UNIT-VIII

APPLICATION LAYER

- Domain Name Space
- DNS in Internet
- Electronic Mail
- FTP
- WWW
- HTTP
- SNMP
- Multimedia
- Network Security

Domain Name Space (DNS)

* Domain Name System.

- Name space
- Domain Name Space
- Distribution of Name space
- DNS in the Internet
- Resolution
- DNS Messages
- Types of Records
- Registrars
- Dynamic Domain Name System (DDNS)
- Encapsulation

Namespace:

↳ The names assigned to machines must be carefully selected from a Namespace with complete control over the binding b/w the names & IP addresses.

↳ A name space that maps each address to a unique name can be organized in two ways

1) Flat Namespace – It is assigned to an address → sequence of characters without structures.

2) hierachical Name space
[Ex: challenger.fhda.edu]
challenger.berkeley.edu
challenger.smart.com.

Domain Name space (DNS):

To achieve hierarchical Name space, DNS was designed. In this design the names are defined in an inverted tree structure with the root at the top.

↳ A tree can have only 128 levels: level 0 – level 127

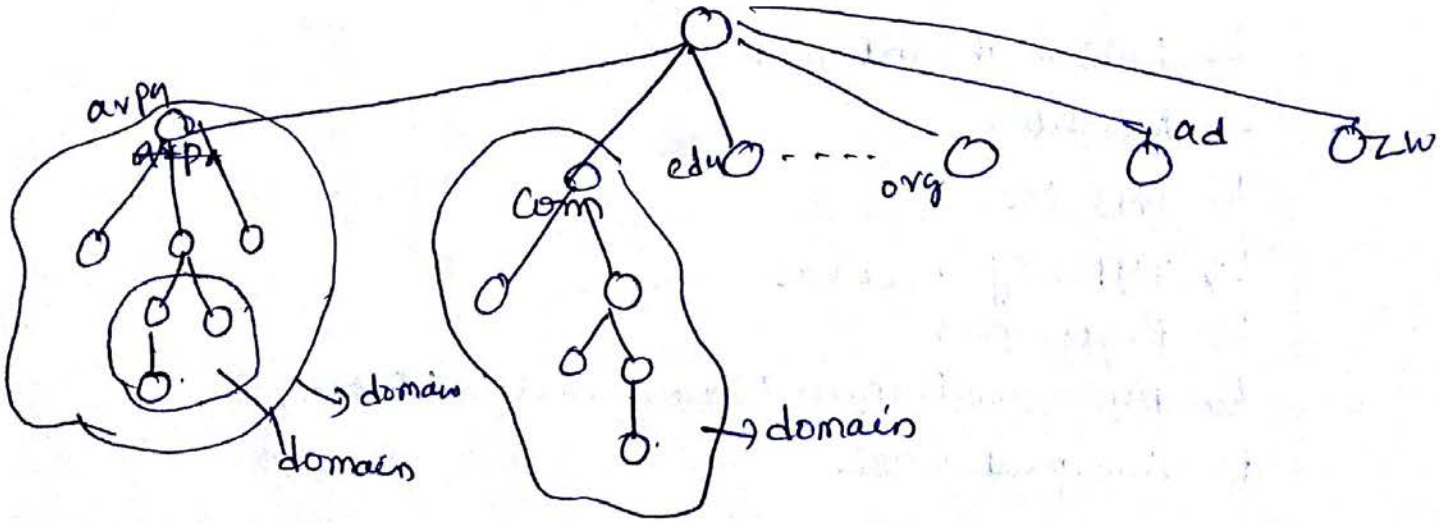
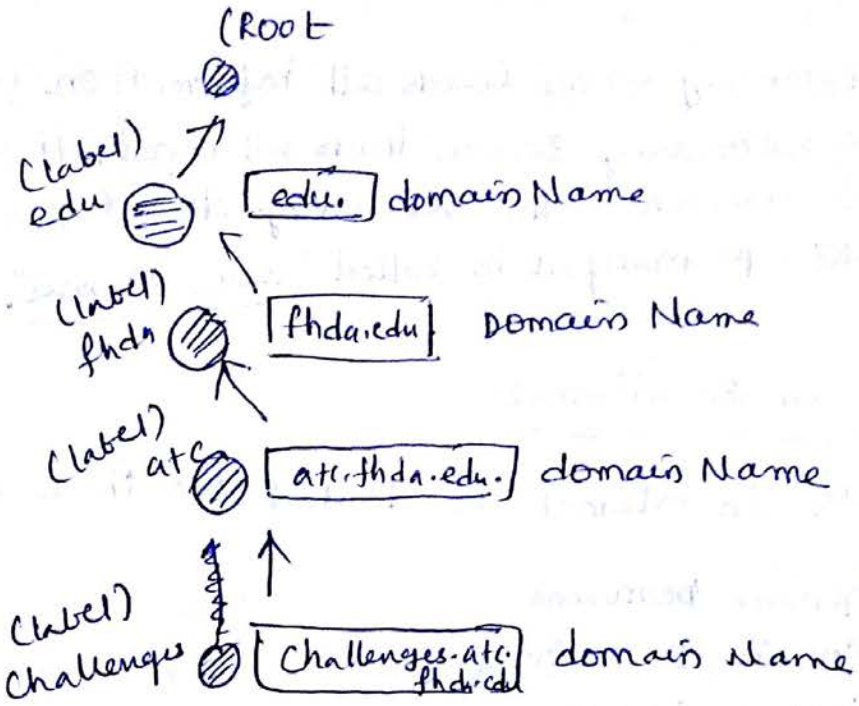


Fig: DNS.

↳ Each node in a tree, has a label, which is a string of maximum 63 characters. The root label is a NULL string (empty string)

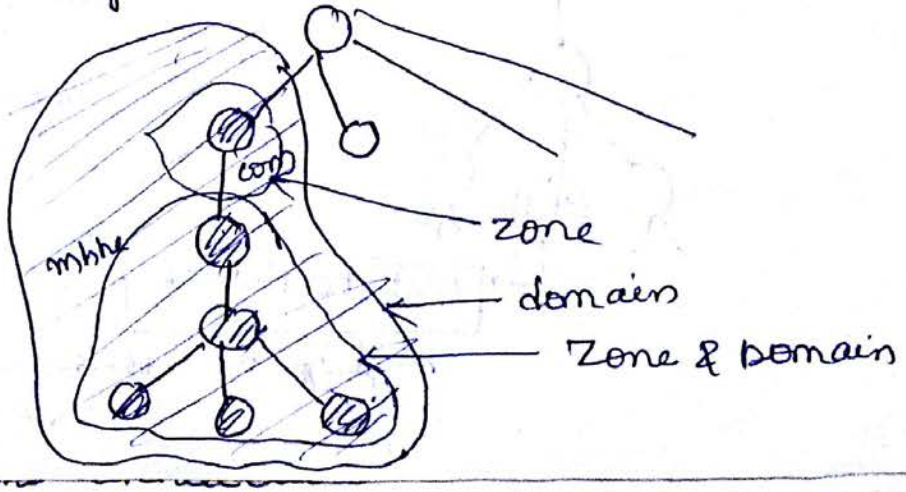
↳ DNS requires that children of a node have different Labels, which guarantees the uniqueness of the domain names.



Distribution of Name Space:

The information contained in the DNS must be stored. However, it is very inefficient & also un-reliable to have just one computer store such a huge amount of information. It is inefficient because responding to requests from all over the world places a heavily load on the system. It is not unreliable because any failure makes the data inaccessible.

↳ The following figure shows zones & Domains:



↳ DNS defines two types of servers:

↳ primary server &
↳ secondary server.

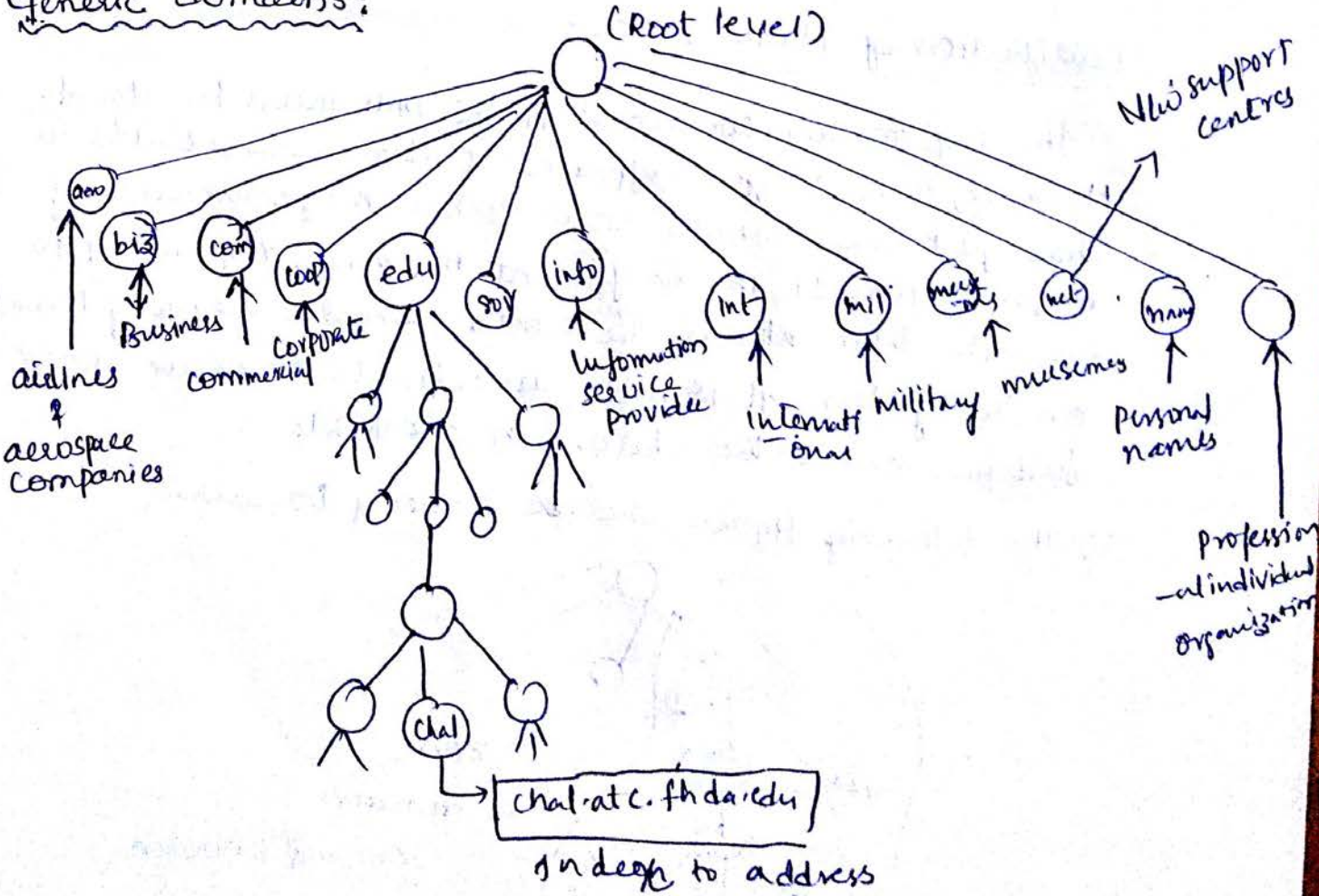
↳ A primary server loads all information from the disk file; the secondary server loads all information from the primary server. when the secondary downloads information from the primary, it is called "Zone transfer".

DNS in the internet

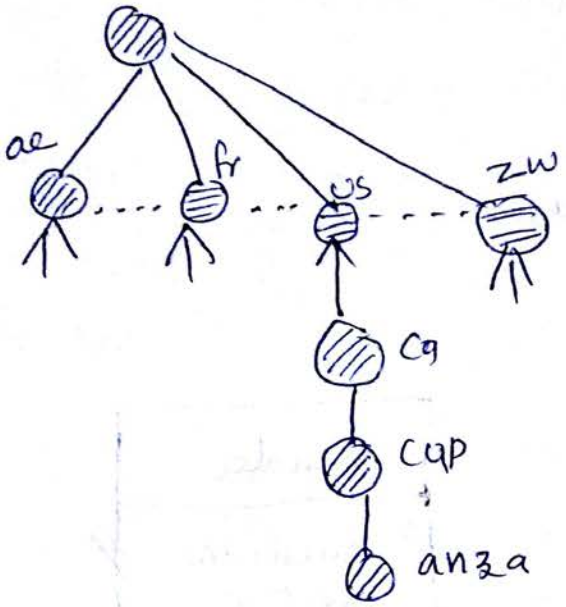
DNS in internet is divided into three different sections:

↳ Generic Domains
↳ Country Domains &
↳ Inverse Domain.

Generic domains:



Country domains:

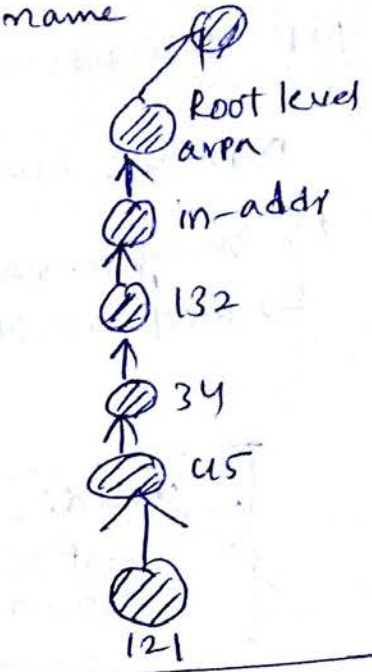


anza.cup.ca.us

Index to addresses

Inverse Domain

↳ It maps address to a name



121.45.34.132.in-addr.arpa

↓ Index to names

Resolution: (name-add, add-name)

Mapping a name to an address / an address to name is called name-address Resolution

↳ A host that needs to map an address to a name (or) a name to address calls a DNS client & a "Resolver"

(i) Mapping Names to Address:

In most cases, the resolver gives a domain name to the server & asks for the corresponding address.

↳ Then the server checks the generic domains (or) the country domains to find mapping

ex: chal.atc.fhda.edu
ch.fhda.cu.ca.us

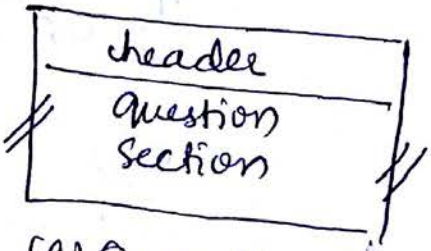
② Mapping Address to Name:

Ex: "121.45.34.132.in-addr.arpa"

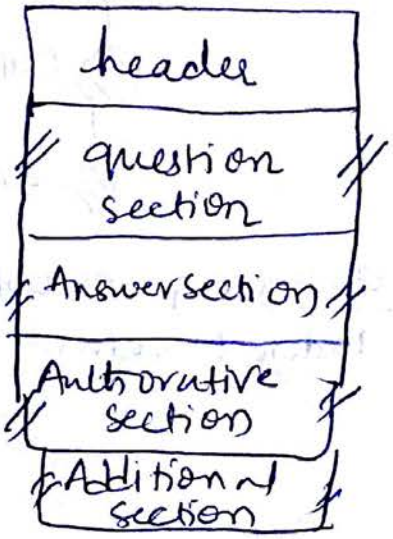
DNS Messages:

DNS has two types of messages:

↳ query messages
↳ Response Messages.



(a) Query Msg.



(b) Response msg.

⇒ The header format is shown as:

Identification	flags
No. of question records	No. of answer records (all 0's in query msg)
No. of authorative records (all 0's in query msg)	No. of additional records (all 0's in query msg)

(header)

Types of Records:

Two types of Records are used in DNS.

1) Question Record (It is used by client to get information from a server, it contains Domain Name)

↳ Resource Record → Each domain name is associated with a record called a Resource record

→ The Server database consists of resource records.

↳ Resource records are also what is returned by the server to the client.

Registrars

↳ New domains are added to DNS, through a Registrar, It is a commercial entity accrediated by ICANN.

↳ A registrar first verifies that the requested domain name is unique & then enters it into the DNS database. A fee is charged.

↳ Today, there are many registrars; their names & addresses can be found at, http://www.internic.net

Ex: Domain Name: ws.wonderful.com
Ip address : 200.200.200.5

Dynamic-DNS (DDNS)

When DNS was designed, no one predicted that there would be so many address changes.

↳ In DNS, when there is a change, such as adding a new host, removing a host, changing an Ip address, the change must be made to the DNS master file.

↳ These types of changes involve a lot of manual updating

↳ The DNS master file is updated dynamically

∴ Dynamic DNS (DDNS) was designed.

Encapsulation

↳ DNS can be either UDP/TCP, using the well-known port 53.

↳ UDP is used when the size of the response message.

Remote Logging

In the Internet, users may want to run application programs at a remote site & create results that can be transferred to their local site,

Ex: Students may want to connect to their university computer lab from their home to access application programs for doing homework assignments/projects,

TELNET: It is a client/server application program. It is a general-purpose client/server application program.

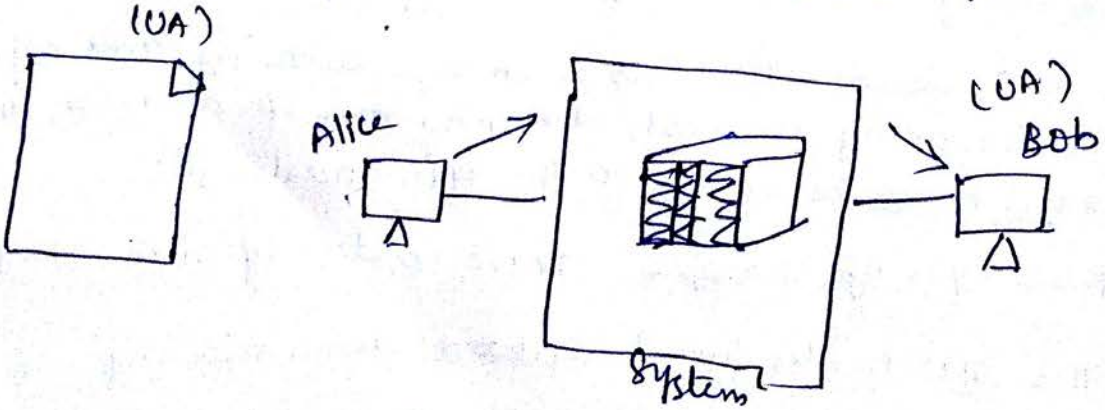
↳ Telnet is designed for

* Timesharing environment
* Logging.

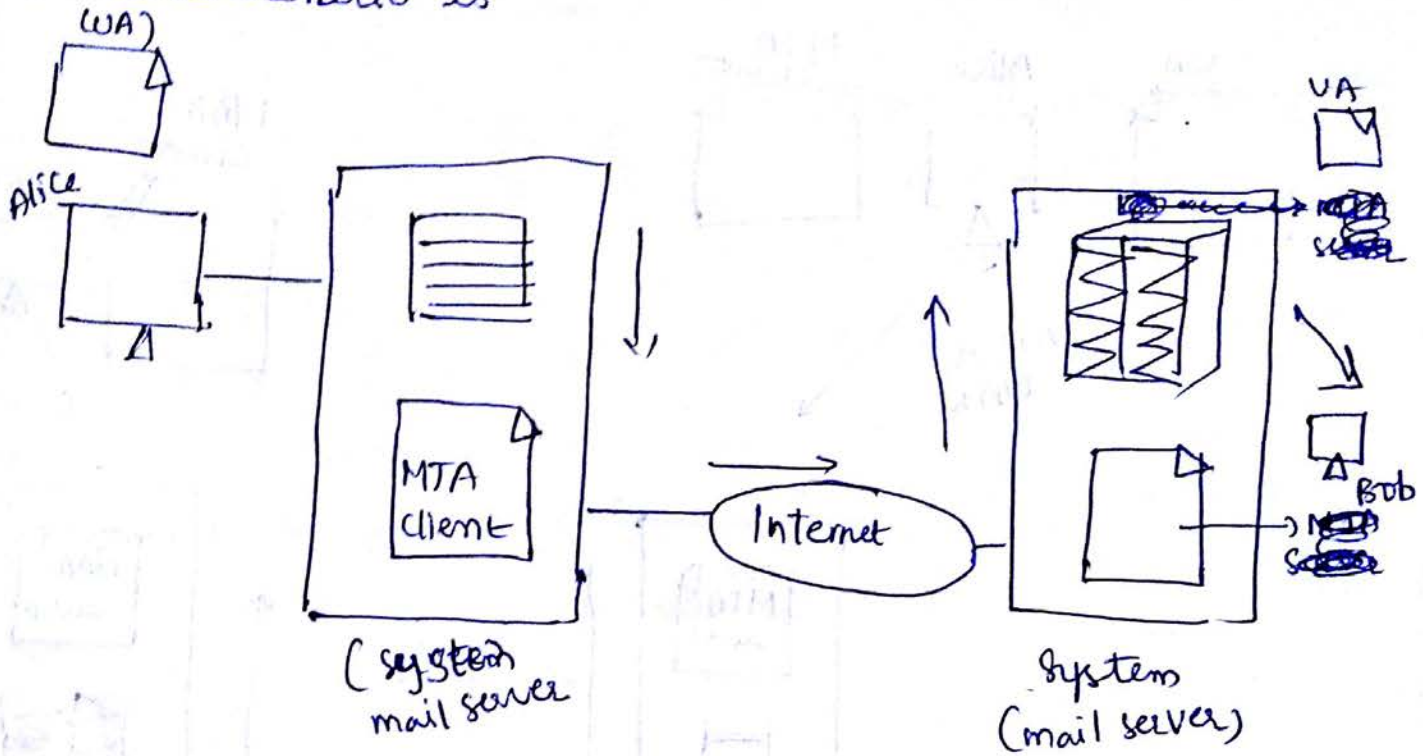
Electronic Mail:

↳ One of the popular Internet services is e-mail.

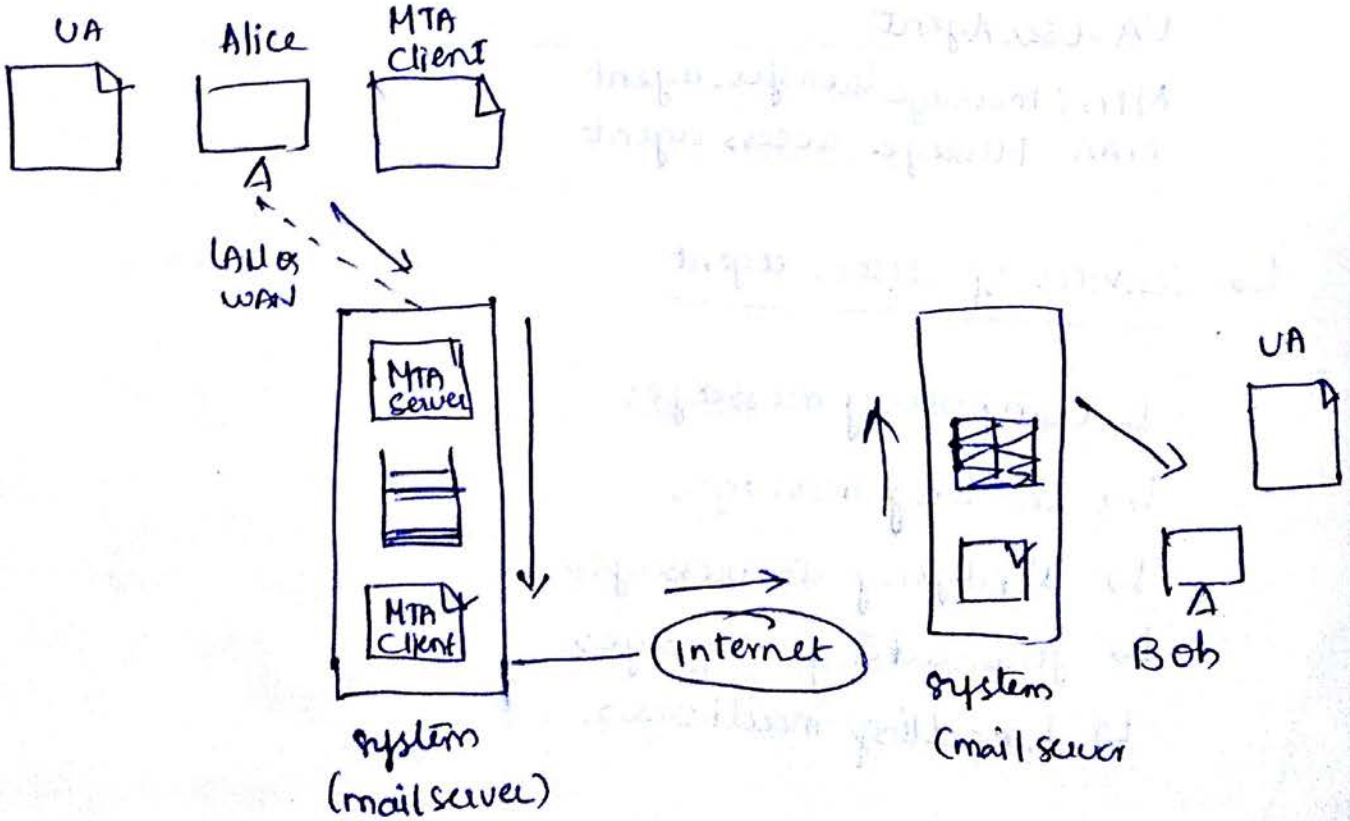
↳ Architecture of e-mail (first Scenario is)



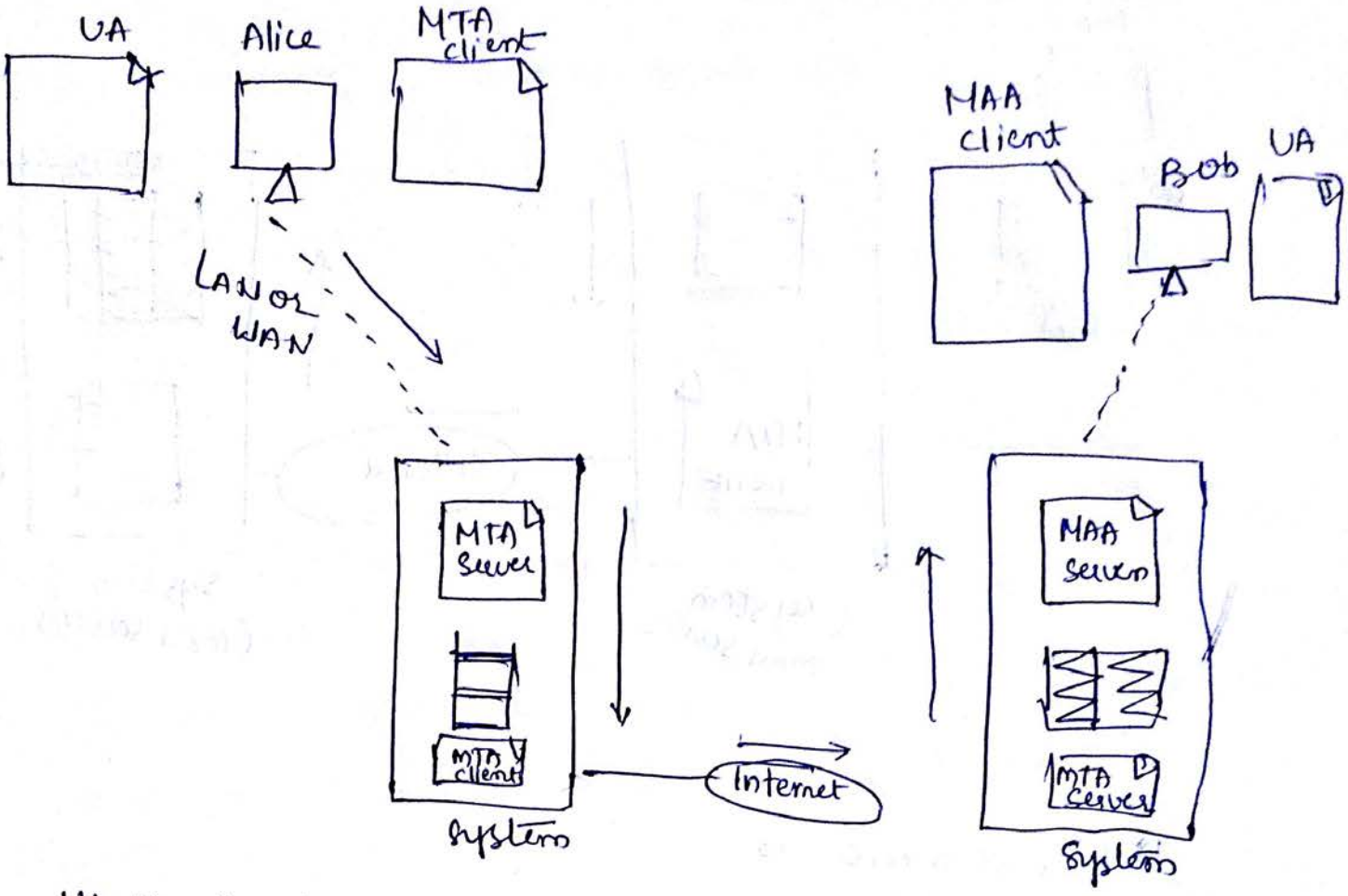
↳ Second scenario is



↳ Third scenario is



Fourth Scenario is



UA: User Agent

MTA: message transfer agent

MAA: Message access agent

↳ Services of user agent

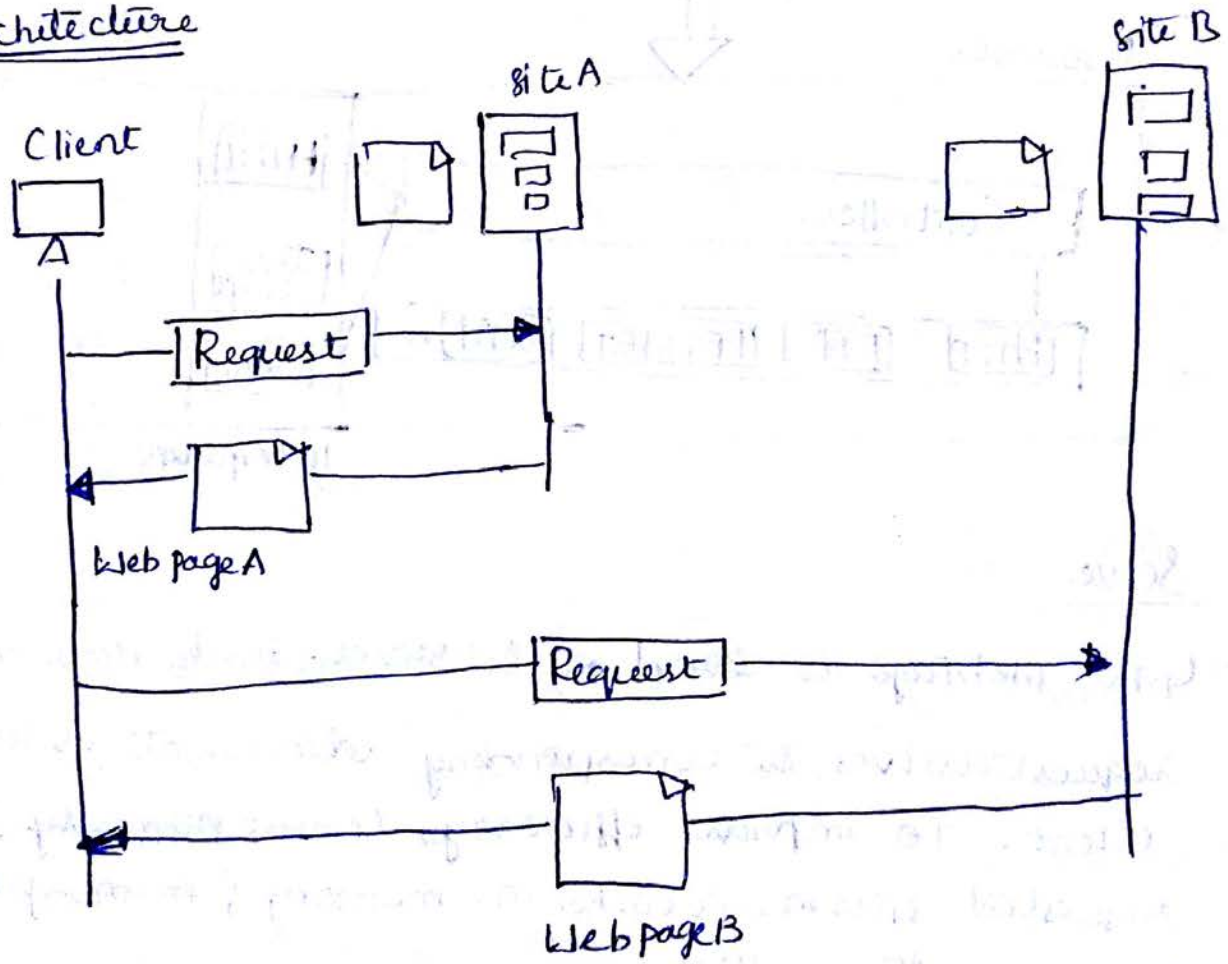
- ↳ Composing messages
- ↳ Reading messages
- ↳ Replying to messages
- ↳ forwarding messages
- ↳ handling mailboxes.

WWW (World Wide Web):

↳ The WWW is a repository of information linked together from points all over world. The WWW has a unique combination of flexibility, portability, & user-friendly features that distinguish from other services provided by the Internet.

↳ The WWW project was initiated by CERN (European Laboratory for particle physics) to create a systems to handle distributed resources necessary for scientific research.

Architecture



Client

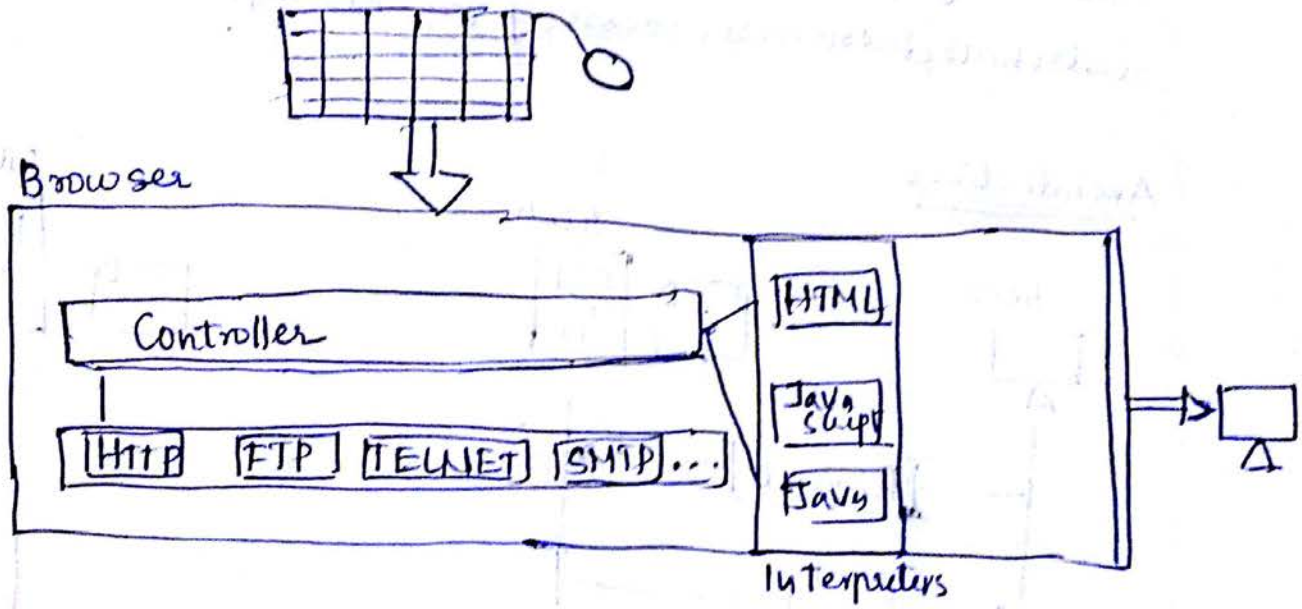
↳ A variety of vendors offer commercial browsers that interpret & display a web documents, and all use nearly the same architecture.

↳ Each browser usually consists of three parts: a) Controller, client protocol & interpreters.

↳ A controller receives the input from the keyboard or the mouse & uses the client programs to access the document.

↳ The client protocol can be one of the protocols such as FTP or HTTP.

↳ The interpreter can be HTML, Java or Java Script



Server

↳ The webpage 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 requested files in a cache in memory; memory is faster to access than disk.

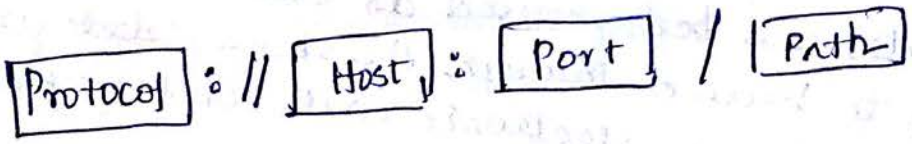
↳ A server can also become more efficient through multi-threading or multi-processing. In this case, a server can answer more than one request at a time.

Uniform Resource Locator

↳ A client that wants to access a Web page needs the address. To facilitate the access of documents distributed throughout the world, HTTP uses locators.

↳ the Uniform resource Locator (URL) is a standard for specifying any kind of information on the Internet.

↳ URL defines four things: Protocol, host computer, port & Path.



↳ The protocol is the client / server program used to retrieve the document. [FTP & HTTP]. The most common today is HTTP.

↳ The Host is the computer on which the information is located, although the name of the computer can be an alias.

↳ the URL can optionally contain the port number of the server. If the port is included, it is inserted b/w the host & the path, & it is separated from the host by the colon.

↳ Path is the pathname of the file where the information is located.

Cookies

↳ The WWW was originally designed as a stateless entity. A client sends a request; a server responds. Their relationship is over.

↳ The original WWW, retrieving publicly available documents, exactly fits this purpose.

↳ Today the Web has other functions; some are listed here

1) Some websites need to allow access to registered clients only.

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

3) Some websites are used as portals: the user selects the web pages he wants to see.

4) Some websites are just advertising.

↳ For these purposes, the cookie mechanism was devised.

* Creation & Storage of cookies.

↳ the creation & storage of cookies depends on the implementation; however, the principle is same.

1) When a server receives a request from a client, it stores information on about the client in a file or string. The information may include the domain name of the client, the contents of the cookie, a time stamp & other information depending on the implementation.

↳ The server includes the cookie in the response that it sends to the client.

↳ when the client receives the response, the browser stores the cookie in the cookie directory, which is sorted by the domain server name.

Using cookies.

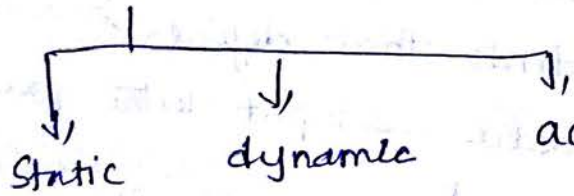
↳ when a client sends a request to a server, the browser looks in the cookie directory to see if it can find a cookie sent by that server. If found, the cookie is included in the request.

↳ when the server receives the request, it knows that this is an old client, not a new one.

↳ the contents of the cookie are never read by the browser or disclosed to the user.

↳ It is a cookie made by the server & eaten by the server.

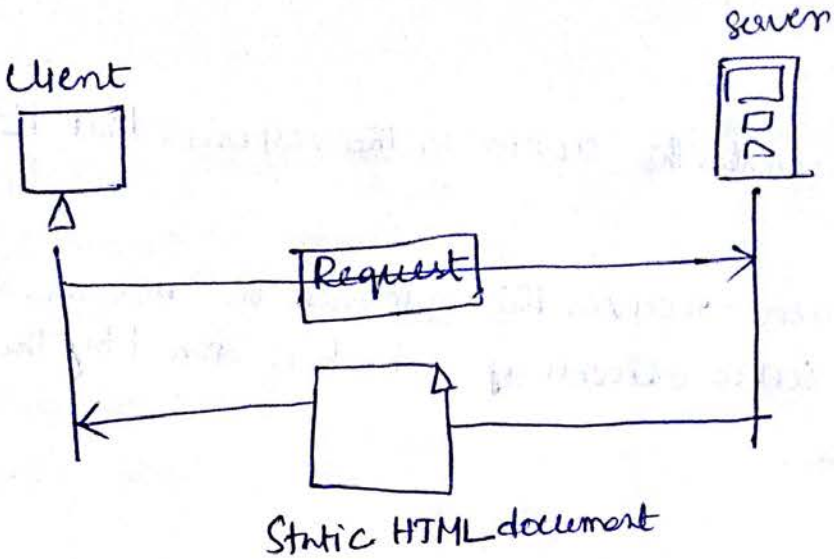
Web Documents [the documents in the WWW can be grouped into three broad categories:



↳ The category is based on the time at which the contents of the document are determined.

Static Documents.

These are fixed-content documents that are created and stored in a server. The client can get only a copy of the document.



Static HTML document

↳ HTML (Hypertext Markup language) is a language for creating Web pages.

Dynamic Documents

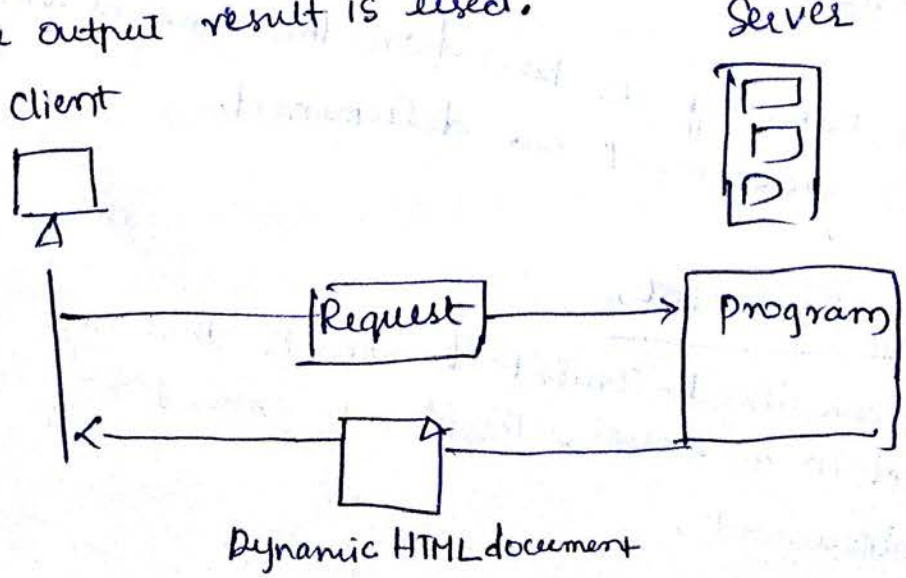
↳ A dynamic document is created by a Web Server when ever a browser requests the document.

Q1 Dynamic document is the retrieval of time & date from a server.

↳ Common Gateway Interface (CGI)

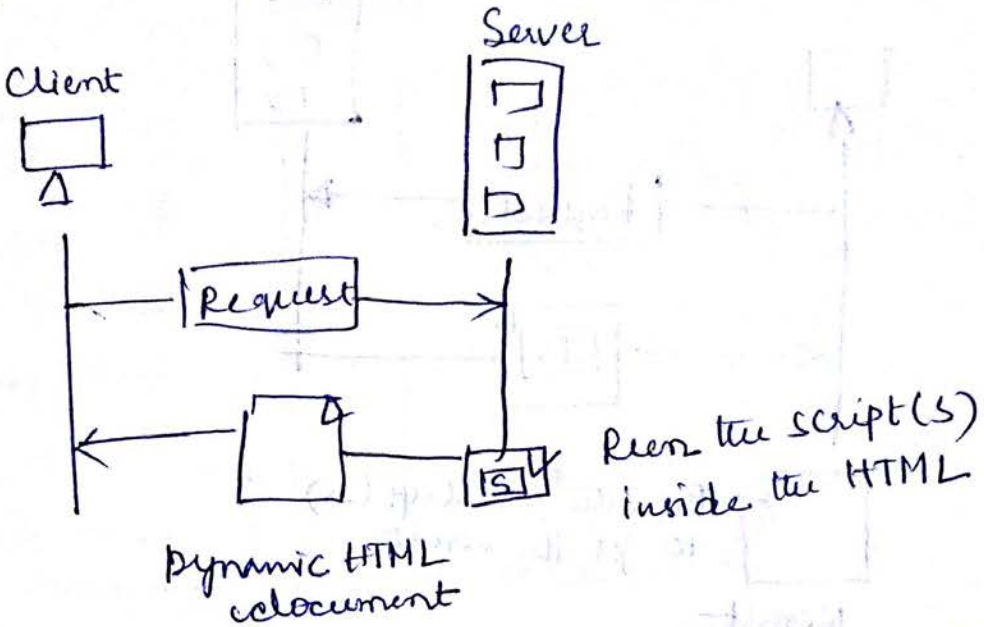
It is a technology that creates and handles dynamic documen

-ts. CGI is a set of standards that defines how a dynamic document is written, how data are input to the program, & show the output result is used.



Dynamic HTML document

↳ Scripting Technologies for Dynamic Documents

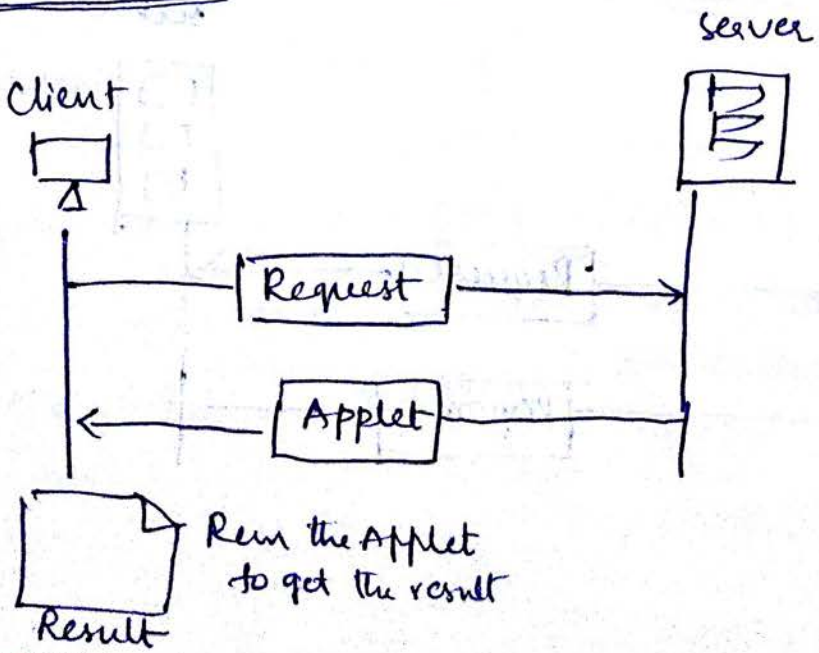


↳ Hypertext Preprocessor (PHP), Java Server pages (JSP), Active Server pages (ASP).

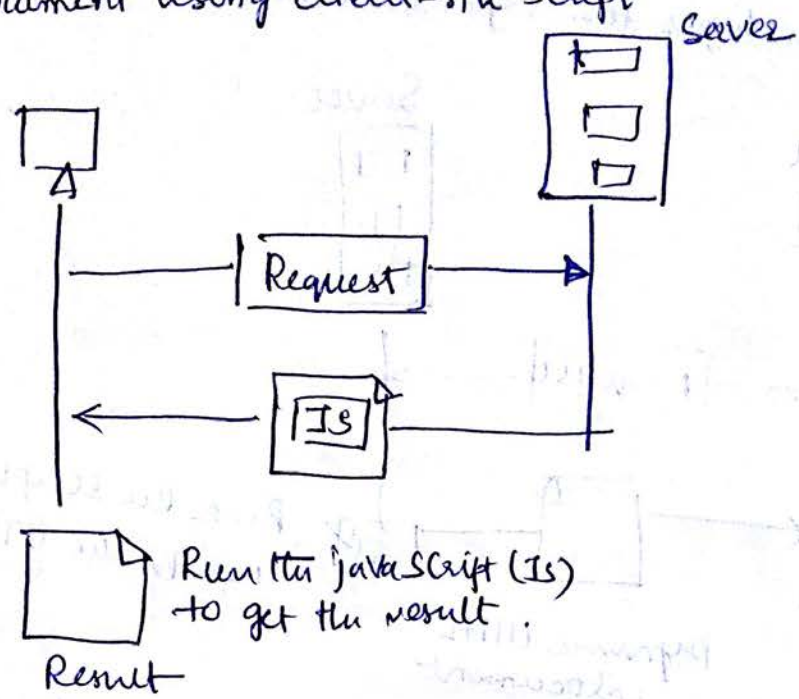
Active Documents

↳ For many Applications, there is a need of a program or a script to be run at the client site. These are called active documents.

↳ Java Applets (Applets.java)



↳ Java Script
Active document using client-site script



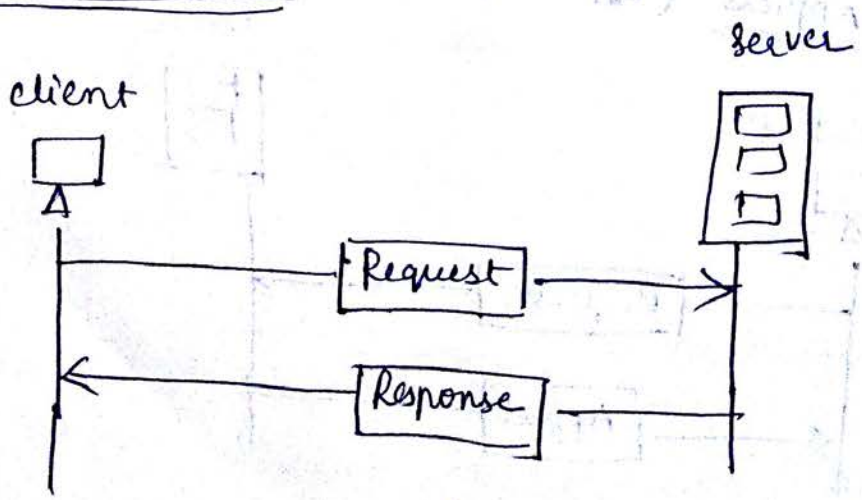
HTTP (HyperText Transfer Protocol)

↳ The HTTP is a protocol used mainly to access data on the World wide Web.

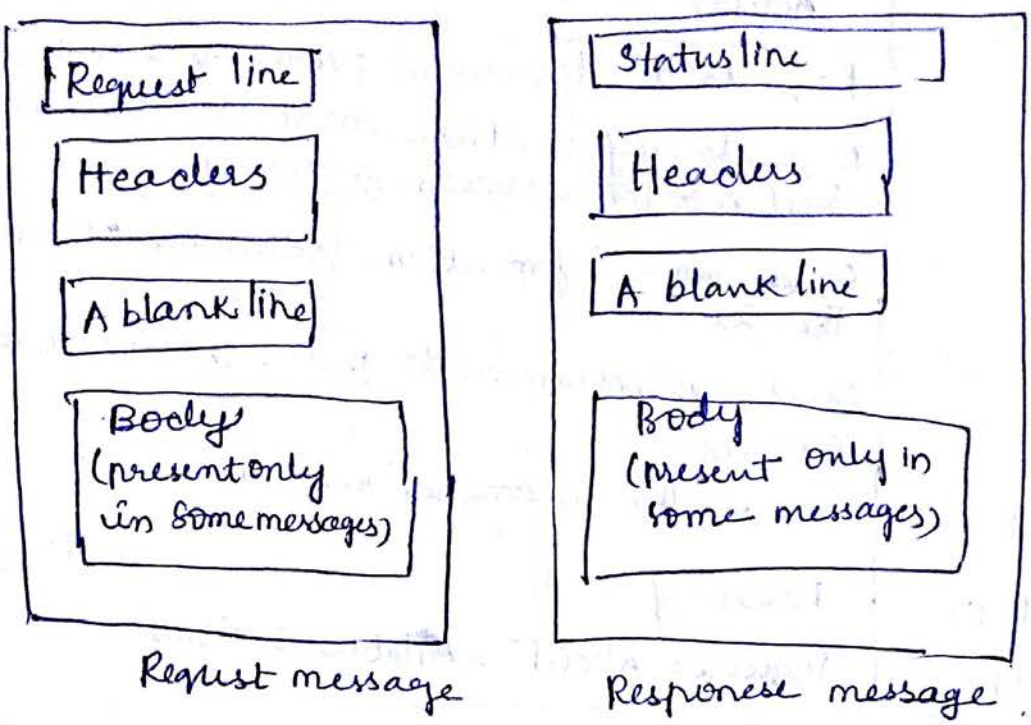
↳ HTTP functions as a combination of FTP & SMTP

↳ HTTP uses the services of TCP on well-known port-80.

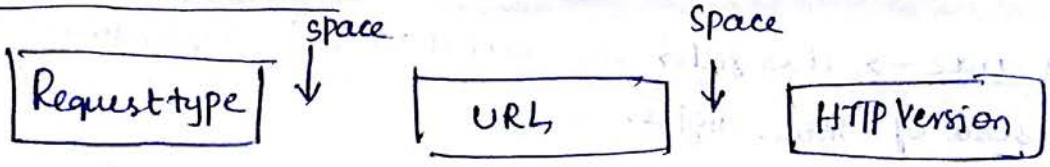
* HTTP Transaction



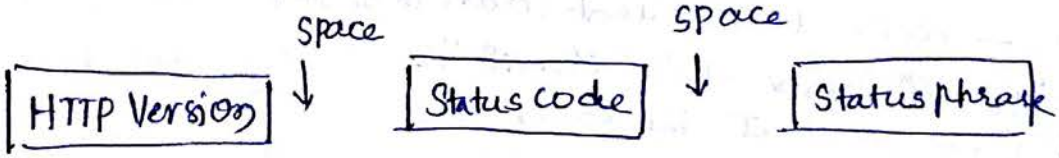
Request & Response Messages.



Request & status lines



a) Request line



b) status line

* **Request type:** → This field is used in request message.

In version of 1.1 of HTTP several request types are defined.

↳ The request type is categorized into methods.

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

Version → The most current version of HTTP is 1.1

Status code → This field is used in the response message. It consists of three digits.

- 100 — Informational
- 200 — successful request
- 300 — redirect the client to another URL
- 400 — An error at the client site
- 500 — error at the server site.

Status phrase → It is used in response message. It explains status code in text form.

Status codes

Code	phrase	Description
		Informational
100	Continue	The initial part of the request has been received & the client may continue with its request
101	Switching	The server is complying with a client request to switch protocols defined in the upgrade header.

Success		
200	OK	The request is successful
201	Created	A new URL is created
202	Accepted	The request is accepted, but it is not immediately acted upon
204	No content	There is no content in the body

Redirection		
301	moved permanently	The requested URL is no longer used by the servers
302	moved temporarily	The requested URL has moved temporarily
304	Not modified	The document has not been modified

Client Error		
400	Bad request	There is a syntax error in the request.
401	Unauthorized	The request lacks proper authorization.
403	Forbidden	Service is denied
404	Not found	The document is not found.
405	Method not allowed	The method is not supported in this URL.
406	Not acceptable	The format requested is not acceptable

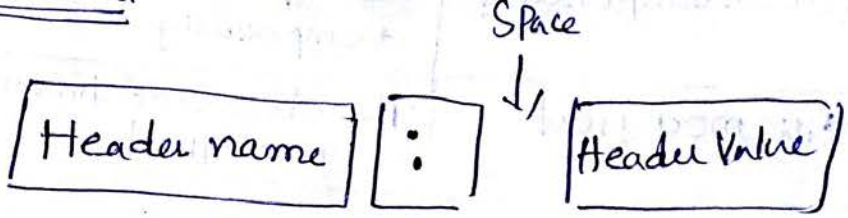
Server Error		
500	Internal Server error	There is an error, such as a crash, at the server site
501	Not implemented	The action requested cannot be performed
503	Service Unavailable	The service is temporarily unavailable but may be requested in the future

Header: Exchanges additional information b/w the client & the server. A header line belongs to one of 4 categories: General header, request header, response header & entity header.

↳ A request message can contain only general, request, & entity headers.

↳ A response message can contain only general, request, response & entity headers.

Header format



General header → Gives general information about the message and can be present in both a request & a response.

Header	Description
Cache-Control	Specifies information about caching
Connection	Shows whether the connection should be closed or not
Date	Shows the current date
MIME-Version	Shows the MIME Version used
Upgrade	Specifies the preferred communication protocol

Request header → Can be present only in a request message. It specifies the client's configuration & the client's preferred document format.

Header	Description
Accept	Shows the medium 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 the ~~language the client can~~ permission the client has
From	Shows the e-mail address of the user
Host	Shows the host & port number of the server.
If-modified-since	sends the document if newer than specified date
If-match	sends the document only if it matches given tag
If-non-match	sends the document only if it does not match given tag
If-range	sends only the portion of the document that is missing
If-unmodified-since	Sends the document if not changed since specified date
Referer	Specifies the URL of the linked document
User-agent	Identifies the client program.

* Response header:- The response header can be present only in a response message. It specifies the server's configuration & special information about the request.

Header	Description
Accept-range	shows if server accepts the range requested by client
Age	shows the age of document
Public	shows the supported list of methods
Retry-after	specifies the date after which the server is available
Server	shows the server name & version number

* Entity header:- The entity header gives information about the body of the document.

Header	Description
Allow	List valid methods that can be used with a URL
Content-encoding	specifies the encoding scheme
Content-language	Specifies the language
Content-length	shows the length of the document
Content-range	Specifies the range of the document
Content-type	specifies the medium type
Etag	Gives an entity tag
Expires	Gives the date & time when contents may change
Last-modified	Gives the date & time of the last change
Location	specifies the location of the created or moved document.

Body: The body can be present in a request or response message. Usually, it contains the document to be sent or received.

Persistent Versus Non persistent Connection.

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

- In a non-persistent 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 & closes the connection.
  3. The client reads the data until it encounters & end-of-file marker; it then closes the connection.
- Persistent connection, the server leaves the connection open for more requests after sending a response. The server can close the connection at the request of a client or if a time-out has been reached. The sender usually sends the length of the data with each response.

Proxy Server

- HTTP supports proxy servers. A proxy server is a computer that keep copies of responses to recent requests.
- The HTTP client sends a request to the proxy server. The proxy server checks its cache.
- The proxy server reduces the load on the original server, decreases traffic & improve latency.

FTP (File Transfer Protocol)

↳ Transferring files from one computer to another is one of the most common tasks expected from a N/w or internet working environment.

↳ As a matter of fact, the greatest volume of data exchange in the Internet today is due to file transfer.

File Transfer protocol (FTP)

↳ FTP is the standard mechanism provided by TCP/IP for copying a file from one host to another.

↳ Although transferring files from one system to another seems simple & straightforward, some problems must be dealt with first.

↳ FTP differs from other client/server applications in that it establishes two connections b/w the hosts.

↳ One connection is used for data transfer, the other for control information.

↳ Separation of commands & data transfer makes FTP more efficient.

↳ The control connection uses very simple rules of communication.

↳ We need to transfer only a line of command or a line of response at a time.

↳ The data connection, on the other hand, needs more complex rules due to the variety of data.

↳ For TCP, both connections are treated the same.

↳ FTP uses two well-known TCP ports: port 21 is used for the control connection & port 20 is used for the data connection.

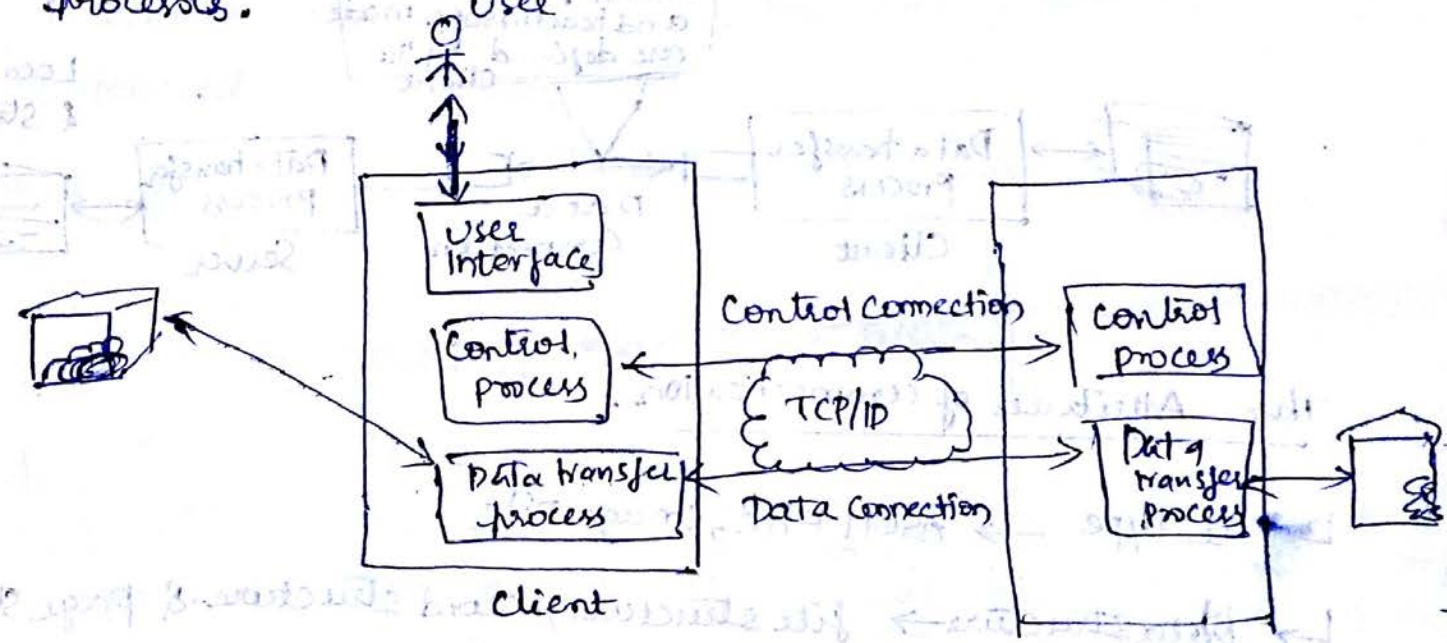
↳ The client has three components: User Interface, client control process, and the client data transfer process.

↳ The server has two components:

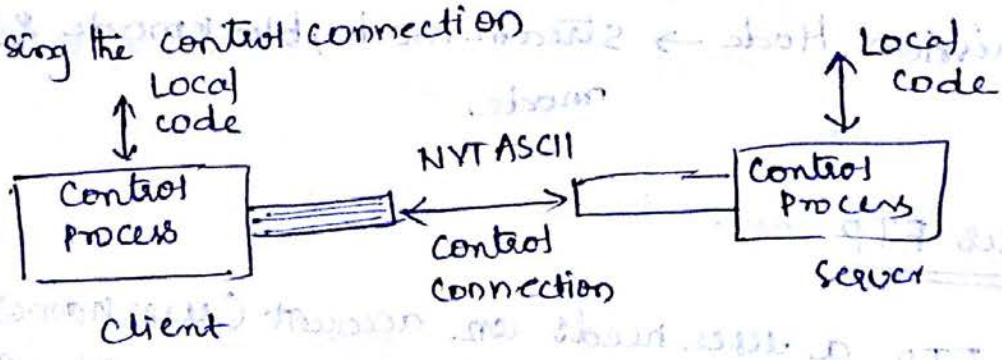
1) The server control process
2) The server data transfer process.

↳ The control connection is made b/w the control processes.

↳ The data connection is made b/w the data transfer processes.



* Using the control connection

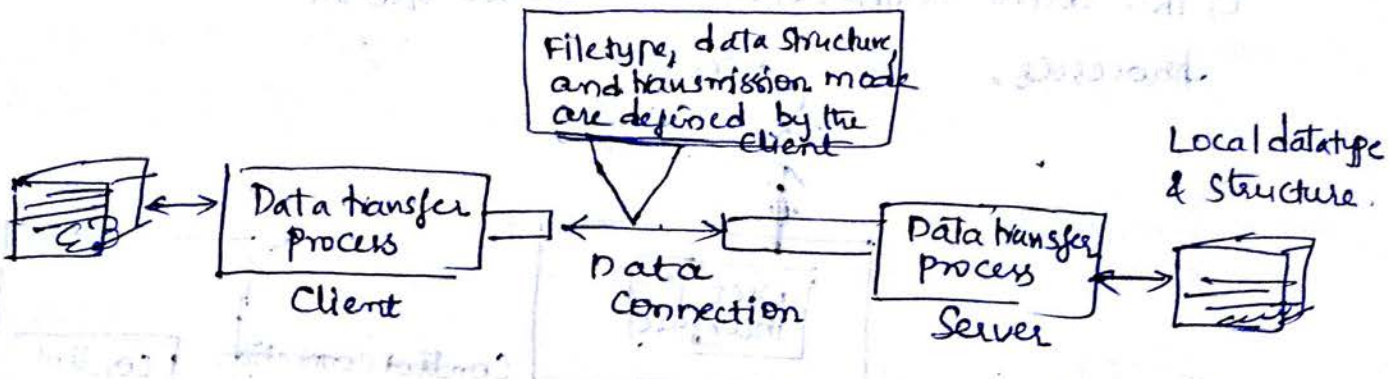


* Using the data connection

1) A file is to be copied from the server to the client. This is called retrieving a file. It is done under the supervision of the RETR command.

2) A file is to be copied from the client to server. This is called storing a file. It is done under the supervision of the STOR command.

3) A list of directory or file names is to be sent from the server to the client. This is done under the supervision of the LIST command.



Three Attributes of communication.

↳ File Type → ASCII File, Image File

↳ Data structure → file structure, record structure & page structure

↳ Transmission Mode → stream mode, block mode & compressed mode.

Anonymous FTP

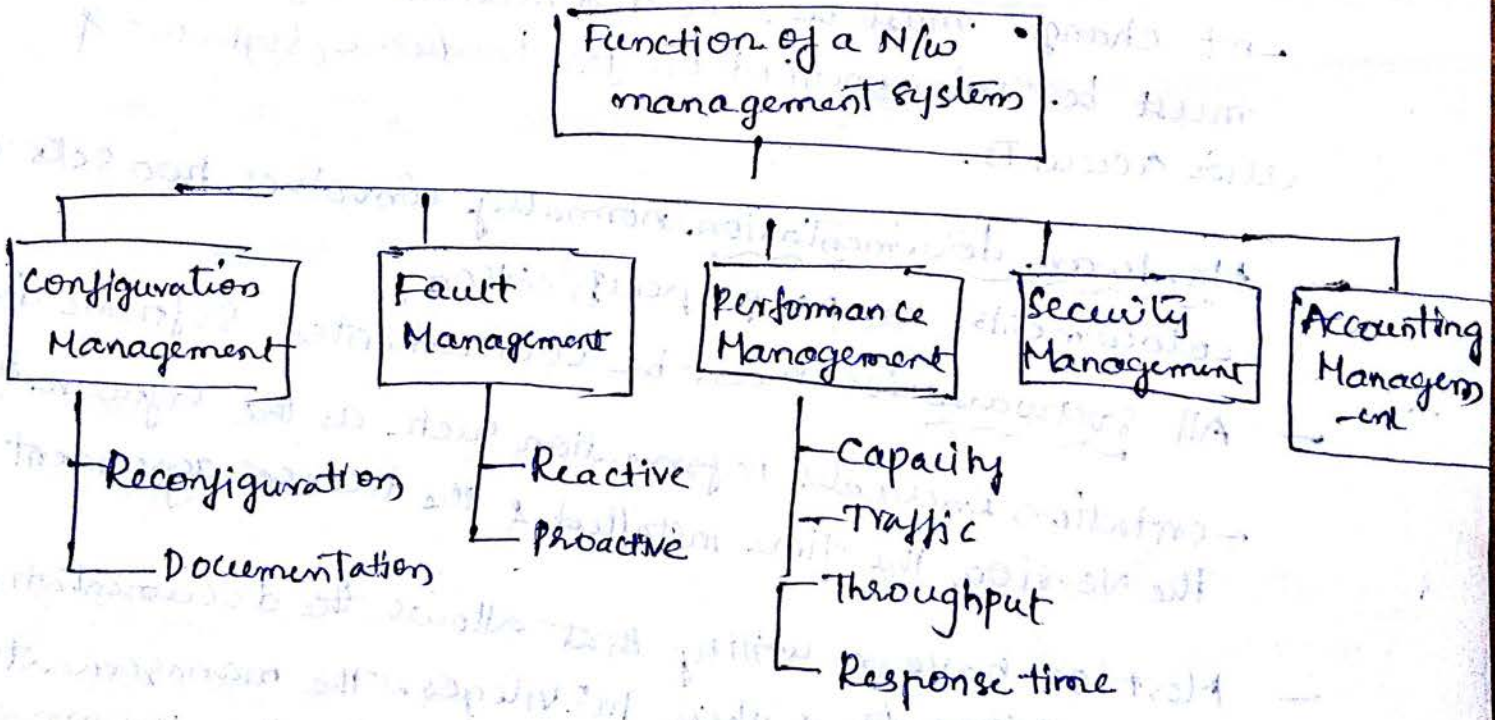
To use FTP, a user needs an account (user name) & password on the remote server. Some sites have a set of files available for public access, to enable anonymous FTP. To access these files, a user does not need to have an account or password. Instead, the user can be anonymous as the user name & guest is the password.

SNMP: Simple Network Management Protocol.

↳ Network management can be defined as monitoring, testing, configuring & trouble shooting N/w components to meet a set of requirements defined by an organization.

↳ To accomplish this task, a N/w management system uses hardware, s/w & humans. We concentrate on the most common management system, the Simple Network Management Protocol (SNMP).

* Functions of Network management system



↳ Configuration Management

A large N/w is usually made up of hundreds of entities that are physically or logically connected to one another. These entities have an initial configuration when the N/w is set up, but can change with time.

- Reconfiguration, means adjusting the H/w components & features can be a daily occurence in a large N/w.
- There are three types of reconfigurations.
  1) Hardware reconfiguration covers all changes to the H/w
  2) Software reconfiguration covers all changes to the S/w
  3) User-account reconfiguration is not simply adding or deleting users on a systems.

- Documentation, The original H/w configuration and each subsequent change must be recorded meticulously. This means there must be documentation for hardware, software & user accounts.
- Hardware documentation normally involves two sets of documents: maps & specification.
- All softwares ~~doc~~ must be documentes. Software documentation includes information such as the software type, the version, the time installed & the license agreement.
- Most OS have a utility that allows the documentation of user accounts & their privileges. The management must make sure that the files with this information are updated & secured.

ca

↳ Fault management

Comptr Nlw today are made up of hundreds & sometimes thousands of components, proper operation of the Nlw depends on the proper operation of each component individually & in relation to each other. Fault management is the area of Nlw management that handles this issue.

— An effective fault management system has two subsystems: reactive fault management & proactive fault management.

* Reactive fault management: It is responsible for detecting, isolating, correcting, & recording faults. It handles short-term solutions to faults.

steps:

1) To detect the exact location of the fault
2) To isolate the fault.
3) To correct the fault,
4) After fault is corrected, it must be documented.

* Proactive Fault management: Tries to prevent faults from occuring. Although this is not always possible, some types of failures can be predicted & prevented.

↳ Performance management

It is closely related to fault management, tries to monitor & control the Nlw to ensures that it is running as efficiently as possible.

↳ It tries to quantify performance by using some measurable quantity such as capacity, traffic, throughput, or response time.

Capacity → one factor that must be monitored by a performance management system is the capacity of the N/w. Every N/w has a limited capacity, and the performance management system must ensure that it is not used above this capacity.

Traffic → Traffic can be measured in two ways: internally and externally.

↳ Internal traffic is measured by the number of packets (or bytes) traveling inside the N/w.

↳ External traffic is measured by the exchange of packets (or bytes) outside the N/w.

Throughput

↳ We can measure the throughput of an individual device (such as a router) or a part of the N/w.

↳ Performance management monitors the throughput to make sure that it is not reduced to unacceptable levels.

Response Time

↳ Response time is normally measured from the time a user requests a service to the time the service is granted.

↳ Security Management

It is responsible for controlling access to the N/w based on the predefined policy.

data connection

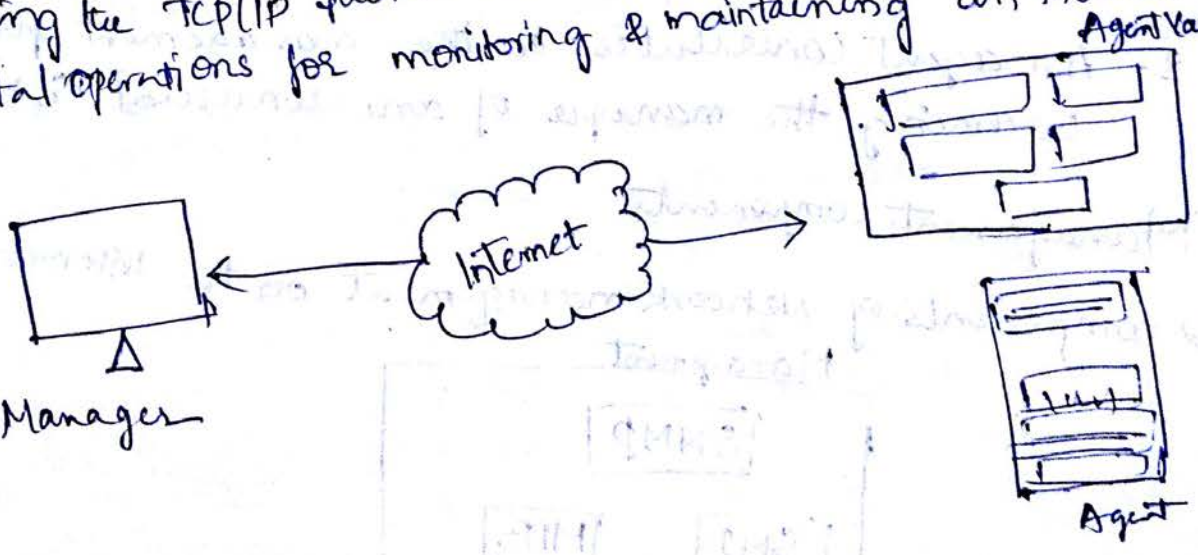
Accounting Management.

↳ Accounting Management is the control of user's access to N/w resources through charges. Under accounting management, individual users, departments, divisions, or even projects are charged for the services they receive from the N/w. Organizations use an accounting management system for the following reasons:

1) It prevents users from monopolizing limited N/w resources.
2) It prevents users from using the system inefficiently
3) N/w managers can do short- and long-term planning based on the demand for Network use.

SNMP (Simple N/w Management Protocol)

↳ SNMP is a framework for managing devices in an internet using the TCP/IP protocol suite. It provides a set of ~~fact~~ fundam ental operations for monitoring & maintaining an 'internet.



* Manager & Agents.

↳ A management station, called a manager, is a host that runs the SNMP client program.

↳ A managed station, called an agent, is a router (or a host) that runs the SNMP server program.

↳ Management is achieved through simple interaction b/w a manager & an agent.

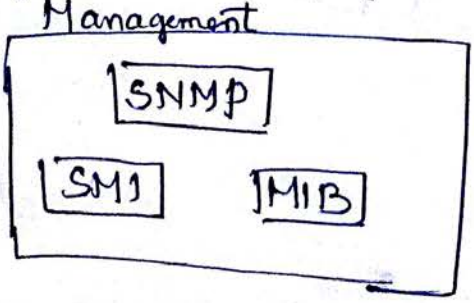
↳ Agents can also contribute to the management process. The server program running on the agent can check the environment, & if it notices something unusual, it can send a warning message, called a trap, to the manager.

↳ In other words, management with SNMP is based on three basic ideas:

1. A manager checks an agent by requesting information that reflects the behavior of the agent.
2. A manager forces an agent to perform a task by resetting values in the agent database.
3. An agent contributes to the management process by warning the manager of an unusual situation.

Management components

↳ Components of Network management on the Internet



data connection.

Role of SNMP

↳ SNMP defines the format of packets exchanged b/w a manager and an agent. It reads and changes the status (values) of objects (variables) in SNMP packets.

Role of SMI (Structure of Management Information)

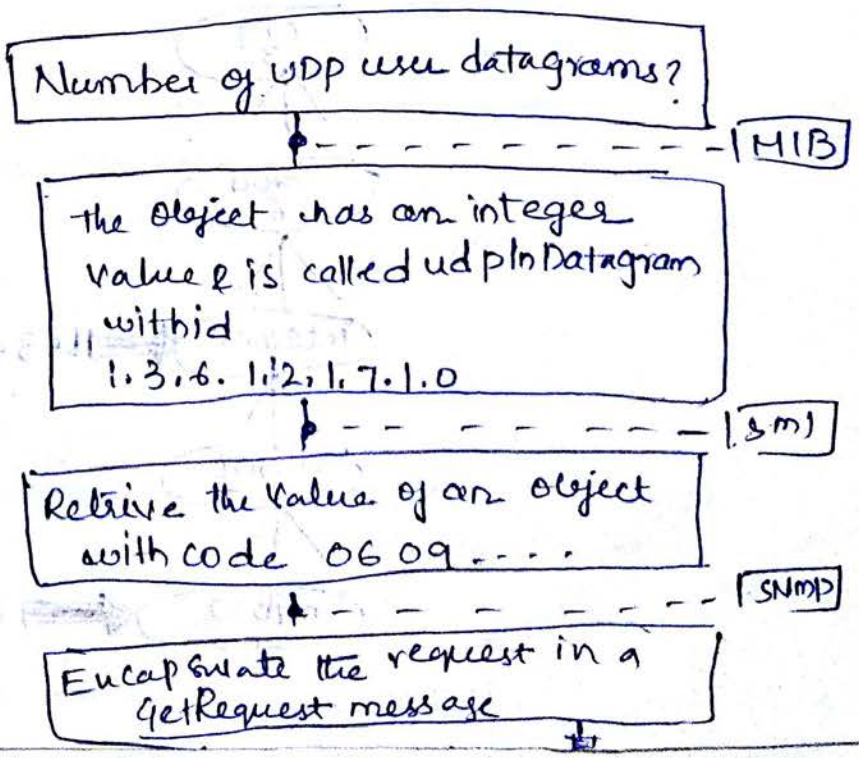
↳ SMI defines the general rules for naming objects, defining object types (including range and length), and showing how to encode objects & values.

↳ SMI does not define the number of objects an entity should manage or name the objects to be managed or define the association b/w the objects & their value.

Role of MIB (Management Information Base)

↳ MIB creates a collection of named objects, their types, and their relationships to each other in an entity to be managed.

* Management overview

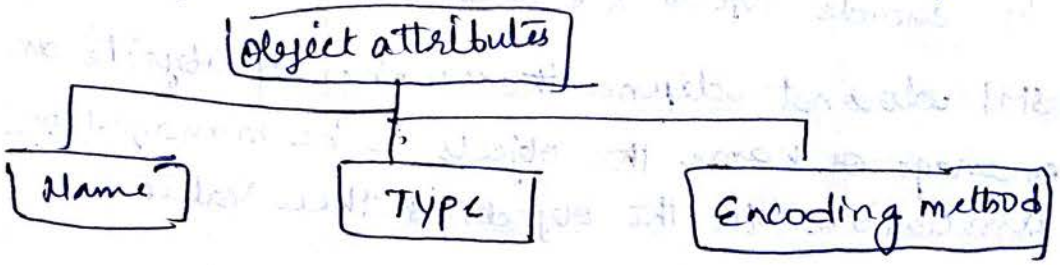


* Structure of Management Information.

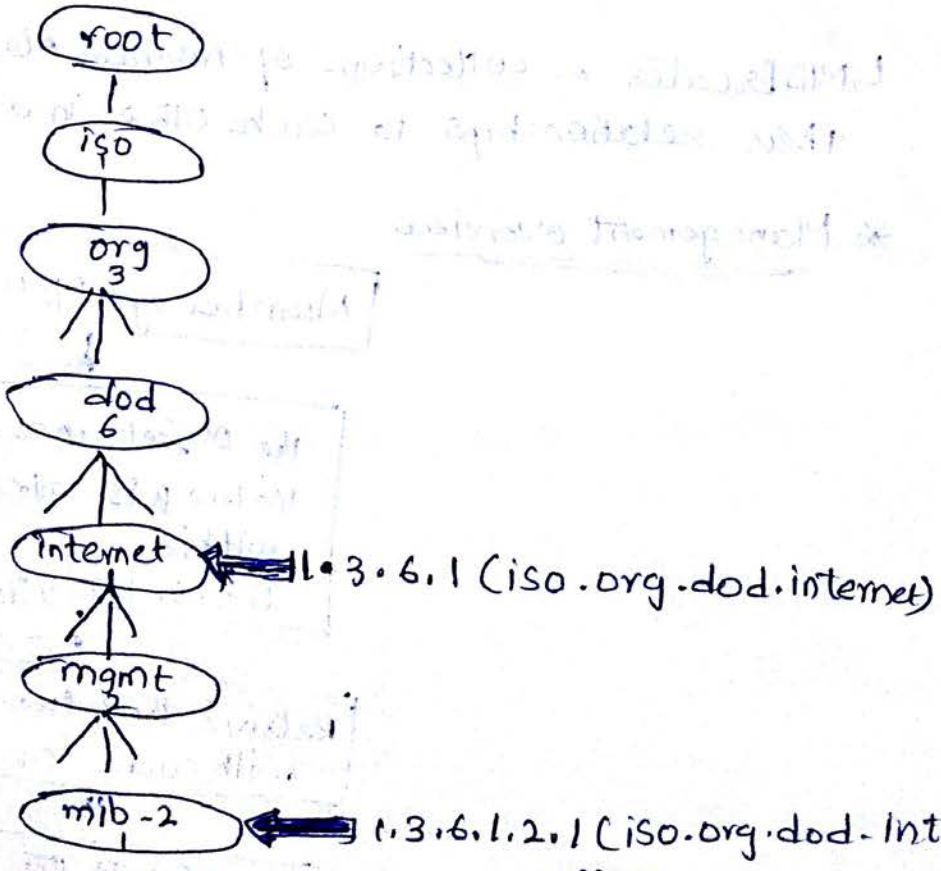
↳ The structure of Management Information, Version 2 (SMIv2) is compo-nent for network management. Its functions are

1. To name objects
2. To define the type of data that can be stored in an object
3. To show how to encode data for transmission over the N/w.

↳ SMI is a guideline for SNMP. It emphasizes three attributes to handle an object: name, datatype, and encoding method.



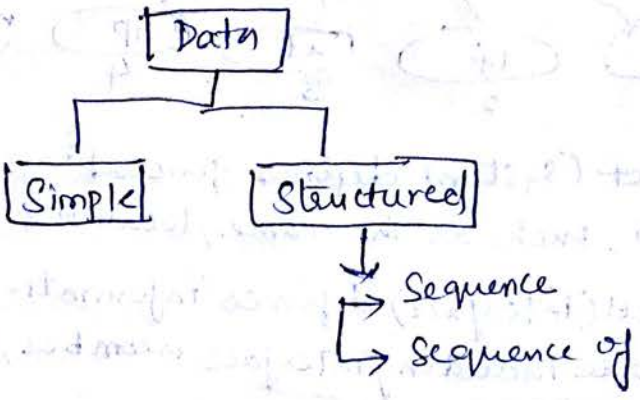
Object identifier



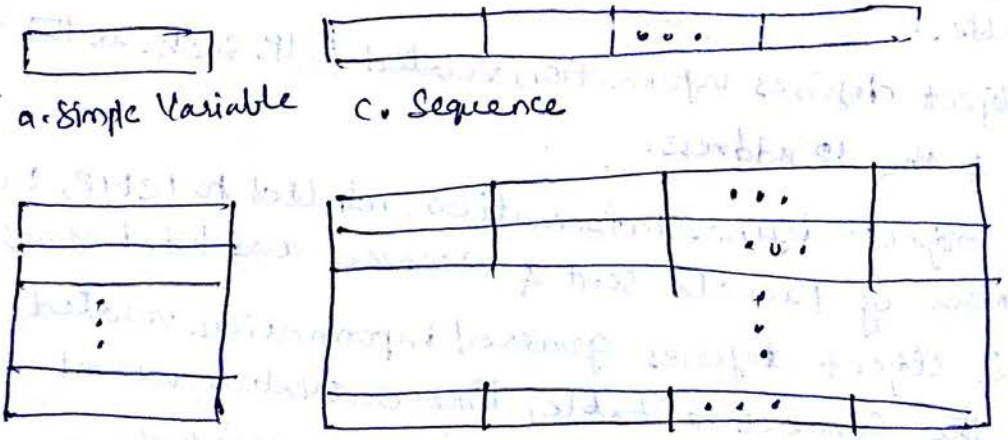
↳ iso.org.dod.internet.mgmt.mib-2 → 1.3.6.1.2.1

↳ All objects managed by SNMP are given an object identifier. The object identifier always starts with 1.3.6.1.2.1

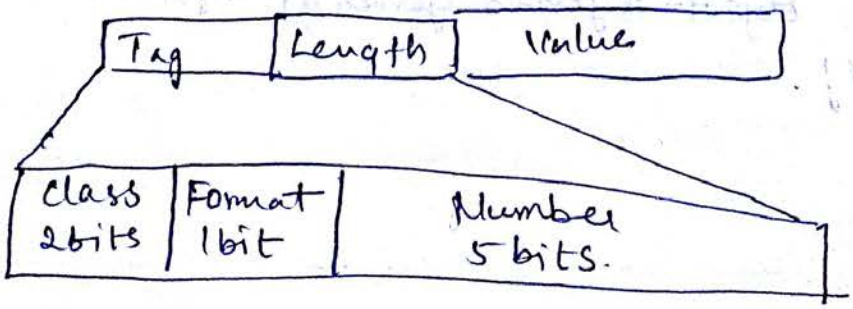
Type



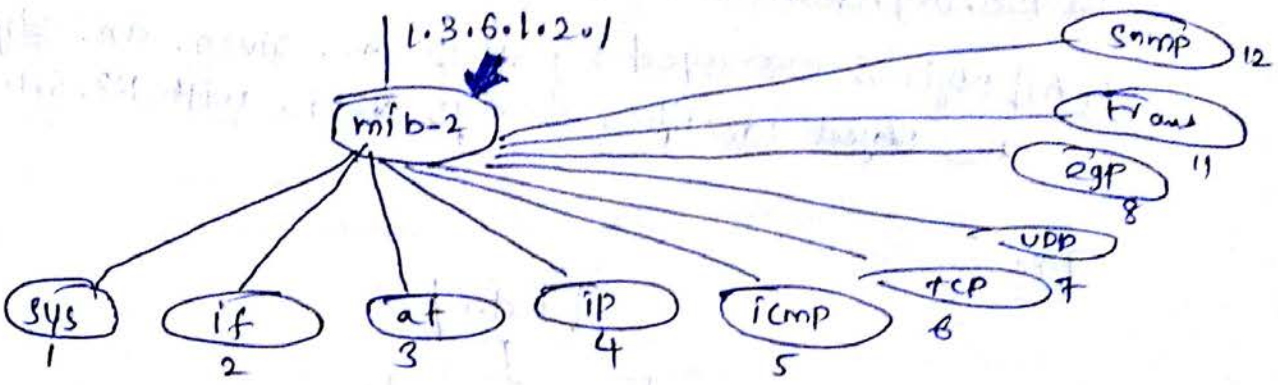
Conceptual data types



Encoding Method



Management Information Base (MIB)



sys: This object (System) defines general information about the node (system), such as the name, location & lifetime.

if: This object (Interface) defines information about all the interfaces of the node including interface number, physical address, & IP address.

at: This object (address translation) defines the information about the ARP table.

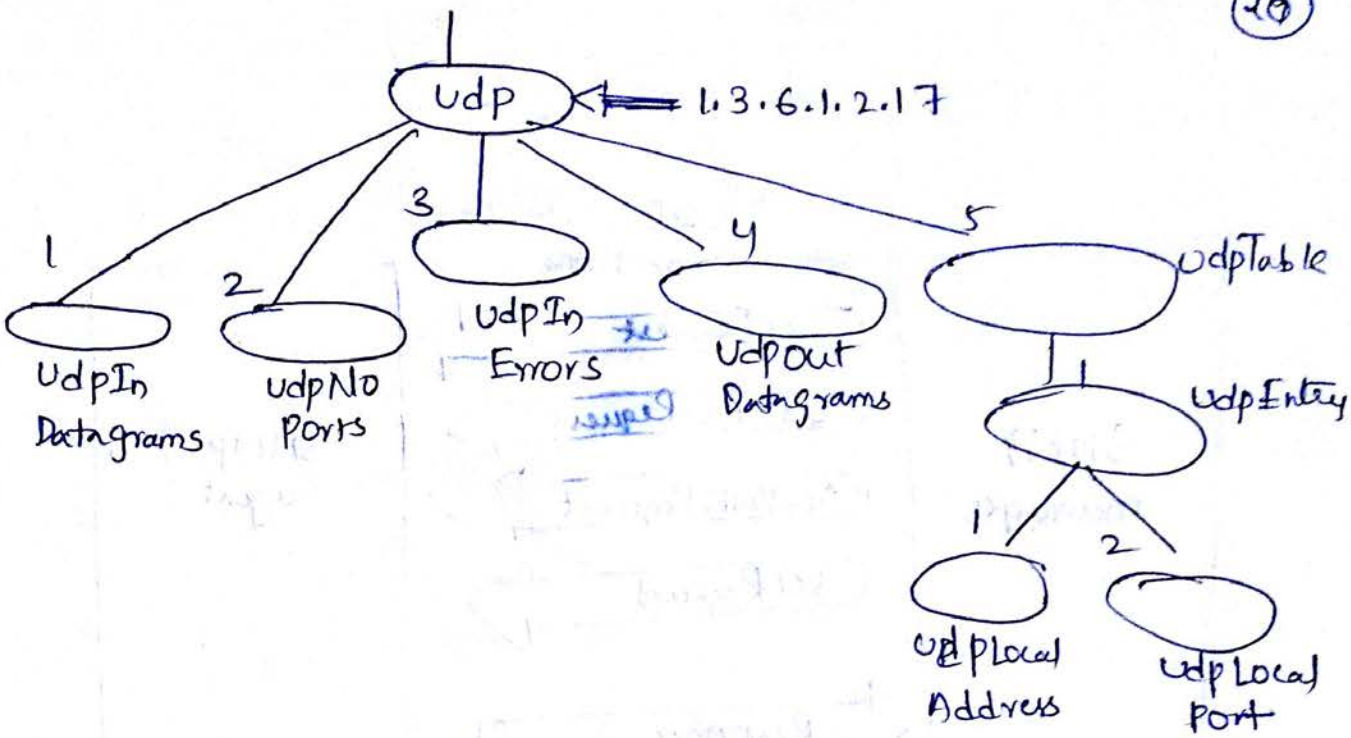
ip: This object defines information related to IP, such as the routing table & the IP address.

icmp: This object defines information related to ICMP, such as the number of packets sent & received and total errors created.

tcp: This object defines general information related to TCP, such as the connection table, time-out value, number of ports, and number of packets sent and received.

UdP: This object defines general information related to UDP, such as the number of ports & number of packets sent & received.

SNMP: This object defines general information related to SNMP itself.



SNMP

↳ SNMP uses both SMI and MIB in Internet network Management. It is an application program that allows

1. A manager to retrieve the value of an object defined in an agent
2. A manager to store a value in an object defined in an agent
3. An agent to send an alarm message about an abnormal situation to the managers.

PDUs

↳ SNMPv3 defines eight types of packets (or PDUs):
GetRequest, GetNextRequest, GetBulkRequest, setRequest, Response, Trap, InformRequest, and Report.

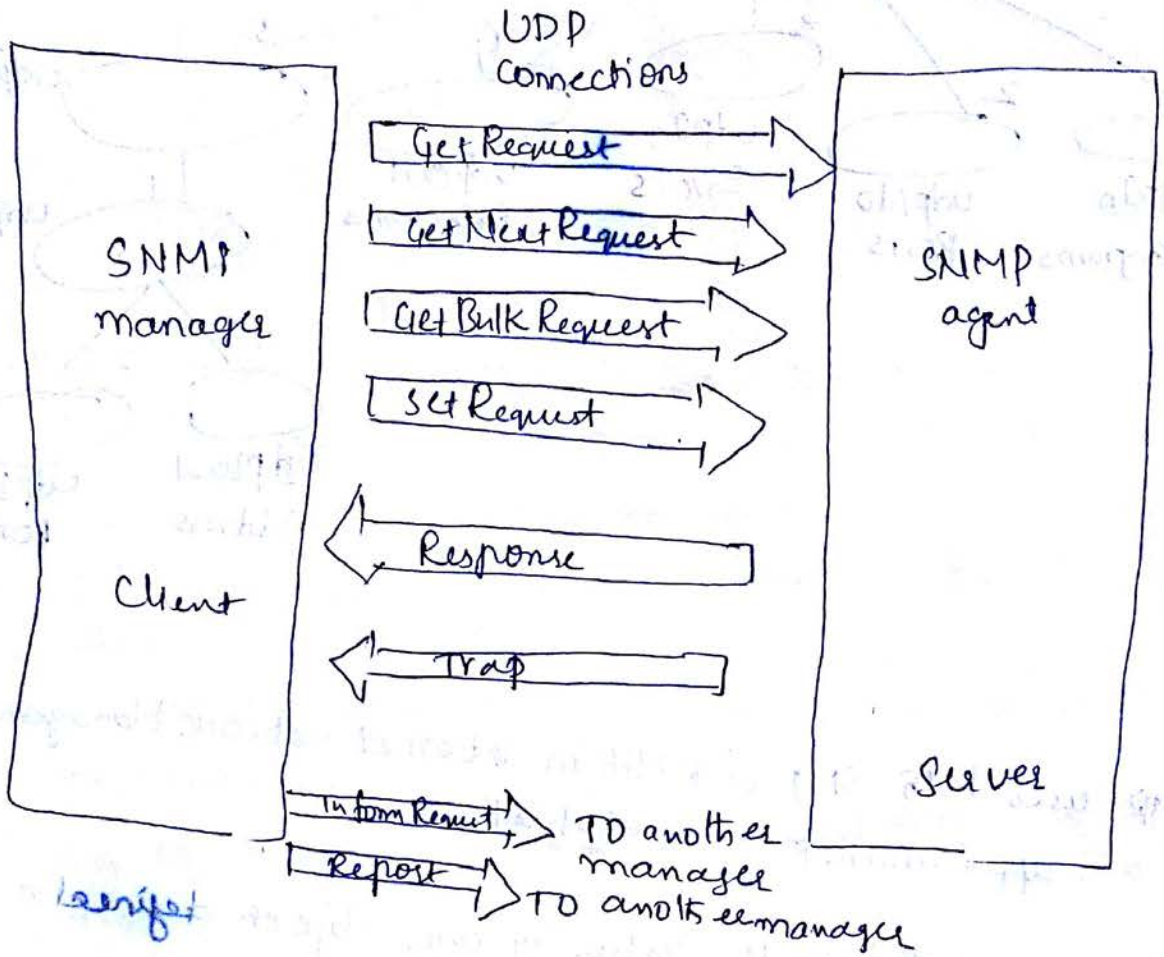
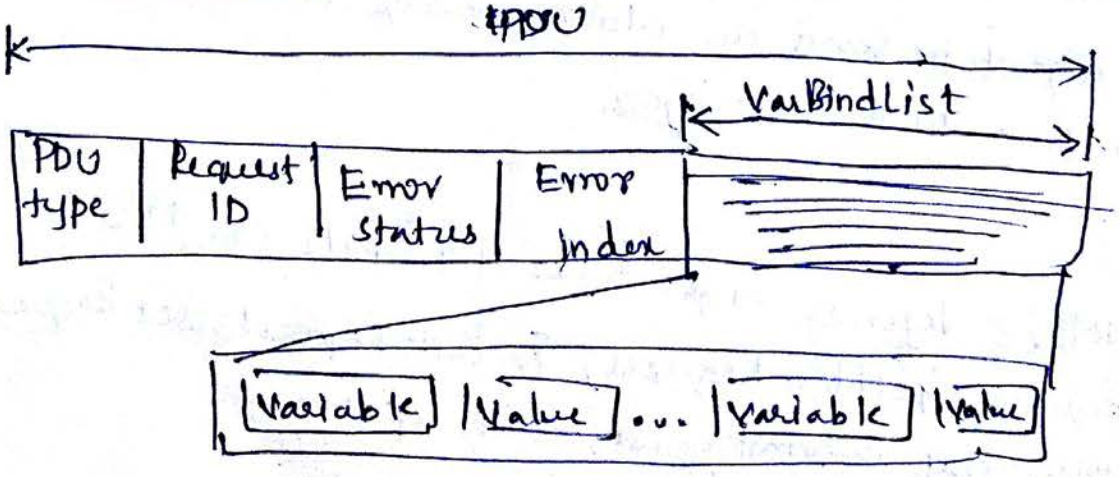
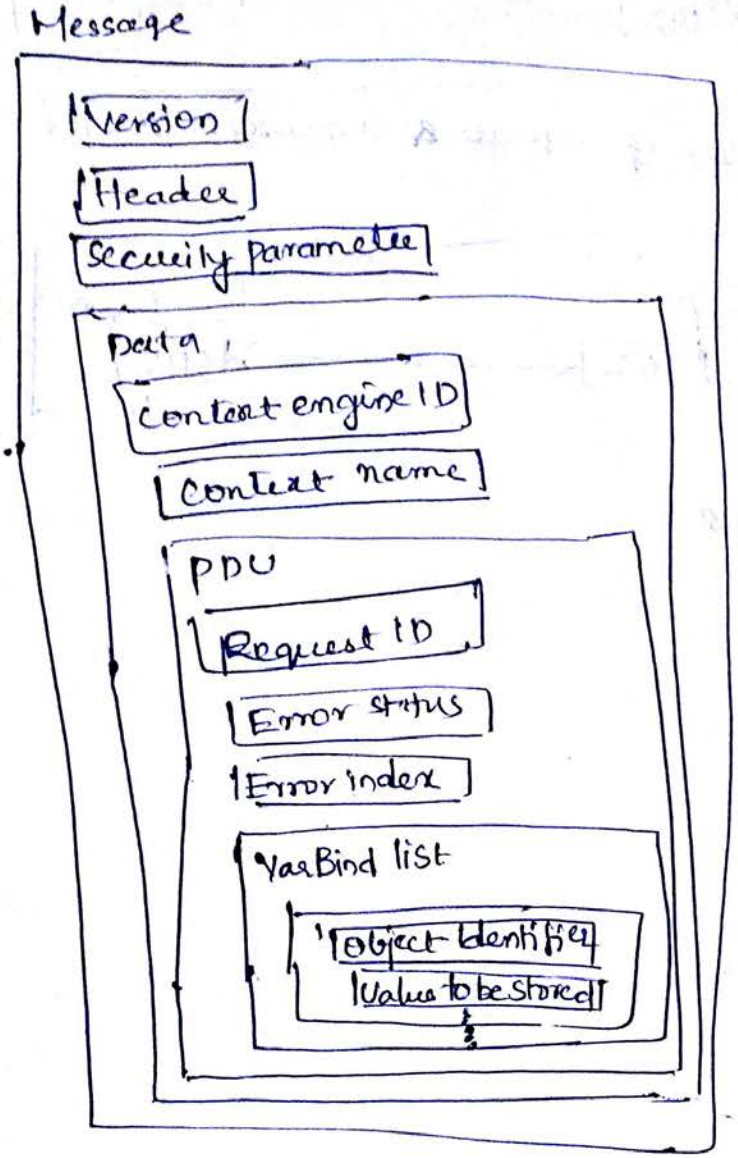


Fig: SNMP PDUs

SNMP PDU format

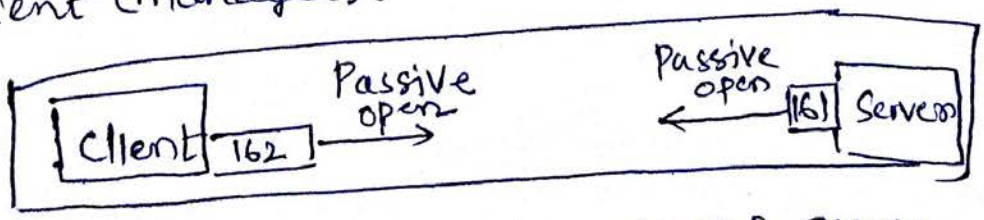


Messages

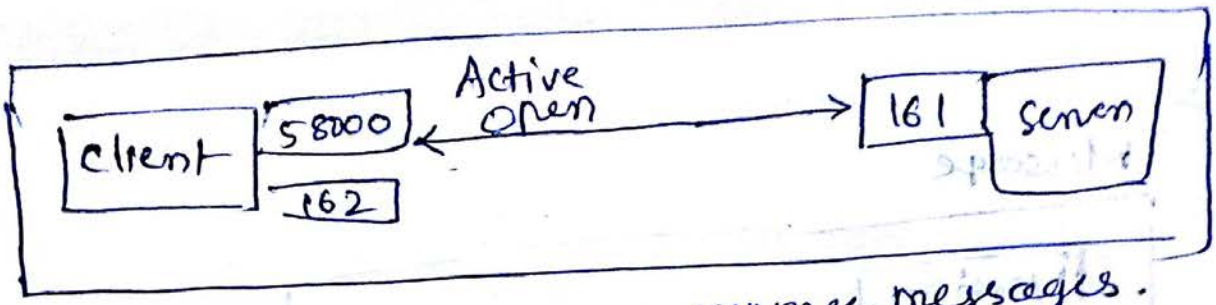


UDP ports

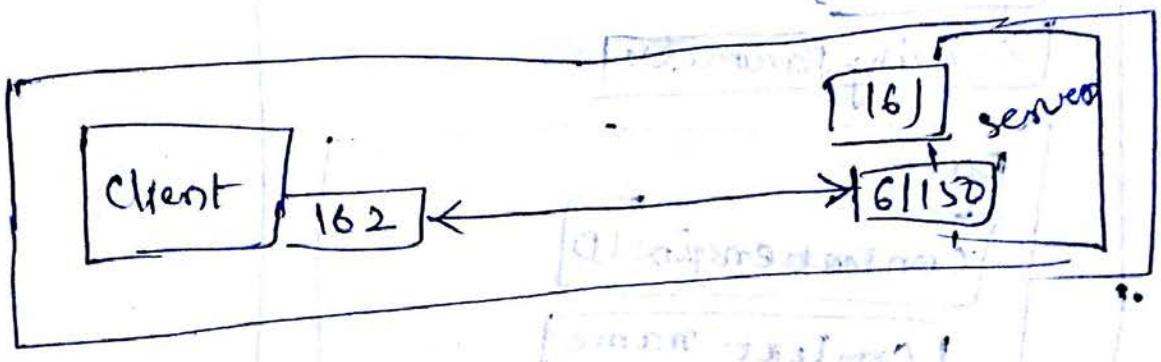
↳SNMP uses the services of UDP on two well-known ports, 161 & 162. The well-known port 161 is used by the server (agent), and the well-known port 162 is used by the client (manager).



a) Passive open by both client & server



b. Exchange of request & response messages.



c. Server