U23CS405 COMPUTER NETWORKS

UNIT I INTRODUCTION AND APPLICATION LAYER **9**

Data Communication - Networks – Network Types – Protocol Layering – TCP/IP Protocol suite – OSI Model – Introduction to Sockets - Application Layer protocols: HTTP – FTP – Email protocols (SMTP - POP3 - IMAP - MIME) – DNS – SNMP

UNIT II TRANSPORT LAYER 9

Introduction - Transport-Layer Protocols: UDP – TCP: Connection Management – Flow control - Congestion Control - Congestion avoidance (DECbit, RED) – SCTP – Quality of Service

UNIT III NETWORK LAYER **9**

Switching: Packet Switching - Internet protocol - IPV4 – IP Addressing – Subnetting - IPV6, ARP, RARP, ICMP, DHCP

UNIT IV ROUTING **9**

Routing and protocols: Unicast routing - Distance Vector Routing - RIP - Link State Routing – OSPF – Path-vector routing - BGP - Multicast Routing: DVMRP – PIM.

UNIT V DATA LINK AND PHYSICAL LAYERS **9**

Data Link Layer – Framing – Flow control – Error control – Data-Link Layer Protocols – HDLC – PPP - Media Access Control – Ethernet Basics – CSMA/CD – Virtual LAN – Wireless LAN (802.11) - Physical Layer: Data and Signals - Performance – Transmission media- Switching – Circuit Switching.

UNIT I - INTRODUCTION AND PHYSICAL LAYER

Data Communication - Networks – Network Types – Protocol Layering – TCP/IP Protocol suite –OSI Model – Introduction to Sockets - Application Layer protocols: HTTP – FTP – Email protocols (SMTP - POP3 - IMAP - MIME) – DNS – SNMP

INTRODUCTION TO NETWORKS

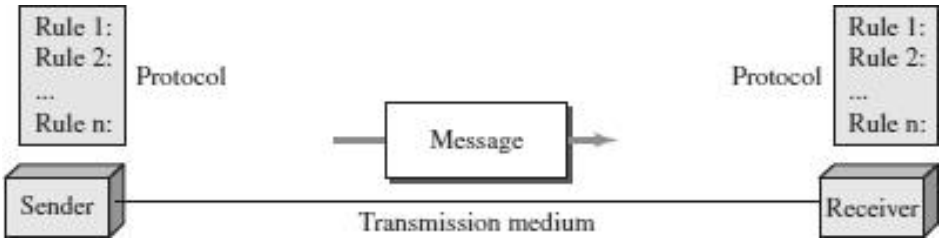
- A network is a set of devices (often referred to as nodes) connected by communication links.
- A node can be a computer, printer, or any other device capable of sending or receiving data generated by other nodes on the network.
- When we communicate, we are sharing information. This sharing can be local or remote.

CHARACTERISTICS OF A NETWORK

The effectiveness of a network depends on three characteristics.

1. ***Delivery***: The system must deliver data to the correct destination.
2. ***Accuracy***: The system must deliver data accurately.
3. ***Timeliness***: The system must deliver data in a timely manner.

COMPONENTS INVOLVED IN A NETWORK PROCESS



The five components are:

1. **Message** - It is the information to be communicated. Popular forms of information include text, pictures, audio, video etc.
2. **Sender** - It is the device which sends the data messages. It can be a computer, workstation, telephone handset etc.
3. **Receiver** - It is the device which receives the data messages. It can be a computer, workstation, telephone handset etc.
4. **Transmission Medium** - It is the physical path by which a message travels from sender to receiver. Some examples include twisted-pair wire, coaxial cable, radio waves etc.
5. **Protocol** - It is a set of rules that governs the data communications. It represents an agreement between the communicating devices. Without a protocol, two devices may be connected but not communicating.

KEY ELEMENTS OF PROTOCOL

- ***Syntax***: Refers to the structure or format of the data, meaning the order in whichthey are presented.
- ***Semantics***: Refers to the meaning of each section of bits.
- ***Timing***: Refers to two characteristics. (1). When data should be sent and (2). Howfast they can be sent.

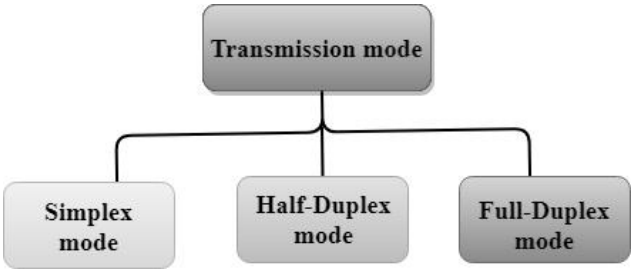
TRANSMISSION MODES

- The way in which data is transmitted from one device to another device is knownas **transmission mode**.
- The transmission mode is also known as the communication mode.
- Each communication channel has a direction associated with it, and transmissionmedia provide the direction. Therefore, the transmission mode is also known as adirectional mode.
- The transmission mode is defined in the physical layer.

Types of Transmission mode

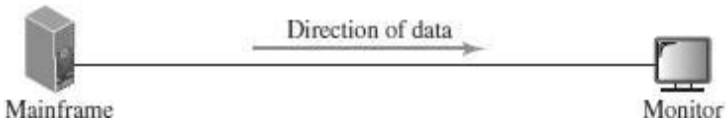
The Transmission mode is divided into three categories:

- Simplex Mode
- Half-duplex Mode
- Full-duplex mode (Duplex Mode)



SIMPLEX MODE

- In Simplex mode, the communication is unidirectional, i.e., the data flow in one direction.
- A device can only send the data but cannot receive it or it can receive the data butcannot send the data.
- This transmission mode is not very popular as mainly communications require thetwo-way exchange of data. The simplex mode is used in the business field as in sales that do not require any corresponding reply.
- The radio station is a simplex channel as it transmits the signal to the listeners butnever allows them to transmit back.
- **Keyboard and Monitor** are the examples of the simplex mode as a keyboard canonly accept the data from the user and monitor can only be used to display the data on the screen.
- The main advantage of the simplex mode is that the full capacity of the communication channel can be utilized during transmission.



Advantage of Simplex mode:

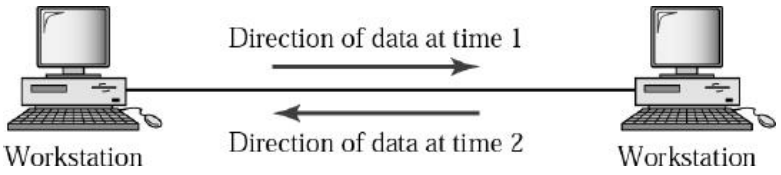
- In simplex mode, the station can utilize the entire bandwidth of the communication channel, so that more data can be transmitted at a time.

Disadvantage of Simplex mode:

- Communication is unidirectional, so it has no inter-communication between devices.

HALF-DUPLEX MODE

- In a Half-duplex channel, direction can be reversed, i.e., the station can transmitand receive the data as well.
- Messages flow in both the directions, but not at the same time.
- The entire bandwidth of the communication channel is utilized in one direction ata time.
- In half-duplex mode, it is possible to perform the error detection, and if any erroroccurs, then the receiver requests the sender to retransmit the data.
- A **Walkie-talkie** is an example of the Half-duplex mode.
- In Walkie-talkie, one party speaks, and another party listens. After a pause, the other speaks and first party listens. Speaking simultaneously will create the distorted sound which cannot be understood.



Advantage of Half-duplex mode:

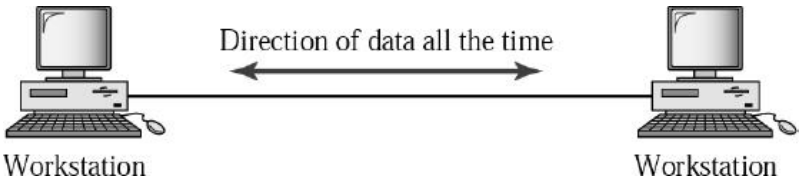
- In half-duplex mode, both the devices can send and receive the data and also canutilize the entire bandwidth of the communication channel during the transmission of data.

Disadvantage of Half-Duplex mode:

- In half-duplex mode, when one device is sending the data, then another has towait, this causes the delay in sending the data at the right time.

FULL-DUPLEX MODE

- In Full duplex mode, the communication is bi-directional, i.e., the data flow inboth the directions.
- Both the stations can send and receive the message simultaneously.
- Full-duplex mode has two simplex channels. One channel has traffic moving inone direction, and another channel has traffic flowing in the opposite direction.
- The Full-duplex mode is the fastest mode of communication between devices.
- The most common example of the full-duplex mode is a T**elephone network**. When two people are communicating with each other by a telephone line, bothcan talk and listen at the same time.



Advantage of Full-duplex mode:

- Both the stations can send and receive the data at the same time.

Disadvantage of Full-duplex mode:

- If there is no dedicated path exists between the devices, then the capacity of the communication channel is divided into two parts.

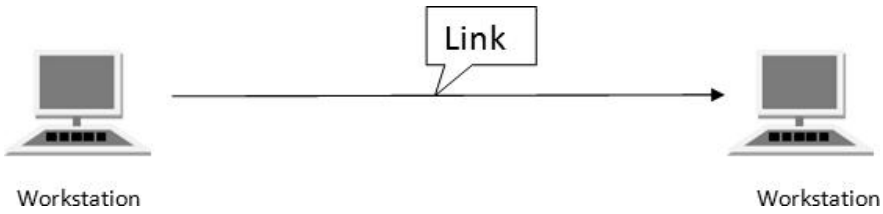
COMPARISON - SIMPLEX, HALF-DUPLEX AND FULL-DUPLEX MODE

BASIS FOR COMPARISON	SIMPLEX MODE	HALF-DUPLEX MODE	FULL-DUPLEX MODE
Direction of communication	Communication is unidirectional.	Communication is bidirectional, but one at a time.	Communication is bidirectional.
Send/Receive	A device can only send the data but cannot receive it orit can only receive the data but cannotsend it.	Both the devices can send and receive the data, but one at a time.	Both the devices can send and receive the data simultaneously.
Example	Radio, Keyboard, and monitor.	Walkie-Talkie	Telephone network.

LINE CONFIGURATION / LINE CONNECTIVITY

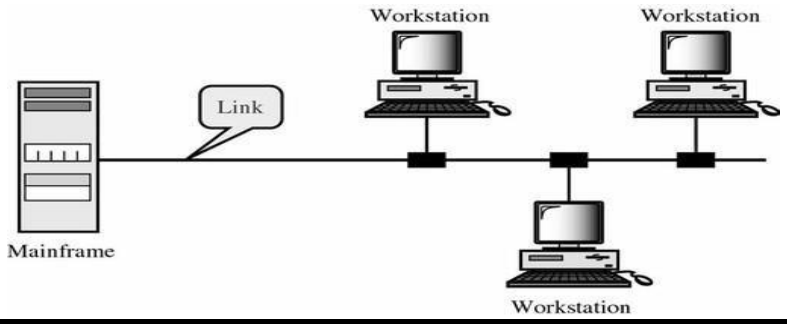
Line configuration refers to the way two or more communication devices attach to a link. A link is a communications pathway that transfers data from one device to another. There are two possible line configurations:

i. ***Point to Point*** **(PPP):** Provides a dedicated Communication link between two devices. It is simple to establish. The most common example for Point-to-Point connection is a computer connected by telephone line. We can connect the two devices by means of a pair of wires or using a microwave or satellite link.



ii. ***MultiPoint:*** It is also called ***Multidrop*** configuration. In this connection two ormore devices share a single link. There are two kinds of Multipoint Connections.

- **Spatial Sharing**: If several devices can share the link simultaneously,it is called Spatially shared line configuration
- **Temporal (Time) Sharing**: If users must take turns using the link, then it's called Temporally shared or Time Shared Line Configuration.

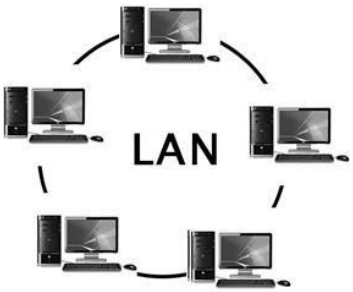


NETWORK TYPES

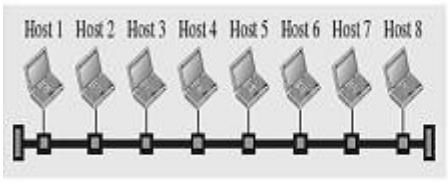
- A computer network is a group of computers linked to each other that enables the computer to communicate with another computer and share their resources, data, and applications.
- A computer network can be categorized by their size.
- A computer network is mainly of three types:
    1. Local Area Network (LAN)
    2. Wide Area Network (WAN)
    3. Metropolitan Area Network (MAN)

LOCAL AREA NETWORK (LAN)

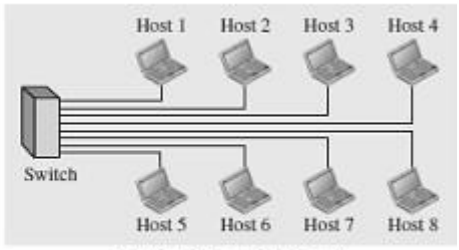
- Local Area Network is a group of computers connected to each other in a small area such as building, office.
- LAN is used for connecting two or more personal computers through a communication medium such as twisted pair, coaxial cable, etc.



- 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.
- LAN can be connected using a common cable or a Switch.



a. LAN with a common cable (past)

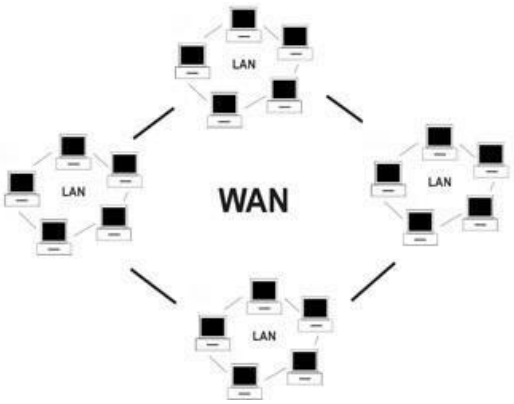


b. LAN with a switch (today)

Advantages of LAN	***Disadvantages of LAN***
• Resource Sharing • Software Applications Sharing. • Easy and Cheap Communication • Centralized Data. • Data Security • Internet Sharing	• High Setup Cost • Privacy Violations • Data Security Threat • LAN Maintenance Job • Covers Limited Area

WIDE AREA NETWORK (WAN)

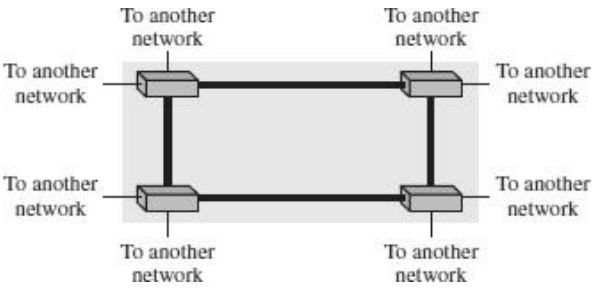
- A Wide Area Network is a network that extends over a large geographical areasuch as states or countries.
- A Wide Area Network is quite bigger network than the LAN.



- A Wide Area Network is not limited to a single location, but it spans over a large geographical area through a telephone line, fiber
- 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.
- WAN can be either a point-to-point WAN or Switched WAN.

Point-to-point WAN

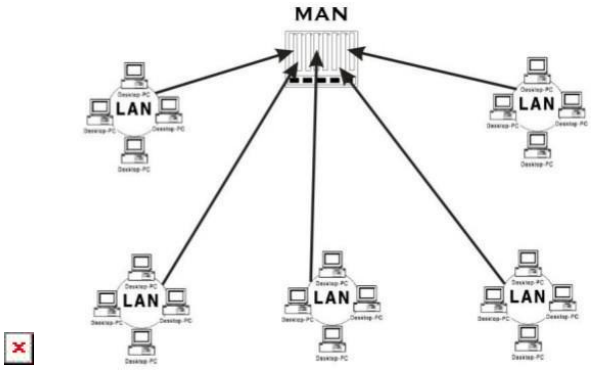
Switched WAN



***Advantages of Wide Area Network*:**	***Disadvantages of Wide Area Network*:**
○ Large Geographical area ○ Centralized data ○ Exchange messages ○ Sharing of software and resources ○ High bandwidth	○ Security issue ○ Needs Firewall & antivirus software ○ High Setup cost ○ Troubleshooting problems

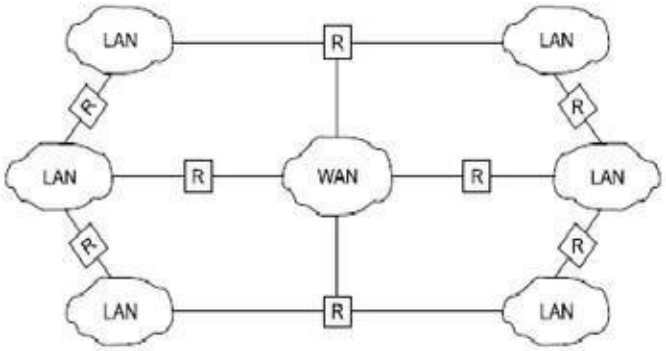
METROPOLITAN AREA NETWORK (MAN)

- A metropolitan area network is a network that covers a larger geographic area by interconnecting a different LAN to form a larger network.
- It generally covers towns and cities (50 km)
- In MAN, various LANs are connected to each other through a telephoneexchange line.
- Communication medium used for MAN are optical fibers, cables etc.
- It has a higher range than Local Area Network (LAN). It is adequate for distributed computing applications.



INTERNETWORK

- An internetwork is defined as two or more computer network LANs or WAN.
- An Internetwork can be formed by joining two or more individual networks bymeans of various devices such as routers, gateways and bridges.
- An interconnection between public, private, commercial, industrial, or government computer networks can also be defined as **internetworking**.

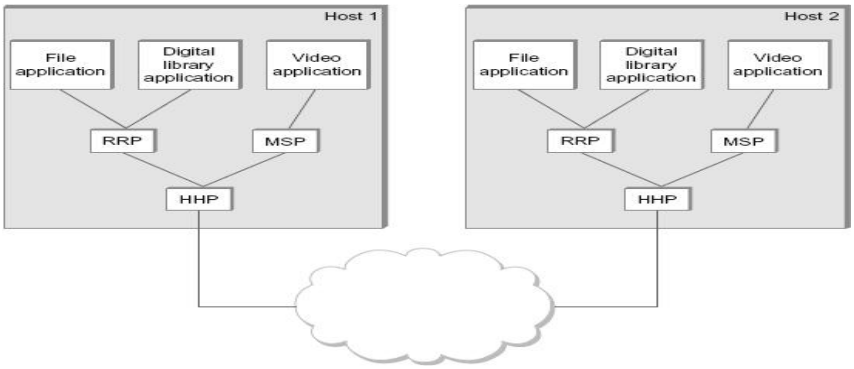


Types of Internetworks

Extranet	***Intranet***
An extranet is used for information sharing. The access to the extranet is restricted to only those users who have login credentials. An extranet is the lowest level of internetworking. It can becategorized as **MAN**, **WAN** or other computer networks. An extranet cannot have a single **LAN**, at least it must have one connection to the **external network**.	An intranet belongs to an organization which is only accessible by the **organization's employee** or members. The main aim of the intranet is to share the information and resources among the organization employees. An intranet provides the facility to work in groups and for teleconferences.

PROTOCOL LAYERING

- In networking, a protocol **defines the rules** that both the sender and receiver andall intermediate devices need to follow to be able **to communicate effectively**.
- A protocol provides a communication service that the process uses to exchange messages.
- When communication is simple, we may need only one simple protocol.



- When the communication is complex, we may need to divide the task between different layers, in which case we need a protocol at each layer, or **protocol layering.**
- Protocol layering is that it allows us to separate the services from the implementation.
- A layer needs to be able to receive a set of services from the lower layer and to give the services to the upper layer.
- Any modification in one layer will not affect the other layers.

Basic Elements of Layered Architecture

- **Service:** It is a set of actions that a layer provides to the higher layer.
- **Protocol:** It defines a set of rules that a layer uses to exchange the information with peer entity. These rules mainly concern about both the contents and order of the messages used.
- **Interface:** It is a way through which the message is transferred from one layer to another layer.

Features of Protocol Layering

1. **It decomposes the problem of building a network** into more manageable components.
2. **It provides a more modular design**.

Principles of Protocol Layering

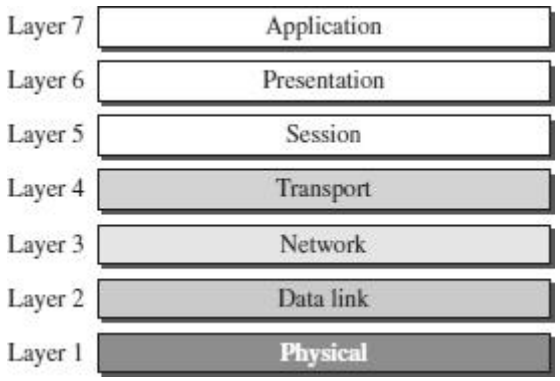
1. The first principle dictates that if we want bidirectional communication, we need to make each layer so that it is able to perform two opposite tasks, one in each direction.
2. The second principle that we need to follow in protocol layering is that the two objects under each layer at both sites should be identical.

Protocol Graph

- The set of protocols that make up a network system is called a **protocol graph**.
- The nodes of the graph correspond to protocols, and the edges represent a dependence relation.
- For example, the Figure below illustrates a protocol graph consists of protocols ***RRP (Request/Reply Protocol)*** and ***MSP (Message Stream Protocol)*** implement two different types of process-to-process channels, and both depend on the ***HHP (Host-to- Host Protocol)*** which provides a host-to-host connectivity service

OSI MODEL

- OSI stands for **Open System Interconnection.**
- It is a reference model that describes how information from a software application in one computer moves through a physical medium to the software application in another computer.
- OSI consists of seven layers, and each layer performs a particular network function.
- OSI model was developed by the International Organization for Standardization (ISO) in 1984, and it is now considered as an architectural model for the inter- computer communications.
- OSI model divides the whole task into seven smaller and manageable tasks. Each layer is assigned a particular task.
- Each layer is self-contained, so that task assigned to each layer can be performed independently.



ORGANIZATION OF THE OSI LAYERS

The OSI model is divided into two layers: **upper layers and lower layers**.

UPPER LAYERS

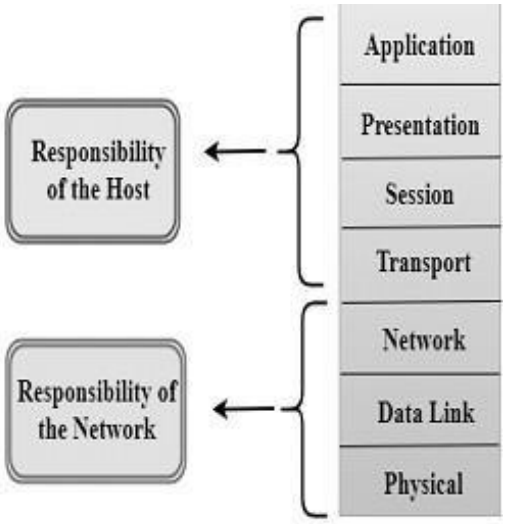
(Responsibility of the Host)

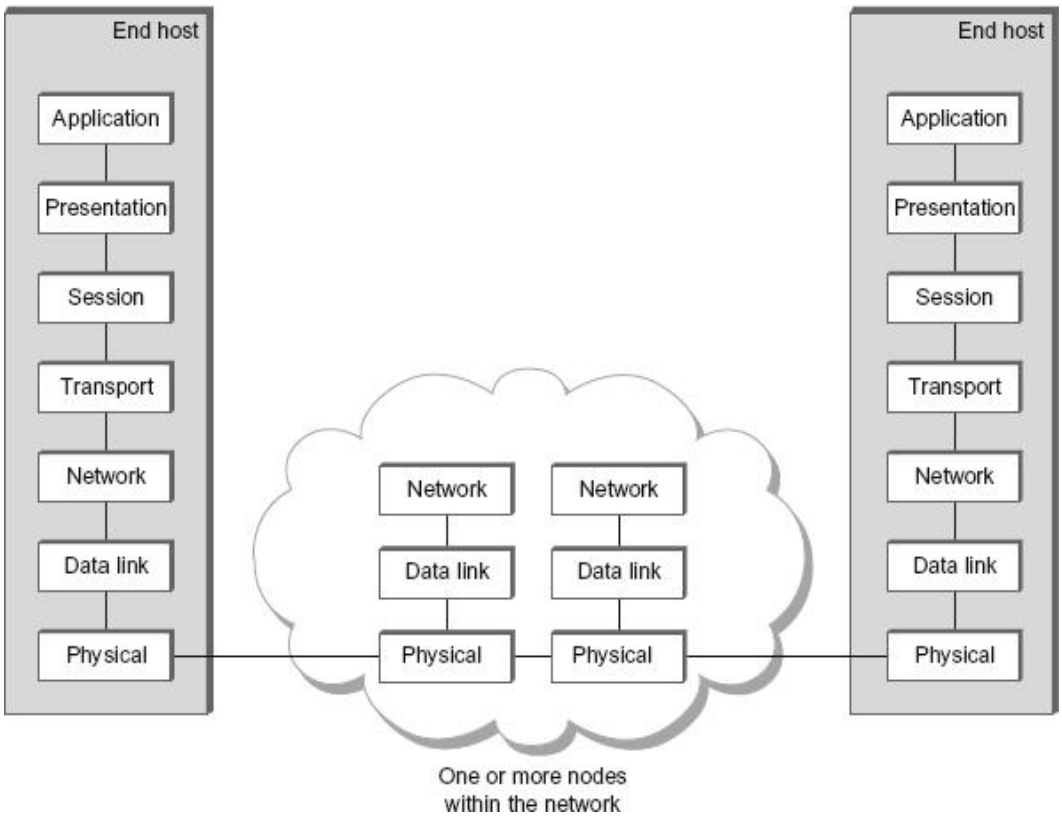
The upper layers of the OSI model mainly deals with the application related issues.
They are implemented only in the software.

LOWER LAYERS

(Responsibility of the Network)

The lower layers of the OSI model deals with the data transport issues.
They are implemented in hardware and software.





FUNCTIONS OF THE OSI LAYERS

1. PHYSICAL LAYER

The physical layer coordinates the functions required to **transmit a bit stream over a physical medium.**
The physical layer is concerned with the following functions:

- **Physical characteristics of interfaces and media -** The physical layer definesthe characteristics of the interface between the devices and the transmissionmedium.
- **Representation of bits -** To transmit the stream of bits, it must be encoded to signals. The physical layer defines the type of encoding.
- **Signals:** It determines the type of the signal used for transmitting the information.
- **Data Rate or Transmission rate -** The number of bits sent each second –is also defined by the physical layer.
- **Synchronization of bits -** The sender and receiver must be synchronized at thebit level. Their clocks must be synchronized.
- **Line Configuration -** In a point-to-point configuration, two devices are connected together through a dedicated link. In a multipoint configuration, a link is shared between several devices.
- **Physical Topology -** The physical topology defines how devices are connected tomake a network. Devices can be connected using a mesh, bus, star or ring topology.
- **Transmission Mode -** The physical layer also defines the direction oftransmission between two devices: simplex, half-duplex or full-duplex.

2. DATA LINK LAYER

It is responsible for **transmitting frames from one node to the next node**.The other responsibilities of this layer are

- **Framing -** Divides the stream of bits received into data units called frames.
- **Physical addressing** – If frames are to be distributed to different systems on the network, data link layer adds a header to the frame to define the sender and receiver.
- **Flow control**- If the rate at which the data are absorbed by the receiver is lessthan the rate produced in the sender,the Data link layer imposes a flow ctrl mechanism.
- **Error control**- Used for detecting and retransmitting damaged or lost frames and to prevent duplication of frames. This is achieved through a trailer added at the end of the frame.
- **Medium Access control** -Used to determine which device has control over the link at any given time.

3. NETWORK LAYER

This layer is responsible for the **delivery of packets from source to destination**.
It determines the best path to move data from source to the destination based on the network conditions, the priority of service, and other factors.
The other responsibilities of this layer are

- **Logical addressing** - If a packet passes the network boundary, we need another addressing system for source and destination called logical address. This addressing is used to identify the device on the internet.
- **Routing** – Routing is the major component of the network layer, and it determines the best optimal path out of the multiple paths from source to the destination.

4. TRANSPORT LAYER

It is responsible for **Process-to-Process** delivery. That is responsible for source-to- destination (end-to-end) delivery of the entire message, It also ensures whether the message arrives in order or not.
The other responsibilities of this layer are

- **Port addressing / Service Point addressing** - The header includes an address called port address / service point address. This layer gets the entire message to the correct process on that computer.
- **Segmentation and reassembly** - The message is divided into segments and each segment is assigned a sequence number. These numbers are arranged correctly onthe arrival side by this layer.

- **Connection control** - This can either be **connectionless or connection oriented.**
  - The connectionless treats each segment as an individual packet anddelivers to the destination.
  - The connection-oriented makes connection on the destination side beforethe delivery. After the delivery the termination will be terminated.
- **Flow control** - The transport layer also responsible for flow control but it is performed end-to-end rather than across a single link.
- **Error Control -** Error control is performed end-to-end rather than across thesingle link.

5. SESSION LAYER

This layer **establishes, manages and terminates connections between applications**.The other responsibilities of this layer are

- **Dialog control** - Session layer acts as a dialog controller that creates a dialog between two processes or we can say that it allows the communication between two processes which can be either half-duplex or full-duplex.
- **Synchronization**- Session layer adds some checkpoints when transmitting the data in a sequence. If some error occurs in the middle of the transmission of data, then the transmission will take place again from the checkpoint. This process is known as Synchronization and recovery.

6. PRESENTATION LAYER

It is concerned with the **syntax and semantics of information exchanged between twosystems.**
The other responsibilities of this layer are

- **Translation** – Different computers use different encoding system, this layer is responsible for interoperability between these different encoding methods. It will change the message into some common format.
- **Encryption and decryption**-It means that sender transforms the original information to another form and sends the resulting message over the n/w. and vice versa.
- **Compression and expansion**-Compression reduces the number of bits contained in the information particularly in text, audio and video.

7. APPLICATION LAYER

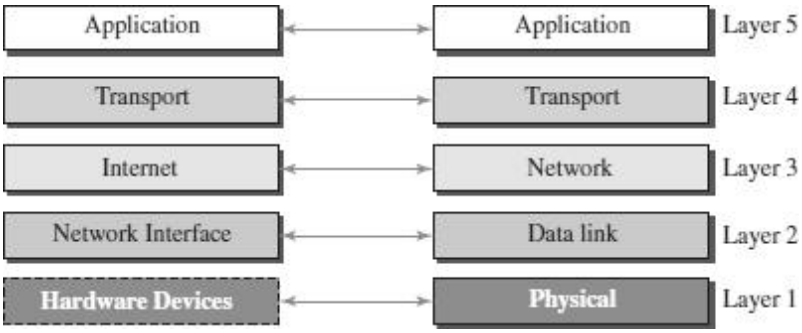
This layer **enables the user to access the network**. It handles issues such as networktransparency, resource allocation, etc. This allows the user to log on to remote user.
The other responsibilities of this layer are

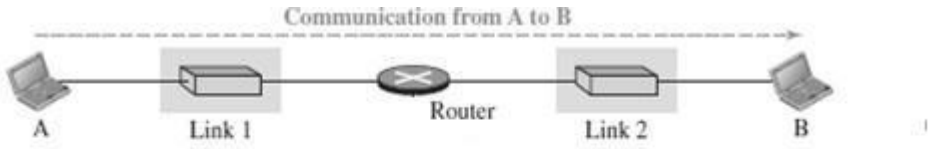
- **FTAM (File Transfer, Access, Management)** - Allows user to access files in aremote host.

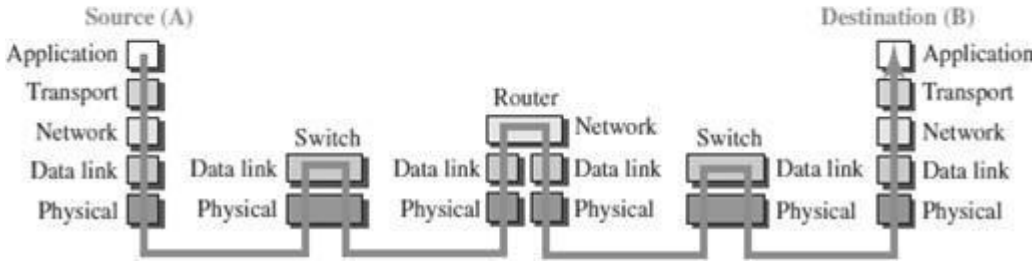
- **Mail services** - Provides email forwarding and storage.
- **Directory services** - Provides database sources to access information aboutvarious sources and objects.

TCP / IP PROTOCOL SUITE

- The TCP/IP architecture is also called as Internet architecture.
- It is developed by the US Defense Advanced Research Project Agency (**DARPA**)for its packet switched network (**ARPANET**).
- TCP/IP is a protocol suite used in the Internet today.
- It is a 4-layer model. The layers of TCP/IP are
    1. **Application layer**
    2. **Transport Layer (TCP/UDP)**
    3. **Internet Layer**
    4. **Network Interface Layer**







APPLICATION LAYER

- An application layer incorporates the function of top three OSI layers. An application layer is the topmost layer in the TCP/IP model.
- It is responsible for handling high-level protocols, issues of representation.
- This layer allows the user to interact with the application.
- When one application layer protocol wants to communicate with anotherapplication layer, it forwards its data to the transport layer.
- Protocols such as FTP, HTTP, SMTP, POP3, etc running in the application layer provides service to other program running on top of application layer

TRANSPORT LAYER

- The transport layer is responsible for the reliability, flow control, and correctionof data which is being sent over the network.
- The two protocols used in the transport layer are **User Datagram protocol and Transmission control protocol**.
  - **UDP** – UDP provides connectionless service and end-to-end delivery of transmission. It is an unreliable protocol as it discovers the errors but notspecify the error.
  - **TCP** – TCP provides a full transport layer services to applications. TCP is a reliable protocol as it detects the error and retransmits the damaged frames.

INTERNET LAYER

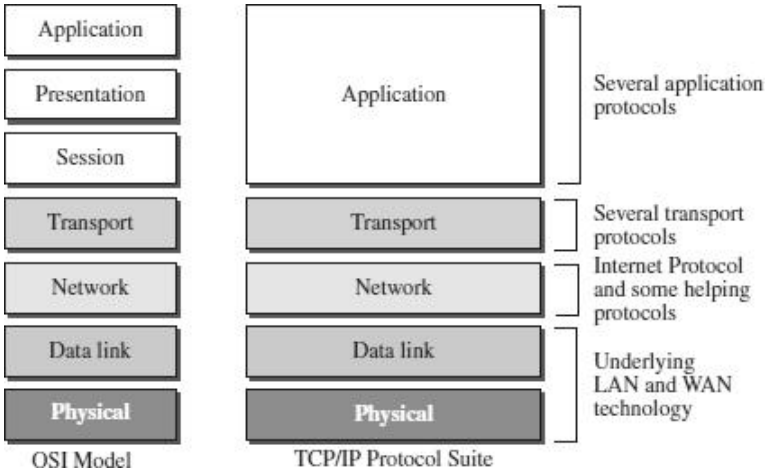
- The internet layer is the second layer of the TCP/IP model.
- An internet layer is also known as the network layer.

- The main responsibility of the internet layer is to send the packets from any network, and they arrive at the destination irrespective of the route they take.
- Internet layer handle the transfer of information across multiple networks throughrouter and gateway .
- IP protocol is used in this layer, and it is the most significant part of the entire TCP/IP suite.

NETWORK INTERFACE LAYER

- The network interface layer is the lowest layer of the TCP/IP model.
- This layer is the combination of the Physical layer and Data Link layer defined inthe OSI reference model.
- It defines how the data should be sent physically through the network.
- This layer is mainly responsible for the transmission of the data between two devices on the same network.
- The functions carried out by this layer are encapsulating the IP datagram into frames transmitted by the network and mapping of IP addresses into physical addresses.
- The protocols used by this layer are Ethernet, token ring, FDDI, X.25, frame relay.

COMPARISON - OSI MODEL AND TCP/IP MODEL



S.No	OSI MODEL	TCP/IP MODEL
1	Defined before advent of internet	Defined after the advent of Internet.
2	Service interface and protocolsare clearly distinguished before	Service interface and protocols were not clearly distinguished before
3	Internetworking not supported	TCP/IP supports Internet working
4	Strict layering	Loosely layered
5	Protocol independent standard	Protocol Dependant standard
6	Less Credible	More Credible
7	All packets are reliably delivered	TCP reliably delivers packets, IP doesnot reliably deliver packets

SOCKETS

A **socket** is one endpoint of a **two-way** communication link between two programs running on the network. The socket mechanism provides a means of inter-process communication (IPC) by establishing named contact points between which the communication take place.

Like 'Pipe' is used to create pipes and sockets is created using **'socket'** system call. The socket provides bidirectional **FIFO** Communication facility over the network. A socket connecting to the network is created at each end of the communication. Each socket has a specific address. This address is composed of an IP address and a port number.

Socket are generally employed in client server applications. The server creates a socket, attaches it to a network port address then waits for the client to contact it. The client creates a socket and then attempts to connect to the server socket. When the connection is established, transfer of data takes place.

Types of Sockets: There are two types of Sockets: the **datagram** socket and the **stream** socket.

1. **Datagram Socket:** This is a type of network which has connection less point for sending and receiving packets. It is similar to mailbox. The letters (data) posted into the box are collected and delivered (transmitted) to a letterbox (receiving socket).
2. **Stream Socket:** In Computer operating system, a stream socket is type of interprocess communications socket or network socket which provides a connection-oriented, sequenced, and unique flow of data without record boundaries with well-defined mechanisms for creating and destroying connections and for detecting errors. It is similar to phone. A connection is established between the phones (two ends) and a conversation (transfer of data) takes place.

HTTP (HYPERTEXT TRANSFER PROTOCOL)

- The HyperText Transfer Protocol (HTTP) is used to define how the client-server

programs can be written to retrieve web pages from the Web.

- 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.
- HTTP is a *stateless* request/response protocol that governs client/server communication.
- An HTTP client sends a request; an HTTP server returns a response.
- The server uses the port number 80; the client uses a temporary port number.
- HTTP uses the services of TCP, a connection-oriented and reliable protocol.
- HTTP is a text-oriented protocol. It contains *embedded* URL known as links.

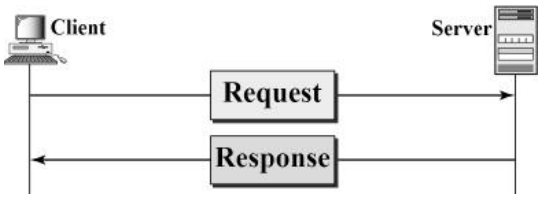
- When hypertext is clicked, browser opens a new connection, retrieves file fromthe server and displays the file.
- Each HTTP message has the general form

START_LINE <CRLF>
MESSAGE_HEADER <CRLF>
<CRLF> MESSAGE_BODY <CRLF>

where <CRLF> stands for carriage-return-line-feed.

HTTP REQUEST AND RESPONSE MESSAGES

- The HTTP protocol defines the format of the request and response messages.



- **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.

HTTP REQUEST MESSAGE

Request Line
Request Header : Value
Body (optional)

- The first line in a request message is called a *request line*.
- After the request line, we can have zero or more *request header* lines.
- The *body* is an optional one. It contains the comment to be sent or the file to be published on the website when the method is PUT or POST.

Request Line

- There are three fields in this request line - M*ethod, URL* and V*ersion*.
- The Method field defines the request types.

The URL field defines the address and name of the corresponding web page.
The Version field gives the version of the protocol; the most current version of HTTP is 1.1.

- Some of the Method types are

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

Request Header

- Each request header line sends additional information from the client to the server.
- Each header line has a header name, a colon, a space, and a header value.
- The value field defines the values associated with each header name.
- Headers defined for request message include

Header	*Description*
User-agent	Identifies the client program
Accept	Shows the media format the client can accept
Accept-charset	Shows the character set the client can handle
Accept-encoding	Shows the encoding scheme the client can handle
Accept-language	Shows the language the client can accept
Authorization	Shows what permissions the client has
Host	Shows the host and port number of the client
Date	Shows the current date
Upgrade	Specifies the preferred communication protocol
Cookie	Returns the cookie to the server
If-Modified-Since	If the file is modified since a specific date

Body

- The *body* can be present in a request message. It is optional.
- Usually, it contains the comment to be sent or the file to be published on the website when the method is PUT or POST.

Conditional Request

- A client can add a condition in its request.
- In this case, the server will send the requested web page if the condition is metor inform the client otherwise.
- One of the most common conditions imposed by the client is the time and datethe web page is modified.
- The client can send the header line *If-Modified-Since* with the request to tell theserver that it needs the page only if it is modified after a certain point in time.

HTTP RESPONSE MESSAGE

Status Line
Response Header : Value
Body

- The first line in a request message is called a status *line*.
- After the request line, we can have zero or more *response header* lines.
- The *body* is an optional one. The body is present unless the response is an error message

Status Line

- The Status line contains three fields - *HTTP version, Status code, Statusphrase*
- The first field defines the version of HTTP protocol, currently 1.1.
- The status code field defines the status of the request. It classifies the HTTPresult. It consists of three digits.

  1*xx*–Informational, 2*xx*– Success, 3*xx*–Redirection,
  4*xx*–Client error, 5*xx*–Server error
- The Status phrase field gives brief description about status code in text form.
- Some of the Status codes are

Code	Phrase	Description
100	Continue	Initial request received, client to continue process
200	OK	Request is successful
301	Moved permanently	Requested URL is no longer in use
404	Not found	Document not found
500	Internal server error	An error such as a crash, at the server site

Response Header

- Each header provides additional information to the client.
- Each header line has a header name, a colon, a space, and a header value.
- Some of the response headers are:

Response Header	Description
Content-type	specifies the MIME type
Expires	date and time up to which the document is valid
Last-modified	date and time when the document was last updated
Location	specifies location of the created or moved document

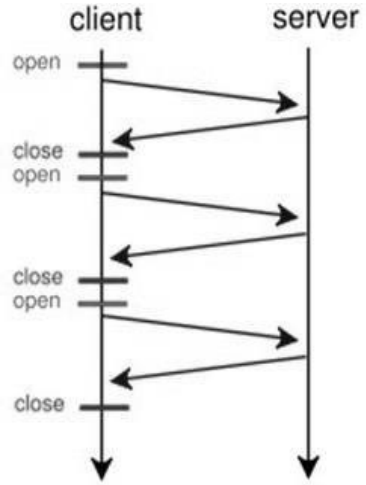
Body

- The body contains the document to be sent from the server to the client.
- The body is present unless the response is an error message.

HTTP CONNECTIONS

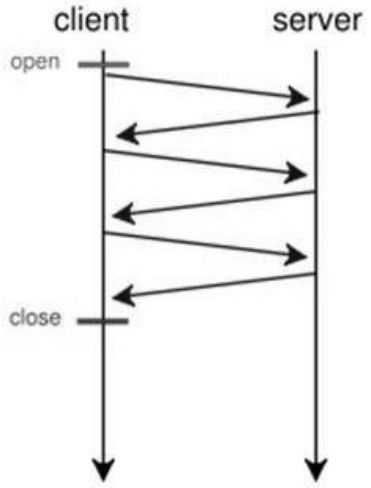
- HTTP Clients and Servers exchange multiple messages over the same TCP connection.
- If some of the objects are located on the same server, we have two choices: to retrieve each object using a new TCP connection or to make a TCP connection and retrieve them all.
- The first method is referred to as a *non-persistent connection,* the second as a *persistent connection*.
- HTTP 1.0 uses *non-persistent* connections and HTTP 1.1 uses *persistent* connections.

NON-PERSISTENT CONNECTIONS



- In a non-persistent connection, one TCP connection is made for each request/response.
- Only one object can be sent over a singleTCP connection
- The client opens a TCP connection and sends a request.
- The server sends the response and closesthe connection.
- The client reads the data until it encounters an end-of-file marker.
- It then closes the connection.

PERSISTENT CONNECTIONS



- HTTP version 1.1 specifies a persistent connectionby default.
- Multiple objects can be sent over a single TCP connection.
- In a persistent connection, the server leaves the connection open for more requests after sending a response.
- The server can close the connection at the request ofa client or if a time-out has been reached.
- Time and resources are saved using persistent connections. Only one set of buffers and variables needs to be set for the connection at each site.
- The round-trip time for connection establishment and connection termination is saved.

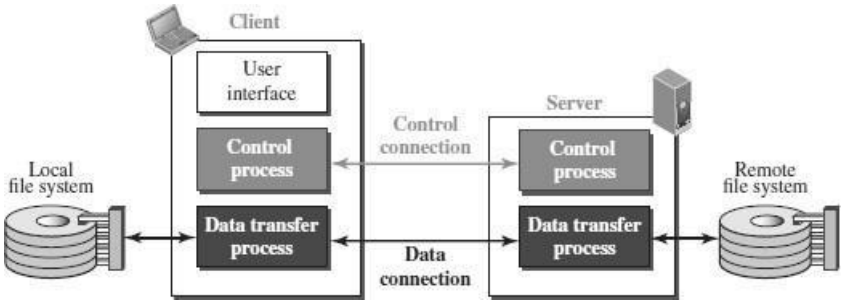
FTP (FILE TRANSFER PROTOCOL)

- FTP stands for File transfer protocol.
- FTP is a standard internet protocol provided by TCP/IP used fortransmitting the files from one host to another.
- It is mainly used for transferring the web page files from their creator to the computer that acts as a server for other computers on the internet.
- It is also used for downloading the files to computer from other servers.
- Although we can transfer files using HTTP, FTP is a better choice totransfer large files or to transfer files using different formats.

FTP OBJECTIVES

- It provides the sharing of files.
- It is used to encourage the use of remote computers.
- It transfers the data more reliably and efficiently.

FTP MECHANISM



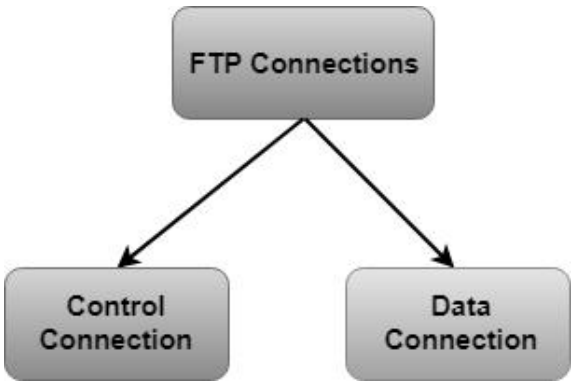
- The above figure shows the basic model of the FTP.
- The FTP client has three components:
  - user interface, control process, and data transfer process.
- The server has two components:
  - server control process and server data transfer process.

FTP CONNECTIONS

- There are two types of connections in FTP -

Control Connection and Data Connection.

- The two connections in FTP have different lifetimes.
- The control connection remains connected during the entire interactive FTP session.
- The data connection is opened and then closed for each file transfer activity.When a user starts an FTP session, the control connection opens.
- While the control connection is open, the data connection can be opened andclosed multiple times if several files are transferred.
- FTP uses two well-known TCP ports:
  - Port 21 is used for the control connection
  - Port 20 is used for the data connection.



- **Control Connection:**
  - The control connection uses very simple rules for communication.
  - Through control connection, we can transfer a line of command or lineof response at a time.
  - The control connection is made between the control processes.
  - The control connection remains connected during the entire interactiveFTP session.

- **Data Connection:**
  - The Data Connection uses very complex rules as data types may vary.
  - The data connection is made between data transfer processes.
  - The data connection opens when a command comes for transferring thefiles and closes when the file is transferred.

FTP COMMUNICATION

- FTP Communication is achieved through commands and responses.
- FTP Commands are sent from the client to the server
- FTP responses are sent from the server to the client.
- FTP Commands are in the form of ASCII uppercase, which may or may not be followed by an argument.
- Some of the most common commands are

Command	*Description*
ABOR	Abort the previous command
CDUP	Change to parent directory
CWD	Change to another directory
DELE	Delete a file
LIST	List subdirectories or files
MKD	Create a new directory
PASS	Password
PASV	Server chooses a port
PORT	Client chooses a port
PWD	Display name of current directory
QUIT	Log out of the system
RETR	Retrieve files; files are transferred from server to client
RMD	Delete a directory
RNFR	Identify a file to be renamed
RNTO	Rename the file
STOR	Store files; file(s) are transferred from client to server
STRU	Define data organization (**F**: file, **R**: record, or **P**: page)
TYPE	Default file type (**A**: ASCII, **E**: EBCDIC, **I**: image)
USER	User information
MODE	Define transmission mode (**S**: stream, **B**: block, or **C**: compressed

- Every FTP command generates at least one response.
- A response has two parts: a three-digit number followed by text.
- The numeric part defines the code; the text part defines needed parameter.

Code	*Description*	*Code*	*Description*
125	Data connection open	**250**	Request file action OK
150	File status OK	**331**	User name OK; password is needed
200	Command OK	**425**	Cannot open data connection
220	Service ready	**450**	File action not taken; file not available
221	Service closing	**452**	Action aborted; insufficient storage
225	Data connection open	**500**	Syntax error; unrecognized command
226	Closing data connection	**501**	Syntax error in parameters or arguments
230	User login OK	**530**	User not logged in

FTP FILE TYPE

- FTP can transfer one of the following file types across the data connection: ASCII file, EBCDIC file, or image file.

FTP DATA STRUCTURE

- FTP can transfer a file across the data connection using one of the following data structure : *file structure, record structure,* or *page structure*.
- The file structure format is the default one and has no structure. It is a continuous stream of bytes.
- In the record structure, the file is divided into *records*. This can be used only with text files.
- In the page structure, the file is divided into pages, with each page having a page number and a page header. The pages can be stored and accessed randomly or sequentially.

FTP TRANSMISSION MODE

- FTP can transfer a file across the data connection using one of the followingthree transmission modes: *stream mode, block mode,* or *compressed mode*.
- The stream mode is the default mode; data are delivered from FTP to TCP as a continuous stream of bytes.
- In the block mode, data can be delivered from FTP to TCP in blocks.
- In the compressed mode, data can be compressed and delivered from FTP toTCP.

FTP FILE TRANSFER

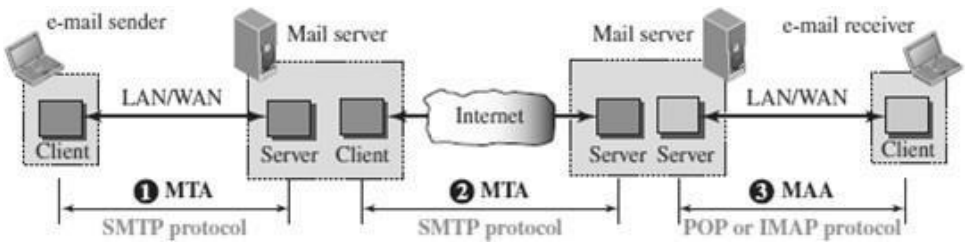
- File transfer occurs over the data connection under the control of thecommands sent over the control connection.
- File transfer in FTP means one of three things:
  - retrieving a file (server to client)
  - storing a file (client to server)
  - directory listing (server to client).

FTP SECURITY

- FTP requires a password, the password is sent in plaintext which isunencrypted. This means it can be intercepted and used by an attacker.
- The data transfer connection also transfers data in plaintext, which is insecure.
- To be secure, one can add a Secure Socket Layer between the FTP applicationlayer and the TCP layer.
- In this case FTP is called SSL-FTP.

EMAIL (SMTP, MIME, IMAP, POP3)

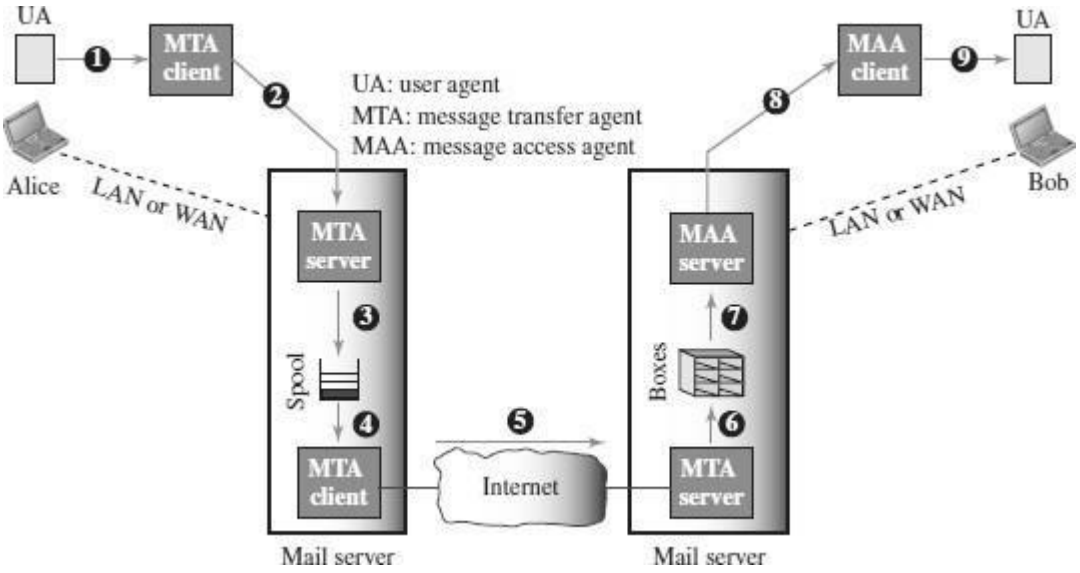
- One of the most popular Internet services is electronic mail (E-mail).
- Email is one of the oldest network applications.
- The **three main components of an Email** are
  1. User Agent (UA)
  2. Message Transfer Agent (MTA) – SMTP
  3. Message Access Agent (MAA) - IMAP ,POP



When the sender and the receiver of an e-mail are on the same system, we needonly two User Agents and no Message Transfer Agent

- When the sender and the receiver of an e-mail are on different system, we need two UA, two pairs of MTA (client and server), and two MAA (client and server).

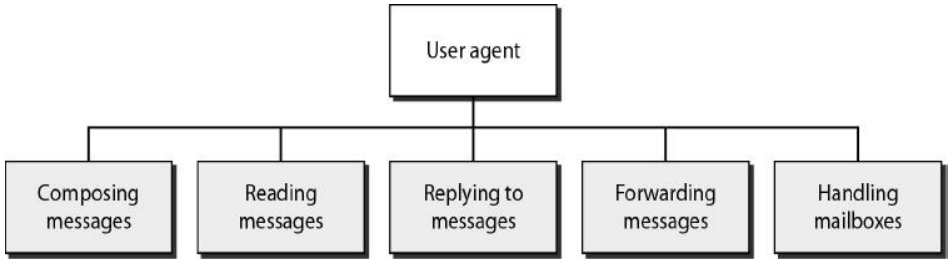
WORKING OF EMAIL



- When Alice needs to send a message to Bob, she runs a UA program to prepare the message and send it to her mail server.
- The mail server at her site uses a queue (spool) to store messages waiting to be sent. The message, however, needs to be sent through the Internet from Alice's site to Bob's site using an MTA.
- Here two message transfer agents are needed: one client and one server.
- The server needs to run all the time because it does not know when a client willask for a connection.
- The client can be triggered by the system when there is a message in the queueto be sent.
- The user agent at the Bob site allows Bob to read the received message.
- Bob later uses an MAA client to retrieve the message from an MAA server running on the second server.

USER AGENT (UA)

- The first component of an electronic mail system is the user agent (UA).
- It provides service to the user to make the process of sending and receiving a message easier.
- A user agent is a software package that composes, reads, replies to, andforwards messages. It also handles local mailboxes on the user computers.



There are two types of user agents: **Command-driven and GUI-based**.

Command driven

- Command driven user agents belong to the early days of electronic mail.
- A command-driven user agent normally accepts a one-character command fromthe

keyboard to perform its task.

- Some examples of command driven user agents are *mail, pine,* and *elm*.

GUI-based

- Modern user agents are GUI-based.
- They allow the user to interact with the software by using both the keyboardand the mouse.
- They have graphical components such as icons, menu bars, and windows thatmake the services easy to access.
- Some examples of GUI-based user agents are *Eudora* and *Outlook***.**

MESSAGE TRANSFER AGENT (MTA)

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

MESSAGE ACCESS AGENT (MAA)

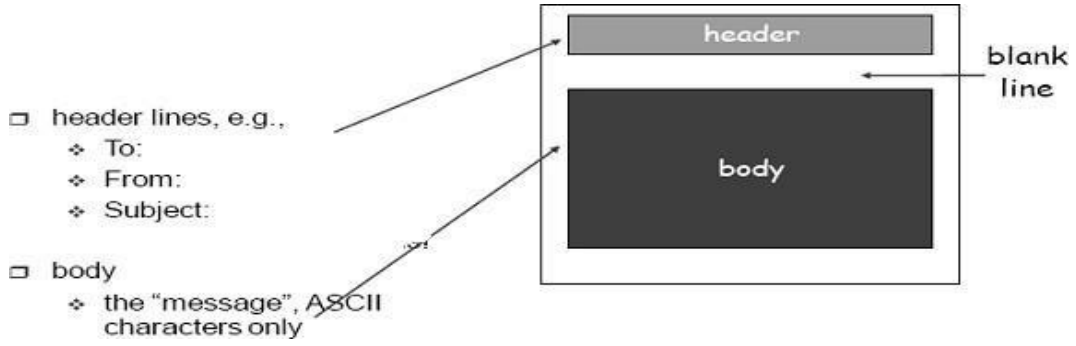
- MAA is a software that pulls messages out of a mailbox.
- POP3 and IMAP4 are examples of MAA.

ADDRESS FORMAT OF EMAIL

- E-mail address is *userid @ domain* where *domain* is hostname of the *mail server*.

MESSAGE FORMAT OF EMAIL

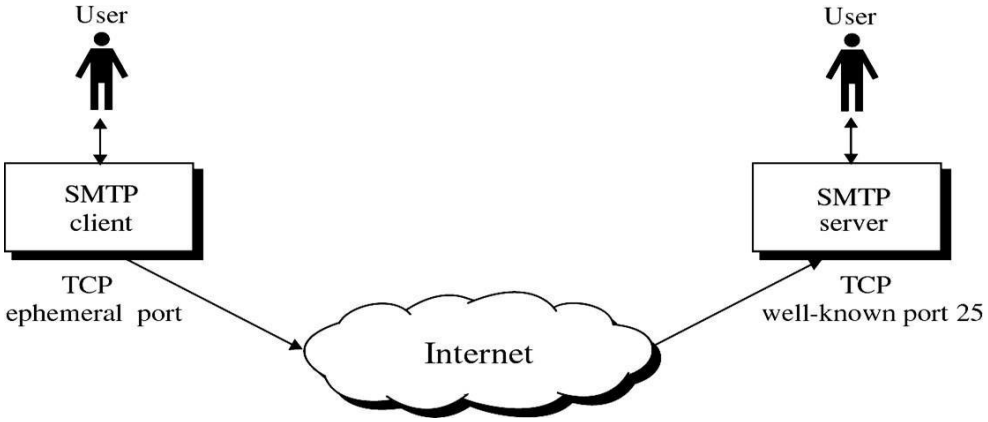
- Email message consists of two parts namely *header* and *body*.
- Each header line contains *type* and *value* separated by a colon (:).
- Some header contents are:
  - **From:** identifier sender of the message.
  - **To:** mail address of the recipient(s).
  - **Subject:** says about purpose of the message.
  - **Date:** timestamp of when the message was transmitted.
- Header is separated from the body by a *blank* line.Body contains the *actual* message.



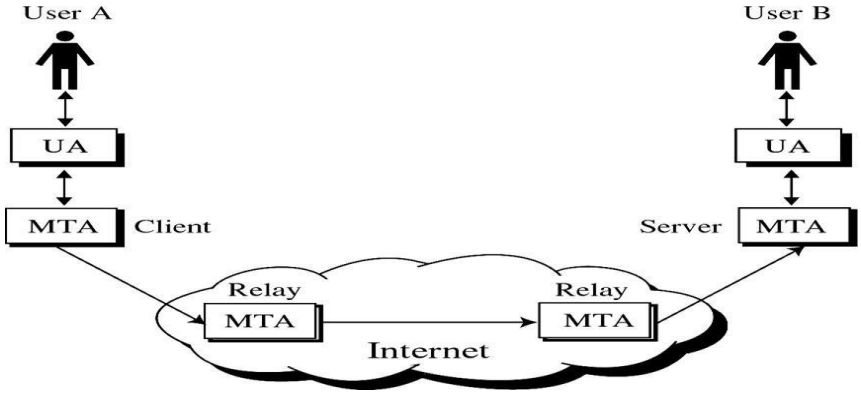
- Email was extended in 1993 to carry many different types of data: audio,video, images, Word documents, and so on.
- This extended version is known as **MIME**(Multipurpose Mail Extension).

SIMPLE MAIL TRANSFER PROTOCOL (SMTP)

- SMTP is the standard protocol for transferring mail between hosts in theTCP/IP protocol suite.
- SMTP is not concerned with the format or content of messages themselves.
- SMTP uses information written on the *envelope* of the mail (message header),but does not look at the *contents* (message body) of the envelope.

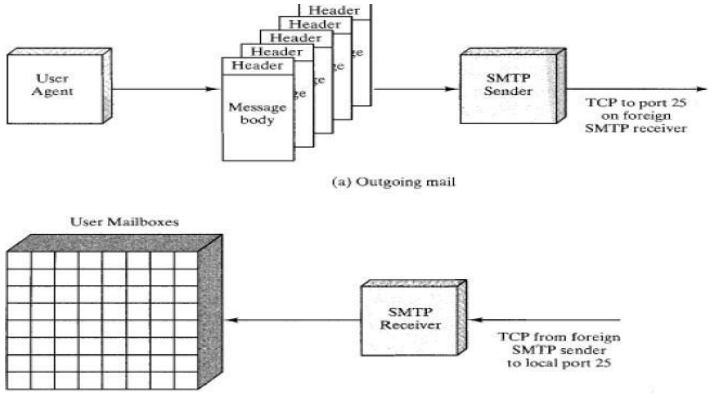


- SMTP clients and servers have two main components
  - **User Agents(UA)** – Prepares the message, encloses it in an envelope.
  - **Mail Transfer Agent (MTA)** – Transfers the mail across the internet
- SMTP also allows the use of Relays allowing other MTAs to relay the mail.



SMTP MAIL FLOW

-
-



To begin, mail is created by a user-agent program in response to user input.
Each created message consists of a header that includes the recipient's email address and other information, and a message body containing the message to be sent.

- These messages are then queued in some fashion and provided as input to an SMTP Sender program.

SMTP COMMANDS AND RESPONSES

- The operation of SMTP consists of a series of commands and responsesexchanged between the SMTP sender and SMTP receiver.
- The initiative is with the SMTP sender, who establishes the TCP connection.
- Once the connection is established, the SMTP sender sends commands over the connection to the receiver.
- The command is from an MTA client to an MTA server; the response is froman MTA server to the MTA client.

SMTP Commands

- Commands are sent from the client to the server. It consists of a keywordfollowed by zero or more arguments. SMTP defines 14 commands.

SMTP commands

Keyword	*Argument(s)*	*Description*
HELO	Sender's host name	Identifies itself
MAIL FROM	Sender of the message	Identifies the sender of the message
RCPT TO	Intended recipient	Identifies the recipient of the message
DATA	Body of the mail	Sends the actual message
QUIT		Terminates the message
RSET		Aborts the current mail transaction
VRFY	Name of recipient	Verifies the address of the recipient
NOOP		Checks the status of the recipient
TURN		Switches the sender and the recipient
EXPN	Mailing list	Asks the recipient to expand the mailing list
HELP	Command name	Asks the recipient to send information about the command sent as the argument
SEND FROM	Intended recipient	Specifies that the mail be delivered only to the terminal of the recipient, and not to the mailbox
SMOL FROM	Intended recipient	Specifies that the mail be delivered to the terminal *or* the mailbox of the recipient
SMAL FROM	Intended recipient	Specifies that the mail be delivered to the terminal *and* the mailbox of the recipient

SMTP Responses

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

SMTP OPERATIONS

SMTP Responses

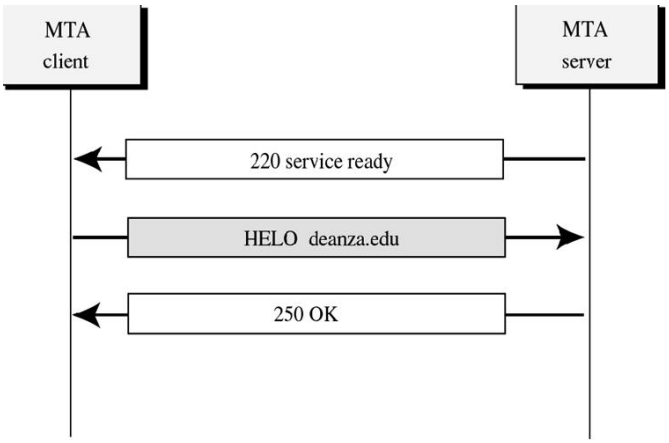
Code	*Description*
	Positive Completion Reply
211	System status or help reply
214	Help message
220	Service ready
221	Service closing transmission channel
250	Request command completed
251	User not local; the message will be forwarded
	Positive Intermediate Reply
354	Start mail input
	Transient Negative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

Basic SMTP operation occurs in three phases:

1. Connection Setup
2. Mail Transfer
3. Connection Termination

Connection Setup

- An SMTP sender will attempt to set up a TCP connection with a target hostwhen it has one or more mail messages to deliver to that host.
- The sequence is quite simple:
  1. The sender opens a TCP connection with the receiver.
  2. Once the connection is established, the receiver identifies itself with "Service Ready".
  3. The sender identifies itself with the HELO command.
  4. The receiver accepts the sender's identification with "OK".
  5. If the mail service on the destination is unavailable, the destination host returns a "Service Not Available" reply in step 2, and the process is terminated.

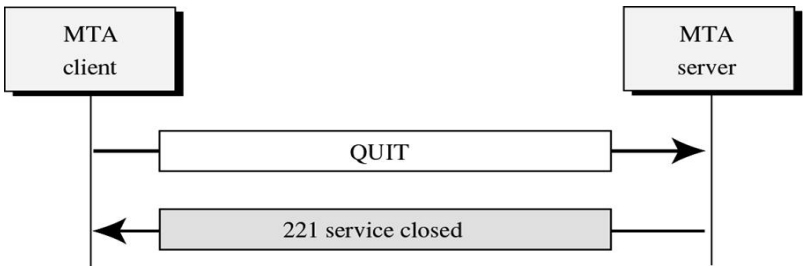


Mail Transfer

- Once a connection has been established, the SMTP sender may send one ormore messages to the SMTP receiver.
- There are three logical phases to the transfer of a message:
  1. A MAIL command identifies the originator of the message.
  2. One or more RCPT commands identify the recipients for this message.
  3. A DATA command transfers the message text.

Connection Termination

- The SMTP sender closes the connection in two steps.
- First, the sender sends a QUIT command and waits for a reply.
- The second step is to initiate a TCP close operation for the TCP connection.
- The receiver initiates its TCP close after sending its reply to the QUITcommand.



LIMITATIONS OF SMTP

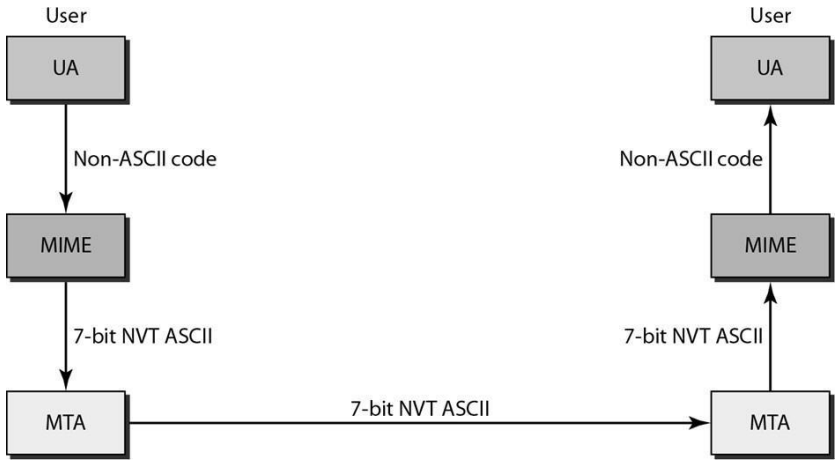
- SMTP cannot transmit executable files or other binary objects.
- SMTP cannot transmit text data that includes national language characters, as these are represented by 8-bit codes with values of 128 decimal or higher, and SMTP is limited to 7-bit ASCII.
- SMTP servers may reject mail message over a certain size.
- SMTP gateways that translate between ASCII and the character code EBCDICdo not use a consistent set of mappings, resulting in translation problems.
- Some SMTP implementations do not adhere completely to the SMTP standards defined.
- Common problems include the following:

1. Deletion, addition, or recording of carriage return and linefeed.
2. Truncating or wrapping lines longer than 76 characters.
3. Removal of trailing white space (tab and space characters).
4. Padding of lines in a message to the same length.

Conversion of tab characters into multiple-space characters

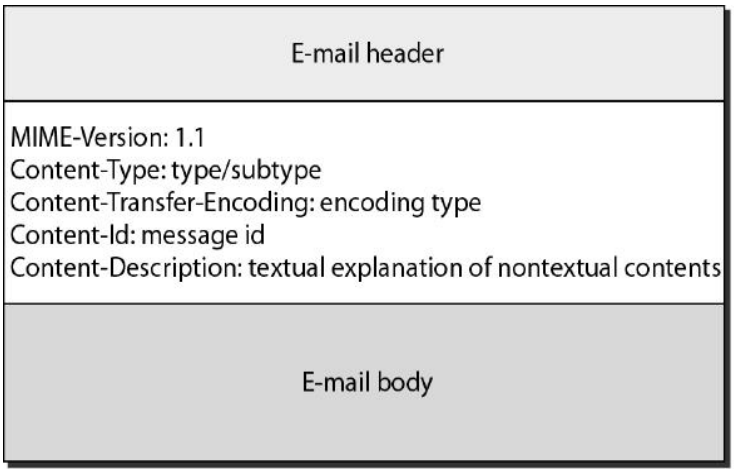
MULTIPURPOSE INTERNET MAIL EXTENSION (MIME)

- SMTP provides a basic email service, while MIME adds multimedia capability to SMTP.
- MIME is an extension to SMTP and is used to overcome the problems and limitations of SMTP.
- Email system was designed to send messages only in *ASCII* format.
  - Languages such as French, Chinese, etc., are not supported.
  - Image, audio and video files cannot be sent.
- MIME adds the following features to email service:
  - Be able to send multiple attachments with a single message;
  - Unlimited message length;
  - Use of character sets other than ASCII code;
  - Use of rich text (layouts, fonts, colors, etc)
  - Binary attachments (executables, images, audio or video files, etc.), whichmay be divided if needed.
- MIME is a protocol that *converts* non-ASCII data to 7-bit NVT (NetworkVirtual Terminal) ASCII and vice-versa.



MIME HEADERS

- Using headers, MIME describes the type of message content and the encodingused.
- *Headers* defined in MIME are:
  - MIME-Version- current version, i.e., 1.1
  - Content-Type - message type (text/html, image/jpeg, application/pdf)
  - Content-Transfer-Encoding - message encoding scheme (eg base64).
  - Content-Id - unique identifier for the message.
  - Content-Description - describes type of the message body.

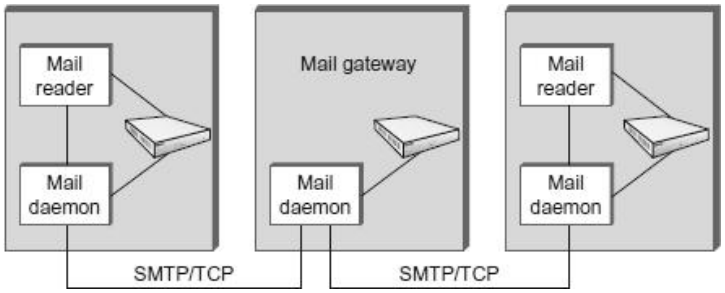


MIME CONTENT TYPES

- There are seven different major types of content and a total of 14 subtypes.
- In general, a content type declares the general type of data, and the subtype specifies a particular format for that type of data.
- MIME also defines a multipart type that says how a message carrying morethan one data type is structured.
- This is like a programming language that defines both base types (e.g., integersand floats) and compound types (e.g., structures and arrays).
- One possible multipart subtype is mixed, which says that the message containsa set of independent data pieces in a specified order.
- Each piece then has its own header line that describes the type of that piece.
- The table below lists the MIME content types:

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

MESSAGE TRANSFER IN MIME



- MTA is a mail daemon (sendmail) active on hosts having mailbox, used to sendan email.
- Mail passes through a sequence of *gateways* before it reaches the recipient mailserver.
- Each gateway stores and forwards the mail using Simple mail transfer protocol (SMTP).
- SMTP defines communication between MTAs over TCP on port 25.
- In an SMTP session, sending MTA is *client* and receiver is *server*. In each exchange:
- Client posts a command (HELO, MAIL, RCPT, DATA, QUIT, VRFY, etc.)
- Server responds with a code (250, 550, 354, 221, 251 etc) and an explanation.
- Client is identified using HELO command and verified by the server
- Client forwards message to server, if server is willing to accept.
- Message is terminated by a line with only single period (.) in it.
- Eventually client terminates the connection.

IMAP (INTERNET MAIL ACCESS PROTOCOL)

- IMAP is an Application Layer Internet protocol that allows an e-mail client toaccess e-mail on a remote mail server.
- It is a method of accessing electronic mail messages that are kept on a possiblyshared mail server.
- IMAP is a more capable wire protocol.
- IMAP is similar to SMTP in many ways.
- IMAP is a client/server protocol running over TCP on port 143.
- IMAP allows multiple clients simultaneously connected to the same mailbox, and through flags stored on the server, different clients accessing the same mailbox at the same or different times can detect state changes made by other clients.
- In other words, it permits a "client" email program to access remote message stores as if they were local.
- For example, email stored on an IMAP server can be manipulated from a desktop computer at home, a workstation at the office, and a notebook computer while travelling, without the need to transfer messages or files back and forth between these computers.
- IMAP can support email serving in three modes:
  - ***Offline***

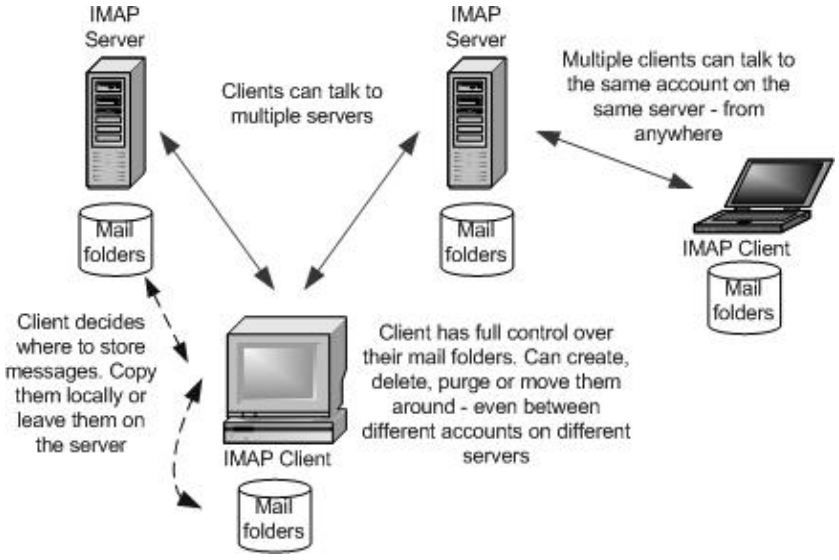
    This means that any information you put into the software isn't stored on your computer but instead on your internet connection.
  - ***Online***

    Users may connect to the server, look at what email is available, and access it online. This looks to the user very much like having local spool

files, but they're on the mail server.

- ***Disconnected operation***

  A mail client connects to the server, can make a "cache" copy of selected messages, and disconnects from the server. The user can then work on the messages offline, and connect to the server later and resynchronize the server status with the cache.



OPERATION OF IMAP

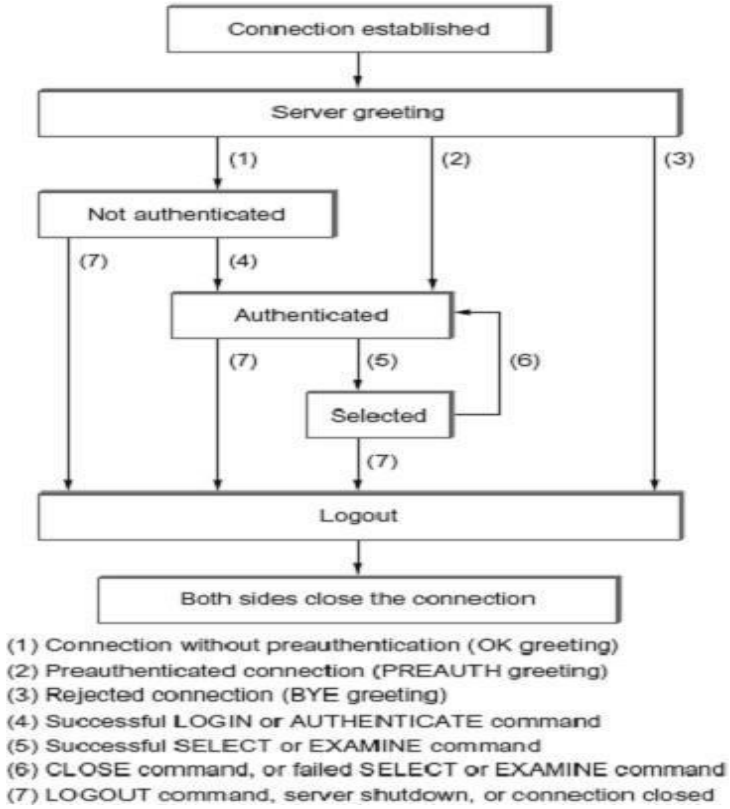
- The mail transfer begins with the client authenticating the user and identifyingthe mailbox they want to access.
- ***Client Commands***
  LOGIN, AUTHENTICATE, SELECT, EXAMINE, CLOSE, and LOGOUT
- ***Server Responses***
  OK, NO (no permission), BAD (incorrect command),
- When user wishes to FETCH a message, server responds in MIME format.
- Message *attributes* such as size are also exchanged.
- *Flags* are used by client to report user actions.
  SEEN, ANSWERED, DELETED, RECENT

IMAP4

- The latest version is IMAP4. IMAP4 is more powerful and more complex.
- IMAP4 provides the following extra functions:
  - A user can check the e-mail header prior to downloading.
  - A user can search the contents of the e-mail for a specific string ofcharacters prior to downloading.
  - A user can partially download e-mail. This is especially useful if bandwidth is limited and the e-mail contains multimedia with high bandwidth requirements.
  - A user can create, delete, or rename mailboxes on the mail server.
  - A user can create a hierarchy of mailboxes in a folder for e-mail storage.

ADVANTAGES OF IMAP

- With IMAP, the primary storage is on the server, not on the local machine.
- Email being put away for storage can be foldered on local disk, or can be

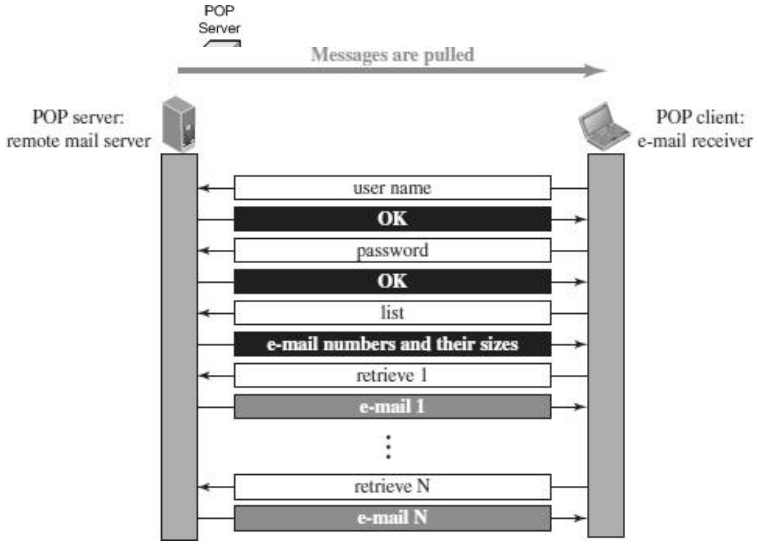


  foldered on the IMAP server.
- The protocol allows full user of remote folders, including a remote folder hierarchy and multiple inboxes.
- It keeps track of explicit status of messages, and allows for user-defined status.
- Supports new mail notification explicitly.
- Extensible for non-email data, like netnews, document storage, etc.
- Selective fetching of individual MIME body parts.
- Server-based search to minimize data transfer.
- Servers may have extensions that can be negotiated.

POST OFFICE PROTOCOL (POP3)

- Post Office Protocol (POP3) is an application-layer Internet standard protocol used by local e-mail clients to retrieve e-mail from a remote server over a TCP/IP connection.
- There are two versions of POP.
  - The first, called *POP2,* became a standard in the mid-80's and requires SMTP to send messages.
  - The current version, POP3, can be used with or without SMTP. POP3 uses TCP/IP port 110.
- POP is a much simpler protocol, making implementation easier.
- POP supports offline access to the messages, thus requires less internet usagetime
- POP does not allow search facility.
- In order to access the messages, it is necessary to download them.
- It allows only one mailbox to be created on server.
- It is not suitable for accessing non mail data.
- POP mail moves the message from the email server onto the local computer, although there is usually an option to leave the messages on the email server as well.
- POP treats the mailbox as one store, and has no concept of folders.

- POP works in two modes namely, ***delete*** **and** ***keep*** mode.
  - In ***delete mode***, mail is *deleted* from the mailbox after retrieval. The delete mode is normally used when the user is working at their permanent computer and can save and organize the received mail after reading or replying.
  - In ***keep mode***, mail after reading is *kept* in mailbox for later retrieval. The keep mode is normally used when the user accesses her mail away from their primary computer.



- POP3 client is *installed* on the recipient computer and POP server on the mailserver.
- Client *opens* a connection to the server using TCP on port 110.
- Client sends username and password to *access* mailbox and to retrievemessages.

POP3 Commands

POP commands are generally abbreviated into codes of three or four lettersThe following describes some of the POP commands:

1. **UID -** This command opens the connection
2. **STAT -** It is used to display number of messages currently in the mailbox
3. **LIST -** It is used to get the summary of messages
4. **RETR -**This command helps to select a mailbox to access the messages
5. **DELE -** It is used to delete a message
6. **RSET -** It is used to reset the session to its initial state
7. **QUIT -** It is used to log off the session

DIFFERENCE BETWEEN POP AND IMAP

SNo.	POP	IMAP
1	Generally used to support single client.	Designed to handle multiple clients.
2	Messages are accessed offline.	Messages are accessed online although it also supports offline mode.
3	POP does not allow search facility.	IMAP offers ability to search emails.
4	All the messages have to be downloaded.	It allows selective transfer of messages tothe client.

5	Only one mailbox can be created onthe server.	Multiple mailboxes can be created on the server.
6	Not suitable for accessing non-maildata.	Suitable for accessing non-mail data i.e. attachment.
7	POP commands are generally abbreviated into codes of three or four letters. Eg. STAT.	IMAP commands are not abbreviated, they are full. Eg. STATUS.
8	It requires minimum use of server resources.	Clients are totally dependent on server.
9	Mails once downloaded cannot be accessed from some other location.	Allows mails to be accessed from multiple locations.
10	The e-mails are not downloaded automatically.	Users can view the headings and senderof e-mails and then decide to download.
11	POP requires less internet usage time.	IMAP requires more internet usage time.

Advantages of IMAP over POP

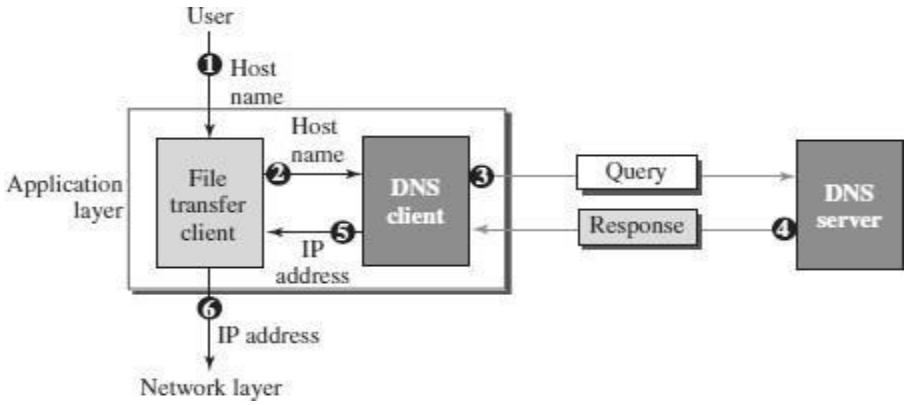
- IMAP is more powerful and more complex than POP.
- User can *check* the e-mail header prior to downloading.
- User can *search* e-mail for a specific string of characters prior to downloading.
- User can download *partially,* very useful in case of limited bandwidth.

User can create, delete, or rename *mailboxes* on the mail server.

DOMAIN NAME SYSTEM(DNS)

- Domain Name System was designed in 1984.
- DNS is used for name-to-address mapping.
- The DNS provides the protocol which allows clients and servers tocommunicate with each other.
- Eg: Host name like www.yahoo.com is translated into numerical IP addresses like 207.174.77.131
- Domain Name System (DNS) is a distributed database used by TCP/IP applications to map between hostnames and IP addresses and to provide electronic mail routing information.
- Each site maintains its own database of information and runs a server program that other systems across the Internet can query.

WORKING OF DNS

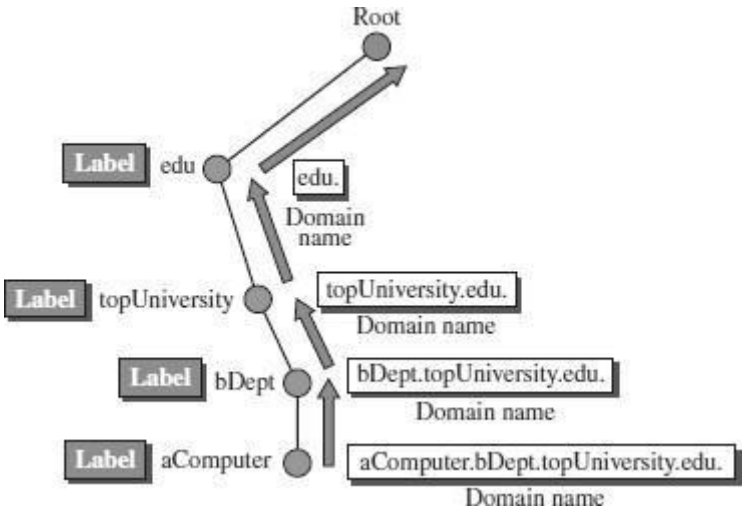


The following six steps shows the working of a DNS. It maps the host name to an IP address:

1. The user passes the host name to the file transfer client.
2. The file transfer client passes the host name to the DNS client.
3. Each computer, after being booted, knows the address of one DNS server. The DNS client sends a message to a DNS server with a query that gives the file transfer server
   name using the known IP address of the DNS server.
4. The DNS server responds with the IP address of the desired file transfer server.
5. The DNS server passes the IP address to the file transfer client.
6. The file transfer client now uses the received IP address to access the file transfer server.

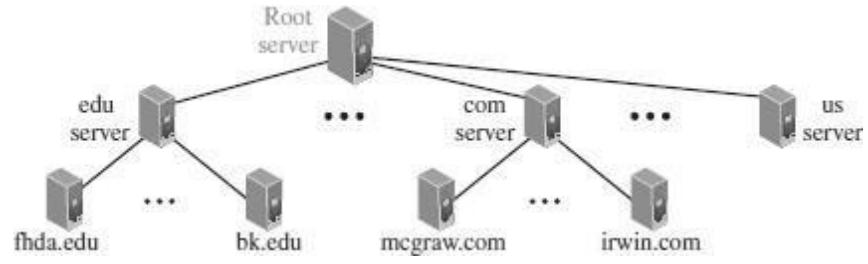
Domain Name

- Each node in the tree has a label called as domain name.
- A full domain name is a sequence of labels separated by dots (**.**)
- The domain names are always read from the node up to the root.
- The last label is the label of the root (null).
- This means that a full domain name always ends in a null label, which means the last character is a dot because the null string is nothing.
- If a label is terminated by a null string, it is called a ***fully qualified domainname (FQDN)***.
- If a label is not terminated by a null string, it is called a ***partially qualified domain name (PQDN).***



HIERARCHY OF NAME SERVERS

- The way to distribute information among DNS servers is to divide the wholespace into many domains based on the first level.
- Let the root stand-alone and create as many domains as there are first level nodes.
- Because a domain created this way could be very large,
- DNS allows domains to be divided further into smaller domains.
- Thus, we have a hierarchy of servers in the same way that we have a hierarchyof names.

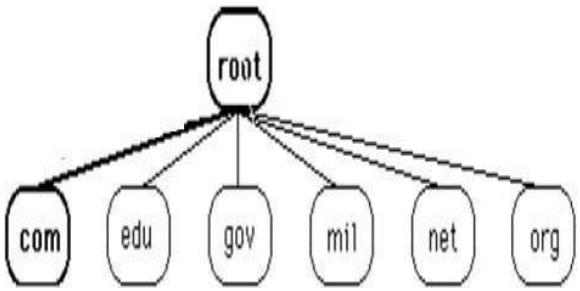


DNS IN THE INTERNET

- DNS is a protocol that can be used in different platforms.
- In the Internet, the domain name space (tree) is divided into three different sections - ***Generic domains, Country domains, and Inverse domain***.

Generic Domains

- The generic domains define registered hosts according to their genericbehavior.
- Each node in the tree defines a domain, which is an index to the domain namespace database.
- The first level in the generic domains section allows seven possible threecharacter levels.
- These levels describe the organization types as listed in following table**.**



.COM	Commercial Organizations
.EDU	Educational institutions
.GOV	Government institutions
.MIL	Military groups
.NET	Major network support centers
.INT	International Organizations
.ORG	Nonprofit Organizations

Country Domains

- The country domains section follows the same format as the generic domainsbut uses two characters for country abbreviations
- E.g.; ***in*** for ***India***, ***us*** for ***United States*** etc) in place of the three character organizational abbreviation at the first level.
- Second level labels can be organizational, or they can be more specific,national designation.
- India for example, uses state abbreviations as a subdivision of the country domain us. (e.g., ca.in.)

Inverse Domains

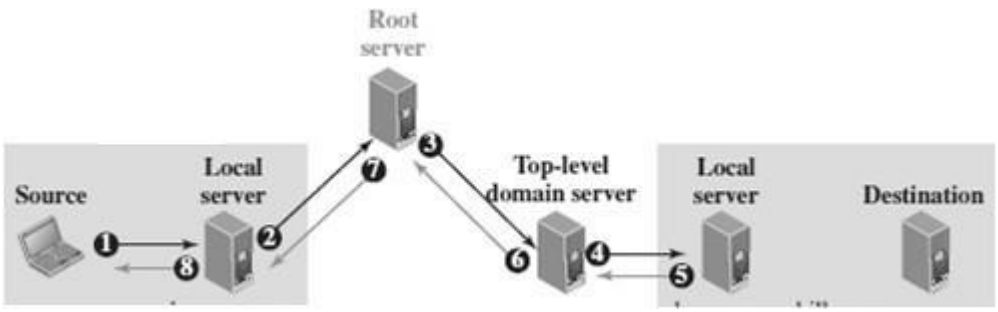
- Mapping an address to a name is called Inverse domain.

- The client can send an IP address to a server to be mapped to a domain name and it is called *PTR(Pointer) query*.
- To answer queries of this kind, DNS uses the inverse domain

DNS RESOLUTION

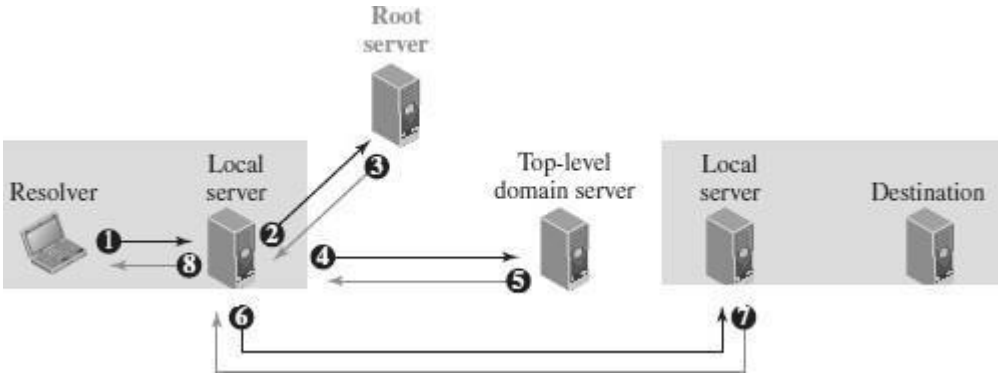
- Mapping a name to an address or an address to a name is called name address resolution.
- DNS is designed as a client server application.
- A host that needs to map an address to a name or a name to an address calls a DNS client named a ***Resolver***.
- The Resolver accesses the closest DNS server with a mapping request.
- If the server has the information, it satisfies the resolver; otherwise, it either refers the resolver to other servers or asks other servers to provide the information.
- After the resolver receives the mapping, it interprets the response to see if it is areal resolution or an error and finally delivers the result to the process that requested it.
- A resolution can be either ***recursive or iterative***.

Recursive Resolution



- The application program on the source host calls the DNS resolver (client) to find the IP address of the destination host. The resolver, which does not know this address, sends the query to the local DNS server of the source (Event 1)
- The local server sends the query to a root DNS server (Event 2)
- The Root server sends the query to the top-level-DNS server (Event 3)
- The top-level DNS server knows only the IP address of the local DNS server at the destination. So, it forwards the query to the local server, which knows the IPaddress of the destination host (Event 4)
- The IP address of the destination host is now sent back to the top-level DNS server (Event 5) then back to the root server (Event 6), then back to the source DNS server, which may cache it for the future queries (Event 7), and finally back to the source host (Event 8).

Iterative Resolution



- In iterative resolution**,** each server that does not know the mapping, sends theIP address of the next server back to the one that requested it.
- The iterative resolution takes place between two local servers.
- The original resolver gets the final answer from the destination local server.
- The messages shown by Events 2, 4, and 6 contain the same query.
- However, the message shown by Event 3 contains the IP address of the top-level domain server.
- The message shown by Event 5 contains the IP address of the destination localDNS server
- The message shown by Event 7 contains the IP address of the destination.
- When the Source local DNS server receives the IP address of the destination, it sends it to the resolver (Event 8).

DNS CONNECTIONS

- DNS can use either UDP or TCP.
- In both cases the well-known port used by the server is port 53.
- UDP is used when the size of the response message is less than 512 bytes because most UDP packages have a 512-byte packet size limit.
- If the size of the response message is more than 512 bytes, a TCP connection isused.

DNS REGISTRARS

- New domains are added to DNS through a *registrar.* A fee is charged.
- A registrar first verifies that the requested domain name is unique and thenenters it into the DNS database.
  - Today, there are many registrars; their names and addresses can be found at

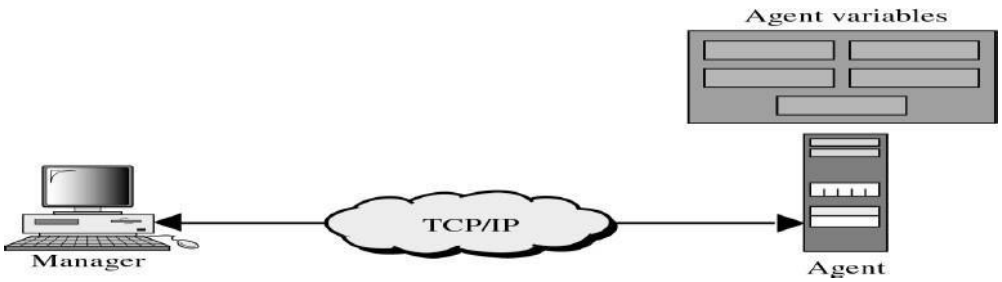
    http://www.intenic.net
- To register, the organization needs to give the name of its server and the IP address of the server.
- For example, a new commercial organization named *wonderful* with a server named *ws* and IP address 200.200.200.5, needs to give the following information to one of the registrars:

  Domain name: *ws.wonderful.com* ***IP address:*** *200.200.200.5*

SNMP (SIMPLE NETWORK MANAGEMENT

PROTOCOL)

- The **Simple Network Management Protocol (SNMP)** is a framework for managing devices in an internet using the TCP/IP protocol suite.
- SNMP is an application layer protocol that monitors and manages routers, distributed over a network.
- It provides a set of operations for monitoring and managing the internet.
- SNMP uses services of UDP on two well-known ports: 161 (Agent) and 162 (manager).
- SNMP uses the concept of ***manager*** and ***agent***.



SNMP MANAGER

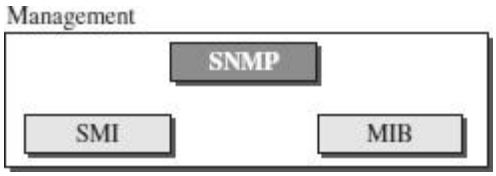
- A manager is a host that runs the SNMP client program
- The manager has access to the values in the database kept by the agent.
- A manager checks the agent by requesting the information that reflects the behavior of the agent.
- A manager also forces the agent to perform a certain function by resettingvalues in the agent database.
- For example, a router can store in appropriate variables the number of packets received and forwarded.
- The manager can fetch and compare the values of these two variables to see ifthe router is congested or not.

SNMP AGENT

- The agent is a router that runs the SNMP server program.
- The agent is used to keep the information in a database while the manager is used to access the values in the database.
- For example, a router can store the appropriate variables such as a number of packets received and forwarded while the manager can compare these variablesto determine whether the router is congested or not.
- Agents can also contribute to the management process.
- A server program on the agent checks the environment, if something goes wrong, the agent sends a warning message to the manager.

SNMP MANAGEMENT COMPONENTS

- Management of the internet is achieved through simple interaction between a manager and agent.
- Management is achieved through the use of two protocols:
  - Structure of Management Information (SMI)
  - Management Information Base (MIB).



Structure of Management Information (SMI)

- To use SNMP, we need rules for naming objects.
- SMI is a protocol that defines these rules.
- SMI is a guideline for SNMP
- It emphasizes three attributes to handle an object: name, data type, andencoding method.
- Its functions are:
  - ❖ To name objects.
  - ❖ To define the type of data that can be stored in an object.
  - ❖ To show how to encode data for transmission over the network.

Name

- ✓ SMI requires that each managed object (such as a router, a variable in a router,a value,etc.) have a unique name. To name objects globally.
- ✓ SMI uses an ***object identifier*****,** which is a hierarchical identifier based on a tree structure.
- ✓ The tree structure starts with an unnamed root. Each object can be definedusing a sequence of integers separated by dots.
- ✓ The tree structure can also define an object using a sequence of textual names separated by dots.

Type of data

- ✓ The second attribute of an object is the type of data stored in it.
- ✓ To define the data type, SMI uses **Abstract Syntax Notation One (ASN.1)** definitions.
- ✓ SMI has two broad categories of data types: *simple* and *structured.*
- ✓ The **simple data types** are atomic data types. Some of them are taken directlyfrom ASN.1; some are added by SMI.
- ✓ SMI defines two **structured data types:** *sequence* and *sequence of.*
  - ▪ ***Sequence -*** A *sequence* data type is a combination of simple data types, not necessarily of the same type.
  - ▪ ***Sequence of -*** A *sequence of* data type is a combination of simple data types all of the same type or a combination of sequence data types all of the same type.

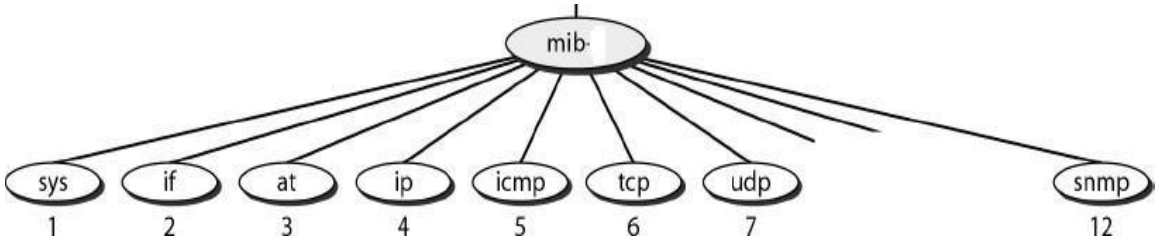
Encoding data

- ✓ SMI uses another standard, ***Basic Encoding Rules (BER),*** to encode data to be transmitted over the network.
- ✓ BER specifies that each piece of data be encoded in triplet format (TLV): tag, length, value

Management Information Base (MIB)

The Management Information Base (MIB) is the second component used in networkmanagement.

- Each agent has its own MIB, which is a *collection* of objects to be managed.
- MIB classifies objects under groups.



MIB Variables

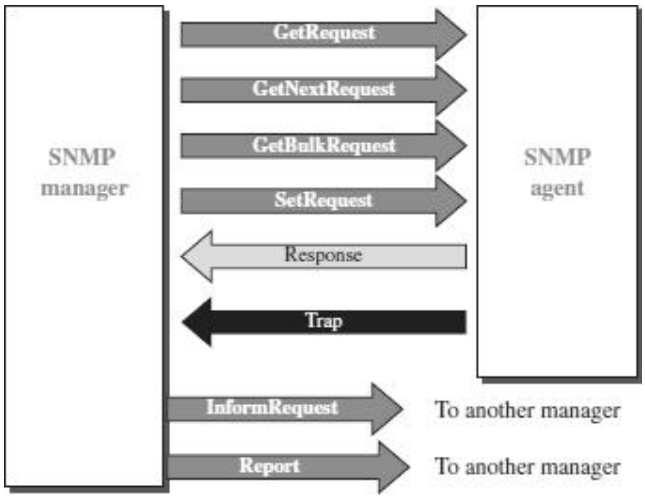
MIB variables are of two types namely *simple* and *table*.

- Simple variables are accessed using *group-id* followed by *variable-id* and 0
- Tables are ordered as *column-row* rules, i.e., column by column from top to bottom. Only *leaf* elements are accessible in a table type.

SNMP MESSAGES/PDU

SNMP is request/reply protocol that supports various operations using PDUs. SNMP defines eight types of protocol data units (or PDUs):

GetRequest, GetNext-Request, GetBulkRequest, SetRequest, Response, Trap, InformRequest,* and *Report



GetRequest

- The GetRequest PDU is sent from the manager (client) to the agent (server) to retrieve the value of a variable or a set of variables.

GetNextRequest

- The GetNextRequest PDU is sent from the manager to the agent to retrieve thevalue of a variable.

GetBulkRequest

- The GetBulkRequest PDU is sent from the manager to the agent to retrieve a large amount of data. It can be used instead of multiple GetRequest and GetNextRequest PDUs.

SetRequest

- The SetRequest PDU is sent from the manager to the agent to set

(store) avalue in a variable.

Response

- The Response PDU is sent from an agent to a manager in response to GetRequest or GetNextRequest. It contains the value(s) of the variable(s) requested by the manager.

Trap

- The Trap PDU is sent from the agent to the manager to report an event. For example, if the agent is rebooted, it informs the manager and reports the time ofrebooting.

InformRequest

- The InformRequest PDU is sent from one manager to another remote manager to get the value of some variables from agents under the control of the remote manager. The remote manager responds with a Response PDU.

Report

- The Report PDU is designed to report some types of errors between managers.