Objective of the Course:

:

The student should be made to:

- Understand the division of network functionalities into layers.
- Be familiar with the components required to build different types of networks
- Be exposed to the required functionality at each layer
- Learn the flow control and congestion control algorithms

Syllabus

UNIT I FUNDAMENTALS & LINK LAYER 9

Building a network –Requirements -Layering and protocols –Internet Architecture – Network software - Performance Link layer Services -Framing - Error Detection -Flow control

UNIT II MEDIA ACCESS & INTERNETWORKING 9

Media access control -Ethernet (802.3) -Wireless LANs –802.11 –Bluetooth - Switching and bridging –Basic Internetworking (IP,CIDR, ARP, DHCP,ICMP)

UNIT III ROUTING 9

Routing (RIP, OSPF, metrics) – Switch basics –Global Internet (Areas, BGP, IPv6), Multicast – addresses – multicast routing (DVMRP, PIM)

UNIT IV TRANSPORT LAYER 9

Overview of Transport layer - UDP -Reliable byte stream (TCP) - Connection management -Flow control -Retransmission –TCP Congestion control -Congestion avoidance (DECbit, RED) –QoS –Application requirements

UNIT V APPLICATION LAYER 9

Traditional applications -Electronic Mail (SMTP, POP3, IMAP, MIME) – HTTP –Web Services –DNS -SNMP

TOTAL: 45 PERIODS

OUTCOMES:

- At the end of the course, the student should be able to:
- Identify the components required to build different types of networks
- Choose the required functionality at each layer for given application
- Identify solution for each functionality at each layer
- Trace the flow of information from one node to another node in the network

EC8551-COMMUNICATION NETWORKS

UNIT I FUNDAMENTALS & LINK LAYER : 9

Building a network – Requirements - Layering and protocols - Internet Architecture – Network software – Performance ; Link layer Services - Framing - Error Detection - Flow control

UNIT II MEDIA ACCESS & INTERNETWORKING : 9

Media access control - Ethernet (802.3) - Wireless LANs – 802.11 – Bluetooth - Switching and bridging – Basic Internetworking (IP, CIDR, ARP, DHCP,ICMP)

UNIT III ROUTING : 9

Routing (RIP, OSPF, metrics) – Switch basics – Global Internet (Areas, BGP, IPv6), Multicast – addresses – multicast routing (DVMRP, PIM)

UNIT IV TRANSPORT LAYER : 9

Overview of Transport layer – UDP - Reliable byte stream - Connection management - Flow control - Retransmission – TCP Congestion control - Congestion avoidance (DECbit, RED) – QoS – Application requirements

TEXT BOOK:

1. Larry L. Peterson, Bruce S. Davie, “Computer Networks: A Systems Approach”, Fifth Edition, Morgan Kaufmann Publishers, 2011

REFERENCES:

1. James F. Kurose, Keith W. Ross, “Computer Networking - A Top-Down Approach Featuring the Internet”, Fifth Edition, Pearson Education, 2009.

2. Nader. F. Mir, “Computer and Communication Networks”, Pearson Prentice Hall Publishers, 2010.

3. Ying-Dar Lin, Ren-Hung Hwang, Fred Baker, “Computer Networks: An Open Source Approach”, Mc Graw Hill Publisher, 2011.

4. Behrouz A. Forouzan, “Data communication and Networking”, Fourth Edition, Tata McGraw – Hill, 2011. E

UNIT I FUNDAMENTALS & LINK LAYER

UNIT II MEDIA ACCESS & INTERNETWORKING

UNIT III ROUTING

UNIT IV TRANSPORT LAYER

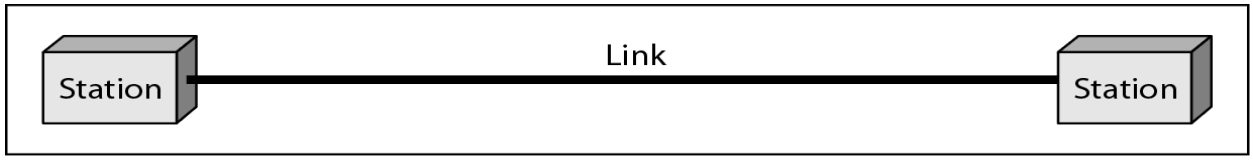
UNIT V APPLICATION LAYER :

UNIT-I FUNDAMENTALS & LINK LAYER

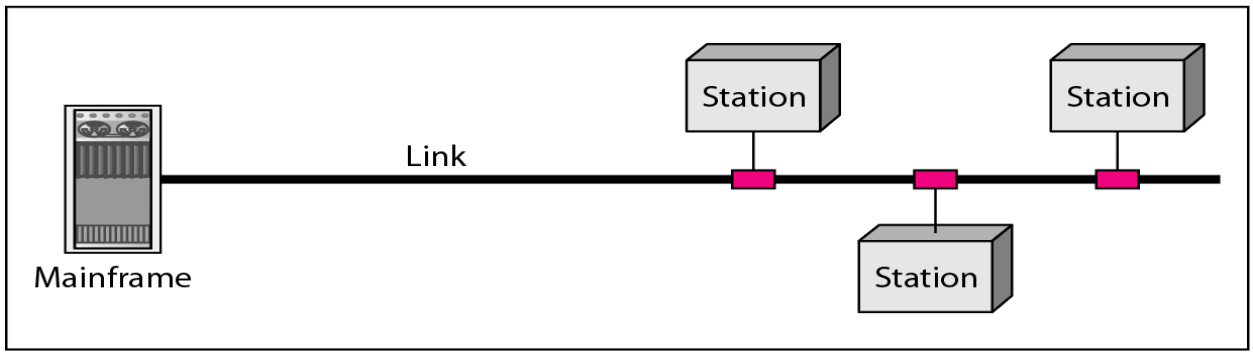
Building a network – Requirements - Layering and protocols - Internet Architecture – Network software – Performance ; Link layer Services - Framing - Error Detection - Flow control.

INTRODUCTION:

- Number of computers are interconnected to form a computer network. Usually computer communication is the mostly preferred because of it's high speed and low cost.
- **Type of connection :**There are two possible type of connections Point-to-point and Multipoint
- A **point-to-point** connection provides a dedicated link between two devices. The entire link is reserved for transmission between those two devices. Ex. Change of television channel by infrared remote control.
- A **multipoint** (also called multidrop) connection is one in which more than two specific devices share a single link.



a. Point-to-point

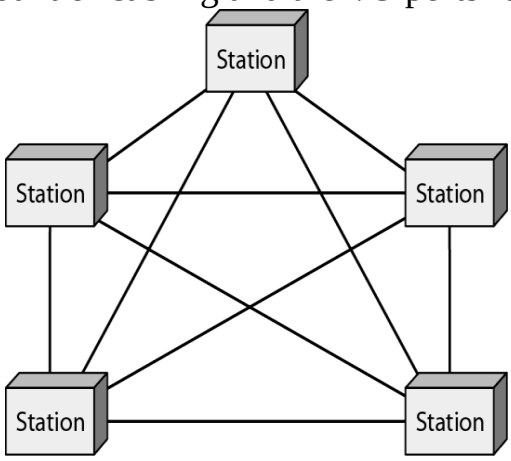


b. Multipoint

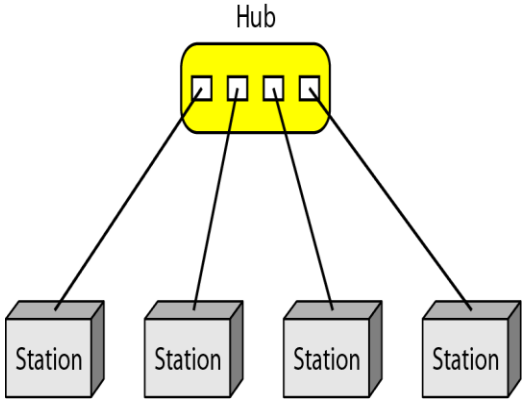
- **Physical Topology :**Physical Topology refers to the way in which the hosts are connected.
- The basic topologies are

Mesh
Star
Bus and
Ring

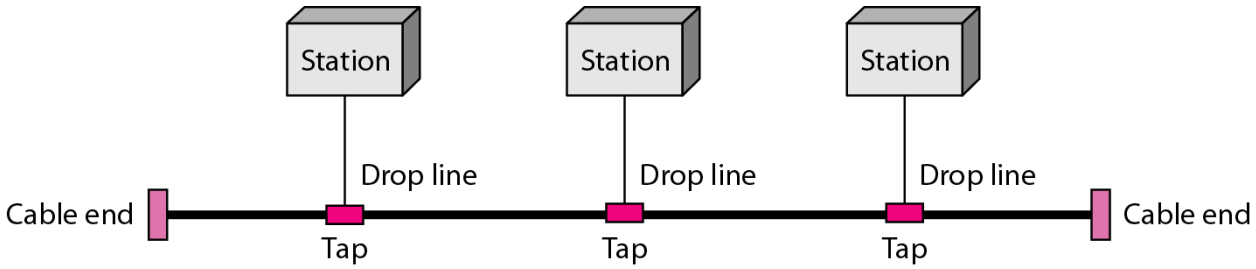
- **Mesh :**In a mesh topology each device has a dedicated point to point link to every other device. **Merits:** It eliminates the traffic problems. **Demerits:** The amount of cabling and the I/O ports required is more



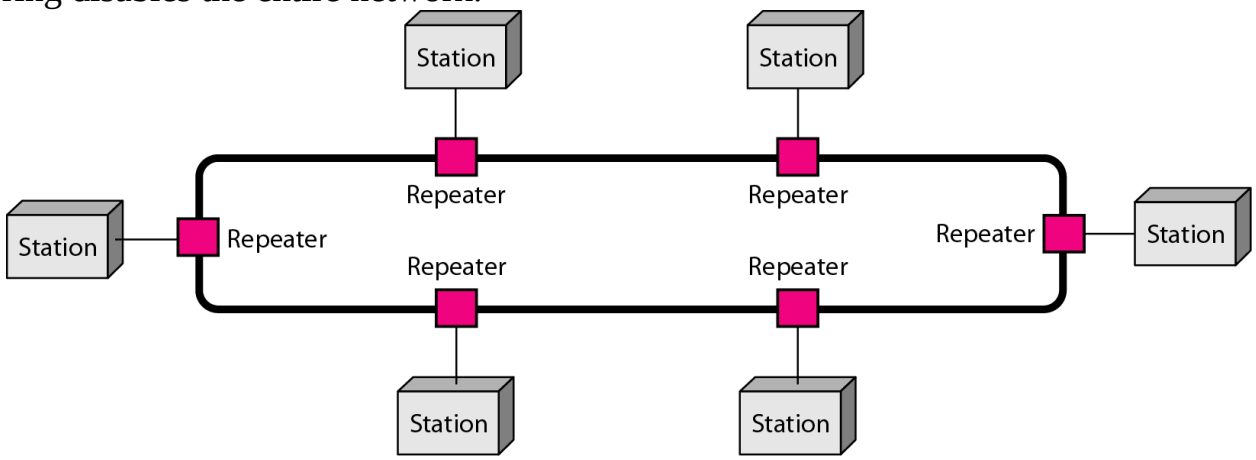
- **Star topology** Each device has a dedicated point to point link only to a central controller usually called a hub. If one device has to send data to another it sends the data to the controller, which then sends the data to the other connected device. **Merits :**Less expensive than a mesh topology. Installation and reconfigure is easy. **Demerits** Require more cable compared to bus and ring topologies.



- **Bus :**One long cable acts as a backbone to link all the devices in a network **merits :**Ease of installation. Bus uses less cabling than mesh or star topologies. **Demerits:** Difficult reconnection and isolation.



- **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 **Merits:** Easy to install and reconfigure. **Demerits** A break in the ring disables the entire network.



1.1.BUILDING A NETWORK:

Computer Networks means an interconnected network of computers. Computer networking support variety of applications like teleconferencing, video-on demand, digital library etc..

Computer networks are built primarily from general purpose programmable hardware ie) they are not dedicated for a particular application. They can carry many different types of data , supporting wide range of applications.

For building a network we need to know the following,

1.Requirements that different from application to applications .

2.The Network architecture .

3.How to implement a network with the key elements

4.Evaluation of performance.

1.2. REQUIREMENTS:

For knowing the requirements of the new applications we should know the following concepts.

1.2.1. Perspectives:

Requirement may be different for different persons, based on this we divide this in to three,

Requirement of an application programmer:

He need the service that his application need. Example if a message is send , he need the acknowledgement whether it is reached or not.

Requirement of an operator:

His need is, it should be easy to administer and manage the systems. Example fault identification in his network.

Requirement of a network designer:

List the properties of cost effective design.

1.2.2.Scalable connectivity:

A network is used to connect a set of computers. Some networks connect only a few machines. Some networks connect large number of devices (Internet).

Link: A network having two or more computers can be connected by using co-axial cables or with optical cable. This physical medium is called as link.

Nodes: The computers that are connected to the link are called as nodes.

When two nodes are connected by a physical medium it is called point-to-point link ,When more than two nodes share a medium it is called multiple-access

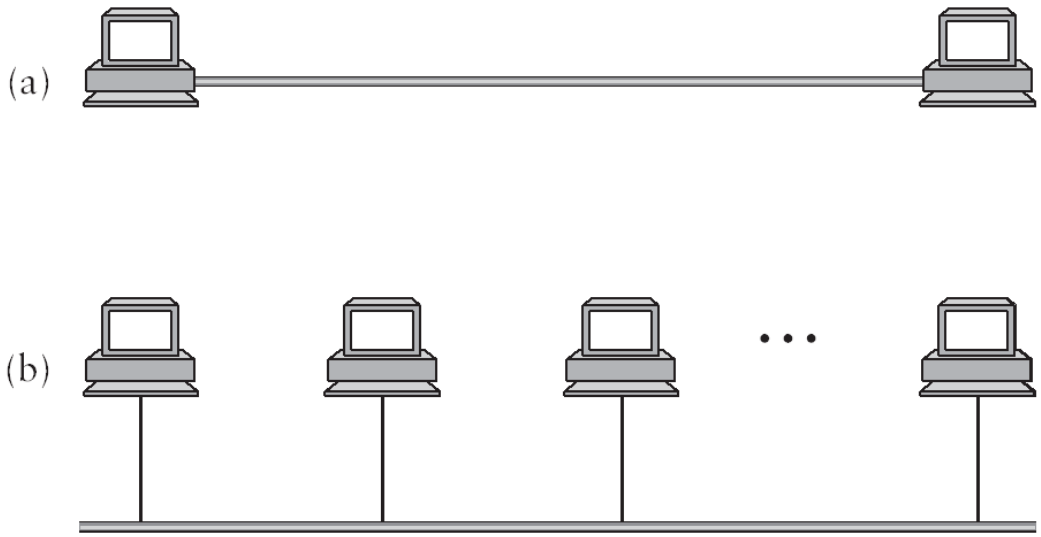


Fig: **1.1.a) point-to-point link b) multiple access link**

Data communication between the nodes is done by forwarding the data from one link to another. This method of forwarding data between nodes form a ***switched network*.**

Two common types of switched network are

- Circuit switched – e.g. Telephone System
- Packet switched – e.g. Postal System

Packet Switched Network

In this network nodes send discrete blocks of data to each other. These blocks can be called as ***packet or message.*** This network follows **Store and forward strategy** It means Each node receives a complete packets through the link, stores in internal memory and then forwards to next node.

Circuit Switched Network

It first establishes a circuit across the links and allows source node to send stream of bits through this circuit to the destination node .The representation of network is given by cloud symbol.

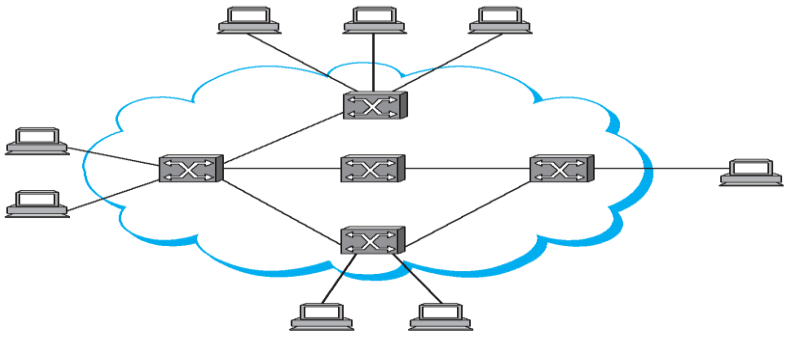


Fig 1.2.Switched network

In the figure ,
Nodes inside the cloud (Switches) – Implement the network
Nodes outside the cloud (host) - Use the network

- **Internetwork :** Set of independent network are interconnected to form inter network or **internet.**
- Node that is connected to two or more network is called **router or gateway.** It is responsible for forwarding data between the networks.
- **Addressing :** Each node want to say to which other node it want to communicate. This is done by assigning address to each node. when a source node wants to deliver message to destination node, it specifies the address of destination node.
- Switches and Routers use this address to decide how to forward the message. This process based on address is called **Routing**.
- **Unicast** – sending message to single node.
- **Broadcast** – Sending message to all the nodes on the network.
- **Multicast** – Sending message to some subnet not to all.

1.2.3.Cost-effective resource sharing:

System resources are shared among multiple users by multiplexing .
Methods:
1. Synchronous Time Division Multiplexing(STDM) - Divide time into equal sized slots.
2. Frequency Division Multiplexing (FDM) :Transmit each flow at different frequency
Drawbacks of First two methods are

- If one user does not have data to send then its time slot will be wasted while the others have data to transmit.
- No of frequency channels allocated are fixed and it cannot be resized.

3. Statistical Multiplexing: Statistical methods combine the ideas of both STDM and FDM

- Data from each user is transmitted based on demand so no wastage of time slots or frequency.
- It defines upper bound on size of data and it is referred as packet.

1.2.4.Support for common services:

Communication between a pair of processes is done by request / reply basis. The process which sends request is referred as client and the one which honors the request is referred as server.

This can be done using channels. Two types of channels are

- Request / Reply channels
- Message stream Channels

Reliability: To get the reliable network, it is necessary to find how network fails.

Three classes of failures

- Bit error
- Packet loss
- Physical link and node failure

1.2.5. Manageability:

Managing a network includes making change to the network as new nodes are added to the network. The changes may be increased traffic, troubleshooting etc

1.3. NETWORK ARCHITECTURE:

Network architecture helps in design and implementation of network. Two commonly used architecture are,

- OSI Architecture
- Internet or TCP/IP architecture

1.3.1 LAYERING AND PROTOCOL:

Layering provides two features.

- It reduces the problem of building a network into more manageable components.
- It provides more modular design. To add some new service, it is enough to modify the functionality at one layer, reusing the functions provided at all the other layers.
- Layer is the hardware and software that implement the protocol.

Protocols:

- A protocol is a set of rules that governs data communication. It defines what is communicated, how it is communicated, and when it is communicated.
- The key elements of a protocol are syntax, semantics and timing.
- Each protocol defines two different interfaces.
  Service interface – It is used by other objects on the same computer.
  Peer interface – It is used by peer on another machine.

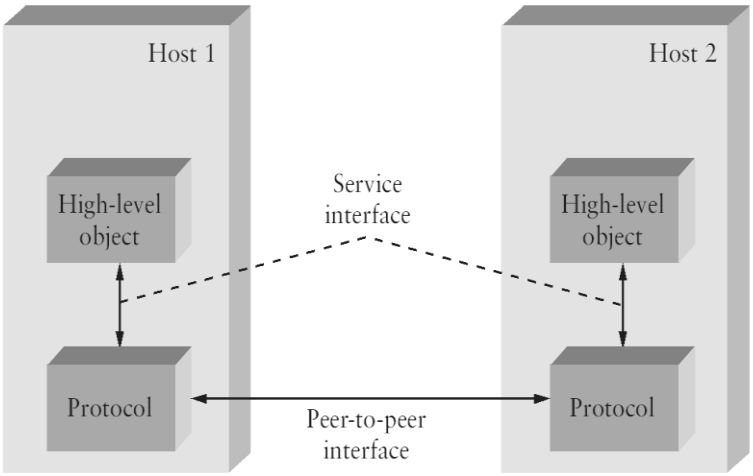


Fig:1.3. Service and peer interfaces.

- Peer-to- peer communication is direct only when they communicate directly with the hardware.
- Peer to peer communication is in-direct when one system communicate with other by passing message to some lower level protocols.
- There are multiple protocols present at every level ,each providing different communication service.
- The protocols that are used for making a network are represented by a protocol graph.

- A protocol graph is shown below. For example, the host 1 wants to send a message to its host 2 .
- In this case, the file application asks RRP to send the message .
- RRP then uses the services of HHP, which transmits the message to its peer on the other machine.
- Once the message has arrived at protocol HHP on host 2, HHP passes the message up to RRP, which in turn delivers the message to the file application

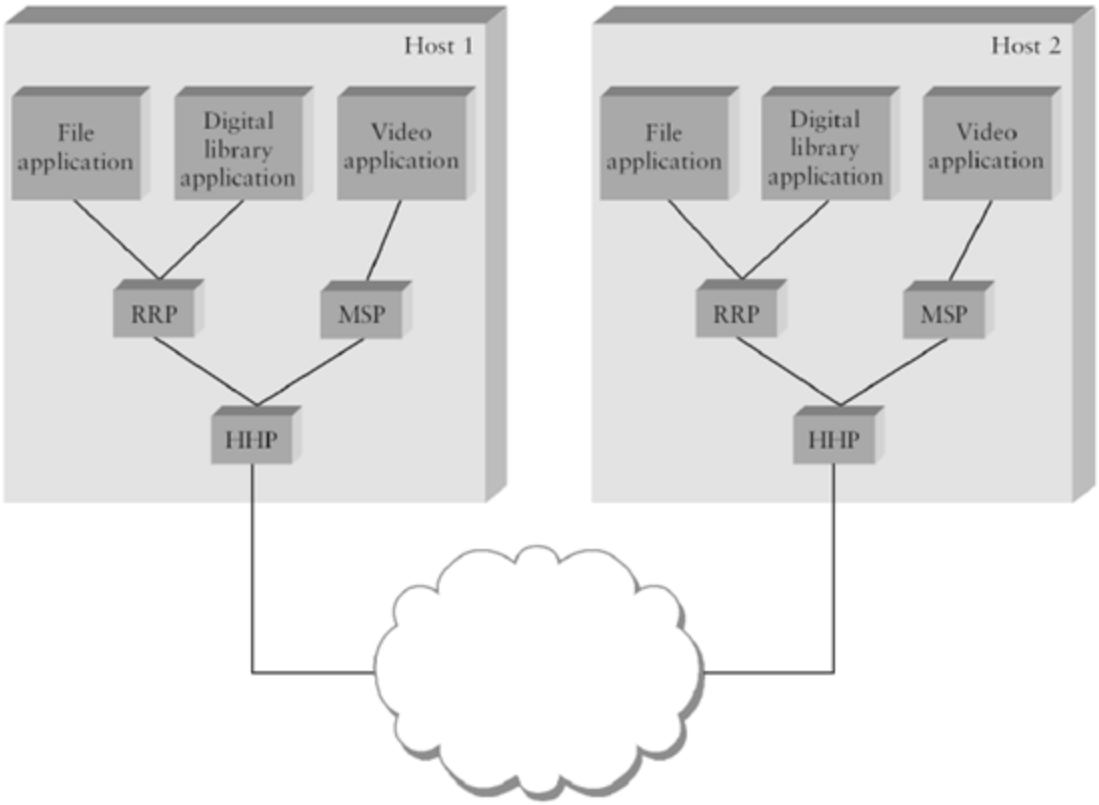


Fig:1.4 Example of a protocol graph.

Encapsulation:

Control information must be added with the data to instruct the peer how to handle the received message. It will be added into the header or trailer.

Header – Information will be added to the front of message.
Trailer – Information will be added at the end of the message
Payload or message body – Data send by the program

In this case data is encapsulated with new message created by protocol at each level.

For example if host 1 want to communicate with host 2 ,the application program will send the data to the RRP. The RRP does not bother what sort of information is the data. The RRP attach its header with the data and transmit it to the HHP. The HHP attach a header with the RRP's message . Then HHP sends the message to its peer over some network, when the message arrives at the destination host, it is processed in the opposite order. This is illustrated in the below figure.

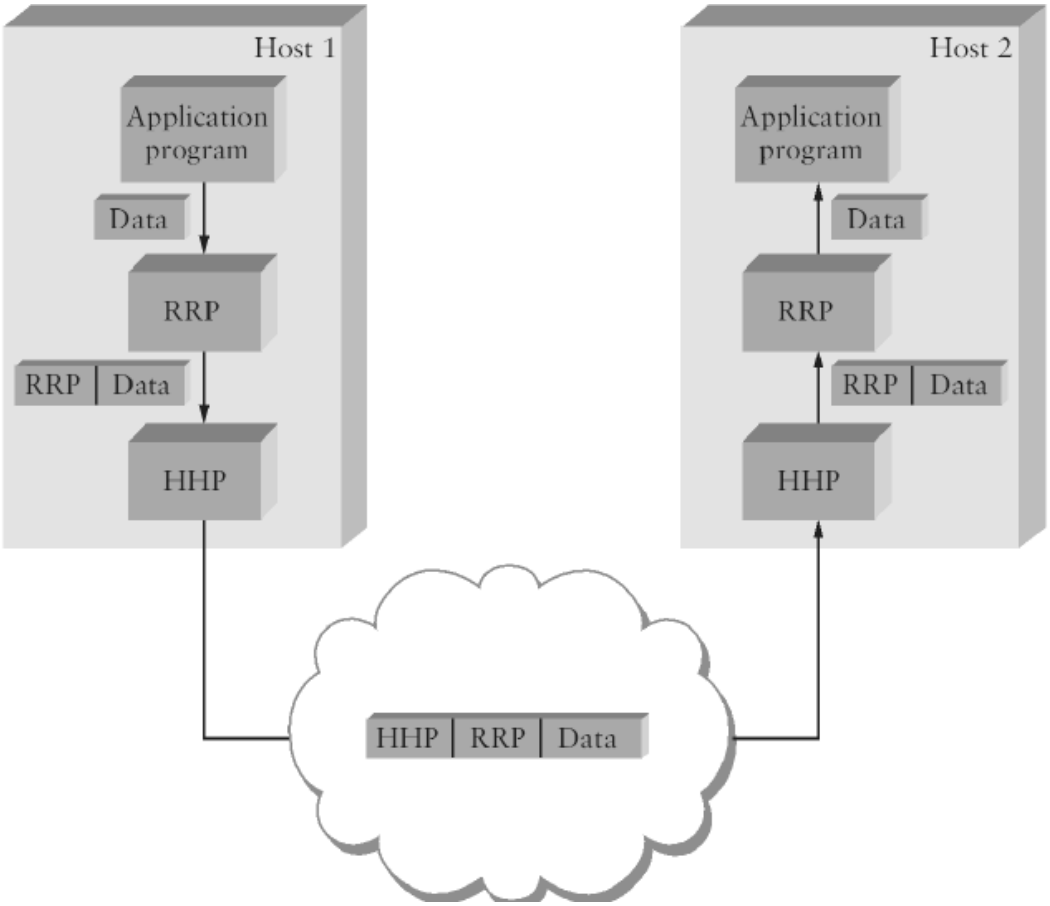


Fig:1.5. High-level messages are encapsulated inside of low-level messages.

Multiplexing and De-Multiplexing:

The fundamental idea of packet switching is to multiplex data over a single physical link. This can be achieved by adding identifier to the header message The messages are de-multiplexed at the destination side.

OSI Architecture:

ISO defines a common way to connect computer by the architecture called Open System Interconnection(OSI) architecture. Network functionality is divided into seven layers.

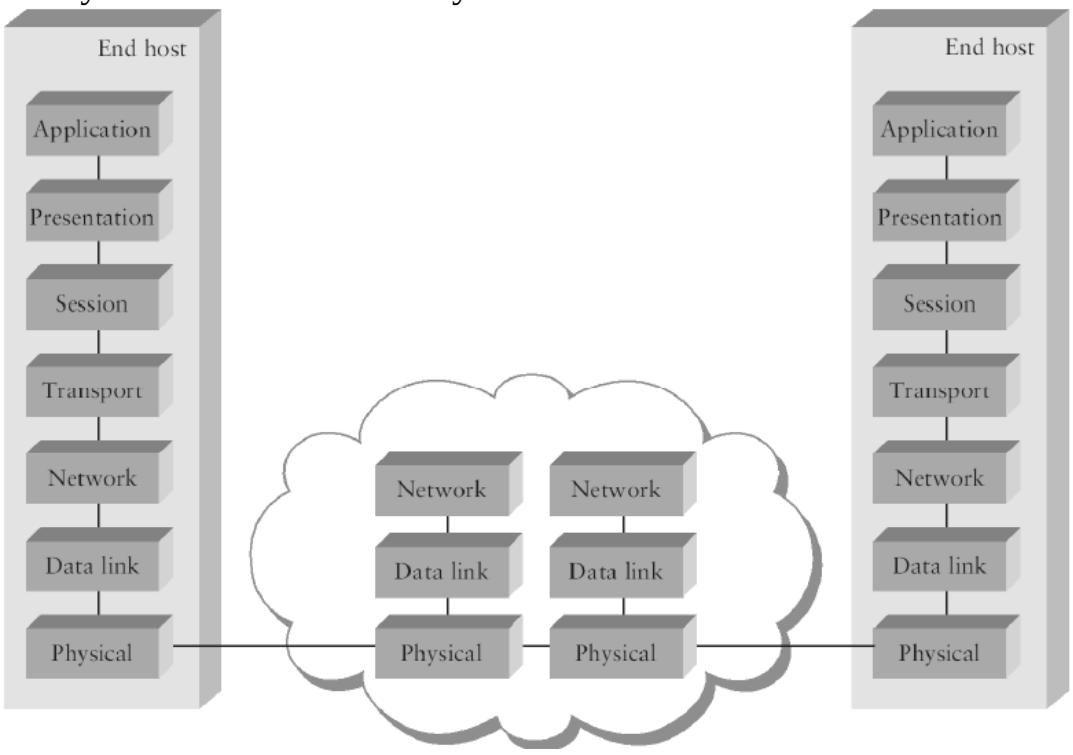


Fig :1.6.OSI model

Organization of the layers:

The 7 layers can be grouped into 3 subgroups,

1. Network Support Layers:

Layers 1,2,3 - Physical, Data link and Network are the network support layers.

2. Transport Layer:

Layer4, transport layer, ensures end-to-end reliable data transmission on a single link.

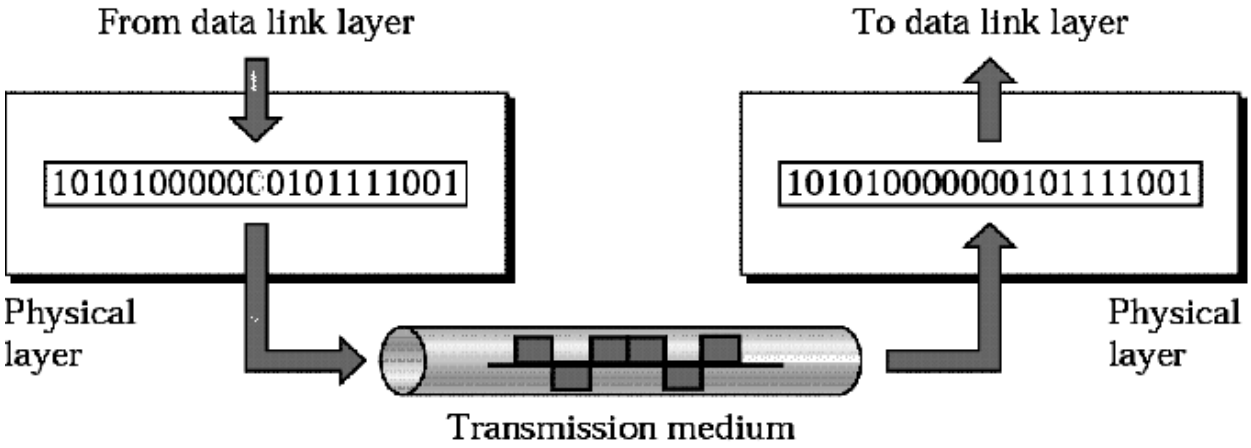
3. User Support Layers

Layers 5,6,7 – Session, presentation and application are the user support layers. They allow interoperability among unrelated software systems.

Functions of the Layers:

1. Physical Layer

The physical layer coordinates the functions required to transmit a bit stream over a physical medium.

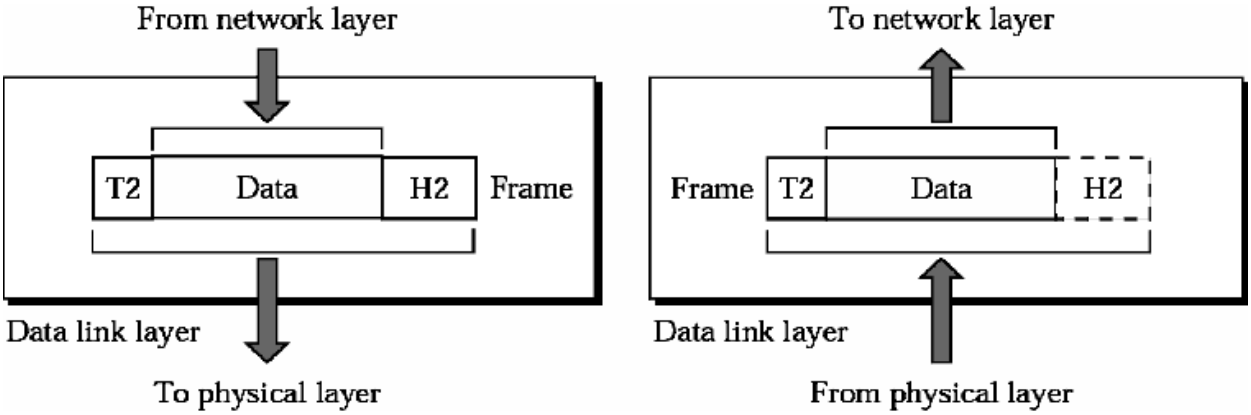


The physical layer is concerned with the following:

- **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:

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



The other responsibilities of this layer are

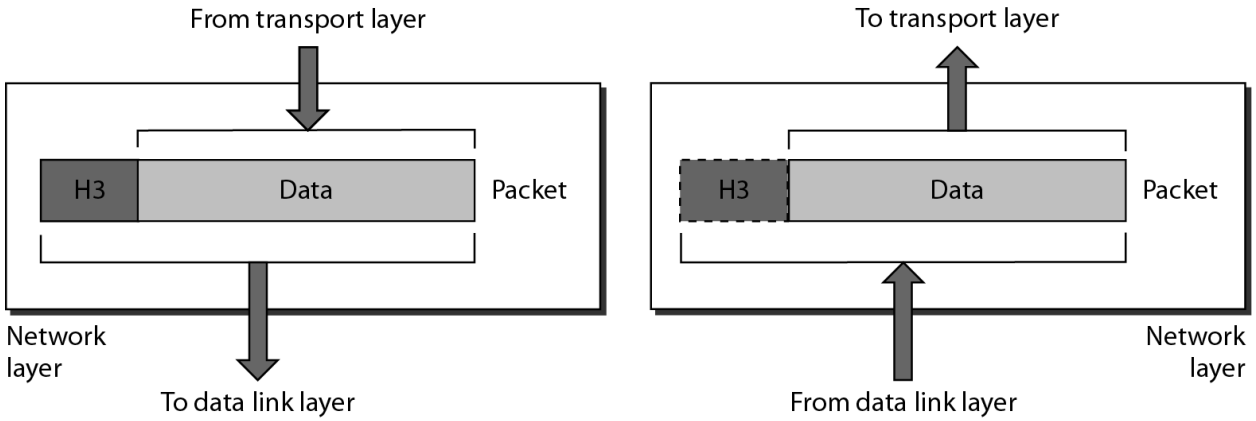
- **Framing -** Divides the stream of bits received into data units called frames.
- **Physical addressing** – If frames are to be distributed to different systems on the n/w , data link layer adds a header to the frame to define the sender and receiver.
- **Flow control**- If the rate at which the data are absorbed by the receiver is less than the rate produced in the sender ,the Data link layer imposes a flow ctrl 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:

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.

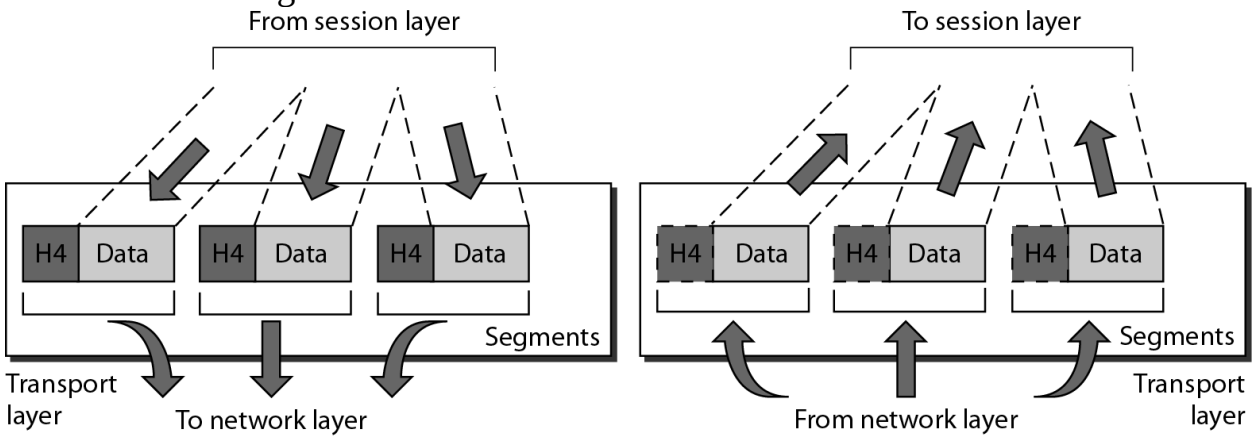
The other responsibilities of this layer are,

- **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:

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

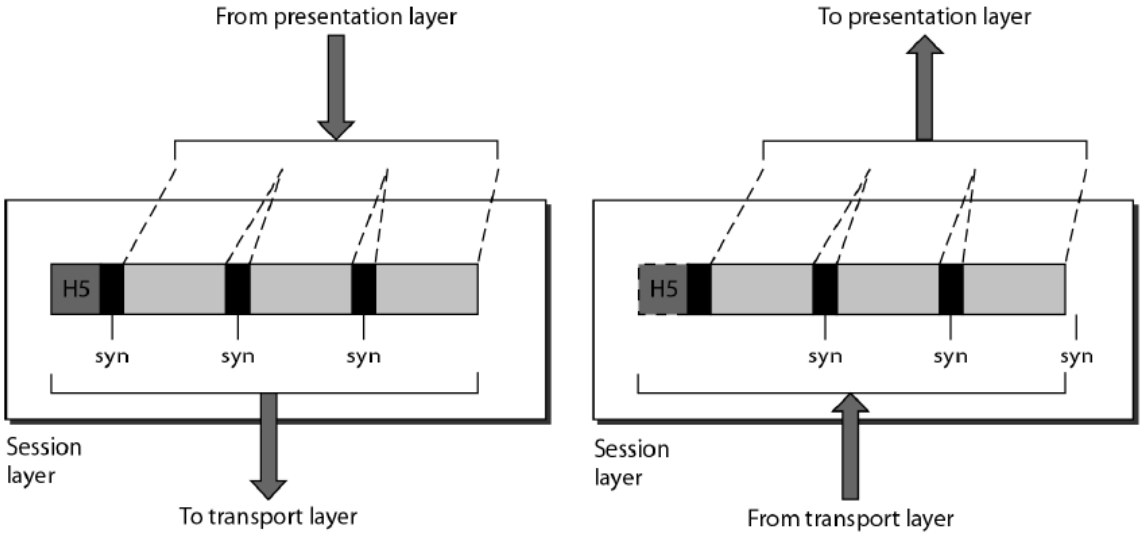


The other responsibilities of this layer are

- **Port addressing** - The header include the 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 be **connectionless or connection-oriented.** In connectionless each frame is treated as a packet and delivered to the destination. In connection oriented a connection is established to the destination side . After the delivery the connection will be terminated.
- **Flow and error control** - Similar to data link layer, but process to process take place.

5. Session layer:

This layer establishes, manages and terminates connections between applications.

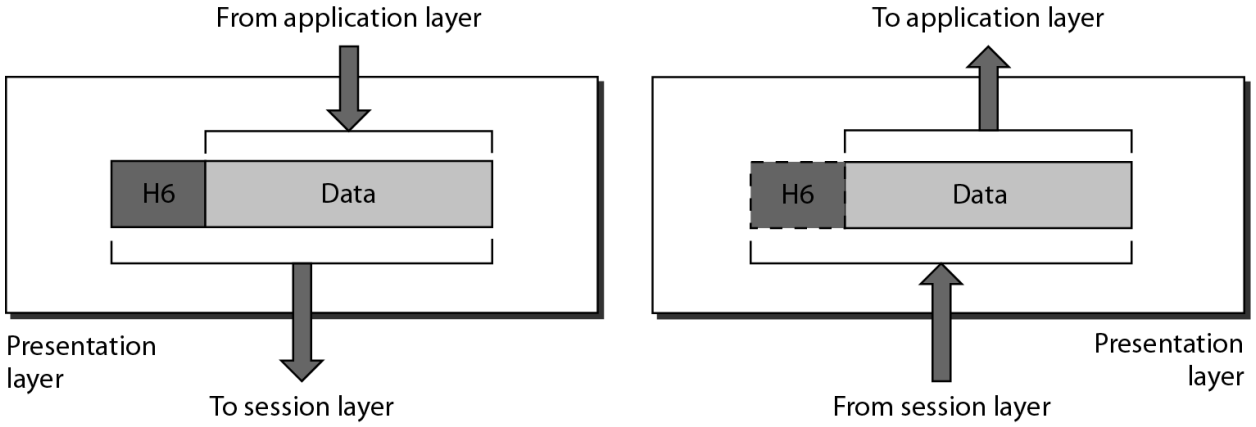


The other responsibilities of this layer are

- **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.

6.Presentation layer:

It is concerned with the syntax and semantics of information exchanged between two systems.



The other responsibilities of this layer are,

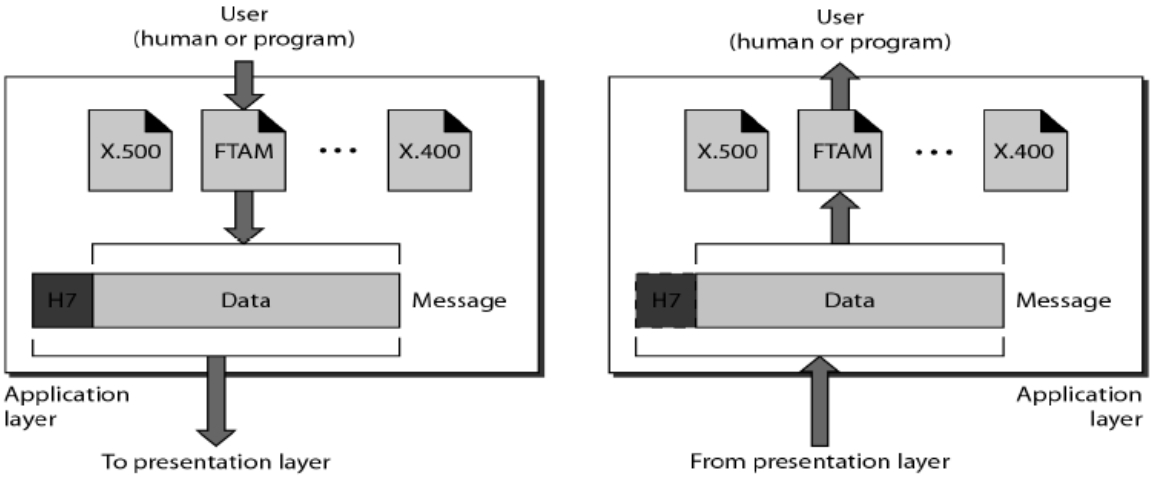
- **Translation** – Different computers use different encoding system, this layer is responsible for interoperability between these different

encoding methods. It will change the message into some common format.

- **Encryption and decryption**-It means that sender transforms the original information to another form and sends the resulting message over the n/w. and vice versa.
- **Compression and expansion**-Compression reduces the number of bits contained in the information particularly in text, audio and video.

7.Application layer:

This layer enables the user to access the n/w.



The other responsibilities of this layer are,

- **FTAM(file transfer, access, mgmt)** - Allows user to access files in a remote host.-
- **Mail services** - Provides email forwarding and storage.
- **Directory services** - Provides database sources to access information about various sources and objects.

1.3.2. INTERNET ARCHITECTURE:

The Internet architecture, which is also sometimes called the **TCP/IP architecture** is shown in the below Figure .

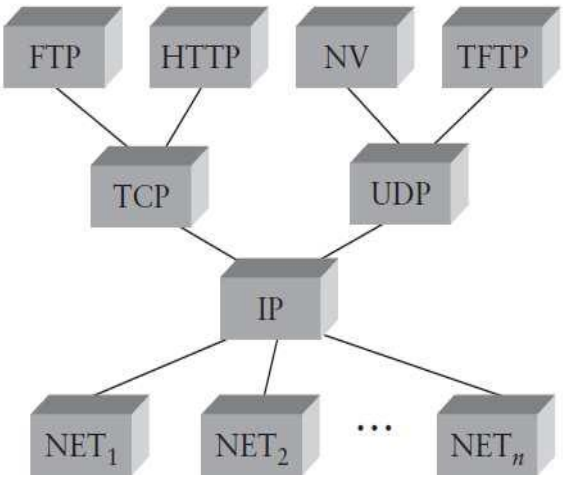


Fig: 1.7.Internet protocol graph.

An alternative representation is given in Figure below.

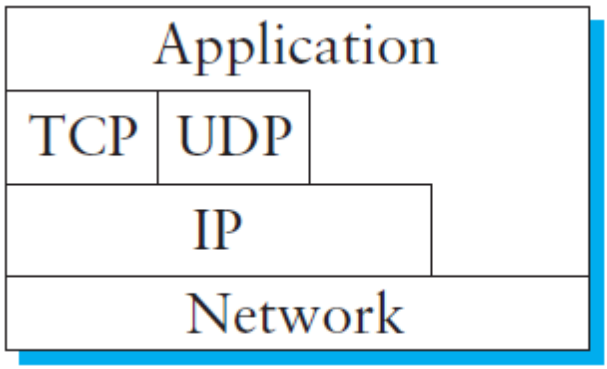


Fig :1.8. Alternative view of the Internet architecture.

- The Internet architecture evolved from 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 R&D funding agencies of the U.S. Department of Defense.
- The internet and ARPANET were developed before the OSI model.

Layers of Internet Architecture:

- Internet Architecture uses a four-layer model
- At the lowest level a wide variety of network protocols, like NET1, NET2, and so on are used.

- 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, Ethernet or wireless protocols is present in this layer.
- The second layer consists of a single protocol—the Internet Protocol (IP).
- This protocol 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 provides a reliable byte-stream channel, and UDP provides an unreliable datagram delivery channel.
- TCP and UDP are called end-to-end protocols , they can also be called as transport protocols.
- The fourth layer consist of a range of application protocols, such as FTP, HTTP, telnet ,and SMTP that enables the interoperation of popular applications.

Features of Internet:

The internet architecture has three features,

- The Internet architecture does not imply strict layering.
- Second, internet architecture uses hourglass shape—it is wide at the top narrow in the middle, and wide at the bottom. This says the fact that IP serves as the focal point for the architecture ,it defines a common method for exchanging packets among a wide collection of networks.
- The third feature of the Internet architecture is that for including a protocol officially in the architecture they must produce both a protocol specification and at least one representative implementations of the specification.

1.4. NETWORK SOFTWARE:

- The success of internet greatly depend on the software running on general purpose computers because by making small changes in the program it can be used for other applications.
- Example for such new applications are electronic commerce ,video conferencing etc…

Application programming interface (sockets):

- While implementing a network application, interface will be exported by the network.
- All computer systems implement their network protocol by using the interfaces which are part of the operating system (OS).
- The interface is called as network application programming interface (API).

Each operating system has its own API, at the same time it can also be imported to other systems by using socket interfaces for developing the application ex) java socket library.

Each protocol provides a certain set of services and the API provide certain set of syntax for using those protocols.

The socket interface defines,

- operation for creating a socket
- attaching the socket to the network
- sending/receiving message through the socket
- closing the socket.

1.The socket is created by using the operation:

int socket (int domain, int type, int protocol)

2.Attaching the socket to the network can be done using the three operations.

i) int bind(int socket ,struct sockaddr*address, int addr_len)

It binds the newly created socket to the specified address

ii) int listen(int socket,int backlog)

It defines how many socket can be pending on the specified socket

iii) int accept(int socket,struct sockaddr*address,int*addr_len)

it is a blocking operation that does not return until a remort participant has established a connection

3. sending and receiving message through the socket:

Once the connection is established the sending and receiving process starts, this is done by the operation,

int send(int socket, char*message, int msg_len, int flag)

int recv(int socket, char *buffer, int buff_len, int flag)

1.5. PERFORMANCE:

Network performance can be found by knowing,

1. Bandwidth and latency
2. Delay-bandwidth product
3. High speed networks
4. Application performance need

1. **Bandwidth and latency:**

The two fundamental ways for measuring network performance are,

1. Bandwidth(throughput)
2. Latency(Delay)

Bandwidth: It is the number of bits that can be transmitted over the network in a certain period of time. Example if bandwidth is 10 million bits/sec it is able to transmit 10 million bits in one second.

Latency: It says how long a message take to travel from one network to the other. Latency completely depend on time

Round trip time(RTT): The time taken to send a message from one end of the network to the other end and back.

Latency have three components,

1.Speed of light propagation delay

2. Amount of time taken to transmit a unit of data

3.Queuing delays

i)Speed of light propagation delay: This delay can be calculated if the distance between the two nodes are known, Because bits propagate at the speed of light, in vaccum light travel at a speed of $3*10^8$m/s. where else it travels at a speed of $2*10^8$m/s in optical fiber and $2.3*10^8$m/s in copper cable.

ii). Amount of time taken to transmit a unit of data : it is a function of

network bandwidth and the size of the packet in which the data is carried

iii). Queuing delays : usually switches store and forward ,so some delays could be encountered

Therefore latency is the total of above three.

Latency= propagation + transmit +Queue

Propagation= distance/speed of light

Transmit = size/bandwidth

2.Delay bandwidth product:

The delay bandwidth product is important to know when constructing high performance network because it corresponds to how many bits the sender must transmit before the first bit arrive at the receiver.

Most of the time delay bandwidth product refer to the RTT* bandwidth.

3.High speed networks:

High speed networks bring a considerable change in the Bandwidth availability for applications. This does not mean that latency is changing, example , for transmitting a 1Mbps data through a 1 Mbps network needs 80 round trip time to transmit the data where else for a 1Gbps network does not even require a 1RTT. For both cases the latency is the same.
The relation ship between throughput and latency is given as,

Throughput= Transfer size/ transfer time

Where,

Transfer time=RTT = 1/ Bandwidth * Transfer size

4.Application performance needs:

Depending on applications the bandwidth vary. Some applications mention the upper bound on how much bandwidth they need, example video applications.

1.6.LINK LAYER SERVICES:

The link layer services are,

- Framing
- Error control
- Flow control

1.6.1. FRAMING :

To transmit frames over the node it is necessary to mention start and end of each frame. There are three techniques to solve this frame.

- Byte-Oriented Protocols (BISYNC, PPP, DDCMP)
- Bit-Oriented Protocols (HDLC)
- Clock-Based Framing (SONET)

1.Byte Oriented protocols:

In this method frames are considered as collection of bytes .

Some of the byte oriented protocols are,

1.**BISYNC** (Binary Synchronous Communication protocol).
2. **DDCMP**(Digital Data Communication Message Protocol).
3. **PPP** (Point-to-point protocol)

BISYNC (Binary Synchronous Communication protocol).

Sentinel Approach:

The BISYNC protocol frame format is shown below,

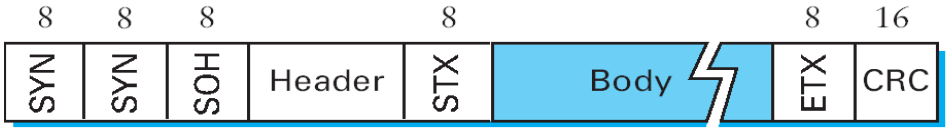


Fig:1.9. BISYNC Frame format

- 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.
- The frame format also includes a field labeled CRC (cyclic redundancy check) that is used to detect transmission errors.

The problem with the sentinel approach 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; the DLE character is also escaped (by preceding it with an extra DLE) in the frame body. This approach is called **character stuffing**.

Point-to-Point protocol (PPP):

The more recent is the Point-to-Point Protocol (PPP). The format of PPP frame is

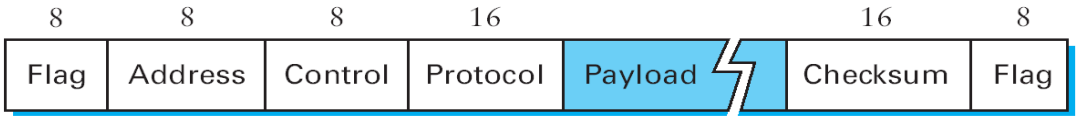


Fig:1.10. PPP Frame Format

- The Flag field has 01111110 as starting sequence.
- The Address and Control fields usually contain default values
- The Protocol field is used for de-multiplexing . The frame payload size can he negotiated, but it is 1500 bytes by default.
- The PPP frame format is unusual that several of the field sizes are negotiated rather than fixed.
- Negotiation is conducted by a protocol called LCP (Link Control Protocol).

- LCP sends control messages encapsulated in PPP frames—such messages are denoted by an LCP identifier in the PPP Protocol.

DDCMP(Digital Data Communication Message Protocol).

Byte-Counting Approach

The number of bytes contained in a frame can he included as a field in the frame header. DDCMP protocol is used for this approach. The frame format is

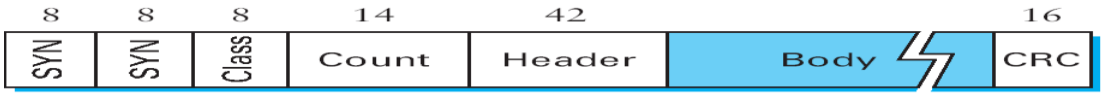


Fig: 1.11.DDCMP frame format

- COUNT Field specifies how many bytes are contained in the frame's body.
- Sometime count field will be corrupted during transmission, so the receiver will accumulate as many bytes as the COUNT field indicates. This is sometimes called a **framing error**.
- The receiver will then wait until it sees the next SYN character.

2.Bit-Oriented Protocols (HDLC)

In this, frames are viewed as collection of bits. The format is

Fig:1.12. HDLC Frame Format

- HDLC denotes both the beginning and the end of a frame with the bit sequence 01111110.
- This sequence might appear anywhere in the body of the frame, it can be avoided by bit stuffing.
- On the sending side, any time five consecutive 1's have been transmitted from the body of the message, the sender inserts a 0 before transmitting the next bit.
- On the receiving side, five consecutive 1's arrived, the receiver makes its decision based on the next bit it sees .If the next bit is a 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.

- By looking at the next bit, the receiver can distinguish between these two cases:
  1. If it sees a 0 (i.e., the last eight bits it has looked at are 01111110), then it is the end-of- frame marker.
  2. If it sees a 1 (i.e., the last eight bits it has looked at are 01111111), then there must have been an error and the whole frame is discarded.

3. Clock-Based Framing (SONET):

- Synchronous Optical Network Standard is used for long distance transmission of data over optical network.
- It supports multiplexing of several low speed links into one high speed links.
- An STS-1 frame is used in this method.

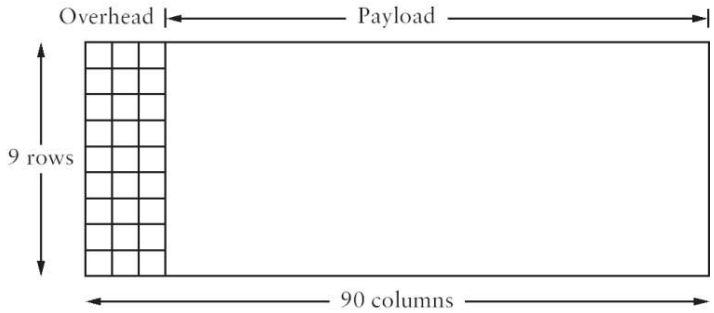


Fig :1.13.Sonnet STS-1 frame

- It is arranged as nine rows of 90 bytes each, and the first 3 bytes of each row are overhead, with the rest being available for data.
- The first 2 bytes of the frame contain a special bit pattern, and these bytes enable the receiver to determine where the frame starts.
- The receiver looks for the special bit pattern consistently, once in every 810 bytes, since each frame is 9 x 90 = 810 bytes long.

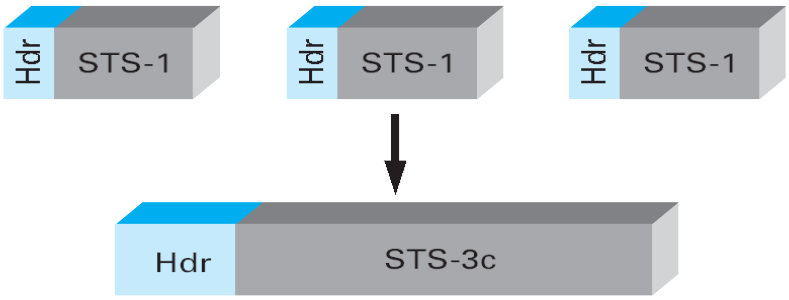


Fig:1.14. STS-N Frame

The STS-N consist of N STS-1 frames, where the bytes from these frames are interleaved; that is, a byte from the first frame is transmitted, then a byte from the second frame is transmitted, and so on.

- Payload from these STS-1 frames can he linked together to form a larger

STS-N payload, such a link is denoted STS-Nc.

- Because of the complexity of SONET detailed use of all the other overhead bytes can not be explained.
- The overhead bytes of a SONET frame are encoded using NRZ, this is the simple encoding where 1s are high and 0s are low

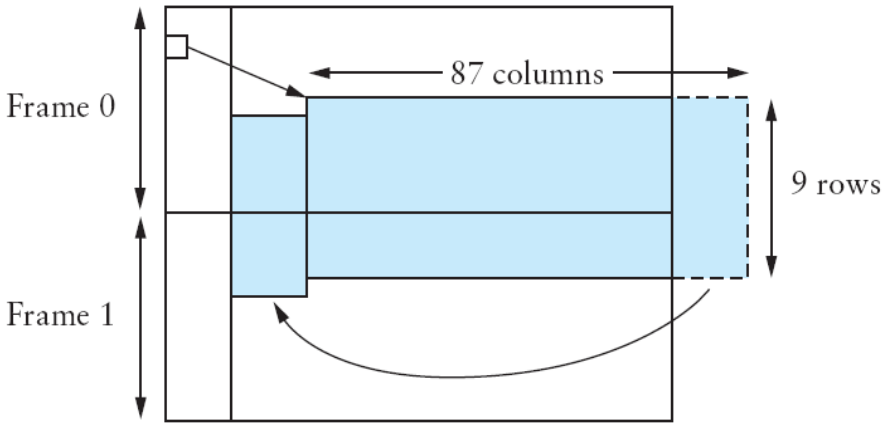


Fig:1.15. SONET frames out of phase.

1.6.2 .ERROR CONTROL:

When a data is transmitted from a source to destination, error may occur from natural sources and man made sources. Data communication errors are naturally classified as single bit, multiple bit or burst bit error.

Single bit error: Only one character is affected in a message

Burst –bit –error: Two or more consecutive bits are affected in a message. These types of error have to be controlled. Error control is of two types

- Error detection
- Error correction

1.ERROR DETECTION:

- It is a process of monitoring data transmission and determining when error have occurred. The most common error detection techniques are redundancy checking.
- **Redundancy:**

  One method is to send every data twice, so that receiver checks every bit of two copies and detect error.

  Drawbacks:
  - Sends n-redundant bits for n-bit message.
  - Many errors are undetected if both the copies are corrupted.

Error Detection Techniques:

Instead of adding entire data, some bits are appended to each unit. This is also called redundant bit but the bits added will not give any new information. These bits are called error detecting codes.

The three error detection techniques are,

- Two dimensional parity check
- Checksum
- Cyclic redundancy check

Two-Dimensional Parity:

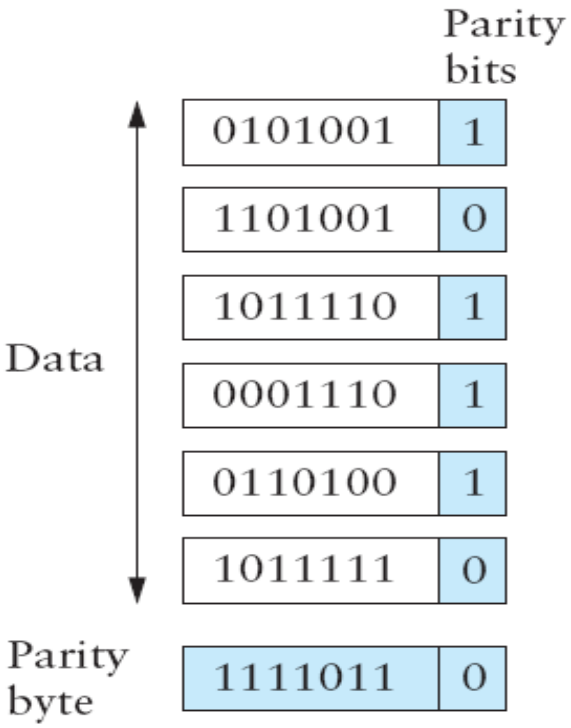
It is based on "simple"(one-dimensional) parity.

One dimensional parity

- It is adding one extra bit to a 7-bit code to balance the number of 1s in the byte.
- There are two types of parity, odd parity and even parity.
- If the number of ones in the eight bit is odd it is called odd parity
- If the number of ones in the eight bit is even it is called even parity

Two-dimensional parity

- It does a similar calculation for each bit position across each of the bytes contained in the frame. This results in an extra parity byte for the entire frame, in addition to a parity bit for each byte.



- Notice that the third bit of the parity byte is 1 since there are an odd number of 1s in the third bit across the 6 bytes in the frame.

Internet Checksum Algorithm:

- The idea behind the Internet checksum is very simple, you add up all the words that are transmitted and then transmit the result of that sum. The result is called the checksum.
- The receiver performs the same calculation on the received data and compares the result with the received checksum.
- If any transmitted data, including the checksum itself, is corrupted, then the results will not match, so the receiver knows that an error occurred.
- Consider the data being check summed as a sequence of 16-bit integers. Add them together and then take the ones complement of the result. That 16-bit number is the checksum.
- Example , If we have three 16-bit words 0110011001100000 , 0101010101010101,1000111100001100.

The sum of fist two 16-bit words ,

0110011001100000 +
0101010101010101

1011101110110101

3rd word 1000111100001100

```
----------------------------
 10100101011000001
----------------------------
 0100101011000010
```

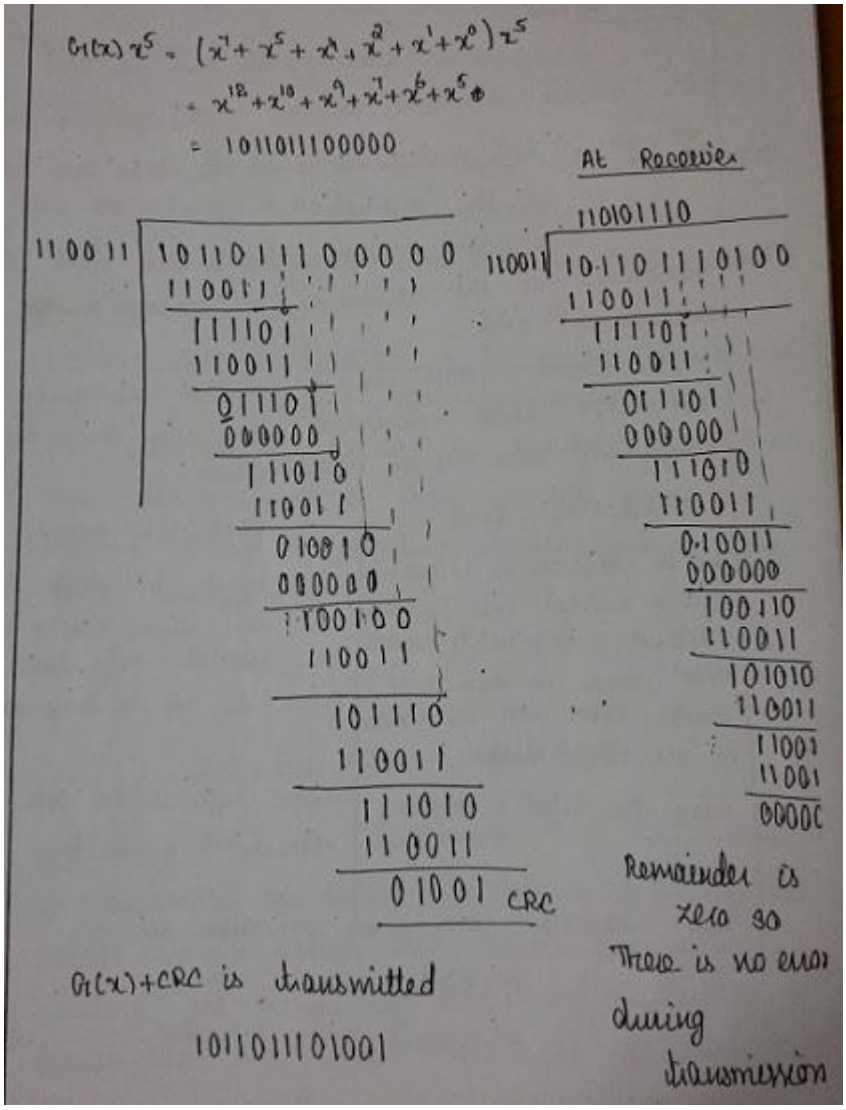
The ones complement is 1011010100111101, all the four 16-bit word are transmitted ,at the receiving end all the four words are added if we get the result as 1111111111111111, No error have occurred else there is error.

Cyclic Redundancy Check:

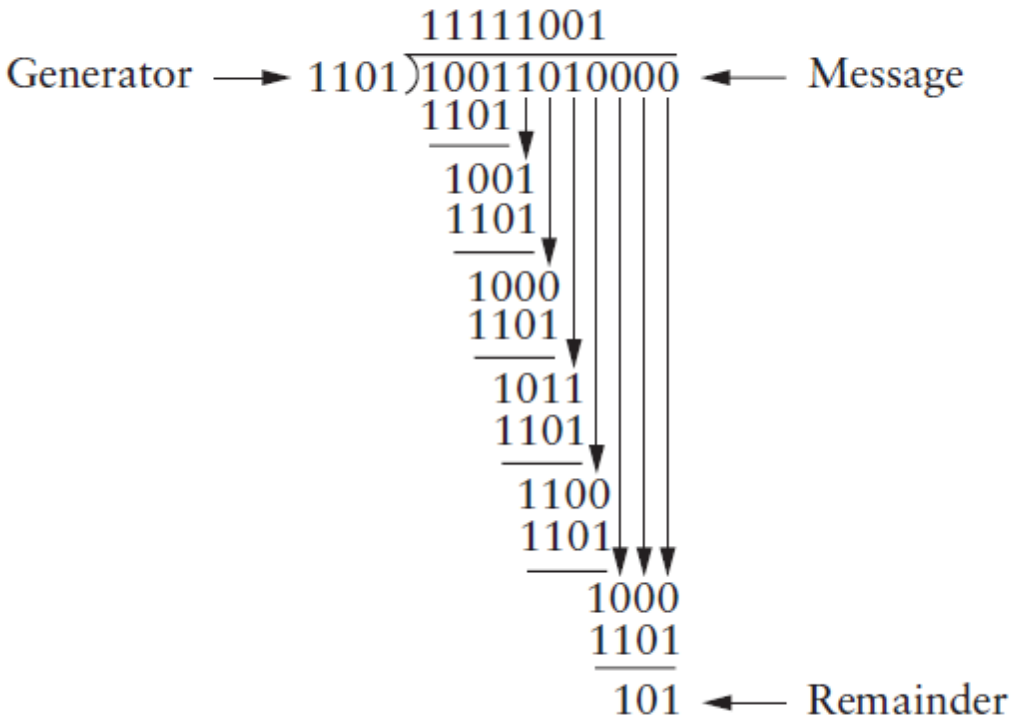
- Cyclic redundancy checks uses a powerful mathematics to avoid error. For example, a 32-bit CRC gives strong protection against common bit errors in messages that are thousands of bytes long.
- We can thus think of a sender and a receiver as exchanging polynomials with each other.
- For example, an 8-bit message consisting of the bits 10110111 corresponds to the polynomial

  $G(X)=x^7 +x^5 +x^4 + x^2 +x^1 +x^0$
- For the purposes of calculating a CRC, a sender and receiver have to agree on a divisor polynomial, P(x). P(x) is a polynomial of degree k. For example, suppose $C(x) = x^5 + x^4 +x^1+ x^0$. In this case, k = 5.
- When a sender wishes to transmit a message G(x) that is n + 1 bits long, what is actually sent is the (n + 1)-bit message plus k bits. We call the complete transmitted message, including the redundant bits, R(x).
- If P(x) is transmitted over a link and there are no errors introduced during transmission, then the receiver should be able to divide R(x) by P(x) exactly, leaving a remainder of zero.
- On the other hand, if some error is introduced into R(x) during transmission, then P(X) can not be exactly divisible by P(x), and thus the receiver will obtain a nonzero remainder, indicating an error has occurred.

EXAMPLE 1:



- **Example 2:** Consider the message G(X)= x7 + x4 + x3 + x1, or 10011010, $P(X)=x^3+x^2+1$
- Multiply G(X) by x3, since our divisor polynomial is of degree 3. This gives 10011010000.
- We divide this by P(x), which corresponds to 1101 in this case. that the remainder of the example calculation is 101
- so we add this 101 with the original message 10011010 and send 10011010101.
- At the receiver side 10011010101 is divided by 1101 if the remainder is zero no error had occurred, if the remainder is non-zero then error had occurred.



- so we actually send 10011010101.

1.6.3. FLOW CONTROL :

- Flow control is a mechanism of recovering the lost frames.
- To recover the lost frames a link-level protocol must be developed
- Such a protocol is developed employing two fundamental mechanisms,
    1. Acknowledgements
    2. Timeouts
- An acknowledgment (ACK) is a small control frame that a protocol sends back to its peer saying that it has received an earlier frame.
- If the sender does not receive an acknowledgment after a reasonable amount of time, then it retransmits the original frame. This action of waiting a reasonable amount of time is called a **timeout**.
- This way of using acknowledgement and timeout to achieve reliable transmission is called Automatic repeat request(ARQ)

Stop-and-Wait:[Error correction –retransmission technique]

The simplest ARQ scheme is the stop-and-wait algorithm. The idea of stop-and-wait is ,

- After transmitting one frame, the sender waits for an acknowledgment before transmitting the next frame.
- If the acknowledgment does not arrive after a certain period of time, the sender times out and retransmits the original frame.
- The figure below shows the protocol's behavior.

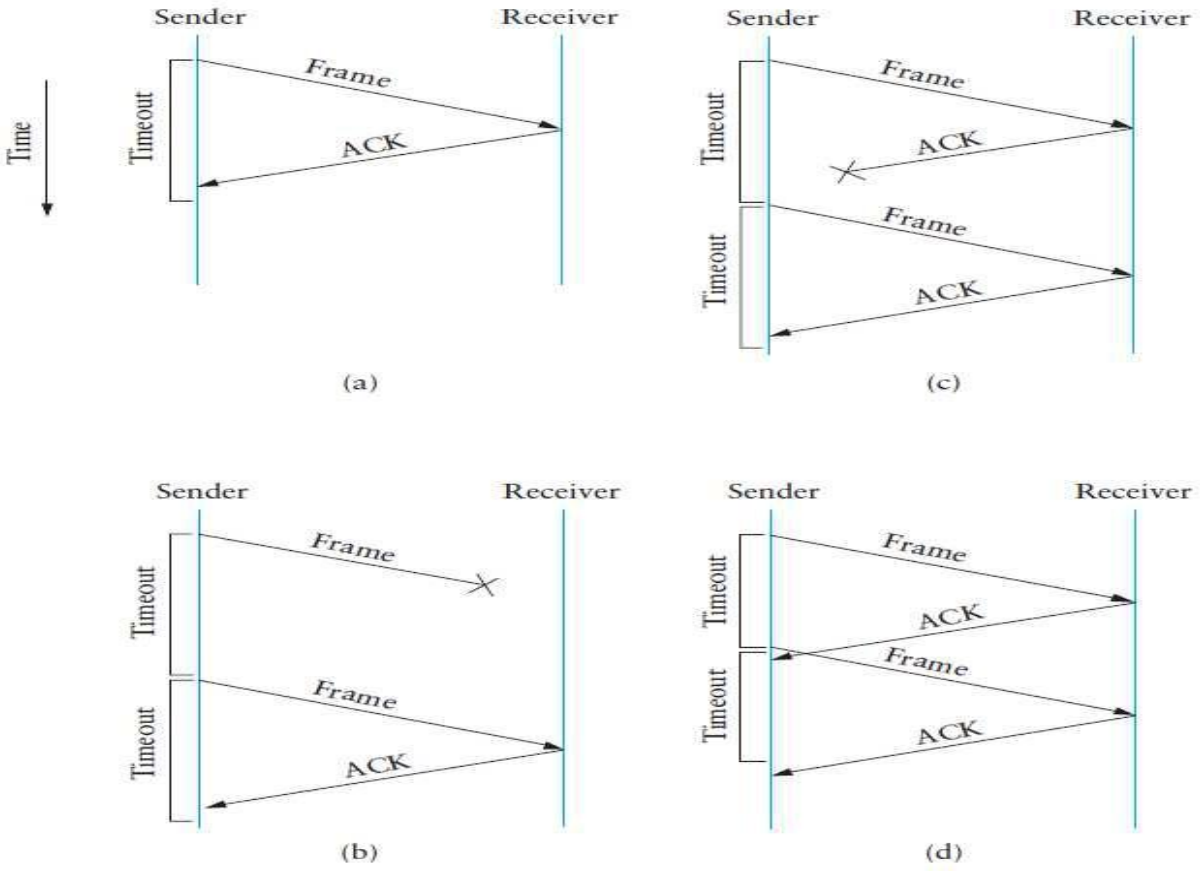


Fig:1.16 stop & wait ARQ

- The sending side is represented on the left and the receiving side is represented on the right, and time flows from top to bottom.
- Scenarios (a) shows the ACK is received before the timer expires, (b) shows the original frame is lost, (c) shows the ACK is lost and (d) shows the timeout fires too soon
- In the case of (c) and (d), the sender times out and re-transmits the original frame. But the receiver will think that it is the next frame, since it correctly received and acknowledged the first frame.

- To avoid this problem, the header for a stop-and-wait protocol includes a 1-bit sequence number. The sequence number can take on the values 0 and 1 and the sequence numbers used for each frame alternate.
- This is shown in the figure below,

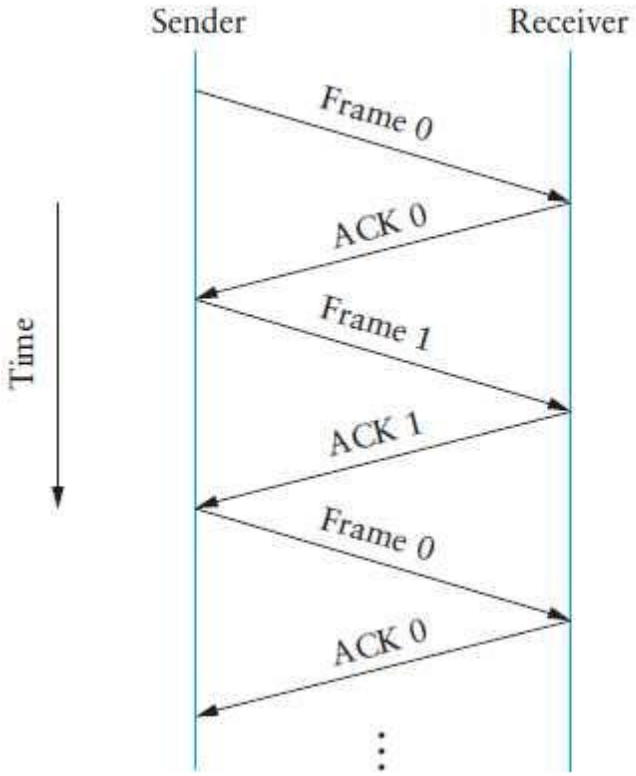


Fig:1.17 Frame sequence number assignment

- The drawback of the stop-and-wait algorithm is that it allows the sender to have only one frame on the link at a time, and this is far below the link's capacity.
- This algorithm is uses only one-eighth of the link 'capacity.
- To use the link fully, we'd like the sender to transmit up to eight frames before waiting for an acknowledgment. This principle is called as keeping the pipe full, and the algorithm that allows us to do this is called **sliding window.**

Sliding Window Algorithm

The sliding window algorithm works as follows.

- First, the sender assigns a sequence number, denoted SeqNum, to each frame. Let the SeqNum be infinite.
- The sender maintains three variables:
    1. The send window size (SWS) which gives the upper bound on the number of frames that the sender can transmit
    2. LAR denotes the sequence number of the last acknowledgment received.
    3. LFS denotes the sequence number of the last frame sent.
- The sender also maintains that $LFS - LAR \leq SWS$

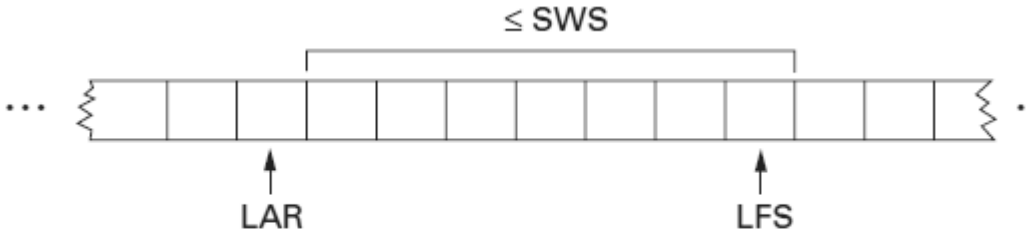


Fig:1.18. Sliding window on sender

- When an acknowledgment arrives, the sender moves LAR to the right, thereby allowing the sender to transmit another frame. Also, the sender attach a timer with each frame it transmits, and it retransmits the frame if the timer expire before an ACK is received.

Fig: Sliding window on receiver

- The receiver maintains the following three variables:
    1. The receive window size(RWS) which gives the upper bound on the number of frames that the receiver can accept.
    2. LAF denotes the sequence number of the largest acceptable frame
    3. LFR denotes the sequence number of the last frame received.

- The receiver also maintains $LAF - LFR \leq RWS$
- When a frame with sequence number arrives, the receiver takes the following action.

1. If SeqNum ≤ LFR or SeqNum > LAF, then the frame is outside
   the receiver's window and it is discarded.

2. If LFR < SeqNum ≤ LAF, then the frame is within the receiver's window and it is accepted.

- Now the receiver needs to decide whether or not to send an ACK .
- Let SeqNumToAck denote the largest sequence number not yet acknowledged, if all frames with sequence numbers less than or equal to SeqNumToAck have been received ,the receiver send acknowledgement.
- It then sets LFR =SeqNumToAck and adjusts and LAF = LFR + RWS.
- Suppose if frame 7 and 8 are received whereas frame 6 is not yet received ACK will not be sent. ACK will be send only after receiving frame 6. At the same time if frame 6 is lost or delayed a timeout occur and the sender will retransmit the 6^{th} frame and the sender is unable to advance its window until frame 6 is acknowledged. This means that when packet losses occur, this scheme is no longer keeping the pipe full.
- To overcome this problem we are going for another scheme in which the receiver send acknowledgement to the frame that it has received.
- For example if frame 7 and 8 are received and 6 is not received the receiver send acknowledgement to frame 7 and frame 8. So that the sender will know that frame 7 and 8 were received and it fill the next frames to be sent in the window.
- The sending window size is selected according to how many frames we want to sent; SWS is easy to compute for a given delay × bandwidth product .
- The receiver can set RWS to whatever it wants. Two common settings are
  - ✓ RWS = 1, which implies that the receiver will not buffer any frames that arrive out of order,
  - ✓ RWS = SWS, which implies that the receiver can buffer any of the frames the sender transmits.

Finite Sequence Numbers and Sliding Window:

- Now let us assume that the frame sequence number is finite. For example, a 3-bit field means that there are eight possible sequence numbers, 0 . . . 7.
- If SWS ≤ MaxSeqNum − 1, ie)7-1.
- If RWS = 1, then MaxSeqNum ≥ SWS + 1 is sufficient.
- If RWS is equal to SWS, and If we have the eight sequence numbers 0 through 7, and SWS = RWS = 7.
- Suppose the sender transmits frames 0..6, they are successfully received, but the ACKs are lost .The receiver is now expecting frames 7, but the sender times out and sends frames 0..6. This have to be avoided..
- Therefore when RWS = SWS, or stated more precisely, SWS < (MaxSeqNum + 1)/2.

UNIT II MEDIA ACCESS & INTERNETWORKING

Media access control - Ethernet (802.3) - Wireless LANs – 802.11 – Bluetooth - Switching and bridging – Basic Internetworking (IP, CIDR, ARP, DHCP,ICMP)

2.1 MEDIUM ACCESS CONTROL:-

Some network topologies share a common medium with multiple nodes. Sometimes number of devices may send and receive data at that time so that collision may occur. There are rules for sharing media and controlling collision. The two commonly used methods are:

1. CSMA/Collision Detection
2. CSMA/Collision Avoidance

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

The basic idea is When a station has a frame to transmit:

1) Listen for Data Transmission on Cable
2) When Medium is Quiet
   a) Transmit Frame, Listening for Collision
   b) If collision is heard, stop transmitting, wait random time, and transmit again.

Frame format

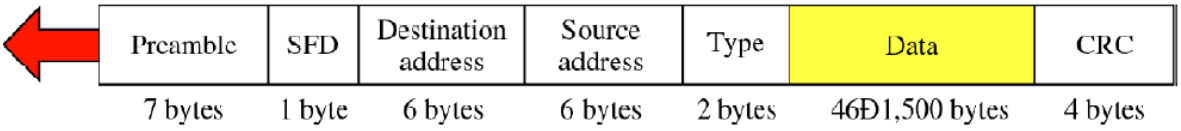


Fig 2.1 : Frame format

Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA)

- CSMA/CA (Carrier Sense Multiple Access/Collision Avoidance) is a protocol for carrier transmission in 802.11 networks.
- Unlike CSMA/CD which deals with transmissions after a collision has occurred, CSMA/CA prevent collisions before they happen.
- In CSMA/CA, as soon as a node receives a packet that is to be sent, it checks whether the channel is clear.
- If the channel is clear, then the packet is sent. If the channel is not clear, the node waits for a randomly chosen period of time, and then checks again whether the channel is clear.
- This period of time is called the backoff factor, and is counted by a backoff counter.

- If the channel is clear when the backoff counter reaches zero, the node transmits the packet. If the channel is not clear when the backoff counter reaches zero, the backoff counter is set again, and the process starts.

2.2 ETHERNET (802.3)

- Ethernet is a multiple-access network ie) a set of nodes send and receive frames over a shared link.
- The fundamental problem faced by the Ethernet is how to mediate access to a shared medium fairly and efficiently.
- A 10-Mbps Ethernet standard was developed in 1978 which is the basis for IEEE standard 802.3.
    1. 100-Mbps version is called Fast Ethernet
    2. 1000-Mbps version is called Gigabit Ethernet.
- Both 100-Mbps and 1000-Mbps Ethernets are used in full-duplex, point-to-point configurations ie) they are used in switched networks.

PHYSICAL PROPERTIES:

- An Ethernet can be a coaxial cable with an impedance of 50 ohms up to 500m.
- Hosts are connect to an Ethernet segment by tapping into it.
- A *transceiver(a small device)* is attached to the tap ,it detects when the line is idle and drives the signal when the host is transmitting.
- It also receives incoming signals.
- The transceiver is, in turn, connected to an Ethernet adaptor, which is plugged into the host.

Ethernet Transceiver and adaptor

- Multiple Ethernet segments can be joined together by repeaters. A repeater is a device that forwards digital signals,
- Only four repeaters can be placed between any two hosts, meaning that an Ethernet has a total reach of only 2500 m.
- Ethernet is limited to supporting a maximum of 1024 hosts.

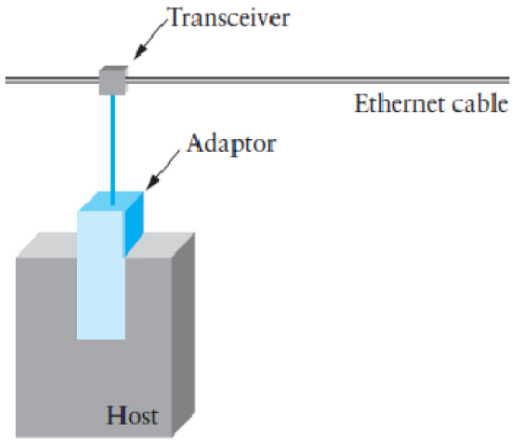


Fig 2.2 Ethernet Transceiver and adaptor

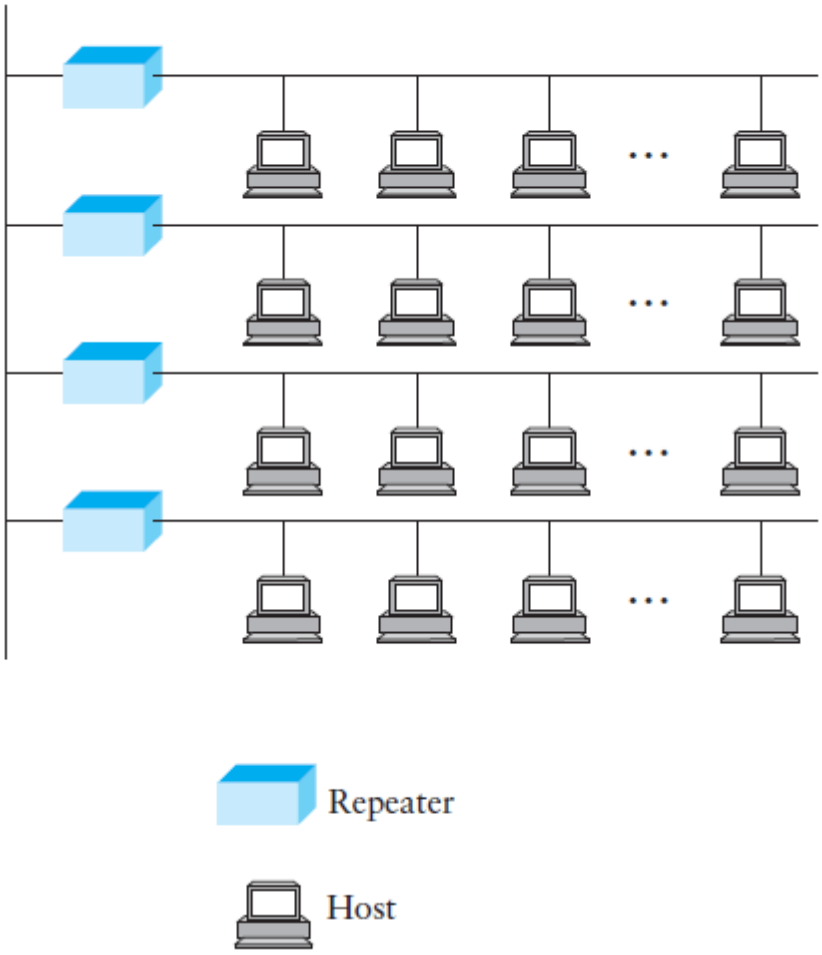


Fig 2.3 Ethernet Repeater:

- Any signal placed on the Ethernet by a host is broadcast over the entire network; that is, the signal is propagated in both directions, and repeaters forward the signal on all outgoing segments.
- Terminators attached to the end of each segment absorb the signal and keep it from bouncing back and interfering with trailing signals.
- The Ethernet uses the Manchester encoding scheme. 4B/5B encoding as well as,8B/10B encoding is used today for high speed Ethernets.
- Ethernet can be constructed from a thinner cable known as10Base2 or 10Base5 (the two cables are commonly called thin-net and thick-net).

Ethernet Hub:

A hub send the received information to all its other ports. It is also called as a multi way repeater. When multiple Ethernets are connected by means of a hub data transmitted by any one host on the Ethernet reaches all the other host.

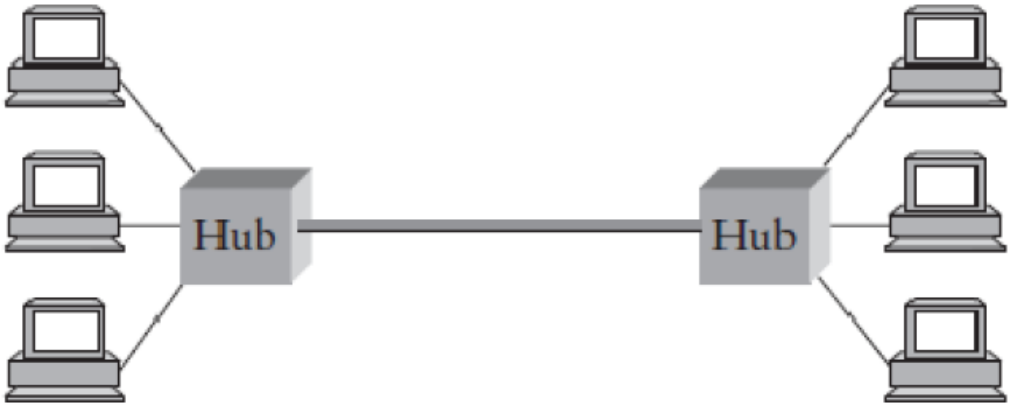


Fig 2.4 Ethernet Hub

ACCESS PROTOCOL:

- The algorithm that controls access to the shared Ethernet link is called the Ethernet's *media access control (MAC).*
- It is typically implemented in hardware on the network adaptor.

Frame Format:

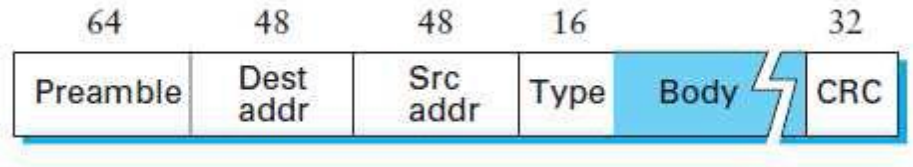


Fig 2.5 Frame format

- The 64-bit preamble allows the receiver to synchronize with the signal; it is a sequence of alternating 0sand 1s.

- Both the source and destination hosts are identified with a 48-bit address.
- The packet type field serves as the de-multiplexing key; that is, it identifies where the frame should be delivered.
- Each frame contains up to 1500 bytes of data. Minimally, a frame must contain at least 46 bytes of data.
- The reason for this minimum frame size is that the frame must be able to detect a collision. Each frame includes a 32-bit CRC.
- The Ethernet is a bit-oriented framing protocol .Ethernet frame has a 14-byte header: two 6-byte addresses and a 2-byte type field.

Addresses

- Every Ethernet host in the world has a unique Ethernet address.
- Technically, the address belongs to the adaptor, not the host; it is usually burned into ROM.
- Ethernet addresses are typically printed in a form humans can read as a sequence of six numbers separated by colons.
- For example, 8:0:2b:e4:b1:2 is the human-readable representation of Ethernet address 00001000 00000000 00101011 11100100 10110001 00000010
- To ensure that every adaptor gets a unique address, each manufacturer of Ethernet devices is allocated a different prefix that must be prepended to the address on every adaptor they build.
- For example, Advanced Micro Devices has been assigned the 24-bit prefix x080020 (or 8:0:20).
- Each frame transmitted on an Ethernet is received by every adaptor connected to that Ethernet. Each adaptor recognizes those frames addressed to its address and passes only those frames on to the host
- Ethernet adaptor receives all frames and accepts
    1. frames addressed to its own address
    2. frames addressed to the broadcast address
    3. frames addressed to a multicast address
    4. all frames, if it has been placed in promiscuous mode

Transmitter Algorithm

When the adaptor has a frame to send and the line is idle, it transmits the frame
Immediately .The upper bound of 1500 bytes in the message means that the adaptor can occupy the line for only a fixed length of time.

When an adaptor has a frame to send and the line is busy, it waits for the line to go idle and then transmits immediately.

- Since there is no centralized control it is possible for two (or more) adaptors to begin transmitting at the same time, because both found the line to be idle or because both had been waiting for a busy line to become idle.
- When this happens, the two (or more) frames are said to collide on the network.
- Since Ethernet supports collision detection, it 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 transmit a 32-bit jamming sequence and then stops the transmission. Thus, a transmitter will minimally send 96 bits in the case of a collision: 64-bit preamble plus 32-bit jamming sequence.
- This way of adopter sending 96 bits is called **a runt frame.**

Worst-case scenario: (a) A sends a frame at time t; (b) A's frame arrives at B at time t + d; (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.

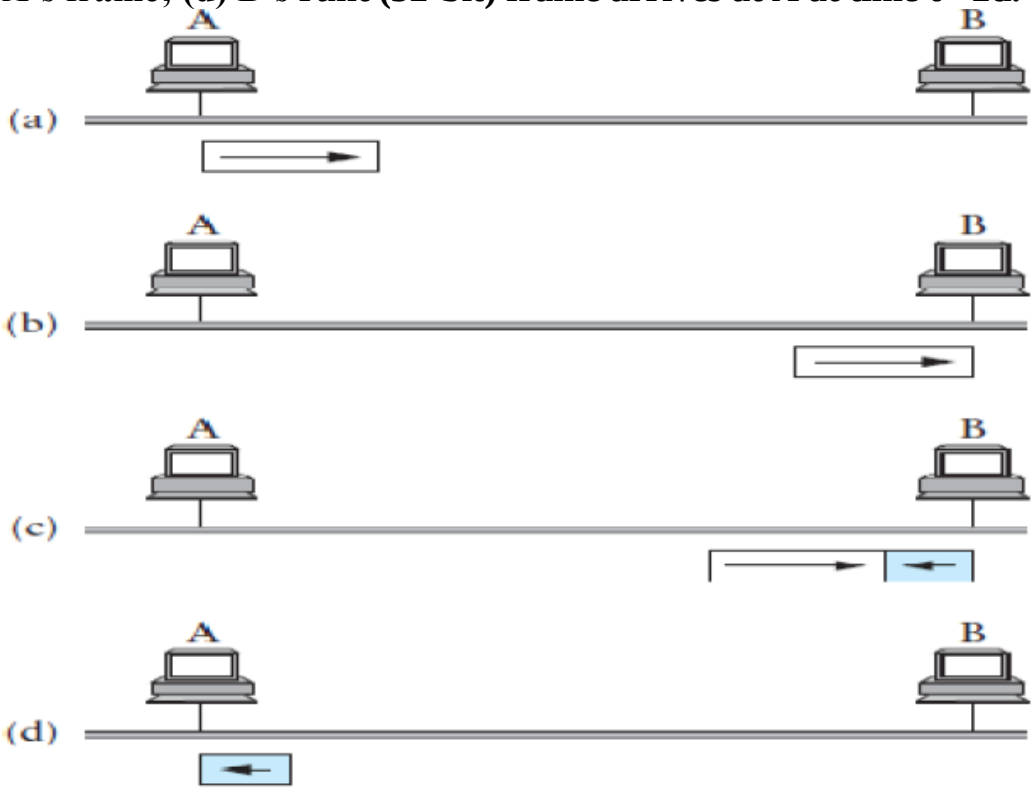


Fig:2.6. Worst case scenario

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

Experience with Ethernet:

Advantages:

- Ethernet is extremely easy to administer and maintain: There are no switches that can fail, no routing or configuration tables that have to be kept up-todate, and it is easy to add a new host to the network.
- It is inexpensive: Cable is cheap, and the only other cost is the network adaptor on each host.

Drawbacks:

- Utilization - 30%.
- Too much of the network's capacity is wasted by collisions.
- Most Ethernets have fewer than 200 hosts.
- Similarly, most Ethernets are far shorter than 2500 m.
- provide an end-to-end flow-control mechanism.

2.3 WIRELESS LAN

Wireless networking is a rapidly evolving technology for connecting computers. Wireless technologies are categorized on the basis of data rate they provide and how far apart the communication nodes can be placed. Other important differences they include which part of the electromagnetic spectrum they are using and how much power they consume.

2.3.1 802.11 / Wi-Fi.

802.11 is designed for use in a limited geographical area.

Physical properties:

- 802.11 was designed to operate on three different physical media—two based on spread spectrum radio and one based on diffused infrared.

- The radio-based versions currently run at 11 Mbps, but may soon run at 54 Mbps.
- The idea behind spread spectrum is to spread the signal over a wider frequency band than normal, so as to minimize interference from other devices.
- For example, **frequency hopping is a spread** spectrum technique that involves transmitting the signal over a random sequence of frequencies; that is, first transmitting at one frequency, then a second, then a third, and so on.
- A second spread spectrum technique, called **direct sequence,** achieves the same effect by representing each bit in the frame by multiple bits in the transmitted signal.
- For each bit the sender wants to transmit, it actually sends the exclusive-OR of that bit and n random bits.
- As with frequency hopping, the sequence of random bits is generated by a pseudorandom number generator known to both the sender and the receiver.
- The transmitted values, known as an n-bit chipping code, spread the signal across a frequency band that is n times wider than the frame would have otherwise required.
- 802.11 defines one physical layer using frequency hopping and a second using direct. The third using infrared signals.
- The transmission is diffused, that is they do not need a clear line of sight. This technology has a range of up to about 10 m and is limited to the inside of buildings only.

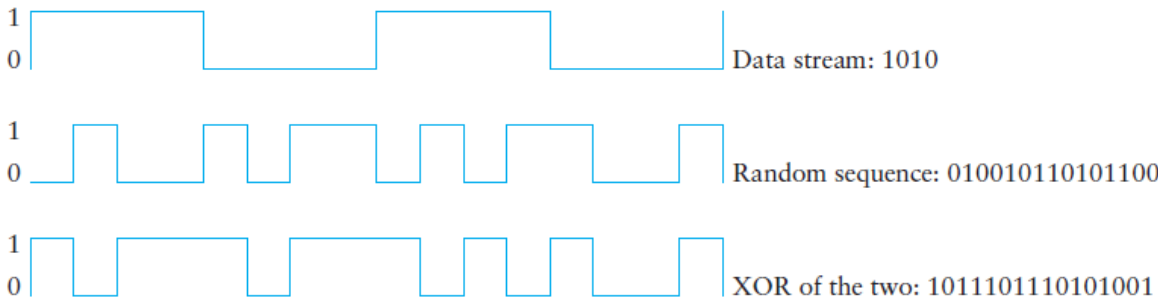


Fig:2.7. Example 4-bit chipping sequence.

Collision Avoidance :

- At first glance, a wireless protocol seems to follow exactly the same algorithm as the Ethernet ie) wait until the link becomes idle before transmitting

- Where each of four nodes is able to send and receive signals that reach just the nodes to its immediate left and right. For example, B can exchange frames with A and C but it cannot reach D, while C can reach B and D but not A.
- 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. A and C are said to be hidden nodes with respect to each other.

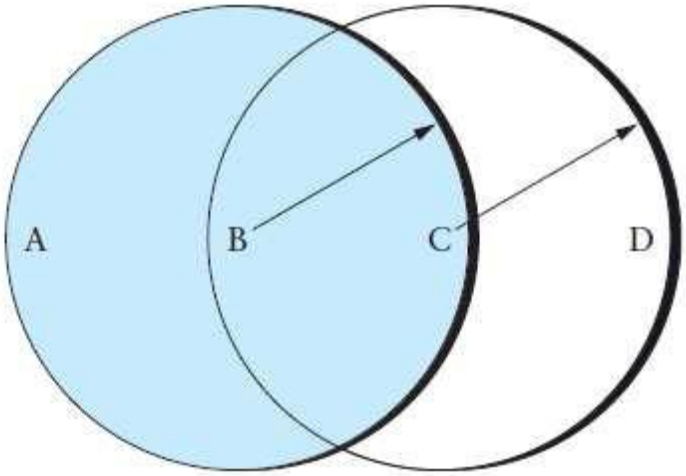


Fig:2.8.Example wireless network

- 802.11 avoid these problems with an algorithm called Multiple Access with Collision Avoidance (MACA). The idea is the 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. Specifically, the 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 (i.e., it specifies the length of the data frame to be transmitted).
- The receiver then 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 the receiver is busy . The receiver sends an ACK to the sender after successfully receiving a frame. All nodes must wait for this ACK before trying to transmit.
- Second, when two or more nodes detect an idle link and try to transmit an RTS frame at the same time, their RTS frames will collide with each other.

- 802.11 does not support collision detection, but instead the senders realize the collision has happened when they do not receive the CTS frame after a period of time, in which 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 node .

Distribution System

- 802.11 would be suitable for a network with mesh topology.
- Moreover, one of the advantages of a wireless network is that nodes are free to move around ,they are not wired
- The set of directly reachable nodes may change over time.
- Nodes are free to directly communicate with each other as just described, but in practice, the operate within this structure.
- Instead of all nodes being created equal, some nodes are allowed to roam (e.g., your laptop) and some are connected to a wired network infrastructure.
- The latter are called access points (AP), and they are connected to each other by a so-called distribution system.

Access points connected to a distribution network

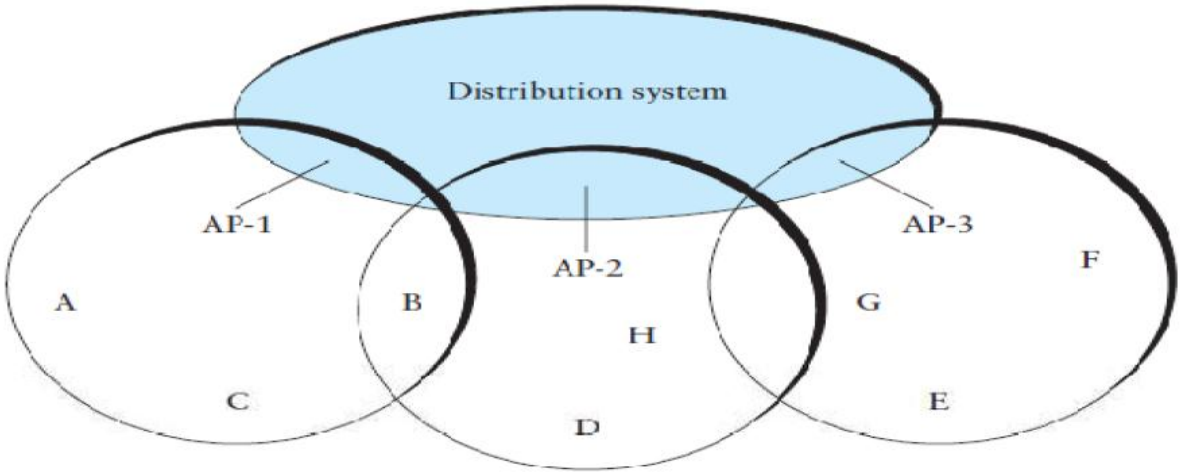


Fig:2.9.Access points connected to a distribution network

- In this example distribution system connects three access points, each of which services the nodes in some region. Each of these regions is analogous to a cell in a cellular phone system, with the APs playing the same role as a base station.
- Distribution network runs at layer 2 of the ISO architecture; that is, it does not depend on any higher-level protocols.

- Although two nodes can communicate directly with each other if they are within reach of each other, the idea behind this configuration is that each node associates itself with one access point.
- For node A to communicate with node E, for example, A first sends a frame to its access point (AP-1), which forwards the frame across the distribution system to AP-3, which finally transmits the message to node E. In this example distribution system that connects three access points, each of which services the nodes in some region. Each of these regions is analogous to a cell in a cellular phone system, with the APs playing the same role as a base station.

The technique for selecting an AP is called *scanning and involves the following* four steps:

1 .The node sends a Probe frame.
2 All APs within reach reply with a Probe Response frame.
3. The node selects one of the access points and sends that AP an Association Request frame.
4. The AP replies with an Association Response frame.

- A node engages this protocol whenever it joins the network, as well as when it does not get signal from its current AP.
- Whenever a node acquires a new AP, the new AP notifies the old AP of the change through the distribution system.

Node mobility:

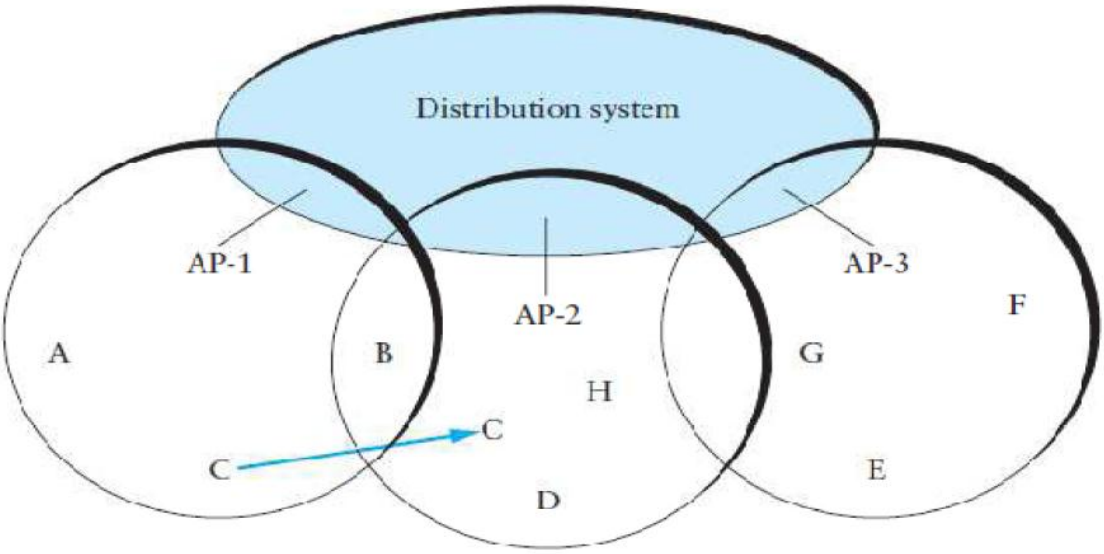


Fig:2.10.Node Mobility

- where node C moves from the cell serviced by AP-1 to the cell serviced by AP-2. it sends Probe frames, which result in Probe

Response frames from AP-2. At some point, C prefers AP-2 over AP-1, and so it associates itself with that access point.

- The mechanism just described 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 rates supported by the AP. This is called *passive scanning, and 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.

Frame Format

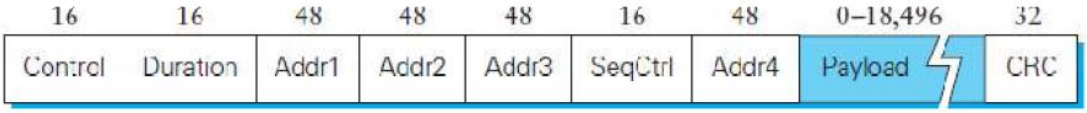


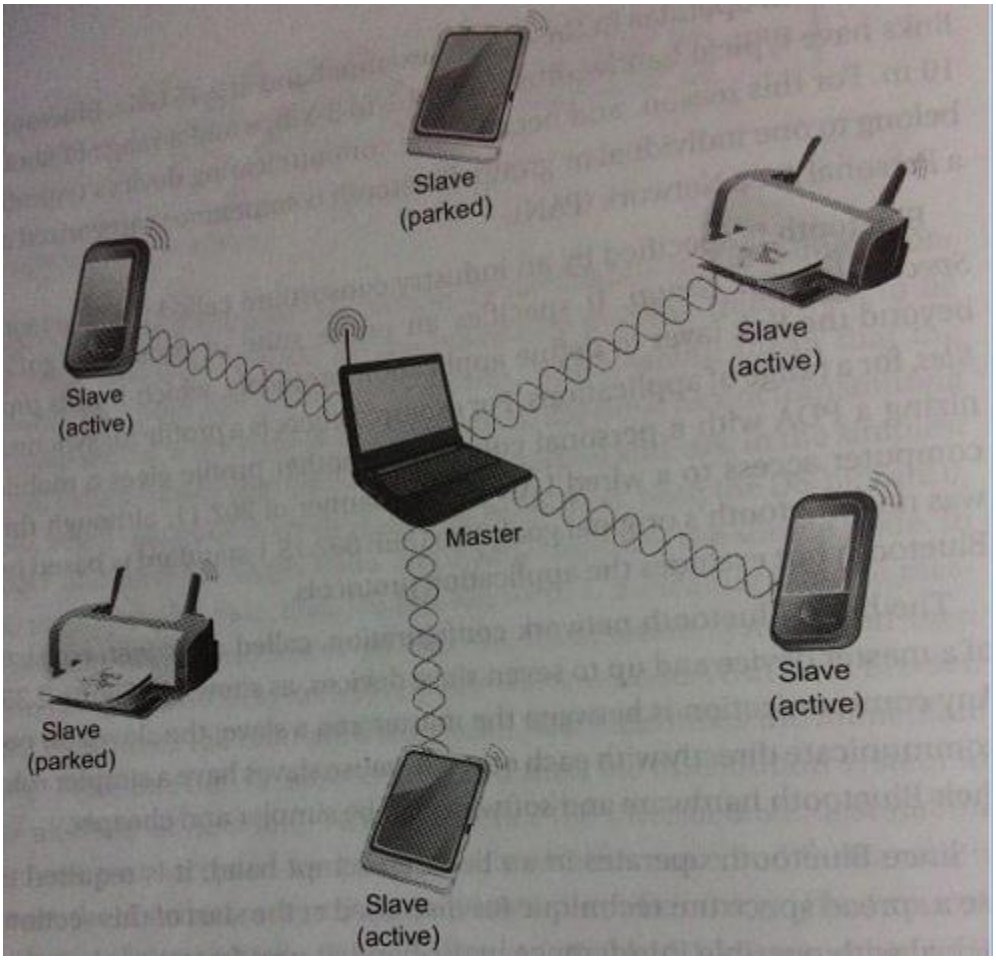
Fig 2.11 Frame format

- The frame contains the source and destination node addresses, each of which are 48 bits long; up to 2312 bytes of data; and a 32-bit CRC.
- The Control field contains three subfields : a 6-bit Type field that indicates whether the frame carries data, is an RTS or CTS frame, or is being used by the scanning algorithm ,and a pair of 1-bit fields—called ToDS and FromDS—that are described below.
- In the simplest case, when one node is sending directly to another, both the DS bits are 0, Addr1 identifies the target node, and Addr2 identifies the source node. In the most complex case, both DS bits are set to 1, indicating that the message went from a wireless node onto the distribution system, and then from the distribution system to another wireless node. With both bits set, Addr1 identifies the ultimate destination, Addr2 identifies the immediate sender.
- Addr3 identifies the intermediate destination (the one that accepted the frame from a wireless node and forwarded it across the distribution system),and Addr4 identifies the original source.
- Addr1 corresponds to E, Addr2 identifies AP-3, Addr3 corresponds to AP- 1, and Addr4 identifies A.

2.4.BLUETOOTH:

- Bluetooth is used for short range wireless communication between devices like computers mobile phones etc .

- Bluetooth is a more convenient alternative to connecting two devices with wire.
- Bluetooth radio use low power transmission thus saving the communication resources
- Bluetooth operates in the 2.45GHz band.
- Bluetooth links have a bandwidth of around 1 to 2 Mbps and a rang of 10 m.
- Bluetooth is also called as personal area network (PAN)
- It can also be used for file transfer operations from one mobile phone to another.
- Bluetooth uses omnidirectional radio waves that can through walls or other non-metal barriers.
- Basic Bluetooth network configuration is called **piconet**
- **Piconet** consist of a master device and up to seven slave devices .this is shown in the below figure
- Any communication is between the master and the slave , slave do not communicate with each other.
- Slave have a simpler role , Bluetooth hardware and software can be simpler and cheaper.
- Bluetooth use a spread-spectrum technique to deal with possible interferences. It uses frequency hopping with 79 channels ,using each for 625 microseconds at a time.
- This provide a time slot for Bluetooth to use synchronous TDM
- There can be only one primary or master station in each piconet.
- The communication between the primary and the secondary can be one-to-one or one-to-many.



• All communication is between master and a slave. Salve-slave communication is not possible.

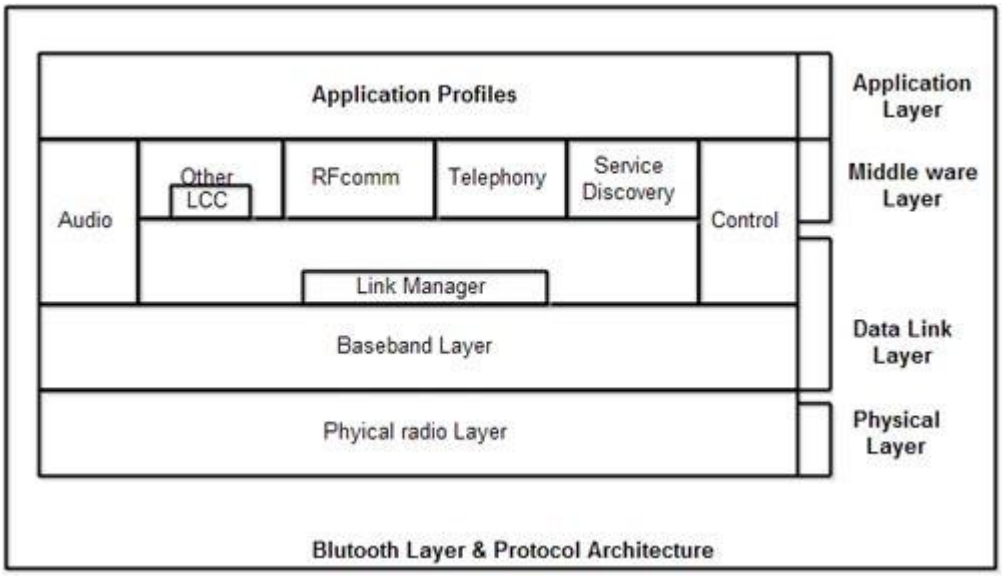
• In addition to seven active slave station, a piconet can have upto 255 parked nodes. These parked nodes are secondary or slave stations and cannot take part in communication until it is moved from parked state to active state.

Bluetooth layers and Protocol Stack

• Bluetooth standard has many protocols that are organized into different layers.

• The layer structure of Bluetooth does not follow OS1 model, TCP/IP model or any other known model.

• The different layers and Bluetooth protocol architecture.



Radio Layer

• The Bluetooth radio layer corresponds to the physical layer of OSI model.

• It deals with ratio transmission and modulation.

• The radio layer moves data from master to slave or vice versa.

• It is a low power system that uses 2.4 GHz ISM band in a range of 10 meters.

• This band is divided into 79 channels of 1MHz each. Bluetooth uses the Frequency Hopping Spread Spectrum (FHSS) method in the physical layer to avoid interference from other devices or networks.

• Bluetooth hops 1600 times per second, *i.e.* each device changes its modulation frequency 1600 times per second.

• In order to change bits into a signal, it uses a version of FSK called GFSK *i.e.* FSK with Gaussian bandwidth filtering.

Baseband Layer

• Baseband layer is equivalent to the MAC sublayer in LANs.

• Bluetooth uses a form of TDMA called TDD-TDMA (time division duplex TDMA).

• Master and slave stations communicate with each other using time slots.

• The master in each piconet defines the time slot of 625 μsec.

• In TDD- TDMA, communication is half duplex in which receiver can send and receive data but not at the same time.

• If the piconet has only no slave; the master uses even numbered slots (0, 2, 4, ...) and the slave uses odd-numbered slots (1, 3, 5, ...). Both master and slave communicate in half duplex mode. In slot 0, master sends & secondary receives; in slot 1, secondary sends and primary receives.

• If piconet has more than one slave, the master uses even numbered slots. The slave sends in the next odd-numbered slot if the packet in the previous slot was addressed to it.

• In Baseband layer, two types of links can be created between a master and slave. These are:

1. Asynchronous Connection-less (ACL)

• It is used for packet switched data that is available at irregular intervals.

• ACL delivers traffic on a best effort basis. Frames can be lost & may have to be retransmitted.

• A slave can have only one ACL link to its master.

• Thus ACL link is used where correct delivery is preferred over fast delivery.

• The ACL can achieve a maximum data rate of 721 kbps by using one, three or more slots.

2. Synchronous Connection Oriented (SCO)

• sco is used for real time data such as sound. It is used where fast delivery is preferred over accurate delivery.

• In an sco link, a physical link is created between the master and slave by reserving specific slots at regular intervals.

• Damaged packet; are not retransmitted over sco links.

• A slave can have three sco links with the master and can send data at 64 Kbps.

Logical Link, Control Adaptation Protocol Layer (L2CAP)

• The logical unit link control adaptation protocol is equivalent to logical link control sublayer of LAN.

• The ACL link uses L2CAP for data exchange but sco channel does not use it.

• The various function of L2CAP is:

1. Segmentation and reassembly

• L2CAP receives the packets of upto 64 KB from upper layers and divides them into frames for transmission.

• It adds extra information to define the location of frame in the original packet.

• The L2CAP reassembles the frame into packets again at the destination.

2. Multiplexing

• L2CAP performs multiplexing at sender side and demultiplexing at receiver side.

• At the sender site, it accepts data from one of the upper layer protocols frames them and deliver them to the Baseband layer.

• At the receiver site, it accepts a frame from the baseband layer, extracts the data, and delivers them to the appropriate protocol1ayer.

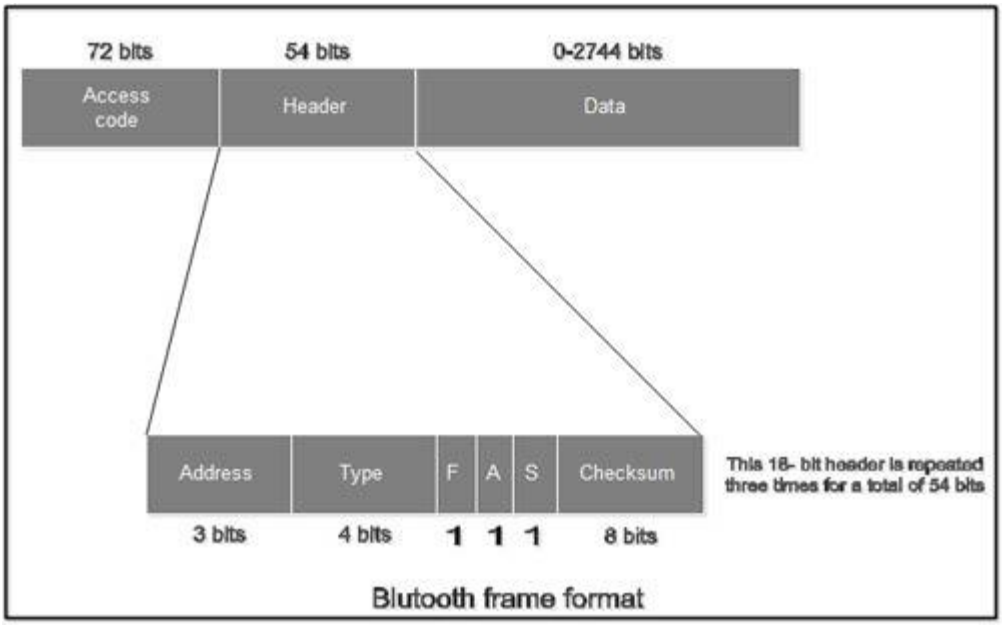
3. Quality of Service (QOS)

• L2CAP handles quality of service requirements, both when links are established and during normal operation.

• It also enables the devices to negotiate the maximum payload size during connection establishment.

Bluetooth Frame Format

The various fields of blue tooth frame format are:



1. **Access Code**: It is 72 bit field that contains synchronization bits. It identifies the master.

2. **Header**: This is 54-bit field. It contain 18 bit pattern that is repeated for 3 time.

The header field contains following subfields:

(i) **Address**: This 3 bit field can define upto seven slaves (1 to 7). If the address is zero, it is used for broadcast communication from primary to all secondaries.

(ii)Type: This 4 bit field identifies the type of data coming from upper layers.

(iii) **F**: This flow bit is used for flow control. When set to 1, it means the device is unable to receive more frames.

(iv) **A**: This bit is used for acknowledgement.

(v) **S**: This bit contains a sequence number of the frame to detect retransmission. As stop and wait protocol is used, one bit is sufficient.

(vi) **Checksum**: This 8 bit field contains checksum to detect errors in header.

3. **Data**: This field can be 0 to 2744 bits long. It contains data or control information coming from upper layers

2.5 SWITCHING AND BRIDGING:

A switch is a mechanism that allows us to interconnect links to form a large network. It is also called as a multi-input ,multi-output device that transfers packets from an input port to one or more output ports . A star topology has several attractive properties:

- Even though a switch has a fixed number of inputs and outputs, which limits the number of hosts that can be connected to a single switch, large networks can be built by interconnecting a number of switches.
- We can connect switches to each other and to hosts using point-to-point links.
- Adding a new host to the network by connecting it to a switch does not mean that the hosts already connected will get worse performance from the network.

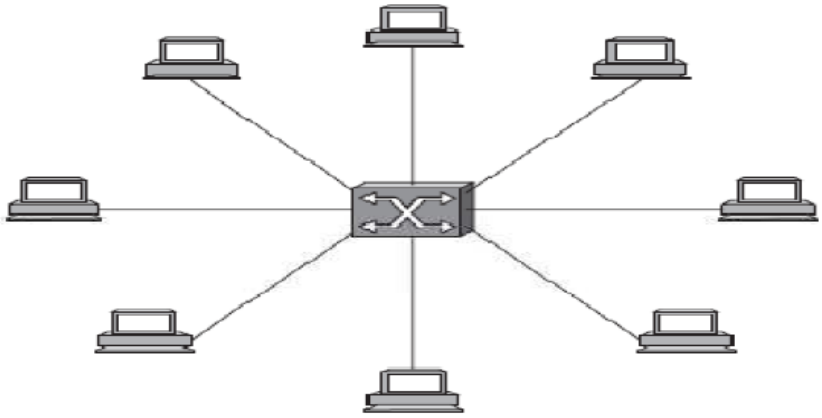


Fig 2.12 . A switch provides a star topology

- Switched networks are considered more scalable.
- 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 sometimes referred to as either switching or forwarding, and in terms of the OSI architecture, it is the main function of the network layer

.

Approaches for switching and forwarding:

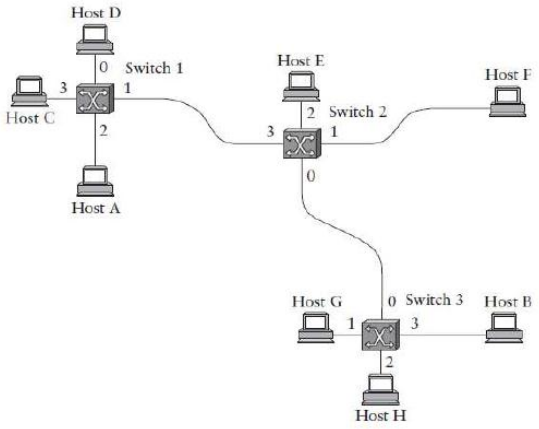
- The first is the datagram or connectionless approach. The second is the virtual circuit or connection-oriented approach.
- A third approach, source routing, is less common than these other two, but it is simple to explain and does have some useful applications .All host must have globally unique identifier.

DATAGRAMS:

It is sure that every packet contains enough information to enable any switch to decide how to get it to its destination. That is, every packet contains the complete destination address.

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

Datagram forwarding: an example network



Forwarding table for switch 2.

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

Fig 2.13: Datagram forwarding and table

Routing through datagrams:

- This particular table shows the forwarding information that switch 2 needs to forward datagram's in the example network.
- It is easy to draw such a table when a complete map of a simple network is known.
- It is a lot harder to create the forwarding tables in large, complex networks with dynamically changing topologies and multiple paths between destinations.
- That harder problem is known as routing.

Characteristics of the Connectionless(datagram) Networks

- A host can send a packet anywhere at any time. When a host sends a packet, it doesn't know if the network is capable of delivering it or if the destination host is 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 too have no serious effect on communication if it is possible to find an alternate route for the failure and update the forwarding table accordingly.

Virtual Circuit Switching:

- It uses the concept of a *virtual circuit.* which is also called a connection-oriented model, It requires that we first set up a virtual connection from the source host to the destination host before any data is sent.

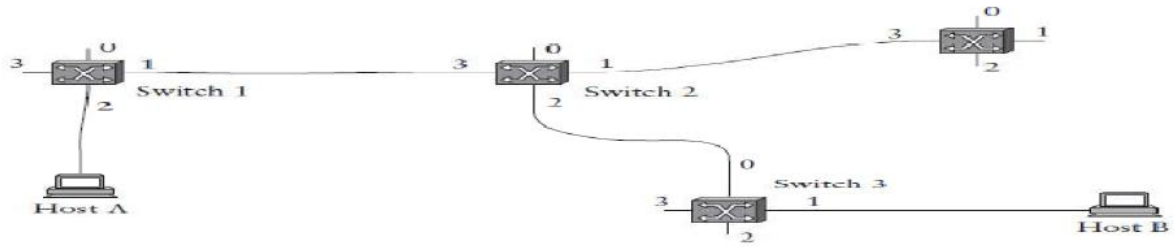


Fig 2.14: An example of a virtual circuit network

In this example where host A again wants to send packets to host B.

- We can think of this as a two-stage process.
- The first stage is ―connection setup.
- The second is data transfer.
- In the connection setup phase, it is necessary to establish connection in each of the switches between the source and destination hosts.

Virtual circuit table contains:

- A virtual circuit identifier (VCI) 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.

There are two broad classes of approach to establishing connection state.

1.Permanent virtual circuit (PVC)

It have a network administrator configure the state, in this case the virtualcircuit is called permanent(PVC) . It may be thought of as a long-lived or
administratively configured VC.

2.Switched Virtual Circuit (SVC)

Here a host send messages into the network to cause the state to be established. This is referred to as signaling , and the resulting virtual circuits are
said to be switched.

Example PVC construction

Let's assume that a network administrator wants to manually create a new virtual connection from host A to host B.The characteristic of a switched virtual circuit (SVC) is that a host may set up and delete such a VC dynamically without the involvement of a network administrator.

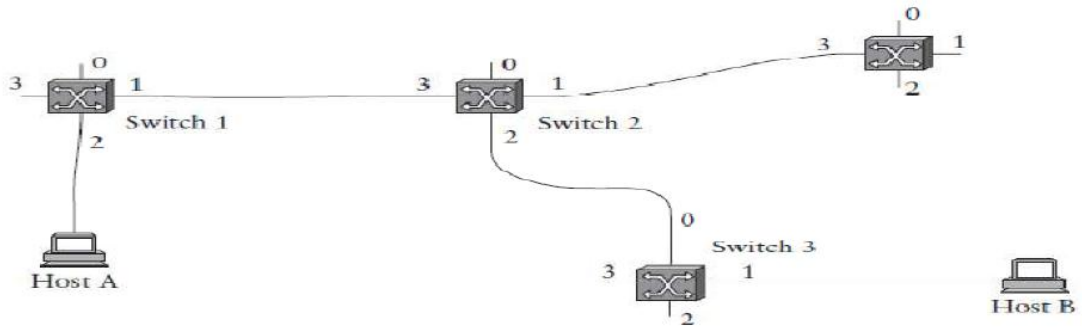


Fig 2.15 Example network

Note that an SVC should more accurately be called a ―signalled VC, since it is the use of signalling (not switching) .

Incoming Interface	Incoming VCI	Outgoing Interface	Outgoing VCI
2	5	1	11

(a)

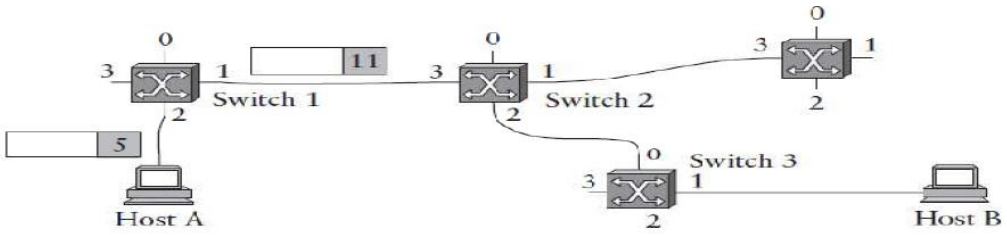
Incoming Interface	Incoming VCI	Outgoing Interface	Outgoing VCI
3	11	2	7

(b)

Incoming Interface	Incoming VCI	Outgoing Interface	Outgoing VCI
0	7	1	4

(c)

Fig 2.16 table



A packet makes its way through a virtual circuit network.

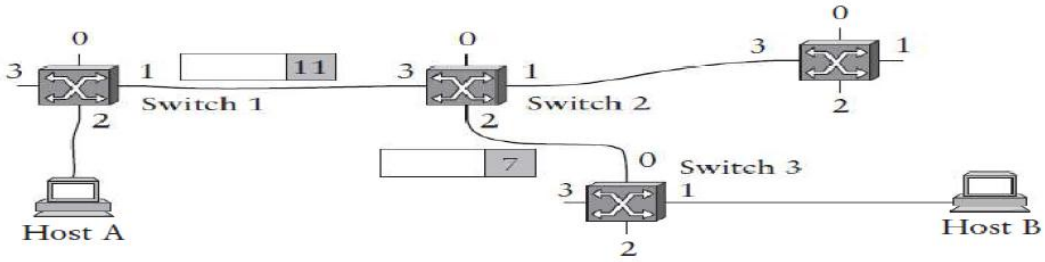


Fig 2.17 A packet is sent into a virtual circuit network and its path establishment

Svc construction:

- To start the signalling process, host A sends a setup message to the switch 1.The setup message contains, the complete destination address of host B.
- When switch 1 receives the connection request, in addition to sending it on to switch 2, it creates a new entry in its virtual circuit table for this new connection. This entry is exactly the same as shown previously in virtual circuit Table.
- The main difference is that now the task of assigning an unused VCI value on the interface is performed by the switch. In this example, the switch picks the value 5. The virtual circuit table now has the following information:
- When packets arrive on port 2 with identifier 5, send them out on port 1. Another issue is that, somehow, host A will need to learn that it should put the VCI value of 5 in packets that it wants to send to B.
- When switch 2 receives the setup message, it performs a similar process; in this example it picks the value 11 as the incoming VCI value.
- Similarly, switch 3 picks 7 as the value for its incoming VCI. Each switch can pick any number it likes, as long as that number is not currently in use for some other connection on that port of that switch.
- As noted above, VCIs have ―link local scope; that is, they have no global significance.

- Finally, the setup message arrives at host B. Assuming that B is healthy and willing to accept a connection from host A, it too allocates an incoming VCI value, in this case 4.
- This VCI value can be used by B to identify all packets coming from host A. When host A no longer wants to send data to host B, it tears down the connection by sending a teardown message to switch 1. The switch removes the relevant entry from its table and forwards the message on to the other switches in the path, which similarly delete the appropriate table entries.

The most popular examples of virtual circuit technologies are Frame Relay and asynchronous transfer mode.

BRIDGES AND LAN SWITCHES:

1. Switch that is used to forward packets between shared-media LANs such as Ethernets.
2. It is also called as LAN switches.
3. Historically they have also been referred to as bridges.
4. A pair of Ethernets interconnected by using repeater.
5. An alternative way is to put a node between the two Ethernets and have the node forward frames from one Ethernet to the other, this node is called bridge.
6. And a collection of LANs connected by one or more bridges is usually said to form an extended LAN.
7. This node would be in promiscuous mode.
8. For example, while a single Ethernet segment can carry only 10 Mbps of total traffic, an Ethernet bridge can carry as much as 10n Mbps, where n is the number of ports (inputs and outputs) on the bridge.

Learning Bridges:

Consider the bridge shown n below figure,

1. Whenever the bridge receives a frame on port 1 that is addressed to host A, it would not forward the frame to port 2; there is need because host A have already received the frame on the LAN connected to port 1.
2. If a frame addressed to host A was received on port 2, the bridge forward the frame to port 1.
3. Bridges forward frames on datagram model.
4. It also uses forwarding table to forward the incoming frames to the output node.
5. The idea is each bridge inspect the source address in all the frames it receives.

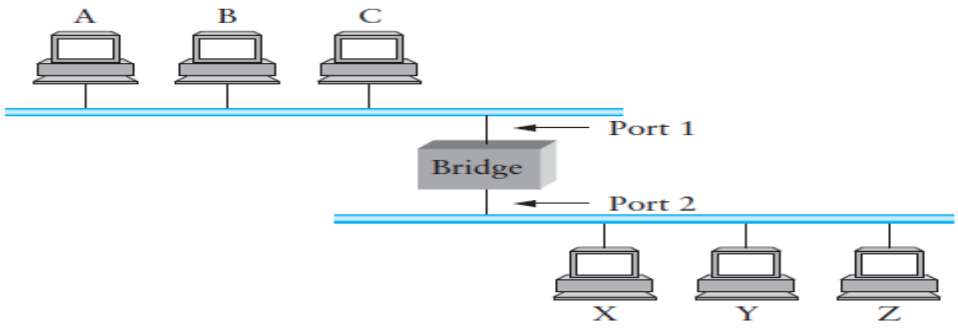


Fig 2.20: Illustration Of A Learning Bridge.

Forwarding Table:

1. When a bridge first boots, this table is empty, entries are added over time.
2.Also, a timeout is associated with each entry, and the bridge discards the entry after a specified period of time.

Host	Port
A	1
B	1
C	1
X	2
Y	2
Z	2

Fig:2.21 Forwarding table maintained by a bridge

Implementation:

1. Structure Bridge Entry defines a single entry in the bridge"s forwarding table.
2. These are stored in a Map structure. (which supports mapCreate, mapBind, and MapResolve operations.
3. The constant MAX TTL specifies how long an entry is kept in the table.

Table Creation Routine

- The routine that updates the forwarding table when a new packet arrives is given by updateTable.

```
#define BRIDGE_TAB_SIZE  1024 /* max. size of bridging table */
#define MAX_TTL           120 /* time (in seconds) before an
                                 entry is flushed */

typedef struct {
    MacAddr     destination;  /* MAC address of a node */
    int         ifnumber;     /* interface to reach it */
    u_short     TTL;          /* time to live */
    Binding     binding;      /* binding in the Map */
} BridgeEntry;

int     numEntries = 0;
Map     bridgeMap = mapCreate(BRIDGE_TAB_SIZE,
                              sizeof(BridgeEntry));
```

- The arguments passed are the source MAC address contained in the packet and the interface number on which it was received.
- Table is Updated when new Entries are added.

Spanning Tree Algorithm

1) Loops in a extended LAN, purpose—to provide redundancy in case of failure.
2) bridges must be able to correctly handle loops.
3) This problem is addressed by having the distributed spanning tree algorithm.
4) If you think of the extended LAN as being represented by a graph that possibly has loops (cycles).
5) Then a spanning tree is a sub-graph of this graph that covers (spans) all the vertices, but contains no cycles.
6) That is, a spanning tree keeps all of the vertices of the original graph, but throws out some of the edges.

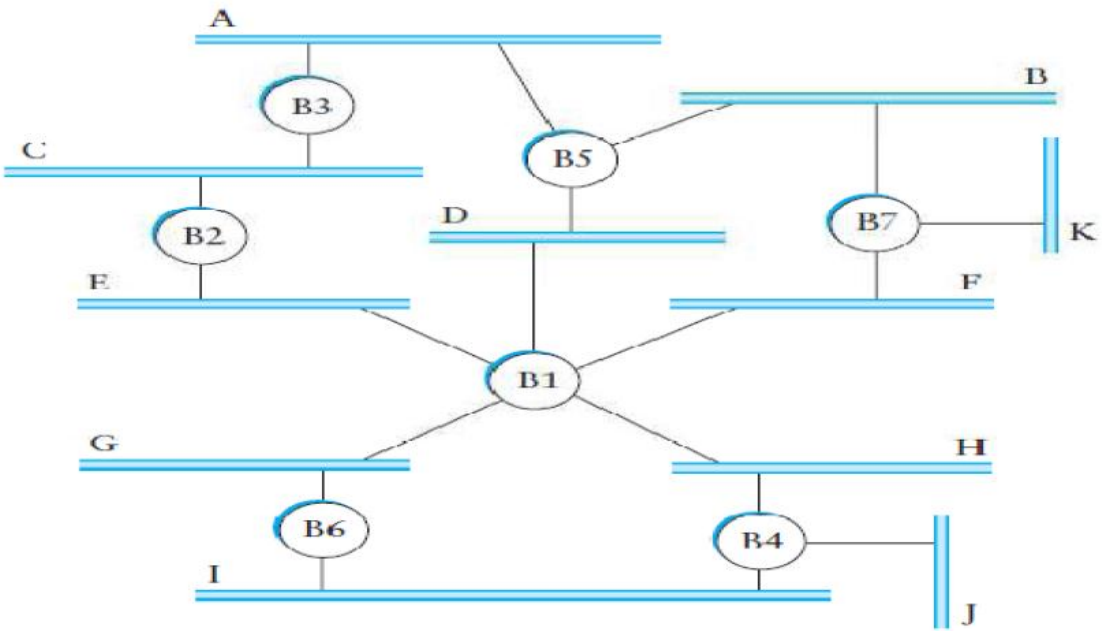


Fig2.22 :Extended LAN with loops

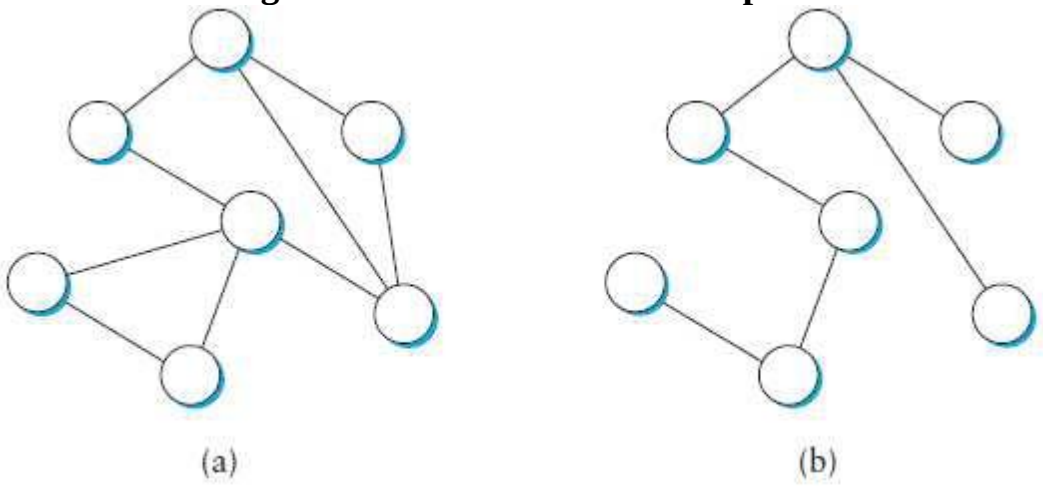


Fig 2.23 : a) a cyclic graph; (b) a corresponding spanning tree

1. The spanning tree algorithm, which was developed by Radia Perlman at Digital.
2.It is a protocol used by a set of bridges to agree upon a spanning tree for a particular extended LAN.
3. The main idea of the spanning tree is for the bridges to select the ports over which they will forward frames.
4. The algorithm selects ports as follows. Each bridge has a unique identifier; for our purposes, we use the labels B1, B2, B3, and so on.
5. The algorithm first elects the bridge with the smallest id as the root of the spanning tree; exactly how this election takes place is described below.
6. The root bridge always forwards frames out over all of its ports.

Path Selection

1. Each bridge computes the shortest path to the root and notes which of its ports is on this path
2. This port is also selected as the bridge's preferred path to the root.
3. Finally, all the bridges connected to a given LAN elect a single designated bridge that will be responsible for forwarding frames toward the root bridge.
4. Designated bridge is the one that is closest to the root, and if two or more bridges are equally close to the root, then the bridges‟ identifiers are used to break ties; the smallest id wins.

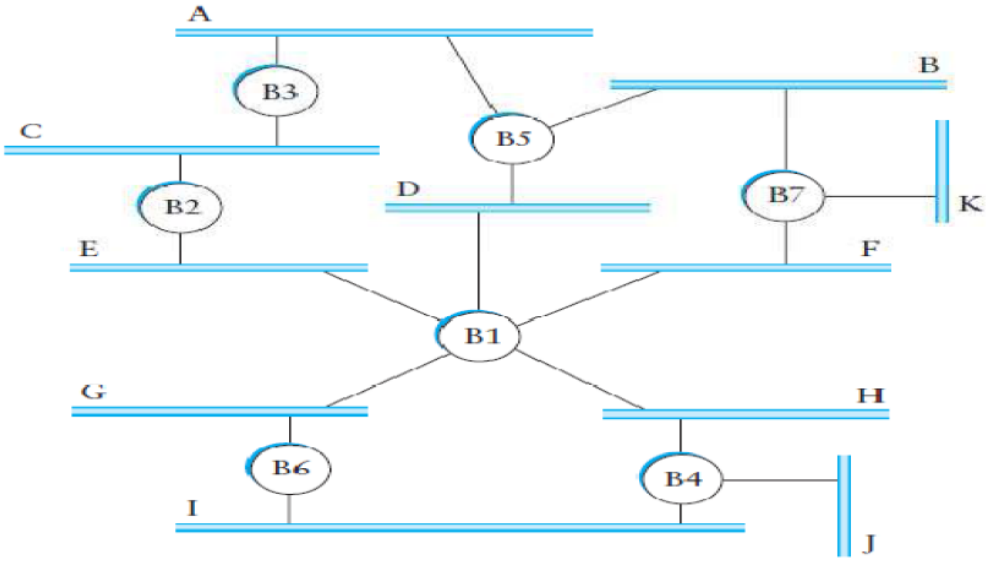


Fig2.24:Extended LAN with loops.

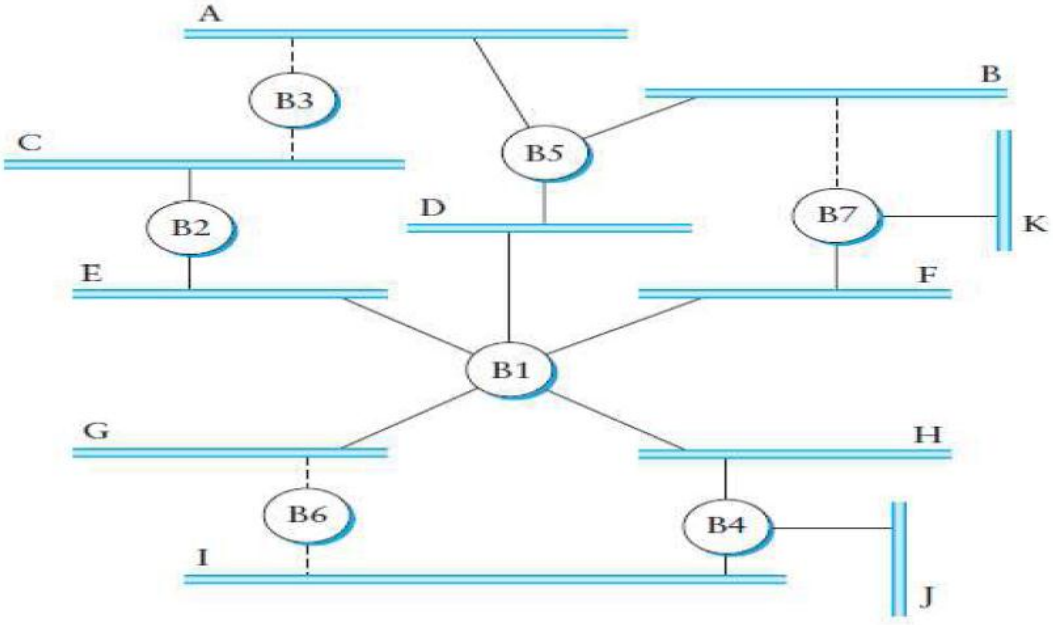


Fig2.25: Spanning tree with some ports not selected

1. B1 is the root bridge, since it has the smallest id.

2. Notice that both B3 and B5 are connected to LAN A, but B5 is the designated bridge since it is closer to the root.
3. Similarly, both B5 and B7 are connected to LAN B, but in this case, B5 is the designated bridge since it has the smaller id; both are an equal distance from B1.
4. The bridges in an extended LAN do not have the luxury of being able to see the topology of the entire network.
5. Instead, the bridges have to exchange configuration messages with each other and then.
6. Based on this the root or a designated bridge will be decide.

The configuration messages contain three pieces of information.

1. The id for the bridge that is sending the message
2. The id for what the sending bridge believes to be the root bridge
3. The distance, measured in hops, from the sending bridge to the root bridge.

Initially, each bridge thinks it is the root and so it sends a configuration message out on each of its ports identifying itself as the root and giving a distance to the root of 0. Upon receiving a configuration message over a particular port, the bridge
checks to see if that new message is better than the current best configuration message recorded for that port. The new configuration message is considered "better" than the currently recorded information if

- It identifies a root with a smaller id or
- It identifies a root with an equal id but with a shorter distance or
- The root id and distance are equal, but the sending bridge has a smaller id.
- If the new message is better than the currently recorded information, the bridge discards the old information and saves the new information.
- However, it first adds 1 to the distance-to-root field since the bridge is one hop farther away from the root than the bridge that sent the message.

1. B3 receives (B2, 0, B2).
2. Since 2 < 3, B3 accepts B2 as root.
3. B3 adds one to the distance advertised by B2 (0) and thus sends (B2, 1,B3)toward B5.
4. Meanwhile, B2 accepts B1 as root because it has the lower id, and it sends (B1, 1, B2) toward B3.

5. B5 accepts B1 as root and sends (B1, 1, B5) toward B3.

6. B3 accepts B1 as root, and it notes that both B2 and B5 are closer to the root.

7. than it is. Thus B3 stops forwarding messages on both its interfaces.

Broadcast and Multicast

- Bridges must also support these broadcast and multicast transmissions.
- Broadcast is simple—each bridge forwards a frame with a destination broadcast address out on each active (selected) port other than the one on which the frame was received.
- Multicast can be implemented in exactly the same way, with each host deciding for itself whether or not to accept the message.
- Not all the LANs in an extended LAN necessarily have a host that is a member of a particular multicast group. Consider the figure spanning tree with some ports are not selected.
- In that a frame sent to group M by a host on LAN A , If there is no host on LAN J that belongs to group M, then there is no need for bridge B4 to forward the frames over that network.
- On the other hand, not having a host on LAN H that belongs to group M

does not necessarily mean that bridge B1 can avoid forwarding multicast frames onto LAN H. It all depends on whether or not there are members of group M on LANs I and J.

It learns exactly the same way that a bridge learns whether it should forward a unicast frame over a particular port—by observing the source addresses that it receives over that port. In particular, each host that is a member of group M must periodically send a frame with the address for group M in the source field of the frame header. This frame would have as its destination address the multicast address for the bridges.

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.

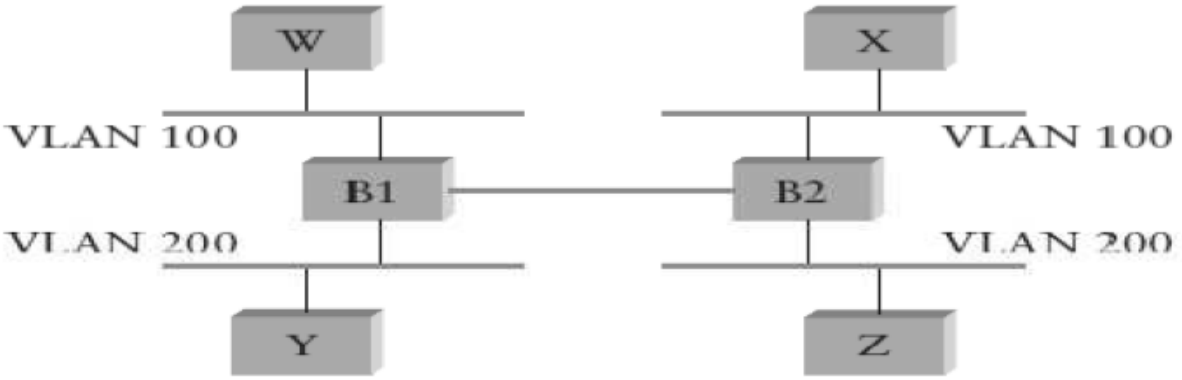


Fig 2.26:Two virtual LANs share a common backbone

- 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.
- Now let‟s suppose that we define the segments connected to hosts W and X as being in one VLAN, which we‟ll call VLAN 100. We also define the segments that connect to hosts Y and Z as being in VLAN 200.
- To do this, we need to configure a VLAN ID on each port of bridges B1 and B2. The link between B1 and B2 is considered to be in both VLANs.
- When a packet sent by host X arrives at bridge B2, the bridge observes that it came in a port that was configured as being in VLAN 100.
- It inserts a VLAN header between the Ethernet header and its payload. The interesting part of the VLAN header is the VLAN ID; in this case, that ID is set to100.
- The bridge now applies its normal rules for forwarding to the packet, with the extra restriction that the packet may not be sent out an interface that is not part of VLAN 100.
- Thus, under no circumstances will the packet—even a broadcast packet— be sent out the interface to host Z, which is in VLAN 200. The packet is, however, forwarded to bridge B1, which follows the

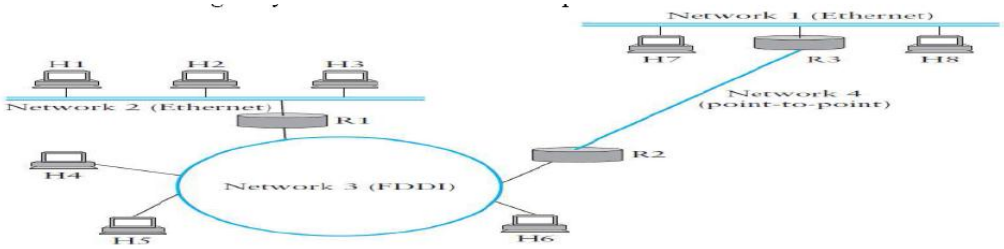
same rules, and thus may forward the packet to host W but not to host Y.

- Their main advantage is that they allow multiple LANs to be transparently connected; that is, the networks can be connected without the end hosts having to run any additional protocols.
- Disadvantage: The latency between any pair of hosts on an extended LAN becomes both larger and more highly variable.

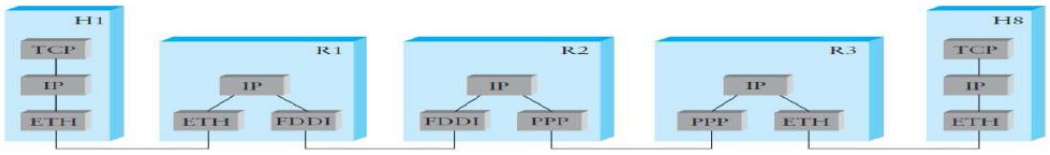
2.6 BASIC INTERNETWORKING:

2.6.1 INTERNET PROTOCOL (IP):

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.



A simple internetwork. H*n* = host; R*n* = router.



A simple internetwork, showing the protocol layers used to connect H1 to H8 in the previous

Fig2.27 A simple internetwork and the protocol layer used to connect H1to H8

Service model- It defines the host-to-host services you want to provide. It has two parts: an **addressing scheme**, and datagram model.

Datagram (connectionless) model:

This service model is called *best effort*.

Datagram Delivery

- The IP datagram is fundamental to the Internet Protocol.

- Datagram is a type of packet that can be sent in a connectionless manner over a network.
- Every datagram carries enough information to the network for forwarding 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.
- The best-effort means that if something goes wrong and the packet gets lost, corrupted, misdelivered, or in any way fails to reach its destination, the network does nothing. It does not make any attempt to recover from the failure. This is sometimes called an *unreliable* service.
- Best-effort, connectionless service is the simplest service from an internetwork.
- Best-effort delivery does not just mean that packets can get lost. Sometimes they can get delivered out of order, and sometimes the same packet can get delivered more than once. The higher-level protocols or applications that run above IP need to be aware of all these possible failure modes.

Packet Format:

The packet format is shown in below figure,

Version: This field-denotes version of IP. The current version of IP is 4, and it is sometimes called IPv4.

HLen:- It specifies the length of the header in 32-bit words. When there are no options, which is most of the time, the header is 5 words (20 bytes) long.

TOS (8 bits-type of service) field - its basic function is to allow packets to be treated differently based on application needs.

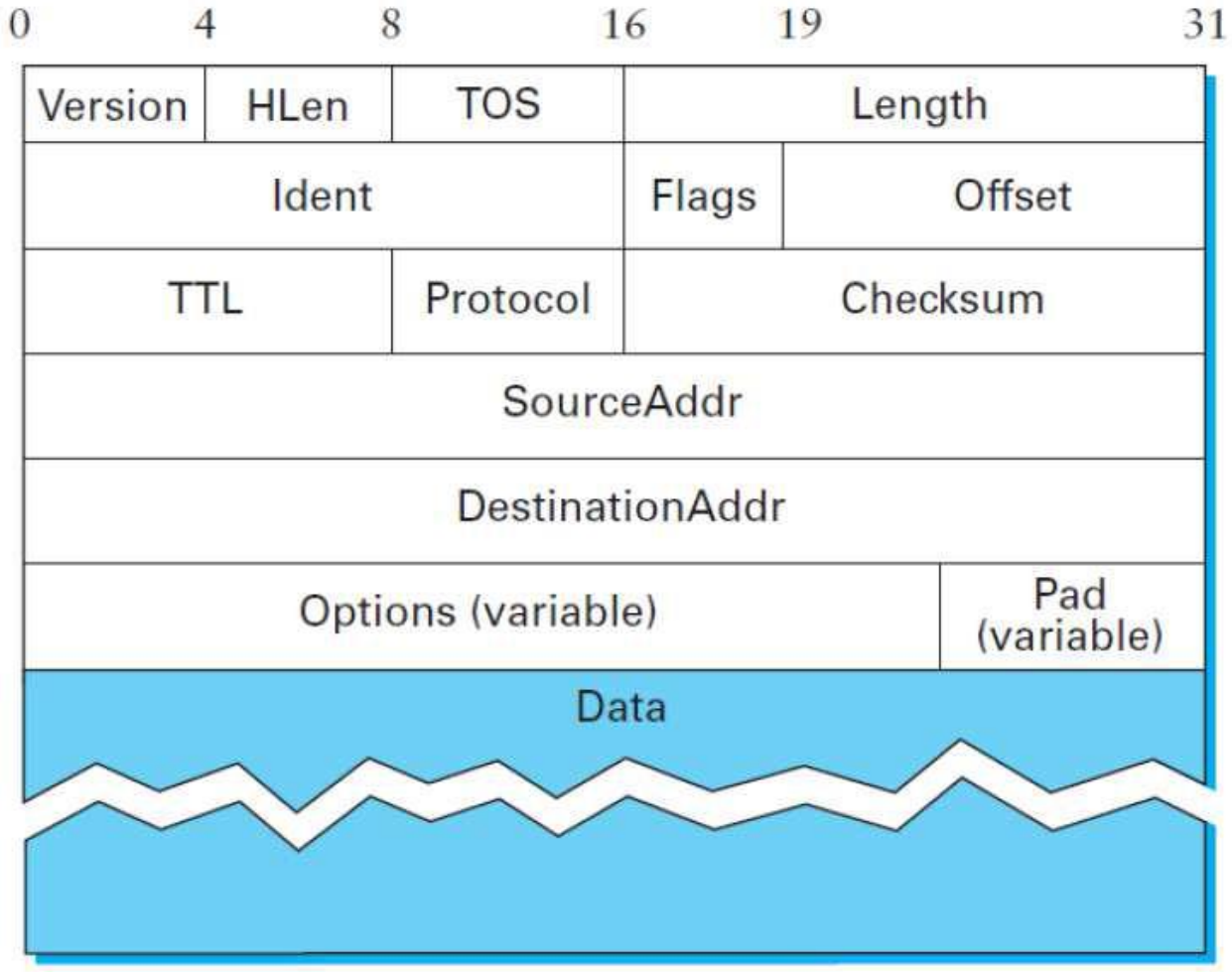


Fig:2.28 Frame format

Length (16 bits)-length of the datagram, including the header. Unlike the HLen field, the Length field counts bytes rather than words. Thus, the maximum size of an IP datagram is 65,535 bytes. IP supports Fragmentation and Reassembly .

Ident field-to identify the fragments, all the fragments have same id , it is same as original packet's id.

Flags field –it has three bits first bit is always 0(default),second bit is DF(don't

fragment)-denote no need to fragment the packet, third bit is More Fragment-denote whether the current fragment has follow up fragments or not(if it is 1-more fragments,0-this is the last fragment).

Offset field-meaning that there are more fragments to follow, and it sets the Offset to 0, since this fragment contains the first part of the original

datagram. The data carried in the second fragment starts with the 513th byte of the original data, so the Offset field in this header is set to 64

TTL field(time to live)- was set to a specific number of seconds that the packet would be allowed to live and routers along the path would decrement this field until it reached 0.

The **Protocol** field- is simply a demultiplexing key that identifies the higher-level

protocol to which this IP packet should be passed. There are values defined for TCP (6), UDP (17

The **Checksum** field(16 bits)-Error detection bits.

Source address-to identify the source.

Destination address-to identify the destination.

Finally, there may be a number of **options** at the end of the header. The presence or absence of options may be determined by examining the header length (HLen) field. While options are used fairly rarely, a complete IP implementation must handle them all.

Fragmentation and Reassembly:

In a heterogeneous network collection each network has its own packet size.

For example, an Ethernet can accept packets up to 1500 bytes long, while FDDI packets may be 4500 bytes long. Therefore the IP service model should provide datagrams of small size that can fit inside one packet on any network technology, or it provide a means of segmentation and reassembly

For example, two hosts connected to FDDI networks that are interconnected by a point-to-point link need to send packets small to fit on an Ethernet.

The central idea here is that every network type has a maximum transmission unit (MTU), which is the largest IP datagram that it can carry in a frame. Note that this value is smaller than the largest packet size on that network because the IP datagram needs to fit in the payload of the link-layer frame. When a host sends an IP datagram, it can choose any size that it wants

Fragmentation is necessary if the path to the destination includes a network with a smaller MTU. Fragmentation occurs in a router when it receives a datagram that it wants to forward over a network that has an MTU that is smaller than the received datagram .To reassembled the fragments at the receiving host, they all carry the same identifier in the Ident field. This identifier is chosen by the sending host and is intended to be unique among all the datagrams that might arrive at the destination from this source over

some reasonable time period. IP does not attempt to recover from missing fragments.

Let us see the example when host H1 sends a datagram to host H8 as shown in the following Figure. 1500 bytes for the two Ethernets, 4500 bytes for the FDDI network, and 532 bytes for the point-to-point network, then a 1420-byte datagram sent from H1 moves through the first Ethernet and the FDDI network without fragmentation but must be fragmented into three datagrams at router R2. These three fragments are then forwarded by router R3 across the second Ethernet to the destination host .

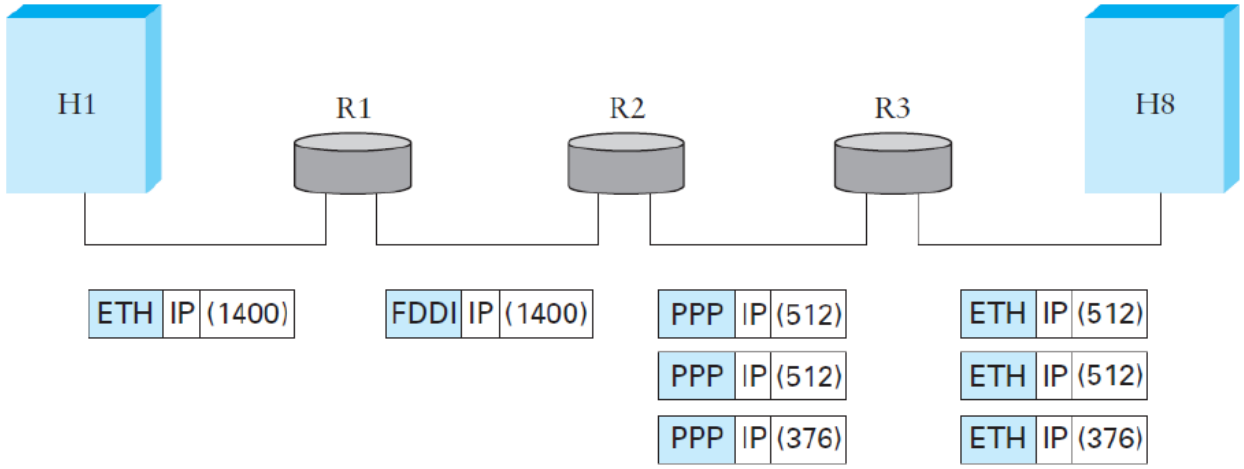


Fig.2.29.Example network

Two important points regarding Fragmentation and Reassembly:

1. Each fragment is itself a self-contained IP datagram that is transmitted over a sequence of physical networks, independent of the other fragments.

2 Each IP datagram is re-encapsulated for each physical network over which it travels. Header fields used in IP fragmentation.

(a) Unfragmented packet; (b) frag-mented packets.

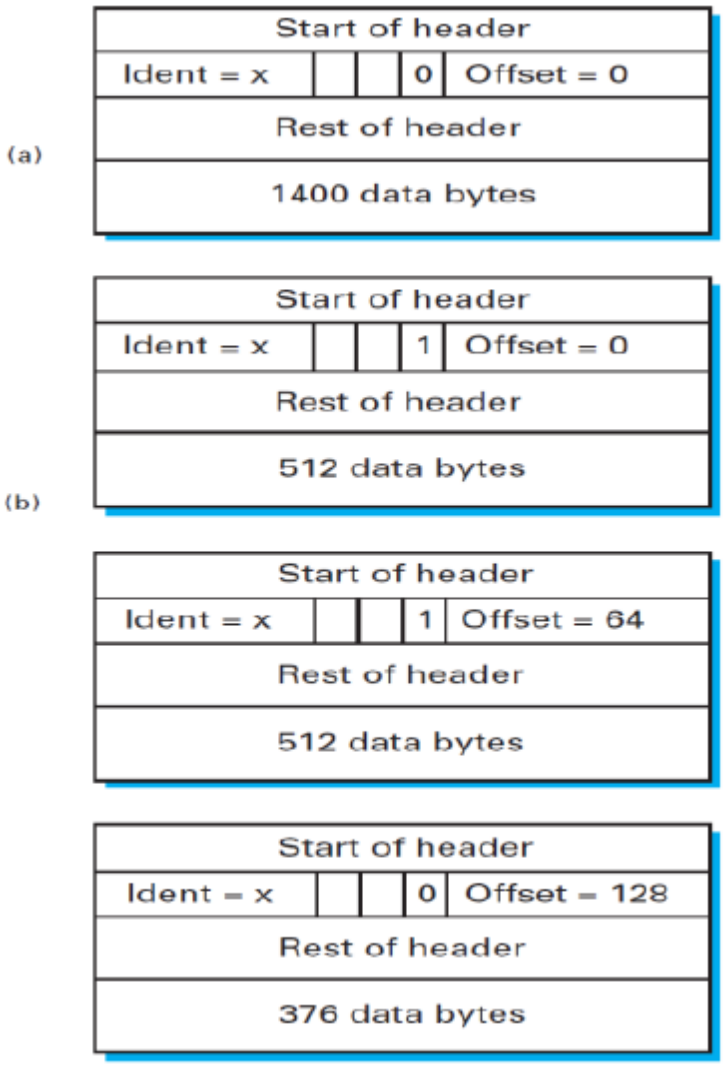


Fig2.30.(a) Unfragmented packet; (b) frag-mented packets.

Implementation

First, we define the key data structure (FragList) that is used to hold the individual fragments that arrive at the destination. Incoming fragments are saved in this data structure until all the fragments in the original datagram have arrived, at which time they are reassembled into a complete datagram and passed up to some higher-level protocol.

Global Addresses:

IP addresses are hierarchical, which means that they are made up of several parts IP addresses consist of two parts, a network part and a host part. The network part of an IP address identifies the network to which the host is attached; The host part then identifies each host uniquely on that

particular network. IP addresses are divided into three classes as class A, class B and class C.

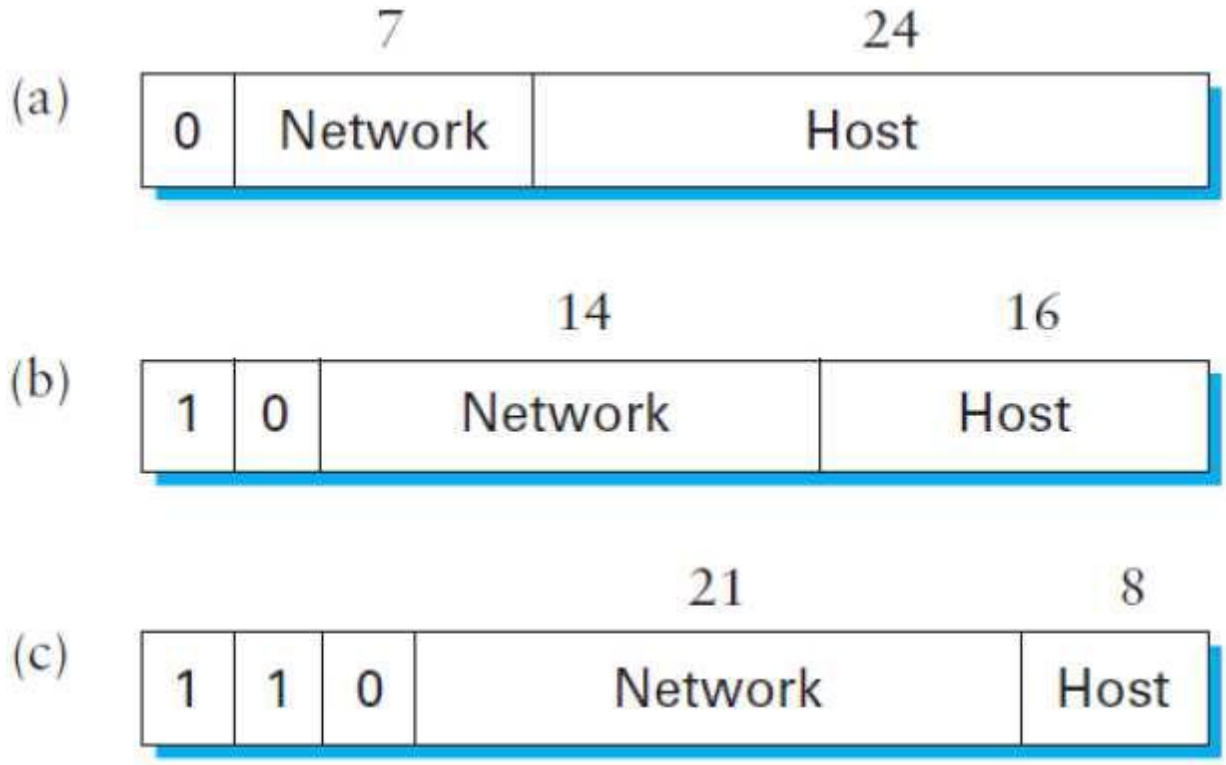


Fig.2.31 IP addresses : (a)class A;(b)class B;(c)class C.

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, Class B addresses allocate 14 bits for the network and 16 bits for the host, and class C addresses have only 8 bits for the host and 21 for the network part.

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:

Forwarding is the process of taking a packet from an input and sending it out on the appropriate outputThe main points in forwarding IP datagrams are the following:

- Every IP datagram contains the IP address of the destination host.
- The network part of an IP address uniquely identifies a single physical network that is part of the larger Internet.
- All hosts and routers that share the same network part of their address are connected to the same physical network and can thus communicate with each other by sending frames over that network.
- Every physical network that is part of the Internet has at least one router

Forwarding IP datagram:

- A datagram is sent from a source host to a destination host, through several routers along the way.
- Any node, whether it is a host or a router, first tries to know whether it is connected to the same physical network as the destination.
- To do this, it compares the network part of the destination address with the network part of its address
- If the node is not connected to the same physical network as the destination node, then it needs to send the datagram to a router.
- In general, each node will have a choice of several routers, and so it needs to pick the best one, or at least one that has a reasonable chance of getting the datagram closer to its destination.
- The router that it chooses is known as the next hop router. The router finds the correct next hop by consulting its forwarding table.
- The forwarding table is conceptually just a list of _NetworkNum, NextHop_ pairs.

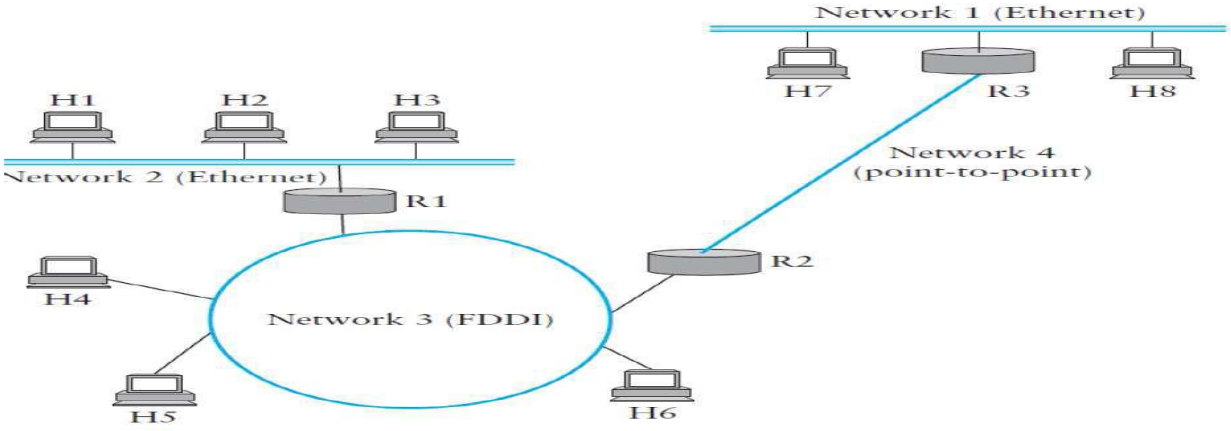


Fig 2.32.

Consider the above figure First, suppose that H1 wants to send a datagram to H2. Since they are on the same physical network, H1 and H2

have the same network number in their IP address. Thus, H1 delivers the datagram directly to H2 over the Ethernet.

Now suppose H1 wants to send a datagram to H8. Since these hosts are on different physical networks, they have different network numbers, so H1 sends the datagram to a router. R1 is the only choice (the default router) ,so H1 sends the datagram over the Ethernet to R1. R1's default router is R2; R1 then sends the datagram to R2 over the token ring network.

FORWARDING TABLE FOR ROUTER R2

NetworkNum	NextHop
1	R3
2	R1

COMPLETE FORWARDING TABLE FOR R2

NetworkNum	NextHop
1	R3
2	R1
3	Interface 1
4	Interface 0

2.6.2 CLASSLESS ROUTING (CIDR)

- Classless inter-domain routing (CIDR,) is a technique that addresses two issues in the Internet:
    1. The growth of routing tables as more and more network numbers need to be stored in them
    2. The 32-bit IP address space has to be exhausted 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, giving an address assignment efficiency of 2/255 = 0.78%; a network with 256 hosts needs a class B address, for an efficiency of only 256/65,535 = 0.39%.
- Even though subnetting can help us to assign addresses carefully, it does not help for more than 255 hosts
- One way to deal with that is to say no to any AS that requests a class B address unless they need 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, that means 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 those networks is the same.
- If we had assigned a class B address to the AS, the same routing information could be stored in one table entry. However, our address assignment efficiency would then be only 16 × 255/65,536 = 6.2%.
- This problem can be overcome by using CIDR.
- CIDR, minimize the number of routes that a router needs to know .
- To do this, CIDR helps us to *aggregate* routes. That is, it lets us use a single entry in a forwarding table to tell us how to reach a lot of different networks.
- From the name, it does this by breaking 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 the top 20 bits of all the addresses in this range are the same (11000000 00000100 0001). Thus, what we have effectively created is a 20-bit network number—something that

is between a class B network number and a 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.
- Consider the example Figure The two corporations served by the provider network have been assigned adjacent 20- bit network prefixes. Route Aggregate with CIDR

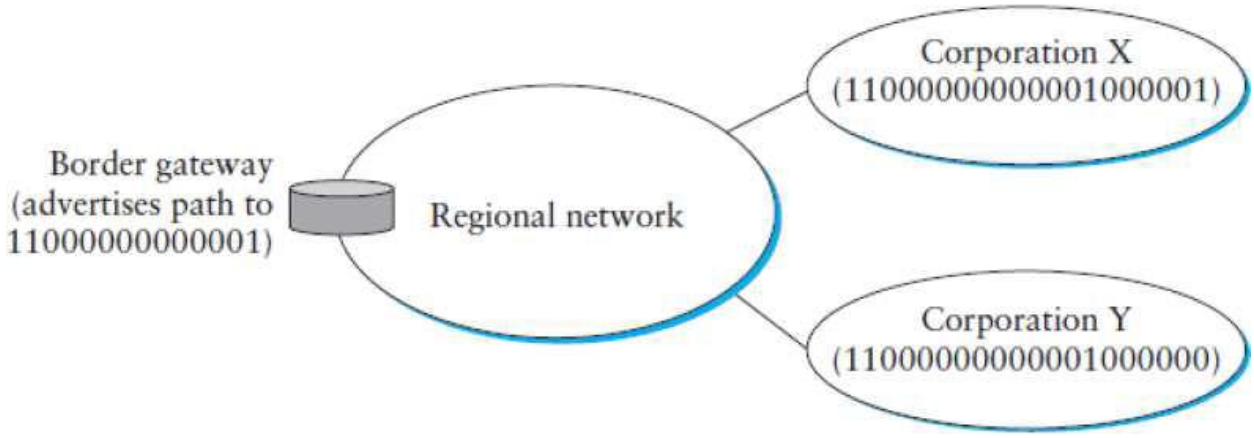


Fig 2.33

- 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.

2.6.3 ADDRESS RESOLUTION PROTOCOL (ARP):

The main issue is that IP is that the datagram contain IP addresses, which can not be understand the destination host, therefore the IP address have to Thus, we need to translate the IP address to a link-level address . So the IP data gram is
encapsulate with link-level address and send to the destination or to a router that forward the datagram to destination.

One simple way to map an IP address into a physical network address is to encode a host's physical address .

- For example, a host with physical address 00100001 01001001 (which has the decimal value 33 in the upper byte and 81 in the lower byte) might be given the IP address 128.96.33.81.
- In this case the networks physical address can not be greater than 16 bits therefore it will not work for 48-bit Ethernet addresses.
- A general solution is to maintain a table of address pairs, which maps IP address to physical address.
- This table is managed by the system administrator. This can be accomplished using the Address Resolution Protocol (ARP).
- The goal of ARP is to enable each host on a network to build up a table of mappings between IP addresses and link-level addresses.
- Since these mappings may change over time ,the entries are timed out periodically and removed. This happens on the order of every 15 minutes. The set of mappings currently stored in a host is known as the ARP cache or ARP table.
- . If a host wants to send an IP datagram to a host (or router) that it knows to be on the same network (i.e., the sending and receiving node have the same IP network number), it first checks for a mapping in the cache.
- If no mapping is found, it invoke the Address Resolution Protocol over the network. It does this by broadcasting an ARP query onto the network.
- This query contains the IP address of the target IP . Each host receives the query and checks to see if it matches its IP address. If it match, the host sends a response message that contains its link-layer address back to the originator of the query. The originator adds the information contained in this response to its ARP table.
- The query message also includes the IP address and link-layer address of the sending host. Thus, when a host broadcasts a query message, each host on the network can learn the sender's link level and IP addresses and place that information in its ARP table.

- However, not every host adds this information to its ARP table. If the host already has an entry for that host in its table, it ―refreshes this entry; that is, it resets the length of time until it discards the entry.

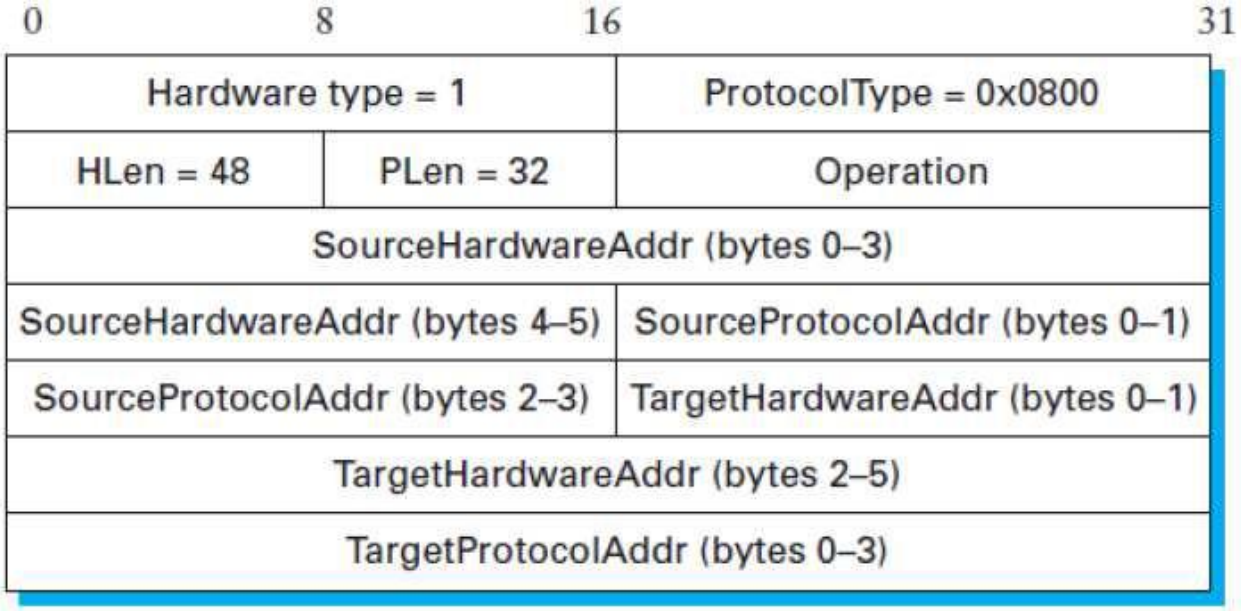


Fig:2.34 ARP packet format for mapping IP addresses into Ethernet addresses.

The ARP packet shown above contains:

- a Hardware Type field, which specifies the type of physical network (e.g.Ethernet).
- a ProtocolType field, which specifies the higher-layer protocol (e.g., IP).
- HLen (―hardware address length) and PLen (―protocol address length) fields,which specify the length of the link-layer address and higher-layer protocol address, respectively.
- an Operation field, which specifies whether this is a request or a response.

the source and target hardware (Ethernet) and protocol (IP) addresses.

2.6.4.DYNAMIC HOST CONTROL PROTOCOL(DHCP)

Most host operating systems provide a way for a system administrator, or a user, to manually configure the IP information needed by a host. The drawbacks of such manual configuration are.

- Lot of work have to be done to configure all the host in a large network directly.

- Also it is more prone to error, because it is necessary to make sure that every host gets a correct network number, no two network get the same IP address.

For these reasons, automated configuration methods are required. The primary method uses a protocol known as the Dynamic Host Configuration Protocol (DHCP).

- DHCP depends on a DHCP server that is responsible for providing configuration information to hosts. There is at least one DHCP server for an administrative domain.
- At the simplest level, the DHCP server can function just as a centralized repository for host configuration information. Consider, for example, the problem of administering addresses in the internetwork of a large company.
- DHCP saves the network administrators from walking to every host in the company with a list of addresses and network map in hand and configuring each host manually.
- Instead, the configuration information for each host could be stored in the DHCP server and automatically retrieved by each host when it is booted or connected to the network.
- A more sophisticated use of DHCP saves the network admininstrator from even having to assign addresses to individual hosts. In this model, the DHCP server maintains a pool of addresses that it handed over to hosts on demand.
- This considerably reduces the amount of configuration an administrator must do
- To contact a DHCP server, a newly booted or attached host sends a DHCPDISCOVER message to a special IP address (255.255.255.255) that is an IP broadcast address.
- This means it will be received by all hosts and routers on that network. one of these nodes is the DHCP server for the network.
- The server then reply to the host. This is shown in the below figure.

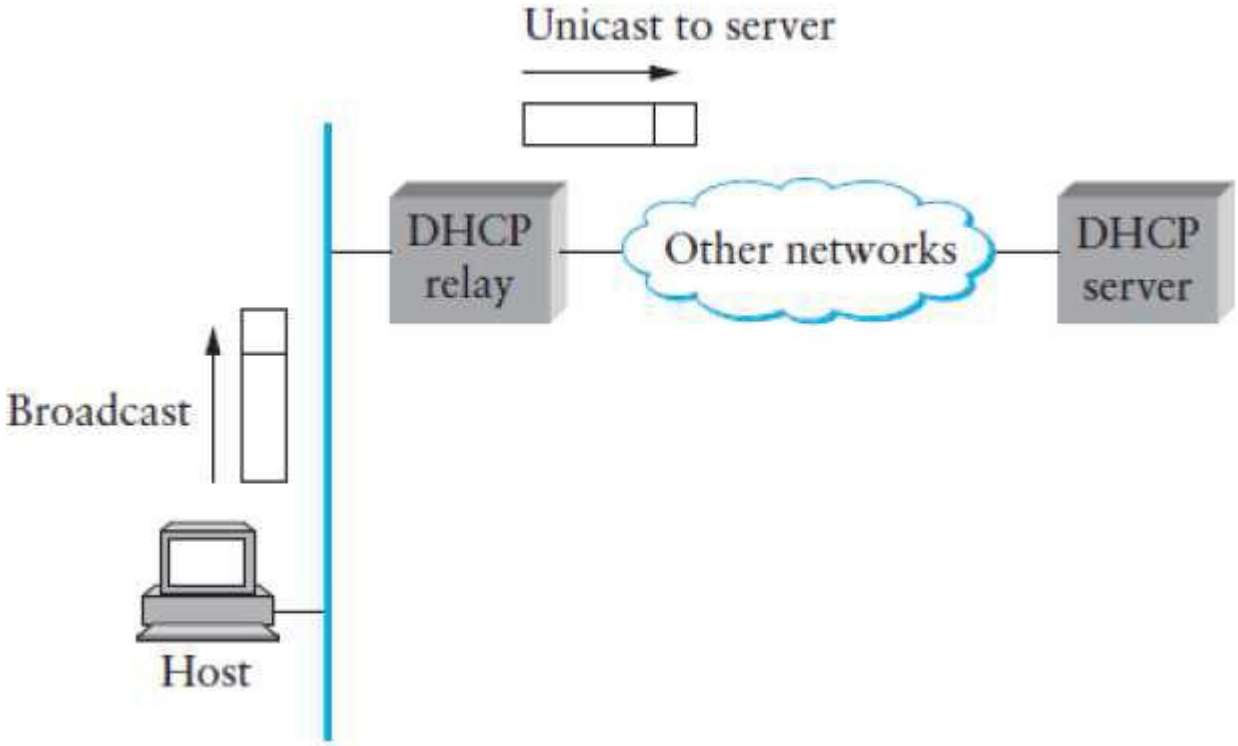


Fig 2.35

- When a relay agent receives a DHCPDISCOVER message, it unicasts it to the DHCP server and awaits the response, which it will then send back to the requesting client. A DHCP relay agent receives a broadcast DHCPDISCOVER message from a host and sends a unicast DHCPDISCOVER message to the DHCP server.

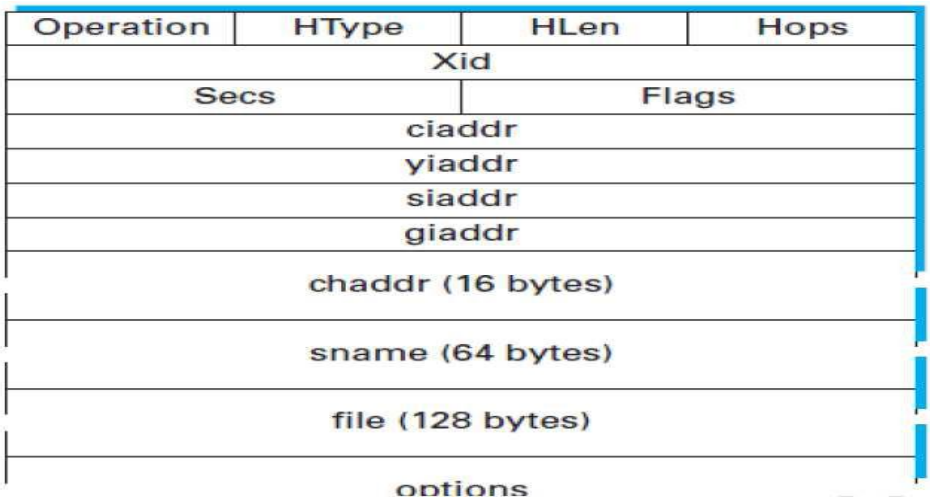


Fig:2.36 DHCP packet format:

2.6.5 INTERNET CONTROL MESSAGE PROTOCOL(ICMP)

- IP is always configured with a companion protocol, known as the Internet Control Message Protocol (ICMP),
- ICMP 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.
- For example, ICMP defines error messages indicating that the destination host is unreachable (perhaps due to a link failure), that the reassembly process failed, that the TTL had reached 0, that the IP header checksum failed, and so on.
- ICMP also defines a handful of control messages that a router can send back to a source host.
- One of the most useful control messages, called an ICMP-Redirect, tells the source host that there is a better route to the destination.
- ICMP-Redirects are used in the following situation. Suppose a host is connected to a network that has two routers attached to it, called R1 and R2, where the host uses R1 as its default router. If R2 is a better choice for a particular destination address, it sends an ICMP-Redirect back to the host, instructing it to use R2 for all future datagrams addressed to that destination.
- The host then adds this new route to its forwarding table.
- ICMP also provides the basis for two widely used debugging tools ,ping and trace route
- ping uses ICMP echo message to determine if a node is reachable and alive.
- Traceroute uses techniques to determine the set of routers along the path to destination.

UNIT-III ROUTING

Routing (RIP, OSPF, metrics) – Switch basics – Global Internet (Areas, BGP, IPv6), Multicast – addresses – multicast routing (DVMRP, PIM)

3.1.ROUTING:

The routers and switches forward the received packet to the destination by looking at the destination address , they select the output ports which are the best choice for reaching the destination by using a routing table or forwarding table . The fundamental problem in routing is how the routers get the information in the routing table .

- A forwarding table must contain enough information for packet forwarding that is the a row in the forwarding table contains the mapping from a network number to an outgoing interface and some MAC information, such as the Ethernet address of the next hop.
- The routing table, is built up by the routing algorithms. It generally contains mappings from network numbers to next hops. It may also contain information about how this information was learned, so that the router will be able to decide when it should discard some information.

Network Number	NextHop
10	171.69.245.10

(a)

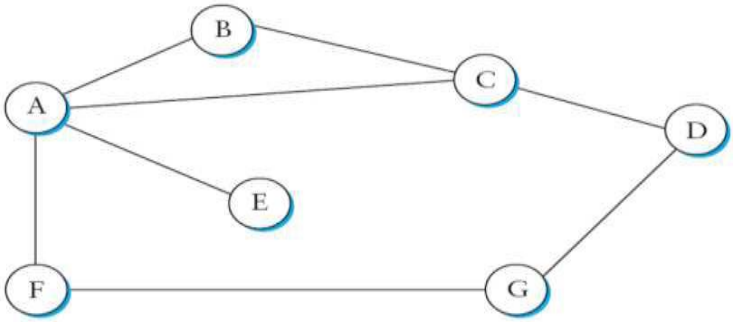
Network Number	Interface	MAC Address
10	if0	8:0:2b:e4:b:1:2

Fig 3.1 Example rows from (a) routing and (b) forwarding tables

Two main classes of routing protocols: ***distance vector*** **and** ***link state.***

3.1.1.DISTANCE VECTOR ROUTING:

Each node constructs a one-dimensional array (a vector) containing the distances (costs) to all other nodes and distributes that vector to its immediate neighbors. The starting assumption for distance-vector routing is that each node knows the cost of the link to each of its directly connected neighbors. A link that is down is assigned an infinite cost.



Destination	Cost	NextHop
B	1	B
C	1	C
D	∞	—
E	1	E
F	1	F
G	∞	—

Fig 3.2 initial routing table

Information Stored at Node	Distance to Reach Node						
	A	B	C	D	E	F	G
A	0	1	1	∞	1	1	∞
B	1	0	1	∞	∞	∞	∞
C	1	1	0	1	∞	∞	∞
D	∞	∞	1	0	∞	∞	1
E	1	∞	∞	∞	0	∞	∞
F	1	∞	∞	∞	∞	0	1
G	∞	∞	∞	1	∞	1	0

3.3 Initial distance stored at each node.

We may consider each row in Table 2 as a list of distances from one node to all other nodes, representing the current beliefs of that node. Initially, each node sets a cost of 1 to its directly connected neighbors and ∞ to all other nodes.

- Thus, A initially believes that it can reach B in one hop and that D is unreachable.
- The routing table stored at A reflects this set of beliefs and includes the name of the next hop that A would use to reach any reachable node. Initially, then, A's routing table would look like Table 1.
- The next step in distance-vector routing is that every node sends a message to its directly connected neighbors containing its personal list of distances.
- For example node F tells node A that it can reach node G at a cost of 1; A also knows it can reach F at a cost of 1, so it adds these costs to get the cost of reaching G by means of F.
- This total cost of 2 is less than the current cost of infinity, so A records that it can reach G at a cost of 2 by going through F. Similarly, A learns from C that D can be reache from C at a cost of 1; it adds this to the cost of reaching C (1) and decides that D can be reached via C at a cost of 2, which is better than the old cost of infinity.
- At the same time, A learns from C that B can be reached from C at a cost of 1, so it concludes that the cost of reaching B via C is 2.
- Since this is worse than the current cost of reaching B (1), this new information is ignored.

- At this point, A can update its routing table with costs and next hops for all nodes in the network. The result is shown in Table 3below

Destination	Cost	NextHop
B	1	B
C	1	C
D	2	C
E	1	E
F	1	F
G	2	F

Fig:3.4

- In the absence of any topology changes, it only takes a few exchanges of information between neighbors before each node has a complete routing table. The process of getting consistent routing information to all the nodes is called *convergence*.
- Table 4 shows the final set of costs from each node to all other nodes when routing has converged. We must stress that there is no one node in the network that has all the information in this table—each node only knows about the contents of its own routing table. The beauty of a distributed algorithm like this is that it enables all nodes to achieve a consistent view of the network in the absence of any centralized authority.

Information Stored at Node	Distance to Reach Node						
	A	B	C	D	E	F	G
A	0	1	1	2	1	1	2
B	1	0	1	2	2	2	3
C	1	1	0	1	2	2	2
D	2	2	1	0	3	2	1
E	1	2	2	3	0	2	3
F	1	2	2	2	2	0	1
G	2	3	2	1	3	1	0

Fig 3.5 : Final distances stored at each node (global view)(Table 4)

- First we note that there are two different circumstances under which a given node decides to send a routing update to its neighbors. One of these circumstances is the *periodic* update. In this case, each node

automatically sends an update message every so often, even if nothing has changed.

- This serves to let the other nodes know that this node is still running. It also makes sure that they keep getting information that they may need if their current routes become unviable. The frequency of these periodic updates varies from protocol to protocol, but it is typically on the order of several seconds to several minutes.
- The second mechanism, sometimes called a *triggered* update, happens whenever a node receives an update from one of its neighbors that causes it to change one of the routes in its routing table. That is, whenever a node's routing table changes, it sends an update to its neighbors, which may lead to a change in their tables, causing them to send an update to their neighbors.
- The second mechanism, sometimes called a *triggered* update, happens whenever a node receives an update from one of its neighbors that causes it to changeone of the routes in its routing table. That is, whenever a node's routing table changes,it sends an update to its neighbors, which may lead to a change in their tables, causing them to send an update to their neighbors.

Disadvantages

- The routing tables for the network do not stabilize. This situation is known as the *countto- infinity* problem. One technique to improve the time to stabilize routing is called *split horizon*. The idea is that when a node sends a routing update to its neighbors, it does not send those routes it learned from each neighbor back to that neighbor. For example, if B has the route (E, 2, A) in its table, then it knows it must have learned this route from A, and so whenever B sends a routing update to A, it does not include the route (E, 2) in that update. In a stronger variation of split horizon, called *split horizon with poison reverse*, B actually sends that route back to A, but it puts negative information in the route to ensure that A will not eventually use B to get to E. For example, B sends the route (E, ∞) to A.

3.1.2.ROUTING INFORMATION PROTOCOL (RIP):

This is one of the most widely used routing protocol in IP networks. Routing protocol in internetworking differ very slightly from the graph model. In graph model router advertise the cost of reaching other routers but here router advertise the cost of reaching other networks.

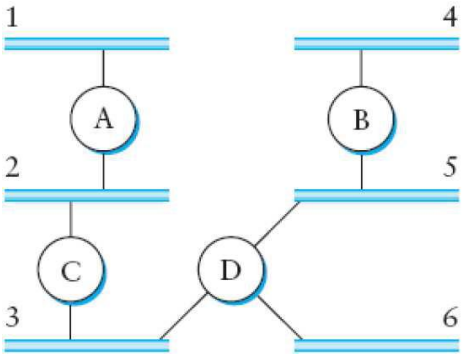


Fig 3.6 Example network running RIP

- For example in the above figure router C advertises router A that it can reach network 2 and 3 at a cost of 0, network 5 and 6 at a cost of 1 and network 4 at cost 2.
- The principle of routing algorithm is to find the shortest distance for reaching other networks and update the table.
- Example if router A learn from router B that X can be reached at a lower cost through B than the next hop in the routing table ,it updates the table with shortest path.
- RIP is the straight forward implementation of distance-vector routing. The packet format of RIP(version 2) is shown below.

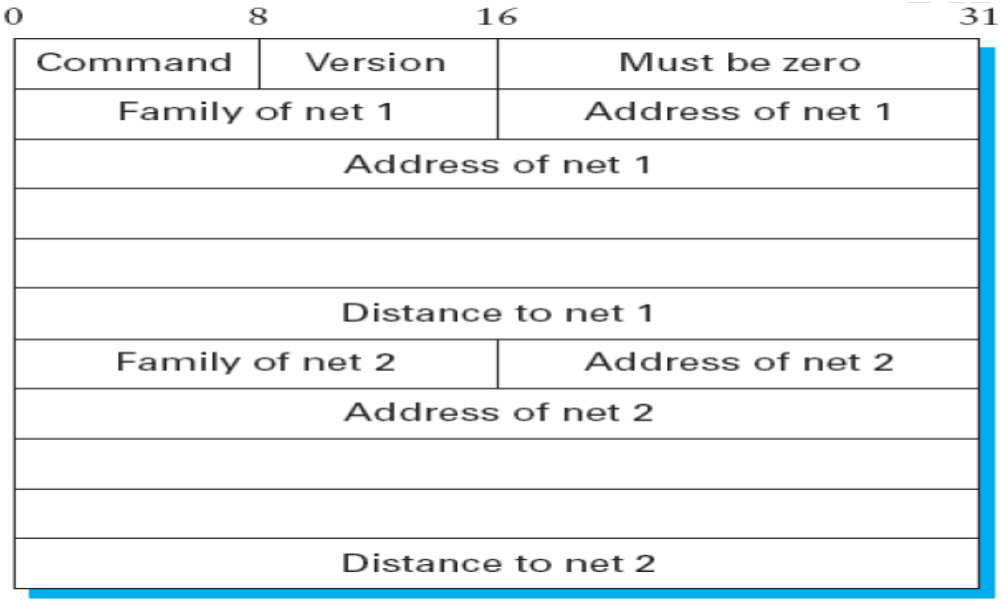


Fig:3.7

- Router running RIP send their advertisement every 30 seconds

- A router also send an update message whenever its routing table is changed
- It supports multiple address families
- Rip uses the simplest approach
- Valid distance are 1to 15 with 16 representing infinity

Limitation:

- RIP is suitable for only small networks with hop less than 16.

3.1.3.LINK STATE ROUTING(LSR):

The basic assumption of link state routing is same as distance vector routing. The basic idea of link state routing is,

1. Every node knows how to reach its immediately connected neighbor
2. This information is disseminated to every node to build a complete map of the network.

Reliable flooding:

It is the process of making sure whether all the nodes participating in the routing protocol get the copy of the link state information from all the other nodes.

The updated packets also called as link state packet contains the following information.

1. The ID of the node that created the LSP
2. A list of directly connected neighbors of that node with cost
3. A sequence number
4. A time to live for this packet

The first two are used for route calculation, the last two for flooding the packet to all nodes.

Flooding works in the following way:

- Consider a node X that receives a copy of LSP from node Y from the same routing domain. X checks if it already has a copy.

- If it already have it compares the sequence number of the arrived copy with the already stored copy and update the table with the copy having highest sequence number.
- If it already don't have a copy it will store it.
- Then X passes the updated LSP to all its neighbors

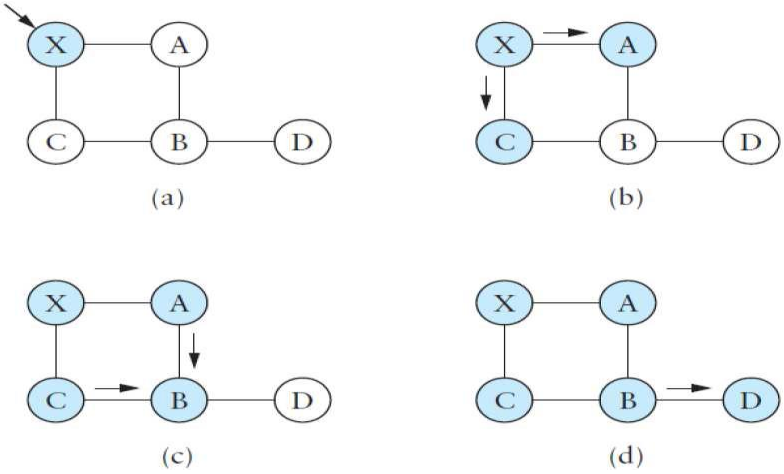


Fig 3.8

- The above figure shows an LSP flooding in a small network. It says LSP arrives at node X, which send it to neighbor A and C. A and C send it to B . B receive two identical copies if LSP and accept the one which arrives first .It then passes the LSP to D.
- As RIP each node generates LSP packets under two conditions,
    1. Expiry of a periodic timer
    2. Change in topology
- Change in topology occur if any of its neighbor is disconnected. This failure of link can be detected by sending "hello" message to neighbors. If a hello message is not received from a neighbor for a long time this link is considered to be failed and the network LSP packet is flooded to the network.

Route calculation:

- Once a node has a copy of LSP's from all other node, it create a complete map of the network and from the map best route to each destination can be found
- The best route can be found using **Dijkstra's shortest path algorithm as folow**
- Let N denotes the set of nodes in the graph.

- l(i,j) non-negative cost associated with the edge between I,j ϵ N
- l(i,j) = infinity if no edges connect I,j
- Let S ϵ N which is the node executing the algorithm
- M is set of nodes incorporated by the algorithm
- C(n) is the cost of path
- The algorithm is defined as,

  M= {S} --→ Start with M containing S nodes
  For each n in N-{S}
  C(n) = l (s,n)

While (N not equal to M)

M=M U{W} such that c(w) is the minimum for all W in (N-M)

For each n in (N-M) [add low cost node c(w) with M]

C(n) = MIN [c(n) , c(w)+ l(w,n)]

Practically a switch calculates its routing table using realization of Dijkstra's algorithm called forward search algorithm. Each switch maintain two list,

- Tentative
- Confirmed

The algorithm works as follows,

1. Initialize the confirmed list with an entry of myself cost 0
2. The nodes just added to confirmed list are called next nodes and select its LPS
3. For each neighbor of next calculate the cost . if the neighbor is currently on the tentative list and the cost is less than current cost then replace the current entry with (Neighbor, cost , Nexthop)
4. If the tentative list is empty ,stop otherwise go to step 2

Properties of link state routing: Advantages

1. It stabilizes quickly
2. It does not generate much traffic
3. Responds fastly for topology changes or node failures

Drawback:

- The amount of information stored at each node is large.

3.1.4.THE OPEN SHORT PATH FIRST PROTOCOL (OSPF)

- one of the most widely used link state routing protocol is OSPF
- OSPF add number of features to the basic link state algorithm. Some features are

Authentication of routing message:

The routing algorithm disperse information from one node to many other nodes, and the entire network may be impacted by bad information from one node. To avoid this and identify the correct nodes taking part in the protocol authentication messages are added. Early version of OSPF uses a simple 8-byte password for authentication

Additional hierarchy:

OSPF introduces another layer of hierarchy in routing by partitioning a domain in to areas. Therefore a router in a domain does not want to know how to reach each network in the domain ,just it need to have knowledge about the area ,thereby reducing the storage in each node.

Load balancing:

OSPF assigns same cost to multiple path of a destination there by making better use of available network capacity.

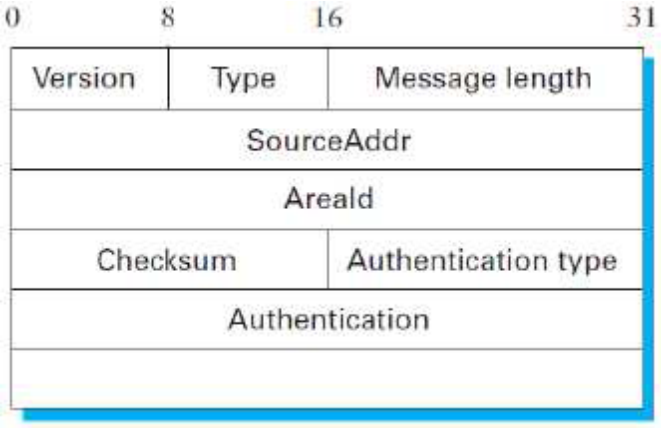


Fig.3.9 OSPF Header format

The OSPF header format is shown above,

Version: This field is set to 2

Type: This may take value 1 to 5

Source address: Identifies the sender of the message

Areald: It is a 32-bit identifier of the area in which the node is located

Checksum: 16-bit field for data protection

Authentication: 0-no authentication

1- Authenticated with simple password
2- Cryptographic authentication

There are five OSPF message types:

Type1: It is a hello message that a router send to its peer to say it is still alive .

The remaining messages are used to request , send and acknowledge the receipt of link-state message

The basic building block of OSPF link-state message is link-state advertisement (LSA) . one message may contain many advertisements. The packet format of a type 1 link state advertisement is shown below.

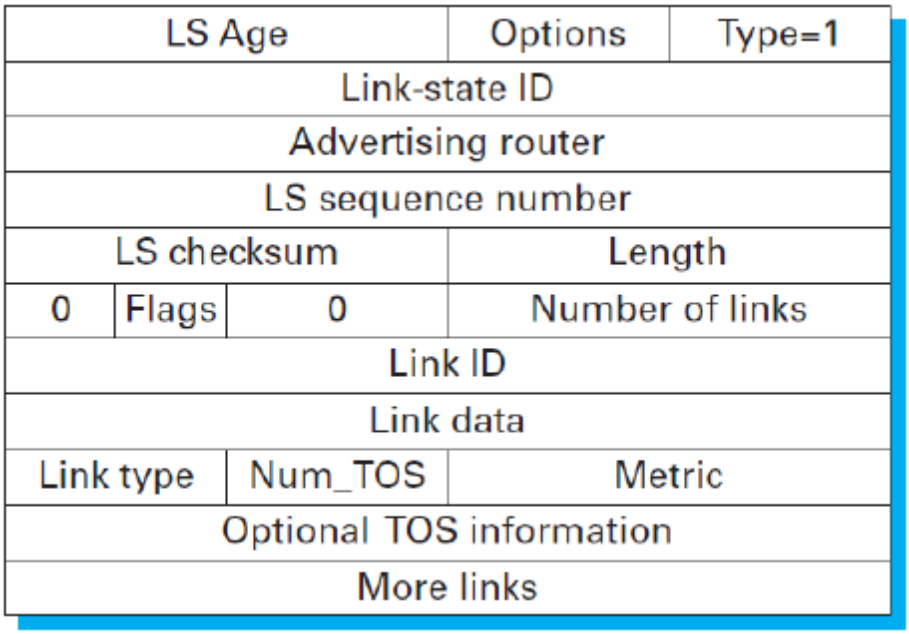


Fig:3.10 Type 1 link state advertisement

- Type 1 LSA's are used to advertise the cost of links between routers
- Type 2 LSA's are used to advertise networks to which the advertising router is connected
- Other types are used to support additional hierarchies

LS Age: This is same as Time to live but it counts up and the LSA expires when the age reaches a defined maximum value.

Type: This field says this is a type 1 LSA

Link state IP and advertising routers: Each carries a 32-bit identifier for the router that created this LSA

LS sequence number: to detect old or duplicate LSA's

LS checksum: It is used to check whether data is corrupted or not

Length: It is length in bytes of the complete LSA

Link ID: It represent each link in LSA

metric: It says about the cost of the link

TOS: Depending on TOS different metric value can be assigned for a link.

3.1.5.METRICS:

Let us see some ways to calculate link cost with high efficiency.

First way: one way to calculate link cost is to assign a cost of 1 to each link. The least cost route will be the one with the fewest hops.

Drawbacks:

1. It does not distinguish between links on a latency basis
2. It does not distinguish between routers on a capacity basis
3. It does not distinguish between links based on their current load

Various method for calculating the cost of link were tested with ARPANET

- First queue length method was used ie) A link with 10 packets waiting to be transmitted is assigned more cost than a link with 5 packets queued to be transmitted

Drawbacks:

- The packet will be moved to the shortest queue rather than towards the destination
- It does not take either bandwidth or latency of the link in to consideration.

The second technique uses delay as a measurement of load. The cost calculation is done as follow

1. Each incoming packet was time stamped with its time of arrival at the router (Arrival time) and time of departure from the router (Depart time).
2. The node compute delay from the packet as ,

   Delay=(Depart time- Arrival Time) + Transmission time + latency

(Depart time- Arrival Time) represent the amount of time the packet was delayed due to load . If the ACK did not arrive but the packet is timed out then the packet will be re-transmitted. If more re-transmission occur the link is less reliable and it should be avoided

Advantage: It consider the bandwidth and latency

Dis Adv: Under heavy load cost is high

Third approach: is "revised ARPANET routing metric" . It compresses the dynamic range of metric and smooth the variation of metric with time.

Smoothing was achieved by several mechanisms,

First the link utilization of current transmission is found and is averaged with the previous report utilization to suppress sudden changes

Second by knowing how the metric change over time can be smoothened by compressing, Compression of dynamic range can be achieved by feeding the measured utilization ,the link type and the link speed in to a function that is graphically shown below.

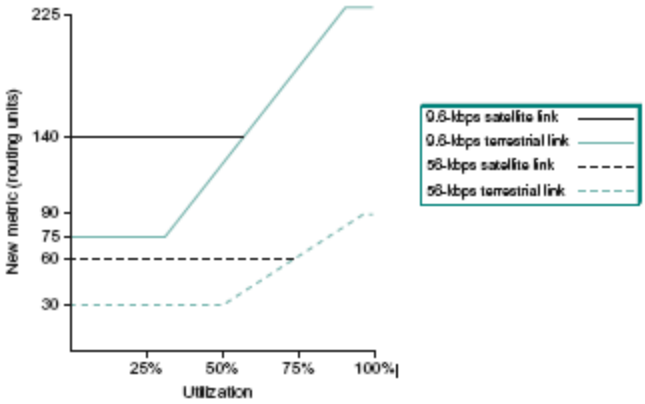


Fig 3.11.ARPANET routing metric

Conclusion:

Metric changes rarely and it changes only under the control of a network administrator. One common approach for setting metric is to use a constant multiplied by 1/link-bandwidth.

3.2.SWITCH BASICS:

Switch is a network node that forwards packets from input to output based on header information in each packet . It differs from router mainly in that it does not interconnect network of different type.

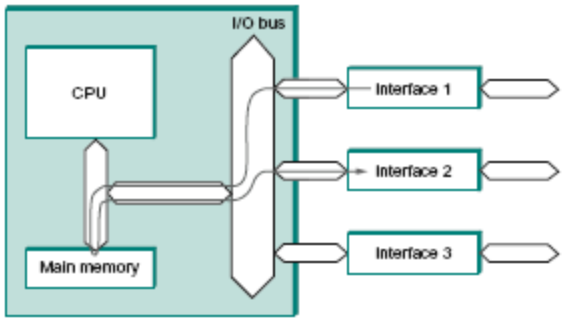


Fig 3.12. general purpose processor used as a packet switch

The above figure shows a processor with three network interfaces used as a switch . A packet arriving at interface 1 and delivered at interface 2 is shown by its path in figure. Here direct memory access technique is used ie) the data is moved directly to memory without moving to CPU . When the packet is in memory the CPU examines its header to determine to which interface the packet should be sent.

The main problem of using a general purpose processor as a switch is that its performance is limited since all packets must pass through a single point. In the example each packet crosses the input output bus twice and is written and read from main memory once. The highest throughput of such a device is either half the main memory bandwidth or half the I/O bus bandwidth.

Example a machine with 133MHz ,64-bit wide I/O bus can transmit data at a peak rate of 8Gbps. Since packets have to cross the bus twice its speed is limited to 4Gbps. This is not suitable for high end routers in the core of the internet .The formula for finding throughput is ,

Throughput = PPS*Bits per packets.

Throughput is observed rate at which data is sent through a channel ,PPS is packets per second .

Now large number of switch designs are developed to reduce contention and provide high throughput.

Some contention is unavoidable like if many input device has to send data to a single output .However if the destination is different having

different input then a well designed switch will move data from input to output parallel, thus increasing the throughput.

3.3.THE GLOBAL INTERNET:

Global internet is one that connects many different organizations . Lets take the below figure as an example of Global internet.

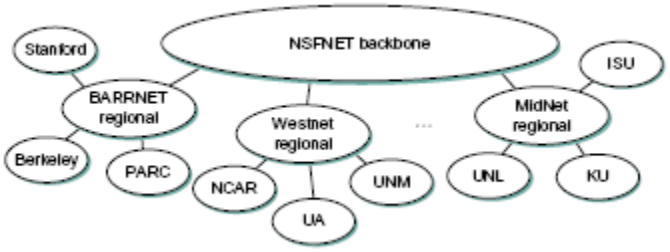


Fig 3.13. Basic structure of internet in 1990

In the above figure end user site (Stanford university) is connected to the service provider network (eg: BARRNET) . In 1990 many service providers served a limited geographical area and they are known as regional networks. These regional networks are connected by a backbone national science foundation ,NSF and was called NSFNET backbone.

Different service provider may use different protocols for routing therefore each providers are called as autonomous system(AS) . The main aim is to improve scalability

- In terms of routing and
- In terms of address utilization

The scalability can be improved by introducing hierarchy . Three such examples of introducing hierarchy are

- Routing areas
- Interdomain routing (BGP)
- IPV6

3.3.1.ROUTING AREA:

- It is one example of using hierarchy to scale up the routing system.

- A link state routing protocol can be used to partition a routing domain in to sub-domains called areas.
- An area is a set of routers that are configured to exchange information with one another.
- The following figure shows how routing domains are divided in to areas.

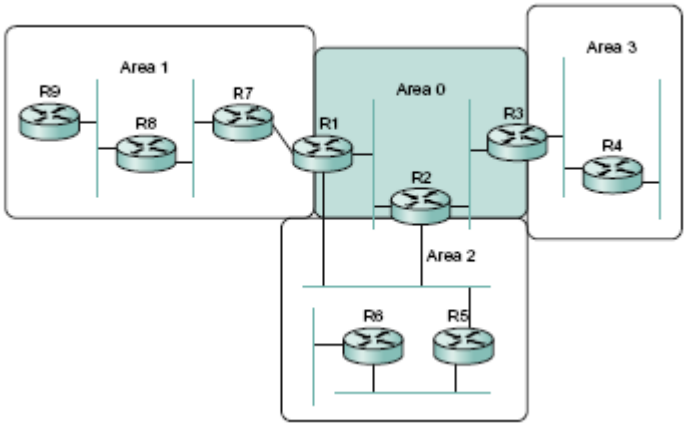


Fig 3.14

- Routers R1 , R2 and R3 are members of the backbone area ie) area 0.
- They are also member of non-backbone area. R1 is a member of area 1 and area 2.
- A router that is a member of a backbone area and a non-backbone area are called ‘area border router’ (ARB)
- Routers that are at the edge of autonomous systems are called AS border routers.
- All the routers in an area send link state advertisement to each other and thus form a routing and forwarding table. But routers in far domain do not get knowledge about the other . For example R4 in area 3 will never see an advertisement from R8 in area 1. The solution to this can be achieved as follows
  1. R1 receives advertisement from all routers in area 1 and send it to area 0
  2. From area 0 the information t reach routers in area 1 can be known by routers in non-backbone network of the domain

- In area 2 there are two ABRS R1 and R2 . Therefore the routers in area2 have to decide through which ARB they have to reach the backbone network for example if the destination is in area 1 R1 is the better choice.
- When dividing a domain in to areas administrators make a tradeoff between scalability and optimality of routing .Consider the example R4 and R5 were directly connected but still packets can not flow because they belong to different areas, they can flow only through the backbone area
- To overcome this administrators use the "Virtual link" between routers ie) for example a virtual link could be established between R8 to R1 . Now R8 become a part of backbone network and can communicate with other routers in area 0.

3.3.2.INTERDOMAIN ROUTING (BGP):

An internet service provider (ISP) is a network that connects number of autonomous systems .

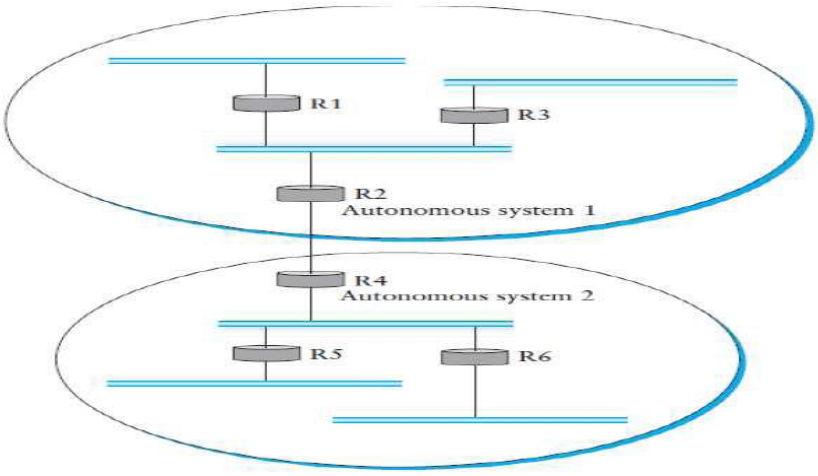


Fig .3.15

Each autonomous system in the internet acts as a domain and the use of AS is to provide an additional hierarchy thus improving scalability.

The routing problem is divided in to two parts,

1. Inter domain routing

2. Intra domain routing

The inter domain protocol in Internet is Broad gateway protocol (BGP)

Basics of BGP:

- Each AS has one or more border routers through which packets enter and leave the AS
- In the above figure routers R2 and R4 are border routers . The border router is also called the IP router and it forward packets between autonomous systems
- Each AS participating in BGP must have a BGP speaker ,BGP does not belong to distance vector routing or link-state routing
- It is a path vector protocol ie) BGP advertise complete path as an enumerated list of autonomous systems to reach a particular network.
- The BGP works as follows ,consider the figure shown below,

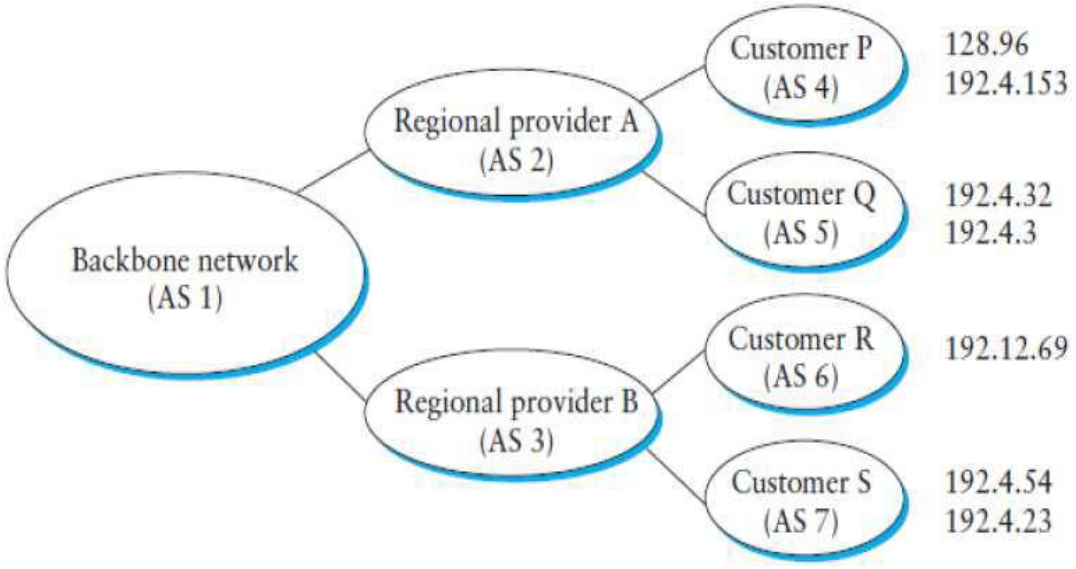


Fig 3.16. Example of a network running BGP

- Actually AS is classified in to three,
  1. **Stub AS** : It is an AS that has only single connection with another one AS .It is used for local traffic Example: small corporation.

2. **Multitone AS :** It has connection to more than one AS ,example is large corporation
3. **Transit AS:** It has connection to more than one AS and it carry both transit and local traffic.

- Let the providers AS2 and AS3 are transit networks AS4,AS5,AS6 and AS7 are stubs,AS2 can advertise that it can reach P and Q as the networks 128.96,192.4.153,192.4.32 and 192.4.3 can be reached directly from AS2.
- The backbone network receiving this advertisement can advertise "The networks 128.96,192.4.153,192.4.32 and 192.4.3 can be reached along the path <AS1 , AS2>".
- Similarly the backbone network advertise the networks "192.12.69 , 192.4.54 and 192.4.23 can be reached along the path <AS1 , AS 3>".
- An important job of BGP is to prevent looping paths ,consider the figure below,

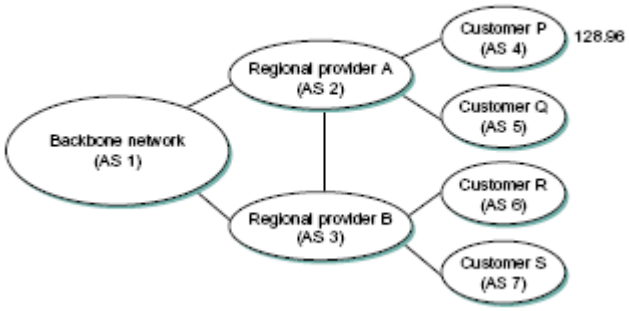


Fig 3.17.

- In the above figure AS2 and AS3 are connected forming a loop. For example if AS1 advertise that it can reach 128.96 through <AS1 ,AS2 > . This advertisement reaches AS3 and it will be transferred to AS2 .So AS2 thing the best way to reach 128.96 is <AS3 ,AS1,AS2 > . This is called looping because AS2 can directly connect to 128.96. this looping problem is avoided in BGP by assigning unique AS number. The unique AS number is a 16-digit number

- There may be several possible router to a destination , the BGP speaker choose the best path according to it's own policy then advertise .
- If the path fail or there is a change in policy it will cancel the previous advertisement by a negative advertisement called withdrawn route. The BGP packet format is shown below

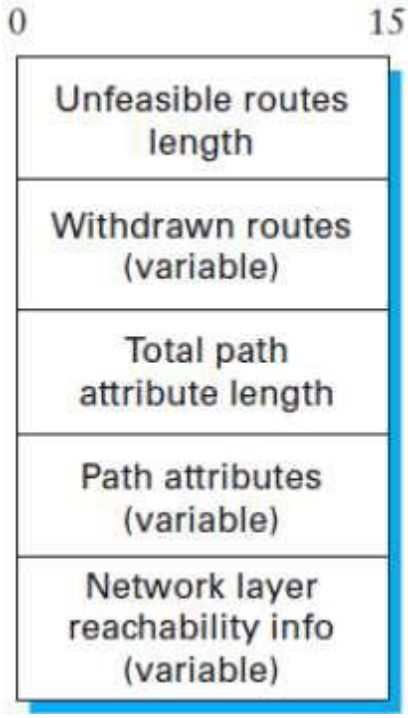


Fig 3.18. Packet format of BGP

- BGP run on top of TCP, A BGP speaker simply send a keep alive message saying 'I am still here and nothing has changed' still there is no change . If there is a change it won't send keep alive message.

Common AS relation ship and policies:

The three common relationship and policies are ,

1.Provider-customer: The providers connect their customers in the Internet. The customer may be a corporation or an ISP.Here the policy is to advertise all the routers " I know and learnt from my customer"

2. Customer-provider: He was to send traffic to internet through his provider and policy is to advertise my own prefixes and routers learned from my customer to my provider.

3. Peer: Here two providers exchange information with each other.

Integrating Interdomain and Intradomain Routing:

This concept is explained with an example and the figure is shown below.

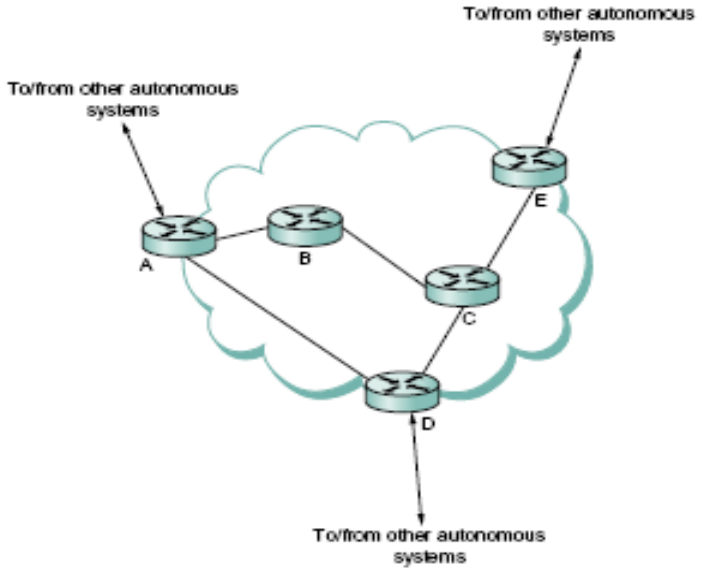
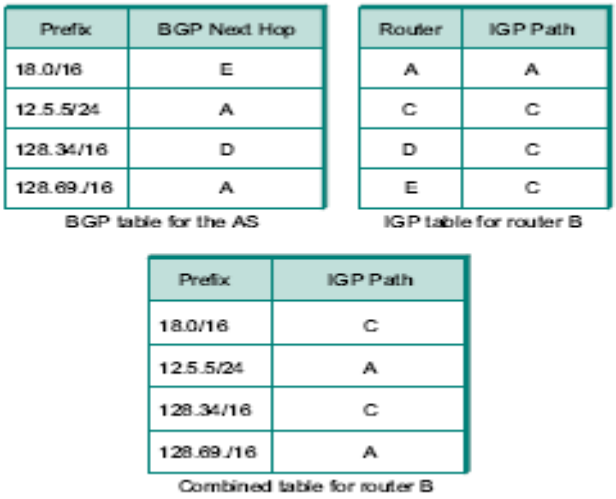


Fig 3.19

→A,D,E are three border routers that speak BGP to other autonomous systems and learn how to reach various prefixes.



Prefix	BGP Next Hop
18.0/16	E
12.5.5/24	A
128.34/16	D
128.69./16	A

BGP table for the AS

Router	IGP Path
A	A
C	C
D	C
E	C

IGP table for router B

Prefix	IGP Path
18.0/16	C
12.5.5/24	A
128.34/16	C
128.69./16	A

Combined table for router B

Fig 3.20

➔ These routers communicate with B and C as well as router in other AS by building a mesh of iBGP

➔ The top left table shows the information that router B learns from its iBGP session.

➔ From this table we can say some routers are best reached through A, some through D and E

➔ At the same time each router run's its own protocol like OSPF,RIP etc

➔ From these separate protocols B learns how to reach other nodes inside the domain as shown in top right table

➔ Finally in the bottom table B put all the information's together

➔ It's state prefix 18.0/16 reaches border router E through C.

3.3.3. INTERNETWORKING PROTOCOL VERSION 6 (IPV6).

The network layer protocol in the TCP\IP protocol suit is IPV4. IPV4 has some deficiencies that make it unsuitable for the fast growing internet . They are,

- The use of address space is inefficient
- For accommodating real time audio and video transmission ,minimum delay strategies and reservation of resources are needed which are not available in IPV4
- No encryption or authentication is provided by IPV4

To overcome these deficiencies IPV6 also known as Ipng(Internetworking Protocol Next Generation) was proposed. The advantages of IPV6 over IPV4 are,

- **Large address space:** IPV6 has 128nbit address where IPV4 has 32-bit address
- **Better Header Format:** In IPV6 header format options are separated from the base header and inserted when needed between the base header and the upper-layer data. This speeds up the routing process.
- **New options:** IPV6 have new options to allow for additional functionalities
- **Support for resource allocation:** Flow label mechanism is added in IPV6 for allowing source to request special handling of the packet
- **Allowance for extension:** It allows extension of protocol if required by new application.
- **Support for more security:** The encryption and authentication options in IPV4 provide confidentiality and integrity of the packets.

IPV6 address and routing:

- IPV6 provides a 128 bit address space, where IPV4 provide only 32-bit address space
- **Address notation:** The standard representation for writing IPV6 address is X:X:X:X:X:X:X:X where X is a hexadecimal representation of a 16-bit piece of the address.
- **Example:** 47CD:4422:1234:AC02:1234:A456:0124
- There are few special type if IPV6 addresses for example, an address with a large number of continuous 0's can be written more compactly by omitting all the 0 field's .Thus 47CD:0000:0000:0000:0000:0000:A456:0124 can be written as , 47CD:A456:0124
- Likewise 47CD:4422:1234:0002:1234:0000:0124 can be written as 47CD:4422:1234:2:1234:0124.

Categories of addressing:IPV6 defines three types of addresses,

i. Unicast address
ii. Anycast address

iii. Multicast address

Unicast address: A unicast address defines a single computer . The packet sent to a unicast address should be delivered to that computer.

Anycast address: An anycast address defines a group of computers whose address have the same prefix.

Multicast address: Multicast address defines a group of computers that may or may not share the same prefix, may or may not connected to the same physical network.

Address space assignment: The IP address is divided in to two parts**.** one is the type prefix which defines the purpose of the address ,this code is unique and the remaining part.

Type prefix	Type	fraction
0000 0000	Reserved	1/256
0000 001	NSAP	1/128

Provider based unicast address:

This address is generally used by a normal host as a unicast address. The address format is shown below.

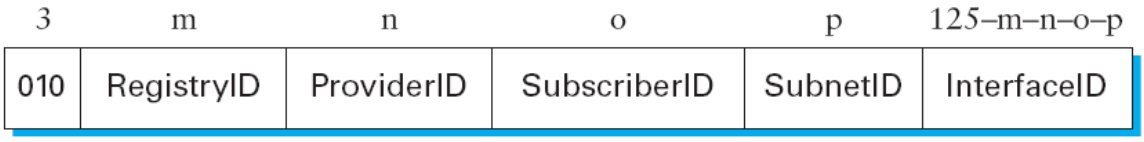


Fig 3.21

Type identifier: This three bit field defines the address as a provider-based address

Registry identifier: This five bit field indicates the agency that has registered the address. Currently three registry centers have been defined.

1. INTERNIC[Code 11000]
2. RIPNIC [code 01000]
3. APNIC [10100]

Provider identifier: This variable length field identifies the provider for internet access . A 16-bit length is recommended for this field .

Subscriber Identifier: When an organization subscribes through a provider , it is assigned a 24-bit subscriber identification.

Subnet Identifier: It defines a specific network under the territory of the subscriber. A 32-bit length is recommended for this field.

Node identifier: It defines the identity of the node connected to a subnet. A length of 48-bit is recommended for this field.

IPV6 packet:

The IPV6 packet format is shown below in figure 3.19.

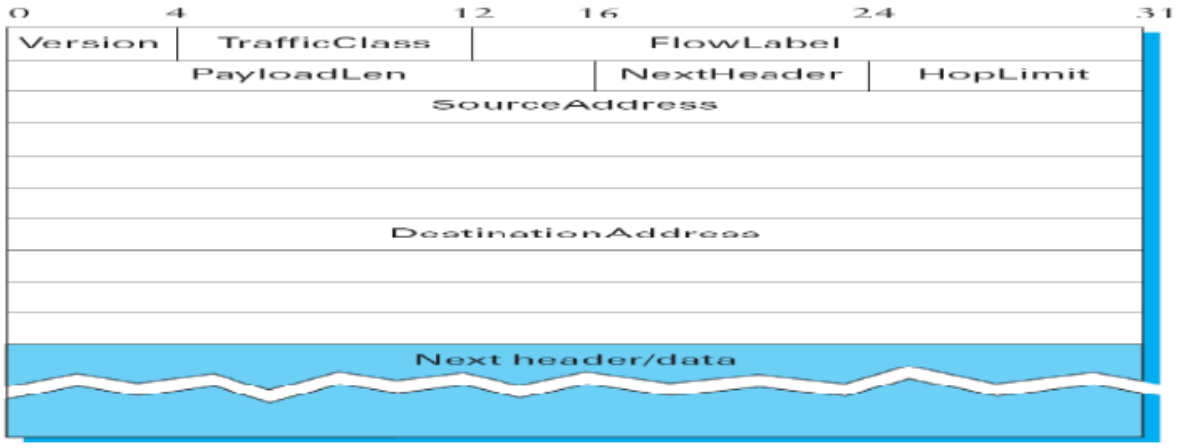


Fig 3.22

It starts with a

Version field: It is 6 for IPV6

Traffic Class and flow label: Related to quality of service issues

Payload Len: It gives the length of the packet excluding the IPV6 header measured in bytes

Next header field:

- It replaces both the IP options and the protocol field of IPV4

- If options are required then they are carried inone or more special headers following the IP header, this is indicated by the nextHeader field
- If there is no special headers then NextHeader field is the demux key identifying the high level protocol running over IP ie) TCP or UDP

Hop limit: This field is simply the TTL field .This field is 8-bit long

Source address: It is a16-bit field internet address that identifies the original source of the datagram

Destination address: It is a 16-byte internet address that identifies the final destination of the datagram.

3.4.MULTICAST:

It is a special form of broadcast in which packets are delivered to a specified subgroup of network hosts.

- The basic IP multicast mode is a many-to-many model based on multicast groups, where each group has its own IP multicast address
- The hosts that are members of a group receive copies of any packets send to that groups multicast address
- A host can be in multiple group and it can join and leave group freely by telling it's local route using protocol
- Multicast address are associated with an abstract group, the membership of which changes over time
- The original multicast service model allows any host t send multicast traffic to a group. It does not have to be a member of the group, there may be any number of such sender to a group.
- Using multicast identical packets can be sent to each member of a host , by sending a single copy.
- The sending host does not need to know individuals unicast IP address
- Compared to unicast IP ,IP multicast is more scalable because it eliminates the redundant traffic

- In unicast if three receivers need a packet ,three packets have to be send but it multicast it is enough to send one copy. The router will make copies whenever it need to forward

Source specific multicast:

- A mode of multicast in which a group may have only a single sender

3.4.1.Multicast addressing:

- In IP address space is reserved for multicast. In IPV4 multicast addresses are assigned in the class D address space . In IPV6 also a portion is reserved for multicast address.
- In IPv4 multicast address is 28-bit . It is difficult to interconnect with hardware on LAN , because Ethernet multicast address is 23-bits. Therefore to connect the IP multicast to Ethernet ,a 28-bit IP address have to be mapped to 23 bits out of 28 and ignoring the higher order 5 bits.
- If an Ethernet joins an IP multicast group, it configured its Ethernet interface to receive packets. It not only receives packets from the desired host but also from other member of the multicast group
- Therefore the receiving host must examine the IP header of the packet to determine whether it belong to it or not. This place an additional burden
- In some switching networks these unwanted packets are recognized by the switch and discarded.

3.4.2.Multicast Routing:

Multicast routing is a process by which multicast forwarding table are built. The two multicast routing protocols are,

i) Distance vector multicast routing protocol(DVMRP)
ii) PIM

3.4.1.1.DISTANCE VECTOR MULTICAST ROUTING PROTOCOL (DVMRP)

Distance vector routing for unicast is extended to support multicast, The resulting protocol is called distance vector multicast routing protocol (DVMRP).

- In distance vector algorithm ,each router maintains a table of <Destination. Cost, NextHop> and exchange a list of <Destination,cost> with its directly connected neighbor
- This algorithm is extended for multicast using two steps,
    a. We create a broadcast mechanism that forward a packet to all networks connected in the inter network
    b. It prune back networks that do not have host s that belong to the multicast group
- DVMRP is also called as flood and prune protocol .when a multicast packet is received from source S , the router forwards the packet on all outgoing lines except the link through which it receives the packet thus avoiding multicast.
- Two major limitations are,
    1.It flood the packet ,It can not avoid LAN's that are not the members of multicast
    2.The router will forward the packets to all links except the link through which it received .This will not check for shortest path
- The second limitation is eliminated by avoiding duplicate broadcast when more than one router is connected to a LAN. This is achieved by naming the shortest path to source S router as parent router. Only the parent router have to broadcast.
- A router can learn whether it is parent router based on the distance vector message it interchange with it's neighbour
- This technique of eliminating duplicate broadcast packets is called Reverse path broadcast(RPB) or reverse path forwarding (RPF).
- To avoid forwarding of packets to hosts that are not members of a multicast group ,
- First we need to determine whether it is a leaf node or not . A leaf network is one which has only parent nodes. If it is not a leaf network use the message that the host uses to identify whether they are members of a group

- The second stage is to propagate the message 'no member of G' through the shortest path . This is done by sending <Destination, cost> to its neighbor
- Then the information can be send from router to router.

3.4.1.2.PROTOCOL INDEPENDENT MULTICAST(PIM).

PIM operates in two modes,

1. Sparce mode
2. Dense mode

- PIM-DM uses a flood and prune algorithm like DVMRP and suffers from the same scaling problem
- PIM-SM is the dominant routing protocol and it can be used with any uni-cast routing
- It routers join the multicast distribution tree using PIM protocol message known as join message
- This join message is send to the router called rendezvous point (RP). This router allow receivers to learn about senders
- Two types of tree can be constructed from PIM-SM
    1. Shared Tree→ This can be used by all senders
    2. Source specific tree → This can be used only by specific sending host.
- When a router send a join message towards the RP for a group G it is send using normal unicast IP transmission
- This is shown in the following figure a

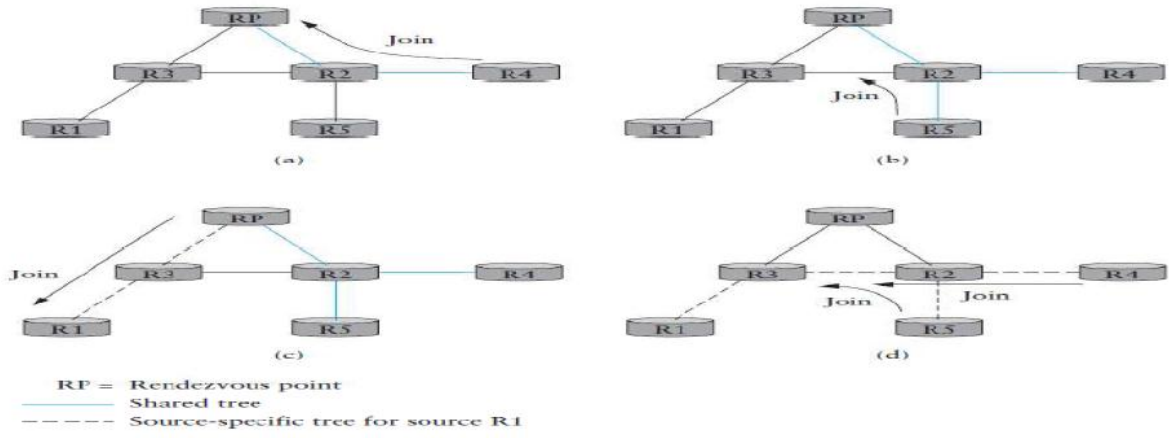


Fig 3.23

- In the above figure router R4 is sending a join message to RP from group G
- The join message have to pass through some sequence of routers to reach the RP in our case it passes through R2
- Each router on the path see the join message and create a forwarding table for the shared tree called (*,G)
- If more routers want to join RP ,the same process will be repeated forming new branches of a tree
- Suppose a host want to send a message to the group G it will create a packet with the multicast group address and send it to a router on its local network called the Designated router (DA).
- If DR is R1 ,still now R1 is not a shared tree of RP so it can not forward instead it tunnels it to the unicast IP address of IP.
- The RP after receiving the packet finds the packet addressed to the multicast group .So it send it to the shared tree ,now the packet reaches R2,R4 and R5.
- Even the packet reaches the destination it has the drawbacks of,
    1. Bandwidth in efficiency
    2. Cost wasted in encapsulation and de-capsulation
- Therefore instead of tunneling RP sends a join message to R1 through R3. Therefore R3 learns about the group and hence DR can multicast the packet instead of tunneling
- This join message send by RP is specific to that particular sender .Hence it is sender-specific. This is refer as (S,G).
- **Source specific tree:** When the path from sender to receiver through RP is longer than the shortest possible path , the sender will send a source-specific joint towards the receiver thus establishing a path without RP. This is called source specific tree, for example in fig d R4 establishes a path to R1 not involving RP.

UNIT-IV TRANSPORT LAYER

Overview of Transport layer – UDP - Reliable byte stream (TCP) - Connection management - Flow control - Retransmission – TCP Congestion control - Congestion avoidance (DECbit, RED) – QoS – Application requirements.

4.1.OVERVIEW OF TRANSPORT LAYER:

Transport layer support process-to-process communication ie)transport layer architecture supports communication between application program running in end nodes . This is also called as end-to-end protocol.

The transport layer provides the following ,

- Guarantees message delivery
- Delivers message in the same order they are send
- Delivers almost one copy of each message
- Supports arbitrarily large messages
- Supports synchronization between sender and the receiver
- Allows the receiver to apply flow control to the sender
- Supports multiple application process on each host

The transport layer does not include all the functions that application want. Example authentication or encryption .some limitations of transport layer are that it may,

- Drop messages
- Reorder messages
- Deliver duplicate copies of a given message
- Limit message to some finite size
- Deliver message after an arbitrarily long delay

To overcome these limitations some protocols were developed to provide

- De-multiplexing service
- A reliable byte stream sevice

- A request /replay service
- A service for real time applications

For demultiplexing and reliable byte stream two protocols are used,

- User datagram protocol (UDP)
- Transmission control protocol (TCP)
- Request /replay service is provided by RPC protocol
- Two widely used RPC protocols are,
    1. Sun RPC
    2. DCE-RPC

4.2.USER DATAGRAM PROTOCOL (UDP) or simple demultiplexer.

- User datagram protocol is an example of transport protocol that provide process-to- process communication between hosts.
- The address of the target process can be easily identified with an OS assigned process ID (Pid). This process is used only in closed distribution systems in which a single OS runs on all hosts and assigns each process an unique ID.
- UDP indirectly identifies the process using an abstract locator called a port.
- The basic idea is the source process send message to a port and the destination process receive message from the port.
- The header for an end-to-end protocol that implements the de-multiplexing function typically containing an identifier(port) for both the sender and the receiver of the message .
- The UDP header format is shown below.

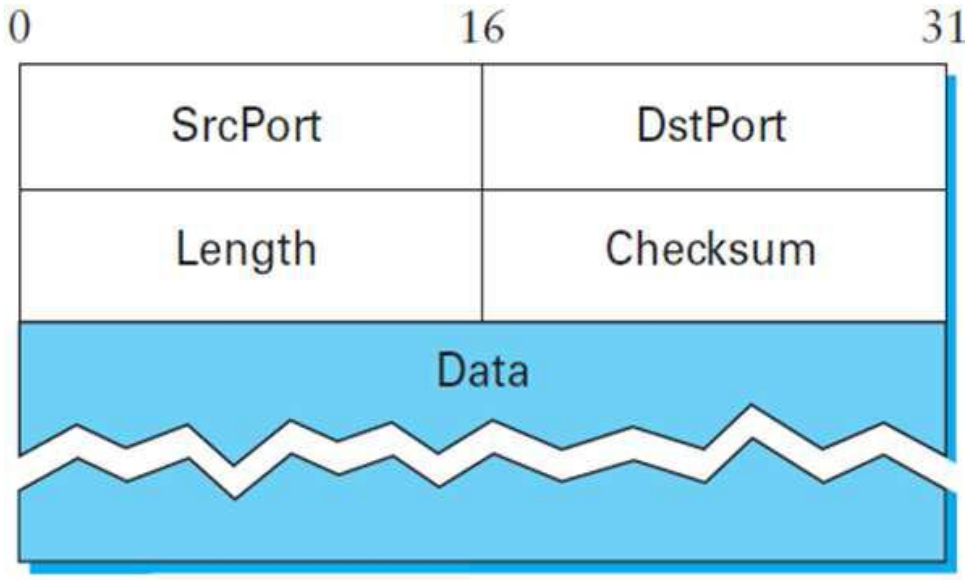


Fig 4.1 UDP Header format

- **UDP port field –** Source port is 16-bit long . Therefore there are up to 64K possible ports.
- The next issue is how a process learns the port for the process to which it want to send a message
- A client process indicates a message and exchange with a server process ,for this client need to identify the server's port. Mostly servers accept message at a well known port ,example the DNS receives message at well known port 53 on each host
- The mail service listens for message at port 517 and so on
- This mapping is published periodically in an RFC and is available on most unix systems in files
- An alternate method is to use port mapper service in which a client send a message to the port mapper .
- In the message the client mention the type of service , so by knowing that the port mapper assigs a port
- This method makes it easy to change the port associated with different services over time and for each host to use a different port for the same server.
- A port is implemented using a message queue as shown below

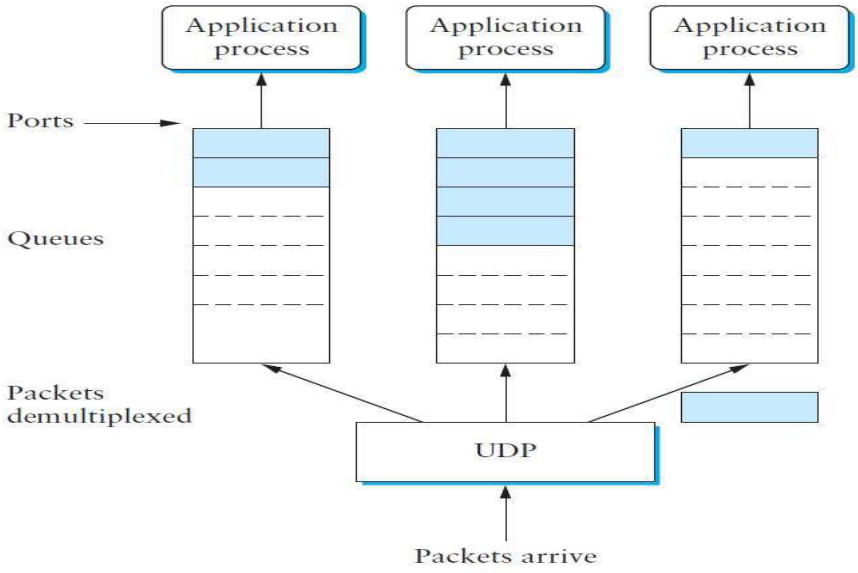


Fig 4.2 A message queue

- When a message arrives ,the UDP protocol send the message to the queue
- If the queue is full the message is discarded
- When a application process wants the message from queue , one is removed from the front of the queue.
- If the queue is empty the application process stops, still it receives a message in queue.
- It ensures the correctness of the message by using checksum ie) It adds up a 16-bit word using ones complement arithmetic and takes the ones complement of the result .
- The UDP heade takes the UDP header , the content of the message body ,the pseudoheader a s input
- The pseudoheader consist of the following fields ,protocol number, source IP address, Destination IP address and UDP length field
- The use of pseudoheader is to verify whether the packet is delivered to the correct destination
- For example if the destination IP address is changed during transmission of a packet , this can be detected by the UDP checksum
- Example , If we have three 16-bit words 0110011001100000 , 0101010101010101,1000111100001100.

The sum of fist two 16-bit words ,

0110011001100000 +
0101010101010101

1011101110110101

3rd word 1000111100001100

10100101011000001

0100101011000010

The ones complement is 1011010100111101, all the four 16-bit word are transmitted ,at the receiving end all the four words are added if we get the result as 1111111111111111, No error have occurred else there is error.

4.3.RELIABLE BYTE STREAM (TCP)

- The internets transmission control protocol is the most widely used protocol. It offers a reliable connection- oriented byte – stream service
- TCP is widely used for many applications because it prevent data from messing and reordering

Features:

- TCP guaranties reliable in-order delivery of packets
- It is full –duplex protocol ie) each TCP connection support a pair of byte streams
- It include flow control mechanism which limits the amount of information a sender can send
- Like UDP, TCP supports demultiplexing .
- It has a congestion control mechanism

4.3.1.TCP segment format:

TCP is a byte oriented protocol where the sender writes the byte in to a TCP connection and the receiver reads out the TCP connection. This is shown in below figure.

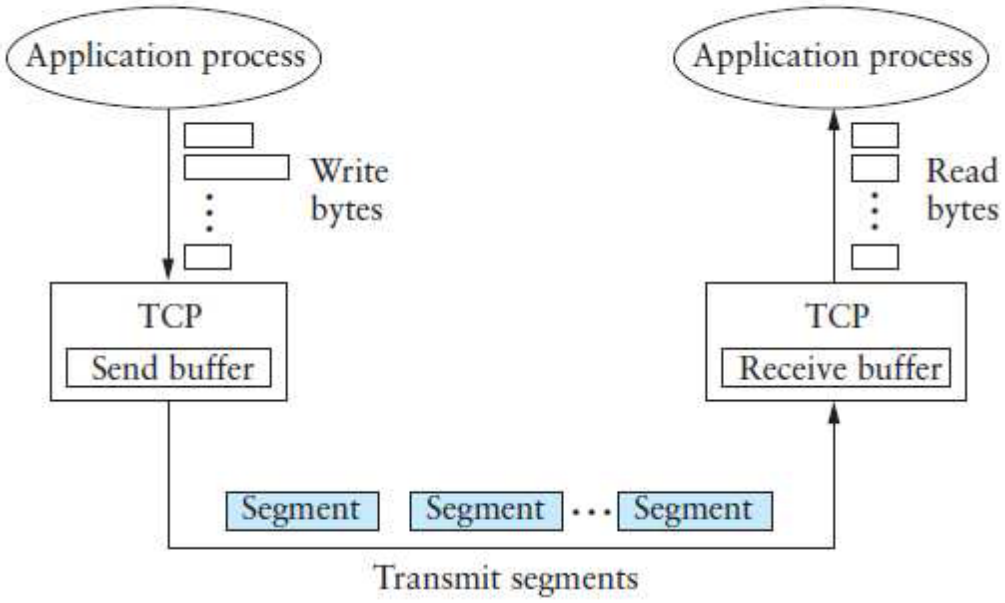


Fig4.3.TCP managing a byte stream

The packet exchange between TCP peers in the above figure are called segments . The structure of a TCP segment is shown below,

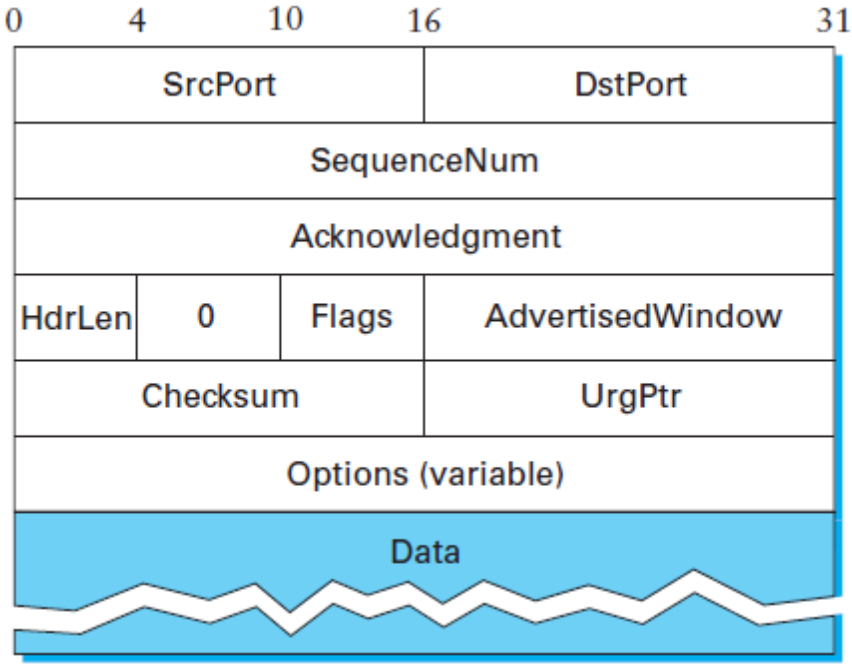


Fig 4.4. TCP Header format

Source and destination port: These fields are used for multiplexing /de-multiplexing data from/ to upper layer application ,They also include a checksum field.

Sequence number field & Acknowledgement number field: Both are 32-bit field. Both are used by the TCP sender and receiver in implementing a reliable data transfer service

Advertise window: It is a 16-bit field for flow control ,it indicates the number of bytes a receiver is willing to accept.

Header length field: It is a 4 bit field that specifies the length of TCP header in 32-bit word. TCP header is of variable length due to TCP options.

Flag field: It contain 6 bits.

1. ACK bit : Represent acknowledgement
2. RST bit : Abort connection
3. SYN bit : Establish connection
4. FIN bit : Terminates connection
5. PSH bit : Pass data to upper layer immediately
6. URG bit : Represent urgent data in the segment

Checksum: Used for error detection

Optional: It is used when a sender and receiver negotiate the maximum segment size(MSS).

4.3.2.Connection establishment and termination:

- The TCP connection begins when a client (caller) calls a server (callee).
- A party that want to initiate a connection performs an ctive open , while a party willing to accept a connection performs passive open

TCP connection establishment:

When a process running in one host (client) want to initiate a connection with another process in another host (server) , the client application process first informs the client TCP that it want to establish a connection to a process in the server . The TCP in the client then proceed to establish a TCP connection with the TCP in server as follow

Step 1:

- The client side TCP send a special TCP segment to the server side TCP.
- In the special segment the SYN bit is set to 1
- Hence this segment is called SYN segment

- Also the client randomly chooses a initial sequence number (client_isn) and put in the sequence number field of the segment.
- This segment is encapsulated within an IP datagram and sent to the server

Step 2:

Once the IP datagram reach the server host , the server extracts the TCP SYN segment from the datagram, allocates TCP buffer and variable to the connection and send a connection granted segment to the client TCP. This connection granted have three important piece of information,

1. The SYN bit is set to 1
2. The acknowledgement field is set to client-isn+1
3. The server chooses it's own initial sequence number (server_isn) and puts this value in the sequence number field of TCP segment header

This segment is called as SYNACK segment

Step 3:

After receiving the SYNACK segment the client also allocates buffer and variable to the connection . The client host send another segment to server host acknowledging the connection –grant segment. The SYN bit is set to 0 since connection is established.

After these three steps the client and server host can send segment containing data to each other , In all these data segments SYN=0 . Therefore to establish a connection three packets are sent between the two hosts and this is called **three way handshaking** and is shown in below figure.

Fig.4.5 (3.38 KUROSE)

Connection termination:

- If the client decides to close the connection ,the client application process issues a close command . The client TCP send a TCP segment to the server , in this segment the FIN bit is set to 1.
- When the server receives this segment it send the client an acknowledgement segment in return . The server then send its own shutdown segment which has the FIN bit set to 1
- Finally the client acknowledges the server's shutdown segment . Now all the resources in the two hosts are de-allocated
- Thus the connection is terminated.
- During the life of TCP connection , the sequence of TCP states visited by the client and server are shown in below figure

Fig 4.6. (KUROSE 3.40 3.41)

- Suppose a host receives a TCP SYN packet with destination port 80 , but the host is not accepting connection on port 80, then the host will send a special reset segment to the source, where RST=1. I don't have a socket fot the segment , do not resend the segment.

4.3.3.FLOW CONTROL:

The host on each side of the TCP connection have a receive buffer. When the TCP connection receives bytes that are correct and in sequence , the data are placed in the receive buffer. The associated application process will read data from the buffer when they need.

If the receiving application is busy with some other task and did not read the data in the buffer for a long time the sender can overflow the buffer by sending too much data too quickly.

TCP provides a flow control mechanism to prevent overflowing of receiver buffer. Flow control is thus a speed matching technique ie) matching the rate at which the sender is sending to the rate at which the receiver is receiving.

TCP provides flow control by making the sender maintain a receive window. This receive window is used to give the sender an idea of how much free buffer space is available at the receiver. Let us discuss the receive window with an example

Example: suppose Host A sending a large file to Host B over a TCP connection . Host B allocates a receive buffer to this connection denoting it's size by Rcv buffer. The application process in host B reads from the buffer.

Since TCP does not support overflow of buffer,

Last byte Rcvd –Last byte read ≤Rcv Buffer

Where ,

Last byte Read = The number of last byte read from the buffer

Last byte Rcvd = The number of last byte in the data stream that has arrived from the network

Rcv window-Rcv Buffer –[Last byte Rcvd-Last byte read]

Rcv window = Receive window which is dynamic

The receive window is shown below,

Fig.4.7. 3.37 (Kurose)

- Host B tells Host A how much space it has in the buffer by placing its current value of Rcv window in the receive window field of every segment it sends to A.

- Initially Host B sets Rcv window= Rcv Buffer
- Host A keeps track of two variables Lastbyte send and Last byte Acked
- Therefore the amount of un acknowledged data that A has sent is ,LastByte sent-Last byte Acked
- If Last byte sent- Last byte Acked ≤ Rcv window then there wont be over flow

The minor problem in this technique is if the Host B's receive buffer is full so that Rcvwindow=0 After advertising Rcv window =0 to Host A ,at the same time if Host B does not have any information to sent ,As time go receive buffer of B may have space but this information is not known to A

To solve this problem Host A have to continue to send segment with one data byte to Host B when Rcvwindow =0 .Host B will acknowledge when its buffer has space.

4.3.4.RETRANSMISSION:

- TCP guarantees reliable data delivery so it retransmits each segment if an ACK is not received in a certain period of time
- TCP sets this timeout as a function of the RTT it expects between two ends of the connection
- RTT is the round trip time ie) the time taken by a bit of information to propagate from one end of a link (or) channel to the other and back again
- This method of choosing timeout value is not easy. So adaptive transmission is used.

Original Algorithm:

- This is a simple algorithm for calculating the timeout between a pair of Hosts
- When every time TCP sends a data segment it records the time , when an acknowledgment is received for that it again reads the time.
- It takes the difference between the two time as sample RTT
- TCP then computes an estimated RTT as a weighted average between the previous Estimate and the new sample.
- Estimated RTT = α * Estimated RTT +(1-α) sample RTT

  Where α = smoothing factor for TCP and it's value lies between 0.8 and 0.9
- TCP calculate timeout as ,

$$\text{Timeout} = 2^* \text{ Estimated RTT}$$

Karna/patridge Algorithm:

The drawbacks of the original algorithm is, the ACK doesnot acknowledge a transmission ,it acknowledges the reception of data. That is when a segment is retransmitted and then an ACK arrives at the sender, It is not possible to determine whether the ACK is associated with the first (or) second transmission of data ,so estimating RTT is difficult.
Consider the case shown in below figure,

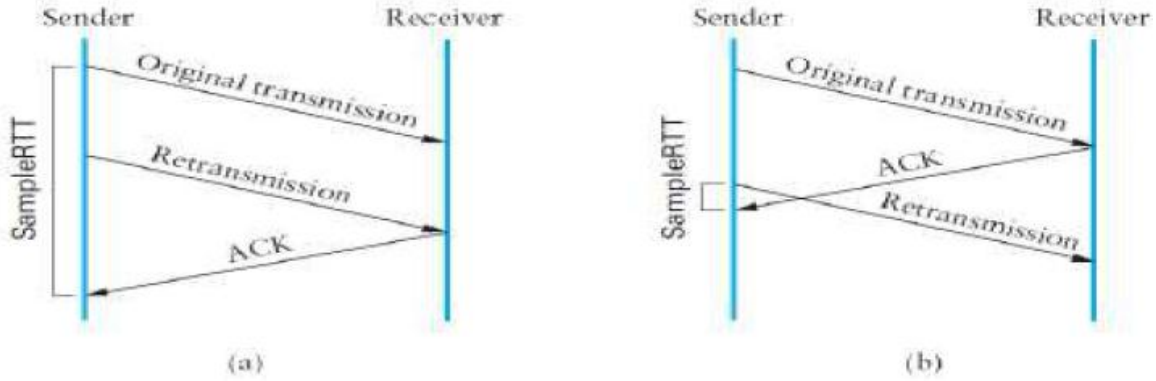


Fig.4.8. Associating the ACK with (a) original transmission versus (b) retransmission.

Casea: If we assure that the ACK is for the first transmission then sample RTT is too large.
Case b: If we assume that the ACK is for the second transmission then thw sample RTT is too small.

The solution to this is the karna/partridge algorithm proposed in 1981. In this algorithm RTT is measured for segments that have been sent only once, by this when the TCP retransmits it sets the next timeout to be twice the last timeout.

This technique too does not eliminate the problem completely.

Jacobson/Karels Algorithm:

A better approach to determine RTT as well as to avoid congestion was developed by Jacob and karels in 1998. In this approach as usual sample RTT is measured then the timeout is ca**lculated as follow**

Difference = Sample RTT- Estimated RTT
Estimated RTT = Estimated RTT + (δ*difference)
Deviation = Deviation + δ(difference- deviation)
Where, δ is a fraction between 0 and 1
Timeout =μ * Estimated RTT +φ * deviation
Where μ =1 and φ=4

When variance is small, Timeout is close to Estimated RTT.

4.3.5.TCP CONGESTION CONTROL:

- The main aim of congestion control is each source must determine how much capacity is available in the network and so how much packet can be safely transmitted
- Once a packet is transmitted the source waits for the acknowledgement. If ACK is received the source knows that the packet have reached the network safely and now it send the new packet thus avoiding congestion .Since TCP uses ACK for transmission of packets TCP is said to be self-clocking
- Determining the available capacity is not an easy case, so an algorithm is used to regulate the sending rate as a function of perceived congestion .The algorithm is called "TCP Congestion Control Algorithm"
- This algorithm has three major components,
    1.Additive Increase/Multiplicative Decrease
    2.Slow Start
    3.Fast Retransmit and Fast Recovery

Additive Increase/Multiplicative Decrease

The basic idea behind TCP congestion control is the sender has to reduce the sending rate by decreasing its congestion window size when a loss occur. The other TCP connections passing through the congestion router also should reduce the congestion window size. By how much the congestion window size can be reduced is known by a multiplicative decrease approach ie) halving the size of congestion window after a loss.

Similarly the TCP should increase it's congestion window size when there is no congestion. The congestion window size is increased little each time it receives the ACK. When there is no congestion for each round trip time the congestion window size can be increased by 1 MSS.

.

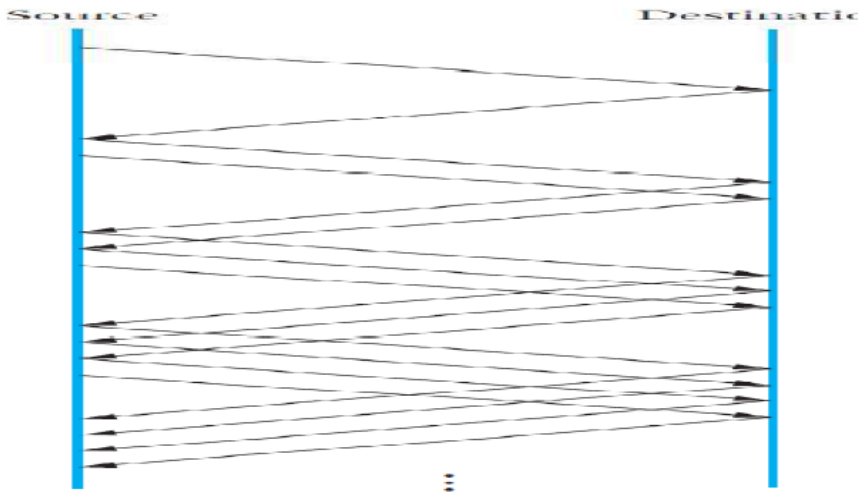


Fig 4.9. Packets in transit during additive increase, with one packet being added each RTT

Concluding a TCP sender additively increase it's rate when its ene-to-end path is congestion free and multiplicative decrease when the path is congested . So congestion control is often referred to as an additive ,multiplicative decrease(AIMD) algorithm.

- Specifically, the congestion window is incremented as follows each time an ACK arrives:

Increment = MSS × (MSS/CongestionWindow)

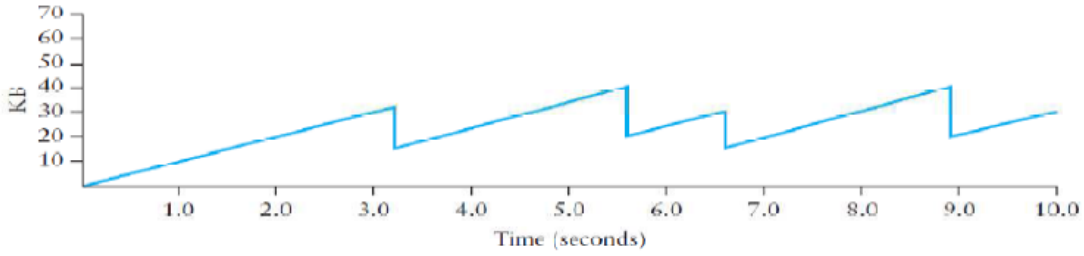


Fig 4.10. Typical TCP Sawtooth

Slow Start

- When a TCP connection begins , the value of congwin is initialized to 1MSS, resulting an initil sending rate of MSS/RTT.
- Example if MSS =500 bytes & RTT =200msec, sending rate is 500/200ms=2.5Kbps
- Since the available bandwidth is much larger than MSS/RTT , the TCP sender increases the rate exponentially by doubling its value of congwin every RTT.

- TCP sender continue to increase it's sending rate until there is a loss event
- If a loss is encountered at that time congwin is cut in to half and then increases linearly.
- Thus TCP sender egins by transmitting at a slow rate which is called slow start(SS) and then increases exponentially
- If the increase is linear instead of exponential after slow start it will take long time to know the available bandwidth. At the same time if exponential increase is used the chance of reducing window size to half is more.
- Example if a source is able to send 16 packets through the network successfully for the next transmission the window size is doubled so that 32 packets are transmitted. If there is only a capacity of 16 packets can be transmitted ,the remaining 16 will be dropped, whose outcome is worst
- To over come this drawback a mechanism called quick start is used , Here the TCP sender ask for an initial sending rate greater than slow rate .This request rate is put in the SYN packet as an IP option
- Routers along the path examines the option, evaluate the current level of congestion and decides whether the rate is acceptable . If all the routers on the path agree for this rate then a transmission better than slowrate can be achieved
- If any one of the router on the path does not agree for the request rate ,only slow start rate is adopted

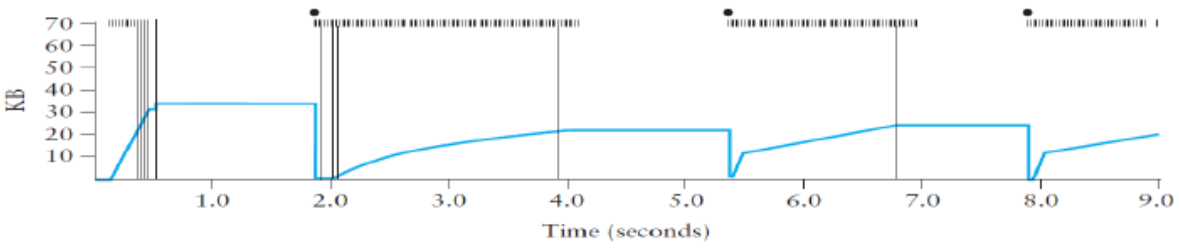


Fig 4.11. Behaviour of TCP congestion control

Fast Retransmit and Fast Recovery:

- The implementation of TCP time out leads to a long period of time during which the connection went dead. To overcome this fast retransmission mechanism was adopted.

Fast transmission:

- The receiver sends the acknowledgement every time when it receives a packet ,Example as in below figure

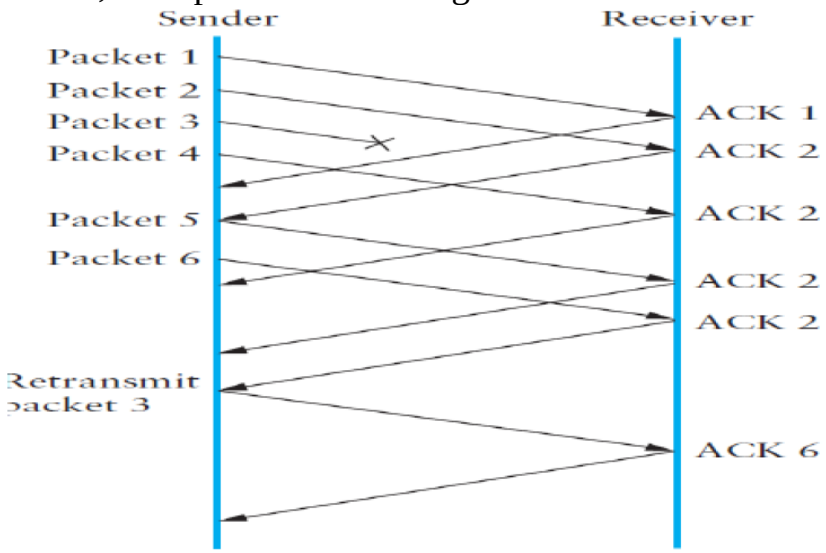


Fig 4.12. Fast retransmission based on duplicate acknowledgement

- when packet 1 and 2 are received ACK 1 and 2 are send
- Suppose if a packet arrives out of order ie) packet 4 is arrived before the arrival of packet 3 the sender will resend the previous acknowledgement ie) again ACK 2 is send
- By seeing the duplicate acknowledgment ACK 2 the sender knows packet 3 is lost
- The source may decide the packet is delayed instead of lost and waits for few duplicate acknowledgement before retransmission .
- By using fast retransmit mechanism the timeout problem cannot be completely eliminated
- Finally a mechanism called fast recovery was implemented , in which a slow start is used only at the beginning of the connection and whenever a timeout occur, at all the other times the congestion window is following a pure additive increase/ multiplicative decrease pattern.

4.4.CONGESTION AVOIDANCE MECHANISM:

- TCP control's congestion once it happens .So when it tries to find the point at which congestion occurs ,it loads the network. Thus losses are encountered.
- Congestion avoidance is different from congestion control.

- Congestion avoidance predict when congestion will occur and reduce the host's sending rate before packets are discarded
- Two congestion avoidance mechanism are
  i) DEC bit
  ii) RED

4.4.1 DECbit

It is a congestion control mechanism in which routers set a bit in the header of a routing packet when congestion is about to occur. By seeing the bit set the source reduces the sending rate .

The algorithm is described below,

1. A single congestion bit is added to the packet header.
2. A router sets this bit if its average queue length is greater than or
   equal to 1 at the time the packet arrives .
3. This average queue length is measured from the last byte + idle cycle ,plus the current busy cycle
4. Busy means when the router is transmitting ,idle means when it is not transmitting
5. The queue length as a function of time is shown in below figure
6. The router calculate the area under the curve and divide it by time interval to compute the average queue length.
7. After setting the congestion bit the packet is transmitted to the destination host .
8. The destination host copies this congestion bit into the ACK which is send to the source
9. The source monitor how many of its transmitted packets congestion bits are set
10. It has a congestion window , if less than 50% of the packets congestion bits are set. The congestion window size is increased by one packet.
11. If more than 50 % of the packets congestion bit are set the congestion window size is decreased by 0.875
12. The increase by 1 and decrease by 0.875 was selected because additive increase/ multiplicative decrease makes the mechanism stable.

.

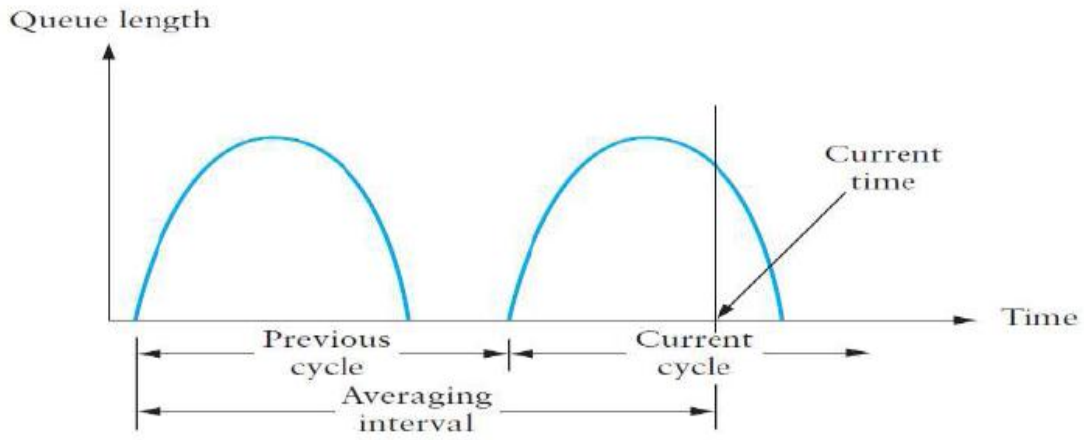


Fig 4 13. computing average queue length at a router

4.4.2.Random Early Detection (RED):

- RED is similar to DECbit , here too the route is programmed to monitor it's queue length notify the source to adjust the congestion window , when congestion is about to ooccur.
- RED differs from DECbit in two major ways.

First difference:

- RED does not explicitly send congestion notification instead it implicitly notifies the source by dropping a packet.
- Thus the source is notified by timeout or duplicate ACK.
- Therefore RED can be used in conjugation with TCP
- In the case the router drops few packets before the buffer is full so as to avoid dropping of number of packets after the buffer is full.
- Thus the router notify the source to adjust its congestion window.

Second Difference:

- The second difference between RED and DECbit is how RED decides when to drop a packet . This dropping of packets is done based on some drop probability whenever the queue length exceeds some drop level . This idea is called early random drop.
- The algorithm is as below,
  AvgLen =(1-weight) *AvgLen + Weight * SampleLen
  Where, 0< weight < 1
  sampleLen is the length of the queue

AvgLen is the average queue length .

- Average queue length is used rather than instantaneous queue length to improve accuracy . Because in internet traffic queue can become full very quickly and then become empty again .
- So only by using average calculation long-live congestion can be detected.
- This is shown in figure below,

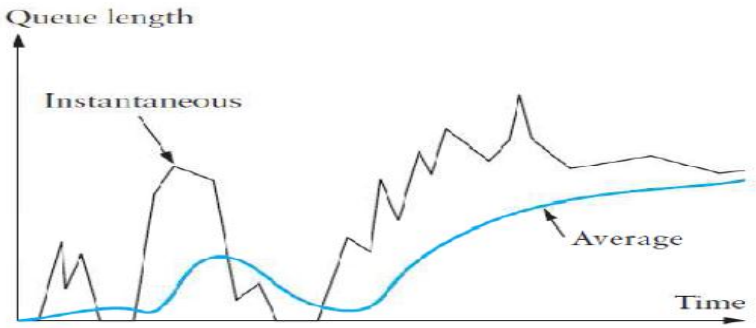


Fig 4 .14. weight's running average queue length

.

- RED has two queue length thresholds

1. Max Threshold
2. Min Threshold

- When a packet arrives at the gateway, RED compares the current AvgLen with these two thresholds, according to the following rules.

if AvgLen ≤ MinThreshold −→ queue the packet
if MinThreshold < AvgLen < MaxThreshold
−→ calculate probability P −→ drop the arriving packet with probability P
if MaxThreshold ≤ AvgLen −→ drop the arriving packet
Calculation of Drop Probability

TempP = MaxP × (AvgLen − MinThreshold)/(MaxThreshold − MinThreshold)

P = TempP/(1 − count × TempP)

count keeps track of how many newly arriving packets have been queued.

AvgLen has been between the two thresholds

- If avglength is smaller than the lower threshold no action is taken and if the average queue length is larger than the upper threshold ,then the packet is always dropped.
- If the average queue length is between the two threshold then the newly arriving packet is dropped with some probability P. This situation is depicted in figure below

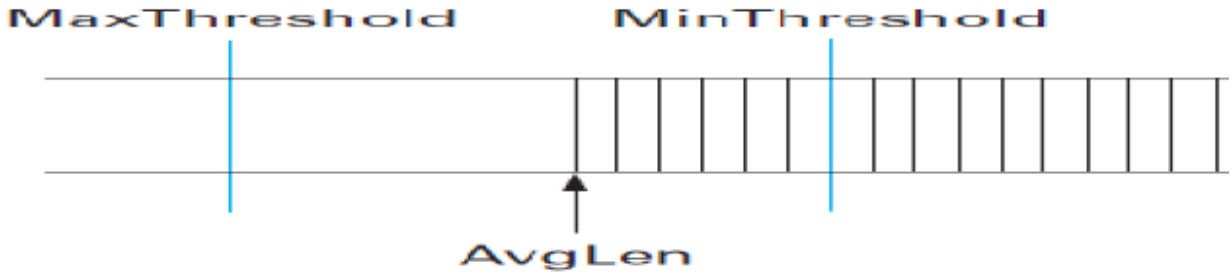


Fig4.15. RED thresholds on a FIFO queue

- The approximate relation ship between P and Avglen is shown below

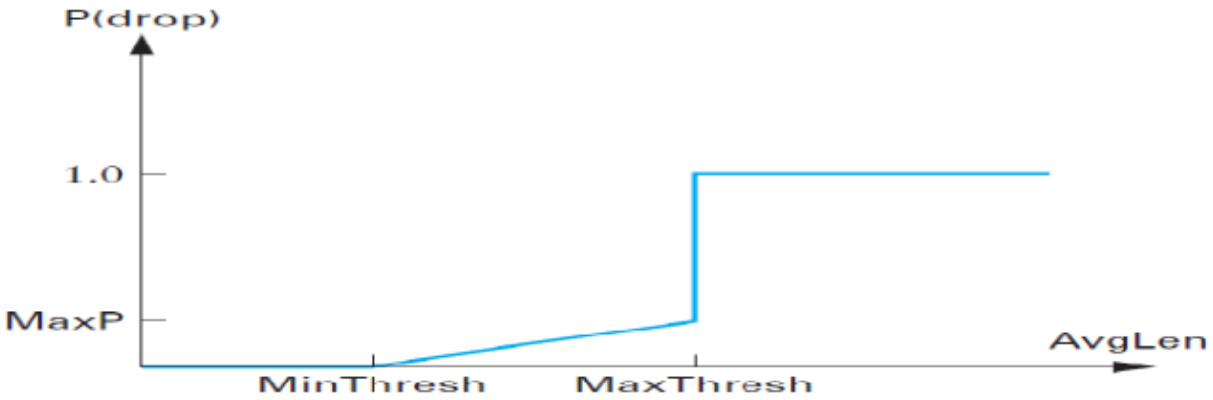


Fig.4.16. Drop probability function for RED

4.5.QUALITY OF SERVICE

- Multimedia applications that combine audio, video and data are converted to stream of data for digital transmission.
- When they are converted to stream of data large bandwidth is needed
- This bandwidth requirement can be reduced using some coding techniques.
- Other requirement include timelines of delivery.
- Applications that are sensitive to timeline of data are real time applications

- An important characteristic of real time application is that they need assurance from the network
- A network that provides these different levels of services is often said to support quality of service.
- **Quality of service:** Packet delivery guarantees provided by a network architecture . usually related to performance guarantees , such as bandwidth and delay. The internet offers a best-effort delivery service meaning that every effort is made to deliver a packet but delivery is not guaranteed.

4.5.1.Application Requirements

- The applications are of two types,
  1. Real-time applications
  2. Non real time applications(Traditional applications)

Real time Audio Example :

An audio application is shown in below figure,

Fig.4.17.Real-Time Audio Example

- The signal generated by the micro phone is sampled and digitized using analog to digital converter.
- These digital samples are placed in packets and send to the network
- The network forward it to the receiver
- At the receiving host the data must be played back at the same rate as sampling rate
- That is if the samples are sampled at a rate of125us the receiving host should receive samples every 125us
- If any of the sample did not arrive in time the data is of no use
- But we can not assure that every sample can arrive in time since packets encounter queues in switching or routing, The length of the queue vary with time.
- To overcome this we can use a buffer at the receiving side to reserve the data, hereby providing a storage of packets waiting to be played back at the right time.

- By doing so if the packet is delayed a short time it goes into the buffer until it's playback time arrives
- If the packet is delayed a long time there will be loss of data.
- This is shown in below figure

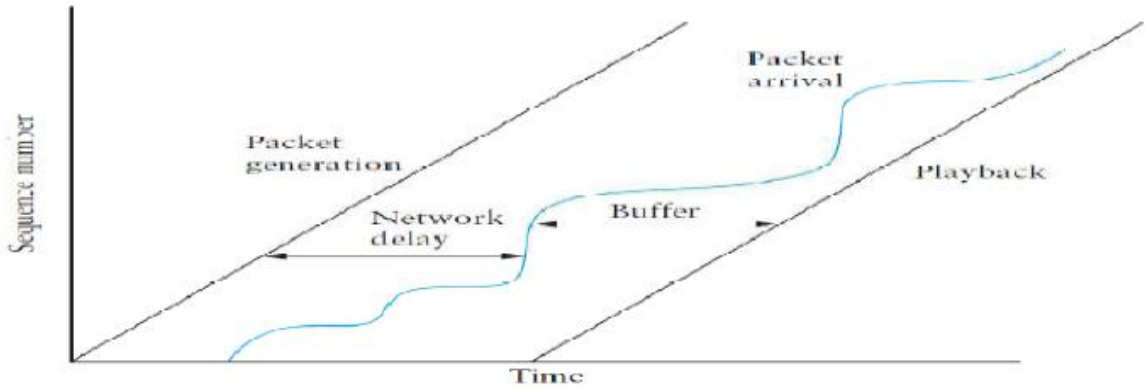


Fig 4.18.A playback buffer.

- If the packet arrives later it is of no use and is discarded ,a better approach is shown below in the figure

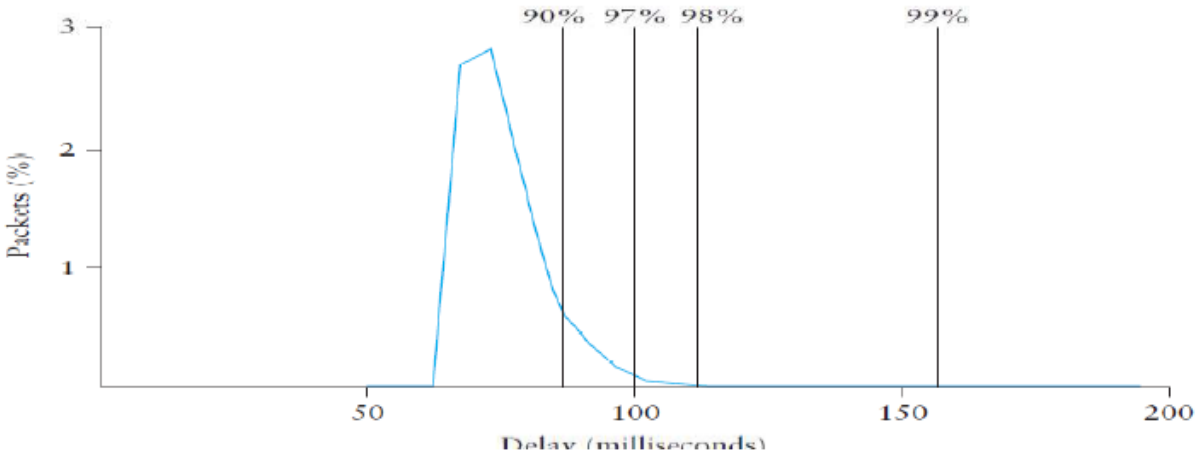


Fig 4.19. Example distribution of delays for an Internet connection.

- Here delay is measured for a certain path across the internet
- From the figure we can say if we select a latency of 100ms 97% of the packet arrive properly ie) only 3 out of 100 packtets is lost.

Taxonomy of Real-Time Applications:

The taxonomy of application sis summerised in the below figure,

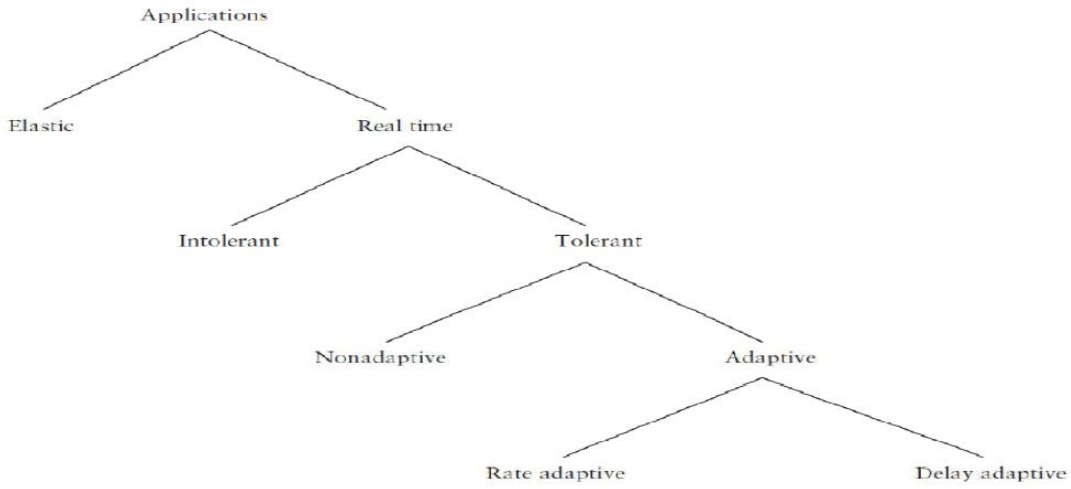


Fig 4.20 Taxonomy of applications

The applications are categorized based on their characteristic.

1. **Tolerance of loss of data:**

   In an audio example if some samples are lost , the lost audio sample can be interpolated from the surrounding sample thereby the audio can be understood .If more and more samples are lost quality of the audio declaim. So here the loss us tolerable

   Consider the other case if a sample is lost in a robot control program and is an instruction to control robot arm here the loss is in-tolerant.

2. **Adaptability:**

   An audio application is able to adopt to the amount of delay that packet experience as they travel the network. Suppose the playback point is set such that the packets arriving within 300ms are buffered and played.

   If we notice that the packets are arriving within 100ms , the playback point can be changed accordingly to improve perception. Adaptability is of two types,

   1. Delay Adaptive
   2. Rate Adaptive

Delay adaptive Applications:

Application's that can adjust their playback point are delay-adaptive applications.

Rate adaptive Applications:

Ex: Many video coding algorithms can trade off bit rate versus quality, Thus if we find that the network can support a certain bandwidth we can set out coding parameters accordingly.

Approaches to QoS Support :

There are approaches that provide a range of quality of service . They are divided into two,

1. Fine grained approach
2. Coarse grained approach

Fine grained approach

This provide Qos to individual applications or flows .Ex Integrated service associated with RSVP.

Coarse grained approach

It provide Qos to large classes of data or aggregated traffic . Ex .Differentiated services.

UNIT-V APPLICATION LAYER

Traditional applications -Electronic Mail (SMTP, POP3, IMAP, MIME) – HTTP – Web Services – DNS – SNMP

5.1 TRADITIONAL APPLICATIONS:

- We call world wide web and electronic mail as traditional applications. Both of these applications uses the request /replay paradigm.

5.1.1 ELECTRONIC MAIL:

- Email is one of the oldest network applications.

Message Format

- RFC 822 defines messages to have two parts: a header and a body. Both parts are represented in ASCII text.
- The message header is a series of <CRLF>-terminated lines. (<CRLF> stands for carriage-return + line-feed, which are a pair of ASCII control characters often used to indicate the end of a line of text.)
- The header is separated from the message body by a blank line.
- Each header line contains a type and value separated by a colon.
- Many of these header lines are familiar to users since they are asked to fill them out when they compose an email message.
- For example, the To: header identifies the message recipient, and the Subject: header says something about the purpose of the message. Other headers are filled in by the underlying mail delivery system.
- Examples include Date: (when the message was transmitted), From: (what user sent the message), and Received: (each mail server that handled this message).
- RFC 822 was extended in 1993 (and updated again in 1996) to allow email messages to carry many different types of data: audio, video, images ,Word documents, and so on.
- MIME consists of three basic pieces.
- The first piece is a collection of header lines that augment the original set defined by RFC 822.
- These header lines describe, in various ways, the data being carried in the message body.
- They include

- MIME-Version: (the version of MIME being used),
- Content-Description: (a human-readable description of what's in the message, analogous to the Subject: line),
- Content-Type: (the type of data contained in the message), and Content-Transfer-Encoding: (how the data in the message body is encoded).
- The second piece is definitions for a set of content types (and subtypes).
- For example, MIME defines two different still image types, denoted image/gif and image/jpeg, each with the obvious meaning.
- As another example, text/plain refers to simple text you might find in a vanilla 822-style message, while text/richtext denotes a message that contain marked up text (e.g., text using special fonts, italics, etc.).
- As a third example, MIME defines an application type, where the subtypes correspond to the output of different application programs (e.g. application/postscript and application/msword).
- MIME also defines a multipart type that says how a message carrying more than one data type is structured. This is like a programming language that defines both base types (e.g., integers and floats) and compound types (e.g., structures and arrays).

```
MIME-Version: 1.0
Content-Type: multipart/mixed; boundary="--------417CA6E2DE4ABCAFBC5"
From: Alice Smith <Alice@cisco.com>
To: Bob@cs.Princeton.edu
Subject: promised material
Date: Mon, 07 Sep 1998 19:45:19 -0400

----------417CA6E2DE4ABCAFBC5
Content-Type: text/plain; charset=us-ascii
Content-Transfer-Encoding: 7bit

Bob,

Here's the jpeg image and draft report I promised.

--Alice

----------417CA6E2DE4ABCAFBC5
Content-Type: image/jpeg
Content-Transfer-Encoding: base64

. . . unreadable encoding of a jpeg figure

----------417CA6E2DE4ABCAFBC5
Content-Type: application/postscript; name="draft.ps"
Content-Transfer-Encoding: 7bit

. . . readable encoding of a PostScript document
```

MIME

- Electronic mail has a simple structure.
- It can send messages only in NVT 7-bit ASCII format.
- For example, it cannot be used for languages that are not supported by 7-bit ASCII characters (such as French, German, Hebrew, Russian, Chinese, and Japanese).
- Also, it cannot be used to send binary files or video or audio data.

- Multipurpose Internet Mail Extensions (MIME) is a supplementary protocol that allows non- ASCII data to be sent through e-mail.
- MIME transforms non-ASCII data at the sender site to NVT ASCII data and delivers them to the client to be sent through the Internet. The message at the receiving side is transformed back to the original data
- We can think of MIME as a set of software functions that transforms non-ASCII data (stream of bits) to ASCII data and vice versa.

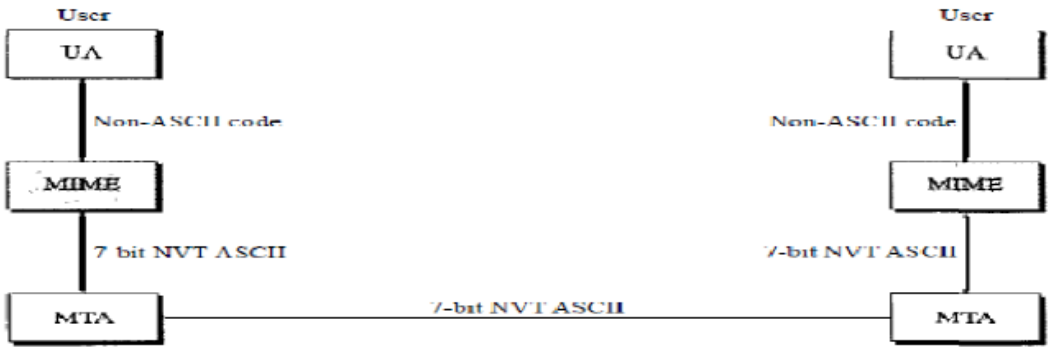


Fig: 5.1.MIME

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

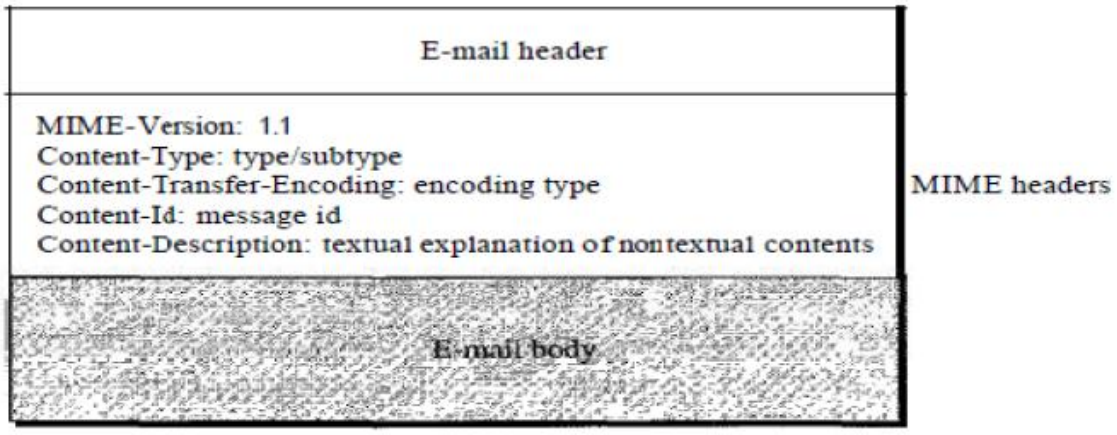


Fig: 5.2.MIME header

- MIME-Version This header defines the version of MIME used. The current version is 1.1.

- Content-Type This header defines the type of data used in the body of the message.
- The content type and the content subtype are separated by a slash. Depending on the subtype, the header may contain other parameters.

Data types and subtypes in MIME

- MIME allows seven different types of data

Type	*Subtype*	*Description*
Text	Plain	Unformatted
	HTML	HTML format (see Chapter 27)
Multipart	Mixed	Body contains ordered parts of different data types
	Parallel	Same as above, but no order
	Digest	Similar to mixed subtypes, but the default is message/ RFC822
	Alternative	Parts are different versions of the same message
Message	RFC822	Body is an encapsulated message
	Partial	Body is a fragment of a bigger message
	External-Body	Body is a reference to another message
Image	IPEG	Image is in IPEG format
	GIF	Image is in GIF format
Video	MPEG	Video is in MPEG format
Audio	Basic	Single-channel encoding of voice at 8 kHz
Application	PostScript	Adobe PostScript
	Octet-stream	General binary data (8-bit bytes)

- Content-Transfer-Encoding This header defines the method used to encode the messages into Os and Is for transport. content
- Content transfer encoding:

Content-Transfer-Encoding: <type>

Type	*Description*
7-bit	NVT ASCII characters and short lines
8-bit	Non-ASCII characters and short lines
Binary	Non-ASCII characters with unlimited-length lines
Base-64	6-bit blocks of data encoded into 8-bit ASCII characters
Quoted-printable	Non-ASCII characters encoded as an equals sign followed by an ASCII code

- Content-Id This header uniquely identifies the whole message in a multiple-message environment.

Content-Id: id=<content-id>

- Content-Description This header defines whether the body is image, audio, or video.

Content-Description: <description>

SMTP

Message Transfer

- SMTP—the protocol used to transfer messages from one host to another.
- First, users interact with a mail reader when they compose, file, search, and read their email.
- Most Web browsers now include a mail reader.
- Second, there is a mail daemon (or process) running on each host. this process as playing the role of a post office:
- Mail readers give the daemon messages they want to send to other users, the daemon uses SMTP running over TCP to transmit the message to a daemon running on another machine, and the daemon puts incoming messages into the user's mailbox (where that user's mail reader can later find it).
- While it is certainly possible that the sendmail program on a sender's machine establishes an SMTP/TCP connection to the sendmail program on the recipient's machine, in many cases the mail traverses one or more mail gateways on its route from the sender's host to the receiver's host.
- Like the end hosts, these gateways also run a sendmail process. It's not an accident that these intermediate nodes are called ―gateways since their job is to store and forward email messages, much like an ―IP gateway (which we have referred to as a router) stores and forwards IP datagrams.

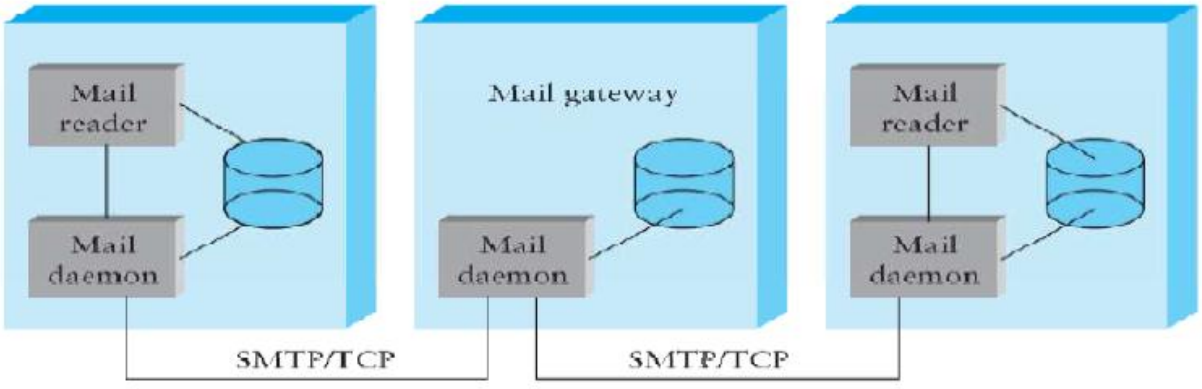


Fig:5.3. Sequence of mail gateways store and forward email messages

- The forwarding gateway maintains a database that maps users into the machine on which they currently want to receive their mail; the sender need not be aware of this specific name.

- Another reason is that the recipient's machine may not always be up, in which case the mail gateway holds the message until it can be delivered.
- Each SMTP session involves a dialog between the two mail daemons, with one acting as the client and the other acting as the server. Multiple messages might be transferred between the two hosts during a single session.
- SMTP is best understood by a simple example. The following is an exchange between sending host cs.princeton.edu and receiving host cisco.com. In this case, user Bob at Princeton is trying to send mail to users Alice and Tom at Cisco.

HELO cs.princeton.edu
250 Hello daemon@mail.cs.princeton.edu [128.12.169.24]
MAIL FROM:<Bob@cs.princeton.edu>
250 OK
RCPT TO:<Alice@cisco.com>
250 OK
RCPT TO:<Tom@cisco.com>
550 No such user here
DATA
354 Start mail input; end with <CRLF>.<CRLF>
Blah blah blah...
...etc. etc. etc.
<CRLF>.<CRLF>
250 OK
QUIT
221 Closing connection

- As you can see, SMTP involves a sequence of exchanges between the client the server. In each exchange, the client posts a command (e.g., HELO, MAIL, RCPT,DATA, QUIT) and the server responds with a code (e.g., 250, 550, 354, 221).

Mail Reader

- The final step is for the user to actually retrieve his or her messages from the mailbox read them, reply to them, and possibly save a copy for future reference. The user performs all these actions by interacting with .a mail reader.
- In many cases, this reader is just a program running on the same machine as the user's mailbox resides, in which case it simply reads and writes the file that implements the mailbox.

- In other cases, the user accesses his or her mailbox from a remote machine using yet another protocol, such as the Post Office Protocol (POP) or the Internet Message Access Protocol (IMAP).

Message Transfer Agent: SMTP

- The actual mail transfer is done through message transfer agents. To send mail, a system must have the client MTA, and to receive mail, a system must have a server MTA.
- The formal protocol that defines the MTA client and server in the Internet is called the Simple Mail Transfer Protocol (SMTP).

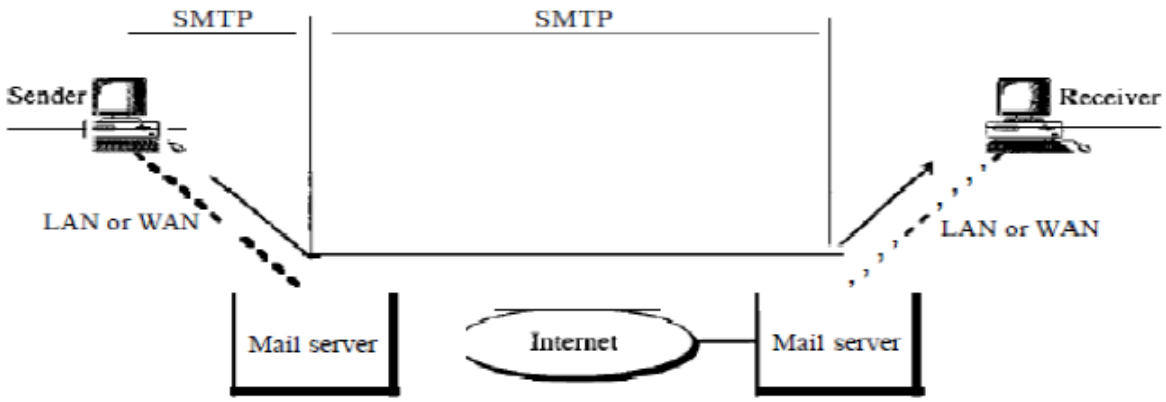


Fig: 5.4.SMTP range

SMTP simply defines how commands and responses must be sent back and forth

Commands and Responses

- SMTP uses commands and responses to transfer messages between an MTA client and an MTA server.

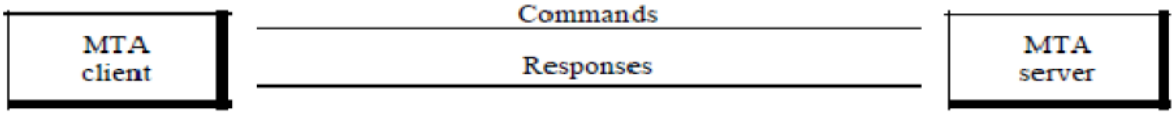


Fig 5.5. command/ response

- Each command or reply is terminated by a two-character (carriage return and line feed) end-of line token.
- Commands Commands are sent from the client to the server.
- It consists of a keyword followed by zero or more arguments.
- SMTP defines 14 commands. The first five are mandatory; every implementation must support these five commands. The next three are often used and highly recommended.
- The last six are seldom used.

Keyword	*Argument(s)*
HELO	Sender's host name
MAIL FROM	Sender of the message
RCPTTO	Intended recipient of the message
DATA	Body of the mail
QUIT	
RSET	
VRFY	Name of recipient to be verified
NOOP	
TURN	
EXPN	Mailing list to be expanded
HELP	Command name

- Responses Responses are sent from the server to the client. A response is a three digit code that may be followed by additional textual information. Mail Transfer Phases The process of transferring a mail message occurs in three phases: connection establishment, mail transfer, and connection termination.

Keyword	*Argument(s)*
SEND FROM	Intended recipient of the message
SMOLFROM	Intended recipient of the message
SMALFROM	Intended recipient of the message

Code	*Description*
	Positive Completion Reply
211	System status or help reply
214	Help message
220	Service ready
221	Service closing transmission channel
250	Request command completed
251	User not local; the message will be forwarded
	Positive Intermediate Reply
354	Start mail input
	Transient Negative Completion Reply
421	Service not available
450	Mailbox not available
451	Command aborted: local error
452	Command aborted: insufficient storage
	Permanent Negative Completion Reply
500	Syntax error; unrecognized command
501	Syntax error in parameters or arguments
502	Command not implemented
503	Bad sequence of commands
504	Command temporarily not implemented
550	Command is not executed; mailbox unavailable
551	User not local
552	Requested action aborted; exceeded storage location
553	Requested action not taken; mailbox name not allowed
554	Transaction failed

$ telnet mail.adelphia.net 25
Trying 68.168.78.100 ...
Connected to mail.adelphia.net (68.168.78.100).

Connection Establishment
220 **mta13.adelphia.net SMTP** serverready Fri, **6 Aug** 2004 . .
HELO mail.adelphia.net
250 mta13.adelphia.net

```
========== Mail Transfer ==========
MAIL FROM: forouzanb@adelphia.net
  250 Sender: <forouzanb@adelphia.net> Ok
RCPT TO: forouzanb@adelphia.net
  250 Recipient <forouzanb@adelphia.net> Ok
DATA
  354 Ok Send data ending with <CRLF>.<CRLF>
From: Forouzan
TO: Forouzan

This is a test message
to show SMTP in action.

========== Connection Termination ==========
  250 Message received: adelphia.net@mail.adelphia.net
QUIT
  221 mta13.adelphia.net SMTP server closing connection
Connection closed by foreign host.
```

Message Access Agent: POP and IMAP

- They are called a pull protocol; the client must pull messages from the server.
- The direction of the bulk data is from the server to the client. The third stage uses a message access agent
- Currently two message access protocols are available: Post Office Protocol, version 3 (POP3) and Internet Mail Access Protocol, version 4 (IMAP4).

IMAP

- IMAP is similar to SMTP in many ways.
- It is a client/server protocol running over TCP, where the client (running on user's desktop machine) issues commands in the form of <CRLF>-terminated ASCII text lines and the mail server (running on the machine that maintains the user's mailbox) responds in kind.

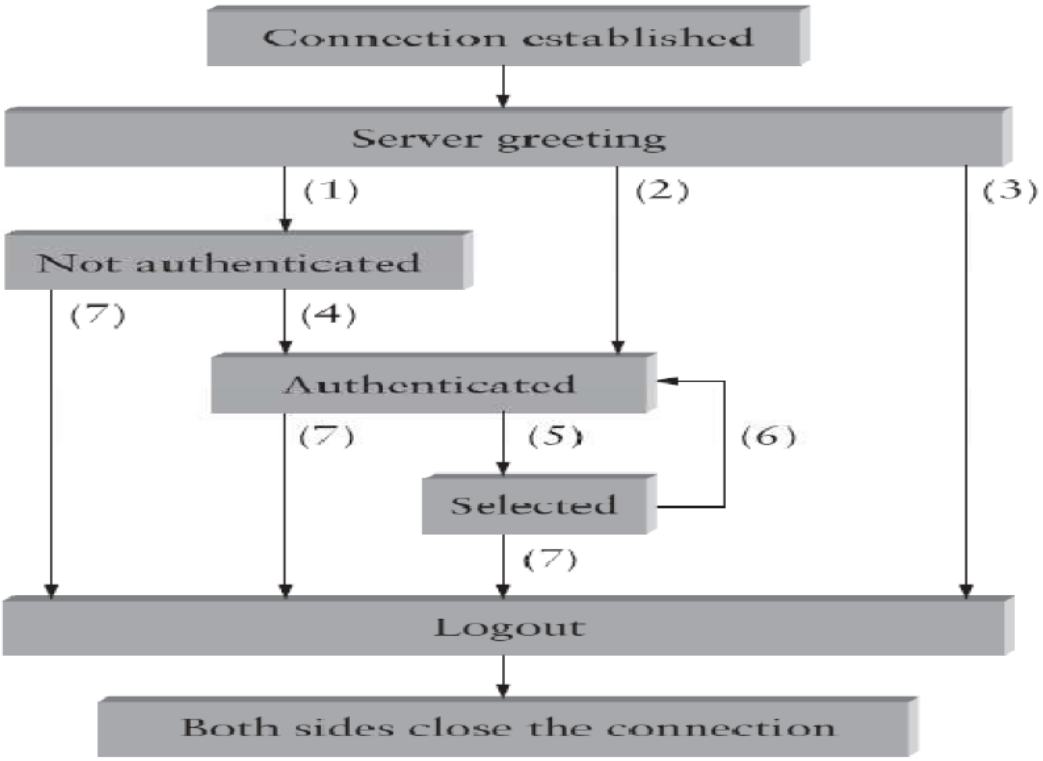


Fig:5.6. IMAP

- In this diagram, LOGIN, AUTHENTICATE, SELECT, EXAMINE, CLOSE, and LOGOUT are example commands that the client can issue, while OK is one possible server response.
- Other common commands include FETCH, STORE, DELETE, and EXPUNGE, with the obvious meanings. Additional server responses include NO (client does not have permission to perform that operation) and BAD (command is ill formed).
- When the user asks to FETCH a message, the server returns it in MIME format and the mail reader decodes it. In addition to the message itself, IMAP also defines a set of message attributes that are exchanged as part of other commands, independent of transferring the message itself.
- Message attributes include information like the size of the message, but more interestingly, various flags associated with the message (e.g., Seen, Answered, Deleted, and Recent).
- These flags are used to keep the client and server synchronized; that is, when the user deletes a message in the mail reader, the client needs to report this fact to the mail server.
- Later, should the user decide to expunge all deleted messages, the client issues an EXPUNGE command to the server,which knows to actually remove all earlier deleted messages from the mailbox.

IMAP4

- Another mail access protocol is Internet Mail Access Protocol, version 4 (IMAP4).
- IMAP4 is similar to POP3, but it has more features; IMAP4 is more powerful and more complex.
- POP3 is deficient in several ways. It does not allow the user to organize her mail on the server; the user cannot have different folders on the server. (Of course, the user can create folders on her own computer.).
- In addition, POP3 does not allow the user to partially check the contents of the mail before downloading.
- IMAP4 provides the following extra functions:
- A user can check the e-mail header prior to downloading.
- A user can search the contents of the e-mail for a specific string of characters prior to downloading.
- A user can partially download e-mail. This is especially useful if bandwidth is limited and the email contains multimedia with high bandwidth requirements.
- A user can create, delete, or rename mailboxes on the mail server. A user can create a hierarchy of mailboxes in a folder for e-mail storag

POP3

- **Mail access starts with the client when the user needs to download e-mail from the mailbox on the mail server.**
- **The client opens a connection to the server on TCP port 110.**
- It then sends its user name and password to access the mailbox.
- The user can then list and retrieve the mail messages, one by one.
- POP3 has two modes: the delete mode and the keep mode.
- In the delete mode, the mail is deleted from the mailbox after each retrieval.
- In the keep mode, the mail remains in the mailbox after retrieval. The delete mode is normally used when the user is working at her permanent computer and can save and organize the received mail after reading or replying.
- The keep mode is normally used when the user accesses her mail away from her primary computer (e.g., a laptop). The mail is read but kept in the system for later retrieval and organizing.

5.1.2 World Wide Web (HTTP)

- Any Web browser has a function that allows the user to ―open a URL. URLs (uniform resource locators) provide information about the location of objects on the Web; they look like the following.

http://www.cs.princeton.edu/index.html

- If you opened that particular URL, your Web browser would open a TCP connection to the Web server at a machine called www.cs.princeton.edu and immediately retrieve and display the file called index.html.
- Most files on the Web contain images and text, and some have audio and video clips.
- They also include URLs that point to other files, and your Web browser will have some way in which you can recognize URLs and ask the browser to open them. These embedded URLs are called hypertext links.
- When you select to view a page, your browser (the client) fetches the page from the server using HTTP running over TCP. LikeSMTP, HTTPis a text-oriented protocol.

At its core, each HTTP message has the general form
START_LINE <CRLF>
MESSAGE_HEADER <CRLF>
<CRLF>
MESSAGE_BODY <CRLF>
<CRLF> stands for carriage-return-line-feed.
The first line (START LINE) indicates whether this is a request message or a response message.
There are zero or more of these MESSAGE HEADER lines—the set is terminated by a blank line—
each of which looks like a header line in an email message.

- HTTP defines many possible header types, some of which pertain to request messages, some to response messages, and some to the data carried in the message body.
- Instead of giving the full set of possible header types, though, we just give a handful of representative examples.
- Finally, after the blank line comes the contents of the requested
- message (MESSAGE BODY); this part of the message is typically empty for request messages.

Request Messages

- The first line of an HTTP request message specifies three things: the operation to be performed, the Web page the operation should be performed on, and the version of HTTP being used.
- Two most widely used request operations are
- GET (fetch the specified Web page) and
- HEAD (fetch status information about the specified Web page). The former is obviously used when your browser wants to retrieve and display a Web page.

- The latter is used to test the validity of a hypertext link or to see if a particular page has been modified since the browser last fetched it.

Operation	Description
OPTIONS	request information about available options
GET	retrieve document identified in URL
HEAD	retrieve metainformation about document identified in URL
POST	give information (e.g., annotation) to server
PUT	store document under specified URL
DELETE	delete specified URL
TRACE	loopback request message
CONNECT	for use by proxies

HTTP request operations

For example, the START LINEGET http://www.cs.princeton.edu/index.html HTTP/1.1 says that the client wants the server on host www.cs.princeton.edu to return the page named index.html. This particular example uses an absolute URL. It is also possible to use a relative identifier and specify the host name in one of the MESSAGE HEADER lines; for example,

GET index.html HTTP/1.1
Host: www.cs.princeton.edu

Response Messages

- Like request messages, response messages begin with a single START LINE.
- In this case, the line specifies the version of HTTP being used, a three-digit code indicating whether or not the request was successful, and a text string giving the reason for the response.

For example, the START LINE
HTTP/1.1 202 Accepted indicates that the server was able to satisfy the request, while HTTP/1.1 404 Not Found indicates that it was not able to satisfy the request because the page was not found.

Code	Type	Example Reasons
1xx	Informational	request received, continuing process
2xx	Success	action successfully received, understood, and accepted
3xx	Redirection	further action must be taken to complete the request
4xx	Client Error	request contains bad syntax or cannot be fulfilled
5xx	Server Error	server failed to fulfill an apparently valid request

Five types of HTTP result codes

TCP Connections:

- The original version of HTTP (1.0) established a separate TCP connection for each data item retrieved from the server.
- It's not too hard to see how this was a very inefficient mechanism: Connection setup and teardown messages had to be exchanged between the client and server even if all the client wanted to do was verify that it had the most recent copy of a page.
- Thus, retrieving a page that included some text and a dozen icons or other small graphics would result in 13 separate TCP connections being established and closed.
- The most important improvement in the latest version of HTTP (1.1) is to allow persistent connections—the client and server can exchange multiple request/response messages over the same TCP connection.
- Persistent connections have two advantages
- First, they obviously eliminate the connection setup overhead, thereby reducing the load on the server, the load on the network caused by the additional TCP packets, and the delay perceived by the user.
- Second, because a client can send multiple request messages down a single TCP connection, TCP's congestion window mechanism is able to operate more efficiently.

Caching

- how to effectively cache Web pages.

Caching has many benefits:

- the client's perspective, a page that can be retrieved from a nearby cache can be displayed much more quickly than if it has to be fetched from across the world.

- From the server's perspective, having a cache intercept and satisfy a request reduces the load on the server.
- Caching can be implemented in many different places:
- a user's browser can cache recently accessed pages, and simply display the cached copy if the user visits the same page again.
- ISPs can cache pages
- The machine is caching pages on behalf of the site, and they configure their browsers to connect directly to the caching host. This node is sometimes called a proxy.
- ISP router
- This router can peek inside the request message and look at the URL for the requested page. If it has the page in its cache, it returns it.
- If not, it forwards the request to the server and watches for the response to fly by in the other direction.
- When it does, the router saves a copy in the hope that it can use it to satisfy a future request.
- cache needs to make sure it is not responding with an out-of-date version of the page.
- the server assigns an expiration date (the Expires header field) to each page it sends back to the client (or to a cache between the server and client).
- The cache remembers this date and knows that it need not re verify the page each time it is requested until after that expiration date has passed.
- After that time (or if that header field is not set) the cache can use the HEAD or conditional GET operation (GET with If-Modified-Since header line) to verify that it has the most recent copy of the page.
- More generally, there is a set of ―cache directives that must be obeyed by all caching mechanisms along the request/response chain.
- These directives specify whether or not a document can be cached, how long it can be cached, how fresh a document must be, and so on.

5.2 WEB SERVICES:

5.2.1 DOMAIN NAME SERVICE (DNS)

- Naming service- can be developed to map user-friendly names into router-friendly addresses.
- Name services are sometimes called middleware because they fill a gap between applications and the underlying network.
- Host names differ from host addresses in two important ways. First, they are usually of variable length and mnemonic, thereby making them easier for humans to remember.

- (In contrast, fixed-length numeric addresses are easier for routers to process.)
- Second, names typically contain no information that helps the network locate (route packets toward) the host. Addresses, in contrast, sometimes have routing information embedded in them; flat addresses (those not divisible into component parts) are the exception.
- First, a name space defines the set of possible names.
- A name space can be either flat (names are not divisible into components) or hierarchi-cal (Unix file names are the obvious example). Second, the naming system maintains a collection of bindings of names to values.
- The value can be anything we want the naming system to return when presented with a name; in many cases it is an address.
- a resolution mechanism is a procedure that, when invoked with a name, returns the corresponding value. A name server is a specific implementation of a resolution mechanism that is available on a network and that can be queried by sending it a message.
- The Internet has a particularly well-developed naming system—**the domain name system (DNS).**
- Because of its large size, the Internet has a particularly well-developed naming system in place— the domain name system (DNS).
- Early in its history, when there were only a few hundred hosts on the Internet, a central authority called the Network Information Center (NIC) maintained a flat table of name-to-address bindings; this table was called hosts.txt.
- Whenever a site wanted to add a new host to the Internet, the site administrator sent email to the NIC giving the new host's name/address pair.
- This information was manually entered into the table, the modified table was mailed out to the various sites every few days, and the system administrator at each site installed the table on every host at the site.
- It should come as no surprise that the hosts.txt approach to naming did not work well as the number of hosts in the Internet started to grow.
- Therefore, in the mid-1980s, the domain naming system was put into place.

- DNS employs a hierarchical name space rather than a flat name space, and the ―table of bindings that implements this name space is partitioned into disjoint pieces and distributed throughout the Internet.
- These subtables are made available in name servers that can be queried over the network.

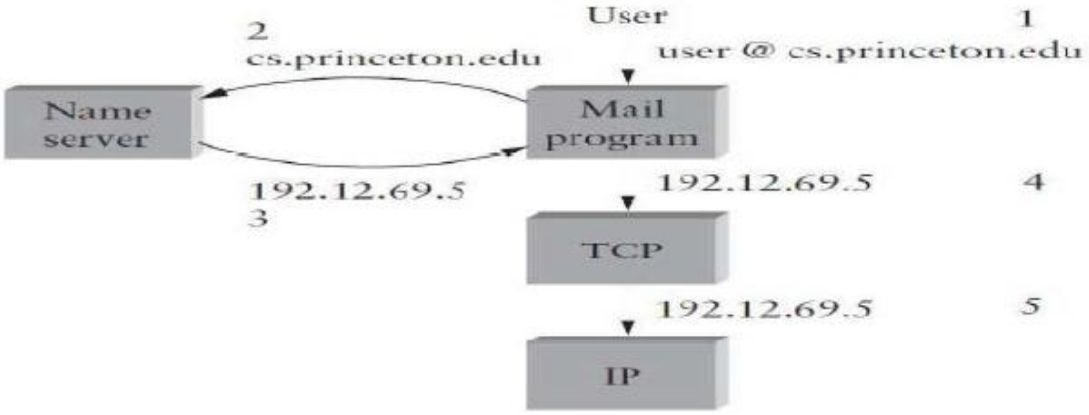


Fig 5.7 Names translated into addresses, where the numbers 1–5 show the se- quence of steps in the process.

Domain Hierarchy

- DNS implements a hierarchical name space for Internet objects.
- DNS names are processed from right to left and use periods as the separator.
- the DNS hierarchy can be visualized as a tree, where each node in the tree corresponds to a domain and the leaves in the tree correspond to the hosts being named.
- There ―big six domains for each country : edu, com, gov, mil, org, and net.

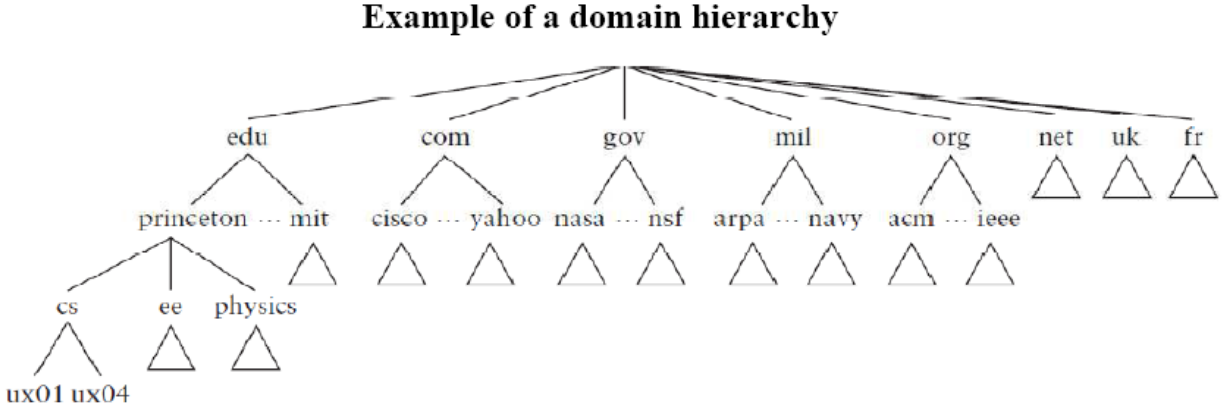


Fig 5.8.Example of a domain hierarchy

Name Servers

- The first step is to partition the hierarchy into subtrees called zones.
- Each zone can be thought of as corresponding to some administrative authority that is responsible for that portion of the hierarchy.
- Specifically, the information contained in each zone is implemented in two or more name servers.
- Each name server, in turn, is a program that can be accessed over the Internet.
- Clients send queries to name servers, and name servers respond with the requested information. For example, the top level of the hierarchy forms a zone that is managed by the NIC.

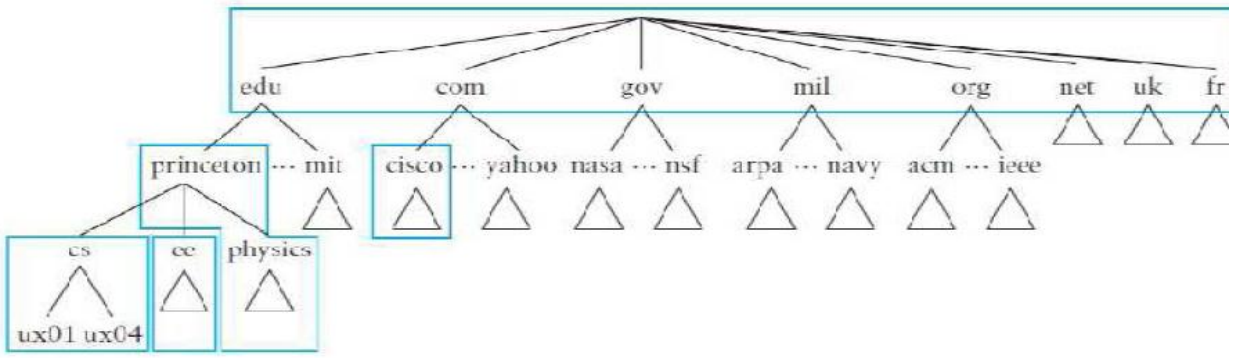


Fig 5.9.Domain hierarchy partitioned into zones

- Sometimes the response contains the final answer that the client wants, and sometimes the response contains a pointer to another server that the client should query next.

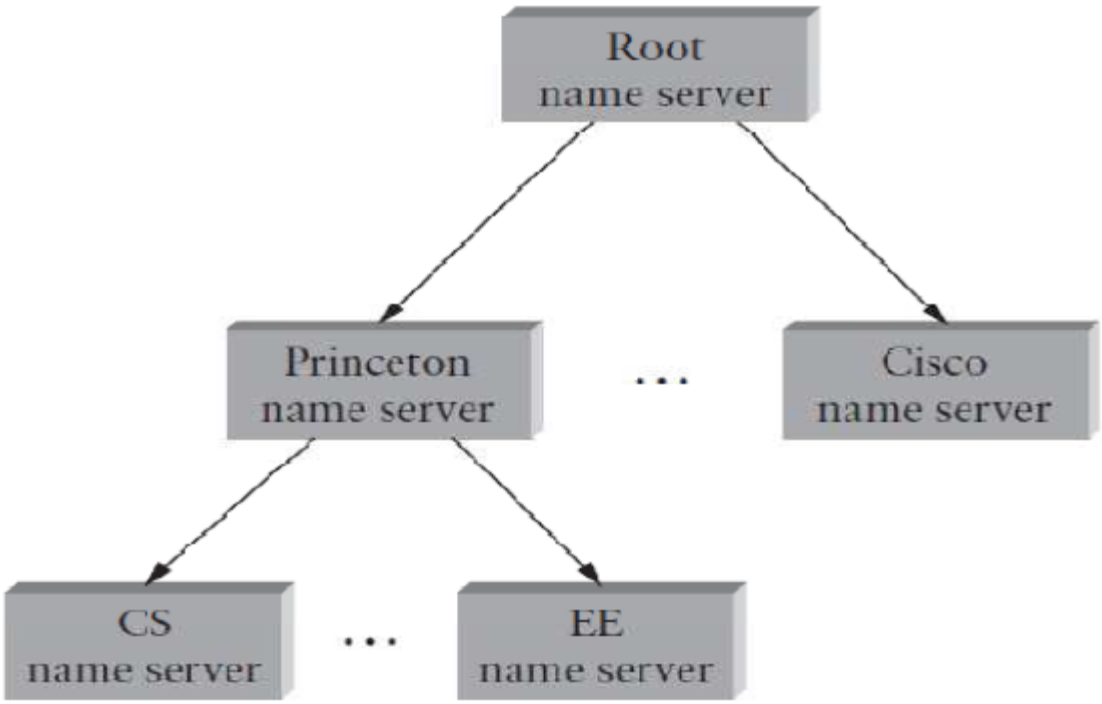


Fig 5.10.Hierarchy of name servers

- Each name server implements the zone information as a collection of resource records. In essence, a resource record is a name-to-value binding, or more specifically, a 5-tuple that contains the following fields: Name, Value, Type, Class, TTL
- The Name and Value fields are exactly what you would expect, while the Type field specifies how the Value should be interpreted.
- For example, Type = A indicates that the Value is an IP address. Thus, A records implement the
- name-to-address mapping we have been assuming.
- NS: The Value field gives the domain name for a host that is running a name server that knows how to resolve names within the specified domain.
- CNAME: The Value field gives the canonical name for a particular host; it is used to define aliases.
- MX: The Value field gives the domain name for a host that is running a mail server that accepts messages for the specified domain.
- The Class field was included to allow entities other than the NIC to define useful record types.

- To date, the only widely used Class is the one used by the Internet; it is denoted IN. Finally, the TTL field shows how long this resource record is valid.
- It is used by servers that cache resource records from other servers; when the TTL expires, the server must evict the record from its cache.
- First, the root name server contains an NS record for each second-level server. It also has an A record that translates this name into the corresponding IP address.
- Taken together, these two records effectively implement a pointer from the root name server to each of the second-level servers.

princeton.edu, cit.princeton.edu, NS, IN
cit.princeton.edu, 128.196.128.233, A, IN
cisco.com, ns.cisco.com, NS, IN
ns.cisco.com, 128.96.32.20, A, IN

- Next, the domain princeton.edu has a name server available on host cit.princeton edu that contains the following records.
- Note that some of these records give the final answer (e.g., the address for host saturn.physics.princeton.edu), while others point to third-level name servers.

cs.princeton.edu, gnat.cs.princeton.edu, NS, IN
gnat.cs.princeton.edu, 192.12.69.5, A, IN
ee.princeton.edu, helios.ee.princeton.edu, NS, IN
helios.ee.princeton.edu, 128.196.28.166, A, IN
jupiter.physics.princeton.edu, 128.196.4.1, A, IN
saturn.physics.princeton.edu, 128.196.4.2, A, IN
mars.physics.princeton.edu, 128.196.4.3, A, IN
venus.physics.princeton.edu, 128.196.4.4, A, IN
cs.princeton.edu, gnat.cs.princeton.edu, MX, IN
cicada.cs.princeton.edu, 192.12.69.60, A, IN
cic.cs.princeton.edu, cicada.cs.princeton.edu, CNAME, IN
gnat.cs.princeton.edu, 192.12.69.5, A, IN
gna.cs.princeton.edu, gnat.cs.princeton.edu, CNAME, IN
www.cs.princeton.edu, 192.12.69.35, A, IN
cicada.cs.princeton.edu, roach.cs.princeton.edu, CNAME, IN

Name Resolution

- Resolving a name actually involves a client querying the local server, which in turn acts as a client that queries the remote servers on the original client's behalf.

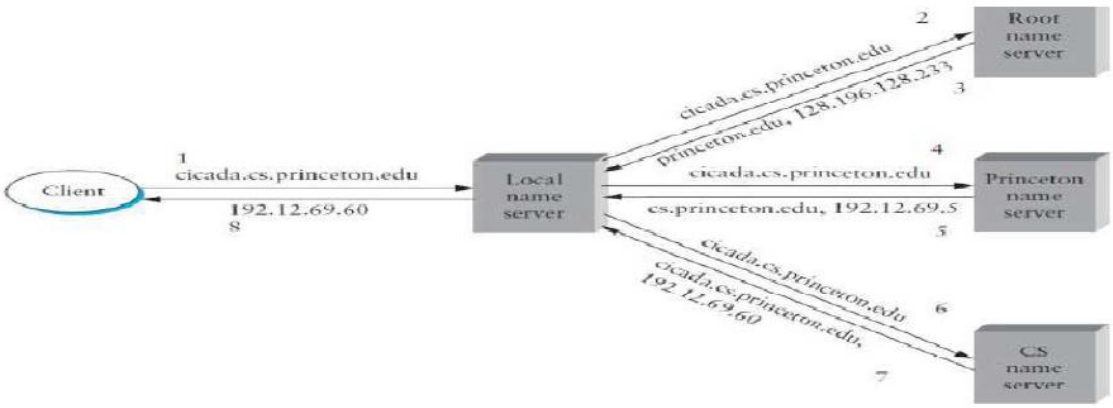


Fig 5.11.Name resolution in practice, where the numbers 1–8 show the sequence of steps in the process.

Advantages

1.All the hosts in the Internet do not have to be kept up-to-date on where the current root servers are located; only the servers have to know about the root.

2. The local server gets to see the answers that come back from queries that are posted by all the local clients. The local server caches these responses and is sometimes able to resolve future queries without having to go out over the network.

5.2.2 SNMP

- This means we need a protocol that allows us to read, and possibly write, various pieces of state information on different network nodes. The most widely used protocol for this purpose is the Simple Network Management Protocol (SNMP).
- SNMP is essentially a specialized request/reply protocol.
- It supports two kinds of request messages: GET and SET.
- GET- is used to retrieve a piece of state from some node, SET- is used to store a new piece of state in some node.
- Whenever the administrator selects a certain piece of information that he or she wants to see, the client program uses SNMP to request that information from the node in question.
- An SNMP server running on that node receives the request, locates the appropriate piece of information, and returns it to the client program, which then displays it to the user.
- There is only one complication to this , Exactly how does the client indicate which piece of information it wants to retrieve, and likewise, how does the server know which variable in memory to read to satisfy the request.

- SNMP depends on a companion specification called the management information base (MIB). The MIB defines the specific pieces of information—the MIB variables—that you can retrieve from a network node.
- The current version of MIB, called MIB-II, organizes variables into 10 different groups.
- System: general parameters of the system (node) as a whole, including where t he node is located, how long it has been up, and the system's name.
- Interfaces: information about all the network interfaces (adaptors) attached to this node, such as the physical address of each interface, how many packets have been sent and received on each interface.
- Address translation: information about the Address Resolution Protocol (ARP), and in particular, the contents of its address translation table.
- IP: variables related to IP, including its routing table, how many datagrams it has successfully forwarded, and statistics about datagram reassembly.
- Includes counts of how many times IP drops a datagram for one reason or another.
- TCP: information about TCP connections, such as the number of passive and active opens, the number of resets, the number of timeouts, default timeout settings, and so on.
- Per-connection information persists only as long as the connection exists.

- UDP: information about UDP traffic, including the total number of UDP datagrams that have been sent and received.

- Two problems remain.

- First, we need a precise syntax for the client to use to state which of the MIB variables it wants to fetch.

- Second, we need a precise representation for the values returned by the server. Both problems are addressed using ASN.1.

- The MIB uses this identification system to assign a globally unique identifier to each MIB variable

- These identifiers are given in a ―dot‖ notation, not unlike domain names.

- For example, 1.3.6.1.2.1.4.3 is the unique ASN.1 identifier for the IP-related MIB variable ipInReceives; this variable counts the number of IP datagrams that have been received by this node. Thus, network management works as follows. The SNMP client puts the ASN.1 identifier for the MIB variable it wants to get into the request message, and it sends this message to the server.
- Abstract Syntax Notation One (ASN.1) is an ISO standard that defines, among other things, a representation for data sent over a network. The representation-specific part of ASN.1 is called the Basic Encoding Rules (BER).
- The server then maps this identifier into a local variable (i.e., into a memory location where the value for this variable is stored), retrieves the current value held in this variable, and uses ASN.1 BER to encode the value it sends back to the client.