Diploma in Computer Hardware Engineering

COMPUTER NETWORKS

(5151 Rev 2015)

Module 1 - Part 1 of 3

Presenter: Sreejesh NG
Lecturer in Computer Hardware Engineering
Government Polytechnic College, Cherthala

Ref: Data Communications and Networking, 5E by Forouzan

Course General Outcomes:

Sl.	G.O	On completion of this course the student will be able :
1	1	To Understand the concept of TCP/IP Protocol
2	1	To Understand the concept of Network Layer
3	1	To Understand the concept of Transport Layer
4	1	To Understand the concept of Application Layer

Specific Outcomes:

MODULE I. REVIEW OF NETWORK MODELS

1.1 Understand TCP/IP Protocol

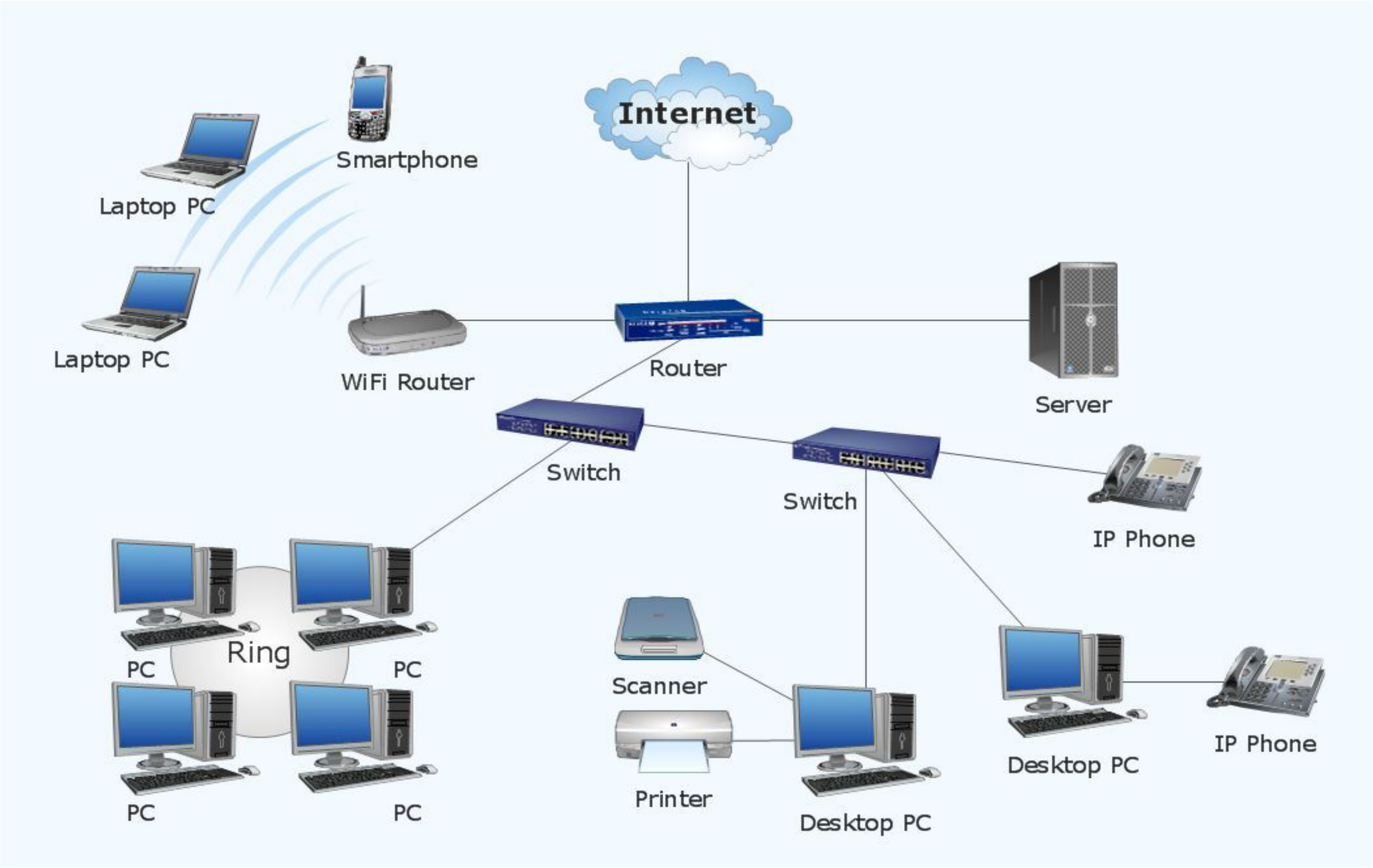
1.1.1 Illustrate computer networks
1.1.2 Identify TCP/IP Protocol suite.
1.1.3 Explain the functionalities of layers in TCP/IP
1.1.4 Define Addressing of TCP/IP.
1.1.5 Describe about Wired LAN – Ethernet
1.1.6 State IEEE 802 project
1.1.7 Illustrate standard Ethernet
1.1.8 Describe about Wireless LAN.
1.1.9 State IEEE 802.11
1.1.10 Explain LAN connecting devices.
1.1.11 Explain the architecture of Virtual LANs.

MODULE 1 - TCP/IP PROTOCOL

- Introduction to computer networks - physical structure, topology, types.
- TCP/IP - architecture, Description of layers, addressing.
- Wired LAN - Ethernet protocol - IEEE project 802 - Standard Ethernet - characteristics, addressing, implementation.
- Wireless LAN - architectural comparison, characteristics, access control - IEEE 802.11 - architecture.
- LAN connecting devices - hub, switch, router.
- Virtual LAN - architecture, membership, configuration.

Ref: Data Communications and Networking - Behrouz A. Forouzan -
McGraw Hill Edn.-Fourth Edition/Fifth Edition

What is a network?

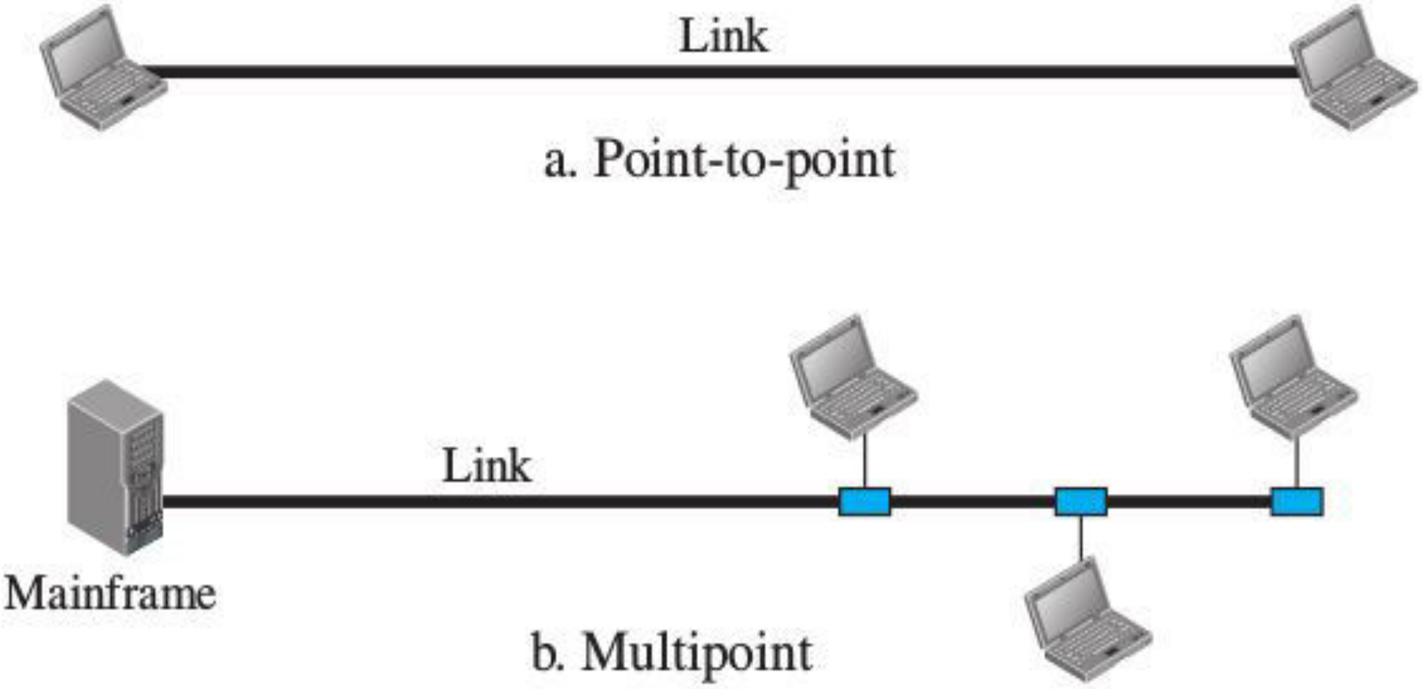


What is a network?

- The interconnection of a set of devices capable of communication.
- A device can be a *host* (or an end system) such as PC, mobiles,..
- A device can be a *connecting device* such as a router, switch, modem,..
- All these devices are connected through wired or wireless *transmission media* such as cable or air.

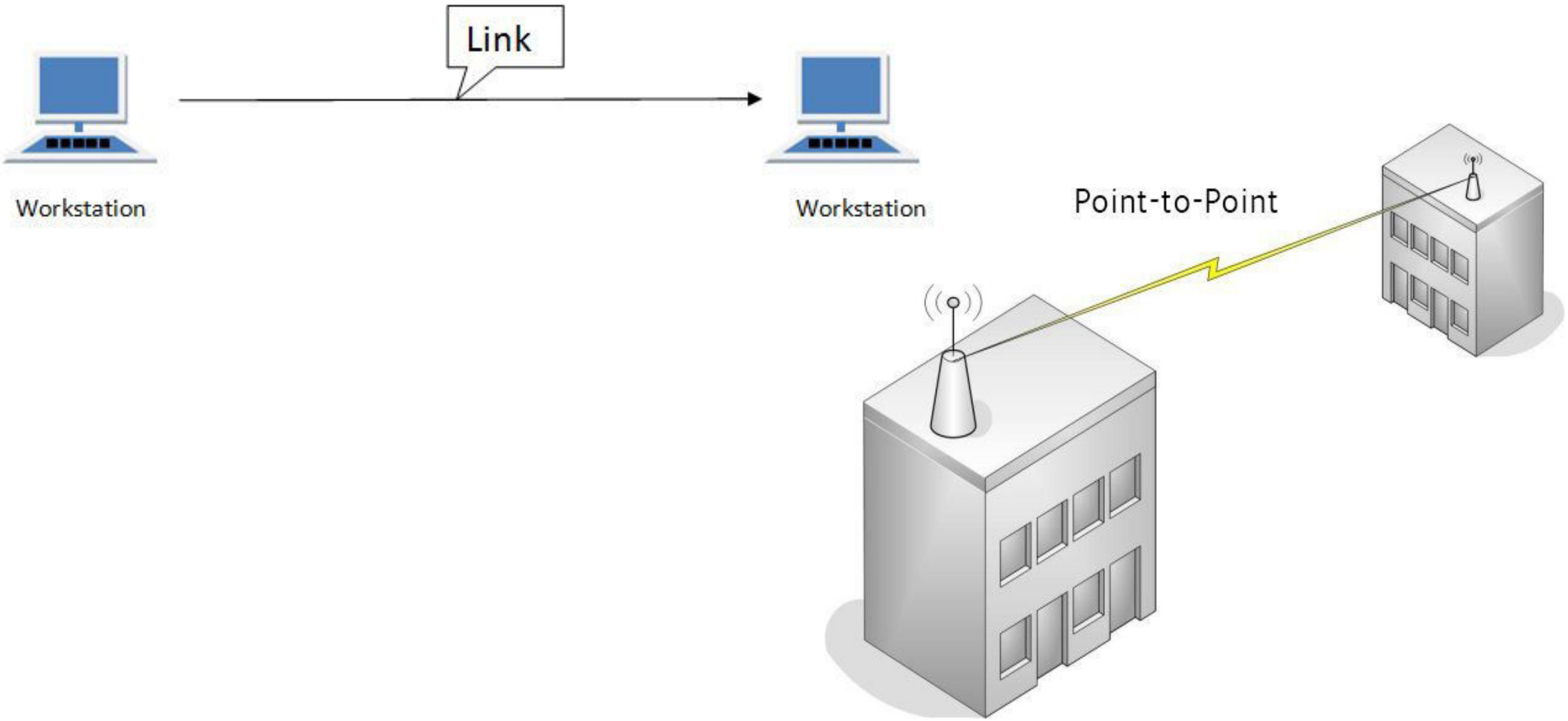
Physical Structures

- Defines how the devices are connected physically.
- The connections/communication pathways are called **links.**
- There are two possible types of connections: **point-to-point** and **multipoint**.



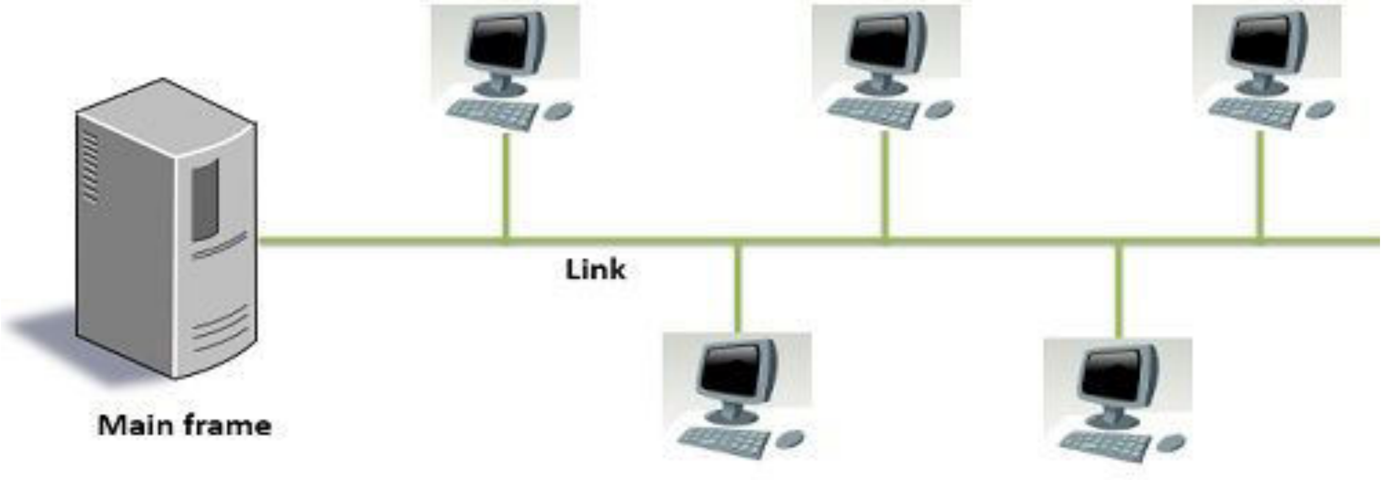
Point-to-point Connections

- Provides a dedicated link between two devices.
- The entire capacity of the link is used by the devices.
- The connections can be wired or wireless.



Multipoint (multidrop) Connections

- More than two specific devices share a single link.
- The capacity of the channel is **shared**, either spatially or temporally.
- If all the devices use the link simultaneously, it is **spatially shared**.
- If users must take turns, it is a **timeshared** connection.



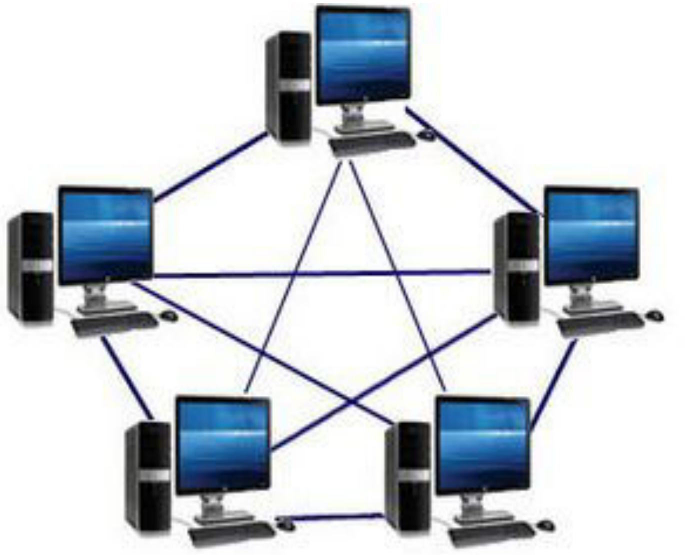
Multipoint Connection

Physical Topology

- The way in which a network is laid out physically.
- two or more links form a topology.
- **The topology of a network is the geometric representation of the relationship of all the links and linking devices (usually called nodes) to one another**.
- There are four basic topologies possible (as per text):
  - Mesh
  - Star
  - Bus
  - Ring.
- Many other forms are also possible.

Mesh Topology

- Every device has a dedicated point-to-point link to every other device.
- If there are *N* nodes we need *N(N-1)/2* bidirectional physical links.

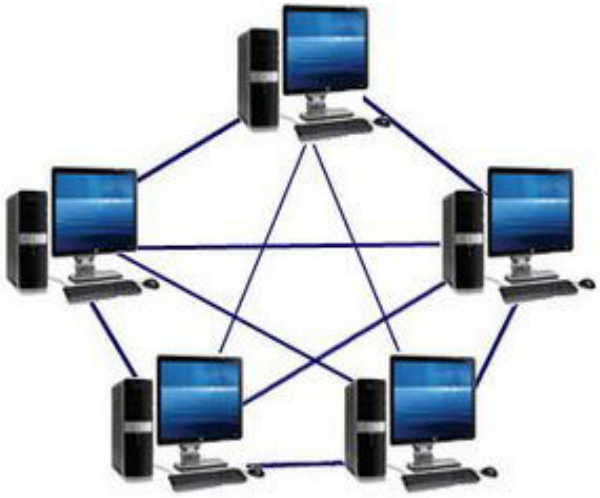


Advantages

- Each link carries its own data load.
- No traffic issues.
- Robust. If one link goes down, it does not interrupt others.
- Privacy and security.
- Fault identification and fault isolation easy.

Mesh Topology

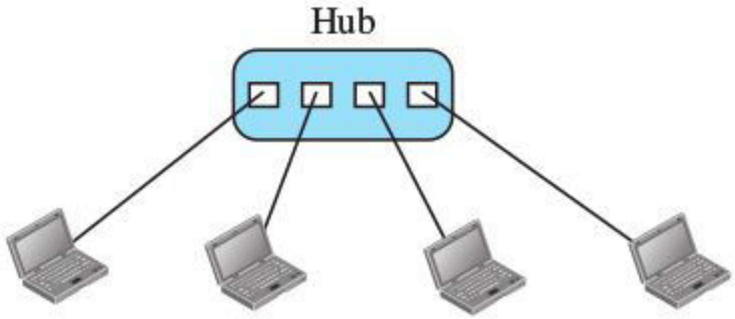
Disadvantages



- Amount of cabling and the number of I/O ports are high, hence expensive.
- installation and reconnection are difficult.
- Bulk amount of cabling reduces space.

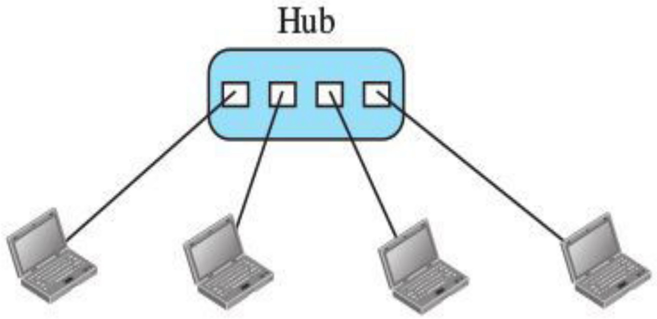
- One practical example of a mesh topology is the connection of telephone regional offices in which each regional office needs to be connected to every other regional office.

Star Topology



- Each device has a dedicated point-to-point link only to a central controller, usually called a hub.
- The devices are not directly linked to one another.
- If one device wants to send data to another, it sends the data to the controller, which then relays the data to the other connected device.
- Most commonly used in LANs.

Star Topology



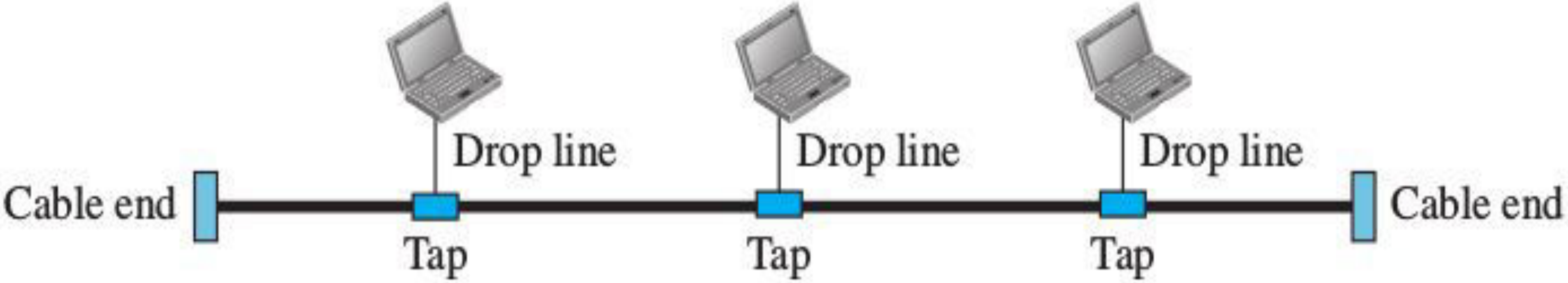
Advantages

- Each device needs only one link
- and one I/O port to connect it to any number of others.
- Hence less expensive.
- Easy to install and reconfigure.
- Robust: If one link goes down, only that node is affected.
- Easy fault identification.

Disadvantages

- If the central node (hub) goes down, entire network will fail.
- More cabling than ring and bus topologies.

Bus Topology



- A multipoint connection network.
- One long cable acts as a backbone to link all the devices in a network.
- Nodes are connected to the bus cable by drop lines and taps.
- Signal weakens at taps, hence the number of taps are limited.

Bus Topology

Advantages

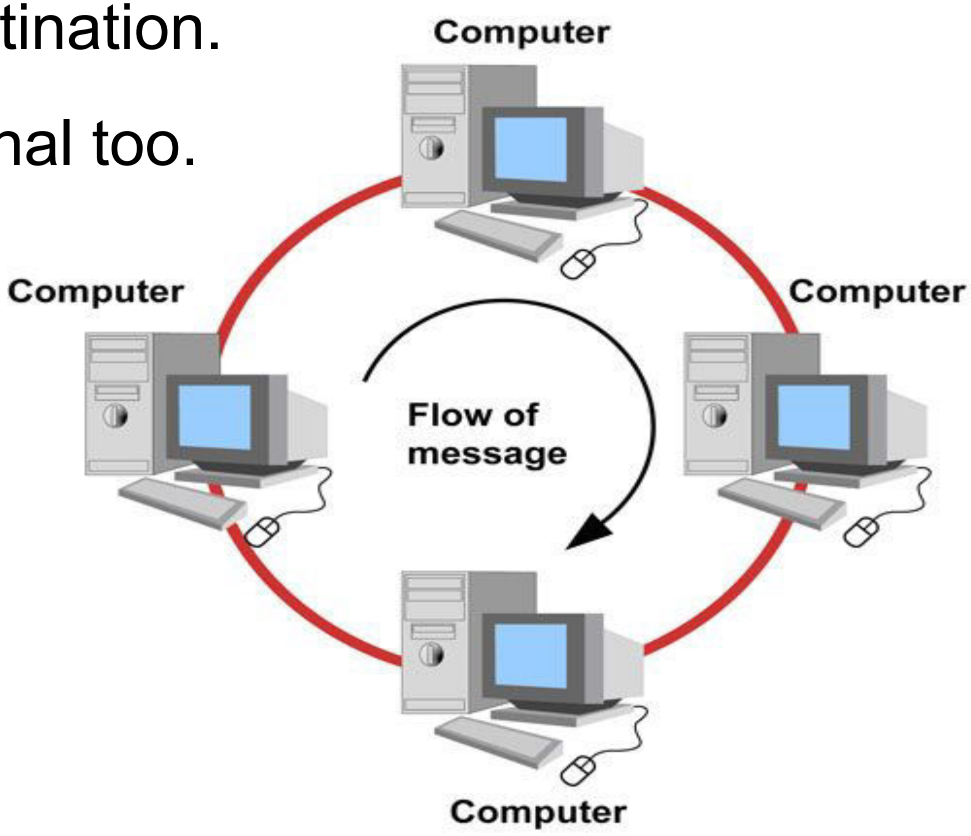
- Easy installation.
- Less expensive.
- Less cabling.

Disadvantages

- Very difficult in fault isolation and reconnection.
- Difficult to add new devices.
- Signal bouncing causes quality degradation.
- Cable break ruins the network on both sides.

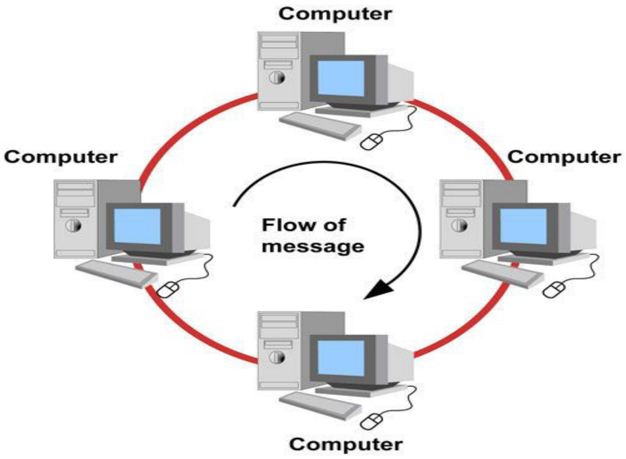
Ring Topology

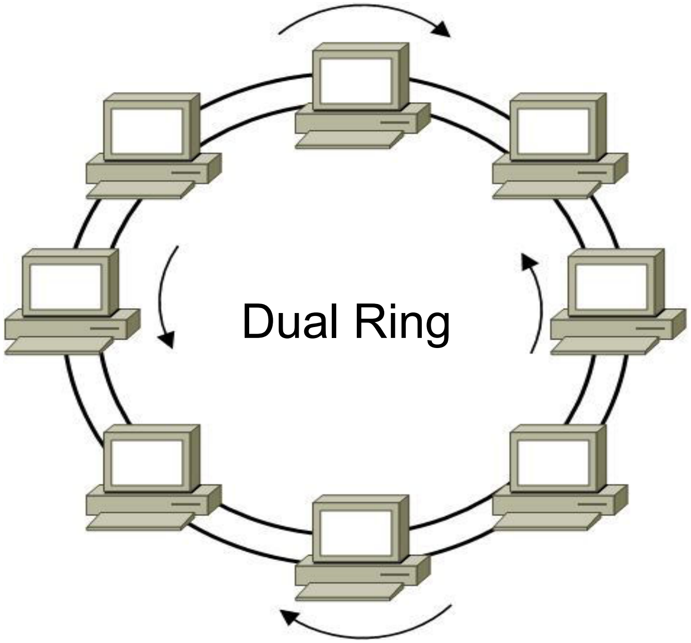
- Each device has a dedicated point-to-point connection with only the two devices on either side of it.
- A signal is passed along the ring in one direction, from device to device, until it reaches its destination.
- Each device amplifies the signal too.



Ring Topology

- In a ring topology, a signal (token) circulates all the time in one direction. When a system want to transmit a packet, it attaches the packet to this signal.
- If a system does not receive the signal within a specific time, it can issue an alarm. Thus faults can be identified.
- Unidirectional traffic is a disadvantage. This can be overcome by dual ring in both directions.





Ring Topology

Advantages

- Easy to install and reconfigure. To add or delete a device requires changing only two connections.
- Fault isolation is simplified.

Disadvantages

- Unidirectional traffic. (Not in *dual ring*)
- In a simple ring, a break in the ring (such as a disabled station) can disable the entire network.

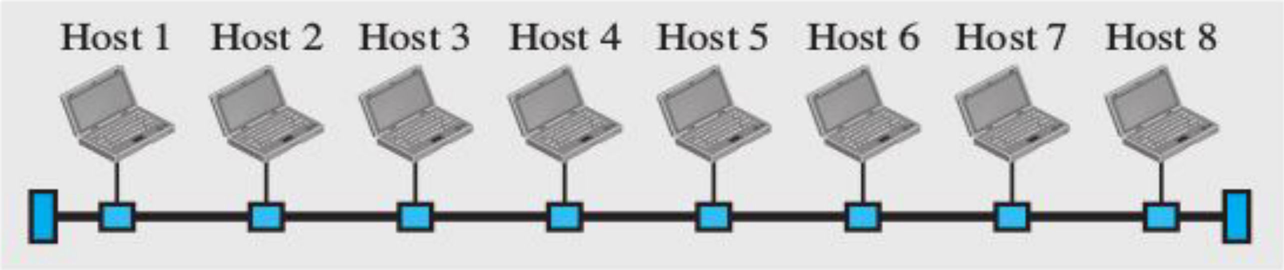
NETWORK TYPES

- Networks can be divided based on many criteria such as topology, geographical coverage, etc.
- Most common type of division is based on geographical coverage.
- Such a division has the following types:
  - Local Area Network (LAN)
  - Wide Area Network (WAN)
  - Some other types are Metropolitan Area Network (MAN), Campus Area Network (CAN), Personal Area Network (PAN), Storage Area Network (SAN), etc.
- The major forms are LAN and WAN

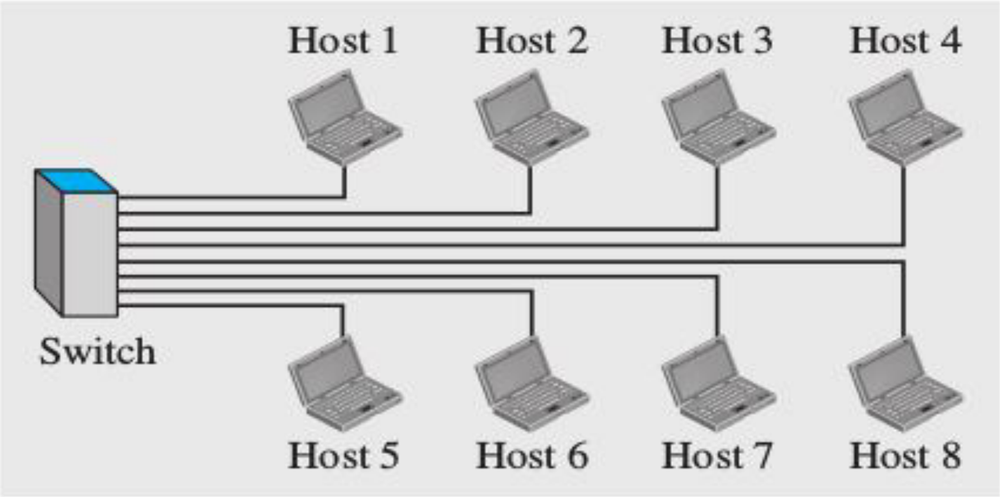
LAN

- LAN is a network that covers relatively short distance.
- A networked office building, school, or home usually contains a single LAN, though sometimes one building will contain a few small LANs (perhaps one per room), and occasionally a LAN will span a group of nearby buildings.
- Most are privately owned.
- In the past, bus topology was used, but now star topololy is used.
- Each LAN has its own identifier. Each member in the LAN has its own address.

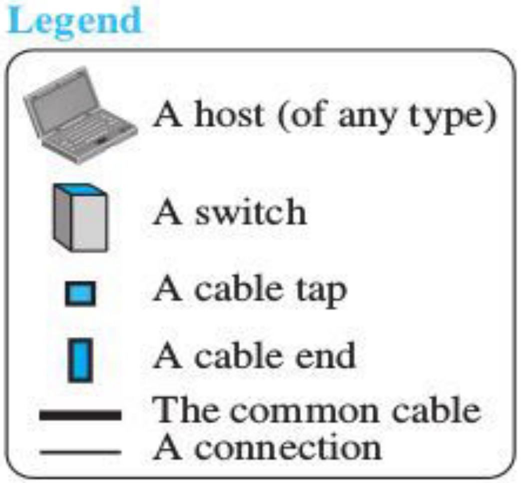
LAN

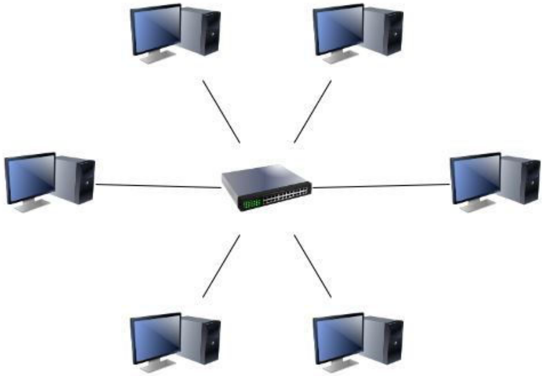


a. LAN with a common cable (past)



b. LAN with a switch (today)



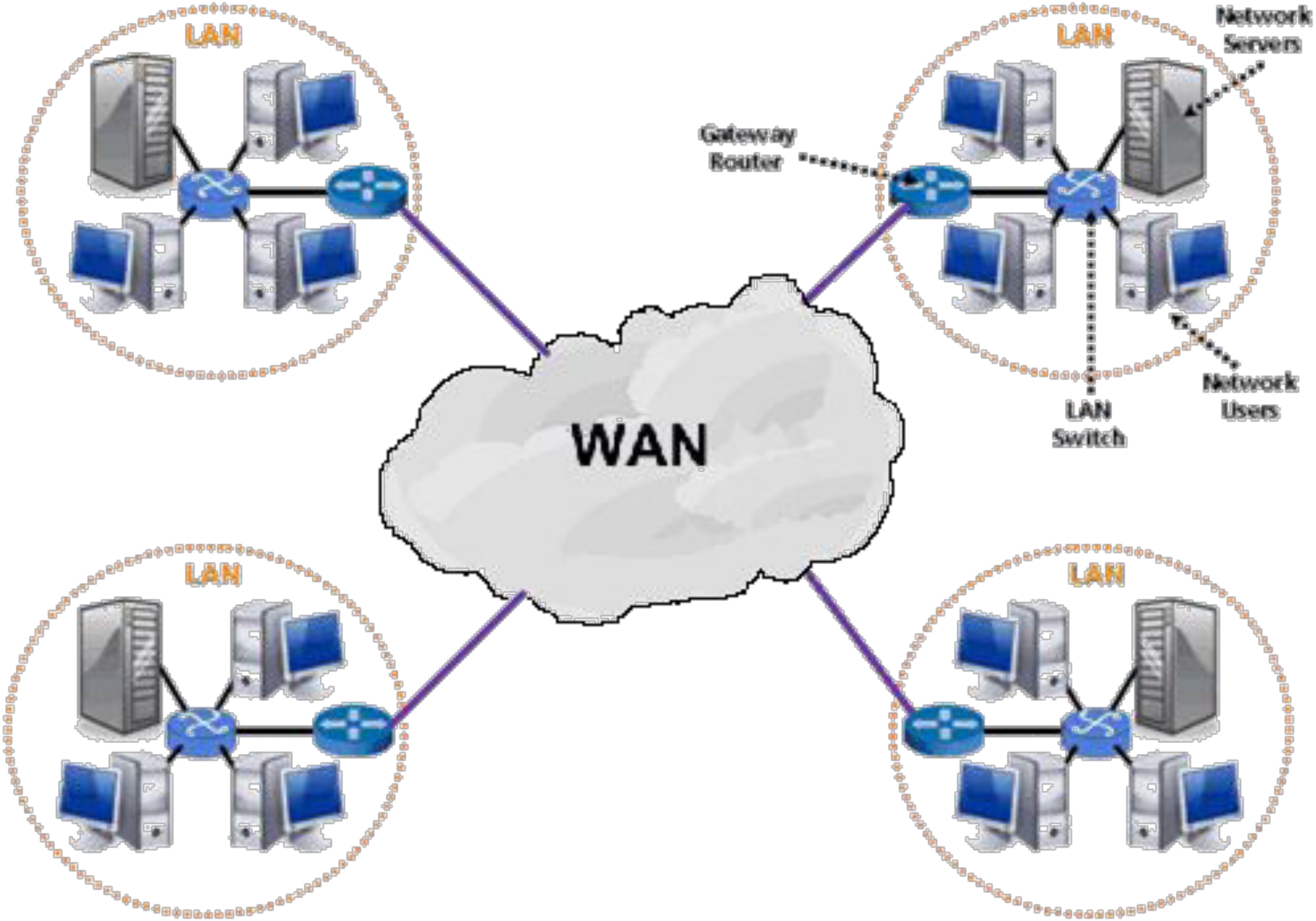


WAN

- A network that covers a large physical distance such as a state, country or continent.
- A WAN contains several LANs.
- The Internet is a WAN that covers the entire earth.
- A LAN interconnects hosts; a WAN interconnects connecting devices such as switches, routers, or modems.
- WAN is normally created and run by communication companies and leased by an organization that uses it.

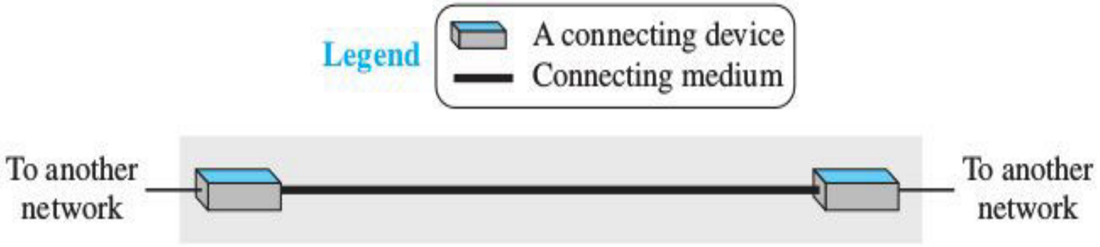
Wide Area Network

WAN

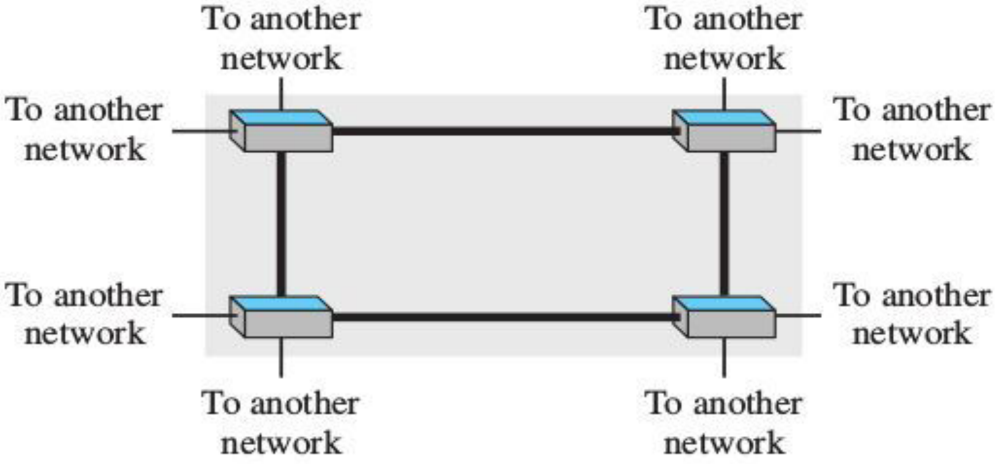

LAN
LAN
Network Servers
Gateway Router
WAN
LAN Switch
Network Users
LAN
LAN

WAN...

- Two distinct examples of WANs today: point-to-point WANs and switched WANs.
- Point-to-Point WAN: Connects two communicating devices through a transmission media (cable or air).

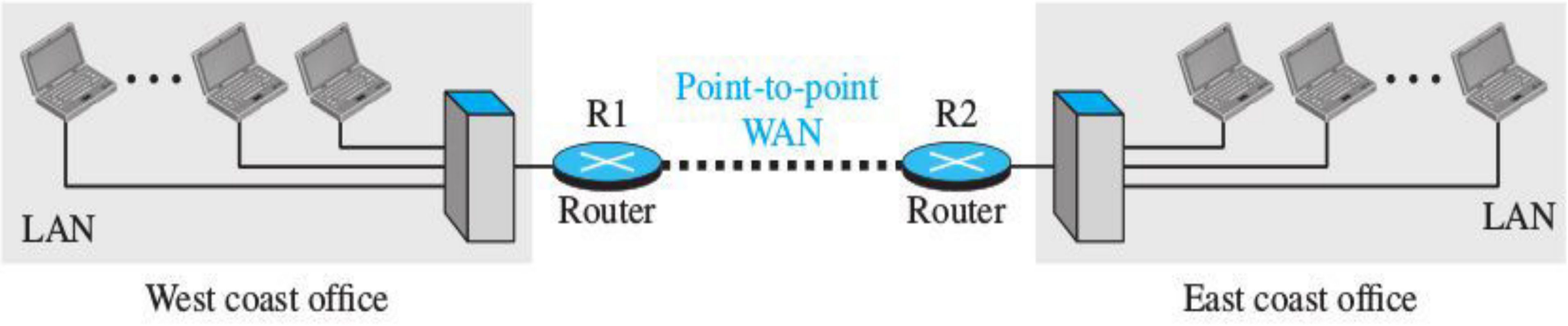


- Switched WAN: It is a network with more than two ends.
  - It is used in the backbone of global communication today.
  - It is a combination of several point-to-point WANs that are connected by switches.



Internetwork

- Today, LANs and WANs are interconnected to make an internetwork or internet.
- Example:
  - Assume that an organization has two offices, one on the east coast and the other on the west coast.
  - To make the communication between employees at different offices possible, the management leases a point-to-point dedicated WAN from a service provider, such as a telephone company, and connects the two LANs.

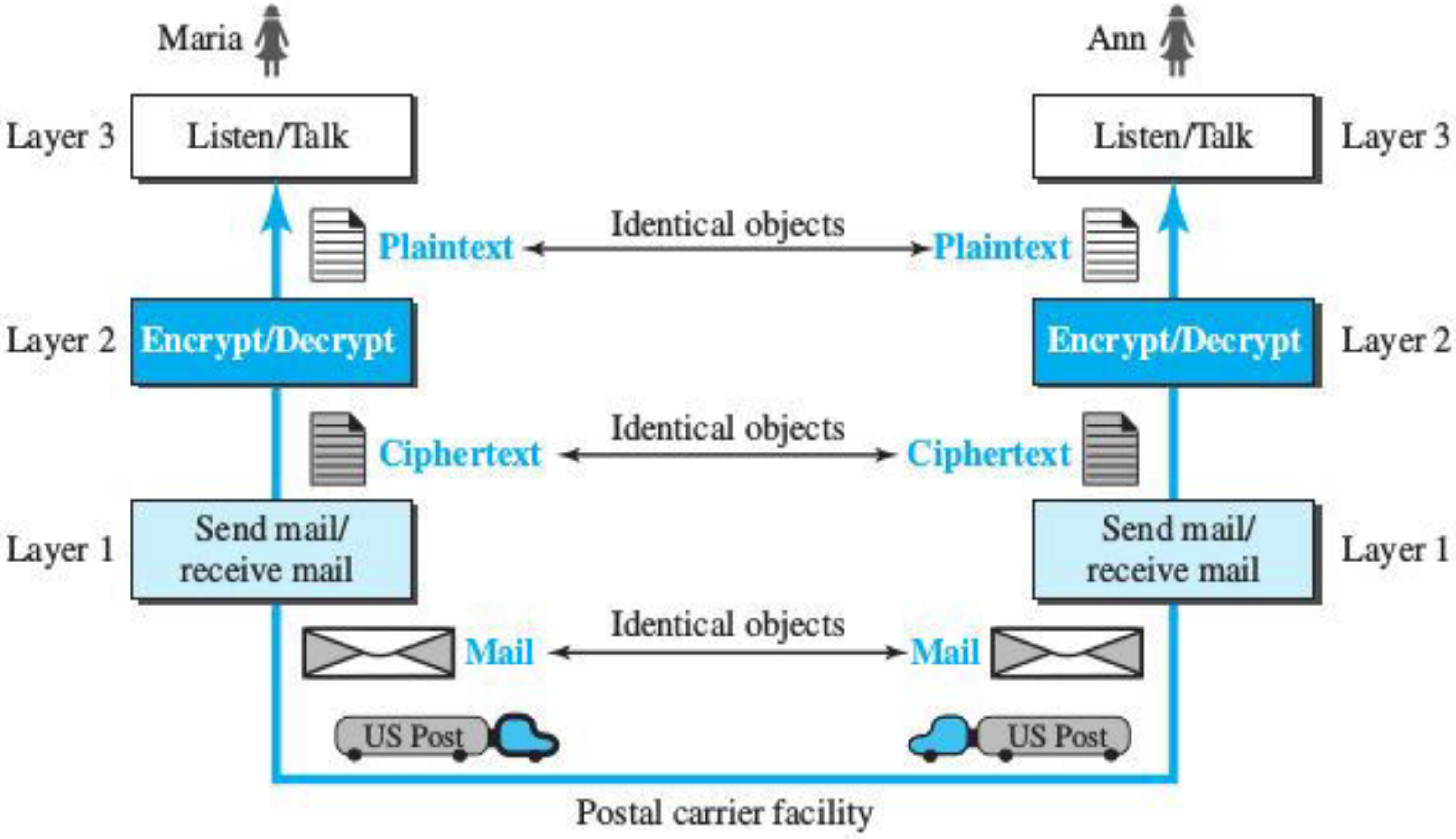


PROTOCOL LAYERING

- Protocol: a protocol defines the rules that both the sender and receiver and all intermediate devices need to follow to be able to communicate effectively.
- The functions to be performed are divided into different layers; this is called protocol layering.
- A layer receives a set of services from the lower layer and gives the services to the upper layer.
- There are intermediate systems that need only some layers, but not all layers.

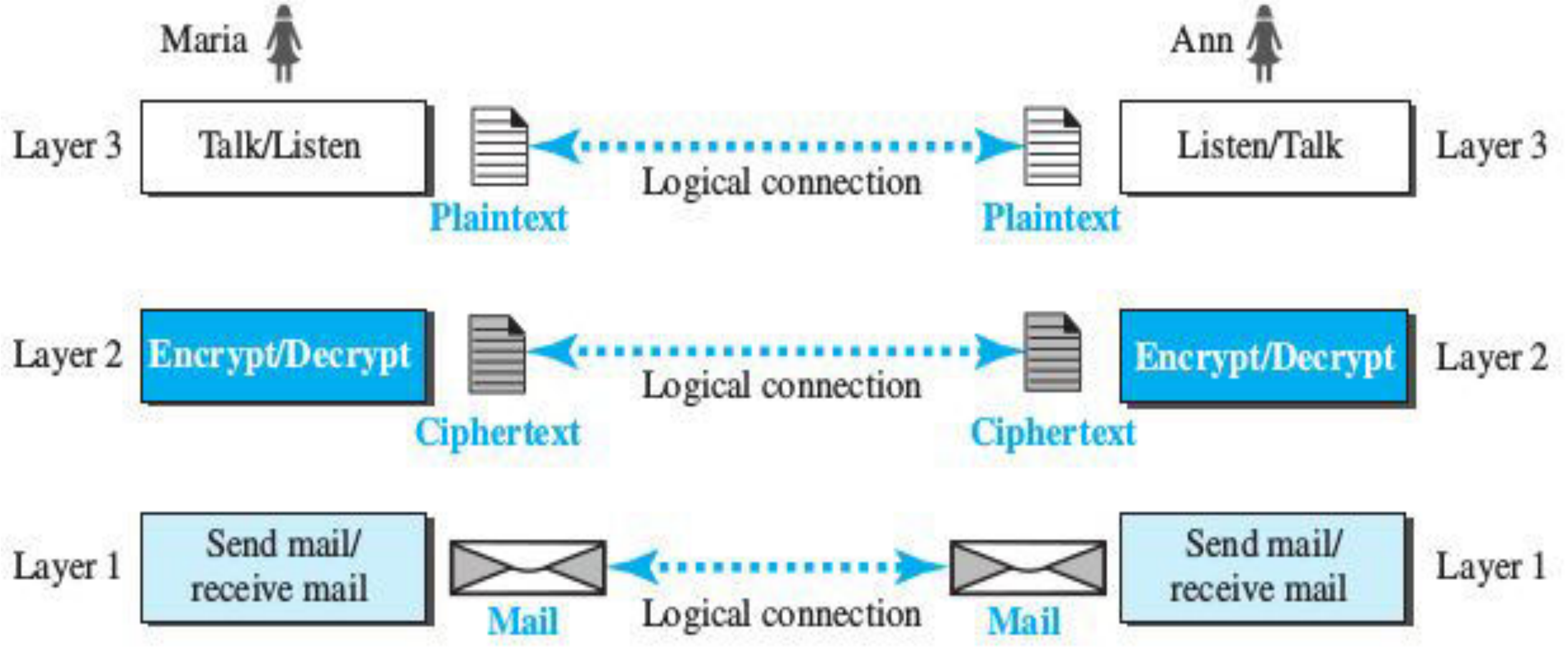
PRINCIPLES OF PROTOCOL LAYERING

- First Principle: If we want bidirectional communication, we need to make each layer so that it is able to perform two opposite tasks, one in each direction.
- Second Principle: The two objects under each layer at both sites should be identical.



Logical Connections

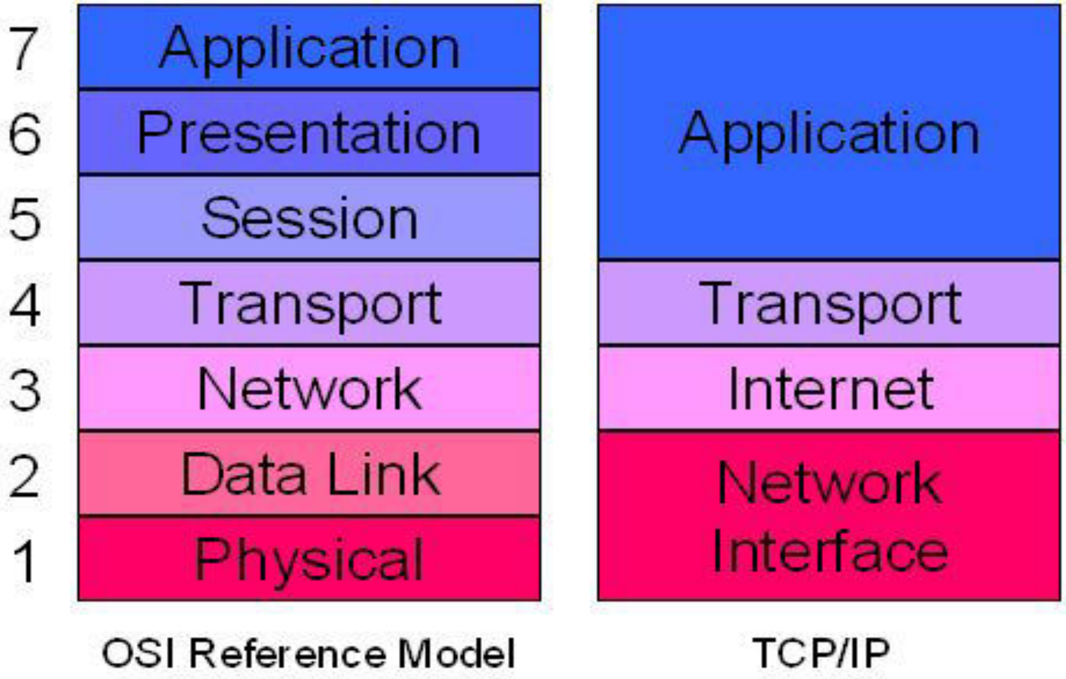
- The communication is said to occur between the same layers at both ends. This connection is said to be the logical connection.

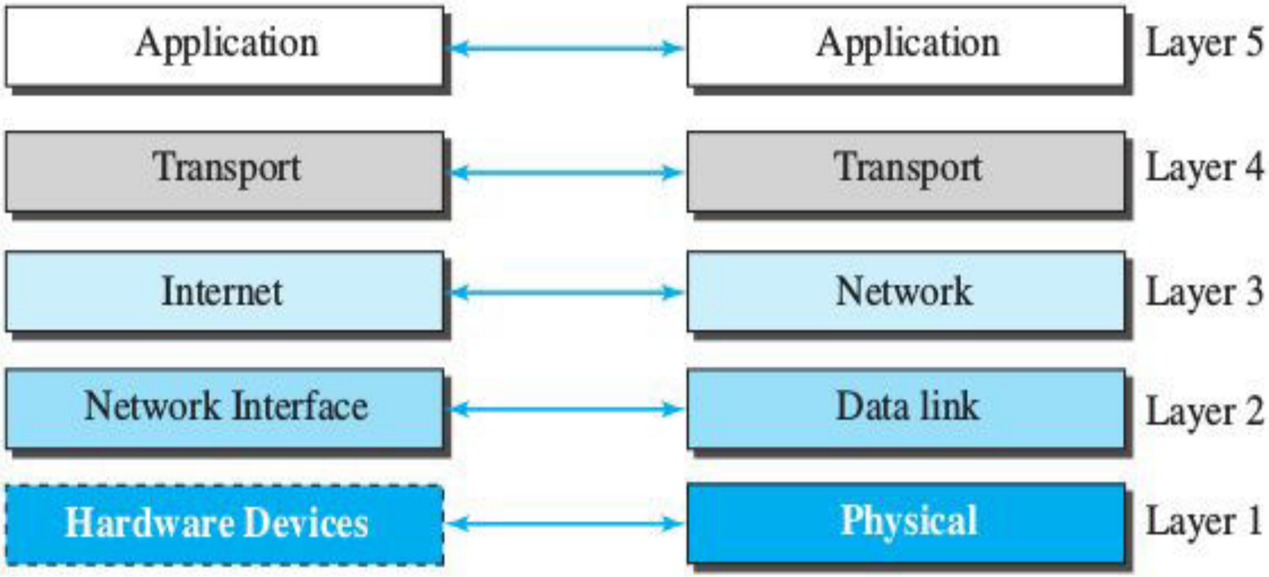


TCP/IP PROTOCOL SUITE

- TCP/IP: Transmission Control Protocol/Internet Protocol
- TCP/IP is a protocol suite (a set of protocols organized in different layers) used in the Internet today.
- It is a hierarchical protocol made up of interactive modules, each of which provides a specific functionality.
- Each upper level protocol is supported by the services provided by one or more lower level protocols.
- The original TCP/IP protocol suite was defined as four software layers built upon the hardware. Today, however, TCP/IP is thought of as a five-layer model.

TCP/IP PROTOCOL SUITE: Layered Architecture

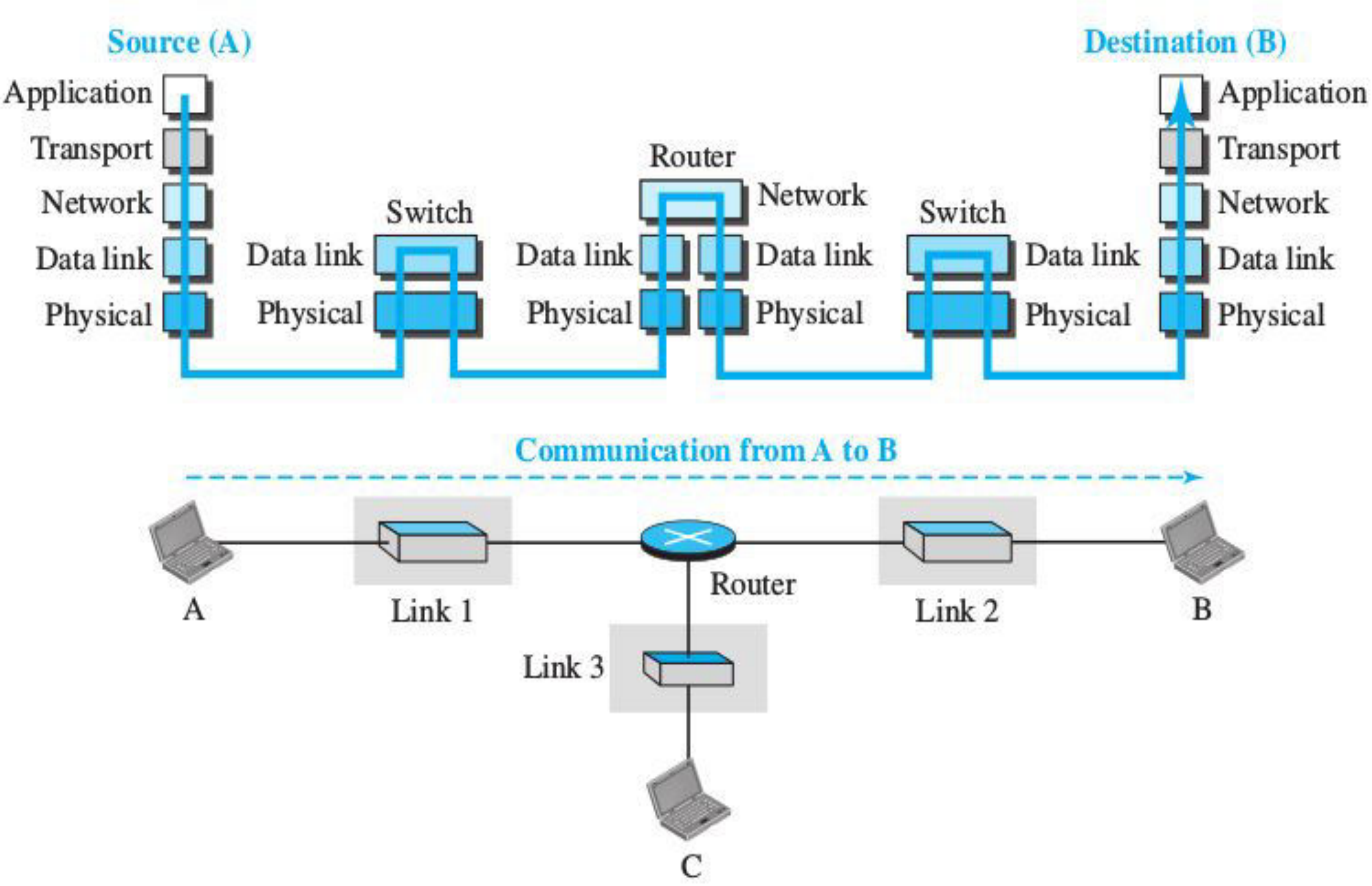




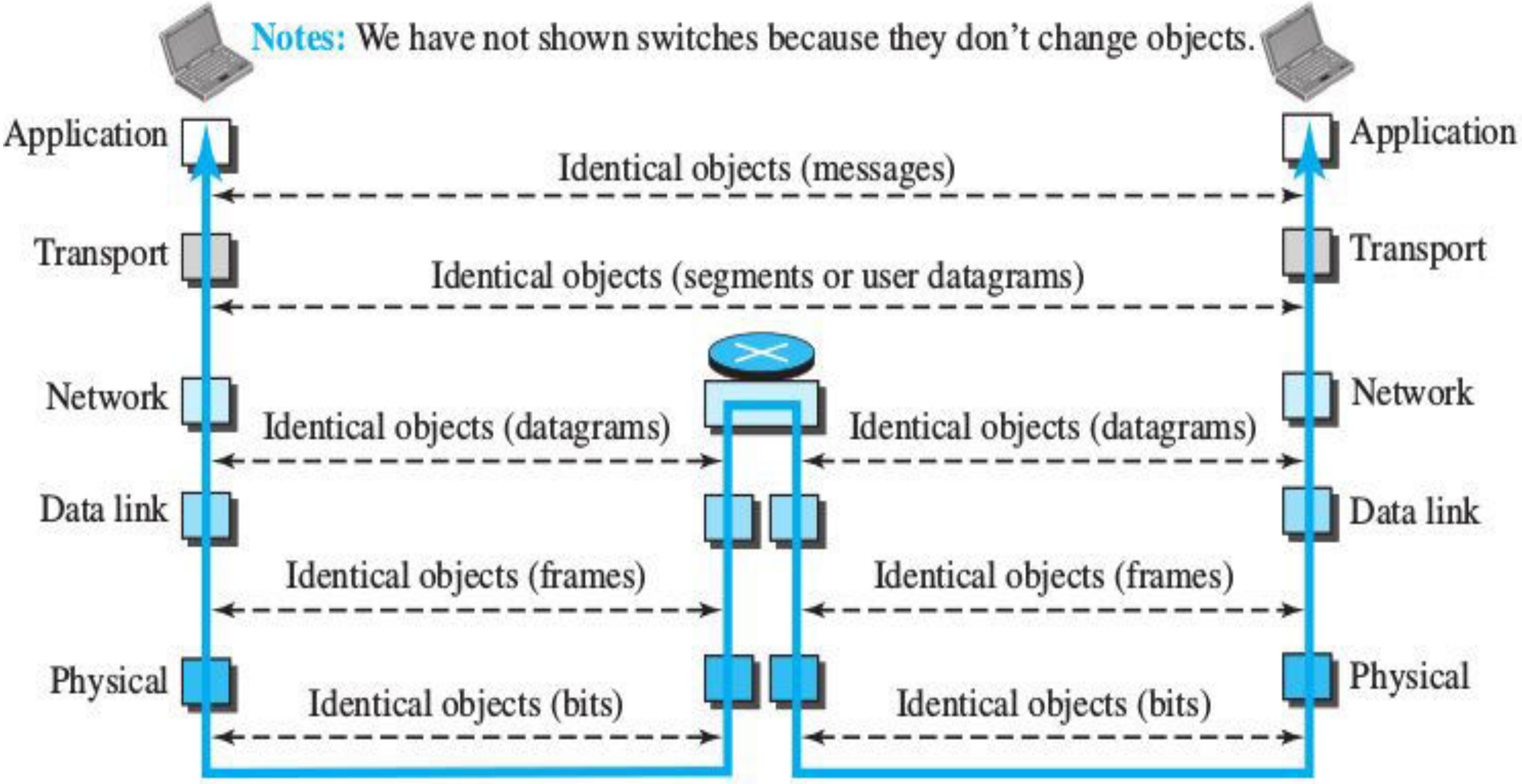
a. Original layers

b. Layers used in this book

TCP/IP PROTOCOL SUITE: Layered Architecture

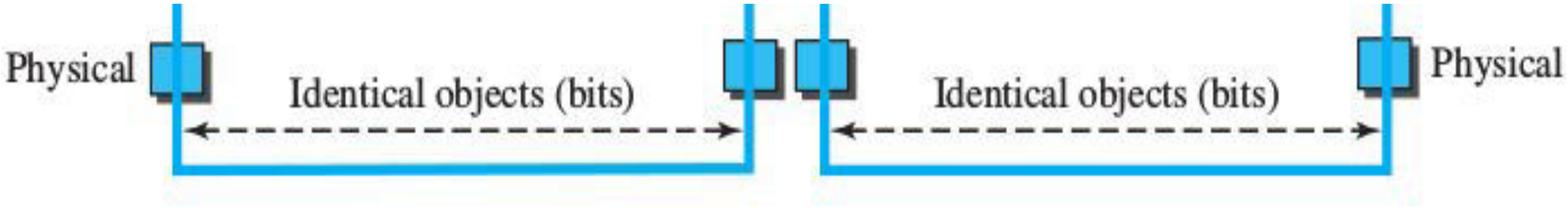


TCP/IP PROTOCOL SUITE: Logical Connections


Notes: We have not shown switches because they don't change objects.
Application
Application
Identical objects (messages)
Transport
Transport
Identical objects (segments or user datagrams)
Network
Network
Identical objects (datagrams)
Identical objects (datagrams)
Data link
Data link
Identical objects (frames)
Identical objects (frames)
Physical
Physical
Identical objects (bits)
Identical objects (bits)

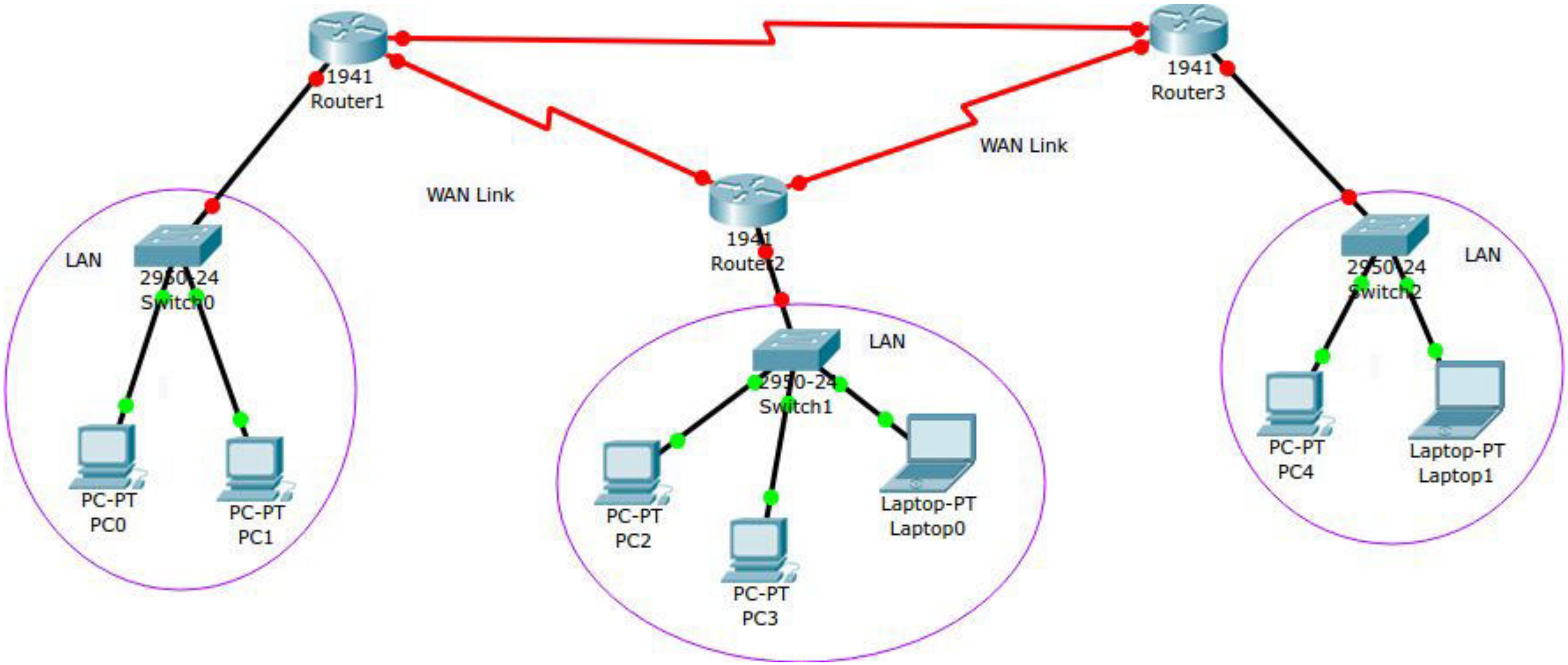
TCP/IP: Description of Each Layer

- Layer 1: Physical Layer
  - Carry individual bits across the link through wired or wireless media (cable or air).
  - The logical unit between two physical layers in two devices is a bit.



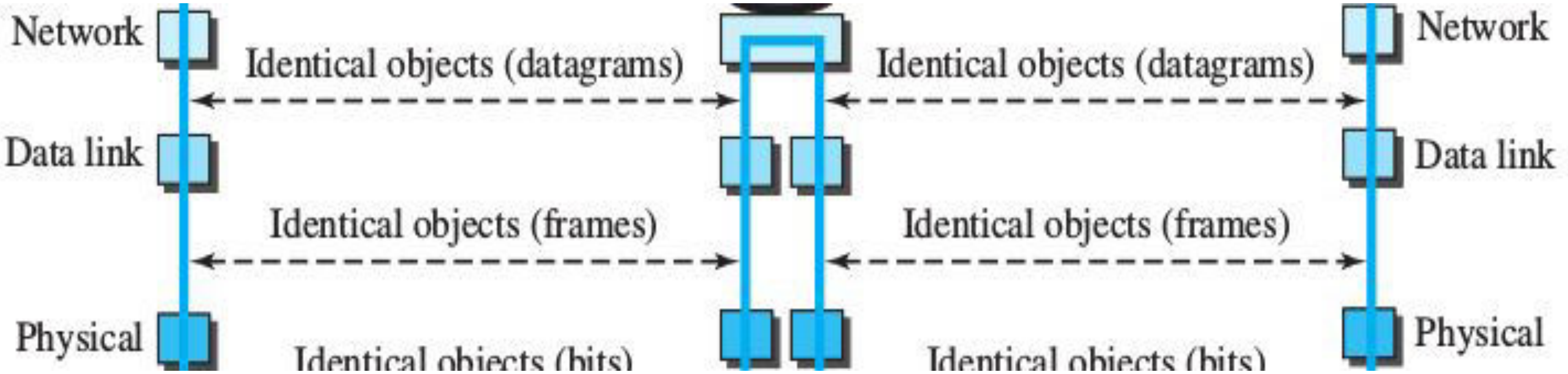
TCP/IP: Description of Each Layer

- Layer 2: Data-link Layer (DLL or Network Interface Layer)
  - An internet is made up of several links (LANs and WANs) connected by routers.
  - For a packet to to travel from a source to destination, the router chooses the best path.
  - The DLL is responsible for the travel of packets across the links.



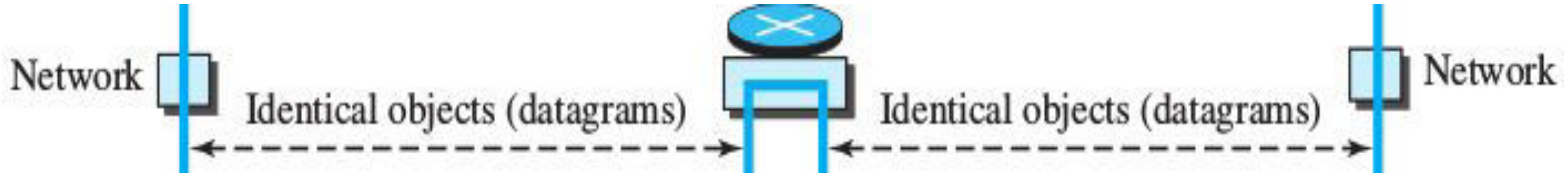
TCP/IP: Description of Each Layer

- Layer 2: Data-link Layer (DLL or Network Interface Layer)...
  - There is no specific protocol in DLL, but supports many standard and proprietary protocols.
  - Protocols vary according to the physical channel (wired / wireless, LAN / WAN).
  - Each link-layer protocol may provide a different service.
  - Some link-layer protocols provide complete error detection and correction, some provide only error correction.
  - The DLL converts the data packets into **frames.**
  - Ethernet and Token Ring are working in this layer.


Network
Identical objects (datagrams)
Identical objects (datagrams)
Network
Data link
Data link
Identical objects (frames)
Identical objects (frames)
Physical
Identical objects (bits)
Identical objects (bits)
Physical

TCP/IP: Description of Each Layer

- Layer 3: Network Layer (Internet Layer)
  - Responsible for creating a connection between the source computer and the destination computer (host-to-host communication).
  - Network layer in the routers does the routing of data packets.
  - The major protocol in this layer is **Internet Protocol (IP)**.
  - The data packet in this layer is called **datagram**.
  - IP defines the format and structure of datagrams.
  - Routing is done by IP.
  - IP is a **connectionless** protocol that provides *no flow control, no error control,* and *no congestion control* services.



TCP/IP: Description of Each Layer

- Layer 3: Network Layer (Internet Layer)...
  - The network layer also includes unicast (one-to-one) and multicast (one-to-many) routing protocols.
  - Some auxiliary protocols that help IP are ICMP(Internet Control Message Protocol), IGMP (Internet Group Management Protocol), DHCP (Dynamic Host Configuration Protocol) and ARP (Address Resolution Protocol).

TCP/IP: Description of Each Layer

- Layer 4: Transport Layer
  - Also makes an end-to-end connection, not between systems but between ports of the source and destination systems.
  - Two major protocols are TCP and UDP.
  - TCP (Transmission Control Protocol) is a connection oriented protocol, ie, it first establishes a logical connection between transport layers at two hosts, then transfer data and at last breaks the connection.
  - TCP provides flow control, error control and congestion control, hence high overhead.
  - Because of reliability, TCP is used in most applications which includes accuracy than speed.
  - The TCP data packets are called **segments**.

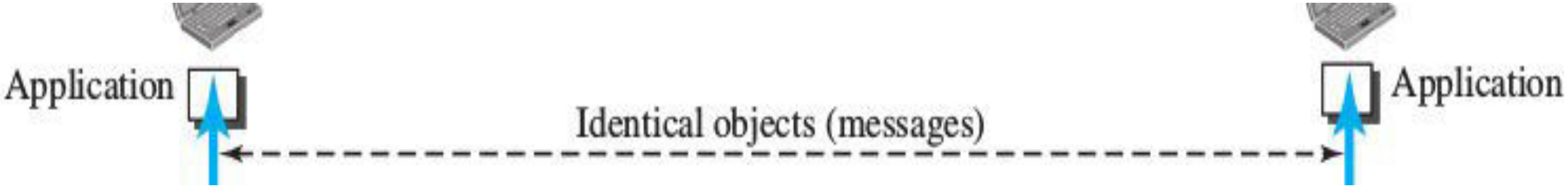
TCP/IP: Description of Each Layer

- Layer 4: Transport Layer
  - UDP (User Datagram Protocol), is a connectionless protocol that transmits user datagrams without first creating a logical connection.
  - The UDP data packets are called user datagrams.
  - Each user datagram is an independent entity without being related to the previous or the next one.
  - UDP is a simple protocol that does not provide flow, error, or congestion control.
  - Hence very less overhead.
  - Used in applications where speed is more important than accuracy.

  - A new protocol, **Stream Control Transmission Protocol (SCTP)** is designed to respond to new applications that are emerging in the multimedia.

TCP/IP: Description of Each Layer

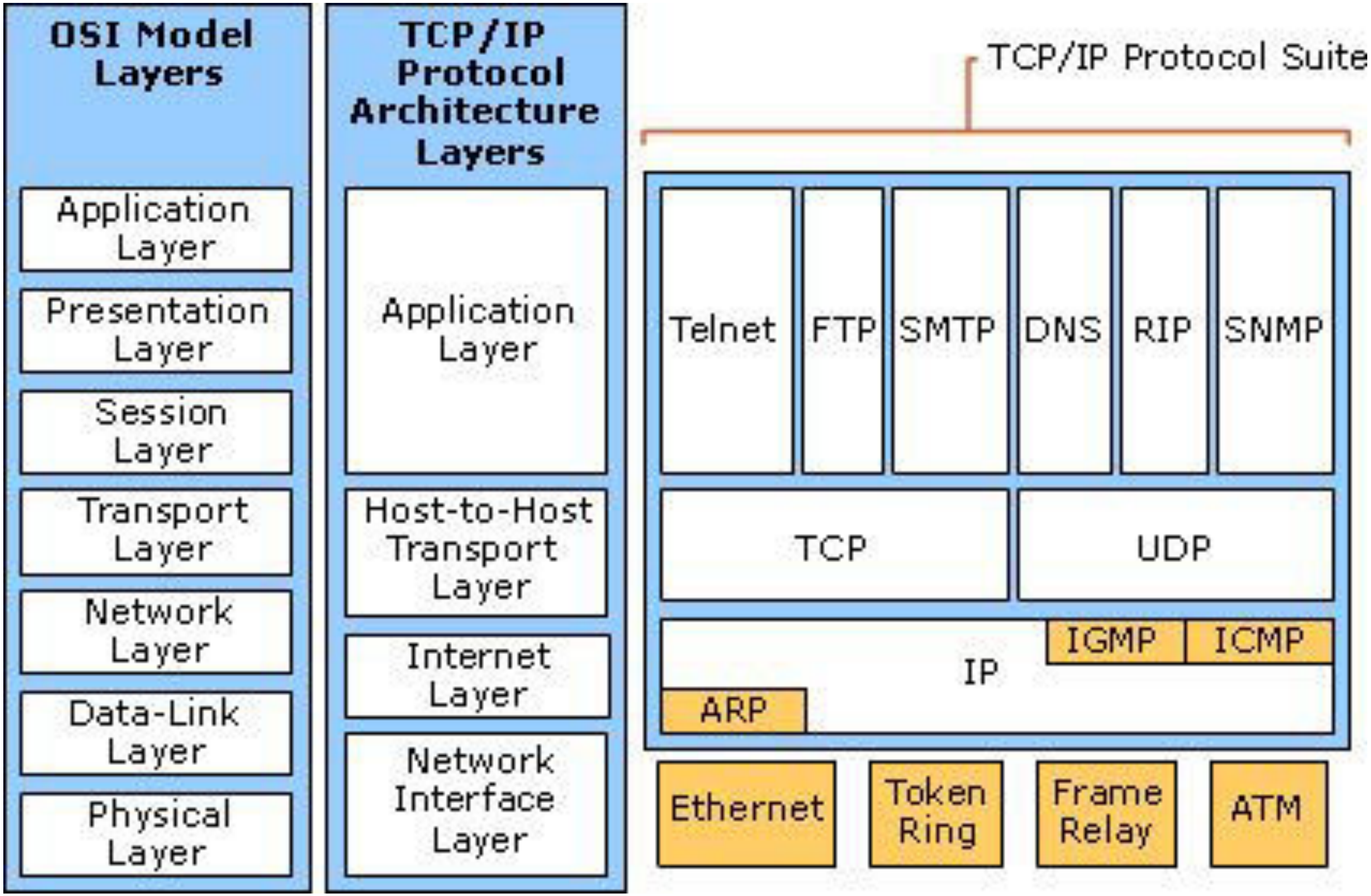
- Layer 5: Application Layer
  - The data is transferred in the form of **messages**.
  - The two application layers (source and destination) exchange messages between each other as though there were a bridge between the two layers.
  - Communication at the application layer is between two **processes** (two programs running at this layer, eg: browser).
  - Thus **process-to-process** communication is the duty of the application layer.



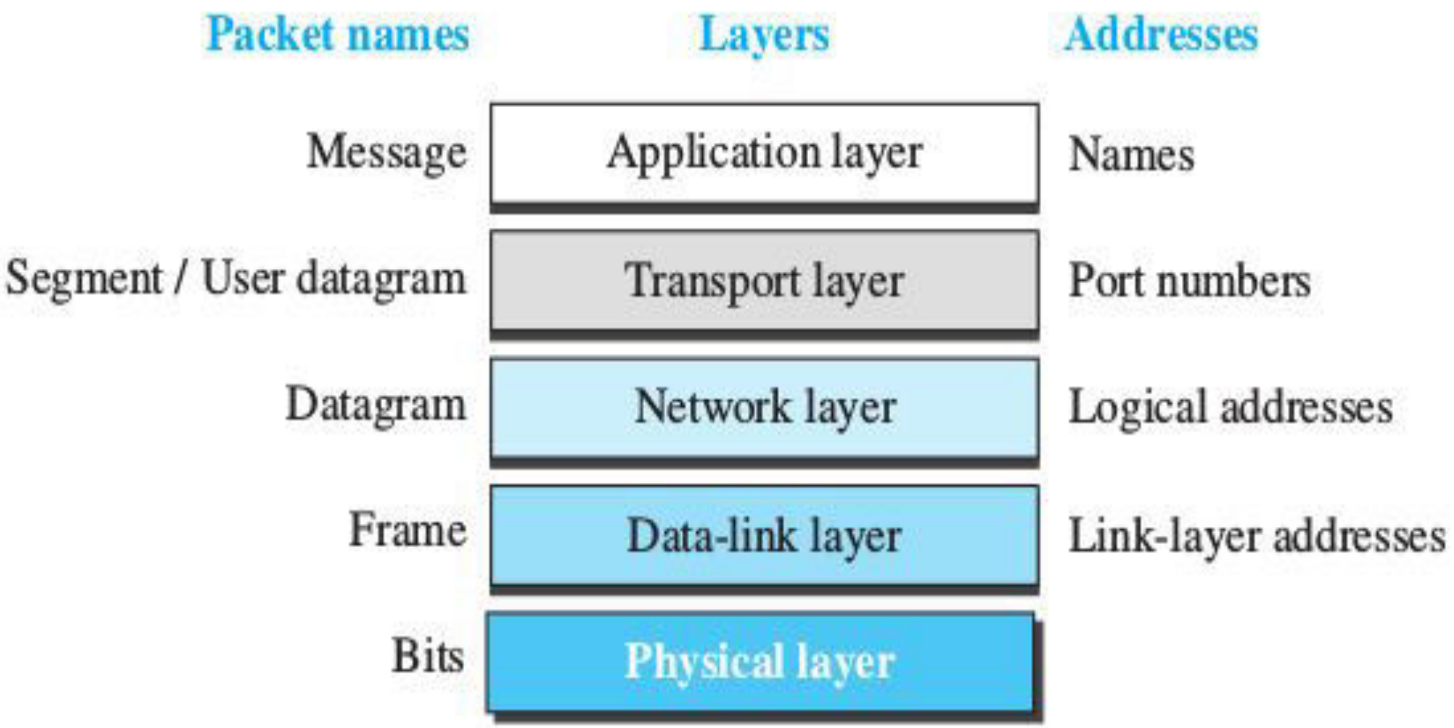
TCP/IP: Description of Each Layer

- Layer 5: Application Layer...
- Common protocols are;
  - The Hypertext Transfer Protocol (HTTP) is a vehicle for accessing the World Wide Web (WWW).
  - The Simple Mail Transfer Protocol (SMTP) is the main protocol used in electronic mail (e-mail) service.
  - The File Transfer Protocol (FTP) is used for transferring files from one host to another.
  - The Terminal Network (TELNET) and Secure Shell (SSH) are used for accessing a site remotely.
  - The Simple Network Management Protocol (SNMP) is used by an administrator to manage the Internet at global and local levels.
  - The Domain Name System (DNS) is used by other protocols to find the network-layer address of a computer.
  - The Internet Group Management Protocol (IGMP) is used to collect membership in a group.

TCP/IP: Description of Each Layer

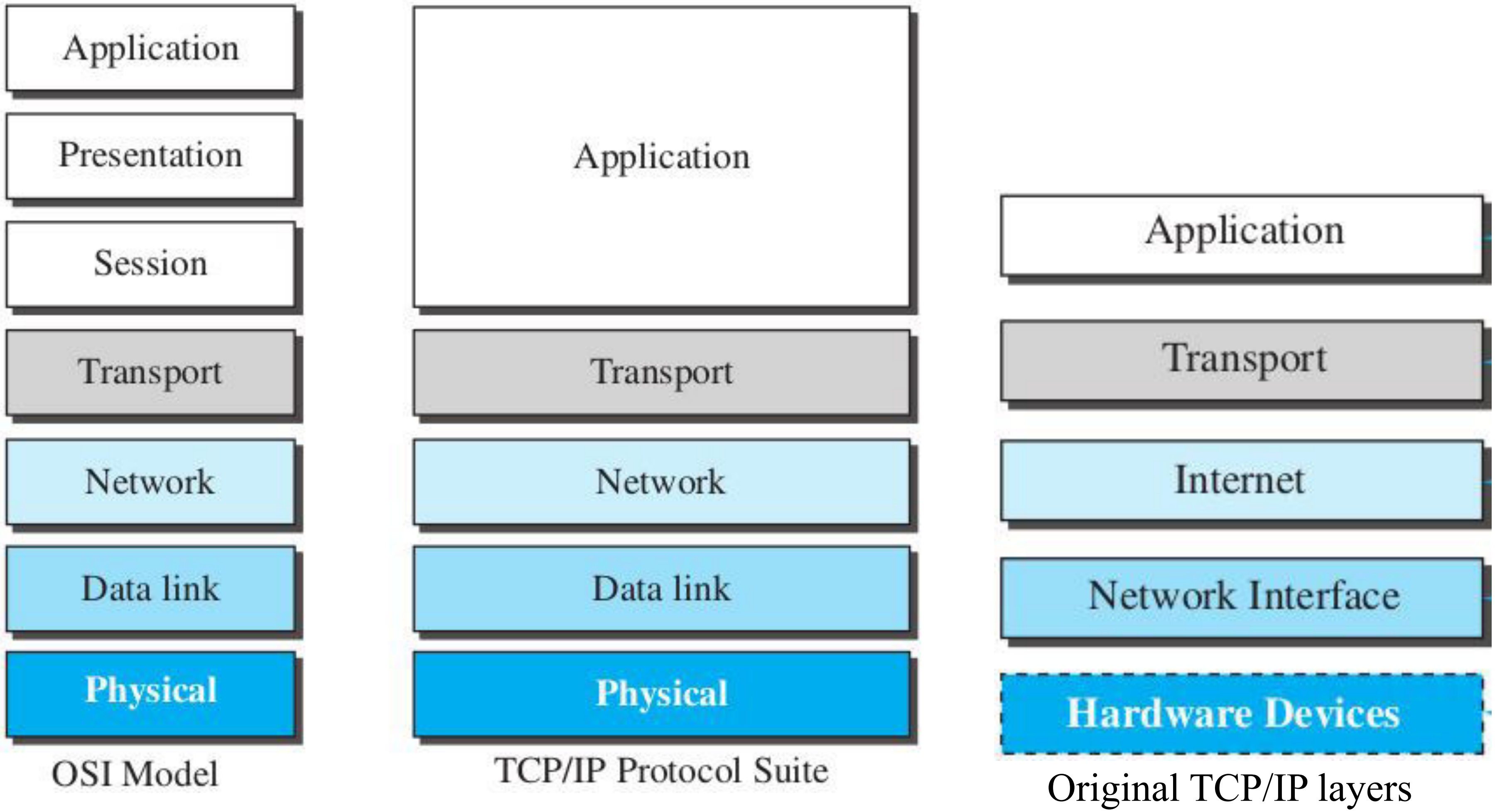


TCP/IP: Addressing



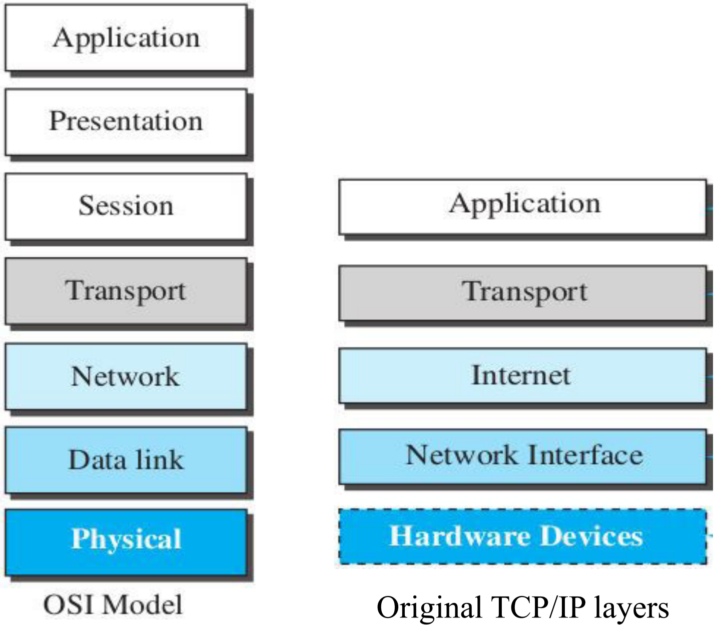
Packet names	Layers	Addresses	Example
Message	Application layer	Names	(google.com)
Segment / User datagram	Transport layer	Port numbers	(Port 80 for http)
Datagram	Network layer	Logical addresses	(IP: 192.168.1.5)
Frame	Data-link layer	Link-layer addresses	(MAC: 4a:12:c6:84:70:9b)
Bits	Physical layer		(bits only)

OSI vs TCP/IP


Application
Presentation
Session
Transport
Network
Data link
Physical
OSI Model
Application
Transport
Network
Data link
Physical
TCP/IP Protocol Suite
Application
Transport
Internet
Network Interface
Hardware Devices
Original TCP/IP layers

OSI vs TCP/IP

1. Number of layers is 7 in ISO-OSI but 4 in TCP/IP.
2. OSI has a physical layer but TCP/IP is created above the hardware.
3. In the OSI model, the top three layers are separate, but in TCP/IP these three layers are combined into one application layer.
4. TCP/IP has more than one transport-layer protocol. Some of the functionalities of the session layer are available in some of the transport-layer protocols.
5. The functionalities of the session and presentation layers are included in the applications/softwares in TCP/IP.



End of Part 1 of Module 1

Thank you

Diploma in Computer Hardware Engineering

COMPUTER NETWORKS

(5151 Rev 2015)

Module 1 - Part 2 of 3

Presenter: Sreejesh NG
Lecturer in Computer Hardware Engineering
Government Polytechnic College, Cherthala

Ref: Data Communications and Networking, 5E by Forouzan

Course General Outcomes:

Sl.	G.O	On completion of this course the student will be able :
1	1	To Understand the concept of TCP/IP Protocol
2	1	To Understand the concept of Network Layer
3	1	To Understand the concept of Transport Layer
4	1	To Understand the concept of Application Layer

Specific Outcomes:

MODULE I. REVIEW OF NETWORK MODELS

1.1 Understand TCP/IP Protocol

- 1.1.1 Illustrate computer networks
- 1.1.2 Identify TCP/IP Protocol suite.
- 1.1.3 Explain the functionalities of layers in TCP/IP
- 1.1.4 Define Addressing of TCP/IP.
- 1.1.5 Describe about Wired LAN – Ethernet
- 1.1.6 State IEEE 802 project
- 1.1.7 Illustrate standard Ethernet
- 1.1.8 Describe about Wireless LAN.
- 1.1.9 State IEEE 802.11
- 1.1.10 Explain LAN connecting devices.
- 1.1.11 Explain the architecture of Virtual LANs.

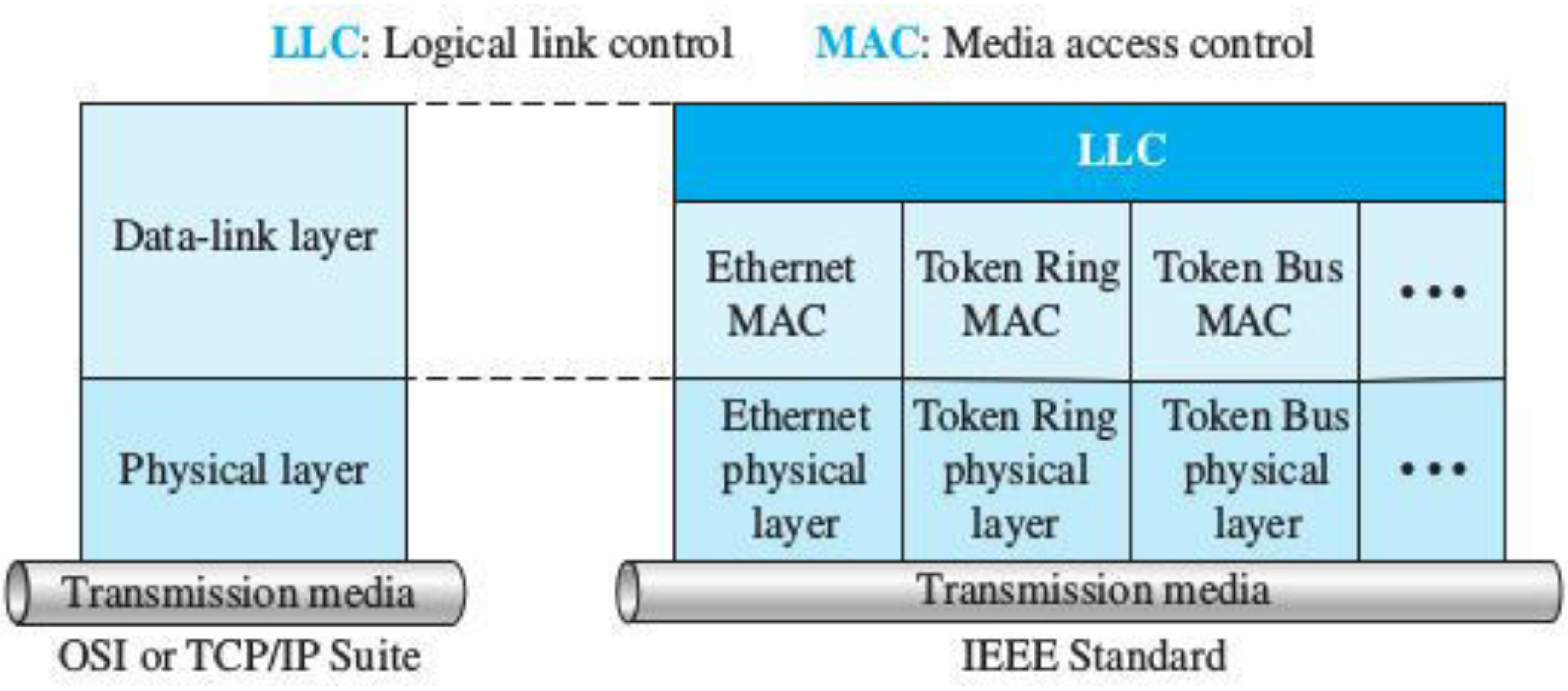
MODULE 1 - TCP/IP PROTOCOL

- Introduction to computer networks - physical structure, topology, types.
- TCP/IP - architecture, Description of layers, addressing.
- Wired LAN - Ethernet protocol - IEEE project 802 - Standard Ethernet - characteristics, addressing, implementation.
- Wireless LAN - architectural comparison, characteristics, access control - IEEE 802.11 - architecture.
- LAN connecting devices - hub, switch, router.
- Virtual LAN - architecture, membership, configuration.

Ref: Data Communications and Networking - Behrouz A. Forouzan -
McGraw Hill Edn.-Fourth Edition/Fifth Edition

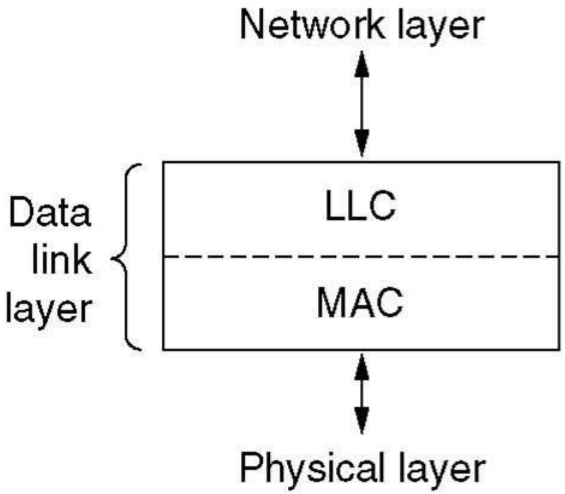
Wired LAN: IEEE 802

- The IEEE created a Project 802 to set standards to enable intercommunication among equipment from a variety of manufacturers.
- Specifies the functions of Physical layer and Data Link layer of LAN protocols (such as Ethernet, Token Ring, Token Bus, etc).
- The IEEE has subdivided the data-link layer into two sublayers: **Logical Link Control (LLC)** and **Media Access Control (MAC)** sublayers.

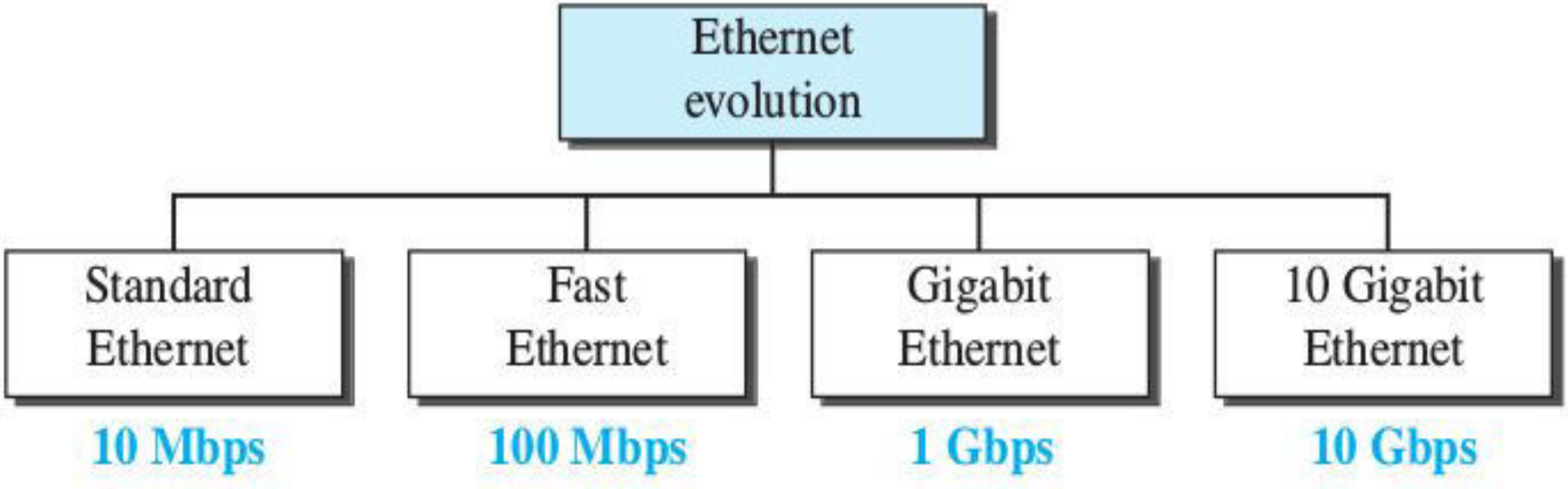


Wired LAN: IEEE 802

- **Logical Link Control (LLC)** sublayer:
  - Does Flow control, error control, and part of the framing.
  - Provides a single link-layer control protocol for all IEEE LANs.
  - Provide interconnectivity between different LANs because it makes the MAC sublayer transparent.
- **Media Access Control (MAC)** sublayer:
  - Specific access method for each LAN.
    - Eg: CSMA/CD for Ethernet, token-passing for Token Ring.
  - Does part of the framing.



Wired LAN: Ethernet



Standard Ethernet Ethernet type with the data rate of 10 Mbps.

Characteristics: Connectionless and Unreliable

- Connectionless service:
  - No connection establishment or connection release for frame transfer.
  - Each frame sent is independent of the previous or next frame.
  - The sender sends a frame whenever it has it; the receiver *may or may not* be ready for it.
  - The sender *may* overwhelm the receiver with frames, which *may* result in dropping frames.
  - If a frame drops or if the frame has errors, the sender will not know about it.
  - Ethernet is using IP at the network layer which is also connectionless.
  - If the transport layer protocol is UDP, the frame lose will be suffered by the Application Layer.
  - If the transport layer protocol is TCP, the sender TCP does not receive acknowledgment for its segment and sends it again.

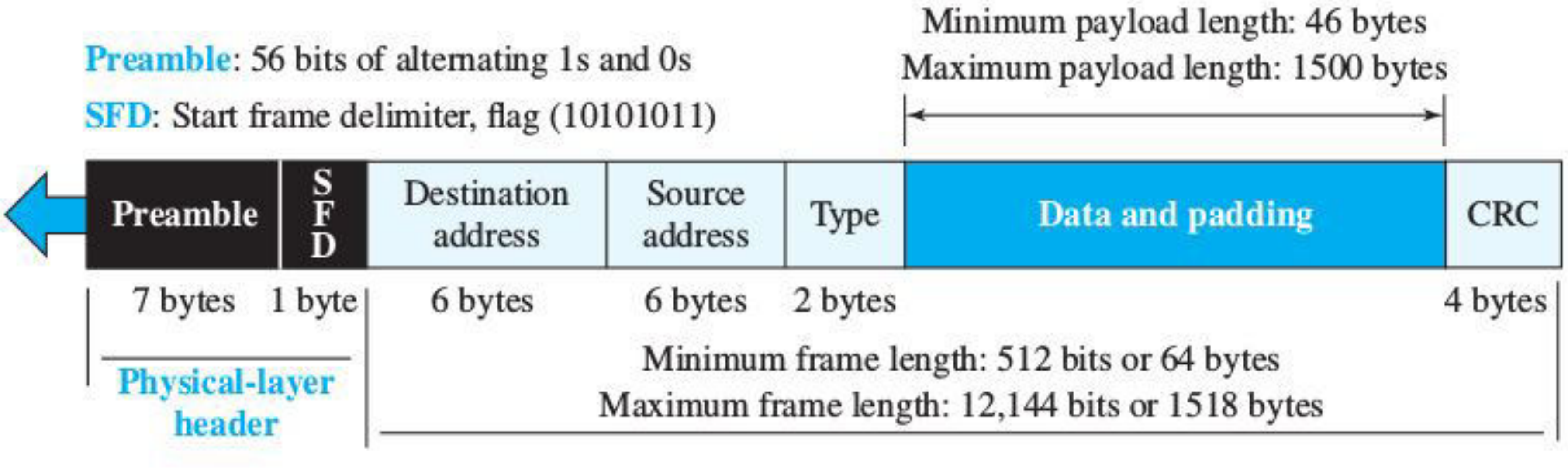
Standard Ethernet

Characteristics...

- Unreliable Service:
  - Ethernet is also unreliable like IP and UDP.
  - If a frame is corrupted during transmission and the receiver finds out about the corruption, the receiver drops the frame silently. It is the duty of high-level protocols to find out about it.
  - The receiver can find out the occurrence of corruption with the help of CRC-32 field in the frame.

Standard Ethernet: Frame Format

- **Preamble:**
  - 7 bytes (56 bits), alternate 1s and 0s (ie, **101010101010…..**)
  - To indicate that a frame is coming.
  - Enables the receiver to synchronize its clock if it's out of synchronization.
  - Not the part of frame, but added by the physical layer.



Preamble: 56 bits of alternating 1s and 0s

SFD: Start frame delimiter, flag (10101011)

Minimum payload length: 46 bytes

Maximum payload length: 1500 bytes

Preamble	SFD	Destination address	Source address	Type	Data and padding	CRC
7 bytes	1 byte	6 bytes	6 bytes	2 bytes		4 bytes

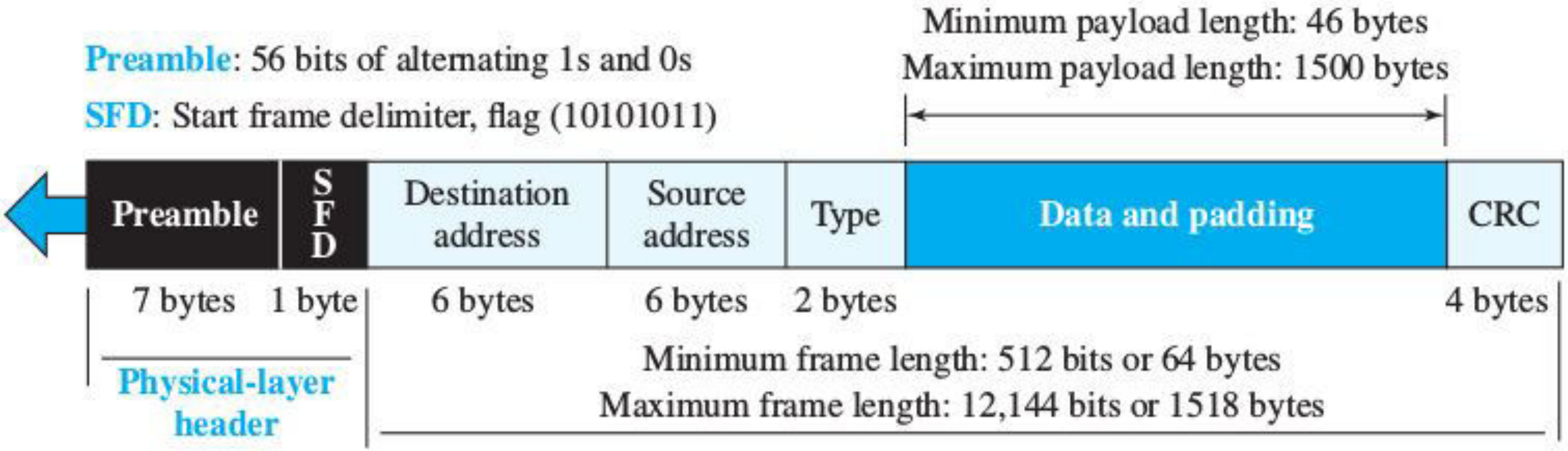
Physical-layer header

Minimum frame length: 512 bits or 64 bytes

Maximum frame length: 12,144 bits or 1518 bytes

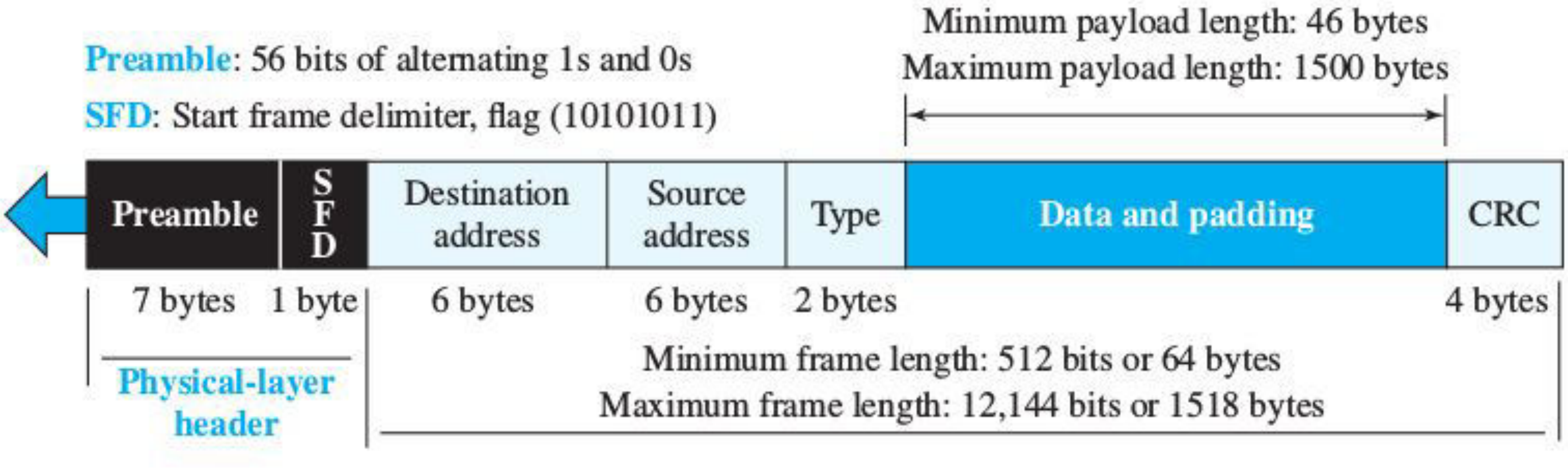
Standard Ethernet: Frame Format

- **Start frame delimiter (SFD):**
  - One byte. It is **10101011**.
  - Last chance of synchronization.
  - Indicate that the next field is destination address.
  - Indicate that the the frame is starting.
  - Added by the physical layer.


Preamble: 56 bits of alternating 1s and 0s
SFD: Start frame delimiter, flag (10101011)
Minimum payload length: 46 bytes
Maximum payload length: 1500 bytes
Preamble
S F D
Destination address
Source address
Type
Data and padding
CRC
7 bytes
1 byte
6 bytes
6 bytes
2 bytes
4 bytes
Physical-layer header
Minimum frame length: 512 bits or 64 bytes
Maximum frame length: 12,144 bits or 1518 bytes

Standard Ethernet: Frame Format

- **Destination address (DA):**
    - Six byte (48 bits) Physical Address (MAC Address or link-layer address or hardware address) of the destination host.
    - When the receiver sees its own link-layer address, or a multicast address for a group that the receiver is a member of, or a broadcast address, it decapsulates the data from the frame and passes the data to the upper-layer.



Preamble: 56 bits of alternating 1s and 0s

SFD: Start frame delimiter, flag (10101011)

Minimum payload length: 46 bytes
Maximum payload length: 1500 bytes

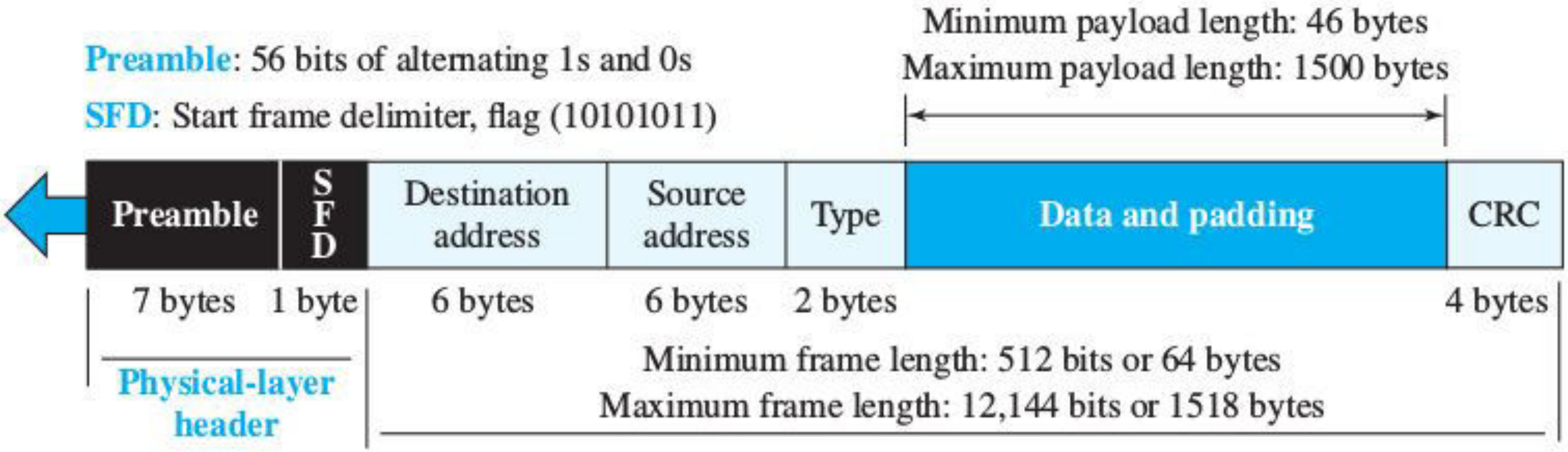
Preamble	SFD	Destination address	Source address	Type	Data and padding	CRC
7 bytes	1 byte	6 bytes	6 bytes	2 bytes		4 bytes

Physical-layer header

Minimum frame length: 512 bits or 64 bytes
Maximum frame length: 12,144 bits or 1518 bytes

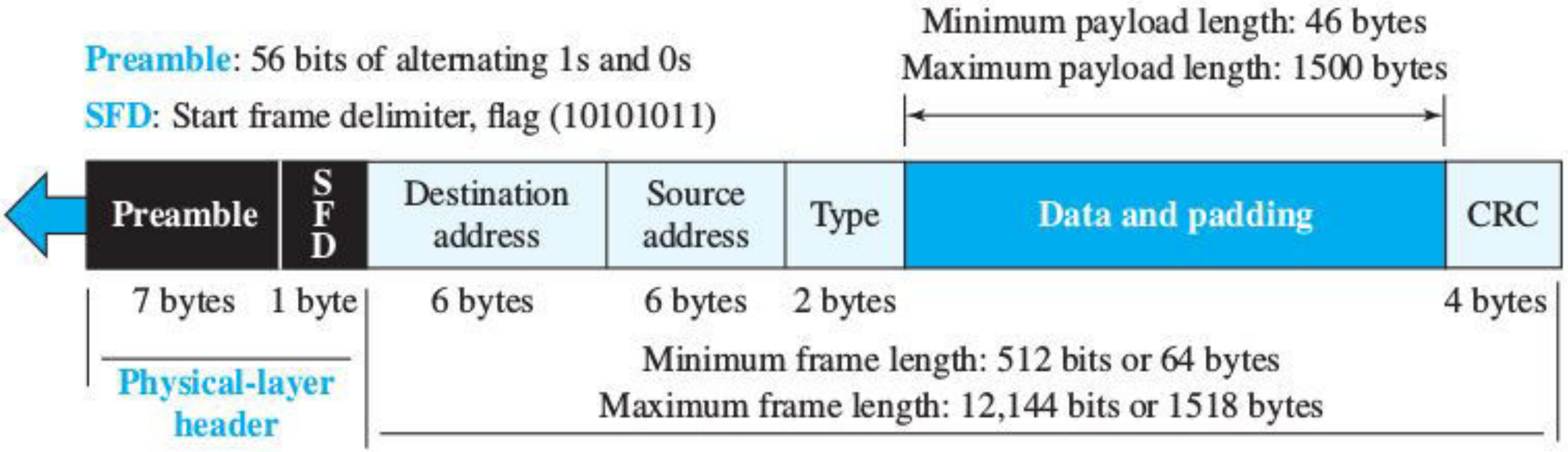
Standard Ethernet: Frame Format

- **Source address (SA).**
  - This field is also six bytes and contains the link-layer address of the sender of the packet.


Preamble: 56 bits of alternating 1s and 0s
SFD: Start frame delimiter, flag (10101011)
Minimum payload length: 46 bytes
Maximum payload length: 1500 bytes
Preamble
SFD
Destination address
Source address
Type
Data and padding
CRC
7 bytes
1 byte
6 bytes
6 bytes
2 bytes
4 bytes
Physical-layer header
Minimum frame length: 512 bits or 64 bytes
Maximum frame length: 12,144 bits or 1518 bytes

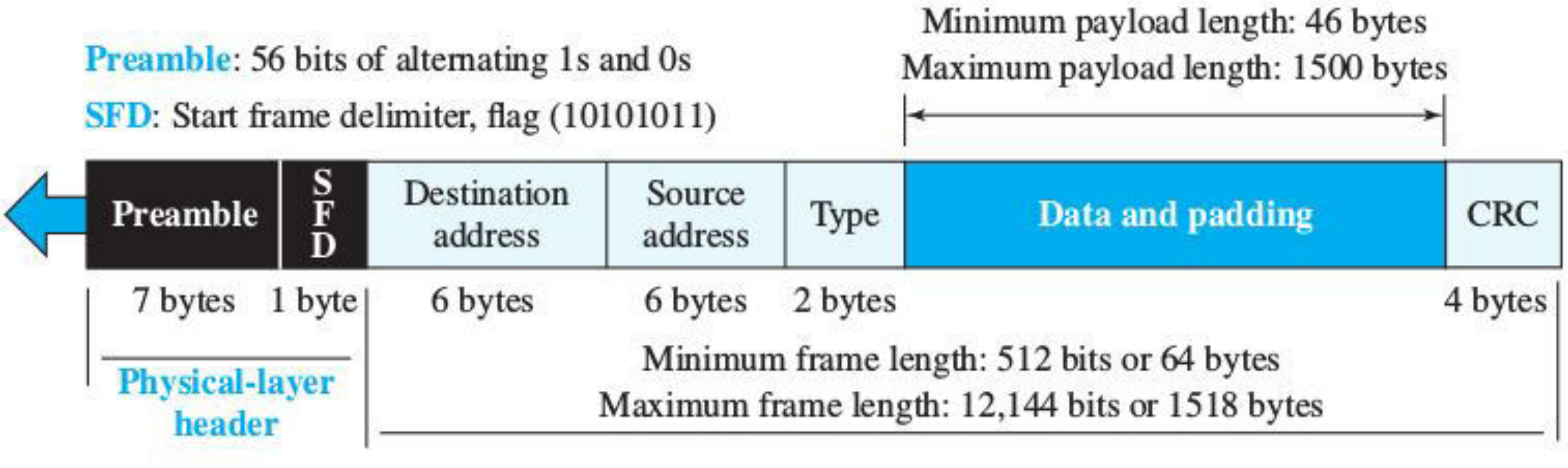
Standard Ethernet: Frame Format

- **Type:**
  - Two bytes.
  - This field defines the upper-layer protocol whose packet is encapsulated in the frame.
  - This protocol can be IP, ARP, OSPF, and so on.
  - It is used for multiplexing and demultiplexing.
  - Now, this field has two purposes. Values of 1500 and below mean that it is used to indicate the size of the payload, while values above indicate that it is used as an Type, to indicate which protocol is encapsulated in the payload of the frame.


Preamble: 56 bits of alternating 1s and 0s
SFD: Start frame delimiter, flag (10101011)
Minimum payload length: 46 bytes
Maximum payload length: 1500 bytes
Preamble
SFD
Destination address
Source address
Type
Data and padding
CRC
7 bytes
1 byte
6 bytes
6 bytes
2 bytes
4 bytes
Physical-layer header
Minimum frame length: 512 bits or 64 bytes
Maximum frame length: 12,144 bits or 1518 bytes

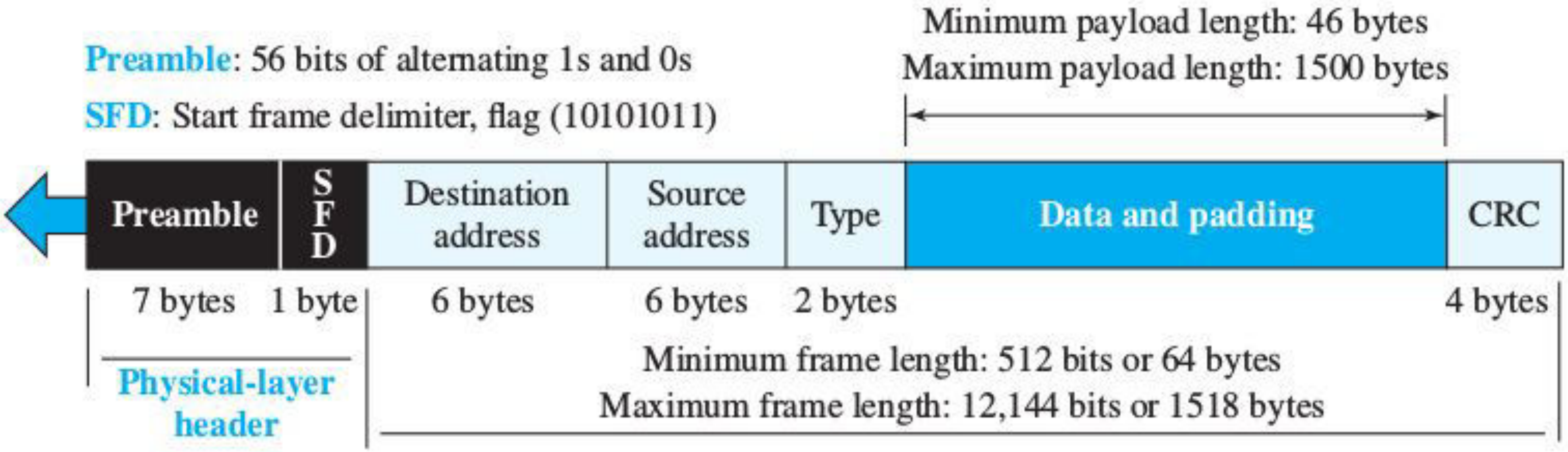
Standard Ethernet: Frame Format

- **Data:**
  - This field carries data encapsulated from the upper-layer protocols.
  - Minimum of 46 bytes and a maximum of 1500 bytes.
  - If the data coming from the upper layer is more than 1500 bytes, it should be fragmented and encapsulated in more than one frame.
  - If it is less than 46 bytes, it is to be padded with extra zeros to make it 46.
  - A padded data frame is delivered to the upper-layer protocol as it is (without removing the padding), which means that it is the responsibility of the upper layer to remove or to add the padding.

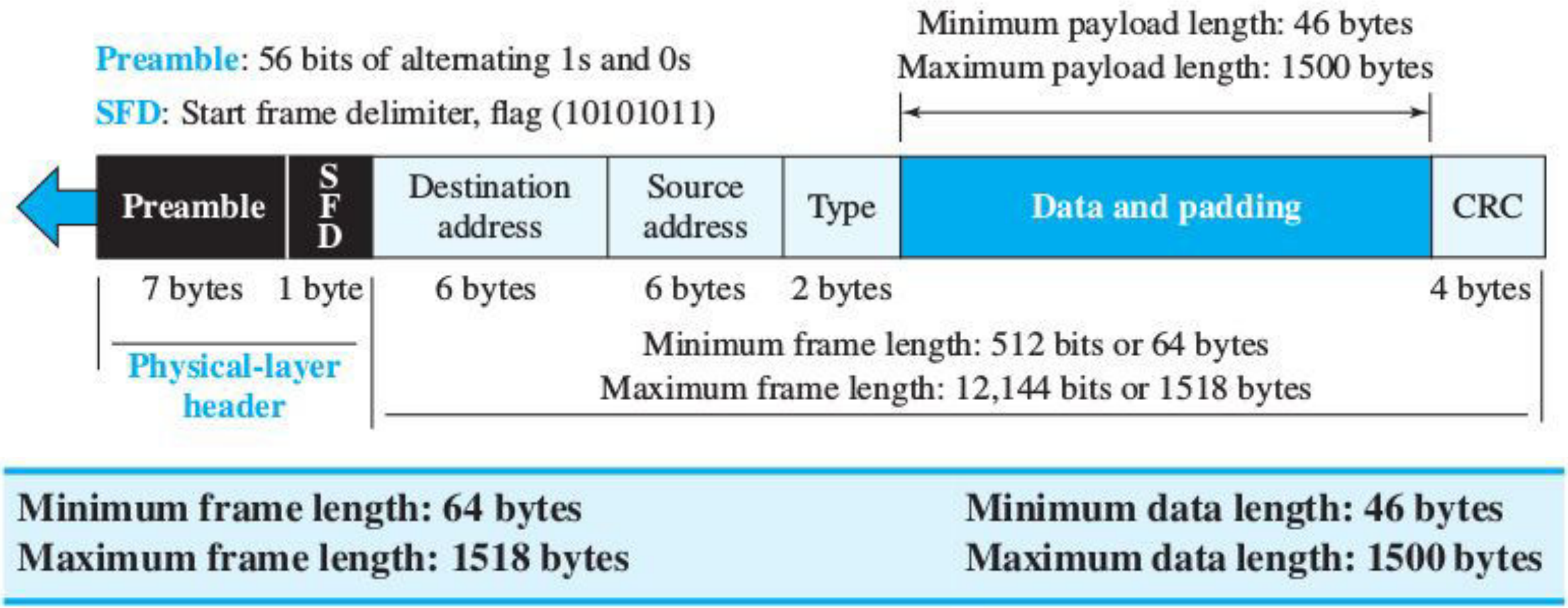

Preamble: 56 bits of alternating 1s and 0s
SFD: Start frame delimiter, flag (10101011)
Minimum payload length: 46 bytes
Maximum payload length: 1500 bytes
Preamble
SFD
Destination address
Source address
Type
Data and padding
CRC
7 bytes
1 byte
6 bytes
6 bytes
2 bytes
4 bytes
Physical-layer header
Minimum frame length: 512 bits or 64 bytes
Maximum frame length: 12,144 bits or 1518 bytes

Standard Ethernet: Frame Format

- **CRC (Cyclic Redundancy Check):**
  - 4 bytes (32 bits).
  - Error detection information, in this case a CRC-32.
  - CRC is calculated over the addresses, types, and data field.
  - If the receiver calculates the CRC and finds that it is not zero (corruption in transmission), it discards the frame.


Preamble: 56 bits of alternating 1s and 0s
SFD: Start frame delimiter, flag (10101011)
Minimum payload length: 46 bytes
Maximum payload length: 1500 bytes
Preamble
SFD
Destination address
Source address
Type
Data and padding
CRC
7 bytes
1 byte
6 bytes
6 bytes
2 bytes
4 bytes
Physical-layer header
Minimum frame length: 512 bits or 64 bytes
Maximum frame length: 12,144 bits or 1518 bytes

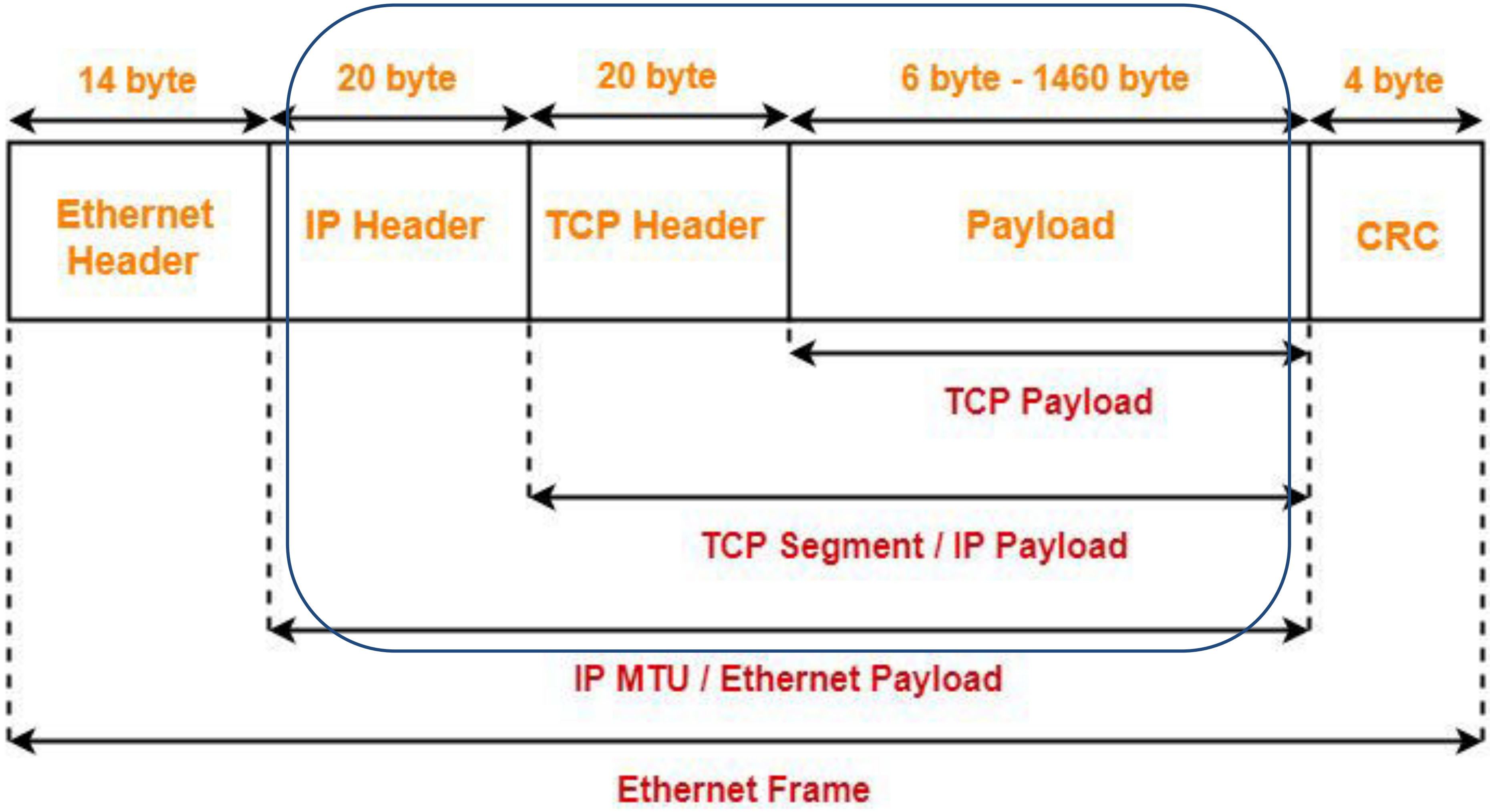
Standard Ethernet: Frame Length


Preamble: 56 bits of alternating 1s and 0s
SFD: Start frame delimiter, flag (10101011)
Minimum payload length: 46 bytes
Maximum payload length: 1500 bytes
Preamble
SFD
Destination address
Source address
Type
Data and padding
CRC
7 bytes
1 byte
6 bytes
6 bytes
2 bytes
4 bytes
Physical-layer header
Minimum frame length: 512 bits or 64 bytes
Maximum frame length: 12,144 bits or 1518 bytes
Minimum frame length: 64 bytes
Maximum frame length: 1518 bytes
Minimum data length: 46 bytes
Maximum data length: 1500 bytes

Reasons for maximum length:

1. To reduce the buffer size, as memory was expensive at that time.
2. To prevent one device from monopolizing the medium.

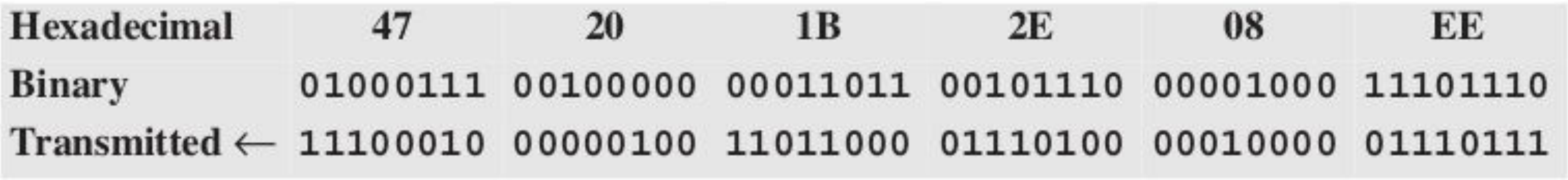
Standard Ethernet: A Frame



Minimum frame length: 64 bytes	Minimum data length: 46 bytes
Maximum frame length: 1518 bytes	Maximum data length: 1500 bytes

Standard Ethernet: Addressing

- Every NIC (Network Interface Card) has a unique 6 bytes (48bits) address.
- This address is called **Physical Address, Hardware Address** or **MAC Address**.
- It is written in hexadecimal notation, with a colon between bytes, like ***47:20:1B:2E:08:EE***.
- Ethernet uses the **MAC address** to transfer the frames from source to destination.
- The transmission is **left to right, byte by byte**; however, **for each byte, the least significant bit is sent first and the most significant bit is sent last**.



Hexadecimal	47	20	1B	2E	08	EE
Binary	01000111	00100000	00011011	00101110	00001000	11101110
Transmitted ←	11100010	00000100	11011000	01110100	00010000	01110111

Standard Ethernet: Addressing..

Unicast, Multicast, and Broadcast Addresses:

- A source address is always a unicast address.
- The destination address may be unicast, multicast, or broadcast.
- For *unicast address*, the least significant bit of the first byte will be *zero*. It will be *one* for *multicast* and *broadcast* addresses.
- By receiving the *first bit* of the frame, the destination device can understand whether the address is unicast or not.

Eg:

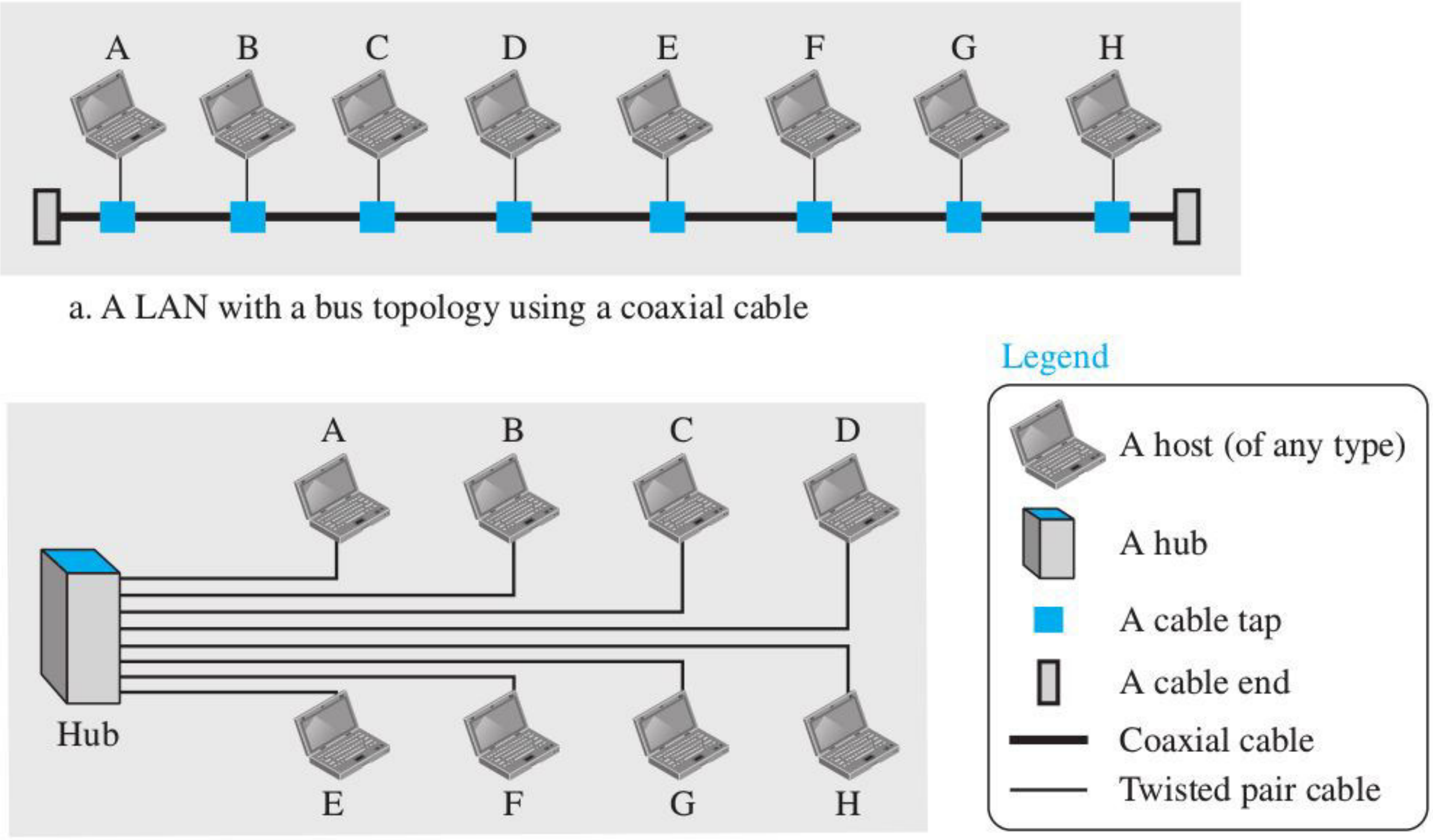
- 4A:30:10:21:10:1A - The first byte is transmitted as **0**1010010 - Unicast
- 47:20:1B:2E:08:EE - The first byte is transmitted as **1**1100010 - Multicast
- FF:FF:FF:FF:FF:FF - The first byte is transmitted as **1**1111111 - Broadcast

Reason for this

Transmission in the standard Ethernet is always broadcast, no matter if the intention is unicast, multicast, or broadcast.

Standard Ethernet: Access Method

- CSMA/CD (Carrier Sense Multiple Access/Collision Detection).
- Suppose the device A wish to send a frame to D (in the following figure)



a. A LAN with a bus topology using a coaxial cable

b. A LAN with a star topology using a hub

Standard Ethernet: Access Method - CSMA/CD

- ‘A’ senses the carrier. ie, measures the energy level of the medium.
- If no energy signal on the medium for about 100 μs, ‘A’ start sending a copy of the frame.
- If the signal energy level is not zero, ‘A’ continuously monitors the medium until it is free for about 100 μs and then it starts sending the frame.
- ‘A’ senses the carrier until 512 bits are sent. Then it discards the copy with it.
- If ‘A’ senses a collision within the 512 bits, it stops sending and raise an alarm signal of 48 bits.
- It then increment the value of K (number of attempts) and waits for some seconds (a random number between 0 and 2^k-1). Then sense the carrier again.
    - K=1 -> (0,1)
    - K=2 -> (0,1,2,3)
    - K=3 -> (0,1,2,3,4,5,6,7)
- When K reaches 15, it aborts the transmission of the frame and tries again.

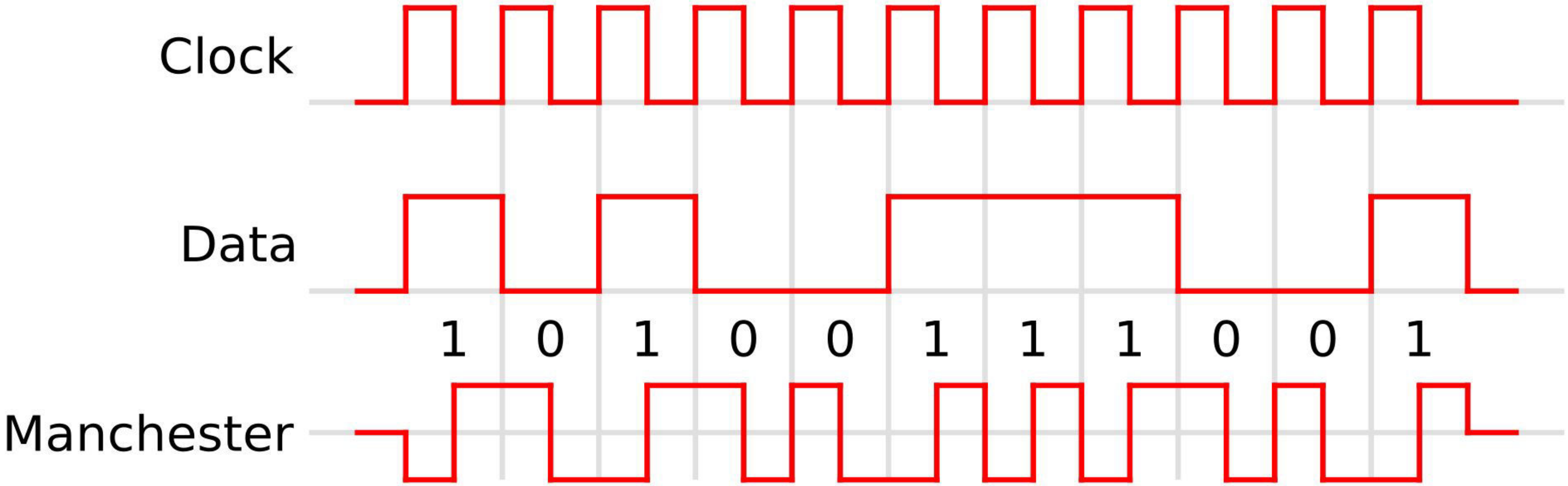
Standard Ethernet: Implementation

- Four Standard Ethernet implementations became popular during the 1980s.
    - 10Base5
    - 10Base2
    - 10Base-T
    - 10Base-F
- In the nomenclature 10BaseX,
    - 10 defines the data rate (10 Mbps)
    - The term Base means baseband (digital) signal
    - X approximately defines
        - The maximum size of the cable in 100 meters (for example 5 for 500 or 2 for 185 meters)
            - OR
        - The type of cable, T for unshielded twisted pair cable (UTP) and F for fiber-optic.
- Data are converted to a digital signal using the Manchester scheme.

Standard Ethernet: Implementation

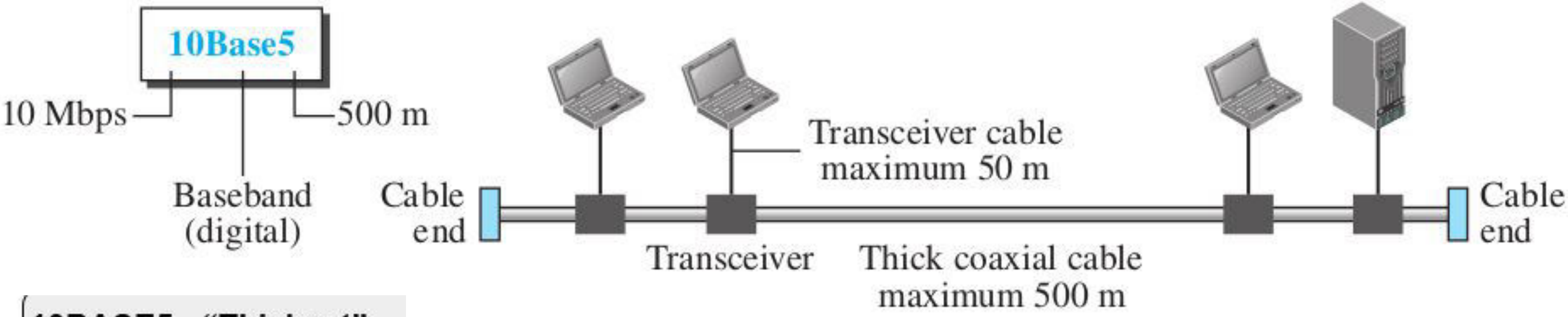
Manchester Encoding Scheme:

- Always has a transition at the middle of each bit period which indicates a data bit.
- It may have a transition at the start of the period also. Transitions at the period boundaries do not carry information.
- As per IEEE 802.3 (Ethernet) standards, a logic 0 is represented by a high-low signal sequence and a logic 1 is represented by a low-high signal sequence.



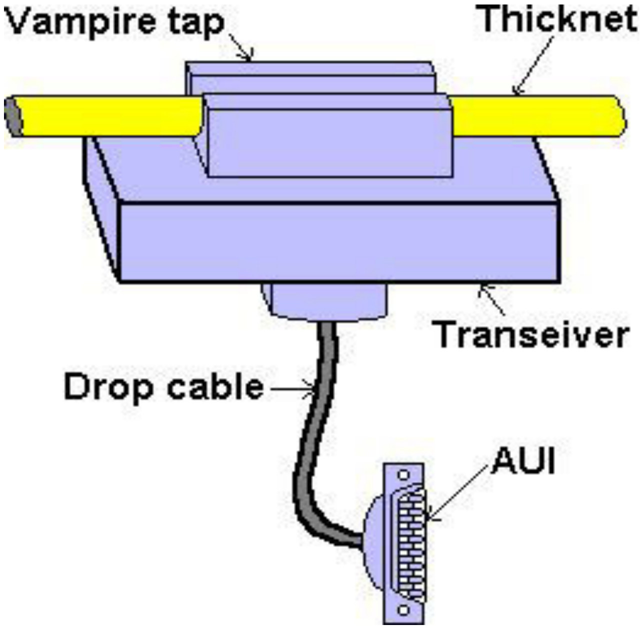
Standard Ethernet Implementation: 10Base5

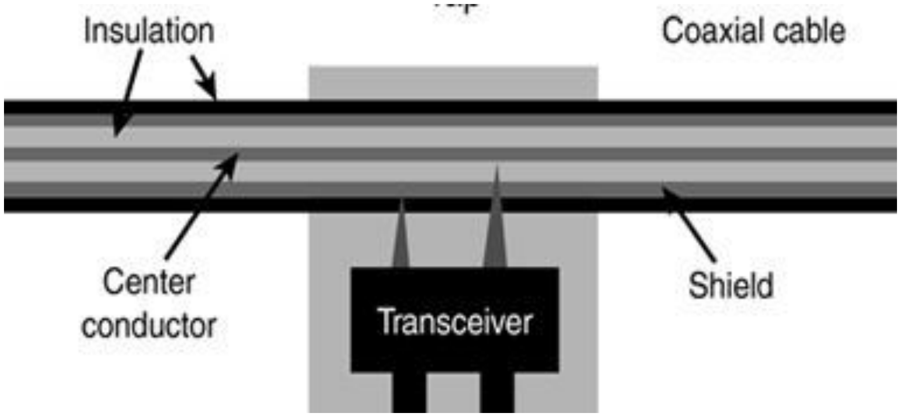
- Also called *Thick Ethernet* or *Thicknet*
- Bus topology
- Used Thick coaxial cable as backbone.
- The cable is roughly the size of a garden hose and too stiff to bend with your hands.
- The cable ends are grounded to avoid signal bouncing.



Standard Ethernet Implementation: 10Base5...

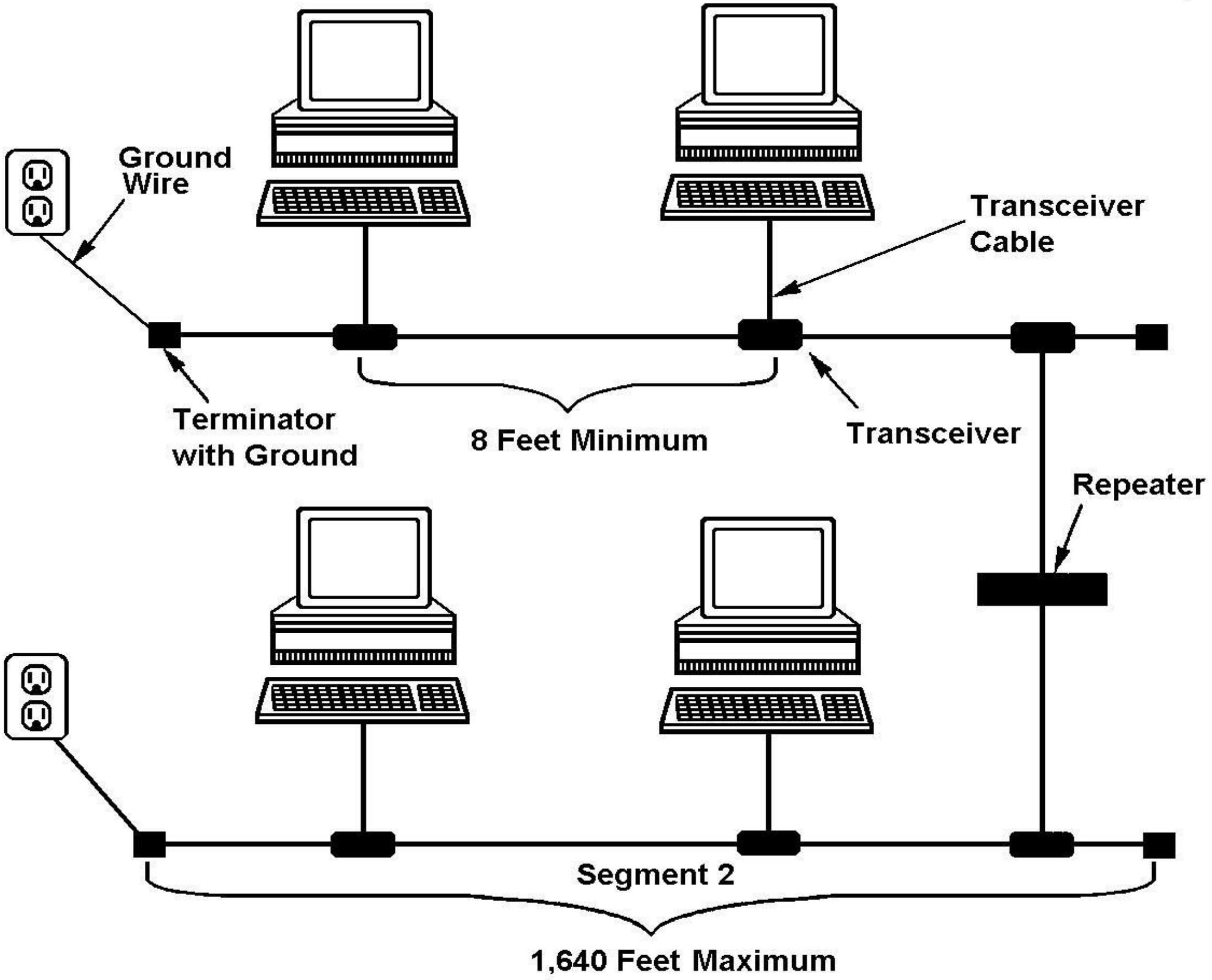
- An external transceiver (transmitter/receiver) connected via a vampire tap to the coaxial cable.
- The transceiver is responsible for transmitting, receiving, and detecting collisions.
- The transceiver is connected to the station via a transceiver cable that provides separate paths for sending and receiving.





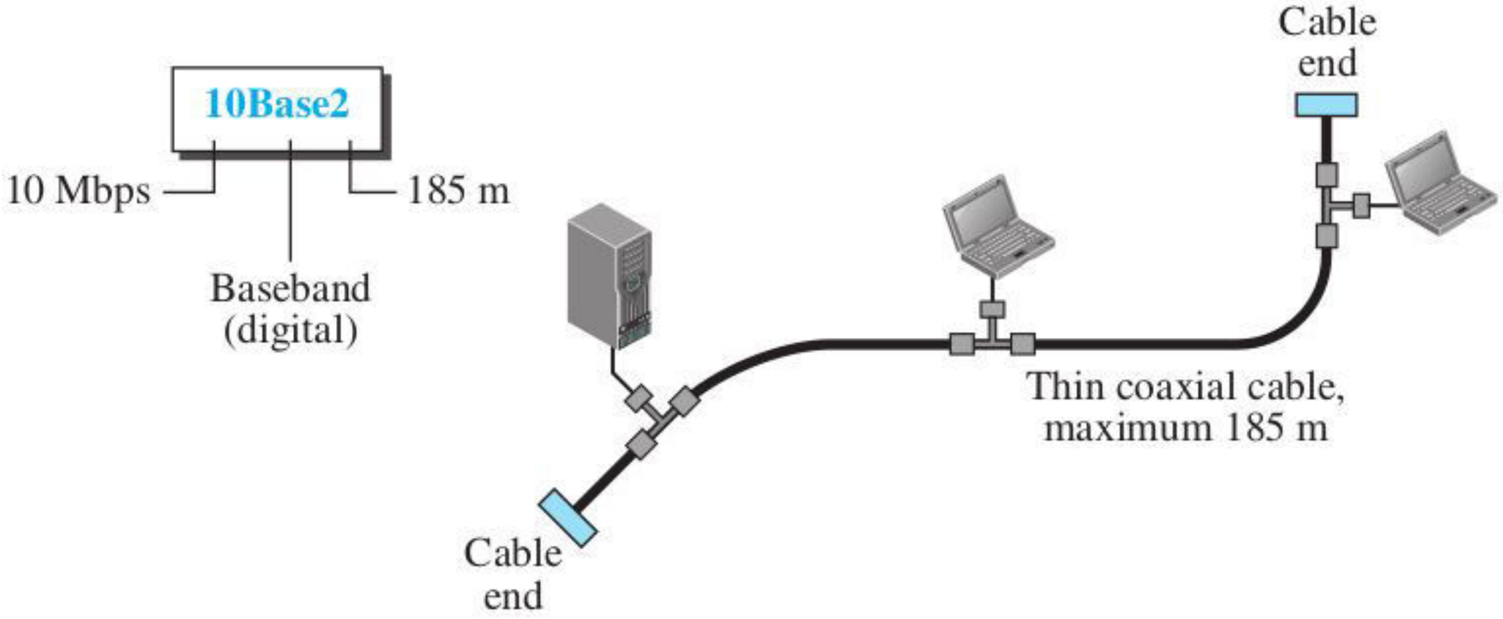
Standard Ethernet Implementation: 10Base5...

- The maximum length of the coaxial cable must not exceed 500 m, otherwise, there is excessive degradation of the signal.
- If a length of more than 500 m is needed, up to five segments, each a maximum of 500 meters, can be connected using repeaters.



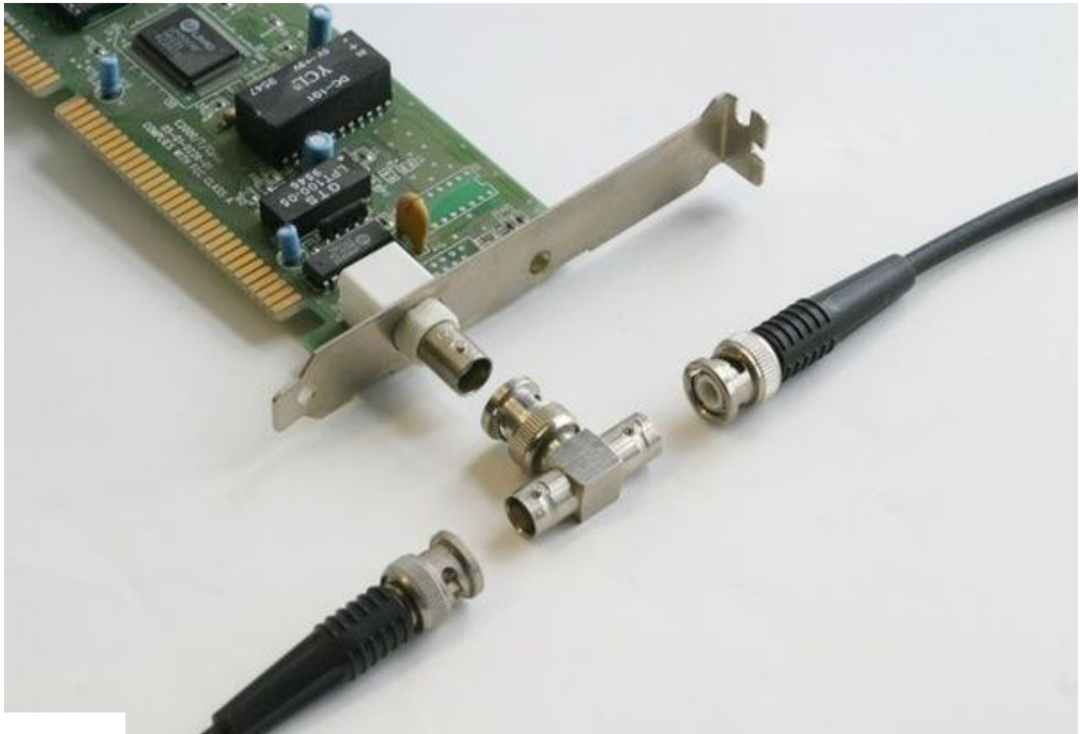
Standard Ethernet Implementation: 10Base2

- Also called Thin Ethernet or Thinnet or Cheapernet.
- Bus topology.
- The cable is thin and flexible and can be bent to pass very close to the stations.
- The transceiver is normally part of the network interface card (NIC).
- The collision here occurs in the thin coaxial cable.
- More cost effective and simpler installation than Thicknet.
- The length of each segment cannot exceed 185 m.



Standard Ethernet Implementation: 10Base2

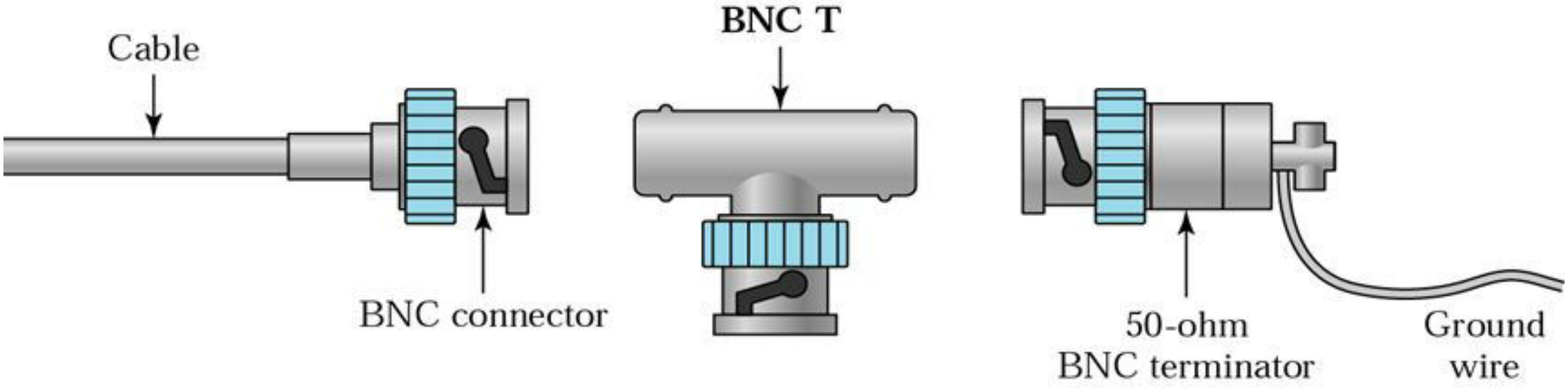
- Use different types of BNC Connectors at the cable ends, taps and joints.
- BNC: Bayonet Neill–Concelman



BNC Barrel Connector

BNC T Connector

Standard Ethernet Implementation: 10Base2



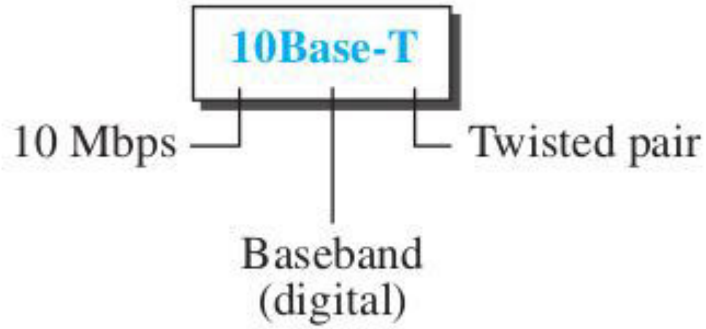
BNC Terminator

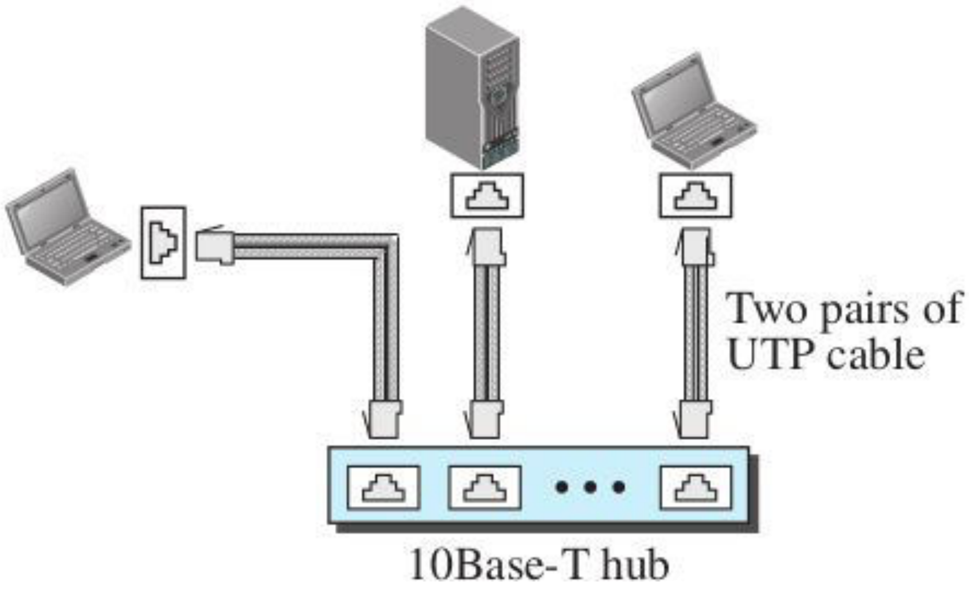
BNC Terminator

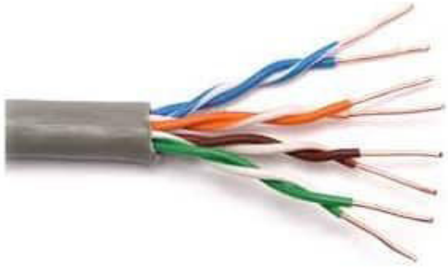
BNC T connector

Standard Ethernet Implementation: 10Base-T

- 10Base-T: **Twisted-pair Ethernet**.
- Uses Star topology.





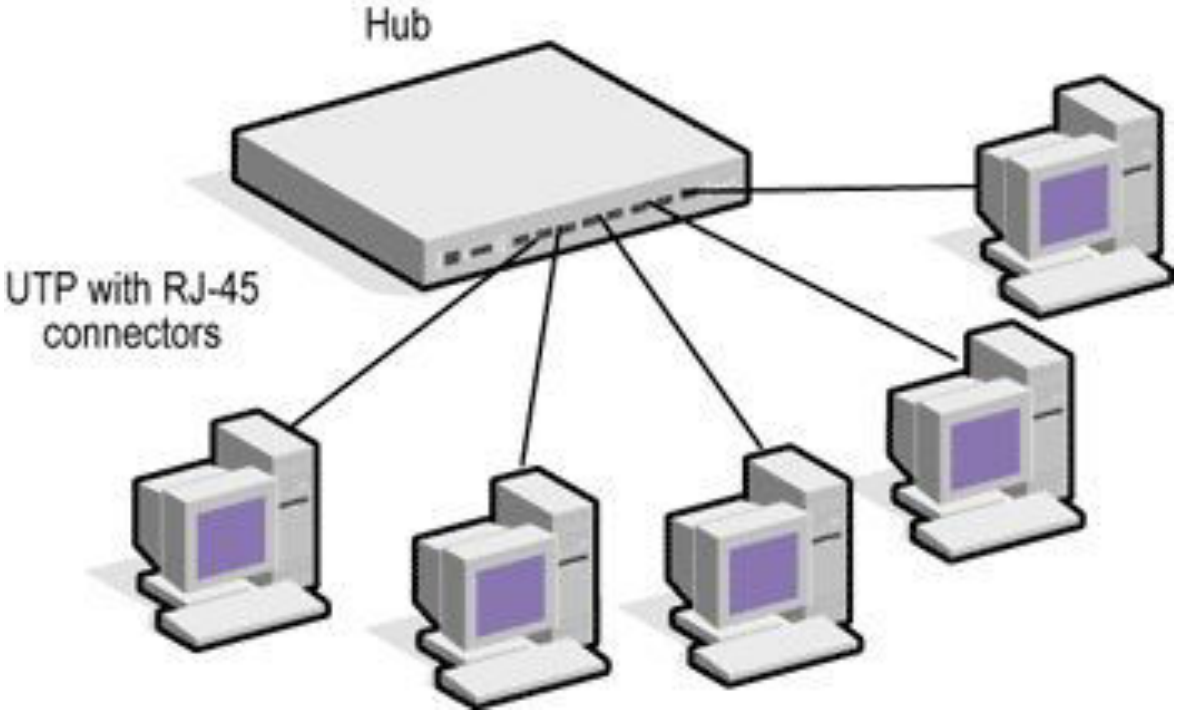

UTP Cable

STP Cable

Standard Ethernet Implementation: 10Base-T

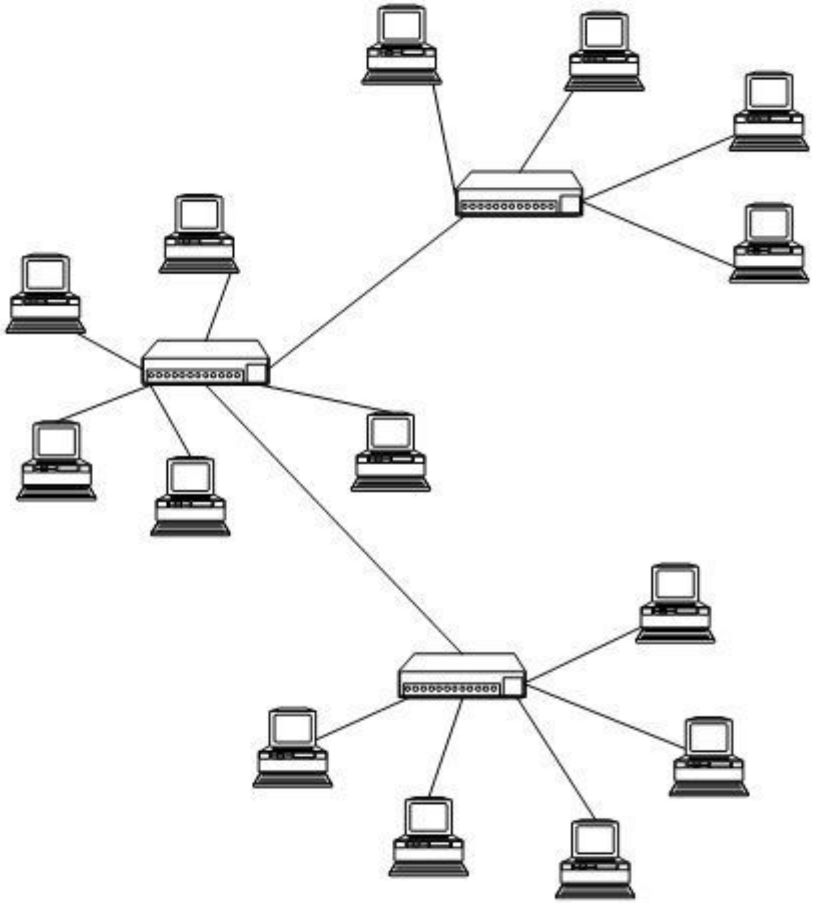
- The computers are connected to a hub using a twisted cable and RJ-45 connector.

RJ-45 Connector



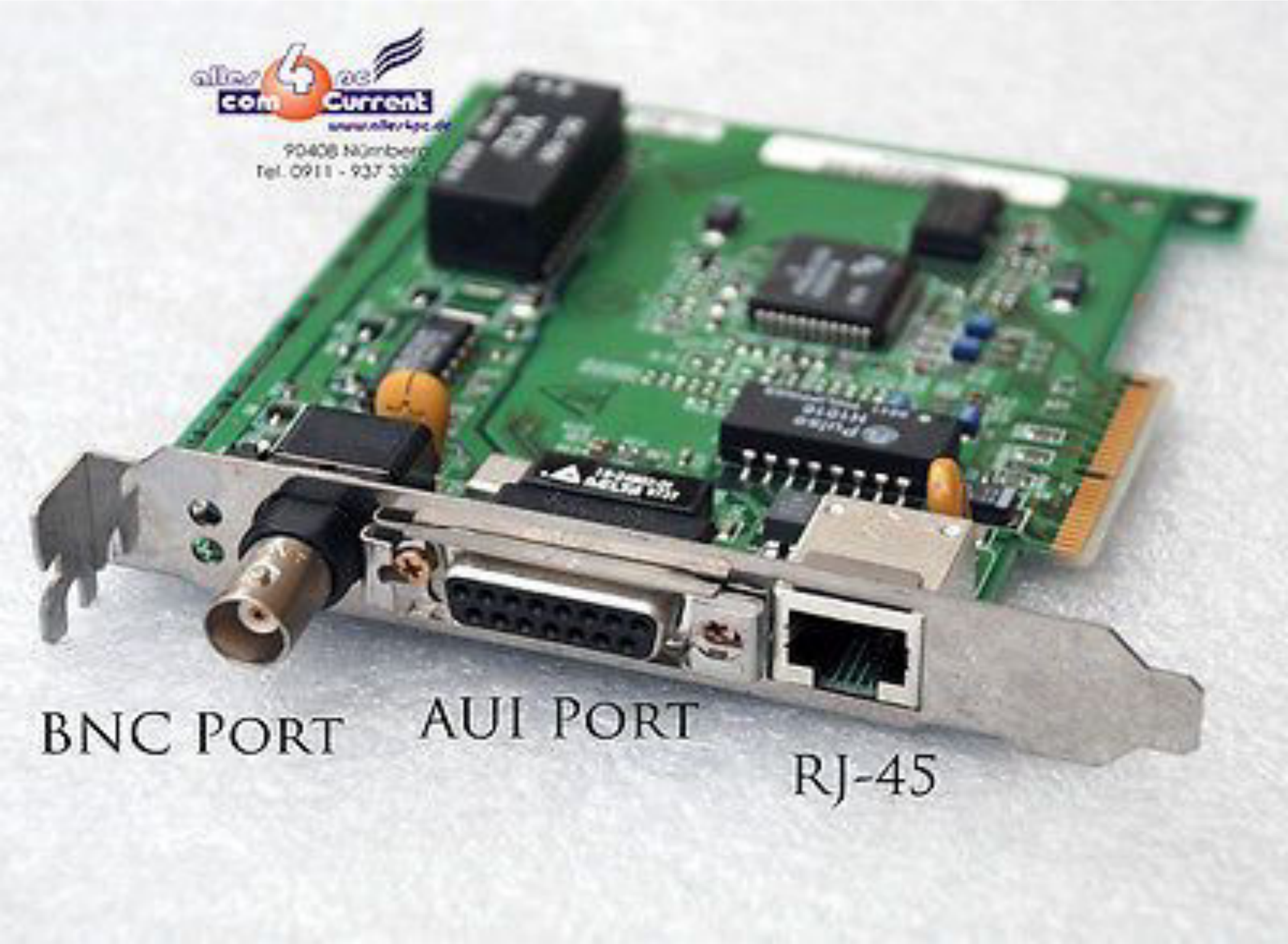
Standard Ethernet Implementation: 10Base-T

- The twisted pair cable has two pairs of wires for sending and receiving data.
- Any collision here happens in the hub.
- The maximum length of the twisted cable is defined as 100 m, but can be extended using extended-star topology (given below).



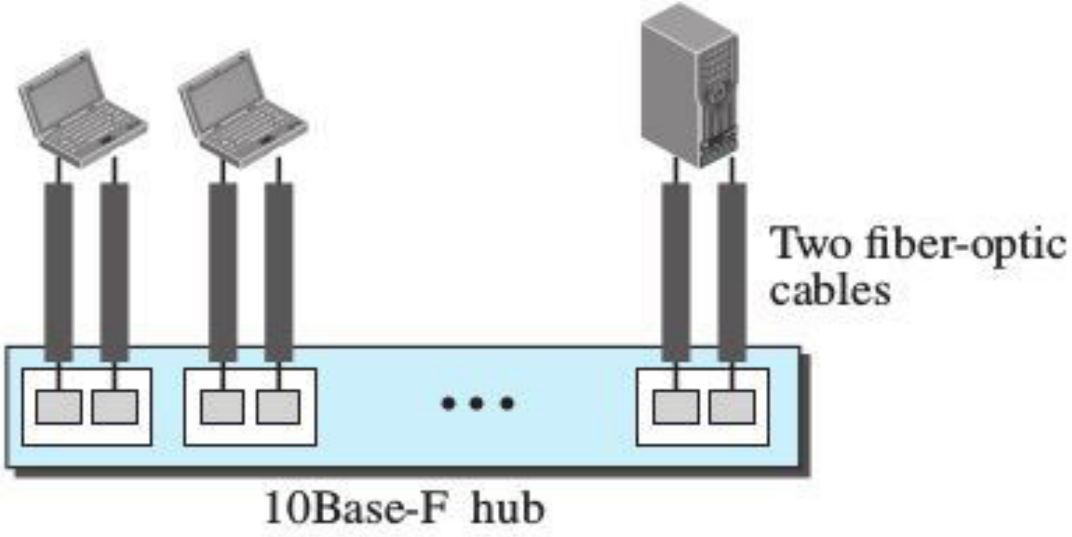
Standard Ethernet Implementation: 10BaseT

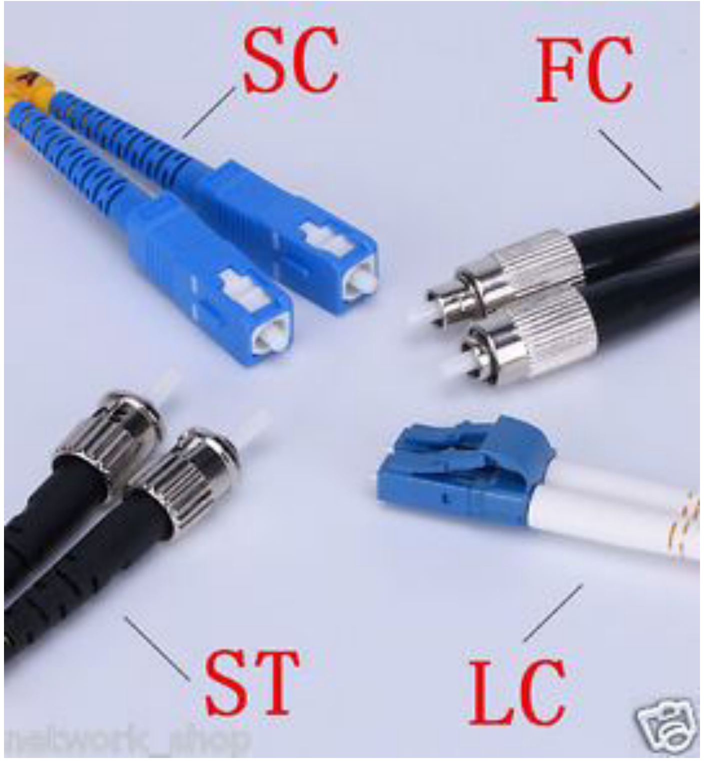
- The below NIC supports 10Base2, 10Base5 and 10Base-T Ethernet



Standard Ethernet Implementation: 10Base-F

- 10Base-F: **Fiber Ethernet**
- Uses a star topology to connect stations to a hub.
- Maximum length is 2km
- Use ST connectors mostly, but other variants are also available.



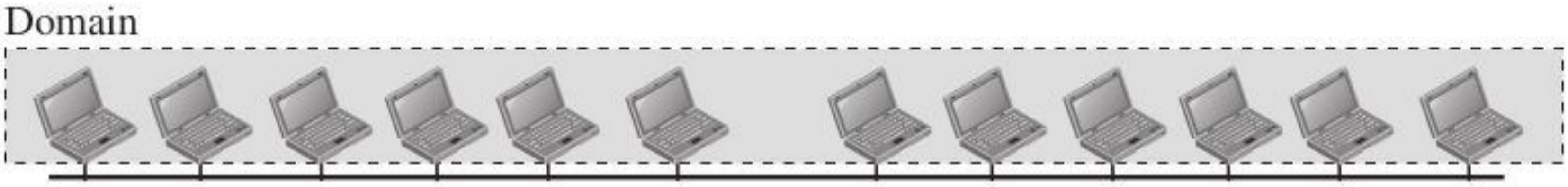


Different fibre-optic connectors

Back of a Fibre-Optic hub

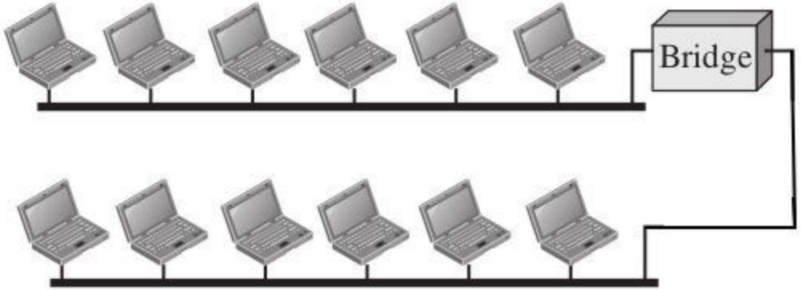
Changes in the Standard

- Bridged Ethernet
  - ○ Raising the bandwidth
  - ○ Separating collision domains

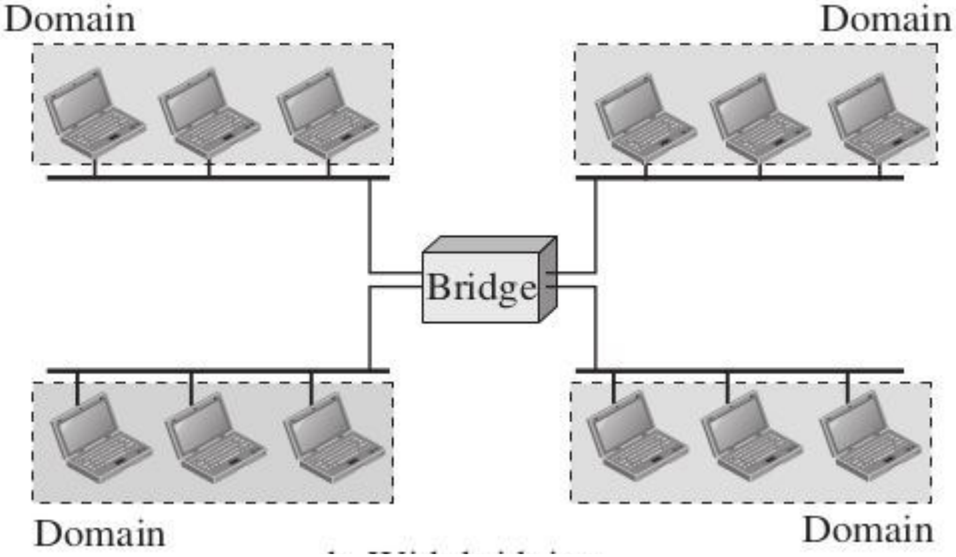


a. Without bridging

2-port bridge

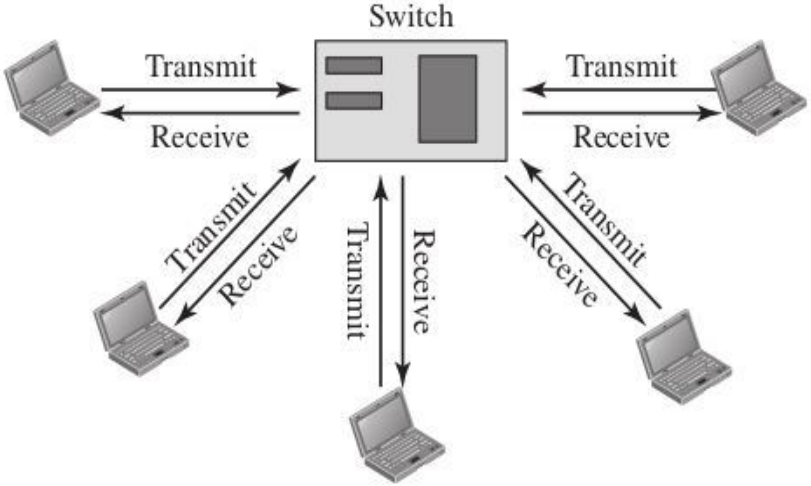


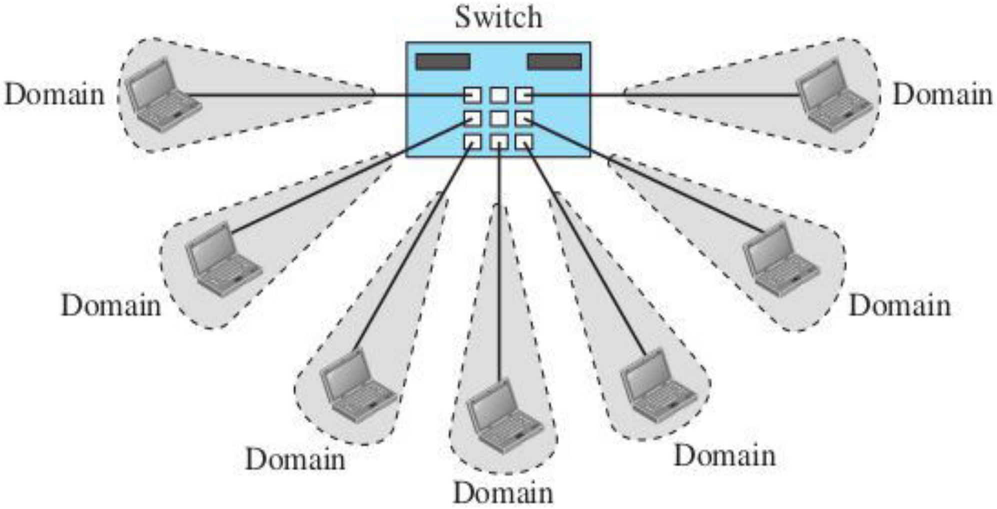
4-port bridge



Changes in the Standard

- Switched Ethernet
  - Switch is an N-port bridge.
  - Hubs replaced by layer-2 switch.
  - Point-to-point bidirectional path between device and switch.
  - Full-duplex Ethernet, hence effectively 20 Mbps.
  - Switches provided with buffers.
  - No need of CSMA/CD.



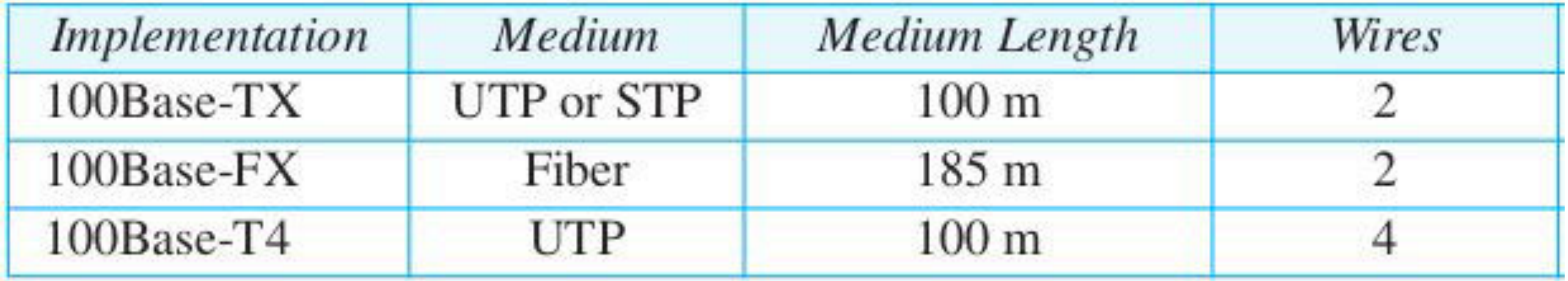


Fast Ethernet: 100 Mbps

- The goals of Fast Ethernet
  - Upgrade the data rate to 100 Mbps (10 times that of Standard Ethernet).
  - Make it compatible with Standard Ethernet.
  - Keep the same 48-bit address.
  - Keep the same frame format.
- Drop the bus topology and use a passive hub and star topology with reduced network length.
- Or use link layer switches with buffer and full-duplex connection.
- Use **Autonegotiation**
  - Allows two devices to negotiate the mode or data rate of operation.
  - Allows incompatible devices to connect each other (10 <-> 100 Mbps).
  - Allows one device to have multiple capabilities.
  - Allows a station to check a hub's capabilities.

Fast Ethernet: 100 Mbps

- Topology: point-to-point or star.
- Implementations:
  - 100Base-TX uses two pairs of twisted-pair cable (either category 5 UTP or STP).
  - 100Base-FX uses two pairs of fiber-optic cables.
  - 100Base-T4, was designed to use category 3 or higher UTP. The implementation uses four pairs of UTP for transmitting 100 Mbps. This was implemented to reuse the old 10Base-T wiring.



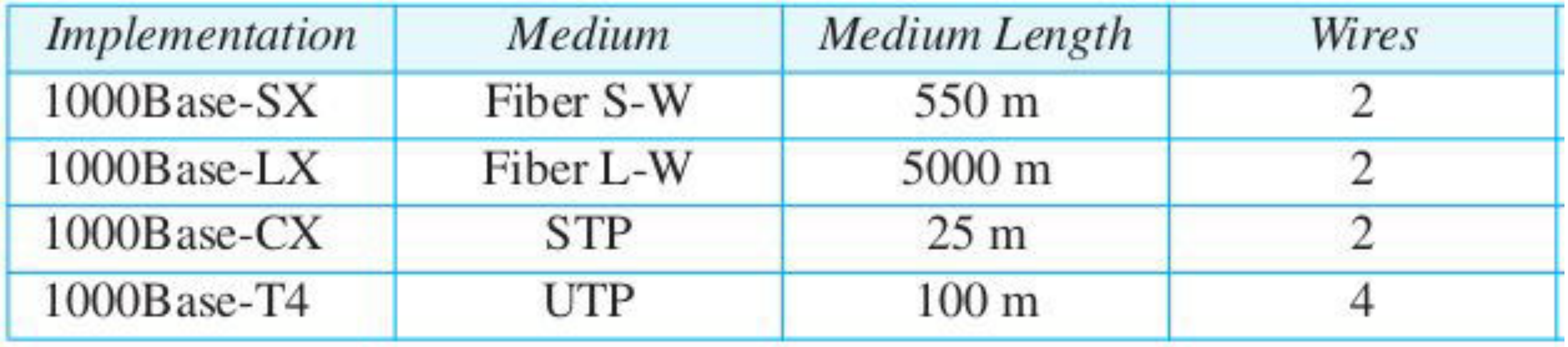
Implementation	*Medium*	*Medium Length*	*Wires*
100Base-TX	UTP or STP	100 m	2
100Base-FX	Fiber	185 m	2
100Base-T4	UTP	100 m	4

Gigabit Ethernet: 1000 Mbps

- IEEE 802.3z
- The goals of the Gigabit Ethernet:
  - Upgrade the data rate to 1 Gbps.
  - Make it compatible with Standard or Fast Ethernet.
  - Use the same 48-bit address.
  - Use the same frame format.
  - Keep the same minimum and maximum frame lengths.
  - Support **autonegotiation** as defined in Fast Ethernet.
- Used two approaches for medium access;
  - Full-duplex Mode: Use switches with buffers, hence no CSMA/CD (most common).
  - Half-Duplex Mode: Use hub, hence CSMA/CD (very rare).

Gigabit Ethernet: 1000 Mbps

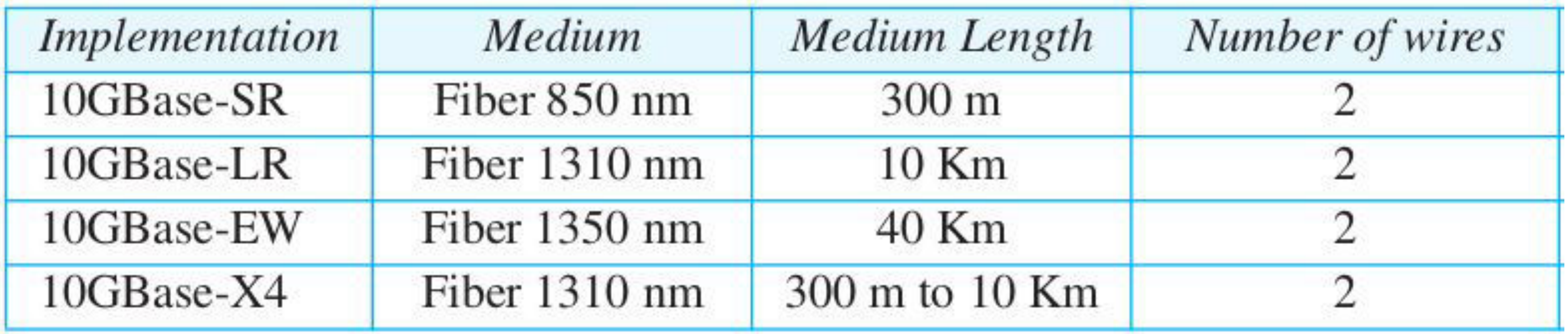
- Topology: point-to-point or star.
- Implementations: categorized as either a two-wire or a four-wire implementation.
    - The two-wire implementations use fiber-optic cable (1000Base-SX, short-wave, or 1000Base-LX, long-wave), or STP (1000Base-CX).
    - The four-wire version uses category 5 twisted-pair cable (1000Base-T) to reuse Fast Ethernet wiring.



Implementation	*Medium*	*Medium Length*	*Wires*
1000Base-SX	Fiber S-W	550 m	2
1000Base-LX	Fiber L-W	5000 m	2
1000Base-CX	STP	25 m	2
1000Base-T4	UTP	100 m	4

10 Gigabit Ethernet: 10000 Mbps

- IEEE 802.3ae
- The goals of the Gigabit Ethernet:
  - ○ Upgrade the data rate to 10 Gbps.
  - ○ Use the same frame format.
  - ○ Keep the same frame lengths.
  - ○ Allow the interconnection of LANs, MANs, and WAN possible.
- Implemented with Fiber-optic technology.
- Used full-duplex mode with no CSMA/CD.



Implementation	*Medium*	*Medium Length*	*Number of wires*
10GBase-SR	Fiber 850 nm	300 m	2
10GBase-LR	Fiber 1310 nm	10 Km	2
10GBase-EW	Fiber 1350 nm	40 Km	2
10GBase-X4	Fiber 1310 nm	300 m to 10 Km	2

Diploma in Computer Hardware Engineering

COMPUTER NETWORKS

(5151 Rev 2015)

Module 1 - Part 3 of 3

Presenter: Sreejesh NG
Lecturer in Computer Hardware Engineering
Government Polytechnic College, Cherthala

Ref: Data Communications and Networking, 5E by Forouzan

Course General Outcomes:

Sl.	G.O	On completion of this course the student will be able :
1	1	To Understand the concept of TCP/IP Protocol
2	1	To Understand the concept of Network Layer
3	1	To Understand the concept of Transport Layer
4	1	To Understand the concept of Application Layer

Specific Outcomes:

MODULE I. REVIEW OF NETWORK MODELS

1.1 Understand TCP/IP Protocol

- 1.1.1 Illustrate computer networks
- 1.1.2 Identify TCP/IP Protocol suite.
- 1.1.3 Explain the functionalities of layers in TCP/IP
- 1.1.4 Define Addressing of TCP/IP.
- 1.1.5 Describe about Wired LAN – Ethernet
- 1.1.6 State IEEE 802 project
- 1.1.7 Illustrate standard Ethernet
- 1.1.8 Describe about Wireless LAN.
- 1.1.9 State IEEE 802.11
- 1.1.10 Explain LAN connecting devices.
- 1.1.11 Explain the architecture of Virtual LANs.

MODULE 1 - TCP/IP PROTOCOL

- Introduction to computer networks - physical structure, topology, types.
- TCP/IP - architecture, Description of layers, addressing.
- Wired LAN - Ethernet protocol - IEEE project 802 - Standard Ethernet - characteristics, addressing, implementation.
- Wireless LAN - architectural comparison, characteristics, access control - IEEE 802.11 - architecture.
- LAN connecting devices - hub, switch, router.
- Virtual LAN - architecture, membership, configuration.

Ref: Data Communications and Networking - Behrouz A. Forouzan -
McGraw Hill Edn.-Fourth Edition/Fifth Edition

Wireless LAN

- Architectural Comparison between Wired and Wireless LAN
  - Medium
  - Hosts
  - Isolated LANs
  - Connection to Other Networks
  - Moving between Environments

Wireless LAN

Architectural Comparison between Wired and Wireless LAN

- Medium

Wired LAN	Wireless LAN
● Use wired medium ● Point-to-point and full duplex ● Dedicated medium for each device.	● Use wireless medium ● Broadcast ● All devices share the same medium.

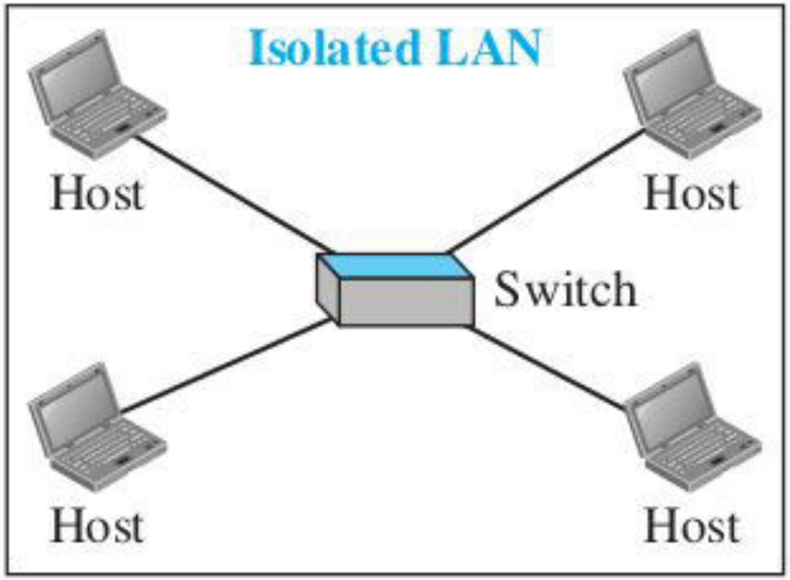
- Hosts

Wired LAN	Wireless LAN
● Host is always connected. ● Movement is limited.	● No need of physical connection. ● Free movement.

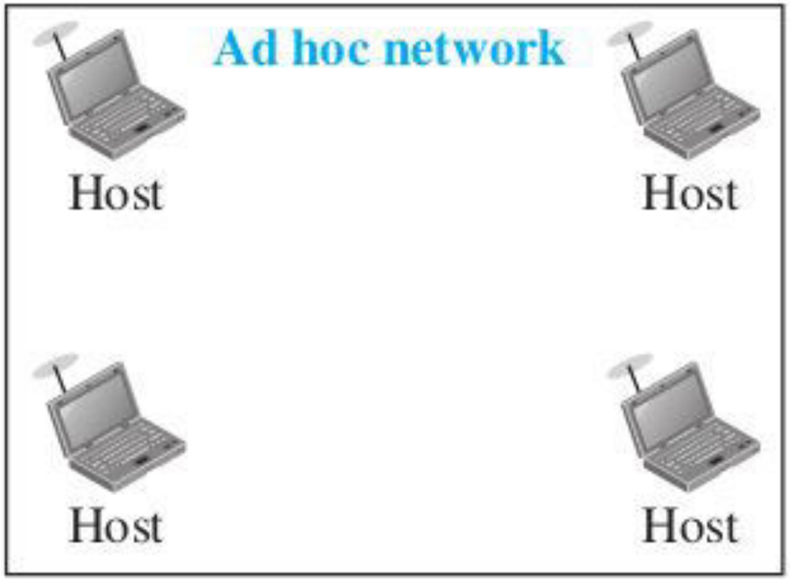
Wireless LAN

- Isolated LANs

Wired LAN	Wireless LAN
● A wired isolated LAN is a set of hosts connected via a link-layer switch	● A wireless isolated LAN, called an ad hoc network in wireless LAN terminology, is a set of hosts that communicate freely with each other.



Wired

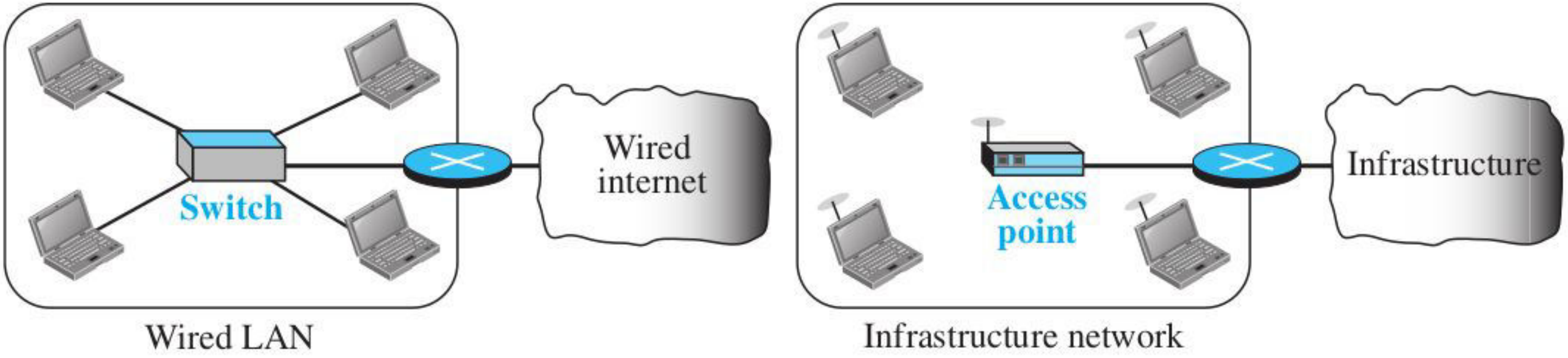


Wireless

Wireless LAN

- Connection to Other Networks

Wired LAN	Wireless LAN
● Can be connected to another network or an internetwork such as the Internet using a *router*.	● May be connected to a wired infrastructure network, to a wireless infrastructure network, or to another wireless LAN through an *access point.* ● An Adhoc network with an access point is called Infrastructure network.



Wireless LAN

- Moving between Environments
  - A wired LAN or a wireless LAN operates only in the lower two layers of the TCP/IP protocol suite.
  - Suppose we have a wired LAN connected to the Internet through a router or modem.
    - If we have to change the network to wireless, change the wired NIC with a wireless NIC and replace the link-layer switch with an access point.

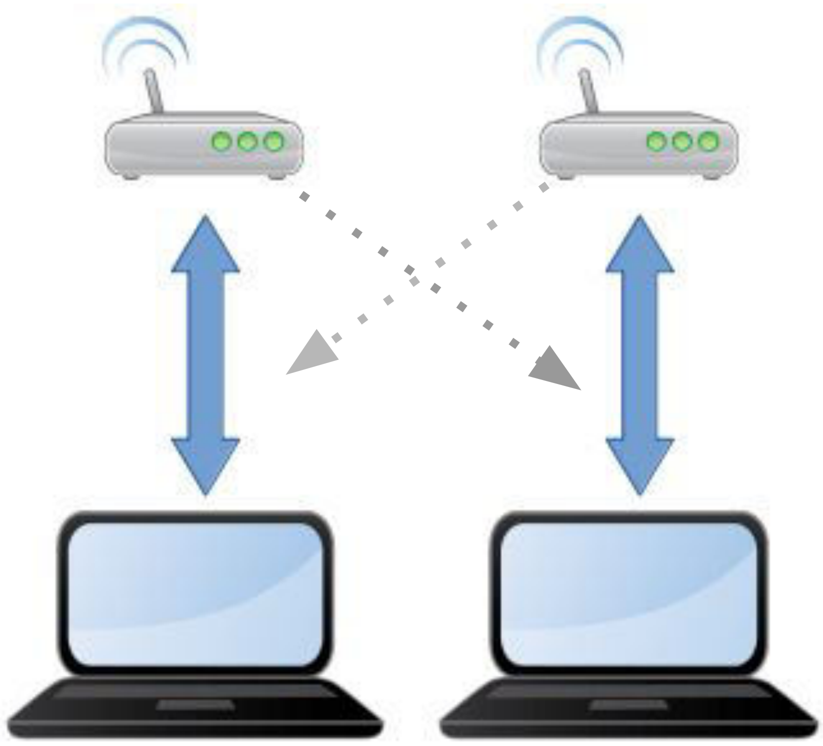
Wireless LAN: Characteristics

- There are several characteristics of wireless LANs that either do not apply to wired LANs or the existence of which is negligible and can be ignored.
- Some are;
  - Attenuation
  - Interference
  - Multipath Propagation
  - Error

- **Attenuation:**
  - The strength of electromagnetic signals decreases rapidly because the signal disperses in all directions.
  - Only a small portion of it reaches the receiver.
  - If the sender works on battery power, situation becomes worse.

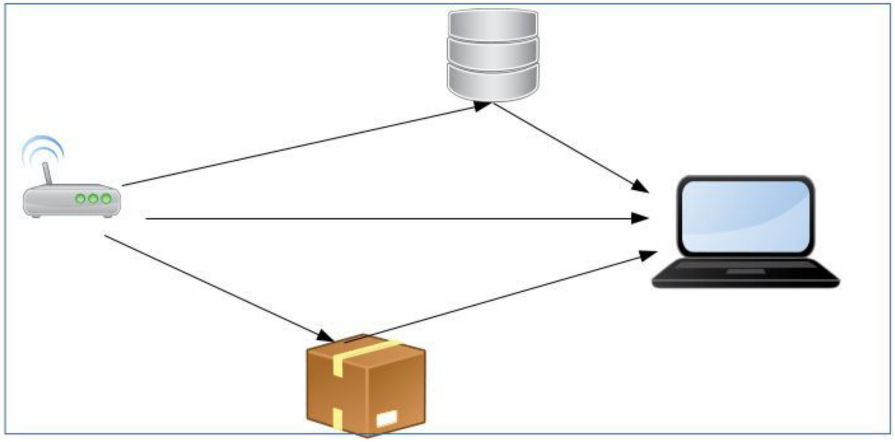
Wireless LAN: Characteristics

- **Interference**
  - a receiver may receive signals not only from the intended sender, but also from other senders if they are using the same frequency band.



Wireless LAN: Characteristics

- **Multipath Propagation**
  - A receiver may receive more than one signal from the same sender because electromagnetic waves can be reflected back from obstacles such as walls, the ground, or objects.
  - The receiver receives the same signal at different phases, and this makes the signal less recognizable.



- **Error**
  - All the above factors cause errors in the signal.
  - Error is measured by Signal-to-Noise Ratio (SNR).
  - If SNR is high, the signal is stronger than the noise. If SNR is low, the signal is corrupted by the noise and the data cannot be recovered.

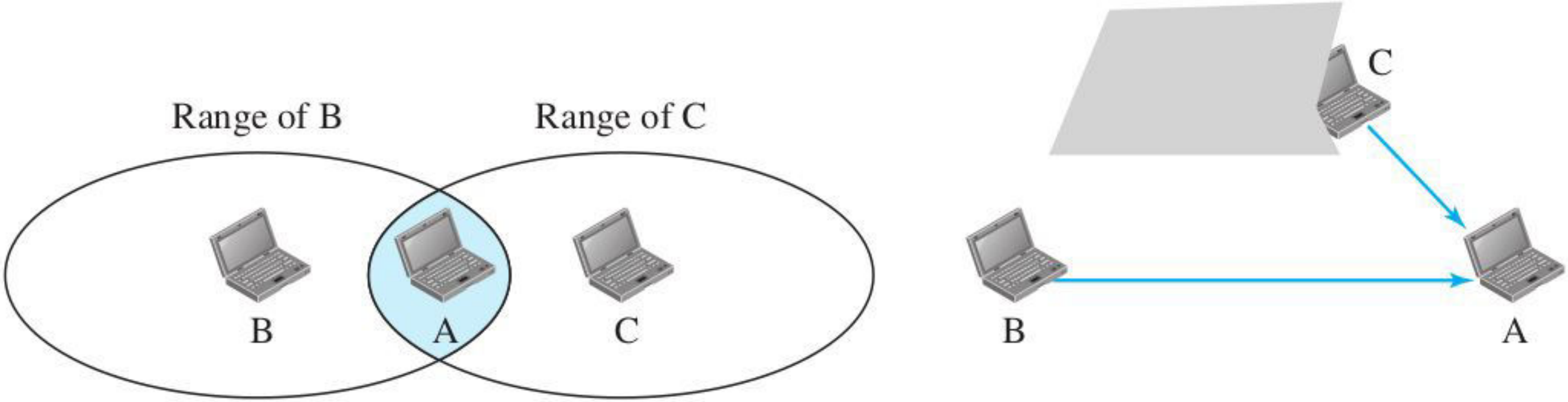
Wireless LAN: Access Control

- CSMA/CD method to access the shared medium **does not** work for wireless LAN due to the following reasons.
  1. To detect a collision, **a host needs to send and receive at the same time** (sending the frame and receiving the collision signal), which means the host needs to work in a duplex mode. Wireless hosts do not have enough power to do so (the power is supplied by batteries). They can only send or receive at one time.
  2. Because of the **hidden station problem**, collision may occur but not be detected.
  3. The **distance between stations** can be great. Signal fading could prevent a station at one end from hearing a collision at the other end.

- To overcome the above three problems, Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) was invented for wireless LANs.

Wireless LAN: Access Control

- Hidden Station Problem (or Hidden Terminal Problem)

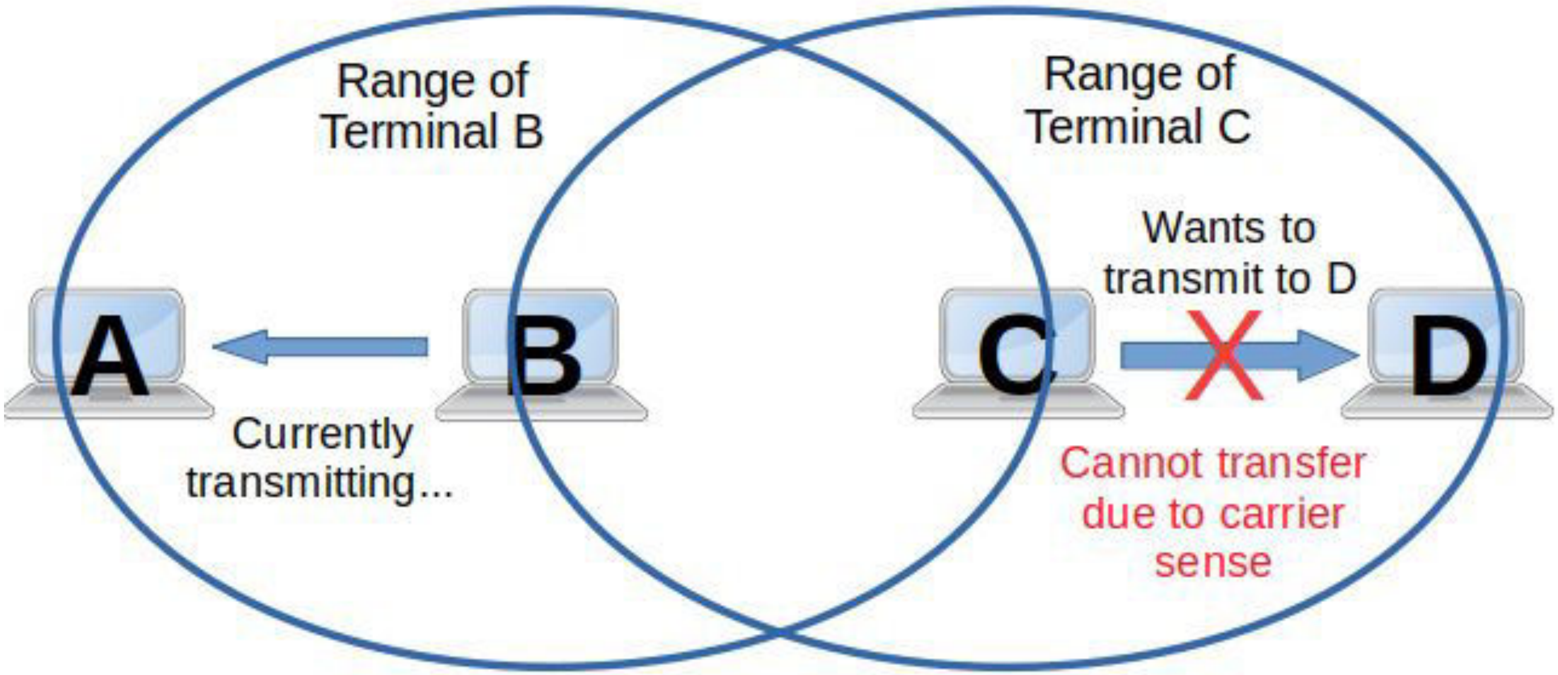


a. Stations B and C are not in each other's range.

b. Stations B and C are hidden from each other.

  - A station may not be aware of another station's transmission due to some obstacles or range problems, collision may occur but not be detected.

- Exposed Station Problem (or Exposed Terminal Problem)



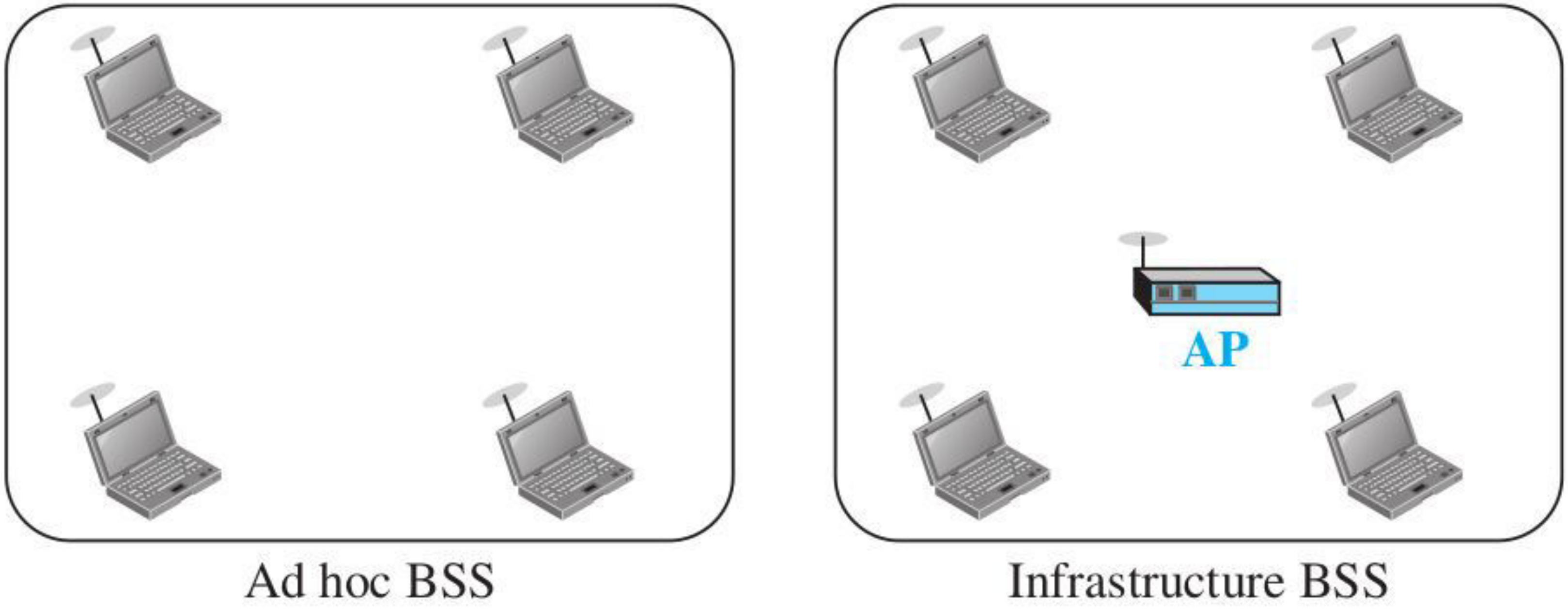
IEEE 802.11 PROJECT

- IEEE 802.11 is the specification for wireless LAN.
- It covers physical and data link layers.
- The WiFi (Wireless Fidelity) is a technology for Wireless LAN and is certified by a non-profit organization WiFi Alliance.
- All components in a wireless network are referred to as stations (STA).
- All stations are equipped with wireless network interface controllers (WNICs).
- Wireless stations fall into two categories: Wireless Access Points, and clients.
- The Access Points creates a communication path for other stations to connect with.
- Clients are end devices such as laptops, mobile phones, and all devices that have the wireless network interface.

- **Architecture**
- Two basic services:
  - Basic service set (BSS)
  - Extended service set (ESS)

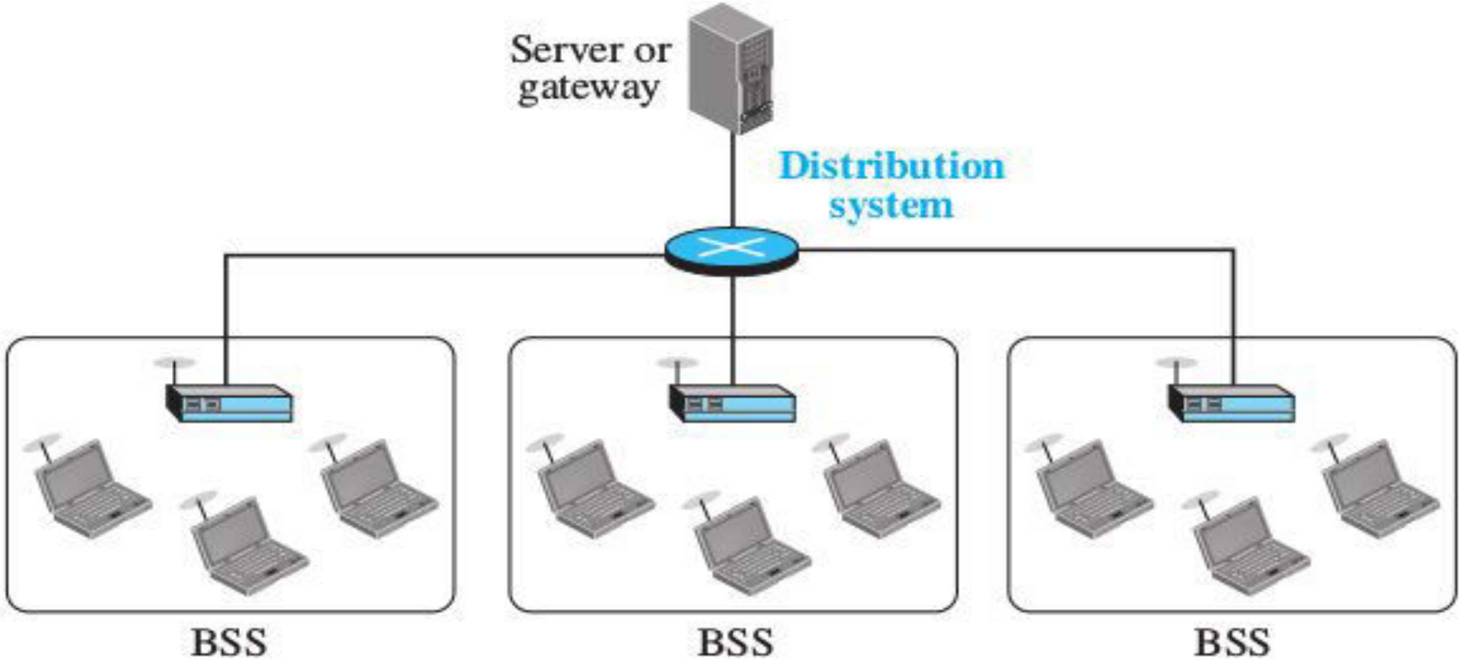
Basic Service Set (BSS)

- Building blocks of a wireless LAN.
- Made of stationary or mobile wireless stations and an optional central base station, known as the Access Point (AP).
- The BSS without an AP is a stand-alone network and cannot send data to other BSSs. It is called an ad hoc architecture.
- A BSS with an AP is sometimes referred to as an infrastructure BSS.
- Every BSS has an identification (ID) called the BSSID.



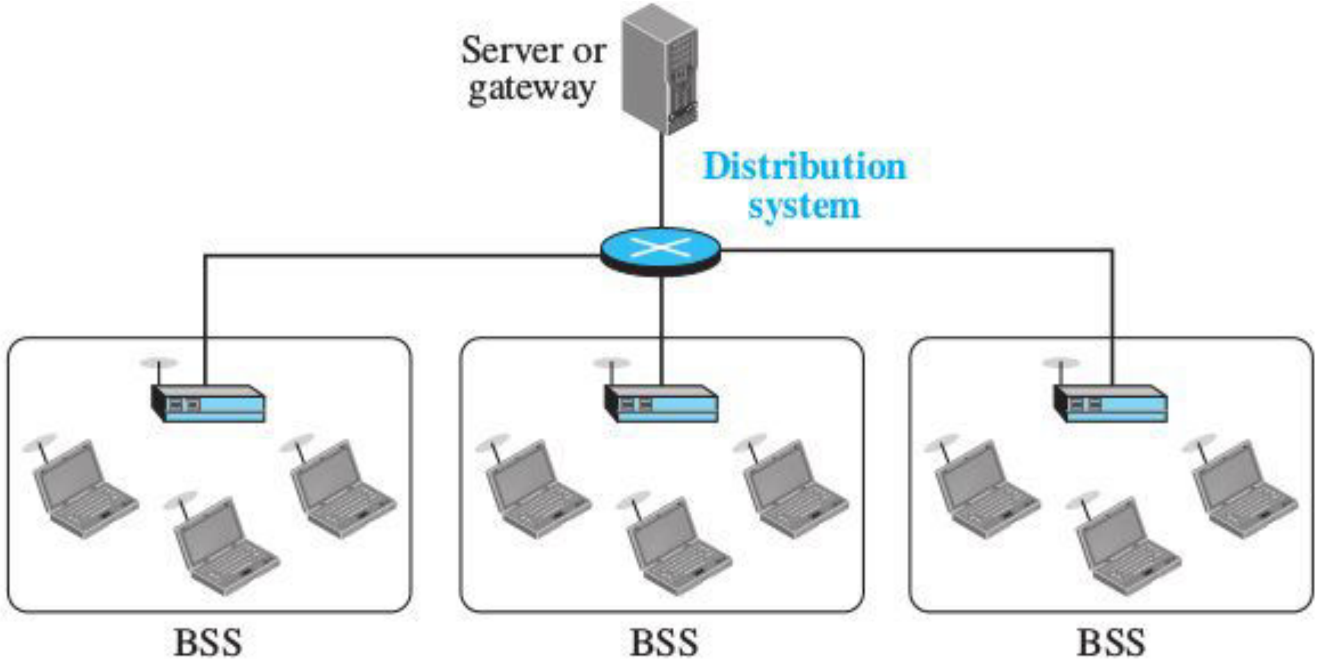
Extended Service Set (ESS)

- Made up of two or more BSSs with APs.
- Here the BSSs are connected through a distribution system, which is a wired or a wireless network.
- The distribution system connects the APs in the BSSs.
- ESS uses two types of stations: mobile and stationary.
- The mobile stations are normal stations inside a BSS.
- The stationary stations are AP stations that are part of a wired LAN.
- Communication between a station in a BSS and the outside BSS occurs via the AP.



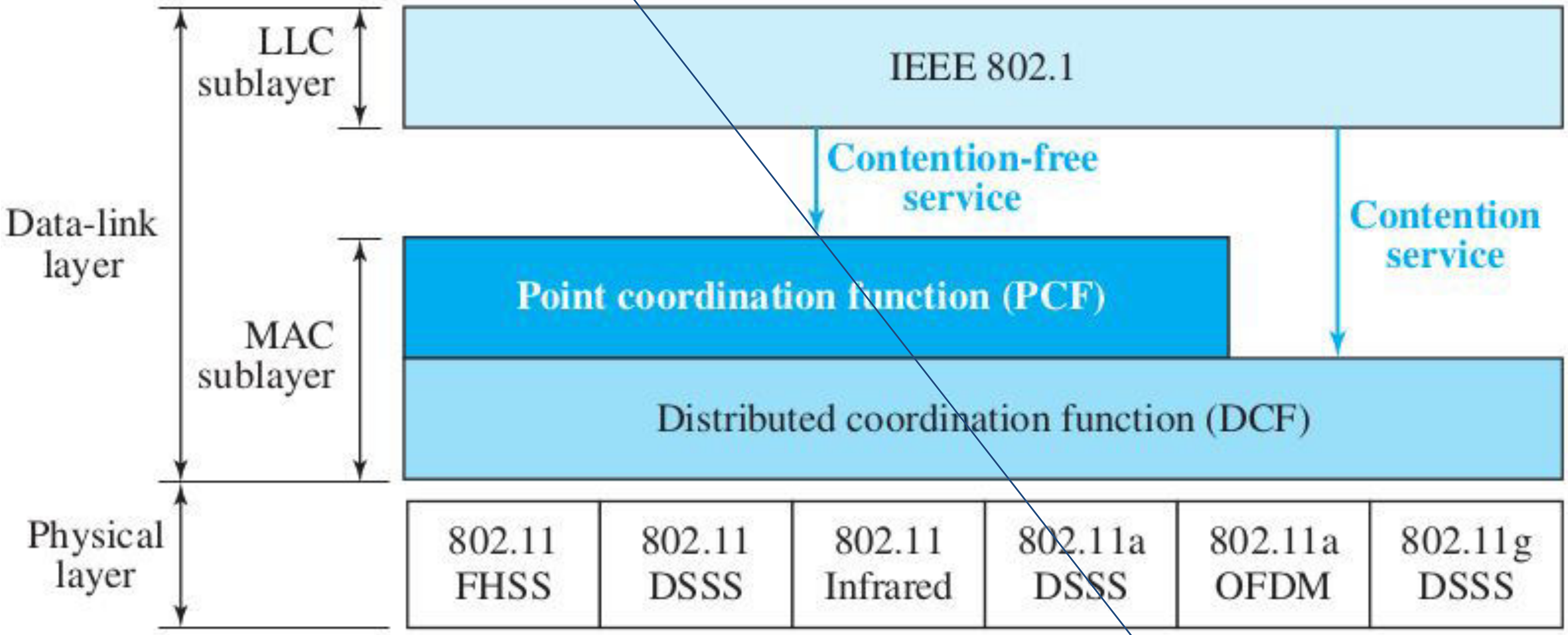
Station Types

- IEEE defines three types of stations based on the station mobility in WLAN.
- A station with **no-transition mobility** is either stationary (not moving) or moving only inside a BSS.
- A station with **BSS-transition mobility** can move from one BSS to another, but inside one ESS.
- A station with **ESS-transition mobility** can move from one ESS to another, but the communication may not be continuous.



MAC Sublayer

- IEEE 802.11 defines two MAC sublayers:
    1. Distributed Coordination Function (DCF)
    2. Point Coordination Function (PCF)

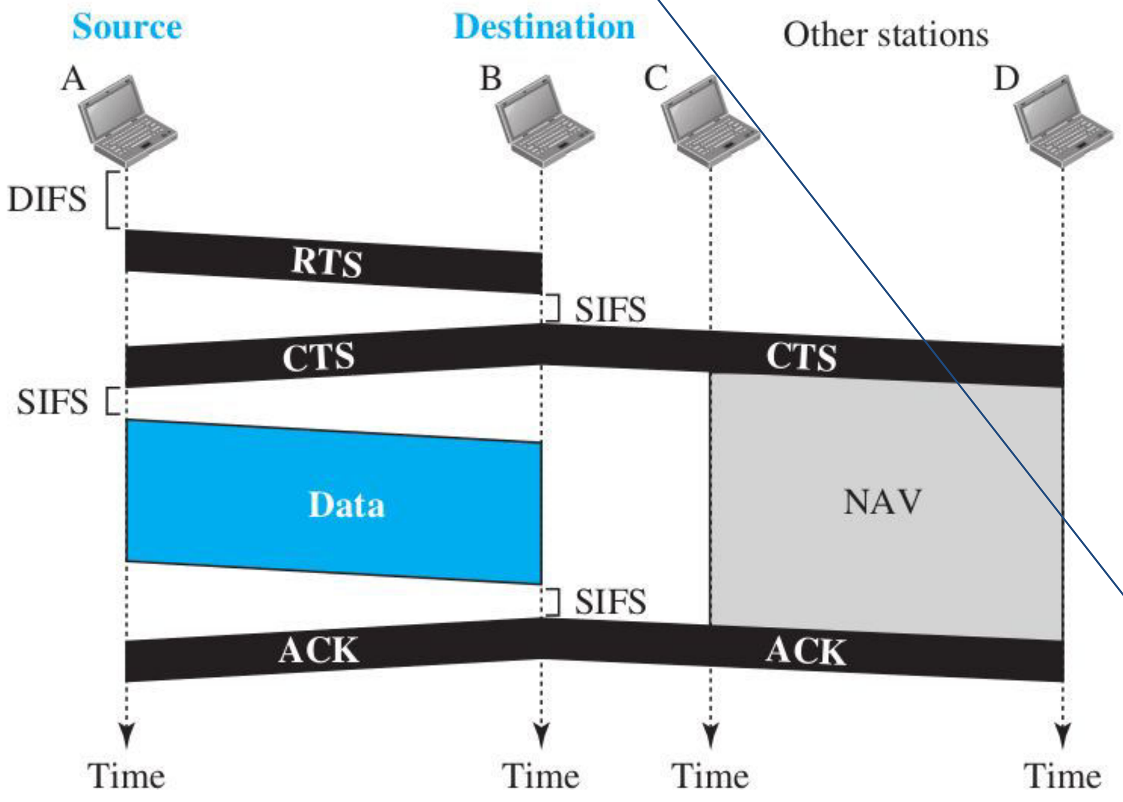


Distributed Coordination Function (DCF)

- Used in Ad hoc network
- Contention service
- Uses CSMA/CA as the access method

Distributed Coordination Function (DCF)

- Frame Exchange Timeline:



- DIFS: distributed interframe space - time for which the channel found idle
- SIFS: short interframe space
- RTS: request to send
- CTS: clear to send
- NAV: network allocation vector - a timer for all other devices to wait based on the time needed for data transfer. Set during RTS
- ACK: acknowledgement

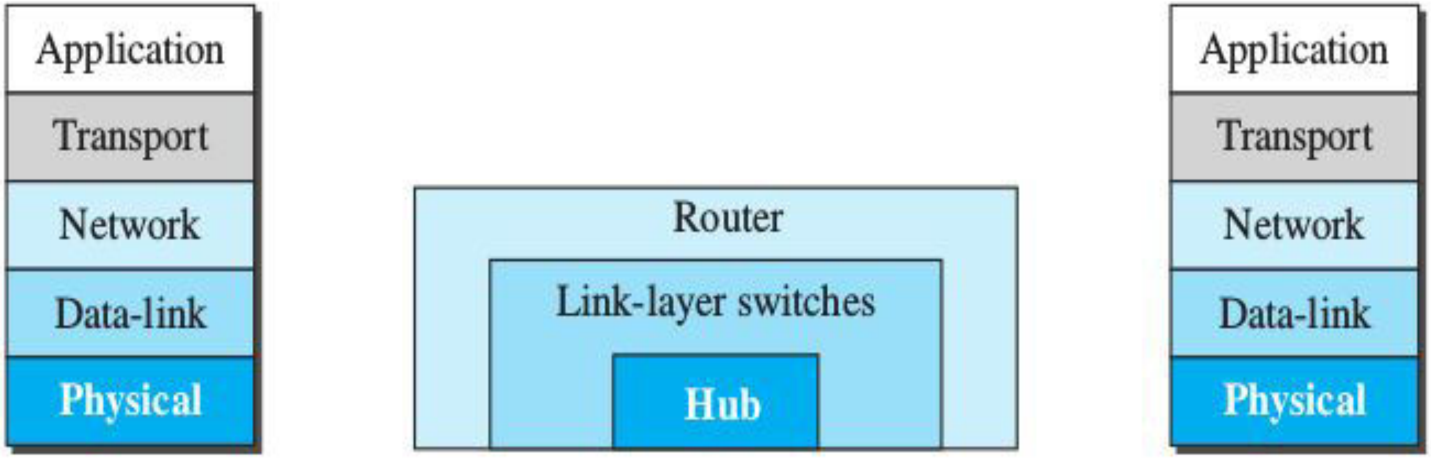
- Collision During Handshaking: Collision identified if no CTS received. Retry after a backoff.
- Hidden-Station Problem solved

MAC Sublayer: Point Coordination Function (PCF)

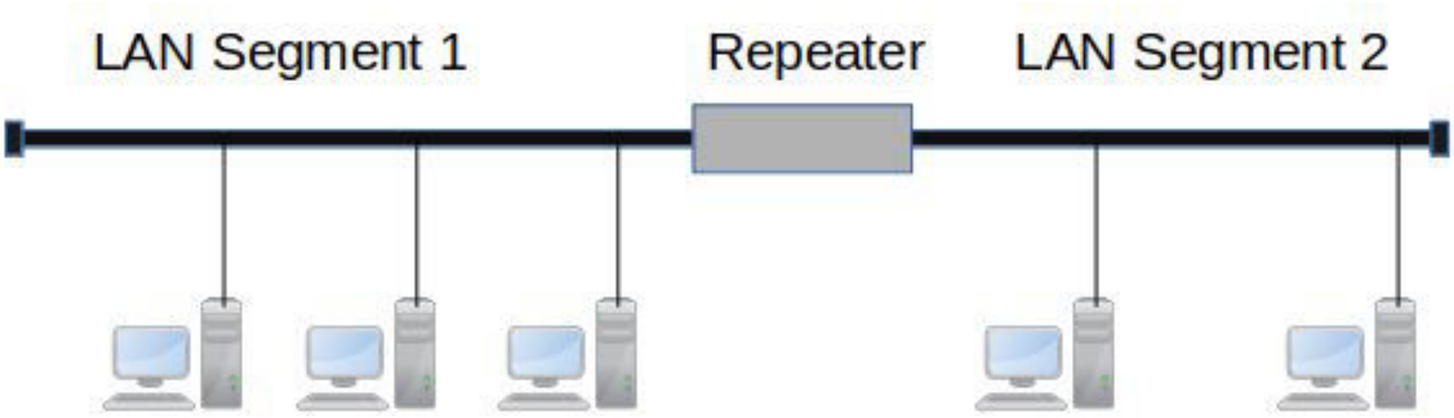
- optional access method.
- used if an Access Point is present (ie, in infrastructure network).
- has a centralized, contention-free polling access method.
- AP performs polling for stations one after another, sending any data they have to the AP.

LAN Connecting Devices

- Connecting devices to connect hosts together to make a network or to connect networks together to make an internet.
- Operates in different layers of TCP/IP protocol suite.
- Major connecting devices are;
  1. Repeater
  2. **Hub (or Multiport Repeater)**
  3. Bridge
  4. **Switch (or Multiport Bridge)**
  5. Access Point
  6. **Router**
  7. Gateway
  8. Firewall
  9. Modem

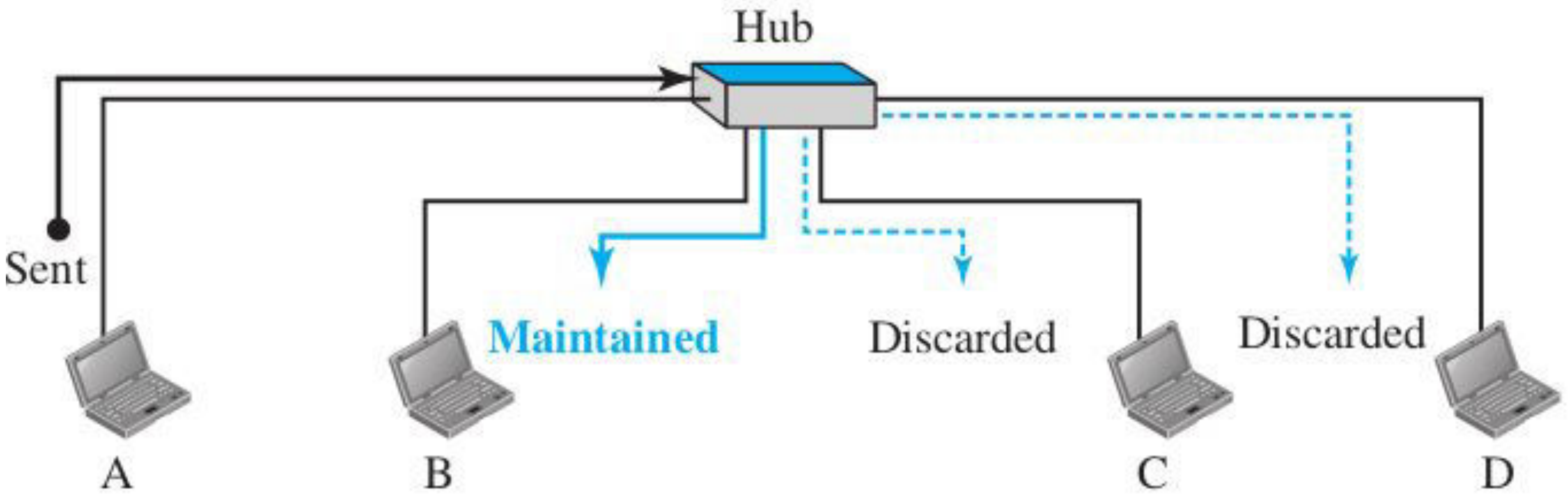


- **<u>Repeaters</u>:**
  - Layer 1 (Physical layer) device that *regenerates* and *retimes* the original transmission signal between two cable segments.
  - At first only two ports (bus topology), later multiport (star topology).
  - No filtering capability.



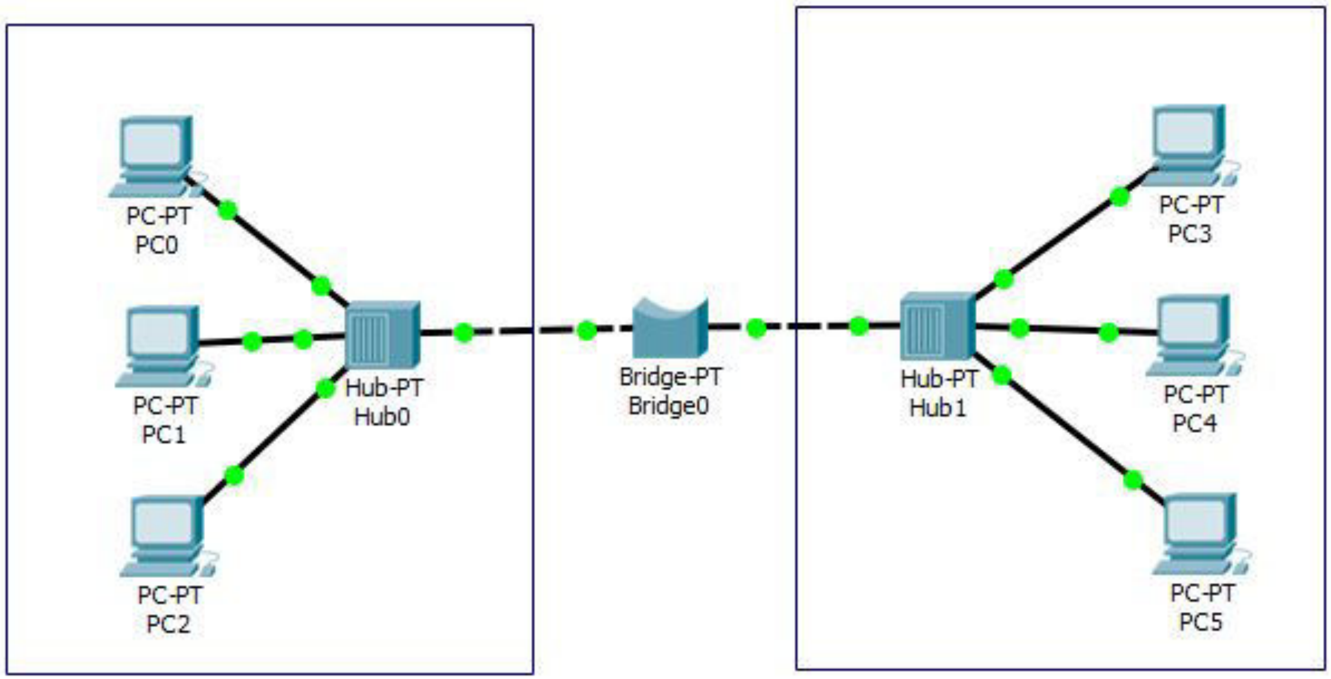
- **Hub**:
  - A multiport repeater; no intelligence (ie, it cannot understand any addresses or data in the packet).
  - Works in half-duplex mode.
  - *Active hubs (*or simply *'hub')*: Each port is a repeater that regenerates the signal for each connection.
  - *Passive hubs*: Do not regenerate the signal at all.
  - Hubs usually create a star topology with the Ethernet devices.
  - Creates a single collision domain.
  - Performance degrades as the number of hubs/devices increases.

- When a device A sends a packet to B, the hub forwards the packet to all the devices, but only B will accept it.



Bridge

- A bridge connects two network segments.
- Generally a two port device.
- A Layer-2 device, ie, bridging is done in the data link layer.
- A multiport bridge connects more than two network segments, which is the basis for network switches.

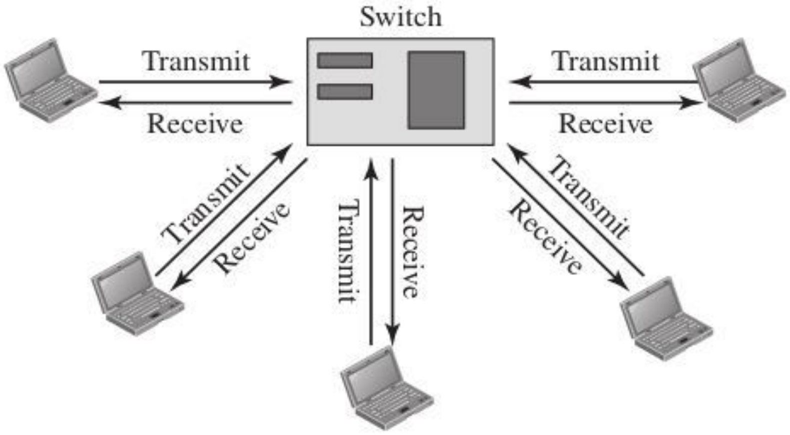


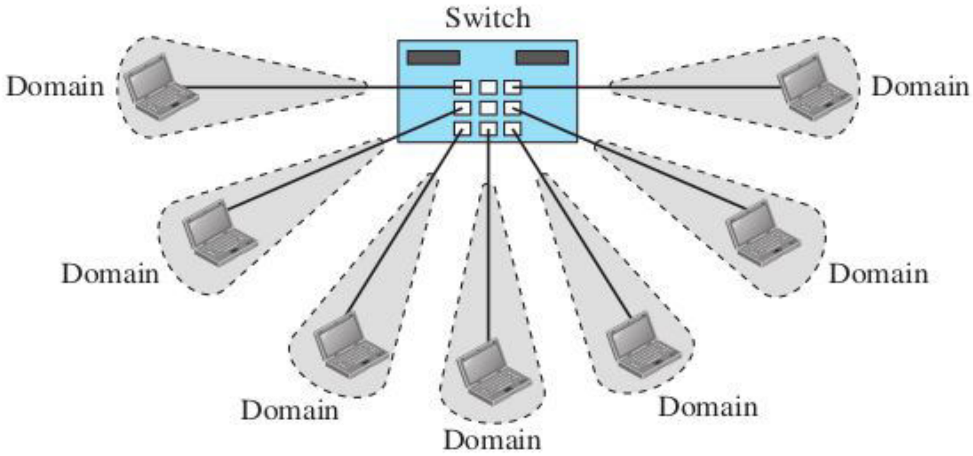
Switch (switching hub, bridging hub, officially **MAC bridge)**

Different network switches

Switch (switching hub, bridging hub, officially MAC bridge)

- Multiport bridge, first created by a company called Kalpana.
- Layer 2 device. Ie, works in data link layer and physical layer.
- Visually similar to hub.
- Mainly in twisted pair connections; but available in Fibre Channel and Asynchronous Transfer Mode (ATM).
- Filtering: Forwards frames based on MAC addresses.
- Switched *point-to-point* connections between communicating devices; hence *multiple collision domains*, *full use of band width* and *full duplex connections*.





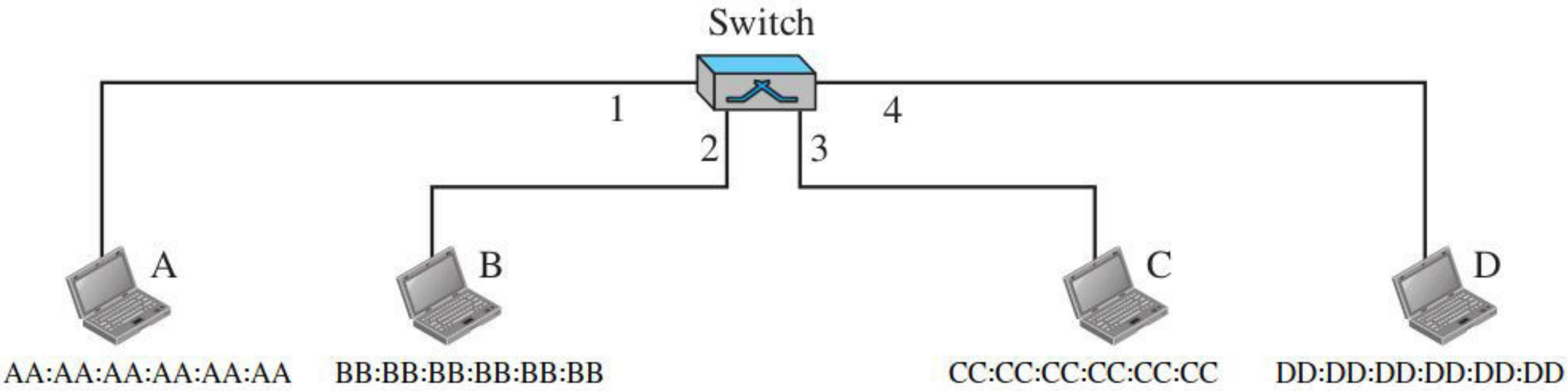
Switches: Transparent Switches

- Most switches are Transparent Switches, means the hosts (or stations) are completely unaware of the switch's existence.
- If a switch is added or deleted from the system, there in no need of reconfiguration in the stations.
- Transparent switches must do the following:
  - Frames must be forwarded from one station to another.
  - The forwarding table is automatically made by learning frame movements in the network.
  - Loops in the system must be prevented.

Switch...

- Switches that can work at network layer are called layer-3 switches.
- Switches do two major actions;
  - Forwarding
    - Send the packet only to the intended recipient.
  - Learning
    - Automatically update its switching table (MAC table) based on the incoming packets.
    - When the system is up, the MAC table will be empty and hence, the switch acts like a hub. ie, it just broadcasts the packets.
    - When the devices begin to transfer packets, the switch understands the MAC address of the devices connected to each port by reading the <u>source address</u> fields and updates the <u>MAC address table</u> and floods the packet to all other ports.
    - By continuous updation, the switch gets all the MAC addresses of the devices connected to the ports and does the forwarding.

Learning in switches



Address	Port

a. Original

Address	Port
AA:AA:AA:AA:AA:AA	1

b. After A sends a frame to D

Address	Port
AA:AA:AA:AA:AA:AA	1
DD:DD:DD:DD:DD:DD	4

c. After D sends a frame to B

Address	Port
AA:AA:AA:AA:AA:AA	1
DD:DD:DD:DD:DD:DD	4
BB:BB:BB:BB:BB:BB	2

d. After B sends a frame back to D

Address	Port
AA:AA:AA:AA:AA:AA	1
DD:DD:DD:DD:DD:DD	4
BB:BB:BB:BB:BB:BB	2
CC:CC:CC:CC:CC:CC	3

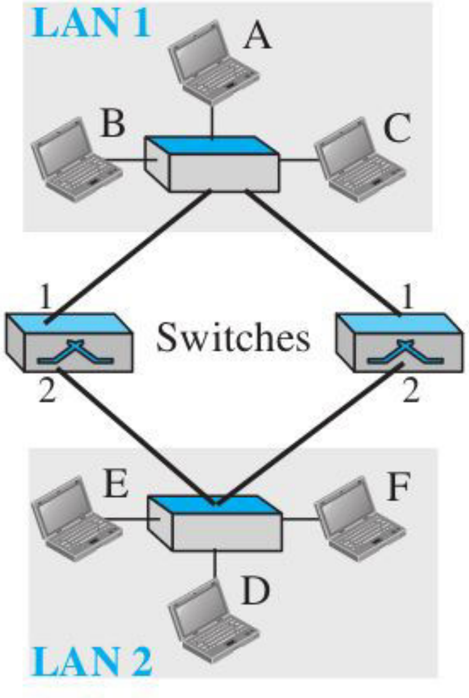
e. After C sends a frame to D

Cisco SG110D 110 Series 5-Port Unmanaged Network Switch

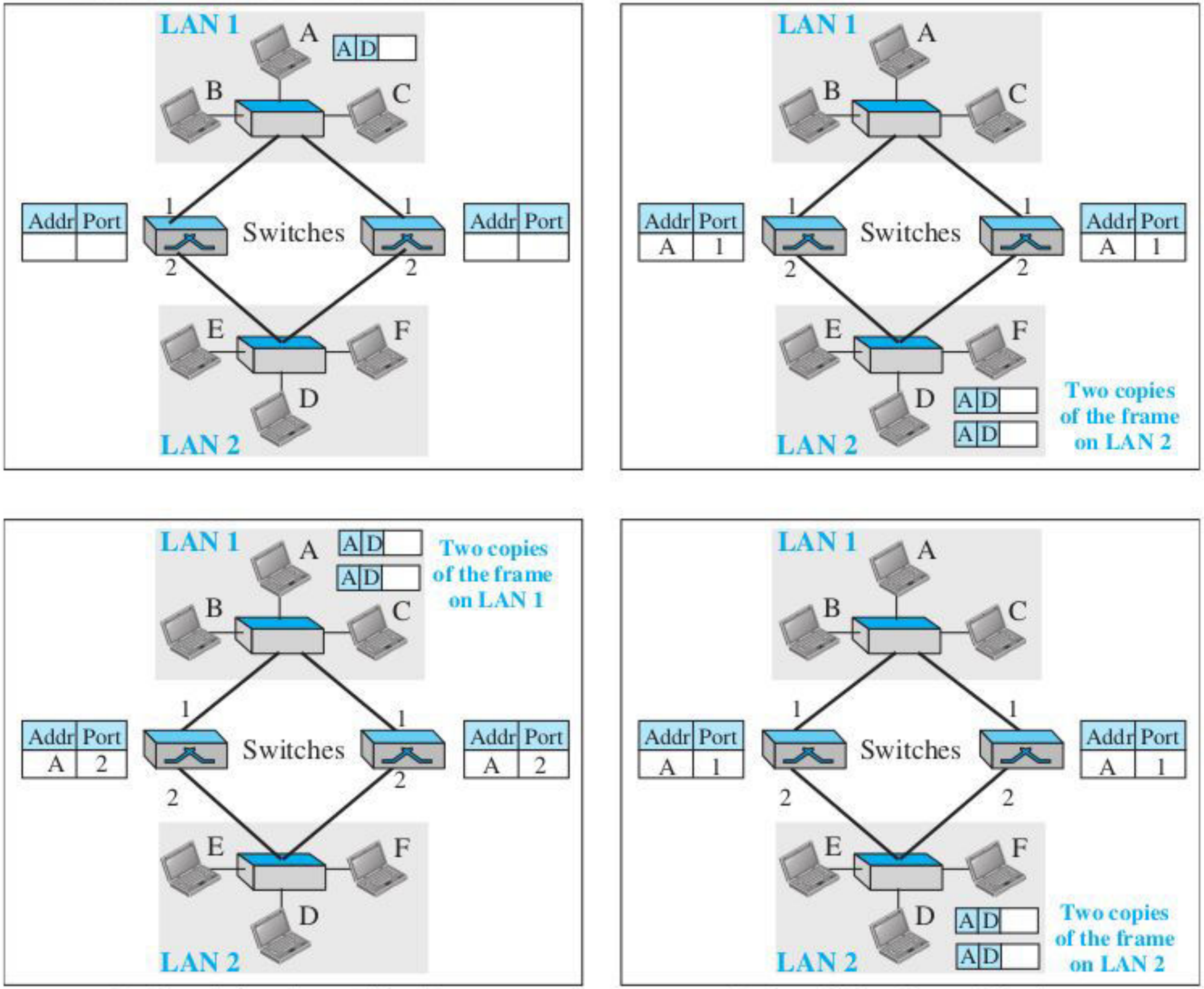
Cisco SG300-10 10-Port Gigabit Managed Switch

Loop Problem in Switches

- Sometimes administrators use redundant switches to make the network reliable (ie, if one switch fails the other will take over until the failed one is repaired or replaced).
- This can create loops in the system.
- Loops are created when two or more broadcasting LANs (those using hubs, for example) are connected by more than one switch.



Loop Problem



a. Station A sends a frame to station D

b. Both switches forward the frame

c. Both switches forward the frame

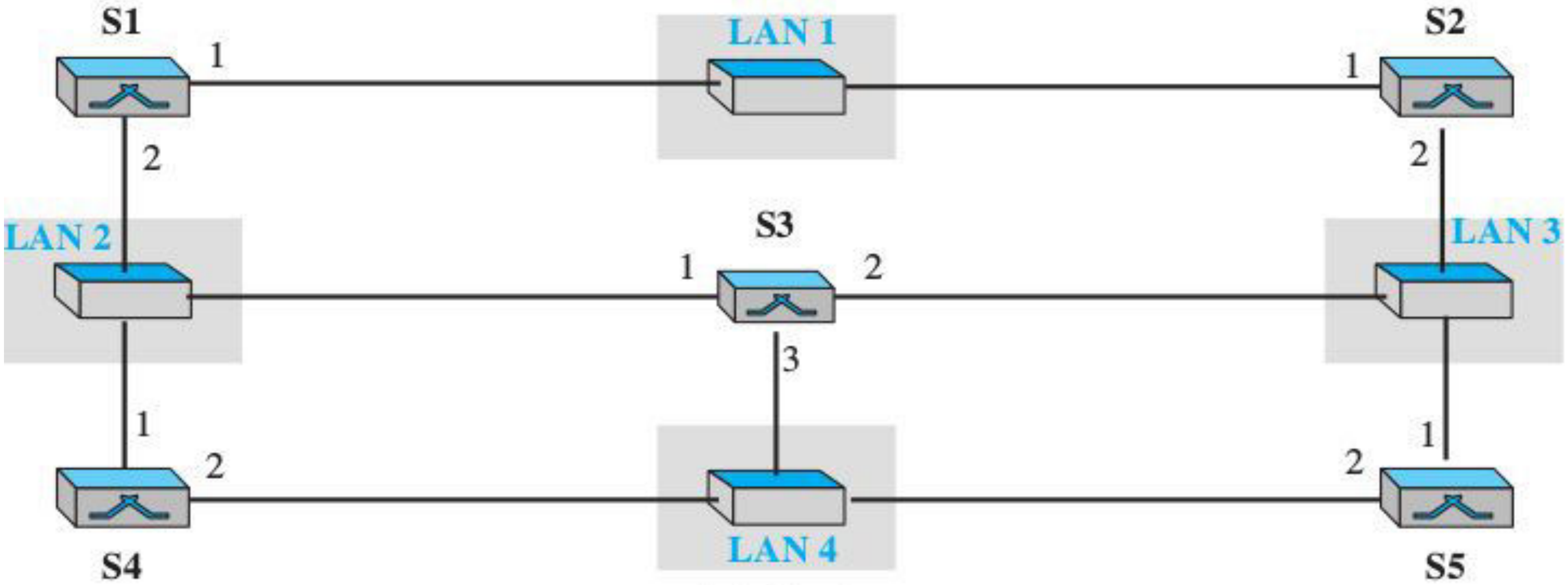
c. Both switches forward the frame

LAN 1
A
B
C
A D
Addr Port
1
2
Switches
LAN 2
E
F
D
Addr Port
A 1
Two copies of the frame on LAN 2
Addr Port
A 2
Two copies of the frame on LAN 1

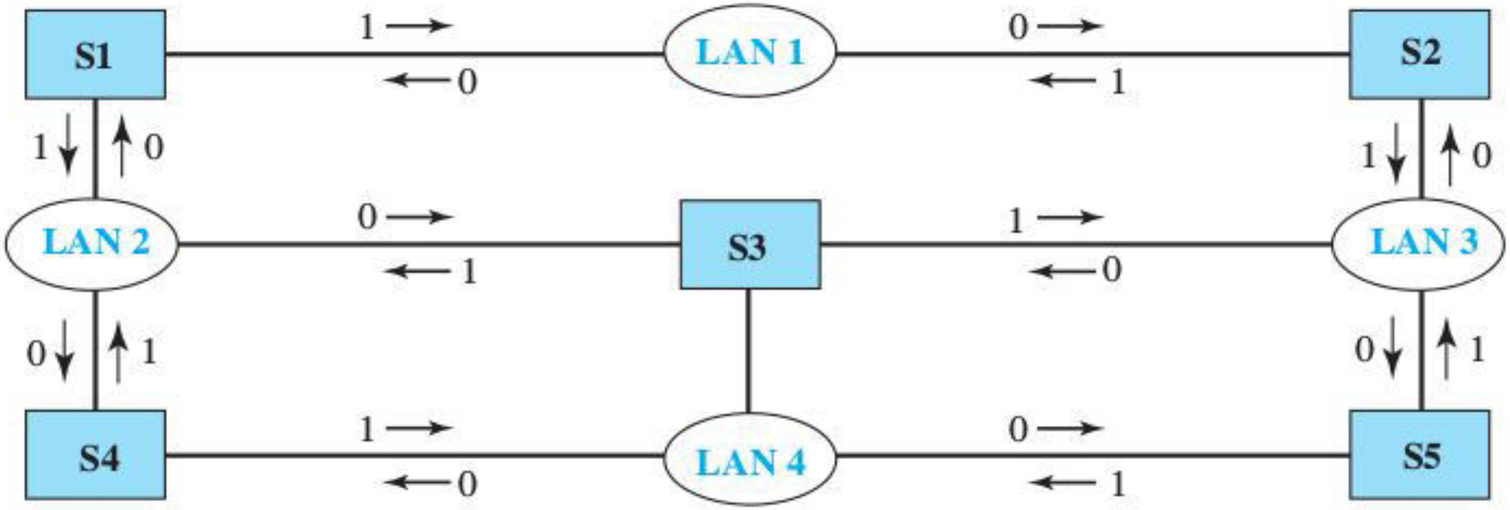
Solving the Loop Problem – Spanning Tree Algorithm

- Spanning tree is a graph which has no loop, but every node can be reached from every other node. And there is only one path between every two nodes.
- Algorithm:
  1. Each switch broadcasts its own unique ID so that all switches know which one has the smallest ID.
  2. The switch with the smallest ID is selected as the root switch of the tree.
  3. The algorithm tries to find the shortest path (a path with the shortest cost) from the root switch to every other switch or LAN. Shortest path can be found using algorithms such as Dijkstra algorithm.
  4. The combination of the shortest paths creates the shortest tree.
  5. Based on the spanning tree, we mark the ports that are part of it, the forwarding ports, which forward a frame that the switch receives. We also mark those ports that are not part of the spanning tree, the blocking ports, which block the frames received by the switch.

Solving the Loop Problem – Spanning Tree Algorithm

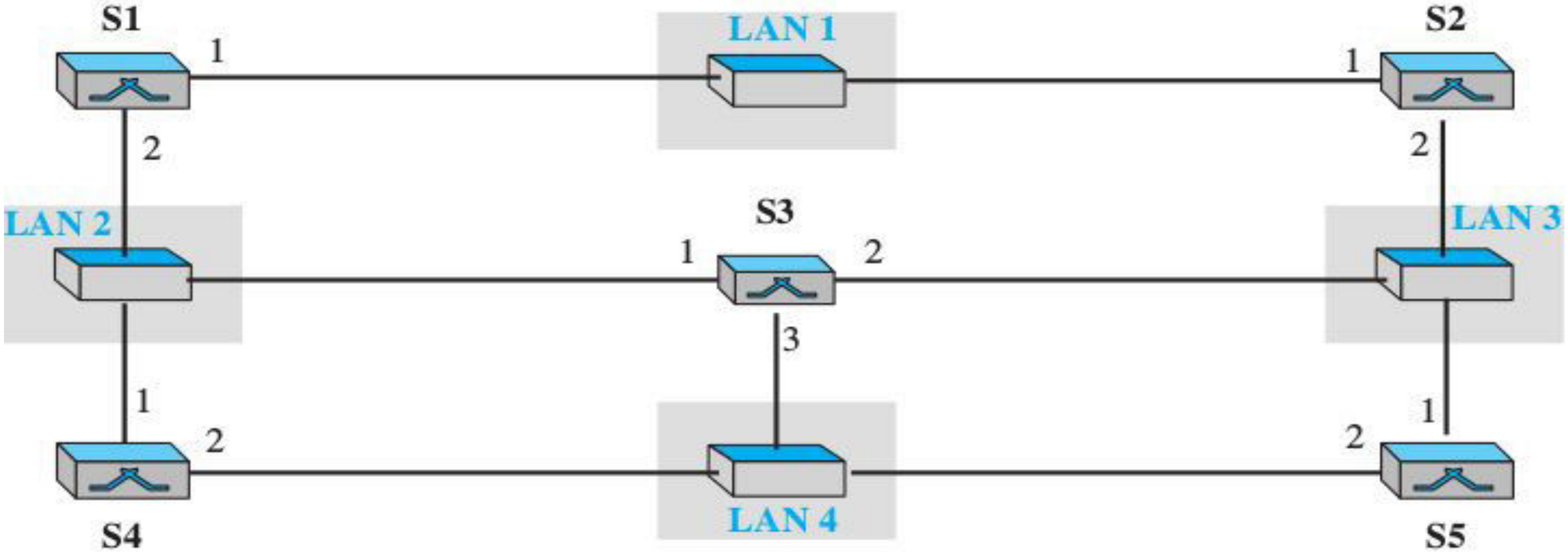


a. Actual system

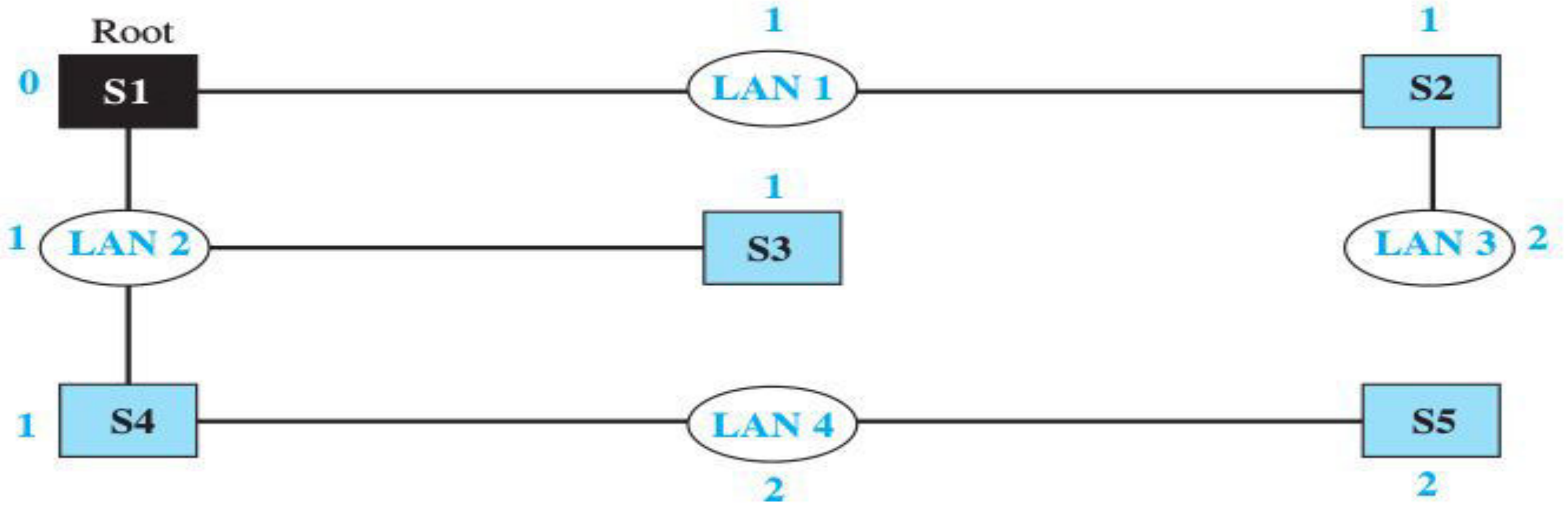


b. Graph representation with cost assigned to each arc

Solving the Loop Problem – Spanning Tree Algorithm



a. Actual system



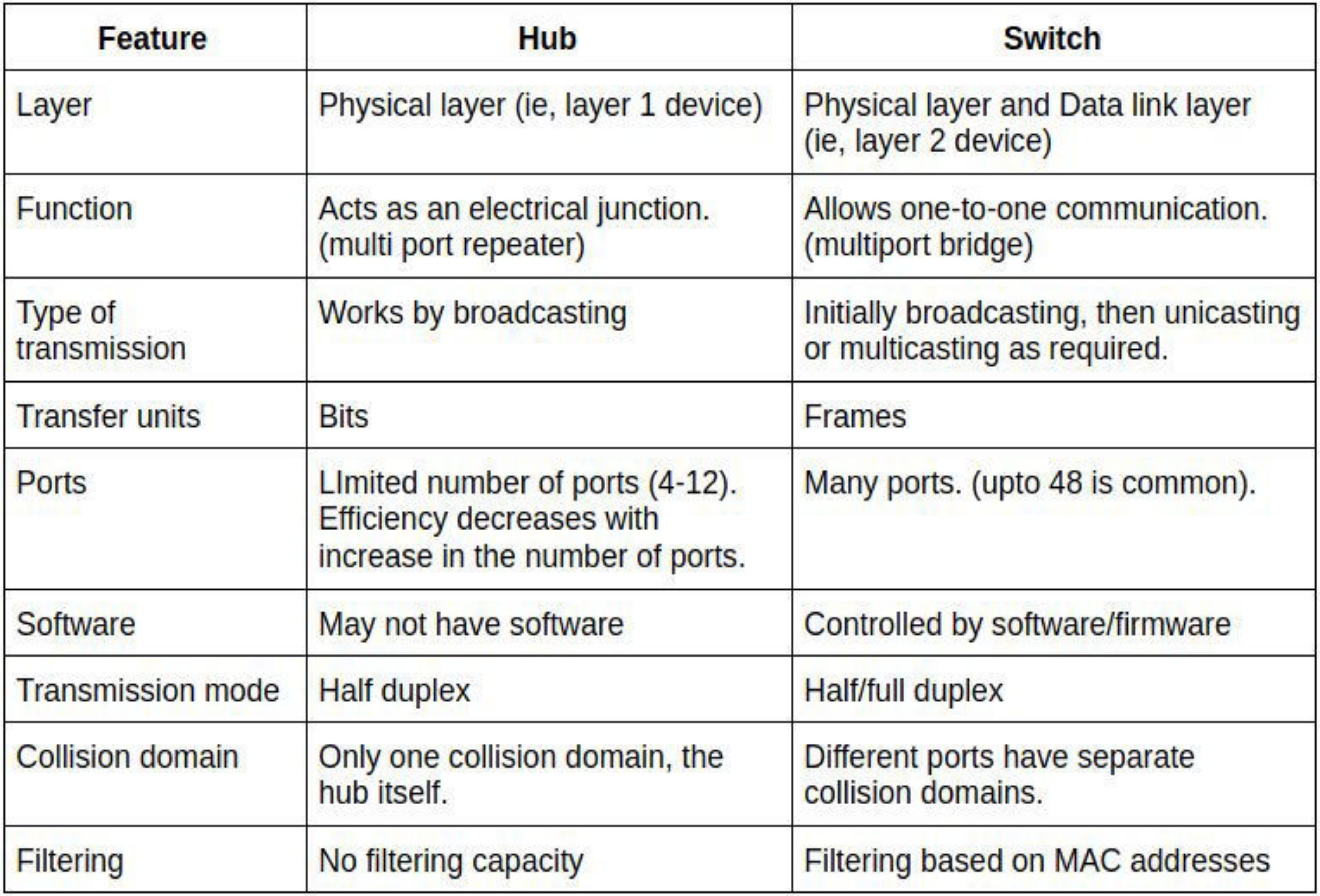
Spanning Tree

Advantages of Switches over Hub

- Collision Elimination
    - In a switched LAN, there is no need for carrier sensing and collision detection; each host can transmit at any time.

- Connecting Heterogeneous Devices
    - A link-layer switch can connect devices that use different protocols at the physical layer (data rates) and different transmission media as long as the format of the frame at the data-link layer does not change.

Hub vs Switch



Feature	Hub	Switch
Layer	Physical layer (ie, layer 1 device)	Physical layer and Data link layer (ie, layer 2 device)
Function	Acts as an electrical junction. (multi port repeater)	Allows one-to-one communication. (multiport bridge)
Type of transmission	Works by broadcasting	Initially broadcasting, then unicasting or multicasting as required.
Transfer units	Bits	Frames
Ports	LImited number of ports (4-12). Efficiency decreases with increase in the number of ports.	Many ports. (upto 48 is common).
Software	May not have software	Controlled by software/firmware
Transmission mode	Half duplex	Half/full duplex
Collision domain	Only one collision domain, the hub itself.	Different ports have separate collision domains.
Filtering	No filtering capacity	Filtering based on MAC addresses

Router

- Connects two or more different networks.
- Layer 3 device.
- Works at network layer, data link layer and physical layer.
    - At physical layer: regenerates the bits
    - At data link layer: checks the MAC addresses
    - At network layer: checks the network-layer addresses.
- Connects a LAN to another, a LAN to a WAN or a WAN to another.
- Thus router is an internetworking device.
- Reads the IP address in the packet and with the help of routing tables, forwards the packet to the destination.
- May have different types of physical layer connections such as copper cables, fiber optic, or wireless transmission, and can support different network layer standards.

Router...

- 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.

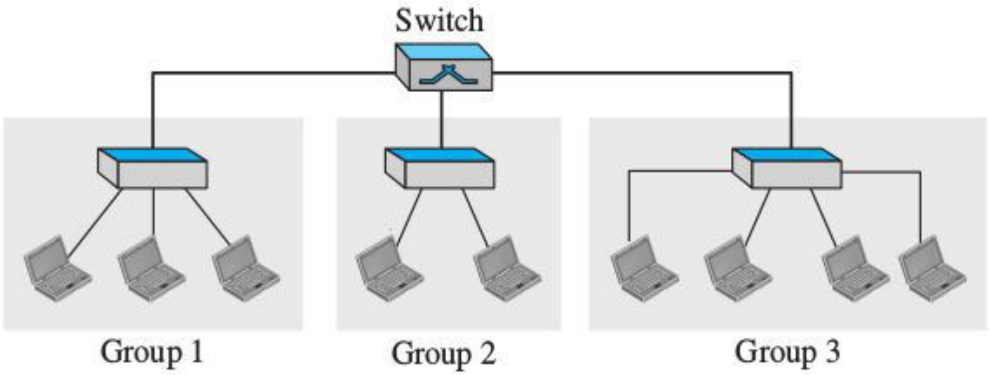
3) A router changes the link-layer address of the packet (both source and destination) when it forwards the packet.

Cisco Routers

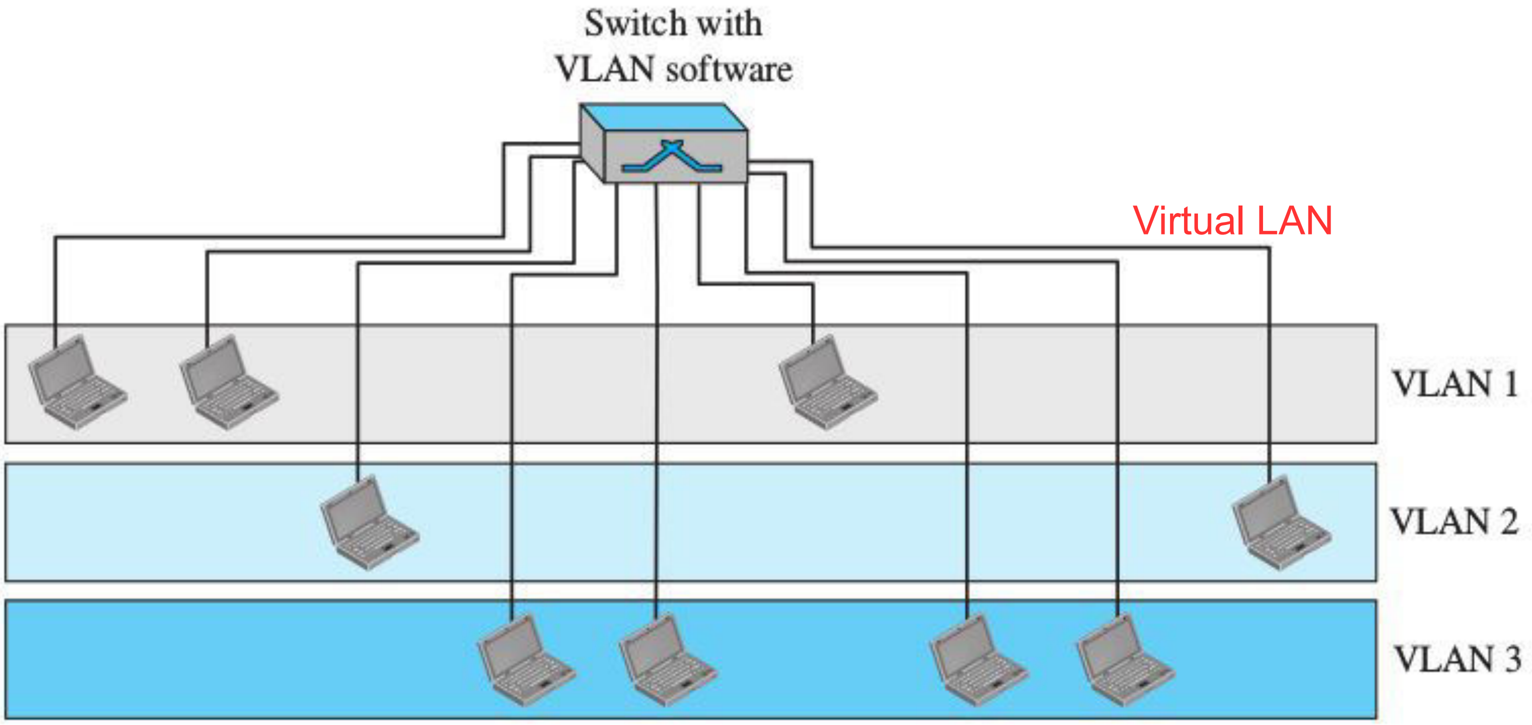
Virtual LAN (VLAN)

- Divides a LAN into several logical segments, instead of physical segments.
- Each VLAN can be a work group in the organization.
- The group membership in the VLAN is defined in software, not by hardware.
- The stations are seemed to be the part of a single LAN, but they are virtually separated and comes under different Virtual LANs, without changing the physical connections or host configurations.
- A VLAN might comprise a subset of the ports on a single switch or subsets of ports on multiple switches.
- By default, systems on one VLAN don't see the traffic associated with systems on other VLANs on the same network.
- Thus VLAN creates different broadcast domains.

Virtual LAN (VLAN)

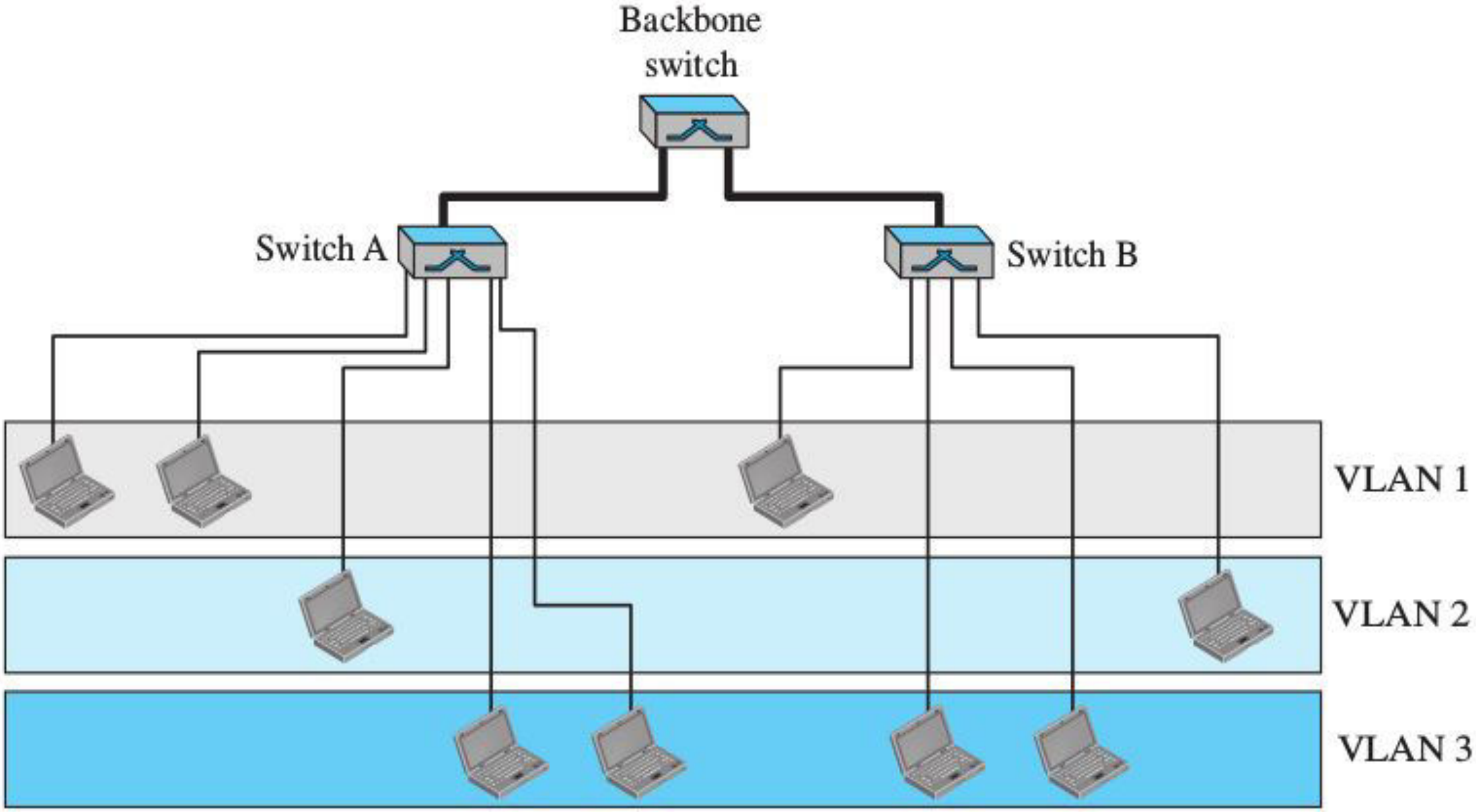


Ordinary LAN



Virtual LAN

Virtual LAN (VLAN)



VLAN with three switches.
Switch A and switch B can be in different buildings.

VLAN Membership

- Vendors use different characteristics such as interface numbers (port numbers), MAC addresses, IP addresses, IP multicast addresses, or a combination of two or more of these to group the VLANs.
- Interface Numbers: Some VLAN vendors use switch interface numbers as a membership characteristic.
    - Eg: Ports 2,6,8 for VLAN1, Ports 1,3,7 for VLAN2, etc.
- MAC Addresses: Some VLAN vendors use the 48-bit MAC address as a membership characteristic.
    - Eg: 00:A0:C9:14:C8:29 and 00:1C:B3:09:85:15 belong to VLAN1
- IP Addresses
    - Eg: 192.168.1.2 and 192.168.1.45 belong to VLAN1
- Multicast IP Addresses
- Combination: The software available from some vendors allows all these characteristics to be combined.

VLAN Configuration

- Three ways: manually, automatically and semi automatically.

1) Manual Configuration: The administrator uses the VLAN software to setup and maintain the VLAN manually.
2) 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.
3) Semi Automatic Configuration: A semiautomatic configuration is somewhere between a manual configuration and an automatic configuration. Usually, the initializing is done manually, with migrations done automatically.

- **Advantages:**
  - Cost and Time Reduction
  - Creating Virtual Work Groups
  - Security

End of Module 1

Diploma in Computer Hardware Engineering

COMPUTER NETWORKS

(5151 Rev 2015)

Module 2 - Part 1

Presenter: Sreejesh NG
Lecturer in Computer Hardware Engineering
Government Polytechnic College, Cherthala

Ref: Data Communications and Networking, 5E by Forouzan

Course General Outcomes:

Sl.	G.O	On completion of this course the student will be able :
1	1	To Understand the concept of TCP/IP Protocol
2	1	To Understand the concept of Network Layer
3	1	To Understand the concept of Transport Layer
4	1	To Understand the concept of Application Layer

Specific Outcomes:

2.1 Understand Network Layer

2.1.1 Explain Network layer services
2.1.2 Illustrate network layer performance
2.1.3 Describe IPV4 addresses
2.1.4 Define DHCP
2.1.5 Explain Internet Protocol
2.1.6 State security of IPV4 datagram
2.1.7 Describe routing algorithms
2.1.8 Differentiate between unicasting, multicasting, and broadcasting

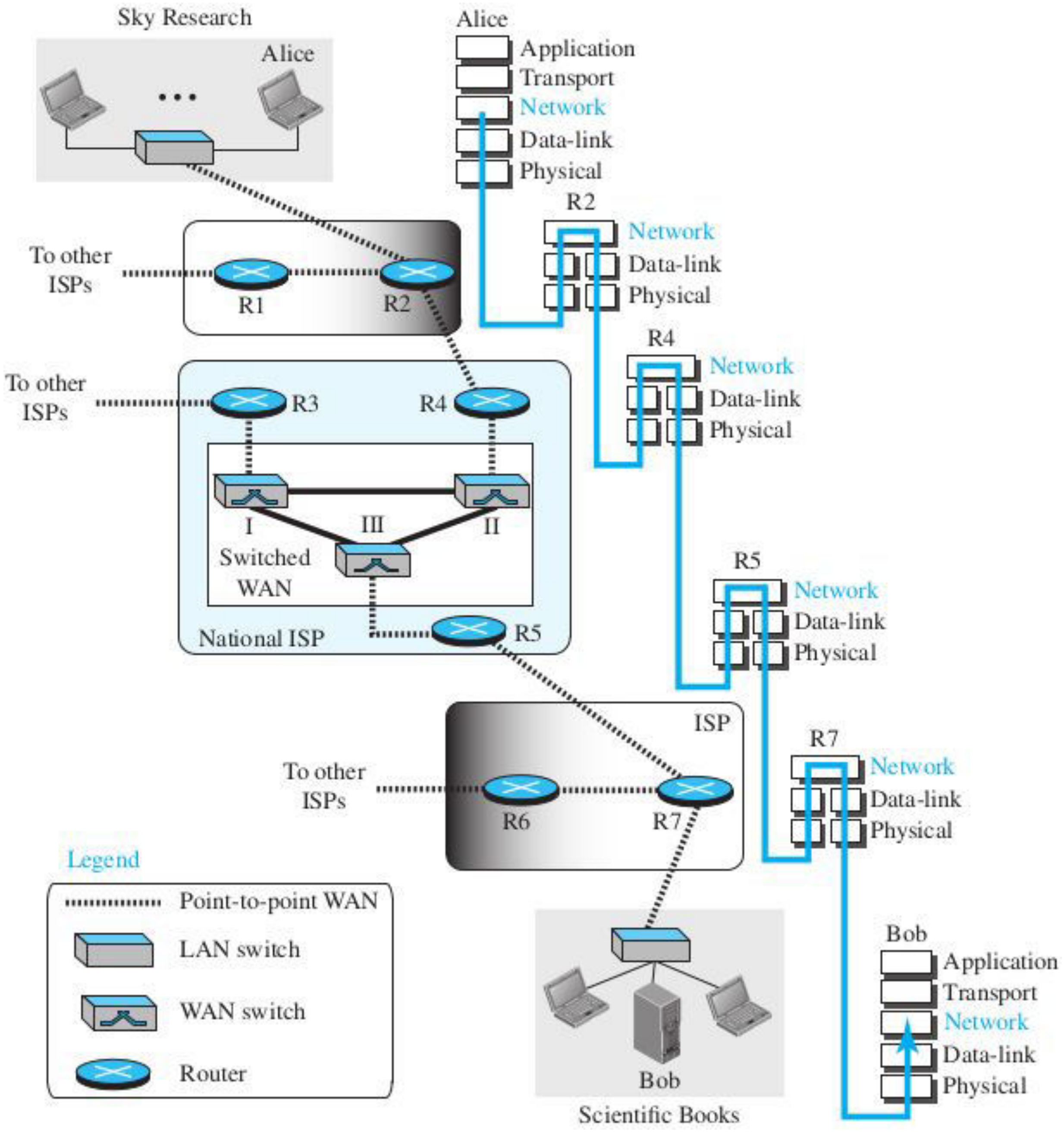
MODULE 2 - NETWORK LAYER

- Network layer services – Packetizing, routing and forwarding, other services
- Performance – delay, throughput, packet loss, congestion control
- IPV4 address – address space, classful addressing, classless addressing, subnetting
- DHCP
- Internet protocol (IP) – datagram format, fragmentation
- IPV4 datagram security
- Routing algorithms
    - Distance-vector (DV)
    - Link-state (LS)
    - Path vector (PV)
- Unicasting, multicasting, broadcasting

Ref: Data Communications and Networking - Behrouz A. Forouzan -
McGraw Hill Edn.- Fifth Edition

Network Layer

- End-to-end delivery



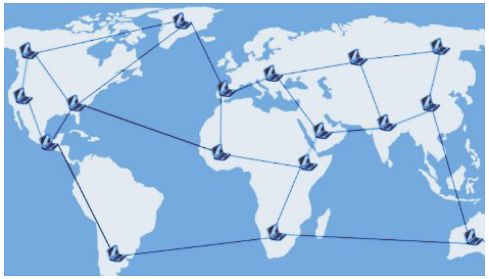
Network Layer Services

1. Packetizing
2. Routing and Forwarding
3. Other Services
    - *Expected* services are
        - Error Control
        - Flow Control
        - Congestion Control
        - Quality of Service
        - Security

1. **Packetizing**

- *Packetizing*: Encapsulating the payload (data received from upper layer) in a network-layer packet at the source and decapsulating the payload from the network-layer packet at the destination.
- Carry a payload from the source to the destination without changing it or using it.
- If the packet is too large, it is fragmented by the intermediate routers.
- All the fragments have the same header as the original (especially source and destination addresses), with small changes to specify fragments.
- The fragments are reassembled at the destination.

2. Routing and Forwarding

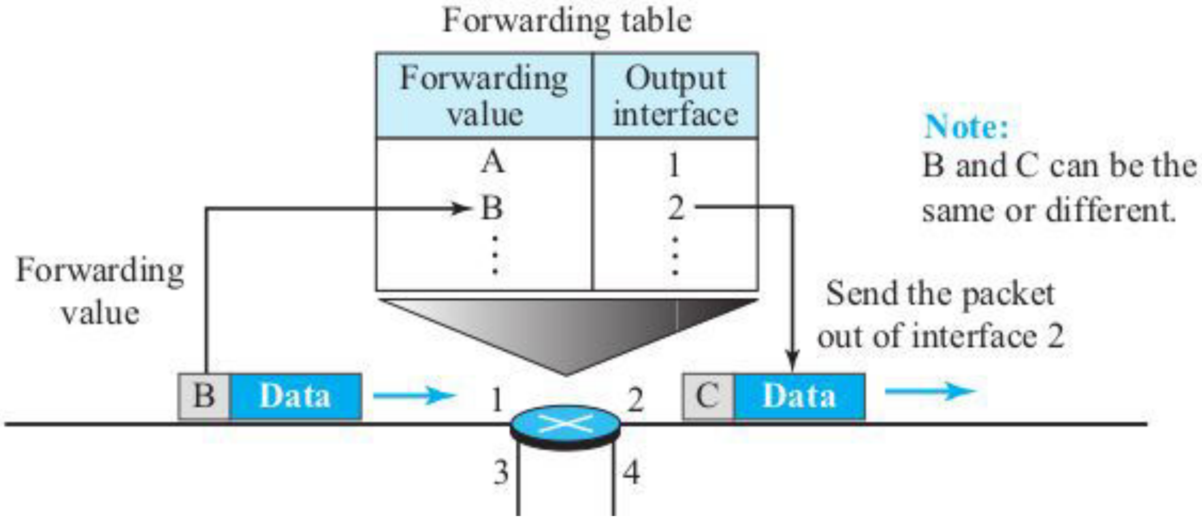


- *Routing*:
  - In a large network, there will be a number of routes in between the source and destination devices.
  - Network layer finds the best route based on some specific strategies (load, bandwidth, hops, etc.)
  - The strategies are mostly defined by the *Routing protocols*.
  - The strategies are used to create a decision-making table, called the *routing table* for each router.

Routing is applying strategies and running some routing protocols to create the decision-making tables for each router.

2. **Routing and Forwarding...**

- *Forwarding:*
  - The action applied by a router when a packet arrives at one of its interfaces.
  - Forwarding is done with the help of *Forwarding table* or *Routing table.*
  - On receiving a packet at an interface, the router reads the destination address/label in the incoming packet, find the output interface number from the table and forwards the packet.
  - The packet forwarding can be;
    - to another attached network (in unicast routing) *or*
    - to some attached networks (in multicast routing).



Forwarding table

Forwarding value	Output interface
A	1
B	2
⋮	⋮

Note:
B and C can be the same or different.
Forwarding value
Send the packet out of interface 2
B
Data
1
2
C
Data
3
4

3. **Other Services**

- Other services expected from this layer are;
  - Error Control
  - Flow Control
  - Congestion Control
  - Quality of Service
  - Security

3. **Other Services**

3.1 Error Control

- Packets in the network layer may be fragmented at routers, which makes error checking at this layer inefficient.
- A checksum field in the datagram controls any corruption in the header, but not in the whole datagram.

Internet uses an auxiliary protocol, ICMP, that provides some kind of error control if the datagram is discarded or has some unknown information in the header.

3. Other Services..

3.2 Flow Control

- Network layer does not directly provide flow control, because;

  1. To make the network layer at the receiver is so simple that it may rarely be overwhelmed.
  2. The upper layers can implement buffers to receive data from the network layer.
  3. Flow control is provided for most of the upper-layer protocols, so another level of flow control makes the network layer more complicated and less efficient.

3. **Other Services..**

3.3 Congestion Control

- Congestion: Too many datagrams are present in an area of the internet.
- Happens if the number of datagrams sent by source computers is beyond the capacity of the network or routers.
- Hence, some routers may drop some of the datagrams.
- Due to the error control mechanism at the upper layers, the sender may send duplicates of the lost packets.
- If the congestion continues, sometimes the system collapses and no datagrams are delivered.

3. **Other Services..**

3.4 Quality of Service (QoS)

- To keep the network layer simple and untouched, QoS is implemented in upper layers.

3.5 Security

- The network layer was created with no security provision.
- To make the network layer secure, a connection oriented virtual layer service (called IPSec) is created.

NETWORK LAYER PERFORMANCE

- Can be measured in terms of;
  - Delay
  - Throughput
  - Packet loss
- Congestion control also improves performance.

NETWORK LAYER PERFORMANCE

1. **Delay**
   - A packet, from its source to its destination, encounters delays.
   - Can be subdivided into;
     1. Transmission delay
     2. Propagation delay
     3. Processing delay
     4. Queuing delay.
     5. Total Delay

NETWORK LAYER PERFORMANCE

1.(i) Transmission Delay

- A sender puts the bits in a packet on the line one by one.
- If the first bit of the packet is put on the line at time t_1 and the last bit is put on the line at time t_2 , transmission delay of the packet is $(t_2 - t_1)$.
  - Delay_{tr} = (Packet length) / (Transmission rate).
- The longer the packet, the longer the transmission delay.
- Eg: For a Fast Ethernet LAN (100 million bits/sec) with a packet size of 10,000 bits, the transmission delay is (10,000)/(100,000,000) or 100 microseconds.

NETWORK LAYER PERFORMANCE

1.(ii) Propagation Delay

- The time taken for a bit to travel from point A to point B in the transmission media.
  - Delay_{pg} = (Distance) / (Propagation speed).
- Eg: If the distance of a cable link in a point-to-point WAN is 2000 meters and the propagation speed of the bits in the cable is 2×10^8 meters/second, then the propagation delay is 10 microseconds.

NETWORK LAYER PERFORMANCE

1.(iii) Processing Delay

- The time required for a router or a destination host to receive a packet from its input port, remove the header, perform an error detection procedure, and deliver the packet to the output port (in the case of a router) or deliver the packet to the upper-layer protocol (in the case of the destination host).
- May be different for each packet, but normally is calculated as an average.
  - ○ Delay_{pr} = Time required to process a packet in a router or a destination host

NETWORK LAYER PERFORMANCE

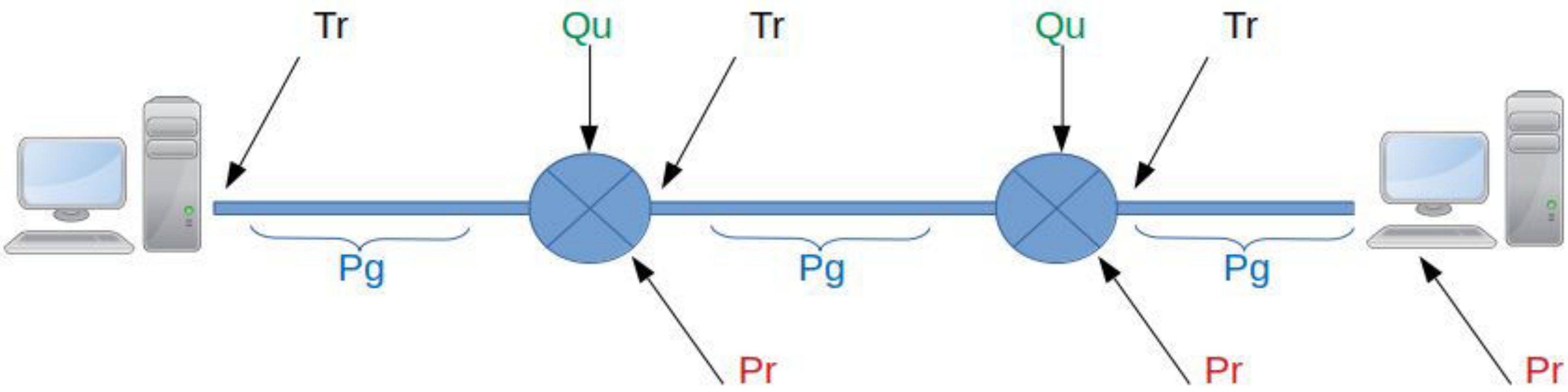
1.(iv) Queuing Delay

- Happen in a router.
- A router has an input queue connected to each of its input ports to store packets waiting to be processed.
- Also an output queue connected to each of its output ports to store packets waiting to be transmitted.
- Queuing delay is the time a packet waits in the input queue and output queue of a router.
    - Delay_{qu} = The time a packet waits in input and output queues in a router

NETWORK LAYER PERFORMANCE

1.(v) Total Delay

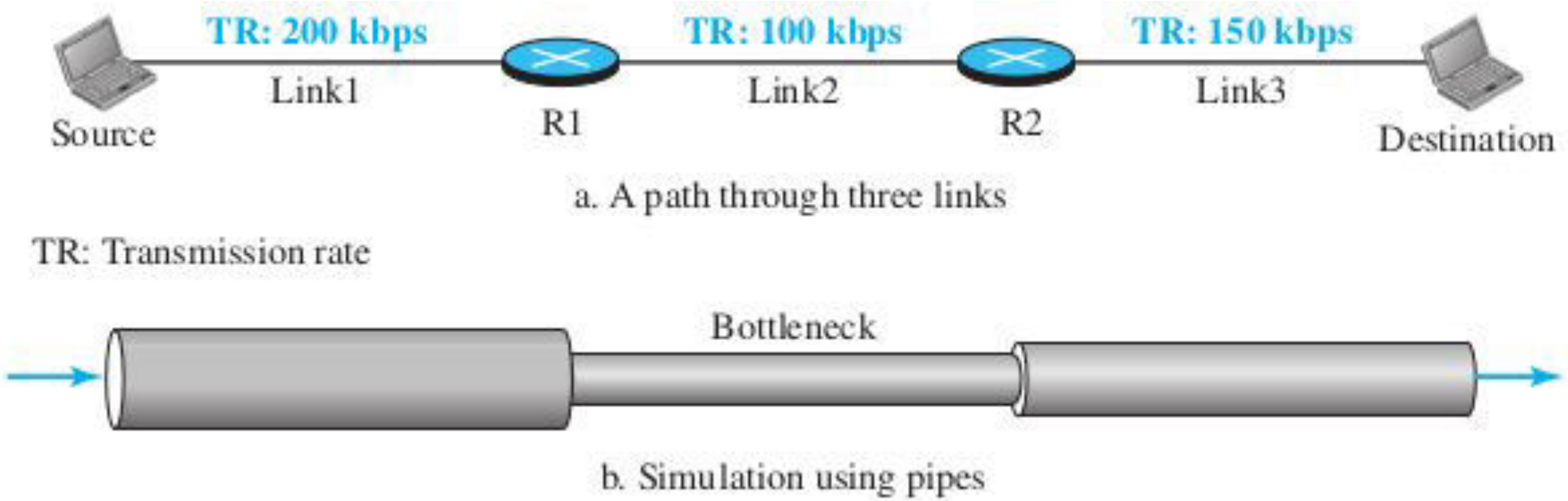
- The total delay (source-to-destination delay) a packet encounters is the sum of all the above delays in all the devices and routers that a packet transfers between the source and destination, including them.
- If there are **n** routers in between;
  - Total delay = (n + 1) (Delay$_{tr}$ + Delay$_{pg}$ + Delay$_{pr}$) + (n) (Delay$_{qu}$)



NETWORK LAYER PERFORMANCE

2. Throughput

- It is the number of bits passing through the point in a second
- Same as transmission rate of data at that point.
- It varies from link to link.
- Depends on the link with the lowest transmission rate.
- In a path with n links in series,
  - Throughput = minimum {TR_1 , TR_2 , . . . TR_n }.
- In the below figure, the average throughput is 100kbps.



a. A path through three links

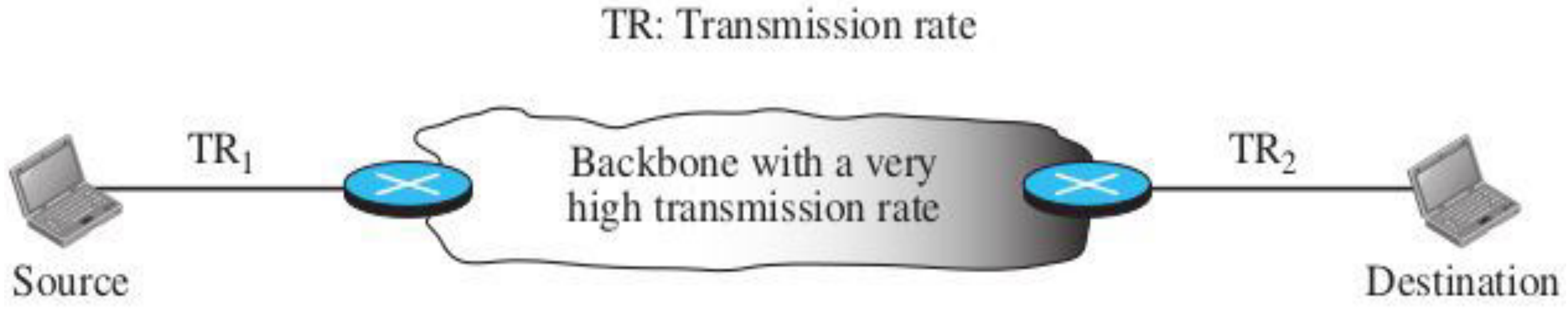
TR: Transmission rate

b. Simulation using pipes

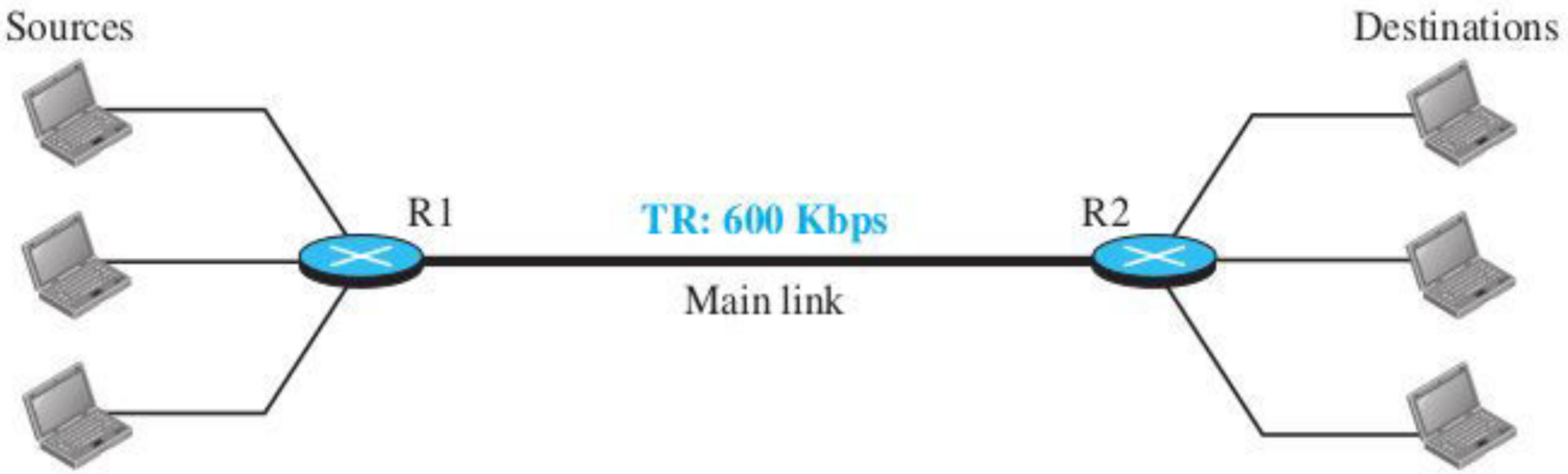
NETWORK LAYER PERFORMANCE

2. Throughput…

- A path through the Internet backbone: The throughput depends on TR1 and TR2.



- Effect of throughput in shared links: The throughput depends on the shared devices. Here the rate may be 200kbps.



NETWORK LAYER PERFORMANCE

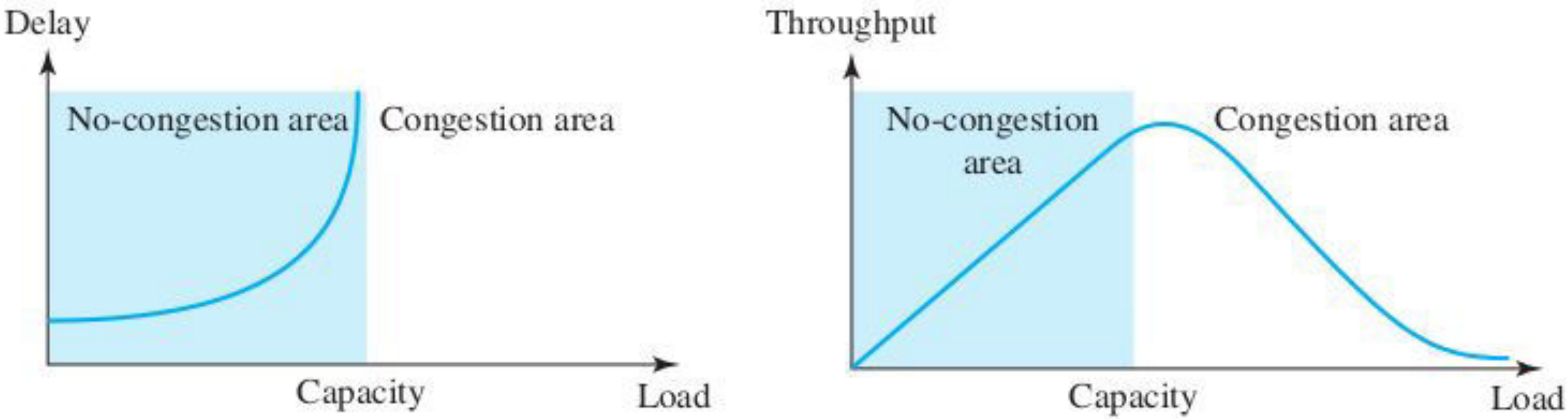
3. Packet loss

- Limited input buffers in routers.
- If this buffer goes full, other incoming packets will be discarded.
- These packets are to be resent, this in turn affects the network.
- Packet loss represents the number of packets dropped.

NETWORK LAYER PERFORMANCE

Congestion Control

- Proper congestion control improves network performance.
- Congestion is related to <u>delay</u> and <u>throughput</u>.



a. Delay as a function of load

b. Throughput as a function of load

NETWORK LAYER PERFORMANCE

Congestion Control...

- Congestion control refers to techniques and mechanisms that can either prevent congestion before it happens or remove congestion after it has happened.
- Two broad categories:
  - ■ open-loop congestion control (prevention)
  - ■ closed-loop congestion control (removal).

NETWORK LAYER PERFORMANCE

Open-Loop Congestion Control

- Policies are applied to prevent congestion before it happens.
- Congestion control is handled by either the source or the destination.
- Policies are;

    i. Retransmission Policy
    ii. Window Policy
    iii. Acknowledgment Policy
    iv. Discarding Policy
    v. Admission Policy

NETWORK LAYER PERFORMANCE

Open-Loop Congestion Control

1. Retransmission Policy
   - If the sender feels that a sent packet is lost or corrupted, the packet needs to be retransmitted.
   - Retransmission increases congestion.
   - The retransmission policy and the retransmission timers must be designed to optimize efficiency and at the same time prevent congestion.
2. Window Policy
   - The Selective Repeat window is better than the Go-Back-N window for congestion control.
   - The Selective Repeat window sends the specific packets that have been lost or corrupted.
3. Acknowledgment Policy
   - If the receiver does not acknowledge every packet it receives, it may slow down the sender and help prevent congestion.

NETWORK LAYER PERFORMANCE

Open-Loop Congestion Control

4. Discarding Policy
   - A good discarding policy by the routers may prevent congestion and at the same time may not harm the integrity of the transmission.
   - Eg: in audio transmission, the policy is to discard less sensitive packets when congestion is likely to happen.
5. Admission Policy
   - Prevent congestion in virtual-circuit networks.
   - Switches in a flow first check the resource requirement of a flow before admitting it to the network.
   - A router can deny establishing a virtual-circuit connection if there is congestion in the network or if there is a possibility of future congestion.

NETWORK LAYER PERFORMANCE

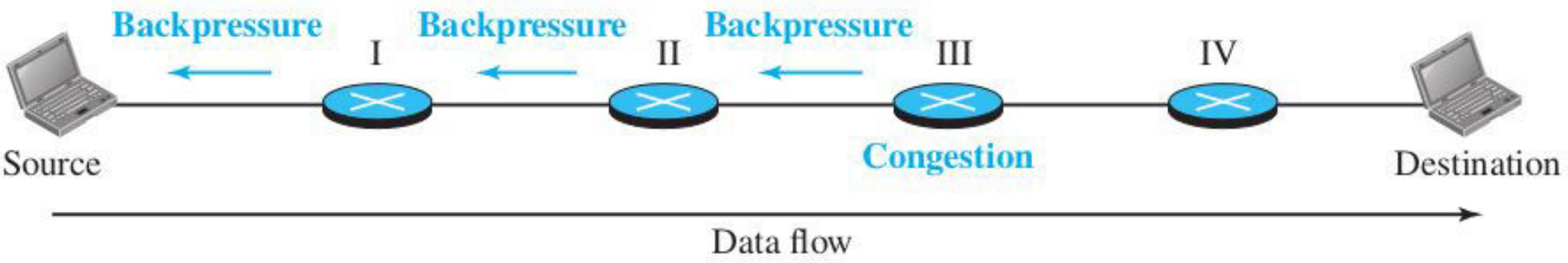
Closed-Loop Congestion Control

- Policies are applied to alleviate congestion after it happens.
- Policies are;

  i. Backpressure
  ii. Choke Packet
  iii. Implicit Signaling
  iv. Explicit Signaling

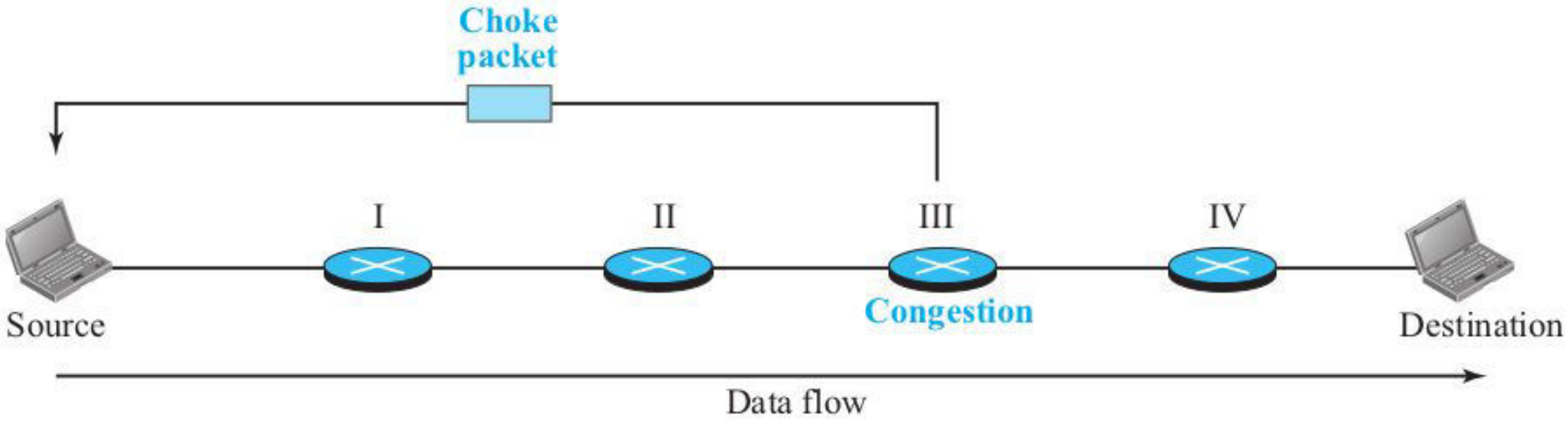
NETWORK LAYER PERFORMANCE

1. Backpressure
   - ○ A node-to-node congestion control that starts with a node and propagates, in the opposite direction of data flow, to the source.
   - ○ Method:
     - ■ A congested node stops receiving data from the immediate upstream node or nodes.
     - ■ This may cause the upstream node or nodes to become congested, they, in turn, reject data from their upstream node or nodes, and so on.
   - ○ Only used in Virtual Circuit network in which each node knows the upstream node.



NETWORK LAYER PERFORMANCE

2. Choke Packet
    - A choke packet is a packet sent by a node to the source to inform it of congestion.
    - When a router in the Internet is overwhelmed with IP datagrams, it may discard some of them, but it informs the source host directly, using a *source quench ICMP message*.



NETWORK LAYER PERFORMANCE

3. Implicit Signaling
   - There is no communication between the congested node or nodes and the source.
   - The source guesses that there is congestion somewhere in the network from other symptoms.
     - Eg: Delay in acknowledgement, or no acknowledgement for some time.
   - Thus the source slows down.

NETWORK LAYER PERFORMANCE

4. Explicit Signaling
   - The node that experiences congestion can explicitly send a signal to the source or destination.
   - Here the signal is included in the packets that carry data.
   - Can occur in either the forward or the backward direction.
   - Mostly in ATM (Asynchronous Transfer Mode) networks.

End of Part 1

Thank You

Diploma in Computer Hardware Engineering

COMPUTER NETWORKS

(5151 Rev 2015)

Module 2 - Part 2

Presenter: Sreejesh NG
Lecturer in Computer Hardware Engineering
Government Polytechnic College, Cherthala

Ref: Data Communications and Networking, 5E by Forouzan

Course General Outcomes:

Sl.	G.O	On completion of this course the student will be able :
1	1	To Understand the concept of TCP/IP Protocol
2	1	To Understand the concept of Network Layer
3	1	To Understand the concept of Transport Layer
4	1	To Understand the concept of Application Layer

Specific Outcomes:

2.1 Understand Network Layer

2.1.1 Explain Network layer services
2.1.2 Illustrate network layer performance
2.1.3 Describe IPV4 addresses
2.1.4 Define DHCP
2.1.5 Explain Internet Protocol
2.1.6 State security of IPV4 datagram
2.1.7 Describe routing algorithms
2.1.8 Differentiate between unicasting, multicasting, and broadcasting

MODULE 2 - NETWORK LAYER

- Network layer services – Packetizing, routing and forwarding, other services
- Performance – delay, throughput, packet loss, congestion control
- IPV4 address – address space, classful addressing, classless addressing, subnetting
- DHCP
- Internet protocol (IP) – datagram format, fragmentation
- IPV4 datagram security
- Routing algorithms
    - Distance-vector (DV)
    - Link-state (LS)
    - Path vector (PV)
- Unicasting, multicasting, broadcasting

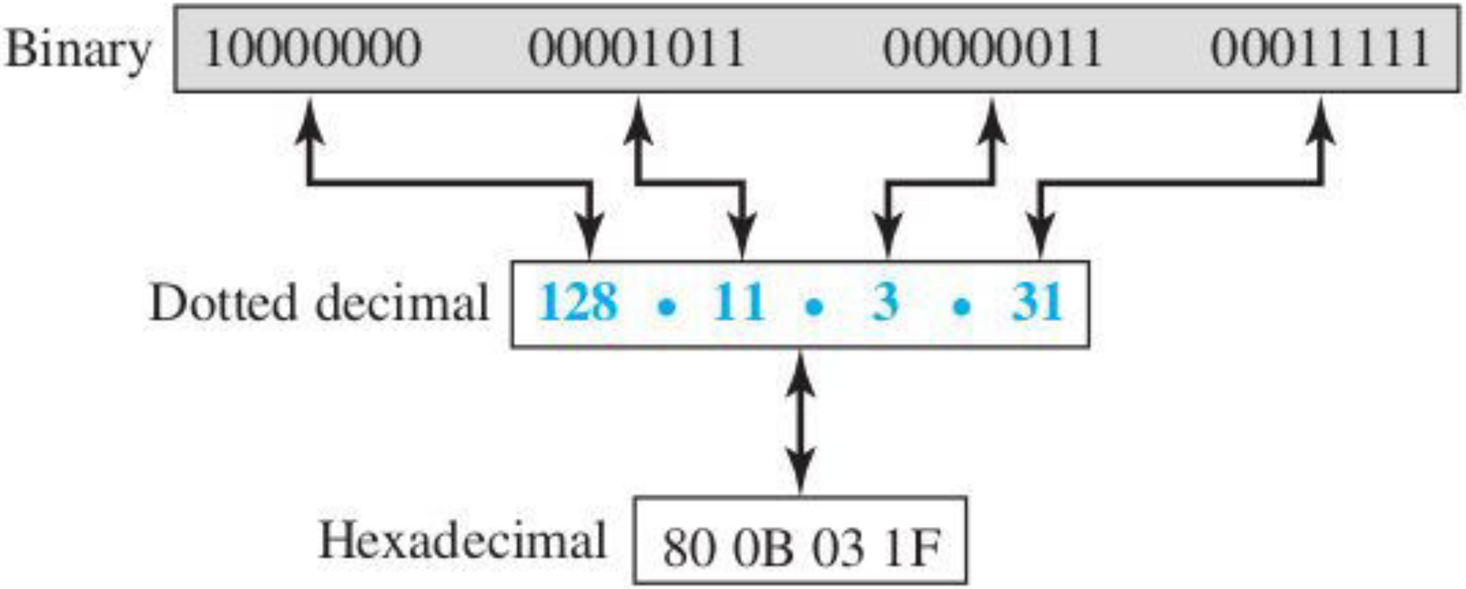
Ref: Data Communications and Networking - Behrouz A. Forouzan -
McGraw Hill Edn.- Fifth Edition

IPV4 ADDRESS

- The identifier used in the internet layer (network layer) of the TCP/IP protocol suite to identify the connection of each device to the Internet is called the Internet address or Internet Protocol address (IP address).
- There are two versions of IP addresses: IPv4 (32 bit) and IPv6 (128 bit).
- IPv4 address is a 32-bit address that uniquely and universally defines the connection of a host or a router to the Internet.
- The IP address is the address of the connection, not the host or the router, because if the device is moved to another network, the IP address may be changed.
- If a device has two connections to the Internet, via two networks, it has two IPv4 addresses.

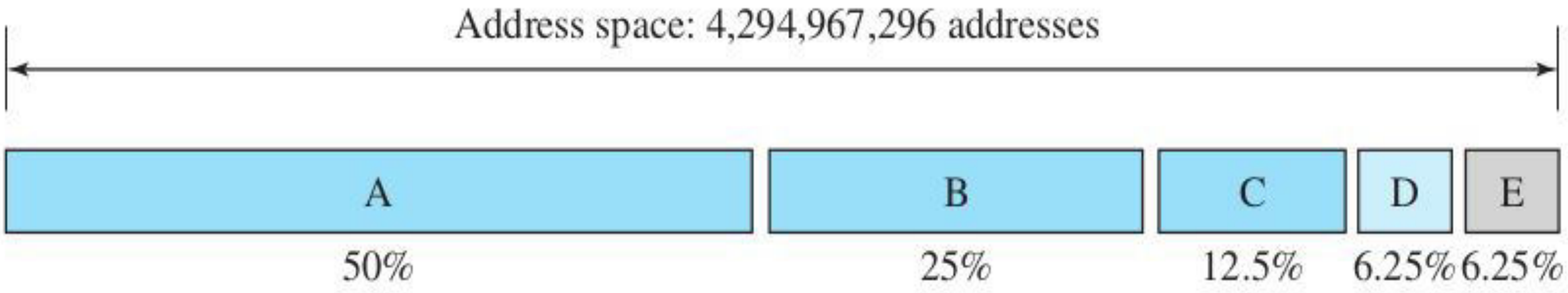
IPV4 ADDRESS

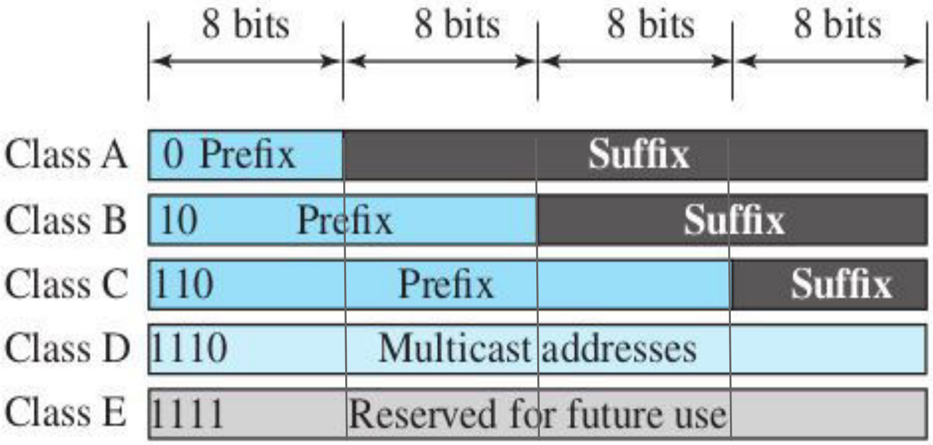
- **Address Space**
  - An address space is the total number of addresses used by the protocol.
  - IPv4 uses 32-bit addresses, which means that the address space is 2^{32} or 4,294,967,296 (more than four billion).
- **Notation**

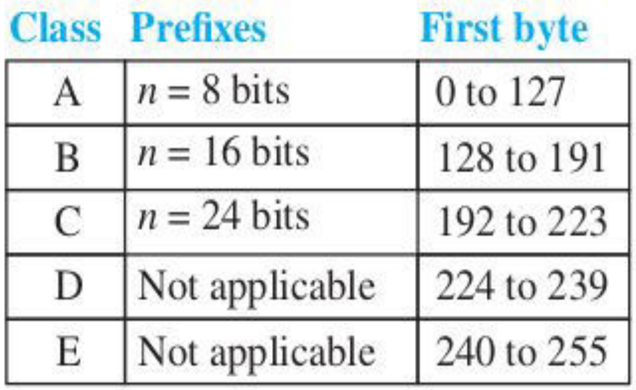


IPV4 ADDRESS

- **Classful Addressing**
  - The whole address space was divided into 5 classes: Class A, B, C, D & E.
  - Each address is divided into prefix and suffix.
  - To accommodate both small and large networks, three fixed-length prefixes were designed (n = 8, n = 16, and n = 24).



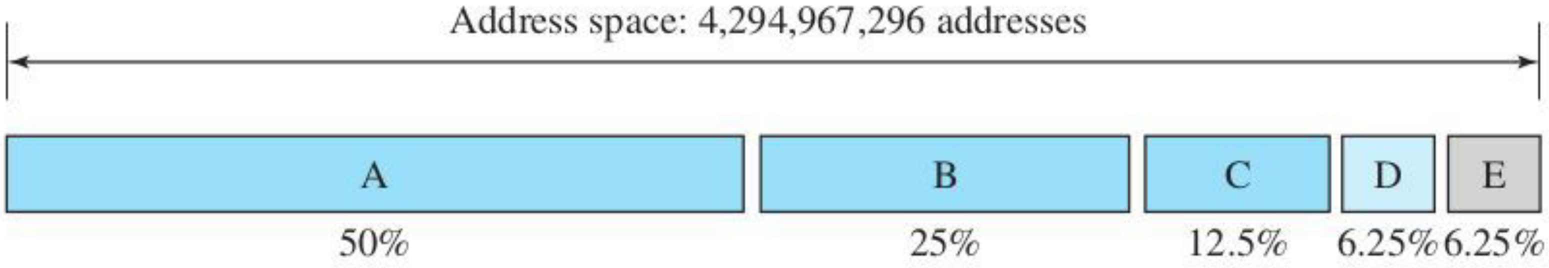


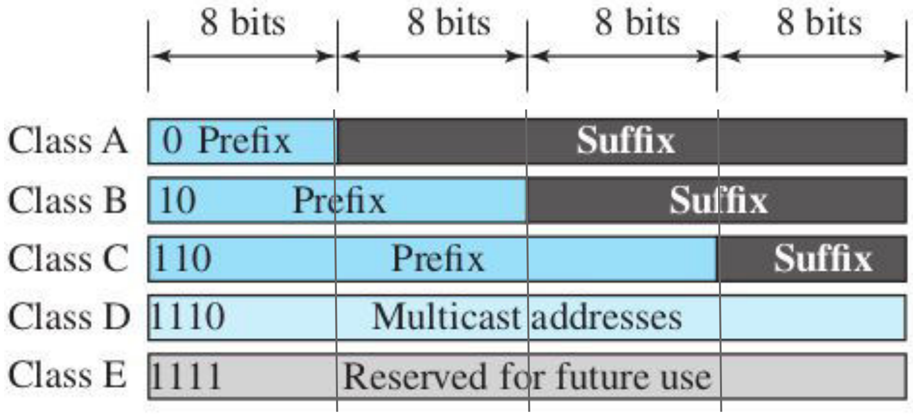


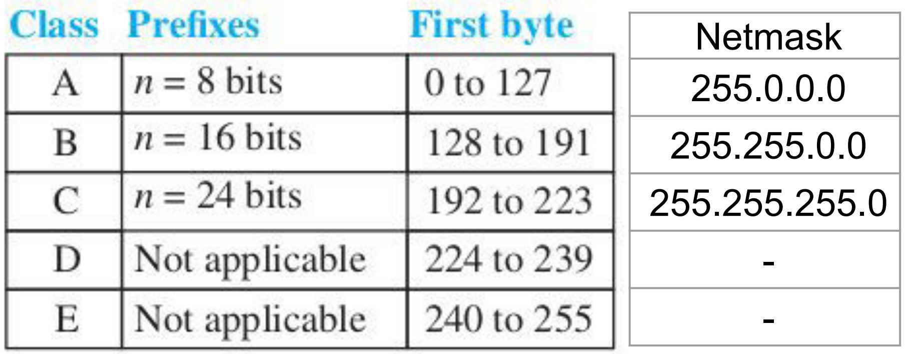
Class	Prefixes	First byte	Netmask
A	$n = 8$ bits	0 to 127	255.0.0.0
B	$n = 16$ bits	128 to 191	255.255.0.0
C	$n = 24$ bits	192 to 223	255.255.255.0
D	Not applicable	224 to 239	-
E	Not applicable	240 to 255	-

IPV4 ADDRESS

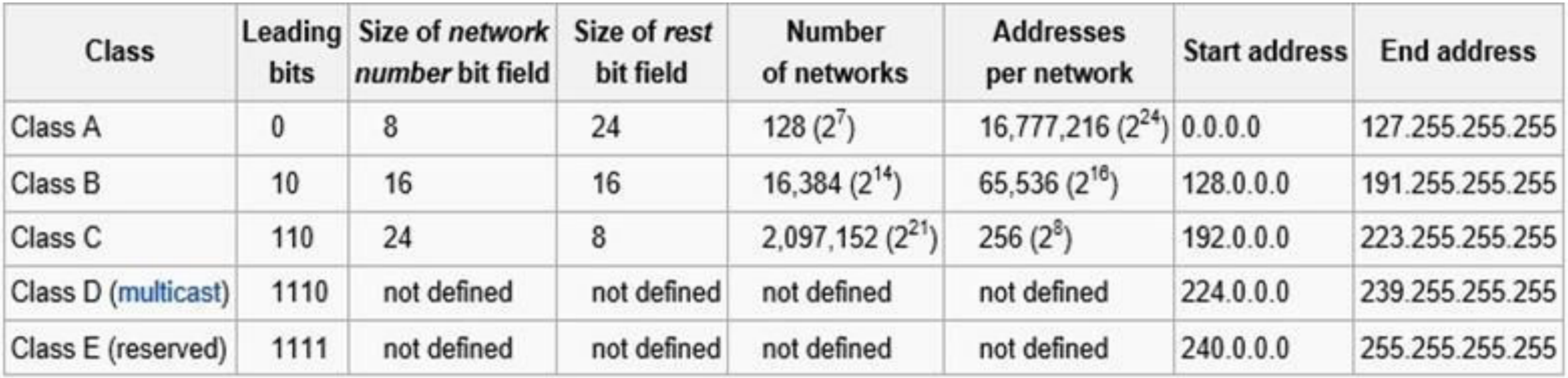
- **Classful Addressing**







Class	Prefixes	First byte	Netmask
A	*n* = 8 bits	0 to 127	255.0.0.0
B	*n* = 16 bits	128 to 191	255.255.0.0
C	*n* = 24 bits	192 to 223	255.255.255.0
D	Not applicable	224 to 239	-
E	Not applicable	240 to 255	-



Class	Leading bits	Size of *network number* bit field	Size of *rest* bit field	Number of networks	Addresses per network	Start address	End address
Class A	0	8	24	128 (2^{7})	16,777,216 (2^{24})	0.0.0.0	127.255.255.255
Class B	10	16	16	16,384 (2^{14})	65,536 (2^{16})	128.0.0.0	191.255.255.255
Class C	110	24	8	2,097,152 (2^{21})	256 (2^{8})	192.0.0.0	223.255.255.255
Class D (multicast)	1110	not defined	not defined	not defined	not defined	224.0.0.0	239.255.255.255
Class E (reserved)	1111	not defined	not defined	not defined	not defined	240.0.0.0	255.255.255.255

IPV4 ADDRESS

- **Extracting Information from a Classful Address**
    - Eg: 192.168.1.5
        - Class C (ie, 24 bits network part, 8 bits host part)
        - Total addresses in this network: 256
        - Network: 192.168.1.0
        - First host address: 192.168.1.1
        - Last host address: 192.168.1.254
        - Broadcast address in this network: 192.168.1.255
        - Max. number of hosts in this network: 254
        - Netmask (or commonly *subnet mask*): 255.255.255.0

IPV4 ADDRESS

- **Address depletion in Classful Addressing**
  - Due to fixed length network fields, there were no more addresses available for new organizations and individuals that needed to be connected to the Internet.
  - Eg: Class A can be assigned to only 128 organizations in the world, but each organization needs to have a single network (seen by the rest of the world) with 16,777,216 nodes (computers in this single network).
    - Too few organizations with that number of computers.
  - Many Class B addresses remained unused, but couldn’t be given to others.
  - Class C networks are more, but each have very few addresses (256).

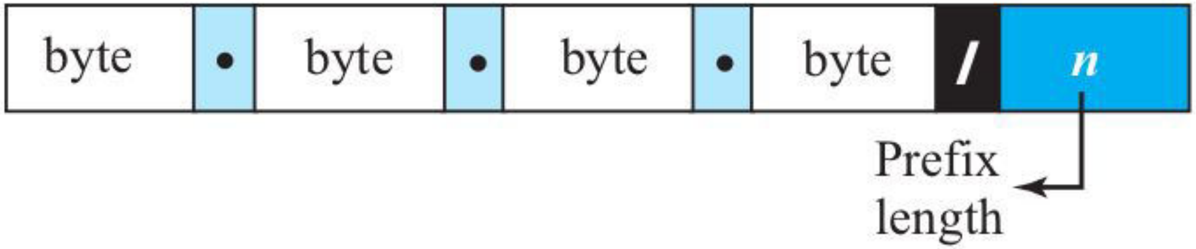
IPV4 ADDRESS

- **Subnetting and Supernetting in Classful Addressing**
  - In **subnetting**, a class A or class B block is divided into several subnets.
    - Eg: If a network in class A is divided into four subnets, each subnet has a prefix of n_{sub} = 10 bits.
    - Most organizations were unhappy in dividing their network.
  - **Supernetting** was devised to combine several class C blocks into a larger block to be attractive to organizations that need more than the 256 addresses available in a class C block.
    - Did not work as it made the routing of packets more difficult.

Advantage of Classful addressing: With a given address, easy to find its class. No information regarding the network is needed in the packet processing.

IPV4 ADDRESS

- **Classless Addressing**
  - Variable number of prefix bits based on requirement.
  - Divides the network into **blocks** of variable length (ie, number of nodes), which is a power of 2.
  - An address in Class A can be thought of as a classless address with prefix length 8, and so on.
  - An address with prefix length n is represented with a slash notation as given below;



Examples:

12.24.76.8/8

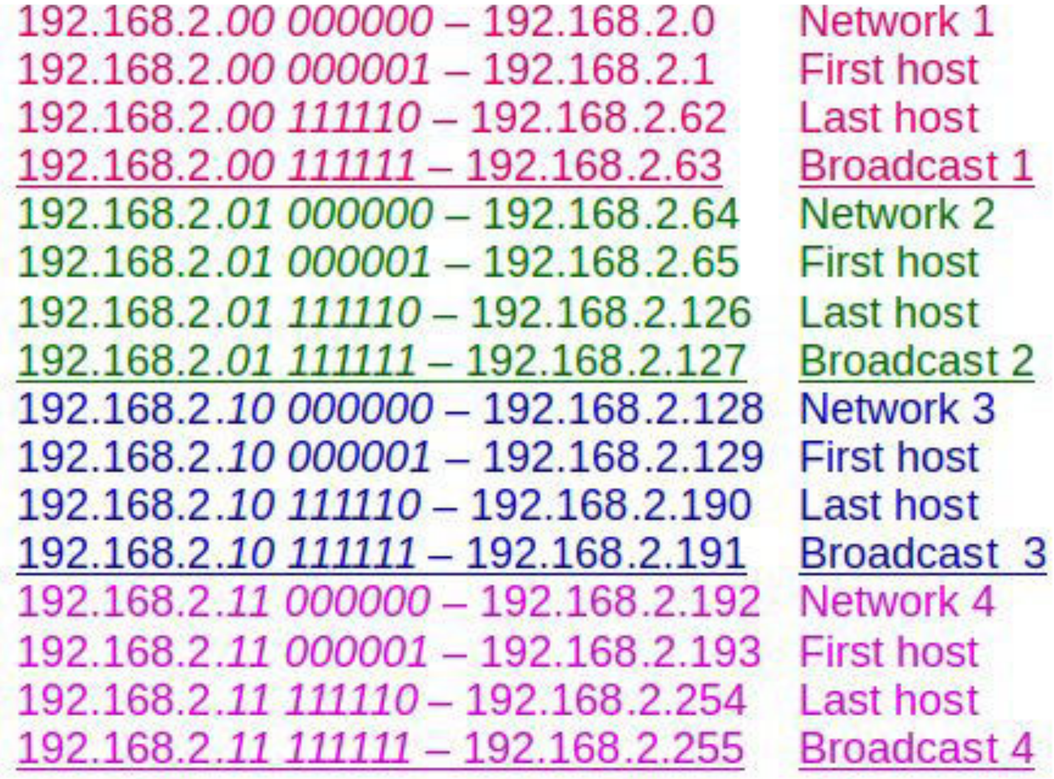
23.14.67.92/12

220.8.24.255/25

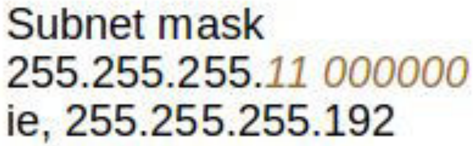
- This notation is officially called as classless interdomain routing or CIDR notation.

IPV4 ADDRESS

- **Subnetting in Classless network**
  - Dividing a network into smaller subnets by increasing the number of bits in the prefix (network part).
    - eg: We have a network 192.168.2.0/24, ie 254 hosts in this network.
    - It can be divided into 4 subnets by changing 2 bits from suffix to prefix. It will become four */26* networks.



192.168.2.*00 000000* – 192.168.2.0	Network 1
192.168.2.*00 000001* – 192.168.2.1	First host
192.168.2.*00 111110* – 192.168.2.62	Last host
192.168.2.*00 111111* – 192.168.2.63	Broadcast 1
192.168.2.*01 000000* – 192.168.2.64	Network 2
192.168.2.*01 000001* – 192.168.2.65	First host
192.168.2.*01 111110* – 192.168.2.126	Last host
192.168.2.*01 111111* – 192.168.2.127	Broadcast 2
192.168.2.*10 000000* – 192.168.2.128	Network 3
192.168.2.*10 000001* – 192.168.2.129	First host
192.168.2.*10 111110* – 192.168.2.190	Last host
192.168.2.*10 111111* – 192.168.2.191	Broadcast 3
192.168.2.*11 000000* – 192.168.2.192	Network 4
192.168.2.*11 000001* – 192.168.2.193	First host
192.168.2.*11 111110* – 192.168.2.254	Last host
192.168.2.*11 111111* – 192.168.2.255	Broadcast 4



Subnet mask
255.255.255.*11 000000*
ie, 255.255.255.192

IPV4 ADDRESS

- **Extracting Information from a Classless Address**
  - For an address with prefix length *n*,
    - Total number of addresses in the block, N = 2^{32-n}.
    - First address (ie, network address): the address with *32-n* rightmost bits set to 0.
    - Last address (ie, broadcast address): the address with *32-n* rightmost bits set to 1.
  - Eg: 167.199.170.82/ 27
    - Address: 10100111 11000111 10101010 01010010
    - First Address: 10100111 11000111 10101010 01000000 167.199.170.64/ 27
    - Last Address: 10100111 11000111 10101010 01011111 167.199.170.95/ 27

IPV4 ADDRESS

- **Special Addresses:**
  - This-host address:
    - 0.0.0.0/32
    - It is used whenever a host needs to send an IP datagram but it does not know its own address to use as the source address.
    - Mostly used at system booting time.
  - Limited broadcast address:
    - 255.255.255.255/32
    - It is used whenever a router or a host needs to send a datagram to all devices in a network.
    - The other routers in the network block the packet having this address as the destination; the packet cannot travel outside the network.

IPV4 ADDRESS

- **Special Addresses:**
  - Loopback Address:
    - The block 127.0.0.0/8
    - A packet with one of the addresses in this block as the destination address never leaves the host.
    - Used to test some software in the machine, or to test network drivers.

  - Multicast Addresses:
    - The block 224.0.0.0/4 is reserved for multicast addresses.

IPV4 ADDRESS

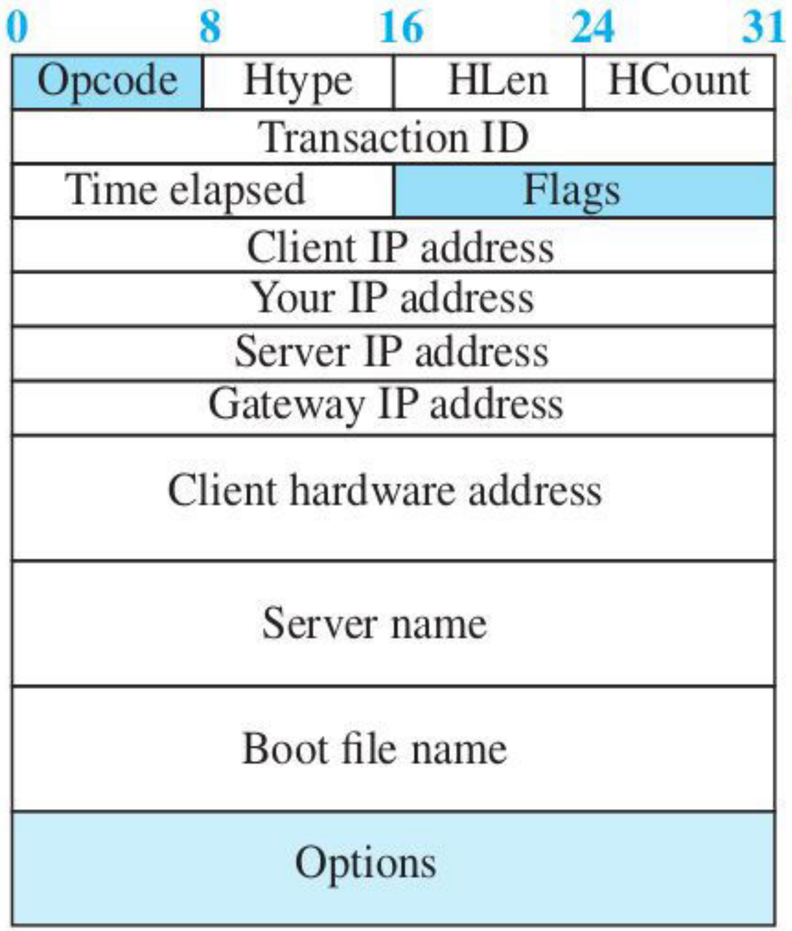
- **Special Addresses:**
    - Private addresses:
        - Four blocks are assigned as private addresses:
            1. 10.0.0.0/8 - Class A (one network)
            2. 172.16.0.0/12 - Class B: 172.16.x.x to 172.31.x.x (16 networks)
            3. 192.168.0.0/16 - Class C: 192.168.x.x (256 networks with 254 hosts each)
            4. 169.254.0.0/16 - Link local addresses which is obtained to a system when IP functionality fails (eg: DHCP failure). Microsoft termed this addressing as Automatic Private IP Addressing (APIPA).
        - The *first three blocks* are used in private LANs.

Dynamic Host Configuration Protocol (DHCP)

- An Application layer protocol that helps the network layer to assign IP address to hosts.
- Assign IP addresses **dynamically/automatically** to each host in the network.
- Also provide other network parameters such as subnet mask, DNS server address, gateway address etc.
- Uses **client-server** paradigm.
- Can **assign IP parameters permanently / temporarily** (for some time) to hosts.
- Uses a request-reply mechanism.

Dynamic Host Configuration Protocol (DHCP)

- Message Format
  - Same format for request and reply

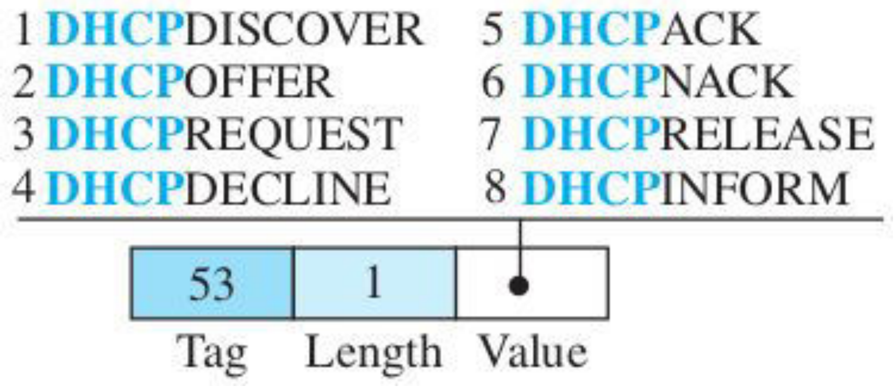


Fields:

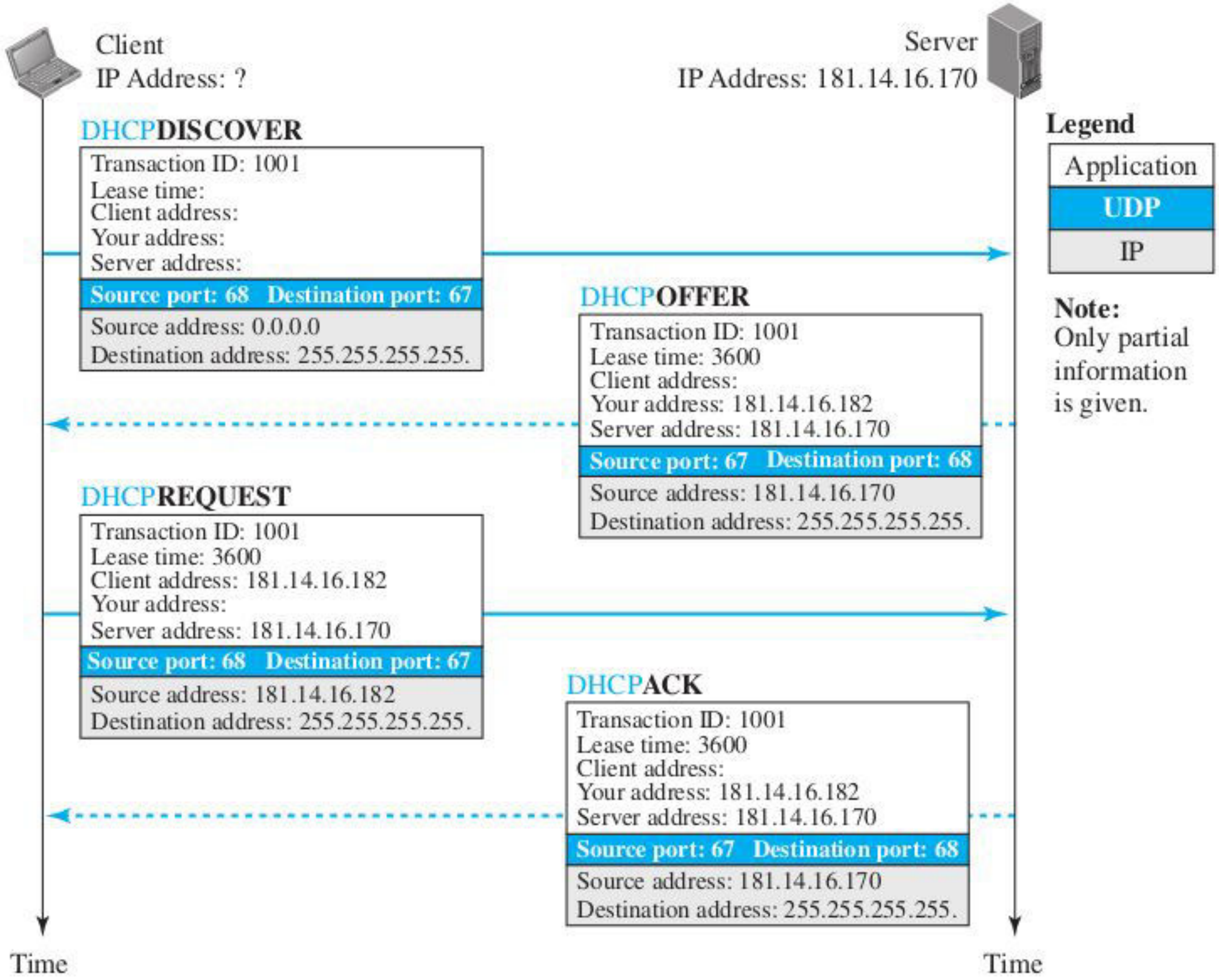
Opcode: Operation code, request (1) or reply (2)
Htype: Hardware type (Ethernet, ...)
HLen: Length of hardware address
HCount: Maximum number of hops the packet can travel
Transaction ID: An integer set by the client and repeated by the server
Time elapsed: The number of seconds since the client started to boot
Flags: First bit defines unicast (0) or multicast (1); other 15 bits not used
Client IP address: Set to 0 if the client does not know it
Your IP address: The client IP address sent by the server
Server IP address: A broadcast IP address if client does not know it
Gateway IP address: The address of default router
Server name: A 64-byte domain name of the server
Boot file name: A 128-byte file name holding extra information
Options: A 64-byte field with dual purpose described in text

Dynamic Host Configuration Protocol (DHCP)

- Message Format
  - Options: 64 bytes
    - Carry either additional information or some specific vendor information.
    - If the option field is present, it will start with a **magic cookie** (a four byte number - 99.130.83.99), the remaining 60 bytes are options.
    - An option is composed of three fields: a 1-byte **tag** field, a 1-byte **length** field, and a variable-length **value** field.
    - If the tag field is 53, the value field defines one of the 8 message types.



Dynamic Host Configuration Protocol (DHCP): Operation



Client
IP Address: ?

Server
IP Address: 181.14.16.170

Legend

- Application
- UDP
- IP

Note:
Only partial information is given.

DHCP**DISCOVER**

Transaction ID: 1001
Lease time:
Client address:
Your address:
Server address:
Source port: 68 Destination port: 67
Source address: 0.0.0.0
Destination address: 255.255.255.255.

DHCP**OFFER**

Transaction ID: 1001
Lease time: 3600
Client address:
Your address: 181.14.16.182
Server address: 181.14.16.170
Source port: 67 Destination port: 68
Source address: 181.14.16.170
Destination address: 255.255.255.255.

DHCP**REQUEST**

Transaction ID: 1001
Lease time: 3600
Client address: 181.14.16.182
Your address:
Server address: 181.14.16.170
Source port: 68 Destination port: 67
Source address: 181.14.16.182
Destination address: 255.255.255.255.

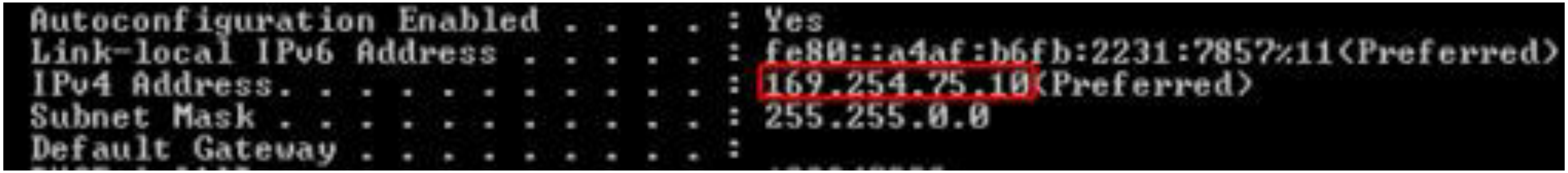
DHCP**ACK**

Transaction ID: 1001
Lease time: 3600
Client address:
Your address: 181.14.16.182
Server address: 181.14.16.170
Source port: 67 Destination port: 68
Source address: 181.14.16.170
Destination address: 255.255.255.255.

Dynamic Host Configuration Protocol (DHCP): Operation

- Using FTP
  - In the DHCPACK message, the server defines the pathname of a file in which the client can find complete information such as the address of the DNS server.

- If the DHCP process fails (say, DHCP server is not present), the client itself assigns an IP address in the range 169.254.x.x
  - This is called Auto IP, or APIPA (Automatic Private IP Addressing) address.



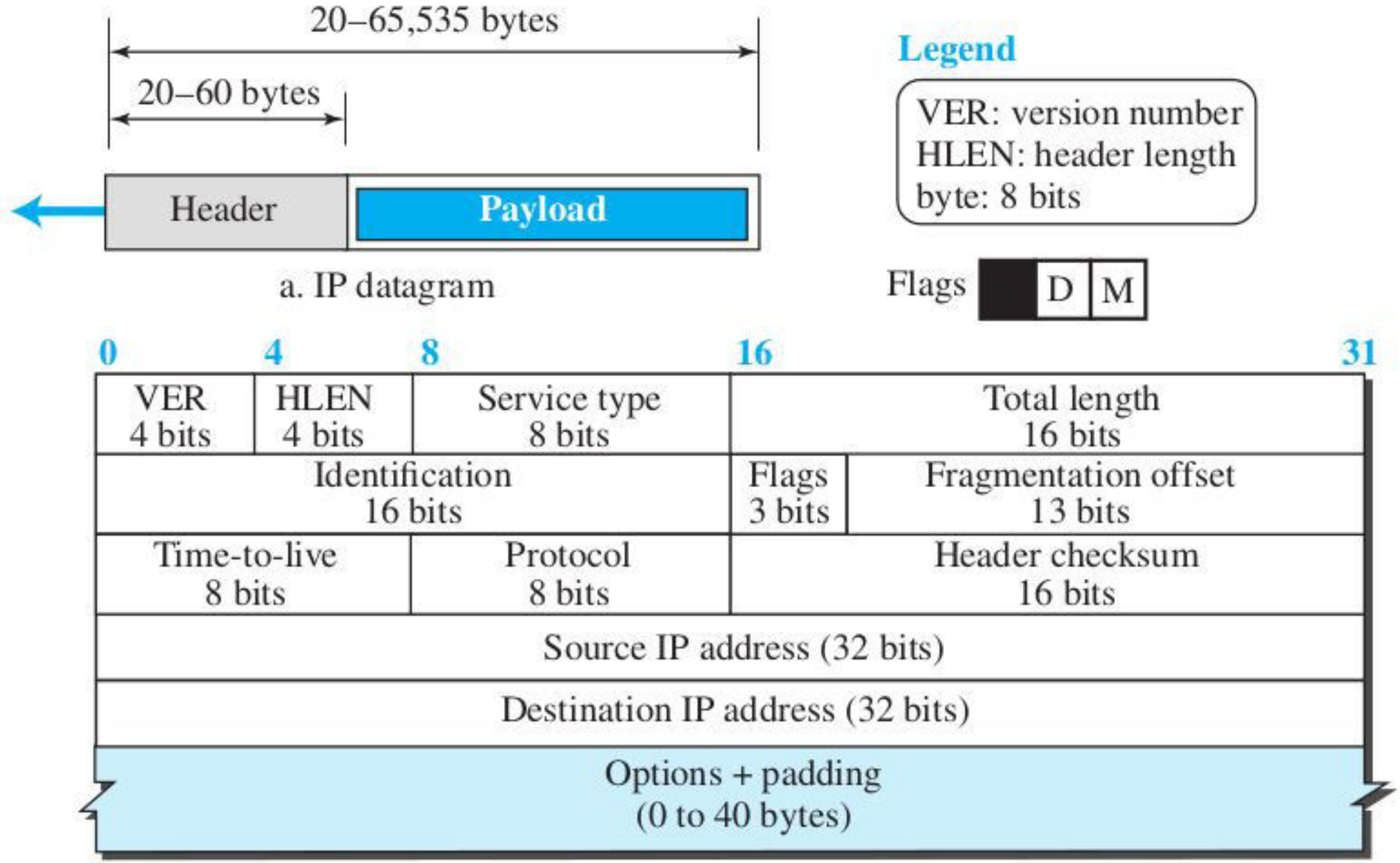
```
Autoconfiguration Enabled . . . . : Yes
Link-local IPv6 Address . . . . . : fe80::a4af:b6fb:2231:7857%11(Preferred)
IPv4 Address. . . . . . . . . . . : 169.254.75.10(Preferred)
Subnet Mask . . . . . . . . . . . : 255.255.0.0
Default Gateway . . . . . . . . . :
```

INTERNET PROTOCOL (IP)

- A connectionless protocol.
- An unreliable datagram protocol to provide best-effort delivery service.
- IPv4 packets can be corrupted, be lost, arrive out of order, or be delayed, and may create congestion for the network.
- If reliability is important, IPv4 must be paired with a reliable transport-layer protocol such as TCP.
- Supported by many auxiliary protocols, such as;
  - The Internet Group Management Protocol (**IGMP**) helps IPv4 in multicasting.
  - Internet Control Message Protocol (**ICMPv4**) helps IPv4 to handle some errors that may occur in the network-layer delivery.
  - Address Resolution Protocol (**ARP**) is used to glue the network and data-link layers in mapping network-layer addresses to link-layer addresses.

INTERNET PROTOCOL

- **Datagram Format**



a. IP datagram

Legend

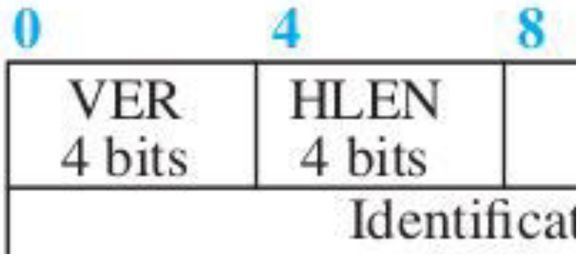
VER: version number
HLEN: header length
byte: 8 bits

<table>
<tr><td>VER
4 bits</td><td>HLEN
4 bits</td><td>Service type
8 bits</td><td colspan="2">Total length
16 bits</td></tr>
<tr><td colspan="3">Identification
16 bits</td><td>Flags
3 bits</td><td>Fragmentation offset
13 bits</td></tr>
<tr><td colspan="2">Time-to-live
8 bits</td><td>Protocol
8 bits</td><td colspan="2">Header checksum
16 bits</td></tr>
<tr><td colspan="5">Source IP address (32 bits)</td></tr>
<tr><td colspan="5">Destination IP address (32 bits)</td></tr>
<tr><td colspan="5">Options + padding
(0 to 40 bytes)</td></tr>
</table>

b. Header

INTERNET PROTOCOL: Datagram format

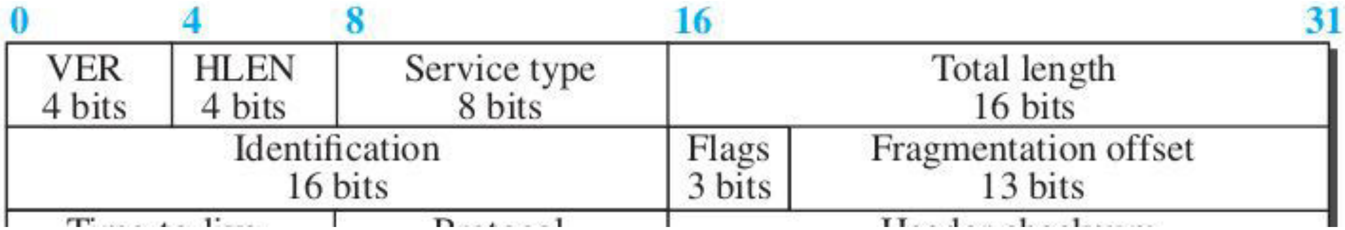
- **Version Number:**
  - 4-bit
  - Defines the version of the IPv4 protocol, which has the value of 4 (0100_2)


0 4 8

VER 4 bits	HLEN 4 bits	
Identificat		

- **Header Length:**
  - 4-bit
  - Defines the total length of the datagram header in 4-byte words.
  - Varies from **5 to 15** (means 20 bytes to 60 bytes)
  - In binary, 0101 to 1111
  - Eg: If the value in the header length field is 8, it means the header has 8x4=32 bytes, which also means 20 bytes basic header and 12 bytes options.

INTERNET PROTOCOL: Datagram format



0	4	8	16	31
VER 4 bits	HLEN 4 bits	Service type 8 bits	Total length 16 bits	
Identification 16 bits			Flags 3 bits	Fragmentation offset 13 bits

- **Service Type:**
    - 8-bits
    - Specifies *differentiated services (DiffServ)* for different types of protocols.

- **Total Length:**
    - 16 bit field.
    - Defines the total length (**header plus data**) of the IP datagram in bytes.
    - Most useful when zero-padding occurs due to small datagram size (Standard Ethernet frame needs minimum 46 bytes payload).

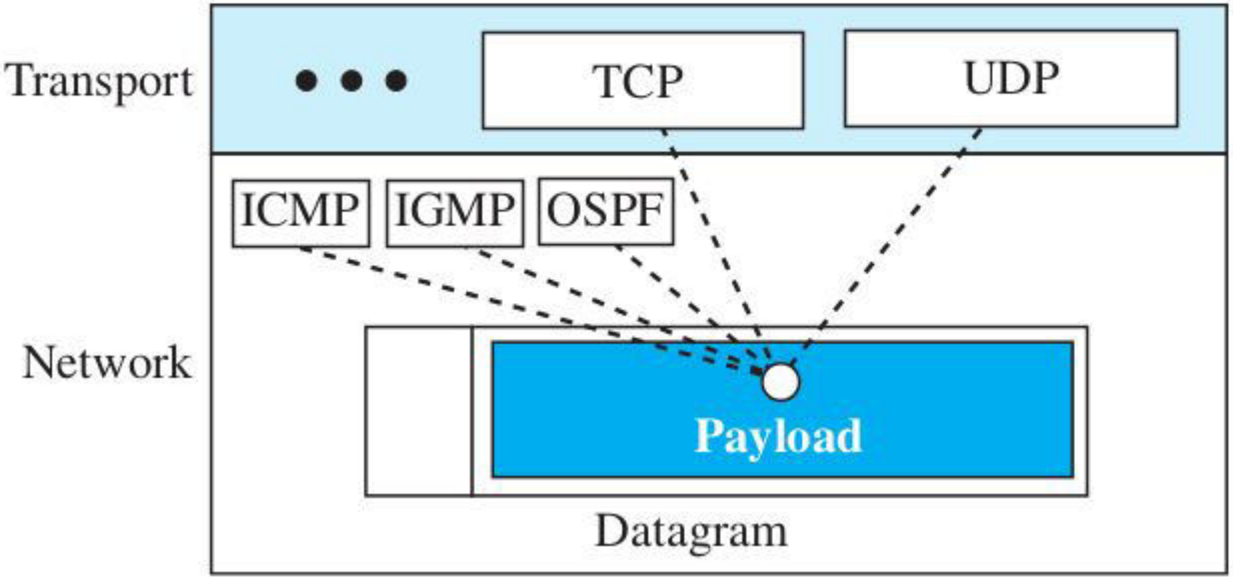
- **Identification, Flags, and Fragmentation Offset:**
    - Total 32 bits
    - Related to the fragmentation of the IP datagram when the size of the datagram is larger than the underlying network can carry.

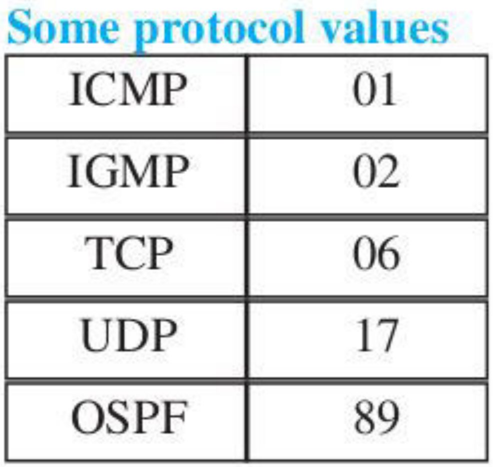
INTERNET PROTOCOL: Datagram format

- **Time-to-live (TTL)**
    - 8 bits
    - To control the maximum number of hops (routers) visited by the datagram.
    - When a source host sends the datagram, it stores a number in this field (which may be double the maximum number routers up to the destination).
    - Each router decrements this value when the datagram reaches the router.
    - When the TTL becomes zero, the packet is discarded.

INTERNET PROTOCOL: Datagram format

- **Protocol**
  - 8 bits.
  - Specifies the protocol of the payload.





Some protocol values

ICMP	01
IGMP	02
TCP	06
UDP	17
OSPF	89

INTERNET PROTOCOL: Datagram format

- **Header checksum:**
  - 16 bits.
  - Set at the source to protect the header from errors.
  - Also recalculated at the routers as values of some fields such as TTL changes at routers.
  - Checksum is the complement of the sum of other fields calculated using 1s complement arithmetic.

- **Source and Destination Addresses:**
  - 4 bytes each.

INTERNET PROTOCOL: Datagram format

- **Options:**
  - Single-byte and multiple-byte options.
  - Used for testing and debugging

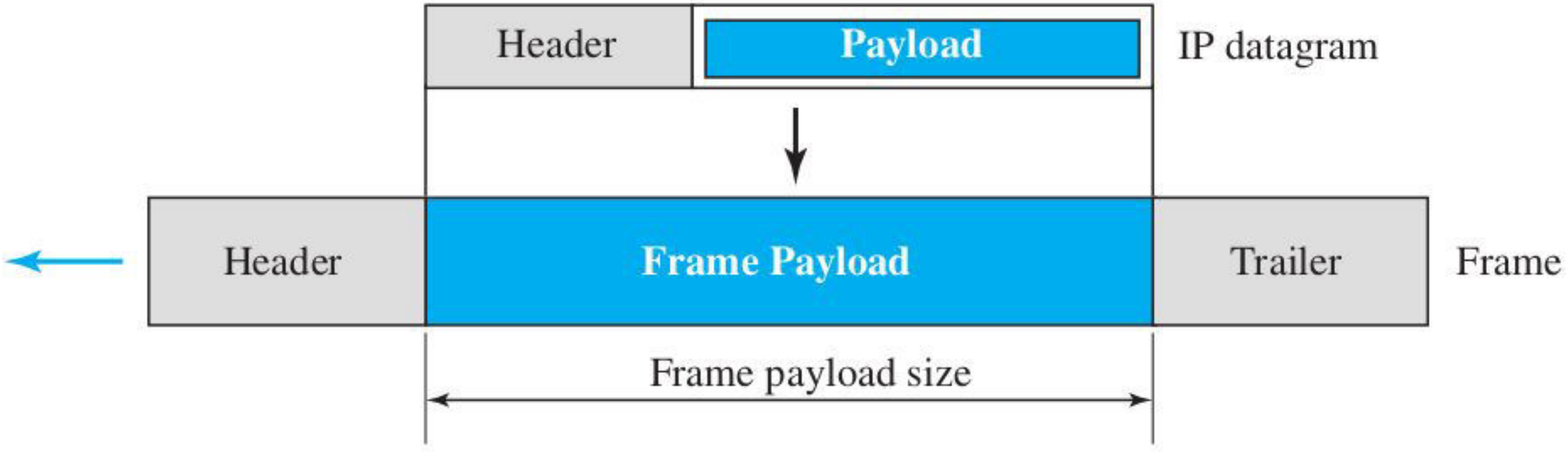
- **Payload:**
  - The packet coming from other protocols that use the service of IP.

INTERNET PROTOCOL: Test Questions

- **An IPv4 packet has arrived with the first 8 bits as $(0100\ 0010)_2$ The receiver discards the packet. Why?**
  - First 4 bits: 0100=4, ie, IPv4 -> ok
  - Second 4 bits: 0010=2 -> ERROR as header length field must have a minimum value of 5

INTERNET PROTOCOL

- **Fragmentation**
  - Occurs if the datagram size is greater than the **Maximum Transfer Unit (MTU)** of the data link layer.
  - Fragmentation is the process of dividing the datagram into small fragments so as to limit the size within the MTU.
  - **MTU**: the maximum size of the payload that can be encapsulated in a frame; it is the restriction imposed by the hardware and software.
    - MTU of Standard Ethernet is 1500 bytes.



MTU: Maximum size of frame payload

INTERNET PROTOCOL

- **Fragmentation**
  - Occurs at the source or at the intermediate routers.
  - Reassembly occurs at the destination only, as the fragments may go through different routes.
  - Fields Related to Fragmentation
    - Identification - 16 bits
    - Flags - 3 bits
    - Fragmentation offset - 13 bits

INTERNET PROTOCOL: Fragmentation

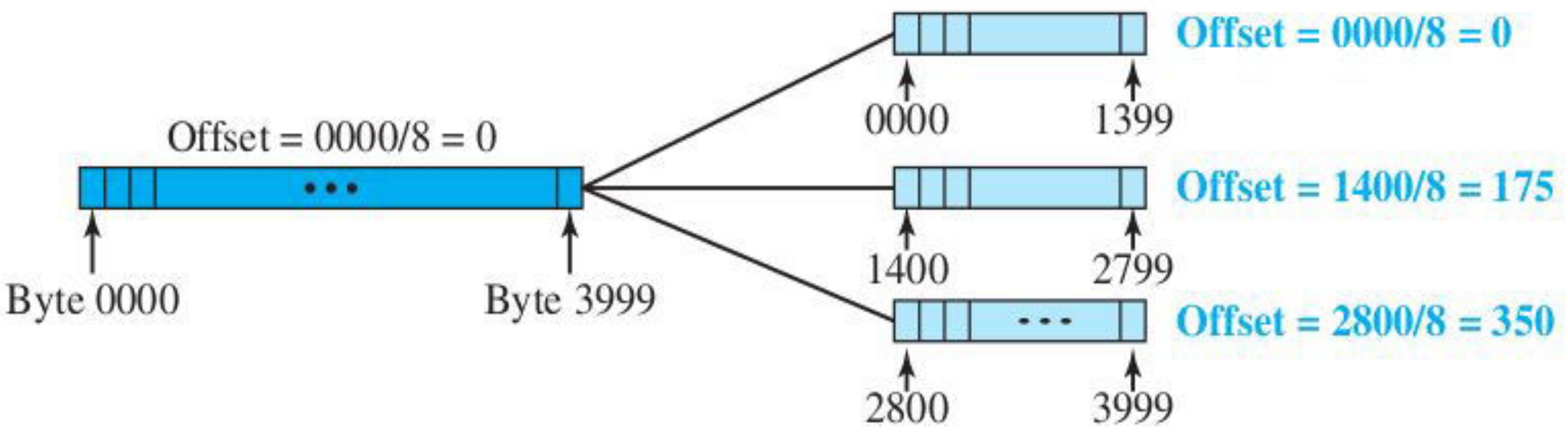
- **Identification field:** 16 bits
  - Identifies a datagram originating from the source host.
  - All fragments of a datagram will have the same identification number.
  - Helps the destination in reassembling the datagram.
  - The combination of the identification and source IP address must uniquely define a datagram as it leaves the source host.
  - The source uses a counter for this field.

INTERNET PROTOCOL: Fragmentation

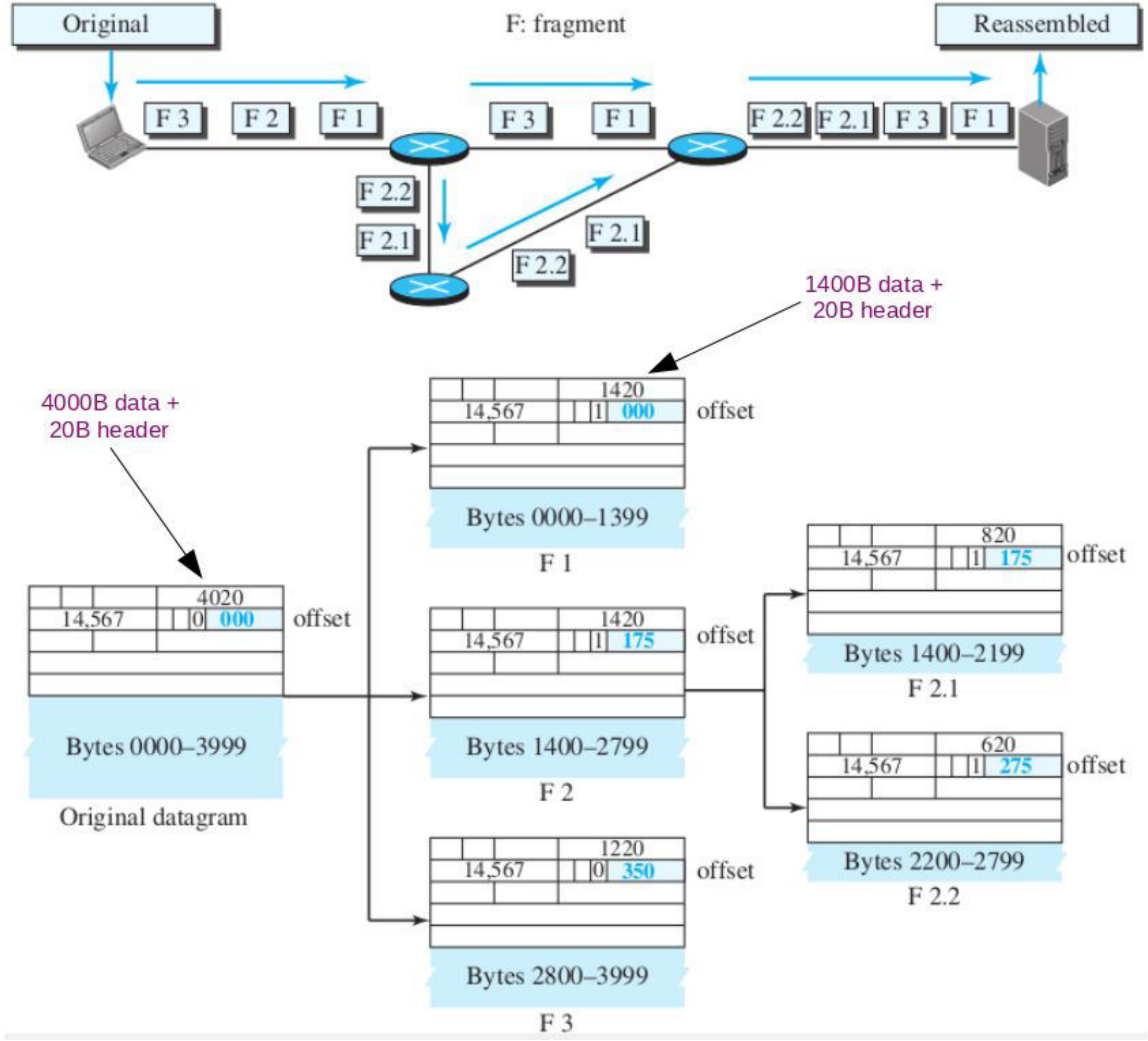
- **Flags field**: 3 bits
  - 1. Reserved (not used)
  - 2. D bit (**Do not fragment** bit):
    - Value 0: Can be fragmented, if needed.
    - Value 1:
      - The machine must not fragment the datagram.
      - Discards the datagram and sends an ICMP error message to the source host
  - 3. M bit (**More fragments** bit):
    - Value 0: This is the last/only fragment of the datagram.
    - Value 1: There are more fragments after it.

INTERNET PROTOCOL: Fragmentation

- **Fragmentation offset field**: 13 bits
  - Shows the relative position of this fragment with respect to the whole datagram.
  - It is the offset of the data in the original datagram measured in units of 8 bytes.
  - Calculation Example:
    - Consider a datagram with 4000 bytes data; divided with a maximum of 1400 bytes. (data bytes numbered 0 to 3999)



INTERNET PROTOCOL: Fragmentation example



1400B data + 20B header

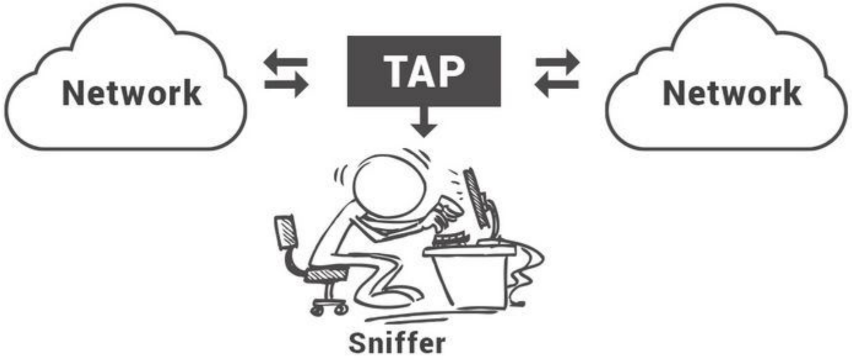
4000B data + 20B header

Security of IPv4 Datagrams

- Internet Protocol was implemented without any security concerns.
- Three major security issues that are particularly applicable to the IP are;
  - Packet sniffing
  - Packet modification
  - IP spoofing.

Security of IPv4 Datagrams

- Packet sniffing:
  - An intruder may intercept an IP packet and make a copy of it.
  - It is a passive attack, in which the attacker does not change the contents of the packet.
  - Very difficult to detect because the sender and the receiver may never know that the packet has been copied.
  - Encryption of the packet can make the attacker's effort useless.



Security of IPv4 Datagrams

- Packet Modification:
  - The attacker intercepts the packet, changes its contents, and sends the new packet to the receiver.
  - The receiver believes that the packet is coming from the original sender.
  - Can be detected using a data integrity mechanism.

- IP Spoofing
  - An attacker can masquerade as somebody else and create an IP packet that carries the source address of another computer.
  - This type of attack can be prevented using an origin authentication mechanism.

Security of IPv4 Datagrams

- **IPSec**
  - A protocol to protect from the IP security issues.
  - Provide data authentication, integrity, and confidentiality. It also defines the encrypted, decrypted and authenticated packets.
  - Mostly used in Virtual Private Networks (VPN).
  - IPSec works in conjunction with IP and makes a connection oriented service between the endpoints.
  - IPSec also provides;
    - *Defining Algorithms and Keys*
    - *Packet Encryption*; prevents sniffing attack
    - *Data Integrity*; prevents packet modification
    - *Origin Authentication*; prevents IP spoofing

End of Part 2 of Module 2

Thank You

Diploma in Computer Hardware Engineering

COMPUTER NETWORKS

(5151 Rev 2015)

Module 2 - Part 3

Presenter: Sreejesh NG
Lecturer in Computer Hardware Engineering
Government Polytechnic College, Cherthala

Ref: Data Communications and Networking, 5E by Forouzan

Course General Outcomes:

Sl.	G.O	On completion of this course the student will be able :
1	1	To Understand the concept of TCP/IP Protocol
2	1	To Understand the concept of Network Layer
3	1	To Understand the concept of Transport Layer
4	1	To Understand the concept of Application Layer

Specific Outcomes:

2.1 Understand Network Layer

- 2.1.1 Explain Network layer services
- 2.1.2 Illustrate network layer performance
- 2.1.3 Describe IPV4 addresses
- 2.1.4 Define DHCP
- 2.1.5 Explain Internet Protocol
- 2.1.6 State security of IPV4 datagram
- 2.1.7 Describe routing algorithms
- 2.1.8 Differentiate between unicasting, multicasting, and broadcasting

MODULE 2 - NETWORK LAYER

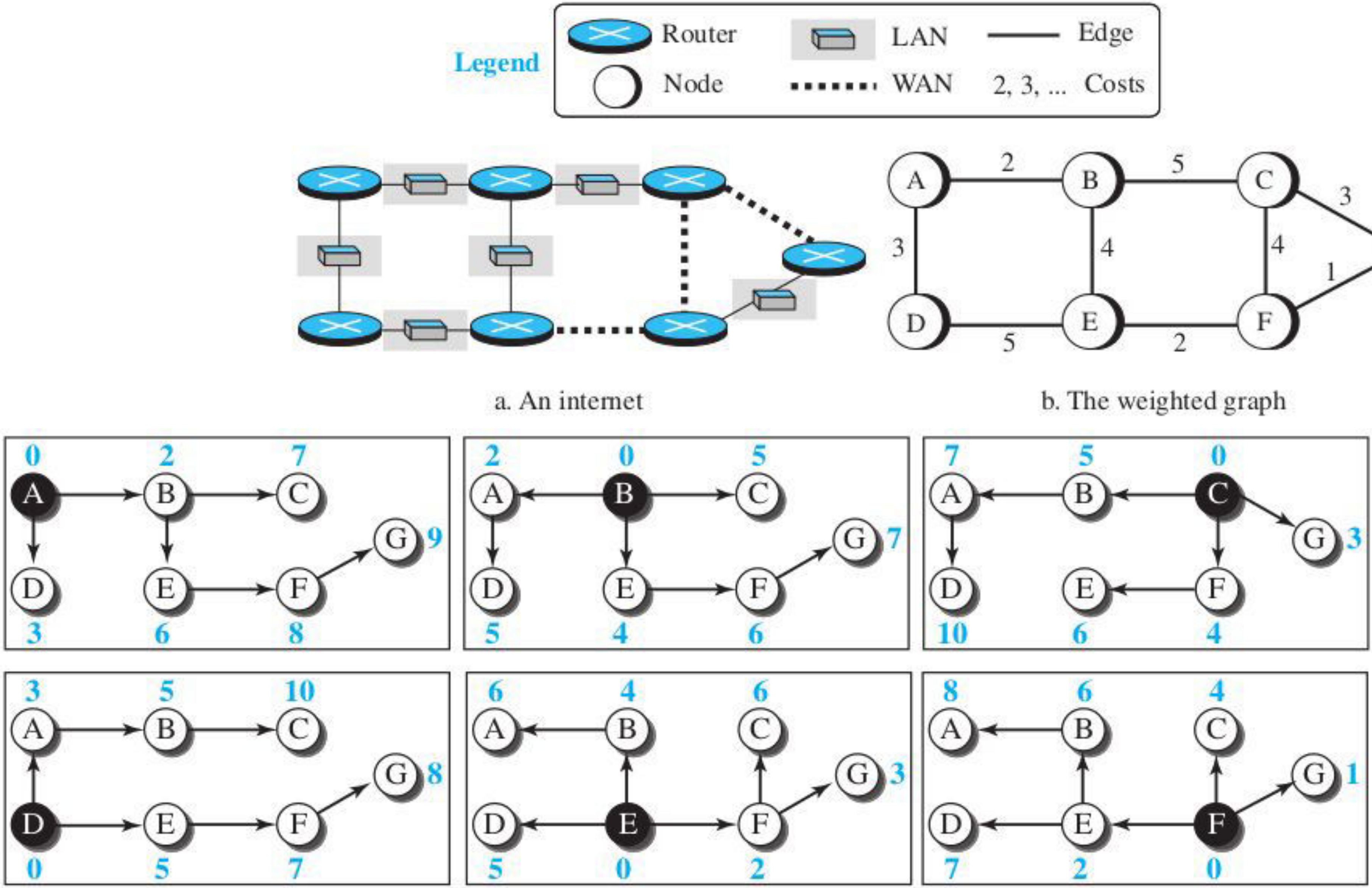
- Network layer services – Packetizing, routing and forwarding, other services
- Performance – delay, throughput, packet loss, congestion control
- IPV4 address – address space, classful addressing, classless addressing, subnetting
- DHCP
- Internet protocol (IP) – datagram format, fragmentation
- IPV4 datagram security
- Routing algorithms
  - Distance-vector (DV)
  - Link-state (LS)
  - Path vector (PV)
- Unicasting, multicasting, broadcasting

Ref: Data Communications and Networking - Behrouz A. Forouzan -
McGraw Hill Edn.- Fifth Edition

ROUTING ALGORITHMS

- Designed to find the least cost paths between routers.
- Basis for different routing protocols.
- Differs the way they interpret the least cost and the way they create the least-cost tree for each node.
- Major Algorithms:
  - Distance Vector (DV) routing algorithm
  - Link State routing (LS) algorithm
  - Path Vector (PV) routing algorithm

Least Cost Trees


Legend
Router
Node
LAN
WAN
Edge
2, 3, ... Costs
A 2 B 5 C
3 4 4 3 1 G
D 5 E 2 F
A 0, B 2, C 7, D 3, E 6, F 8, G 9
A 2, B 0, C 5, D 5, E 4, F 6, G 7
A 7, B 5, C 0, D 10, E 6, F 4, G 3
A 3, B 5, C 10, D 0, E 5, F 7, G 8
A 6, B 4, C 6, D 5, E 0, F 2, G 3
A 8, B 6, C 4, D 7, E 2, F 0, G 1

a. An internet

b. The weighted graph

Sample least cost **trees** from some nodes (root node is indicated as black node)

Distance Vector (DV) Routing Algorithm

- The first thing each node creates is its own least-cost tree with the rudimentary information it has about its immediate neighbors.
- The incomplete trees are exchanged between immediate neighbors to make the trees more and more complete and to represent the whole internet.
- A router continuously tells all of its neighbors what it knows about the whole internet.

Distance Vector Routing Algorithm

- Bellman-Ford Equation
  - Heart of DV routing
  - Used to find the least cost (shortest distance) between a source node, x, and a destination node, y, through some intermediary nodes (a, b, c), when the costs between the source and the intermediary nodes (say ***c***) and the least costs between the intermediary nodes and the destination (say ***D***) are given.

$$D_{xy} = \min\{(c_{xa} + D_{ay}), (c_{xb} + D_{by}), (c_{xc} + D_{cy}), \ldots\}$$

  - To update an existing least cost with a least cost through an intermediary node, such as z, if the latter is shorter.

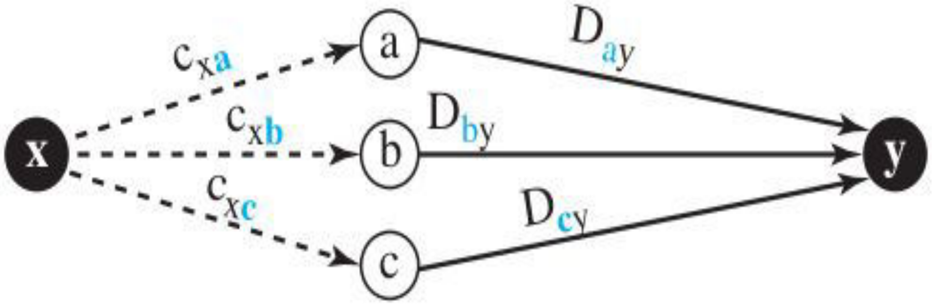
$$D_{xy} = \min\{D_{xy}, (c_{xz} + D_{zy})\}$$

Distance Vector Routing Algorithm

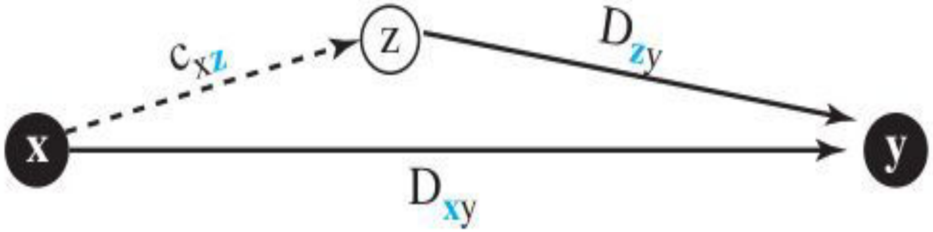
- Bellman-Ford Equation
  - c_{xa} - Cost between **x** and **a**
  - D_{ay} - Least cost between **a** and **y**

$$D_{xy} = \min\{(c_{xa} + D_{ay}), (c_{xb} + D_{by}), (c_{xc} + D_{cy}), \ldots\}$$

$$D_{xy} = \min\{D_{xy}, (c_{xz} + D_{zy})\}$$



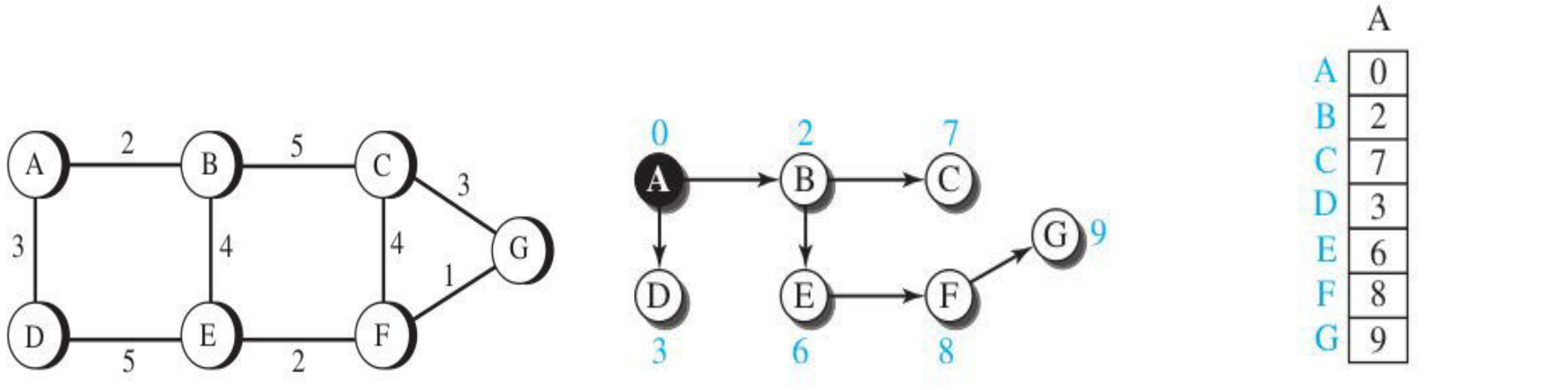
a. General case with three intermediate nodes



b. Updating a path with a new route

Distance Vector Routing Algorithm

- Distance Vectors
  - a one-dimensional array to represent the least cost tree.
  - the **name** of the distance vector defines the root, the **indexes** define the destinations, and the **value** of each cell defines the least cost from the root to the destination.
  - Gives only the least costs to the destinations, not the paths.



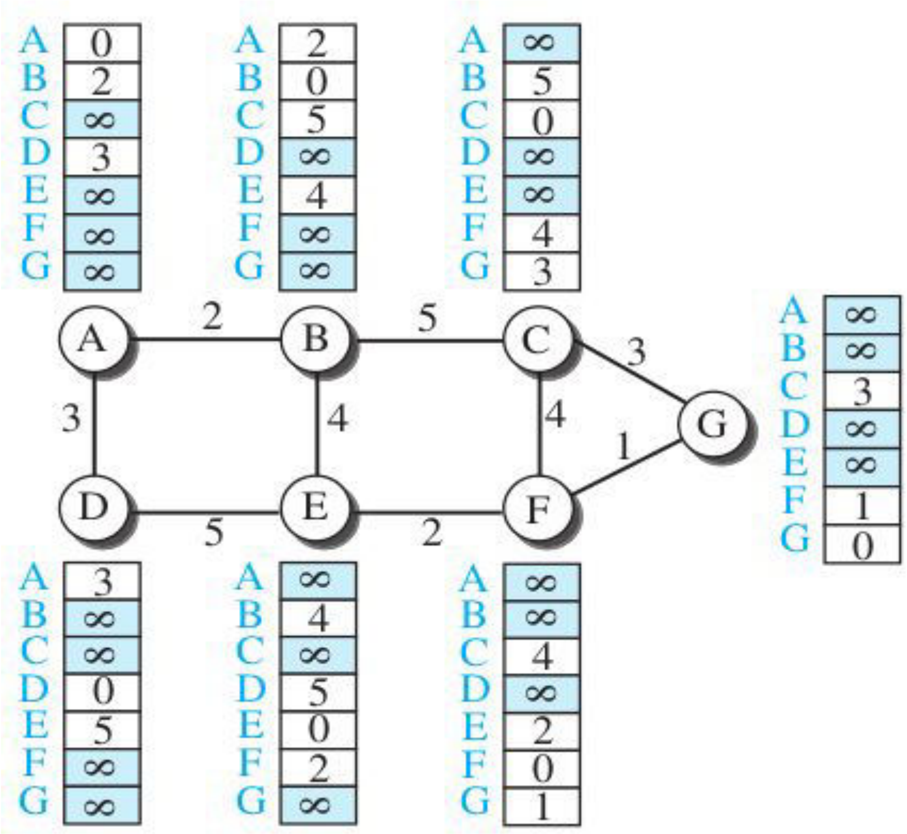
	A
A	0
B	2
C	7
D	3
E	6
F	8
G	9

a. Tree for node A

b. Distance vector for node A

Distance Vector Routing Algorithm - Steps

1. On booting, each node sends some greeting messages out of its interfaces and discovers the identity of the immediate neighbors and the distance between itself and each neighbor.
2. Makes a simple distance vector by these values and set other values as infinity.



A	0
B	2
C	∞
D	3
E	∞
F	∞
G	∞

A	2
B	0
C	5
D	∞
E	4
F	∞
G	∞

A	∞
B	5
C	0
D	∞
E	∞
F	4
G	3

A	∞
B	∞
C	3
D	∞
E	∞
F	1
G	0

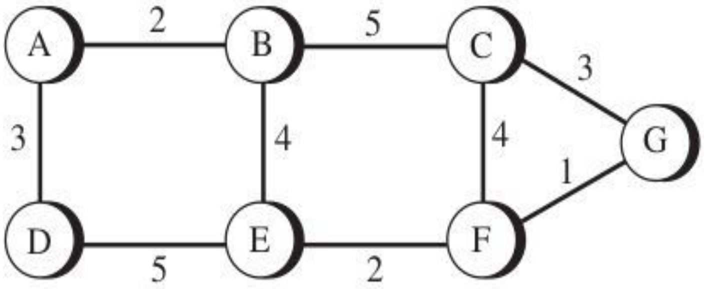
A	3
B	∞
C	∞
D	0
E	5
F	∞
G	∞

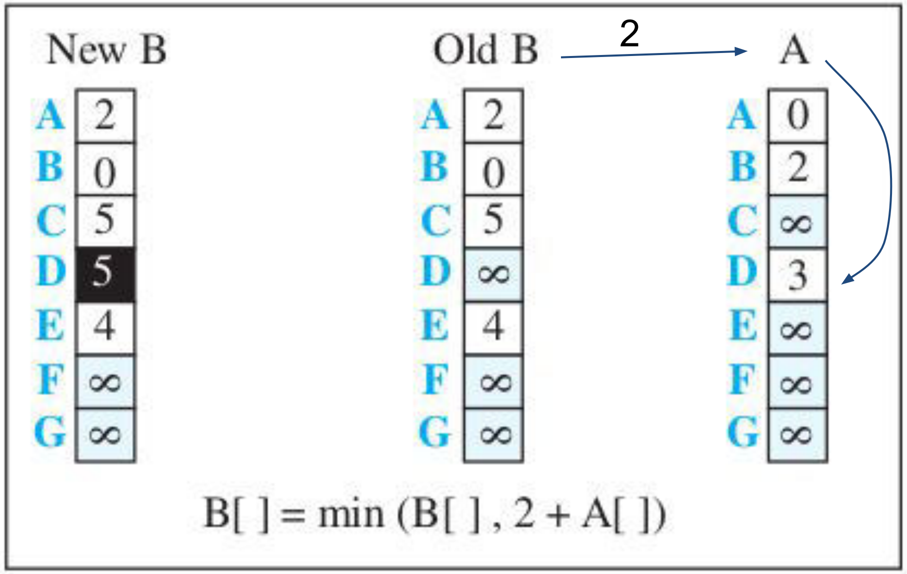
A	∞
B	4
C	∞
D	5
E	0
F	2
G	∞

A	∞
B	∞
C	4
D	∞
E	2
F	0
G	1

Distance Vector Routing Algorithm - Steps

3. Then it sends a copy of the vector to all its immediate neighbors.
4. After a node receives a distance vector from a neighbor, it updates its distance vector using the Bellman-Ford equation (second case: Update).

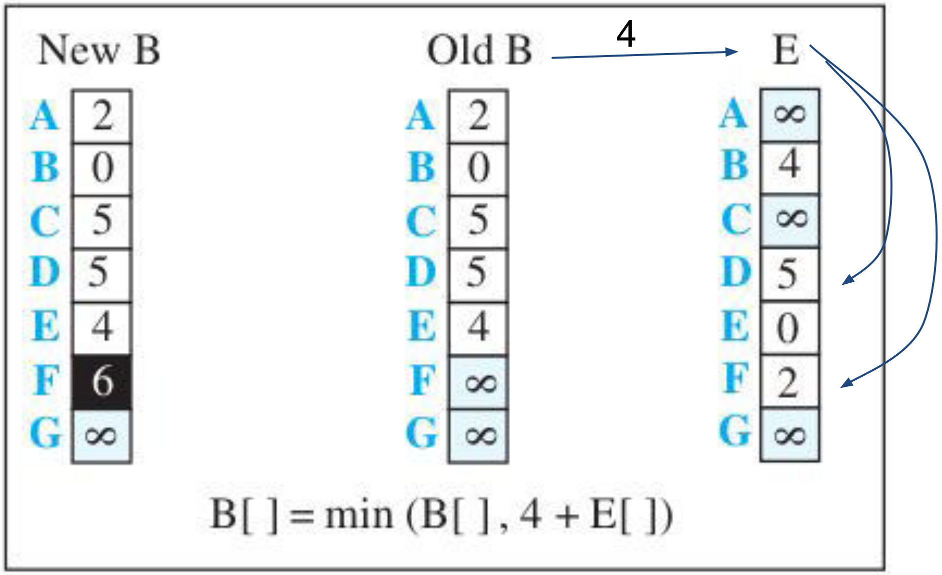




	New B	Old B	A
A	2	2	0
B	0	0	2
C	5	5	∞
D	5	∞	3
E	4	4	∞
F	∞	∞	∞
G	∞	∞	∞

B[ ] = min (B[ ], 2 + A[ ])

a. First event: B receives a copy of A's vector.



	New B	Old B	E
A	2	2	∞
B	0	0	4
C	5	5	∞
D	5	5	5
E	4	4	0
F	6	∞	2
G	∞	∞	∞

B[ ] = min (B[ ], 4 + E[ ])

b. Second event: B receives a copy of E's vector.

Distance Vector Routing Algorithm - Steps

5. After updating a node, it immediately sends its updated vector to all neighbors.
6. This process is repeated continuously and eventually each node finds the least cose tree.

Distance Vector Routing Algorithm - Steps

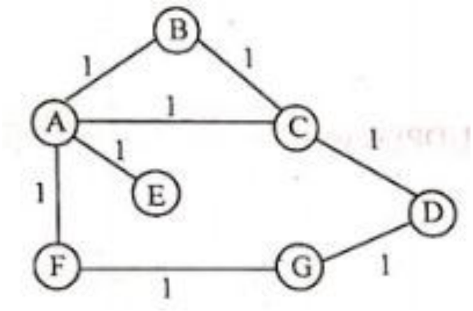
```
Distance_Vector_Routing ( )
{
    // Initialize (create initial vectors for the node)
    D[ myself ] = 0
    for (y = 1 to N)
    {
        if (y is a neighbor)
            D[y] = c[ myself ][ y ]
        else
            D[y] = ∞
    }
  Send vector {D[1], D[2], ..., D[N]} to all neighbors
```

Distance Vector Routing Algorithm - Steps

```
…
    // Update (improve the vector with the vector received from a neighbor)
    repeat (forever)
    {
        wait (for a vector $D_w$ from a neighbor w or any change in the link)
        for (y = 1 to N)
        {
            $D[y] = \min ( D[y], (c[myself ][w] + D_w[y ]) )$ // Bellman-Ford equation
        }
        if (any change in the vector)
            send vector {D[1], D[2], ..., D[N]} to all neighbors
    }
} // End of Distance Vector
```

Distance Vector Routing Algorithm - Example

- By applying the algorithm, find the routing table maintained at node B.



Initial Distance Vectors

	A	B	C	D	E	F	G
A	0	1	1	∞	1	1	∞
B	1	0	1	∞	∞	∞	∞
C	1	1	0	1	∞	∞	∞
D	∞	∞	1	0	∞	∞	1
E	1	∞	∞	∞	0	∞	∞
F	1	∞	∞	∞	∞	0	1
G	∞	∞	∞	1	∞	1	0

Final Distance Vectors

	A	B	C	D	E	F	G
A	0	1	1	2	1	1	2
B	1	0	1	2	2	2	3
C	1	1	0	1	2	2	2
D	2	2	1	0	3	2	1
E	1	2	2	3	0	2	3
F	1	2	2	2	3	0	1
G	2	3	2	1	3	1	0

Routing Table of B

	B	Hop
A	1	A
B	0	B
C	1	C
D	2	C
E	2	A
F	2	A
G	3	A

Distance Vector Routing Algorithm

- Problem
    - Decrease in cost (good news) propagates quickly, but increase in cost (bad news) propagates very slowly.
    - Link breakage will be propagated slowly.
    - May lead to Count to Infinity problem.

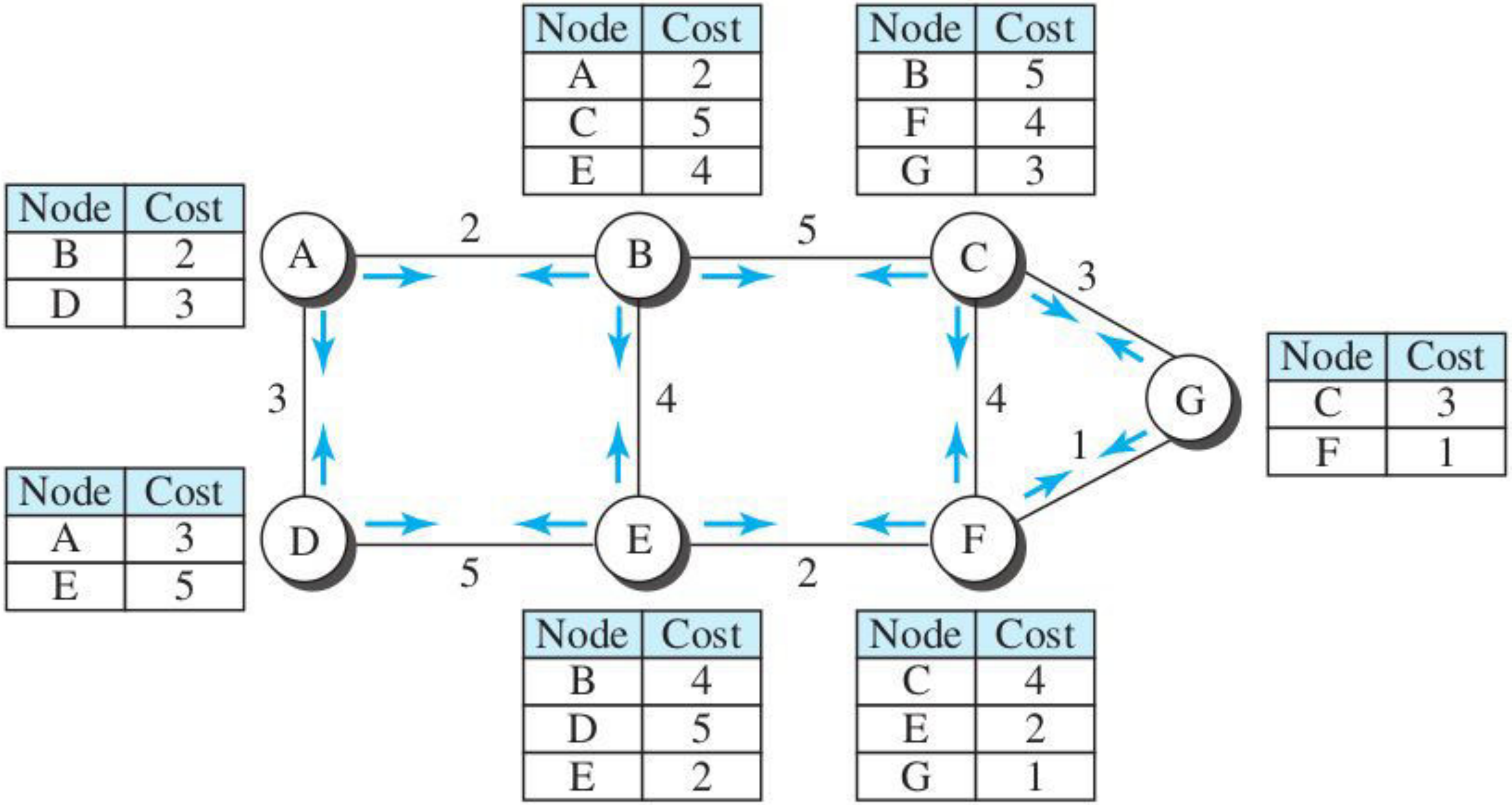
- Solution
    - Split Horizon
        - B sends a part of its distance vector to A, which contains the costs that are not obtained from A.
    - Poison Reverse:
        - B sends its distance vector to A with all entries obtained from A as infinity.

Link State (LS) Routing Algorithm

- States of the links are defined by their cost.
- If the cost is infinity, it means that the link is broken or doesn’t exist.
- Steps:
  1. First, each node finds the identity of all of its neighbors and the cost to them through greetings messages and makes a Link State Packet (**LSP**) with these two information.
  2. Each node sends its LSP to all the interfaces except through which the information came. This process is called **flooding**.

Link State (LS) Routing Algorithm

3. On receiving an LSP, each node compares it with its own LSP and keep the newer one discarding the other.



Node	Cost
B	2
D	3

Node	Cost
A	2
C	5
E	4

Node	Cost
B	5
F	4
G	3

Node	Cost
C	3
F	1

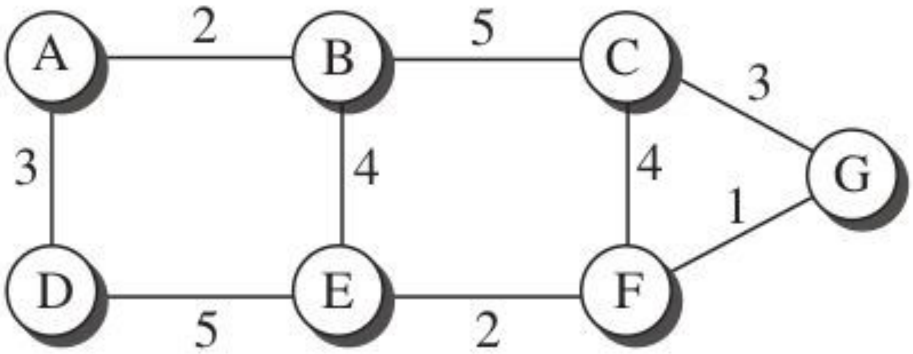
Node	Cost
A	3
E	5

Node	Cost
B	4
D	5
E	2

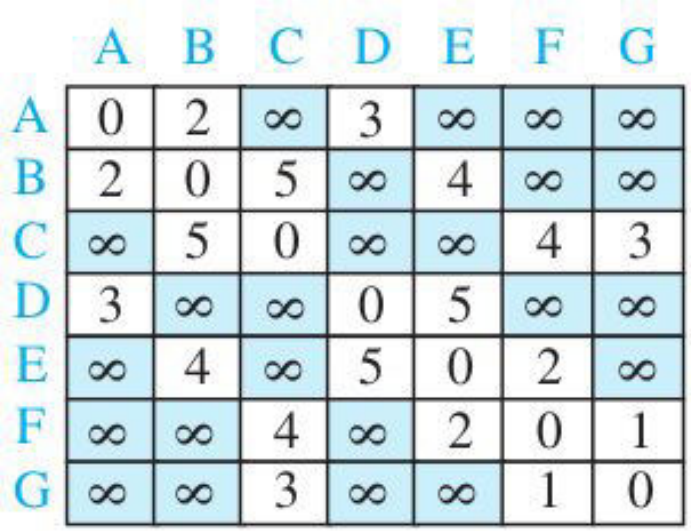
Node	Cost
C	4
E	2
G	1

Link State (LS) Routing Algorithm

4. Within some time, the flooding stops as all the nodes gets the information about all the other nodes in the network. Then each node creates a Link State Database (LSDB) with this information and the LSDB will be the same for all the nodes.



a. The weighted graph



	A	B	C	D	E	F	G
A	0	2	∞	3	∞	∞	∞
B	2	0	5	∞	4	∞	∞
C	∞	5	0	∞	∞	4	3
D	3	∞	∞	0	5	∞	∞
E	∞	4	∞	5	0	2	∞
F	∞	∞	4	∞	2	0	1
G	∞	∞	3	∞	∞	1	0

b. Link state database

Link State (LS) Routing Algorithm

4. Each node then computes its own least cost tree using Dijkstra's Algorithm. The algorithm is;

(1) The node chooses itself as the root of the tree, creating a tree with a single node, and sets the total cost of each node based on the information in the LSDB.

```
Dijkstra's Algorithm ( )
 {          // Initialization
     Tree = {root}
          // Tree is made only of the root
     for (y = 1 to N)              // N is the number of nodes
     {
          if (y is the root)
               D[y] = 0             // D[y] is shortest distance from root to node y
          else if (y is a neighbor)
               D[y] = c[root][y]    // c[x][y] is cost between nodes x and y in LSDB
          else
               D[y] = ∞
      }
```

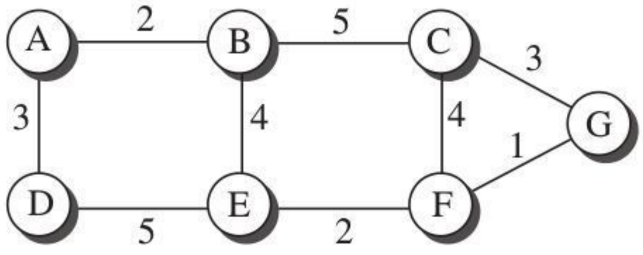
Link State (LS) Routing Algorithm

4. Each node then computes its own least cost tree using Dijkstra's Algorithm. The algorithm is;

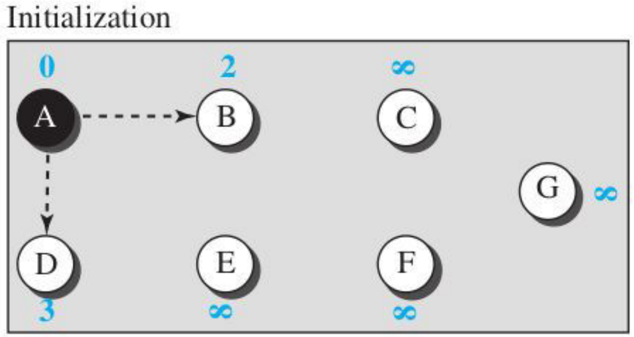
```
…
  // Calculation
repeat
{
      find a node w, with D[w] minimum among all nodes not in the Tree
      Tree = Tree ∪ {w}          // Add w to tree
            // Update distances for all neighbors of w
      for (every node x, which is a neighbor of w and not in the Tree)
            D[x] = min { D[x], (D[w] + c[w][x]) }
} until (all nodes included in the Tree)
} // End of Dijkstra
```

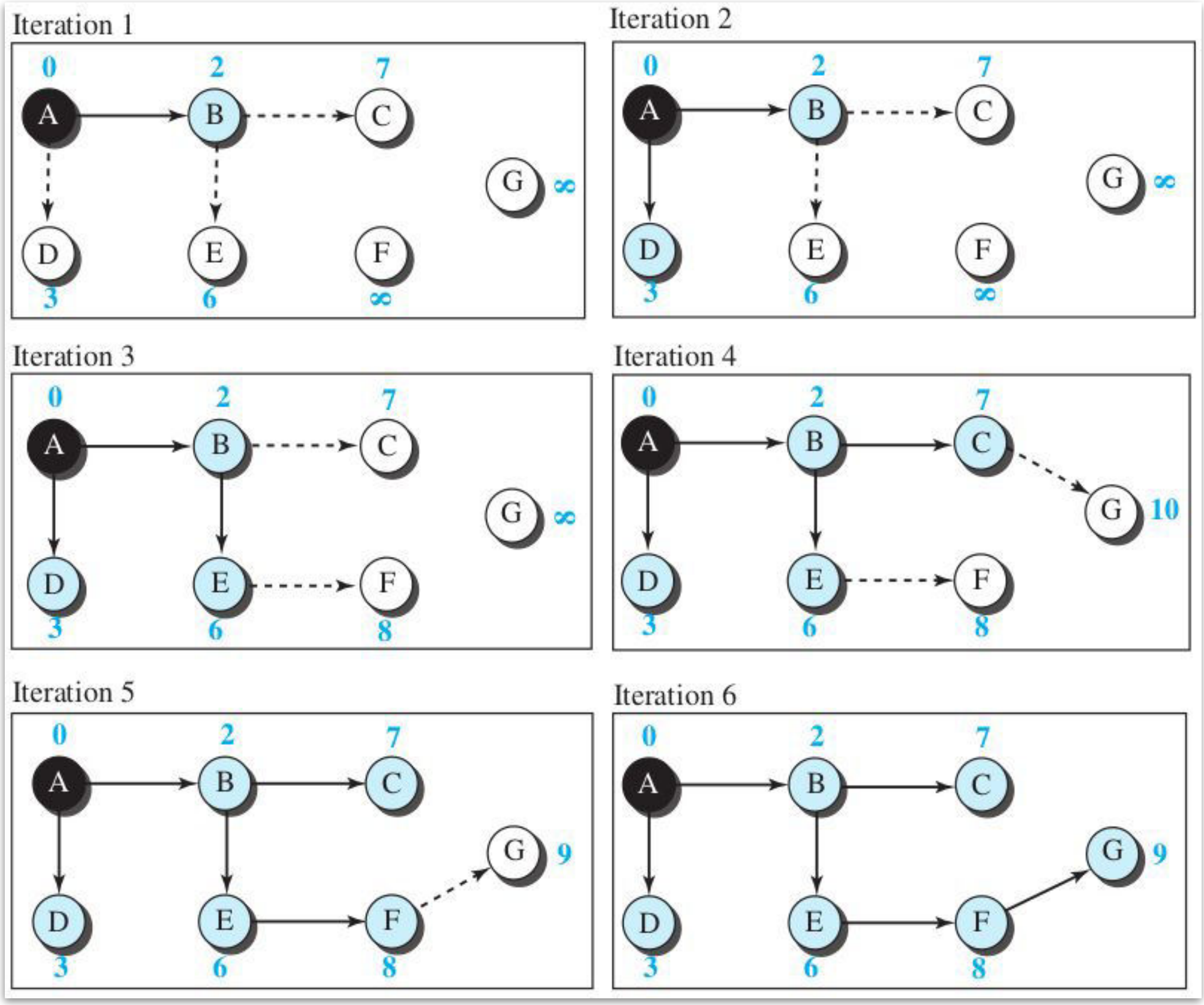
(2) The node selects one node, among all nodes not in the tree, which is closest to the root, and adds this to the tree. After this, the cost of all other nodes not in the tree needs to be updated. This process is repeated until all the nodes are added to the tree.

Link State (LS) Routing Algorithm



Tree creation in node A



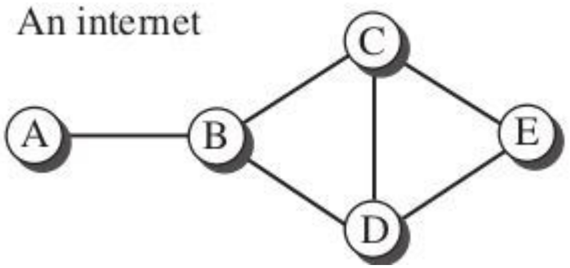


Path-Vector (PV) Routing

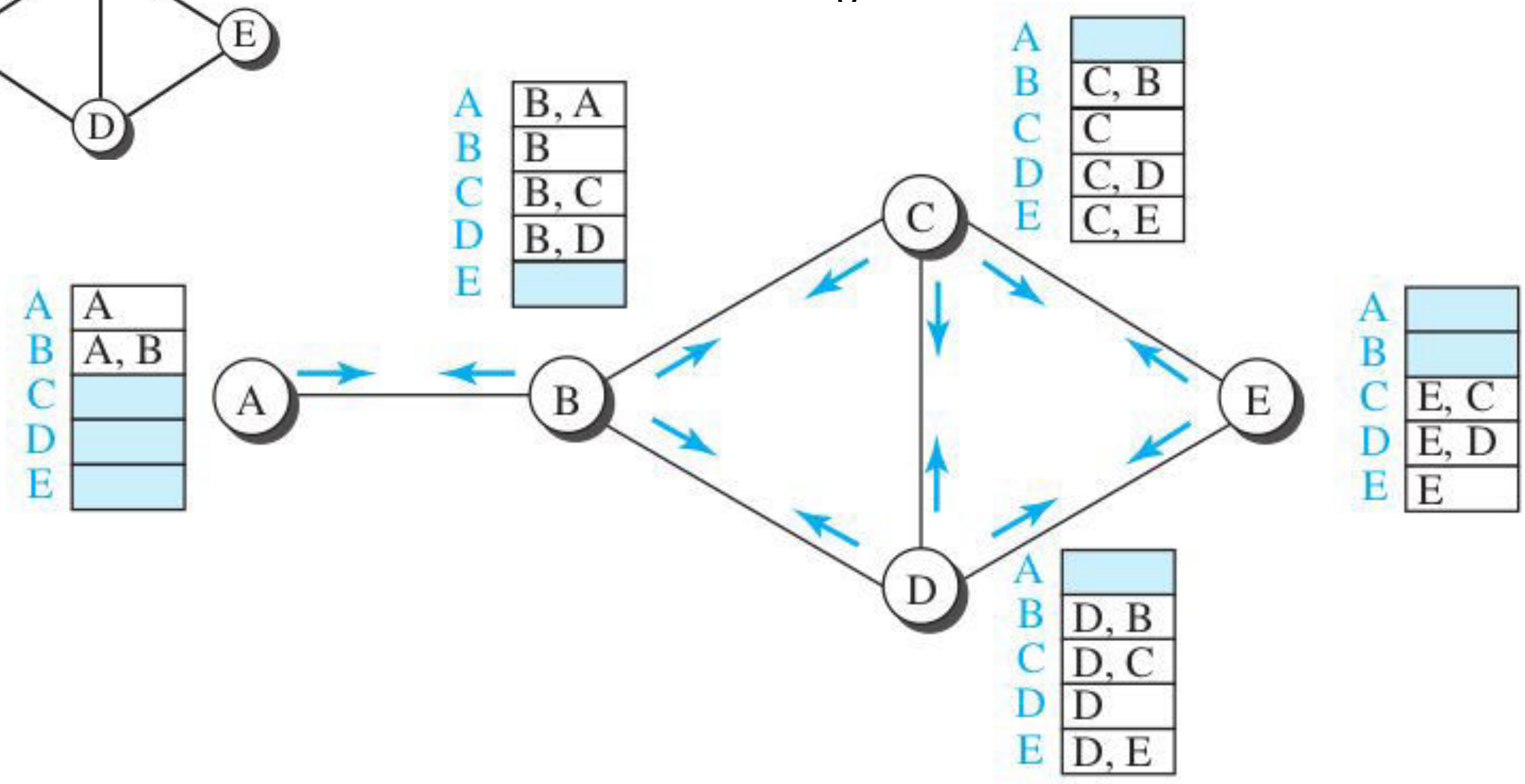
- Not based on least-cost path, but **best-cost** path.
- Does not have the drawbacks of DV and LS routing.
- Mainly used by **Internet Service Providers (ISP)**.
- Used when **certain policies are to be imposed on routers**; such as to avoid some path.
- The best route is determined by the source using the policy it imposes on the route. In other words, **the source can control the path**.
- Thus each router has its own spanning tree .
- A source may apply several policies at the same time.

Path-Vector (PV) Routing Algorithm

1. When a node is booted up, it creates a **path vector** based on the information from its neighbors through greetings; then passes these path vectors to its neighbors.



Path vectors at booting time



A	A
B	A, B
C	
D	
E	

A	B, A
B	B
C	B, C
D	B, D
E	

A	
B	C, B
C	C
D	C, D
E	C, E

A	
B	
C	E, C
D	E, D
E	E

A	
B	D, B
C	D, C
D	D
E	D, E

Path-Vector (PV) Routing Algorithm

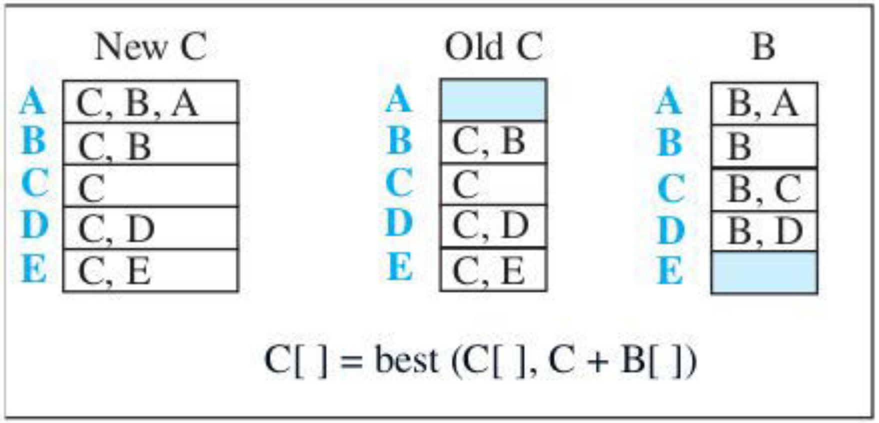
2. Then the neighbors update their vectors by using these informations and so on.

   The updation occurs based on the equation;

   Path(x, y) = best { Path(x, y), [(x + Path(v , y)] } for all v ’s in the internet.

   The operator (+) means to add x to the beginning of the path.

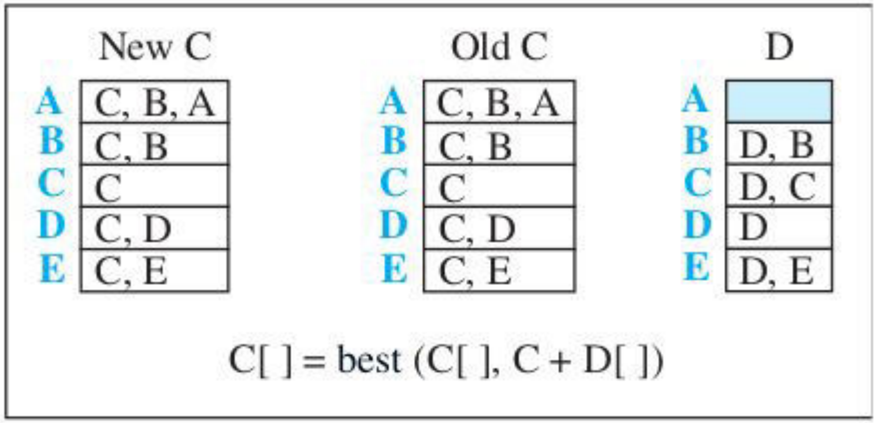
3. If Path (v, y) includes x, that path is discarded to avoid a loop in the path.



	New C	Old C	B
A	C, B, A		B, A
B	C, B	C, B	B
C	C	C	B, C
D	C, D	C, D	B, D
E	C, E	C, E	

C[ ] = best (C[ ], C + B[ ])

Event 1: C receives a copy of B’s vector



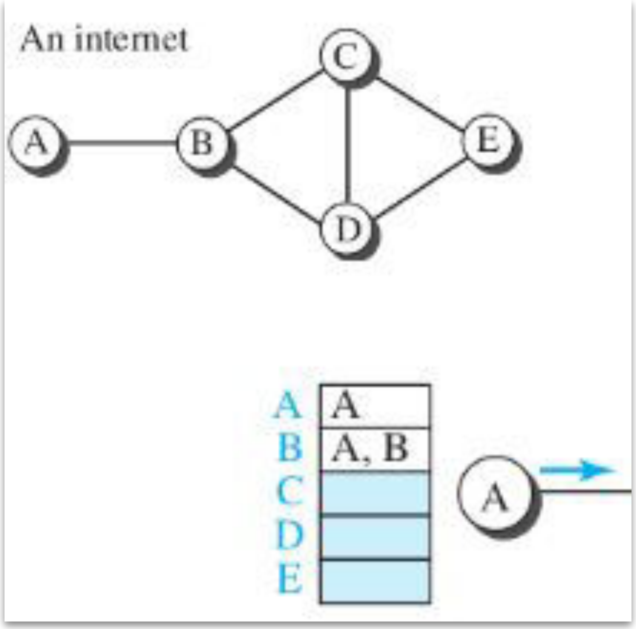
	New C	Old C	D
A	C, B, A	C, B, A	
B	C, B	C, B	D, B
C	C	C	D, C
D	C, D	C, D	D
E	C, E	C, E	D, E

C[ ] = best (C[ ], C + D[ ])

Event 2: C receives a copy of D’s vector

Path-Vector (PV) Routing Algorithm

```
Path_Vector_Routing ( )
{    // Initialization
     for (y = 1 to N)
     {
          if (y is myself)
               Path[y] = myself
          else if (y is a neighbor)
               Path[y] = myself + neighbor node
          else
               Path[y] = empty
     }
     Send vector { Path[1], Path[2], ..., Path[y] } to all neighbors
```

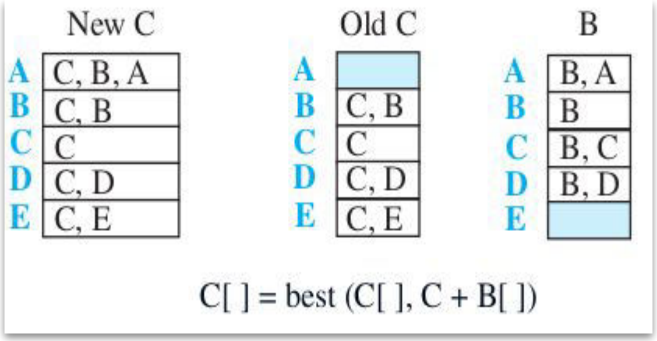


A	A
B	A, B
C	
D	
E	

Path-Vector (PV) Routing Algorithm

```
// Update
repeat (forever)
{
    wait (for a vector Path w from a neighbor w)
    for (y = 1 to N)
    {
        if (Path w includes myself)
            discard the path // Avoid any loop
        else
            Path[y] = best {Path[y], (myself + Path w [y])}
    }

    If (there is a change in the vector)
        Send vector {Path[1], Path[2], ..., Path[y]} to all neighbors
}
} // End of Path Vector
```



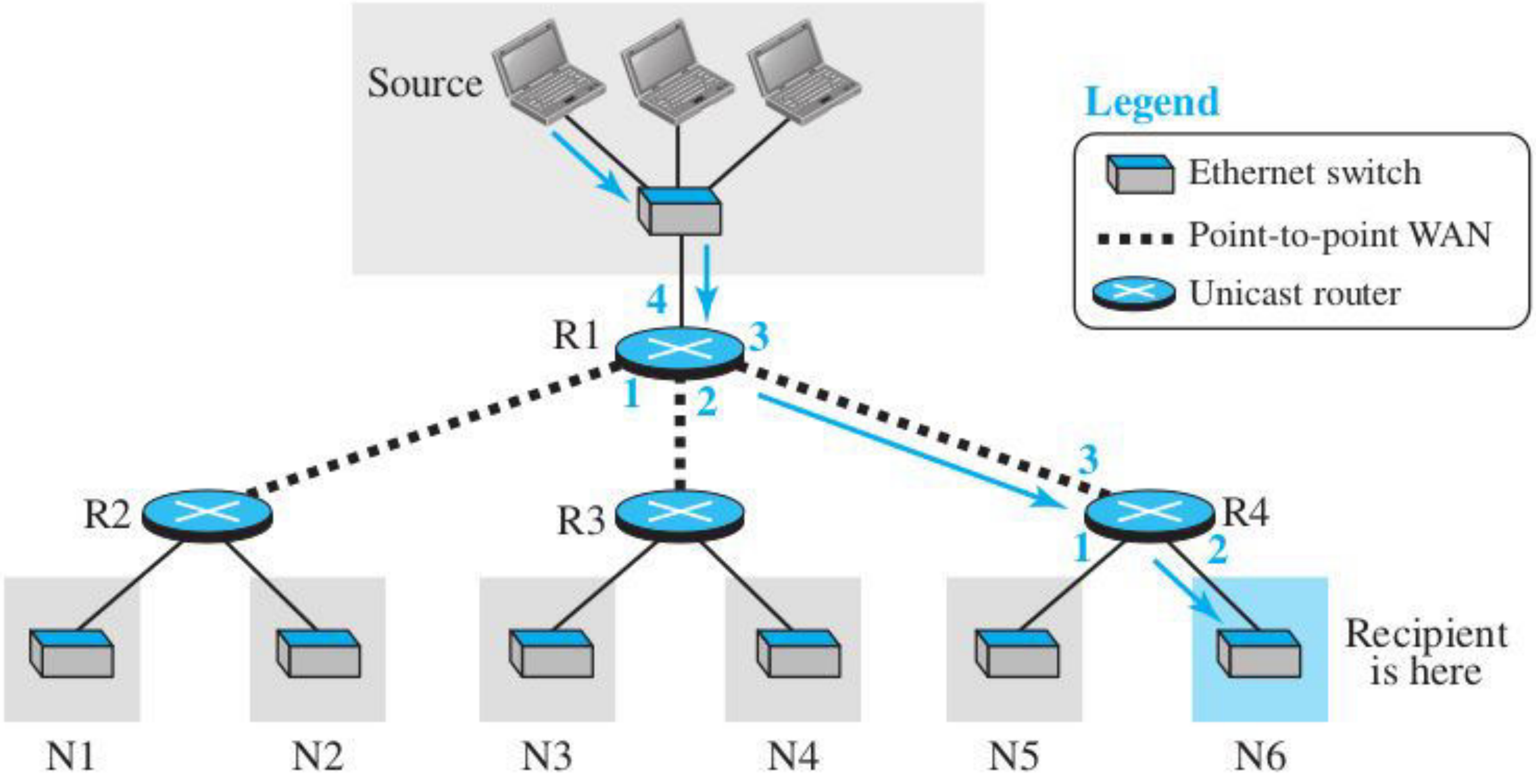
	New C	Old C	B
A	C, B, A		B, A
B	C, B	C, B	B
C	C	C	B, C
D	C, D	C, D	B, D
E	C, E	C, E	

C[ ] = best (C[ ], C + B[ ])

Unicasting, Multicasting and Broadcasting

Unicasting:

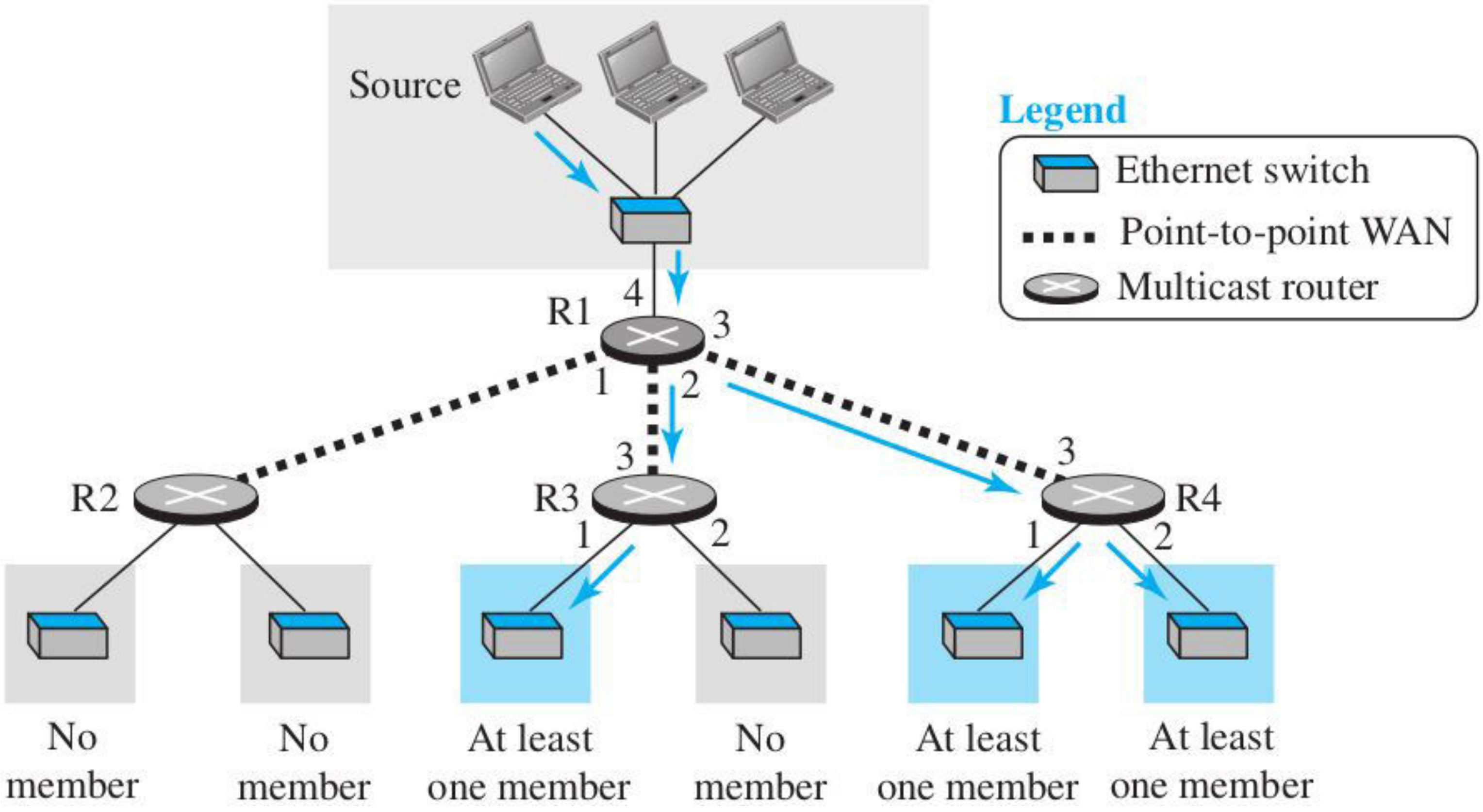
- One source and one destination network.
- One to one relationship between source and destination.
- Each router in the path of the datagram forward the packet to <u>one and only one</u> of its interfaces.



Unicasting, Multicasting and Broadcasting

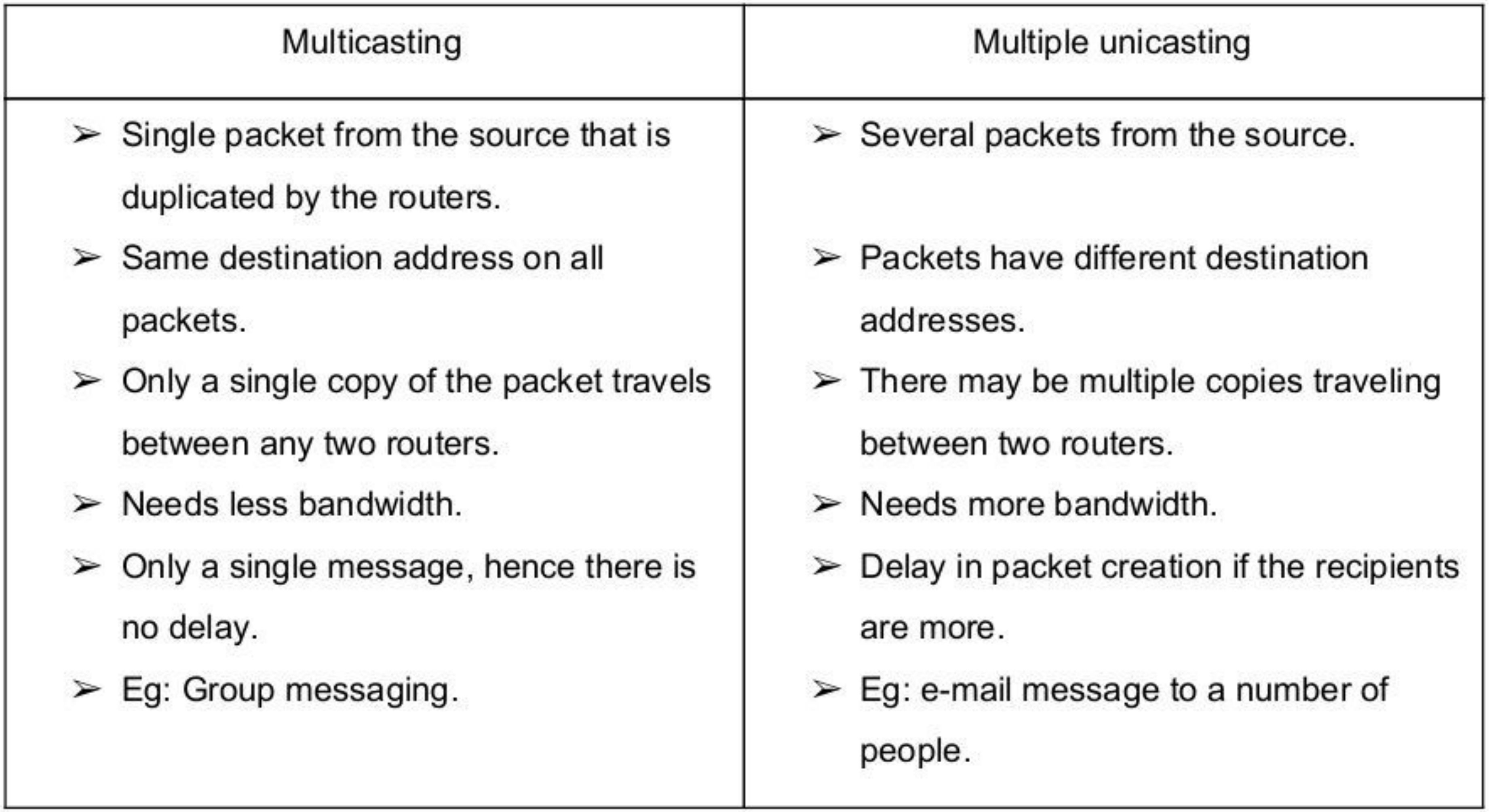
Multicasting:

- One source and a group of destinations.
- The relationship is one to many.
- The source address is a unicast address, but the destination address is a group address.



Unicasting, Multicasting and Broadcasting

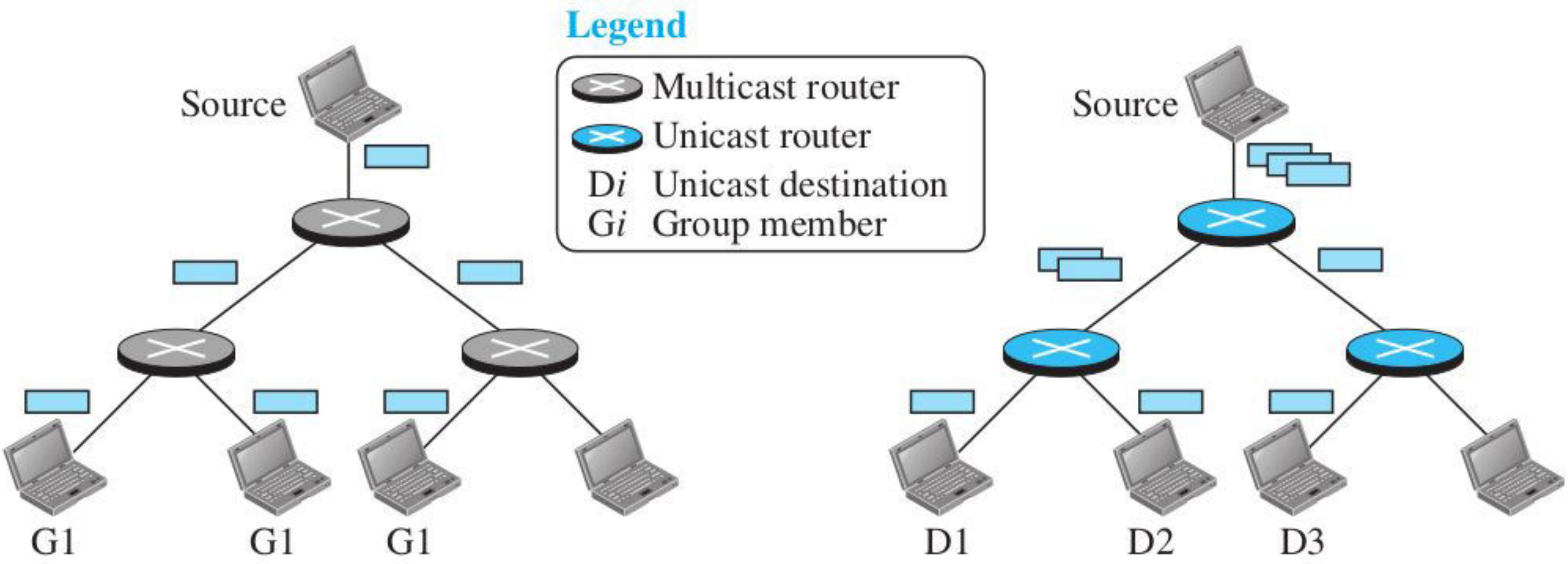
Multicasting versus Multiple Unicasting:



Multicasting	Multiple unicasting
➢ Single packet from the source that is duplicated by the routers. ➢ Same destination address on all packets. ➢ Only a single copy of the packet travels between any two routers. ➢ Needs less bandwidth. ➢ Only a single message, hence there is no delay. ➢ Eg: Group messaging.	➢ Several packets from the source. ➢ Packets have different destination addresses. ➢ There may be multiple copies traveling between two routers. ➢ Needs more bandwidth. ➢ Delay in packet creation if the recipients are more. ➢ Eg: e-mail message to a number of people.

Unicasting, Multicasting and Broadcasting

Multicasting versus Multiple Unicasting:



a. Multicasting

b. Multiple unicasting

Unicasting, Multicasting and Broadcasting

Multicasting Applications:

❏ Access to Distributed Databases

❏ Information Dissemination: Different types of business groups

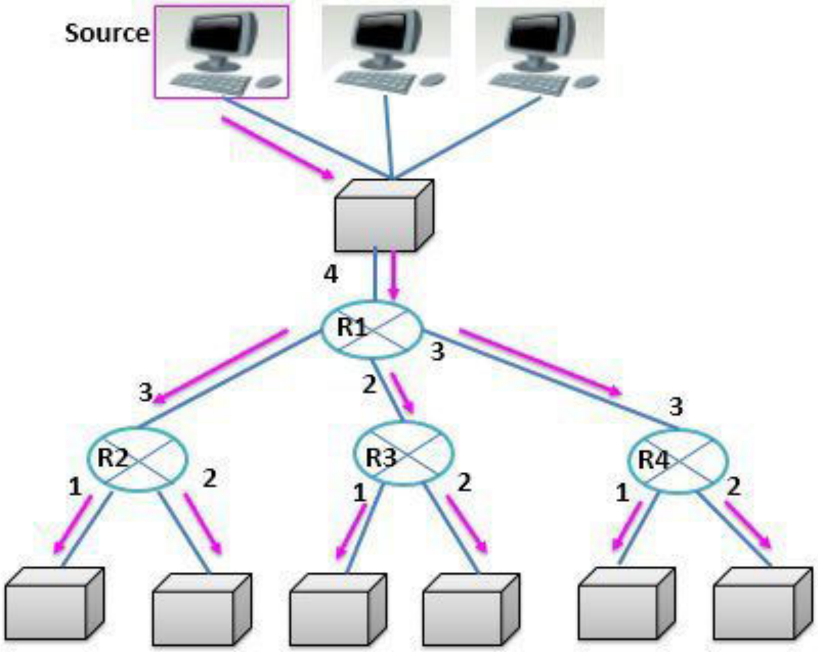
❏ Teleconferencing.

❏ Distance Learning.

Unicasting, Multicasting and Broadcasting

Broadcasting

- One-to-all communication.
- A host sends a packet to all hosts in an internet.
- Not generally used as it may create a huge volume of traffic and use a huge amount of bandwidth.
- Partial broadcasting is done in the Internet.
- Controlled broadcasting may also be done in a domain mostly as a step to achieve multicasting.



End of Part 3 of Module 2

Thank You

Diploma in Computer Hardware Engineering

COMPUTER NETWORKS

(5151 Rev 2015)

Module 3 - Part 1

Presenter: Sreejesh NG
Lecturer in Computer Hardware Engineering
Government Polytechnic College, Cherthala

Ref: Data Communications and Networking, 5E by Forouzan

Course General Outcomes:

Sl.	G.O	On completion of this course the student will be able :
1	1	To Understand the concept of TCP/IP Protocol
2	1	To Understand the concept of Network Layer
3	1	To Understand the concept of Transport Layer
4	1	To Understand the concept of Application Layer

Specific Outcomes:

3.1 Understand Transport Layer

- 3.1.1 Explain Transport layer services
- 3.1.2 Explain Transport layer protocols
- 3.1.3 Explain User Datagram Protocol (UDP).
- 3.1.4 Explain Transmission Control Protocol (TCP).
- 3.1.5 Describe Stream Control Transmission Protocol (SCTP).

MODULE III - TRANSPORT LAYER

- Transport layer services – Process-to-process communication, encapsulation and decapsulation, pushing, flow control, error control, congestion control, connectionless and connection oriented protocols
- Transport layer protocols – simple, stop and wait, go back-N, selective repeat, piggy backing
- UDP – user datagram, services, applications
- TCP – services, features, segment, connection
- SCTP – services, features

Ref: Data Communications and Networking - Behrouz A. Forouzan -
McGraw Hill Edn.- Fifth Edition

TRANSPORT LAYER

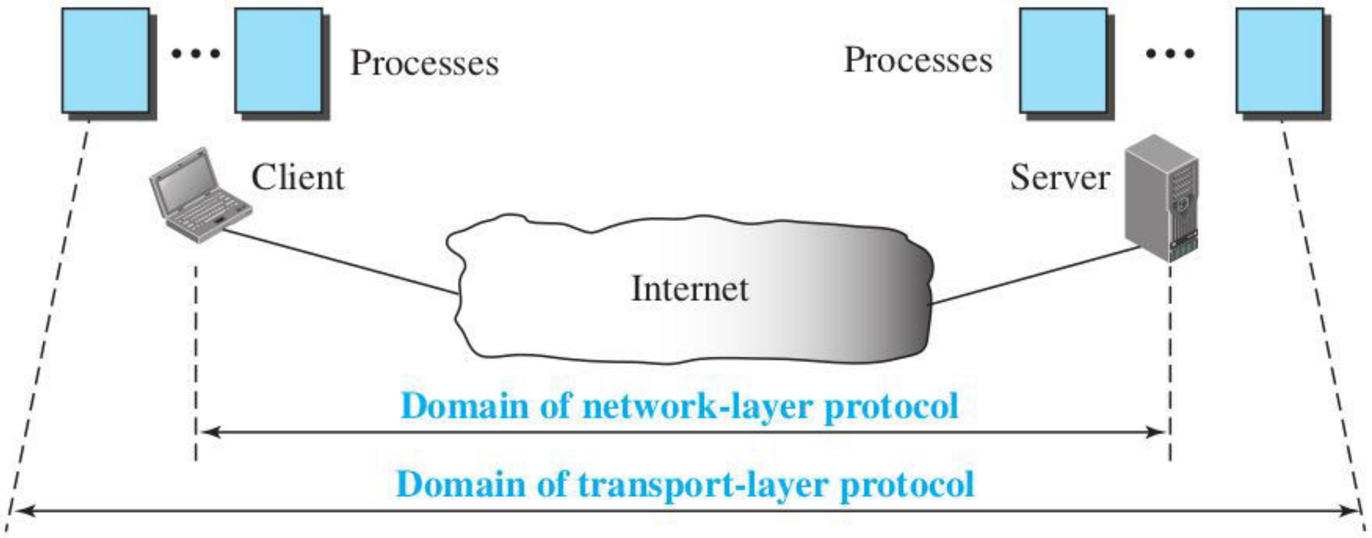
- Provides services to the application layer; it receives services from the network layer.
- Provides a process-to-process communication between two application layers

Services

1. Process-to-Process Communication
2. Addressing through Port Numbers
3. Encapsulation and Decapsulation
4. Multiplexing and Demultiplexing
5. Flow Control
6. Error Control
7. Congestion Control
8. Connectionless and Connection-oriented services.

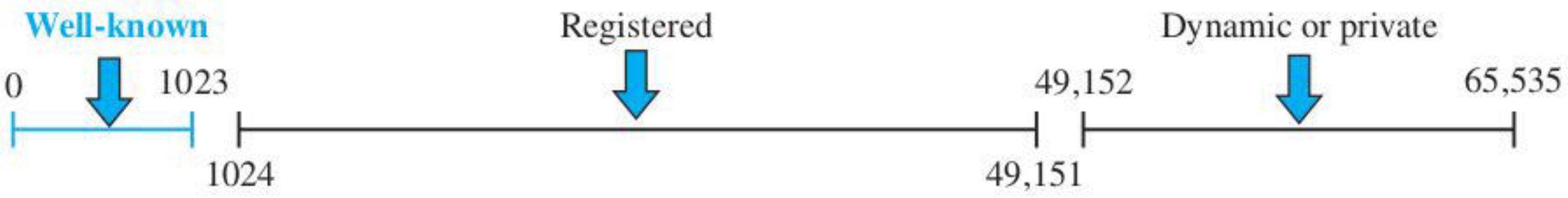
Process-to-Process Communication

- A process is an application-layer entity (running program) that uses the services of the transport layer.
- Difference between host-to-host communication and process-to-process communication?


Processes
Processes
Client
Server
Internet
Domain of network-layer protocol
Domain of transport-layer protocol

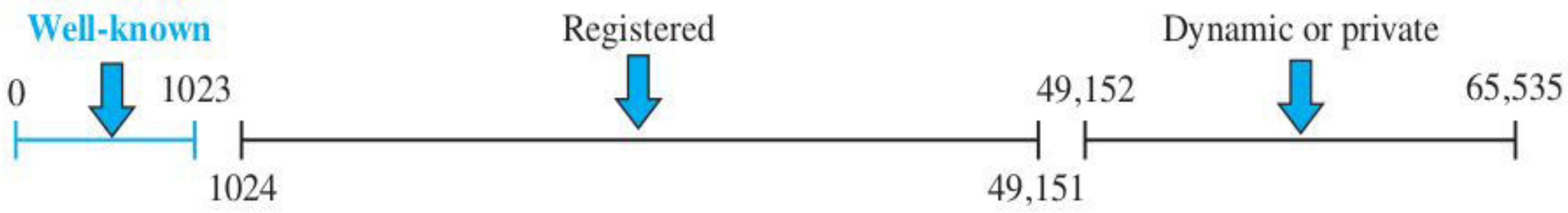
Addressing

- Hosts are identified by IP addresses, and the processes are identified by **port numbers**.
- Port numbers are of **16 bits** (0 - 65535).
- Most process-to-process communication happens through **client-server** paradigm.
- Client and server processes are identified by the port numbers.
- Ports division by ICANN (Internet Corporation for Assigned Names and Numbers):



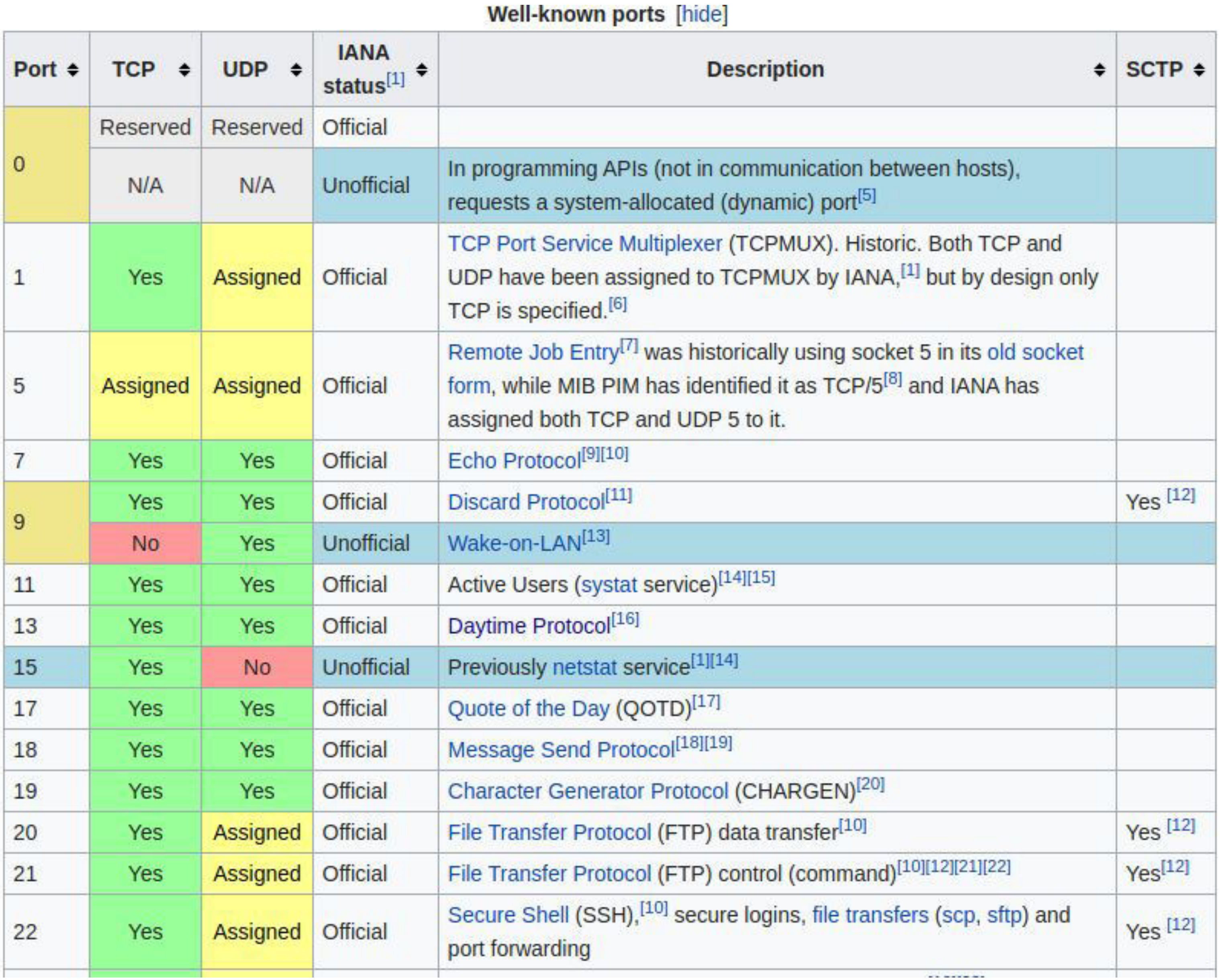
Addressing

- Ports division by ICANN (Internet Corporation for Assigned Names and Numbers):



- **Well-known ports**: 0-1023. Assigned and controlled by ICANN.
    - Eg: Port 80 - http server
- **Registered ports**
- **Dynamic ports**

Addressing: Well-known ports list - Wikipedia

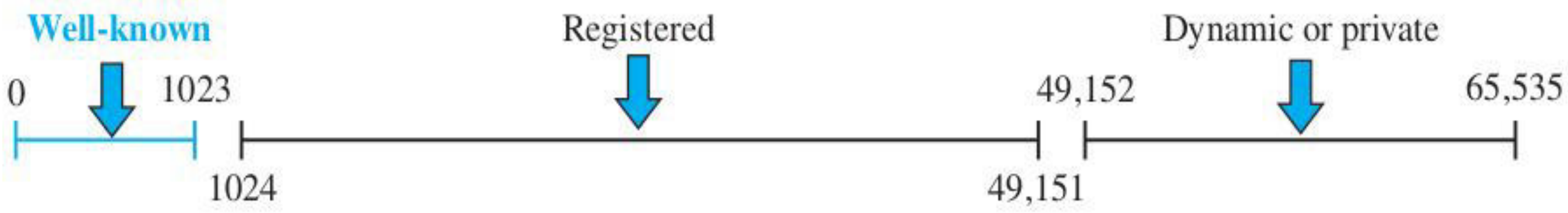


Well-known ports [hide]

Port	TCP	UDP	IANA status[1]	Description	SCTP
0	Reserved	Reserved	Official		
	N/A	N/A	Unofficial	In programming APIs (not in communication between hosts), requests a system-allocated (dynamic) port[5]	
1	Yes	Assigned	Official	TCP Port Service Multiplexer (TCPMUX). Historic. Both TCP and UDP have been assigned to TCPMUX by IANA,[1] but by design only TCP is specified.[6]	
5	Assigned	Assigned	Official	Remote Job Entry[7] was historically using socket 5 in its old socket form, while MIB PIM has identified it as TCP/5[8] and IANA has assigned both TCP and UDP 5 to it.	
7	Yes	Yes	Official	Echo Protocol[9][10]	
9	Yes	Yes	Official	Discard Protocol[11]	Yes [12]
	No	Yes	Unofficial	Wake-on-LAN[13]	
11	Yes	Yes	Official	Active Users (systat service)[14][15]	
13	Yes	Yes	Official	Daytime Protocol[16]	
15	Yes	No	Unofficial	Previously netstat service[1][14]	
17	Yes	Yes	Official	Quote of the Day (QOTD)[17]	
18	Yes	Yes	Official	Message Send Protocol[18][19]	
19	Yes	Yes	Official	Character Generator Protocol (CHARGEN)[20]	
20	Yes	Assigned	Official	File Transfer Protocol (FTP) data transfer[10]	Yes [12]
21	Yes	Assigned	Official	File Transfer Protocol (FTP) control (command)[10][12][21][22]	Yes[12]
22	Yes	Assigned	Official	Secure Shell (SSH),[10] secure logins, file transfers (scp, sftp) and port forwarding	Yes [12]

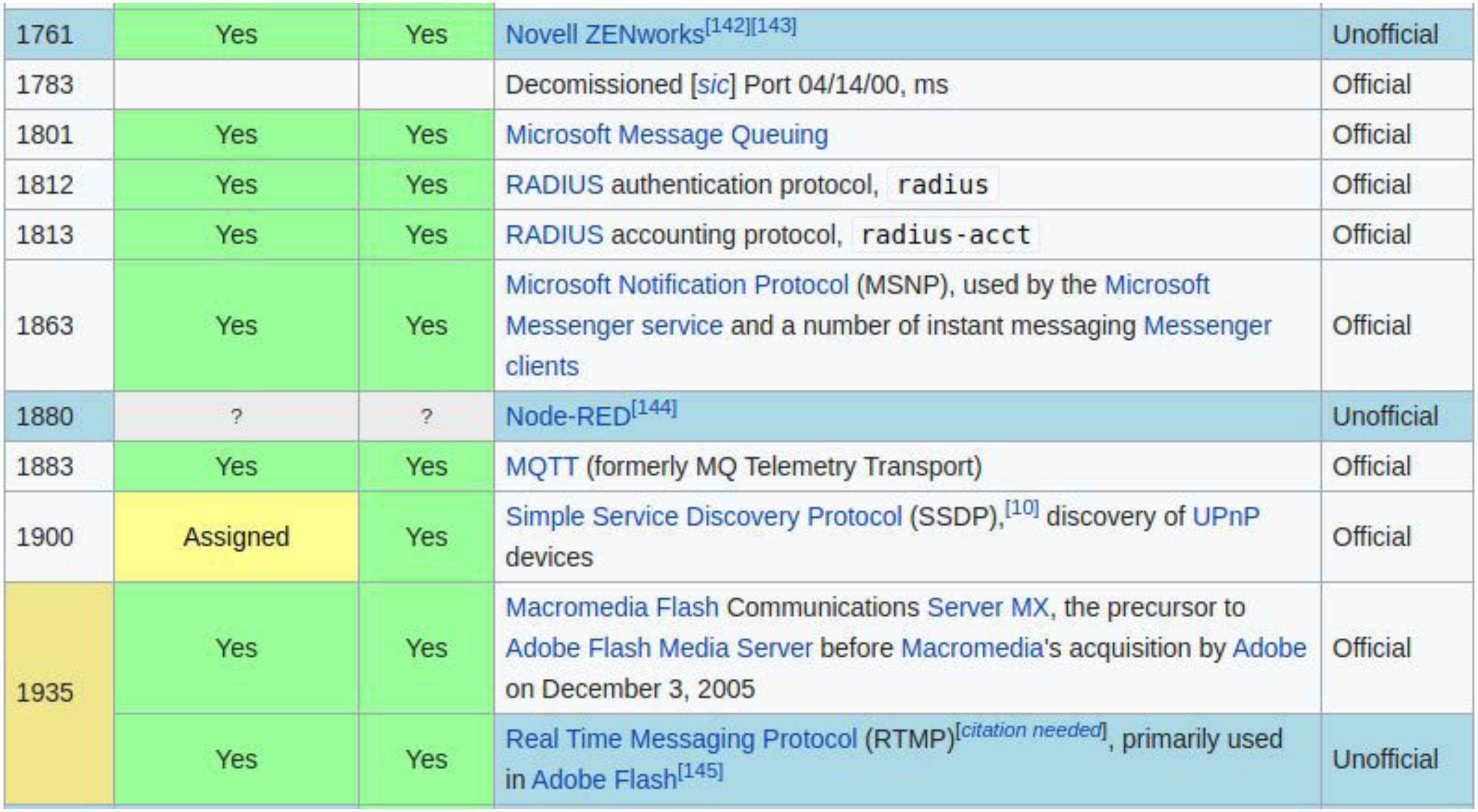
Addressing

- Ports division by ICANN (Internet Corporation for Assigned Names and Numbers):



- **Well-known ports**: 0-1023. Assigned and controlled by ICANN.
  - Eg: Port 80 - http server
- **Registered ports**: 1024 - 49151. They can only be registered with ICANN to prevent duplication.
  - Eg: Port 2196 - Apple Push Notification Service, feedback service
- **Dynamic ports**

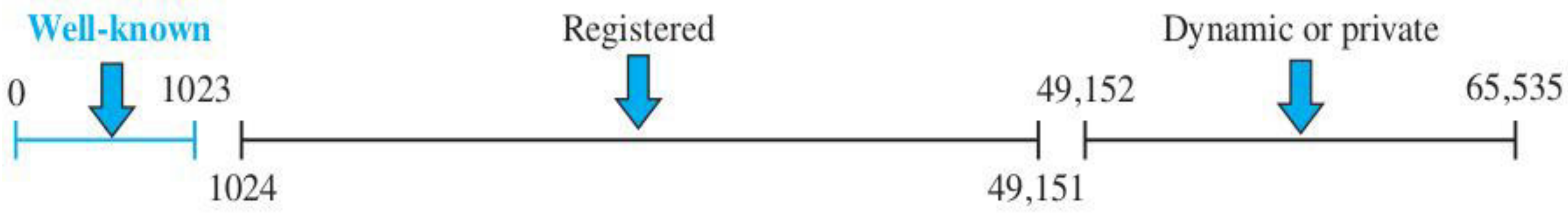
Addressing: Registered ports list - Wikipedia



1761	Yes	Yes	Novell ZENworks[142][143]	Unofficial
1783			Decomissioned [*sic*] Port 04/14/00, ms	Official
1801	Yes	Yes	Microsoft Message Queuing	Official
1812	Yes	Yes	RADIUS authentication protocol, `radius`	Official
1813	Yes	Yes	RADIUS accounting protocol, `radius-acct`	Official
1863	Yes	Yes	Microsoft Notification Protocol (MSNP), used by the Microsoft Messenger service and a number of instant messaging Messenger clients	Official
1880	?	?	Node-RED[144]	Unofficial
1883	Yes	Yes	MQTT (formerly MQ Telemetry Transport)	Official
1900	Assigned	Yes	Simple Service Discovery Protocol (SSDP),[10] discovery of UPnP devices	Official
1935	Yes	Yes	Macromedia Flash Communications Server MX, the precursor to Adobe Flash Media Server before Macromedia's acquisition by Adobe on December 3, 2005	Official
	Yes	Yes	Real Time Messaging Protocol (RTMP)[*citation needed*], primarily used in Adobe Flash[145]	Unofficial

Addressing

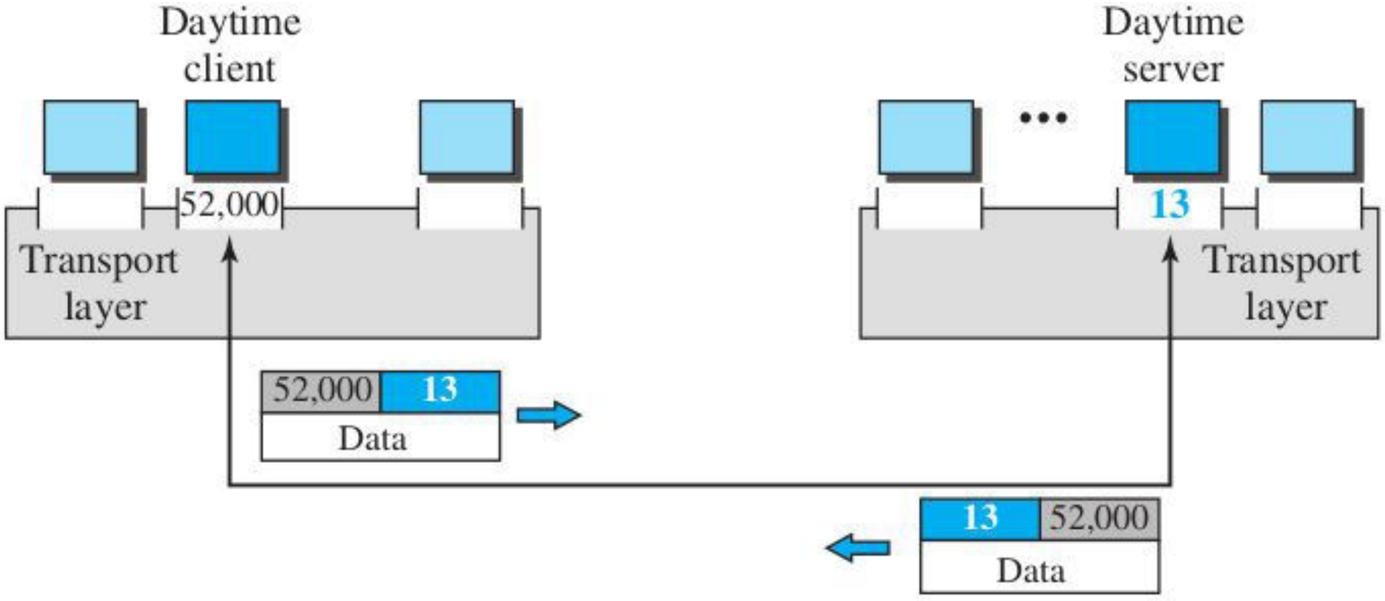
- Ports division by ICANN (Internet Corporation for Assigned Names and Numbers):



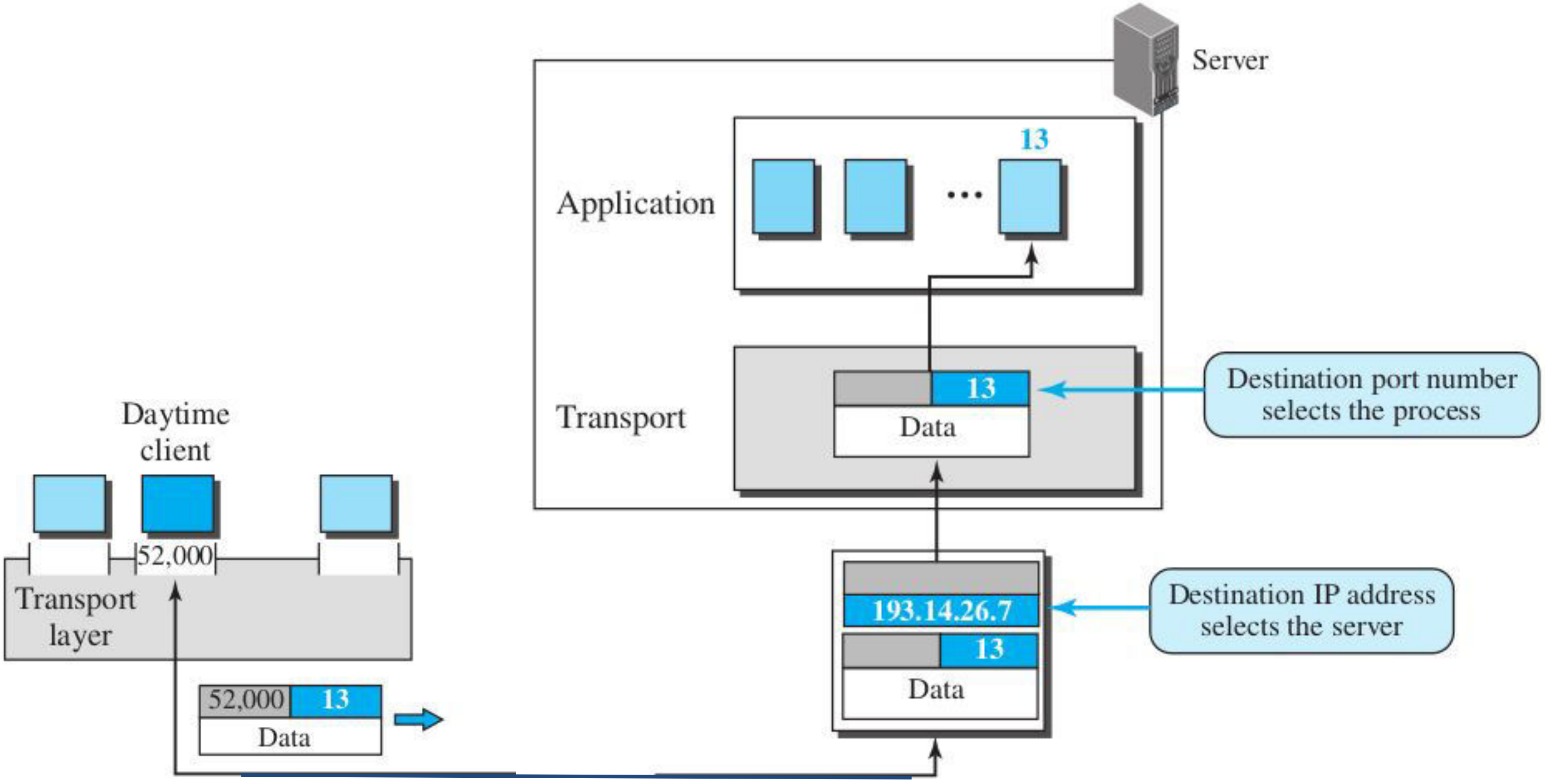
- **Well-known ports**: 0-1023. Assigned and controlled by ICANN.
  - Eg: Port 80 - http server
- **Registered ports**: 1024 - 49151. They can only be registered with ICANN to prevent duplication.
  - Eg: Port 2196 - Apple Push Notification Service, feedback service
- **Dynamic ports**: 49152 - 65535. Can be used as temporary or private port numbers.

Addressing

- Mostly, clients use ephemeral ports (also called short-lived ports) which will be greater than 1023.
    - For some services, clients use well-known ports.
        - Eg: DHCP client uses port 68
- Mostly, servers use well-known ports (upto 1023).
    - Eg: http server uses port 80, DHCP server use port 67



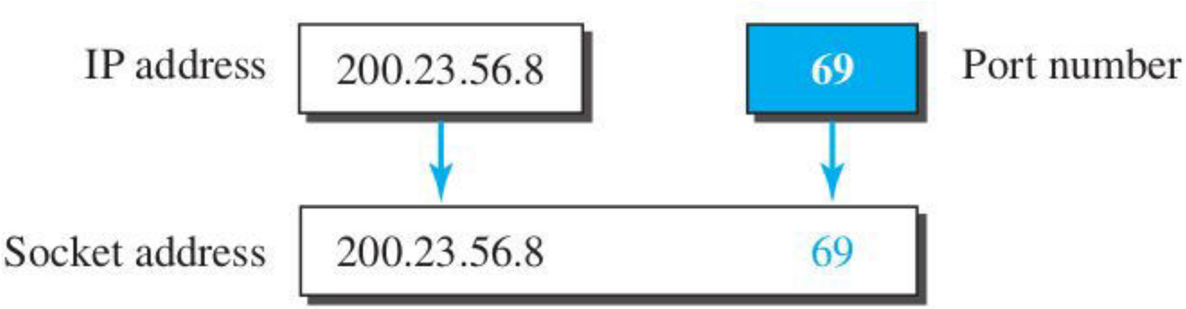
Addressing


Server
13
Application
...
Transport
13
Data
Destination port number selects the process
Daytime client
52,000
Transport layer
52,000
13
Data
193.14.26.7
13
Data
Destination IP address selects the server

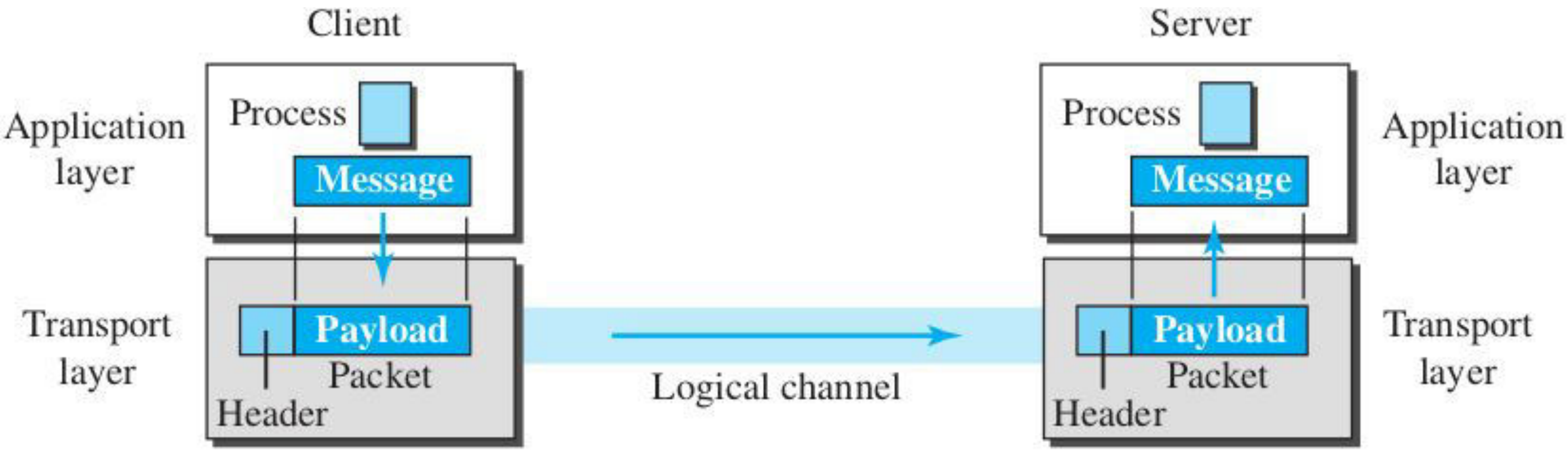
Addressing

Socket Addresses:

- It is the **combination of an IP address and a port number**
- Uniquely identifies a process.
- To use the services of the transport layer in the Internet, a pair of socket addresses are needed: the client socket address and the server socket address.
- These are part of network layer and transport layer headers.



Encapsulation and Decapsulation

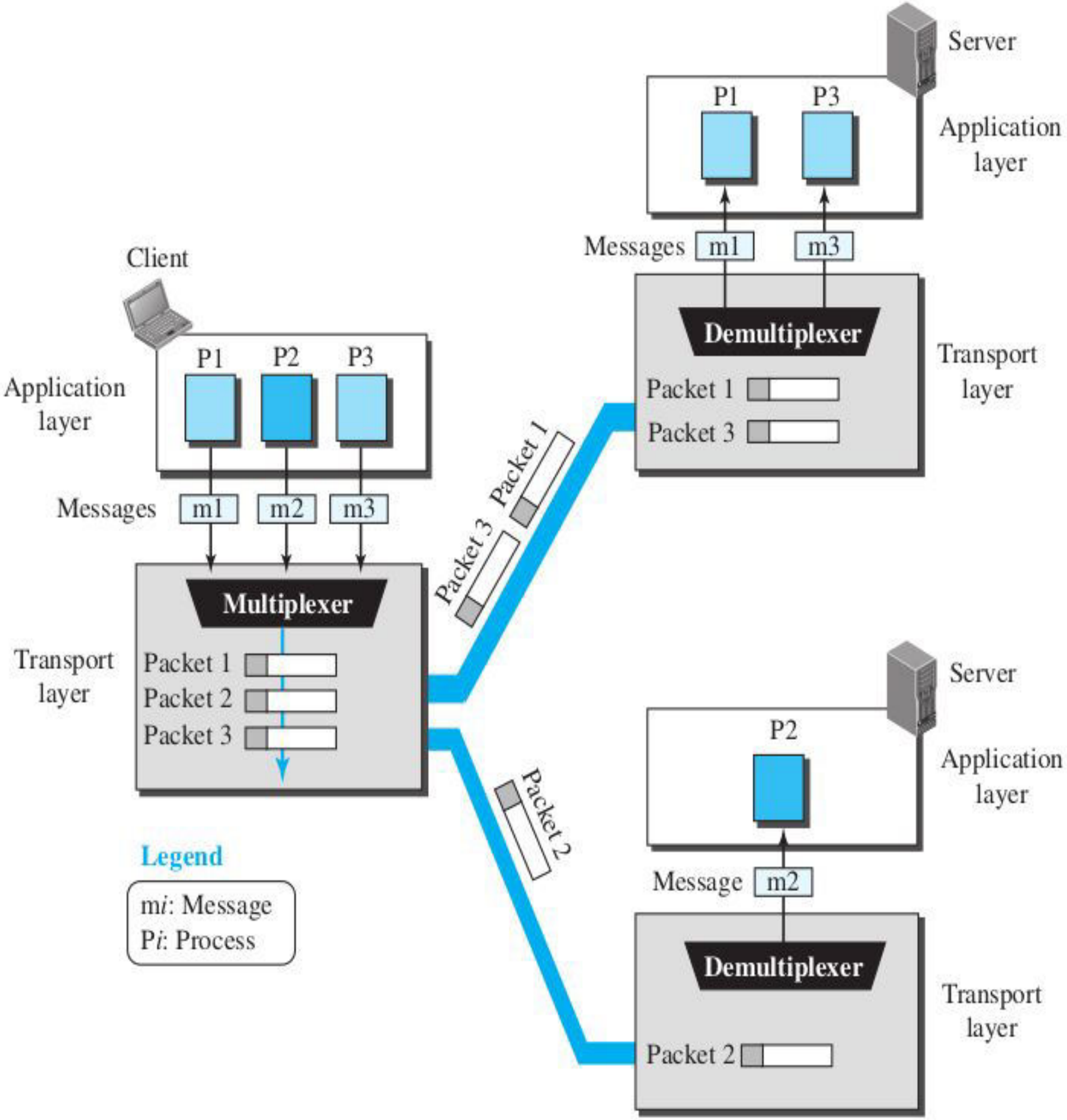


a. Encapsulation

b. Decapsulation

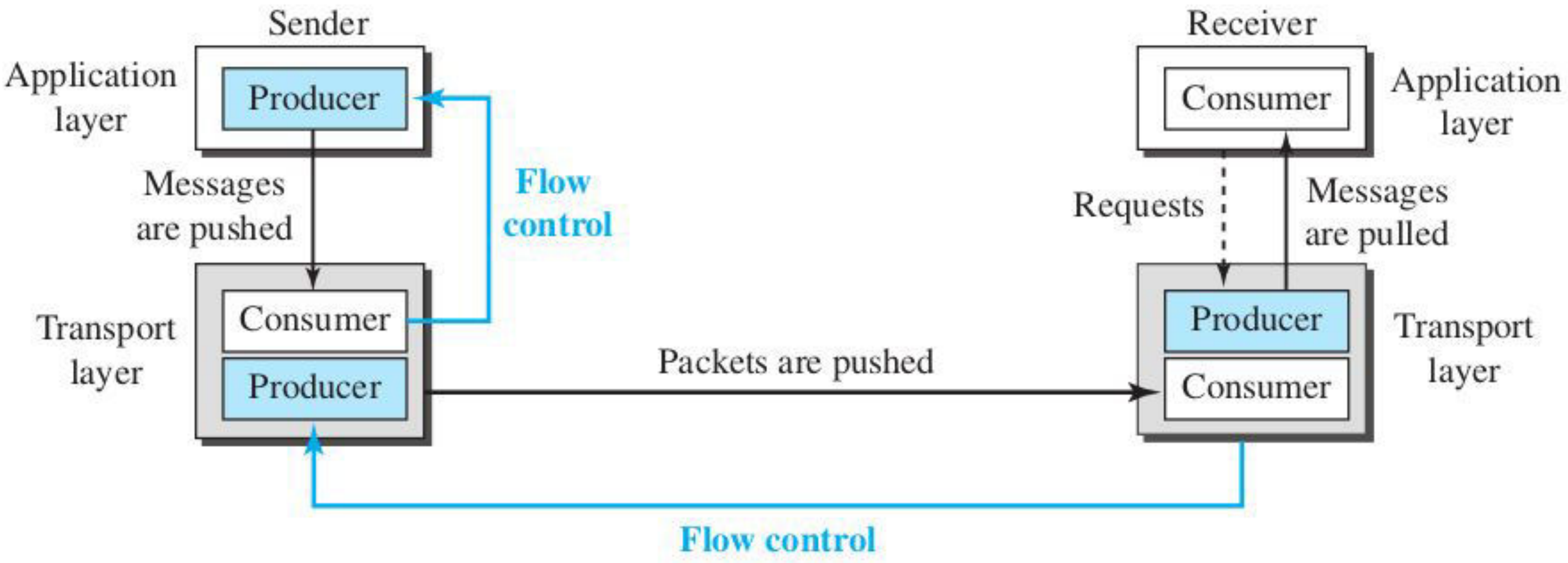
Multiplexing and Demultiplexing

- The transport layer at the source performs multiplexing (many to one).
- The transport layer at the destination performs demultiplexing (one to many).



Flow Control

- Four entities are present in a transport layer communication:
  - Sender process, sender transport layer, receiver transport layer, and receiver process.
- Sending and receiving transport layers use **buffers** to hold packets.



Error Control

- Involves the **sending** and **receiving transport layers**.
- Error control at the transport layer is responsible for;

  1. Detecting and discarding corrupted packets.
  2. Keeping track of lost and discarded packets and resending them.
  3. Recognizing duplicate packets and discarding them.
  4. Buffering out-of-order packets until the missing packets arrive.

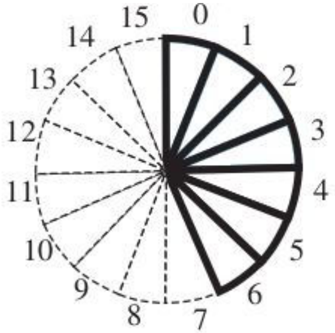
- Use **sequence number** for error control, which is modulo 2^m if there are m-bits in the sequence number field of the header.
- **Acknowledgement with timers** are also used to resend packets, if the packet is lost, corrupted, the acknowledgement itself is lost or delayed.

Combination of Flow and Error Control

- By using numbered buffers at the sender and receiver.
- Working:
  - **Sender:** When a packet is prepared to be sent, we use the number of the next free location, x, in the buffer as the sequence number of the packet. When the packet is sent, a copy is stored at memory location x, awaiting the acknowledgment from the other end. When an acknowledgment related to a sent packet arrives, the memory location is freed.
  - **Receiver**: When a packet with sequence number y arrives, it is stored at the memory location y until the application layer is ready to receive it. An acknowledgment can be sent to announce the arrival of packet y.
- This is implemented as **Sliding Window**.

Combination of Flow and Error Control

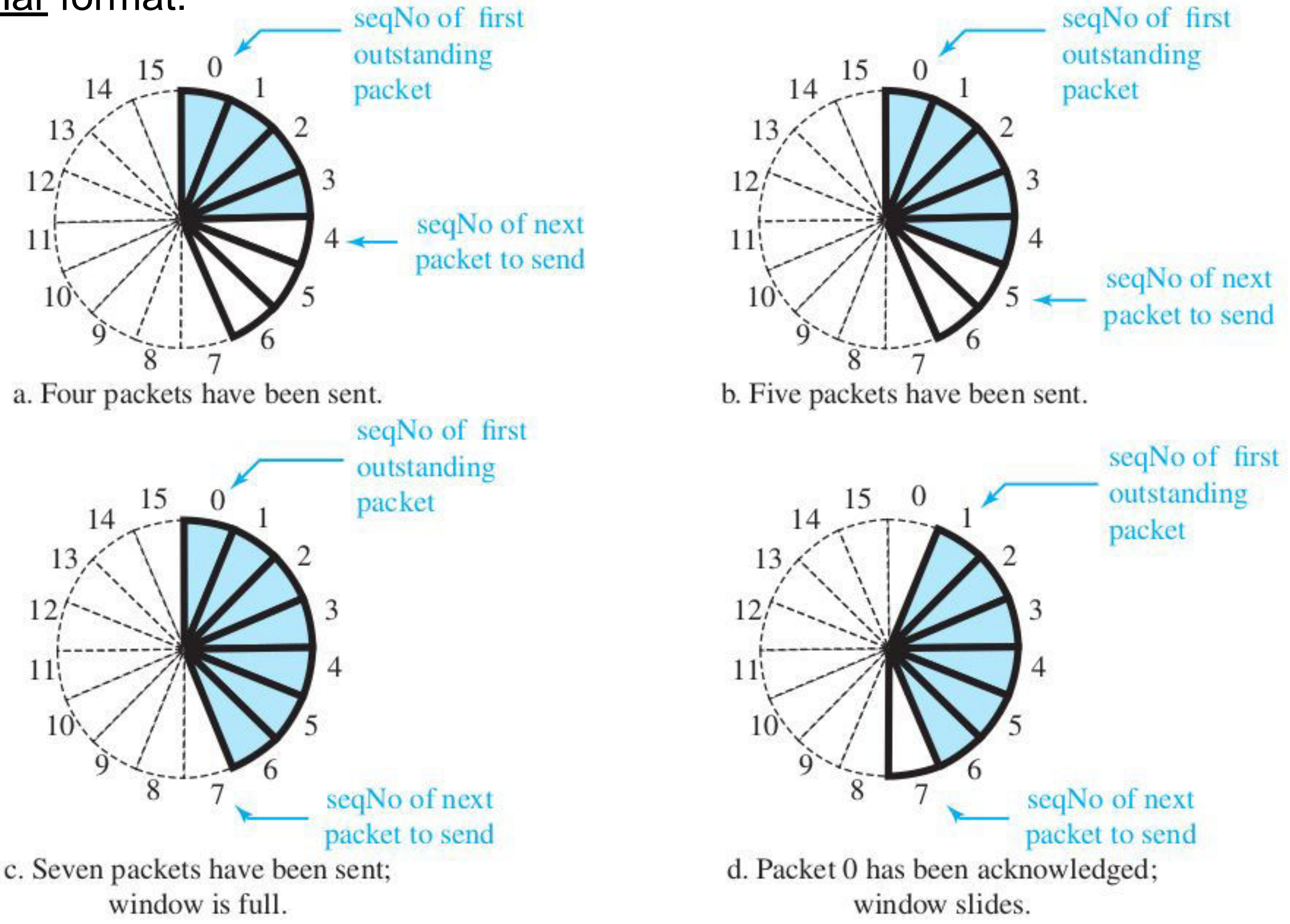
- **Sliding Window**:
  - Since the sequence numbers use modulo 2^m , a circle can represent the sequence numbers from 0 to $2^m - 1$.
  - The buffer is represented as a set of slices, called the sliding window, that occupies part of the circle at any time.
    - Eg: A buffer with modulo 2^4 sequence number and a sliding window of size 7



  - At the **sender** site, when a packet is sent, the corresponding slice is marked. When all the slices are marked, it means that the buffer is full and no further messages can be accepted from the application layer. When an acknowledgment arrives, the corresponding slice is unmarked. If some consecutive slices from the beginning of the window are unmarked, the window slides over the range of the corresponding sequence numbers to allow more free slices at the end of the window.

Combination of Flow and Error Control

- **Sliding Window**: 4-bit sequence numbers and 7-sized sliding window in circular format.



seqNo of first outstanding packet

seqNo of next packet to send

a. Four packets have been sent.

seqNo of first outstanding packet

seqNo of next packet to send

b. Five packets have been sent.

seqNo of first outstanding packet

seqNo of next packet to send

c. Seven packets have been sent; window is full.

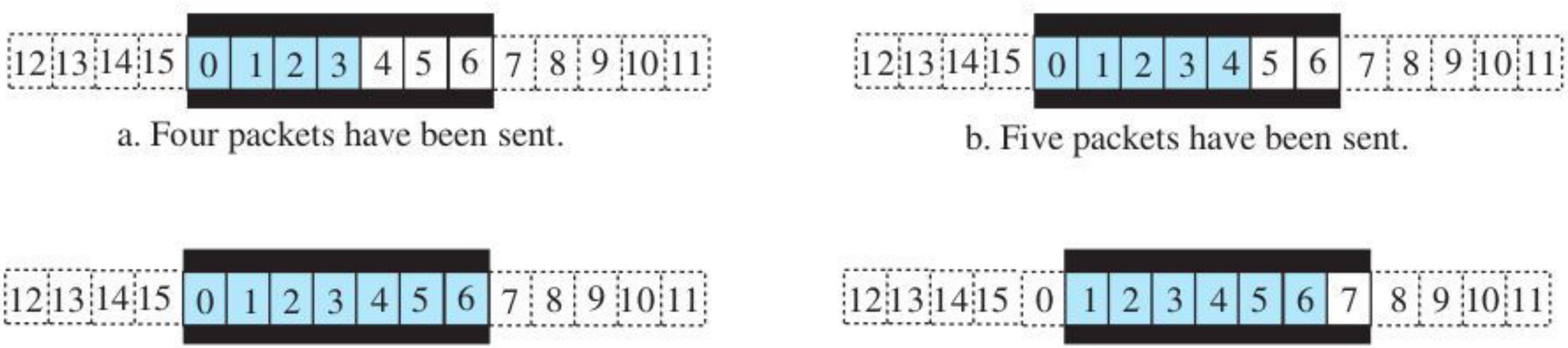
seqNo of first outstanding packet

seqNo of next packet to send

d. Packet 0 has been acknowledged; window slides.

Combination of Flow and Error Control

- **Sliding Window**: 4-bit sequence numbers and 7-sized sliding window in linear format.



a. Four packets have been sent.

b. Five packets have been sent.

c. Seven packets have been sent; window is full.

d. Packet 0 has been acknowledged; window slides.

Congestion Control

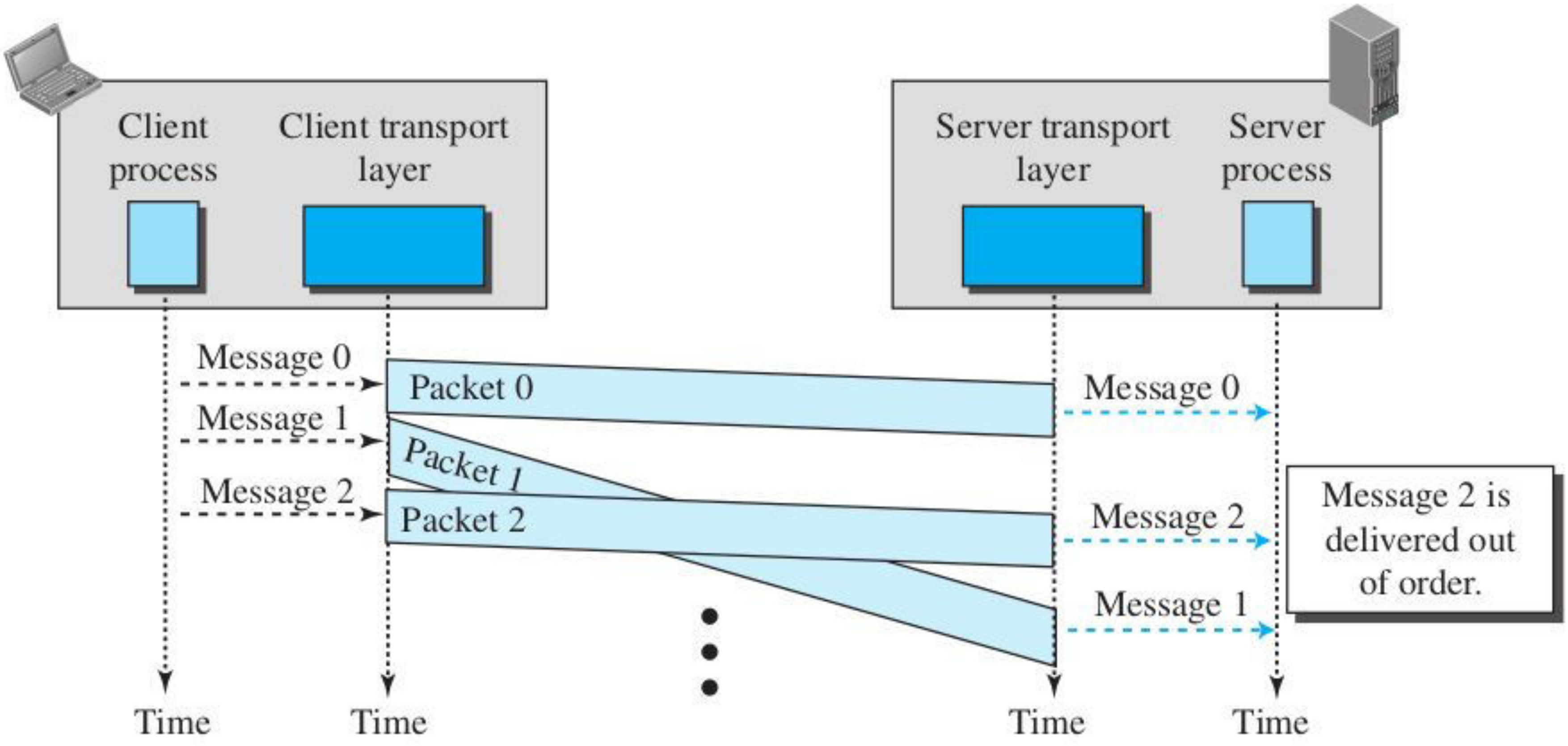
- Congestion occur if the *load* on the network is greater than the *capacity* of the network.
- Congestion at the transport layer is actually the result of congestion at the network layer.

Connectionless and Connection-Oriented Protocols

- Both are provided at the transport layer.
- Not related to physical paths of packets.
- Connectionless service at the transport layer means **independency** between packets; connection-oriented means **dependency**.
- **Connectionless Service:**
    - The source process (application program) needs to divide its message into chunks of data of the size acceptable by the transport layer and deliver them to the transport layer one by one.
    - The transport layer treats each chunk as independent unit, encapsulate it and send to the receiver **without any numbering**.
    - The receiver may get these packets **out of order**, decapsulates them and deliver the packets in that order only.
    - No flow control, error control, or congestion control can be effectively implemented in a connectionless service.

Connectionless and Connection-Oriented Protocols

- **Connectionless Service:**

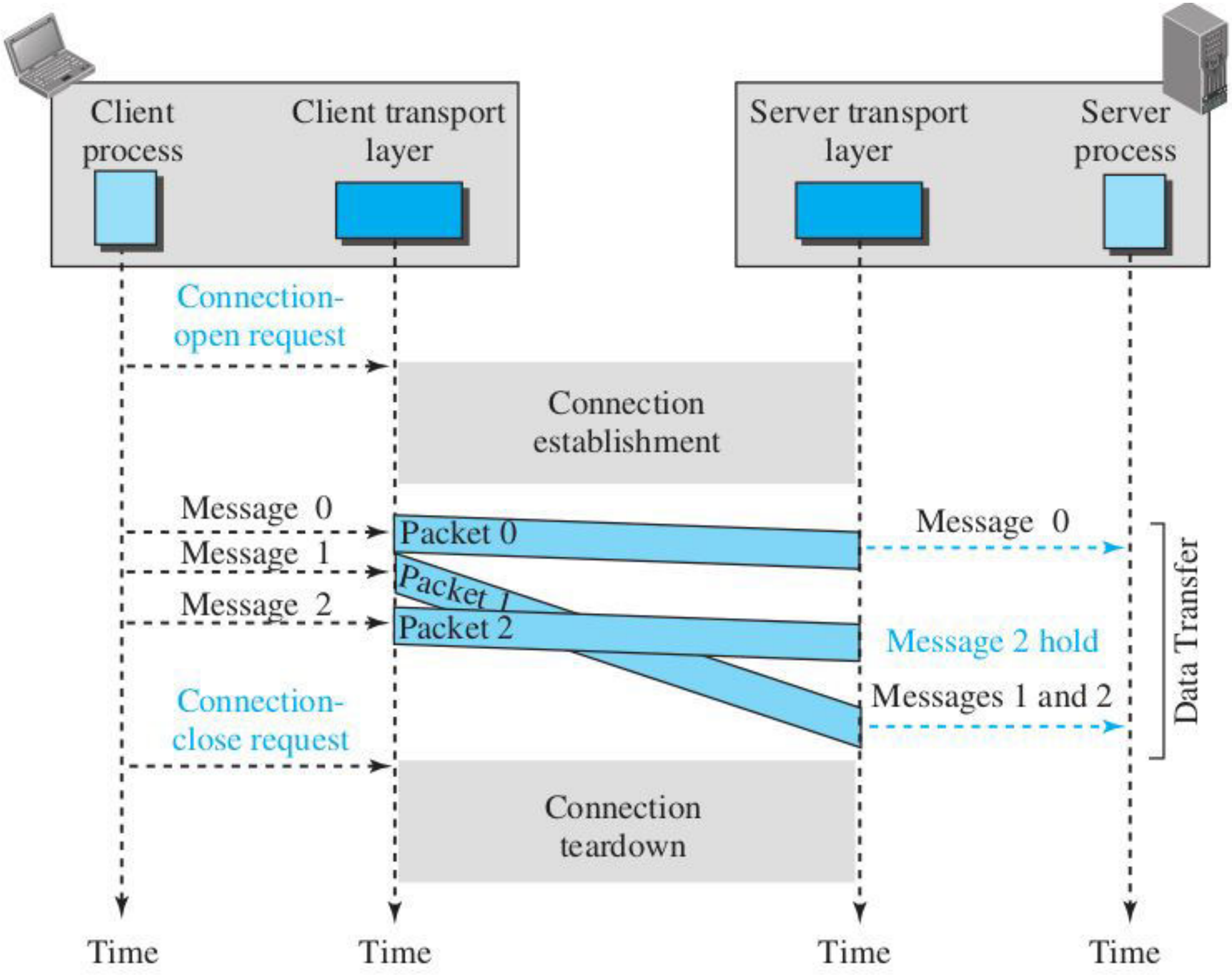


Connectionless and Connection-Oriented Protocols

- **Connection-Oriented Service:**
  - The client and the server first need to **establish a logical connection** between themselves. Then **data exchange** happens. After this, the connection needs to be **torn down**.
  - The connection-oriented protocol at the transport layer happens over either a connectionless or connection-oriented protocol at the network layer.
  - Flow control, error control, and congestion control are implemented in a connection-oriented protocol.

Connectionless and Connection-Oriented Protocols

- **Connection-Oriented Service:**

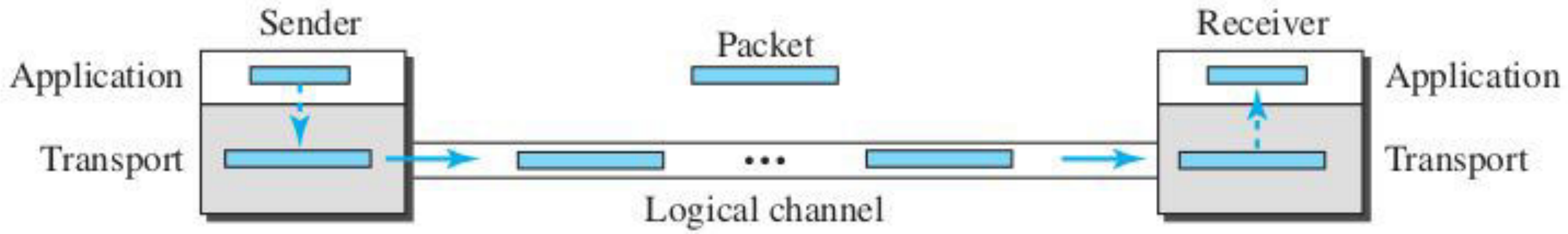


3.1.2 TRANSPORT-LAYER PROTOCOLS

- Flow control and error control are implemented by different protocols in the transport layer.
- The below list represents such protocols in unidirectional style.
  - Simple Protocol
  - Stop-and-Wait Protocol
  - Go -Back-N (GBN) Protocol
  - Selective-Repeat Protocol
  - Bidirectional Protocol: Piggybacking

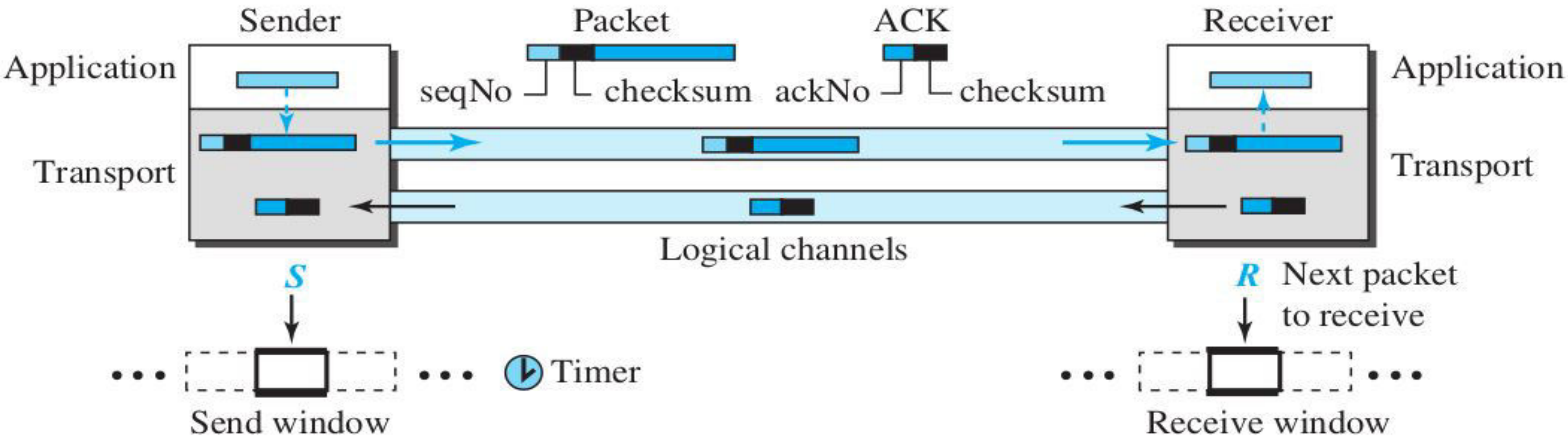
Simple Protocol

- Connectionless.
- No flow control, no error control.
- The sender sends packets one after another without even thinking about the receiver.



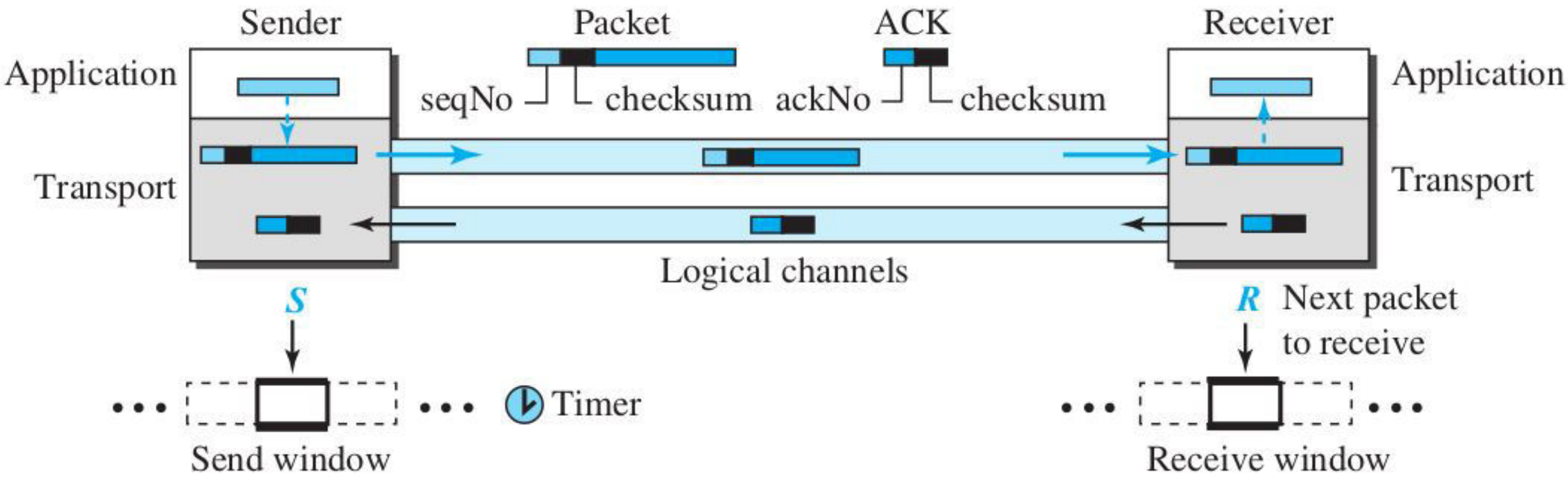
Stop-and-Wait Protocol

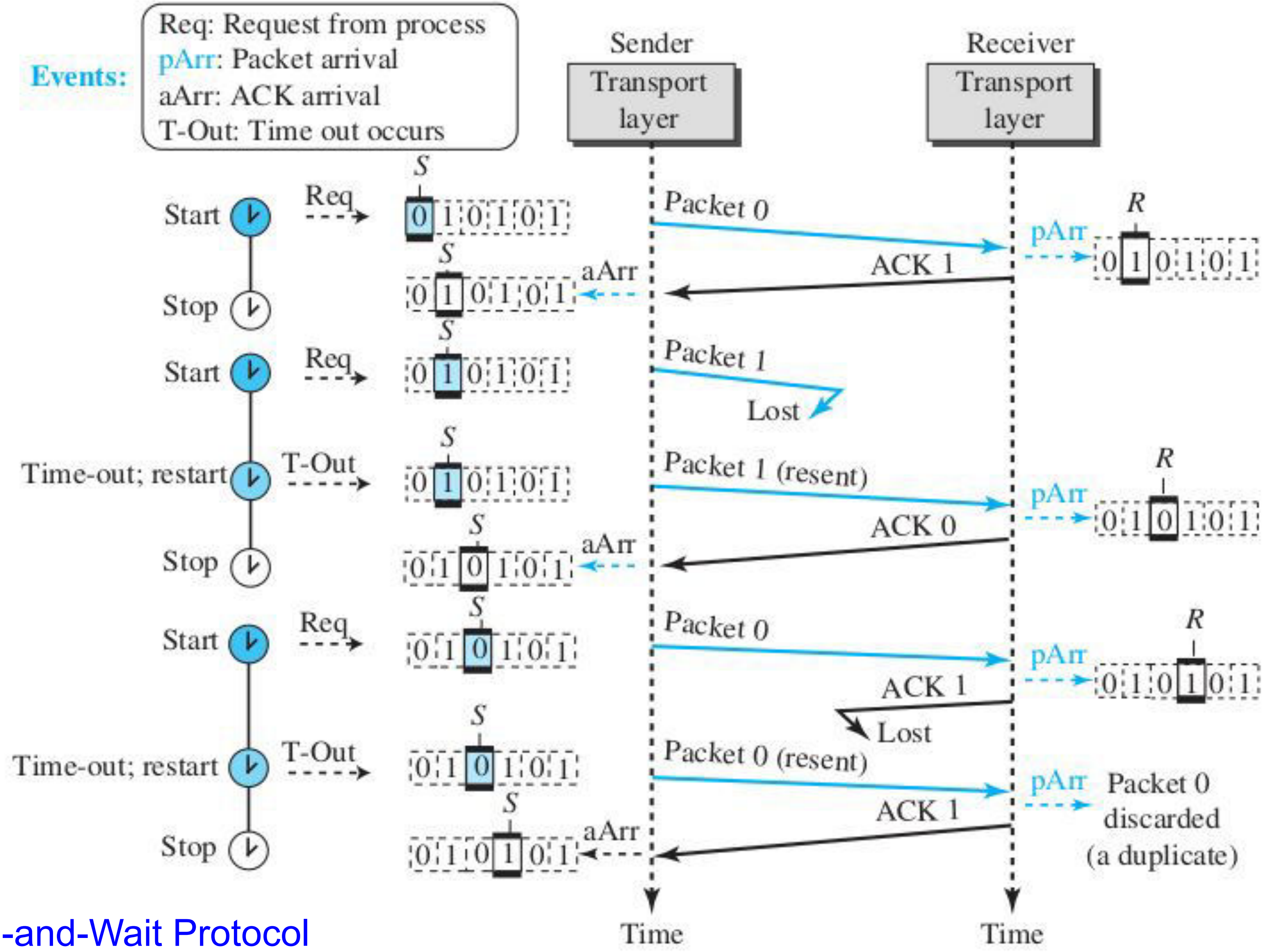
- Connection-oriented protocol with flow control and error control.
- Use sequence number to assure the packet order and acknowledgement to assure the error-free packet delivery.
- Use **checksum** in both data and acknowledgement to detect error.
- The Ack number represents the **next sequence number expected**.
- A **timer** is used at the sender to limit the time for waiting for Ack.
- Working: The sender sends a packet and waits for the Ack. If the Ack is received before the timer expires, the next packet can be processed and sent. If the timer expires, the same packet is resent.



Stop-and-Wait Protocol

- Both the sender and the receiver use a sliding window of **size 1**.
- Only 1-bit sequence number (ie, modulo-2). The numbers are **0, 1, 0, 1**, ...
- The sender has a control variable S (sender), that points to the only slot in the send window.
- The receiver has a control variable R (receiver), that points to the only slot in the receive window.





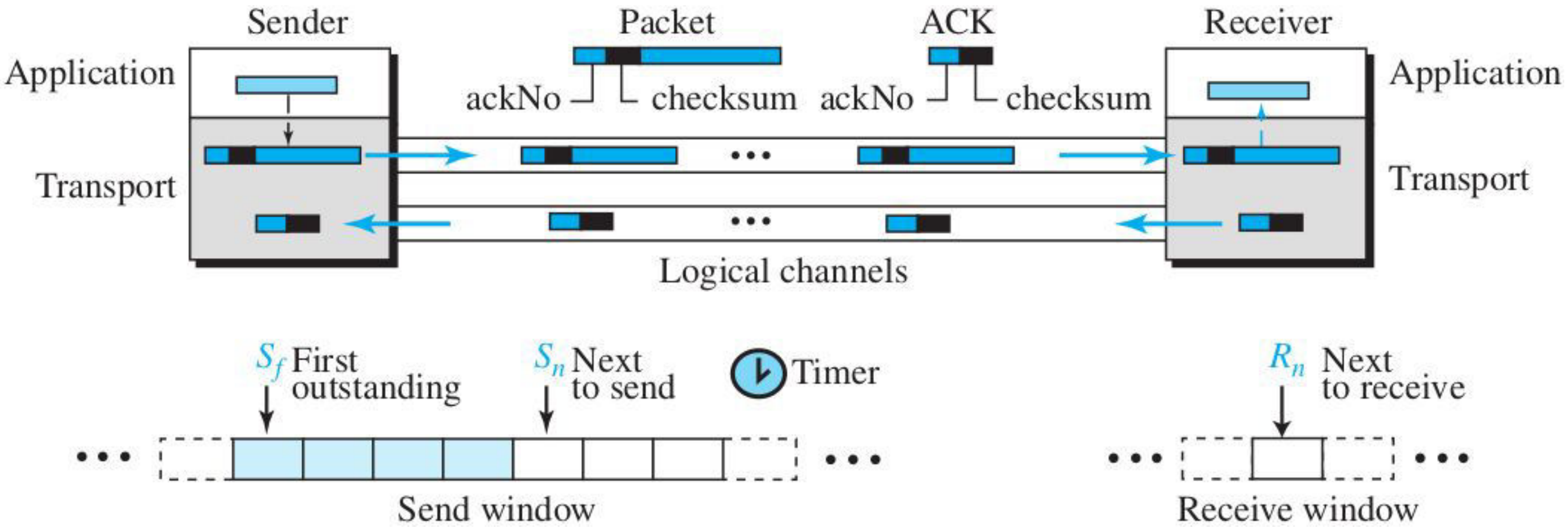
Events:

Req: Request from process
pArr: Packet arrival
aArr: ACK arrival
T-Out: Time out occurs

Stop-and-Wait Protocol

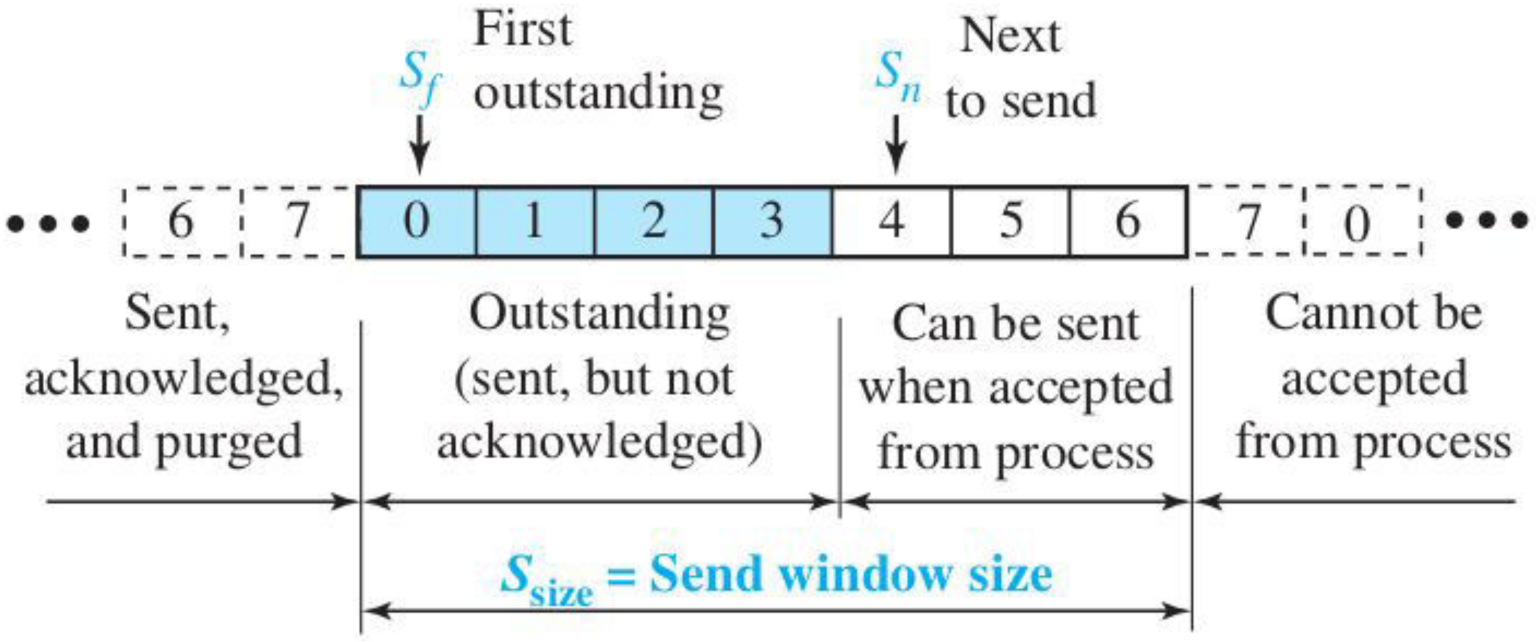
Go-Back-N Protocol (GBN)

- To improve efficiency of Stop-and-Wait protocol, **multiple packets can be sent while the sender is waiting for Ack.**
- Several data packets and acknowledgments can be in the channel at the same time.
- The receiver can **only buffer one packet**.
- Sequence numbers: modulo 2^m (eg: if m=3, sequence nos. 0,1,2,..,7,0,1,2,..)
- Ack numbers: **Cumulative** and defines the sequence number of the **next packet expected.**



Go-Back-N Protocol (GBN)

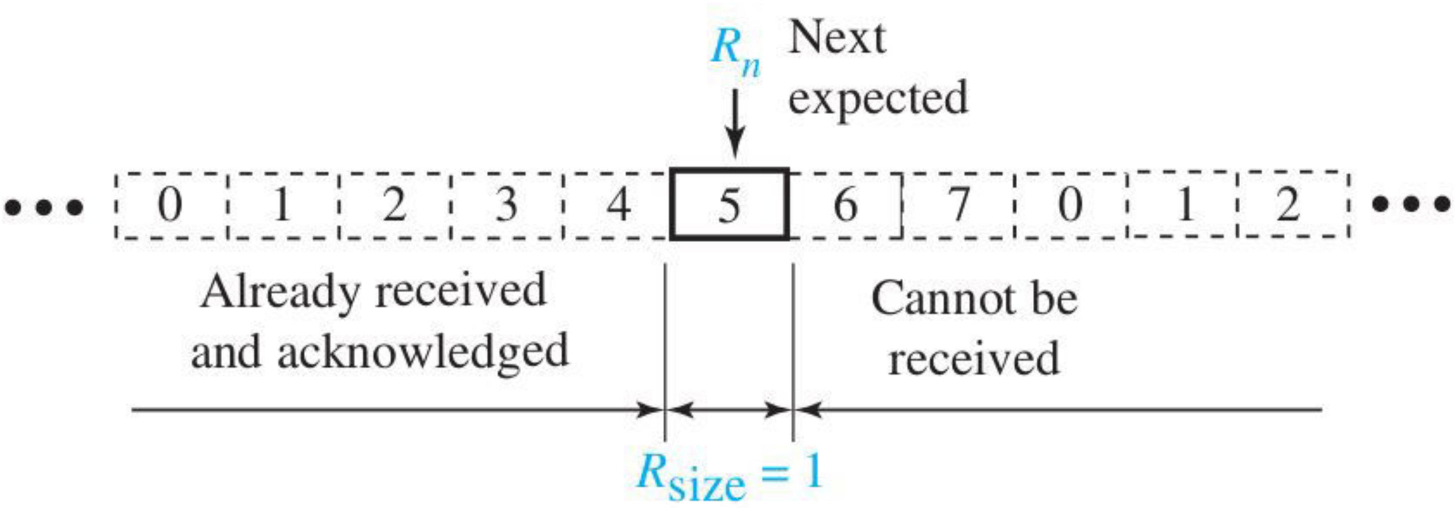
- Send Window: An imaginary box covering the sequence numbers of the data packets that **can be in transit** or **can be sent**.
  - maximum size 2^m-1 (ie, for modulo 8, the window size can be upto 7).
  - The send window at any time divides the possible sequence numbers into four regions; as in the figure.



S_f First outstanding
S_n Next to send
6
7
0
1
2
3
4
5
6
7
0
Sent, acknowledged, and purged
Outstanding (sent, but not acknowledged)
Can be sent when accepted from process
Cannot be accepted from process
S_{size} = Send window size

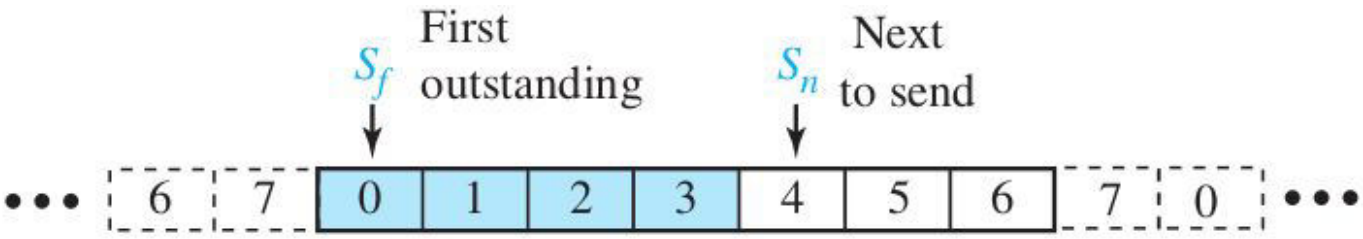
Go-Back-N Protocol (GBN)

- Receive Window:
  - Size 1
  - One control variable, R_n pointing to the next packet expected.
  - Accepts only the specific packet with the sequence number R_n. Any packet arriving other than this is discarded and needs to be resent.
  - When a correct packet is received, the window slides, R_n = (R_n + 1) modulo 2^m.



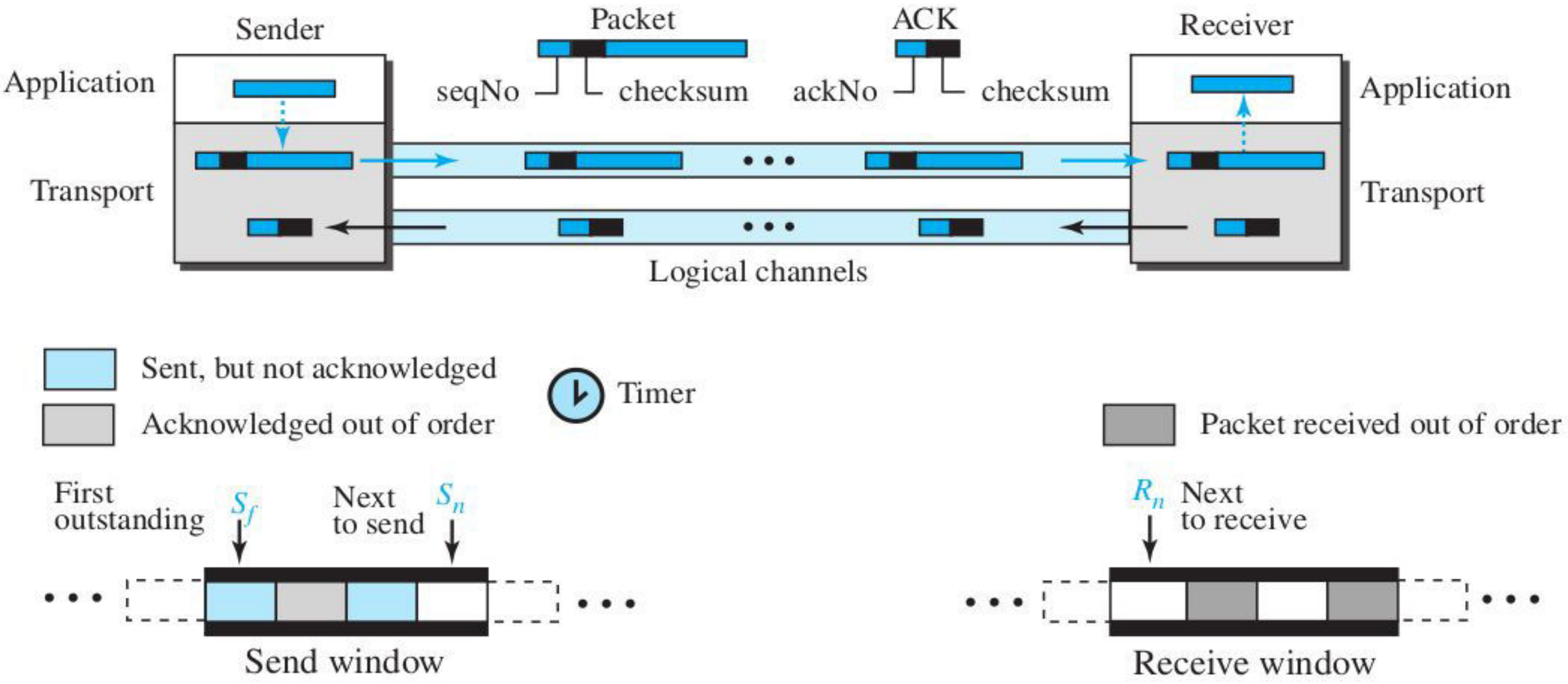
Go-Back-N Protocol (GBN)

- Timers: Keeps only one timer practically.
- Resending packets:
  - When the timer expires, the sender resends all outstanding packets.
  - In the below figure, let the packet from the window 0 to 3 have been sent and the timer starts at slot 0. If an Ack for 0 to 3 has not come within the timer expiry, the **window will resent all the packets from 0 again.** That is why this protocol is called Go-Back-N.



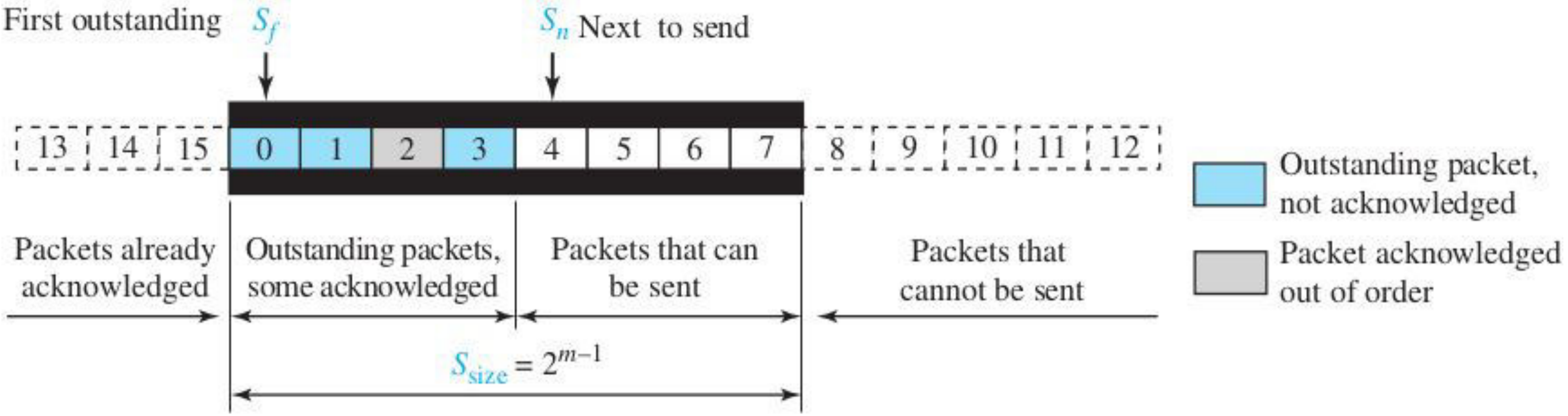
Selective-Repeat (SR) Protocol

- To improve GBN protocol.
- Resends **only selective packets, those that are actually lost**.
- **Send and Receive window will have the same size** and can be **maximum 2^{m-1}** for a modulo 2^m sequence number (eg: m=3, seq. nos. 0-7, window size 4).



Selective-Repeat (SR) Protocol

Send Window (for m=4):



First outstanding S_f

S_n Next to send

Packets already acknowledged

Outstanding packets, some acknowledged

Packets that can be sent

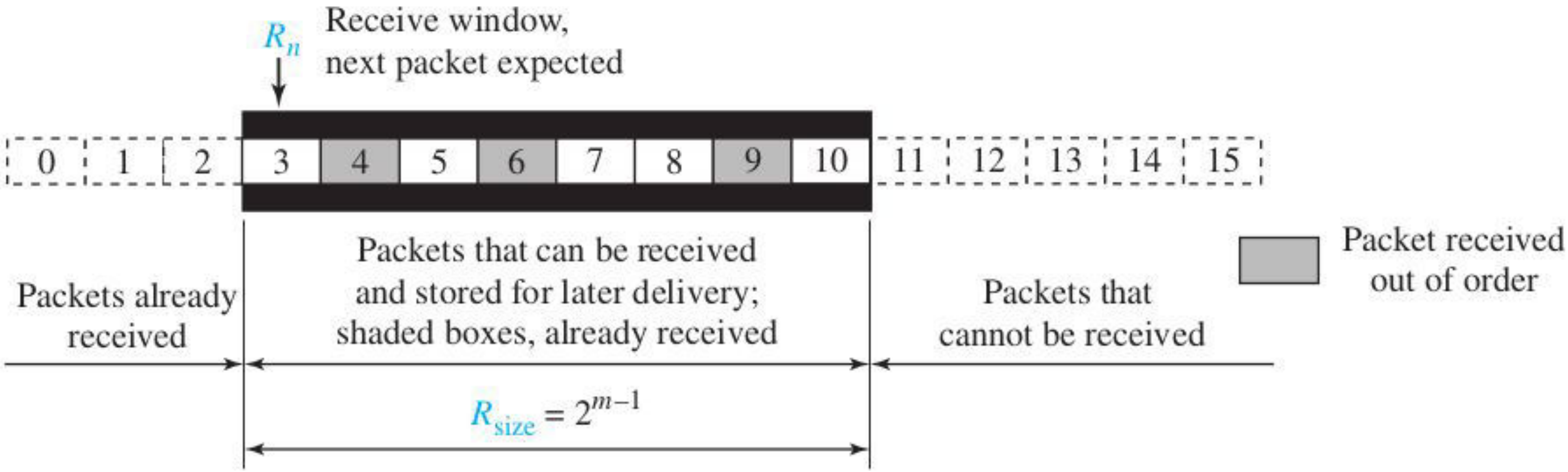
Packets that cannot be sent

$S_{size} = 2^{m-1}$

Outstanding packet, not acknowledged

Packet acknowledged out of order

Receive Window:



R_n Receive window, next packet expected

Packets already received

Packets that can be received and stored for later delivery; shaded boxes, already received

Packets that cannot be received

$R_{size} = 2^{m-1}$

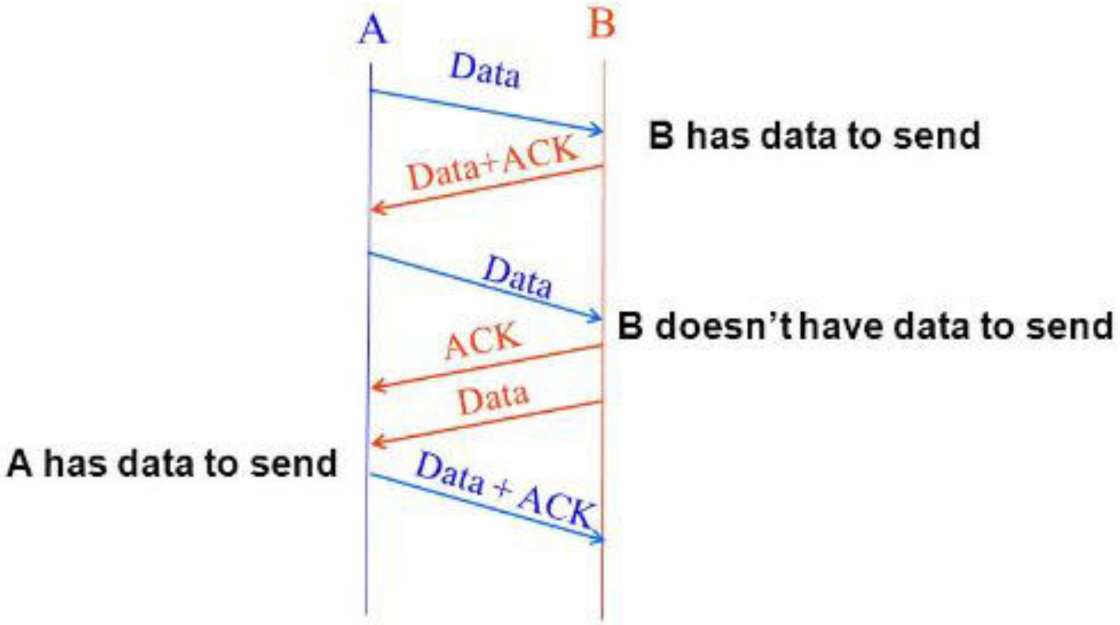
Packet received out of order

Selective-Repeat (SR) Protocol

- Timer:
  - Theoretically, Selective-Repeat uses one timer for each outstanding packet.
  - Most transport-layer protocols that implement SR use only a single timer.
- Acknowledgement:
  - An ackNo defines the sequence number of a single packet **that is received safe and sound**, not the next packet expected.

Bidirectional Protocols: Piggybacking

- In real communication, data packets are normally flowing in both directions.
- The GBN, SR and other protocols can be implemented in bidirectional form.
- Piggybacking is used to improve the efficiency of the bidirectional protocols.
- When a packet is carrying data from A to B, it can also carry Ack feedback of A about arrived packets from B, and vice versa.



End of Part 1 of Module 3

Thank You

Application Layer

- Provides services to the user.
- Communication is provided using a logical connection between the source and destination application layers.

Services

- Highest layer in the protocol suite.
- Do not provide service to any layers, but receive service from transport layer.
- Hence new protocols can be easily added.
- All protocols in the lower four layers are standardized, but application layer protocols can be standardized or non-standarized.
- Standarized:
    - can be used anywhere in the internet.
    - provides a standard set of services.
- Non-standardized:
    - created for proprietary use in some offices for private use.
    - use the standard services provided by transport layer.

Application-Layer Paradigms

- Two types paradigms: client-sever (traditional) and peer-to-peer.
- Client-Server
  - An application program, called the server process, runs continuously, waiting for another application program, called the client process, to make a connection through the Internet and ask for service.
  - Server processes do some specific services; but clients can ask for a number of services to different servers.
  - Many clients can ask for a service to a single server. Hence the server will be loaded heavily and thus need a powerful computer to work.
  - Mostly the service return some type of income for the server in order to encourage such an arrangement.
  - Eg: WWW, e-mail
- Peer-to-Peer (P2P)
  - No server process running all the time and waiting for the client processes to connect.
  - One device provide service to the other, and vice versa, and both can be at the same time. Ie, responsibility is shared between peers.
  - Eg: BitTorrent, IPTV, Internet Telephony, Skype
- Sometimes these two paradigms can be mixed.

World Wide Web (WWW or Web)

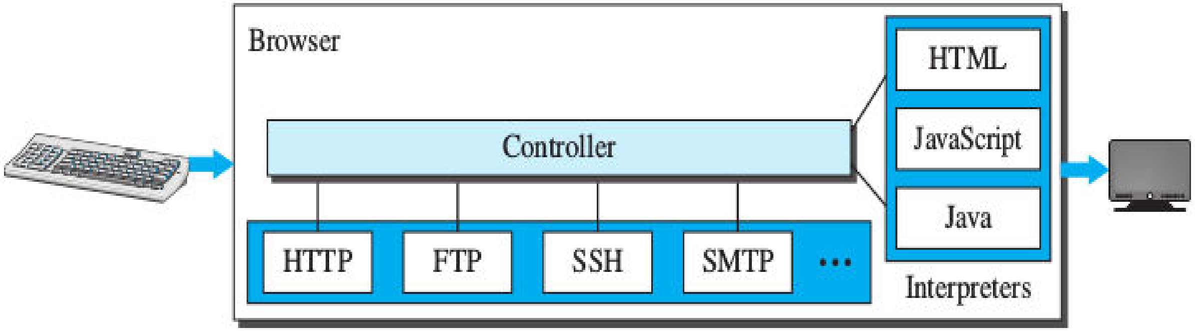
- First proposed by Tim Berners-Lee.
- Repository of information in which the documents, called web pages, are distributed all over the world and related documents are linked together.
- Linking of web pages was achieved using a concept called hypertext, now hypermedia.
- Now interactive and streaming data are available such as gaming, radios, etc.

Architecture

- Distributed client-server service, in which a client using a browser can access a service using a server.
- Service provided is distributed over many locations called sites.
- Each site holds one or more web pages. Each web page can contain some links to other web pages in the same or other sites.
- Thus a web page can be simple (no links) or composite (contains atleast a link).
- Each web page is a file with a name and address.
- Accessing a web page is done by a web client from a web server.

Web Client (Browser)

- Consists of three parts: a controller, client protocols, and interpreters.
- Controller receives input from the keyboard or the mouse and uses the client programs (protocols) to access the document.
- Controller uses one of the interpreters to display the document on the screen.
- Eg: Firefox, Chrome, Opera



Web Server

- The web page is stored at the server.
- Each time a request arrives, the corresponding document is sent to the client.
- Efficiency can be improved by caching, multithreading or multiprocessing.
- Eg: Apache and Microsoft Internet Information Server.

Uniform Resource Locator (URL)

- A web page must be identified by a unique identifier.
- For this, we use the Protocol to be used by the browser, Host id of the server, Port to be accessed and the actual Path of the file to be accessed.
- Protocol: http, ftp
- Host: IP address or domain name (such as *example.com*)
- Port: A 16-bit integer. Normally pre-defined for client-server appication. Eg. HTTP port number is 80.
- Path: The location and the name of the file in the underlying operating system.
- A Uniform Resource Locator (URL) is a combination of all these.
- The common form is given below.

protocol://host/path Used most of the time

Eg: http://www.tekerala.org/staff_login.php

protocol://host:port/path Used when port number is needed

Eg: http://www.example.com:8080/samples/index.html

Web Documents

- Classified into three: static, dynamic and active.

- Static Documents:
  - Fixed-content documents that are created and stored in a server.
  - Client can get a copy only.
  - The document content in the server can be changed, but the user cannot change them.
  - The user can see the document copy in the browser.
  - Some languages used for static document creation are: HyperText Markup Language (HTML), Extensible Markup Language (XML), Extensible Style Language (XSL), and Extensible Hypertext Markup Language (XHTML).

- Dynamic Documents:
  - Created by a web server whenever a browser requests the document.
  - Whenever a request arrives from the client, the web server runs an application program or a script that freshly creates the dynamic document every time.
  - Some languages for this are Java Server Pages (JSP), Active Server Pages (ASP), etc.

- **Active Documents**
  - Documents containing scripts that works at the client side.
  - Such as a program that creates animated graphics on the screen or a program that interacts with the user.
  - When a browser requests an active document, the server sends a copy of the document or a script.
  - This can be done using Java applet, javascript, etc.

HyperText Transfer Protocol (HTTP)

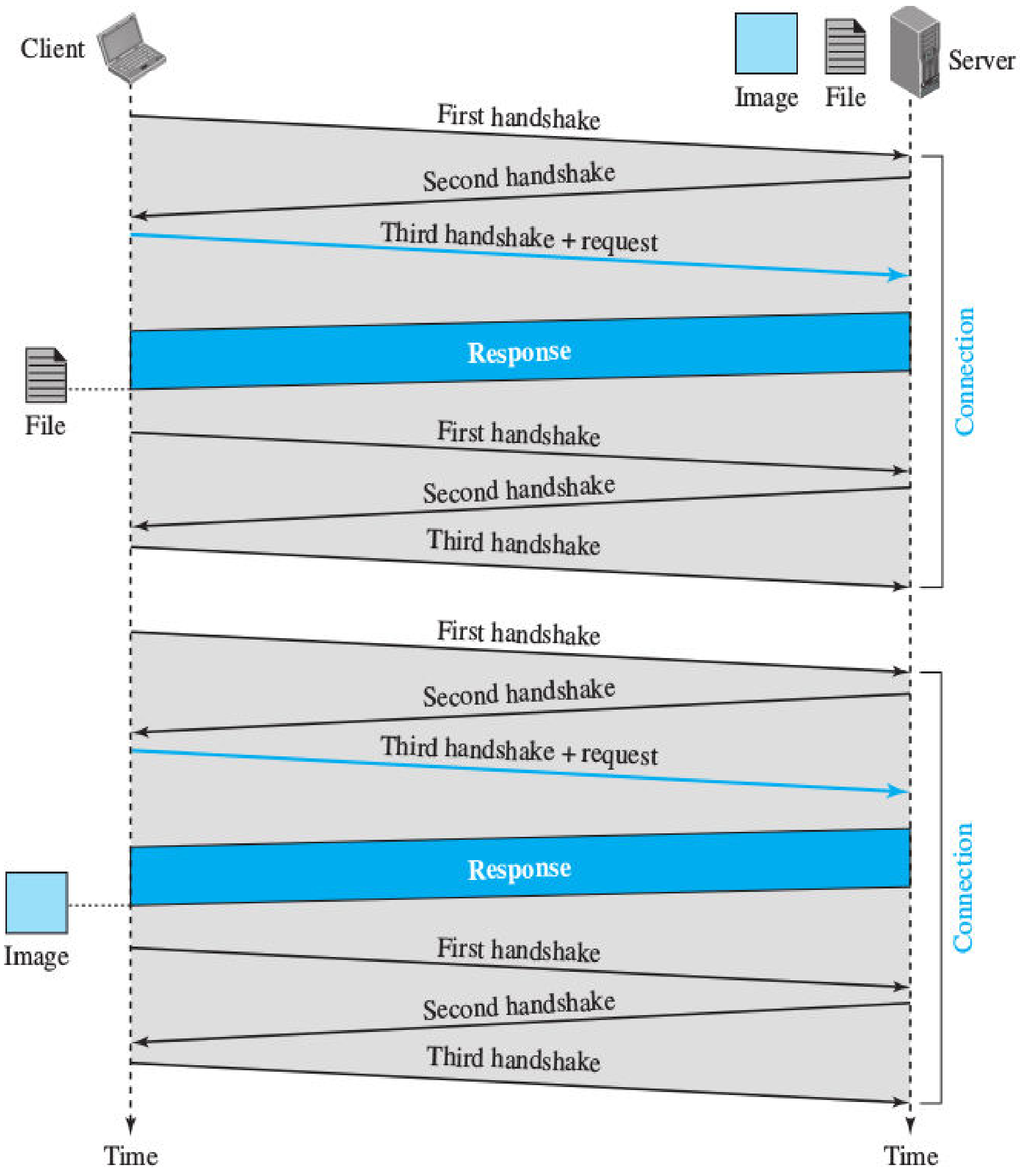
- Used to define how the client-server programs can be written to retrieve web pages from the Web.
- An HTTP client sends a request; an HTTP server returns a response.
- The server uses the port number 80; the client uses a temporary port number.
- Uses the services of TCP (ie, connection oriented and reliable).
- To display a web document on the browser, there may be a need for several objects, that may reside in a number of servers. Each one is accessed using TCP connections (ie, connection establishment, data transfer, connection termination).
- Some of these objects may be on the same server. Here the client can *use a new TCP connection for each object* or *make a TCP connection and retrieve them all*.
- Nonpersistent Connection:
  - One TCP connection is made for each request/response.

    1. The client opens a TCP connection and sends a request.
    2. The server sends the response and closes the connection.
    3. The client reads the data until it encounters an end-of-file marker; it then closes the connection.
  - If a file contains links to N different pictures in different files (all located on the same server), the connection must be opened and closed N + 1 times.
  - Very high overhead.

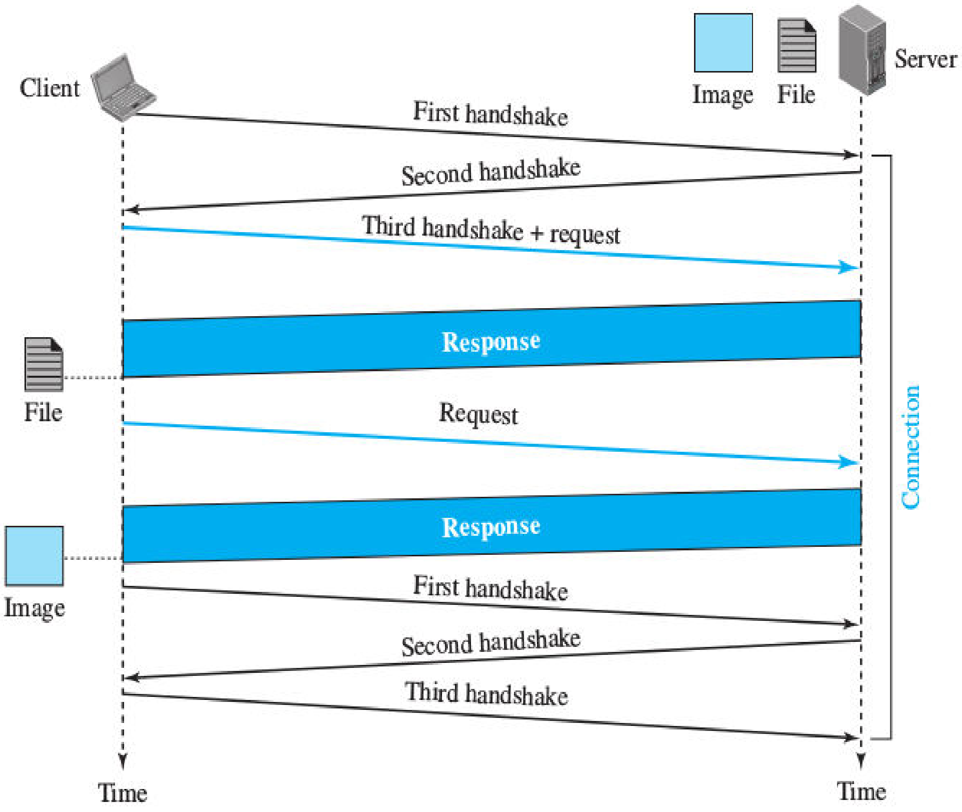
HyperText Transfer Protocol (HTTP)...

- Persistent Connections
  - Present HTTP versions have persistent connections by default.
  - Here the server opens a connection, send the response and remains open for more requests.
  - The server can close the connection at the request of a client or if a time-out has been reached.
  - For static documents, the sender sends the length of the data with each response.
  - For active and dynamic documents, the server informs the client that the length is not known and closes the connection after all the data are sent.
  - Less time and resource usage.
  - Only one set of buffers and variables are neede for the whole transfer.
  - Less round trip time.

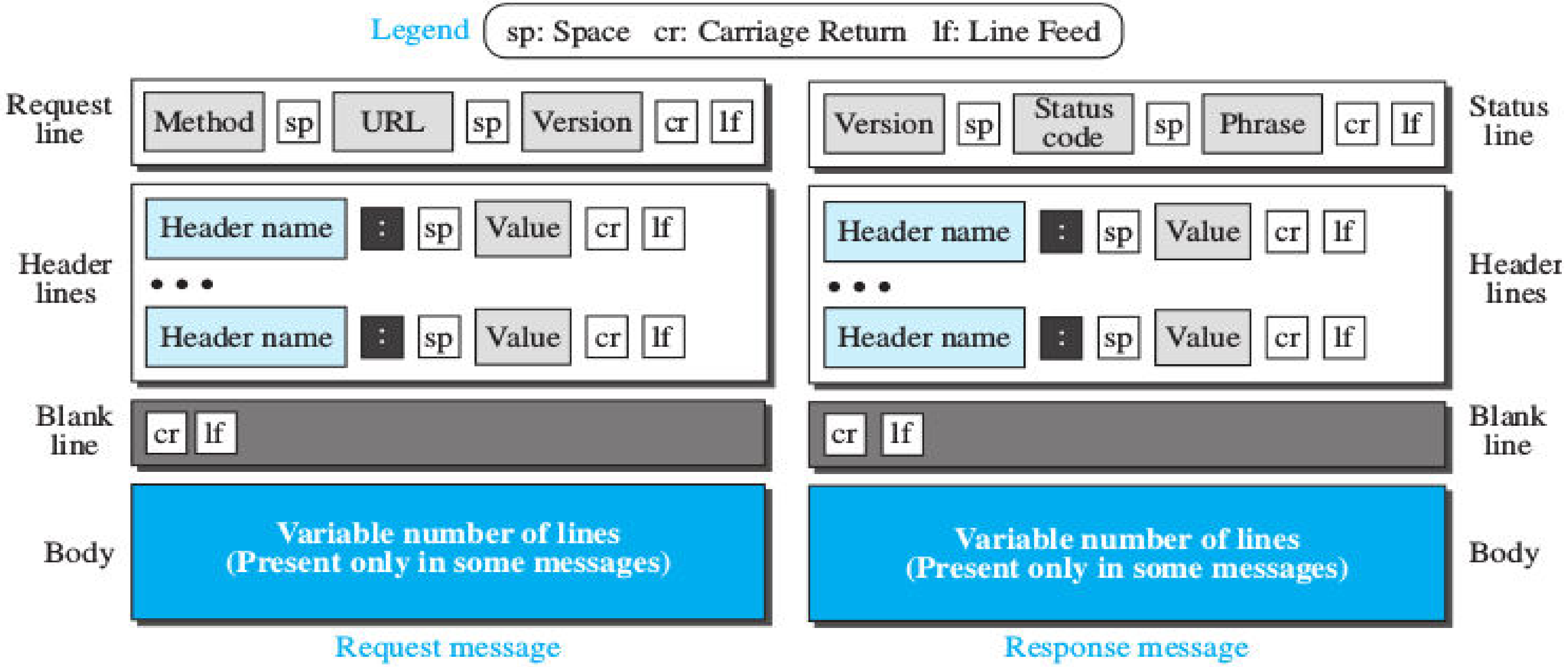
Non-persistent Connection



Persistent Connection



- **HTTP Message Formats**



```
GET /docs/index.html HTTP/1.1
Host: www.nowhere123.com
Accept: image/gif, image/jpeg, */*
Accept-Language: en-us
Accept-Encoding: gzip, deflate
User-Agent: Mozilla/4.0
(blank line)
```

```
HTTP/1.1 200 OK
Date: Sun, 18 Oct 2009 08:56:53 GMT
Server: Apache/2.2.14 (Win32)
Last-Modified: Sat, 20 Nov 2004 07:16:26 GMT
ETag: "10000000565a5-2c-3e94b66c2e680"
Accept-Ranges: bytes
Content-Length: 44
Connection: close
Content-Type: text/html
X-Pad: avoid browser bug

<html><body><h1>It works!</h1></body></html>
```

Request Methods

- GET
  - To send a request to the server.
  - Most common.
  - No body.
- HEAD
  - When the client need some information about the web page from the server, such as page modified date.
  - Used to to test the validity of the URL.
  - No body.
- PUT
  - Inverse of GET.
  - Put some information in to the server (if permitted).
- POST
  - To send some information to the server or to modify the page.
- TRACE
  - An *echo* service for debugging.
- DELETE
  - To delete a web page from the server (if permitted).
- OPTIONS
  - Inquires about available options and properties.

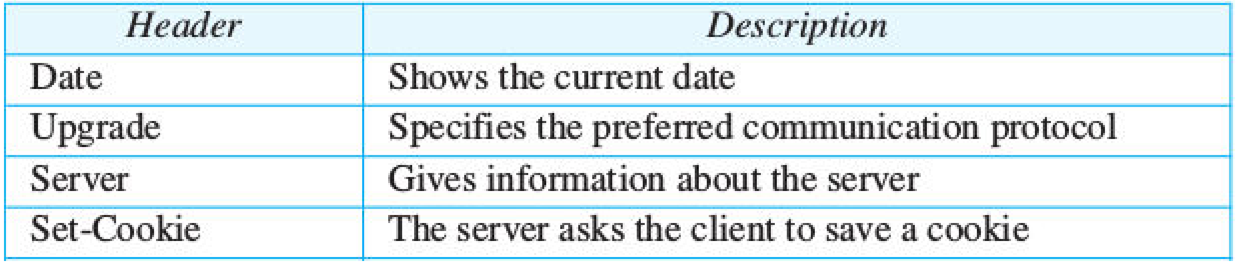
Request Header field

- It contains one or more headers.
- They are additional information from the client.
- Eg: the client can request that the document be sent in a special format.

Response Message

- Contains version, status code, phrase, headers and the body of the response.
- Version: http version such as 1.0, 1.1, etc.
- Some common status codes are;
  - 200 – OK: The request succeeded.
  - 204 - No Content: The document contains no data.
  - 301 - Moved Permanently: The resource has permanently moved to a different URI.
  - 401 - Not Authorized: The request needs user authentication.
  - 403 – Forbidden: The server has refused to fulfill the request.
  - 404 - Not Found: The requested resource does not exist on the server.
  - 408 - Request Timeout: The client failed to send a request in the time allowed by the server.
  - 500 - Server Error: Due to a malfunctioning script, server configuration error or similar.

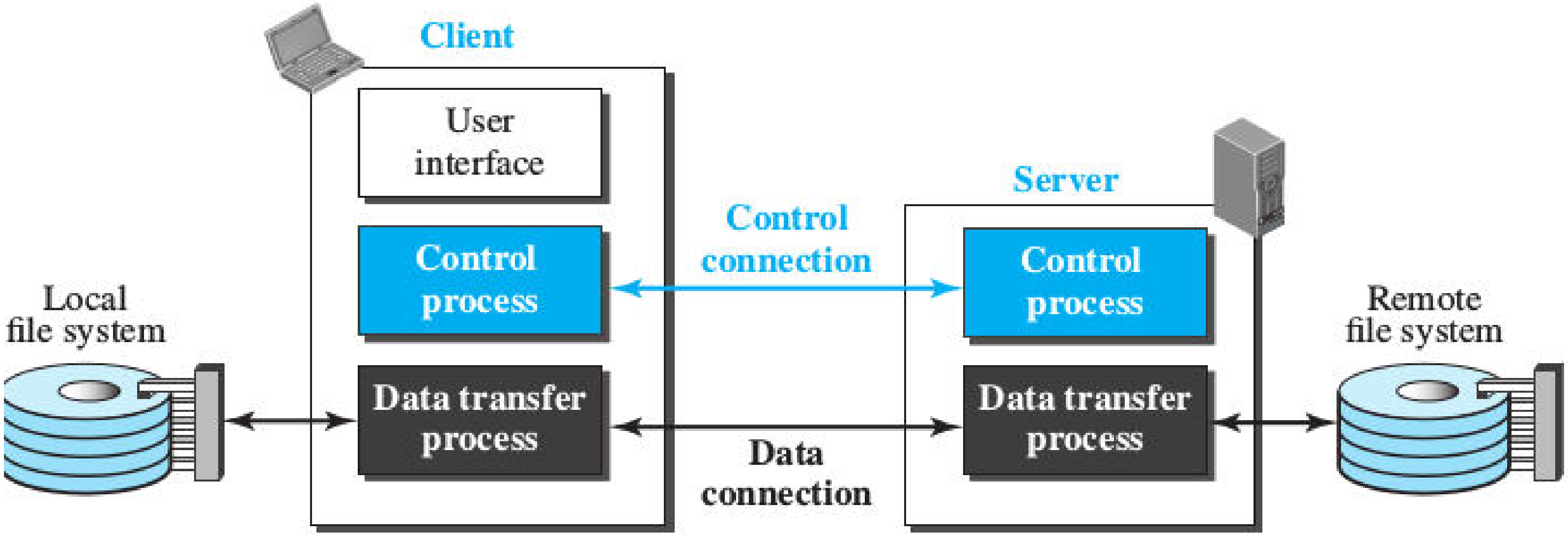
- **Response Message Headers**
- Some header values are;



Header	*Description*
Date	Shows the current date
Upgrade	Specifies the preferred communication protocol
Server	Gives information about the server
Set-Cookie	The server asks the client to save a cookie

File Transfer Protocol (FTP)

- Used to transfer files from one host to another.
- File transfer means retrieving a file (server to client), storing a file (client to server), or directory listing (server to client).
- Uses TCP/IP.
- Transfers files eventhough both hosts have different directory structures, naming convensions, data representation.
- The basic model for FTP is;



- The client has three components: the user interface, the client control process, and the client data transfer process. The server has two components: the server control process and the server data transfer process.
- The control connection is made between the control processes. The data connection is made between the data transfer processes.

- When a user starts an FTP session, the control connection opens.
- Control connection remains connected during the entire interactive FTP session. Data connection is opened and then closed for each file transfer activity.
- While the control connection is open, the data connection can be opened and closed multiple times if several files are transferred.
- TCP port 21 is used for the control connection, and port 20 is used for the data connection.

Control Connection

- Communication through commands and responses as ASCII characters.
- Each line is terminated with a two-character (carriage return and line feed) end-of-line token.
- Commands are sent from client to server as uppercase letters and responses from server to client as digits and letters.
- Some commands are:
- ABOR (Abort the previous command), DELE *filename* (delete a file), QUIT (log out)
- Every FTP command generates one or more responses.
- Response format: <*a three digit code*> <*description*>
- Eg: 125 Data connection open, 230 User login OK

Data Connection

- To transfer data.
- Server uses its port 20.
- The client defines the type of file to be transferred, the structure of the data, and the transmission mode through control connection before every data connection is made.
- The steps for data connection are;

1) The client issues a passive open request to the server from an ephemeral port.
2) Using the PORT command the client sends this port number to the server.
3) The server receives the port number and issues an active open using the well-known port 20 and the received ephemeral port number.

File Type

- FTP transfers the following file types: ASCII file, EBCDIC (Extended Binary Coded Decimal Interchange Code) file, or image file.

Data Structure

- FTP uses file structure, record structure, or page structure for file transfer.
- File structure is continuous stream of bytes.
- Record structure is used for text files by dividing the file into records.
- In the page structure, the file is divided into pages, with each page having a page number and a page header. The pages can be stored and accessed randomly or sequentially.

Transmission Mode

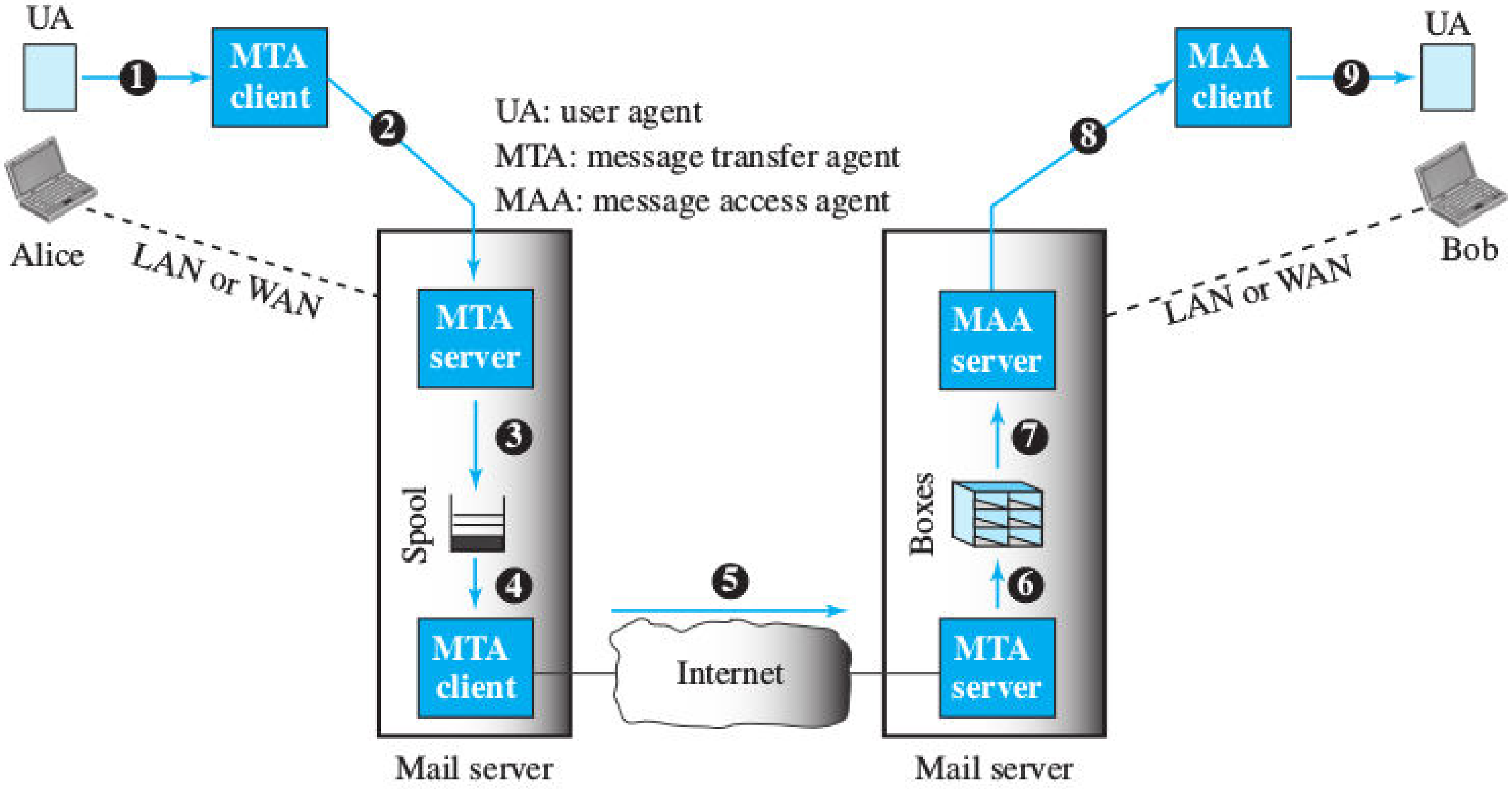
- FTP uses stream mode, block mode, or compressed mode.
- The stream mode is the default mode; data are delivered from FTP to TCP as a continuous stream of bytes.
- In the block mode, data can be delivered from FTP to TCP in blocks. In this case, each block is preceded by a 3-byte header.

ELECTRONIC MAIL

- E-mail is a one-way transaction.
- If A sends an e-mail to B, it is a one-way transaction. If B replies to A that is another one-way transaction.
- Intermediary servers always keep running to transfer the e-mails; the users are just clients.

Architecture

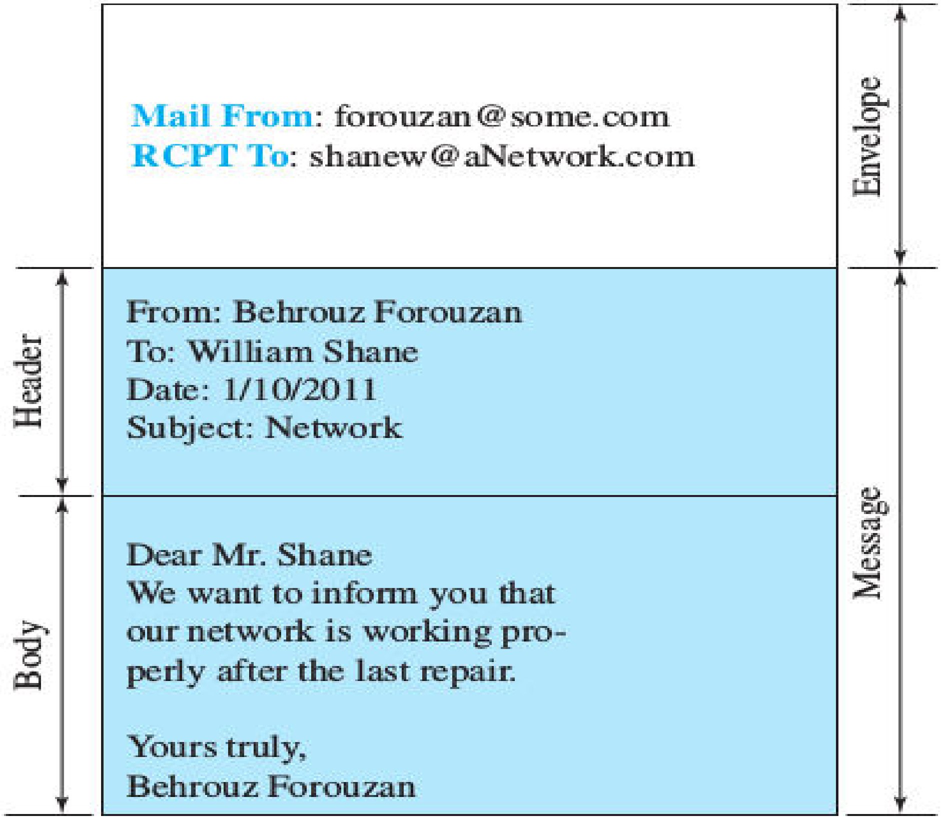
- A general scenario: The sender and the receiver of the e-mail, Alice and Bob respectively, are connected via a LAN or a WAN to two mail servers.
- The administrator has created one mailbox for each user where the received messages are stored. A mail-box is part of a server hard drive, a special file with permission restrictions. Only the owner of the mailbox has access to it.
- The administrator has also created a queue (spool) to store messages waiting to be sent.
- To send an e-mail, Alice and Bob use three different agents: a user agent (UA), a message transfer agent (MTA), and a message access agent (MAA).
- UA: program to prepare the message and send it to the sender's mail server. Receiver UA interprets the mail. CLI (Eg: elm, pine) or GUI (Eg: outlook)
- MTA: A push program to transfer the e-mail from one point to the other.
- MAA: A pull program to get the e-mail from one point to the other.



- The electronic mail system needs two UAs, two pairs of MTAs (2 client and 2 server), and a pair of MAAs (client and server).

Sending Mail

- An e-mail has an envelope and a message. The envelope usually contains the sender address, the receiver address, and other information. The message contains the header and the body. The header of the message defines the sender, the receiver, the subject of the message, and some other information. The body of the message contains the actual information to be read by the recipient.



Mail From: forouzan@some.com
RCPT To: shanew@aNetwork.com

From: Behrouz Forouzan
To: William Shane
Date: 1/10/2011
Subject: Network

Dear Mr. Shane
We want to inform you that our network is working properly after the last repair.

Yours truly,
Behrouz Forouzan

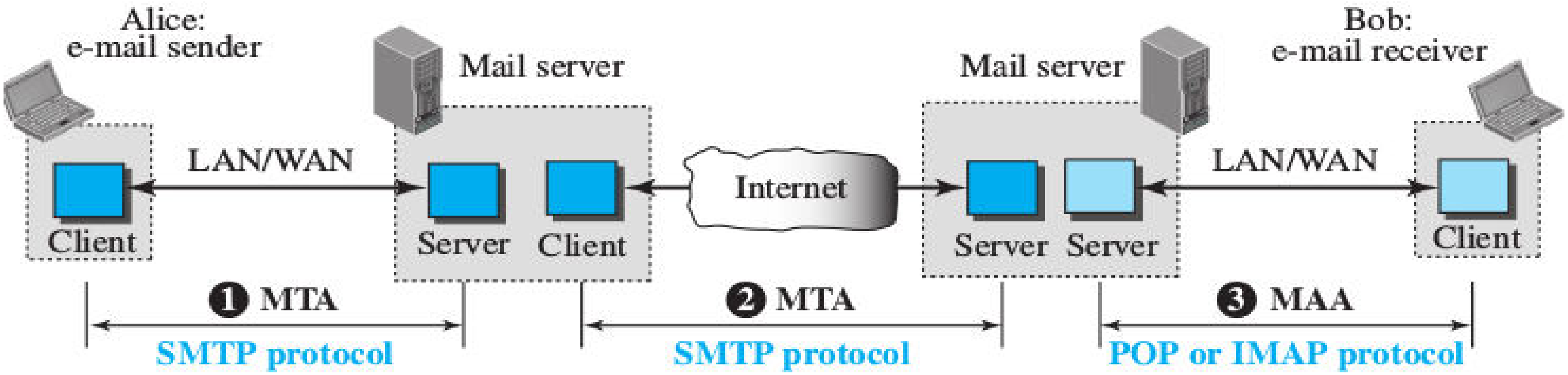
Electronic mail

Receiving Mail

- The user agent is triggered by the user (or a timer). If a user has mail, the UA informs the user with a notice.

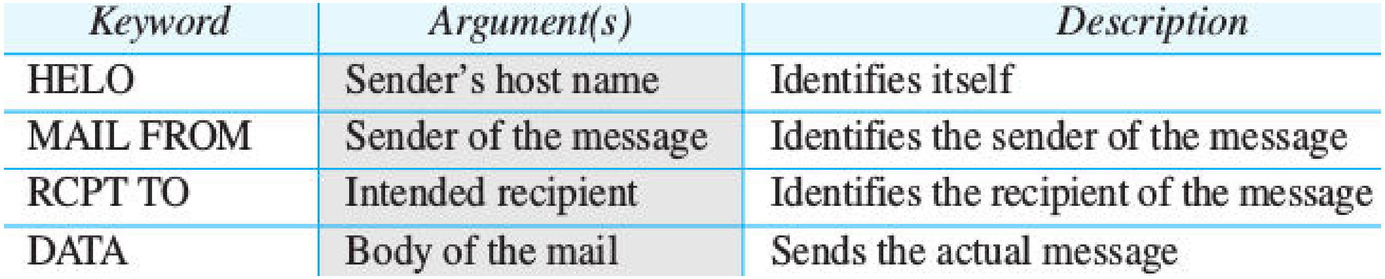
Addresses

- Consists of two parts: a local part (called *user mailbox*) and a domain name (called *mail servers* or *exchangers*), separated by an @ sign.



Message Transfer Agent: SMTP

- The formal protocol that defines the MTA client and server in the Internet.
- SMTP is used two times, between the sender and the sender’s mail server and between the two mail servers.
- Done by commands and responses.
- Command format: Keyword: argument(s)
    - Eg:



Keyword	*Argument(s)*	*Description*
HELO	Sender’s host name	Identifies itself
MAIL FROM	Sender of the message	Identifies the sender of the message
RCPT TO	Intended recipient	Identifies the recipient of the message
DATA	Body of the mail	Sends the actual message

- Responses: A three-digit code that may be followed by additional textual information. Sent from the server to the client.
    - Eg: 220 Service ready

      250 Request command completed

      450 Mailbox not available

Mail Transfer Phases

- Three phases:
  1) Connection establishment
  2) Mail transfer
  3) Connection termination.

1. Connection Establishment: Client TCP connect to well-known port 25 of the server and does the following three steps:

1) If the server is ready it sends a code 220 (service ready), otherwise 421 (service not available).
2) The client sends the HELO message to identify itself, using its domain name address.
3) The server responds with code 250 (request command completed) or some other code depending on the situation.

2. Message Transfer: Eight steps

1) The client sends the MAIL FROM message to introduce the sender of the message. It includes the mail address of the sender (mailbox and the domain name). This step is needed to give the server the return mail address for returning errors and reporting messages.

2) The server responds with code 250 or some other appropriate code.

3) The client sends the RCPT TO (recipient) message, which includes the mail address of the recipient.

4) The server responds with code 250 or some other appropriate code.

5) The client sends the DATA message to initialize the message transfer.

6) The server responds with code 354 (start mail input) or some other appropriate message.

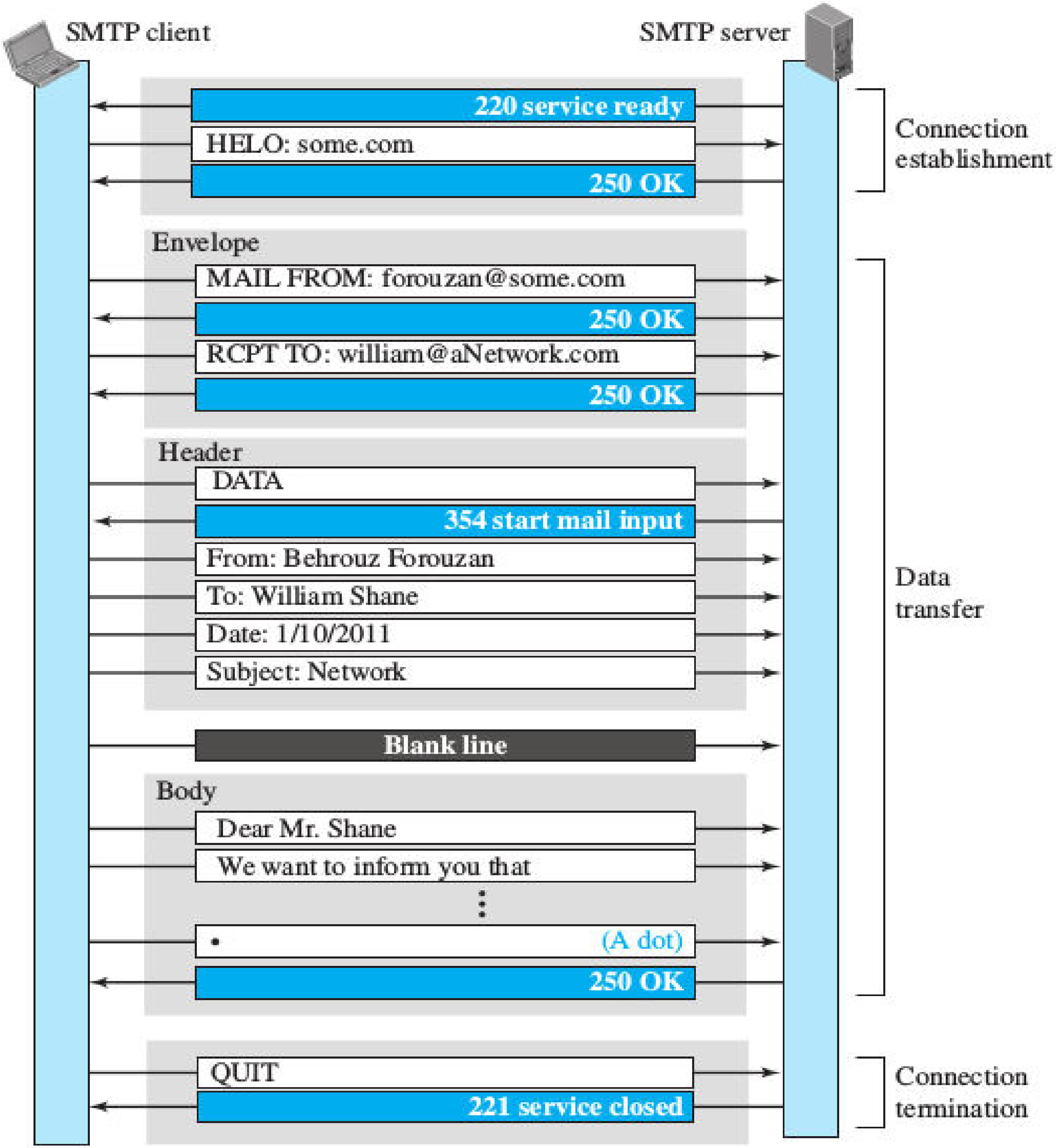
7) The client sends the contents of the message in consecutive lines. Each line is terminated by a two-character end-of-line token (carriage return and line feed). The message is terminated by a line containing just one period (.).

8) The server responds with code 250 (OK) or some other appropriate code.

Connection Termination: Two steps**:**

1) The client sends the QUIT command.

2) The server responds with code 221 or some other appropriate code.



SMTP client

SMTP server

220 service ready

HELO: some.com

250 OK

Connection establishment

Envelope

MAIL FROM: forouzan@some.com

250 OK

RCPT TO: william@aNetwork.com

250 OK

Header

DATA

354 start mail input

From: Behrouz Forouzan

To: William Shane

Date: 1/10/2011

Subject: Network

Blank line

Body

Dear Mr. Shane

We want to inform you that

⋮

• (A dot)

250 OK

Data transfer

QUIT

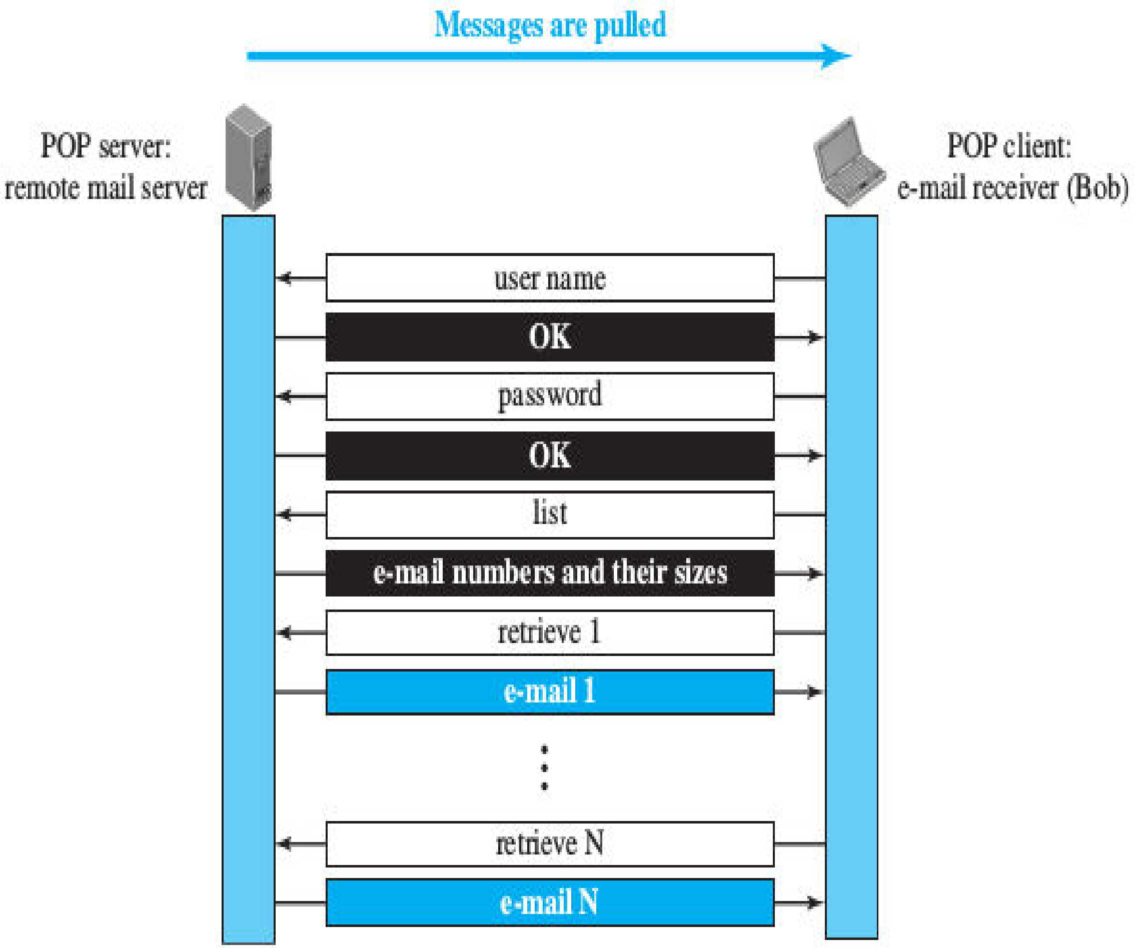
221 service closed

Connection termination

Message Access Agent: POP and IMAP

- To pull the message from server, the client uses a Message Access Agent (MAA)
- Currently used protocols are POP and IMAP

  - **Post Office Protocol, version 3 (POP3)**
  - Simple but limited in functionality.
  - To access a mail, the client opens a connection to the server on port 110.
  - Message access is shown:
  - POP3 has two modes: the **delete** mode and the **keep** mode. In the delete mode, the mail is deleted from the mailbox after each retrieval. In the keep mode, the mail remains in the mailbox after retrieval.
  - Does not allow to organize the mails on the server, to create folders or to partially check the contents before downloading.



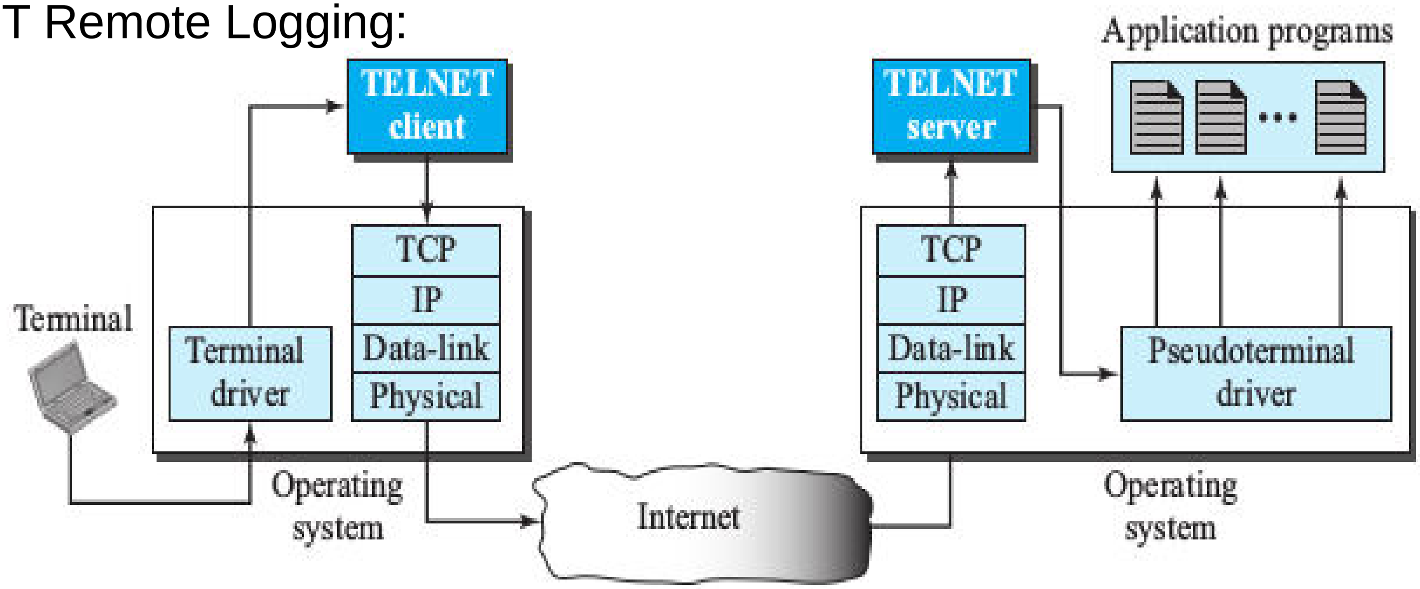
Internet Mail Access Protocol, version 4 (IMAP4)

- More complex and more features than POP3.
- IMAP provides the following additional features:
   1) A user can check the e-mail header prior to downloading.
   2) A user can search the contents of the e-mail for a specific string of characters prior to downloading.
   3) A user can partially download e-mail. This is especially useful if bandwidth is limited and the e-mail contains multimedia with high bandwidth requirements.
   4) A user can create, delete, or rename mailboxes on the mail server.
   5) A user can create a hierarchy of mailboxes in a folder for e-mail storage.

TELNET (TErminaL NETwork)

- A remote logging protocol.
- Remote Logging: A client can log into the computer at server site and use the services available there.
- TELNET requires a logging name and password, but vulnerable to hacking because it sends all data including the password in plaintext (not encrypted). Hence a secure protocol called Secure Shell (SSH) is more common.
- Used to learn the procudures and issues in remote logging.
- Used by network administrators for debugging purposes.
- To access TELNET, the client and server systems must have TELNET installed.

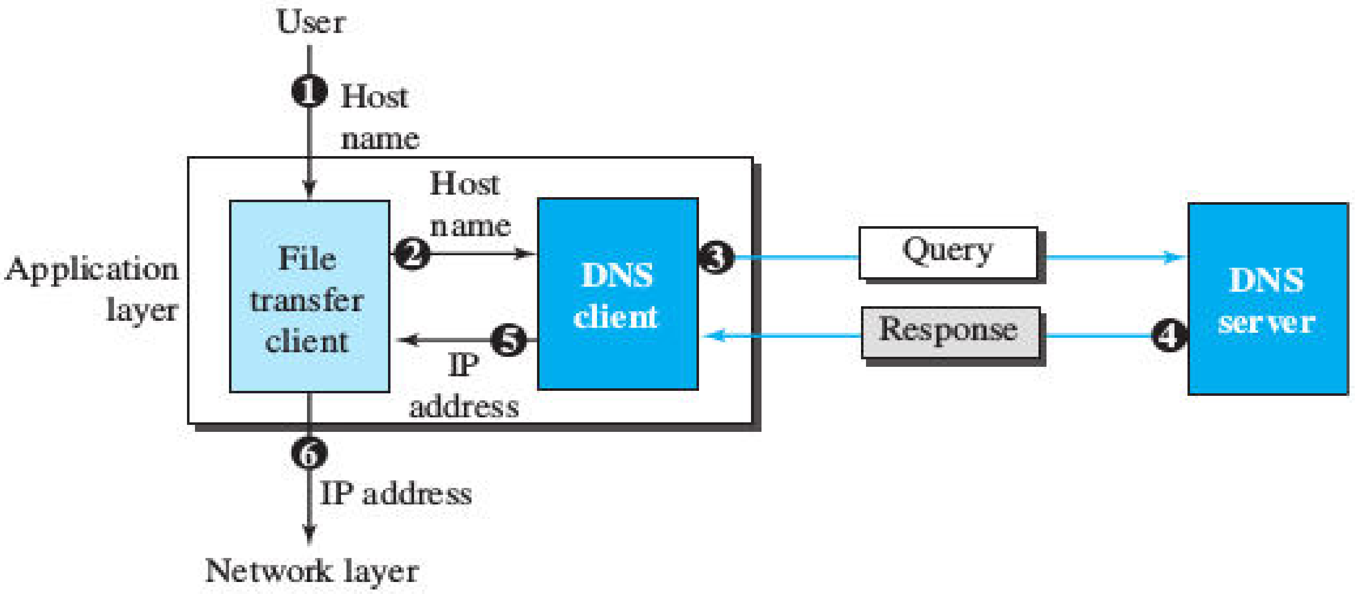
TELNET Remote Logging:



- The user keystrokes are accepted by a terminal driver and pass them to TELNET client.
- The TELNET client converts the text into a Universal Character set called Network Virtual Terminal (NVT) characters and delivers them to the local TCP/IP stack. (NVT is needed because the client and the server systems may have different working environment and operating system).
- At the server, the NVT text are accepted by the TCP/IP stack and pass them to TELNET server, which changes the characters to the corresponding characters.
- The OS cannot accept the text/commands from a TELNET server, but can accept from a terminal driver. Here a pseudoterminal driver accepts the text from the TELNET server and pass them to the OS. The OS then passes the characters to the appropriate application program.

DOMAIN NAME SYSTEM (DNS)

- Machines can understand only IP (numeric) address while people can remember only names.
- Hence IP addresses are mapped to some names and the whole such information in the network are distributed all over the Internet.
- Any host that needs the mapping from names to IP address can contact the nearby computer that holds this information.
- This method is called the DOMAIN NAME SYSTEM (DNS) and the computer that holds such mapping information is called a DNS server.
- The DNS client sends the name to the DNS server which returns the corresponding IP address.
- Consider a file transfer scenario where the user gives a host name instead of IP adrress:



Name Space

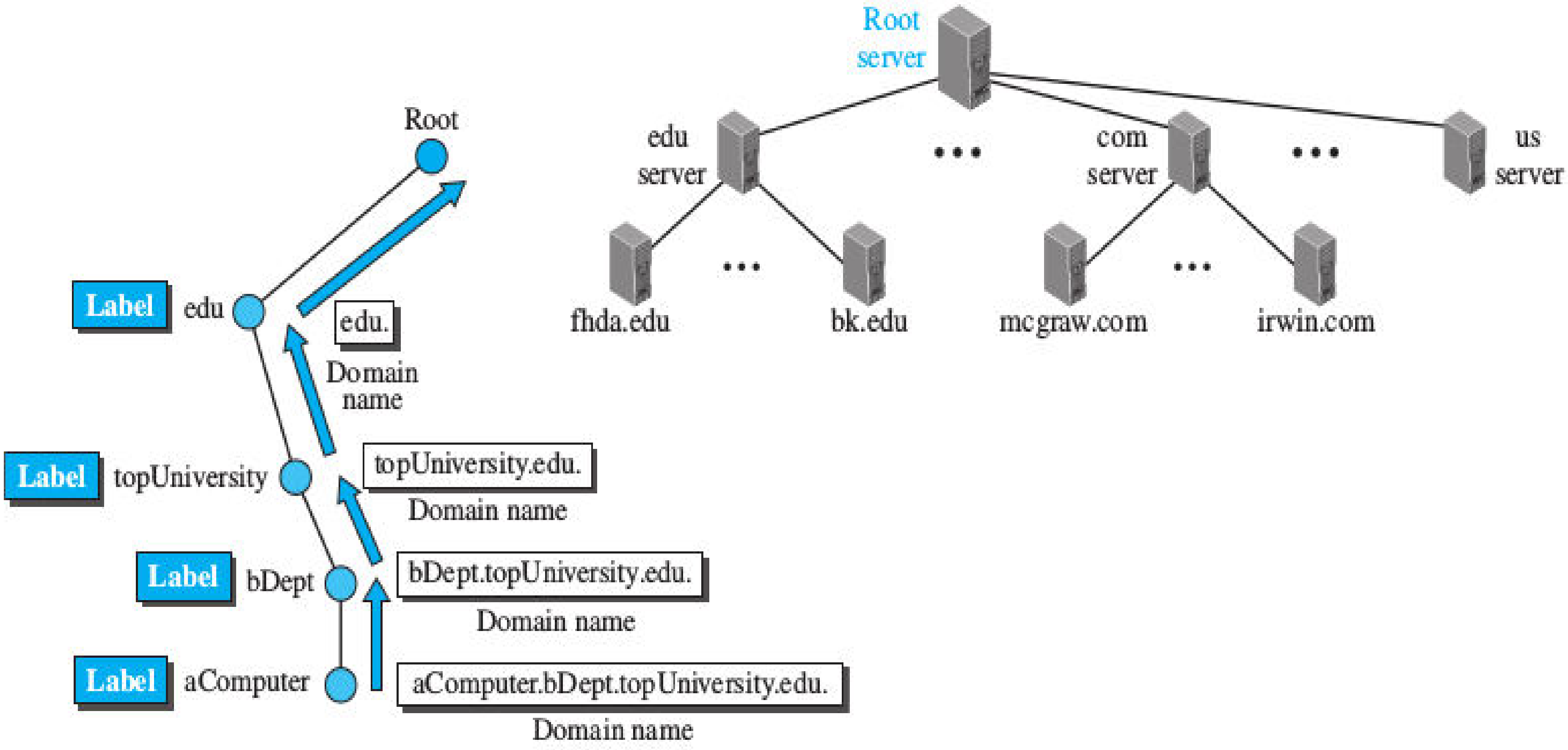
- It is a collection of names and addresses; both are unique.
- Names can be organized in two ways: flat or hierarchical.
- Flat Name Space: A name is assigned to an address, one by one. It can't be used for large number of names-address combinations.
- Hierarchical Name Space: Each name is made of several parts. The first part can define the nature of the organization, the second part can define the name of an organization, the third part can define departments in the organization, and so on.

  Eg: kerala.bsnl.co.in, www.indiana.edu

Domain Name Space

- Designed to implement hierarchical name space.
- The names are defined in an inverted-tree structure with the root at the top. The tree can have only 128 levels: level 0 (root) to level 127.
- Each node in the tree has a label, which is a string with a maximum of 63 characters. The root label is a null string (empty string).
- Each node in the tree has a domain name. A full domain name is a sequence of labels separated by dots (.).
- The domain names are always read from the node up to the root.
- The last label is the label of the root (null).

- If a label is terminated by a null string, it is called a fully qualified domain name (FQDN). Otherwise it is called a partially qualified domain name (PQDN).
- A domain is a subtree of the domain name space.
- For efficiency, domains are stored in different places called domain servers.



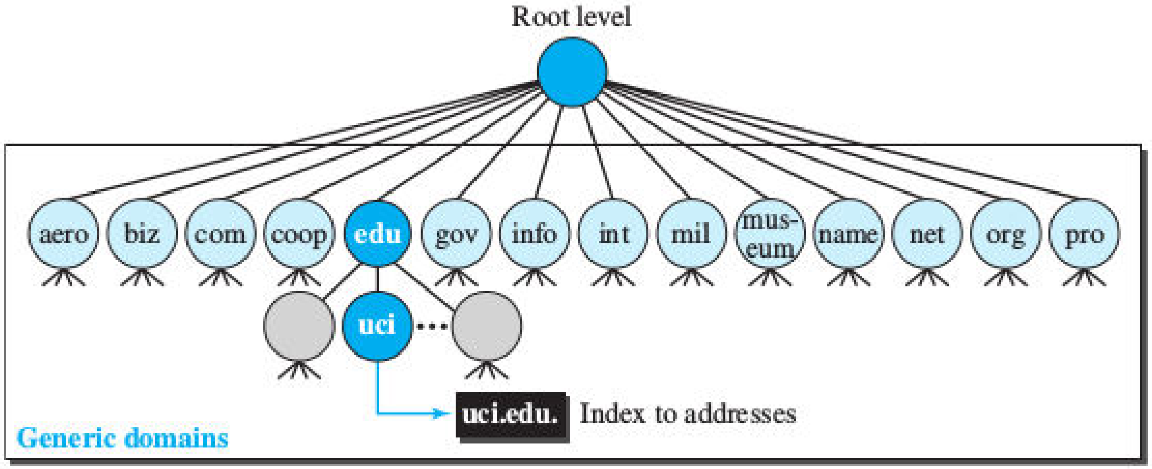
- Since the complete domain hierarchy can't be stored on a single server, it is divided among different servers; each one is said to have an authority of a *zone*.
- A root server is a server whose zone consists of the entire tree.
- DNS defines two types of servers: primary and secondary.
- A primary server is a server that stores a file about the zone for which it is an authority. It is responsible for creating, maintaining, and updating the zone file. It stores the zone file on a local disk.
- A secondary server is a server that transfers the complete information about a zone from another server (primary or secondary) and stores the file on its local disk. The secondary server neither creates nor updates the zone files.

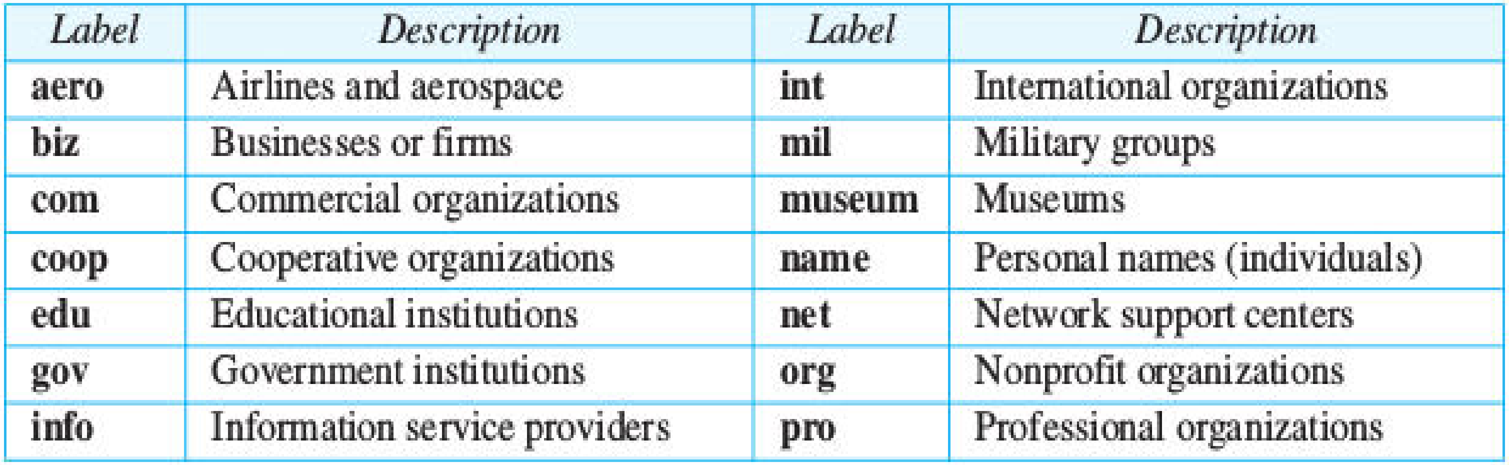
DNS in the Internet

- The domain name space was divided into three: Generic domains, Country domains and Inverse domains. But Inverse domain doesnot exist now.

1. Generic Domains:

- Based on generic behaviour.
- Each node in the tree defines a domain, which is an index to the domain name space database.
- There are 14 generic domain labels.

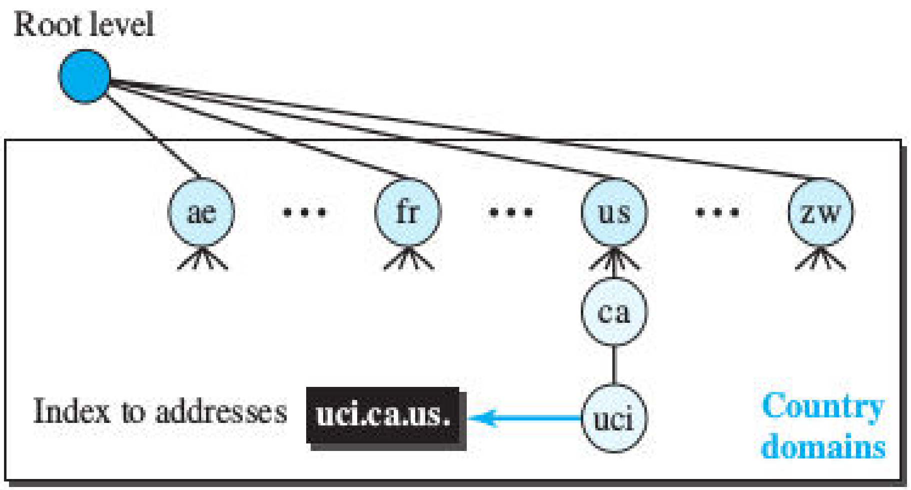




Label	*Description*	*Label*	*Description*
aero	Airlines and aerospace	**int**	International organizations
biz	Businesses or firms	**mil**	Military groups
com	Commercial organizations	**museum**	Museums
coop	Cooperative organizations	**name**	Personal names (individuals)
edu	Educational institutions	**net**	Network support centers
gov	Government institutions	**org**	Nonprofit organizations
info	Information service providers	**pro**	Professional organizations

2. Country Domains

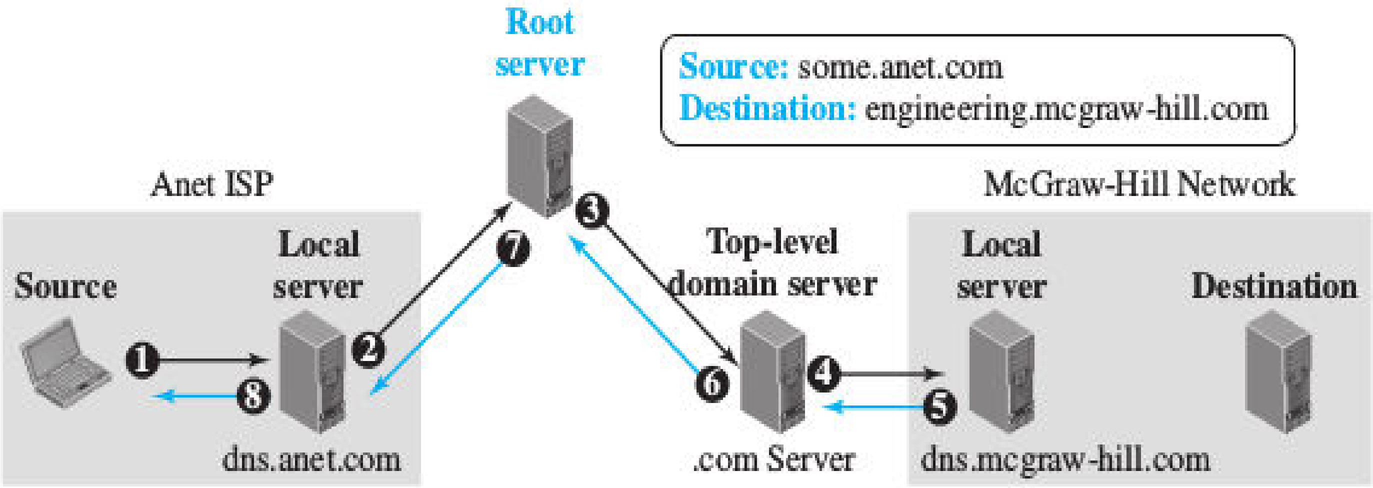
- Uses two-character country abbreviations (Eg: *in* for India).
- Second labels can be organizational, or they can be more specific national designations, and so on. (Eg: *co.in)*



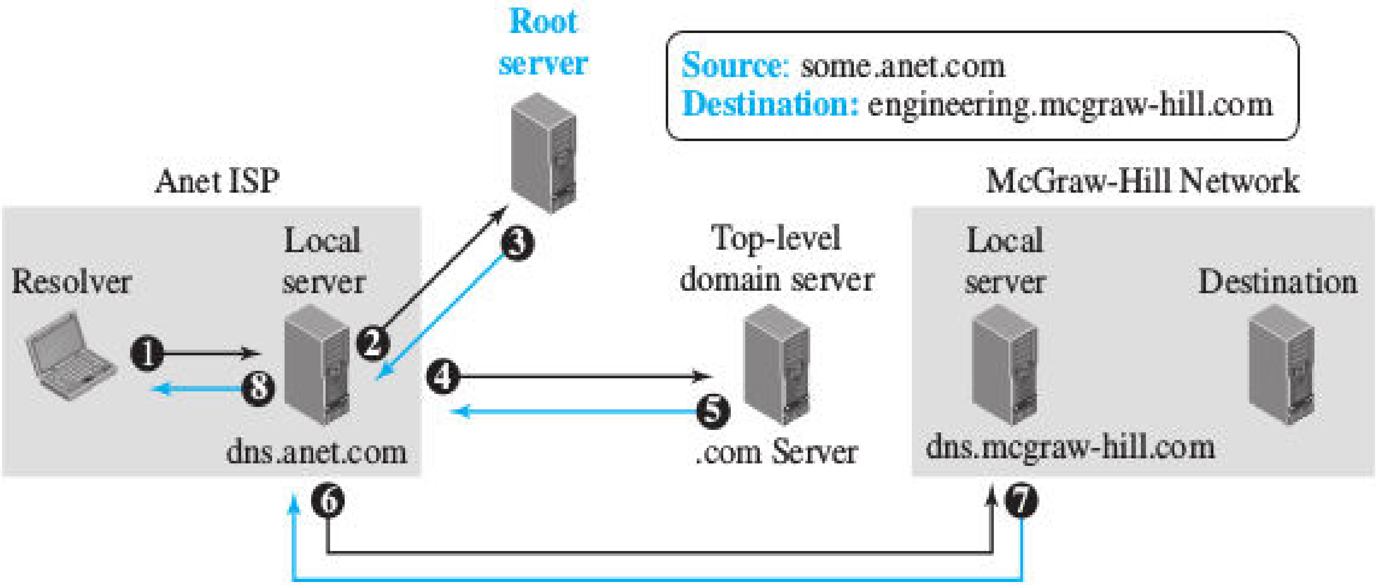
Resolution

- Mapping a name to an address is called name-address resolution.
- A host that needs to map an address to a name or a name to an address calls a DNS client called a *resolver,* which in turn requests the nearest DNS server.
- That DNS server resolves the request. Otherwise, it either refers the resolver to other servers or asks other servers to provide the information.
- After the resolver receives the mapping, it interprets the response to see if it is a real resolution or an error, and finally delivers the result to the process that requested it.
- A resolution can be either recursive or iterative.

- Recursive Resolution:

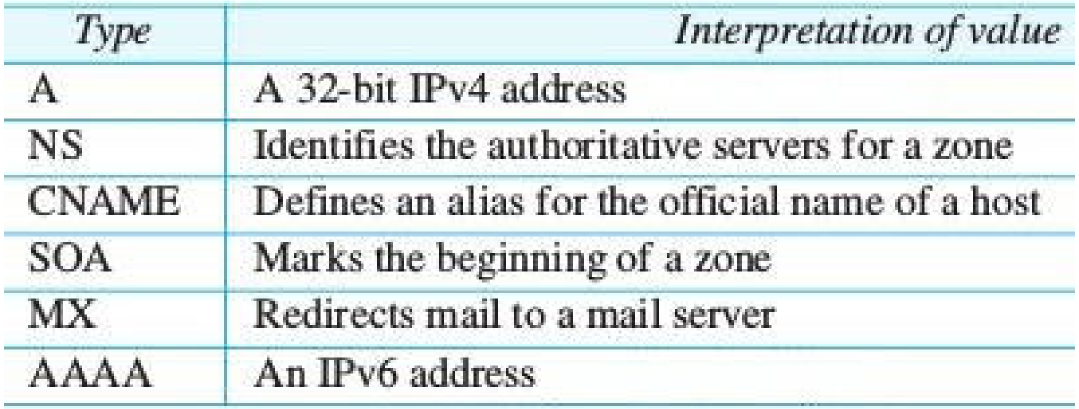


- Iterative Resolution:



Resource Records

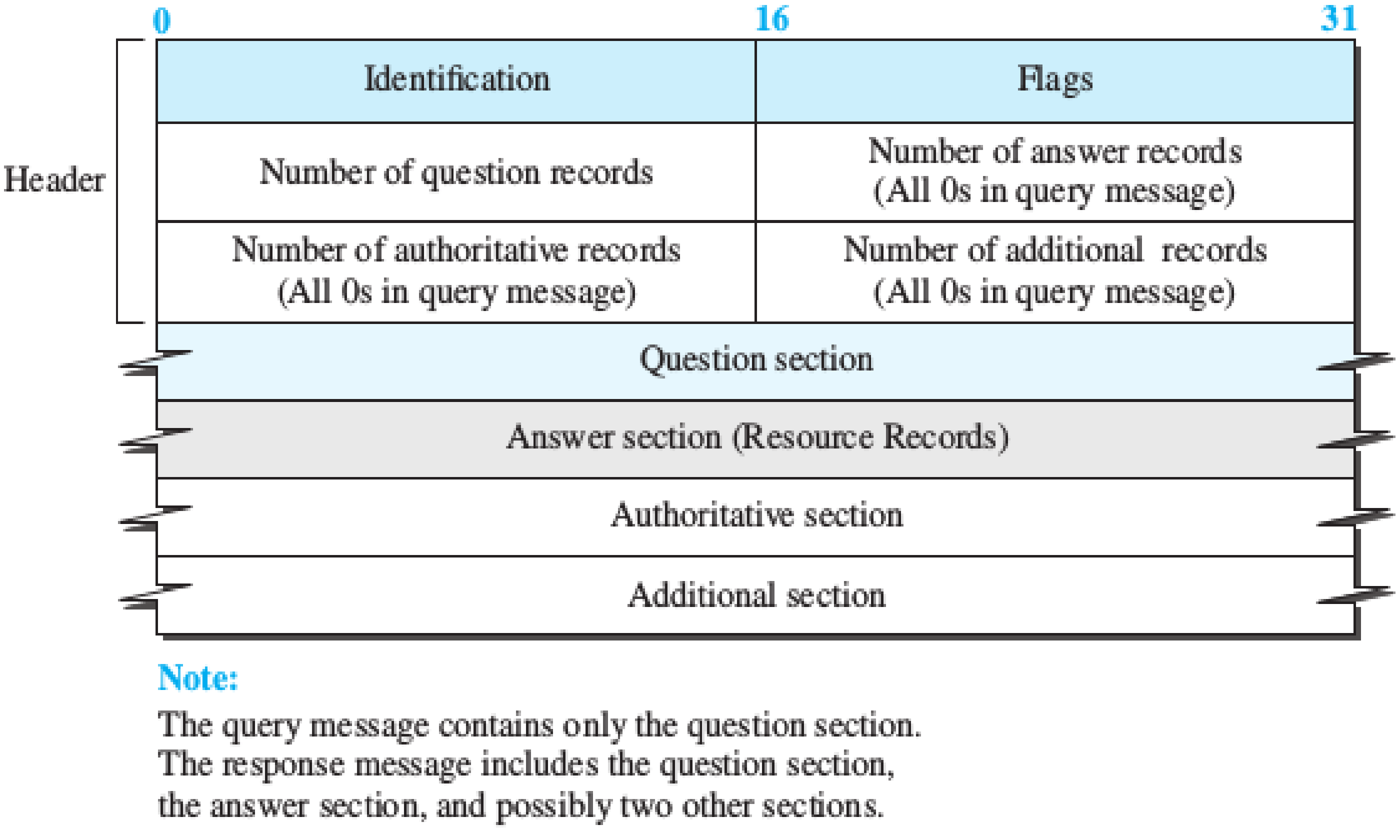
- The zone information associated with a server is implemented as a set of resource records.
- A name server stores a set of resource records.
- A resource record: (Domain Name, Type, Class, TTL, Value)
- Domain Name: identifies the record.
- Type: The Value depends on the Type parameter. (Eg: for Type: A, Value: IP v4 Address)
- Class: Type of network (Here the Internet).
- TTL: Time for which the information is valid.



Type	*Interpretation of value*
A	A 32-bit IPv4 address
NS	Identifies the authoritative servers for a zone
CNAME	Defines an alias for the official name of a host
SOA	Marks the beginning of a zone
MX	Redirects mail to a mail server
AAAA	An IPv6 address

DNS Messages

- Query and Response; both have the same format.



0 16 31

Header	Identification	Flags
	Number of question records	Number of answer records (All 0s in query message)
	Number of authoritative records (All 0s in query message)	Number of additional records (All 0s in query message)
	Question section	
	Answer section (Resource Records)	
	Authoritative section	
	Additional section	

Note:

The query message contains only the question section.
The response message includes the question section,
the answer section, and possibly two other sections.

Identification: To identify a response for a query
Flags: To know whether the message is a query, response or error.

Encapsulation

- DNS can use either TCP or UDP based on the message size.
- If the message size is less than 512 Bytes, UDP is used.
- If it is more than 512 Bytes, TCP is used.
- If the size is not known, UDP is used. On transmitting, if the size exceeds 512 Bytes, the UDP packet is truncated and a new TCP connection is created and query/response starts again.

Dynamic DNS (DDNS)

- Since the Internet is vary large, manual updation of DNS entries isnot possible.
- Dynamic DNS updates the DNS server data dynamically.
- When a new name-IP pair is bound, normally the DHCP informs a primary DNS server.
- The primary server makes changes in its records and notifies the secondary DNS servers actively or pasively.
- Active notification: The primary DNS server sends message to the secondary ones about the updation.
- Passive notification: The secondary DNS servers periodically checks the priary DNS servers whether there is any update.
- In either case the secondary server requests information about the entire zone (called the *zone transfer*).