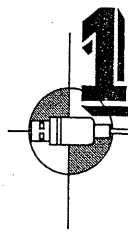
CONTENTS



We n
the depth
discusses
networkin

1.1 Wha

(a) Comp
Comp
infor

(b) In th
Now
comp

1.2 Diff
Syst

(1) **Defin**
comp
they

A Dis
to the

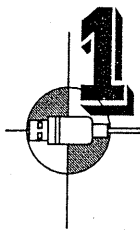
(2) In CN
invok

(3) E.g. c

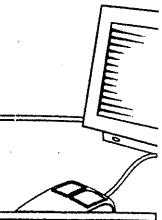
1.3 The

(1) The n

(2) It dea
Open
comm
with



INTRODUCTION TO COMPUTER NETWORK



Marks	
Dec. 03	50 M
May 04	60 M
Dec. 04	6 M
May 05	24 M
Dec. 05	33 M
May 06	20 M
Dec. 06	9 M
May 07	30 M

We now begin with our beautiful and exciting voyage into the depths of Computer Networks. This introductory chapter discusses the models, network standardization and various networking issues.

1.1 What is a Computer Network ?

(a) Computer Network means a collection of interconnected computers or devices. Computers and devices are said to be interconnected if they can exchange information.

(b) In the olden days a single computer served all the needs of an organization. Now a large number of separate computers do the job much faster. These computers are interconnected thus forming *computer networks*.

1.2 Difference Between a Computer Network (CN) and a Distributed System (DS)

(1) **Definition :** Computer Network (CN) means a collection of interconnected computers or devices. Computers and devices are said to be interconnected if they can exchange information.

A Distributed System (DS) is a collection of independent computers that appear to the user as a single coherent system.

(2) In CN, movement of files is invoked by the system. In DS, movement of files are invoked by the user.

(3) E.g. of CN : LAN E.g. of DS : WWW

1.3 The ISO-OSI Reference Model

(1) The model is proposed by the ISO (International Standards Organization).

(2) It deals with connecting *open systems*.

Open Systems : Systems from different manufacturers which are open for communications with other systems and can share data as well as applications with each other.

(3) *OSI (Open Systems Interconnection)* : It is a set of technical standards, approved by the ISO, that provides a basis for data communications.

The 7 Layers

(1) **Physical Layer :**

- It is concerned with the actual physical attachment to the network i.e. it deals with the means of connecting two nodes in a network.
- It deals with transmitting raw bits over the communication channel.
- The design issues here deal with mechanical, electrical timing interfaces and the physical transmission medium which lies below the physical layer.

(2) **Data Link Layer :** It breaks the data into frames and passes it to the network layer. It also does;

- *Error Control* : To control transmission errors.
- *Flow Control* : To prevent the drowning of a slow receiver by a fast transmitter.
- *Access Control* : Control access to the shared channel. A special section of the DLL called the *Medium Access Control sublayer* deals with this.

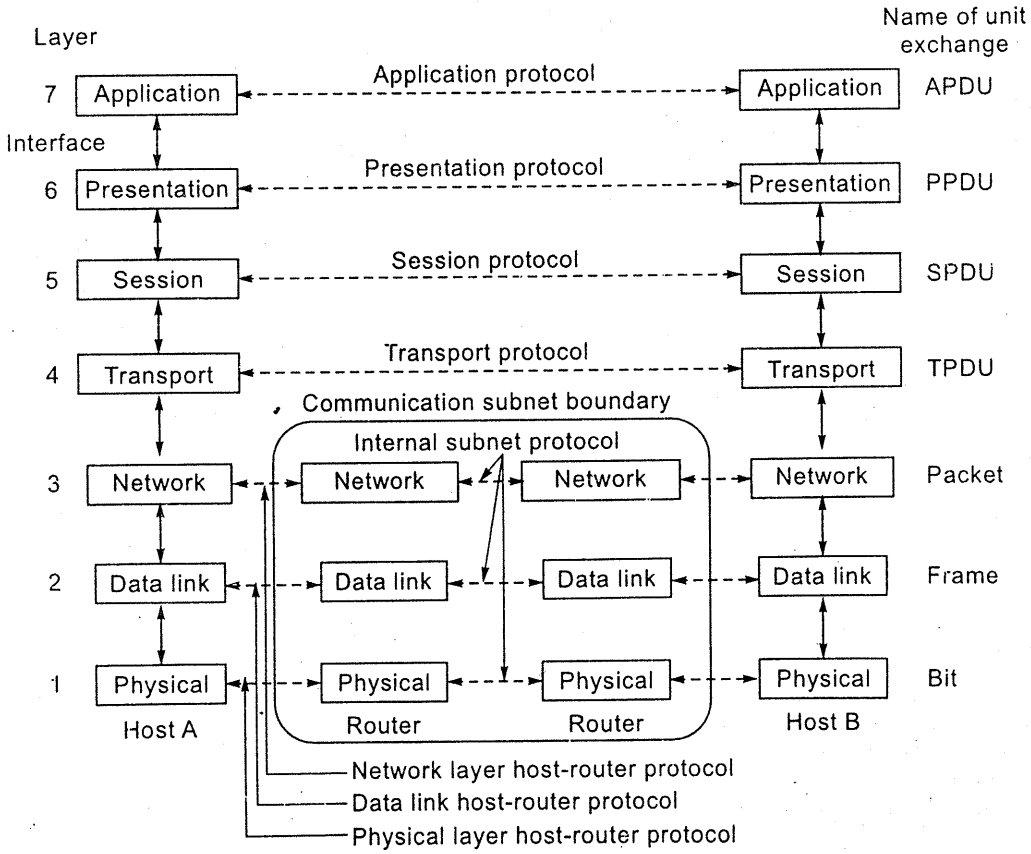


Fig. 1.1 : 7 layers of ISO-OSI.

(3) **Network Layer :** It has the responsibility of performing source to destination delivery of packets. It focuses on :

- Dynamic routing.
- Congestion control.
- Quality of service.
- Addressing.
- Integration of heterogeneous networks.

(4) **Transport Layer :** It deals with :

- Control of data flow in the network.
- Ensuring no loss of data.
- Ensuring that destination is not inundated with data.
- Ensuring that all pieces arrive correctly at the other end.
- Transport layer is a true end to end layer.

(5) **Session Layer :** Its features are :

- *Dialogue Control* : Keeping track of whose turn it is to transmit.
- *Token Management* : Preventing two parties from attempting the same critical operation at the same time.
- *Synchronization* : Checkpointing long transactions to allow them to continue from where they were after a crash.

(6) **Presentation Layer :** It is concerned with the following :

- Syntax of information.
- Semantics of information.
- Compression.
- Encoding of information.

(7) **Application Layer :** Application layer provides user interfaces and support for services like :

- E-mail.
- Remote file access.
- File transfer.
- Shared database management.

1.4 Layered Architecture

May 05 [Q. 3(a)], Dec. 05 [Q. 3(a)], May 07 [Q. 5(a)] *Explain the need for the layered architecture of computer network. Also explain data transmission and protocols in layered architectures.* **(10 M)**

(1) Networks can be very complex. To reduce the design complexity most networks are organized as a stack of *layers*.

(2) The basic issues are :

- Number of layers.
- Name of each layer.
- Contents of each layer.
- Functions of each layer.

(3) Purpose of layered architecture :

- Each layer should offer services to the layer above it.
- Higher layers should be shielded from details of how the service is provided to it by the lower layers.
- Modularization eases maintenance and updating of the system.
- Without layering, each new application has to be reimplemented for every network technology.

(4) **Protocol** : It is an agreement between communicating parties on how communication is to proceed.

The functions of a protocol are :

- Encapsulation, segmentation and reassembly of messages.
- Connection control.
- Ordered delivery.
- Error control, flow control and multiplexing.
- Addressing.

(5) **Peers** : Entities that make the corresponding layers on different machines. Peers communicate using protocols.

(6) **Interface** : It defines which services the lower layer makes available to the layer immediately above it.

(7) **Principles of Layered Architecture :**

- A layer should be created only when a new level of abstraction is required.
- Each layer should perform a well defined function.

- The
inte
- The
flow
- Dist
- The

Host A

Applicati
Presenta
Sessio
Transpo
Networ
Data lin
Physica

Wi

D_n is repres

(8) No data
layer o
immedi
through

In case
from H
Presenta
Header,

A *header*
bits for
physical

On the
at each

- The function of each layer should be chosen with a goal to define internationally standardized protocols.
- The layer boundaries should be chosen so as to minimize the information flow between layers.
- Distinct functions should not be present in the same layer.
- The architecture should not become unwieldy.

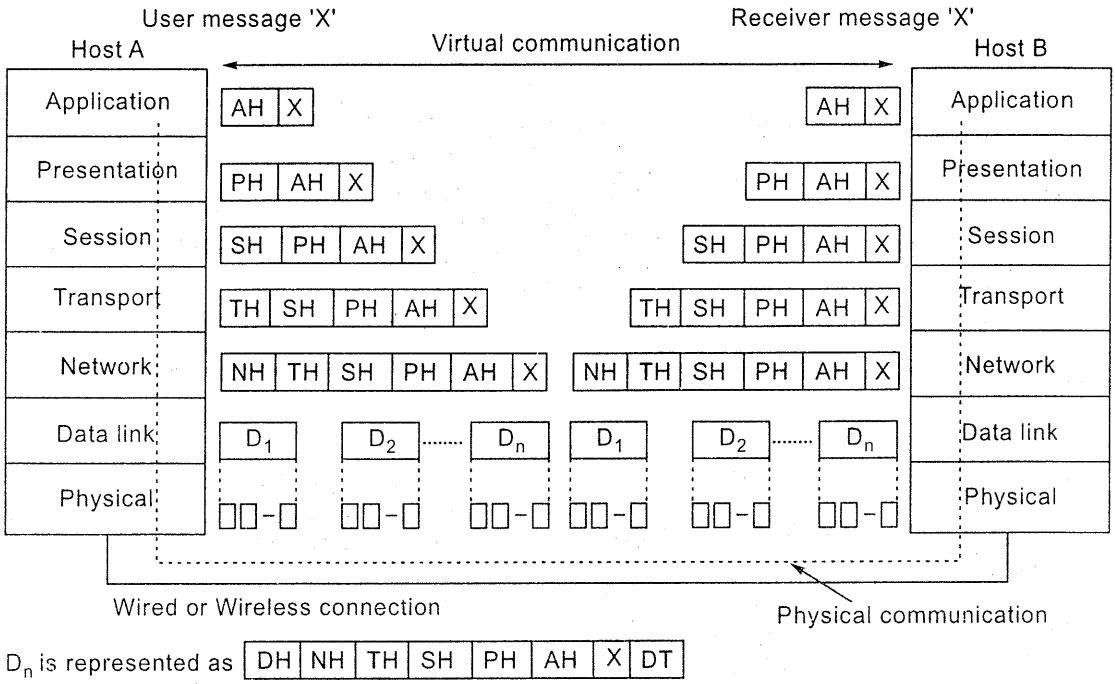


Fig. 1.2 : Data transfer across the OSI layers.

(8) No data is directly transferred from a layer on one machine to the corresponding layer on another machine. Instead each layer passes information to the layer immediately below it until lowest layer is reached. Now information is passed through the physical medium.

In case of the OSI model (refer above figure) : Where a message 'x' is transmitted from Host A to Host B. At each layer, a header (AH : Application Header, PH : Presentation Header, SH : Session Header, TH : Transport Header, NH : Network Header, DH : Data link Header) is attached to the message.

A *header and trailer* is attached at data link layer. The trailer includes checksum bits for error detection at link level. The process continues till we reach the physical layer where data is transferred to the receiver via the physical medium.

On the receivers side we move upwards and the various headers are stripped off at each layer till data arrives at the receiving process.

1.5 Protocol and Service Interface

Dec. 03 [Q. 1(a)] *What is the difference between protocol and service interface ? Explain your answer in terms of the OSI 7 layer model.* **(10 M)**

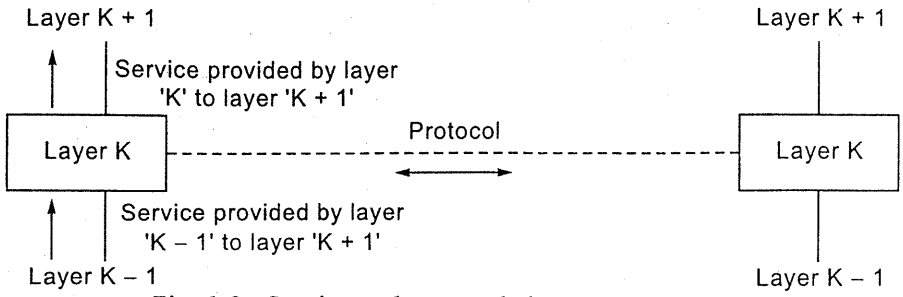


Fig. 1.3 : Service and protocols between the layers.

Service Interface :

- A service is a set of operations that a layer performs for a layer above it.
- The service defines what operations the layer is prepared to perform but it says nothing about how these operations will be implemented.
- Thus we can say that a service is an interface between two layers; the lower layer being the service provider and the upper layer being the service user.

Protocol :

- A protocol is a set of rules governing the format and meaning of messages that are exchanged by the peer entities. Basically a protocol is an agreement between the peer entities (layers) on how communication is to proceed.
- Entities use protocols to implement their service definition.
- Entities are free to change their protocols at will, provided they do not change the service visible to their users.

Conclusion :

- The service and the protocol are completely decoupled.
- A service is like an abstract data type or an object in an object oriented language. It defines the operations that can be performed on an object but it does not specify how these operations are implemented.
- A protocol relates to the implementation of the service and as such is not visible to the user of the service.

(For a 10 mark answer also mention services provided by each layer to the layer above it from section 1.3.

Protocols used in the layers are:

Data Link Layer : HDLC

Network Layer : IP, OSPF, BGP

Transport Layer : TCP, UDP

All these protocols will be covered in their respective chapters.)

1.6 TCP/IP Protocol Stack

Dec. 03 [Q. 5(b)] *Draw the protocol stack of Internet and write Protocol Data Unit.* **(10 M)**

May 04 [Q. 1(b)] *List two ways in which the OSI reference model and TCP/IP reference model are the same and two ways in which they are different ?* **(10 M)**

(Write short note on OSI 7 layers and the TCP/IP 4 layers given below + the similarities and differences)

The Internet uses the TCP/IP protocol Stack

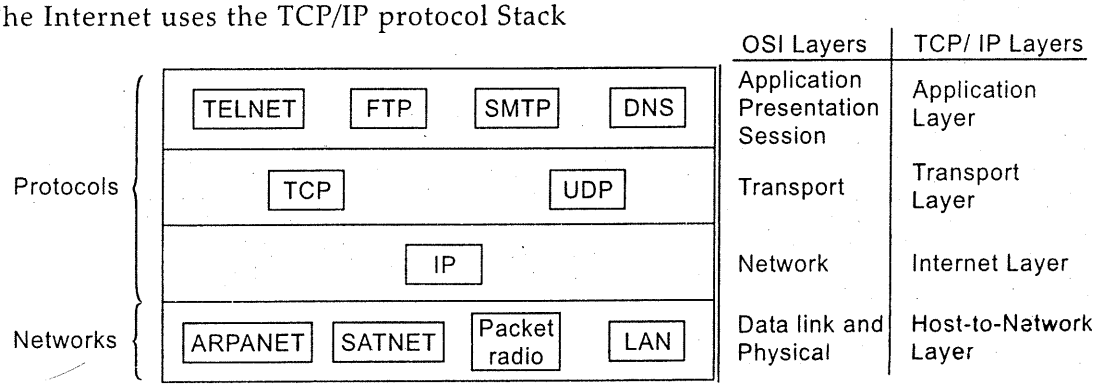


Fig. 1.4 : Layers of TCP/IP.

It has the following layers :

(1) Host to Network Layer

- Also called as Network Interface Layer.
- Same as Physical Layer + Datalink layer of OSI Model.
- Responsible for accepting and transmitting IP datagrams.
- **Protocol Data Unit (PDU)** : Frames.

(2) Internet Layer

- It permits hosts to inject packets into the network and makes these packets reach their destination.
- It defines a packet format and protocol called the Internet Protocol (IP).
- Main focus is on Packet Routing.
- **Protocol Data Unit (PDU)** : Packets.

(3) Transport Layer

- It has two main protocols

  (1) **TCP** :

  (i) Reliable, connection oriented.

  (ii) Allows byte stream from one machine to be delivered to any other machine on the network.

  (iii) Handles flow control and error control.

(2) **UDP :**

(i) Unreliable, connectionless.

(ii) Used for client-server type queries where prompt delivery is more important than reliability.

(iii) Does not implement flow or error control.

- **Protocol Data Unit (PDU) :** TPDU / Message / Segment.

(4) Application layer

- Includes services that use transport layer to deliver data.
- Contains protocols like
  (1) TELNET
  (2) FTP
  (3) SMTP
  (4) HTTP
  (5) DNS
- **Protocol Data Unit (PDU) :** APDU.

Similarities between OSI and TCP/IP *(Mention Only if Asked)*

(1) Both models are based on concept of "Stack of Independent Protocols".

(2) All Layers from bottom till Transport Layer provide *End to End* Transport Service.

(3) All layers above a Transport Layer are *application oriented* and use the Transport Service.

Differences between OSI and TCP/IP *(Mention Only if Asked)*

(1) The protocols in OSI are better hidden and can be replaced relatively easily as technology changes.

(2) OSI supports both connectionless and connection oriented communication in network layer but only connection oriented communication in the transport layer. TCP/IP supports only connectionless communication in the network layer and both connectionless and connection oriented communication in the transport layer.

(3) The OSI model has 7 layers whereas the TCP/IP model is composed of 4 layers.

(4) In TCP/IP the protocol came first and the model was made based on the protocol. In ISO-OSI the model was devised before the protocols were invented.

1.7 Terminology

(a) Network Interface Card (NIC) :

A NIC is present on each machine that is connected to the network.

The basic functionality of the NIC is to handle the connection between the machine and the network.

(b) Interne

The Int

(c) WWW

WWW
informa

(d) Web Si

It is a c

(e) HTTP :

It is a p

(f) DNS : I

General
It is dif
for hum

Thus DI
those ne

(g) FTP : Fi

It is an
to anoth

(h) TELNET

It is a p
can conn

1.8 Networ

Dec. 05 [Q.

Explain the st

Dec. 06 [Q. (

(1) Many ne
things s
networks
network

(2) Not only
increase
mass pro

(3) Standard

De facto
without a

(b) Internet :

The Internet is a network of networks.

(c) WWW : World Wide Web :

WWW refers to a set of Internet protocols and software that together present information to a user in a format called Hypertext.

(d) Web Site :

It is a collection of Web Pages stored on the Web Server.

(e) HTTP : Hyper Text Transfer Protocol :

It is a protocol that governs the dialogue between a Client and the Server.

(f) DNS : Domain Name System :

Generally each computer on a network is given an address called an IP address. It is difficult for humans to remember these addresses. Thus to make it simpler for humans to identify computers on a network we use Domain Names.

Thus DNS can be defined as a system of identifying networks and computers on those networks by some Domain Names, for ease of reference by humans.

(g) FTP : File Transfer Protocol :

It is an application layer protocol used for transferring files from one computer to another.

(h) TELNET : TErminaL NETwork :

It is a protocol used to facilitate *Remote Login* so that user on a client computer can connect to a server on a remote system.

1.8 Network Standardization

Dec. 05 [Q. 5(a)] *What is Standardization ? What is the need of Standardization ? Explain the standardization of computer network in brief ?* **(5 M)**

Dec. 06 [Q. 6(b)] *Explain CCITT.* **(2 M)**

(1) Many network vendors and suppliers exist, each with its own ideas of how things should be done. Without coordination between the different types of networks there would be complete chaos. The only way out is to agree on some network standards.

(2) Not only do standards allow different computers to communicate, but they also increase the market for products using the standard. A larger market leads to mass production that decrease price and further increase acceptance.

(3) Standards fall into two categories : *de facto and de jure.*

De facto (Latin for "from the fact") standards are those that have just happened, without any formal planning. The IBM PC is the *de facto* standard for small-office

and home computers. UNIX is the *de facto* standard for operating systems in university computer science departments.

De jure (Latin for "by law") standards are formal, legal standards adopted by some authorized standardization body.

Telecommunications Standards

- The telecommunication facilities in different countries can be controlled by the government or can be privately controlled.
- With different suppliers of services, there is clearly a need to provide compatibility on a worldwide scale.
- In 1865, representatives from many European governments met to form the predecessor to today's ITU (International Telecommunication Union).
- ITU has three main sectors :
  (1) Radiocommunications Sector (ITU-R): ITU-R is concerned with allocating radio frequencies worldwide.
  (2) Telecommunications Standardization Sector (ITU-T): ITU-T's task is to make technical recommendations about telephone, telegraph, and data communication interfaces. From 1956 to 1993, ITU-T was known as CCITT, an acronym for its French name: Comite Consultatif International Telegraphique et Telephonique.
  (3) Development Sector (ITU-D).
- ITU- T has four classes of members :
  (1) *National Governments* : ITU-T has about 200 governmental members, including almost every member of the United Nations.
  (2) *Sector Members* : There are approximately 500 sector members, including telephone companies (e.g., AT&T), telecom equipment manufacturers (e.g., Nokia), computer vendors (e.g., Compaq), chip manufacturers (e.g., Intel) and media companies (e.g., Sony).
  (3) *Associate Members* : Associate members are smaller organizations that are interested in a particular *Study Group*.
  (4) *Regulatory Agencies* : Regulatory agencies are the folks who watch over the telecom business, such as the U.S. Federal Communications Commission.

International Standards

International standards are produced and published by ISO (International Standards Organization). Its members are national standard organizations of 89 different countries. The US representative in ISO is ANSI(American National Standards Institute).

ISO issues standards on a truly vast number of subjects, ranging from nuts and bolts

to telephone pole coatings. Over 13,000 standards have been issued, including the OSI standards.

Other major players in the standards world are :

(1) *NIST (National Institute of Standards and Technology)* : It issues standards that are mandatory for purchases made by the U.S. Government

(2) *IEEE (Institute of Electrical and Electronics Engineers)* : IEEE has a standardization group that develops standards in the area of electrical engineering and computing.

Internet Standards

The worldwide Internet has its own standardization mechanisms, very different from those of ITU- T and ISO.

Evolution of Internet Committees :

(1) 1983 - IAB (Internet Activities Board)

(2) In 1989, the IAB was reorganized into the IRTF (Internet Research Task Force) which concentrates on long-term research and the IETF (Internet Engineering Task Force) which concentrate on short-term engineering issues.

(3) Later, the Internet Society was created, populated by people interested in the internet.

1.9 Connection Oriented and Connectionless Services

Dec. 03 [Q. 2(b)] *Write advantages, and disadvantages of connection oriented service with connection less service and write example application.* **(10 M)**

May 04 [Q. 1(a)] *What are the principle differences between connection less and connection oriented communication ? Characterize, all the aspects in terms of quality and service.* **(10 M)**

Network layer has to provide end to end network service. There are two ways in which packets/data units can be sent between two transport entities :

(1) Connection Oriented Network Service (CONS)

(2) Connectionless Network Service (CLNS)

I. Connection Oriented Network Service (CONS)

(1) It is also known as a Virtual Circuit.

(2) The connection is dedicated for the entire session.

(3) All Packets of a Message take the same route.

(4) The burden to deliver the Packets in the correct order is the responsibility of the Network Layer.

Working of CONS :

(1) *Establishment Phase* : Establishes the connection and the connection parameters.

(2) *Data Transfer Phase* : Transfers data and performs error control and flow control.
(3) *Connection Release Phase* : Connection is shut down.

Advantages :

(1) Each packet contains short virtual circuit number as address.
(2) Complexity of transport layer is less.
(3) Congestion control is easy.
(4) It allows error and flow control
(5) Reliability is high.
(6) Quality of service is simple.

Disadvantages :

(1) Circuit setup is required therefore routing flexibility is low. Low routing flexibility means that once a circuit is fixed all packets follow the same path and this path does not change even if the path suffers from congestion.
(2) In case of router failure and if that router lies in the virtual circuit all packets are terminated.
(3) Complexity of network layer is high

The CONS is modeled after the *Telephone system*.

Process :

(1) Pick up the phone and dial the number (*Connection Establishment*).
(2) Talk (*Data Transfer*).
(3) Disconnect the phone (*Connection Release*).

Another example for CONS is Internet.

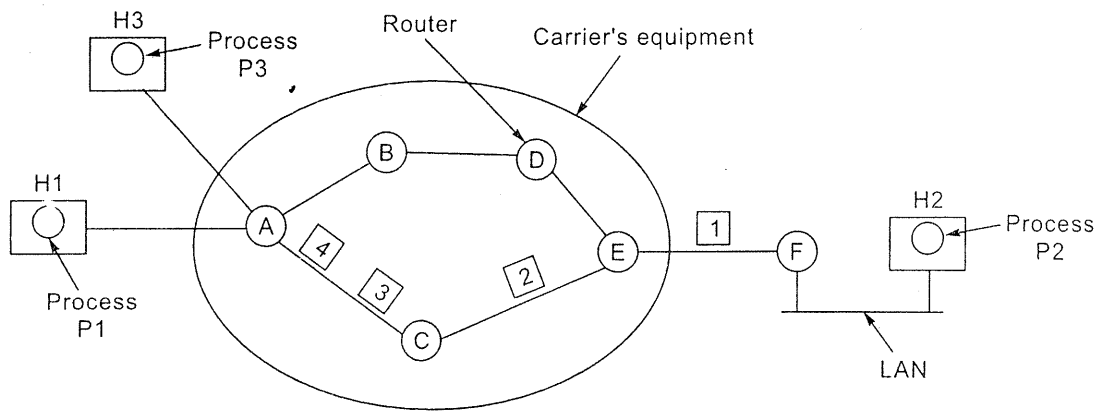


Fig. 1.5 : Routing within a virtual circuit subnet.

Consider that the packets 1, 2, 3, 4 are to be sent from Process1 (P1) on HOST1 (H1) to Process2 (P2) on Host2 (H2). A, B, C, D, E, F are routers. All the packets follow the

path A → C, C → E and E → F to reach the destination. This shows that before the actual transfer of packets a connection has to be made between the sender and the receiver and all the packets must follow this path.

II. Connectionless Network Service (CLNS)

(1) It is also known as datagram.

(2) There is no dedicated connection for the entire session.

(3) Different packets of the same message may take the different routes.

(4) The burden to deliver the packets in the correct order is the responsibility of the Transport Layer.

Working of CLNS :

(1) It works like Postal System.

(2) Each packet contains full source and destination address.

(3) Different packets of the same message may take the different routes.

(4) The burden to deliver the packets in the correct order is the responsibility of the Transport Layer.

Advantages :

(1) Circuit setup is not required therefore routing flexibility is high. Routing flexibility is high means that the packets can change their route due to congestion or router failure.

(2) In case of router failure and if that router lies in the virtual circuit only packets through that router are lost or rerouted (***Note** : All packets need not follow the same path in CLNS therefore the number of lost packets due to a router failure is low*).

(3) Complexity of network layer is low.

Disadyantages :

(1) Each packet contains entire address of source and destination.

(2) Complexity of transport layer is high.

(3) Congestion control is difficult.

(4) It cannot conduct error and flow control.

(5) Reliability is low.

(6) Quality of service is difficult.

The CONS is modeled after the Postal System.

Process :

(1) Each letter carries the entire address of source and destination.

(2) Each letter is routed independent of other letters.

(3) If two letters are sent to same person it is possible that the recipient receives the first letter after the second letter i.e. letters may not be delivered in the same order in which they are sent.

Another example is the ATM Network.

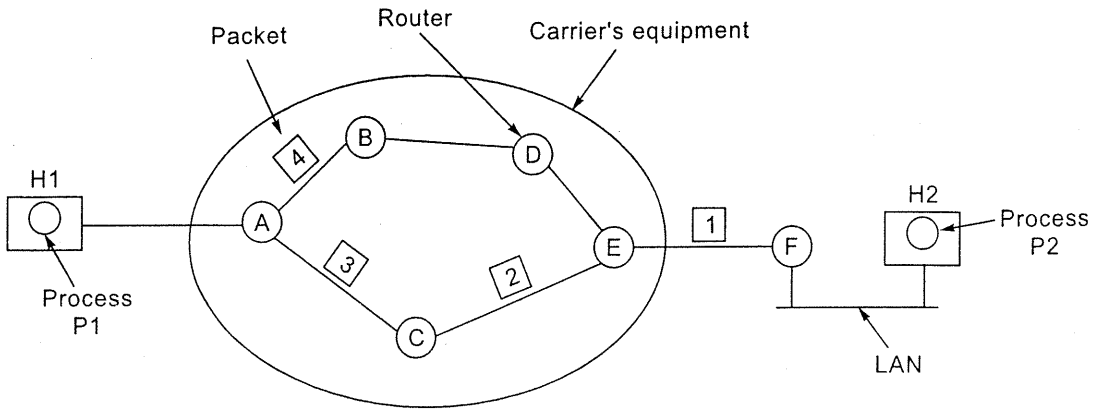


Fig. 1.6 : Routing within a datagram subnet.

Consider that the packets 1, 2, 3, 4 are to be sent from Process1 (P1) on HOST1 (H1) to Process2 (P2) on Host2 (H2). A, B, C, D, E, F are routers. The packets 1, 2, 3 follow the path A → C,C → E and E → F to reach the destination. The packet 4 takes the path A → B,B → D,D → E and E → F. This shows that in a CLNS system all packets need not follow the same path.

1.10 Domain Name System (DNS)

May 04 [Q. 5(a)] *What is "Domain Name System" and "Name Server" ? How are they related ?* **(10 M)**

Dec. 04 [Q. 7] *Write a short note on DNS.* **(3 M)**

May 05 [Q. 7] *Write a short note on DNS.* **(7 M)**

Dec. 05 [Q. 2(c)] *Write a short note on DNS.* **(4 M)**

May 07 [Q. 6(a)] *Explain DNS system.* **(10 M)**

Generally each computer on a network is given an address called the *IP address*. It is difficult for humans to remember these addresses. Thus to make it simpler for humans to identify computers on a network we use *Domain Names*.

Thus DNS can be defined as a system of identifying networks and computers on those networks by some domain names, for ease of reference by humans.

Commonly used *suffixes* to specify *domain names :*

com ⇒ commercial organization E.g. yahoo.**com**

edu ⇒ educational organization.

org ⇒ non profit organization.
net ⇒ network support group.
gov ⇒ government institution.
mil ⇒ military group.

Country specific suffixes :

in ⇒ India.
uk ⇒ United Kingdom.
jp ⇒ Japan.

The DNS Name Space

(1) The internet is divided into hundreds of domains. Each domain can be further subdivided into sub-domains which can be further sub divided and so on.

(2) This creates a tree like structure as shown

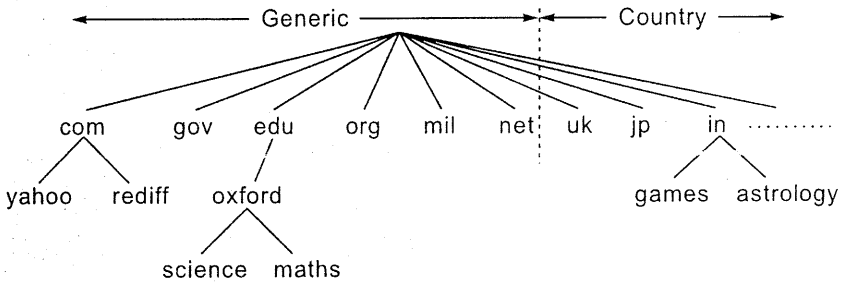


Fig. 1.7 : Tree structure of DNS.

- *Leaf* represent lowest level domain that cannot be classified further but it can have several hosts connected to it.
- The topmost domains are classified into :
- *Generic Domains* : These are the domains registered in the United States.
  E.g. com, edu, org, net, gov, mil, int
  Country Domains : They are specific to a country. E.g. in, jp, uk

Domain Name Servers

- Humans find it easy to work with *Domain Names* but Computers work only with *IP addresses*. Therefore there must exist a way to map between the Domain Names and the IP addresses.
- This is done by Domain Name Servers that are computers containing the database of Domain Names with their corresponding IP addresses and the software that map between Domain Names and IP addresses.
- Therefore Domain Name Servers accept and service requests for mapping from Domain names to IP addresses.

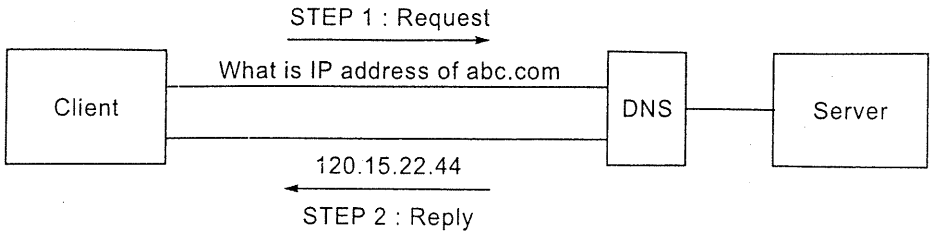


Fig. 1.8 : Working of the domain name server.

- We do not maintain all the information on a single server. Instead we use a Distributed Database because :
  (1) It is faster to retrieve data from a Distributed Database.
  (2) Distributed Database is more reliable as failure of one server will not result in the loss of all the information.
- Generally each server has control over a single domain alongwith the subdomains and the hosts present on that domain.
  E.g.: The "Server" for the Domain *edu* will have control over the servers for *oxford, science, maths* plus the hosts connected to edu. The server for *oxford* will control the servers of *science and math*, plus the hosts present on *oxford*. The servers for *science, maths* will contain information specific to their domains plus their individual hosts.
- What a "Server" has control over is called as the *zone of that server*.

Conclusion

Hence DNS is the *application* used to map between Domain Names and IP Addresses whereas the Domain Name Servers are used to perform this application.

1.11 X.25 and Frame Relay

May 04 [Q. 6(b)] *Differentiate X.25 and Frame Relay.* **(10 M)**

Dec. 04 [Q. 7], Nov. 06 [Q. 6(b)] *Write short Note on X.25.* **(3 M, 2 M)**

(1) X.25 is a communications packet switching protocol which was designed to transfer data over a *Wide Area Network*.

(2) X.25 specifies an interface between a Host System (DTE : Data Terminal Equipment) and a Packet Switching Network (DCE : Data Communication Equipment)

(3) It is a protocol that was specialized in handling noisy, lower-speed networks by supplying error detection and correction mechanisms.

(4) Although most packet switching technology are generally connectionless in nature, X.25 establishes virtual circuits that allow it to be connection oriented.

Therefore X.25 functions as follows :

The connection is established, the data is transferred, and then the connection is terminated.

(5) However, with the advancement in technology more sophisticated systems have come into the picture which demand higher network performance.

(6) The transmission capacity of X.25 is around 64 kbps, this capacity is just not enough in today's computer world.

(7) The new player in the protocols field which has been the first protocol designed to substitute X.25 is Frame Relay. This technology is very widespread today and commonly known as "a tuned-up version of X.25" and "the X.25 of the 1990s".

(8) The main idea behind frame relay is : X.25 had a fixed bandwidth available. It wasted portions of its bandwidth. On the other hand Frame relay can dynamically allocate bandwidth.

(9) To achieve this goal, the designers of this protocol had to concede some functionalities like flow control and error correction. Therefore, while X.25 can handle circuits with high bit error rates more efficiently, frame relay is more fitting when networks already have high transmission quality.

(10) Types of virtual circuit used by frame relay :

- **Switched Virtual Circuit (SVC) :** Connection is set up and brought down as and when needed.
- **Permanent Virtual Circuit (PVC) :** These are permanent dedicated connections.
- **Multicast Virtual Circuit (MVC) :** Connections between users which allows users to use SVC and PVC connections.

Differences of Properties Between X.25 and Frame Relay

(1) *Reliability and Integrity* : X.25 supply mechanisms that guarantees flow control and data integrity. Frame Relay, on the other hand, doesn't guarantee data integrity (it has only optional error detection mechanism but no error correction)

(2) *Cost* : In X.25 error correction is done using a mechanism of store and forward that requires expensive buffering while Frame Relay is cheaper, because it does not perform error correction and hence does not require buffering.

(3) *Speed* : X.25 can be very slow. For example, it's terrible for telephony, because the human ear can't stand such delays. The store-and-forward mechanism of X.25 caused it to have a delay. Frame Relay, on the contrary, reduces the transmission delay significantly since it does not store and forward, but simply switches to the destination. Frame Relay switches packets end to end about 20 times faster.

(4) *Buffer Space* : Another problem of using X.25 protocol is the fact that it requires a massive buffering capability in order to support the store-and-forward mechanism of data transfer which makes it very inefficient to use. One of the main reasons that Frame Relay is relatively so cheap is that its requirements of storage are minimal.

(5) *Digital/Analog* : X.25 can operate over either digital or analog facilities. Frame relay and ATM technologies need high quality digital transmission facilities.

(6) *Correctness* : A benefit of X.25 is due to the seniority of this protocol. X.25 has been created in the mid 1970's so it has seniority of over 30 years. Consequently, it is stable and debugged very well.

(7) In X.25 because of the store-and-forward nature of packet switching, data transfer equipments do not have to use the same speed. For example, you can have a host connected at 64 kbps communicating with several remote sites connected with 19 kbps lines.

(8) *Layers* : X.25 is defined for layers 1, 2 and 3 of the ISO model, while frame relay is defined for layers 1and 2 only. This means that frame relay has significantly less processing to do at each node, which improves throughput.

(9) *Flow Control* : X.25 routers have to acknowledge each frame; in case of errors, frames have to be retransmitted and acknowledged. Frame Relay doesn't perform flow control. Frame Relay relies on flow control performed by higher layer protocols.

(10) *Packets /Frames* : X.25 prepares and sends packets, while frame relay prepares and sends frames. X.25 packets contain several fields used for error and flow control, none of which is needed by frame relay. The frames in frame relay contain an expanded address field that enables frame relay nodes to direct frames to their destinations with minimal processing.

Conclusion

Although Frame Relay and other modern technologies are taking over from X.25; X.25 still cannot be totally removed due to the following reasons :

(1) There is still more than $10 billion worth of X.25 equipment around the world.

(2) Although now X.25 is outdated there is still a core of applications in which X.25 is still used, because the costs to change and replace it with another technology is high and not economical.

(3) Examples of the applications that use X.25 are applications that can't stand errors at all. Like banks that are responsible for large transactions of data. X.25 protocol that guarantees accurate and reliable transfer of information is suitable in such cases.

1.12 WWW : World Wide Web

May 04 [Q. 7(a)] *Describe in detail the working of WWW. What is the difference between WWW and Internet ?* **(10 M)**

Dec. 03 [Q. 4(a)] *Distinguish between the application and application layer protocol. Use world wide web as an example.* **(10 M)**

(Explain WWW section 1.12 and HTTP section 1.13)

(1) The World Wide Web is used for accessing linked documents spread out over millions of machines over the Internet.

(2) *Web Pages* : The web consists of a collection of documents called Web Pages.

(3) *Web Site* : A collection of web pages is called a Web Site.

(4) Each Web Page may contain *Links* to other pages anywhere else in the World. Such strings of text that are links to other pages are called as *Hyper Links*.

(5) *Hypertext* : Hypertext documents contain only text.

(6) *Hypermedia* : Hypermedia documents can contain pictures, graphics and sound.

(7) Browsers interpret and display a web document. The steps of the browser are :

- Controller accepts input from the keyboard/mouse and calls the client program.
- Client program access the required document. (E.g. of client programs: HTTP, SMTP, and FTP.)
- Controller calls interpreter to display/interpret the document. (E.g. of interpreter : HTML, JAVA.)

(8) Types of documents in WWW :

- *Static Documents* : They are documents that are created and stored on the server. Clients get a copy of the document when requested. The content of the documents do not change, hence the name *Static*.

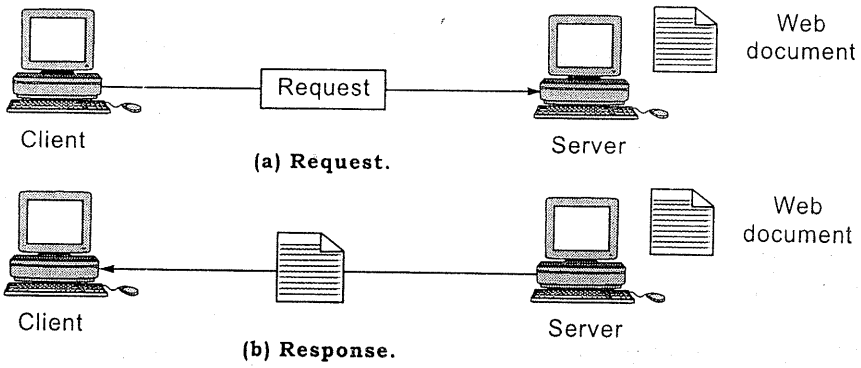
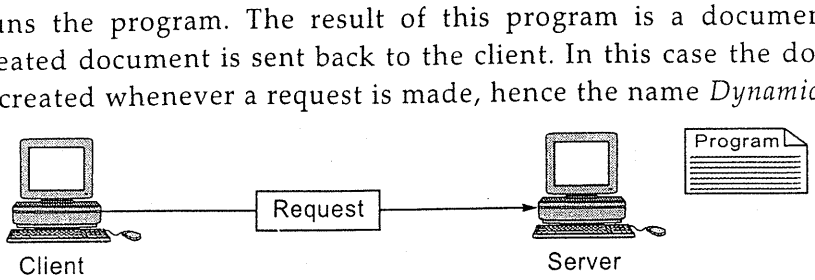
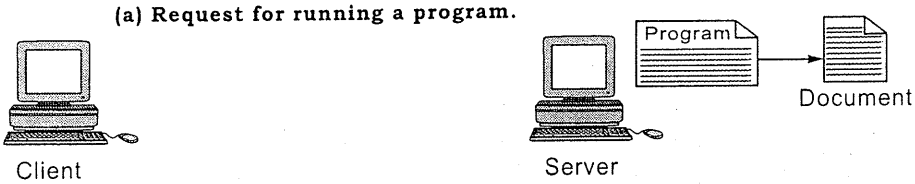


Fig. 1.9 : Static documents.

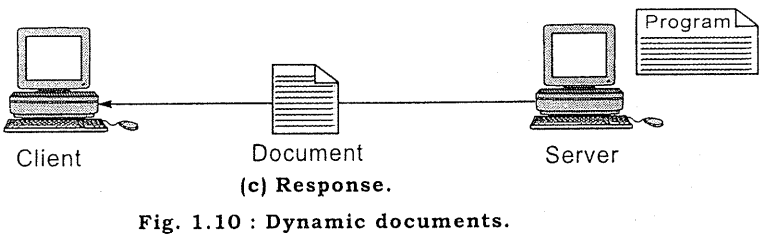
- *Dynamic Documents* : The client requests the server to run a program. The server runs the program. The result of this program is a document. This newly created document is sent back to the client. In this case the document is newly created whenever a request is made, hence the name *Dynamic*.



(a) Request for running a program.



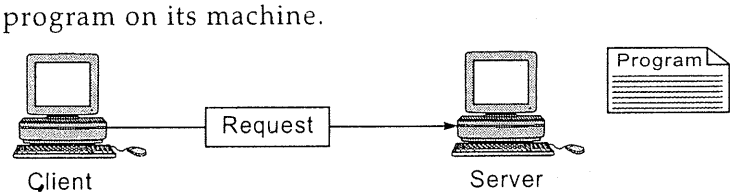
(b) Running the program and creating the document.



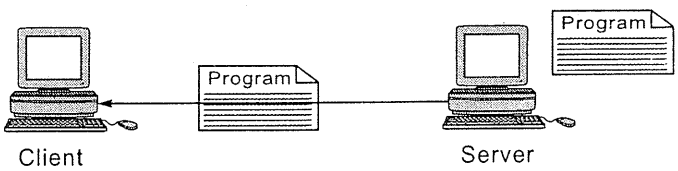
(c) Response.

Fig. 1.10 : Dynamic documents.

- *Active Documents* : In this case the client requests the server for a copy of the program. The server sends a copy of the program to the client. The client can now run the program on its machine.



(a) Request for a copy of a program.



(b) Sending a copy of the program.

(c) Running the program and creating the document.

Fig. 1.11 : Active documents.

Conclusion

- The *Internet* is an immense group of large networks joined together. These networks are comprised of collections of smaller networks.
- The *World Wide Web (WWW)*, or simply Web, is a means of accessing information over the medium of the Internet. It is an information-sharing model that is built on top of the Internet network structure. The Web uses HTTP, one of the languages spoken over the Internet, to transmit data. The Web also utilizes browsers, such as Internet Explorer or Netscape Navigator to access documents called Web pages that are linked to each other via hyperlinks. Web documents can contain graphics, sounds, text, video and programming code like Java.
- *Simply Put* : All of the web sites in the world, taken together, make up the *World Wide Web*. The *Internet* is the worldwide network of interconnected computers, including both *web servers* and *computers* like the one on your desk that run *web browser* software.

1.13 HTTP and SMTP

Dec. 03 [Q. 4(b)] *What is the difference between push protocol and pull protocol? Explain using HTTP and SMTP.* **(10 M)**

Dec. 04 [Q. 7] *Short note : SMTP.* **(3 M)**

Dec. 05 [Q. 2(c)] *Short note : SMTP.* **(4 M)**

May 07 [Q. 6(b)] *Explain SMTP.* **(10 M)**

PUSH Protocol : Pushes the message from the sender to the receiver even if the receiver does not want it.

POP Protocol : Gets the message from the server when the recipient asks for it.

1.13.1 SMTP : Simple Mail Transfer Protocol

(a) A client establishes a TCP connection.

(b) Messages are transferred from sender to receiver in the following stages :

[I] **First Stage**

(1) Mail goes from *user agent* to *local server*.

(2) In this stage mail is sent to *local server* and not *remote server* as the *remote server* may be busy at that time.

(3) In this stage *user agent* acts as SMTP Client and *local server* acts as SMTP Server.

[II] **Second Stage**

(1) Now when the *remote server* becomes free the mail is transferred from *local server* to *remote server*.

(2) In this stage *local server* acts as SMTP Client and *remote server* acts as SMTP Server.

[III] Third Stage : Now the intended receiver retrieves the mail from its mail box on the remote server using a PULL protocol. A PULL protocol like IMAP4 or POP3 and not the PUSH protocol of SMTP is used because :

(1) The receivers computer may not be on when the sender sends a message.

(2) Thus the message is stored on the remote server, in the mailbox of the receiver, till the receiver turns his computer on.

(3) When the receiver turns on his computer he must retrieve his mail from the server. The protocols that can be used for mail retrieval are IMAP4 (Internet Mail Access Protocol version 4) or POP3 (Post Office Protocol version 3).

Thus, SMTP is a PUSH protocol that is used to send message from sender to receiver's mailbox.

IMAP4 and POP3 are PULL protocols that are used by the receiver to retrieve his mails from his mailbox.

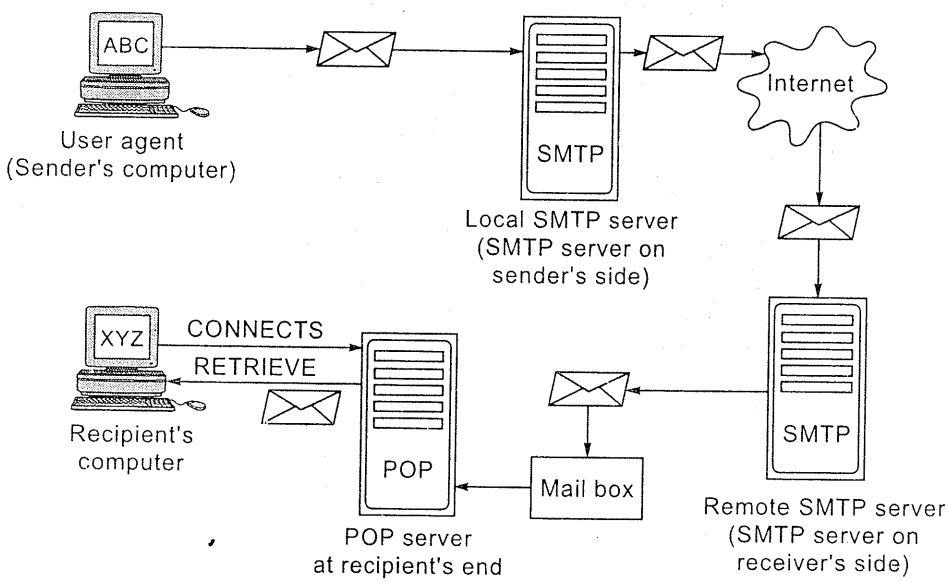


Fig. 1.12 : SMTP and POP.

1.13.2 HTTP

(1) The **Hypertext Transfer Protocol (HTTP)** is used to access data on the World Wide Web. HTTP is a protocol which governs the dialogue between the client and the server.

(2) **Basics**

(a) TCP is used so that neither browsers nor servers have to worry about lost messages, duplicate messages or acknowledgements.

(b) HTTP functions like a combination of FTP and SMTP.

FTP

(i) It is similar to FTP because it transfers files.

(ii) It is simpler than FTP because it uses only one connection whereas normal FTP uses 2 connections.

SMTP

(i) HTTP is like SMTP because the data transferred between the client and the server is similar to SMTP messages.

(ii) Unlike SMTP messages, the HTTP messages are not read by humans; they are read and interpreted by the HTTP server and the browser on the client side.

(3) **Working**

(a) A browser contacts a server to establish a TCP connection with it.

(b) The HTTP software on the client sends a request to the server. The HTTP software on the server interprets this request and sends the response to the client.

(c) HTTP Commands :

GET : Request by a client to obtain a web page from the server.

PUT : Request by a client to store a web page on the server.

POST : Request by a client to update contents of a web page on the server.

DELETE: Request by client to remove a web page from the server.

(4) *Difference Between WWW and Other Protocols like HTTP, MIME, FTP :* WWW is an application, which requires protocols like HTTP, MIME and FTP to implement it.

1.14 ISDN Overview

[I]

May 04 [Q. 8(b)] *Describe ISDN, its categories, content and working.* **(10 M)**

Need of ISDN

(1) A long time ago, the entire telephone network was analog. This was bad, because as a voice went farther down the line the quality became worse and worse as noise crept in. And there was no way to eliminate the noise.

(2) Digital encoding promised a way to encode the audio such that you'd know what the signal was supposed to be. As noise crept in, you could eliminate it throughout the phone network.

(3) This solved many of the phone company's problems. However for a variety of reasons, it has been attractive to make the phone network completely digital, from *end to end*. For computer users, this is ideal, because we can eliminate those

clumsy modems, and will hopefully benefit from higher speed. For the phone companies, they can eliminate the last of the noise and loss from the audio data.

What is ISDN ?

(1) Integrated Services Digital Network (ISDN) is a type of *circuit switched telephone* network system, designed to allow digital transmission of voice and data over ordinary telephone copper wires, resulting in better quality and higher speeds than available with *analog* systems.

(2) The original ISDN Standards were B-ISDN(Broadband ISDN) and N-ISDN (Narrowband ISDN).

(3) ISDN was built on TDM (Time Division Multiplexing).

(4) In the phrase "Integrated Services Digital Network",

- *Integrated Services* refers to ISDN's ability to deliver at minimum two simultaneous connections, in any combination of data, voice, video, and fax, over a single line.
- *Digital* refers to its purely digital transmission, as opposed to the analog transmission of *plain old telephone service*.
- *Network* refers to the fact that ISDN is not simply a point-to-point solution but is a network which includes the local telephone exchange, the telephone user and all of the telecommunications and switching equipment in between.

Configurations

In ISDN, there are two types of channels, B (for "Bearer") and D (for "Delta"). B channels are used for data (which may include voice), and D channels are intended for signaling and control (but can also be used for data).

There are two kinds of access to ISDN.

(1) Basic Rate Interface (BRI)

(i) Consists of two B channels, each with bandwidth of 64 Kbit/s, and one D channel with a bandwidth of 16 Kbit/s.

(ii) Together these three channels can be designated as 2B + D.

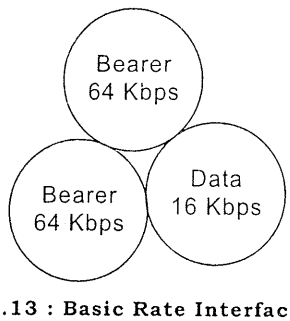


Fig. 1.13 : Basic Rate Interface (BRI).

(2) Primary Rate Interface (PRI)

(i) Contains a greater number of B channels and a D channel with a bandwidth of 64 Kbit/s each.

(ii) The number of B channels for PRI varies according to the nation: in North

America and Japan it is 23B +1D, with an aggregate bit rate of 1.544 Mbit/s; in Europe and Australia it is 30B+1D, with an aggregate bit rate of 2.048 Mbit/s.

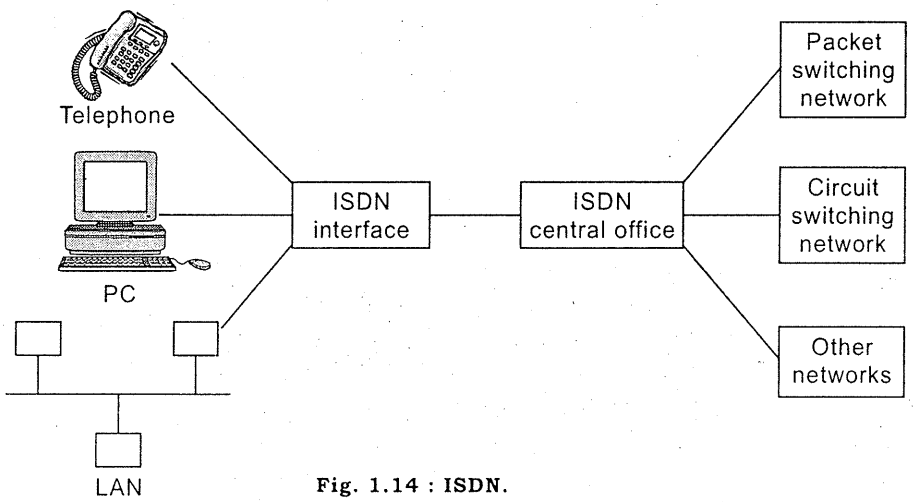


Fig. 1.14 : ISDN.

[II] Broadband Technologies Versus Narrowband Technologies

We differentiate between broadband and narrowband networks in that broadband networks can be used for many different traffic characteristics, while narrowband networks are used for one.

Narrowband technologies are Ethernet, Token Ring, FDDI, and certain types of WAN links. Narrowband technologies can be adapted to run over broadband technologies, but it is not possible to run broadband technologies over narrowband. Broadband technologies, on the other hand, are not limited to a single link protocol (like Ethernet) or to data communication only. Broadband networks use frame, or frame trains, and special methods to encapsulate and control the information sent through it. The information does not have to be computer related at all. For example, voice - over-IP is not a broadband technology because the audio is digitized and sent as IP packets, mostly over Ethernet, whereas on broadband, voice can be sent as a separate channel without encapsulating it into a computer communication protocol.

Broadband networks can be adapted to form the physical layer for LAN technologies (broadband ISDN or ATM), or to run subsets of TCP/IP directly (such as PPP over SDH).

The three major groups of broadband technologies are:

- Synchronous Optical Network (SONET, North America)
- Synchronous Digital Hierarchy (SDH, Europe)
- Broadband ISDN (ATM)

ATM can function by itself or run over SONET or SDH. The latter is more common.

Broadband ISDN

Higher speed transmission media will be available. Narrowband ISDN was basically a stopgap measure. But it's development allowed time for the design of Broadband ISDN to be even more ambitious.

Broadband ISDN (B-ISDN) is similar to Narrowband ISDN only in the most basic aspects. The reference configurations are largely the same, but every bit of the underlying design will be replaced. The bus will by necessity have to be a much faster network, as will the network. The signaling will be changed, as will most other details. The house will most likely need to be rewired yet again, and all the physical connectors are likely to change.

B-ISDN is very strongly related to ATM. This is because ATM will provide a consistent data encapsulation scheme that can be used throughout the network.

Conclusion :

(1) B-ISDN is a change from metal cable to fiber optic cable at all levels.

(2) Services provided by B-ISDN

  (a) **Interactive services** such as video conferencing, voice mail etc.

  (b) **Distributive services :** These are services given to subscriber without the subscribers having to request the service whenever it requires it.

(3) Physical specifications of B-ISDN

  (a) B-ISDN is divided into layers similar to ATM. Its differences are

    (i) Access Methods : It defines 3-access methods

      - 155.520 Mbps full duplex
      - 155.520 Mbps output/ 622.080 Mbps input
      - 622.080 Mbps full duplex.

    (ii) Functional Grouping : Same as N-ISDN

    i.e. B-NT1, B-NT2, B-TE1, B-TE2 and B-TA

    (iii) Reference Point : Same as N-ISDN i.e. R, S, T and U.

1.15 Network Operating Systems

May 05 [Q. 7] *Network Operating System.* (7 M)

(1) A network operating system (NOS) causes a collection of independent computers to act as one system.

(2) A network operating system is analogous to a desktop operating system like DOS or OS/2, except it operates over more than one computer.

(3) A netw
includi
which

(4) At a ba
as disk
and se
admin
periph

(5) *Redirec*
someth
Networ
being r
NOS, u
designa

(6) *Server S*
need to
resourc
every c
comput
called a
can also

(7) *File Serv*
service a
whole d
disk bec
administ

(8) *Server Op*
user mac
many us
does not

basically a
Broadband

most basic
bit of the
be a much
most other
ne physical

provide a
work.

vithout the

(7 M)

computers

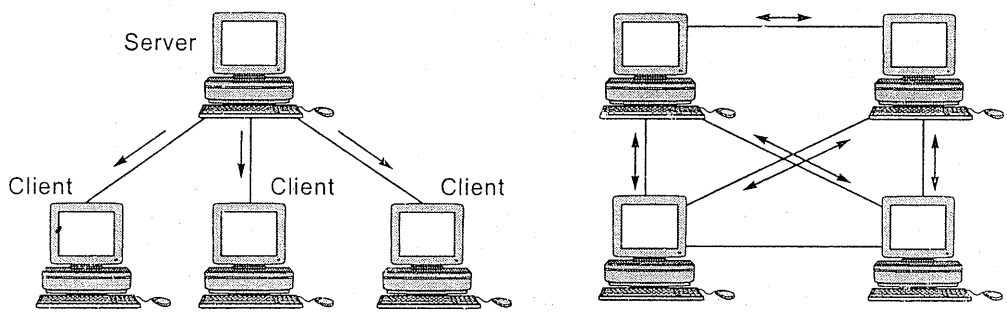
system like

(3) A network operating system controls the operation of the network system, including who uses it, when they can use it, what they have access to, and which network resources are available.

(4) At a basic level, the NOS allow network users to share files and peripherals such as disks and printers. Most NOSs' do much more. They provide data integrity and security by keeping people out of certain resources and files. They have administrative tools to add, change, and remove users, computers, and peripherals from the network.

(5) *Redirection :* At the heart of the NOS is redirection. Redirection is taking something headed in one direction and making it go in a different direction. Network operating systems depend heavily on redirection; in this case data is being redirected from one computer to another over the network cable. With a NOS, users don't need to know about redirection; they just type the drive designator or print from their word processors as always.

(6) *Server Software :* A NOS is made of a redirector and a server. Not all machines need to run the server software, because not all computers need to share their resources. But all network workstations must run redirector software because every client has to be able to put data onto the network. With some NOSs, the computer running the server software cannot be used as a workstation. This is called a *dedicated server*. With some other NOSs, all workstations on the network can also be servers. This is a *non dedicated server* setup.



Dedicated server. **Non dedicated server.**

Fig. 1.15

(7) *File Service :* A file server's primary task is to make files available to users. File service allows users to share the files on a server. The server PC can make its whole disk, certain directories, or certain files available. The file server's hard disk becomes an extension of each user's PC. The NOS can let the network administrator determine which users are allowed to use which files.

(8) *Server Operation :* The server software makes a single-user computer into a multi-user machine. Instead of just one user, a server has many users. A NOS allows many users to share the server's peripherals, printers, disks, and plotters, but it does not allow multiple users to share its processor.

Conclusion :

Network Operating System

- A network is a collection of machines that communicate with each other.
- A network operating system controls the interactions between the machines on the network.
- It basically controls the procedure of packaging data at sender machine and sending these packets to receiver machine.
- In some networks a server cannot be used as a workstation. This is called dedicated server.
- In other networks a server can also work as a workstation. This is called non dedicated server setup.

1.16 Linux Operating System

Dec. 05 [Q. 7(b)] *Short note : Linux Operating System.* (10 M)

May 06 [Q. 2a] *List 10 commands in Linux/Windows platforms used for networking.* (10 M)

Dec. 06 [Q. 7] *Case study of Linux.* (5 M)

- Linux is an operating system that was initially created as a hobby by a young student, Linus Torvalds. Linus had an interest in Minix, a small UNIX system, and decided to develop a system that exceeded the Minix standards. In 1994 the version 1.0 of the Linux Kernel was released.
- The Kernel is the heart of all Linux systems, and its source code is freeware that is freely available to everyone. It is this kernel that forms the base around which a Linux operating system is developed.
- There are now literally hundreds of companies and organizations and an equal number of individuals that have released their own versions of operating systems based on the Linux kernel.
- Apart from the fact that it's freely distributed, Linux's functionality, adaptability and robustness, has made it the main alternative for proprietary Unix and Microsoft operating systems. IBM, Hewlett-Packard and other giants of the computing world have embraced Linux and support its ongoing development.
- Linux has an official mascot, Tux, the Linux penguin
- Linux, is only the kernel of the operating system, the part that controls hardware, manages files and separates processes. There are several combinations of Linux with sets of applications that form a complete operating system. Each of these combinations is called a distribution of Linux.

- New bugs may creep up in the different distributions of Linux and will be fixed as time goes on.
- One thing to be aware of is that Linux is developed using an open and distributed model, instead of a closed and centralized model like much other software. This means that the current development version is always made public so that anybody can use it. The result is that whenever a version with new functionality is released, it almost always contains bugs, but it also results in a very rapid development so that the bugs are found and corrected quickly, often in hours, as many people work to fix them.
- In contrast, the closed and centralized model means that there is only one person or team working on the project, and they only release software that they think is working well. Often this leads to long intervals between releases, long waiting for bug fixes, and slower development.

Networking Commands in Linux :

(1) **ifport(8)** : If some of your network cards are fancy and support multiple transceiver types, then you start configuration by using this command to specify which transceiver type you will use.

(2) **ifconfig(8)** : This command is used to configure network interfaces.

(3) **route(8)** : Once its interfaces are configured, your machine can receive packets from the network. But where should outgoing packets be sent ? Making this decision is called "routing," and it is made by consulting the system's routing table. The destination address of every outgoing packet is checked against every line of this table; if a matching line is found then the packet is sent out the interface listed on that line of the table; if no match is found the system returns the error "Unreachable host." The route command is the tool used to display or modify the routing table.

(4) **usernetctl(1)** : Sometimes users are granted the ability to bring certain interfaces up or down on their own.

(5) **ipfwadm(8)** : It is used to configure your machine to act as a firewall.

(6) **arp(8)** : When the system transmits a packet, it has to send it to a particular physical-layer address. The ARP (Address Resolution Protocol) table normally uses an automatic mechanism to find what physical addresses go with which IP addresses.

(7) **ping(8)** : The ping (Packet Internet Groper) program is used to query another system and ensure a connection is active.

(8) **traceroute(8)** : While ping gives information about the performance of the network path between two hosts, traceroute will actually show the route.

(9) **host(1), nslookup(8), dig(1)** : Perform queries on DNS.

(10) **inetd(8)** : It is usually run when networking is activated. It grabs control of the ports for FTP and telnet.

(11) **tcpdchk(8)** : The tcpdchk command scans the files and reports any errors or omissions it finds.

(12) **sendmail(8)** : This is the traditional Unix program for sending and receiving email over the Internet.

Networking Commands in Windows

(1) **net** : Used to start, stop, and view many networking operations.

(2) **ipconfig** : Displays the IP address and other TCP/IP configuration information for your workstation.

(3) **hostname** : Displays the Microsoft networking computer name; available only in Windows NT, 2000, and XP.

(4) **lpq** : Displays the print queue status of an LPD printer; available only in Windows NT, 2000, and XP.

(5) **ping** : Verifies existence of remote host (connectivity).

(6) **nbtstat** : NetBIOS over TCP/IP; gives statistics and technical NetBIOS information for the TCP/IP layer.

(7) **netstat** : Returns protocol statistics and current TCP/IP connections.

(8) **ipxroute** : Displays and modifies IPX routing tables.

(9) **route** : Manipulates TCP/IP routing information.

(10) **tracert** : Displays route taken by an ICMP to a remote host.

(11) **finger** : Displays information about the user.

(12) **arp** : Displays or modifies information in the ARP (Address Resolution Protocol) cache.

1.17 Network Hardware

(1) Networks can be classified based on their *transmission technology* and their *scale* (i.e. geographic area that the network can span).

(a) *Classification Based on Transmission Technology*

(i) **Broadcast Links** : In this type of transmission technology a message is sent to all the computers on the network. The message contains a field which specifies the intended receiver. All the computers which receive the message check if their name is there on the *intended recipient list*. If *yes*, then those computers accept the message. If *no*, then those computers ignore the message.

(ii) Point-to-Point Links : In this type of transmission technology a message is transmitted only to the intended recipient. The other computers on the network do not receive the message.

(b) *Classification Based on Scale*

This classification concentrates on the extent of geographical area that a network can span. We have the following types

(i) Personal Area Network (PAN)

- Meant for one person.
- Small.
- E.g. Connecting mouse and keyboard to the CPU.

(ii) LAN, MAN and WAN are explained in the next point.

(iii) Internetwork : It is a connection of two or more networks.

Interprocessor Distance	Processors Located in Same	Example
1 m	Square meter	Personal area network
10 m	Room	Local area network
100 m	Building	
1 km	Campus	
10 km	City	Metropolitan area network
100 km	Country	Wide area network
1000 km	Continent	
10,000 km	Planet	The Internet

Fig. 1.16 : Classification of interconnected processors by scale.

(2) LAN, WAN and MAN

A computer network that spans a relatively small area is called a *LAN*. Most LANs are confined to a single building or group of buildings. However, one LAN can be connected to other LANs over any distance via telephone lines and radio waves. A system of LANs connected in this way is called a *wide-area network (WAN)*.

Most LANs connect workstations and personal computers. Each node (individual computer) in a LAN has its own CPU with which it executes programs, but it also is able to access data and devices anywhere on the LAN. This means that *many users can share expensive devices, such as laser printers, as well as data*. Users can also use the LAN to communicate with each other, by sending *e-mail or engaging in chat sessions.*

Metropolitan Area Networks or *MANs* are large computer networks usually spanning a campus or a city. They typically use wireless infrastructure or optical fiber connections to link their sites.

For instance a university or college may have a MAN that joins together many of their local area networks (LANs) situated around site of a fraction of a square kilometer. Then from their MAN they could have several wide area network (WAN) links to other universities or the Internet. Specifically, this type of MAN is known as a *campus area network*.

(3) **Wireless Networks**

(a) **System Interconnection** : Interconnects a mouse/ keyboard to a computer using blue tooth.

(b) **Wireless LANs and WANs** : Interconnects LANs using radio waves and antennas.

(4) **Internetwork** *(See Internetworking Devices from Section 2.9)*

(5) **Home Networks**

It basically is a futuristic idea which envisions all the devices in a house to be internetworked. This means that any device at home will be able to exchange information with any other house hold device.

1.18 Network Software

It includes :

(1) **Protocols and Services.** *(Section 1.5)*

(2) **Layered Architecture.** *(Section 1.4)*

(3) **Connection Oriented and Connectionless Services.** *(Section 1.9)*

(4) **Service Primitives :**

(a) We have learnt about *services* and protocols. But how exactly are *services* implemented ?

(b) This is done by a set of operations called *primitives*.

(c) The primitives tell the service what is to be done to perform an action.

(d) An example set of services are shown in the *figure 1.17*.

Primitive	Meaning
LISTEN	Block and wait for an incoming connection
CONNECT	Establish a connection with a waiting peer
RECEIVE	Block and wait for an incoming message
SEND	Send a message to the peer
DISCONNECT	Terminate a connection

Fig. 1.17 : Five service primitives for implementing a simple connection-oriented service.

(e) **Explanation**

(i) **LISTEN** : A server will execute LISTEN to say that it is ready to accept connections from clients. After executing LISTEN the server gets blocked till a client tries to establish a connection with it.

(ii) **CONNECT** : When a client wants some data from a server it must connect to the server. To connect to a server the client executes the CONNECT primitive.

(iii) **SEND/RECEIVE** : These primitives are used to send/receive messages to/from other computers.

(iv) **DISCONNECT** : After communication between the client and the server is complete the connection must be disconnected.

1.19 Network Applications

(a) Now that we have studied the basics of computer networks let us see how *networking* is actually used in todays world.

(b) The applications that we will cover in this section are :

(i) Business Applications.

(ii) Home Applications.

(iii) Mobile users.

(iv) Social Issues.

(c) **Business Application**

(i) Companies generally have different groups of computers. Each group handles one of the following operations :

- Monitor production.
- Keep track of inventories.
- Manage payroll.

All these groups of computers share information via a network.

(ii) *Resource Sharing*

- Resource sharing deals with sharing *Physical Resources and Information.*
- **Physical Resource Sharing**

  A simple explanation of this type of sharing is that a single printer can be shared by a number of users over a network.
- **Information Sharing**

  Now-a-days companies store all their information such as customer information, inventory, accounts etc. online. This makes it possible for users to access information stored on computer which are located far away, thus eliminating the so called "*tyranny of geography*".

(iii) *The Client-Server Model*

- The model consists of powerful computers called servers (which store the data) and relatively less powerful computers called *clients* (which request for remote data stored on the servers.)
- The clients are connected to the server via a network as shown in the *figure 1.18.*

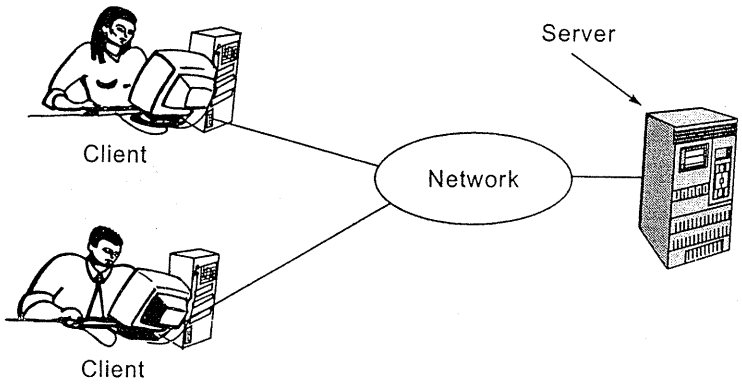


Fig. 1.18 : A network with two clients and one server.

- *Working*

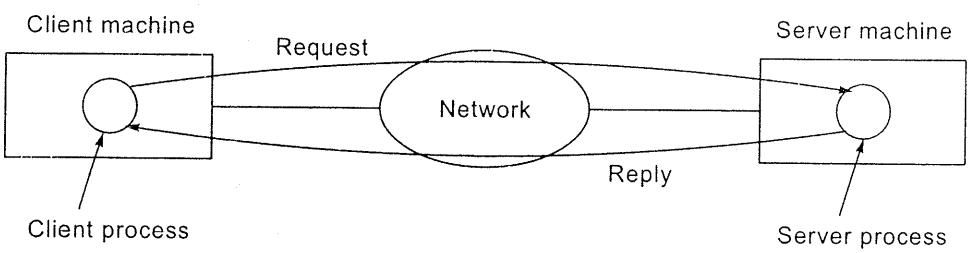


Fig. 1.19 : The client-server model involves requests and replies.

- ♦ Client requests for data.
- ♦ Server processes request.
- ♦ Server sends requested data i.e. reply.

(iv) Network facilitates easy communication. e.g. e-mail.

(v) Network facilitates fast communication. e.g. when an online document is modified the changes are visible immediately.

(vi) Video conferencing using networks eliminates the need to physically travel to different destinations for meetings.

(vii) *E-Commerce (Electronic Commerce) :* Customers can buy products via the internet thus eliminating the need to physically go shopping.

(d) Home Applications

(i) Computers are used by home users for word processing, games but mainly for the Internet. The following points show the applications of internet.

(a) **Access to Remote Information :** Generally we access the Internet to search for information. Here the client-server model is used, where clients search for information and servers maintain databases where information is stored.

(b) **Person to Person Communication** : This includes e-mail, instant messaging, news groups and peer to peer communication.

Peer to Peer Communication : Everyone familiar with Napster knows what this is. In this system there is no fixed client/server. A person can communicate and share data directly with others. Napster was shutdown as it was considered to be infringing on copyright laws.

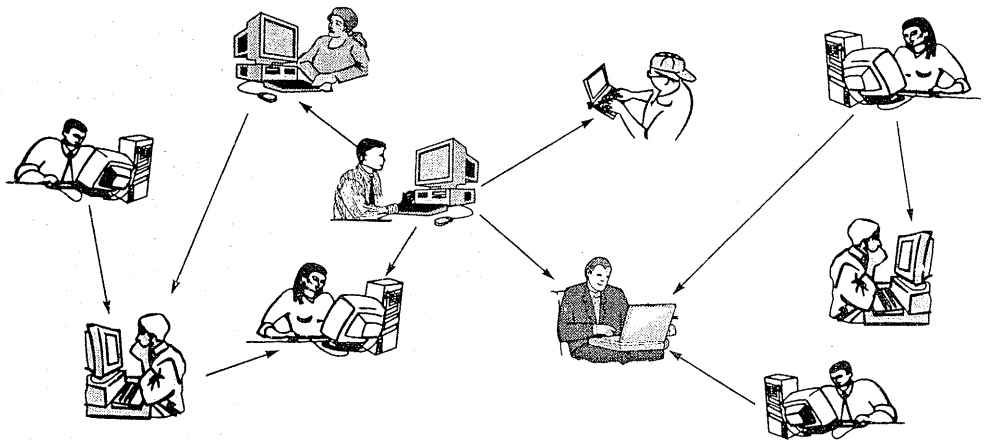


Fig. 1.20 : In peer-to-peer system there are no fixed clients and servers.

(c) **Interactive Entertainment :** It includes video-on-demand and gaming.

(d) **E-Commerce (Electronic Commerce) :** The various forms of e-commerce and their example applications are shown in the table.

Tag	Full Name	Example
B2C	Business-to-Consumer	Ordering book on-line
B2B	Business-to-Business	Car manufacturer ordering tires from supplier
G2C	Government-to-Consumer	Government distributing tax forms electronically
C2C	Consumer-to-Consumer	Auctioning second-hand products on line
P2P	Peer-to-Peer	File sharing

Fig. 1.21 : Some forms of e-commerce.

(e) **Mobile Users**

(i) Let us first understand the difference between *mobile* and *wireless*. Both these terms don't exactly mean the same thing. *Mobile* means that the computer can work even if it is moved geographically. *Wireless* means that a group of computers can be connected without using wires.

Wireless	Mobile	Applications
No	No	Desktop computers
No	Yes	Notebook computer
Yes	No	Desktops connected using wireless routers
Yes	Yes	PDA

Fig. 1.22 : Some forms of e-commerce.

- Table shows that a desktop computer is generally wired to a LAN and is also stationary (unless someone wants to carry the entire desktop computer along with him wherever he goes !).
- A notebook computer can be used anywhere (mobile) but to access the internet it should be connected to a telephone line (wired).
- Now-a-days in labs of our colleges the desktop computers are connected to each other using wireless LAN.
- PDA's are wireless and mobile.

(f) **Social Issues**

The Internet is no doubt a boon to mankind but as always there are some disadvantages with the Internet also. These are

(i) Attacks on countries.

(ii) Attacks on religions.

(iii) Pornography.

(iv) People can send anonymous messages to insult others.

(v) Spam.

(vi) Viruses.

(vii) Identify theft whereby a thief poses as another individual and buys goods or conducts transactions on his name.

(viii) Copyright violation (E.g. Napster)

Network security helps to avoid all these problems we study network security in further chapters.

1.20 Exam Question

May 06 [Q. 3(a)] *You are surfing the net from your terminal. Identify software and hardware components and relate them with ISO-OSI layered model. (Write about section 1.17 and 1.18 in short)* **(10 M)**

(a) **Physical Layer : Modem** : Short for modulator-demodulator. A modem is a device or program that enables a computer to transmit data over telephone or cable lines. Computer information is stored digitally, whereas information transmitted over telephone lines is transmitted in the form of analog waves. A modem converts between these two forms.

(b) **Datalink Layer : PPP** *(Discussed in Chapter on DLL, Section 3.6)*

(c) **Network Layer : IP Address** *(Discussed in Chapter on Network Layer)*

(d) **Transport Layer : TCP** *(Discussed in Chapter on Transport Layer)*

(e) **Session Layer : Session :** A session is the amount of time a user uses a website.

(f) **Presentation Layer : SSL (Secure Socket Layer)** : It is a protocol developed by Netscape for safely transmitting private documents(like credit card number) via the Internet.

(g) **Application Layer : Browser** : A browser is a software application used to locate and display web pages.

Extra Stuff : Not relevant from exam point of view.

[I]

For those of you who still have a few doubts about the working of the 7 Layers of the ISO - OSI model, here is a simple real life example to help you imagine. It shows the basic working of a how company managers send messages to each other.

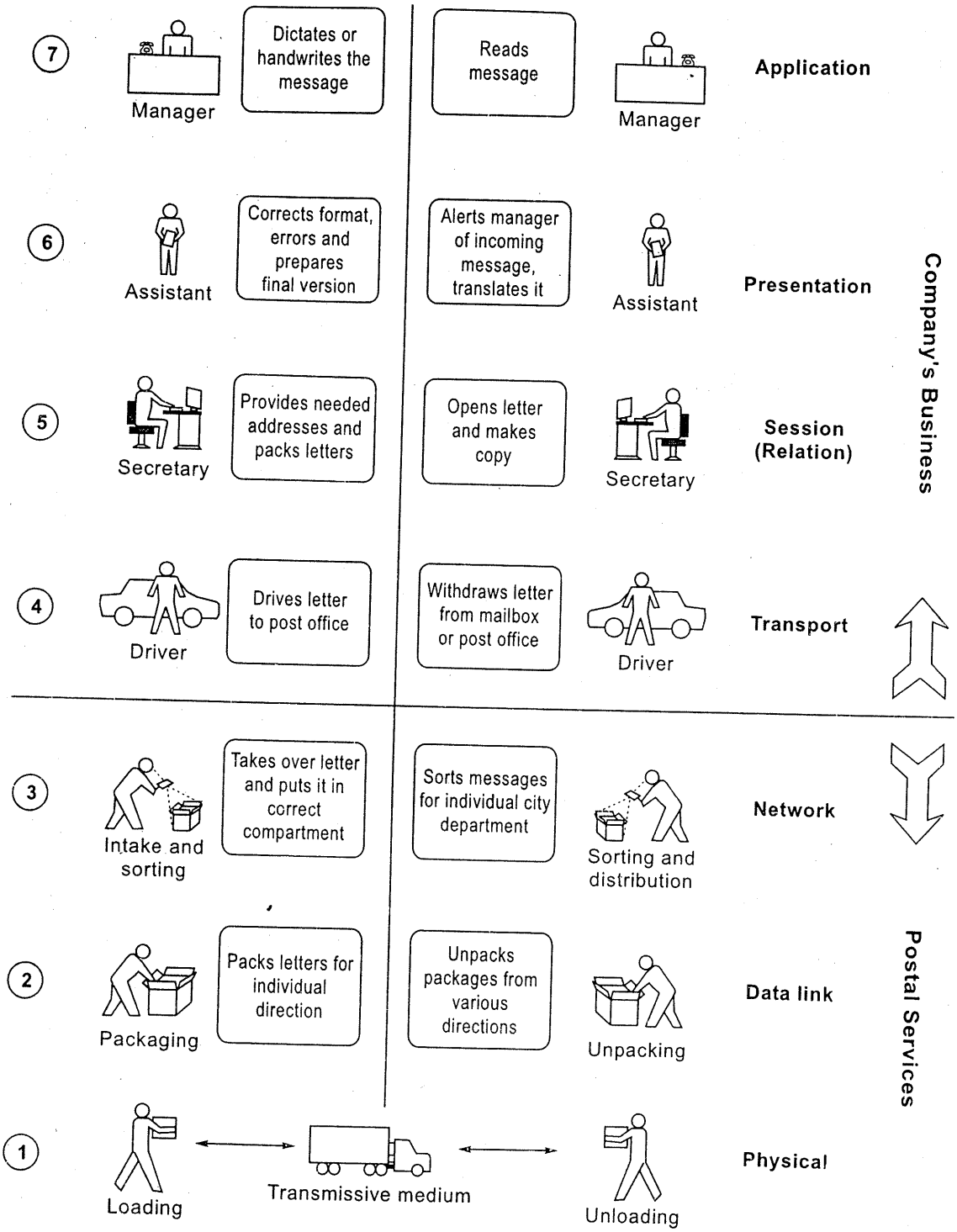


Fig. 1.23 : 7 Layer of ISO-OSI.

[II] Unified View of Layer, Protocols and Services

- **What is a unified view ?**
  - As seen before, network architecture is made up of layers.
  - Similar requirements of addressing, multiplexing and error and flow control exist at all the layers.
  - A unified view helps us understand the similarity between the protocols at the different layers.

- **Peer Proceses :**
  - Processes which are at the same layer on different machines.
  - Peer processes communicate accross a *peer interface.*

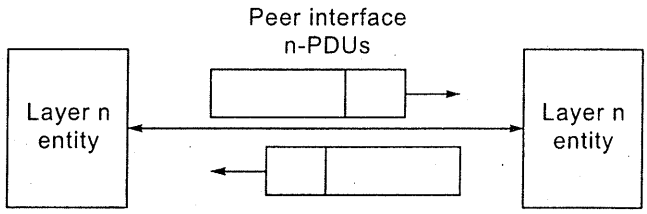


Fig. 1.24 : Peer-to-peer communication.

- Processes at the n^{th} layer are also known as *layer n entities.*
- *Layer n entities* communicate by exchanging **PDUs** (Protocol Data Units).

- **Working :**

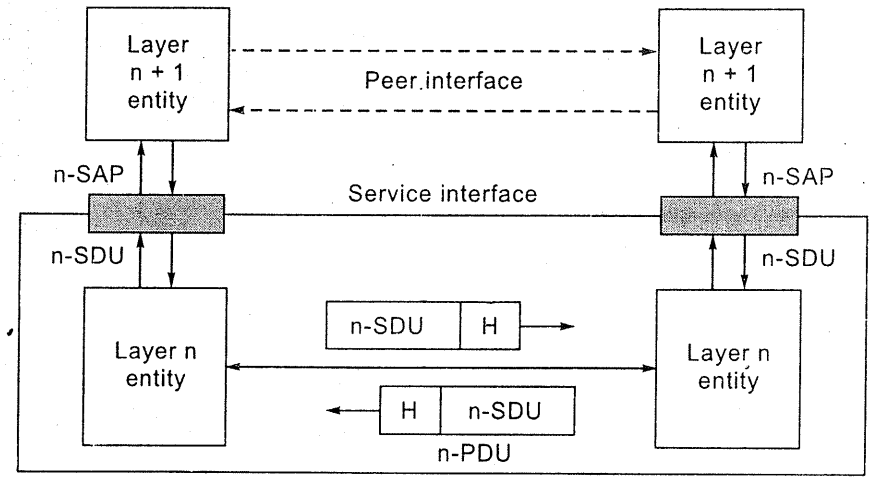


Fig. 1.25 : Layer services : SDUs are exchanged between layers while PDUs are exchanged within a layer.

- Suppose a layer n + 1 entity wants to send a PDU to its peer process.
- **Step 1 :** The **n + 1 layer** process passes a block of information to the **n layer** process below it. This block of information consists of *control information* and *a layer n-SDU* (Service Data Unit). A layer n-SDU is same as a layer n + 1-PDU. This information is passed via the *layer n-SAP* (Service Access Point) across a *service interface.*

- **Step 2** : The layer n entity uses the control information passed to it to create a header. This header is attached to the SDU. Header + SDU gives the layer n-PDU.
- **Step 3** : This layer n-PDU is passed to the layer n *peer process*.
- **Step 4** : The layer n peer process uses the header within the PDU to execute the layer n protocol.
- **Step 5** : The layer n-SDU i.e. the layer n + 1-PDU is passed to the layer n peer process.
- This is how PDUs are transmitted between peer processes and also shows how lower layers service the layers above themselves.

- The service provided by a layer can be *connection-oriented* or *connection-less*.
- **Connection-oriented Service** : It has three phases
  - *Establishment* : Establish a connection between two layer n-SAPs.
  - *Transfer* : Transfer n-SDUs using the layer n protocol.
  - *Release* : Break the connection and release the resources.
- **Connection-less Service** :
  - It does not require a connection to be set up.
  - A SDU is transmitted directly through the SAP.
  - The control information passed from layer n + 1 to layer n must contain all the address information required to transfer the SDU.
- **Confirmed Services** : Sender gets acknowledgment when receiver receives data.
- **Unconfirmed Services** : No acknowledgment is used.
- Connection-Oriented Service is always confirmed service whereas connection-less service can be either confirmed or unconfirmed.
- **Segmentation and Reassembly** :
  - Some systems have a maximum limit on the number of bytes that can be sent as a unit.
  - E.g. : In the *figure 1.26* the n-SDU is too large and therefore *segmented* at the sender into n-PDUs. These PDUs are *reassembled* at the receiver.

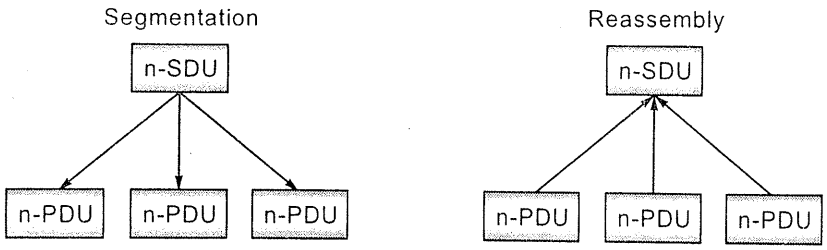


Fig. 1.26 : Segmentation and reassembly.

- **Blocking and Unblocking :**
  - If the layer n-SDUs are too small and result in inefficiency; then we use *blocking* which blocks several layer n-SDUs into a single layer n-PDU.
  - The receiver uses *unblocking* to get the individual SDUs from the PDU.

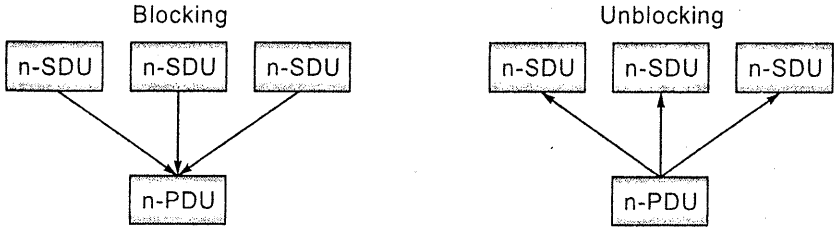


Fig. 1.27 : Blocking and unblocking.

- **Multiplexing and Demultiplexing :**
  - *Multiplexing* : Many n + 1 layers share the service provided by the same n layer.
  - *Demultiplexing* : The layer n at the receiver must deliver received SDUs to the appropriate n + 1 layers. This is done by using a *multiplexing tag* to determine which SDU belongs to which user.

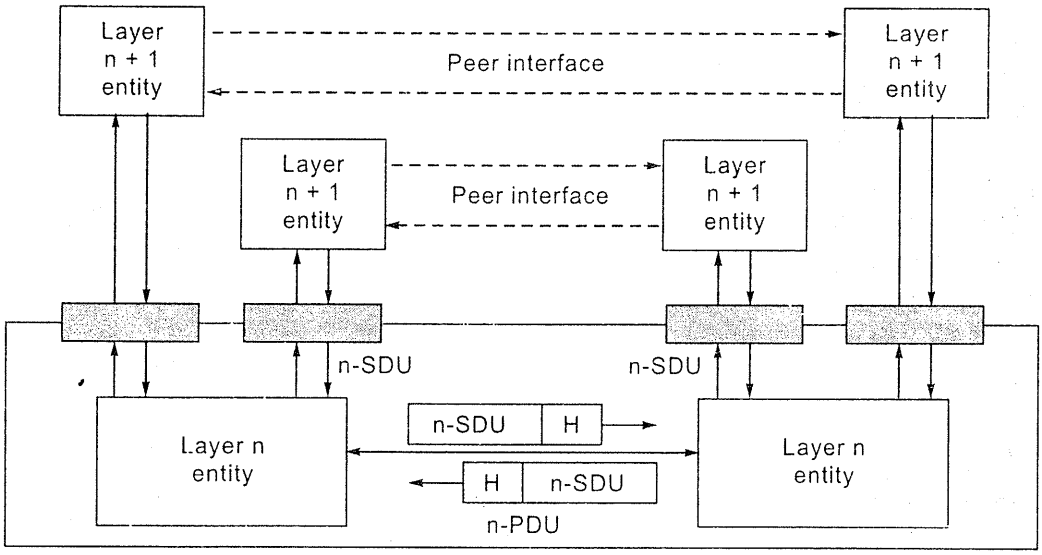
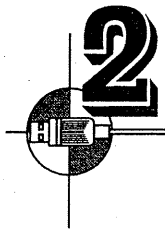


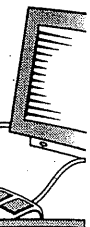
Fig. 1.28 : Multiplexing involves sharing of layer n service by multiple layer n + 1 users.

- **Splittinging and Recombing :**
  - *Splitting* : A single n + 1 layer uses services of multiple n layers at the sender.
  - *Recombing* : At the receiver the SDUs obtained from each of the n layers is combined to give to the n + 1 layer.

The essence of the entire discussion above is that similar needs occur at different layers and these can be met by a common set of services defined above.



PHYSICAL LAYER



This chapter covers Layer 1 of the ISO-OSI model. Sections covering guided and wireless media, satellites, telephone and mobile networks along with cable television are present.

Marks	
Dec. 03	10 M
May 04	20 M
Dec. 04	8 M
May 05	37 M
Dec. 05	20 M
May 06	10 M
Dec. 06	35 M
May 07	30 M

2.1 Guided Transmission Media

May 06 [Q. 3(b)] *State the different physical media properties. Also write about twisted pair cables.* **(10 M)**

Dec. 06 [Q. 3(a)] *Compare the performance characteristics of coaxial, twisted pair and fiber optic transmission media.* **(10 M)**

May 07 [Q. 3(a)] *State the different physical media properties. Explain any one in detail.* **(10 M)**

Physical Media Properties

(1) **Bandwidth** : A range within a band of frequencies or wavelengths. It signifies the amount of *data* that can be transmitted in a fixed amount of time. For *digital devices*, the bandwidth is usually expressed in bits per second (*bps*) or *bytes* per second. For *analog* devices, the bandwidth is expressed in cycles per second, or Hertz (Hz).

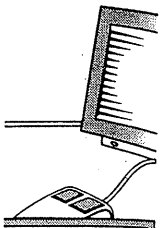
(2) **Interference** : Interference is the *superposition* of two or more signals that can distort the original signals.

(3) **Data Transfer Rate** : The speed with which *data* can be transmitted from one *device* to another. Data rates are often measured in *megabits* or *megabytes* per second. These are usually abbreviated as Mbps and MBps, respectively. In general, the greater the *bandwidth* of a given path, the higher the data transfer rate.

The various types of media are :

(1) **Magnetic Media**

- Data is written on magnetic tape or floppy disk.
- Bandwidth is excellent.
- Delay characteristics are poor.



larks
3 : 10 M
l : 20 M
4 : 8 M
; : 37 M
5 : 20 M
; : 10 M
6 : 35 M
' : 30 M

)out twisted
(10 M)
ed pair and
(10 M)
)e in detail.
(10 M)

It signifies
. For *digital*
)r *bytes* per
second, or

ls that can

l from one
gabytes per
:ctively. In
ıta transfer

(2) Twisted Pair

- It consists of two insulated copper wires twisted helically along each other. The copper wires are generally 1 mm thick.
- **Purpose of Twisting**
  - **(i)** When wires are twisted the waves from the different twists cancel out, as a result the wire radiates less.
  - **(ii)** Twisting reduces the interference between adjacent pairs of cables.
- It can be used for either analog or digital transmission.
- Bandwidth depends on the thickness of the wire and the distance traveled.
- **Application :** Telephone System.
- **Types :**

(i) *Unshielded Twisted Pair*

- It is the most common type of medium used.
- It is suitable for data and voice communication.
- The pair of insulated copper wires is not covered by an external metal casing.
- The important types of UTP are :
  - (1) Category 3
    - Four twisted pairs are grouped.
    - Was used in office telephone systems before 1988
  - (2) Category 5
    - Similar to category 3 but with more number of twists per centimeter to reduce the cross-talk. Cross Talk is the electromagnetic signal given from one channel that acts as noise for another channel.
  - (3) Category 6 and 7
    - Being developed to have bandwidth of 250 MHz and 600 MHz respectively.

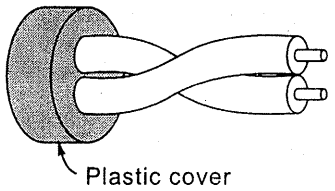


Fig. 2.1 : Unshilded twisted-pair (UTP).

(ii) *Shielded Twisted Pair*

- The pair of insulated copper wires is covered by an external metal foil or braided mesh covering.

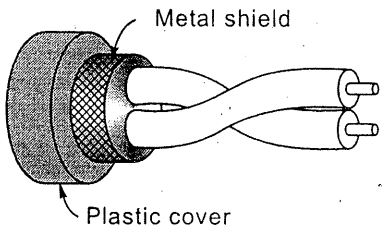


Fig. 2.2 : Shilded twisted-pair (STP).

- This casing provides protection from external electromagnetic noise.
- It eliminates *cross-talk*, which is the electromagnetic signal given from one channel that acts as noise for another channel.

(3) Co-axial Cable

- It consists of a stiff copper wire as the core surrounded by an insulating material. This insulating material is further enclosed within a cylindrical conductor (generally a mesh).
- Co-axial cables have a higher bandwidth and a better noise immunity.
- Bandwidth close to 1 GHz.
- It is mainly used for cable television.
- Types
  - (i) *50-ohm* : Used for digital transmission.
  - (ii) *75-ohm* : Used for analog transmission.

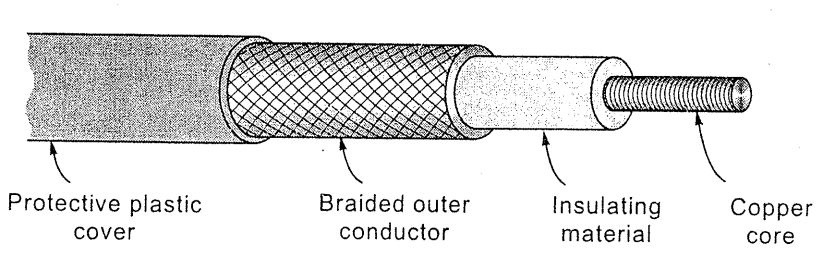


Fig. 2.3 : A coaxial cable.

4. Fiber Optic Cable

- *Optical* means light. *Fiber* is considered as a wire made of glass or plastic. Fiber Optics is sending light signals down hair-thin strands of glass or plastic fiber.
- Used for transmission over large distances without loss of signal.
- Types
  - (i) Multimode and Singlemode fiber are the two types of fiber in common use.
  - (ii) Both fibers are 125 microns in outside diameter (a micron is one-millionth of a meter.)
  - (iii) Multimode fiber has light traveling in the core in many rays, called *modes*. It has a bigger core (almost always 62.5 microns). Singlemode fiber has a much smaller core, only about 9 microns, so that the light travels in only one ray.

(iv) Bandwidth of Multimode Fiber = 2 GHz.

(v) Bandwidth of Single Mode Fiber is nearly infinity.

- **Working**

(a) *Refraction :* When a ray of light moves from one medium to another the ray is refracted at the boundary separating the two media.

(b) For angles of refraction greater than the *critical angle* the light is refracted back into the same medium and does not pass to the other medium. This is called *Total Internal Reflection.*

(Note : This is similar to the ray getting reflected at the boundary of the two media.)

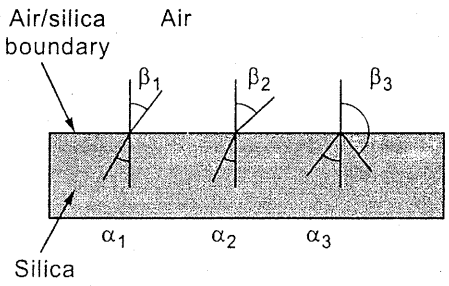


Fig. 2.4(a) : Three examples of a light ray from inside a silica fiber impinging on the air/silica boundary at different angles.

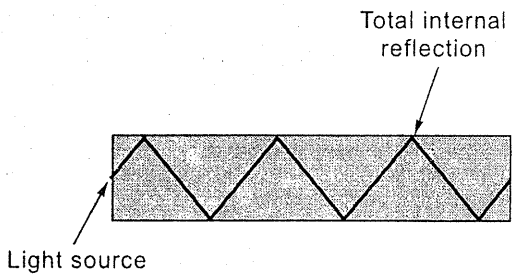


Fig. 2.4(b) : Light trapped by total internal reflection.

- **Construction**

(a) The glass "core" is at the centre. Light is passed through this core. It is very delicate.

(b) The core is surrounded by an optical material called the "cladding" that traps the light in the core using an optical technique called "total internal reflection."

(c) A thick plastic called the "jacket" covers the cladding.

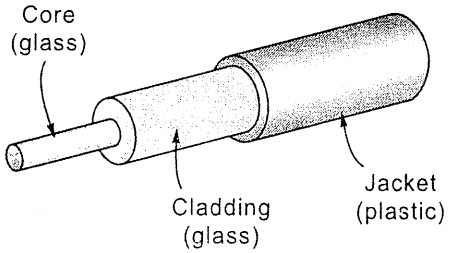


Fig. 2.5(a) : Side view of a single fiber.

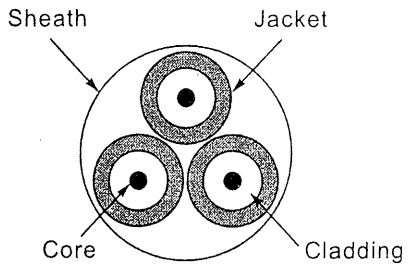


Fig. 2.5(b) : End view of a sheath with three fibers.

2.2 Wireless Transmission

Types :

(1) Radio transmission.

(2) Infrared transmission and millimeter waves.

(3) Microwave transmission.

(4) Lightwave transmission.

Sr. No.	Radio Transmission	Infrared and Millimeter Waves	Lightwave Transmission
(1) Generation	Easy to generate.	Easy to generate.	Easy to install.
(2) Travel Distances	It can travel large distances.	It can travel small distances.	It can travel large distances.
(3) Penetration	High : It can penetrate buildings.	It cannot penetrate solids.	Cannot penetrate rain and fog.
(4) Direction	It travels in all directions.	Relatively Directional.	Unidirectional.
(5) Application	Ham Radio and Military operations.	Remote controls for TV, VCR etc.	Connect the LANs in two buildings.

Microwave Transmission

(1) It uses transmitting and receiving antennas.

(2) Transferring signals from one antenna to another is called a hop.

(3) A signal may have to go through many hops to reach its destination.

(4) While determining microwave paths we have to keep in mind "The Line Of Sight".

(5) **Applications** : Distance telephone communication, cellular phones etc.

(6) **Penetration** : Can pass through thin solids. It has difficulty in passing through buildings.

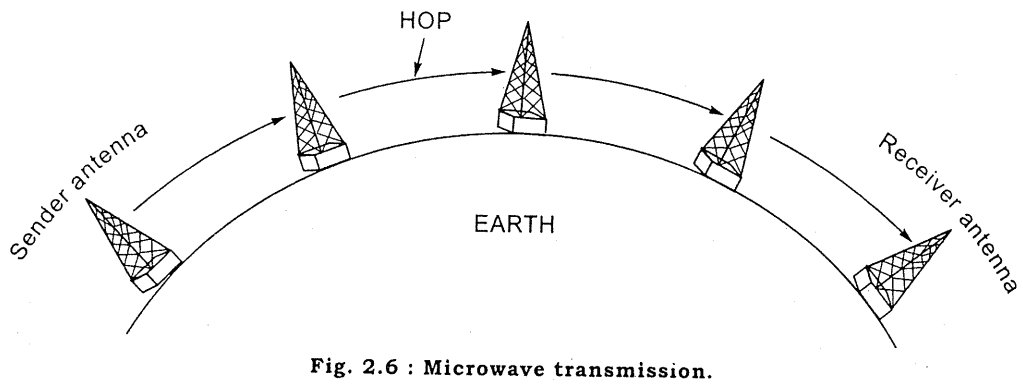


Fig. 2.6 : Microwave transmission.

2.3 Satellite Communication

A Satellite is a specialized wireless receiver / transmitter that orbit around the earth.

The process is very simple :

(1) The transmitter on the earth sends a signal to the satellite on a frequency called the UPLINK FREQUENCY.

(2) The satellite then sends the signal to the receiver on the ground on a frequency called the DOWNLINK FREQUENCY.

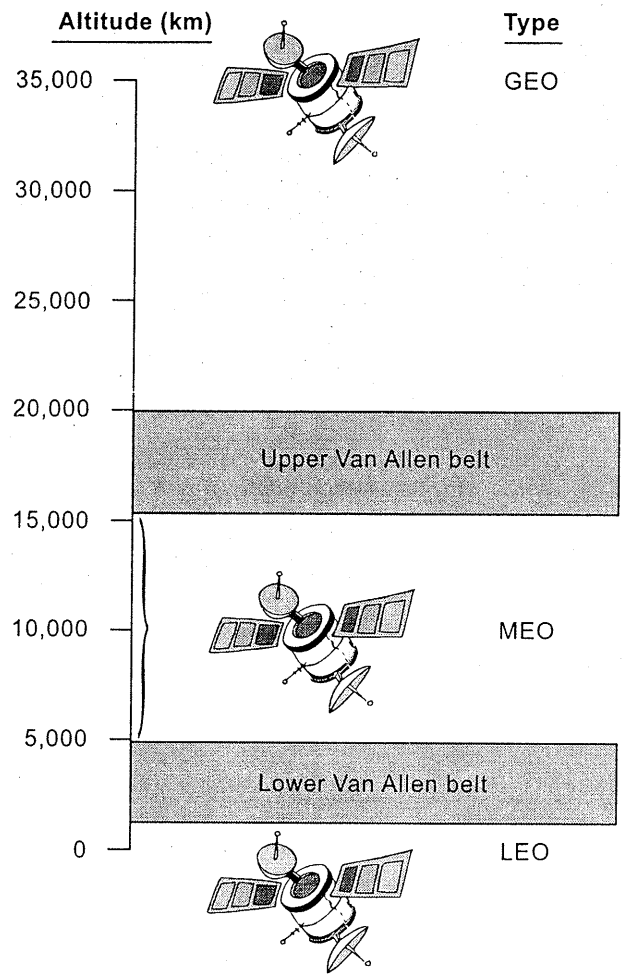


Fig. 2.7 : Satellite and their altitude.

[A] GEO (Geostationary)

(1) Present at 350,786 km from the earth.

(2) Satellite moves at the same speed as the earth, so as to ensure constant communication between the transmitter and the receiver present on earth.

(3) Requires high power transmitters as they are quite far from the earth.

(4) Present above UPPER VAN ALLEN BELT (A *Van Allen Belt* is a belt that contains charged particles)

(5) **Application :** Television Signal Broadcasting.

[B] MEO (Medium Earth Orbit)

(1) Present at 5000 - 15,000 km from the earth.

(2) Requires low power transmitters as they are not very far from the earth.

(3) Present between UPPER and LOWER VAN ALLEN BELT.

(4) **Application :** Land and sea navigation. The GPS (Global Positioning System) satellites are present in this belt.

[C] LEO (Low Earth Orbit)

(1) Present at 500-1,000 km from the earth.

(2) Requires very low power transmitters as they are very close to the earth.

(3) Present below LOWER VAN ALLEN BELT.

(4) **Application :** Mobile telephony.

2.4 Fiber Optic v/s Satellite

Sr. No.	Property	Fiber Optic	Satellite
(1)	Potential Bandwidth	Higher	Lower
(2)	Mobile Communication	Not possible	Possible
(3)	Broadcasting	Not practical	Helpful as messages can be sent to thousands of ground stations at once.
(4)	Setup Due to Terrain	Difficult to setup in places with hostile terrain.	No such problem.
(5)	Rapid Deployment	Infrastructure needs to be setup therefore cannot be rapidly developed.	Can be rapidly developed.
(6)	Security	Difficult to tap hence it provides high security.	Information can be received by unauthorized user therefore it provides low security.

2.5 The Public Switched Telephone Network

(1) The telephone network is organized as a highly redundant multi-level hierarchy.

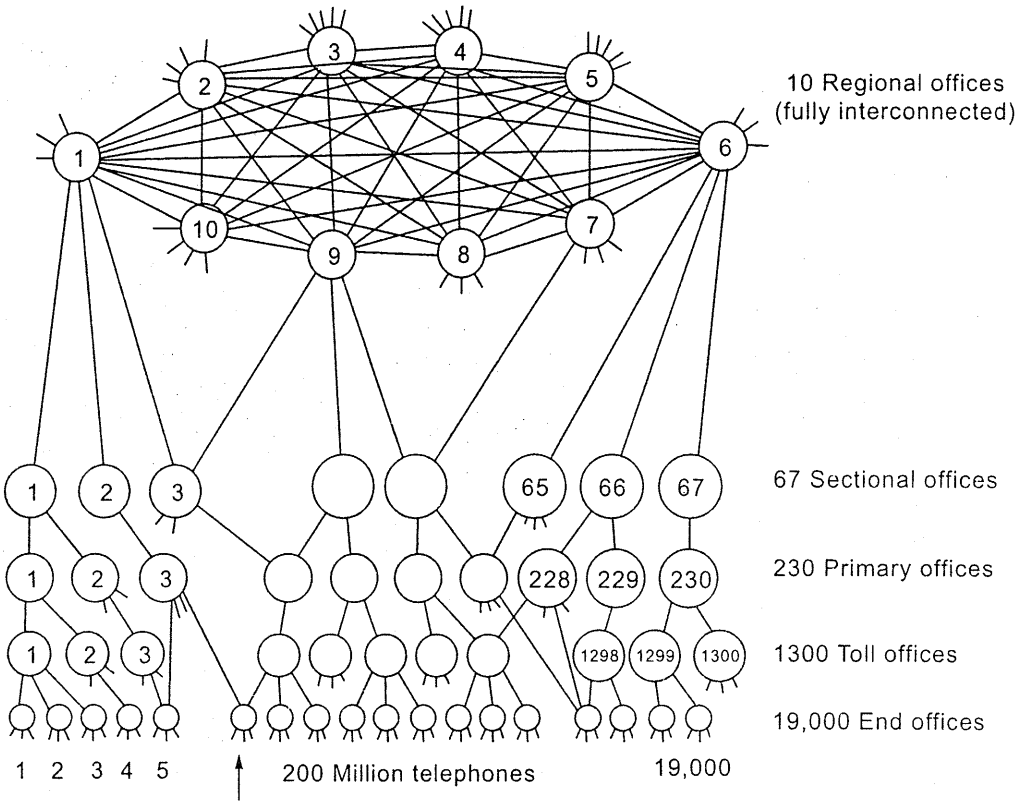


Fig. 2.8 : Telephone network.

(2) **Local Loop**

(a) Each telephone has 2 wires coming out of it that go directly to the telephone company's nearest end office. This two wire connection between each subscriber and the end office is called a *local loop*.

(b) The wires are Category 3 twisted pairs.

(3) **End Office**

(a) If a subscriber attached to a given end office calls another subscriber attached to the same end office; then the switching mechanism within the end office sets up a direct electrical link between the 2 local loops.

(b) If the called telephone is attached to another end office then some different procedure is used.

(4) **Toll Office**

(a) Each end office has a number of outgoing lines to one or more nearby switching centers called *toll offices*.

(b) If both caller's and callee's end offices are connected to the same toll office, the connection will be established within the toll office. Otherwise a path will have to be established somewhere higher up in the hierarchy.

(5) **Primary Office :** Several toll offices are connected to a primary office.

(6) **Sectional Office :** Several primary offices are connected to a sectional office.

(7) **Regional Office**

(a) Several sectional offices are connected to a regional office.

(b) The regional offices are connected using a mesh topology.

(8) The primary, regional and sectional offices form a network by which toll offices are connected. Such a network is called a *switching center*.

(9) The primary, regional and sectional exchanges communicate with each other through high bandwidth *Intertoll Trunks*.

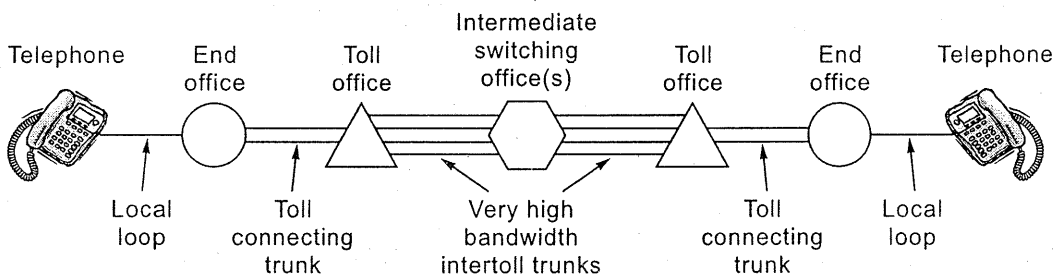


Fig. 2.9 : A typical circuit route for a medium distance call.

2.6 Switching

Dec. 03 [Q. 2(a)] *Explain the difference of circuit switching, packet switching and message switching using a diagram.* **(10 M)**

May 04 [Q. 2(b)] *What is the difference between circuit switching and packet switching ?* **(10 M)**

May 05 [Q. 4(a)] *Differentiate circuit switching, packet switching and message switching. Draw neat diagrams if any.* **(10 M)**

Dec. 05 [Q. 5(c)] *Explain the difference of circuit switching, packet switching and message switching using a diagram.* **(10 M)**

Dec. 06 [Q. 4(a)] *Explain circuit switching, packet switching and message switching with suitable timing diagrams.* **(10 M)**

May 07 [Q. 1(a)] *Explain circuit switching, packet switching and message switching with suitable example.* **(10 M)**

What is a Switch ?

Switches are devices capable of creating temporary connections between two or more devices linked to the switch but not to each other.

Types

(1) Cir

(2) Pac

(3) Mes

Sr. No.	
(1)	C c d d d n o te b ou
(2)	Pa 1. 2.
(3)	Al pa
(4)	Ba 1. 2.
(5)	Co tim
(6)	Swi

Types of Switching :

(1) Circuit Switching.

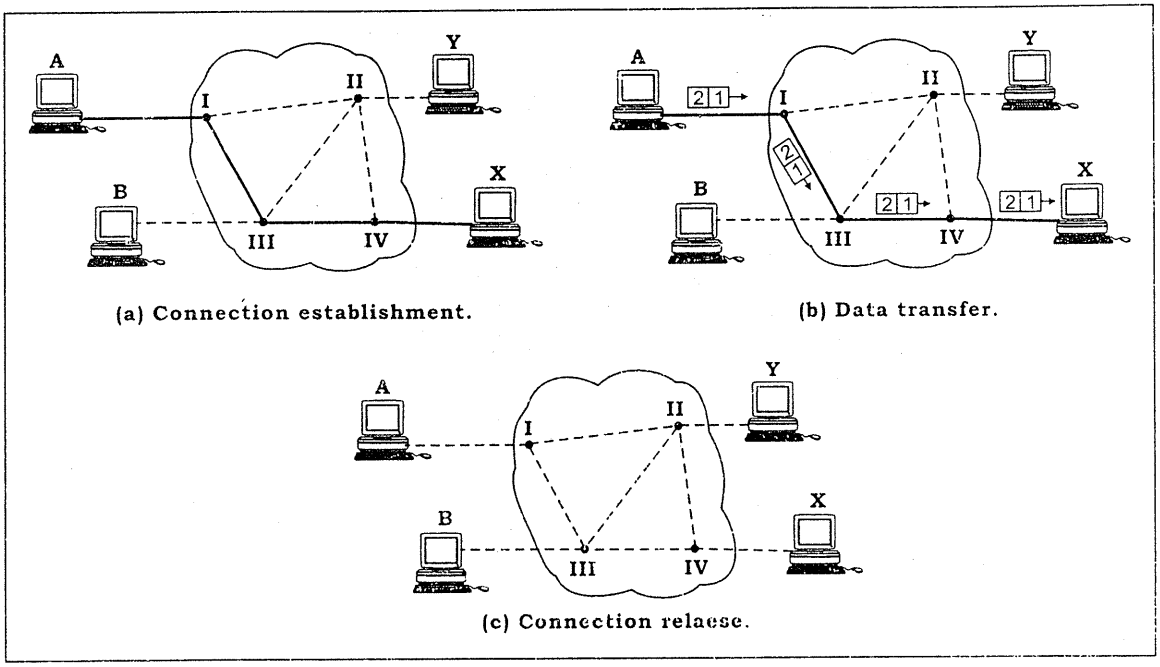
(2) Packet Switching.

(3) Message Switching.

Sr. No.	Circuit Switching	Packet Switching	Message Switching
(1)	Creates direct physical connection between 2 devices. A circuit switch is a device that has a number of devices as inputs and a number of devices as outputs. There can be a temporary connection between any input and output device.	In packet switching data is transferred in discrete units called packets.	In message switching the data stream in not divided into packets. Entire data stream is sent as a message from one node to another.
(2)	**Path** 1. Physical connection between transmitter and receiver exists. 2. Needs an end to end path before transmission begins.	1. Physical path between transmitter and receiver does not exist. 2. Does not need end to end path before transmission.	1. Physical path between transmitter and receiver does not exist. 2. Does not need end to end path before transmission.
(3)	All packets follow same path and arrive in order.	Packets travel independently and may not arrive in order.	The Message is not divided into packets. The store and forward policy is used.
(4)	**Bandwidth** 1. Must be reserved in advance. 2. Therefore bandwidth may be wasted.	1. Need not be reserved in advance. 2. Therefore cannot be wasted.	1. Need not be reserved in advance. 2. Therefore cannot be wasted.
(5)	Congestion occurs at setup time.	Congestion occurs on every packet.	Congestion does not occur.
(6)	Switch crash IS fatal.	Switch crash is NOT fatal.	Switch crash IS fatal.

(7)	**Application** Telephone networks.	Internet.	Telegraph networks.
(8)	**Information Type** Analog voice or PCM digital voice.	Binary.	Morse, Baudot, ASCII.
(9)	**Multiplexing** Circuit Multiplexing.	Packet Multiplexing.	Message Multiplexing.

Diagrams

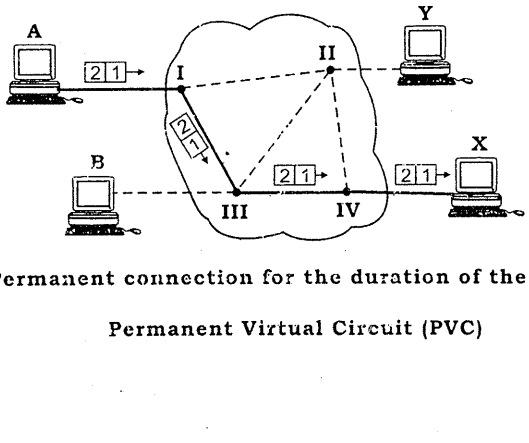


(a) Connection establishment.

(b) Data transfer.

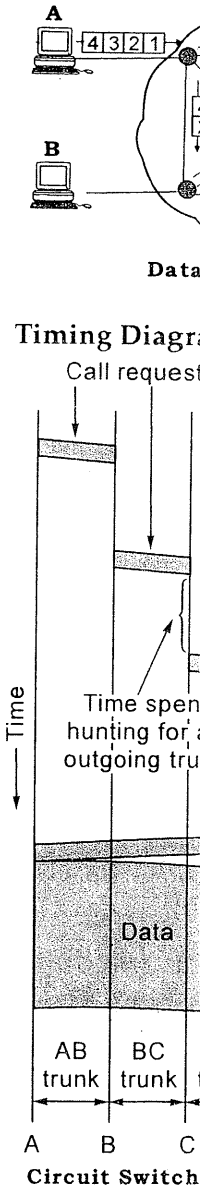
(c) Connection relaese.

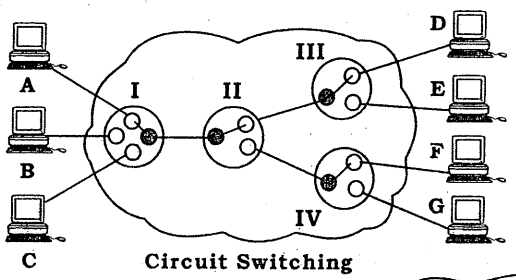
Switched Virtual Circuit (SVC)



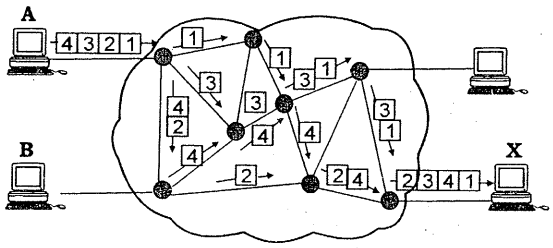
Permanent connection for the duration of the lease.

Permanent Virtual Circuit (PVC)

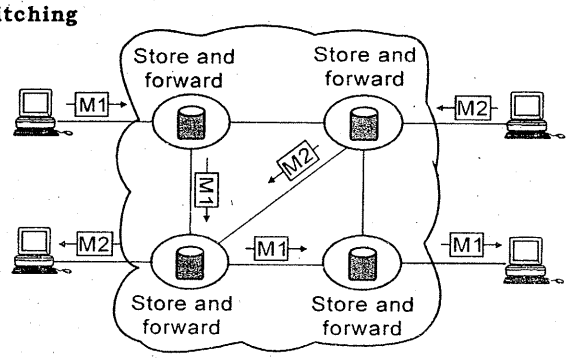




Circuit Switching



Datagram Packet Sitching



Message Switching

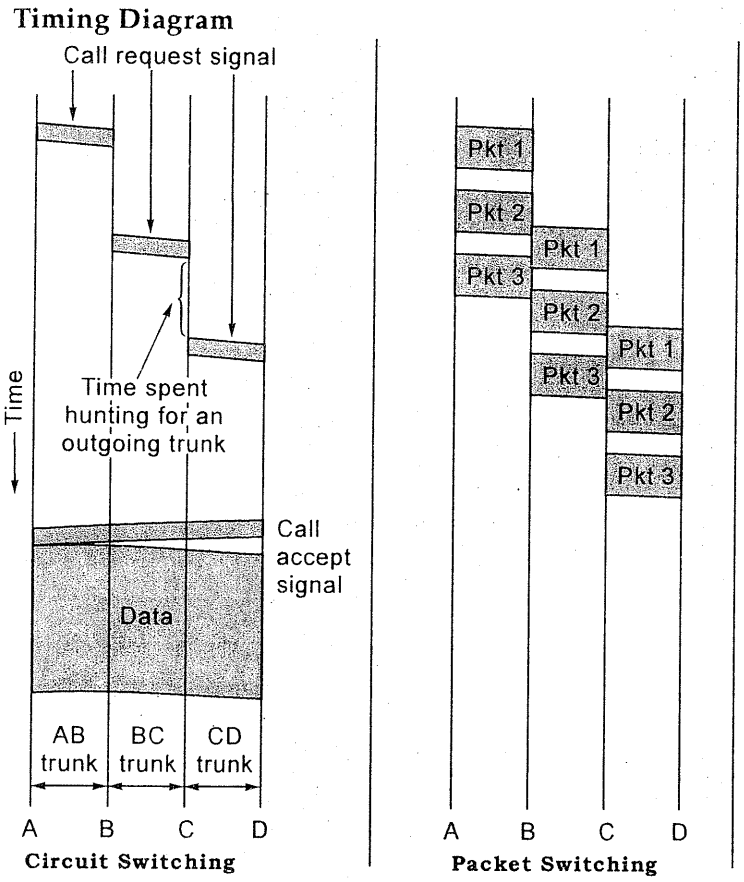


Fig.2.10 : Types of switching.

2.6.1 Packet Switching

(a) In *packet switched network* data are transmitted in discrete units called *packets*.

(b) In a packet along with *data* there also exists a *header* which contains the sender and recipient address.

(c) The two methods of transferring packets are :

(i) Virtual Circuit or Connection Oriented Service (CONS).

(ii) Datagram or Connectionless Service (CLNS).

These concepts have been covered in section 1.9.

2.7 The Mobile Telephone System

(1) An area is divided into cells.

(2) Each cell is controlled by a Base Station (Cell Office).

(3) Cell Offices are controlled by a Mobile Telephone Switching Office (MTSO). The MTSO is used to co-ordinate the communication between the Base Stations and the Telephone Central Offices.

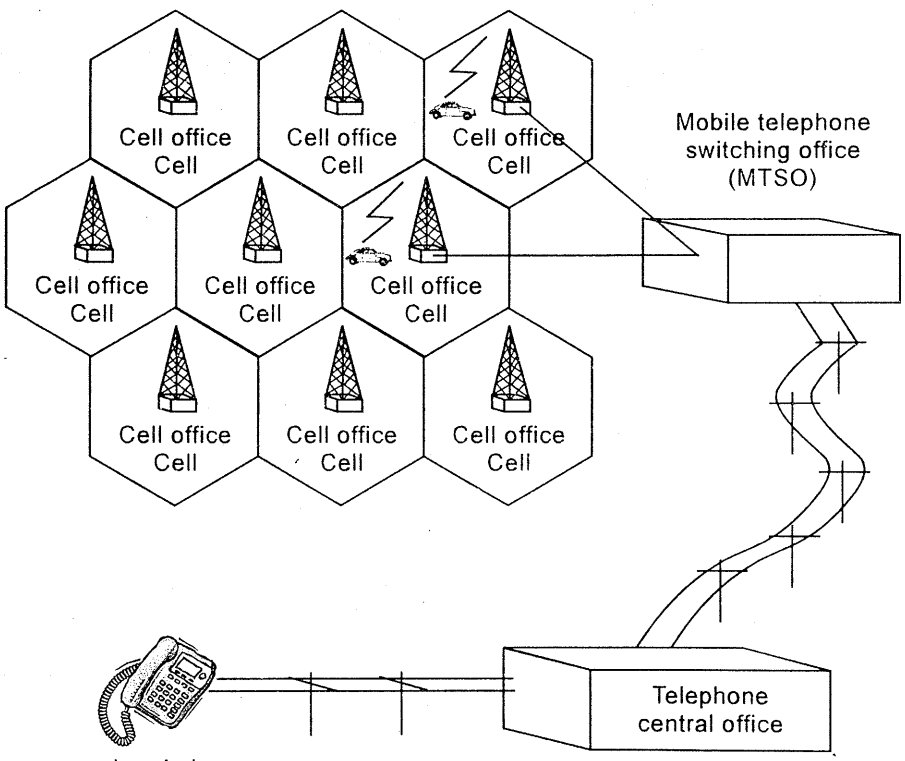


Fig. 2.11 : Mobile telephone system.

(4) Handoff

- When a mobile phone leaves a cell, the control of the cell on the mobile phone becomes less.
- The Base Station sends a message to the MTSO telling it that the mobile phone has left its area.
- Now the MTSO asks all the neighbouring cells to check if the mobile phone is present in their area.
- The cell in which the mobile phone is present sends a message to the MTSO. Now, the ownership for the mobile phone is given to this cell. This is called *Handoff.*
- Types of Handoff

(a) *Hard Handoff*

(i) Mobile Phone can communicate with only one Base Station at a time.

(ii) When mobile phone moves from one cell to another cell the communication with the old cell has to be broken before a new communication is setup with the new cell. This is called *rough transition.*

(b) *Soft Handoff*

(i) Mobile Phone can communicate with two Base Stations at a time.

(ii) When mobile phone moves from one cell to another cell the communication with the old cell need not be broken before setting up new communication with the new channel. This is called *smooth transition.*

(5) *Forward Setup Channel :* Carries messages from the Base Station to a mobile phone present within its cell.

(6) *Reverse Setup Channel :* Carries messages from the mobile phone to the Base station which controls the cell in which the mobile phone is present.

(7) Working

(a) *Call from Land Line to Mobile Phone*

- The Telephone Central Office sends number to the MTSO.
- MTSO asks all the Base Stations to check if the mobile phone is present in their area. This process is called *Paging.*
- The Base Stations broadcast this request to all the mobile phones in its area through the *forward setup channel.*
- The desired mobile receives the request and sends a message to the base station on the *reverse setup channel.*

- The corresponding Base Station tells the MTSO that the mobile phone is present in its area.
- MTSO sets up a forward and reverse voiced channel to the call.

(b) *Call from Mobile Phone to Land Line*

- Mobile Phone sends a request via the *reverse setup channel* to its base station.
- Base station sends the land line number to be contacted to the MTSO.
- The MTSO sends this number to the Telephone Central Office.
- The Telephone Central Office tracks the land line and allocates forward and reverse voice channel to the call.

2.8 Cable Television

Cable television is a mature industry. Cable operators are beginning to provide services like Internet access to the home at very affordable rates. This section briefly discusses the CATV infrastructure, how it is being upgraded to accommodate interactive communication services.

In the late 1940s and early 1950s the television industry provided *broadcast signals* only to the most populous areas. Some regions had poor TV reception either because of obstructions or long distances from signal transmitters. Cable television provided a workable solution.

The early cable providers constructed large antennas on hilltops or buildings for improved TV reception, and then strung coaxial cable from the antenna to the local community. Out of this environment the acronym CATV, representing Community Antenna Television was born.

With the advent of satellite broadcasts to cable systems in the 1970s, cable operators were able to provide more channels than were available over the traditional airwaves.

System Architecture

(1) **Headend** : The headend is the center of CATV activity. It is here where external signals such as satellite, microwave, and local TV station broadcasts are received.

(2) **Trunks** : A number of trunks, carry the signals from the headend to a series of distribution points.

(3) **Feeder Cables** : The smaller distribution or feeder cables branch out from the trunks and are responsible for serving local neighborhoods.

(4) **Drop Cables** : Feeder cables are tapped at periodic locations to furnish the coaxial drop cables that enter directly into the customer's premises.

(5) **Terminal Equipment** : The Terminal Equipment takes in input from the drop cable and drives the devices.

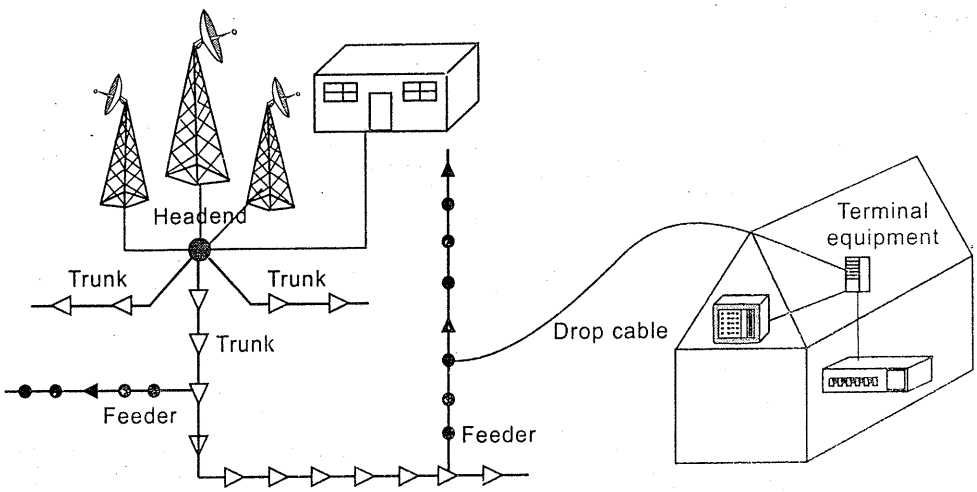


Fig. 2.12 : Traditional cable television architecture.

Compared to the telephone industry, cable television systems do have a truly high-bandwidth delivery system to the home.

Recent Cable System Developments : In the late 1980's the coaxial cable was replaced with fiber optics. Since signals transmitted by optical fiber can be carried for significantly longer distances, fewer amplifiers are needed. This results in fewer points of failure, lower maintenance costs, and better signal quality.

A current state-of-the-art cable architecture is a hybrid fiber/coax (HFC) combination.

Cablevision's Access Plaza

Access Plaza is a combination of hardware and software that lets users surf the Internet, get interactive news, and do home banking all through the coaxial drop cable that enters their homes.

Cable Modems : The key piece of terminal equipment that enables such access is the cable modem. *Figure 2.13* shows how a generic cable modem might be incorporated into an existing home drop cable, and connected to a general-purpose computer.

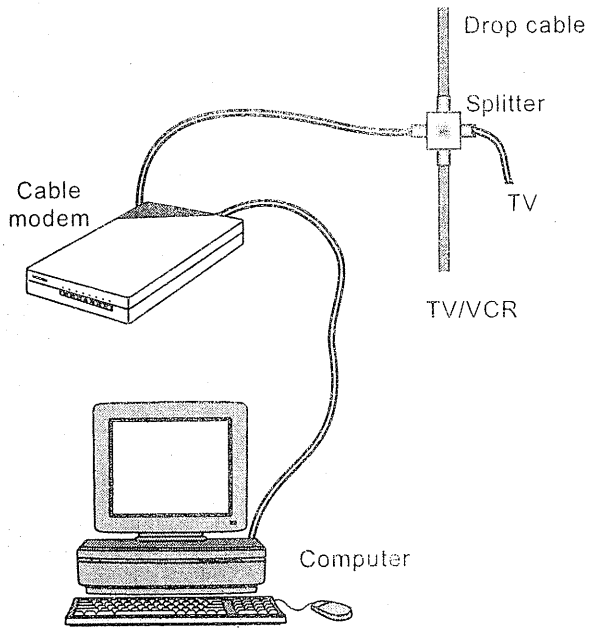


Fig. 2.13 : Generic cable modem connection.

2.9 Internetworking Devices

May 04 [Q. 8(a)] *What is internetworking ? Describe the devices used in the internetworking.* **(10 M)**

Dec. 04 [Q. 1(a)] *Give the functionality of each of the following inter-networking devices and mention which layer each one operates in : Repeater, Hubs, Bridges, Switches, Routers.* **(8 M)**

May 05 [Q. 2(a)] *Write detail note on internetworking covering devices, issues, problems and remedies.* **(10 M)**

May 05 [Q. 6(b)] *Give the functionality of each of the following inter-networking devices and mention which layer each one operates in : Repeater, Hubs, Bridges, Switches, Routers.* **(10 M)**

Dec. 05 [Q. 1(a)] *Give the functionality of each of the following inter-networking devices and mention which layer each one operates in : Repeater, Hubs, Bridges, Switches, Routers, Gateway.* **(10 M)**

Dec. 06 [Q. 1(a)] *What is hub ? Explain various types of hub ? How hub is different from switch ?* **(5 M)**

May 07 [Q. 4(a)] *Give the functions of repeater, hub, bridge, switch, router, gateway.* **(10 M)**

When two networks want to communicate with each other an internetworking device must be placed at the junction of the two networks. The devices used for this purpose are :

(1) Physical Layer : Repeater

- Signals get attenuated as they travel along the line.
- Process of the repeater :
  - (i) Receive the signal.
  - (ii) Regenerate the signal.
  - (iii) Transmit the regenerated signal.

Fig. 2.14 : Repeater.

(2) Data Link Layer : Bridges

- Process of a bridge :
  - (i) Regenerate the signal.
  - (ii) Read the address present on the signal.

(iii) Find the Port Number by using the address in the Bridge Table.

(Note : Bridge Table is a table which has the address of each station along with the port number on which the station is connected)

(iv) Transmit the regenerated signal on the port found in the previous step.

Address	Port
712B13456141	1
712B13456142	1
642B13456112	2
642B13456113	2

Bridge table

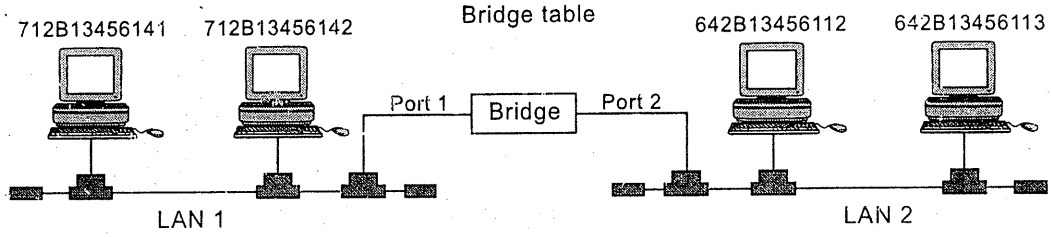


Fig. 2.15 : Bridge.

(3) Data Link Layer : Switches

- It is more efficient than bridging.
- Process of a switch :
  - (i) Regenerate the signal.
  - (ii) Buffer the incoming packet.
  - (iii) Check the address and decide the outgoing line.
  - (iv) Retransmit the packet only if the line is idle.

(4) Network Layer : Router

- They decide on the most efficient path that the packets should take while flowing from one network to another.
- A number of routing protocols can be used to route packets. These will be covered in the chapter on the network layer.

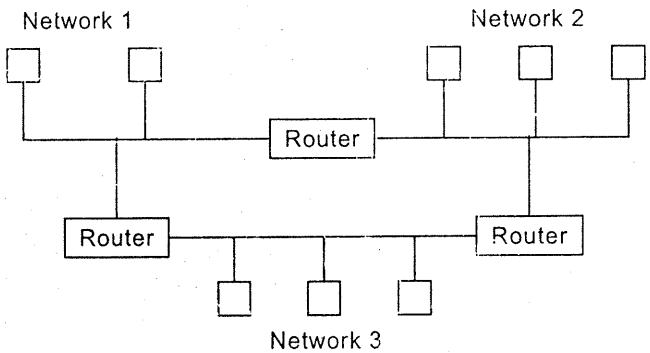


Fig. 2.16 : Router.

(5) All layers : Gateways

- A gateway is a protocol converter.
- It translates messages from one network's protocol to another network's protocol.
- The software to implement a gateway is generally loaded on the routers.

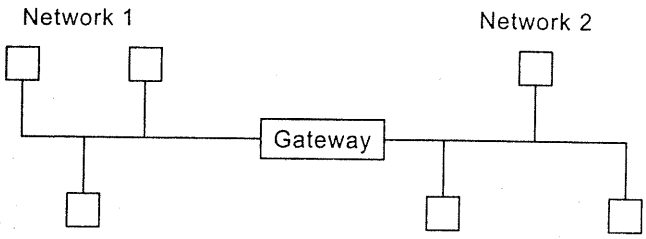


Fig. 2.17 : Gateway.

(6) Hubs

- **Simple Hubs** : Physical Layer, **Intelligent Hubs** : Data Link Layer.
  - (i) A network can be expanded by using hubs. Basically a hub is a device into which many devices can be connected at the same time.
  - (ii) Hubs also have the capacity to regenerate the incoming signal.
  - (iii) *Simple Hubs* : They may/may not regenerate the signal before transmission.
  - (iv) *Intelligent Hubs* : They regenerate the incoming signal. They perform "Intelligent Path Selection".

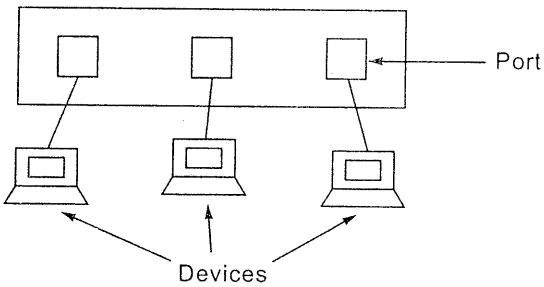


Fig. 2.18 : Hub.

Which device is in which layer ?

Application layer	Application gateway
Transport layer	Transport gateway
Network layer	Router
Data link layer	Bridge, switch
Physical layer	Repeater, hub

Fig. 2.19(a) : Which device is in which layer.

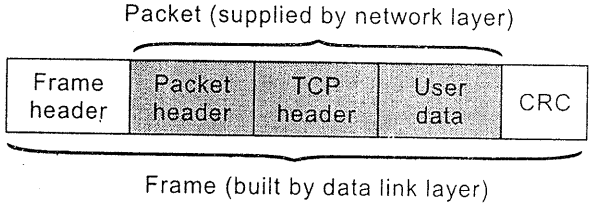


Fig. 2.19(b) : Frames, packets, and headers.

Issues and Problems Faced in Internetworking :

(1) Networks may have radically different technology.

(2) Each network may be using a different protocol.

(3) As new hardware devices are generated new software and protocols also have to be developed so as to interface it with the old devices.

(4) Problems like protocol conversion, ordering of packets and address conversions arise when we try to interface a connection oriented network with a connectionless network.

(5) The maximum number of packets supported by different networks is different.

(6) Quality of service of different networks may be different.

(7) Error control, flow control and congestion control among different networks is different.

Remedies

Internetworking devices that should be used :

(1) Use hubs / repeaters in the physical layer.

(2) Use bridges / switches in the datalink layer.

(3) Use routers in the network layer. A router that can handle networks with different protocols is called a *multiprotocol router*.

(4) Transport gateways are to be used in the transport layer. E.g. A transport gateway can allow packets to flow between a TCP network and an SNA network.

(5) Application gateways are to be used in the application layer.

The *figure 2.20* shows different networks interconnected to each other via a FDDI ring.

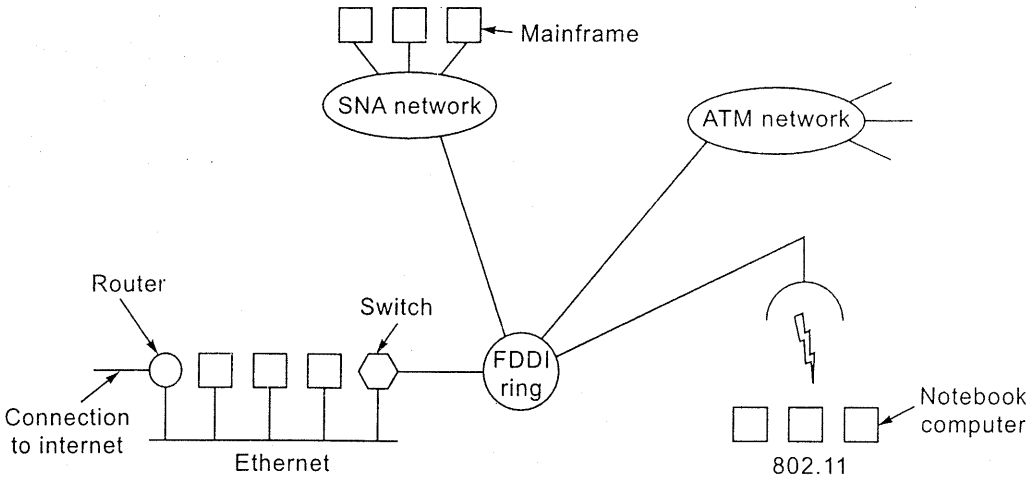


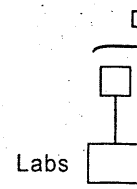
Fig. 2.20 : A collection of interconnected network.

2.10 LAN for An Institute

May 05 [Q. 7] *Design of a LAN network for your institute.* **(7 M)**

Dec. 05 [Q. 3(b)] *Suppose you want to set up a LAN based computer lab in your organization/institution. Identify the requirements. Explain all the components involved with their specification. List all the assumptions.* **(10 M)**

Dec. 06 [Q. 2(a)] *Identify the requirements to setup a LAN based laboratory with 20 terminals. The lab should provide Linux, Windows and database servers. List all the components with their specifications including various softwares. List the assumptions and security features provided.* **(10 M)**



(1) An institute can have N number of labs.

(2) All hosts within a lab can be connected using a Hub. Hub: Basically a hub is a device into which many devices can be connected at the same time.

(3) Now all the Hubs in the different labs can be connected using a switch. A switch performs efficient bridging.

(4) This switch is connected to a router. Routers decide on the most efficient path that the packets should take while flowing from one network to another.

A router can be used to connect the network of one building to a network in another building.

(5) Software required :

- Network drivers
- Protocol Stacks
- Operating Systems
- Web Server
- Browsers
- Dial up software
- Firewall software

(6) Hardware Required

- Computers
- NIC
- Cables
- Hubs
- Switches
- Router
- Modem
- Printer
- Scanner

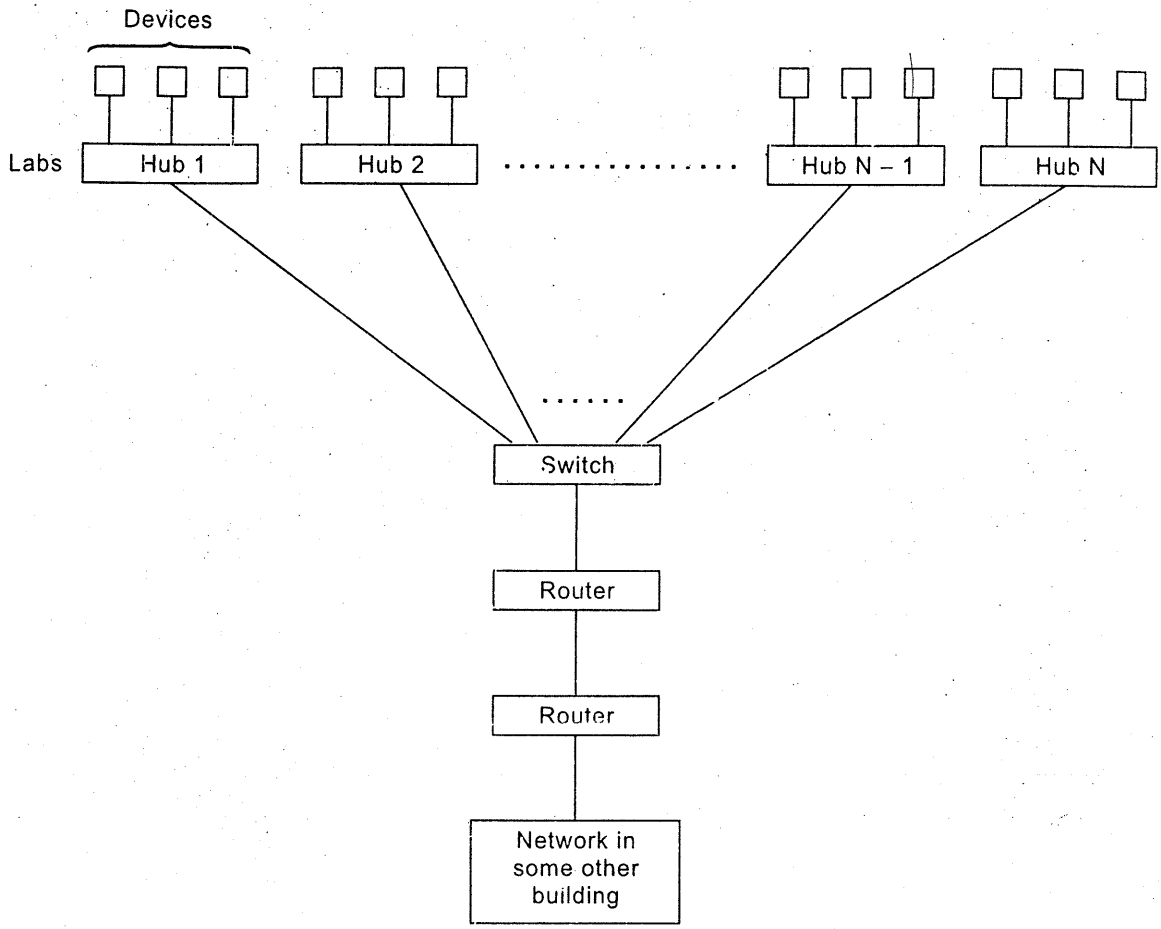


Fig. 2.21 : LAN system for an institute.

3 DATA LINK LAYER (DLL)

DLL is Layer 2 of the ISO-OSI model. Here concepts related to DLL designing, elementary DLL protocols, sliding window protocols, HDLC and PPP are explained.

Marks	
Dec. 03	40 M
May 04	20 M
Dec. 04	20 M
May 05	10 M
Dec. 05	31 M
May 06	-
Dec. 06	20 M
May 07	10 M

3.1 Data Link Layer Design Issues

Dec. 06 [Q. 3(b)] *Describe any five functions of data link layer with suitable examples.* **(10 M)**

The functions of the DLL are :

(1) **Providing Services to the Network Layer :** The DLL takes packets from the network layer of the source machine and transfers them to the network layer on the destination machine. The services provided for this purpose are :

(i) *Unacknowledged Connectionless Service* : No acknowledgements are used. It is a connectionless service.

(ii) *Acknowledged Connectionless Service* : Acknowledgements are used. It is a connectionless service.

(iii) *Acknowledged Connection-oriented Service* : Acknowledgements are used. It is a connection-oriented service.

(2) **Error Control :** To achieve Error control the following techniques are used :

(i) *Acknowledgements* : When the receiver correctly receives the data, it sends an acknowledgement to the sender. (Used in the Stop and Wait protocol)

(ii) *Timer* : The sender maintains a timer which is set to a time which is enough for the data to reach the receiver and for the acknowledgement from the receiver to reach back to the sender.

(iii) *Sequence Numbers* : Sequence Numbers are used by the receivers to decide if they are receiving new frames or duplicate frames.(Used in Simplex Protocol for a Noisy Channel)

(3) **Flow Control :** DLL regulates the flow of data so that receivers are not swamped by fast senders.

(4) Framing

(a) The DLL takes packets from the Network Layer and converts them into frames. The components of a frame are :

(i) Header (ii) Payload Field (iii) Trailer.

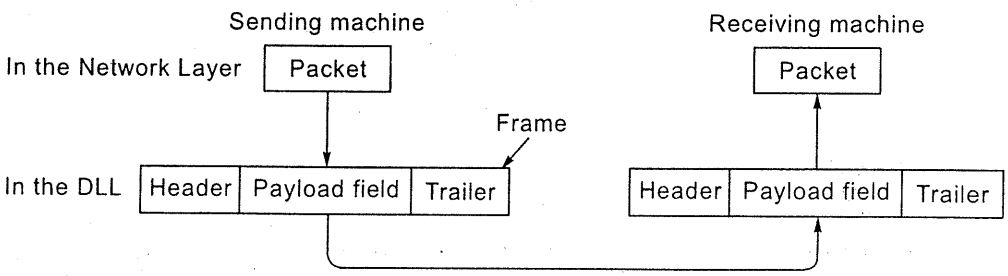
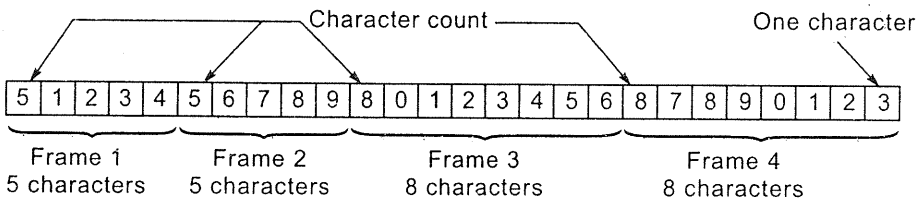


Fig. 3.1 : Relationship between packets and frames.

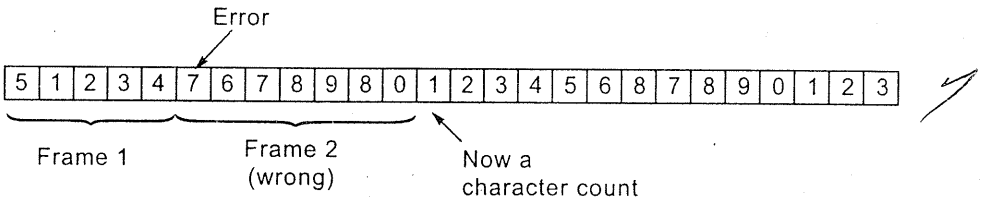
DLL divides the stream of bits received from the network layer into manageable data units called *frames*.

(b) Techniques of Framing

(i) Character Count : The first field in the header specifies the number of characters in the frame. Refer *figure 3.2(a)*.



(a) Without errors.



(b) With one error.

Fig. 3.2 : Character count.

- **Disadvantage :** Count may get garbled by a transmission error. Refer *figure 3.2(b)*.

(ii) Character Stuffing

- Each frame starts and ends with a FLAG byte. Thus adjacent frames are separated by two flag bytes.
- *Problem :* It is possible that FLAG is actually a part of the data.

- *Solution :* At the sender an ESC character is inserted just before the FLAG byte present in the data. At the receiver the ESC is removed from the data.

  Now if an ESC is present in the data then an extra ESC is inserted before it in the data. This extra ESC is removed at the receiver.

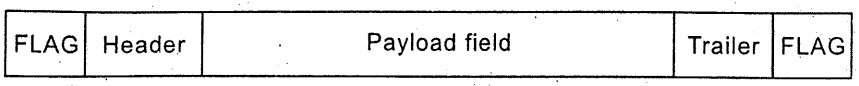


Fig. 3.3(a) : A frame delimited by flag bytes.

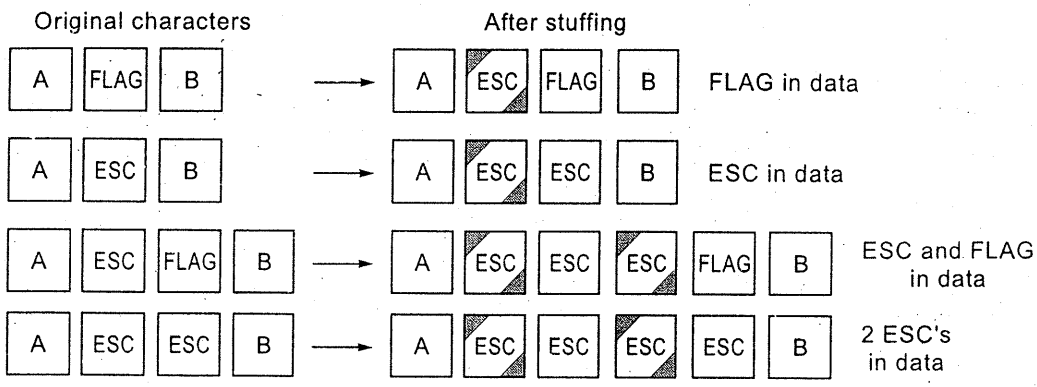


Fig. 3.3(b) : Four examples of byte sequences before and after byte stuffing. The shaded blocks represent the stuffed characters.

(iii) Bit Stuffing

- Each frame begins and ends with a special bit pattern 01111110 called the *flag byte.*
- When senders DLL encounters 5 consecutive 1's in the data it automatically stuffs a '0' bit in the outgoing data stream.
- When the receiver encounters 5 consecutive 1's followed by a '0' it removes the '0' bit.

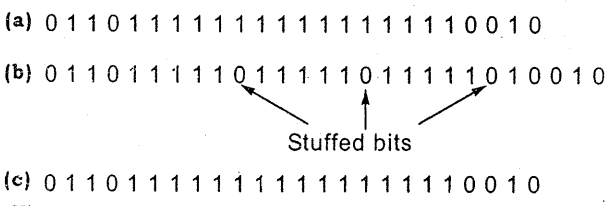


Fig. 3.4 : Bit stuffing. (a) The original data. (b) The data as they appear on the line. (c) The data as they are stored in the receiver's memory after destuffing.

(5) DLL performs Error Detection using CRC (section 3.3) and Error Correction using Hamming Code (section 3.2).

3.2 Hamming Code (Error Correcting Code)

The key to the Hamming Code is the use of extra parity bits to allow the identification of a single error.

Create the code word as follows :

(1) Mark all bit positions that are powers of two as parity bits. (positions 1, 2, 4, 8, 16, 32, 64, etc.)

(2) All other bit positions are for the data to be encoded. (positions 3, 5, 6, 7, 9, 10, 11, 12, 13, 14, 15, 17, etc.)

(3) Each parity bit calculates the parity for some of the bits in the code word.

(4) The position of the parity bit determines the sequence of bits that it alternately checks and skips.

- Position 1 : check 1-bit, skip 1-bit, check 1-bit, skip 1-bit, etc. (1, 3, 5, 7, 9, 11, 13, 15, ...)
- Position 2 : check 2-bits, skip 2-bits, check 2-bits, skip 2-bits, etc. (2, 3, 6, 7, 10, 11, 14, 15, ...)
- Position 4 : check 4-bits, skip 4-bits, check 4-bits, skip 4-bits, etc. (4, 5, 6, 7, 12, 13, 14, 15, 20, 21, 22, 23,...)
- Position 8 : check 8-bits, skip 8-bits, check 8-bits, skip 8-bits, etc. (8-15, 24-31, 40-47, ...)
- Position 16 : check 16-bits, skip 16-bits, check 16-bits, skip 16-bits, etc. (16-31, 48-63, 80-95,...)
- Position 32 : check 32-bits, skip 32-bits, check 32-bits, skip 32-bits, etc. etc. (32-63, 96-127, 160-191, ...)

(5) Set a parity bit to 1 if the total number of ones in the positions it checks is odd. Set a parity bit to 0 if the total number of ones in the positions it checks is even.

Here is an example :

A byte of data : 1 0 0 1 1 0 1 0

Create the data word, leaving spaces for the parity bits : _ _ 1 _ 0 0 1 _ 1 0 1 0

Calculate the parity for each parity bit

- Position 1 checks bits 1, 3, 5, 7, 9, 11 :
  ? _ 1 _ 0 0 1 _ 1 0 1 0. Even parity, so set position 1 to a 0 : ***0*** *_ 1 _ 0 0 1 _ 1 0 1* 0
- Position 2 checks bits 2, 3, 6, 7, 10, 11 :
  0 *? 1 _ 0 0 1 _ 1 0 1* 0. Odd parity, so set position 2 to a 1 : 0 ***1 1*** *_ 0 0 1 _ 1 0 1* 0
- Position 4 checks bits 4, 5, 6, 7, 12 :
  0 1 1 *? 0 0 1 _ 1 0 1* ***0***. Odd parity, so set position 4 to a 1 : 0 1 1 ***1 0 0 1*** *_ 1 0 1* ***0***
- Position 8 checks bits 8, 9, 10, 11, 12 :
  0 1 1 1 0 0 1 *? 1 0 1 0*. Even parity, so set position 8 to a 0: 0 1 1 1 0 0 1 ***0 1 0 1 0***
- Code word : 0 1 1 1 0 0 1 0 1 0 1 0. .

Finding and Fixing a Bad Bit

The above example created a code word of 0 1 1 1 0 0 1 0 1 0 1 0. Suppose the word that was received was 0 1 1 1 0 0 1 0 1 1 1 0 instead. Then the receiver could calculate which bit was wrong and correct it. The method is to verify each check bit. Write down all the incorrect parity bits. Doing so, you will discover that parity bits 2 and 8 are incorrect. It is not an accident that 2 + 8 = 10, and that bit position 10 is the location of the bad bit. In general, check each parity bit, and add the positions that are wrong, this will give you the location of the bad bit.

Dec. 05 [Q. 4(b)] *Find Hamming code for 1001101.* **(5 M)**

Solution :

Step 1 : Parity Bit Positions : _ _ 1 _ 0 0 1 _ 1 0 1

Step 2 : For bit 1 : Check positions 3 = 1, 5 = 0, 7 = 1, 9 = 1, 11 = 1.
Even, therefore bit 1 = 0.

Step 3 : For bit 2 : Check positions 3 = 1, 6 = 0, 7 = 1, 10 = 0, 11 = 1.
Odd, therefore bit 2 = 1.

Step 4 : For bit 4 : Check positions 5 = 0, 6 = 0, 7 = 1. Odd, therefore bit 4 = 1.

Step 5 : For bit 8 : Check positions 9 = 1, 10 = 0, 11 = 1. Even, therefore bit 8 = 1.

Answer : 0 1 1 1 0 0 1 1 1 0 1

3.3 Cyclic Redundancy Check (CRC) a.k.a. Polynomial (Error Detecting Code)

May 04 [Q. 3(a)] *What is CRC ? Write the algorithm for computing checksum.* **(10 M)**

Dec. 06 [Q. 4(b)] *What is CRC ? Write the algorithm for computing checksum and explain with suitable example.* **(10 M)**

(1) In CRC we use strings of '0' and '1' bits to represent coefficients of polynomials.

(2) A polynomial of degree K will have K+1 bits in the bit string.
E.g. 1 1 0 0 0 1 represents the polynomial
$1x^5 + 1x^4 + 0x^3 + 0x^2 + 0x^1 + 1x^0 = 1x^5 + 1x^4 + 1x^0$

(3) The sender and the receiver must agree upon a generator polynomial G(x) in advance. The most significant bit and the least significant bit of G(x) must be 1.

(4) Calculating Checksum :

- Let G(x) be the generator polynomial and M(x) be the actual polynomial.
- Let r be the degree of G(x).
- Let m be the number of bits in the bit string of M(x).
- Append r '0' bits to the lower-order end of the frame.
- Now the bit string will contain m + r bits. And corresponds to the polynomial x^r. M(x)
- Divide the new bit string, x^r. M(x), by G(x) (In this case we use Modulo 2 Division)

- Now subtract the remainder if any from x^r. M(x) using Modulo 2 Subtraction.
- Now this gives the frame to be transmitted called the *Checksummed Frame.*

(Note : Modulo 2 Division uses Modulo 2 Subtraction. Modulo 2 Subtraction is similar to X-OR)

Examples

(1) Dec. 03 [Q. 1(b)], Dec. 05 [Q. 6(c)] *Consider an error detecting CRC with the generator 1 0 1 0 1. Assume the CRC bits follows the data bits in any transmission.*

(i) *Compute the transmitted bit sequence for the data bit sequence 0 1 1 0 1 1 0 1.*

(ii) *The string of bits 1 1 0 0 1 1 0 0 1 1 0 0 is received. Is it acceptable, and if so what is the data bit sequence.* **(10 M)**

(i)

Step 1 : G(x) = 1 0 1 0 1

M(x) = 0 1 1 0 1 1 0 1

Step 2 : r = 4

Step 3 : m = 8

Step 4 : x^r M(x) = 0 1 1 0 1 1 0 1 0 0 0 0 [As r = 4 we append 4 zeros to M(x)]

Step 5 : Modulo 2 division

```
            1 1 1 1 0 1 1 1
        ---------------------
1 0 1 0 1 | 0 1 1 0 1 1 0 1 0 0 0 0
          | 1 0 1 0 1
          | -----------
          | 1 1 0 0 0
          |   1 0 1 0 1
          |   -------
          |   0 1 1 0 1 1
          |     1 0 1 0 1
          |     -------
          |     0 1 1 1 0 0
          |       1 0 1 0 1
          |       -------
          |       0 1 0 0 1 1
          |         1 0 1 0 1
          |         -------
          |         0 0 1 0 0 0 0
          |           1 0 1 0 1
          |           -------
          |           0 1 1 0 1 0
          |             1 0 1 0 1
          |             -------
          |             0 1 1 1 1 0
          |               1 0 1 0 1
          |               -------
          |               0 1 0 1 1
```

Step 6 : Modulo 2 subtraction

```
  0 1 1 0 1 1 0 1 0 0 0 0
- -------------- 0 1 0 1 1
  0 1 1 0 1 1 0 1 1 0 1 1
```

Step 7 : T = $\underbrace{0\,1\,1\,0\,1\,1\,0\,1\,1\,0\,1\,1}_{\text{Message}\quad\text{Check-sum}}$

(ii) Received bit string = 1 1 0 0 1 1 0 0 1 1 0 0

For the received string to be accepted it should give zero remainder on dividing by G(x).

```
10101 | 1 1 0 0 1 1 0 0 1 1 0 0
        1 0 1 0 1
        0 1 1 0 0 1
          1 0 1 0 1
          0 1 1 0 0 0
            1 0 1 0 1
            0 1 1 0 1 0
              1 0 1 0 1
              0 1 1 1 1 1
                1 0 1 0 1
                0 1 0 1 0 1
                  1 0 1 0 1
                  0 0 0 0 0 0 0
```

As the remainder = 0. The string is accepted.

(2) May 05 [Q. 5(b)] *For Message Frame 1 1 0 1 0 1 1 0 1 1 and G(x) = $x^4 + x + 1$. Show transmitted frame.* **(5 M)**

Step 1 : $G(x) = x^4 + x + 1$

$= 1\,x^4 + 0\,x^3 + 0\,x^2 + 1\,x^1 + 1\,x^0$

$= 1 \quad 0 \quad 0 \quad 1 \quad 1$

$M(x) = 1\,1\,0\,1\,0\,1\,1\,0\,1\,1$

Step 2 : $r = 4$

Step 3 : $m = 10$

Step 4 : $x^r M(x) = 1\,1\,0\,1\,0\,1\,1\,0\,1\,1\,0\,0\,0\,0$ [Append 4 zeros to M(x) as r = 4]

Step 5 : Modulo 2 division

```
                1 1 0 0 0 0 1 0 1
10011 | 1 1 0 1 0 1 1 0 1 1 0 0 0 0
        1 0 0 1 1
        0 1 0 0 1 1
          1 0 0 1 1
          0 0 0 0 0 1 0 1 1 0
                1 0 0 1 1
                0 0 1 0 1 0 0
                    1 0 0 1 1
                    0 0 1 1 1 0  → Remainder
```

Step 6 : Mo

3.4 Eleme

Dec. 03 [Q. *window prot*

(1) Unrestr

- **Assu**
  - (i)
  - (ii)
  - (iii)
- The
  The
  the c
- In th
- The

```
/* Protoco
   sender
   free an
   infinit
   data ou

typedef en
#include "

void sende
{
  frame s;
  packet bu

  while (t
     from_
     s.info
     to_phy
     }
}
```

dividing

Step 6 : Modulo 2 subtraction

```
  1 1 0 1 0 1 1 0 1 1 0 0 0 0
- 1 1 1 0
-------------------------------
  1 1 0 1 0 1 1 0 1 1 1 1 1 0
```

Step 7 : $\therefore$ T = $\underbrace{1\,1\,0\,1\,0\,1\,1\,0\,1\,1\,1\,1\,0}_{\text{Message}}\ \underbrace{1\,1\,1\,0}_{\text{Check-sum}}$

3.4 Elementary Data Link Protocols

Dec. 03 [Q. 8(a)], May 05 [Q. 1(c)], May 07 [Q. 7(b)] *Explain stop and wait and sliding window protocol with example and suitable diagrams.* **(10 M)**

(1) Unrestricted Simplex Protocol

- **Assumptions**
  - (i) Simplex means that data is transmitted in one direction only.
  - (ii) Infinite Buffer Space is available at the receiver.
  - (iii) Communication channel is error free.
- The basic working is that the sender sends data onto the line as fast as it can. The data is saved on the receiver's buffer. The receiver accepts and processes the data as and when it can.
- In this case acknowledgements are not used.
- The pseudocode for the unrestricted simplex protocol is given by

+ x + 1.
(5 M)

```
/* Protocol provides for data transmission in one direction only, from
   sender to receiver. The communication channel is assumed to be error
   free and the receiver is assumed to be able to process all the input
   infinitely quickly. Consequently, the sender just sits in a loop pumping
   data out onto the line as fast as it can. */

typedef enum {frame_arrival} event_type;
#include "protocol.h"

void sender1(void)
{
  frame s;                          /* buffer for an outbound frame */
  packet buffer;                    /* buffer for an outbound packet */

  while (true) {
     from_network_layer(&buffer);   /* go get something to send */
     s.info = buffer;               /* copy it into s for transmission */
     to_physical_layer(&s);         /* send it on its way */
     }
}
```

```
void reveiver1(void)
{
  frame r;
  event_type event;              /* filled in by wait, but not used here */

  while (true) {
     wait_for_event(&event);     /* only possibility is frame_arrival */
     from_physical_layer(&r);    /* go get the inbound frame */
     to_network_layer(&r.info); /* pass the data to the network layer */
     }
}
```

Fig. 3.5 : An unristricted simplex protocol.

Note : This pseudocode has never been asked in the exam.

(2) Simplex Stop and Wait Protocol

- **Assumptions**
  - (i) Data is delivered in one direction only as it is of the Simplex form.
  - (ii) There is NO infinite buffer space available at the receiver. Therefore the problem in this case is that the receiver may not be able to process data as fast as the sender is sending the data. This leads to a problem called *Flooding*.
  - (iii) Communication channel is assumed to be error free.

- *Problem* : Flooding of a slow receiver by a fast sender.

  Solution : In this case the sender cannot send data as and when it wants. Instead a sender sends one frame; when the receiver receives the frame it sends an acknowledgement to the sender which allows the sender to send the next frame. The sender cannot send frames as and when it wants to but it must wait for an acknowledgement before sending a frame.
- Communication channel will be half duplex as there is an alternation of flow between the sender (who sends the data frame) and the receiver (who sends the acknowledgement).
- **Advantage** : Simplicity.
- **Disadvantage** : Inefficiency to handle heavy loads as receiver has to keep sending an acknowledgement before the sender can send a new frame.
- We can see that the name Stop and Wait is given to the protocol because the sender sends a frame, then STOPs transmission and has to WAIT for an acknowledgement before sending a new frame.

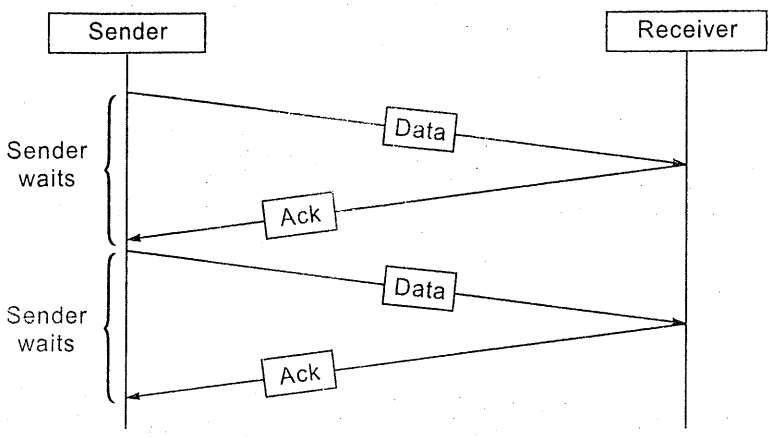


Fig. 3.6 : Simplex stop and wait protocol.

Calculation of Efficiency of the Stop and Wait Protocol

$$\text{Utilization } \mu = \frac{t_f}{t_f + 2\,t_p} \quad \text{(Standard formula)}$$

where t_f = Time required to transmit a standard frame

t_p = Propagation delay (i.e. time required for a bit to go from transmitter to receiver)

$$\text{Now, } \mu = \frac{1}{1 + 2\frac{t_p}{t_f}}$$

$$\text{Let } A = \frac{t_p}{t_f}$$

$$\therefore \mu = \frac{1}{1 + 2A} \Rightarrow \eta \text{ (efficiency)} = \frac{1}{1 + 2A} \times 100\ \%$$

Conclusion : As $\mu\uparrow$, $A\downarrow$ $\therefore \mu \propto t_f$ $\left(\text{Since } A = \frac{t_p}{t_f}\right)$

- The pseudocode for the simplex stop and wait protocol is given by

```
/* Protocol 2 (stop-and-wait) also provides for a one-directional flow of
   data from sender to receiver. The communication channel is once again
   assumed to be error free, as in protocol 1. However, this time, the
   receiver has only a finite buffer capacity and a finite processing
   speed, so the protocol must explicitly prevent the sender from flooding
   the receiver with data faster than it can be handled. */

typedef enum {frame_arrival}event_type;
#include "protocol.h"
void sender2(void)
{
  frame s;                    /* buffer for an outbound frame */
  packet buffer;              /* buffer for an outbound packet */
```

```
    event_type event;          /* frame_arrival is the only possibility */
    while (true) {
        from_network_layer(&buffer);  /* go get something to send */
        s.info = buffer;              /* copy it into s for transmission */
        to_physical_layer(&s);        /* send frame */
        wait_for_event(&event);       /* wait for acknowledgment */
    }
}

void receiver2(void)
{
    frame r, s;                       /* buffers for frames */
    event_type event;          /* frame_arrival is the only possibility */
    while (true) {
        wait_for_event(&event);       /* only possibility is frame_arrival */
        from_physical_layer(&r);      /* go get the inbound frame */
        to_network_layer(&r.info);    /* pass the data to the network layer */
        to_physical_layer(&s);        /* send a dummy frame to awaken sender */
    }
}
```

Fig. 3.7 : A simplex stop-and-wait protocol.

Note : This pseudocode has never been asked in the exam.

(3) Simplex Protocol for a Noisy Channel a.k.a. Positive Acknowledgement with Retransmission (PAR) a.k.a. Automatic Repeat Request (ARQ)

- **Assumptions**
  - (i) Data is delivered in one direction only (as it is of the Simplex form.)
  - (ii) There is NO infinite buffer space available at the receiver. Therefore the problem in this case is that the receiver may not be able to process data as fast as the sender is sending the data. This leads to a problem called Flooding. Flooding is solved in the same way as in case of the Stop and Wait Simplex Protocol.
  - (iii) Communication channel is assumed to have errors.
- *Problem* : Communication channel is assumed to have errors. Lost and damaged frames have to be re-transmitted.

  Solution : Here the sender maintains a Timer. The timer is set to a time which is enough for the data frames to reach the receiver + the acknowledgement to travel from the receiver to the sender. Now if the acknowledgement is not received by the sender in the specified time; it signifies that the frame sent by the sender is either lost or damaged and must be re transmitted.

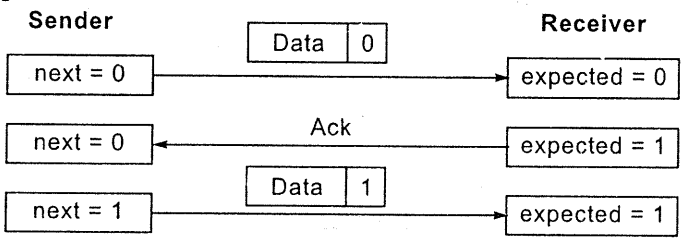
- *Problem with Above Solution :* There might be a case in which the frame is sent properly from the sender to the receiver. The receiver sends an acknowledgement to the sender. But what happens if this acknowledgement is lost ? Now as the sender has not received the acknowledgement the timer at the sender will timeout signifying that the sender must retransmit the frame again. But this is not right since the receiver has already correctly received the frame and this frame is actually a duplicate frame.

  Solution : Each frame has a sequence number. Now if the above problem occurs the receiver can check the sequence number to see if it is receiving a duplicate frame or a new frame.

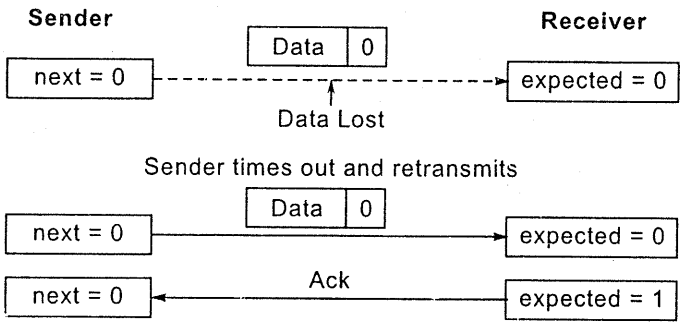
  The sender remembers the sequence number of the next frame to send. The receiver remembers the sequence number of the next frame expected.

Message Format : | Data | Seq. No. |

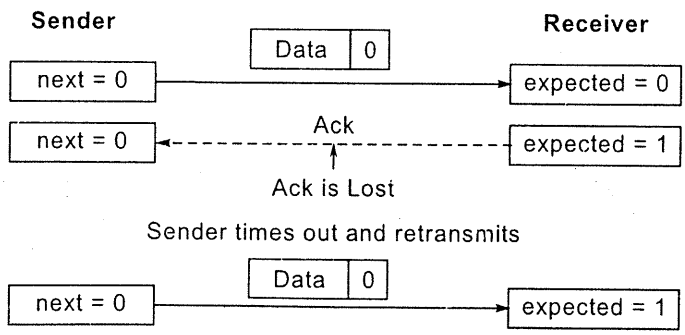
Normal Case



Data Lost



Ack Lost



Receiver checks the sequence number and realizes that this is a duplicate frame. The receiver therefore discards this frame.

- The pseudocode for the simplex protocol for a noisy channel is given by

```
/* Protocol 3 (par) allows unidirectional data flow over an unreliable
   channel. */

#define MAX_SEQ 1           /* must be 1 for protocol 3 */
typedef enum {frame_arrival, cksum_err, timeout} event_type;
#include "protocol.h"

void sender3(void)
{
  seq_nr next_frame_to_send;     /* seq number of next outgoing frame */
  frame s;                       /* scratch variable */
  packet buffer;                 /* buffer for an outbound packet */
  event_type event;

  next_frame_to_send = 0;        /* initialize outbound sequence numbers */
  from_network_layer(&buffer);   /* fetch first packet */
  while (true) {
   s.info = buffer;              /* construct a frame for transmission */
   s.seq = next_frame_to_send;   /* insert sequence number in frame */
   to_physical_layer(&s);        /* send it on its way */
   start_timer(s.seq);           /* if answer takes too long, time out */
   wait_for_event(&event);       /* frame_arrival, cksum_err, timeout */
   if (event==frame_arrival) {
     from_physical_layer(&s);    /* get the acknowledgement */
     if (s.ack==next_f rame_to_send) {
        stop_timer(s.ack);  /* turn the timer off */
        from_network_layer(&buffer);  /* get the next one to send */
        inc(next_frame_to_send);      /* invert next_frame_to_send */
     }
   }
  }
}

void receiver3(void)
{
  seq_nr frame_expected;
  frame r, s;
  event-type event;
```

```
    frame-expected = 0;
    while (true) {
     wait_for_event(&event);   /* possibilities: frame_arrival, cksum_err */
     if(event == frame-arrival){      /* a valid frame has arrived. */
       from_physical_layer(&r);       /* go get the newly arrived frame */
       if(r.seq == frame_expected){ /*this is what we have been waiting for*/
          to_network_layer(&r.info); /* pass the data to the network layer */
          inc(frame_expected);   /* next time expect the other sequence nr */
       }
       s.ack = 1 - frame_expected;    /* tell which frame is being acked */
       to_physical_layer(&s);         /* send acknowledgement */
     }
    }
}
```

Fig. 3.8 : A positive acknowledgement with retransmission protocol.

Note : This pseudocode has never been asked in the exam.

3.5 Sliding Window Protocol

Dec. 03 [Q. 7(a)] *Which protocol Go-Back-N or selective-Repeat makes more efficient use of network bandwidth ? Why ?* **(10 M)**

Dec. 03 [Q. 8(a)] *Explain stop and wait and sliding window protocol.* **(10 M)**

May 04 [Q. 3(b)] *Explain a one - bit sliding window protocol in detail.* **(10 M)**

May 05 [Q. 1(c)], May 07 [Q. 7(b)] *Explain stop and wait and sliding window protocol with example and suitable diagrams.* **(10 M)**

Dec. 05 [Q. 6(a)] *Explain n bit sliding window protocol.* **(6 M)**

(1) Frames are transmitted in both directions.

(2) It requires a full duplex communication channel.

(3) It uses *Piggy Backing* which means that the outgoing acknowledgement is delayed so that they can be hooked to the next outgoing data frame.

(4) *Sending Window* : Represents the frame numbers that have been sent but are yet not acknowledged.

(5) *Receiving Window* : Represents the frame numbers that the receiver can accept.

The sliding window protocols are

[I] 1-Bit Sliding Window Protocol

(i) 1-bit sliding window protocol has a maximum window size of 1. (i.e. the sending and receiving window has the size = 1-bit)

(ii) It uses Stop and Wait because, sender transmits a frame and waits for an acknowledgement before sending next frame.

(iii) *Working*

(a)

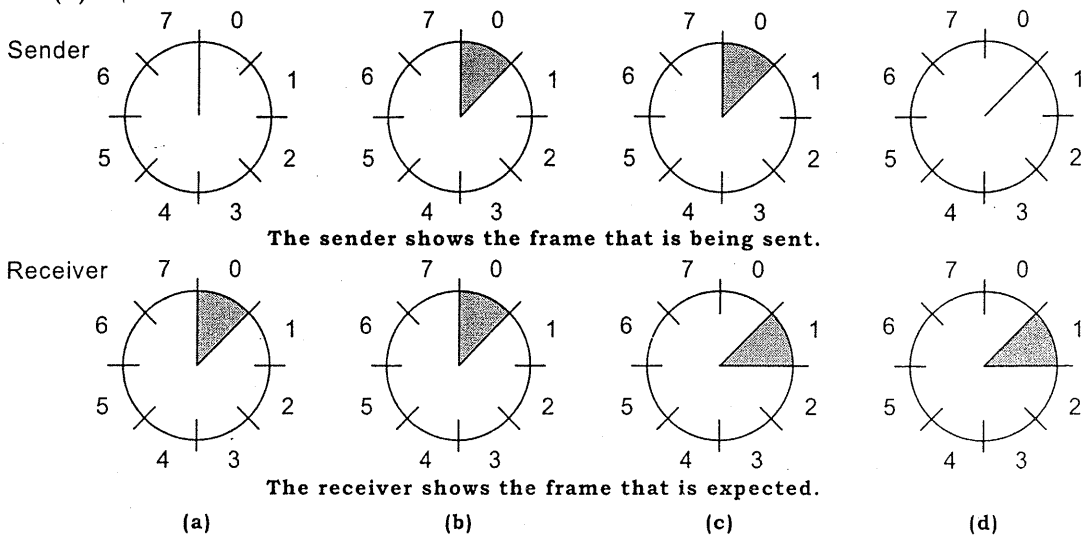


Fig. 3.9 : Sliding window of size 1, with a 3-bit sequence number. (a) Initially. (b) After the first frame has been sent. (c) After the first frame has been received. (d) After the first acknowledgement has been received.

(b)

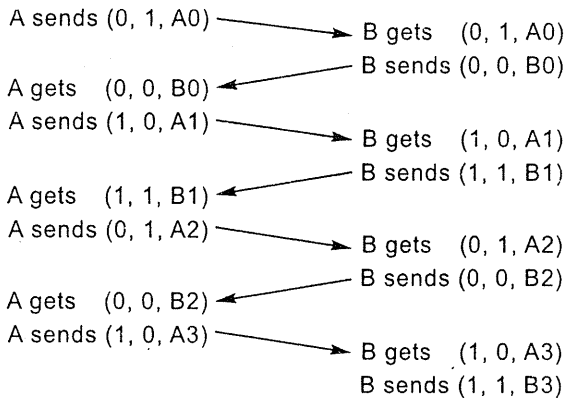


Fig. 3.10 : 1-Bit sliding window. The message format is (Sequence Number, Acknowledgement Number, Packet Number)

- A sends packet with seq. no. : 0, ack. no. : not important in the first step (assume it to be 1), packet no. : A0.
- B gets packet with seq. no. : 0, ack. no. : 1, packet no. : A0.
- B sends packet with seq. no. : 0, it acknowledges that it has received the packet with seq. no. : 0 from A and hence makes ack. no. : 0. The packet that B sends is : B0.
- A gets packet with seq. no. : 0, ack. no. : 0, packet no. : B0.
- A sends packet with seq. no. : 1, it acknowledges that it has received the

packet with seq. no. : 0 from B and hence makes ack. no. : 0. The packet no. it sends is : A1.

- B gets packet with seq. no. : 1, ack. no. : 0, packet no. : A1.
- B sends packet with seq. no. : 1, it acknowledges that it has received the packet with seq. no. : 1 from A and hence makes ack. no. : 1. The packet that it sends is : B1.
- The process continues.

Disadvantages

(1) Slow process since it uses stop and wait protocol.

(2) Data transfer rate is low.

[II] GO BACK N Protocol

(1) In this case when a damaged frame arrives the receiver simply discards all the subsequent frames.

(2) It can transfer more than one frame at a time thus it is faster than the 1-bit sliding window protocol.

(3) **Working :** Sender sends N frames and waits for an acknowledgment, if r^{th} frame is in error, packets received after r^{th} frame will be discarded and it starts re-sending from r^{th} to N^{th} frame.

E.g. Assume window size = 4 frames.

Suppose sender sends frames 0 to 3.

It has to now wait for the acknowledgements before it can proceed.

As each successive acknowledgement is received the window slides forward and the sender can send the next frames.

Suppose an acknowledgement of the 0 frame is lost then the sender discards all the frames after the lost frame (i.e. 1, 2, 3) and retransmits from the 0^{th} frame.

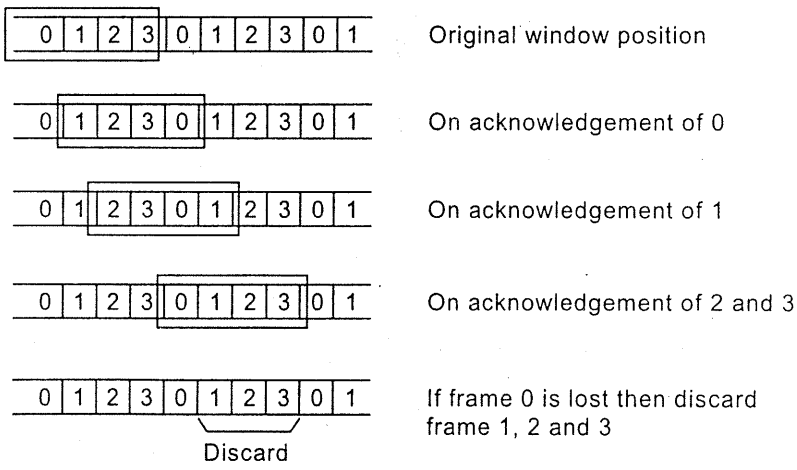


Fig. 3.11 : Go back N protocol.

(4) Max Senders Window Size = $2^k - 1$.

Reason :

(a) Suppose that we keep window size = 2^k.

(b) Assume k = 3. ∴ $2^k = 8$

(c) At time t1 A sends frames 0 to 7 to B.

(d) B receives each of them in order and at time t2 it sends acknowledgement for the most recent frame i.e. frame 7.

Note : In the sliding window protocol acknowledgement of any frame K acts as an acknowledgement of all the frames coming before the frame K. ∴ An ack for frame 7 acts as an ack for all the frames from 0 to 7.

(e) Suppose that this acknowledgement gets lost.

(f) Now B does not know that the acknowledgement it sent is lost.

(g) A does not receive the acknowledgement and times out. On timing out it resends the frames 0 to 7 to B.

(h) But this is a duplicate of the frames that was already received by B.

(i) This problem occurs because 2 consecutive windows contain the same frame numbers.

Reducing the frame size by 1 solves this problem.

(j) Hence maximum sender window size $2^k - 1$.

(5) Max Receivers Window size = 1.

Reason : The frames are always received in order.

(6) The frames must be received in order.

[III] SELECTIVE REPEAT Protocol

(1) The Disadvantage of the 1-Bit sliding window protocol is that only one bit is transferred at a time.

(2) The Disadvantage of the Go Back N protocol is that the frames coming after a corrupt frame are all discarded and again need to be re-transmitted. Thus channel bandwidth is wasted in re-transferring the correct frames coming after the corrupt frame

(3) In Selective Repeat the frames coming after the corrupt frame are stored in a buffer. The corrupt frame is re-transmitted and then the frames from the buffer can be used.

(4) Sending window same as Go Back N protocol.

(5) Receiving window defines the frames that can be received; hence in this case it must be greater than 1.

(6) Receiving need not be in order.

- A frame can be received in any order as long as it is in the receiving window.
- E.g. : If receiving window {0, 1, 2, 3, 4, 5, 6}.
- If arriving frame is present in the receiving window it is buffered. A frame cannot be given to the network layer till all its predecessors have arrived. For every frame in the receiving window there is a buffer.

(7) **Working**

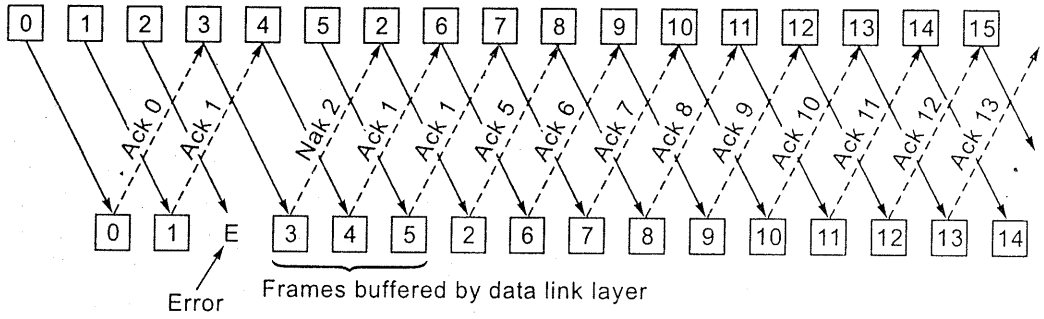


Fig. 3.12 : Selective repeat protocol.

Basic Idea : A bad frame that is received is discarded, but good frames received after it are buffered.

In Selective repeat a *negative acknowledgement* (NAK) is sent by the receiver to the sender when it detects an error. NAKs make the sender retransmit the erroneous frame.

E.g. In the above *figure 3.12* frames 0 and 1 are correctly received and acknowledged. Frame 2 is lost. When frame 3 arrives at the receiver, the receiver notices that it has missed a frame, so it sends back a NAK for 2 but buffers 3. When frames 4 and 5 arrive, they, too, are buffered. Eventually, the NAK 2 gets back to the sender, which immediately resends frame 2. When frame 2 arrives, the data link layer now has 2, 3, 4, and 5 and can pass all of them to the network layer in the correct order. It can also acknowledge all frames up to and including 5, as shown in the *figure 3.11*.

(8) The size of the sending and the receiving window must be 2^{k-1}, because :

- Receiving need not be in order. This causes a problem.
- Consider the error case when the window size used is $2^k - 1$.
- Suppose that we have a 3-bit sequence number. So the sender is permitted to send upto 7 frames before waiting for an acknowledgement. The sender transmits 0 to 6.

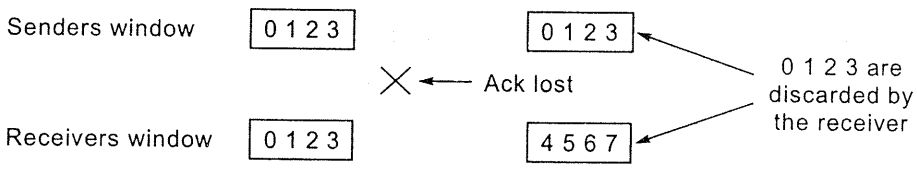
Senders window [0 1 2 3 4 5 6] 7 0 1 2 3 4 5 6 7

Receivers window [0 1 2 3 4 5 6] 7 0 1 2 3 4 5 6 7

- The receiving window allows it to accept any frame with sequence number between 0 and 6. All 7 frames arrive correctly so the receiver acknowledges them by sending an acknowledgement for frame 6. The receiver advances its window to allow the receipt of the next set of sequence number. Which is:

Receivers window 0 1 2 3 4 5 6 [7 0 1 2 3 4 5] 6

- Now suppose the acknowledgement gets lost. The sender has not received an acknowledgement so it times out and resends the frame 0 to 6. The receiver can accept the frames from 0 to 5 since they are present in the receiver window.
- But all these frames are duplicate frames that the receiver already has.
- This problem is caused because, in the receiving window, the new range of sequence numbers overlapped with the old range of sequence numbers.
- The solution is to make sure that after the receiving window slides the old range has nothing in common with the new range.
- This can be done by making the maximum size of the receiving window $= 2^{k-1}$.



3.6 Example Data Link Protocols

[A] HDLC : Bit oriented Protocol

[B] PPP : Used for Internet.

[A] HDLC (High-Level Data Link Control)

Evolution of HDLC

(1) IBM developed SDLC (Synchronous Data Link Control) protocol.

(2) ANSI modified SDLC to become ADCCP (Advanced Data Communication Control Procedure).

(3) ISO modified SDLC to become HDLC (High-level Data Link Control).

Features

(1) HDLC is a bit oriented protocol.

(2) It uses bit stuffing.

HDLC Stations

(1) **Primary Station** : Acts as Master in a Master Slave type link. Primary Stations issue *Commands*.

(2) **Secondary Station** : Acts as a Slave in a Master-Slave type link. Secondary Stations issue *Responses to Commands* given by the primary station.

(3) **Combined Station** : They perform functions of both Primary and Secondary stations. They can issue commands and responses.

Modes of Operation of HDLC

(1) **Normal Response Mode (NRM)** : Works in Master-Slave mode with Primary station acting as Master and Secondary stations acting as Slaves.

(2) **Asynchronous Balanced Mode (ABM)** : All stations in the link are treated as equals with NO Master-Slave setup. It uses Combined Stations.

Frame Structure

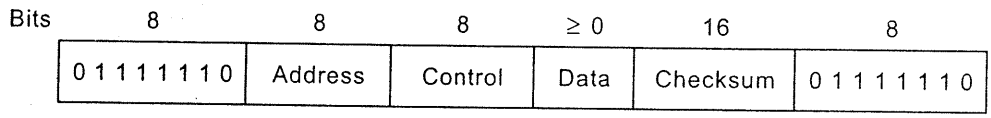


Fig. 3.13 : Frame format for bit-oriented protocols (HDLC).

The frame starts and ends with a bit sequence of 0 1 1 1 1 1 1 0 which is called as a *flag sequence*.

The Address Field : Identifies the terminal.

The Control Field : Used for sequence numbers, acknowledgements, and other purposes.

The Data Field : It contains the data.

The Checksum Field : It is a cyclic redundancy code.

Frame Types :

There are three kinds of frames : Information, Supervisory, and Unnumbered.

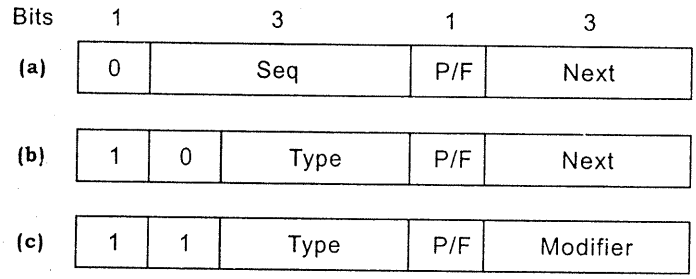


Fig. 3.14 : Control field of (a) an information frame, (b) a supervisory frame, (c) an unnumbered frame.

(a) The Information Frame

(1) The protocol uses a sliding window.

(2) Up to seven frames may be unacknowledged at any instant.

(3) **Seq Field :** It contains a 3-bit frame sequence number.

(4) **Next Field :** It is used to piggyback the acknowledgement. Instead of piggybacking the number of the last frame received correctly, the number of the next frame expected is piggybacked.

(5) **P/F bit stands for Poll/Final.** When used as P, the computer is inviting the terminal to send data. All the frames sent by the terminal, except the final one, have the P/F bit set to P. The final one is set to F.

(b) **Supervisory Frames**

Type Field : Used to specify the different types of the Supervisory frames.

- **Type 0 :** It is an acknowledgement frame used to indicate the next frame expected.
- **Type 1 :** It is used to indicate that a transmission error has been detected. The sender has to re-transmit the frames beginning from frame specified in the NEXT field.
- **Type 2 :** It is used to signal certain temporary problems with the receiver. It acknowledges all frames up to but not including NEXT and it tells the sender to stop sending.
- **Type 3 :** It calls for retransmission of only a specific frame.

(c) **Unnumbered Frame**

(1) It is sometimes used for control purposes.

(2) It can also carry data when unreliable connectionless service is needed.

HDLC Commands

(1) **DISC (DISConnect) :** Command that allows a machine to signal that it is going down.

(2) **SNRM (Set Normal Response Mode) :** Command that allows a machine to signal that it has just come back on-line.

(3) **SABM (Set Asynchronous Balanced Mode)**

(4) **SABME and SNRME :** Same as SABM and SNRM, respectively, except that they enable an extended frame format that uses 7-bit sequence numbers instead of 3-bit sequence numbers.

(5) **FRMR (FRaMe Reject)**

(6) **UA (Unnumbered Acknowledgement) :** It is a Frame which is used to acknowledge a control frame.

(7) **UI (Unnumbered Information) :** This data is not passed to the receiving network layer but is for the receiving data link layer itself.

[B] PPP-

The inter
(hosts and
Some poi
other data

Differenc

(1) PPP i

(2) PPP u

PPP Provi

(1) Frami

(2) The fr

(3) **LCP** (
up, te
longe

(4) **NCP**
indep

Working :
provider t

(1) The PC

(2) After t
connec
param

(3) Once t
config
will be

[B] PPP-The Point-to-Point Protocol : The Data Link Layer in the Internet.

The internet is a network of networks. The Internet consists of individual machines (hosts and routers) and the communication networks that connect them.

Some point-to-point data link protocol is required for *framing, error control, and the other data link layer functions*. The one used in the Internet is called *PPP*.

Difference between PPP and HDLC

(1) PPP is character oriented HDLC is bit oriented.

(2) PPP uses byte stuffing HDLC uses bit stuffing

PPP Provides the Following Features

(1) Framing.

(2) The frame format also handles Error Detection.

(3) **LCP (Link Control Protocol)** : A link control protocol is used for bringing lines up, testing them, negotiating options, and bringing lines down when they are no longer needed.

(4) **NCP (Network Control Protocol)** : Is used to negotiate network-layer options independent of the network layer protocol.

Working : Let us consider the scenario of a home user calling up an Internet service provider to make a home PC a temporary Internet host.

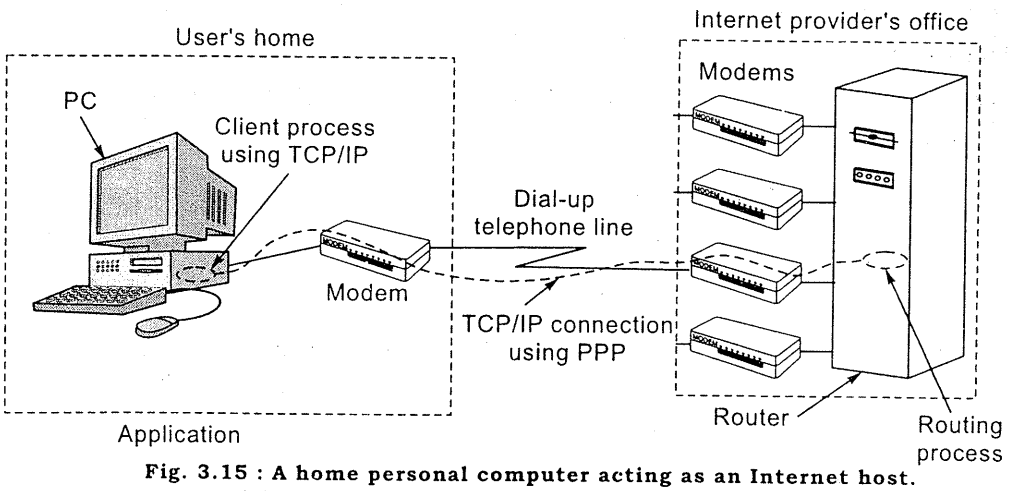


Fig. 3.15 : A home personal computer acting as an Internet host.

(1) The PC first calls the provider's router via a modem.

(2) After the router's modem has answered the phone and established a physical connection, the PC sends the router a series of LCP which select the PPP parameters to be used.

(3) Once the parameters have been agreed upon, a series of NCP packets are sent to configure the network layer. If the TCP/IP protocol stack is used, an IP address will be needed. The NCP assigns the IP address to the host PC.

(4) At this point, the PC is now an Internet host and can send and receive IP packets.

(5) When the user has finished, NCP tears down the network layer connection and frees up the IP address.

(6) Then LCP shuts down the data link layer connection.

(7) Finally, the computer tells the modem to hang up the phone, releasing the physical layer connection.

Frame Format

The PPP frame format was chosen to closely resemble the HDLC frame format

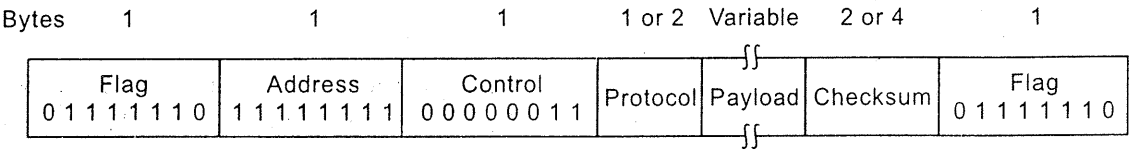


Fig. 3.16 : The PPP full frame format for unnumbered mode operation.

(1) PPP frames begin and end with the standard HDLC flag byte (0 1 1 1 1 1 1 0).

(2) **Address Field :** It is always set to the binary value 1 1 1 1 1 1 1 1 to indicate that all stations are to accept the frame.

(3) **Control Field :** It is always set to the value 0 0 0 0 0 0 1 1. This value indicates an unnumbered frame.

(4) **Protocol Field :** Specifies the kind of packet that is present in the Payload field.

(5) **Payload Field :** It's a variable length field which has the payload.

(6) **Checksum Field :** A 2-byte or 4-byte checksum is stored.

Phases for Bringing a Line Up and Down

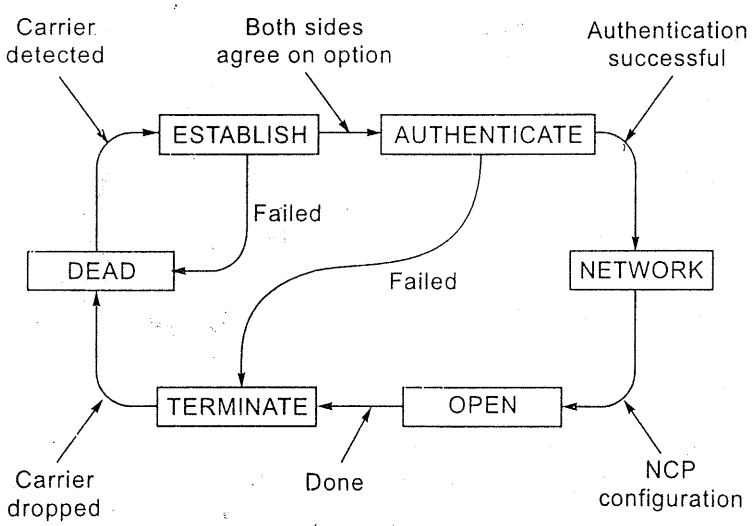


Fig. 3.17 : A phase diagram for bringing a line up and down.

(1) DEAD state means that no physical layer carrier is present and no physical layer connection exists.

(2) After physical connection is established, the line moves to ESTABLISH.

(3) At that point LCP Option Negotiation begins, which, if successful, leads to AUTHENTICATE.

(4) In the AUTHENTICATE phase the two parties can check on each other's identities if desired.

(5) When the NETWORK phase is entered, NCP protocol is invoked to configure the network layer.

(6) If the configuration is successful, OPEN is reached and data transport can take place.

(7) When data transport is finished, the line moves into the TERMINATE phase, and from there, back to DEAD when the carrier is dropped.

3.7 Derivations

Expected Questions : (1) *Derive the transmission efficiency of Go-Back-N ARQ. State the effect of Bit-Error Rate and Delay-Bandwidth Product on the transmission efficiency.* **(10 M)**

(2) *Explain performance issues in ARQ.* **(10 M)**

3.7.1 Derivation of Efficiency of ARQ Protocols

n_t = Number of transmissions to deliver a frame successfully.

n_t = i if the first i – 1 transmissions are erroneous and i^{th} transmission is error free. Therefore

$$P[n_t = i] = (1 - P_f)P_f^{i-1} \quad \text{for } i = 1, 2, 3, \ldots \qquad \ldots(1)$$

For each unsuccessful frame we will have a time out (t_{out}). For each successful frame we will have a transmission delay (t_0).

Thus average time to transmit a frame is

$$E[t_{SW}] = t_0 + \sum_{i=1}^{\infty}(i-1)t_{out}\,P[n_t = i] = t_0 + \sum_{i=1}^{\infty}(i-1)t_{out}\,(1 - P_f)P_f^{i-1}$$

$$= t_0 + \frac{t_{out}\,P_f}{1 - P_f} \qquad \ldots(2)$$

Assuming that $t_{out} = t_0$ we get.

$$E[t_{SW}] = \frac{t_0}{1 - P_f} \qquad \ldots(3)$$

Thus transmission time for stop and wait protocol is

$$R_{eff} = \frac{n_f - n_0}{E[t_{SW}]} = (1 - P_f)\frac{n_f - n_0}{t_0} = (1 - P_f)\, R^0_{eff} \quad ...(4)$$

The transmission efficiency is given by

$$\eta_{SW} = \frac{\frac{n_f - n_0}{E[t_{SW}]}}{R} = (1 - P_f)\frac{1 - \frac{n_0}{n_f}}{1 + \frac{n_a}{n_f} + \frac{2(t_{prop} + t_{proc})R}{n_f}} = (1 - P_f)\eta_0 \quad ...(5)$$

In Go-Back-N ARQ let W_S be the window size. The total time to deliver a frame is

$$\begin{aligned} E[t_{GBN}] &= t_f\left\{1 + W_S \sum_{i=1}^{\infty} (i - 1)\, P[n_t = i]\right\} \\ &= t_f\left\{1 + W_S \sum_{i=1}^{\infty} (i - 1)\,(1 - P_f)P_f^{i-1}\right\} \\ &= t_f\left\{1 + W_S \frac{P_f}{1 - P_f}\right\} = t_f\left\{\frac{1 + (W_S - 1)P_f}{1 - P_f}\right\} \end{aligned} \quad ...(6)$$

Thus the transmission time for Go-Back-N is

$$R_{eff} = \frac{n_f - n_0}{E[t_{GBN}]} = (1 - P_f)\frac{n_f - n_0}{t_f\{1 + (W_S - 1)P_f\}} = (1 - P_f)\frac{1 - \frac{n_0}{n_f}}{1 + (W_S - 1)P_f} R \quad ...(7)$$

The transmission efficiency is

$$\eta_{GBN} = \frac{\frac{n_f - n_0}{E[t_{GBN}]}}{R} = (1 - P_f)\frac{1 - \frac{n_0}{n_f}}{1 + (W_S - 1)P_f} \quad ...(8)$$

In selective Repeat each transmission error has retransmission of only the specific frame. Thus average time to transmit the frame is

$$\begin{aligned} E[t_{SR}] &= t_f\left\{1 + \sum_{i=1}^{\infty} (i - 1)(1 - P_f)P_f^{i-1}\right\} \\ &= t_f\left\{1 + \frac{P_f}{1 - P_f}\right\} = t_f \frac{1}{1 - P_f} \end{aligned} \quad ...(9)$$

The effective transmission time

$$R_{eff} = (1 - P_f)\left(1 - \frac{n_0}{n_f}\right) R \quad ...(10)$$

Thus transmission efficiency is

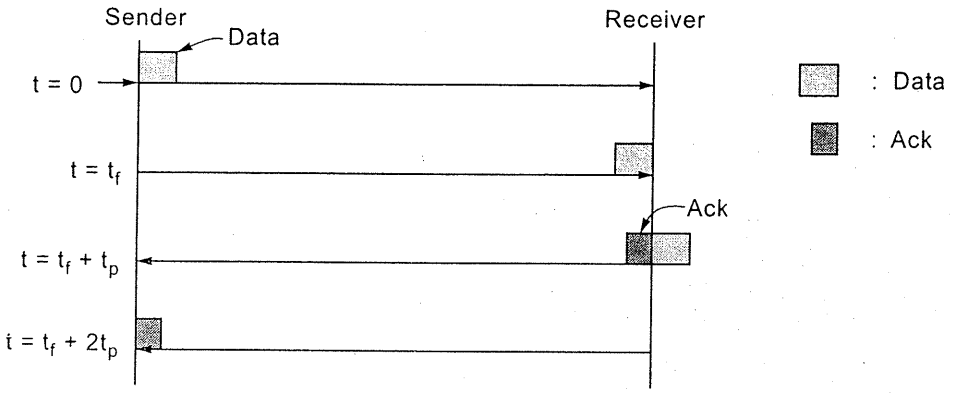
$$\eta_{SR} = (1 - P_f)\left(1 - \frac{n_0}{n_f}\right) \quad ...(11)$$

3.7.2 Protocol Performance

(1) Consider an example where sender transmits "Data" to receiver and receiver sends "Acknowledgment" to the sender.

Let t_f = Time required to transmit a standard frame.

t_p = Propagation delay.



(a) At t = 0, sender starts sending "Data" frame.

(b) At t = t_f, sender finishes transmission of "Data" frame.

(c) At t = t_f + t_p, receiver receives "Data" frame and immediately sends an acknowledgment.

(d) At t = (t_f + t_p) + t_p = t_f + 2t_p, sender receives the acknowledgment "ACK".

(2) As the sender has used the link only for time t_f the efficiency "η" is given by

$$\eta = \frac{t_f}{t_f + 2t_p} = \frac{\text{Amount of time sender uses the link}}{\text{Total amount of time}}$$

$$= \frac{1}{1 + 2A} \quad \text{where } A = \frac{t_p}{t_f}$$

(3) $$A = \frac{t_p}{t_f}$$

but $t_p = \frac{d}{v}$ where d = Distance of link and v = Velocity of propagation

$$\therefore t_p = \frac{d}{v} \left(\text{as time} = \frac{\text{Distance}}{\text{Speed}} \right)$$

Also $t_f = \frac{L}{R}$ where L = Length of frame and R = Rate of transmission.

$$\therefore A = \frac{d/v}{L/R} = \frac{Rd}{Lv}$$

We know that

$$\eta = \mu \propto t_f$$

$$\text{also } \eta = \mu \propto \frac{1}{A} = \frac{Lv}{Rd}$$

But if L increases errors will occur and effective throughput is less. Thus, stop and wait is suitable when propagation time is less (i.e. v is high).

3.8 Exam Questions

Dec. 03 [Q. 8(b)] *1 Gbps CSMA/CD LAN is to be designed over 1 km cable without repeater. The cable supports signal speed of 200,000 km/sec. What is the minimum frame size that Data Link Layer should be consider ?* **(10 M)**

Solution : Data rate = 1 Gbps = 10^9 bps

Length of cable = 1 km

Signal speed = 2×10^5 km/sec

L_{min} = ?

For CSMA/CD for minimum frame size

$$t_f = 2\,t_p$$

2×10^5 km $\rightarrow$ 1 sec

1 km $\rightarrow$ t_p sec

$$\therefore\ t_p = \frac{1}{2 \times 10^5} = 5\ \mu\text{ sec}$$

$$\therefore\ t_p = 5\ \mu\text{ sec}$$

$$\therefore\ t_f = 2\,t_p = 10\ \mu\text{ sec}$$

10^9 bits $\rightarrow$ 1 sec

L_{min} bits $\rightarrow$ 10 μ sec

$$\therefore\ L = \frac{10\ \mu \times 10^9}{1}$$

$$L = 10{,}000 \text{ bits}$$

Dec. 04 [Q. 2(b)] *Explain sliding window protocol. Draw the sender and receiver windows for a system using Go-Back-N sliding window system given that* **(10 M)**

(i) Frame 0 is sent; Frame 0 is ACK

(ii) Frames 1 and 2 are sent; Frames 1 and 2 are ACK.

(iii) Frames 3, 4, 5 are sent. Frame 4 is ACK. Timer for frame 5 expires.

(iv) Frames 5, 6, 7 are sent, Frames 4 through 7 are ACK.

Solution :

In the question the frame numbers are from 0 to 7, i.e. $8 = 2^3$ frames. The size of sending window will be $2^k - 1$, i.e. $2^3 - 1 = 8 - 1 = 7$.

For Go Back N

Size of senders window = 7

Size of receivers window = 1

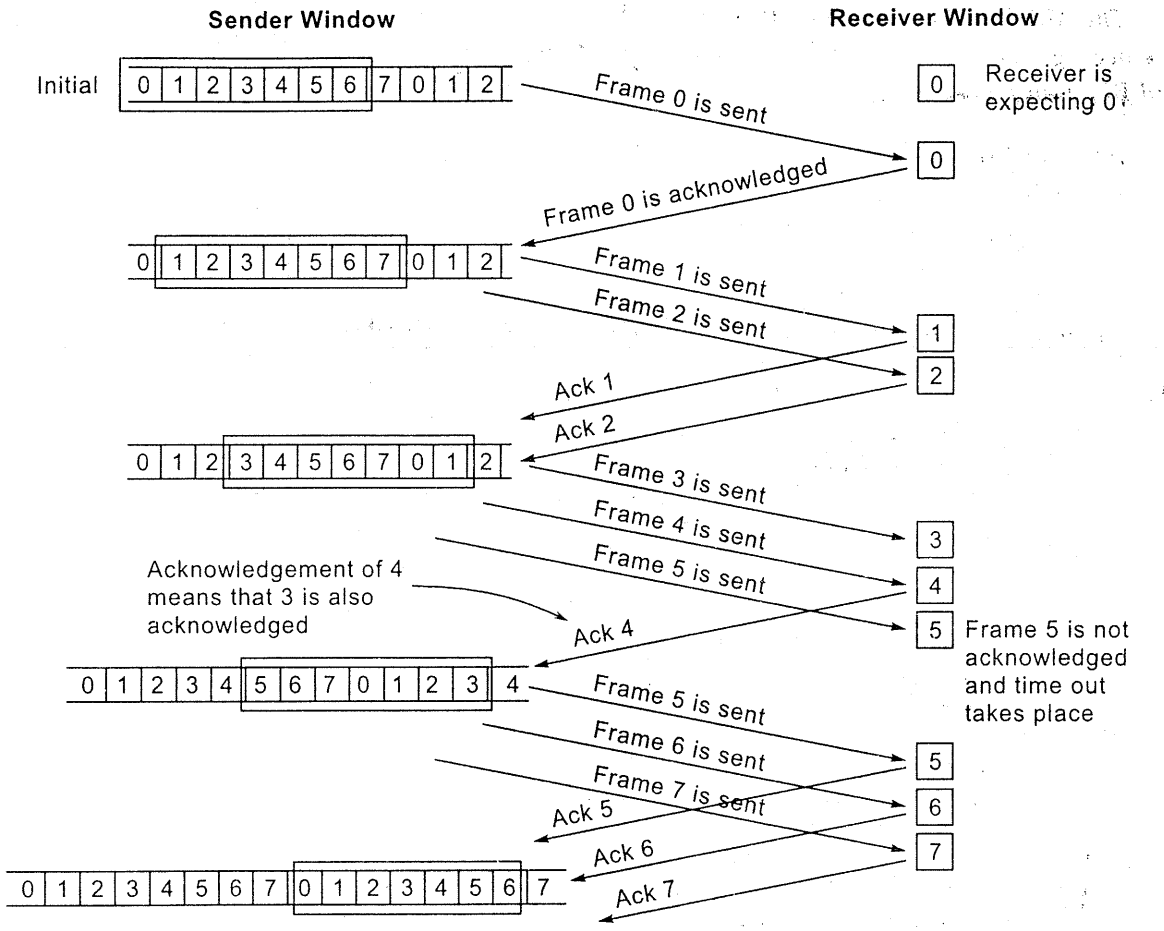


Fig. 3.18 : Go-Back-N example.

4 MEDIUM ACCESS SUBLAYER

The MAC is the lower sublayer of the DLL. Major attention has been given to topics on Multiple Access Protocols, Ethernet and Bluetooth.

Marks	
Dec. 03	20 M
May 04	10 M
Dec. 04	30 M
May 05	26 M
Dec. 05	14 M
May 06	30 M
Dec. 06	25 M
May 07	10 M

Introduction to Medium Access Control (MAC) Sublayer

(1) LAN functions are done in the DLL and the Physical Layer.

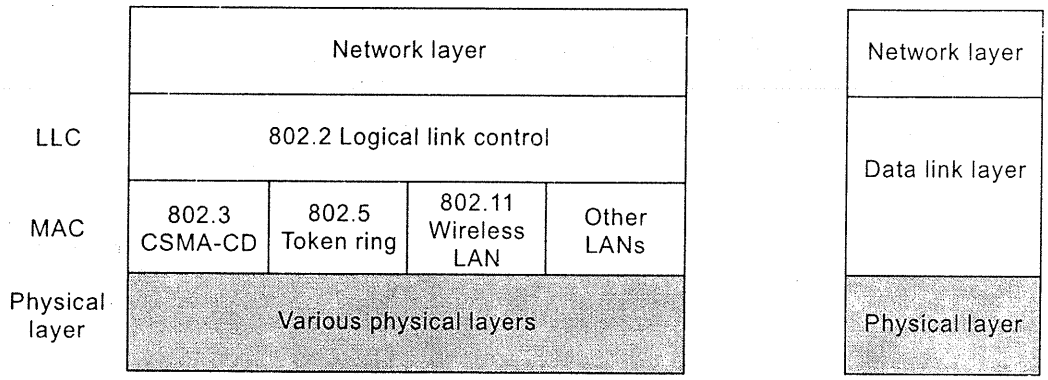


Fig. 4.1 : IEEE 802 LAN standards.

(2) DLL is divided into two layers
   (i) Logical Link Control (LLC) sublayer.
   (ii) MAC sublayer.

(3) MAC sublayer deals with co-ordinating access to the shared physical medium.

(4) There are many MAC standards like 802.3, 802.5 etc.

(5) Each standard has a set of physical layers over which it can operate.

(6) MAC provides *connectionless datagram service* to LLC sublayer.

(7) No error control required as transmission in LANs is relatively error free.

(8) A MAC entity accepts a block of data from LLC or Network layer and constructs a PDU containing
   (i) Source and destination MAC address.
   (ii) CRC check sum.

(9) Thus the main task of a MAC entity is to use the MAC protocol to decide when to transmit frames into the shared medium.

Introduction to the Logical Link Control (LLC) Layer

(1) The LLC layer operates over the MAC sublayer.

(2) It has two functions :
   (i) It enhances the datagram service offered by the MAC sublayer to provide some of the HDLC services.
   (ii) It is used to exchange frames between LANs that use different MAC protocols.

(3) LLC provides the following 3 HDLC services :
   (a) *Unacknowledged connectionless service*
      (i) It is non-reliable.
      (ii) It does not do error control, flow control and sequencing. These functions will have to be done at higher layers.
   (b) *Reliable Connection-oriented Service :*
      Steps :
      (i) Establish connection.
      (ii) Data transfer.
      (iii) Disconnect the connection.
      It performs error control, flow control ans sequencing.
   (c) *Acknowledged Connectionless Service :*
      It provides request (polling) and response (acknowledgment) service.

(4)

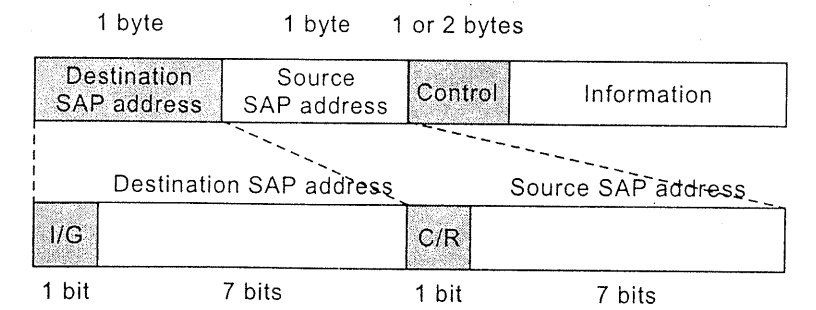


Fig. 4.2 : LLC PDU structure.

(a) A single work station may be involved with several data exchanges over the same physical connection. SAP (Service Access Point) distinguishes each of these logical connections.

(b) The upper part of the *figure 4.2* shows the LLC PDU structure. which consist of :

(i) Destination SAP address.

(ii) Source SAP address.

(iii) Control Information.

(iv) Information i.e. the packet given by the network layer.

(c) The Destination SAP address consists of I/G which specifies if the address is of an Individual or a Group.

(d) The C/R bit in the Source SAP address identifies if a frame is a command or a response frame.

(5) The *figure 4.3* shows how a packet is encapsulated while moving from the Network Layer through LLC sublayer to MAC sublayer.

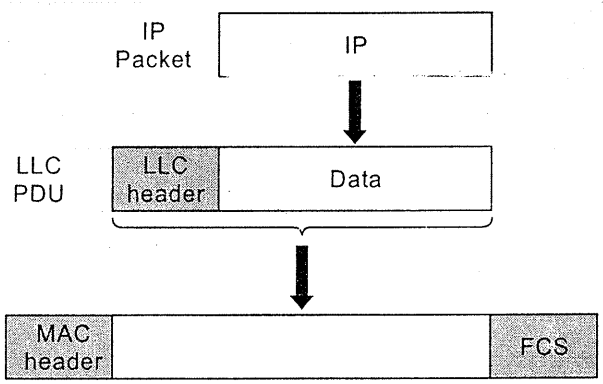


Fig. 4.3 : Packet encapsulation into a MAC frame.

4.1 Channel Allocation Problem

(1) Static Channel Allocation

- It uses Frequency Division Multiplexing.

  Frequency Division Multiplexing : If there are N users, the bandwidth of the channel is divided into N equal sized portions and each user is given one portion.
- If there are fewer than N users then one or more portions of the bandwidth will be wasted.
- We cannot have more than N users even if some users hardly use the portion assigned to them.

(2) Dynamic Channel Allocation

- *Station Model :* The model consists of Stations that generate frames for transmission. Once a frame is generated at a station, the station does nothing until the frame has been successfully transmitted.
- *Single Channel Assumption :* All stations transmit and receive frames from the same channel.
- *Collision Assumption :*
  Collision : If two stations transmit frames simultaneously, the frames overlap in time and the resulting signal is garbled. This is known as collision. All stations can detect collisions. A collided frame must be retransmitted.
- Time in a network can be either :
  (a) *Continuous Time* : Time is not divided and frame transmission can start at any time.
  (b) *Slotted Time* : Time is divided into slots and frame transmission always begins at the start of a slot. For no collision to take place a slot must contain only one frame.
- *Network can be either :*
  (a) *Carrier Sense* : A station can sense the channel to see if anyone is using it. If the channel is being used then the station will not attempt to use the channel.
  (b) *No Carrier Sense* : A station cannot sense the channel to see if anyone is using it. A station just transmits a frame as and when it is generated and later on it is notified if a collision took place or if the frame was successfully transmitted.

4.2 Multiple Access Protocols

Dec. 06 [Q. 7(3)] *Write short note on ALOHA.* **(5 M)**

Types :

[I] *Aloha :* Pure Aloha and Slotted Aloha.

[II] *CSMA Protocols :* 1-Persistent, Non-persistent and P-Persistent CSMA. CSMA/CD

[III] *Collision Free Protocols :* Bit Map Protocol and Binary Countdown.

[IV] *Wireless LAN Protocols :* MACA, MACAW.

[I] ALOHA

In Aloha systems there is No Carrier Sense.

(1) Pure Aloha

- Time is not slotted and stations can transmit whenever they want to.
- There is high possibility of collision. The colliding frames will be destroyed.
- If frames collide and get destroyed then sender waits for a random amount of time and resends the frame.

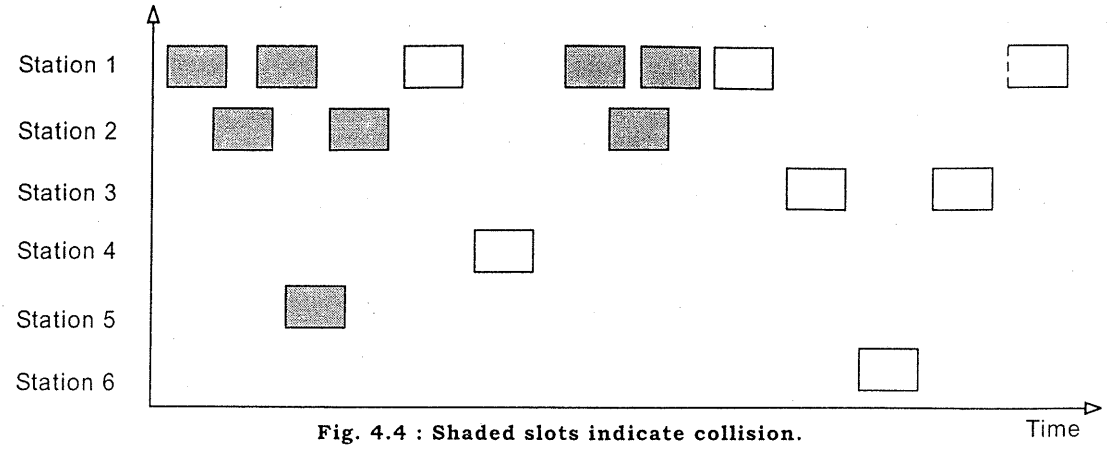


Fig. 4.4 : Shaded slots indicate collision.

(2) Slotted Aloha

- Time is divided into slots.
- Frame transmission begins at the start of a slot.
- If two or more nodes transmit in the same slot a collision is detected.

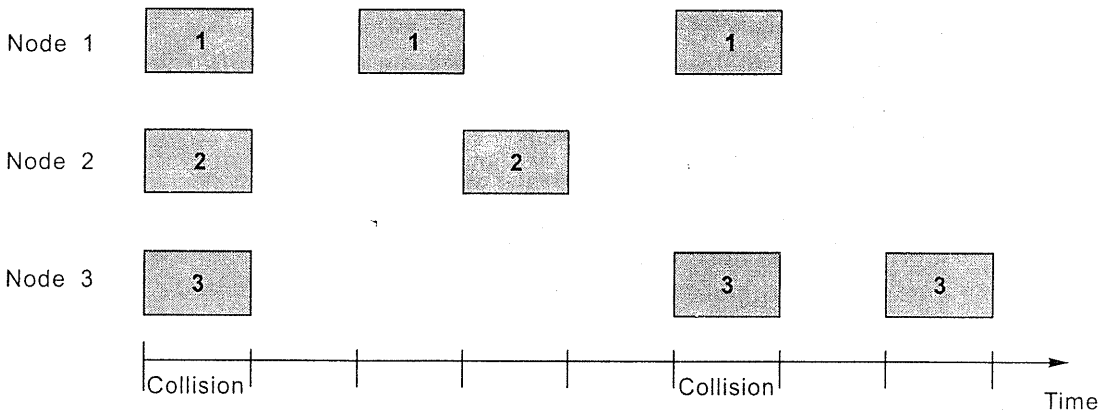
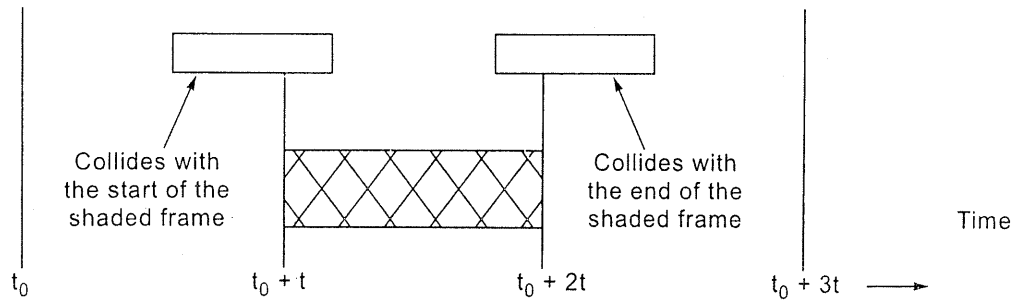


Fig. 4.5 : Random access mechanism in slotted Aloha protocol.

Efficiency of Aloha

Dec. 04 [Q. 1(b)] *Derive the formula for measuring the efficiency of the ALOHA system and explain how the efficiency is increased for slotted ALOHA.* **(10 M)**



Terms Used

(1) **Frame Time :** Time required to transmit a frame.

(2) **G :** Average number of new + old frames generated per frame time.
(Old frames are the frames which have to be retransmitted due to a collision)

(3) **S :** Average number of new frames generated per frame time.

(4) **P_0 :** Probability that the frame does not suffer collision.

(5) **VP (Vulnerable Period) :** Time for which a station should not transmit anything to avoid collision with the shaded frame.

For pure Aloha, vulnerable period = 2 times slots.

For slotted Aloha, vulnerable period = 1 time slot.

(Reason : In pure Aloha the frame can start at any time, so collision can take place in any of the 2 time slots.

In slotted Aloha, frame can start at only 2 times (t_0 and $t_0 + t$). If a frame starts at t_0, it will not collide with the shaded frame. If a frame starts at $t_0 + t$ it will collide with the shaded frame. Therefore collision can take place only in 1 time slot.)

Derivation

At low load $S \approx 0$ and $G \approx 0$

$\therefore S = G$

At high load $S = G \cdot P_0$ .....(I)

According to Poissons distribution formula

$$P_k = \frac{e^{-m} \cdot m^k}{k!} \quad \left[\begin{array}{l} m = \text{mean} \\ k = \text{random variable} \end{array}\right]$$

$P_0 = e^{-m}$ {putting k = 0}

$S = G\, e^{-m}$ .....(II) {putting $P_0 = e^{-m}$ in (I)}

For Pure Aloha

VP = 2 time slots

$\therefore m = 2G$

Putting m = 2G in (II) we get

$$S = G\, e^{-2G}$$

Differentiating w.r.t. G and equate it to zero

$$\frac{dS}{dG} = \frac{d}{dG}(G\, e^{-2G}) = 0$$

$$G\, e^{-2G}(-2) + e^{-2G}(1) = 0$$

$$e^{-2G}(1 - 2G) = 0$$

$$\therefore (1 - 2G) = 0$$

$$\therefore G = 0.5$$

$$\text{At } G = 0.5,\ S_{max} = G\,e^{-2G} = 0.5\,e^{-2 \times 0.5}$$
$$\therefore S_{max} = 0.184$$
$$\therefore \eta_{max} = S_{max} \times 100\ \% = 18.4\ \%$$

Slotted Aloha

$$VP = 1 \text{ time slot}$$
$$\therefore m = G$$

Putting m = G in (II) we get

$$S = G\,e^{-m} = G\,e^{-G}$$

Differentiating w.r.t. G and equate it to zero

$$\frac{dS}{dG} = \frac{d}{dG}(G\,e^{-G}) = 0$$
$$G\,e^{-G}(-1) + e^{-G}(1) = 0$$
$$e^{-G}(1 - G) = 0$$
$$\therefore (1 - G) = 0$$
$$\therefore G = 1$$
$$\text{At } G = 1,\ S_{max} = G\,.\,e^{-G} = 1 \times e^{-1}$$
$$\therefore S_{max} = 0.368$$
$$\therefore \eta_{max} = S_{max} \times 100\ \% = 36.8\ \%$$

[II] CSMA (Carrier Sense Multiple Access) Protocols

Carrier Sense : A station can sense the channel to see if anyone is using it. If the channel is being used then the station will not attempt to use the channel.

Types :

(a) 1-Persistent
(b) Non-persistent
(c) P-Persistent CSMA
(d) CSMA/CD

(a) **1-Persistent**

- When a station needs to send data it first listens to the channel.
- If the channel is busy the station waits till the channel becomes free.
- When the channel becomes free a station can transmit a frame.
- A collision occurs when 2 stations detect an idle channel at the same time and simultaneously send frames.
- If a collision occurs the station waits a random amount of time and starts all over again.
- It is called 1-persistent as the station will transmit with a probability of 1 when it finds the channel idle.

Drawbacks :

- *Propagation Delay* : It is possible that just after a station begins transmitting, another station becomes ready to send and it will sense the channel. If the first station's signal has not yet reached the 2^{nd} station, the 2^{nd} station will sense an idle channel and will begin sending its data. This will lead to a collision.
- Assume that Station 2 and Station 3 are waiting for Station 1 to finish its transmission. Immediately after Station 1 finishes transmitting both -Station 2 and Station 3 begin transmitting at the same time thus leading to a collision.

Advantage : Due to carrier sense property 1-persistent CSMA gives better performance than the ALOHA systems.

(b) Non-Persistent CSMA

- A station senses the channel when it wants to send data.
- If the channel is idle the station begins sending.
- However, if the channel is busy, the station does not continually sense the channel like 1-persistent CSMA. Instead, it waits a random period of time and then checks the channel again.

Disadvantage : This leads to longer delays than 1-persistent CSMA.

Advantage : This algorithm leads to better channel utilization.

(c) P-Persistent CSMA

- It is used for slotted channels.
- When a station becomes ready to send, it senses the channel.
- If channel is idle, station transmits within that slot with a probability p and defers from sending with a probability $q = 1 - p$. If $p > q$ then the station transmits else if $p < q$ then the station does not transmit and waits till the next slot and again checks if $p > q$ or $p < q$.
- This process is repeated until either the frame has been transmitted or another station has started transmitting.

(d) CSMA/CD(CSMA with Collision Detection)

Dec. 03 [Q. 6(a)] *Explain how ethernet (IEEE 802.3) works when 2 or more station want to transmit a frame ?* **(10 M)**

Dec. 04 [Q. 2(a)] *How are collisions handled by a 1-persistant CSMA protocol ? (Ans: Binary Exponential Backoff Algorithm). Give an example of a collision-free multiple access protocol and explain it in detail. (Refer section 4.2 III)* **(10 M)**

(1) Ethernet(IEEE 802.3) sends data using CSMA/CD(CSMA with Collision Detection).

(2) CSMA was an improvement over ALOHA as the channel was sensed before transmission begins.

(3) Now a further improvement on CSMA, in the form of CSMA with Collision Detection has been brought about. In this the stations abort their transmission as soon as they detect a collision.

(4) **Working :**

(a) If two stations sense the channel to be idle they begin transmitting simultaneously and cause a collision.

(b) A collision is indicated by a high voltage.

(c) Both the stations monitor the channel for a collision and stop transmitting as soon as a collision is detected.

(d) Now the stations wait for a random amount of time and check if channel is free.

(e) The process continues.

(5) *How long will it take a station to realize that a collision has taken place ?*

- Let the time for a signal to propagate between the two farthest stations be τ.
- Assume that at time t_0, one station begins transmitting.
- Let's call the most distant station as B.
- At time $\tau - \epsilon$, which is an instant before the signal arrives at B, B itself senses an idle channel and begins transmitting. A collision occurs one instant later at time τ.
- B detects the collision almost instantly and stops, but little noise burst caused by the collision does not get back to the original station until time $\tau + \tau = 2\tau$.
- In other words, in the worst case a station cannot be sure that it has seized the channel until it has transmitted for 2τ without hearing a collision.

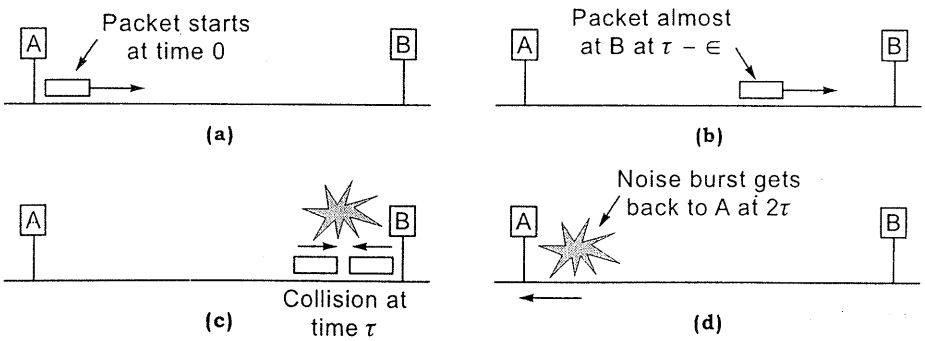


Fig. 4.6 : Collision detection can take as long as 2τ.

(6) *What happens after a collision takes place ?*

Binary Exponential Backoff Algorithm

Assume a collision takes place for the first time. Now, each station randomly waits for either 0 or 1 slot times before trying again. If the two stations pick the same waiting time, they will collide again. After the second collision, each station picks either 0, 1, 2, or 3 at random and waits for that number of slot times.

In general, after i collisions, a random number between 0 and $2^i - 1$ is chosen, and that number of slots is skipped.

However, after ten collisions have taken place, the randomization interval is frozen at a maximum of 1023 slots.

After 16 collisions, the controller reports failure back to the computer.

[III] Collision Free Protocols

(1) Bit Map Protocol

- There is a contention period in which stations have to specify whether they want to send data or not.
- If there are N stations there are N slots in the contention period.
- **Step 1 :** If a station j wants to send a frame it has to set its corresponding contention slot j, to bit 1 and if it does not want to send a frame it has to set its contention slot j, to bit 0. All stations are given a chance to specify if they want to send a frame or not.

  Step 2 : The stations with a bit 1 in their contention slot are now allowed to transmit their frames one at a time in numerical order.

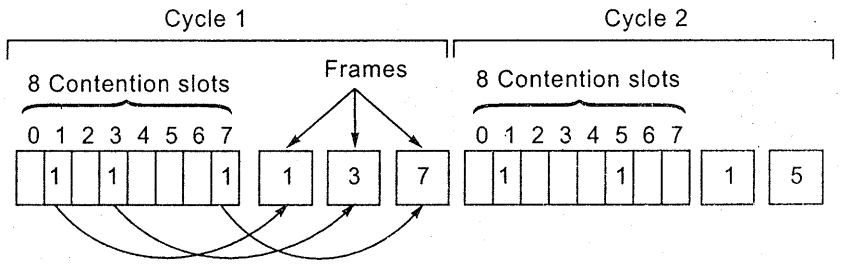


Fig. 4.7 : The basic bit-map protocol.

In the *figure 4.7* stations 1, 3, 7, have specified that they want to send a frame by putting a '1' in their contention slots. Now they are allowed to transmit their frame in order. In cycle 2, station 1, 5 want to transmit a frame and they are allowed to do so in numerical order.

(2) Binary Countdown

- In the bit map protocol there is an overhead of 1 bit per station (for the contention period) and hence it is not feasible in a system with thousands of stations.

- In binary countdown all the stations that want to transmit have to broadcast their address as a binary bit string.
- The bits of all the addresses of the stations that want to transmit are Boolean ORed together starting from the higher order bit.
- If a station with a 0 in its higher order bit sees that the answer of the OR operation is 1, it removes itself from contention.
- The process continues for lower order bits till only one station remains.

E.g. If stations 0010, 0100, 1001, and 1010 are all trying to get the channel, in the first cycle the highest order bits (i.e. the left most bits) which are 0, 0, 1, and 1, respectively are ORed together to form 1. Stations 0010 and 0100 see the 1 and know that a higher-numbered station is competing for the channel, so they give up for the current round. Stations 1001 and 1010 continue.

The next bit is 0, and both stations continue. The next bit is 1, so station 1001 gives up. The winner is station 1010 and it is allowed to transmit.

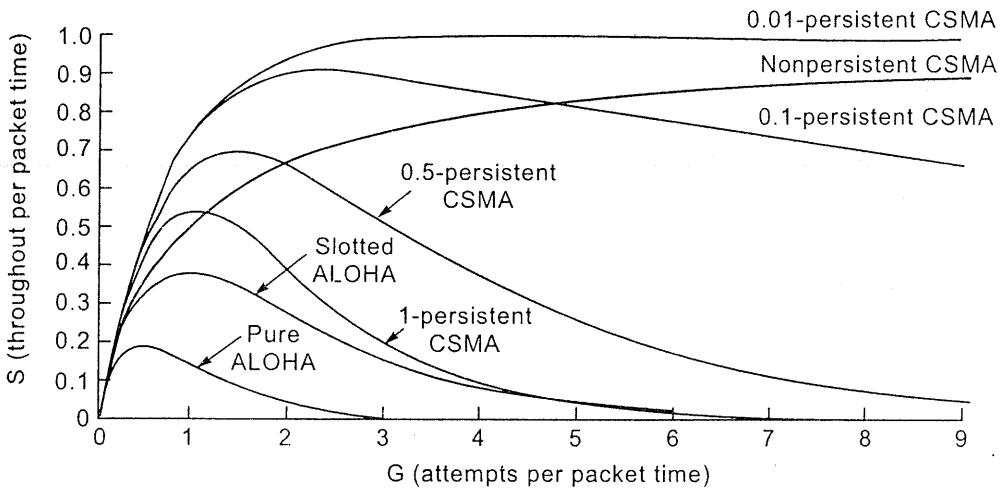


Fig. 4.8 : Comparison of the channel utilization versus load for various random access protocols.

[IV] Wireless LAN Protocol

May 04 [Q. 7(b)], May 05 [Q. 4(b)] *Describe the working of a wireless LAN protocol.*
(10 M)

Dec. 04 [Q. 4(b)] *Describe the working of a multiple-access protocol for wireless LAN's. Could CSMA be used instead ? Justify (Ans : CSMA cannot be used due to the Hidden and Exposed Station Problem.)*
(10 M)

May 06 [Q. 6(a)] *Explain Hidden Station and Exposed Station problem in wireless LAN.*
(10 M)

A system of portable computers that communicate by radio waves is called *Wireless LAN*.

A type of wireless LAN technology is the IEEE 802.11

IEEE 802.11

It has 2 modes of operation

(1) *With Base Station* : Data is transferred from one computer to another via a Base Station (Access Point). Various networks can be joined together by connecting the base stations of all these networks to a portal.

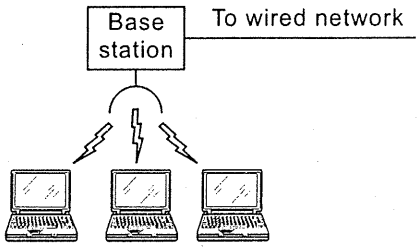


Fig. 4.9 : Wireless networking with a base station.

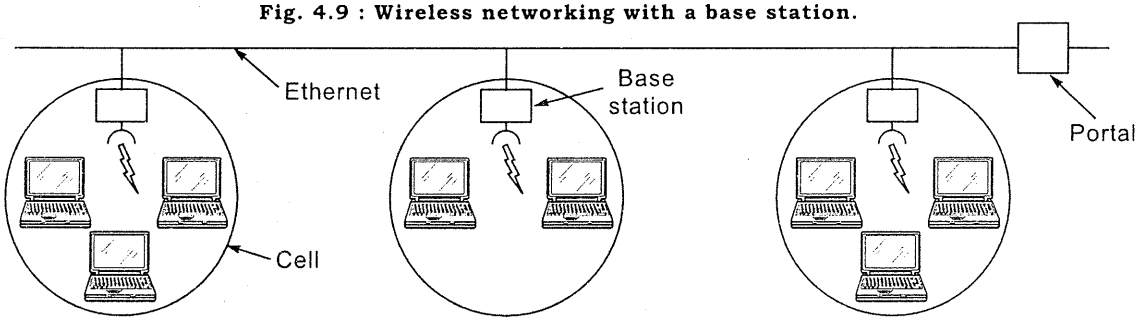


Fig. 4.10 : A multicell 802.11 network.

(2) *Without Base Station* : A computer directly sends data to another computer. In this case various networks cannot be connected to each other. It is also known as *adhoc networking*.

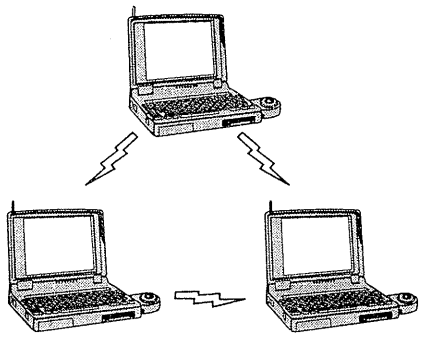
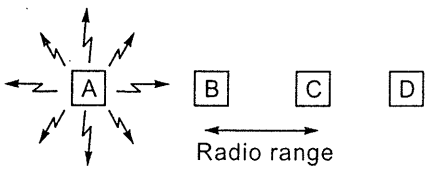


Fig. 4.11 : Adhoc networking.

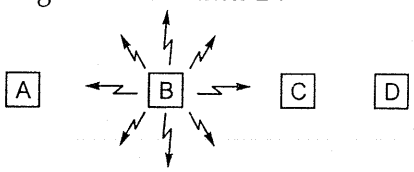
Fundamentals of Wireless LAN

- All radio transmitters have a fixed range.
- If a receiver is within the range of two active senders the resulting signal received by the receiver will be garbled.

- *Hidden Station Problem :*
  - In the diagram we assume that the range of a station is upto its adjacent stations i.e. the range of B is A and C; the range of C is B and D .
  - Assume that A is transmitting to B. Now if C wants to transmit to B it first senses the medium. As A is hidden from C (i.e. A is out of C's range); C will sense the channel as idle even though A is actually transmitting to B. According to C the channel is idle and it will start transmitting which will lead to collision.



- *Exposed Station Problem :*
  - In the diagram we assume that the range of a station is upto its adjacent stations i.e. the range of B is A and C; the range of C is B and D.
  - Assume that B is transmitting to A. If C senses the medium it will realize that a transmission is going on and falsely concludes that it cannot transmit to D.



Wireless LAN Protocols

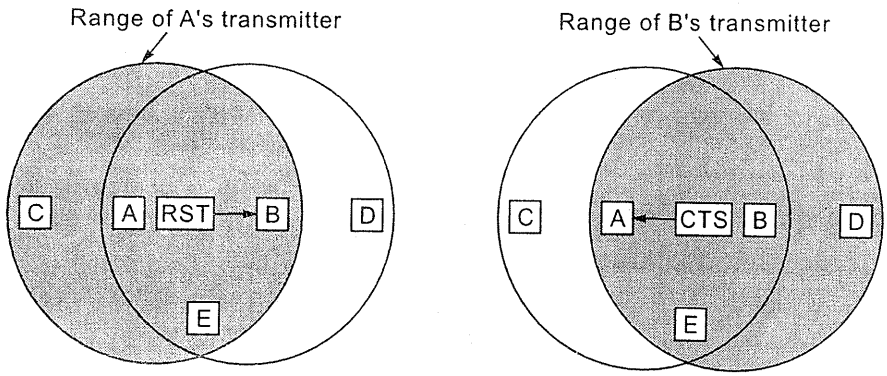
(1) MACA (2) MACAW

(1) MACA (Multiple Access with Collision Avoidance)

The idea behind MACA is :

- The Sender sends a handshake frame to receiver.
- In response the receiver sends a frame to the sender.
- All stations within the range of the receiver can detect this response transmitted from the receiver to the sender and understand that they should not transmit for the duration of the upcoming data frame.

E.g. If A wants to send a frame to B



(1) A sends a RTS(Request To Send) frame to B. The RTS contains the length of the Data frame to be transmitted.
(2) B replies with a CTS (Clear To Send) frame back to A.
(3) When A receives the CTS it begins transmission.
(4) Now consider station C which is in the range of A but out of the range of B.
C hears the RTS from A to B.
C does NOT hear the CTS from B to A as it is outside the range of B.
C can transmit while the data is being transferred from A to B as it is outside the range of B.
(5) Consider Station D which is within the range of B but outside the range of A.
D does not hear the RTS from A to B.
D hears the CTS from B to A and therefore D remains silent for the duration of the transmission from A to C.
(6) Consider Station E which is within the range of B and A.
E hears both, the RTS from A to B as well as the CTS from B to A.
As E hears the CTS from B to A, therefore E remains silent for the duration of the transmission from A to B.

(2) MACAW (MACA for Wireless)

- It is a fine tuned version of MACA.
- Its advancements over MACA are :
  (1) They introduced the concept of an ACK (acknowledgment) frame.
  (2) They introduced the feature of Carrier Sensing.
  (3) They decided to run the backoff algorithm for each data stream rather than for each station.
  (4) They made mechanisms to reduce congestion.

The protocol stack of 802.11 is given

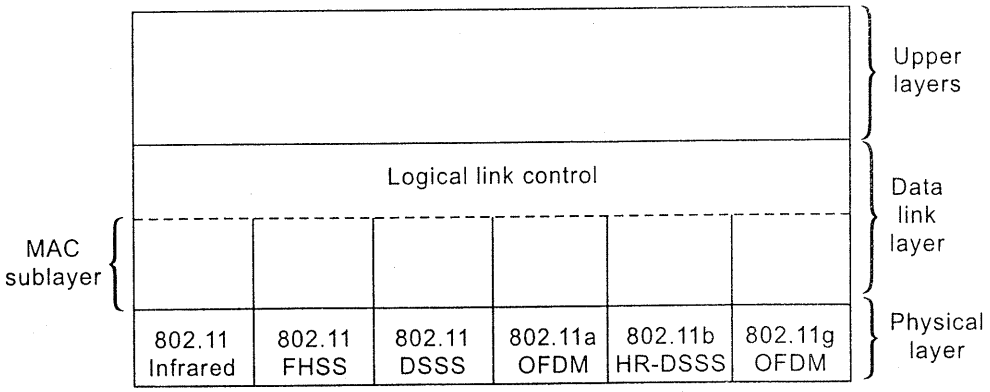


Fig.4.12 : Part of the 802.11 protocol stack.

In the figure the physical layer, MAC sublayer and LLC sublayer do their normal functions.

4.3 IEEE Standard 802.3 (Ethernet)

IEEE Standard 802.3 uses 1-persistent CSMA/CD LAN.

4.3.1 Cabling

Name	Cable	Max Segment	Nodes per Segment	Advantage
10Base5	Thick co-axial	500 m	100	Original cable but obsolete now
10Base2	Thin co-axial	185 m	30	No hub is needed
10BaseT	Twisted Pair	100 m	1024	Cheapest
10BaseF	Fiber Optic	2000 m	1024	Best between buildings

10Base5 : (10 implies 10 Mbps; Base implies Baseband transmission; 5 implies 500 m is the max segment length)

- A.k.a Thick Ethernet.
- Connections are made using Vampire Taps in which a pin is carefully forced halfway into the co-axial cable core.
- Operates at 10 Mbps.
- Uses Baseband signaling.
- Can support segments upto 500 m.
- Was used for backbones but is now obsolete.
- Uses Time Domain Reflectometry to detect cable breaks.

Time Domain Reflectometry : A pulse is sent through the cable. This pulse hits the break in the cable and an echo is reflected back. By measuring the time between the sending of the pulse and the receiving of the echo we can detect where the break has occurred.

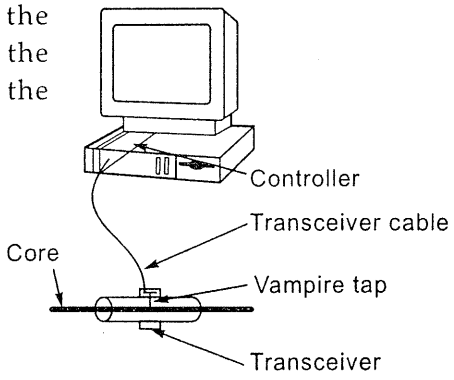


Fig. 4.13 : Ethernet cabling : 10Base5.

10Base2 :

- A.k.a
- Conne
- Opera
- Uses B
- Can su
- It is th
- Uses detect

10BaseT : (T m

- Twistec
- Operat
- Uses Ba
- Can su
- Mainte
- Hubs a

10BaseF : (F m

- Fiber op
- It is con
- Operate
- Uses Ba
- Can sup
- It has th

4.3.2 Ethernet

Bytes: 7

Prea

10Base2 :

- A.k.a Thin Ethernet
- Connections are made using T junctions.
- Operates at 10 Mbps.
- Uses Baseband signaling.
- Can support segments upto 185 ≈ 200 m.
- It is the cheapest.
- Uses Time Domain Reflectometry to detect cable breaks.

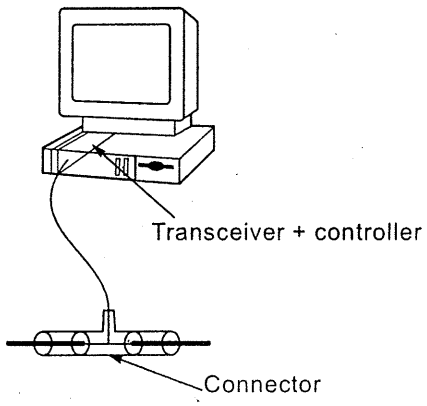


Fig. 4.14 : Ethernet cabling : 10Base2.

10BaseT : (T means twisted pair cables are used)

- Twisted pair cable is used.
- Operates at 10 Mbps.
- Uses Baseband signaling.
- Can support segments upto 100 m.
- Maintenance is easy.
- Hubs are used.

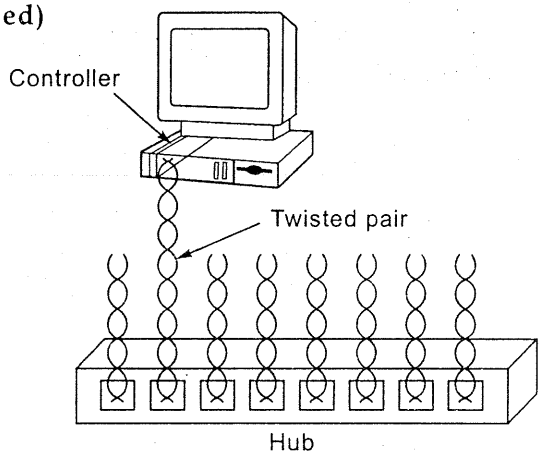
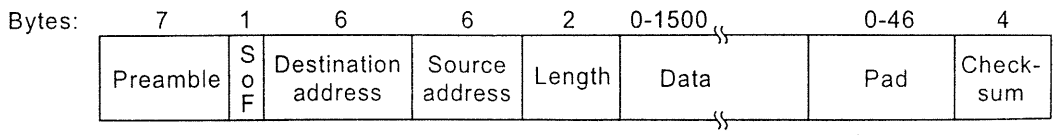


Fig. 4.15 : Ethernet cabling : 10BaseT.

10BaseF : (F means fiber optic cables are used)

- Fiber optic cable is used.
- It is comparatively expensive.
- Operates at 10 Mbps.
- Uses Baseband signaling.
- Can support segments upto 2000 m.
- It has the best noise immunity.

4.3.2 Ethernet Frame Structure



It consists of 8 fields

(1) Preamble

- Size = 7-bytes.
- It contains 56-bits of alternating 0's and 1's that tells the receiver that a frame is coming.

(2) Start of Frame Delimiter

- Size = 1-byte
- It contains 10101011 bit pattern.
- The last 2-bits "11" tell the receiver that the next field is the destination address.

(3) Destination Address

- Size = 6-bytes
- It contains physical address of the destination.
- Types
  a) Unicast : Only 1 destination.
  b) Multicast : Many destinations.
  c) Broadcast : All the stations in the network are destinations. All bits are 1.

(4) Source Address

- Size = 6-bytes.
- It contains physical address of the source machine.

(5) Length Field

- Size= 2-bytes.
- Frames must be at least 64-bytes from Destination address to checksum.
- Frames with fewer than 64-bytes are padded to 64-bytes with the PAD field.

(6) Data Field

- Size= 0 to 1500-bytes.

(7) Pad Field

- Size = 0 to 46-bytes.
- Used for padding.

(8) CRC

- Size = 4-bytes.
- It maintains the CRC code.

4.3.3 Propa

May 06 [Q.

The propaga
that mediu
maximum pc
In the follo
kilometers p

From these v
pair, which h
million bits
network.
The maximur
the maximum
divided by 19
one end of th
may occur.

4.3.4 Fast Et

(1) Provide e
(2) The fram
(3) To increa
decreased
(4) For MAC
bytes or r
(5) In fast e
twisted p
(6) It uses Ba

4.3.3 Propagation Delay in Ethernet

May 06 [Q. 5(a)] *Explain the significance of propagation delay in Ethernet.* **(10 M)**

: that a frame

The propagation speed of a medium refers to the speed that the data travels through that medium. Propagation delays differ between mediums, which affect the maximum possible length of the Ethernet topology running on that medium.

In the following table, c refers to the speed of light in a vacuum, or 300,000 kilometers per second.

e destination

Medium	Propagation	Speed
Thick Coax	.77 c	231,000 km/sec
Thin Coax	.65 c	195,000 km/sec
Twisted Pair	.59 c	177,000 km/sec
Fiber	.66 c	198,000 km/sec

From these values, the size of a bit on 10BaseT can be calculated. 10BaseT is twisted pair, which has a propagation delay of 177,000 km/sec. 177,000 km/sec divided by 10 million bits per second is 17.7 meters, or the size of a single bit on a 10BaseT network.

All bits are 1.

The maximum propagation delay through the network can be calculated by dividing the maximum length by the speed. For 10Base2 thin coax network, this is 185 meters divided by 195,000 km/sec, or 950 nanoseconds. If the actual propagation delay from one end of the network to the other is greater than 950 nanoseconds, late collisions may occur.

ecksum.

e PAD field.

4.3.4 Fast Ethernet (100 Mbps LAN)

(1) Provide ethernet LANs operating at 100 Mbps.

(2) The frame format, interfaces and procedures are the same as 10 Mbps Ethernet.

(3) To increase speed from 10 Mbps to 100 Mbps the frame transmission time is decreased by a factor of 10.

(4) For MAC protocol to work properly either minimum frame size must be made 640 bytes or maximum length between two stations must be made 250 meters.

(5) In fast ethernet the physical layers are entirely based on hub topology using twisted pair and optical fibers. Co-axial cables are not used.

(6) It uses Base band signaling.

(7) In *Fast Ethernet* we have two types of hubs.

(a) *Shared Hub* : Uses CSMA-CD for collisions. Only one station can transmit at a time.

(b) *Switched Hub* : It does not use CSMA-CD. Instead it buffers incoming frames and then performs switching and multiplexing. All stations can transmit at the same time.

(8) **Cabling :**

	100BaseT4	100BaseT	100BaseFX
Medium	Twisted pair category 3 UTP four pairs	Twisted pair category 5 UTP two pairs	Optical fiber multimode two strands
Maximum segment length	100 m	100 m	2 km
Signaling speed	25 MHz	125 MHz	—
Topology	Star	Star	Star

Table 4.1 : IEEE 802.3 Fast Ethernet medium alternatives.

4.3.5 Gigabit Ethernet

(1) Provide ethernet LANs operating at 1000 Mbps.

(2) Frame format, interfaces and procedures are the same as 10 Mbps Ethernet.

(3) Frame size is increased to 512 bytes.

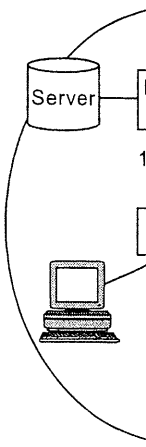
Reason : If we keep the frame size as 64 bytes, the transmission will be completed before the sender senses a collision. Thus frame size must be increased.

(4) *Cabling* :

	1000BaseSX	1000BaseLX	1000BaseCX	1000BaseT
Medium	Optical fiber multimode two strands	Optical fiber single mode two strands	Shielded copper cable	Twisted pair category 5 UTP
Maximum segment length	550 m	5 km	25 m	100 m
Topology	Star	Star	Star	Star

Table 4.2 : IEEE 802.3 Gigabit Ethernet medium alternatives.

5. *Campus N*



De

As shown

(a) Fast

(b) Gigal ether

4.4 Manche

Dec. 06 [Q. 1

Binary Encoc

- A 0 b
- None ambi
- Ambi idle s

5. *Campus Network* :

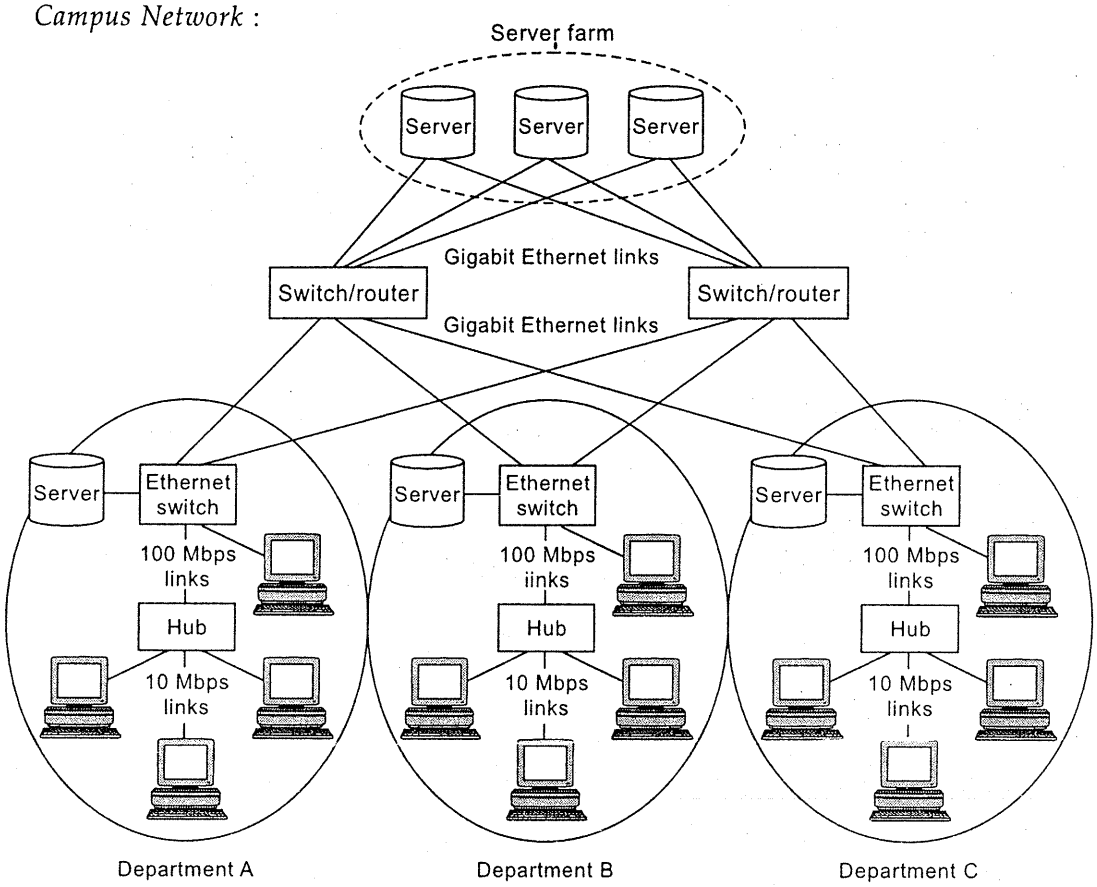


Fig. 4.16 : Deployment of ethernet in a campus network.

As shown in the *figure 4.16.*

(a) Fast ethernet is used within a department.

(b) Gigabit Ethernet is used to connect the departments. Thus, in general, gigabit ethernet is used for backbones.

4.4 Manchester Encoding

Dec. 06 [Q. 1(b)] *Draw NRZ and Manchester encoding signal transitions for 1 0 0 0 0 1 0 1 1 1 1.* **(5 M)**

Binary Encoding : (a.k.a NRZ : Non Return to Zero Encoding)

- A 0 bit is represented as 0 Volts and a 1 bit is represented as 5 volts.
- None of the versions of Ethernet uses binary encoding because it leads to ambiguities.
- Ambiguities are caused because stations cannot tell the difference between an idle sender (0 volts) and a 0 bit (0 volts).

What is needed is a way for receivers to unambiguously determine the start, end, or middle of each bit without reference to an external clock.

Two such approaches are:

(a) Manchester Encoding (b) Differential Manchester Encoding.

(a) Manchester Encoding

- In Manchester encoding, each bit period is divided into two equal intervals. A binary 1 bit is sent by having the voltage set high during the first interval and low in the second one. A binary 0 is just the reverse : first low and then high.
- This scheme ensures that every bit period has a transition in the middle, making it easy for the receiver to synchronize with the sender.
- All Ethernet systems use Manchester encoding due to its simplicity.
- A disadvantage of Manchester encoding is that it requires twice as much bandwidth as straight binary encoding because the pulses are half the width. For example, to send data at 10 Mbps, the signal has to change 20 million times/sec.

(b) Differential Manchester Encoding

- It is a variation of basic Manchester encoding. In it, a 1 bit is indicated by the absence of a transition at the start of the interval. A 0 bit is indicated by the presence of a transition at the start of the interval. In both cases, there is a transition in the middle as well.
- The differential scheme requires more complex equipment but offers better noise immunity.
- The high signal is + 0.85 volts and the low signal is - 0.85 volts, giving a DC value of 0 volts.
- Ethernet uses Manchester encoding, but other LANs (e.g., the 802.5 token ring) use Differential Manchester encoding .

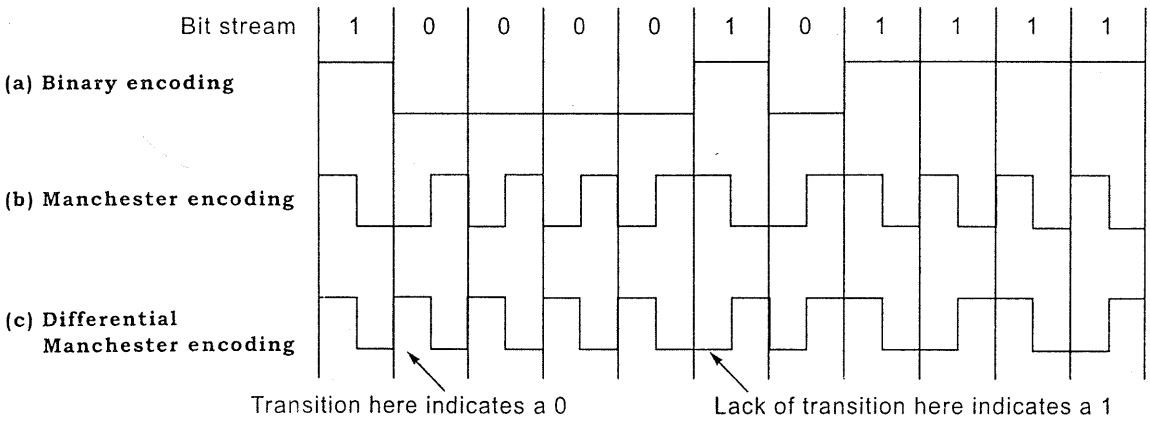


Fig. 4.17

May 05 [Q.

(a) Mancheste

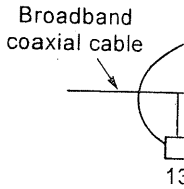
(b) NRZ encodi

4.5 IEEE 80

Dec. 03 [Q.

May 05 [Q. 1

Dec. 05 [Q. 6



Attribute
Working

May 05 [Q. 5(c)] *Draw NRZ and Manchester Encoding signal transitions for 1 0 0 0 0 1 0 1 1 1 1 data stream.* **(5 M)**

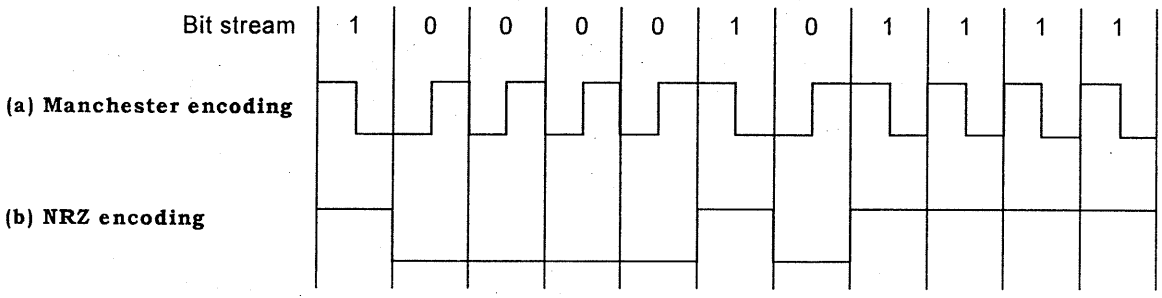


Fig. 4.18

4.5 IEEE 802.3, 802.4, 802.5

Dec. 03 [Q. 7(b)], Dec. 06 [Q. 2(b)] *Compare IEEE 802.3, 802.4, 802.5.* **(10 M)**

May 05 [Q. 1(b)] *Compare 802.3 and 802.5.* **(6 M)**

Dec. 05 [Q. 6(b)] *Compare IEEE802.4 and 802.5.* **(4 M)**

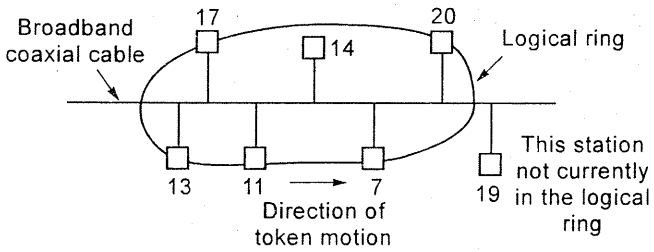


Fig. 4.19 : A token bus.

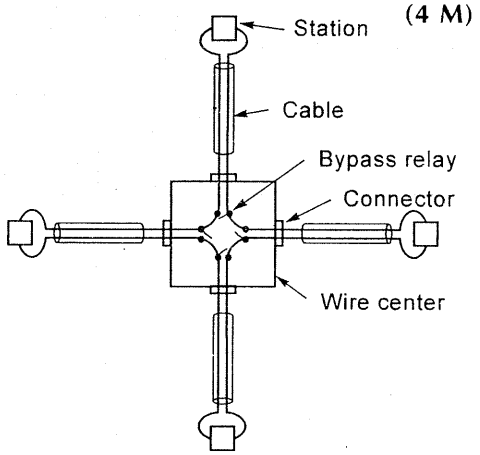


Fig. 4.20 : Four stations connected via a wire center.

Attribute	802.3(Ethernet)	802.4(Token Bus)	802.5(Token Ring)
Working	CSMA/CD	Nodes connected to the bus are arranged in a logical ring. Only a node with a token may send a frame. The token is passed along the logical ring.	When all stations are idle a token circulates around the ring. Whenever a station wants to transmit a frame it takes the token. As there is only one token, only one station will be able to take it and transmit. As the bits that the sender station put on the ring return, the sender removes them and replaces the token onto the ring.

Attribute	802.3(Ethernet)	802.4(Token Bus)	802.5(Token Ring)
Structure	Physically-BUS, TREE Logically-BUS	Physically-BUS Logically-RING	Physically-RING Logically-RING
Protocol	1-Persistent CSMA/CD	Token Bus	Token Ring
Size of data field in bytes	0 to 1500	0 to 8182	No limit
Data rate	10 Mbps	10 Mbps	4 Mbps
Simplicity	Simple	Complex	Complex
Collisions	May occur	Collision free	Collision free
Minimum frame required	64 bytes	No restriction on minimum frame size	No restriction on minimum frame size
Priorities given to frames?	No	Yes	Yes
Pad bytes used ?	Yes	No	No

4.6 Bluetooth

Dec. 05 [Q. 7], Dec. 06 [Q. 7(4)] *Write short note on Bluetooth.* **(10 M), (5 M)**

May 06 [Q. 2(b)], May 07 [Q. 3(b)] *List Bluetooth features. Explain Network Formation Process.* **(10 M)**

- The Ericsson Mobile company became interested in connecting its mobile phones with other devices without cables. Together with 4 other companies, which included IBM, Intel, Nokia and Toshiba, it formed a Special Interest Group to develop a wireless standard for interconnecting devices using short range low power inexpensive wireless radios.
- Bluetooth is a wireless, short range, low power and inexpensive radio solution that allows small devices to communicate with each other.
- The technology is being extended to work with desktop, so that printers, scanners etc. can be connected to the PCs without cables.
- This concept has been termed as PAN (Personal Area Network).
- Specifications :
  - Radio Frequency Band : 2.4 GHz
  - Data Rate : 20 Mbps
  - Range = 10 m

Architecture :

(1) When wo
(2) Bluetooth
(3) Communi slave com
(4) Owing to referred t
(5) Upto 7 sla inactive (*p*
(6) Multiple p
(7) A collectio

Bluetooth App

(1) *Synchroniz* and updat
(2) Mice and interventi touch.
(3) *Internet Br* or Portabl through a
(4) *All in One*
  - In the
  - At hom
  - When charge
(5) The cordl wirelessly

Architecture :

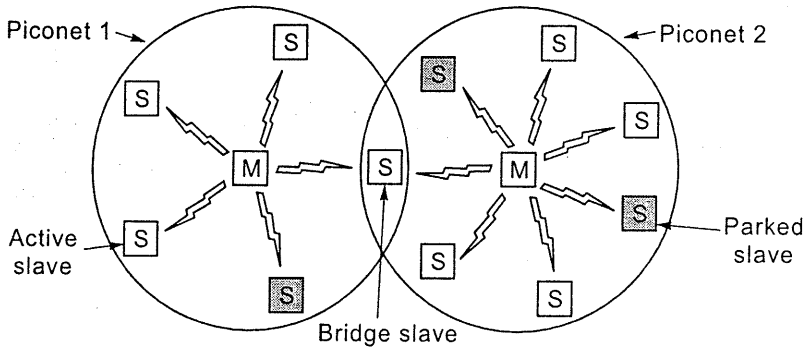


Fig. 4.21 : Two piconet's can be connected to form a scatternet.

(1) When working in a Bluetooth network, the device is either a Master or a Slave.

(2) Bluetooth uses a centralized TDM system with the master controlling the clock.

(3) Communication can only take place between master and slave. Direct slave to slave communication is not possible.

(4) Owing to the small area (10 m) in which Bluetooth operates the networks are referred to as *piconets*.

(5) Upto 7 slaves can be *active* in a piconet. There can be many other slaves that are inactive (*parked*).

(6) Multiple piconets are connected via a bridge node.

(7) A collection of piconets is called a *scatternet*.

Bluetooth Applications :

(1) *Synchronization :* Bluetooth enabled PDA, PC and laptop all talk to each other and update their respective files to the most recent ones.

(2) Mice and keyboards will identify themselves to the computer without intervention, or could also be used to command TVs, videos or hi-fis at the touch.

(3) *Internet Bridge :* With Bluetooth, portable computing devices (i.e. Notebook, PDA or Portable PC) can surf the Internet anywhere, through a mobile phone or through a wired connection.

(4) *All in One Phone :* The same phone can be used everywhere.

- In the office, the phone works as an intercom with no telephony charges.
- At home it functions as a cordless phone with fixed line charges.
- When the user is on the move it functions as a cellular phone with cellular charge.

(5) The cordless headset. Connect a headset to the mobile phone or mobile PC wirelessly to free your hand for more important tasks.

(6) By installing a Bluetooth network in your office you can do away with the complex and tedious task of networking between the computing devices, yet have the power of connected devices. No longer would you be bound to fixed locations where you can connect to the network.

(7) Bluetooth provides a more general file transfer facility.

(8) In meetings and conferences, the user can share information instantly with other participants.

(9) Anybody will be able to access a gas station, local supermarket or department store consumer information through his mobile phone or PDA, when she/he is near to the premises.

(10) Bluetooth defines a client-server relationship for object exchange.

Frame Structure :

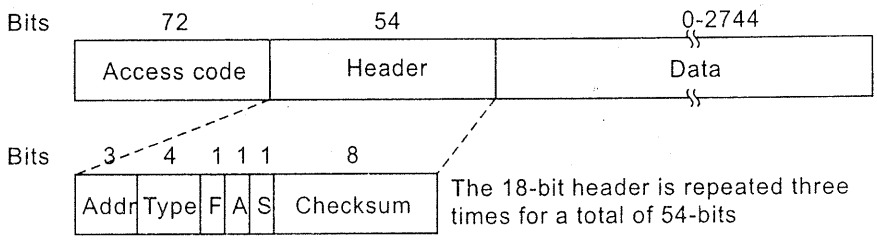


Fig. 4.22 : A typical Bluetooth data frame.

- 72-bit access code identifies the master node.
- 54-bit header includes three repeats of an 18-bit header.
- 3-bit address identifies one of the *eight* nodes. (1 master + 7 slaves)
- 4-bit type indicates :
  - Frame type (ACL, SCO, poll, or null)
  - Type of error correction
  - How many slots long the frame is.
- Flow bit is set when the slave buffer is full.
- Acknowledge bit piggybacks the ack of a receiving frame.
- Sequence bit is used for retransmission (stop-and-wait protocol).
- 8-bit check-sum.
- The header repeats three times. If all headers are the same, the frame is accepted; if differ, majority wins.

4.7 Data Link La

4.7.1 LAN Bridges

(1) A bridge is a d
an *extended LA*

(2) LANs may ha
reformatting th

(3) Function of a B

(i) Read all th

(ii) If there is
accept the

(iii) The same p

(iv) As *filtering*
at Data-Lin

(v) This gives
between a
repeater.
perform fi
checks the
the frame
where the
be forward
simply fo
frame on
lines.

4.7 Data Link Layer Switching

4.7.1 LAN Bridges

(1) A bridge is a device that connects two or more similar or dissimilar LANs to create an *extended LAN.*

(2) LANs may have different MAC frame structure, thus bridges handle this by reformatting the frames.

(3) Function of a Bridge.

(i) Read all the frames being transmitted on LAN 1.

(ii) If there is a frame whose MAC address is addressed to a station on LAN2 accept the frame. and retransmit the frame on LAN 2. This is called *filtering.*

(iii) The same process is carried out for frames on LAN2.

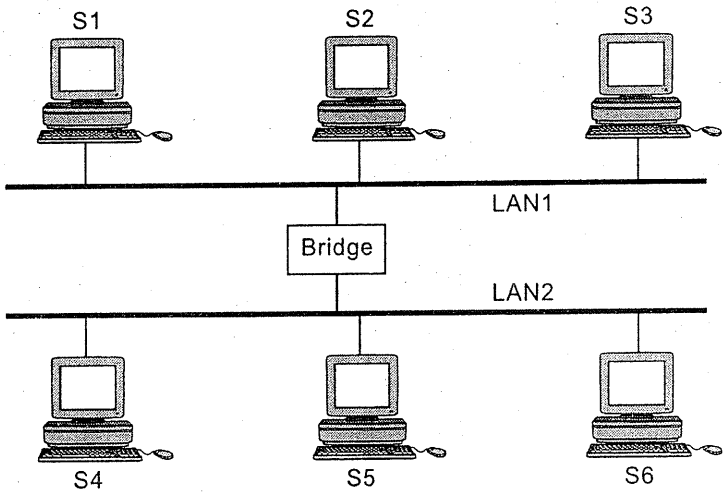


Fig. 4.23 : A bridged LAN.

(iv) As *filtering* is done based on the MAC addresses, we say that Bridges operate at Data-Link Layer.

(v) This gives the difference between a bridge and a repeater. A bridge can perform filtering i.e. it checks the destination of the frame and decides where the frame should be forwarded. A repeater simply forwards the frame on *all* its *outgoing lines.*

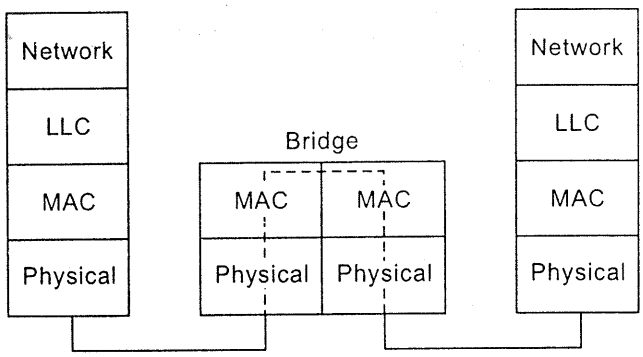


Fig. 4.24 : Interconnection by a bridge.

4.7.2 Transparent Bridges

Expected Question : *Explain the spanning tree algorithm with reference to frame forwarding, address learning and loop resolution.* **(10 M)**

(1) A transparent bridge is a bridge which the stations are completely unaware of. Thus stations need not be reconfigured if the bridge is added/removed from the system.

(2) The three criteria that must be satisfied by transparent bridges are.

(a) *Forwarding* : Frames must properly be forwarded from one station to another.

(b) *Learning* : The forwarding table is updated (learned) by monitoring frame.

(c) *Loops* must be avoided.

(3) Forwarding : As explained in section 4.7.1.

(4) Learning

(a) Static Forwarding Tables :

(i) Used initially.

(ii) Administrator would make forwarding entries manually during bridge setup.

(iii) *Disadvantage*

- If station was added/removed the table had to be modified manually.
- If stations MAC address changed (by changing the network card of the station) the table entry had to be changed manually.

(b) Dynamic Forwarding Table :

(i) Overcomes drawbacks of static forwarding table.

(ii) In this the table entries are made/modified by *learning* from the frame movements.

(iii) To *learn* the bridge uses both the *destination* and the *source address*.

- The *destination address* is used to make forwarding decision based on the table entry.
- The *source address* is used for adding/updating the table entry in the bridge.

(iv) Exam

L

L

Ac

(a

(a) Initially A
bridge forv
frame floods
learns that
table.

(b) Now if E w
connected
by inspecti
connected t

(c) When B w
followed ar

(5) **Loop Problem**

(a) *Redundant*

(b) *Transparent*
system adn
does not w

(c) The *loop pro*

reference to frame
(10 M)

letely unaware of.
removed from the

re.

ation to another.

itoring frame.

lly during bridge

ified manually.

twork card of the

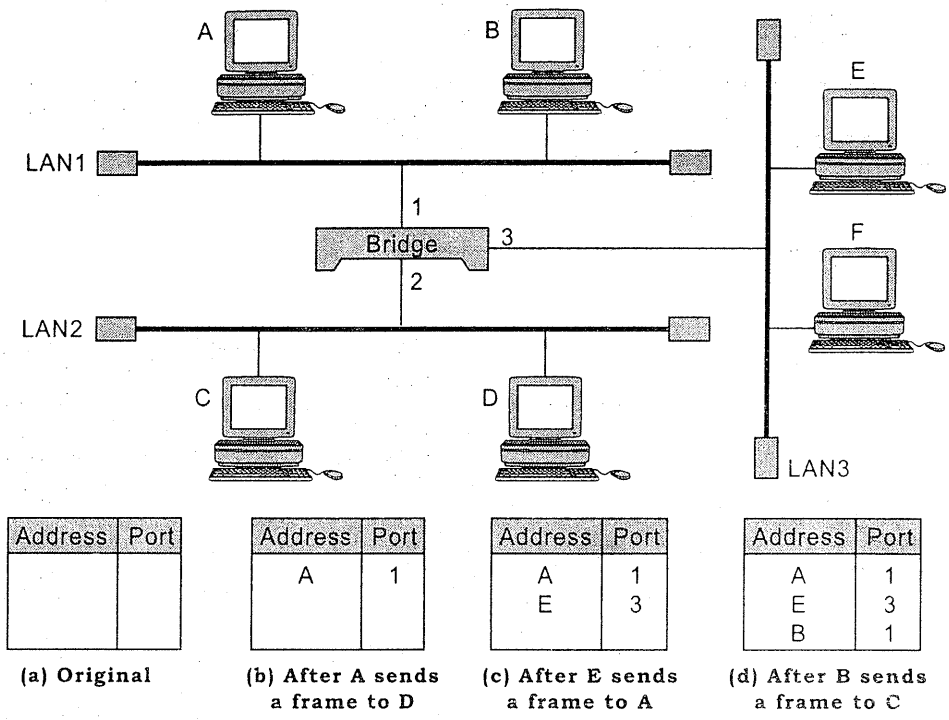
from the frame

e address.

ecision based on

ble entry in the

(iv) Example :



Address	Port

(a) Original

Address	Port
A	1

(b) After A sends a frame to D

Address	Port
A	1
E	3

(c) After E sends a frame to A

Address	Port
A	1
E	3
B	1

(d) After B sends a frame to C

Fig. 4.25 : A learning bridge and the process of learning.

(a) Initially A wants to send a frame to D. Initially the bridge table is empty. The bridge forwards the frame coming from A on port 1 to all the 3 ports i.e. the *frame floods the network*. However by looking at the source address the bridge *learns* that A is on the LAN connected to port 1 and makes on entry in its table.

(b) Now if E wants to send a frame to A. The bridge knows that A lies on the LAN connected to port 1. Therefore the frame is forwarded only on port 1. Further by inspecting the *Source address* of the frame it *learns* that E lies on the LAN connected to port 3 and accordingly makes an entry in its table.

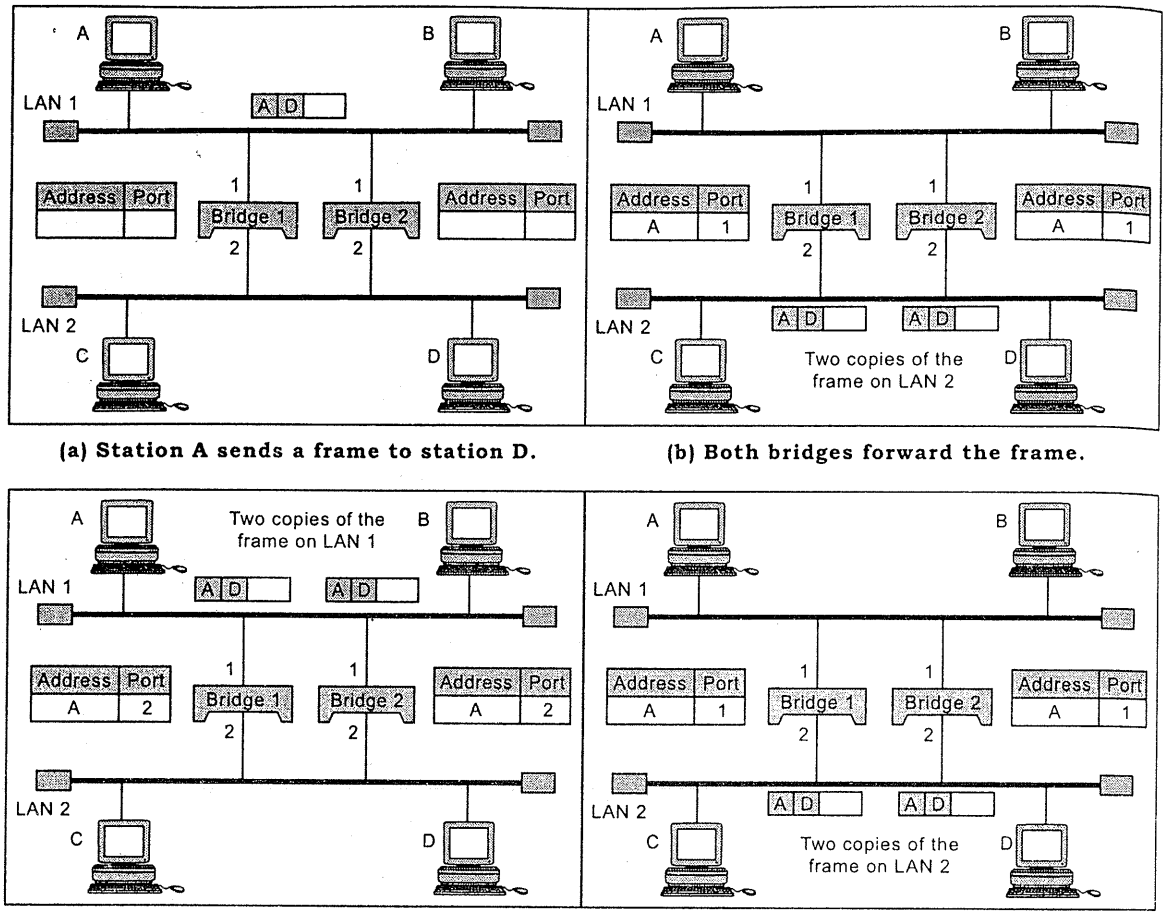
(c) When B wants to send a frame to C, the same procedure as point (a) is followed and an entry for B is made in the table.

(5) **Loop Problem**

(a) *Redundant Bridges* : More than one bridge between the same pair of LANs.

(b) *Transparent bridges* work properly if there are no redundant bridges, but system administrators like to have redundant bridges so that if one bridge does not work the other bridge can be used to transfer data.

(c) The *loop problem* due to redundant bridges.



(a) Station A sends a frame to station D.

(b) Both bridges forward the frame.

(c) Both bridges forward the frame.

(d) Both bridges forward the frame.

Fig. 4.26 : Loop problem in a learning bridge.

(i) Bridge 1 and Bridge 2 are redundant bridges between LAN 1 and LAN 2.

(ii) If A wants to send frame to D, as both the bridge tables are empty, both forward the frame (i.e. floods the network) and also make an entry for A in their bridge tables.

(iii) Now there are 2 copies of the frame in LAN 2. The copy sent out by Bridge 1 is received at Bridge 2 in LAN 2. As Bridge 2 does not have information about the *destination address* D it forwards the frame (i.e. floods the network) and the frame again reaches LAN 1.

(iv) Similarly the frame arriving at LAN 2 via Bridge 2 is forwarded back to LAN 1 via Bridge 1.

(v) Now in LAN 1 we again have two copies of the frame and the steps (iii) and (iv) are repeated.

(vi) This process is repeated continuously and it is called the loop problem. To solve this problem the **Spanning Tree** approach is used.

(6) Spanning Tre

(a) A *spannin*

(b) This mean loop exists

(c) *Figure 4.2* *4.27(b)* sho weight de delay in t from Bridg

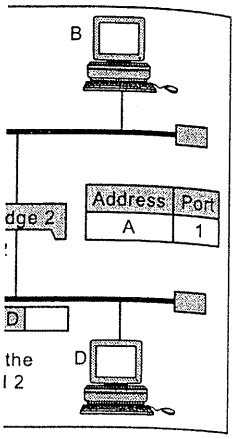
(b

Fig. 4.2

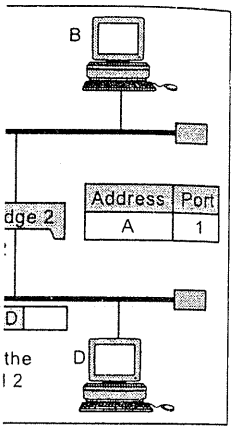
(d) Process

(i) Every its ID ID. Th we ass

(ii) The sp Bridge two pa choose



rd the frame.



rd the frame.

AN 1 and LAN 2.
s are empty, both
ke an entry for A

copy sent out by
2 does not have
ls the frame (i.e.

orwarded back to

and the steps (iii)

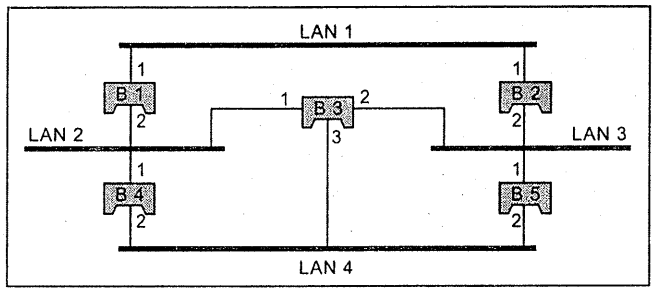
he loop problem.
d.

(6) **Spanning Tree**

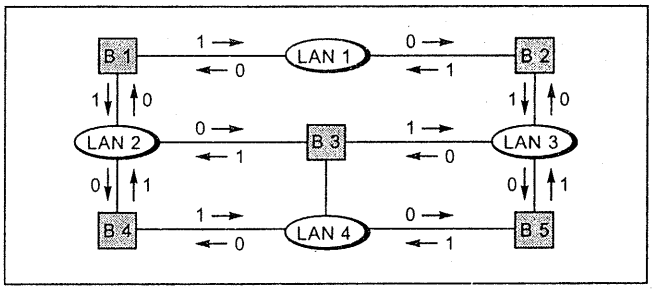
(a) A *spanning tree* is a *graph* having no loops.

(b) This means that a LAN can reach another LAN through one path only (i.e. no loop exists).

(c) *Figure 4.27(a)* and *(b)* shows a system with 4 LANs and 5 bridges. The *figure 4.27(b)* shows the graph representation. Each connecting arc has an associated weight depending on the delay, bandwidth or the number of hops. We select delay in terms of number of hops. By convention we see that hop count is '1' from Bridge to LAN and '0' from LAN to Bridge.



(a) Actual system.



(b) Graph representation with cost assigned to each arc.

Fig. 4.27 : A system of connected LANs and its graph representation.

(d) Process

(i) Every Bridge has a built-in ID, which is unique. Each bridge broadcasts its ID so that all the bridges can find out which bridge has the smallest ID. The bridge with the smallest ID is called the *Root Bridge*. In the figure we assume that B1 is the *Root Bridge*.

(ii) The spanning three algorithm finds the shortest path from the Root Bridge to all the other Bridges depending on the weights of each arc. If two paths have the same total weight the system administrator can choose any one. The shortest paths can be seen in *figure 4.28*.

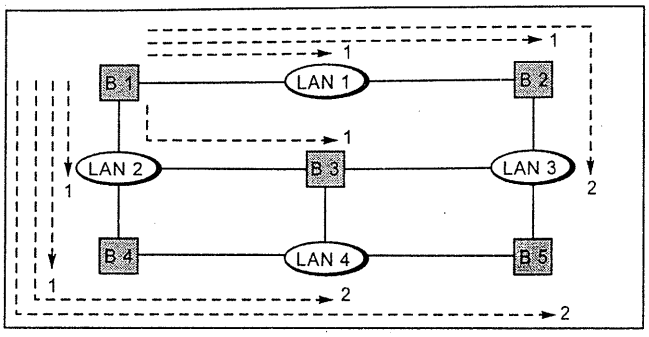


Fig. 4.28 : Finding the shortest paths in a system of bridges.

(iii) The tree consisting only of the shortest paths from the Root Bridge to all other bridges is called *spanning tree* or *shortest tree.*

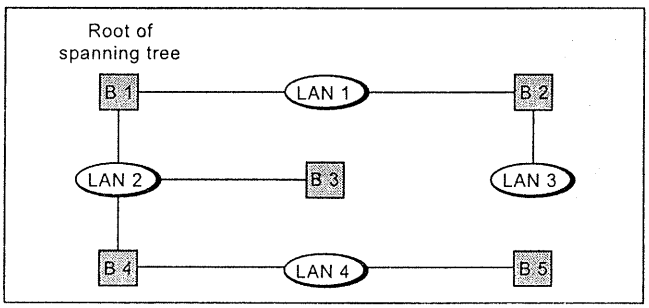
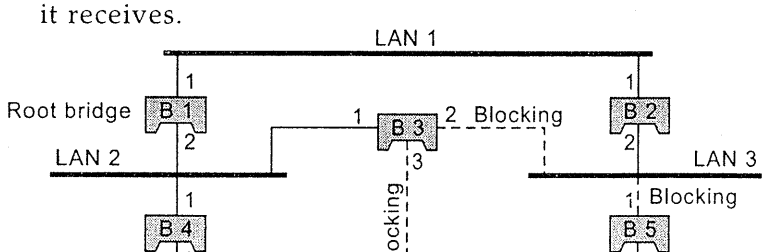


Fig. 4.29 : Finding the spanning tree in a system of bridges.

(iv) Based on the spanning tree we have *Forwarding Ports* or *Blocking Ports* in the network.

- *Forwarding Ports* : Ports on which the bridge **can** forward the frames which it receives.
- *Blocking Ports* : Ports on which the bridge **cannot** forward frames which it receives.



Ports 2 and 3 of bridge B 3 are blocking ports (no frame is sent out of these ports).
Port 1 of bridge B 5 is also a blocking port (no frame is sent out of this port).

Fig. 4.30 : Forwarding and blocking ports after using spanning tree algorithm.

(v) Only a single path exists between any two LANs in the above network and there is no possibility of loops.

(vi) Each b
algorith
betwee
of brid

4.7.3 Source Routi

(1) It is another wa

(2) **Procedure** :

(a) In such a s
station but a
the destinati

(b) This means
with the adc

(c) The source
with the des

(d) Source Rout
today.

(3) Thus we see th
forwarding and
Routing these de
the destination s

(4) The frame form
source routing is
by the *figure 4.31*

4.7.4 Bridges Conne

Bridges should b
LAN and Token Rings

(a) **Frame Format** : E

(b) **Maximum Data S**
don't have any
they must discar

(c) **Data Rate** : Eth
Mbps data rate. T

(d) **Bit Order** : Son
least significant b

(vi) Each bridge has a software package which carries out the *spanning tree algorithm* dynamically. BPDUs (Bridge Protocol Data Units) are sent between bridges to update the spanning tree in case of addition/removal of bridges.

4.7.3 Source Routing Bridges

(1) It is another way to prevent loops in a system with redundant bridges.

(2) **Procedure :**

(a) In such a system the sending station not only determines the destination station but also all the intermediate bridges that the frame must visit to reach the destination.

(b) This means that the frame contains the source and destination address along with the addresses of the intermediate bridges.

(c) The source gets the addresses of these bridges by exchanging *special frames* with the destination before sending the actual data frame.

(d) Source Routing Bridges were used in Token Ring LANs but are not common today.

(3) Thus we see that in Transparent Bridges decisions regarding filtering frames, forwarding and blocking are performed by the bridges itself. Whereas in Source Routing these decisions are performed at the source station with a little help from the destination station.

(4) The frame format for source routing is given by the *figure 4.31.*

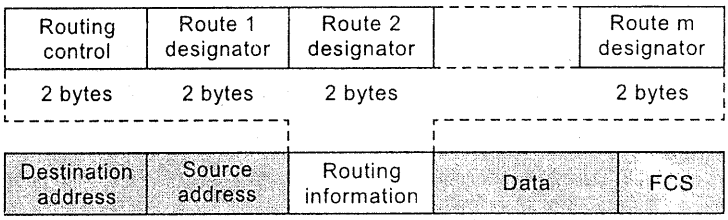


Fig. 4.31 : Frame format for source routing.

4.7.4 Bridges Connecting Different LANs or Mixed Media Bridges

Bridges should be able to connect different LANs such as Ethernet LAN, Wireless LAN and Token Rings. The issues to be resolved are :

(a) **Frame Format :** Each LAN uses a different frame format.

(b) **Maximum Data Size :** Ethernet allows frames of 1500 bytes whereas Token Ring's don't have any limit. As bridges *cannot perform* fragmentation and re-assembly they must discard frames which are too large for the destination system.

(c) **Data Rate :** Ethernet has a 10 Mbps data rate whereas wireless LAN has a 1 Mbps data rate. The bridge performs *buffering* to compensate for this difference.

(d) **Bit Order :** Some LANs send most significant bit first whereas others send the least significant bit first. Thus bits must be reordered by the bridges.

(e) **Security** : Wireless LANs use encryption whereas Ethernet does not use any security measure. Thus a bridge receiving an encrypted frame from a Wireless LAN must decrypt it before sending it to an Ethernet LAN.

(f) **Multimedia Support** : Some LANs support *Multimedia* whereas others do not. The bridge must consider this fact.

(g) Two approaches to bridging between Transparent Bridge Domains (e.g. Ethernet) and Source Routing Bridge Domains (e.g. Token Ring) have been proposed which are :

(i) Translational Bridging.

(ii) Source-Route Transparent Bridging.

(h) **Status Bits** : Token Ring use status bits such as :

- A : Indicates if destination *address* is recognized.
- C : Indicates if frame is *copied.*
- E : *Errors.*

The Ethernet does not use any such status bits. There is no clear solution on how to handle this problem of status bits.

4.8 Broadband Wireless

4.8.1 Introduction

- The IEEE 802.16 Working Group on *Broadband Wireless Access Standards* aims to prepare formal specifications for the global deployment of broadband *Wireless Networks.*
- **802.16 Standards** : The first 802.16 standard was approved in December 2001. It delivered a standard for point to multipoint Broadband Wireless transmission in the 10-66 GHz band, with only a Line of Sight capability.
- **802.16a** : 802.16a was an amendment to 802.16 and delivered a point to multipoint capability in the 2-11 GHz band.
- **802.16b** : This standard operates in the 5 GHz ISM band.

4.8.2 The 802.16 Protocol Stack

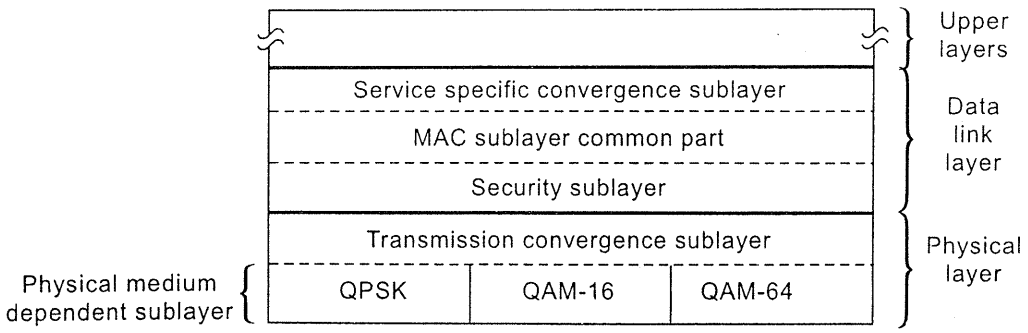


Fig. 4.32 : The 802.16 protocol stack.

Refer *figure 4.3*

(a) The physical la

(i) Bottom pa

(ii) The Transı technologi

(b) The data link la

(i) **Security S**

(ii) **MAC Subl** channel. If

(iii) **Service S** Network la

4.8.3 Physical Lay

- 802.16 uses Sc between 1.25 M
- It supports ada a highly efficie poorer, a more
- In intermediate
- PHY provide s provide higher correction perf

4.8.4 MAC Sublaye

- The 802.16 MA technologies su and how data i
- It also describe exchange durin transfer.
- MAC layer incl and handover r
- A key feature subscriber stati base station .
- 802.16 to provic

Refer *figure* 4.32.

(a) The physical layer deals with actual transmission :

(i) Bottom part of physical layer contains the actual transmission technologies.

(ii) The Transmission Convergence Sublayer hides the difference between these technologies from the datalink layer.

(b) The data link layer consists of three sublayers :

(i) **Security Sublayer :** Does encryption, decryption and key management.

(ii) **MAC Sublayer Common Part :** It manages sending and receiving over the channel. If provides connection oriented service.

(iii) **Service Specific Convergence Sublayer :** It provides an interface to the Network layer.

4.8.3 Physical Layer

- 802.16 uses Scalable OFDMA to carry data, supporting channel bandwidths of between 1.25 MHz and 20 MHz, with up to 2048 sub-carriers.
- It supports adaptive modulation and coding, so that in conditions of good signal, a highly efficient 64 QAM coding scheme is used, whereas where the signal is poorer, a more robust BPSK coding mechanism is used.
- In intermediate conditions, 16 QAM and QPSK can also be employed.
- PHY provide support for Multiple-in Multiple-out (MIMO) antennas in order to provide higher bandwidth and Hybrid automatic repeat request for good error correction performance.

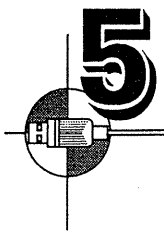
4.8.4 MAC Sublayer

- The 802.16 MAC describes a number of *Convergence Sublayers* which describe how technologies such as Ethernet, ATM and IP are encapsulated on the air interface, and how data is classified, etc.
- It also describes how secure communications are delivered, by using secure key exchange during authentication, and encryption using AES or DES during data transfer.
- MAC layer include power saving mechanisms (using *Sleep Mode* and *Idle Mode*) and handover mechanisms.
- A key feature of 802.16 is that it is a connection oriented technology. The subscriber station cannot transmit data until it has been allocated a channel by the base station .
- 802.16 to provides strong support for Quality of Service (QoS).

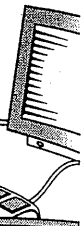
Upper layers

Data link layer

Physical layer



NETWORK LAYER



This chapter deals with the Network Layer. It includes routing algorithms, congestion control algorithms, IPv4, IPv6, OSPF and BGP.

Marks	
Dec. 03	-
May 04	20 M
Dec. 04	46 M
May 05	14 M
Dec. 05	19 M
May 06	20 M
Dec. 06	22 M
May 07	20 M

5.1 Introduction

The network layer is concerned with taking packets from the source and delivering them to the destination. This may require making may hops at intermediate routers along the way.

To achieve this, the network must know the topology of the network (i.e. the arrangement of routers) and choose appropriate paths through it.

5.2 Network Layer Design Issues

(1) **Store and Forward Packet Switching :** A host transmits a packet to the nearest router. The nearest router stores the packet and waits for the entire packet to arrive so that it can verify the checksum. If the checksum verification is successful the packet is routed to the next router along the path. This process continues till the packet reaches its destination.

(2) **Services Provided to the Transport Layer :** Connection Oriented and Connectionless Service (Section 1.9).

5.3 Introduction to Routing Algorithms

Dec. 04 [Q. 5(a)], May 05 [Q. 6(a)], May 07 [Q. 7(a)] *What are the different types of Routing Algorithms ? Explain anyone in detail.* **(10 M)**

Dec. 05 [Q. 4(a)] *What is static routing ? What are the advantages of dynamic routing ? Explain shortest path routing in detail.* **(10 M)**

Routing

It is a process by which packets are efficiently transported from source to destination.

The algorithms that make the routing decisions are called *Routing Algorithms*. In

simple terms this
packet must be se

Properties of Rou

(1) *Correctness a*
and should b

(2) *Robustness :*
network topo

(3) *Efficiency :* Ma

(4) *Stability :* A ro

(5) *Fairness and C*
satisfying on
property. Th
achieved.

Types of Routing

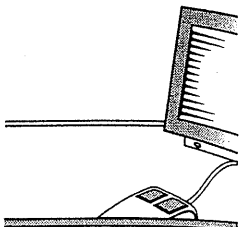
(1) **Non Adaptive**

- In this typ
the presen
- As this typ
the netwo
increases t
- The only a
- E.g. Shorte

(2) **Adaptive or D**

- In this typ
on the pres
- As this typ
network, i
the routing
- The disadv
- E.g. Distan

The Optimality P



Marks

Dec. 03 : -
May 04 : 20 M
Dec. 04 : 46 M
May 05 : 14 M
Dec. 05 : 19 M
May 06 : 20 M
Dec. 06 : 22 M
May 07 : 20 M

simple terms this means that the routing algorithms make decisions as to how a packet must be sent from one router to another.

Properties of Routing Algorithms

(1) *Correctness and Simplicity :* The routing algorithms should give correct results and should be simple to understand.

(2) *Robustness :* A routing Algorithm must work even if there is a change in the network topology.

(3) *Efficiency :* Maximum efficiency is desired from a routing algorithm.

(4) *Stability :* A routing algorithm must be able to cope with congestion.

(5) *Fairness and Optimality :* These properties are obvious. The only problem is that satisfying one of these properties leads to the dissatisfaction of the other property. Therefore a trade-off between fairness and optimality must be achieved.

Types of Routing Algorithms

(1) Non Adaptive or Static Routing

- In this type of routing algorithm the routing decisions are not dependent on the present condition of traffic and topology of the network.
- As this type of routing does not depend on the present condition of traffic in the network, it has poor performance during heavy traffic. Even if traffic increases the routing decisions do not change.
- The only advantage of this method is that it is simple to implement.
- E.g. Shortest Path Routing, Flooding

(2) Adaptive or Dynamic Routing

- In this type of routing algorithm the routing decisions are made depending on the present condition of traffic and topology of the network.
- As this type of routing depends on the present condition of traffic in the network, it has good performance during heavy traffic. As traffic increases the routing decisions change so as to handle the heavy traffic.
- The disadvantage of this method is that it is complex.
- E.g. Distance Vector Routing, Hierarchical Routing.

The Optimality Principle

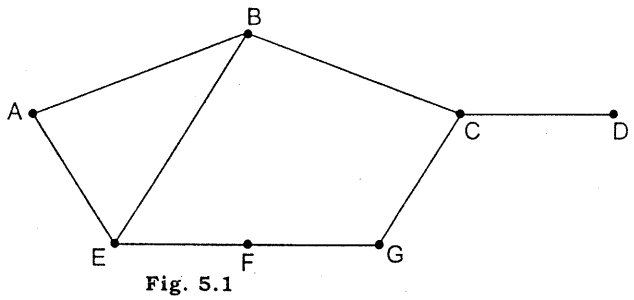


Fig. 5.1

It states that "If router J is on the optimal path from router I to router K; then the optimal path from J to K also falls along the same route."

Consider the path from A to D in the *figure 5.1*. The optimal path (shortest path) is from A → B → C → D.

Now from the Optimality principle the optimal path of B to D also falls along the same route. Therefore the optimal path from B to D is B → C → D.

Note : In the above case we have decided optimality based on distance. Therefore the optimal path is the shortest path. Other metrics can also be used to decide optimality.

Consequence of the Principle

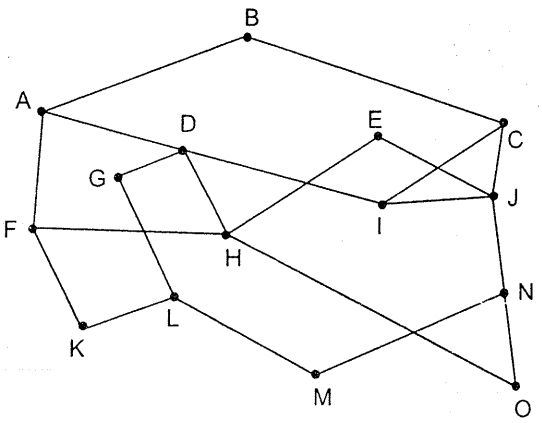


Fig. 5.2(a) : A subnet.

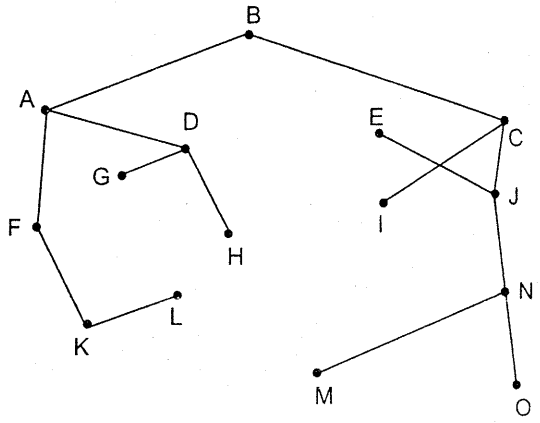


Fig. 5.2(b) : A sink tree for router B.

The set of paths from all the sources to a given destination [node B in the *figure 5.2(b)*] forms a Tree called as a *Sink Tree*. In a sink tree the distance metric is generally the number of hops. A sink tree is not necessarily unique.

5.4 Routing Algorithms

[I] Shortest Path Routing

This is the same as the Dijkstras Algorithm for finding shortest path between two nodes.

Dijkstra's algorithm is used to calculate the shortest route from source router to destination router.

Basics of Algorithm

- Each router is known as a node.
- Each node has a
  - (1) Cost value
  - (2) A pointer to the node that is before it in the path being taken.
  - (3) A state [permanent or tentative].

- Initially e
  value of i
- The sour
  still to nu
- The sour
  assigns th
  these nod
- After this
  permaner
  marked-p
- When the
  cost value
- To find th
  the pointe

To illustrate how t

Given the followi

Steps to Finding t

(1) Mark node

(2) Calculate t

(3) Mark these
nodes to 'a'

(4) Choose the
namely 'b' a

(5) Calculate t
'd' = 5 + 3

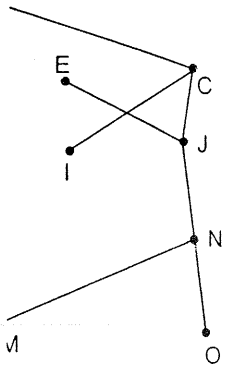
(6) Mark the n
to 'b'.

ıter K; then the

ıhortest path) is

ı falls along the

refore the optimal
'.



e for router B.

: B in the *figure*
etric is generally

st path between

source router to

aken.

- Initially each node is given a state of tentative , a pointer to null and , a cost value of infinity.
- The source node is marked as permanent and it's cost value as 0 and pointer still to null.
- The source node then calculates the cost values to its adjacent nodes and assigns the cost values of these nodes to the values calculated , the pointer of these nodes now points to the source node.
- After this is done the tentative node with the lowest cost value is marked permanent and the whole process is performed again with this newly marked-permanent node.
- When the goal is reached , the total cost of the path is the cost value in the cost value field in the node.
- To find the route taken to achieve this shortest path simply work back from the pointers in the nodes starting with the destination node.

To illustrate how this algorithm works lets work through an example.

Given the following graph :

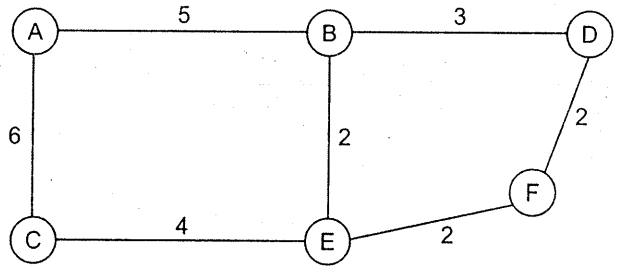


Fig. 5.3

Steps to Finding the Shortest Path form Node/Machine 'a' to Node/Machine 'f'

(1) Mark node 'a' as permanent with no pointer value and a cost of 0.

(2) Calculate the cost values of the machines that 'a' can reach : 'b' = 5 , 'c' = 6.

(3) Mark these nodes 'b' and 'c' with these costs and set the pointer in these nodes to 'a'.

(4) Choose the node marked tentative with the shortest non-infinity cost value, namely 'b' and mark this node permanent.

(5) Calculate the cost values of the tentative machines that 'b' can reach :
'd' = 5 + 3 = 8 , 'e' = 5 + 2 = 7.

(6) Mark the nodes'd' and 'e' with these costs and set the pointer in these nodes to 'b'.

(7) Choose the node marked tentative with the shortest non-infinity cost value, namely 'c' and mark this node permanent.

(8) Calculate the cost values of the tentative machines that 'c' can reach :
'e' = 6 + 4 = 10.

(9) Since 'e' has already been marked by 'b' as having cost 7 then we do not change the values of 'e'.

(10) Choose the node marked tentative with the shortest non-infinity cost value , namely 'e' and mark this node permanent.

(11) Calculate the cost values of the tentative machines that 'e' can reach :
'f' = 7 + 2 = 9.

(12) Mark the node 'f' with this cost value and set the pointer in the node to 'e'.

(13) Choose the node marked tentative with the shortest non-infinity cost value , namely 'd' and mark this node permanent.

(14) Calculate the cost values of the tentative machines that 'd' can reach :
'f' = 8 + 2 = 10.

(15) Since 'f' has already been marked by 'e' as having cost 9 then we do not change the values of 'f'.

(16) Choose the node marked tentative with the shortest non-infinity cost value , namely 'f' and mark this node permanent.

(17) The cost value of 'f' is 9 set by 'e'. This value 9 is the total cost of the shortest path from 'a' to 'f'.

(18) To find the path taken work back from 'f' using the pointer → this gives 'e' ,

(19) Working back from 'e' using it's pointer → this gives 'b'.

(20) Working back from 'b' using it's pointer → this gives 'a'.

'a' is in fact the source node .

This means we can find the whole path from the source to the goal.

'f' → 'e' → 'b' → 'a'.

Thus the solution of the shortest path form 'a' to 'f' is : a b e f with a total cost of 9.

[II] Flooding

- Every incoming packet is sent on every outgoing line except the line it has arrived on.
- *Problem* : In this method duplicate packets keep getting generated at each node and therefore network traffic is high. To use this method a large bandwidth is required.

- A solution t
  counter is
  discarded.
- The advanta
  (1) *Robustn*
  (2) *First pa*
  reaches

[III] Distance Vec

May 04 [Q. 2(a)] *W*
table structure.

- Now-a-days,
  the static on
  into account
  (DVR) and li
- The distance
  such as the
  the RIP .
- Distance vec
  table (the tab
  (a) The best
  (b) Which li
  These tables

Routing Table

- Each router
  for, each rout
- This entry co
  (a) The prefe
  (b) An estima
- The distance
  number of pa
- Each router is
  (a) If the met
  (b) If the me
  ECHO pa
  neighbou
  The value

- A solution to this problem is to maintain a hop counter with each packet. The counter is decremented at each hop. If the counter reaches 0 the packet is discarded.
- The advantages of this method are :
  (1) *Robustness :* The system does not fail due to failure of one of the links.
  (2) *First packet to reach follows the shortest path :* Always the first packet reaches the destination via the shortest path.

[III] Distance Vector Routing

May 04 [Q. 2(a)] *What does routing mean and how does it work ? Describe routing table structure.* **(10 M)**

- Now-a-days, networks generally use dynamic routing algorithms rather than the static ones because static algorithms do not take the current network load into account. Two popular dynamic algorithms are distance vector routing (DVR) and link state routing (LSR), in this section we will study the DVR.
- The distance vector routing algorithm is sometimes called by other names such as the Bellman-Ford routing algorithm, Ford Fulkerson algorithm and the RIP .
- Distance vector routing algorithms operate by having each router maintain a table (the table is called a *vector*) giving :
  (a) The best known distance to each destination and
  (b) Which line to use to get there.
  These tables are updated by exchanging information with the neighbors.

Routing Table

- Each router maintains a routing table indexed by, and containing one entry for, each router in the subnet.
- This entry contains two parts :
  (a) The preferred outgoing line to use for that destination and
  (b) An estimate of the time or distance to that destination.
- The distance metric used might be number of hops, delay in milliseconds or number of packets queued along the path.
- Each router is assumed to know the "distance" to each of its neighbors.
  (a) If the metric is hops, the distance is just one hop.
  (b) If the metric is delay, the router can measure it directly with special ECHO packets (The sender sends an ECHO packet to its neighbour. The neighbour just timestamps the packet and sends it back to the sender. The value of the timestamp is the delay value).

(c) If the metric used is number of packets queued along the path then the router examines each queue.

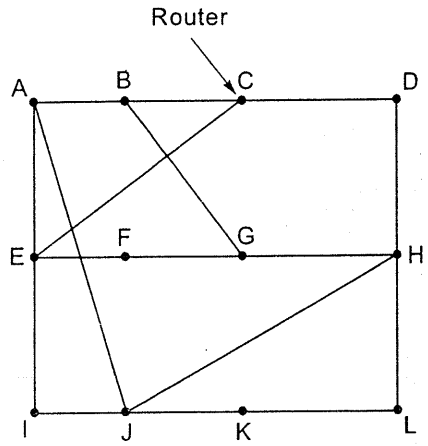


Fig. 5.4(a) : A subnet.

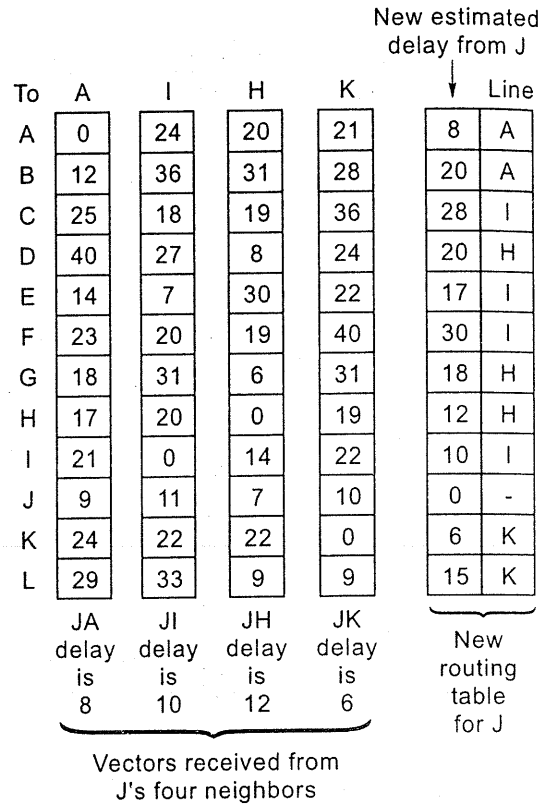


Fig. 5.4(b) : Input from A, I, H, K, and the new routing table for J.

Example

- In the above example delay is used as a metric and the router knows the delay to each of its neighbors.
- Every T msec each router sends a list of its estimated delays to each destination, to each of its neighbours. It also receives a similar list from each neighbor.
- **Rule** : Imagine that X and Y are neighbours. X sends a table to Y saying that the delay between router X and router I is Xi. Now Y knows that the delay to its neighbour X is m. It calculates the delay to router I via router X as m + Xi. Now if this value is less than the current value stored for I, the value is updated in Y's router table.
- This updating process is illustrated in the *figure 5.4*. J has neighbours A, I, H and K. Suppose that J has delay to its neighbors, A, I, H, and K as 8, 10, 12, and 6 msec, respectively.

e path then the

New estimated delay from J

	Line
8	A
20	A
28	I
20	H
17	I
30	I
18	H
12	H
10	I
0	-
6	K
15	K

ay

New routing table for J

A, I, H, K, and ıble for J.

ɔuter knows the

delays to each ar list from each

to Y saying that that the delay to ıter X as m + Xi. r I, the value is

eighbours A, I, H ıd K as 8, 10, 12,

- Understand how J computes its new route to router G. It knows that it can get to A in 8 msec, and A claims to be able to get to G in 18 msec, so J knows it can count on a delay of 26 msec to G via A. Similarly, it computes the delay to G via I, H, and K as 41 (31 + 10), 18 (6 + 12), and 37 (31 + 6) msec, respectively. The least of these values is 18, so it makes an entry in its routing table that the delay to G is 18 msec and that the route to use is via H. The same calculation is performed for all the other destinations.

The Count-to- Infinity Problem

Dec. 04 [Q. 3(b)] *Explain Count-to-infinity .problem with the help of an example. It is a drawback of which algorithm ?* **(10 M)**

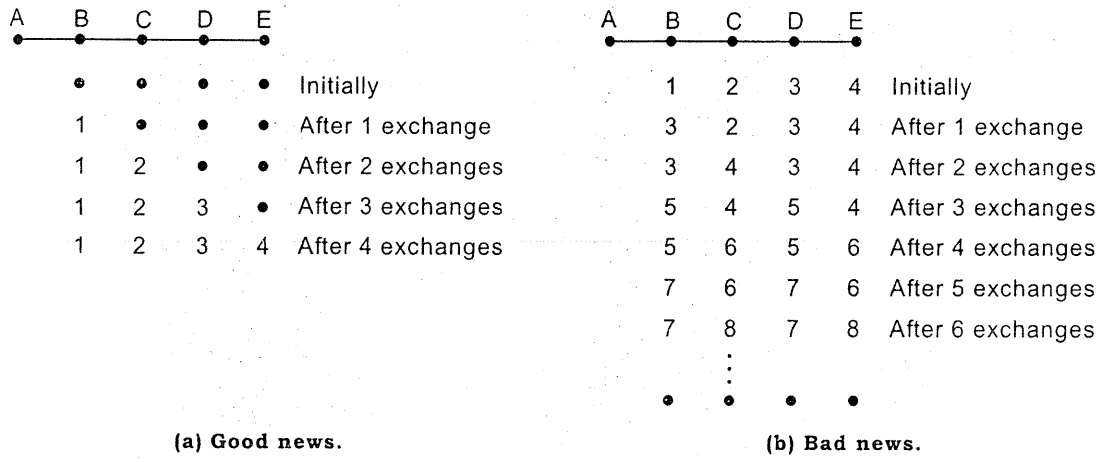


Fig. 5.5 : The count-to-infinity problem.

Distance vector routing works in theory but has a serious drawback in practice : It reacts rapidly to good news, but leisurely to bad news.

a) **Good News**

- To see how fast good news propagates, consider the five-node (linear) subnet in the *figure 5.5(a)*, where the delay metric is the number of hops.
- Suppose A is down initially and all the other routers know this and they have all recorded the delay to A as infinity.
- When A comes up, the other routers learn about it via the vector exchanges. B now makes an entry in its routing table that A is one hop away to the left.
- All the other routers still think that A is down. At this point, the routing table entries for A are as shown in the second row of the *figure 5.5(a)*.
- On the next exchange, C learns that B has a path of length 1 to A, so it updates its routing table to indicate a path of length 2. At each successive hop D and E hear the good news as shown in the figure.
- Clearly, the good news is spreading at the rate of one hop per exchange.

(b) Bad News

- Now let us consider the situation of *figure 5.5(b)*. All the lines and routers are initially up. Routers B, C, D, and E have distances to A of 1, 2, 3, and 4, respectively. Suddenly A goes down.
- At the first packet exchange, B does not hear anything from A. Fortunately, C says : Do not worry; I have a path to A of length 2. B is unaware that C's path runs through B itself. As a result, B thinks it can reach A via C, with path length of 3.
- D and E do not update their entries for A on the first exchange.
- On the second exchange, C notices that each of its neighbors (B, D) claims to have a path to A of length 3. It picks one of them at random and makes its new distance to A as 4, (third row of the figure). Subsequent exchanges are shown in the rest of *figure 5.5(b)*.
- From this figure, it should be clear why bad news travels slowly. Gradually, all routers work their way up to infinity.
- The problem is that when X tells Y that it has a path somewhere, Y has no way of knowing whether it itself is on the path.

[IV] Link State Routing

Distance vector routing was used in the ARPANET until 1979, when it was replaced by link state routing. Two problems caused its demise.

(a) It did not take line bandwidth into account when choosing routes.

(b) The count to infinity problem

The idea behind *link state routing* can be stated as five parts.

(1) Learning About the Neighbours

When a router is booted, its first task is to learn who is it neighbours are. A router sends a HELLO packet on each of its lines. The neighbouring routers, connected on the end of these lines, respond to the HELLO packet by sending their network address.

(2) Measuring Line Cost

The link state routing algorithm requires each router to know, or at least have a reasonable estimate, of the delay to each of its neighbors. The most direct way to determine this delay is to send a special ECHO packet over the line that the other side is required to send back immediately.

(3) Building Link States Packets

Once the information needed for the exchange has been collected, the next step

is for each rout
the identity of
neighbours. Th
six routers is gi

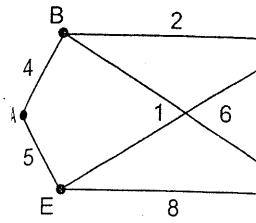


Fig. 5.6(a) : The Su

4) Distributing th

The Basic Distri
distribute the li
a sequence num
packet comes in
it is forwarded
discarded.

5) Computing the

Once a router
subnet. Now Dij
destinations.

OSPF, which is c

V] Hierarchical R

- We have seer
Therefore as
will also incr
therefore the
- To reduce th
routing techn
- In this metho
know how to
- Consider the

is for each router to build a packet containing all the data. The packet starts with the identity of the sender, followed by a sequence number and age and a list of neighbours. The example of subnet and corresponding link state packets for all six routers is given in *figure 5.6(a)* and *figure 5.6(b)*.

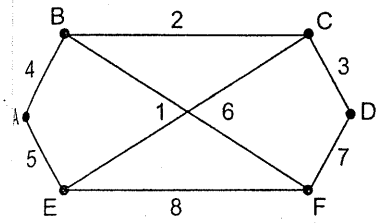


Fig. 5.6(a) : The Subnet.

A	
Seq.	
Age	
B	4
E	5

B	
Seq.	
Age	
A	4
C	2
F	6

C	
Seq.	
Age	
B	2
D	3
E	1

D	
Seq.	
Age	
C	3
F	7

E	
Seq.	
Age	
A	5
C	1
F	8

F	
Seq.	
Age	
B	6
D	7
E	8

Fig. 5.6(b) : Link state packets.

4) Distributing the Link State Packets

The Basic Distribution Algorithm : The fundamental idea is to use flooding to distribute the link state packets. To keep the flood in check, each packet contains a sequence number. Routers keep track of all the packets they see. When a new packet comes in, it is checked against the list of packets already seen. If it is new, it is forwarded on all except the one it arrived on. If it is a duplicate, it is discarded.

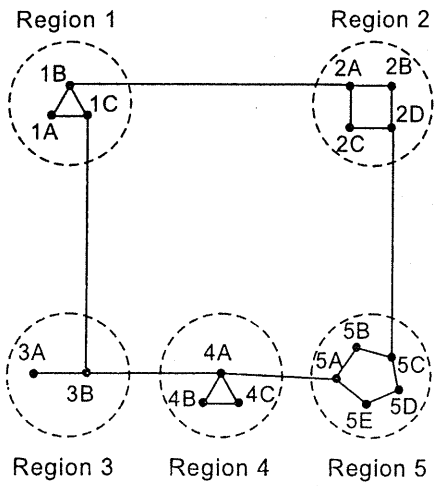
5) Computing the New Routes

Once a router has a full set of link state packets it can construct the entire subnet. Now Dijkstras algorithm can be used to find shortest path to all possible destinations.

OSPF, which is covered later in this chapter, uses link state algorithm.

V] Hierarchical Routing

- We have seen that a router table has an entry for each router in the network. Therefore as number of routers in a network increases the router table size will also increase. In practical networks the number of routers is very large therefore the number of rows needed in a router table is also very large.
- To reduce the size of the router table for each router we use hierarchical routing technique.
- In this method the network is subdivided into regions. Routers in a region know how to route packets within the same region but not outside it.
- Consider the following diagram :



(a)

Full table for 1A

Dest.	Line	Hops
1A	-	-
1B	1B	1
1C	1C	1
2A	1B	2
2B	1B	3
2C	1B	3
2D	1B	4
3A	1C	3
3B	1C	2
4A	1C	3
4B	1C	4
4C	1C	4
5A	1C	4
5B	1C	5
5C	1B	5
5D	1C	6
5E	1C	5

(b)

Hierarchical table for 1A

Dest.	Line	Hops
1A	-	-
1B	1B	1
1C	1C	1
2	1B	2
3	1C	2
4	1C	3
5	1C	4

(c)

Fig. 5.7 : Hierarchical routing.

- The *figure 5.7* shows that the division of the network into subnetworks. It also shows the full table and the hierarchical table for node 1A.
- E.g. In Distance Vector Routing a network with 720 routers has 720 rows in each router table.

  However if the network is divided into 24 regions with 30 routers in each region then each router will have 30 + 23 = 53 entries because

  (1) 30 entries for each router within the same subnetwork.

  (2) 23 entries for the remaining subnetworks.
- *Disadvantage* : The path followed by this method may not always be the shortest path. In the above figure the best route from 1A to 5C is via region 2 (5 hops); but with hierarchical routing all traffic to region 5 goes via region3 (6 hops).
- As shown above, we divide a network into subnetworks so as to reduce the number of entries in the router table. Sometimes the number of routers in a subnetwork is also too large. We have to further divide the subnetworks into sub-subnetw router table
- Consider th divided into

[VI] Broadcast Ro

Sending a packet to
The following meth

Method I :

A distinct pa

Method II : Floodin

Every incom arrived on.

The problem *required for th*

Method III : Multide

- It is used wh not necessari
- In this metho
- The router determines th router prepa destinations t

Method IV : Spannin

- A spanning tr
- The spanning the routers.
- Each router When an inco all the spann Thus in this w

Note : The difference only one rout the packet is i

erarchical table for 1A

Dest.	Line	Hops
1A	-	-
1B	1B	1
1C	1C	1
2	1B	2
3	1C	2
4	1C	3
5	1C	4

(c)

ito subnetworks. It
e 1A.

ers has 720 rows in

30 routers in each
use

not always be the
o 5C is via region 2
5 goes via region3

so as to reduce the
ber of routers in a
e subnetworks into

sub-subnetworks. This process continues till the number of entries in the router table is sufficiently small.

- Consider that a network has N routers. Optimally the network should be divided into ln *N* levels. Each router table will contain e.ln *N* entries.

[VI] Broadcast Routing

Sending a packet to many or all the destinations is called as *broadcasting*.

The following methods can be used for broadcasting.

Method I :

A distinct packet is sent to each destination. This process wastes bandwidth.

Method II : Flooding

Every incoming packet is sent on every outgoing line except the line it has arrived on.

The problem that flooding faces is the same as before : *Too much bandwidth is required for the excess traffic.*

Method III : Multidestination Routing

- It is used when the router wants to send a packet to many destinations but not necessarily all the destinations.
- In this method each packet contains a list of destinations.
- The router checks all the destinations contained in the packet and determines the output lines on which the packet needs to be forwarded. The router prepares a packet for each such line and includes in each, a list of destinations that will use that line.

Method IV : Spanning Tree

- A spanning tree is built for the router initiating the broadcast.
- The spanning tree will consists of all the routers but not all the links between the routers.
- Each router should know which of its lines belong to the spanning tree. When an incoming packet comes to a router, the router sends the packets on all the spanning tree lines except the line on which the packet arrived on. Thus in this way there is less traffic here compared to flooding.

Note : The difference between broadcasting and flooding is that in flooding there is only one router for whom the packet is intended for. In case of broadcasting the packet is intended for all the routers on the network.)

Method V : Reverse Path Forwarding (RPF)

Process :

(1) Suppose X wants to send a broadcast to all the routers in the network. This means that X is the source of the broadcast.

(2) When a packet arrives at a router Y, the router Y checks to see on which line the packet has arrived.

(3) If Y uses the same line when it has to send a packet to X, then we can safely say that the packet has followed the optimum path between X and Y and it is the first packet to arrive at Y. This line is known as the *preferred line*. Now the packet is forwarded along all the lines except for the preferred line(i.e. the line on which the packet arrived on).

(4) However if a packet arrived on any other line apart from the preferred line then the packet will be considered as a duplicate and discarded.

E.g.

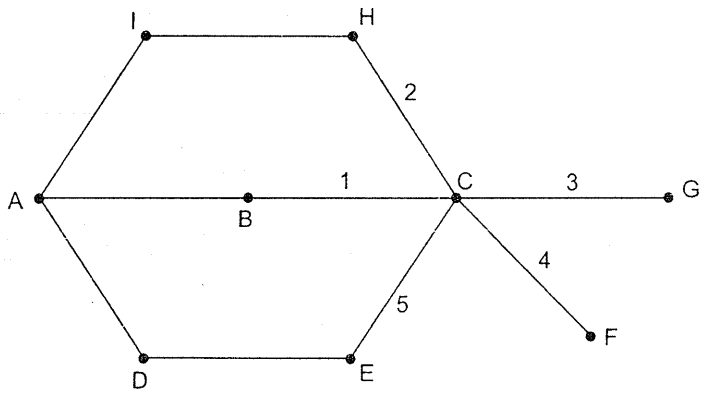


Fig. 5.8

- In the above *figure 5.8*, A wants to broadcast to all the other routers in the network.
- Now if C receives a packet on link 1; C will check that suppose it had to send a packet to A which link would it put the packet on. The answer is Link 1. Therefore the packet coming to C on link 1 has followed the optimum path between A and C. As the packet has followed the optimum path between A and C this packet is the first packet that has arrived at C and will be accepted. Now the packet will be forwarded on links 2, 3, 4, 5.
- Consider the case that a packet arrives at C on Link 2. Now C checks whether it will use link 2 to send a packet from C to A. The answer is NO. Therefore C considers this packet as a duplicate packet and discards it.

This method has the following advantages over the previous broadcasting methods :

(1) This method does not need spanning trees as in Method IV

(2) The packets

(3) The packet d

[VII] Multicast Ro

- Some applica groups. A pr the same gro message to broadcasting becomes ineff
- In multicast consists of c leave/join a g
- All the router

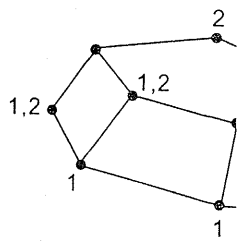


Fig. 5.9(a) : A

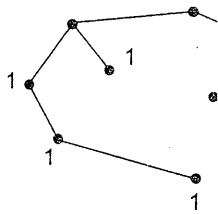


Fig. 5.9(c) : A multicas

- Consider the *f* processes that
  - Each rout
  - A spannin
  - When a p examined to the rou
  - *Figure 5.9(*
  - *Figure 5.9(*
  - If a netwo average of

(2) The packets need not contain a list of destinations as in Method III

(3) The packet does not need a hop counter as in case of Method II.

[VII] Multicast Routing

- Some applications require that distinct processes should work together in groups. A process should be able to communicate with all other processes in the same group. If the group is small, processes can send a point-to-point message to all the other members of the group. If the group is large broadcasting can be used. But if the groups are very large broadcasting becomes inefficient and hence Multicast Routing must be used.
- In multicast routing we require group management (group management consists of creating groups, destroying groups and allowing processes to leave/join a group)
- All the routers must be aware of which processes are a part of which groups.

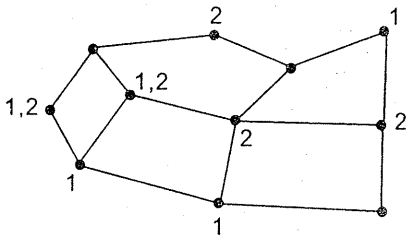


Fig. 5.9(a) : A network.

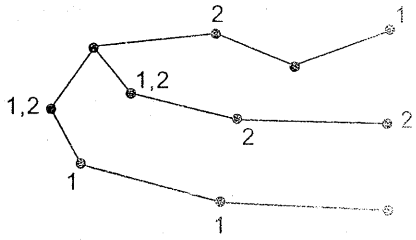


Fig. 5.9(b) : A spanning tree for the leftmost router.

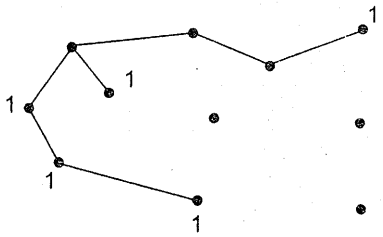


Fig. 5.9(c) : A multicast tree for group 1.

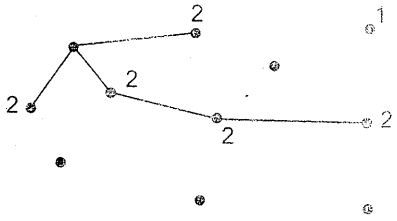


Fig. 5.9(d) : A multicast tree for group 2.

- Consider the *figure 5.9* where there exist 2 groups, 1 and 2. Some routers have processes that belong to any one group or both the groups.
  - Each router has a spanning tree which contains all other routers.
  - A spanning tree for the left most router is shown in the *figure 5.9(b).*
  - When a process sends a multicast packet to a group, the spanning tree is examined and pruned. The pruned spanning tree will only contain links to the routers that contain group members.
  - *Figure 5.9(c)* shows the pruned spanning tree for group 1.
  - *Figure 5.9(d)* shows the pruned spanning tree for group 2.
  - If a network contains N groups with M members in each group then an average of N * M pruned spanning trees will be stored.

5.5 Congestion Control

Dec. 05 [Q. 2(c)] *Short Note : Congestion Control.* (4 M)

Congestion

- When excess packets are present in a part of the network the performance degrades. This is called as congestion.
- Congestion occurs when the system load is greater than the system resources. Congestion can be caused when :
  - **(1)** Many packets are routed through the same node (router).
  - **(2)** There is low bandwidth.
  - **(3)** There are slow processors at the routers.
  - **(4)** If a part of the network is upgraded and the remaining network is left as it is, congestion will occur at the non upgraded section of the network.
  - **(5)** Bursty traffic.

Difference Between Congestion Control and Flow Control

Sr. No.	Congestion Control	Flow Control
(1)	It ensures that the network is able to carry the offered traffic.	It ensures that a fast sender does not drown a slow receiver.
(2)	It is a global issue i.e. it relates to all hosts and routers.	It relates to the point to point traffic between senders and receivers.
(3)	It is generally done by the router.	It is generally done by the sender.

Types of Congestion Control Algorithms

(1) Open Loop Algorithms

They make sure that congestion does not occur in the first place. They aim at designing the system in such a way that congestion cannot occur.

(2) Closed Loop Algorithms

They let congestion occur and then take corrective steps. The general steps followed by a closed loop algorithm are :

(a) Monitor the system and find out where congestion is occurring.

(b) Pass this information to the places where action can be taken.

(c) Take actions to correct the problem.

5.6 Open Loop

There are two

[I] Prevention Po

[II] Traffic Shapin

[I] Prevention

These systems ar
letting it happen
layers are given b

Layer
Transpor
Network
Data link

Data Link Layer

(a) The *retrans*
transmits u
all outstanc

(b) If receivers
be transmit
in selective
than go bac

(c) *Acknowled*
acknowledg
traffic. Ho
reverse traf

5.6 Open Loop Techniques for Congestion Control

(4 M)

There are two methods :

[I] Prevention Policies

[II] Traffic Shaping

[I] Prevention Policies

These systems are designed to minimize congestion in the first place, rather than letting it happen and reacting after the fact. The prevention policies for the various layers are given below :

Layer	Policies
Transport	• Retransmission policy. • Out-of-order caching policy. • Acknowledgement policy. • Flow control policy. • Timeout determination.
Network	• Virtual circuits versus datagram inside the subnet. • Packet queuing and service policy. • Packet discard policy. • Routing algorithm. • Packet lifetime management.
Data link	• Retransmission policy. • Out-of-order caching policy. • Acknowledgement policy. • Flow control policy.

Fig. 5.10 : Policies that affect congestion.

Data Link Layer

(a) The *retransmission policy* decides how fast a sender times out and what it transmits upon timeout. A jumpy sender times out quickly and retransmits all outstanding packets.

(b) If receivers routinely discard all *out-of-order packets*, these packets will have to be transmitted again later, creating extra load. Thus selective repeat (because in selective repeat packets need not be delivered in order) is clearly better than go back n (because in go back n packets must be delivered in order).

(c) *Acknowledgement policy* also affects congestion. If each packet is acknowledged immediately, the acknowledgement packets generate extra traffic. However, if acknowledgements are saved up to piggyback onto reverse traffic the load reduces.

(d) A tight *flow control* scheme helps fight congestion.

Network Layer

(a) The choice between *using virtual circuits and using datagrams* affects congestion since many congestion control algorithms work only with virtual-circuit subnets.

(b) *Packet queuing and service policy* decides to whether routers have one queue per input line, one queue per output line, or both. It also decides the order in which packets are processed (e.g. priority based).

(c) *Discard policy* decides which packet is dropped when there is no storage space.

(d) A good *routing algorithm* can help avoid congestion.

(e) *Packet lifetime* management deals with how long a packet may live before being discarded. If it is too long, lost packets may clog up the network for a long time, but if it is too short, packets may time out before reaching their destination.

Transport Layer

(a) In the transport layer, the same issues occur as in the data link layer

(b) In addition the *timeout interval* is determined. If the timeout interval is too short, extra packets will be sent unnecessarily. If it is too long the response time will suffer whenever a packet is lost.

[II] Traffic Shaping

- One of the main causes of congestion is the bursty nature of traffic.
- Traffic Shaping forces bursty traffic to be transmitted at a uniform rate.

Dec. 06 [Q. 6(a)] *Explain leaky bucket algorithm in detail. Also explain advantages and disadvantages of same compared with token bucket algorithm.* **(10 M)**

Techniques for Traffic Shaping

(1) Leaky Bucket Algorithm

(2) Token Bucket Algorithm

(1) Leaky Bucket Algorithm

- Host is connected to the network by an interface. This interface is actually a queue.
- The host sends an unregulated flow of traffic to the queue (interface). If the queue is not full the packet is appended to the queue. If the queue is full the packet is discarded.

- Now the
  1 packet
  network.
- Hence in
  traffic (b
  host is re
  it on the
- The only
  packets a
  at a slow
  Even du
  rate rema

(2) Token Bucke

- Host is co
  bucket. A
- The host
  flow to th
  be transm
  must capt
  present in
- If the hos
  to the bu
  getting ac
  Generally
  amount c
  accumulat
- Due to t
  getting acc
  handled b
  case the ra
  saved in
  leaky buck
  be constan

- Now the queue (interface) puts 1 packet per clock tic into the network.
- Hence in this way unregulated traffic (bursty traffic) from the host is regulated before putting it on the network.
- The only problem is that the packets are put on the network at a slow rate (1 per clock tic). Even during large bursts this rate remains the same.

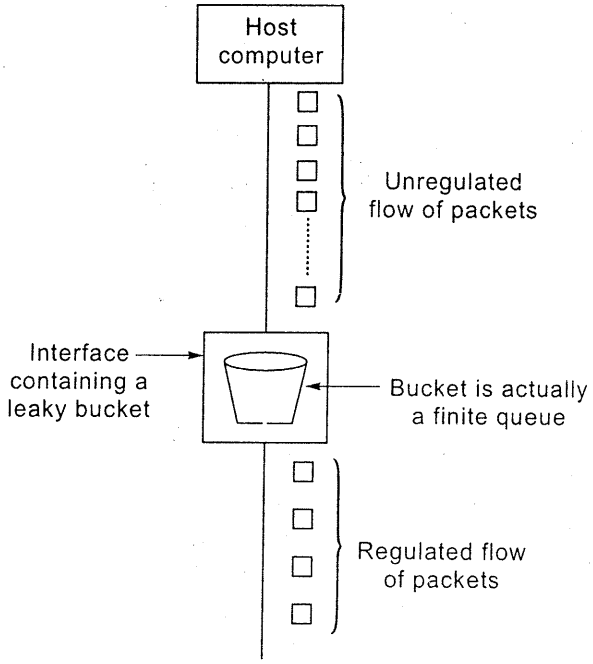


Fig. 5.11 : A leaky bucket with packets.

(2) Token Bucket Algorithm

- Host is connected to the network by an interface. This interface is actually a bucket. A token is generated in the bucket every ΔT seconds.
- The host sends an unregulated flow to the bucket. For a packet to be transmitted to the network, it must capture and destroy a token present in the bucket.
- If the host is not sending packets to the bucket the tokens keep getting accumulated in the bucket. Generally there is a maximum amount of tokens that can be accumulated in the bucket.
- Due to this feature of tokens getting accumulated, bursts can be handled better. Therefore in this case the rate increases if tokens are saved in the bucket, whereas in leaky bucket the rate will always be constant (1 packet per clock tic).

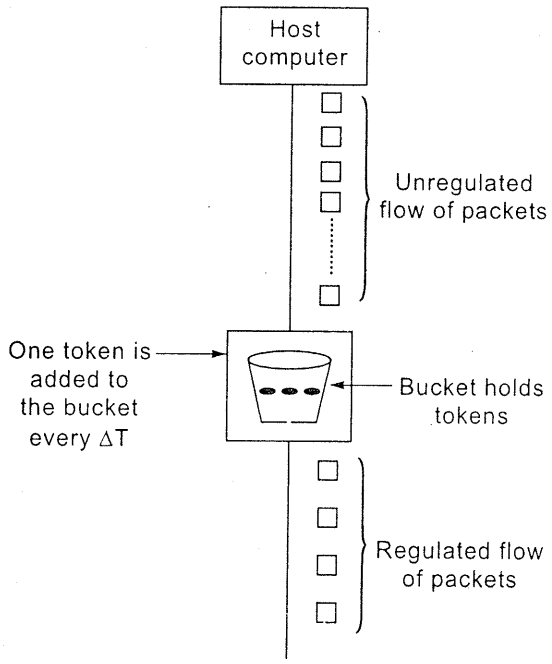


Fig. 5.12 : Token bucket with packets.

5.7 Closed Loop Techniques for Congestion Control

[I] Congestion Control in Virtual Circuit Subnets

(1) Admission Control

Once congestion has been detected no more virtual circuits are set up until the problem has gone away.

(2) Alternative Approach

Virtual Circuits are set up but they do not pass through the areas which are facing congestion.

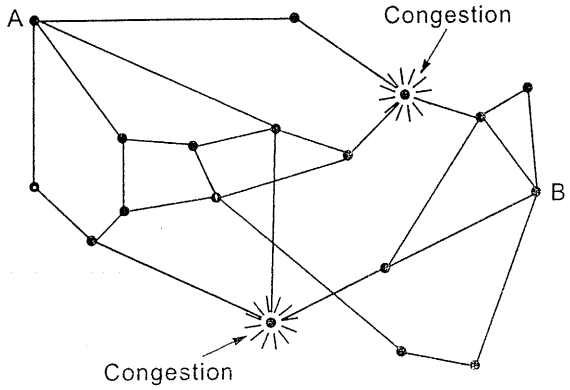


Fig. 5.13(a) : A congested subnet.

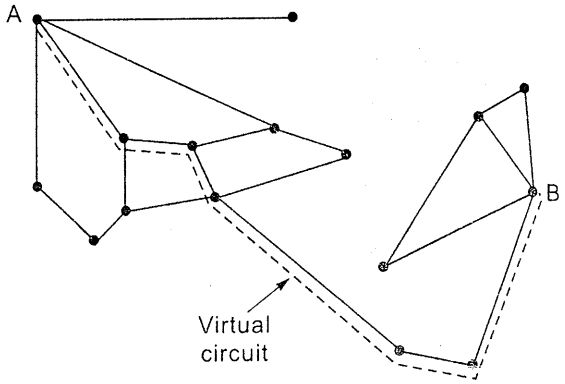


Fig. 5.13(b) : A redrawn subnet that eliminates the congestion.A virtual circuit from A to B is also shown.

Suppose A wants to set up a connection with B. The shortest path between A and B is through the nodes facing congestion. To avoid this situation we can redraw the virtual circuit as shown in *figure 5.13(b)*.

(3) Agreement Negotiation

- An agreement is made between the host and the subnet when the virtual circuit is set up.
- The agreement decides on parameters such as volume of traffic, shape of traffic, quality of service and other network parameters.
- These resources (bandwidth, table and buffer space) are reserved along the path when the virtual circuit is set up. Due to this reservation of resources congestion cannot occur.

[II] Congestion Control in Datagram Subnets

Dec. 04 [Q. 3(a)] *Give one approach of Congestion Control in Datagram Subnets.* **(10 M)**

- These methods can be used in both Datagram and Virtual Circuit Subnets.
- Let u be the utilization of an output line of a router. The value of u is between 0 and 1. The relation between u and f (instantaneous utilization) is given by

u_r

- Whenever a p
on which the
lines

If u is NOT
congestion.

If u is above
(congestion sta

If a line enters w
congestion. Any on

(1) Warning Bit.

(2) Choke Packets.

(3) Hop by Hop Ch

(1) The Warning B

There is a sp
header. This bi
enters the wa
packet arrives
content of this
set, a message i
with the acknov
this message,
traffic it is putti

(2) Choke Packets

- When a line
the router se
sender of t
packet conta
destination.
the sender t
the destinati
When the so
packet it red
percentage.
- The original
it does not g
packets on its

$$u_{new} = a \cdot u_{old} + (1 - a) \cdot f \qquad \text{where 'a' is a constant.}$$

- Whenever a packet arrives at a router, the router first decides on the output line on which the packet will be forwarded on. The router checks the u of its output lines

  If u is NOT above a certain threshold value we can say that there is no congestion.

  If u is above the threshold value, the output line enters the warning state (congestion state).

If a line enters warning state some action should be taken to get rid of the congestion. Any one of the following actions can be taken :

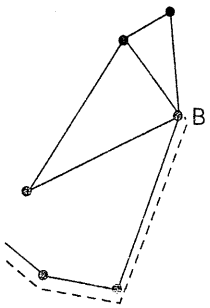
(1) Warning Bit.

(2) Choke Packets.

(3) Hop by Hop Choke Packets.

et up until the

:eas which are



**: eliminates the
› B is also shown.**

ath between A
:uation we can

ıen the virtual

·affic, shape of

rved along the
ɔn of resources

ubnets. **(10 M)**

Subnets.

u is between 0
given by

(1) The Warning Bit

There is a special bit in the packet header. This bit is set if the output line enters the warning state. When the packet arrives at the destination, the content of this bit is checked. If the bit is set, a message is sent to the sender along with the acknowledgement. On receiving this message, the sender reduces the traffic it is putting on the network.

(2) Choke Packets

- When a line goes into warning state, the router sends a choke packet to the sender of the message. The choke packet contains the address of the destination. The choke packet tells the sender that the path leading to the destination is getting congested. When the source host gets the choke packet it reduces the traffic by some percentage.
- The original packet is tagged so that it does not generate any more choke packets on its way to the destination.

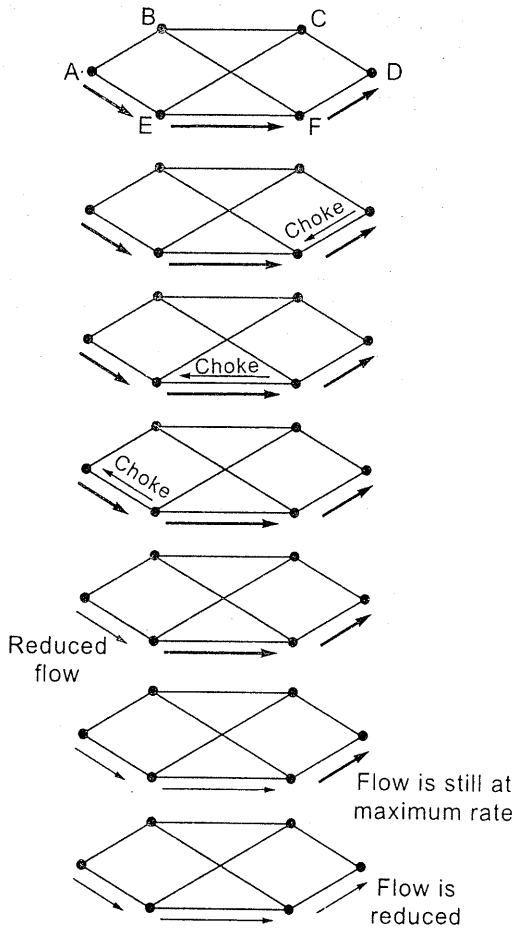


Fig. 5.14 : A choke packet that affects only the source.

- The sender ignores the choke packets for the same destination for a specified period of time, since these will be the choke packets generated due to the packets that were already underway before the 1st choke packet reached the sender. If after the specified period the sender again receives the choke packet it means that the network is still congested and the traffic from the sender must be further reduced by some percentage.
- Consider the above *figure 5.14*. In this case D is assumed to be congested. Now D will send a choke packet to the sender (A) and ask A to reduce the traffic. Only when the choke packet reaches A, will the traffic be reduced.

(3) Hop by Hop Choke Packets

- The above method of choke packets will not work well over long distances. This is because the flow will be reduced only when the choke packet reaches the sender.
- In the Hop by Hop method the flow will be reduced at each stage. Consider the *figure 5.15*.
- Assume that D is congested. Now D will send a choke packet to the sender A. When the choke packet reaches F the flow from F will be reduced. Similarly when the choke packet reaches E the flow from E will be reduced. Finally when the choke packet reaches A the flow from A will be reduced. Thus in this case the choke packet causes the flow to be reduced at each stage; whereas in the previous method the flow was reduced only when the choke packet reached the sender.
- The above procedure is achieved by buffering the extra flow at the routers thereby only letting a limited flow of packets through.

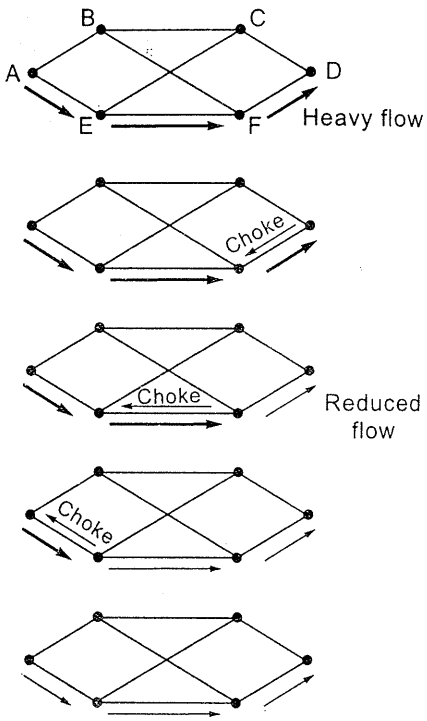


Fig. 5.15 : A choke packet that affects each hop it passes through.

5.8 Load Shedd

When a route
discarded.

Should old or new

It depends on the

(1) File Transfer

- In file trans
- If an old p
  from this ga

(2) Multimedia

- In multimed
- In multime
  should retai

5.9 Quality of Se

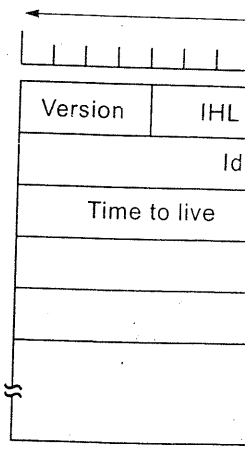
Quality of serv
(Refer section 5.6 [II

5.10 The Network

Dec. 05 [Q. 4(c)] *De*

May 06 [Q. 1(a)] *List*

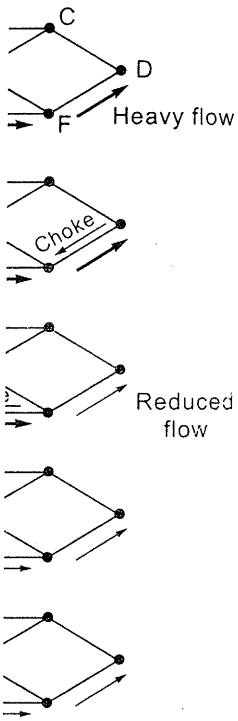
Dec. 06 [Q. 6(b)] *Exp*



The IP header has a 2(

n for a specified
rated due to the
cket reached the
eives the choke
traffic from the

to be congested.
A to reduce the
: be reduced.



packet that affects
sses through.

5.8 Load Shedding

When a router receives more packets than it can handle, the packets are simply discarded.

Should old or new packets be discarded?

It depends on the application :

(1) File Transfer

- In file transfer applications the new packet is discarded.
- If an old packet is discarded, a gap is created in the file and all the packets from this gap have to be retransmitted.

(2) Multimedia

- In multimedia applications the old packets are discarded.
- In multimedia consecutive packets are not very different; therefore we should retain the newer packets as they contain more up-to-date information.

5.9 Quality of Service

Quality of service in the Network Layer Basically consists of : Traffic Shaping (Refer section 5.6 [II]).

5.10 The Network Layer in Internet

Dec. 05 [Q. 4(c)] *Describe classification of IP addresses in IPv4.* **(5 M)**

May 06 [Q. 1(a)] *List the special IP addresses.* **(5 M)**

Dec. 06 [Q. 6(b)] *Explain IP address.* **(2 M)**

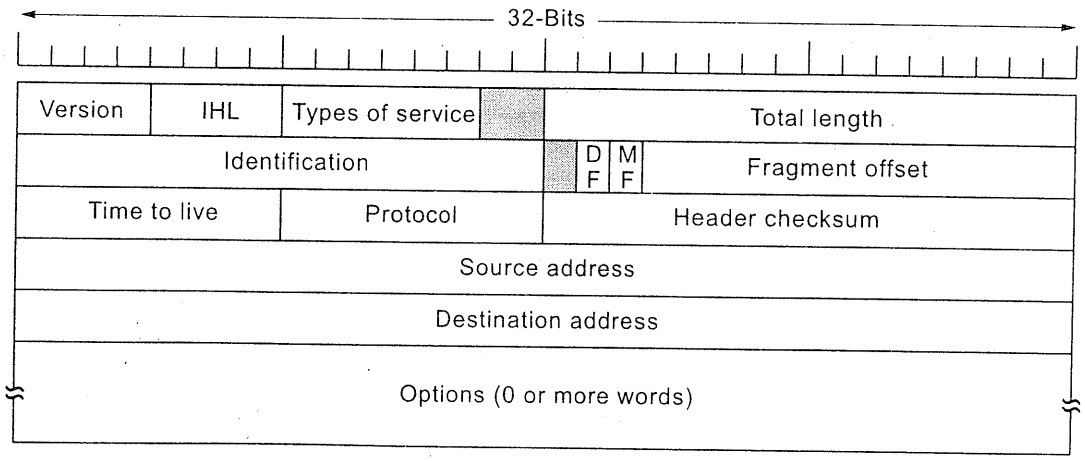


Fig. 5.16 : The IPv4 (Internet Protocol) header.

The IP header has a 20-byte fixed part followed by a variable length optional part.

(1) *Version (Length 4-bits)*

This field indicates the IP-version used for this packet. Typically 4.

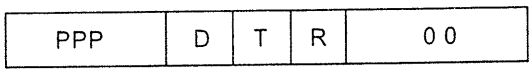
(2) *IHL : Internet Header Length (Length 4-bits)*

This indicates the length of the header. Minimum length of header = 20-bytes.

Minimum value of this field = 20-bytes/ 32-bits = 5. (20 bytes = Minimum length of header and 32-bits because header format is in terms of 32-bit words.)

(3) *Type of Service (Length 8-bits)*

These are rarely used. If one or more of these bits are set, they indicate how routers should handle this packet.



PPP = Precedence
D = Delay Attributes
T = Throughput Attributes
R = Reliability Attributes

Fig. 5.17

- *Precedence* is used to assign a level of priority to a data unit.
- The *delay* field indicates whether a delay should be applied when sending the packet.
- The *throughput* field can be used to indicate that high throughput should be used with the particular data unit.
- The *reliability* field is used to indicate whether or not this data unit requires high reliability or normal service.

(4) *Total length of packet (Length 16-bits)*

It gives the total length of the packet, including this header and including the data sent. Maximum total length = $2^{16} \Rightarrow 0$ to 65535.

(5) *Identification (Length 16-bits)*

A number identifying this packet. Numbering packets is useful when fragmenting packets. All the fragments of the same packet have the same identification number.

(6) *Flags (Length 3-bits)*

(i) The first of these bits is reserved for future use. For now, it should be set to 0.

(ii) **DF (Do Not Fragment)** : The second bit indicates whether this packet may be fragmented by the router (0) or not (1).

(iii) MF (More
last fragme

(7) *Fragmentation O*

- If the total
divided into
- All these fra
packet 100 b
maximum si
first will be
fragment). T
Furthermore
fragment of
original pack

(8) *Time to Live (Len*

- This is used
eternity.
- Each router
value.
- If a router re
send a messa
- Max value wi

(9) *Protocol (Length 8*

The protocol use

(10) *Header Checksum(*

- Before sendin
- The receiver
receiver can t

(11) *Source Address (Le*

The *IP-address* of

(12) *Destination(Length*

The *IP-address* of

(13) *Options (optional)*

If required, route

(iii) MF (More Fragment) : The third bit tells the receiver whether this was the last fragment of the packet (0) or not (1).

(7) *Fragmentation Offset. (Length 13-bits)*

- If the total length of the packet is too large for a network to handle, it is divided into smaller fragments.
- All these fragments have the same identification number. If, for example, a packet 100 bytes large is sent, but the network can only handle packets with a maximum size of 50 bytes, the original packet is fragmented in 2 others: the first will be 50 bytes large and will have a fragmentation offset of 0 (first fragment). The second will be 50 bytes long, and will have an offset of 50. Furthermore, it will have set its third Flag, telling the receiver this is the last fragment of the packet. Now, the receiver can completely reconstruct the original packet.

(8) *Time to Live (Length 8-bits)*

- This is used to make sure no packet will wander through the internet for eternity.
- Each router that handles this packet, will deduct 1 from the Time to Live value.
- If a router receives a packet with a TTL of 0, it will discard the packet, and send a message to the source indicating that the TTL-value has reached zero.
- Max value within TTL= $2^8 - 1 = 255$.

(9) *Protocol (Length 8-bits)*

The protocol used in the packet. Typically 06 for TCP or 17 for UDP.

(10) *Header Checksum(Length 16-bits)*

- Before sending, the sender calculates a checksum.
- The receiver calculates this checksum again - if the value was changed, the receiver can tell that the packet was damaged during transit.

(11) *Source Address (Length 32-bits)*

The *IP-address* of the sender of this packet.

(12) *Destination(Length 32-bits)*

The *IP-address* of the intended receiver.

(13) *Options (optional)*

If required, routers or gateways can define custom options here.

IP-addresses

- An IP address is a 32-bit binary number that contains two separate pieces of information :

  (a) *Network identifier* : Identifies the network (a group of computers).

  (b) *Host identifier* : Identifies a specific computer on the network.

  While computers work with IP addresses as 32-bit binary values, humans normally use the dotted-decimal notation. A binary address and its dotted-decimal equivalent are shown below. Note that the 32-bit address is divided into four 8-bit fields called octets.

  11000000.10101000.00001010.00000101 = 192.168.10.5

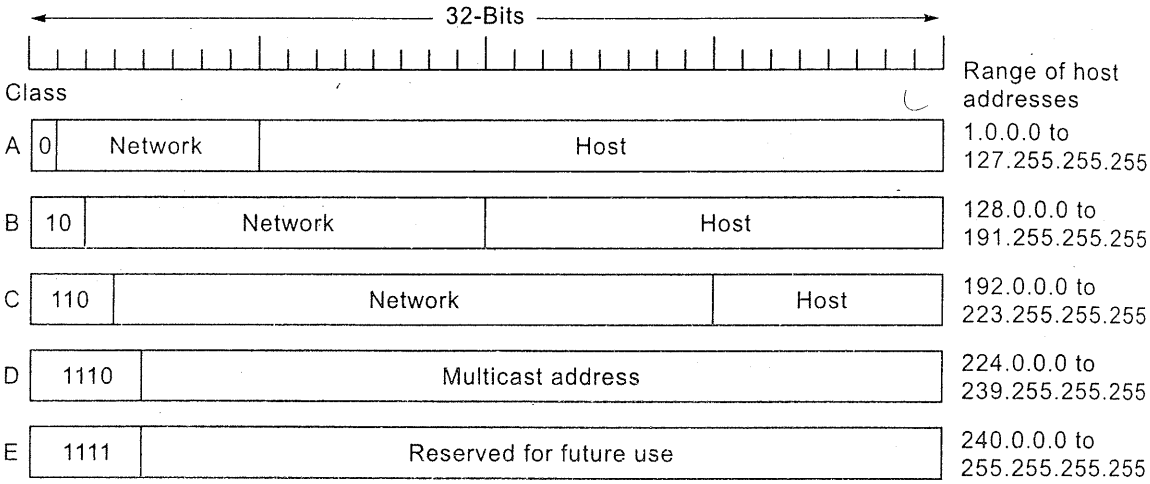


Fig. 5.18 : IP address formats.

Class A

- It is identified by the first bit set as 0.
- The next 7-bits is the *network identifier*, and the remaining 24-bits identify *hosts*.
- Network number 127 is reserved for loopback testing. Address range 127.0.0.0 to 127.255.255.255 are used for loopback testing.
- Total address range is from 1.0.0.0 to 127.255.255.255
- 24 bits are used to identify hosts. Therefore the Maximum Number of hosts per network = $2^{24} - 2 = 16777214$ (one address is reserved for broadcast and one for network therefore we subtract 2)

Class B

- Identified by the first 2-bits set as 10.
- The next 14-bits is the *network identifier*, and the remaining 16-bits identify *hosts*.
- Address Range 128.0.0.0 to 191.255.255.255

- 16 bits are u
  per network

Class C

- Identified by
- The next 21
  hosts.
- Address Ran
- 8-bits are us
  per network

Class D

- A class D sch
- The first 4-bi
  group of hos

Class E

- A class E was

Special IP Addresses

0 0 0 0 0 0 0 0 0 0 0 0

0 0 . . .

1 1 1 1 1 1 1 1 1 1 1 1

Network

127

- The value 0 mea
  are being booted.
- IP addresses wit
  addresses allow
  number.
- The address consi
- The addresses wi
  machines to send
- Finally, all addres
  Packets sent to th
  locally and treate
  local network wit

- 16 bits are used to identify hosts. Therefore the Maximum Number of hosts per network = $2^{16} - 2 = 65{,}534$.

lass C

- Identified by the first 3-bits set as 110.
- The next 21-bits is the *network identifier*, and the remaining 8-bits identify *hosts*.
- Address Range 192.0.0.0 to 223.255.255.255
- 8-bits are used to identify hosts. Therefore the Maximum Number of hosts per network = $2^8 - 2 = 254$.

lass D

- A class D scheme also exists for multicasting.
- The first 4-bits (1110) identify the class, and the remaining 28-bits refer to a group of hosts, all of which receive the same multicast message.

lass E

- A class E was also defined for future use, with the first four bits being 1111.

pecial IP Addresses

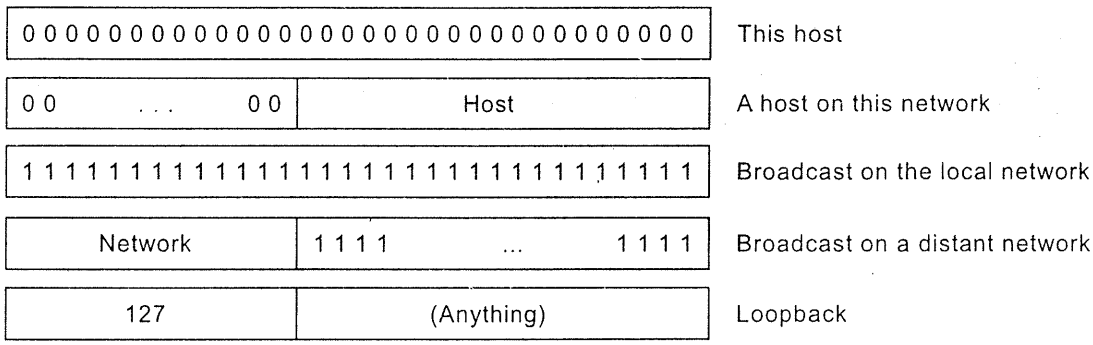


Fig. 5.19 : Special IP addresses.

a) The value 0 means this host. The IP address 0.0.0.0 is used by hosts when they are being booted.

b) IP addresses with 0 as network number refer to the current network. These addresses allow machines to refer to their own network without knowing its number.

c) The address consisting of all 1s allows broadcasting on the local network.

d) The addresses with a proper network number and all 1s in the host field allow machines to send broadcast packets to distant LANs anywhere in the Internet.

e) Finally, all addresses of the form 127.xx.yy.zz are reserved for loopback testing. Packets sent to that address are not put out onto the wire; they are processed locally and treated as incoming packets. This allows packets to be sent to the local network without the sender knowing its number.

5.11 Subnetting

Dec. 06 [Q. 5(a)] *Explain subnetting and masking with suitable examples.* **(10 M)**

May 07 [Q. 1(b)] *Explain subnetting.* **(10 M)**

Subnet Mask

A subnet mask is an IP address feature that indicates which bits in the IP address define the network and which bits define the host. The standard subnet masks used for the class A, B, and C networks are shown in the following table, along with the binary equivalent :

Class	Subnet Mask (Decimal)	Subnet Mask (Binary)
Class A	255.0.0.0	11111111 00000000 00000000 00000000
Class B	255.255.0.0	11111111 11111111 00000000 00000000
Class C	255.255.255.0	11111111 11111111 11111111 00000000

Note how the binary 1s indicate the bits that are used for the network address portion of the IP address. They essentially "mask out" the network address to reveal the host address.

As an example, a class B address of 128.10.50.25 and a class B subnet mask of 255.255.0.0 are shown in the following table. The mask indicates that the first two bytes are the network address, so the last two bytes are the host address.

Class B address	128.10.50.25	10000000 00001010 00110010 00011001
Class B subnet mask	255.255.0.0	11111111 11111111 00000000 00000000

Note : (IP address) BOOLEAN AND (SUBNET MASK) = NETWORK ADDRESS

Subnetting

- A subnet is a logical subsection of an IP network. *Subnetting is used to divide a large network into two or more smaller networks that are easier to manage.*
- You create subnets to separate groups of hosts for security reasons, for traffic control purposes, or other reasons.
- A service provider with a large block of IP addresses creates subnets so that it can allocate blocks of IP addresses to subscribers.
- Just like networks on the Internet, routers are required at subnet boundaries to transmit packets from one subnet to another.
- In terms of addressing, subnetting is equivalent to adding a third level to the Internet addressing hierarchy. The normal two levels are the *network level* and the *host level.* Subnetting creates a sublevel of networks within each network.
- The key to subnetting is to use some of the host bits for the subnet address.

The two-pa
<network i
Class B ad
1 0
Class B a
1 0
Fig
The subnet ide
subnets exist (e
four subnets ar
address 192.168
11000000.101
To subnet this a
many subnetwo
the third octet
decimal number
192.168.0
192.168.3
192.168.6
192.168.9
192.168.1
192.168.1
192.168.1
192.168.2
Note : When we use on
use subnetting
Two level clas
Three level clas

The two-part IP address becomes a three-part IP address as shown here:

<network identifier> <subnet identifier> <host identifier>

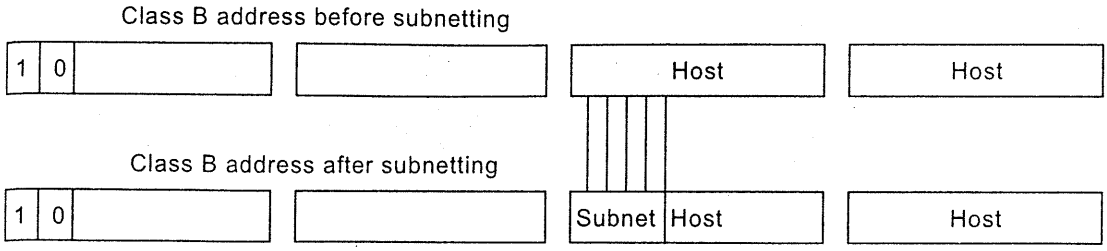


Fig. 5.20 : Subnetting in Class B using 5 subnetting bits.

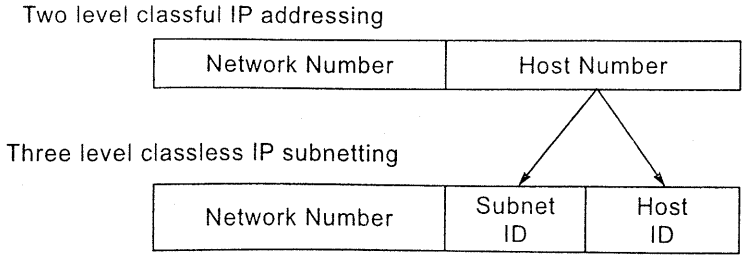
The subnet identifier must be at least 1 bit. A 1-bit value means that only two subnets exist (either the bit is 0 or 1). If the subnet identifier is two bits, then four subnets are possible (00, 01, 10, 11). Assume you are using the IP network address 192.168.0.0. The bit pattern for this address is :

11000000.10101000.00000000.00000000

To subnet this address, you use one or more of the host bits, depending on how many subnetworks are needed. In the table given below, the underlined bits in the third octet are used for subnetting. A 3-bit value provides eight possible decimal numbers as shown, giving eight subnets (subnets 0, 32, 64, 96, etc.).

192.168.0.0	11000000.10101000.<u>000</u>00000.00000000
192.168.32.0	11000000.10101000.<u>001</u>00000.00000000
192.168.64.0	11000000.10101000.<u>010</u>00000.00000000
192.168.96.0	11000000.10101000.<u>011</u>00000.00000000
192.168.128.0	11000000.10101000.<u>100</u>00000.00000000
192.168.160.0	11000000.10101000.<u>101</u>00000.00000000
192.168.192.0	11000000.10101000.<u>110</u>00000.00000000
192.168.224.0	11000000.10101000.<u>111</u>00000.00000000

Note : When we use only the classes A,B,C the addresses are called Classful addresses. When we use subnetting the addresses are called Classless addresses.)



5.12 Routing in the Internet

Dec. 04 [Q. 7(a)] *How the Routing done in the Internet ? Explain one Interior Gateway Routing Protocol. (Ans. : OSPF)* **(10 M)**

What is an Autonomous System ?

- An Autonomous System (AS) is either a single network or a group of networks that is controlled by a common network administrator. An autonomous system is assigned a globally unique number, sometimes called an Autonomous System Number (ASN).
- Networks within an autonomous system communicate routing information to each other using an Interior Gateway Protocol (IGP).
- An autonomous system shares routing information with other autonomous systems using the Border Gateway Protocol (BGP). Previously, the Exterior Gateway Protocol (EGP) was used.

Internet Control Protocols

May 04 [Q. 6(a)] *Describe Address Resolution Protocol. What are difficulties for having a mobile IP ? What can be the solution ?* **(10 M)**

Dec. 04 [Q. 7(b)] *ARP and RARP both map addresses from one space to another. In this respect they are similar, In what way do they differ ?* **(4 M)**

May 06 [Q. 1(b)] *Explain the working of internetworking in terms of IP and MAC addresses.* **(10 M)**

(Ans. : ARP and RARP)

The internet control protocols used in the network layer are :

(1) ICMP

Internet Control Message Protocol (ICMP) is a required protocol tightly integrated with IP. ICMP messages are delivered in IP packets. Some of ICMP's functions are to :

- Announce network errors, such as a host or entire portion of the network being unreachable, due to some type of failure. A TCP or UDP packet directed at a port number with no receiver attached is also reported via ICMP.
- Announce network congestion. When a router begins buffering too many packets, due to an inability to transmit them as fast as they are being received, it will generate ICMP Source Quench messages. Directed at the sender, these messages should cause the rate of packet transmission to be slowed.

- Assist Troubl
  packet on a
  times and co
- Announce T
  discarding th
  fact.

(2) ARP - Address I

- It is used fo
  address to the
- If A has the I
  wants its phy
  which it is c
  having the IP
  receive the A
  B responds by

A

A

(3) RARP-Reverse Ad

- A machine al
  address. It has
- E.g. Suppose
  which contain
  and respond v

- Assist Troubleshooting. ICMP supports an Echo function, which just sends a packet on a round-trip between two hosts. It measures average round-trip times and computing loss percentages.
- Announce Timeouts. If an IP packet's TTL field drops to zero, the router discarding the packet will often generate an ICMP packet announcing this fact.

(2) ARP - Address Resolution Protocol

- It is used for logical to physical address conversion i.e it converts the *IP address* to the *Physical address*.
- If A has the IP address of B (let us assume that the IP address of B is IB) and wants its physical address then A will send an ARP packet to all the hosts to which it is connected. The ARP message is a request from A to the host having the IP address IB to return its physical address. Now all the hosts will receive the ARP message from A but only B will recognize its IP address 'IB'. B responds by sending its physical address(PB).

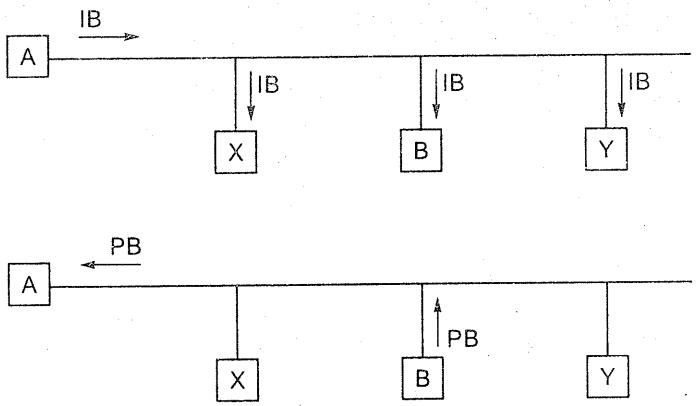


Fig. 5.21 : ARP.

(3) RARP-Reverse Address Resolution Protocol

- A machine always knows its *Physical address* but it may not know its *IP address*. It has to obtain its IP address by contacting a server.
- E.g. Suppose A wants to find its IP address, A will broadcast a RARP request which contains its physical address (PA). The server will accept the request and respond with the IP address of A (IA).

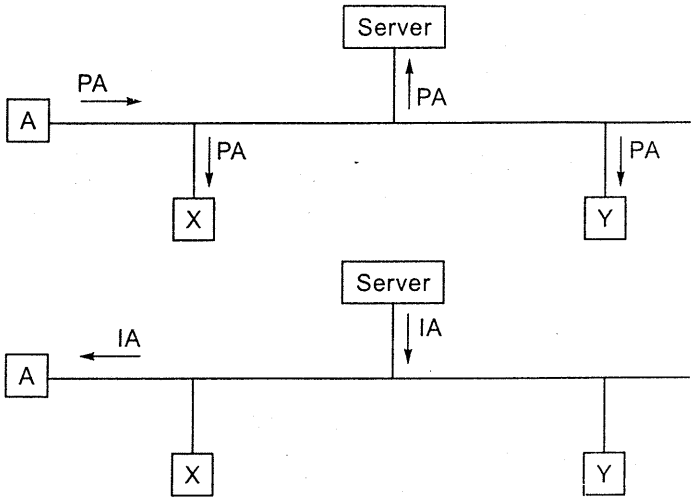


Fig. 5.22 : RARP.

(4) BOOTP (Bootstrap Protocol)

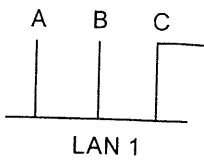
- Allows a diskless client machine to discover its own IP address, the address of a server host, and the name of a file to be loaded into memory and executed.

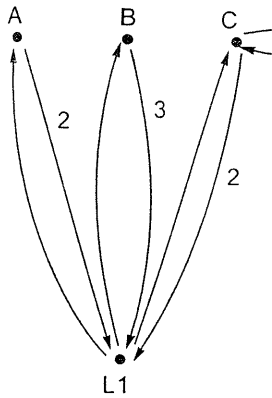
(5) DHCP (Dynamic Host Configuration Protocol)

- It is used to control vital networking parameters of hosts with the help of a server.
- DHCP is backward compatible with Bootstrap protocol.

(6) OSPF (Open Shortest Path First) : An Interior Gateway Routing Protocol.

- The original Internet interior gateway protocol was a distance vector protocol based on the Bellmann-Ford algorithm. It worked inefficiently as ASes got larger. It also suffered from the count-to-infinity problem. In 1988, the Internet Engineering Task force began work on its successor called OSPF(Open Shortest Path First) which became a standard in 1990.
- OSPF supports three kinds of connections and networks :
  - Point-to-point lines between exactly two routers.
  - Multiaccess networks with broadcasting.
  - Multiaccess networks without broadcasting.
- OSPF abstracts the networks, routers, and lines into a directed graph in which each arc is assigned a cost. It then computes the shortest path based on the weights of the arcs. Below figure represents the graphical representation network. *Figure 5.23(a)* shows an AS and *figure 5.23(b)* shows the direct graph of *(a)*.





- OSPF distingu
  - Internal r
  - Area bord
  - Backbone
  - AS bound

Examples of a

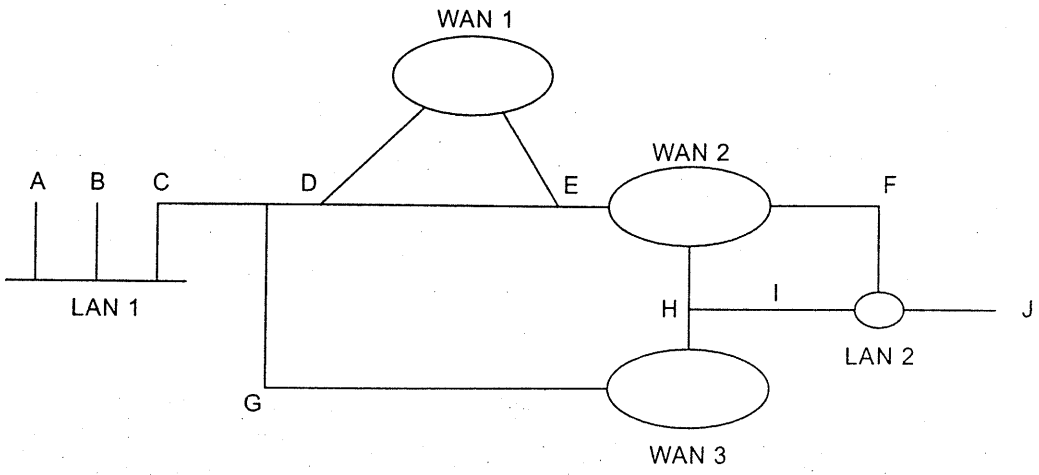


Fig. 5.23(a) : An autonomous system.

lress, the address
nto memory and

vith the help of a

g Protocol.

ce vector protocol
ently as ASes got
oblem. In 1988,
successor called
1990.

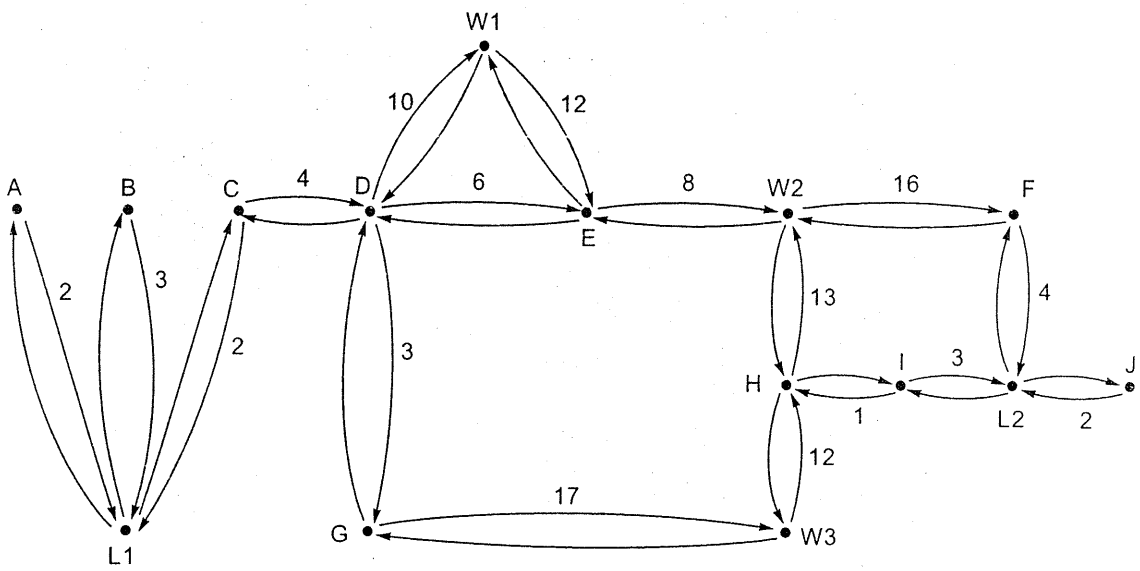


Fig. 5.23(b) : A graph representation of (a).

lirected graph in
ortest path based
s the graphical
ure 5.23(b) shows

- OSPF distinguishes four classes of routers :
  - Internal routers are wholly within one area.
  - Area border routers connect two or more areas.
  - Backbone routers are on the backbone.
  - AS boundary routers talk to routers in other ASes.

  Examples of all four classes of routers are illustrated in *figure 5.24.*

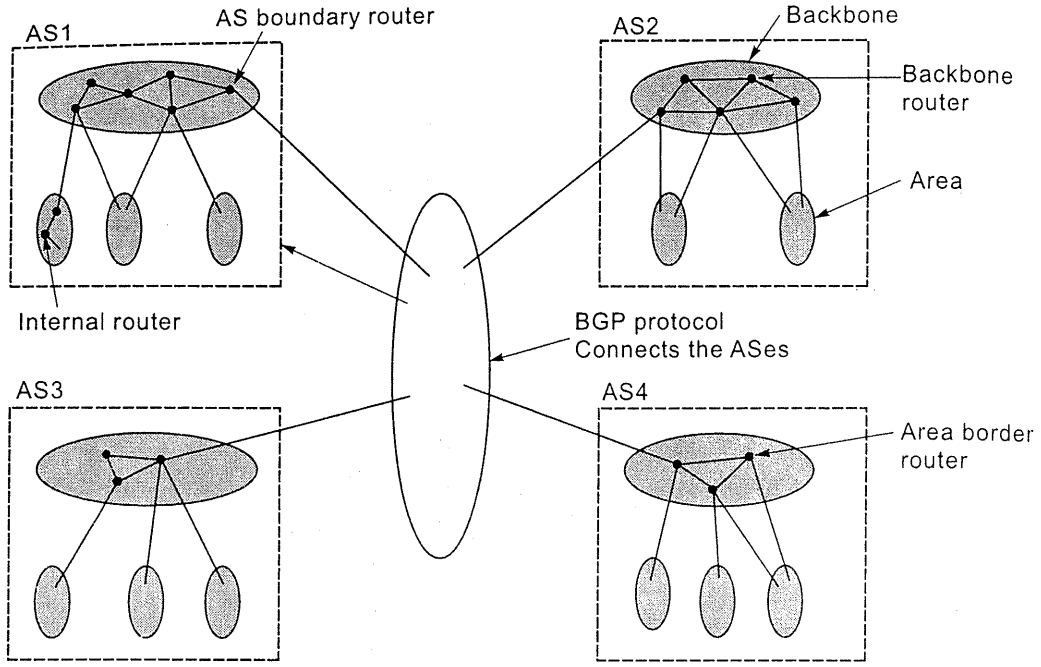


Fig. 5.24 : The relation between ASes, backbones, and areas in OSPF.

- Message Types
  - **(i)** *Hello :* As the name suggests, these messages are used as a form of greeting, to allow a router to discover other adjacent routers on its local links and networks. The messages establish relationships between neighbouring devices (called adjacencies) and communicate key parameters about how OSPF is to be used in the autonomous system or area.
  - **(ii)** *Database Description :* These messages contain descriptions of the topology of the AS or area. That is, they convey the contents of the link-state database (LSDB) for the autonomous system or area from one router to another.
  - **(iii)** *Link State Request :* These messages are used by one router to request updated information about a portion of the LSDB from another router. The message specifies exactly which link(s) about which the requesting device wants more current information.
  - **(iv)** *Link State Update :* These messages contain updated information about the state of certain links on the LSDB. They are sent in response to a Link State Request message.
  - **(v)** *Link State Acknowledgment :* These messages provide reliability to the link-state exchange process, by explicitly acknowledging receipt of a Link State Update message.

(7) Border Gatew

Fig. 5.2

- Inside an A (Border Gat
- Different p ASes becaus
  - (a) All an i as poss about p
  - (b) Exterior AS may and en willing service. their cu are plac involve these pc
- From the po lines connec between a bc
- Pairs of BG connections.
- BGP is fund most others
  - (a) Instead keeps tr each nei

:kbone

Backbone router

Area

Area border router

OSPF.

ed as a form of
uters on its local
nships between
mmunicate key
omous system or

criptions of the
ıtents of the link-
r area from one

router to request
n another router.
ch the requesting

nformation about
in response to a

reliability to the
ging receipt of a

(7) Border Gateway Protocol (BGP) : An Exterior Gateway Routing Protocol.

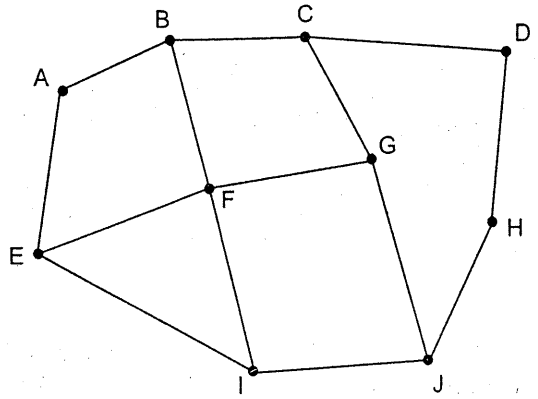


Fig. 5.25(a) : A set of BGP routers.

Information F receives from its neighbours about D

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

Fig. 5.25(b) : Information sent to F.

- Inside an AS the recommended routing protocol is OSPF. Between ASes, BGP (Border Gateway Protocol) is used.
- Different protocols are needed for routing in an AS and routing between ASes because :
  (a) All an interior gateway protocol has to do is move packets as efficiently as possible from the source to the destination. It does not have to worry about politics.
  (b) Exterior gateway protocol routers have to worry about politics. E.g : An AS may be unwilling to carry transit packets originating in a foreign AS and ending in a different foreign AS. On the other hand, it might be willing to carry transit traffic for other foreign ASes that paid it for this service. E.g : Telephone companies might be happy to act as a carrier for their customers, but not for others. These conditions and restrictions that are placed on the exterior routing are called as policies. Typical policies involve political, security, or economic considerations. BGP takes care of these policies.
- From the point of view of a BGP router, the world consists of ASes and the lines connecting them. Two ASes are considered connected if there is a line between a border router in each one.
- Pairs of BGP routers communicate with each other by establishing TCP connections.
- BGP is fundamentally a distance vector protocol, but quite different from most others such as DVR because :
  (a) Instead of maintaining just the cost to each destination, each BGP router keeps track of the path used. Similarly, instead of periodically giving each neighbor its estimated cost to each possible destination, each BGP

router tells its neighbors the exact path it is using.

As an example, consider the BGP routers shown in *figure 5.25* In particular, consider F's routing table. Suppose that it uses the path FGCD to get to D. When the neighbors give it routing information, they provide their complete paths, as shown in *figure 5.25(b)* (for simplicity, only destination D is shown here).

After all the paths come in from the neighbors, F examines them to see which is the best. It quickly discards the paths from I and E, since these paths pass through F itself. The choice is then between using Band G.

(b) BGP easily solves the count-to-infinity problem. For example, suppose G crashes F then receives routes from its three remaining neighbors. These routes are BCD, IFGCD, and EFGCD. It can see that the two latter routes are pointless, since they pass through F itself. Therefore it chooses FBCD as its new route.

(8) Multicast Protocols

- IP Multicasting uses class D addresses.
- Multicasting provides an efficient way of disseminating data from a sender to a group of receivers.
- Data destined for the receivers in a multicast group is sent to a single multicast address.

(9) Mobile Routing

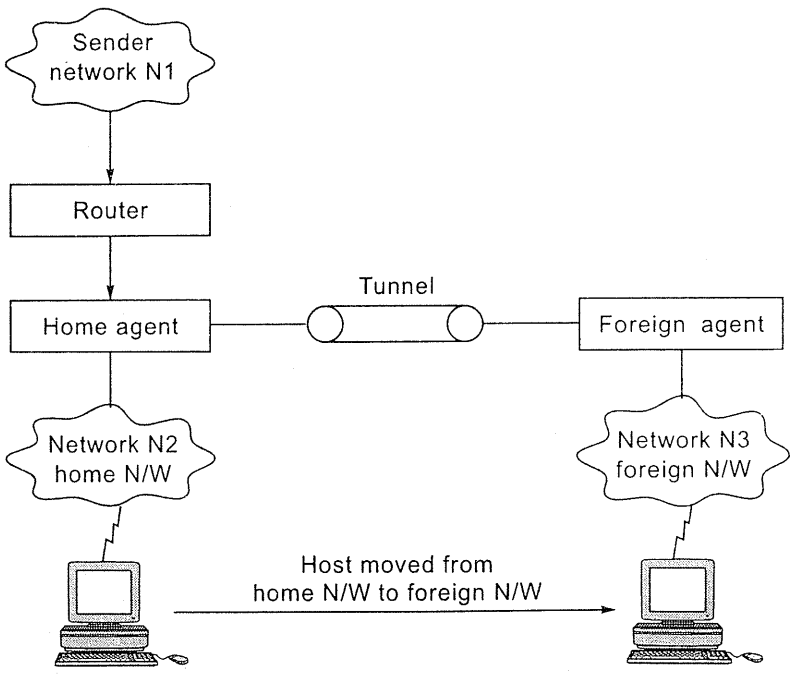


Fig. 5.26 : Mobile routing.

An IP address cor

(a) A network id

(b) A host identif

- Let us ta identifier
- aaa.bbb is
- What will network w
  - Now netwo it will
  - xxx.yy
  - A sim addres addres machi
- The correc
  - A hom *agent.*
  - A fore should
  - *Registr foreign*
  - *Agent* the *ho* the fo addres
  - *Tunnel* node, *foreign*
  - The *ho* asks it

figure 5.25 In
the path FGCD
formation, they
(for simplicity,

nes them to see
d E, since these
ing Band G.
ıple, suppose G
eighbors. These
wo latter routes
t chooses FBCD

rom a sender to

ent to a single

An IP address consists of two parts :

(a) A network identifier which identifies the network.

(b) A host identifier which identifies the host on the network.

- Let us take an IP address aaa.bbb.ccc.ddd, let aaa.bbb be the network identifier and ccc.ddd be the host identifier.
- aaa.bbb is called the *home network* of ccc.ddd
- What will happen if the host is moved from the aaa.bbb network to another network whose identifier is xxx.yyy?
  - Now all the messages for aaa.bbb.ccc.ddd will come to the aaa.bbb network but as the ccc.ddd host is not on the aaa.bbb network anymore it will not get the messages.
  - xxx.yyy is called the *foreign network* of ccc.ddd
  - A simple solution will be to give a machine a totally brand new IP address whenever it moves. But this solution will lead to too many IP addresses being created for the same machine over time and all the other machines on the network will have to be told about this change.
- The correct solution to this problem is:
  - A home network that allows its nodes to be mobile should have a *home agent*.
  - A foreign network that allows nodes from other network to visit it should have a *foreign agent*.
  - *Registration* : When a node visits a foreign network, it must tell the *foreign agent* about its presence.
  - *Agent Discovery* : The *foreign agent* now contacts the *home agent* and tells the *home agent* that a node from the home network is presently visiting the foreign network. The *foreign agent* also sends the *home agent* the IP address of the foreign network. This address is called a *care-of-address*.
  - *Tunneling* : When a message arrives at the home network for the mobile node, the *home agent* re-routes the message to the *care-of-address*. The *foreign agent* receives the message and sends it to the mobile node.
  - The *home agent* gives the sender of the message the *care-of-address* and asks it to send all the future messages directly to the *foreign agent*.

5.13 IPv6

May 06 [Q. 4(a)] *List 10 important features of IPv6 including addressing, flow control,error control,security,mobility,quality of service.* **(10 M)**

IPv4 has served the Internet community well, but its limited address space has caused problems. The next generation of the IP protocol is IPv6, which was designed to resolve most of the problem inherent in IPv4.

Goals of IPv6

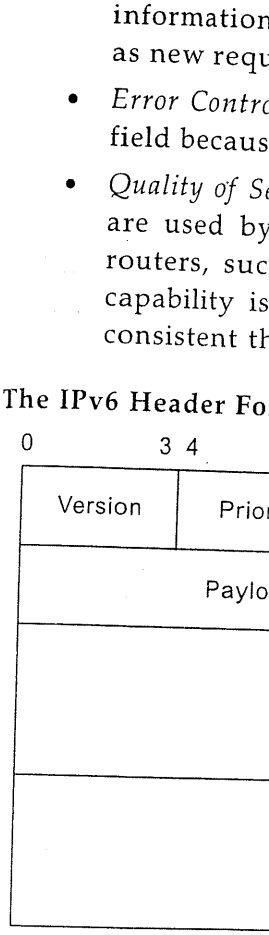
(1) Support billions of hosts, even with inefficient address space allocation.
(2) Reduce the size of the routing tables.
(3) Simplify the protocol, to allow routers to process packets faster.
(4) Provide better security (authentication and privacy) than current IP.
(5) Pay more attention to type of service, particularly for real-time data.
(6) Aid multicasting by allowing scopes to be specified.
(7) Make it possible for a host to roam without changing its address.
(8) Allow the protocol to evolve in the future.
(9) Permit the old and new protocols to coexist for years.

Features of IPv6

- *Addressing* : The most important feature of IPv6 is its longer 128-bit address space, compared to 32-bits for IPv4. While IPv4 is limited to four billion (2^{32}) addresses IPv6 supports 3.4×10^{38} (2^{64}) addresses.
- Another major improvement of IPv6 is the *simplification* of the header. It contains only seven fields (versus 13 in IPv4). This change allows routers to process packets faster and thus improve throughput.
- *Security* : It supports authentication, encryption of data, and data integrity. Header authentication can guarantee that a packet is from the real source address.
- Another major improvement was better support for *options*. It is simple for routers to skip over options not intended for them. This feature speeds up packet processing time.
- *Flow Control* : More than 1 flow can be underway from a source to a destination. Uses a flow label field in IPv6 header to fully specify a flow in combination with the source/destination
- Traffic can be *prioritized* across routers.
- IPv6 supports *mobile users*, unlike IPv4, which assumes that users always attach to networks at the same place.
- *Autoconfiguration* reduces system configuration and management.

- IPv6 suppo
  assigned to
  the address
- An *extensi*
  information
  as new requ
- *Error Contro*
  field becaus
- *Quality of Se*
  are used by
  routers, suc
  capability is
  consistent th

The IPv6 Header Fo



Version : 4-bit IP vers

Traffic Class or Prio
desired delivery prior

Flow Label : 24-bit fi
as video messages.

Payload Length : 16-
packet following the I

Next Header : 8-bit s
the IPv6 header. It use

dressing, flow
(10 M)

ess space has
was designed

tion.

P.
a.

128-bit address
our billion (2^{32})

the header. It
lows routers to

data integrity.
the real source

It is simple for
ture speeds up

a source to a
ecify a flow in

it users always

nt.

- IPv6 supports *expanded multicasting and anycasting*. An anycast address may be assigned to multiple devices that provide the same service. Packets sent to the address will go to the device that is closest or most available.
- An *extension scheme* is also included so that senders can add custom information into a datagram. This will allow flexible expansion of the design as new requirements appear.
- *Error Control :* IPv4 has a checksum field for error control. IPv6 has no such field because the same error control function is offered by TCP and UDP.
- *Quality of Service :* The Flow Label and the Priority fields in the IPv6 header are used by a host to identify packets that need special handling by IPv6 routers, such as non-default quality of service or "real-time" service. This capability is important to support applications that require some degree of consistent throughput, delay, and jitter.

The IPv6 Header Format

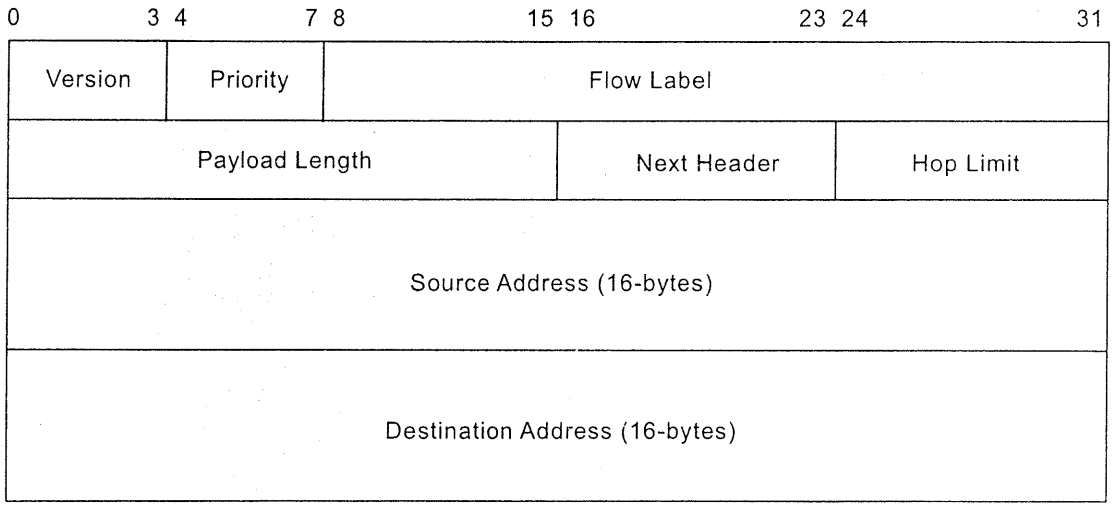


Fig. 5.27 : IPv6 header.

Version : 4-bit IP version number = 6.

Traffic Class or Priority : 4-bit priority value. It enables a source to identify the desired delivery priority of its packets.

Flow Label : 24-bit field. It is used to give packets some special type of traffic, such as video messages.

Payload Length : 16-bit unsigned integer. Length of payload, i.e., the rest of the packet following the IPv6 header, in octets.

Next Header : 8-bit selector. It identifies the type of header immediately following the IPv6 header. It uses the same values as the IPv4 Protocol Field.

Hop Limit : 8-bit unsigned integer. Decremented by 1 by each node that forwards the packet. The packet will be discarded if Hop Limit is decremented to zero. Similar to IPv4 which contained time-to-live in seconds.

Source Address : 128-bits. The address of the sender of the packet.

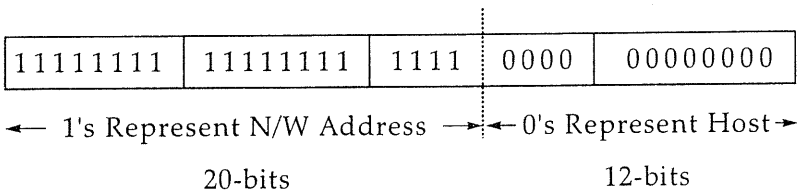
Destination Address : 128-bits. The address of the intended recipient of the packet.

5.14 Exam Questions

Dec. 04 [Q. 1(a)] *A network on the internet has a subnet mask of 255. 255. 240. 0. What is the maximum number of hosts it can handle ?* **(2 M)**

May 05 [Q. 1(a)] *A class A network on the internet has a subnet mask of 255. 255. 244. 0. What is the maximum number of hosts per subnet ?* **(4 M)**

Subnet Mask : 255. 255. 240. 0 which is

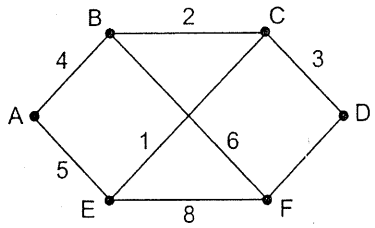


Maximum number of hosts = $2^n - 2$

where n is number of bits in host part

∴ Maximum number of hosts = $2^{12} - 2$ = 4094 hosts.

Dec. 04 [Q. 5(b)]



Consider the subnet shown above. Distance vector routing is used and the following vectors have just come to router C.

From B : (5, 0, 8, 12, 6, 2)

From D : (16, 12, 6, 0, 9, 10)

From E : (7, 6, 3, 9, 0, 4)

Measure delays to B, D and E are 6, 3, and 5 respectively. What is C's new routing table ? Give both the outgoing line to use and the expected delay. **(10 M)**

Solution :

A	
B	
C	
D	
E	
F	
	CB

Working f

CBA = C

CDA = C

CEA = C

As CBA is least we e

that forwards the
to zero. Similar to

nt of the packet.

---

255. 255. 240. 0.
(2 M)

k of 255. 255. 244.
(4 M)

]

t→

d and the following

s new routing table ?
(10 M)

Solution :

	B	D	E	C	
				Delay	Outgoing line
A	5	16	7	11	B
B	0	12	6	6	B
C	8	6	3	0	-
D	12	0	9	3	D
E	6	9	0	5	E
F	2	10	4	8	B
	CB = 6	CD = 3	CE = 5		

Working for A

CBA = CB + BA = 6 + 5 = 11

CDA = CD + DA = 3 + 16 = 19

CEA = CE + EA = 5 + 7 = 12

As CBA is least we enter delay = 11 and outgoing line as B in the routing table.

6 TRANSPORT LAYER

The transport layer is the heart of the whole protocol hierarchy. Fundamentals of the transport services, sockets, addressing, TCP and UDP have been covered in this chapter.

Marks	
Dec. 03	40 M
May 04	10 M
Dec. 04	20 M
May 05	10 M
Dec. 05	12 M
May 06	20 M
Dec. 06	7 M
May 07	20 M

6.1 The Transport Service

Dec. 05 [Q. 2(a)] *Discuss the services offered by the transport layer.* **(8 M)**

Dec. 06 [Q. 7(5)] *Write short note on QoS in transport layer. (Also section 6.4.8)* **(5 M)**

May 07 [Q. 5(b)] *Discuss various services offered by the transport layer.* **(10 M)**

6.1.1 Services Provided to the Upper Layers

- The *purpose* of the transport layer is to provide efficient, reliable, cost-effective data transport to its users without depending on the type of physical network currently in use. The users of the transport services are processes present in the application layer. To achieve this goal the transport layer makes use of services provided by the network layer.
- The Hardware/Software in the transport layer that provides the services is called as the *Transport Entity*.

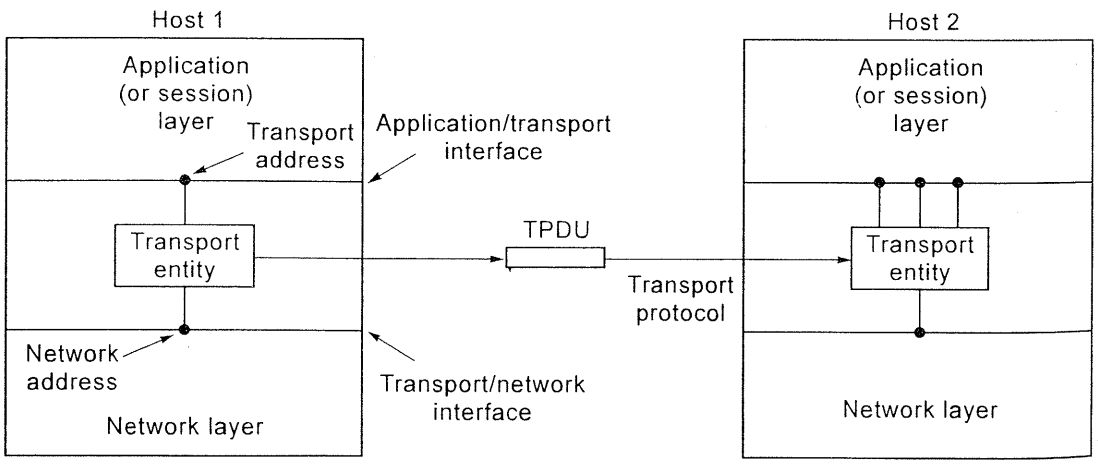


Fig. 6.1 : The network, transport, and application layers.

- The relationsl shown in the
- There are tv *connectionless.* connectionless *"If the services layer ?"* The machine but t layer code rur network layer, network layer the network la
- The transport Lost packets ca
- The network s transport layer implemented a network service
- As the transpor programmers c programs will w
- Due to the abov
  (a) *Transport S* (Physical to
  (b) *Transport S* Application

1.2 Transport Service

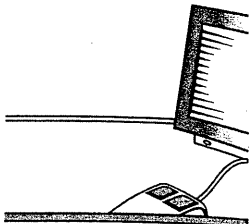
We have seen th nsport service is relia

rminology :

DU : *Transport Proto* to another tra

ckets: Messages that layer on anoth

mes : Messages that layer on anoth



Marks

Dec. 03 :	40 M
May 04 :	10 M
Dec. 04 :	20 M
May 05 :	10 M
Dec. 05 :	12 M
May 06 :	20 M
Dec. 06 :	7 M
May 07 :	20 M

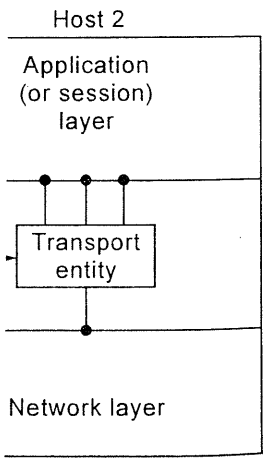
(8 M)

ction 6.4.8) **(5 M)**

'er. **(10 M)**

ent, reliable, cost-
g on the type of
nsport services are
goal the transport

des the services is



rs.

- The relationship between the network, transport and application layers is shown in the *figure 6.1.*
- There are two types of transport services - *connection oriented* and *connectionless*. These services are similar to the connection oriented and connectionless services provided by the network layer. The question arises: *"If the services in both the layers are the same then what is the use of the transport layer ?"* The answer is that the transport layer code runs on the users' machine but the network layer code runs on the routers. As the network layer code runs on routers the users don't have any real control over the network layer, and this is why the users cannot do anything to improve the network layer performance. This is why the transport layer is put on top of the network layer so that it can improve the quality of service.
- The transport service is more reliable than the underlying network service. Lost packets can be detected and compensated for by the transport layer.
- The network service calls may be different from network to network. The transport layer instructions (also called the *transport layer primitives*) are implemented as calls to library procedures to make them independent of network service primitives.
- As the transport layer primitives are independent of the network, application programmers can write codes using a standard set of primitives and the programs will work on a wide variety of networks.
- Due to the above reasons the 7 layer hierarchy is divided into 2 sections
  (a) *Transport Service Provider* : This consists of the bottom layers 1 to 4. (Physical to Transport)
  (b) *Transport Service User* : This consists of the top 3 layers. (Session to Application)

1.2 Transport Service Primitives

We have seen that the network service is generally unreliable whereas the nsport service is reliable

rminology :

PDU : *Transport Protocol Data Unit* : Messages that are sent from a transport entity to another transport entity.

ckets: Messages that are sent from a network layer on one machine to a network layer on another machine. They contain TPDUs.

ames : Messages that are sent from a data link layer on one machine to a data link layer on another machine. They contain Packets.

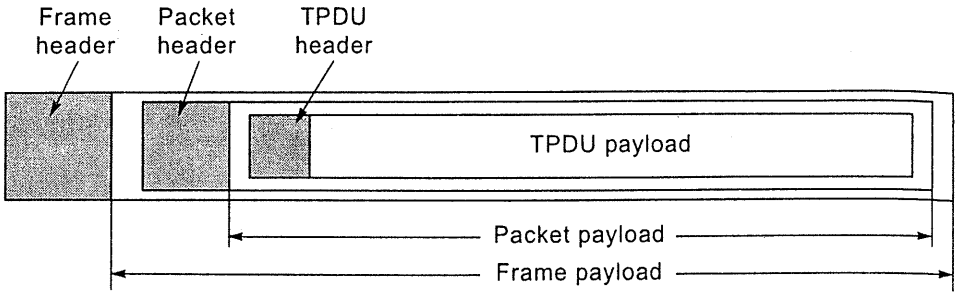


Fig. 6.2 : Nesting of TPDUs, packets, and frames.

Transport Service Primitives :

Consider a system consisting of a server and a number of clients.

(1) **LISTEN** : It is executed by a server. It means that the Server is ready to serve a client. The server is *blocked* until a client requests for its service.

(2) **CONNECT** : When a client wants to talk to a server it executes this primitive. The transport layer carries out this primitive by *blocking* the client and sending a CONNECTION REQUEST TPDU to the server. If the server is blocked on a LISTEN, then the server will be unblocked and a CONNECTION ACCEPTED TPDU is sent back to the client. Now the client will get unblocked and the connection will be established.

(3) **SEND and (4) RECEIVE** : After the connection is established between the client and the server, data can be exchanged using the SEND and RECEIVE primitives.

(5) **DISCONNECT** : When a connection is no longer needed it must be released to free up the resources. This is done using the DISCONNECT primitive.

6.1.3 Sockets

Dec. 06 [Q. 6(b)] *Explain the following term - socket.* **(2 M)**

The primitives that are widely used for Internet programming are :

(1) **SOCKET** : It creates a new end point. It allocates table space for the new end point, within the transport entity. It has to be used by both, the server as well as the client, to create server socket and client socket respectively.

(2) **BIND** : Newly created *server* sockets do not have a network address. The BIND primitive assigns a network address to a server socket. The network address is used so that clients can connect to the *server* socket. The BIND primitive is not required for client sockets.

(3) **LISTEN** : It is used to maintain a queue of the clients who want to connect to the server socket. The difference between this primitive and the LISTEN primitive in the previous section is that, the LISTEN in this case is NOT a *blocking* LISTEN (i.e. the server will not be blocked on executing this LISTEN).

(4) **ACCEPT** : It blocks the server and makes it wait for an incoming connection.

5) **CONNECT** : It
server.

6) **SEND and (7)** R
and the server,

8) **CLOSE** : The co

6.2 Transport Pro

The transport se

6.2.1 Addressing in

Dec. 03 [Q. 6(b)] *Ho*
coming segments.

typical host has
enerate data that i
ansmission. This *m*
estinations.
imilarly, a device's
emultiplexed, so th
hem.

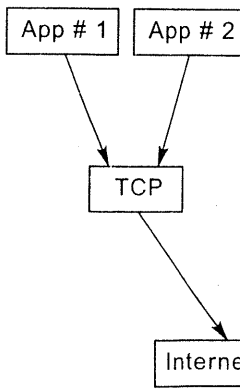
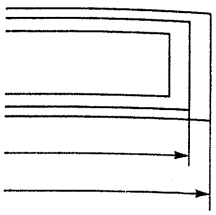


Fig. 6.3



5) **CONNECT** : It blocks the client and tries to establish a connection with the server.

6) **SEND and (7) RECEIVE** : After the connection is established between the client and the server, data can be exchanged using the SEND and RECEIVE primitives.

8) **CLOSE** : The connection is released.

6.2 Transport Protocol Elements

The transport service is implemented by a transport protocol.

6.2.1 Addressing in TCP and UDP

Dec. 03 [Q. 6(b)] *How are the port numbers used by TCP / UDP in demultiplexing incoming segments.* **(10 M)**

A typical host has many application processes running on it. These processes generate data that is sent to either TCP or UDP, which in turn passes it to IP for transmission. This *multiplexed stream of datagrams* is sent out by the IP layer to various destinations.

Similarly, a device's IP layer is receives datagrams, these datagrams need to be demultiplexed, so they end up at the correct process on the device that receives them.

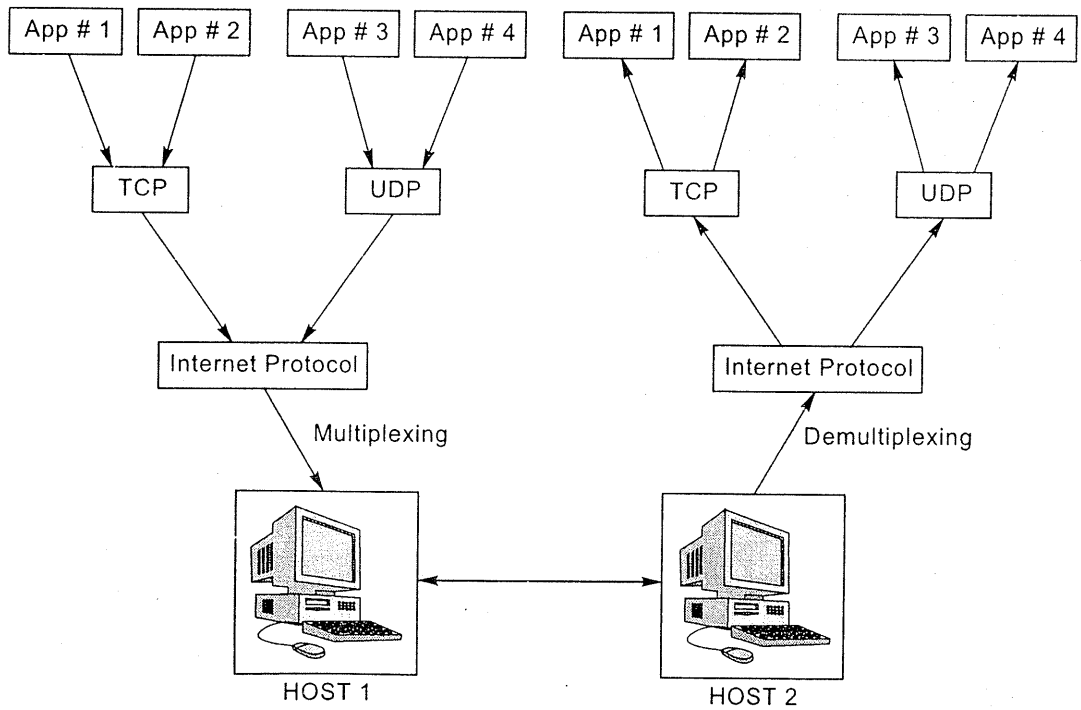


Fig. 6.3 : Multiplexing and Demultiplexing stream of datagrams.

Multiplexing and Demultiplexing Using Ports :

The question is : How do we demultiplex a sequence of IP datagrams that need to go to different application processes on the *same host* ? Let's consider a particular *host* bearing the IP address 24.156.79.20. How does the IP layer know which packet goes to which process on the *same host* ?

The first part of the answer lies in the *Protocol* field included in the header of each IP datagram. This field carries a code that identifies the protocol that sent the data. Since most end-user applications use TCP or UDP at the transport layer, the *Protocol* field in a received datagram tells IP to pass data to either TCP or UDP as appropriate.

But both TCP and UDP are used by many applications at once. This means TCP or UDP must figure out which process to send the data to. To make this possible, an additional addressing element is necessary. This transport layer address is called a *port.*

A port can also be called a TSAP (Transport Service Access Point).
An IP address can also be called a NSAP(Network Service access Point).

Key Concept : TCP/IP transport layer addressing is accomplished using TCP and UDP ports. Each port number within a particular IP device identifies a particular application process.

Source Port and Destination Port Numbers :

In both UDP and TCP messages, two addressing fields appear, for a Source Port and a Destination Port. These are analogous to *the fields for source address and destination address at the IP level,* but at a higher level of detail. While *the source address and destination address at the IP level* identify the source machine and the destination machine; The Source Port and a Destination Port identify the originating process on the source machine, and the destination process on the destination machine.

TCP and UDP port numbers are 16-bits in length.

Summary of Port Numbers for Datagram Transmission and Reception :

So, to summarize, here's the basics of how transport-layer addressing (port addressing) works in TCP and UDP :

- **Sending Datagrams :** An application specifies the source and destination port it wishes to use for the communication. These are encoded into the TCP or UDP header, depending on which transport layer protocol the application is using. When TCP or UDP passes data to IP, IP indicates the protocol type (TCP or UDP) in the Protocol field of the IP datagram. The source and destination p
message, with
- **Receiving Da**
Protocol field
UDP). TCP o
appropriate pr

Note : I should point
TCP/IP. For exa

6.2.2 Connection Esta

[I] Connection Esta

To establish a connect
CONNECTION REQU
REPLY (This is called a
Problems arise when th

Consider the problem

A user establishes a co
transfer money from
terminates the connec
serious problem will oc
at the bank and the ba
repeatedly establish a
someone else's account

To solve this problem w

(a) Normal Procedure

Step 1 : Host 1 send
REQUEST T
sequence numb

Step 2 : Host 2 replies
acknowledging
own initial sequ

Step 3 : Host 1 acknow
sequence numb
data TPDU.

destination port numbers are encapsulated as part of the TCP or UDP message, within the IP datagram's data area.

- **Receiving Datagrams :** The IP software receives the datagram, inspects the Protocol field and decides to which protocol the datagram belongs (TCP or UDP). TCP or UDP receives the datagram and passes its contents to the appropriate process based on the destination port number.

Note : I should point out that the term port has many meanings aside from this one in TCP/IP. For example, a physical outlet in a network device is often called a port.

6.2.2 Connection Establishment and Connection Release

[I] Connection Establishment

To establish a connection, it would seem that one transport entity should send a CONNECTION REQUEST TPDU and should wait for a CONNECTION ACCEPTED REPLY (This is called a 2 Way Handshake). But actually the process is not so simple. Problems arise when the network can loose, store and duplicate packets.

Consider the problem :

A user establishes a connection with a bank and sends messages telling the bank to transfer money from the users account to somebody else's account and then terminates the connection. Now if this message is stored and duplicated then a serious problem will occur. These stored and duplicated messages will keep surfacing at the bank and the bank will consider them as new messages. Thus the bank will repeatedly establish a connection and transfer money from the user account to someone else's account ! This is the problem that exists with the 2 way handshake.

To solve this problem we use the 3 way handshake :

(a) Normal Procedure :

Step 1 : Host 1 sends a CONNECTION REQUEST TPDU containing a sequence number x to Host 2.

Step 2 : Host 2 replies with an ACK TPDU acknowledging x and announcing its own initial sequence number, y.

Step 3 : Host 1 acknowledges Host 2's initial sequence number and sends the first data TPDU.

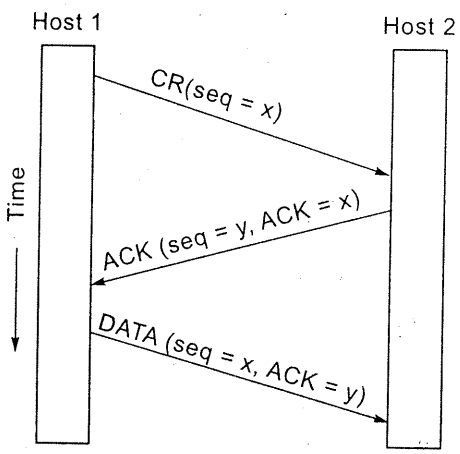


Fig. 6.4 : Normal operation.

(b) 3 Way Handshake Eliminates Duplicate Packets Problem

Step 1 : A duplicate CONNECTION REQUEST TPDU containing a sequence number x is sent to Host 2 from an old connection. Host 2 is not aware that this is a duplicate TPDU.

Step 2 : Host 2 replies with an ACK TPDU acknowledging x and announcing its own initial sequence number, y.

Step 3 : Host 1 realizes that a duplicate packet from an old connection has been received by Host 2. Host 1 rejects the connection.

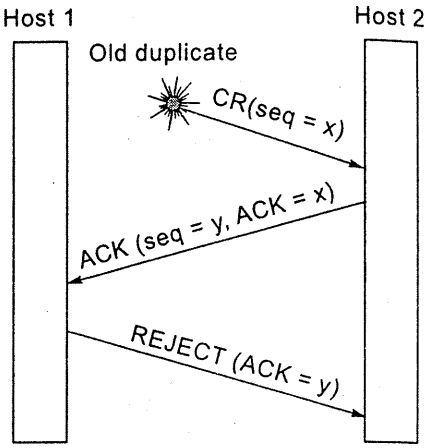


Fig. 6.5 : Old duplicate CONNECTION REQUEST appearing out of nowhere.

(c) 3 Way Handshake Eliminates Duplicate Packets and Duplicate Acknowledgements Problem

Step 1 : As in the previous case Host 2 gets a duplicate CONNECTION REQUEST TPDU and acknowledges it.

Step 2 : Note that Host 2 has proposed using y as the initial sequence number.

Step 3 : There is a delayed data TPDU which has sequence number=x and is acknowledging z. When this TPDU reaches Host 2, Host 2 will reject it because the acknowledgement in the TPDU is for z and not y.

Step 4 : Also Host 1 will realize that a duplicate packet has been received by Host 2 and will thus reject the connection.

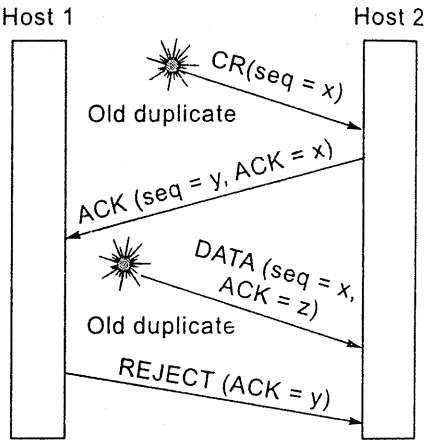


Fig. 6.6 : Duplicate CONNECTION REQUEST and duplicate ACK.

[II] Connection Release

There are two styles of terminating a connection :

(1) **Asymmetric Release** : It means that no more data can be sent or received by the host issuing the release.

(2) **Symmetric Release** : It means that the host issuing the release cannot send any more data but data can be received by it.

Release using the 3 Way Handshake

(a) Normal Case

Step 1 : Host 1 send
REQUEST
connection

Step 2 : When Host
sends a DR
a timer.

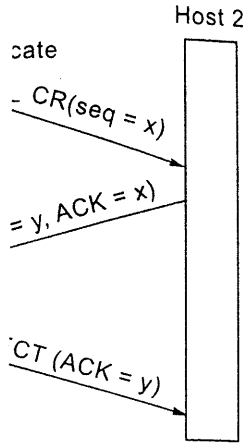
Step 3 : Now Hos
connection
TPDU to H
receives the
also release

(b) ACK is Lost

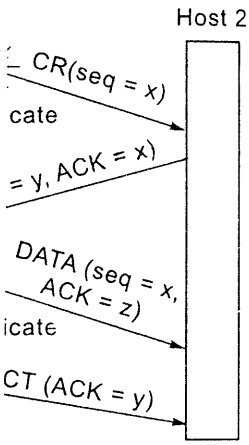
When ACK is lo
on Host 2 time
releases the conn

(c) DR from Host 2

As Host 1 does r
response it time
the DR.



licate CONNECTION ing out of nowhere.



icate CONNECTION id duplicate ACK.

t or received by the

se cannot send any

(a) Normal Case

Step 1 : Host 1 sends DISCONNECTION REQUEST TPDU(DR) to initiate connection release.

Step 2 : When Host 2 receives this it also sends a DR to Host 1 and starts a timer.

Step 3 : Now Host 1 releases the connection and sends an ACK TPDU to Host 2. When Host 2 receives the ACK TPDU it can also release its connection.

(b) ACK is Lost

When ACK is lost, the timer started on Host 2 times out and Host 2 releases the connection anyway.

(c) DR from Host 2 is Lost

As Host 1 does not get the expected response it times out and resends the DR.

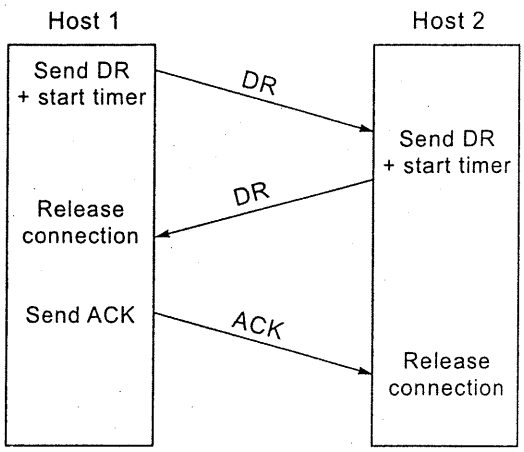


Fig. 6.7 : Normal case of three-way handshake.

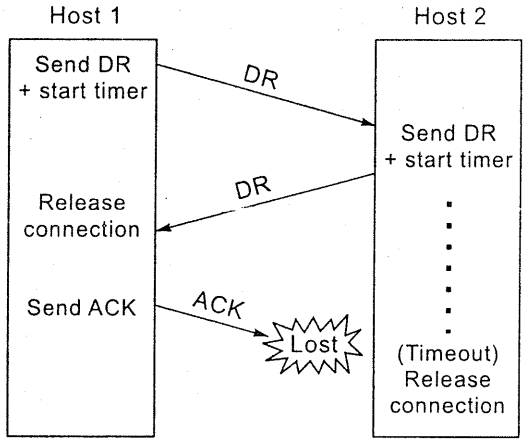


Fig. 6.8 : Final ACK lost.

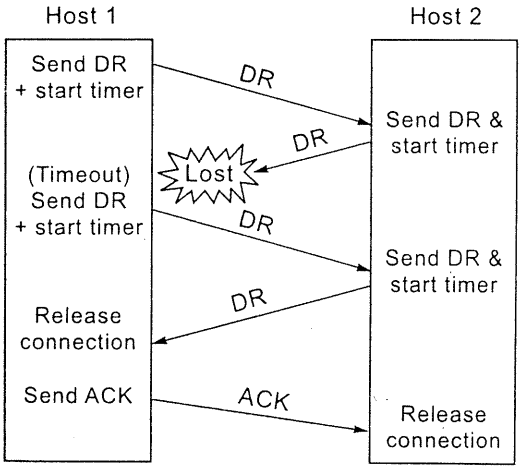


Fig. 6.9 : Response lost.

(d) Response lost and Subsequent DRs lost

In this case after Host 1 times-out N number of times it will give up and simply release the connection. Meanwhile Host 2 will also timeout and release the connection.

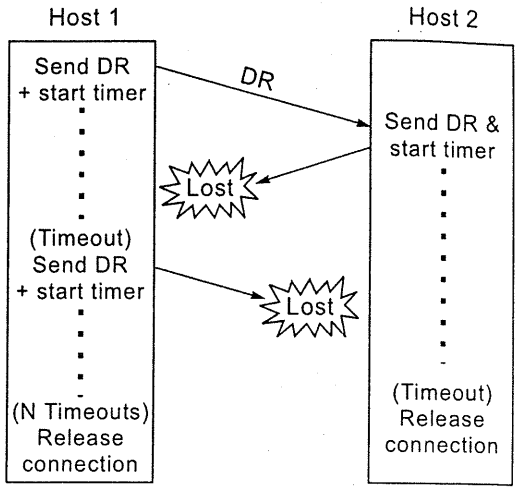


Fig. 6.10 : Response lost and subsequent DRs lost.

6.3 UDP : User Datagram Protocol

May 06 [Q. 5(b)] *Why does UDP exist ? Would it not have been enough to let the user process send raw IP packets ?* **(5 M)**

It is a connectionless protocol. This means that, UDP provides a way for applications to send *segments* without having to establish a connection.

Segment : A segment consists of a header followed by a payload.

The UDP Header

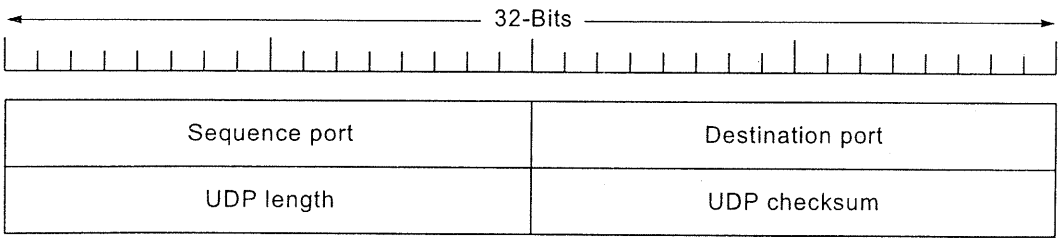


Fig. 6.11 : The UDP header.

- The Source and Destination Ports identify the end processes within the source and destination machines.
- The Length field includes the length of the header and the data.
- The Checksum is optional and is mainly used in error handling.

UDP Over Raw IP

The purpose of using UDP over raw IP is the addition of the source and destination ports. In short, the IP address identifies the machine whereas the port number identifies a particular process on the machine.

Disadvantages

UDP does not do t

- Error contr
- Flow contr
- Retransmis

Applications

UDP is used in cli
the server respond
times-out and tries

E.g. DNS is an appl

6.3.1 Remote Proc

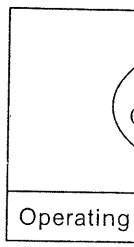


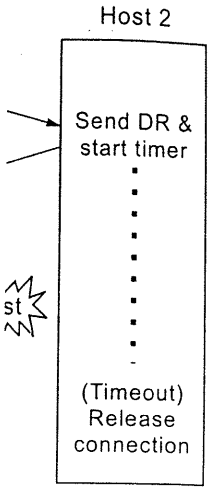
Fig. 6.12 :

The idea is to

To call a remote pro
client stub.

Similarly sever is bo
local.

(1) Client calls clie
(2) Client stub pac
send it.
(3) Client Kernel se
(4) Server Kernel p
(5) Server stub call



Disadvantages

UDP does not do the following :

- Error control.
- Flow control.
- Retransmission of a corrupt segment.

Applications

UDP is used in client-server situations. The client sends a request to the server and the server responds with a reply. If either the request or the reply is lost the client times-out and tries again. There is no need for initial connection setup.

E.g. DNS is an application that uses UDP.

6.3.1 Remote Procedure Call

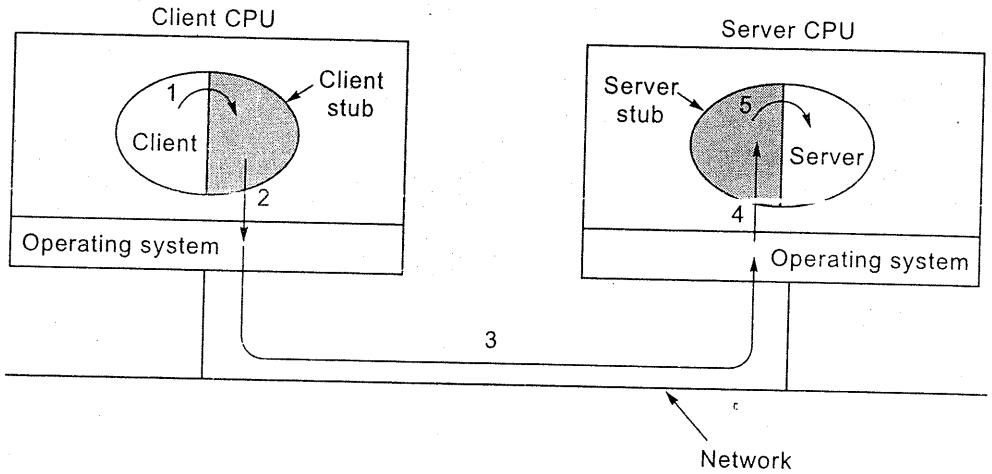


Fig. 6.12 : Steps in making a remote procedure call. The stubs are shaded.

The idea is to make a RPC call look as much like a local procedure call.

To call a remote procedure, client must be bound to a small library procedure called a client stub.

Similarly sever is bound to a server stub. These stubs hide the fact that the call is not local.

(1) Client calls client stub.

(2) Client stub packs parameters into message (marshaling), makes system call to send it.

(3) Client Kernel sends message from client to server.

(4) Server Kernel passes message to server stub.

(5) Server stub calls procedure with un-marshaled parameters.

The reply traces same path in the other direction.

RPC uses UDP when it suffices (e.g., when there are large parameters or results), otherwise it uses TCP.

6.3.2 Real-time Transport Protocol (RTP)

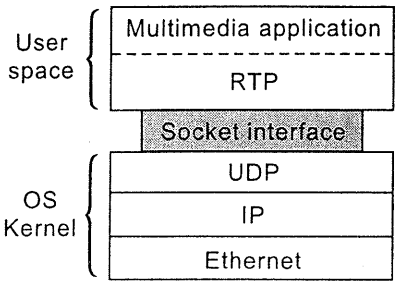


Fig. 6.13(a) : The position of RTP in the protocol stack.

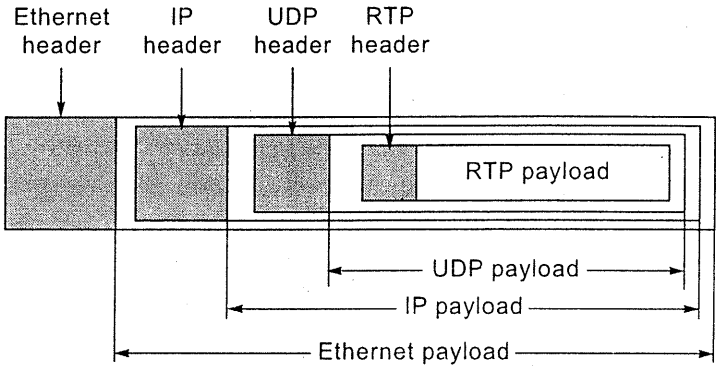


Fig. 6.13(b) : Packet nesting.

- It supports multimedia data transfer over Internet.
- It uses UDP as a transport mechanism.
- Its services include :
  - Payload type identification (type of coding used).
  - Sequence numbering.
  - Timestamping (generation time of packet).
  - Delivery monitoring.
- It supports data transfer to multiple destinations using multicast distribution.
- The packet sequence numbers included in RTP allow the receiver to reconstruct the sender's packet sequence, but sequence numbers might also be used to determine the proper location of a packet, (for example in video decoding packet need not be in sequence.)
- It uses the RTP Control Protocol (RTCP), to monitor the quality of service and to convey information about the participants in an on-going session.
- The *figure 6.13* shows the following :
  (a) Position of RTP in the protocol stack.
  (b) The nesting of the RTP payload.

6.4 TCP : Trans

TCP was des
internetwork.

Why do we say that

Different netwo
bandwidths, dela
dynamically adap
service.

Each machine supp

A *TCP Entity* accep
each piece as a sep
TCP Entity, the *TC*

Functions of TCP

- Retransmit
- Order the d

6.4.1 The TCP Ser

- Both the se
number cor
established
- A socket ma
- Port Numb
reserved for

Port	
21	
23	
25	
69	
79	
80	
110	
119	

- All TCP conn
- TCP does not

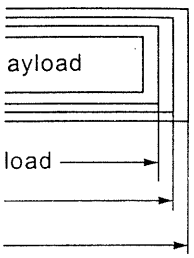
rs or results),

6.4 TCP : Transmission Control Protocol

TCP was designed to provide reliable end-to-end byte stream over an unreliable internetwork.

Why do we say that an Internetwork may be unreliable ?

Different networks within an Internetwork may have different topologies, bandwidths, delays, packet sizes and other parameters. Thus we need TCP to dynamically adapt to the properties of the Internetwork and provide a reliable service.



sting.

Each machine supporting TCP has a *TCP Entity*.

A *TCP Entity* accepts data from local processes, breaks them up into pieces and sends each piece as a separate IP datagram. When the IP datagrams arrive at the receivers *TCP Entity*, the *TCP Entity* reconstructs the original data.

Functions of TCP

- Retransmit corrupt datagrams, if and when needed.
- Order the datagrams into proper sequence at the receiver.

6.4.1 The TCP Services

- Both the sender and the receiver have sockets. Each socket has a socket number consisting of IP address and Port. A connection must be explicitly established between the sender socket and the receiver socket.
- A socket may be used for multiple connections at the same time.
- Port Numbers between 1 and 1024 are called *well known ports* and are reserved for standard services.

Port	Protocol	Use
21	FTP	File transfer
23	Telnet	Remote login
25	SMTP	E-mail
69	TFTP	Trivial file transfer protocol
79	Finger	Lookup information about a user
80	HTTP	World Wide Web
110	POP-3	Remote e-mail access
119	NNTP	USENET news

Fig. 6.14 : Some assigned ports.

- All TCP connections are end to end and full duplex
- TCP does not support multicasting or broadcasting.

- A TCP connection is a byte stream and not a message stream. Message boundaries are not preserved. E.g. If the sending process does four 512 byte writes to the TCP stream, those may be delivered to the receiving process as four 512 byte chunks or two 1024 byte chunks or one 2048 byte chunk.
- **PUSH :** When an application gives data to TCP, TCP may send it immediately or buffer it. If the application wants the data to be sent immediately it can use the PUSH flag in the header.
- **Urgent Data :** To deal with situations where a certain part of a data stream needs to be sent with a higher priority than the rest, TCP *incorporates* an "urgent" function. When critical data needs to be sent, the application signals this to its TCP layer, which transmits it with the URG bit set in the TCP header, bypassing any lower-priority data that may have already been queued for transmission

6.4.2 The TCP Protocol

- Each byte on a TCP connection has its own unique 32-bit sequence number.
- TCP entities exchange data in the form of segments. (A segment consists of the header along with the data bytes)
- TCP entities use the *sliding window protocol* : When a sender transmits a segment, it starts a timer.When a segment arrives at the destination, the receiving TCP entity sends back an acknowledgement segment which contains the next sequence number it expects to receive. If the sender times-out before the acknowledgement is received it retransmits the segment again.

6.4.3 TCP Segment Header

May 04 [Q. 4(b)] *Describe the TCP segment header in detail.* **(10 M)**

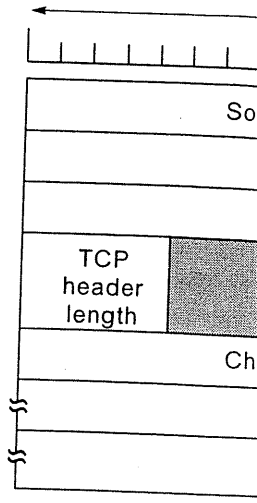
Dec. 04 [Q. 6(b)] *Explain TCP segment header format in detail.* **(10 M)**

May 05 [Q. 2(b)] *Draw and explain TCP header.* **(10 M)**

May 06 [Q. 5(b)] *State the different TCP flags.* **(5 M)**

In TCP each segment must have a 20-byte header. Refer figure 6.15.

- **Source Port.** 16-bits.
  Identifies the host process on the senders machine.
- **Destination Port.** 16-bits.
  Identifies the host process on the receivers machine.
  The source and destination port together uniquely identify a connection.



- **Sequence Nun**
  The sequence
  set, the sequer
  byte is initial
  section 6.4.4)
- **Acknowledgm**
  If the ACK bit i
  the receiver is
- **Header Length**
  The number of
  begins. The len
- There is a 6-bit
- **Flags**
  - **URG.** 1-bit.
    The URG b
  - **ACK.** 1-bit.
    ACK is set t
  - **PSH.** 1-bit.
    It is the Pus
  - **RST.** 1-bit.
    The Reset fl
  - **SYN.** 1-bit.
    The SYN bit

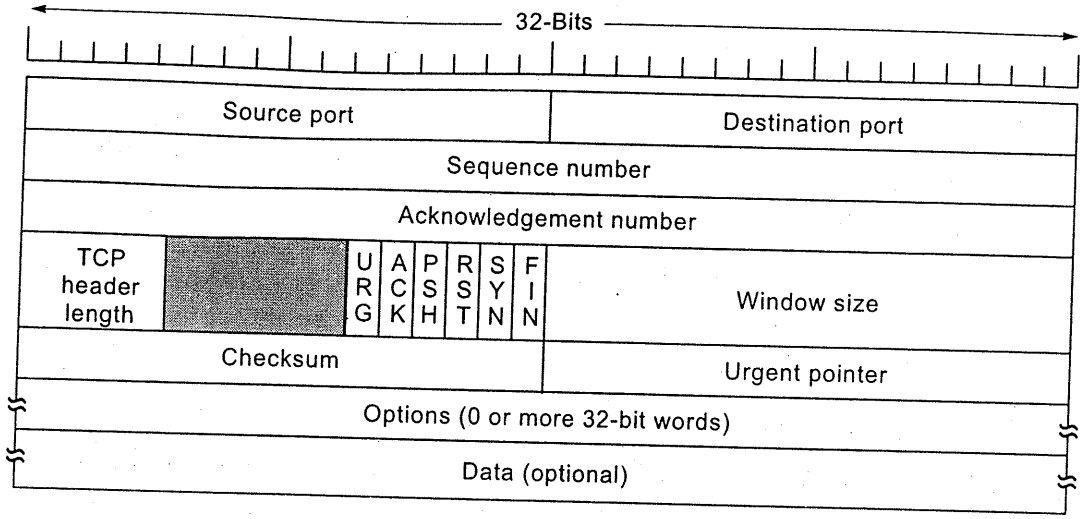


Fig. 6.15 : The TCP header.

- **Sequence Number.** 32-bits.

  The sequence number of the first data byte in this segment. If the SYN bit is set, the sequence number is the initial sequence number and the first data byte is initial sequence number + 1. (The working of SYN bit is covered in section 6.4.4)

- **Acknowledgment Number.** 32-bits.

  If the ACK bit is set, this field contains the value of the next sequence number the receiver is expecting to receive.

- **Header Length.** 4-bits

  The number of 32-bit words in the TCP *header*. This indicates where the data begins. The length of the TCP header is always a multiple of 32-bits.

- There is a 6-bit field which is kept for future use

- **Flags**
  - **URG.** 1-bit.

    The URG bit is set to one if the urgent pointer is in use.
  - **ACK.** 1-bit.

    ACK is set to 1 if the acknowledgement number is valid.
  - **PSH.** 1-bit.

    It is the Push flag which indicates pushed data.
  - **RST.** 1-bit.

    The Reset flag is used to reset a connection.
  - **SYN.** 1-bit.

    The SYN bit is used to establish connections.

- **FIN.** 1-bit.
  The Finish Flag is set to 1 to release a connection. It specifies that the sender has no more data to transmit.

- **Window Size.** 16-bits, unsigned.
  It signifies the number of data bytes that the receiver is willing to accept.

- **Checksum.** 16-bits.
  Checksum algorithm simply adds the 16-bit words of the header, data and the pseudo header in one's complement form and then takes the one's complement of the sum. If a segment contains an odd number of header and text octets to be checksummed, the last octet is padded on the right with zeros to form a 16-bit word for checksum purposes. The pad is not transmitted as part of the segment. While computing the checksum, the checksum field itself is replaced with zeros.
  The checksum also covers a 96-bit *pseudo header* conceptually prefixed to the TCP header.
  The *pseudo header* contains the following fields :

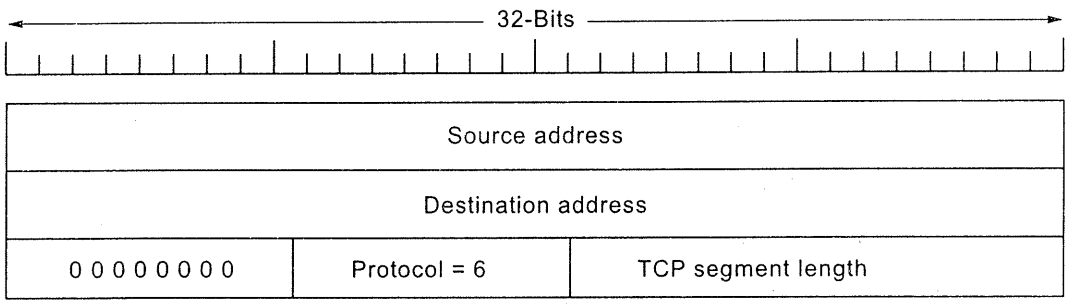


Fig. 6.16 : The pseudoheader.

  - 32-bit address of source and destination machines.
  - The protocol number(the value of protocol number is 6 for TCP)
  - Byte count for the segment (including the header)
  - *UDP uses the same pseudo header for its checksum.*
  - The pseudo header helps detect misdelivered packets.

- **Urgent Pointer.** 16-bits, unsigned.
  This field communicates the current value of the urgent pointer as a positive offset from the sequence number in this segment. The urgent pointer points to the sequence number of the octet following the urgent data. This field is only interpreted in segments with the URG control bit set.

- **The Options Field**
  It provides a way to add extra facilities not covered by the regular header.

  - The mos
    maximur
  - As ment
    importan
    N protoco

6.4.4 TCP Connectio

Dec. 03 [Q. 3(a)] *Des*
TCP end point but t
eslabish communicatio

Dec. 03 [Q. 3(b)] *Sho*
*(**Note :** Also write abou*

TCP establishes conne
The steps in brief are :

- When the se
  connection by
- When a client
- In the CONNE
  SYN bit ON a
  establish a co
  acknowledging
- The server who
  accept or reje
  acknowledgeme
  (SYN SEQ = y,
  arrow shown in

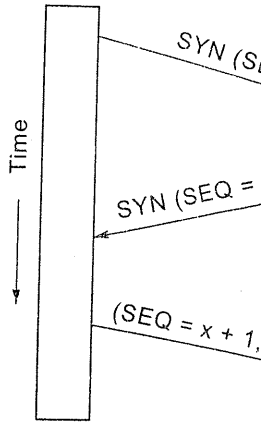


Fig. 6.17(a) : TCP conn
in the nor

- The most important option is the one that allows hosts to decide the maximum packet that it is willing to accept.
- As mentioned before, TCP uses Sliding Window Protocol. Another important option is that *selective repeat* can be used instead of the *go back N protocol.*

6.4.4 TCP Connection Establishment

Dec. 03 [Q. 3(a)] *Desribe a scenario in which a TCP connection request is received by a TCP end point but there is infact no corresponding TCP end point who wishes to eslablish communication with receiver.* ***(Note :** Also write about section 6.2.2)* **(10 M)**

Dec. 03 [Q. 3(b)] *Show how TCP connection setup protects against the situation in (a).* ***(Note :** Also write about section 6.2.2)* **(5 M)**

TCP establishes connections using the *3 way handshake* discussed earlier.

The steps in brief are :

- When the server (HOST 2) is free it passively waits for an incoming connection by executing LISTEN primitive.
- When a client wants to connect to a server it uses the CONNECT primitive.
- In the CONNECT primitive the client (HOST 1) sends a TCP segment with SYN bit ON and ACK bit OFF. (SYN=ON means that the client wants to establish a connection and ACK = OFF means that the client is not acknowledging anything.)
- The server who is on LISTEN receives this CONNECT segment and can either accept or reject the connection. If it accepts a segment it sends an acknowledgement as shown in the figure 6.17(a).

  (SYN SEQ = y, ACK = x + 1). Now the client can send data using the 3rd arrow shown in figure 6.17(a).

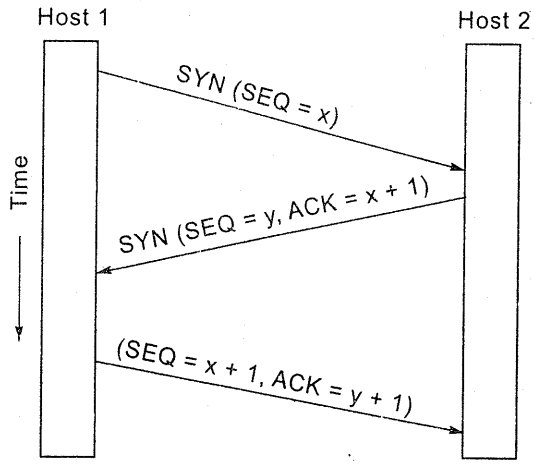


Fig. 6.17(a) : TCP connection establishment in the normal case.

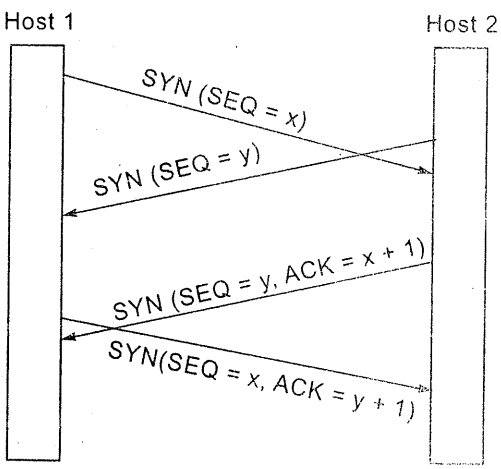


Fig. 6.17(b) : Call collision.

- Suppose that two hosts want to simultaneously establish a connection between the same two sockets, the sequence of events is shown in figure 6.17(b). The steps result in only one connection being established, not two, because the connection is identified by its end points.

6.4.5 TCP Connection Release

- Consider a connection between Host 1 and Host 2.
- If Host 1 has no more data to transmit it can close its side of the connection by sending a TCP segment with the FIN bit set. When this FIN is acknowledged by Host 2 it means that Host 1 has no more data to transmit, but Host 2 can still send data to Host 1, i.e. Host 1 cannot send more data but it can still receive data. Only when Host 2 does a FIN can we say that the connection is terminated.
- Thus 4 TCP segments are needed to terminate a connection : 2 FINs and 2 Acknowledgements.

6.4.6 TCP Connection Management Modelling

Dec. 03 [Q. 3(b)] *Draw the space time diagram for protocol message exchange and explain how the protocol works.* **(5 M)**

The steps required to establish and release a connection can be represented with a Finite State Machine which has 11 states as shown in *figure 6.18* and *6.19*.

State	Description
CLOSED	No connection is active or pending
LISTEN	The server is waiting for an incoming call
SYN RCVD	A connection request has arrived; wait for ACK
SYN SENT	The application has started to open a connection
ESTABLISHED	The normal data transfer state
FIN WAIT 1	The application has said it is finished
FIN WAIT 2	The other side has agreed to release
TIMED WAIT	Wiat for all packets to die off
CLOSING	Both sides have tried to close simultaneously
CLOSE WAIT	The other side has initiated a release
LAST ACK	Wait for all packets to die off

Fig. 6.18 : The states used in the TCP connection management finite state machine.

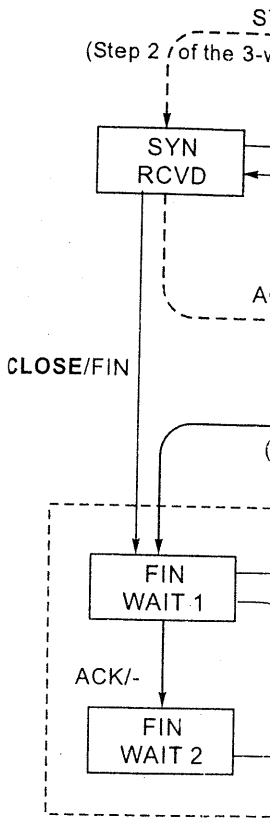


Fig. 6.19 : TCP
is the normal patl
The light
event caus

6.4.7 TCP Transmis

- As mentione
can accept.
- Consider a re
bytes, the rec

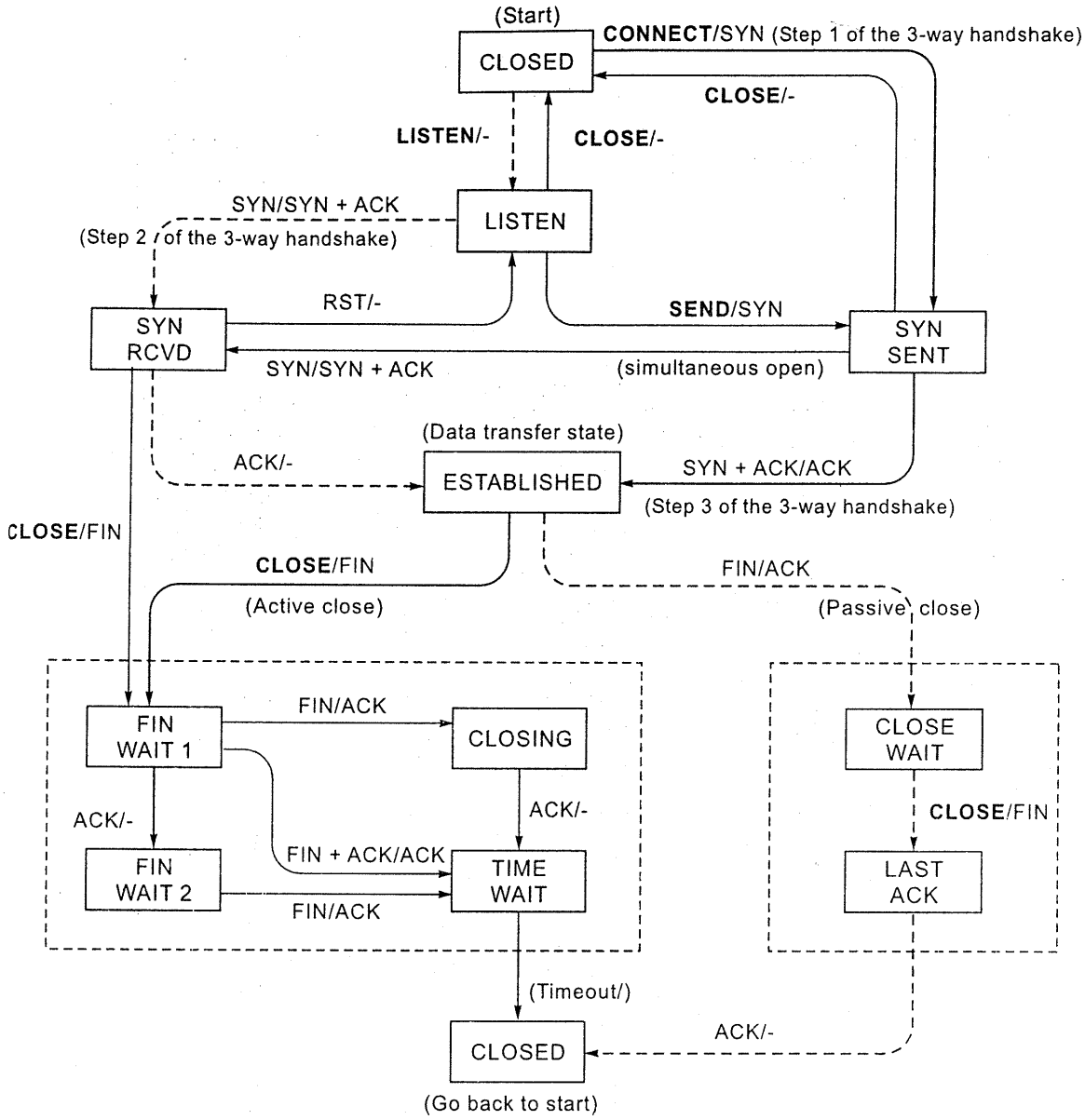


Fig. 6.19 : TCP connection management finite state machine. The heavy solid line is the normal path for a client. The heavy dashed line is the normal path for a server. The light lines are unusual events. Each transition is labeled by the event causing it and the action resulting from it, separated by a slash.

6.4.7 TCP Transmission Policy

- As mentioned earlier the receiver can specify the *size of the window* which it can accept.
- Consider a receiver which has a 4096 byte buffer. If the sender transmits 2048-bytes, the receiver will use 2048-bytes from its buffer and will have only 2048-

bytes left in its buffer. Therefore the receiver will now specify that it can accept a maximum window size of 2048. Now the sender sends 2048-bytes; thus the receivers buffer is now completely used and it will specify that it has 0 bytes left. Thus the sender must stop sending till the receiver can free some of its buffer space and specify a larger window size.

- **Nagle's Algorithm**
  - **(a)** Nagle's algorithm is a means of improving the efficiency of *TCP/IP* networks by reducing the number of packets that need to be sent over the network.
  - **(b)** Nagle's document describes what he called the 'small packet problem', where an application repeatedly emits data in small chunks, frequently only 1-byte in size.
  - **(c)** Since *TCP* packets have a 40-byte header (20-bytes of TCP header and 20-bytes of IP header), this results in a 41 byte packet for 1 byte of useful information, a huge overhead.
  - **(d)** The Nagle algorithm works by coalescing a number of small outgoing messages, and sending them all at once. Specifically, as long as there is a sent packet for which the sender has received no acknowledgement, the sender should keep buffering its output until it has a full packet's worth of output, so that output can be sent all at once.
  - **(e)** *Basic idea is that till the sender does not receive an ACK for the previous packet that it sent; it keeps collecting information to send in its next packet.*

- **Silly Window Syndrome**

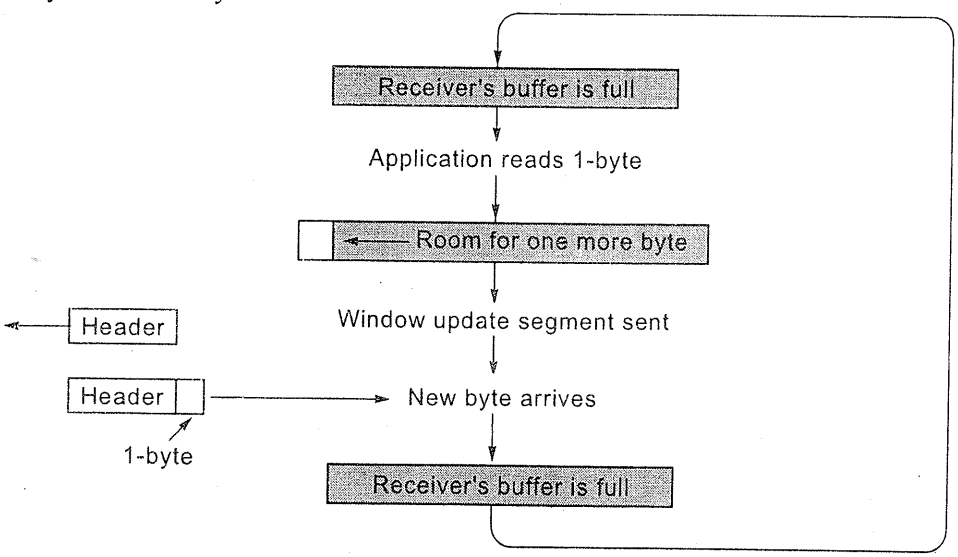


Fig. 6.20 : Silly window syndrome.

(a) Refer *figur*
Now supp
window u
1 byte. W
full and th
will contin
Thus too n
byte.

(b) A simple s
the receive
buffer is e
whichever

.4.8 Congestion Con

lay 06 [Q. 6(b)], May

)ec. 06 [Q. 7(5)] *Write*

When the load of a netw
Congestion occurs beca

a) The Receivers Buffe
buffer size. If the s
example in the *fig*
receiver. As long as
contain, no water w

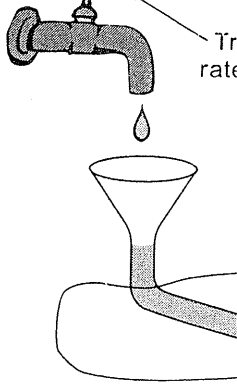


Fig. 6.21 : A fas
low capa

pecify that it can
sends 2048-bytes;
specify that it has
iver can free some

iciency of *TCP/IP*
:d to be sent over

l packet problem',
hunks, frequently

CP header and 20-
r 1 byte of useful

of small outgoing
long as there is a
owledgement, the
ull packet's worth

the previous packet
t packet.

(a) Refer *figure 6.20*. Consider the situation when the receiver's buffer is full. Now suppose one byte in the buffer is released, the receiver will send a window update to the sender saying that it can accept a window of size 1 byte. When the sender sends one byte, the buffer will again become full and the receiver will specify that its window size is 0. This process will continue as and when one byte of the receiver's buffer becomes free. Thus too many segments are sent to and fro for the transfer of a single byte.

(b) A simple solution to this problem was suggested by Clark. He said that the receiver must not send a window update to the sender unless half its buffer is empty or it can accept the MSS(Maximum Segment Size), whichever is smaller.

6.4.8 Congestion Control

May 06 [Q. 6(b)], May 07 [Q. 2(a)] *How TCP controls congestion ?* **(10 M)**

Dec. 06 [Q. 7(5)] *Write short note on QoS in transport layer. (Also section 6.1)* **(5 M)**

When the load of a network is more than it can handle, congestion takes place. Congestion occurs because :

a) **The Receivers Buffer is Small** : The receiver can specify a window based on its buffer size. If the sender sticks to this window size, problems will not occur. For example in the *figure 6.21* we see a thick pipe leading to a small-capacity receiver. As long as the sender does not send more water than the bucket can contain, no water will be lost.

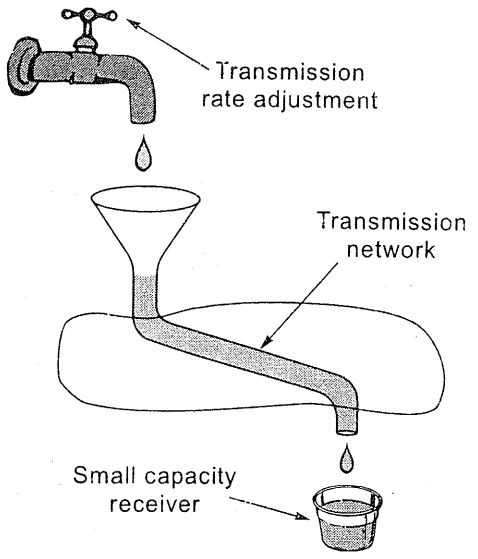


Fig. 6.21 : A fast network feeding a low capacity receiver.

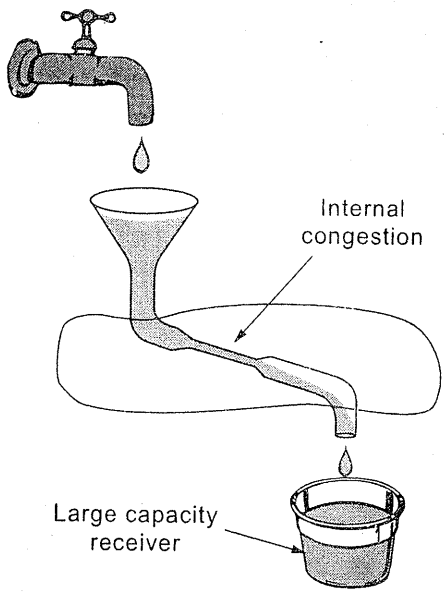


Fig. 6.22 : A slow network feeding a high capacity receiver.

(b) **The Internal Congestion Due to the Network :** In *figure 6.22*, the limiting factor is not the bucket capacity, but the internal carrying capacity of the network. If too much water comes in too fast, some water will be lost by overflowing the funnel from the top.

Thus two potential problems exist : (I) Receiver capacity and (II) Network Capacity.

To handle these 2 problems each sender maintains two windows :

(I) *The window the receiver has granted :* This specifies the number of bytes that the receivers buffer can accept.

(II) *The congestion window :* This signifies the number of bytes that the network can handle without congestion.

The number of bytes that the sender may send is the minimum of the two windows.

E.g. If the receiver says "Send 8 KB" but the sender knows that bursts of more than 4 KB clog the network, it sends 4 KB.

Working

(1) When a connection is established, the sender initializes the Congestion Window to the Maximum Segment Size in use on the connection.

(2) It then sends one maximum segment. If this segment is acknowledged before the timer goes off, it increases the size of the congestion window to Two Maximum Size Segments (double its size) and sends two segments. Generally, if the congestion window is n segments and if all n are acknowledged on time, the congestion window is doubled to 2n segments.

(3) The congestion window keeps growing exponentially (doubling the size of the congestion window) until either a timeout occurs or the receiver's window is reached.

(4) The points 1, 2, 3 is called the Slow Start Algorithm. The points 6, 7, 8 specify how this algorithm is used in Internet Congestion Control.

(5) The Internet Congestion Control algorithm uses the *Receiver Window* the *Congestion Window* along with a third parameter called the *"Threshold"* which is initially set to 64 KB.

(6) When a timeout occurs, the threshold is set to half of the current congestion window, and the congestion window is reset to one maximum segment.

(7) The Congestion Window again starts growing exponentially (using the Slow Start Algorithm) till the Threshold level is reached.

(8) After the Threshold level is reached the Congestion Window grows linearly (by one maximum segment for each burst) instead of exponentially (double the size).

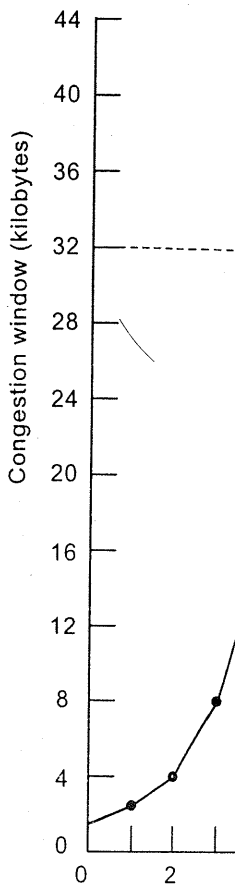


Fig.

E.g.

(1) The maximum

(2) Initially, the c threshold is set

(3) The congestion (32 KB). From t

(4) Consider that the current wi all over again. linear after the

(5) If no more tim the size of the constant as lon not change size

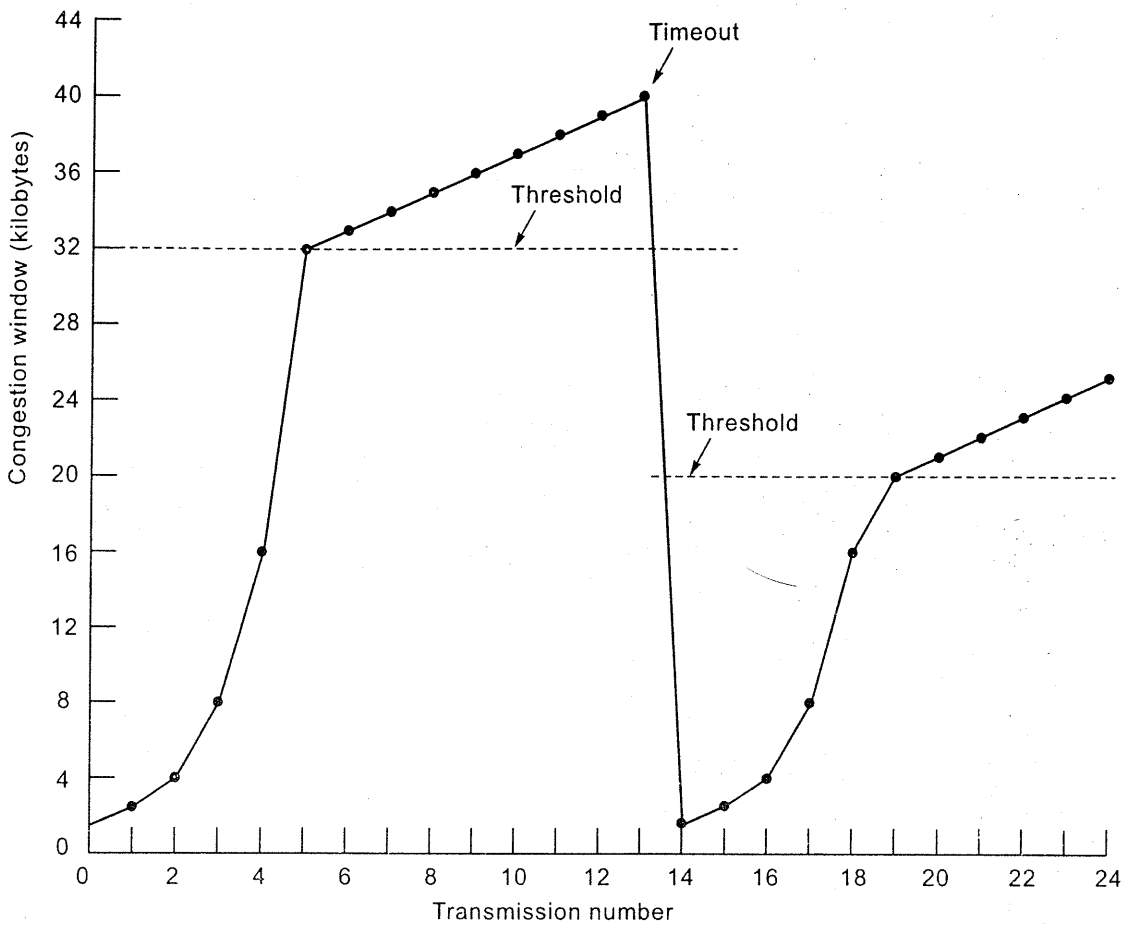


Fig. 6.23 : An example of the Internet congestion algorithm.

E.g.

(1) The maximum segment size here is 1 KB.

(2) Initially, the congestion window was 64 KB, but a timeout occurred, so the threshold is set to 32 KB and the congestion window to 1 KB .

(3) The congestion window then grows exponentially until it hits the threshold (32 KB). From the threshold it grows linearly.

(4) Consider that a timeout occurs at Transmission 13. The threshold is set to half the current window (by now 40 KB, so half is 20 KB), and slow start is initiated all over again. The increase is exponential till the threshold (20 KB) and becomes linear after the threshold is reached.

(5) If no more timeouts occur, the congestion window will continue to grow up to the size of the receiver's window. At that point, it will stop growing and remain constant as long as there are no more timeouts and the receiver's window does not change size.

6.4.9 TCP Timer Management

What is the use of a timer ?

When a segment is sent, retransmission timer starts at the sender.

- If the segment is ACKed, the timer stops.
- If it times out, the segment is retransmitted and timer starts again.

How long should the timeout interval be ?

- **Round Trip Time(RTT) :** It is the time between a segment is sent and ACK comes back.
- If timeout is too short, unnecessary retransmissions must be done.
- If timeout is too long, long retransmission delay occurs.
- Determining the round-trip time :
  - TCP keeps RTT variable.
  - When segment is sent, TCP measures how long it takes to get ACK back (M).
  - $RTT = \alpha * RTT + (1 - \alpha)M$.
  - α : smoothing factor; determines weight given to previous estimate.
  - Typically, $\alpha = 7/8$.
- Determining timeout value :
  - Measure RTT variation, or $|RTT - M|$.
  - Keeps smoothed value of cumulative variation
  - $D = \alpha * D + (1 - \alpha)|RTT - M|$.
  - Timeout = $RTT + 4 * D$.

Karn's Algorithm

- When a segment timesout it must be sent again.
- When an ACK comes in, it becomes unclear to decide if it is an acknowledgement for the first transmission or for the later one. This causes difficulty in estimating the RTT.
- Karn proposed not to update RTT on any retransmitted segment.

Persistence Timer

- A receiver sends the sender a Window Size = 0. Later the receiver sends a message to the sender updating this window size, but this message gets lost. Now both the sender and the receiver are waiting for each other to do something. This is a Deadlock.
- To avoid thi
  sender send
  size.
- If window s
  the data can

Keepalive Timer

- It causes one

TIME_WAIT Timer

- It Makes sur

6.5 Wireless TCP a

- In theory TC
  network lay
  the transpor
  TCP and UD
- Timeouts ca
  (1) Congest
  (2) Lost pac
- **Problem wit**
  Timeout lea
  increased tra
- **Difference b**
  (1) In wire
  is to m
  congest
  (2) In a wi
  the wir
  lost pac
  makes t
  are rece
  10 will
  from 80
  Thus if
  and if it
- A path from
  media. A s

- To avoid this we have a Persistence Timer. When persistence timer goes off, sender sends a probe to the receiver; receiver replies with its current window size.
- If window size = 0, persistence timer is set again. If window size is non zero the data can be sent to the receiver.

Keepalive Timer

- It causes one side to check if the other side is still there.

TIME_WAIT Timer

- It Makes sure all segments die after connection is closed.

6.5 Wireless TCP and UDP

- In theory TCP and UDP should be independent of the technology used in the network layer. In practice there is a relation between the protocols used in the transport layer and the technology used in the network layer. Generally TCP and UDP are configured differently for wired and wireless networks.
- Timeouts can be caused because of 2 factors :
  (1) Congestion or
  (2) Lost packets.
- **Problem with Timeouts**

  Timeout leads to retransmission of packets. Retransmission of packets lead to increased traffic in the network which may lead to added congestion.
- **Difference between Wired and Wireless Networks**
  (1) In wired networks, timeouts are caused by congestion. Thus the solution is to make the sender send packets less vigorously so as to reduce congestion.
  (2) In a wireless network timeouts are generally caused by lost packets as the wireless networks are not very reliable. A solution to this is to send lost packets again as quickly as possible. Slowing down the sender just makes the problem worse. E.g. If for every 100 packets 20 are lost and 80 are received. Now if we reduce the sending rate to 50, correspondingly 10 will be lost and 40 will be received , thus reducing the throughput from 80 to 40.

  Thus if timeout occurs in a wired network the sender should slow down and if it occurs on a wireless network the sender should try harder.
- A path from a sender to the receiver may be made up of wired and wireless media. A solution is to divide the network into 2 parts as shown in the

figure 6.24. The first connection from the Sender to the Base Station uses Wired Medium and the second connection from the Base Station to the Receiver uses Wireless Medium.

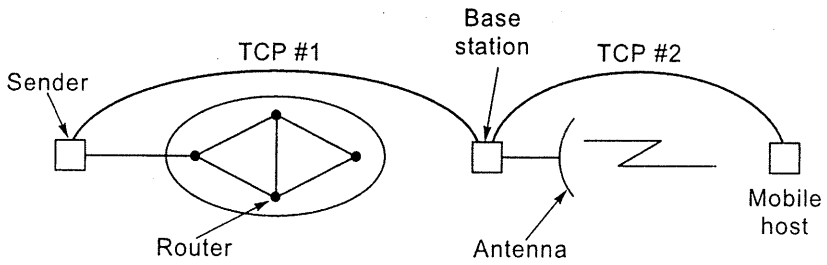


Fig. 6.24 : Splitting a TCP connection into two connections.

- In case of UDP the problem is that programs expect UDP to be very reliable.

6.6 Transactional TCP (T/TCP)

- It is a way of combining the efficiency of using RPC (using UDP) with the reliability of TCP.
- The working of the T/TCP is as follows :

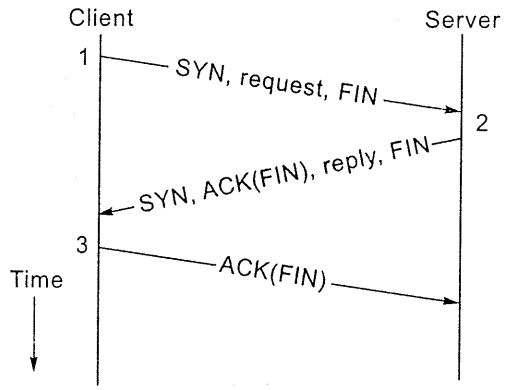


Fig. 6.25 : RPC using T/TCP.

(1) The client first sends a packet containing the SYN bit, the request and the FIN. In effect this means that the client tells the server" SYN: I want to establish a connection, Request: Here is the Data , FIN: I am done."

(2) The server gets the above message and replies with a packet containing a SYN, an Acknowledgement, a reply and a FIN. The server effectively wants to say" I acknowledge your FIN, here is the answer, I am done."

(3) The client acknowledges the servers FIN and the protocol terminates in three stages.

6.7 Difference

Dec. 03 [Q. 5(a)]
when the client in

Dec. 04 [Q. 6(a)]
list the services o
difference betwee

Dec. 05 [Q. 2(b)]

Sr. No.	
(1)	Uses Conne
(2)	TCP is high
(3)	TCP packet
(4)	TCP is con oriented.
(5)	TCP connec
(6)	It does NC and Multica
(7)	It provides control.
(8)	It supports
(9)	Application (a) TELNET capability (b) FTP(Use between (c) SMTP(de

(If asked for 10 mar and 6.4.3)

6.7 Difference Between TCP and UDP

Dec. 03 [Q. 5(a)] *Explain the difference of UDP and TCP socket of Server side especially when the client initiates the connection or request to the server.* **(10 M)**

Dec. 04 [Q. 6(a)] *What do TCP and UDP use as transport layer addresses (refer 6.2.1) ? list the services offered by both to their upper layers (refer 6.1.1). Bring out the major difference between the two (refer 6.7).* **(10 M)**

Dec. 05 [Q. 2(b)] *Differentiate between TCP and UDP.* **(4 M)**

Sr. No.	TCP	UDP
(1)	Uses Connection Oriented Service.	Uses Connectionless Service.
(2)	TCP is highly reliable	UDP is comparatively unreliable.
(3)	TCP packet is called Segment.	UDP packet is called User Datagram.
(4)	TCP is complex as it is connection oriented.	UDP is simpler to implement than TCP as UDP does not require connection establishment as it is a connectionless service.
(5)	TCP connection is a Byte Stream.	UDP connection is a Message Stream.
(6)	It does NOT support Broadcasting and Multicasting.	It supports Broadcasting and Multicasting.
(7)	It provides error control and flow control.	It does NOT provide error control and flow control.
(8)	It supports Full Duplex Transmission.	It does not support Full Duplex Transmisson.
(9)	Application of TCP : (a) TELNET(Provides remote login capability), (b) FTP(Used for transfer of files between hosts), (c) SMTP(delivers e-mail).	Applicaions of UDP : (a) SNMP(Used to collect management information from network devices) (b) DNS(Used to map between IP addresses and domain names).

(If asked for 10 marks then also show the header formats of TCP and UDP from section 6.3 and 6.4.3)

7 APPLICATION LAYER AND SESSION LAYER

Marks	
Dec. 03	-
May 04	10 M
Dec. 04	-
May 05	-
Dec. 05	-
May 06	10 M
Dec. 06	2.5 M
May 07	-

The Application and Session layer are present in the higher part of the ISO-OSI hierarchy. Topics on SNMP, E-mail and Session Management are discussed in detail here.

7.1 Simple Network Management Protocol

(1) The Simple Network Management Protocol (SNMP) is an application layer protocol that facilitates the exchange of management information between network devices.

(2) It is part of the Transmission Control Protocol/Internet Protocol (TCP/IP) protocol suite.

(3) Two versions of SNMP exist : SNMP version 1 (SNMPv1) and SNMP version 2 (SNMPv2). Both versions have a number of features in common, but SNMPv2 offers additional protocol operations.

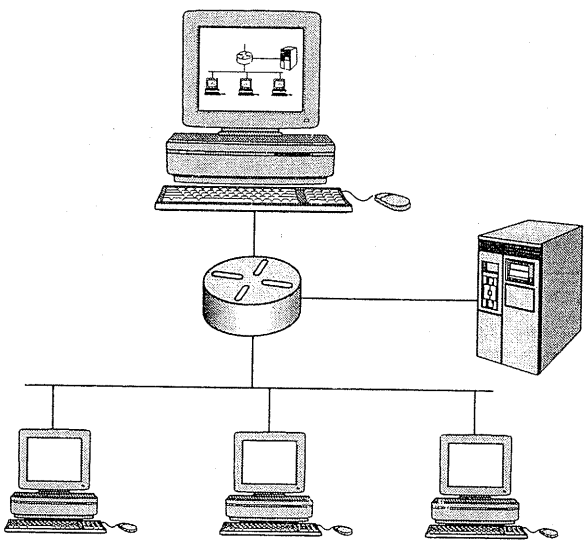


Fig. 7.1: SNMP facilitates the exchange of network information between devices.

SNMP Basic C

An SNMP-manag

Fig. 7.2 : An S

(1) **Managed De** devices colle available to N and bridges,

(2) **Agent** : It is device. Its f SNMP.

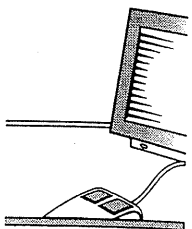
(3) **Network Ma** and controls three compon

SNMP Basic Co

Managed devices read, write, trap,

Read : Used by N

Write : Used by N



Marks
03 : -
04 : 10 M
04 : -
05 : -
05 : -
06 : 10 M
06 : 2.5 M
07 : -

ication layer
ion between

col (TCP/IP)

MP version 2
but SNMPv2

evices.

SNMP Basic Components

An SNMP-managed network consists of three key components :

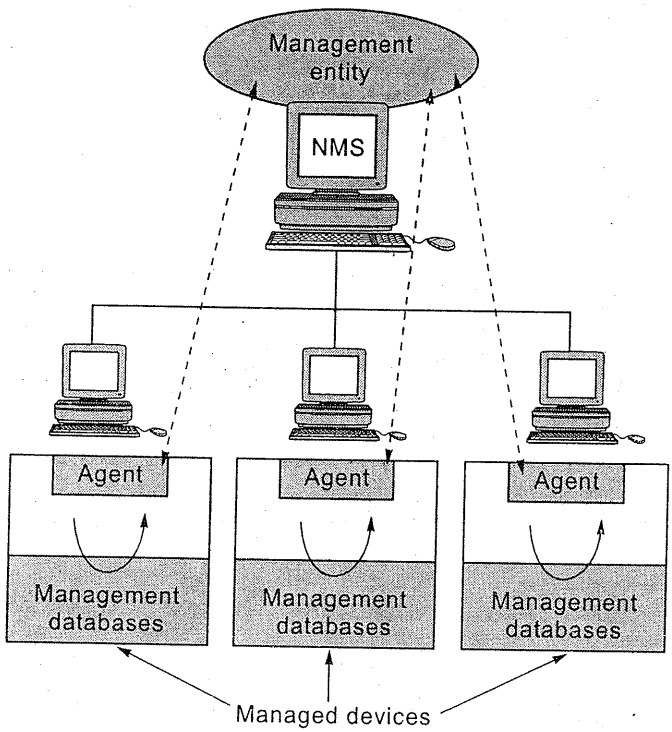


Fig. 7.2 : An SNMP-managed network consists of managed devices, agents, and NMSs.

(1) **Managed Device :** It is a network node that contains an SNMP Agent. Managed devices collect and store management information and make this information available to NMSs using SNMP. They can be routers and access servers, switches and bridges, hubs etc.

(2) **Agent :** It is a network-management software module that resides in a managed device. Its function is to translate information into a form compatible with SNMP.

(3) **Network Management Systems (NMS) :** It executes applications that monitor and controls the managed devices. *Figure* 7.2 illustrates the relationships of these three components.

SNMP Basic Commands

Managed devices are monitored and controlled using four basic SNMP commands : read, write, trap, and traversal operations.

Read : Used by NMS to read the information maintained by managed devices.

Write : Used by NMS to write the information to managed devices.

Trap : Used by Managed Devices to report events to the NMS.

Traversal Operations : Used by the NMS to determine which variables a managed device supports.

7.2 E-Mail

May 06 [Q. 7(b)] *How are mails sent and received . Show with diagrams. Also draw mail headers.* **(10 M)**

Dec. 06 [Q. 5(b(i))] *Explain the Internet services (i) E-mail.* **(2.5 M)**

- **E-mail** allows individuals to communicate using computers and the internet.

  The first e-mail systems consisted of file transfer protocols, with the receipients address in the first line of the message

- The problems faced by this technique were :
  - Sending messages to a group of people was inconvenient.
  - The sender never knew if the message arrived or not.
  - The user interface was not very user-friendly.
  - It was not possible to send messages which contained a mixture of text, drawings, fax and voice.

- **Architecture**

  Email systems consists of two subsystems :

  (1) *The User Agent :* It allows users to compose and read the messages. It basically provides a user interface to the e-mail system. It is a program that accepts commands from a user and carries out operations like composing, receiving and replying to messages.

  (2) *The Message Transfer Agents :* They are responsible for relaying mail from a sender to the receiver.

- A key idea in email is the distinction between the envelope and it contents. The envelope encapsulates the message.

- **Envelope :** The envelope contains information such as the destination address, priority and security level. The message transport agents use the envelope for routing.

- **Message :** The message inside the envelope consists of two parts :

  (1) *The Header :* It contains control information for the user agents.

  (2) *The Body :* The actual information that the sender wanted to send to the receiver.

Mr. Dinesh De
18 Dillon Lan
Sea Woods,
Navi Mumbai
Maharashtra.

P
1
F
M
A

Subject: Invoice 100

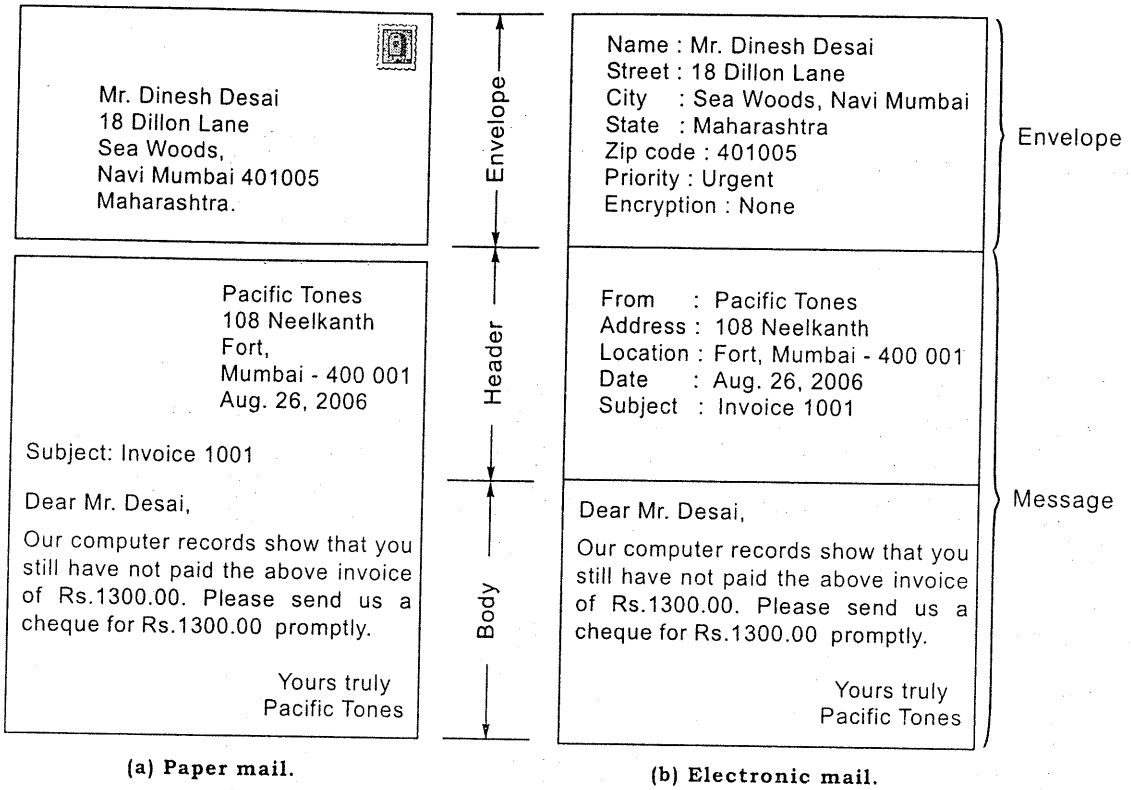
Dear Mr. Desai,

Our computer record
still have not paid th
of Rs.1300.00. Ple
cheque for Rs.1300.

(a) Paper

- **Functions of E**
  (1) *Compositio*
  (2) *Transfer :* F
  (3) *Reporting* delivered/
  (4) *Displaying*
  (5) *Disposition* receiving i

- **Additional Fur**
  (1) *Mailing Lis*
  (2) *CC :* Carbo
  (3) *BCC :* Blin
  (4) Priority of
  (5) Encrypted
  (6) An e-mail



(a) Paper mail. (b) Electronic mail.

Fig. 7.3 : Envelopes and messages.

- **Functions of E-mail**
  (1) *Composition* : Process of creating mails and answers.
  (2) *Transfer* : Process of moving mails from sender to receiver.
  (3) *Reporting* : Process of telling the sender if the mail he sent was delivered/rejected/lost.
  (4) *Displaying* : Process of displaying incoming messages for the users to read.
  (5) *Disposition* : Concerns what the recipient does with the message after receiving it. A user may delete/read/save a message.

- **Additional Functions**
  (1) *Mailing List* : It is a list of e-mail addresses.
  (2) *CC* : Carbon Copies can be sent.
  (3) *BCC* : Blind Carbon Copies can be sent.
  (4) Priority of a mail can be set.
  (5) Encrypted e-mail can be sent.
  (6) An e-mail can be forwarded from one account to another automatically.

- **The User Agent**

It has 2 main functions :

(a) Sending email.

(b) Receiving/Reading email.

(a) Sending E-mail

(1) To send an e-mail a user must provide the message to be sent, the destination address and possibly some other parameters.

(2) The message can be created with a stand-alone text editor or a special text editor which comes in built with the email system.

(3) The destination address must be in a format that the user agents can understand. The general form of addresses is username@dns-address.

(4) Most e-mail systems have nicknames that allow users to select a persons nickname and get the email address.

(5) Nearly all email systems provide mailing lists, so that a user can send a mail to a list of people with a single command.

(b) Reading E-mail

- When a user agent is started up it looks at the user's mailbox for incoming email.
- Then it may announce the number of messages in the mailbox or display a one line summary of each message and then wait for a command.

The one line summary of each mail looks as follows :

#	Flags	Bytes	Sender	Subject
1	K	1030	asw	Changes to MINIX
2	KA	6348	trudy	Not all Turdys are nasty
3	K F	4519	Amy n. Wong	Request for information
4		1236	bal	Bioinformatics
5		104110	kaashoek	Material on peer-to-peer
6		1223	Frnak	Re: Will you review a grant proposal
7		3110	guido	Our paper has been accepted
8		1204	dmr	Re: My student's visit

Fig. 7.4 : An example display of the contents of a mailbox.

- The various fields have some significance :

(1) *Message Number (#)* : The number of the message.

(2) *Flags* : 'K' means that the message has been read and is *kept* in the mail

box. 'A' mea
been *forwar*

(3) *Bytes* : Tells

(4) *Sender* : Say

(5) *Subject* : Giv

- **Message Transf**

It consists of th

(1) *SMTP* : Sim
sender's ma

(2) *POP3 and IM*
from the rec

(a) POP3(Post Offi

Process : The
mail server to
server. Now the

(1) It works w
computers
computers
computers.

(2) Email is dow

(3) Email has to

(4) It uses low a

(5) It cannot ma

(6) It is simple

(7) Users canno

(8) Users canno

(9) The content

(b) IMAP4 (Interne

Process : It per
The messages
client can mani

Features :

(1) It is benefici

(2) E-mail is sto

(3) E-mail has t

box. 'A' means that the mail has been *answered*. 'F' means that the mail has been *forwarded*.

(3) *Bytes :* Tells the receiver how long the message is.

(4) *Sender :* Says who the sender is.

(5) *Subject :* Gives a brief summary of the message.

- **Message Transfer Agents**

It consists of the following protocols :

(1) *SMTP :* Simple mail transfer protocol. It is used to send a mail from the sender's machine to the receiver's server. (Explained in section 1.13, chapter 1)

(2) *POP3 and IMAP4 :* They are used by the receiver to retrieve (download) mail from the receiver's server.

(a) POP3(Post Office Protocol Version3)

Process : The user accesses the mail server and downloads the mails from the mail server to its own computer. The messages are now removed from the mail server. Now the user can go *offline and read his messages.*

(1) It works well only when the user is using a single computer. If multiple computers are used the mail messages will be downloaded on multiple computers which will result in a sprinkling of messages over all the computers.

(2) Email is downloaded from the *Server* and stored on the *User Computer.*

(3) Email has to be read *Offline.*

(4) It uses low amount of server resources as the mail is read offline.

(5) It cannot maintain multiple mailboxes for a user.

(6) It is simple compared to IMAP.

(7) Users cannot organize mail.

(8) Users cannot maintain the mail in different folders.

(9) The contents of the mail cannot be partially checked before downloading.

(b) IMAP4 (Internet Mail Access Protocol Version 4)

Process : It permits the user to access the messages stored on the mail server. The messages are NOT transferred from the server to the user computer. The client can manipulate the message directly on the mail server.

Features :

(1) It is beneficial to those who access their mail from different computers.

(2) E-mail is stored on the Server.

(3) E-mail has to be read *Online.*

(4) It uses high amount of server resources as the mail is accessed from the server itself.

(5) It can maintain multiple mailboxes for a user.

(6) It is complex compared to POP.

(7) Folders can be created to organize the mail.

(8) Users can check the mail headers before downloading.

(9) Mails can be partially downloaded due to bandwidth constraints.

- **Message Formats**

(a) RFC 822

The User Agent builds the message and passes it to the Message Transfer Agent which then uses some of the header field to construct the envelope.

RFC 822 header fields related to the message transport are :

Header	Meaning
To:	E-mail address(es) of primary recipient(s)
Cc:	E-mail address(es) of secondary recipient(s)
Bcc:	E-mail address(es) of blind carbon copies
From:	Person or people who created the message
Sender:	E-mail address of the actual sender
Received:	Line added by each transfer agent along the route
Return-Path:	Can be used to identify a path back to the sender

Fig. 7.5(a) : RFC 822 header fields related to message transport.

RFC 822 header fields rused by useragents and humans are :

Header	Meaning
Date:	The date and time the message was sent
Reply-To:	E-mail address to which replies should be sent
Message-Id:	Unique number for referencing this message later
In-Reply-to:	Message-Id of the message to which this is a reply
References:	Other relevant Message-Ids
Keywords:	User-chosen keywords
Subject:	Short summary of the message for the one-line display

Fig. 7.5(b) : Some fields used in the RFC 822 message header.

(b) MIME : Multi

The basic ide
additional fea
encoding rules
MIME message

The MIME hea

Header
MIME-Vers
Content-De
Content-Id:
Content-Tra
Content-Ty

7.3 Session Man

May 04 [Q. 4(a)] *Ex*
interaction between

[A] Dialogue Mana

(1) Dialogue ma

(2) Only the us

(3) The token
primitive.

[B] Session Mainte

(1) The message

(2) The user has

(3) Consider a
computers.
session is ma

(b) MIME : Multipurpose Internet Mail Extensions

The basic idea of MIME is to continue to use RFC 822 format. MIME has additional features which add structure to the message body and define the encoding rules for the non ASCII messages. As it is an extension of the RFC 822 , MIME messages can be sent with the existing mail programs and protocols.

The MIME headers are as follows :

Header	Meaning
MIME-Version:	Identifies the MIME version
Content-Description:	Human-readable string telling what is in the message
Content-Id:	Unique identifier
Content-Transfer-Encoding:	How the body is wrapped for transmission
Content-Type:	Type and format of the content

Fig. 7.6 : RFC 822 header fields added by MIME.

7.3 Session Management

May 04 [Q. 4(a)] *Explain how a session layer establishes, maintains and synchronises the interaction between two communicating hosts.* **(10 M)**

[A] Dialogue Management

(1) Dialogue management is implemented using data tokens.

(2) Only the user holding the token can transmit data.

(3) The token can be given from one user to another using the S- Token-Give primitive.

[B] Session Maintenance and Activity

(1) The message stream is split into logical units called activities.

(2) The user has to decide how to divide the message stream into activities.

(3) Consider a session where many files have to transferred between two computers. In this case an activity will be the transfer of a single file. The session is made up of many activities as shown in *figure 7.7*.

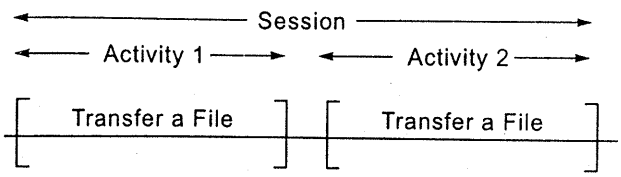


Fig. 7.7 : A session with activities.

[C] Session Synchronization

(1) Synchronization is needed within an activity.

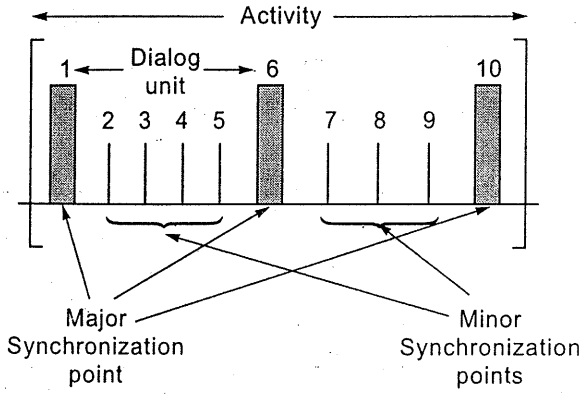


Fig. 7.8 : An activity.

(2) There are two types of synchronization points :

(a) *Major Synchronization Point :*

(i) They separate logically separate pieces of work within an activity

(ii) While re-synchronizing we can go back till the most recent synchronization point.

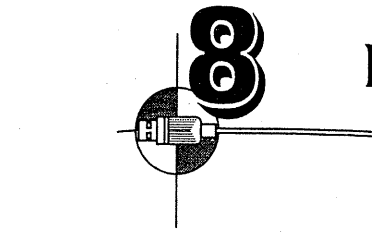
(iii) In the *figure 7.8* suppose we are at 8. Now, we can resynchronize only from 6 onwards.

(iv) Units delimited by major synchronization points are called *dialog units.*

(b) *Minor Synchronization :*

(i) Minor Synchronization points, unlike Major Synchronization points, allow us to traverse backward if we want to resynchronize.

(ii) In the *figure 7.8* suppose we want to resynchronize when we are at 9. Now when we traverse backward, we come to 8 which is a Minor Synchronization point, therefore we can pass it. Next comes 7 which is again a Minor Synchronization point so it can be passed. Next comes 6 which is a Major Synchronization point and therefore we have to stop our backward movement here.



Over time, Netw of networks. The va here.

8.1 Need for Net

(1) Protect data du

(2) Guarantee that

8.2 Cryptography

May 05 [Q. 7] *Crypt*

Dec. 05 [Q. 7(d)] *Cr*

SENDER | A B C

Plaintext

(a) **Plaintext** : It is

(b) **Ciphertext** : En the intended re

(c) **Encryption** : En

(d) **Decryption** : De

(e) **Key** : It is used

(f) **Process** :

(1) Sender write

(2) Plain text m

(3) Ciphertext is

(4) The receiver

(g) **Substitution Ci** another letter o

8 NETWORK SECURITY

Marks	
Dec. 03	-
May 04	10 M
Dec. 04	-
May 05	34 M
Dec. 05	24 M
May 06	15 M
Dec. 06	10 M
May 07	10 M

Over time, Network Security has become an essential part of networks. The various encryption standards are discussed here.

8.1 Need for Network Security

(1) Protect data during their transmission.

(2) Guarantee that data transmission are authentic.

8.2 Cryptography

May 05 [Q. 7] *Cryptography.* **(7 M)**

Dec. 05 [Q. 7(d)] *Cryptography.* **(10 M)**

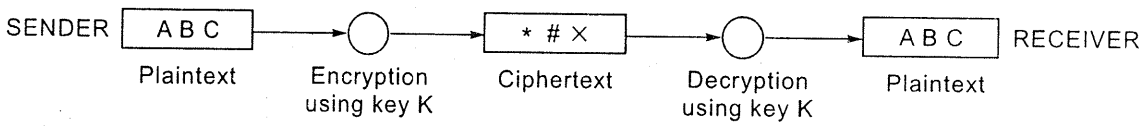


Fig. 8.1: Cryptography.

(a) **Plaintext :** It is the human language that is commonly understood.

(b) **Ciphertext :** Encoded messages which can only be understood by the sender and the intended receiver.

(c) **Encryption :** Encoding plaintext to ciphertext.

(d) **Decryption :** Decoding cipertext to plaintext.

(e) **Key :** It is used to transform plaintext to cipertext and vice versa.

(f) **Process :**

(1) Sender writes message in plaintext.

(2) Plain text message is converted to ciphertext using key K.

(3) Ciphertext is sent over the channel to the receiver.

(4) The receiver uses key K to convert ciphertext back to plaintext.

(g) **Substitution Ciphers :** In this each letter or a group of letters are replaced by another letter or group of letters.

(1) *Monoalphabetic Substitution :* Each alphabet is replaced by an alphabet that is x alphabets after the actual alphabet in the set of english alphabets.

Suppose x = 2

Then A B C D becomes

C D E F

(2) *Polyalphabetic Substitution :* Here substitution depends on

(i) The character itself.

(ii) Position of character in the text.

(h) Transposition Ciphers : In this case the characters do not change, only the position of the characters change.

Process and example :

(1) Select a key. Let the key be COMP.

(2) Number the alphabets in the key according to the way they occur in the set of english alphabets ∴ C O M P gives 1 3 2 4.

(3) Let the message to be sent be "I GOT THE MONEY".

(4) Write the message to be sent in the table as shown and form the ciphertext.

C	O	M	P
1	3	2	4
I	G	0	T
T	H	E	M
O	N	E	Y

1 2 3 4

ITO OEE GHN TMY Cipertext

Fig. 8.2

(i) The art of breaking ciphers is called *cryptanalysis.*

(j) The art of creating ciphers is called *cryptology.*

(k) The types of intruders are :

- **Passive Intruder :** Only listens to messages.
- **Active Intruder :** Can listen, modify and send fraudulent messages.

(l) From the cryptanalyst's point of view, cryptanalysis has 3 variations :

- **Ciphertext - Only Problem :** When the cryptanalyst has ciphertext only and no plain text.
- **Known Plaintext Problem :** When the cryptanalyst has matched some ciphertext and plaintext.
- **Chosen Plaintext Problem :** The cryptanalyst has the ability to encrypt pieces of plaintext.

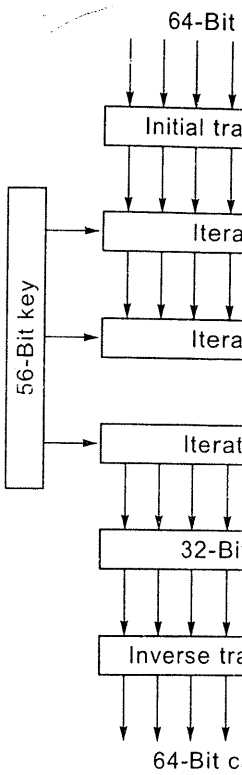
8.3 Symmetric

(1) In this a sing

(2) Only the sen

(3) Basically sen uses the sam *8.1*). E.g. Dat

Data Encryptio



(a) Genera

Refer *figure 8.3(a).*

(1) Provides end

(2) Uses 16 iterati

(3) Plaintext is en

(4) Key can be an

(5) First stage is Then there is

alphabet that is x
abets.

change, only the

y occur in the set

C	O	M	P
I	3	2	4
I	G	0	T
T	H	E	M
O	N	E	Y

IY Cipertext

g. 8.2

essages.

tions :

iphertext only and

as matched some

y to encrypt pieces

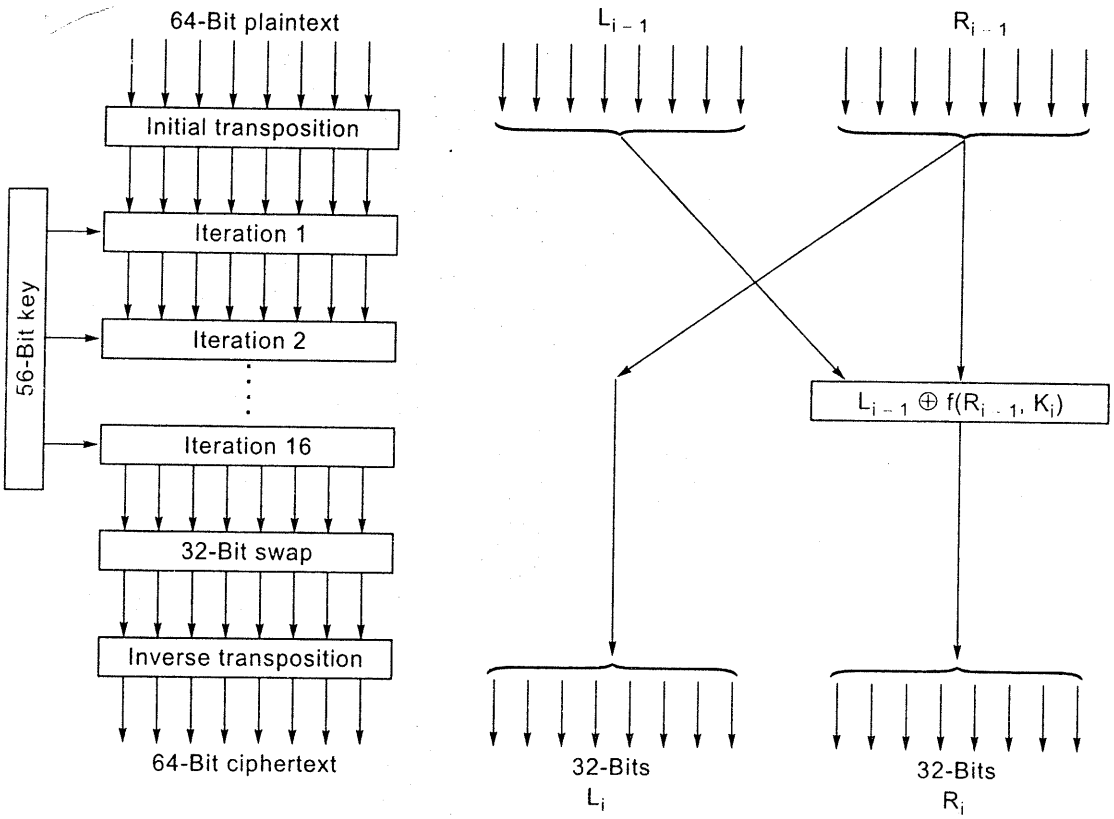
8.3 Symmetric Key Encryption

(1) In this a single key is used for encryption and decryption.

(2) Only the sender and the intended receiver know what the key is.

(3) Basically sender uses the key to change the plaintext to ciphertext. The receiver uses the same key to convert the ciphertext back to the plaintext. (Refer *figure 8.1*). E.g. Data Encryption Standard (DES).

Data Encryption Standard (DES)



(a) General outline. **(b) Detail of one iteration. (⊕ means exclusive OR)**

Fig. 8.3 : The data encryption standard.

Refer *figure 8.3(a)*.

(1) Provides end to end encryption between sender and receiver.

(2) Uses 16 iterations. It has a total of 19 stages.

(3) Plaintext is encrypted as blocks of 64-bits.

(4) Key can be any 56-bit number.

(5) First stage is transposition of 64-bit plain text. This is followed by 16 iterations. Then there is a 32-bit swap where the leftmost 32-bit are exchanged with the

rightmost 32-bits. Last stage is inverse transposition.

(6) Details of a single iteration is shown in *figure 8.3(b)*.

(a) The left output is a copy of the right i/p.

(b) The right output is bitwise X-OR of left input and function of right input and key.

8.4 Public Key Encryption a.k.a. Asymmetric Key Encryption

May 05 [Q. 3(a)], Dec. 05 [Q. 1(b)], May 07 [Q. 4(b)] *Explain RSA algorithm with suitable example for public key security.* **(10 M)**

May 06 [Q. 7(6)] *Write short note on RSA.* **(5 M)**

(1) In this 2 different keys are used for encryption and decryption.

(2) Sender encrypts the plaintext using a key known as the "*Public Key*".

(3) The receiver decrypts the ciphertext using a key known as "*Private Key*". No other key can be used for decryption.

E.g. The RSA algorithm.

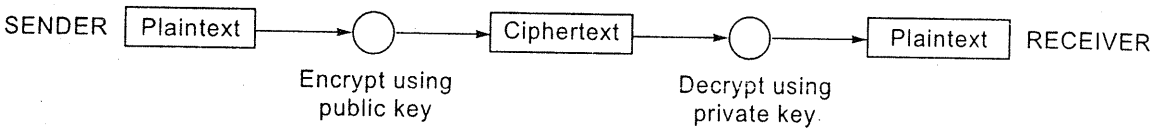


Fig. 8.4

The RSA Algorithm

(1) It was developed by Ron Rivest, Adi Shamir and Len Adleman. (Hence the name RSA)

(2) The *public key* is known by everyone. The *private key* is known only by the receiver. Therefore only the intended receiver having the *private key* can decrypt the ciphertext.

(3) **Algorithm**

(a) Choose two large prime numbers p and q.

(b) Compute $n = p \times q$

and $z = (p - 1) \times (q - 1)$

(c) Chose a number relatively prime to z i.e. which is not a factor of z. Call this the decryption key 'd'.

(d) Find encryption key 'e' such that

$e \times d = 1 \bmod z$.

E.g. **Step (a)** Let $p = 7$ and $q = 17$

Step (b) Calculate n and z

$$n = p \times q = 7 \times 17 = 119$$

$$z = (p - 1) \times (q - 1) = 6 \times 16 = 96$$

Step (c) Se

Let

Step (d) To

e ×

∴ e ×
96

O

e ×

Now we tak

For k = 4 w

8.5 Digital Cert

May 05 [Q. 7] *Dig*

Dec. 05 [Q. 2(c)]

Dec. 06 [Q. 7(1)]

(1) A digtal certif

(2) A digital cert
Authority (CA)

(3) A CA provide
checked the id

(4) Eg. of CA's are

(5) The figure 8
information :
Validity, ID nu

Step (c) Select 'd' such that it is relatively prime to z i.e. it is not a factor of z.

$$z = 96 = 2 \times 2 \times 2 \times 2 \times 2 \times 3$$

Let $d = 5$.

Step (d) To find 'e' such that

$$e \times d = 1 \bmod z$$

$$\therefore \frac{e \times d}{96} = k + \frac{1}{96}$$

Quotient, Remainder

On multiplying by 96 we get

$$e \times d = 96\,k + 1$$

$$e = \frac{96\,k + 1}{5}$$

Now we take the least value of k which gives e as an integer.

For k = 4 we get e = 77.

8.5 Digital Certificate

May 05 [Q. 7] *Digital Certificates.* **(7 M)**

Dec. 05 [Q. 2(c)] *Digital Certificates.* **(4 M)**

Dec. 06 [Q. 7(1)] *Write short note on Digital certificates.* **(5 M)**

(1) A digtal certificate is used for authenticating a Web client or a Web server.

(2) A digital certificate is issued by a reliable authority called the *Certification Authority* (CA).

(3) A CA provides a digital certificate to a requester only when it has throughly checked the identity of the requester.

(4) Eg. of CA's are : VeriSign, Satyam Infoway, Cybertrest etc.

(5) The figure 8.5 shows the layout of the certificate. It has the following information : Holders name, CA's name and signature, Public key of holder, Validity, ID number.

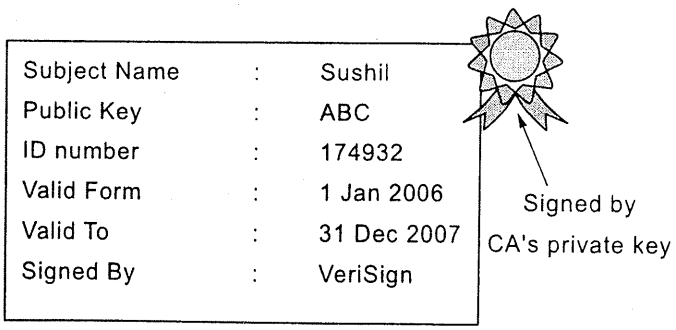


Fig. 8.5 : Digital certificate.

(6) Working of Certificate

(i) If customer 'C' wants to send his credit card details to shop 'S'; C will want to verify the identity of S.

(ii) C will ask S for S's digital certificate.

(iii) S sends its digital certificate to C.

(iv) Now the certificate sent has the *public key* of S.

(v) C uses this public key of S to encrypt his credit card details and sends the encrypted details to S.

(vi) S can decrypt the credit card details of C using its *private key*.

(vii) In this way only S will be able to decrypt the credit card details of C.

8.6 Digital Signatures

May 06 [Q. 1. b(ii)] *Suppose Bobby has got message M, private key sk1 and public key pk1 and Bob has got private key sk2 and public key pk2. Bobby computes* $x = E_{sk1}(M)$, $y = E_{pk2}(H(M))$ *where E is encryption and H is hash. Now she sends this (x, y) to Bob. State the security goals achieved and not achieved.* **(5 M)**

It involves the standard process of encryption and decryption.

Working : Let the sender be A and the receiver be B.

(1) A encrypts the plaintext to ciphertext using the public key of B.

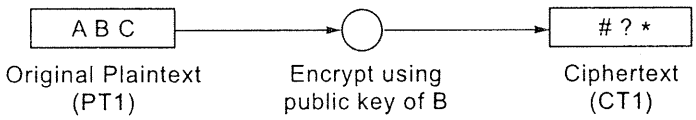


Fig. 8.6

(2) Now A does not directly send the ciphertext to B. It first creates a *Message Digest* (MD) by using a *Hashing Function* on the plaintext. This message digest is encrypted with A's private key. This is called the *Digital Signature*.

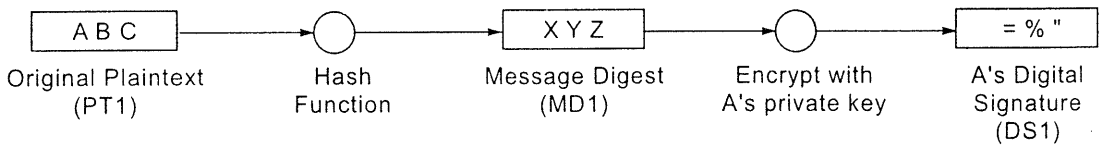


Fig. 8.7

(3) Now A concatenates the ciphertext (CT1) and the digital signature (DS1) and sends it to B. B receives the message as CT2 and DS2.

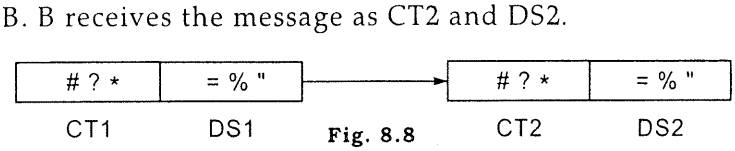


Fig. 8.8

(4) B decrypts c

(5) Now to chec
B decrypts
digest. Let u

(6) B applies the
MD3 and MI
Plain

(7) Thus we can
Integrity, M
achieve the
point will be

8.7 Firewall —

(1) Consider Or
accounts dat

(2) In order to
being.

(3) **Definition :**
barrier betwe

8.8 Social Issu

(1) **Privacy :** It i
can see abou

'S'; C will want

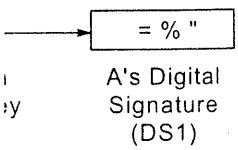
.s and sends the

ails of C.

---

1 and public key
utes x = $E_{sk1}(M)$,
his (x, y) to Bob.
(5 M)

a *Message Digest*
essage digest is
e.



ıature (DS1) and

(4) B decrypts ciphertext CT2 into plaintext PT2 using its private key.

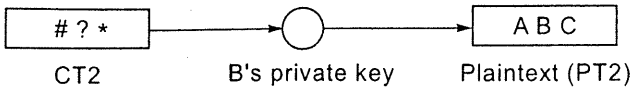


Fig. 8.9

(5) Now to check if the message actually came from A, B does the following :

B decrypts A's Digital Signature using A's public key. This gives B the message digest. Let us call the message digest as MD2.

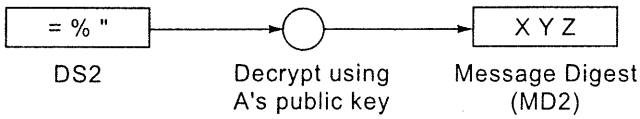


Fig. 8.10

(6) B applies the Hash Function to PT2 to get Message Digest MD3. Now B checks if MD3 and MD2 are the same. If yes then the message was actually sent by A.

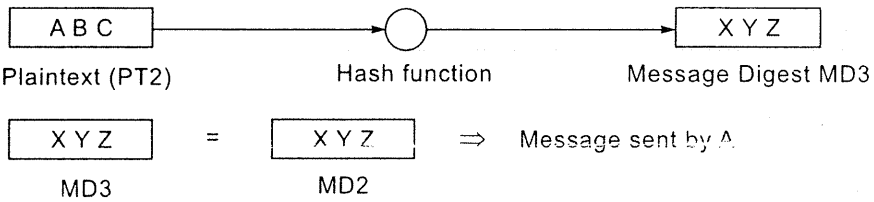


Fig. 8.11

(7) Thus we can see that Digital Signatures achieves the security goals of Message Integrity, Message Authentication and Message Nonrepudiation. It does not achieve the goals of Message Confidentiality and Entity Authentication (This point will be covered in detail in section 8.9).

8.7 Firewall

(1) Consider Online Banking System. It is possible for a hacker to hack into the accounts data base and make malicious changes.

(2) In order to prevent these kinds of attacks the concept of firewall came into being.

(3) **Definition :** A *firewall* is a set of routers with additional facilities that act as a barrier between to or more network segments.

8.8 Social Issues

(1) **Privacy :** It is the term used to show that individuals want to limit what others can see about them.

(2) **Freedom of Speech :** Individuals can say, write and publish whatever they want. Its opposite is **Censorship :** The government controls what people can read, write and publish. Generally the following sites are censored/banned on the internet.

(a) Adult content.

(b) Anti-government propaganda.

(c) Religious propaganda that may lead to unrest.

(3) **Copyright (The Right to Copy) :** It is a set of exclusive rights granted by governments to regulate use of an idea. Generally these rights are of limited duration. Symbol ©.

8.9 Network Security Services

May 04 [Q. 5(b)] *What are the issues in communication security ?* **(10 M)**

May 05 [Q. 5(a)] *Explain various issues in network security.* **(10 M)**

Network security provides the following services :

(1) Message Confidentiality.

(2) Message Integrity.

(3) Message Authentication.

(4) Message Nonrepudiation.

(5) Entity Authentication.

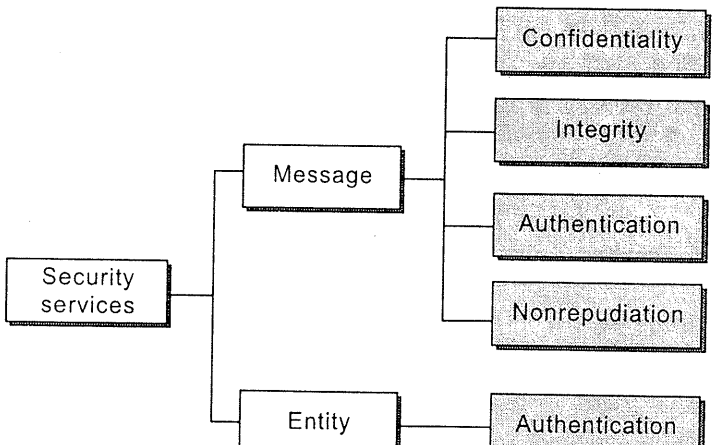


Fig. 8.12 : Security services related to the message or entity.

[I] Message Confidentiality :

(a) This means that the transmitted message must make sense only to the intended receiver.

(b) To achieve t
the receiver.

(c) This can be c

(i) Symmetri

(ii) Asymmet

[II] Message Int

(a) This means th

(b) The need for
Bob writes a
the will but n
we do not nee

(c) Integrity can
Message Diges
Message Dige

[III] Message Aut

(a) In this the rec
be sure that th

(b) To Provide m
(MAC) which
point number

[IV] Message Non

(a) This means th
did send.

(b) Message Nonr
section 8.6.

[V] Entity Authe

(a) This means tha
system resourc

(b) This can be imp

(i) **Passwords**
given acce

(ii) **Challenge**
is known c
the entity
number an

(b) To achieve this the message must be encrypted at the sender and decrypted at the receiver.

(c) This can be done using :

(i) Symmetric key encryption. (Refer section 8.3)

(ii) Asymmetric key encryption. (Refer section 8.4)

[II] Message Integrity :

(a) This means that data must reach the receiver exactly as it was sent.

(b) The need for integrity can be understood with the following example : Suppose Bob writes a will. Now the will is made public i.e. people have to be told about the will but no one should be allowed to change the will. Therefore in this case we do not need confidentiality but we need integrity.

(c) Integrity can be maintained by using a *Hash functions* on the plaintext to get a *Message Digest* which is covered in section 8.6 point number 2 and 6. The Message Digest provides Message Integrity.

[III] Message Authentication :

(a) In this the receiver needs to be sure of the senders identity i.e. the reciever must be sure that the message is not being sent by an imposter.

(b) To Provide message authentication we need a Message Authentication Code (MAC) which is obtained by using a key on a Message Digest. (Refer section 8.6 point number 2, 5 and 6)

[IV] Message Nonrepudiation :

(a) This means that a sender must not be able to deny sending a message he or she did send.

(b) Message Nonrepudiation can be obtained by using digital signature covered in section 8.6.

[V] Entity Authentication :

(a) This means that the entity (user) should be verified before giving it access to the system resources.

(b) This can be implemented in the following ways :

(i) **Passwords** : The user must login by providing a password before he/she is given access to the system resources.

(ii) **Challenge - Response** : In this case each entity has a unique function which is known only by the entity and the system. The system sends a *challenge* to the entity which is a random number. The entity applies function to this number and sends back the answer given by the function, as the *response*.

Now the system checks the anser; if the answer is correct the entity is given access to the system resources.

(iii) Digital Certificates : Covered in section 8.5.

Note : From the above section we see that Digital Signatures provide Message Integrity, Message Authentication and Message Nonrepudiation.

8.10 Key Management

May 06 [Q. 4(b)] *How to achieve authentication with secret key ? Justify it (Refer section 8.4 and 8.6). also describe authentication with KDC mechanism. (Refer section 8.10)* **(10 M)**

In Symmetric and Asymmetric Encryption both the sender and the receiver have keys. *How are these keys given to the sender and the receiver ?*

(a) Distribution of Keys in Symmetric Key Encryption :

(i) Symmetric Key Encryption needs a shared secret key between the sender and reciver.

(ii) The distribution of this key creates a problem : Suppose 'X' wants to communicate with 1 million people, he will need to exchange secret keys with 1 million people. This cannot be done over the Internet as it is not a secure channel. Instead a process called KDC is used.

(iii) KDC (Key Distribution Center) :

In this method a secret key is established between the KDC and each member. E.g. The secret key between A and KDC is K_A.

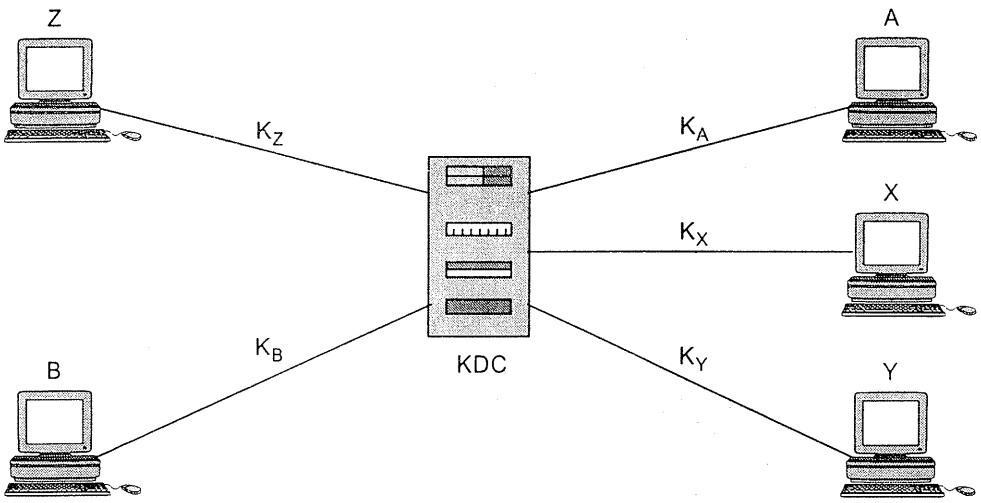


Fig. 8.13 : KDC.

entity is given

vide Message

---

ustify it (Refer
(Refer section
(10 M)

receiver have

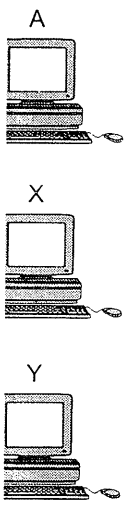
en the sender

'X' wants to
ge secret keys
: as it is not a

DC and each

- Suppose A wants to send a confidential message to B the following process will be followed :
  - The aim of this process is for both A and B to be given K_{AB} which they can use to send messages to each other using Symmetric Key Encryption method. K_{AB} is called *Session Key*.
  - **Step 1** : A sends a message to the KDC saying that it wants to establish a Symmetric Session betwee 'B' and itself. The message contains the retgistered identity of A and B.
  - **Step 2** : KDC receives the message and sends a message to A which is encrypted using K_A. The message contains 2 things :
    (i) K_{AB}.
    (ii) Ticket : The ticket consists of the registered identities of A and B along with KAB. The ticket is encrypted using K_B.

    When 'A' receives this message, it decrypts the message to get K_{AB} and the ticket. 'A' cannot decrypt the Ticket because it is encrypted using K_B.
  - **Step 3** : Now A sends the Ticket to B. B decrypts the Ticket and gets K_{AB}.
  - Now as A and B both have K_{AB}. They can use Symmetric Key Encryption method using key as K_{AB}.



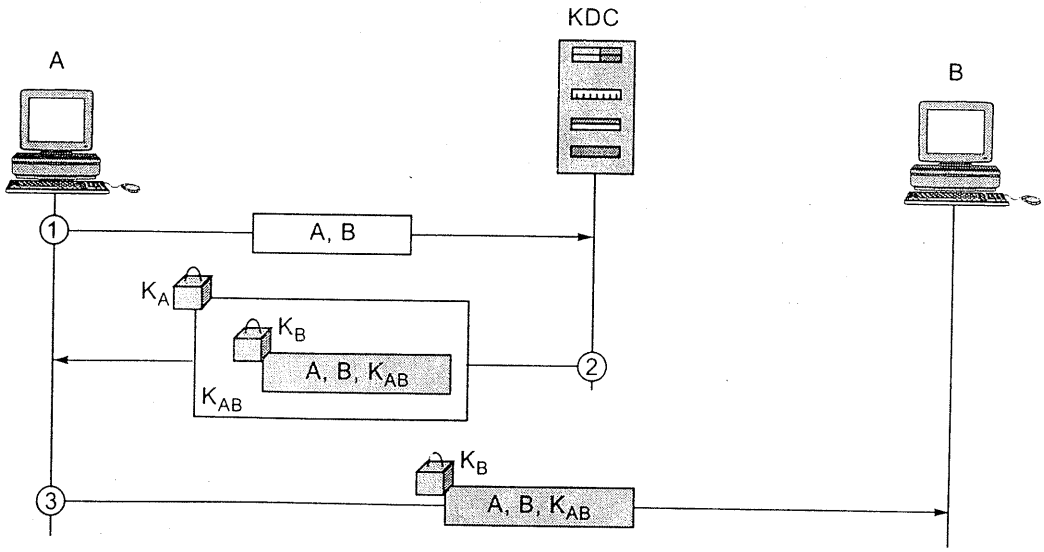


Fig. 8.14 : Creating a session key between A and B using KDC.

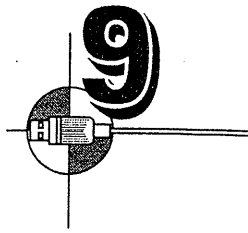
(b) Distribution of Keys in Asymmetric Key Encryption :

(i) In Asymmetric Key Encryption technique there is a public key and a private key. The private key should be known only by the user.

The public key should be known by everyone who wants to send a message to the user.

(ii) The method to distribute the public key are as follows :

- *Public Announcement :* In this case the public key of a user is announced publicly. This is done by putting your public key on your web site for all to see.
- *Trusted Center :* A more secure approach is to have a Trusted Center keep a directory of all public keys. This center will entertain requests for public keys.
- *Certification Authority :* The previous method puts a huge load on the Trusted Centers. A better way is for a Certification Authority to give Digital Certificates to the users. The public key can be obtained from the Digital Certificate of the user. (Refer section 8.5).

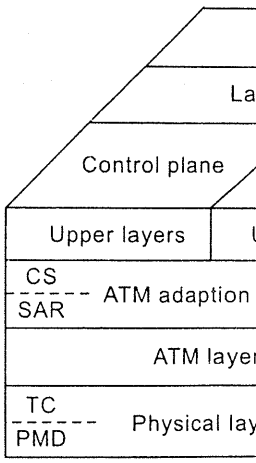


The ATM prot
of networks. We w
ATM architecture a

9.1 What is ATM

ATM (Asynch
interconnection of

9.2 The ATM Re



The ATM refe
and TCP/IP whic
defined as being t
ATM layer, and the
that.

9 ATM

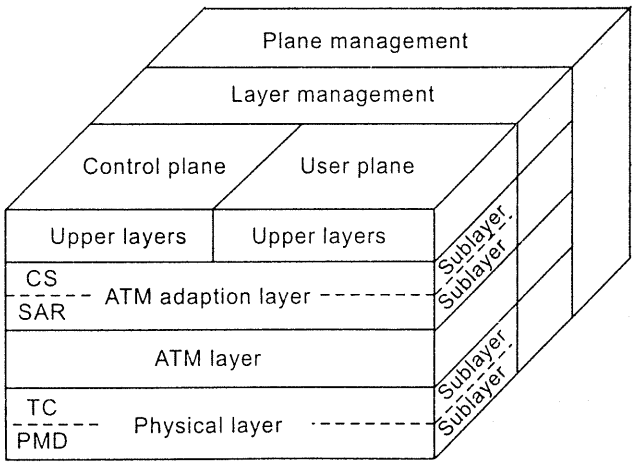
Marks	
Dec. 03	-
May 04	-
Dec. 04	-
May 05	-
Dec. 05	-
May 06	10 M
Dec. 06	-
May 07	10 M

The ATM protocol facilitates high speed interconnection of networks. We will now see the ATM Reference Model, the ATM architecture and the ATM Header formats.

9.1 What is ATM ?

ATM (Asynchronous Transfer Mode) is a protocol which allows high speed interconnection of all the worlds networks.

9.2 The ATM Reference Model



CS : Convergence sublayer
SAR : Segmentation and reassembly sublayer
TC : Transmission convergence sublayer
PMD : Physical medium dependent sublayer

Fig. 9.1 : The ATM reference model.

The ATM reference model is shown in the figure above. Unlike the earlier OSI and TCP/IP which were two dimensional reference models, the ATM model is defined as being three-dimensional. It consists of three layers, the Physical layer, the ATM layer, and the ATM adaptation layer, plus whatever users want to put on top of that.

(a) Physical Layer

- The physical layer deals with the physical medium: voltages, bit timing, and various other issues.
- ATM cells can be transmitted independent of the transmission medium.
- It is divided into two regions
  (1) PMD : Physical Medium Dependent sublayer
  (2) TC : Transmission Convergence sublayer.

(b) ATM Layer

The ATM layer deals with cells and cell transport.

(1) It defines the layout of a cell and tells what the header fields mean.

(2) It deals with establishment and release of virtual circuits.

(3) It performs congestion control.

(c) AAL (ATM Adaptation Layer)

The higher layers allow users to send packets larger than a cell. The AAL segments these packets, transmits the cells individually, and reassembles them at the other end.

It is divided into 2 sublayers :

(1) SAR : Segmentation And Reassembly sublayer.

(2) CS : Convergence Sublayer.

(d) User Plane

The user plane deals with

(1) Data transport

(2) Flow control

(3) Error correction.

(e) Control Plane

The control plane is concerned with connection management.

(f) Layer and Plane Management

The layer and plane management functions relate to resource management and interlayer coordination.

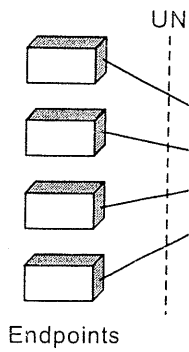
The various la
table :

OSI layer	AT lay
3/4	AA
2/3	AT
2	
1	Phys

Fig.

9.3 Architecture

- The user device
  interface (UNI)
  through netwo



, bit timing, and

ı medium.

mean.

ı cell. The AAL
sembles them at

ıanagement and

The various layers and sublayers with their functions are listed in the following table :

OSI layer	ATM layer	ATM sublayer	Functionality
3/4	AAL	CS	Providing the standard interface(convergence)
		SAR	Segmentation and reassembly
2/3	ATM		Flow control Cell header generation/extraction Virtual circuit/path management Cell multiplexing/demultiplexing
2	Physical	TC	Cell rate decoupling Header checksum generation and verification Cell generation Packing/unpacking cells from the enclosing envelope Frame generation
1		PMD	Bit timing Physical network access

Fig. 9.2 : The ATM layers and sublayers, and their functions.

9.3 Architecture

- The user devices, called the endpoints, are connected through a user-to-network interface (**UNI**) to the switches inside the network. The switches are connected through network-to-network interfaces (**NNIs**).

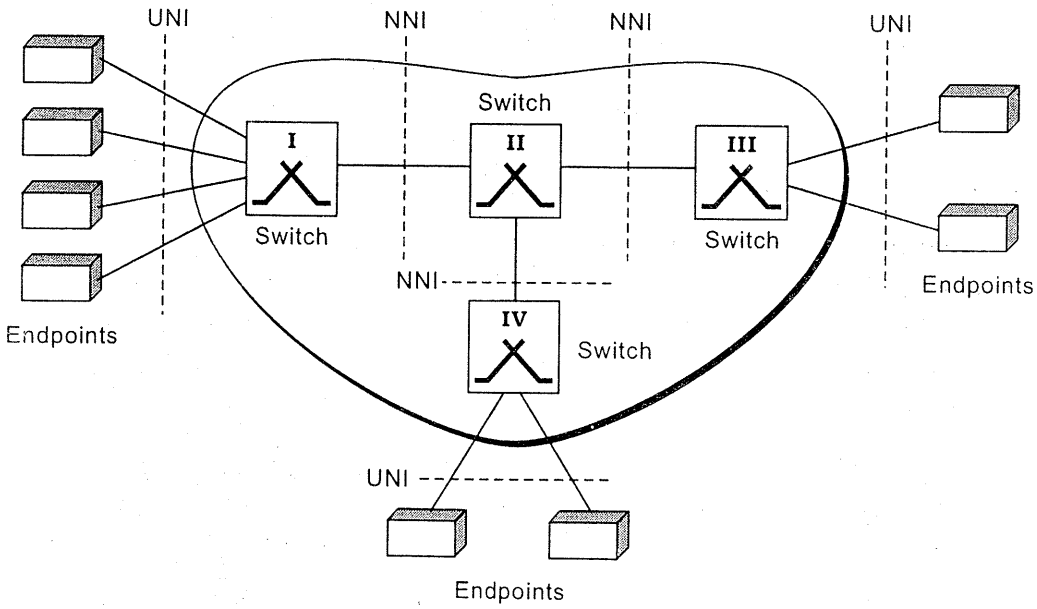


Fig. 9.3 : Architecture of an ATM network.

- **Virtual Connection**

  Connection between two endpoints is accomplished through

  - *Virtual Circuits (VCs) :* VCs logically connect two points. All cells belonging to a single message follow the same virtual circuit and remain in their original order until they reach their destination.
  - *Virtual Paths (VPs) :* A VP is a combination of virtual circuits that are bundled together because parts of their paths are the same.
  - *Transmission Paths (TPs) :* A combination of VPs is a TP. A transmission path (TP) is the physical connection (wire, cable, satellite, and so on).

  Practical Example :

  - Think of two switches as two cities. A transmission path is the set of all highways that directly connect the two cities. Think of a virtual path as one single highway that connects two cities. Think of a virtual circuit as the lanes of a highway.

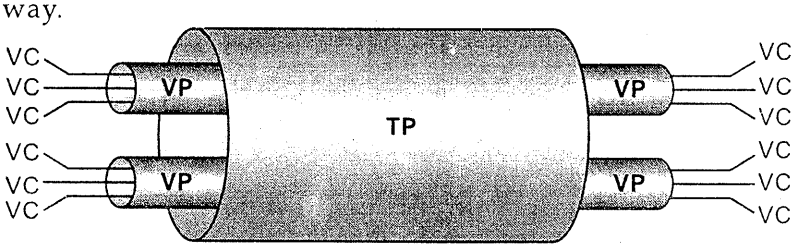


Fig. 9.4 : TP, VPs and VCs.

- **Identifiers**

  In a virtual circuit network, to route data from one endpoint to another, the virtual connections need to be identified. For this purpose, the designers of ATM created a hierarchical identifier with two levels: a virtual path identifier (VPI) and a virtual-circuit identifier (VCI). The VPI defines the specific VP, and the VCI defines a particular VC inside the VP. The VPI is the same for all virtual connections that are bundled (logically) into one VP.

- **Bit lengths of VPI and VCI**

  The lengths of the VPIs for UNIs and NNIs are different. In a UNI, the VPI is 8-bits, whereas in an NNI, the VPI is 12-bits. The length of the VCI is the same in both interfaces (16-bits). We therefore can say that a virtual connection is identified by 24-bits in a UNI and by 28-bits in NNI

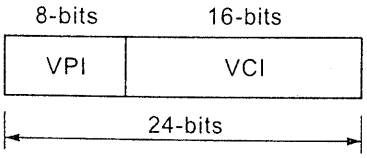


Fig. 9.5(a) : VPI and VCI in a UNI

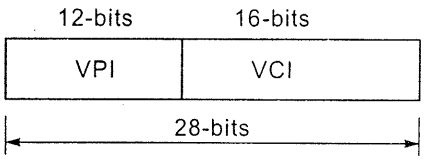


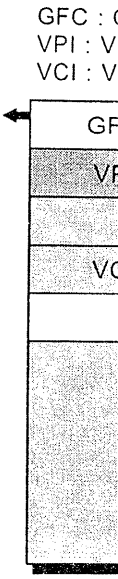
Fig. 9.5(b) : VPI and VCI in an NNI

- **Cells**

  The basic da
  long with 5 b

9.4 Header For

ATM uses tw
and another for n



(a) **Generic Flow**
UNI level.

(b) **Virtual Path I**
field in an NN

(c) **Virtual Circui**

(d) **Payload Type**

(e) **Cell Loss Prio**

(f) **Header Error**

- **Cells**

The basic data unit in an ATM network is called a cell. A cell is only 53 bytes long with 5 bytes allocated to the header and 48 bytes carrying the payload.

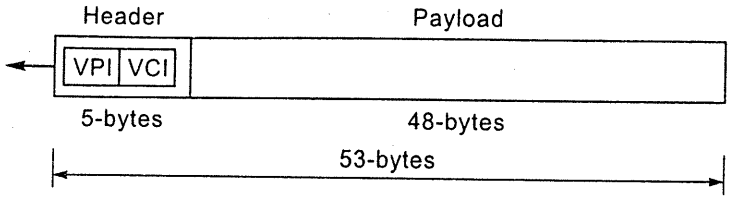


Fig. 9.5(a) : An ATM cell.

9.4 Header Format

ATM uses two formats for the header, one for user-to-network interface (UNI) and another for network-to-network interface (NNI) cells.

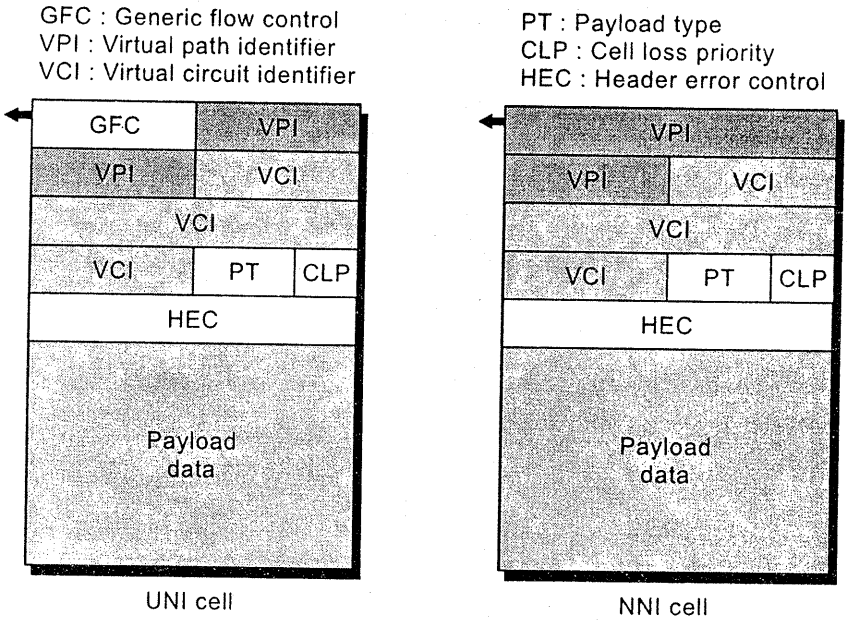


Fig. 9.6 : ATM headers.

(a) **Generic Flow Control (GFC)** : The 4-bit GFC field provides flow control at the UNI level.

(b) **Virtual Path Identifier (VPI)** : The VPI is an 8-bit field in a UNI cell and a 12-bit field in an NNI cell .

(c) **Virtual Circuit Identifier (VCI)** : The VCI is a 16-bit field in both frames.

(d) **Payload Type (PT)** : It defines the payload as userdata or managerial information.

(e) **Cell Loss Priority (CLP)** : The 1-bit CLP field is provided for congestion control.

(f) **Header Error Correction (HEC)** : It is a CRC.

9.5 ATM Adaptation Layer

May 06 [Q. 7(a)], May 07 [Q. 2(b)] *Explain ATM Adaptation Layer. Also describe VPI and VCI concept. (Also refer to section 9.2, 9.3)* **(10 M)**

The ATM Adaptation Layer (AAL) relays ATM cells between ATM Layer and higher layers.

(a) When relaying information received from the higher layers, it segments the data into ATM cells.

(b) When relaying information received from the ATM Layer, it must reassemble the payloads into a format that higher layers can understand.

This operation, which is called *Segmentation and Reassembly (SAR)*, is the main task of AAL.

Different AALs were defined to support different traffic or service expected. The service classes and the corresponding types of AALs were as follows :

AAL1 : *Constant Bit Rate (CBR) service*

It supports a connection-oriented service in which the bit rate is constant. Examples of this service include 64-Kbit/sec voice, fixed-rate uncompressed video and leased lines for private data networks.

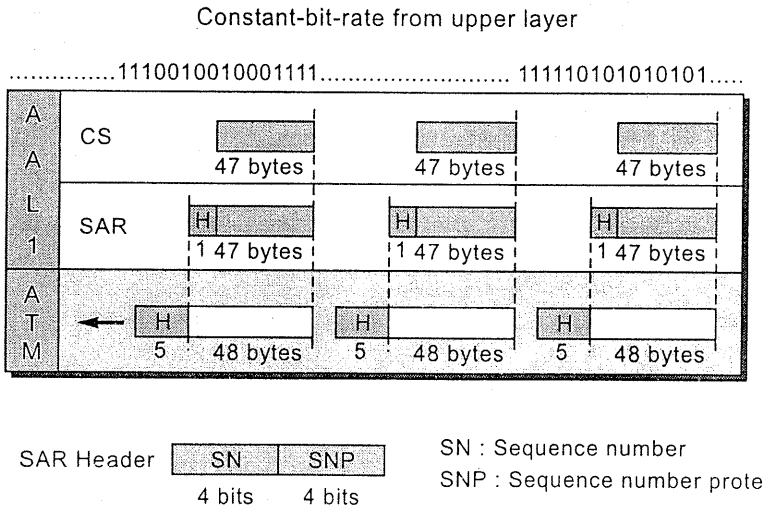


Fig. 9.7 : AAL1.

(a) Sequence Number (SN) : used to order the bits.

(b) Sequence Number Protection (SNP) : Used to protect SN.

AAL2 : *Variable*

It suppo
but req
include
bounde
original

f

A
A
L
2
A
T
M

CS Heade

SAR Head

(a) CID
(b) LI
(c) PPT
(d) UUI
(e) HEC
(f) SF

AAL3/4 : *Adaptatio*

This AAL
but not t
for conn
addressi

describe VPI
(10 M)

yer and higher

ments the data

reassemble the

ne main task of

expected. The

ate is constant.
e, fixed-rate
orks.

ection

N.

AAL2 : *Variable Bit Rate (VBR) service*

It supports a connection-oriented service in which the bit rate is variable but requires a bounded delay for delivery. Examples of this service include compressed packetized voice or video. The requirement on bounded delay for delivery is necessary for the receiver to reconstruct the original uncompressed voice or video.

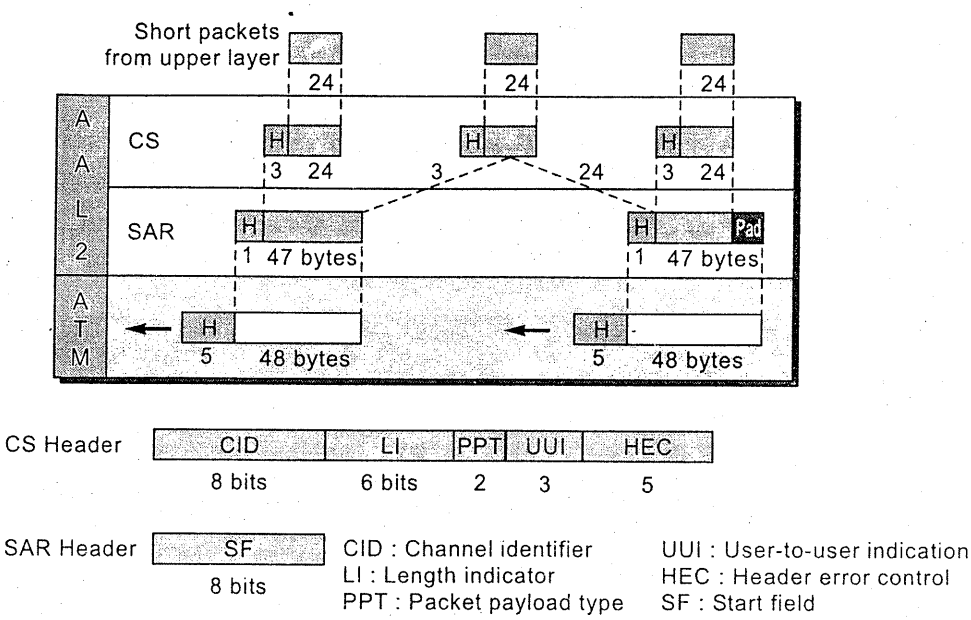


Fig. 9.8 : AAL2.

(a) CID : Defines channel.
(b) LI : Indicates how much of final packet is data.
(c) PPT : Defines type of packet.
(d) UUI : Used by end-to-end users.
(e) HEC : Corrects errors in the header.
(f) SF : Defines offset from beginning of packet.

AAL3/4 : *Adaptation for data services*

This AAL is recommended for transfer of data which is sensitive to loss, but not to delay. The AAL may be used for connection oriented as well as for connectionless services, since functions like routing and network addressing are performed on the network layer.

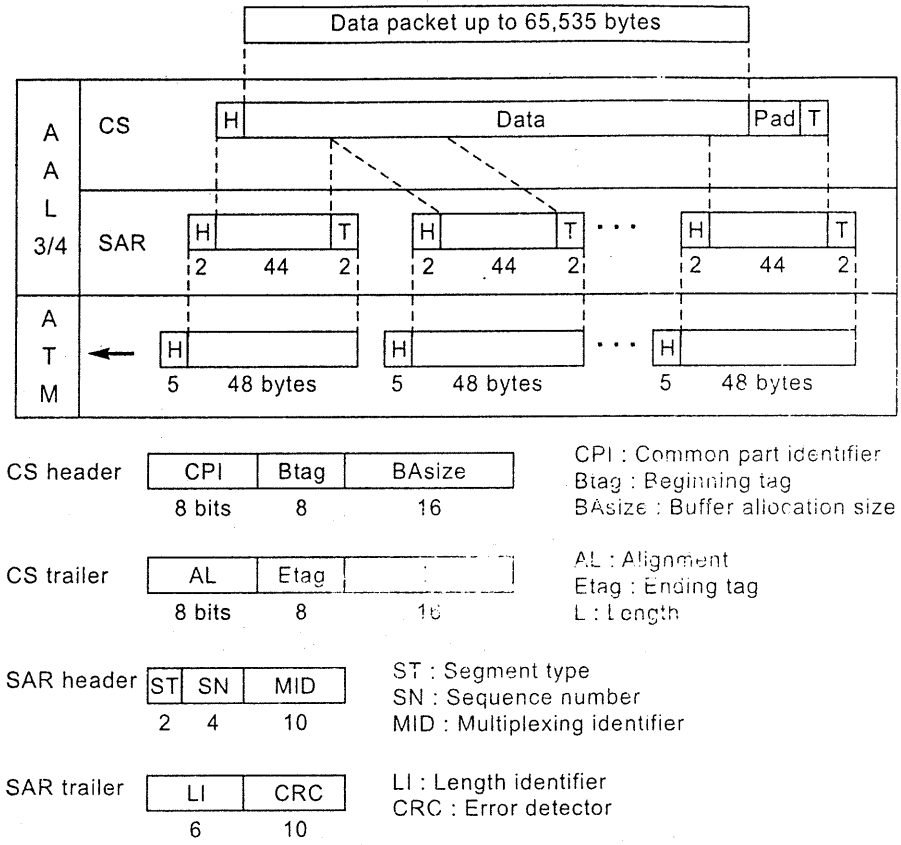


Fig. 9.9 : AAL3/4.

(a) CPI : Defines how subsequent fields are to be interpreted.

(b) B tag : Identifies cells belonging to same packet.

(c) BA size : Tells receiver what size of buffer is needed for incoming data.

(d) AL : Used to make trailer 4 bytes long.

(e) E tag : Serves as an Ending Tag.

(f) L : Indicates length of data unit.

(g) ST : Specifies position of the segment in the message.

(h) SN : Same as before.

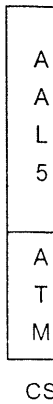
(i) MID : Identifies cells coming from different data flows and multiplexed on same connection.

(j) LI : Defines how much of packet is data.

(j) CRC : CRC for entire data unit.

AAL5 : *Adap*

This
servi
detec



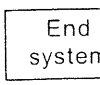
(a)
(b)
(c)
(d)

9.6 PNNI Rou

(1) Definition
that provid

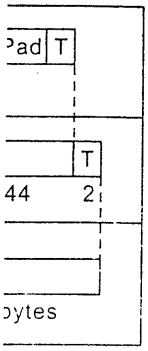
(2) It is how r

(3) It is a swit



(4) Features

- Used ir
- Has mu
- At conn
- Topolog
done a



rt identifier
tag
location size

AAL5 : *Adaptation for data services*

This AAL is recommended for high speed connection oriented data service. This AAL offers a service with less overhead and better error detection.

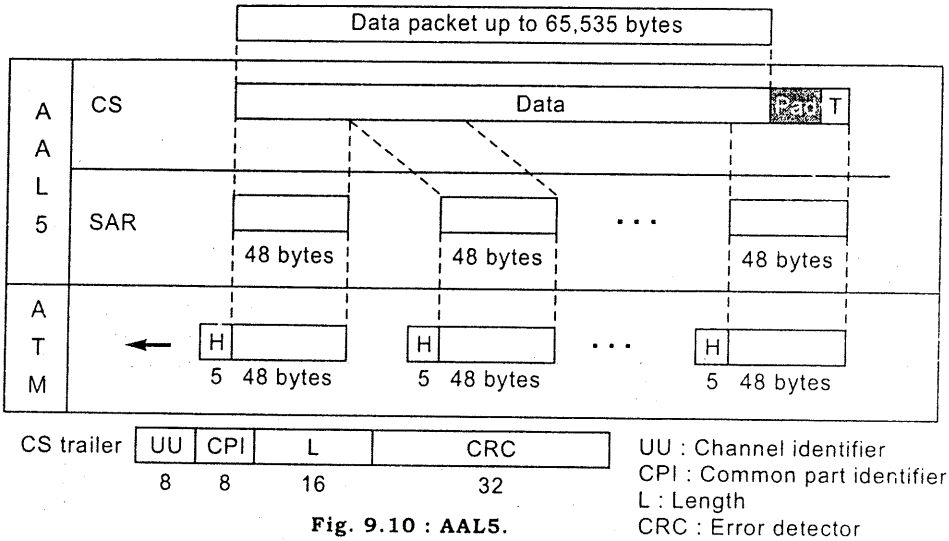


Fig. 9.10 : AAL5.

(a) UU (User to User) : Used by end users.
(b) CPI : Defined before.
(c) L : Gives length of original data.
(d) CRC : Used for error control on entire data unit.

9.6 PNNI Routing

preted.

d for incoming

(1) **Definition** : Private Network to Network Interface is a link state routing protocol that provides dynamic ATM routing with QoS support.

(2) It is how routing is done in ATM networks.

(3) It is a switch-switch interface in a private network.

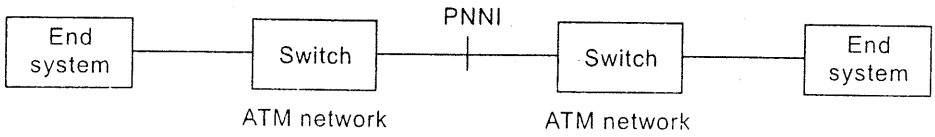


Fig. 9.11

age.

ata flows and

(4) **Features**

- Used in point-to-point or point-to-multipoint connections.
- Has multiple levels of hierarchy.
- At connection setup it can route around failed components.
- Topology need not be manually fed into the nodes. Topology discovery is done automatically at the nodes.

(5) **Addressing**

- Addresses are 20 bytes long.

(6) PNNI routing uses Link State Routing.
{For Link State Routing refer section (5.4 [IV])}

(7) **Hierarchical Division**

- In case of a small network we don't need hierarchical division and all the nodes are in the same domain.

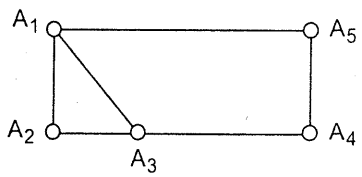


Fig. 9.12 : Small network.

- In case of large networks, the network is divided into many domains as shown.

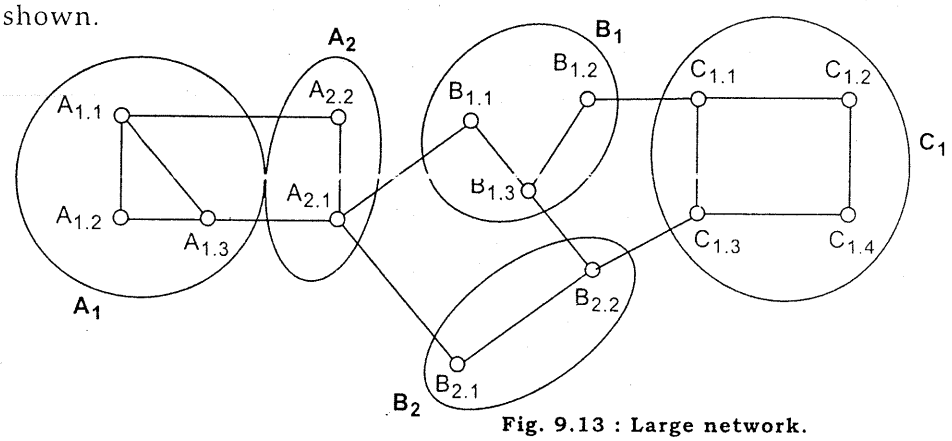


Fig. 9.13 : Large network.

As shown in the *figure 9.13* we have 3 domains named by A, B and C. These domains can be further divided into subdomains as shown.(A_1, A_2, B_1, B_2, C_1)

(8) **Terminology**

(a) **Peer Group :** A group of nodes at the same hierarchy.

(b) **Border Node :** Node that crosses the boundary of a peer group.

(c) **Child Node :** Nodes at next lower hierarchy level.

(d) **Parent Node :** Node at next higher hierarchy level.

(e) **Links :** Links between nodes.

(f) **Peer Group Leader (PGL) :** A node that represent the entire group at the next higher hierarchy level (need not be a border node).

(g) **PNNI Topology State Element (PTSE) :** Routing information that is flooded in a peer group.

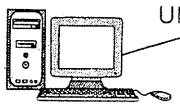
(h) **PNNI Topology State Packet (PTSP) :** A packet containing one PTSE.

(9) PNNI uses
(Basically w
previous no

9.7 ATM Signa

Introduction

- A signali
- An ATM
traffic inf
- At each
connectio
make the
- ATM sig
Network-
is an inte
between



- *ATM Foru*
standardiz
- ATM Foru
network si
- ITU-T only
ITU-T.

(9) PNNI uses *Crank back* and *Alternate routing*.
(Basically when transmission failure takes place the message is sent back to a previous node from where re-routing can be done)

9.7 ATM Signalling

Introduction

- A signaling system is responsible for connection setup and tear-down.
- An ATM signaling system has to convey ATM service parameters such as QoS, traffic information through the ATM network.
- At each network node, resources have to be checked to decide if the connection can be accepted and routing has to be done. These multiple tasks make the ATM signaling system a complicated system.
- ATM signaling consists of User-Network Interface signaling (UNI) and Network-Network Interface signaling (NNI) as shown in the figure. An UNI is an interface between an end system and a network. A NNI is the interface between network nodes (or network clouds).

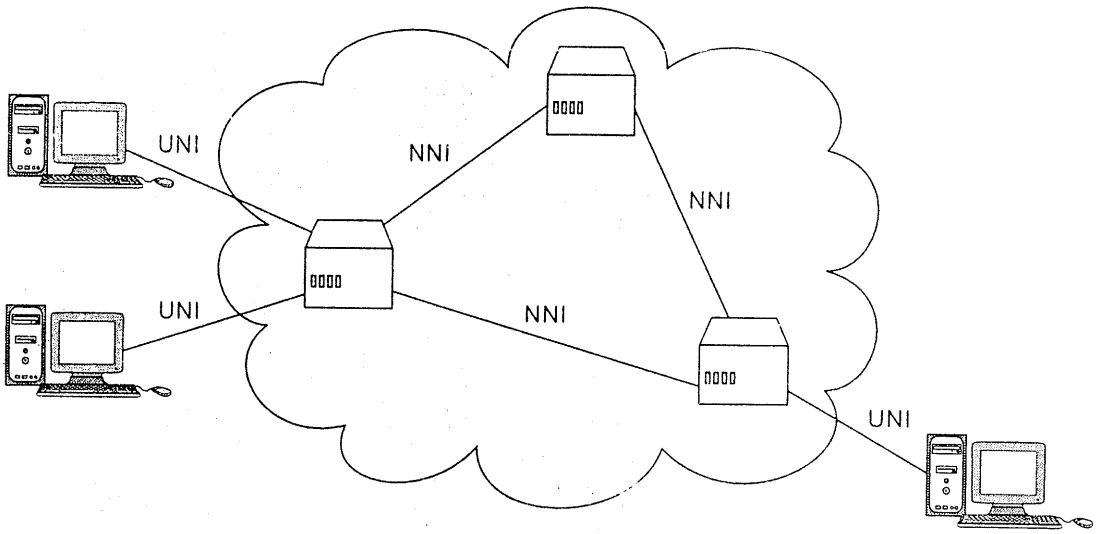


Fig. 9.14

- *ATM Forum* and *ITU-T* are the two bodies which do the ATM signaling standardization.
- ATM Forum specifies UNI3.1, UNI4.0 for UNI signaling, and PNNI for network signaling.
- ITU-T only specifies UNI signaling. No NNI signaling protocol is defined by ITU-T.

- **Signaling Protocols**
- **ATM Forum :**

  UNI 3.0. : The earliest UNI signaling protocol that support only point-to-point connections.

  UNI 3.1. : Significant modification of UNI 3.0. Support point-to-multipoint connections.

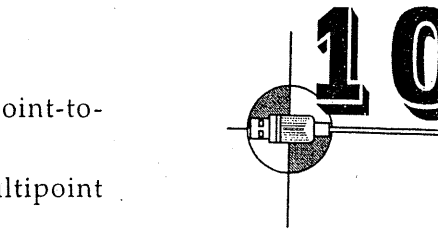
  UNI 4.0 : Support Leaf Initiated Join multipoint connections.

  Private Network-Network Interface (PNNI) for network node signaling .
- ITU-T

  Q931 : Signaling protocol for narrow-band ISDN.

  Q2931 : Point-to-point signaling protocol for both ISDN and B-ISDN.

  Q2971 : Addition to Q2931 to support point-to-multipoint connections.



Now we con
the knowledge
Concepts of Qua
such miscellaneo
questions will als

[I]

Dec. 04 [Q. 4(a
the requirements

Quality of Ser
communications

Quality of servic

(1) Traffic Shap

(2) Flow Charac

(a) *Reliabilit*

Lack of
leads to

(b) *Delay*

Source-to
should b
whereas

(c) *Jitter*

Jitter is d
between

E.g. If 4
packets h
Jitter.

If 4 packe
packet ha

For appli
should be

10 MISCELLANEOUS

Marks	
Dec. 03	-
May 04	-
Dec. 04	10 M
May 05	-
Dec. 05	5 M
May 06	5 M
Dec. 06	23 M
May 07	-

Now we come to this Miscellaneous section which needs the knowledge that we gained in the previous chapters. Concepts of Quality of Service, various formulae and other such miscellaneous theory will be covered here. A few expected questions will also be discussed.

[I]

Dec. 04 [Q. 4(a)] *Briefly explain the primary parameters that are the requirements to provide Quality of Service in networks.* **(10 M)**

Quality of Service can be defined as *"The performance specification of a communications channel or system."*

Quality of service includes the following:

(1) **Traffic Shaping :** (Section 5.6)

(2) **Flow Characteristics :** The Flow Characteristics are:

(a) *Reliability*

Lack of reliability means loosing a packet or an acknowledgement which leads to retransmission.

(b) *Delay*

Source-to-Destination Delay is an important flow characteristic. The delay should be minimum in case of telephony, audio and video conferencing; whereas the delay in file transfer or e-mail can be large.

(c) *Jitter*

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

E.g. If 4 packets depart at times 0, 1, 2, 3 and arrive at times 20, 21, 22, 23, all packets have same delay which is 20 units of time. This is an example of Low Jitter.

If 4 packets depart at times 0, 1, 2, 3 and arrive at times 21, 23, 21, 28, each packet has a different delay : 21, 22, 19, 25. This is an example of High Jitter.

For applications like telephony , audio and video conferencing the jitter should be extremely low.

(d) *Bandwidth*

Different applications need different bandwidths. E.g. Video conferencing needs much higher bandwidth than e-mail.

(3) **Miscellaneous**

(a) *Throughput* : It is the Number of bytes transferred per second.

(b) *Error Ratio* : It is the fraction of error messages in the total number of messages.

(c) *Protection* : Thwarts unauthorized access.

(d) *Priority* : The connections can be prioritized.

[II] Formulae

(1) $$A = \frac{t_p}{t_f}$$

t_f : Single frame transmission time.

t_p : Propagation delay : total time line is engaged in transmission of a single frame.

u : Utilization.

(2) **Stop and Wait Efficiency**

$$\eta = \frac{t_f}{t_f + 2t_p} \times 100\ \%$$

$$t_f = \frac{L}{\text{Bit Rate i.e. Data Rate}}$$

L : Number of bits in a frame.

(3) **Sliding Window Protocol**

$$\text{Window Size } w = \frac{t_f + 2t_p}{t_f}$$

Propagation delay = Round Robin Propogation Delay = $2t_p$

(4) **Stop and Wait Utilization**

$$u = \frac{t_f}{t_f + 2t_p}$$

(5) **Slotted Aloha**

(a) Probability that avoids collision = e^{-G}.

(b) Probability of a collision = $1 - e^{-G}$.

(c) Probability of (k - 1) collisions = Probability of success on k^{th} attempt

$$= e^{-G}(1 - e^{-G})^{(k-1)}$$

(d) Expect

(Wher

(6) **Utilizatio**

μ =

(7) **Pure Aloh**

Channel ba

Usable ba

(8) **Slotted Alo**

Channel ba

Usable ba

Tips

(1) t_f alwa

(2) L alwa

(3) For a sa

(4) If they

then wr

(5) For min

(9) **Noisy Chan**

C = B

C : C

B : Ba

SNR : Si

(10) **Noiseless C**

b = 2

b : Bi

B : Ba

L : Nu

(d) Expected number of transmissions $E = e^G$

(Where G is number of requests per slot.)

(6) Utilization for Pipelining Protocol

$$\mu = \frac{w}{1 + 2A}$$

(7) Pure Aloha

Channel bandwidth = 18.4 %

$$\text{Usable bandwidth} = \text{Rate} \times \frac{18.4}{100}$$

(8) Slotted Aloha

Channel bandwidth = 36.8 %

$$\text{Usable bandwidth} = \text{Rate} \times \frac{36.8}{100}$$

Tips

(1) t_f always in bits.

(2) L always in Kbps.

(3) For a satellite channel ; $2t_p$ = 540 msec.

(4) If they give "3-bit sequence numbers are used" then write k = 3, $w_{max} = 2^k - 1 = 8 - 1 = 7$.

(5) For minimum frame size in CSMA/CD ; $t_f = 2t_p$.

(9) Noisy Channel : Shannon Capacity

$C = B \log (1 + SNR)$ (log is to base 2)

C : Capacity of the channel in bps.

B : Bandwidth.

SNR : Signal to Noise Ratio.

(10) Noiseless Channel : Nyquist Bit Rate

$b = 2 B \log L$ (log is to base 2)

b : Bit rate.

B : Bandwidth.

L : Number of levels.

[III]

Dec. 05 [Q. 5(b)] *We have a channel with a 1 MHz bandwidth. The SNR for this channel is 63; what is the appropriate bit rate and signal level ?* **(5 M)**

Solution : First, we use the Shannon formula to find our upper limit.

$$C = B \log_2 (1 + SNR) = 10^6 \log_2 (1 + 63) = 10^6 \log_2 (64) = 6 \text{ Mbps}$$

The Shannon formula gives us 6 Mbps as the upper limit. For better performance we use something lower, 4 Mbps for example. Then we use the Nyquist formula to find the number of signal levels.

$$4 \text{ Mbps} = 2 \times 1 \text{ MHz} \times \log_2 L$$
$$\therefore L = 4$$

[IV]

Explain Pipelining in the Sliding Window Protocol.

- In the 1-bit Sliding Window protocol the sender sends a frame , then waits for an acknowledgement before sending the next frame. This waiting leads to low efficiency.
- If we allow the sender to transmit more than 1 frame before waiting for an acknowledgement the efficiency increases.
- The best situation will be if the sender keeps sending something without filling up the receivers buffer, until the ack for the initial frame arrives.

[V]

Service Models

(1) Service models are of two types :
  (a) Connection Oriented Service Models (CONS)
  (b) Connection-less Service Models (CLNS)

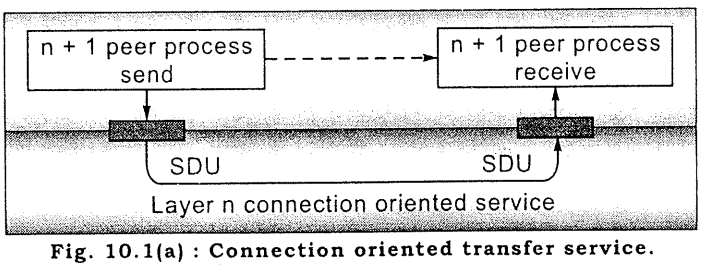


Fig. 10.1(a) : Connection oriented transfer service.

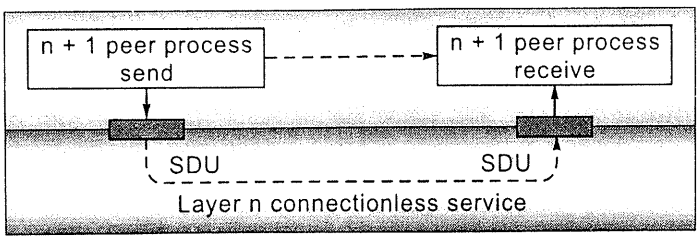


Fig. 10.1(b) : Connectionless transfer service.

(2) Features of (

(a) Initial S

(i) CON

- Set
- Set inf
- *Cor* res

(ii) CLN

- No
- SDU
- SDU alw

(b) Type of T

(i) CON

(ii) CLNS

(c) Addressi

(i) CONS

(ii) CLNS route

(d) Reliabilit

(i) CONS

(ii) CLNS

(e) Error Cor

(i) CONS

(ii) CLNS

(f) Applicati

(i) CONS reliab

(ii) CLNS

)r this channel
(5 M)

6 Mbps
rformance we
rmula to find

:hen waits for
; leads to low

vaiting for an

vithout filling

(2) Features of CONS and CLNS

(a) Initial Setup

(i) **CONS**

- Setup is required.
- Setup initializes the *one level lower* states and establishes a **pipe** for information transfer as shown in the figure above.
- *Connection Release* must be done to remove state information and release resources.

(ii) **CLNS**

- No connection setup required.
- SDUs may follow different paths to reach their destination.
- SDUs may also arrive out-of-order at their destination. (In CONS, SDUs always arrive in order).

(b) Type of Transfer

(i) CONS : Stream of information is transferred.

(ii) CLNS : Blocks of information are transferred.

(c) Addressing

(i) CONS : Destination address needed only during setup.

(ii) CLNS : Destination address needed on every block as every block is routed independently.

(d) Reliability

(i) CONS : More.

(ii) CLNS : Less.

(e) Error Control

(i) CONS : Done at Network Layer.

(ii) CLNS : Done at Transport Layer.

(f) Applications

(i) CONS : Audio and video (as arrival of data is in order and connection is reliable.)

(ii) CLNS : Text.

End to End Versus Hop by Hop

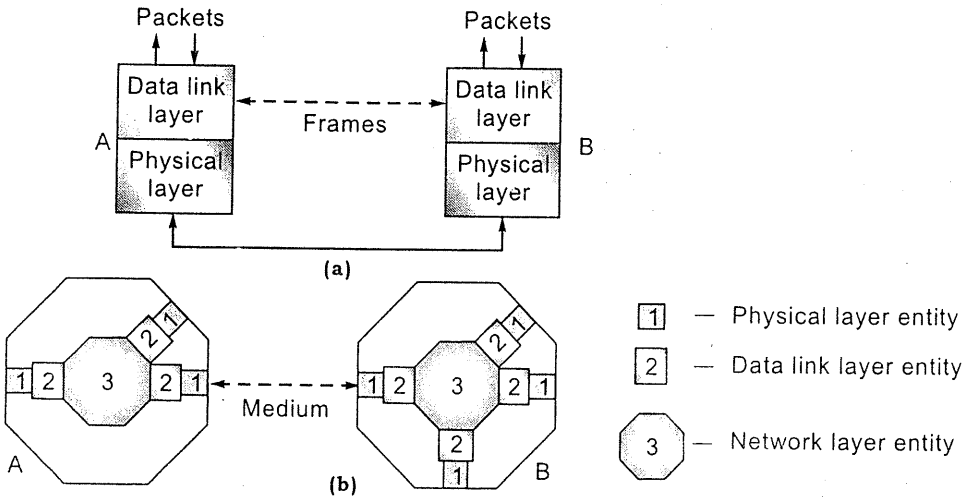


Fig. 10.2 : Peer-to-peer protocol across a single hop.

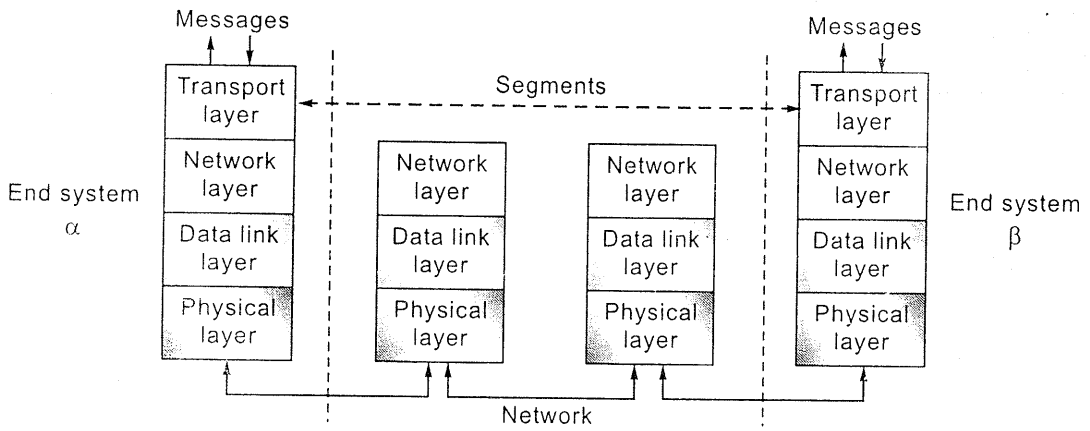


Fig. 10.3 : Peer-to-peer protocol operating end to end across a network protocol stack view.

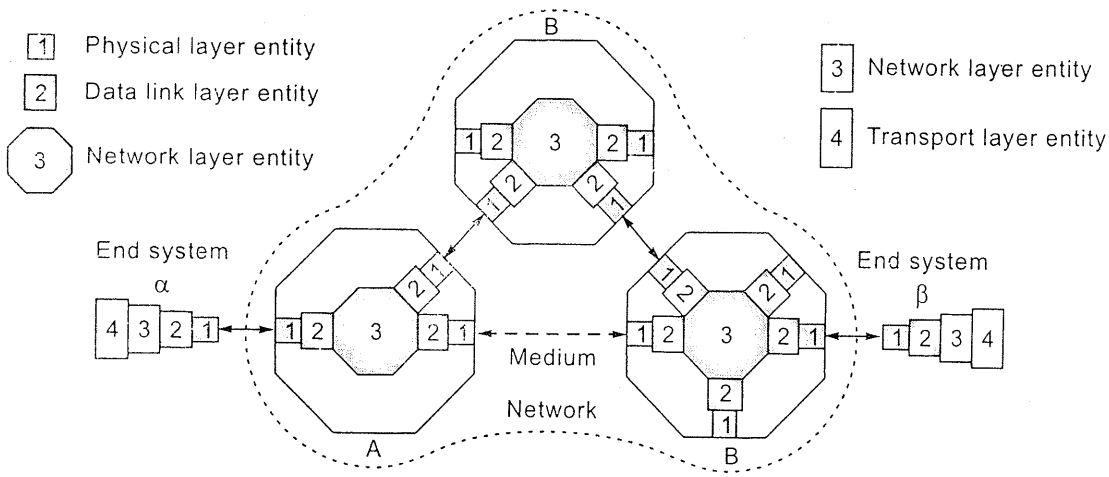


Fig. 10.4 : Peer-to-peer protocol operating end to end across a network-spatial view.

Peer to peer pro
(end-to-end).

(a) **Hop by Ho**
(1) Informa
(2) The Da
them a
informa
(3) Delivery

(b) **End-to-End**
(1) Informa
(2) The Tra
end fash
(3) Packets

(c) **Error Contro**
(1) Conside
10.5.

(2) Once dat
that data
(3) In Hop-b
Hop is m
(4) Incase of
final dest

[VI]

If 2 clients make
each reply ?

- Two child
individual

Peer to peer protocols can occur either across a single hop or across an entire network (end-to-end).

(a) Hop by Hop

(1) Information reaches destination in one hop.

(2) The Data Link Layer is responsible for taking network layer packets, framing them and transmitting them. i.e. the DLL is responsible for transmitting information across a single hop.

(3) Delivery is *in order* as same physical link is used.

(b) End-to-End

(1) Information takes multiple hops to reach destination.

(2) The Transport Layer is responsible for transporting information in an end to end fashion.

(3) Packets may follow different paths and hence *out-of order* delivery is possible.

(c) Error Control

(1) Consider that data has to be transmitted from block 1 to block 4 in the *figure 10.5*.

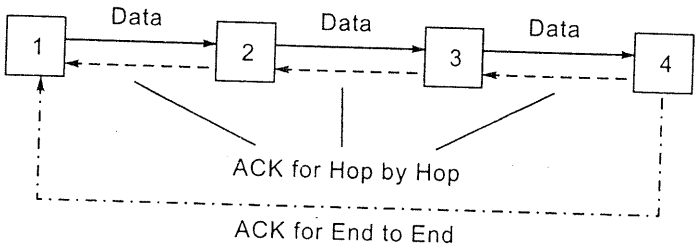


Fig. 10.5

(2) Once data is transmitted an acknowledgment must be sent to sender to say that data has been successfully received.

(3) In Hop-by-Hop acknowledgment must take place at each hop. Thus Hop-by-Hop is more reliable but is more complex.

(4) Incase of End to End an acknowledgment is sent only when data reaches its final destination (block 4). Thus this is less reliable but simpler.

[VI]

If 2 clients make a request on the same port, how does server decide where to send each reply ?

- Two child processes called threads are started on the server to satisfy the two individual requests.

- Each of these child processes on the server have a different socket number. (Each socket has a socket number consisting of IP address and Port)
- A connection is established between the server child process socket and the client socket.
- The reply to each request is routed to the respective client using the IP Number and Port of the client.

[VII]

Which layers of the OSI model perform Error Control ? How do they implement Error Control ?

(a) **Data Link Layer**

To achieve Error control the following techniques are used

(1) *Acknowledgements* : When the receiver correctly receives the data it sends an acknowledgement to the sender. (Used in the Stop and Wait protocol)

(2) *Timer* : The sender maintains a timer which is set to a time which is enough for the data to reach the receiver and for the acknowledgement from the receiver to reach back to the sender.

(3) *Sequence Numbers* : Sequence Numbers are used by the receivers to decide if they are receiving new frames or duplicate frames.(Used in Simplex Protocol for a Noisy Channel)

(4) *CRC* is used.

(5) *Hamming Code* is also used.

(b) **Network Layer**

IPv4 has a checksum field for error control. IPv6 has no such field because the same error control function is offered by TCP and UDP.

(c) **Transport Layer**

It uses a checksum mentioned in section 6.4.3.

[VIII]

What types of addresses are used at the Data Link Layer, Network Layer and Transport Layer?

- **Data Link Layer** : MAC addresses.
- **Network Layer** : IP addresses.
- **Transport Layer** : Port Addresses.

[IX]

May 06 [Q. 1(a)
Transport Layers.

- **Physical Lay**
  but timers ar
- **Data Link L**
  and Sliding W
- **Network Lay**
  may live befo
  networks for
  before reachi
- **Transport La**
  Handshake.

[X] Windows

- Windows 200
  operating sy
  symmetric mu
- It is part of
  released on F
- It was succee
  April 2003.
- Windows 200
- Windows 20
  Advanced Ser
- Additionally,
  and Windows
  2001 and runs
- Windows 200
  standard syst
- There was su
  languages.
- It supports th
  as well as bas
- The Windows
  ◇ Active Dir

et number.
ort)
ocket and the
using the IP
ey implement
a it sends an
ol)
is enough for
the receiver to
decide if they
Protocol for a
cause the same
ork Layer and

[IX]

May 06 [Q. 1(a)] *State the use of timers in the Physical, Data Link, Network and Transport Layers.* **(5 M)**

- **Physical Layer :** The use of timers in the physical layer is not distinctly specified but timers are used in Satellite Communication and in Wireless Media (802.11).
- **Data Link Layer :** Timers are used in the Simplex protocol for Noisy Channel and Sliding Window protocol.
- **Network Layer :** *Packet Lifetime Management :* It deals with how long a packet may live before being discarded. If it is too long, lost packets may clog up the networks for a long time, but if it is too short, packets may sometimes time out before reaching their destination.
- **Transport Layer :** Timers are used in TCP Timer Management and in the 3 Way Handshake.

[X] Windows 2000

- Windows 2000 is a preemptible, interruptible, graphical and business-oriented operating system that was designed to work with either uniprocessor or symmetric multi-processor 32-bit Intel x86 computers.
- It is part of the Microsoft Windows NT line of operating systems and was released on February 17, 2000.
- It was succeeded by Windows XP in October 2001 and Windows Server 2003 in April 2003.
- Windows 2000 is classified as a hybrid kernel operating system.
- Windows 2000 was made available in four editions: Professional, Server, Advanced Server, and Datacenter Server.
- Additionally, Microsoft offered Windows 2000 Advanced Server Limited Edition and Windows 2000 Datacenter Server Limited Edition, which were released in 2001 and runs on 64-bit Intel Itanium microprocessors.
- Windows 2000 has functionality including Microsoft Management Console and standard system administration applications.
- There was support for people with disabilities and also support for different languages.
- It supports the Windows NT file system, NTFS 3.0, the Encrypting File System, as well as basic and dynamic disk storage.
- The Windows 2000 Server family has additional functionality such as
  - ◇ Active Directory services

- ◇ Distributed File System
- ◇ Fault-redundant storage volumes.

- Windows 2000 has become the target of a number of high-profile virus attacks such as Code Red and Nimda, and more than seven years after its release, continues to receive patches for security vulnerabilities on a near-monthly basis.
- Windows 2000 consists of two main layers(modes): a user mode and a kernel mode.
- The User mode refers to the mode in which user programs are run. Such programs are limited in terms of what system resources they have access to.
- The Kernel mode has unrestricted access to the system memory and external devices.
- All user mode applications access system resources via the kernel mode.
- The Encrypting File System (EFS) introduced *strong file system-level encryption* to Windows. It allows any folder or drive on an NTFS volume to be encrypted transparently by the end user
- Windows 2000 supports DirectX 9.0c. The majority of games written for recent versions of DirectX could therefore run on Windows 2000.
- Windows 2000 has now been superseded by newer Microsoft operating systems.

Comparison Between Linux and Windows

Name	Linux	Windows
Creator	Linus Torvalds	Microsoft
First Public Release	1992	November 20, 1985
Predecessor	Unix, Minix	MS-DOS
Latest Version	Kernel 2.6.21.5	Windows Vista (NT 6.0)
Cost	Free	XP Home OEM $79, Vista Home Retail $199, Business $299, Ultimate $399

[XI] TELNET, FTP and Directory Services

Dec. 06 [Q. 5(b)] *Explain the following Internet services : (ii) FTP (iii) TELNET (iv) Directory services.* (7.5 M)

TELNET

- TELNET is a client-server protocol, based on a reliable connection-oriented transport
- TELNET (
  area netwo
- It was dev
- TELNET h
  available f
- Most netw
  TELNET se
  to dominat
- Example, a
  doing so, h
  of his serv
  account in
  computer

FTP

- FTP - File
  another ov
- FTP is a p
  TCP/IP pro
- There are t
- The FTP s
  connection
- The Client
  Once conn
  the server,
- All comput
  connected
  on that net
- Application
  physical se

Directory Ser

- A **directory**
  a computer
- A directory
  organizing
  printers, us
- A director

- TELNET (TELetype NETwork) is a network protocol used on the Internet or local area network (LAN).
- It was developed in 1969
- TELNET has been available on most Unix systems for many years and are available for virtually all platforms.
- Most network equipment and OSs with a TCP/IP stack support some kind of TELNET service server for their remote configuration . Recently, SSH has begun to dominate remote access for Unix-based machines.
- Example, a user might use TELNET from home to check his mail at school. In doing so, he would be using a telnet client to connect from his computer to one of his servers. Once the connection is established, he would then log in with his account information and execute operating system commands remotely on that computer

FTP

- FTP - File Transfer Protocol - is used to transfer data from one computer to another over the Internet, or through a network.
- FTP is a protocol used for exchanging files over a network that supports the TCP/IP protocol.
- There are two computers involved in an FTP transfer: a server and a client.
- The FTP Server, running FTP server software, listens on the network for connection requests from other computers.
- The Client computer, running FTP client software, tries to connect to the server. Once connected, the client can upload files to the server, download files from the server, rename or delete files on the server and so on.
- All computer platforms support the FTP protocol. This allows any computer connected to a TCP/IP based network to manipulate files on another computer on that network regardless of which operating systems are involved.
- Applications for FTP: game servers, voice servers, internet hosts, and other physical servers.

Directory Services

- A **directory service** is defined as a software that stores information and manages a computer network's users and the network resources.
- A directory service is a used for locating, managing, administrating, and organizing common items and network resources (volumes, folders, files, printers, users, groups, devices, telephone numbers and other objects).
- A directory service is not the information repository (database where

information about the users and the resources are stored) .The service uses information stored in the repository and manages the repository.

- Example:
  - ◊ In the case of the X.500 distributed directory services model, the directory service is called a *"naming service"* and it maps the names of network resources to their respective network addresses.
  - ◊ In such a system a user does not have to remember the physical address of the network resource, he just has to provide the name and the service will locate the resource.
- **Replication and Distribution** are important concepts related to a directory service.
  - ◊ **Replication** : It indicates that the same namespace (the same object) is copied to another directory server. This will increase the redundancy and throughput.
  - ◊ **Distribution** : It indicates that multiple directory servers, that hold different name spaces, are interconnected to form a distributed directory service.

[XII] Bandwidth, Latency and Virtual Protocol

Dec. 06 [Q. 6(b)] *Explain the following terms : Bandwidth, Latency, Virtual protocol.* **(6 M)**

[a] Bandwidth

- **Bandwidth** is the difference between the upper and lower cutoff frequencies for a communication channel
- It is typically measured in hertz.
- Bandwidth in Hertz is a central concept in electronics, information theory, radio communications, signal processing etc.
- **Digital Bandwidth** refers to data rate measured in bit/s. Example channel capacity (digital bandwidth capacity) or throughput (digital bandwidth consumption).

[b] Latency

- Latency is the time taken for a message to get from its source to its destination.
- We will cover latencies in Packet Switched networks and in Satellite Transmission.

Packet-switched Networks

- The latency in a Packet switched network is either : *one-way* (the time from the source sending a packet to the destination receiving it), or *round-trip* (the one-

way later
destinatio

- **PING** : It
- One-way
of packet
- The time
measured

Satellite Tra

- Although
a noticeabl
- The time
humans. T
satellite.

[c] Virtual R

- This is a nc
- It is design
on the sam
- This is achi
routers are
the actual r
the data fai
- The physica
Physical ro
something

[XIII] Code f

Dec. 06 [Q. 1(
encrypted data ir

The following is
During encrypti
back by 1 charac

```
// Program
   ahead by
#include<iost
#include<ccni

void main()
```

service uses

:he directory of network

al address of ? service will

a directory

ect) is copied ndancy and

ıold different ;ervice.

protocol.

(6 M)

equencies for

theory, radio

nple channel l bandwidth

destination.

in Satellite

time from the *trip* (the one-

way latency from source to destination plus the one-way latency from the destination back to the source).

- **PING** : It can be used to measure round-trip latency.
- One-way latency for a link can be more strictly defined as the time from the *start* of packet *transmission* to the *start* of packet *reception*.
- The time from the *start* of packet *reception* to the *end* of packet *reception* is measured separately and called "*transmission delay*".

Satellite Transmission

- Although satellite communication takes place at the speed of light, nevertheless a noticeable latency is developed over long distances.
- The time difference may not be much more than a second, but is noticeable by humans. This time lag is due to the massive distances between Earth and the satellite.

[c] Virtual Router Redundancy Protocol (VRRP)

- This is a non-proprietary redundancy protocol described in RFC 3768.
- It is designed to increase the availability of the default gateway servicing hosts on the same subnet.
- This is achieved by advertising a "virtual router" in which two or more physical routers are then configured to stand for the virtual router, with only one doing the actual routing at any given time. If the current physical router that is routing the data fails, the other physical router automatically replaces it.
- The physical router that is currently forwarding data is called the *master router*. Physical routers standing by to take over from the master router in case something goes wrong are called *backup routers*.

[XIII] Code for Encryption and Decryption

Dec. 06 [Q. 1(c)] *Write a pseudo-code to encrypt given text data and decrypt the encrypted data into original text data..* **(10 M)**

The following is the code for encryption and decryption using a substitution cipher. During encryption we move ahead by 1 character and during decryption we move back by 1 character.

```
// Program to Implement a Substitution Cipher which Moves text
   ahead by 1 Character
#include<iostream.h>
#include<ccnio.h>

void main()
```

```
{
clrscr();
int n;
char PT[10],CT[10],MPT[10];
//PT=Plain Text,CT=Ciper Text,MPT=Plain Text Obtained on Decryption

cout<<"Enter number of characters=";
cin>>n;

cout<<"\nENTER TEXT TO BE ENCRYPTED=";
for(int i=0;i<n;i++)
{
cin>>PT[i];
}

/* ENCRYPTION USING SUBSTITUTION CIPHER */
for(i=0;i<n;i++)
{
CT[i]=PT[i]+1;
}
cout<<"\nCIPHER TEXT=";
for(i=0;i<n;i++)
{
cout<<CT[i];
}

/* DECRYPTION OF SUBSTITUTION CIPHER */
for(i=0;i<n;i++)
{
MPT[i]=CT[i]-1;
}
cout<<"\nPLAIN TEXT=";
for(i=0;i<n;i++)
{
cout<<MPT[i];
}

getch();
}
```

OUTPUT

```
Enter number of characters=5
ENTER TEXT TO BE ENCRYPTED=ABCDE
CIPHER TEXT=BCDEF
PLAIN TEXT=ABCDE
```

The pseudocod

(1) Start

(2) Initialise 'n
plain text,
cipher text

(3) Accept "n"

(4) Accept plai

(5) ENCRYPTI
resultant st

(6) Display "C

(7) DECRYPTI
resultant st

(8) Display"MI

(9) Stop

cryption

The pseudocode can be given as follows:

(1) Start

(2) Initialise 'n' as integer to hold number of characters, PT as string to hold the plain text, CT as string to hold cipher text and MPT as string to hold decrypted cipher text.

(3) Accept "n" from the user.

(4) Accept plaintext form the user and store it in PT.

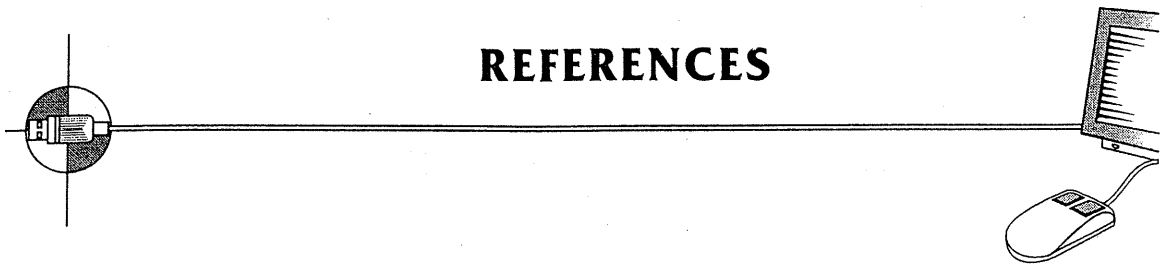
(5) ENCRYPTION: Perform Encryption of PT by moving character ahead by 1. Store resultant string in "CT"

(6) Display "CT"

(7) DECRYPTION: Perform Decryption of CT by moving character back by 1. Store resultant string in "MPT"

(8) Display"MPT"

(9) Stop



REFERENCES

- Andrew S. Tanenbaum - *Computer Networks*
- Forouzan - *Data Communications and Networking*