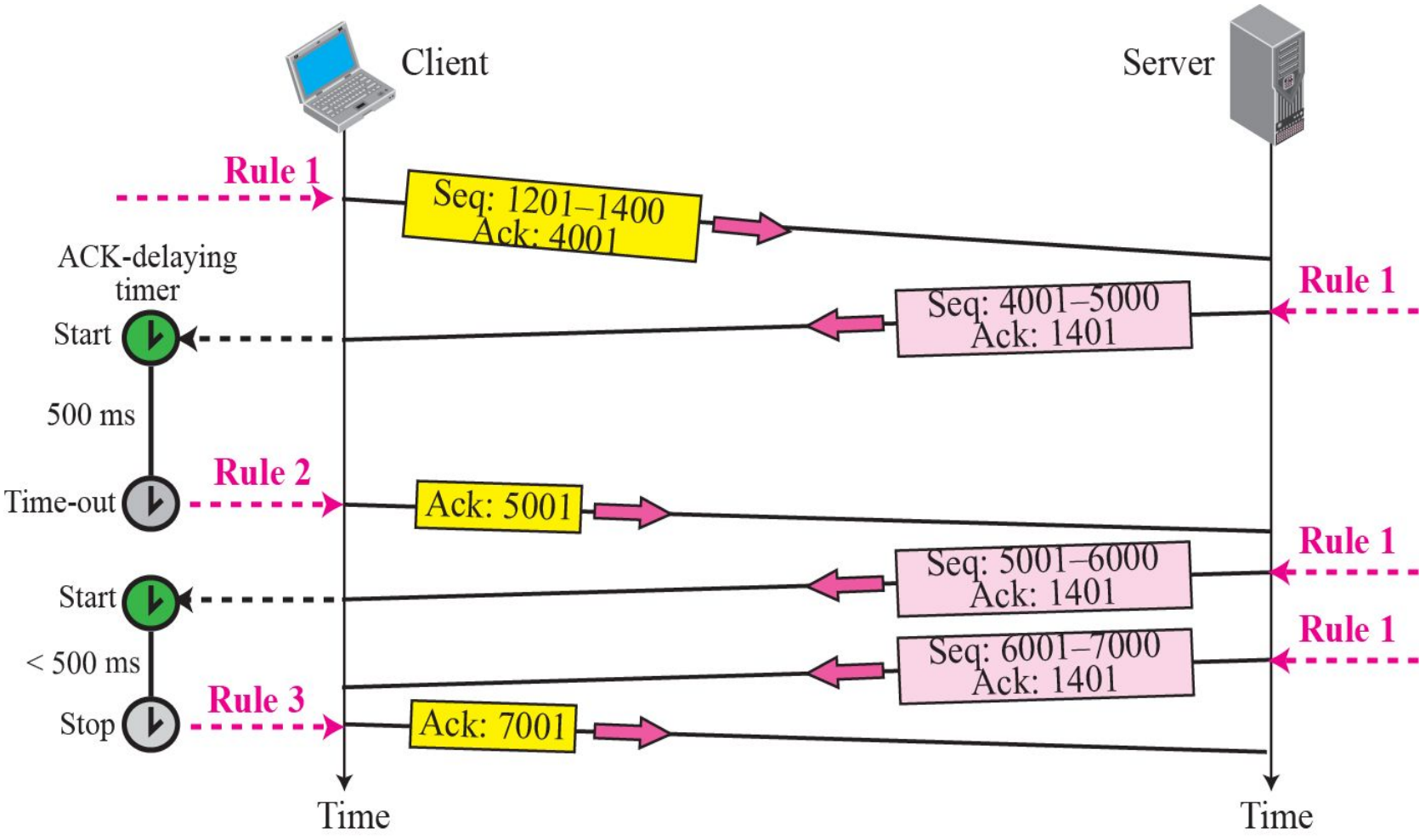
b. Opening and closing of receive window

Error Control

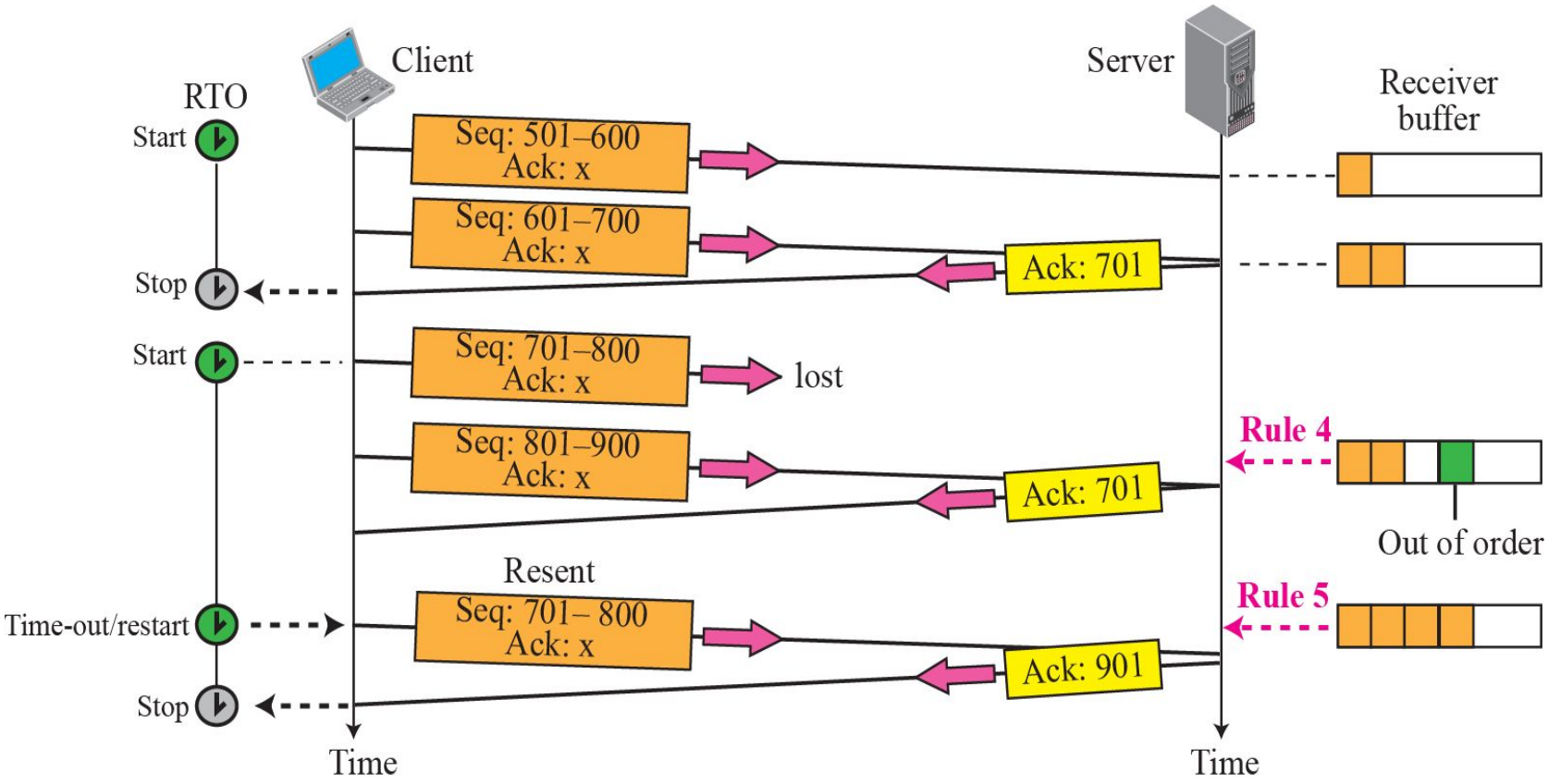
- TCP is a reliable transport layer protocol
  - Application program that delivers a stream of data to TCP relies on TCP to deliver the entire stream to the application program on the other end *in order*, *without error, and without any part lost or duplicated*.

- Error control in TCP is achieved through the use of three tools
  - Checksum
  - Acknowledgment
  - Time-Out

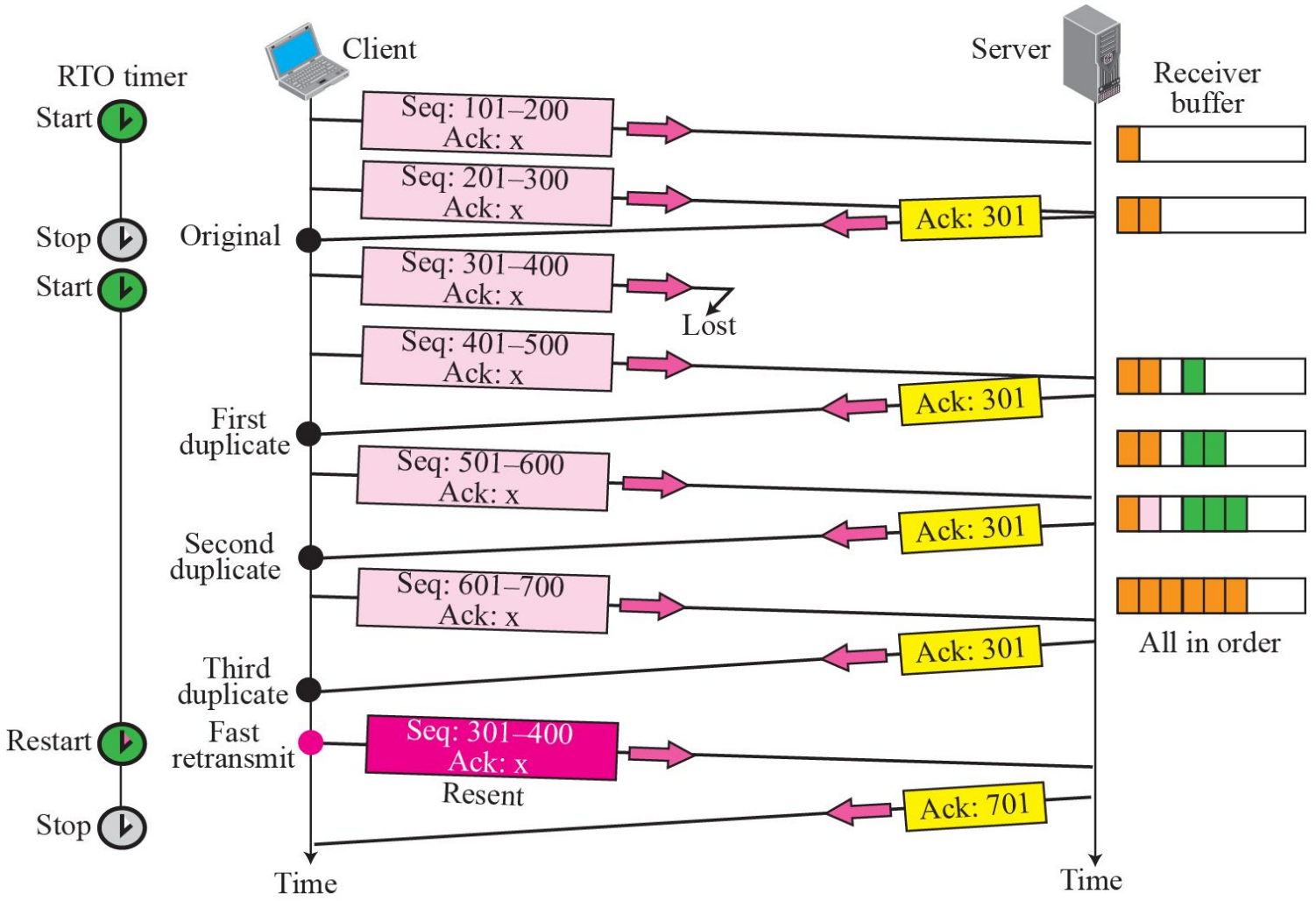
Normal Operation



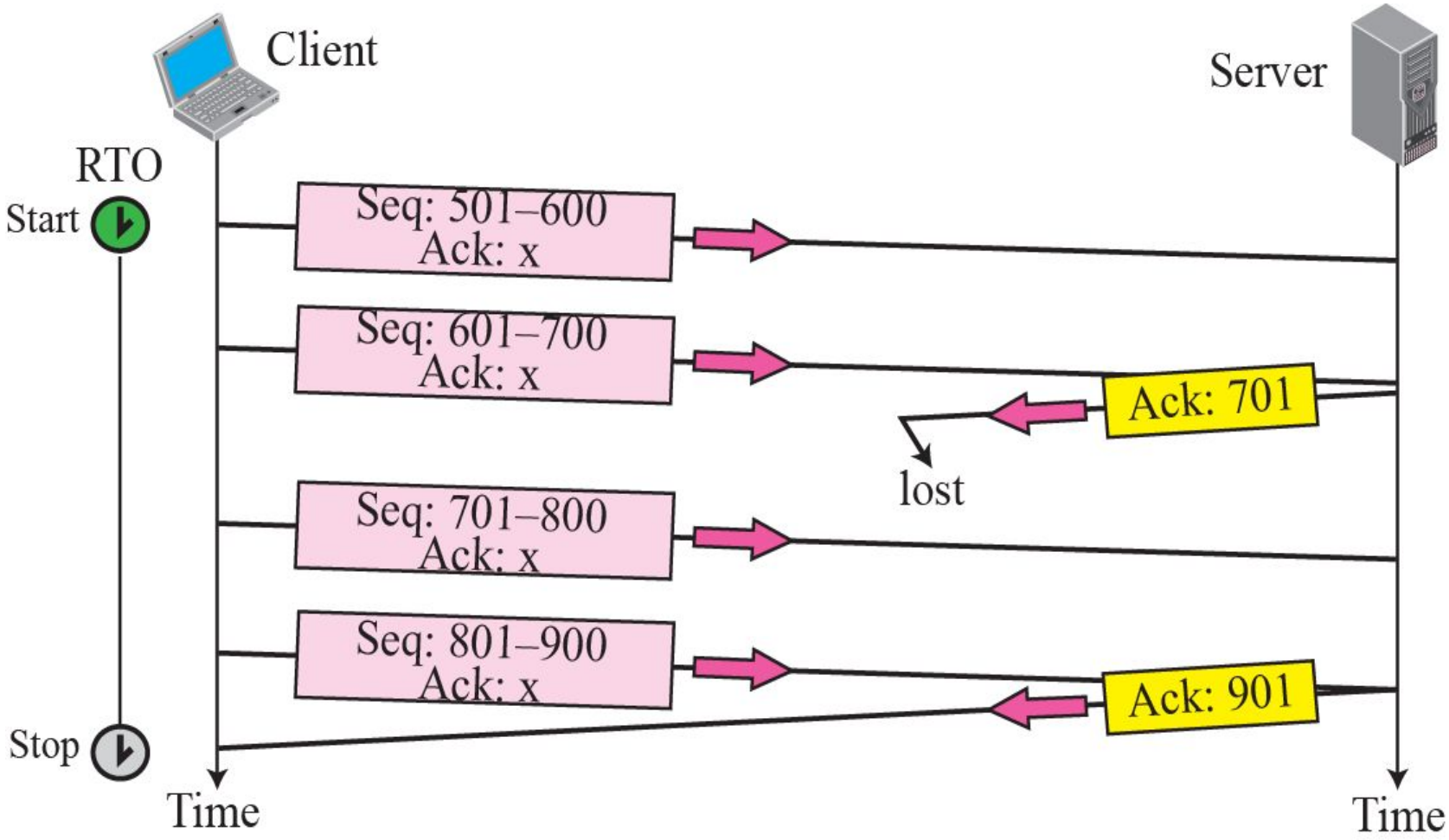
Lost Segment



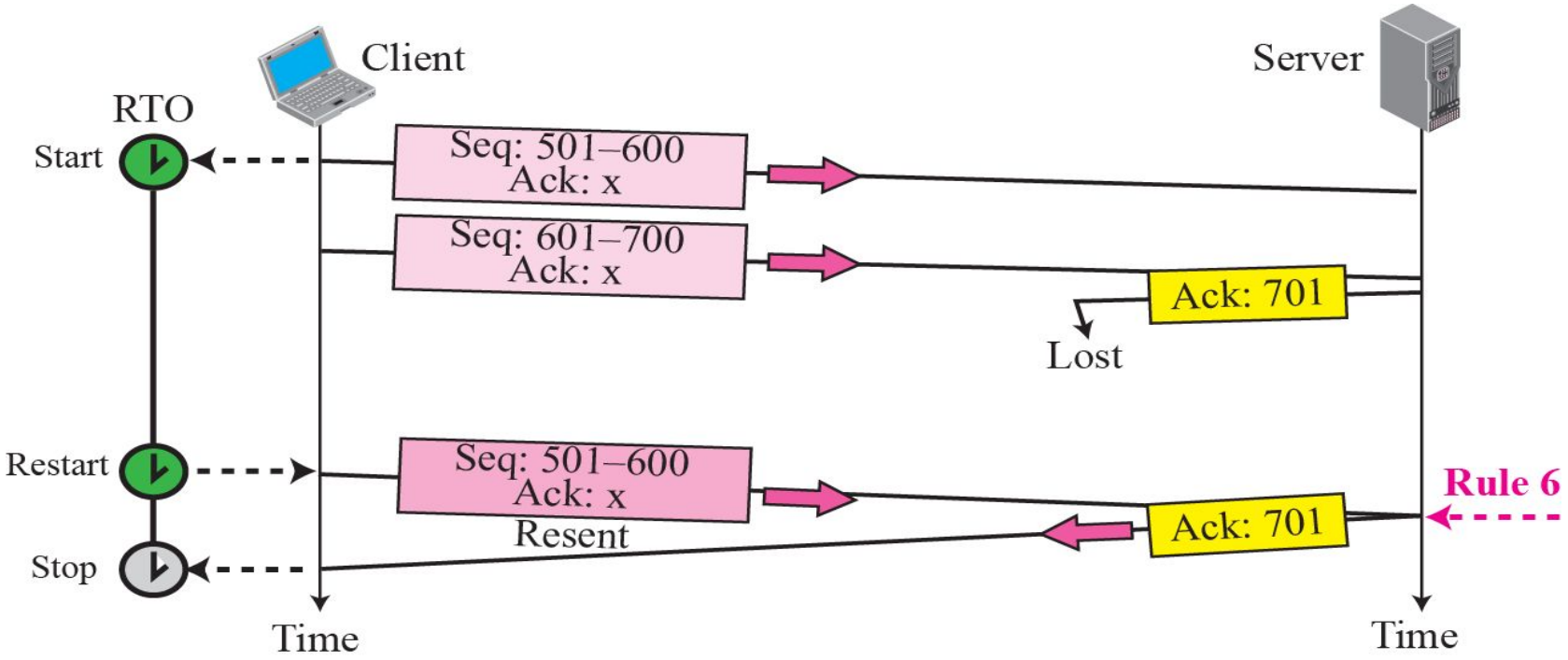
Fast Retransmission



Lost Acknowledgment



Lost Acknowledgment Corrected by Resending a Segment



- Lost acknowledgments may create deadlock if they are not properly handled.

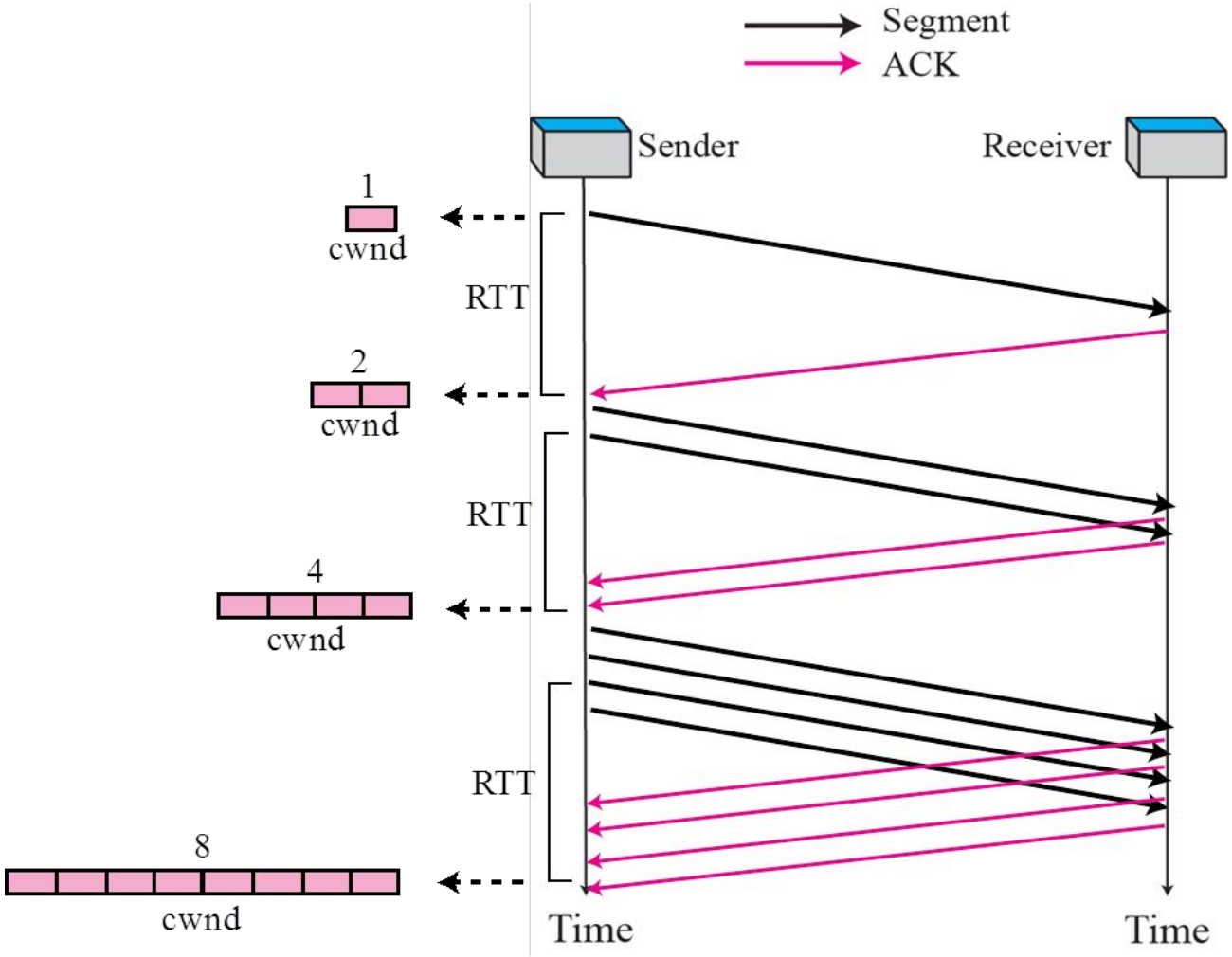
Congestion Control

- Congestion in a network may occur if the load on the network is greater than the capacity of the network

- Congestion control refers to the mechanism and techniques to control the congestion and keep the load below the capacity

- Congestion in a network or internetwork occurs because routers and switches have queues.
- Congestion window
  - Today, TCP protocols include that the sender's window size is not only determined by the receiver but also by congestion in the network
  - Actual window size = minimum (rwnd, cwnd)

Slow Start, Exponential Increase



Congestion Control (cont'd)

- In the slow start algorithm, the size of the congestion window increases exponentially until it reaches a threshold.

Start □ cwnd = 1

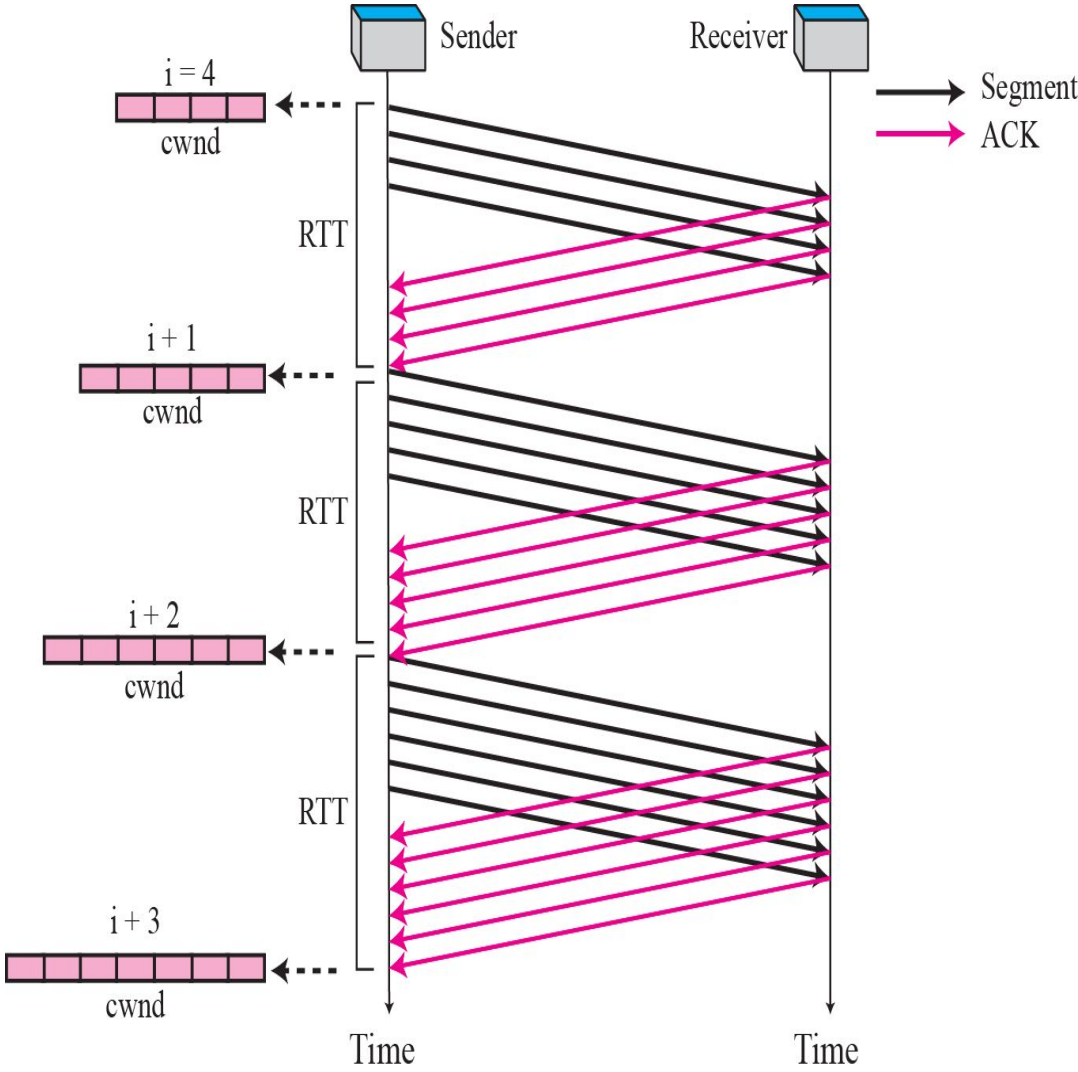
After 1 RTT □ cwnd = 1 x 2 = 2 □ 2^1

After 2 RTT □ cwnd = 2 x 2 = 4 □ 2^2

After 3 RTT □ cwnd = 4 x 2 = 8 □ 2^3

Congestion Avoidance, Additive Increase

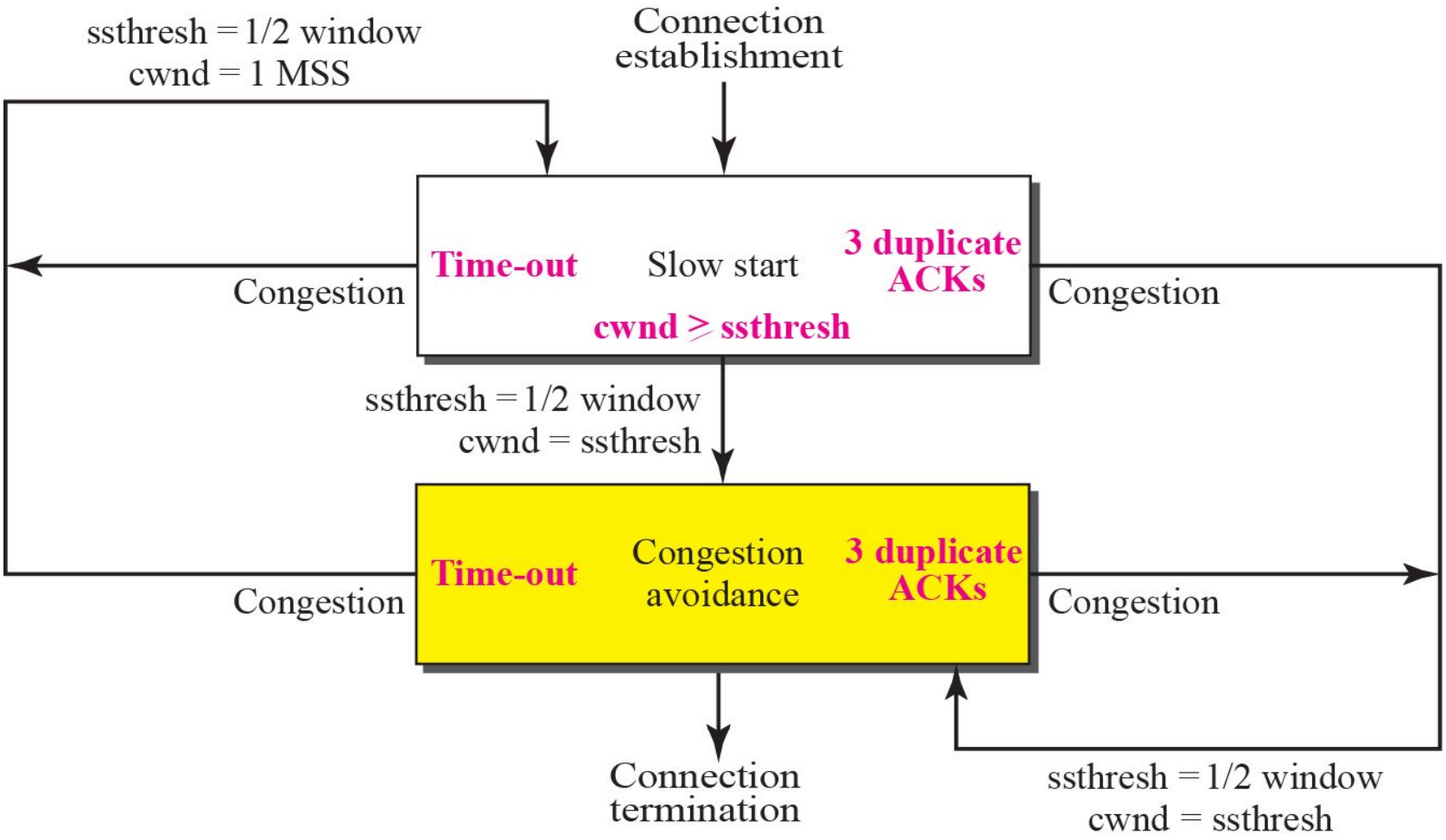
- When the size of the congestion window reaches the slow start threshold, in the congestion avoidance algorithm, the size of the congestion window increases additively until congestion is detected



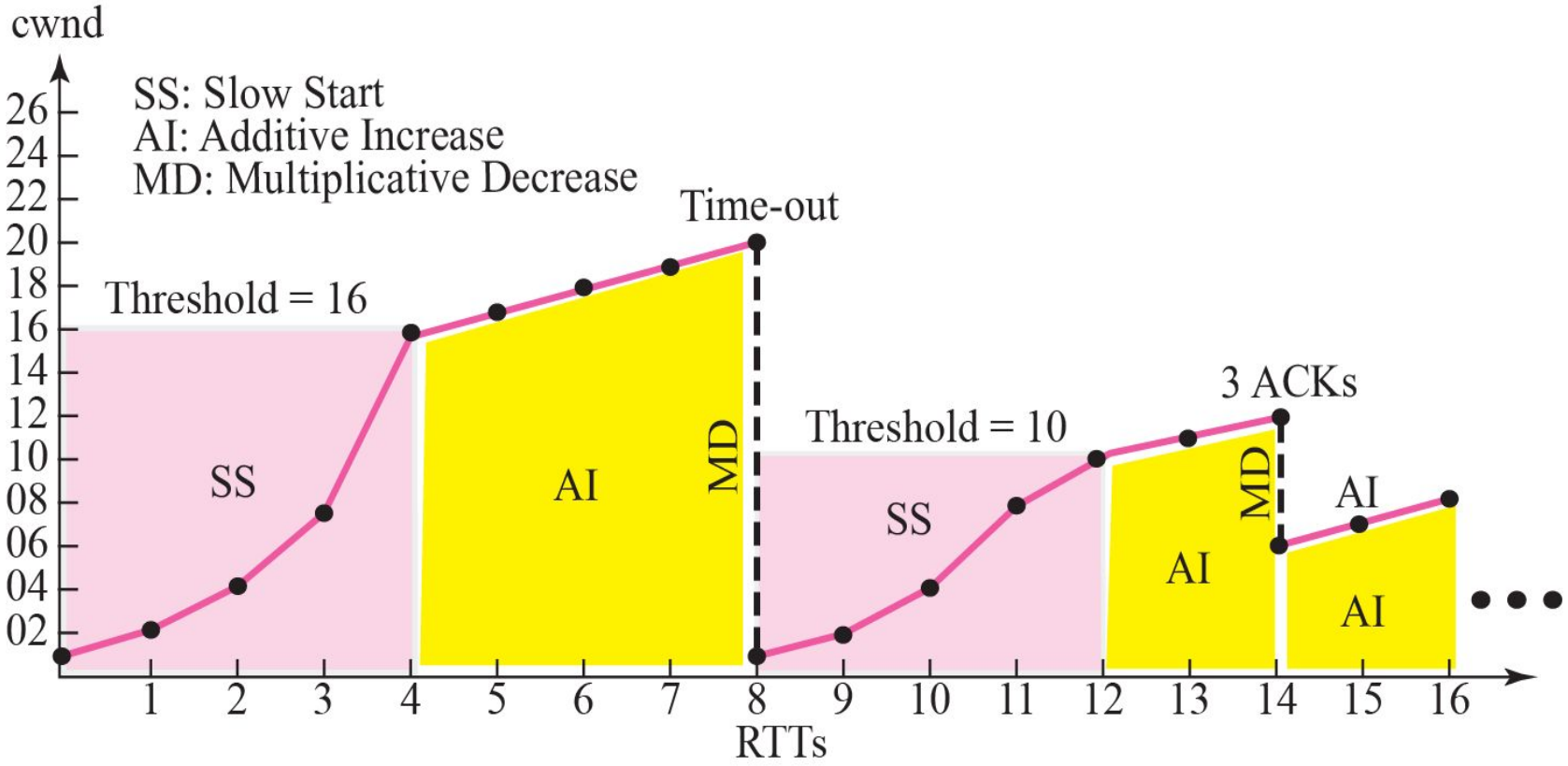
Congestion Detection : Multiplicative Decrease

- Most implementations react differently to congestion detection:
  - If detection is by time-out, a new slow start phase starts.
  - If detection is by three ACKs, a new congestion avoidance phase starts.

TCP Congestion Policy Summary



Congestion Example



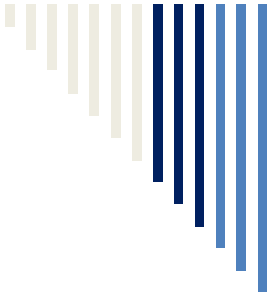
Summary

- Transmission Control Protocol (TCP) is one of the transport layer protocols in the TCP/IP protocol suite. TCP provides process-to-process, full-duplex, and connection-oriented service. The unit of data transfer between two devices using TCP software is called a segment; it has 20 to 60 bytes of header, followed by data from the application program.

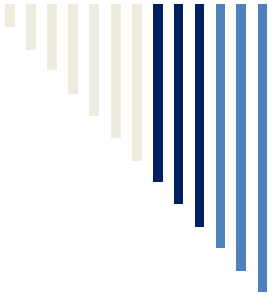
- A TCP connection consists of three phases: connection establishment, data transfer, and connection termination. Connection establishment requires three-way handshaking; connection termination requires three- or four-way handshaking. TCP software is normally implemented as a finite state machine. (FSM)

Summary

- TCP uses flow control, implement as a sliding window mechanism, to avoid overwhelming a receiver with data. The TCP window size is determined by the receiver-advertised window size (rwnd) or the congestion window size (cwnd), whichever is smaller. The window can be opened or closed by the receiver, but should not be shrunk. The bytes of data being transferred in each connection are numbered by TCP. The numbering starts with a randomly generated number.
- TCP uses error control to provide a reliable service. Error control is handled by the checksum, acknowledgement, and time-out. Corrupted and lost segments are retransmitted and duplicate segments are discarded. Data may arrive out of order and temporarily stored by the receiving TCP, but TCP guarantees that no out-of-order segment is delivered to the process. In modern implementations, a retransmission occurs if the retransmission timer expires or three duplicate ACK segments have arrived.



Transport layer protocols



UDP

USER DATAGRAM PROTOCOL (UDP)

The User Datagram Protocol (UDP) is called a connectionless, unreliable transport protocol. It does not add anything to the services of IP except to provide process-to-process communication instead of host-to-host communication.

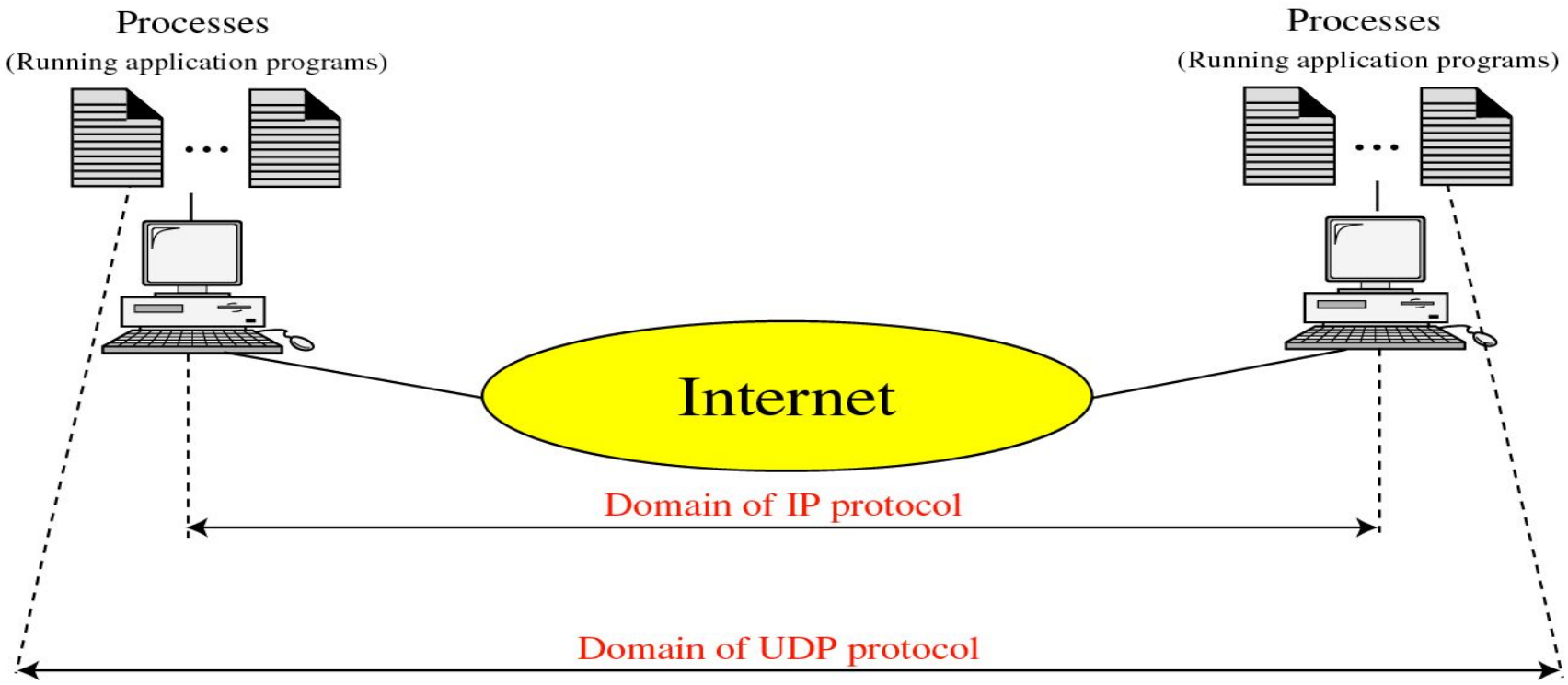
Topics discussed in this section:

Well-Known Ports for UDP
User Datagram
Checksum
UDP Operation
Use of UDP

Process-to-process communication (cont'd)

- IP is responsible for communication at the computer level (host-to-host communication)

- UDP is responsible for delivery of the message to the appropriate process

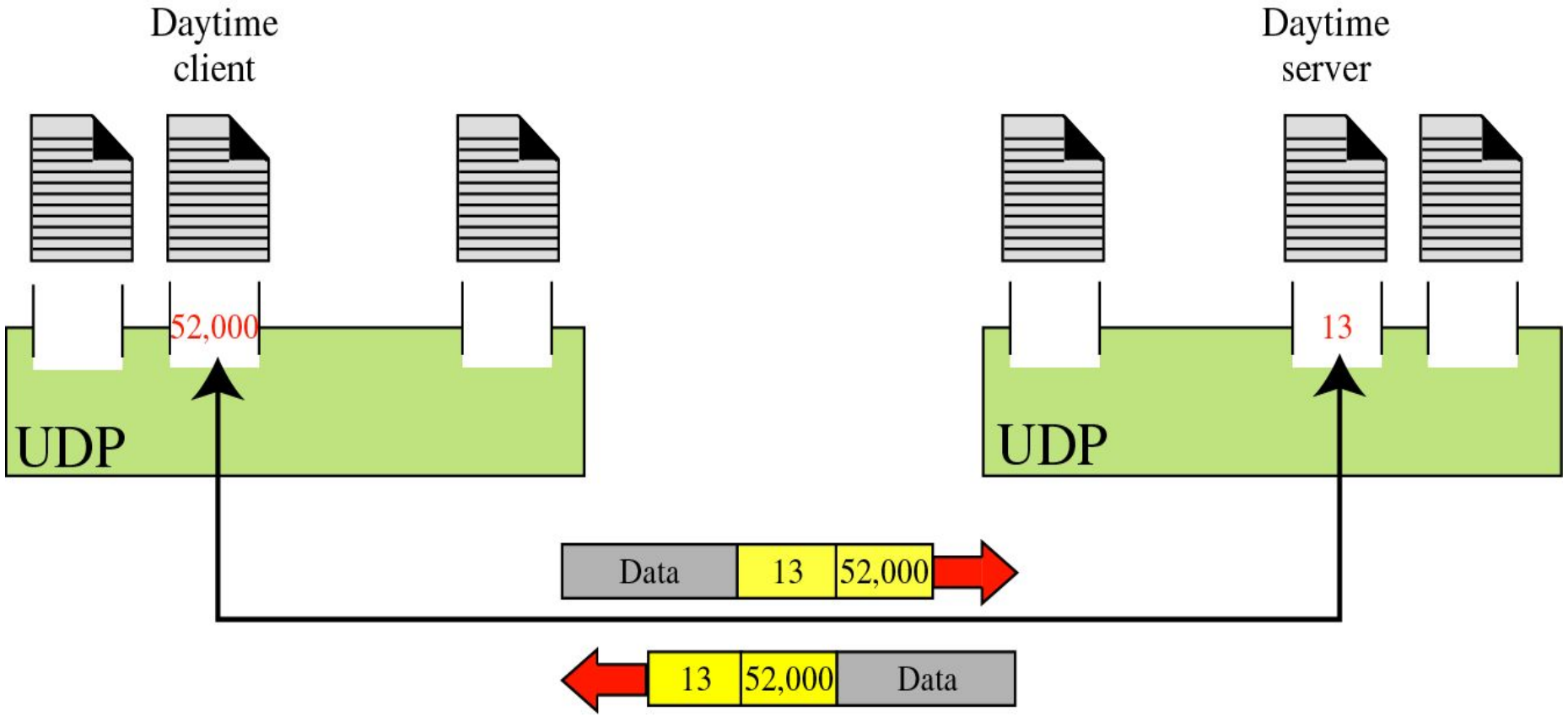


Process-to-process communication (cont'd)

- Port Numbers
  - used in client-server paradigm
  - used for defining processes
  - integers between 0 and 65,535
  - The client program defines itself with a port number, chosen randomly by the UDP software running on the client host
    - the ephemeral port number
  - But, the server process must also define itself with a port number that is not randomly chosen
    - using well-known port number

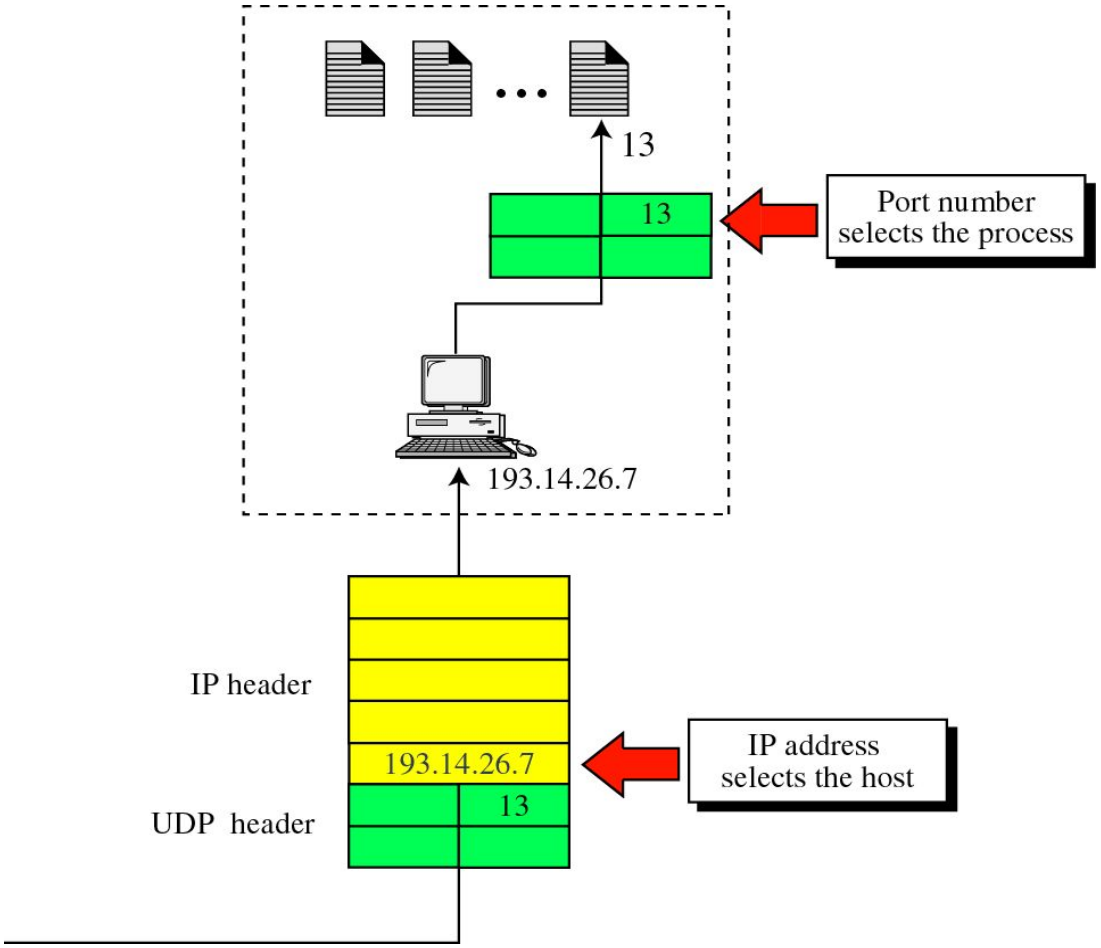
Process-to-process communication (cont'd)

- Example : Daytime process
  - for client, an ephemeral (temporary) port number



Process-to-process communication (cont'd)

- IP addresses versus port numbers

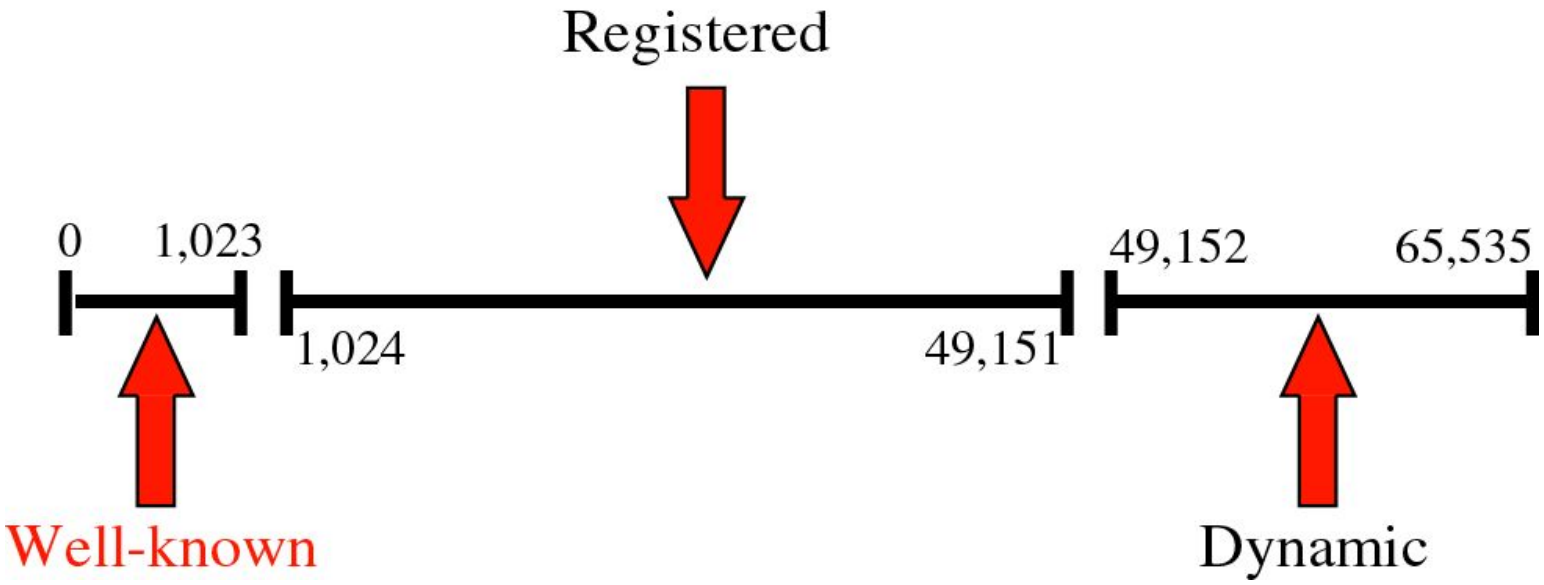


Process-to-process communication (cont'd)

- IANA (Internet Assigned Numbers Authority)
  - port numbers divided into 3 ranges
    - well-known ports : ranging from 0 to 1,023
    - registered ports : ranging from 1,024 to 49,151
      - They are not controlled by IANA. But they can only be registered with IANA to prevent duplication.
    - dynamic ports : ranging from 49,152 to 65,535
      - neither controlled nor registered
      - can be used any process.
      - are ephemeral ports
- Ranges Used by Other Systems
  - Other operating system may use ranges other than IANA's for the well-known and ephemeral ports
    - BSD Unix has 3 ranges: reserved, ephemeral, and non-privileged

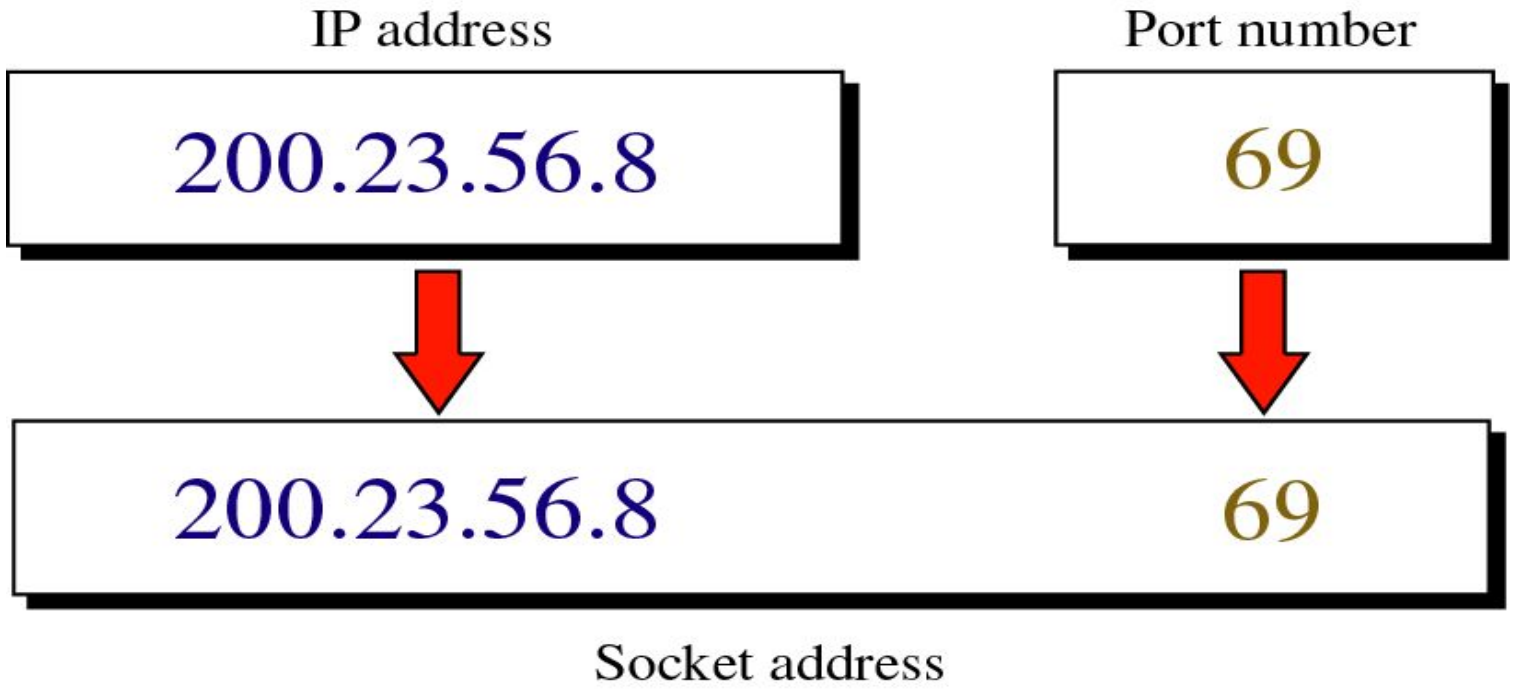
Process-to-process communication (cont'd)

- IANA ranges



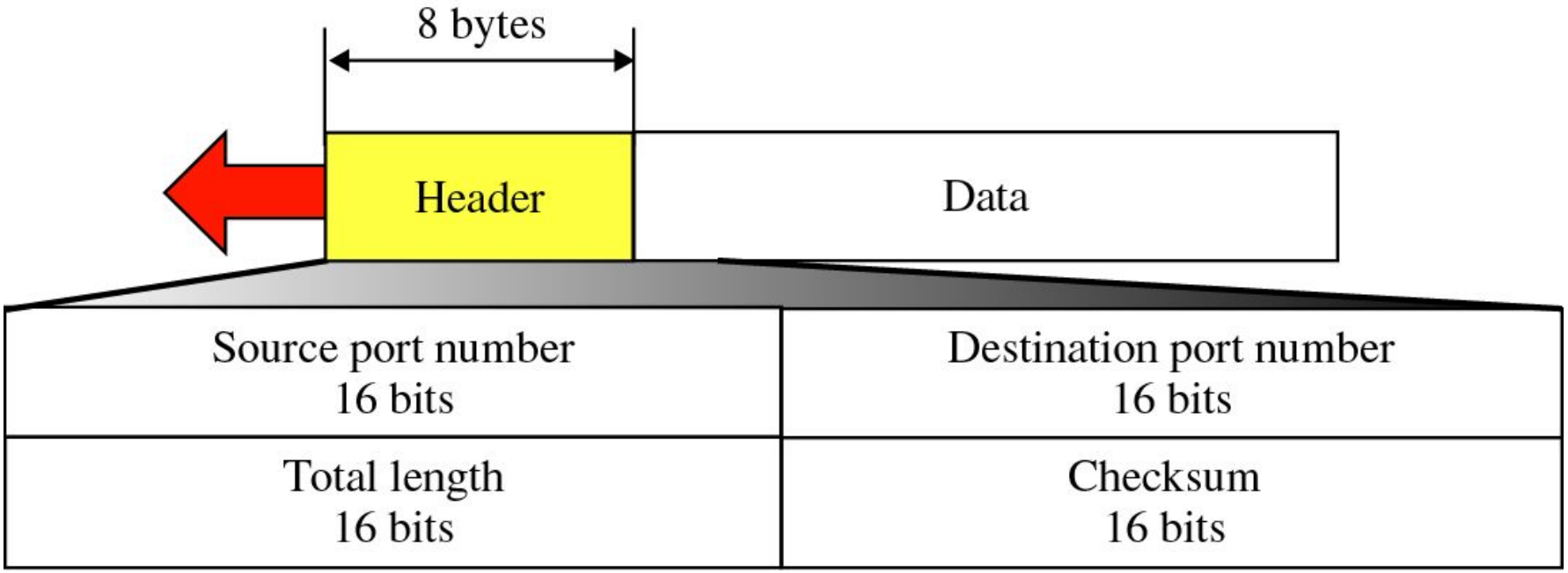
Process-to-process communication (cont'd)

- Socket Address
  - the combination of an IP address and a port number



User Datagram

- User datagram format



User Datagram

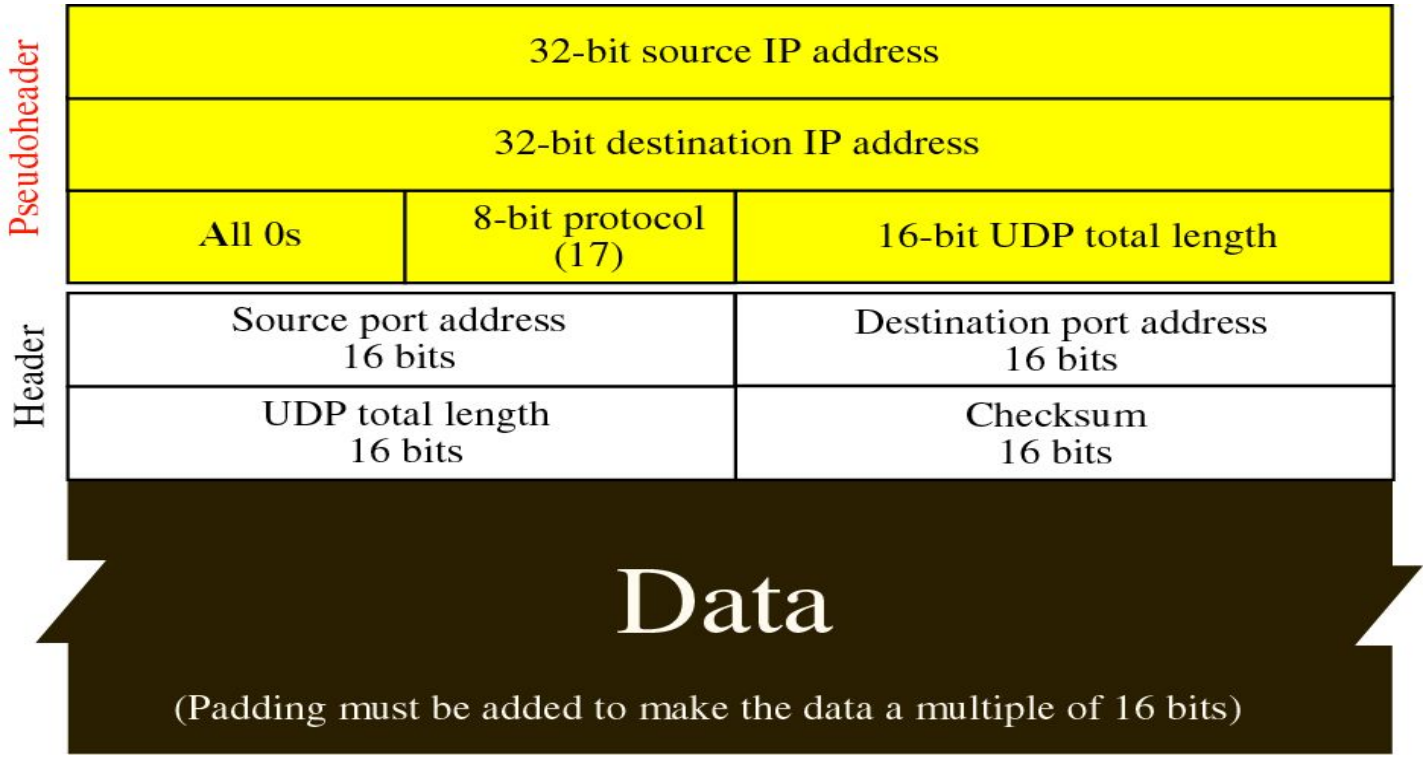
- Source port number
  - In case of the client (a client sending a request), having ephemeral port number requested by the process
  - In case of the server (a sever sending response), having a well-known port number
- Destination port number
  - Used by the process running on the destination host
- Length
  - Defining the total length of the user diagram, header + data
  - Minimum 8 bytes. (header + no data)
  - the length of data : 0 to 65,507 (65,535 – 20 – 8) bytes
  - UDP length = IP Length – IP header's length
- Checksum
  - used to detect errors over the entire user datagram

Checksum

- UDP checksum calculation is different from the checksum for IP and ICMP
- It includes as follows.
  - pseudoheader : part of the header of the IP packet
  - UDP header
  - data from the application layer

Checksum (cont'd)

- Pseudoheader added to the UDP datagram



Checksum (cont'd)

- Checksum Calculation at Sender
  1. Add the pseudoheader to the UDP datagram
  2. Fill the checksum field with zeros
  3. Divide the total number of bytes into 16-bit words
  4. If the total number of bytes is not even, add one byte of padding (all 0s)
  5. Add all 16-bit sections using one's complement arithmetic.
  6. Complement the result, and insert it in the checksum field
  7. Drop the pseudoheader and added padding
  8. Deliver the UDP user datagram to the IP software for encapsulation

Checksum (cont'd)

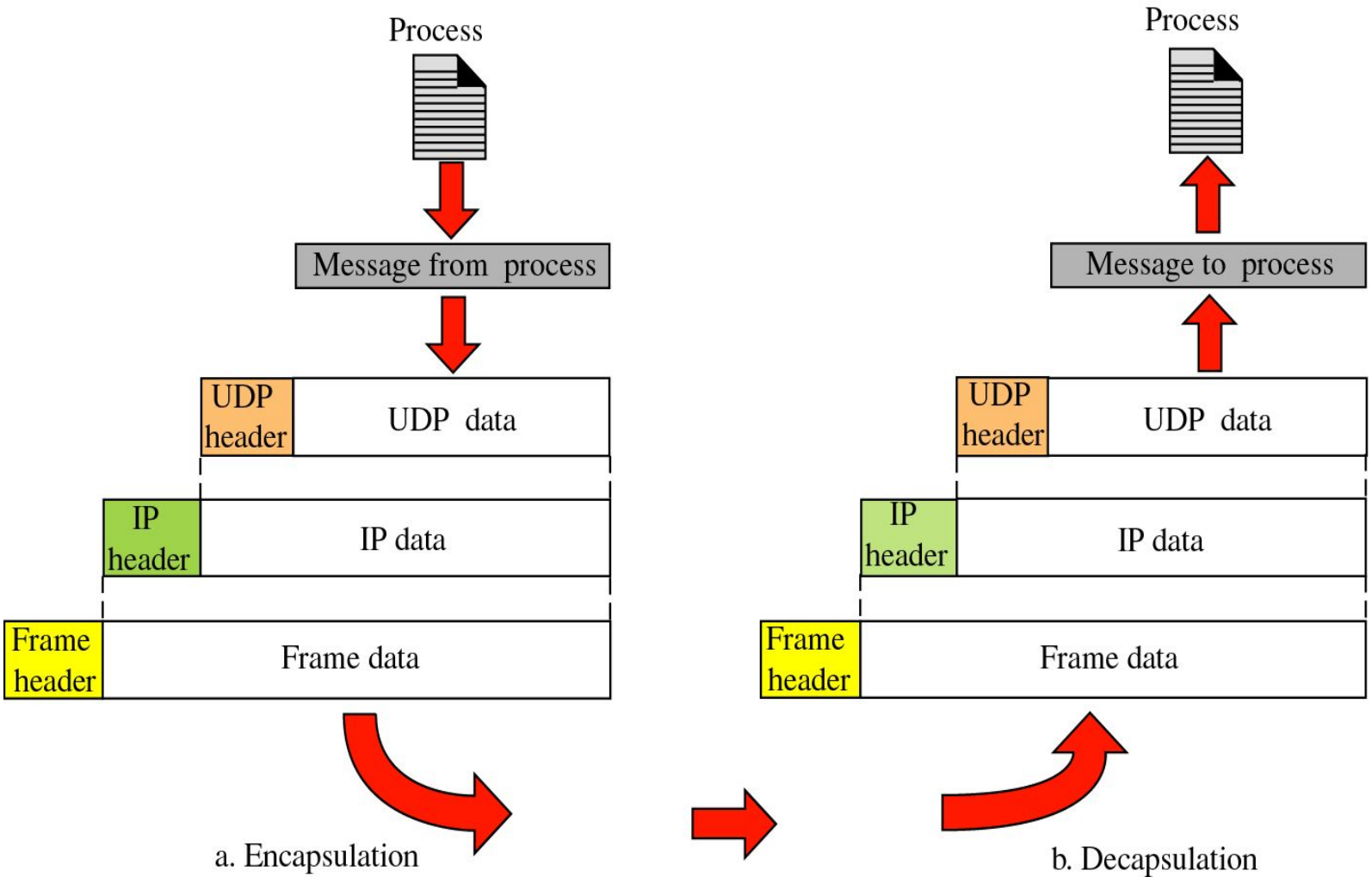
153.18.8.105			
171.2.14.10			
All 0s	17	15	
1087		13	
15		All 0s	
T	E	S	T
I	N	G	All 0s

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

UDP Operation

- Connectionless Services
  - each user datagram sent by UDP is an independent datagram
  - each user datagram can travel a different path
- Flow and error control
  - no flow control, hence no windowing mechanism
    - The receiver may overflow with incoming messages
  - no error control mechanism except for the checksum
    - the sender does not know if a message has been lost or duplicated
  - So, the process using UDP provides these mechanism

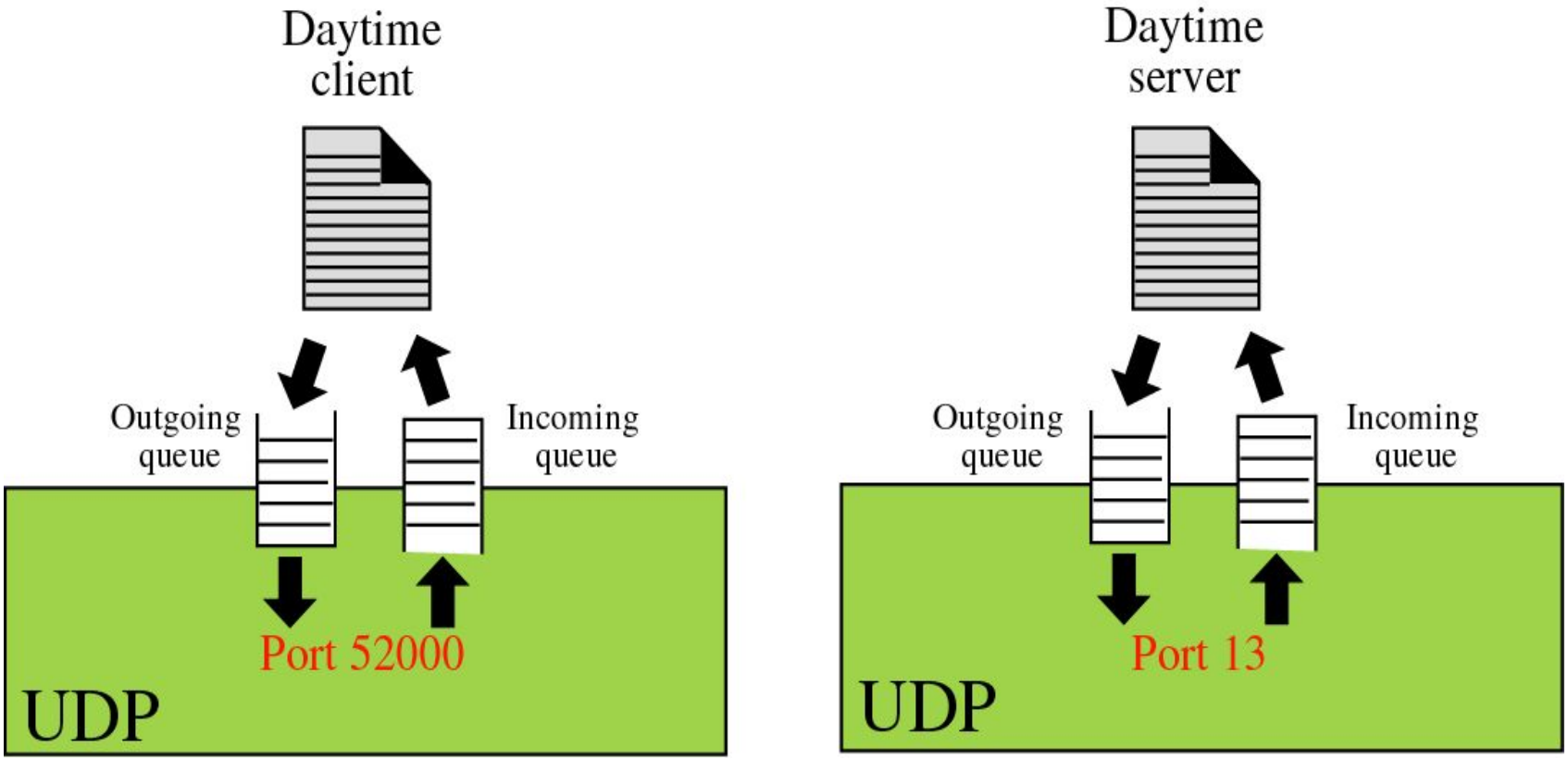
UDP Operation (cont'd)



UDP Operation (cont'd)

- Queuing
  - The queues function as long as the process is running.
  - If an outgoing queue is happened overflow, the operating system can ask the client process to wait before sending any more messages.
  - When a message arrives for a client, check an incoming queue. If there is no such incoming queue, UDP discard the user datagram and ask the ICMP protocols to send a port unreachable message to the server.
  - At the server, the queues remain open as long as the server is running
    - An outgoing queue can overflow. If this happens, the operating system asks the server to wait before sending any more messages

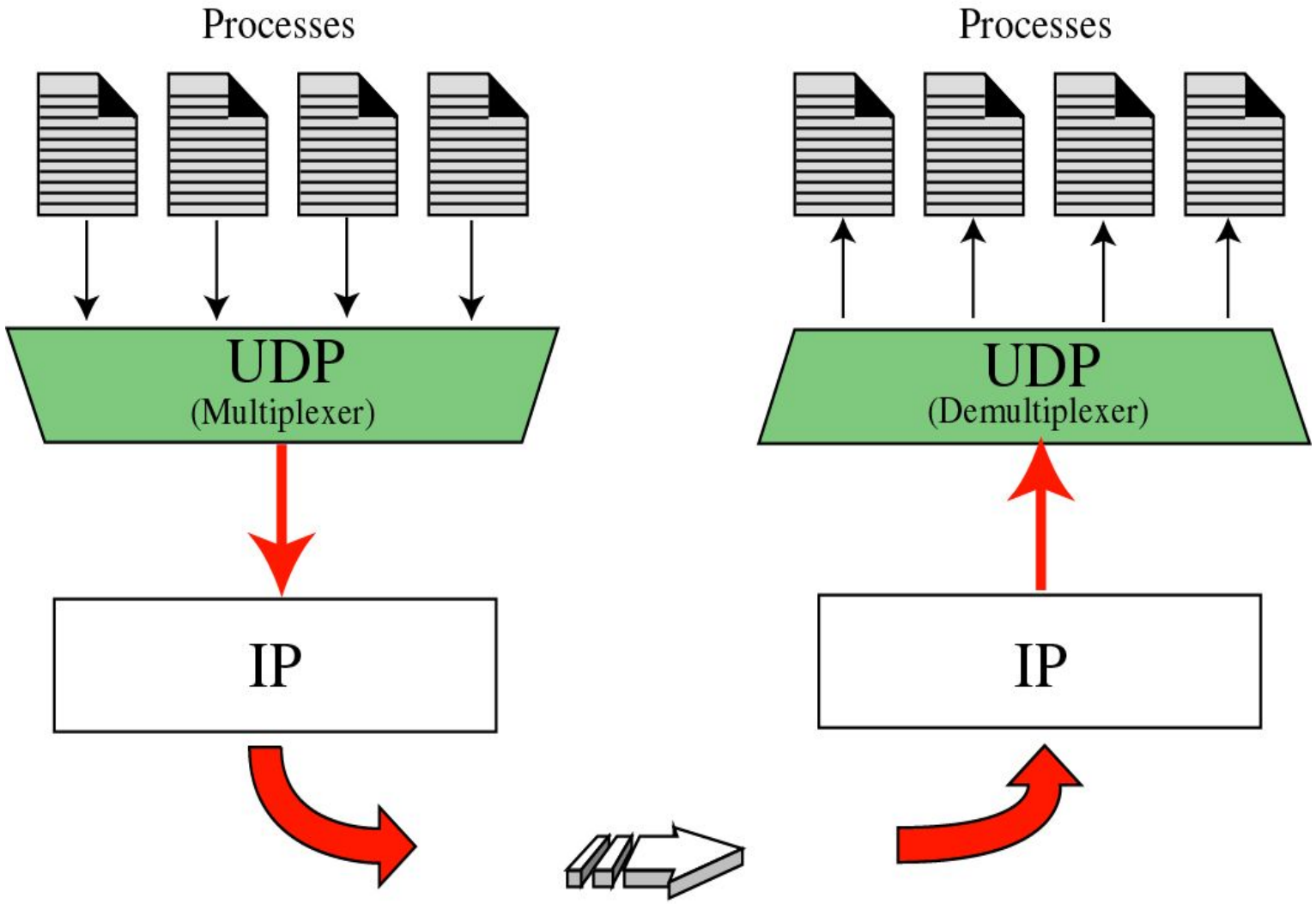
UDP Operation (cont'd)



UDP Operation (cont'd)

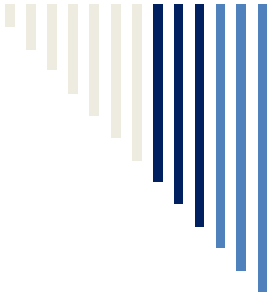
- Multiplexing
  - At the sender site, there may be several processes that need to send user datagrams
    - differentiating by their assigned port numbers
- Demultiplexing
  - At the receiver site, there is only one UDP
    - UDP receives user datagrams from IP.
    - After error checking and dropping of the header, UDP delivers each message to the appropriate port based on the port numbers

UDP Operation (cont'd)

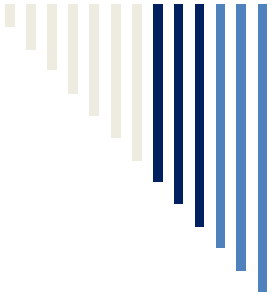


Use of UDP

- The following lists some uses of the UDP protocols
  - suitable for a process that requires simple request-response communication and with little concern for flow and error control
    - not used for a process that needs to send bulk data, such as FTP
  - suitable for a process with internal flow and error control mechanisms
    - TFTP process including flow and error control
  - suitable transport protocol for multicasting and broadcasting
    - multicasting and broadcasting capabilities are embedded in the UDP software, but not in the TCP software
- used for management protocol such as SNMP
- used for some route updating protocol such as RIP



Transport layer protocols



STCP

SCTP

Stream Control Transmission Protocol (SCTP) is a new reliable, message-oriented transport layer protocol. SCTP, however, is mostly designed for Internet applications that have recently been introduced. These new applications need a more sophisticated service than TCP can provide.

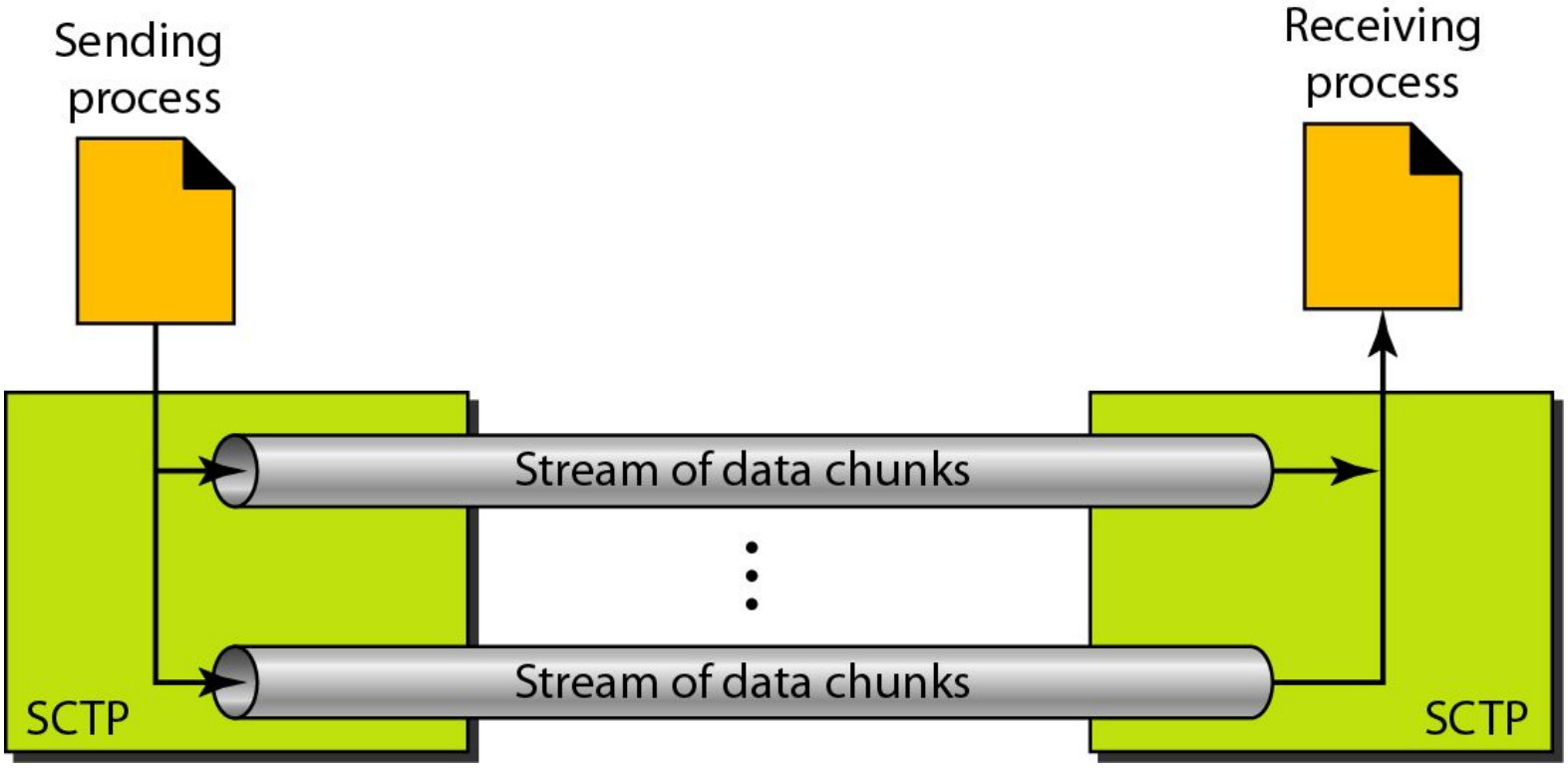
Topics discussed in this section:

SCTP Services and Features
Packet Format
An SCTP Association
Flow Control and Error Control

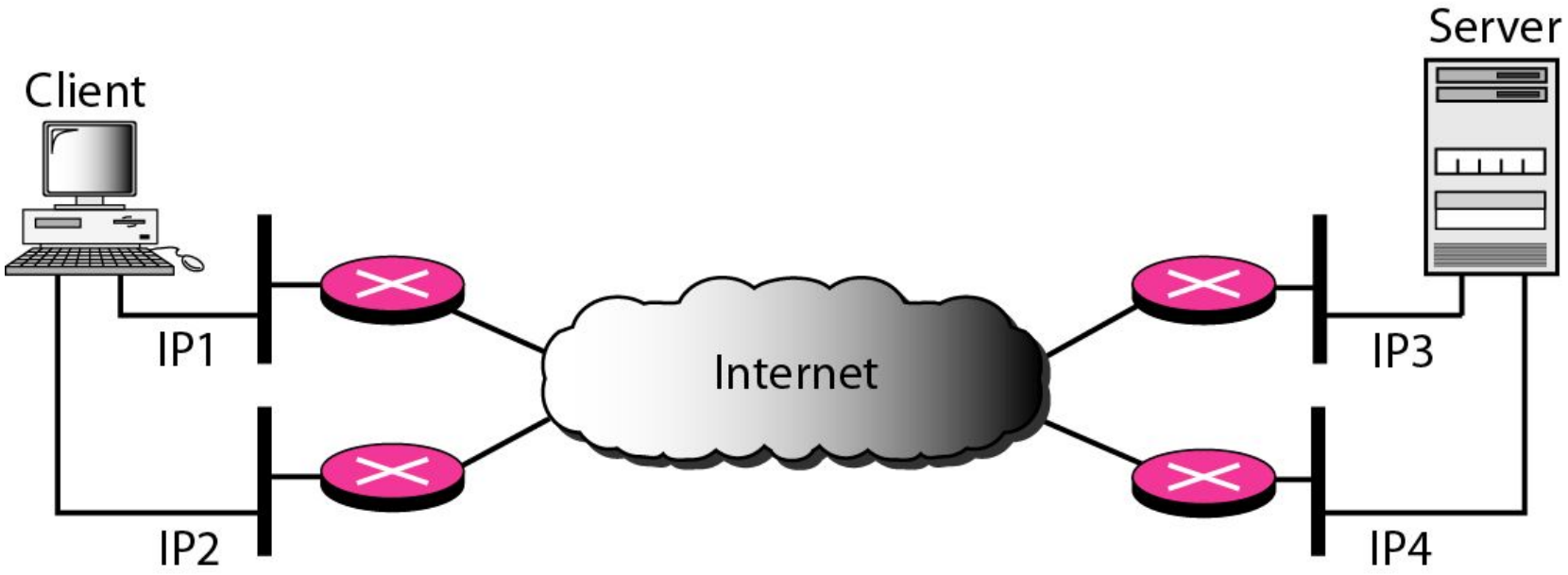
SCTP

- SCTP is a message-oriented, reliable protocol that combines the best features of UDP and TCP.
- An association in SCTP can involve multiple streams .
- Supports Multistreaming facility.
- Supports Multihoming .

Multiple-stream concept



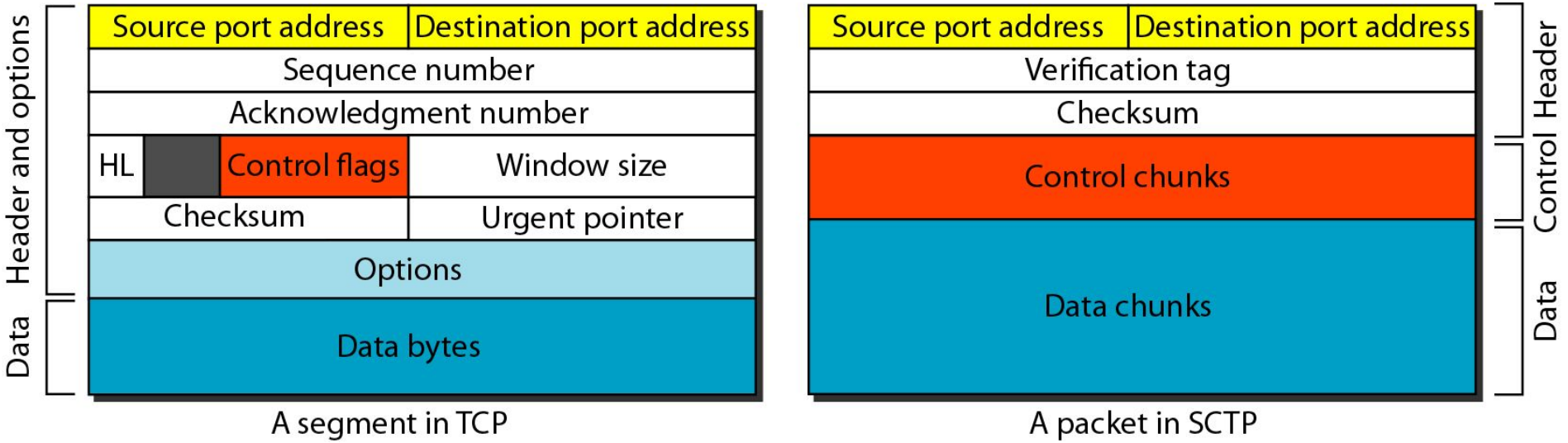
Multihoming concept



SCTP

- In SCTP, a data chunk is numbered using a TSN.
- To distinguish between different streams, SCTP uses an SI.
- To distinguish between different data chunks belonging to the same stream, SCTP uses SSNs.
- TCP has segments; SCTP has packets.

Comparison between a TCP segment and an SCTP packet

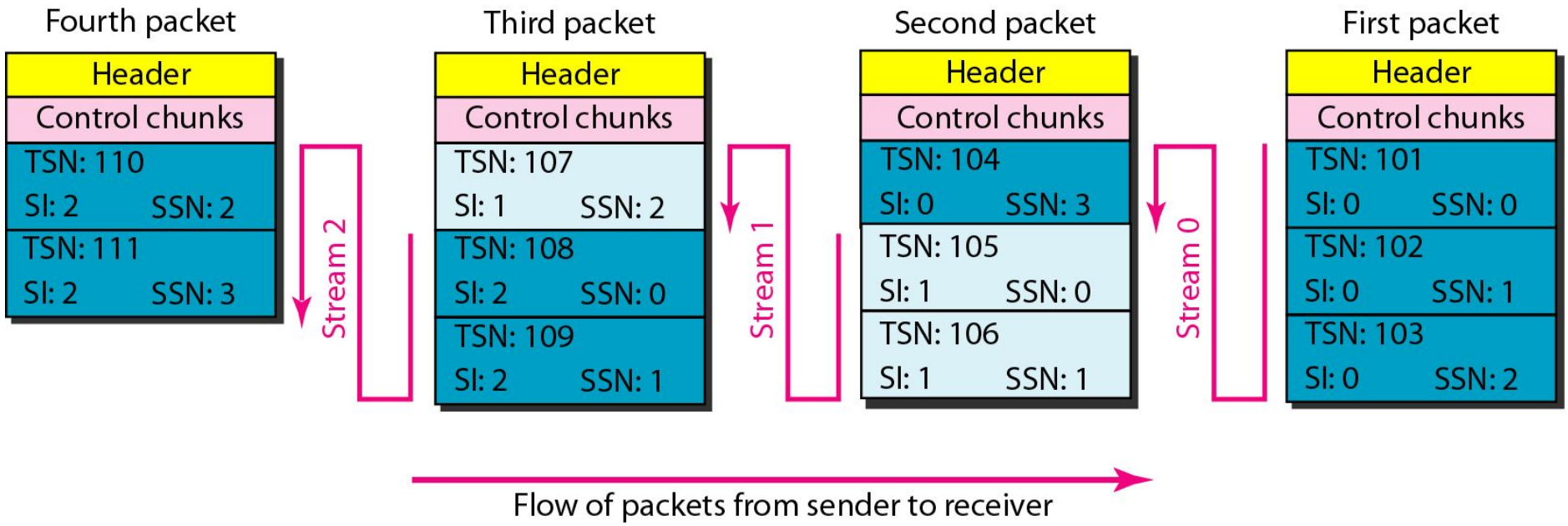


SCTP

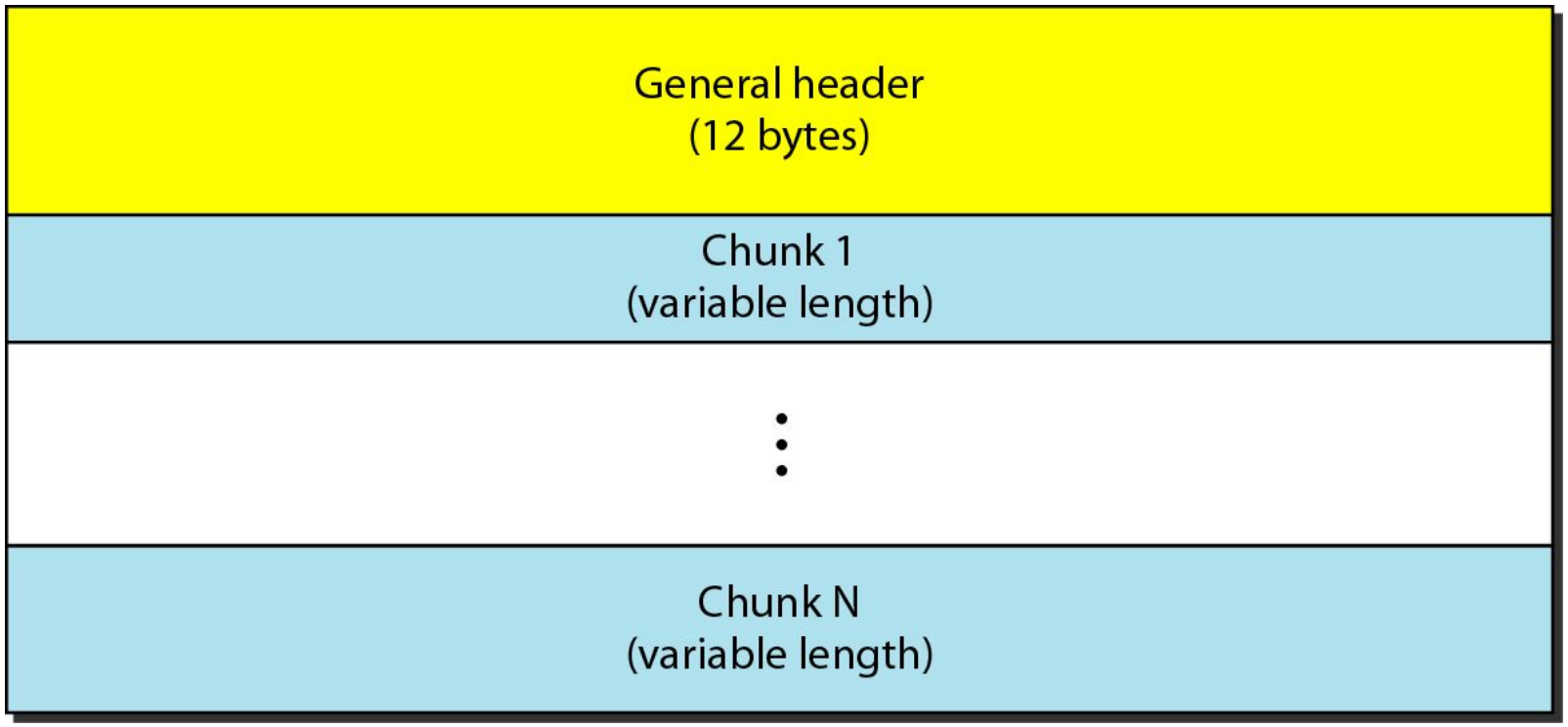
- In SCTP, control information and data information are carried in separate CHUNKS
- Data chunks are identified by three items: TSN, SI, and SSN.

- TSN is a cumulative number identifying the association; SI defines the stream; SSN defines the chunk in a stream.
- In SCTP, acknowledgment numbers are used to acknowledge only data chunks;
- control chunks are acknowledged by other control chunks if necessary.
- In SCTP, acknowledgment numbers are used to acknowledge only data chunks;
- control chunks are acknowledged by other control chunks if necessary.

Packet, data chunks, and streams



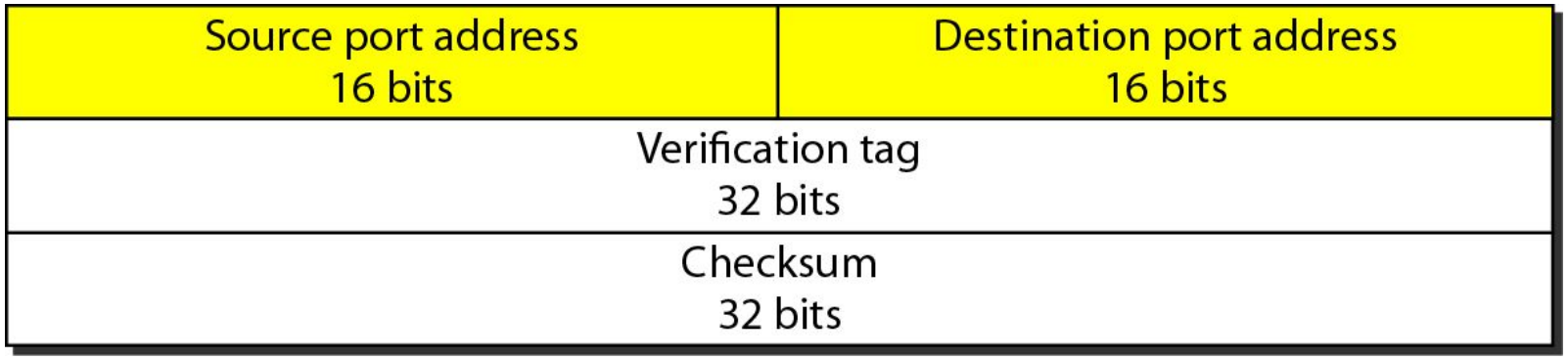
SCTP packet format



SCTP

- In an SCTP packet, control chunks come before data chunks
- No other chunk is allowed in a packet carrying an INIT or INIT ACK chunk.
- A COOKIE ECHO or a COOKIE ACK chunk can carry data chunks.
- In SCTP, only DATA chunks consume TSNs.
- DATA chunks are the only chunks that are acknowledged.
- The acknowledgment in SCTP defines the cumulative TSN, the TSN of the last data chunk received in order.

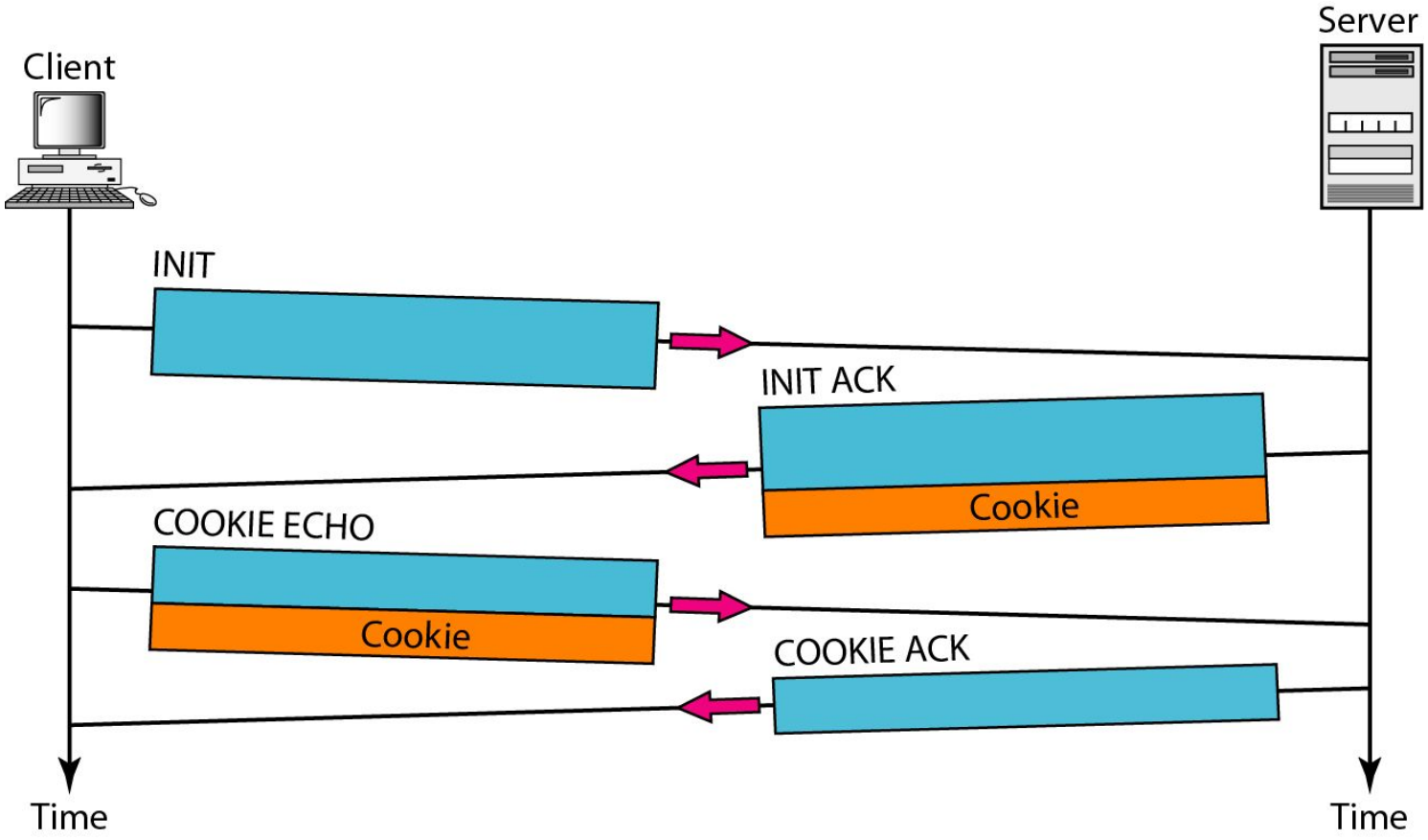
General header



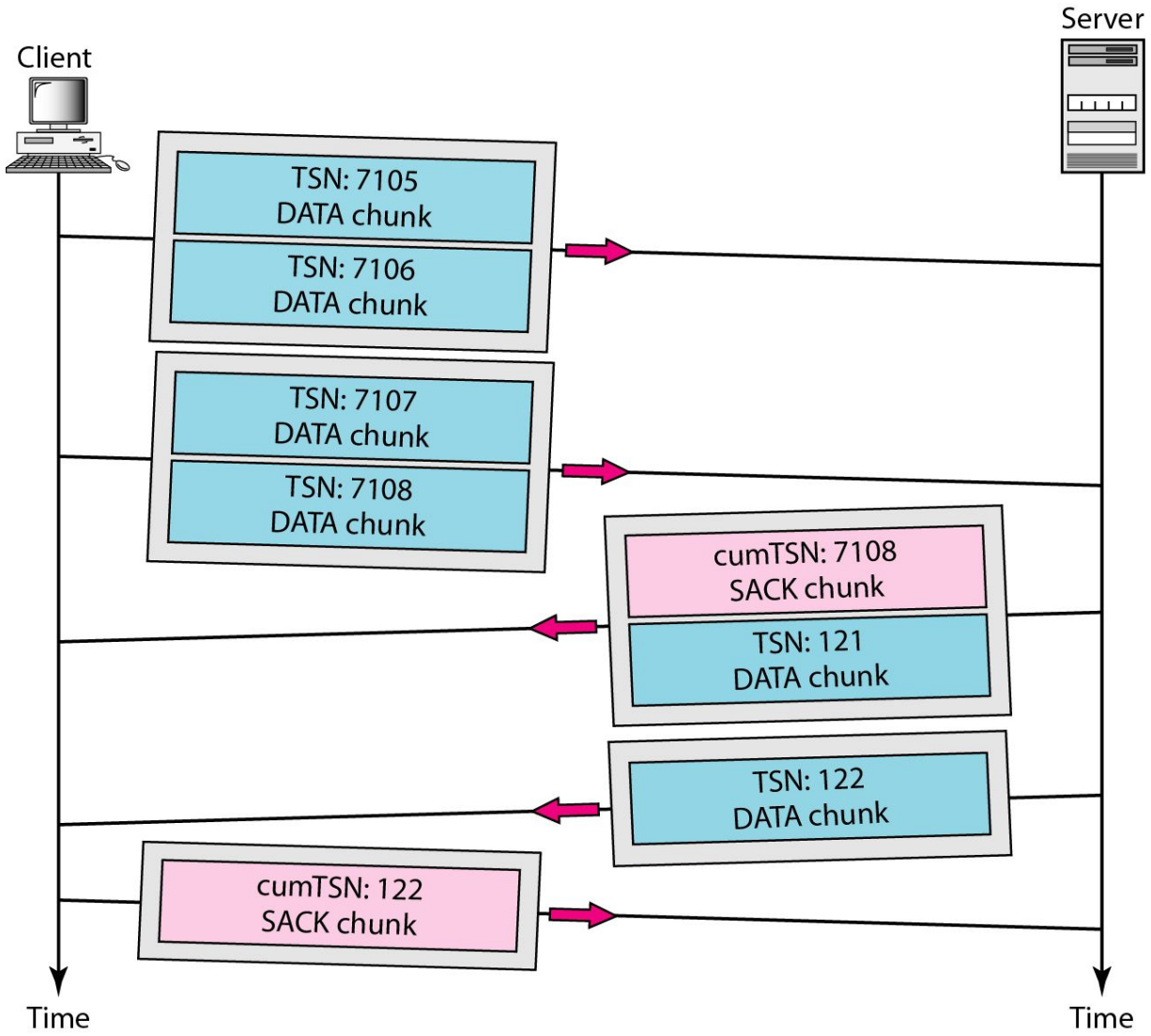
Chunks

Type	*Chunk*	*Description*
0	**DATA**	User data
1	**INIT**	Sets up an association
2	**INIT ACK**	Acknowledges INIT chunk
3	**SACK**	Selective acknowledgment
4	**HEARTBEAT**	Probes the peer for liveliness
5	**HEARTBEAT ACK**	Acknowledges HEARTBEAT chunk
6	**ABORT**	Aborts an association
7	**SHUTDOWN**	Terminates an association
8	**SHUTDOWN ACK**	Acknowledges SHUTDOWN chunk
9	**ERROR**	Reports errors without shutting down
10	**COOKIE ECHO**	Third packet in association establishment
11	**COOKIE ACK**	Acknowledges COOKIE ECHO chunk
14	**SHUTDOWN COMPLETE**	Third packet in association termination
192	**FORWARD TSN**	For adjusting cumulative TSN

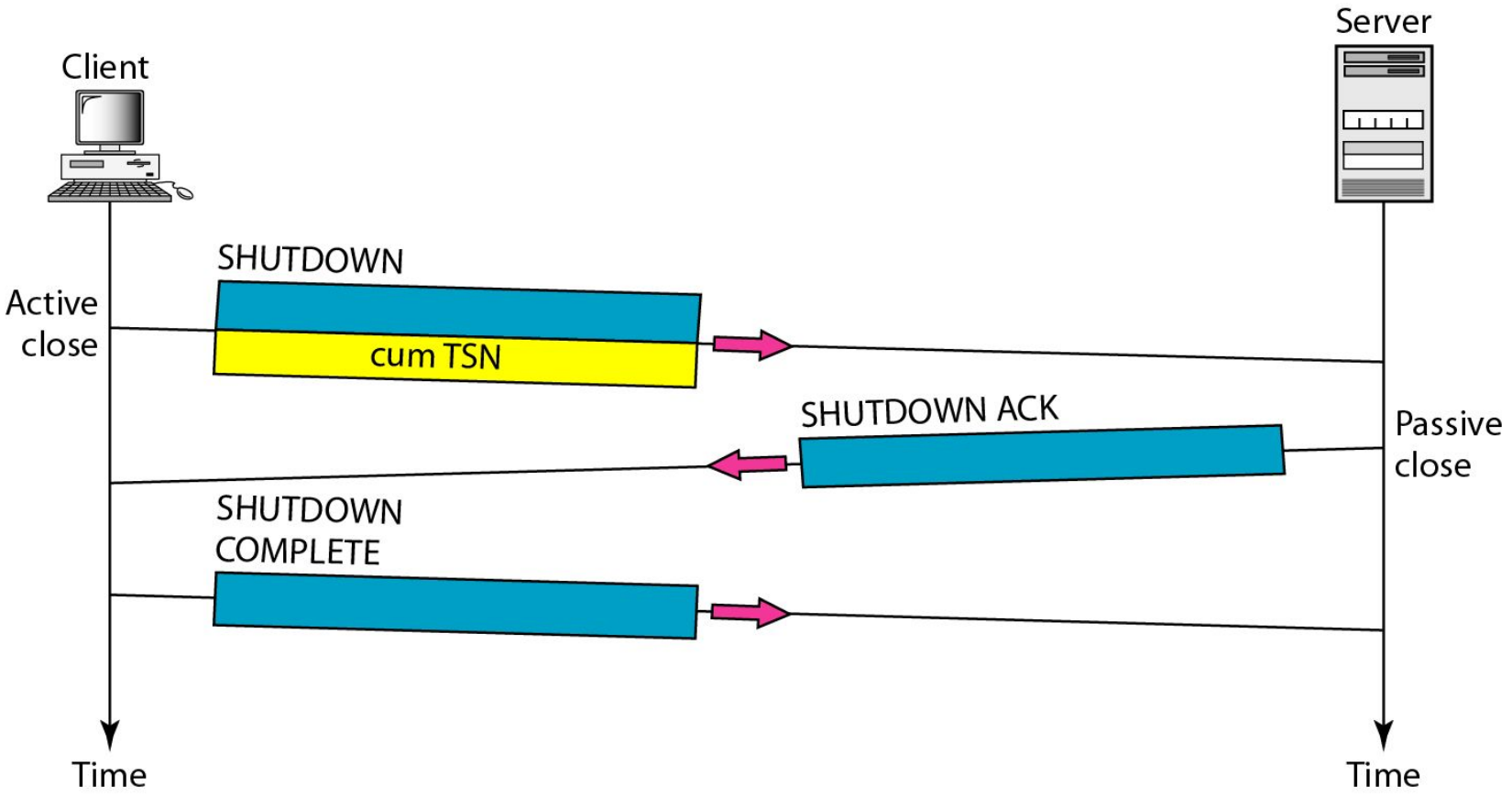
Four-way handshaking



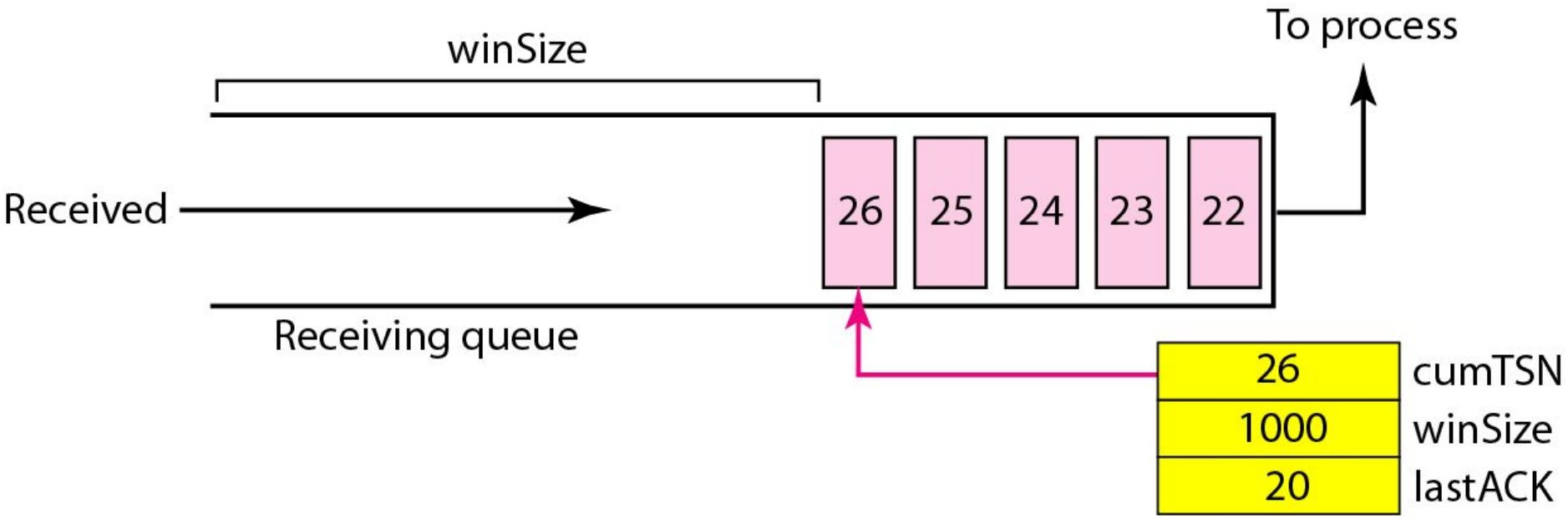
Simple data transfer



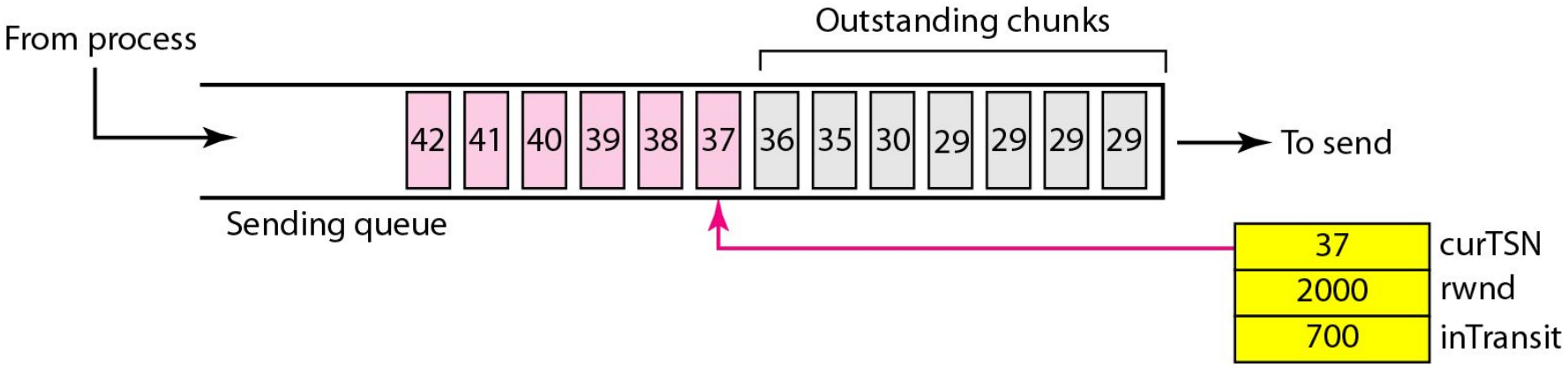
Association termination



Flow control, receiver site



Flow control, sender site



Flow control scenario

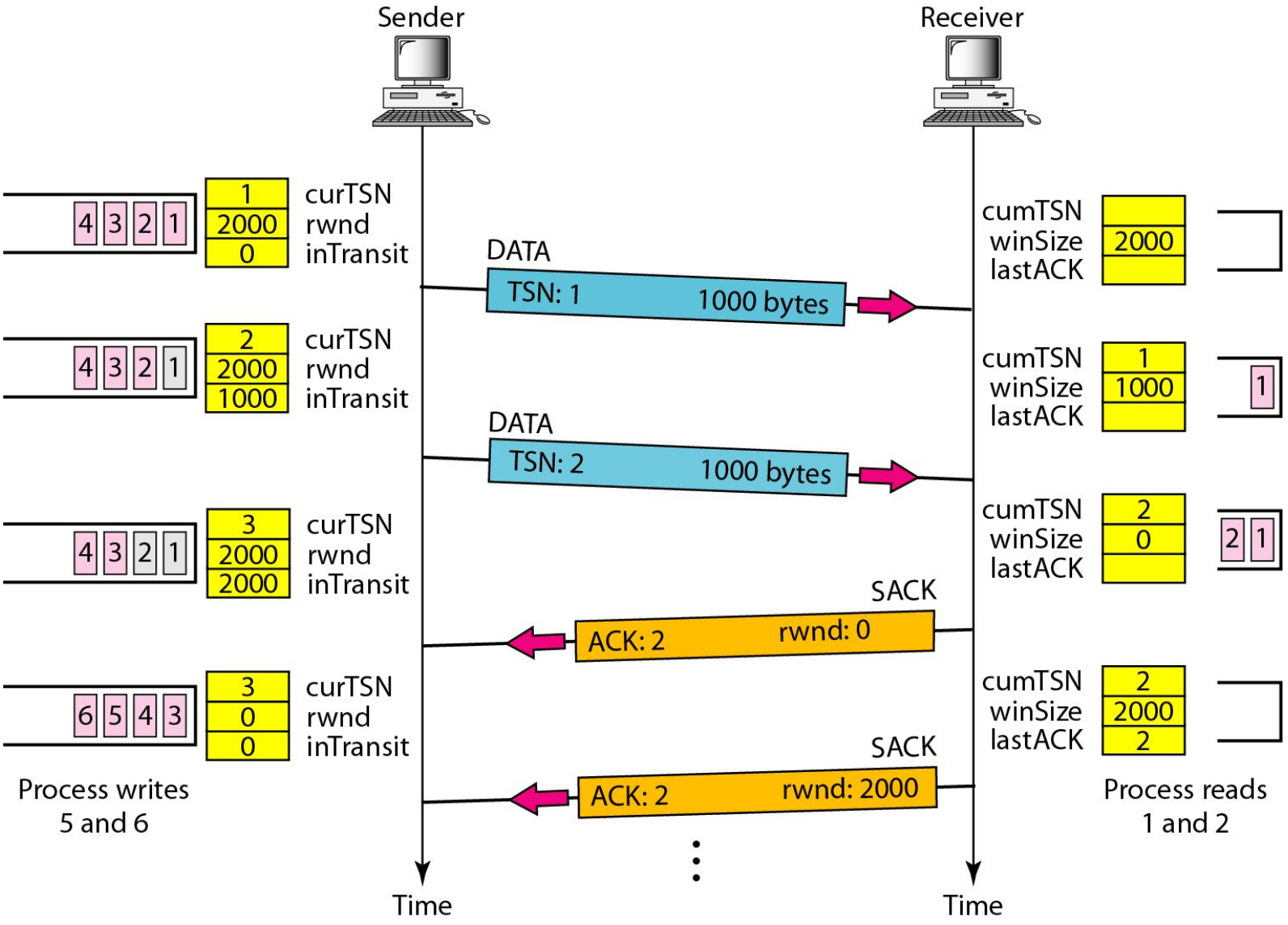
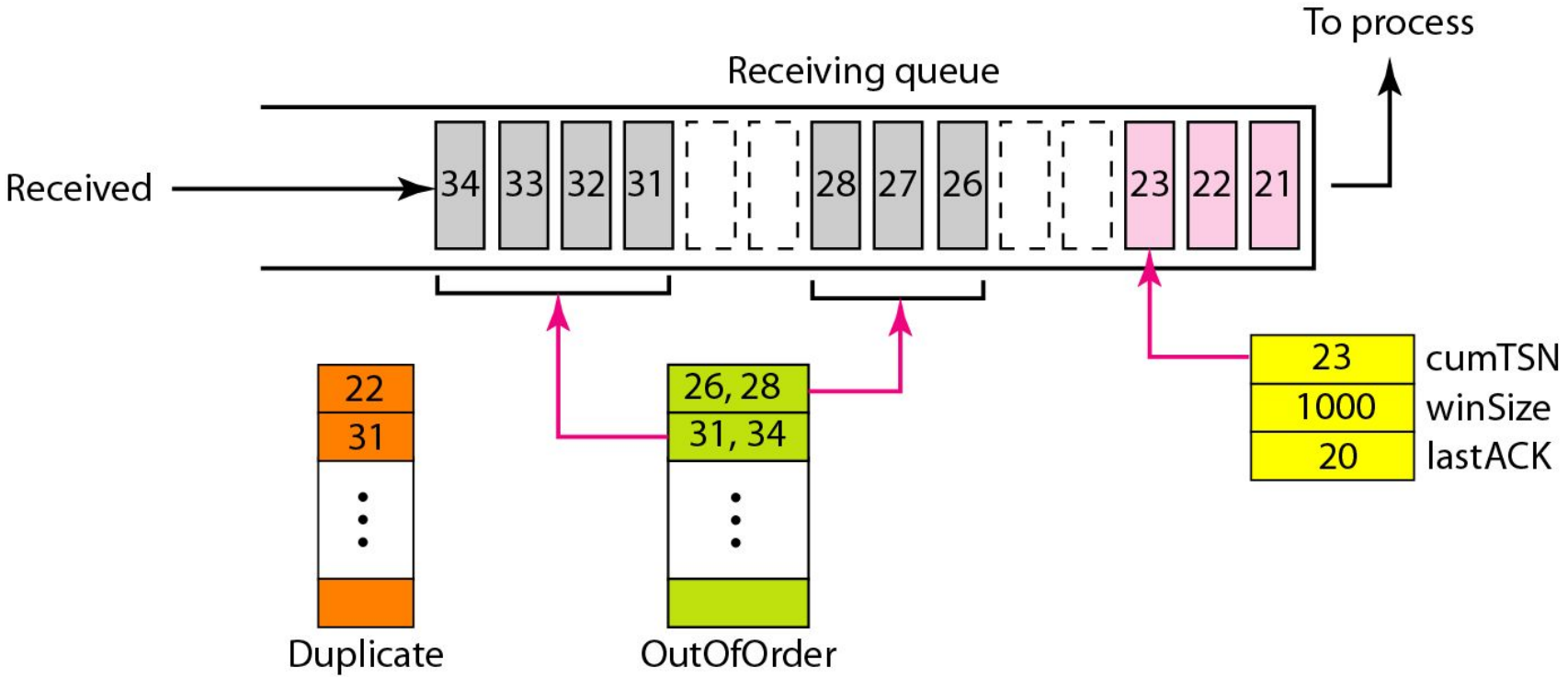
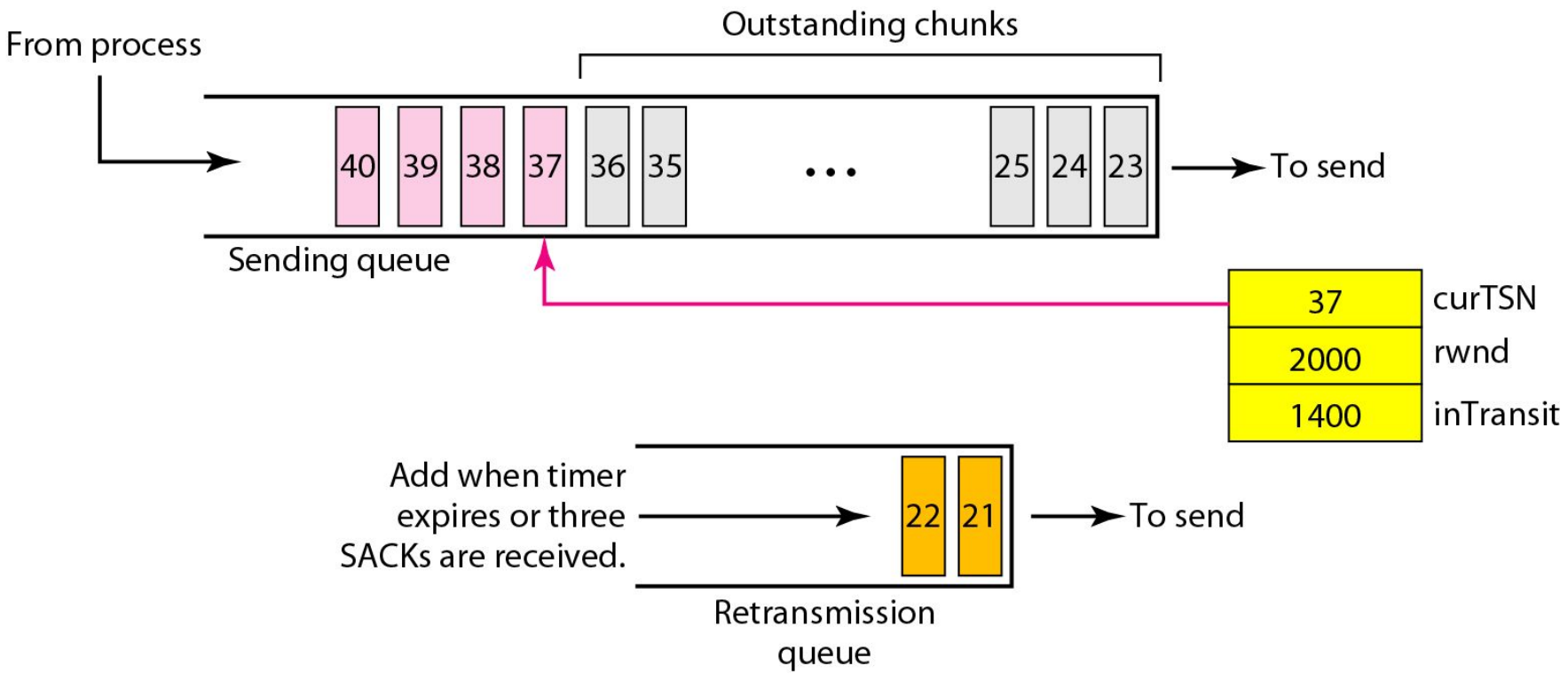


Figure 23.39 *Error control, receiver site*



Error control, sender site



SCTP

- SCTP provides the call signaling over IP

- The benefits of SCTP, multi-homing and multi-streaming, will allow development of numerous new devices which include the ability to support and carry this new protocol.

SUMMARY

- Transport layer provides connection oriented service and connectionless service.
- Transport layer protocols are TCP,UDP & SCTP.
- TCP provides connection oriented service.its reliable one
- UDP provides connectionless service.its unreliable
- SCTP is used in multistreaming.

REVIEW QUESTIONS

1. What are the functions of transport layer?
2. List the features of TCP protocol?
3. Give the differences between TCP & UDP?
4. What are the special features of UDP & SCTP?
5. Give the differences between TCP ,UDP & SCTP?

THANK YOU