5. Web Application & Service Protocol

DD MM YYYY

23.1 :- ARCHITECTURE

* First Scenario -

1) In the first scenario, the sender and the receiver of the e-mail are users (or application programs) on the same mail server; they are directly connected to a shared mail server.
2) The administrator has created the mailbox for each user where the received messages are stored.
3) A mailbox is part of local hard drive, a special file with permission restrictions. only the owner of the mailbox has access to it.
4) When Alice needs to send a message to Bob, she runs a user agent (UA) program to prepare the message & store it in Bob's mailbox.

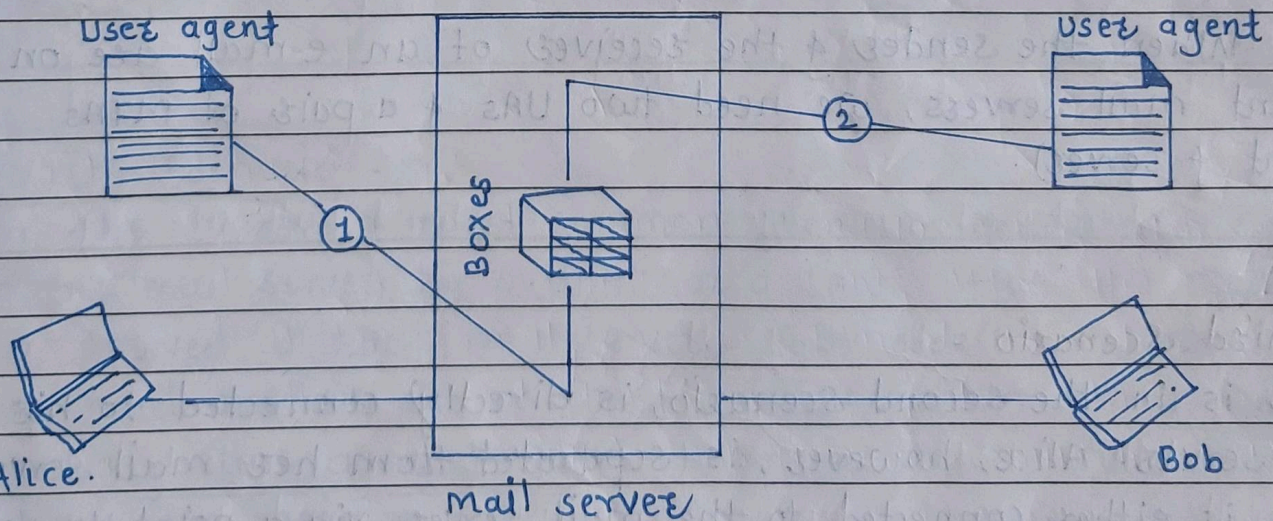


Fig 23.1 First Scenario.

When the sender & the receiver of an e-mail are on the same mail server; we need only two user agents.

* Second Scenario -

1) In the second scenario, the sender & the receiver of the e-mail are users (or application programs.) on two diffe mail servers.

2) The message needs to be sent over the Internet. Here w need user agents (UAS) & message transfer agents (MTA

UA - user agent

MTA - message transfer agent.

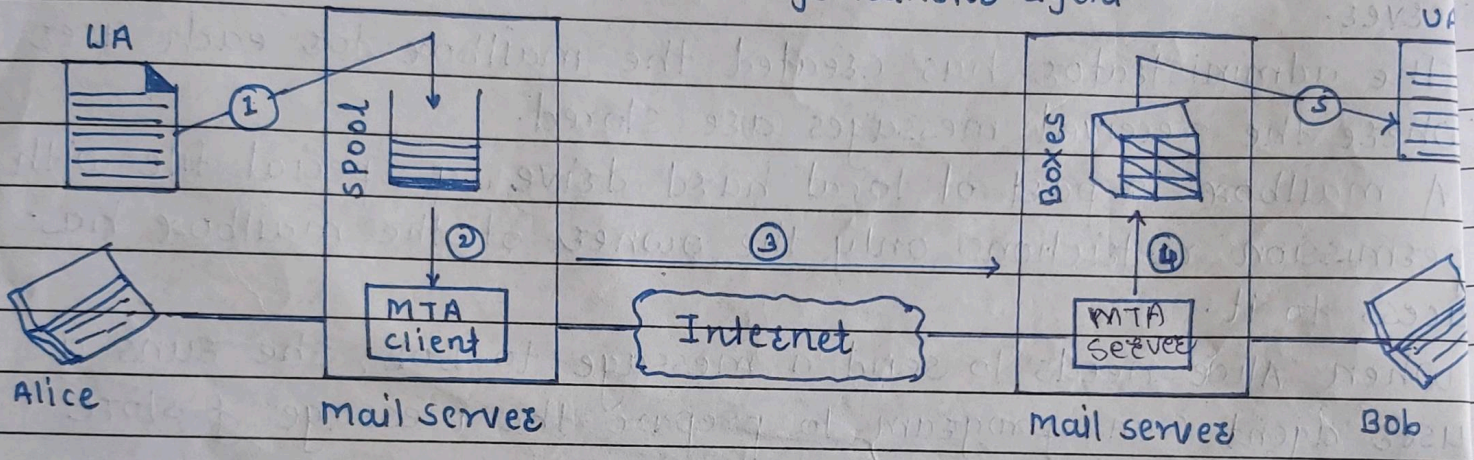


Fig. 23.2 Second scenario.

When the sender & the receiver of an e-mail are on different mail servers, we need two UAs & a pair of MTAs (client & server).

* Third Scenario -

1) Bob, is in the second scenario, is directly connected to his mail server. Alice, however, is separated from her mail server.

2) Alice is either connected to the mail server via a point-to-point WAN - such as a dial-up modem, a DSL, or a cable modem - or she is connected to a LAN in a organization that uses one mail server for handling e-mails; all users need to send send their messages to this mail server.

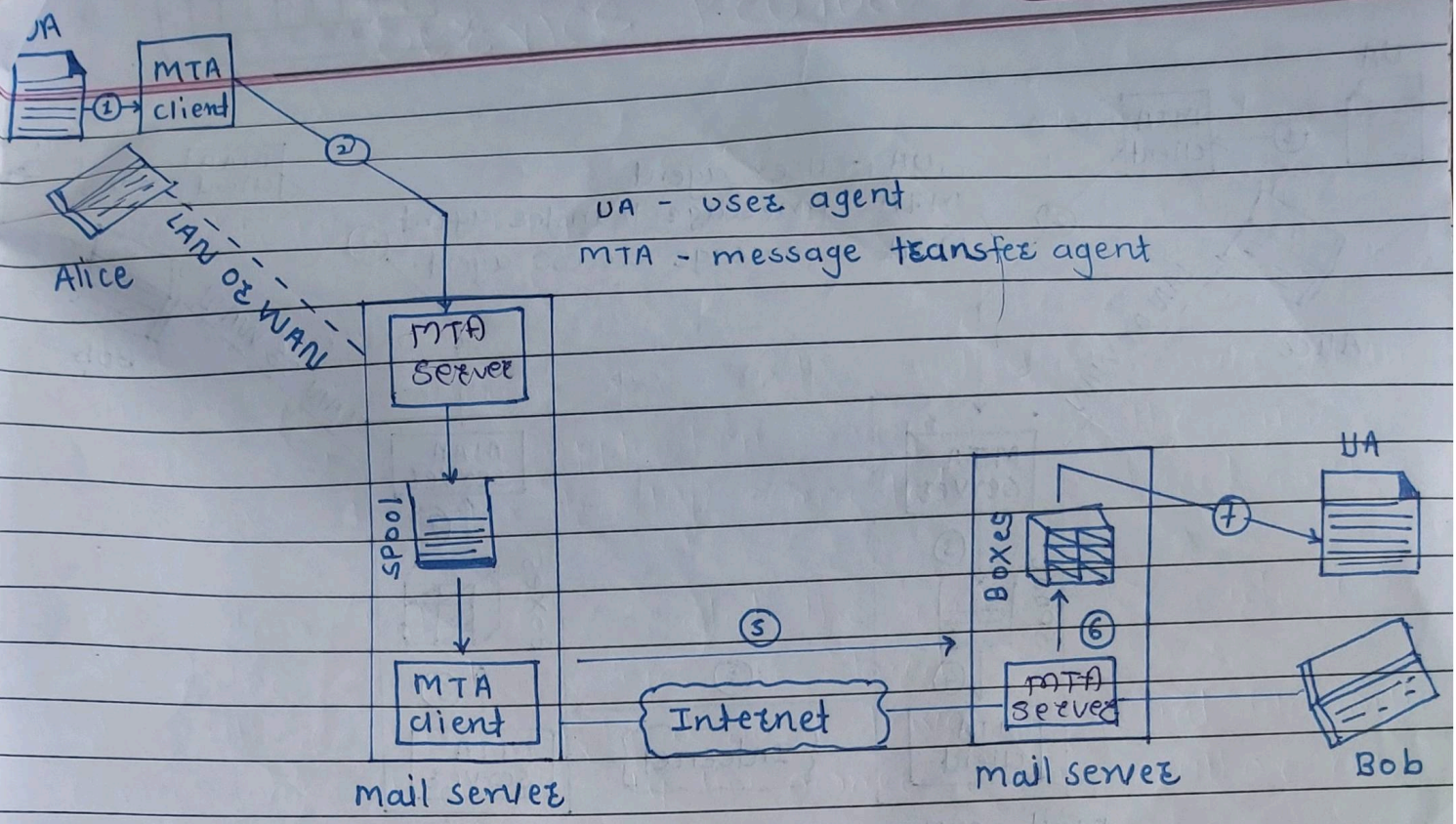


Fig 23.3 Third scenario

When the sender is connected to the mail server via a LAN or a WAN, we need two UAs & two pairs of MTAs (client & server)

* Fourth Scenario -

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

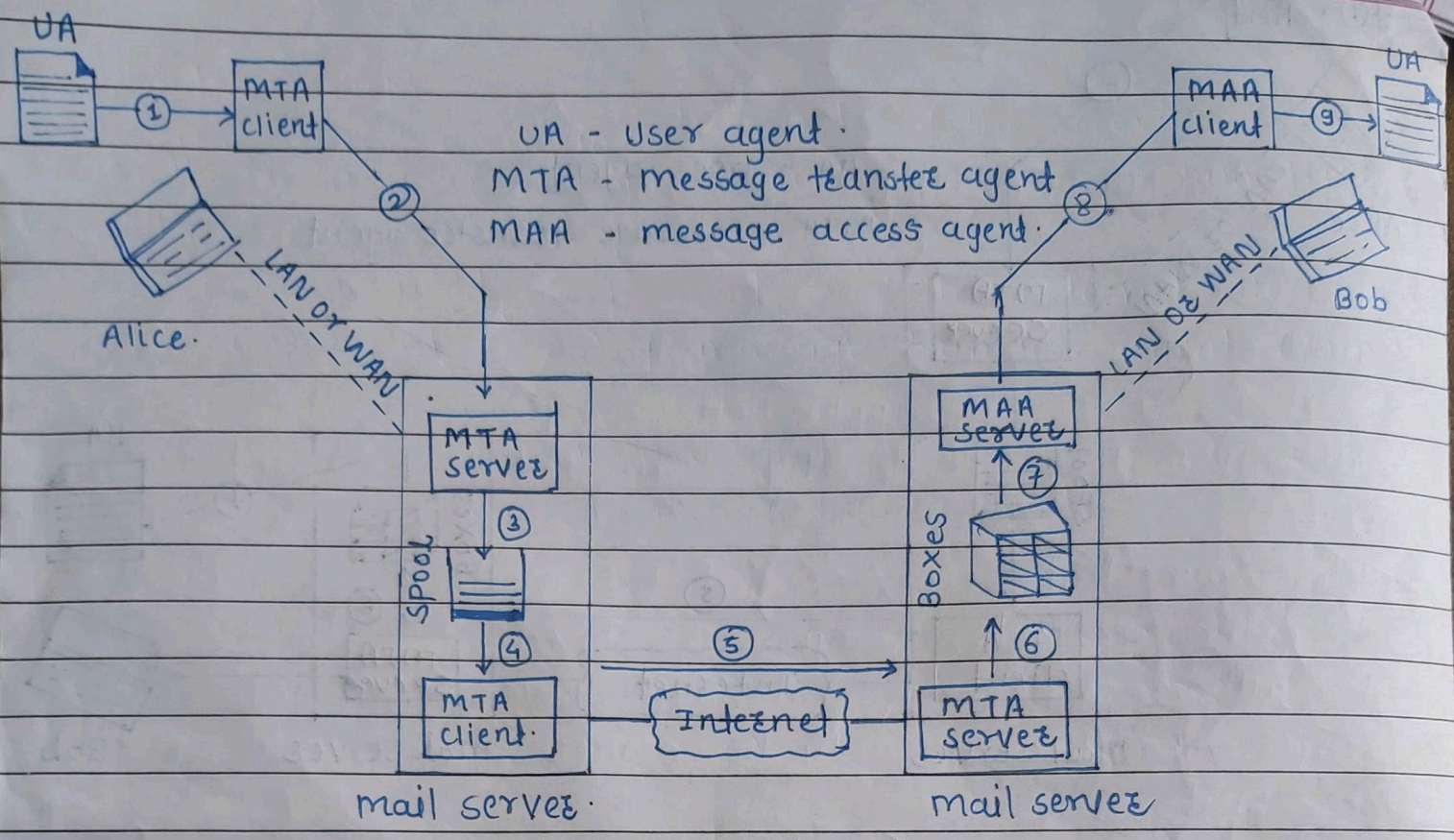
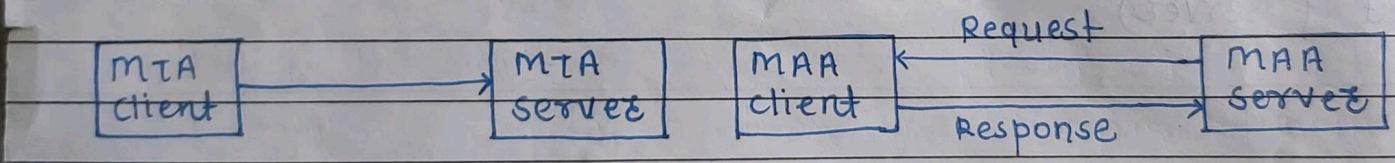


Fig 23.4 Fourth Scenario.

* Push vs pull -



a. client pushes messages. b. client pulls messages.

Fig 23.5 Push vs Pull

When both sender & receiver are connected to the mail server via a LAN or a WAN, we need two UAs, two pairs of MTAs (client & server), and a pair of MAAs (client & server). This is the most common situation today.

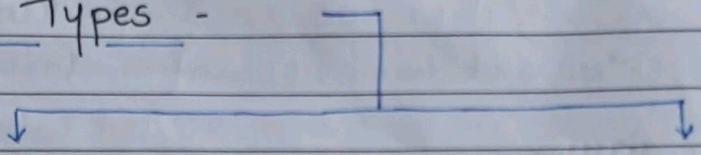
23.2 :- USER AGENT

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 & receiving a message easier.

* Services Provided by a User Agent -

A user agent is a software package (program) that composes, reads, replies to, and forwards messages. It also handles local mailboxes on the user computers.

* User Agent Types -



① command-driven ② GUI-based

① command driven -

1) The command driven are agents belong to the early days of electronic mail.
2) They are still present as the underlying user agents in servers.
3) A command-driven user agent normally accepts a one character command from the keyboard to perform its task.
4) For example,-
a user can type the character r, at the command prompt, to reply to the sender of the message, or type the character R to Reply to the sender & all recipients.

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

② GUI-based -

1) They contain graphical user interface (GUI) components that allow the user to interact with the software by using both the keyboard & the mouse.

2) The graphical components such as icons, menu bars, and windows that make the services easy to access.

Some examples of GUI-based user agents are Eudora, Outlook, and Netscape.

* (Sending) mail -

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

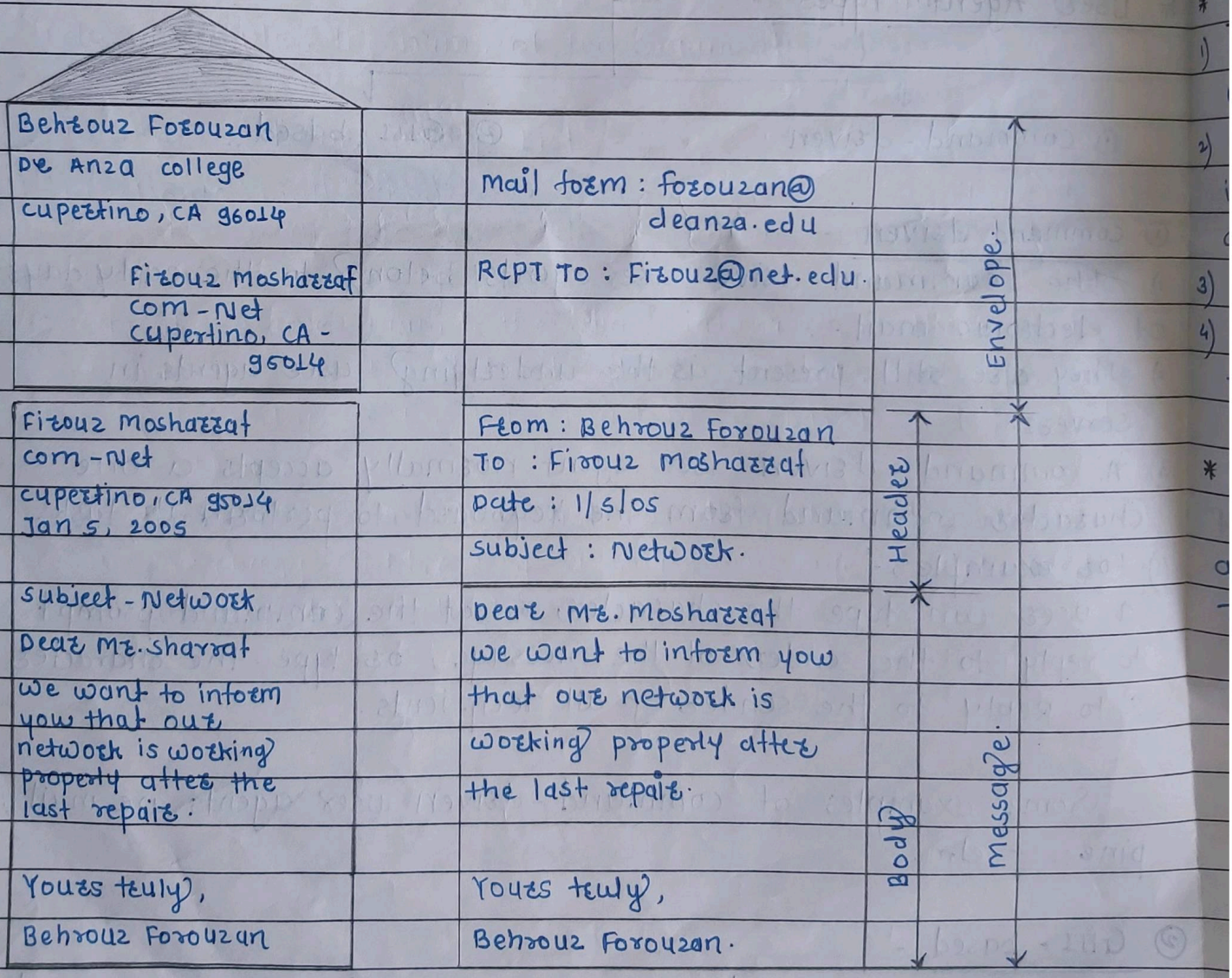


Fig. 23.6 Format of an e-mail.

* Envelope -

The envelope usually contains the sender address, t receiver address, and other information.

* Message -

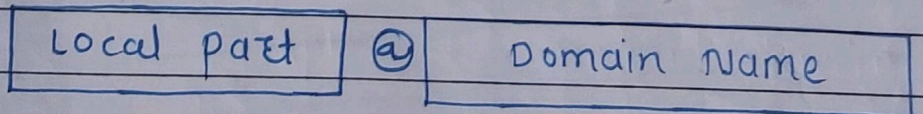
The message contains the header and the body. The header of the message defines the sender, the receiver, the subject of the message, & some other information. The body of the message contains the actual information to be read by the recipient.

* Receiving mail -

1) The user agent is triggered by the user (or a timer). If a user has mail, the UA informs the user with a notice.
2) If the user is ready to read the mail, a list is displaye in which each line contains a summary of the information about a particular message in the mail box.
3) select any of the messages & display its contents on the sc
4) It is includes sender mail address, the subject, and the ti the mail was sent or received.

* Addresses -

The deliever mail, a mail handling system must use an addressing system with unique addresses.
In this Internet, the addresses consists of two parts: a local part & a domain name, separated by an @ sign.



Mailbox address of the recepient.

The domain name of the mail server.

Fig 23.7. Email addresses

* Local Part -

The Local part defines the same name of a special file, called the user mailbox, where all of the mail received for a user is stored for retrieval by the message access agent.

* Domain Name -

1) The second part of the adress is the domain name. An organization usually selects one or more hosts to receive & send e-mail; they are sometimes called mail servers or exchangers.
2) The domain name assigned to each mail exchanger either comes from the DNS database or is a logical name. (for example, the name of the organization.)

* Mailing List or Group List -

1) Electronic mail allows one name, an alias, to represent several different e-mail addresses; this is called a mailing list.
2) Every time a message is to be sent, the system checks the recipient's name against the alias database; if there is a mailing list for the defined alias, separate messages, one for each entry in the list, must be prepared and handed to the MTA.
3) If there is no mailing list for the alias, the name itself is the receiving address & a single message is delievered to the mail transfer entity.

2] Responses -

Responses are sent from the server to the client.

* Table 23.2 - 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 Intermidiate Reply.
354	start mail input
	Transient Negative completion Reply.
421	service is 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.

* Mail Transfer Phases - The process of transferring a mail message occurs in three phases: 1) connection establishment
2) mail transfer 3) connⁿ termination.

1] Connection Establishment -

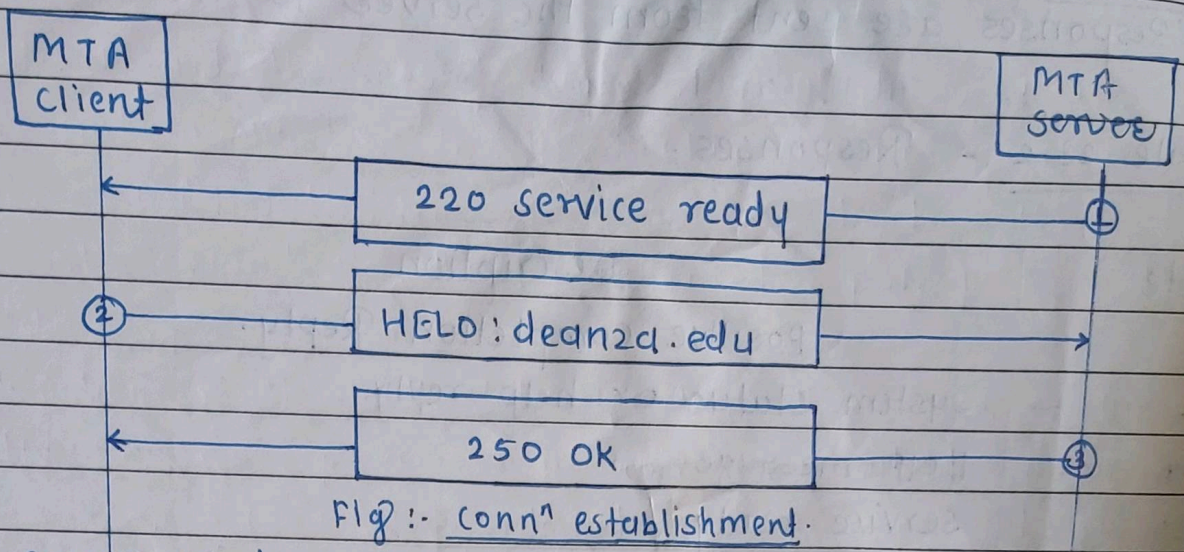


Fig :- Connⁿ establishment.

1) The server sends code 220 to tell the client that it is ready to receive mail. If the server is not ready, it sends code 421.
2) The client sends the HELO message to identify itself using its domain name address. Remember that during TCP connⁿ establishment, the sender & receiver know each other through their IP addresses.
3) The server responds with code 250 or some other code depending on the situation.

2] Message Transfer -

1) The client sends the MAIL FROM message to introduce the sender of the message. This step is needed to give the server the return mail address for returning errors & reporting messages.
2) The server responds with code 250 or some other appropriate code. 3]
3) The client sends the RCPT TO message (recipient), which includes the mail address of the receipient.
4) The server responds with code 250 or some other appropriate code.
5) The client sends the DATA message to initialize the msg transfer.
6) The server responds with code 354 (start mail input) or some other appropraiate message.
7) The client sends the contents of the message in consecutive lines. each line is terminated by a two-character end-of-line token (carriage return & line feed). The msg is terminated by a line containing just one period.
8) The server responds with code 250 (OK) or some other appropriate

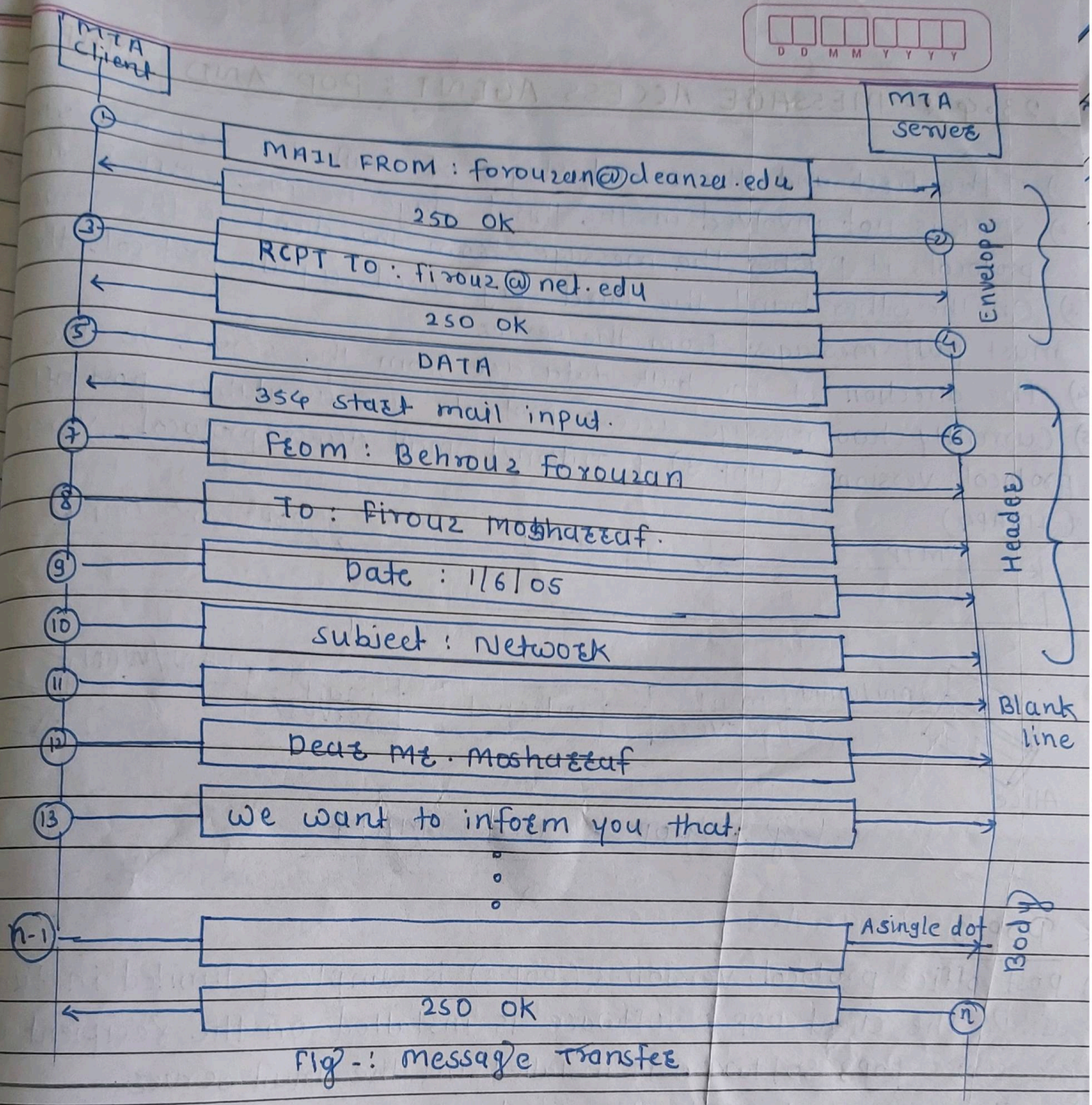


Fig -: message transfer

Connection Termination -

After the message is transferred successfully, the client terminates the connection. This phase involves two steps.

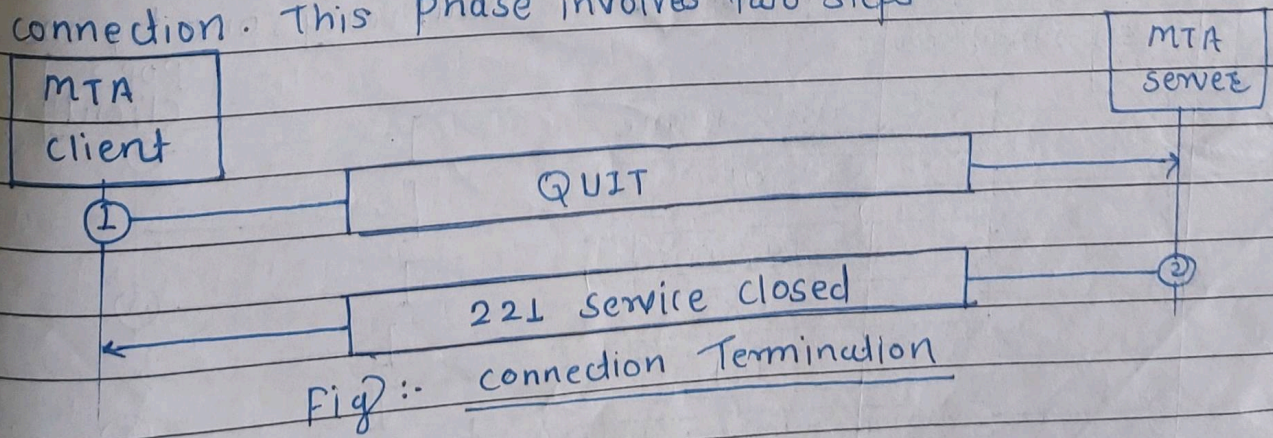


Fig :- connection Termination

1) The client sends the QUIT command.
2) The server responds with code 221 or some other appropriate code.

After the conn^n termination phase, the TCP conn^n must be closed.

23.4 :- MESSAGE ACCESS AGENT : POP AND IMAP

1) The first and the second stages of mail delievery use SMTP.
2) SMTP is not involved in the third stage because SMTP is a push protocol; it pushes the message from the client to the server.
3) On the other hand, the third stage needs a pull protocol; the client must pull messages from the server.
4) The direction of the bulk data are from the server to the client.
5) Currently two message access protocols are available: post office protocol, version 3 (POP 3) & Internet mail Access protocol, version 4 (IMAP4).

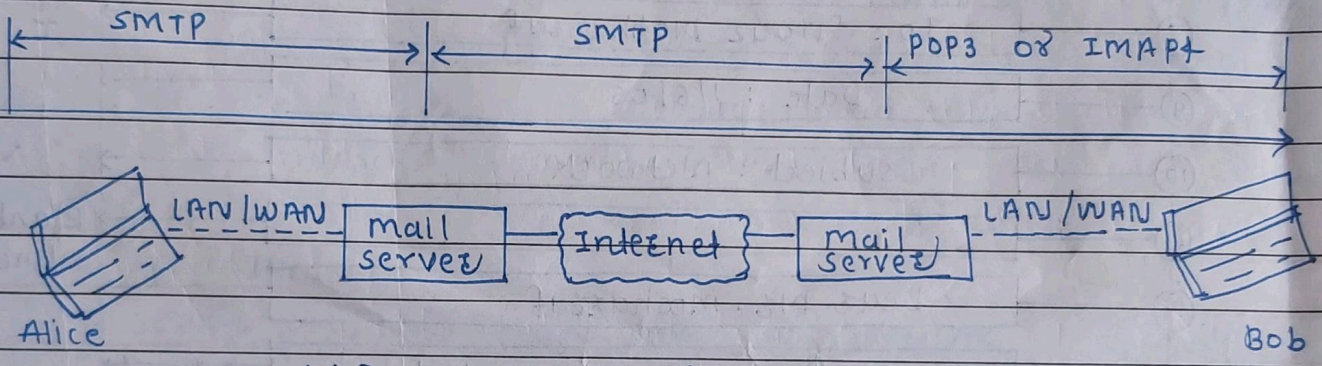


Fig :- POP 3 and IMAP4

* POP 3 -

1) Post office protocol, version 3 (POP3) is simple & limited in functionality. The client POP 3 software is installed on the recipient computer, the server pop3 software is installed on the mail server.
2) mail access starts with the client when the user needs to download its e-mail from the mailbox on the mail server.
3) The user can then list & retrive the mail msg, one by one.

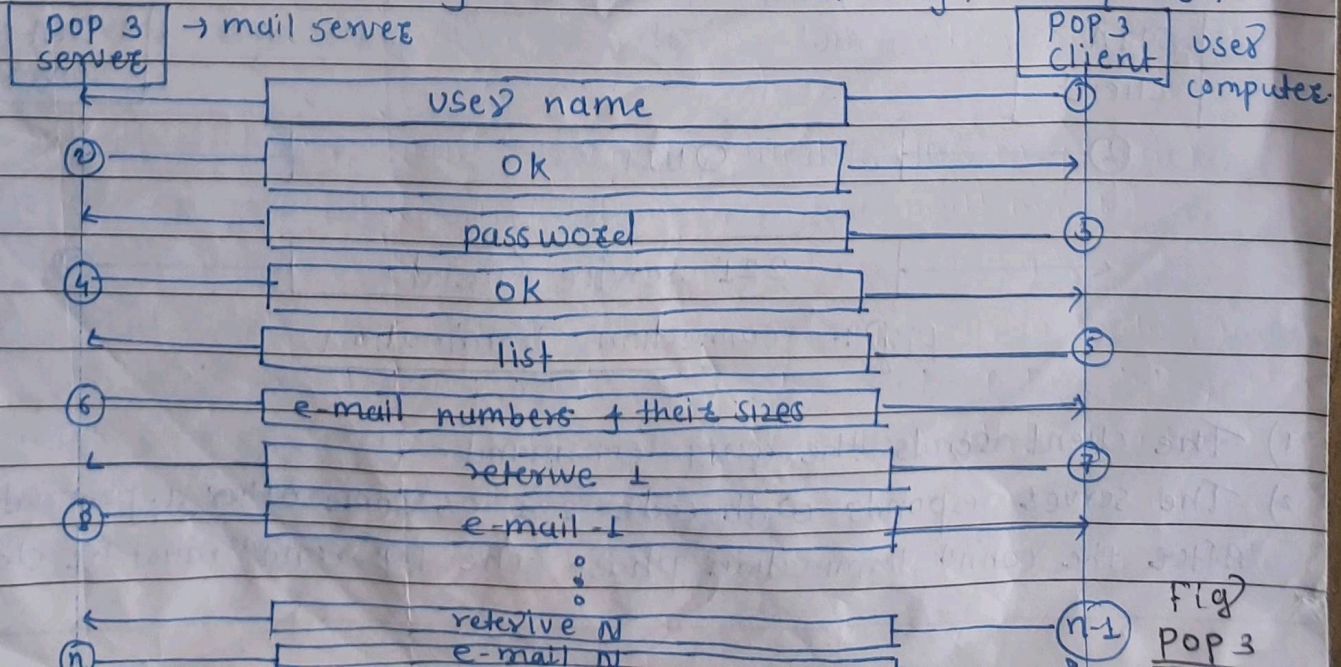


Fig POP 3

23.3 :- MESSAGE TRANSFER AGENT : SMTP.

1) The actual mail transfer is done through message transfer agents (MTAs).
2) To send mail, a system must have the client MTA, and to receive mail, a system must have a server MTA.
3) The formal protocol that defines the MTA client & server in the Internet is called simple mail Transfer protocol (SMTP).

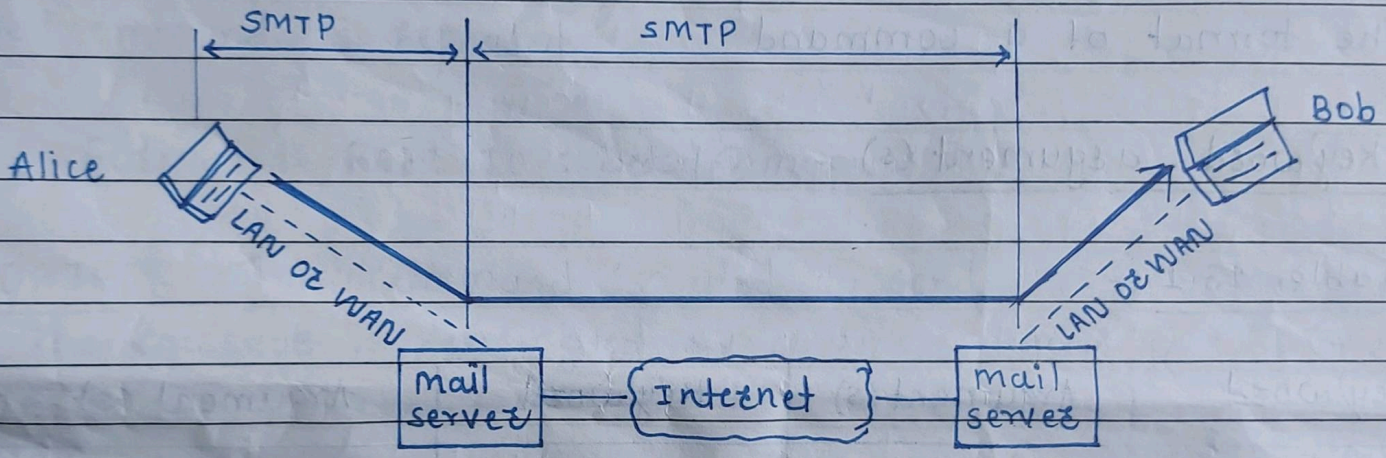


Fig. 23.8 SMTP range.

4) SMTP is used two times, betn the sender & the sender's mail server & betn the two mail servers.
5) SMTP simply defines how command & responses must be sent back & forth.
6) Each network is free to choose a software package for implementation.

---

* Commands and Responses -

SMTP uses commands and Responses to transfer messages between an MTA client & an MTA server.

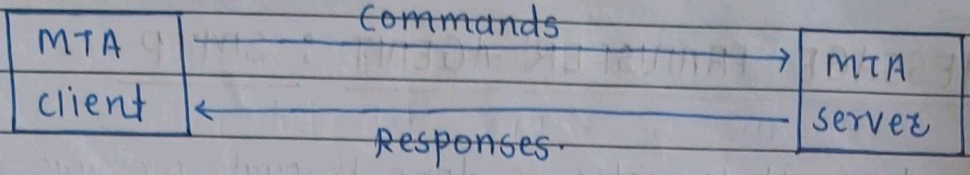


Fig. 23.9. Commands & Responses.

Each command or reply is terminated by a two-character (carriage return & line feed) end-of-line token.

1] Commands -

Commands are sent from the client to the server. The format of a command.

* Keyword : argument (s) -

* Table 23.1 -

Keyword	Argument (s)	Keyword	Argument (s)
HELO	Sender's host name.	NOOP	
MAIL FROM	Sender of the message.	TURN	
RCPT TO	Intended recipient	EXPN	mailing list.
DATA	Body of the mail.	HELP	Command name.
QUIT		SEND FROM	Intended recipient.
RESET		SMOL FROM	Intended recipient
VRFY	Name of recipient	SMAL FROM	Intended recipient.

1] HELO - This command is used by the client to identify itself. The argument is the domain name of the client host.

Format :- HELO : challenger.atc.fhda.edu

MAIL FROM - This command is used by the client to identify the sender of the message. The argument is the e-mail address of the sender.

Format :- MAIL FROM : forouzan@challenger.atc.fhda.edu

3] RCPT TO - This command is used by the client to identify the intended recipient of the message. The argument is the e-mail address of the recipient. If there are multiple recipients, the command is repeated.

Format :- RCPT TO : betsy@mcgraw-hill.com.

4] DATA :- This command is used to send the actual message. The message is terminated by a line containing just one period.

Format :- DATA
This is the message.
to be sent to the McGraw-Hill
company.

5] QUIT - The command terminates the message.

Format :- QUIT

6] RSET - The command aborts the current mail transaction. The stored information about the sender & recipient is deleted.

Format :- RSET.

7] VRFY - This command is used to verify the address of the recipient, which is sent argument.

Format :- VRFY : betsy@mcgraw-hill.com.

8] NOOP - This command is used by the client to chek the status of the recipient. It requires an answer from the recipient.

Format :- NOOP.

9] TURN - This command lets the sender & the recipient switch positions, whereby the sender becomes the recipient & vice versa.

Format :- TURN.

10] EXPN - This command asks the receiving host to expand the mailing list sent as the arguments & to return the mailbox addresses of recipients that comprise the list.

Format :- EXPN : x y z

11] HELP - This command asks the recipient to send information about the command sent as the argument.

Format :- HELP : mail.

12] SEND FROM - This command specifies that the mail is to be delivered to the terminal of the recipient, & not the mailbox.

Format :- SEND FROM : forouzan@fhda.atc.edu.

13] SMOL FROM - This command specifies that the mail is to be delievered to the terminal or the mail box of the recipient.

Format :- SMOL FROM : forouzan @ fhda.atc.edu.

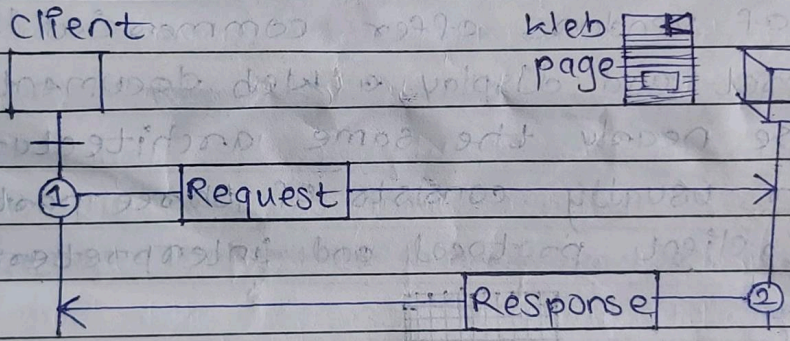
14] SMAL FROM - This command specifies that the mail is to be delie-vered to the terminal & the mailbox recipient. argument is the address of the sender. Format :- SMAL FROM : forouzan @ fhda.atc.edu.

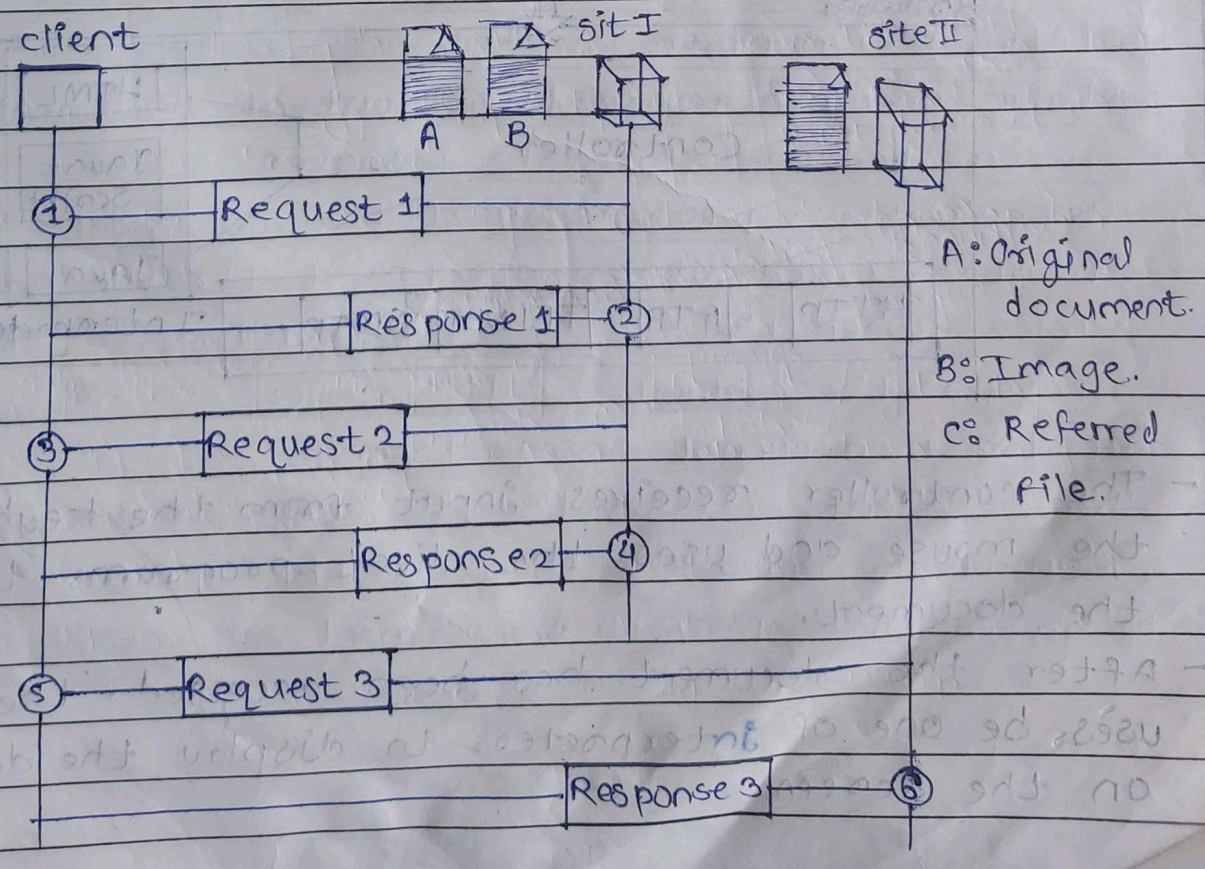
5. Web Application & Service Protocols

D D M M Y Y Y Y

* Architecture of HTTP :—

- The WWW today is a distributed client-user service, in which a client using a browser can access a service using a server.
- However, the service provided is distributed over many locations called sites.
- Each site holds one or more documents referred to as Web pages.
- However, can contain some links to other web pages in the same or other sites. In other words, a web page can be simple or composite.
- A simple web page has no link to other web pages; a composite web page has one or more links to other web pages. Each web page is a file with a name & address.





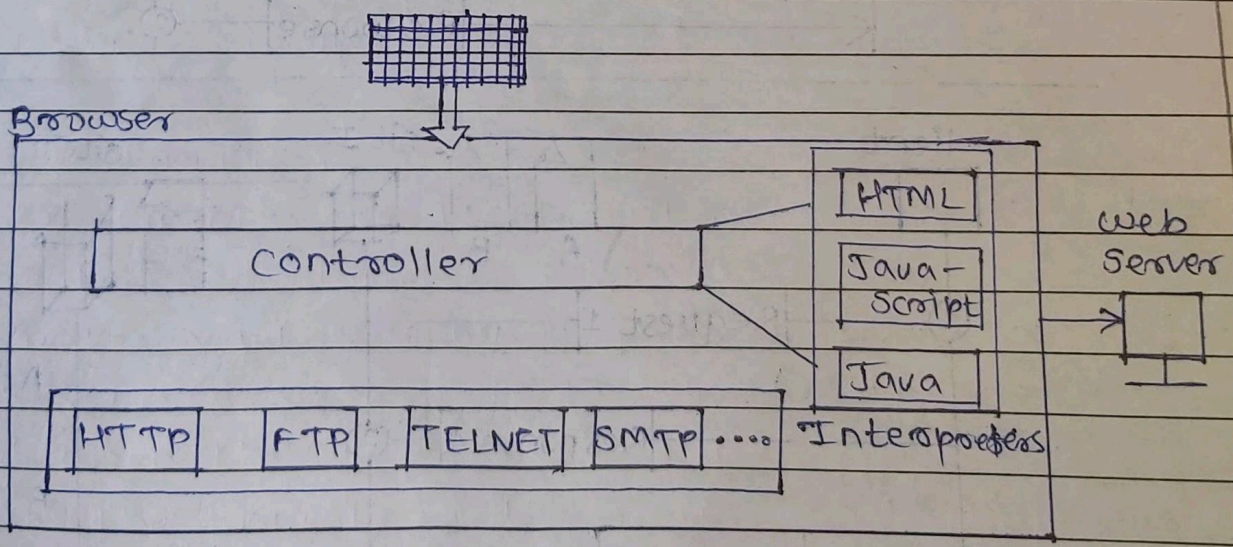
(1) Hypertext and Hypermedia :—

- **Hypertext :-**
  - Hypertext means creating documents that refer to other documents.
  - In Hypetext document, a part of text can be defined as a link to another document.
  - When a hypertext is viewed with a browser, the link can be clicked to retrieve the other document.

- **Hypermedia :-**
  - Hypermedia is a term applied to document that contains links to other textual document or documents containing graphics, video, or audio.

(2) Web Client (Browser) :—

- A variety of vendors offer commercial "browsers" that interpret and display a Web document, and all of them use nearly the same architecture.
- Each browser usually consists of three parts: a controller, client protocol and interpreters.



- The controller receives input from the keyboard or the mouse and uses the client programs to access the document.
- After the document has been accessed, the controlle uses be one of interpreteas to display the document on the screen.

- client protocol can be one of the protocols described previously such as FTP, or TELNET, or HTPP.
- The interpreter can be HTML, Java, or JavaScript, depending on the type of document.

(3) Web Server :-
- The web page is stored at the server, each time a client request arrives, the corresponding document is sent to the client.
- To improve efficiency, servers normally store request -ed files in a cache in memory; memory is faster to access than disk.
- A server can also become more efficient through multithreading or multiprocessing. In this case, a server can answer more than one request at a time. Some popular web servers include Apach and microsoft Internet Information Server.

(4) Uniform Resource Locator (URL) :-
- A client that wants to access a web page needs the file name and the address. To facilitate the access of documents distributed throughout the world.
- HTTP uses locators. The uniform resource locator (URL) is a standard locator for specifying any kind of information on the Internet.
- The URL defines four things & protocol, host computer, port and path.

URL-

Protocol	://	Host	:	Port	/	Path

- protocol is the client-server application program used to retrieve the document.
- Many different protocols can retrieve a document; among them are Gopher, FTP, HTTP, News, and TELNET. The most common today is HTTP.

- The host is the domain name of the computer on which the information is located.
- Web pages are usually stored in computers. & computers are given domain name aliases that usually ~~stored~~ begin with the characters "www".
- This is not mandatory, however, as the host can have any domain name.
- The URL can optionally contain the port number of the server. If the port is included, it is inserted bet the host and the path, and it is separated from the host by a colon.
- Path is the pathname of the file where the information is located. Note that the path can itself contain slashes. that in the UNIX operating system, separate the direct -ories from the subdirectories and files.

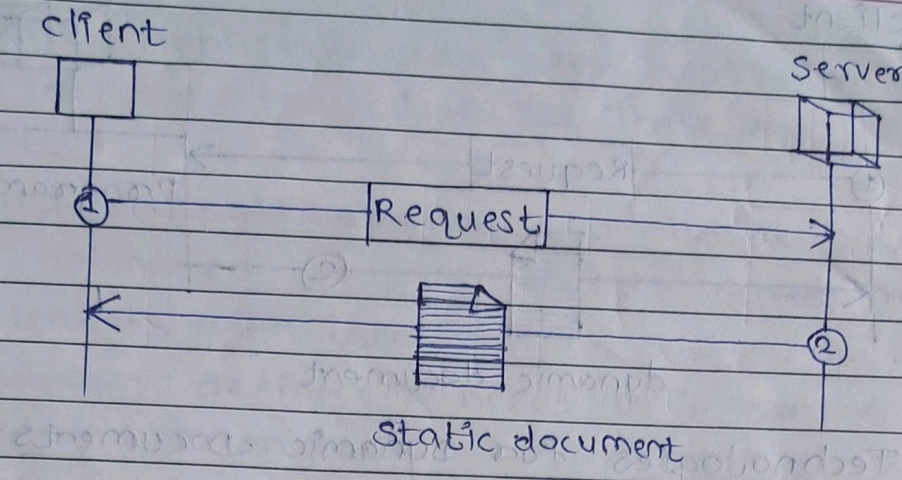
---

* Web Documents :—

- The documents in the WWW can be grouped into three broad categories: static, dynamic and active.
- The category is based on the time the contents of doc. -ument are determined.

(1) Static Documents :—

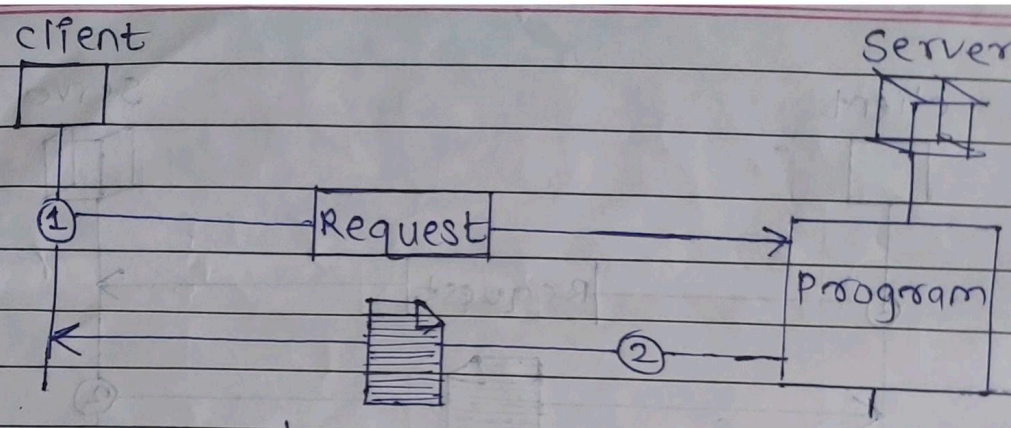
- Static documents are fixed-content documents that are created and stored in a server.
- The client can get a copy of the document only.
- In other words, the contents of the file are deter -mined when the file is created, not when it is used Ofcourse, the contents in the server can be changed, but the user cannot change them.
- When a client accesses the document, a copy of the document is sent. The user can then use a browsing program to display the document.



- Static documents are prepared using one of the several languages: Hypertext Markup Language (HTML), Extensible Markup language (XML), Extensible style Language (XSL), and Extended Markup Languages (XHTML). We discuss these languages in Appendix E.

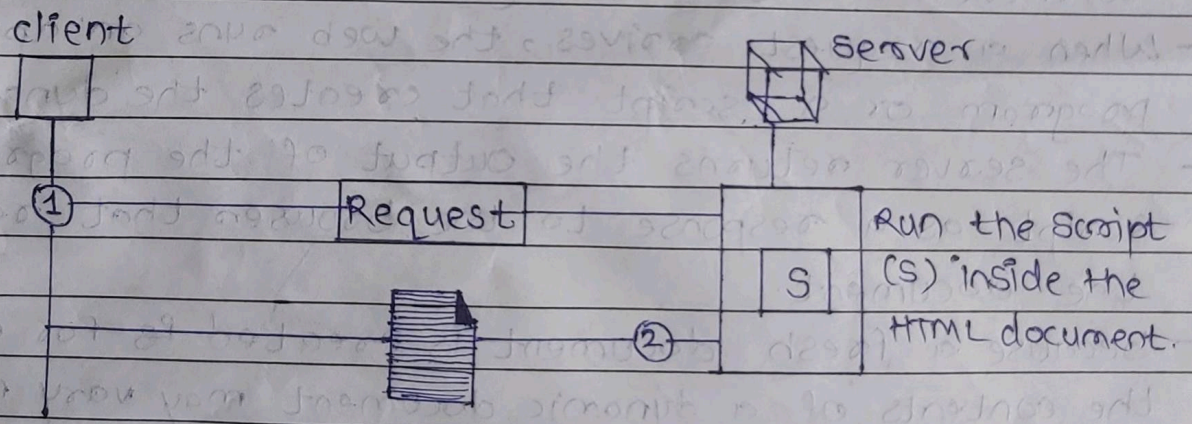
(2) Dynamic Documents:—

- A dynamic document is created by a web server whenever a browser requests the document.
- When a request arrives, the web runs an application program or a script that creates the dynamic doc.
- The server returns the output of the program or script as a response to the browser that requested the document.
- Because a fresh document is created fs for each request, the contents of a dynamic document may vary many from one request to another.
- simple example of a dynamic document is the retrieval of the time and date from a server.
- Time and date are kinds of information that are dynamic in that they change from moment to moment.
- The client can ask the server to run a program such as the date program in UNIX & send the result of the program to the client.



dynamic document

- Scripting Technologies for Dynamic Documents:
  - The problem with CGI technology is the inefficiency that results if part of the dynamic document that is to be created is fixed and not changing from request to request.
  - example, assume that we need to retrieve a list of spare parts, their availability, & prices for a specific car brand.
  - Although the availability and prices vary from time to time, the name, description, and picture of the part are fixed.



Dynamic document.

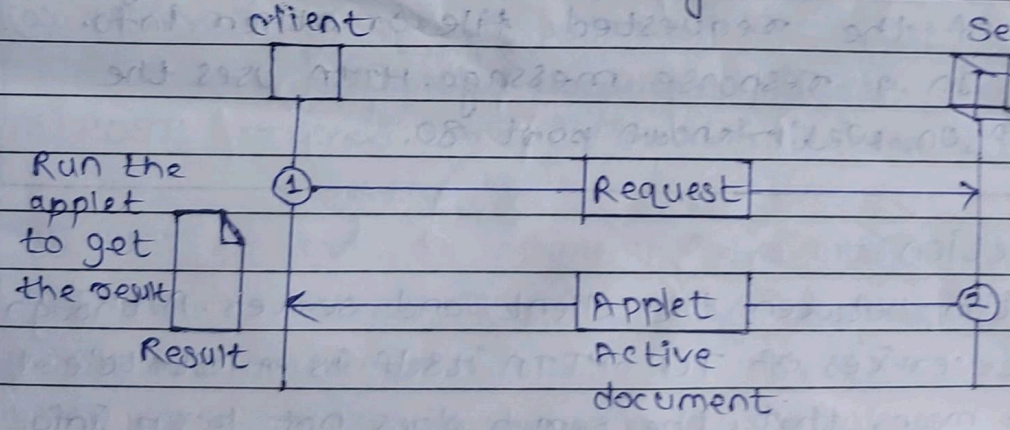
  - Among the most common are Hypertext Preprocessor (PHP), Perl language; Java Server Pages (JSP), which uses the Java language for scripting; Active Server Pages (ASP), a Microsoft product, which uses Visual Basic language for scripting; and ColdFusion, which embeds SQL database queries in the HTML document.

(3) Active Documents :-

- For many Applications, we need a program or a script to be run at the client site. These are called active documents.
- For example, suppose we want to run a program that creates animated graphics on the screen or a program that interacts with users.
- The program definitely needs to be run at the client site where the animation or interaction takes place.
- When a browser requests an active document, the server sends a copy of the document or a script. document is then run at the client (browser) site.

• Java Applets —

- One way to create an active document is to use Java applets. Java is a combination of a high-level programming Language.
- An applet is a program written in Java on the server. It is compiled and ready to run.



• Java Script :-

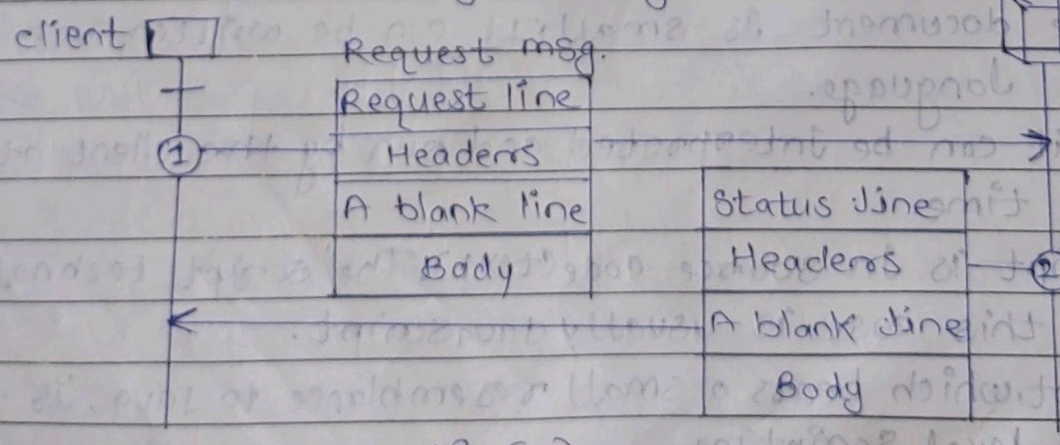
- The idea of scripts in dynamic documents can also be used for active documents. If the active part of the document is small, It can be written in a scripting language.
- Then it can be interpreted and run by the client at the same time.
- The script is in source code (text). The script technology used in this case is usually JavaScript.
- JavaScript, which bears a small resemblance to Java, is a very high level scripting.

* HTTP (Hypertext Transfer Protocol):-

- The Hypertext Transfer Protocol (HTTP) is a protocol used mainly to access data on the world wide web.
- HTTP functions like a combination of FTP & SMTP. It is similar to FTP because it transfers files & uses the services of TCP.
- However, it is much simpler than FTP because it uses one TCP connection. There is no separate control connect. only data are transferred bet^n the client & the server.
- HTTP is like SMTP because the data transferred bet^n the client and the server look like SMTP messages.
- In addition, the format of the messages is controlled by MIME-like headers. Unlike SMTP, the HTTP messages are not destined to be read by humans; they are read and interrupt by the HTTP server and HTTP client (browser).
- SMTP messages are stored and forwarded, but HTTP msg are delivered immediately. The commands from the client to the server are embedded in a request msg.
- The contents of the requested file or other info. are embedded in a response message. HTTP uses the services of TCP on well-known port 80.

* HTTP Transaction:-

- HTTP transaction between the client and server. Although HTTP uses the services of TCP, HTTP itself is a stateless protocal, which means that the server does not keep information about the client.
- The client initializes the transaction by sending a request. The server replies by sending a response.



Fig(a)

Request Message:—

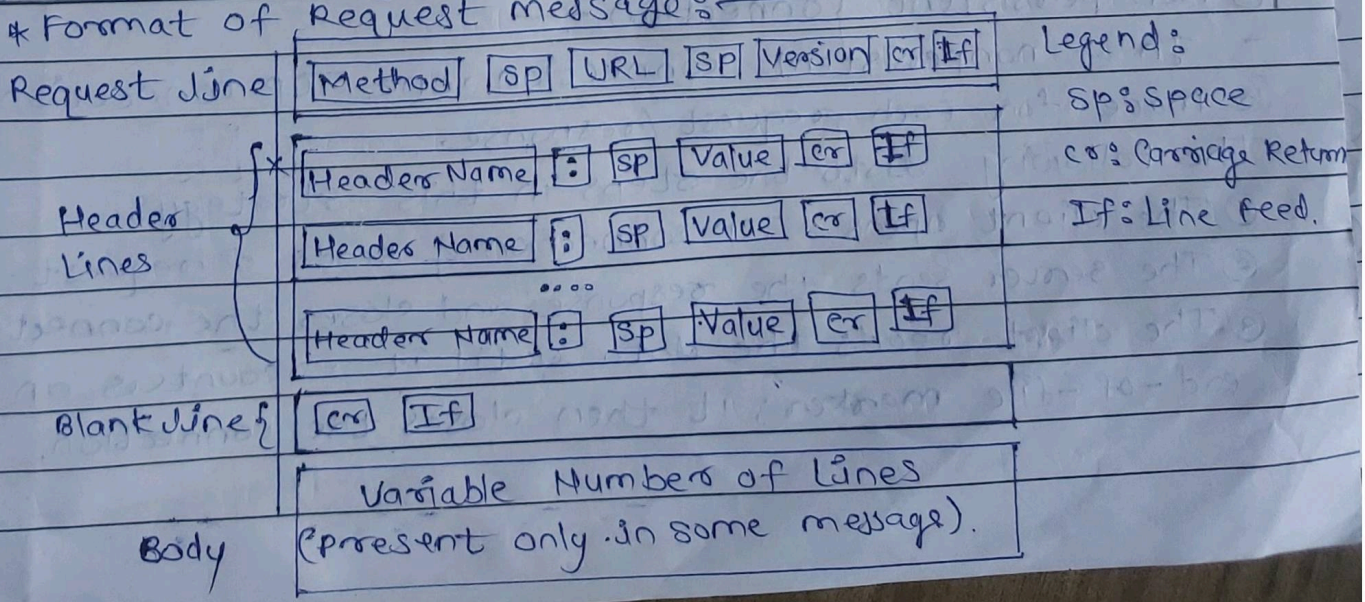
The format of the request is fig(a). A request message consists of a request line, a header, and sometimes a body.

▸ Request line:—

- The first line in a request message is called a request line. There are three fields in this line separated by character delimiter as in fig(a).
- The fields are called methods, URL & version. These 3 should be separated by a space character.
- At the end two character, a carriage return followed by a line feed terminate the line.
- The methods field defines the request type. In version 1.1 of HTTP several methods are defined.

Methods:	Action
(1) GET	– Requests a document from the server.
(2) HEAD	– Requests information about a document but not the document itself.
(3) POST	– Sends some information from the client to the server.
(4) PUT	– Sends a document from the server to the client
(5) TRACE	– Echoes the incoming request.
(6) CONNECT	– Reserved.
(7) DELETE	– Remove the web page.
(8) OPTIONS	– Enquires about available options.

* Format of Request message:—

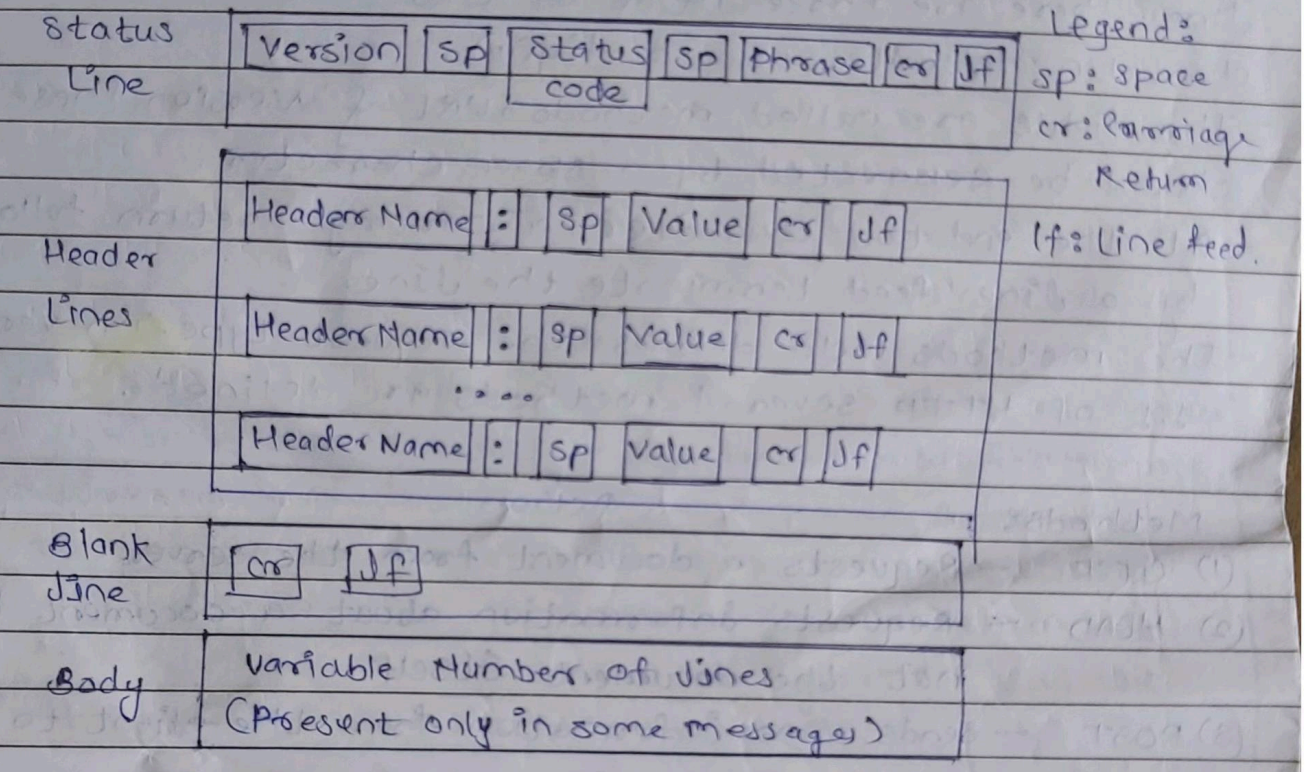


▸ Response Message:—

A response message consists of a status line, header ~~files~~ lines, a blank line and sometimes a body.

↳ Status line: The first line in a response message is called the status line.

Format of Response Message:—



Persistence:—

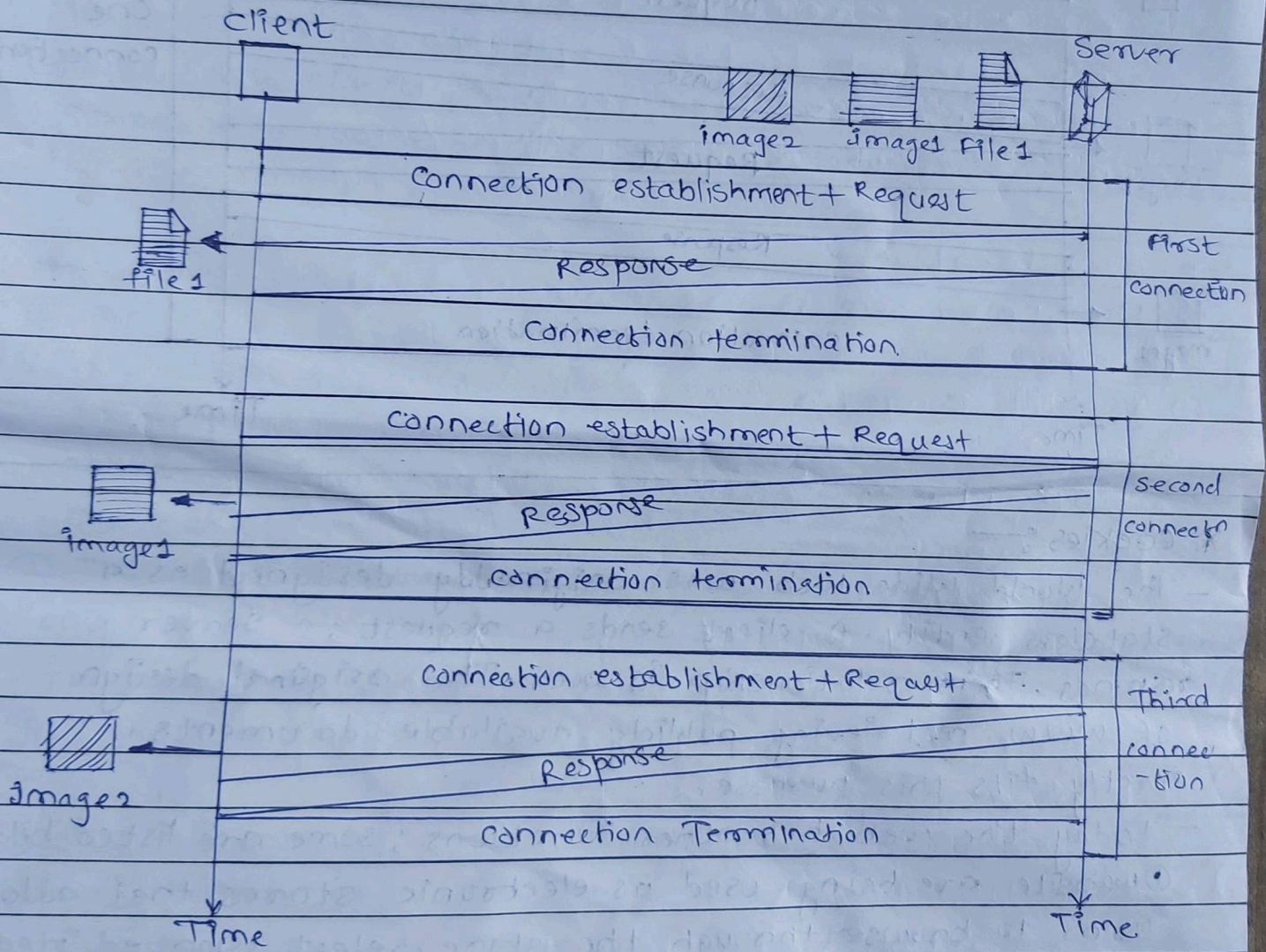
HTTP, prior to version 1.1, specified a nonpersistent connection, while a persistent connection is the default in version 1.1.

① Non-persistent connection —

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

- In this strategy, if a file contains link to N different pictures in different files (all located on the same server), the connection must be opened & closed N+1 times.
- The nonpersistent strategy imposes high overhead on the server because the server needs N+1 diffn buffers & requires a slow start procedure each time a connectn is opened.

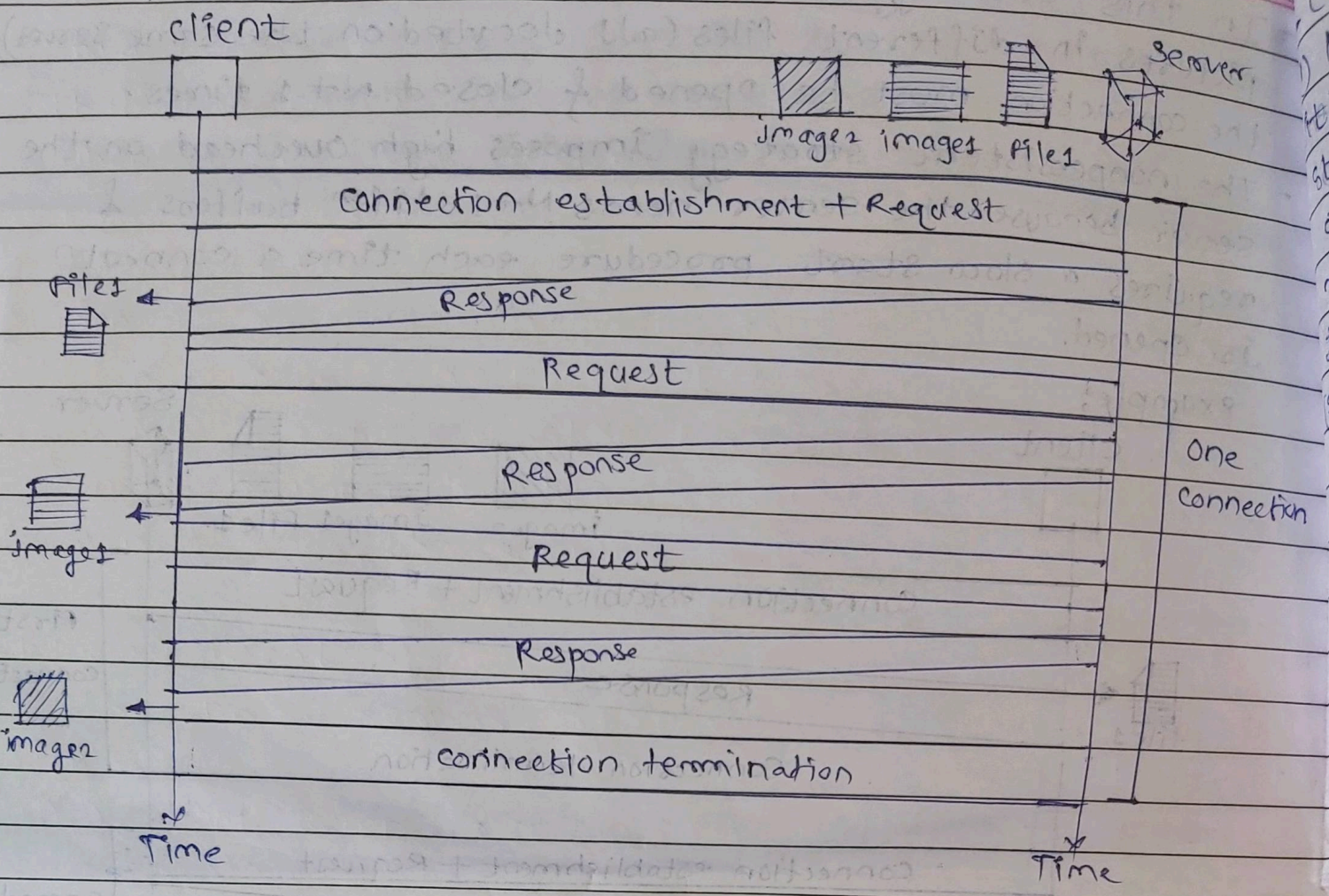
example:



② Persistent connection :—

- HTTP version 1.1 specifies a persistent connection by default. In a persistent connection, the server leaves the connection open for more requests of a client or if a time out has been reached.
- The senders usually sends the length of the data with each response.

Example:-



* Cookies :-

- The World Wide Web was originally designed as a stateless entity. A client sends a request; a server responds. Their relationship is over. The original design of www, retrieving publicly available documents, exactly fits this purpose.
- Today the web has other functions; some are listed below:

① website are being used as electronic stores that allow users to browse through the store, select wanted items, put them in an electronic cart, and pay at the end with a credit card.

② some websites need to allow access to registered clients only.

③ some websites are used as portals: user selects the web pages he want to see.

④ some websites are just advertising.

► Creating and storing cookies:—

(1) When a server receives a request from a client, it stores information about the client in a file or a string. The information may include the domain name of the client, the contents of the cookie (such as name, registration number, and so on), a timestamp, and other information depending on the implementation.

(2) The server includes the cookie in the response that it sends to the client.

(3) When the client receives the response, the browser stores the cookie in the cookie directory, which is stored by the domain server name.

► Using Cookies:—

(1) An electronic store (e-commerce) can use a cookie for its client shoppers. When a client selects an item & insert it into a cart, a cookie that contains info. about the item, such as it's number/price is sent to the browser.

(2) The site that restricts access to registered clients only sends a cookie to the client when the client registers for the first time.

(3) A web portal uses the cookie in a similar way.

(4) A cookie is also used by advertising agencies. The advertising agency supplies only a URL that gives the banner address instead of the banner itself.

* Proxy Server: (Web Caching).

- HTTP supports proxy servers. A proxy server is a computer that keeps copies of responses to recent requests.
- The HTTP client sends a request to the proxy server.
- The proxy server checks its cache. If the response is not stored in the cache, the proxy server & stored for future requests from other clients.

- The proxy server reduces the load on the original server, decreases traffic, and improves latency.
- that the proxy server acts both as a server & client. when it receives a request from a client for which it has a response, it acts as a server & sends the response to the client.
- when it receives a request from a client for which it does not have a response, it first acts as a server & sends a request to the target server.
- when the response has been received, it acts again as a server & sends the response to the client.

▶ Proxy Locations:-

The proxy servers are normally located at the client site.

(1) A client computer can also be used as a proxy server server in a small capacity that stores responses to requests often invoked by the client.

(2) In a company, a proxy server may be installed on the computer LAN to reduce the load going out of & coming into the LAN.

(3) An ISP with many customers can install a proxy server to reduce the load going out of and coming into the ISP network.

▶ Cache Update:-

One solution is to store the list of sites whose information remains the same for a while.

e.g: a news agency may change its news page every morning. This means that a proxy server can get the news early in the morning & keep it until the next day.

Ch. 6. MULTIMEDIA

D D M M Y Y Y Y

We can divide audio and video services into three broad categories :-

1] Streaming stored audio / video
2] Streaming live audio / video
3] Interactive audio / video

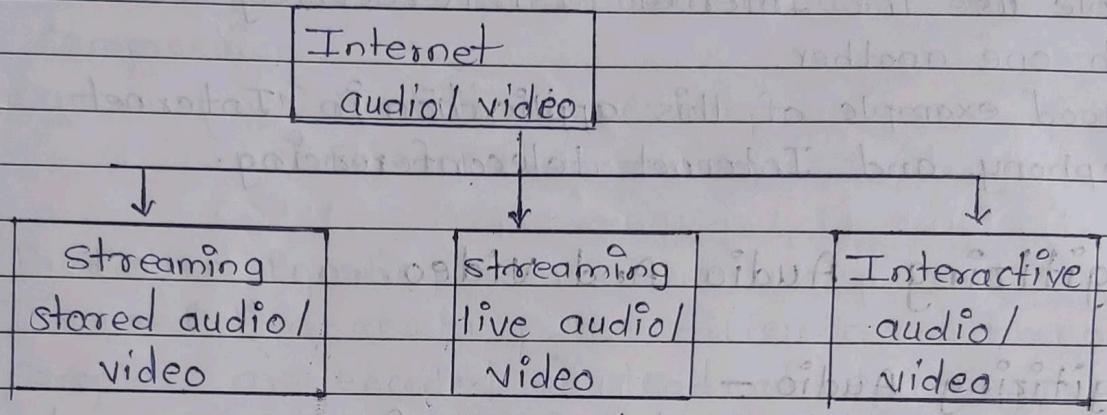


Fig. Internet audio / video

1] Streaming stored audio / video -

- In the first category, streaming stored audio/ video. the files are compressed and stored on a server
- A client downloads the files through the Internet This is sometimes referred to as on-demand audio / video.
- Examples of stored audio files are songs, symphonies, books on tape, and famous lectures.
- Example of stored video files are movies, TV shows, and music video clips.

2] Streaming live audio / video -

- In the second category, streaming live audio / video, a user listens to broadcast audio and video through the Internet.
- A good example of this type of application is the Internet radio. Some radio stations broadcast their programs only on the Internet; many broadcast them both on the Internet and on the air.

- Internet TV is not popular yet, but many people believe that TV stations will broadcast their program on the Internet in the future.

3] Interactive audio / video -

- In the third category, interactive audio / video, people use the Internet to interactively communicat with one another.
- A good example of this application is Internet telephony and Internet teleconferencing.

* Digitizing Audio And Video.

Digitizing Audio -

- When sound is fed into a microphone, an electroni analog signal is generated that represents the sound amplitude as a function of time. The signal is called an analog audio signal.
- An analog signal, such as audio, can be digitized to produce a digital signal.

Digitizing Video -

- A video consist of a sequence of frames.
- If the frames are displayed on the screen fast enough, we get an impression of motion.
- The reason is that our eyes cannot distinguish the rapidly flashing frames as individual ones.
- There is no standard number of frames per second; in north America 25 frames per second is common. to avoid a condition known as flickering, a frame need to be refreshed.
- The TV industry repaints each frame twice. This means 50 frames need to be sent, or if their is memory at sender site. 25 frames with each frame repainted from the memory.

AUDIO AND VIDEO COMPRESSION

Audio Compression -

- Audio compression can be used for speech or music.
- for Speech, we need to compress a 64-kHz digitized signal.
- for music, we need to compress a 1.411 MHz signal.
- Two categories of techniques are used for audio compression.
    1) Predictive encoding
    2] Perceptual encoding.

1) Predictive Encoding -

- In predictive encoding, the differences between the samples are encoded instead of encoding all the sampled values.
- This type of compression is normally used for speech.
- Several standards have been defined such as GSM (13 kbps), G.729 (8 kbps) and G.723.3 (6.4 or 5.3 kbps).

2] Perceptual Encoding : MP3

- The most common compression technique that is used to create CD-quality audio is based on the perceptual encoding technique.
- MP3 (MPEG audio layer 3), a part of the MPEG standard (discussed in the video compression section) uses this technique.
- Perceptual encoding is the study of how people perceive sound.
- In frequency masking, a loud sound in a frequency range can partially or totally mask softer sound in another frequency range.
- In temporal masking, a load sound can numb our ears for a short time even after the sound has ceased.

- MP3 uses these two phenomena, frequency & temporal masking, to compress audio signals.
- The technique analyzes and divides the spectrum into several group.

Video Compression -

- Video is composed of multiple frames. Each frame is one image.
- We can compress video by first compressing images.
- Two standards are prevalent in the market.
    1) Joint Photographic Experts (JPEG)
    2) Moving Picture Experts Group (MPEG)

1] Image Compression : JPEG.

- If the picture is not in color (gray scale), each pixel can be represented by an 8-bit integer.
- If the picture is in colour, each pixel can be represented by 24 bits, with each 8-bits representing red, blue, or green.

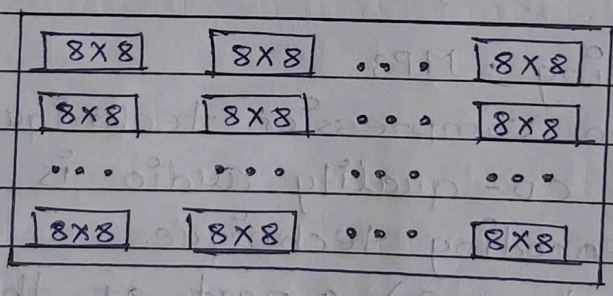
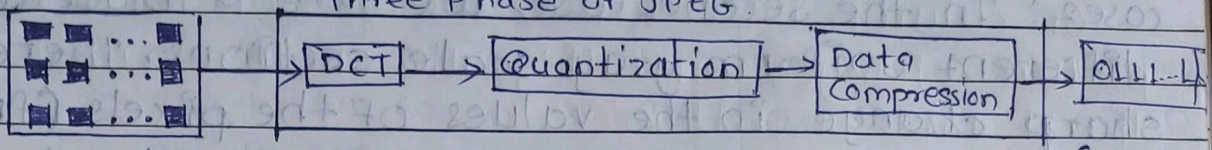


Fig. JPEG gray scale.

* JPEG Process

- The purpose of dividing the picture into blocks is to decrease the no. of calculations because, as you will see shortly, the no. of mathematical operations for each picture
- The whole idea of JPEG is to change the picture into linear (vector) set of numbers that reveals the redundancies.

Three Phase of JPEG

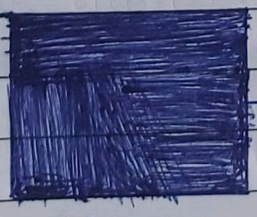


Discrete Cosine Transform (DCT):-

In this step, each block of 64 pixels goes through a transformation called the discrete cosine transform (DCT). The transformation changes the 64 values so that the relation relationships between pixels are kept but the redundancies are revealed. We do not give the formula here, but we do show the results of the transformation for three cases.

case 1: In this case, we have a block of uniform gray, and the value of each pixel is 20. When we do the transformations, we get a non-zero value for the first element (upper left corner); the rest of the pixels have a value T(0,0) is the average (multiplied by a constant) of the P(x,y) values and is called the dc value (direct current, borrowed from electrical engineering). The rest of the values, called ac values, in T(m,n) represent changes in the pixel values, But because there are no changes, the rest of the values are 0's.

case 1: Uniform gray scale

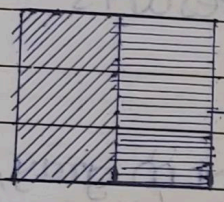


Picture

P(x,y)	T(m,n)
20 20 20 20 20 20 20 20	160 0 0 0 0 0 0 0
20 20 20 20 20 20 20 20	0 0 0 0 0 0 0 0
20 20 20 20 20 20 20 20	0 0 0 0 0 0 0 0
20 20 20 20 20 20 20 20	0 0 0 0 0 0 0 0
20 20 20 20 20 20 20 20	0 0 0 0 0 0 0 0
20 20 20 20 20 20 20 20	0 0 0 0 0 0 0 0
20 20 20 20 20 20 20 20	0 0 0 0 0 0 0 0
20 20 20 20 20 20 20 20	0 0 0 0 0 0 0 0

case 2: In the second case, we have a block with two different uniform gray scale sections. There is a sharp change in the values of the pixels (from 20 to 50). When we do the transformations, we get a dc value as well as non-zero ac values. However, there are only a few non-zero values clustered around the dc value. Most of the values are 0.

Case 2: two sections



Picture

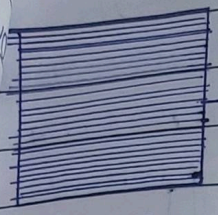
P(x,y)		T(m,n)
20 20 20 20 50 50 50 50		280 2109 0 39 0 225 0 22
20 20 20 20 50 50 50 50		0 0 0 0 0 0 0 0
20 20 20 20 50 50 50 50		0 0 0 0 0 0 0 0
20 20 20 20 50 50 50 50	→	0 0 0 0 0 0 0 0
20 20 20 20 50 50 50 50		0 0 0 0 0 0 0 0
20 20 20 20 50 50 50 50		0 0 0 0 0 0 0 0
20 20 20 20 50 50 50 50		0 0 0 0 0 0 0 0
20 20 20 20 50 50 50 50		0 0 0 0 0 0 0 0

case 3: In the third case, we have a block that changes gradually. That is, there is no sharp change between the values of neighboring pixels. When we do the transformations, we get a dc value, with many non-zero ac values also.

We can say the following:

1) The transformation creates table T from table P.
2) The dc value is the average value (multiplied by a const.) of the pixels.
3) The ac values are the changes.
4) Lack of changes in neighboring pixels creates 0s.

case 3:- gradient gray scale



picture

20	30	40	50	60	70	80	90
20	30	40	50	60	70	80	90
20	30	40	50	60	70	80	90
20	30	40	50	60	70	80	90
20	30	40	50	60	70	80	90
20	30	40	50	60	70	80	90
20	30	40	50	60	70	80	90
20	30	40	50	60	70	80	90

P(x,y)

400	2146	0	231	21	3	21	28
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

T(m,n)

Quantization :-

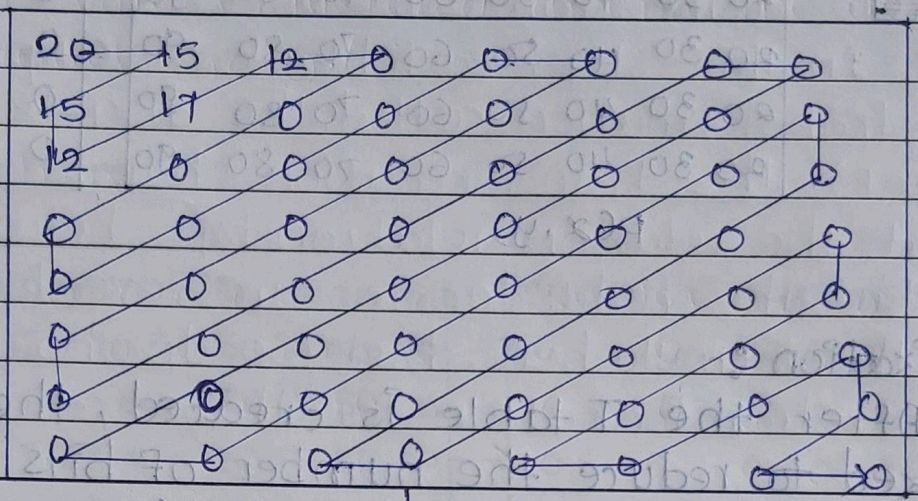
After the T table is created, the values are quantized to reduce the number of bits needed for encoding. Previously in quantization, we dropped the fraction from each value and kept the integer part. Here, we divide the number by a constant and then drop the fraction. This reduces the required number of bits even more. In most implementations, a quantizing table (8 by 8) defines how to quantize each value. The divisor depends on the position of the value in the T table. This is done to optimize the number of bits and the number of 0's for each particular application. Note that the only phase in the process that is not completely reversible is the quantizing phase. We lose some information here that is not recoverable. As a matter of fact, the only ~~person~~ reason that JPEG is called lossy compression is because of this quantization phase.

- compression :-

After quantization, the values are read from the table, and redundant 0s are removed. However, to cluster the 0's together, the table is read diagonally in a zigzag fashion rather than row by row

or column by column. The reason is that if the picture changes smoothly, the bottom right corner of the T table is all 0's.

Fig. Reading the table



Video compression: MPEG :-

The moving picture Experts Group (MPEG) method is used to compress video. In principle, a motion picture is a rapid flow of a set of frames, where each frame is an image. In other words, a frame is a spital combination of pixels and a video is a temporal combination of frames that are sent one after another. Compressing video, then means spatially compressing each frame and temporally compressing a set of frames.

spital compression :-

The spital compression of each frame is done with JPEG (or modification of it). Each frame is a picture that can be independently compressed.

Temporal compression:-

In temporal compression, redundant frames are removed. When we watch television, we receive 50 frames per second. However, most of the consecutive frames are almost the same. For example, when someone is talking, most of the frame is the same as the previous one except for the segment of the frame around the lips, which changes from one frame to another.

To temporally compress data, the MPEG method first divides frames into three categories:

1) I-frames
2) P-frames
3) B-frames

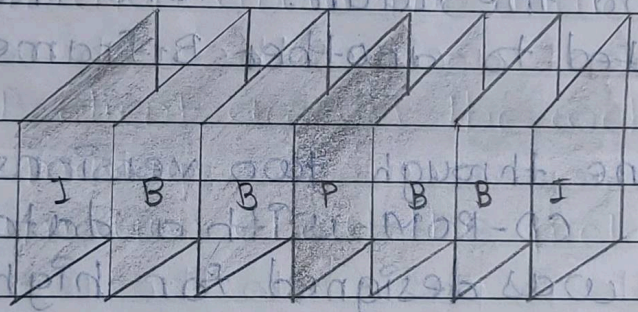
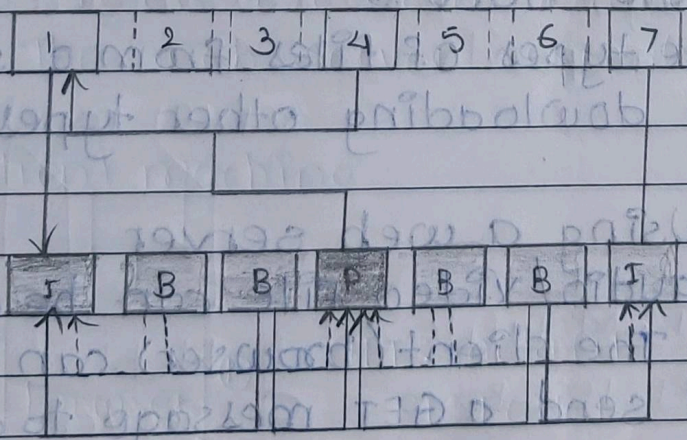


Fig. MPEG frames



- I-frames :- An intracoded frame (I-frame) is an independent frame that is not related to any other frame. They are present at regular intervals. An I-frame must appear periodically to handle some sudden change in the frame that the previous and following

frames cannot show. I-frames are independent of other frames and cannot be constructed from other frames.

- P-frames :- A predicted frame (P-frame) is related to the preceding I-frame or P-frame. In other words, each P-frame contains only the changes from the preceeding frame. The changes, however, cannot cover a big segment.

- B-frames :- A bidirectional frame (B-frame) is related to the preceding and following I-frame or P-frame. In other words, each B-frame is relative to the past and the future. Note that a B-frame is never related to another B-frame.

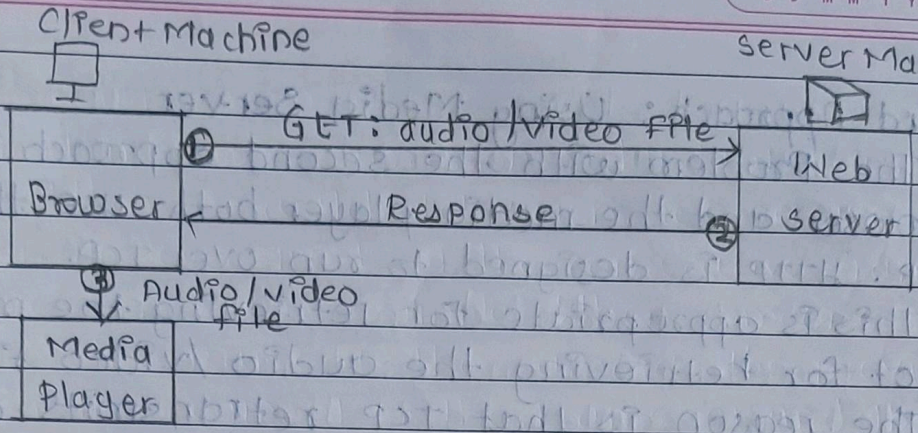
MPEG has gone through two versions. MPEG1 was designed for a CD-ROM with a data rate of 1.5 MbPS. MPEG2 was designed for high-quality DVD with a data rate of 3 to 6 MbPS.

- Streaming Stored Audio/Video :-

Downloading these types of files from a server can be different from dowloading other types of files.

1) First Approach : Using a web server

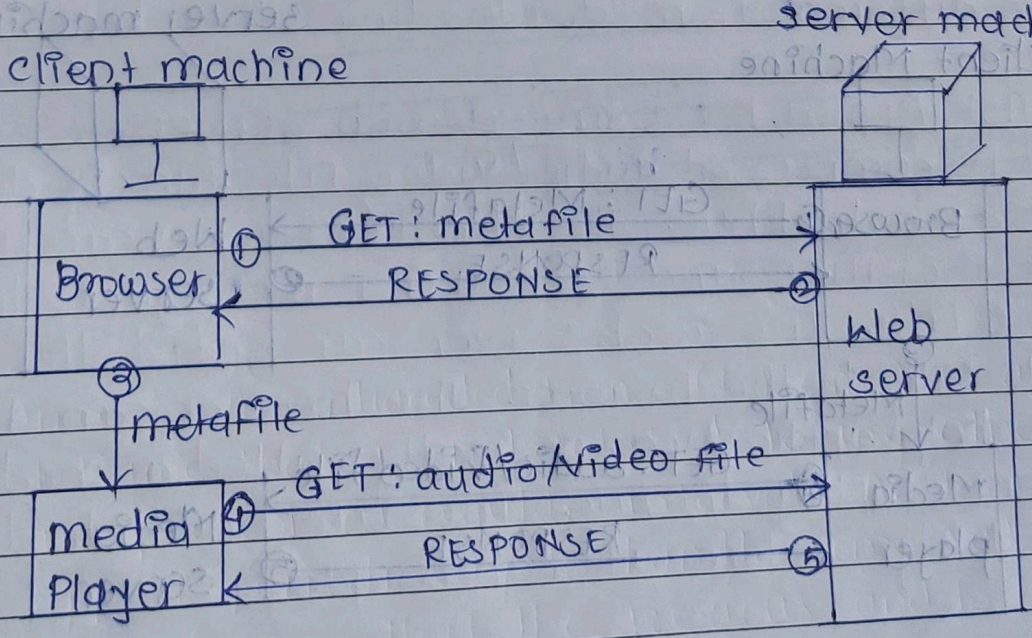
A compressed audio/video file can be downloaded as a text file. The client (browser) can use the services of HTTP and send a GET message to download the file. The web server can send the compressed file to the browser. The browser can then use a help application, normally called a media player, to play the file. The file needs to download completely before it can be played.



2) Second Approach: Using a web server with Metafile

In another approach, the media player is directly connected to the web server for downloading the audio/video file. The web server stores two files: the actual audio/video file and a metafile that holds information about the audio/video file.

1) The HTTP client accesses the web server using the GET message.
2) The information about the metafile comes in the response.
3) The metafile is passed to the media player.
4) The media player uses the URL in the metafile to access the audio/video file.
5) The web server responds.



3) Third Approach: Using Media Server

1) The Problem with the second approach is that the browser and the media player both use the services of HTTP. HTTP is designed to run over TCP.

2) This is appropriate for retrieving the metafile, but not for retrieving the audio/video file.

3) The reason is that TCP retransmits a lost or damaged segment, which is counter to the philosophy of streaming.

4) We need to dismiss TCP and it's error control; we need to use UDP.

5) However, HTTP, which accesses the web server and the web server itself are designed for TCP.

Steps :-

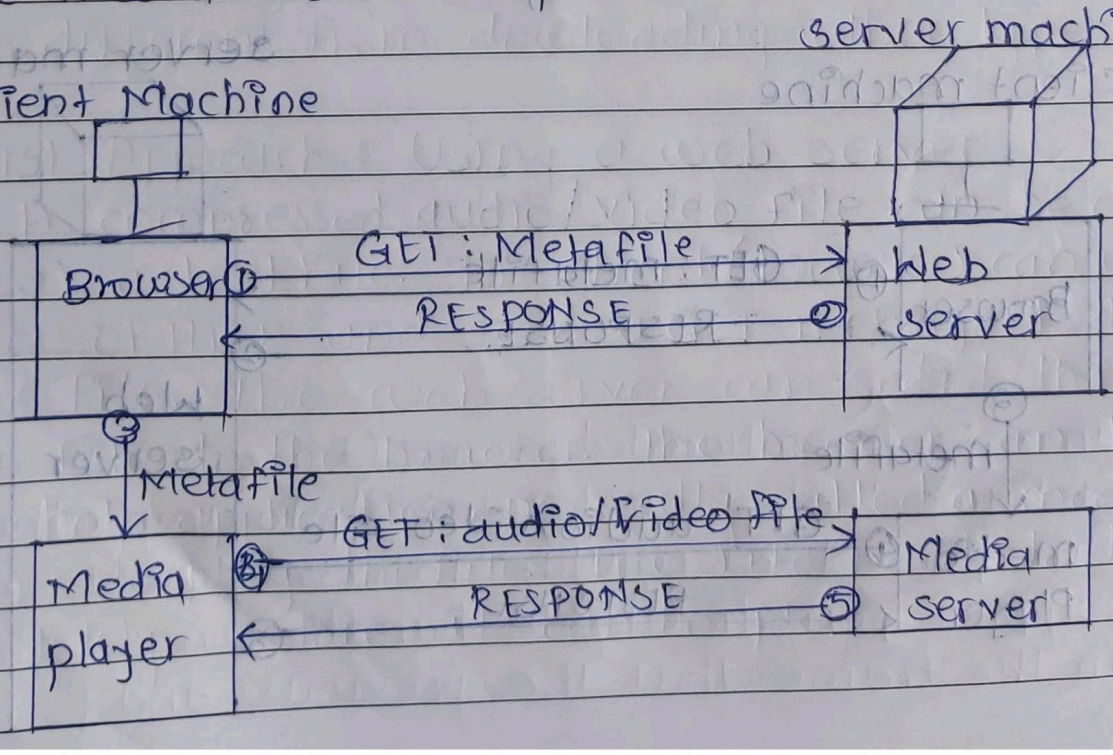
1) The HTTP client accesses the web server using a GET message.

2) The information about the metafile comes in the response.

3) The metafile is passed to the media player.

4) The media player uses the URL in the metafile to access the media server to download the file. Downloading can take place by any protocol that uses UDP.

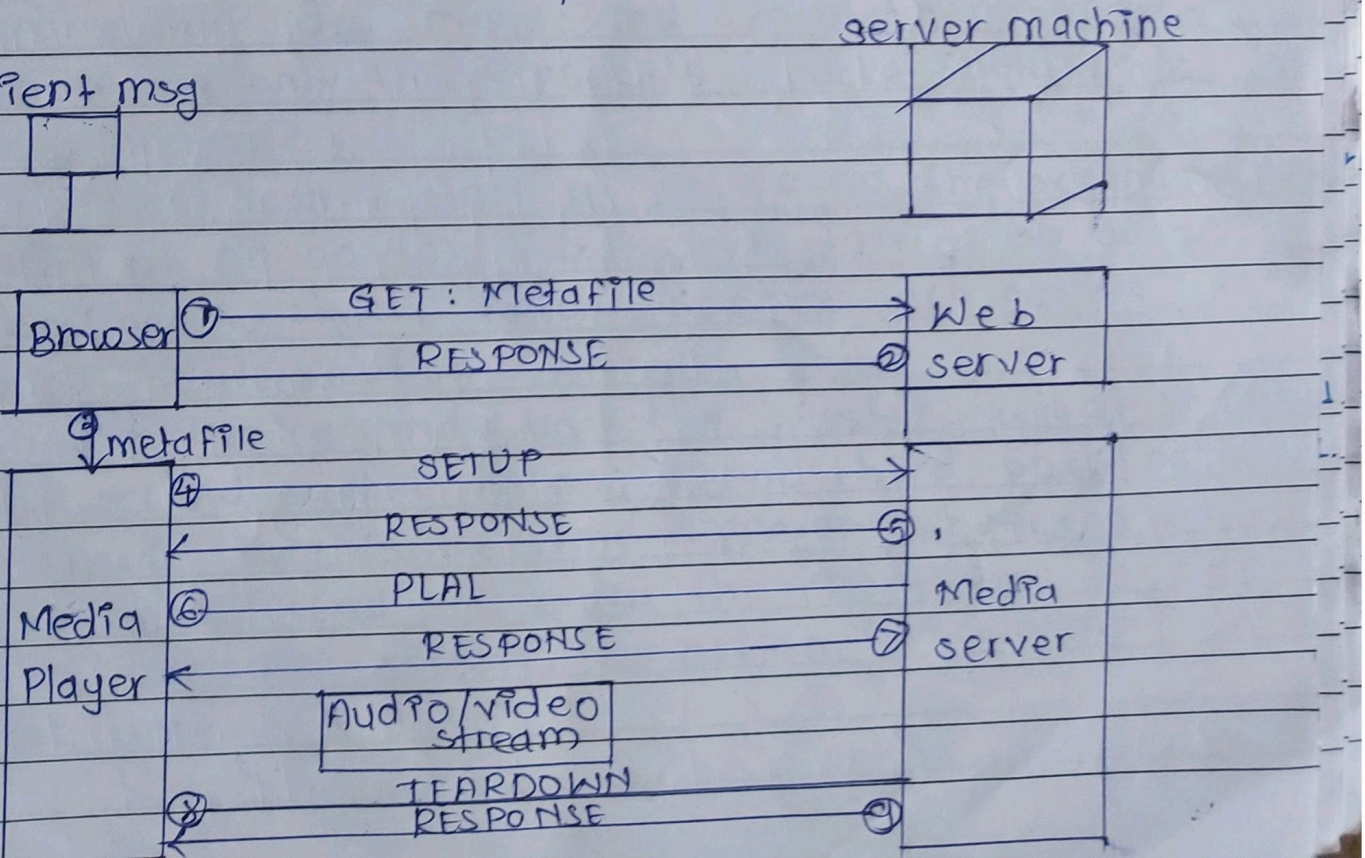
5) The media server responds.



4) Fourth Approach : Using a media server and RTSP

i) The Real time streaming Protocol (RTSP) is a control protocol designed to add more functionalities to the streaming process. Using RTSP, we can control the playing of audio/video. RTSP is an out-of-band control protocol that is similar to the second connection in FTP.

1) The HTTP client accesses the web server using a GET message.
2) The information about the metafile comes in the response.
3) The metafile is passed to the media player.
4) The media player sends a SETUP message to create a connection with the media server.
5) The media server responds.
6) The media player sends PLAY message to start playing.
7) The audio/video file is downloaded using another protocol that runs over UDP.
8) The connection is broken using the TEARDOWN msg.
9) The media server responds.



- STREAMING LIVE Audio/Video :-

i) Streaming live audio/video is similar to the broadcasting of audio and video by radio and TV stations.

ii) Instead of broadcasting to the air, the stations broadcast through the internet.

iii) There are several similarities between streaming stored audio/video.

iv) They are both sensitive to delay; neither can accept retransmission. However, there is a difference.

v) In the first application, the communication is unicast and on-demand.

vi) In the second, the communication is multicast and live.

vii) Live streaming is better suited to the multicast services of IP and the use of protocols such as UDP and RTP.

Examples: - Internet Radio, Internet Television (ITV), Internet protocol television (IPTV)

Real time Interactive Audio/Video -

In real time interactive audio/video, people communicate one another in real time. The internet phone or voice over IP an example of type of application. vide conferencing is another example that allows pepole to communicate visually and orally.

characteristics -

Before discussing the protocol used in this class of application, we discuss some characteristics of real time audio/video communication.

- **Time Relationship -**
  - Real time data on a packet switched network require the preservation of time relationship between packet of session
  - For example, Let us assume that real time video server creates live video images and sends them online.
  - There are only three packets & each packet hold 10 sec of video information.
  - The first packet start at 00:00:00, the second packet start at 00:00:10 & third packt start at 00:00:20 Also imagine that it takes 1 sec for each packet to reach the destination.
  - The reciver can play back the first packet at 00:00:01 the second packet 00:00:11 and third packt 00:00:21.
  - Although there is 1-6 time difference between what the server send and what the client ~~seen~~ sees on the computer screen, The action is happening in real time.

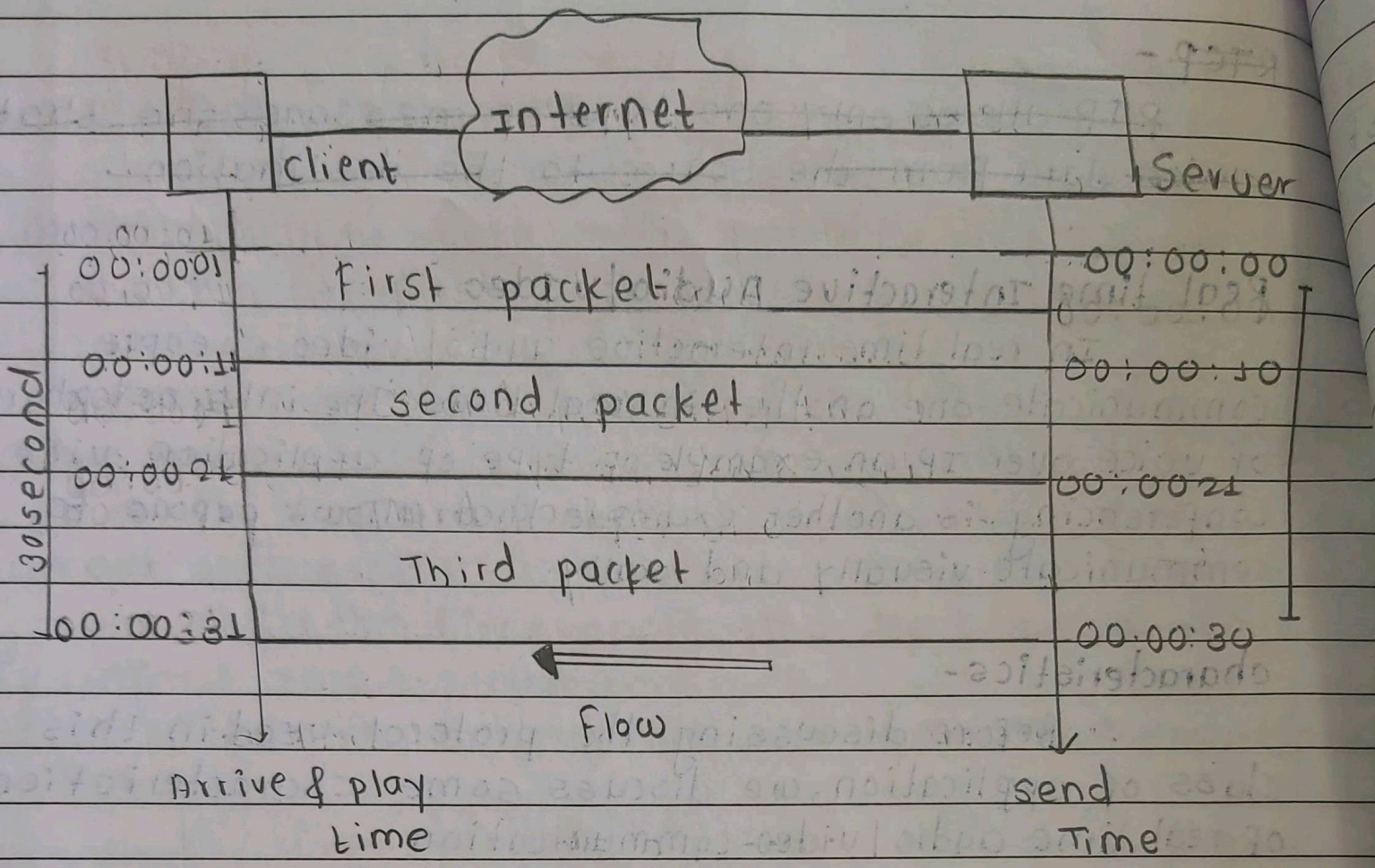


Fig. Time Relationship

- But what happen if the packet arrived with different delays? For example, the first packet arrives (00:00:01) (1-s delays), the second arrives 00:00:15 (5-s delays) and third arrives 00:00:27 (7-s delays). If reciver start playing the first packet at 00:001,
  It will finish at 00:00:11.
- However the next packet has not yet arrived it arrives 4 sec ~~letter~~ later.
- There is gap between first & second packet and between & 2nd and 3rd as the video is viewd at the remote side. This phenomenon is called as jitter.

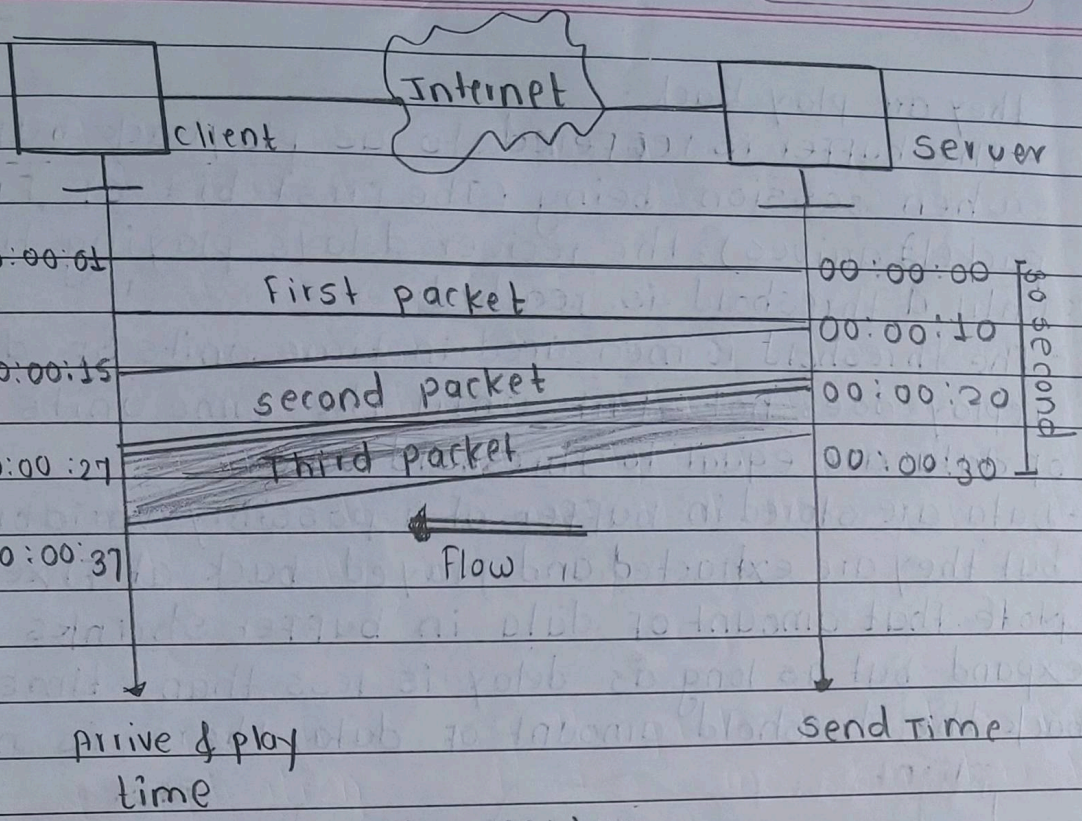


Fig- jitter

- **Timestamp-**

- one solution to jitter is the use of timestamp. IF each packet has a timestamp that shows the time it was produced relative to first (or previous) packet,
- Then the reciver can added this time to the time at which it starts the ptayback.
- In other words the reciver knows when packet is to be played.
- Imagine the first packet in the previous example has timestamp of 0, the second has a timestamp of 10, third timestamp of 20.
- The reciver start playing back the first packet at 00:00:08, the second will be played at 00:00:18, and the third at 00:00:28. There are no gaps betn the packets

- **playback buffer :-**

- To be able to seprate the arrival time from the playback time, we need buffer to store the data until

they are play back.
- The buffer is referred to as playback buffer. when session being (The first bit of first packet arrives), the reciver delays playing the data until a threshold is reached.
- The threshold is measured in time units of data. The replay does not start until the time units of data are equal to threshold value.
- Data are stored in buffer at a possibly variable rate, but they are extracted and played back at fixed rate.
- Note that amount of data in buffer shrinks or expand but as long as delay is less than time to play back the threshold amount of data, there is no jitter.

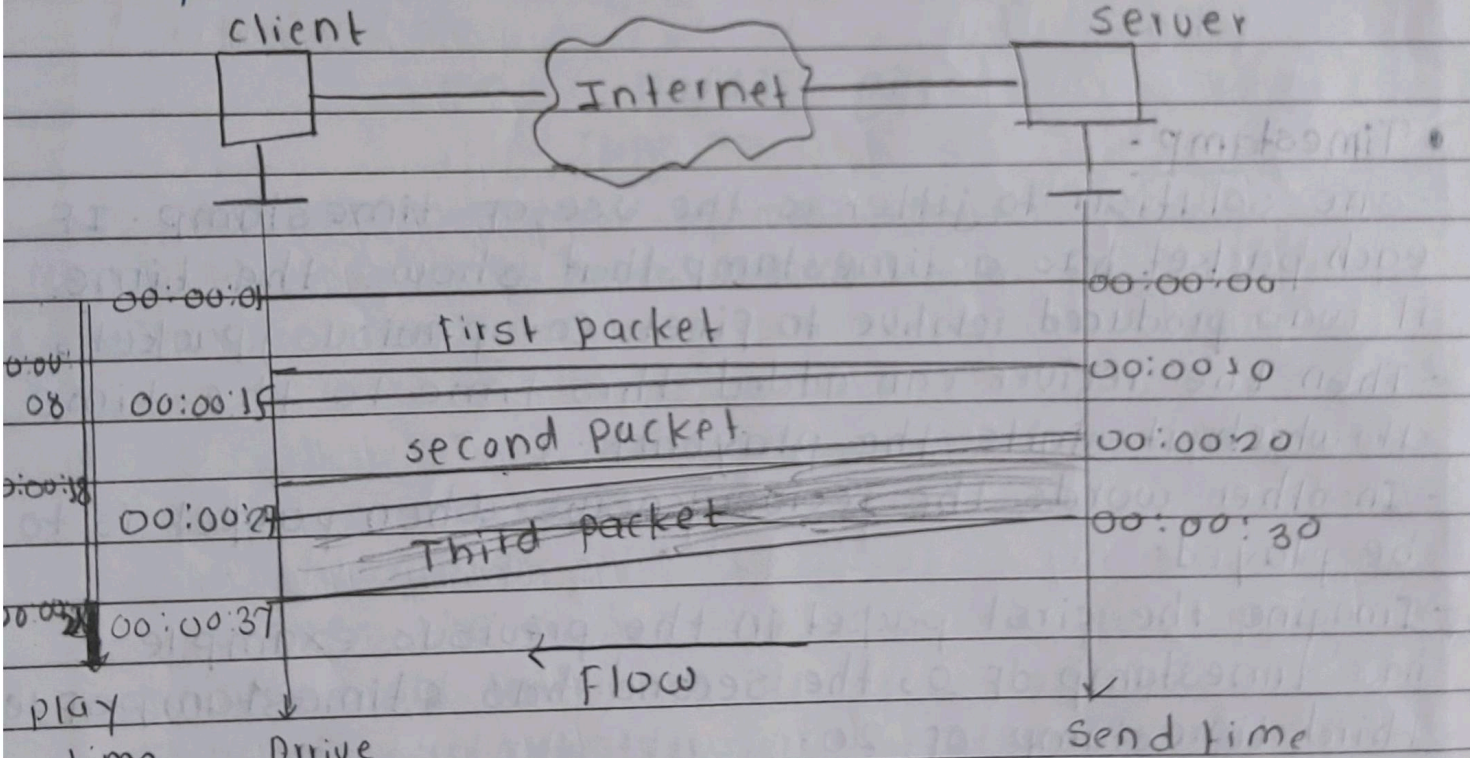


Fig Timestamp.

To prevent jitter, we can timestamp the packet and seprate the arrival time from the playback time.

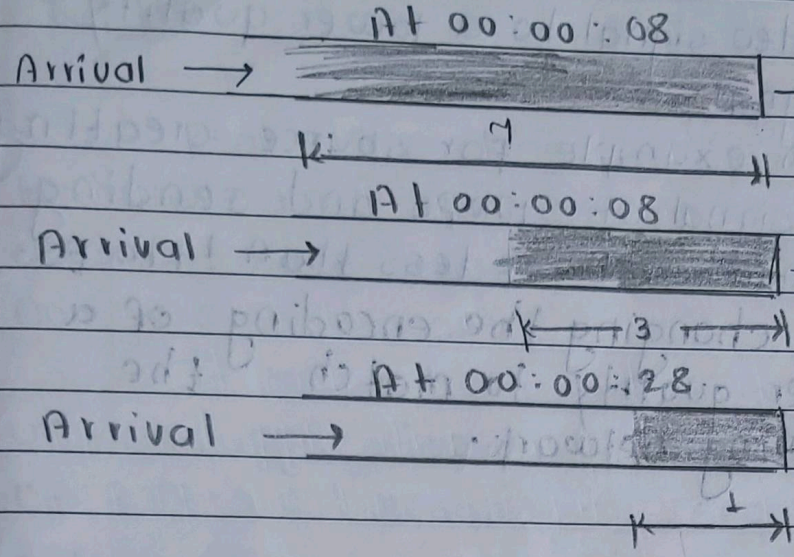


Fig. playback buffer

A playback buffer is required to real time traffic.

- ordering-

In addition to time relationship and timestamp for real time traffic, one more features is needed. we need sequence number for each packet.

The time stamp alone cannot inform the reciver if packet is lost, the reciver reciving just two packets with timestamp 0 & 20.

The reciver assumes thet the packet with timestamp 20 is the second packet produced 20s after the first.

A sequence number on each packet is required for real time traffic.

- multicasting-

multimedia play a primary role in audio or video conferencing. The traffic can be heavy & the data are distributed using multicasting method.

Real time traffic needs the support of multicasting.

- Translation-

some time real time traffic needs translation. A translator is computer that can change the format

of high bandwidth video signal to a lower quality narrow bandwidth signal.
- This is needed, for example for source creating a high quality video signal at 5Mbps and sending to a recipient having bandwidth of less than 1Mbps.
Translation means changing the encoding of a payload to a lower quality to match the bandwidth of reciving network.

- **Mixing-**
If there is more than one source that can send data at the same time.
Mixing means combining several streams of traffic into stream.

- **Support from Transport Layer protocol-**
The procedures mentioned in the privious sectn can be implemented in the application layer However they are so common real time application that implementation in the transport layer protocol is preferable. Let see which of the existing transport layer is suitable for this type of traffic
TCP is not suitable for interactive traffic
It has no provision for time stamping and it does not support multicasting.
However, it does provide ordering
- TCP with all so its sophistication, is not suitable for interactive multimedia traffic beacuse we cannot allowed retransmission of packet.
- UDP is more suitable for more interactive multimedia traffic. UDP support multicasting and has no retransmission strategy.
- UDP is more suitable than TCP for interractive traffic However we need the services of RTP another transport layer protocol to make up for deficiencies of UDP.

RTP-

Real time Transport protocol is protocol designed to handle real-time traffic on the internet. RTP does not have delivery mechanism (multicasting, port numbers and so on).

It must be used with UDP. RTP stands betn UDP and application program. The main contribution of RTP are timestamping, sequencing and mixing facilities.

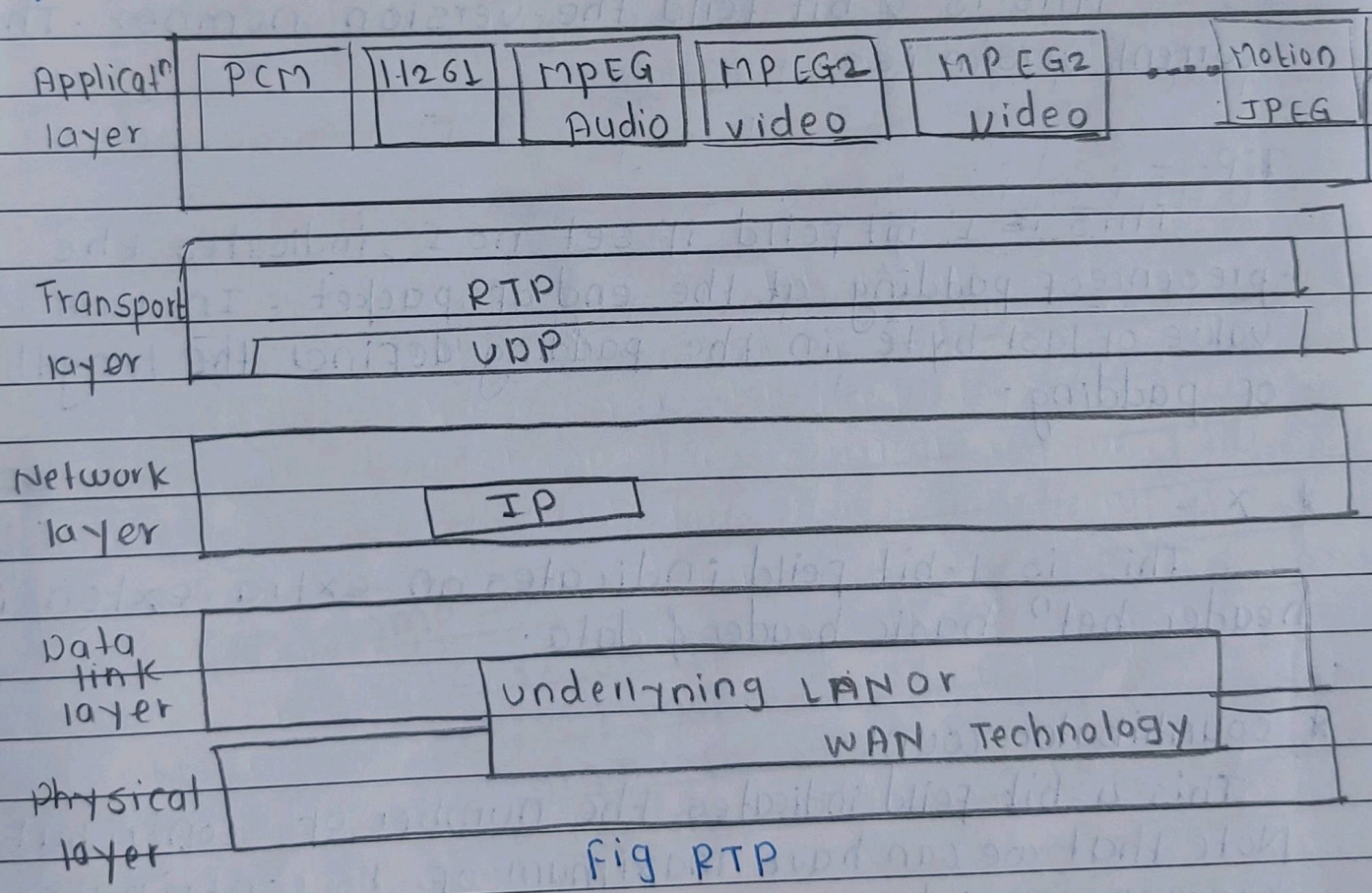
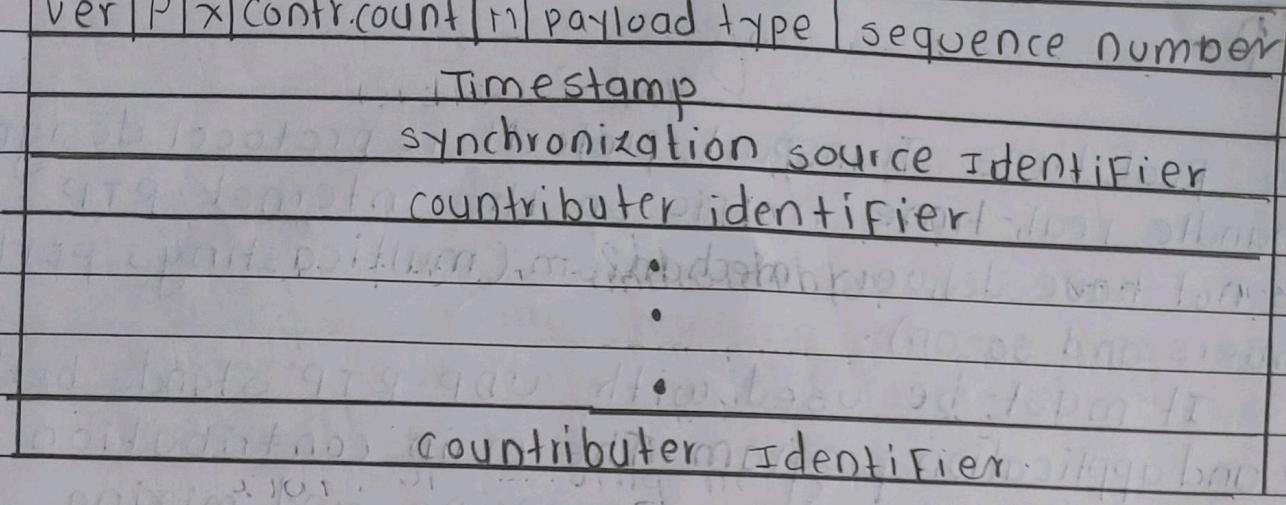


fig RTP

RTP packet format-

Fig shows the format of the RTP packet header. The format is very simple and genral enough to cover all real time application.

A description of each feild follows-



*ver -

This is 2-bit feild the version number. The current version is 2.

* P. -

This is 1 bit feild if set no 1, indicates the presence of padding at the end of packet. In this case value of last byte in the padding defines the length of padding.

* X -

This is 1-bit feild indicates an extra extension header bet^n basic header & data.

* countributor count-

This 4-bit feild indicates the number of contributor. Note that we can have maximum of 15 contributor beacuse 4-feild only allows a number between 0 & 15.

* M -

This one bit feild indicates type of pay load. seural payload have been defined so far. we list common application in Table. A discussion of the type is beyond the scope of book.

Type	Application	Type	Application
0	PCMU Audio	7	LPC audio
1	1016	8	PCMA audio
2	G721 audio	9	G722 Audio
3	GSM audio	10-11	L16 audio
5-6	DVI4 audio	14	MPEG audio

Type	Application
15	G728
26	Motion JPEG
31	H.261
32	MPEG1 video
33	MPEG2 video

* sequence Number -
  - This feild 16 bit in length. It is used to number the RTP packets
  - The sequence number of first packet is chosen randomly : It is incremented by 1 each subsequent packet.

* Timestamp -
  - This is 32 bit feild that indicates the time relationship between packets.
  - The timestamp for first packet is random number The time clock for application is 160.

* synchronization source identifier -
  If there is only one source, this 32 bit feild defines the source. However if there are seural sources the mixer is the synchronization source & other source are contributors.

* contributor Identifier -
  This 32 bit identifier (maximum 15)

define a source. when there is more than one source in session, the mixer is synchronization source & the remaining sources & are the contributors.

UDP port -

- Although RTP is itself a transport layer protocol. the RTP packet is not encapsulated directly in a IP datagram.
- Instead RTP treated like an application programm. and it encapsulated in a UDP user datagram.
- However, unlike other application program well known port is assigned to RTP.
- The port can selected on demand with only one restrictn.
- The port no. must be even number. the next no is used by the companion of RTP. Real time Transport control protocol (RTCP).

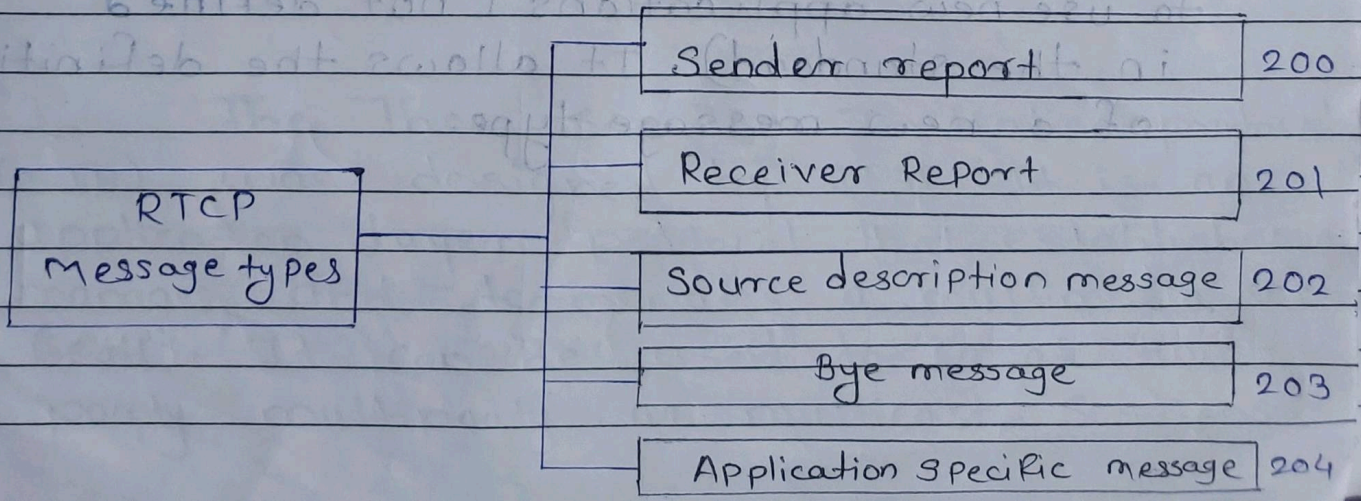
RTCP (Real-time Transport Control Protocol)

RTP allows only one type of message, one that carries data from the source to the destination. In many cases, there is a need for other messages in a session. These messages control the flow and quality of data and allow the recipient to send feedback to the source or sources.

Real-Time Transport Control Protocol (RTCP) is a protocol designed for this purpose. RTCP has five types of messages as shown in Fig. 25.20. The number next to each box defines the type of the message.

Sender Report

The Sender-report is sent periodically by the active senders in a conference to report transmission and reception statistics for all RTP packets set during the interval. The Sender report includes an absolute timestamp, which is the number of seconds elapsed since midnight January 1, 1970. The absolute timestamp allows the receiver to synchronize different RTP message. It is particularly important when both audio and video are transmitted.



Receiver Report:

The receiver report is for passive participants, those that do not send RTP packets. The report informs the sender and other receiver about the quality of service.

Source Description Message:

The source periodically sends a source description message to give additional information about itself. This information can be name, e-mail address, telephone number, and address of the owner or controller of the source.

Bye Message:

A source sends a bye message to shut down a stream. It allows the source to announce that it is leaving the conference. Although other sources can detect the absence of a source, this message is a direct announcement. It is also very useful to a mixer.

Application-Specific Message:

The Application-specific message is a packet for an application that wants to use new applications (not defined in the standard). It allows the definition of a new message type.

UDP Port

RTCP, like RTP, does not use a well-known UDP Port. It uses a temporary port. The UDP Port chosen must be the number immediately following the UDP port selected for RTP, which makes it an odd-number port.

RTCP uses an odd-numbered UDP port number that follows the port number selected for RTP.

* VOICE OVER IP

Let us concentrate on one real-time interactive audio/video application: voice over IP or Internet telephony. The idea is to use the internet as a telephone network with some additional capabilities. Instead of communication over a circuit-switched network, this application allows communication between two parties over the packet-switched Internet. Two protocols have been assigned designed to handle this type of communication. SIP and H.323. We briefly discuss both.

SIP :-

~~The~~ The Session Initiation protocol (SIP) was designed by IETF. It is an application layer protocol that establishes, manages and terminates a multimedia session (call). It can be used to create two-party, multiparty, or multicast sessions.

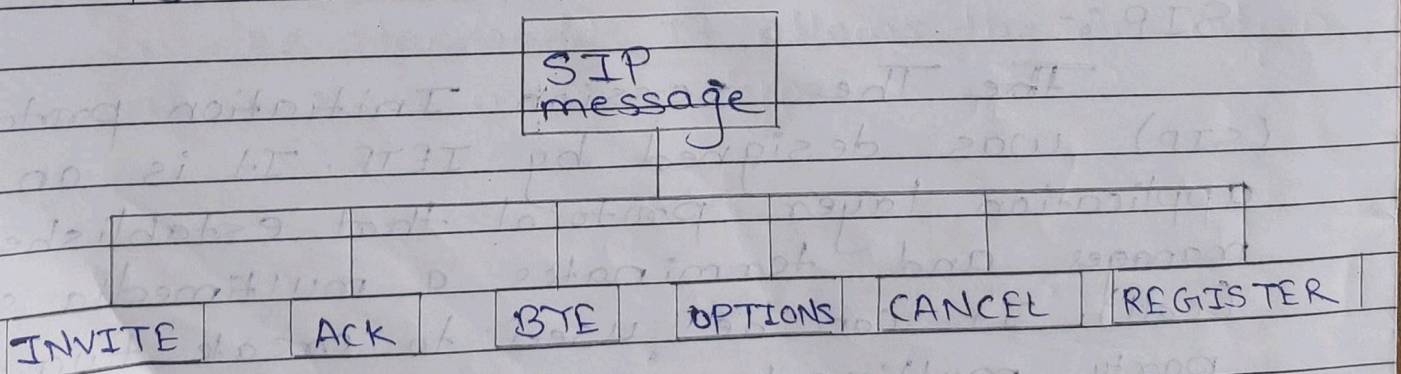
SIP is designed to be independent of the underlying transport layer, it can run on either UDP, TCP, or SCTP.

Messages:

SIP is a text-based protocol like HTTP. SIP like HTTP, uses messages. Six messages are defined as shown in Fig 25.21

Each message has a header and a body. The header consists of several lines that describe the structure of the message. caller's capability, media type, and so on. We give a brief description of each message. Then we show their applications in a simple session.

The caller initialize a session with the INVITE message. After the callee answers the call, the caller sends an Ack message for confirmation. The BYE message terminates a session. The OPTIONS message quiries a machine about its capabilities. The CANCEL message cancels an already started initialization process. The REGISTER message makes a connection when the callee is not available.



Addresses:

In a regular telephone communication a telephone number identifier the sender, and another telephone number identifies the Sender, and another telephone number identifies the receiver. SIP is very flexible. In SIP, an e-mail address, an IP address, a telephone number and other types of addresses can be used to identify the sender and receiver. However, the address needs to be in SIP format (also called scheme.

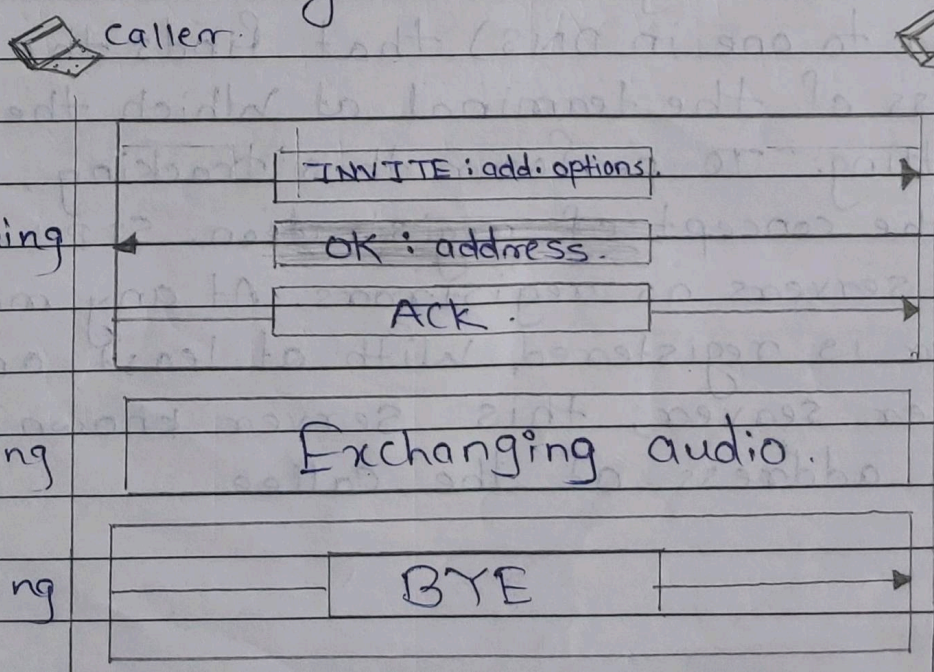
sip.bob@201.23.45.78	sip.bob@Phda.edu
IPV4 address	E-mail address

sip.bob@408-864-8900

Phone number

Simple Session:

A simple session using SIP consists of three modules: establishing, communicating and terminating.



Establishing a session :-

Establishing a session in SIP requires a three-way handshake. The caller sends an INVITE message, using UDP, TCP, or SCTP to begin the communication. If the callee is willing to start the session, she sends a reply message. To confirm that a reply code has been received, the caller sends an ACK message.

Communicating :-

After the session has been established, the caller and the callee can communicate using two temporary ports.

Terminating the session :-

The session can be terminated with a BYE message sent by either party.

Tracking the callee :

What happen if the callee is not sitting at her terminal? she may be away from her system or at another terminal. She may not even have a fixed IP address if DHCP is being used. SIP has a mechanism (similar to one in DNS) that finds the IP address of the terminal at which the callee is sitting. To perform this tracking, SIP uses the concept of registration. SIP defines some servers as registrars. At any moment a user is registered with at least one register server; this server known the IP address of the callee.

When a caller needs to communicate with the Callee, the Caller can use the e-mail address instead of the IP address in the INVITE message. The message goes to proxy server. The proxy server sends a lookup message (not part of SIP) to some register server that has registered the callee.

When the proxy server receives a reply msg from the registerr server, the proxy server takes the caller INVITE msg and inserts the newly discovered IP address of the callee. This msg is then sent to the callee.