COMPUTER NETWORKS

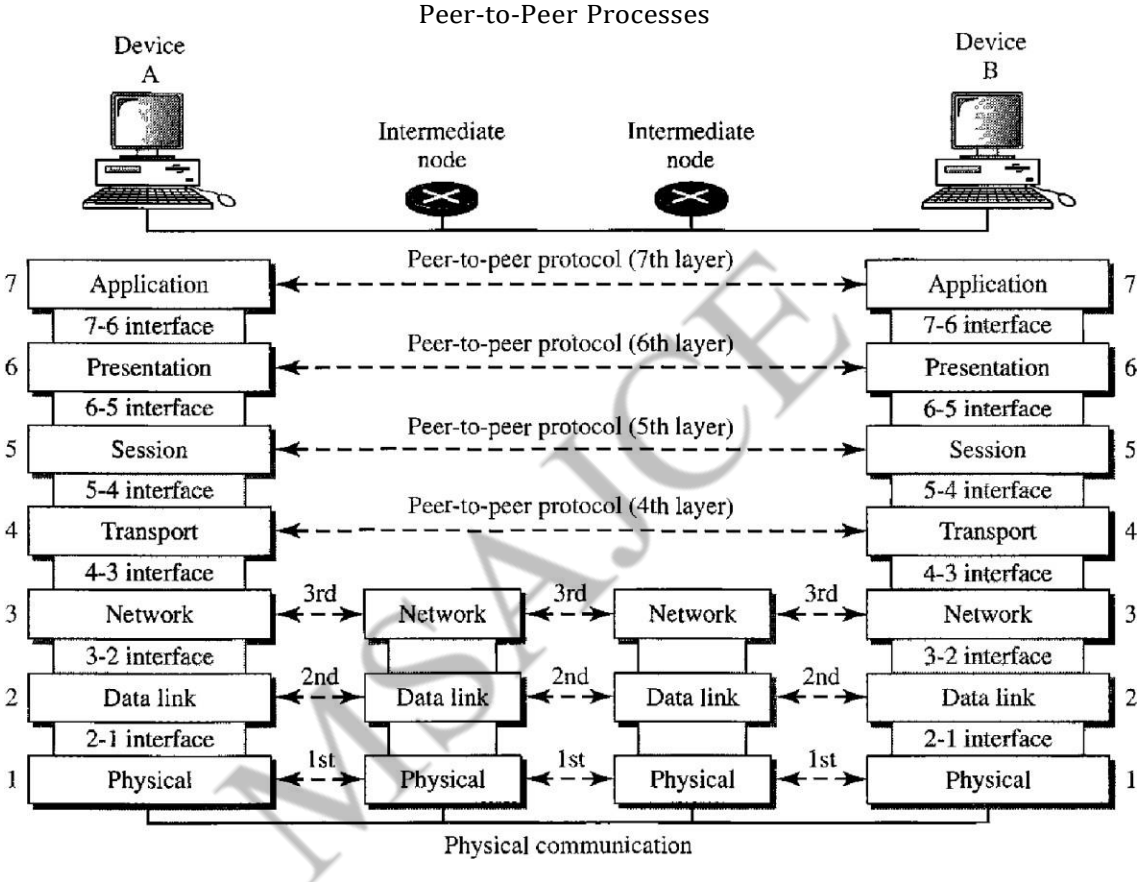
CS6551 COMPUTER NETWORKS

UNIT - 1

OSI MODEL (Open System Interconnection)

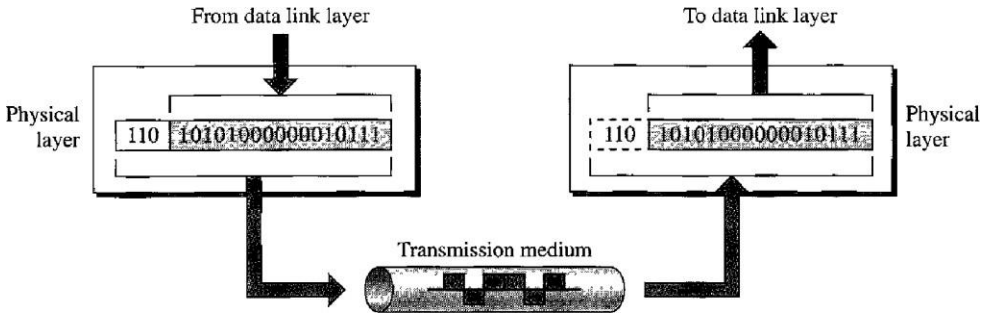
The OSI model is a layered frame work for the design of network systems that allows communication between all types of computer systems. It consists of seven separate but related layers.

The OSI model is composed of seven ordered layers: physical (layer1), data link (layer 2), network (layer 3), transport (layer 4), session (layer 5), presentation (layer 6), and application (layer 7). Figure 2.3 show s the layers involved when a message is sent from device A to device B.



Layers 1,2, and 3 - physical, data link, and network - are the network support layers; Layers 5,6, and 7 - session, presentation, and application - can be thought of as the user support layers; Layer 4, the transports layer, links the two subgroups and ensures that what the lower layers have transmitted is in a form that the upper layers .

Physical layer



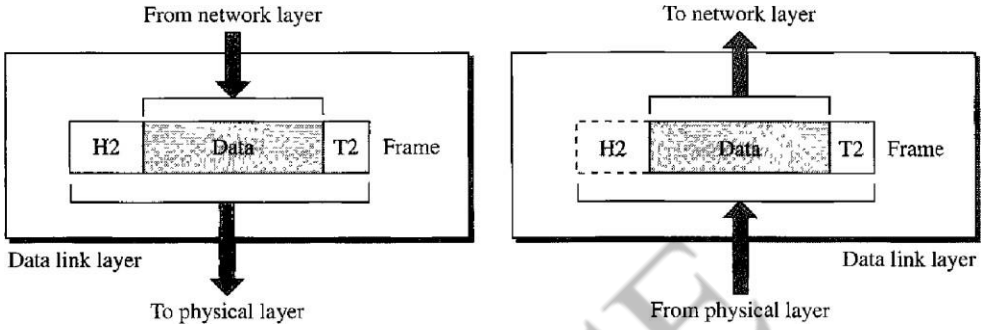
It deals with the mechanical and electrical specifications of the interface and transmission medium. It defines the procedures and functions. The physical layer is responsible for movements of individual bits from one hop (node) to the next.

Data Link Layers

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

Framing. The data link layer divides the stream of bits received from the network layer into manageable data units called frames.

Physical addressing. The data link layer adds a header to the frame to define the sender and/or receiver of the frame.

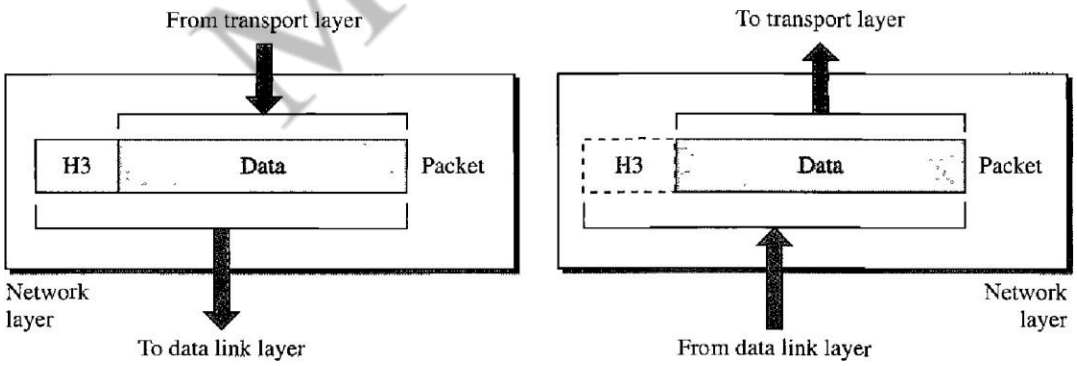


Network Layer

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

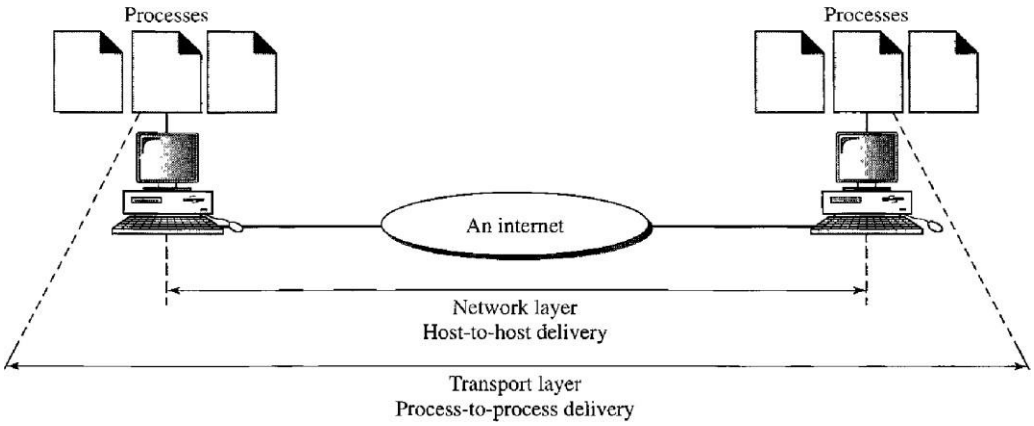
Logical addressing. The physical addressing implemented by the data link layer handles the addressing problem locally. If a packet passes the network boundary. The network layer adds a header to the packet coming from the upper layer that, among other things, includes the logical addresses of the sender and receiver.

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



Transport Layer

The transport layer is responsible for process-to-process delivery of the entire message. A process is an application program running on a host. Whereas the network layer oversees source-to-destination delivery of individual packets, it does not recognize and relationship between those packets. It treats each are independently, as though each piece belonged to a separate message, whether or not it does. The transport layer, on the other hand, ensures that the whole message arrives intact and in order, overseeing both error control and flow control at the source-to-destination level. The transport layer is responsible for the delivery of a message from one process to another.

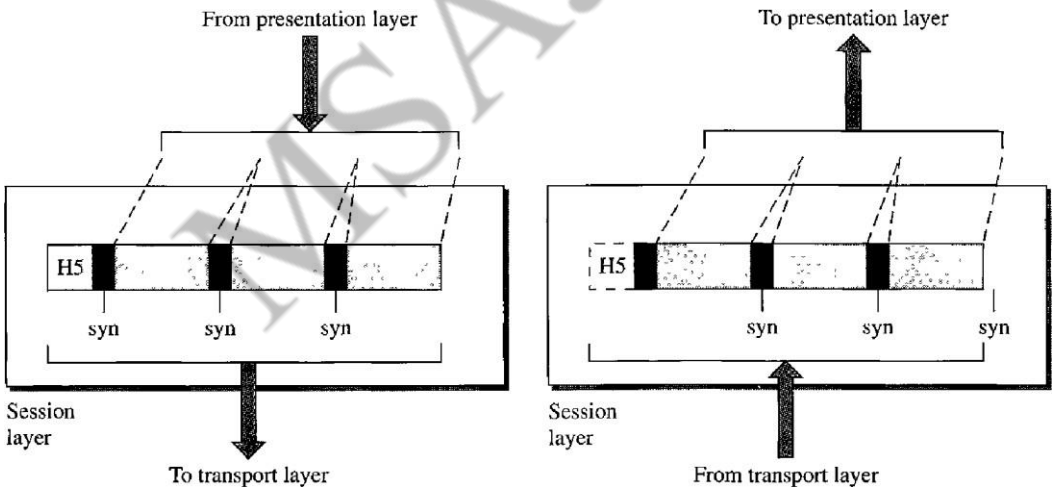


Session Layer

The session layer is the network dialog controller. It establishes, maintains, and synchronizes the interaction among communication systems. The session layer is responsible for dialog control and synchronization.

Dialog control. The session layer allows two systems to enter into a dialog. It allows the communication between two processes to take place in either half-duplex (one way at a time) or full-duplex (two way at a time) mode.

Synchronization. The session layer allows a process to add checkpoints, or synchronization points, to a stream of data. For example, if a system is sending a file of 2000 pages, it is advisable to insert checkpoints after every 100 pages to ensure that each 100-page unit is received and acknowledged independently. In this case, if a crash happens during the transmission of page 523, the only pages that need to be resent after system recovery are pages 501 to523. Pages previous to 501 need not be resent.



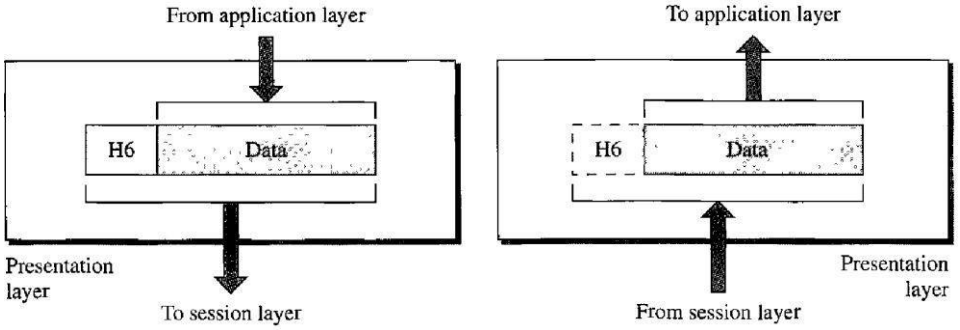
Presentation Layer

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

Translation. The processes (running programs) in two systems are usually exchanging information in the form of character strings, numbers, and so on. The information must be changed to bit streams before being transmitted. Because different computers use different encoding systems, the presentation layer is responsible for interoperability between these different encoding methods. The presentation layer at the sender changes the information from its sender-dependent format into a common format. The presentation layer at the receiving machine changes the common format into its receiver-dependent format.

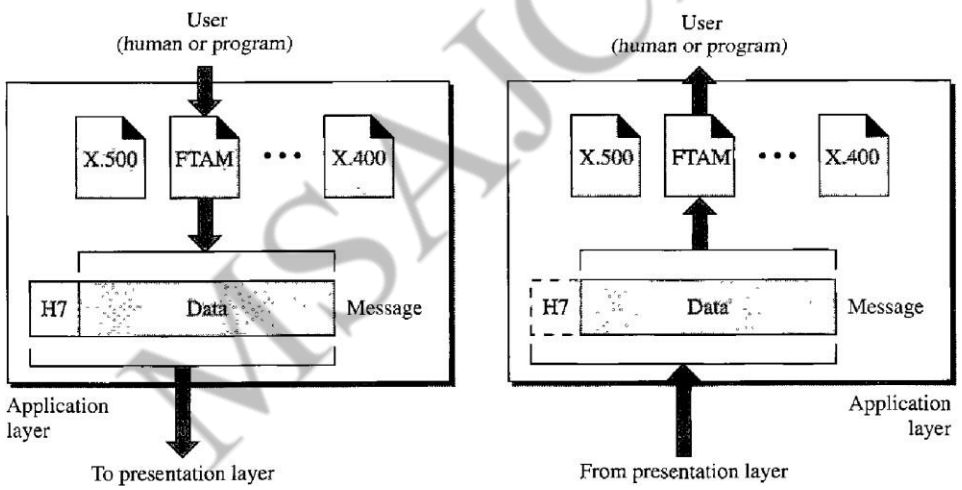
Encryption. To carry sensitive information, a system must be able to ensure privacy. Encryption means that the sender transforms the original information to another form and sends the resulting message out over the network. Decryption reverses the original process to transform the message back to it original form.

Compression. Data compression reduces the number of bits contained in the information. Data compression becomes particularly important in the transmission of multimedia such as text, audio, and video.

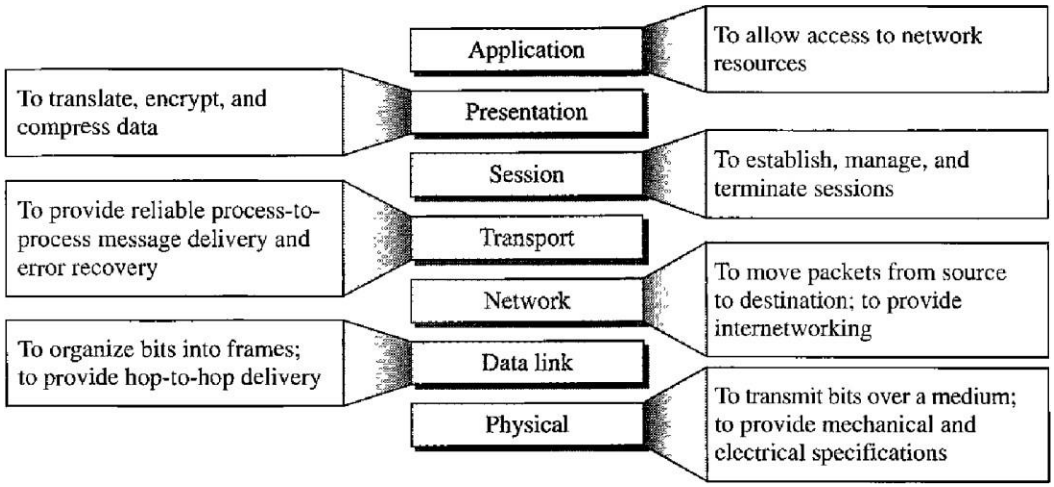


Application Layer

The application layer enables the user, whether human or software, to access the network. It provides user interfaces and support for services such as electronic mail, remote file access and transfer, shared database management, and other types of distributed information services. The application layer is responsible for providing services to the user.



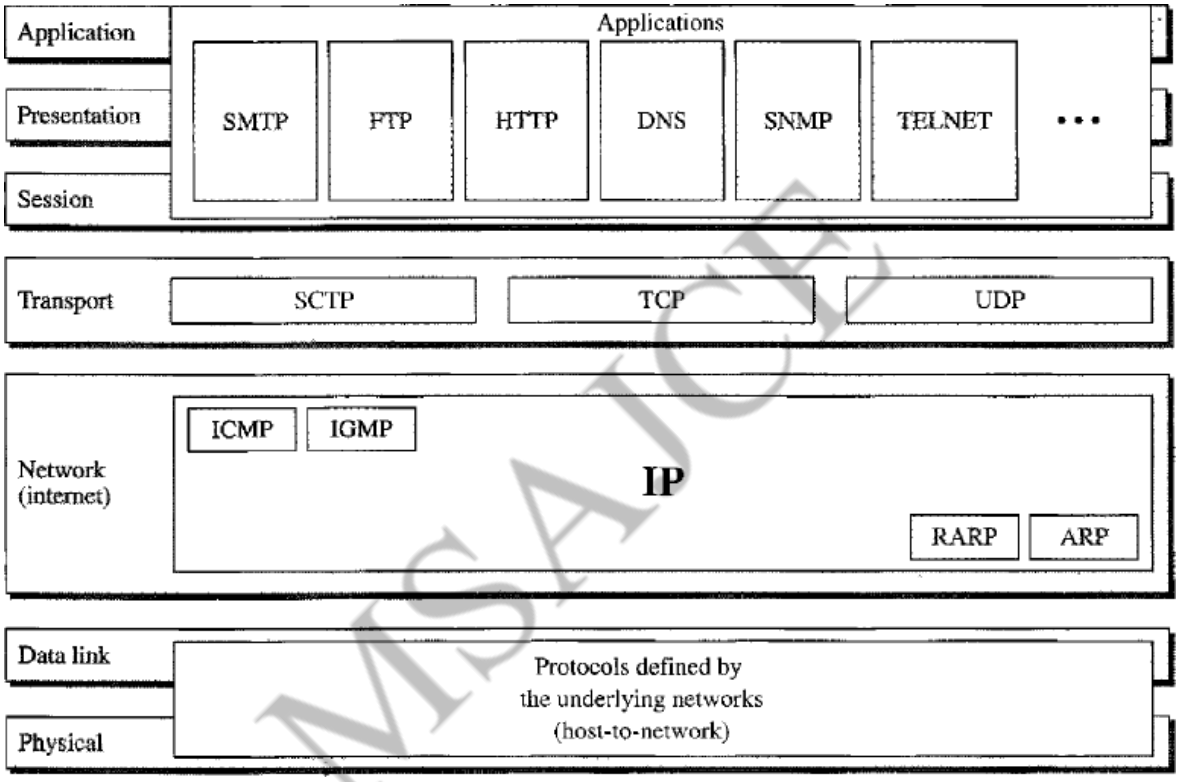
Summary of Layers



TCP/IP PROTOCOL SUITE:

The TCP/IP Protocol suite was developed prior to the OSI model. Therefore, the layers in the TCP/IP Protocol suite do not exactly match those in the OSI model. The original TCP/IP Protocol suite was defined as having four layers :host-to-network, internet, transport and application. However, when TCP/IP is compared to OSI, we can say that the host-to-network layer is equivalent to the combination of the physical and data link layers. The internet layer is equivalent to the network layer, and application layer is roughly doing the job of the session, presentation and application layers with the transport layers in TCP/IP taking care of part of the duties of the session layer.

The three top most layers in the OSI model, are represented in TCP/IP by a single layer called the application layer .



At the transport layer, TCP/IP defines three protocols: Transmission Control Protocol(TCP), User Datagram Protocol(UDP),Stream Control Transmission Protocol(SCTP).

Physical and Data Link Layers

At the Physical and Data Link Layers , TCP/IP does not define any specific protocol. It supports all the standard and proprietary protocols. A network in a TCP/IP internetwork can be a local-area network or a wide-area network.

Network Layer

TCP/IP supports the Internetworking Protocol(IP) uses for supporting protocols: ARP, RARP, ICMP, IGMP.

Internetworking Protocol(IP)

The Internetworking Protocol(IP) is the Transmission mechanism used by the TCP/IP Protocol. It is an unreliable and connectionless Protocol a best-effort delivery service. The term best-effort means that IP provides no error checking or tracking.

IP transports data in packets called Datagrams, each of which is transported separately. Datagrams can travel along different routes and can arrive out of sequence or be

duplicated. IP does not keep track of the routes and has no facility for recording Datagrams once they arrive at their destination.

Address Resolution Protocol

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

Reverse Address Resolution Protocol

The Reverse Address Resolution Protocol(RARP) allows a host to discover its Internet address when it knows only its physical address.

Internet Control Message Protocol

The Internet Control Message Protocol (ICMP) is mechanism used by hosts and gateways to send notification of datagram problems back to the sender. ICMP sends query and error reporting messages.

Internet Group Message Protocol

The Internet Group Message Protocol (IGMP) is used to facilitate the simultaneous transmission of message to a group of recipients.

Transport Layer

IP is host-to-host protocol, meaning that it can deliver a packet from one physical device to another. UDP and TCP are transport level protocols responsible for delivery of a message from a process to another process.

User Datagram Protocol

The User Datagram Protocol (UDP) is the simpler of the two standard TCP/IP Transport Protocols. It is a process to process protocol that adds only port addresses, checksum error control, and length information to the data from the upper layer .

Transmission Control Protocol

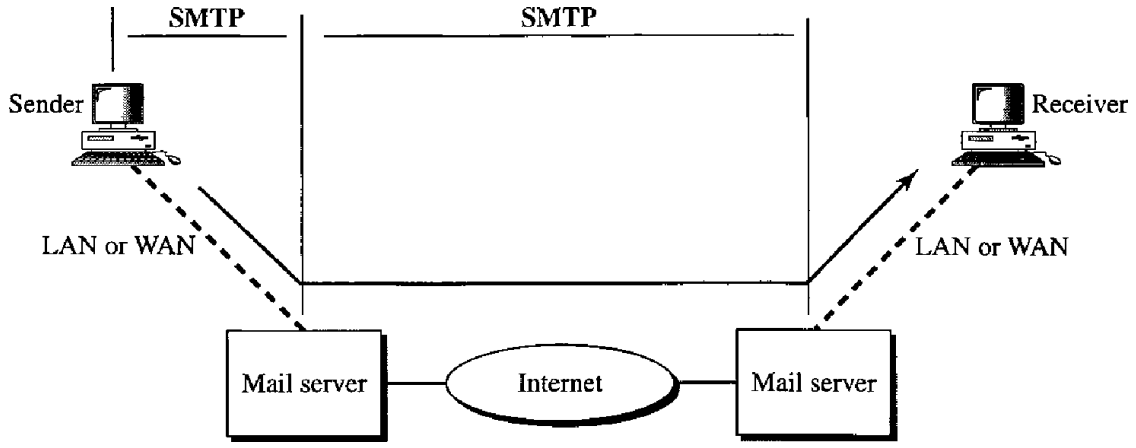
TCP is a reliable stream transport protocol. The term stream, in this context, means connection-oriented: A connection must be established between both ends of a transmission before either can transmit data .

At the sending end of each transmission, TCP divides a stream of data into smaller units called segments. Each segment includes a sequence number for recording after a receipt, together with an acknowledgement number for the segments received. Segments are carried across the internet inside of IP datagrams . At the receiving end, TCP collects each datagram as it comes in and recorders the transmission based on sequence numbers

Stream Control Transmission protocol: The Stream Control Transmission protocol (SCTP) provides support for newer applications such as voice over the internet.

Session, Presentation & Application Layer

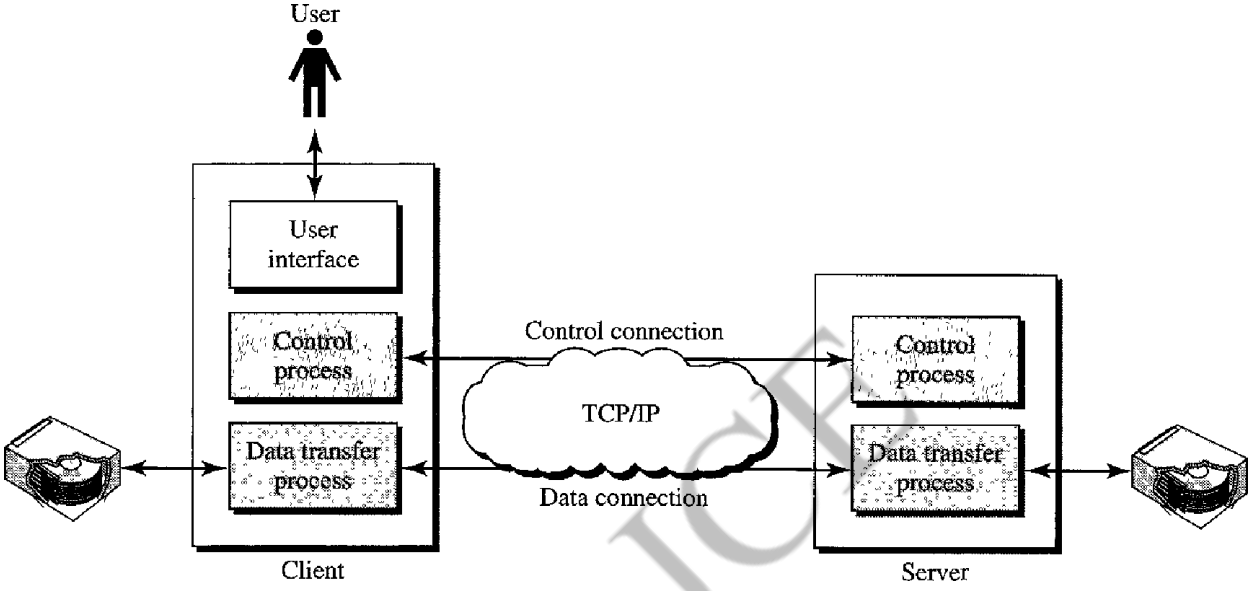
SMTP



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 define the MTA client and server in the Internet is called the ***Simple Mail Transfer Protocol (SMTP).***

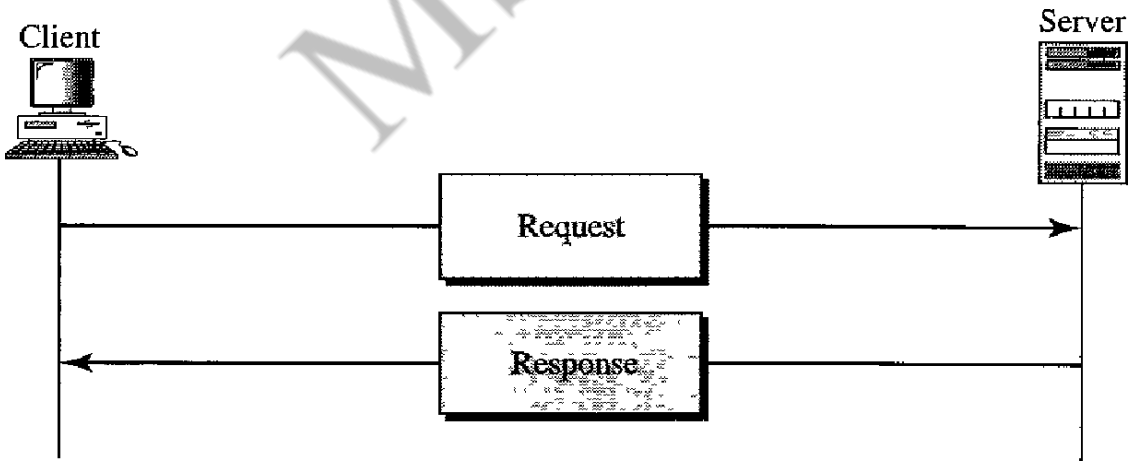
FTP

File Transfer Protocol is the standard mechanism provided by TCP/IP for copying a file from one host to aother. FTP uses two well-known TCP ports : Port 21 is used for the control connection, and port 20 is used for the data connection.



HTTP

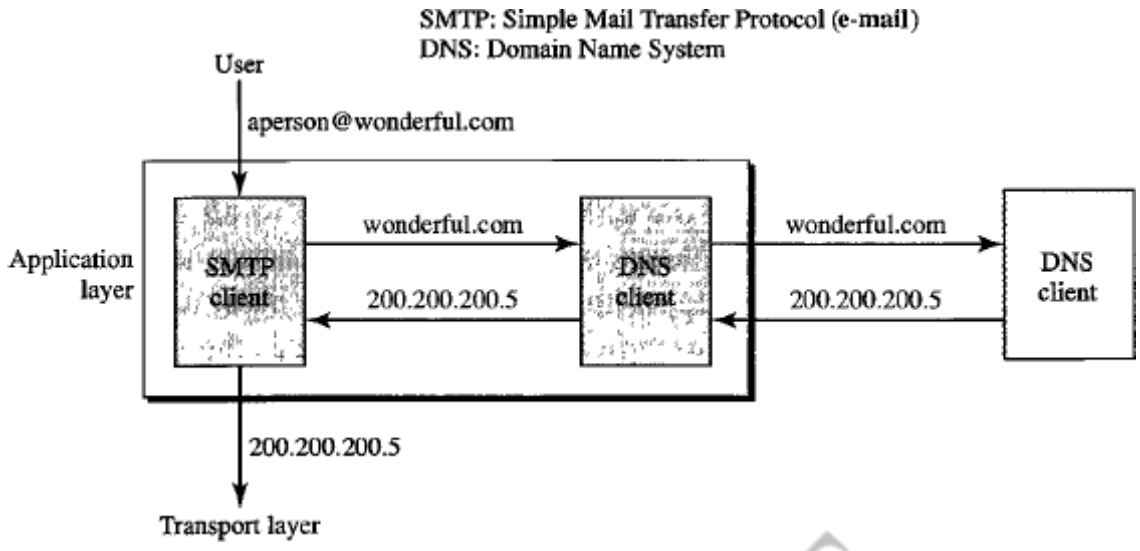
Hypertext Transfer Protocol is a protocol used to access data on the World Wide Wed. HTTP functions as a combination of FTP and SMTP. It is similar to FTP because it transfers files and uses the services of TCP. However it is much simpler than FTP because it uses only one TCP connection. There is no separate control connection, only data are transferred between the client and server.



DNS

Domain Name System client/server program can support an e-mail program to find the IP address of an e-mail recipient. A user of an e-mail program may know the e-mail address of the recipient. The DNS client program sends a request to a DNS server to map the e-mail address to corresponding IP address.

Ex: Domain Name System - **wonderful.com**, IP - **200.200.200.5**

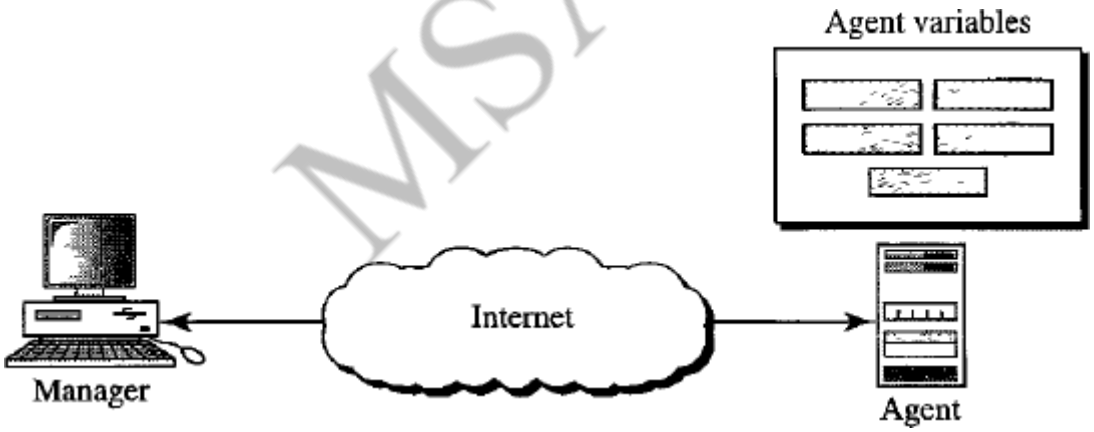


SNMP

Simple Network Management Protocol uses the concept of manager and agent. That is a manager usually a host, controls and monitors a set of agents, usually routers.

Manager (host) runs the SNMP client program, agent (router) runs the SNMP server program. Management is achieved through simple interaction between a manager and an agent.

Agent keeps performance information in a database. The manger has access to the values in the database. For example, a router can store the number of packets received and forwarded. The manager can fetch and compare the values of these two variables to see if the router is congested or not.



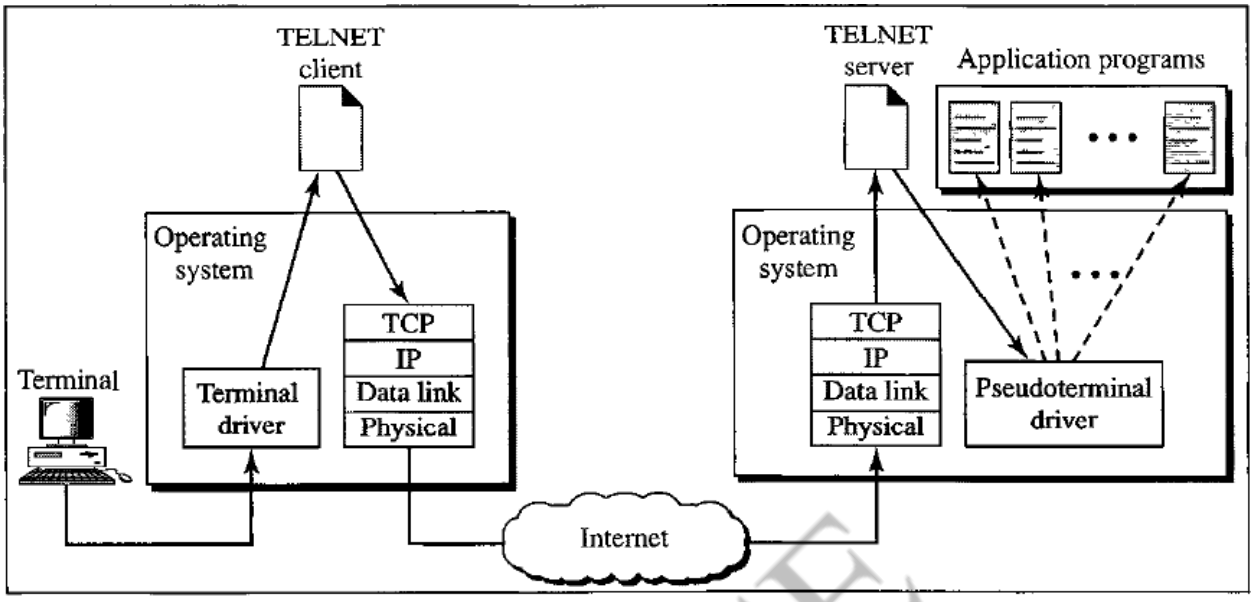
TELNET

TErminal NETwork. It establishes the connection to a remote system in such a way that the local terminal appears to be a terminal at the remote system.

When a user wants to access an application program or utility located on a remote machine, uses the remote-login. TELNET client and server programs come into use. The user sends the keystrokes to the terminal driver, where the local operating system accepts the characters but does not interpret them. The characters are sent to the TELNET client, which transforms the characters to a universal character set called ***network virtual terminal (NVT)*** characters and delivers them to the local TCP/IP.

The commands or text in NVT form, travel through the Internet and arrive at the TCP/IP stack at the remote machine. Here the characters are delivered to the operating system and passed to the TELNET server, which changes the characters to the corresponding

characters understandable by the remote computer using the software called a ***pseudoterminal driver*** software.

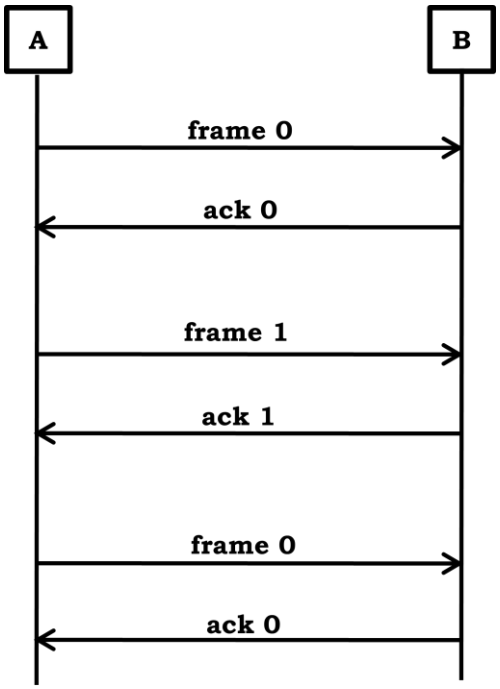


b. Remote log-in

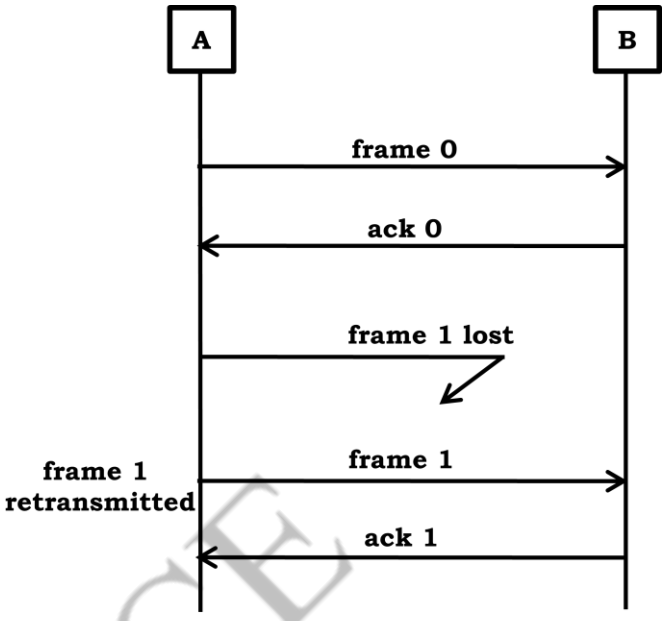
FLOW AND ERROR CONTROL

1. STOP and WAIT protocol

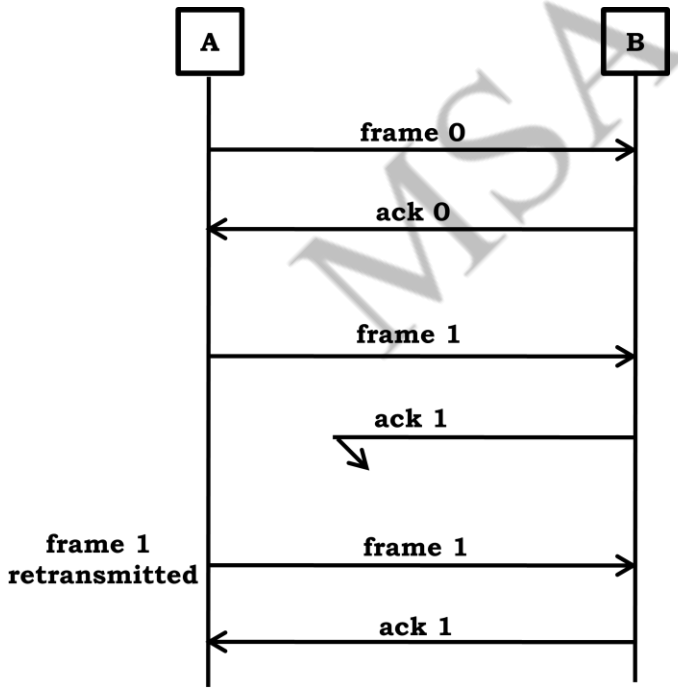
Case 1 : Normal



Case 2 : frame lost

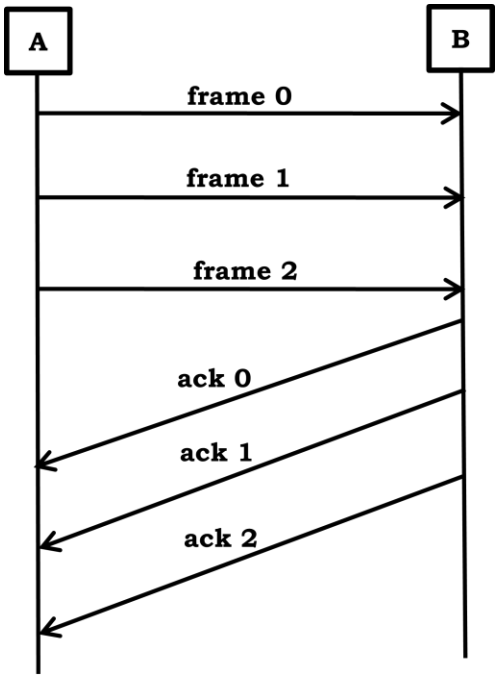


Case 3 : ack lost

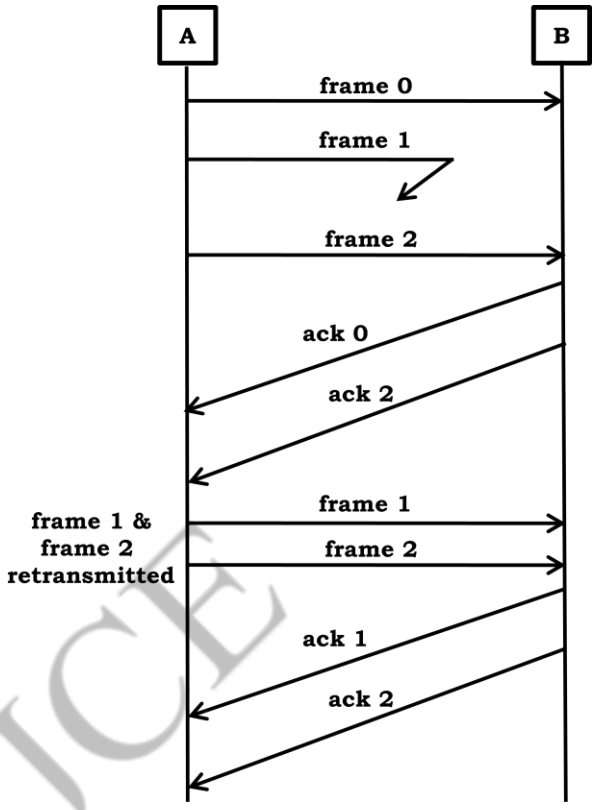


2. Go-Back-N protocol

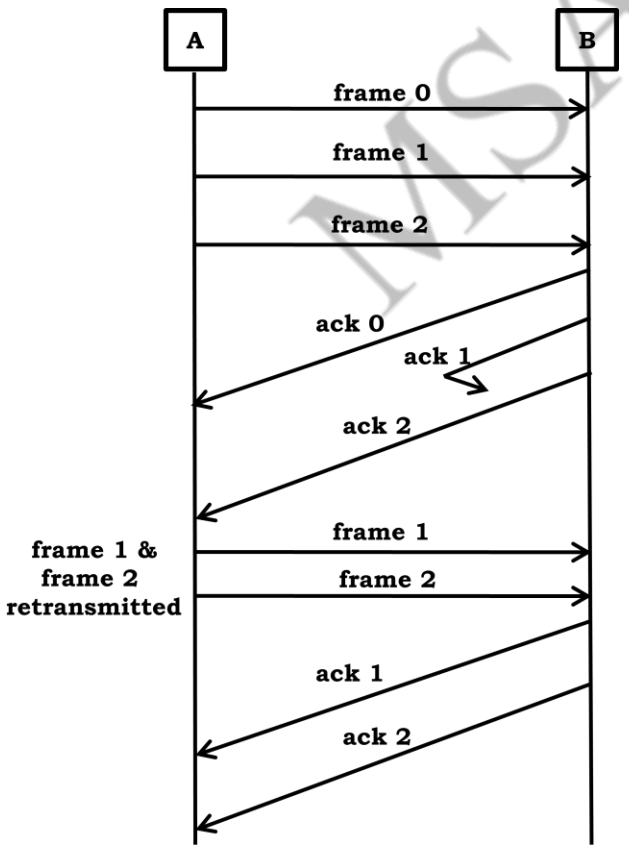
Case 1 : Normal



Case 2 : frame lost

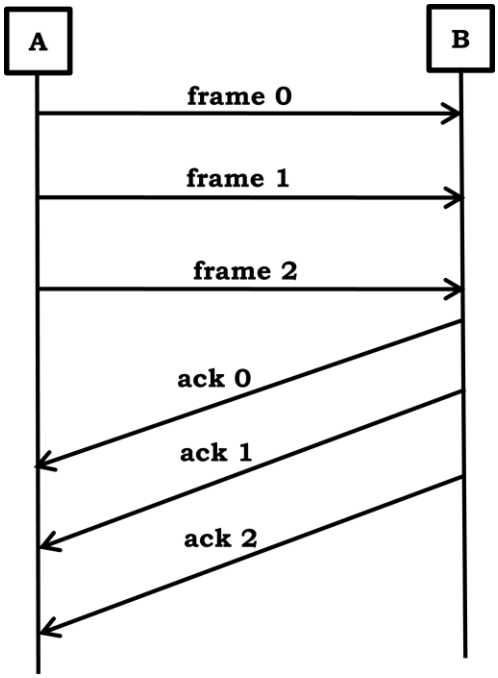


Case 3 : ack lost

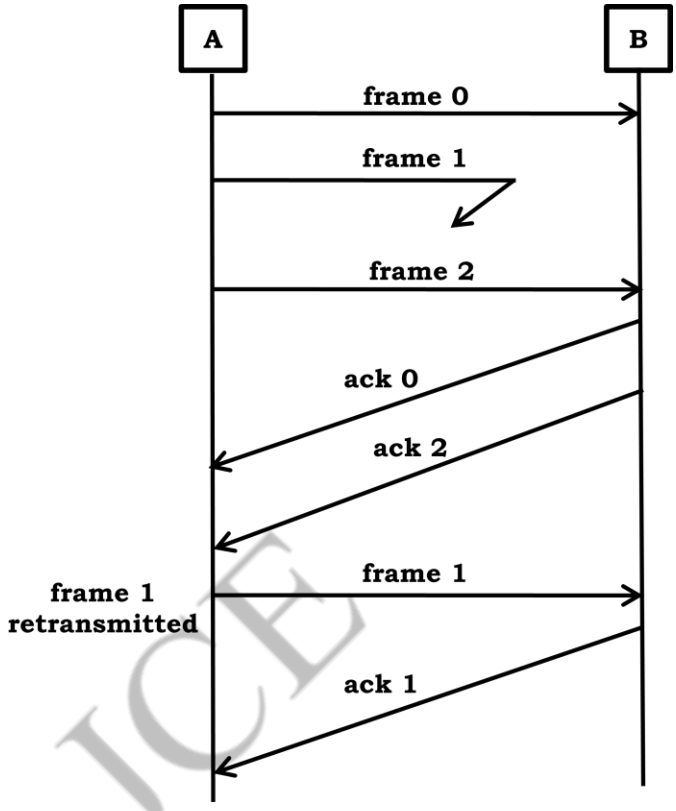


3. SELECTIVE REPREAT protocol

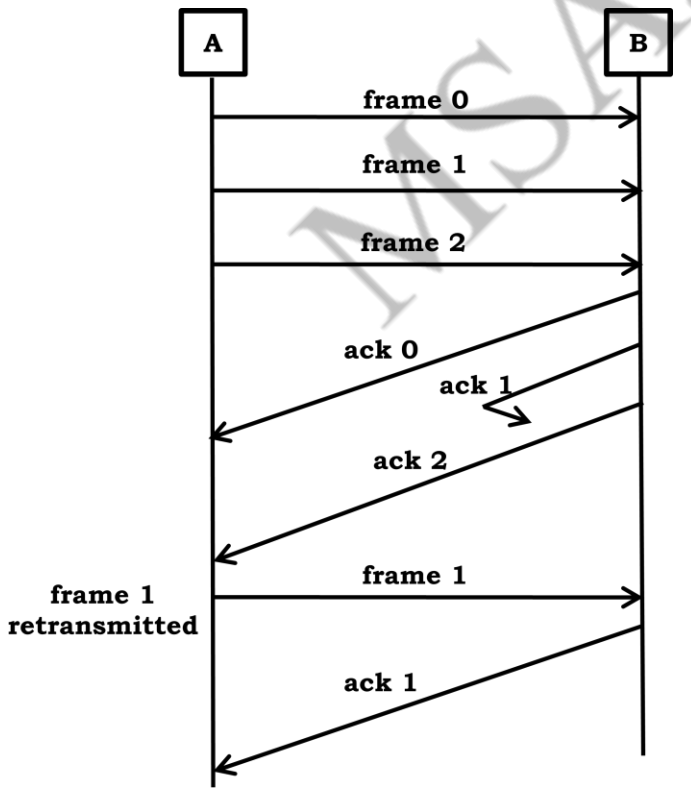
Case 1 : Normal



Case 2 : frame lost



Case 3 : ack lost



FRAMING

FIXED-SIZE FRAMING

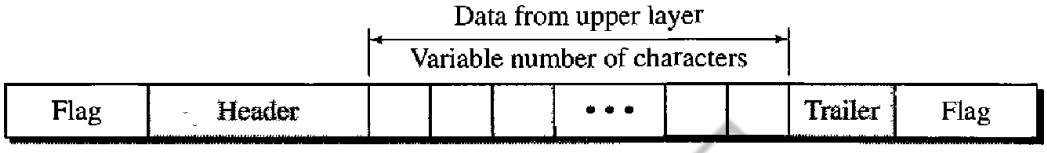
In fixed-size framing, there is no need for defining the boundaries of the framing; the size itself can be used as a delimiter. An example of this type of framing is the ATM wide- area network, which uses frames of fixed size called cells.

VARIABLE -SIZE FRAMING

In variable -size framing, we need a way to define the end of the frame and the beginning of the next. A character -oriented approach and a bit - oriented approach.

Character-oriented protocols

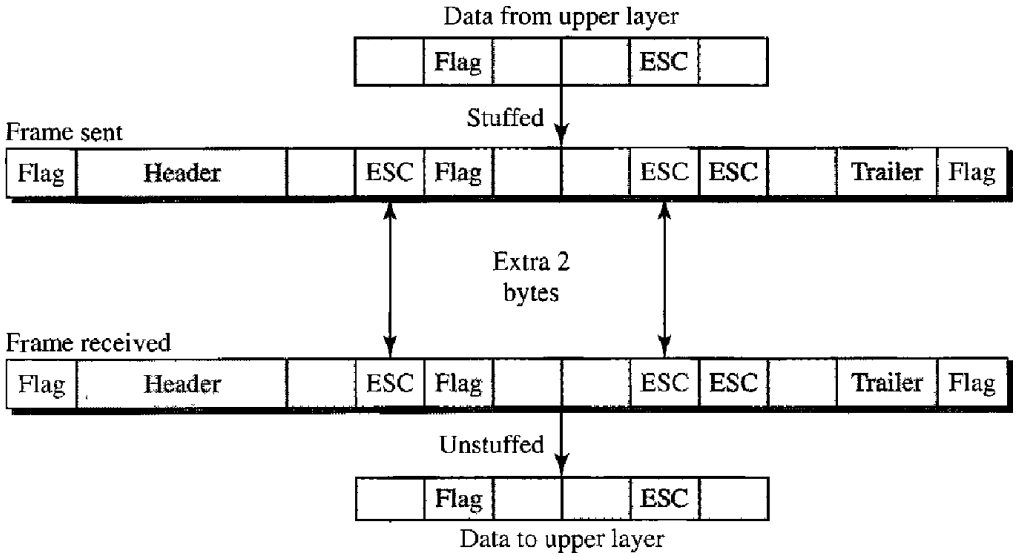
In a character - oriented protocol, data to be carried are 8- bit character from coding system such as a ASCII . The header, which normally carries the source and destination address and control information, and the trailer, which carries error detection or error correction redundant bits, are also multiples of 8nbits. To separate one frame from the next, an 8-bit (1-byte) flag is added at the beginning and the end of a frame.



Character -oriented framing was popular when only text was exchanged by the data link layers. The flag could be selected to be any character not used for text communication. Now, however, we send others types of information such as graphs, audio, and video. Any pattern used for the flag could also be part of the information. If this happens, the receive , when it encounters this pattern in the middle of the data, thinks it has reached the end of the frame .

To fix this problem, a byte-stuffing strategy was added to character -oriented framing . In the byte stuffing (or character stuffing), a special byte is added to the data section of the frame when there is a character with the same pattern as the flag. The data selection is stuffed with an extra byte. This byte is usually called the escape character (ESC), which has a predefined bit pattern. Whenever the receiver encounters the ESC character, it removes it from the data selection and treats the next character as data, not a delimiting flag.

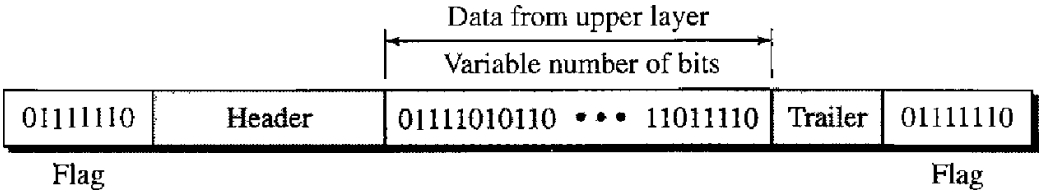
Byte stuffing by the escape character allows the presence of the flag in the data section of the frame. The escape character that are part of the text must also be marked by another escape character .In other words, if the escape character is the part of the text , an extra one is added to show that the second one is part of the text.



Character - oriented protocols present another problem in data communications. The universal coding systems in use today, such as Unicode , have 16- bit and 32- bit characters that conflict with 8-bit characters. We can say that in general, the tendency is moving toward the bit -oriented protocols that we discuss next.

Bit- oriented protocols

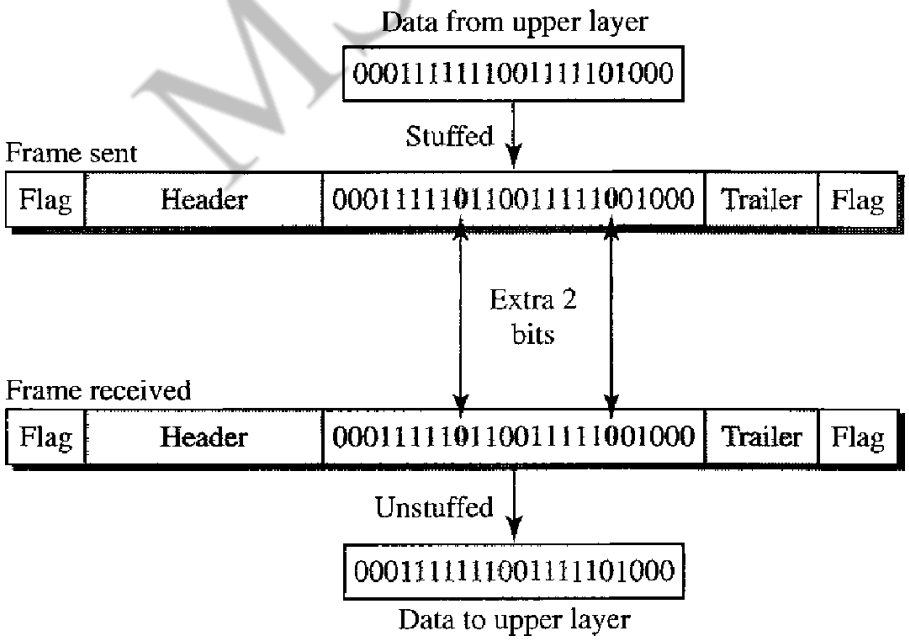
In the bit-oriented protocol, the data section of a frame is a sequence of bits to be interpreted by the upper layer as text, graphic, audio, video, and so on.



However, in addition to headers (and possible trailers), we still need a delimiter to separate one frame from the other. Most protocols use a special 8- bit pattern flag 01111110 as the delimiter to define the beginning and the end of the frame. If the flag pattern appears in the data, we need to somehow inform the receiver that this not the end of the frame. We do this by stuffing 1 single bit (instead of 1 byte) to prevent the pattern from looking like a flag.

The strategy is called bit stuffing. In bit stuffing, if a 0 and five consecutive 1 bits are encountered, an extra 0 is added. This extra stuffed bit is eventually removed from the data by the receiver. Note that the extra bit is added after one 0 followed by five 1s regardless of the value of the next bit. This guarantees that the flag field sequence does not inadvertently appear in the frame.

This means that if the flag like pattern 01111110 appears in the data, it will change to 011111010(stuffed) and is not a mistaken as a flag by the receiver. The real flag 01111110 is not stuffed by the sender and is recognized by the receiver.



HDLC

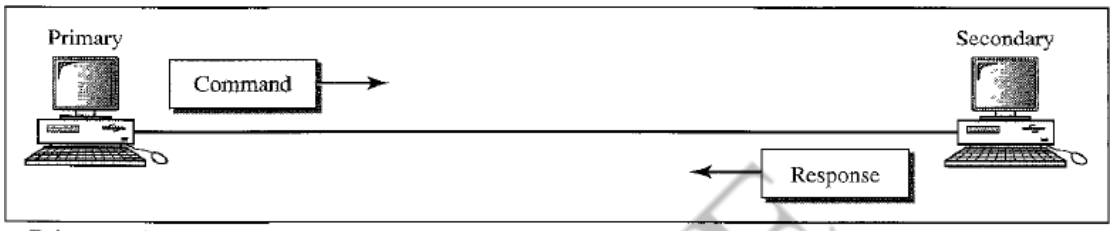
High - Level Data Link Control is a bit-oriented protocol for communicationn over point-to-point and multipoint links.

Configurations and Transfer Modes

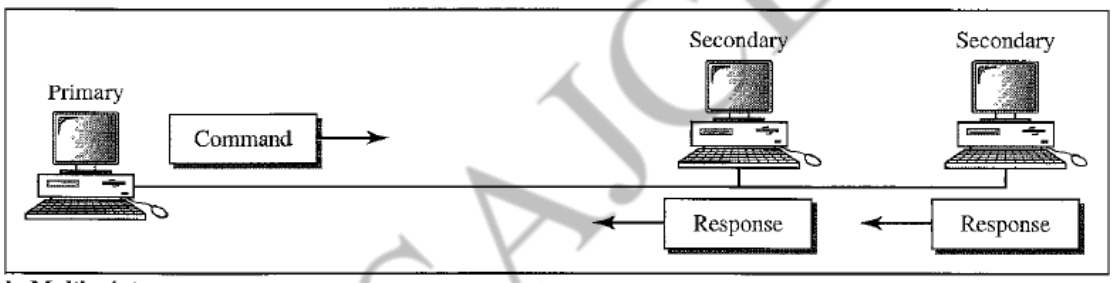
HDLC provides two common transfer modes that can be used in differednt configurations : ***normal responnse mode (NRM) and asynchronous balanced mode (ABM).***

Normal Response Mode

In normal response mode, the station configuration is unbalanced. One primary station and multiple secondary stations. A primary station can send command, a secondary station can only respond. The NRM is used for both point-to -point and multiple - point links.



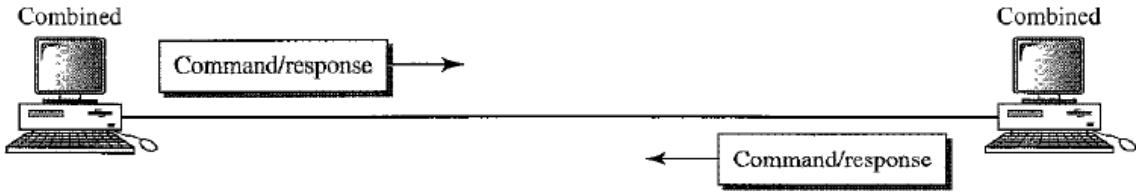
a. Point-to-point



b. Multipoint

Asynchronous Balanced Mode

In asynchronous balanced mode (ABM), the configuration is balanced. The link is point-to-point, and each station can function as a primary and a secondary. This is common mode today.



Frames

HDLC defines three types of frames : information frames (I-frames), supervisory frames (S-frames), and unnumbered frames (U-frames). Each type of frame serves as an envelope for the transmission of a different type of message.

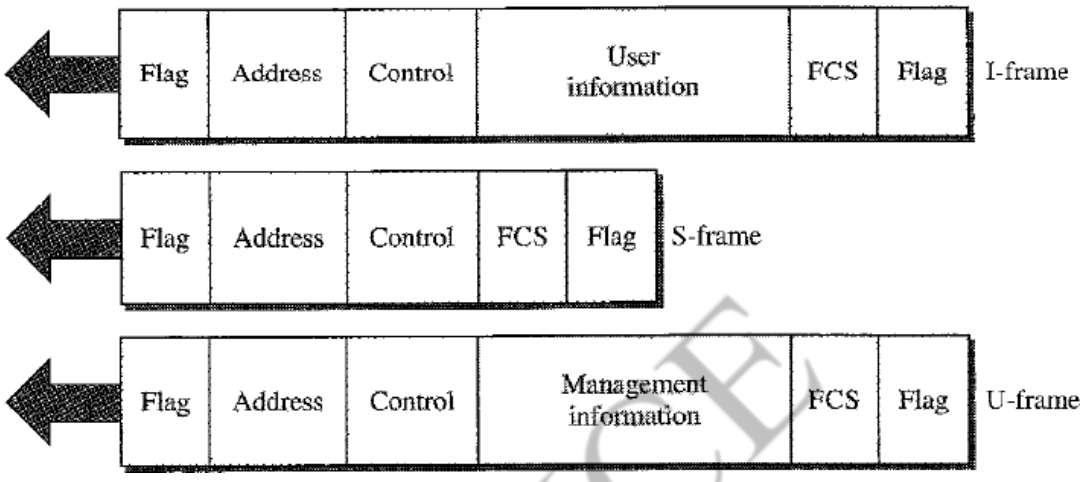
I-frame are used to transport user data and control information relating to user data (piggybacking).

S-frames are used only to transport control information.

U-frames are reserved for system management. Information carried by U-frame is intended for managing the link itself.

Piggybacking is used to improve the efficiency of the bidirectional protocols. When a frame is carrying data from A to B, it can also carry control information about arrived frames from B, when a frame is carrying data from B to A, it can also carry control information about the arrived frames from A.

Frame Format
Each frame in HDLC may contain up to six fields : flag field, address field, control field, information field, frame check sequence (FCS) and ending flag field.



Fields

Flag field of HDLC frame is an 8-bit sequence with the bit pattern 01111110 that identifies both the beginning and the end of a frame and serves as a synchronization pattern for the receiver.

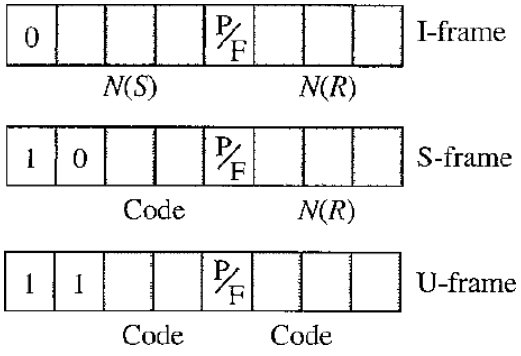
Address field contains the address of the secondary station. It can be 1 byte or several bytes long, depending on the needs of the network. One byte can identify up to 128 stations.

Control field is a 1 or 2 byte segment of the frame used for flow and error control. The interpretation of bits in this field depends on the frame type.

Information field contains the users data from the network layer or management information. Its length can vary from one network to another.

FCS field - Frame check sequence is the HDLC error detection field. It can contain either a 2 or 4 byte ITU-T CRC.

Control Field
Control field determines the type of frame and defines its functionality



First bit defines the frame type

0 - I-frame, 10 - S-frame 11 - U-frame

Control field for I-frames

The next 3 bits, called N(S) define the sequence number of the frame. 3 bit can define Sequence number between 0 to 7.

The last 3 bits, called N(R), correspond to the acknowledgment number when piggybacking is used.

Single bit between N(S) and N(R) is called the P/F bit. When it is set to 1, it means poll or final. Poll means when the frame is sent by a primary station to a secondary. Final means when the frame is sent by a secondary to a primary.

Control field for S-frames

The last 3 bits called N(R) used for acknowledgment number (ACK) or negative acknowledgement number (NAK), The 2 bit called code is used to define the type of S-frame.

Receive ready (RR) - 00 define that frame acknowledges the receipt of a safe and sound frame or group of frames.

Receive not ready (RNR) - 10 define that acknowledges the receipt of frame and it announces that the receiver is busy and cannot receive more frames.

Reject (REJ) – 01 used *in Go-Back-N* to improve efficiency, before the sender time expires**,** the last frame is lost or damaged is informed to sender.

Selective reject (SREJ) – 11 used in Selective Repeat**.** the last frame is lost or damaged is informed to sender.

Control field for U-frames

Unnumbered frames are used to exchange session management and control information between connected devices. 2 bit prefix before P/F and suffix 3 bit, together 5 bit can be used to create up to 32 different types of U-frames.

Ex: 00 001 - SNRM - set normal response mode
11 011 - SNRME - set normal response mode, extended

POINT-TO-POINT PROTOCOL

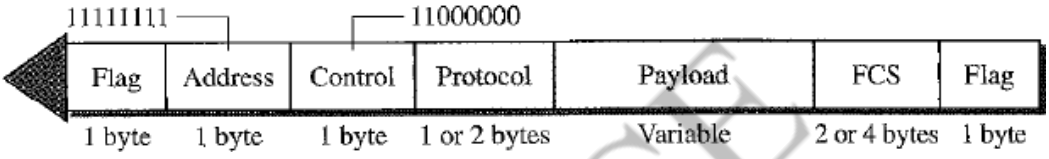
Today, millions of Internet users who need to connect their home computers to the server of a Internet service provider use PPP. The majority of these users have a traditional modem. They are connected to the Internet through a telephone line, which provides the services of the physical layer. But to control and manage the transfer of data, there is a need for a point-to-point protocol at the data link layer.

PPP provides several services:

1. PPP defines the format of the frame to be exchanged between devices.
2. PPP defines how network layer data are encapsulated in the data link frame
3. PPP defines how two devices can authenticate each other.
4. PPP provides connections over multiple links.
5. PPP provides network address configuration.

Framing

PPP is a byte-oriented protocol.



Flag. PPP frame starts and ends with a 1 byte flag with 01111110 pattern

Address. It is the constant values and set to 11111111 (broadcast).

Control This field is set to constant value 11000000. PPP does not provide any flow control. Error control is also limited to error detection. This means that this field is not needed at all.

Protocol This field defines what is being carried in the data field : either user data or other information.

Payload field This field carries either the user data or other information. The data field is a sequence of bytes with the default of a maximum of 1500 bytes.

FCS The frame check sequence is simply a 2 byte or 4 byte standard CRC.

Transition Phases

Dead- In the dead phase the link is not being used. There is no active carrier and the line is quiet.

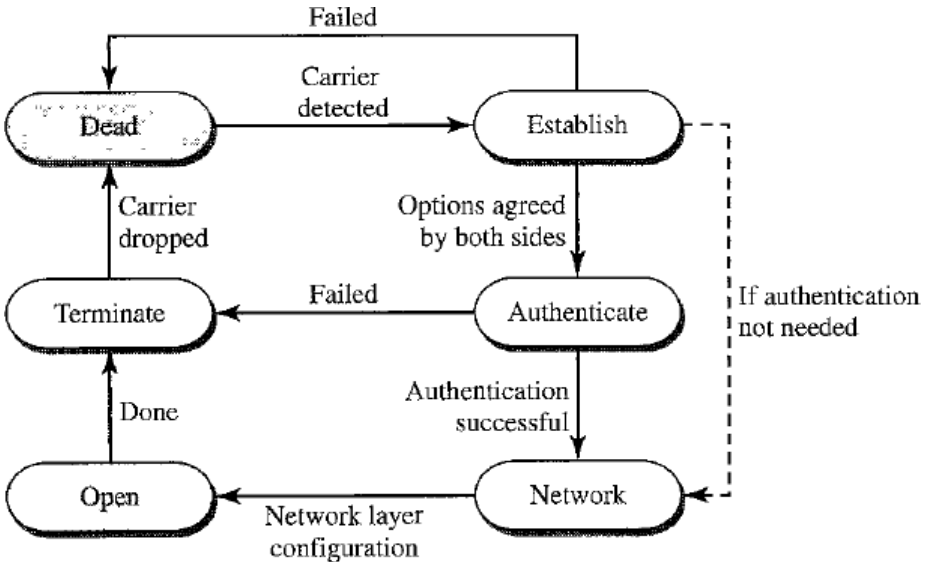
Establish. When one of the nodes starts the communication, the connection goes into this phase. The system goes to the authentication phase or directly to the networking phase.

Authenticate The authentication phase is optional, the two nodes may decide, during the establishment phase, not to skip this phase. They send several authentication packets, if the result is successful the connection goes to the networking phase, otherwise it goes to the termination phase.

Network PPP specifies that two nodes establish a network layer agreement before data at the network layer can be exchanged. The reason is that PPP supports multiple protocols at the network layer.

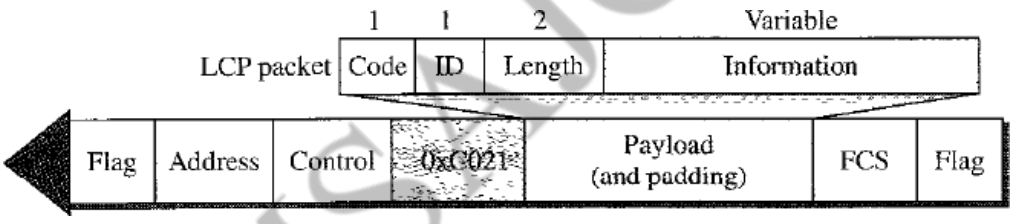
Open In the open phase, data transfer takes place. When a connection reaches this phase, the exchange of data packets can be started. The connection remains in this phase until one of the endpoints wants to terminate the connection.

Terminate In the termination phase the connection is terminated. Several packets are exchanged between the two ends and closing the link.



Link Control Protocol (LCP) is responsible for establishing, maintaining, configuring, and terminating links. Both endpoints of the link must reach an agreement about the options before the link can be established.

All LCP packets are carried in the payload field of the PPP frame with the protocol field set to C021 in hexadecimal. The code field defines the type of LCP packet. There are 11 types of packets.

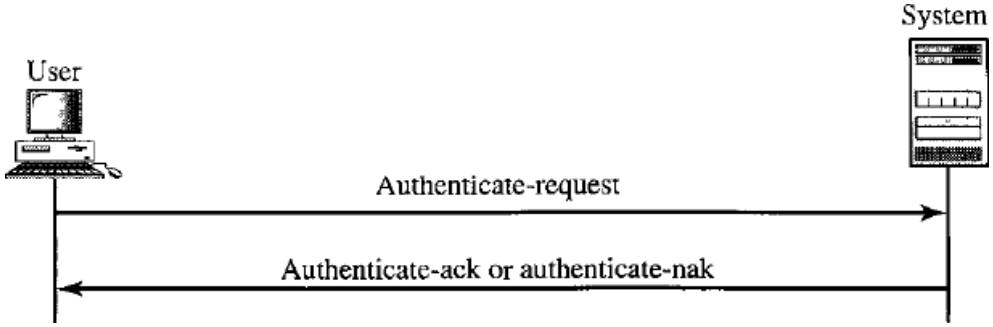


Authentication Protocols

Authentication plays a very important role in PPP because PPP is designed for use over dial-up links where verification of user identity is necessary

Password Authentication Protocol (PAP) is a simple authentication procedure

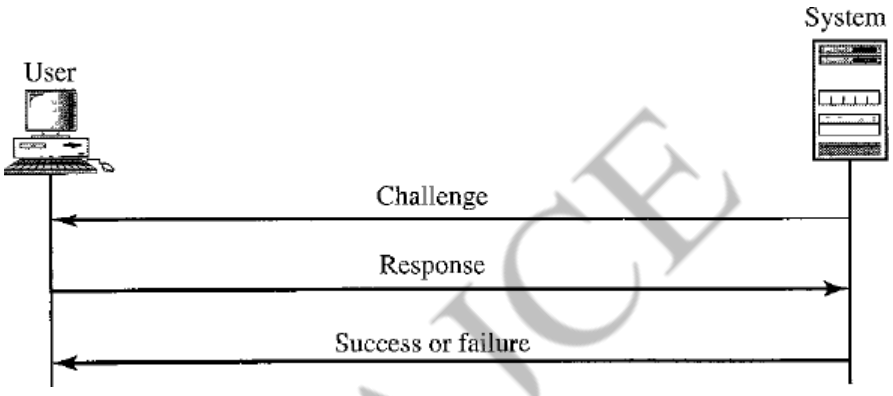
1. The user who wants to access a system sends an authentication identification – username and password.
2. The system checks the validity of the identification and password and either accepts or denies connection.



Challenge Handshake Authentication Protocol (CHAP) is a three-way hand-shaking authentication protocol that provides greater security than PAP. In this method, the password is kept secret, it is never sent online.

1. The system sends the user a challenge packet containing a challenge value, usually a few bytes.
2. The user applies a predefined function that takes the challenge value and the user's own password and creates a result. The user sends the result in the response packet to the system.
3. The system does the same. It applies the same function to the password of the user and the challenge value to create a result. If the result created is the same as the result sent in the response packet, access is granted. Otherwise it is denied.

CHAP is more secure that PAP, especially if the system continuously changes the challenge value. Even if the intruder learns the challenge value and the result, the password is still secret.



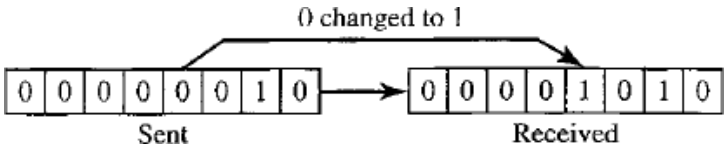
ERROR DETECTION

Types of Errors

Whenever bits flow from one point to another, they are subject to unpredictable changes because of interference. This interference can change the bit value in the signal.

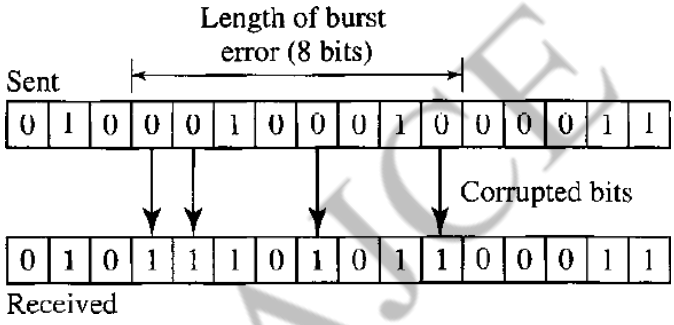
Single-Bit Error

In a single-bit error, a 0 is changed to 1 or 1 to 0.



Bust Error

In a burst error, 2 or more bits in the data unit have changed from 1 to 0 or from 0 to 1. Burst error is more chance to occur than a single-bit error. The duration of noise is normally longer than the duration of 1 bit, which means that when noise affects data, it affects a set of bits. The number of bits affected depends on the data rate and duration of noise.



Error can be detected in two methods: ***Block coding and convolution coding.***

Modul0-2 Arithmetic

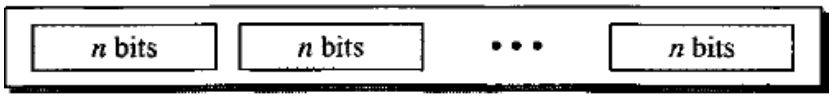
It is encoding technique used in block coding. In this arithmetic, the modulus N is 2. We can use only 0 and 1. Following shows the add or subtract 2 bits.

Adding :	0 + 0 = 0	0 + 1 = 1	1 + 0 = 1	1 + 1 = 0
Subtracting :	0 + 0 = 0	0 + 1 = 1	1 + 0 = 1	1 + 1 = 0

Notice particularly that addition and subtraction give the same results. In this arithmetic we use the XOR operation for both addition and subtraction. The result of an XOR operation is 0 if two bits are the same, the result is 1 if two bits are different.

BLOCK CODING (ERROR DETECTION)

2^k Datawords, each of k bits



2^n Codewords, each of n bits (only 2^k of them are valid)

In block coding, we divide our message into blocks, each of k bits called **datawords**. We add r redundant bits to each block to make the length n=k+r. The resulting n bit blocks are called **codewords**. With k bits, we can create a combination of 2k datawords, with n bits we can create a combination of 2n codewords.

Example: 1

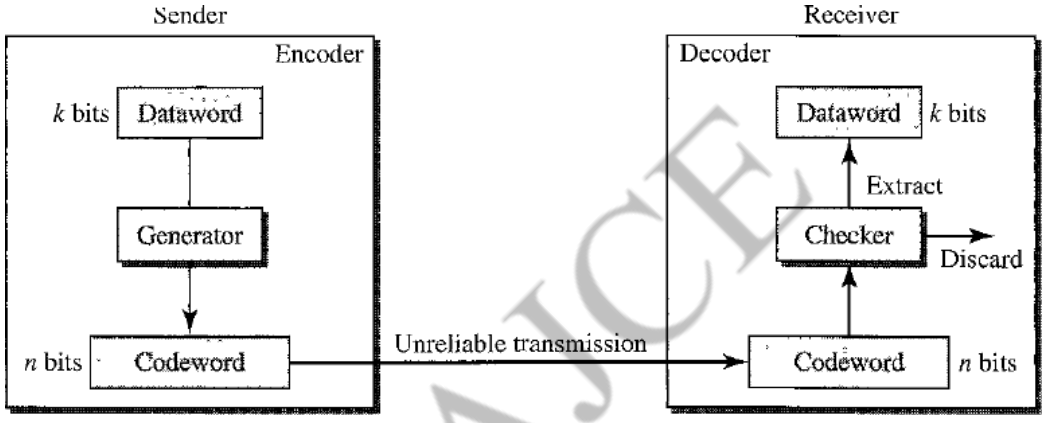
The 4B/5B block coding, k = 4 and n = 5. So we have 2k = 16 datawords and 2n = 32 codewords, 16 out of 32 codewords are used for message transfer and the rest are either used for other purposes or unused.

Error Detection

If the following two conditions are met, the receiver can detect a change in the original codeword.

1. The receiver has a list of valid codewords
2. The original codeword has changed to an invalid one.

Process of error detection in block coding



The sender creates codewords out of datawords by using a generator that applies the rules and procedures of encoding. Each codeword sent to the receiver may change during transmission. If the received codeword is the same as one of the valid codewords, the word is accepted, the corresponding dataword is extracted for use. If the received codeword is not valid, it is discarded. However, if the codeword is corrupted during transmission but the received word still matches a valid codeword, the error remains undetected. This type of coding can detect only single errors. Two or more errors may remain undetected.

Example: 2

Let us assume that k =2 and n =3, following show the list of datawords and codewords

Datawords	Codewords
00	000
01	011
10	101
11	110

Assume the sender encodes the dataword 01 as 011 and sends it to the receiver. Consider the following cases:

1. The receiver receives 011. It is a valid codeword. The receiver extracts the dataword 01 from it.
2. The codeword is corrupted during transmission, and 111 is received (the leftmost bit is corrupted). This is not a valid codeword and is discarded.
3. The codeword is corrupted during transmission, and 000 is received (the right two bits are corrupted). This is a valid codeword. The receiver incorrectly extracts the dataword 00. Two corrupted bits have made the error undetectable.

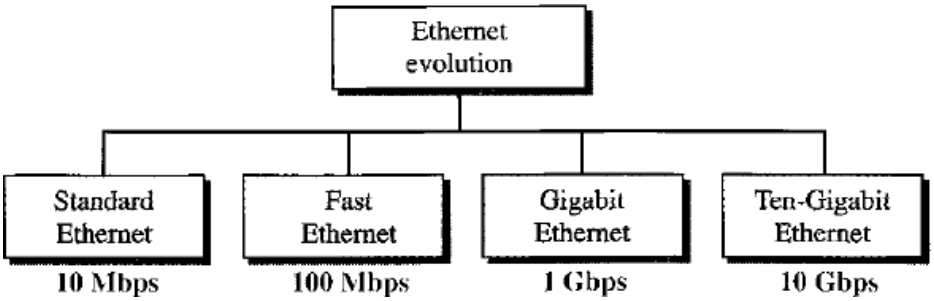
UNIT - 2

STANDARD ETHERNET (802.3)

Standard Ethernet (10 Mbps), fast Ethernet (100Mbps), gigabit Ethernet(1gbps), and Ten-Gigabit Ethernet(10Gbps).

Ethernet evolution

IEE Project 802 has created a sub layer called media access control that defines the specific access method for each LAN. For example it defines CSMA/CD as the media for Ethernet LANs and the token passing method for Token Ring and Token Bus LANs.



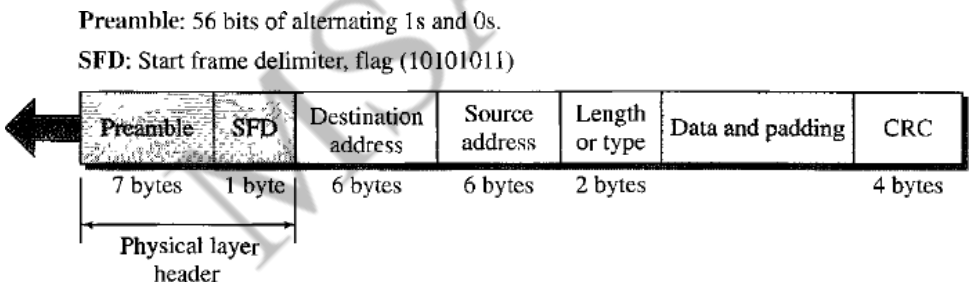
MAC SUB LAYER

In standard Ethernet, the MAC sub layer governs the operation of the access method. It also frames data received from the upper layer and passes them to the physical layer.

Frame Format

The Ethernet frame contains seven fields: permeable, SFD, DA, SA, Length or type of protocols data unit (PDU), upper- layer data, and the CRC. Ethernet does not provide any mechanism for acknowledging received frames

.



Preamble: The first field of the 80.23 frame contains 7 bytes (56 bits) of alternating 0s and 1s that alerts the receiving system to the coming frame and enables it to synchronize its input timing.

Start frame delimiter (SFD): The second field (1 byte: 10101011) signals the beginning of the frame. The SFD warns the station or stations that this is the last chance for synchronization. The last 2 bits is 11 and alerts the receiver that the next field destination address.

Destination address (DA): The DA field is 6 bytes and contains the physical address of the destination station or stations to receive the packet.

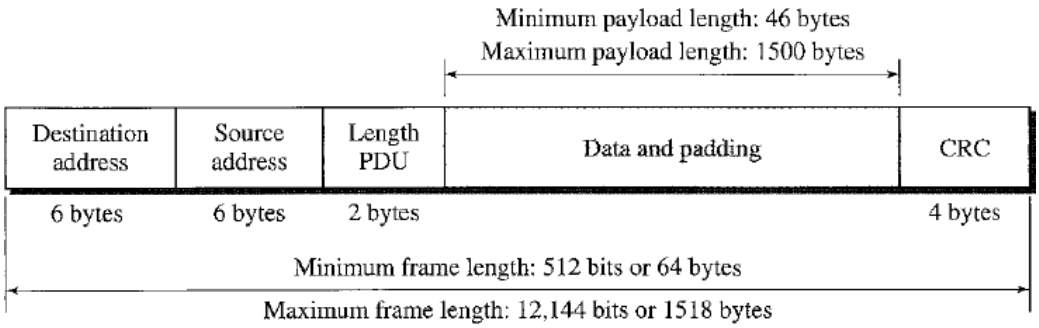
Source address (SA): The SA field is also 6 bytes and contains the physical address of the sender of the packet.

Length or Type: The type field to define the upper-layer protocol using the MAC frame to define the number of bytes in the data field.

Data: This field carries data encapsulated from the upper – layer protocols. It is a minimum of 46 and a maximum of 1500 bytes, as we will see later.

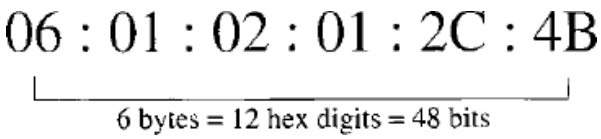
CRC: The last field contains error detection information.

Frame Length Ethernet has imposed restrictions on both the minimum and maximum lengths of a frame.

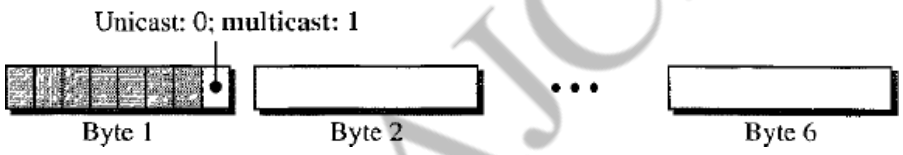


Addressing

Each station on an Ethernet network (such as a PC, workstation, or printer) has its own network interface card (NIC). The NIC fits inside the station and provides the station with a 6 – byte physical address in hexadecimal notation, with a colon between the bytes.



Unicast, Multicast and Broadcast Addresses: A source address is always a unicast address- the frame comes from only one station. The destination address, however, can be unicast, multicast, or broadcast. If the least significant bit of the first byte in a destination address is 0, the address is unicast; otherwise, it is multicast.



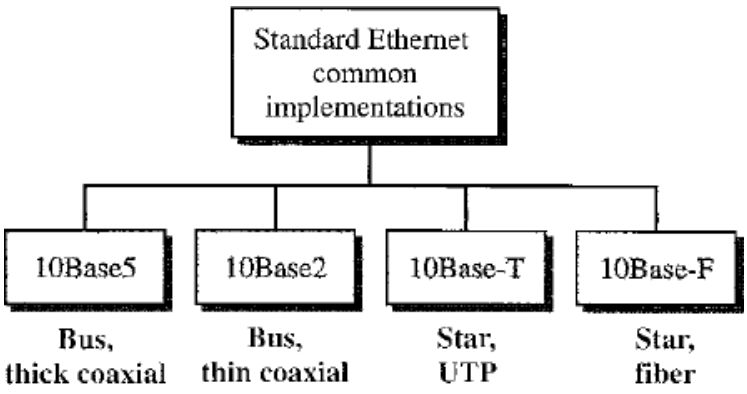
A unicast destination address defines only one recipient; the relationship between the sender and the receiver is one to one. A multicast destination address defines a group of addresses; the relationship between the sender and the receiver is one to many. A broad cast destination address is forty- eight 1s.

PHYSICAL HEADING

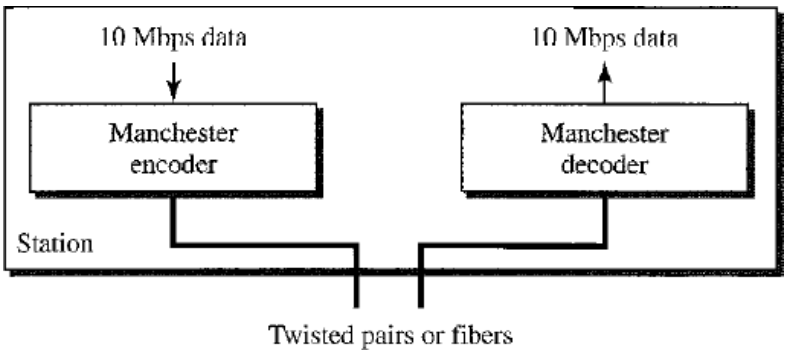
The standard Ethernet defines several physical layer implementations; four of the most common, are shown in Figure 13.8

Encoding and Decoding

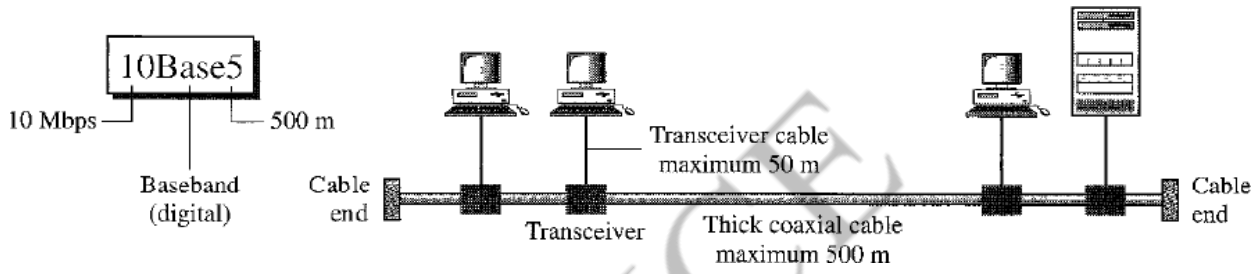
All the standard implementation use digital signalling (baseband) at 10Mbps. At the sender, data are converted to a digital signal using the Manchester scheme; at the receiver, the received signal is interpreted as Manchester and decoded into data.



ENCODING



10Base5: Thick Ethernet

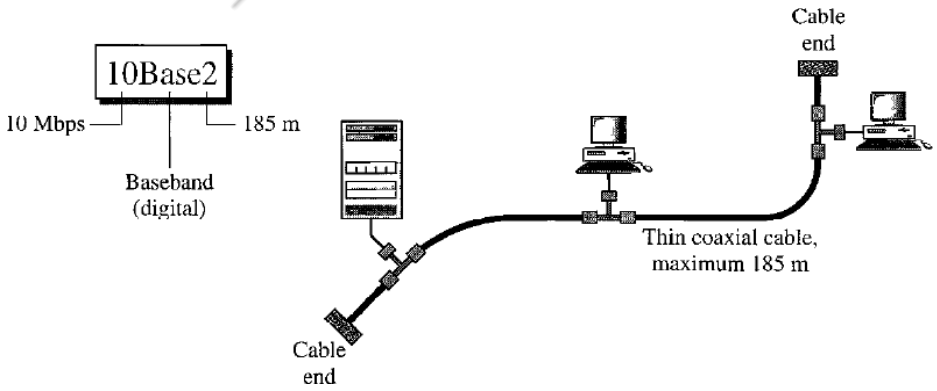


The first implementation is called 10Base5, thick Ethernet, or Thicknet. 10Base5 was the first Ethernet specification to use a busy topology with an external transceiver (transmitter/receiver) connected via a tap to a thick coaxial cable.

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

10Base2: Thin Ethernet

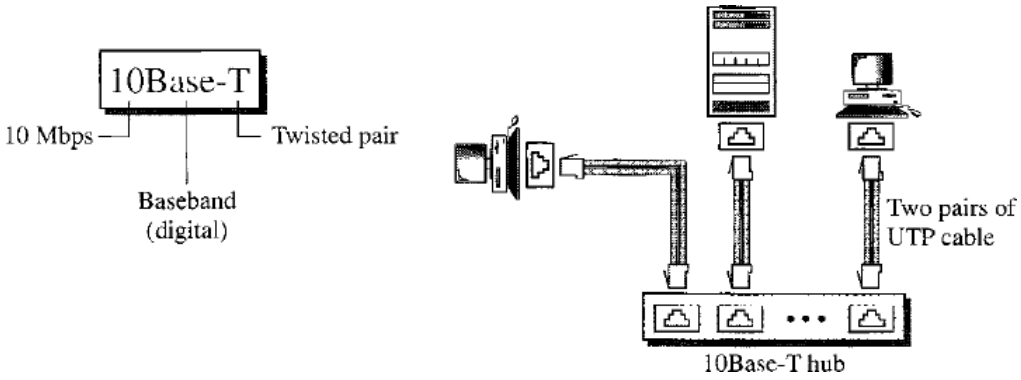
The second implementation is called 10Base2, Ethernet, or Cheapernet. 10Base2 also uses a bus topology, but the cable is much thinner and more flexible. The transceiver is normally part of the network interface card (NIC), which is installed inside the station.



The length of each segment cannot exceed 185m (close to 200m)due to high level of attenuation in thin coaxial cable.

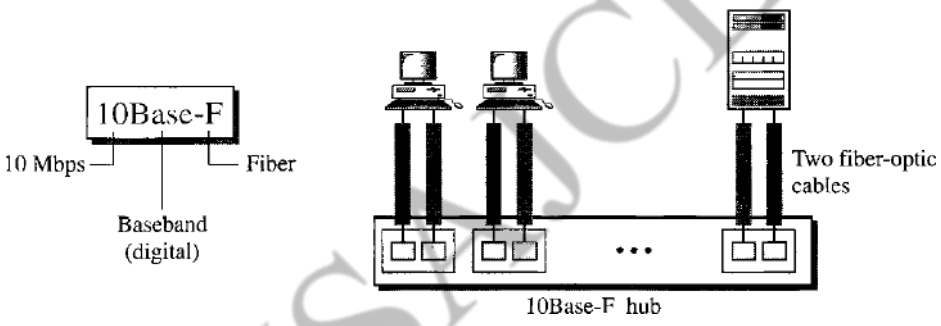
10Base-T: Twisted - pair Ethernet

The third implementation is called 10Base-T or Twisted – pair Ethernet. 10Base-T uses a physical star topology .The stations are connected to a hub via two pairs of twisted cable. The maximum length of the twisted cable here is defined as 100m, to minimize the effect of attenuation in the twisted cable.



10Base-F: Fiber Ethernet

Although there are several types of optical fiber 10-Mbps Ethernet, the most common is called 10Base-F. 10Base-F uses a star topology to connect stations to a hub. The stations are connected to the hub using two fiber-optic cables.



Characteristics	*10Base5*	*10Base2*	*10Base-T*	*10Base-F*
Media	Thick coaxial cable	Thin coaxial cable	2 UTP	2 Fiber
Maximum length	500 m	185 m	100 m	2000 m
Line encoding	Manchester	Manchester	Manchester	Manchester

FAST ETHERNET

Fast Ethernet is backward –compatible with standard Ethernet. The goals of fast Ethernet can be

1. Upgrade the data rate to 100Mbps.
2. Make it compatible with Standard Ethernet.
3. Keep the same 48-bit address.
4. Keep the same frame format.
5. Keep the same minimum and maximum frame lengths.

MAC SUBLAYER

A main consideration in the evolution of Ethernet from 10 to 100Mbps was to keep the MAC sub layer untouched. However, a decision was made to drop the bus topologies and keep only the star topology. For the star topology, there are choices, half duplex. In the half-duplex approach, the stations are connected via a hub; in the full duplex approach, the connection is made via a switch with buffers at each port.

Autonegotiation

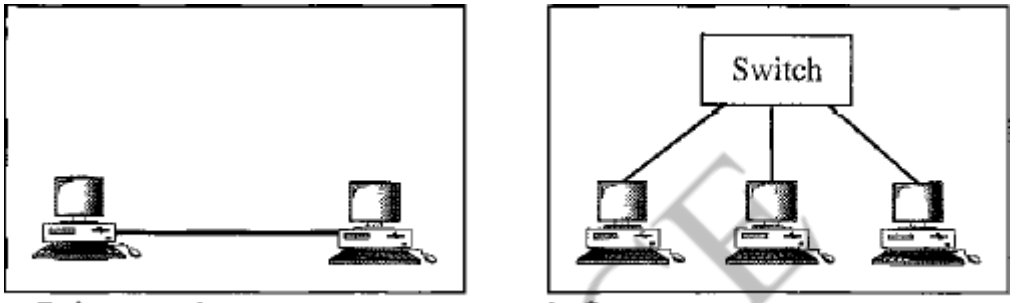
A new feature added to fast Ethernet is called autonegotiation. It allows a station or a hub a range of capabilities. Autonegotiation allows two devices to negotiate the mode or data rate of operation. It was designed particularly for the following purposes:

1. To allow incompatible devices to connect to one another. For example, a device with a maximum capacity of 10Mbps can communicate with a device with a 100Mbps capacity
2. To allow one devices to have multiple capabilities.
3. To allow a station to check a hub's capabilities.

PHYSICAL LAYER

Topology

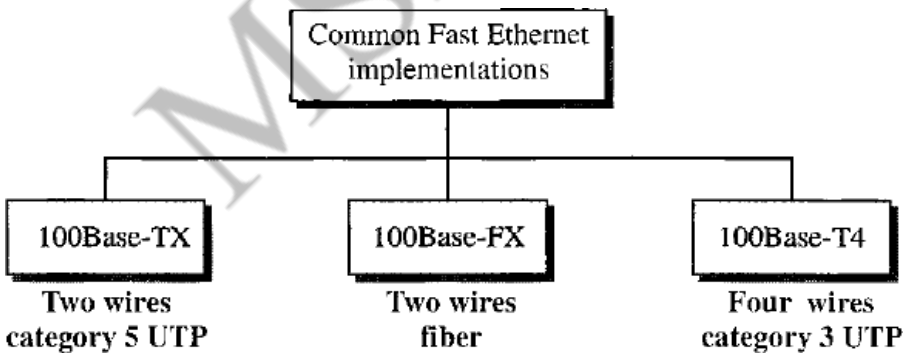
Fast Ethernet is designed to connect two or more stations together. If there are only two stations, they can be connected point to point. Three or more stations need to be connected in a star topology with a hub or a switch at the centre.



a. Point-to-point b. Star

Implementation

Fast Ethernet implementation at the physical layer can be categorized as either two-wire or four –wire. The two wire implementation can be either category 5 UTP (100Base-TX) or fiber –optic cable (100Base-FX). The four –wire implementation is designed only for category 3 UTP (100Base-T4).



Encoding

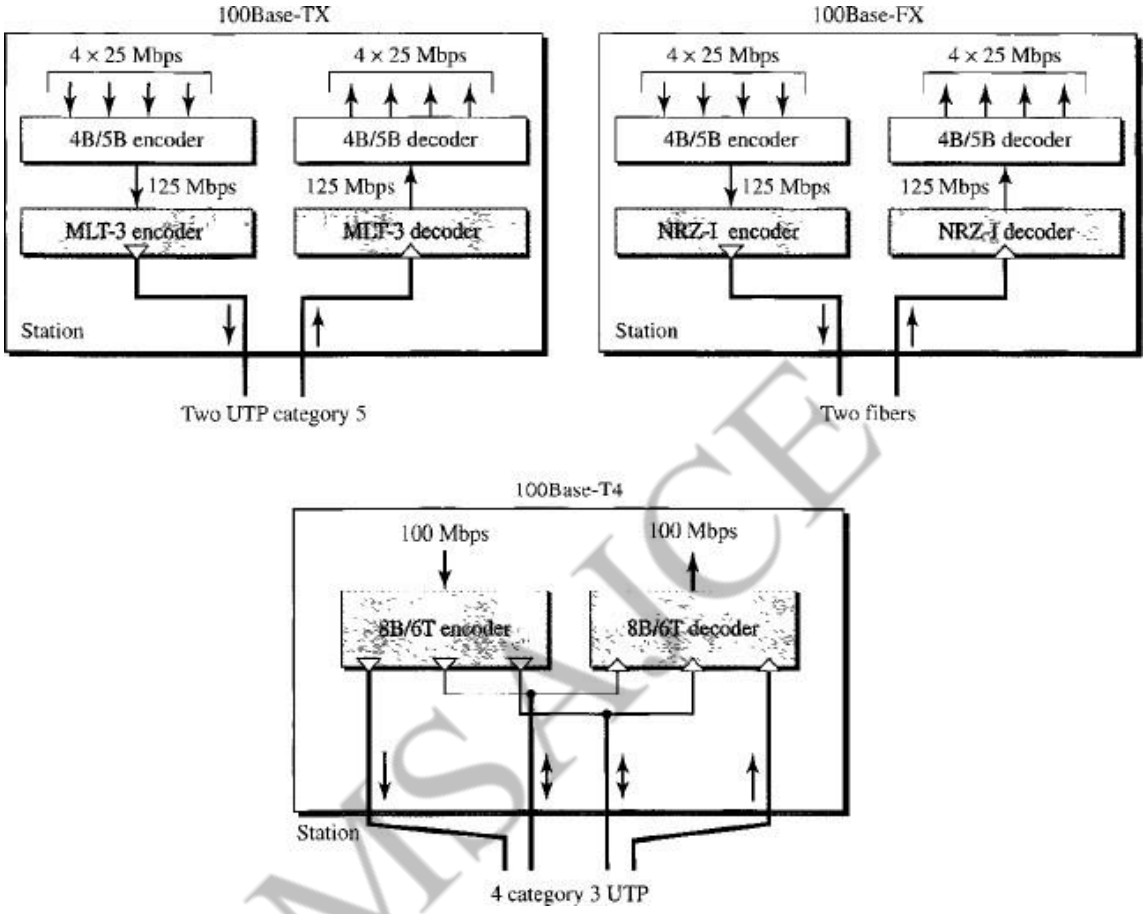
Manchester encoding needs a 200 –Mbaud bandwidth for a data rate of 100Mbps, which makes it unsuitable for a medium such as twisted-pair cable. For this reason, the fast Ethernet designers sought some alternative encoding/decoding scheme. However, it was found that one scheme would not perform equally well for all three implementations. Therefore, three different encoding schemes.

100Base-TX uses two pairs of twisted – pair cable 4B/5B block coding is used to provide bit synchronization by preventing the occurrence of a long sequence of 0s and 1s.This create a data rate of 125Mbps, which is fed into MLT-3 for encoding.

100Base-FX uses two pairs of fiber-optic cables. Optical fiber can easily handle high bandwidth requirements by using simple encoding schemes. The designers of 100Base-FX

selected the NRZ-I encoding scheme for this implementation. The designers used 4B/5B block encoding as we described for 100base –TX. The block encoding increases the bit rate from 100 to 125 Mbps, which can easily be handled by fiber-optical cable.

100Base-T4 uses four pairs of UTP for transmitting 100 Mbps. Encoding/decoding in 100Base-T4 is more complicated, each twisted – pair cannot easily handle more than 25Mbaud. One pair switches between sending and receiving. Three pairs of UTP category 3, however, can handle only 75 Mbaud signal. In 8B/6T, eight data elements are encoded as six signal elements. This means that 100 Mbps uses only (6/8) x 100Mbps, or 75 Mbaud.



GIGABIT ETHERNET

The need for an even higher data rate resulted in the design of the Gigabit Ethernet protocol (1000 Mbps). The goals of the Gigabit Ethernet design can be summarized as follows:

1. Upgrade the data rate to 1 Gbps.
2. Make it compatible with Standard or Fast Ethernet.
3. Use the same 48-bit address.
4. Use the same frame format.
5. Keep the same minimum frame lengths.
6. To support autonegotiation as defined in Fast Ethernet.

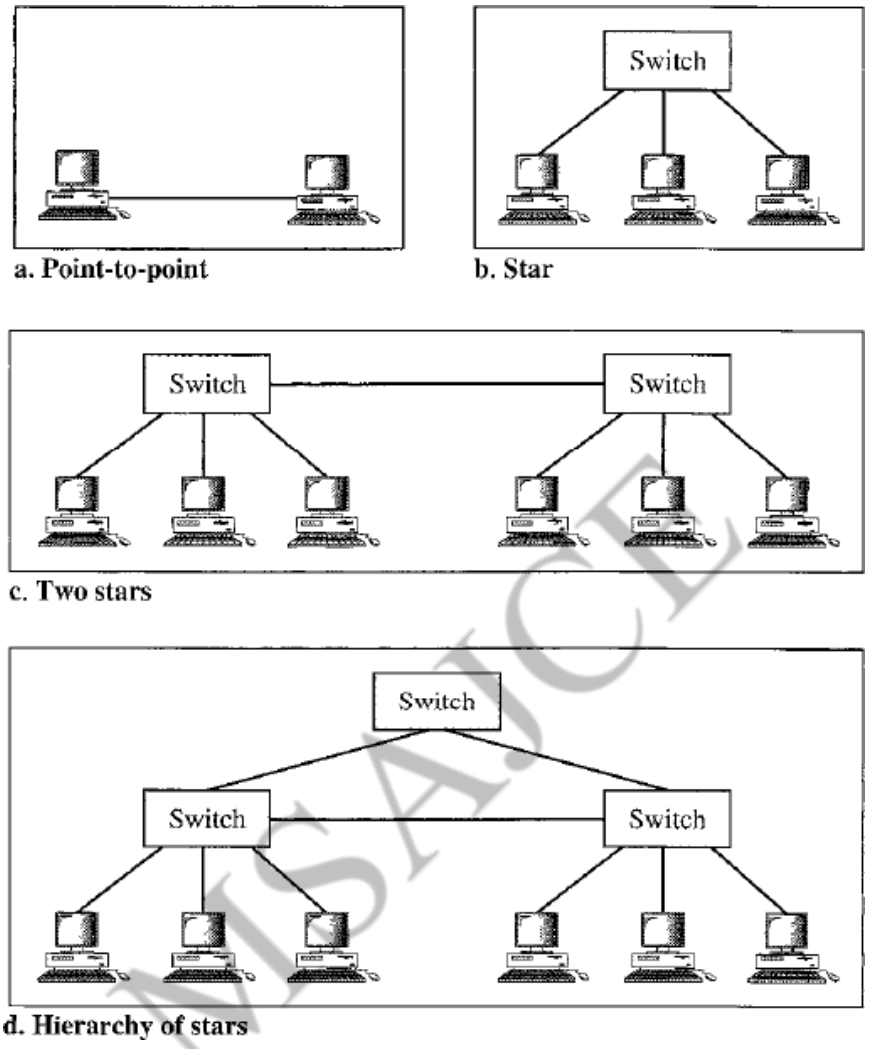
MAC Sub layer

A main consideration in the evolution of Ethernet was to keep the MAC sublayer untouched. However, to achieve a data rate 1 Gbps , this was no longer possible .Gigabit Ethernet has two distinctive approaches for medium access: half – duplex and full-duplex .

Physical layer

Topology

Gigabit Ethernet is designed to connect two or more stations. If there are only two stations, they can be connected point to point. Three or more stations need to be connected in a star topology with a hub or a switch at the center. Another possible configuration is to connect several star topologies or let a star topology be part of another.



a. Point-to-point

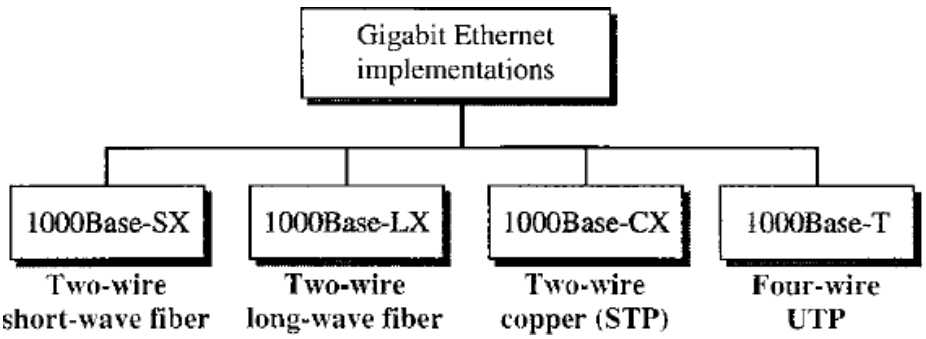
b. Star

c. Two stars

d. Hierarchy of stars

Implementation

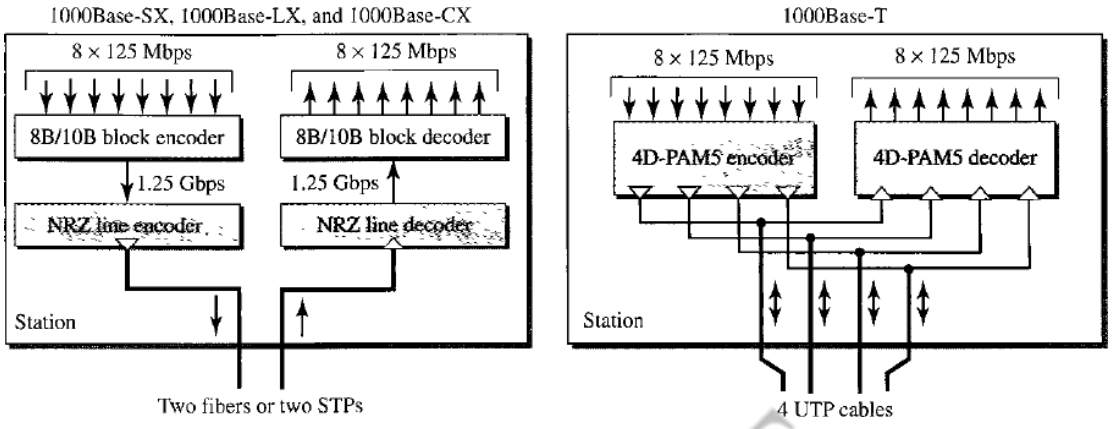
Gigabit Ethernet can be categorized as two wire 1000Base-SX, short –wave, or 1000Base-LX, long –wave , STP(1000Base- CX). The four wire 1000Base-T



Gigabit Ethernet cannot use the Manchester encoding scheme because it involves a very high bandwidth (2GBaud).The two –wire implementations use an NRZ scheme, but NRZ does not self –synchronize properly. To synchronize bits, particularly at this high data rate,8B/10B block encoding.

This block encoding prevents long sequences of 0s or 1s in the stream, but the resulting stream is 1.25Gps. Note that in this implementation, one wire (fiber or STP) is used for sending and one for receiving.

In the four-wire implementation it is not possible for output, because each wire would need to carry 500Mbps, which exceeds the capacity for category 5UTP. As a solution, 4D-PAM5 encoding, is used to reduce the bandwidth. Thus, all four wires are involved in both input and output; each wire carries 250Mbps, which is in the range for category 5 UTP cable.



Characteristics	*1000Base-SX*	*1000Base-LX*	*1000Base-CX*	*1000Base-T*
Media	Fiber short-wave	Fiber long-wave	STP	Cat 5 UTP
Number of wires	2	2	2	4
Maximum length	550 m	5000 m	25 m	100 m
Block encoding	8B/10B	8B/10B	8B/10B	
Line encoding	NRZ	NRZ	NRZ	4D-PAM5

TEN GIGABIT ETHERNET

1. Upgrade the data rate to 10Gbps.
2. Make it compatible with standard, fast, and Gigabit Ethernet.
3. Use the same 48-bit address.
4. Use the same frame format.
5. Keep the same minimum and maximum frame lengths.
6. Allow the interconnection of existing LANs into a metropolitan area network (MAN) or a wide area network (WAN).
7. Make Ethernet compatible with technologies such as Frame Relay and ATM.

MAC Sub layer

Ten-Gigabit Ethernet operates only in full duplex mode which means there is no need for contention;

Physical Layer

The physical layer in Ten-Gigabit Ethernet is designed for using fiber- optic cable over long distances. Three implementations are the most common: 10GBase-S, 10GBase-L, and 10GBase-E.

Characteristics	*10GBase-S*	*10GBase-L*	*10GBase-E*
Media	Short-wave 850-nm multimode	Long-wave 1310-nm single mode	Extended 1550-mm single mode
Maximum length	300 m	10 km	40 km

WIRELESS LANS

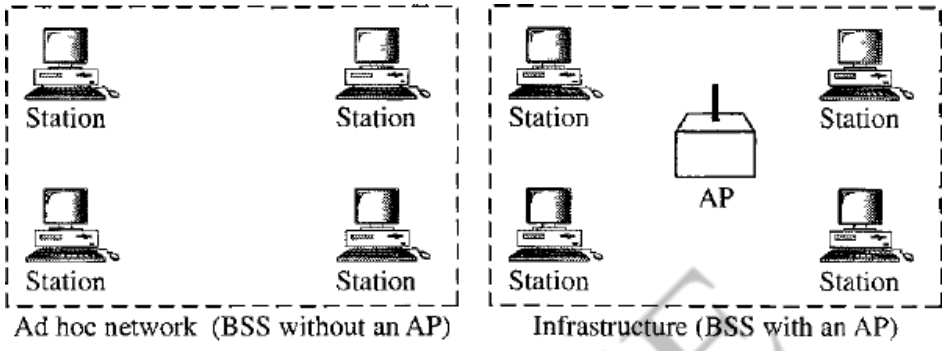
IEEE 802.11

IEEE has defined the specifications for a wireless LAN, called IEEE 802.11, which covers the physical and data link layers.

Architecture

The standard defines two kinds of services: the basic service set (BSS) and the extended service set (ESS).

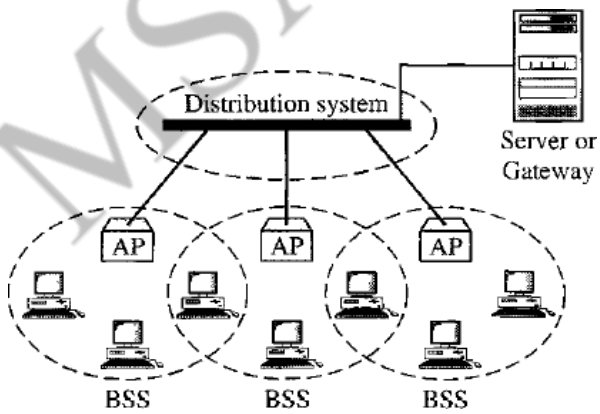
Basic Service Set (BSS)



IEEE 802.1 defines the **basic service set (BSS)** as the building block of a wireless LAN. A basic service set is made of stationary or mobile wireless stations and an optional central base station, known as the **access point (AP)**. Figure 14.1 shows two sets in this standard.

The BSS without an AP is a stand-alone network and cannot send data to other BSSs. It is called an ad hoc architecture. In this architecture, stations can form a network without the need of an AP; they can locate one another and agree to be part of a BSS. A BSS with an AP is sometimes referred to as and infrastructure network.

Extended Service Set (ESS)



An extended service set (ESS) is made up of two or more BSSs with APs. In this case, the BSSs are connected through a distribution system, which is usually a wired LAN. The distribution system connects the APs in the BSSs. The extended service set uses two types of station: mobile and stationary. The mobile stations ate normal stations inside a BSS. The stationary stations are AP stations that are part of a wired LAN.

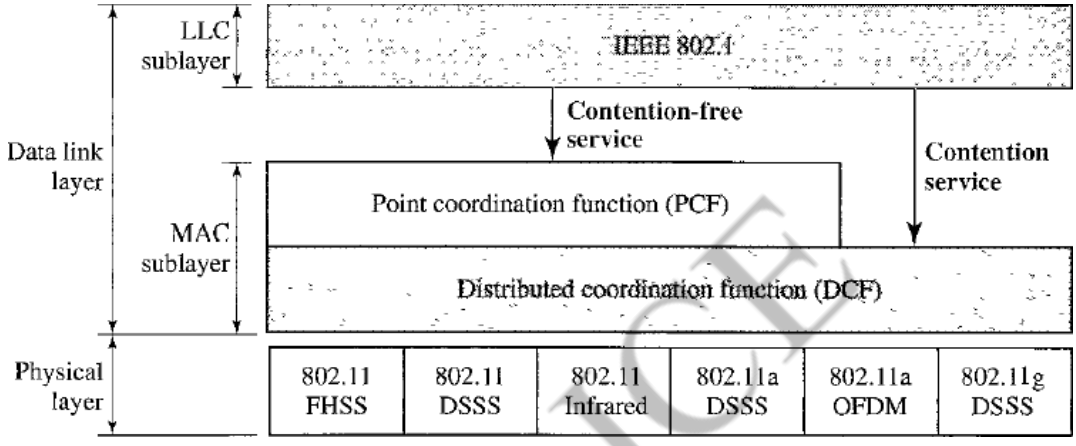
When BSSs are connected, the stations within reach of one another can communicate without the use of an AP. However, communication between two stations in two different BSSs usually occurs via two APs. The idea is similar to communication is a cellular network if we consider each BSS to be a cell and each a AP to be a base station. Note that a mobile station can belong to more than one BSS at the same time

Station Types

IEEE 802.1 defines three types of stations based on their mobility in a wireless LAN: no-transition, BSS-transition, and ESS-transition mobility. A station with no-transition mobility is either stationary (not moving) or moving only inside a BSS. A station with BSS-transition mobility can move from one BSS to another, but the movement is confined inside one ESS. A station with ESS-transition mobility can move from one ESS to another. However, IEEE 802.11 does not guarantee that communication is continuous during the move.

MAC Sublayer

IEEE 802.11 defines two MAC sublayers: the distributed coordination function (DCF) and point coordination function (PCF), the LLC sublayer



Distributed Coordination Function

DCF uses CSMA/CA as the access method. Wireless LANs cannot implement CSMA/CD for three reasons

1. For collision detection a station must be able to send data and receive collision signals at the same time. This can mean costly stations and increased bandwidth requirements.
2. Collision may not be detected because of the hidden station problem. We will discuss this problem later in the chapter.
3. The distance between stations can be great. Signal fading could prevent a station at one end from hearing a collision at the other end.

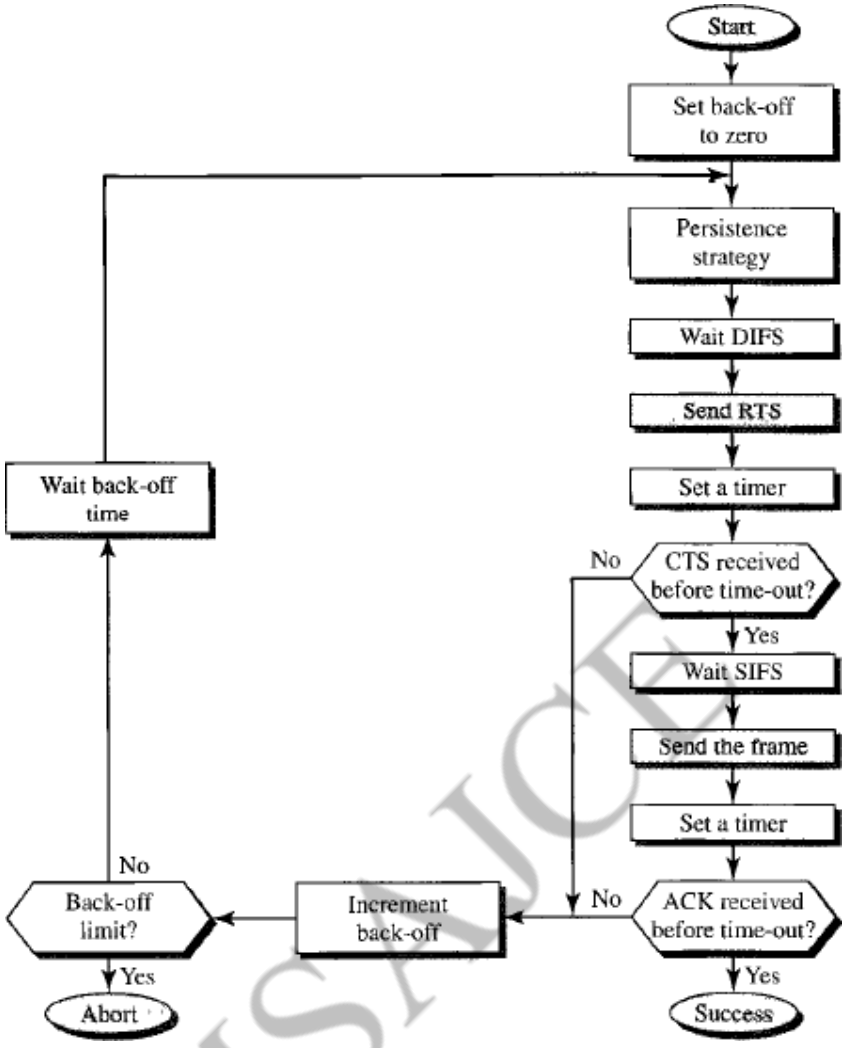
Process Flowchart shows the process flowchart for CSMA/CA as used in wireless LANs.

Frame Exchange Time Line shows the exchange of data and control frames in time.

1. Before sending a frame, the source station senses the medium by checking the energy level at the carrier frequency.
   a. The channel uses a persistence strategy with back-off until the channel is idle.
   b. After the station is found to be idle, the station waits for a period of time called the distributed interframe space (DIFS); then the station sends a control frame called the request to send (RTS).
2. After receiving the RTS and waiting a period of tie called the short interframe space (SIFS), the destination station sends a control frame, called the clear to send (CTS), to the source station. This control frame indicates that the destination station is ready to receive data.

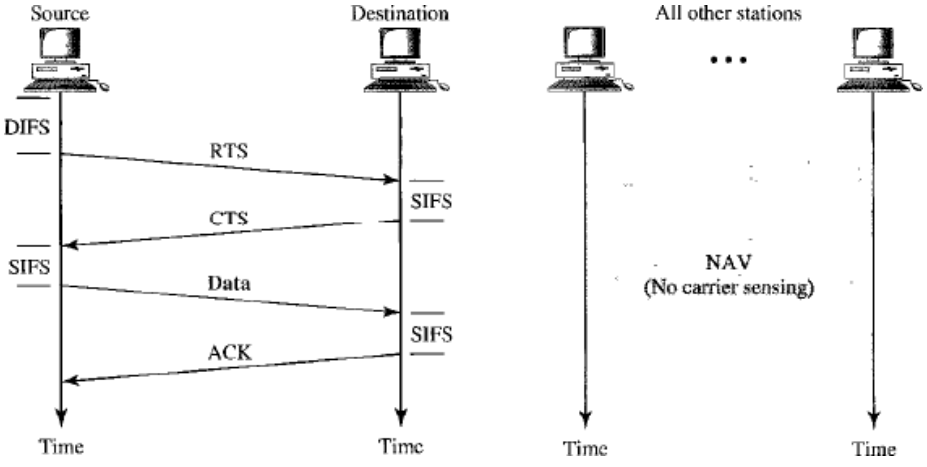
3. The source station sends data after waiting an amount of time equal of SIFS.

4. The destination station, after waiting an amount of time equal to SIFS, sends an acknowledgment to show that the frame has been received. Acknowledgment is needed is this protocol because the station does not have any means to check for the successful arrival of its data at the destination. On the other hand, the lack of collision in CSMA/CD is a kind of indication to the source that data have arrived.



CSMA/CA flow chart

Network Allocation Vector How do other stations defer sending their data if one station acquires access? In other words, how is the collision avoidance aspect of this protocol accomplished? The dey is a feature called NAV.



CSMA/CA and NAV

When a station send an RTS frame, it includes the duration of time that it needs to occupy the channel. The stations that are affected by this transmission create a timer called a network allocation vector (NAV) that show s how much time must pass before these stations are allowed to check the channel for idleness. Each time a station accesses the system and sends an RTS frame, other stations start their NAVA. In other words, each station, before sensing the physical medium to see if it is idle, first checks its NAV to see if it has expired. Figure 14.5 shows the idea of NAV.

Collision During Handshaking

What happens if there is collision during the time when RTS or CTS control frames are in transition, often called the handshaking period? Two or more stations may try to send RTS frames at the same time. These control frames ma collide. However, because there is no mechanism for collision detection, the sender assumes there has been a collision if it has not received a CTS frame from the receiver. The back-off strategy is employed, and the sender tries again.

Point Coordination Function (PCF)

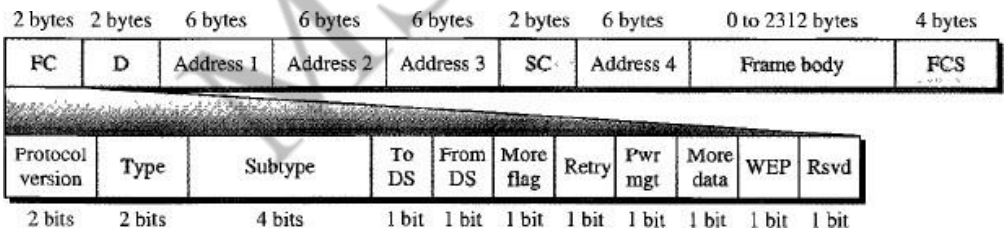
The point coordination function (PCF) is an optional access method that can be implemented in an infrastructure network (not in an ad ho network). It is implemented on top of the DCF and is used mostly for time-sensitive transmission. **PCF** has a centralized, contention-free polling access method. The AP performs polling for stations that are capable of being polled. The stations are polled one after another, sending any data they have to the AP.

To give priority to PCF over DCF, another set of interframe spaces has been defined: PIFS and SIFS. The SIFS is the same as that in DCF, but the PIFS (PCF IFS) is shorter than the DIFS. This means that if, at the same time, a station wants to use only DCF and an AP wants to use only DFS and an AP wants to use PCF, the AP has priority.

Due to the priority of PCF over DCF, stations that only use DCF may not gain access to the medium. To prevent this, a repetition interval has been designed to cover both contention-free (PCF) and contention-based (DCF) traffic. The repetition interval, which is repeated continuously, starts with a special control frame, called a beacon frame. When the stations hear the beacon frame, they start their NAV for the duration of the contention-free period of the repetition interval.

Frame Format

The MAC layer frame consists of nine fields,



Field	*Explanation*
Version	Current version is 0
Type	Type of information: management (00), control (01), or data (10)
Subtype	Subtype of each type (see Table 14.2)
To DS	Defined later
From DS	Defined later
More flag	When set to 1, means more fragments
Retry	When set to 1, means retransmitted frame
Pwr mgt	When set to 1, means station is in power management mode
More data	When set to 1, means station has more data to send
WEP	Wired equivalent privacy (encryption implemented)
Rsvd	Reserved

BLUETOOTH

Bluetooth is a wireless LAN technology designed to connect devices of different functions such as telephones, notebooks, computers, cameras, printers, coffee makers and so on. A Bluetooth is a LAN is an ad hoc network , which means that the network is formed spontaneously; the devices, sometimes called gadgets, find each other and make a network called a piconet.

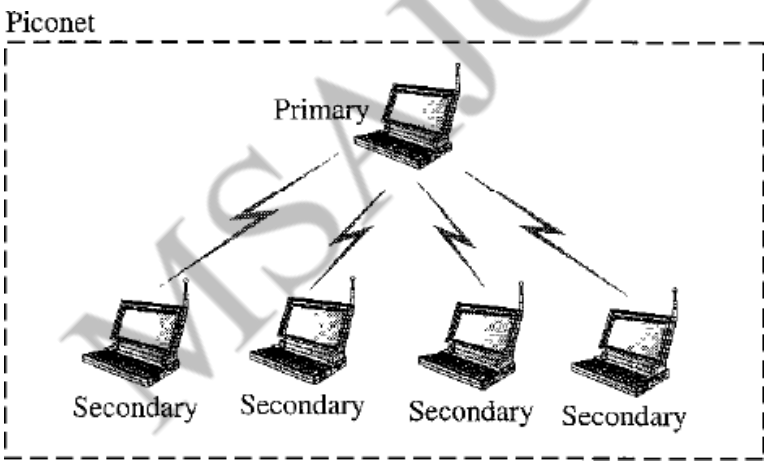
A Bluetooth LAN can even be connected to the internet if one of the gadgets as these capability. A Bluetooth LAN, by nature cannot be large .If there are many gadgets that try to connect, there is chaos.

Bluetooth technology has several applications. Peripheral Devices such as a wireless mouse or keyboard can communicate with the computer through this technology. monitering devices can communicate with sensor devices in a small healthcare center. Home security devices can use this technology to connect different sensors to the main security controller.
Architecture

Bluetooth defines two types of network : **Piconet and Scatternet.**

Piconets

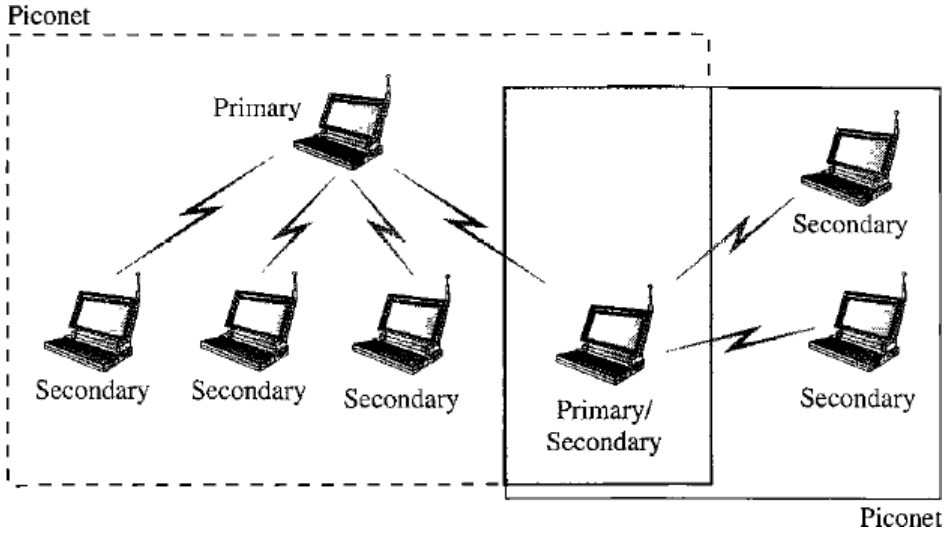
A Bluetooth network s called a Piconet, or a small net. A piconet can have upto eight station, one of which is called the primary; the rest are secondaries.all the secondary stations synchronise their clocks and hopping sequence with the primary.Note that a piconet can have only one primary station. The communication between the primary and the secondary can be one-to-one or one-to-many.



Although a piconet can have a maximum of seven secondaries, an additional eight secondaries can be in the parked state. A secondary in a parked state is synchronized with the primary, but cannot take part in communication until it is moved from the parked state means that an active station must go to the parked state.

Scatternet

Piconets can be combine to form what is called a scatternet. A secondary station in one piconet can be primary in another piconet. The station can receive messages from the primary in the first piconet (as a secondary) and, acting as a primary, deliver them to secondaries in the second piconets. A station can be a member of two piconets.

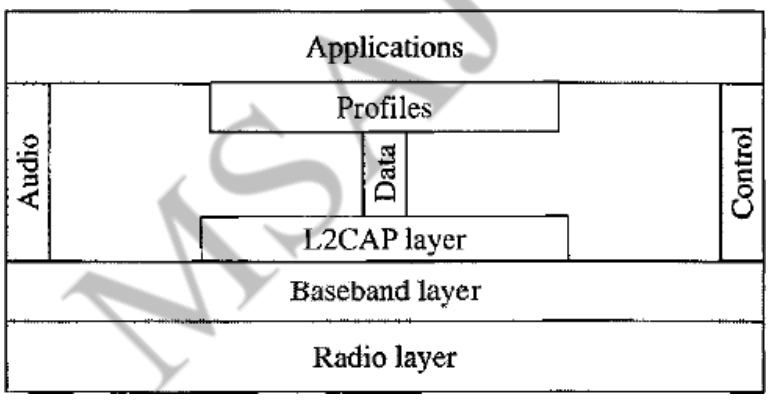


Bluetooth Devices

A Bluetooth Devices has a built-in short-range radio transmitter. The current data rate is 1 Mbps with a 2.4-GHz bandwidth. This means that there is a possibility of interference between the IEEE 802.11b wireless LANs and Bluetooth LANs.

Bluetooth Layers

Bluetooth uses several layers that do not exactly match those of the Internet model we have defined in this book. Fig 14.21 shows these layers.



Radio Layer

The Radio layer is a roughly equivalent to the physical layer of the internet model. Bluetooth devices are low-power and have a range of 10 m.

Band

Bluetooth uses a 2.4-GHz ISM band divided into 79 channels of 1MHz each.

FHSS

Bluetooth uses the frequency-hopping spread spectrum (FHSS) method in the physical layer to avoid interference from other devices or other networks. Bluetooth hops 1600 times per second, which means that each device changes its modulation frequency 1600 times per second. A device uses a frequency for only 625μs (1/1600 s) before it hops to another frequency; the dwell time is 625μs.

Modulation

To transform bits to a signal, Bluetooth uses a sophisticated version of FSK, called GFSK (FSK with Gaussian bandwidth filtering). GFSK has a carrier frequency. Bit 1 is represented by a frequency deviation above the carrier; bit 0 is represented by a frequency deviation below the carrier.

Baseband Layer

The baseband layer is roughly equivalent to the MAC sublayer in LANs. The access method is TMDA. The primary and secondary communicate with each other using time slots. The length of a time slot is exactly the same as the dwell time, 625µs. this means that during the time that one frequency is used, a sender sends a frame to a secondary, or a secondary sends a frame to the primary. Note that the communication is only between the primary and a secondary; secondaries cannot communicate directly with one another.

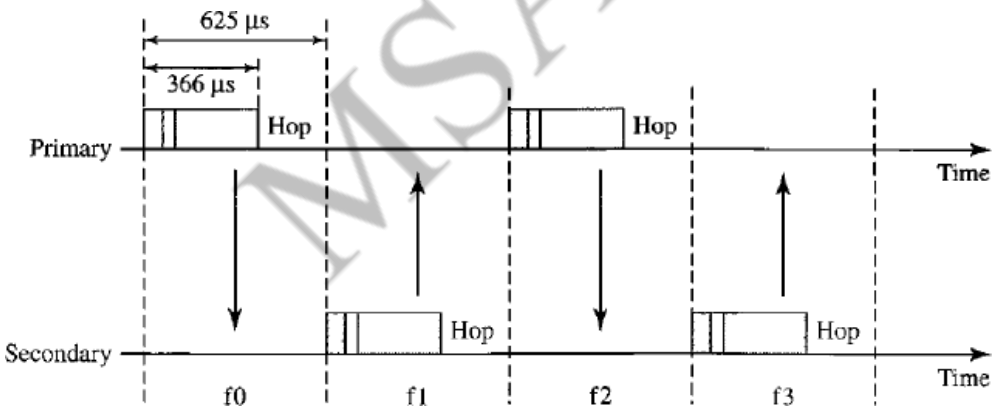
TDMA

A Bluetooth uses a form of TDMA that is called a TDD-TDMA(time division duplex TDMA). TDD-TDMA is a kind of half-duplex communication in which the secondary and receiver send and receive data, but not at the same time (half-duplex); however, the communication for each direction uses different hops .this is similar to walkie-talkies using different carrier frequencies.

Single-secondary communication

If the piconet has only one secondary, the TDMA operation is very simple. The time divided into slots of 625µs. The primary uses even numbered slots (0,2,4..etc); the secondary uses odd numbered slots (1,3,5..etc). TDD-TDMA allows the primary and the secondary to communicate in half-duplex mode.

In slot 0, the primary sends, and the secondary receives; in slot 1, the secondary sends and the primary receives. The cycle is repeated. Figure 14.22 shows the concept.



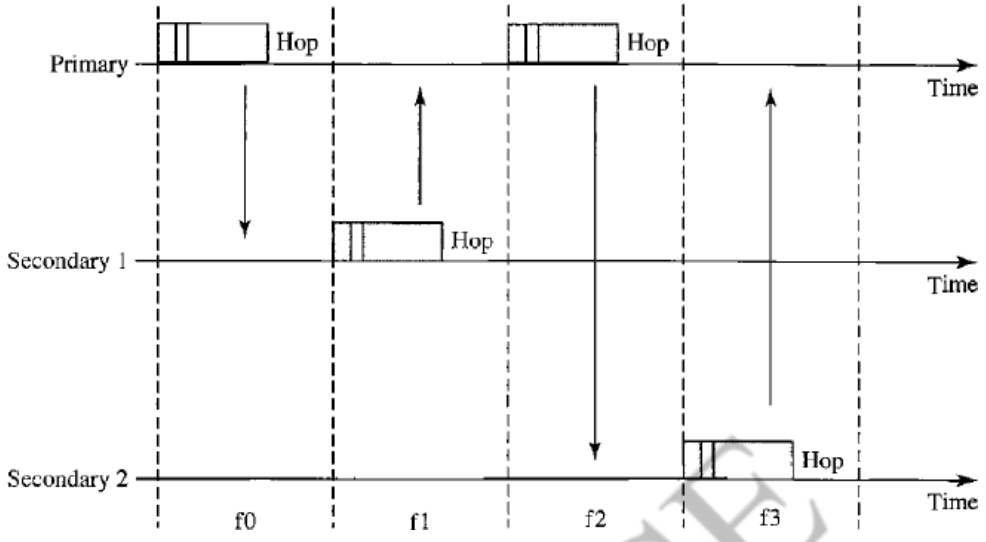
Multiple-Secondray Communication

The process is a little more involved if there is more than one secondary in the piconet. Again, the primary uses the even-numbered slots, but a secondary sends in the next odd-numbered slots. If the packet in the previous slot was addressed to it. All secondaries listen on even-numbered slots, but only one secondary sends in any odd-numbered slots. Fig 14.23 shows scenario.

Let us elaborate on the figure.

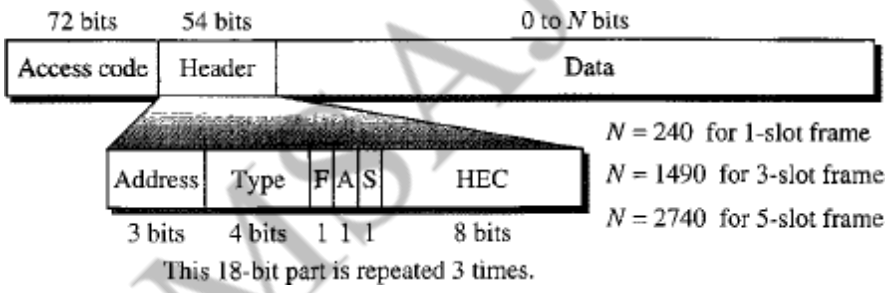
1. In slot 0, the primary sends a frame to Secondary1.
2. In slot 1, only secondary 1 sends a frame to the primary because the previous frame was addressed to secondary 1; other secondaries are silent.

3. In slot 2, the primary sends a frame to secondary 2.
4. In slot 3, only secondary 2 sends a frame to the primary because the previous frame was addressed to secondary 2; other secondaries are silent.
5. The cycle continues.



FRAME FORMAT

Frame in the baseband layer can be one of three types : one-slot, three-slot, five slot.



Three-slot frame

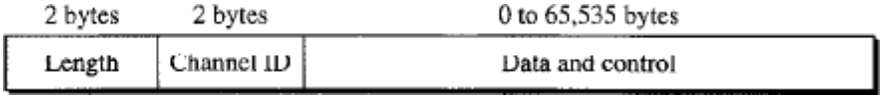
ACCESS CODE : This 72 bit field normally contains synchronization bits and the identifier of the primary to distinguish the frame of one piconet from another.

L2CAP

The Logical Link Control and Adaptation protocol, or L2CAP (L2 here means LL),is roughly equivalent to the LLC sublayer in LANs. It is used for data exchange on an ACL link; SCO channels do not use L2CAP.

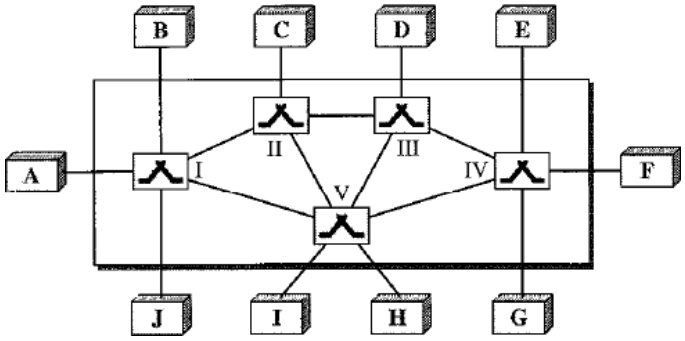
The 16-bit length field defines the size of the data, in bytes, coming from the upper layers. Data can be upto 65, 535 bytes,The channel ID (CID) defines a unique identifier for the virtual channel created at this level.

The L2CAP has specific duties: multiplexing, segmentation and reassembly, quality of service (QoS), and group management.

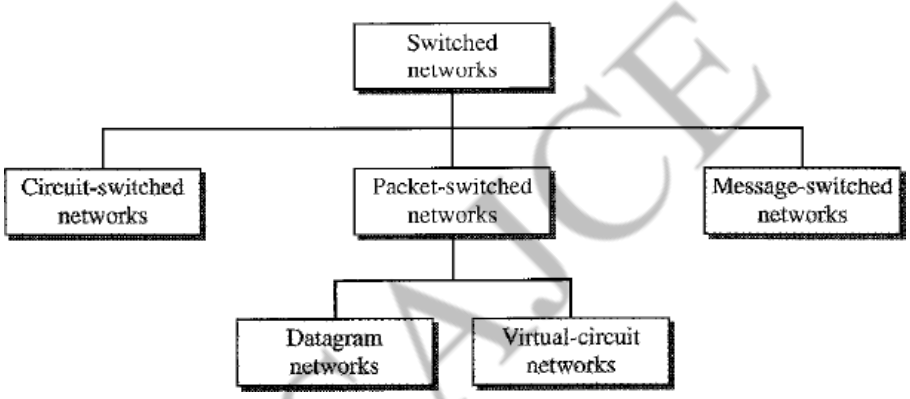


SWITCHED NETWORK

A network is a set of connected devices, Whenever we have multiple devices, we have the problem of how to connect them to make one-to-one communication possible. A better solution is switching. A switched network consists of a series of interlinked nodes called switches.



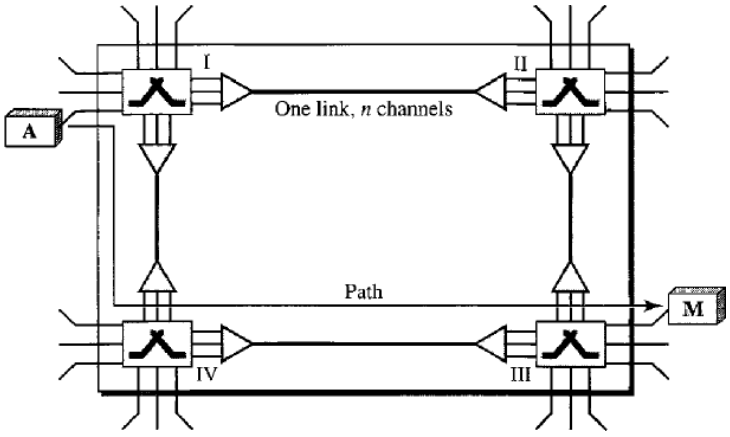
Switched network



Taxonomy of switched networks

CIRCUIT-SWITCHED NETWORKS

A circuit-switched network consist of a set of switches connected by a physical links. A connection between two stations is a dedicated path made of one or more links. However, each connection uses only one dedicated channel on each link. Each link is normally divided into n channels by using FDM or TDM.



A trivial circuit switched network

Three phases

The actual communication in a circuit-switched network requires three phases: connection setup, data transfer, and connection teardown.

Setup phase

Before the two parties (or multiple parties in conference call) can communicate, a dedicated circuit (combination of channels in links) needs to be established. The end systems are normally connected through dedicated lines to the switches, so connection setup means creating dedicated channels between the switches. For example, in figure, when system a needs to connect system M, it sends a setup request that includes the address of system M, to switch I. Switch I finds a channel between itself and switch IV that can be dedicated for this purpose. Switch I then sends the request to switch IV, which finds a dedicated channel between itself and switch III. Switch III informs system M of system A's intension at this time.

In the next step of making a connection, an acknowledgement from system M needs to be sent in the opposite direction of system A. Only after system A receives this acknowledgement is the connection established.

Note that end-to-end addressing is required for correcting a connection between the two end systems. These can be, for example, the addresses of the computers assigned by the administrator in an TDM network, or telephone numbers in an FDM network.

Data Transfer Phase

After the establishment of the dedicated circuits (channels), the two parties can transfer data.

Teardown Phase

When one of the parties need to disconnect, a signal is sent to each switch to release the resources.

PACKET SWITCHED

It has two type Datagram Networks, Virtual Circuit Networks

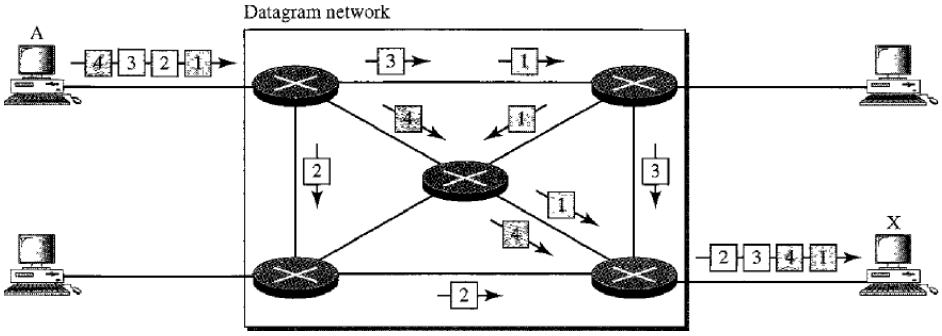
1. DATAGRAM NETWORKS

In data communications, we need to send messages from one end systems to another. If the message is going to pass through a packet-switched network, it needs to be divided into packets of fixed or variable size. The size of the packet is determined by the network and the governing protocol.

In packet switching, there is no resource allocation for a packet. This means that there is no reserved bandwidth on the links, and there is no scheduled processing time for each packet. Resources are allocated on demand. The allocation is done on a first-come, first-served basis. When a switch receives a packet, no matter what is the source or destination, the packet must wait if there are other packets being processed. As with other systems in our daily life, this lack of reservation may create delay. For example, if we do not have a reservation at a restaurant, we might have to wait.

In a datagram network, each packet is treated independently of all others. Even if a packet is part of a multi packet transmission, the network treats it as tough it existed alone. Packets in this approach are referred to as data grams.

Figure shows how the datagram approach is used to deliver four packets from station A to station X. The switches in a datagram network are traditionally referred to as routers. That is why we use a different symbol for the switches in the figure.



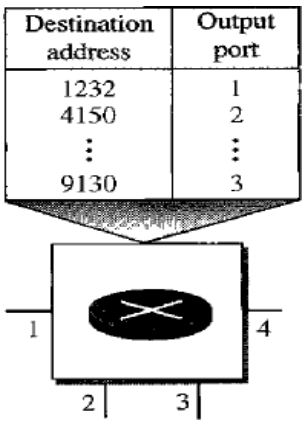
Data gram network with four switches

In this example, all four packets (or datagrams) belong to the same message, but may travel different paths to reach their destination. This is so because the links may be involved in carrying packets from other sources and do not have the necessary bandwidth available to carry all the packets from A to X. This approach can cause the datagrams of a transmission to arrive at their destination out of order with different delays between the packets. Packets may also be lost or dropped because of a lack of resources. In most protocols, it is the responsibility of a upper layer protocol to recorder the datagrams or ask for lost datagrams before passing them on to the application.

The datagram networks are sometimes referred to as connection less networks. The term connection less here means that the switch (packet switch) does not keep information about the connection state. There are no setup or teardown phases. Each packets is treated the same by a switch regard less of its sources or destination.

Routing Table

If there are no setup or teardown phases, how are there packets routed to their destinations in a datagram network? In this type of network, each switch (or packet switch) has a routing table which is based on the destination address. The routing tables are dynamic and are updated periodically. The destination addresses and the corresponding forwarding output ports are recorded in the tables. This is different from the table of a circuit-switched network in which each entry is created when the setup phase is completed and deleted when the teardown phase is over. Figure8.8 shows the routing table for a switch.



Routing table in a datagram network

Destination Address

Every packet in a datagram network carries a header that contains, among other information, the destination address of the packet. When the switch receives the packet, this

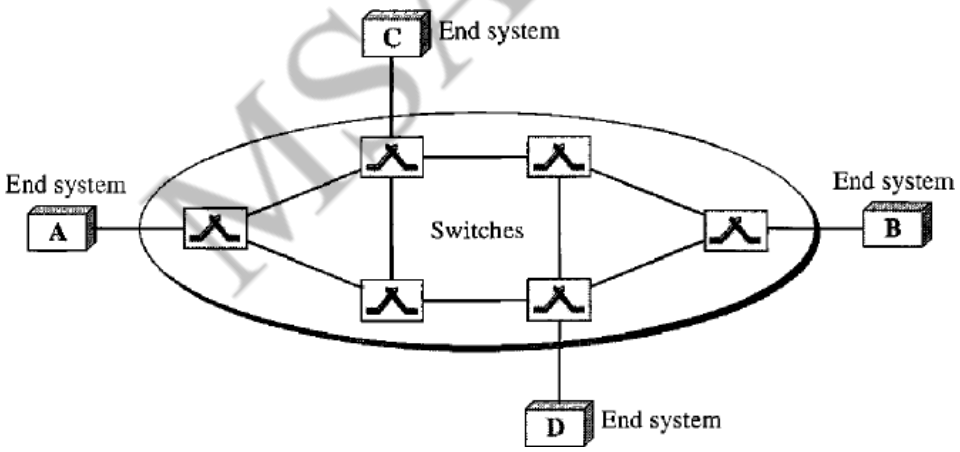
destination address is examined; the routing table is consulted to find the corresponding port through which the packet should be forwarded. This address, unlike the address in a virtual-circuit-switched network, remains the same during the entire journey of the packet.

2. VIRTUAL-CIRCUIT NETWORKS

A Virtual-circuit network is a cross between a circuit-switched network and a datagram network. It has some characteristics of both.

1. As in a circuit-switched network, there are setup and teardown phases in addition to a data transfer phase .
2. Resources can be allocated during the setup phase, as in a circuits-switched network, or on demand, as in a datagram network.
3. As in a datagram network, data are packetized and each packet carries an address in the header. However, the address in the header has local jurisdiction (It defines what should be the next switch and the channel on which the packet is being carried), not end-to-end jurisdiction. The reader may ask how the intermediate switches know where to send the packet if there is no final destination address carried by a packet. The answer will be clear when we discuss virtual-circuit identifiers in the next section.
4. As in a circuits-switched networks, all packets follow the same path established during the connection.
5. A virtual-circuit networks is normally implemented in the data link layer, while a circuit-switched network is implemented in the physical layer and a datagram network in the network layer. But this may change in the figure.

   This figure is an example of a virtual-circuit network. The network has that allow traffic form sources to destinations. A source or destination can be a computer, packets switch, bridge or any other device that connect other networks.



Virtual circuit network

Addressing

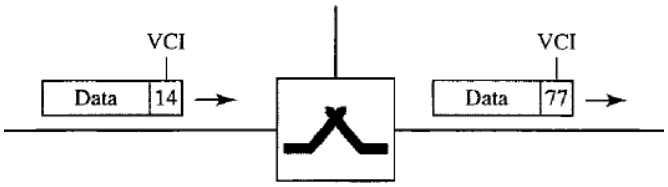
In a virtual-circuit network, two types of addressing are involved: global and local (virtual-circuit identifier).

Global Addressing

A source or a destination leads to have a global address –an address that can be unique in the scope of the network or internationally if the network is part of an international network. However, we will see that a global address in virtual-circuits network is used only to create a virtual-circuit identifier, as discussed next.

Virtual-circuit identifier

The identifier that is actually used for data transfer is called the virtual-circuit identifier (VCI). A VCI, unlike a global address, is a small number that has only switch scope; it is used by a frame between two switches. When it leaves, it has different VCI. Figure8.11 shows how the VCI in a data frame changes from one switch to another. Note that a VCI does not need to be a large number since each switch can use its own unique set of VCIs.



Virtual circuit identifier

Three Phases

Three phases in a virtual-circuit network: setup, data transfer, and teardown

Data Transfer Phase

To transfer a frame from a source to its destination, all switches need to have a table entry for this virtual circuit. The table, in its simplest form, has four columns. This means that the switch holds four pieces of information for each virtual circuit that is already set up. We show later how the switches make their table entries, but for the moment we assume that each switch has a table with entries for all active virtual circuits. Figure shows such a switch and its corresponding table.

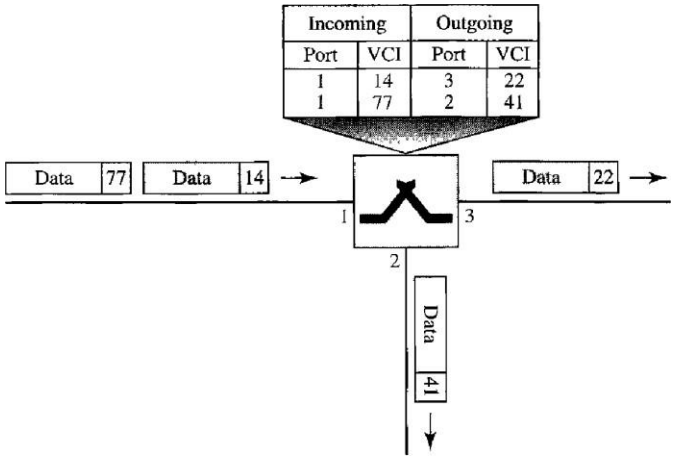
Figure shows a frame arriving at port 1 with a VCI of 14. When the frame arrives, the switch looks in its table to find port 1 and a VCI of 14.When it is found, the switch knows to change the VCI to 22 and send out the frame from port 3.

Figure shows how a frame from source A reaches destination B and how its VCI changes during the trip. Each switch changes the VCI and routes the frame.

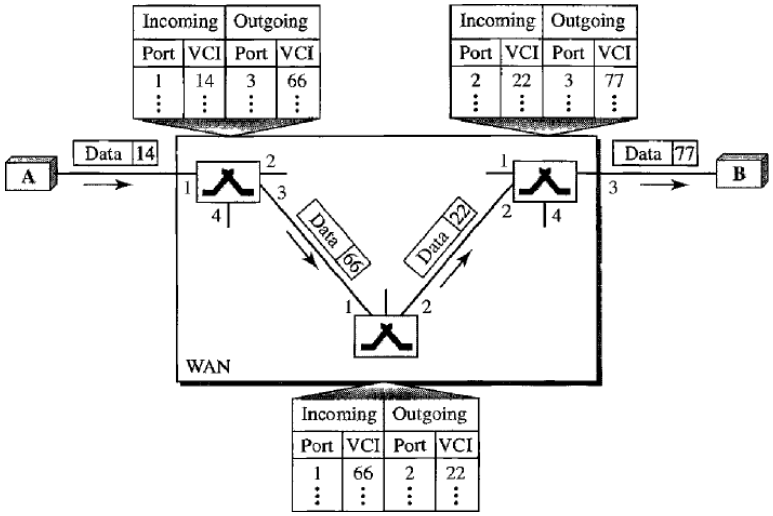
The data transfer phase is active until the source sends all its frames to the destination. The procedure at the switch is the same for each frame of messages. The process creates a virtual circuit, not a real circuit, between the sources and destination.

Setup Phase

In the setup phase, A switch creates an entry for a virtual circuit. For example, suppose source A needs to create a virtual circuit to B. Two steps are required: the setup request and the acknowledgement.



Switch and tables in a virtual circuit

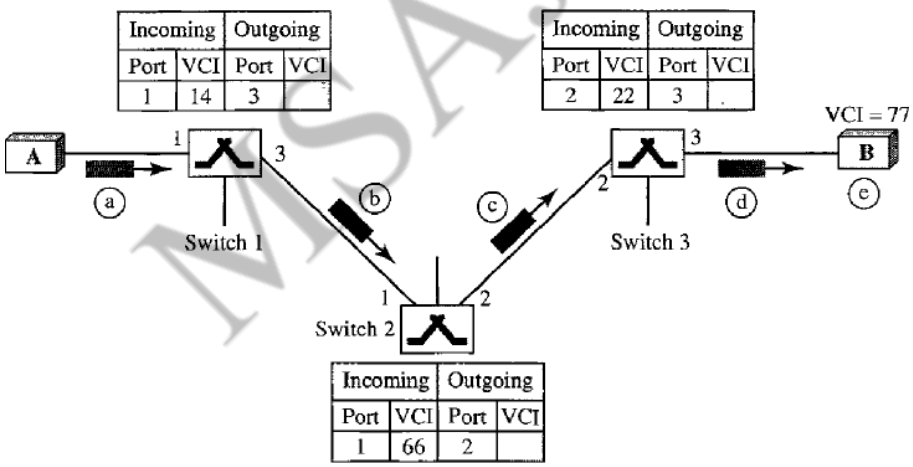


Source to destination data transfer in a virtual circuit network

Setup Request

A setup request frame is sent from the source to the destination. Figure8.14 shows the process a. source A sends a setup frame to switch 1.

b. Switch 1 receives the setup request frame. It knows that a frame going from A to B goes out through port 3. The switch, in the setup phase, acts as a packet switch; it has a routing table which is different from the switching table. For the moment, assume that it knows the output port a switch creates an entry in it table for this virtual circuit.



Setup request in a virtual circuit network

But it is only able to fill three of four columns. The switch assigns the incoming port one and chooses an available incoming VCI (14) and the outgoing port (3). It does not yet know the outgoing VCI, which will be found during the acknowledgement step. The switch then forwards the frame through port 3 to switch 2.

c. Switch 2 receives the setup request frame. The same events happen here as at switch 1; three columns of the table are completed: in this case, incoming port (1), incoming VCI (66), and outgoing port (2).

d. Switch 3receives the setup request frame. Again, three columns are completed: incoming port (2), incoming VCI (22) and outgoing port (3).

e. Destination B receives the setup frame, and if it is ready to receives frames from A, it assigns a VCI to the incoming frames that comes from A, in this case 77. This VCI lets the destination know that the frames come from A, and not other sources.

Acknowledgement

A special frame, called the acknowledgement frame, completes the entries in the switching tables. Figure shows the process.

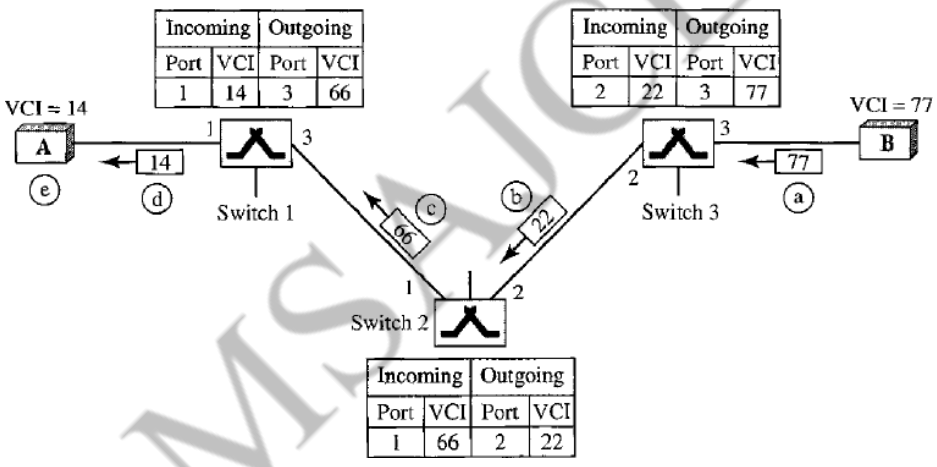
a. The destination sends an acknowledgement to switch 3. The acknowledgement carries the global source and destination addresses so the switch knows which entry in the table is to be completed. The frame also carries VCI 77, chosen by the destination as the incoming VCI for frames from A. Switch 3 uses this VCI to complete the outgoing VCI column for this entry. Note the 77 is the incoming VCI for destination B, but the outgoing VCI for switch 3.

b. Switch 3 sends an acknowledgement to switch 2 that contains its incoming VCI in the table, chosen in the previous step. Switch 2 uses this as the outgoing VCI in the table.

c. Switch 2 sends an acknowledgement to switch 1 that contains its incoming VCI in the table, chosen in the previous step. Switch 1 uses this as the outgoing VCI in the table.

d. Finally switch 1 sends an acknowledgement to source A that contains its incoming VCI in the table, chosen in the previous table.

e. The source uses this as the outgoing VCI for the data frames to be sent to destination B.



Setup Acknowledgement in a virtual-circuit network

Teardown Phase

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

MESSAGE SWITCHING NETWORKS

In message switching, each switch stores the whole message and forwards it to the next switch. Although, we don't see message switching at lower layers, it is still used in some applications like electronic mail (e-mail).

CIDR (Classless Interdomain Routing)

IPv4 addressing is classified as Classfull addressing and Classless addressing. In classful addressing, the address space is divided into five classess: A,B,C,D and E. This can be given in binary notation or dotted-decimal notation. If the address is given in binary notation, the first few bits can immediately tell us the class of the address.

Netid and Hostid

In classfull addressing, an IP address in class A,B or C is divided into netid and hostid. These parts are of varying lengths, depending on the class of the address. This concept doest not apply to classes D and E.

In class A, one byte defines the netid and three bytes define the hosted. In class B, two bytes define the netid and two bytes define the hosted. In class C, three bytes define the netid and one byte defines the hostid.

Mask

Although the length of the netid and hostid (in bits) is predetermined in classful addressing, we can also use a mask, a 32-bit number made of contiguous 1s followed by contiguoug 0s.

Class	Binary	Dotted-Decimal	CIDR
A	11111111 00000000 00000000 00000000	255.0.0.0	/8
B	11111111 11111111 00000000 00000000	255.255.0.0	/16
C	11111111 11111111 11111111 00000000	255.255.255.0	/24

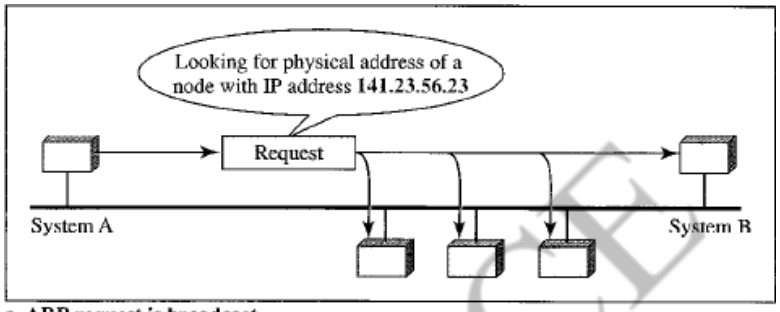
The mask can help us to find the netid and the hostid. The mask for a class A address has eight 1s, which means the first 8 bits of any address in class A define the netid, the next 24 bits define the hostid.

In table CIDR column shows the mask in the form /n where n can be 8, 16 or 24 in classful addressing. This notation is also called slash notation or **Classless Interdomain Routing (CIDR)** notation. The notation is used in classless addressing

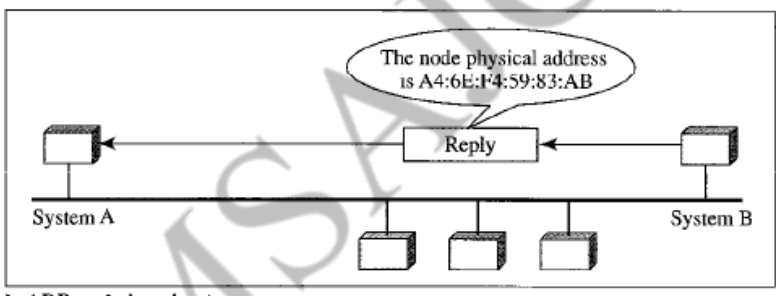
ARP : MAPPING LOGICAL TO PHYSICAL ADDRESS

Anytime a host or a route has an IP datagram to send to another host or router, it has the logical(IP) address of the receiver. The logical (IP) address is obtained from the DNS if the sender is the host or it is found in a routing table if the sender is a router. But the IP datagram must be encapsulated in a frame to be able to pass through the physical network. This means that the sender needs the physical address of the receiver. The host or the router sends an ARP query packet. The packet includes the physical and IP address of the sender and the IP address of the receiver. Because the sender does not know the physical address of the receiver, the query is broadcast over the network.

Every host or router on the network receives and processes the ARP query packet, but only the intended recipient recognizes its IP address and sends back an ARP response packet. The response packet contains the recipient's IP and physical address. The packet is unicast directly to the inquirer by using the physical address received in the query packet.



a. ARP request is broadcast



b. ARP reply is unicast

The system on the left(A) has a packet that needs to be delivered to another system(B) with IP address 141.23.56.23. System A needs to pass the packet to its data link layer for the actual delivery, but it does not know the physical address of the recipient. It uses the services of ARP by asking the ARP protocol to send a broadcast ARP request packet to ask for the physical address of a system with an IP address of 141.23.56.23.

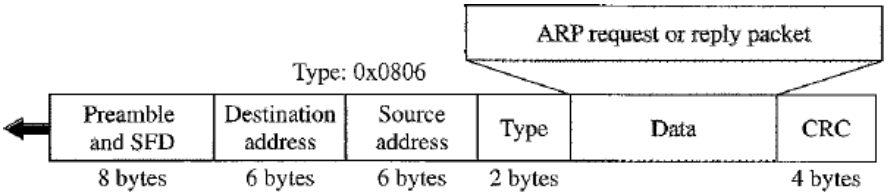
This packet is received by every system on the physical network, but only system B will answer it, as shown in figure 21.1b. System B sends an ARP reply packet that includes its physical address. Now system A can send all the packets it has for this destination by using the physical address it received.

Cache Memory

Using ARP is inefficient if system A needs to broadcast an ARP request for each IP packet it needs to send to system B. It could have broadcast the IP packet itself. ARP can be useful if the ARP reply is cached (kept in cache memory for a while) because a system normally sends several packets to the same destination. A system that receives an ARP reply stores the mapping in the cache memory and keeps it for 20 to 30 minutes unless the space in the cache is exhausted. Before sending an ARP request, the system first checks its cache to see if it can find the mapping.

Encapsulation

An ARP packet is encapsulated directly into a data link frame. For example, in figure 21.3 an ARP packet is encapsulated in an Ethernet frame. Note that the type field indicates that the data carried by the frame are an ARP packet.



Operation

Let us see how ARP functions on a typical internet. First we describe the steps involved. Then we discuss the four cases in which a host or router needs to use ARP. These are the steps involved in an ARP process.

1. The sender knows the IP address of the target. We will see how the sender obtains this shortly.
2. IP asks ARP to create an ARP request message, filling in the sender physical address, the sender IP address, and the target IP address. The target physical address field is filled with 0s.
3. The message is passed to the data link layer where it is encapsulated in a frame by using the physical address of the sender as the source address and the physical broadcast address as the destination address.
4. Every host or router receives the frame. Because the frame contains a broadcast destination address, all stations remove the message and pass it to ARP. All machines except the one targeted drop the packet. The target machine recognizes its IP address.
5. The target machine replies with an ARP reply message that contains its physical address. The message is unicast.
6. The sender receives the reply message. It now knows the physical address of the target machine.
7. The IP datagram, which carries data for the target machine, is now encapsulated in a frame and is unicast to the destination.

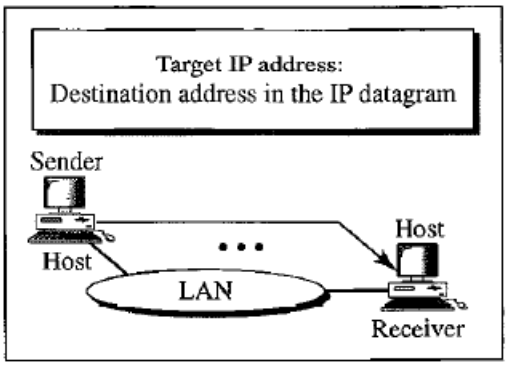
Four Different cases

The following are four cases in which the services of ARP can be used (see figure 21.4)

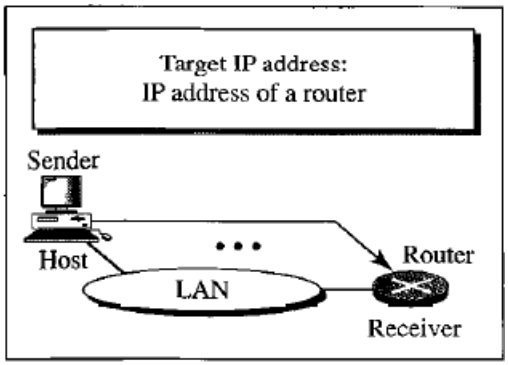
1. The sender is ahost and wants to send a packet to another host on the same network. In this case, the logical address that must be mapped to a physical address is the destination IP address in the datagram header.
2. The sender is ahost and wants to send a packet to another host on another network. In this case, the host looks at its routing table and finds the IP address of the next hop (router) for this destination. If it does not have a routing table, it looks for the IP address of the default router. The IP address of the router becomes the logical address that must be mapped to a physical address.
3. The sender is a router that has received a datagram destined for a host on another network. It checks its routing table and finds the IP address of the next router. The IP address of the next router becomes the logical address that must be mapped to a physical address.

   The sender is a router that has received a datagram destined for a host on the same network. The destination IP address of the datagram becomes the logical address that must be mapped to a physical address.
4. The sender is ahost and wants to send a packet to another host on another network. In this case, the host looks at its routing table and finds the IP address of the next hop (router) for this destination. If it does not have a routing table, it looks for the IP address

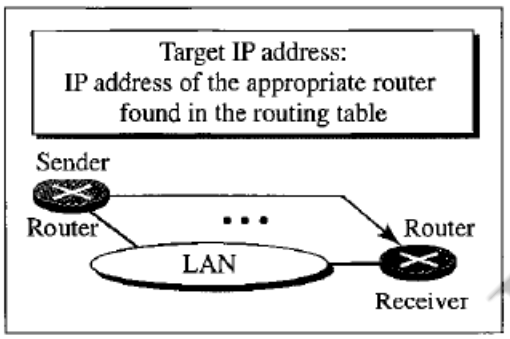
of the default router. The IP address of the router becomes the logical address that must be mapped to a physical address.



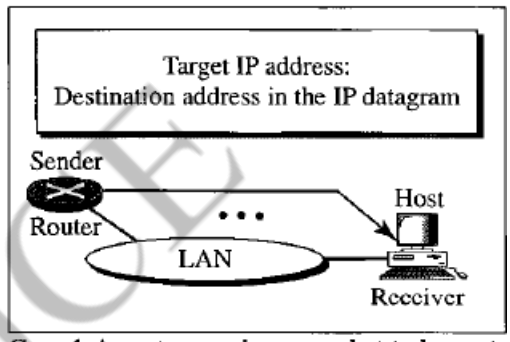
Case 1. A host has a packet to send to another host on the same network.



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



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



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

5. The sender is a router that has received a datagram destined for a host on another network. It checks its routing table and finds the IP address of the next router. The IP address of the next router becomes the logical address that must be mapped to a physical address.
6. The sender is a router that has received a datagram destined for a host on the same network. The destination IP address of the datagram becomes the logical address that must be mapped to a physical address.

DHCP

BOOTP is not a dynamic configuration protocol. When a client requests its IP address, the BOOTP server consults a table that matches the physical address of the client with its IP address. This implies that the binding between the physical address of the client already exists. The binding is predetermined.

However, what if a host moves from one physical network to another? What if a host wants a temporary IP address? BOOTP cannot handle these situations because the binding between the physical and IP address is static and fixed in stable until changed by the administrator. BOOTP is a static configuration protocol.

The dynamic host configuration protocol (DHCP) has been devised to provide static and dynamic address allocation that can be manual or automatic.

DHCP provides static and dynamic address allocation that can be manual or automatic.

Static Address Allocation: In this capacity DHCP acts as BOOTP does. It is backward compatible with BOOTP, which means a host running the BOOTP client can request a static address from a DHCP server. A DHCP server has a database that statically binds physical address to IP address.

Dynamic Address Allocation: DHCP has a second database with a pool of available IP address. This second database makes DHCP dynamic. When a DHCP client requests a temporary IP address, the DHCP server goes to the pool of available (unused) IP address and assigns an IP address for s negotiable period of time.

When a DHCP client sends a request to a DHCP server, the server first checks its static database. If an entry with the requested physical address exists in the static database, the permanent IP address of the client is returned. On the other hand, if the entry does not exists in the static database, the server selects an IP address from the available pool, assigns the address to the client, and adds the entry to the dynamic database.

The dynamic aspect of DHCP is needed when a host moves from network to network or is connected and disconnected from a network (as is a subscriber to a service provider). DHCP provides temporary IP address for a limited time.

The address assigned from the pool is the temporary address. The DHCP server issues a lease for a specific time. When the lease expires, the client must either stop using the IP address or renew the lease. The server has the option to agree or disagree with the renewal. If the server disagrees, the client stops using the address.

Manual and automatic configuration One major problem with the BOOTP protocol is that the table mapping the IP address to physical addresses needs to be manually configured. This means that every time there is a change in a physical or IP address, the administrator needs to manually enter the changes. DHCP, on the other hand, allows both manual and automatic configurations. Static addresses are created manually: dynamic addresses are created automatically.

ICMP

IP provides unreliable and connectionless datagram delivery. It was designed this way to make efficient use of network resources. The IP protocol is a best-effort delivery service that delivers a datagram from its original source to its final destination. However, it has two deficiencies from its original source to its final destination. However, it has two deficiencies: lack of error control and lack of assistance mechanisms.

The IP protocol has no error –reporting or error-correcting mechanisms. What happens if something goes wrong? What happens if a router must discard a datagram because it cannot find a router to the final destination, or because the time –to- live field has zero value? What happens if the final destination host must discard all fragments of a datagram because it has not received all fragments within a predetermined time limit? These are examples of situations where an error has occurred and IP protocol has no build –in mechanisms to notify the original host.

The IP protocol also lacks a mechanism for host and management queries. A host sometimes needs to determine if a router or another host is alive. And sometimes a network administrator needs information from another host or router.

The internet control message protocol (ICMP) has been designed to compensate for the above two deficiencies. It is a companion to the IP protocol.

Types of messages

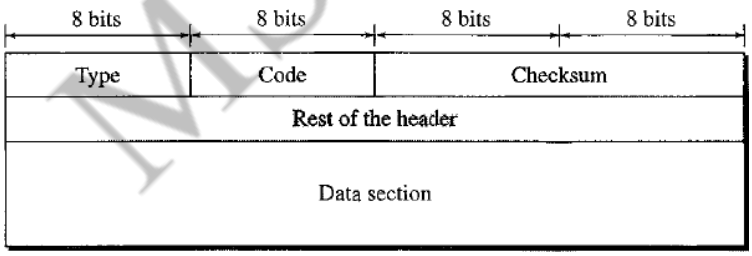
ICMP messages are divided into two broad categories:
error-repeating messages and query messages.

The error –reporting messages report problems that the router or a host (destination) may encounter when it processes an IP packet.

The query messages, which occur in pairs, help a host or a network manager get specific information from a router or another host. For example, nodes can discover their neighbours. Also, hosts can discover and learn about routers on their network, and routers can help node redirect its messages.

Message Format

ICMP message has 8-byte header and a variable-size data section. ICMP type defines the type of the message. The code field specifies the reason for the particular message type. The last common field is the checksum field. The rest of the header is specific for each message type.



Error reporting

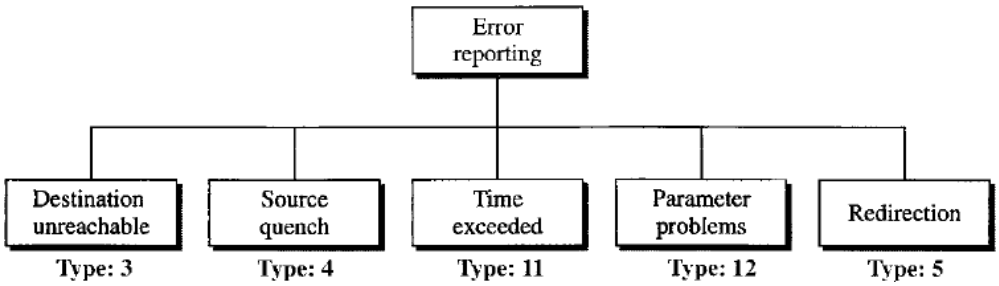
One of the main responsibilities of ICMP is to report errors. Although technology has produced increasingly reliable transmission media, errors still exist and must be handled. IP, is an unreliable protocol. This means error checking and error control are not concern of IP. ICMP designed, in part, to compensate for this short coming. However, ICMP does not correct errors-it simply reports them. Error correction is left to higher –level protocols. Error messages are always sent to the original source because the only information available in the datagram about the route is the source and destination IP addresses. ICMP uses the source IP address to send the error message to the source(originator) of the datagram.

Five types of errors are handled: **destination unreachable, source quench, time exceeded, parameter problems, and redirection.**

The following are the important points about ICMP error messages:

1. No ICMP error message will be generated in response to a datagram carrying an ICMP error message.
2. No ICMP error message will be generated for a fragmented datagram that is not the first fragment.

3. No ICMP error message will be generated for a datagram having multicast address.
4. No ICMP error message will be generated for a datagram having a special address such as 127.0.0.0 or 0.0.0.0.



Destination unreachable

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

Source quench

The IP protocol is a connectionless protocol. There is no communication between the source host, which produces the datagram, the routers, which forward it, and destination host, which processes it. One of the ramifications of this absence of communication is the lack of flow control. IP does not have a flow control mechanism embedded in the protocol. The lack of flow control can create a major problem in the operation of IP: congestion. The source host never knows if it is producing datagrams faster than can be forwarded by routers or processed by destination host.

The lack of flow control can create congestion in routers or the destination host. A router or a host has a limited-size queue (buffer) for incoming datagrams waiting to be forwarded (in case of router) or to be processed (in case of a host). If the datagrams are received much faster than they can be forwarded or processed, the queue may overflow. In this case, the router or the host has no choice but to discard some of the datagrams. The source – quench message in ICMP was designed to add a kind of flow control to the IP. When a router or host discards a datagram due to congestion, it sends a source –quench message to the sender of the datagram. This message has two purposes. First, it informs the source that the datagram has been discarded. Second, it warns the source that there is congestion somewhere in the path and that the source should slow down (quench) the sending process.

Time exceeded

The time –exceeded message is generated in two cases: routers use routing tables to find the next hop (next router) that must receive the packet. If there are errors in one or more routing tables, a packet can travel in a loop or a cycle, going from one router to the next or visiting a series of routers endlessly. Each datagram contains a field called time to live that controls the situation. When a datagram visits a router, the value of this field is decremented by 1. When the time-to –live value reaches 0, after decrementing, the router discards the datagram. However, when the datagram is discarded, a time –exceeded message must be sent by the router to the original source. Second, a time –exceeded message is also generated when all fragments that make up a message arrive at the destination host within a certain limit.

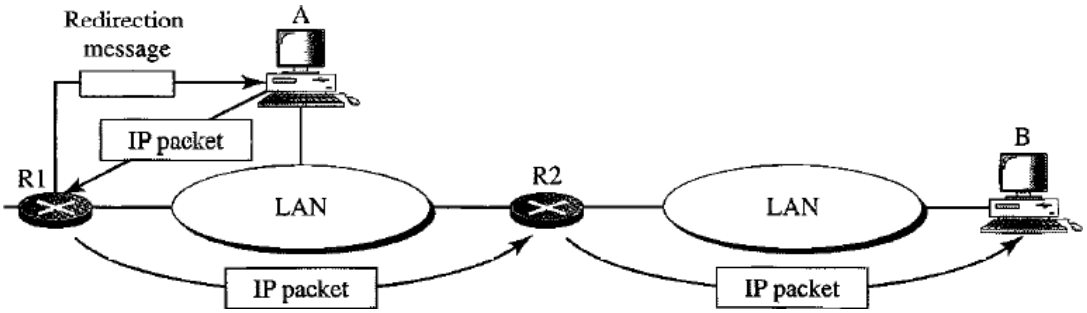
Parameter problem

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

Redirection

When a router needs to send a packet destined for another network, it must know IP address of the next appropriate router. The same is true if the sender is a host. Both routers and hosts, then, must have a routing table to find address of router or the next router. Routers take part in routing update process, and are supposed to update constantly. Routing is dynamic.

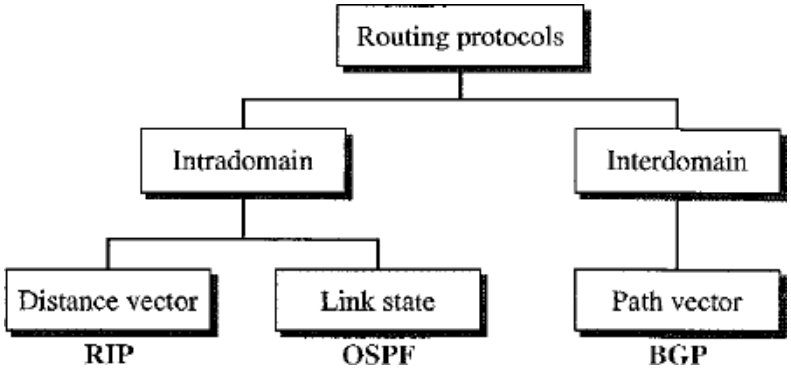
However, for efficiency, host do not take part in routing update process because there are many more host in an internet than routers. Updating the routing table of the hosts dynamically produces unacceptable traffic. The hosts usually use static routing. When a host comes up, its routing table has a limited number of entries. It usually knows the IP address of only one router, the default router. For this reason, the host may send a datagram, which is destined for another network, to the wrong router. In this case, the router that receives the datagram will forward the datagram to the correct router. However, to update the routing table of the host, it sends a redirection message to the host. Host A wants to send a datagram to host B.



Router R2 is obviously the most efficient routing choice, but host A did not choose router R2. The datagram goes to R1 instead. Router R1, after consulting its table, finds that the packet should have gone to R2. It sends the packet to R2 and, at the same time, sends a redirection message to host A. host A's routing table can now be updated.

UNIT - 3

ROUTING PROTOCOLS

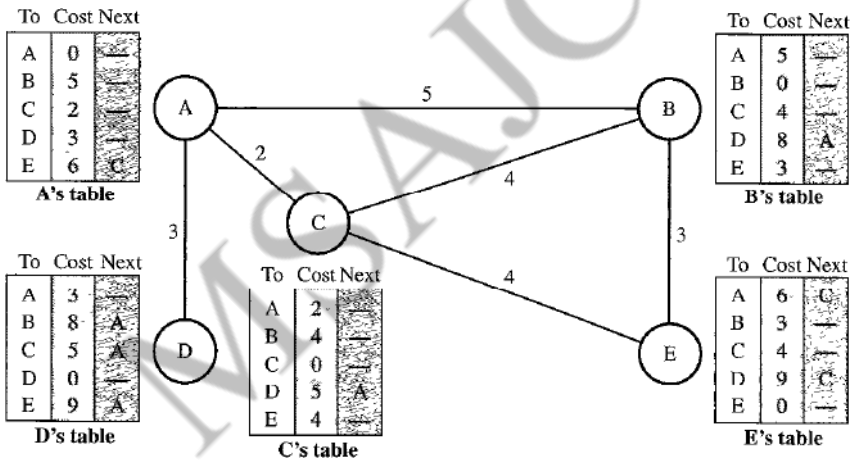


Routing Information Protocol (RIP) is implemented in **Distance Vector protocol**.
Open Shortest Path First (OSPF) is implemented in **Link state protocol**.
Border Gateway protocol (BGP) is implemented in **Path vector protocol**.

DISTANCE VECTOR ROUTING (RIP)

In distance vector routing, the least-cost route between any nodes is the route with minimum distance. In this protocol, as the name implies, each node maintains a vector (table) of minimum distance to every node. The table at each node also guides the packets to the desired node by showing the next stop in the route (next-hop routing).

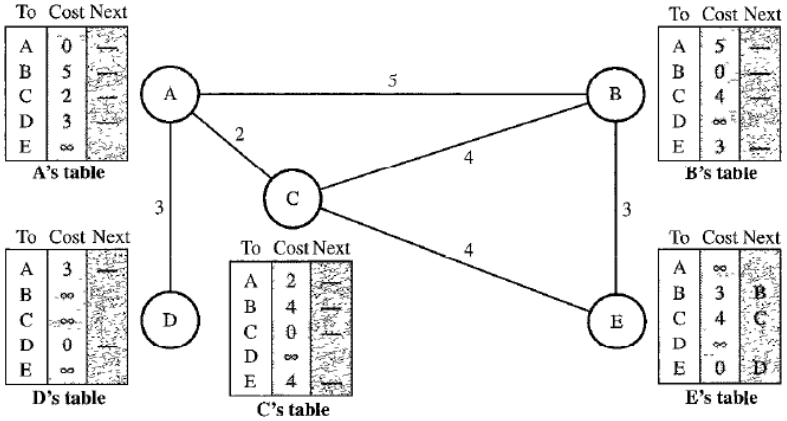
We can think of nodes as the cities in an area and the lines as the roads connecting them. A table can show a tourist the minimum distance between cities.



Distance vector routing table

The table for node A shows how we can reach any nodefrom this node. For example, our least cost to reach node E is 6. The route passes through C.

Initialization



Initialization of tables in distance vector routing

Each node knows how to reach any other node and the cost. At the beginning, however, this is not the case. Each node can know only the distance between itself and its immediate neighbours, those directly connected to it. So for the moment, we assume that each node can send a message to the immediate neighbours and find the distance between itself and these neighbours. Figure 22.15 shows the initial tables for each node. The distance for any entry that is not a neighbour is marked as infinite (unreachable).

Sharing

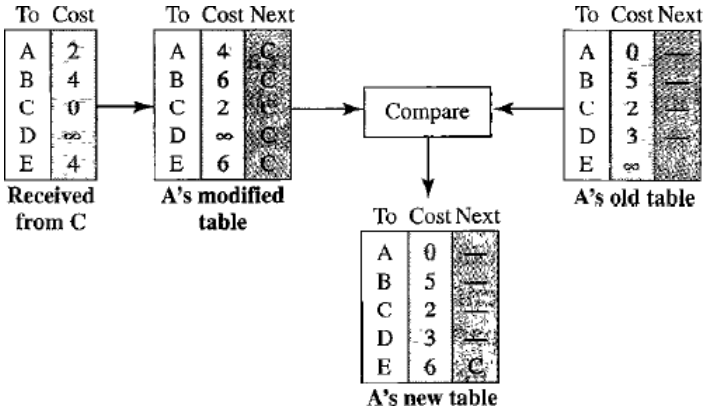
The whole idea of the distance vector routing is the sharing of information between neighbors. Although node A does not know about node E, node C does. So if the node C shares its routing table with A, node A can also know how to reach node E. on the other hand, node C does not know how to reach node D, but node A does. If the node A shares its routing table with node C, as immediate neighbors, can improve their routing table if they help each other.

There is only one problem. How much of the table must be shared with each neighbour? A node is not aware of neighbour's table. The best solution for each node is to send its entire table to the neighbour and let the neighbour decide what part to use and what part to discard. However, the third column of a table (next stop) is not useful for neighbor. When the neighbour receives a table, this column needs to be replaced with the sender's name. If any of the rows can be used, the next node is the sender of the table. A node therefore can send only the first two columns of its table to any neighbour. In other words,sharing here means sharing only the first two columns.

Updating

When a node receives a two-column table from a neighbour, it needs to update its routing table. Updating takes three steps:

1. The receiving node needs to add the cost between itself and the sending node to each value in the second column. The logic is clear. If node C claims that its distance to a destination is x mi, and the distance between A and C is y mi, then the distance between A and that destination, via C, is x+y mi .
2. The receiving node needs to add the name of the sending node to each row as the third column if the receiving node uses information from any row. The sending node is the next node in the route.
3. The receiving node needs to compare each row of its old table with the corresponding row of the modified version of the received table.
   a. If the next- node entry is different, the receiving node chooses the rowwith the smaller cost. If there is a tie, the old one is kept.
   b. If the next- node entry is the same, the receiving node chooses the new row. For example, suppose node C has previously advertised a route to node X with distance 3. Suppose that now there is no path between C and X; node C now advertises this route with a distance of infinity. Node A must not ignore this value even though its old entry is smaller. The old route does not exist anymore. The new route has a distance of infinity.



Updating in distance vector routing

There are several points we need to emphasize here. First, as we know from mathematics, when we add any number to infinity, the result is still infinity. Second, the modified table shows how to reach A from A via C. if A needs to reach itself via C, it needs to go to c and come back, a distance of 4. Third, the only benefit from this updating of node A is the last entry, how to reach E. Previously, node A did not know how to reach E (distance of infinity); now it knows that the cost is 6 via C.

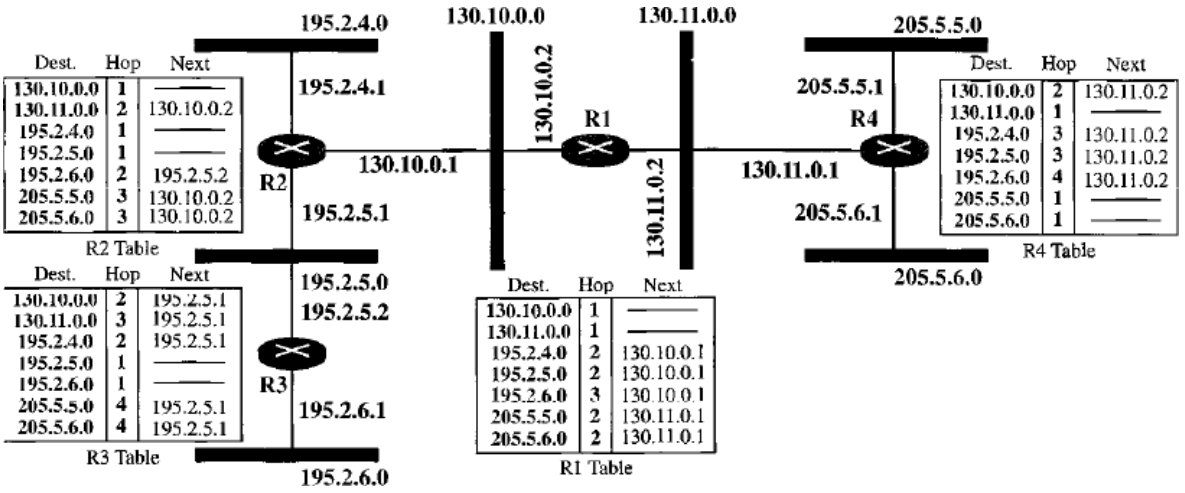
Each node can update its table by using the tables received from other nodes. In a short time, if there is no change in the network itself, such as a failure in a link, each node reaches a stable condition in which the contents of its table remains the same.

RIP

The Routing Information Protocol (RIP) is an intradomain routing protocol used inside an autonomous system. It is very simple protocol based on distance vector routing. RIP implements distance vector routing directly with some considerations:

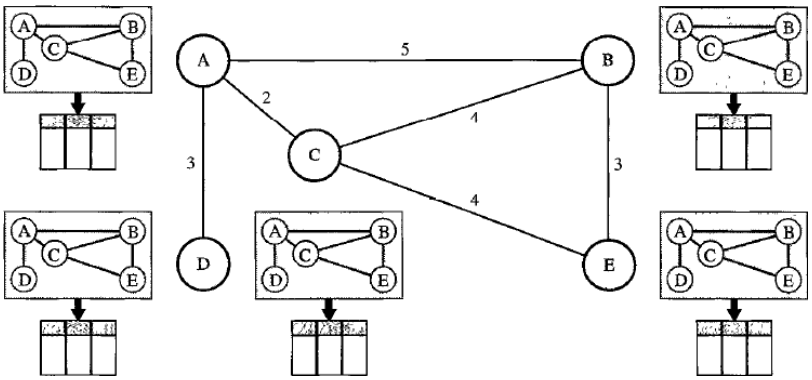
1. In an autonomous system, we are dealing with routers and networks (links). The routers have routing tables; networks do not.
2. The destination in a routing table is a network, which means the first column defines a network address.
3. The metric used by RIP is very simple; the distance is defined as the number of links (networks) to reach the destination. For this reason, the metric in an RIP is called a hop count.
4. Infinity is defined as 16, which means that any route in an autonomous system using RIP cannot have more than 15 hops.
5. The next- node column defines the address of the router to which the packet is to be sent to reach its destination.

Figure shows an autonomous system with seven networks and four routers. The table of each router is also shown. Let us look at the routing table for R1. The table has seven entries to show how to reach each network in the autonomous system. Router R1 is directly connected to networks 130.10.0.0 and 130.11.0.0, which means that there are no next – hop entries for these two network.To send a packet to one of the three networks at the far left, router R1 needs to deliver the packet to R2. The next-node entry for these three networks is the interface of router R2 with IP address 130.10.0.1. To send a packet to the two networks at the far right, route R1 needs to send the packet to the interface of router R4 with IP address 130.11.0.1. The other tables can be explained similarly.



LINK STATE ROUTING

Link sate routing has a different philosophy from that of distance vector routing. In link state routing, if each node in the domain has the entire topology of the domain the list of nodes and links, how they are connected including the type, cost (metric), and condition of the links (up or down) the node can use Dijkstra's algorithm to build a routing table.

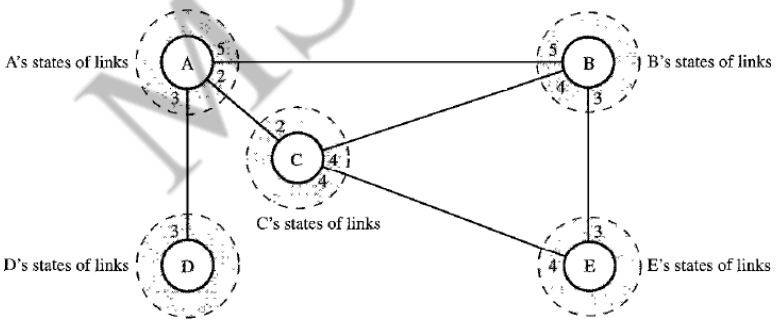


Concept of Link State Routing

The figure shows a simple domain with five nodes. Each node uses the same topology to create a routing table, but the routing table for each node is unique because the calculations are based on different interpretations of the topology. This is analogous to a city map. While each person may have the same amp, each needs to take a different route to reach her specific destination.

The topology must be dynamic, representing the latest state of each node and each link. If there are changes in any point in the network (a link is down, for example), the topology must be updated for each node.

How can a common topology be dynamic and stored in each node? No node can know the topology at the beginning or after a change somewhere in the network. Link state routing is based on the assumption that, although the global knowledge: it knows the state (type, condition, and cost) of its link. In other words, the whole topology can be compiled from the partially knowledge of each node. Figure 22.21 shows the same domain as in figure 22.20, indicating the part of the knowledge belonging to each node.



Link state knowledge

Node A knows that it is connected to node B with metric 5, to node C with metric 2, and to node D with metric 3. Node C knows that it is connected to node A with metric 2, to node B with metric 4, and to node E with metric 4. Node D knows that it is connected only to node A with metric 3. And so on. Although there is an overlap in the knowledge, the overlap guarantees the creation of a common topology a picture of the whole domain for each node.

Building routing tables

In link state routing, four sets of actions are required to ensure that each node has the routing table showing the least-cost node to every other node.

1. Creation of the states of the links by each node, called the link state packet (LSP).
2. Dissemination of LSPs to every other router, called flooding, in an efficient and reliable way.
3. Formation of a shortest path tree for each node.
4. Calculation of a routing table based on the shortest path tree.

Creation of link state packet (LSP)

A link state packet can carry a large amount of information. For the moment, however, we assume that it carries a minimum amount of data: the node identity, the list of links, a sequence number, and age. The first two, node identity and the list of links, are needed to make the topology. The third sequence number , facilitates flooding and distinguishes new LSPs from old ones. The fourth, age, prevents old LSPs from remaining in the domain for a long time. LSPs are generated on two occasions:

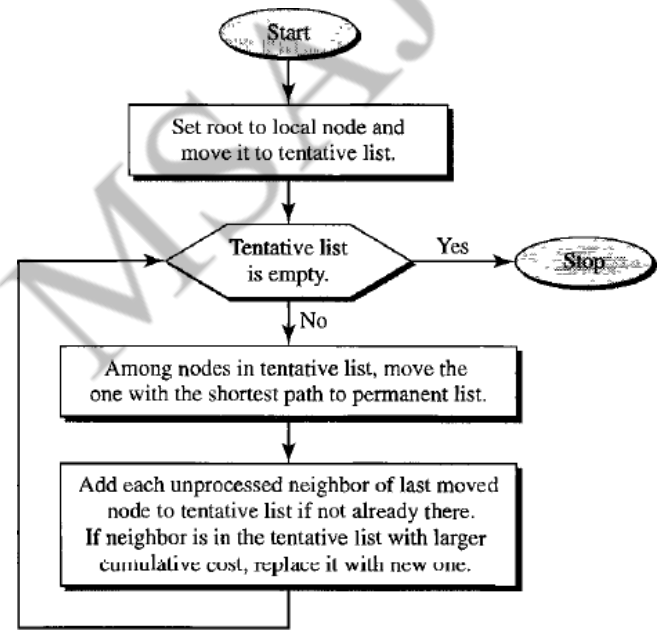
1. When there is a change in the topology of the domain. Triggering of LSP dissemination is the domain way of quickly informing any node in the domain to update its topology.
2. On a periodic basis. The period in this case is much longer compared to distance vector routing. As a matter of fact, there is no actual need for this type of LSP dissemination. It is done to ensure that old information is removed from the domain. The timer set for periodic dissemination is normally in the range of 60 mi or 2h based on implementation. A longer period ensures that flooding does not create too much traffic on the network.

Flooding of LSPs

After a node has prepared an LSP, it must be dissemination to all other nodes, not only to its neighbors. The process is called flooding and based on the following:

1. The creating node sends a copy of the LSP out of each interface.
2. A node that receives an LSP compares it with the copy it may already have .if the newly arrived LSP is older than the one it has (found by checking the sequence number), it discards the LSP and keeps the new one.
   a. It discards the old LSP and keeps the new one.
   b. It sends a copy of it out of each interface except the one from which the packet arrived. This guarantees that flooding stops somewhere in the domain (where a node has only one interface).

Formation of shortest path tree: Dijkstra algorithm



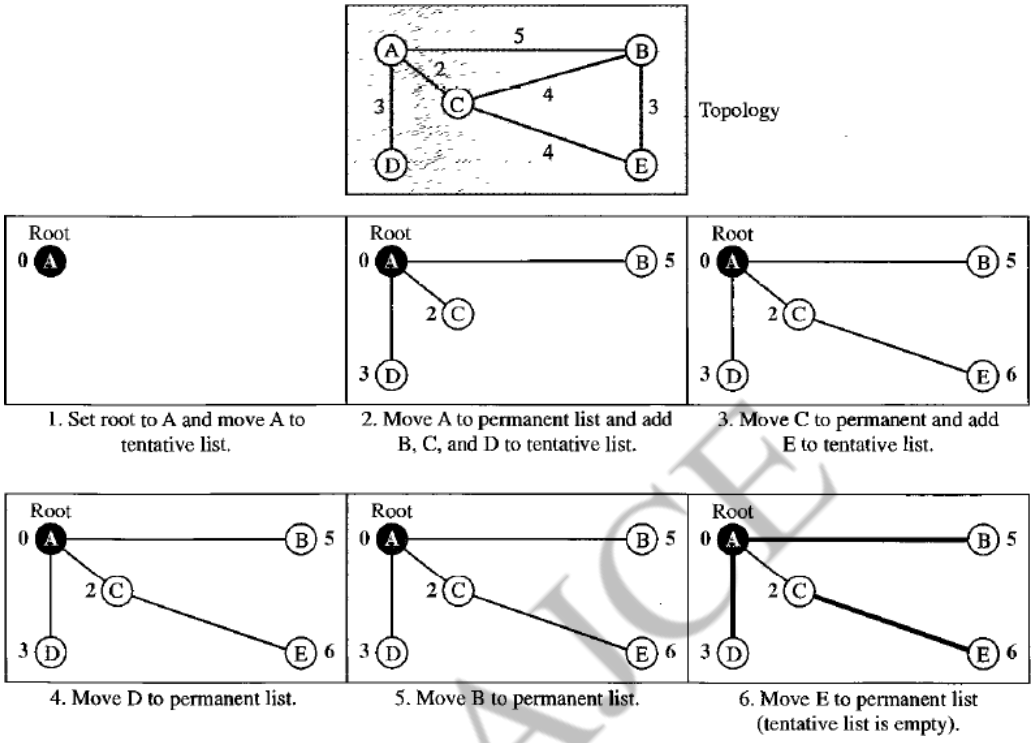
After receiving all LSPs, each node will have a copy of the whole topology. However, the topology is not sufficient to find the shortest path to every other node; a shortest path tree is needed.

A tree is agraph of nodes and links; one node is called the root. All other nodes can be reached from the root through only one single route. A shortest path tree is atree in which the path between the root and every other node is the shortest. What we need for each node is a shortest path tree with that node as the root.

The **Dijkstra algorithm** creates a shortest path tree from agraph. The algorithm divides the nodes into two sets: tentative and permanent. It finds the neighbors of a current node,

makes them tentative, examines them, and if they pass the criteria, makes them permanent. We can informally define the algorithm by using the flowchart
Let us apply the algorithm to node A of our sample graph. To find the shortest path in each step, we need the cumulative cost from the root to each node, which is shown next to the node.

The following shows the steps. At the end of each step, we show the permanent (filled circles) and the tentative(open circles) nodes and lists with the cumulative costs.



Calculation of Routing Table from Shortest Path Tree

Each node uses the shortest path tree protocol to construct its routing table. The routing table shows the cost of reaching each node from the root.

Node	*Cost*	*Next Router*
A	0	—
B	5	—
C	2	—
D	3	—
E	6	C

Routing table for Node A

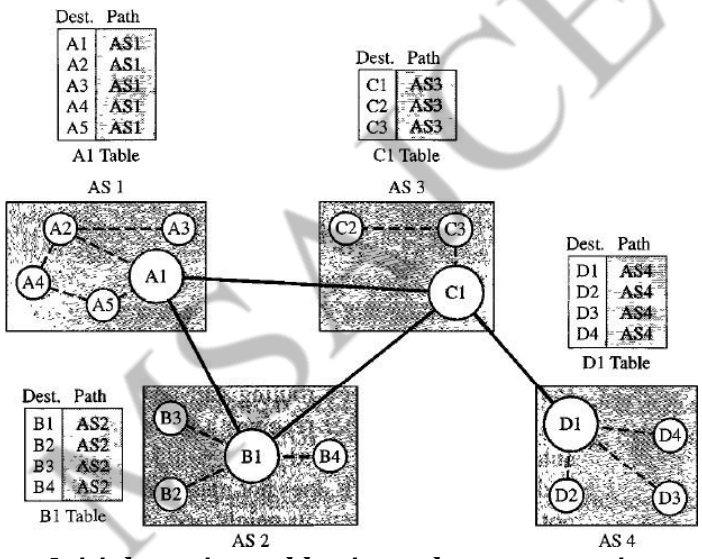
PATH VECTOR ROUTING

Distance vector and link state routing are both intradomain routing protocols. They can be used inside an autonomous system, but not between autonomous systems. These two protocols are not suitable for interdomain routing mostly because of scalability. Both of these routing protocols become intractable when the domain of operation becomes large. Distance vector routing is subject to instability if there are more than a few hops in the domain of operation. Link state routing needs a huge amount of resources to calculate routing tables. It also creates heavy traffic because of flooding. There is aneed for a third routing protocol which we call path vector routing.

Path vector routing proved to be useful for interdominance routing. The principle of path vector routing is similar to that of distance vector routing .in path vector routing, we assume that there is one node (there can be more, but one is enough for our conceptual discussion) in each autonomous system. Let us call it the speaker node. The speaker node in an AS creates a routing table and distances and advertises it to the speaker nodes in the neighboring Ass. The idea is the same as for distance vector routing except that only speaker nodes in each AS can communicate with each other. However, what is advertised is different. A speaker node advertises the path, not the metric of the nodes, in its autonomous system or other autonomous systems.

Initialization

At the beginning, each speaker node can know only the reachability of nodes inside its autonomous system.



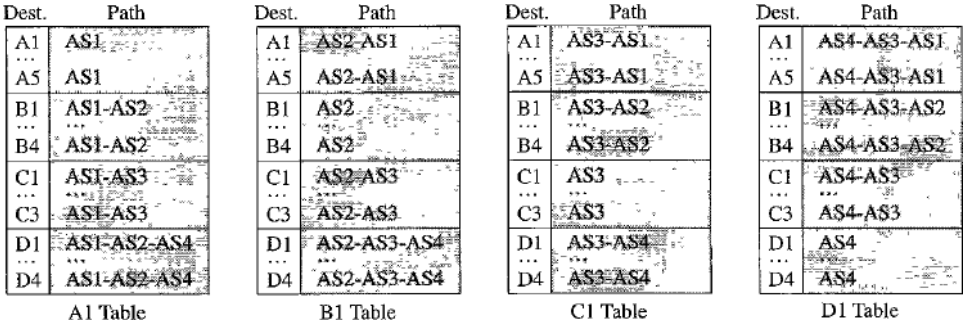
Initial routing tables in path vector routing

Fig shows the initial tables for each speaker node in a system made of four ASs.
Node A1 is the speaker node for AS1, B1 for AS2, C1 for AS3, and D1 for AS4. Node A1 creates an initial table that shows A1 to A5 are located in AS1 and can be reached through it. Node B1 advertises that B1 to B4 are located in AS2 and can be reached through B1. And so on.

Sharing Just as in distance vector routing, in path vector, a speaker in an autonomous system shares its table with immediate neighbours. Node A1 shares its table with nodes D1,B1 and A1. Node B1 shares its table with C1 and A1. Node D1 shares its table with C1.

Updating When a speaker node receives a two column table from a neighbour, it updates its own table by adding the nodes that are not in its routing table and adding tits own autonomous system and the autonomous system that sent the table. After a while each speaker has a table and knows how to reach each node in other Ass.
According to the figure, if router A1 receives a packet for nodes A3, it knows that the path is in AS1, but if it receives a packet for D1, it knows that the packet should go from AS1 to AS2 and then to AS3. The routing table shows the path completely. On the other hand, if node D1 in AS4 receives a packet for node A2, it knows it should go through AS4, AS3 and AS1.

Loop prevention: The instability of distance vector routing and the creation of loops can be avoided in path vector routing. When a router receives a message, it checks to see if its autonomous system is in the path list to the destination. If it is, looping is involved and the message is ignored.



Dest.	Path
A1	AS1
...	
A5	AS1
B1	AS1-AS2
...	...
B4	AS1-AS2
C1	AS1-AS3
...	...
C3	AS1-AS3
D1	AS1-AS2-AS4
...	...
D4	AS1-AS2-AS4

A1 Table

Dest.	Path
A1	AS2-AS1
...	
A5	AS2-AS1
B1	AS2
...	...
B4	AS2
C1	AS2-AS3
...	...
C3	AS2-AS3
D1	AS2-AS3-AS4
...	...
D4	AS2-AS3-AS4

B1 Table

Dest.	Path
A1	AS3-AS1
...	
A5	AS3-AS1
B1	AS3-AS2
...	...
B4	AS3-AS2
C1	AS3
...	...
C3	AS3
D1	AS3-AS4
...	...
D4	AS3-AS4

C1 Table

Dest.	Path
A1	AS4-AS3-AS1
...	
A5	AS4-AS3-AS1
B1	AS4-AS3-AS2
...	...
B4	AS4-AS3-AS2
C1	AS4-AS3
...	...
C3	AS4-AS3
D1	AS4
...	...
D4	AS4

D1 Table

Stabilised tables for three autonomous systems

Policy routing: Policy routing can be easily implemented through path vector routing. When a router receives a message, it can check the path. It can ignore that path and that destination. It does not update its routing table with its path, and it does not send this message to its neighbors.

Optimum path: What is the optimum path vector routing? We are looking for a path to a destination that is the best for the organization that runs the autonomous system. We definitely cannot include metrics in this route because each autonomous system that is included in the path may use a different criterion for the metric. One system may use internally, RIP,which defines hop count as the metric; another may use OSPF with minimum delay defined as the metric .The optimum path is the path that fits the organization. In each autonomous may have more than one path to a destination. For example, path from AS4 to AS1 can be AS4-AS3-AS2-AS1,or it can be AS4-AS3-AS1. For the tables,we chose theone that had the smaller number of autonomous system, but this is not always the case. Other criteria, such as security, safety , and reliability, can also be applied.

BGP

Border Gateway Protocol (BGP) is an interdomain routing protocol using apath vector routing. It first appeared in 1989 and as gone through four versions.

Types of autonomous systems: As we said before, the internet is divided into hierarchical domains called autonomous system. For example, a large corporation that manages its own network and has full control over it is an autonomous system. We can divide autonomous system into three categories: stub, multihomed, and transit.

Stub AS: A stub AS only one connection to another AS. theinterdomaindata traffic in a stub AS can be either created or terminated in the AS. the hosts in the AS can receive data coming from hosts in other ASs. Data traffic, however, cannot pass through a stub AS.A stub AS is either a source or a sink. A good example of a stub AS is a small corporation or a small local ISP.

Multihomed AS: A multihomed AS has a more than one connection to other Ass, but it is still only a source or sink for data traffic. It can receive data traffic from more than one AS. it can send data traffic to more the one AS, but there is no transient traffic. It does not allow data coming from one AS and going to another AS to pass through. A good example of a multihomed AS is a large corporation that is connected to more than one regional or national AS that does not allow transient traffic.

Transit AS: A transit AS is a multihomed AS that also allows transient traffic. Good examples of transit Ass are national and international ISPSs (internet backbones) .

Path Attributes: In our previous examples, we discussed a path for a destination network. The path was present as a list of autonomous system, but is, in fact, a list of attributes. Each attributes gives some information about the path. The list of attributes helps the receiving router make a more-informed decision when applying its policy.

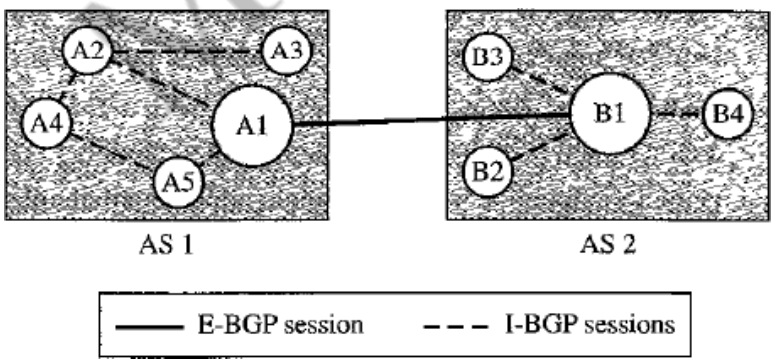
Attributes are divided into two broad categories: well- known and optional. A well- known attribute is one that every BGP router must recognize. An optional attribute is one that needs not be recognized by every router.

Well-known attributes are themselves divided into two categories: mandatory and discretionary. A well-known mandatory attributes is one that must appear in the description of a route. A well-known discretionary attribute is one that must be recognized by each router, but is not required to be included in every update message. One well-known mandatory attribute is ORIGIN. This defines the source of the routing information (RIP, OSPF, and so on). Another well- known mandatory attribute is AS_PATH. This defines the list of autonomous systems through which the destination can be reached. Still another well-known mandatory attribute is NEXT-HOP, which defines the next router to which the data packet should be sent.

The optional attributes can also be subdivided into two categories: transitive and nontransitive. An optional transitive attribute is one that must be passed to the next router by the router that has not implemented this attribute. An optional nontransitive attribute is one that must be discarded if the receiving router has not implemented it.

BGP Sessions: The exchange of routing information between two routers using BGP takes place in a session. A session is a connection that established between two BGP routers only for the sake of exchanging routing information. To create a reliable environment, BGP uses the services of TCP. In other words, a session at the BGP level, as an application program, is a connection at the TCP level. However, there is a subtle difference between a connection in TCP made for BGP and other application programs. When a TCP connection is created for BGP, it can last for a long time, until something unusal happens. For this reason, BGP sessions are sometimes referred to as semi-permanent connection.

External And Internal BGP: If we want to precise, BGP can have two types of sessions: external BGP (E-BGP) and international BGP (I-BGP) sessions. The E-BGP session is used to exchange information between two speaker nodes belonging to two different autonomous systems. The I-BGP session, on the other hand, is used to exchange routing information between two routers inside an autonomous system.

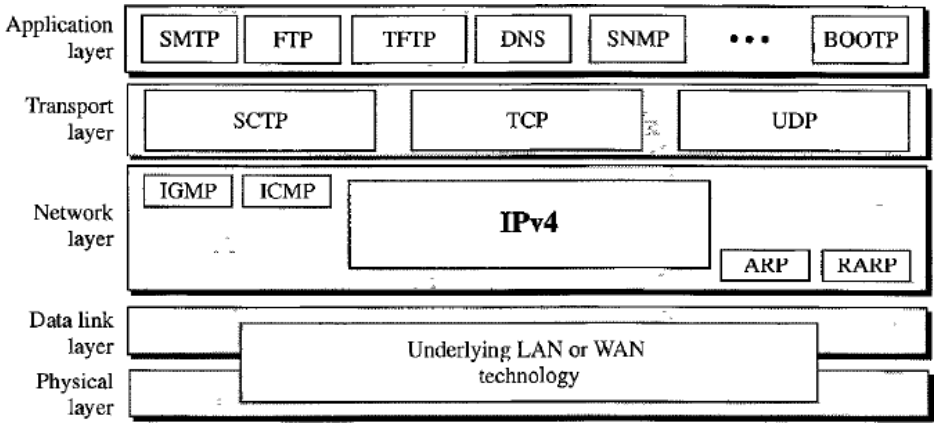


Internal and External BGP sessions

The session established between AS1 and AS2 is an E-BGP session. The two speaker routers exchange information they know about networks in the networks in the internet. However, these two routers need to collect information from other routers in the autonomous systems.

IPv4

The internet protocol version 4(IPv4) is the delivery mechanism used by the TCP/IP protocols.



IPv4 in TCP/IP protocol suite

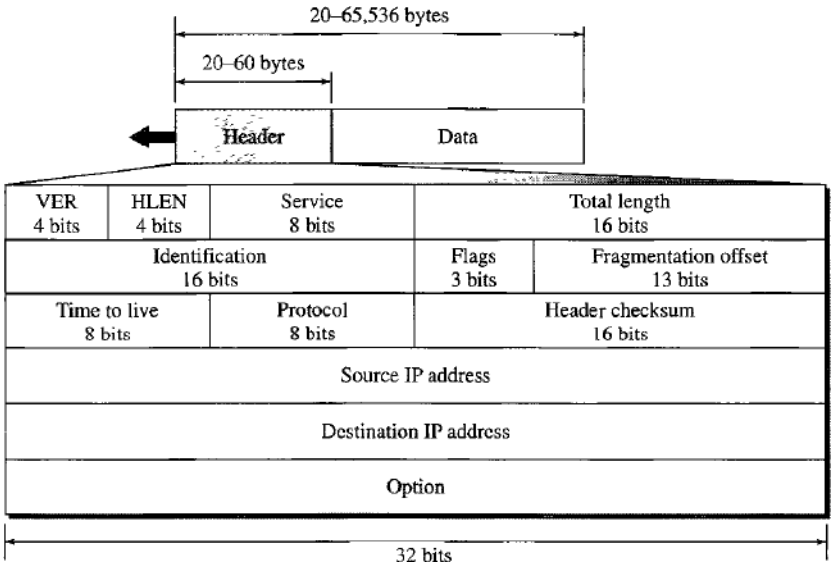
IPv4 is an unreliable and connectionless datagram protocol -a best -effort delivery service. The term best effort means that IPv4 provides no error control or flow control (except for error detection on the header). IPv4 assumes the unreliability of the underlying layers and does it best to get a transmission through to its destination, but with no guarantees.

If reliability is important, IPv4 must be paired with a reliable protocol such as TCP. An example of a more commonly understood best-effort delivery service is the post office. The post office does it best to deliver the mail but does not always succeed. If an unregistered letter is lost, it is up to the sender or would be recipient to discover the loss and rectify the problem. The post office itself does not keep track of every letter and cannot notify a sender of loss or damage.

IPv4 is also connectionless protocol for a packet-switching network that uses the datagram approach. This means that each datagram is handled independently, and each datagram can follow a different route to destination. This implies that datagrams sent by the same source to the same destination could arrive out of order. Also, some could be lost or corrupted during transmission. Again, IPv4 relies on a higher-level protocol to take care of all these problems.

Datagrams

Packets in the IPv4 layer are called datagrams.



IPv4 datagram format

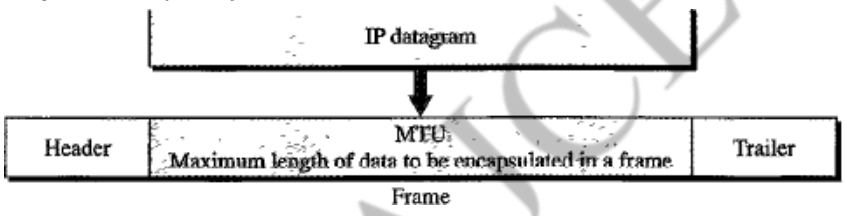
A datagrams is a variable length packet consisting of two parts: header and data. The header is 20 to 600 bytes in length and contains information essential to routing and delivery. It is customary in TCP/IP to show header in 4-byte sections.

- **Version(VER):** This 4-bit field defines the version of the IPv4 protocol. Currently the version is 4. However, version 6(or IPng) may totally replace version 4 in the future. This field tells the IPv4software running in processing the machine that the datagram has the format of version 4. All fields must be interpreted as specified in the fourth version of the protocol. If the machine is using some other version of IPv4, the datagram is discarded rather than interpreted incorrectly.
- **Header length (HLEN):** This 4-bit filed defines the total length of the datagram header in 4byte words. This field is needed because the length of the header is variable (between 20 and 60 bytes). When there are no options, the header length is 20 bytes and the value of this filed is 5 (5X4=20). When the option field is at its maximum size, the value of this field is 15 (15X4=60).
- **Services:** IETF has changed the interpretation and name of this 8-bit field. This field, previously called service type, is now called differentiated services.

Fragmentation

A datagram can travel through different networks. Each router decapsulates the IPv4 datagram from the frame it receives, processes it, and then encapsulates it in another frame.

Maximum Transfer Unit (MTU)



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

Each data link layer protocol has its own frame format in most protocols. When a datagram is encapsulated in a frame, the total size of datagram must be less than this maximum size. The MTU depends on the physical network protocol.

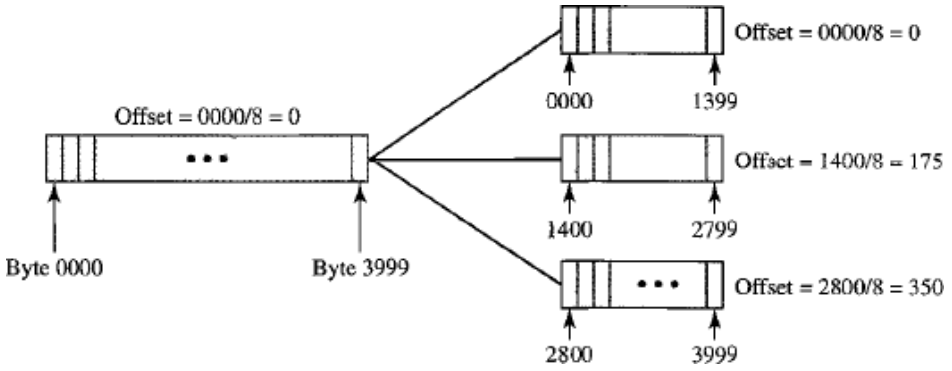
To make the IPv4 protocol independent of the physical network, the designers decided to make the maximum length of the IPv4 datagram equal to 65,535 bytes. This makes transmission more efficient if we use a protocol with an MTU of this size. We must divide the datagram to make it possible to pass through these networks. This is called fragmentation.

Fields Related to Fragmentation

Identification: This 16-bit field identifies a datagram originating from the source host. The combination of the identification and source IPv4 address must uniquely define a datagram as it leaves the source host.

Flags: This is a 3-bit field. The first bit is reserved. The second bit is called the do not fragment bit. If its value is 1, the machine must not fragment the datagram. If it cannot pass the datagram through any available physical network, it discards the datagram.

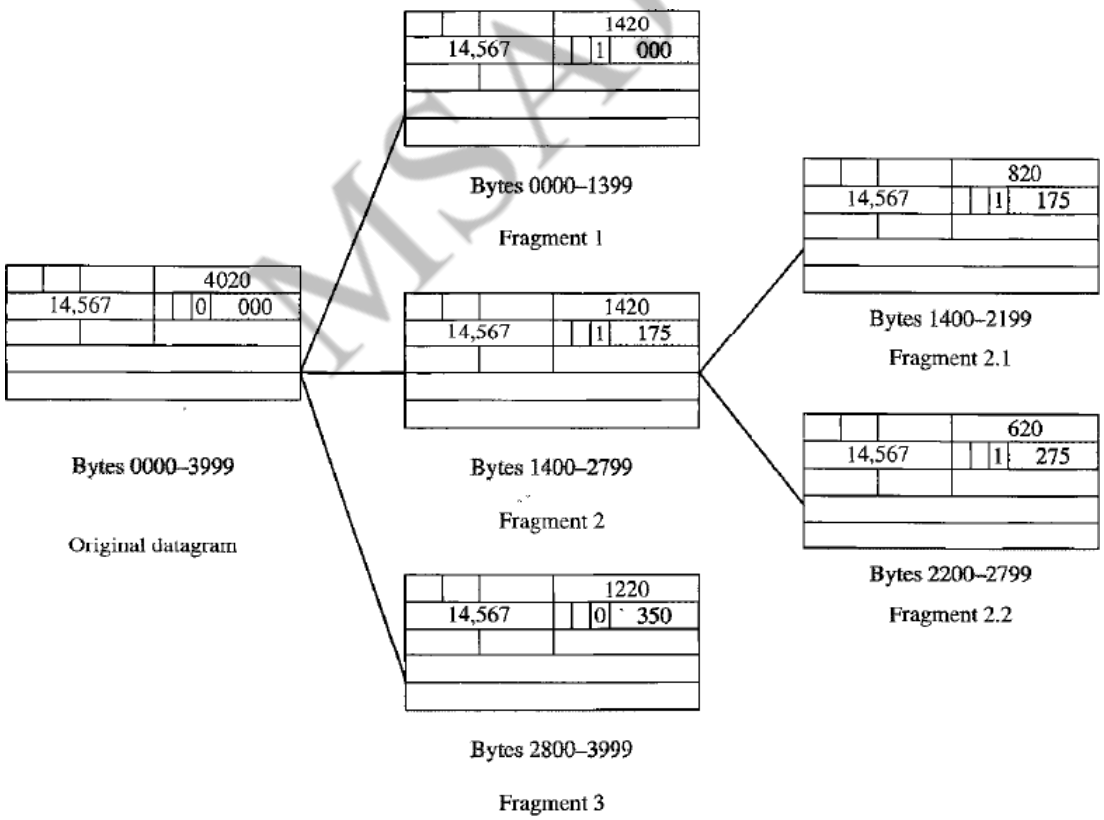
Fragmentation offset. This 13 bit field shows the relative position of this fragment with respect to the whole datagram. It is the offset of the data in the original datagram measured in units of 8 bytes.



Fragment example

The size of 4000 bytes fragmented into three fragments.

Fragment	Off Set address
0 to 1399	0/8 = 0
1400 to 2799	1400/8=175
2800 to 3999	2800/8 =350



Detailed fragmentation example

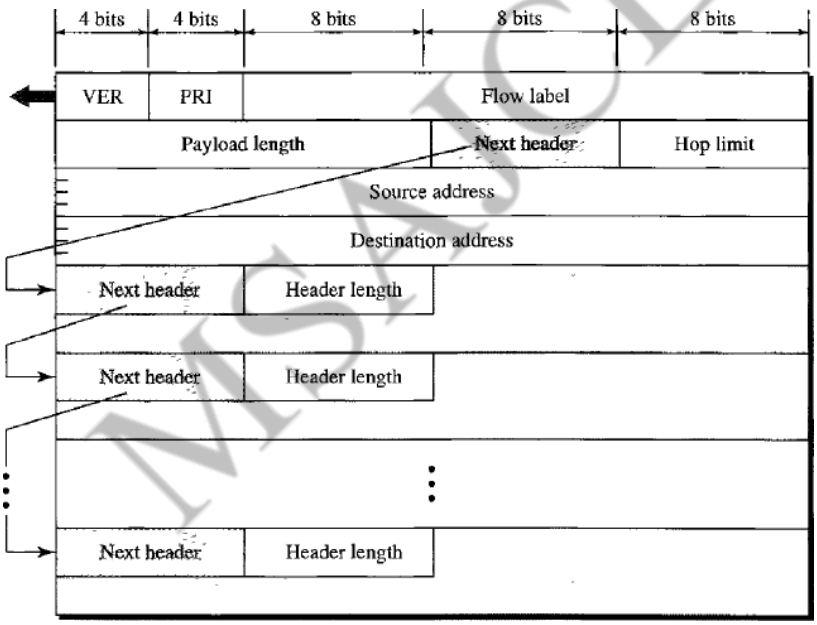
IPv6

IPv4 has some deficiencies (listed below) that make it unsuitable for the fast growing internet.

- Despite all short-term solutions, such as subnetting, class less addressing, and NAT, address depletion is still a long-term problem in the internet .
- The internet must accommodate real time audio and video transmission. This type of transmission requires minimum delay strategies and reservation of resources not provided in the IPv4 design.
- The internet must accommodate encryption and authentication of data for some applications. No encryption or authentication is provided by IPv4.

The format and the length of the IP address were changed along with the packet format. Related protocols, such as ICMP, were also modified. Other protocols in the network layer, such as ARP, RARP and IGMP, were either deleted or included in the ICMPv6 protocol. Routing protocols, such as RIP and OSPF, were also slightly modified to accommodate these changes. Communication experts predict that IPv6 and its related protocols will soon replace the current IP version.

The adoption of IPv6 has been slow. The season is that the original motivation for its development, depletion of IPv6 addresses, has been remedied by short-term strategies such as class less addressing and NAT. However the fast spreading use of the internet, and new services such as mobile IP, IP telephony, and IP capable mobile telephony, may eventually requires the total replacement of IPv4 with IPv6.



Format of an IPv6 datagram

Advantages

The next generation IP, or IPv6 has some advantages over IPv4 that can be summarized has follows:

- **Larger address space**. An IPv6 address is 128 bits long. compared with the 32 bit address of IPv4, this is a huge (2^{96}) increase in the addresses space.
- **Better header format**. IPv6 uses a new header format in which option are separated from the base header and inserted, when needed, between the base header and the upper layer data. This simplifies and speeds up the routing process because most of the options do not need to be checked by routers.
- **New options**. IPv6 has new options to allow for additional functionalities.
- **Allowance for extensions**. IPv6 is designed to allow the extension of the protocol if required by new technologies or applications.

- **Support for resource allocation**. In IPv6, the type-of-service field has been removed, but a mechanism (called flow label) has been added to enable the source to request special handling of the packet. This mechanism can be used to support traffic such as real time audio and video.
- **Support for more security**. The encryption and authentication options in IPv6 provide confidentiality and integrity of the packet.

Packet format

Each packet is composed of the mandatory base header followed by the payload. The payload consists of two parts: optional extension headers and data from an upper layer. The base header occupies 40 bytes, whereas the extension headers and data from the upper layer contain upto 65,535 bytes of information.

Base Header

The base header with its eight fields . These fields are as follows:

- **Version.** This 4-bit field defines the version number of the IP. For the IPv6, the value is 6.
- **Priority**. The 4-bit priority fields defines the priority of the packet with respect to traffic congestion.
- **Flow label**. The flow label is a special handling for a particular flow of data.
- **Payload length**. Length of the IP datagram excluding the base header.
- **Next header**. The next header is an 8-bit field defining the header that follows the base header in the datagram. The next header is either one of the optional extension headers used by IP or the header of an encapsulated packet such as UDP or TCP. Each extension header also contains this field. Note that this field in version 4 is called the protocol.
- **Hop limit**. The 8-bit hop limit field serves the same purpose as the TTL field in IPv4.
- Source address. The source address field is a 16-byte (128 bit) internet address that identifies the original source of the datagram.
- **Destination address**. The destination address field is a 16-byte (128 bit) internet address that usually identifies the final destination of the datagram. However, if source routing is used, this field contains the address of the next router.

IPv6 Addresses

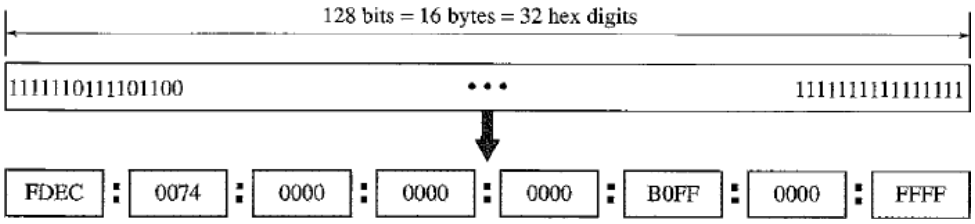
Despite all short-term solutions, such as class less addressing, dynamic host configuration protocol (DHCP), and NAT, address depletion is still a long-term problem for internet. This and other problems in the IP protocol itself, such as lack of accommodation for real-time audio and video transmission, and encryption and authentication of data for some applications, have been the motivation for IPv6. In this section, we compare the address structure of IPv6 to IPv4.

Structure

An IPv6 address consists of 16-bytes (octets); it is 128 bits long.

Hexadecimal colon notation

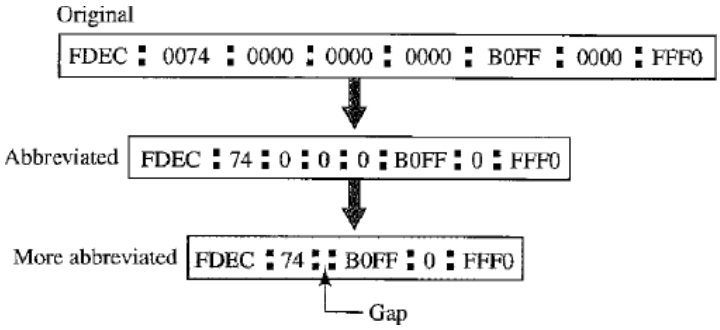
To make addresses more readable, IPv6 specifies hexadecimal colon notation. In this notation, 128-bits is divided into 8 sections, each two bytes in length. Two bytes in hexadecimal notation requires four hexadecimal digits. Therefore, the address consists of 32 hexadecimal digits, with every four digits separated by a colon.



IPv6 address in binary and hexadecimal colon notation

Abbreviation

Although the IP address, even in hexadecimal format, is very long, many of the digits are zeros. In this case, we can abbreviate the address. The leading zeros of a section (four digits between two colons) can be omitted. Only the leading zeros can be dropped, not the trailing zeros.



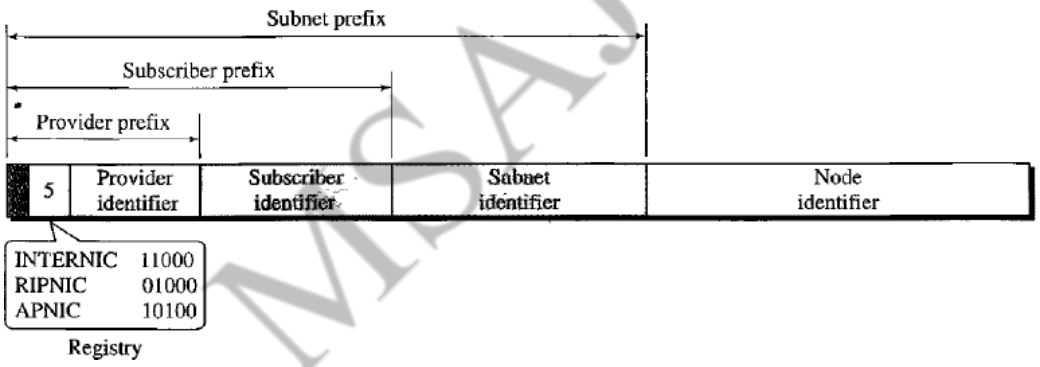
Abbreviated IPv6 addresses

Addresses space

IPv6 has a much larger address space; 2^{128} addresses are available.

Unicast Addresses

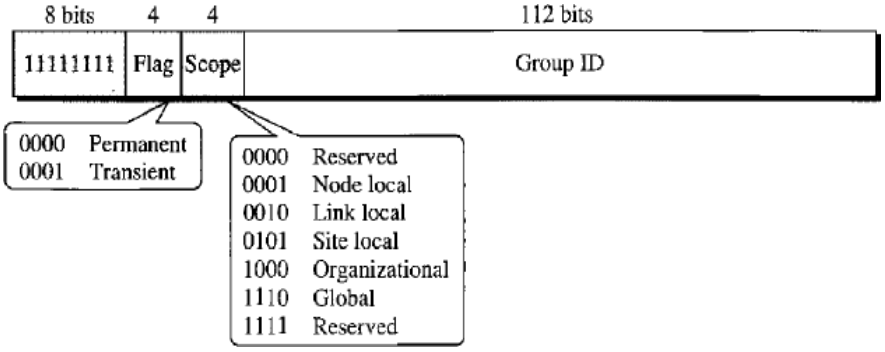
A unicast addresses defines a single computer. The packet sent to a unicast address must be delivered to that specific computer. IPv6 defines two types of unicast addresses: geo graphically based and provider-based. We discuss the second type here; the first type is left for future definition. The provider-based address is generally used by a normal host as a unicast address.



Prefixes for provider-based unicast address

Multicast address

Multicast addresses are used to define a group of hosts instead of just one. A packet sent to a multicast address must be delivered to each member of the group.



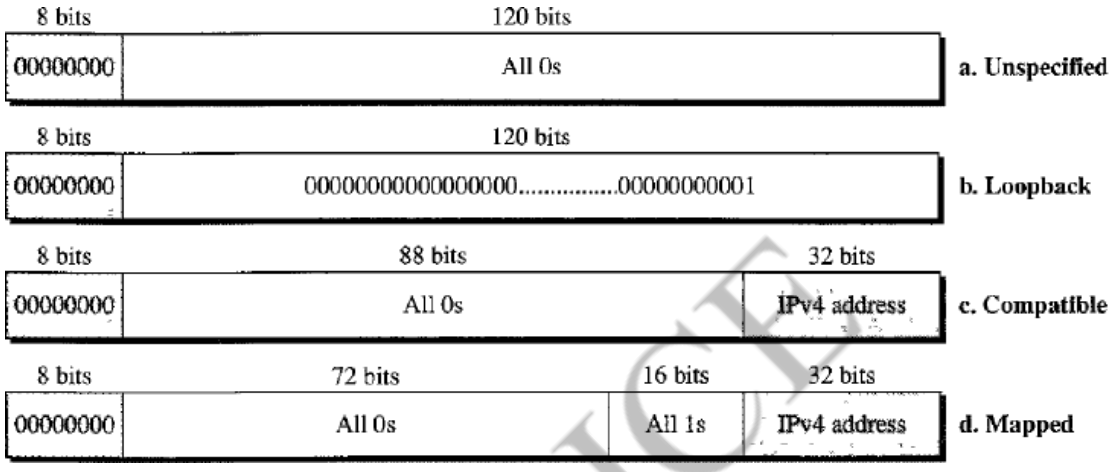
Multicast address in IPv6

Anycast addresses

IPv6 also defines anycast addresses. An anycast addresses, like a multicast addresses, also defines a group of nodes. However, a packet destined for an anycast address is delivered to only one of the members of the anycast group. The nearest, one (the one with the shortest route. Although the definition of an anycast address is still debatable, one possible is to assing an anycast address to all routers of an ISP that covers a large logical area in the internet. The routers outside the ISP deliver a packet destined for the ISP to the nearest ISP router. No block is assigned for anycast addresses.

Reserved Addresses

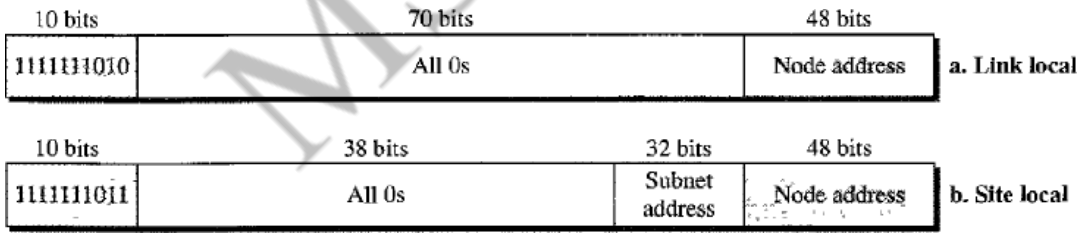
Another category in the addresses space is the reserved addresses. These addresses start with eight 0's (type prefix is 00000000).



Reserved addresses in IPv6

Local Addresses

These addresses are used when an organization wants to use IPv6 protocol without being connected to the global internet. In otherwords, they provide addressing for private network. Nobody outside the organization can send a message to a nodes using these addresses.

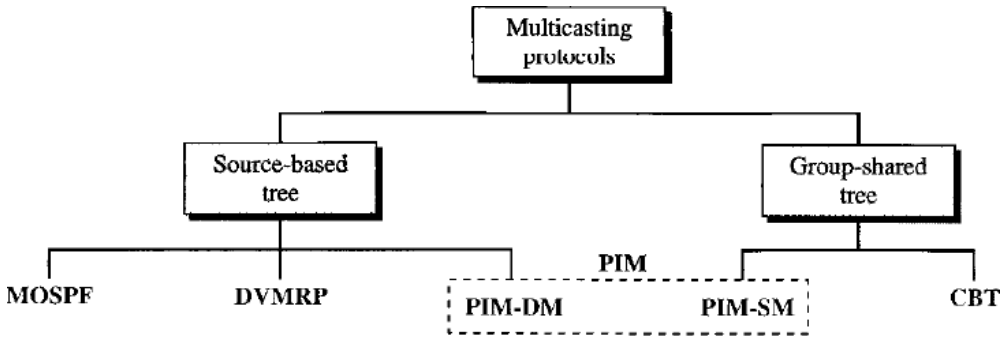


Local Addresses in IPv6

Comparison between IPv4 and IPv6 packet Headers

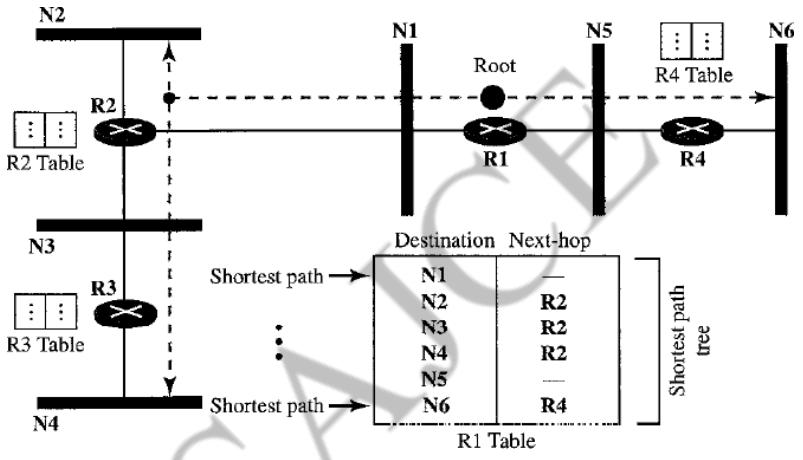
1. The header length field is eliminated in IPv6 because the length of the header is fixed in this version.
2. The service type field is eliminated in IPv6. The priority and flow label fields together take over the function of the service type field.
3. The total length field is eliminated in IPv6 and replaced by the payload length field.
4. The identification, flag, and offset fields are eliminated from the base header in IPv6. They are included in the fragmentation extension header.
5. The TTL field is called hop limit in IPv6.
6. The protocol field is replaced by the next header field.
7. The header checksum is eliminated because the checksum is provided by upper-layer protocols; it is therefore not needed at this level.
8. The option fields in IPv4 are implemented as extension headers in IPv6.

MULTICAST ROUTING



Unicast Routing:

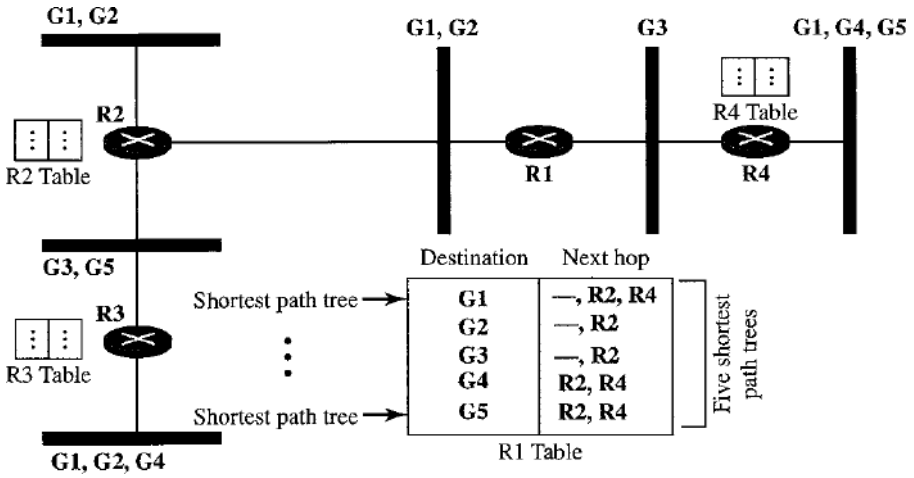
In unicast routing, when a router receives a packet to forward, it needs to find the shortest path to destination of the packet. The router consults its routing table for the particular destination. The next-hop entry corresponding to the destination is the start of the shortest path. The router knows the shortest path for each destination, which means that the router has a shortest path tree to optimally reach all destinations.



Multicast routing:

When a router receives a multicast packet, the situation is different from when it receives a unicast packet. A multicast packet may have destinations in more than one network. Forwarding of a single packet to members of a group requires a shortest path tree. If we have n groups, we may need n shortest path trees. We can imagine the complexity of multicast routing. Two approaches have been used to solve the problem: source- based trees and group- shared trees.

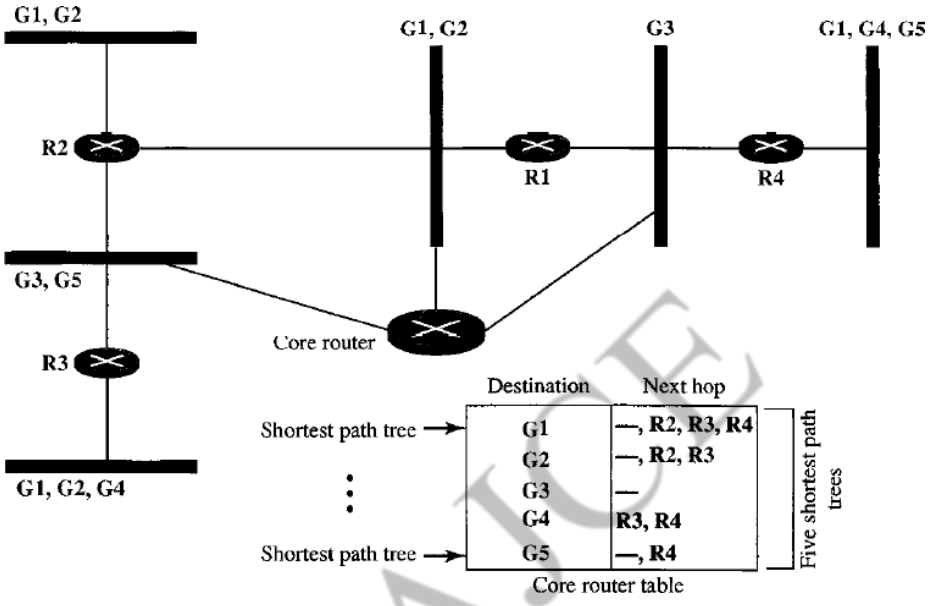
Source- based tree:



In the source-based tree approach, each router needs to have one shortest path tree for each group. The shortest path tree for a group defines the next hop for each network that has loyal member(s) for that group. We assume that we have only five groups in the

domain: G1, G2, G3, G4, and G5. At the moment G1 has loyal members in four networks, G2 in three, G3 in two, G4 in two, and G5 in two. We have shown the names of the groups with loyal members on each network. There is one shortest path tree for each group; therefore there are five shortest path trees for five groups. If router R1 receives a packet with destination addressG1, it needs to send a copy of the packet to the attached network, a copy to router R2, and a copy to router R4 so that all members of G1 can receive a copy. In this approach, if the number of groups is m, each router needs to have m shortest path trees, one for each group. We can imagine the complexity of the routing table if we have hundreds or thousands of groups. However, we will show how different protocols manage to alleviate the situation.

Group shared tree:



In the group shared tree approach, instead of each router having m shortest path trees, only one designated router, called the center core or rendezvous router, takes the responsibility of distributing multicast traffic. The corehas m shortest path trees in its routing table. The rest of the router In the domain have none. If a router receives a multicast packet, it encapsulates the packet in a unicast packet and send it to the ore router .The core router removes the multicast packet from its capsule and consults its routing table to route the packet.

MULTICAST LINK STATE ROUTING

We discussed unicast link state routing in section 22.3. We said that each router creates a shortest path tree by using Dijkstra's algorithm. The routing table is a translation of the shortest path tree. Multicast link state routing is a direct extension of unicast routing and uses a sources –based tree approach. Although unicast routing is quite involved, the extension to multicast routing is very simple and straight forward.

Recall that in unicast routing, each node needs to advertise the state of its links. For multicast routing, a needs to revise the interpretation of state. A node advertises every group which has any loyal member on the link. Here the meaning of state is "what groups are active on this link". The information about the group comes from IGMP. Each router running IGMP solicits the hosts on the link to find out the membership status.

When a router receives all these LSPs, it creates n (n is the number of groups) topologies, from which n shortest path trees are made by using Dijkstra`s algorithm. So each router has a routing table that represents as many shortest path trees asa there are groups.

The only problem with this protocol is the time and space needed to create and save the many shortest path trees. The solution is to create the trees only when needed. When a router receives a packet with a multicast destination address, it runs the Dijkstra algorithm to calculate the shortest path tree for that group. The result can be cached in case there are additional packets for destination.

MOSPF (Multicast open shortest path first)

MOSPF protocol is an extension of the OSPF protocol that uses multicast link state routing to create source- based trees. The protocol requires a new link state update packet to associate the unicast address of a host with the group address or addresses the host is sponsoring. This packet is called the group- membership LSA. In this way, we can include in the tree only the hosts (using their unicast address) that belongs to a particular group. In other words, we make atree that contains all the hosts belonging to agroup, but we use the unicast address of the host in the calculation. For efficiency, the router calculates the shortest path trees on demand (when it receives the first multicast packet). In addition, the tree can be saved in cache memory for future use by the same source/group pair. MOSPF is a data-driven protocol; the first time an MOSPF router sees a diagram with agiven source and group address, the router constructs the Dijkstra shortest path tree.

MULTICAST DISTANCE VECTOR: DVMRP

Multicast distance vector routing

Unicast distance vector routing is very simple; extending it to support multicast routing is complicated. Multicast routing does not allow a router to send its routing table to its neighbors. The ideas is to create a table from scratch by using the information from the unicast distance vector tables.

Multicast distance vector routing uses source- based trees, but the router never actually makes a routing table. When a router receives a multicast packet, it forwards the packet as though it is consulting a routing table. We can say that the shortest path tree is evanescent. After its use (after a packet is forwarded) the table is destroyed.

To accomplish this, the multicast distance vector algorithm uses a process based on four decision-making strategies. Each strategy is built on its predecessor. We explain them one by one and see how each strategy can improve the shortcomings of the previous one.

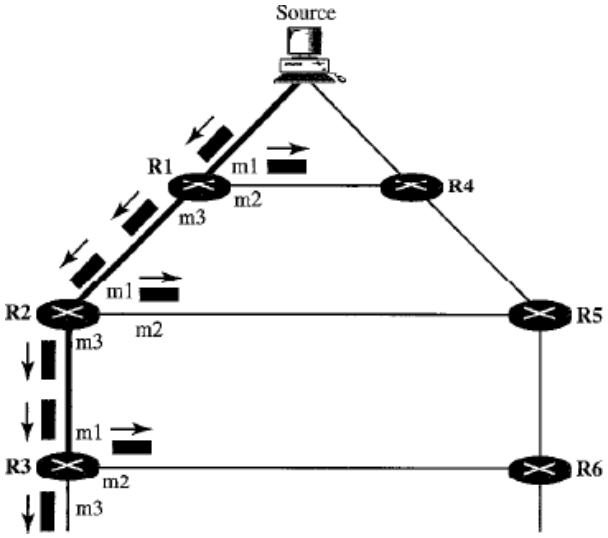
Flooding is the first strategy that comes to mind. A router receives a packet and, without even looking at the destination group address, sends it out from every interface except the one form which it is received. Flooding accomplishes the first goal of multicasting: every network with active members receives the packet. However, so will networks without active members. This is a broadcast, not a multicast. There is another problem: it creates loops. A packet that has left the router may come back again from another interface or the same interfaceand be forwarded again. Some flooding protocols keep a copy of the packet for a while and discard any duplicates to avoid loops. The next strategy, reverse path forwarding, corrects this defect.

Reverse Path Forwarding (RPF) is a modified flooding strategy. To prevent loops, only one copy is forwarded; the other copies are dropped. In RPF, a router forwards only the copy that has traveled the shortest path from the source to router. To find this copy, RPF uses the unicast routing table. The router receives a packet and extracts the source address (a unicast address). It consults its unicast routing table as though it wants to send a packet to the source address. The routing table tells the router the next hop. If the multicast packet has just come from the hop defined in the table, the packet has traveled the shortest path from the source to the router because the shortest path is reciprocal in unicast distance vector routing protocols. If the path from A to B is the shortest, the it is also the shortest from B to A. The router forwards the packet if it has traveled from the shortest path; it discards it otherwise.

This strategy prevents loops because there is always one shortest path from the source to the router. If a packet leaves the router and comes back again, it has not traveled the shortest path.

The shortest path tree as calculated by routers R1, R2, and R3 is shown by a thick line. When R1 receives packet from the source through the interface m1, it consults its routing table and finds that the shortest path from R1 to the source is through the interface m1. The packet is forwarded. However if the copy of the packet has arrived through the interface m2, it is discarded because m2 is not does not define the shortest path from R1 to the source. The story is the same with R2 and R3.you may wonder what may happens if a copy of a packet that arrives at the m1 interface of R3,travels through R6,R5,R2,and then entersR3 through interface m1.this interface is the correct interface for R3. Is the copy of the packet forwarded? the answer is that this scenario never happens because when the

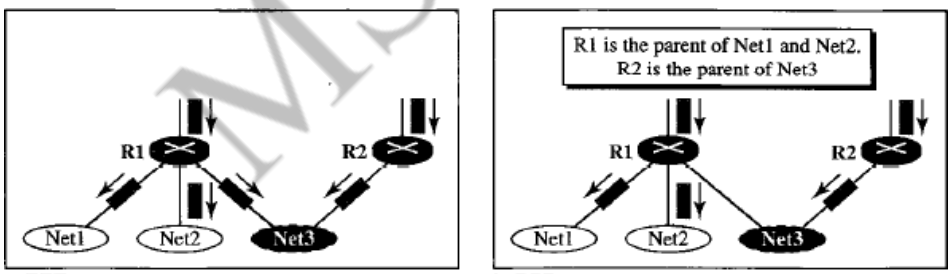
packet goes from R5 to R2,it will be discarded by R2 and never reaches R3. The upstream routers thus preventing confusion for he download stream routers.



Reverse Path Broadcasting (RPB): RPF guarantees that each networks receive a copy of the multicast packet without formation of loops. However, RPF does not guarantee that each network receives only one copy; a network may receive two or more copies. The reason is that RPF is not based other destination address (a group of address);forwarding is based on the source address. To visualize the problem, let us look.

Net3 in this figure receives two copies of the packet even through each router just sends out one copy from each interface. There is duplication because a tree has not been made; instead of a tree we have a graph. Net3 has two parents: routers R2 and R4.
To eliminate duplication, we must define only one parent router for each network. We must have this restriction: a network can receive a multicast packet from a particular source only through a destination parent router.

Now the policy is clear. For each sources, the router sends the packet only out of those interfaces for which it is the designated parent. This policy is called reverse path broadcasting (RPB). RPB guarantees that the packet reaches every network and that very network receives only one copy.



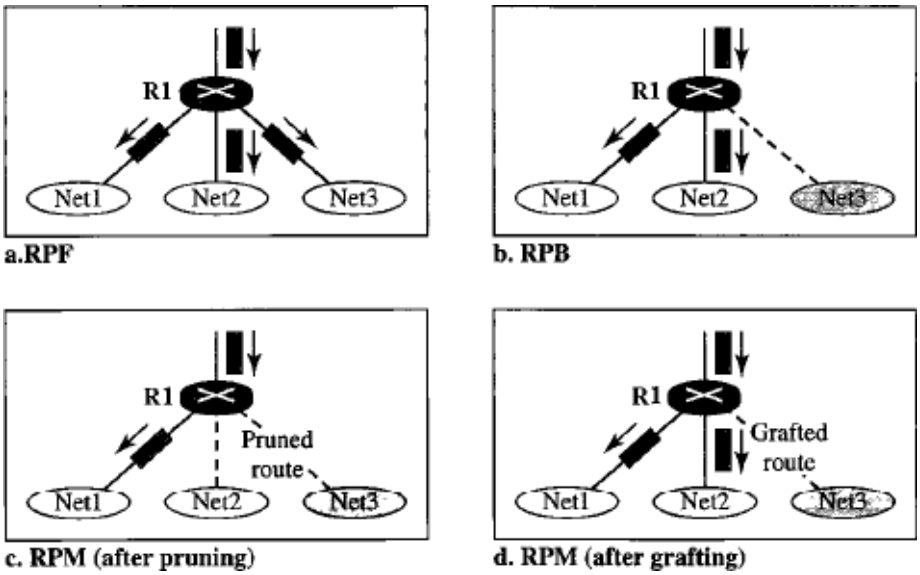
RPF Versus RPB

The reader may ask how the designated parent is determined. The designated parent router can be the router with the shortest path to the source. Because the routers periodically send updating packets to each other (in RPS), they can easily determine which router in the neighbourhood has the shortest path to the source (when interpreting the source as the destination). If more than one router qualifies, the router with the smallest IP address is selected.

RPB creates a shortest path broadcast tree from the source to each destination.it guarantees that each destination receives one and only one copy of the packet.

Reverse Path Multicasting (RPM): As you may have noticed, RPB does not multicast the packet, it broadcast it. This is not efficiency, the multicast packet must reach only those networks that have active members for that particular group. This is called reverse path

multicasting(RPM). To convert into broadcasting to multicasting, protocol uses two procedures, pruning and grafting.



The designated parent router of each network is responsible for holding the membership information. This is done through the IGMP protocol. The process starts when a router connected to a network finds that there is no interest in a multicast packet. The router sends a prune message to the upstream router can stop sending multicast messages for this group through that interface. Now if this router receives prune messages from all downstream routers, it, in turn, sends a prune message to its upstream router.

What if aleaf router (a router at the bottom of the tree) has sent a prune message but suddenly realizes, through IGMP, that one of its networks is again interested in receiving the multicast packet? It can send a graft message. The graft message forces the upstream router to resume sending the multicast messages.

RPM adds pruning and grafting to RPB to create a multicast shortest path tree that supports dynamic membership changes.

DVMRP:

The distance vector multicast routing protocol (DVMP) is an implementation of multicast distance vector routing. It is a source-based routing protocol, based on RIP.

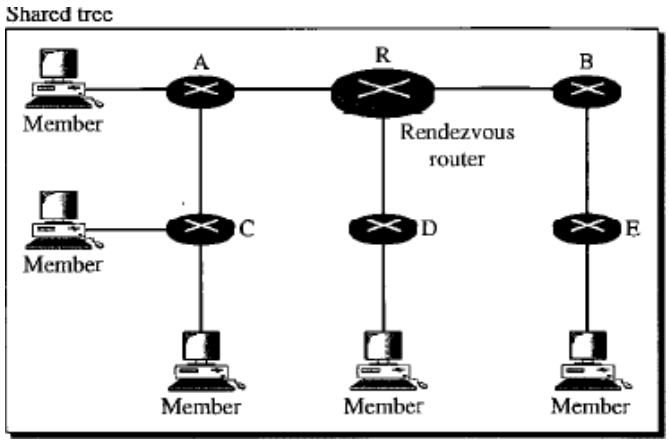
CBT

The core- based tree (CBT) protocol is a group-shared protocol that uses a core as the root of the tree. The autonomous system is divided into regions, and a core (centre router or rendezvous router) is chosen for each region.

Formation of a tree: After the rendezvous point is selected, every router is informed of the unicast address of the selected router. Each router then sends a unicast join message (similar to a grafting message) to show that it wants to join the group. This message passes through all the routers that are located between the sender and rendezvous router. Each intermediate router extracts the necessary information from the message, such as the unicast address of the sender and the interface through which the packet has arrived, and forwards the message to the next router in the path. When the rendezvous router has received all join messages from every member of the group, the tree is formed. Now every router knows its upstream router(the router that leads to the root) and the downstream, router (the router that leads to the leaf).

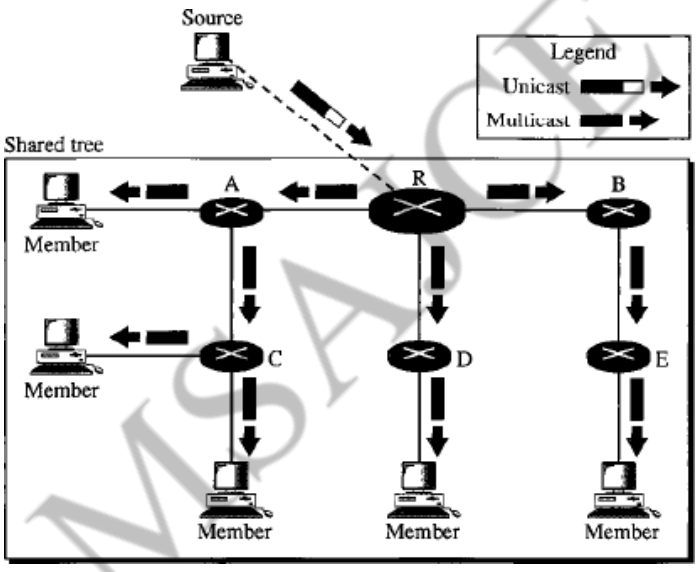
If a router wants to leave the group, it sends a leave message to its upstream router. The upstream router removes the link to that router from the tree and forwards the message to its upstream router and so on.

The render may have noticed two differences between DVMRP and MOSPF, on one hand, and CBT, on the other. First, the tree for first two is made from the root up; the tree for CBT is formed from the leaves down. Second, in DVMRP, the tree is first made(broadcasting and then pruned ;in CBT, there is no tree at beginning; the joining(grafting)gradually makes the tree.



Group shared tree

Sending Multicast Packets: after formation of the tree, any source (belonging to the group or not) can send amulticast packet to all the members of the group. It simply sends the packet to the rendezvous router, using the unicast address of the rendezvous router; the rendezvous router distributes the packet to all the member of the group. Note that the source host can be any of the hosts inside the shared tree or any host outside the shared tree.



Selecting the rendezvous router

This approach is simple except for one point. How do we select a rendezvous router to optimize the process and multicasting as well? Several methods have been implemented. However, this topic is beyond the scope of this book, and we leave it to more advanced books.

In summary, the core-based tree(CBT) is a group-shared tree, centre-based protocol using one tree per group. One of the routers in the tree is called the core. A packet is sent from the source to members of the group following this procedure:

1. The source, which may or may not be part of the tree, encapsulates the multicast packet inside a unicast packet with the unicast destination address of the core and sends it to the core. This part of delivery is done using a unicast address; the only recipient is the core router.
2. The core decapsulates the unicast packet and forward it to all interested interfaces.
3. Each router that receives the multicast packet, in turn, forwards it to all interested interfaces.

In CBT, the source sends the multicast packet (encapsulated in unicast packet) to the core router. The core router decapsulates the packet and forwards it to all interested interfaces.

PIM

Protocol independent multicast (PIM) is the name given to two independent multicast routing protocols: protocol independent multicast, dense mode (PIM-DM) and protocol independent multicast, sparse mode(PIM-SM). Both protocols are unicast protocol-dependent, but the similarity ends here. We discuss each separately.

PIM-DM: PIM-DM is used when there is a possibility that each router is involved in multicasting (dense mode). In this environment, the use of a protocol that broadcasts the packet is justified because almost all routers are involved in the process.

PIM-DM is used in a dense multicast environment, such as a LAN.

PIM-DM is a source- based tree routing protocol that uses RPF and pruning and grafting strategies for multicasting. Its operation is like that of DVMRP; however, unlike DVMRP, it does not depend on a specific unicasting protocol. It assumes that the autonomous system is using a unicast protocol and each router has a table that can find the outing interface that has an optimal path to a destination. This unicast protocol can be a distance vector (RIP) or link state protocol (OSPF).

PIM-DM uses RPF and pruning and grafting strategies to handle multicasting, however, it is independent of the underlying unicast protocol.

PIM-SM: PIM-SM is used when there is a slightly possibility that each router is involved in multicasting (sparse mode). In this environment, the use of a protocol that broadcasts the packet is not justified; a protocol such as CBT that uses a group-shared tree is more appropriate.

PIM-SM is used in a sparse multicast environment such as WAN.

PIM-SM is a group-shared tree routing protocol that has a rendezvous point (RP) as the source of the tree. Its operation is like CBT; however, it is simpler because it does not require acknowledgment from a join message. In addition, it creates a backup set of RPs for each region to cover RP failures.

One of the characteristics of PIM-SM is that it can switch from a group-shared tree strategy to a source-based tree strategy when necessary. This can happen if there is a dense area of activity far from the RP. That area can be more efficiently handled with a source-based tree strategy instead of a group-shared tree strategy.

AREAS, METRICS

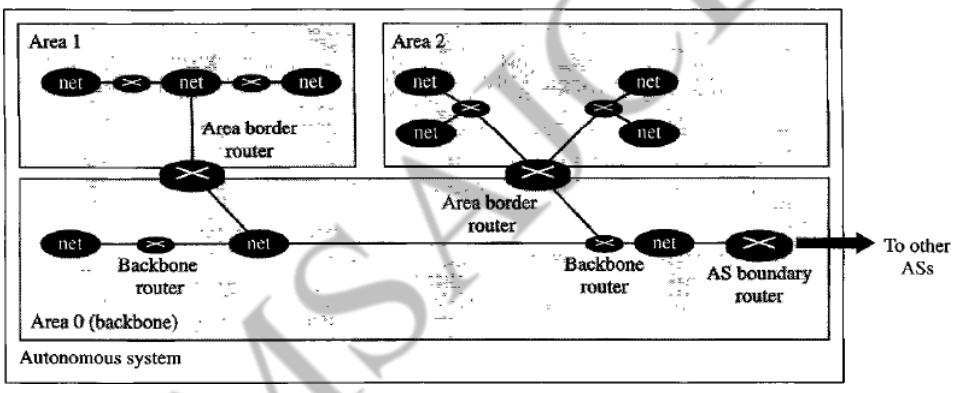
The open shortest path first or OSPF protocol is an intradomain routing protocol based on link state routing. Its domain is also an autonomous system.

Areas: to handle routing efficiently and in a timer manner, OSPF divides an autonomous system into areas. An area is acollection of networks, hosts, and routers all contained within an autonomous system. An autonomous system can be divided into many different areas. All networks inside an area must be connected.

Routers inside an area flood the area with routing information. At the border of an area, special routers called area border routers summarize the information about the area and send it to another areas. Among the areas inside an autonomous system is a special area called backbone. In other words, the backbone; all the areas inside an autonomous system must be connected to the backbone. In another words, the backbone serves as a primary area and the other areas as secondary areas. This does not mean that the routers within areas cannot be connected to each other, however. The routers inside the backbone are called the backbone routers. Note that a backbone router can also be an area border router.

If, because of some problem, the connectivity between a backbone and an area is broken, a virtual link between routers must be created by an administrator to allow continuity of the functions of the backbone as the primary area.

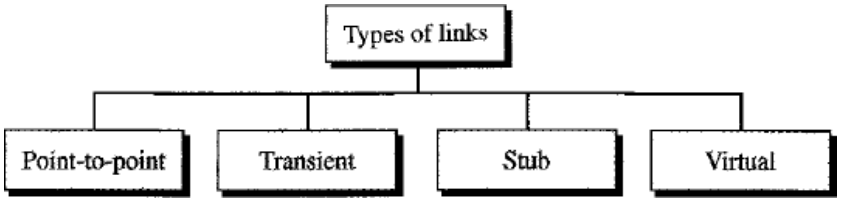
Each area has an area identification. The area identification of the backbone is zero.



Areas in an autonomous system

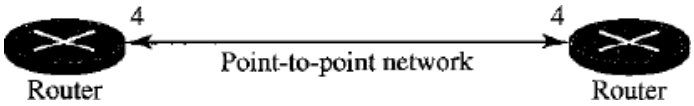
Metric: The OSPF protocol allows the administrator to assigns a cost, called the metric, to each route. The metric can be based on a type of service (minimum delay, maximum throughput, and so on). As a matter of fact, a router can have multiple routing tables, each based on a different type of service.

Types of links: in OSPF terminology, a connection is called a link. Four types of links have been defined: point to point, transient, stub, and virtual.



A point-to-point link connects two routers without any other host or router in between. In other words, the purpose of the link (network) is just to connect the two routers. An example of this type of link is two routers connected by a telephone line or a T line. There is no need to assign a network address to this type of link. Graphically, the routers are

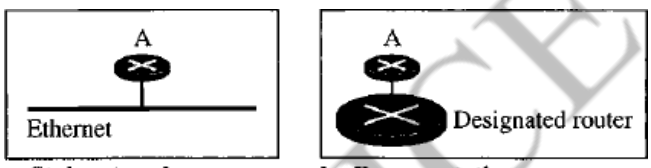
represented by nodes, and the link is represented by a bidirectional edge connecting nodes. The metric, which are usually the same, are shown at the two ends, one for each direction. In other words, each router has only one neighbour at the other side of the link.



Point to point link

A **transient link** is a network with several routers attached to it. The data can enter through any of the routers and leave through any routers. All LANs and some WANs with two or more routers are of this type. In this case, each router has many neighbors. Router A has routers B,C,D, E as neighbors. Router B has routers A,C,D, and E as neighbors.

A **stub link** is a network that is connected to only one router. The data packets enter the network through this single router and leave the network through this same router. This is a special case of the transient network. We can show this situation using the router as a node and using the designated router for the network. However, the link is only one-directional from the router to the network.



a. Stub network **b. Representation**

When the link between two routers is broken, the administration may create a **virtual link** between them, using a longer path that is probably goes through several routers.

UNIT - 4

USER DATA PROGRAM PROTOCOL (UDP)

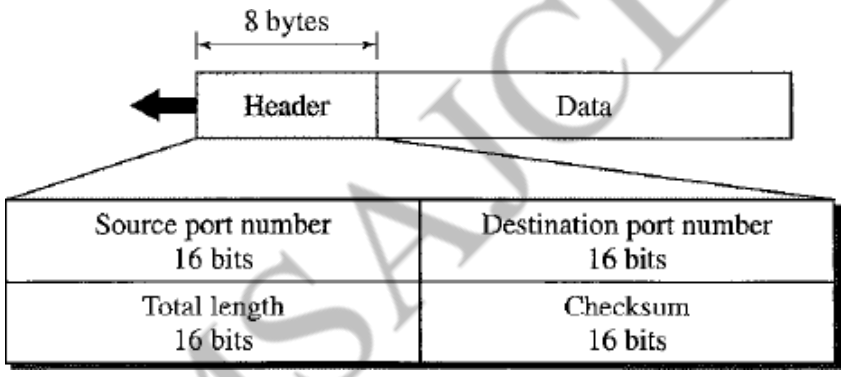
The user data(UDP) is called a connectionless, unreliable transport protocol. It does not add anything to the services of IP expect to provide process-to-process communication instead of host-to-host communication. Also, it performs very limited error checking.

If UDP is so powerless, why would a process want to use it? With the disadvantage come some advantages. UDP is a very simple protocol using a minimum of overhead. If a process wants to send a small message and does not care much about reliability, it can use UDP. Sending a small message by using UDP takes much less interaction between the sender and receiver than using TCP or SCTP.

USER DATAGRAM:

UPD packets, called user datagram, have a fixed- size header of 8 bytes.
The fields are as follows:

Source Port Number: this is port used by the process running on the source host. It is a16 bites long, which means that the port number can rate from 0 to 65,535. If the source host is the client (a client sending a request),the port number, in most cases, is an ephemeral port number requested by the process and chosen by the UDP software running on the source host. If the source host is the server (a server sending a response), the port number, in most cases, is a well-known port number.



Destinatin Port Number: this is the port number used by the process running the host. It is also 16 bits long. If the destination host is the server (a client sending a request), the port number, in a number, in most cases, is a well-known port number. If the destination host is the client (a server sending a response), the port number, in most cases, is an ephemeral port number. In such a cases, the server copies the ephemeral port number it has received in the request packet.

Length: This is a 16-bit field that defines the total length of the user datagram, header plus data. The 16 bits can definesa total length of 0 to 65,535 bytes. However, the total length needs to much less because a UDP user datagram is stored in an IP datagram with a total length of 65,535 byte.

The length field in a UDP user datagram is actually not necessary. A user datagram is encapsulated in an IP address is encapsulated in an IP datagram. There is a field in the IP datagram that defines the total length. There is another field in the IP datagram that defines the length of the header. So if we subtract the value of the second field from the first, we can deduce the length of the UDP datagram that is encapsulated in IP datagram.

UDP length= IP length – IP header's length

However, the designers of the UDP protocol felt that it was more efficient for the designation UDP to calculate the length of the data from the information provided in the UDP user datagram rather than ask the IP software to supply this information. We should

remember that when the IP software delivers the UDP user datagram to UDP layer, it has already dropped the IP header.

Checksum:
This field is used to detect errors over the entire user datagram (header plus data)

UDP OPERATION
UDP uses concepts common to the transport layer.

Connectionless services
As mentioned previously, UDP provides a connectionless service. This means that each user datagram sent by UDP is an independent datagram. There is no relationship between the different user datagrams even if they are coming from the same source process and going to the same destination program. The user datagrams are not numbered. Also, there is no connection established and no connection termination, as is the case for TCP. This means that each user datagram can travel on a different path.

One of the ramifications of being connectionless is that the process that the process that uses UDP cannot send a stream of data to UDP and except UDP to chop them into different related user datagrams. Instead each request must be small enough to fit into one user datagram. Only those processes sending short message should use UDP.

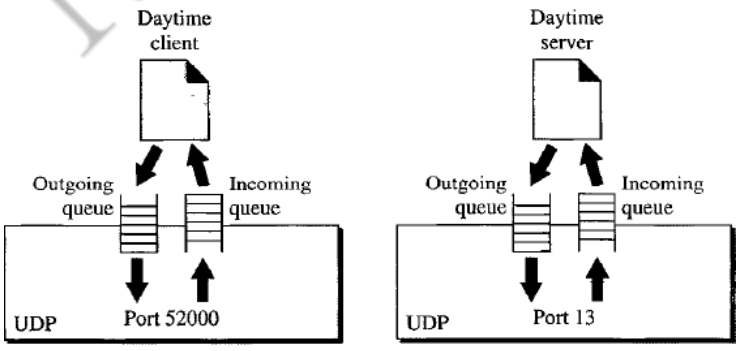
Flow and error control
UDP is a very simple, unreliable transport protocol. There is no flow control and hence no window mechanism. The receiver may overflow with incoming messages.

There is no error control mechanism inUDP except for the checksum. This means that the sender does not know if amessage has been lost or duplicated. When the receiver detects an error through the checksum, the user datagram is silently discarded.

The lack of flow control and error control means that the process using UDP should provide these mechanisms.

Encapsulation And Decapsulation
To send a message from one process to another, the UDP protocol encapsulates and decapsulatesmessages in an IP diagram.

Queuing



At the client site, when a process starts, it requires a port number form the operating system. Some implementations create both an incoming and an outgoing queue associated with each process. Other implementations create only an incoming queue associated with each process.

If a process wants to communicate with multiple processes, it obtains only one port number and eventually one outgoing and one incoming queue. When the process terminates the queues are destroyed.

The Client process can send messages to the outgoing queue by using the source port number specified in the request. UDP removes the messages one by one and, after adding the UDP header, delivers them to IP.

When a message arrives for a client, UDP checks to see if an incoming queue has been created for the port number specified in the destination port number field of the user datagram. If there is such queue, UDP sends the received user datagram to the end of the queue. If there is no such queue, UDP discards the user datagram and asks the ICMP protocol to send a port unreachable message to the server.

At the server site, the mechanism of creating queues is different. In its simplest form, a server asks for incoming and outgoing queues, using its well-known port, when it starts running. The queues remain open as long as the server is running.

When a message arrives for a server, UDP checks to see if an incoming queue has been created for the port number specified in the destination port number field of the user datagram. If there is such a queue, UDP sends the received user datagram to the end of the queue. If there is no such queue, UDP discards the user datagram and asks the ICMP protocol to send a port unreachable message to the client. All the incoming messages for one particular server, whther coming from the same or a different client, are sent to the same queue.

When a server wants to respond to a client, it sends messages to the outgoing queue, using the source port number specified in the request. UDP removes the messges one by one and, after adding the UDP header, delivers them to IP.

USES OF UDP:

The following lists some uses of the UDP protocol:

UDP is suitable for a process that requires simplest requires simple request-response communication with the little concern for the flow and error control. It is not usually used for the process such as FTP that needs to send bulk data.

UDP is suitable for a process with internal flow and error control mechanisms. For example, the trivial first transfer protocol (TFTP) process includes flow and error control. It can easily use UDP.

UDP is suitable transport protocol for multicasting. Multicasting capability is embedded in the UDP software but not in the TCP software.

UDP is used for management process such as SNMP.

UDP is used for some route updating protocols such as routing information protocol (RIP).

TCP

TCP uses flow and error control mechanisms at the transport level. TCP is called connection oriented, reliable transport protocol. Its acts connection-oriented and reliability features to the services of IP.

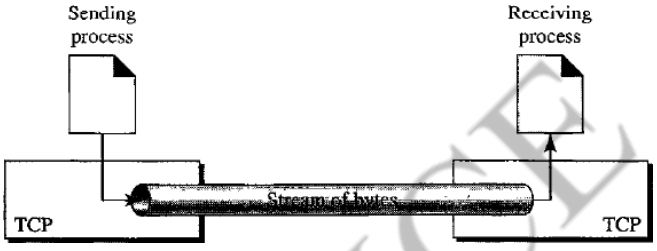
Process- to-process communication

Like UDP, TCP provides process-to process communication using port numbers.

Stream delivery service

TCP, unlike UDP, is a stream-oriented protocol. In UDP, a process (an application program) sends messages, with predefined boundaries, to UDP for delivery. UDP for delivery UDP adds its own header to each of these messages and delivers them to IP for transmission. Each message from the process is called a user datagram and becomes, eventually, one IP datagram. Neither IP nor UDP recognizes any relationship between datagrams.

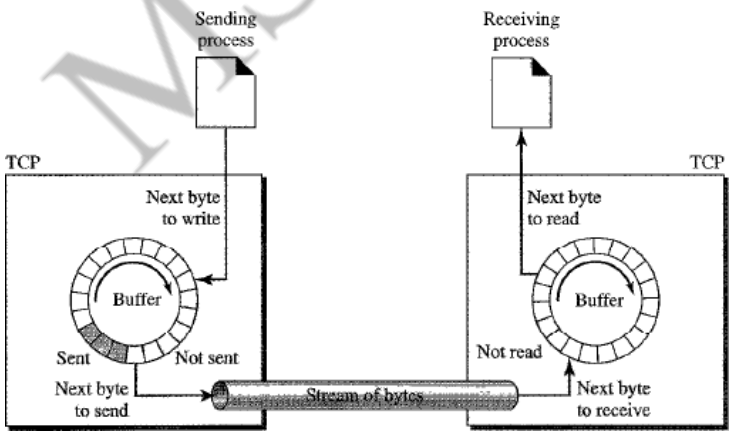
TCP, on the other hand, allows the sending process to deliver data as a stream of bytes and allows the receiving process to obtain data as a stream of bytes. TCP creates an environment in which the two process to obtain data as a stream of bytes. TCP creates an environment in which the two processes seem to be connected by an imaginary "tube" that carries their data across the internet. The sending process produces (writes to) the stream of bytes, and the receiving process consumes (reads from) them.



Stream delivery

Sending and Receiving Buffers

Because the sending and receiving processes many not write or read data at the same speed, TCPneeds buffers for storage. There are two buffers, the sending buffer and receiving buffer, one for each direction. One way implementation a buffer is to use a circular array of 1- byte locations. For simplicity, we have shown two buffers of 20 bytes each; normally the buffers are hundreds or thousands of bytes, depending on the implementation. We also show the buffers as the same size, which is not always the case.

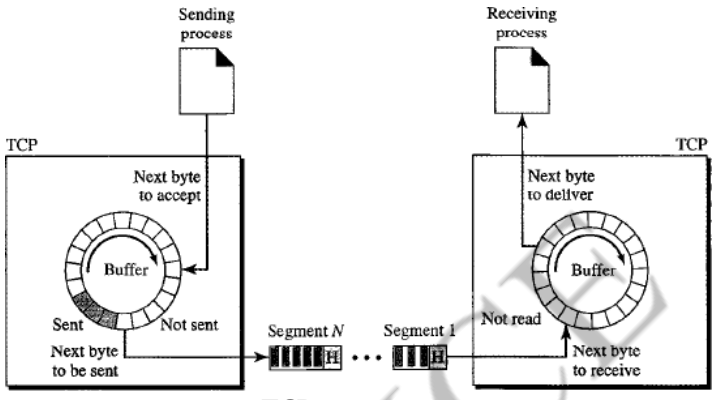


Sending and Receiving buffers

At the sending site, the buffer has three types of chambers. The white section contains empty chambers that can be filled by the sending process (producer). The gray area holds bytes that have been sent but yet acknowledgment. The colored area contains bytes to be sent by sending TCP. TCP may be able to send only part of this colored section. This could be due to the slowness of receiving process or perhaps to congestion in the network. Also note that after the bytes in the gray chambers are acknowledged, the chambers are recycled and available for use by the sending process. This is why we show a circular buffer.

The operation of the buffer at the receiving site is simpler. The circular buffer is divided into two areas. The white area contains empty chambers to be filled by bytes receiving from the network. The colored sections contain receiving bytes that can be read by the receiving process. When a byte is read by the receiving process, the chamber is recycled and added to the pool of empty chambers.

Segments: Although buffering handles the disparity between the speed of producing and consuming processes, we need one more step before we can send data. The IP layer, as a service provider for TCP, needs to send data in packets, not as a stream of bytes. At the transport layer,TCP groups a number of bytes together into a packet called a segment. TCP adds a header to each segment (for control purposes) and delivers the segment to the IP layer for transmission. The segments are encapsulated in IP datagrams and transmitted. This entire operation is transparent to the receiving process. Later we will see that segments may be received out of order, lost, or corrupted and resent. All these are handled by TCP with the receiving process unaware of any activities.



TCP segments

Note that the segments are not necessarily the same size. In reality, segments carry hundreds, if not thousands, of bytes.

Full-duplex communication

TCP offers full-duplex service, in which data can flow in both directions at the same time. Each TCP then has sending and receiving buffer, and segments move in both directions.

Connections-oriented services:

TCP, unlike UDP, is a connection-oriented protocol. When a process at a site A wants to send and receive data from another process at site B, the following occurs:

1. The two TCPs establish a connection between them.
2. Data are exchanged in both directions.
3. The connection is terminated.

Note that this is a virtual connection, not a physical connection. The TCP segment is encapsulated in aIP datagram and can be send out of order, or lost, or corrupted, and then resend. Each may use a different path to reach the destination. There is no physical connection. TCP creates a stream-oriented environment in which it accepts the responsibility of delivering the bytes in order to the other site. The situation is similar to creating a bridge that spans multiple islands and passing all the bytes from one island to another in one single connection.

Reliable Services

TCP is a reliable transport protocol. It uses an acknowledgement mechanism to check the safe and sound arrival of data. We will discuss feature future in the section on error control.

TCP features

To provide the services mentioned in the previous section, TCP has several features that are briefly summarized in this section.

Numbering System

Although the TCP software keeps track of the segments being transmitted or received, there is no field for a segment number value in the segment header. Instead, there are two fieldfor a segment number value in the sequence number and the acknowledgment number. These two fields refer to the byte number and not the segment number.

Byte Number: TCP numbers all the data bytes that are transmitted in a connection. Numbering is independent in each direction. When TCP receives bytes of data from a process, it stores them in sending buffer and numbers them. The numbering does not necessarily start from 0.instead, TCP generates a random number between0 and 2^32-1 for the number of the first byte. For an example, if the random number happens to be 1057 and the total data to be sent are 6000 bytes, the bytes are numbered from 1057 to 7056. We will see that byte numbering is used for flow and error control.

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

Acknowledgement number:
when a connection is established, both patties can send and receive data at the same time. Each party numbers the bytes, usually with a different starting byte number. The sequence in each direction shows the number of the first byte carried by the segment. Each party also uses an acknowledgement number to confirm the bytes it has received.

Flow control:
TC, unlike UDP, provides flow control. The receiver of the data controls the amount of data that are to be send by the sender. This is done to prevent the receiver from being over-whelmed with data. The numbering system allows TCP to use a byte-oriented flow control.

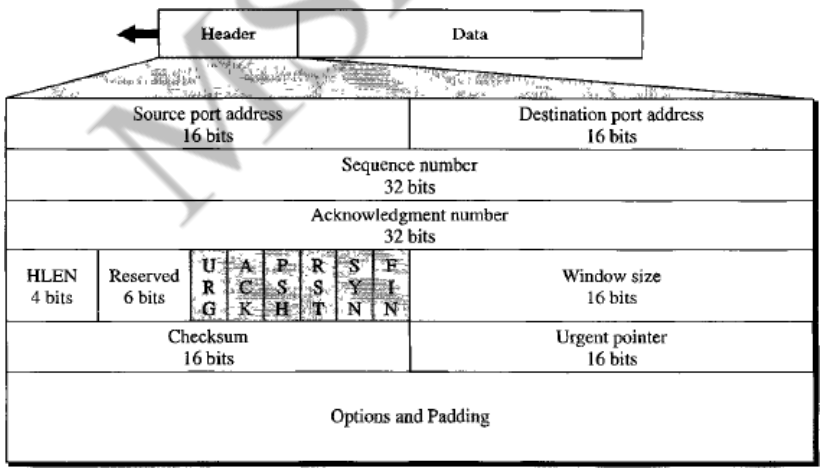
ERROR CONTROL:
To provide reliable services, TCP implements an error control mechanism. Although error control considers a segment as the unit of data for error detection (loss or corrected segments), error control is byte–oriented.

CONGESTION CONTROL
TCP, unlike UDP, takes into account congestion in the network. The amount of data sent by a sender is not only controlled by the receiver (flow control), but is also determined by the level of congestion in the network.

Segment

A packet in TCP is called a segment.

Format



TCP segment format

The segment consists of a 20-to 60-byte header, followed by data from the application program. The header is 20 bytes if there are no options and up to 60 bytes if it contains options.

Source port address
This is a 16-bit field that defines the port number of the application program in the host that is sending the segment. This serves the same purpose as the source port address in the UDP header.

Destination port address
This is a 16-bit field that defines the port number of the application program in the host that is receiving the segment. This serves the same purpose as the destination port address in the UDP header.

Sequence number
This 32-bit field defines the number assigned to the first byte of data contained in this segment. TCP, is a stream protocol. To ensure connectivity, each byte to be transmitted is numbered. The sequence number tells the destination which byte in this sequence comprises the first byte in the segment. During connection establishment, each party uses a random number generator to create an initial sequence number (ISN), which is usually different in each direction.

Acknowledgment number
This is 32-bit field defines the byte number that the receiver of the segment is expecting to receive from the other party. If the receiver of the segment has successfully received byte number x from the other party, it defines x+1 as the acknowledgment number. Acknowledgment and data can be piggybacked together.

Header length
This 4-bit field indicates the number of 4-byte words in the TCP header. The length of the header can be between 20 and 60 bytes. Therefore, the value of this field can be between 5 (5X 4=20) and 15 (15 X 4=60).

Reserved This is a 6 –bit field reserved for future use.

Control
This field defines 6 different control bits or flags. One or more bits can be set at a time.

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

These bits enable flow control, connection establishment and termination, connection abortion, and the mode of data transfer in TCP.

Window size This field defines the size of the window, in bytes, that the other party must maintain. Note that the length of this field is 16 bits, which means that the maximum size of the window is 65,535 bytes. This value is normally referred to as the receiving window (rwnd) and is determined by the receiver. The sender must obey the dictation of the receiver in this case.

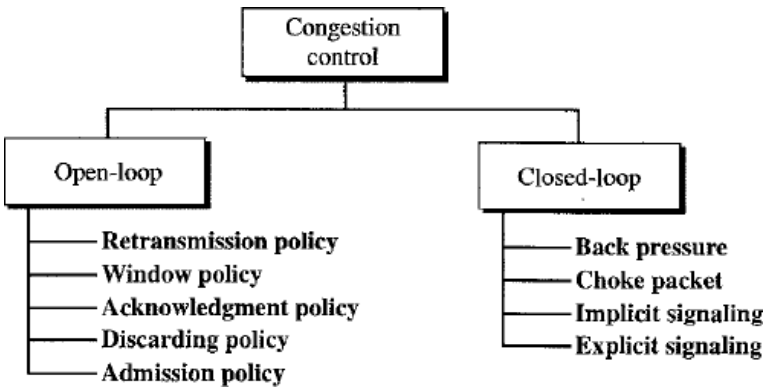
Checksum This is 16-bit field contains the checksum. The calculation of the checksum for TCP follows the same procedure as the one described for UDP. However, the inclusion of the checksum in the UDP datagram is optional, whereas the inclusion of the checksum for TCP is mandatory. The same pseudoheader, serving the same purpose, is added to the segment. for the TCP pseudo header, the value for the protocol field is 6.

Urrgent Pointer This 16-bit field, which is valid only if the urgent flag is set, I used when the segment contains urgent data. It defines the number that must be added to the sequence number to obtain the number of the last urgent byte in the data section of the segment.

Options
There can be up to 40 bytes of optional information in the TCP header.

CONGESTION CONTROL

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



Open-loop congestion control

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

Retransmission policy

Retransmission is sometimes unavoidable. If the sender feels that a sent packet is lost or corrupted, the packet needs to be retransmitted. Retransmission in general may increase congestion in the network. However, a good retransmission policy can prevent congestion. The retransmission policy and retransmission timers must be designed to optimize efficiency and at the same time prevent congestion. For example, the retransmission policy used by TCP is designed to prevent or alleviate congestion.

Window policy

The type of window at the sender may also affect congestion. The selective repeat window is better than the Go-back-N window for congestion control. In the Go-Back-N window, when the timer for a packet times out, several packets may be resent, although some may have arrived safe and sound at the receiver. This duplication may make the congestion worse. The selective repeat window, on the other hand, tries to send the specific packets that have been lost or corrupted.

Acknowledgment policy

The acknowledgment policy imposed by the receiver may also affect congestion. If the receiver does not acknowledge every packets it receives, it may slow down the sender and help prevent congestion. Several approaches are used in this case. A receiver may send an acknowledgment only if the packet to be sent or a special timer expires. A receiver may decide to acknowledge only N packets at a time. We need to know that the acknowledgments are also part of the load in a network. Sending fewer acknowledgment means imposing less load on a network.

Discarding policy

A good discarding policy by the routers may prevent congestion and at the same time may not harm the integrity of the transmission. For example, in audio transmission, if the policy is to discard less sensitive packets when congestion is likely to happen, the quality of sound is still preserved ar alleviated.

Admission policy

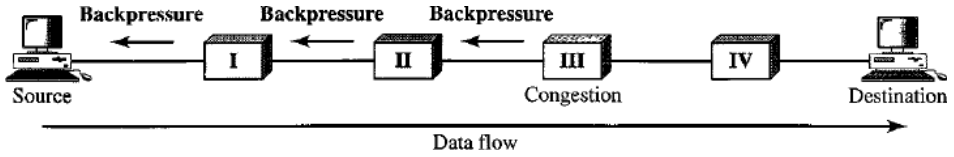
An admission policy, which is a quality -of- service mechanisms, can also prevent congestion in virtual-circuit networks. Switches in a flow first check the resource requirement of a flow before admitting it to the network . A router can deny establishing a virtual circuit if there is congestion in the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

Closed-loop congestion control

Closed-loop congestion control mechanisms try to alleviate congestion after it happens. Several mechanisms have been used by different protocols. We describe a few of them here.

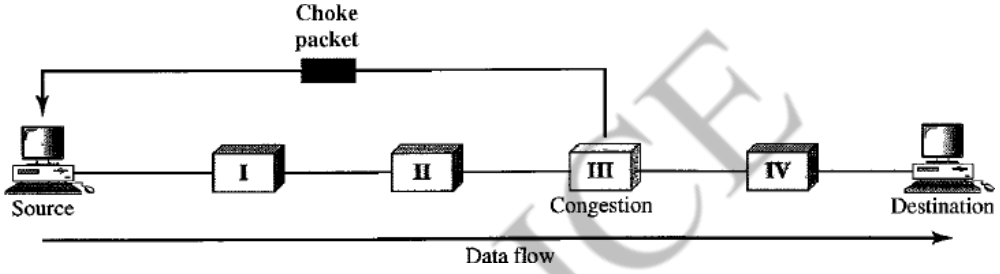
Backpressure

The technique of backpressure refers to a congestion control mechanisms in which a congested node stops receiving data from the immediate upstream node or nodes. This may cause the upstream node or nodes to become congested, and they, in turn, reject data from their upstream nodes or nodes. And so on. Backpressure is a node-to-node congestion control that starts with a node and propagates, in the opposite direction of data flow, to the source. The backpressure technique can be applied only to virtual circuit networks, in which each node knows the upstream node from which a flow of data is coming.



Node III has more input data than it can handle.it drops some packets in its input buffer and informs node II to slow down. Node II, in turn may be congested because it is slowing down the output flow of data. If node II is congested, it informs node I to slow down, which in turn may create congestion. Note that the pressure on node III is moved backward to the source to remove the congestion.

Choke packet



A choke packet is a packet sent by a node to the source to inform it of congestion. Note the difference between the backpressure and choke packet methods. In backpressure, the warning is from one node to its upstream node, although the warning may eventually reach the source station. In choke packet method, the warning is from the router, which has encountered congestion, to the source station directly. The intermediate nodes through which the packet has travelled are not warned. When a router in the internet is overwhelmed with IP datagrams, it may discard some of them; but it informs the source host, using a source quench ICMP message. The warning message goes directly to the source station; the intermediate routers, and does not take any action.

Implicit signalling

In implicit signalling, there is no communication between the congested node or nodes and the source. The source guesses that there is a congested somewhere in the network from other symptoms. For example, when a source sends several packets and there is no acknowledgment for a while, one assumption is that the network is congested. The delay in receiving an acknowledgment is interpreted as congestion in the network; the source should slow down.

Explicit signalling

The node that experiences congestion can explicitly send a signal to the source or destination. The explicit signalling method, however, is different from the choke packet method. In choke packet method, a separate packet is used for this purpose; in the explicit signalling method, the signal included in the packets that carry data. Explicit signalling, as in frame relay congestion control, can occur in either the forward or backward direction.

Backward signalling

A bit can be set in a packet moving in the direction opposite to the congestion. This bit can warn the source that there is congestion and that it needs to slow down to avoid the discarding of packets.

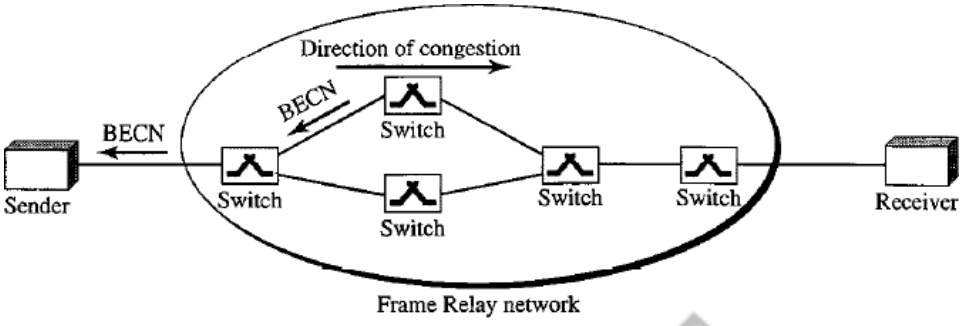
Forward signalling

A bit can be set in a packet moving in the direction to the congestion. This bit can warn the destination that there is congestion. The receiver in this case can use policies, such as slowing down the acknowledgments, to alleviate the congestion.

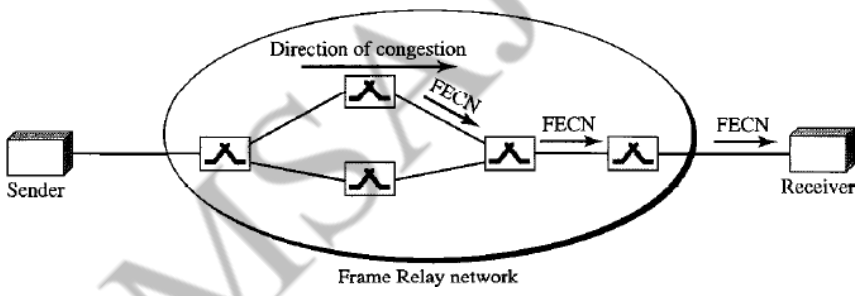
CONGESTION AVOIDANCE

For congestion avoidance, the frame relay protocol uses 2 bits in the frame to explicitly warn the source and the destination of the presence of congestion.

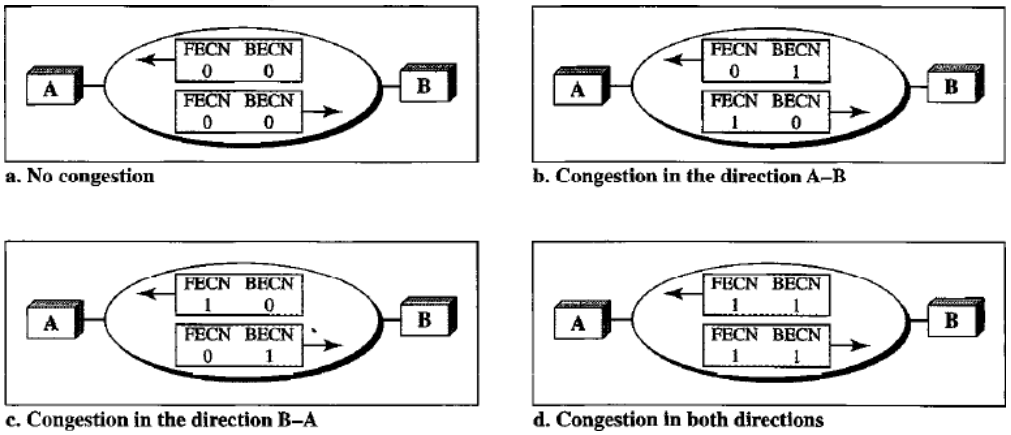
BECN: The backward explicit congestion notification (BECN) bit warns the sender of congestion in the network. One might ask how this is accomplished since the frames are travelling away from the sender. In fact, there are two methods. The switch can use response frames from the receiver (full-duplex mode), or else the switch can use a predefined connection (DLCI=1023) to send special frames for this specific purpose. The sender can respond to this warming by simply reducing the data rate.



FECN: The forward explicit congestion notification (FECN) bit is used to warn the receiver of congestion in the network. It might appear that receiver cannot do anything to relieve the congestion. However, the frame relay protocol assumes that the sender and receiver are communicating with each other and are using some type of flow control at a higher level. For example, if there is an acknowledgment mechanisms at this higher level, the receiver can delay the acknowledgment, thus forcing the sender to slow down.



When two endpoints are communicating using a frame relay network, four situations may occur with regard to congestion.



Four cases of congestion

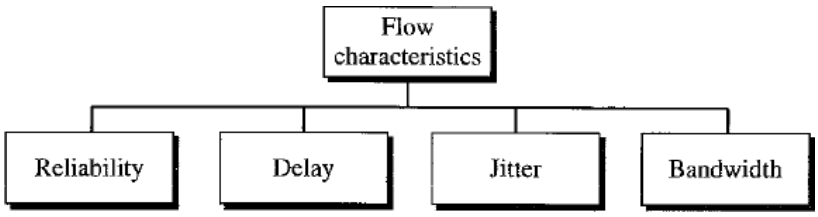
It does not guarantee the minimum of a service, such as bandwidth, to application such as real - time audio and video. If such an application accidently gets extra bandwidth, it may be detrimental to other applications, resulting in congestion.

QUALITY OF SERVICE

Quality of service (QOS) is an internetworking issue that has been discussed more than defined.we can informally define the quality of service as something flow seeks to attain.

Flow characteristics

Traditionally, four types of characteristics are attributed to a flow: reliability, delay, jitter , and bandwidth.



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

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

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

For applications such as audio and video, the first case is completely acceptable: the second case is not. For these applications, it does not matter if the packets arrive with a short or long delay as long as the delay is the same for all packets. For this application, the second case is not acceptable.

Jitter is defined as the variation in the packet delay. High jitter means the difference between delays is large: low jitter means the variation is small.

If the jitter is high, some action is needed in order to use the received data.

Bandwidth : Different applications need different bandwidths. In video conferencing we need to send millions of bits per second to refresh a colour screen while the total number of bits in an e-mail may not reach even a million.

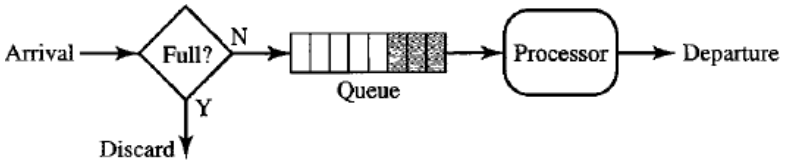
TECHNIQUES TO IMPROVE QoS

Scheduling

Packets from different flows arrive at a switch or router for processing. A good scheduling techniques treats the different flows in a fair and appropriate manner. Several scheduling techniques are designed to improve the quality of service. FIFO queuing, priority queuing and weighted fair queuing.

FIFO Queuing

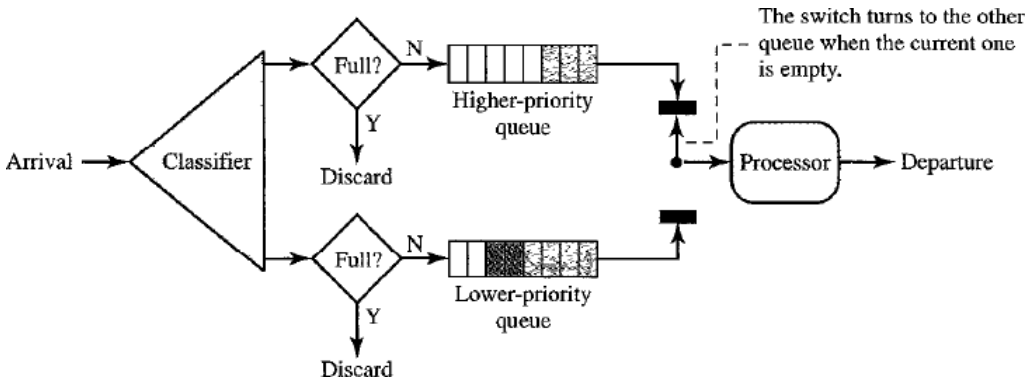
In first-in, first-out (FIFO) queuing, packets wait in a buffer (queue) until the node(router or switch) is ready to process them. If the average arrival rate is higher than the average processing rate, the queue will fill up and new packets will be discarded. A FIFO queue is familiar to those who have had to wait for a bus at a bus stop.



FIFO queue

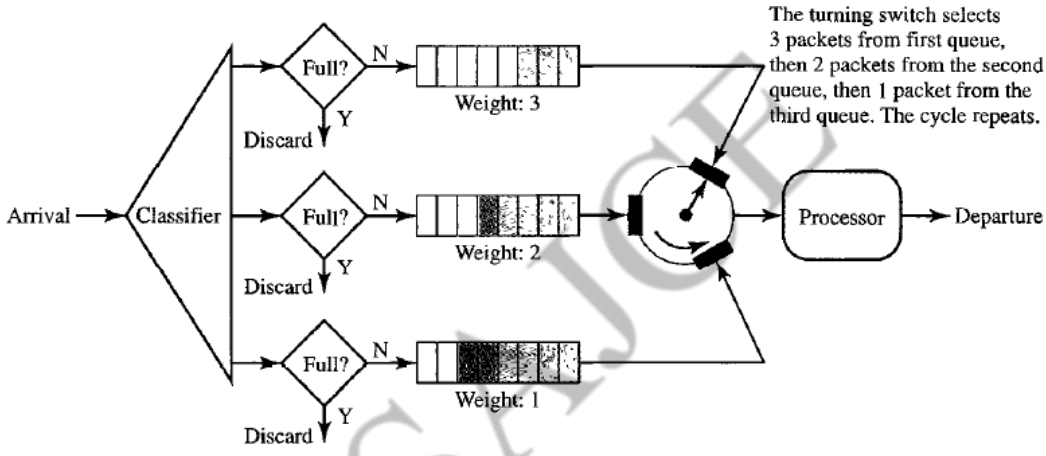
Priority queuing

In priority queuing, packets are first assigned to a priority class. Each priority class has its own queue. The packets in the highest-priority queue are processed first. Packets in the lowest-priority queue are processed last. Note that the system does not stop serving a queue until is empty.



A priority queue can provide better QoS than the FIFO queue because higher-priority traffic, such as multimedia, can reach the destination with less delay. However, there is a potential drawback. If there is a continuous flow in a higher-priority queue, the packets in the lower priority queue will never have a chance to be processed. This is a condition called starvation.

Wieght fair queuing

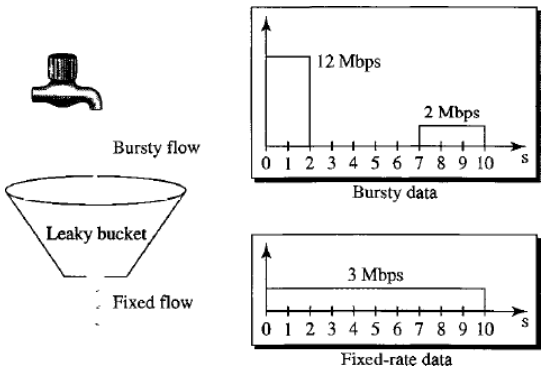


A better scheduling is weighted fair quenching. In this technique, the packets are still assigned to different class and admitted to different queues. The queues, however, are weighted based on priority of the queues; higher priority means a higher weight. The system packets in each queue based on the corresponding weight. For example, if the weights are3,2, and 1, three packets are processed from the second queue, and one from the third queue. If the system does not impose priority on the classes, all weights can be equal. In this way, we have fair queuing with priority.

TRAFFIC SHAPING

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

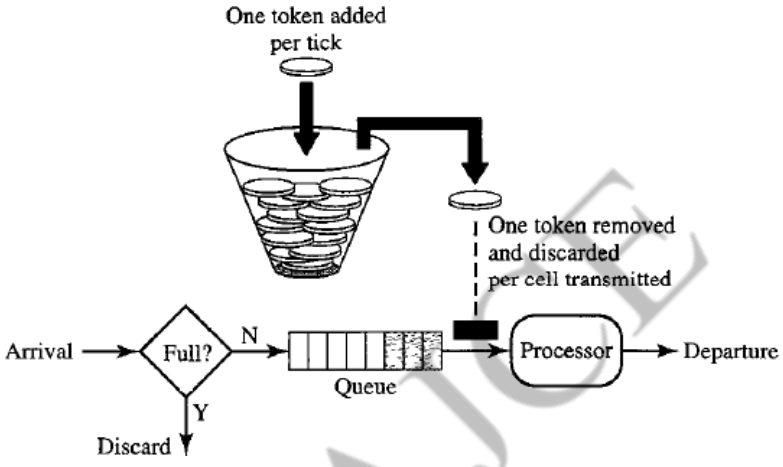
Leaky bucket:



If a bucket has as a small hole at the bottom, at the water leaks from the buckets at a constant rate as long as there is water in the bucket. The rate at which the water leaks does not depend on the rate at the water is input to the bucket unless the bucket is empty. The input rate can vary, but the output rate remains constant. Similarly, in networking, a technique called leaky bucket can smooth out bursty traffic. Bursty chunks are stored in the bucket and sent out at an average rate.

The host sends a burst of data at a rate of 12Mpps for 2s, for a total of 24 Mbits of data. The host is silent for 5 s and then sends data at a rate of 2MBPS r 3 s, for a total of 6 Mbits of data. In all, the host has sent 30 Mbits of data in 10 s. the leaky bucket smooth the traffic by sending out data at the rate of 3 Mbps during at the same 10 s. without the leaky bucket, the beginning burst may have hurt the network by consuming more bandwidth than is set aside for this host. We can also see that the leaky bucket may prevent congestion.

Token bucket:



The leaky bucket is very restrictive. It does not credit an idle host. For example, If a host is not sending for a while, its bucket becomes empty. Now if the host has bursty data, the leaky bucket allows only an average rat. The time when the host was idle is not taken into account. On the other hand, the token bucket algorithm allows idle host to accumulate credit for the future in the form of tokens. For each tick of the clock, the system sends n tokens to the bucket. The system removes one token for every cell(or byte) of data sent. For example, if n is 100 and the host is idle for 100 ticks, the bucket collects 10,000 tokens. Now the host can consume all these tokens In one ticket with 10.000 cells, or the host takes 1000 ticks with 10 cells per tick. In other words, the host can send bursty data as long as the bucket is not empty.

The token bucket can easily be implemented with a counter. The token is initialized to zero each time, a token is added, the counter is incremented by 1. Each time a unit of data is sent, the counter is decremented by 1. When the counter is zero, the host cannot send data.

Combining Token Bucket and Leaky Bucket

The two techniques can be combined to credit an idle host and at the same time regulate the traffic. The leaky bucket is applied after the token bucket, the rate of the leaky bucket needs to be higher than the rate of token dropped in the bucket.

Resource Reservation

A flow of data needs resources such as a buffer, bandwidth, CPU time, and so on. The quality of service is improved if these resources are reserved beforehand. Integrated services type QOS depends heavily on resource reservation to improve the quality of service.

Admission Control

Admission control refers to the mechanism used by a router, or a switch, to accept or reject a flow based on predefined parameters called flow specifications. Before a router accepts a flow for processing, it checks the flow specifications to see if its capacity (in terms of bandwidth, buffer size, CPU speed) and its previous commitments to other flows can handle the new flow.

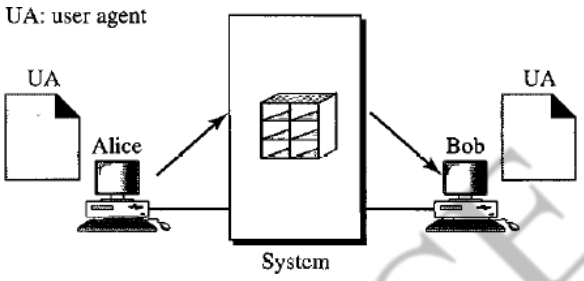
UNIT - 5

ELECTRONIC MAIL

One of the most popular Internet services is electronic mail (e-mail). The architecture of e-mail has four scenarios. The fourth scenario is the most common in the exchange of email.

First scenario

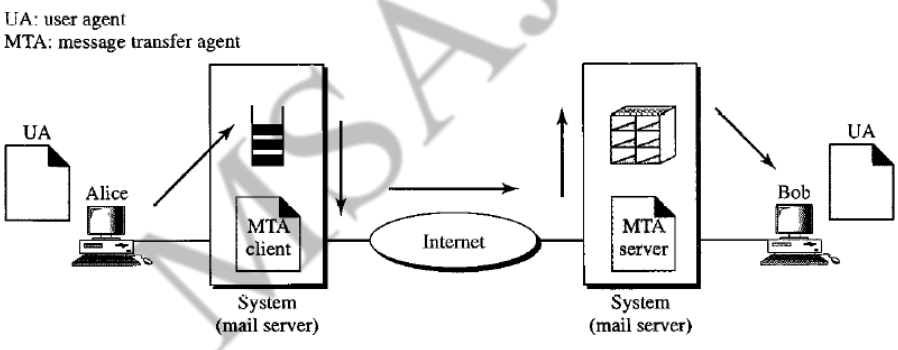
The sender and receiver of the e-mail are users on the same system, they are directly connected to a shared system. The administrator has created one mailbox for each user where the received messages are stored. A mailbox is part of a local hard drive, a special file with permission restrictions. Only the owner of the mailbox has access to it. When Alice, a user, needs to send a message to Bob, another user, Alice runs a user agent (UA) program to prepare the message and store it in Bob's mailbox. The message has the sender and recipient mail box addresses (name of files). Bob can retrieve and read the contents of his mailbox at his convenience, using a user agent.



First scenario in electronic mail

Second Scenario

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



Second scenario

Alice needs to use a user agent program to send her message to the system at her own site. The system (sometimes called the mail server) at her site uses a queue to store messages waiting to be sent. Bob also needs a user agent program to retrieve messages stored in the mailbox of the system at his site. The message, however, needs to be sent through the Internet from Alice's site to Bob's site. Here two message transfer agents are needed: one client and one server. Like most client/server programs on the Internet, the server needs to run all the time because it does not know when a client will ask for a connection. The client, on the other hand, can be alerted by the system when there is a message in the queue to be sent.

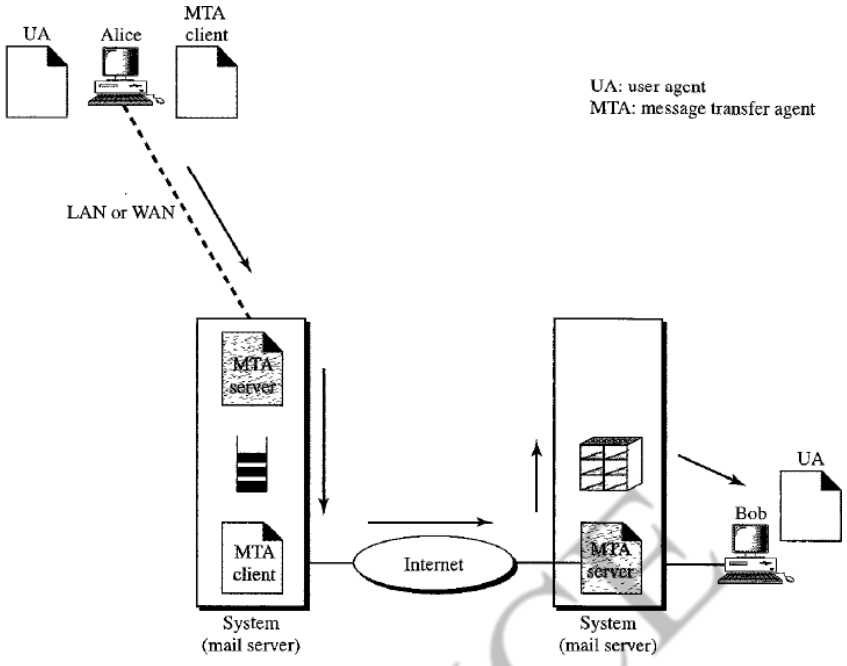
Third Scenario

In the third scenario, Bob, as in the second scenario, is directly connected to his system. Alice, however, is separated from her system. Either Alice is connected to the system via a point-to-point WAN, such as a dial-up modem, a DSL, or a cable mode, or it is connected to a LAN in an organization that uses one mail server for handling e-mails- all users need to send their messages to this mail server.

Alice still needs a user agent to prepare her message. She then needs to send the message through the LAN or WAN. This can be done through a pair of message transfer agents (client and server). Whenever Alice has a message to send, she calls the user agent which, in turn, calls the MTA client. The MTA client establishes a connection with the MTA server on the system, which is running all the time.

The system at Alice's site queues all messages received. It then uses an MTA client to send the messages to the system at Bob's site, the system receives the message and stores it in Bob's mailbox.

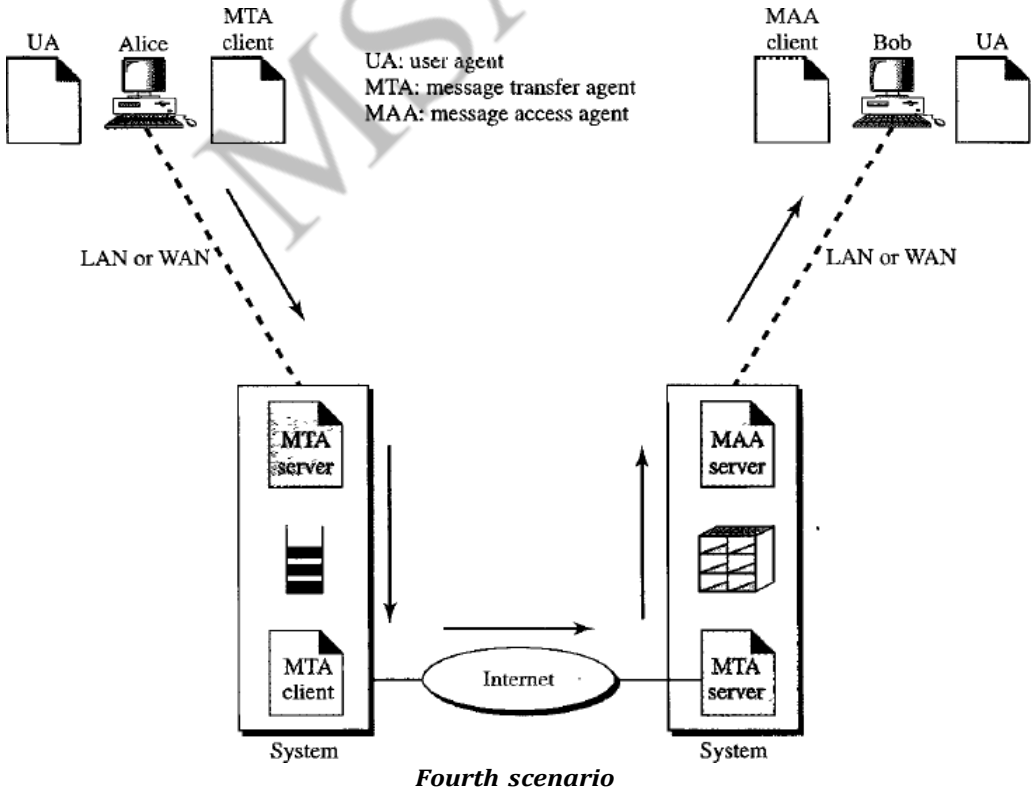
At his convenience, Bob uses his user agent to retrieve the message and reads it. Note that we need two pairs of MTA client/server programs.



Third scenario

Fourth Scenario

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

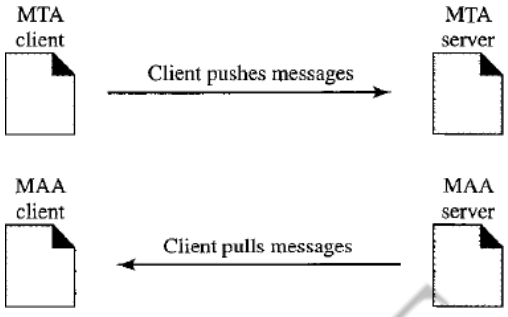


Fourth scenario

There are two important points here. First, Bob cannot bypass the mail server and use the MTA server directly. To use MTA server directly, Bob would need to run the MTA server all the time because he does not know when a message will arrive. This implies that Bob must keep his computer on all the time if he is connected to his system through a LAN. If he is connected through a WAN, he must keep the connection up all the time. Neither of these situations is feasible today.

Second, note that Bob needs another pair of client/server programs, message access programs. This is so because an MTA client/server program is a push program: the client pushes the message to the server. Bob needs a pull program. The client needs to pull the message from the server.

User Agent

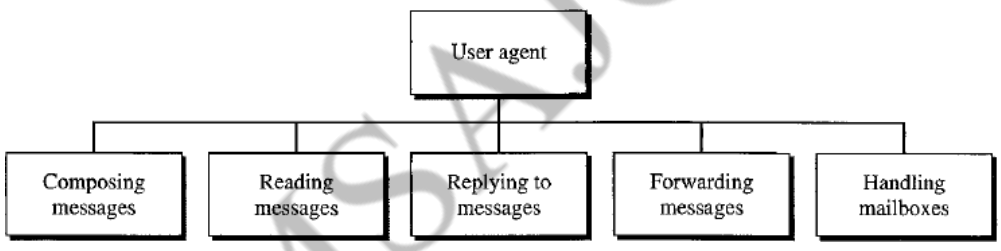


Push versus pull in electronic mail

It provides service to the user to make the process of sending and receiving a message.

Services Provided by a User Agent

A user agent is a software package (program) that composes, reads, replies to, and forwards message. It also handles mailboxes.



Services of user agent

Composing Messages A user agent helps the user compose the e-mail message to be sent out. Most user agents provide a template on the screen to be filled in by the user. Some even have a built-in editor that can do spell checking, grammar checking.

Reading Messages The second duty of the user agent is to read the incoming messages. When a user invokes a user agent, if first checks the mail in the incoming mailbox.

Replying to Messages Replying is defined as sending a message to the sender or recipients of the copy Forwarding is defined as sending the message to a third party. A user agent allows the receiver to forward the message, with or without extra comments, to a third party.

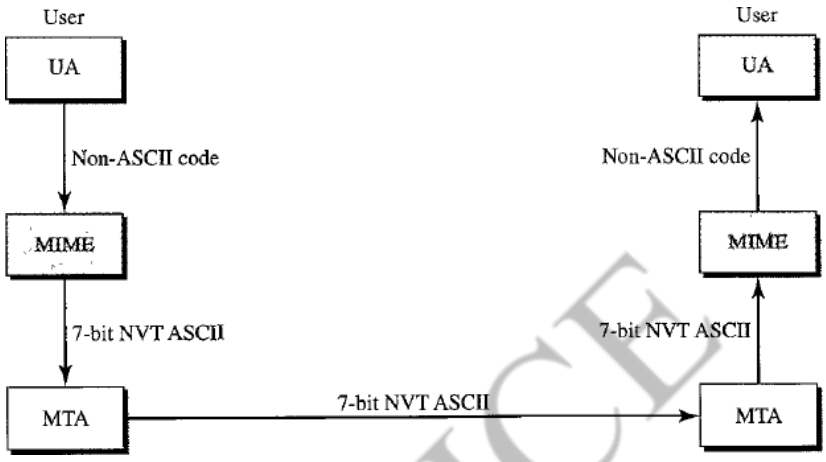
Handling Mailboxes A user agent normally creates two mailboxes: an inbox and an outbox. Each box is a file with a special format that can be handled by the user agent. The inbox keeps all the received e-mails until they are deleted by the user. The outbox keeps all the sent e-mails until the user deletes them. Most user agents today are capable of creating customized mailboxes.

Sending Mail To send mail, the user, through the UA, creates mail that looks very similar to postal mail. It has an envelope and a message.

MIME

Electronic mail has a simple structure. It can send messages only in NVT 7-bit ASCII format. **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 MTA to be sent through the Internet. The message at the receiving side is transformed back to the original data.

MIME as a set of software function that transforms non-ASCII data to ASCII data and vice versa.

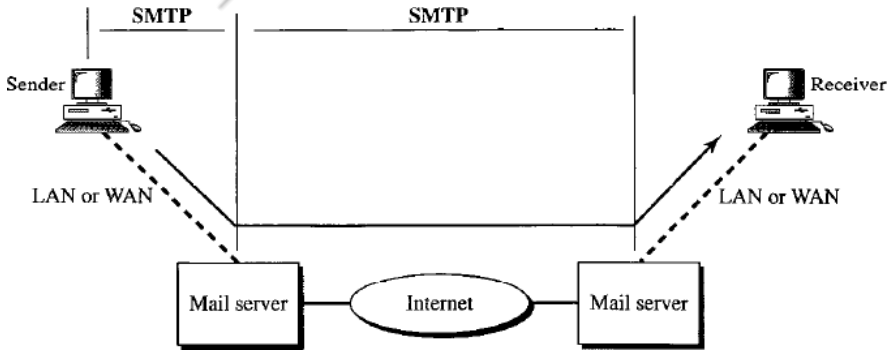


MIME defines five headers that can be added to the original e-mail header section to define the transformation,

1. MIME – version, 2. Content-Type, 3. Content-Transfer-Encoding, 4.Content-Id, 5. Content Desciption.

Message Transfer Agent : SMTP

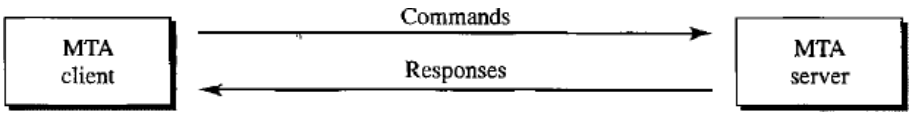
The actual mail transfer is done through message transfer agents. To send mail, a system must have the client MTA, and to receive mail, a system must have server MTA. The formal protocol that defines the MTA client and server in the Internet is called the Simple Mail Transfer Protocol (SMTP). Two pairs of MTA client/server programs are used in the most common situation (as in fourth scenario).



SMTP is used two times, between the sender and the sender's mail server and between the two mail servers. It 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 MTA server.



Commands It is sent from the client to server, It consists of keyword followed by arguments.

Keyword	Arguments
HELO	Senders host name
MAIL FROM	Sender of the message
RCPT TO	Intended recipient of the message
DATA	Body of the mail

Responses Responses are sent from the server to the client. A response is a three-digit code that may be followed by addition textual information.

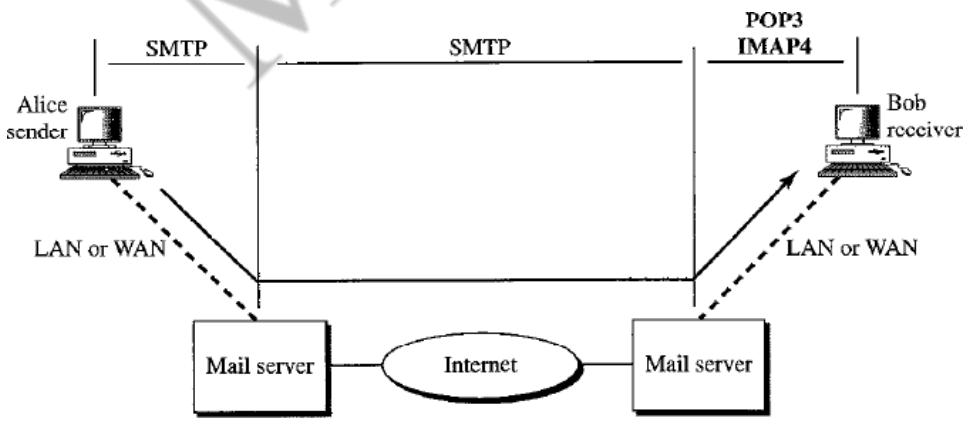
Code	Description
220	Service ready
354	Start mail input
421	Service not available
450	Mailbox not available

Main Transfer Phases The process of transferring a mail message occurs in three phases: connection establishment, mail transfer and connection termination.

MEESAGE ACCESS AGENT: POP AND IMAP

The first and the second stage of mail delivery use SMTP. However, SMTP is not involved in the third stages because SMTP is a push protocol; it pushes the message from the client to the server. On the other words, the direction of the bulk data (messages) is from the client to the server. On the other hand, the third stage needs 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).** The position of these two protocols in the most common situation.

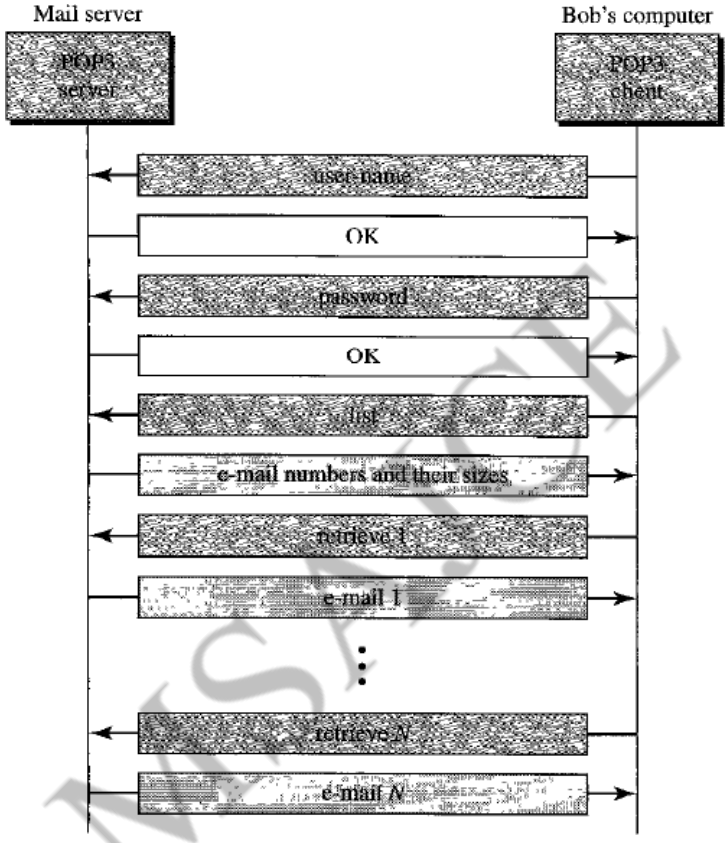


POP3

Post office protocol, version 3(POP3) is simple and limited in functionality. The client POP3 software is installed on the recipient computer; the server POP3 software is installed on the mail server.

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 replaying. 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 alter retrieval and organizing.



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.

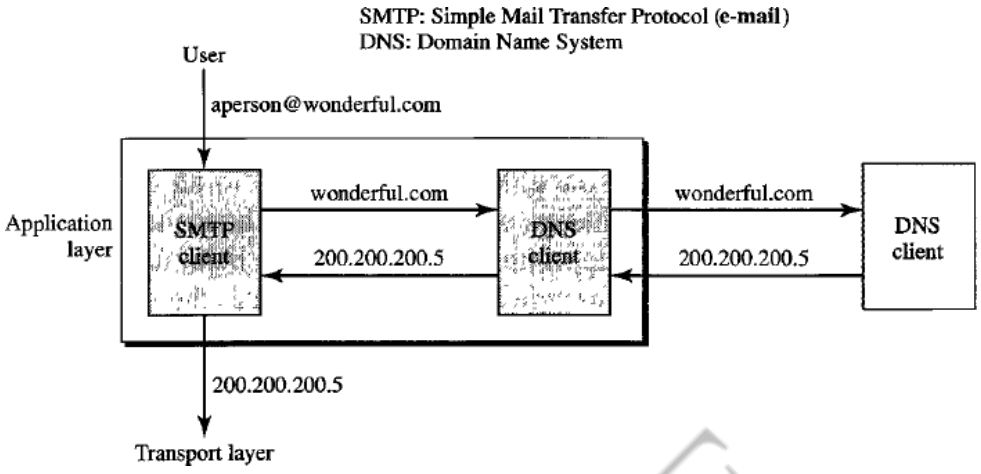
POP3 is deficient in several ways. It doesnot allow the user to organize her mail on the server; the user cannot have different folders on the server.(of course, the user can create folders on her own computer). In addition, POP3 does not allow the user to partially check the contents of the mail before downloading.

IMAP4 provides the following extra following

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

DNS (Domain Name System)

DNS client/server program can support an e-mail program to find the IP address of an e-mail recipient. A user of an e-mail program may know the e-mail address of the recipient. The IP protocol needs the IP address. The DNS client program sends a request to a DNS server to map the e-mail address to the corresponding IP address.



If the email id and user are few, it is possible to maintain the details in one server, but today, email user are huge volume so it is not possible to keep the huge volume of details in one computer, and if any user changes their details, this to be updated, this also becomes complicate and create network traffic.

Solution is to divide this huge amount of information into smaller parts and store each part on a different computer. In this method, the host that needs mapping can contact the closest computer holding the needed information. This method is used by the **Domain Name System (DNS).**

NAME SPACE

The names assigned to machines must be carefully selected from a name space with complete control over the binding between the names and IP addresses. In other words, the names must be unique because the addresses are unique. A name space that maps each address to a unique name can be organized in two ways: flat or hierarchical.

Flat Name Space

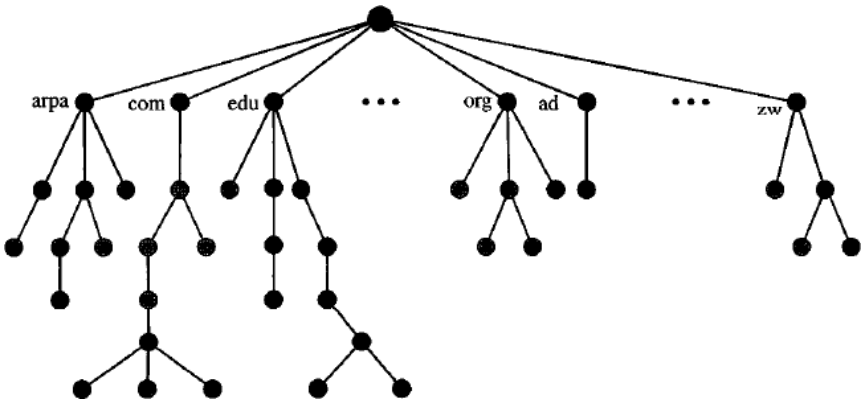
A name in this space is a sequence of characters without structure. The names may or may not have a common section. The disadvantage is it cannot be used in a large system such as Internet because it must be centrally controlled to avoid ambiguity and duplication.

Hierarchical Name Space

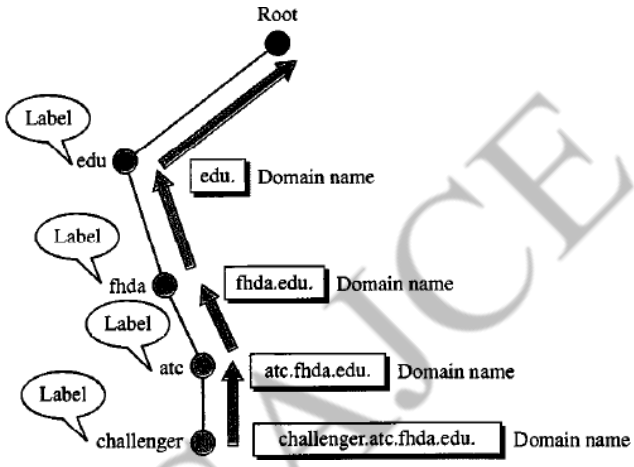
Each name is made of several parts. The first part can define the nature of the organization, the second part can define the name of an organization, the third part can define departments in the organization, and so on. In this case, the authority can assign the part of the name that defines the nature of the organization. Rest of the name can be given to the organization itself. The organization can add suffixes (or prefixes) to the name to define it host.

DOMAIN NAME SPACE

To have a hierarchical name space, a domain name space is designed. In this design the names are defined in an inverted-tree structure with the root at the top. The tree can have only 128 levels: level 0 to level 127.



Domain name space



Labels and Domain Name

Label Each node in the tree has a label, which is a string with a maximum of 63 characters.
Domain Name Each node in the tree has a domain name. A full domain name is a sequence of labels separated by dots(.). The domain names are always read from the down node to up node.

Fully Qualified Domain Name
If a label is terminated by a null string, it is called a fully qualified domain name (FQDN). It contains the full name of a host. Example **challenger.atc.fhda.edu,** it is fully qualified because a computer named challenger installed at the Advanced Technology Center (ATC). A DNS server can only match a FQDN to an address. Note that the name must end with a null label, but because null means nothing, the label ends with a dot (.).

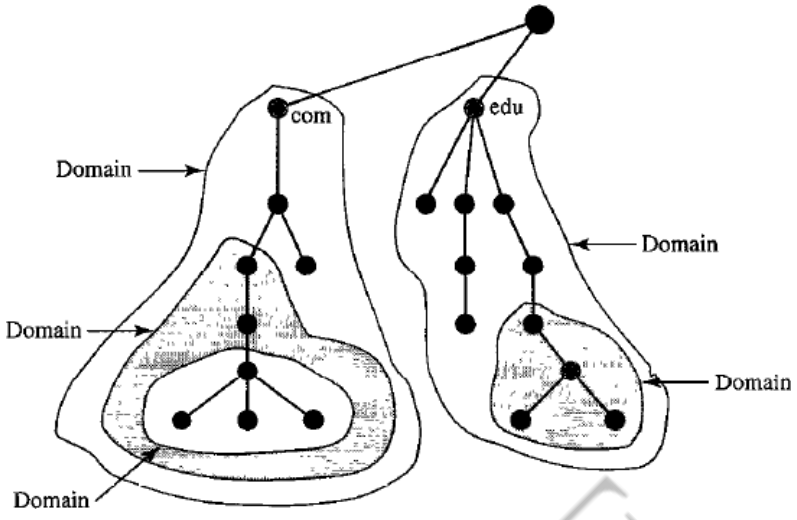
Partially Qualified Domain Name
If a label is not terminated by a null string, it is called a partially qualified domain name (PQDN). It starts from a node, but it does not reach the root. It is used when the name to be resolved belongs to the same site as the client. Here the resolver can supply the missing part, called the suffix, to create an FQDN. For example, if a user at the fhda.edu site wants to get the IP address of the challenger computer, it can define the partial name

challenger

The DNS client adds the suffix atc.fhda.edu. before passing the address to the DNS server. The DNS client normally holds a list of suffixes.

Domain

A domain is a subtree of the domain name space. The name of the domain is the domain name of the node at the top of the subtree. Note that a domain may itself be divided into domains (of **subdomains**)

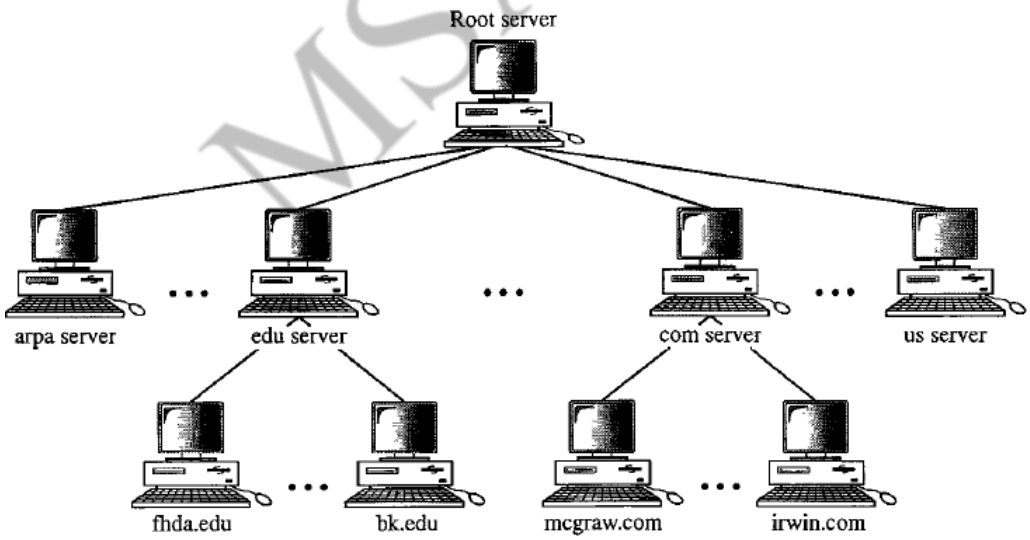


DISTRIBUTION OF NAME SPACE

The information contained in the domain name space must be stored. However, it is very inefficient and also unreliable to have just one computer store such a huge amount of information. It is inefficient because responding to requests from all over the world places a heavy load on the system.

Hierarchy of Name Servers

The solution to these problems is to distribute the information among many computers called DNS servers. Divide the whole space into many domains based on the first level. DNS allows domains to be divided further into smaller domains (subdomains). Each server can be responsible for either a large or a small domain.



Hierarchy of Name Servers

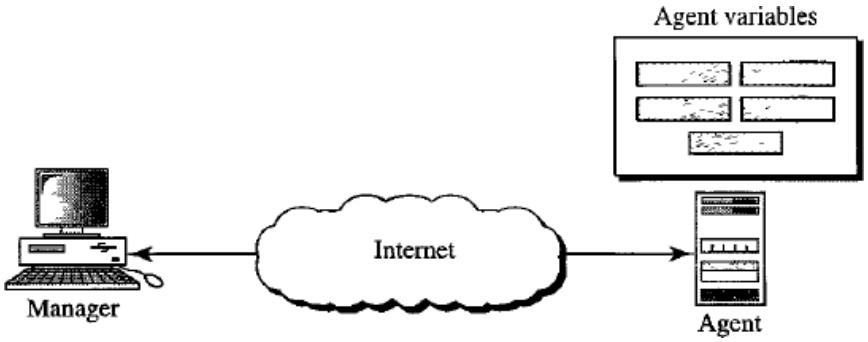
Zone : Since the complete domain name hierarchy cannot be stored on a single server, it is divided among many servers. What a server is responsible for or has authority over is called a zone. Server makes a database called a zone file and keeps all the information for every node under the domain.**Root Server** is a server whose zone consists of the whole tree. A root server usually does not store any information about domains but delegates its authority to other servers, keeping references to those servers.

SNMP (SIMPLE NETWORK MANAGEMETN PROTOCOL)

SNMP is a framework for managing devices in an internet using the TCP/IP protocol suite. It provides a set of fundamental operations for monitoring and maintaining and internet.

Concept

SNMP uses the manager and agent. Manager usually host, controls and monitors a set of agents called routers.



Managers and Agents

Manger is a host that runs the SNMP client program. Agent is a router that runs the SNMP server program. Management is achieved through simple interaction between a manager and an agent.

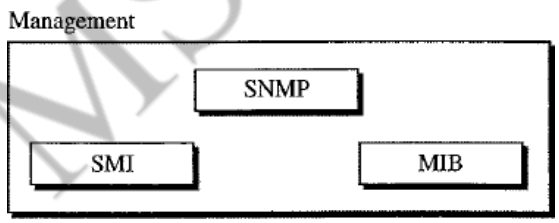
The agent keeps performance information in database. The manager access the values in database. Agent stores the number of packets received and forwarded. The manager can fetch and compare the values of these two variables to see if the router is congested or not.

Manager perform action on agent, reboot the agent remotely at any time to make value of the counter is 0. Agents also contribute in management process. The server program running on the agent can check any unusual, it can send a warning message called **trap** to the manager.

Management Components

To do management tasks, SNMP uses two other protocols : **Structure of Management Information (SMI)** and **Management Information Base (MIB).**

Role of SNMP



Components of network management on the Internet

SNMP define the format of the packet to be sent from a manager to an agent and vice versa. It also interprets the result and creates statistics. SNMP is responsible for reading and changing these values.

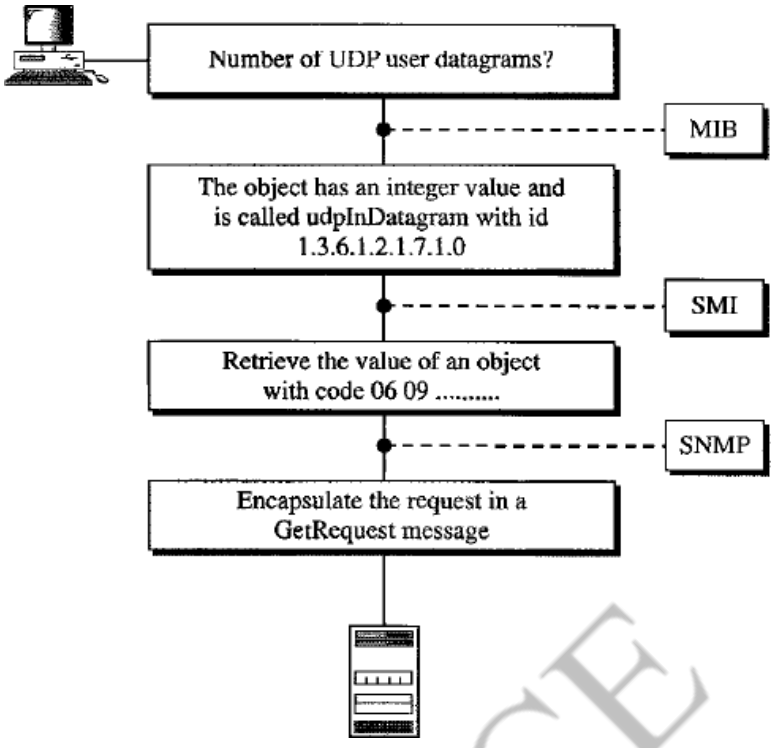
Role of SMI

SMI defines the general rules for naming objects, defining object types (including range and length), and showing how to encode objects and values.

SMI does not define the number of objects and entity should manage or name the objects to be managed or define the association between the objects and their values.

Role of MIB

MIB creates a collection of named objects, their types, and their relationships to each other in an entity to be managed.

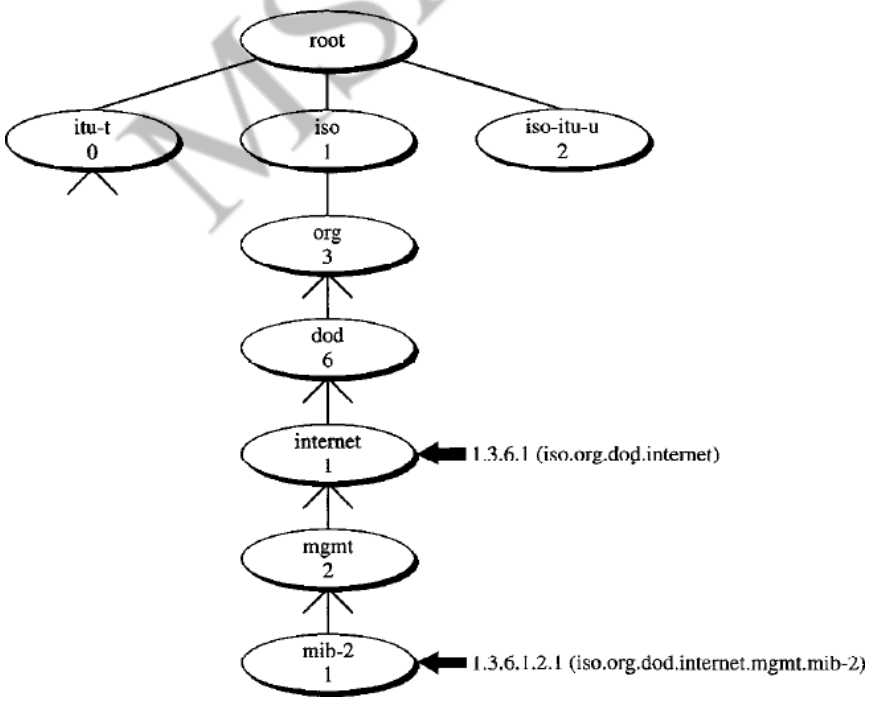


Management overview

Structure of Management Information

The structure of Management Information is a component for network management. It functions are

1. To name objects
2. To define the type of data that can be stored in an object
3. To show how to encode data for transmission over the network



SMI is a guideline for SNMP. It emphasizes three attributes to handle an object : **name, data type and encoding method.**

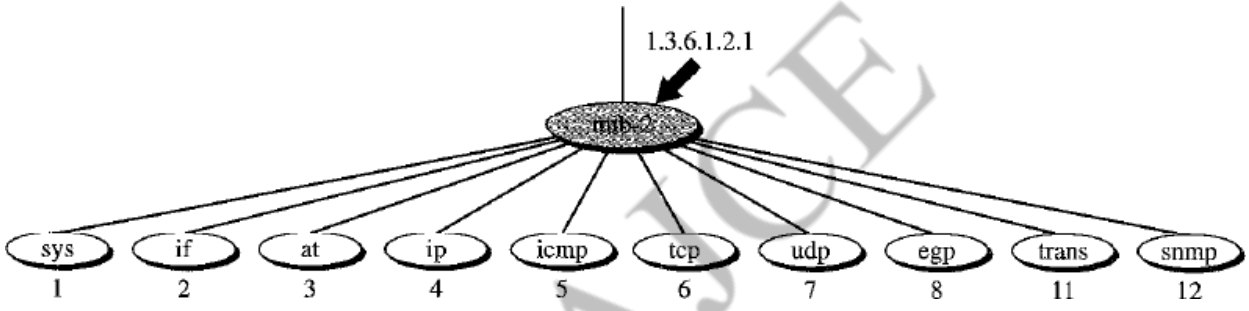
Name : SMI requires that each managed object (such as a router, a variable in a router, a value) have a unique name. To name objects globally, SMI uses an **object identifier**, which is a hierarchical identifier based on a tree structure.

The tree structure starts with an unnamed root. Each object can be defined by using a sequence of integers separated by dots. The tree structure can also define an object by using a sequence of textual names separated by dots. The integer-dot representation is used in SNMP. The name-dot notation is used by people. Ex : iso.org.dod.internet.mgmt.mib-2 → 1.3.6.1.2.1

Type : The second attribute of an object is the type of data stored in it. To define the data type SMI uses fundamental **Abstract syntax notation 1 (ASN.1)** definitions and adds some new definitions. SMI has two data type : ***simple and structured.***

Simple Type - Integer, String, IPAddress and Bits
Structure Type - sequence (struct) and sequence of (array).

Management Information Base (MIB)
It is the second component used in network management. Each agent has its own MIB, which is a collection of all the objects that the manager can manage. The object in MIB are categorized under 10 different groups: system, interface, address translation, ip, icmp, tcp, udp, egp, transmission, and snmp.



Sys (system) defines general information about the node.
if *(interface)* defines information about all the interfaces of the node.
at (address translation) define the information about the ARP table.
ip defines information related to IP, such as the routing table and the IP address.
icmp (ICMP) defines information related to ICMP, such as the number of packets sent and received and total errors created.
tcp defines general information related to TCP, such as connection table, number of ports.
udp defines general information related to UDP, such as number of ports and number of packets sent and received.
snmp defines general information related to SNMP itself.

SNMP
SNMP uses both SMI and MIB in Internet network management. It is an application program that allows
1. A manager to retrieve the value of an object defined in an agent.
2. A manager to store a value in an object defined in an agent.
3. An agent to send an alarm message about an abnormal situation to the manager

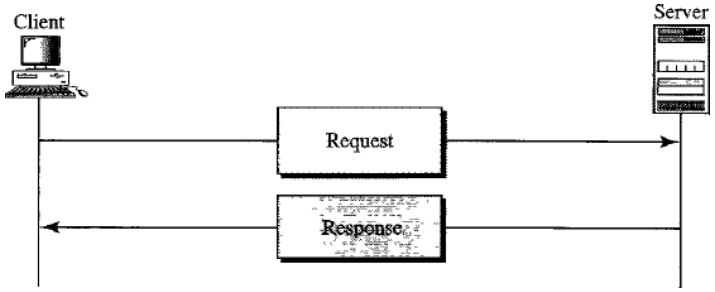
PDU
SNMP defines eight types of packets: GetRequest, GetNextRequest, GetBulkRequest, SetRequest, Response, Trap, InformaRequest, and Report.

Messages
SNMP does not send only a PDU, it embeds the PDU in a message. A message has four elements : version, header, security and data.

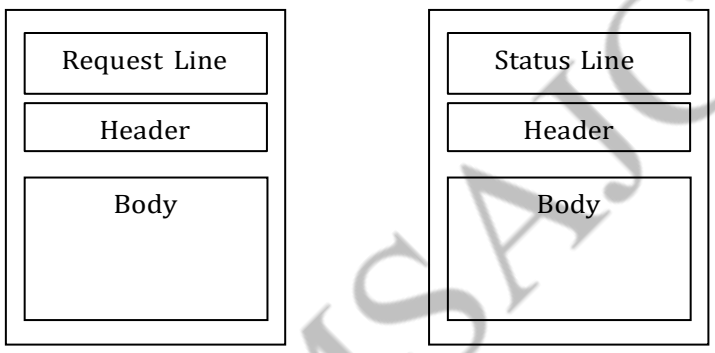
UDP Ports SNMP uses the services of UDP on two well-known ports, 161 and 162. The port 161 is used by the server and 162 is used by client.

HTTP (Hypertext Transfer Protocol)
It is a protocol used mainly to access data on the World Wide Web. HTTP functions as a combination of FTP and SMTP. It transfers files like FTP, data transferred between the client and server.

HTTP Transaction
HTTP transaction is done between the client and server. HTTP is stateless protocol. The client initializes the transaction by sending a request message. The server replies by sending a response.



Messages
The formats of the request and response message are similar. A request message consists of a ***request line, a header, and sometimes a body***. A response message consists of a ***status line, a header, and sometime a body***.



Request Message ***Response Message***

Request and Status Line The first line in a request message is called a request line, the first line in the response message is called the status line.

Request type: This field is used in the request message. Several request types are defined. The request type is categorized into methods.

Method	Action
GET	Requests a document from the server
HEAD	Requests information about a document but not the document itself
POST	Sends some information from the client to the server
PUT	Sends a document from the server to the client
TRACE	Echoes the incoming request
CONNECT	Reserved
OPTION	Inquires about available options

Status Code This field is used in the response message. It consists of three digits.

Code	Phrase	Description
200	OK	The request is successful
201	Created	A new URL is created
202	Accepted	The request is accepted, but it is not immediately acted upon.

Header: The header exchanges additional information between the client and the server. The client can request document to be sent in a special format. The header can consist of one or more header lines. Each header line has header name, a colon, a space, and a header value.
Header line has four categories : general header, request header, response header and entity header.

General header: The general header gives general information about the message and can be present in both a request and response. Ex: Connection, MIME-version, Date

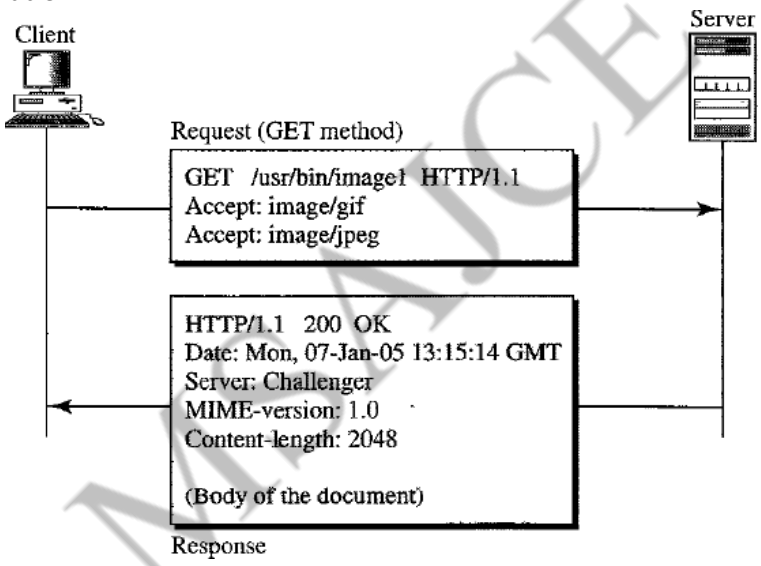
Request header: The request header can be present only in a request message. It specifies the clients preferred document format. Ex : Accept, From, Host.

Response header: The response header can be present only in a response message. It specifies the servers configuration and special information about the request. Ex: Age, Server.

Entity header: The entity header gives information about the body of the document. Although it is mostly present in response messages, some request messages, such as POST or PUT methods, that contain a body also use this type of header. Ex: Allow, Expires, Location.

Body:
The body can be present in a request or response message. Usually, it contains the document to be sent or received.

Example HTTP operation



Persistent Versus NonPersistent connection

Nonpersistent Connection
One TCP connection is made for each request/response.
The following lists the steps in this strategy
1. The client opens a TCP connection and sends a request.
2. The server sends the response and closes the connection.
3. The client reads the data until it encounters an end-of-file marker, it then closes the connection.

Persistent Connection
In this connection server leaves the connection open for more requests after sending a response. The server can close the connection at the request of a client or if a time-out has been reached. The sender usually sends the length of the data with each response. However, there are some occasions when the sender does not know the length of the data This is the case when a document is create dynamically or actively. In these cases, the server informs the client that the length is not known and closes the connection after sending the data so the client knows that the end of the dat has been reached.

2 - MARKS

UNIT -1

1. Define Protocol.

Protocol is a set of rules that perform data communication. It defines what is communicated, how it is communicated and when it is communicated. Sender and receiver perform data communication by following protocol.

2. Define point-to-point connection.

It provides a dedicated direct link between two devices. The entire capacity of the link is reserved for transmission between these two devices only.

3. Explain single bit and burst error.

Single bit error – only one bit gets changes due to external interference.
Burst error – many bits gets changes due to external interference.

4. Define frame

The data link layer divides the stream of bits received from the network layer into manageable data units called frame.

5. Define encryption and decryption?

Encryption is the process of transmitting the original message to another form, that is unknown form.
Decryption is the process of converting unknown form to known form that is to original message.

6. Define IP

The internet layer in the TCP/IP model defines an official packet format and protocol called internetworking protocol (IP). IP is a host-to-host protocol that it can deliver a packet from one physical device to another device.

7. What is TCP

TCP is reliable connection-oriented protocol that allows segments (stream of data into smaller units) on one machine to be delivered without error on any other machine in the Internet.

8. What is ARP

Address Resolution protocol is used to find the physical address of a device, if its IP (logical) address is known.

9. What is RARP.

Reverse Address Protocol allows a host to discover its IP address when it knows physical address.

10. What is jitter.

Jitter refers to the variation in the packet arrival time, that is, an uneven delay in the delivery of audio or video packets.

11. Define flow control.

Flow control refers to a set of procedures used to restrict the amount of data that the sender can send before receiving an acknowledgement from the receiver.

12. What is piggybacking?

It means combining data to be sent and acknowledgement of the frame received in one single frame.

13. What is meant by byte stuffing?

It is the process of adding 1 extra byte whenever there is a flag or escape character in the text.

UNIT - 2

1. Define LAN.
It creates network that allows a number of independent devices to communicate directly with each other.

2. Write the functions of LLC layer.
It performs the functions of flow control, error control, logical addresses, control information, and data from the upper layers are packaged into a packed called protocol data unit (PDU).

3. Write the functions of MAC
Media access control (MAC) is the lower sub layer of the data link layer. It defines the specific access method and the framing format for each LAN and resolves the contention for the shared media, but LLC is common for all.

4. Define Manchester encoding.
It is bit encoding scheme that transmits the exclusive – OR of the clock and the Non-Return to Zero (NRZ) – encoded data, which is used on the Ethernet to avoid the clock recovery problem.

5. Define unicast, multicast and broadcast.
Unicast is communication between one-to-one.
Multicast is communication between one-to-many.
Broadcast is communication between many-to-many.

6. What is exponential backoff?
A retransmission strategy that doubles the delay interval between each retransmission attempt is a general technique known as exponential backoff.

7. What is frequency hopping spread spectrum?
In frequency hopping spread spectrum that involves transmitting of signal over a random sequence of frequencies. The first signal transmitting at one frequently, then a second signal transmitting at another frequency, then a third, and so on.

8. Define hidden node problem.
Hidden node problem occurs on a wireless network, when two nodes are sending to a common destination but are unaware that the other exists.

9. Define exposed node problem.
It occurs on a wireless network when two nodes receive signals from a common source but each one is able to reach other nodes that do not receive this signal.

10. Define piconet.
Upto eight devices can communicate in a small network called a piconet. Each piconet has exactly one can act as master(M) and all other devices connected to the master must act as slaves.

11. Define scatternet.
One Bluetooth devices can operate simultaneously on two piconets, acting as a bridge between the two. Combination of two or more piconet is called scatternet.

12. Define switches.
Switches are hardware and software devices capable of creating temporary connections between two or more devices linked to the switch but not each other.

13. Define VCI
VCI is uniquely identifies the connection link at the switch and is carried inside the header of the packets that belong to this connection. It is a small number in a data frame changes from one switch to another switch used for data transfer.

UNIT - 3

1. What is routing?
Routing is a network-wide process that determines the end-to-end paths that packets take from source to destination by exchanging nodes topology information to build correct forwarding tables.

2. What is MOSPF?
Multicast Open Shortest Path First (MOSPF) is a multicast routing protocol and is an extension of the OSPF protocol that uses multicast link state routing to create source based least-cost tree.

3. Define RPB and RPM
Reverse Path Broadcasting (RPB) creates a shortest path broadcast tree from the source to each destination. It guarantees that each destination receives only one copy of the packet.
RPB does not multicast the packet, it performs only broadcasting. This is not efficient. To increase efficiency, the multicast packet must reach only those networks that have active members of that particular group. This is called reverse path multicasting (RPM).

4. Define convergence
The process of getting consistent routing information to all the nodes is called convergence.

5. Define RIP
Routing information protocol (RIP) is a simple protocol intradomain routing protocol used inside an Autonomous System (AS) based distance vector routing algorithm, in which each router shares, at regular intervals, its knowledge about the entire AS with its neighbours.

6. Define LSR
Link state routing (Low-cost path algorithm) performs that the information on directly connected neighbors and current link costs are flooded to all routers, each router uses this information to build a view of the network which is the base to make forwarding decisions.

7. What is meant by flooding?
The router sends all its like-state information about its neighbors, to its neighbors, then the neighbors forward this information to its neighbors and so on. Thereby, every router receives the copy of the same information. This process continues until the information has reached all the nodes in the network.

8. Define OSPF
Open shortest part first (OSPF) is protocol widely used for intra-AS routing in the Internet. It is the popular intradomain routing protocol based on link state routing that uses flooding of link-state information and a Dijkstra least-cost path algorithm.

9. Define metric of OSPF
The OSPF allows the administrator to assign a cost, called the metric, to each router link, for traffic on the specified TOS. The metric can be based on a type of service. The lower metric for the TOS value is low delay and a high metric for everything else.

10. Define an area.
The link-state routing protocols such as OSPF can be used to partition a routing domain (AS) into sub domains called areas, to improve scalability, which is a set of adjacent routers that administratively configured to exchange full routing information with each other.

11. What is an address space?
An address space is “the total number of addresses used by the protocol”. For an N bits address, 2N bits address space can be used, because each bit can have two different values (0 or 1).

UNIT - 4

1. Define process-to-process delivery.
Delivery of a packet from the source host process to the destination host process. This is called as process-to-process delivery.

2. Define socket addresses.
A port identifies a single application on a single computer. The term socket address or simple socket is a combination of an IP address and a port address.

3. What is UDP?
User datagram protocol (UDP) is a connectionless and unreliable TCP/IP transport layer protocol. UDP allows computers to send data without needing to establish a virtual connection.

4. What is shrinking in the sliding window?
Shrinking the window means that moving the right wall to the left, it revoking the eligibility of some bytes for sending.

5. Define congestion
When too many packets are present in the subnet, the performance of the network will be degraded. This is called congestion.

6. Define congestion control.
Congestion control refers to the techniques and mechanisms that can either prevent congestion, before it happens or remove congestion, after it has happened. It maintains the load below the capacity.

7. Define congestion avoidance.
The linear increase phase of TCP's congestion control protocol instead of an exponential one is known as congestion avoidance.

8. What is DECbit?
Destination experience congestion (DEC) bit is congestion-avoidance technique implemented in which routers to notify the endpoints of imminent congestion to avoid congestion by adds a bit in the header of packets sent. Its utility is to predict possible congestion and prevent it. The endpoints decrease their sending rates when a certain percentage of received packets have the bit set.

9. What is meant by RED?
Random early detection (RED) is a queuing discipline for router, when a router is almost congested and drop packets randomly to alert the senders to slow down.

10. What is QoS?
Quality of Service is the overall performance of a telephony or computer network, particularly the performance seen by the users of the network, which provide the packet delivery guarantees, which is related to bandwidth and delay.

UNIT - 5

1. What is the function of User Agent (UA)
User agents for electronic mail is a software package (program) that prepares the messages, creates the envelope, and puts the message in the envelope. The UA is normally a program used to send and receive mail. It also handles mailboxes.

2. Define MIME
Multipurpose Internet Mail Extensions (MIME) as a set of software functions that transform non-ASCII data to ASCII data and vice versa.

3. Define WWW
The World Wide Web (WWW) or the web is an architectural framework for accessing linked documents spread out over millions of machines all over the Internet. The WWW is a repository of information linked together from points all over the world.

4. What is HTTP?
The Hyper Text Transfer Protocol (HTTP) is an application-layer protocol based on a request/reply paradigm and used to access data on the WWW.

5. What are persistent connections?
The server leaves the TCP connection open after sending a response. The client and server can exchange subsequent multiple request/response messages can be sent over the same TCP connection.

6. Define DNS.
The Domain Name System (DNS) is a client/server application that identifies each host on the Internet with a unique user-friendly name.
It is a directory lookup service that provides a mapping between the name of a host on the Internet and its numerical address.

7. Define FQDN
Fully Qualified Domain Name (FQDN) is a domain name consisting of labels beginning with the full name of the host and going back through each level to the root node.

8. Define PQDN
A Partially Qualified Domain Name (PQDN) is a domain name that does not include all the levels between the host and the root node.

9. Define Zone
Zone is defined as a contiguous part which is a server responsible for or has authority over the entire tree (a collection of one or more sub domains within a domain).

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

11. Define SNMP?
Simple Network Management Protocol (SNMP) is developed for use as a network management tool for managing devices in networks and internetworks operating TCP/IP. This internet protocol provides a set of fundamental operations for monitoring of hosts, routers and maintaining an Internet.

12. Define SIM
Structure of Management Information (SIM) is a component used in network management. Its functions are to name objects to define the types of data that can be stored in an object, and to show how to encode data for transmission over the network.

13. Define MIB
Each agent has its own Management Information Base (MIB), which is a collection of all objects. A management station performs the monitoring function by retrieving the value of MIB objects.

B.E./B.Tech. DEGREE EXAMINATION, NOV/DEC 2016

Sixth Semester

Electronics and Communication Engineering

CS 6551 – COMPUTER NETWORKS

(Common to Fourth Semester – Computer Science and Engineering / Fifth Semester – Information Technology)

(Regulations 2013)

Time : Three Hours **Maximum : 100 Marks**

Answer ALL questions

PART – A (10 X 2 = 20 Marks)

1. List the services provided by data link layer.
2. Write the mechanism of stop and wait flow control.
3. What is meant by exponential backoff?
4. What is scatternet?
5. Define VCI.
6. What is fragmentation and reassembly?
7. Give the comparison of unicast, multicast and broadcast routing.
8. Differentiate between TCP and UDP.
9. Expand POP3 and IMAP4.
10. What is persistent HTTP?

PART – B (5 X 16 = 80 Marks)

11. (a) Draw the OSI network architecture and explain the functionalities of each layer in detail. (16)

OR

(b) (i) Discuss in detail about the network performance measures. (8)
(ii) Explain selective-repeat ARQ flow control method. (8)

12. (a) Explain the physical properties of Ethernet 802.3 with necessary diagram of Ethernet transceiver and adapter. (16)

OR

(b) With a neat sketch explain about IP service model, packet format, Fragmentation and reassembly. (16)

13. (a) Discuss in detail about open source shortest path routing with neat diagrams. (16)

OR

(b) Discuss in detail about any two Multicast routing with neat sketches. (16)

14. (a) Explain various fields of the TCP header and the working of the TCP protocol. (16)

OR

(b) How is congestion controlled? Explain in detail about congestion control techniques in transport layer. (16)

15. (a) Give a detailed note on DNS operation. (16)

OR

(b) (i) Explain in detail about SNMP messages. (8)
(ii) Illustrate the role of POP3 in Electronic mail Applications. (8)

B.E./B.Tech. DEGREE EXAMINATION, MAY/JUNE 2016

Sixth Semester

Electronics and Communication Engineering

CS 6551 – COMPUTER NETWORKS

(Common to Fourth Semester – Computer Science and Engineering / Fifth Semester – Information Technology)

(Regulations 2013)

Time : Three Hours **Maximum : 100 Marks**

Answer ALL questions

PART – A (10 X 2 = 20 Marks)

1. Define flow control.
2. Write the parameters used to measure network performance.
3. Define hidden node problem.
4. What is Bluetooth?
5. Expand ICMP and write the function.
6. Write the types of connecting devices in internetworking.
7. What do you mean by slow start in TCP congestion?
8. List the different phases used in TCP connection.
9. Define URL.
10. Mention the different levels in domain name space.

PART – B (5 X 16 = 80 Marks)

11. (a) Explain any two error detection mechanism in detail. (16)

OR

(b) Explain in detail about:
(i) HDLC (8)
(ii) PPP (8)

12. (a) Give the comparison between different wireless technologies? Enumerate 802.11 protocol stack in detail. (16)

OR

(b) Write a short on :
(i) DHCP (8)
(ii) ICMP (8)

13. (a) With a neat diagram explain Distance vector routing protocol. (16)

OR

(b) Explain about IPv6? Compare IPv4 and IPv6. (16)

14. (a) Define UDP. Discuss the operations of UDP. Explain UDP checksum with one example. (16)

OR

(b) Explain in detail the various TCP congestion control mechanisms. (16)

15. (a) (i) Describe how SMTP protocol is used in E-mail applications. (8)
(ii) Explain HTTP with an example. (8)

OR

(b) Explain in detail about Web service architecture. (16)