

## **CS19541 COMPUTER NETWORKS**

### **UNIT-I FUNDAMENTALS AND DATA LINK LAYER**

Building a network – Requirements – Layering and protocols – Internet Architecture – Network software – Application Programming Interface (sockets) - Performance – Link layer Services - Framing – Error Detection and Correction - Reliable transmission

### **UNIT-II MEDIA ACCESS AND INTERNETWORKING**

Media Access Protocols – ALOHA - CSMA/CA/CD –Ethernet – Wireless LANs - 802.11- Bluetooth - Switching and Forwarding - Bridges and LAN Switches – Basic Internetworking- IP Service Model – IP fragmentation - Global Addresses – ARP - DHCP – ICMP- Virtual Networks and Tunnels.

### **UNIT-III ROUTING**

Routing – Network as Graph - Distance Vector – Link State – Global Internet – Subnetting - Classless Routing (CIDR) - BGP- IPv6 – Multicast routing - DVMRP- PIM.

### **UNIT-IV TRANSPORT LAYER**

Overview of Transport layer – UDP – TCP - Segment Format – Connection Management – Adaptive Retransmission - TCP Congestion control - Congestion avoidance (DECbit, RED) – QoS – Application requirements.

### **UNIT-V APPLICATION LAYER**

E-Mail (SMTP, MIME, POP3, IMAP), HTTP – DNS - FTP - Telnet – web services - SNMP – MIB – RMON.

#### **Text Books**

1. Larry L. Peterson, Bruce S. Davie, “Computer Networks: A Systems Approach”, Fifth Edition, Morgan Kaufmann Publishers Inc., 2011.
2. Behrouz A. Forouzan, “Data Communications and Networking”, Fifth Edition, McGrawHill, 2017

#### **Reference Books**

1. William Stallings, “SNMP, SNMPv2, SNMPv3 and RMON 1 and 2”, Third Edition, Pearson Edition, 2009.
2. James F. Kurose, Keith W. Ross, ” Computer Networking - A Top-Down Approach Featuring the Internet”, Seventh Edition, Pearson Education, 2017.
3. Andrew S. Tanenbaum, David J. Wetherall, “Computer Networks”, 5th Edition, Prentice Hall publisher, 2010.
4. William Stallings, “Data and Computer Communications”, Eighth Edition, Pearson Education, 2011.

REC-CSE

### UNIT-I FUNDAMENTALS AND DATA LINK LAYER

Building a network – Requirements – Layering and protocols – Internet Architecture – Network software –Application Programming Interface (sockets) - Performance – Link layer Services - Framing– Error Detection and Correction - Reliable transmission

#### **Computer network**

Computer network is defined as the interconnection of nodes (computers and other devices) connected by a communication channel (wired or wireless) that facilitates communication among users and allows them to share resources (Information, hardware and software resources.)

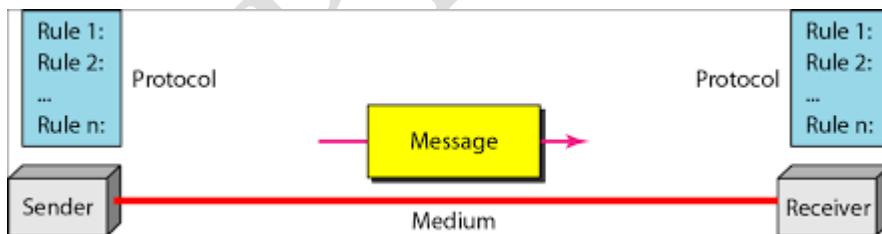
#### **INTRODUCTION**

Data Communications is the transfer of data or information between a sender and a receiver. The sender transmits the data and the receiver receives it.

The effectiveness of a data communication depends on three characteristics,

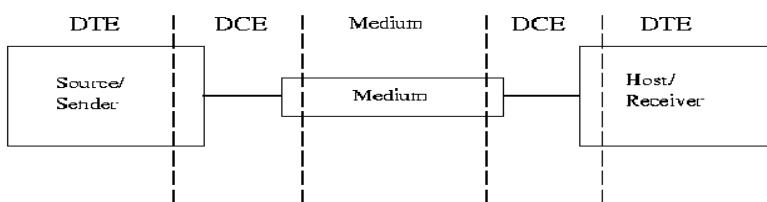
- Delivery** : The system must deliver data to correct destination.
- Accuracy** : The system must deliver data accurately.
- Timeliness** : The system must deliver data in a timely manner.

#### **Components of data communication**



- **Sender:** It is the transmitter of data. Some examples are Terminal, Computer, and Mainframe.
- **Medium:** The communication stream through which the data is being transmitted. Some examples are: Cabling, Microwave, Fiber optics, Radio Frequencies (RF), Infrared Wireless
- **Receiver:** The receiver of the data transmitted. Some examples are Printer, Terminal, Mainframe, and Computer.
- **Message:** It is the data that is being transmitted from the Source/Sender to the Destination/Receiver.

- **Protocol:** It is the set of rules and regulations (resides in the form of software and hardware) that are to be followed for communication. If protocol is not present it implies the nodes are connected but they can't communicate.



**Figure 1.2: DTE and DCE**

- **DCE:** The interface between the Sender and the Medium, and the Medium & the Receiver is called the DCE (Data Communication Equipment) and is a physical piece of equipment.
- **DTE:** Data Terminal Equipment is the Telecommunication name given to the Source and Receiver's equipment.

### **Direction of Data Flow**

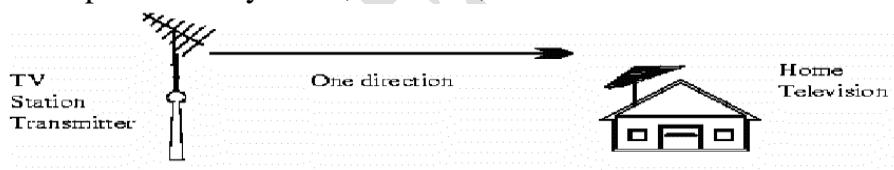
It defines how the data flows between two end points. Based on the direction and time of flow there are three kinds of data flow,

1. Simplex
2. Half-Duplex
3. Full-duplex.

#### **Simplex:**

In this type of data communication, the data flows in only one direction on the data communication line (medium).

Examples are Keyboard, Monitor, Radio and Television broadcasts

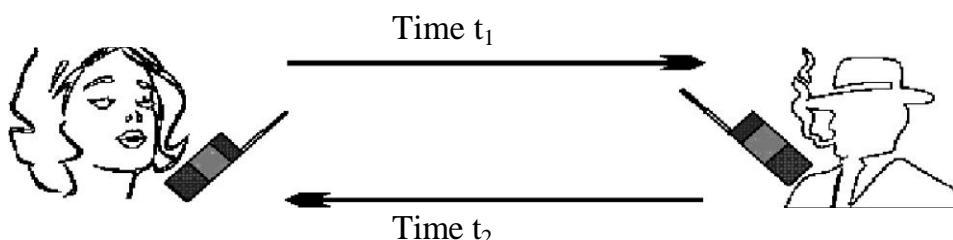


**Figure 1.3: Simplex Data Flow**

#### **Half-Duplex:**

In this type of data communication, the data flows in both directions but at a time in only one direction on the data communication line.

Example Conversation on walkie-talkies is a half-duplex data flow.



**Figure: Half-Duplex Data Flow**

### **Full-Duplex:**

In this type of data communication, the data flows in both directions simultaneously. Example Telephones and Modems

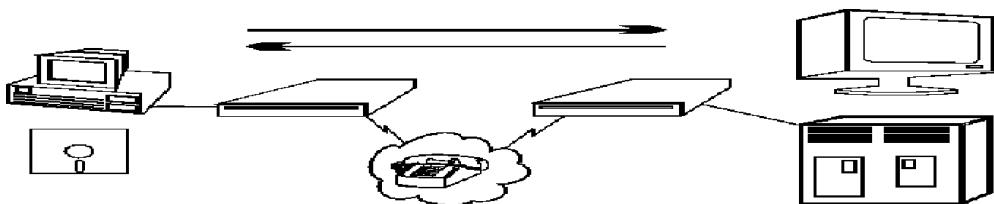


Figure: Full-Duplex Data Flow

### **Types of Connections / Line configuration / Direct links**

There are two types of line configuration, they are

1. point to point
2. multipoint

#### **Point to Point**

- It provides a dedicated link between two devices.
- The entire capacity (bandwidth) of the link is reserved for transmission between these two devices.  
The two devices are connected by means of a pair of wires or using a microwave or satellite link.
- Eg : Computers connected by telephone line  
PPP connection between remote and TV

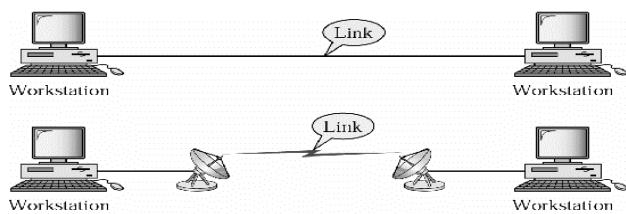


Figure: Point-to-point link

#### **Multipoint (Multiple Access)**

- It is a connection in which more than two devices share a single link.
- In this environment a single channel is shared, either spatially or temporally.
- If several devices can use the link at the same time it said to be spatially shared.
- If the devices take turn to use the link then it is referred to as timesharing.

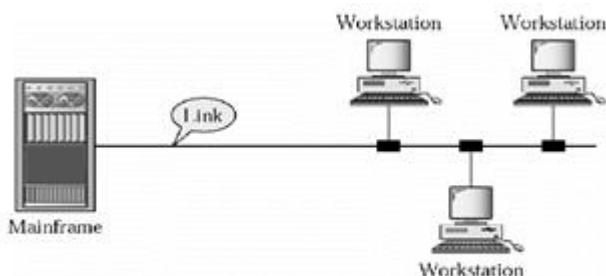


Figure : Multipoint link

## **Different Communication Modes**

### **1) Unicast - one to one**

Unicast packets are sent from host to host. The communication is from a single host to another single host.

### **2) Broadcast – one to all**

Broadcast is when a single device is transmitting a packet to all other devices in a given address range.

### **3) Multicast – one to many**

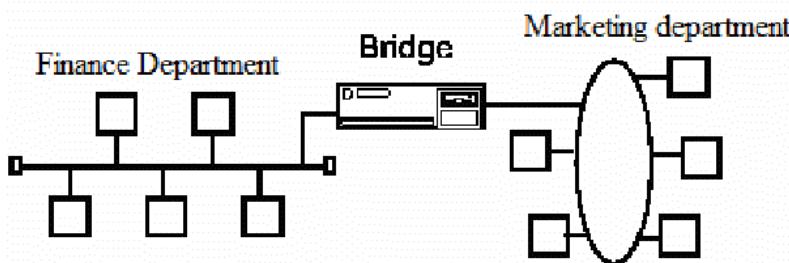
Multicast enables a single device to communicate with a specific set of hosts.

## **Categories of networks**

The three primary categories of network are Local Area Network (LAN), Metropolitan Area Network (MAN), and Wide Area Network (WAN). The category into which a network fall is determined by its size, ownership, the distance it covers and its physical architecture.

### **LAN**

- A LAN is usually privately owned and links the devices in a single office, building or campus.
- A LAN can be as simple as two PCs or it can extend throughout a company. LAN size is limited to a few kilometers.
- The most widely used LAN technology is the Ethernet technology developed by the Xerox Corporation.



**Figure 1.8: Local Area Network**

### **MAN**

- A MAN is designed to extend over an entire city.
- It could be a single network such as cable TV network or connect a number of LANs into a larger network.
- A MAN can be owned by a private company or it may be a service provided by a public company, such as local telephone company.
- Telephone companies provide a popular MAN service called (SMDS) Switched Multi-megabit Data Services.

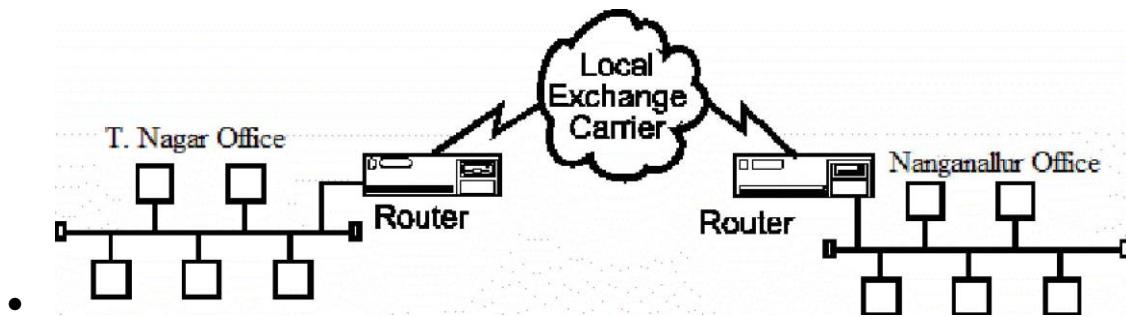


Figure: Metropolitan Area Network

### WAN

- A WAN provides long distance transmission of data, voice, image and video information over large geographic areas.
- Transmission rates are typically 2 Mbps, 34 Mbps, 45 Mbps, 155 Mbps and 625 Mbps. WAN utilize public, leased, or private communication equipment usually in combinations and therefore span an unlimited number of miles.
- A WAN that is wholly owned and used by a single company is referred to as an Enterprise Network.

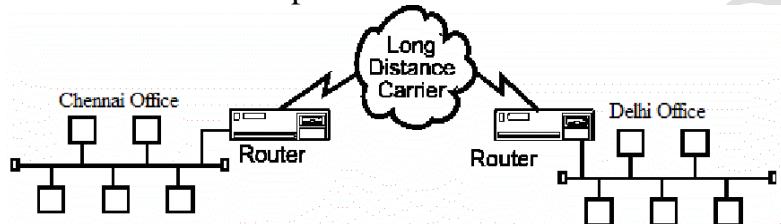


Figure 1.10: Wide Area Network

### PAN

- A Personal Area Network (PAN) is the interconnection of devices within the range of an individual person, typically within a range of 10 meters.
- For example, a person traveling with a laptop, a personal digital assistant (PDA), and a portable printer could interconnect them without having to plug anything in, using some form of wireless technology.
- Typically, this kind of personal area network could also be interconnected without wires to the Internet or other networks.



Figure: Personal Area Network

## **Topology**

Physical Topology refers to the fashion in which nodes in the network is laid out physically. The topology of a network is the geometric representation of the relationship of all the links and the linking devices tone another. It defines the physical layout of the network.

The basic topologies are:

- Mesh
- Star
- Bus
- Ring
- Hybrid (combination of other types)

### **Mesh**

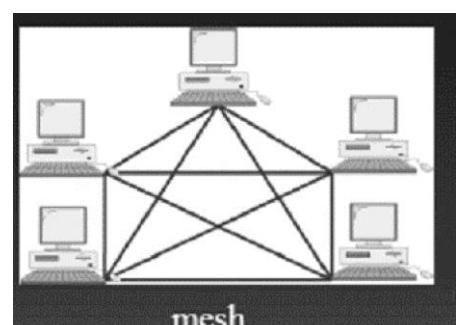
- In a mesh topology each device has a dedicated point to point link to every other device.
- The term dedicated means that the link carries traffic only between the two devices it connects.
- A fully connected mesh network therefore has  $n(n-1)/2$  physical channels to link  $n$  devices. To accommodate that many links every device on the network should have  $(n-1)$  I/O ports.

### **Merits**

- Eliminates the traffic problems that occur when the links are shared by multiple devices.
- If one link becomes unusable, it does not incapacitate the entire system.
- Since every message travels along a dedicated line only the intended recipient will receive the message and hence the data is secure.

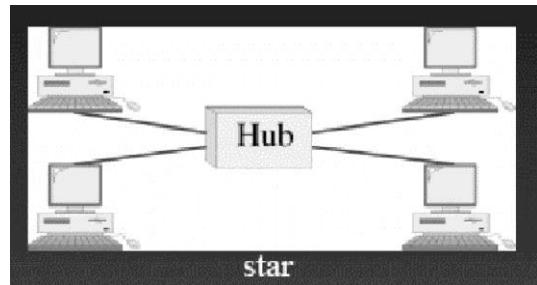
### **Demerits**

- The amount of cabling and the I/O ports required increases with the number of devices connected in the network
- Installation and reconnection are difficult
- The sheer bulk of the wire accommodates more space than available.
- The hardware required to connect each link can be prohibitively expensive.



## **Star**

- Each device has a dedicated point to point link only to a central controller usually called a hub.
- If one device has to send data to another it sends the data to the controller, which then relays the data to the other connected device.



## **Merits**

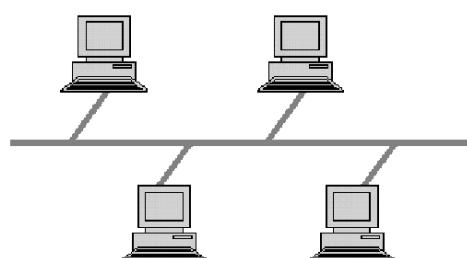
- Less expensive than a mesh topology. Each device needs only one link and I/O port.
- Installation and reconfigure is easy.
- Robustness. If one link fails only that link is affected.
- Requires less cable than a mesh.

## **Demerits**

- Require more cable compared to bus and ring topologies.
- Failure of the central controller incapacitates the entire network.

## **Bus**

- One long cable acts as a backbone to link all the devices in a network.
- Nodes are connected to the bus cable by drop lines and taps.
- A drop line is a connection running between the device and the main cable.
- A tap is a connector that either splices into the main cable to create a contact with a metallic core.
- As the signal travels farther and farther, it becomes weaker. So there is limitation in the number of taps a bus can support and on the distance between those taps.



## **Merits**

- Ease of installation.
- Bus uses less cabling than mesh or star topologies.

Figure 1.15: Bus Topology

## **Demerits**

- Difficult reconnection and isolation.
- Signal reflection at the taps can cause degradation in quality.
- A fault or break in the bus cable stops all transmission.
- It also reflects signals back in the direction of origin creating noise in both directions.

## **Ring**

- Each device has a dedicated point to point connection only with 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 the destination.
- Each device in the ring incorporates a repeater

### **Merits:**

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

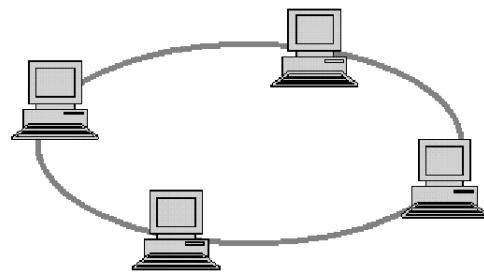


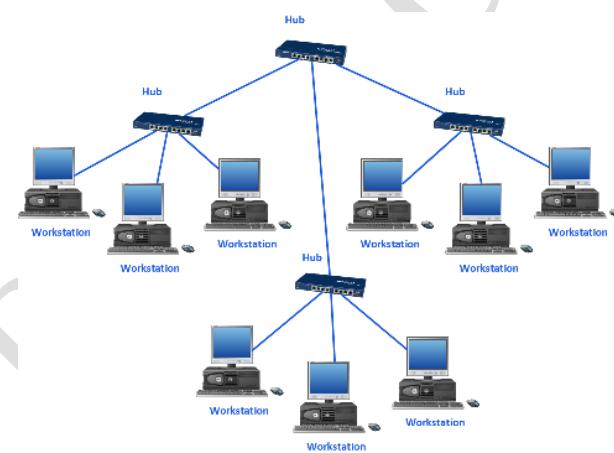
Figure 1.16: Ring Topology

### **Demerits**

- A break in the ring disables the entire network. It can be solved by using a dual ring or a switch capable of closing off the break.

## **Hybrid Topology**

- A hybrid topology is a type of network topology that uses two or more other network topologies, including bus topology, mesh topology, ring topology, star topology, and tree topology.



## **Requirements**

**The following are the requirements to be followed to build any network,**

- ✓ Identification of constraints and requirements
- ✓ Connectivity need to be decided
- ✓ Cost-effective resource sharing
- ✓ Support for common services

## Identification of constraints and requirements of a network

Three groups of people might list their requirements for a network,

1. Application Programmer
  - List the services that his application needs: delay bounded delivery of data
2. Network Designer
  - Designs a cost-effective network with sharable resources
3. Network Provider
  - List the characteristics of a system that is easy to manage

## Connectivity in a network:

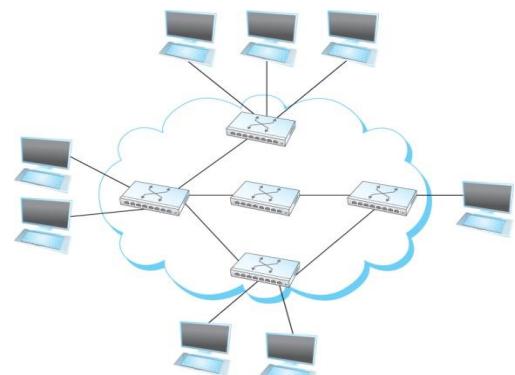
### Links, Nodes, and Clouds

Network connectivity occurs at many different levels. At the lowest level, a network can consist of two or more computers directly connected by some physical medium, such as a coaxial cable or an optical fiber. We call such a physical medium a link, and we often refer to the computers it connects as nodes.

A set of computers can be indirectly connected. A set of independent networks (clouds) are interconnected to form an internetwork. A node that is connected to two or more networks is commonly called a router or gateway. A router/gateway forwards messages from one network to another.

### Switched Network

Those nodes that are attached to at least two links run software that forwards data received on one link out on another. If organized in a systematic way, these forwarding nodes form a switched network. There are numerous types of switched networks, of which the two most common are circuit switched and packet switched.



### Types of switched networks

#### **1. Circuit Switched**

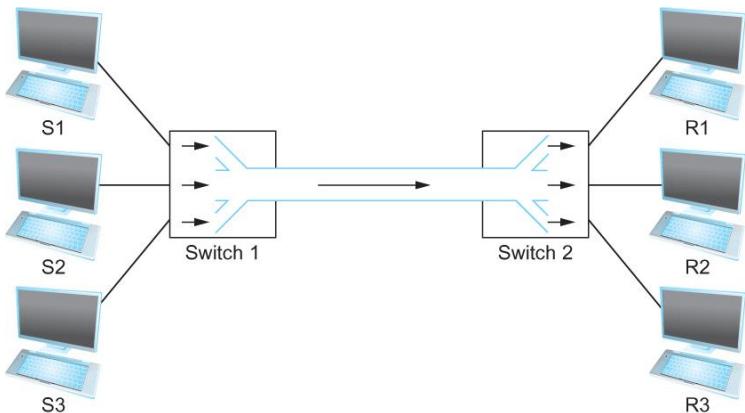
First establishes a dedicated circuit across a sequence of links and then allows the source node to send a stream of bits across this circuit to a destination node.

#### **2. Packet Switched networks**

It uses a strategy called store-and-forward. Each node in a store-and-forward network first receives a complete packet over some link, stores the packet in its internal memory, and then forwards the complete packet to the next node.

## Cost-Effective Resource Sharing

- Resource: links and nodes
- How to share a link?
  - **Multiplexing**  
Analogy to a timesharing computer system.
  - **De-multiplexing**



**Multiplexing multiple logical flows over a single physical link**

### Synchronous Time-division Multiplexing (STDM)

- Equal-sized quanta
- Round-robin fashion
- Time slots/data transmitted in predetermined slots

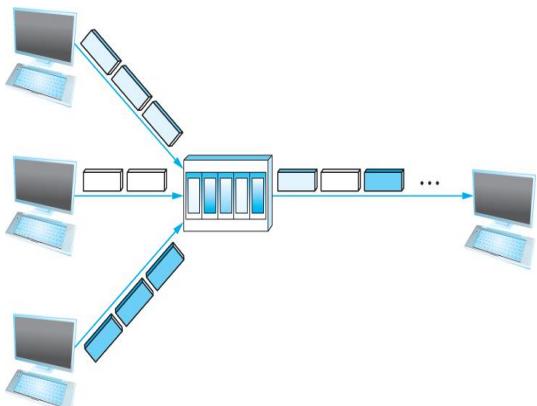
### Frequency Division Multiplexing(FDM)

Eg:Diff. TV stations with diff. frequencies.

- Both STDM and FDM waste resources and hard to accommodate changes (fixed time slots and frequencies)

### Statistical Multiplexing

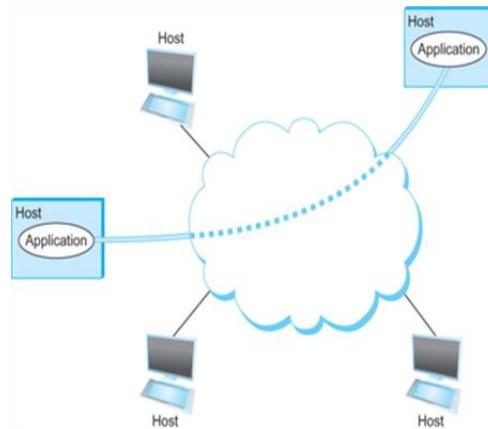
Like STDM: sharing over time but data is transmitted based on demand rather than during a predetermined time slot.



A switch multiplexing packets from multiple sources onto one shared link

### Support for Common Services

- Logical Channels
  - Application-to-Application communication path or a pipe
- Client/Server
- Two types of communication channel
  - Request/Reply Channels
  - Message Stream Channels



Process communicating over an abstract channel

## **NETWORK ARCHITECTURES**

Network designers have developed general blueprints—usually called network architectures—that guide the design and implementation of networks.

1. Layering & protocols
2. OSI layers
3. Internet Architecture

### **Layering and Protocols**

Layering provides two nice features.

- It decomposes the problem of building a network into more manageable components.
- It provides a more modular design.

First, it decomposes the problem of building a network into more manageable components. Rather than implementing a monolithic piece of software that does everything you will ever want, you can implement several layers, each of which solves one part of the problem.

Second, it provides a more modular design. If you decide that you want to add some new service, you may only need to modify the functionality at one layer, reusing the functions provided at all the other layers.

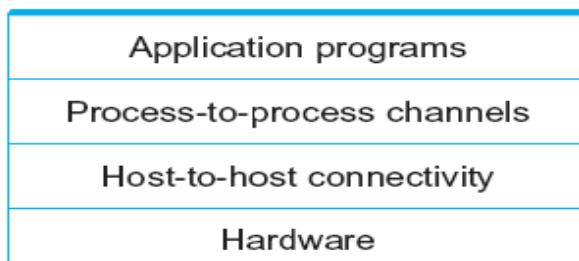
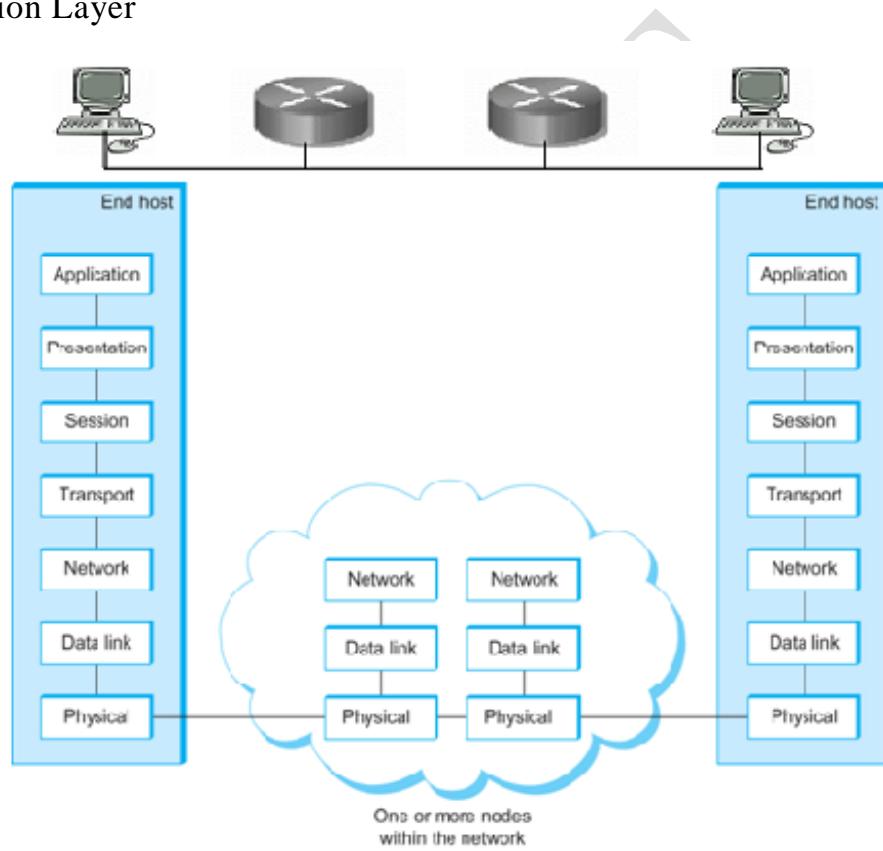


Fig: Example of a layered network system

## **ISO - OSI reference Model**

- The International Standards Organization (ISO) - Open Systems Interconnect (OSI) is a standard which defines a set of rules to govern the data communication between two devices without worrying about the underlying architecture of the devices. It describes the functionalities for transfer of data between each layer. Each layer has a specific function.
    - *ISO is the organization; OSI is the model.*
    - There are 7 Layers in the OSI model
1. Physical Layer
  2. Datalink Layer
  3. Network Layer
  4. Transport Layer
  5. Session Layer
  6. Presentation Layer
  7. Application Layer



### **Two interfaces**

- Service interface
- Peer interface

Service interface- defines the operations that local objects can perform on the protocol.

Peer interface- defines the form and meaning of messages exchanged between protocol peers to implement the communication service.

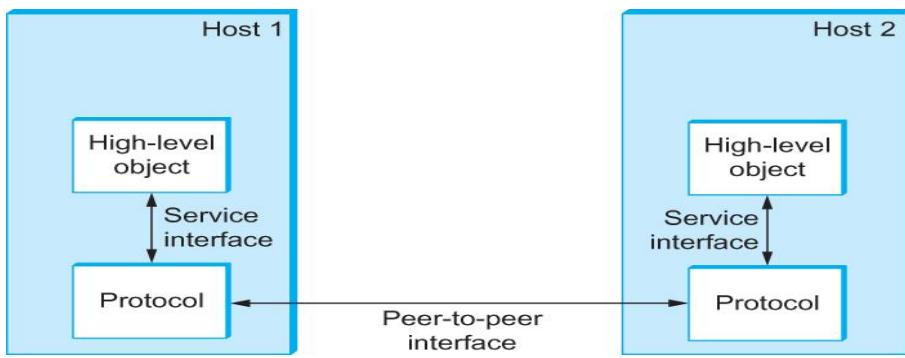
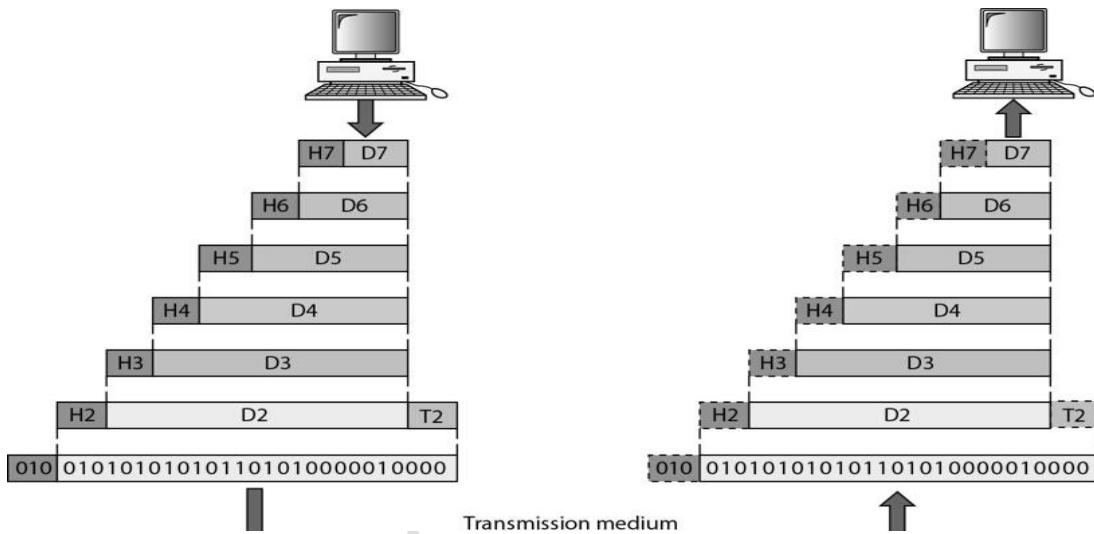


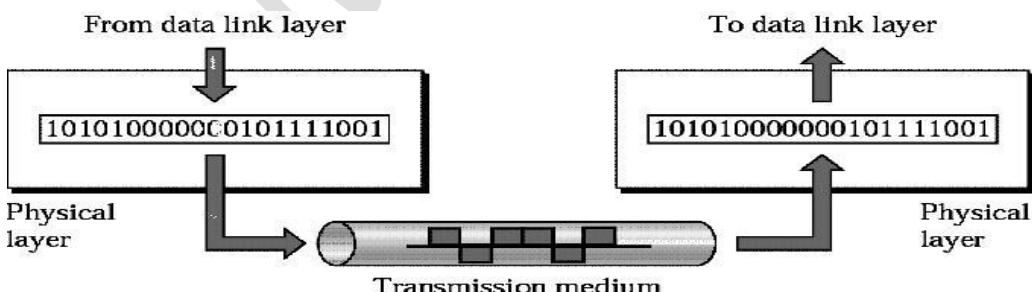
Fig : Service and Peer Interfaces

### A Data exchange using the OSI model



#### 1. PHYSICAL LAYER (bit level transmission)

The physical layer coordinates the functions required to transmit a bit stream over a physical medium. It is responsible for moving bits from one node to the next.



The physical layer is concerned with the following:

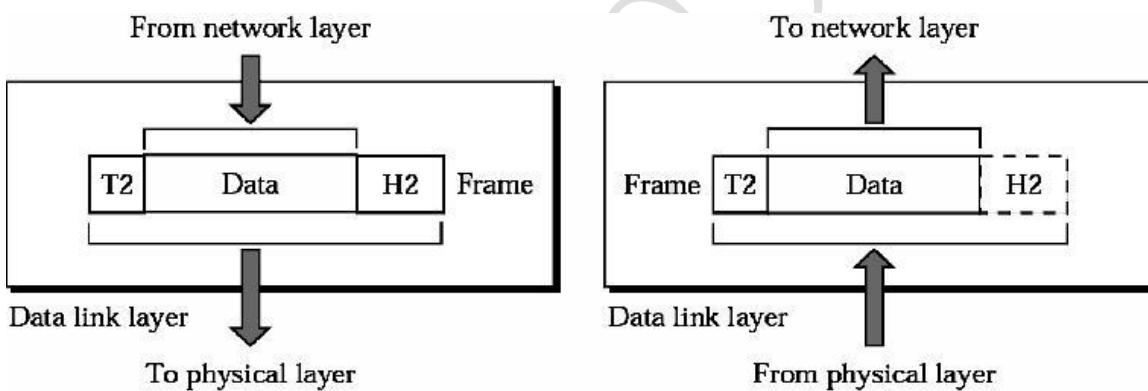
- ✓ **Physical characteristics of interfaces and media** - The physical layer defines the characteristics of the interface between the devices and the transmission medium.
- ✓ **Representation of bits** - To transmit the stream of bits, it must be encoded to signals. The physical layer defines the type of encoding.
- ✓ **Data Rate or Transmission rate** - The number of bits sent each second – is also defined by the physical layer.

- ✓ **Synchronization of bits** - The sender and receiver must be synchronized at the bit level. Their clocks must be synchronized.
- ✓ **Line Configuration** - In a point-to-point configuration, two devices are connected together through a dedicated link. In a multipoint configuration, a link is shared between several devices.
- ✓ **Physical Topology** - The physical topology defines how devices are connected to make a network. Devices can be connected using a mesh, bus, star or ring topology.
- ✓ **Transmission Mode** - The physical layer also defines the direction of transmission between two devices: simplex, half-duplex or full-duplex.

## 2. **DATA LINK LAYER**

(Node to node delivery or hop to hop delivery of frames)

It is responsible for transmitting frames from one node to next node.



The other responsibilities of this layer are

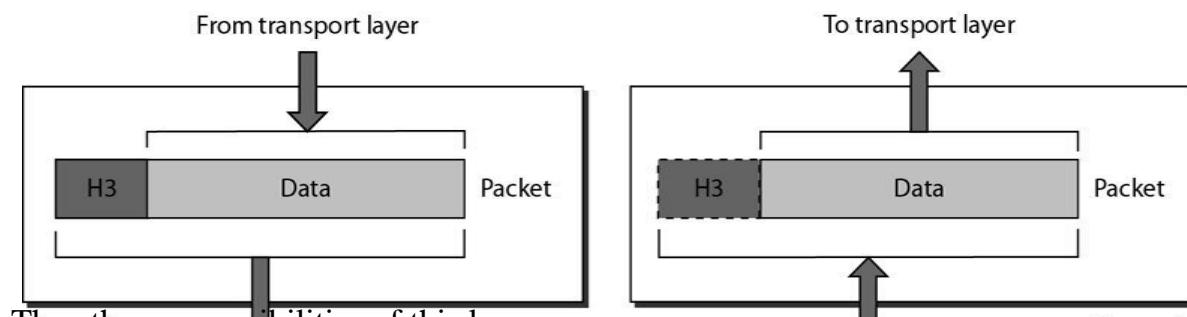
- ✓ **Framing** - Divides the stream of bits received into data units called frames.
- ✓ **Physical addressing** – If frames are to be distributed to different systems on the n/w, data link layer adds a header to the frame to define the sender and receiver.
- ✓ **Flow control**- If the rate at which the data are absorbed by the receiver is less than the rate produced in the sender, the Data link layer imposes a flow control mechanism.
- ✓ **Error control**- Used for detecting and retransmitting damaged or lost frames and to prevent duplication of frames. This is achieved through a trailer added at the end of the frame.
- ✓ **Access control** -Used to determine which device has control over the link at any given time.

### **3. NETWORK LAYER**

**(Source to destination delivery of individual packets)**

This layer is responsible for the delivery of packets from source to destination. It is mainly required, when it is necessary to send information from one network to another.

It ensures that each packet gets from its point of origin to its final destination.

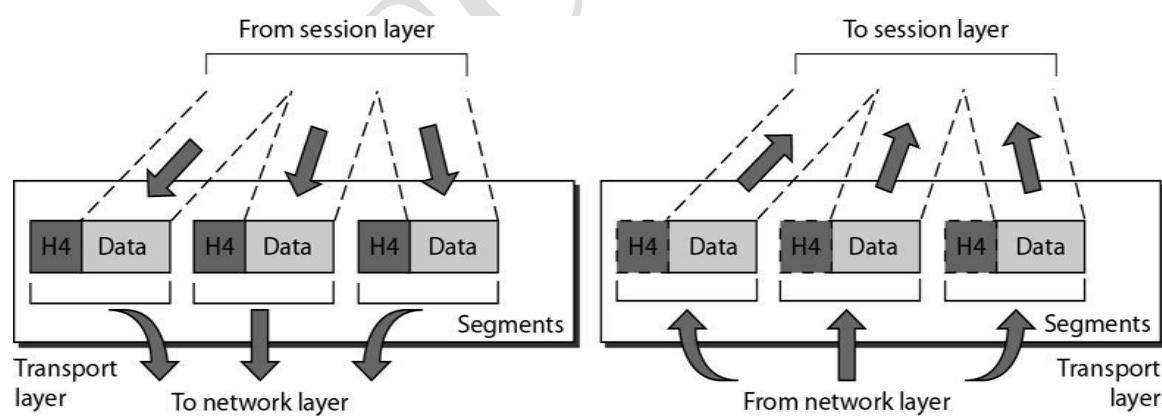


- ✓ **Logical addressing** - If a packet passes the n/w boundary, we need another addressing system for source and destination called logical address.
- ✓ **Routing** – The devices which connects various networks called routers are responsible for delivering packets to final destination.

### **4. TRANSPORT LAYER**

**(Source to destination delivery of entire message)**

- ✓ It is responsible for **Process to Process** delivery of entire message.
- ✓ It also ensures whether the message arrives in order or not.



The other responsibilities of this layer are

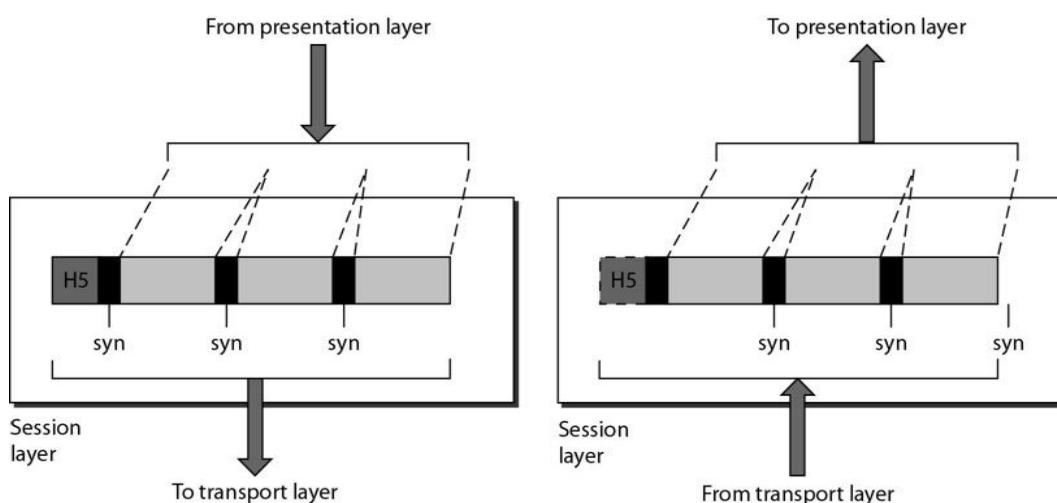
- ✓ **Port addressing** - The header in this must therefore include a address called port address. This layer gets the entire message to the correct process on that computer.
- ✓ **Segmentation and reassembly** - The message is divided into segments and each segment is assigned a sequence number. These numbers are arranged correctly on the arrival side by this layer.

- ✓ **Connection control** - This can either be **connectionless or connection-oriented**. The connectionless treats each segment as an individual packet and delivers to the destination. The connection-oriented makes connection on the destination side before the delivery. After the delivery the termination will be terminated.
- **Flow and error control** - Similar to data link layer, but process to process take place. ie end to end error & flow control

## 5. SESSION LAYER

**(Responsible for dialog control and synchronization)**

This layer establishes, manages and terminates connections between applications.



The other responsibilities of this layer are

- ✓ **Dialog control** - This session allows two systems to enter into a dialog either in half duplex or full duplex.
- ✓ **Synchronization** - This allows to add checkpoints into a stream of data. Example: If a system is sending a file of 2000 pages, check points may be inserted after every 100 pages to ensure that each 100 page unit is advised and acknowledged independently. So if a crash happens during the transmission of page 523, retransmission begins at page 501, pages 1 to 500 need not be retransmitted.

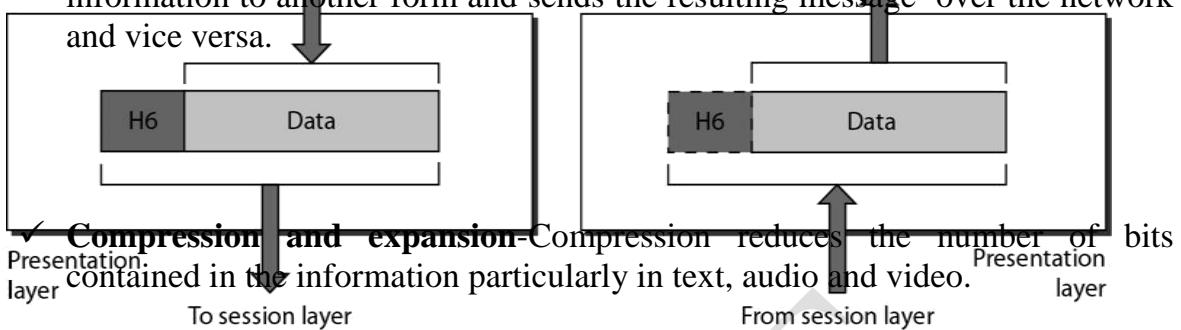
## 6. PRESENTATION LAYER

**Presentation Layer is concerned with the syntax and semantics of the information exchanged between two systems.**

The presentation layer is responsible for translation, compression, and encryption. It is concerned with the syntax and semantics of information exchanged between two systems.

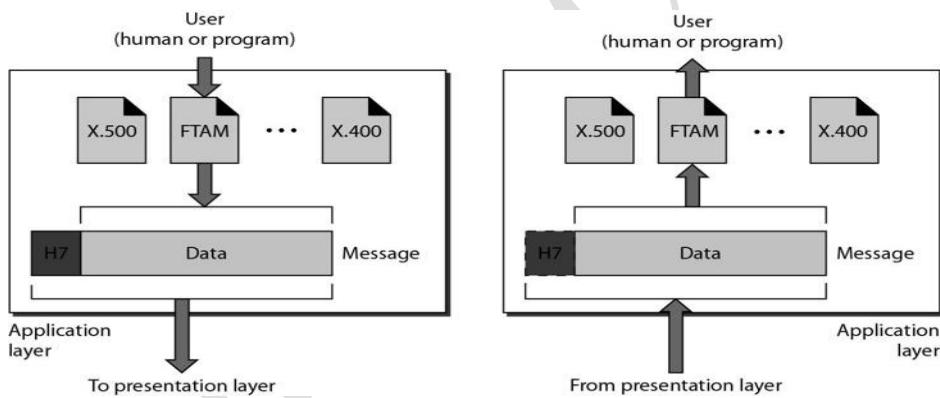
The other responsibilities of this layer are

- ✓ **Translation** – Different computers use different encoding system, this layer is responsible for interoperability between these different encoding methods. It will change the message into some common format.
- ✓ **Encryption and decryption**-It means that sender ~~transforms~~ <sup>applies</sup> the original information to another form and sends the resulting message over the network and vice versa.



## 7. APPLICATION LAYER

This layer enables the **user to access the network resources**. This allows the user to log on to remote user.



The other responsibilities of this layer are

- ✓ **FTAM(file transfer,access,mgmt)** - Allows user to access files in a remote host.
- ✓ **Network Virtual terminal:** A network virtual terminal is a software version of a physical terminal and allows a user to log on to a remote host.
- ✓ **Mail services** - Provides email forwarding and storage.
- ✓ **Directory services** - Provides database sources to access information about various sources and objects.

## Internet(TCP/IP) Architecture

The Internet architecture, which is sometimes called the TCP/IP architecture after its two main protocols. The Internet architecture evolved out of experiences with an earlier packet-switched network called the ARPANET. Both the Internet and the ARPANET were funded by the Advanced Research Projects Agency (ARPA), one of the research and development funding agencies of the U.S. Department of Defense. The Internet and ARPANET were around before the OSI model, and the experience gained from building them was a major influence on the OSI reference model.

### Layers of TCP / IP model

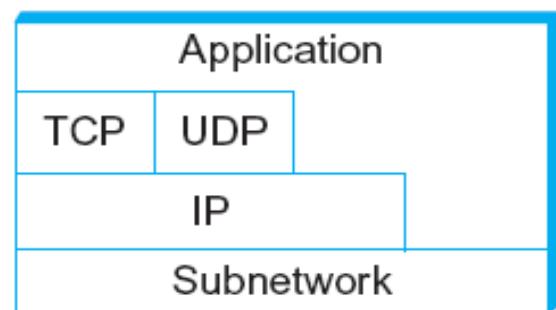
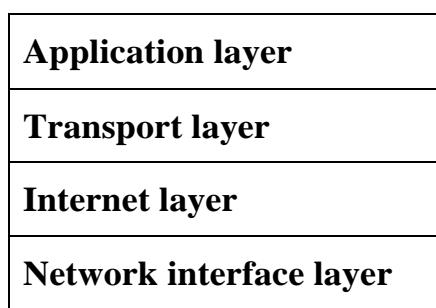
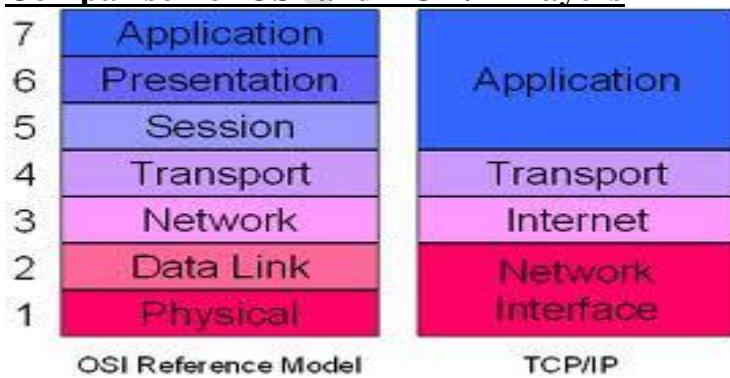


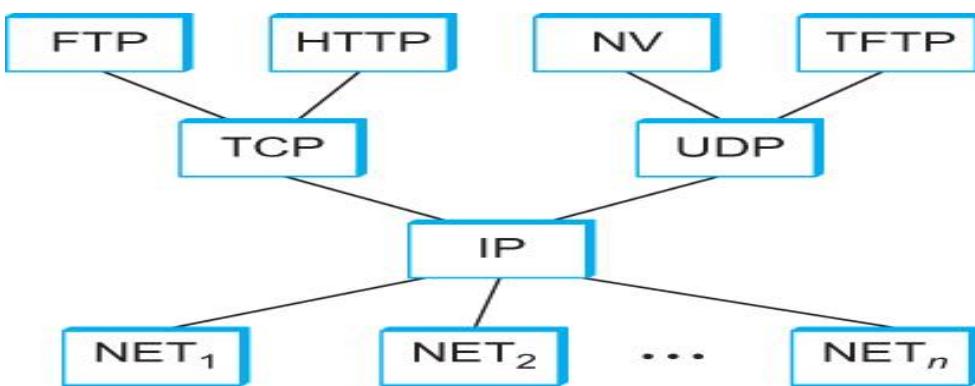
Fig: Alternative view of the Internet architecture. The “Network Interface” layer shown here is sometimes referred to as the “sub-network” or “link” layer.

While the 7-layer OSI model can, with some imagination, be applied to the Internet, a 4-layer model is often used instead.

The TCP/IP protocol suite was developed prior to the OSI model. Therefore, the layers in the TCP/IP protocol suite do not exactly match those in the OSI model.

### Comparison of OSI and TCP / IP layers





**fig: Internet Protocol Graph**

At the lowest level is a wide variety of network protocols, denoted NET<sub>1</sub>, NET<sub>2</sub>, and so on. In practice, these protocols are implemented by a combination of hardware (e.g., a network adaptor) and software (e.g., a network device driver). For example, you might find Ethernet or wireless protocols (such as the 802.11 Wi-Fi standards) at this layer. (These protocols in turn may actually involve several sublayers, but the Internet architecture does not presume anything about them.)

The second layer consists of a single protocol—the Internet Protocol (IP). This is the protocol that supports the interconnection of multiple networking technologies into a single, logical internetwork.

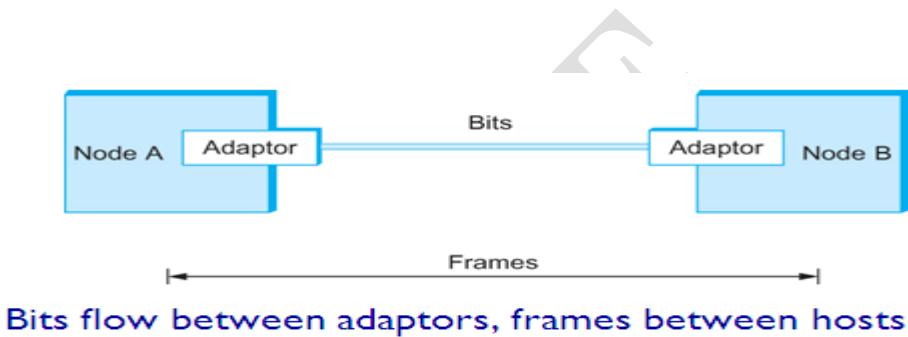
The third layer contains two main protocols—the Transmission Control Protocol (TCP) and the User Datagram Protocol (UDP). TCP and UDP provide alternative logical channels to application programs: TCP provides a reliable byte-stream channel, and UDP provides an unreliable datagram delivery channel (datagram may be thought of as a synonym for message). In the language of the Internet, TCP and UDP are sometimes called end-to-end protocols, although it is equally correct to refer to them as transport protocols.

Running above the transport layer is a range of application protocols, such as HTTP, FTP, Telnet (remote login), and the Simple Mail Transfer Protocol (SMTP), that enable the interoperation of popular applications.

## Framing-Datalink Layer

Data link layer divides the stream of bits received from the upper layer (network layer) into manageable data units called frames. It adds a header to the frame to define the physical address (source address & destination address).

- Blocks of data (called frames at this level) are exchanged between nodes.
- It is the network adaptor that enables the nodes to exchange frames.
- When node A wishes to transmit a frame to node B, it tells its adaptor to transmit a frame from the node's memory.
- This results in a sequence of bits being sent over the link.
- The adaptor on node B then collects together the sequence of bits arriving on the link and deposits the corresponding frame in B's memory.
- Recognizing exactly what set of bits constitutes a frame—that is, determining where the frame begins and ends—is the central challenge faced by the adaptor.



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

**Variable-Size Framing:** In variable-size framing, we need a way to define the end of the frame and the beginning of the next. Three approaches were used for this purpose:

### Framing Protocols:

- (i) Byte-oriented protocols
- (ii) Bit-oriented protocols
- (iii) Clock-based protocols

### Byte-oriented protocols:

- sentinel approach
  1. BISYNC – Binary Synchronous Communication
  2. PPP – Point-to-point Protocol
- Byte-counting approach
  3. DDCMP- Digital Data Communication Message Protocol

### **Bit-oriented protocols:**

- (i) HDLC – High Level Data Link Control

### **Clock-based protocols:**

- (i) SONET - Synchronous Optical Network

## **Byte oriented Protocols**

### **Sentinel Approach**

- It uses special characters known as sentinel characters to indicate where the frame starts and ends.

### **Binary Synchronous Communication (BISYNC)**

- The beginning of a frame is denoted by sending a special SYN (synchronization) character.
- The data portion of the frame is then contained between special *sentinel characters*: STX (start of text) and ETX (end of text).
- The SOH (start of header) field serves much the same purpose as the STX field.



## **BISYNC Frame Format**

- The problem with the sentinel approach, of course, is that the ETX character might appear in the data portion of the frame.
- BISYNC overcomes this problem by "escaping" the ETX character by preceding it with a DLE (data-link-escape) character whenever it appears in the body of a frame; the DLE character is also escaped (by preceding it with an extra DLE) in the frame body. This approach is often called **byte stuffing** or **character stuffing** because extra characters are inserted in the data portion of the frame.

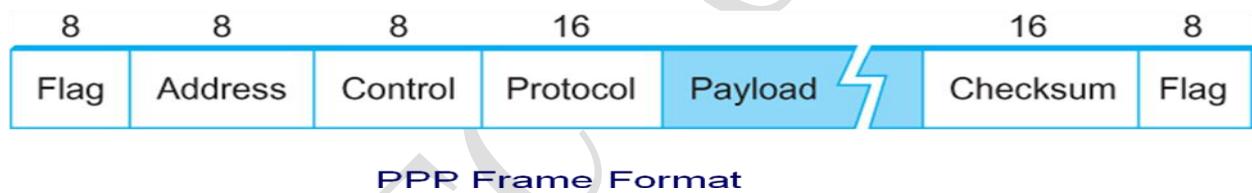
### **Example**

- **Solution: Byte-stuffing** (or "character stuffing")
  - Sender: insert a special **escape character** DLE before any occurrence of ETX in Data portion of frame
    - Data Link Escape (DLE) = 00010000
    - For consistency, must also "escape" any occurrences of DLE in the data
  - Receiver: while looking for ETX, if DLE is encountered, throw it away and treat the following character as data



### **PPP - Point-to-Point Protocol (PPP):**

- It is similar to BISYNC in that it uses character stuffing. The format for a PPP frame is given below.
- The special start-of-text character, denoted as the Flag field is 01111110, which is byte stuffed if it occurs within the payload field.
- The Address and Control fields usually contain default values.
- The Address field which is always set to the binary value 11111111, indicates that all stations are to accept the frame. This value avoids the issue of using data link addresses.
- The default value of the Control field is 00000011. This value indicates an unnumbered frame. In other words, PPP does not provide reliable transmission using sequence numbers and acknowledgements.
- The Protocol field is used for demultiplexing. It identifies the high-level protocol such as IP or IPX (an IP-like protocol developed by Novell).
- The frame payload size can be negotiated, but it is 1500 bytes by default.
- The Checksum field is either 2 (by default) or 4 bytes long.
- The PPP frame format is unusual in that several of the field sizes are negotiated rather than fixed.
- This negotiation is conducted by a protocol called LCP (Link Control Protocol). PPP and LCP work in tandem: LCP is also involved in establishing a link between two peers when both sides detect the **carrier signal**.



### **Byte-counting approach**

#### **DDCMP protocol – Digital Data Communication Message Protocol:**

- The COUNT field specifies how many bytes are contained in the frame's body.
- One danger with this approach is that a transmission error could corrupt the COUNT field, in which case the end of the frame would not be correctly detected. This is sometimes called a **framing error**.
- The receiver will then wait until it sees the next SYN character to start collecting the bytes that make up the next frame.



**DDCMP Frame Format**

## Bit-Oriented Protocols (HDLC)

- A bit oriented protocol is not concerned with byte boundaries—it simply views the frame as a collection of bits.
- The Synchronous Data Link Control (SDLC) protocol developed by IBM is an example of a bit-oriented protocol; SDLC was later standardized by the ISO as the High-Level Data Link Control (HDLC) protocol.



- HDLC denotes both the beginning and the end of a frame with the distinguished bit sequence 01111110.
- This sequence is also transmitted during any times that the link is idle so that the sender and receiver can keep their clocks synchronized.
- Because this sequence might appear anywhere in the body of the frame—in fact, the bits 01111110 might cross byte boundaries—bit-oriented protocols use a technique known as bit stuffing.

**Bit stuffing in the HDLC protocol works as follows.**

- On the sending side, whenever 0 followed by five consecutive 1's has been seen in the body of the message the sender inserts a 0.
- On the receiving side, when 0 followed by five consecutive 1's arrive, the receiver makes its decision based on the next bit it sees (i.e., the bit following the five 1's).
- If the next bit is 0, it must have been stuffed, and so the receiver removes it.
- If the next bit is a 1, then one of two things is true:  
Either this is the end-of frame marker or an error has been introduced into the bit stream.

**Data from upper layer**

00011111100111101000



**Frame sent**

Flag	Header	00011111011001111001000	Trailer	flag
------	--------	-------------------------	---------	------

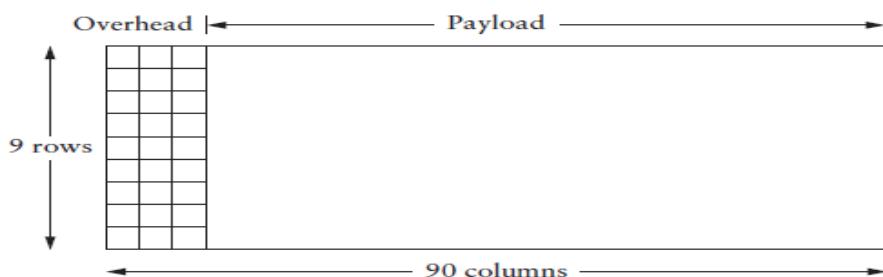
## After receiving 5 1s

- next bit
  - 0 >> **stuffed bit** >> removed
    - Example
      - bits received 0111 1101 010; bits retained (data): 0111 11010
      - bits received 0111 1100 010; bits retained (data): 0111 110010
  - 1 >> bits received 0111 1110 or 0111 1111
    - next bit
      - 0 **END of Frame** marker
      - 1 **Error**, frame is discarded; receiver waits for next 0111 1110 to start receiving next frame

## Clock based Protocols

### Clock-Based Framing (SONET):

- A third approach to framing is exemplified by the Synchronous Optical Network (SONET) standard.
- SONET addresses both the framing problem and the encoding problem.
- It also addresses a problem that is very important for phone companies—the multiplexing of several low-speed links onto one high-speed link.
- No bit stuffing is used, so that a frame's length does not depend on the data being sent.
- A STS-1 (lowest-speed SONET link runs at 51.84 Mbps.) frame is shown in Figure 2.13.
- It is arranged as nine rows of 90 bytes each.
- The first 3 bytes of each row are overhead, with the rest being available for data that is being transmitted over the link.
- The first 2 bytes of the frame contain a special bit pattern, and it is these bytes that enable the receiver to  
the receiver looks for the special bit pattern consistently, hoping to see it appearing once every 810 bytes, since each frame is  $9 \times 90 = 810$  bytes long.
- When the special pattern turns up in the right place enough times, the receiver concludes that it is in sync and can then interpret the frame correctly.



**Fig : A SONET STS-1 Frame**

For example, 64 Kbps of a SONET link's capacity is set aside for a voice channel that is used for maintenance.

The overhead bytes of a SONET frame are encoded using Non Return to Zero (NRZ), the simple encoding described in the previous section where 1s are high and 0s are low.

However, to ensure that there are plenty of transitions to allow the receiver to recover the sender's clock, the payload bytes are *scrambled*.

This is done by calculating the exclusive-OR (XOR) of the data to be transmitted and by the use of a well-known bit pattern. The bit pattern, which is 127 bits long, has plenty of transitions from 1 to 0, so that XORing it with the transmitted data is likely to yield a signal with enough transitions to enable clock recovery.

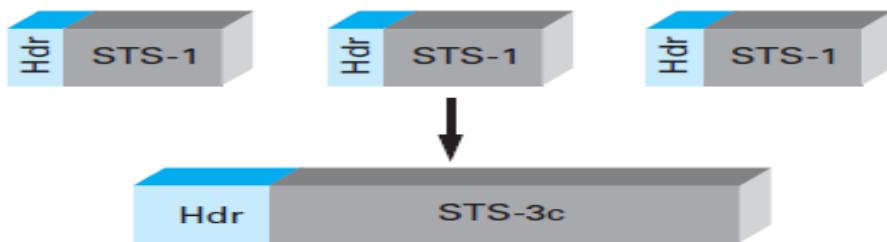


Fig : Three STS-1 Frames multiplexed onto one STS-3c frame.

## **ERROR DETECTION AND CORRECTION**

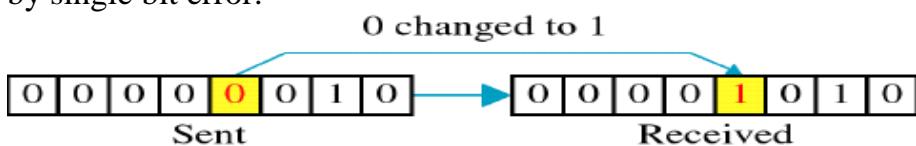
### **ERROR**

Data can be corrupted during transmission. Signals flows from one point to another. They are subjected to unpredictable interferences from heat, magnetism and other forms of electricity. For reliable communication, errors must be detected and corrected.

### **TYPES OF ERRORS**

- **Single bit Error:**

The term single bit error means that only one bit of a given data unit is changed from 1 to 0 or 0 to 1. 010101 is changed to 110101 here only one bit is changed by single bit error.

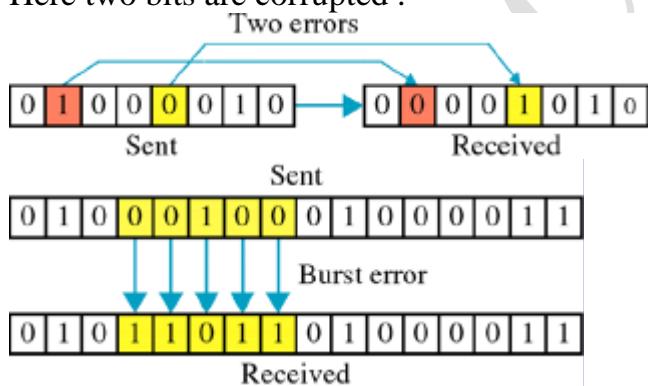


- **Burst Error:**

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

Example:

Here two bits are corrupted .



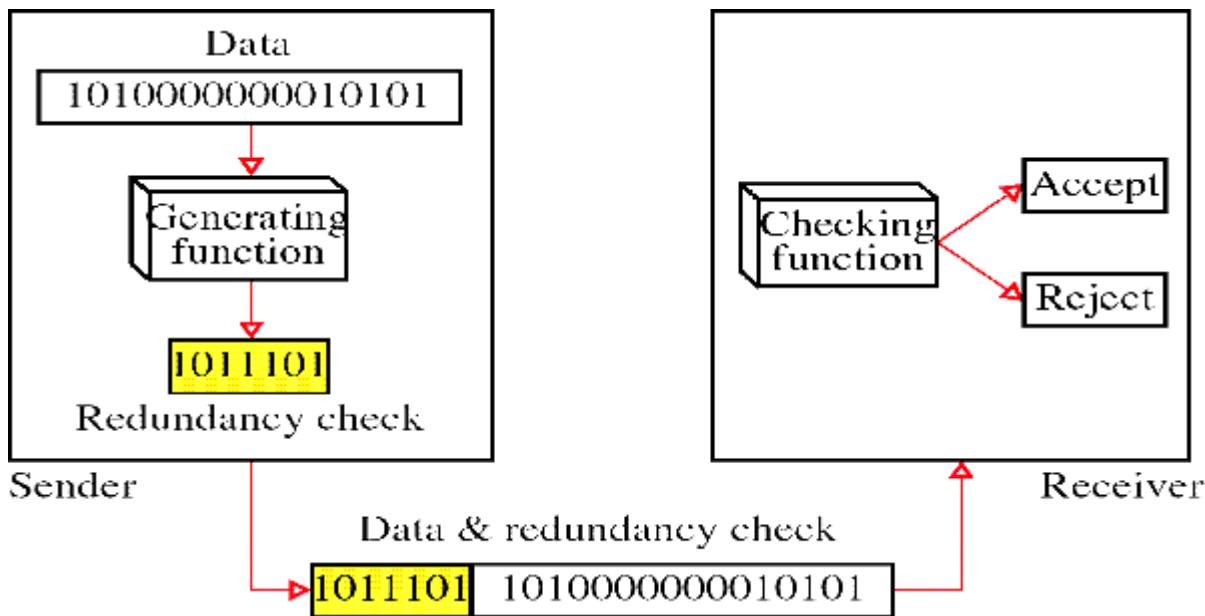
**Errors can happen in the following ways,**

- The bits in the frame can be inverted, anywhere within the frame including the data bits or the frame's control bits.
- Additional bits can be inserted into the frame, before the frame or after the frame and
- Bits can be deleted from the frame.

### **ERROR DETECTION**

#### **Redundancy**

Error detection use the concept of redundancy, which means adding extra bits for detecting errors .i.e., instead of repeating the entire data stream, a shorter group of bits may be appended to the end of each unit.



### Error Detection methods

1. **Parity check**
  - o *Simple parity check ( Vertical Redundancy Check)*
  - o Two dimensional Parity Check (*Longitudinal Redundancy Check*)
2. **Cyclic redundancy check**
3. **Checksum**

#### Parity check

A redundant bit called *parity bit*, is added to every data unit so that the total number of 1's in the unit becomes even (or odd).

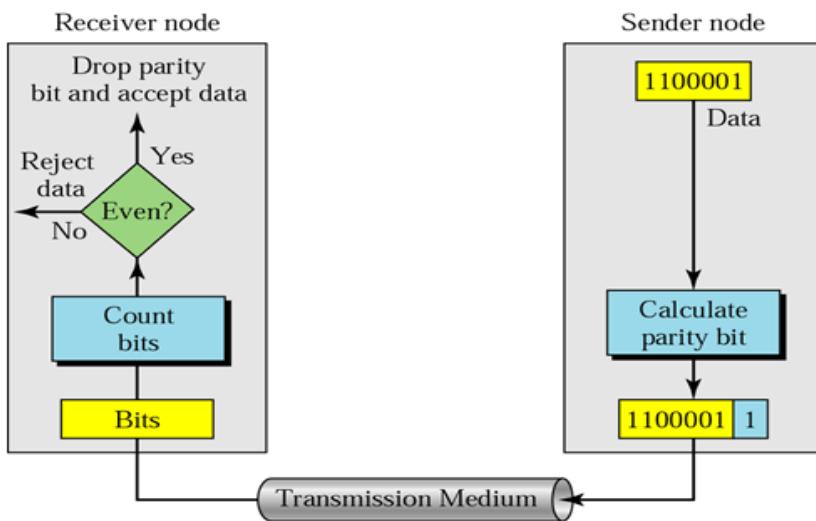
#### Simple parity check ( Vertical Redundancy Check)

##### **Sender Side:**

In a simple parity check a redundant bit is added to a stream of data so that total number of 1's in the data become even or odd.

##### **Receiver Side:**

The total data bit+ Redundant bit is then passed through parity checking function. For even parity, it checks for even number of 1's and for odd parity it checks even number of 1's. If the check fails, it implies error is detected the data is rejected otherwise accepted.

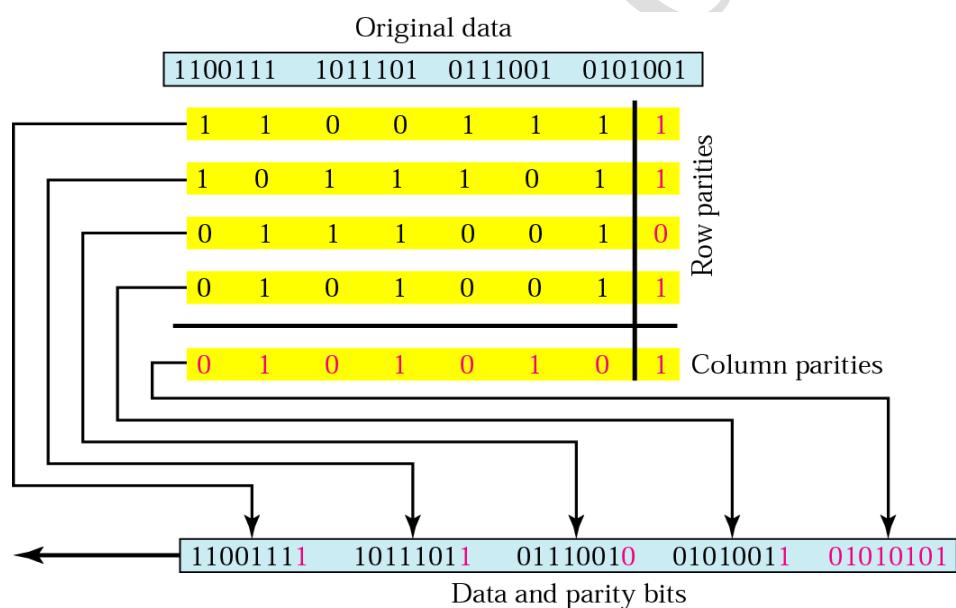


**Note :**

- Simple parity check can detect all single-bit errors. It can detect burst errors only if the total number of bits changed in each data unit is odd.

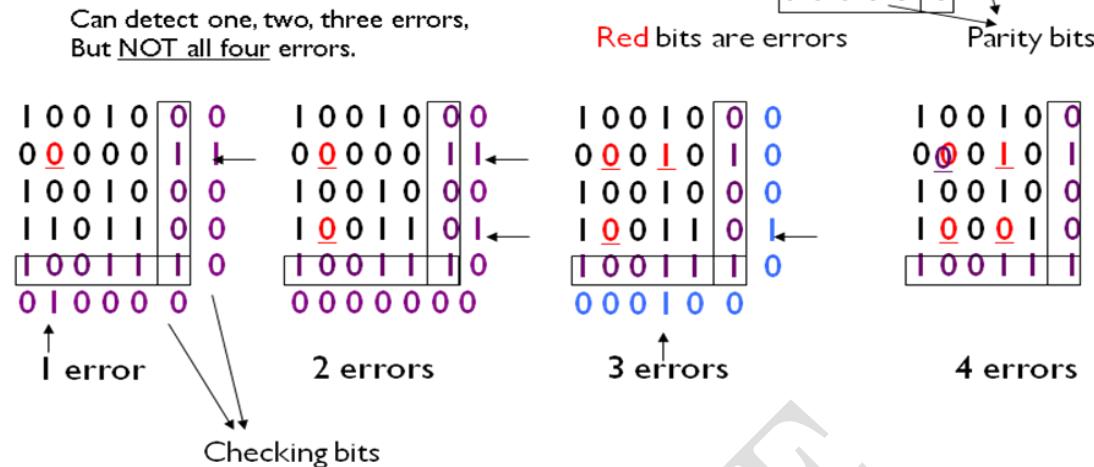
### Two dimensional Parity Check (Longitudinal Redundancy Check)

In two-dimensional parity check, a block of bits is divided into rows to form a 2-dimensional array; add single parity check bits to each row and each column; transmit row-by-row.



### Example 2

1. Data blocks are organized into table
2. Last column: check bits for rows
3. Last row: check bits for columns
4. Can detect and correct single bit error



### CYCLIC REDUNDANCY CHECK (CRC)

CRC is based on binary division. In CRC, a sequence of redundant bits, called the CRC or the CRC remainder is generated by performing binary division (using a predetermined divisor) and it is appended to the end of the data unit at the sender side.

At the receiver side, if the resulting unit (data + CRC) becomes exactly divisible by the same predetermined binary number (divisor), the data is accepted. If a remainder results it indicates that the data unit has been damaged in transit and therefore must be rejected.

### **STEP BY STEP PROCEDURE**

- First a starting of  $n$  0's is appended to the data unit. The number  $n$  is one less than the number of bits in the predetermined divisor, which is  $n+1$  bit.
- The newly elongated data unit is divided by the divisor, using a process called binary division. The remainder resulting from this division is the CRC.
- The CRC of  $n$  bits derived in step 2 replaces the appended 0s at the end of the data unit and the data is transmitted.(Note: CRC can also be all 0's)
- The data unit arrives at the receiver data first, followed by the CRC. The receiver treats the whole stream as unit and divides it by the same divisor that was used to find the CRC remainder.
- If the string arrives without error, the CRC checker yields a remainder of zero and the data unit passes. If the stream has been changed in transit, the division yields a non zero remainder and the data does not pass.

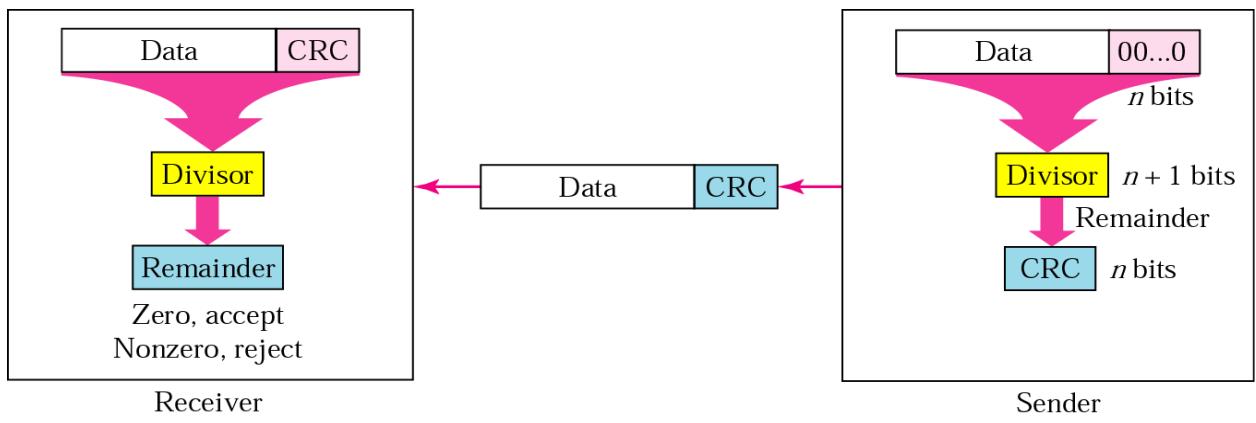


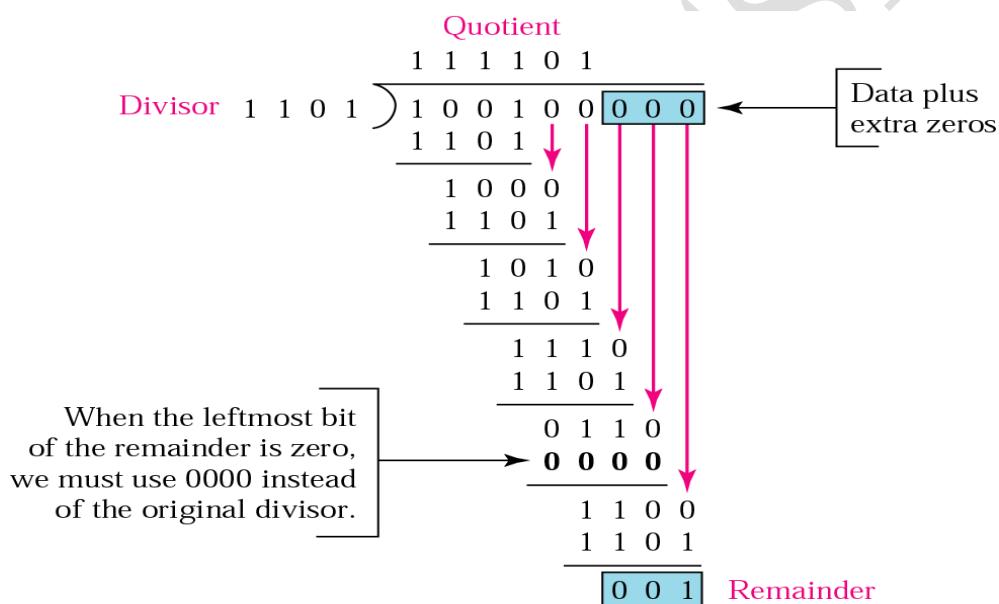
Figure : CRC generator and checker

## CRC GENERATOR AND CHECKER

Example: Data is 100100 and Divisor is 1101

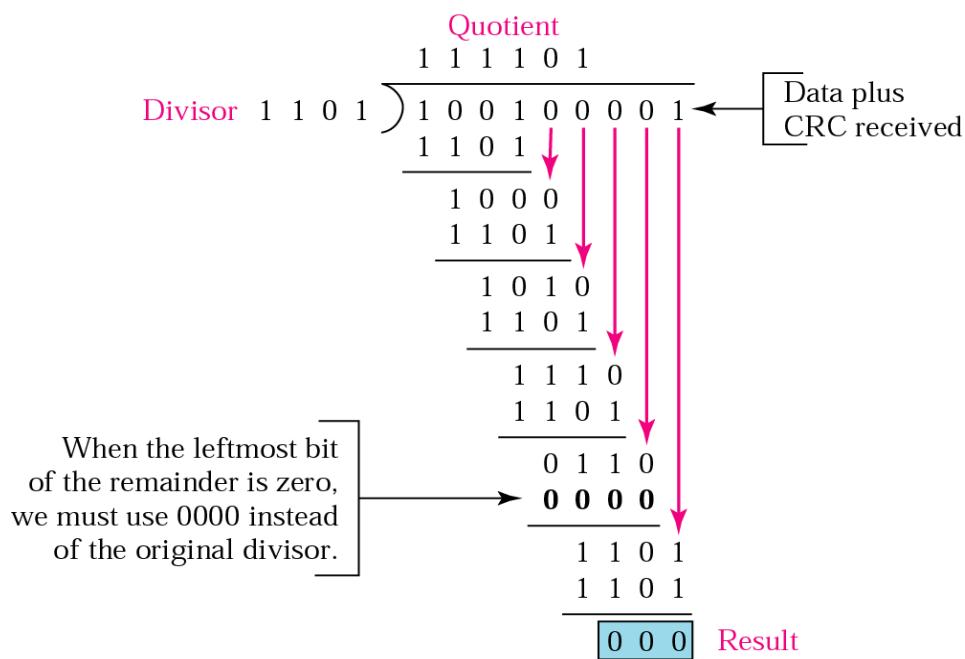
### CRC GENERATOR

It uses modulo-2 division. The following figure shows this process.



### CRC CHECKER

- A CRC checker function is exactly as the generator does. After receiving the data appended with the CRC, it does the same modulo-2 division.
- If the remainder is all 0s, the CRC is dropped and the data are accepted; otherwise, the received stream of bits is discarded and data are resent.
- The following figure shows the process of division in the receiver.

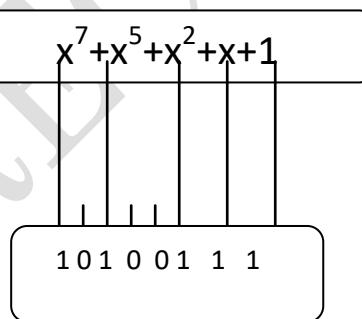


## POLYNOMIALS

The divisor in the CRC most often represented not as a string of 1s and 0s, but as an algebraic polynomial. The polynomial format is useful to solve the concept mathematically.

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

A polynomial



The polynomial generator should have following properties:

1. It should have at least two terms.
2. The coefficient of the term  $x^0$  should be 1.
3. It should not be divisible by  $x$ .
4. It should be divisible by  $x+1$ .

There are several different standard polynomials used by popular protocols for CRC generation.

Name	Polynomial	Application
CRC-8	$x^8 + x^2 + x + 1$	ATM header
CRC-10	$x^{10} + x^9 + x^5 + x^4 + x^2 + 1$	ATM AAL
CRC-16	$x^{16} + x^{12} + x^5 + 1$	HDLC
CRC-32	$x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1$	LANs

### CRC Performance

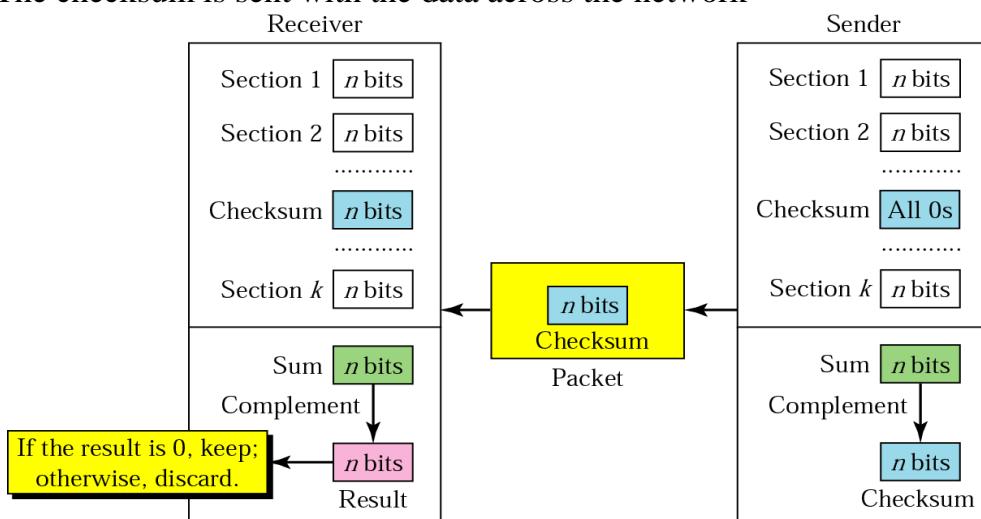
- All burst errors with length **less than or equal** to the degree of the polynomial ( $r$ )
- CRC can detect **all** burst errors that affect an **odd number** of bits
- CRC can detect **all** burst errors of length **equal** to the degree of the polynomial+1 ( $r+1$ ) with probability  $1 - (1/2)^{r-1}$ .
- CRC can detect burst errors of length **greater** than the degree of the polynomial+1 ( $r+1$ ) with probability  $1 - (1/2)^r$   
(where  $r$  is the degree of the CRC polynomial)

### CHECKSUM

The checksum is based on the redundancy.

#### CHECKSUM GENERATOR (sender side)

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



## CHECKSUM CHECKER (receiver side)

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

### EXAMPLE

Suppose the block of 16 bits is to be sent using a checksum of 8 bits.

10101001 00111001

the numbers are added using one's complement arithmetic

$$\begin{array}{r} 10101001 \quad \text{block 1} \\ 00111001 \quad \text{block 2} \\ \hline 11100010 \quad \text{initial checksum} \\ \hline 11100010 \quad \text{sum} \\ \hline 00011101 \quad \text{Checksum (1's complement value of sum)} \end{array}$$

The data sent is 10101001 00111001 00011101

Now the receiver receives the pattern with no error

10101001 00111001 00011101

the receiver adds these three sections, it will get all ones, which, after complementing, is all 0s and shows that there is no error.

$$\begin{array}{r} 10101001 \\ 00111001 \\ 00011101 \\ \hline 11111111 \quad \text{sum} \\ \hline 00000000 \quad \text{complement (DATA is correct)} \end{array}$$

suppose there is a burst error of length 5 that affects four bits.

10101111 1111001 00011101

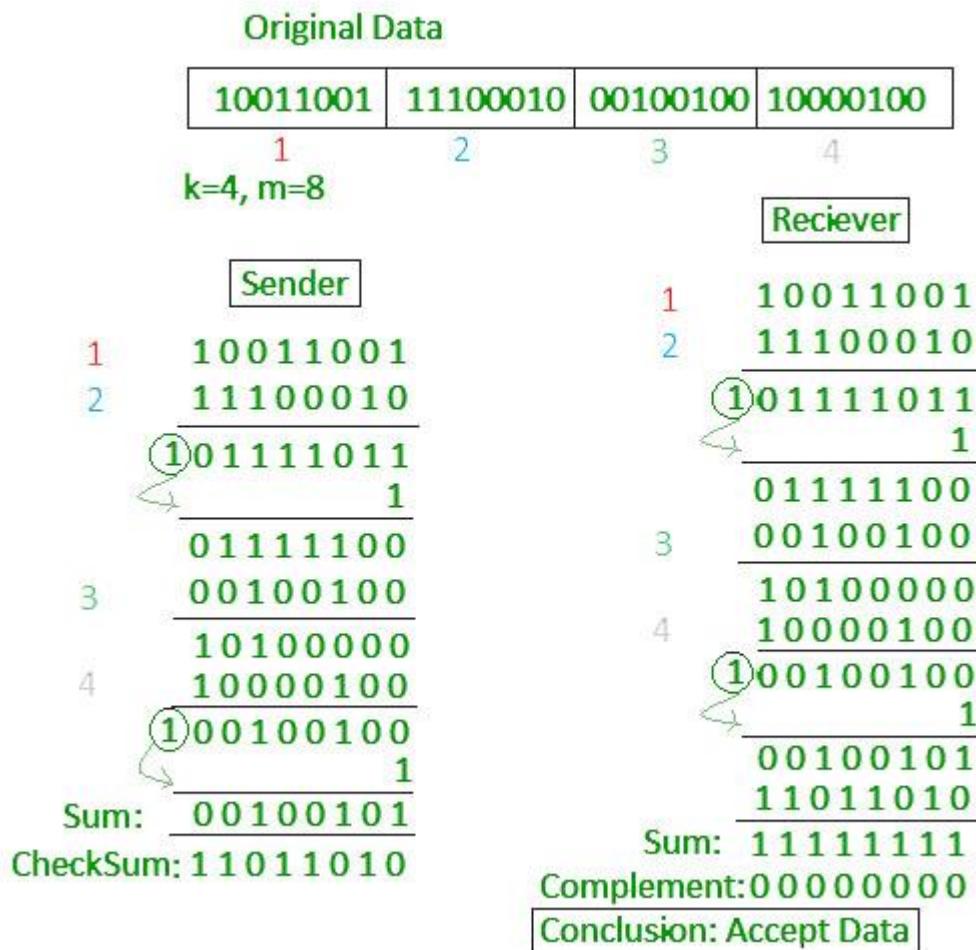
when the receiver adds these sections, it gets

$$\begin{array}{r} 10101111 \\ 11111001 \\ \hline (1)10101000 \\ \quad \quad \quad 1 \text{ (carry 1)} \\ \hline 10101001 \\ 00011101 \\ \hline 11000111 \text{ (carry 1)} \end{array}$$

1 1 0 0 0 1 0 1    sum  
 0 0 1 1 1 0 0 0    complement

As the complement is non zero we assume that (the data is errored)

### Example 2:



## PERFORMANCE

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

## Error correction

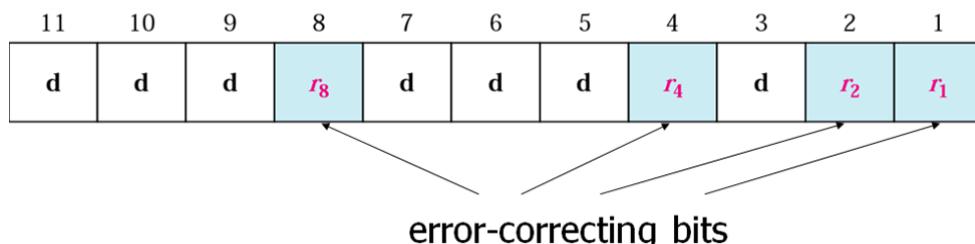
- Two methods
  1. Retransmission after detecting error
    - retransmit the data if error is detected
  2. Forward error correction (FEC)
    - hamming code is used to correct the error

## HAMMING CODE

- ✓ A *minimum number of redundancy bits* needed to correct any single bit error in the data
- ✓ A minimum of 4 redundancy bits is needed if the number of data bits is 4.
- ✓ Redundancy bits in the Hamming code are placed in the codeword bit positions that are a power of 2
- ✓ Each redundancy bit is the parity bit for a different combination of data bits
- ✓ Each data bit may be included in more than one parity check.
- ✓ But the r bits are also transmitted along with data; hence
- ✓  $2^r \geq k+r+1$

Number of Redundant Bits

Number of data bits <i>k</i>	Number of redundancy bits <i>r</i>	Total bits <i>k + r</i>
1	2	3
2	3	5
3	3	6
4	3	7
5	4	9
6	4	10
7	4	11



### Redundant Bit Calculation

$r_1$  will take care of these bits.

11	9	7	5	3	1					
d	d	d	$r_8$	d	d	d	$r_4$	d	$r_2$	$r_1$

$r_2$  will take care of these bits.

11	10	7	6	3	2					
d	d	d	$r_8$	d	d	d	$r_4$	d	$r_2$	$r_1$

$r_4$  will take care of these bits.

7	6	5	4							
d	d	d	$r_8$	d	d	d	$r_4$	d	$r_2$	$r_1$

$r_8$  will take care of these bits.

11	10	9	8							
d	d	d	$r_8$	d	d	d	$r_4$	d	$r_2$	$r_1$

For  $r_1$ ,

The bits considered are -1,3,5,7,9,11(to decide on these bits, follow this trick, since  $r_1$  bit, start from 1<sup>st</sup> bit, take one and leave one and so on)

For  $r_2$ ,

The bits considered are -2,3,6,7,10,11(to decide on these bits, follow this trick, since  $r_2$  bit, start from 2<sup>nd</sup> bit, take two and leave two and so on)

For  $r_4$ ,

The bits considered are -4,5,6,7(to decide on these bits, follow this trick, since  $r_4$  bit, start from 4<sup>th</sup> bit, take four and leave four and so on)

For  $r_8$ ,

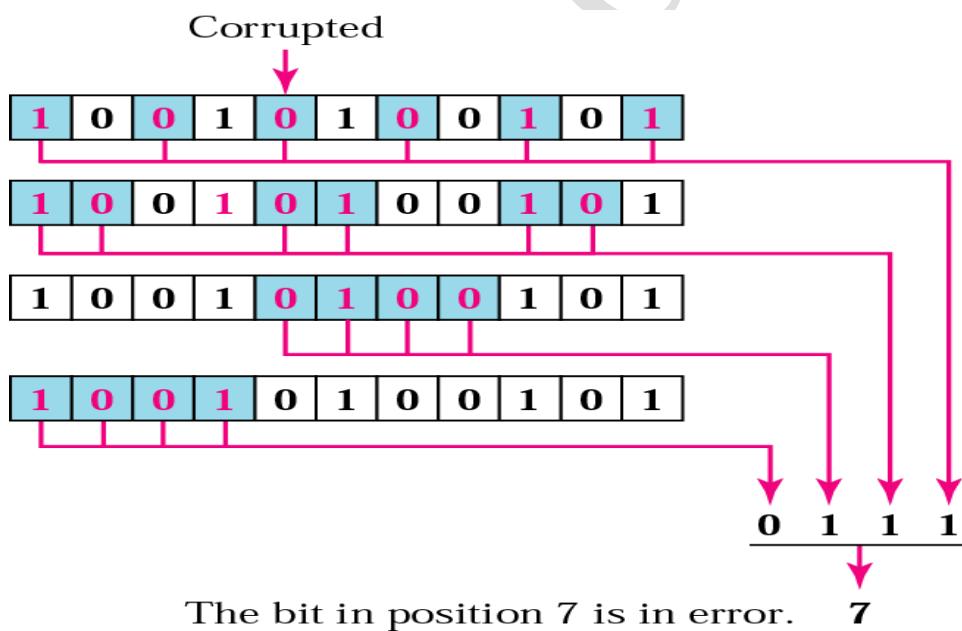
The bits considered are -8,9,10,11(to decide on these bits, follow this trick, since  $r_8$  bit, start from 8<sup>th</sup> bit, take eight and leave eight and so on)

### Example: Hamming Code

<table border="1" style="border-collapse: collapse; width: 100%;"> <tr><td>1</td><td>0</td><td>0</td><td></td><td>1</td><td>1</td><td>0</td><td></td><td>1</td><td></td><td></td><td></td></tr> <tr><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td></td></tr> </table> Adding $r_1$ <table border="1" style="border-collapse: collapse; width: 100%;"> <tr><td>1</td><td>0</td><td>0</td><td></td><td>1</td><td>1</td><td>0</td><td></td><td>1</td><td></td><td></td><td>1</td></tr> <tr><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td></td></tr> </table> Adding $r_2$ <table border="1" style="border-collapse: collapse; width: 100%;"> <tr><td>1</td><td>0</td><td>0</td><td></td><td>1</td><td>1</td><td>0</td><td></td><td>1</td><td>0</td><td></td><td>1</td></tr> <tr><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td></td><td>1</td></tr> </table> Adding $r_4$ <table border="1" style="border-collapse: collapse; width: 100%;"> <tr><td>1</td><td>0</td><td>0</td><td></td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td></td><td>1</td></tr> <tr><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td></td><td>1</td></tr> </table> Adding $r_8$ <table border="1" style="border-collapse: collapse; width: 100%;"> <tr><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td></td></tr> <tr><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td></td></tr> </table>	1	0	0		1	1	0		1				11	10	9	8	7	6	5	4	3	2	1		1	0	0		1	1	0		1			1	11	10	9	8	7	6	5	4	3	2	1		1	0	0		1	1	0		1	0		1	11	10	9	8	7	6	5	4	3	2		1	1	0	0		1	1	0	0	1	0		1	11	10	9	8	7	6	5	4	3	2		1	1	0	0	1	1	1	0	0	1	0	1		11	10	9	8	7	6	5	4	3	2	1		Data: <b>1 0 0 1 1 0 1</b>
1	0	0		1	1	0		1																																																																																																																	
11	10	9	8	7	6	5	4	3	2	1																																																																																																															
1	0	0		1	1	0		1			1																																																																																																														
11	10	9	8	7	6	5	4	3	2	1																																																																																																															
1	0	0		1	1	0		1	0		1																																																																																																														
11	10	9	8	7	6	5	4	3	2		1																																																																																																														
1	0	0		1	1	0	0	1	0		1																																																																																																														
11	10	9	8	7	6	5	4	3	2		1																																																																																																														
1	0	0	1	1	1	0	0	1	0	1																																																																																																															
11	10	9	8	7	6	5	4	3	2	1																																																																																																															

### Example: Correcting Error

- Receiver receives **10010100101**



## **FLOW CONTROL (RELIABLE TRANSMISSION)**

Flow control is a technique used for the following reasons,

- If sender sends frames faster than recipient processes, then buffer overflow occurs at the receiver side.
- Receiver needs some time to process incoming frames.

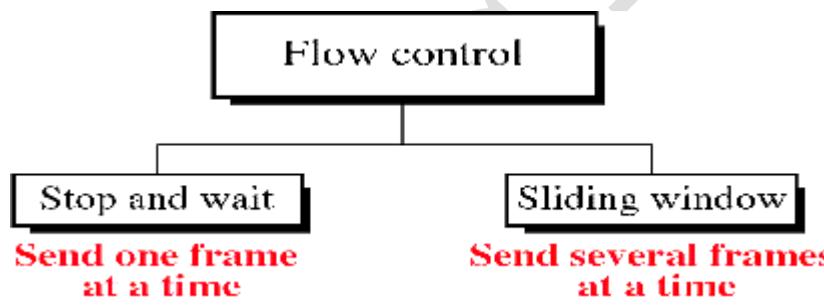
Flow control coordinates the amount of data that can be sent before receiving acknowledgement from the receiver. It is one of the most important duties of the data link layer.

- ***Flow control Refers to a set of procedures used to restrict the amount of data that the sender can send before waiting for acknowledgment from the receiver.***

### ACK

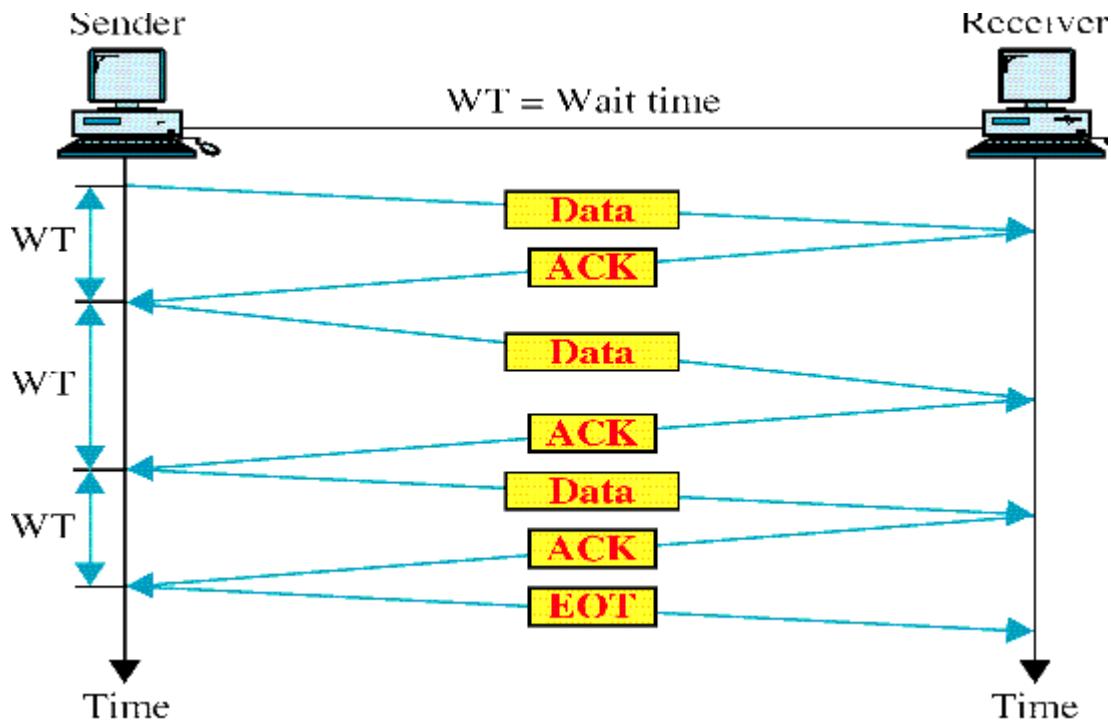
- An *acknowledgement* (ACK for short) is a small control frame that a protocol sends back to its peer saying that it has received the earlier frame.
  - A control frame is a frame with header only (no data).
  - The receipt of an *acknowledgement* indicates to the sender of the original frame that its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

## MECHANISMS



### 1. Stop and Wait Protocol

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



*Timeline showing 4 different scenarios for the stop-and-wait algorithm*

- The ACK is received before the timer expires;*
- the original frame is lost;*
- the ACK is lost;*
- the timeout fires too soon*

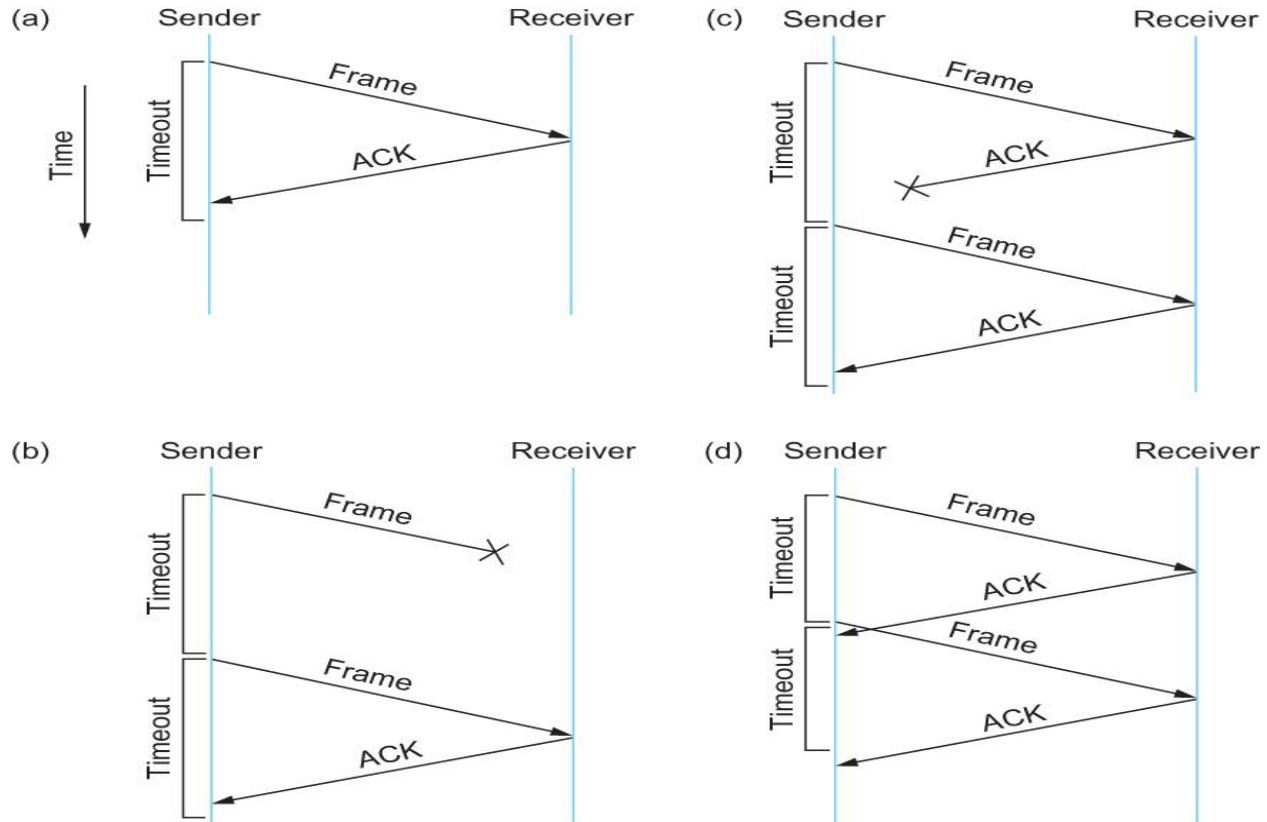
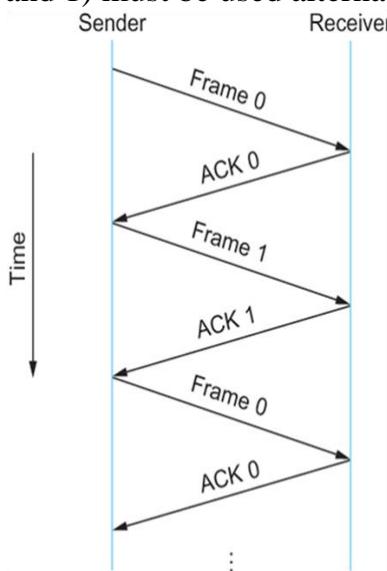


Fig: illustrates four different scenarios that result from the basic algorithm. The sending side is represented on the left, the receiving side is depicted on the right, and time flows from top to bottom.

- ✓ In Fig (a) ACK is received before the timer expires, (b) and (c) show the situation in which the original frame and the ACK, respectively, are lost, and (d) shows the situation in which the timeout fires too soon..
- ✓ Suppose the sender sends a frame and the receiver acknowledges it, but the acknowledgment is either lost or delayed in arriving. This situation is in (c) and (d). In both cases, the sender times out and retransmit the original frame, but the receiver will think that it is the next frame, since it correctly received and acknowledged the first frame.
- ✓ This makes the receiver to receive the duplicate copies. To avoid this two sequence numbers (0 and 1) must be used alternatively.



Timeline for stop-and-wait with 1-bit sequence number

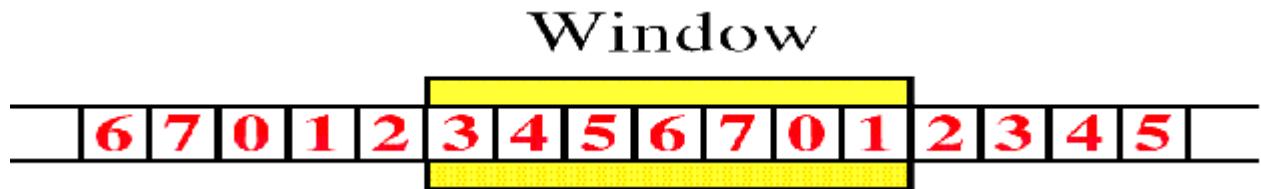
The main drawback of the stop-and-wait algorithm is that it allows the sender have only one outstanding frame on the link at a time.

## 2. Sliding Window

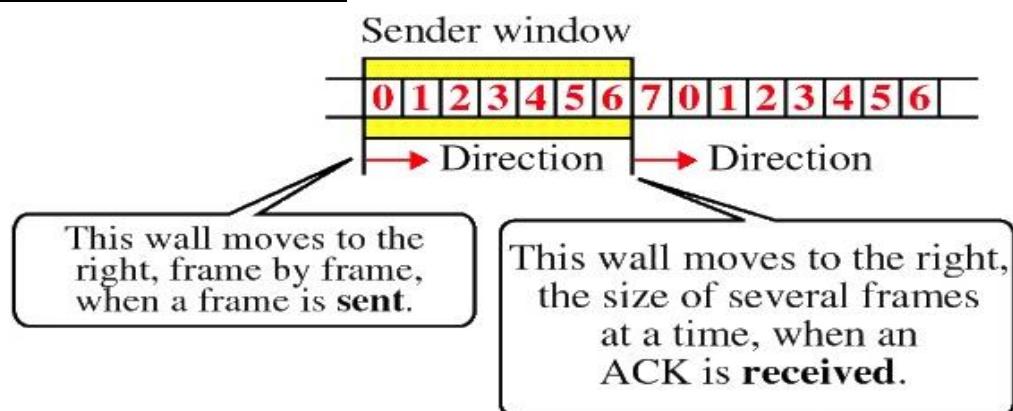
- ✓ The sender can transmit several frames before needing an acknowledgement.
- ✓ Frames can be sent one right after another meaning that the link can carry several frames at once and its capacity can be used efficiently.
- ✓ The receiver acknowledges only some of the frames, using a single ACK to confirm the receipt of multiple data frames
- ✓ Sliding Window refers to imaginary boxes at both the sender and the receiver.
- ✓ Window can hold frames at either end and provides the upper limit on the number of frames that can be transmitted before requiring an acknowledgement.
- ✓ Frames are numbered modulo-n which means they are numbered from 0 to n-1
- ✓ For eg. If n=8 the frames are numbered 0,1,2,3,4,5,6,7. i.e the size of the window is n -1.
- ✓ When the receiver sends ACK it includes the number of the next frame it expects to receive.
  - ✓ When the sender sees an ACK with the number 5, it knows that all frames up

through number 4 have been received.

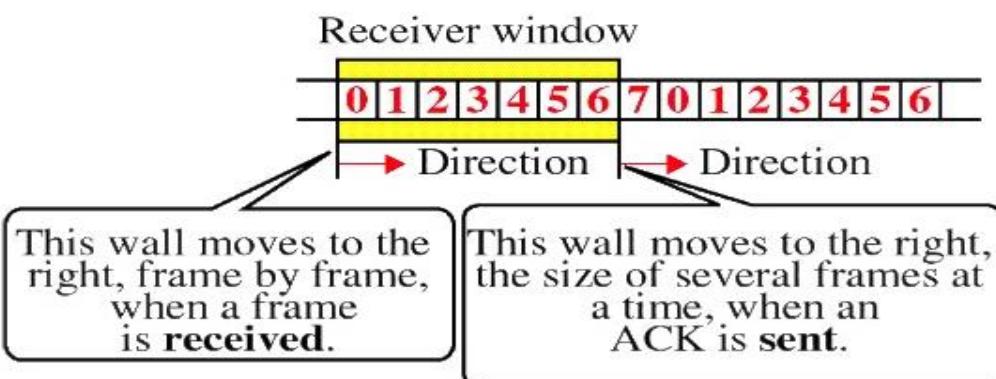
### Sliding Window



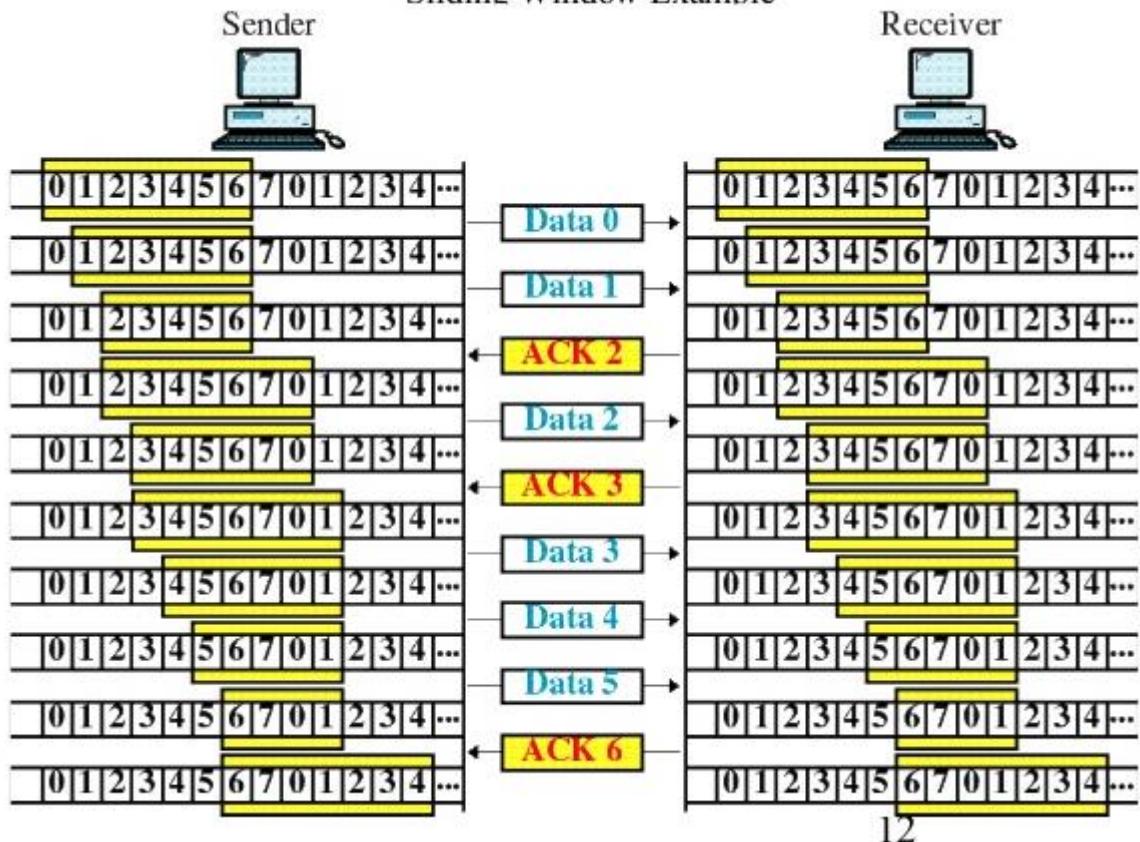
### Sender Sliding Window



### Receiver Sliding Window



### Sliding Window Example



## UNIT-II MEDIA ACCESS AND INTERNETWORKING

Media Access Protocols – ALOHA - CSMA/CA/CD –Ethernet – Wireless LANs - 802.11- Bluetooth - Switching and Forwarding - Bridges and LAN Switches – Basic Internetworking- IP Service Model – IP fragmentation - Global Addresses – ARP - DHCP – ICMP- Virtual Networks and Tunnels.

### **Media access control**

The data link layer is divided into two sub layers,

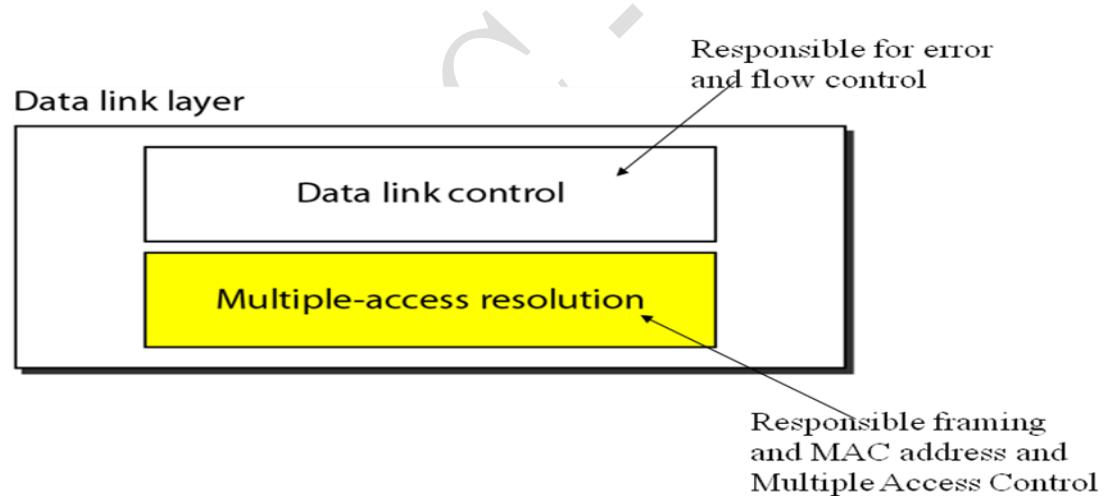
1. Logical link Control Layer
2. Medium Access Control Layer

#### **Logical Link Control Layer:**

The LCL layer is concerned with managing traffic (flow and error control) over the physical medium and may also assign sequence numbers to frames and track acknowledgements.

#### **Medium Access Control Layer:**

Media Access Control layer is one of two sublayers of the Data Link Control layer and is concerned with sharing the physical connection to the network among several computers.



#### **Medium Access Control:**

- **Problem:** When two or more nodes transmit at the same time, their frames will collide and the link bandwidth is **wasted** during collision  
How to coordinate the access of multiple sending/receiving nodes to the shared link???
- **Solution:** We need a **protocol** to coordinate the transmission of the active nodes

- These protocols are called **Medium or Multiple Access Control (MAC)** **Protocols** belong to a **sublayer** of the data link layer called **MAC** (Medium Access Control)
- What is expected from Medium Access Control Protocols:
  1. Main task is to **minimize collisions** in order to **utilize the bandwidth** by:
  2. Determining **when** a station can use the link (medium)
  3. **what** a station should do when the link is **busy**
  4. **what** the station should do when it is involved in **collision**
  5. To avoid the multiple access problem is used in which the station is forced to sense the medium before transmitting. This is called as Carrier Sense Multiple Access (CSMA).

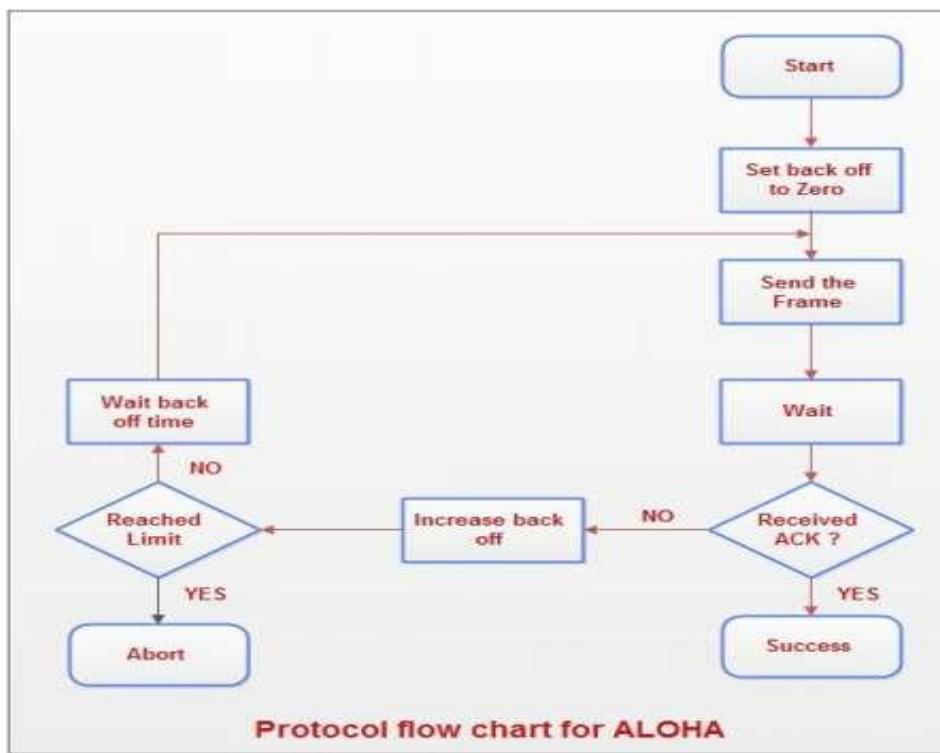
**The two commonly used medium access protocols are**

1. ALOHA
2. CSMA
3. CSMA/Collision Detection (CSMA/CD)
4. CSMA/Collision Avoidance(CSMA/CA)

**ALOHA**

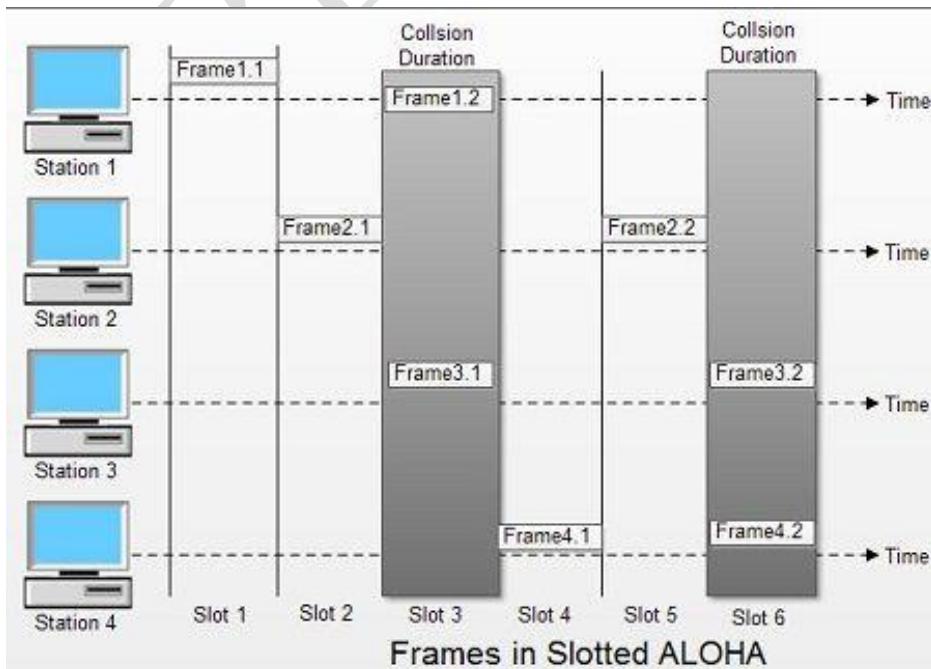
**PURE ALOHA**

- Each source (transmitter) in a network sends data whenever there is a frame to send.
- If the frame successfully reaches the destination (receiver), the next frame is sent.
- When two or more stations transmit simultaneously, there is collision and the frames are destroyed.
- If acknowledgement is not received within specified time, the station assumes that the frame (or acknowledgement) has been destroyed.
- If the frame is destroyed because of collision the station waits for a random amount of time and sends it again. This waiting time must be random otherwise same frames will collide again and again.
- Therefore, pure ALOHA dictates that when time-out period passes, each station must wait for a random amount of time before resending its frame. This randomness will help avoid more collisions.



## SLOTTED ALOHA

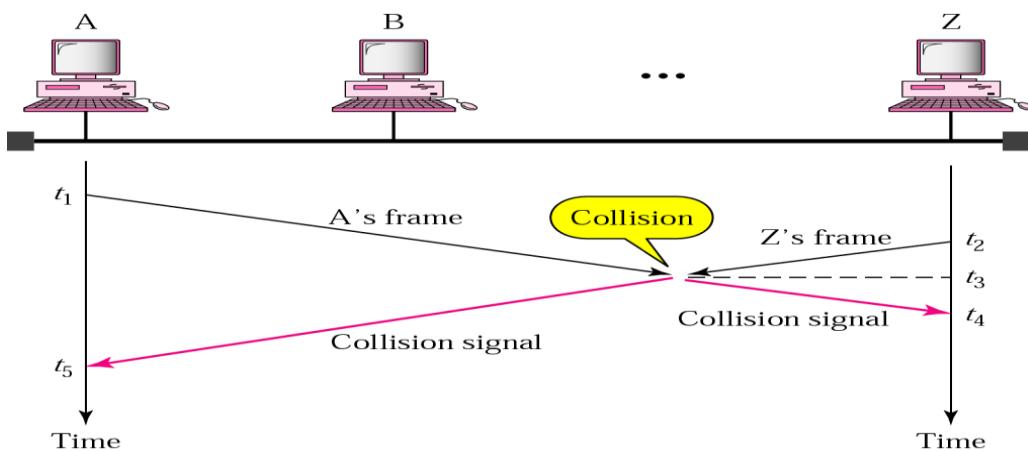
- The time of the shared channel is divided into discrete intervals called slots
- The stations can send a frame only at the beginning of the slot and only one frame is sent in each slot.
- In slotted ALOHA, if any station is not able to place the frame onto the channel at the beginning of the slot *i.e.* it misses the time slot then the station has to wait until the beginning of the next time slot.
- In slotted ALOHA, there is still a possibility of collision if two stations try to send at the beginning of the same time slot



## CSMA (Carrier Sense Multiple Access)

CSMA is based on the principle “Sense Before Transmit or Listen Before Talk”. Each station must listen before transmitting.

- → If a frame was sent by a station, All stations knows immediately so they can **wait before start sending**
  - → A station with frames to be sent, should **sense the medium** for the presence of another transmission (carrier) before it starts its own transmission
- CSMA can **reduce** the possibility of collision but it cannot eliminate it.
  - Collision can only happen when more than one station begin transmitting within a short time (the **propagation time** period)

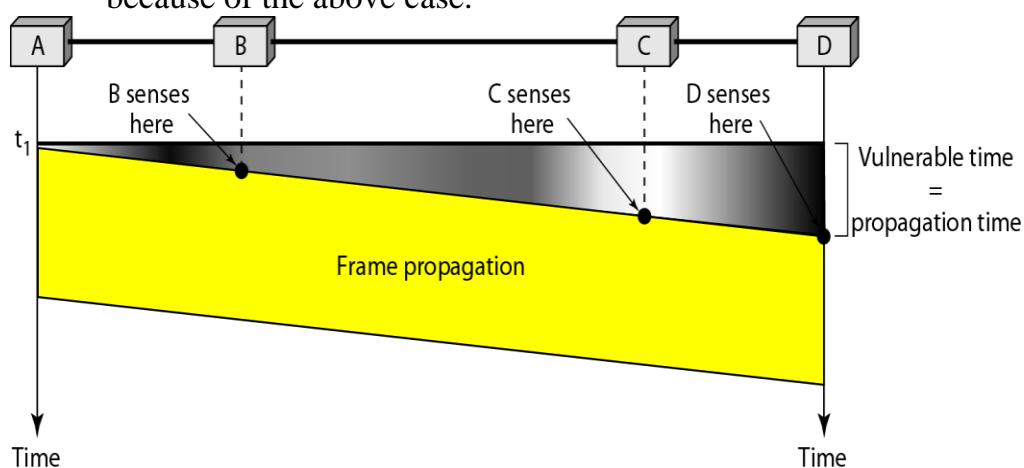


The possibility of collision still exists because of propagation delay; when a station sends a frame, it still takes time (although very short) for the first bit to reach every station and for every station to sense it. In other words, a station may sense the medium and find it idle, only because the first bit sent by another station has not yet been received.

### Vulnerable time

Vulnerable time for CSMA is the **maximum propagation time**

- The longer the propagation delay, the worse the performance of the protocol because of the above case.



- When a station sends a frame, and any other station tries to send a frame during this time, a collision will result.
- But if the first bit of the frame reaches the end of the medium, every station will already have heard the bit and will refrain from sending. Figure shows the worst case. The vulnerable time for CSMA is the propagation time  $tp$ .
- The leftmost station A sends a frame at time  $t1$  which reaches the rightmost station D at time  $t1 + tp$ . The gray area shows the vulnerable area in time and space.

### Types of CSMA Protocols

Different CSMA protocols that determine:

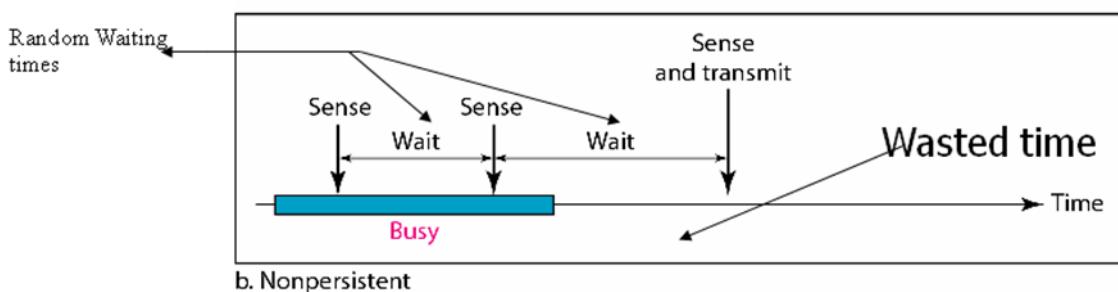
- What a station should do when the medium is **idle**?
- What a station should do when the medium is **busy**?

The channel can access the medium using any of three persistence methods,

1. Non-Persistent CSMA
2. 1-Persistent CSMA
3. p-Persistent CSMA

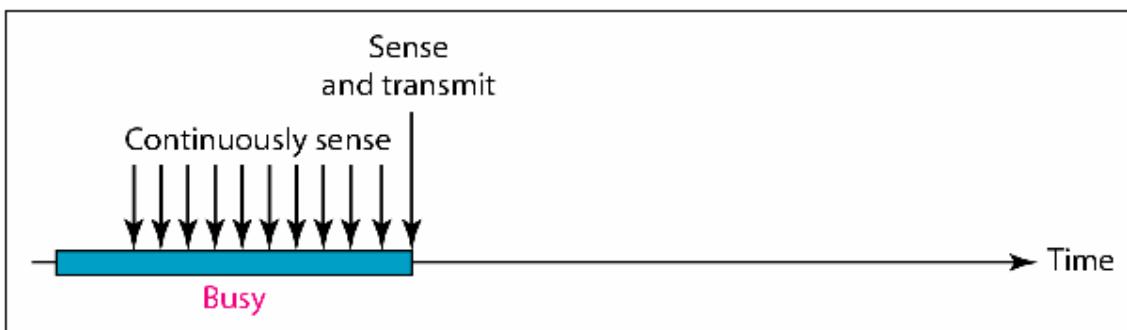
### No persistent CSMA

- A station with frames to be sent, should sense the medium
  1. If medium is idle, **transmit**; otherwise, go to 2
  2. If medium is busy, (**backoff**) wait a **random amount of time** and repeat 1
- Non-persistent Stations are **deferential (respect others)**
- Performance:
  1. Random delays reduces probability of collisions because two stations with data to be transmitted will wait for different amount of times.
  2. Bandwidth is **wasted** if waiting time (backoff) is large because medium will remain idle following end of transmission even if one or more stations have frames to send



## 1-persistent CSMA

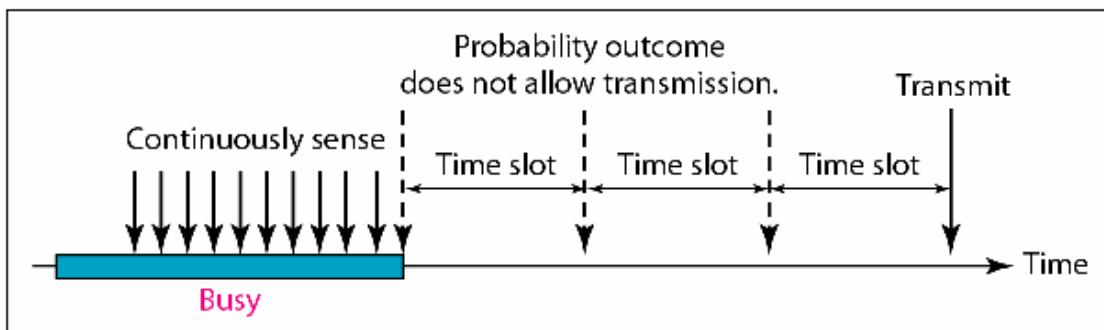
- To avoid idle channel time, 1-persistent protocol used
- Station wishing to transmit listens to the medium:
  1. If medium idle, **transmit** immediately;
  2. If medium busy, **continuously listen** until medium becomes idle; then transmit immediately with probability 1
- Performance
  - 1-persistent stations are **selfish**
  - If two or more stations becomes ready at the same time, **collision guaranteed**



a. 1-persistent

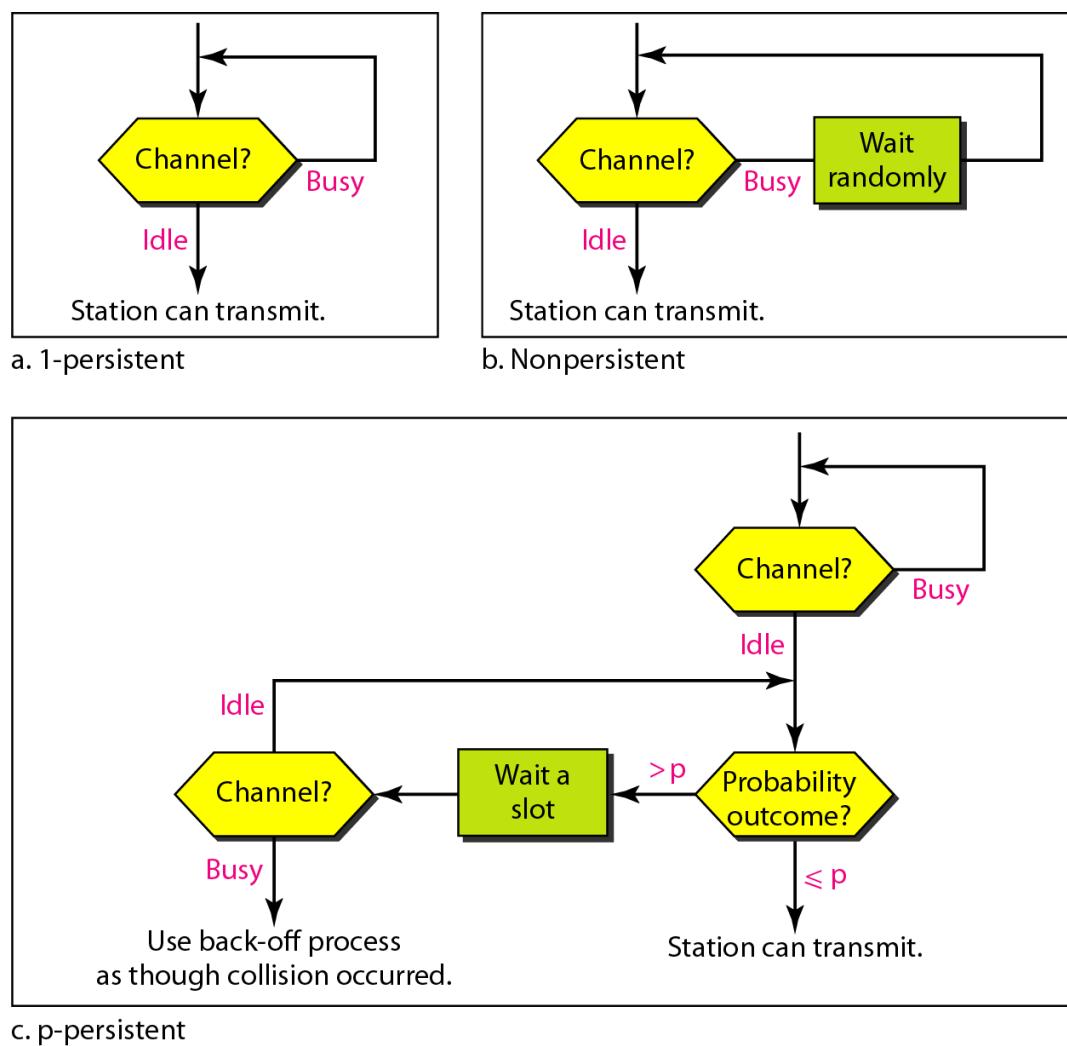
## P-persistent CSMA

- Time is divided to slots where each Time unit (slot) typically equals **maximum propagation delay**
- Station wishing to transmit listens to the medium:
  1. If medium idle,
    - transmit with probability ( $p$ ), OR
    - wait **one time unit (slot)** with probability ( $q = 1 - p$ ), then repeat 1.
  2. If medium busy, **continuously listen until idle** and repeat step 1
  3. Performance
    - Reduces the possibility of collisions like **nonpersistent**
    - Reduces channel idle time like **1-persistent**



c. p-persistent

### Flow diagram for three persistence methods

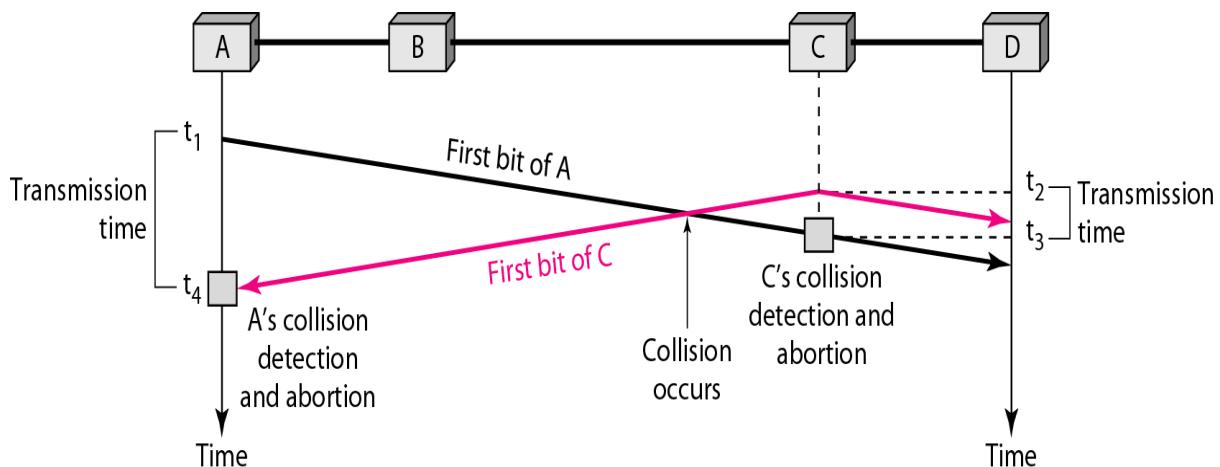


### CSMA/CD (CSMA - Collision Detection)

- **CSMA (all previous methods) has an inefficiency:**
  - If a collision has occurred, the channel is **unstable** until colliding packets have **been fully transmitted**
- **CSMA/CD (Carrier Sense Multiple Access with Collision Detection) overcomes this as follows:**
  - While transmitting, the sender is **listening to medium** for collisions.
  - Sender **stops transmission** if collision has occurred **reducing channel wastage**.

CSMA/CD is Widely used for bus topology LANs (IEEE 802.3, Ethernet).

### Collision of the first bit in CSMA/CD

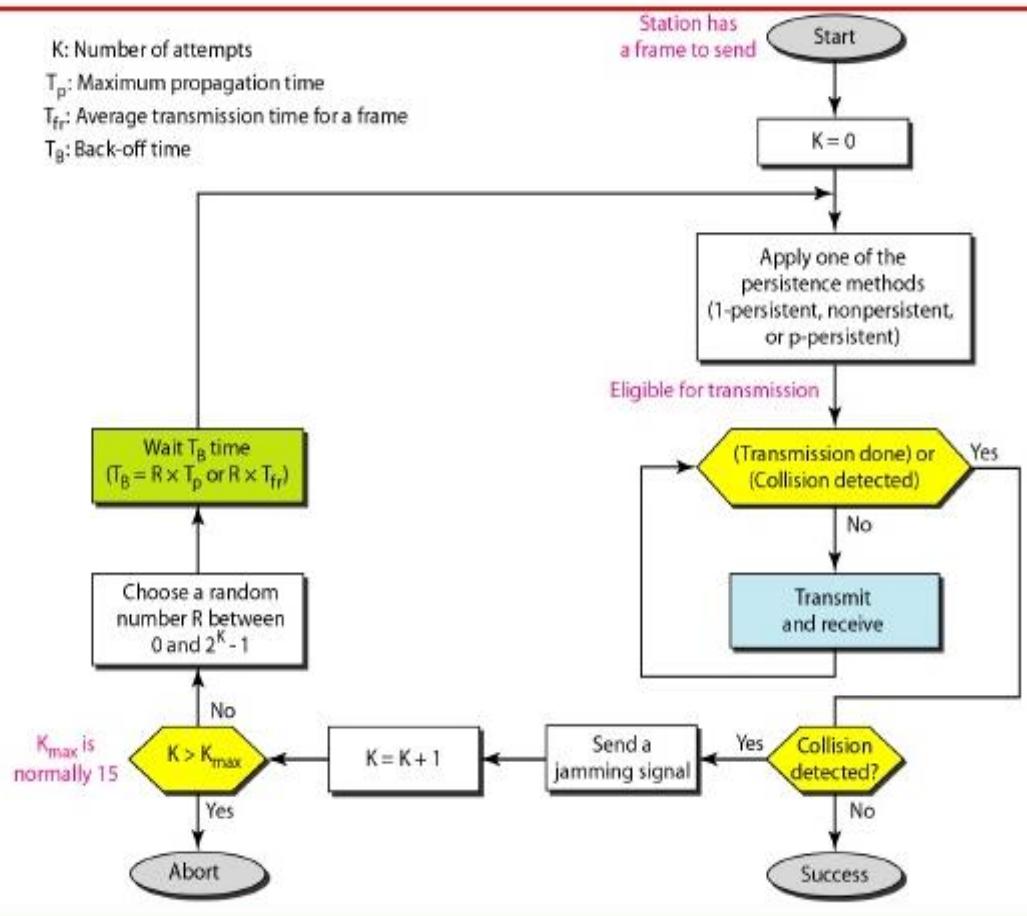


To better understand CSMA/CD, let us look at the first bits transmitted by the two stations involved in the collision. Although each station continues to send bits in the frame until it detects the collision, we show what happens as the first bits collide. In Figure , stations A and C are involved in the collision.

At time  $t_1$ , station A has executed its persistence procedure and starts sending the bits of its frame. At time  $t_2$ , station C has not yet sensed the first bit sent by A. Station C executes its persistence procedure and starts sending the bits in its frame, which propagate both to the left and to the right. The collision occurs sometime after time  $t_2$ . Station C detects a collision at time  $t_3$  when it receives the first bit of A's frame. Station C immediately (or after a short time, but we assume immediately) aborts transmission. Station A detects collision at time  $t_4$  when it receives the first bit of C's frame; it also immediately aborts transmission.

Looking at the figure, we see that A transmits for the duration  $t_4 - t_1$ ; C transmits for the duration  $t_3 - t_2$ . Later we show that, for the protocol to work, the length of any frame divided by the bit rate in this protocol must be more than either of these durations. At time  $t_4$ , the transmission of A's frame, though incomplete, is aborted; at time  $t_3$ , the transmission of C's frame, though incomplete, is aborted.

**Figure 12.14 Flow diagram for the CSMA/CD**

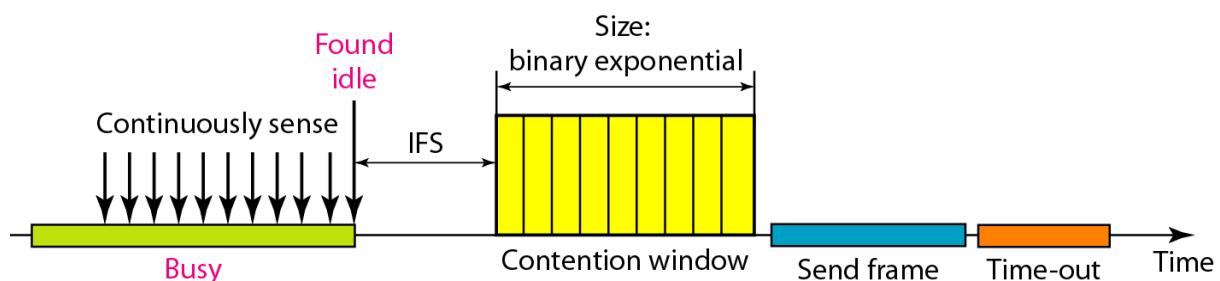


### CSMA/CA (CSMA – Collision Avoidance)

- Carrier Sense Multiple Access with Collision Avoidance
- Used in a network where collision cannot be detected
  - E.g., wireless LAN

Collisions are avoided in CSMA / CD using three strategies,

1. Inter Frame Space [IFS]
2. Contention window
3. Acknowledgements



## Inter Frame Sequence [IFS]

- Collisions are avoided by transmitting if a channel is found idle
- If the channel is found idle, the station does not send immediately
- It waits for a period of time called interframe space (IFS)

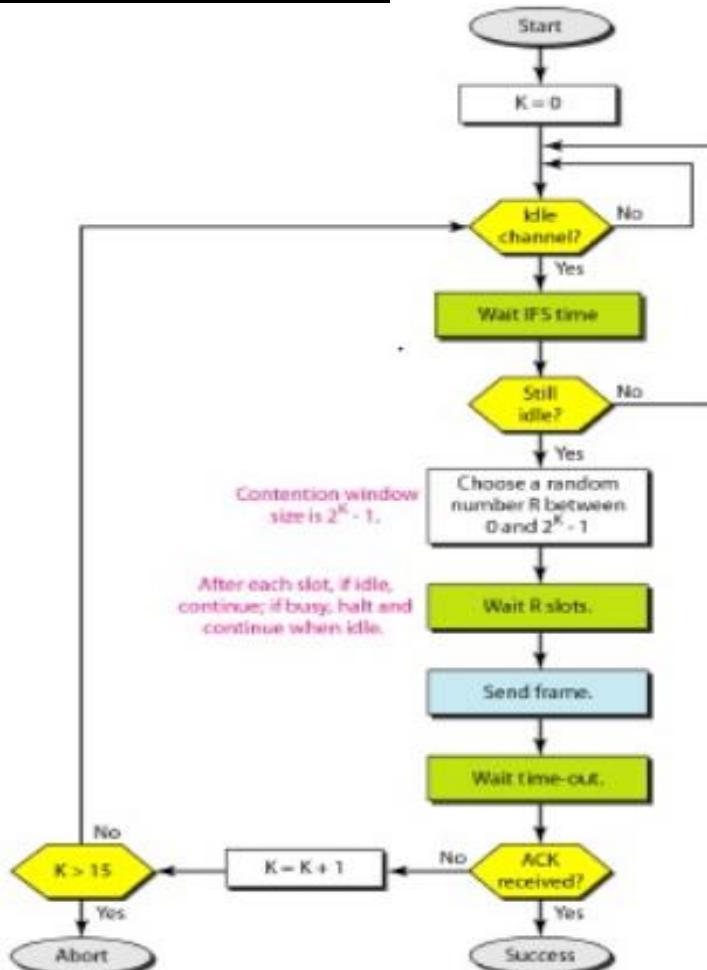
## Contention window

- contention window is an amount of time divided into slots.
- The station that is ready to send chooses a random number of slots as its wait time.
- The number of slots in the window changes according to binary exponential back off ie) it is set to one slot the first time and then doubles each time the station cannot detect an idle channel.
- In CSMA/CA, if the station finds the channel busy, it does not restart the timer of the contention window; it stops the timer and restarts it when the channel becomes idle

## Acknowledgements

- Collisions can be avoided by setting positive acknowledgements to assure that the receiver has received the frame or not.

## Flow Diagram for CSMA / CA:



## Ethernet (802.3)

### Wired LAN

- Most successful local area networking technology of last 20 years.
- Ethernet is a multiple access network with a set of nodes that send and receive frames over a shared link.

### Uses CSMA/CD technology

- Carrier Sense Multiple Access with Collision Detection.
- A set of nodes send and receive frames over a shared link.
- Carrier sense means that all nodes can distinguish between an idle and a busy link.
- Collision detection means that a node listens as it transmits and can therefore detect when a frame it is transmitting has collided with a frame transmitted by another node.

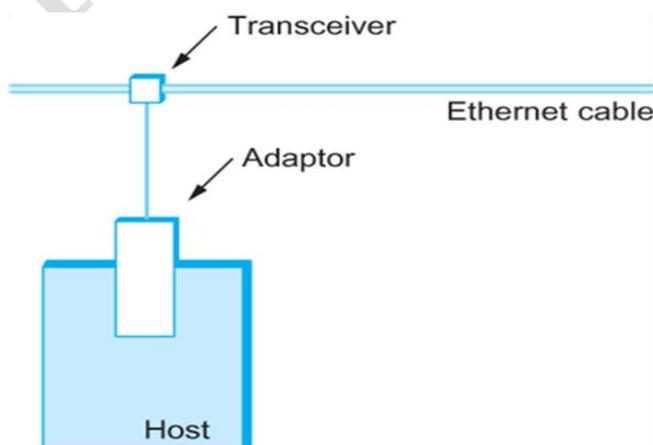
### Ethernet can operate at A SPEED

- 100 Mbps (Fast Ethernet)
- 1000 Mbps (Gigabit Ethernet)

Ethernet also used in full duplex, point-to-point configuration

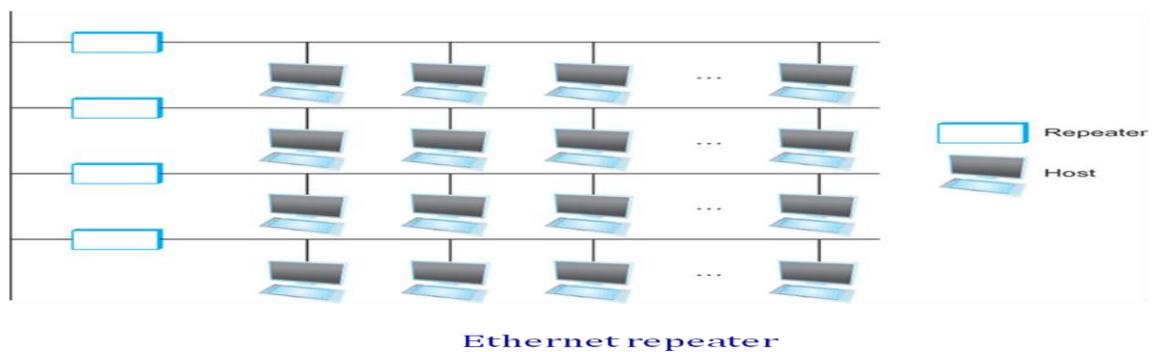
### Physical Properties

- An Ethernet segment is implemented on a coaxial cable of up to 500 m.
- Hosts connect to an Ethernet segment by tapping into it.
- A transceiver (a small device directly attached to the tap) detects when the line is idle and drives signal when the host is transmitting.
- The transceiver also receives incoming signal.
- The transceiver is connected to an Ethernet adaptor which is plugged into the host.
- The protocol is implemented on the adaptor.



**Ethernet transceiver and adaptor**

- Multiple Ethernet segments Can be joined together by repeaters.
- A *repeater* is a device that strengthens and forwards weakened digital signals.



Any signal placed on the Ethernet by a host is broadcast over the entire network

- Signal is propagated in both directions.
- Repeaters forward the signal on all outgoing segments.
- Terminators attached to the end of each segment absorb the signal.

**Table : Types of 10 Mbps Ethernets**

Name	Cable	Max. segment	Nodes /segment	Signaling technique	Topology used	Cable diameter	Access Method	advantage
10Base5	Thick coax	500 m	100	Baseband (Manchester)	Bus	10	CSMA/CD	Good For back bones
10Base2	Thin coax	200 m	30	Baseband (Manchester)	Bus	5	CSMA/CD	Cheapest system
10Base-T	Twisted pair	100 m	1024	Based band (Manchester)	Star	0.4 to 0.6	CSMA/CD	Easy maintenance
10Base-F	Fiber optics	2000m	1024	Manchester/ on-off	Star	62.5/125 μm	CSMA/CD	Best between buildings

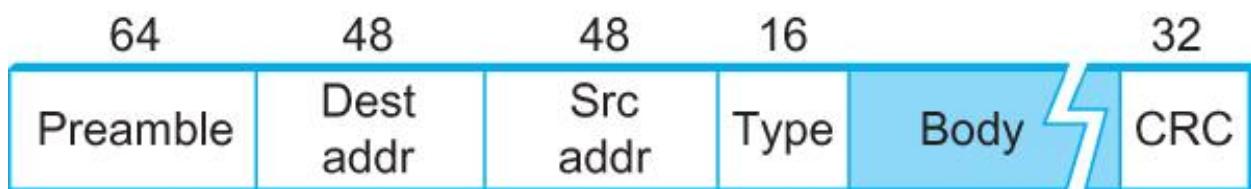
**10Base5 cable.** 10Base5 cable or "**Thick Ethernet**" or Thicknet is the cable which is the oldest in the category. it is called as thicknet because of the use of thick coaxial cable. The cable is marked after each 2.5 meters.

**10Base2 Cable.** 10Base2 cable also called "**Thin Ethernet**" or **Thinnet or cheapnet or cheapernet**, was designed after the thick Ethernet cable. This type of cable is usually thin, flexible and bends easily. It also make use of bus topology. It is also a coaxial cable that is having a smaller diameter than the 10Base5 cable.

**10BaseT Cable.** 10BaseT cable or "**Twisted Pair**" Cable is cheapest and easiest to maintain. This type of cabling is most popular among local area networks. It make use of unshielded twisted pair and provides maximum segment length of 100 m. It make use of start topology. In this type of network, every station is having a wired link to a central device, called "Hub".

**10BaseF Cable.** 10BaseF cable or "**Fiber Optics Cable**" is the most efficient and fastest cable in the category of cables for 802 LANs. The fiber optic cable is very expensive as compared to above discussed cables but it offers a very high data transmission speed and noise immunity. This type of cabling is preferred for running networks between buildings or widely separated hubs. It has the highest length per cable segment i.e. 2000 meters and it can support 1024 nodes per cable segment.

## Ethernet Frame Format



### **Preamble (64bit):**

- allows the receiver to synchronize with the signal .  
(sequence of alternating 0s and 1s).

**Host and Destination Address** physical addresses of the nodes(48bit each).

**Packet type** (16bit): - acts as demux key to identify the higher level protocol

### **Data (up to 1500 bytes)**

- Minimally a frame must contain at least 46 bytes of data.  
Frame must be long enough to detect collision.

**CRC (32bit)** - error detection

## Ethernet Addresses

- Each host on an Ethernet (in fact, every Ethernet host in the world) has a unique Ethernet Address.
- The address belongs to the adaptor, not the host.
  - It is usually burnt into ROM.
- Ethernet addresses are typically printed in a human readable format
  - As a sequence of six numbers separated by colons.
  - Each number corresponds to 1 byte of the 6 byte address and is given by a pair of hexadecimal digits, one for each of the 4-bit nibbles in the byte
  - For example, 8:0:2b:e4:b1:2 is
    - 00001000 00000000 00101011 11100100 10110001 00000010
- To summarize, an Ethernet adaptor receives all frames and accepts
  - Frames addressed to its own address
  - Frames addressed to the broadcast address
  - Frames addressed to a multicast address if it has been instructed

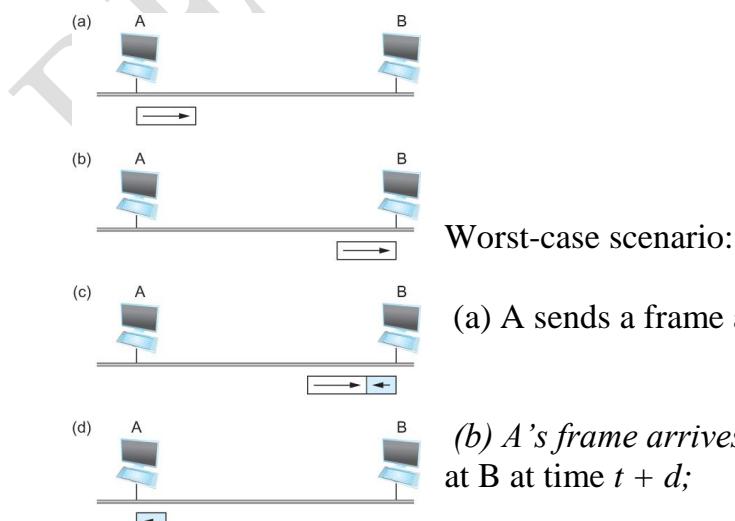
## Ethernet Transmitter Algorithm

- Algorithm is defined as follows, “When the adaptor has a frame to send and the line is idle, it transmits the frame immediately”
- When the adaptor has a frame to send and the line is busy, it waits for the line to go idle and then transmits immediately.
- The Ethernet is said to be 1-persistent protocol because an adaptor with a frame to send transmits with probability 1 whenever a busy line goes idle. When this happens, the two (or more) frames are said to be collide on the network.
- Since Ethernet supports collision detection, each sender is able to determine that a collision is in progress.

- At the moment an adaptor detects that its frame is colliding with another, it first makes sure to transmit a 32-bit jamming sequence and then stops transmission.
  - Thus, a transmitter will minimally send 96 bits in the case of collision 64-bit preamble + 32-bit jamming sequence
- One way that an adaptor will send only 96 bit (called a runt frame) is if the two hosts are close to each other.
  - Had they been farther apart, They would have had to transmit longer, and thus send more bits, before detecting the collision.
- The worst case scenario happens when the two hosts are at opposite ends of the Ethernet.
- To know for sure that the frame its just sent did not collide with another frame, the transmitter may need to send as many as 512 bits.
  - Every Ethernet frame must be at least 512 bits (64 bytes) long.
    - 14 bytes of header + 46 bytes of data + 4 bytes of CRC

### Ethernet Transmitter Algorithm

- A begins transmitting a frame at time  $t$
- $d$  denotes the one link latency
- The first bit of A's frame arrives at B at time  $t + d$
- Suppose an instant before host A's frame arrives, host B begins to transmit its own frame
- B's frame will immediately collide with A's frame and this collision will be detected by host B
- Host B will send the 32-bit jamming sequence
- Host A will not know that the collision occurred until B's frame reaches it, which will happen at  $t + 2 * d$
- Host A must continue to transmit until this time in order to detect the collision
- Host A must transmit for  $2 * d$  to be sure that it detects all possible collisions



(c) B begins transmitting at time  $t + d$  and collides with A's frame;

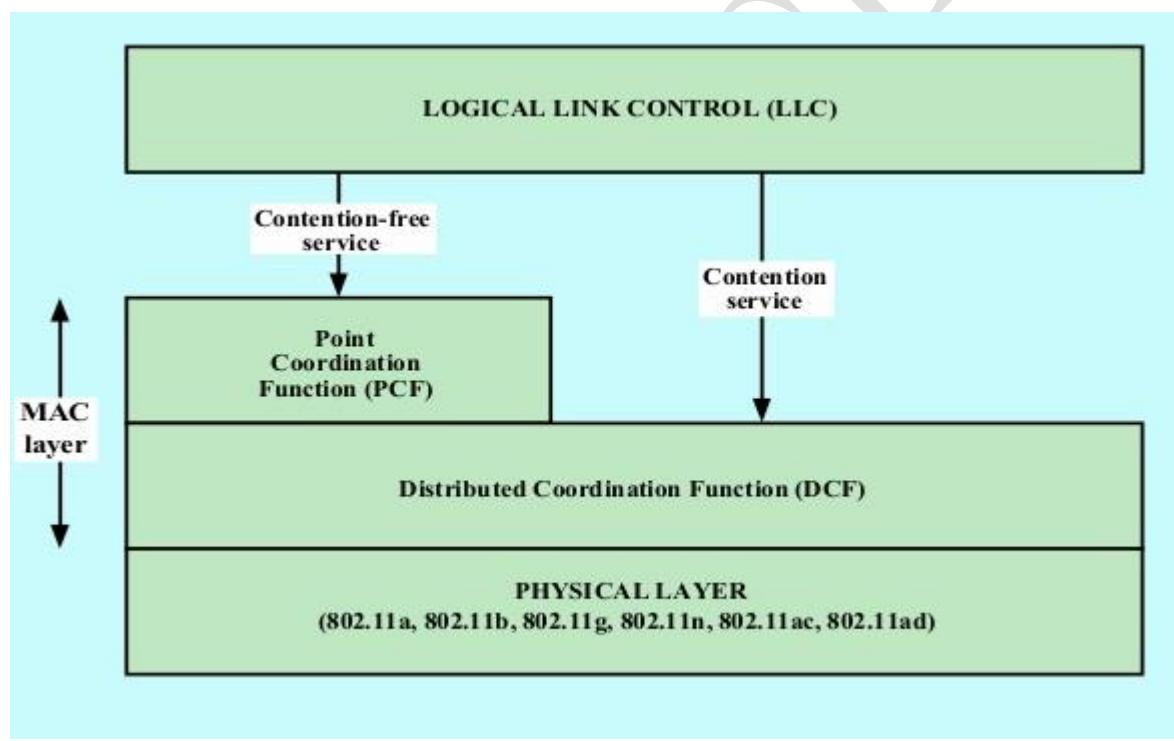
(d) B's runt (32-bit) frame arrives at A at time  $t + 2d$ .

- Once an adaptor has detected a collision, and stopped its transmission, it waits a certain amount of time and tries again.
- Each time the adaptor tries to transmit but fails, it doubles the amount of time it waits before trying again.
- This strategy of doubling the delay interval between each retransmission attempt is known as Exponential Backoff.

### **Wireless LAN (IEEE 802.11) / Wi-Fi**

- Wireless networking that is Standard IEEE 802.11 is a rapidly evolving technology for connecting computers. It is also known as Wi-Fi.
- Like Ethernet and token ring siblings, 802.11 is designed for use in a limited geographical area (homes, office buildings, campuses)

### **Protocol Architecture**

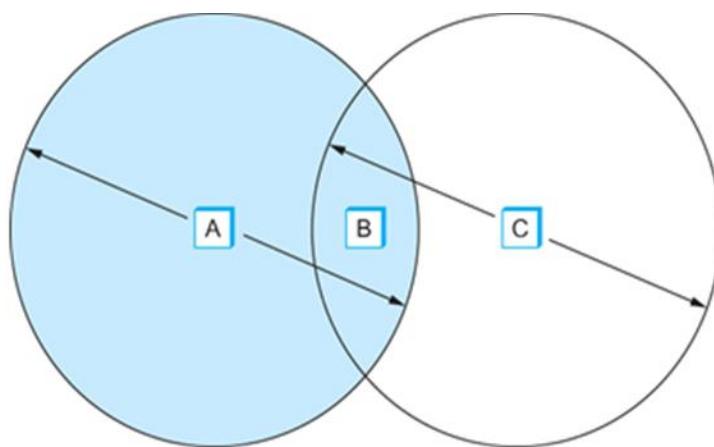


### **Physical Layer Properties:**

- 802.11 runs over six different physical layer protocols (so far). Five are based on spread spectrum radio, and one on diffused infrared. The fastest runs at a maximum of 54 Mbps.
- Original 802.11 standard defined two radio-based physical layer standard
  - One using the frequency hopping
    - Over 79 1-MHz-wide frequency bandwidths
  - Second using direct sequence
    - Using 11-bit chipping sequence
  - Both standards run in the 2.4-GHz and provide up to 2 Mbps

- Then physical layer standard 802.11b was added
  - Using a variant of direct sequence 802.11b provides up to 11 Mbps
  - Uses license-exempt 2.4-GHz band
- Then came 802.11a which delivers up to 54 Mbps using OFDM
  - 802.11a runs on license-exempt 5-GHz band
- Most recent standard is 802.11g which is backward compatible with 802.11b
  - Uses 2.4 GHz band, OFDM and delivers up to 54 Mbps

### Example of a wireless network:

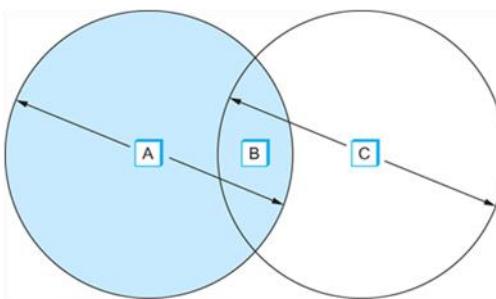


### IEEE 802.11 Multiple Access Collision Avoidance (MACA)

- Multiple Access with Collision Avoidance (MACA) is a protocol for slotted media access control used in wireless LAN data transmission.
- MACA is used to avoid data collisions caused by hidden station problems as well as simplifying known station problems.

### Hidden nodes problem

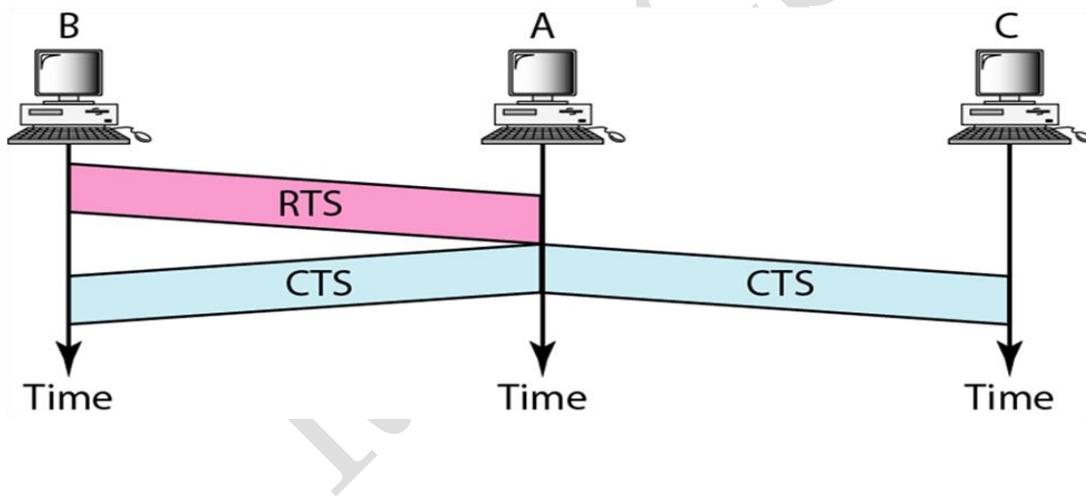
- Suppose both A and C want to communicate with B and so they each send it a frame.
  - A and C are unaware of each other since their signals do not carry that far
  - These two frames collide with each other at B
    - But unlike an Ethernet, neither A nor C is aware of this collision
  - A and C are said to hidden nodes with respect to each other



The “Hidden Node” Problem. Although A and C are hidden from each other, their signals can collide at B. (B’s reach is not shown.)

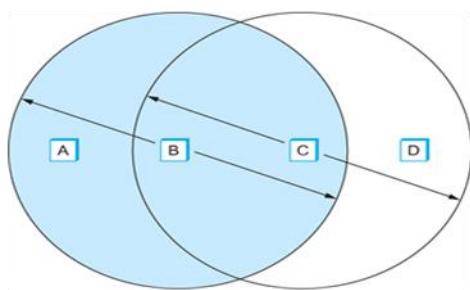
- **Key Idea-Solution**
  - Sender and receiver exchange control frames with each other before the sender actually transmits any data.
  - This exchange informs all nearby nodes that a transmission is about to begin
  - Sender transmits a Request to Send (RTS) frame to the receiver.
    - The RTS frame includes a field that indicates how long the sender wants to hold the medium
  - Length of the data frame to be transmitted
  - Receiver replies with a Clear to Send (CTS) frame
    - This frame echoes this length field back to the sender
- Any node that sees the CTS frame knows that
  - it is close to the receiver, therefore
  - cannot transmit for the period of time it takes to send a frame of the specified length
- Any node that sees the RTS frame but not the CTS frame
  - is not close enough to the receiver to interfere with it, and
  - so is free to transmit

*Use of handshaking to prevent hidden station problem*



### **Exposed node problem**

- Another problem called exposed node problem occurs
  - Suppose B is sending to A. Node C is aware of this communication because it hears B's transmission.
  - It would be a mistake for C to conclude that it cannot transmit to anyone just because it can hear B's transmission.
  - Suppose C wants to transmit to node D.
  - This is not a problem since C's transmission to D will not interfere with A's ability to receive from B.

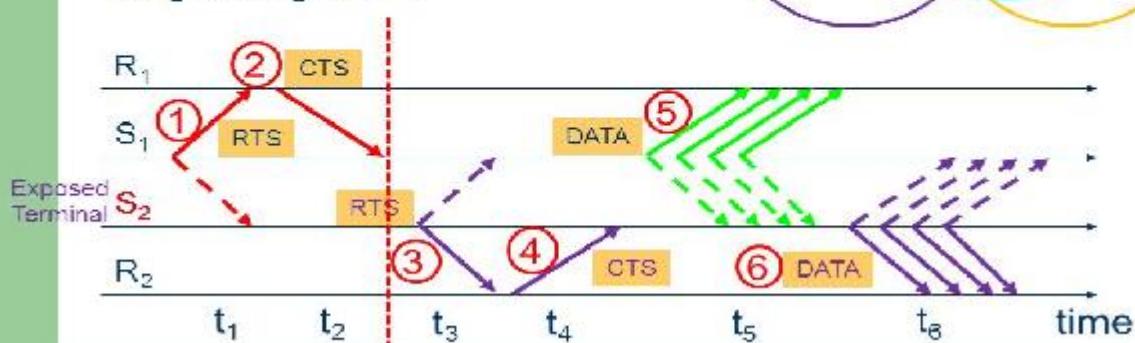
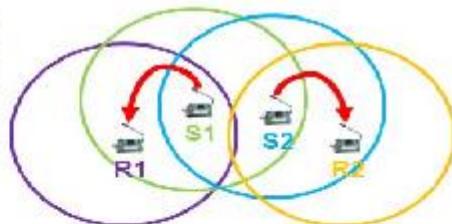


C->D progress  
If B->A not possible  
Since C is exposed to B

Although B and C are exposed to each other's signals, there is no interference if B transmits to A while C transmits to D. (A and D's reaches are not shown.)

### Solution for exposed node problem:

- When a node hears an RTS from a neighboring node, but not the corresponding CTS, that node can deduce that it is an *exposed terminal* and is permitted to transmit to other neighboring nodes.



### MACA: Medium Access Collision Avoidance:

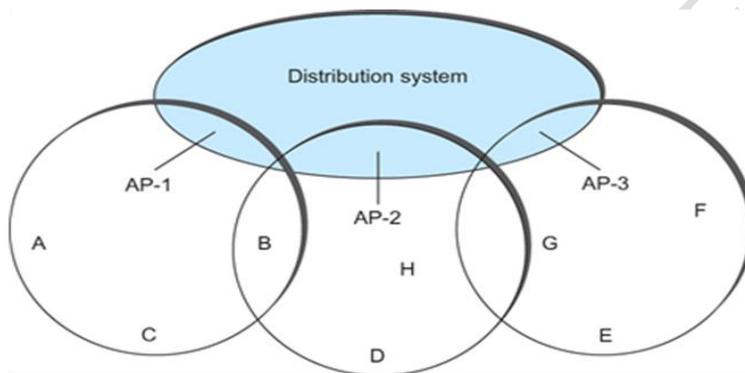
MACA is for Wireless LANs

- Receiver sends an ACK to the sender after successfully receiving a frame
- All nodes must wait for this ACK before trying to transmit
- If two or more nodes detect an idle link and try to transmit an RTS frame at the same time
  - Their RTS frame will collide with each other
- 802.11 does not support collision detection
  - So the senders realize the collision has happened when they do not receive the CTS frame after a period of time
  - In this case, they each wait a random amount of time before trying again.
  - The amount of time a given node delays is defined by the same *exponential backoff* algorithm used on the Ethernet.
- 802.11 is suitable for an ad-hoc configuration of nodes that may or may not be able to communicate with all other nodes.
- Nodes are free to move around

- The set of directly reachable nodes may change over time
- To deal with this mobility and partial connectivity,
  - 802.11 defines additional structures on a set of nodes
  - Instead of all nodes being created equal,
    - some nodes are allowed to roam
    - some are connected to a wired network infrastructure
- they are called *Access Points* (AP) and they are connected to each other by a so-called *distribution system*

### **IEEE 802.11 – Distribution System**

- Following figure illustrates a distribution system that connects three access points, each of which services the nodes in the same region
- Each of these regions is analogous to a cell in a cellular phone system with the APIs playing the same role as a base station
- The distribution network runs at layer 2 of the ISO architecture

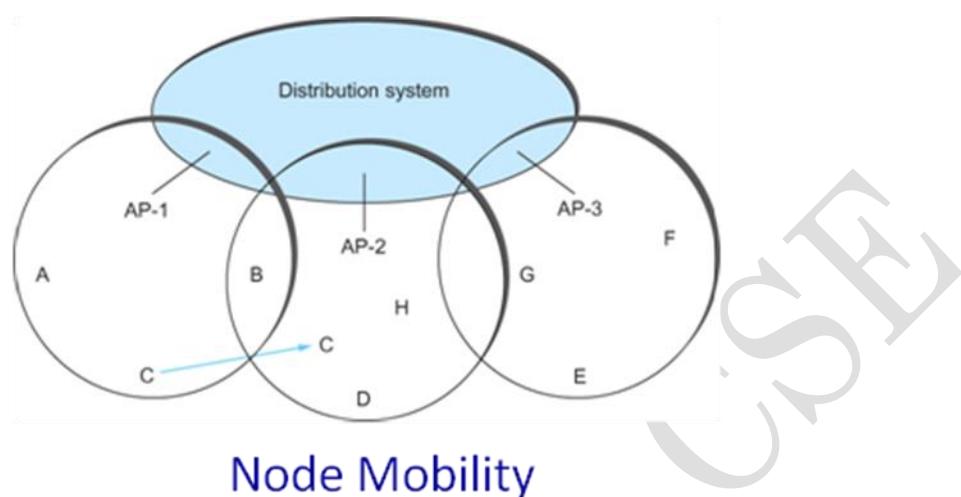


#### **Access points connected to a distribution network**

- Although two nodes can communicate directly with each other if they are within reach of each other, the idea behind this configuration is
  - Each node associates itself with one access point
  - For node A to communicate with node E, A first sends a frame to its AP-1 which forwards the frame across the distribution system to AP-3, which finally transmits the frame to E
- The technique for selecting an AP is called scanning
  - The node sends a *Probe* frame
  - All APs within reach reply with a *Probe Response* frame
  - The node selects one of the access points and sends that AP an *Association Request* frame
  - The AP replies with an *Association Response* frame
- A node engages this protocol whenever
  - it joins the network, as well as
  - when it becomes unhappy with its current AP
    - This might happen, for example, because the signal from its current AP has weakened due to the node moving away from it
    - Whenever a node acquires a new AP, the new AP notifies the old AP of the change via the distribution system

## Node Mobility

- Consider the situation shown in the following figure when node C moves from the cell serviced by AP-1 to the cell serviced by AP-2.
- As it moves, it sends *Probe* frames, which eventually result in *Probe Responses* from AP-2.
- At some point, C prefers AP-2 over AP-1 , and so it associates itself with that access point.
  - This is called ***active scanning*** since the node is actively searching for an access point



- APs also periodically send a *Beacon* frame that advertises the capabilities of the access point; these include the transmission rate supported by the AP
  - This is called ***passive scanning***
  - A node can change to this AP based on the *Beacon* frame simply by sending it an *Association Request* frame back to the access point.

## IEEE 802.11 – Frame Format

16	16	48	48	48	16	48	0–18,496	32
Control	Duration	Addr1	Addr2	Addr3	SeqCtrl	Addr4	Payload	CRC

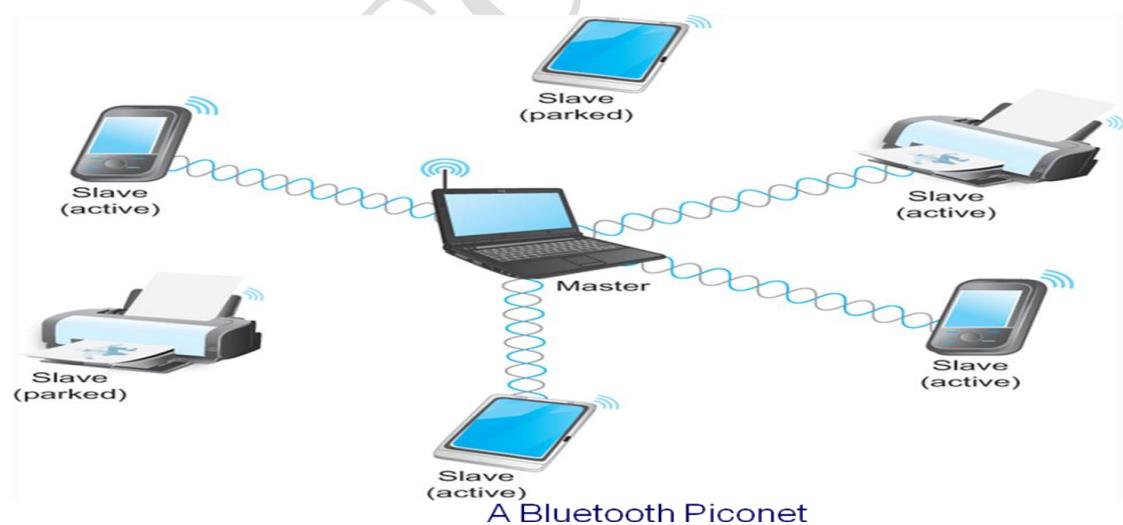
- Source and Destinations addresses: each 48 bits
- Data: up to 2312 bytes
- CRC: 32 bit
- Control field: 16 bits
  - Contains three subfields (of interest)
    - 6 bit **Type** field: indicates whether the frame is an RTS or CTS frame or being used by the scanning algorithm
    - A pair of 1 bit fields : called **ToDS** and **FromDS**
- Addr1 – target node
- Addr2 – immediate sender
- Addr3 – intermediate destination node
- Addr4 – original source

## **Bluetooth (802.15.1)**

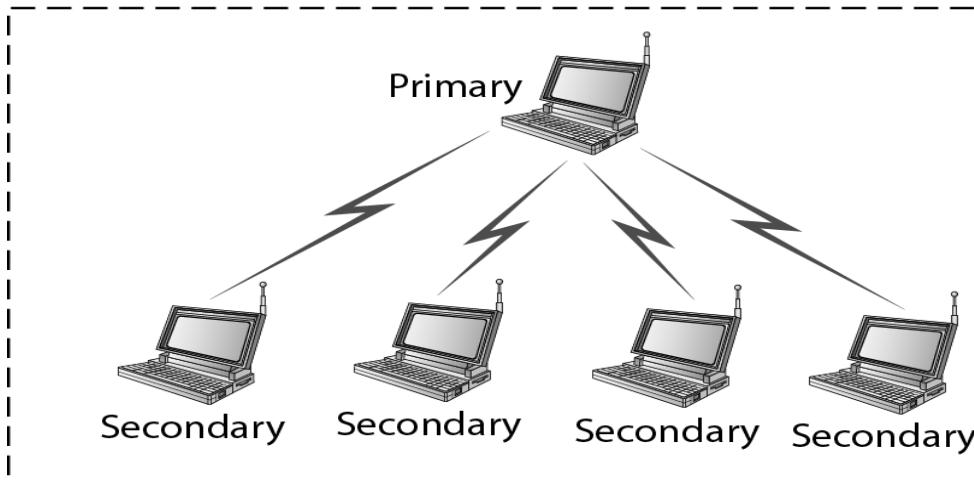
Bluetooth is a wireless LAN technology designed to connect devices of different functions such as telephones, notebooks, computers, cameras, printers, coffee makers, and so on. A Bluetooth LAN is an ad hoc network, which means that the network is formed spontaneously. Used for very short range communication between mobile phones, PDAs, notebook computers and other personal or peripheral devices.

Bluetooth is an adhoc network which means the network is formed spontaneously ie the devices find each other and make a network called piconet.

- Has a range of only 10 m
- Communication devices typically belong to one individual or group
  - Sometimes categorized as Personal Area Network (PAN)
- Version 2.0 provides speeds up to 2.1 Mbps
- Power consumption is low
- Bluetooth is specified by an industry consortium called the Bluetooth Special Interest Group
- It specifies an entire suite of protocols, going beyond the link layer to define application protocols, which it calls *profiles*, for a range of applications
  - There is a profile for synchronizing a PDA with personal computer
  - Another profile gives a mobile computer access to a wired LAN
- The basic Bluetooth network configuration is called a *piconet*
  - Consists of a master device and up to seven slave devices
  - Any communication is between the master and a slave
  - The slaves do not communicate directly with each other
  - A slave can be *parked*: set to an inactive, low-power state



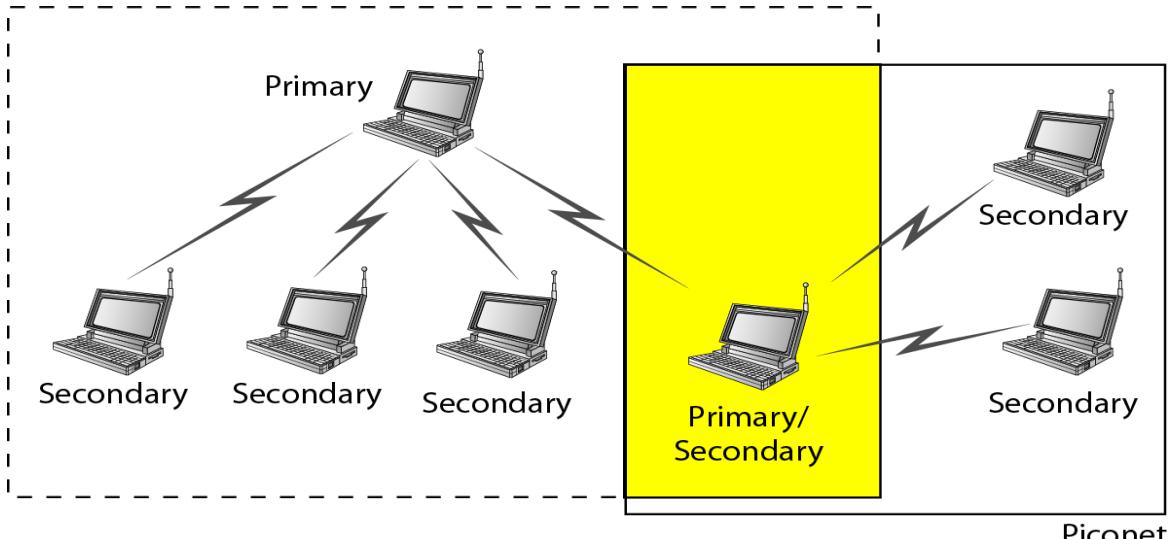
## Piconet



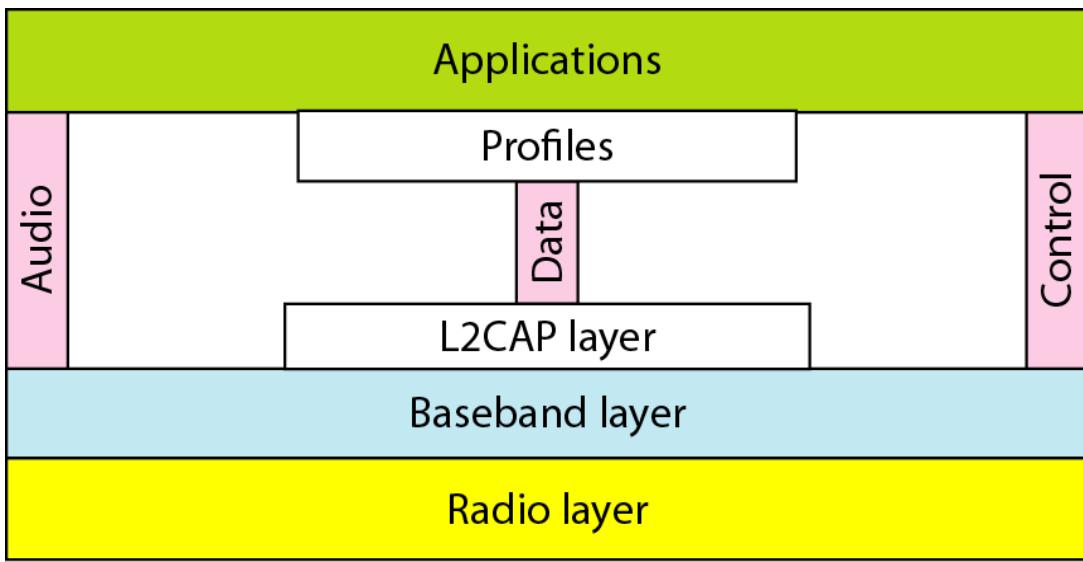
## Scatternet

A *scatternet* is a number of interconnected piconets that supports communication between more than 8 devices. Scatternets can be formed when a member of one piconet (either the master or one of the slaves) elects to participate as a slave in a second, separate piconet. The device participating in both piconets can relay data between members of both ad hoc networks.

## Piconet



## Bluetooth layers



#### L2CAP Layer

- Logical Link Control and Adaptation Layer
- Used for data exchange, It can do multiplexing

#### Baseband Layers

- Equivalent to MAC sublayer in LAN
- The primary and secondary stations communicate with each other using time slots (TDMA-Time Division Multiple Access)

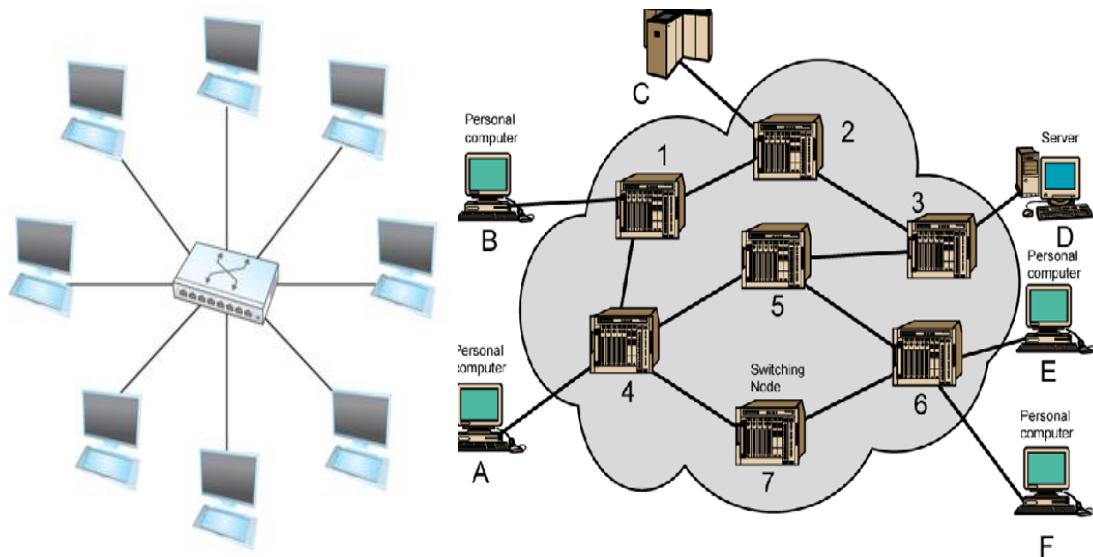
#### Radio layer

- Equivalent to physical layer of the internet model. Bluetooth devices are low-power and have a range of 10 m

### Switch

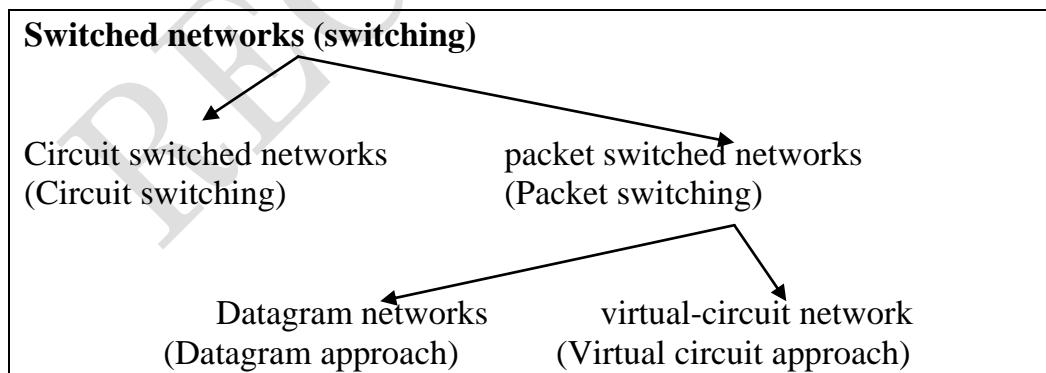
1. A mechanism that allows us to interconnect links(LANs) to form a large network is called **switching**.
2. Switches are devices capable of creating temporary connections between two or more devices linked to the switch.
3. A switch's primary job is to receive incoming packets on one of its links and to transmit them on some other link. This function is referred as switching and forwarding.
4. A multi-input, multi-output device which transfers packets from an input to one or more outputs.

## EXAMPLE:



### Advantages of switches:

1. Large networks can be built by interconnecting a number of switches.
2. We can connect switches to each other and to hosts using point-to-point links, which typically means that we can build networks of large geographic scope.
3. Adding a new host to the network by connecting it to a switch does not necessarily mean that the hosts already connected will get worse performance from the network.
4. A switch is connected to a set of links and for each of these links, runs the appropriate data link protocol to communicate with each node attached on link.



### Circuit switched networks

In circuit switched network, it consists of set of switches connected by physical links. A connection between two stations is a dedicated path made of one or more links. Physical layer is responsible for circuit switching.

The communication in circuit-switched network has three phases,

- 1) Connection setup
- 2) Data transfer
- 3) Teardown (connection termination)

In these types of networks, data are not packetized and there is a continuous flow by the source station to destination station.

## Packet switched networks

If the message is passing through a packet switched network, it needs to be divided into packets of fixed or variable size. This type of switching is done by network layer.

The two approaches commonly used are,

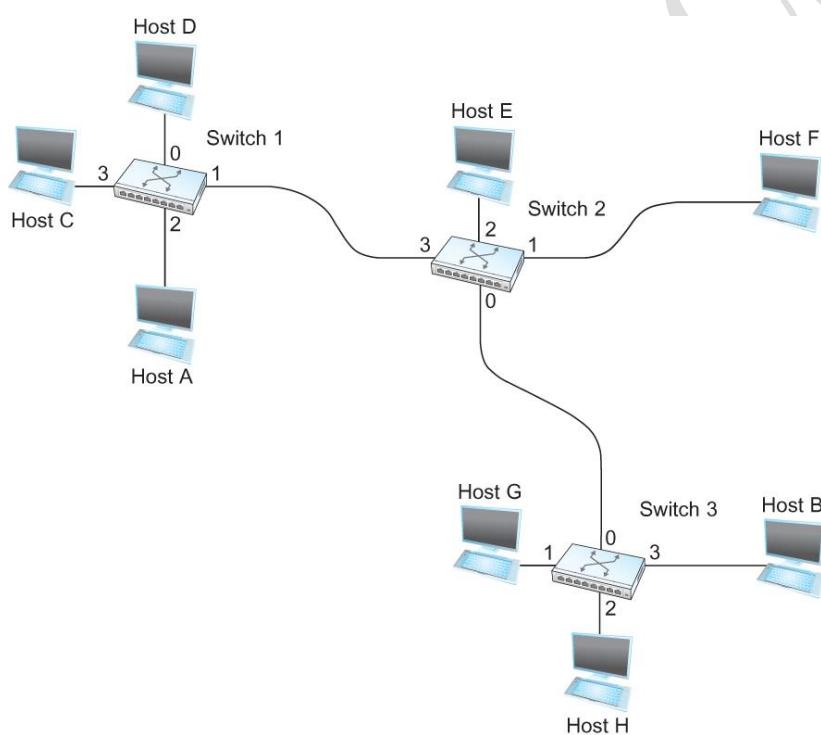
- *Datagram or Connectionless approach*
- *Virtual circuit or Connection-oriented approach*

### Datagram approach

Every packet contains enough information i.e. destination address that enable any switch to decide where the packet has to go.

Consider An example network with many hosts for datagram forwarding,

- To decide how to forward a packet, a switch consults a forwarding table (sometimes called a *routing table*)



Destination	Port
A	3
B	0
C	3
D	3
E	2
F	1
G	0
H	0

**Forwarding Table for Switch 2**

Fig :  
Datagram forwarding : an example network

### Characteristics of Connectionless (Datagram) Network

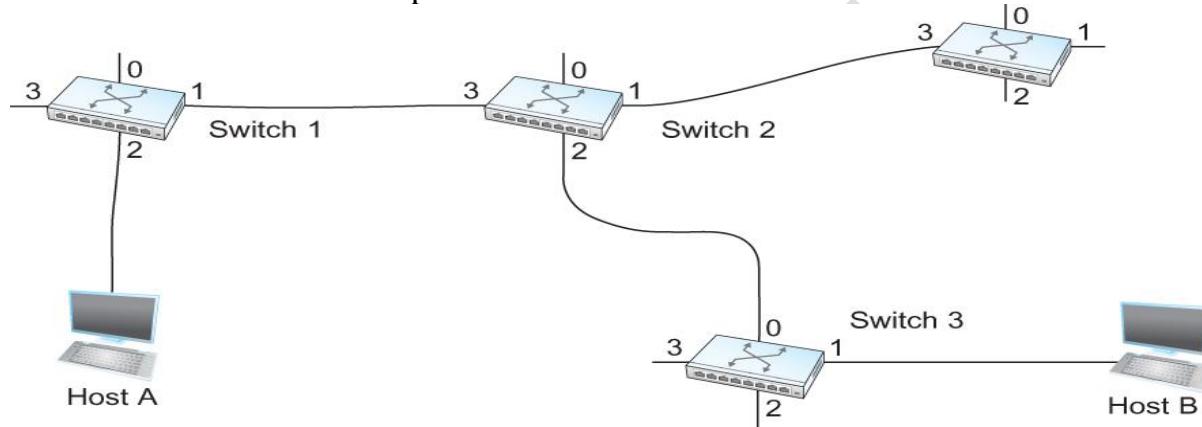
- A host can send a packet anywhere at any time.
- When a host sends a packet, it has no way of knowing if the network is capable of delivering it or if the destination host is even up and running.
- Each packet is forwarded independently of previous packets that might have been sent to the same destination.

- Thus two successive packets from host A to host B may follow completely different paths
- A switch or link failure might not have any serious effect on communication if it is possible to find an alternate route around the failure and update the forwarding table accordingly.

### **Virtual Circuit Switching**

- Widely used technique for packet switching
- Uses the concept of **virtual circuit (VC)**
- Also called a connection-oriented model
- First set up a virtual connection from the source host to the destination host and then send the data

- Host A wants to send packets to host B



**Fig : An example of a virtual circuit network**

Two-stage process

- Connection setup
- Data Transfer

### **Connection setup**

- Establish “connection state” in each of the switches between the source and destination hosts
- The connection state for a single connection consists of an entry in the “VC table” in each switch through which the connection passes

One entry in the VC table on a single switch contains

- A virtual circuit identifier (VCI) that uniquely identifies the connection at this switch and that will be carried inside the header of the packets that belong to this connection
- An incoming interface on which packets for this VC arrive at the switch
- An outgoing interface in which packets for this VC leave the switch

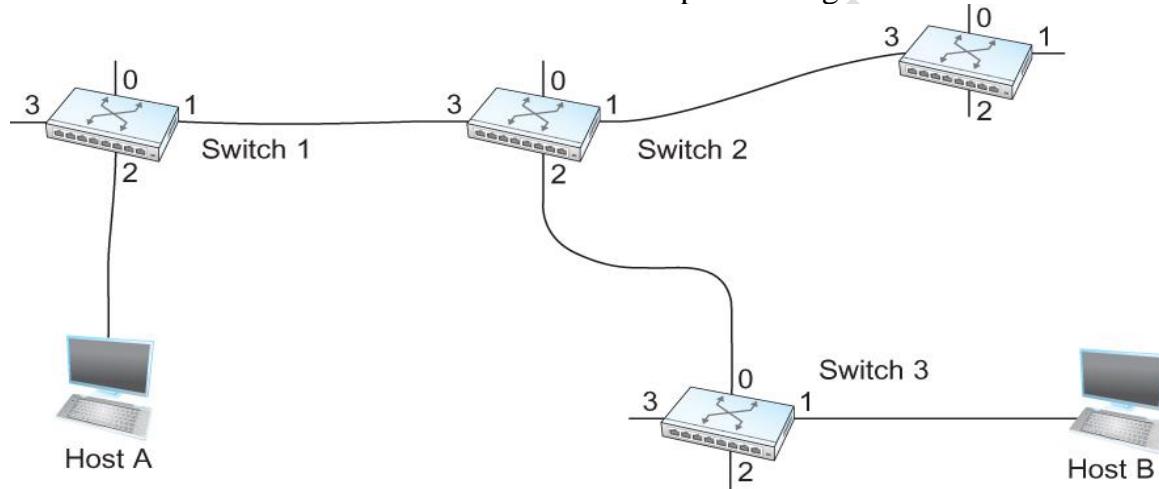
A potentially different VCI that will be used for outgoing packets.

Two broad classes of approach to establishing connection state

- Network Administrator will configure the state
  - The virtual circuit is permanent (PVC)
  - The network administrator can delete this
  - Can be thought of as a long-lived or administratively configured VC
- A host can send messages into the network to cause the state to be established
  - This is referred as signalling and the resulting virtual circuit is said to be switched (SVC)
  - A host may set up and delete such a VC dynamically without the involvement of a network administrator

Let's assume that a network administrator wants to manually create a new virtual connection from host A to host B

- First the administrator identifies a path through the network from A to B



The administrator then picks a VCI value that is currently unused on each link for the connection

- For our example,
  - Suppose the VCI value 5 is chosen for the link from host A to switch 1
  - 11 is chosen for the link from switch 1 to switch 2
  - So the switch 1 will have an entry in the VC table

For switch 1

Incoming Interface	Incoming VC	Outgoing Interface	Outgoing VC
2	5	1	11

Similarly, suppose

- VCI of 7 is chosen to identify this connection on the link from switch 2 to switch 3
- VCI of 4 is chosen for the link from switch 3 to host B
- Switches 2 and 3 are configured with the following VC table

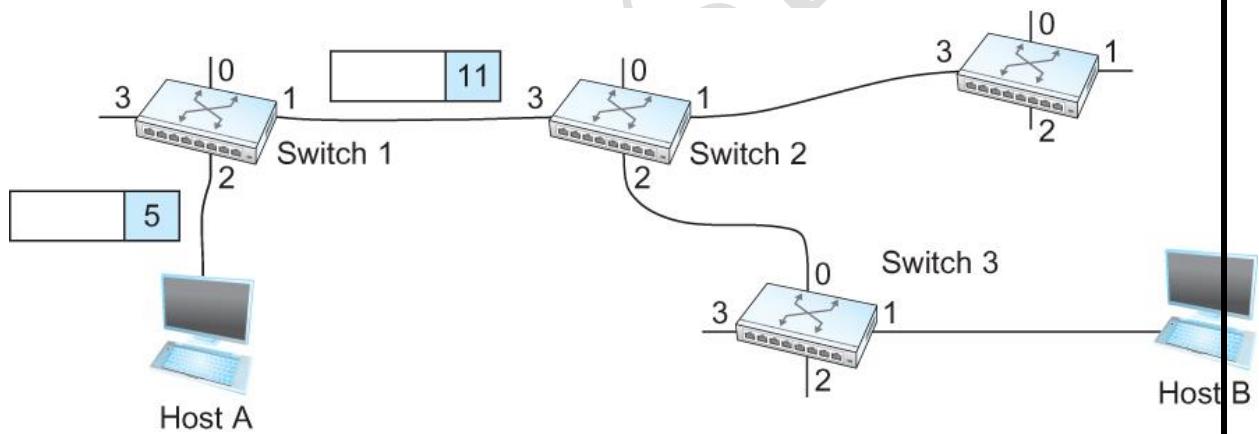
Switch 2

Incoming Interface	Incoming VC	Outgoing Interface	Outgoing VC
3	11	2	7

Switch 3

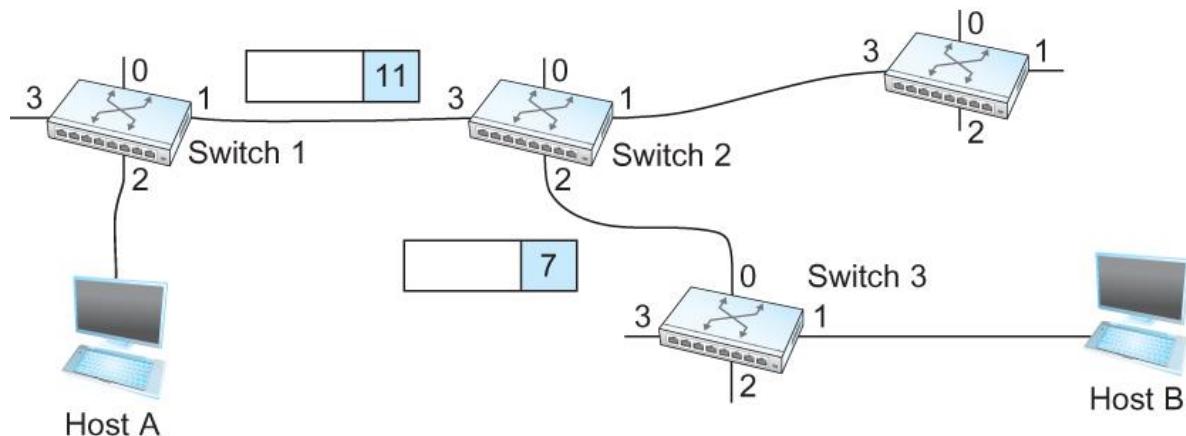
Incoming Interface	Incoming VC	Outgoing Interface	Outgoing VC
0	7	1	4

- For any packet that A wants to send to B, A puts the VCI value 5 in the header of the packet and sends it to switch 1
- Switch 1 receives any such packet on interface 2, and it uses the combination of the interface and the VCI in the packet header to find the appropriate VC table entry.
- The table entry on switch 1 tells the switch to forward the packet out of interface 1 and to put the VCI value 11 in the header



Packet will arrive at switch 2 on interface 3 bearing VCI 11

- Switch 2 looks up interface 3 and VCI 11 in its VC table and sends the packet on to switch 3 after updating the VCI value appropriately
- This process continues until it arrives at host B with the VCI value of 4 in the packet
- To host B, this identifies the packet as having come from host A



## Data transfer

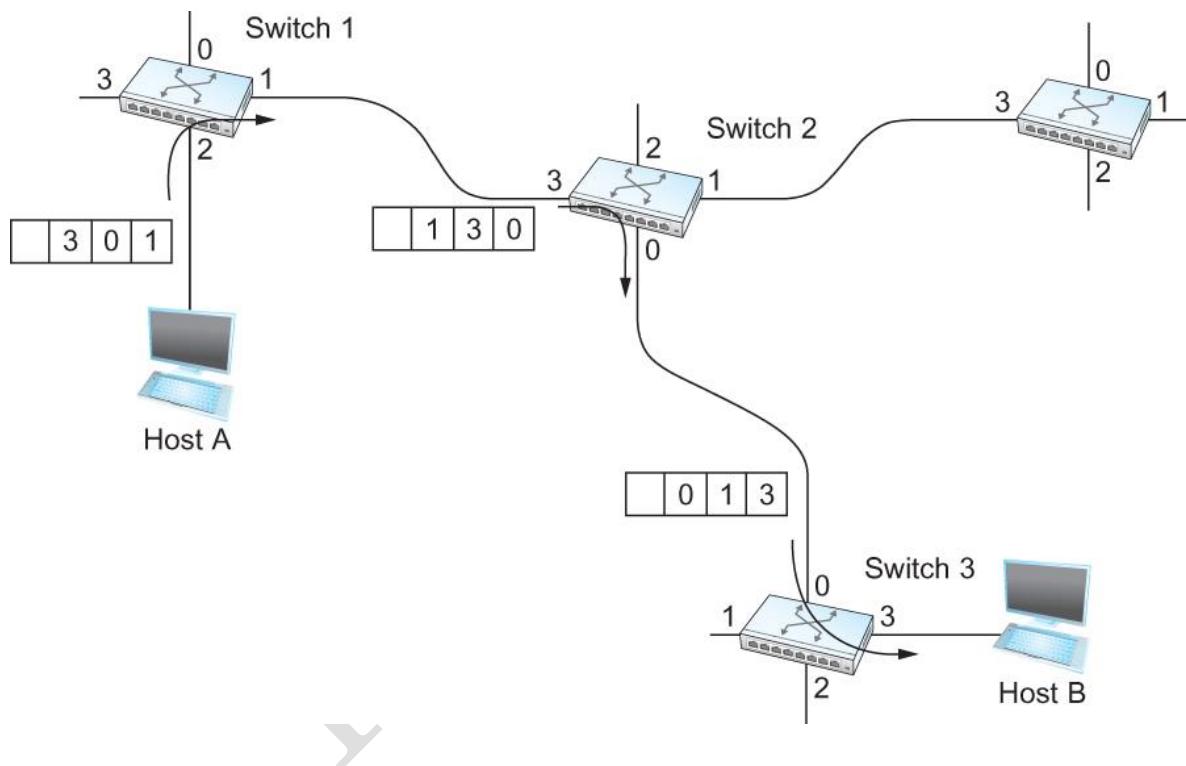
The data transfer phase is active until the source sends all its frame to destination when the connection state is established.

## Teardown

In this phase, source A, after sending all frames to B, sends a special frame called teardown request. Destination B responds with a teardown confirmation frame and all switches delete the corresponding entry from their tables.

## Source Routing

- Its a type if less commonly used approach in packet switching
- All the information about network topology that is required to switch a packet across the network is provided by the source host



Circuit switching	Packet switching
The connection between two station is a dedicated path using physical links	Virtual connection is established between two station
Physical layer is responsible	Network layer is responsible
Messages are not packetized and there is a continuous flow of messages	Messages are divided into packets of fixed or variable size
Used in telephone companies	Used in switched WANs such as frame relay and ATM networks
-	Types are <ul style="list-style-type: none"> <li>• Datagram approach</li> <li>• Virtual circuit approach</li> <li>• Source routing</li> </ul>

## Bridges

- Bridge is a connecting device that is used to connect two or more LANs. It operates in both the physical and the data link layer.
- As a physical layer device, it regenerates the signal it receives.
- As a data link layer device, the bridge can check the physical (MAC) addresses (source and destination) contained in the frame.
- Switch that is used to forward packets between shared-media LANs such as Ethernets.
- It is also called as LAN switches.

### Transparent bridges

It is a bridge in which the stations are completely unaware of bridge's existence. If a bridge is added or deleted from the system, reconfiguration is not needed.

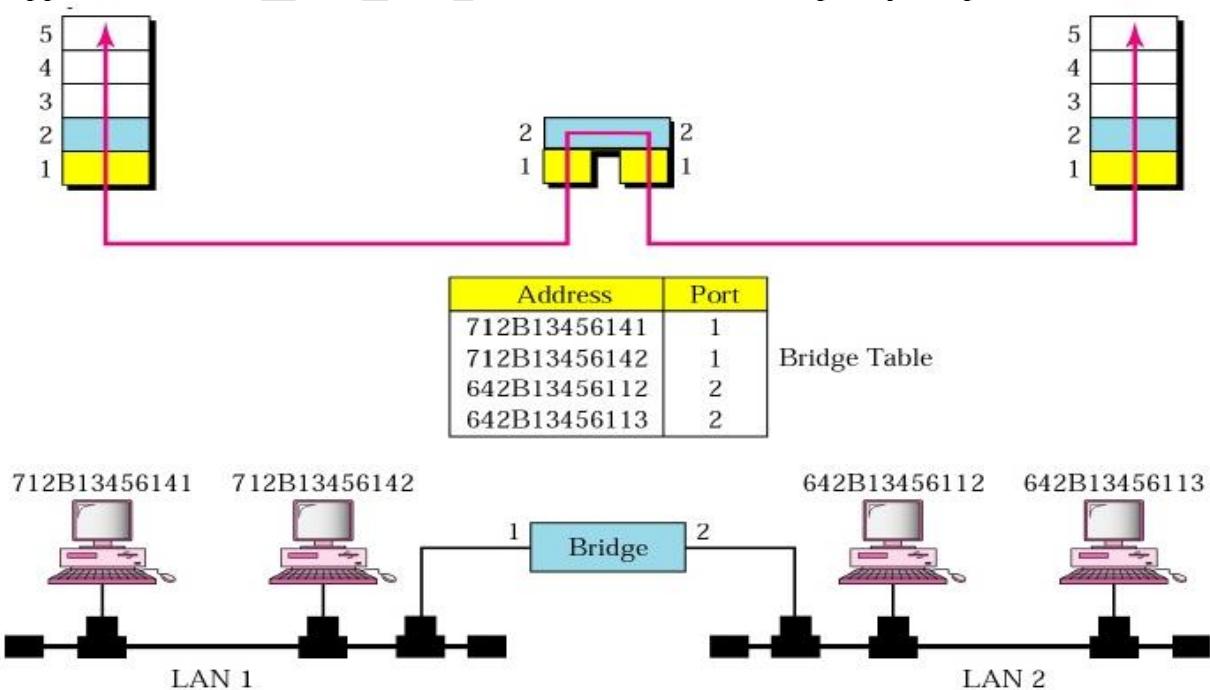
According to the IEEE 802.1 d specification, a system equipped with transparent bridges must meet three criteria, as follows

### Functions of bridges (three criteria)

1. Frame filtering or forwarding
2. Learning
3. Avoidance of loops in system

### Frame filtering or forwarding

Bridges has a table which is used for filtering or forwarding decisions. It can check the destination address of frame and decide if the frame has to be forwarded or dropped. If the frame is to be forwarded, the decision must specify the port.



Let us give an example. In Figure two LANs are connected by a bridge. If a frame destined for station 712B13456142 arrives at port 1, the bridge consults its table to find the departing port. According to its table, frames for 712B 13456142 leave through port 1; therefore, there is no need for forwarding, and the frame is dropped. On the other hand, if a frame for 712B13456141 arrives at port 2, the departing port is port 1 and the frame is forwarded.

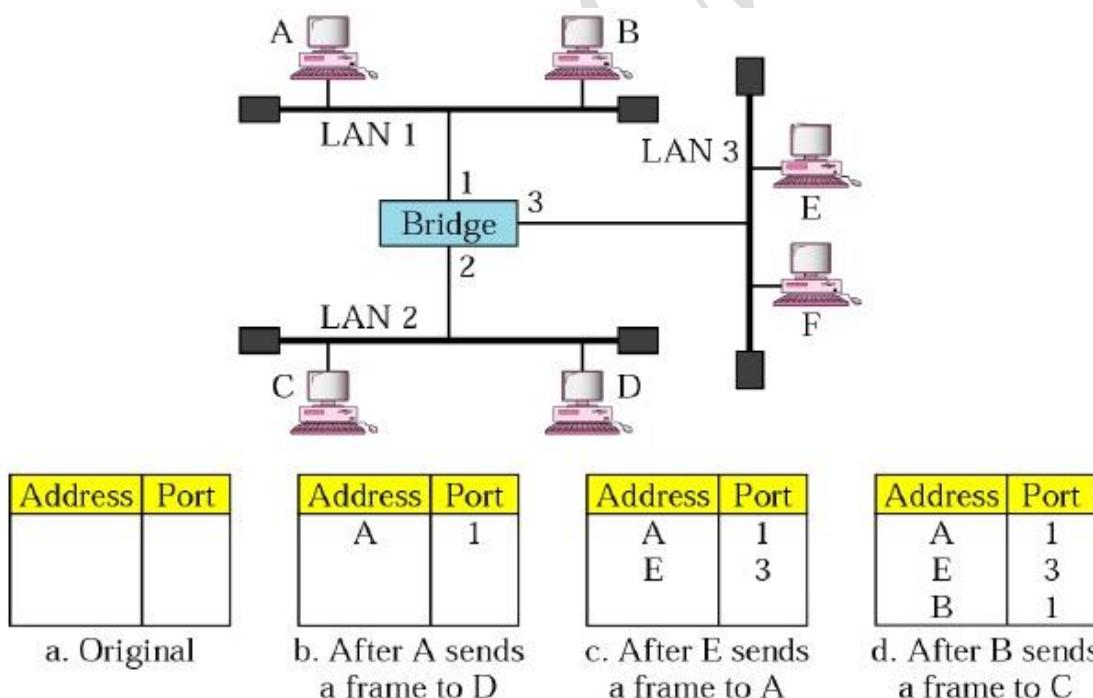
## Learning

The earliest bridges had forwarding tables that were static. The systems administrator would manually enter each table entry during bridge setup. Although the process was simple, it was not practical.

If a station was added or deleted, the table had to be modified manually. The same was true if a station's MAC address changed, which is not a rare event.

A better solution to the static table is a dynamic table that maps addresses to ports automatically. To make a table dynamic, we need a bridge that gradually learns from the frame movements.

To do this, the bridge inspects both the destination and the source addresses. The destination address is used for the forwarding decision (table lookup); the source address is used for adding entries to the table and for updating purposes.



- When station A sends a frame to station D, the bridge does not have an entry for either D or A. The frame goes out from all three ports; the frame floods the network. However, by looking at the source address, the bridge learns that station A must be located on the LAN connected to port 1. This means that frames destined for A, in the future, must be sent out through port 1. The bridge adds this entry to its table. The table has its first entry now.

2. When station E sends a frame to station A, the bridge has an entry for A, so it forwards the frame only to port 1. There is no flooding. In addition, it uses the source address of the frame, E, to add a second entry to the table.
3. When station B sends a frame to C, the bridge has no entry for C, so once again it floods the network and adds one more entry to the table.
4. The process of learning continues as the bridge forwards frames.

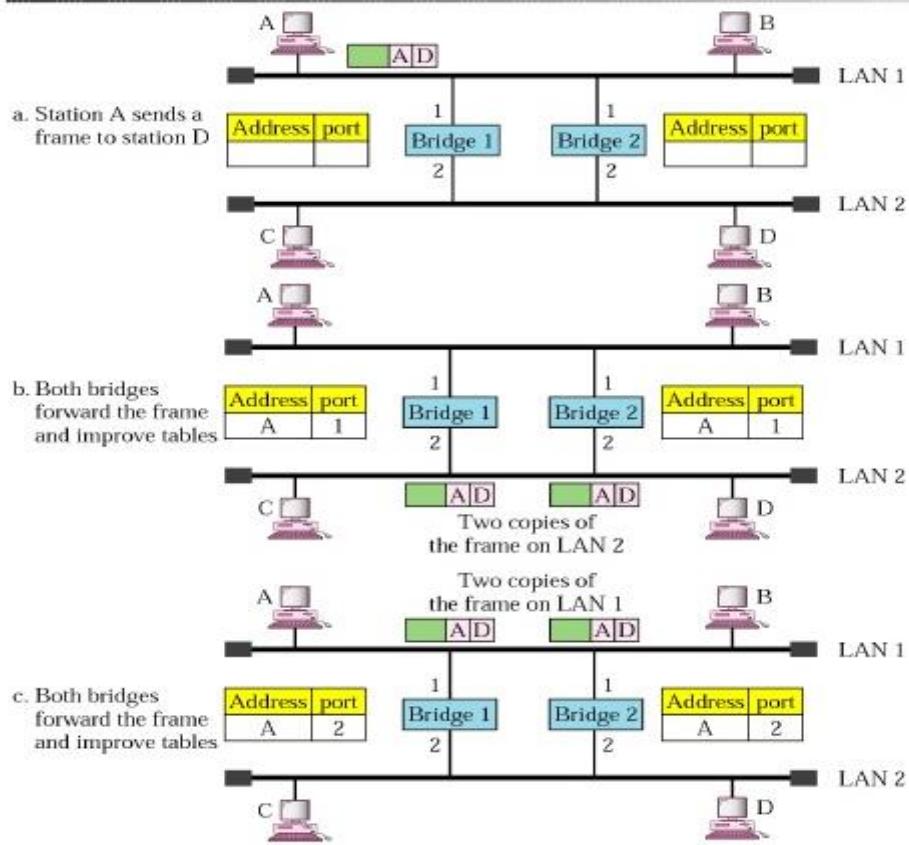
### **Loop Problem**

Loop Problem Transparent bridges work fine as long as there are no redundant bridges in the system. Systems administrators, however, like to have redundant bridges (more than one bridge between a pair of LANs) to make the system more reliable.

If a bridge fails, another bridge takes over until the failed one is repaired or replaced. Redundancy can create loops in the system, which is very undesirable.

Figure shows a very simple example of a loop created in a system with two LANs connected by two bridges.

1. Station A sends a frame to station D. The tables of both bridges are empty. Both forward the frame and update their tables based on the source address A.
2. Now there are two copies of the frame on LAN 2. The copy sent out by bridge 1 is received by bridge 2, which does not have any information about the destination address D; it floods the bridge. The copy sent out by bridge 2 is received by bridge 1 and is sent out for lack of information about D. Note that each frame is handled separately because bridges, as two nodes on a network sharing the medium, use an access method such as CSMA/CD. The tables of both bridges are updated, but still there is no information for destination D.
3. Now there are two copies of the frame on LAN 1. Step 2 is repeated, and both copies flood the network.
4. The process continues on and on. Note that bridges are also repeaters and regenerate frames. So in each iteration, there are newly generated fresh copies of the frames. To solve the looping problem, the IEEE specification requires that bridges use the spanning tree algorithm to create a loopless topology.

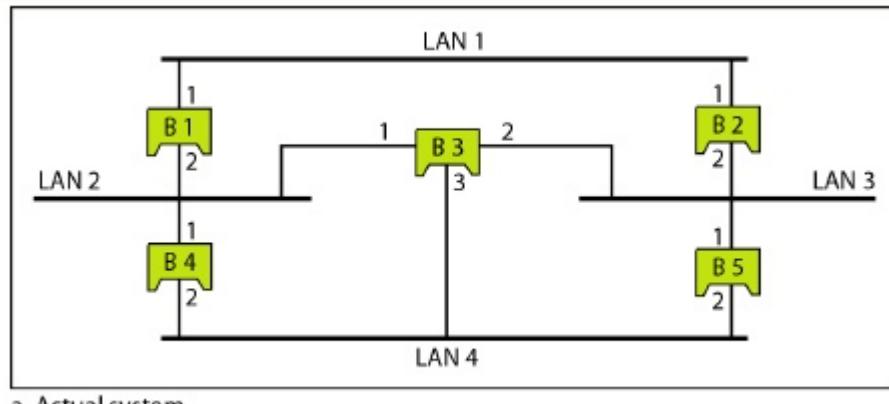


## Spanning Tree

In graph theory, a **spanning tree** is a graph in which there is no loop. In a bridged LAN, this means creating a topology in which each LAN can be reached from any other LAN through one path only (no loop).

We cannot change the physical topology of the system because of physical connections between cables and bridges, but we can create a logical topology that overlays the physical one.

The following figure shows a system with four LANs and five bridges.

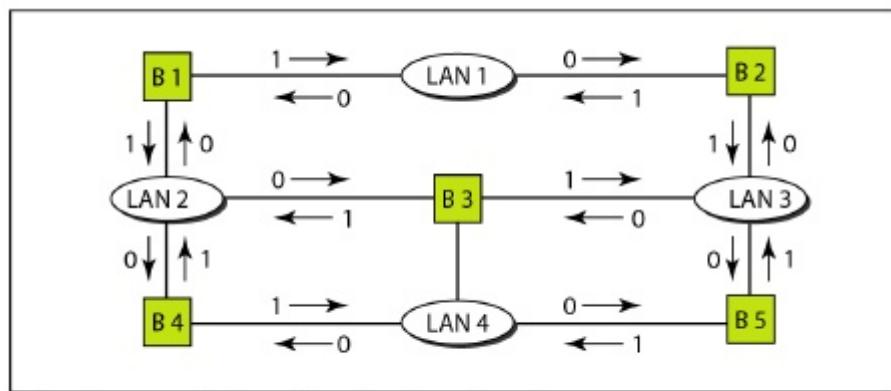


a. Actual system

The graph representation with cost assigned to each arc of above system is as follows.

The connecting arcs show the connection of a LAN to a bridge and vice versa. To find the spanning tree, we need to assign a cost (metric) to each arc.

[ Note : assign 1 from Bridge to LAN, assign 0 from LAN to Bridge ]

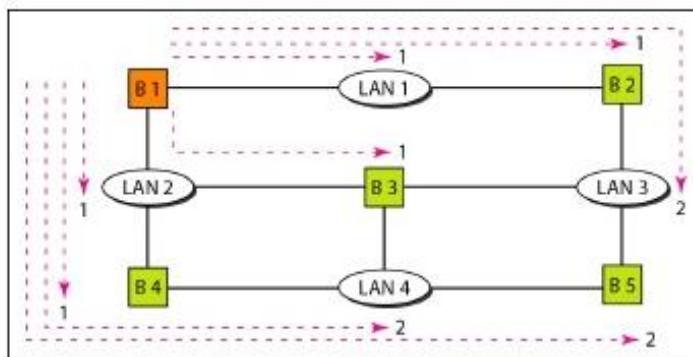


b. Graph representation with cost assigned to each arc

In spanning tree approach, the cost to each arc is assigned by system administrator.

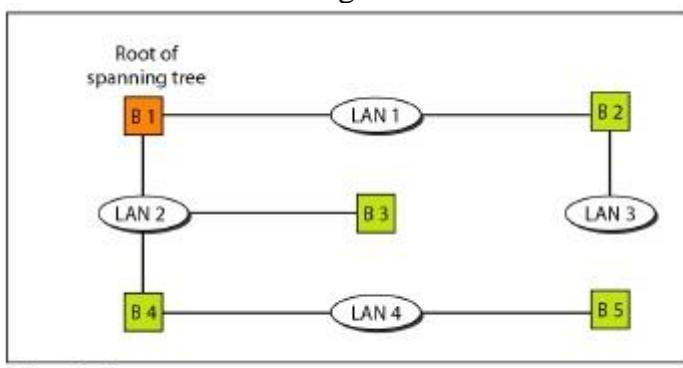
The process to find the spanning tree involves three steps:

1. Every bridge has a built-in ID (normally the serial number, which is unique). Each bridge broadcasts this ID so that all bridges know which one has the smallest ID. The bridge with the smallest ID is selected as the *root bridge* (root of the tree). We assume that bridge B1 has the smallest ID. It is, therefore, selected as the root bridge.
2. The algorithm tries to find the shortest path (a path with the shortest cost) from the root bridge to every other bridge or LAN as follows,



a. Shortest paths

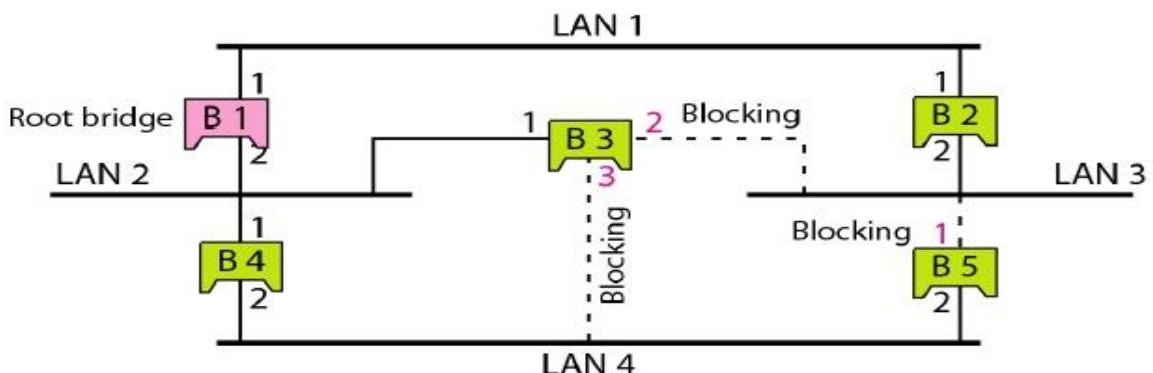
3. The combination of the shortest paths creates the shortest tree, which is also shown in the above figure.



b. Spanning tree

4. Based on the spanning tree approach some ports are forwarding ports, which forward a frame that the bridge receives. Some ports are blocking ports which block the frames received by bridge.

#### ***Forwarding and blocking ports after using spanning tree algorithm***



Ports 2 and 3 of bridge B3 are blocking ports. Port1 of bridge B5 is also blocking port.

- Spanning tree algorithm is a dynamic algorithm.
- Each bridge is equipped with a software process that carries the process dynamically. Each bridge send special messages to one another called bridge protocol data units (BPDUs) to update spanning tree.

#### **Limitations of Bridges**

On the issue of scale, it is not realistic to connect more than a few LANs by means of bridges, where in practice “few” typically means “tens of.”

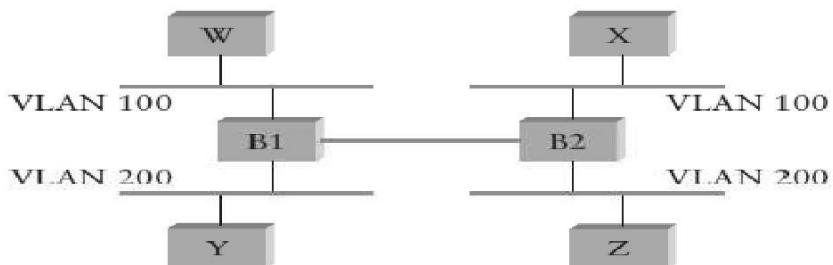
One reason for this is that the spanning tree algorithm scales linearly; that is, there is no provision for imposing a hierarchy on the extended LAN.

A second reason is that bridges forward all broadcast frames.

One approach to increasing the scalability of extended LANs is the *virtual LAN(VLAN)*. VLANs allow a single extended LAN to be partitioned into several seemingly separate LANs.

Each virtual LAN is assigned an identifier (sometimes called a *color*), and packets can only travel from one segment to another if both segments have the same identifier.

#### **Two virtual LANs share a common backbone**



shows four hosts on four different LAN segments. In the absence of VLANs, any broadcast packet from any host will reach all the other hosts.

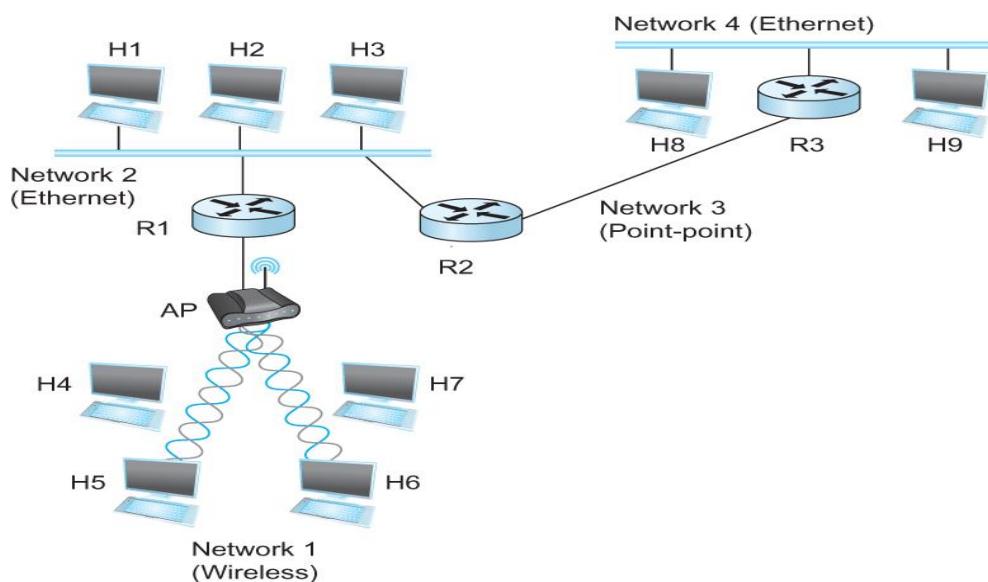
## **Internetworking**

The term —internetwork or sometimes just internet refers to an arbitrary collection of networks interconnected to provide some sort of host to- host packet delivery service.

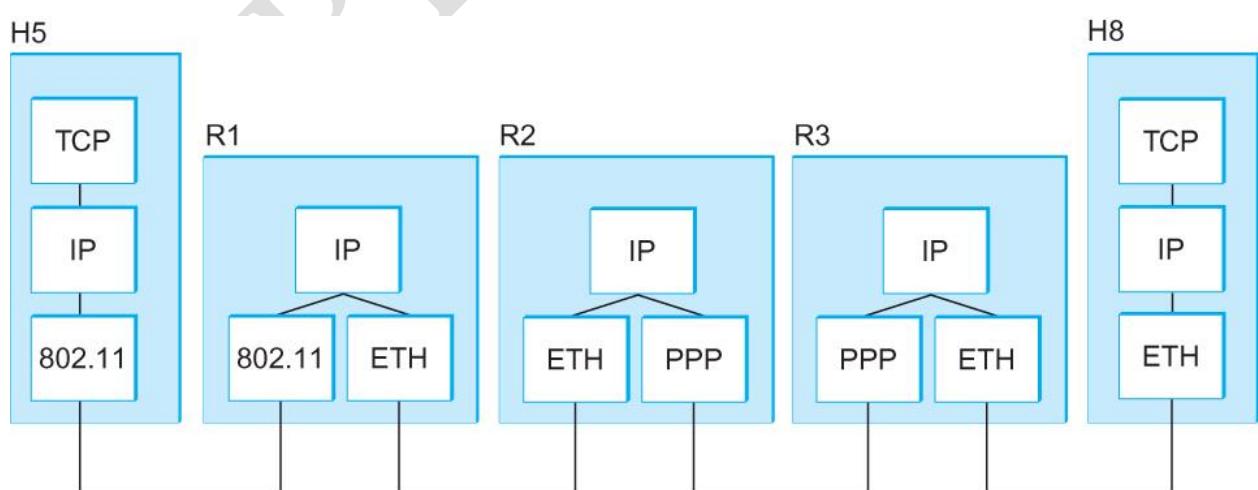
*internetwork is often referred to as a “net work of networks” because it is made up of lots of smaller networks.al network built out of a collection of physical networks.*

The nodes that interconnect the networks are called routers. They are also sometimes called gateways.

The Internet Protocol is the key tool used today to build scalable, heterogeneous internetworks. It was originally known as the Kahn-Cerf protocol after its inventors.



**Fig: A simple internetwork. Hn = host; Rn = router.**



**Fig : A simple internetwork, showing the protocol layers used to connect H5 to H8**

## **IP (Internet Protocol)**

- IP stands for Internet Protocol
- Key tool used today to build scalable, heterogeneous internetworks
- It runs on all the nodes in a collection of networks and defines the infrastructure that allows these nodes and networks to function as a single logical internetwork

### **Service Model**

Service model is, the host-to-host services you want to provide.

- It has two parts
  - 1. Datagram Delivery Model
    - Connectionless model for data delivery
  - 2. Global Addressing Scheme
    - Provides a way to identify all hosts in the network

### **Datagram Delivery**

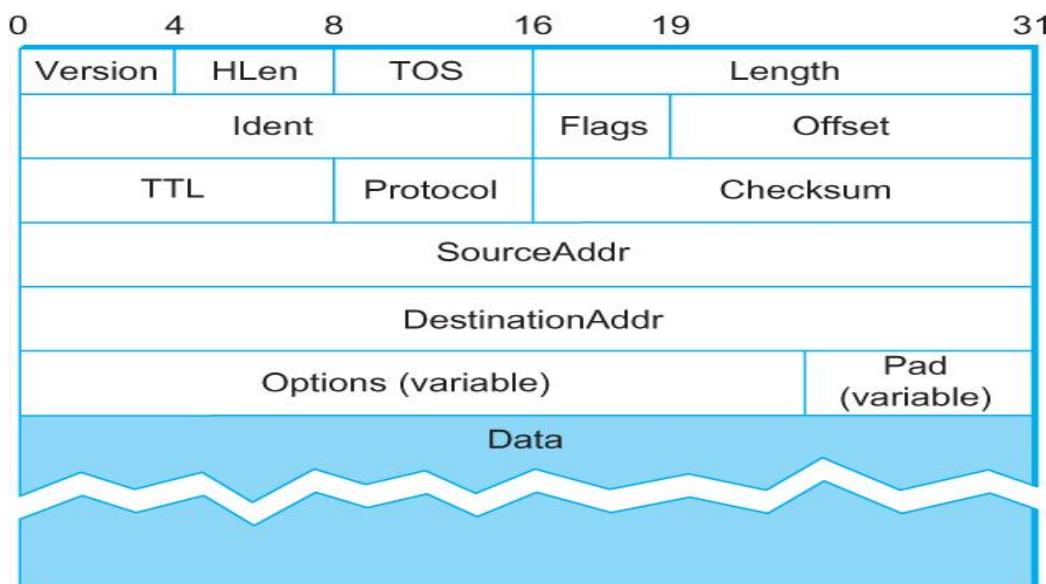
5. Datagram is a type of packet that happens to be sent in a connectionless manner over a network. Every datagram carries enough information to let the network forward the packet to its correct destination; there is no need for any advance setup mechanism to tell the network what to do when the packet arrives.

- Best-effort delivery (unreliable service)
  - » packets are lost or
  - » packets are delivered out of order or
  - » duplicate copies of a packet are delivered or
  - » packets can be delayed for a long time

### **Advantages of Datagram Delivery Model**

- ✓ Best-effort, connectionless service
- ✓ simplest service model
- ✓ asking for a reliable packet delivery service may need to include a lot of extra functionality into the router.
- ✓ It enables IP to “run over anything” ie today IP can run over many network technologies
- ✓ Higher level protocols such as TCP that run over IP is aware of all failure modes.

## IPv4 Packet Format



- Version (4): currently 4
- Hlen (4): number of 32-bit words in header
- TOS (8): type of service (not widely used)
- Length (16): number of bytes in this datagram. Max size of IP datagram is 65535 Bytes which is not supported for physical networks. So IP supports fragmentation and reassembly.
- Ident (16): used by fragmentation
- Flags/Offset (16): used by fragmentation
- TTL (8): specifies no of sec that a packet would be allowed to live
- Protocol (8): demux key (TCP=6, UDP=17)
- Checksum (16): for error detection
- DestAddr & SrcAddr (32)

## IP Fragmentation and Reassembly

- Each network has some MTU (Maximum Transmission Unit)
  - Ethernet (1500 bytes), FDDI (4500 bytes)
- Strategy
  - Fragmentation occurs in a router when it receives a datagram that it wants to forward over a network which has (MTU < datagram)
  - Reassembly is done at the receiving host
  - All the fragments carry the same identifier in the *Ident* field
  - Fragments are self-contained datagrams
  - IP does not recover from missing fragments

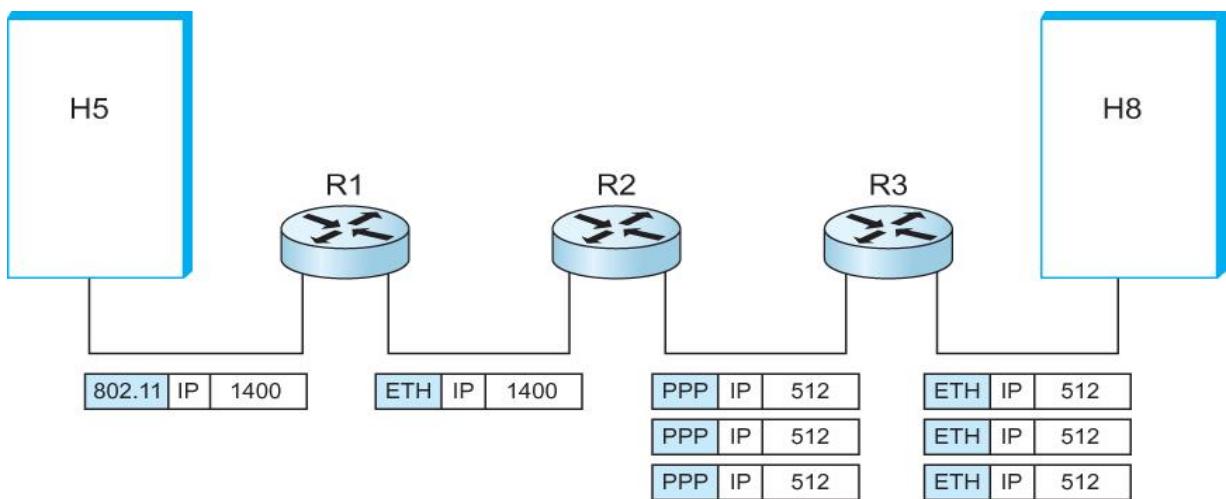


Fig: IP datagrams traversing the sequence of physical networks

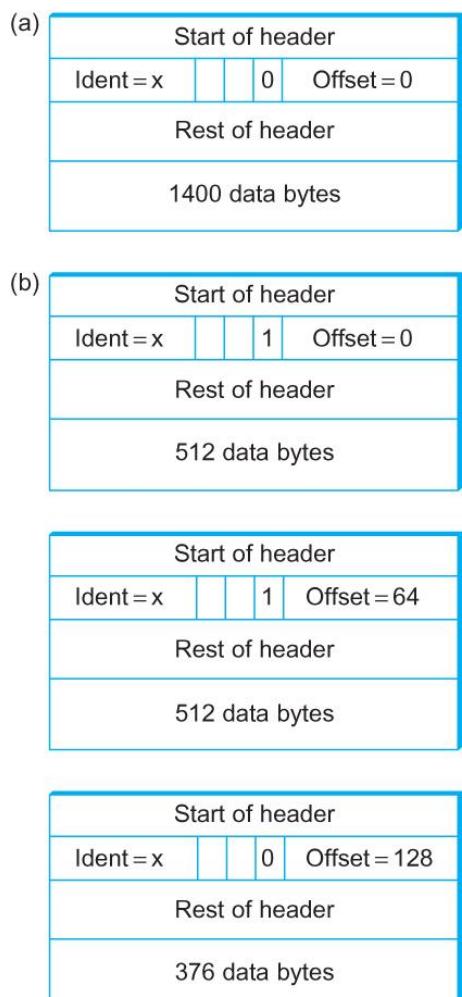
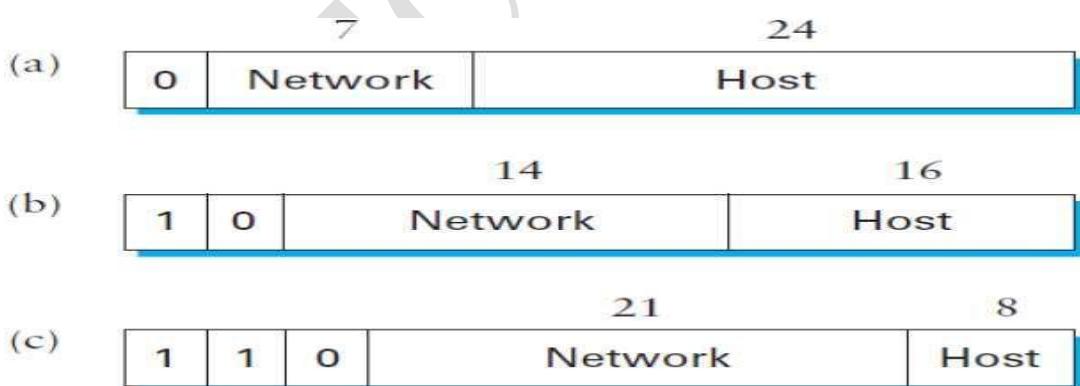


Fig: Header fields used in IP fragmentation. (a) Unfragmented packet; (b) fragmented packets

## Global Addresses

- Global uniqueness is the first property that should be provided in an addressing scheme.
- Ethernet addresses are globally unique, but that alone does not suffice for an addressing scheme in a large internetwork. Ethernet addresses are also *flat*, which means that they have no structure and provide very few clues to routing protocols.
- In contrast, IP addresses are *hierarchical*, by which we mean that they are made up of several parts that correspond to some sort of hierarchy in the internetwork.
- **Specifically, IP addresses consist of two parts, a network part and a host part.** This is a fairly logical structure for an internetwork, which is made up of many interconnected networks.
- The network part of an IP address identifies the network to which the host is attached; all hosts attached to the same network have the same network part in their IP address.
- The host part then identifies each host uniquely on that particular network.
  - IP addresses are divided into three different classes.
- (There are also class D addresses that specify a multicast group, and class E addresses that are currently unused.)



**IP addresses : (a)class A;(b)class B;class C.**

The class of an IP address is identified in the most significant few bits. If the first bit is 0, it is a class A address. If the first bit is 1 and the second is 0, it is a class B address.

If the first two bits are 1 and the third is 0, it is a class C address. Thus, of the approximately 4 billion possible IP addresses, half are class A, one quarter are class B, and one-eighth are class C.

Each class allocates a certain number of bits for the network part of the address and the rest for the host part.

Class A networks have 7 bits for the network part and 24 bits for the host part, meaning that there can be only 126 class A networks (the values 0 and 127 are reserved), but each of them can accommodate up to  $2^{24} - 2$  (about 16 million) hosts (again, there are two reserved values).

Class B addresses allocate 14 bits for the network and 16 bits for the host, meaning that each class B network has room for 65,534 hosts. Finally, class C addresses have only 8 bits for the host and 21 for the network part.

Therefore, a class C network can have only 256 unique host identifiers, which means only 254 attached hosts (one host identifier, 255, is reserved for broadcast, and 0 is not a valid host number). However, the addressing scheme supports 221 class C networks.

By convention, IP addresses are written as four *decimal* integers separated by dots. Each integer represents the decimal value contained in 1 byte of the address, starting at the most significant. For example, the address of the computer on which this sentence was typed is 171.69.210.245.

## Datagram Forwarding in IP

- Strategy
  - every datagram contains destination's address
  - if directly connected to destination network, then forward to host
  - if not directly connected to destination network, then forward to some router
  - forwarding table maps network number into next hop
  - each host has a default router
  - each router maintains a forwarding table

## Algorithm

```
if (NetworkNum of destination = NetworkNum of one of my interfaces) then
    deliver packet to destination over that interface
else
    if (NetworkNum of destination is in my forwarding table) then
        deliver packet to NextHop router
    else
        deliver packet to default router
For a host with only one interface and only a default router in its forwarding table, this
simplifies to
if (NetworkNum of destination = my NetworkNum)then
    deliver packet to destination directly
else
    deliver packet to default router
```

## **SUBNETTING**

The original intent of IP addresses was that the network part would uniquely identify exactly one physical network.

It has two drawbacks.

1. Address assignment inefficiency.
2. Assigning network number to every physical network uses the IP address space potentially much faster.

Assigning many network numbers has another drawback that becomes apparent when you think about routing.

**Subnetting provides an elegantly simple way to reduce the total number of network numbers that are assigned.**

The idea is to take a single IP network number and allocate the IP addresses with that network number to several physical networks, which are now referred to as *subnets*.

- First, the subnets should be close to each other.
- This is because at a distant point in the Internet, they will all look like a single network, having only one network number between them.
- This means that a router will only be able to select one route to reach any of the subnets, so they had better all be in the same general direction.
- A perfect situation in which to use subnetting is a large campus or corporation that has many physical networks.
- From outside the campus, all you need to know to reach any subnet inside the campus is where the campus connects to the rest of the Internet.
- The mechanism by which a single network number can be shared among multiple networks involves configuring all the nodes on each subnet with a *subnet mask*.
- With simple IP addresses, all hosts on the same network must have the same network number.
- The subnet mask enables us to introduce a *subnet number; all hosts on the same* physical network will have the same subnet number, which means that hosts may be on different physical networks but share a single network number.
- For example, suppose that we want to share a single class B address among several physical networks. We could use a subnet mask of 255.255.255.0.
- (Subnet masks are written down just like IP addresses; this mask is therefore all 1s in the upper 24 bits and 0s in the lower 8 bits.)
- In effect, this means that the top 24 bits(where the mask has 1s) are now defined to be the network number, and the lower 8 bits (where the mask has 0s) are the host number.
- Since the top 16 bits identify the network in a class B address, we may now think of the address as having not two parts but three: a network part, a subnet part, and a host part.

## Subnet Addressing

Network number	Host number
----------------	-------------

Class B address

11111111111111111111111111111111	00000000
----------------------------------	----------

Subnet mask (255.255.255.0)

Network number	Subnet ID	Host ID
----------------	-----------	---------

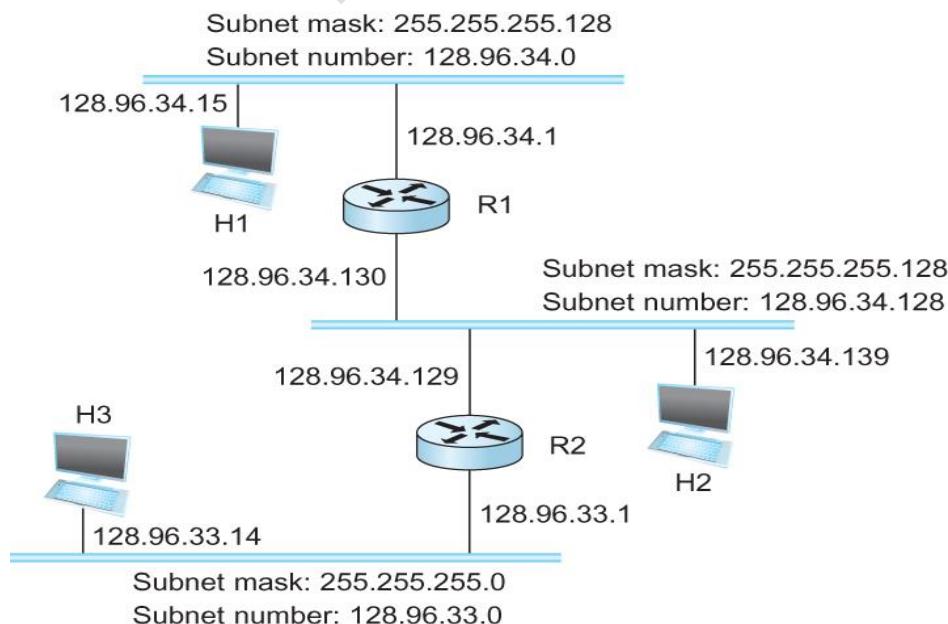
Subnetted address

- What subnetting means to a host is that it is now configured with both an IP address and a subnet mask for the subnet to which it is attached.
- For example, host H1 in Figure 4.26 is configured with an address of 128.96.34.15 and a subnet mask of 255.255.255.128.
- The bitwise AND of these two numbers defines the subnet number of the host and of all other hosts on the same subnet.
- In this case, 128.96.34.15 AND 255.255.255.128 equals 128.96.34.0, so this is the subnet number for the topmost subnet.

## Forwarding Packet through Subnet

When the host wants to send a packet to a certain IP address, the first thing it does is to perform a bitwise AND between its own subnet mask and the destination IP address. If the results are not equal, the packet needs to be sent to a router to be forwarded to another subnet.

For example, if H1 is sending to H2, then H1 ANDs its subnet mask (255.255.255.128) with the address for H2 (128.96.34.139) to obtain 128.96.34.128. This does not match the subnet number for H1 (128.96.34.0) so H1 knows that H2 is on a different subnet. Since H1 cannot deliver the packet to H2 directly over the subnet, it sends the packet to its default router R1.



To support subnetting, the routing table must now hold entries of the form (SubnetNumber, SubnetMask, NextHop). To find the right entry in the table, the router ANDs the packet's destination address with the SubnetMask for each entry in turn; if the result matches the SubnetNumber of the entry, then this is the right entry to use, and it forwards the packet to the next hop router indicated.

### **Forwarding Algorithm**

```
D = destination IP address
for each entry < SubnetNum, SubnetMask, NextHop>
D1 = SubnetMask & D
if D1 = SubnetNum
if NextHop is an interface
    deliver datagram directly to destination
else
    deliver datagram to NextHop (a router)
```

### **Example Subnetting Table**

SubnetNumber	SubnetMask	NextHop
128.96.34.0	255.255.255.128	Interface 0
128.96.34.128	255.255.255.128	Interface 1
128.96.33.0	255.255.255.0	R2

### **Classless Routing (CIDR)**

Classless Inter Domain Routing (CIDR, pronounced —cider) is a technique that addresses two scaling concerns in the Internet.

- Do not use classes to determine network ID
- Assign any range of addresses to network
  - Use common part of address as network number
  - E.g., addresses 192.4.16 - 192.4.31 have the first 20 bits in common.  
Thus, we use these 20 bits as the network number
  - netmask is /20, /xx is valid for almost any xx
- Enables more efficient usage of address space (and router tables)

The growth of backbone routing tables as more and more network numbers need to be stored in them, and the potential for the 32-bit IP address space to be exhausted well before the four-billionth host is attached to the Internet.

This address space exhaustion is called address assignment inefficiency.

The inefficiency arises because the IP address structure, with class A, B, and C addresses, forces us to hand out network address space in fixed-sized chunks of three very different sizes.

- A network with two hosts needs a class C address
  - Address assignment efficiency =  $2/255 = 0.78$
- A network with 256 hosts needs a class B address
  - Address assignment efficiency =  $256/65535 = 0.39$

Even though subnetting can help us to assign addresses carefully, it does not get around the fact that any autonomous system with more than 255 hosts, or an expectation of eventually having that many, wants a class B address.

As it turns out, exhaustion of the IP address space centers on exhaustion of the class B network numbers.

One way to deal with that would seem to be saying no to any AS (Autonomous Systems) that requests a class B address unless they can show a need for something close to 64K addresses, and instead giving them an appropriate number of class C addresses to cover the expected number of hosts. Since we would now be handing out address space in chunks of 256 addresses at a time, we could more accurately match the amount of address space consumed to the size of the AS.

For any AS with at least 256 hosts (which means the majority of ASs), we can guarantee an address utilization of at least 50%, and typically much more.

This solution, however, raises a problem that is at least as serious: excessive storage requirements at the routers.

- If a single AS has, say 16 class C network numbers assigned to it,
  - Every Internet backbone router needs 16 entries in its routing tables for that AS
  - This is true, even if the path to every one of these networks is the same
- If we had assigned a class B address to the AS
  - The same routing information can be stored in one entry
  - Efficiency =  $16 \times 255 / 65,536 = 6.2\%$

CIDR, therefore, tries to balance the desire to minimize the number of routes that a router needs to know against the need to hand out addresses efficiently.

To do this, CIDR helps us to *aggregate* routes.

- Uses a single entry in the forwarding table to tell the router how to reach a lot of different networks
- Breaks the rigid boundaries between address classes

To understand how this works, consider our hypothetical AS with 16 class C network numbers. Instead of handing out 16 addresses at random, we can hand out a block of *contiguous* class C addresses.

- Suppose we assign the class C network numbers from 192.4.16 through 192.4.31
- Observe that top 20 bits of all the addresses in this range are the same (11000000 00000100 0001)
  - We have created a 20-bit network number (which is in between class

B network number and class C number in terms of the number of hosts that it can support)

In other words, we get both the high address efficiency of handing out addresses in chunks smaller than a class B network and a single network prefix that can be used in forwarding tables.

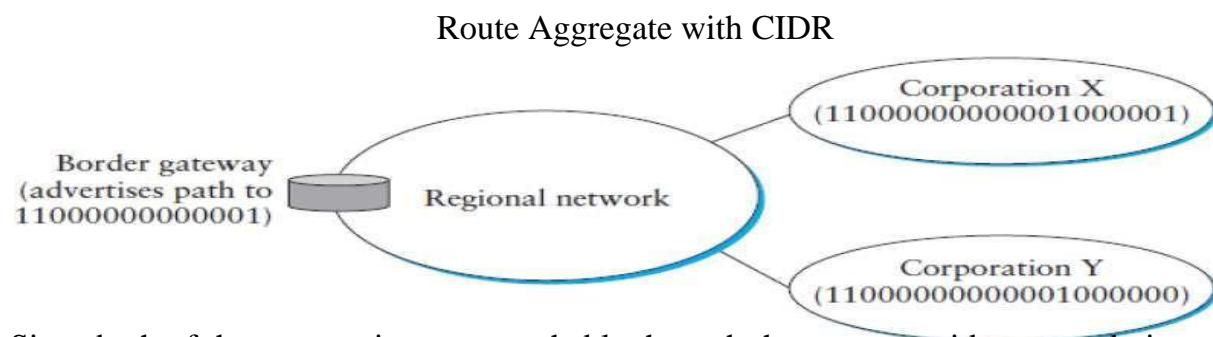
Observe that for this scheme to work, we need to hand out blocks of class C addresses that share a common prefix, which means that each block must contain a number of class C networks that is a power of two.

All we need now to make CIDR solve our problems is a routing protocol that can deal with these —classless addresses, which means that it must understand that a network number may be of any length.

Modern routing protocols do exactly that. The network numbers that are carried in such a routing protocol are represented simply by \_length, value\_ pairs, where the length gives the number of bits in the network prefix—20 in the above example.

- Requires to hand out blocks of class C addresses that share a common prefix
- The convention is to place a /X after the prefix where X is the prefix length in bits
- For example, the 20-bit prefix for all the networks 192.4.16 through 192.4.31 is represented as 192.4.16/20
- By contrast, if we wanted to represent a single class C network number, which is 24 bits long, we would write it 192.4.16/24
- How do the routing protocols handle this classless addresses
  - It must understand that the network number may be of any length
- Represent network number with a single pair  
**<length, value>**
- All routers must understand CIDR addressing

Consider the example Figure The two corporations served by the provider network have been assigned adjacent 20- bit network prefixes.



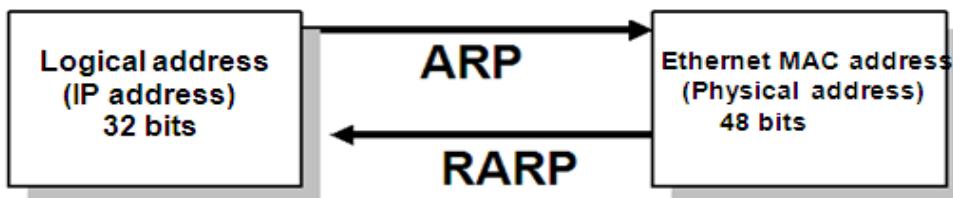
Since both of the corporations are reachable through the same provider network, it can advertise a single route to both of them by just advertising the common 19-bit prefix they share.

In general, it is possible to aggregate routes repeatedly if addresses are assigned carefully.

This means that we need to pay attention to which provider a corporation is attached to before assigning it an address if this scheme is to work. One way to accomplish that is to assign a portion of address space to the provider and then to let the network provider assign addresses from that space to its customers.

### ARP Address Resolution Protocol

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



#### ARP operation

Two operations are

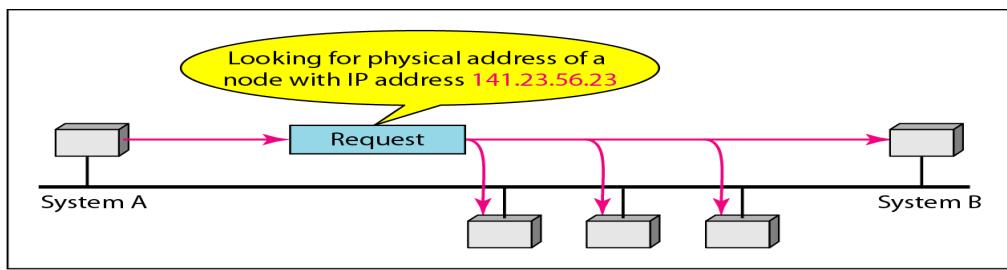
1. ARP request
2. ARP reply

#### ARP request

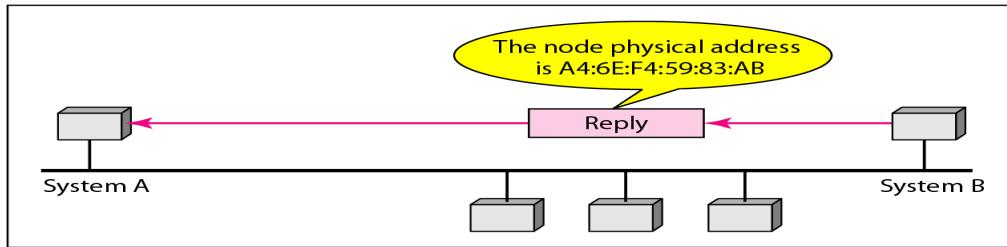
To determine the physical address for a particular IP, the node broadcasts an ARP query or ARP request containing that IP address in the network.

#### ARP reply

The node with that IP address responds to the sender of the ARP request with an ARP response or ARP reply containing its link-layer address or physical address,



a. ARP request is broadcast



b. ARP reply is unicast

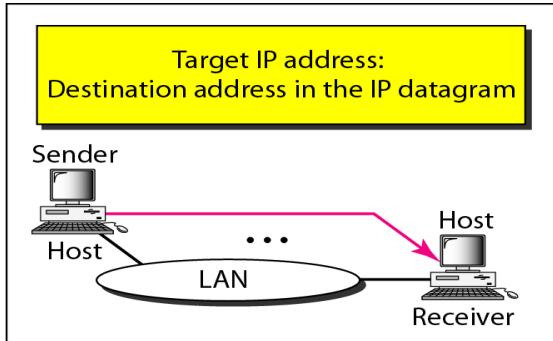
For providing mapping between logical address and physical address, each nodes maintain an ARP cache or ARP table. This table maintains an address pairs that map logical address to physical address.

### ARP Packet Format

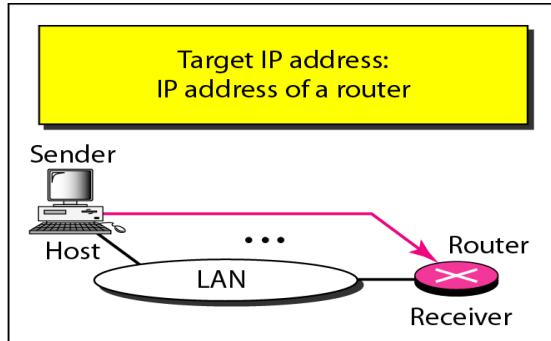
0	8	16	31
Hardware type = 1		ProtocolType = 0x0800	
HLen = 48	PLen = 32	Operation	
SourceHardwareAddr (bytes 0–3)			
SourceHardwareAddr (bytes 4–5)		SourceProtocolAddr (bytes 0–1)	
SourceProtocolAddr (bytes 2–3)		TargetHardwareAddr (bytes 0–1)	
TargetHardwareAddr (bytes 2–5)			
TargetProtocolAddr (bytes 0–3)			

- HardwareType: type of physical network (e.g., Ethernet)
- ProtocolType: type of higher layer protocol (e.g., IP)
- HLEN : length of physical addresses in bytes eg : ethernet – 6 bytes
- PLEN: length of logical addresses in bytes. (eg: IPv4 – 4 bytes)
- Operation: request or response
- Source/Target Physical/Protocol addresses

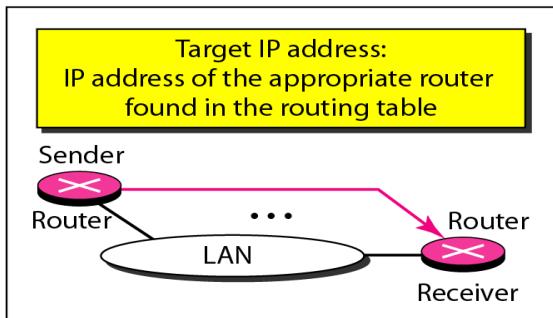
### Four cases using ARP



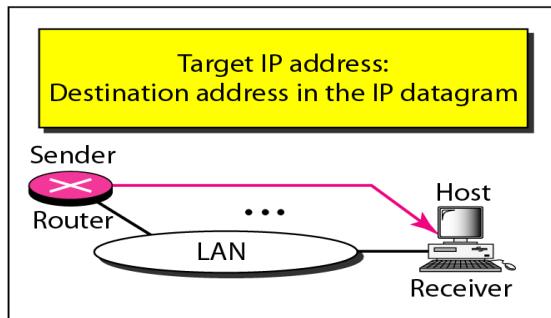
Case 1. A host has a packet to send to another host on the same network.



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



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



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

### RARP

Reverse Address Resolution Protocol (RARP)

Finds the logical address for a machine that knows only its physical address.

## **DHCP - Dynamic Host Configuration Protocol**

- For Mapping Physical to Logical Address can use RARP/BOOTP/DHCP
- DHCP server is responsible for providing configuration information to hosts
- DHCP is a protocol that dynamically assigns IP addresses to host.

### **Need for dynamic host configuration**

1. Ethernet addresses are configured into network by manufacturer and they are unique
2. IP addresses must be unique on a given internetwork but also must reflect the structure of the internetwork
3. Most host Operating Systems provide a way to manually configure the IP information for the host
4. Drawbacks of manual configuration
  - i. A lot of work to configure all the hosts in a large network
  - ii. Configuration process is error-prone
5. For this reason, automated Configuration Process is required
6. DHCP server is responsible for providing configuration information to hosts. There is at least one DHCP server for an administrative domain
7. DHCP server maintains a pool of available addresses.
8. DHCP is a protocol that dynamically assigns IP addresses to host.
9. DHCP relies on the existence of a DHCP server which is responsible for providing configuration information to host.
10. The use of DHCP avoids the network administrator from assigning addresses to individual host.

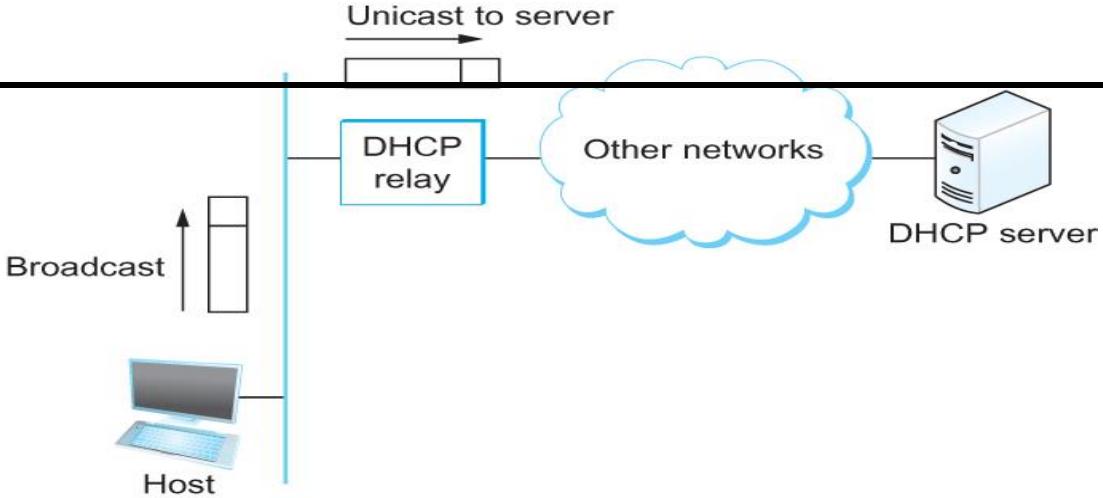
### **The Purpose of DHCP**

- Provide dynamic allocation of IP client configuration for a lease period
- Eliminate the work necessary to administer a large IP Network

The first problem faced in DHCP is server discovery. To avoid this, a host undergo the following activities,

1. Newly booted or attached host sends DHCPDISCOVER message to a special IP address (255.255.255.255) which is an IP broadcast address that is received by all hosts and routers on that network.
2. DHCP uses the concept of relay agent, there is at least one relay agent on each network.
3. When a relay agent receives a DHCPDISCOVER message, it unicasts it to DHCP server and awaits for the response, which is then sent back to the requesting client.

The process of relaying a message from a host to remote DHCP server is as follows,



**FIGURE : A DHCP relay agent receives a broadcast DHCPDISCOVERmessage from a host and sends a unicast DHCPDISCOVER to the DHCP server.**

### DHCP packet format

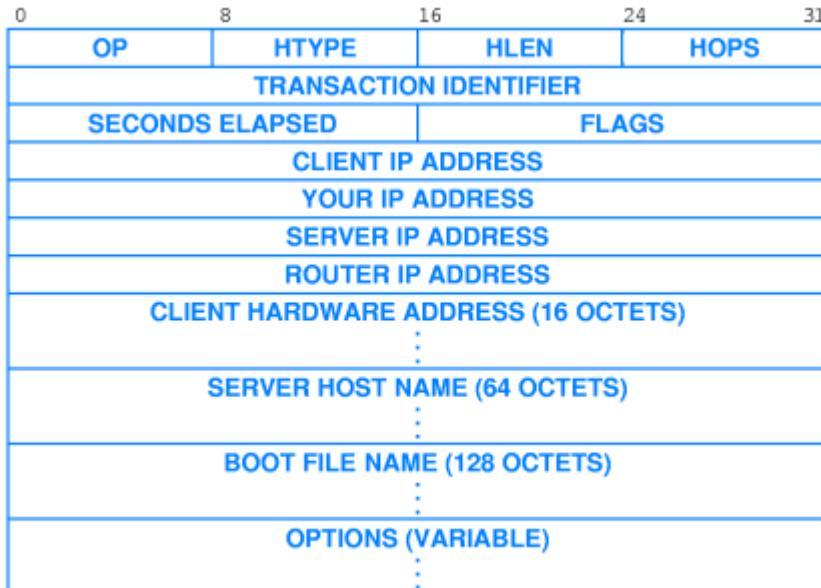
Figure shows the format of a DHCP message. The message is actually sent using a protocol called the User Datagram Protocol (UDP) that runs over IP. UDP does in this context is to provide a demultiplexing key that says, “This is a DHCP packet.”

DHCP is derived from an earlier protocol called BOOTP, and some of the packet fields are thus not strictly relevant to host configuration.

Operation	HType	HLen	Hops
Xid			
Secs		Flags	
ciaddr			
yiaddr			
siaddr			
giaddr			
chaddr (16 bytes)			
sname (64 bytes)			
file (128 bytes)			
options			

- Operation : 1 (*Request*), 2(*Reply*)
- Hardware Type: 1 (*for Ethernet*)
- Hardware address length: 6 (*for Ethernet*)
- Hop count: *set to 0 by client*
- Transaction ID: *Integer (used to match reply to response)*
- Seconds: *number of seconds since the client started to boot*
- ciaddr : client IP address
- yiaddr : your IP address
- siaddr : server IP address
- giaddr : gateway IP address

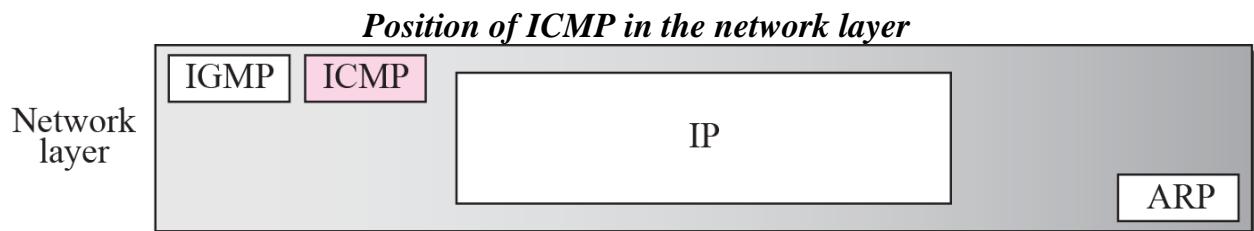
- chaddr : client's hardware address(Ethernet address)
- sname : server host name
- file : boot file name
- *client fills in the information that it has, leaves rest blank*



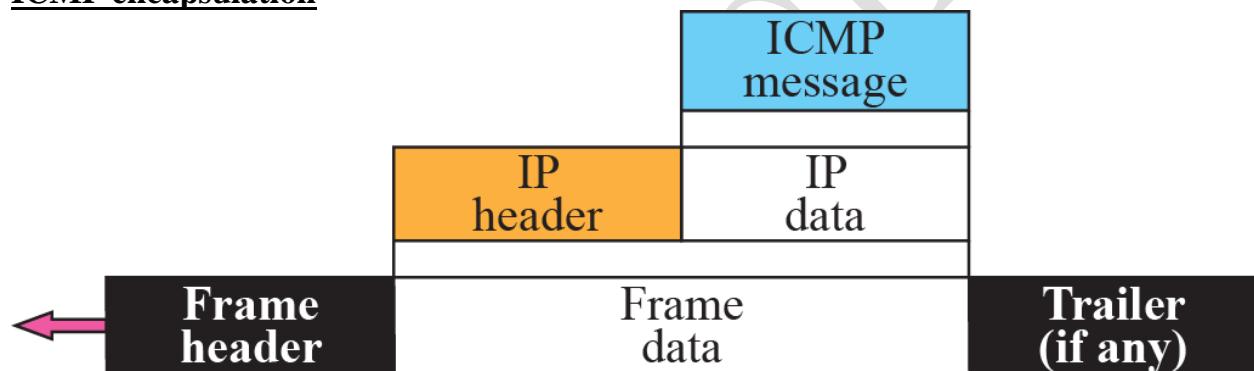
## **ICMP (Internet Control Message Protocol)**

ICMP is used for error and control information

- Defines a collection of error messages that are sent back to the source host whenever a router or host is unable to process an IP datagram successfully
- ***ICMP always reports error messages to the original source.***



### **ICMP encapsulation**



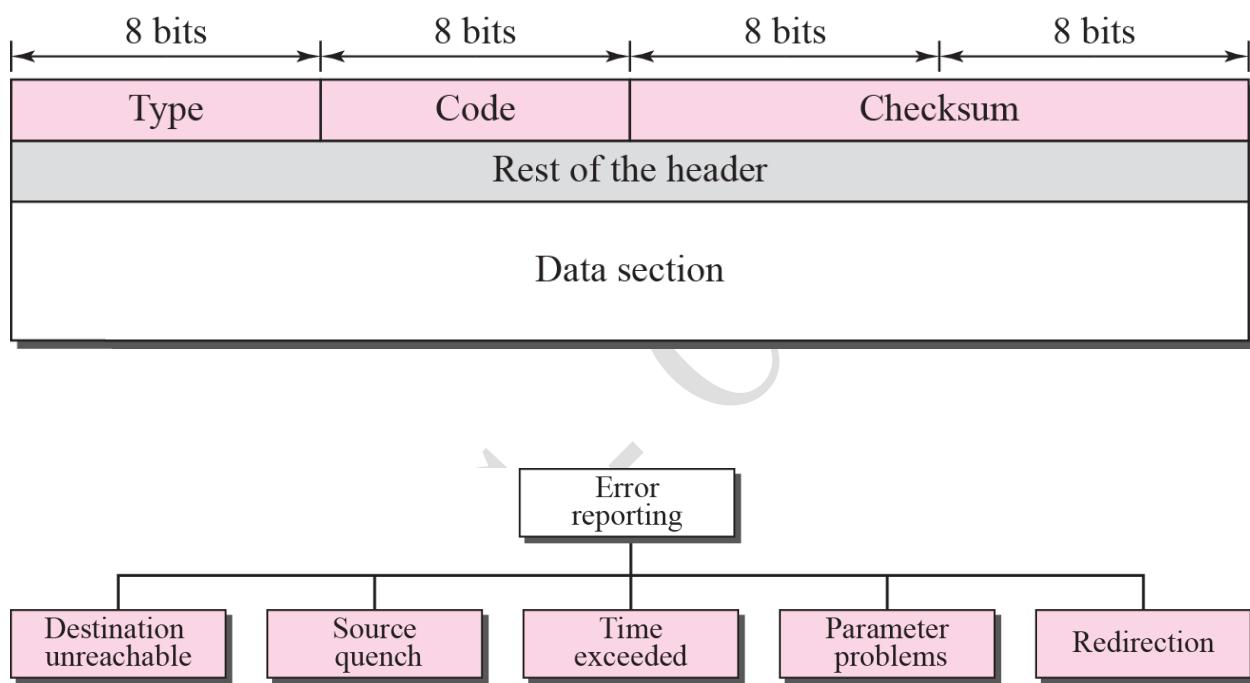
**ICMP messages are divided into two broad categories:**

1. error-reporting messages
  - The error-reporting messages report problems that a router or a host (destination) may encounter when it processes an IP packet.
2. query messages.
  - The query messages, which occur in pairs, help a host or a network manager get specific information from a router or another host.

*ICMP messages*

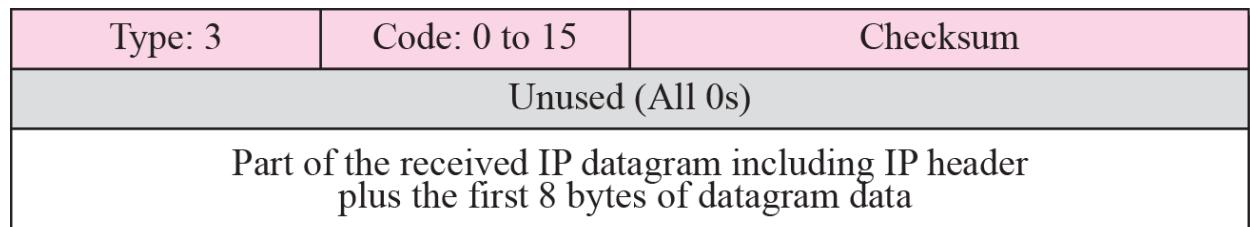
Category	Type	Message
Error-reporting messages	3	Destination unreachable
	4	Source quench
	11	Time exceeded
	12	Parameter problem
	5	Redirection
Query messages	8 or 0	Echo request or reply
	13 or 14	Timestamp request or reply

**General format of ICMP messages**



**Destination-unreachable**

When a router cannot route a datagram or a host cannot deliver a datagram, the datagram is discarded and the router or the host sends a destination-unreachable message back to the source host that initiated the datagram. Note that destination-unreachable messages can be created by either a router or the destination host.



**Source Quench**

A source-quench message informs the source that a datagram has been discarded due to congestion in a router or the destination host. The source must slow

down the sending of datagram's until the congestion is relieved.

Type: 4	Code: 0	Checksum
Unused (All 0s)		
Part of the received IP datagram including IP header plus the first 8 bytes of datagram data		

### **Time Exceeded**

When the final destination does not receive all of the fragments in a set time, it discards the received fragments and sends a time-exceeded message to the original source.

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

### **Parameter-problem**

Any ambiguity in the header part of a datagram can Create serious problems as the datagram travels through the Internet. If a router or the destination host discovers an ambiguous or missing value in any field of the datagram, it discards the datagram and sends a parameter-problem message back to the source.

Type: 12	Code: 0 or 1	Checksum
Pointer	Unused (All 0s)	
Part of the received IP datagram including IP header plus the first 8 bytes of datagram data		

### **Redirection**

ICMP Redirect is sent by a router (R1) to the sender of an IP datagram (host) when the datagram should have been sent to a different router (R2)

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

### **Echo-request and echo-reply**

- Echo-request and echo-reply messages can be used by network managers to check the operation of the IP protocol.
- Echo-request and echo-reply messages can test the reachability of a host. This

is usually done by invoking the ping command.

Type 8: Echo request

Type 0: Echo reply

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

### **Timestamp-request and timestamp-reply message**

Timestamp-request and timestamp-reply messages can be used to calculate the round-trip time between a source and a destination machine even if their clocks are not synchronized.

Type 13: request

Type 14: reply

Type: 13 or 14	Code: 0	Checksum
Identifier		Sequence number
Original timestamp		
Receive timestamp		
Transmit timestamp		

### **Virtual Networks and Tunnels**

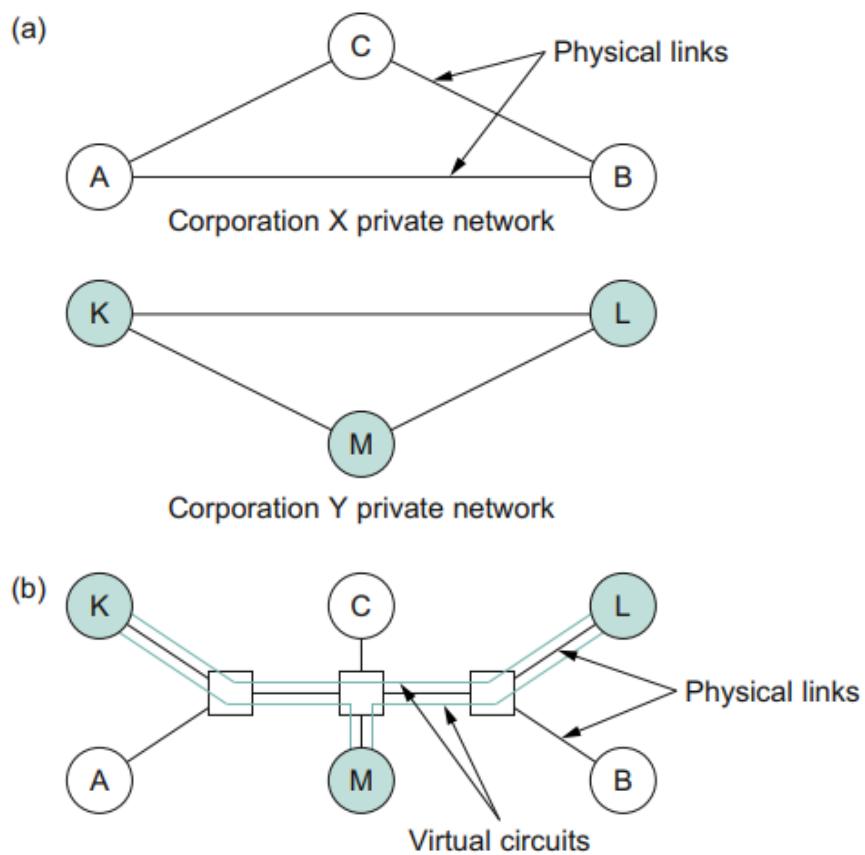
There are many situations where more controlled connectivity is required. An important example of such a situation is the virtual private network (VPN).

The term VPN is heavily overused and definitions vary, but intuitively we can define a VPN by considering first the idea of a private network. Corporations with many sites often build private networks by leasing transmission lines from the phone companies and using those lines to interconnect sites.

In such a network, communication is restricted to take place only among the sites of that corporation, which is often desirable for security reasons. To make a private network virtual, the leased transmission lines—which are not shared with any other corporations—would be replaced by some sort of shared network. A virtual circuit (VC) is a very reasonable replacement for a leased line because it still provides a logical point-to-point connection between the corporation's sites.

For example, if corporation X has a VC from site A to site B, then clearly it can send packets between sites A and B. But there is no way that corporation Y can get its packets delivered to site B without first establishing its own virtual circuit to site B, and the establishment of such a VC can be administratively prevented, thus preventing unwanted connectivity between corporation X and corporation Y.

Figure 3.26(a) shows two private networks for two separate corporations. In Figure 3.26(b) they are both migrated to a virtual circuit network.



**FIGURE 3.26** An example of virtual private networks: (a) two separate private networks; (b) two virtual private networks sharing common switches.

The limited connectivity of a real private network is maintained, but since the private networks now share the same transmission facilities and switches we say that two virtual private networks have been created. In Figure 3.26, a virtual circuit network (using Frame Relay or ATM, for example) is used to provide the controlled connectivity among sites. It is also possible to provide a similar function using an IP network—an internetwork—to provide the connectivity.

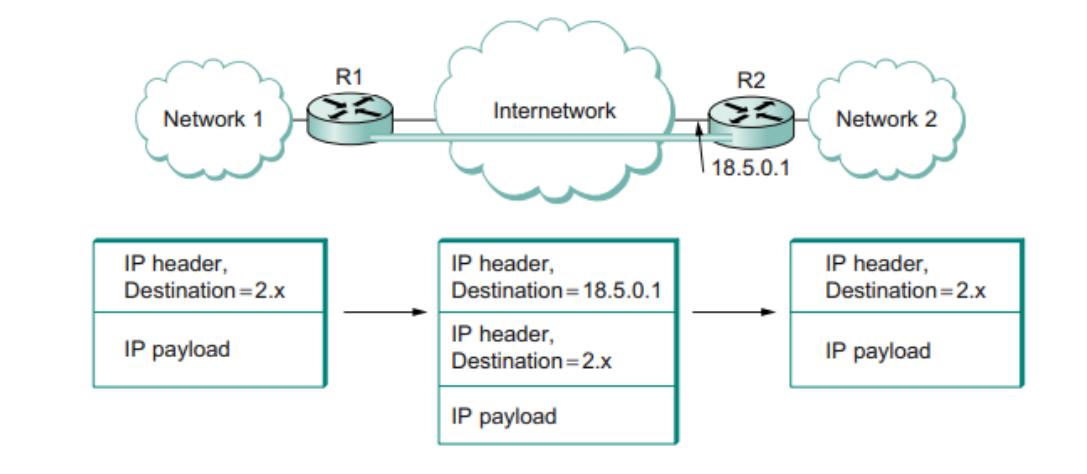
However, we cannot just connect the various corporations' sites to a single internetwork because that would provide connectivity between corporation X and

corporation Y, which we wish to avoid. To solve this problem, we need to introduce a new concept, the IP tunnel.

We can think of an IP tunnel as a virtual point-to-point link between a pair of nodes that are actually separated by an arbitrary number of networks. The virtual link is created within the router at the entrance to the tunnel by providing it with the IP address of the router at the far end of the tunnel.

Whenever the router at the entrance of the tunnel wants to send a packet over this virtual link, it encapsulates the packet inside an IP datagram. The destination address in the IP header is the address of the router at the far end of the tunnel, while the source address is that of the encapsulating router.

In the forwarding table of the router at the entrance to the tunnel, this virtual link looks much like a normal link. Consider, for example, the network in Figure 3.27. A tunnel has been configured from R1 to R2 and assigned a virtual interface number of 0. The forwarding table in R1 might therefore look like Table 3.8



**FIGURE 3.27** A tunnel through an internetwork. 18.5.0.1 is the address of R2 that can be reached from R1 across the internetwork.

**Table 3.8 Forwarding Table for Router R1 in Figure 3.27**

NetworkNum	NextHop
1	Interface 0
2	Virtual interface 0
Default	Interface 1

R1 has two physical interfaces. Interface 0 connects to network 1; interface 1 connects to a large internetwork and is thus the default for all traffic that does not match something more specific in the forwarding table. In addition, R1 has a virtual interface, which is the interface to the tunnel. Suppose R1 receives a packet from network 1 that contains an address in network 2. The forwarding table says this packet should be sent out virtual interface 0. In order to send a packet out this interface, the router takes the packet, adds an IP header addressed to R2, and then proceeds to forward the packet as if it had just been received. R2's address is 18.5.0.1; since the network number of this address is 18, not 1 or 2, a packet destined for R2 will be forwarded out the default interface into the internetwork.

Once the packet leaves R1, it looks to the rest of the world like a normal IP packet destined to R2, and it is forwarded accordingly. All the routers in the internetwork forward it using normal means, until it arrives at R2. When R2 receives the packet, it finds that it carries its own address, so it removes the IP header and looks at the payload of the packet. What it finds is an inner IP packet whose destination address is in network 2. R2 now processes this packet like any other IP packet it receives. Since R2 is directly connected to network 2, it forwards the packet on to that network. Figure 3.27 shows the change in encapsulation of the packet as it moves across the network.

#### Tunneling does have its downsides.

One is that it increases the length of packets; this might represent a significant waste of bandwidth for short packets. Longer packets might be subject to fragmentation, which has its own set of drawbacks. There may also be performance implications for the routers at either end of the tunnel, since they need to do more work than normal forwarding as they add and remove the tunnel header

Finally, there is a management cost for the administrative entity that is responsible for setting up the tunnels and making sure they are correctly handled by the routing protocols.

REC-CSE