

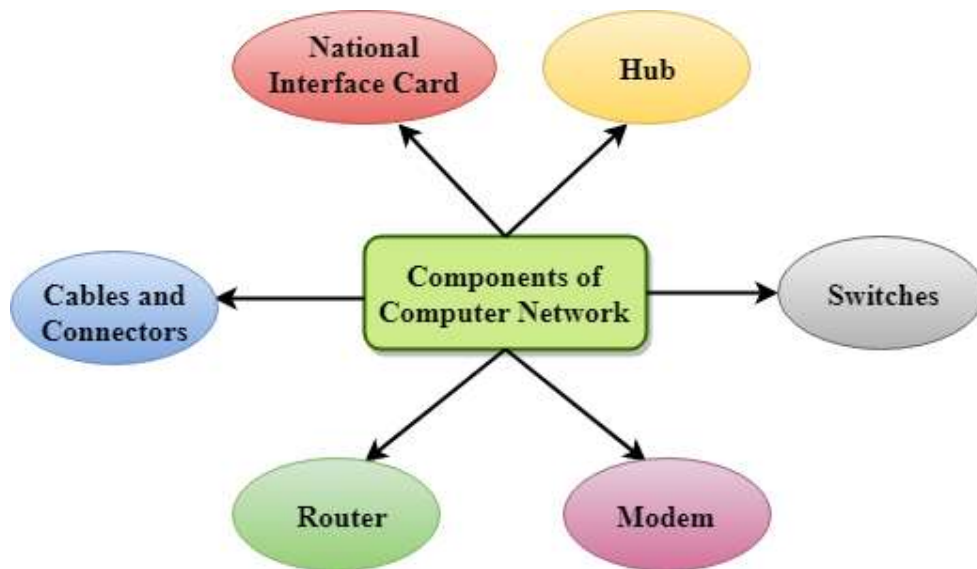
# COMPUTER NETWORKS

## UNIT I

### Computer Networks:

- **Computer Network** is a group of computers connected with each other through wires, optical fibres or optical links so that various devices can interact with each other through a network.
- The aim of the computer network is the sharing of resources among various devices.
- In the case of computer network technology, there are several types of networks that vary from simple to complex level.

### Components of Computer Network:



### Uses of Computer Network

#### Business Applications:

- **Resource sharing:** Resource sharing is the sharing of resources such as programs, printers, and data among the users on the network without the requirement of the physical location of the resource and user.
- **Server-Client model:** Computer networking is used in the **server-client model**. A server is a central computer used to store the information and maintained by the system administrator. Clients are the machines used to access the information stored in the server remotely.
- **Communication medium:** Computer network behaves as a communication medium among the users. For example, a company contains more than one computer has an email system which the employees use for daily communication.
- **E-commerce:** Computer network is also important in businesses. We can do the business over the internet. For example, amazon.com is doing their business over the internet, i.e., they are doing their business over the internet.

### Home Applications:

Some of the most important uses of the Internet for home users are as follows:

- Access to remote information
- Person-to-person communication
- Interactive entertainment
- Electronic commerce

### Mobile Users:

Mobile computers, such as notebook computers and Mobile phones, is one of the fastest-growing segment of the entire computer industry. Although wireless networking and mobile computing are often related, they are not identical, as the below figure shows.

Wireless	Mmobile	Applications
No	No	Desktop computers in offices
No	Yes	A notebook computer used in a hotel room
Yes	No	Networks in older, unwired buildings
Yes	Yes	Portable office; PDA for store inventory

### Social Issues:

- Spamming.
- Privacy.
- Notifications.
- Access to information.
- Impact on employability.
- Potential for misuse.
- Unauthorized access.
- Risk for child safety, etc.

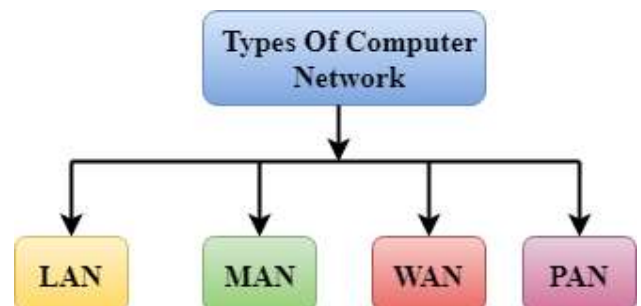
### NETWORK HARDWARE:

A computer network is a group of computers linked to each other that enables the computer to communicate with another computer and share their resources, data, and applications.

A computer network can be categorized by their size.

A **computer network** is mainly of **four types**:

- LAN(Local Area Network)
- PAN(Personal Area Network)
- MAN(Metropolitan Area Network)
- WAN(Wide Area Network)



### LAN(Local Area Network)

- Local Area Network is a group of computers connected to each other in a small area such as building, office.
- LAN is used for connecting two or more personal computers through a communication medium such as twisted pair, coaxial cable, etc.
- It is less costly as it is built with inexpensive hardware such as hubs, network adapters, and ethernet cables.
- The data is transferred at an extremely faster rate in Local Area Network.
- Local Area Network provides higher security.



### PAN(Personal Area Network)

- Personal Area Network is a network arranged within an individual person, typically within a range of 10 meters.
- Personal Area Network is used for connecting the computer devices of personal use is known as Personal Area Network.
- **Thomas Zimmerman** was the first research scientist to bring the idea of the Personal Area Network.
- Personal Area Network covers an area of **30 feet**.
- Personal computer devices that are used to develop the personal area network are the laptop, mobile phones, media player and play stations.

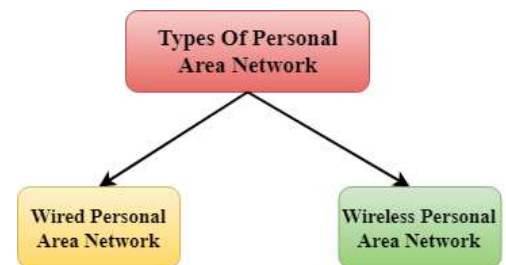


### **There are two types of Personal Area Network:**

- Wired Personal Area Network
- Wireless Personal Area Network

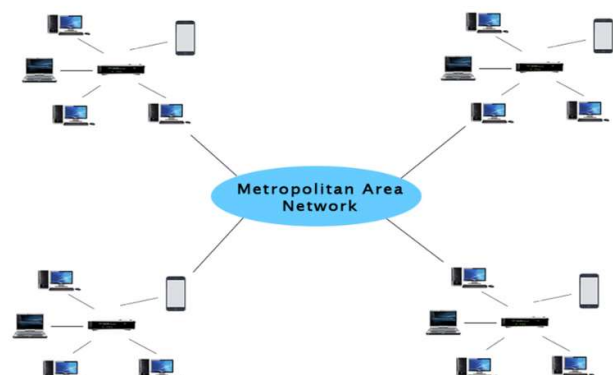
**Wireless Personal Area Network:** Wireless Personal Area Network is developed by simply using wireless technologies such as WiFi, Bluetooth. It is a low range network.

**Wired Personal Area Network:** Wired Personal Area Network is created by using the USB.



### MAN(Metropolitan Area Network)

- A metropolitan area network is a network that covers a larger geographic area by interconnecting a different LAN to form a larger network.
- Government agencies use MAN to connect to the citizens and private industries.
- In MAN, various LANs are connected to each other through a telephone exchange line.
- The most widely used protocols in MAN are RS-232, Frame Relay, ATM, ISDN, OC-3, ADSL, etc.



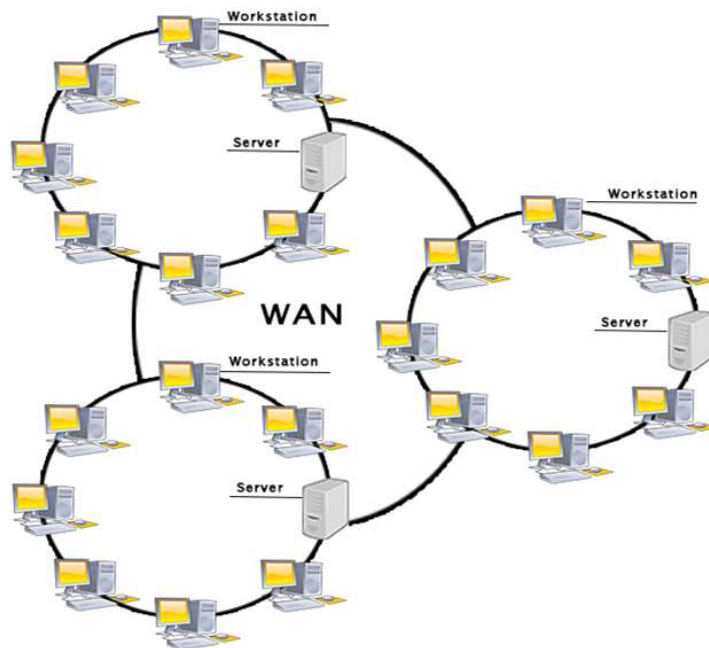
- It has a higher range than Local Area Network(LAN).

#### Uses Of Metropolitan Area Network:

- MAN is used in communication between the banks in a city.
- It can be used in an Airline Reservation.
- It can be used in a college within a city.
- It can also be used for communication in the military.

#### WAN(Wide Area Network)

- A Wide Area Network is a network that extends over a large geographical area such as states or countries.
- A Wide Area Network is quite bigger network than the LAN.
- A Wide Area Network is not limited to a single location, but it spans over a large geographical area through a telephone line, fibre optic cable or satellite links.
- The internet is one of the biggest WAN in the world.
- A Wide Area Network is widely used in the field of Business, government, and education.



#### Network Software:

A **protocol** is simply defined as a set of rules and regulations for data communication. Rules are basically defined for each and every step and process at time of communication among two or more computers. Networks are needed to follow these protocols to transmit data successfully. All protocols might be implemented using hardware, software, or combination of both of them. There are three aspects of protocols given below:

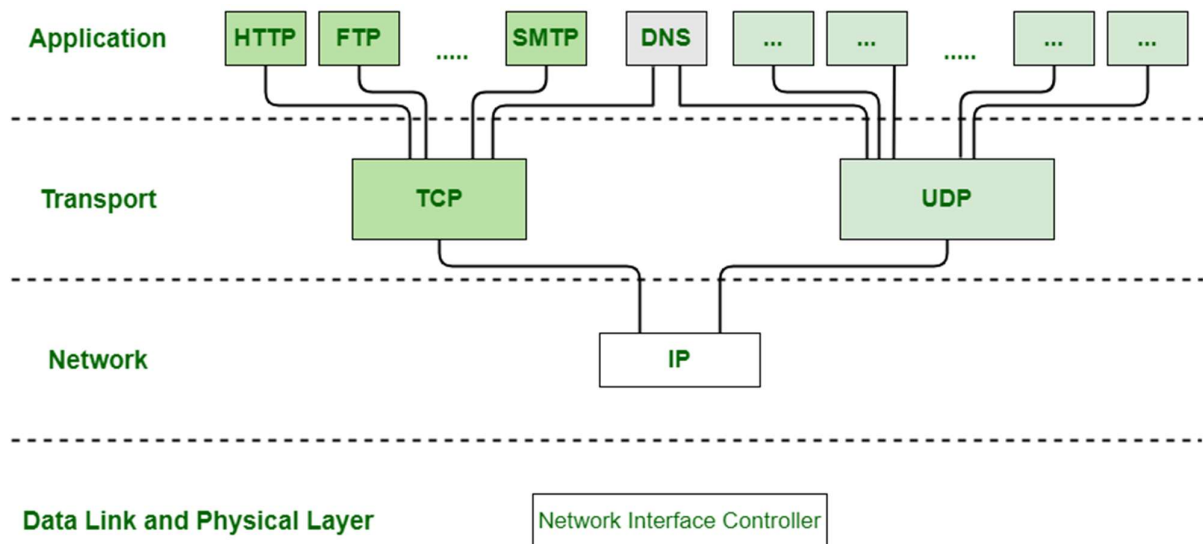
- **Syntax –**  
It is used to explain data format that is needed to be sent or received.
- **Semantics –**  
It is used to explain exact meaning of each of sections of bits that are usually transferred.

- **Timings –**

It is used to explain exact time at which data is generally transferred along with speed at which it is transferred.

### Protocol Hierarchies:

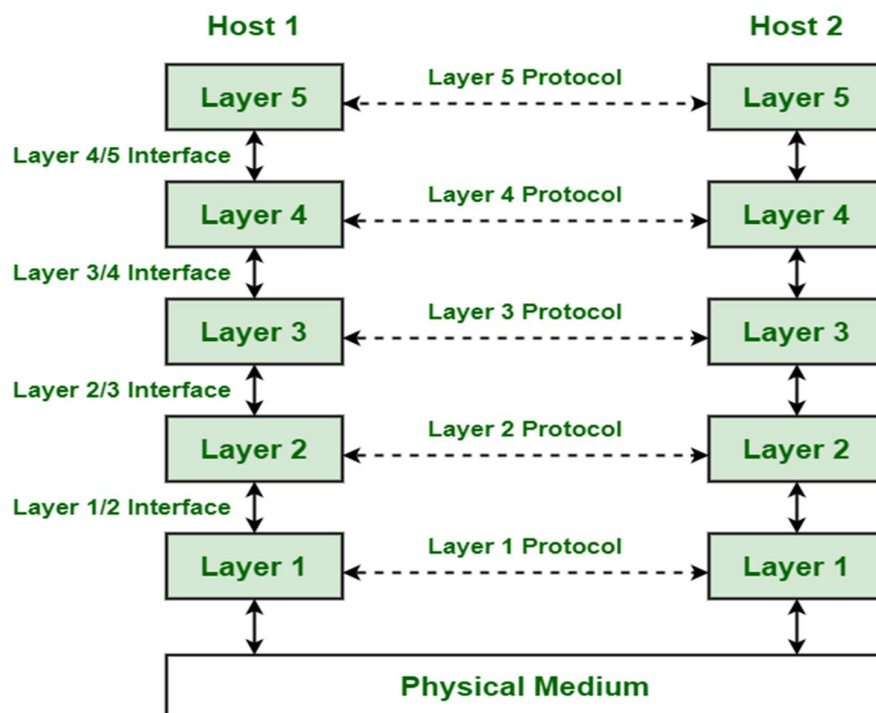
Generally, Computer networks are comprised of or contain a large number of pieces of hardware and software. To just simplify network design, various networks are organized and arranged as a stack of layers of hardware and software, one on top of another. The number, name, content, and function of each layer might vary and can be different from one network to another. The main purpose of each of layers is just to offer and provide services to higher layers that are present. Each and every layer has some particular task or function. In programming, this concept is very common. The networks are organized and arranged as different layers or levels simply to reduce and minimize complexity of design of network software.



### Protocol Hierarchy, modified according to (BADACH et al. 2003)

#### Example :

Below is diagram representing a five-layer network. The diagram shows communication between Host 1 and Host 2. The data stream is passed through a number of layers from one host to other. Virtual communication is represented using dotted lines between peer layers. Physical communication is represented using solid arrows between adjacent layers. Through physical medium, actual communication occurs. The layers at same level are commonly known as peers. The peer basically has a set of communication protocols. An interface is present between each of layers that are used to explain services provided by lower layer to higher layer.



## Physical Hierarchies

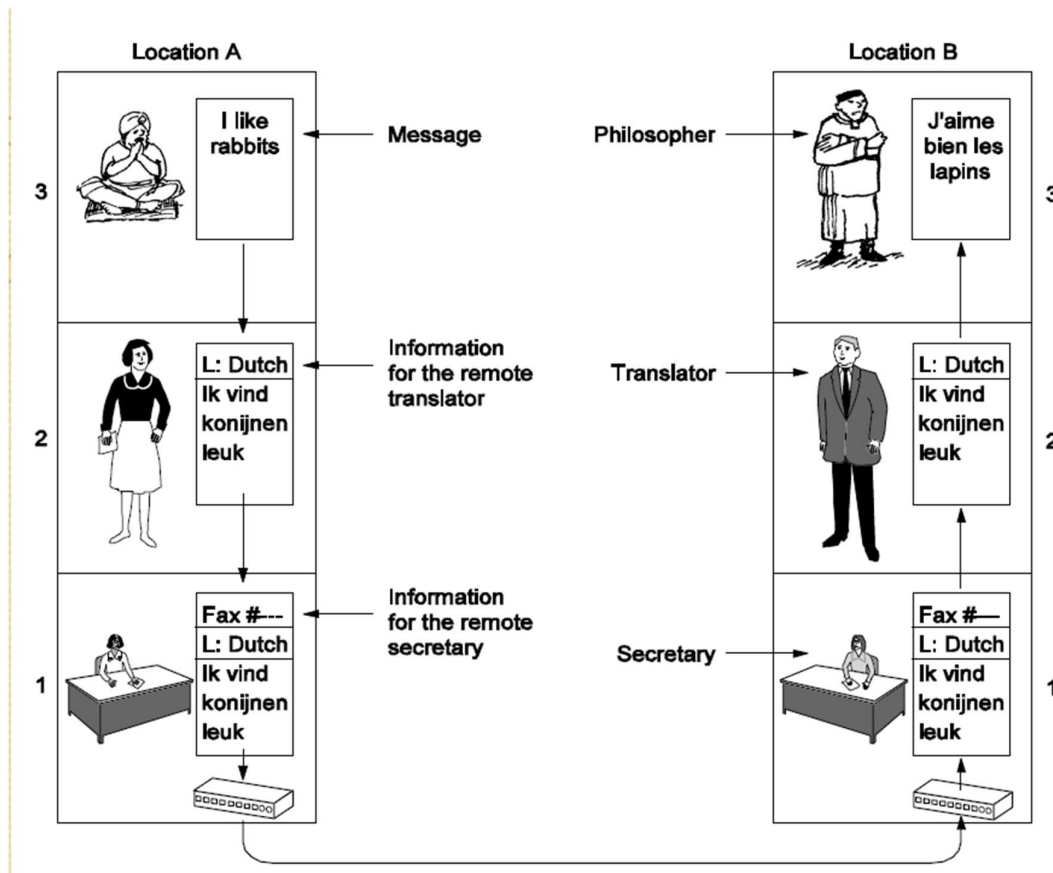
### Advantages :

- The layers generally reduce complexity of communication between networks
- It increases network lifetime.
- It also uses energy efficiently.
- It does not require overall knowledge and understanding of network.

### EXAMPLE

- Two philosophers (layer 3), one of whom speaks Urdu and English and one of whom speaks Chinese and French.
- Since they have no common language, they each engage a translator (layer 2)
- Translators in turn contacts a secretary (layer 1).
- Philosopher 1 passes a message (in English) across the 2/3 interface to his translator, saying “I like rabbits,”
- The translators have agreed on a neutral language known to both of them, Dutch, so the message is converted to “Ik vind konijnen leuk.” The choice of language is the layer 2 protocol and is up to the layer 2 peer processes.
- The translator then gives the message to a secretary for transmission, by, for example, fax (the layer 1 protocol).
- When the message arrives, it is translated into French and passed across the 2/3 interface to philosopher 2.
- Each protocol is completely independent of the other ones
- The translators can switch from Dutch to say, HINDI, provided that they both agree, and neither changes his interface with either layer 1 or layer 3.

- Similarly, the secretaries can switch from fax to e-mail or telephone without disturbing (or even informing) the other layers.



### Design Issues for the Layers

- Addressing
- Error Control
- Flow Control
- Multiplexing
- Routing

### Addressing

- A Network has many computers
- Some means is needed to specify with whom sender wants to talk.
- Since multiple destinations are there, -----some form of addressing is needed in order to specify a specific destination.
- Rules for data transfer
- In some systems, data only travel in one direction; in others, data can go both ways
- The protocol must also determine how many channels the connection corresponds to and what their priorities are. •Many networks provide at least two channels per connection, one for normal data and one for urgent data.

### Error Control

- Error control is an important issue because physical communication circuits are not perfect.



- Many error-correcting codes are known, but both ends of the connection must agree on which one is being used.
- Also the receiver must have some way of telling the sender which messages have been correctly received and which have not.

Issues like-----

- ☐ Not all communication channels preserve the order of messages sent on them.
- ☐ To deal with a possible loss of sequencing, the protocol must make explicit provision for the receiver to allow the pieces to be reassembled properly.
- ☐ An obvious solution is to number the pieces

Another issue is.....

- ☐ Fast Sender and Slow receiver
- ☐ Solutions like acknowledgement
- ☐ Other solutions -----°limit the sender to an agreed-on transmission rate. This subject is called flow control.
- ☐ Inability to accept long messages.
- ☐ This property leads to mechanisms for disassembling, transmitting, and then reassembling messages

### Multiplexing

- ☐ To set up a separate connection for each pair of communicating processes is inconvenient or expensive
- ☐ the underlying layer may use the same connection for multiple, unrelated conversations
- ☐ Multiplexing is needed in the physical layer

### Routing

- ☐ When there are multiple paths between Source & Destination—A Route must be chosen.
- ☐ Sometimes this decision must split over two or more Layers.
- ☐ High Level Decision vs. Low Level Decision based on current traffic load, Known as Routing.

### Service Primitives

Service generally includes set of various primitives. A primitive simply means Operations. A Service is specified by set of primitives that are available and given to user or other various entities to access the service. All these primitives simply tell the service to perform some action or to report on action that is taken

-----  
 High Level Decision vs. Low Level Decision based on current traffic load, Known as Routing.

Connection oriented service	Connectionless Service
<ul style="list-style-type: none"> <li>• Modeled after Telephone System</li> <li>• You pick up phone---dial numb---talk---hang up</li> <li>• Similarly connection oriented service first establish the connection ---uses the connection and then releases it</li> <li>• In most cases bits arrive in the same order as released.</li> <li>• In some cases sender and receiver negotiate about parameters like maximum message size, quality of service etc.</li> </ul>	<ul style="list-style-type: none"> <li><input type="checkbox"/> Modeled after a postal service</li> <li><input type="checkbox"/> Each message carries full destination address</li> <li><input type="checkbox"/> Each one is routed through the system independent of all the others</li> <li><input type="checkbox"/> Order may not be necessarily followed</li> </ul>

by peer entity. Each of the protocol that communicates in layered architecture also communicates in peer-to-peer manner with some of its remote protocol entity. Primitives are called calling functions between the layers



that are used to manage communication among the adjacent protocol layers i.e., among the same communication node. The set of primitives that are available generally depends upon the nature of the service that is being provided. Classification of Service

Primitives of Connection-Oriented Service:

**Listen** - When server is ready to accept request of incoming connection, it simply put this primitive into action. Listen primitive simply waiting for incoming connection request.

**Connect** - This primitive is used to connect the server simply by creating or establishing connection with waiting peer.

**Accept** - This primitive simply accepts incoming connection form peer.

**Receive** - These primitives afterwards block the server. Receive primitive simply waits for incoming message.

**Send** -This primitive is put into action by the client to transmit its request that is followed by putting receive primitive into action to get the reply. Send primitive simply sends or transfer the message to the peer.

**Disconnect** - This primitive is simply used to terminate or end the connection after which no one will be able to send any of the message.

Primitives of Connectionless Service:

**Unit data** - Unit data primitive is simply required to send packet of data or information.

**Report** - This primitive is required for getting details about the performance and working of the network such as delivery statistics or report.

## REFERENCE MODEL:

### The OSI Model

The model is called the ISO OSI (Open System Interconnection) model because it deals with connecting open systems, that is, system that are open for communication with other systems. The following figure shows the overview of OSI Reference Model:

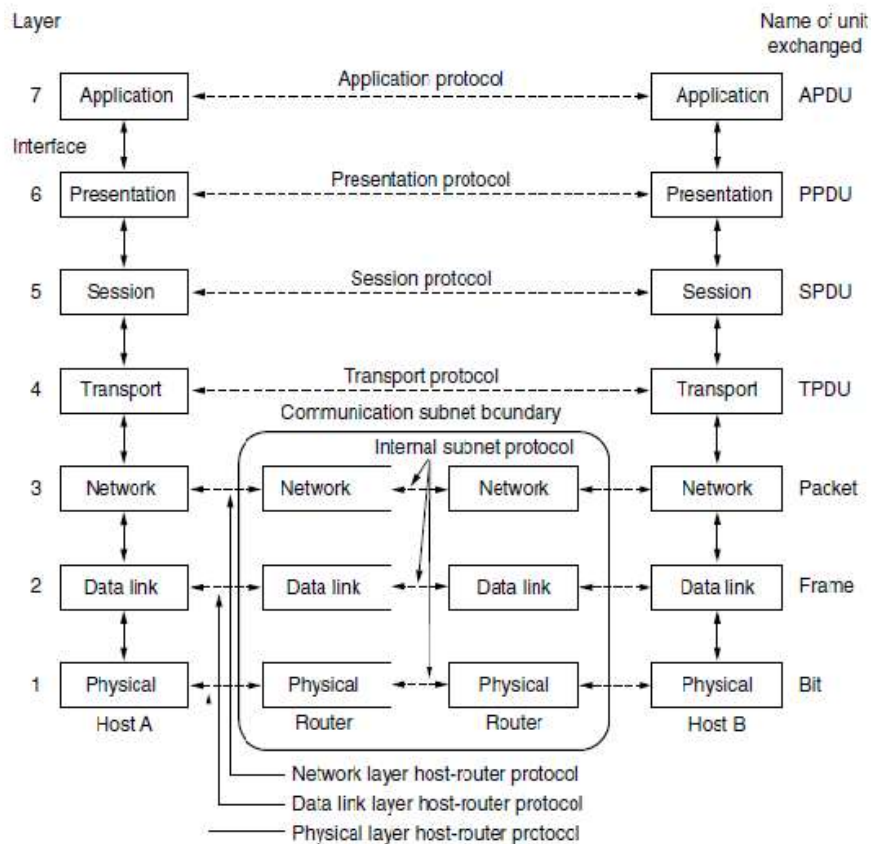


Fig. OSI Reference Model

The OSI model has seven layers. The principles that were applied to arrive at the seven layers can be summarised as follows:

- A layer should be created where a different abstraction is needed.
- Each layer should perform a well-defined function.
- The function of each layer should be chosen with an eye toward defining an intentionally standardised protocol.

### 1. Physical Layer.

- The physical layer is concerned with transmitting raw bits over a communication channel.
- The design issues have to do with making sure that when one side sends a 1 bit, it is received by the other side as a 1 bit, not as 0 bit.
- The design issues here largely deal with mechanical, electrical and timing interfaces and the physical transmission medium which lies below the physical layer.

### 2. Data Link Layer.

- The main task of the data link layer is to transform a raw transmission facility into a line that appears free of undetected transmission errors to the network layer.
- It accomplishes the task by having the sender break up the input data into data frames (typically a few hundred or thousand bytes) and the transmit the frame sequentially.
- If the service is reliable the receiver confirms correct receipt of each frame by sending back an acknowledgement frame.

### 3. Network Layer.

- The network layer handles routing among nodes within a packet switched network.

- At this layer, the unit of data exchange among nodes is typically called a packet.

#### 4. Transport Layer.

- The transport layer then implements what we have to do up to this point been calling a process-to-process channel.
- Here, the unit of data exchanged is commonly called a message rather than a packet or a frame.
- It runs on the end hosts and not on intermediate switches or routers.

#### 5. Session Layer.

- The session layer allows users on different machines to establish a session between them.
- Session after various services includes dialog control (keeping track of whose turn is to transmit), token management (preventing two parties from attempting the same critical operation at the same time), Synchronization (checkpointing long transmission to allow them to continue from where they were after a crash).

#### 6. Presentation Layer.

- Unlike lower layer, which is mostly concerned with moving bits around, the presentation layer is concerned with the system and semantics of the information transmitted.
- It manages abstract data structures and allows higher level data structures to be defined and exchanged.

#### 7. Application Layer.

- The application layer contains a variety of protocols that are commonly needed by users.
- One widely used application protocol is HTTP (Hyper Text Transfer Protocol) which is the basis for the World Wide Web(www).

#### TCP/IP REFERENCE MODEL

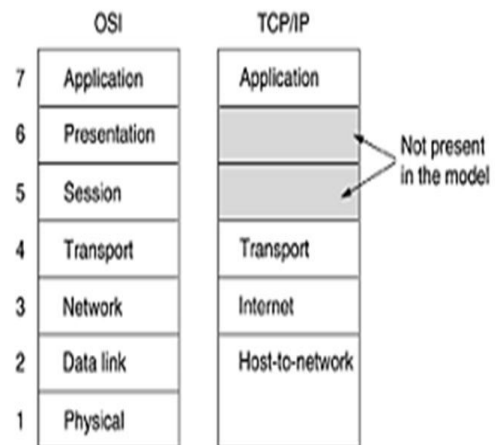
TCP/IP reference model named after two of its primary protocols. That is Transmission Control Protocol which resides in Transport Layer and the other one is Internet Protocol which resides in Internet Layer

#### What is a Protocol?

Protocol is a set of rules. Thus Transmission Control Protocol (TCP) means it is a set rules followed to perform the functionalities of Transport Layer. Similarly Internet Protocol means it is a set of rules followed to perform the functions of Internet layer in the TCP/IP reference model

**1) Host to Network Layer:** The job of this layer is to transmit the packets given by the Internet layer from the Host to the network to which the Host connected.

**2) Internet Layer:** The job of this layer is to create the packet which is known as Datagram and route it to the proper destination. It adds a lot of control information so as to make the packet reach the correct destination. The protocol present in this layer is known as Internet Protocol (IP). IP is a connectionless protocol. Different packets created in this layer may not follow the same path to reach the destination. For example you want to send some packets to a Server in New York then some packets may



be sent through a specific route and some packets may be sent through some other route and may not be the same route. **Why is it doing this?** This is because, if a specific route through which it sent the packet becomes busy then it will send the packet through some other route. Because of this it may so happen that the packet may not reach the destination in sequence and it will be the job of upper layer to arrange the packets in proper sequence and recreate the message.

**3) Transport layer:** The task of this layer is to establish the connection between the peer entities in the source and destination host. Two protocols reside in this layer one is Transmission Control Protocol (TCP) which is reliable, connection oriented protocol and the other one is the User Datagram Protocol (UDP) which is unreliable, connectionless protocol. Other function of TCP is to flow control between the slower and faster entities. TCP is also responsible for the errorless transmission of packets from source to destination. At the source TCP divides the stream of bits it receives from the upper layer and creates packets and gives it to the Internet Layer. At the destination it assembles the packets received in proper sequence even if the packets are received out of sequence, check for errors and if any, it is notified to the Transport layer at the source machine and the Transport Layer at the source machine resends the requested packet.

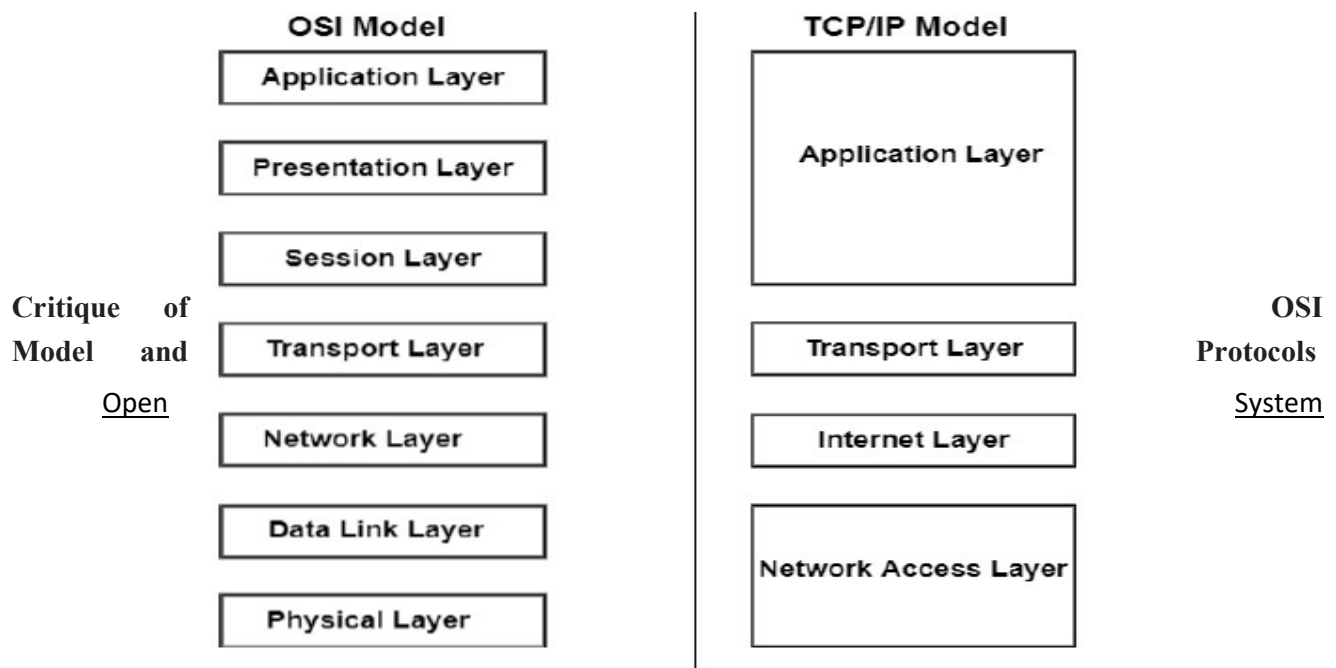
**4) Application Layer:** Different types of application and the protocols reside in this layer and they handle different types of communication

Comparison of OSI and TCP/IP Models:

Following are the differences between OSI and TCP/IP Reference Model –

OSI	TCP/IP
OSI represents <b>Open System Interconnection</b> .	TCP/IP model represents the Transmission Control Protocol / Internet Protocol.
OSI is a generic, protocol independent standard. It is acting as an interaction gateway between the network and the final-user.	TCP/IP model depends on standard protocols about which the computer network has created. It is a connection protocol that assigns the network of hosts over the internet.
The OSI model was developed first, and then protocols were created to fit the network architecture's needs.	The protocols were created first and then built the TCP/IP model.
It provides quality services.	It does not provide quality services.
The OSI model represents defines administration, interfaces and conventions. It describes clearly which layer provides services.	It does not mention the services, interfaces, and protocols.
The protocols of the OSI model are better unseen and can be returned with another appropriate protocol quickly.	The TCP/IP model protocols are not hidden, and we cannot fit a new protocol stack in it.

OSI	TCP/IP
It is difficult as distinguished to TCP/IP.	It is simpler than OSI.
It provides both connection and connectionless oriented transmission in the network layer; however, only connection-oriented transmission in the transport layer.	It provides connectionless transmission in the network layer and supports connecting and connectionless-oriented transmission in the transport layer.
It uses a vertical approach.	It uses a horizontal approach.
The smallest size of the OSI header is 5 bytes.	The smallest size of the TCP/IP header is 20 bytes.
Protocols are unknown in the OSI model and are returned while the technology modifies.	In TCP/IP, returning protocol is not difficult.



Interconnection (OSI) model is reference model that is used to describe and explain how does information from software application in one of computers moves freely through physical medium to software application

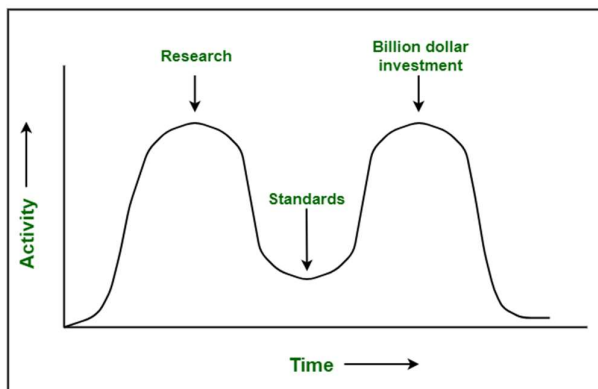
on another computer. This model consists of total of seven layers and each of layers performs specific task or particular network function.

Although, OSI model and its protocols even TCP/IP models and its protocols are not perfect in each and manner. There is bit of criticism that has been noticed and directed at both of them. The most striking and unfortunate issue concerning OSI model is that it is perhaps the most-studied and most widely accepted network structure and yet it is not model that is really implemented and largely used. The important reasons why happen is given below:

### 1. Bad Timing :

In the OSI model, it is very essential and important to write standards in between trough i.e., apocalypse of two elephants. Time of standards is very critical as sometimes standards are written too early even before research is completed. Due to this, OSI model was not properly understood. The timing was considered bad because this model was finished and completed after huge and significant amount of research time. Due to this, the standards are ignored by these companies.

When the OSI came around, this model was perfectly released regarding research, but at that time TCP/IP model was already receiving huge amounts of investments from companies and manufacturers did not feel like investing in OSI model. So, there were no initial offerings for using OSI technique. While every company waited for any of other companies to firstly use this model technique, but unfortunately none of company went first to use this model. This is first reason why OSI never happen.



**Apocalypse of the Two Elephants**

### 2. Bad Technology :

OSI models were never taken into consideration because of competition TCP/IP protocols that were already used widely. This is due to second reason that OSI model and its protocols are flawed that means both of them have fundamental weakness or imperfection or defect in character or performance or design, etc. The idea behind choosing all of seven layers of OSI model was based more on political issues rather than technical. Layers are more political than technical.

OSI model, along with all of its associated service definitions and protocols, is highly complex. On the other hand, other two layers i.e. Data link layer and network layer both of them are overfull. Documentation is also highly complex due to which it gets very difficult to implement and is not even very efficient in operation or function. Error and flow control are also duplicated i.e., reappear again and again in multiple layers or each layer. On the other hand, most serious and bad criticism is that this model is also dominated by communications mentality.

### 3. Bad Implementations :

The OSI model is extraordinarily and much more complex due to which initial implementations were very

slow, huge, and unwidely. This is the third reason due to which OSI became synonymous with poor quality in early days. It turned out to not be essential and necessary for all of seven layers to be designed together to simply make things work out.

On the other hand, implementations of TCP/IP were more reliable than OSI due to which people started using TCP/IP very quickly which led to large community of users. In simple words, we can say that complexity leads to very poor or bad implementation. It is highly complex to be effectively and properly implemented.

#### **4. Bad Politics :**

OSI model was not associated with UNIX. This was fourth reason because TCP/IP was largely and closely associated with Unix, which helps TCP/IP to get popular in academia whereas OSI did not have this association at that time.

On the other hand, OSI was associated with European telecommunications, European community, and government of USA. This model was also considered to be technically inferior to TCP/IP. So, all people on ground reacted very badly to all of these things and supported much use of [TCP/IP](#).

Even after all these bad conditions, OSI model is still general standard reference for almost all of networking documentation. There are many organizations that are highly interested in OSI model. All of networking that is referring to numbered layers like layer 3 switching generally refers to OSI model. Even, an effort has also been made simply to update it resulting in revised model that was published in 1994.

#### **Critique of the TCP/IP Reference Model**

The TCP/IP model and protocols have their problems too. First, the model does not clearly distinguish the concepts of service, interface, and protocol. Good software engineering practice requires differentiating between the specification and the implementation, something that OSI does very carefully, and TCP/IP does not. Consequently, the TCP/IP model is not much of a guide for designing new networks using new technologies.

Second, the TCP/IP model is not at all general and is poorly suited to describing any protocol stack other than TCP/IP. Trying to use the TCP/IP model to describe Bluetooth, for example, is completely impossible.

Third, the host-to-network layer is not really a layer at all in the normal sense of the term as used in the context of layered protocols. It is an interface (between the network and data link layers). The distinction between an interface and a layer is crucial, and one should not be sloppy about it.

Fourth, the TCP/IP model does not distinguish (or even mention) the physical and data link layers. These are completely different. The physical layer has to do with the transmission characteristics of copper wire, fiber optics, and wireless communication. The data link layer's job is to delimit the start and end of frames and get them from one side to the other with the desired degree of reliability. A proper model should include both as separate layers. The TCP/IP model does not do this.

Finally, although the IP and TCP protocols were carefully thought out and well implemented, many of the other

protocols were ad hoc, generally produced by a couple of graduate students hacking away until they got tired.



The protocol implementations were then distributed free, which resulted in their becoming widely used, deeply entrenched, and thus hard to replace. Some of them are a bit of an embarrassment now. The virtual terminal protocol, TELNET, for example, was designed for a ten-character per second mechanical Teletype terminal. It knows nothing of graphical user interfaces and mice. Nevertheless, 25 years later, it is still in widespread use.

In summary, despite its problems, the OSI model (minus the session and presentation layers) has proven to be exceptionally useful for discussing computer networks. In contrast, the OSI protocols have not become popular. The reverse is true of TCP/IP: the model is practically nonexistent, but the protocols are widely used. Since computer scientists like to have their cake and eat it, too, in this book we will use a modified OSI model but concentrate primarily on the TCP/IP and related protocols, as well as newer ones such as 802, SONET, and Bluetooth. In effect.

## UNIT II

### PHYSICAL LAYER:

#### INTRODUCTION:

Physical layer in the OSI model plays the role of interacting with actual hardware and signaling mechanism. Physical layer is the only layer of OSI network model which actually deals with the physical connectivity of two different stations. This layer defines the hardware equipment, cabling, wiring, frequencies, pulses used to represent binary signals etc.

Physical layer provides its services to Data-link layer. Data-link layer hands over frames to physical layer. Physical layer converts them to electrical pulses, which represent binary data. The binary data is then sent over the wired or wireless media.

#### Signals

When data is sent over physical medium, it needs to be first converted into electromagnetic signals. Data itself can be analog such as human voice, or digital such as file on the disk. Both analog and digital data can be represented in digital or analog signals.

- **Digital Signals**

Digital signals are discrete in nature and represent sequence of voltage pulses. Digital signals are used within the circuitry of a computer system.

- **Analog Signals**

Analog signals are in continuous wave form in nature and represented by continuous electromagnetic waves.

#### Transmission Impairment

When signals travel through the medium they tend to deteriorate. This may have many reasons as given:

- **Attenuation**

For the receiver to interpret the data accurately, the signal must be sufficiently strong. When the signal passes through the medium, it tends to get weaker. As it covers distance, it loses strength.

- **Dispersion**

As signal travels through the media, it tends to spread and overlaps. The amount of dispersion depends upon the frequency used.

- **Delay distortion**

Signals are sent over media with pre-defined speed and frequency. If the signal speed and frequency do not match, there are possibilities that signal reaches destination in arbitrary fashion. In digital media, this is very critical that some bits reach earlier than the previously sent ones.

- **Noise**

Random disturbance or fluctuation in analog or digital signal is said to be Noise in signal, which may distort the actual information being carried. Noise can be characterized in one of the following class:

- **Thermal Noise**

Heat agitates the electronic conductors of a medium which may introduce noise in the media. Up to a certain level, thermal noise is unavoidable.

- **Intermodulation**

When multiple frequencies share a medium, their interference can cause noise in the medium. Intermodulation noise occurs if two different frequencies are sharing a medium and one of them has excessive strength or the component itself is not functioning properly, then the resultant frequency may not be delivered as expected.

- **Crosstalk**

This sort of noise happens when a foreign signal enters into the media. This is because signal in one medium affects the signal of second medium.

- **Impulse**

This noise is introduced because of irregular disturbances such as lightening, electricity, short-circuit, or faulty components. Digital data is mostly affected by this sort of noise

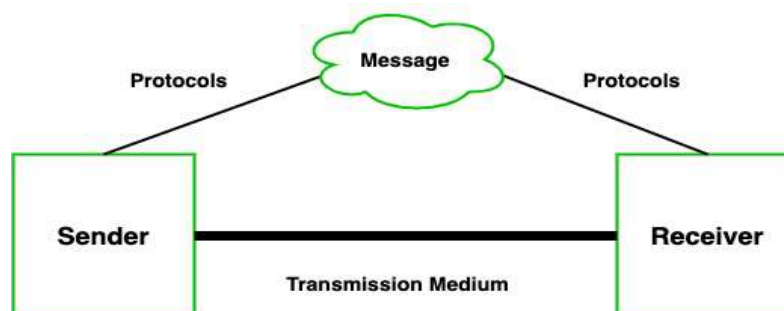
### THEORETICAL BASIS FOR DATA COMMUNICATION:

Human beings are the only creatures on the earth who are able to communicate with each other through the medium of language. But humans take this gift to another extent. Distance, time, and physical existence of the person don't matter in communication these days because they build a communication system through which they can communicate or share data like images, videos, text, files, etc with their loved ones anytime anywhere. Communication is defined as a process in which more than one computer transfers information, instructions to each other and for sharing resources. Or in other words, communication is a process or act in which we can send or receive data. A network of computers is defined as an interconnected collection of autonomous computers. Autonomous means no computer can start, stop or control another computer.

### **Components of Data Communication**

A communication system is made up of the following components:

1. **Message:** A message is a piece of information that is to be transmitted from one person to another. It could be a text file, an audio file, a video file, etc.
2. **Sender:** It is simply a device that sends data messages. It can be a computer, mobile, telephone, laptop, video camera, or workstation, etc.
3. **Receiver:** It is a device that receives messages. It can be a computer, telephone mobile, workstation, etc.
4. **Transmission Medium / Communication Channels:** Communication channels are the medium that connect two or more workstations. Workstations can be connected by either wired media or wireless media.
5. **Set of rules (Protocol):** When someone sends the data (The sender), it should be understandable to the receiver also otherwise it is meaningless. For example, Sonali sends a message to Chetan. If Sonali writes in Hindi and Chetan cannot understand Hindi, it is a meaningless conversation.



Therefore, there are some set of rules (protocols) that is followed by every computer connected to the internet and they are:

- **TCP(Transmission Control Protocol):** It is responsible for dividing messages into packets on the source computer and reassembling the received packet at the destination or recipient computer. It also makes sure that the packets have the information about the source of the message data, the destination of the message data, the sequence in which the message data should be re-assembled, and checks if the message has been sent correctly to the specific destination.
- **IP(Internet Protocol):** Do You ever wonder how does computer determine which packet belongs to which device. What happens if the message you sent to your friend is received by your father? Scary Right. Well! IP is responsible for handling the address of the destination computer so that each packet is sent to its proper destination.

### Type of data communication

As we know that data communication is communication in which we can send or receive data from one device to another. The data communication is divided into three types:

1. **Simplex Communication:** It is one-way communication or we can say that unidirectional communication in which one device only receives and another device only sends data and devices uses their entire capacity in transmission. For example, IoT, entering data using a keyboard, listening music using a speaker, etc.
2. **Half Duplex communication:** It is a two-way communication or we can say that it is a bidirectional communication in which both the devices can send and receive data but not at the same time. When one device is sending data then another device is only receiving and vice-versa. For example, walkie-talkie.
3. **Full-duplex communication:** It is a two-way communication or we can say that it is a bidirectional communication in which both the devices can send and receive data at the same time. For example, mobile phones, landlines, etc.

### Communication Channels

Communication channels are the medium that connects two or more workstations. Workstations can be connected by either wired media or wireless media. It is also known as a transmission medium. The transmission medium or channel is a link that carries messages between two or more devices. We can group the communication media into two categories:

- Guided media transmission
- Unguided media transmission

1. **Guided Media:** In this transmission medium, the physical link is created using wires or cables between two or more computers or devices, and then the data is transmitted using these cables in terms of signals. Guided media transmission of the following types:

1. **Twisted pair cable:** It is the most common form of wire used in communication. In a twisted-pair cable, two identical wires are wrapped together in a double helix. The twisting of the wire reduces the crosstalk. It is known as the leaking of a signal from one wire to another due to which signal can corrupt and can cause network errors. The twisting protects the wire from internal crosstalk as well as external forms of signal interference. Types of Twisted Pair Cable :

- **Unshielded Twisted Pair (UTP):** It is used in computers and telephones widely. As the name suggests, there is no external shielding so it does not protect from external interference. It is cheaper than STP.
- **Shielded Twisted Pair (STP):** It offers greater protection from crosstalk due to shield. Due to shielding, it protects from external interference. It is heavier and costlier as compare to UTP.

**2. Coaxial Cable:** It consists of a solid wire core that is surrounded by one or more foil or wire shields. The inner core of the coaxial cable carries the signal and the outer shield provides the ground. It is widely used for television signals and also used by large corporations in building security systems. Data transmission of this cable is better but expensive as compared to twisted pair.

**3. Optical fibers:** Optical fiber is an important technology. It transmits large amounts of data at very high speeds due to which it is widely used in internet cables. It carries data as a light that travels inside a thin glass fiber. The fiber optic cable is made up of three pieces:

1. **Core:** Core is the piece through which light travels. It is generally created using glass or plastic.
2. **Cladding:** It is the covering of the core and reflects the light back to the core.
3. **Sheath:** It is the protective covering that protects fiber cable from the environment.

**2. Unguided Media:** The unguided transmission media is a transmission mode in which the signals are propagated from one device to another device wirelessly. Signals can wave through the air, water, or vacuum. It is generally used to transmit signals in all directions. Unguided Media is further divided into various parts :

**1. Microwave:** Microwave offers communication without the use of cables. Microwave signals are just like radio and television signals. It is used in long-distance communication. Microwave transmission consists of a transmitter, receiver, and atmosphere. In microwave communication, there are parabolic antennas that are mounted on the towers to send a beam to another antenna. The higher the tower, the greater the range.

**2. Radio wave:** When communication is carried out by radio frequencies, then it is termed radio waves transmission. It offers mobility. It consists of the transmitter and the receiver. Both use antennas to radiate and capture the radio signal.

**3. Infrared:** It is short-distance communication and can pass through any object. It is generally used in TV remotes, wireless mouse, etc.

### GUIDED TRANSMISSION MEDIA

It is defined as the physical medium through which the signals are transmitted. It is also known as Bounded media.

#### Types Of Guided media:

##### Twisted pair:

Twisted pair is a physical media made up of a pair of cables twisted with each other. A twisted pair cable is cheap as compared to other transmission media. Installation of the twisted pair cable is easy, and it is a lightweight cable. The frequency range for twisted pair cable is from 0 to 3.5KHz.

A twisted pair consists of two insulated copper wires arranged in a regular spiral pattern.

The degree of reduction in noise interference is determined by the number of turns per foot. Increasing the number of turns per foot decreases noise interference.



#### Types of Twisted pair:

##### Unshielded Twisted Pair:

An unshielded twisted pair is widely used in telecommunication. Following are the categories of the unshielded twisted pair cable:

- **Category 1:** Category 1 is used for telephone lines that have low-speed data.
- **Category 2:** It can support upto 4Mbps.
- **Category 3:** It can support upto 16Mbps.
- **Category 4:** It can support upto 20Mbps. Therefore, it can be used for long-distance communication.
- **Category 5:** It can support upto 200Mbps.

#### **Advantages Of Unshielded Twisted Pair:**

- It is cheap.
- Installation of the unshielded twisted pair is easy.
- It can be used for high-speed LAN.

#### **Disadvantage:**

- This cable can only be used for shorter distances because of attenuation.

#### **Shielded Twisted Pair**

A shielded twisted pair is a cable that contains the mesh surrounding the wire that allows the higher transmission rate.

#### **Characteristics Of Shielded Twisted Pair:**

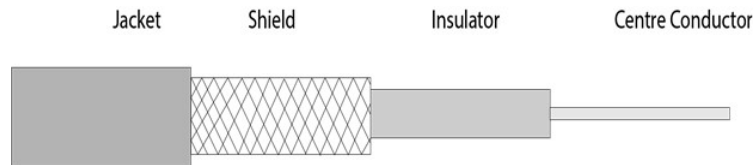
- The cost of the shielded twisted pair cable is not very high and not very low.
- An installation of STP is easy.
- It has higher capacity as compared to unshielded twisted pair cable.
- It has a higher attenuation.
- It is shielded that provides the higher data transmission rate.

#### **Disadvantages**

- It is more expensive as compared to UTP and coaxial cable.
- It has a higher attenuation rate.

#### **Coaxial Cable**

- Coaxial cable is very commonly used transmission media, for example, TV wire is usually a coaxial cable.
- The name of the cable is coaxial as it contains two conductors parallel to each other.
- It has a higher frequency as compared to Twisted pair cable.
- The inner conductor of the coaxial cable is made up of copper, and the outer conductor is made up of copper mesh. The middle core is made up of non-conductive cover that separates the inner conductor from the outer conductor.
- The middle core is responsible for the data transferring whereas the copper mesh prevents from the **EMI**(Electromagnetic interference).



**Coaxial cable is of two types:**

1. **Baseband transmission:** It is defined as the process of transmitting a single signal at high speed.
2. **Broadband transmission:** It is defined as the process of transmitting multiple signals simultaneously.

**Advantages Of Coaxial cable:**

- The data can be transmitted at high speed.
- It has better shielding as compared to twisted pair cable.
- It provides higher bandwidth.

**Disadvantages Of Coaxial cable:**

- It is more expensive as compared to twisted pair cable.
- If any fault occurs in the cable causes the failure in the entire network.

**Fibre Optic**

- Fibre optic cable is a cable that uses electrical signals for communication.
- Fibre optic is a cable that holds the optical fibres coated in plastic that are used to send the data by pulses of light.
- The plastic coating protects the optical fibres from heat, cold, electromagnetic interference from other types of wiring.
- Fibre optics provide faster data transmission than copper wires.

**Diagrammatic representation of fibre optic cable:**



- **Core:** The optical fibre consists of a narrow strand of glass or plastic known as a core. A core is a light transmission area of the fibre. The more the area of the core, the more light will be transmitted into the fibre.
- **Cladding:** The concentric layer of glass is known as cladding. The main functionality of the cladding is to provide the lower refractive index at the core interface as to cause the reflection within the core so that the light waves are transmitted through the fibre.
- **Jacket:** The protective coating consisting of plastic is known as a jacket. The main purpose of a jacket is to preserve the fibre strength, absorb shock and extra fibre protection.

**Following are the advantages of fibre optic cable over copper:**

- **Greater Bandwidth:** The fibre optic cable provides more bandwidth as compared copper. Therefore, the fibre optic carries more data as compared to copper cable.



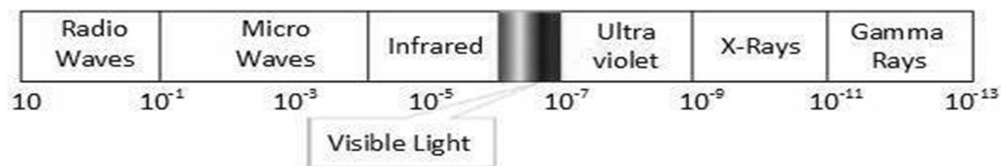
- **Faster speed:** Fibre optic cable carries the data in the form of light. This allows the fibre optic cable to carry the signals at a higher speed.
- **Longer distances:** The fibre optic cable carries the data at a longer distance as compared to copper cable.
- **Better reliability:** The fibre optic cable is more reliable than the copper cable as it is immune to any temperature changes while it can cause obstruct in the connectivity of copper cable.
- **Thinner and Sturdier:** Fibre optic cable is thinner and lighter in weight so it can withstand more pull pressure than copper cable.

## WIRELESS TRANSMISSION MEDIA

Wireless transmission is a form of unguided media. Wireless communication involves no physical link established between two or more devices, communicating wirelessly. Wireless signals are spread over in the air and are received and interpreted by appropriate antennas.

When an antenna is attached to electrical circuit of a computer or wireless device, it converts the digital data into wireless signals and spread all over within its frequency range. The receptor on the other end receives these signals and converts them back to digital data.

A little part of electromagnetic spectrum can be used for wireless transmission.

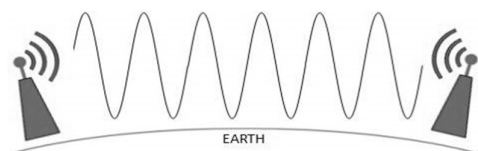


## Radio Transmission

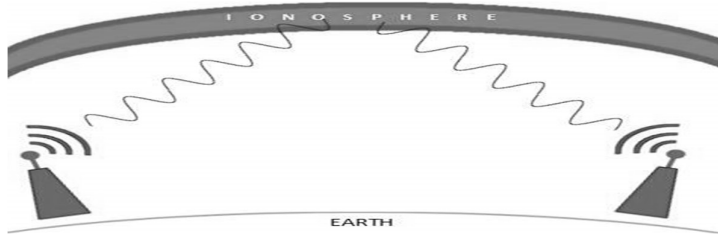
Radio frequency is easier to generate and because of its large wavelength it can penetrate through walls and structures alike. Radio waves can have wavelength from 1 mm – 100,000 km and have frequency ranging from 3 Hz (Extremely Low Frequency) to 300 GHz (Extremely High Frequency). Radio frequencies are sub-divided into six bands.

Radio waves at lower frequencies can travel through walls whereas higher RF can travel in straight line and bounce back. The power of low frequency waves decreases sharply as they cover long distance. High frequency radio waves have more power.

Lower frequencies such as VLF, LF, MF bands can travel on the ground up to 1000 kilometers, over the earth's surface.



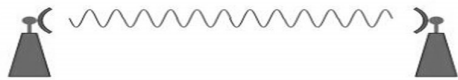
Radio waves of high frequencies are prone to be absorbed by rain and other obstacles. They use Ionosphere of earth atmosphere. High frequency radio waves such as HF and VHF bands are spread upwards. When they reach Ionosphere, they are refracted back to the earth.



### **Microwave Transmission**

Electromagnetic waves above 100 MHz tend to travel in a straight line and signals over them can be sent by beaming those waves towards one particular station. Because Microwaves travels in straight lines, both sender and receiver must be aligned to be strictly in line-of-sight.

Microwaves can have wavelength ranging from 1 mm – 1 meter and frequency ranging from 300 MHz to 300 GHz.



Microwave antennas concentrate the waves making a beam of it. As shown in picture above, multiple antennas can be aligned to reach farther. Microwaves have higher frequencies and do not penetrate wall like obstacles.

Microwave transmission depends highly upon the weather conditions and the frequency it is using.

### **Infrared Transmission**

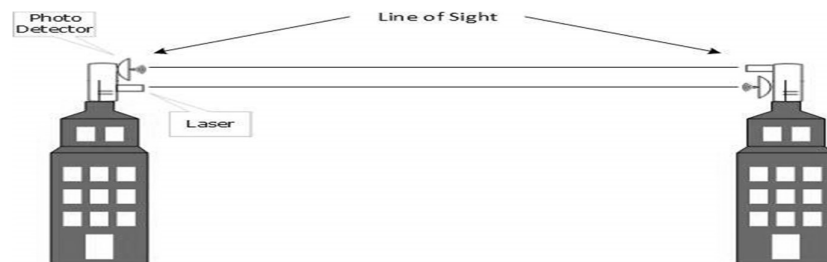
Infrared wave lies in between visible light spectrum and microwaves. It has wavelength of 700-nm to 1-mm and frequency ranges from 300-GHz to 430-THz.

Infrared wave is used for very short range communication purposes such as television and it's remote. Infrared travels in a straight line hence it is directional by nature. Because of high frequency range, Infrared cannot cross wall-like obstacles.

### **Light Transmission**

Highest most electromagnetic spectrum which can be used for data transmission is light or optical signaling. This is achieved by means of LASER.

Because of frequency light uses, it tends to travel strictly in straight line. Hence the sender and receiver must be in the line-of-sight. Because laser transmission is unidirectional, at both ends of communication the laser and the photo-detector needs to be installed. Laser beam is generally 1mm wide hence it is a work of precision to align two far receptors each pointing to lasers source.



Laser works as Tx (transmitter) and photo-detectors works as Rx (receiver).

Lasers cannot penetrate obstacles such as walls, rain, and thick fog. Additionally, laser beam is distorted by wind, atmosphere temperature, or variation in temperature in the path. Laser is safe for data transmission as it is very difficult to tap 1mm wide laser without interrupting the communication channel.

### **COMMUNICATION SATELLITES**

A satellite is an object that revolves around another object. For example, earth is a satellite of The Sun, and moon is a satellite of earth.

A **communication satellite** is a **microwave repeater station** in a space that is used for telecommunication, radio and television signals. A communication satellite processes the data coming from one earth station and it converts the data into another form and send it to the second earth station.

### How a Satellite Works

Two stations on earth want to communicate through radio broadcast but are too far away to use conventional means. The two stations can use a relay station for their communication. One earth station transmits the signal to the satellite.

**Uplink frequency** is the frequency at which ground station is communicating with satellite. The satellite transponder converts the signal and sends it down to the second earth station, and this is called **Downlink frequency**. The second earth station also communicates with the first one in the same way.

### **Advantages of Satellite**

The advantages of Satellite Communications are as follows –

- The Coverage area is very high than that of terrestrial systems.
- The transmission cost is independent of the coverage area.
- Higher bandwidths are possible.

### **Disadvantages of Satellite**

The disadvantages of Satellite Communications are as follows –

- Launching satellites into orbits is a costly process.
- The bandwidths are gradually used up.
- High propagation delay for satellite systems than the conventional terrestrial systems.

### **Earth Orbits**

A satellite when launched into space, needs to be placed in certain orbit to provide a particular way for its revolution, so as to maintain accessibility and serve its purpose whether scientific, military or commercial. Such orbits which are assigned to satellites, with respect to earth are called as **Earth Orbits**. The satellites in these orbits are Earth Orbit Satellites.

The important kinds of Earth Orbits are –

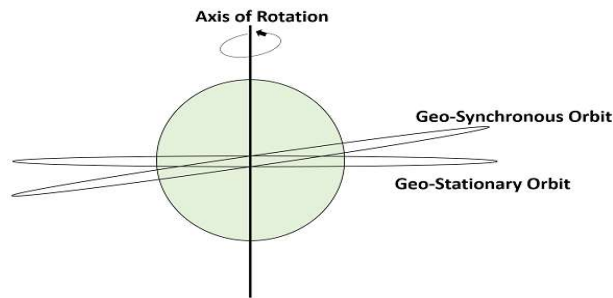
- Geo-synchronous Earth Orbit
- Geo-stationary Earth Orbit
- Medium Earth Orbit
- Low Earth Orbit

### **Geo-synchronous Earth Orbit (GEO) Satellites**

A Geo-synchronous Earth orbit Satellite is one which is placed at an altitude of 22,300 miles above the Earth. This orbit is synchronized with a **side real day** (i.e., 23hours 56minutes). This orbit can **have inclination and eccentricity**.

The same geo-synchronous orbit, if it is **circular** and in the plane of equator, it is called as geo-stationary orbit. These Satellites are placed at 35,900kms (same as geosynchronous) above the Earth's Equator and they keep on rotating with respect to earth's direction (west to east). These satellites are considered **stationary** with

respect to earth and hence the name implies. Geo-Stationary Earth Orbit Satellites are used for weather forecasting, satellite TV, satellite radio and other types of global communications.



The above figure shows the difference between Geo-synchronous and Geo- Stationary orbits. The Axis of rotation indicates the movement of Earth.

The main point to note here is that every Geo-Stationary orbit is a Geo-Synchronous orbit. But every Geo-Synchronous orbit is NOT a Geo-stationary orbit.

### **Medium Earth Orbit (MEO) Satellites**

Medium earth orbit (MEO) satellite networks will orbit at distances of about 8000 miles from earth's surface. Signals transmitted from a MEO satellite travel a shorter distance. This translates to improved signal strength at the receiving end. This shows that smaller, more lightweight receiving terminals can be used at the receiving end.

Since the signal is travelling a shorter distance to and from the satellite, there is less transmission delay. **Transmission delay** can be defined as the time it takes for a signal to travel up to a satellite and back down to a receiving station.

For real-time communications, the shorter the transmission delay, the better will be the communication system. As an example, if a GEO satellite requires 0.25 seconds for a round trip, then MEO satellite requires less than 0.1 seconds to complete the same trip. MEOs operates in the frequency range of 2 GHz and above.

### **Low Earth Orbit (LEO) Satellites**

The LEO satellites are mainly classified into three categories namely, little LEOs, big LEOs, and Mega-LEOs. LEOs will orbit at a distance of 500 to 1000 miles above the earth's surface.

This relatively short distance reduces transmission delay to only 0.05 seconds. This further reduces the need for sensitive and bulky receiving equipment. Little LEOs will operate in the 800 MHz (0.8 GHz) range. Big LEOs will operate in the 2 GHz or above range, and Mega-LEOs operates in the 20-30 GHz range.

The higher frequencies associated with **Mega-LEOs** translates into more information carrying capacity and yields to the capability of real-time, low delay video transmission scheme.

## Public Switched Telephone Network



### PUBLIC SWITCHED TELEPHONE NETWORK:

Public Switched Telephone Network (PSTN) is an agglomeration of an interconnected network of telephone lines owned by both governments as well as commercial organizations.

#### Properties of PSTN

- It is also known as Plain Old Telephone Service (POTS)
- It has evolved from the invention of telephone by Alexander Graham Bell.
- The individual networks can be owned by national government, regional government or private telephone operators.
- Its main objective is to transmit human voice in a recognizable form.
- It is an aggregation of circuit-switched networks of the world.
- Originally, it was an entirely analog network laid with copper cables and switches.
- Presently, most part of PSTN networks is digitized and comprises of a wide variety communicating devices.
- The present PSTNs comprises of copper telephone lines, fibre optic cables, communication satellites, microwave transmission links and undersea telephone lines. It is also linked to the cellular networks.
- The interconnection between the different parts of the telephone system is done by switching centres. This allows multiple telephone and cellular networks to communicate with each other.
- Present telephone systems are tightly coupled with WANs (wide area networks) and are used for both data and voice communications.
- The operation of PSTN networks follows the ITU-T standards.

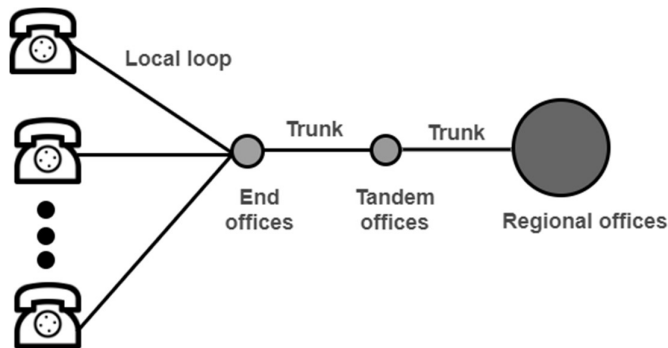
### THE TELEPHONE SYSTEM

In the Telephone system, Telephone Network is used to provide voice communication. Telephone Network uses Circuit Switching. Originally, the entire network was referred to as a plain old telephone system (POTS) which uses analog signals. With the advancement of technology, i.e. in the computer era, there comes a feature to carry data in addition to voice. Today's network is both analogous and digital.

**Major Components of Telephone System:** There are three major components of the telephone system:

1. Local loops
2. Trunks
3. Switching Offices

There are various levels of switching offices such as end offices, tandem offices, and regional offices. The entire telephone network is as shown in the following figure:

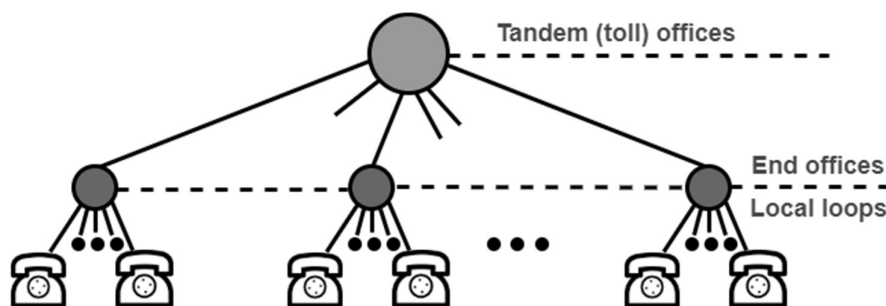


*A telephone system*

**Local Loops:** Local Loops are the twisted pair cables that are used to connect a subscriber telephone to the nearest end office or local central office. For voice purposes, its bandwidth is 4000 Hz. It is very interesting to examine the telephone number that is associated with each local loop. The office is defined by the first three digits and the local loop number is defined by the next four digits defines.

**Trunks:** It is a type of transmission medium used to handle the communication between offices. Through multiplexing, trunks can handle hundreds or thousands of connections. Mainly transmission is performed through optical fibers or satellite links.

**Switching Offices:** As there is a permanent physical link between any two subscribers. To avoid this, the telephone company uses switches that are located in switching offices. A switch is able to connect various loops or trunks and allows a connection between different subscribes.



*Switching offices*

#### **Advantages of Telephone System:**

- It is a circuit-switched network.
- There is no transmission delay as any receiver can be selected.
- It is cheap in price because it is a widely spread network.

#### **Disadvantages of Telephone System:**

- It requires a large time for connection.
- It has a low transmission speed.

#### **Applications of Telephone System:**

- It helps to connect people.
- It is used by business organizations to advertise their products.
- It is also used around the world for recreational purposes.

## CELLULAR RADIO

Cellular radio is an underlying technology for mobile phones, personal communication systems, wireless networking etc. The technology is developed for mobile radio telephone to replace high power transmitter/receiver systems. Cellular networks use lower power, shorter range and more transmitters for data transmission.

### **Features of Cellular Systems**

Wireless Cellular Systems solves the problem of spectral congestion and increases user capacity. The features of cellular systems are as follows –

- Offer very high capacity in a limited spectrum.
- Reuse of radio channel in different cells.
- Enable a fixed number of channels to serve an arbitrarily large number of users by reusing the channel throughout the coverage region.
- Communication is always between mobile and base station (not directly between mobiles).
- Each cellular base station is allocated a group of radio channels within a small geographic area called a cell.
- Neighboring cells are assigned different channel groups.
- By limiting the coverage area to within the boundary of the cell, the channel groups may be reused to cover different cells.
- Keep interference levels within tolerable limits.
- Frequency reuse or frequency planning.
- Organization of Wireless Cellular Network.

Cellular network is organized into multiple low power transmitters each 100w or less.

### **Shape of Cells**

The coverage area of cellular networks are divided into **cells**, each cell having its own antenna for transmitting the signals. Each cell has its own frequencies. Data communication in cellular networks is served by its base station transmitter, receiver and its control unit.

The shape of cells can be either square or hexagon –

#### **Square**

A square cell has four neighbors at distance **d** and four at distance **Root 2 d**

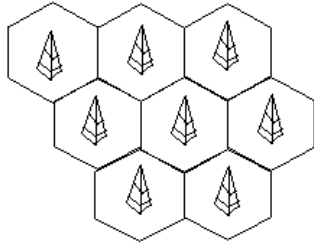
- Better if all adjacent antennas equidistant
- Simplifies choosing and switching to new antenna

#### **Hexagon**

A hexagon cell shape is highly recommended for its easy coverage and calculations. It offers the following advantages –



- Provides equidistant antennas
- Distance from center to vertex equals length of side



## Frequency Reuse

Frequency reusing is the concept of using the same radio frequencies within a given area, that are separated by considerable distance, with minimal interference, to establish communication.

Frequency reuse offers the following benefits –

- Allows communications within cell on a given frequency
- Limits escaping power to adjacent cells
- Allows re-use of frequencies in nearby cells
- Uses same frequency for multiple conversations
- 10 to 50 frequencies per cell

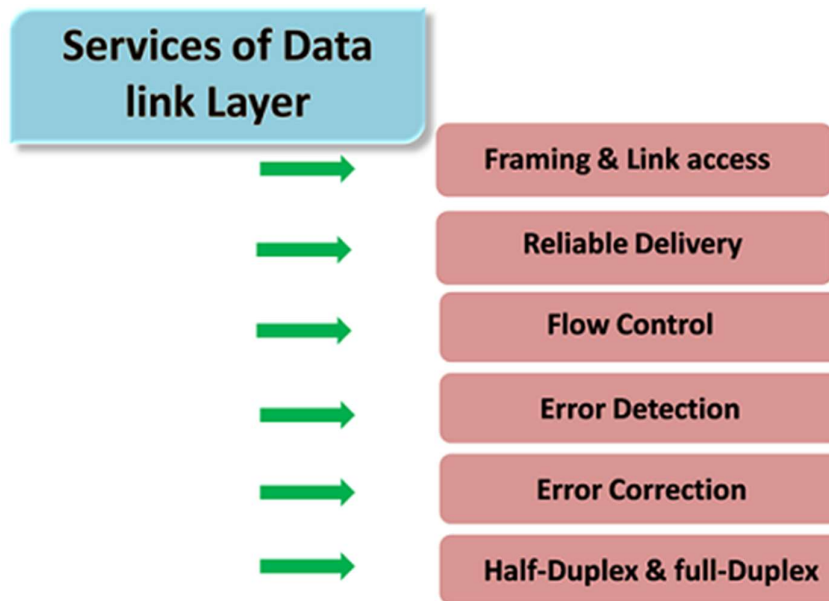
For example, when  $N$  cells are using the same number of frequencies and  $K$  be the total number of frequencies used in systems. Then each cell frequency is calculated by using the formulae  $K/N$ .

In Advanced Mobile Phone Services (AMPS) when  $K = 395$  and  $N = 7$ , then frequencies per cell on an average will be  $395/7 = 56$ . Here, cell frequency is 56.

Data Link Layer

- In the OSI model, the data link layer is a 4<sup>th</sup> layer from the top and 2<sup>nd</sup> layer from the bottom.
- The communication channel that connects the adjacent nodes is known as links, and in order to move the datagram from source to the destination, the datagram must be moved across an individual link.
- The main responsibility of the Data Link Layer is to transfer the datagram across an individual link.
- The Data link layer protocol defines the format of the packet exchanged across the nodes as well as the actions such as Error detection, retransmission, flow control, and random access.
- The Data Link Layer protocols are Ethernet, token ring, FDDI and PPP.
- An important characteristic of a Data Link Layer is that datagram can be handled by different link layer protocols on different links in a path. For example, the datagram is handled by Ethernet on the first link, PPP on the second link.

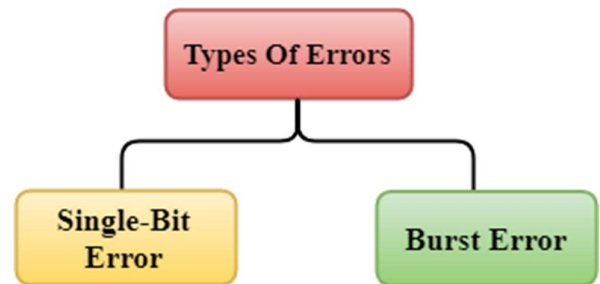
Following services are provided by the Data Link Layer:



- **Framing & Link access:** Data Link Layer protocols encapsulate each network frame within a Link layer frame before the transmission across the link. A frame consists of a data field in which network layer datagram is inserted and a number of data fields. It specifies the structure of the frame as well as a channel access protocol by which frame is to be transmitted over the link.
- **Reliable delivery:** Data Link Layer provides a reliable delivery service, i.e., transmits the network layer datagram without any error. A reliable delivery service is accomplished with transmissions and acknowledgements. A data link layer mainly provides the reliable delivery service over the links as they have higher error rates and they can be corrected locally, link at which an error occurs rather than forcing to retransmit the data.
- **Flow control:** A receiving node can receive the frames at a faster rate than it can process the frame. Without flow control, the receiver's buffer can overflow, and frames can get lost. To overcome this

problem, the data link layer uses the flow control to prevent the sending node on one side of the link from overwhelming the receiving node on another side of the link.

- **Error detection:** Errors can be introduced by signal attenuation and noise. Data Link Layer protocol provides a mechanism to detect one or more errors. This is achieved by adding error detection bits in the frame and then receiving node can perform an error check.
- **Error correction:** Error correction is similar to the Error detection, except that receiving node not only detect the errors but also determine where the errors have occurred in the frame.
- **Half-Duplex & Full-Duplex:** In a Full-Duplex mode, both the nodes can transmit the data at the same time. In a Half-Duplex mode, only one node can transmit the data at the same time.



### Error Detection

When data is transmitted from one device to another device, the system does not guarantee whether the data received by the device is identical to the data transmitted by another device. An Error is a situation when the message received at the receiver end is not identical to the message transmitted.

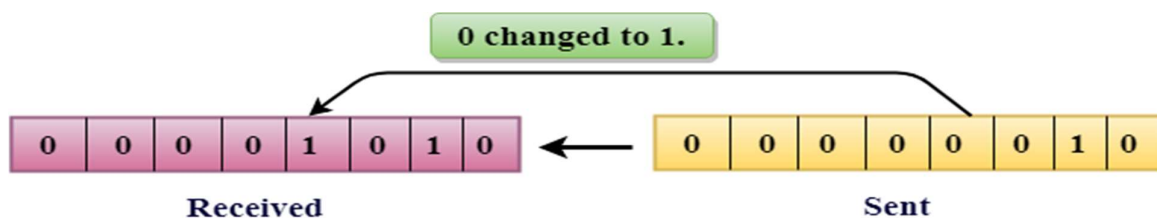
### Types Of Errors

Errors can be classified into two categories:

- Single-Bit Error
- Burst Error

### Single-Bit Error:

The only one bit of a given data unit is changed from 1 to 0 or from 0 to 1.



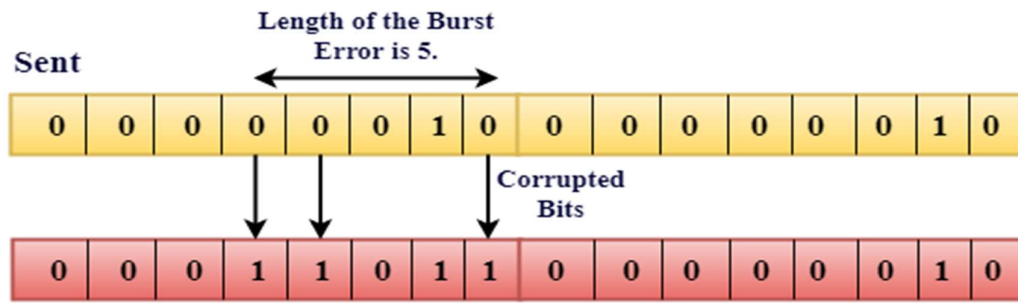
In the above figure, the message which is sent is corrupted as single-bit, i.e., 0 bit is changed to 1.

**Single-Bit Error** does not appear more likely in Serial Data Transmission. For example, Sender sends the data at 10 Mbps, this means that the bit lasts only for 1 ?s and for a single-bit error to occurred, a noise must be more than 1 ?s.

Single-Bit Error mainly occurs in Parallel Data Transmission. For example, if eight wires are used to send the eight bits of a byte, if one of the wire is noisy, then single-bit is corrupted per byte.

### Burst Error:

- The two or more bits are changed from 0 to 1 or from 1 to 0 is known as Burst Error.
- The Burst Error is determined from the first corrupted bit to the last corrupted bit.



### **Received**

- The duration of noise in Burst Error is more than the duration of noise in Single-Bit.
- Burst Errors are most likely to occur in Serial Data Transmission.
- The number of affected bits depends on the duration of the noise and data rate.

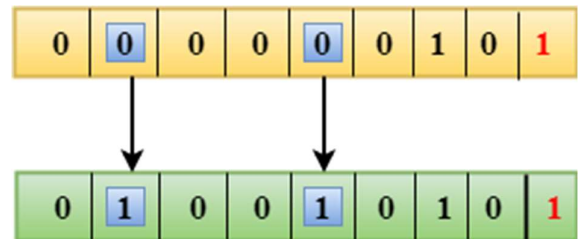
### Error Detecting Techniques:

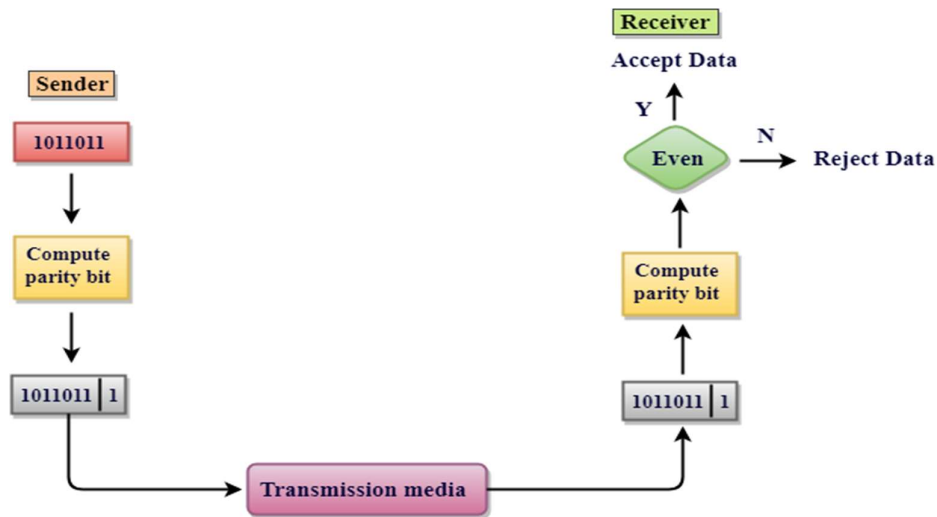
The most popular Error Detecting Techniques are:

- Single parity check
- Two-dimensional parity check
- Checksum
- Cyclic redundancy check

### Single Parity Check

- Single Parity checking is the simple mechanism and inexpensive to detect the errors.
- In this technique, a redundant bit is also known as a parity bit which is appended at the end of the data unit so that the number of 1s becomes even. Therefore, the total number of transmitted bits would be 9 bits.
- If the number of 1s bits is odd, then parity bit 1 is appended and if the number of 1s bits is even, then parity bit 0 is appended at the end of the data unit.
- At the receiving end, the parity bit is calculated from the received data bits and compared with the received parity bit.
- This technique generates the total number of 1s even, so it is known as even-parity checking.



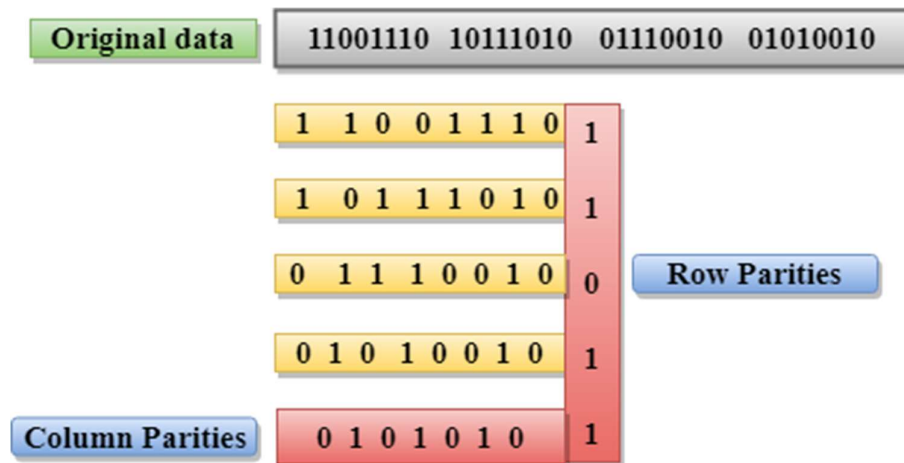


### Drawbacks of Single Parity Checking

- It can only detect single-bit errors which are very rare.
- If two bits are interchanged, then it cannot detect the errors.

### Two-Dimensional Parity Check

- Performance can be improved by using **Two-Dimensional Parity Check** which organizes the data in the form of a table.
- Parity check bits are computed for each row, which is equivalent to the single-parity check.
- In Two-Dimensional Parity check, a block of bits is divided into rows, and the redundant row of bits is added to the whole block.
- At the receiving end, the parity bits are compared with the parity bits computed from the received data.



### Drawbacks Of 2D Parity Check

- If two bits in one data unit are corrupted and two bits exactly the same position in another data unit are also corrupted, then 2D Parity checker will not be able to detect the error.
- This technique cannot be used to detect the 4-bit errors or more in some cases.

### Checksum

A Checksum is an error detection technique based on the concept of redundancy.

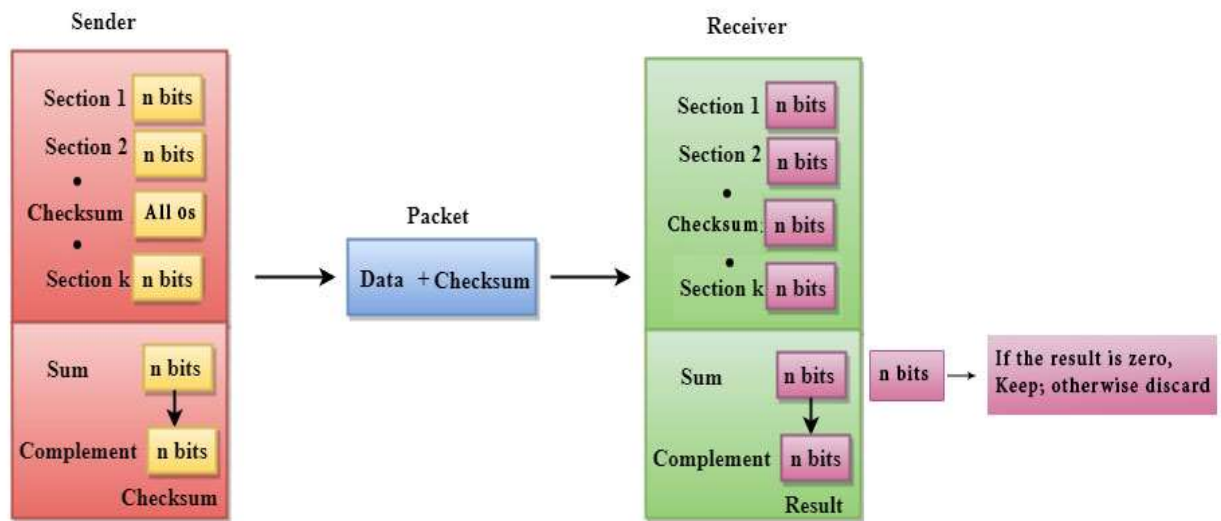
**It is divided into two parts:**

- ✓ Checksum Generator
- ✓ Checksum Checker

### Checksum Generator

A Checksum is generated at the sending side. Checksum generator subdivides the data into equal segments of  $n$  bits each, and all these segments are added together by using one's complement arithmetic. The sum is complemented and appended to the original data, known as checksum field. The extended data is transmitted across the network.

Suppose  $L$  is the total sum of the data segments, then the checksum would be  $?L$



1. The Sender follows the given steps:
2. The block unit is divided into  $k$  sections, and each of  $n$  bits.
3. All the  $k$  sections are added together by using one's complement to get the sum.
4. The sum is complemented and it becomes the checksum field.
5. The original data and checksum field are sent across the network.

### Checksum Checker

A Checksum is verified at the receiving side. The receiver subdivides the incoming data into equal segments of  $n$  bits each, and all these segments are added together, and then this sum is complemented. If the complement of the sum is zero, then the data is accepted otherwise data is rejected.

1. The Receiver follows the given steps:
2. The block unit is divided into  $k$  sections and each of  $n$  bits.
3. All the  $k$  sections are added together by using one's complement algorithm to get the sum.
4. The sum is complemented.

5. If the result of the sum is zero, then the data is accepted otherwise the data is discarded.

### Cyclic Redundancy Check (CRC)

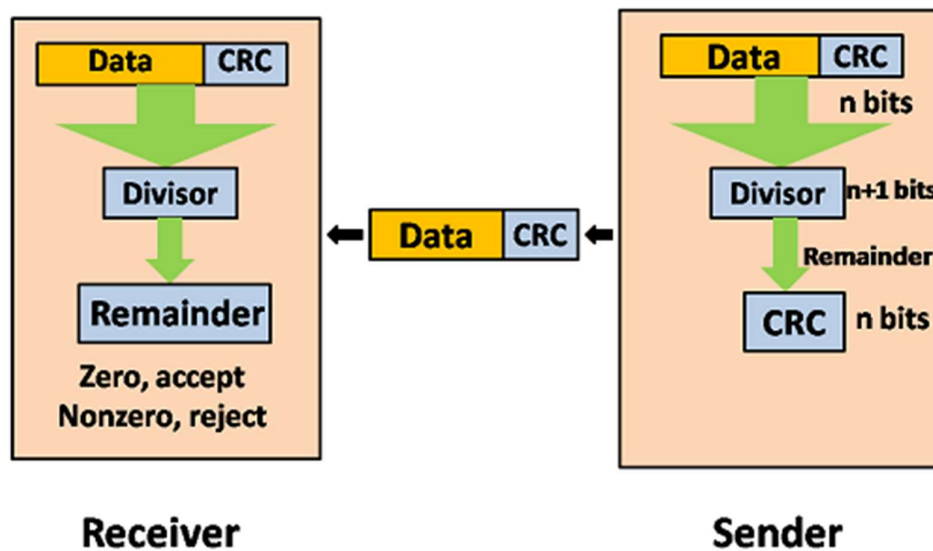
CRC is a redundancy error technique used to determine the error.

#### **Following are the steps used in CRC for error detection:**

- In CRC technique, a string of  $n$  0s is appended to the data unit, and this  $n$  number is less than the number of bits in a predetermined number, known as division which is  $n+1$  bits.
- Secondly, the newly extended data is divided by a divisor using a process known as binary division. The remainder generated from this division is known as CRC remainder.
- Thirdly, the CRC remainder replaces the appended 0s at the end of the original data. This newly generated unit is sent to the receiver.
- The receiver receives the data followed by the CRC remainder. The receiver will treat this whole unit as a single unit, and it is divided by the same divisor that was used to find the CRC remainder.

If the resultant of this division is zero which means that it has no error, and the data is accepted.

If the resultant of this division is not zero which means that the data consists of an error. Therefore, the data is discarded.



Let's understand this concept through an example:

**Suppose the original data is 11100 and divisor is 1001.**

#### **CRC Generator**

- A CRC generator uses a modulo-2 division. Firstly, three zeroes are appended at the end of the data as the length of the divisor is 4 and we know that the length of the string 0s to be appended is always one less than the length of the divisor.
- Now, the string becomes 11100000, and the resultant string is divided by the divisor 1001.
- The remainder generated from the binary division is known as CRC remainder. The generated value of the CRC remainder is 111.
- CRC remainder replaces the appended string of 0s at the end of the data unit, and the final string would be 11100111 which is sent across the network.





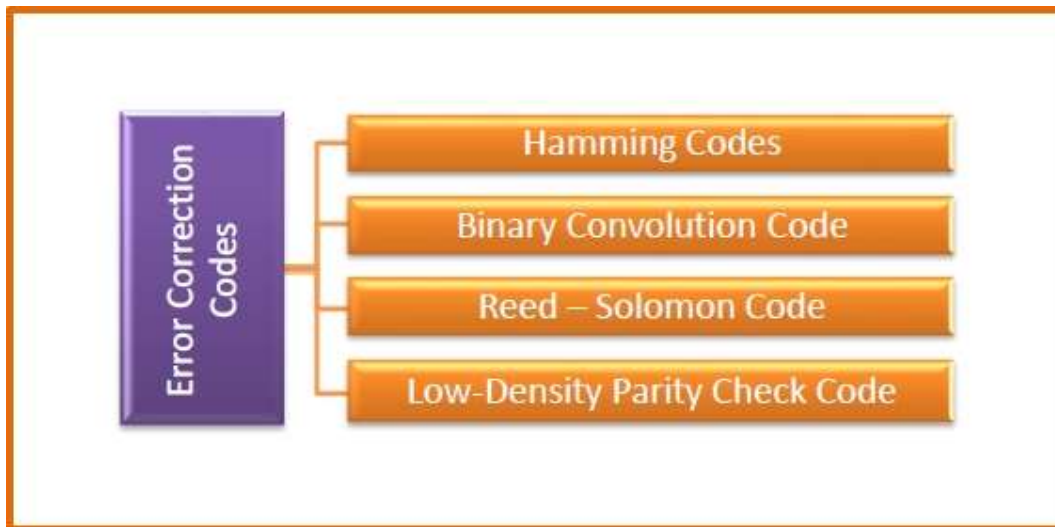
## Types of Error Correcting Codes

ECCs can be broadly categorized into two types, block codes and convolution codes.

- **Block codes** – The message is divided into fixed-sized blocks of bits, to which redundant bits are added for error detection or correction.
- **Convolutional codes** – The message comprises of data streams of arbitrary length and parity symbols are generated by the sliding application of a Boolean function to the data stream.

## Common Error Correcting Codes

There are four popularly used error correction codes.



- **Hamming Codes** – It is a block code that is capable of detecting up to two simultaneous bit errors and correcting single-bit errors.
- **Binary Convolution Code** – Here, an encoder processes an input sequence of bits of arbitrary length and generates a sequence of output bits.
- **Reed - Solomon Code** – They are block codes that are capable of correcting burst errors in the received data block.
- **Low-Density Parity Check Code** – It is a block code specified by a parity-check matrix containing a low density of 1s. They are suitable for large block sizes in very noisy channels.

## Hamming Code

Hamming code is a block code that is capable of detecting up to two simultaneous bit errors and correcting single-bit errors. It was developed by R.W. Hamming for error correction.

In this coding method, the source encodes the message by inserting redundant bits within the message. These redundant bits are extra bits that are generated and inserted at specific positions in the message itself to enable error detection and correction. When the destination receives this message, it performs recalculations to detect errors and find the bit position that has error.

Encoding a message by Hamming Code

The procedure used by the sender to encode the message encompasses the following steps –

- **Step 1** – Calculation of the number of redundant bits.

- **Step 2** – Positioning the redundant bits.
- **Step 3** – Calculating the values of each redundant bit.

Once the redundant bits are embedded within the message, this is sent to the user.

Step 1 – Calculation of the number of redundant bits.

If the message contains  $m$  number of data bits,  $r$  number of redundant bits are added to it so that  $m+r$  is able to indicate at least  $(m+r+1)$  different states. Here,  $(m+r)$  indicates location of an error in each of  $(m+r)$  bit positions and one additional state indicates no error. Since,  $r$  bits can indicate  $2^r$  states,  $2^r$  must be at least equal to  $(m+r+1)$ . Thus the following equation should hold  $2^r \geq m+r+1$

Step 2 – Positioning the redundant bits.

The  $r$  redundant bits placed at bit positions of powers of 2, i.e. 1, 2, 4, 8, 16 etc. They are referred in the rest of this text as  $r_1$  (at position 1),  $r_2$  (at position 2),  $r_3$  (at position 4),  $r_4$  (at position 8) and so on.

Step 3 – Calculating the values of each redundant bit.

The redundant bits are parity bits. A parity bit is an extra bit that makes the number of 1s either even or odd.

The two types of parity are –

- **Even Parity** – Here the total number of bits in the message is made even.
- **Odd Parity** – Here the total number of bits in the message is made odd.

Each redundant bit,  $r_i$ , is calculated as the parity, generally even parity, based upon its bit position. It covers all bit positions whose binary representation includes a 1 in the  $i^{\text{th}}$  position except the position of  $r_i$ . Thus –

- $r_1$  is the parity bit for all data bits in positions whose binary representation includes a 1 in the least significant position excluding 1 (3, 5, 7, 9, 11 and so on)
- $r_2$  is the parity bit for all data bits in positions whose binary representation includes a 1 in the position 2 from right except 2 (3, 6, 7, 10, 11 and so on)
- $r_3$  is the parity bit for all data bits in positions whose binary representation includes a 1 in the position 3 from right except 4 (5-7, 12-15, 20-23 and so on)

Decoding a message in Hamming Code

Once the receiver gets an incoming message, it performs recalculations to detect errors and correct them. The steps for recalculation are –

- **Step 1** – Calculation of the number of redundant bits.
- **Step 2** – Positioning the redundant bits.
- **Step 3** – Parity checking.
- **Step 4** – Error detection and correction

Step 1 – Calculation of the number of redundant bits

Using the same formula as in encoding, the number of redundant bits are ascertained.

$2^r \geq m + r + 1$  where  $m$  is the number of data bits and  $r$  is the number of redundant bits.

Step 2 – Positioning the redundant bits

The  $r$  redundant bits placed at bit positions of powers of 2, i.e. 1, 2, 4, 8, 16 etc.

Step 3 – Parity checking

Parity bits are calculated based upon the data bits and the redundant bits using the same rule as during generation of  $c_1, c_2, c_3, c_4$  etc. Thus

$c_1 = \text{parity}(1, 3, 5, 7, 9, 11 \text{ and so on})$

$c_2 = \text{parity}(2, 3, 6, 7, 10, 11 \text{ and so on})$

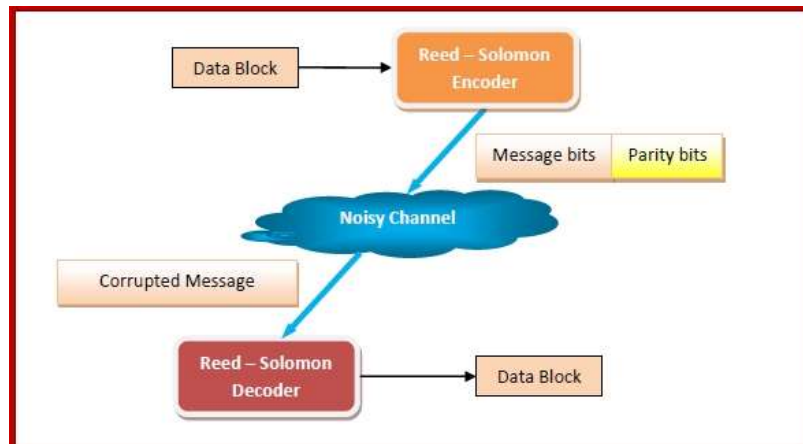
$c_3 = \text{parity}(4-7, 12-15, 20-23 \text{ and so on})$

Step 4 – Error detection and correction

The decimal equivalent of the parity bits binary values is calculated. If it is 0, there is no error. Otherwise, the decimal value gives the bit position which has error. For example, if  $c_1c_2c_3c_4 = 1001$ , it implies that the data bit at position 9, decimal equivalent of 1001, has error. The bit is flipped to get the correct message.

### Reed - Solomon Code

Reed - Solomon error correcting codes are one of the oldest codes that were introduced in 1960s by Irving S. Reed and Gustave Solomon. It is a subclass of non - binary BCH codes. BCH codes (Bose-Chaudhuri-Hocquenghem codes) are cyclic ECCs that are constructed using polynomials over data blocks. A Reed - Solomon encoder accepts a block of data and adds redundant bits (parity bits) before transmitting it over noisy channels. On receiving the data, a decoder corrects the error depending upon the code characteristics.



### Application Areas of Reed-Solomon Codes

The prominent application areas are –

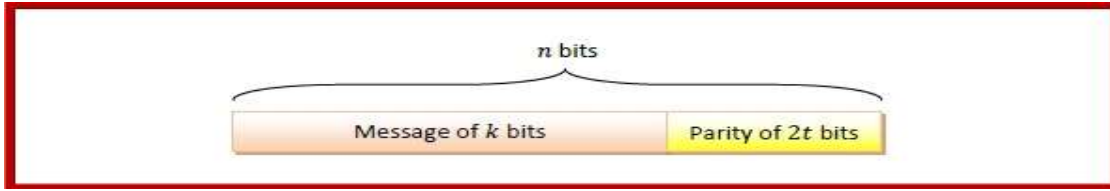
- Storage areas like CDs, DVDs, Blu-ray Discs
- High speed data transmission technologies such as DSL and WiMAX
- High speed modems
- QR Codes
- Broadcast systems such as satellite communications
- Storage systems such as RAID 6

Parameters of Reed - Solomon Codes

- A Reed-Solomon code is specified as  $RS(n, k)$ .
- Here,  $n$  is the block length which is recognizable by symbols, holding the relation,  $n = 2^m - 1$ .
- The message size is of  $k$  bits.
- So the parity check size is  $(n - k)$  bits

- The code can correct up to  $(t)$  errors in a codeword, where  $(2t = n - k)$ .

The following diagram shows a Reed-Solomon codeword –



### Generator Polynomial of Reed Solomon Code

In coding systems with block codes, valid code words consists of polynomials that are divisible by another fixed polynomial of short length. This fixed polynomial is called generator polynomial.

In Reed Solomon code, generator polynomial with factors is constructed where each root is a consecutive element in the Galois field. The polynomial is of the form –

$$g(x) = (x - \alpha) (x - \alpha^2) (x - \alpha^3) \dots (x - \alpha^{2t}) \text{ where } \alpha \text{ is a primitive element.}$$

### Encoding using Reed Solomon Code

The method of encoding in Reed Solomon code has the following steps –

- The message is represented as a polynomial  $p(x)$ , and then multiplied with the generator polynomial  $g(x)$ .
- The message vector  $[x_1, x_2, x_3, \dots, x_k]$  is mapped to a polynomial of degree less than  $k$  such that  $p_x(\alpha_i) = x_i$  for all  $i = 1, \dots, k$
- The polynomial is evaluated using interpolation methods like Lagrange Interpolation.
- Using this polynomial, the other points  $\alpha_{k+1}, \dots, \alpha_n$ , are evaluated.
- The encoded message is calculated as  $s(x) = p(x) * g(x)$ . The sender sends this encoded message along with the generator polynomial  $g(x)$ .

### Decoding using Reed Solomon Code

At the receiving end, the following decoding procedure done –

- The receiver receives the message  $r(x)$  and divides it by the generator polynomial  $g(x)$ .
- If  $r(x)/g(x)=0$ , then it implies no error.
- If  $r(x)/g(x) \neq 0$ , then the error polynomial is evaluated using the expression:  $r(x) = p(x) * g(x) + e(x)$
- The error polynomial gives the error positions.

### Binary Convolutional Codes

In convolutional codes, the message comprises of data streams of arbitrary length and a sequence of output bits are generated by the sliding application of Boolean functions to the data stream.

In block codes, the data comprises of a block of data of a definite length. However, in convolutional codes, the input data bits are not divided into block but are instead fed as streams of data bits, which convolve to output bits based upon the logic function of the encoder. Also, unlike block codes, where the output codeword is dependent only on the present inputs, in convolutional codes, output stream depends not only the present input bits but also only previous input bits stored in memory.

Convolutional codes were first introduced in 1955, by Elias. After that, there were many interim researches by many mathematicians. In 1973, Viterbi developed an algorithm for maximum likelihood decoding scheme, called Viterbi scheme that lead to modern convolutional codes.

### Encoding by Convolutional Codes

For generating a convolutional code, the information is passed sequentially through a linear finite-state shift register. The shift register comprises of  $(K-1)$  stages and Boolean function generators.

A convolutional code can be represented as  $(n, k, K)$  where

- $k$  is the number of bits shifted into the encoder at one time. Generally,  $k = 1$ .
- $n$  is the number of encoder output bits corresponding to  $k$  information bits.
- The code-rate,  $R_c = k/n$ .
- The encoder memory, a shift register of size  $k$ , is the constraint length.
- $n$  is a function of the present input bits and the contents of  $K$ .
- The state of the encoder is given by the value of  $(K - 1)$  bits.

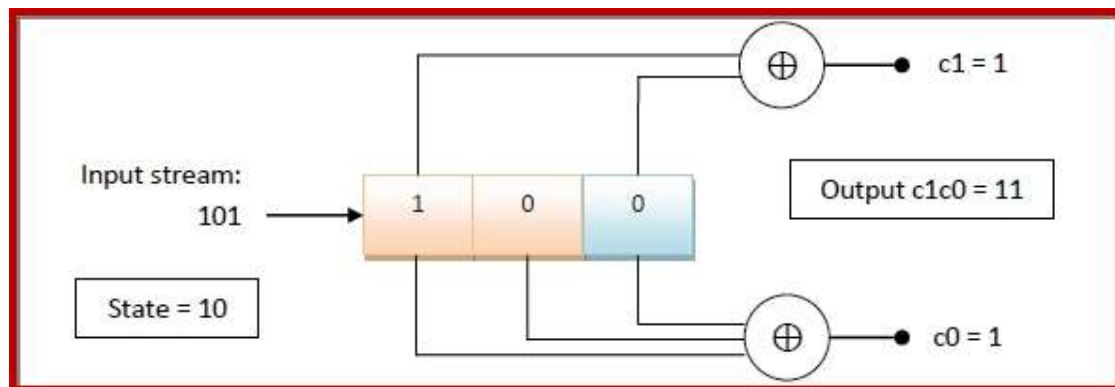
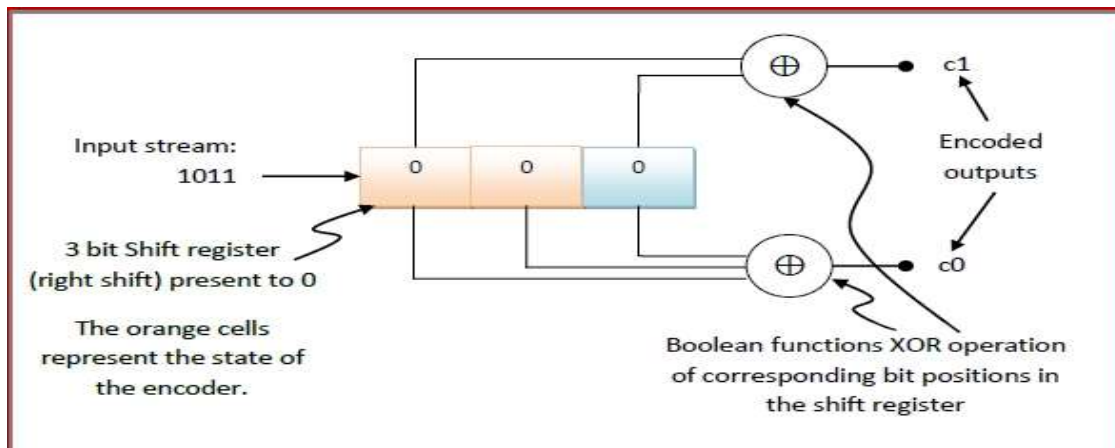
### Example of Generating a Convolutional Code

Let us consider a convolutional encoder with  $k = 1$ ,  $n = 2$  and  $K = 3$ .

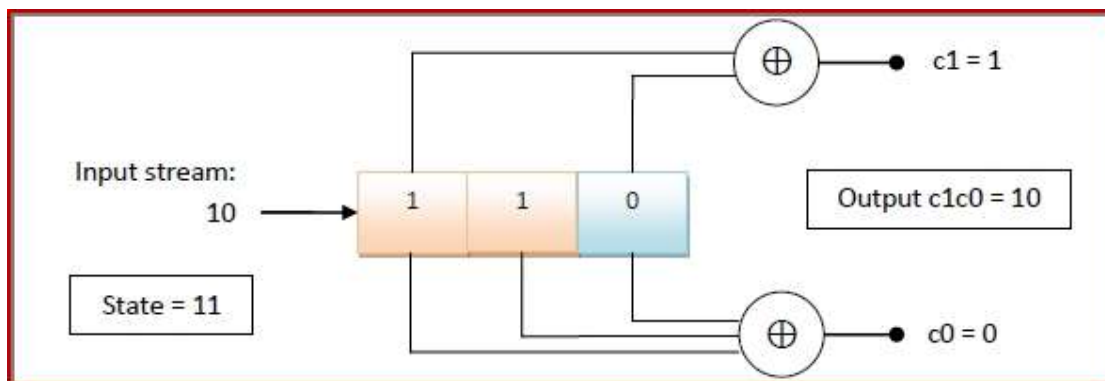
The code-rate,  $R_c = k/n = 1/2$ .

The input string is streamed from right to left into the encoder.

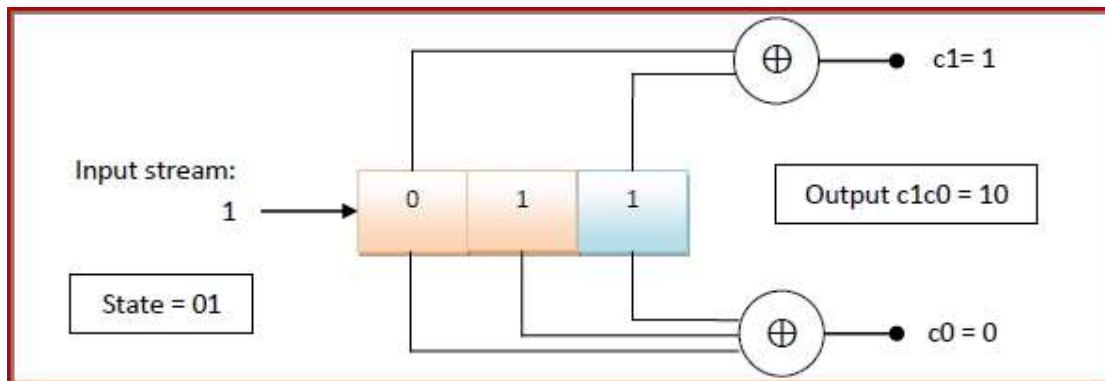
When the first bit, 1, is streamed in the encoder, the contents of encoder will be –



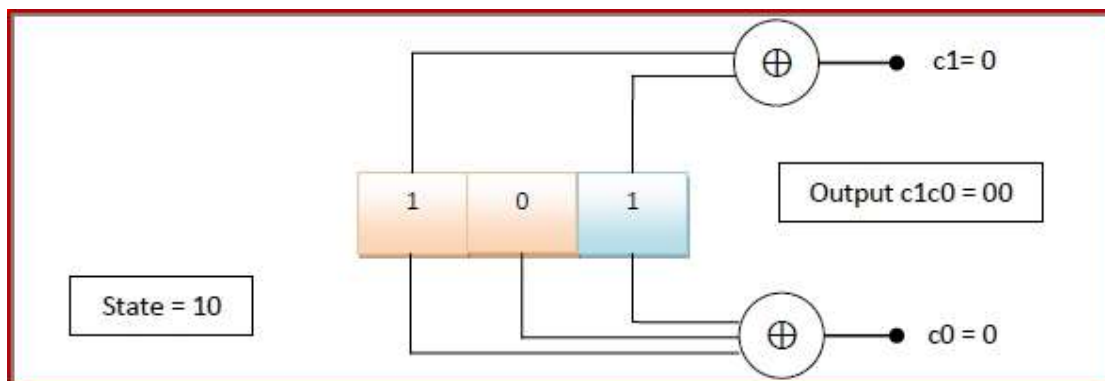
When the next bit, 1 is streamed in the encoder, the contents of encoder will be –



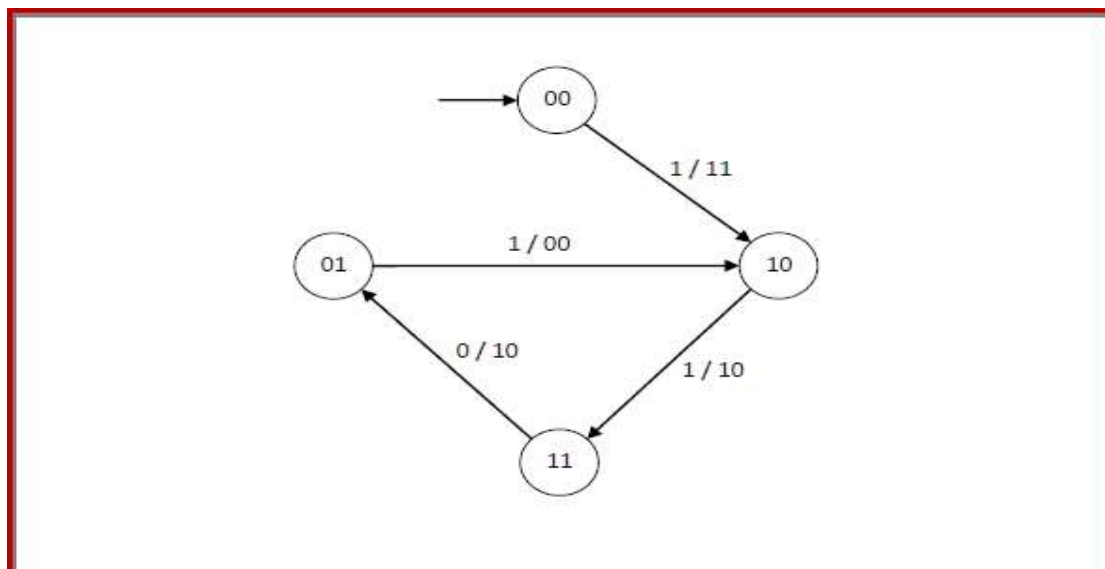
When the next bit, 0 is streamed in the encoder, the contents of encoder will be –



When the last bit, 1 is streamed in the encoder, the contents of encoder will be –



The corresponding state transition diagram will be –





## Low-Density Parity Check Codes

### Encoding by Low-Density Parity Check Codes

A low - density parity check (LDPC) code is specified by a parity-check matrix containing mostly 0s and a low density of 1s. The rows of the matrix represent the equations and the columns represent the bits in the code word, i.e. code symbols.

A LDPC code is represented by , where is the block length, is the number of 1s in each column and is the number of 1s in each row, holding the following properties –

- $j$  is the small fixed number of 1's in each column, where  $j > 3$
- $k$  is the small fixed number of 1's in each row, where  $k > j$ .

#### Example 1 – Parity Check Matrix of Hamming Code

The following parity check matrix Hamming code having , with 4 information bits followed by 3 even parity bits. The check digits are diagonally 1. The parity equations are given alongside –

Information Bits				Parity Bits			
$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$x_6$	$x_7$	
1	1	1	0	1	0	0	$x_5 = x_1 + x_2 + x_3$ <b>OR</b> $x_1 + x_2 + x_3 + x_5 = 0$
1	1	0	1	0	1	0	$x_6 = x_1 + x_2 + x_4$ <b>OR</b> $x_1 + x_2 + x_4 + x_6 = 0$
1	0	1	1	0	0	1	$x_7 = x_1 + x_3 + x_4$ <b>OR</b> $x_1 + x_3 + x_4 + x_7 = 0$

#### Example 2 – Low - Density Parity Check Matrix

This examples illustrates an (12, 3, 4) LDPC matrix, i.e.  $n = 12$ ,  $j = 3$  and  $k = 4$ . This implies that each equation operates on 4 code symbols and each code symbol appears in 3 equations. Unlike parity check matrix of the Hamming code, this code does not have any diagonal 1s in parity bits.

$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$x_6$	$x_7$	$x_8$	$x_9$	$x_{10}$	$x_{11}$	$x_{12}$	Parity Equations
0	0	1	0	0	1	1	1	0	0	0	0	$x_3 + x_6 + x_7 + x_8 = 0$
1	1	0	0	1	0	0	0	0	0	0	1	$x_1 + x_2 + x_5 + x_{12} = 0$
0	0	0	1	0	0	0	0	1	1	1	0	$x_4 + x_9 + x_{10} + x_{11} = 0$
0	1	0	0	0	1	1	0	0	1	0	0	$x_2 + x_6 + x_7 + x_{10} = 0$
1	0	1	0	0	0	0	1	0	0	1	0	$x_1 + x_3 + x_8 + x_{11} = 0$
0	0	0	1	1	0	0	0	1	0	0	1	$x_4 + x_5 + x_9 + x_{12} = 0$
1	0	0	1	1	0	1	0	0	0	0	0	$x_1 + x_4 + x_5 + x_7 = 0$
0	0	0	0	0	1	0	1	0	0	1	1	$x_6 + x_8 + x_{11} + x_{12} = 0$
0	1	1	0	0	0	0	0	1	1	0	0	$x_2 + x_3 + x_9 + x_{10} = 0$

### Decoding of LDPC Codes

There are two possible decoding techniques of LDPC codes –

- In the first technique, the decoder does all the parity checks as per the parity equations. If any bit is contained in more than a fixed number of unsatisfied parity equations, the value of that bit is reversed.



Once the new values are obtained, parity equations are recomputed using the new values. The procedure is repeated until all the parity equations are satisfied.

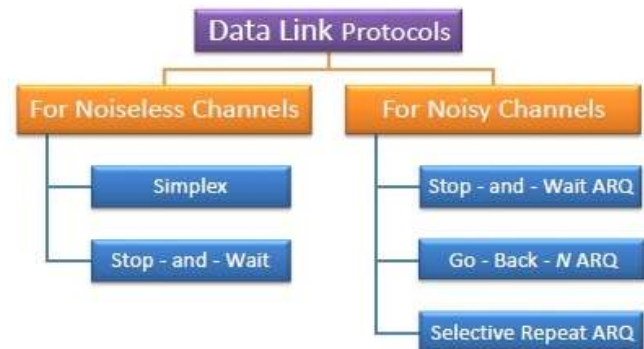
This decoding procedure is simple and but is applicable only when the parity-check sets are small.

- The second method performs probabilistic algorithms on LDPC graphs. The graph is a sparse bipartite graph that contains two sets of nodes, one set representing the parity equations and the other set representing the code symbols. A line connects node in first set to the second if a code symbol is present in the equation. Decoding is done by passing messages along the lines of the graph. Messages are passed from message nodes to check nodes, and from check nodes back to message nodes and their parity values are calculated.

The two subclasses of these methods are belief propagation and maximum likelihood decoding. Though these decoding algorithms are complex, they yield better results than the former.

#### Elementary Data Link Protocols:

Protocols in the data link layer are designed so that this layer can perform its basic functions: framing, error control and flow control. Framing is the process of dividing bit - streams from physical layer into data frames whose size ranges from a few hundred to a few thousand bytes. Error control mechanisms deals with transmission errors and retransmission of corrupted and lost frames. Flow control regulates speed of delivery and so that a fast sender does not drown a slow receiver.



#### Types of Data Link Protocols

Data link protocols can be broadly divided into two categories, depending on whether the transmission channel is noiseless or noisy.

##### Simplex Protocol

The Simplex protocol is hypothetical protocol designed for unidirectional data transmission over an ideal channel, i.e. a channel through which transmission can never go wrong. It has distinct procedures for sender and receiver. The sender simply sends all its data available onto the channel as soon as they are available its buffer. The receiver is assumed to process all incoming data instantly. It is hypothetical since it does not handle flow control or error control.

##### Stop – and – Wait Protocol

Stop – and – Wait protocol is for noiseless channel too. It provides unidirectional data transmission without any error control facilities. However, it provides for flow control so that a fast sender does not drown a slow receiver. The receiver has a finite buffer size with finite processing speed. The sender can send a frame only when it has received indication from the receiver that it is available for further data processing.

##### Stop – and – Wait ARQ

Stop – and – wait Automatic Repeat Request (Stop – and – Wait ARQ) is a variation of the above protocol with added error control mechanisms, appropriate for noisy channels. The sender keeps a copy of the sent frame. It then waits for a finite time to receive a positive acknowledgement from receiver. If the timer expires or a negative acknowledgement is received, the frame is retransmitted. If a positive acknowledgement is received then the next frame is sent.

#### Go – Back – N ARQ

Go – Back – N ARQ provides for sending multiple frames before receiving the acknowledgement for the first frame. It uses the concept of sliding window, and so is also called sliding window protocol. The frames are sequentially numbered and a finite number of frames are sent. If the acknowledgement of a frame is not received within the time period, all frames starting from that frame are retransmitted.

#### Selective Repeat ARQ

This protocol also provides for sending multiple frames before receiving the acknowledgement for the first frame. However, here only the erroneous or lost frames are retransmitted, while the good frames are received and buffered.

#### Sliding Window Protocols:

The sliding window is a technique for sending multiple frames at a time. It controls the data packets between the two devices where reliable and gradual delivery of data frames is needed. It is also used in TCP (Transmission Control Protocol).

In this technique, each frame has sent from the sequence number. The sequence numbers are used to find the missing data in the receiver end. The purpose of the sliding window technique is to avoid duplicate data, so it uses the sequence number.

#### Types of Sliding Window Protocol

Sliding window protocol has three types:

1. A One-Bit Sliding Window Protocol.
2. Go-Back-N ARQ
3. Selective Repeat ARQ

#### **A One-Bit Sliding Window Protocol**

Sliding window protocols are data link layer protocols for reliable and sequential delivery of data frames. The sliding window is also used in Transmission Control Protocol. In these protocols, the sender has a buffer called the sending window and the receiver has buffer called the receiving window.

In one – bit sliding window protocol, the size of the window is 1. So the sender transmits a frame, waits for its acknowledgment, then transmits the next frame. Thus it uses the concept of stop and waits for the protocol. This protocol provides for full – duplex communications. Hence, the acknowledgment is attached along with the next data frame to be sent by piggybacking.

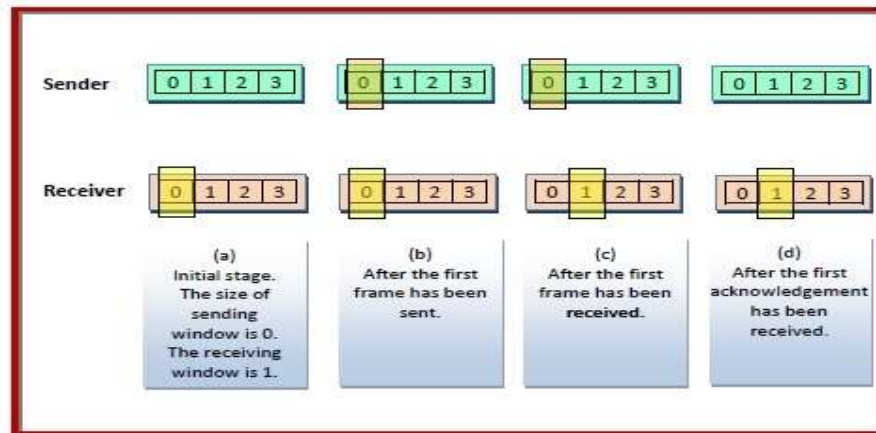
#### Working Principle

The data frames to be transmitted additionally have an acknowledgment field, *ack* field that is of a few bits length. The *ack* field contains the sequence number of the last frame received without error. If this

sequence number matches with the sequence number of the frame to be sent, then it is inferred that there is no error and the frame is transmitted. Otherwise, it is inferred that there is an error in the frame and the previous frame is retransmitted. Since this is a bi-directional protocol, the same algorithm applies to both the communicating parties.

### Illustrative Example

The following diagram depicts a scenario with sequence numbers 0, 1, 2, 3, 0, 1, 2 and so on. It depicts the sliding windows in the sending and the receiving stations during frame transmission.



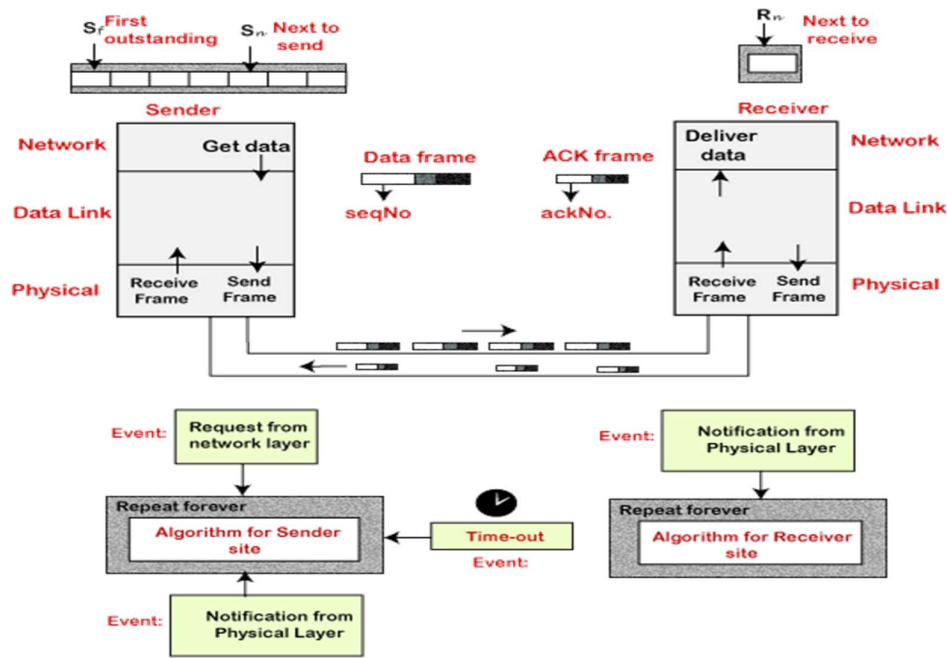
### Go-Back-N ARQ

Go-Back-N ARQ protocol is also known as Go-Back-N Automatic Repeat Request. It is a data link layer protocol that uses a sliding window method. In this, if any frame is corrupted or lost, all subsequent frames have to be sent again.

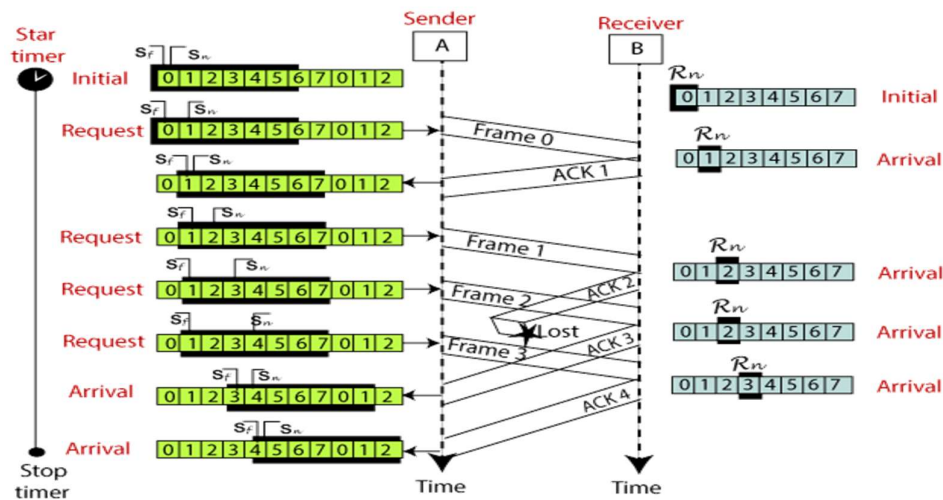
The size of the sender window is N in this protocol. For example, Go-Back-8, the size of the sender window, will be 8. The receiver window size is always 1.

If the receiver receives a corrupted frame, it cancels it. The receiver does not accept a corrupted frame. When the timer expires, the sender sends the correct frame again.

The design of the Go-Back-N ARQ protocol is shown below.



The example of Go-Back-N ARQ is shown below in the figure.

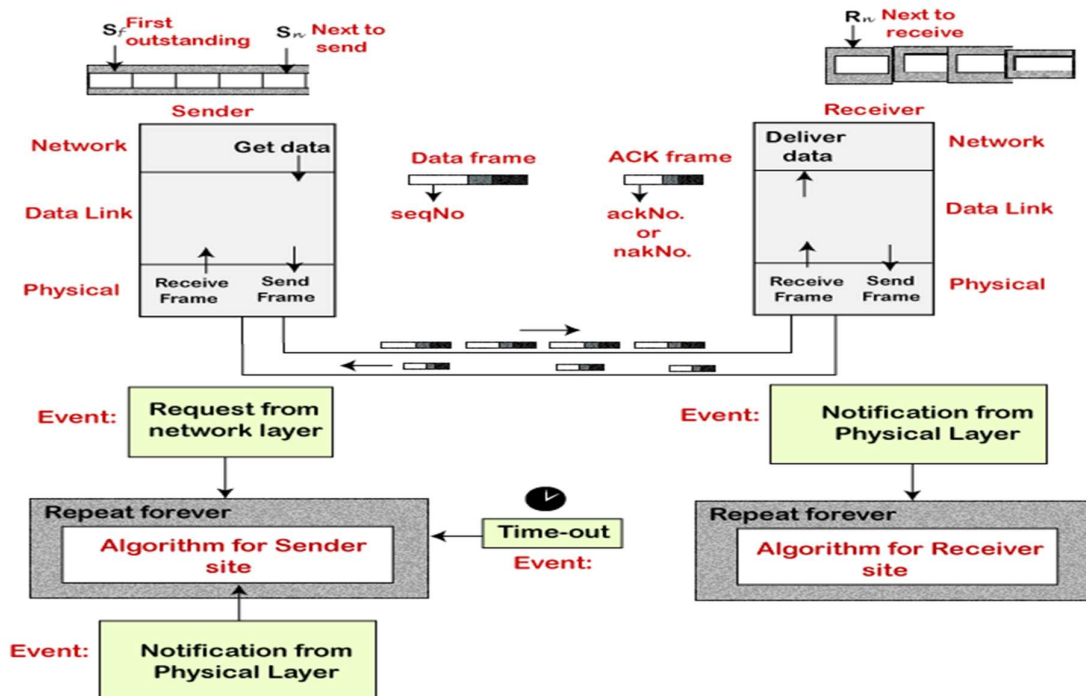


### Selective Repeat ARQ

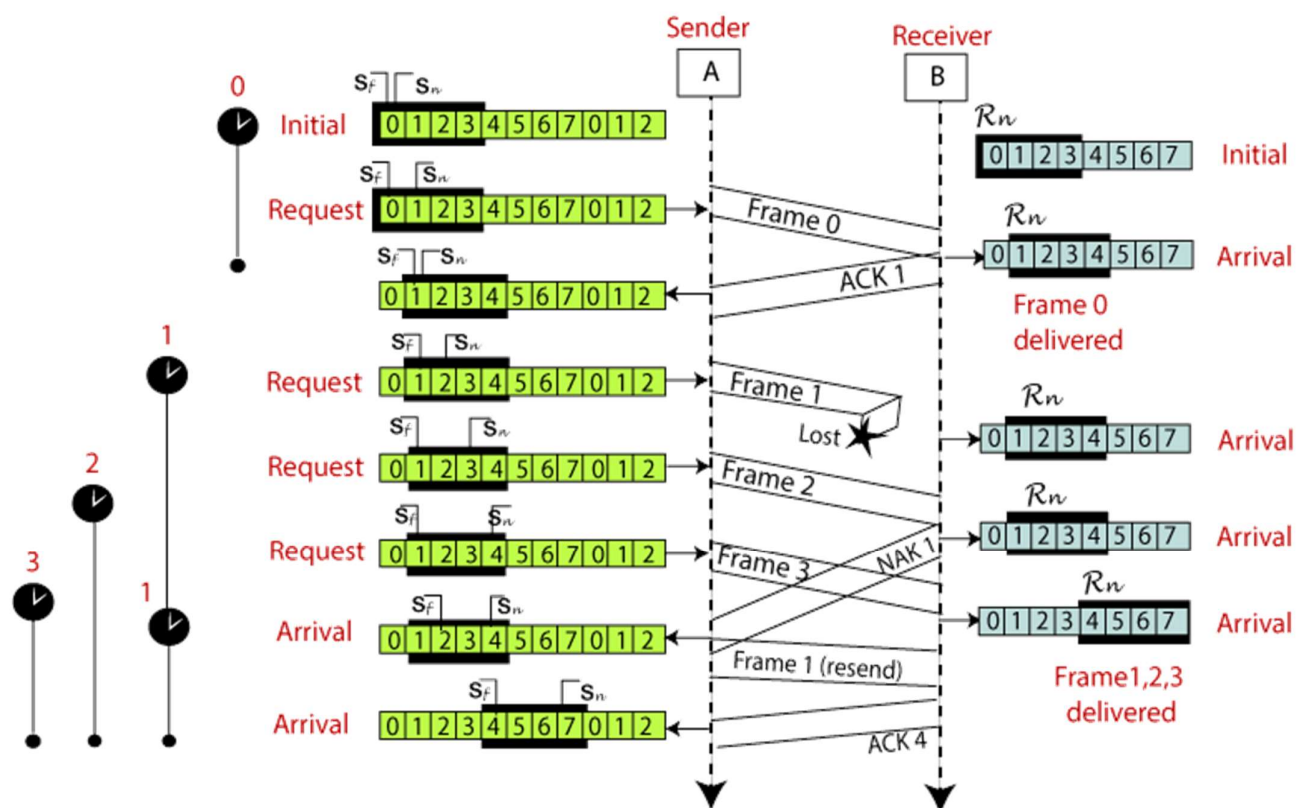
Selective Repeat ARQ is also known as the Selective Repeat Automatic Repeat Request. It is a data link layer protocol that uses a sliding window method. The Go-back-N ARQ protocol works well if it has fewer errors. But if there is a lot of error in the frame, lots of bandwidth loss in sending the frames again. So, we use the Selective Repeat ARQ protocol. In this protocol, the size of the sender window is always equal to the size of the receiver window. The size of the sliding window is always greater than 1.

If the receiver receives a corrupt frame, it does not directly discard it. It sends a negative acknowledgment to the sender. The sender sends that frame again as soon as on the receiving negative acknowledgment. There is no waiting for any time-out to send that frame.

The design of the Selective Repeat ARQ protocol is shown below.



The example of the Selective Repeat ARQ protocol is shown below in the figure.



Difference between the Go-Back-N ARQ and Selective Repeat ARQ

Go-Back-N ARQ	Selective Repeat ARQ
If a frame is corrupted or lost in it, all subsequent frames have to be sent again.	In this, only the frame is sent again, which is corrupted or lost.

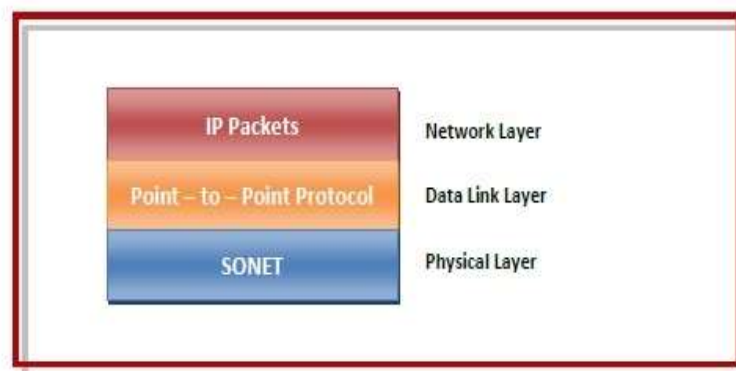
If it has a high error rate, it wastes a lot of bandwidth.	There is a loss of low bandwidth.
It is less complex.	It is more complex because it has to do sorting and searching as well. And it also requires more storage.
It does not require sorting.	In this, sorting is done to get the frames in the correct order.
It does not require searching.	The search operation is performed in it.
It is used more.	It is used less because it is more complex.

### DATA LINK PROTOCOLS EXAMPLE:

#### SONET:

Synchronous optical networking (SONET) is a physical layer protocol for transmitting multiple digital bit streams over optical fiber links that form the backbone of the communication networks. Packet-over-SONET (POS) is a standard that maps IP packets into SONET frames. To implement this mechanism, Point – to – Point Protocol (PPP) runs on IP routers. Point – to – Point Protocol (PPP) is a data link layer protocol that is used to transmit data between two directly connected (point-to-point) computers. It is a byte-oriented protocol that is widely used in broadband communications having heavy loads and high speeds.

The following diagram shows the protocol stack of Packet over SONET (POS) –



#### Features provides by PPP in POS

- **Framing** – It encapsulates the datagram in a frame so that it can be transmitted over the specified physical layer. It delineates the beginning and end of the frames and provides for error detection.
- **Link Control Protocol (LCP)** – It is responsible for establishing, configuring, testing, maintaining and terminating links for transmission. It also imparts negotiation for set up of options and use of features by the two endpoints of the links.
- **Network Control Protocols (NCPs)** – These protocols are used for negotiating the parameters and facilities for the network layer. For every higher-layer protocol supported by PPP, one NCP is there.

## Application of POS

- For sending a large amount of network traffic over the Internet.
- For transmitting IP packets over Wide Area Networks (WANs).
- In resilient packet ring (RPR) standard.

### **The Point-to-Point Protocol:**

Point to Point protocol is used for various reasons like router to router traffic, error detection, supports multiple protocols, IP addresses can be negotiated during connection. PPP provides three features:

1. A framing method that without separates, one frame to another without complexity and also handles error detection.
2. A link control protocol is used for bringing lines up and down when required, negotiating options, testing and is called as LCP (Link Control Protocol). It supports synchronous and asynchronous circuits and bit-oriented and byte-oriented encodings.
3. A way to negotiate network-layer options in a way that is independent of the network layer protocol to be used. The method chosen is to have a different NCP (Network Control Protocol) for each network layer supported.

We will see what happens when a home user calling up an Internet service provider to make a home PC a temporary Internet host. PC calls the provider's router via a modem and when the router's modem has answered the phone then physical connection is made. PC sends the router a series of LCP packets in the payload field of one or more PPP frames.

After this a series of NCP packets are sent for the purpose to configure the network layer. The PC needs an IP address to run a TCP/IP protocol stack. Usually each Internet provider gets a block of IP and then dynamically assigns one to each newly attached PC for the duration of its login session because there are not much IP. If a provider owns x IP addresses, it can have up to z machines logged in simultaneously, but its total customer base may be many times that. The NCP for IP assigns the IP address.

Now, the PC is now an Internet host and can send and receive IP packets, just as hardwired hosts can. When the user is finished, NCP breaks the network layer connection and IP address is taken back. Then LCP shuts down the data link layer connection. Finally, the computer tells the modem to hang up the phone, releasing the physical layer connection

The PPP frame format was chosen to closely resemble the HDLC frame format. The major difference between PPP and HDLC is that PPP is character oriented instead of bit oriented. In particular, PPP uses byte stuffing on dial-up modem lines. Not only can PPP frames be sent over dial-up telephone lines, but they can also be sent over SONET.

### **The PPP full frame format for unnumbered mode operation:**

**BYTES            1                    1                    1                    1 or 2                    variable                    2 or 4                    1**

Flag 01111110	Address 11111111	Control 00000011	Protocol	payload	Checksum	Flag 01111110
------------------	---------------------	---------------------	----------	---------	----------	------------------



All frames in PPP frames starts with the standard HDLC flag byte (01111110), if it occurs within the payload field, it is byte stuffed. Address field is followed next, which is always set to the binary value 11111111 to indicate that all stations are to accept the frame. Using this value avoids the issue of having to assign data link addresses.

The Address field is followed by the Control field, the default value of which is 00000011. This value indicates an unnumbered frame. In other words, PPP does not provide reliable transmission using sequence numbers and acknowledgements as the default. Since the Address and Control fields are always constant in the default configuration, LCP provides the necessary mechanism for the two parties to negotiate an option to just omit them altogether and save 2 bytes per frame.

Fourth PPP field is the Protocol field which tells what kind of packet is in the Payload field. Protocols starting with a 1 bit are used to negotiate other protocols. These include LCP and a different NCP for each network layer protocol supported. The default size of the Protocol field is 2 bytes, but it can be negotiated down to 1 byte using LCP.

The Payload field is variable length, up to some negotiated maximum. If the length whose default length of 1500 bytes is used. Padding may follow the payload if need be. After the Payload field comes the Checksum field, which is normally 2 bytes, but a 4-byte checksum can be negotiated.

### Explanation of Method to bring Internet Lines Up and Down:

**A simplified phase diagram for bringing a line up and down:**

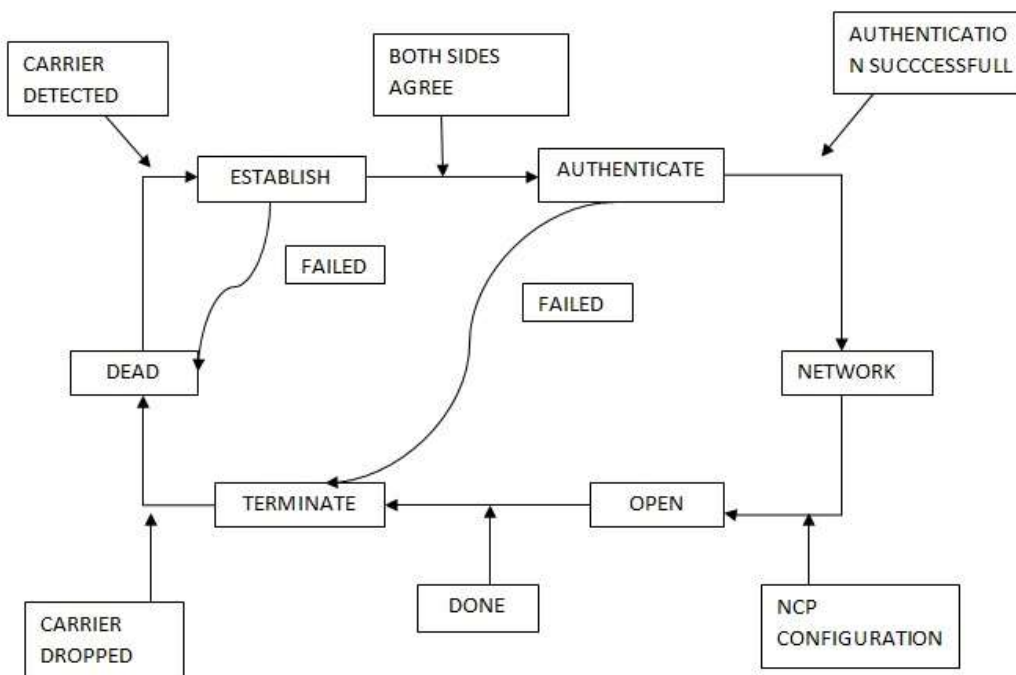


Figure shows the phases that a line goes through when it is brought up, used, and taken down again. This sequence applies both to modem connections and to router-router connections.

The DEAD state means that no physical layer carrier is present and no physical layer connection exists. Then the physical connection is ESTABLISHED. At that point LCP option negotiation begins, which, if successful, leads to AUTHENTICATE. Now the two parties can check on each other's identities if



desired. When the NETWORK phase is entered, the appropriate NCP protocol is invoked to configure the network layer. If the configuration is successful, OPEN is reached and data transport can take place. After data transport, the line moves into the TERMINATE phase, and from there, back to DEAD when the carrier is dropped.

During the ESTABLISH phase LCP negotiates data link protocol options. LCP provides a way for the initiating process to make an effort for the responding process to accept or reject it, in whole or in part, also line quality is verified for two processes to setup connection if line quality is good.. Finally, the LCP protocol also allows lines to be taken down when they are no longer needed.

LCP has several Frames. These are mentioned in the table below:

NAME	DIRECTION	DESCRIPTION
Configure-request	I→R	List of proposed options and values.
Configure-ack	I←R	All options are accepted
Configure-nak	I←R	Some options are not accepted.
Configure-reject	I←R	Some options are not negotiable.
Terminate-request	I→R	Request to shut the line down.
Terminate-ack	I←R	OK, line shut down
Code-reject	I←R	Unknown request received.
Protocol-reject	I←R	Unknown protocol requested.
Echo-request	I→R	Please send this frame.
Echo-reply	I←R	Here is the frame back.
Discard-request	I→R	Just discard this frame (for testing)

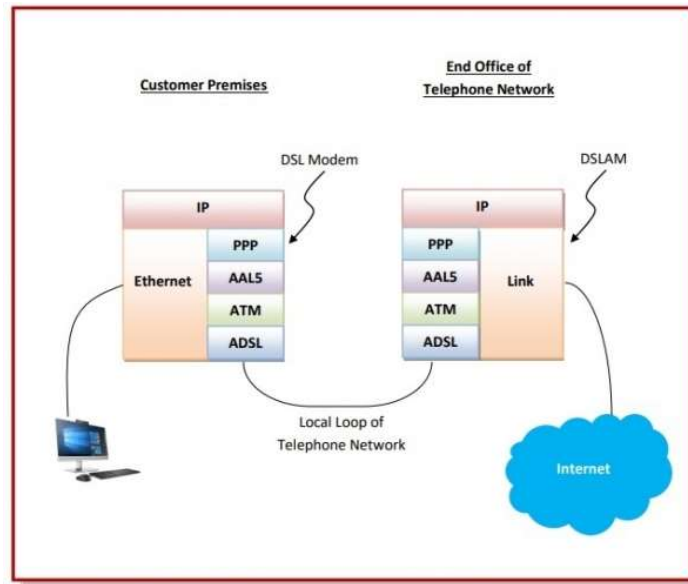
### Asymmetric Digital Subscriber Line (ADSL)

Asymmetric Digital Subscriber Line (ADSL) is a type of broadband communications technology that transmits digital data at a high bandwidth over existing phone lines to homes and businesses. ADSL protocol stack depicts the set of protocols and devices that are used along with ADSL. In order to access ADSL, a Digital Subscriber Line modem (DSL modem) is installed at the customer site. The DSL modem sends data bits over the local loop of the telephone network. The local loop is a two – wire connection between the subscriber's house and the end office of the telephone company. The data bits are accepted at the end office by a device called Digital Subscriber Line Access Multiplexer (DSLAM).

## Working Principle

In the customer site, IP packets are sent to the DSL modem using a link layer like Ethernet. The DSL modem transmits the data through the local loop of the telephone network to the DSLAM at the end office. The DSLAM extracts the IP packets and sends them over the Internet. The protocols start at the physical layer with ADSL. Above it are the protocols ATM (Asynchronous Transfer Mode) and AAL5 (ATM Adaptation Layer 5). At the top of the stack, just below the IP is PPP (Point – to – Point) Protocol.

The following diagram shows the ADSL protocol stack –



## UNIT IV

### References :

[nptel.ac.in](http://nptel.ac.in)

[www.idc-online.com](http://www.idc-online.com)

## NETWORK LAYER

### INTRODUCTION

- The Network Layer is the third layer of the OSI model.
- It handles the service requests from the transport layer and further forwards the service request to the data link layer.
- The network layer translates the logical addresses into physical addresses
- It determines the route from the source to the destination and also manages the traffic problems such as switching, routing and controls the congestion of data packets.
- The main role of the network layer is to move the packets from sending host to the receiving host.

The main functions performed by the network layer are:

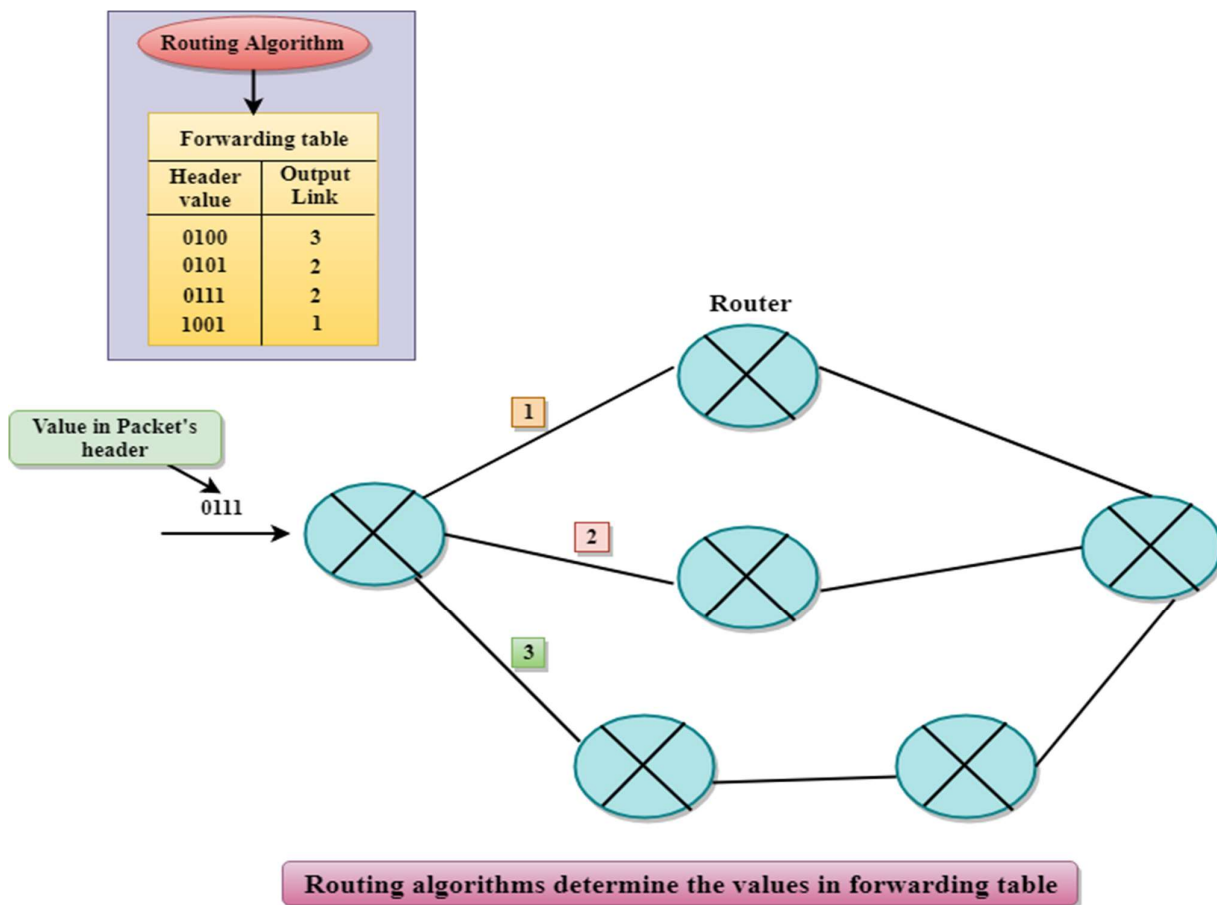
- **Routing:** When a packet reaches the router's input link, the router will move the packets to the router's output link. For example, a packet from S1 to R1 must be forwarded to the next router on the path to S2.
- **Logical Addressing:** The data link layer implements the physical addressing and network layer implements the logical addressing. Logical addressing is also used to distinguish between source and destination system. The network layer adds a header to the packet which includes the logical addresses of both the sender and the receiver.
- **Internetworking:** This is the main role of the network layer that it provides the logical connection between different types of networks.
- **Fragmentation:** The fragmentation is a process of breaking the packets into the smallest individual data units that travel through different networks.

---

### Forwarding & Routing

In Network layer, a router is used to forward the packets. Every router has a forwarding table. A router forwards a packet by examining a packet's header field and then using the header field value to index into the forwarding table. The value stored in the forwarding table corresponding to the header field value indicates the router's outgoing interface link to which the packet is to be forwarded.

For example, the router with a header field value of 0111 arrives at a router, and then router indexes this header value into the forwarding table that determines the output link interface is 2. The router forwards the packet to the interface 2. The routing algorithm determines the values that are inserted in the forwarding table. The routing algorithm can be centralized or decentralized.



#### Services Provided by the Network Layer

- **Guaranteed delivery:** This layer provides the service which guarantees that the packet will arrive at its destination.
- **Guaranteed delivery with bounded delay:** This service guarantees that the packet will be delivered within a specified host-to-host delay bound.
- **In-Order packets:** This service ensures that the packet arrives at the destination in the order in which they are sent.
- **Guaranteed max jitter:** This service ensures that the amount of time taken between two successive transmissions at the sender is equal to the time between their receipt at the destination.
- **Security services:** The network layer provides security by using a session key between the source and destination host. The network layer in the source host encrypts the payloads of datagrams being sent to the destination host. The network layer in the destination host would then decrypt the payload. In such a way, the network layer maintains the data integrity and source authentication services.

#### NETWORK LAYER DESIGN ISSUES

The network layer or layer 3 of the OSI (Open Systems Interconnection) model is concerned delivery of data packets from the source to the destination across multiple hops or links. It is the lowest layer that is concerned with end – to – end transmission. The designers who are concerned with designing this layer needs to cater to certain issues. These issues encompasses the services provided to the upper layers as well as internal design of the layer.

The design issues can be elaborated under four heads –

- Store – and – Forward Packet Switching
- Services to Transport Layer
- Providing Connection Oriented Service
- Providing Connectionless Service

### **Store – and – Forward Packet Switching**

The network layer operates in an environment that uses store and forward packet switching. The node which has a packet to send, delivers it to the nearest router. The packet is stored in the router until it has fully arrived and its checksum is verified for error detection. Once, this is done, the packet is forwarded to the next router. Since, each router needs to store the entire packet before it can forward it to the next hop, the mechanism is called store – and – forward switching.

### **Services to Transport Layer**

The network layer provides service its immediate upper layer, namely transport layer, through the network – transport layer interface. The two types of services provided are –

- Connection – Oriented Service – In this service, a path is setup between the source and the destination, and all the data packets belonging to a message are routed along this path.
- Connectionless Service – In this service, each packet of the message is considered as an independent entity and is individually routed from the source to the destination.

The objectives of the network layer while providing these services are –

- The services should not be dependent upon the router technology.
- The router configuration details should not be of a concern to the transport layer.
- A uniform addressing plan should be made available to the transport layer, whether the network is a LAN, MAN or WAN.

### **Providing Connection Oriented Service**

In connection – oriented services, a path or route called a virtual circuit is setup between the source and the destination nodes before the transmission starts. All the packets in the message are sent along this route. Each packet contains an identifier that denotes the virtual circuit to which it belongs to. When all the packets are transmitted, the virtual circuit is terminated and the connection is released. An example of connection – oriented service is MultiProtocol Label Switching (MPLS).

### **Providing Connectionless Service**

In connectionless service, since each packet is transmitted independently, each packet contains its routing information and is termed as datagram. The network using datagrams for transmission is called datagram networks or datagram subnets. No prior setup of routes are needed before transmitting a message. Each datagram belong to the message follows its own individual route from the source to the destination. An example of connectionless service is Internet Protocol or IP.

### **Quality of Service (QoS)**

Each service can be distinguished by its quality of service. These services can be of two types as explained below –

## Reliable Services

Reliable services are those which never lose data. It is usually a reliable service implemented by having the receiver acknowledgements of receipt of each message. Therefore, the sender is sure that it arrives. For example, remote login requires reliable service. But the declarations introduce overheads and delays, which are sometimes undesirable.

## Unreliable Services

Unreliable services lose minimal data or bits or pixels of the picture, but there is no significant effect on the result. For example, mobile customers should hear a bit of noise on the line or a misinterpret term from time to time than to learn a delay to wait for acceptance.

## ROUTING ALGORITHMS

A routing algorithm is a procedure that lays down the route or path to transfer data packets from source to the destination. They help in directing Internet traffic efficiently. After a data packet leaves its source, it can choose among the many different paths to reach its destination. Routing algorithm mathematically computes the best path, i.e. “least – cost path” that the packet can be routed through.

## SHORTEST PATH ALGORITHM

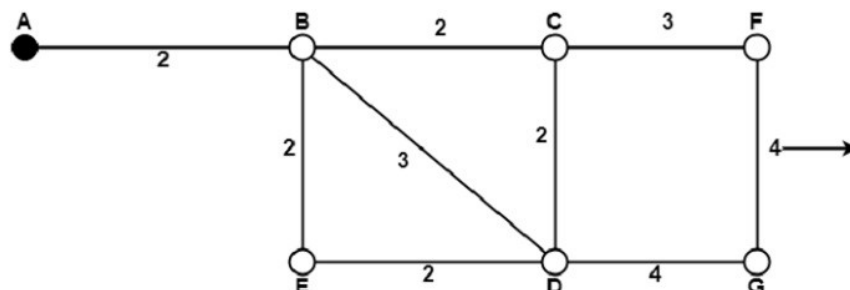
In this algorithm, to select a route, the algorithm discovers the shortest path between two nodes. It can use multiple hops, the geographical area in kilometres or labelling of arcs for measuring path length.

The labelling of arcs can be done with mean queuing, transmission delay for a standard test packet on an hourly basis, or computed as a function of bandwidth, average distance traffic, communication cost, mean queue length, measured delay or some other factors.

In shortest path routing, the topology communication network is defined using a directed weighted graph. The nodes in the graph define switching components and the directed arcs in the graph define communication connection between switching components. Each arc has a weight that defines the cost of sharing a packet between two nodes in a specific direction.

This cost is usually a positive value that can denote such factors as delay, throughput, error rate, financial costs, etc. A path between two nodes can go through various intermediary nodes and arcs. The goal of shortest path routing is to find a path between two nodes that has the lowest total cost, where the total cost of a path is the sum of arc costs in that path.

For example, Dijkstra uses the nodes labelling with its distance from the source node along the better-known route. Initially, all nodes are labelled with infinity, and as the algorithm proceeds, the label may change. The labelling graph is displayed in the figure.



Graphical Representation of Nodes with labeled path

It can be done in various passes as follows, with A as the source.

- Pass 1. B (2, A), C( $\infty$ , -), F( $\infty$ , -), e( $\infty$ , -), d( $\infty$ , -), G 60
- Pass 2. B (2, A), C(4, B), D(5, B), E(4, B), F( $\infty$ , -), G( $\infty$ , -)
- Pass 3. B(2, A), C(4, B), D(5, B), E(4, B), F(7, C), G(9, D)

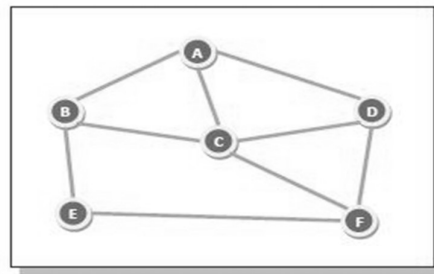
We can see that there can be two paths between A and G. One follows through ABCFG and the other through ABDG. The first one has a path length of 11, while the second one has 9. Hence, the second one, as G (9, D), is selected. Similarly, Node D has also three paths from A as ABD, ABCD and ABED. The first one has a path length of 5 rest two have 6. So, the first one is selected.

All nodes are searched in various passes, and finally, the routes with the shortest path lengths are made permanent, and the nodes of the path are used as a working node for the next round.

### **FLOODING**

Flooding is a non-adaptive routing technique following this simple method: when a data packet arrives at a router, it is sent to all the outgoing links except the one it has arrived on.

For example, let us consider the network in the figure, having six routers that are connected through transmission lines.



Using flooding technique –

- An incoming packet to A, will be sent to B, C and D.
- B will send the packet to C and E.
- C will send the packet to B, D and F.
- D will send the packet to C and F.
- E will send the packet to F.
- F will send the packet to C and E.

Types of Flooding

Flooding may be of three types –

- Uncontrolled flooding – Here, each router unconditionally transmits the incoming data packets to all its neighbours.
- Controlled flooding – They use some methods to control the transmission of packets to the neighbouring nodes. The two popular algorithms for controlled flooding are Sequence Number Controlled Flooding (SNCF) and Reverse Path Forwarding (RPF).
- Selective flooding – Here, the routers don't transmit the incoming packets only along those paths which are heading towards approximately in the right direction, instead of every available paths.

### Advantages of Flooding

- It is very simple to setup and implement, since a router may know only its neighbours.
- It is extremely robust. Even in case of malfunctioning of a large number routers, the packets find a way to reach the destination.
- All nodes which are directly or indirectly connected are visited. So, there are no chances for any node to be left out. This is a main criteria in case of broadcast messages.
- The shortest path is always chosen by flooding.

### Limitations of Flooding

- Flooding tends to create an infinite number of duplicate data packets, unless some measures are adopted to damp packet generation.
- It is wasteful if a single destination needs the packet, since it delivers the data packet to all nodes irrespective of the destination.
- The network may be clogged with unwanted and duplicate data packets. This may hamper delivery of other data packets.

### **DISTANCE VECTOR ROUTING ALGORITHM**

In distance-vector routing (DVR), each router is required to inform the topology changes to its neighboring routers periodically. Historically it is known as the old ARPNET routing algorithm or Bellman-Ford algorithm.

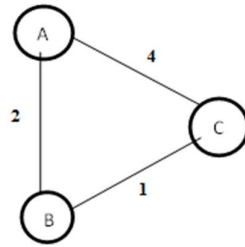
- In DVR, each router maintains a routing table. It contains only one entry for each router. It contains two parts – a preferred outgoing line to use for that destination and an estimate of time (delay). Tables are updated by exchanging the information with the neighbor's nodes.
- Each router knows the delay in reaching its neighbors (Ex – send echo request).
- Routers periodically exchange routing tables with each of their neighbors.
- It compares the delay in its local table with the delay in the neighbor's table and the cost of reaching that neighbor.
- If the path via the neighbor has a lower cost, then the router updates its local table to forward packets to the neighbor.

### Example – Distance Vector Router Protocol

In the network shown below, there are three routers, A, B, and C, with the following weights –  $AB = 2$ ,  $BC = 3$  and  $CA = 5$ .

Step 1 – In this DVR network, each router shares its routing table with every neighbor. For example, A will share its routing table with neighbors B and C and neighbors B and C will share their routing table with A.





Form A	A	B	C
A	0	2	3
B			
C			

Form B	A	B	C
A			
B	2	0	1
C			
Form C	A	B	C
A			
B			
C	3	1	0

Step 2 – If the path via a neighbor has a lower cost, then the router updates its local table to forward packets to the neighbor. In this table, the router updates the lower cost for A and C by updating the new weight from 4 to 3 in router A and from 4 to 3 in router C.

Form A	A	B	C
A	0	2	3
B			

C			
Form B	A	B	C
A			
B	2	0	1
C			
Form C	A	B	C
A			
B			
C	3	1	0

Step 3 – The final updated routing table with lower cost distance vector routing protocol for all routers A, B, and C is given below –

Router A

Form A	A	B	C
A	0	2	3
B	2	0	1
C	3	1	0

Router B

Form B	A	B	C
A	0	2	3
B	2	0	1
C	3	1	0

Router C

Form C	A	B	C
A	0	2	3
B	2	0	1
C	3	1	0

## **LINK STATE ROUTING**

Link state routing is a method in which each router shares its neighbourhood's knowledge with every other router in the internetwork. In this algorithm, each router in the network understands the network topology then makes a routing table depend on this topology.

Each router will share data about its connection to its neighbour, who will, consecutively, reproduce the data to its neighbours, etc. This appears just before all routers have constructed a topology of the network.

In LSP, each node transmits its IP address and the MAC to its neighbor with its signature. Neighbors determine the signature and maintain a record of the combining IP address and the MAC. The Neighbor Lookup Protocol (NLP) of LSP derives and maintains the MAC and IP address of every network frame accepted by a node. The extracted data can support the mapping of MACs and IP addresses.

The link-state flooding algorithm prevents the general issues of broadcast in the existence of loops by having every node maintain a database of all LSP messages. The creator of each LSP contains its identity, data about the connection that has changed status, and also a sequence number.

## **BROADCAST ROUTING**

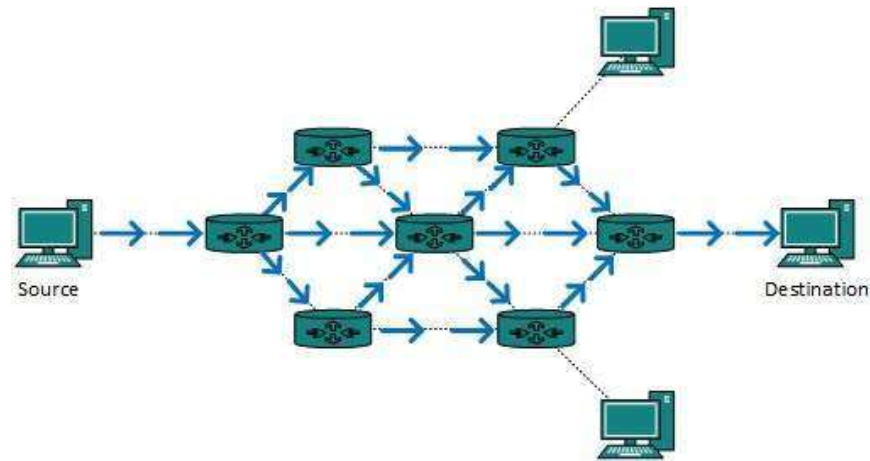
By default, the broadcast packets are not routed and forwarded by the routers on any network. Routers create broadcast domains. But it can be configured to forward broadcasts in some special cases. A broadcast message is destined to all network devices.

Broadcast routing can be done in two ways (algorithm):

- A router creates a data packet and then sends it to each host one by one. In this case, the router creates multiple copies of single data packet with different destination addresses. All packets are sent as unicast but because they are sent to all, it simulates as if router is broadcasting.

This method consumes lots of bandwidth and router must destination address of each node.

- Secondly, when router receives a packet that is to be broadcasted, it simply floods those packets out of all interfaces. All routers are configured in the same way.

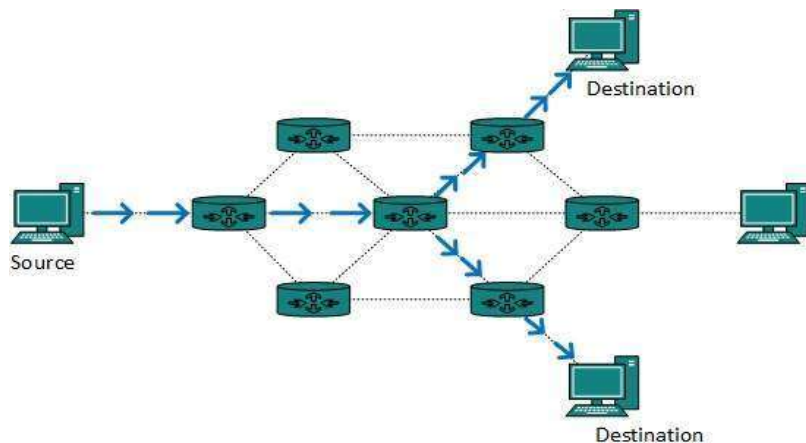


This method is easy on router's CPU but may cause the problem of duplicate packets received from peer routers.

Reverse path forwarding is a technique, in which router knows in advance about its predecessor from where it should receive broadcast. This technique is used to detect and discard duplicates.

### **MULTICAST ROUTING**

Multicast routing is special case of broadcast routing with significance difference and challenges. In broadcast routing, packets are sent to all nodes even if they do not want it. But in Multicast routing, the data is sent to only nodes which wants to receive the packets.

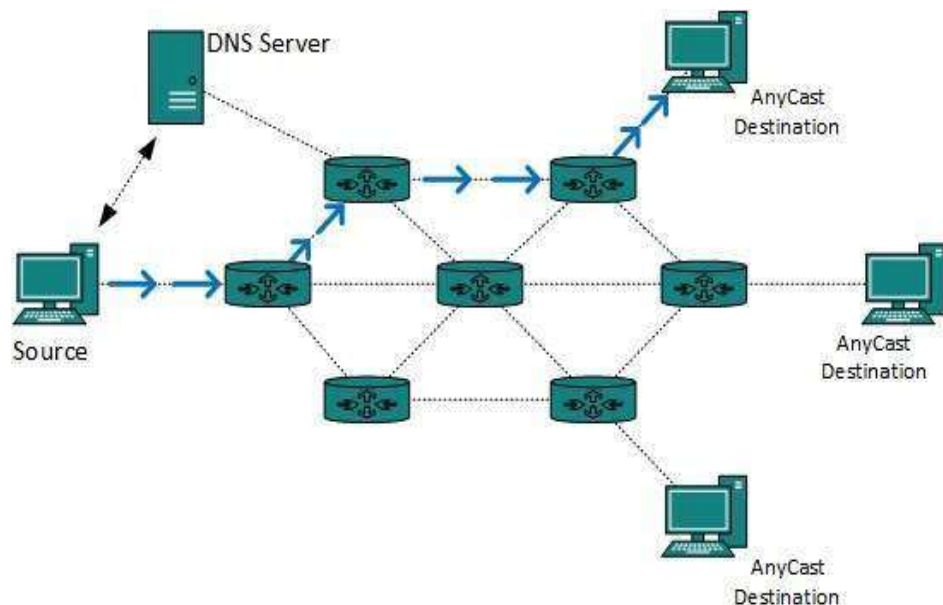


The router must know that there are nodes, which wish to receive multicast packets (or stream) then only it should forward. Multicast routing works spanning tree protocol to avoid looping.

Multicast routing also uses reverse path Forwarding technique, to detect and discard duplicates and loops.

### **ANYCAST ROUTING**

Anycast packet forwarding is a mechanism where multiple hosts can have same logical address. When a packet destined to this logical address is received, it is sent to the host which is nearest in routing topology.



Anycast routing is done with help of DNS server. Whenever an Anycast packet is received it is enquired with DNS to where to send it. DNS provides the IP address which is the nearest IP configured on it.

## **ROUTING FOR MOBILE HOSTS**

### **Types of host:**

- We can distinguish two other kinds of hosts.
- Migratory hosts are basically stationary hosts who move from one fixed site to another from time to time but use the network only when they are physically connected to it.
- Roaming hosts actually compute on the run and want to maintain their connections as they move around.

### **Mobile Host:**

- By the term mobile host, all hosts that are away from home and still want to be connected.
- All hosts are assumed to have a permanent home location that never changes.
- The routing goal in systems with mobile hosts is to make it possible to send packets to mobile hosts using their home addresses and have the packets efficiently reach them wherever they may be.
- The trick, of course, is to find them.
- In the model of Fig, the world is divided up (geographically) into small units called areas, where an area is typically a LAN or wireless cell.

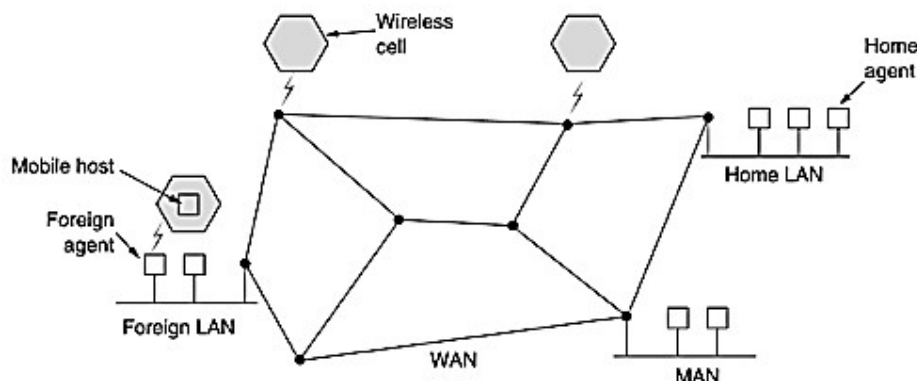


Figure: A WAN to which LANs, MANs, and wireless cells are attached

- Each area has one or more foreign agents, which are processes that keep track of all mobile hosts visiting the area.
- In addition, each area has a home agent, which keeps track of hosts whose home is in the area, but who are currently visiting another area.
- When a new host enters an area, either by connecting to it (e.g., plugging into the LAN) or just wandering into the cell, his computer must register itself with the foreign agent there.

### **CONGESTION CONTROL ALGORITHMS:**

What is **congestion**?

A state occurring in network layer when the message traffic is so heavy that it slows down network response time.

**Effects of Congestion**

- As delay increases, performance decreases.
- If delay increases, retransmission occurs, making situation worse.

**Congestion control algorithms**

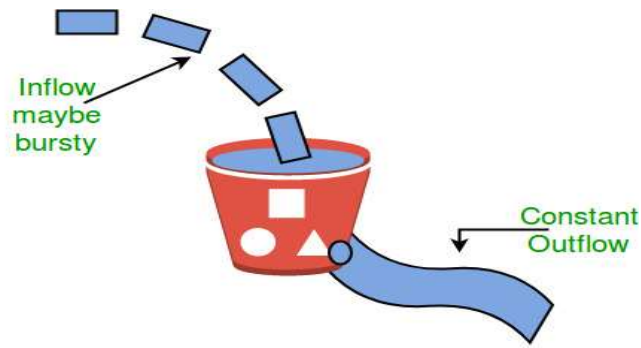
- Congestion Control is a mechanism that controls the entry of data packets into the network, enabling a better use of a shared network infrastructure and avoiding congestive collapse.
- Congestive-Avoidance Algorithms (CAA) are implemented at the TCP layer as the mechanism to avoid congestive collapse in a network.
- There are two congestion control algorithm which are as follows:

**Leaky Bucket Algorithm**

- The leaky bucket algorithm discovers its use in the context of network traffic shaping or rate-limiting.
- A leaky bucket execution and a token bucket execution are predominantly used for traffic shaping algorithms.
- This algorithm is used to control the rate at which traffic is sent to the network and shape the burst traffic to a steady traffic stream.
- The disadvantages compared with the leaky-bucket algorithm are the inefficient use of available network resources.
- The large area of network resources such as bandwidth is not being used effectively.

Example:

Imagine a bucket with a small hole in the bottom.No matter at what rate water enters the bucket, the outflow is at constant rate.When the bucket is full with water additional water entering spills over the sides and is lost.



Similarly, each network interface contains a leaky bucket and the following **steps** are involved in leaky bucket algorithm:

1. When host wants to send packet, packet is thrown into the bucket.
2. The bucket leaks at a constant rate, meaning the network interface transmits packets at a constant rate.
3. Bursty traffic is converted to a uniform traffic by the leaky bucket.
4. In practice the bucket is a finite queue that outputs at a finite rate.

### **Token bucket Algorithm**

- The leaky bucket algorithm has a rigid output design at an average rate independent of the bursty traffic.
- In some applications, when large bursts arrive, the output is allowed to speed up. This calls for a more flexible algorithm, preferably one that never loses information. Therefore, a token bucket algorithm finds its uses in network traffic shaping or rate-limiting.
- It is a control algorithm that indicates when traffic should be sent. This order comes based on the display of tokens in the bucket.
- The bucket contains tokens. Each of the tokens defines a packet of predetermined size. Tokens in the bucket are deleted for the ability to share a packet.
- When tokens are shown, a flow to transmit traffic appears in the display of tokens.
- No token means no flow sends its packets. Hence, a flow transfers traffic up to its peak burst rate in good tokens in the bucket.

**Need** of token bucket Algorithm:-

The leaky bucket algorithm enforces output pattern at the average rate, no matter how bursty the traffic is. So in order to deal with the bursty traffic we need a flexible algorithm so that the data is not lost. One such algorithm is token bucket algorithm.

**Steps** of this algorithm can be described as follows:

1. In regular intervals tokens are thrown into the bucket.  $f$
2. The bucket has a maximum capacity.  $f$
3. If there is a ready packet, a token is removed from the bucket, and the packet is sent.
4. If there is no token in the bucket, the packet cannot be sent.

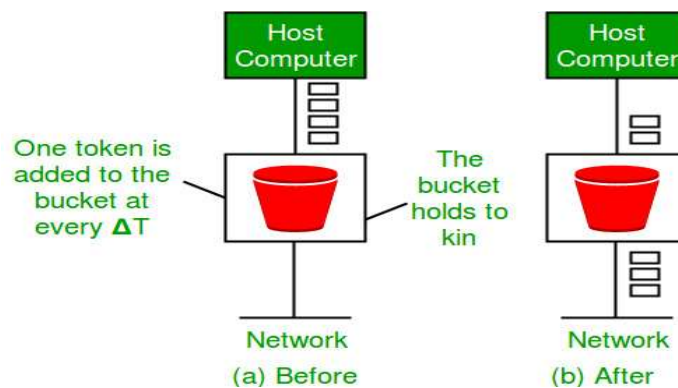
Example:

In figure we see a bucket holding three tokens, with five packets waiting to be transmitted. For a packet to be transmitted, it must capture and destroy one token. In figure (B) We see that three of the five packets have gotten through, but the other two are stuck waiting for more tokens to be generated.

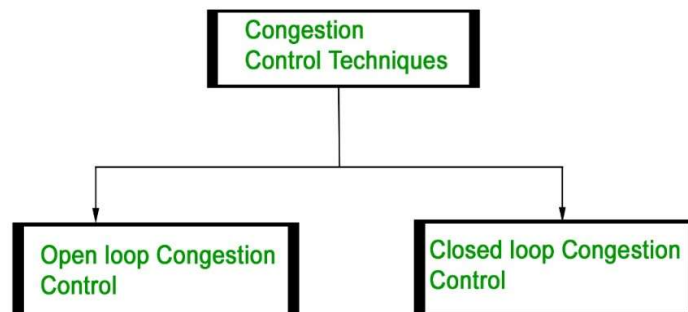
**Ways in which token bucket is superior to leaky bucket:** The leaky bucket algorithm controls the rate at which the packets are introduced in the network, but it is very conservative in nature. Some flexibility is introduced in the token bucket algorithm. In the token bucket, algorithm tokens are generated at each tick (up to a certain limit). For an incoming packet to be transmitted, it must capture a token and the transmission takes place at the same rate. Hence some of the busty packets are transmitted at the same rate if tokens are available and thus introduces some amount of flexibility in the system.

**Formula:**  $M * s = C + \rho * s$  where  $S$  – is time taken  $M$  – Maximum output rate  $\rho$  – Token arrival rate  $C$  – Capacity of the token bucket in byte

Let's understand with an example,



Congestion control refers to the techniques used to control or prevent congestion. Congestion control techniques can be broadly classified into two categories:



### Open Loop Congestion Control

Open loop congestion control policies are applied to prevent congestion before it happens. The congestion control is handled either by the source or the destination.



## **Policies adopted by open loop congestion control –**

### **1. Retransmission Policy :**

It is the policy in which retransmission of the packets are taken care of. If the sender feels that a sent packet is lost or corrupted, the packet needs to be retransmitted. This transmission may increase the congestion in the network.

To prevent congestion, retransmission timers must be designed to prevent congestion and also able to optimize efficiency.

### **2. Window Policy :**

The type of window at the sender's side may also affect the congestion. Several packets in the Go-back-n window are re-sent, although some packets may be received successfully at the receiver side. This duplication may increase the congestion in the network and make it worse. Therefore, Selective repeat window should be adopted as it sends the specific packet that may have been lost.

### **3. Discarding Policy :**

A good discarding policy adopted by the routers is that the routers may prevent congestion and at the same time partially discard the corrupted or less sensitive packages and also be able to maintain the quality of a message. In case of audio file transmission, routers can discard less sensitive packets to prevent congestion and also maintain the quality of the audio file.

### **4. Acknowledgment Policy :**

Since acknowledgements are also the part of the load in the network, the acknowledgment policy imposed by the receiver may also affect congestion. Several approaches can be used to prevent congestion related to acknowledgment. The receiver should send acknowledgement for N packets rather than sending acknowledgement for a single packet. The receiver should send an acknowledgment only if it has to send a packet or a timer expires.

### **5. Admission Policy :**

In admission policy a mechanism should be used to prevent congestion. Switches in a flow should first check the resource requirement of a network flow before transmitting it further. If there is a chance of a congestion or there is a congestion in the network, router should deny establishing a virtual network connection to prevent further congestion.

All the above policies are adopted to prevent congestion before it happens in the network.

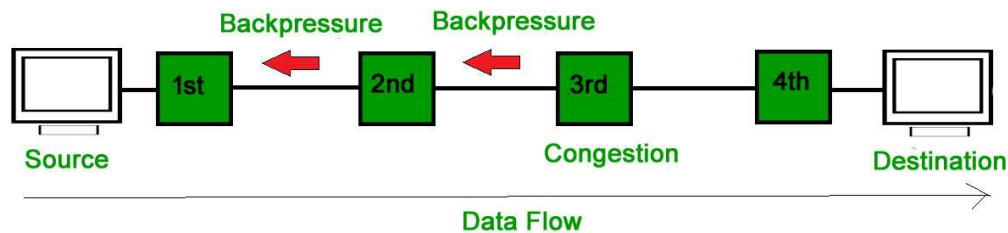
## **Closed Loop Congestion Control**

Closed loop congestion control techniques are used to treat or alleviate congestion after it happens. Several techniques are used by different protocols; some of them are:

### **1. Backpressure :**

Backpressure is a technique in which a congested node stops receiving packets from upstream node. This may cause the upstream node or nodes to become congested and reject receiving data from above nodes.

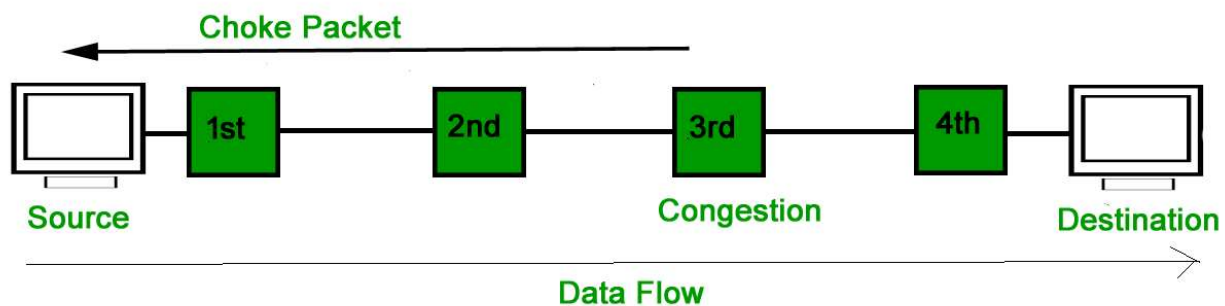
Backpressure is a node-to-node congestion control technique that propagate in the opposite direction of data flow. The backpressure technique can be applied only to virtual circuit where each node has information of its above upstream node.



In above diagram the 3rd node is congested and stops receiving packets as a result 2nd node may be get congested due to slowing down of the output data flow. Similarly 1st node may get congested and inform the source to slow down.

## 2. Choke Packet Technique :

Choke packet technique is applicable to both virtual networks as well as datagram subnets. A choke packet is a packet sent by a node to the source to inform it of congestion. Each router monitors its resources and the utilization at each of its output lines. Whenever the resource utilization exceeds the threshold value which is set by the administrator, the router directly sends a choke packet to the source giving it a feedback to reduce the traffic. The intermediate nodes through which the packets has traveled are not warned about congestion.



## 3. Implicit Signaling :

In implicit signaling, there is no communication between the congested nodes and the source. The source guesses that there is congestion in a network. For example when sender sends several packets and there is no acknowledgment for a while, one assumption is that there is a congestion.

## 4. Explicit Signaling :

In explicit signaling, if a node experiences congestion it can explicitly sends a packet to the source or destination to inform about congestion. The difference between choke packet and explicit signaling is that the signal is included in the packets that carry data rather than creating a different packet as in case of choke

packet technique.

Explicit signaling can occur in either forward or backward direction.

- **Forward Signaling** : In forward signaling, a signal is sent in the direction of the congestion. The destination is warned about congestion. The receiver in this case adopt policies to prevent further congestion.
- **Backward Signaling** : In backward signaling, a signal is sent in the opposite direction of the congestion. The source is warned about congestion and it needs to slow down.

### **QUALITY OF SERVICES:**

**Quality-of-Service (QoS)** refers to traffic control mechanisms that seek to either differentiate performance based on application or network-operator requirements or provide predictable or guaranteed performance to applications, sessions, or traffic aggregates. Basic phenomenon for QoS means in terms of packet delay and losses of various kinds.

#### **Need for QoS –**

- Video and audio conferencing require bounded delay and loss rate.
- Video and audio streaming requires bounded packet loss rate, it may not be so sensitive to delay.
- Time-critical applications (real-time control) in which bounded delay is considered to be an important factor.
- Valuable applications should be provided better services than less valuable applications.

#### **QoS Specification –**

QoS requirements can be specified as:

1. Delay
2. Delay Variation(Jitter)
3. Throughput
4. Error Rate

There are two types of QoS Solutions:

##### **1. Stateless Solutions –**

Routers maintain no fine-grained state about traffic, one positive factor of it is that it is scalable and robust. But it has weak services as there is no guarantee about the kind of delay or performance in a particular application which we have to encounter.

##### **2. Stateful Solutions –**

Routers maintain a per-flow state as flow is very important in providing the Quality-of-Service i.e. providing powerful services such as guaranteed services and high resource utilization, providing protection, and is much less scalable and robust.

#### **Integrated Services (IntServ) –**

1. An architecture for providing QoS guarantees in IP networks for individual application sessions.
2. Relies on resource reservation, and routers need to maintain state information of allocated resources and respond to new call setup requests.

3. Network decides whether to admit or deny a new call setup request.

### **IntServ QoS Components –**

- Resource reservation: call setup signaling, traffic, QoS declaration, per-element admission control.
- QoS-sensitive scheduling e.g WFQ queue discipline.
- QoS-sensitive routing algorithm(QSPF)
- QoS-sensitive packet discard strategy.

### **RSVP-Internet Signaling –**

It creates and maintains distributed reservation state, initiated by the receiver and scales for multicast, which needs to be refreshed otherwise reservation times out as it is in soft state. Latest paths were discovered through “PATH” messages (forward direction) and used by RESV messages (reserve direction).

### **Call Admission –**

- Session must first declare its QoS requirement and characterize the traffic it will send through the network.
- **R-specification:** defines the QoS being requested, i.e. what kind of bound we want on the delay, what kind of packet loss is acceptable, etc.
- **T-specification:** defines the traffic characteristics like bustiness in the traffic.
- A signaling protocol is needed to carry the R-spec and T-spec to the routers where reservation is required.
- Routers will admit calls based on their R-spec, T-spec and based on the current resource allocated at the routers to other calls.

### **Diff-Serv**

Differentiated Service is a stateful solution in which each flow doesn't mean a different state. It provides reduced state services i.e. maintaining state only for larger granular flows rather than end-to-end flows tries to achieve the best of both worlds.

Intended to address the following difficulties with IntServ and RSVP:

#### **1. Flexible Service Models:**

IntServ has only two classes, want to provide more qualitative service classes: want to provide 'relative' service distinction.

#### **2. Simpler signaling:**

Many applications and users may only want to specify a more qualitative notion of service.

### **Streaming Live Multimedia –**

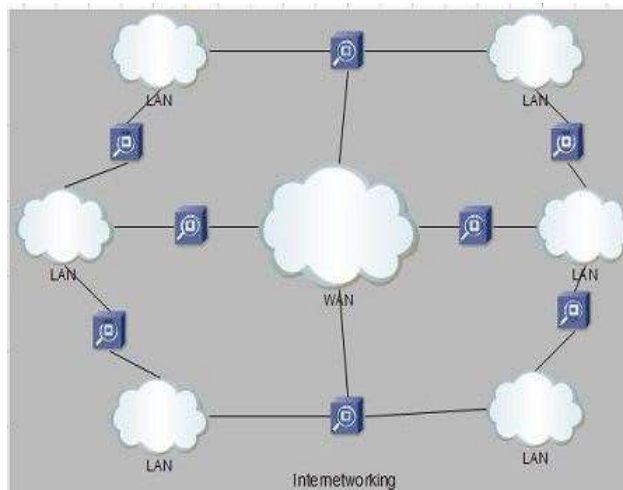
- **Examples:** Internet radio talk show, Live sporting event.
- **Streaming:** playback buffer, playback buffer can lag tens of seconds after and still have timing constraint.
- **Interactivity:** fast forward is impossible, but rewind and pause is possible

### **INTERNET WORKING:**

**Internetworking** started as a way to connect disparate types of [computer](#) networking technology. [Computer](#) network term is used to describe two or more computers that are linked to each other. When two or more computer LANs or WANs or computer network segments are connected using devices such as a *router* and configure by logical addressing scheme with a [protocol](#) such as IP, then it is called as **computer internetworking**.

**Internetworking** is a term used by Cisco. Any interconnection among or between public, private, commercial, industrial, or governmental computer networks may also be defined as an internetwork or “**Internetworking**”.

In modern practice, the interconnected computer networks or **Internetworking** use the [Internet Protocol](#). Two architectural models are commonly used to describe the protocols and methods used in **internetworking**. The standard reference model for **internetworking** is Open Systems Interconnection (OSI).



### Type of Internetworking

**Internetworking** is implemented in Layer 3 (Network Layer) of this model. The most notable example of internetworking is the [Internet](#) (capitalized). There are three variants of internetwork or **Internetworking**, depending on who administers and who participates in them :

- Extranet
- Intranet
- Internet

Intranets and extranets may or may not have connections to the Internet. If connected to the Internet, the intranet or extranet is normally protected from being accessed from the Internet without proper authorization. The Internet is not considered to be a part of the intranet or extranet, although it may serve as a portal for access to portions of an extranet.

### Extranet

An extranet is a **network of internetwork or Internetworking** that is limited in scope to a **single organisation or entity** but which also has **limited connections** to the networks of one or more other usually, but not necessarily, trusted organizations or entities. Technically, an **extranet may also be categorized as a**

MAN, WAN, or other type of network, although, by definition, an extranet cannot consist of a single LAN; it must have at least one connection with an external network.

### **Intranet**

An intranet is a set of interconnected networks or Internetworking, using the Internet Protocol and uses IP-based tools such as web browsers and ftp tools, which is under the control of a single administrative entity. That administrative entity closes the intranet to the rest of the world, and allows only specific users. Most commonly, an intranet is the internal network of a company or other enterprise. A large intranet will typically have its own web server to provide users with [information](#).

### **Internet**

A specific **Internetworking**, consisting of a **worldwide interconnection** of governmental, academic, public, and private networks based upon the Advanced Research Projects Agency Network (**ARPANET**) developed by ARPA of the U.S. **Department of Defense** also **home** to the **World Wide Web (WWW)** and referred to as the '**Internet**' with a capital 'I' to distinguish it from other generic internetworks. Participants in the Internet, or their service providers, use IP Addresses obtained from address registries that control assignments.

#### **Advantages:**

**Increased connectivity:** Internetworking enables devices on different networks to communicate with each other, which increases connectivity and enables new applications and services.

**Resource sharing:** Internetworking allows devices to share resources across networks, such as printers, servers, and storage devices. This can reduce costs and improve efficiency by allowing multiple devices to share resources.

**Improved scalability:** Internetworking allows networks to be expanded and scaled as needed to accommodate growing numbers of devices and users.

**Improved collaboration:** Internetworking enables teams and individuals to collaborate and work together more effectively, regardless of their physical location.

**Access to remote resources:** Internetworking allows users to access resources and services that are physically located on remote networks, improving accessibility and flexibility.

#### **Disadvantages:**

**Security risks:** Internetworking can create security vulnerabilities and increase the risk of cyberattacks and data breaches. Connecting multiple networks together increases the number of entry points for attackers, making it more difficult to secure the entire system.

**Complexity:** Internetworking can be complex and requires specialized knowledge and expertise to set up and maintain. This can increase costs and create additional maintenance overhead.

**Performance issues:** Internetworking can lead to performance issues, particularly if networks are not properly optimized and configured. This can result in slow response times and poor network performance.

**Compatibility issues:** Internetworking can lead to compatibility issues, particularly if different networks are using different protocols or technologies. This can make it difficult to integrate different systems and may require additional resources to resolve.

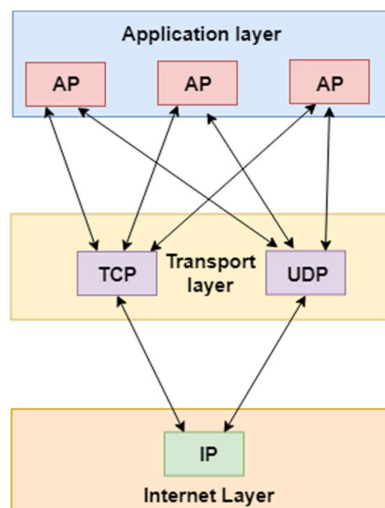
**Management overhead:** Internetworking can create additional management overhead, particularly if multiple networks are involved. This can increase costs and require additional resources to manage effectively.

### References:

[Internetworking – Wiki](#)

### TRANSPORT LAYER:

- The transport layer is a 4<sup>th</sup> layer from the top.
- The main role of the transport layer is to provide the communication services directly to the application processes running on different hosts.
- The transport layer provides a logical communication between application processes running on different hosts. Although the application processes on different hosts are not physically connected, application processes use the logical communication provided by the transport layer to send the messages to each other.
- The transport layer protocols are implemented in the end systems but not in the network routers.
- A computer network provides more than one protocol to the network applications. For example, TCP and UDP are two transport layer protocols that provide a different set of services to the network layer.
- All transport layer protocols provide multiplexing/demultiplexing service. It also provides other services such as reliable data transfer, bandwidth guarantees, and delay guarantees.
- Each of the applications in the application layer has the ability to send a message by using TCP or UDP. The application communicates by using either of these two protocols. Both TCP and UDP will then communicate with the internet protocol in the internet layer. The applications can read and write to the transport layer. Therefore, we can say that communication is a two-way process.

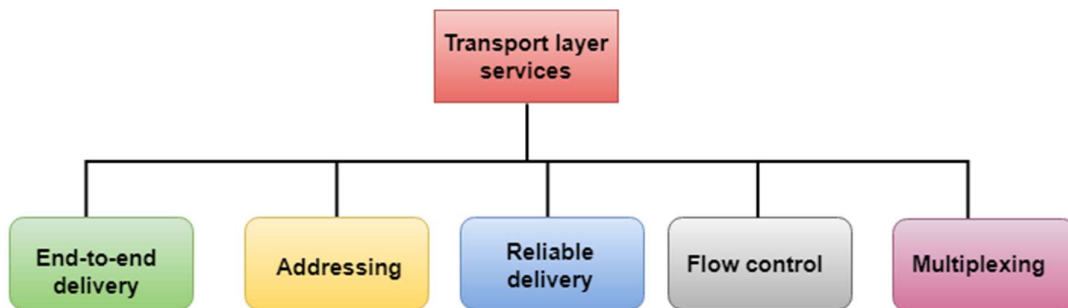


## TRANSPORT SERVICES:

The services provided by the transport layer are similar to those of the data link layer. The data link layer provides the services within a single network while the transport layer provides the services across an internetwork made up of many networks. The data link layer controls the physical layer while the transport layer controls all the lower layers.

**The services provided by the transport layer protocols can be divided into five categories:**

- End-to-end delivery
- Addressing
- Reliable delivery
- Flow control
- Multiplexing



### End-to-end delivery:

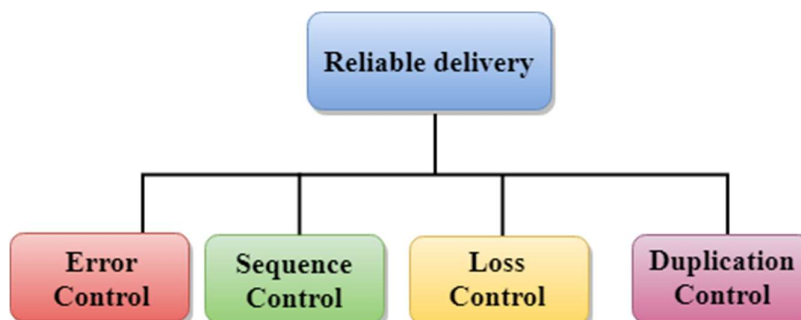
The transport layer transmits the entire message to the destination. Therefore, it ensures the end-to-end delivery of an entire message from a source to the destination.

### Reliable delivery:

The transport layer provides reliability services by retransmitting the lost and damaged packets.

**The reliable delivery has four aspects:**

- Error control
- Sequence control
- Loss control
- Duplication control

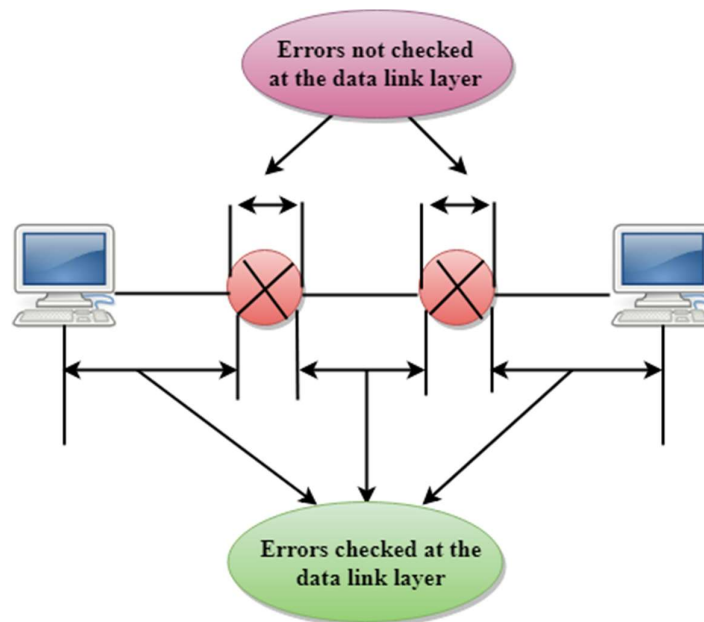


### **Error Control**

- The primary role of reliability is **Error Control**. In reality, no transmission will be 100 percent error-free delivery. Therefore, transport layer protocols are designed to provide error-free transmission.



- The data link layer also provides the error handling mechanism, but it ensures only node-to-node error-free delivery. However, node-to-node reliability does not ensure the end-to-end reliability.
- The data link layer checks for the error between each network. If an error is introduced inside one of the routers, then this error will not be caught by the data link layer. It only detects those errors that have been introduced between the beginning and end of the link. Therefore, the transport layer performs the checking for the errors end-to-end to ensure that the packet has arrived correctly.



### Sequence Control

- The second aspect of the reliability is sequence control which is implemented at the transport layer.
- On the sending end, the transport layer is responsible for ensuring that the packets received from the upper layers can be used by the lower layers. On the receiving end, it ensures that the various segments of a transmission can be correctly reassembled.

### Loss Control

Loss Control is a third aspect of reliability. The transport layer ensures that all the fragments of a transmission arrive at the destination, not some of them. On the sending end, all the fragments of transmission are given sequence numbers by a transport layer. These sequence numbers allow the receiver's transport layer to identify the missing segment.

### Duplication Control

Duplication Control is the fourth aspect of reliability. The transport layer guarantees that no duplicate data arrive at the destination. Sequence numbers are used to identify the lost packets; similarly, it allows the receiver to identify and discard duplicate segments.

### Flow Control

Flow control is used to prevent the sender from overwhelming the receiver. If the receiver is overloaded with too much data, then the receiver discards the packets and asking for the retransmission of packets. This increases network congestion and thus, reducing the system performance. The transport layer is responsible for flow control. It uses the sliding window protocol that makes the data transmission more efficient as well

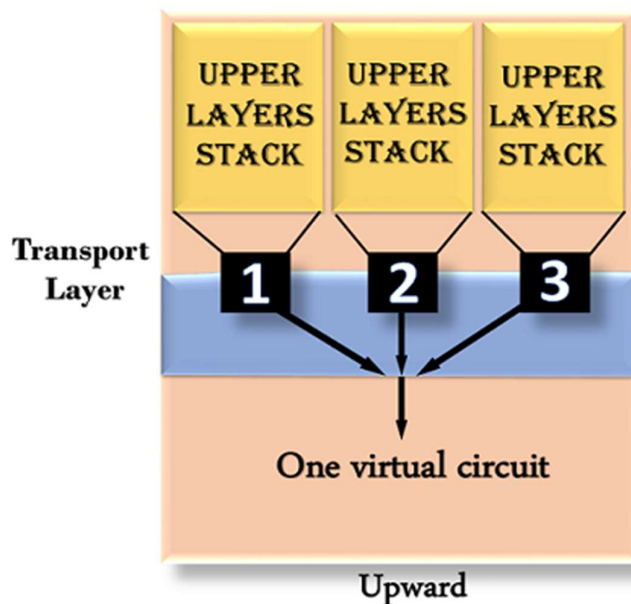
as it controls the flow of data so that the receiver does not become overwhelmed. Sliding window protocol is byte oriented rather than frame oriented.

### Multiplexing

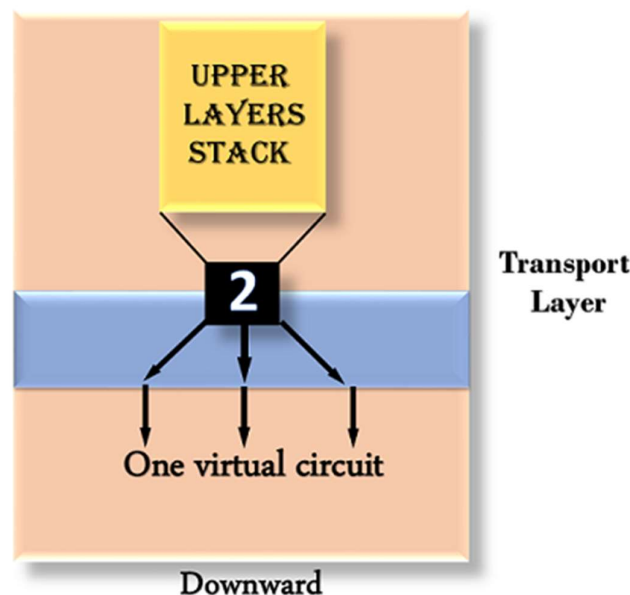
The transport layer uses the multiplexing to improve transmission efficiency.

**Multiplexing can occur in two ways:**

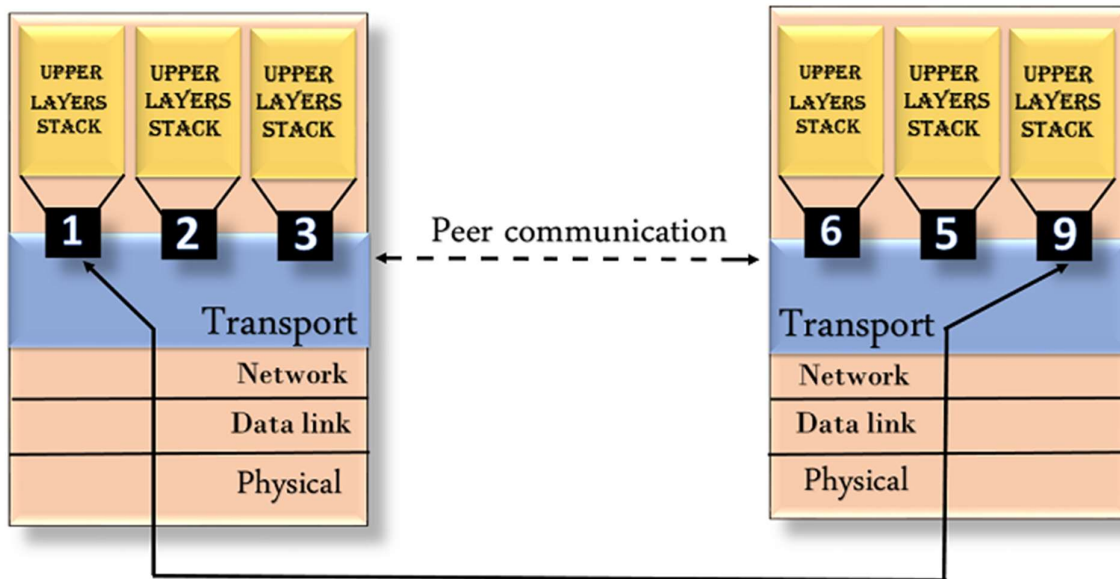
- **Upward multiplexing:** Upward multiplexing means multiple transport layer connections use the same network connection. To make more cost-effective, the transport layer sends several transmissions bound for the same destination along the same path; this is achieved through upward multiplexing.



- **Downward multiplexing:** Downward multiplexing means one transport layer connection uses the multiple network connections. Downward multiplexing allows the transport layer to split a connection among several paths to improve the throughput. This type of multiplexing is used when networks have a low or slow capacity.

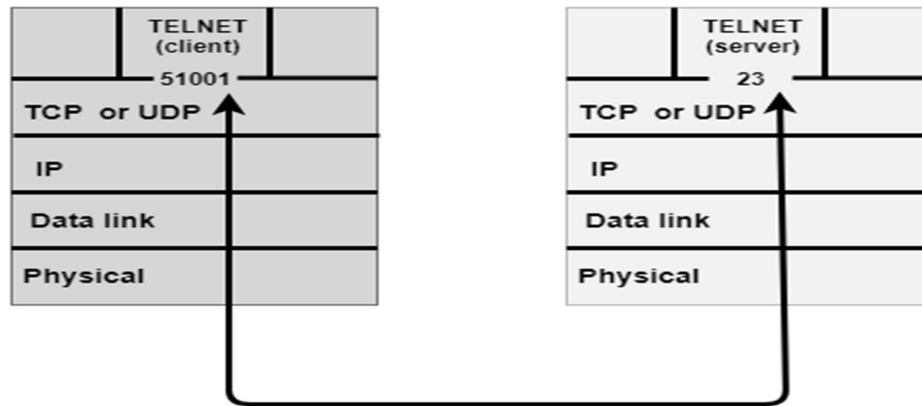


- According to the layered model, the transport layer interacts with the functions of the session layer. Many protocols combine session, presentation, and application layer protocols into a single layer known as the application layer. In these cases, delivery to the session layer means the delivery to the application layer. Data generated by an application on one machine must be transmitted to the correct application on another machine. In this case, addressing is provided by the transport layer.
- The transport layer provides the user address which is specified as a station or port. The port variable represents a particular TS user of a specified station known as a Transport Service access point (TSAP). Each station has only one transport entity.
- The transport layer protocols need to know which upper-layer protocols are communicating.



### **ELEMENTS OF TRANSPORT LAYER PROTOCOL**

- The transport layer is represented by two protocols: TCP and UDP.
- The IP protocol in the network layer delivers a datagram from a source host to the destination host.
- Nowadays, the operating system supports multiuser and multiprocessing environments, an executing program is called a process. When a host sends a message to other host means that source process is sending a process to a destination process. The transport layer protocols define some connections to individual ports known as protocol ports.
- An IP protocol is a host-to-host protocol used to deliver a packet from source host to the destination host while transport layer protocols are port-to-port protocols that work on the top of the IP protocols to deliver the packet from the originating port to the IP services, and from IP services to the destination port.
- Each port is defined by a positive integer address, and it is of 16 bits.



## UDP

- UDP stands for **User Datagram Protocol**.
- UDP is a simple protocol and it provides nonsequenced transport functionality.
- UDP is a connectionless protocol.
- This type of protocol is used when reliability and security are less important than speed and size.
- UDP is an end-to-end transport level protocol that adds transport-level addresses, checksum error control, and length information to the data from the upper layer.
- The packet produced by the UDP protocol is known as a user datagram.

### **User Datagram Format**

The user datagram has a 16-byte header which is shown below:

Source port address 16 bits	Destination port address 16 bits
Total Length 16 bits	Checksum 16 bits
Data	

Where,

- **Source port address:** It defines the address of the application process that has delivered a message. The source port address is of 16 bits address.
- **Destination port address:** It defines the address of the application process that will receive the message. The destination port address is of a 16-bit address.
- **Total length:** It defines the total length of the user datagram in bytes. It is a 16-bit field.
- **Checksum:** The checksum is a 16-bit field which is used in error detection.

### **Disadvantages of UDP protocol**

- UDP provides basic functions needed for the end-to-end delivery of a transmission.
- It does not provide any sequencing or reordering functions and does not specify the damaged packet when reporting an error.
- UDP can discover that an error has occurred, but it does not specify which packet has been lost as it does not contain an ID or sequencing number of a particular data segment.

## TCP

- TCP stands for Transmission Control Protocol.
- It provides full transport layer services to applications.
- It is a connection-oriented protocol means the connection established between both the ends of the transmission. For creating the connection, TCP generates a virtual circuit between sender and receiver for the duration of a transmission.

### Features Of TCP protocol

- **Stream data transfer:** TCP protocol transfers the data in the form of contiguous stream of bytes. TCP group the bytes in the form of TCP segments and then passed it to the IP layer for transmission to the destination. TCP itself segments the data and forward to the IP.
- **Reliability:** TCP assigns a sequence number to each byte transmitted and expects a positive acknowledgement from the receiving TCP. If ACK is not received within a timeout interval, then the data is retransmitted to the destination. The receiving TCP uses the sequence number to reassemble the segments if they arrive out of order or to eliminate the duplicate segments.
- **Flow Control:** When receiving TCP sends an acknowledgement back to the sender indicating the number the bytes it can receive without overflowing its internal buffer. The number of bytes is sent in ACK in the form of the highest sequence number that it can receive without any problem. This mechanism is also referred to as a window mechanism.
- **Multiplexing:** Multiplexing is a process of accepting the data from different applications and forwarding to the different applications on different computers. At the receiving end, the data is forwarded to the correct application. This process is known as demultiplexing. TCP transmits the packet to the correct application by using the logical channels known as ports.
- **Logical Connections:** The combination of sockets, sequence numbers, and window sizes, is called a logical connection. Each connection is identified by the pair of sockets used by sending and receiving processes.
- **Full Duplex:** TCP provides Full Duplex service, i.e., the data flow in both the directions at the same time. To achieve Full Duplex service, each TCP should have sending and receiving buffers so that the segments can flow in both the directions. TCP is a connection-oriented protocol. Suppose the process A wants to send and receive the data from process B. The following steps occur:
  - Establish a connection between two TCPs.
  - Data is exchanged in both the directions.
  - The Connection is terminated.

### TCP Segment Format

Source port address 16 bits								Destination port address 16 bits							
Sequence number 32 bits															
Acknowledgement number 32 bits															
HLEN 4 bits		Reserved 6 bits		U R G	A C K	P S H	R S T	S Y N	F I N	Window size 16 bits					
Checksum 16 bits										Urgent pointer 16 bits					
Options & padding															

Where,

- **Source port address:** It is used to define the address of the application program in a source computer. It is a 16-bit field.
- **Destination port address:** It is used to define the address of the application program in a destination computer. It is a 16-bit field.
- **Sequence number:** A stream of data is divided into two or more TCP segments. The 32-bit sequence number field represents the position of the data in an original data stream.
- **Acknowledgement number:** A 32-bit acknowledgement number acknowledges the data from other communicating devices. If ACK field is set to 1, then it specifies the sequence number that the receiver is expecting to receive.
- **Header Length (HLEN):** It specifies the size of the TCP header in 32-bit words. The minimum size of the header is 5 words, and the maximum size of the header is 15 words. Therefore, the maximum size of the TCP header is 60 bytes, and the minimum size of the TCP header is 20 bytes.
- **Reserved:** It is a six-bit field which is reserved for future use.
- **Control bits:** Each bit of a control field functions individually and independently. A control bit defines the use of a segment or serves as a validity check for other fields.

There are total six types of flags in control field:

- **URG:** The URG field indicates that the data in a segment is urgent.
- **ACK:** When ACK field is set, then it validates the acknowledgement number.
- **PSH:** The PSH field is used to inform the sender that higher throughput is needed so if possible, data must be pushed with higher throughput.
- **RST:** The reset bit is used to reset the TCP connection when there is any confusion occurs in the sequence numbers.
- **SYN:** The SYN field is used to synchronize the sequence numbers in three types of segments: connection request, connection confirmation ( with the ACK bit set ), and confirmation acknowledgement.
- **FIN:** The FIN field is used to inform the receiving TCP module that the sender has finished sending data. It is used in connection termination in three types of segments: termination request, termination confirmation, and acknowledgement of termination confirmation.

- **Window Size:** The window is a 16-bit field that defines the size of the window.
- **Checksum:** The checksum is a 16-bit field used in error detection.
- **Urgent pointer:** If URG flag is set to 1, then this 16-bit field is an offset from the sequence number indicating that it is a last urgent data byte.
- **Options and padding:** It defines the optional fields that convey the additional information to the receiver.

<b>Basis for Comparison</b>	<b>TCP</b>	<b>UDP</b>
Definition	TCP establishes a virtual circuit before transmitting the data.	UDP transmits the data directly to the destination computer without verifying whether the receiver is ready to receive or not.
Connection Type	It is a Connection-Oriented protocol	It is a Connectionless protocol
Speed	slow	high
Reliability	It is a reliable protocol.	It is an unreliable protocol.
Header size	20 bytes	8 bytes
acknowledgement	It waits for the acknowledgement of data and has the ability to resend the lost packets.	It neither takes the acknowledgement, nor it retransmits the damaged frame.

## **PERFORMANCE ISSUES:**

### **Responsibilities of the transport layer**

The responsibilities of the transport layer are as follows –

- It provides a process to process delivery or end to end delivery of the entire message from the sender to the receiver.
- This layer checks for errors during transmission.
- It controls the flow control mechanism and prevents data loss due to the speed mismatch of the sender and receiver.
- This layer divides the stream of bytes received from the upper layer into segments at the sender side and reassembles at the receiver side.

### **Challenges**

The main challenges to designing a transport layer protocol are given below –

- **Dynamic Topology** – Technology is changing day by day and it affects the performance of the transport layer and will be slightly affected by these changes.
- **Power and Bandwidth constraints** – In a wireless network, two main constraints of power and bandwidth are faced. These constraints affect the transport layer.
- **To handle congestion control, reliability and flow control separately** – If we handle congestion control, reliability and flow control separately then the performance of the transport layer is increased. But to handle these separately is the additional control overhead



## UNIT V

### APPLICATION LAYER AND NETWORK SECURITY

#### DNS - Domain Name System

Users prefer to refer to hosts, mailboxes, and other resources not by their binary network addresses but using some ASCII strings, such as `tana@art.ucsb.edu`. Nevertheless, the network itself only understands binary addresses, so some mechanism is required to convert the ASCII string to network addresses. Below we will describe how this mapping is accomplished in the Internet.

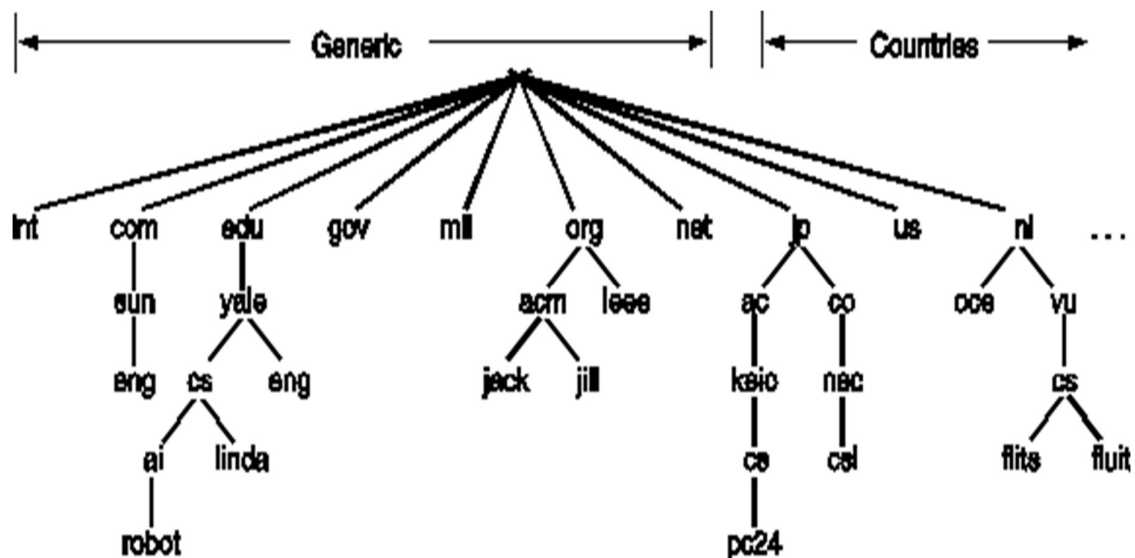
The mapping is done by DNS (the Domain Name System).

The essence of DNS is a hierarchical, domain-based naming scheme and a distributed database system for implementing this naming scheme. It is primarily used for mapping host names and email destinations to IP addresses but can also be used for other purposes. DNS is defined in RFC 1034 and 1035.

The basic scheme of the use of DNS is the following: To map a name onto an IP address, an application program calls a library procedure called the resolver, passing it the name as a parameter. The resolver sends a UDP packet to a local DNS server, which then looks up the name and returns the IP address to the resolver, which then returns it to caller. Armed with the IP address, the program then establish a TCP connection with the destination, or send it UDP packets.

#### The DNS Name Space

Conceptually, the Internet is divided into several hundred top-level *domains*, where each domain covers many hosts. Each domain is partitioned into subdomains, and these are further partitioned, and so on. All these domains can be represented by a tree as in Fig. The leaves of the tree represent domains that have no subdomains. A leaf domain may contain a single host, or it may represent a company and contains thousands of hosts.



The top-level domains are of two kinds: generic and countries. The generic domains are `com` (commercial), `edu` (educational institutions), `gov` (the U.S. federal government), `int` (certain international organizations), `mil` (the U.S. armed forces), `net` (network providers), and `org` (nonprofit organizations). The country domains include one entry for every country, as defined in ISO 3166.

Each domain is named by the path upward from it to the (unnamed) root. The components are separated by periods (pronounced "dot"). Thus the Faculty of Mathematics and Physics of Comenius University is `fmph.uniba.sk`.

Domain names are case insensitive, so `edu` and `EDU` mean the same thing. Component names can be up to 63 characters long, and full path name must not exceed 255 characters.

In principle, domains can be inserted into the tree in two different ways. For example, `cs.yale.edu` could equally well be listed under the US country domain as `cs.yale.ct.us`. In practice, however, nearly all organizations in the U.S. are under a generic domain, and nearly all outside the U.S. are under the domain of their country. There is no rule against registering under two top-level domains, but doing so might be confusing, so few organizations do it.

Each domain controls how it allocates the domains under it. To create a new domain, permission is required of the domain in which it will be included. In this way, name conflicts are avoided. Once a new domain has been created and registered, it can create subdomains without getting permission from anybody higher up the tree.

Naming follows organizational boundaries, not physical networks.

### Resource Records

Every domain can have a set of resource records associated with it. For a single host, the most common record is just its IP address, but many other kinds of resource records also exist. When a resolver gives a domain name to DNS, what it gets back are the resource records associated with that name. Thus the real function of DNS is to map domain names onto resource records.

A resource record is a five-tuple. Although they are encoded in binary, in most expositions resource records are presented in ASCII text, one line per resource record. The format we will use is as follows:

Domain name	Time_to_live	Type	Class	Value
-------------	--------------	------	-------	-------

The `Domain_name` tells the domain to which this record applies. Normally, many records exist for each domain. When a query is made about a domain, all the matching records of the type requested are returned. The `Time_to_live` field gives an indication of how stable the record is. Information that is highly stable is assigned a large value, such as 86400 (the number of seconds in 1 day). Information that are highly volatile is assigned a small value such as 60 (1 minute).

The `Type` field tells, what kind of record this is. The most important types are:

- **SOA** - Start of Authority. Provides the name of the primary source of information about the name server's zone and some further information about it.
- **A** - IP address of a host. It is the most important record type. It holds a 32-bit IP address for some host. If a host has more network connections, and so more IP addresses, it has a resource record for each of them.
- **MX** - Mail exchange. It specifies the name of the host prepared to accept email for the specified domain.

- NS - Name server. It specifies name servers. For example, every DNS database normally has an NS record for each of the top-level domains.
- CNAME - Canonical name. This record allows aliases to be created.
- PTR - Pointer. This is an alias for an IP address. Records of this type are nearly always used to associate a name with an IP address to allow lookups of the IP address and return the name of the corresponding machine. We omit the details of this process here.
- HINFO - Host description. This record allow people to find out what kind of machine and operating system a domain corresponds to.
- TXT - Text. This record allows domains to identify themselves in arbitrary ways.

The Class field is always equal IN for Internet information. For non-Internet information, other codes can be used.

The Value field can contain a number, a domain name, or an ASCII string. The semantics depends on the record type. A short description of the Value fields for each of the principal record types is given in Fig.

Type	Meaning	Value
SOA	Start of Authority	Parameters for this zone
A	IP address of a host	32-Bit integer
MX	Mail exchange	Priority, domain willing to accept email
NS	Name Server	Name of a server for this domain
CNAME	Canonical name	Domain name
PTR	Pointer	Alias for an IP address
HINFO	Host description	CPU and OS in ASCII
TXT	Text	Uninterpreted ASCII text

*The principal DNS resource record types.*

As an example of the kind of information one can find in the DNS database of a domain, see Fig. 7-27. This figure depicts part of a database for the cs.vu.nl domain shown in Fig. 7-25. The database contains seven types of resource records.

```

; Authoritative data for cs.vu.nl
cs.vu.nl.      86400  IN SOA  star boss (952771,7200,7200,2419200,86400)
cs.vu.nl.      86400  IN TXT  "Faculteit Wiskunde en Informatica."
cs.vu.nl.      86400  IN TXT  "Vrije Universiteit Amsterdam."
cs.vu.nl.      86400  IN MX    1 zephyr.cs.vu.nl.
cs.vu.nl.      86400  IN MX    2 top.cs.vu.nl.

flits.cs.vu.nl. 86400  IN HINFO Sun Unix
flits.cs.vu.nl. 86400  IN A      130.37.16.112
flits.cs.vu.nl. 86400  IN A      192.31.231.165
flits.cs.vu.nl. 86400  IN MX    1 flits.cs.vu.nl.
flits.cs.vu.nl. 86400  IN MX    2 zephyr.cs.vu.nl.
flits.cs.vu.nl. 86400  IN MX    3 top.cs.vu.nl.
www.cs.vu.nl. 86400  IN CNAME star.cs.vu.nl
ftp.cs.vu.nl.  86400  IN CNAME zephyr.cs.vu.nl

rowboat        IN A      130.37.56.201
               IN MX    1 rowboat
               IN MX    2 zephyr
               IN HINFO Sun Unix

little-sister  IN A      130.37.62.23
               IN HINFO Mac MacOS

laserjet       IN A      192.31.231.216
               IN HINFO "HP Laserjet III" Proprietary

```

*A portion of a possible DNS database for cs.vu.nl*

The first non-comment line of Fig. gives some basic information about the domain, which will not concern us further.

The next two lines give textual information about where the domain is located.

Then come two entries giving the first and second places to try to deliver email sent to person@cs.vu.nl. The zephyr (a specific machine) should be tried first. If that fails, the top should be tried next.

Next 3 lines tell that the flits is a Sun workstation running UNIX and give both of its IP addresses.

Further three lines give choices for handling email sent to flits.cs.vu.nl.

Next comes an alias, www.cs.vu.nl, so this address can be used without designating a specific machine.

Similarly ftp.cs.vu.nl.

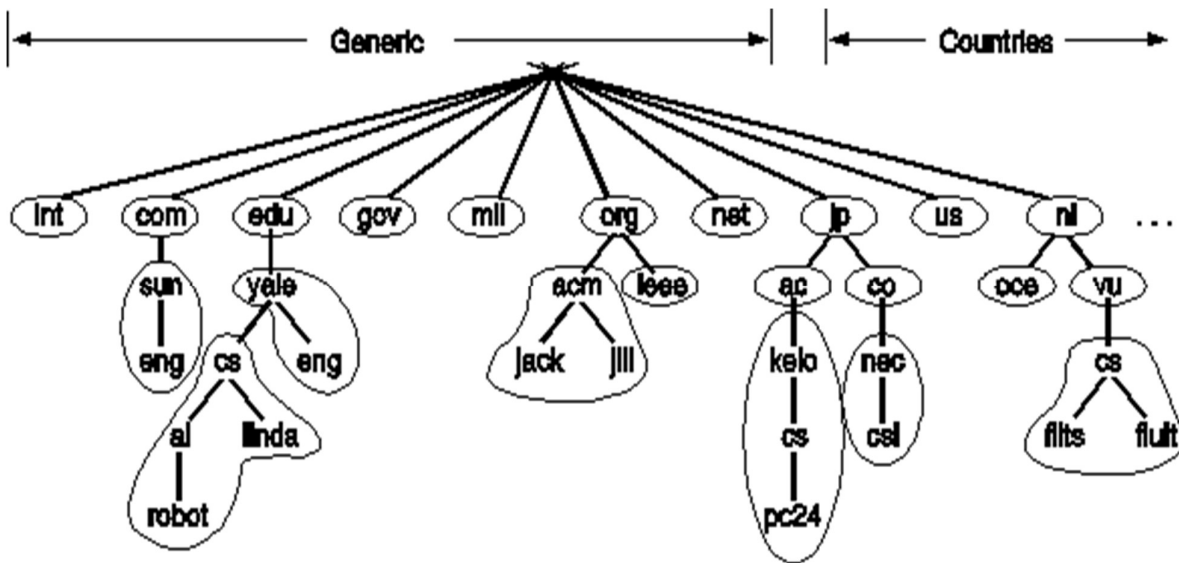
The next four lines contain a typical entry for a workstation, in this case rowboat.cs.vu.nl. The information provided contains the IP address, the primary and secondary mail drops, and information about the machine.

Then comes an entry for a non-UNIX system that is not capable of receiving mail itself, followed by an entry for a laser printer.

IP addresses for root servers needed to look up distant hosts are not in this file. They are present in a system configuration file loaded into the DNS cache when the server is booted. They have very long timeouts, so once loaded, they are never purged from the cache.

### Name Servers

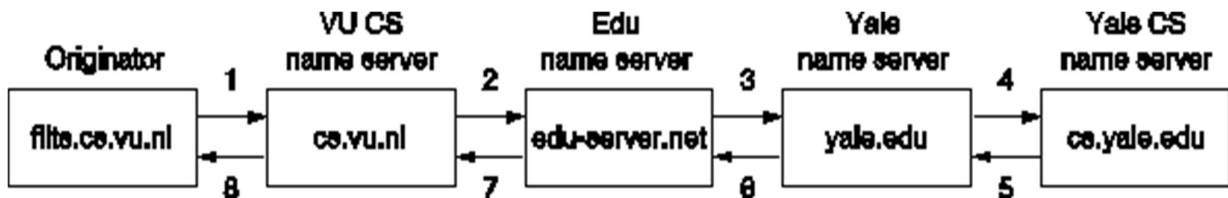
In practice, one single name server cannot contain the entire DNS database. So the DNS name space is divided up into non overlapping zones and each zone contains name servers holding the authoritative information about that zone. Normally, a zone will have one primary name server, which get its information from a file on its disk, and one or more secondary name servers, get their information from the primary name server.



Part of the DNS name space showing the division into zones.

When a resolver has a query about a domain name, it passes the query to one of the local name servers. If the domain being sought falls under the jurisdiction of the name server, it returns the authoritative resource records. An authoritative record is one that comes from the authority that manages the record, and thus is always correct. Authoritative records are in contrast with cached records, which may be out of date.

If, however, the domain is remote and no information about the requested domain is available locally, the name server sends a query message to the top-level name server for the domain requested. If it also does not know the answer, it sends it to one of its children, and so on. When a server with the authoritative resource record is encountered, the response is sent back through single name servers in the chain. For an example, see Fig, here the IP address of the host linda.cs.yale.edu was sought by the resolver on flits.cs.vu.nl.



How a resolver looks up a remote name in eight steps.

Once the record get back to the name server cs.vu.nl, it will be entered into a cache there, in case it is needed later. However, this information is not authoritative, so it should not live too long. This is the reason that the Time\_to\_live field is included in each resource record. It tells remote name servers how long to cache records.

The query method described here is known as a *recursive query*. An alternative form is also possible. In this form, when a query cannot be satisfied locally, the query fails, but the name of the next server on the line to try is returned. This procedure gives the client more control over the search process.

When a DNS client fails to get a response before its timer goes off, it normally will try another server next time.

## Electronic Mail In Computer Networks

### Introduction

**Electronic mail (e-mail)** is a computer-based program that allows users to send and receive messages. E-mail is the electronic version of a letter, but with time and flexibility advantages. While a letter can take anywhere from a week to a couple of months to reach its intended destination, an e-mail is sent virtually almost instantly. Messages in the mail contain not just text but also photos, audio, and video data. A person sending an e-mail is a **sender**, and the person receiving it is the **recipient**.

### What is Electronic Mail in Computer Networks?

Electronic mail is one of the most well-known network services. Electronic mail is a computer-based service that allows users to communicate with one another by exchanging messages. Email information is transmitted via email servers and uses a variety of TCP/IP protocols. For example, the simple mail transfer protocol (SMTP) is a protocol that is used to send messages. Similarly, IMAP or POP receives messages from a mail server.

### Features of Electronic Mail

- **Spontaneity:** In a couple of seconds, you may send a message to anybody on the globe.
- **Asynchronous:** You may send the e-mail and let the recipient view it at their leisure.
- Attachments of data, pictures, or music, frequently in compressed forms, can be delivered as an e-mail to a person anywhere in the world.
- Addresses can be stored in an address book and retrieved instantly.
- Through an e-mail, a user can transfer multiple copies of a message to various individuals.

### Services offered by Electronic Mail

**Composition:** Creating messages and responses is referred to as composition.

**Transfer:** Sending mail from the sender to the receiver is known as a transfer.

**Reporting:** Mail delivery confirmation is known as reporting. It allows users to see if their mail has been delivered, misplaced, or rejected.

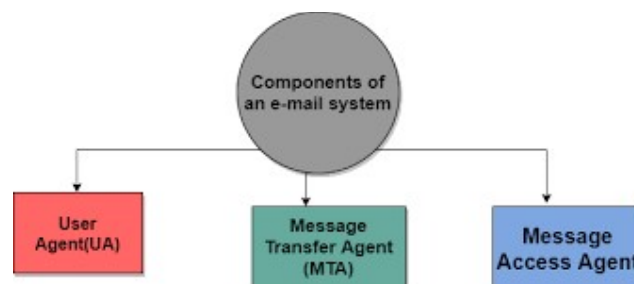
**Displaying:** It refers to presenting messages so that the user can understand them.

**Disposition:** This stage concerns the recipient's actions after receiving mail, such as saving it, deleting it before reading it, or after reading it.

### Components of Electronic Mail

The following are the essential components of an e-mail system:

1. User Agent (UA)
2. Message Transfer Agent (MTA)
3. Message Access Agent



## **User Agent (UA)**

The User-Agent is a simple software that sends and receives mail. It is also known as a mail reader. It supports a wide range of instructions for sending, receiving, and replying to messages and manipulating mailboxes.

Some of the services supplied by the User-Agent are listed below:

- Reading a Message
- Sending a reply to a Message
- Message Composition
- Forwarding a Message
- Handling the Message

## **Message Transfer Agent**

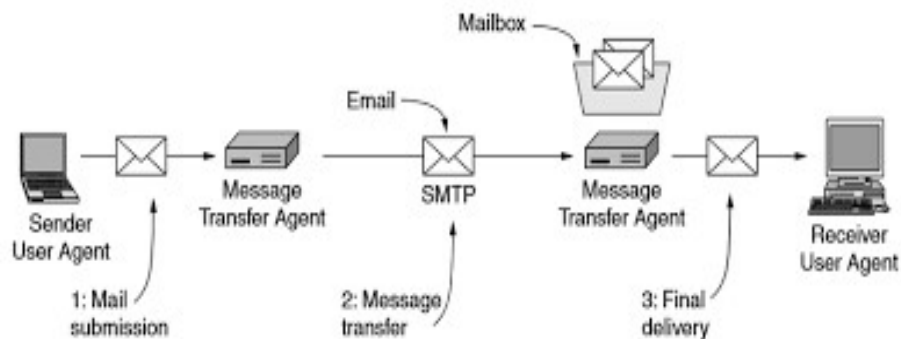
The Message Transfer Agent manages the actual e-mail transfer operation (MTA). Simple Mail Transfer Protocol sends messages from one MTA to another. A system must have a client MTA and a system MTA to send an e-mail. If the recipients are connected to the same computer, it sends mail to their mailboxes. If the destination mailbox is on another computer, it sends mail to the receiver's MTA.

## **Message Access Agent**

The Simple Mail Transfer Protocol is used for the first and second stages of e-mail delivery.

The pull protocol is mainly required at the third stage of e-mail delivery, and the message access agent is used at this point.

## **Architecture of Electronic Mail**



## **Format of E-mail :**

An e-mail consists of three parts that are as follows :

1. Envelope
2. Header
3. Body

These are explained as following below.

### **1. Envelope:**

The envelope part encapsulates the message. It contains all information that is required for sending any e-mail such as destination address, priority and security level. The envelope is used by MTAs for routing message.

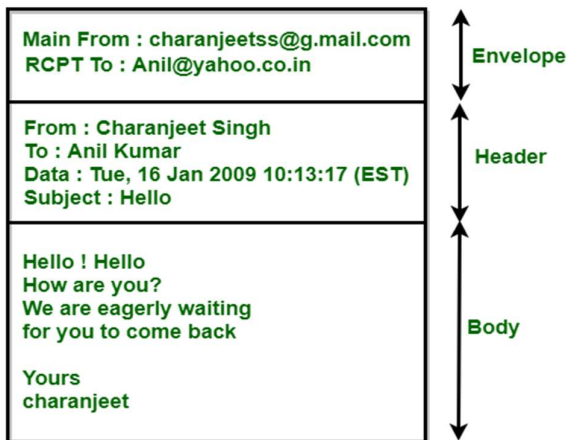
## 2. Header:

The header consists of a series of lines. Each header field consists of a single line of ASCII text specifying field name, colon and value. The main header fields related to message transport are :

1. **To:** It specifies the DNS address of the primary recipient(s).
2. **Cc :** It refers to carbon copy. It specifies address of secondary recipient(s).
3. **BCC:** It refers to blind carbon copy. It is very similar to Cc. The only difference between Cc and Bcc is that it allow user to send copy to the third party without primary and secondary recipient knowing about this.
4. **From :** It specifies name of person who wrote message.
5. **Sender :** It specifies e-mail address of person who has sent message.
6. **Received :** It refers to identity of sender's, data and also time message was received. It also contains the information which is used to find bugs in routing system.
7. **Return-Path:** It is added by the message transfer agent. This part is used to specify how to get back to the sender.

**3. Body:** The body of a message contains text that is the actual content/message that needs to be sent, such as “Employees who are eligible for the new health care program should contact their supervisors by next Friday if they want to switch.” The message body also may include signatures or automatically generated text that is inserted by the sender's email system.

The above-discussed field is represented in tabular form as follows:



In addition to above-discussed fields, the header may also contain a variety of other fields which are as follows :

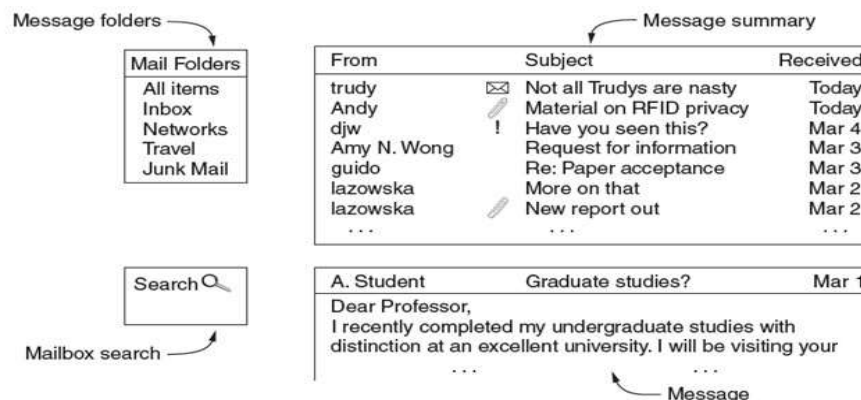
Header	Meaning
Date:	Date and time when the message was sent.
Reply-To:	It contains e-mail address to which replies should be sent.
Message-Id:	It refers to the unique number for referencing this message later.



In-Reply-To:	Message-Id of a message to which this is as a reply.
References:	It contains other relevant message-ids.
Keywords:	User-chosen keywords.
Subject:	It contains short summary of message for one-line display.

### User Agent:

A user agent is any software, acting on behalf of a user, which "retrieves, renders and facilitates end-user interaction with Web content. A user agent is therefore a special kind of software agent. Some prominent examples of user agents are web browsers and email readers. Often, a user agent acts as the client in a client–server system. In some contexts, such as within the Session Initiation Protocol (SIP), the term *user agent* refers to both end points of a communications session.



### Message Format:

#### RFC 5322 - The Internet Message Format:

RFC 5322 is a standard that defines the format of internet messages, such as email messages. It specifies the structure and content of email messages, including the headers, body, and attachments. The standard is maintained by the Internet Engineering Task Force (IETF) and is an important reference for anyone working with email or other internet messages. It is also known as the Internet Message Format Standard.

Here are a few more points about RFC 5322 –

- It replaces an earlier standard called RFC 822, which was published in 1982.
- It is written in a format called Augmented BNF, which is a formal notation used to describe the syntax of computer languages.
- It specifies the following components of an internet message –
  - The envelope, which contains information about the message's origin, destination, and routing.
  - The header, which contains information about the message, such as the subject, sender, recipient, and date.
  - The body, which contains the main content of the message.
  - The attachments, which are files that are included with the message.

- It defines rules for formatting and encoding the various components of an internet message, as well as rules for handling errors and special cases.
- It is widely used as a reference for implementing email and other internet message systems, and is an important part of the internet's infrastructure.

### History

RFC 5322 is the latest version of a standard for formatting internet messages that has evolved over time. The first version of this standard was published in 1982 as RFC 822 (Standard for ARPA Internet Text Messages). This standard was later updated and replaced by RFC 2822 in 2001, which was in turn updated and replaced by RFC 5322 in 2008.

The original standard, RFC 822, was developed by the Internet Engineering Task Force (IETF) as a way to standardize the format of internet messages, including email. It was based on an earlier standard called RFC 733 (Standard for the Format of ARPA Network Text Messages) that was developed in 1977, but added several new features and made other improvements.

Over the years, the standard has been updated to reflect changes in technology and the way that internet messages are used. For example, RFC 5322 includes support for internationalized email addresses and includes additional rules for handling spam and other types of unwanted messages.

### Standards and Organization

The Internet Engineering Task Force (IETF) is a standards organization that develops and promotes voluntary Internet standards. It is an open, international community of network designers, operators, vendors, and researchers concerned with the evolution of the Internet architecture and the smooth operation of the Internet.

The IETF works through a series of working groups, each focused on a specific area of Internet technology. These working groups develop documents called Requests for Comments (RFCs), which are published by the IETF and become Internet standards.

RFC 5322 is one of these standards, published by the IETF as a way to define the format of internet messages, including email. It is an important reference for anyone working with email or other internet messages, and is widely used as a reference for implementing email and other internet message systems.

### Working Principle:

RFC 5322 helps us by providing a standard way to format internet messages, such as email. By using a common format, email systems can interoperate and exchange messages with each other. This allows people to send and receive email from any device or email service, as long as it is compliant with the standard.

In addition, the standard helps to ensure that email messages are properly formatted and easy to read. It defines rules for the structure and content of email messages, including the headers, body, and attachments. It also defines rules for handling errors and special cases, which helps to prevent confusion and miscommunication.

Overall, RFC 5322 helps to make the Internet a more connected and reliable place by providing a common language for email and other internet messages.

RFC 5322 is a standard that defines the format of internet messages, such as email. It is maintained by the Internet Engineering Task Force (IETF) and is an important reference for anyone working with email or other internet messages. The standard specifies the structure and content of email messages, including the envelope, header, body, and attachments, and defines rules for formatting and encoding these components. It is widely used as a reference for implementing email and other internet message systems, and is an important part of the internet's infrastructure.

#### MIME – The Multipurpose Internet Mail Extensions:

MIME stands for Multipurpose Internet Mail Extensions. It is used to extend the capabilities of Internet e-mail protocols such as SMTP. The MIME protocol allows the users to exchange various types of digital content such as pictures, audio, video, and various types of documents and files in the e-mail. MIME was created in 1991 by a computer scientist named Nathan Borenstein at a company called Bell Communications.

MIME is an e-mail extension protocol, i.e., it does not operate independently, but it helps to extend the capabilities of e-mail in collaboration with other protocols such as SMTP. Since MIME was able to transfer only text written file in a limited size English language with the help of the internet. At present, it is used by almost all e-mail related service companies such as Gmail, Yahoo-mail, Hotmail.

#### Need of MIME Protocol

MIME protocol is used to transfer e-mail in the computer network for the following reasons:

1. The MIME protocol supports multiple languages in e-mail, such as Hindi, French, Japanese, Chinese, etc.
2. Simple protocols can reject mail that exceeds a certain size, but there is no word limit in MIME.
3. Images, audio, and video cannot be sent using simple e-mail protocols such as SMTP. These require MIME protocol.
4. Many times, emails are designed using code such as HTML and CSS, they are mainly used by companies for marketing their product. This type of code uses MIME to send email created from HTML and CSS.

#### MIME Header

MIME adds five additional fields to the header portion of the actual e-mail to extend the properties of the simple email protocol. These fields are as follows:

1. MIME Version
2. Content Type
3. Content Type Encoding
4. Content Id
5. Content description

##### **1. MIME Version**

It defines the version of the MIME protocol. This header usually has a parameter value 1.0, indicating that the message is formatted using MIME.

##### **2. Content Type**

It describes the type and subtype of information to be sent in the message. These messages can be of many types such as Text, Image, Audio, Video, and they also have many subtypes such that the subtype of the image can be png or jpeg. Similarly, the subtype of Video can be WEBM, MP4 etc.

### 3. Content Type Encoding

In this field, it is told which method has been used to convert mail information into ASCII or Binary number, such as 7-bit encoding, 8-bit encoding, etc.

### 4. Content Id

In this field, a unique "Content Id" number is appended to all email messages so that they can be uniquely identified.

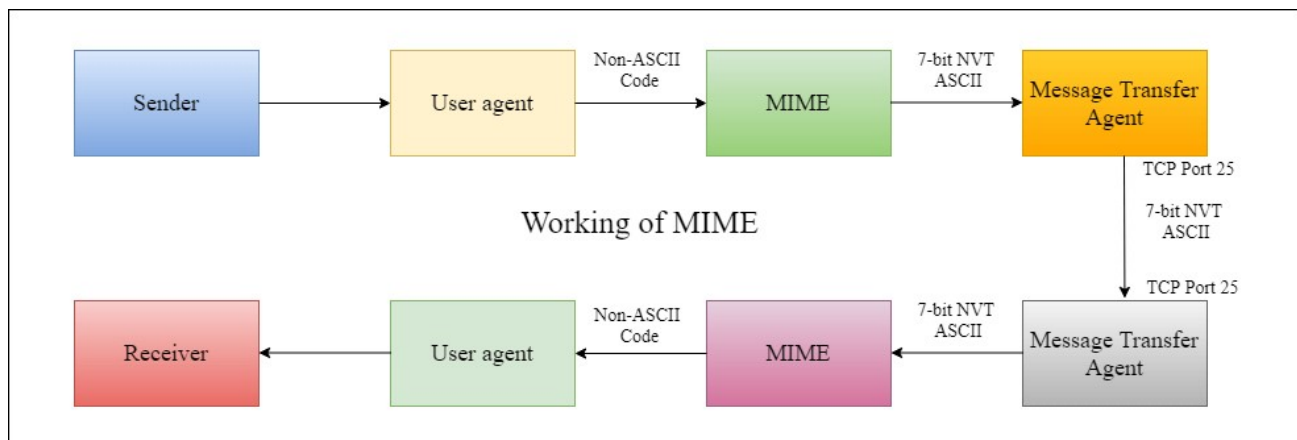
### 5. Content description

This field contains a brief description of the content within the email. This means that information about whatever is being sent in the mail is clearly in the "Content Description". This field also provides the information of name, creation date, and modification date of the file.

#### Example of Content description

Content-Description: attachment; filename = javatpoint.jpeg;  
modification-date = "Wed, 12 Feb 1997 16:29:51 -0500";

Working diagram of MIME Protocol



#### Features of MIME Protocol

1. It supports multiple attachments in a single e-mail.
2. It supports the non-ASCII characters.
3. It supports unlimited e-mail length.
4. It supports multiple languages.

#### Advantage of the MIME

The MIME protocol has the following advantages:

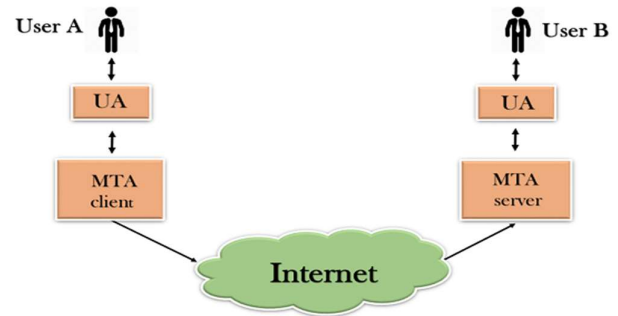
1. It is capable of sending various types of files in a message, such as text, audio, video files.
2. It also provides the facility to send and receive emails in different languages like Hindi, French, Japanese, Chinese etc.
3. It also provides the facility of connecting HTML and CSS to email, due to which people can design email as per their requirement and make it attractive and beautiful.

4. It is capable of sending the information contained in an email regardless of its length.
5. It assigns a unique id to all e-mails

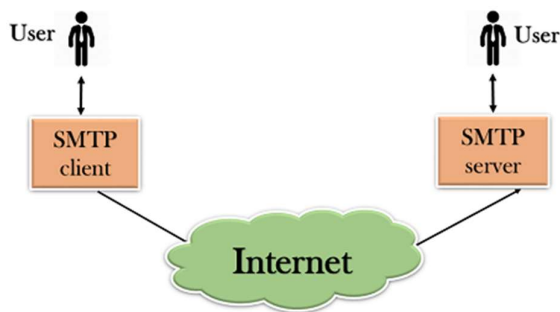
## Message Transfer

### SMTP

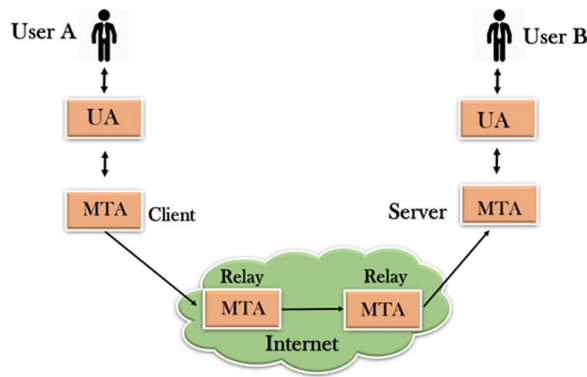
- SMTP stands for Simple Mail Transfer Protocol.
- SMTP is a set of communication guidelines that allow software to transmit an electronic mail over the internet is called **Simple Mail Transfer Protocol**.
- It is a program used for sending messages to other computer users based on e-mail addresses.
- It provides a mail exchange between users on the same or different computers, and it also supports:
  - It can send a single message to one or more recipients.
  - Sending message can include text, voice, video or graphics.
  - It can also send the messages on networks outside the internet.
- The main purpose of SMTP is used to set up communication rules between servers. The servers have a way of identifying themselves and announcing what kind of communication they are trying to perform. They also have a way of handling the errors such as incorrect email address. For example, if the recipient address is wrong, then receiving server reply with an error message of some kind.



### Components of SMTP



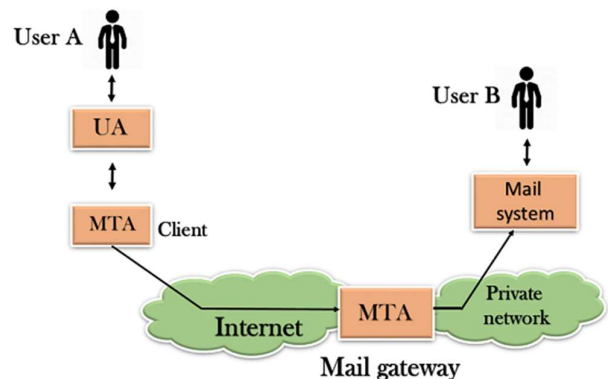
- First, we will break the SMTP client and SMTP server into two components such as user agent (UA) and mail transfer agent (MTA). The user agent (UA) prepares the message, creates the envelope and then puts the message in the envelope. The mail transfer agent (MTA) transfers this mail across the internet.
- SMTP allows a more complex system by adding a relaying system. Instead of just having one MTA at sending side and one at receiving side, more MTAs can be added, acting either as a client or server to relay the email.



- The relaying system without TCP/IP protocol can also be used to send the emails to users, and this is achieved by the use of the mail gateway. The mail gateway is a relay MTA that can be used to receive an email.

### Working of SMTP

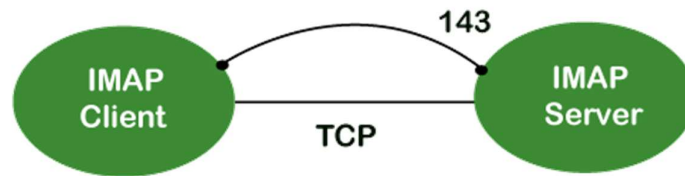
1. **Composition of Mail:** A user sends an e-mail by composing an electronic mail message using a Mail User Agent (MUA). Mail User Agent is a program which is used to send and receive mail. The message contains two parts: body and header. The body is the main part of the message while the header includes information such as the sender and recipient address. The header also includes descriptive information such as the subject of the message. In this case, the message body is like a letter and header is like an envelope that contains the recipient's address.
2. **Submission of Mail:** After composing an email, the mail client then submits the completed e-mail to the SMTP server by using SMTP on TCP port 25.
3. **Delivery of Mail:** E-mail addresses contain two parts: username of the recipient and domain name. For example, vivek@gmail.com, where "vivek" is the username of the recipient and "gmail.com" is the domain name.  
If the domain name of the recipient's email address is different from the sender's domain name, then MSA will send the mail to the Mail Transfer Agent (MTA). To relay the email, the MTA will find the target domain. It checks the MX record from Domain Name System to obtain the target domain. The MX record contains the domain name and IP address of the recipient's domain. Once the record is located, MTA connects to the exchange server to relay the message.
4. **Receipt and Processing of Mail:** Once the incoming message is received, the exchange server delivers it to the incoming server (Mail Delivery Agent) which stores the e-mail where it waits for the user to retrieve it.
5. **Access and Retrieval of Mail:** The stored email in MDA can be retrieved by using MUA (Mail User Agent). MUA can be accessed by using login and password.



### IMAP Protocol

IMAP stands for **Internet Message Access Protocol**. It is an application layer protocol which is used to receive the emails from the mail server. It is the most commonly used protocols like POP3 for retrieving the emails.

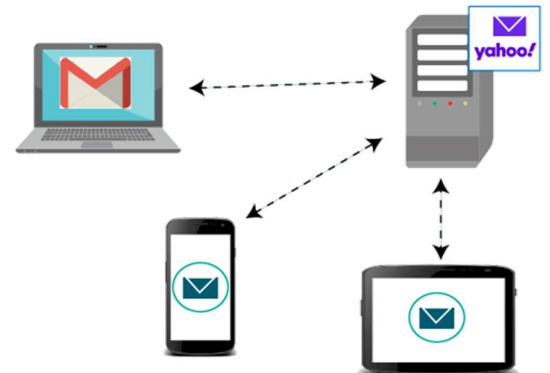
It also follows the client/server model. On one side, we have an IMAP client, which is a process running on a computer. On the other side, we have an IMAP server, which is also a process running on another computer. Both computers are connected through a network.



The IMAP protocol resides on the TCP/IP transport layer which means that it implicitly uses the reliability of the protocol. Once the TCP connection is established between the IMAP client and IMAP server, the IMAP server listens to the port 143 by default, but this port number can also be changed.

By default, there are two ports used by IMAP:

- Port 143: It is a non-encrypted IMAP port.
- Port 993: This port is used when IMAP client wants to connect through IMAP securely.



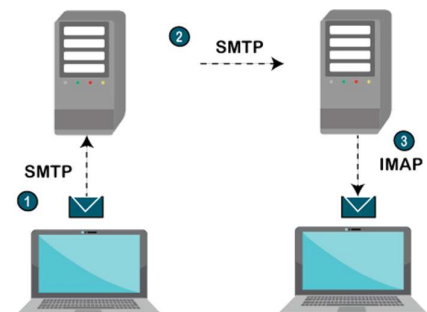
### IMAP General Operation

The IMAP is a client-server protocol like POP3 and most other TCP/IP application protocols. The IMAP4 protocol functions only when the IMAP4 must reside on the server where the user mailboxes are located. In contrast to POP3, the POP3 does not necessarily require the same physical server that provides the SMTP services. Therefore, in the case of the IMAP protocol, the mailbox must be accessible to both SMTP for incoming mails and IMAP for retrieval and modifications.

1. The IMAP uses the Transmission Control Protocol (TCP) for communication to ensure the delivery of data and also received in the order.
2. The IMAP4 listens on a well-known port, i.e., port number 143, for an incoming connection request from the IMAP4 client.

### **Let's understand the IMAP protocol through a simple example.**

The IMAP protocol synchronizes all the devices with the main server. Suppose user have three devices desktop, mobile, and laptop as shown in the above figure. If all these devices are accessing the same mailbox, then it will be synchronized with all the devices. Here, synchronization means that when mail is opened by one device, then it will be marked as opened in all the other devices, if we delete the mail, then the mail will also be deleted from all the other devices. So, we have synchronization between all the devices. In IMAP, we can see all the folders like spam, inbox, sent, etc. We can also create our own folder known as a custom folder that will be visible in all the other devices.



### Webmail:

Webmail is a cloud-based service or Web-based email system that allows you to access and use your email from almost anywhere through an internet connection. Unlike Thunderbird or Microsoft Outlook, it does not need software installation. It is a kind of service, which is provided by certain companies and ISPs (Internet service providers).

Especially, these kinds of server-based email systems are more popular for younger users. As with Microsoft Outlook, where emails are stored on-site in the hardware storage drive and logging into a connection with the server is needed to get email; so, in this situation, these services provide an appropriate option to email services.

For people who frequently away from their computers and use multiple devices, Webmail is a great solution for those people. Gmail, Hotmail, and other mainstream providers are the common examples of webmail from Yahoo!, which offer huge amounts of storage, and almost all are free.

They are very calm to set up and use. Although experts have pointed out, these models have advantages and limitations. With client-side email, users do not need an internet connection to review old emails as they can be archived directly on the computer. However, with webmail, you always need an internet connection to review mails as they are available via the dedicated servers over a network connection. Like some resident systems, webmail systems do not need communications protocols; that is one of another benefit of webmail. Some of the less tech-savvy users are frustrated by mail delivery failures while using continue resident or non-webmail systems, but a webmail product helps to prevent that issue.

Some popular webmail services

In modern times, many webmail services are available for users, which are not software-based. Below, a list contains some the free webmail services.

- **Gmail:** Gmail is a type of Webmail, a free Web-based e-mail service that allows users a gigabyte of storage for messages or other data. It is a very popular email service developed by Google. There are 1.5 billion active users of Gmail by October 2019.
- **Yahoo! Mail:** It is a web and cloud-based messaging solution that is launched by the American company Yahoo! on 8 October 1997. You can connect with your email with one-tap access to your inbox with the help of Yahoo! Mail, and it had 225 million users by January 2020. You can use to create Yahoo account by using this link -<https://overview.mail.yahoo.com/>
- **com:** It is a free web-based e-mail service that allows you to send and receive e-mail on your computer. Somewhat, it is like Google's Gmail service but something different in terms of linking desktop Outlook data. Outlook has two types of versions: Microsoft Outlook and Microsoft Outlook Express. To create an Outlook account, you can use this link <https://signup.live.com/?lic=1>
- **ProtonMail:** Unlike Gmail and Outlook.com., it uses client-side encryption to protect user data and email content, which is founded in 2013. To create a ProtonMail account, use this link - <https://protonmail.com/>
- **Zoho:** It holds a lot of potential for businesses, which is the first of the lesser-known free email accounts for making a list. It is an email service that very user-friendliness. It provides an easier way to accomplish all of your daily tasks by integrating with Google Drive, cloud-based file managers, Box.
- **GMX Mail:** GMX Mail is a free advertising-supported email service that may be accessed via POP3 and IMAP4 protocols as well as through webmail. It is provided by GMX (Global Mail eXchange) in Germany in 1997 that offers 65GB of storage.

### The World Wide Web:

World Wide Web, which is also known as a Web, is a collection of websites or web pages stored in web servers and connected to local computers through the internet. These websites contain text pages, digital images, audios, videos, etc. Users can access the content of these sites from any part of the world over the



internet using their devices such as computers, laptops, cell phones, etc. The WWW, along with internet, enables the retrieval and display of text and media to your device.



The building blocks of the Web are web pages which are formatted in HTML and connected by links called "hypertext" or hyperlinks and accessed by HTTP. These links are electronic connections that link related pieces of information so that users can access the desired information quickly. Hypertext offers the advantage to select a word or phrase from text and thus to access other pages that provide additional information related to that word or phrase.

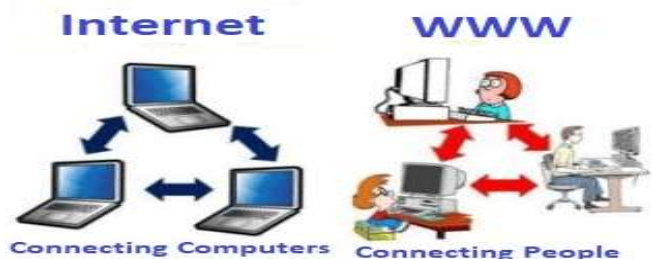
A web page is given an online address called a Uniform Resource Locator (URL). A particular collection of web pages that belong to a specific URL is called a website, e.g., [www.facebook.com](http://www.facebook.com), [www.google.com](http://www.google.com), etc. So, the World Wide Web is like a huge electronic book whose pages are stored on multiple servers across the world.

Small websites store all of their WebPages on a single server, but big websites or organizations place their WebPages on different servers in different countries so that when users of a country search their site they could get the information quickly from the nearest server.

So, the web provides a communication platform for users to retrieve and exchange information over the internet. Unlike a book, where we move from one page to another in a sequence, on World Wide Web we follow a web of hypertext links to visit a web page and from that web page to move to other web pages. You need a browser, which is installed on your computer, to access the Web.

#### Difference between World Wide Web and Internet:

Some people use the terms 'internet' and 'World Wide Web' interchangeably. They think they are the same thing, but it is not so. Internet is entirely different from WWW. It is a worldwide network of devices like computers, laptops, tablets, etc. It enables users to send emails to other users and chat with them online. For example, when you send an email or chatting with someone online, you are using the internet.



**Client-side** means that the processing takes place on the user's computer. It requires browsers to run the scripts on the client machine without involving any processing on the server.

**Server-side** means that the processing takes place on a web server.

This processing is important to execute the tasks required by the user on the web. Since the client-side script is executed on the client's computer, it is visible to the client. On the other hand, the server-side script is executed in the server; hence, it is not visible to the users.

#### Differences between client-side and server-side

##### **Client-side**

- Does not need interaction with the server
- Runs on the user's computer
- Reduces load on the server's processing unit

- Languages used: HTML, CSS, JavaScript

### Server-side

- Requires interaction with the server
- Runs on the web server
- Allows the server to provide dynamic websites tailored to the user. Increases the processing load on server.
- Languages used: PHP, ASP.net, Python

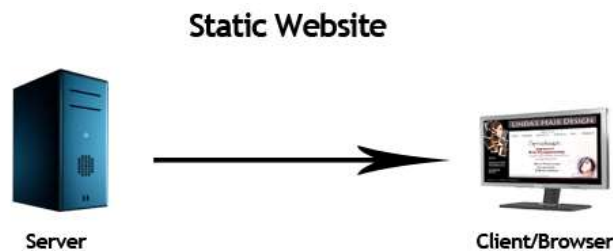
### Static Web Pages:

Static web pages are HTML pages that do not change their content or appearance when they are accessed by different users or at different times. They are designed to display the same information to all users and do not include any interactive elements or dynamic content.

Static web pages are typically used for basic information websites, such as brochure-style websites or personal websites, where the content does not need to change frequently. They are also used for content that does not require any user input or interaction, such as text, images, and other media.

Creating a static web page typically involves writing HTML code and linking to external resources, such as images and stylesheets. Static web pages can be created and edited using a text editor or a specialized HTML editor. They are typically hosted on a web server and accessed through a web browser.

Static web pages are simple and easy to create, but they do not offer the same level of interactivity and functionality as dynamic web pages. Dynamic web pages are generated by the server in real-time and can include interactive elements and dynamic content that changes based on user input or other variables.



### Advantages and disadvantages of static web pages

#### **Advantages**

There are several advantages to using static web pages, including –

**Simplicity** – Static web pages are simple to create and do not require any special programming skills or server-side processing. They are easy to edit and maintain and do not require a database or other complex backend systems.

**Fast loading** – Static web pages do not require any server-side processing and are typically served directly from the web server. This means that they can load faster than dynamic web pages, which may require additional processing time on the server.

**Better performance** – Static web pages can handle a higher volume of traffic and can be served more efficiently by the web server, as they do not require any server-side processing.

**Improved security** – Static web pages are less vulnerable to security threats, as they do not contain any dynamic content or interactivity. This makes them a good choice for websites that do not require any user input or that do not need to store sensitive information.

**Low maintenance** – Static web pages do not require any special maintenance or updates and can be left unchanged for long periods of time. This makes them a good choice for websites with limited resources or that do not need to change their content frequently.

Overall, static web pages are a good choice for simple websites that do not require any user input or dynamic content, or for websites that need to be served efficiently and with a high level of security.

### **Disadvantages**

There are several disadvantages to using static web pages, including –

**Lack of interactivity** – Static web pages do not offer any interactivity or dynamic content, which can limit their appeal and usability for users. They do not allow users to input data or interact with the website in any way, which can make them less engaging and less useful for certain types of content.

**Limited functionality** – Static web pages do not offer the same level of functionality as dynamic web pages, which can make them less useful for complex or interactive websites. They do not allow users to access data or perform tasks that require server-side processing, such as searching, sorting, or filtering data.

**Inflexibility** – Static web pages are not flexible and do not allow for the customization of content or the inclusion of dynamic elements, such as forms or polls. This can make them less suitable for websites that need to change their content frequently or that require user input.

**Maintenance** – Static web pages require manual updates and changes, which can be time-consuming and error-prone. This can make them less suitable for websites with large amounts of content or that need to change their content frequently.

**Limited analytics** – Static web pages do not provide any analytics or tracking information, which can make it difficult to understand how users are interacting with the website and to improve its performance.

Overall, static web pages are a good choice for simple websites that do not require any user input or dynamic content, but they may not be suitable for more complex or interactive websites that require greater flexibility and functionality.

HTML, CSS and JavaScript are the basic languages to build any website.

1. **Create the structure with HTML.** The first thing you have to learn, is HTML, which is the standard markup language for creating web pages.
2. **Style with CSS.** The next step is to learn CSS, to set the layout of your web page with beautiful colors, fonts, and much more.
3. **Make it interactive with JavaScript.** After studying HTML and CSS, you should learn JavaScript to create dynamic and interactive web pages for your users

### **HTML:**

- HTML stands for Hyper Text Markup Language.
- HTML is used to create web pages and web applications.
- HTML is widely used language on the web.
- We can create a static website by HTML only.
- Technically, HTML is a Markup language rather than a programming language.

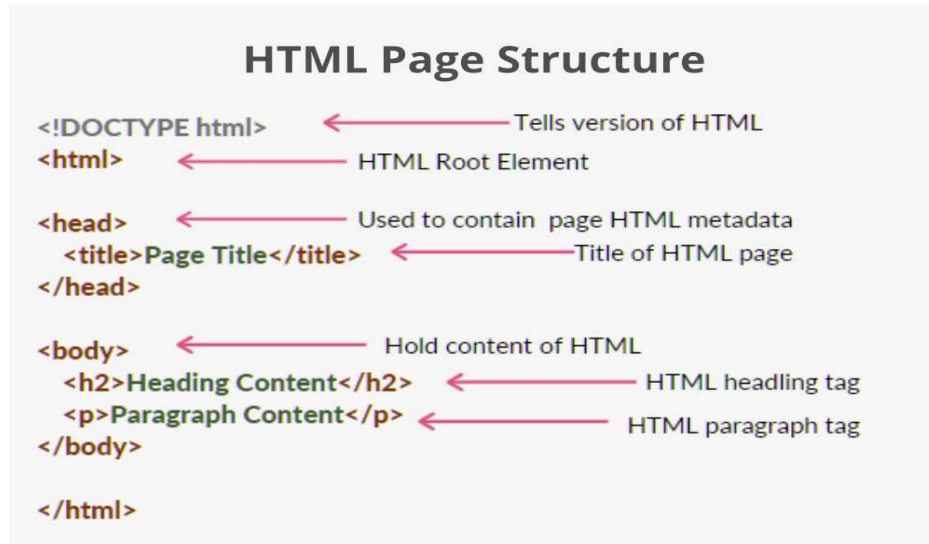
**HTML** stands for **Hyper Text Markup Language**. It is used to design web pages using the **markup language**. HTML is the combination of **Hypertext** and **Markup language**. Hypertext defines the link between the web pages and markup language defines the text document within the tag that define the structure of web pages.

**What is HTML used for ?**

HTML is used to create the structure of web pages that are displayed on the World Wide Web (www). It contains Tags and Attributes that are used to design the web pages. Also, we can link multiple pages using Hyperlinks.

### HTML Basic Format Page Structure

The basic structure of an HTML page is laid out below. It contains the essential building-block elements (i.e. doctype declaration, HTML, head, title, and body elements) upon which all web pages are created.



- **<DOCTYPE! html>** – A doctype or document type declaration is an instruction that tells the web browser about the markup language in which the current page is written. It is not an element or tag. The doctype declaration is not case-sensitive.
- **<html>** – This tag is used to define the root element of HTML document. This tag tells the browser that it is an HTML document. It is the second outer container element that contains all other elements within it.
- **<head>** – This tag is used to define the head portion of the HTML document that contains information related to the document. Elements within the head tag are not visible on the front-end of a webpage.
- **<body>** – The body tag is used to enclose all the visible content of a webpage. In other words, the body content is what the browser will show on the front end.

**Example 1:** This is the basic example of HTML that display the heading and paragraph content.

- HTML

```
<!DOCTYPE html>

<html>

<!-- Head Section content -->

<head>

  <!-- Page title -->

  <title>Basic Web Page</title>

</head>

<!-- Body Section content -->
```

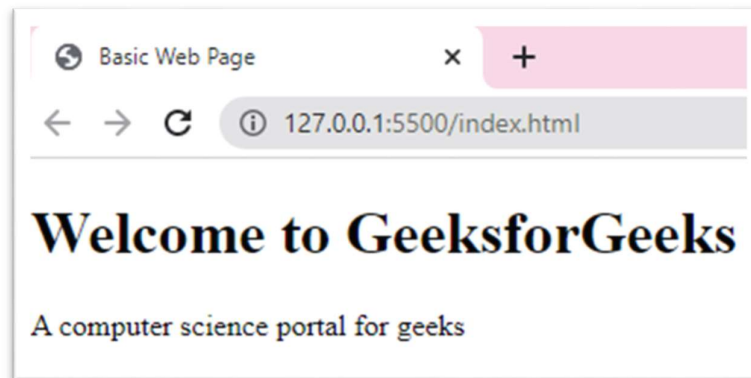
```

<body>
  <!-- Used to display heading content -->
  <h1>Welcome to GeeksforGeeks</h1>
  <!-- Used to display paragrapg content -->
  <p>A computer science portal for geeks</p>
</body>
</html>

```

Run on IDE

**Output:**



**Example 2:** This example describes how to create a simple form using HTML. To create a form, we will use `<form>` tag and inside form tag, we will use some input fields and label elements to display the form.

- HTML

```

<!DOCTYPE html>
<html lang="en">
<!-- Head Section Content -->
<head>
  <!-- Page title -->
  <title>Basic form design using HTML</title>
</head>
<!-- Body Section Content -->
<body>
  <!-- Heading Content -->
  <h1>GeeksforGeeks</h1>
  <h3>Basic form design using HTML</h3>
  <!-- Creating a from -->
  <form action="#">

```

```
<fieldset>

  <legend>Personal Details</legend>

  <!-- Label and input field -->

  <p>
    <label>First name : <input name="firstName" /></label>
  </p>

  <p>
    <label>Last name : <input name="lastName" /></label>
  </p>

  <!-- Label and radio button field -->

  <p>
    Gender :
    <label><input type="radio" name="gender"
      value="male" /> Male</label>
    <label><input type="radio" name="gender"
      value="female" /> Female</label>
  </p>

  <p>
    <label>Email : <input type="email"
      name="email" /></label>
  </p>

  <p>
    <label>Date of Birth : <input type="date"
      name="birthDate"></label>
  </p>

  <!-- Label and textarea field -->

  <p>
    <label>Address :
      <br />
      <textarea name="address" cols="30"
        rows="3"></textarea>
    </label>
  </p>
```

```
<!-- Creating a button -->

<p>

  <button type="submit">Submit</button>

</p>

</fieldset>

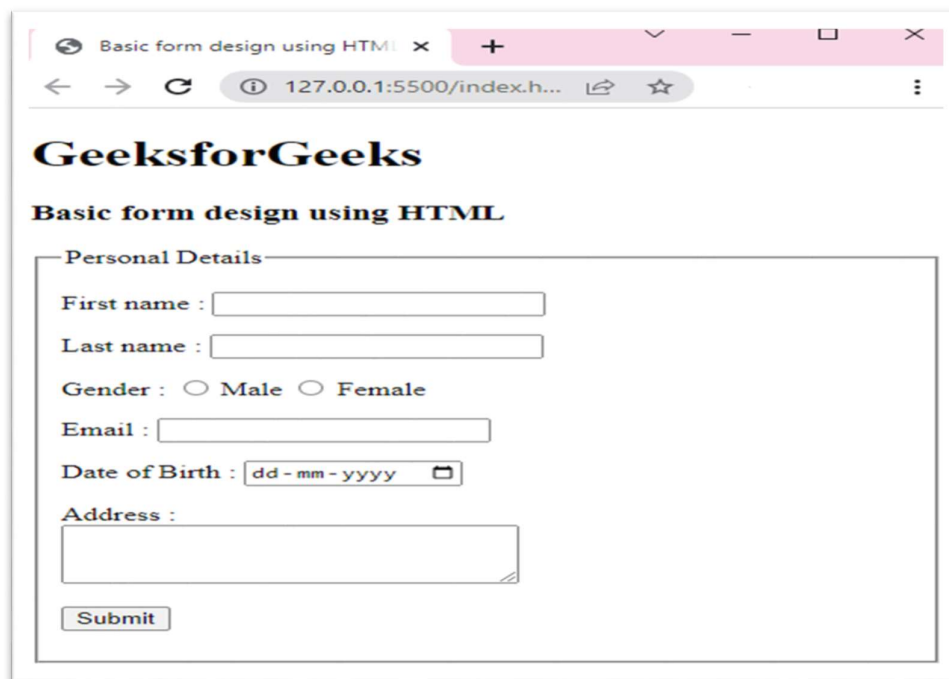
</form>

</body>

</html>
```

Run on IDE

**Output:**



The screenshot shows a web browser window with the title 'Basic form design using HTML'. The address bar shows '127.0.0.1:5500/index.h...'. The page content includes the 'GeeksforGeeks' logo and the heading 'Basic form design using HTML'. Below this is a form titled 'Personal Details' which contains the following fields: 'First name :', 'Last name :', 'Gender : ☐ Male ☐ Female', 'Email :', 'Date of Birth :' (with a date picker showing 'dd-mm-yyyy'), and 'Address :'. A 'Submit' button is located at the bottom of the form.

## Different Versions of HTML

Let's see the significance of the individual Versions of Html in details-

[All in One Software Development Bundle \(600+ Courses, 50+ projects\)](#)

[3000+ Hours of HD Videos](#) | [149 Learning Paths](#) | [600+ Courses](#) | [Verifiable Certificate of Completion](#) | [Lifetime Access](#)

[4.6](#)

### 1. HTML 1.0

- The basic version of HTML has support for basic elements like text controls and images. This was the very basic version of HTML with less support for a wide range of HTML elements. It does not

have rich features like styling and other things that were related to how content will be rendered in a browser.

- The initial version of HTML does not provide support for tables, font support, etc., as it provides us in the latest version.
- We would also like to discuss that W3C did not exist before HTML 2.0; hence it does not show details about HTML 1.

## 2. HTML 2

- HTML version 2.0 was developed in 1995 with basic intention of improving HTML version 1.0
- Now a standard got started to develop so as to maintain common rules and regulations across different browsers. HTML 2.0 has improved a lot in terms of the markup tags. In HTML 2.0 version concept of form came into force. Forms were developed, but still, they had basic tags like text boxes, buttons, etc.
- Also, the table came as an [HTML tag](#). Now, in HTML tag 2.0, browsers also came with the concept of creating their own layers of tags that were specific to the browser itself. W3C was also formed. The main intention of W3C is to maintain standard across different web browsers so that these browsers understand and render HTML tags in a similar manner.

## 3. HTML 3.2

- It was developed in 1997. After HTML 2.0 was developed, the next version of HTML was 3.2
- With version 3.2 of HTML, HTML tags were further improved. It is worth noting that because of W3C standard maintenance, the newer version of HTML was 3.2 instead of 3.
- Now, HTML 3.2 has better support for new form elements. Another important feature what HTML 3.2 implemented was support for CSS. CSS stands for [Cascading Style Sheet](#). It is CSS that provides features to make HTML tags look better on rendering it on browsers. CSS helps to style HTML elements.
- With the upgradation of browsers to HTML 3.2, the browser also supported for [frame tags](#), although HTML specifications still do not support frame markup tags.

## 4. HTML 4.01

- It was developed in 1999. It extended the support of cascading styling sheets. In version 3.2, CSS were embedded in HTML page itself. Therefore, if the website has various web pages to apply to the style of each page, we must place CSS on each web page. Hence there was a repetition of the same block of CSS.
- To overcome this thing, in version 4.01 concept of an external styling sheet emerged. Under this concept, an external CSS file could be developed, and this external styling file could be included in HTML itself. HTML 4.01 provided support for further new tags of HTML.

## 5. HTML5

- This is the latest version of HTML. For a developer, it could be used in 2014. It came up with lots of HTML tags support. [HTML5](#) provided support for new form elements like input element s of different types; geolocations [support tags](#), etc.

### Let us look at a few of the tags which were added to HTML5

- **Email** – New HTML5 tag, which was added, is the input element of type email. This is a form tag, although it could be used outside of a form tag also. This tag checks the validation of the input value. It checks whether the value inserted is a valid email.



- **Password** – This is another form tag that was added to receive a password from the user. Being the password type field, the user types in the field are not visible directly to the user but are represented by special symbols. These symbols save the password from getting revealed on the browser.
- **Audio tag** – This is a new audio tag that was implemented in [HTML5](#). This tag helps to add audio to our web page. We can use this tag to embed an audio clip into a web page. This audio tag could be played on a webpage.
- **Semantic tags** – Semantic tags are also known as structural tags. Structural tags are the tags that provide structure to the HTML page. It helps it divide the HTML page into different structures. These structures get combined into an HTML page itself to form an HTML web page. Few of the important HTML semantic tags are figcaption, <header>, <footer>
- **Section tag** – This tag is used to semantic a section in an HTML page. A [section tag](#) represents a section on a web page.

## 6. W3C HTML Validator

An HTML validator is a web-based tool that is used to maintain or check whether a piece of HTML tag or HTML is valid. An HTML validator follows the standard of W3C to validate an HTML page. It follows the W3C standard.

There are lots of version of HTML which is being developed. From an initial version of 1.0 to the latest version of 5.2, HTML has developed a lot. W3C has also maintained standards so that all browsers could have a common standard to follow. HTML5 has developed a lot with new tags and the support of form elements.

### CSS – Cascading Style Sheets:

CSS stands for Cascading Style Sheets. It is a style sheet language which is used to describe the look and formatting of a document written in markup language. It provides an additional feature to HTML. It is generally used with HTML to change the style of web pages and user interfaces. It can also be used with any kind of XML documents including plain XML, SVG and XUL.

CSS is used along with HTML and JavaScript in most websites to create user interfaces for web applications and user interfaces for many mobile applications.

What does CSS do

- You can add new looks to your old HTML documents.
- You can completely change the look of your website with only a few changes in CSS code.

### Uses of CSS:

These are the three major benefits of CSS:

#### 1) Solves a big problem

Before CSS, tags like font, color, background style, element alignments, border and size had to be repeated on every web page. This was a very long process. For example: If you are developing a large website where fonts and color information are added on every single page, it will become a long and expensive process. CSS was created to solve this problem. It was a W3C recommendation.

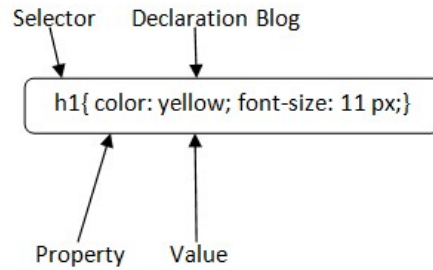
#### 2) Saves a lot of time

CSS style definitions are saved in external CSS files so it is possible to change the entire website by changing just one file.

### 3) Provide more attributes

CSS provides more detailed attributes than plain HTML to define the look and feel of the website.

A CSS rule set contains a selector and a declaration block.



**Selector:** Selector indicates the HTML element you want to style. It could be any tag like `<h1>`, `<title>` etc.

**Declaration Block:** The declaration block can contain one or more declarations separated by a semicolon. For the above example, there are two declarations:

1. `color: yellow;`
2. `font-size: 11 px;`

Each declaration contains a property name and value, separated by a colon.

**Property:** A Property is a type of attribute of HTML element. It could be color, border etc.

**Value:** Values are assigned to CSS properties. In the above example, value "yellow" is assigned to color property.

1. Selector
2. {Property1: value1; Property2: value2; .....;}

### Dynamic Web page

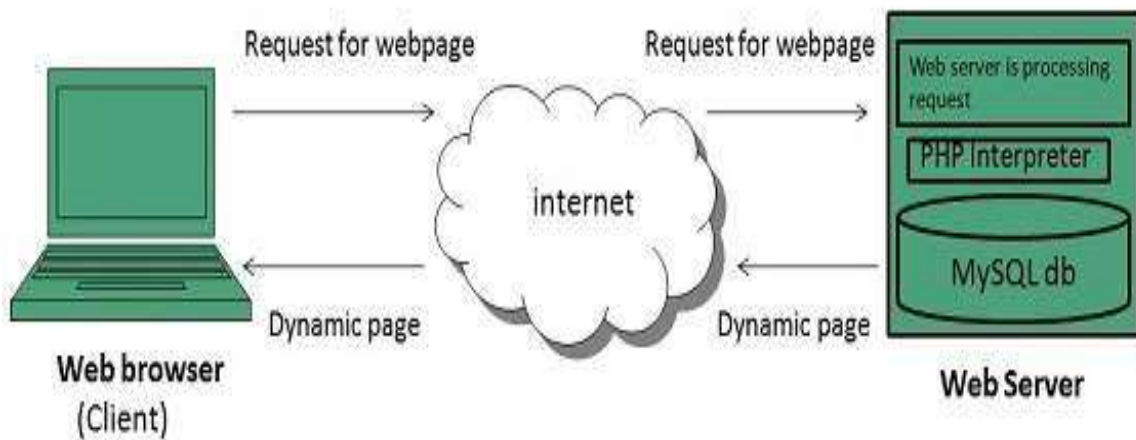
Dynamic web page shows different information at different point of time. It is possible to change a portion of a web page without loading the entire web page. It has been made possible using Ajax technology.

#### Server-side dynamic web page

It is created by using server-side scripting. There are server-side scripting parameters that determine how to assemble a new web page which also include setting up of more client-side processing.

#### Client-side dynamic web page

It is processed using client side scripting such as JavaScript. And then passed in to Document Object Model (DOM).



### Scripting Languages

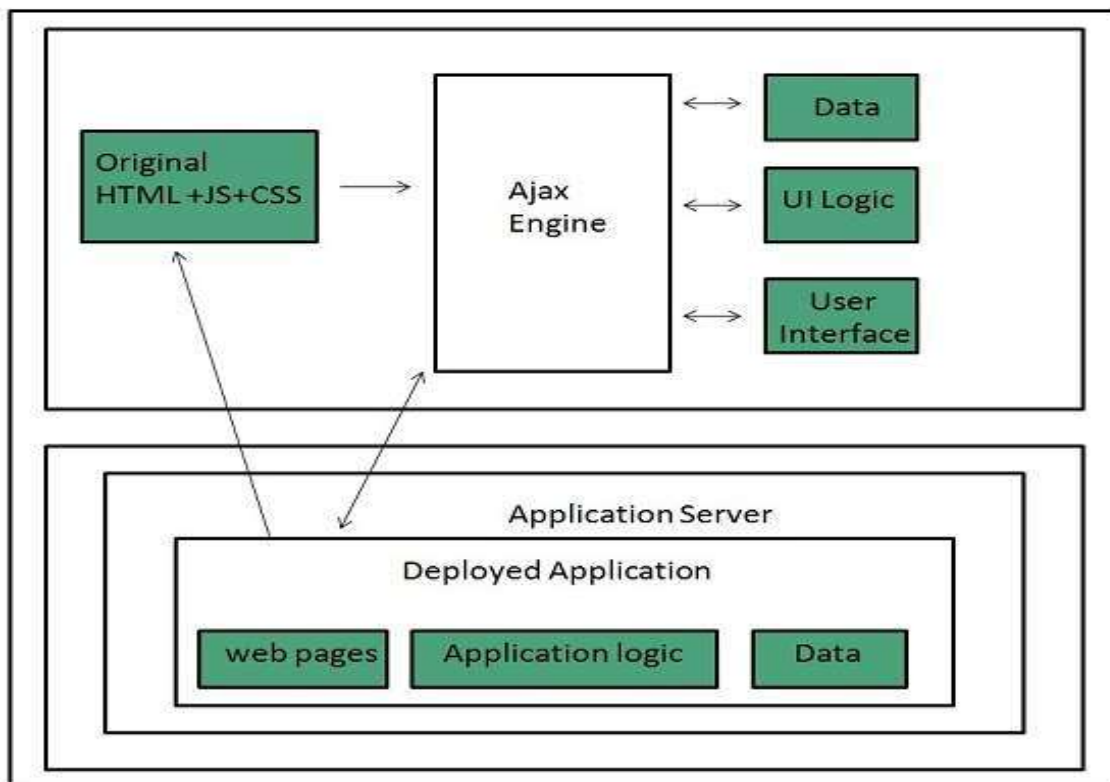
Scripting languages are like programming languages that allow us to write programs in form of script. These scripts are interpreted not compiled and executed line by line.

Scripting language is used to create dynamic web pages.

### Client-side Scripting

**Client-side scripting** refers to the programs that are executed on client-side. Client-side scripts contains the instruction for the browser to be executed in response to certain user's action.

Client-side scripting programs can be embedded into HTML files or also can be kept as separate files.

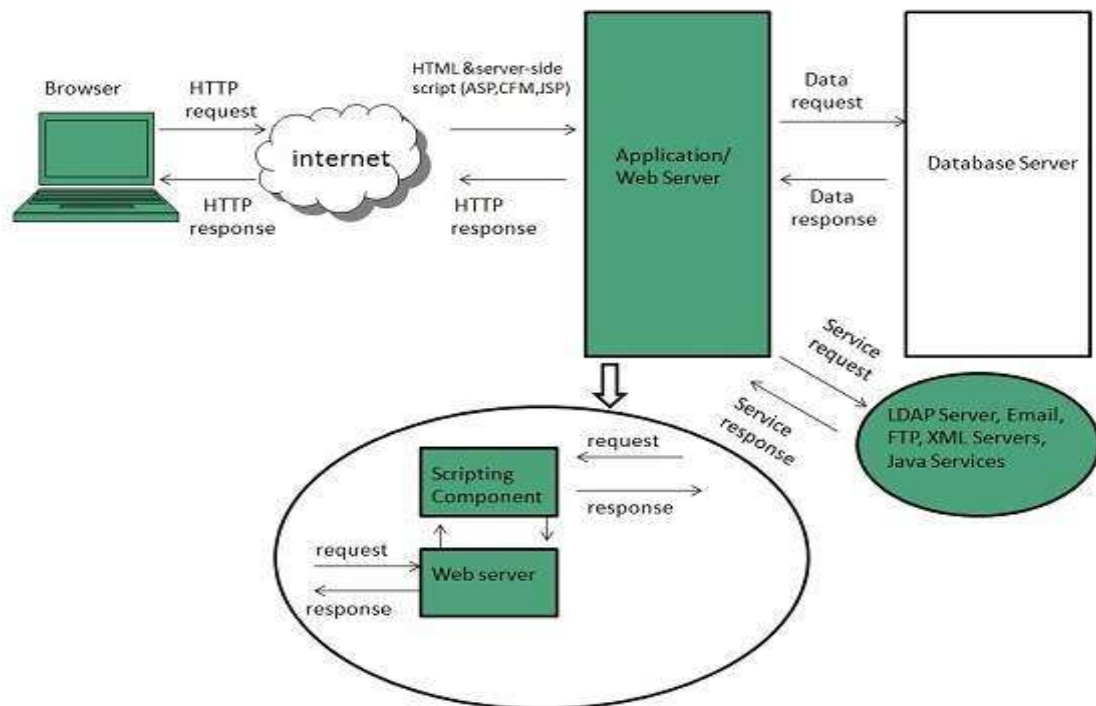


Following table describes commonly used Client-Side scripting languages:

S.NO	Scripting Language Description
1.	<b>JavaScript</b> It is a prototype based scripting language. It inherits its naming conventions from java. All java script files are stored in file having <b>.js</b> extension.
2.	<b>ActionScript</b> It is an object oriented programming language used for the development of websites and software targeting Adobe flash player.
3.	<b>Dart</b> It is an open source web programming language developed by Google. It relies on source-to-source compiler to JavaScript.
4.	<b>VBScript</b> It is an open source web programming language developed by Microsoft. It is superset of JavaScript and adds optional static typing class-based object oriented programming.

### Server-side Scripting

**Sever-side scripting** acts as an interface for the client and also limit the user access the resources on web server. It can also collects the user's characteristics in order to customize response.



Following table describes commonly used Server-Side scripting languages:

S.N.	Scripting Language Description
1.	<b>ASP</b> Active Server Pages (ASP) is server-side script engine to create dynamic web pages. It supports <b>Component Object Model (COM)</b> which enables ASP web sites to access functionality of libraries such as DLL.
2.	<b>ActiveVFP</b> It is similar to PHP and also used for creating dynamic web pages. It uses native <b>Visual Foxpro</b> language and database.
3.	<b>ASP.net</b> It is used to develop dynamic websites, web applications, and web services.
4.	<b>Java</b> Java Server Pages are used for creating dynamic web applications. The Java code is compiled into byte code and run by <b>Java Virtual Machine (JVM)</b> .
5.	<b>Python</b> It supports multiple programming paradigms such as object-oriented, and functional programming. It can also be used as non-scripting language using third party tools such as <b>Py2exe</b> or <b>Pyinstaller</b> .
6.	<b>WebDNA</b> It is also a server-side scripting language with an embedded database system.

#### AJAX – Asynchronous JavaScript and XML:

AJAX tutorial covers concepts and examples of AJAX technology for beginners and professionals. AJAX is an acronym for **Asynchronous JavaScript and XML**. It is a group of inter-related technologies like JavaScript, DOM, XML, HTML/XHTML, CSS, XMLHttpRequest etc. AJAX allows you to send and receive data asynchronously without reloading the web page. So it is fast.

AJAX allows you to send only important information to the server not the entire page. So only valuable data from the client side is routed to the server side. It makes your application interactive and faster.

#### Uses:

There are too many web applications running on the web that are using ajax technology like **gmail, facebook, twitter, google map, youtube** etc.,

#### HTTP:

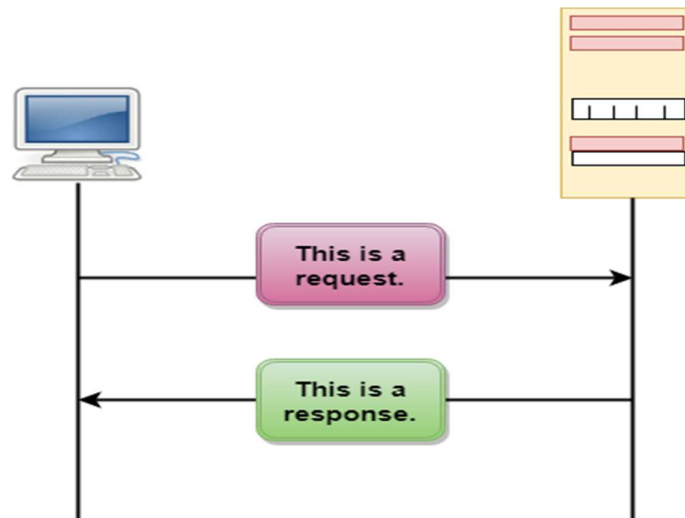
- HTTP stands for **HyperText Transfer Protocol**.
- It is a protocol used to access the data on the World Wide Web (www).
- The HTTP protocol can be used to transfer the data in the form of plain text, hypertext, audio, video, and so on.
- This protocol is known as HyperText Transfer Protocol because of its efficiency that allows us to use in a hypertext environment where there are rapid jumps from one document to another document.

- HTTP is similar to the FTP as it also transfers the files from one host to another host. But, HTTP is simpler than FTP as HTTP uses only one connection, i.e., no control connection to transfer the files.
- HTTP is used to carry the data in the form of MIME-like format.
- HTTP is similar to SMTP as the data is transferred between client and server. The HTTP differs from the SMTP in the way the messages are sent from the client to the server and from server to the client. SMTP messages are stored and forwarded while HTTP messages are delivered immediately.

#### Features of HTTP:

- **Connectionless protocol:** HTTP is a connectionless protocol. HTTP client initiates a request and waits for a response from the server. When the server receives the request, the server processes the request and sends back the response to the HTTP client after which the client disconnects the connection. The connection between client and server exist only during the current request and response time only.
- **Media independent:** HTTP protocol is a media independent as data can be sent as long as both the client and server know how to handle the data content. It is required for both the client and server to specify the content type in MIME-type header.
- **Stateless:** HTTP is a stateless protocol as both the client and server know each other only during the current request. Due to this nature of the protocol, both the client and server do not retain the information between various requests of the web pages.

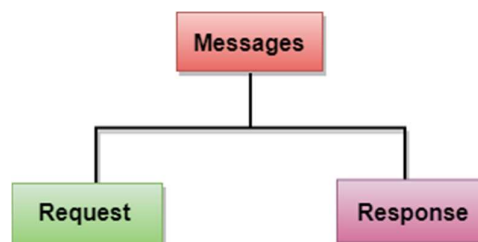
#### HTTP Transactions



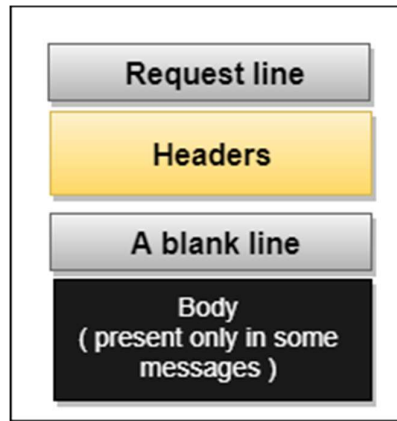
The above figure shows the HTTP transaction between client and server. The client initiates a transaction by sending a request message to the server. The server replies to the request message by sending a response message.

#### Messages

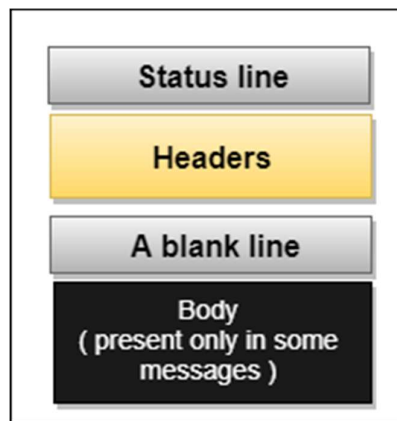
HTTP messages are of two types: request and response. Both the message types follow the same message format.



**Request Message:** The request message is sent by the client that consists of a request line, headers, and sometimes a body.



**Response Message:** The response message is sent by the server to the client that consists of a status line, headers, and sometimes a body.



### Uniform Resource Locator (URL)

- A client that wants to access the document in an internet needs an address and to facilitate the access of documents, the HTTP uses the concept of Uniform Resource Locator (URL).
- The Uniform Resource Locator (URL) is a standard way of specifying any kind of information on the internet.
- The URL defines four parts: method, host computer, port, and path.



- **Method:** The method is the protocol used to retrieve the document from a server. For example, HTTP.
- **Host:** The host is the computer where the information is stored, and the computer is given an alias name. Web pages are mainly stored in the computers and the computers are given an alias name that begins with the characters "www". This field is not mandatory.

- **Port:** The URL can also contain the port number of the server, but it's an optional field. If the port number is included, then it must come between the host and path and it should be separated from the host by a colon.
- **Path:** Path is the pathname of the file where the information is stored. The path itself contains slashes that separate the directories from the subdirectories and files.

### **The Mobile Web:**

The mobile web refers to the use of the internet through handheld mobile devices. Increasingly, smartphones and other devices with wireless data access structures access the same “full” internet traditionally accessed on desktop or laptop computers.

The mobile web most often refers to access via a conventional mobile browser, although the line blurs when it comes to apps. Clearly, these still access the Internet wirelessly, but some differentiate from a browser-based site, as compared to an app specific to one property.

Mobile web access comes with some unique challenges. One is the idea of standardization. The Mobile Web Initiative from the World Wide Web Consortium (W3C) aims to provide standards for mobile web access.

The smaller display screen is another major development hurdle. Many website designers have had trouble adapting web pages to look good on both computers and handheld devices. It can be a tough decision – whether to build one site that accommodates all screen sizes, as opposed to one site for desktops and another for mobile. As HTML5 is more commonly used, the hope is that accomplishing these tasks will require less coding.

Finally, speed is a major issue. A Wi-Fi connection is generally good enough for any web application. Of course, Wi-Fi is not truly mobile, versus a wireless carrier’s 3G or 4G network. Given that many parts of the world do not have strong 3G/4G access, latency is a major mobile development concern.

### **Web Search:**

#### **Web Search Engines**

Web search engines use special software programs (called robots, spiders, or crawlers) to find Web pages and list (or index) all words within each one to make searching large quantities of pages faster. Indexes capture the largest amount of information on the Web, but no index lists everything on the Internet.

Commonly used search engines include Google (<https://www.google.com>) and Bing (<http://www.bing.com>).

In addition to search engines, there are also:

- Specialized web search engines – A tool that has a specialty, usually either a subject or format focus. It ignores the rest of the information on the web. Examples include science.gov (<http://www.science.gov/>) and TinEye Reverse Image Search (<https://www.tineye.com>).
- Metasearch engines – Tools that search multiple web search engines and gives you results from all of them. Some of these return the best results from the search engines they search. Examples include Dogpile (<http://www.dogpile.com>) and WebCrawler (<https://www.webcrawler.com>).
- Web directories – Tools created by editors or trained researchers who categorize or classify web sites by subject. Directories are more selective than search engines. An example is the Directory of Open Access Journals (<https://doaj.org/>).

### **When to Use Them**

Web Search Engines and related web search tools are helpful for locating background information, news (especially if it’s recent), and public opinion. However, scholarly information is often not available through a regular web search. If you do find scholarly information through a web search engine, especially if



you are off campus, you may be asked for payment to access it. Ohio State Libraries can usually get you what you need without additional payment.

### How to Use Them

Use of each tool varies. If a search engine has an advanced search, it may include options such as specifying format, language, domain, or date range.

## **NETWORK SECURITY:**

### Cryptography:

Cryptography refers to the science and art of transforming messages to make them secure and immune to attacks. It is a method of storing and transmitting data in a particular form so that only those for whom it is intended can read and process it. Cryptography not only protects data from theft or alteration but can also be used for user authentication.

### Components

There are various components of cryptography which are as follows –

#### Plaintext and Ciphertext

The original message, before being transformed, is called plaintext. After the message is transformed, it is called ciphertext. An encryption algorithm transforms the plaintext into ciphertext; a decryption algorithm transforms the ciphertext back into plaintext. The sender uses an encryption algorithm, and the receiver uses a decryption algorithm.

#### Cipher

Encryption and decryption algorithms are referred as ciphers. The term cipher is also used to refer to different categories of algorithms in cryptography. This is not to say that every sender-receiver pair needs their very own unique cipher for secure communication. On the contrary, one cipher can serve millions of communicating pairs.

#### Key

A key is a number (or a set of numbers) that the cipher, as an algorithm, operates on. To encrypt a message, we need an encryption algorithm, an encryption key, and plaintext. These create the ciphertext. To decrypt a message, we need a decryption algorithm, a decryption key, and the ciphertext. These reveal the original plaintext.

#### Types

There are two types of cryptography which are as follows –

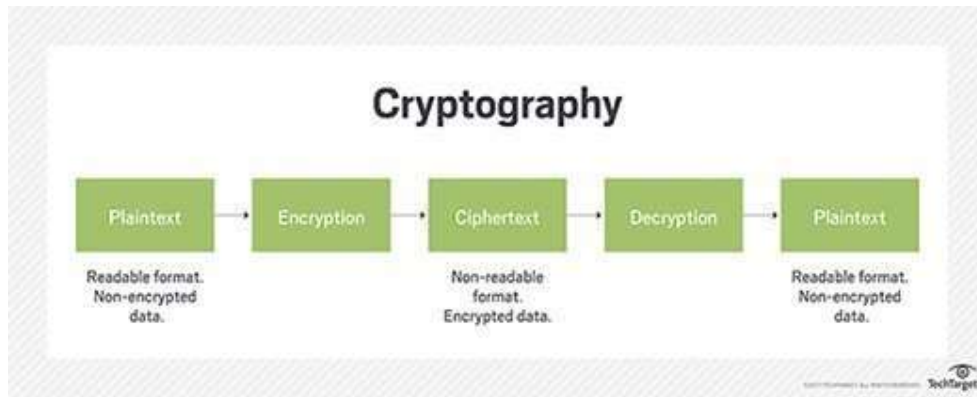
#### Symmetric Key Cryptography

In symmetric-key cryptography, the same key is used by both parties. The sender uses this key and an encryption algorithm to encrypt data; the receiver uses the same key and the corresponding decryption algorithm to decrypt the data.

#### Asymmetric-Key Cryptography

In asymmetric or public-key cryptography, there are two keys: a private key and a public key. The private key is kept by the receiver. The public key is announced to the public.

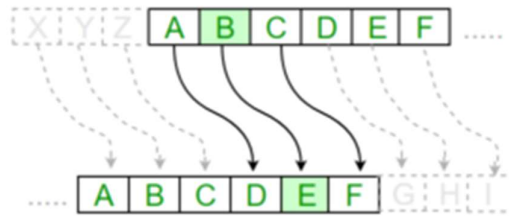
In public-key encryption/decryption, the public key that is used for encryption is different from the private key that is used for decryption. The public key is available to the public, and the private key is available only to an individual.



## Substitution Cipher

Hiding some data is known as encryption. When plain text is encrypted it becomes unreadable and is known as ciphertext. In a Substitution cipher, any character of plain text from the given fixed set of characters is substituted by some other character from the same set depending on a key. For example with a shift of 1, A would be replaced by B, B would become C, and so on.

**Note:** Special case of Substitution cipher is known as [Caesar cipher](#) where the key is taken as 3.



### Examples:

**Plain Text:** I am studying Data Encryption

**Key:** 4

**Output:** M eq wxyhCmrk Hexe IrgvCtxmsr

**Plain Text:** ABCDEFGHIJKLMNOPQRSTUVWXYZ

**Key:** 4

**Output:** EFGHIJKLMNOPQRSTUVWXYZabcd

### Algorithm for Substitution Cipher:

#### Input:

- A String of both lower and upper case letters, called PlainText.
- An Integer denoting the required key.

#### Procedure:

- Create a list of all the characters.
- Create a dictionary to store the substitution for all characters.
- For each character, transform the given character as per the rule, depending on whether we're encrypting or decrypting the text.
- Print the new string generated.

### Transposition Cipher:

Transposition Cipher is a cryptographic algorithm where the order of alphabets in the plaintext is rearranged to form a cipher text. In this process, the actual plain text alphabets are not included.

#### Example

A simple example for a transposition cipher is **columnar transposition cipher** where each character in the plain text is written horizontally with specified alphabet width. The cipher is written vertically, which creates an entirely different cipher text.

Consider the plain text **hello world**, and let us apply the simple columnar transposition technique as shown below

h	e	l	l
o	w	o	r
l	d		

The plain text characters are placed horizontally and the cipher text is created with vertical format as : **holewdlo lr**. Now, the receiver has to use the same table to decrypt the cipher text to plain text.

**Note** – Cryptanalysts observed a significant improvement in crypto security when transposition technique is performed. They also noted that re-encrypting the cipher text using same transposition cipher creates better security.

### One Time Pad Cipher:

One-time pad cipher is a type of Vignere cipher which includes the following features –

- It is an unbreakable cipher.
- The key is exactly same as the length of message which is encrypted.
- The key is made up of random symbols.
- As the name suggests, key is used one time only and never used again for any other message to be encrypted.

Due to this, encrypted message will be vulnerable to attack for a cryptanalyst. The key used for a one-time pad cipher is called **pad**, as it is printed on pads of paper.

Why is it Unbreakable?

The key is unbreakable owing to the following features –

- The key is as long as the given message.
- The key is truly random and specially auto-generated.
- Key and plain text calculated as modulo 10/26/2.
- Each key should be used once and destroyed by both sender and receiver.
- There should be two copies of key: one with the sender and other with the receiver.

#### Encryption

To encrypt a letter, a user needs to write a key underneath the plaintext. The plaintext letter is placed on the top and the key letter on the left. The cross section achieved between two letters is the plain text. It is described in the example below –

Plain text: THIS IS SECRET
OTP-Key : XVHE UW NOPGDZ
-----
Ciphertext: QCPW CO FSRXHS
In groups : QCPWC OFSRX HS

## Decryption

To decrypt a letter, user takes the key letter on the left and finds cipher text letter in that row. The plain text letter is placed at the top of the column where the user can find the cipher text letter.

Two Fundamental Principles of Cryptography:

### 1. Redundancy

- Some redundancy must be there in all the encrypted messages.
- By redundancy here, we mean the information that is not required for understanding the message reducing the chances for a passive intruder to make attacks.
- **Passive intruder** attacks involve putting the stolen information to misuse without understanding it.
- This can be more easily understood by an example of a credit card.
- The credit card number is not alone sent over the internet rather it is accompanied by other side info such as the DOB of the card holder, its validity date and so on.
- Including such info with the card number cuts down on the chances for making up the number.
- Adding a good amount of redundancy prevents the active intruders from sending garbage values and then getting it verified as some valid message.
- The recipient must be capable of determining whether the message is valid or not by doing some inspection and simple calculation.
- **Without redundancy the attackers would simply send junk message and the recipient will decode it as a valid message.**
- However, there is a little concern also with this.
- **N number of zeroes** must not be put at the beginning or the end of the message for redundancy because such messages become easy to be predicted thus facilitating the crypt analysts work.
- Instead of zeroes, a CRC polynomial can be used because it proves to be more work.
- Using cryptographic hash might be even better.
- **Redundancy has also got a role to play in quantum cryptography.**
- Some redundancy is required in the messages for the bob to determine if the message has been tampered.
- Repetition of the message twice is a crude form of redundancy.
- If the two copies are found to be identical, the bob states that somebody is interfering with the transmission or there is a lot of noise.
- But such repetition process to be expensive.
- Therefore, for error detection and correction the methods used are reed Solomon and hamming codes.

## 2. Update

- Measures must be compulsorily taken for the prevention of the attacks by active intruders who might play back the old messages.
- **The longer an encrypted message is held by an active intruder, the more is the possibility that he can break in to it.**
- One good example of this is the UNIX password file.
- For anybody who has an account on the host, the password is accessible.
- A copy of this file can be obtained by the intruders and they can then easily de-crypt the password.
- Also, the addition of the redundancy allows the simplification of the messages' decryption.
- It must be checked whether the message has been sent recently or is an old one.
- One measure for doing so is including a time stamp of few seconds in the message.
- This message then can be saved by the recipient for that many seconds and can be used for comparing with the incoming messages and filtering the duplicates.
- Messages which exceed this time period will be rejected as being too old.

### Apart from the above two principles the following are some other principles of cryptography:

**Ø Authentication:** For ensuring that the message was generated by the sender itself and no one else so that no outsider can claim as being the owner of the message.

**Ø Integrity:** In cryptography, the integrity of the messages must be preserved while sending the message from one host to another. This involves ensuring that the message is not altered on the way. Using cryptographic hash is a way to achieve this.

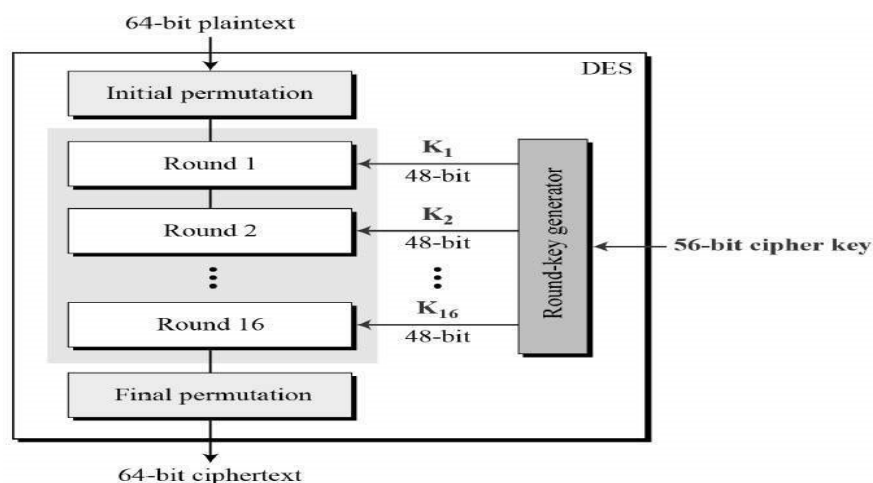
### **Ø Non-repudiation**

#### Symmetric Key Algorithms:

#### Data Encryption Standard:

The Data Encryption Standard (DES) is a symmetric-key block cipher published by the National Institute of Standards and Technology (NIST).

DES is an implementation of a Feistel Cipher. It uses 16 round Feistel structure. The block size is 64-bit. Though, key length is 64-bit, DES has an effective key length of 56 bits, since 8 of the 64 bits of the key are not used by the encryption algorithm (function as check bits only). General Structure of DES is depicted in the following illustration –

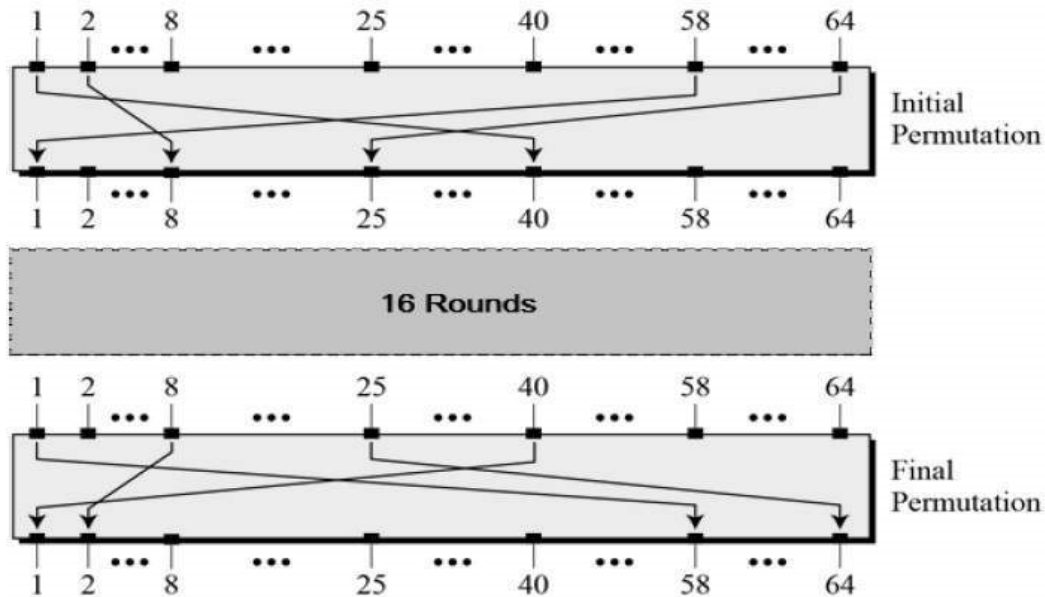


Since DES is based on the Feistel Cipher, all that is required to specify DES is –

- Round function
- Key schedule
- Any additional processing – Initial and final permutation

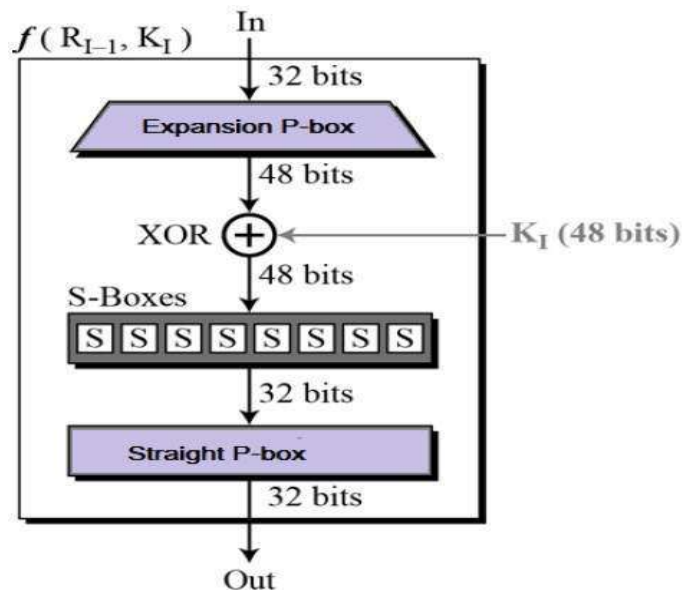
#### Initial and Final Permutation

The initial and final permutations are straight Permutation boxes (P-boxes) that are inverses of each other. They have no cryptography significance in DES. The initial and final permutations are shown as follows –

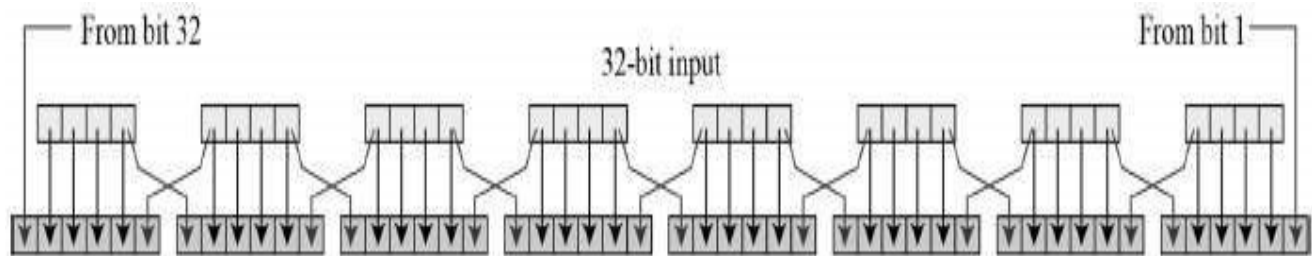


#### Round Function

The heart of this cipher is the DES function,  $f$ . The DES function applies a 48-bit key to the rightmost 32 bits to produce a 32-bit output.



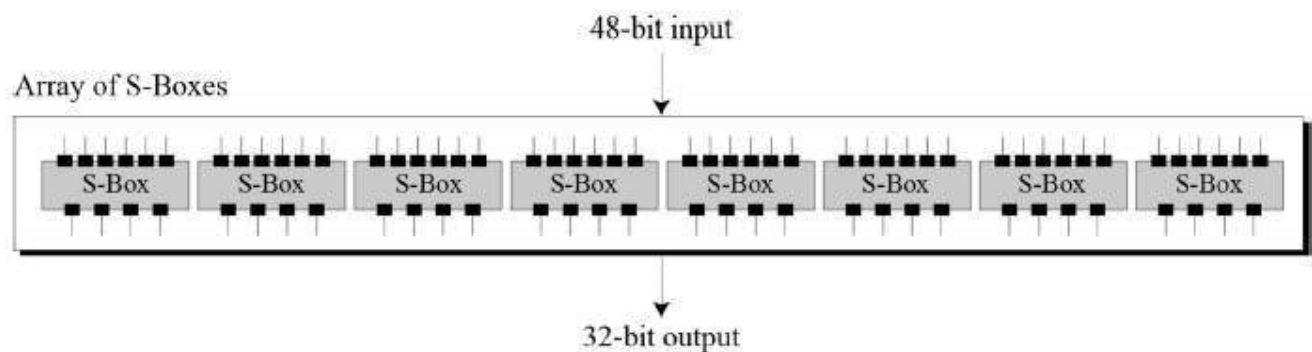
- **Expansion Permutation Box** – Since right input is 32-bit and round key is a 48-bit, we first need to expand right input to 48 bits. Permutation logic is graphically depicted in the following illustration –



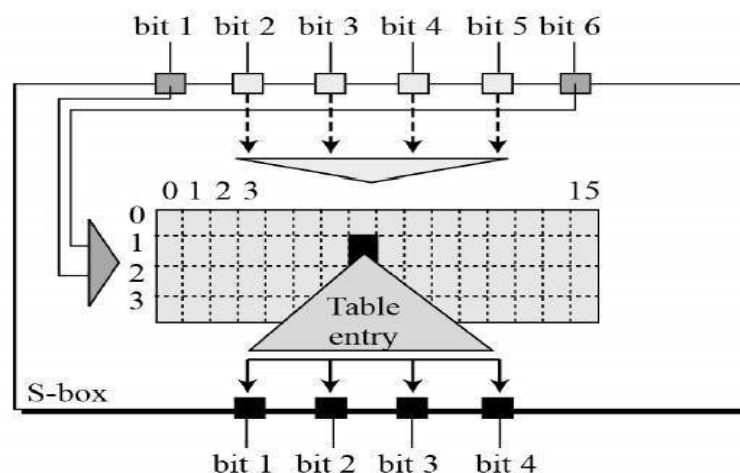
- The graphically depicted permutation logic is generally described as table in DES specification illustrated as shown –

32	01	02	03	04	05
04	05	06	07	08	09
08	09	10	11	12	13
12	13	14	15	16	17
16	17	18	19	20	21
20	21	22	23	24	25
24	25	26	27	28	29
28	29	31	31	32	01

- XOR (Whitener).** – After the expansion permutation, DES does XOR operation on the expanded right section and the round key. The round key is used only in this operation.
- Substitution Boxes.** – The S-boxes carry out the real mixing (confusion). DES uses 8 S-boxes, each with a 6-bit input and a 4-bit output. Refer the following illustration –



- The S-box rule is illustrated below –



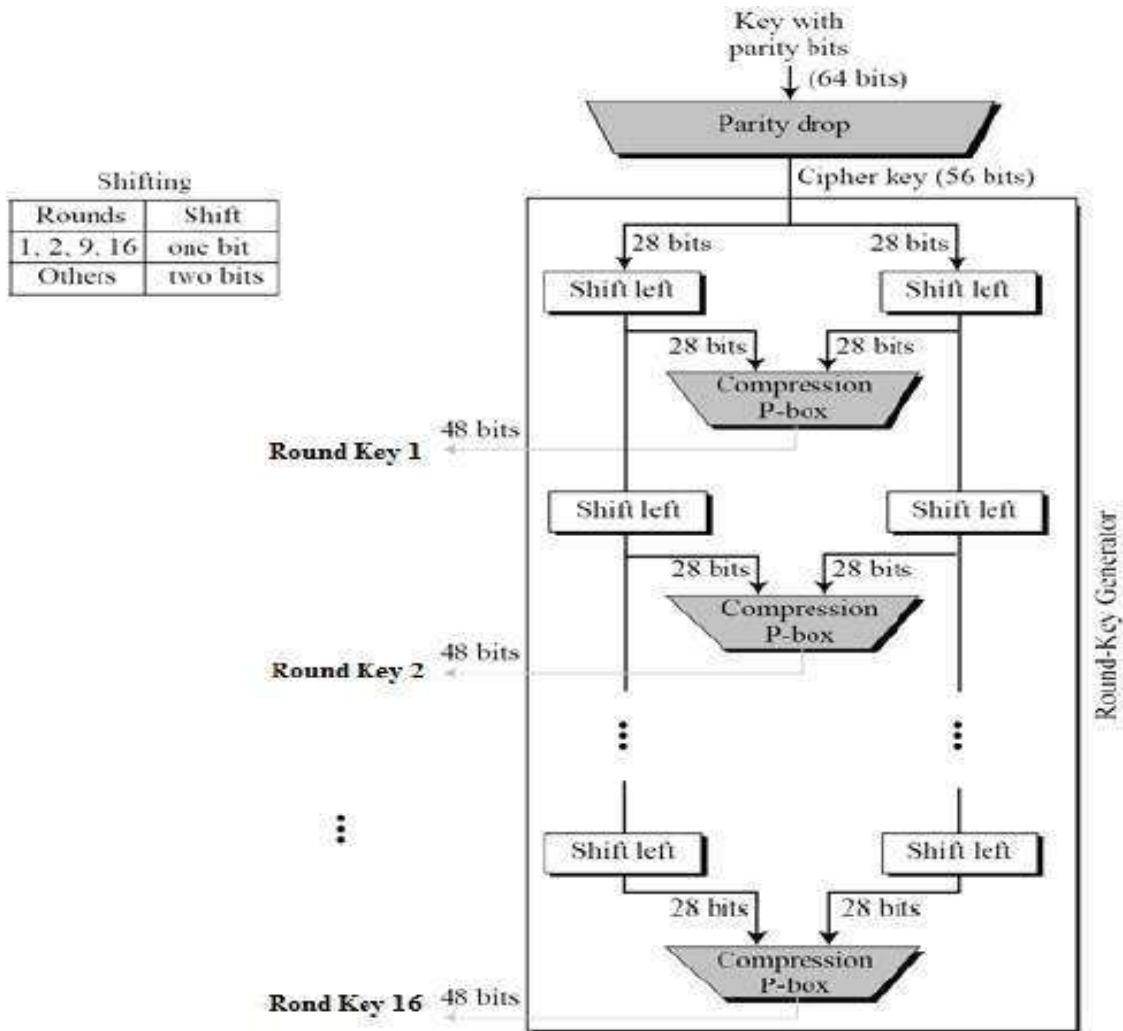
- There are a total of eight S-box tables. The output of all eight s-boxes is then combined in to 32 bit section.

- **Straight Permutation** – The 32 bit output of S-boxes is then subjected to the straight permutation with rule shown in the following illustration:

16	07	20	21	29	12	28	17
01	15	23	26	05	18	31	10
02	08	24	14	32	27	03	09
19	13	30	06	22	11	04	25

### Key Generation

The round-key generator creates sixteen 48-bit keys out of a 56-bit cipher key. The process of key generation is depicted in the following illustration –



The logic for Parity drop, shifting, and Compression P-box is given in the DES description.

### DES Analysis

The DES satisfies both the desired properties of block cipher. These two properties make cipher very strong.

- **Avalanche effect** – A small change in plaintext results in the very great change in the ciphertext.
- **Completeness** – Each bit of ciphertext depends on many bits of plaintext.

During the last few years, cryptanalysis have found some weaknesses in DES when key selected are weak keys. These keys shall be avoided.



DES has proved to be a very well designed block cipher. There have been no significant cryptanalytic attacks on DES other than exhaustive key search.

### Triple DES:

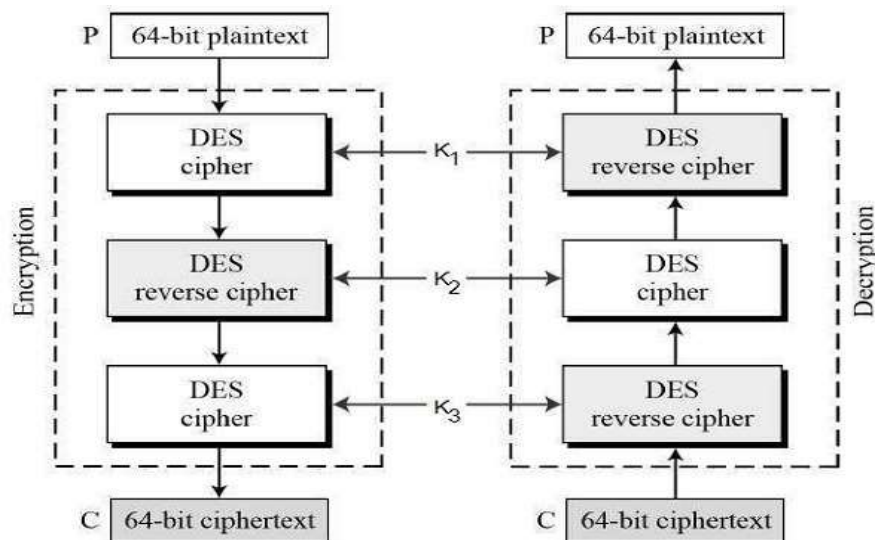
The speed of exhaustive key searches against DES after 1990 began to cause discomfort amongst users of DES. However, users did not want to replace DES as it takes an enormous amount of time and money to change encryption algorithms that are widely adopted and embedded in large security architectures.

The pragmatic approach was not to abandon the DES completely, but to change the manner in which DES is used. This led to the modified schemes of Triple DES (sometimes known as 3DES).

Incidentally, there are two variants of Triple DES known as 3-key Triple DES (3TDES) and 2-key Triple DES (2TDES).

### 3-KEY Triple DES

Before using 3TDES, user first generate and distribute a 3TDES key  $K$ , which consists of three different DES keys  $K_1$ ,  $K_2$  and  $K_3$ . This means that the actual 3TDES key has length  $3 \times 56 = 168$  bits. The encryption scheme is illustrated as follows –



The encryption-decryption process is as follows –

- Encrypt the plaintext blocks using single DES with key  $K_1$ .
- Now decrypt the output of step 1 using single DES with key  $K_2$ .
- Finally, encrypt the output of step 2 using single DES with key  $K_3$ .
- The output of step 3 is the ciphertext.
- Decryption of a ciphertext is a reverse process. User first decrypt using  $K_3$ , then encrypt with  $K_2$ , and finally decrypt with  $K_1$ .

Due to this design of Triple DES as an encrypt–decrypt–encrypt process, it is possible to use a 3TDES (hardware) implementation for single DES by setting  $K_1$ ,  $K_2$ , and  $K_3$  to be the same value. This provides backwards compatibility with DES.

Second variant of Triple DES (2TDES) is identical to 3TDES except that  $K_3$  is replaced by  $K_1$ . In other words, user encrypt plaintext blocks with key  $K_1$ , then decrypt with key  $K_2$ , and finally encrypt with  $K_1$  again. Therefore, 2TDES has a key length of 112 bits.

Triple DES systems are significantly more secure than single DES, but these are clearly a much slower process than encryption using single DES

## Advanced Encryption Standard:

The AES Encryption algorithm (also known as the Rijndael algorithm) is a symmetric block cipher algorithm with a block/chunk size of 128 bits. It converts these individual blocks using keys of 128, 192, and 256 bits. Once it encrypts these blocks, it joins them together to form the ciphertext.

It is based on a substitution-permutation network, also known as an SP network. It consists of a series of linked operations, including replacing inputs with specific outputs (substitutions) and others involving bit shuffling (permutations).

In this tutorial, you will go through some of the standout features that AES offers as a globally standardized encryption algorithm.

What are the Features of AES?

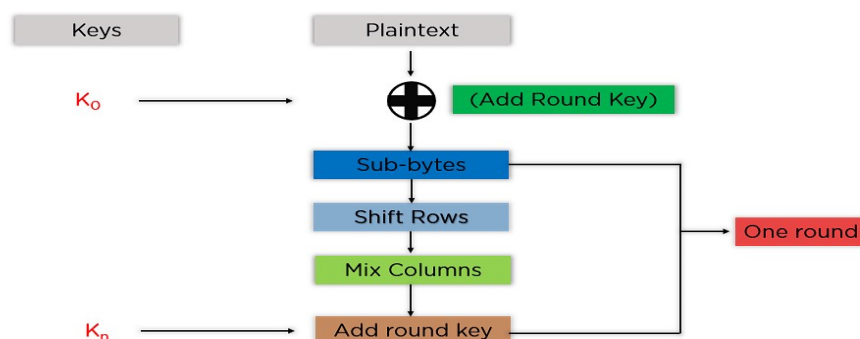
1. **SP Network:** It works on an SP network structure rather than a Feistel cipher structure, as seen in the case of the DES algorithm.
2. **Key Expansion:** It takes a single key up during the first stage, which is later expanded to multiple keys used in individual rounds.
3. **Byte Data:** The AES encryption algorithm does operations on byte data instead of bit data. So it treats the 128-bit block size as 16 bytes during the encryption procedure.
4. **Key Length:** The number of rounds to be carried out depends on the length of the key being used to encrypt data. The 128-bit key size has ten rounds, the 192-bit key size has 12 rounds, and the 256-bit key size has 14 rounds.

How Does AES Work?

To understand the way AES works, you first need to learn how it transmits information between multiple steps. Since a single block is 16 bytes, a 4x4 matrix holds the data in a single block, with each cell holding a single byte of information.

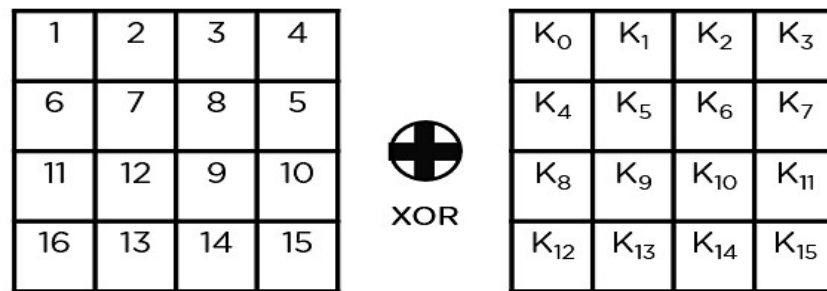
0	1	2	3
4	5	6	7
8	9	10	11
12	13	14	15

The matrix shown in the image above is known as a state array. Similarly, the key being used initially is expanded into  $(n+1)$  keys, with  $n$  being the number of rounds to be followed in the encryption process. So for a 128-bit key, the number of rounds is 16, with no. of keys to be generated being  $10+1$ , which is a total of 11 keys. Steps to be followed in AES

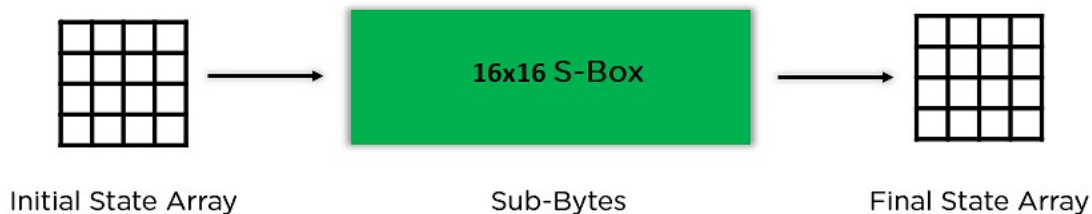


The mentioned steps are to be followed for every block sequentially. Upon successfully encrypting the individual blocks, it joins them together to form the final ciphertext. The steps are as follows:

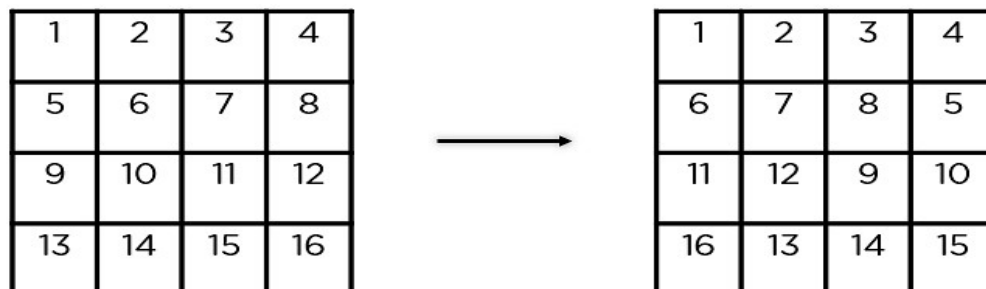
- **Add Round Key:** You pass the block data stored in the state array through an XOR function with the first key generated ( $K_0$ ). It passes the resultant state array on as input to the next step.



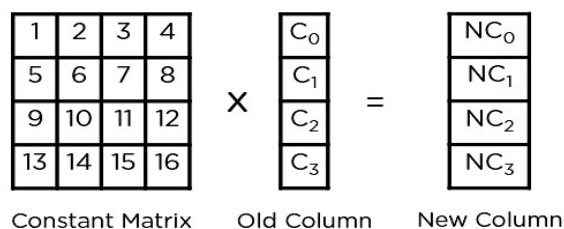
- **Sub-Bytes:** In this step, it converts each byte of the state array into hexadecimal, divided into two equal parts. These parts are the rows and columns, mapped with a substitution box (S-Box) to generate new values for the final state array.



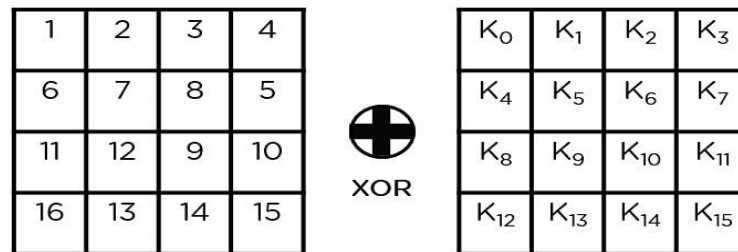
**Shift Rows:** It swaps the row elements among each other. It skips the first row. It shifts the elements in the second row, one position to the left. It also shifts the elements from the third row two consecutive positions to the left, and it shifts the last row three positions to the left.



- **Mix Columns:** It multiplies a constant matrix with each column in the state array to get a new column for the subsequent state array. Once all the columns are multiplied with the same constant matrix, you get your state array for the next step. This particular step is not to be done in the last round.



- **Add Round Key:** The respective key for the round is XOR'd with the state array is obtained in the previous step. If this is the last round, the resultant state array becomes the ciphertext for the specific block; else, it passes as the new state array input for the next round.



### Rijndael:

Rijndael (pronounced rain-dahl) is an Advanced Encryption Standard (AES) algorithm. It replaced the older and weaker Data Encryption Standard (DES) when it was selected as the standard symmetric key encryption algorithm by the National Institute of Standards and Technology (NIST).

Rijndael is an iterated block cipher, meaning that it encrypts and decrypts a block of data by the iteration or round of a specific transformation. It supports encryption key sizes of 128, 192, and 256 bits and handles data in 128-bit blocks.

### **Rijndael as Advanced Encryption Standard**

DES has been in use since 1977. However, by the early 2000s, it began to show security weaknesses. It became obvious that threat actors could use brute-force attacks and crack DES. A better and more secure algorithm was required to encrypt sensitive, unclassified federal information in the U.S. In 2001, NIST encouraged cryptographers to present a more resilient algorithm to replace DES to encrypt mission-critical data.

Five algorithms -- MARS, Rivest Cipher 6, Serpent, Twofish and Rijndael -- were presented.

Ultimately, NIST selected Rijndael as AES. Before confirming its selection, NIST evaluated Rijndael on several factors, including security, cost and implementation. Rijndael was chosen because it offered the best performance, security, efficiency, flexibility and ease of implementation. The NIST selection was formalized when Federal Information Processing Standards 197 was released.

AES is approved in the U.S. for use with government documents that require high-level security clearance.

### **Guiding principles of Rijndael**

Rijndael is named after its two creators: Belgian cryptologists Vincent Rijmen and Joan Daemen. It has its origins in Square, another algorithm designed by the pair. This new algorithm improves upon Square based on three fundamental guiding principles:

1. It can resist all known attacks.
2. It ensures source code compactness and speed on multiple computing platforms.
3. It features a simple design.

## Differences between AES & DES

DES Algorithm	AES Algorithm
Key Length - 56 bits	Key Length - 128, 192, 256 bits
Block Size - 64 bits	Block size - 128 bits
Fixed no. of rounds	No. of rounds dependent on key length
Slower and less secure	Faster and more secure

## Working of Rijndael

In Rijndael, encryption happens through a series of matrix transformations or rounds. The number of rounds are variable, depending on the key or block sizes used:

- 128 bits = 9 rounds
- 192 bits = 11 rounds
- 256 bits = 13 rounds

The Rijndael algorithm is based on byte-by-byte replacement, swap and XOR operations. The procedure is as follows:

- The algorithm generates 10 128-bit keys from the 128-bit key, which are stored in 4x4 tables.
- The plaintext is divided into 4x4 tables, each of 128-bit sizes.
- Each 128-bit plaintext piece goes through a variable number of rounds as mentioned above. The code is generated after the 10th round.

Each round consists of four steps:

1. **Byte Sub.** Each byte of the block is replaced by its substitute in the S-box.
2. **Shift Row.** In a block made of bytes 1 to 16, bytes are arranged in a rectangle and shifted according to block sizes.
3. **Mix Column.** Here, matrix multiplication is performed, where each column is multiplied by the matrix. The bytes being multiplied are treated as polynomials, not as numbers. When results have more than 8 bits, the extra bits are cancelled out by XORing the binary 9-bit string 100011011 with the result. This technique is similar to what is used in in [cyclic redundancy checks](#).
4. **Add Round Key.** Here, the subkey for the current round is XORed.

When Rijndael is performed several times with different round keys, its security increases significantly.

## Advantages and applications of Rijndael

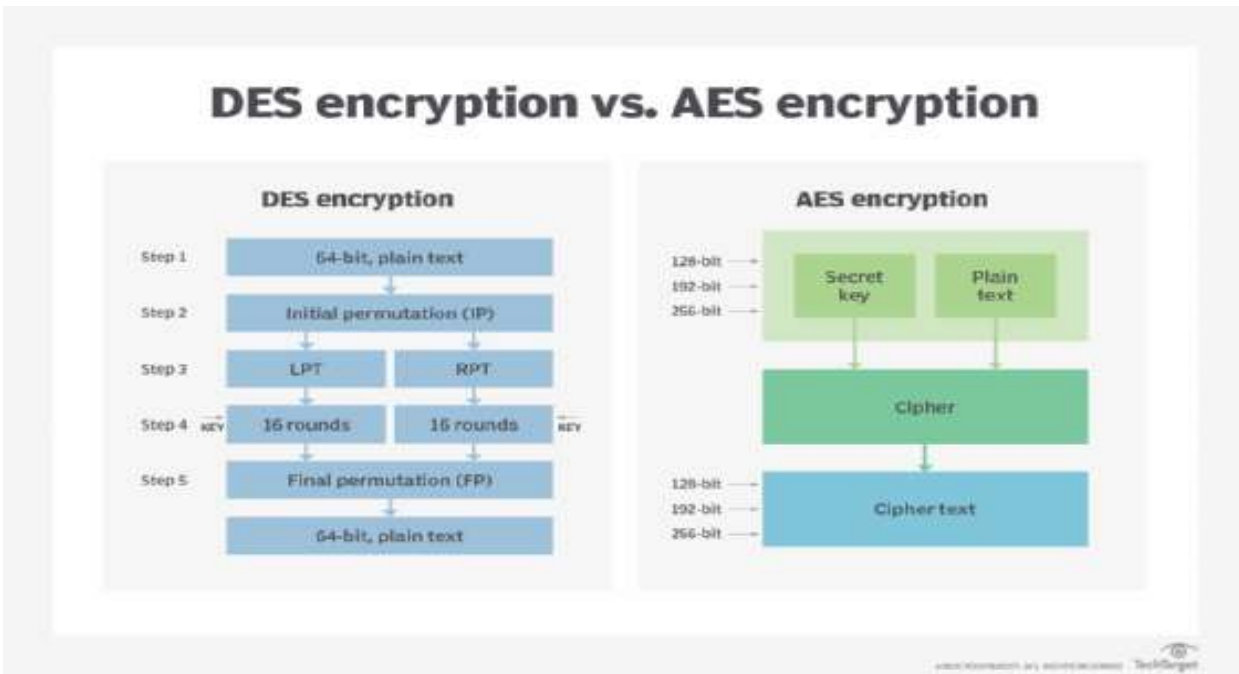
Rijndael has a number of [advantages over DES](#) and [Triple DES](#).

First, its block sizes can mirror those of their respective keys, which places this algorithm over the limits required for AES design conditions. It is also easy to implement with simple components and easily proven properties. Additionally, Rijndael works three times faster than DES in software. In Rijndael, it is possible to use 160- or 224-bit keys. The algorithm also supports block sizes of 160 or 224 bits. This enables greater flexibility in transformation rounds during encryption.

The Square block cipher on which Rijndael is based was vulnerable to several [cyber attacks](#). Rijndael improves the security of this older algorithm by using alternating mix column and mix row transformations to

resist such attacks. Since the key length can vary as desired, the algorithm is also secure for many real-world applications, especially when more transformation rounds are added.

That's why Rijndael is ideal for the secure exchange of keys and to transmit data with a length of 128 or 256 bits.



AES, like the Rijndael algorithm, works differently and has a number of advantages over DES.

Compared to other algorithms that were candidates for AES, Rijndael provides strong security against the following:

- linear cryptanalysis
- differential cryptanalysis
- opportunistic attacks
- power attacks
- [timing attacks](#)

Rijndael also has low [memory](#) requirements, which makes it suitable for space-restricted environments. Its rich algebraic structure makes it possible to easily and quickly assess its security. Plus, it offers high efficiency and performance on many computing platforms and hardware and software environments, including the following:

- large [arrays](#)
- desktops
- laptops
- mobile devices
- [smart cards](#)

### Drawbacks of Rijndael

Rijndael may be vulnerable to a type of attack called the *square attack*. But, in practical terms, this attack cannot compromise the security of the algorithm.

The algorithm is also limited by its inverse cipher that takes more code and cycles and is, therefore, not ideally suited for some implementations.

### Cipher Modes:

A block cipher processes the data blocks of fixed size. Usually, the size of a message is larger than the block size. Hence, the long message is divided into a series of sequential message blocks, and the cipher operates on these blocks one at a time.

### Electronic Code Book (ECB) Mode

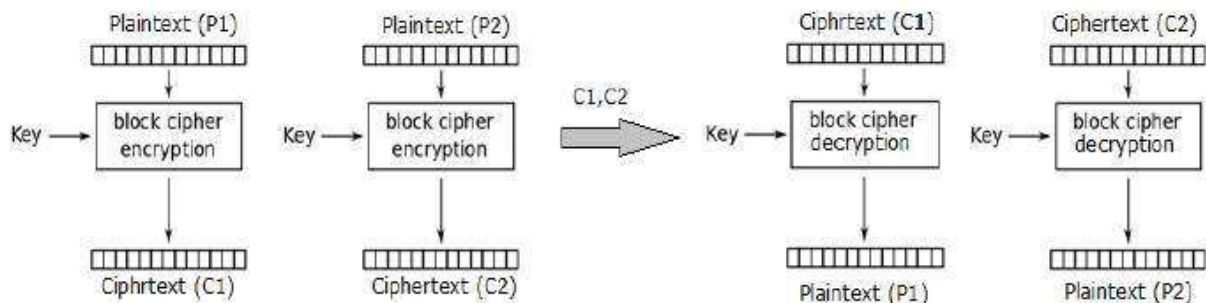
This mode is a most straightforward way of processing a series of sequentially listed message blocks.

#### Operation

- The user takes the first block of plaintext and encrypts it with the key to produce the first block of ciphertext.
- He then takes the second block of plaintext and follows the same process with same key and so on so forth.

The ECB mode is **deterministic**, that is, if plaintext block  $P_1, P_2, \dots, P_m$  are encrypted twice under the same key, the output ciphertext blocks will be the same.

In fact, for a given key technically we can create a codebook of ciphertexts for all possible plaintext blocks. Encryption would then entail only looking up for required plaintext and select the corresponding ciphertext. Thus, the operation is analogous to the assignment of code words in a codebook, and hence gets an official name – Electronic Codebook mode of operation (ECB). It is illustrated as follows –



#### Analysis of ECB Mode

In reality, any application data usually have partial information which can be guessed. For example, the range of salary can be guessed. A ciphertext from ECB can allow an attacker to guess the plaintext by trial-and-error if the plaintext message is within predictable.

For example, if a ciphertext from the ECB mode is known to encrypt a salary figure, then a small number of trials will allow an attacker to recover the figure. In general, we do not wish to use a deterministic cipher, and hence the ECB mode should not be used in most applications.

### Cipher Block Chaining (CBC) Mode

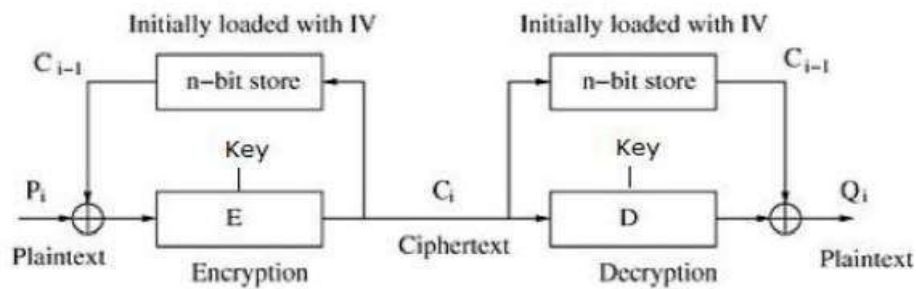
CBC mode of operation provides message dependence for generating ciphertext and makes the system non-deterministic.

#### Operation

The operation of CBC mode is depicted in the following illustration. The steps are as follows –

- Load the n-bit Initialization Vector (IV) in the top register.
- XOR the n-bit plaintext block with data value in top register.

- Encrypt the result of XOR operation with underlying block cipher with key  $K$ .
- Feed ciphertext block into top register and continue the operation till all plaintext blocks are processed.
- For decryption, IV data is XORed with first ciphertext block decrypted. The first ciphertext block is also fed into to register replacing IV for decrypting next ciphertext block.



### Analysis of CBC Mode

In CBC mode, the current plaintext block is added to the previous ciphertext block, and then the result is encrypted with the key. Decryption is thus the reverse process, which involves decrypting the current ciphertext and then adding the previous ciphertext block to the result.

Advantage of CBC over ECB is that changing IV results in different ciphertext for identical message. On the drawback side, the error in transmission gets propagated to few further block during decryption due to chaining effect.

It is worth mentioning that CBC mode forms the basis for a well-known data origin authentication mechanism. Thus, it has an advantage for those applications that require both symmetric encryption and data origin authentication.

### Cipher Feedback (CFB) Mode

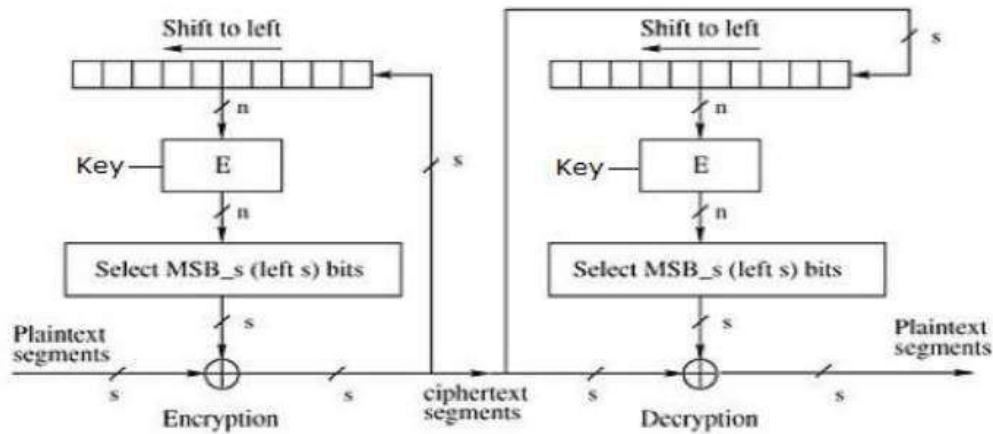
In this mode, each ciphertext block gets 'fed back' into the encryption process in order to encrypt the next plaintext block.

#### Operation

The operation of CFB mode is depicted in the following illustration. For example, in the present system, a message block has a size 's' bits where  $1 < s < n$ . The CFB mode requires an initialization vector (IV) as the initial random n-bit input block. The IV need not be secret. Steps of operation are –

- Load the IV in the top register.
- Encrypt the data value in top register with underlying block cipher with key  $K$ .
- Take only 's' number of most significant bits (left bits) of output of encryption process and XOR them with 's' bit plaintext message block to generate ciphertext block.
- Feed ciphertext block into top register by shifting already present data to the left and continue the operation till all plaintext blocks are processed.
- Essentially, the previous ciphertext block is encrypted with the key, and then the result is XORed to the current plaintext block.
- Similar steps are followed for decryption. Pre-decided IV is initially loaded at the start of decryption.





### Analysis of CFB Mode

CFB mode differs significantly from ECB mode, the ciphertext corresponding to a given plaintext block depends not just on that plaintext block and the key, but also on the previous ciphertext block. In other words, the ciphertext block is dependent of message.

CFB has a very strange feature. In this mode, user decrypts the ciphertext using only the encryption process of the block cipher. The decryption algorithm of the underlying block cipher is never used.

Apparently, CFB mode is converting a block cipher into a type of stream cipher. The encryption algorithm is used as a key-stream generator to produce key-stream that is placed in the bottom register. This key stream is then XORed with the plaintext as in case of stream cipher.

By converting a block cipher into a stream cipher, CFB mode provides some of the advantageous properties of a stream cipher while retaining the advantageous properties of a block cipher.

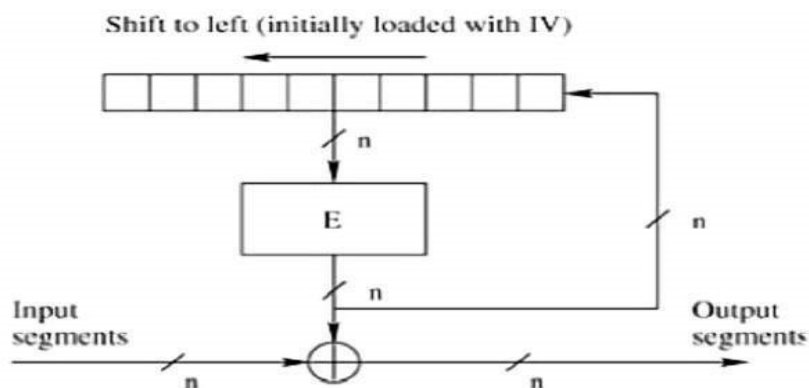
On the flip side, the error of transmission gets propagated due to changing of blocks.

### Output Feedback (OFB) Mode

It involves feeding the successive output blocks from the underlying block cipher back to it. These feedback blocks provide string of bits to feed the encryption algorithm which act as the key-stream generator as in case of CFB mode.

The key stream generated is XOR-ed with the plaintext blocks. The OFB mode requires an IV as the initial random n-bit input block. The IV need not be secret.

The operation is depicted in the following illustration –



### Counter (CTR) Mode

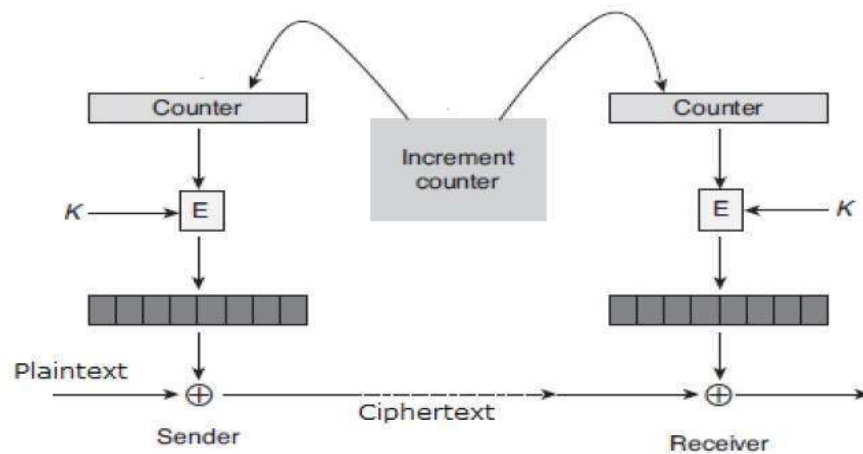
It can be considered as a counter-based version of CFB mode without the feedback. In this mode, both the sender and receiver need to access to a reliable counter, which computes a new shared value each time a

ciphertext block is exchanged. This shared counter is not necessarily a secret value, but challenge is that both sides must keep the counter synchronized.

### Operation

Both encryption and decryption in CTR mode are depicted in the following illustration. Steps in operation are –

- Load the initial counter value in the top register is the same for both the sender and the receiver. It plays the same role as the IV in CFB (and CBC) mode.
- Encrypt the contents of the counter with the key and place the result in the bottom register.
- Take the first plaintext block P1 and XOR this to the contents of the bottom register. The result of this is C1. Send C1 to the receiver and update the counter. The counter update replaces the ciphertext feedback in CFB mode.
- Continue in this manner until the last plaintext block has been encrypted.
- The decryption is the reverse process. The ciphertext block is XORed with the output of encrypted contents of counter value. After decryption of each ciphertext block counter is updated as in case of encryption.



### Analysis of Counter Mode

It does not have message dependency and hence a ciphertext block does not depend on the previous plaintext blocks.

Like CFB mode, CTR mode does not involve the decryption process of the block cipher. This is because the CTR mode is really using the block cipher to generate a key-stream, which is encrypted using the XOR function. In other words, CTR mode also converts a block cipher to a stream cipher.

The serious disadvantage of CTR mode is that it requires a synchronous counter at sender and receiver. Loss of synchronization leads to incorrect recovery of plaintext.

However, CTR mode has almost all advantages of CFB mode. In addition, it does not propagate error of transmission at all.

### Other Ciphers:

### Cryptanalysis:

Cryptanalysis is the study of [ciphertext](#), ciphers and cryptosystems with the aim of understanding how they work and finding and improving techniques for defeating or weakening them. For example, cryptanalysts seek to decrypt ciphertexts without knowledge of the [plaintext](#) source, encryption key or the algorithm used to encrypt it; cryptanalysts also target secure [hashing](#), digital signatures and other cryptographic algorithms.

## Cryptanalysis techniques and attacks

There are many different types of cryptanalysis attacks and techniques, which vary depending on how much information the analyst has about the ciphertext being analyzed. Some cryptanalytic methods include:

- In a *ciphertext-only attack*, the attacker only has access to one or more encrypted messages but knows nothing about the plaintext data, the encryption algorithm being used or any data about the cryptographic key being used. This is the type of challenge that intelligence agencies often face when they have intercepted encrypted communications from an opponent.
- In a *known plaintext attack*, the analyst may have access to some or all of the plaintext of the ciphertext; the analyst's goal in this case is to discover the key used to encrypt the message and decrypt the message. Once the key is discovered, an attacker can decrypt all messages that had been encrypted using that key. Linear cryptanalysis is a type of known plaintext attack that uses a linear approximation to describe how a [block cipher](#). Known plaintext attacks depend on the attacker being able to discover or guess some or all of an encrypted message, or even the format of the original plaintext. For example, if the attacker is aware that a particular message is addressed to or about a particular person, that person's name may be a suitable known plaintext.
- In a *chosen plaintext attack*, the analyst either knows the encryption algorithm or has access to the device used to do the encryption. The analyst can encrypt the chosen plaintext with the targeted algorithm to derive information about the key.
- A *differential cryptanalysis attack* is a type of chosen plaintext attack on block ciphers that analyzes pairs of plaintexts rather than single plaintexts, so the analyst can determine how the targeted algorithm works when it encounters different types of data.
- *Integral cryptanalysis attacks* are similar to differential cryptanalysis attacks, but instead of pairs of plaintexts, it uses sets of plaintexts in which part of the plaintext is kept constant but the rest of the plaintext is modified. This attack can be especially useful when applied to block ciphers that are based on substitution-permutation networks.
- A *side-channel attack* depends on information collected from the physical system being used to encrypt or decrypt. Successful side-channel attacks use data that is neither the ciphertext resulting from the encryption process nor the plaintext to be encrypted, but rather may be related to the amount of time it takes for a system to respond to specific queries, the amount of power consumed by the encrypting system, or electromagnetic radiation emitted by the encrypting system.
- A [dictionary attack](#) is a technique typically used against password files and exploits the human tendency to use passwords based on natural words or easily guessed sequences of letters or numbers. The dictionary attack works by encrypting all the words in a dictionary and then checking whether the resulting hash matches an encrypted password stored in the SAM file format or other password file.
- [Man-in-the-middle attacks](#) occur when cryptanalysts find ways to insert themselves into the communication channel between two parties who wish to exchange their keys for secure communication via asymmetric or [public key infrastructure](#). The attacker then performs a key exchange with each party, with the original parties believing they are exchanging keys with each other. The two parties then end up using keys that are known to the attacker.

Other types of cryptanalytic attacks can include techniques for convincing individuals to reveal their passwords or encryption keys, developing [Trojan horse](#) programs that steal secret keys from victims' computers and send them back to the cryptanalyst, or tricking a victim into using a weakened cryptosystem.

Side-channel attacks have also been known as timing or differential power analysis. These attacks came to wide notice in the late 1990s when cryptographer Paul Kocher was publishing results of his research into timing attacks and differential power analysis attacks on [Diffie-Hellman](#), RSA, Digital Signature Standard (DSS) and other cryptosystems, especially against implementations on [smart cards](#).

## Public Key Algorithms:

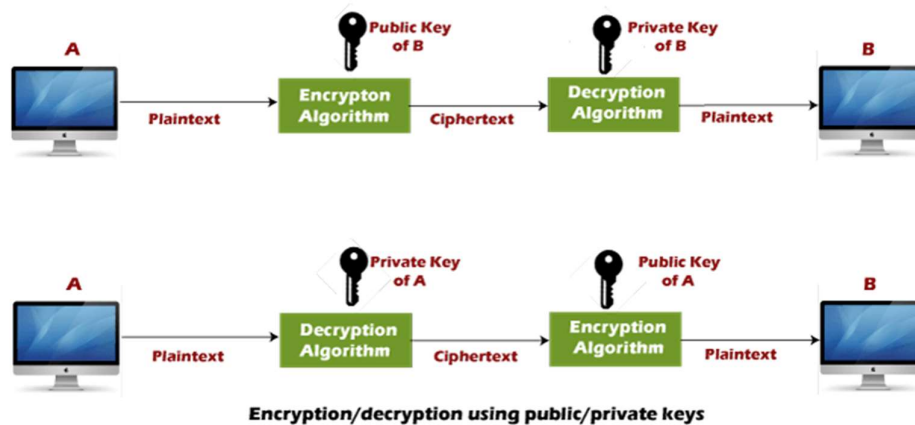
Public key encryption algorithm:

Public Key encryption algorithm is also called the Asymmetric algorithm. Asymmetric algorithms are those algorithms in which sender and receiver use different keys for encryption and decryption. Each sender is assigned a pair of keys:

- **Public key**
- **Private key**

The **Public key** is used for encryption, and the **Private Key** is used for decryption. Decryption cannot be done using a public key. The two keys are linked, but the private key cannot be derived from the public key. The public key is well known, but the private key is secret and it is known only to the user who owns the key. It means that everybody can send a message to the user using user's public key. But only the user can decrypt the message using his private key.

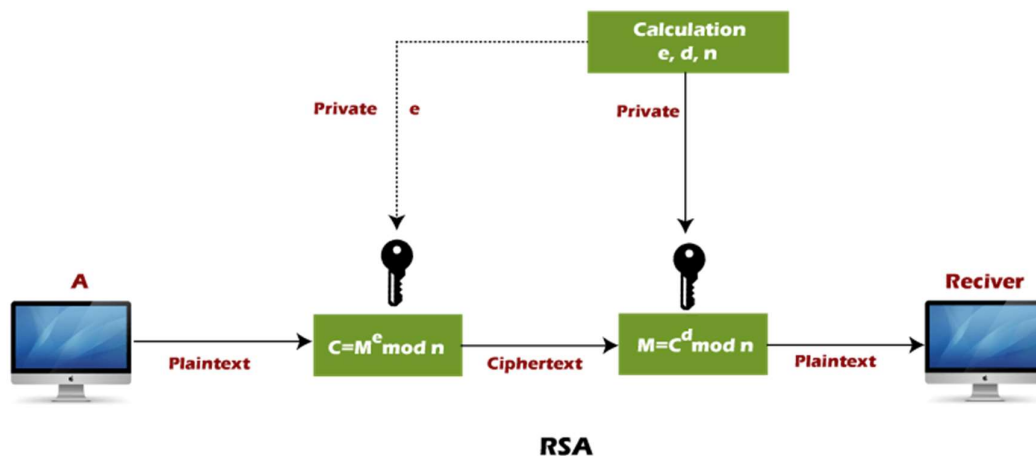
The Public key algorithm operates in the following manner:



- The data to be sent is encrypted by sender A using the public key of the intended receiver
- B decrypts the received ciphertext using its private key, which is known only to B. B replies to A encrypting its message using A's public key.
- A decrypts the received ciphertext using its private key, which is known only to him.

RSA encryption algorithm:

RSA is the most common public-key algorithm, named after its inventors **Rivest, Shamir, and Adelman (RSA)**.



**RSA algorithm uses the following procedure to generate public and private keys:**

- Select two large prime numbers,  $p$  and  $q$ .
- Multiply these numbers to find  $n = p \times q$ , where  $n$  is called the modulus for encryption and decryption.
- Choose a number  $e$  less than  $n$ , such that  $n$  is relatively prime to  $(p - 1) \times (q - 1)$ . It means that  $e$  and  $(p - 1) \times (q - 1)$  have no common factor except 1. Choose " $e$ " such that  $1 < e < \phi(n)$ ,  $e$  is prime to  $\phi(n)$ ,  
 **$\gcd(e, \phi(n)) = 1$**
- If  $n = p \times q$ , then the public key is  $\langle e, n \rangle$ . A plaintext message  $m$  is encrypted using public key  $\langle e, n \rangle$ . To find ciphertext from the plain text following formula is used to get ciphertext  $C$ .  
 **$C = m^e \bmod n$**   
Here,  $m$  must be less than  $n$ . A larger message ( $>n$ ) is treated as a concatenation of messages, each of which is encrypted separately.
- To determine the private key, we use the following formula to calculate the  $d$  such that:  
 **$D_e \bmod \{(p - 1) \times (q - 1)\} = 1$**   
**Or**  
 **$D_e \bmod \phi(n) = 1$**
- The private key is  $\langle d, n \rangle$ . A ciphertext message  $c$  is decrypted using private key  $\langle d, n \rangle$ . To calculate plain text  $m$  from the ciphertext  $c$  following formula is used to get plain text  $m$ .  
 **$m = c^d \bmod n$**

Let's take some example of RSA encryption algorithm:

Example 1:

This example shows how we can encrypt plaintext 9 using the RSA public-key encryption algorithm. This example uses prime numbers 7 and 11 to generate the public and private keys.

**Explanation:**

**Step 1:** Select two large prime numbers,  $p$ , and  $q$ .

$$p = 7$$

$$q = 11$$

**Step 2:** Multiply these numbers to find  $n = p \times q$ , where  $n$  is called the modulus for encryption and decryption.

First, we calculate

$$n = p \times q$$

$$n = 7 \times 11$$

$$n = 77$$

**Step 3:** Choose a number  $e$  less than  $n$ , such that  $n$  is relatively prime to  $(p - 1) \times (q - 1)$ . It means that  $e$  and  $(p - 1) \times (q - 1)$  have no common factor except 1. Choose " $e$ " such that  $1 < e < \phi(n)$ ,  $e$  is prime to  $\phi(n)$ ,  $\gcd(e, \phi(n)) = 1$ .

Second, we calculate

$$\phi(n) = (p - 1) \times (q - 1)$$

$$\phi(n) = (7 - 1) \times (11 - 1)$$

$$\phi(n) = 6 \times 10$$

$$\phi(n) = 60$$

Let us now choose relative prime  $e$  of 60 as 7.

Thus the public key is  $\langle e, n \rangle = (7, 77)$

**Step 4:** A plaintext message  $m$  is encrypted using public key  $\langle e, n \rangle$ . To find ciphertext from the plain text following formula is used to get ciphertext  $C$ .

To find ciphertext from the plain text following formula is used to get ciphertext  $C$ .

$$C = m^e \bmod n$$

$$C = 9^7 \bmod 77$$

$$C = 37$$

**Step 5:** The private key is  $\langle d, n \rangle$ . To determine the private key, we use the following formula  $d$  such that:

$$D_e \bmod \{(p-1) \times (q-1)\} = 1$$

$$7d \bmod 60 = 1, \text{ which gives } d = 43$$

The private key is  $\langle d, n \rangle = (43, 77)$

**Step 6:** A ciphertext message  $c$  is decrypted using private key  $\langle d, n \rangle$ . To calculate plain text  $m$  from the ciphertext  $c$  following formula is used to get plain text  $m$ .

$$m = c^d \bmod n$$

$$m = 37^{43} \bmod 77$$

$$m = 9$$

In this example, Plain text = 9 and the ciphertext = 37

Example 2:

In an RSA cryptosystem, a particular A uses two prime numbers, 13 and 17, to generate the public and private keys. If the public of A is 35. Then the private key of A is .....?.

**Explanation:**

**Step 1:** in the first step, select two large prime numbers,  $p$  and  $q$ .

$$p = 13$$

$$q = 17$$

**Step 2:** Multiply these numbers to find  $n = p \times q$ , where  $n$  is called the modulus for encryption and decryption.

First, we calculate

$$n = p \times q$$

$$n = 13 \times 17$$

$$n = 221$$

**Step 3:** Choose a number  $e$  less than  $n$ , such that  $n$  is relatively prime to  $(p-1) \times (q-1)$ . It means that  $e$  and  $(p-1) \times (q-1)$  have no common factor except 1. Choose " $e$ " such that  $1 < e < \phi(n)$ ,  $e$  is prime to  $\phi(n)$ ,  $\gcd(e, \phi(n)) = 1$ .

Second, we calculate

$$\phi(n) = (p-1) \times (q-1)$$

$$\phi(n) = (13 - 1) \times (17 - 1)$$

$$\phi(n) = 12 \times 16$$

$$\phi(n) = 192$$

$$\text{g.c.d}(35, 192) = 1$$

**Step 3:** To determine the private key, we use the following formula to calculate the d such that:

$$\text{Calculate } d = d_e \text{ mod } \phi(n) = 1$$

$$d = d \times 35 \text{ mod } 192 = 1$$

$$d = (1 + k \cdot \phi(n)) / e \quad [\text{let } k = 0, 1, 2, 3, \dots]$$

**Put k = 0**

$$d = (1 + 0 \times 192) / 35$$

$$d = 1/35$$

**Put k = 1**

$$d = (1 + 1 \times 192) / 35$$

$$d = 193/35$$

**Put k = 2**

$$d = (1 + 2 \times 192) / 35$$

$$d = 385/35$$

$$d = 11$$

The private key is  $\langle d, n \rangle = (11, 221)$

Hence, private key i.e.  $d = 11$

### Digital Signatures:

Digital signatures are the public-key primitives of message authentication. In the physical world, it is common to use handwritten signatures on handwritten or typed messages. They are used to bind signatory to the message.

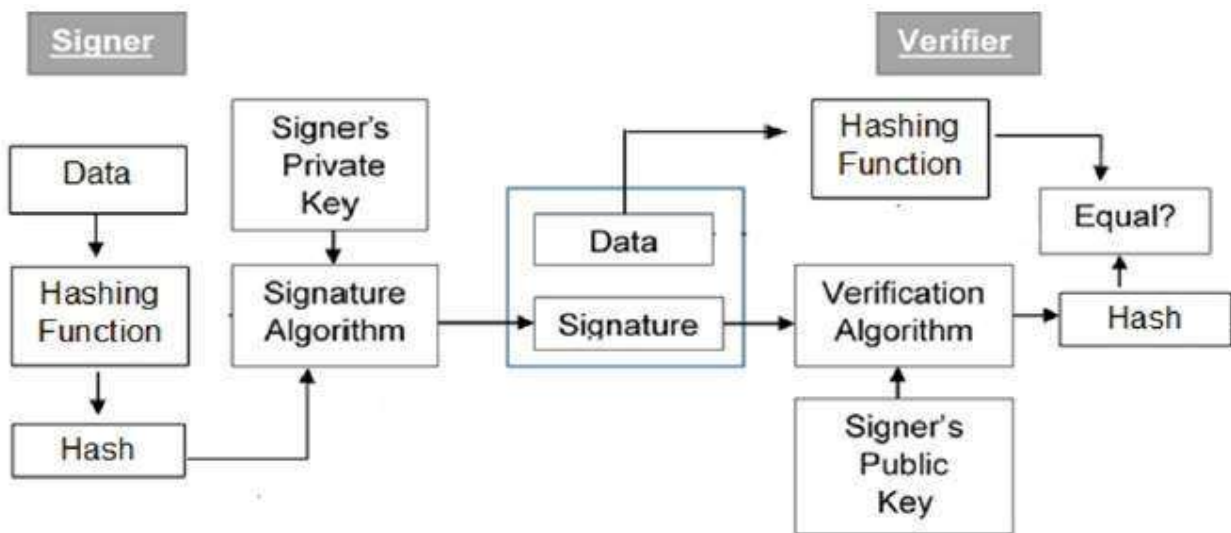
Similarly, a digital signature is a technique that binds a person/entity to the digital data. This binding can be independently verified by receiver as well as any third party.

Digital signature is a cryptographic value that is calculated from the data and a secret key known only by the signer.

In real world, the receiver of message needs assurance that the message belongs to the sender and he should not be able to repudiate the origination of that message. This requirement is very crucial in business applications, since likelihood of a dispute over exchanged data is very high.

### Model of Digital Signature

As mentioned earlier, the digital signature scheme is based on public key cryptography. The model of digital signature scheme is depicted in the following illustration –



The following points explain the entire process in detail –

- Each person adopting this scheme has a public-private key pair.
- Generally, the key pairs used for encryption/decryption and signing/verifying are different. The private key used for signing is referred to as the signature key and the public key as the verification key.
- Signer feeds data to the hash function and generates hash of data.
- Hash value and signature key are then fed to the signature algorithm which produces the digital signature on given hash. Signature is appended to the data and then both are sent to the verifier.
- Verifier feeds the digital signature and the verification key into the verification algorithm. The verification algorithm gives some value as output.
- Verifier also runs same hash function on received data to generate hash value.
- For verification, this hash value and output of verification algorithm are compared. Based on the comparison result, verifier decides whether the digital signature is valid.
- Since digital signature is created by ‘private’ key of signer and no one else can have this key; the signer cannot repudiate signing the data in future.

It should be noticed that instead of signing data directly by signing algorithm, usually a hash of data is created. Since the hash of data is a unique representation of data, it is sufficient to sign the hash in place of data. The most important reason of using hash instead of data directly for signing is efficiency of the scheme.

Let us assume RSA is used as the signing algorithm. As discussed in public key encryption chapter, the encryption/signing process using RSA involves modular exponentiation.

Signing large data through modular exponentiation is computationally expensive and time consuming. The hash of the data is a relatively small digest of the data, hence **signing a hash is more efficient than signing the entire data**.

**Symmetric Key Signatures:**

Symmetric key signatures are a type of digital signature that use a shared secret key to authenticate the sender of a message. They are used to provide authentication, integrity, and non-repudiability for electronic messages and documents.

There are several reasons why symmetric key signatures may be used –

- **Speed and efficiency** – Symmetric key algorithms are generally faster and more efficient than their asymmetric counterparts, making them well-suited for high-volume applications.



- **Simplicity** – Symmetric key signatures are simpler to implement and use than asymmetric key signatures, as they do not require the complex key management infrastructure that is required for asymmetric key signatures.
- **Security** – Symmetric key signatures provide strong security if the secret key is kept secret and is not compromised.

Overall, symmetric key signatures can be an effective tool for authenticating the sender of a message and providing secure communication in certain situations. However, they do have some limitations, such as the need to securely exchange the secret key in advance and the inability to provide non-repudiability for the recipient of the message.

Public key Signatures:

A public key signature is a type of digital signature that uses a pair of keys – a public key and a private key – to authenticate the sender of a message. The private key is kept secret by the sender and is used to create the signature, while the public key is made available to anyone who wants to verify the signature.

Public key signatures are based on the principles of public key cryptography, in which a message encrypted with a public key can only be decrypted with the corresponding private key. This allows the sender of a message to sign the message using their private key, and for the recipient to verify the signature using the sender's public key.

Public key signatures have several advantages over symmetric key signatures. They do not require the sender and recipient to exchange a secret key in advance, and they can provide non-repudiability for the recipient of the message, meaning that the sender cannot later deny having sent the message.

Overall, public key signatures are a widely used tool for secure communication and are an important component of many cybersecurity systems.

#### A Simple PKI Digital Signature Definition & Analogy

A Public Key Infrastructure (PKI) digital signature is a type of digital signature that uses a pair of keys – a public key and a private key – to authenticate the sender of a message. The private key is kept secret by the sender and is used to create the signature, while the public key is made available to anyone who wants to verify the signature.

An analogy for a PKI digital signature is a physical signature on a paper document. Just as a physical signature serves as a way to authenticate the sender of a document, a PKI digital signature serves as a way to authenticate the sender of an electronic message or document.

To create a PKI digital signature, the sender uses their private key to apply a digital signature to the message. This process involves applying a mathematical function to the message, resulting in a unique signature that is specific to the message and the sender's private key.

To verify the PKI digital signature, the recipient uses the sender's public key to apply the same mathematical function to the message and compare the result to the signature. If the two match, it is an indication that the message was indeed sent by the sender and has not been tampered with.

Overall, PKI digital signatures are a widely used tool for secure communication and are an important component of many cybersecurity systems.

#### Why Public Key Signature is important for Internet Communications?

Public key signatures are important for internet communications because they provide a secure and efficient way to authenticate the sender of a message and ensure the integrity of the message.

In the context of internet communications, it is often necessary to exchange messages and documents over public networks, such as the internet, where the security of the communication channel cannot be guaranteed.

Public key signatures provide a way to secure these communications by allowing the sender to apply a digital signature to the message using their private key, and for the recipient to verify the signature using the sender's public key.

Some of the specific advantages of public key signatures for internet communications include –

- **Non-repudiability** – Public key signatures provide non-repudiability for the recipient of the message, meaning that the sender cannot later deny having sent the message. This is an important property for ensuring the authenticity and integrity of electronic communications.
- **Efficient key exchange** – Public key signatures do not require the sender and recipient to exchange a secret key in advance, which can be a challenge in situations where the sender and recipient do not already have a secure channel through which to share the key.
- **Flexibility** – Public key signatures can be used in a variety of applications, including email, file transfer, and online transactions, making them a flexible and widely applicable tool for secure communication.

Overall, public key signatures are an important tool for ensuring the security and integrity of internet communications and are widely used in many cybersecurity systems.

#### 5 Uses for a Public Key Signature

Public key signatures are a widely used tool for secure communication and have many practical applications. Some common uses for public key signatures include –

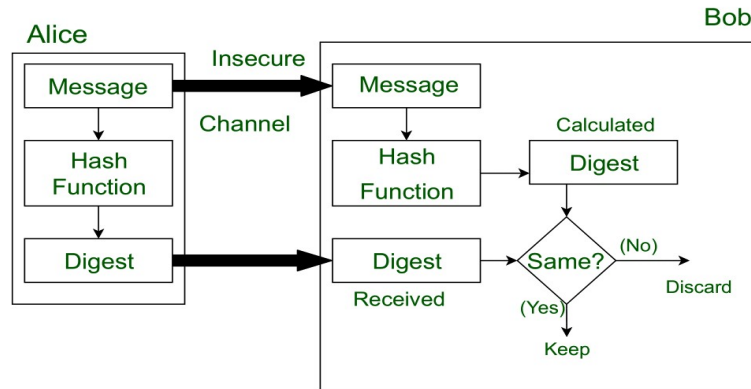
- **Email** – Public key signatures can be used to secure email communications by allowing the sender to apply a digital signature to the message using their private key and for the recipient to verify the signature using the sender's public key. This can help to ensure the authenticity and integrity of the message.
- **File transfer** – Public key signatures can be used to securely transfer files over the internet by allowing the sender to apply a digital signature to the file and for the recipient to verify the signature. This can help to ensure that the file has not been tampered with during transit.
- **Online transactions** – Public key signatures can be used to secure online transactions, such as online banking and e-commerce, by allowing the sender to apply a digital signature to the transaction and for the recipient to verify the signature. This can help to ensure the authenticity and integrity of the transaction.
- **Identity verification** – Public key signatures can be used to verify the identity of a user in an online setting by allowing the user to apply a digital signature to a message using their private key and for the recipient to verify the signature using the user's public key.
- **Document signing** – Public key signatures can be used to sign electronic documents, such as contracts and legal agreements, by allowing the sender to apply a digital signature to the document and for the recipient to verify the signature. This can help to ensure the authenticity and integrity of the document.

Overall, public key signatures have many practical applications and are an important tool for secure communication in a variety of settings.

#### Message Digest:

**Message Digest** is used to ensure the integrity of a message transmitted over an insecure channel (where the content of the message can be changed). The message is passed through a [Cryptographic hash function](#). This function creates a compressed image of the message called **Digest**.

Lets assume, Alice sent a message and digest pair to Bob. To check the integrity of the message Bob runs the cryptographic hash function on the received message and gets a new digest. Now, Bob will compare the new digest and the digest sent by Alice. If, both are same then Bob is sure that the original message is not changed.



This message and digest pair is equivalent to a physical document and fingerprint of a person on that document. Unlike the physical document and the fingerprint, the message and the digest can be sent separately.

- Most importantly, the digest should be unchanged during the transmission.
- The cryptographic hash function is a one way function, that is, a function which is practically infeasible to invert. This cryptographic hash function takes a message of variable length as input and creates a **digest / hash / fingerprint** of fixed length, which is used to verify the integrity of the message.
- Message digest ensures the integrity of the document. To provide authenticity of the message, digest is encrypted with sender's private key. Now this digest is called digital signature, which can be only decrypted by the receiver who has sender's public key. Now the receiver can authenticate the sender and also verify the integrity of the sent message.

SHA 1 and SHA 2:

SHA is the acronym for Secure Hash Algorithm, used for hashing data and certificate files. Every piece of data produces a unique hash that is thoroughly non-duplicable by any other piece of data. The resulting digital signature is unique too as it depends on the hash that's generated out of the data. For the course of the actual communication, symmetric cryptography is used, where the same key that hashes or encrypts data is used to decrypt it.

Digital certificates follow the same hashing mechanism, wherein the certificate file is hashed, and the hashed file is digitally signed by the CA issuing the certificate. The most critical part of any electronic communication is authentication, that is, to make sure that the entity at the other end of the channel is genuinely the one that the session initiator wants to communicate with. That is why the TLS protocol enforces a more stringent authentication measure that uses asymmetric cryptography.

SHA is the cryptographic algorithm adopted by the PKI market for digital signatures. SHA-1 and SHA-2 are two versions of this algorithm. The difference between these two versions lies in the "length" or the "number of bits" that the hashed output (called message digest) contains for a given plaintext input. Logically, the more the number of bits the digest has, the more difficult it is to break it using brute force. The SHA-2 function produces a 256-bit digest (this is the commonly used function in the family of SHA-2; the functions range from 224 to 512-bit) while the SHA-1 function produces a 160-bit digest for the same input. SHA-1 offers weak security as it sometimes gives the same digest for two different data values, owing to its limited bit-length and therefore possible hash combinations, while SHA-2 produces a unique digest for every data value as a large number of combinations are possible in it ( $2^{256}$  possible combinations for a 256-bit function). In

2016, the TLS/SSL industry enforced the move to SHA-2, and this algorithm has been in use until the present day.

MD5:

**MD5** is a cryptographic hash function algorithm that takes the message as input of any length and changes it into a fixed-length message of 16 bytes. MD5 algorithm stands for the **message-digest algorithm**. MD5 was developed as an improvement of MD4, with advanced security purposes. The output of MD5 (Digest size) is always **128 bits**. **MD5** was developed in 1991 by **Ronald Rivest**.

#### Use Of MD5 Algorithm:

- It is used for file authentication.
- In a web application, it is used for security purposes. e.g. Secure password of users etc.
- Using this algorithm, We can store our password in 128 bits format.



#### MD5 Algorithm

##### Working of the MD5 Algorithm:

MD5 algorithm follows the following steps

**1. Append Padding Bits:** In the first step, we add padding bits in the original message in such a way that the total length of the message is 64 bits less than the exact multiple of 512.

Suppose we are given a message of 1000 bits. Now we have to add padding bits to the original message. Here we will add 472 padding bits to the original message. After adding the padding bits the size of the original message/output of the first step will be 1472 i.e. 64 bits less than an exact multiple of 512 (i.e.  $512 \times 3 = 1536$ ).

**Length(original message + padding bits) =  $512 * i - 64$**  where  $i = 1, 2, 3 \dots$

**2. Append Length Bits:** In this step, we add the length bit in the output of the first step in such a way that the total number of the bits is the perfect multiple of 512. Simply, here we add the 64-bit as a length bit in the output of the first step.

i.e. output of first step =  $512 * n - 64$

length bits = 64.

After adding both we will get  $512 * n$  i.e. the exact multiple of 512.

**3. Initialize MD buffer:** Here, we use the 4 buffers i.e. J, K, L, and M. The size of each buffer is 32 bits.

- J = 0x67425301

- K = 0xEDFCBA45

- L = 0x98CBADFE

- M = 0x13DCE476

**4. Process Each 512-bit Block:** This is the most important step of the MD5 algorithm. Here, a total of 64 operations are performed in 4 rounds. In the 1st round, 16 operations will be performed, 2nd round 16 operations will be performed, 3rd round 16 operations will be performed, and in the 4th round, 16 operations will be performed. We apply a different function on each round i.e. for the 1st round we apply the F

function, for the 2nd G function, 3rd for the H function, and 4th for the I function.

We perform OR, AND, XOR, and NOT (basically these are logic gates) for calculating functions. We use 3 buffers for each function i.e. K, L, M.

$$- F(K,L,M) = (K \text{ AND } L) \text{ OR } (\text{NOT } K \text{ AND } M)$$

$$- G(K,L,M) = (K \text{ AND } L) \text{ OR } (L \text{ AND } \text{NOT } M)$$

$$- H(K,L,M) = K \text{ XOR } L \text{ XOR } M$$

$$- I(K,L,M) = L \text{ XOR } (K \text{ OR } \text{NOT } M)$$

After applying the function now we perform an operation on each block. For performing operations we need

- add modulo  $2^{32}$
- $M[i]$  – 32 bit message.
- $K[i]$  – 32-bit constant.
- $\lll n$  – Left shift by n bits.

Now take input as initialize MD buffer i.e. J, K, L, M. Output of K will be fed in L, L will be fed into M, and M will be fed into J. After doing this now we perform some operations to find the output for J.

- In the first step, Outputs of K, L, and M are taken and then the function F is applied to them. We will add modulo  $2^{32}$  bits for the output of this with J.
- In the second step, we add the  $M[i]$  bit message with the output of the first step.
- Then add 32 bits constant i.e.  $K[i]$  to the output of the second step.
- At last, we do left shift operation by n (can be any value of n) and addition modulo by  $2^{32}$ .

After all steps, the result of J will be fed into K. Now same steps will be used for all functions G, H, and I. After performing all 64 operations we will get our message digest.

### **Output:**

After all, rounds have been performed, the buffer J, K, L, and M contains the MD5 output starting with the lower bit J and ending with Higher bits M.

### **Birthday Attack:**

Birthday attack is a type of cryptographic attack that belongs to a class of brute force attacks. It exploits the mathematics behind the birthday problem in probability theory. The success of this attack largely depends upon the higher likelihood of collisions found between random attack attempts and a fixed degree of permutations, as described in the birthday paradox problem.

### **Birthday paradox problem –**

Let us consider the example of a classroom of 30 students and a teacher. The teacher wishes to find pairs of students that have the same birthday. Hence the teacher asks for everyone's birthday to find such pairs. Intuitively this value may seem small. For example, if the teacher fixes a particular date say **October 10**, then the probability that at least one student is born on that day is  $1 - (364/365)^{30}$  which is about 7.9%. However, the probability that at least one student has the same birthday as any other student is around 70% using the following formula:

$$1 - 365!/((365 - n!) * (365^n)) \text{ (substituting } n = 30 \text{ here)}$$

### Derivation of the above term:

#### Assumptions –

1. Assuming a non leap year(hence 365 days).
2. Assuming that a person has an equally likely chance of being born on any day of the year.

Let us consider  $n = 2$ .

$$\begin{aligned}P(\text{Two people have the same birthday}) &= 1 - P(\text{Two people having different birthday}) \\&= 1 - (365/365) * (364/365) \\&= 1 - 1 * (364/365) \\&= 1 - 364/365 \\&= 1/365.\end{aligned}$$

So for  $n$  people, the probability that all of them have different birthdays is:

$$\begin{aligned}P(N \text{ people having different birthdays}) &= (365/365) * (365-1/365) * (365-2/365) * \dots * (365-n+1)/365. \\&= 365! / ((365-n)! * 365^n)\end{aligned}$$

#### Hash function –

A hash function  $H$  is a transformation that takes a *variable sized input  $m$*  and returns a *fixed size string* called a *hash value* ( $h = H(m)$ ). Hash functions chosen in cryptography must satisfy the following requirements:

- The input is of variable length,
- The output has a fixed length,
- $H(x)$  is relatively easy to compute for any given  $x$ ,
- $H(x)$  is one-way,
- $H(x)$  is collision-free.

A hash function  $H$  is said to be one-way if it is hard to invert, where “hard to invert” means that given a hash value  $h$ , it is computationally infeasible to find some input  $x$  such that  $H(x) = h$ .

If, given a message  $x$ , it is computationally infeasible to find a message  $y$  not equal to  $x$  such that  $H(x) = H(y)$  then  $H$  is said to be a weakly collision-free hash function.

A strongly collision-free hash function  $H$  is one for which it is computationally infeasible to find any two messages  $x$  and  $y$  such that  $H(x) = H(y)$ .

Let  $H: M \Rightarrow \{0, 1\}^n$  be a hash function ( $|M| \gg 2^n$ )

Following is a generic algorithm to find a collision in time  $O(2^{n/2})$  hashes.

#### Algorithm:

1. Choose  $2^{n/2}$  random messages in  $M$ :  $m_1, m_2, \dots, m_{n/2}$
2. For  $i = 1, 2, \dots, 2^{n/2}$  compute  $t_i = H(m_i) \Rightarrow \{0, 1\}^n$
3. Look for a collision ( $t_i = t_j$ ). If not found, go back to step 1

We consider the following experiment. From a set of  $H$  values, we choose  $n$  values uniformly at random thereby allowing repetitions. Let  $p(n; H)$  be the probability that during this experiment at least one value is chosen more than once. This probability can be approximated as:

$$p(n; H) = 1 - ((365-1)/365) * ((365-2)/365) * \dots * ((365-n+1)/365)$$

$$p(n; H) = e^{-n(n-1)/(2H)} = e^{-n^2/(2H)}$$

### Digital signature susceptibility –

Digital signatures can be susceptible to birthday attacks. A message  $m$  is typically signed by first computing  $H(m)$ , where  $H$  is a cryptographic hash function, and then using some secret key to sign  $H(m)$ . Suppose Alice wants to trick Bob into signing a fraudulent contract. Alice prepares a fair contract  $m$  and fraudulent one  $m'$ . She then finds a number of positions where  $m$  can be changed without changing the meaning, such as inserting commas, empty lines, one versus two spaces after a sentence, replacing synonyms, etc. By combining these changes she can create a huge number of variations on  $m$  which are all fair contracts.

Similarly, Alice can also make some of these changes on  $m'$  to take it, even more, closer towards  $m$ , that is  $H(m) = H(m')$ . Hence, Alice can now present the fair version  $m$  to Bob for signing. After Bob has signed, Alice takes the signature and attaches to it the fraudulent contract. This signature proves that Bob has signed the fraudulent contract.

To avoid such an attack the output of the hash function should be a very long sequence of bits such that the birthday attack now becomes computationally infeasible.