Specific Instructional Objective

On Completion of this lesson, the students will be able to:

- Define Computer Networks
- State the evolution of Computer Networks
- Categorize different types of Computer Networks
- Specify some of the application of Computer Networks

1.1.1 Introduction

The concept of Network is not new. In simple terms it means an interconnected set of some objects. For decades we are familiar with the Radio, Television, railway, Highway, Bank and other types of networks. In recent years, the network that is making significant impact in our day-to-day life is the **Computer network**. By computer network we mean an interconnected set of autonomous computers. The term autonomous implies that the computers can function independent of others. However, these computers can exchange information with each other through the communication network system. Computer networks have emerged as a result of the convergence of two technologies of this century- Computer and Communication as shown in Fig. 1.1.1. The consequence of this revolutionary merger is the emergence of a integrated system that transmit all types of data and information. There is no fundamental difference between data communications and data processing and there are no fundamental differences among data, voice and video communications. After a brief historical background in Section 1.1.2, Section 1.1.2 introduces different network categories. A brief overview of the applications of computer networks is presented in Section 1.1.3. Finally an outline of the entire course is given in Section 1.1.4.

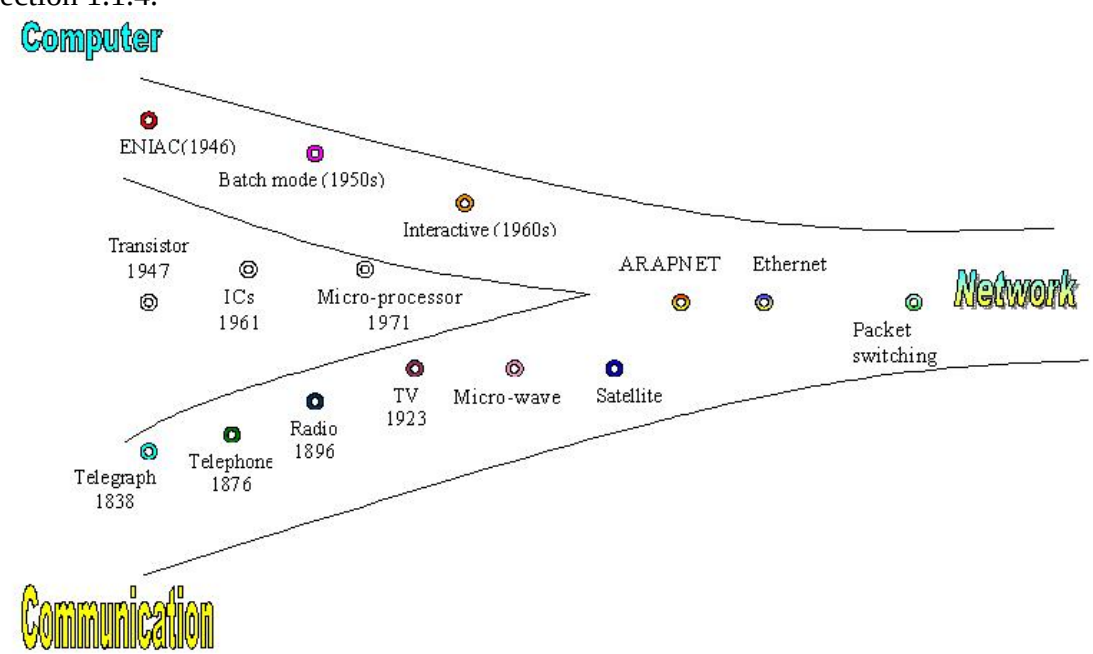


Figure 1.1.1 *Evolution of computer networks*

1.1.2 Historical Background

The history of electronic computers is not very old. It came into existence in the early 1950s and during the first two decades of its existence it remained as a centralized system housed in a single large room. In those days the computers were large in size and were operated by trained personnel. To the users it was a remote and mysterious object having no direct communication with the users. Jobs were submitted in the form of punched cards or paper tape and outputs were collected in the form of computer printouts. The submitted jobs were executed by the computer one after the other, which is referred to as batch mode of data processing. In this scenario, there was long delay between the submission of jobs and receipt of the results.

In the 1960s, computer systems were still centralize, but users provided with direct access through interactive terminals connected by point-to-point low-speed data links with the computer. In this situation, a large number of users, some of them located in remote locations could simultaneously access the centralized computer in time-division multiplexed mode. The users could now get immediate interactive feedback from the computer and correct errors immediately. Following the introduction of on-line terminals and time-sharing operating systems, remote terminals were used to use the central computer.

With the advancement of VLSI technology, and particularly, after the invention of microprocessors in the early 1970s, the computers became smaller in size and less expensive, but with significant increase in processing power. New breed of low-cost computers known as mini and personal computers were introduced. Instead of having a single central computer, an organization could now afford to own a number of computers located in different departments and sections.

Side-by-side, riding on the same VLSI technology the communication technology also advanced leading to the worldwide deployment of telephone network, developed primarily for voice communication. An organization having computers located geographically dispersed locations wanted to have data communications for diverse applications. Communication was required among the machines of the same kind for collaboration, for the use of common software or data or for sharing of some costly resources. This led to the development of computer networks by successful integration and cross-fertilization of communications and geographically dispersed computing facilities. One significant development was the APPANET (Advanced Research Projects Agency Network). Starting with four-node experimental network in 1969, it has subsequently grown into a network several thousand computers spanning half of the globe, from Hawaii to Sweden. Most of the present-day concepts such as packet switching evolved from the ARPANET project. The low bandwidth (3KHz on a voice grade line) telephone network was the only generally available communication system available for this type of network.

The bandwidth was clearly a problem, and in the late 1970s and early 80s another new communication technique known as Local Area Networks (LANs) evolved, which helped computers to communicate at high speed over a small geographical area. In the later years use of optical fiber and satellite communication allowed high-speed data communications over long distances.

1.1.3 Network Technologies

There is no generally accepted taxonomy into which all computer networks fit, but two dimensions stand out as important: **Transmission Technology** and **Scale**. The classifications based on these two basic approaches are considered in this section.

1.1.3.1 Classification Based on Transmission Technology

Computer networks can be broadly categorized into two types based on transmission technologies:

- Broadcast networks
- Point-to-point networks

1.2.3.1.1 Broadcast Networks

Broadcast network have a single communication channel that is shared by all the machines on the network as shown in Figs.1.1.2 and 1.1.3. All the machines on the network receive short messages, called packets in certain contexts, sent by any machine. An address field within the packet specifies the intended recipient. Upon receiving a packet, machine checks the address field. If packet is intended for itself, it processes the packet; if packet is not intended for itself it is simply ignored.

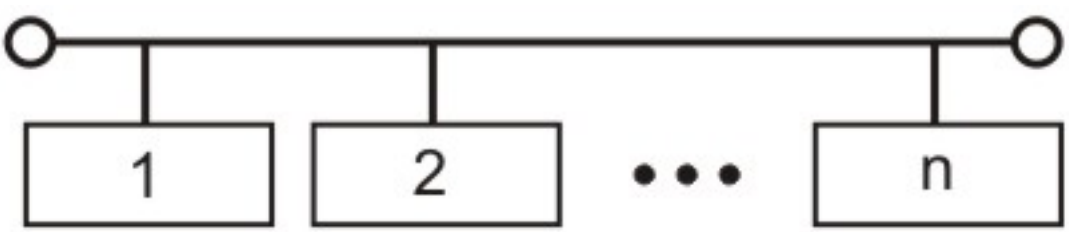


Figure 1.1.2 *Example of a broadcast network based on shared bus*

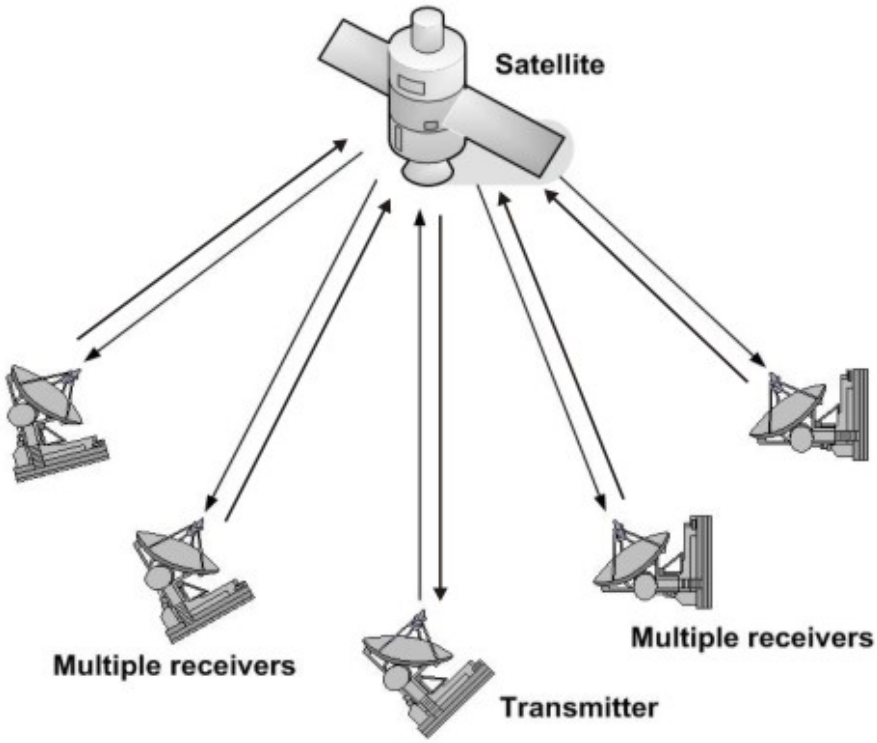


Figure 1.1.3 *Example of a broadcast network based on satellite communication*

This system generally also allows possibility of addressing the packet to all destinations (all nodes on the network). When such a packet is transmitted and received by all the machines on the network. This mode of operation is known as *Broadcast Mode*. Some Broadcast systems also supports transmission to a sub-set of machines, something known as *Multicasting*.

1.2.3.1.2 Point-to-Point Networks

A network based on point-to-point communication is shown in Fig. 1.1.4. The end devices that wish to communicate are called *stations*. The switching devices are called *nodes*. Some Nodes connect to other nodes and some to attached stations. It uses FDM or TDM for node-to-node communication. There may exist multiple paths between a source-destination pair for better network reliability. The switching nodes are not concerned with the contents of data. Their purpose is to provide a switching facility that will move data from node to node until they reach the destination.

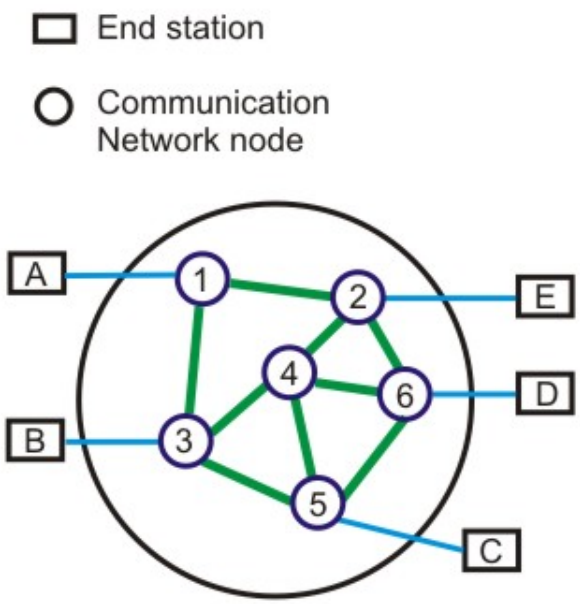


Figure 1.1.4 *Communication network based on point-to-point communication*

As a general rule (although there are many exceptions), smaller, geographically localized networks tend to use broadcasting, whereas larger networks normally use are point-to-point communication.

1.1.3.2 Classification based on Scale

Alternative criteria for classifying networks are their scale. They are divided into Local Area (LAN), Metropolitan Area Network (MAN) and Wide Area Networks (WAN).

1.1.3.2.1 Local Area Network (LAN)

LAN is usually privately owned and links the devices in a single office, building or campus of up to few kilometers in size. These are used to share resources (may be hardware or software resources) and to exchange information. LANs are distinguished from other kinds of networks by three categories: their size, transmission technology and topology.

LANs are restricted in size, which means that their worst-case transmission time is bounded and known in advance. Hence this is more reliable as compared to MAN and WAN. Knowing this bound makes it possible to use certain kinds of design that would not otherwise be possible. It also simplifies network management.

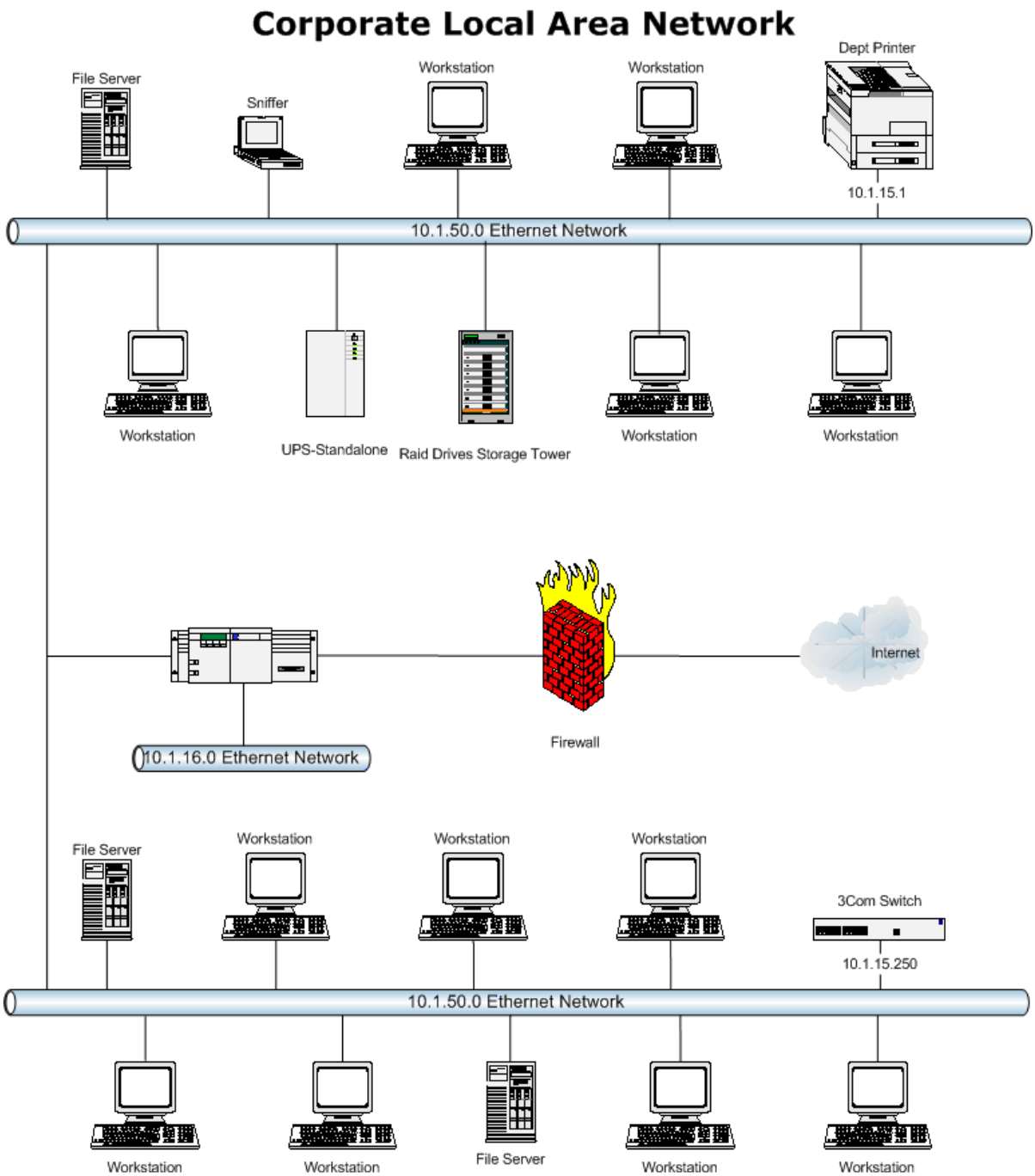


Figure 1.1.5 Local Area Network

LAN typically used transmission technology consisting of single cable to which all machines are connected. Traditional LANs run at speeds of 10 to 100 Mbps (but now much higher speeds can be achieved). The most common LAN topologies are bus, ring and star. A typical LAN is shown in Fig. 1.1.5.

1.1.3.2.2 Metropolitan Area Networks (MAN)

MAN is designed to extend over the entire city. It may be a single network as a cable TV network or it may be means of connecting a number of LANs into a larger network so that resources may be shared as shown in Fig. 1.1.6. For example, a company can use a MAN to connect the LANs in all its offices in a city. MAN is wholly owned and operated by a private company or may be a service provided by a public company.

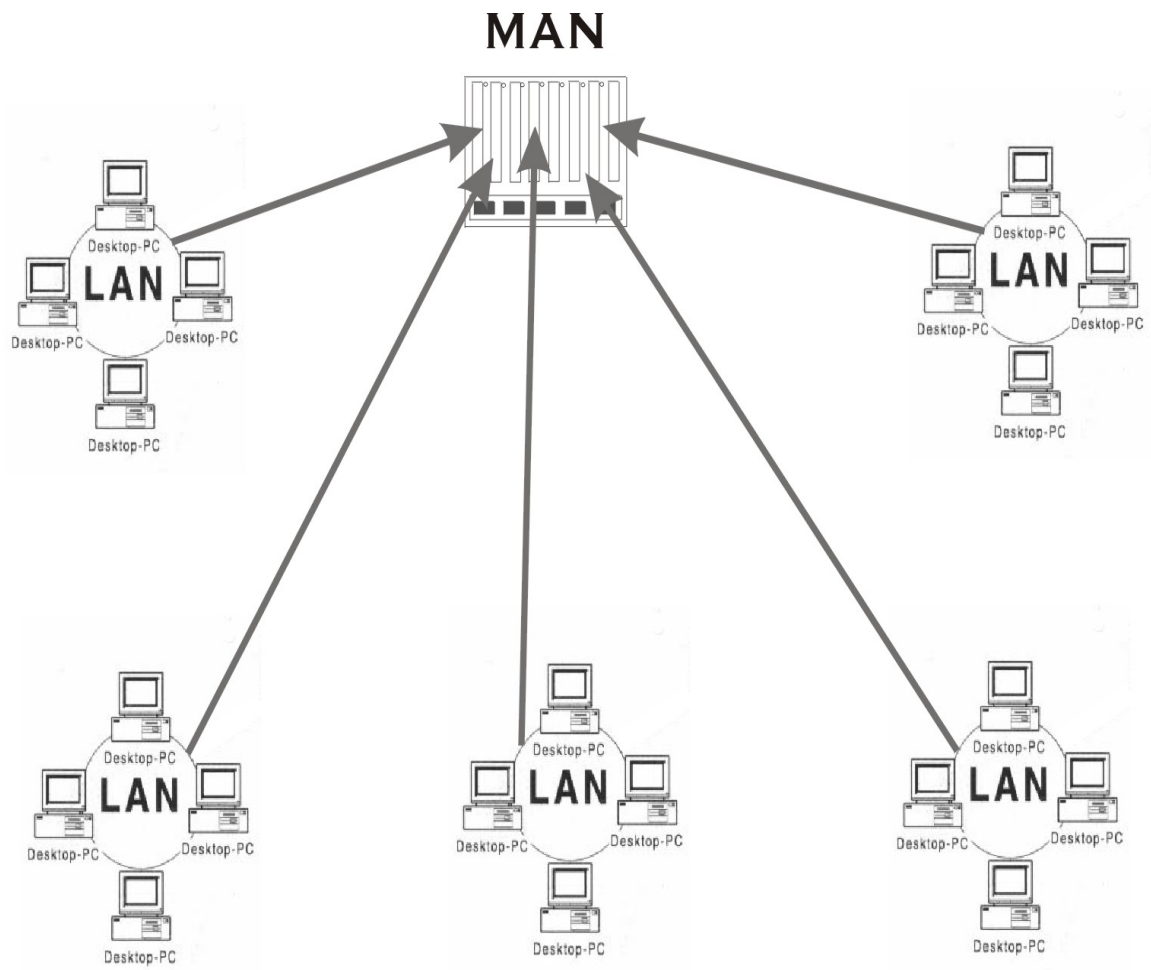


Figure 1.1.6 Metropolitan Area Networks (MAN)

The main reason for distinguishing MANs as a special category is that a standard has been adopted for them. It is **DQDB** (Distributed Queue Dual Bus) or IEEE 802.6.

1.1.3.2.3 Wide Area Network (WAN)

WAN provides long-distance transmission of data, voice, image and information over large geographical areas that may comprise a country, continent or even the whole world. In contrast to LANs, WANs may utilize public, leased or private communication devices, usually in combinations, and can therefore span an unlimited number of miles as shown

in Fig. 1.1.7. A WAN that is wholly owned and used by a single company is often referred to as *enterprise network.*

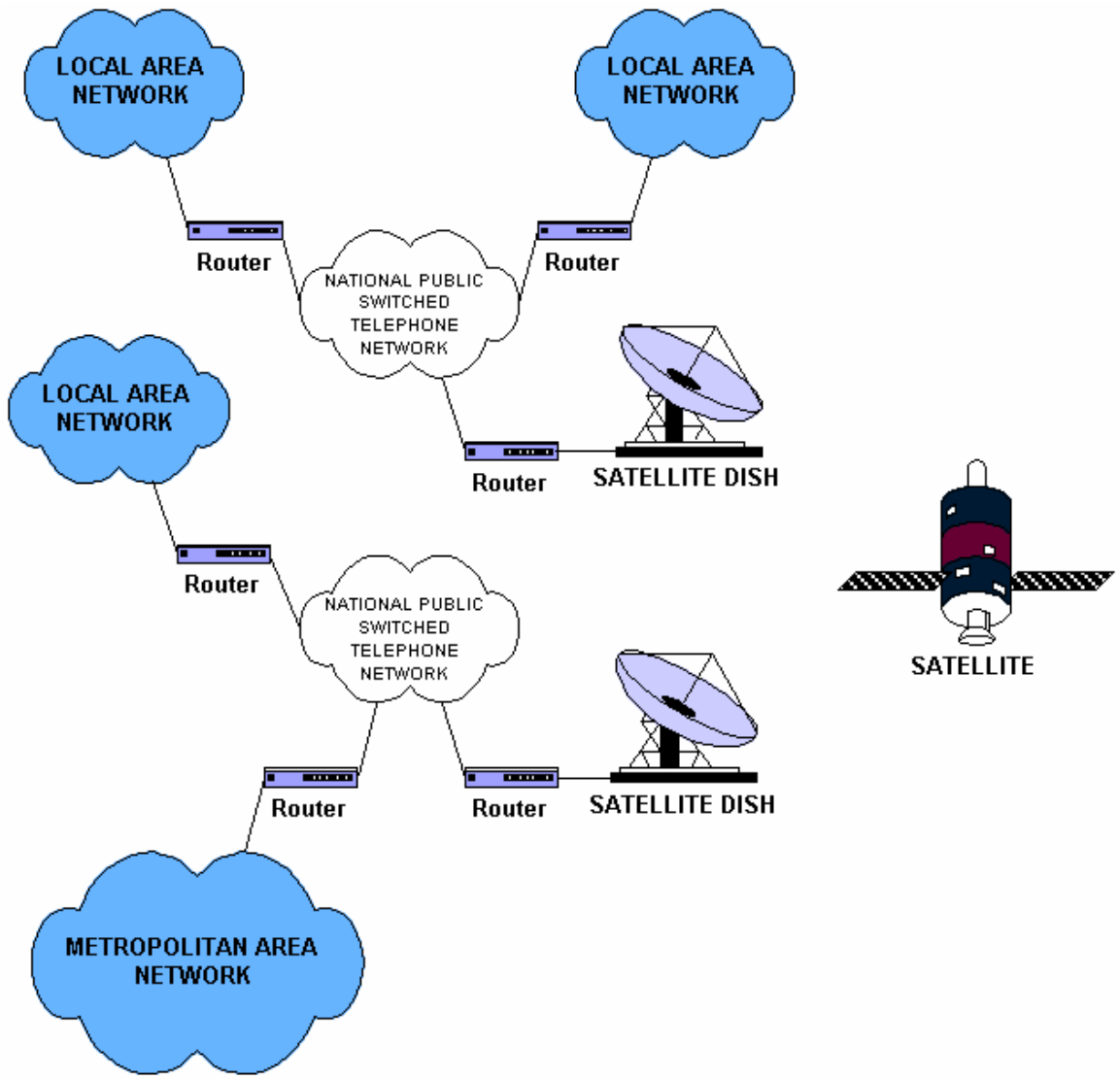


Figure 1.1.7 *Wide Area Network*

1.1.3.2.4 The Internet

Internet is a collection of networks or network of networks. Various networks such as LAN and WAN connected through suitable hardware and software to work in a seamless manner. Schematic diagram of the Internet is shown in Fig. 1.1.8. It allows various applications such as e-mail, file transfer, remote log-in, World Wide Web, Multimedia, etc run across the internet. The basic difference between WAN and Internet is that WAN is owned by a single organization while internet is not so. But with the time the line between WAN and Internet is shrinking, and these terms are sometimes used interchangeably.

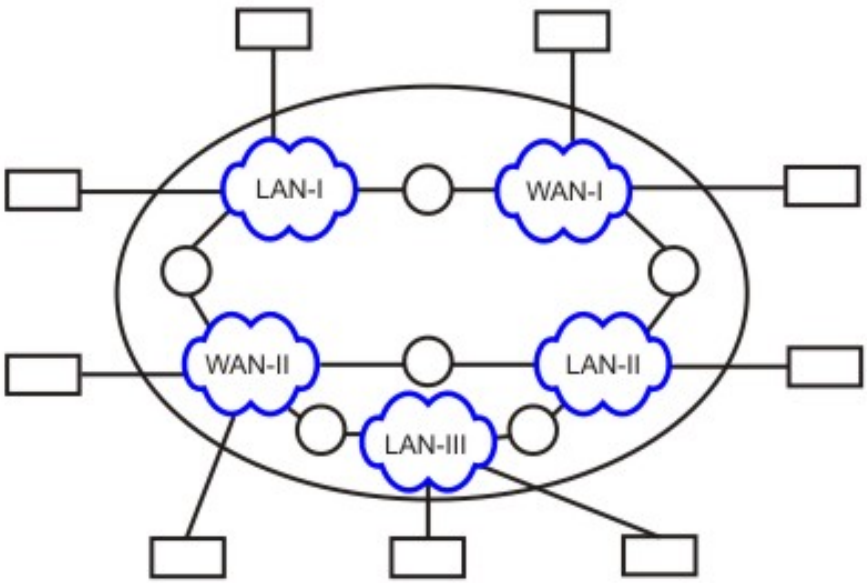


Figure 1.1.8 *Internet – network of networks*

1.1.4 Applications

In a short period of time computer networks have become an indispensable part of business, industry, entertainment as well as a common-man's life. These applications have changed tremendously from time and the motivation for building these networks are all essentially economic and technological.

Initially, computer network was developed for defense purpose, to have a secure communication network that can even withstand a nuclear attack. After a decade or so, companies, in various fields, started using computer networks for keeping track of inventories, monitor productivity, communication between their different branch offices located at different locations. For example, Railways started using computer networks by connecting their nationwide reservation counters to provide the facility of reservation and enquiry from any where across the country.

And now after almost two decades, computer networks have entered a new dimension; they are now an integral part of the society and people. In 1990s, computer network started delivering services to private individuals at home. These services and motivation for using them are quite different. Some of the services are access to remote information, person-person communication, and interactive entertainment. So, some of the applications of computer networks that we can see around us today are as follows:

Marketing and sales: Computer networks are used extensively in both marketing and sales organizations. Marketing professionals use them to collect, exchange, and analyze data related to customer needs and product development cycles. Sales application

includes teleshopping, which uses order-entry computers or telephones connected to order processing network, and online-reservation services for hotels, airlines and so on.

Financial services: Today's financial services are totally depended on computer networks. Application includes credit history searches, foreign exchange and investment services, and electronic fund transfer, which allow user to transfer money without going into a bank (an automated teller machine is an example of electronic fund transfer, automatic pay-check is another).

Manufacturing: Computer networks are used in many aspects of manufacturing including manufacturing process itself. Two of them that use network to provide essential services are computer-aided design (CAD) and computer-assisted manufacturing (CAM), both of which allow multiple users to work on a project simultaneously.

Directory services: Directory services allow list of files to be stored in central location to speed worldwide search operations.

Information services: A Network information service includes bulletin boards and data banks. A World Wide Web site offering technical specification for a new product is an information service.

Electronic data interchange (EDI): EDI allows business information, including documents such as purchase orders and invoices, to be transferred without using paper.

Electronic mail: probably it's the most widely used computer network application.

Teleconferencing: Teleconferencing allows conference to occur without the participants being in the same place. Applications include simple text conferencing (where participants communicate through their normal keyboards and monitor) and video conferencing where participants can even see as well as talk to other fellow participants. Different types of equipments are used for video conferencing depending on what quality of the motion you want to capture (whether you want just to see the face of other fellow participants or do you want to see the exact facial expression).

Voice over IP: Computer networks are also used to provide voice communication. This kind of voice communication is pretty cheap as compared to the normal telephonic conversation.

Video on demand: Future services provided by the cable television networks may include video on request where a person can request for a particular movie or any clip at anytime he wish to see.

Summary: The main area of applications can be broadly classified into following categories:

Specific Functional Objectives

On Completion of this lesson, the students will be able to:

- State the requirement for layered approach
- Explain the basic concept of layering in the network model
- Define entities protocols in networking context
- Describe ISO's OSI Reference Model
- Explain information flow in OSI references Model.
- Explain functions of the seven layers of OSI Model

1.2.1 Basic concept of layering

Network architectures define the standards and techniques for designing and building communication systems for computers and other devices. In the past, vendors developed their own architectures and required that other vendors conform to this architecture if they wanted to develop compatible hardware and software. There are proprietary network architectures such as IBM's SNA (Systems Network Architecture) and there are open architectures like the OSI (Open Systems Interconnection) model defined by the International Organization for Standardization. The previous strategy, where the computer network is designed with the hardware as the main concern and software is afterthought, no longer works. Network software is now highly *structured.*

To reduce the design complexity, most of the networks are organized as a series of **layers** or **levels**, each one build upon one below it. The basic idea of a layered architecture is *to divide the design into small pieces*. Each layer adds to the services provided by the lower layers in such a manner that the highest layer is provided a full set of services to manage communications and run the applications. The benefits of the layered models are modularity and clear interfaces, i.e. open architecture and comparability between the different providers' components.

A basic principle is to ensure independence of layers by defining services provided by each layer to the next higher layer without defining how the services are to be performed. This permits changes in a layer without affecting other layers. Prior to the use of layered protocol architectures, simple changes such as adding one terminal type to the list of those supported by an architecture often required changes to essentially all communications software at a site. The number of layers, functions and contents of each layer differ from network to network. However in all networks, the purpose of each layer is to offer certain services to higher layers, shielding those layers from the details of how the services are actually implemented.

The basic elements of a layered model are services, protocols and interfaces. A *service* is a set of actions that a layer offers to another (higher) layer. *Protocol* is a set of rules that a layer uses to exchange information with a peer entity. These rules concern both the contents and the order of the messages used. Between the layers service interfaces are defined. The messages from one layer to another are sent through those interfaces.

In an n-layer architecture, layer n on one machine carries on conversation with the layer n on other machine. The rules and conventions used in this conversation are collectively known as the *layer-n protocol*. Basically, a protocol is an agreement between the communicating parties on how communication is to proceed. Violating the protocol will make communication more difficult, if not impossible. A five-layer architecture is shown in Fig. 1.2.1**,** the entities comprising the corresponding layers on different machines are called *peers*. In other words, it is the peers that communicate using protocols. In reality, no data is transferred from layer n on one machine to layer n of another machine. Instead, each layer passes data and control information to the layer immediately below it, until the lowest layer is reached. Below layer-1 is the physical layer through which actual communication occurs. The peer process abstraction is crucial to all network design. Using it, the un-manageable tasks of designing the complete network can be broken into several smaller, manageable, design problems, namely design of individual layers.

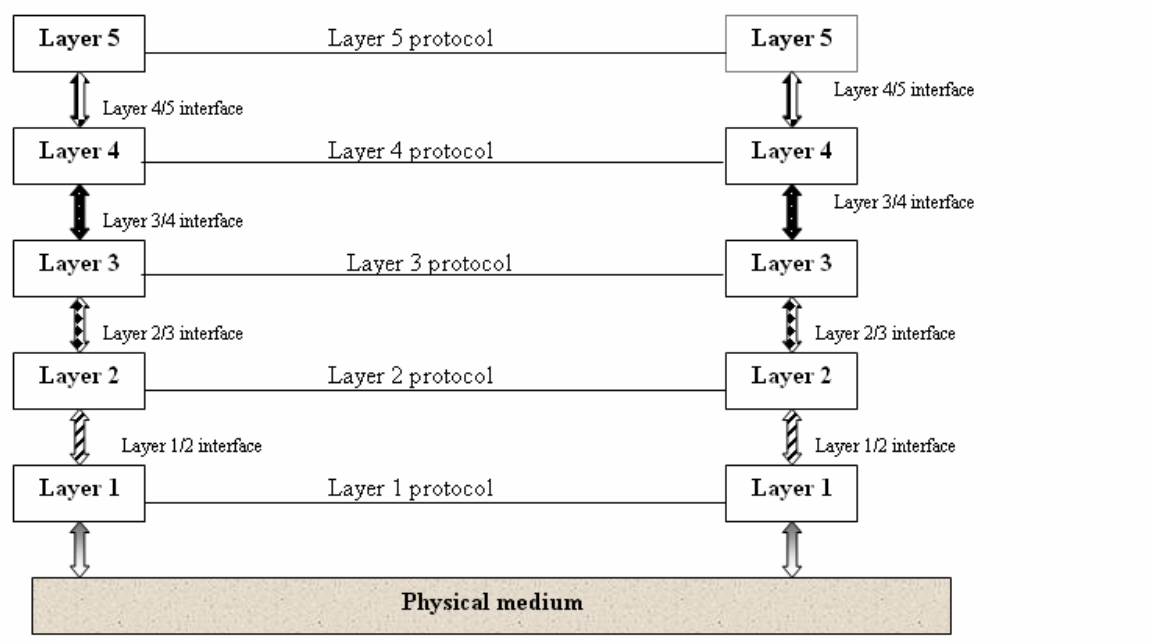


Figure 1.2.1 *Basic five layer architecture*

Between each pair of adjacent layers there is an **interface**. The *interface* defines which primitives operations and services the lower layer offers to the upper layer adjacent to it. When network designer decides how many layers to include in the network and what each layer should do, one of the main considerations is defining clean interfaces between adjacent layers. Doing so, in turns requires that each layer should perform well-defined functions. In addition to minimize the amount of information passed between layers, clean-cut interface also makes it simpler to replace the implementation of one layer with a completely different implementation, because all what is required of new implementation is that it offers same set of services to its upstairs neighbor as the old implementation (that is what a layer provides and how to use that service from it is more important than knowing how exactly it implements it).

A set of layers and protocols is known as **network architecture**. The specification of architecture must contain enough information to allow an implementation to write the program or build the hardware for each layer so that it will correctly obey the appropriate protocol. Neither the details of implementation nor the specification of interface is a part of network architecture because these are hidden away inside machines and not visible from outside. It is not even necessary that the interface on all machines in a network be same, provided that each machine can correctly use all protocols. A list of protocols used by a certain system, one protocol per layer, is called **protocol stack**.

Summary: Why Layered architecture?

1. To make the design process easy by breaking unmanageable tasks into several smaller and manageable tasks (by divide-and-conquer approach).

2. Modularity and clear interfaces, so as to provide comparability between the different providers' components.

3. Ensure independence of layers, so that implementation of each layer can be changed or modified without affecting other layers.

4. Each layer can be analyzed and tested independently of all other layers.

1.2.2 Open System Interconnection Reference Model

The Open System Interconnection (OSI) reference model describes how information from a software application in one computer moves through a network medium to a software application in another computer. The OSI reference model is a conceptual model composed of seven layers, each specifying particular network functions. The model was developed by the International Organization for Standardization (ISO) in 1984, and it is now considered the primary architectural model for inter-computer communications. The OSI model divides the tasks involved with moving information between networked computers into seven smaller, more manageable task groups. A task or group of tasks is then assigned to each of the seven OSI layers. Each layer is reasonably self-contained so that the tasks assigned to each layer can be implemented independently. This enables the solutions offered by one layer to be updated without adversely affecting the other layers.

The OSI Reference Model includes seven layers:

7. *Application Layer*: Provides Applications with access to network services.

6. *Presentation Layer*: Determines the format used to exchange data among networked computers.

5. *Session Layer*: Allows two applications to establish, use and disconnect a connection between them called a session. Provides for name recognition and additional functions like security, which are needed to allow applications to communicate over the network.

4. *Transport Layer*: Ensures that data is delivered error free, in sequence and with no loss, duplications or corruption. This layer also repackages data by assembling long messages into lots of smaller messages for sending, and repackaging the smaller messages into the original larger message at the receiving end.

3. *Network Layer*: This is responsible for addressing messages and data so they are sent to the correct destination, and for translating logical addresses and names (like a machine name FLAME) into physical addresses. This layer is also responsible for finding a path through the network to the destination computer.

2. *Data-Link Layer*: This layer takes the data frames or messages from the Network Layer and provides for their actual transmission. At the receiving computer, this layer receives the incoming data and sends it to the network layer for handling. The Data-Link Layer also provides error-free delivery of data between the two computers by using the physical layer. It does this by packaging the data from the Network Layer into a frame, which includes error detection information. At the receiving computer, the Data-Link Layer reads the incoming frame, and generates its own error detection information based on the received frames data. After receiving the entire frame, it then compares its error detection value with that of the incoming frames, and if they match, the frame has been received correctly.

1. *Physical Layer*: Controls the transmission of the actual data onto the network cable. It defines the electrical signals, line states and encoding of the data and the connector types used. An example is 10BaseT.

1.2.2.1Characteristics of the OSI Layers

The seven layers of the OSI reference model can be divided into two categories: upper layers and lower layers as shown in Fig. 1.2.2.

The upper layers of the OSI model deal with application issues and generally are implemented only in software. The highest layer, the application layer, is closest to the end user. Both users and application layer processes interact with software applications that contain a communications component. The term upper layer is sometimes used to refer to any layer above another layer in the OSI model.

The lower layers of the OSI model handle data transport issues. The physical layer and the data link layer are implemented in hardware and software. The lowest layer, the physical layer, is closest to the physical network medium (the network cabling, for example) and is responsible for actually placing information on the medium .

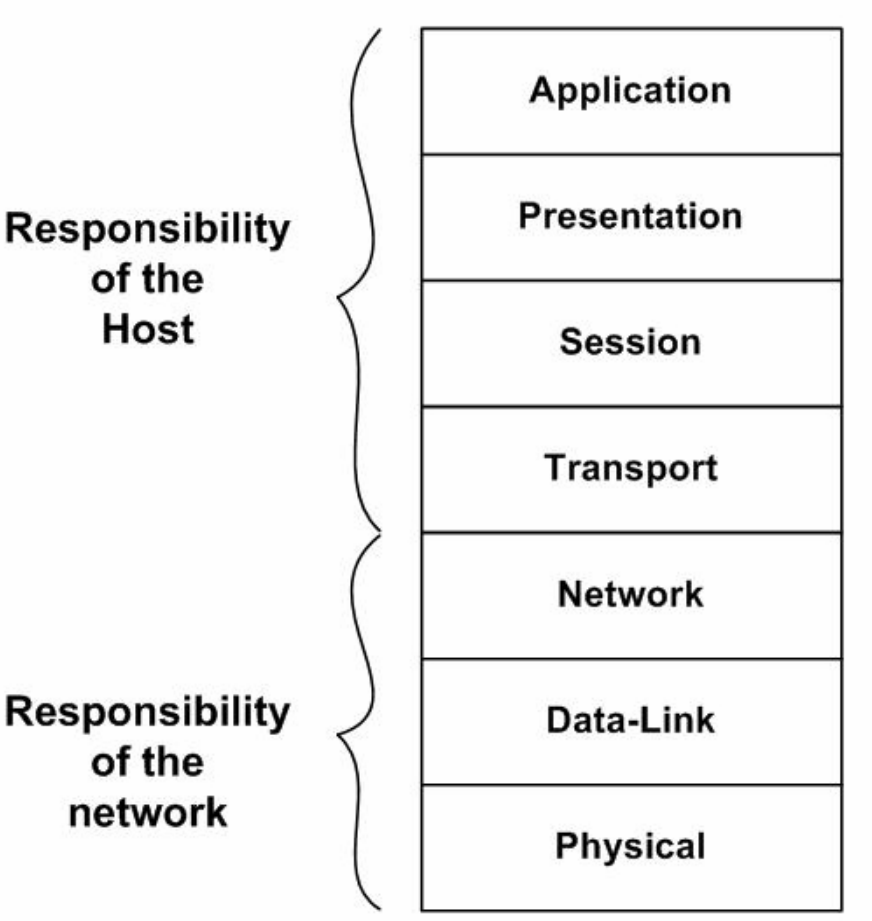


Figure 1.2.2 *Two sets of layers make up the OSI layers*

1.2.2.2 Protocols

The OSI model provides a conceptual framework for communication between computers, but the model itself is not a method of communication. Actual communication is made possible by using communication protocols. In the context of data networking, a **protocol** *is a formal set of rules and conventions that governs how computers exchange information over a network medium*. A protocol implements the functions of one or more of the OSI layers.

A wide variety of communication protocols exist. Some of these protocols include LAN protocols, WAN protocols, network protocols, and routing protocols. LAN protocols operate at the physical and data link layers of the OSI model and define communication over various LAN media. WAN protocols operate at the lowest three layers of the OSI model and define communication over the various wide-area media. Routing protocols are network layer protocols that are responsible for exchanging information between routers so that the routers can select the proper path for network traffic. Finally, network protocols are the various upper-layer protocols that exist in a given protocol suite. Many protocols rely on others for operation. For example, many routing protocols use network

protocols to exchange information between routers. This concept of building upon the layers already in existence is the foundation of the OSI model.

1.2.2.3 OSI Model and Communication between Systems

Information being transferred from a software application in one computer system to a software application in another must pass through the OSI layers. For example, if a software application in System A has information to transmit to a software application in System B, the application program in System A will pass its information to the application layer (Layer 7) of System A. The application layer then passes the information to the presentation layer (Layer 6), which relays the data to the session layer (Layer 5), and so on down to the physical layer (Layer 1). At the physical layer, the information is placed on the physical network medium and is sent across the medium to System B. The physical layer of System B removes the information from the physical medium, and then its physical layer passes the information up to the data link layer (Layer 2), which passes it to the network layer (Layer 3), and so on, until it reaches the application layer (Layer 7) of System B. Finally, the application layer of System B passes the information to the recipient application program to complete the communication process.

1.2.2.4 Interaction between OSI Model Layers

A given layer in the OSI model generally communicates with three other OSI layers: the layer directly above it, the layer directly below it, and its peer layer in other networked computer systems. The data link layer in System A, for example, communicates with the network layer of System A, the physical layer of System A, and the data link layer in System B. Figure1.2.3 illustrates this example.

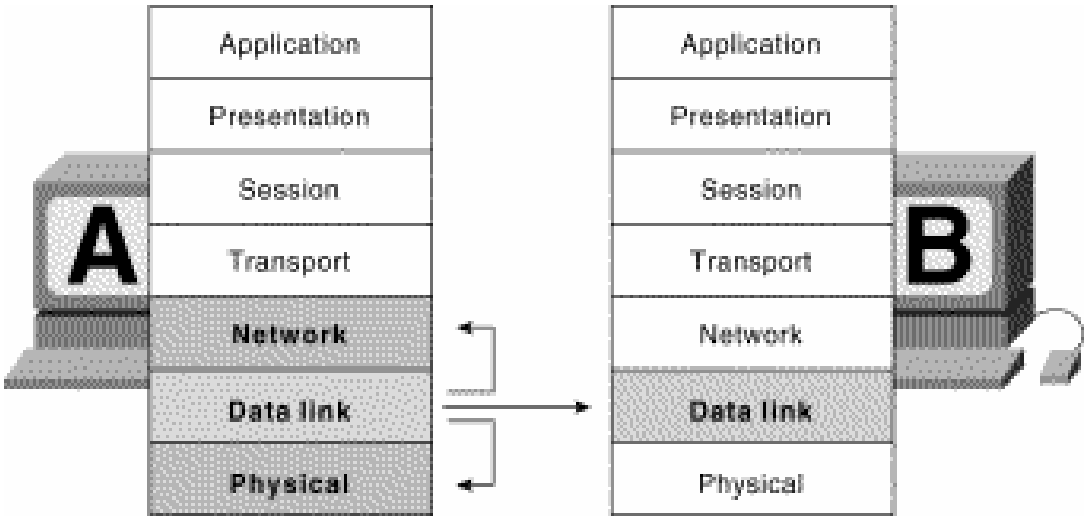


Figure 1.2.3 *OSI Model Layers Communicate with Other Layers*

1.2.3 Services and service access points

One OSI layer communicates with another layer to make use of the services provided by the second layer. The services provided by adjacent layers help a given OSI layer communicate with its peer layer in other computer systems. Three basic elements are involved in layer services: the service user, the service provider, and the service access point (SAP).

In this context, the service user is the OSI layer that requests services from an adjacent OSI layer. The service provider is the OSI layer that provides services to service users. OSI layers can provide services to multiple service users. The SAP is a conceptual location at which one OSI layer can request the services of another OSI layer.

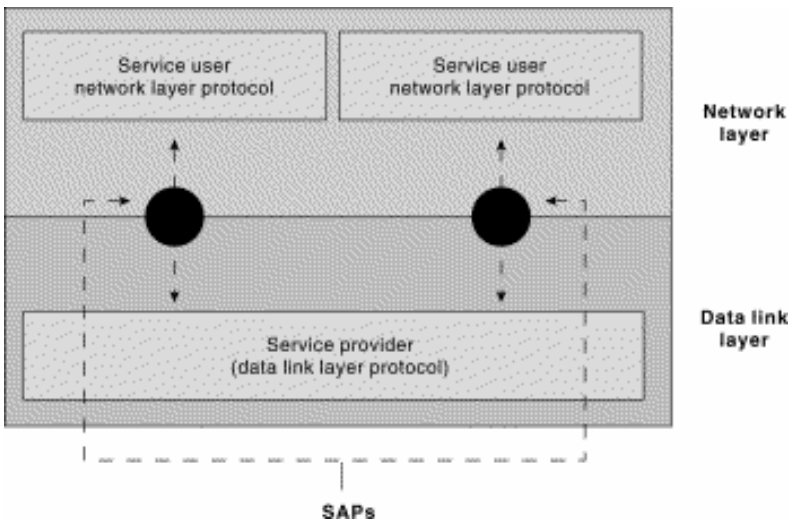


Figure 1.2.4 *Service Users, Providers, and SAPs interact at the Network and Data Link Layers*

1.2.3.1 OSI Model Layers and Information Exchange

The seven OSI layers use various forms of control information to communicate with their peer layers in other computer systems. This control information consists of specific requests and instructions that are exchanged between peer OSI layers.

Control information typically takes one of two forms: headers and trailers. Headers are prepended to data that has been passed down from upper layers. Trailers are appended to data that has been passed down from upper layers. An OSI layer is not required to attach a header or a trailer to data from upper layers.

Headers, trailers, and data are relative concepts, depending on the layer that analyzes the information unit. At the network layer, for example, an information unit consists of a

Layer 3 header and data. At the data link layer, however, all the information passed down by the network layer (the Layer 3 header and the data) is treated as data.

In other words, the data portion of an information unit at a given OSI layer potentially can contain headers, trailers, and data from all the higher layers. This is known as encapsulation. Figure 1-6 shows how the header and data from one layer are encapsulated into the header of the next lowest layer.

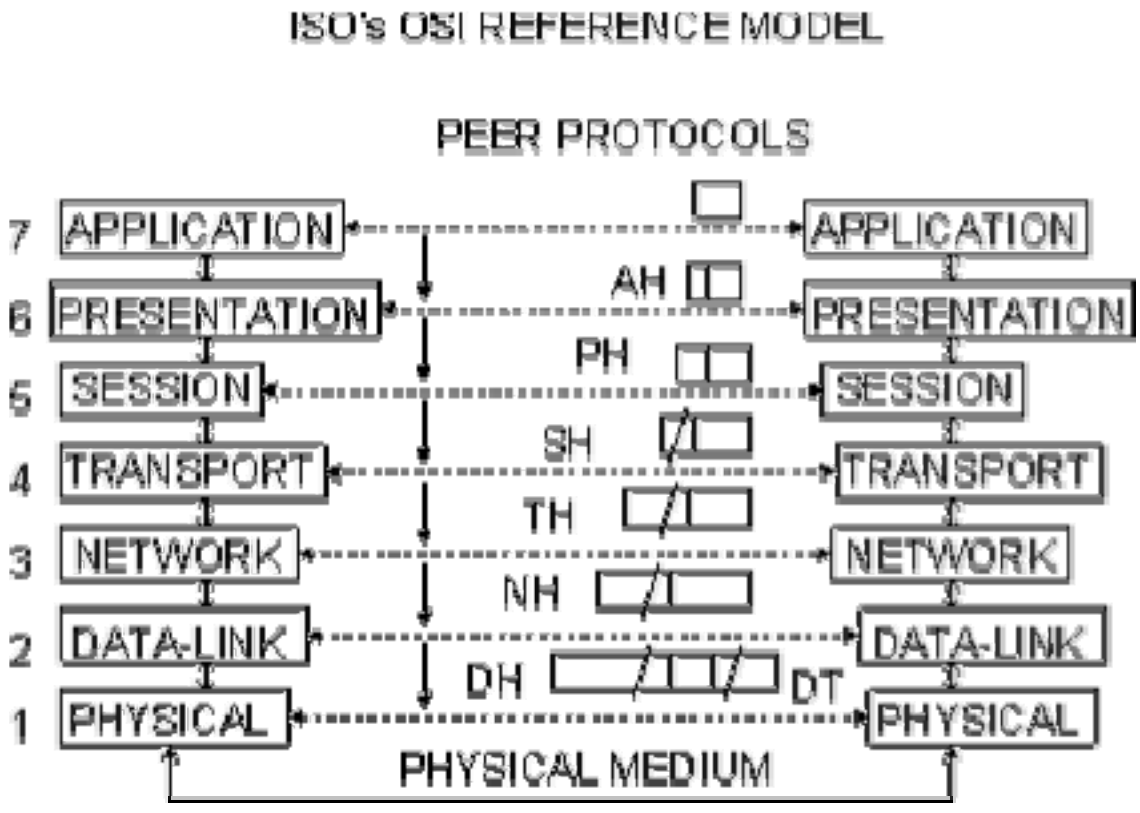


Figure 1.2.6 *Headers and Data can be encapsulated during Information exchange*

1.2.3.2 Information Exchange Process

The information exchange process occurs between peer OSI layers. Each layer in the source system adds control information to data, and each layer in the destination system analyzes and removes the control information from that data.

If system A has data from software application to send to System B, the data is passed to the application layer. The application layer in System A then communicates any control information required by the application layer in System B by pre-pending a header to the data. The resulting information unit (a header and the data) is passed to the presentation layer, which pre-pends its own header containing control information intended for the presentation layer in System B. The information unit grows in size as each layer pre-pends its own header (and, in some cases, a trailer) that contains control information to be

used by its peer layer in System B. At the physical layer, the entire information unit is placed onto the network medium.

The physical layer in System B receives the information unit and passes it to the data link layer. The data link layer in System B then reads the control information contained in the header pre-pended by the data link layer in System A. The header is then removed, and the remainder of the information unit is passed to the network layer. Each layer performs the same actions: The layer reads the header from its peer layer, strips it off, and passes the remaining information unit to the next highest layer. After the application layer performs these actions, the data is passed to the recipient software application in System B, in exactly the form in which it was transmitted by the application in System A.

1.2.4 Functions of the OSI Layers

Functions of different layers of the OSI model are presented in this section.

1.2.4.1 Physical Layer

The physical layer is concerned with transmission of raw bits over a communication channel. It specifies the mechanical, electrical and procedural network interface specifications and the physical transmission of bit streams over a transmission medium connecting two pieces of communication equipment. In simple terns, the physical layer decides the following:

- Number of pins and functions of each pin of the network connector (Mechanical)
- Signal Level, Data rate (Electrical)
- Whether simultaneous transmission in both directions
- Establishing and breaking of connection
- Deals with physical transmission

There exist a variety of physical layer protocols such as RS-232C, Rs-449 standards developed by Electronics Industries Association (EIA).

1.2.4.2 Data Link Layer

The goal of the data link layer is to provide reliable, efficient communication between adjacent machines connected by a single communication channel. Specifically:

1. Group the physical layer bit stream into units called frames. Note that frames are nothing more than ``packets" or ``messages". By convention, we shall use the term ``frames" when discussing DLL packets.

2. Sender calculates the checksum and sends checksum together with data. The checksum allows the receiver to determine when a frame has been damaged in transit or received correctly.

3. Receiver recomputes the checksum and compares it with the received value. If they differ, an error has occurred and the frame is discarded.

4. Error control protocol returns a positive or negative acknowledgment to the sender. A positive acknowledgment indicates the frame was received without errors, while a negative acknowledgment indicates the opposite.

5. Flow control prevents a fast sender from overwhelming a slower receiver. For example, a supercomputer can easily generate data faster than a PC can consume it.

6. In general, data link layer provides service to the network layer. The network layer wants to be able to send packets to its neighbors without worrying about the details of getting it there in one piece.

1.2.4.2.1 Design Issues

Below are the some of the important design issues of the data link layer:

a). Reliable Delivery:

Frames are delivered to the receiver reliably and in the same order as generated by the sender. Connection state keeps track of sending order and which frames require retransmission. For example, receiver state includes which frames have been received, which ones have not, etc.

b). Best Effort:

The receiver does not return acknowledgments to the sender, so the sender has no way of knowing if a frame has been successfully delivered.

When would such a service be appropriate?

1. When higher layers can recover from errors with little loss in performance. That is, when errors are so infrequent that there is little to be gained by the data link layer performing the recovery. It is just as easy to have higher layers deal with occasional loss of packet.

2. For real-time applications requiring ``better never than late" semantics. Old data may be worse than no data.

c). Acknowledged Delivery

The receiver returns an acknowledgment frame to the sender indicating that a data frame was properly received. This sits somewhere between the other two in that the sender keeps connection state, but may not necessarily retransmit unacknowledged frames. Likewise, the receiver may hand over received packets to higher layer in the order in

which they arrive, regardless of the original sending order. Typically, each frame is assigned a unique sequence number, which the receiver returns in an acknowledgment frame to indicate which frame the ACK refers to. The sender must retransmit unacknowledged (e.g., lost or damaged) frames.

d). Framing

The DLL translates the physical layer's raw bit stream into discrete units (messages) called *frames*. How can the receiver detect frame boundaries? Various techniques are used for this: Length Count, Bit Stuffing, and Character stuffing.

e). Error Control

Error control is concerned with insuring that all frames are eventually delivered (possibly in order) to a destination. To achieve this, three items are required: Acknowledgements, Timers, and Sequence Numbers.

f). Flow Control

Flow control deals with throttling the speed of the sender to match that of the receiver. Usually, this is a dynamic process, as the receiving speed depends on such changing factors as the load, and availability of buffer space.

1.2.4.2.2 Link Management

In some cases, the data link layer service must be ``opened'' before use:

- The data link layer uses open operations for allocating buffer space, control blocks, agreeing on the maximum message size, etc.
- Synchronize and initialize send and receive sequence numbers with its peer at the other end of the communications channel.

1.2.4.2.3 Error Detection and Correction

In data communication, error may occur because of various reasons including attenuation, noise. Moreover, error usually occurs as bursts rather than independent, single bit errors. For example, a burst of lightning will affect a set of bits for a short time after the lightning strike. Detecting and correcting errors requires redundancy (i.e., sending additional information along with the data).

There are two types of attacks against errors:

- Error Detecting Codes: Include enough redundancy bits to detect errors and use ACKs and retransmissions to recover from the errors. Example: parity encoding.
- Error Correcting Codes: Include enough redundancy to detect and correct errors. Examples: CRC checksum, MD5.

1.2.4.3 Network Layer

The basic purpose of the network layer is to provide an end-to-end communication capability in contrast to machine-to-machine communication provided by the data link layer. This end-to-end is performed using two basic approaches known as connection-oriented or connectionless network-layer services.

1.2.4.3.1 Four issues:

1. Interface between the host and the network (the network layer is typically the boundary between the host and subnet)

2. Routing

3. Congestion and deadlock

4. Internetworking (A path may traverse different network technologies (e.g., Ethernet, point-to-point links, etc.)

1.2.4.3.2 Network Layer Interface

There are two basic approaches used for sending packets, which is a group of bits that includes data plus source and destination addresses, from node to node called *virtual circuit* and *datagram* methods. These are also referred to as *connection-oriented* and *connectionless* network-layer services. In virtual circuit approach, a *route*, which consists of logical connection, is first established between two users. During this establishment phase, the two users not only agree to set up a connection between them but also decide upon the quality of service to be associated with the connection. The well-known virtual-circuit protocol is the ISO and CCITT *X.25* specification. The datagram is a self-contained message unit, which contains sufficient information for routing from the source node to the destination node without dependence on previous message interchanges between them. In contrast to the virtual-circuit method, where a fixed path is explicitly set up before message transmission, sequentially transmitted messages can follow completely different paths. The datagram method is analogous to the postal system and the virtual-circuit method is analogous to the telephone system.

1.2.4.3.3 Overview of Other Network Layer Issues:

The network layer is responsible for routing packets from the source to destination. The *routing algorithm* is the piece of software that decides where a packet goes next (e.g., which output line, or which node on a broadcast channel).

For connectionless networks, the routing decision is made for each datagram. For connection-oriented networks, the decision is made once, at circuit setup time.

1.2.4.3.4 Routing Issues:

The routing algorithm must deal with the following issues:

- Correctness and simplicity: networks are never taken down; individual parts (e.g., links, routers) may fail, but the whole network should not.
- Stability: if a link or router fails, how much time elapses before the remaining routers recognize the topology change? (Some never do.)
- Fairness and optimality: an inherently intractable problem. Definition of optimality usually doesn't consider fairness. Do we want to maximize channel usage? Minimize average delay?

When we look at routing in detail, we'll consider both adaptive--those that take current traffic and topology into consideration--and non-adaptive algorithms.

1.2.4.3.4 Congestion

The network layer also must deal with congestion:

- When more packets enter an area than can be processed, delays increase and performance decreases. If the situation continues, the subnet may have no alternative but to discard packets.
- If the delay increases, the sender may (incorrectly) retransmit, making a bad situation even worse.
- Overall, performance degrades because the network is using (wasting) resources processing packets that eventually get discarded.

1.2.4.3.5 Internetworking

Finally, when we consider internetworking -- connecting different network technologies together -- one finds the same problems, only worse:

- Packets may travel through many different networks
- Each network may have a different frame format
- Some networks may be connectionless, other connection oriented

1.2.4.3.6 Routing

Routing is concerned with the question: Which line should router J use when forwarding a packet to router K?

There are two types of algorithms:

- **Adaptive algorithms** use such dynamic information as current topology, load, delay, etc. to select routes.
- In **non-adaptive algorithms**, routes never change once initial routes have been selected. Also called static routing.

Obviously, adaptive algorithms are more interesting, as non-adaptive algorithms don't even make an attempt to handle failed links.

1.2.4.4 Transport Layer

The transport level provides end-to-end communication between processes executing on different machines. Although the services provided by a transport protocol are similar to those provided by a data link layer protocol, there are several important differences between the transport and lower layers:

1. User Oriented. Application programmers interact directly with the transport layer, and from the programmers perspective, the transport layer is the ``network''. Thus, the transport layer should be oriented more towards user services than simply reflect what the underlying layers happen to provide. (Similar to the beautification principle in operating systems.)

2. Negotiation of Quality and Type of Services. The user and transport protocol may need to negotiate as to the quality or type of service to be provided. Examples? A user may want to negotiate such options as: throughput, delay, protection, priority, reliability, etc.

3. Guarantee Service. The transport layer may have to overcome service deficiencies of the lower layers (e.g. providing reliable service over an unreliable network layer).

4. Addressing becomes a significant issue. That is, now the user must deal with it; before it was buried in lower levels.

Two solutions:

- Use well-known addresses that rarely if ever change, allowing programs to ``wire in'' addresses. For what types of service does this work? While this works for services that are well established (e.g., mail, or telnet), it doesn't allow a user to easily experiment with new services.

- Use a name server. Servers register services with the name server, which clients contact to find the transport address of a given service.

In both cases, we need a mechanism for mapping high-level service names into low-level encoding that can be used within packet headers of the network protocols. In its general

form, the problem is quite complex. One simplification is to break the problem into two parts: have transport addresses be a combination of machine address and local process on that machine.

***5. Storage capacity* of the subnet.** Assumptions valid at the data link layer do not necessarily hold at the transport Layer. Specifically, the subnet may buffer messages for a potentially long time, and an ``old" packet may arrive at a destination at unexpected times.

6. We need a dynamic flow control mechanism. The data link layer solution of reallocating buffers is inappropriate because a machine may have hundreds of connections sharing a single physical link. In addition, appropriate settings for the flow control parameters depend on the communicating end points (e.g., Cray supercomputers vs. PCs), not on the protocol used.

Don't send data unless there is room. Also, the network layer/data link layer solution of simply not acknowledging frames for which the receiver has no space is unacceptable. Why? In the data link case, the line is not being used for anything else; thus retransmissions are inexpensive. At the transport level, end-to-end retransmissions are needed, which wastes resources by sending the same packet over the same links multiple times. If the receiver has no buffer space, the sender should be prevented from sending data.

7. *Deal with congestion control.* In connectionless Internets, transport protocols must exercise congestion control. When the network becomes congested, they must reduce rate at which they insert packets into the subnet, because the subnet has no way to prevent itself from becoming overloaded.

8. *Connection establishment.* Transport level protocols go through three phases: establishing, using, and terminating a connection. For data gram-oriented protocols, opening a connection simply allocates and initializes data structures in the operating system kernel.

Connection oriented protocols often exchanges messages that negotiate options with the remote peer at the time a connection are opened. Establishing a connection may be tricky because of the possibility of old or duplicate packets.

Finally, although not as difficult as establishing a connection, terminating a connection presents subtleties too. For instance, both ends of the connection must be sure that all the data in their queues have been delivered to the remote application.

1.2.4.5 Session Layer

This layer allows users on different machines to establish session between them. A session allows ordinary data transport but it also provides enhanced services useful in some applications. A session may be used to allow a user to log into a remote time-

sharing machine or to transfer a file between two machines. Some of the session related services are:

1. This layer manages *Dialogue Control*. Session can allow traffic to go in both direction at the same time, or in only one direction at one time.

2. *Token management.* For some protocols, it is required that both sides don't attempt same operation at the same time. To manage these activities, the session layer provides tokens that can be exchanged. Only one side that is holding token can perform the critical operation. This concept can be seen as entering into a critical section in operating system using semaphores.

3. *Synchronization.* Consider the problem that might occur when trying to transfer a 4-hour file transfer with a 2-hour mean time between crashes. After each transfer was aborted, the whole transfer has to start again and again would probably fail. To Eliminate this problem, Session layer provides a way to insert checkpoints into data streams, so that after a crash, only the data transferred after the last checkpoint have to be repeated.

1.2.4.6 Presentation Layer

This layer is concerned with Syntax and Semantics of the information transmitted, unlike other layers, which are interested in moving data reliably from one machine to other. Few of the services that Presentation layer provides are:

1. Encoding data in a standard agreed upon way.

2. It manages the abstract data structures and converts from representation used inside computer to network standard representation and back.

1.2.4.7 Application Layer

The application layer consists of what most users think of as programs. The application does the actual work at hand. Although each application is different, some applications are so useful that they have become standardized. The Internet has defined standards for:

- File transfer (FTP): Connect to a remote machine and send or fetch an arbitrary file. FTP deals with authentication, listing a directory contents, ASCII or binary files, etc.
- Remote login (telnet): A remote terminal protocol that allows a user at one site to establish a TCP connection to another site, and then pass keystrokes from the local host to the remote host.
- Mail (SMTP): Allow a mail delivery agent on a local machine to connect to a mail delivery agent on a remote machine and deliver mail.
- News (NNTP): Allows communication between a news server and a news client.
- Web (HTTP): Base protocol for communication on the World Wide Web.

Review questions

Q-1. Why it is necessary to have layering in a network?

Ans: A computer network is a very complex system. It becomes very difficult to implement as a single entity. The layered approach divides a very complex task into small pieces each of which is independent of others and it allow a structured approach in implementing a network. The basic idea of a layered architecture is *to divide the design into small pieces*. Each layer adds to the services provided by the lower layers in such a manner that the highest layer is provided a full set of services to manage communications and run the applications.

Q-2. What are the key benefits of layered network?

Ans: Main benefits of layered network are given below:

i) Complex systems can be broken down into understandable subsystems.
ii) Any facility implemented in one layer can be made visible to all other layers.
iii) Services offered at a particular level may share the services of lower level.
iv) Each layer may be analyzed and tested independently.
v) Layers can be simplified, extended or deleted at any time.
vi) Increase the interoperability and compatibility of various components build by different vendors.

Q-3. What do you mean by OSI?

Ans: The Open System Interconnection (OSI) reference model describes how information from a software application in one computer moves through a network medium to a software application in another computer. The OSI reference model is a conceptual model composed of seven layers, each specifying particular network functions. The model was developed by the International Standardization Organization (ISO) in 1984, and it is now considered the primary architectural model for inter-computer communications.

Q-4. What are the seven layers of ISO's OSI model?

Ans:- The seven layers are:

Application Layer
Presentation Layer
Session Layer
Transport Layer
Network Layer
Data Link Layer
Physical Layer

Q-5. Briefly write functionalities of different OSI layers?

Ans: The OSI Reference Model includes seven layers. Basic functionality of each of them is as follows:

7. ***Application Layer:*** Provides Applications with access to network services.

6. ***Presentation Layer:***
Determines the format used to exchange data among networked computers.

5. ***Session Layer:*** Allows two applications to establish, use and disconnect a connection between them called a session. Provides for name recognition and additional functions like security, which are needed to allow applications to communicate over the network.

4. ***Transport Layer:*** Ensures that data is delivered error free, in sequence and with no loss, duplications or corruption. This layer also repackages data by assembling long messages into lots of smaller messages for sending, and repackaging the smaller messages into the original larger message at the receiving end.

3. ***Network Layer:*** This is responsible for addressing messages and data so they are sent to the correct destination, and for translating logical addresses and names (like a machine name FLAME) into physical addresses. This layer is also responsible for finding a path through the network to the destination computer.

2. ***Data-Link Layer:*** This layer takes the data frames or messages from the Network Layer and provides for their actual transmission. At the receiving computer, this layer receives the incoming data and sends it to the network layer for handling. The Data-Link Layer also provides error-free delivery of data between the two computers by using the physical layer. It does this by packaging the data from the Network Layer into a frame, which includes error detection information. At the receiving computer, the Data-Link Layer reads the incoming frame, and generates its own error detection information based on the received frames data. After receiving the entire frame, it then compares its error detection value with that of the incoming frames, and if they match, the frame has been received correctly.

1. ***Physical Layer*****:** Controls the transmission of the actual data onto the network cable. It defines the electrical signals, line states and encoding of the data and the connector types used. An example is 10BaseT.

Q-6. How two adjacent layers communicate in a layered network? (or What do you mean by Service Access Point?)

Ans: In layered network, each layer has various entities and entities of layer i provide service to the entities of layer i+1. The services can be accessed through service access

point (SAP), which has some address through which the layer i+1 will access the services provided by layer i.

Q-7. What are the key functions of data link layer?

Ans: Data link layer transfers data in a structured and reliable manner so that the service provided by the physical layer is utilized by data link layer. Main function of data link layer is framing and media access control.

Q8. What do you mean by Protocol?

Ans: In the context of data networking, a **protocol** *is a formal set of rules and conventions that governs how computers exchange information over a network medium.* A protocol implements the functions of one or more of the OSI layers.

Specific Instructional Objectives

At the end of this lesson, the students will be able to:

- Specify what is meant by network topology
- Classify different Network topologies
- Categorize various Network topologies
- Explain the characteristics of the following topologies:
  - Mesh
  - Bus
  - Star
  - Ring
  - Tree
  - Unconstrained

5.1.1 Introduction

Topology refers to the way in which the network of computers is connected. Each topology is suited to specific tasks and has its own advantages and disadvantages. The choice of topology is dependent upon type and number of equipment being used, planned applications and rate of data transfer required, response time, and cost. Topology can also be defined as the *geometrically interconnection pattern* by which the stations (nodes/computers) are connected using suitable transmission media (which can be point-to-point and broadcast). Various commonly used topologies are discussed in the following sections.

5.1.2 Mesh Topology

In this topology each node or station is connected to every other station as shown in Fig. 5.1.1. The key characteristics of this topology are as follows:

Key Characteristics:

- Fully connected
- Robust – Highly reliable
- Not flexible
- Poor expandability

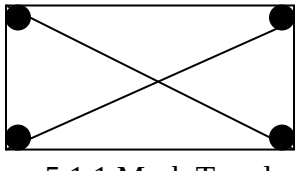


Figure 5.1.1 Mesh Topology

Two nodes are connected by dedicated point-point links between them. So the total number of links to connect n nodes = n(n-1)/2; which is proportional to n^2 . Media used for the connection (links) can be twisted pair, co-axial cable or optical fiber. With this topology there is no need to provide any additional information, that is from where the packet is coming, along with the packet because two nodes have a point-point dedicated

link between them. And each node knows which link is connected to which node on the other end.

Mesh Topology is not flexible and has a poor expandability as to add a new node n links have to be laid because that new node has to be connected to each of the existing nodes via dedicated link as shown in Fig. 5.1.2. For the same reason the cost of cabling will be very high for a larger area. And due to these reasons this topology is rarely used in practice.

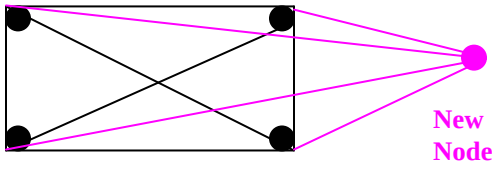


Figure 5.1.2 Adding a new node in Mesh Topology

5.1.3 Bus Topology

In Bus Topology, all stations attach through appropriate hardware interfacing known as a *tap*, directly to a linear transmission medium, or bus as shown in Fig. 5.1.3. Full-duplex operation between the station and the tap allows data to be transmitted onto the bus and received from the bus. A transmission from any station propagates the length of the medium in both directions and can be received by all other stations. At each end of the bus there is a *terminator*, which absorbs any signal, preventing reflection of signal from the endpoints. If the terminator is not present, the endpoint acts like a mirror and reflects the signal back causing interference and other problems.

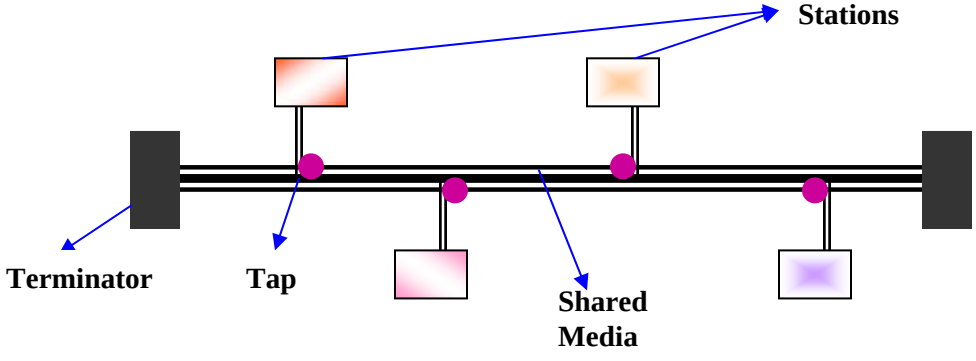


Figure 5.1.3 Bus Topology

Key Characteristics of this topology are:

- Flexible
- Expandable
- Moderate Reliability
- Moderate performance

A shared link is used between different stations. Hence it is very cost effective. One can easily add any new node or delete any node without affecting other nodes; this makes this topology easily expandable. Because of the shared medium, it is necessary to provide

some extra information about the desired destination, i.e. to explicitly specify the destination in the packet, as compared to mesh topology. This is because the same medium is shared among many nodes. As each station has a unique address in the network, a station copies a packet only when the destination address of the packet matches with the self-address. This is how data communications take place among the stations on the bus.

As there are dedicated links in the mess topology, there is a possibility of transferring data in parallel. But in bus topology, only one station is allowed to send data at a time and all other stations listen to it, as it works in a broadcast mode. Hence, only one station can transfer the data at any given time. Suitable medium access control technique should be used so as to provide some way to decide "who" will go next to send data? Usually a distributed medium access control technique, as discussed in the next lesson, is used for this purpose.

As the distance through which signal traverses increases, the attenuation increases. If the sender sends data (signal) with a small strength signal, the farthest station will not be able to receive the signal properly. While on the other hand if the transmitter sends the signal with a larger strength (more power) then the farthest station will get the signal properly but the station near to it may face over-drive. Hence, delay and signal unbalancing will force a maximum length of shared medium, which can be used in bus topology.

5.1.4 STAR Topology

In the star topology, each station is directly connected to a common central node as shown in Fig. 5.1.4. Typically, each station attaches to a central node, referred to as the *star coupler*, via two point-to-point links, one for transmission and one for reception.

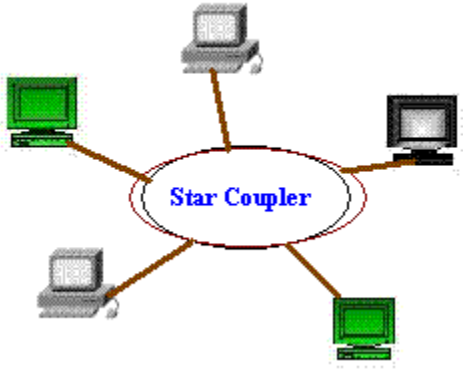


Figure 5.1.4 Star Topology

Key features:

- High Speed
- Very Flexible
- High Reliability
- High Maintainability

In general, there are two alternatives for the operation of the central node.

- One approach is for the central node to operate in a broadcast fashion. A transmission of a frame from one station to the node is retransmitted on all of the

outgoing links. In this case, although the arrangement is physically a star, it is logically a bus; a transmission from any station is received by all other stations, and only one station at a time may successfully transmit. In this case the central node acts as a *repeater*.

- Another approach is for the central node to act as a frame-switching device. An incoming frame is buffered in the node and then retransmitted on an outgoing link to the destination station. In this approach, the central node acts as a *switch* and performs the switching or routing function. This mode of operation can be compared with the working of a telephone exchange, where the caller party is connected to a single called party and each pair of subscriber who needs to talk have a different connection.

Very High speeds of data transfer can be achieved by using star topology, particularly when the star coupler is used in the switch mode. This topology is the easiest to maintain, among the other topologies. As the number of links is proportional to n, this topology is very flexible and is the most preferred topology.

5.1.5 Ring topology

In the ring topology, the network consists of a set of repeaters joined by point-to-point links in a closed loop as shown in Fig. 5.1.5. The repeater is a comparatively simple device, capable of receiving data on one link and transmitting them, bit by bit, on the other link as fast as they are received, with no buffering at the repeater. The links are unidirectional; that is data are transmitted in one direction only and all are oriented in the same way. Thus, data circulate around the ring in one direction (clockwise or counterclockwise).

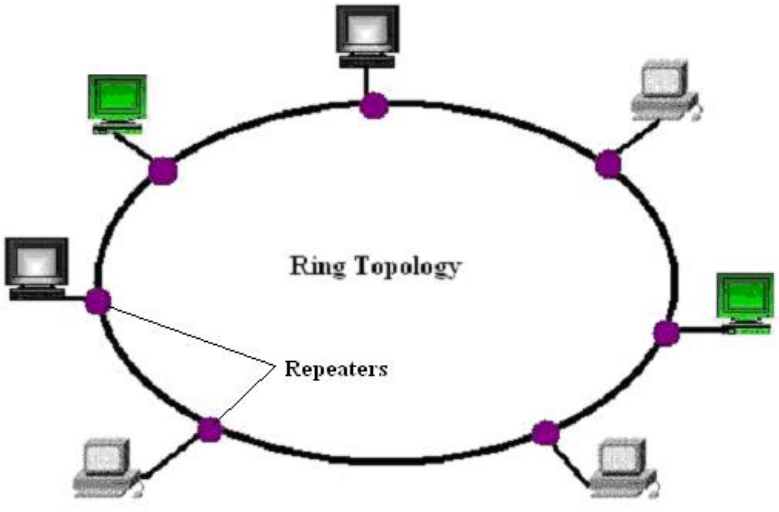


Figure 5.1.5 Ring Topology

Each station attaches to the network at a repeater and can transmit data onto the network through that repeater. As with the bus and tree, data are transmitted in frames.

As a frame circulates past all the other stations, the destination station recognizes its address and copies the frame into a local buffer as it goes by. The frame continues to circulate until it returns to the source station, where it is removed. Because multiple stations share the ring, medium access control is needed to determine at what time each station may insert frames.

How the source knows whether it has to transmit a new packet and whether the previous packet has been received properly by the destination or not. For this, the destination change a particular bit (bits) in the packet and when the receiver sees that packet with the changed bit, it comes to know that the receiver has received the packet.

This topology is not very reliable, because when a link fails the entire ring connection is broken. But reliability can be improved by using *wiring concentrator*, which helps in bypassing a faulty node and somewhat is similar to star topology.

Repeater works in the following three modes:

- **Listen mode**: In this mode, the station listens to the communication going over the shared medium as shown in Fig.5.1.6.

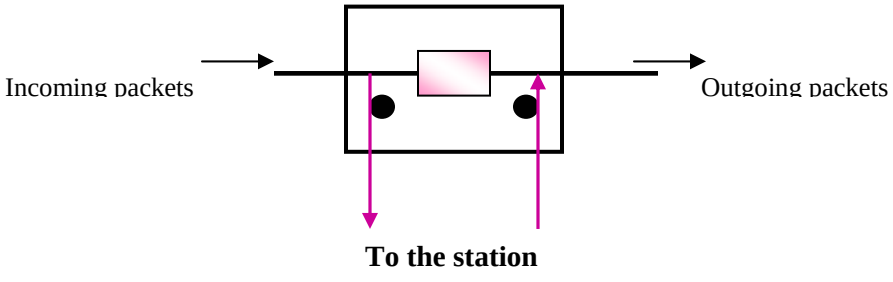


Figure 5.1.6 Repeater in Listen Mode

- **Transmit mode**: In this mode the station transmit the data over the network as shown in Fig. 5.1.7.

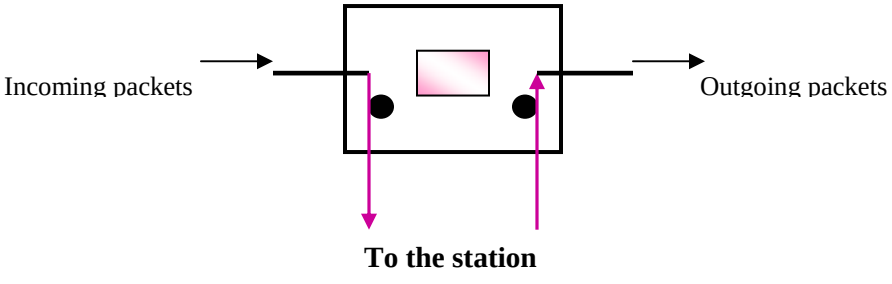


Figure 5.1.7 Repeater in Transmit Mode

- **By-Pass mode**: When the node is faulty then it can be bypassed using the repeater in bypass mode, i.e. the station doesn't care about what data is transmitted through the network, as shown in Fig. 5.1.8. In this mode there is no delay introduced because of this repeater.

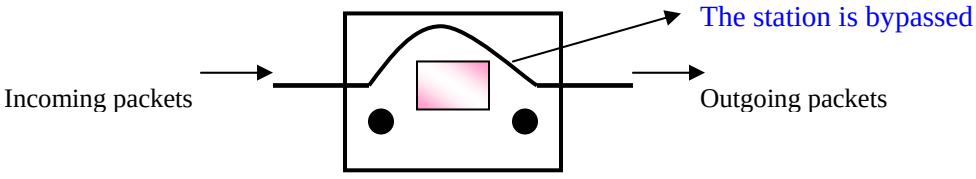


Figure 5.1.8 Repeater in Bypass Mode

5.1.6 Tree Topology

This topology can be considered as an extension to bus topology. It is commonly used in cascading equipments. For example, you have a repeater box with 8-port, as far as you have eight stations, this can be used in a normal fashion. But if you need to add more stations then you can connect two or more repeaters in a hierarchical format (tree format) and can add more stations. In the Fig. 5.1.9, R1 refers to repeater one and so on and each repeater is considered to have 8-ports.

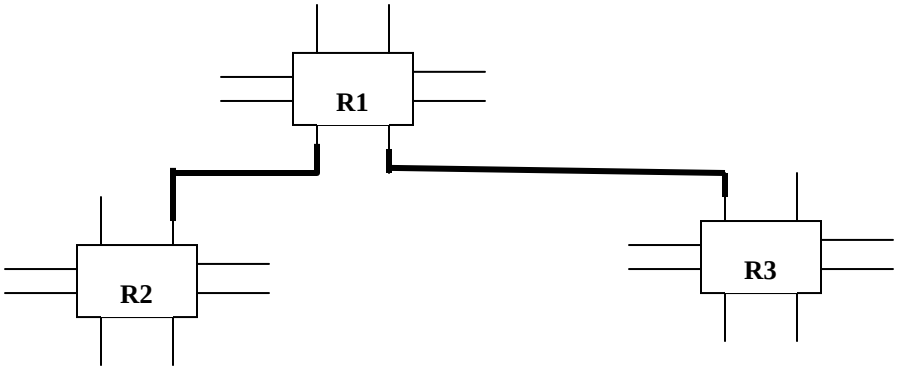


Figure 5.1.9 Tree Topology

This tree topology is very good in an organization as incremental expansion can be done in this way. Main features of this topology are scalability and flexibility. This is because, when the need arises for more stations that can be accomplished easily without affecting the already established network.

5.1.7 Unconstrained Topology

All the topologies discussed so far are symmetric and constrained by well-defined interconnection pattern. However, sometimes no definite pattern is followed and nodes are interconnected in an arbitrary manner using point-to-point links as shown in Fig 5.1.10. Unconstrained topology allows a lot of configuration flexibility but suffers from the complex routing problem. Complex routing involves unwanted overhead and delay.

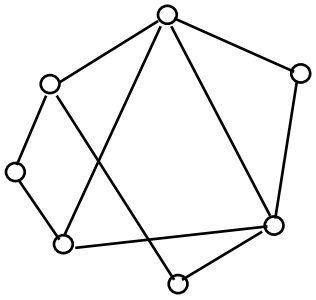


Figure 5.1.10 Unconstrained Topology

5.1.8 Combination of topology and transmission media

Topology and transmission media are interrelated. For example, all the important criteria of a network such as reliability, expandability and performance depend on both the topology and the transmission media used in the network. As a consequence, these two aspects are interrelated. Let us have a look at the various transmission media, which are used for different topologies.

- Twisted pair is suitable for use in star and ring topologies
  - *Cat 3*: voice grade UTP, data rate up to 10 Mbps
  - *Cat 5*: data grade UTP, data rate up to 100 Mbps
- Coaxial cable is suitable for use in bus topology
  - Baseband coaxial cable supports data rates of 20 Mbps at distances up to 2 Km.
- Fiber optics is suitable for use in ring and star topology
  - Gigabit data rates and longer distances.
- Unguided media are suitable for star topology

Fill In The Blanks.

1. Number of links to connect n nodes in a mesh topology is = _____.
2. Mesh Topology is _____ flexible and has a _____ expandability
3. In BUS topology, at each end of the bus is a _____, which absorbs any signal, removing it from the bus.
4. In BUS topology, One can easily add any new node or delete any node with-out affecting other nodes; this makes this topology easily _____.
5. _____ and _____ will force a maximum length of shared medium which can be used in BUS topology.
6. The two alternatives for the operation of the central node in STAR topology are: _____ and _____.
7. In Ring Topology, the links are _____; that is, data are transmitted in _____ direction only and all are oriented in the same way

8. In Ring Topology, Repeater works in 3 modes: _____, _____ and _____.
9. _____ topology can be considered as an extension to BUS topology.
10. _____ is suitable for use in star and ring topologies
11. Coaxial cable is suitable for use in _____ topology.

Solutions.

1. n(n-1)/2
2. not, poor
3. terminator
4. expandable.
5. Delay, signal unbalancing
6. repeater, switch
7. unidirectional, one
8. Listen, Transmit, By-Pass
9. Tree
10. Twisted pair
11. BUS

Short Answer Questions:

Q-1. List out the advantages and drawbacks of bus topology.

Ans: Advantages:

i) Easy to implement
ii) It is very cost effective because only a single segment required
iii) It is very flexible
iv) Moderate reliability.
v) Can add new station or delete any station easily (scalable)

Disadvantages:

i) Required suitable medium access control technique.
ii) Maximum cable length restriction imposed due to delay and signal unbalancing problem.

Q-2. List out the advantages and drawbacks of ring topology.

Ans: Advantages:

i) Data insertion, data reception and data removal can be provided by repeater
ii) It can provide multicast addressing.
iii) Point-to-point links to its adjacent nodes (moderate cost)

Disadvantages:

i) The repeater introduces a delay
ii) The topology fails if any link disconnects or a node fails.
iii) Direct link not provided
iv) It provides complex management

Q-3. Why star topology is commonly preferred?

Ans: It gives high reliability, more flexible and higher bandwidth. Since there is a central control point, the control of network is easy and priority can be given to selected nodes.

Q-4. Is there any relationship between transmission media and topology?

Ans: Yes, medium should be selected based on the topology. For example, for bus topology coaxial cable medium is suitable, and for ring/star topology twisted-pair or optical fiber can be used.

Specific Instructional Objectives

At the end of this lesson the student will be able to:

- Understand the need for circuit switching
- Specify the components of a switched communication network
- Explain how circuit switching takes place
- Explain how switching takes place using space-division and time-division switching
- Explain how routing is performed
- Explain how signalling in performed

4.1.1 Introduction

When there are many devices, it is necessary to develop suitable mechanism for communication between any two devices. One alternative is to establish point-to-point communication between each pair of devices using **mesh topology.** However, mesh topology is impractical for large number of devices, because the number of links increases exponentially (n(n-1)/2, where n is the number of devices) with the number of devices. A better alternative is to use switching techniques leading to switched communication network. In the **switched network** methodology, the network consists of a set of interconnected nodes, among which information is transmitted from source to destination via different routes, which is controlled by the switching mechanism. A basic model of a switched communication is shown in Fig. 4.1.1. The end devices that wish to communicate with each other are called *stations*. The switching devices are called *nodes*. Some nodes connect to other nodes and some are to connected to some stations. Key features of a switched communication network are given below:

- Network Topology is not regular.
- Uses FDM or TDM for node-to-node communication.
- There exist multiple paths between a source-destination pair for better network reliability.
- The switching nodes are not concerned with the contents of data.
- Their purpose is to provide a switching facility that will move data from node to node until they reach the destination.

The switching performed by different nodes can be categorized into the following three types:

- Circuit Switching
- Packet Switching
- Message Switching

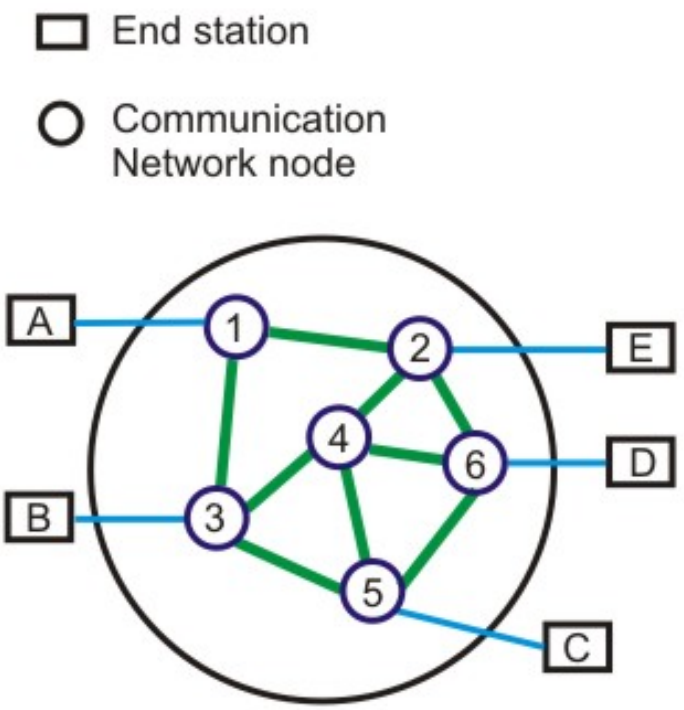


Figure 4.1.1 Basic model of a switched communication network

In this lesson we shall discuss various aspects of circuit switching and discuss how the Public Switched Telephone Network (PSTN), which is based on circuit switching, works.

4.1.2 Circuit switching Technique

Communication via circuit switching implies that there is a dedicated communication path between the two stations. The path is a connected through a sequence of links between network nodes. On each physical link, a logical channel is dedicated to the connection. Circuit switching is commonly used technique in telephony, where the caller sends a special message with the address of the callee (i.e. by dialling a number) to state its destination. It involved the following three distinct steps, as shown in Fig. 4.1.2.

Circuit Establishment: To establish an end-to-end connection before any transfer of data. Some segments of the circuit may be a dedicated link, while some other segments may be shared.

Data transfer:

- Transfer data is from the source to the destination.
- The data may be analog or digital, depending on the nature of the network.
- The connection is generally full-duplex.

Circuit disconnect:

- Terminate connection at the end of data transfer.
- Signals must be propagated to deallocate the dedicated resources.

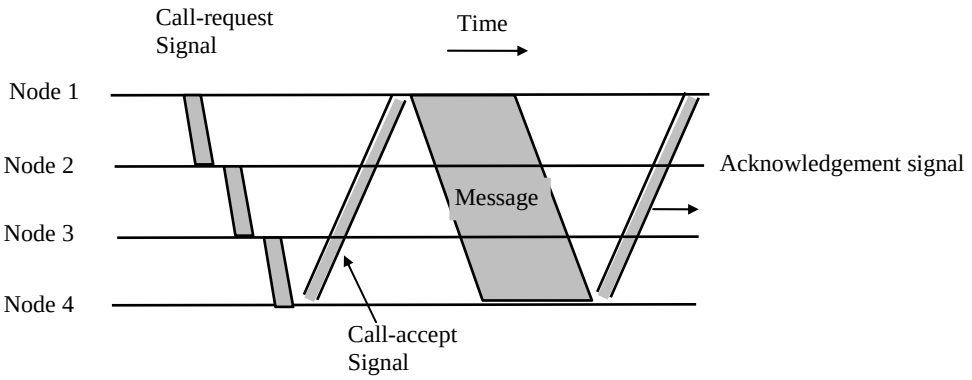


Figure 4.1.2 Circuit Switching technique

Thus the actual physical electrical path or circuit between the source and destination host must be established before the message is transmitted. This connection, once established, remains exclusive and continuous for the complete duration of information exchange and the circuit becomes disconnected only when the source wants to do so.

4.1.3 Switching Node

Let us consider the operation of a single circuit switched node comprising a collection of stations attached to a central switching unit, which establishes a dedicated path between any two devices that wish to communicate.

Major elements of a single-node network are summarized below:

- *Digital switch*: That provides a transparent (full-duplex) signal path between any pair of attached devices.
- *Network interface:* That represents the functions and hardware needed to connect digital devices to the network (like telephones).
- *Control unit:* That establishes, maintains, and tears down a connection.

The simplified schematic diagram of a switching node is shown in Fig. 4.1.3. An important characteristic of a circuit-switch node is whether it is *blocking* or *non-blocking*. A blocking network is one, which may be unable to connect two stations because all possible paths between them are already in use. A non-blocking network permits all stations to be connected (in pairs) at once and grants all possible connection requests as long as the called party is free. For a network that supports only voice traffic, a blocking configuration may be acceptable, since most phone calls are of short duration. For data applications, where a connection may remain active for hours, non-blocking configuration is desirable.

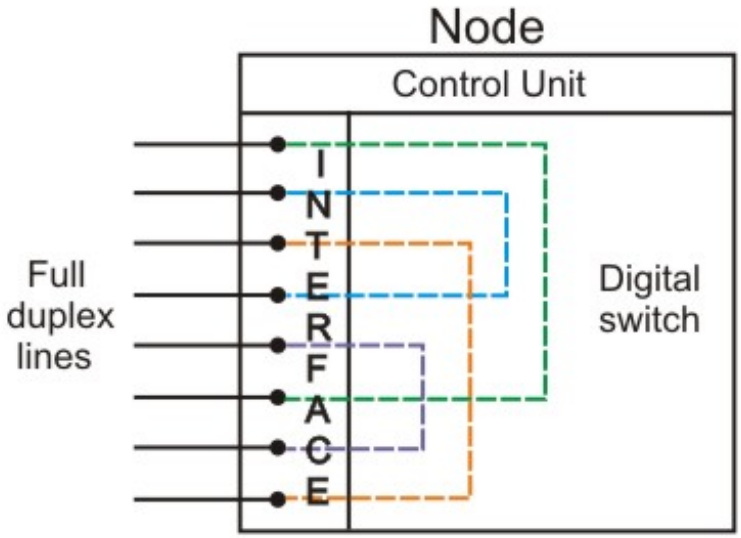


Figure 4.1.3 Schematic diagram of a switching node.

Circuit switching uses any of the three technologies: **Space-division** switches, **Time-division** switches or a **combination of both**. In Space-division switching, the paths in the circuit are separated with each other spatially, i.e. different ongoing connections, at a same instant of time, uses different switching paths, which are separated spatially. This was originally developed for the analog environment, and has been carried over to the digital domain. Some of the space switches are crossbar switches, Multi-stage switches (e.g. Omega Switches). A **crossbar** switch is shown in Fig. 4.1.4. Basic building block of the switch is a metallic crosspoint or semiconductor gate that can be enabled or disabled by a control unit.

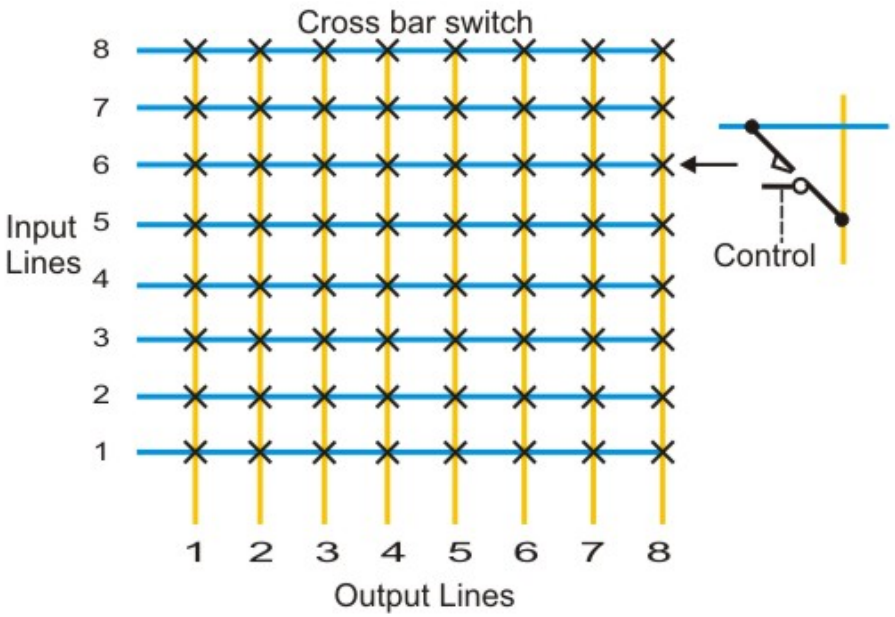


Figure 4.1.4 Schematic diagram of a crossbar switch

Example: Xilinx crossbar switch using FPGAs. It is based on reconfigurable routing infrastructure. It is a high-speed high capacity nonblocking type switch with sizes varying from 64X64 to 1024X1024 and data rate of 200 Mbps.

Limitations of crossbar switches are as follows:

- The number of crosspoints grows with the square of the number of attached stations.
- Costly for a large switch.
- The failure of a crosspoint prevents connection between the two devices whose lines intersect at that crosspoint.
- The crosspoints are inefficiently utilized.
- Only a small fraction of crosspoints are engaged even if all of the attached devices are active.

Some of the above problems can be overcome with the help of *multistage space division* switches. By splitting the crossbar switch into smaller units and interconnecting them, it is possible to build multistage switches with fewer crosspoints.

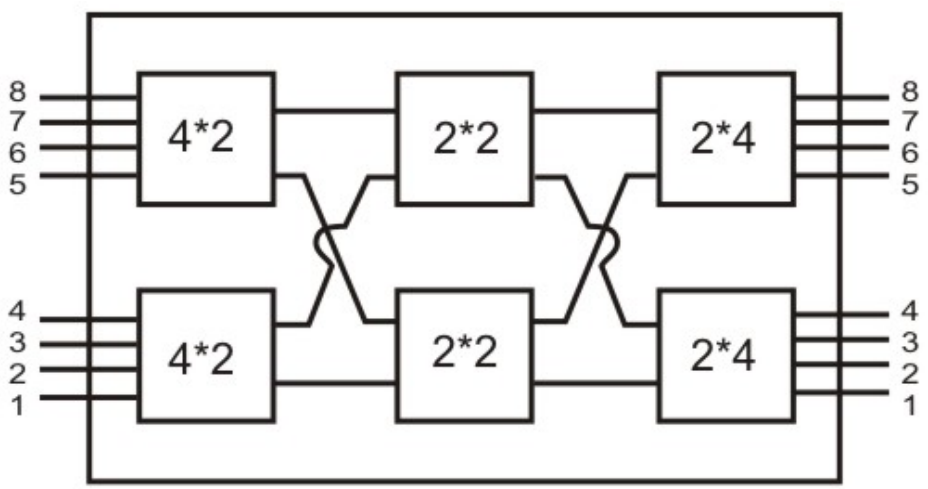


Figure 4.1.5 A three-stage space division switch

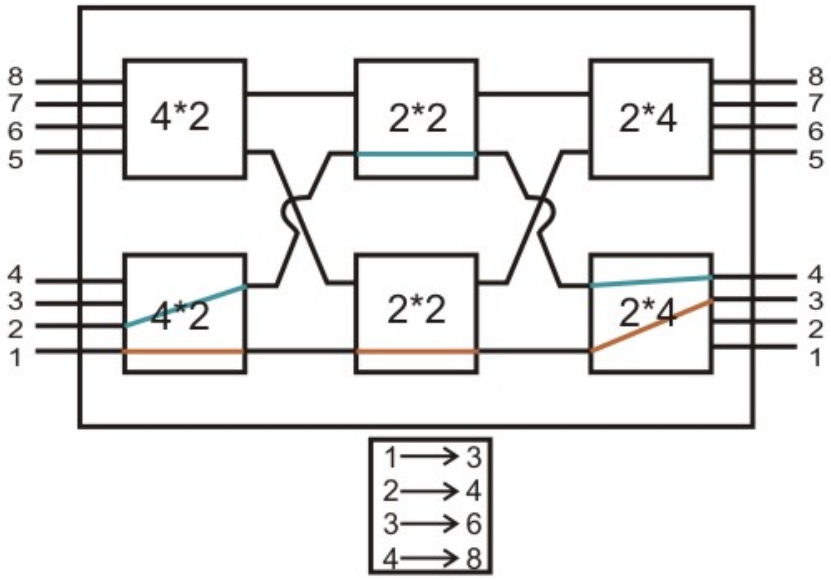


Figure 4.1.6 Block nature of the switch

Figure 4.1.5 shows a three-stage space division switch. In this case the number of crosspoints needed goes down from 64 to 40. There is more than one path through the network to connect two endpoints, thereby increasing reliability. Multistage switches may lead to *blocking*. The problem may be tackled by increasing the number or size of the intermediate switches, which also increases the cost. The blocking feature is illustrated in Fig. 4.1.6. As shown in Fig. 4.1.6, after setting up connections for 1-to-3 and 2-to-4, the switch cannot establish connections for 3-to-6 and 4-to-5.

Time Division Switching

Both voice and data can be transmitted using digital signals through the same switches. All modern circuit switches use digital time-division multiplexing (TDM) technique for establishing and maintaining circuits. Synchronous TDM allows multiple low-speed bit streams to share a high-speed line. A set of inputs is sampled in a round robin manner. The samples are organized serially into slots (channels) to form a recurring frame of slots. During successive time slots, different I/O pairings are enabled, allowing a number of connections to be carried over the shared bus. To keep up with the input lines, the data rate on the bus must be high enough so that the slots recur sufficiently frequently. For 100 full-duplex lines at 19.200 Kbps, the data rate on the bus must be greater than 1.92 Mbps. The source-destination pairs corresponding to all active connections are stored in the control memory. Thus the slots need not specify the source and destination addresses. Schematic diagram of time division switching is shown in Fig. 4.1.7.

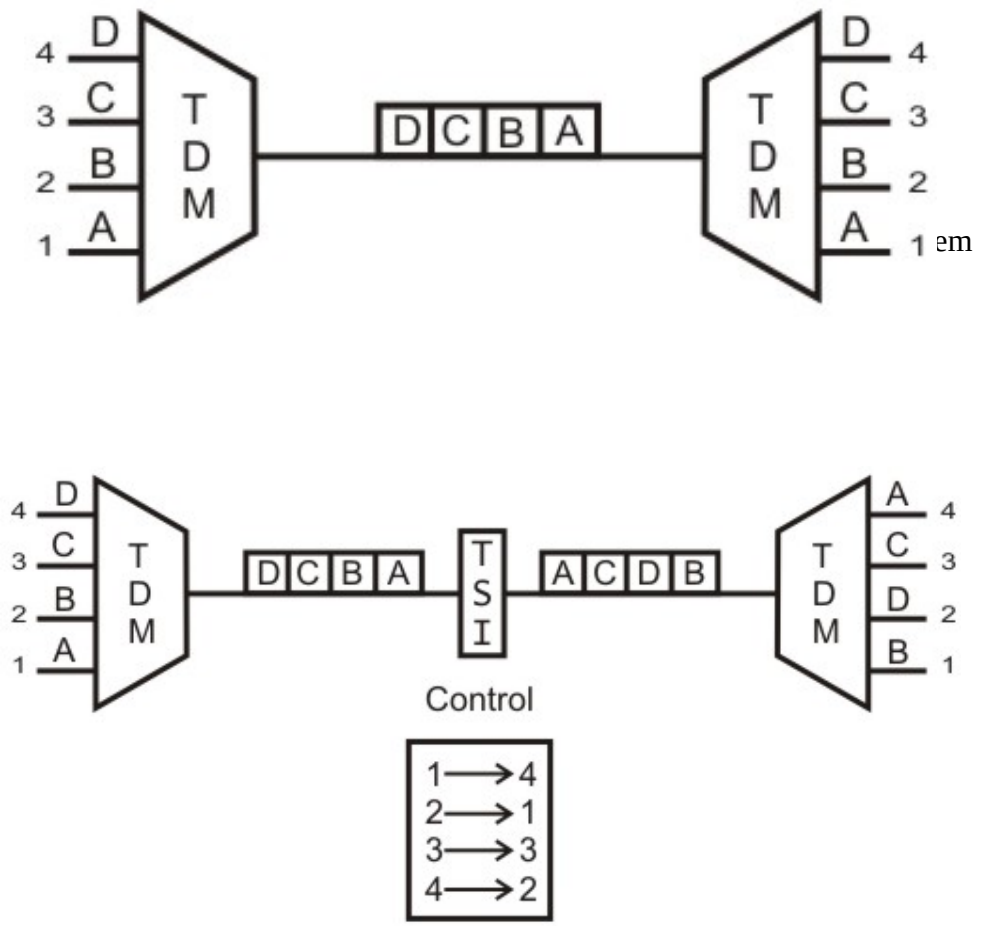


Figure 4.1.8 TDM with Switching using TSI

Time-division switching uses time-division multiplexing to achieve switching, i.e. different ongoing connections can use same switching path but at different interleaved time intervals. There are two popular methods of time-division switching namely, Time-Slot Interchange (TSI) and the TDM bus. TSI changes the ordering of the slots based on desired connection and it has a random-access memory to store data and flip the time slots as shown in Fig. 4.1.8. The operation of a TSI is depicted in Fig. 4.1.9. As shown in the figure, writing can be performed in the memory sequentially, but data is read selectively. In TDM bus there are several input and outputs connected to a high-speed bus. During a time slot only one particular output switch is closed, so only one connection at a particular instant of time as shown in Fig. 4.1.10.

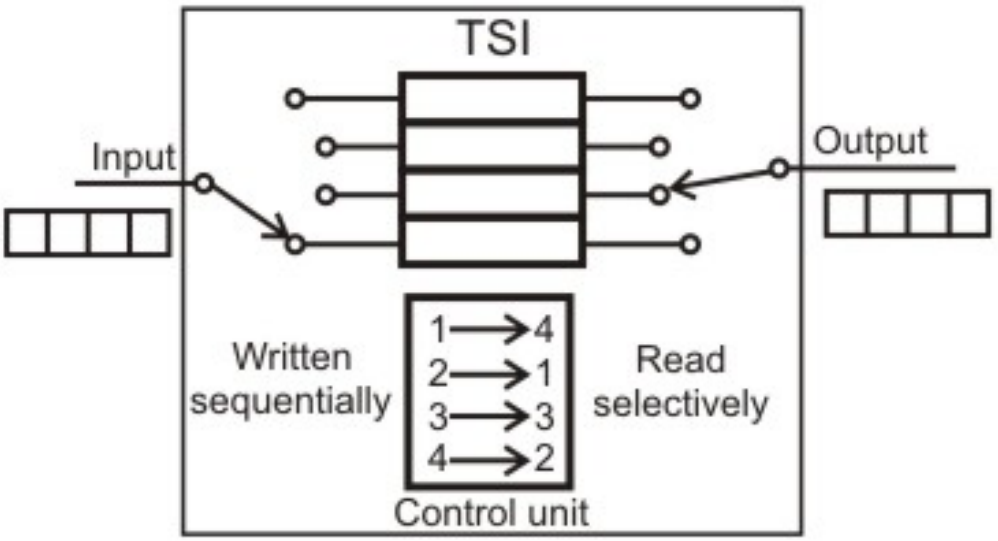


Figure 4.1.9 Operation of a TSI

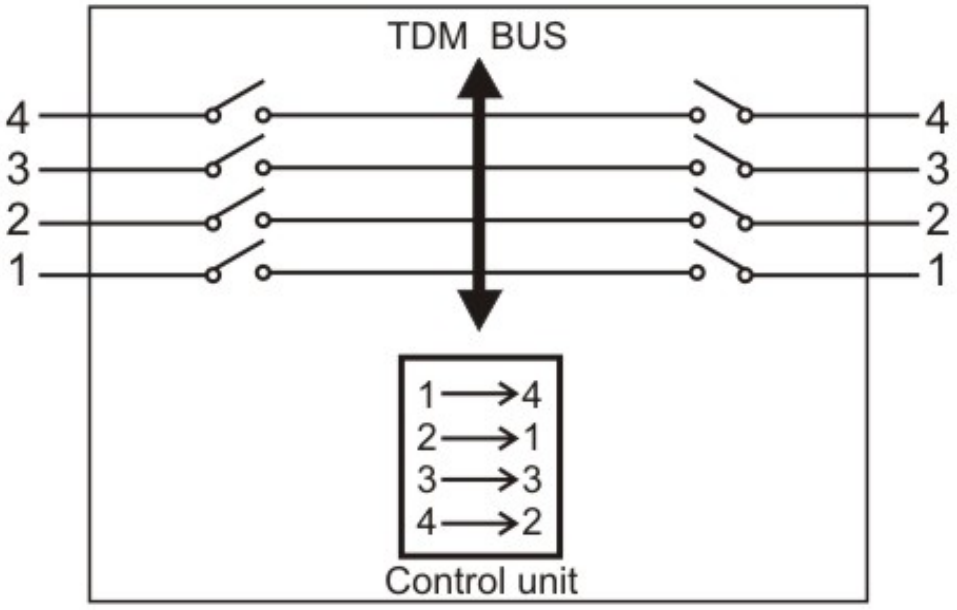


Figure 4.1.10 TDM bus switching

4.1.4 Public Switched Telephone Networks

Public switched telephone network (PSTN) is an example of circuit-switched network. It's also known as Plain Old Telephone Service (POTS). The switching centres used for the switching are organised in different levels, namely: Regional offices (class 1), Section offices (class 2), primary offices (class 3), Toll offices (class 4) and finally End offices

(class 5) as shown in Fig. 4.1.11. Level 1 is at the highest level and Level 5 is the lowest level. Subscribers or the customers are directly connected to these end offices. And each office is connected directly to a number of offices at a level below and mostly a single office at higher level.

Subscriber Telephones are connected, through **Local Loops** to end offices (or central offices). A small town may have only one end office, but large cities have several end offices. Many end offices are connected to one Toll office, which are connected to primary offices. Several primary offices are connected to a section office, which normally serves more than one state. All regional offices are connected using mesh topology. Accessing the switching station at the end offices is accomplished through dialling. In the past, telephone featured rotary or pulse dialling, in which digital signals were sent to the end office for each dialled digit. This type of dialling was prone to errors due to inconsistency in humans during dialling. Presently, dialling is accomplished by Touch-Tone technique. In this method the user sends a small burst of frequency called dual tone, because it is a combination of two frequencies. This combination of frequencies sent depends on the row and column of the pressed pad.

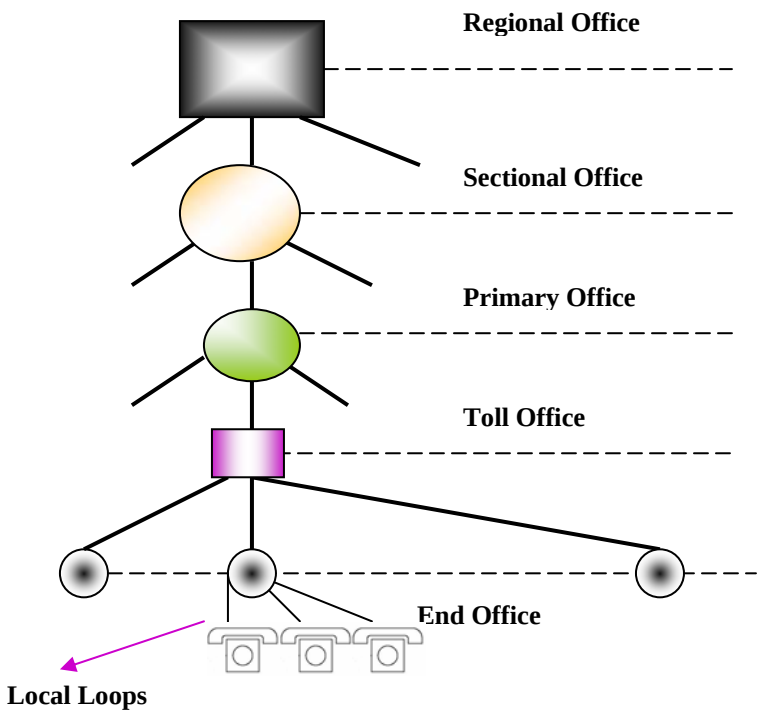


Figure 4.1.11 Basic organization of a Public Switched Telephone Network (PSTN)

The connections are multiplexed when have to send to a switching office, which is one level up. For example, Different connections will be multiplexed when they are to be forwarded from an end-office to Toll office. Figure 4.1.12 shows a typical medium distance telephone circuit.

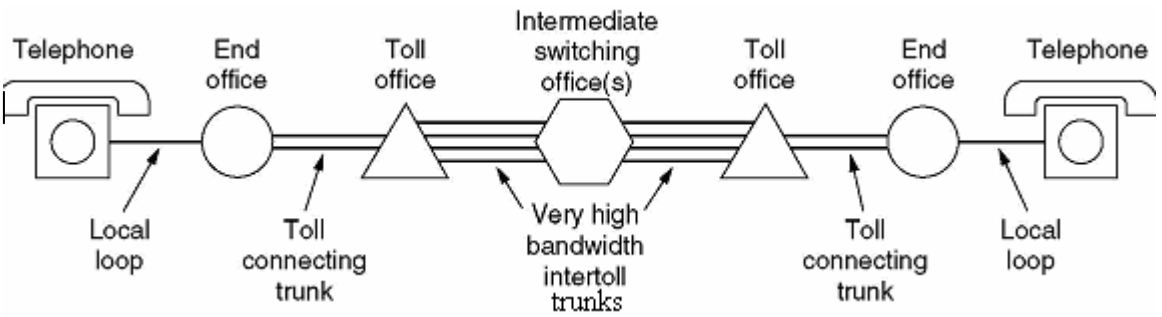


Figure 4.1.12 Typical medium distance telephone circuit

Fill In the Blanks:

1. _____ uses the entire capacity of the link.
2. In _____ switching, each packet of a message need not follow the same path from sender to receiver.
3. In _____ switching all the datagrams of a message follows the same channel of a path.
4. PSTN is an example of _____ network.
5. PSTN is also known as _____.

Ans:

1. Circuit switching
2. Datagram packet
3. virtual circuit
4. circuit switching
5. plain old telephone service (POTS)

Short Answer Questions

Q-1. What are the three basic steps involved in data communication through circuit switching?

Ans: The steps are:

i) Circuit establishment (before data transfer)
ii) Circuit maintenance (When data transfer is going on)
iii) Circuit disconnect (When data transfer is over)

Q-2. Mention the key advantages and disadvantages of circuit switching technique.

Ans: Advantages:

i) After path is established, data communication without delay.
ii) Very suitable for continuous traffic.
iii) It establishes a dedicated path.
iv) No overhead after call setup.
v) It is transparent and data passes in order.

Disadvantages:

i) Provide initial delay for setting up the call.
ii) Inefficient for bursty traffic.
iii) Data rate should be same because of fixed bandwidth.
iv) When load increases, some calls may be blocked.

Q-3. Why data communication through circuit switching is not efficient?

Ans: In data communication, traffic between terminal and server are not continuous. Sometimes more data may come or sometimes there is no data at all. Circuit switching is not efficient because of its fixed bandwidth.

Q-4. Compare the performance of space-division single-stage switch with multi-stage switch.

Ans: Space-division single-stage switch requires more number of crosspoints, nonblocking in nature but provides no redundant path. On the other hand multi-stage switches require lesser number of crosspoints, blocking in nature but provides redundant paths.

4.2.0 Specific Instructional Objectives

At the end of this lesson the student will be able to:

- Explain the need for packet switching
- Explain how packet switching takes place
- Explain different types of packet switching techniques
- Distinguish between virtual-circuit and datagram type packet switching
- Compare circuit switching with packet switching

4.2.1 Introduction

In the preceding lesson we have discussed about circuit switching. In circuit switching, network resources are dedicated to a particular connection. Although this satisfies the requirement of voice communication, it suffers from the following two shortcomings for data communication:

- In a typical user/host data connection, line utilization is very low.
- Provides facility for data transmission at a constant rate.

However, for information transmission applications, the circuit switching method is very slow, relatively expensive and inefficient. First of all, the need to establish a dedicated connection before sending the message itself inserts a delay time, which might become significant for the total message transfer time. Moreover, the total channel remains idle and unavailable to the other users once a connection is made. On the other hand once a connection is established, it is guaranteed and orderly delivery of message is ensured. Unfortunately, the data transmission pattern may not ensure this, because data transmission is bursty in nature. As a consequence, it limits the utility of the method. The problem may be overcome by using an approach known as message switching, which is discussed in Sec. 4.2.2. However, message switching suffers from various problems as discussed in Sec. 4.2.3. To overcome the limitations of message switching, another switching technique, known as packet switching was invented. Various aspects of packet switching have been discussed in Sec. 4.2.4.

4.2.2 Message Switching

In this switching method, a different strategy is used, where instead of establishing a dedicated physical line between the sender and the receiver, the message is sent to the nearest directly connected switching node. This node stores the message, checks for errors, selects the best available route and forwards the message to the next intermediate node.

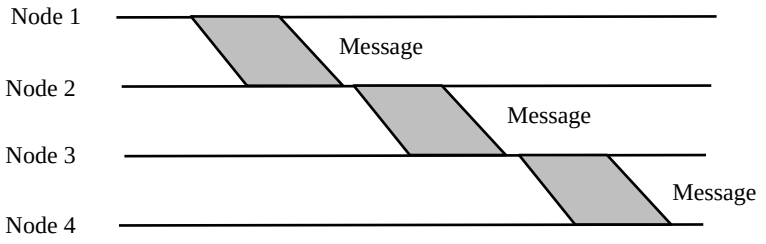


Figure 4.2.1 Message Switching Technique

The line becomes free again for other messages, while the process is being continued in some other nodes. Due to the mode of action, this method is also known as *store-and-forward technology* where the message hops from node to node to its final destination. Each node stores the full message, checks for errors and forwards it.

In this switching technique, more devices can share the network bandwidth, as compared with circuit switching technique. Temporary storage of message reduces traffic congestion to some extent. Higher priority can be given to urgent messages, so that the low priority messages are delayed while the urgent ones are forwarded faster. Through broadcast addresses one message can be sent to several users. Last of all, since the destination host need not be active when the message is sent, message switching techniques improve global communications.

However, since the message blocks may be quite large in size, considerable amount of storage space is required at each node to buffer the messages. A message might occupy the buffers for minutes, thus blocking the internodal traffic.

Basic idea:
- Each network node receives and stores the message
- Determines the next leg of the route, and
- Queues the message to go out on that link.

Advantages:
- Line efficiency is greater (sharing of links).
- Data rate conversion is possible.
- Even under heavy traffic, packets are accepted, possibly with a greater delay in delivery.
- Message priorities can be used, to satisfy the requirements, if any.

Disadvantages: Message of large size monopolizes the link and storage

4.2.3 Packet Switching

The basic approach is not much different from message switching. It is also based on the same ‘store-and-forward’ approach. However, to overcome the limitations of message switching, messages are divided into subsets of equal length called *packets*. This approach was developed for long-distance data communication (1970) and it has evolved

over time. In packet switching approach, data are transmitted in short packets (few Kbytes). A long message is broken up into a series of packets as shown in Fig. 4.2.2. Every packet contains some control information in its header, which is required for routing and other purposes.

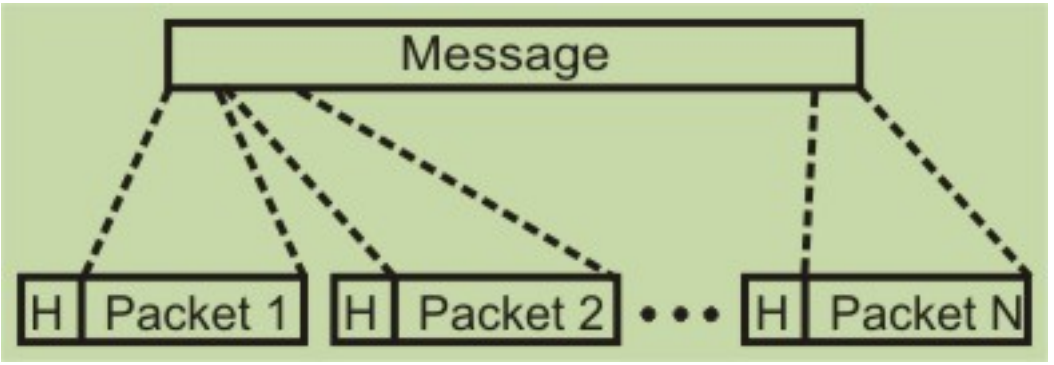


Figure 4.2.2 A message is divided into a number of equal length short packets

Main difference between Packet switching and Circuit Switching is that the communication lines are not dedicated to passing messages from the source to the destination. In Packet Switching, different messages (and even different packets) can pass through different routes, and when there is a "dead time" in the communication between the source and the destination, the lines can be used by other sources.

There are two basic approaches commonly used to packet Switching: **virtual-circuit** packet switching and **datagram** packet switching. In virtual-circuit packet switching a virtual circuit is made before actual data is transmitted, but it is different from circuit switching in a sense that in circuit switching the call accept signal comes only from the final destination to the source while in case of virtual-packet switching this call accept signal is transmitted between each adjacent intermediate node as shown in Fig. 4.2.3. Other features of virtual circuit packet switching are discussed in the following subsection.

4.2.3.1 Virtual Circuit Packet Switching Networks

An initial setup phase is used to set up a route between the intermediate nodes for all the packets passed during the session between the two end nodes. In each intermediate node, an entry is registered in a table to indicate the route for the connection that has been set up. Thus, packets passed through this route, can have short headers, containing only a *virtual circuit identifier* (VCI), and not their destination. Each intermediate node passes the packets according to the information that was stored in it, in the setup phase. In this way, packets arrive at the destination in the correct sequence, and it is guaranteed that essentially there will not be errors. This approach is slower than Circuit Switching, since different virtual circuits may compete over the same resources, and an initial setup phase is needed to initiate the circuit. As in Circuit Switching, if an intermediate node fails, all virtual circuits that pass through it are lost. The most common forms of Virtual Circuit

networks are X.25 and Frame Relay, which are commonly used for public data networks (PDN).

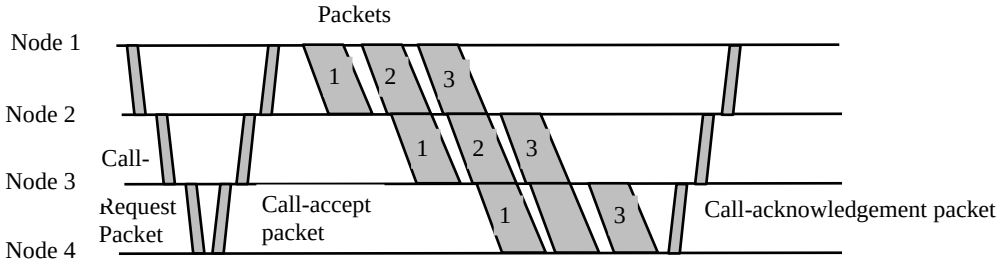


Figure 4.2.3 Virtual circuit packet switching technique

4.2.3.2 Datagram Packet Switching Networks

This approach uses a different, more dynamic scheme, to determine the route through the network links. Each packet is treated as an independent entity, and its header contains full information about the destination of the packet. The intermediate nodes examine the header of the packet, and decide to which node to send the packet so that it will reach its destination. In the decision two factors are taken into account:

- The shortest ways to pass the packet to its destination - protocols such as RIP/OSPF are used to determine the shortest path to the destination.
- Finding a free node to pass the packet to - in this way, bottlenecks are eliminated, since packets can reach the destination in alternate routes.

Thus, in this method, the packets don't follow a pre-established route, and the intermediate nodes (the routers) don't have pre-defined knowledge of the routes that the packets should be passed through.

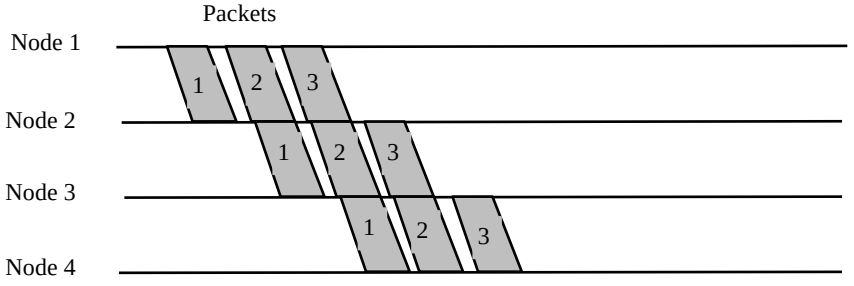


Figure 4.2.4 Datagram Packet switching

Packets can follow different routes to the destination, and delivery is not guaranteed (although packets usually do follow the same route, and are reliably sent). Due to the nature of this method, the packets can reach the destination in a different order

than they were sent, thus they must be sorted at the destination to form the original message. This approach is time consuming since every router has to decide where to send each packet. The main implementation of Datagram Switching network is the Internet, which uses the IP network protocol.

Advantages:

- Call setup phase is avoided (for transmission of a few packets, datagram will be faster).
- Because it is more primitive, it is more flexible.
- Congestion/failed link can be avoided (more reliable).

Problems:

- Packets may be delivered out of order.
- If a node crashes momentarily, all of its queued packets are lost.

4.2.3.3 Packet Size

In spite of increase in overhead, the transmission time may decreases in packet switching technique because of parallelism in transmission as shown in Fig. 4.2.5.

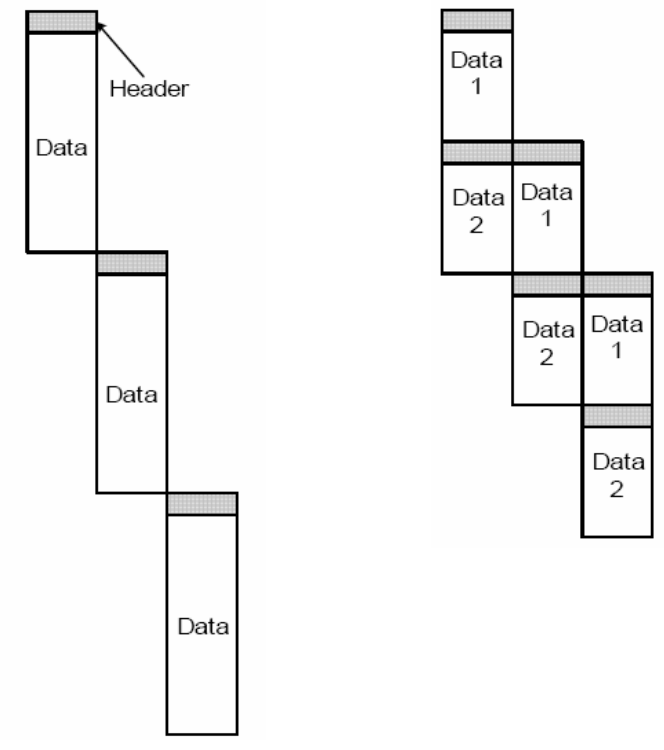


Figure 4.2.5 Reduction of transmission time because of parallelism in transmission in packet switching technique

However, question arises about the optimal size of size of a packet. As packet size is decreased, the transmission time reduces until it is comparable to the size of control information. There is a close relationship between packet size and transmission time as shown in Fig. 4.2.6. In this case it is assumed that there is a virtual circuit from station X to Y through nodes a and b. Times required for transmission decreases as each message is divided into 2 and 5 packets. However, the transmission time increases if each message is divided into 10 packets.

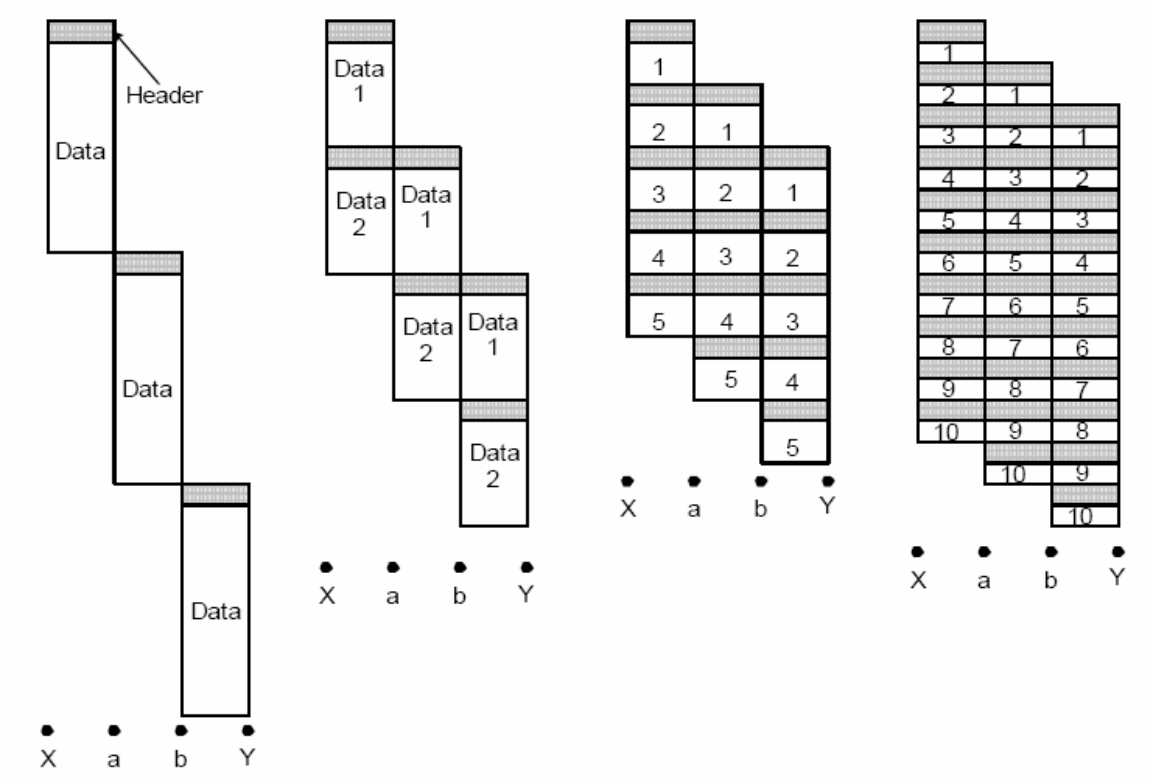


Figure 4.2.6 Variation of transmission time with packet size

4.2.3.4 Virtual Circuit Versus Datagram Packet Switching

Key features of the virtual circuit packet switching approach is as follows:

- Node need not decide route
- More difficult to adopt to congestion
- Maintains sequence order
- All packets are sent through the same predetermined route

On the other hand, the key features of the datagram packet switching are as follows:

- Each packet is treated independently
- Call set up phase is avoided
- Inherently more flexible and reliable

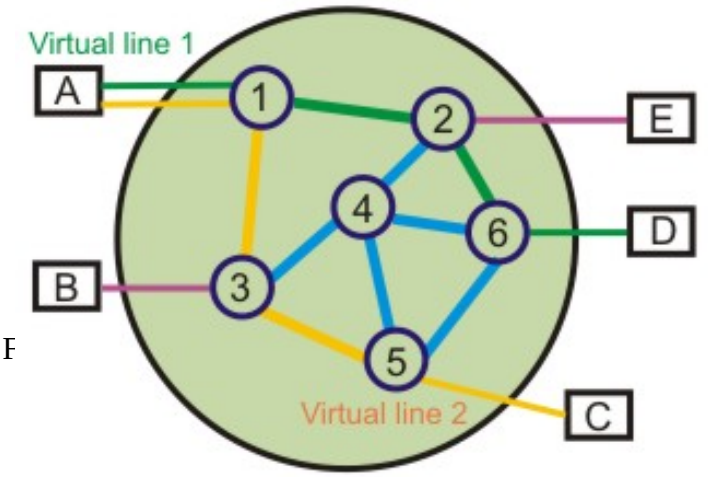
4.2.3.5 External and Internal Operations

There are two dimensions to the problem of whether to use virtual circuit or datagram in a particular situation:

- At the interface between a station and a network node, we may have connection-oriented or connectionless service.
- Internally, the network may use virtual circuits or datagrams.

This leads us to four different scenarios using different VC/DG combinations, which are discussed below.

Scenario 1: External virtual circuit, Internal virtual circuit
In this case a user requests a virtual circuit and a dedicated route through the network is constructed. All packets follow the same route as shown in Fig. 4.2.7.



F circuit

Scenario 2: External virtual circuit, Internal datagram
In this case, the network handles each packet separately. Different packets for the same external virtual circuit may take different routes as shown in Fig. 4.2.8. The network buffers packets, if necessary, so that they are delivered to the destination in the proper order.

Scenario 3: External datagram, Internal datagram
In this case each packet is treated independently from both the user's end and the network's point of view as shown in Fig. 4.2.9.

Scenario 4: External datagram, Internal virtual circuit
In this case, an external user does not see any connections - it simply sends packets one at a time as shown in Fig. 4.2.10. The network sets up a logical connection between stations for packet delivery. May leave such connections in place for an extended period, so as to satisfy anticipated future needs.

A comparison of different switching techniques is given in Table 4.2.1

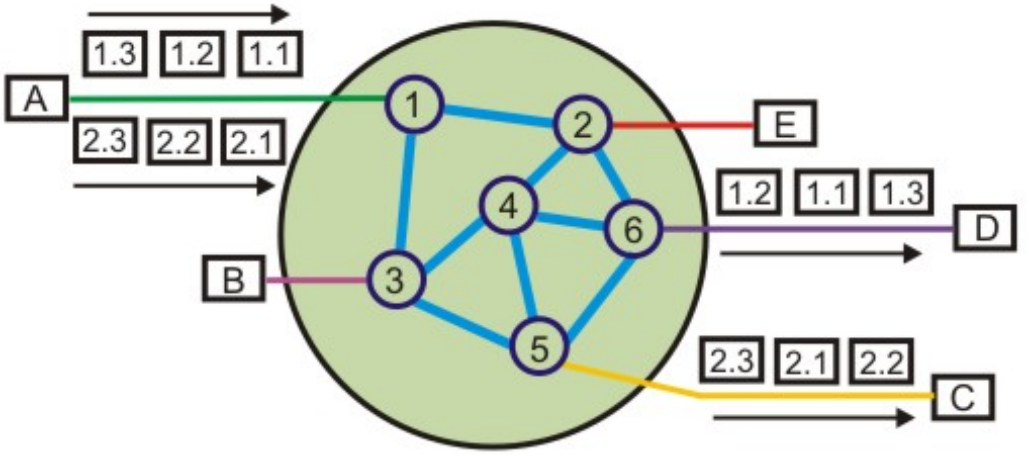


Figure 4.2.8 External virtual circuit and internal datagram

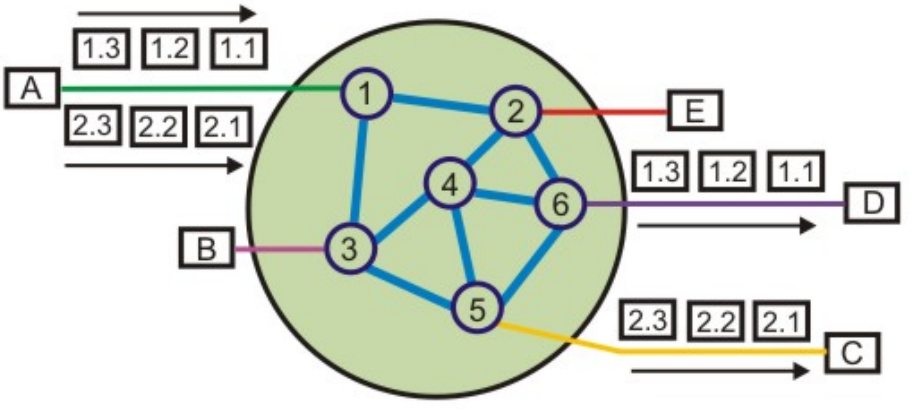


Figure 4.2.9 External datagram and internal datagram

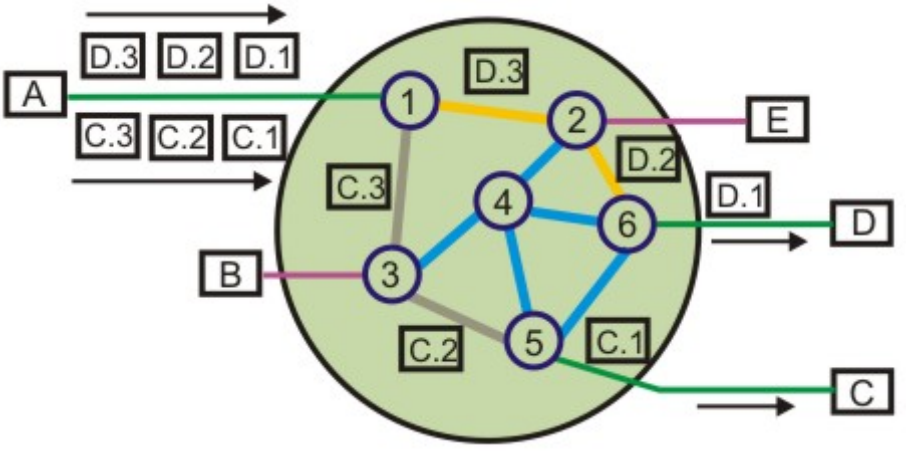


Figure 4.2.10 External datagram and internal virtual circuit

Table 4.2.1 Comparison of the three switching techniques

Circuit Switching	Datagram Packet	Virtual Circuit Packet
Dedicated path	No dedicated path	No dedicated path
Path established for entire conversation	Route established for each packet	Route established for entire conversation
Call set up delay	Packet transmission delay	Call set up delay, Packet transmission delay
Overload may block call set up	Overload increases packet delay	Overload may block call set up and increases packet delay
No speed or code conversion	Speed or code conversion	Speed or code conversio
Fixed bandwidth	Dynamic bandwidth	Dynamic bandwidth
No overhead bits after call set up	Overhead bits in each packet	Overhead bits in each packet

Fill In the Blanks:

1. A switched virtual circuit involves _____.
2. A permanent virtual circuit involves _____.
3. Two basic approaches are common to Packet Switching are _____ packet switching and _____ packet switching.
4. X.25 is a standard for _____ communications.

Ans:

1. creation of link as and when needed
2. permanent link
3. virtual circuit … datagram
4. packet switched communication

Short Answer Questions

Q-1. How the drawback of circuit switching is overcome in message switching?

Ans: Message switching is based on store and forward technique. Instead of establishing a dedicated path, the message is sent to the nearest directly connected node. Each node stores the message, checks for error and forwards it. It allows more devices to share the network bandwidth and one message can be sent to several users. Destination host need not be on at the time of sending message.

Q-2. What is the drawback of message switching? How is it overcome in packet switching?

Ans.: In message switching, large storage space is required at each node to buffer the complete message blocks. On the other hand, in packet switching, messages are divided into subset of equal length, which are generated in the source node and reassembled to get back the initial complete message in destination node. Moreover, to transmit a message of large size, link is kept busy for a long time leading to increase in delay for other messages.

Q-3. What are the key differences between datagram and virtual-circuit packet switching?

Ans: In datagram, the packets are routed independently and it might follow different routes to reach the destination in different order. In virtual-circuit packet switching, first a virtual connection is being established, and all the packets are sent serially through the same path. In this case, packets are received in order.

Q-4. Distinguish between circuit switching and virtual-circuit packet switching.

Ans: - In circuit switching, a dedicated path is established. Data transmission is fast and interactive. Nodes need not have storage facility. However, there is a call setup delay. In overload condition, it may block the call setup. It has fixed bandwidth from source to destination and no overhead after the call setup.

In virtual-circuit packet switching, there is no dedicated path. It requires storage facility and involves packet transmission delay. It can use different speed of transmission and encoding techniques at different segments of the route.

Q-5. How packet size affects the transmission time in a packet switching network?

Ans: Initially, transmission time decreases as packet size is reduced. But, as packet size is reduced and the payload part of a packet becomes comparable to the control part, transmission time increases.

Special Instructional Objective

On completion of this lesson, the student will be able to:

- State the key features of X.25
- Explain the frame format of X.25
- Specify the function of the Packet layer of X.25
- State the limitations of X.25

4.4.1 Introduction

In the early 1970's there were many data communication networks (also known as Public Networks), which were owned by private companies, organizations and governments agencies. Since those public networks were quite different internally, and the interconnection of networks was growing very fast, there was a need for a common network interface protocol.

In 1976 X.25 was recommended as the desired protocol by the **International Consultative Committee for Telegraphy and Telephony** (CCITT) called the **International Telecommunication Union** (ITU) since 1993.

X.25 is a standard for WAN communications that defines how connections between user devices and network devices are established and maintained. X.25 is designed to operate effectively regardless of the type of systems connected to the network. It is typically used in the packet-switched networks (PSNs) of common carriers, such as the telephone companies. Subscribers are charged based on their use of the network.

4.4.2 X.25 Devices and Protocol Operation

X.25 network devices fall into three general categories: data terminal equipment (DTE), data circuit-terminating equipment (DCE), and packet-switching exchange (PSE) as shown in Fig. 4.4.1.

Data terminal equipment (DTE) devices are end systems that communicate across the X.25 network. They are usually terminals, personal computers, or network hosts, and are located on the premises of individual subscribers. Data communication Equipments (**DCEs)** are communications devices, such as modems and packet switches that provide the interface between DTE devices and a PSE, and are generally located in the carrier's facilities.

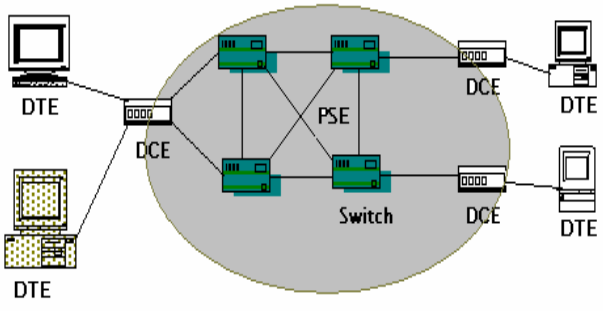


Figure 4.4.1 X.25 network

PSEs are switches that compose the bulk of the carrier's network. They transfer data from one DTE device to another through the X.25 PSN. Figure 4.4.1 illustrates the relationships among the three types of X.25 network devices

Packet Assembler/Disassembler

The *packet assembler/disassembler (PAD)* is a device commonly found in X.25 networks. PADs are used when a DTE device, such as a character-mode terminal, is too simple to implement the full X.25 functionality. The PAD is located between a DTE device and a DCE device, and it performs three primary functions: buffering (storing data until a device is ready to process it), packet assembly, and packet disassembly. The PAD buffers data sent to or from the DTE device. It also assembles outgoing data into packets and forwards them to the DCE device.

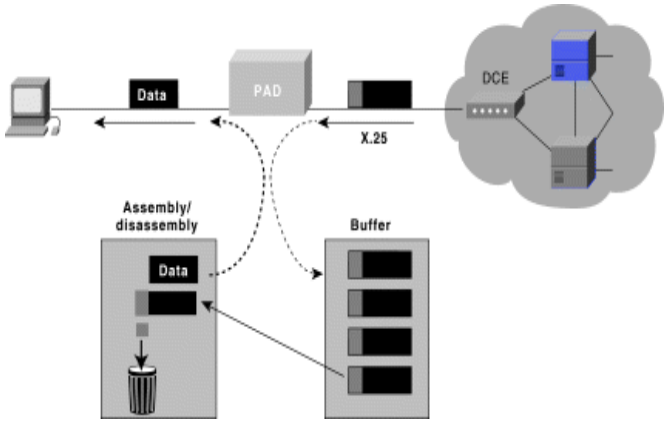


Figure 4.4.2 PADs

(This includes adding an X.25 header.) Finally, the PAD disassembles incoming packets before forwarding the data to the DTE. (This includes removing the X.25 header) Figure 4.4.2 illustrates the basic operation of the PAD when receiving packets from the X.25 WAN.

4.4.3 X.25 session establishment and virtual circuits

Session Establishment

X.25 sessions are established when one DTE device contacts another to request a communication session. It’s up to the receiving DTE whether to accept or refuse the connection. If the request is accepted, the two systems begin full-duplex communication. Either DTE device can terminate the connection. After the session is terminated, any further communication requires the establishment of a new session.

Virtual Circuits

The X.25 is a **packet-switched** virtual circuit network. A *virtual circuit* is a logical connection created to ensure reliable communication between two network devices. A virtual circuit denotes the existence of a logical, bidirectional path from one DTE device to another across an X.25 network. Physically, the connection can pass through any number of intermediate nodes, such as DCE devices and PSEs. Virtual circuits in X.25 are created at the network layer such that multiple virtual circuits (logical connections) can be multiplexed onto a single physical circuit (a physical connection). Virtual circuits are demultiplexed at the remote end, and data is sent to the appropriate destinations.

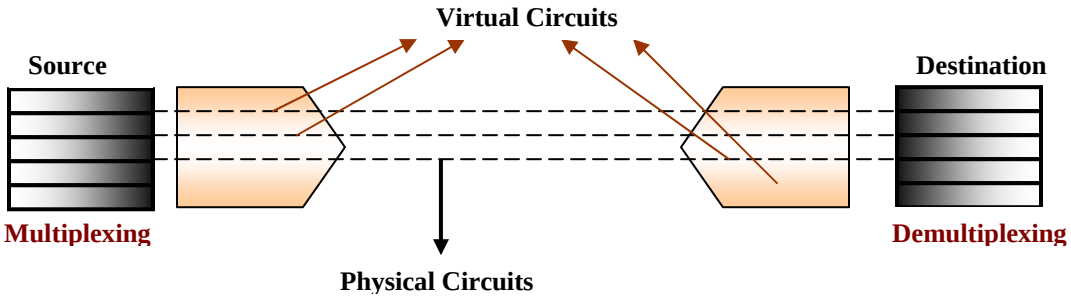


Figure 4.4.3 illustrates separate virtual circuits being multiplexed onto a single physical circuit.

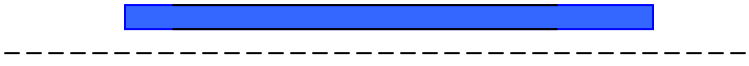


Figure 4.4.3 Physical Circuits and Virtual Circuit

Two types of X.25 virtual circuits exist: switched and permanent. *Switched virtual circuits (SVCs)* are temporary connections used for sporadic data transfers. They require that two DTE devices to establish, maintain, and terminate a session each time the devices need to communicate. *Permanent virtual circuits (PVCs)* are permanently established connections used for frequent and consistent data transfers. PVCs do not require that sessions be established and terminated. Therefore, DTEs can begin transferring data whenever necessary because the session is always active.

The basic operation of an X.25 virtual circuit begins when the source DTE device specifies the virtual circuit to be used (in the packet headers) and then sends the packets to a locally connected DCE device. At this point, the local DCE device examines the packet headers to determine which virtual circuit to use and then sends the packets to the closest PSE in the path of that virtual circuit. PSEs (switches) pass the traffic to the next intermediate node in the path, which may be another switch or the remote DCE device.

When the traffic arrives at the remote DCE device, the packet headers are examined and the destination address is determined. The packets are then sent to the destination DTE device. If communication occurs over an SVC and neither device has additional data to transfer, the virtual circuit is terminated.

4.4.4 X.25 Protocol Suite

The X.25 protocol suite maps to the lowest three layers of the OSI reference model as shown in Figure 4.4.4. The layers are:

- **Physical layer:** Deals with the physical interface between an attached station and the link that attaches that station to the packet-switching node.
    - X.21 is the most commonly used physical layer standard.
- **Frame layer:** Facilitates reliable transfer of data across the physical link by transmitting the data as a sequence of frames. Uses a subset of HDLC known as Link Access Protocol Balanced (LAPB), bit oriented protocol.
- **Packet layer:** Responsible for end-to-end connection between two DTEs. Functions performed are:
    - Establishing connection
    - Transferring data
    - Terminating a connection
    - Error and flow control
    - With the help of X.25 packet layer, data are transmitted in packets over external virtual circuits.

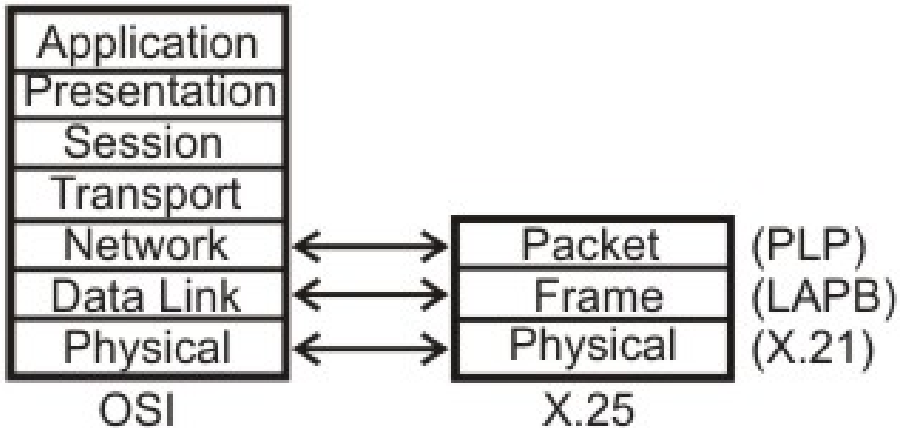


Figure 4.4.4 X.25 Layer mapping with OSI model

Physical Layer

At the physical layer X.21 is specifically defined for X.25 by ITU-T. The X.21 interface operates over eight interchange circuits (i.e., signal ground, DTE common return, transmit, receive, control, indication, signal element timing and byte timing) their functions is defined in recommendation of X.24 and their electrical characteristics in recommendation of X.27. The recommendation specifies how the DTE can setup and clear calls by exchanging signals with the DCE.

The physical connector has 15 pins, but not all of them are used. The DTE uses the **T** and **C** circuits to transmit data and control information. The DCE uses the **R** and **I** circuits for data and control. The **S** circuit contains a signal stream emitted by the DCE to provide timing information so the DTE knows when each bit interval starts and stops. The **B** circuit may also provide to group the bits into byte frames. If this option is not provided the DCE and DTE must begin every control

sequence with at least two SYN characters to enable each other to deduce the implied frame boundary.

Line	Name	From DTE	From DCE
G	Signal ground		
Ga	DTE Common return	**X**	
T	Transmit	**X**	**X**
R	Receive		**X**
C	Control	**X**	
I	Indication		**X**
S	Signal element timing		**X**
B	Byte Timing		**X**

Figure 4.4.5 X.21 signals

Link Layer

The link layer (also called level 2, or frame level) ensures reliable transfer of data between the DTE and the DCE, by transmitting the data as a sequence of frames (a frame is an individual data unit which contains address, control, information field etc.).

The functions performed by the link level include:

- Transfer of data in an efficient and timely fashion.
- Synchronization of the link to ensure that the receiver is in step with the transmitter.
- Detection of transmission errors and recovery from such errors
- Identification and reporting of procedural errors to higher levels, for recovery.

The link level uses data link control procedures, which are compatible with the High Level Data Link (HDLC) standardized by ISO, and with the Advanced Data Communications Control Procedures (ADCCP) standardized by the U.S. American National Standards Institute (ANSI).

There are several protocols, which can be used in the link level:

- **Link Access Protocol, Balanced (LAPB)** is derived from HDLC and is the most commonly used. It enables to form a logical link connection besides all the other characteristics of HDLC.
- **Link Access Protocol (LAP)** is an earlier version of LAPB and is seldom used today.
- **Link Access Procedure, D Channel (LAPD)** is derived from LAPB and it is used for Integrated Services Digital Networks (ISDN) i.e. it enables data transmission between DTEs through D channel, especially between a DTE and an ISDN node.
- **Logical Link Control (LLC)** is an IEEE 802 Local Area Network (LAN) protocol, which enables X.25 packets to be transmitted through a LAN channel.

Now let us discuss the most commonly used link layer protocol, i.e. LAPB. LAPB is a bit-oriented protocol that ensures that frames are correctly ordered and error-free. There are three kinds of frames:

1. **Information:** This kind of frame contains the actual information being transferred and some control information. The control field in these frames contains the frame sequence number. I-frame functions include sequencing, flow control, and error detection and recovery. I-frames carry send- and receive-sequence numbers.
2. **Supervisory:** The supervisory frame (S-frame) carries control information. S-frame functions include requesting and suspending transmissions, reporting on status, and acknowledging the receipt of I-frames. S-frames carry only receive-sequence numbers. There are various types of supervisory frames.
    - RECEIVE READY-Acknowledgment frame indicating the next frame expected.
    - REJECT-Negative acknowledgment frame used to indicate transmission error detection.
    - RECEIVE NOT READY (RNR)-Just as RECEIVE READY but tells the sender to stop sending due to temporary problems.
3. **Unnumbered:** This kind of frames is used only for control purposes. U-frame functions include link setup and disconnection, as well as error reporting. U frames carry no sequence numbers.

Packet Level

This level governs the end-to-end communications between the different DTE devices. Layer 3 is concerned with connection set-up and teardown and flow control between the DTE devices, as well as network routing functions and the multiplexing of simultaneous logical connections over a single physical connection. PLP is the network layer protocol of X.25.

Call setup mode is used to establish SVCs between DTE devices. A PLP uses the X.121 addressing scheme to set up the virtual circuit. The call setup mode is executed on a per-virtual-circuit basis, which means that one virtual circuit can be in call setup mode while another is in data transfer mode. This mode is used only with SVCs, not with PVCs. To establish a connection on an SVC, the calling DTE sends a **Call Request** Packet, which includes the address of the remote DTE to be contacted. The destination DTE decides whether or not to accept the call (the Call Request packet includes the sender's DTE address, as well as other information that the called DTE can use to decide whether or not to accept the call). A call is accepted by issuing a **Call Accepted** packet, or cleared by issuing a **Clear Request** packet. Once the originating DTE receives the Call Accepted packet, the virtual circuit is established and data transfer may take place.

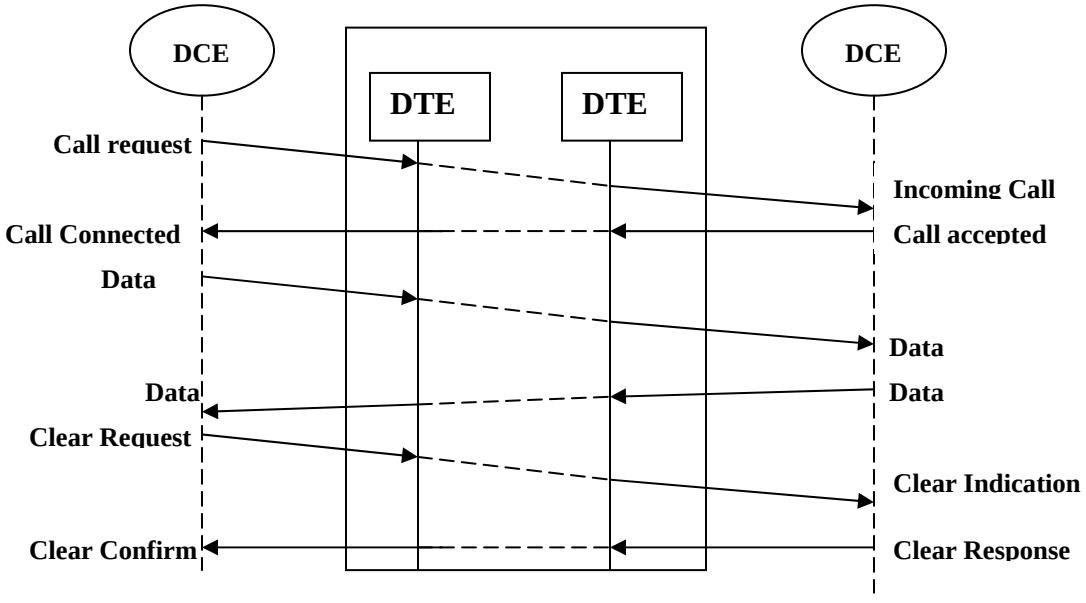


Figure 4.4.6 Different Modes of PLP

Different phases of call set-up, data transfer, call release has been shown in Fig. 4.4.6. The PLP operates in five distinct modes: call setup, data transfer, idle, call clearing, and restarting.

- **Data transfer** mode is used for transferring data between two DTE devices across a virtual circuit. In this mode, PLP handles segmentation and reassembly, bit padding, and error and flow control. This mode is executed on a per-virtual-circuit basis and is used with both PVCs and SVCs.
- **Idle mode** is used when a virtual circuit is established but data transfer is not occurring. It is executed on a per-virtual-circuit basis and is used only with SVCs.
- **Call clearing mode** is used to end communication sessions between DTE devices and to terminate SVCs. This mode is executed on a per-virtual-circuit basis and is used only with SVCs. When either DTE wishes to terminate the call, a **Clear Request** packet is sent to the remote DTE, which responds with a **Clear Confirmation** packet.
- **Restarting mode** is used to synchronize transmission between a DTE device and a locally connected DCE device. This mode is not executed on a per-virtual-circuit basis. It affects all the DTE device's established virtual circuits.

Four types of PLP packet fields exist:

- **General Format Identifier (GFI)**—Identifies packet parameters, such as whether the packet carries user data or control information, what kind of windowing is being used, and whether delivery confirmation is required.
- **Logical Channel Identifier (LCI)**—identifies the virtual circuit across the local DTE/DCE interface.
- **Packet Type Identifier (PTI)**—identifies the packet as one of 17 different PLP packet types.

- **User Data**—Contains encapsulated upper-layer information. This field is present only in data packets. Otherwise, additional fields containing control information are added.

Fill in the Blank:

1. *X.25* is a standard for _____communications.
2. X.25 protocol uses _____ for end-to-end transmission.
3. X.25 operates in _____ layer of OSI.
4. _____ devices are end systems that communicate across the X.25 network.
5. Two types of X.25 virtual circuits exist: _____ and _____.
6. At the physical layer _____ is specifically defined for X.25 by ITU-T.
7. The link level ensures reliable transfer of data between the _____ and the ____.
8. The _____ frame carries control information.
9. _____frame contains the actual information being transferred
10. The PLP Packet is a product of _____ layer in X.25 standard.
11. The PLP _____ is used to transport data from the upper layers in X.25 standard.

Short Answers Questions:

1. In what layers X.25 operates?
Ans: X.25 operates in the network layer.

2. What are the key functions of X.25 protocol?
Ans: Key functions of X.25 protocol are:

i) Call control packets are used for call set-up.
ii) Multiplexing of virtual circuits take place in packet layer.
iii) Both link layer and packet layer performs flow control and error control.

3. What limitation of X.25 is overcome in Frame Relay Protocol?
Ans: In X.25, overhead on the user equipment and the networking equipment is very high and it is also slower (can go up to 64 kbps only), which are overcome in Frame Control Protocol.

4. Explain the functionalities of DTE, DCE, PSE.
Ans: Data terminal equipment devices are end systems that communicate across the X.25 network. They are usually terminals, personal computers, or network hosts, and are located on the premises of individual subscribers. **DCE devices** are communications devices, such as modems and packet switches that provide the interface between DTE devices and a PSE, and are generally located in the carrier's facilities. **PSEs** are switches that compose the bulk of the carrier's network. They transfer data from one DTE device to another through the X.25 PSN

5. Describe the functionalities of Link Level.

Ans: The functions performed by the link level include:

- Transfer of data in an efficient and timely fashion.
- Synchronization of the link to ensure that the receiver is in step with the transmitter.
- Detection of transmission errors and recovery from such errors
- Identification and reporting of procedural errors to higher levels, for recovery.

6. What protocols can be used in Link Level?

Ans: There are several protocols which can be used in the link level:

- **Link Access Protocol, Balanced (LAPB)** is derived from HDLC and is the most commonly used. It enables to form a logical link connection besides all the other characteristics of HDLC.
- **Link Access Protocol (LAP)** is an earlier version of LAPB and is seldom used today.
- **Link Access Procedure, D Channel (LAPD)** is derived from LAPB and it is used for Integrated Services Digital Networks (ISDN) i.e. it enables data transmission between DTEs through D channel, especially between a DTE and an ISDN node.
- **Logical Link Control (LLC)** is an IEEE 802 Local Area Network (LAN) protocol which enables X.25 packets to be transmitted through a LAN channel.

7. Explain the different level of operation of PLP.

Ans: The PLP operates in five distinct modes: call setup, data transfer, idle, call clearing, and restarting.

- **Call setup mode** is used to establish SVCs between DTE devices. A PLP uses the X.121 addressing scheme to set up the virtual circuit. The call setup mode is executed on a per-virtual-circuit basis.
- **Data transfer** mode is used for transferring data between two DTE devices across a virtual circuit. In this mode, PLP handles segmentation and reassembly, bit padding, and error and flow control.
- **Idle mode** is used when a virtual circuit is established but data transfer is not occurring
- **Call clearing mode** is used to end communication sessions between DTE devices and to terminate SVCs. This mode is executed on a per-virtual-circuit basis and is used only with SVCs.
- **Restarting mode** is used to synchronize transmission between a DTE device and a locally connected DCE device. This mode is not executed on a per-virtual-circuit basis. It affects all the DTE device's established virtual circuits.

Special Instructional Objective

- On completion of this lesson, the student will be able to:
- State the limitations of X.25
- Explain the key features of Frame Relay
- Specify the Frame relay frame format
- Explain how congestion control is performed in Frame relay network

4.5.1 Introduction

Frame Relay is a high-performance WAN protocol that operates at the physical and data link layers of the OSI reference model. Frame Relay originally was designed for use across Integrated Services Digital Network (ISDN) interfaces. Today, it is used over a variety of other network interfaces as well. Frame Relay is a simplified form of Packet Switching, similar in principle to X.25, in which synchronous frames of data are routed to different destinations depending on header information. The biggest difference between Frame Relay and X.25 is that X.25 guarantees data integrity and network managed flow control at the cost of some network delays. Frame Relay switches packets end to end much faster, but there is no guarantee of data integrity at all.

As line speeds have increased from speeds below 64kbps to T1/E1 and beyond, the delays inherent in the store-and-forward mechanisms of X.25 become intolerable. At the same time, improvements in digital transmission techniques have reduced line errors to the extent that node-to-node error correction throughout the network is no longer necessary. The vast majority of Frame Relay traffic consists of TCP/IP or other protocols that provide their own flow control and error correction mechanisms. Much of this traffic is fed into the Internet, another packet switched network without any built-in error control.

Because Frame Relay does not 'care' whether the frame it is switching is error-free or not, a Frame Relay node can start switching traffic out onto a new line as soon as it has read the first two bytes of addressing information at the beginning of the frame. Thus a frame of data can travel end-to-end, passing through several switches, and still arrive at its destination with only a few bytes' delay. These delays are small enough that network latency under Frame Relay is not noticeably different from direct leased line connections. As a result, the performance of a Frame Relay network is virtually identical to that of a leased line, but because most of the network is shared, costs are lower.

Frame Relay is an example of a packet-switched technology. Packet-switched networks enable end stations to dynamically share the network medium and the available bandwidth. The following two techniques are used in packet-switching technology:

- Variable-length packets
- Statistical multiplexing

Variable-length packets are used for more efficient and flexible data transfers. These packets are switched between the various segments in the network until the destination is reached.

Statistical multiplexing techniques control network access in a packet-switched network. The advantage of this technique is that it accommodates more flexibility and more efficient use of bandwidth. Most of today's popular LANs, such as Ethernet and Token Ring, are packet-switched networks.

4.5.2 Frame Relay Devices

Devices attached to a Frame Relay WAN fall into the following two general categories:

- Data terminal equipment (DTE)
- Data circuit-terminating equipment (DCE)

DTEs generally are considered to be terminating equipment for a specific network and typically are located on the premises of a customer. In fact, they may be owned by the customer. Examples of DTE devices are terminals, personal computers, routers, and bridges.

DCEs are carrier-owned internetworking devices. The purpose of DCE equipment is to provide clocking and switching services in a network, which are the devices that actually transmit data through the WAN. In most cases, these are packet switches.

The connection between a DTE device and a DCE device consists of both a physical layer component and a link layer component. The physical component defines the mechanical, electrical, functional, and procedural specifications for the connection between the devices. One of the most commonly used physical layer interface specifications is the recommended standard (RS)-232 specification. The link layer component defines the protocol that establishes the connection between the DTE device, such as a router, and the DCE device, such as a switch.

4.5.3 Virtual Circuits

Frame Relay is a virtual circuit network, so it doesn’t use physical addresses to define the DTEs connected to the network. Frame Relay provides connection-oriented data link layer communication. This means that a defined communication exists between each pair of devices and that these connections are associated with a connection identifier. However, virtual circuit identifiers in Frame relay operate at the data link layer, in contrast with X.25, where they operate at the network layer. This service is implemented by using a Frame Relay virtual circuit, which is a logical connection created between two data terminal equipment (DTE) devices across a Frame Relay packet-switched network (PSN).

Virtual circuits provide a bidirectional communication path from one DTE device to another and are uniquely identified by a data-link connection identifier (DLCI). A

number of virtual circuits can be multiplexed into a single physical circuit for transmission across the network. This capability often can reduce the equipment and network complexity required to connect multiple DTE devices.

A virtual circuit can pass through any number of intermediate DCE devices (switches) located within the Frame Relay PSN. Before going into the details of DLCI let us first have a look at the two types of Frame Relay Circuits, namely: switched virtual circuits (SVCs) and permanent virtual circuits (PVCs).

4.5.3.1 Switched Virtual Circuits

Switched virtual circuits (SVCs) are temporary connections used in situations requiring only sporadic data transfer between DTE devices across the Frame Relay network. A communication session across an SVC consists of the following four operational states:

- **Call setup**—The virtual circuit between two Frame Relay DTE devices is established.
- **Data transfer**—Data is transmitted between the DTE devices over the virtual circuit.
- **Idle**—The connection between DTE devices is still active, but no data is transferred. If an SVC remains in an idle state for a defined period of time, the call can be terminated.
- **Call termination**—The virtual circuit between DTE devices is terminated.

After the virtual circuit is terminated, the DTE devices must establish a new SVC if there is additional data to be exchanged. It is expected that SVCs will be established, maintained, and terminated using the same signaling protocols used in ISDN.

4.5.3.2 Permanent Virtual Circuits

Permanent virtual circuits (PVCs) are permanently established connections that are used for frequent and consistent data transfers between DTE devices across the Frame Relay network. Communication across PVC does not require the call setup and termination states that are used with SVCs. PVCs always operate in one of the following two operational states:

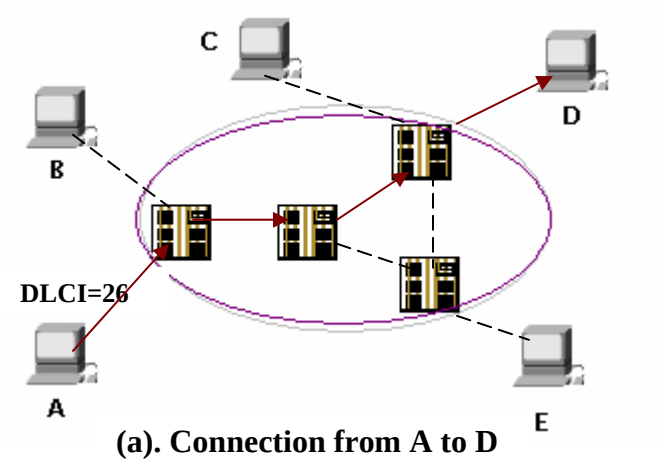
- **Data transfer**: Data is transmitted between the DTE devices over the virtual circuit.
- **Idle**: The connection between DTE devices is active, but no data is transferred.

Unlike SVCs, PVCs will not be terminated under any circumstances when in an idle state. DTE devices can begin transferring data whenever they are ready because the circuit is permanently established.

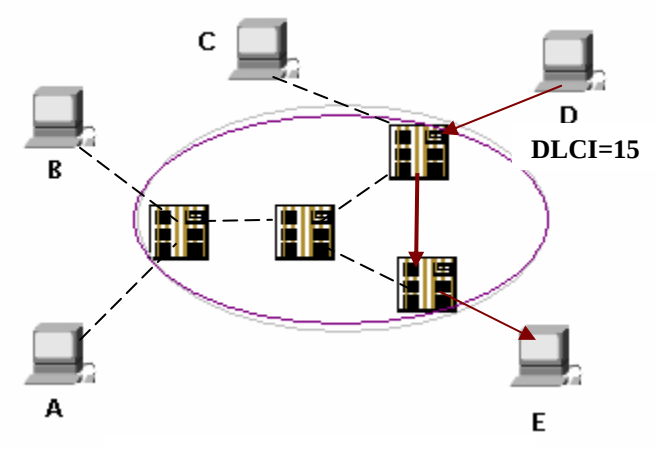
4.5.3.3 Data-Link Connection Identifier (DLCI)

Frame Relay virtual circuits are identified by *data-link connection identifiers (DLCIs)*. DLCI values typically are assigned by the Frame Relay service provider (for example, the

telephone company). Frame Relay DLCIs have local significance, which means that their values are unique in the LAN, but not necessarily in the Frame Relay WAN. The local DTEs use this DLCI to send frames to the remote DTE.

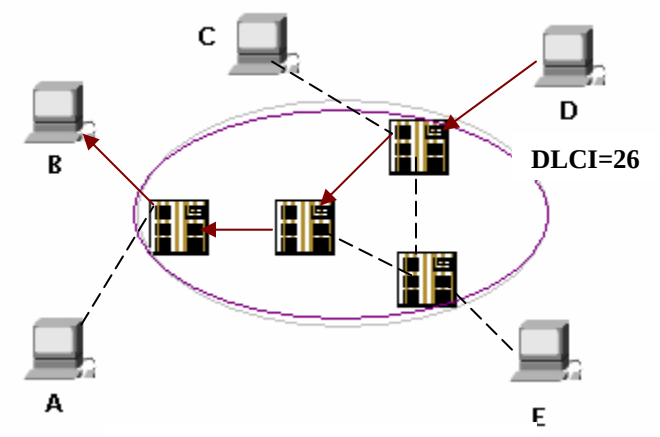


(a). Connection from A to D



(b). Connection from D to E

Figure 4.5.1 shows the assignments of DLCIs, 3 connections have been shown namely, between A to D, D to E, and D to B. Point Here to be noted is that connection in fig.(a) and in fig.(c) uses the same DLCI, this can be done because these DLCIs are used by local DTEs. Bothe the connectons are valid as they define different virtual circuits originating from different DTEs.

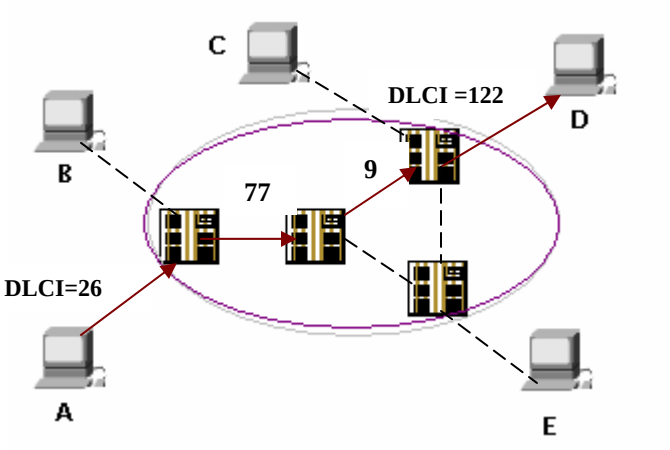


(c). Connection from D to A

Figure 4.5.1 DLCIs connection between different DTEs

4.5.3.4 DLCIs inside the network

DLCIs are not only used to define the virtual circuit between a DTE and a DCE, but also to define the virtual circuit between two DCEs (switches) inside the network. A switch assigns a DLCI to each virtual connection in an interface. This means that two different connections belonging to two different interfaces may have the same DLCIs (as shown in the above figure). In other words, DLCIs are unique for a particular interface.



A connection between DTE A and DTE D has been shown in this figure, DLCI assigned inside the Frame Relay network is also shown in the network. DCEs inside the network use incoming interface – DLCI combination to decide the outgoing interface – DLCI combination to switch out the frame, from that DCE.

Figure 4.5.2 DLCIs inside Frame relay network

Each switch in a Frame relay network has a table to route frames The table matches the incoming interface- DLCI combination with an outgoing interface-DLCI combination. Figure 4.5.3 shows two frames arriving at the switch on interface2, one with DLCI=11 and other with DLCI= 213.

Incoming		**Outgoing**	
Interface	DLCI	Interface	DLCI
1	122	1	41
2	213	1	158
2	11	2	111
1	151	2	210

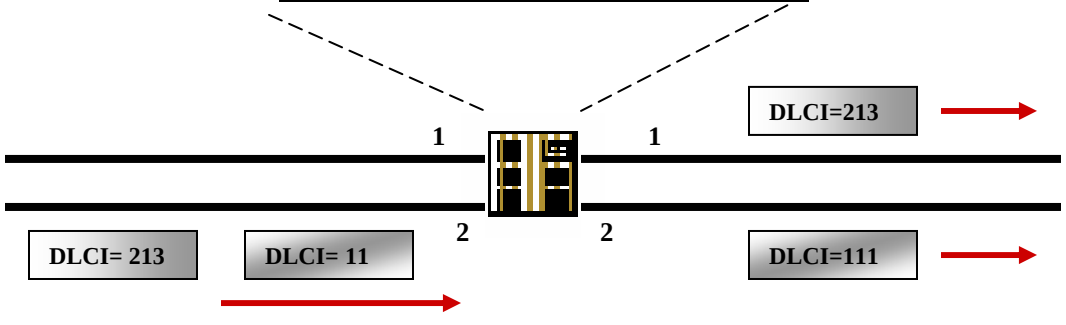


Figure 4.5.3 Frame Relay switch table

4.5.4 Frame Relay Layers

Frame Relay has only 2 layers, namely Physical layer and Data Link layer. And as compared to other layer of packet switching network such as X.25, frame relay has only 1.5 layers whereas X.25 has 2 layers. Frame Relay eliminates all network layer functions and a portion of conventional data-link layer functions.

Physical Layer

No specific protocol is defined for physical layer in frame relay. Frame relay supports any one of the protocols recognized by ANSI, and thus the choice of physical layer protocol is up to the implementer.

Data Link Layer

At Data-link Layer Frame employs a simpler version of HDLC. Simpler version is used because HDLC provides extensive error and flow control fields that are not needed in frame relay.

To understand much of the functionality of Frame Relay, it is helpful to understand the structure of the Frame Relay frame. Figure 4.5.4 depicts the basic format of the Frame Relay frame. Flags indicate the beginning and end of the frame. Three primary components make up the Frame Relay frame: the header and address area, the user-data portion, and the frame check sequence (FCS). The address area, which is 2 bytes in length, is comprised of 10 bits representing the actual circuit identifier and 6 bits of fields related to congestion management. This identifier commonly is referred to as the data-link connection identifier (DLCI).

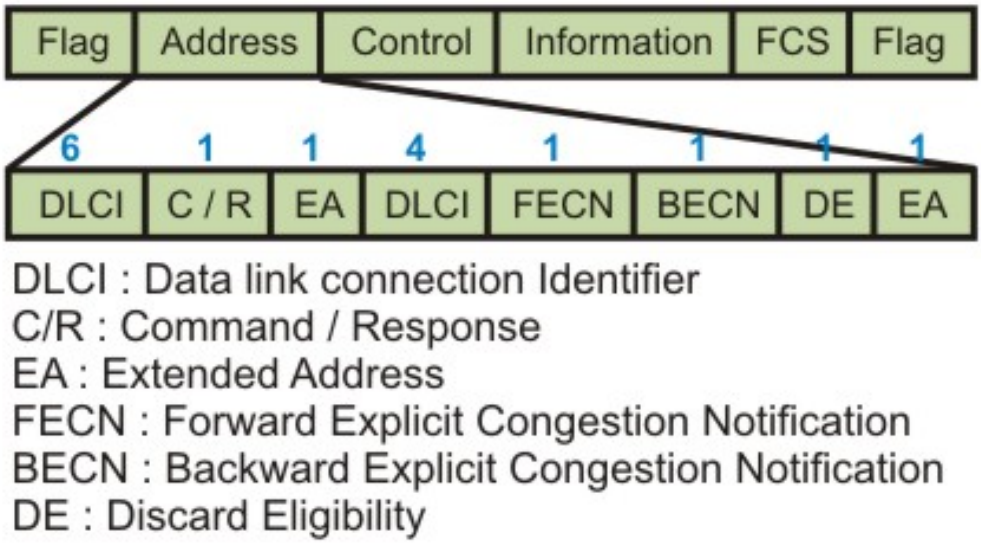


Figure 4.5.4 Frame Relay frame format

- F**lags**—Delimits the beginning and end of the frame. The value of this field is always the same and is represented either as the hexadecimal number 7E or as the binary number 01111110.
- **Address**—Contains the following information:

DLCI—The 10-bit DLCI is the essence of the Frame Relay header. This value represents the virtual connection between the DTE device and the switch. Each virtual connection

that is multiplexed onto the physical channel will be represented by a unique DLCI. The DLCI values have local significance only, which means that they are unique only to the physical channel on which they reside. Therefore, devices at opposite ends of a connection can use different DLCI values to refer to the same virtual connection. The first 6-bits of the first byte make up part 1 of the DLCI, and second part of DLCI uses the first 4-bits of second byte.

E**xtended Address (EA)**—The EA is used to indicate whether the byte in which the EA value is 1 is the last addressing field. If the value is 1, then the current byte is determined to be the last DLCI octet. Although current Frame Relay implementations all use a two-octet DLCI, this capability does allow longer DLCIs to be used in the future. The eighth bit of each byte of the Address field is used to indicate the EA.

C/R—The C/R is the bit that follows the most significant DLCI byte in the Address field. The C/R bit is not currently defined.

Congestion Control—This consists of the 3 bits that control the Frame Relay congestion-notification mechanisms. These are the FECN, BECN, and DE bits, which are the last 3 bits in the Address field.

Forward-explicit congestion notification (FECN) is a single-bit field that can be set to a value of 1 by a switch to indicate to an end DTE device, such as a router, that congestion was experienced in the direction of the frame transmission from source to destination as shown in Fig. 4.5.5. The primary benefit of the use of the FECN and BECN fields is the capability of higher-layer protocols to react intelligently to these congestion indicators. Today, DECnet and OSI are the only higher-layer protocols that implement these capabilities.

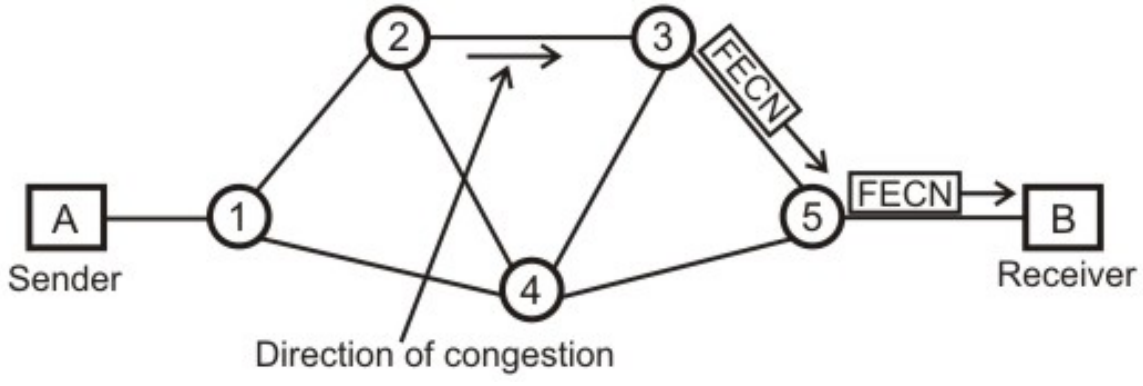


Figure 4.5.5 Forward-explicit congestion notification

Backward-explicit congestion notification (BECN) is a single-bit field that, when set to a value of 1 by a switch, indicates that congestion was experienced in the network in the direction opposite of the frame transmission from source to destination.

Discard eligibility (DE) is set by the DTE device, such as a router, to indicate that the marked frame is of lesser importance relative to other frames being transmitted. Frames

that are marked as "discard eligible" should be discarded before other frames in a congested network. This allows for a basic prioritization mechanism in Frame Relay networks.

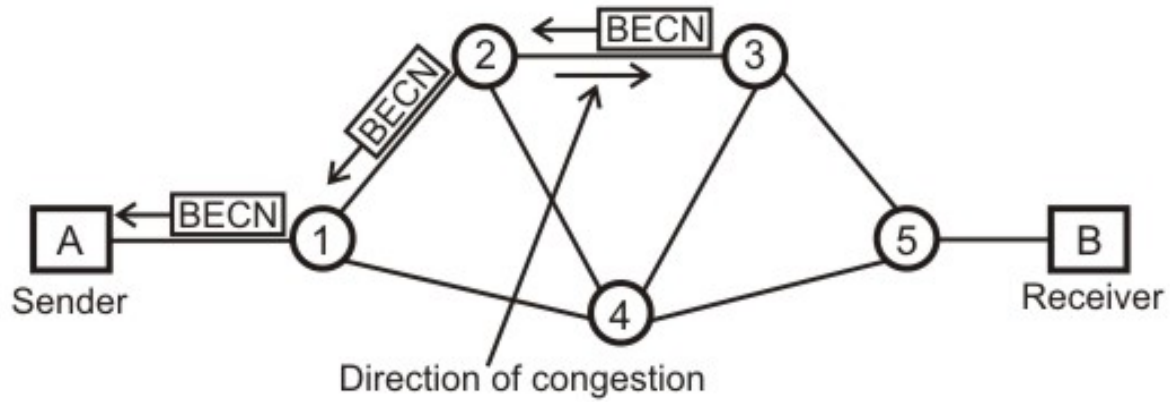


Figure 4.5.6 Backward-explicit congestion notification

• **Data**—Contains encapsulated upper-layer data. Each frame in this variable-length field includes a user data or payload field that will vary in length up to 16,000 octets. This field serves to transport the higher-layer protocol packet (PDU) through a Frame Relay network.

• **Frame Check Sequence**—Ensures the integrity of transmitted data. This value is computed by the source device and verified by the receiver to ensure integrity of transmission.

4.5.5 Summary

- Frame relay operates only in data link and physical layer.
- Frame Relay allows bursty traffic.
- It allows frame size of 9000 bytes, which can accommodate all local area network frames.
- Frame relay is less expensive then other traditional WANs.
- Frame relay provides both Permanent and switched connections.
- Frame relay allow variable-length frames, this may create varying delays for different users. Due to variable delay it is not suitable for real-time communication

Fill in the blanks:

1. Frame Relay is a high-performance _____ protocol.
2. Frame Relay operates at the _____ and _____ layers of the OSI reference model.
3. Frame Relay requires Error Checking at the _____ layer.
4. Frame Relay is a simplified form of _____ switching, similar in principle to _____.
5. Frame Relay is a _____ network.
6. Frame Relay virtual circuits are identified by _____.
7. _____ bit in address field in frame relay is set to one to signify the last address bit.
8. Routing and switching in Frame Relay is performed by _____ layer.
9. _____ data are allowed on a Frame Relay Network.
10. Frame relay is not suited well for _____ due to the delay resulting from varying sizes of Frame.

Answers fill in the blanks

1. WAN
2. Physical, data link
3. Data link
4. Circuit, X.25
5. Virtual switched
6. DLCIs.
7. Extended Address (EA)
8. Data link layer
9. Encapsulated upper layer
10. Real time traffic

Short Answer Questions:

1. Explain few devices used in Frame relay.

Ans: Devices attached to a Frame Relay WAN fall into the following two general categories:

- **Data terminal equipment (DTE)** : DTEs generally are considered to be terminating equipment for a specific network and typically are located on the premises of a customer
- **Data circuit-terminating equipment (DCE)**: DCEs are carrier-owned internetworking devices. The purpose of DCE equipment is to provide clocking and switching services in a network, which are the devices that actually transmit data through the WAN. In most cases, these are packet switches.

2. Distinguish between permanent virtual and switched virtual connections used in Frame relay protocol.

Ans: In permanent virtual connection, the path is fixed and data transfer occurs as with virtual calls, but no call setup or termination is required. On the other hand, in switched virtual connection, the path is a dynamically established virtual circuit using a call set up and call clearing procedure. Many other circuits can share the same path.

3. What are the various states in a Switched virtual circuit connection in Frame Relay?

Ans:

A communication session across an SVC consists of the following four operational states:

- **Call setup**—The virtual circuit between two Frame Relay DTE devices is established.
- **Data transfer**—Data is transmitted between the DTE devices over the virtual circuit.
- **Idle**—The connection between DTE devices is still active, but no data is transferred. If an SVC remains in an idle state for a defined period of time, the call can be terminated.
- **Call termination**—The virtual circuit between DTE devices is terminated.

4. Describe Permanent Virtual switched connection in Frame Relay.

Ans: *Permanent virtual circuits (PVCs)* are permanently established connections that are used for frequent and consistent data transfers between DTE devices across the Frame Relay network. Communication across PVC does not require the call setup and termination states that are used with SVCs. PVCs always operate in one of the following two operational states:

- **Data transfer**—Data is transmitted between the DTE devices over the virtual circuit.

- **Idle**—The connection between DTE devices is active, but no data is transferred.

Unlike SVCs, PVCs will not be terminated under any circumstances when in an idle state. DTE devices can begin transferring data whenever they are ready because the circuit is permanently established.

5. Write a short Note on Data-Link Connection Identifier (DLCI).

Ans: Frame Relay virtual circuits are identified by *data-link connection identifiers (DLCIs)*. DLCI values typically are assigned by the Frame Relay service provider (for example, the telephone company). Frame Relay DLCIs have local significance, which means that their values are unique in the LAN, but not necessarily in the Frame Relay WAN. The local DTEs use this DLCI to send frames to the remote DTE.

DLCIs are not only used to define the virtual circuit between a DTE and a DCE, but also to define the virtual circuit between two DCEs (switches) inside the network. A switch assigns a DLCI to each virtual connection in an interface. This means that two different connections belonging to two different interfaces may have the same DLCIs. In other words, DLCIs are unique for a particular interface.

6. What does extended address field in Frame Relay frame Format specifies?

Ans:
Extended Address (EA) is used to indicate whether the byte in which the EA value is 1 is the last addressing field. If the value is 1, then the current byte is determined to be the last DLCI octet. Although current Frame Relay implementations all use a two-octet DLCI, this capability does allow longer DLCIs to be used in the future. The eighth bit of each byte of the Address field is used to indicate the EA.