MODULE4

APPLICATION LAYER

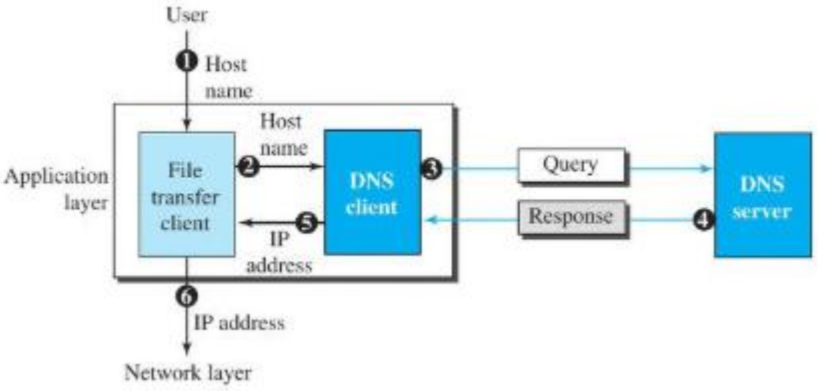
The application layer is responsible for providing services to the user.

The Domain Name System (DNS) is a supporting program that is used by other programs such as e-mail.

A DNS client/server program can support an e-mail program to find the IP address of an e-mail recipient.

DNS is used to map a host name in the application layer to an IP address in the network layer.

Figure 26.28 *Purpose of DNS*



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

- **flat or hierarchical**

1.Flat Name Space

In a flat name space, a name is assigned to an address. A name in this space is a sequence of characters without structure. The **main disadvantage** of a flat name space is that it cannot be used in a large system such as the Internet because it must be centrally controlled to avoid ambiguity and duplication.

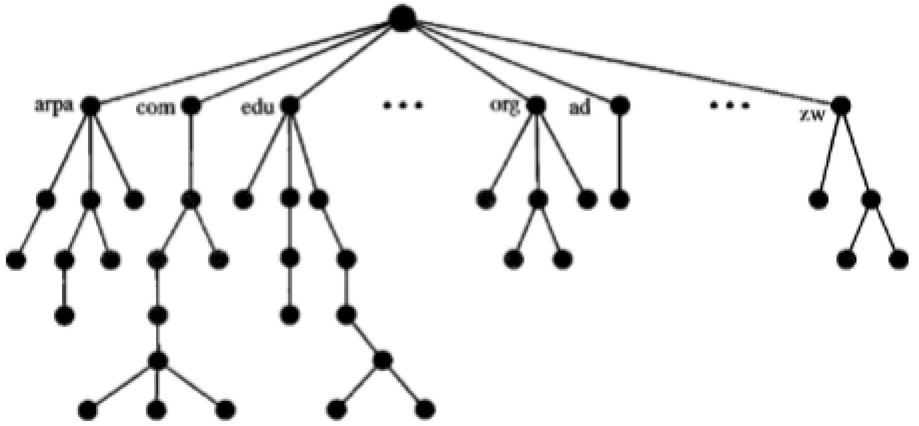
2.Hierarchical Name Space

In a hierarchical name space, each name is made of several parts. The first part can define the nature of the organization, the second part can define the name of an organization, the third part can define departments in the organization, and so on.

DOMAIN NAME SPACE

In this design the names are defined in an inverted-tree structure with the root at the top. The tree can have only 128 levels: level 0 (root) to level 127 (see Figure 25.2).

Figure 25.2 *Domain name space*



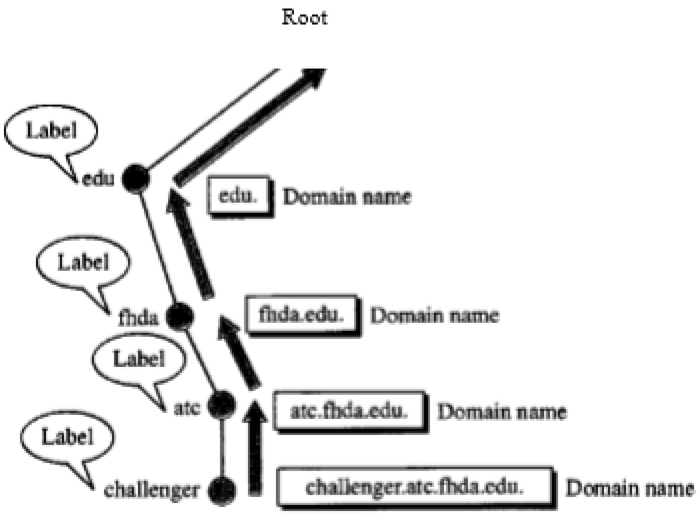
Label

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

Domain Name

Each node in the tree has a domain name. A full domain name is a sequence of labels separated by dots (.). The domain names are always read from the node up to the root. The last label is the label of the root (null). This means that a full domain name always ends in a null label, which means the last character is a dot because the null string is nothing.

Figure 25.3 *Domain names and labels*



Fully Qualified Domain Name

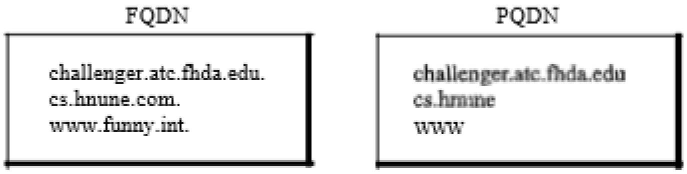
If a label is terminated by a null string, it is called a fully qualified domain name (FQDN). An FQDN is a Fully Qualified Domain Name.

Note that the name must end with a null label, but because null means nothing, the label ends with a dot (.).

Partially Qualified Domain Name

If a label is not terminated by a null string, it is called a partially qualified domain name (PQDN). A PQDN starts from a node, but it does not reach the root.

Figure 25.4 *FQDN and PQDN*

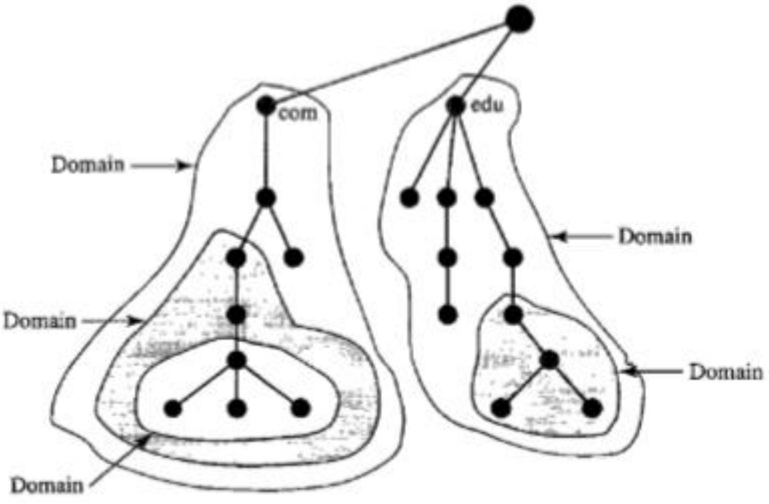


Domain

A domain is a **subtree** of the domain name space.

The name of the domain is the domain name of the node at the top of the subtree. Figure 25.5 shows some domains. Note that a domain may itself be divided into domains (or subdomains as they are sometimes called).

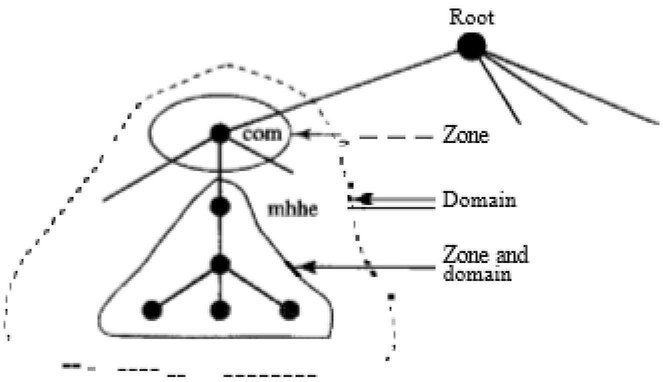
Figure 25.5 *Domains*



Zone

<u>The server makes a database</u> called a zone file and keeps all the information for every node under that domain.

Figure 25.7 *Zones and domains*



Root Server

A root server is a server whose zone consists <u>of the whole tree.</u>

Primary and Secondary Servers

DNS defines two types of servers: primary and secondary.

Primary server:It is responsible for creating, maintaining, and updating the zone file. It stores the zone file on a local disk.

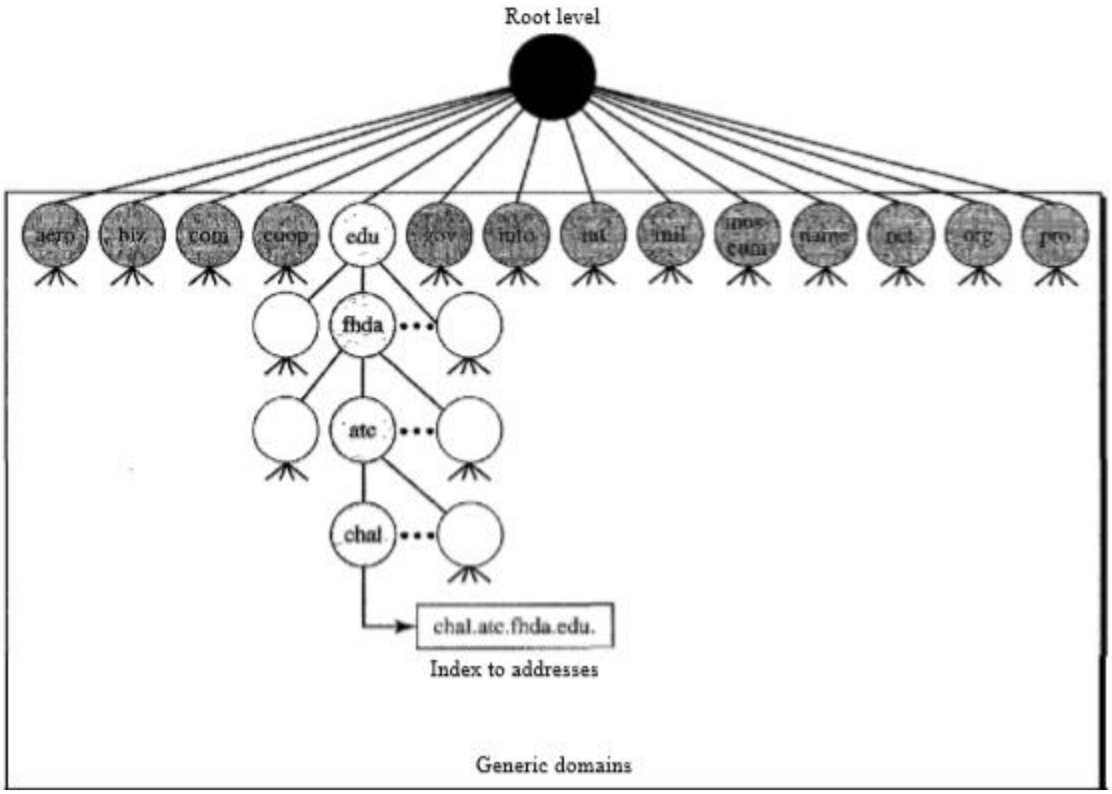
A secondary server is a server that transfers the complete information about a zone from another server (primary or secondary) and stores the file on its local disk.

DNS IN THE INTERNET

DNS is a protocol that can be used in different platforms. In the Internet, the domain name space (tree) is divided into three different sections: **generic domains, country domains, and the inverse domain** (see Figure 25.8).

The generic domains define registered hosts according to their generic behaviour.

Figure 25.9 *Generic domains*



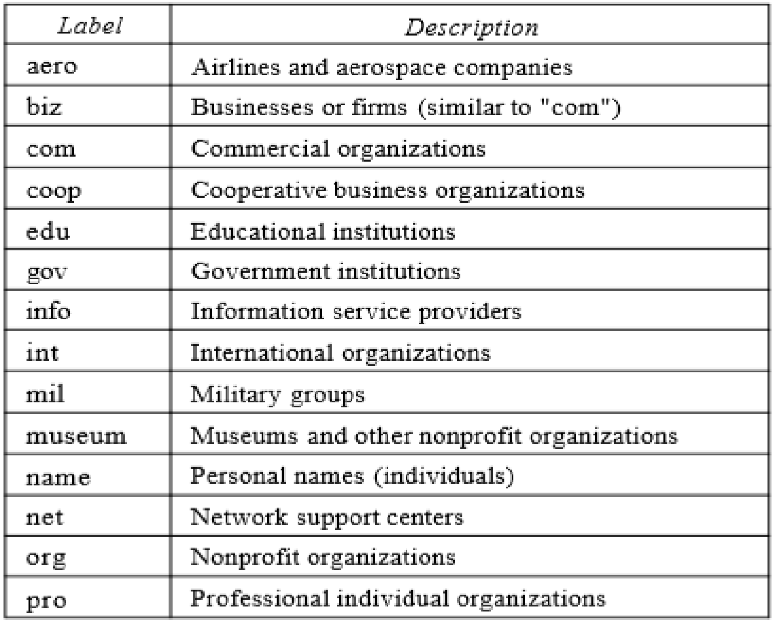
Country Domains

The country domains section uses two-character country abbreviations (e.g., us for United States). Second labels can be organizational, or they can be more specific, national designations. The United States, for example, uses state abbreviations as a subdivision of us (e.g., ca.us.).

Inverse Domain

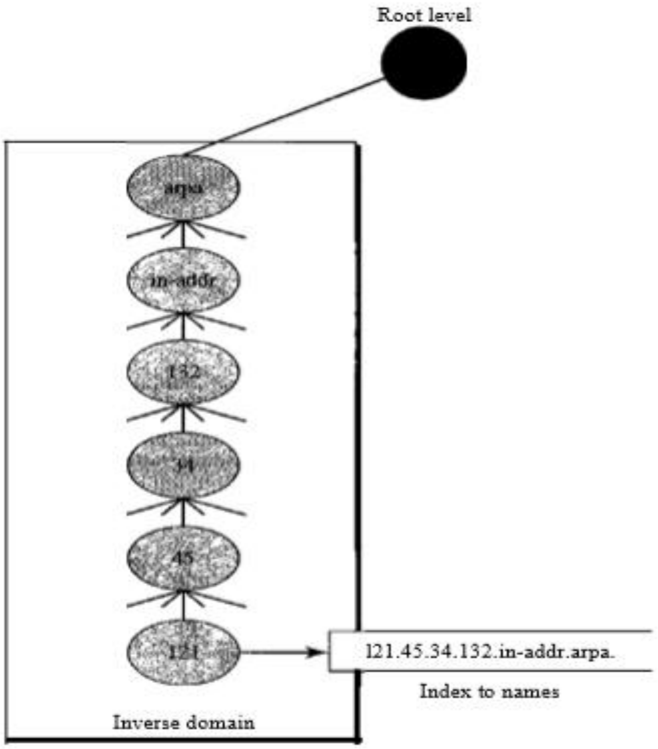
The inverse domain is used to map an address to a name.Fig:

Table 25.1 *Generic domain labels*



Label	*Description*
aero	Airlines and aerospace companies
biz	Businesses or firms (similar to "com")
com	Commercial organizations
coop	Cooperative business organizations
edu	Educational institutions
gov	Government institutions
info	Information service providers
int	International organizations
mil	Military groups
museum	Museums and other nonprofit organizations
name	Personal names (individuals)
net	Network support centers
org	Nonprofit organizations
pro	Professional individual organizations

Figure 25.11 *Inverse domain*



RESOLUTION

Mapping a name to an address or an address to a name is called name-address resolution.

Resolver

DNS is designed as a client/server application. A host that needs to map an address to a name or a name to an address calls **a DNS client called a resolver**.

Mapping Addresses to Names

A client can send an IP address to a server to be mapped to a domain name.

Mapping Names to Addresses

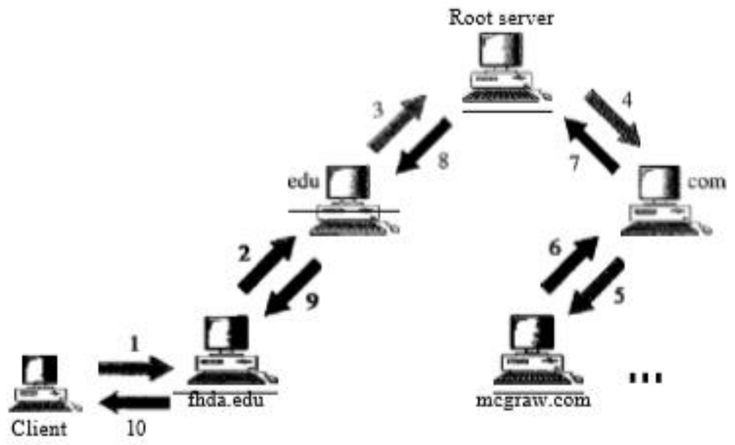
Most of the time, the resolver gives a domain name to the server and asks for the corresponding address.

Resolution:- It is the process of translating IP addresses to domain names

1.Recursive Resolution

The client (resolver) can ask for a recursive answer from a name server. This means that the resolver expects the server to supply the final answer. If the server is the authority for the domain name, it checks its database and responds. If the server is not the authority, it sends the request to another server (the parent usually) and waits for the response. If the parent is the authority, it responds; otherwise, it sends the query to yet another server. When the query is finally resolved, the response travels back until it finally reaches the requesting client. This is called recursive resolution and is shown in Figure 25.12.request:-mcgraw.com

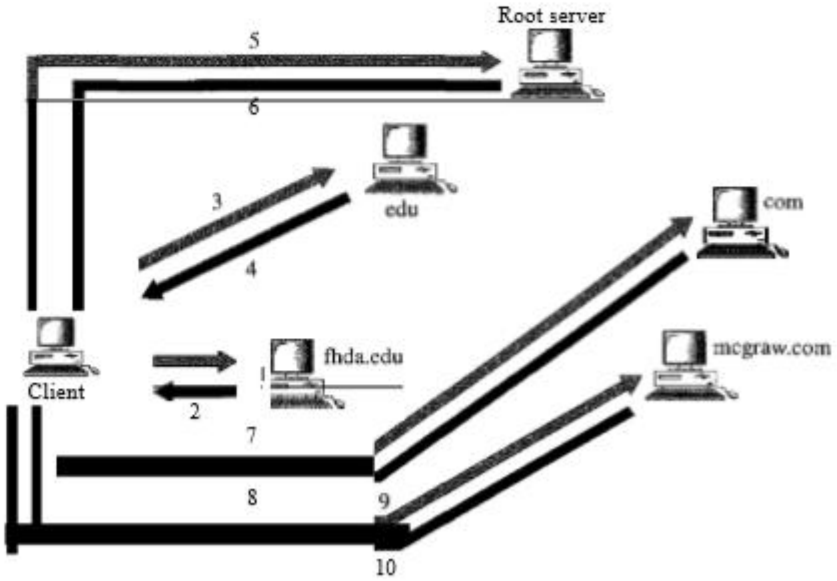
Figure 25.12 *Recursive resolution*



2.Iterative Resolution

If the client does not ask for a recursive answer, the **mapping can be done iteratively**. If the server is an authority for the name, it sends the answer. If it is not, it returns (to the client) the IP address of the server that it thinks can resolve the query. The client is responsible for repeating the query to this second server.eg:-mcgraw.com

Figure 25.13 *Iterative resolution*



DNS MESSAGES

DNS has two types of messages**: query and response.**

Both types have the same format.

The query message consists of a header and question records.

The response message consists of a header, question records, answer records, authoritative records, and additional records

Figure 25.14 *Query and response messages*

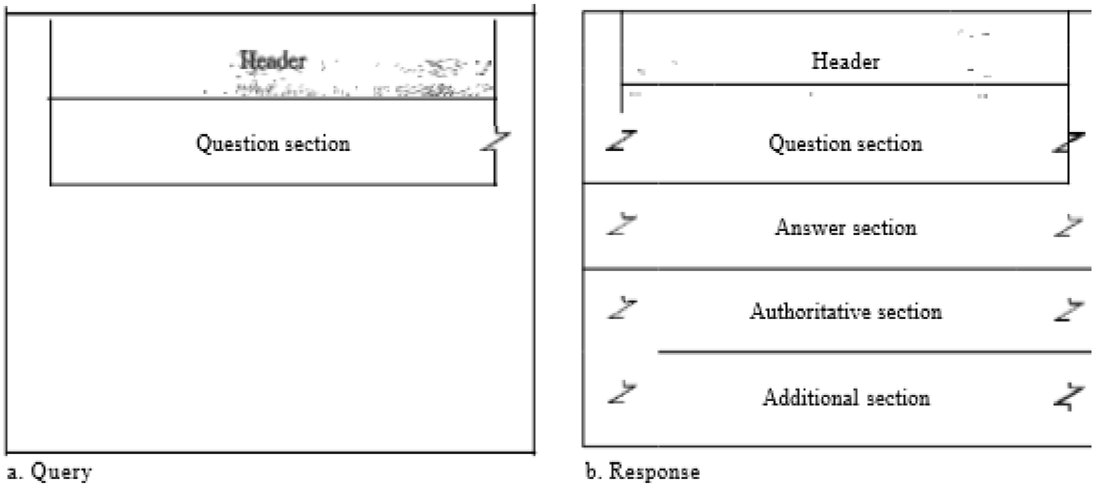
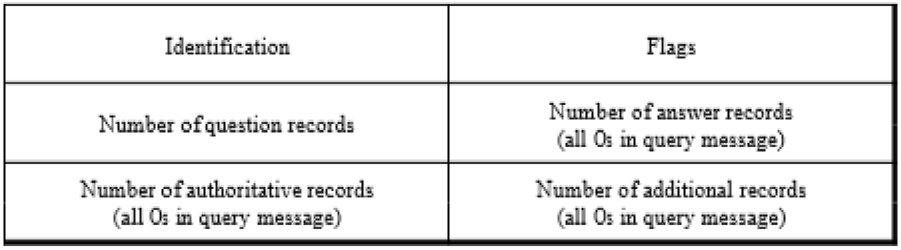


Figure 25.15 *Header format*



Identification	Flags
Number of question records	Number of answer records (all 0s in query message)
Number of authoritative records (all 0s in query message)	Number of additional records (all 0s in query message)

Header

Both query and response messages have the same header format with some fields set to zero for the query messages. The header is 12 bytes.

The identification subfield is used by the client to match the response with the query. The client uses a different identification number each time it sends a query.

The flags subfield is a collection of subfields that define the type of the message.

The number of question records subfield contains the number of queries in the question section of the message.

The number of answer records subfield contains the number of answer records in the answer section of the response message.

The number of authoritative records subfield contains the number of authoritative records in the authoritative section of a response message.

The number of additional records subfield contains the number additional records in the additional section of a response message. Its value is zero in the query message.

Question Section

This is a section consisting of one or more question records. It is present on both query and response messages.

Answer Section

This is a section consisting of one or more resource records. It is present only on response messages. This section includes the answer from the server to the client (resolver).

Authoritative Section

This is a section consisting of one or more resource records. It is present only on response messages.

Additional Information Section

This is a section consisting of one or more resource records. It is present only on response messages. This section provides additional information that may help the resolver.

TYPES OF RECORDS

Two types of records are used in DNS.

The **question records** are used in the question section of the query and response messages.

The **resource records** are used in the answer, authoritative, and additional information sections of the response message.

-Question Record

A question record is used by the client to get information from a server. This contains the domain name.

-Resource Record

Each domain name (each node on the tree) is associated with a record called the resource record. The server database consists of resource records. Resource records are also what is returned by the server to the client.

DYNAMIC DOMAIN NAME SYSTEM (DDNS)

When the 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, or changing an IP address, the change must be made to the DNS master file. These types of changes involve a lot of manual updating.

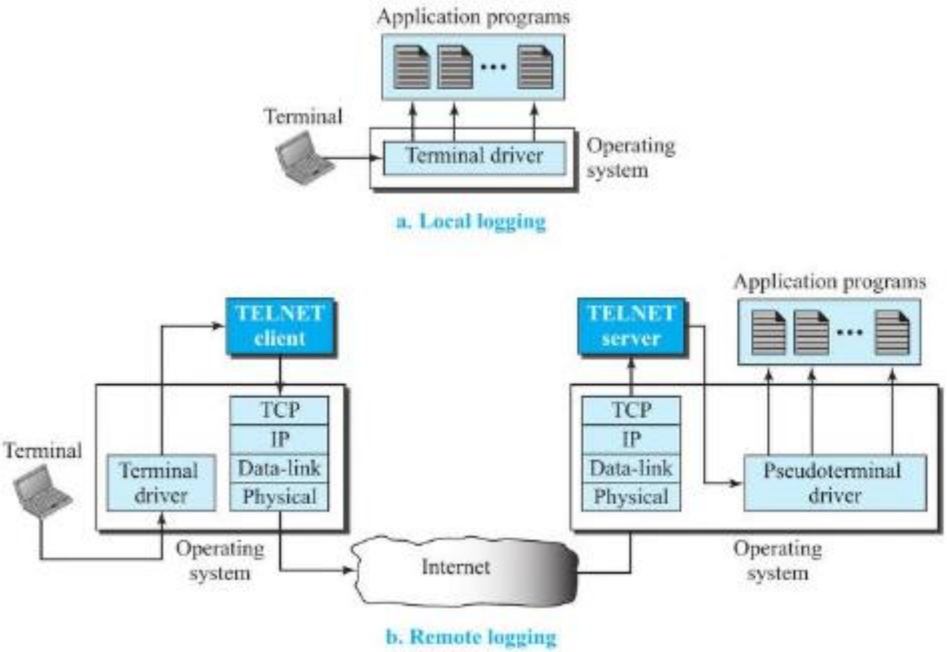
The DNS master file must **be updated dynamically** is called Dynamic Domain Name System (DDNS).

TELNET is an abbreviation for **TErminaL NETwork**. It is the standard TCP/IP protocol for virtual terminal service as proposed by the International Organization for Standards (ISO).

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

- It allows a user at one site to establish a TCP connection to a login server or terminal server at another site.
- A TELNET server generally listens on TCP Port 23.

Figure 26.23 *Local versus remote logging*



a. Local logging

b. Remote logging

When a user logs into a local system is called local logging.

One of the original remote logging protocol is TELNET.

TELNET requires a logging name and password.

For the connections, TELNET uses the TCP protocol.

The TELNET service is offered in the host machine's TCP port 23.

The user at the terminal interacts with the local telnet client.

The TELNET client acts as a terminal accepting any keystrokes from the keyboard, interpreting them and displaying the output on the screen. The client on the computer makes the TCP connection to the host machine's port 23 where the TELNET server answers.

The TELNET server interacts with applications in the host machine and assists in the terminal emulation.

The Telnet *Network Virtual Terminal (NVT)* is a uniform data representation that ensures the compatibility of communication between terminals and hosts that may use very different hardware, software and data formats.

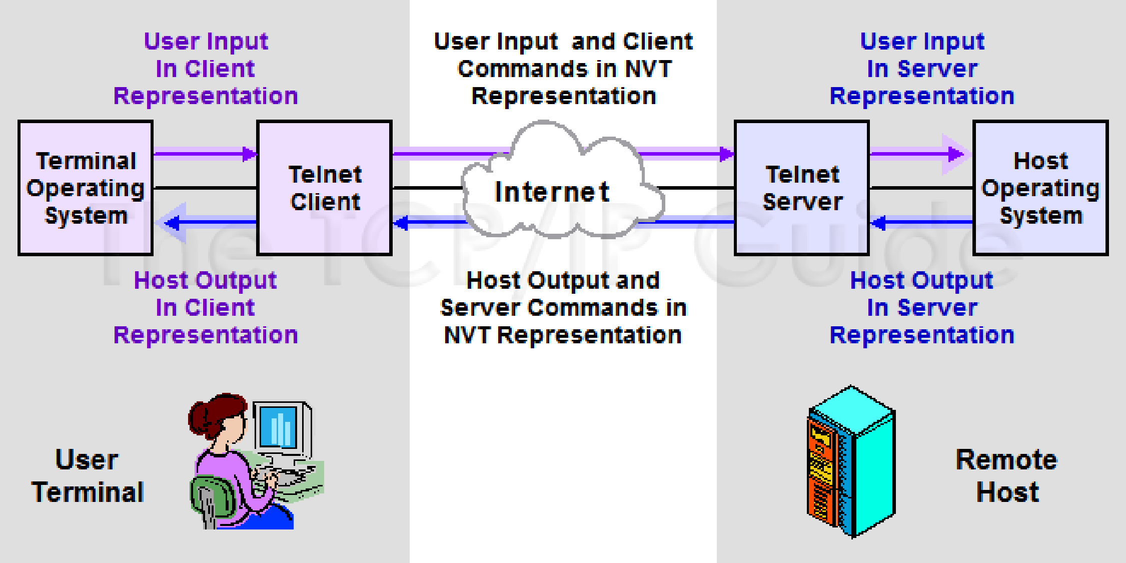


Figure 320: Telnet Communication and the Network Virtual Terminal (NVT)

Telnet uses the Network Virtual Terminal (NVT) representation to allow a user terminal and remote host that use different internal formats to communicate.

Electronic mail(e-mail)

One of the most popular internet services is email.

E-mail allows users to exchange messages. Email considered a one-way transaction.

Architecture

To explain the architecture of e-mail, we give a common scenario, as shown in figure:-

he general architecture of an e-mail system including the three main components: user agent, message transfer agent , and message access agent.

In the common scenario ,the sender and the receiver of the e-mail ,Alice and Bob respectively, are connected via a LAN or a WAN to two mail servers.

The administrator has created one mailbox for each user where the received messages are stored.

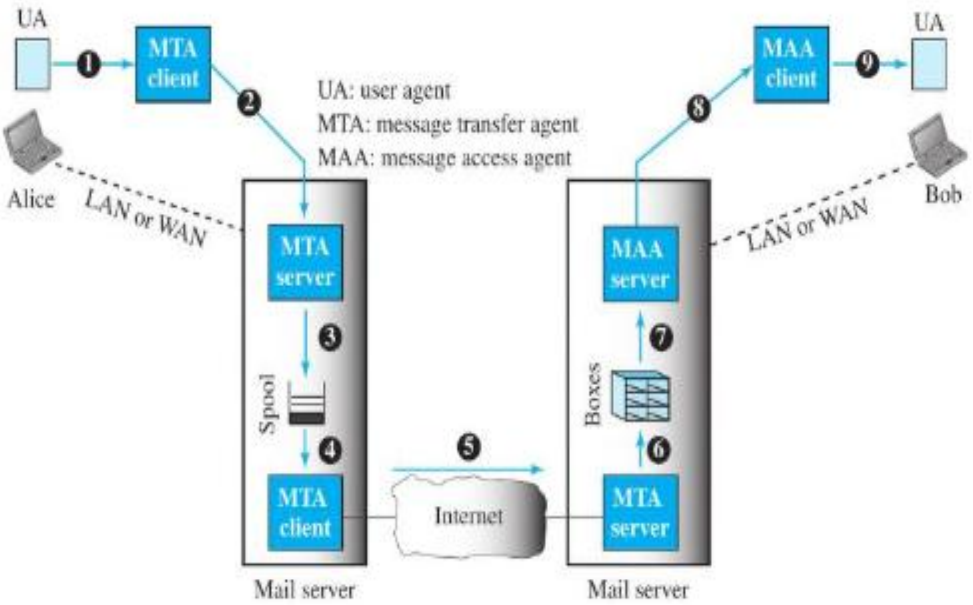
A mailbox is a part of a sever hard drive,a special file with permission restrictions.Only the owner of the mailbox has access to it.

<u>The administrator has also created a queue (spool</u>) to store messages waiting to be sent.

A simple e-mail from Alice to Bob takes different steps as shown in figure below.

Alice and Bob use Three different agents:-**<u>a user agent(UA),a message transfer agent(MTA),, and a message access agent(MAA).</u>**

Figure 26.12 *Common scenario*



1.when Alice needs to send a message to Bob,she run UA program to prepare the message and send it to her mailserver.

2.The mail server at her site uses a queue(spool) to store messages waiting to be sent.

3.The message needs to be sent through the internet from Alice's site to Bob's site using an MTA.

4.Here two message transfer agents are needed:one client and one server.

5.The UA at the Bob site allows Bob to read the received message. Bob later uses an MAA client to retrieve message from an MAA server running on the second server.

6.Bob cannot bypass the mail server and use the MTA server directly. To use MTA server directly,Bob would need to run the MTA server all the time because he does not know when a message will arrive.

7.Bob needs another pair of client-server programs: message access programs.

8.MTA client-server program is a push program: the **client pushes the message to the server.**

9.Bob needs a pull program. **the client needs to pull the message from the server**.

The electronic mail system needs two UAs,two pairs of MTAs (client and server),and a pair of MAAs(client and server).

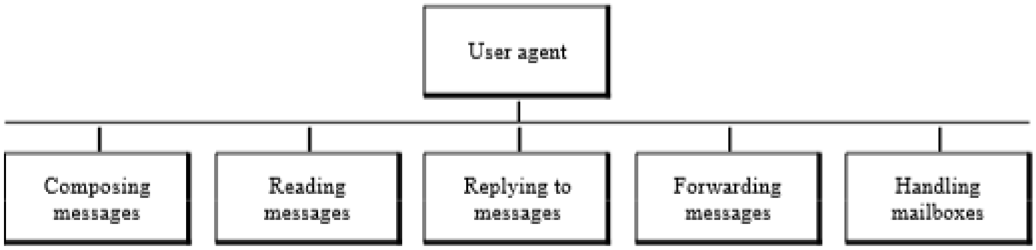
User Agent

The first component of an electronic mail system is the user agent (VA). It provides service to the user to make the process of sending and receiving a message easier.

Services Provided by a UserAgent

A user agent is a software package (program) that composes, reads, replies to, and forwards messages. Italso handles mailboxes. Figure 26.11 shows the services of a typical user agent.

Figure 26.11 *Services ofuser agent*



User Agent Types

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

Command-Driven Command-driven user agents belong to the early days of electronic mail.

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

GUI-Based Modem user agents are GUI-based. They contain graphical-user interface (GUI) components that allow the user to interact with the software by using both the keyboard and the mouse.

Sending Mail

To send mail, the user, through the UA, creates mail that looks very similar to postal mail. Ithas an envelope and a message (see Figure 26.12)

Envelope :The envelope usually contains the sender and the receiver addresses.

Message: The message contains the header and the body. The header of the message defines the sender, the receiver, the subject of the message, and some other information (such as encoding type, as we see shortly). The body of the message contains the actual information to be read by the recipient.

Receiving Mail: If the user is ready to read the mail,a list is displayed in which each line contains a summary of the information about a particular message in the mailbox.

Figure 26.16 *Example 26.12*

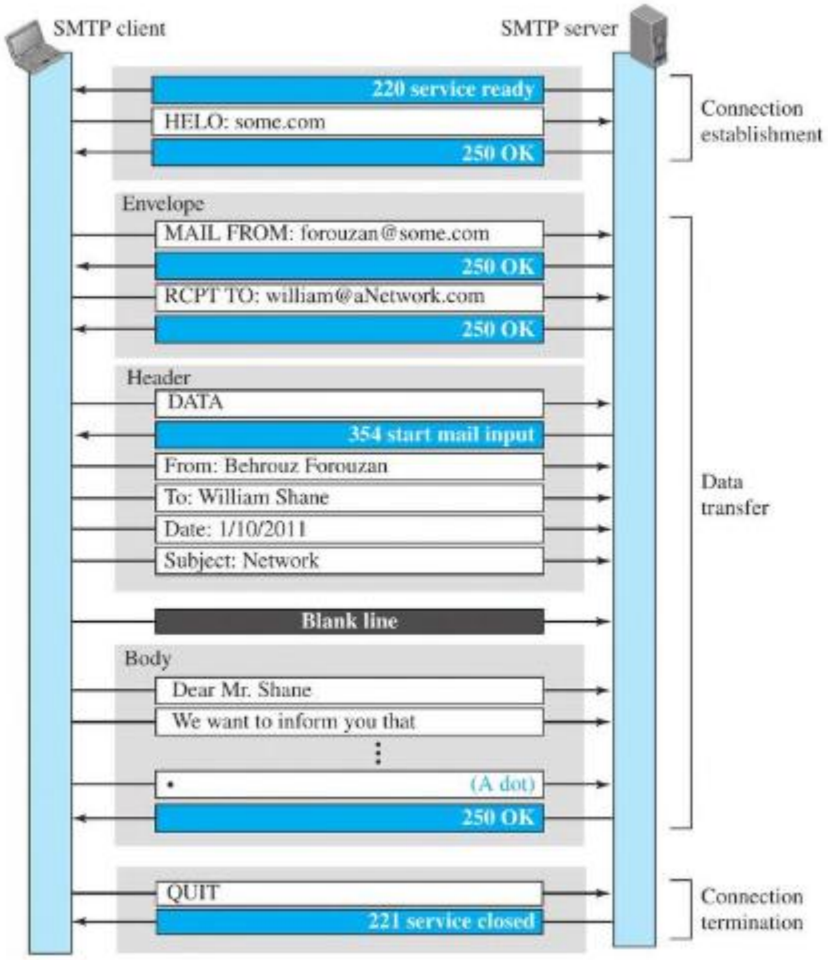
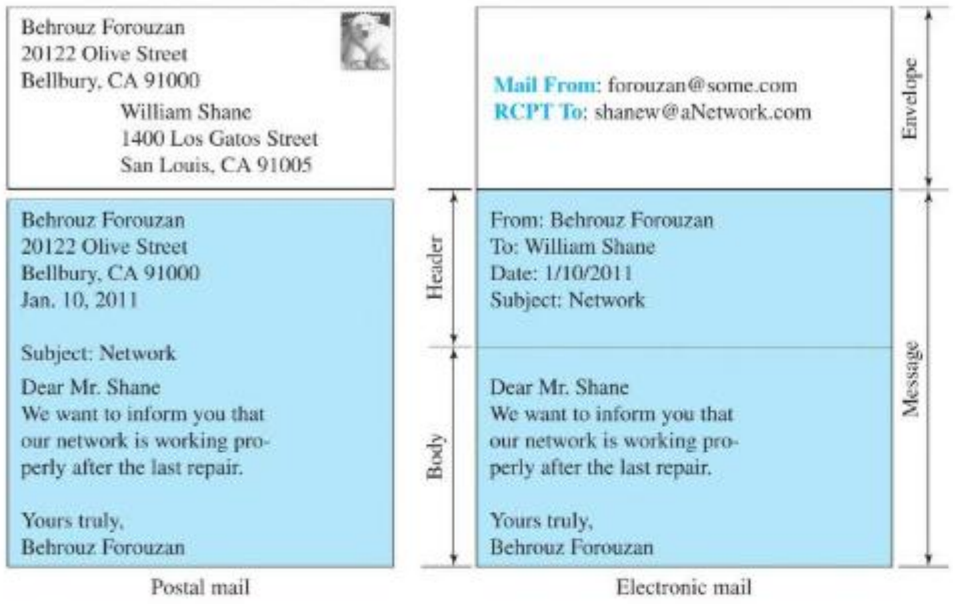


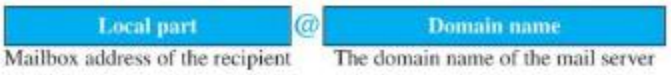
Figure 26.13 *Format of an e-mail*



Addresses

To deliver mail, a mail handling system must use an addressing system with unique addresses. In the Internet, the address consists of **two parts: a local part and a domain name, separated by an @ sign** (see Figure 26.13).

Figure 26.14 *E-mail address*


Local part
@
Domain name
Mailbox address of the recipient
The domain name of the mail server

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

Domain Name : The second part of the address is the domain name. An organization usually selects one or more hosts to receive and send e-mail; the hosts are sometimes called mail servers or exchangers.

Mailing List: Electronic mail allows one name, an alias, to represent several different e-mail addresses; this is called a mailing list.

Advantages:-Inexpensive,Speed,Reliable,global.

Disadvantages:-junk(spam),Forgery.

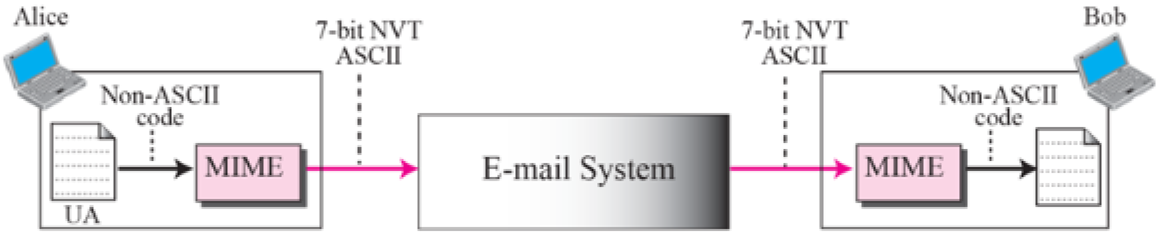
MIME

Electronic mail has a simple structure. It can send messages only in NVT 7-bit ASCII format.

Multipurpose Internet Mail Extensions (MIME) is a supplementary protocol that allows non-ASCII data to be sent through e-mail.

MIME transforms **non-ASCII data** at the sender site to **NVT ASCII data** and delivers them to the client MTA to be sent through the Internet. The message at the receiving side is transformed back to the original data.

We can think of MIME as a set of software functions that transforms non-ASCII data (stream of bits) to ASCII data and vice versa, as shown in Figure :



MIME defines **five headers** that can be added to the **original e-mail header** section to define the transformation parameters:

1. MIME-Version

2. Content-Type

3. Content-Transfer-Encoding

4. Content-Id

5. Content-Description

MIME-Version :-This header defines the version of MIME used. The current version is 1.1.

Content-Type:- This header defines the type of data used in the body of the message.

Content-Transfer-Encoding :-This header defines the method used to encode the messages into Os and Is for transport:

Content-Id :-This header uniquely identifies the whole message in a multiple-message environment.

Content-Description:- This header defines whether the body is image, audio, or video.

Figure 26.15 *MIME header*

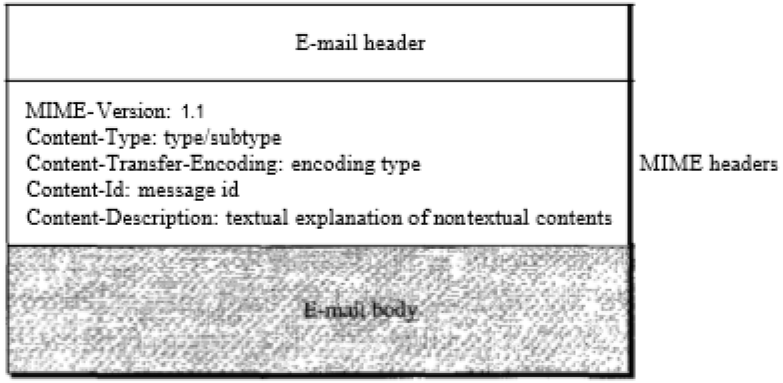
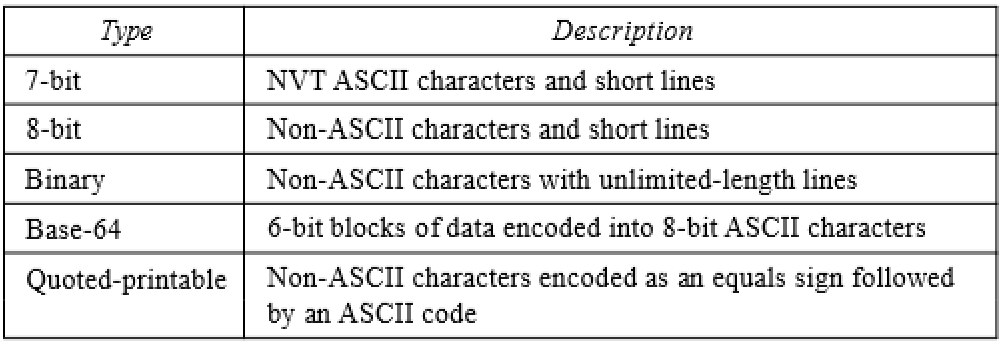

E-mail header
MIME-Version: 1.1
Content-Type: type/subtype
Content-Transfer-Encoding: encoding type
Content-Id: message id
Content-Description: textual explanation of nontextual contents
MIME headers
E-mail body

Table 26.6 *Content-transfer-encoding*



Type	*Description*
7-bit	NVT ASCII characters and short lines
8-bit	Non-ASCII characters and short lines
Binary	Non-ASCII characters with unlimited-length lines
Base-64	6-bit blocks of data encoded into 8-bit ASCII characters
Quoted-printable	Non-ASCII characters encoded as an equals sign followed by an ASCII code

Message Transfer Agent: SMTP

The actual mail transfer is done through **message transfer agents**. To send mail, a system must have the client MTA, and to receive mail, a system must have a server MTA.

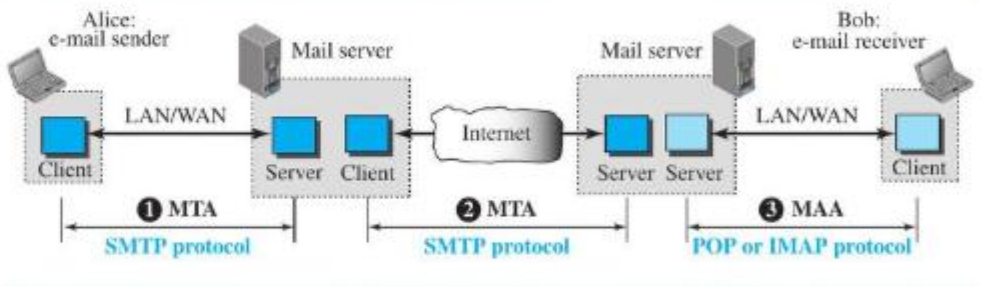
The formal protocol that defines the **MTA client and server** in the Internet is called the **Simple Mail Transfer Protocol (SMTP).**

SMTP is **used two times**, **between the sender and the sender's mail server** and **between the two mail servers.**

Commands and Responses

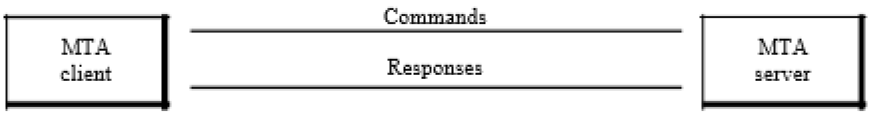
SMTP uses commands and responses to transfer messages between an **MTA client and an MTA server (see Figure 26.17).**

Figure 26.15 *Protocols used in electronic mail*



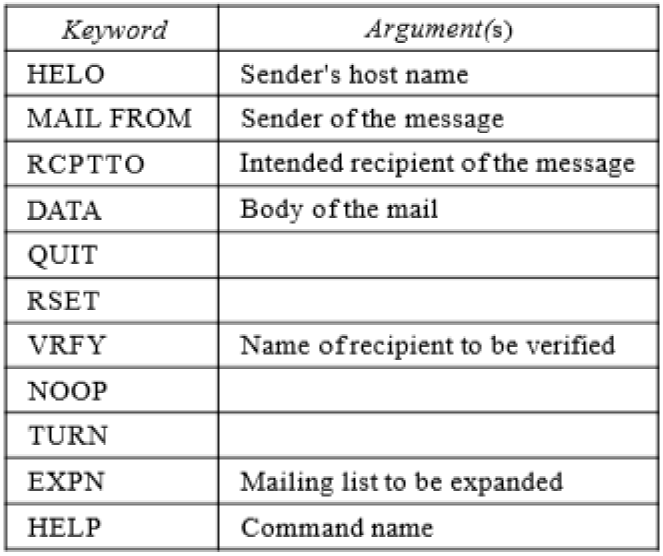
Each command or reply is terminated by **a two-character** (carriage return and line feed) end-of-line token.

Figure 26.17 *Commands and responses*



Commands :-Commands are sent from the client to the server. It consists of a keyword followed by zero or more arguments. SMTP defines 14 commands. The first five are mandatory.

Table 26.7 *Commands*

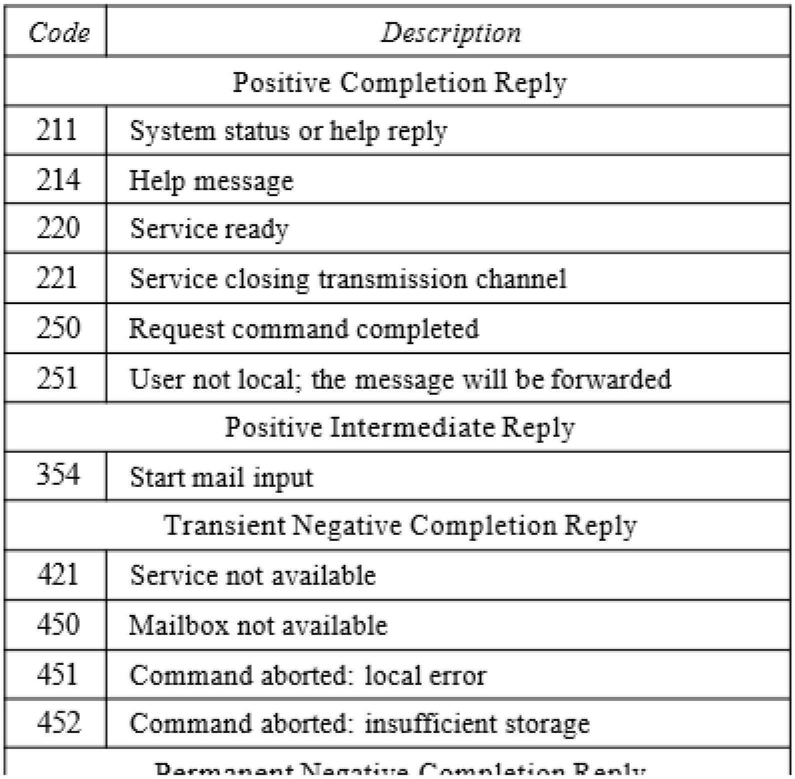


Keyword	*Argument(s)*
HELO	Sender's host name
MAIL FROM	Sender of the message
RCPTTO	Intended recipient of the message
DATA	Body of the mail
QUIT	
RSET	
VRFY	Name of recipient to be verified
NOOP	
TURN	
EXPN	Mailing list to be expanded
HELP	Command name

Responses

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

Table 26.8 *Responses*



Code	*Description*
Positive Completion Reply	
211	System status or help reply
214	Help message
220	Service ready
221	Service closing transmission channel
250	Request command completed
251	User not local; the message will be forwarded
Positive Intermediate Reply	
354	Start mail input
Transient Negative Completion Reply	
421	Service not available
450	Mailbox not available
451	Command aborted: local error
452	Command aborted: insufficient storage
Permanent Negative Completion Reply	

Mail Transfer Phases

The process of transferring a mail message occurs in three phases: connection establishment, mail transfer, and connection termination.

Connection Establishment

After a client has made a TCP connection to well known port 25;the SMTP server starts the connection phase.

Three steps:-

1.The server sends code 220(service ready) to tell the client it is ready to receive mail.If server is not ready ,it sends code 421(service not available).

2.The client sends the HELO message to identify itself, using its domain name address.

3.The server responds with code 250(request command completed)

Message Transfer:-After connection has been established between the SMTP client and server,a single message between a sender and one or more recipients can be exchanged.

Steps are:-

1.The client sends the MAIL FROM message

2.The server responds with code 250

3.The client sends RECPT TO message

4.The server responds with code 250.

5.The client sends the DATA message to initialize the message transfer.

6.The server responds with the code 354(start mail input)

7.The client sends the contents of the message in consecutive lines.

8.The server responds with code 250(OK).

Connection Termination

After the message is transferred successfully, the client terminates the connection.

Steps are:-

1.The client sends the QUIT command.

2.The server responds with code 221.

Message Access Agent: POP and IMAP

The first and the second stages of mail delivery use SMTP. However, SMTP is not involved in the third stage because SMTP is a push protocol; it pushes the message from the client to the server.

In other words, the direction of the bulk: data (messages) is from the client to the server. On the other hand, the third stage needs a pull protoco l; the client must pull messages from the server. The direction of the bulk data is from the server to the client.

The third stage uses a message access agent. **Currently two message access protocols** are available: **Post Office Protocol, version 3 (POP3) and Internet Mail Access Protocol, version 4 (IMAP4).**

POP3

Post Office Protocol, version 3 (POP3) is simple and limited in functionality.

The client POP3 software is installed on the recipient computer.

The server POP3 software is installed on the mail server.

Mail access starts with the client when the user needs to download e-mail from the mailbox on the mail server.

The client opens a connection to the server on TCP port 110.

It then sends its user name and password to access the mailbox.

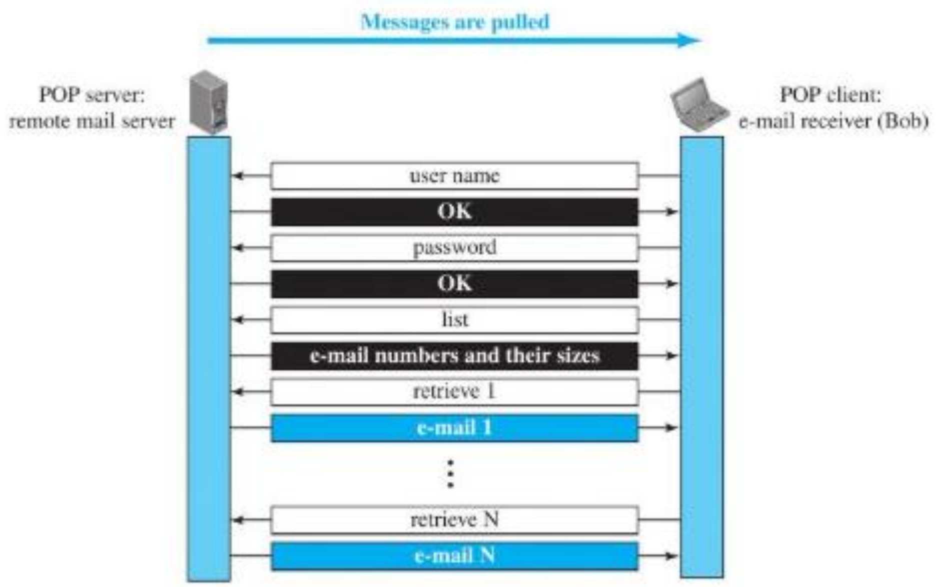
The user can then list and retrieve the mail messages, one by one. Figure 26.17 shows an example of downloading using POP3.

POP3 **has two modes: the delete mode and the keep mode**.

In the delete mode, the mail is deleted from the mailbox after each retrieval.

In the keep mode, the mail remains in the mailbox after retrieval. The delete mode is normally used when the user is working at her permanent computer and can save and organize the received mail after reading or replying.

Figure 26.17 *POP3*



IMAP4 is also a mail accessing agent.

Another mail access protocol is **Internet Mail Access Protocol, version 4** (IMAP4). IMAP4 is similar to POP3, but it has more features; IMAP4 is **more powerful and more complex.**

IMAP4 provides the following extra functions:

- A user can check the e-mail header prior to downloading.
- A user can search the contents of the e-mail for a specific string of characters prior to downloading.
- A user can partially download e-mail. This is especially useful if bandwidth is limited and the e-mail contains multimedia with high bandwidth requirements.
- A user can create, delete, or rename mailboxes on the mail server.
- A user can create a hierarchy of mail boxes in a folder for e-mail storage.

File Transfer Protocol (FTP)

File Transfer Protocol (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 and straightforward.

- FTP uses two well-known TCP ports: Port 21 is used for the control connection, and port 20 is used for the data connection.
- FTP uses the services of TCP.
- The client **has three components**: **user interface, client control process, and the client data transfer process.**
- The server **has two components**: the **server control process** and **the server data transfer process**.
- The control connection is made between the control processes.

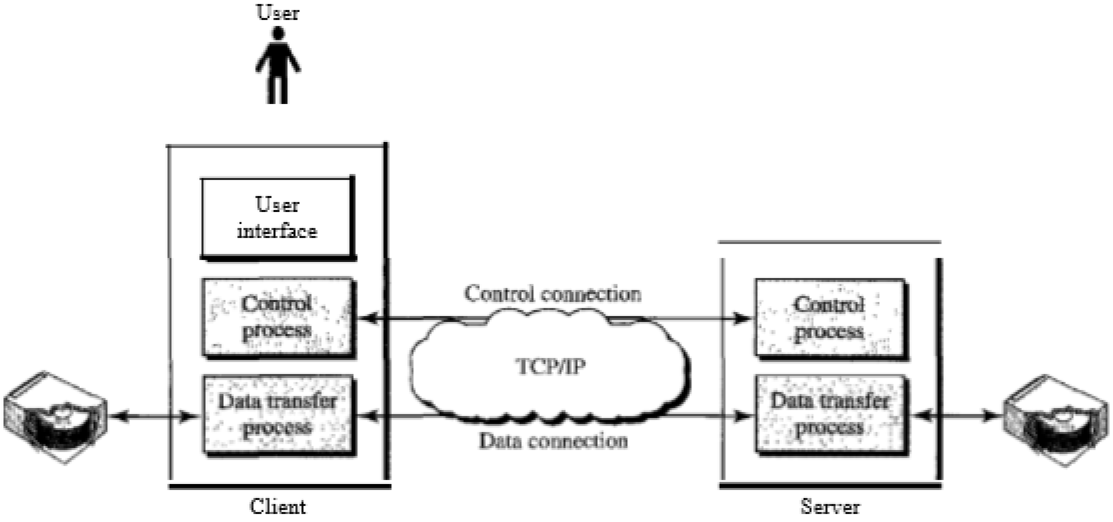
Anonymous FTP

To use FTP, a user needs an account (user name) and a password on the remote server. To access these files, a user does not need to have an account or password. Instead, the user can use anonymous as the user name and guest as the password.

TWO Connections :Data connection and Control Connection

The **data connection** is made between the data transfer processes.

Figure 26.21 *FTP*



Communication over Control Connection

FTP uses the same approach as SMTP to communicate across the control connection.

It uses the 7-bit ASCII character set .

Communication is **achieved through commands and responses**. This simple method is adequate for the control connection because we send one command (or response) at a time.

Each command or response is only one short line.

Communication over Data Connection

The purpose of the data connection is different from that of the control connection. We want to transfer files through the data connection. File transfer occurs over the data connection under the control of the commands sent over the control connection.

However, we should remember that file transfer in FTP means one of three things:

- A file is to be copied from the server to the client. This is called retrieving aft/e. It is done under the supervision of the RETR command,
- A file is to be copied from the client to the server. This is called storing aft/e. It is done under the supervision of the STOR command.
- A list of directory or file names is to be sent from the server to the client

The client must define :the **type of file to be transferred, the structure of the data, and the transmission mode.**

Before sending the file through the data connection, we prepare for transmission through the control connection.

three attributes of communication: file type, data structure, and transmission mode.

File Type

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

Data Structure

FTP can transfer a file across the data connection using one of the following **the structure of the data:-**

- file structure
- Record structure or Page structure

i)The file structure format has no structure.It is a continuous stream of bytes.**ii)In the record structure,**the file is divided into records.This can be used only with text files. **iii)In the page structure**, the file is divided into pages,with each page having a page number and a page header.The pages can be stored in sequentially.

Transmission Mode: FTP can transfer a file across the data connection by using one of the following three transmission modes: **stream mode, block mode, and compressed mode.**

- The stream mode is the default mode.
- In block mode, data can be delivered from FTP to TCP in blocks. In this case, each block is preceded by a 3-byte header. The first byte is called the block descriptor; the next 2 bytes define the size of the block in bytes.
- The compression method normally used is run-length encoding. In a binary file, null characters are usually compressed.

File Transfer

File transfer occurs over the data connection under the control of the commands sent over the control connection.

File transfer in FTP means one of three things:retrieving a file (server to client),storing a file(client to server),and directory listing(server to client)

Security for FTP

Data transfer in plaintext is insecure.To be secure,one can add a **Secure Socket Layer** between the FTP application Layer and the TCP layer.In this case FTP is called **SSL-FTP.**

WEB DOCUMENTS

The documents in the WWW can be grouped into three broad categories: static, dynamic, and active.

Static Documents

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

Dynamic Documents

A dynamic document is created by a Web server whenever a browser requests the document. When a request arrives, the Web server runs an application program or a script that creates the dynamic document. The server returns the output of the program or script as a response to the browser that requested the document.

The client can ask the server to run a program such as the date program in UNIX and send the result of the program to the client.

Dynamic documents are sometimes referred to as server-site dynamic documents.

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 the user.

Java Applets:-One way to create an active document is to use Java applets.

HTML

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

- HTML stands for Hyper Text Markup Language
- HTML describes the structure of a Web page
- HTML consists of a series of elements
- HTML elements tell the browser how to display the content
- HTML elements are represented by tags
- HTML tags label pieces of content such as "heading", "paragraph", "table", and so on

HTML was created by Berners-Lee in late 1991 but "HTML 2.0" was the first standard HTML specification which was published in 1995.

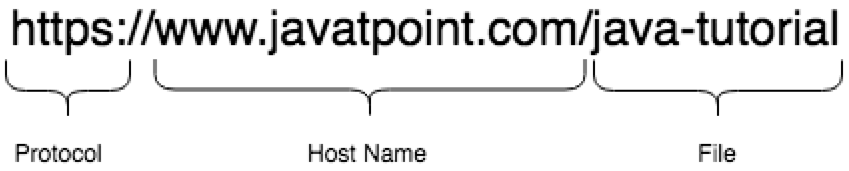
Uniform Resource Locator

A client that wants to access a Web page needs the address.

The uniform resource locator (URL) is a standard for specifying any kind of information on the Internet. The URL defines four things: protocol, host computer, port, and path .

URL is an acronym for Uniform Resource Locator. It points to a resource on the World Wide Web. For example:

1. https://www.javatpoint.com/java-tutorial



A URL contains many information:

1. **Protocol:** In this case, http is the protocol.
2. **Server name or IP Address:** In this case, www.javatpoint.com is the server name.
3. **Port Number:** It is an optional attribute. If we write http//ww.javatpoint.com:80/sonoojaiswal/ , 80 is the port number. If port number is not mentioned in the URL, it returns -1.
4. **File Name or directory name:** In this case, index.jsp is the file name.

HTTP

The Hypertext Transfer Protocol (HTTP) is a protocol used mainly to access data on the World Wide Web. HTTP functions as a combination of FTP and SMTP.

HTTP uses the services of TCP on well-known port80.

HTTP Transaction

Figure 27.12 illustrates the HTTP transaction between the client and server.

Although HTTP uses the services of TCP, HTTP **itself is a stateless protocol**. The client initializes the transaction by sending a request message. The server replies by sending a response.

Messages

The formats of the **request and response messages**. A request message consists of a request line, a header, and sometimes a body. A response message consists of a status line, a header, and sometimes a body.

Figure 27.12 *HTTP transaction*

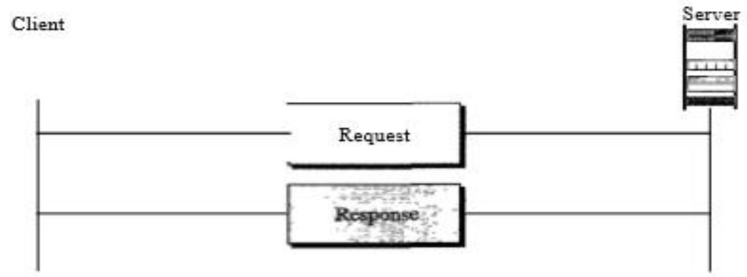
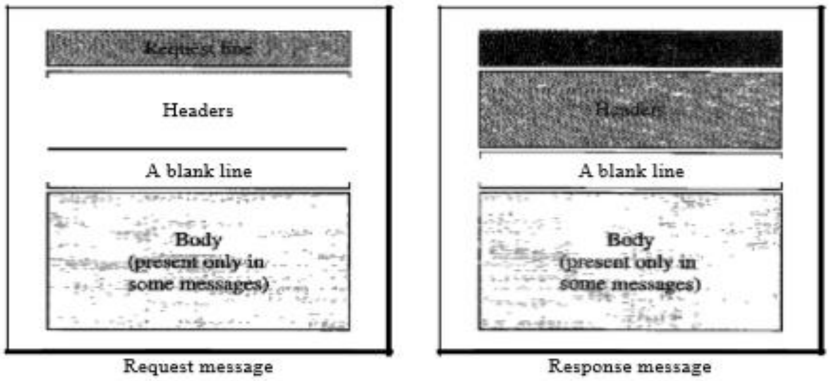


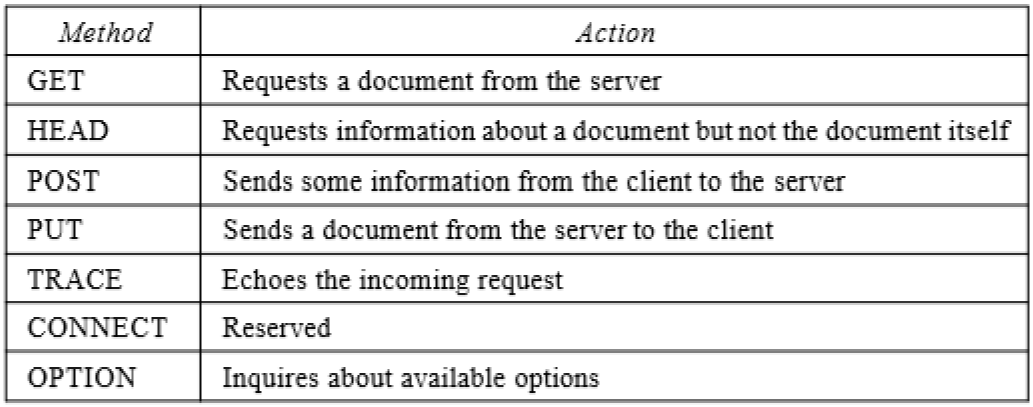
Figure 27.13 *Request and response messages*



Request and Status Lines The first line in a request message is called **a request line; the first line in the response message** is called the status line.

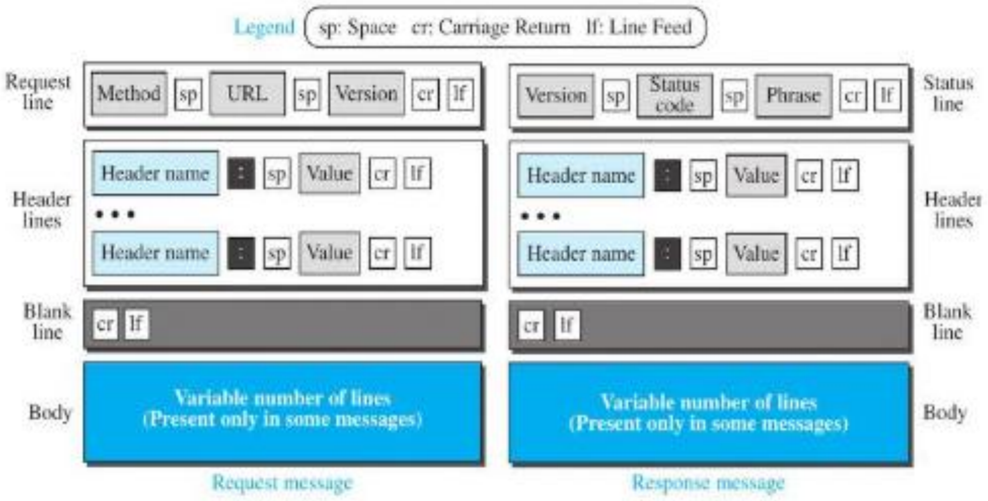
Request type. This field is used in the request message. In version 1.1 of HTTP, several request types are defined. The request type is categorized into methods as defined in Table:-

Table 27.1 *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

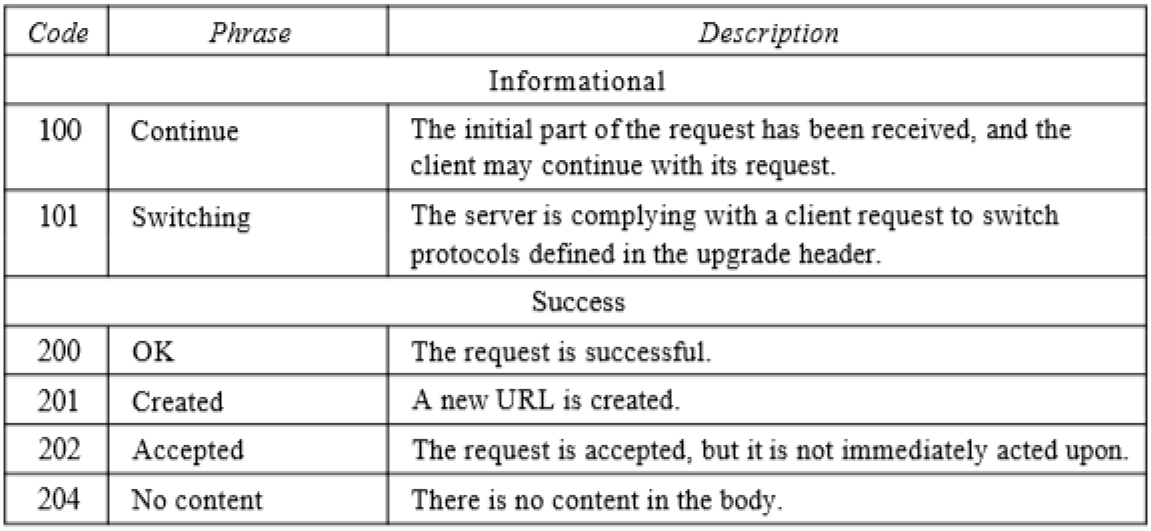
Figure 26.5 *Formats of the request and response messages*



URL:-

- Version. The most current version of HTTP is 1.1.
- Status code. This field is used in the response message.
- Status phrase. This field is used in the response message. It explains the status code in text form.

Table 27.2 *Status codes*



Code	*Phrase*	*Description*
		Informational
100	Continue	The initial part of the request has been received, and 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.

- Header :-The header exchanges additional information between the client and the server.
- Body :-The body can be present in a request or response message. Usually, it contains the document to be sent or received.

<u>Persistent Versus Nonpersistent Connection</u>

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

- **Nonpersistent Connection**

In a nonpersistentconnection, one TCP connection is made for each request/response. The following lists the steps in this strategy:

1. The client opens a TCP connection and 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; itthen closes the connection.

- **Persistent Connection**

HTTP version 1.1 specifies a persistent connection by default.

Proxy Server

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 sends the request to the corresponding server. Incoming responses are sent to the proxy server and stored for future requests from other clients.

The proxy server reduces the **load on the original server, decreases traffic, and improves latency**.

Figure 26.3 *Example 26.3*

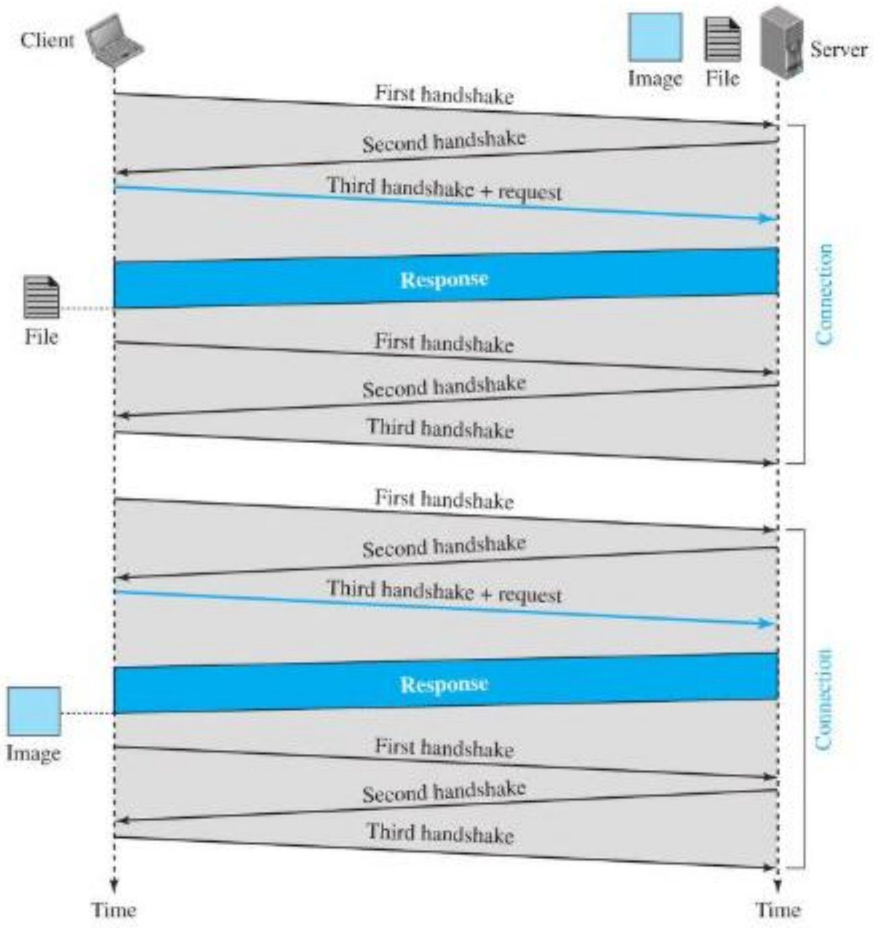


Figure:-Nonpersistent connection

Figure 26.4 *Example 26.4*

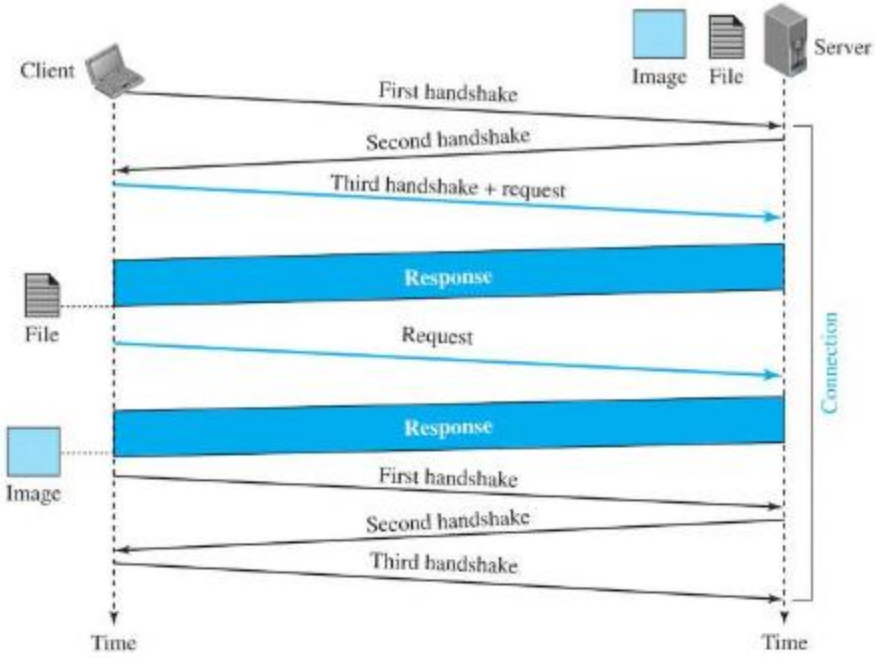


Figure:-persistent connection

WWW stands for World Wide Web.

A technical definition of the World Wide Web is : all the resources and users on the Internet that are using the Hypertext Transfer Protocol (HTTP).The idea of the web was first proposed by **Tim Berners-Lee** .

Identifiers and Character Set

Uniform Resource Identifier (URI) is used to uniquely identify resources on the web and **UNICODE** makes it possible to built web pages that can be read and write in human languages.

Syntax

XML (Extensible Markup Language) helps to define common syntax in semantic web.

Data Interchange

Resource Description Framework (RDF) framework helps in defining core representation of data for web. RDF represents data about resource in graph form.

Taxonomies

RDF Schema (RDFS) allows more standardized description of **taxonomies** and other **ontological** constructs.

Ontologies

Web Ontology Language (OWL) offers more constructs over RDFS. It comes in following three versions:

- OWL Lite for taxonomies and simple constraints.
- OWL DL for full description logic support.
- OWL for more syntactic freedom of RDF

Rules

RIF and **SWRL** offers rules beyond the constructs that are available from **RDFs** and **OWL.** Simple Protocol and **RDF Query Language (SPARQL)** is SQL like language used for querying RDF data and OWL Ontologies.

Proof

All semantic and rules that are executed at layers below Proof and their result will be used to prove deductions.

Cryptography

Cryptography means such as digital signature for verification of the origin of sources is used.

User Interface and Applications

On the top of layer **User interface and Applications** layer is built for user interaction.

The below figure :-**Architecture of web**

Web server:-It is a specialized server that responds to client request.

Web client:- The **web client** can be said as an application or **web** browser (like Google Chrome, Internet Explorer, Opera, Firefox, Safari) which is installed in a computer and used to interact with **Web servers** upon user's request.

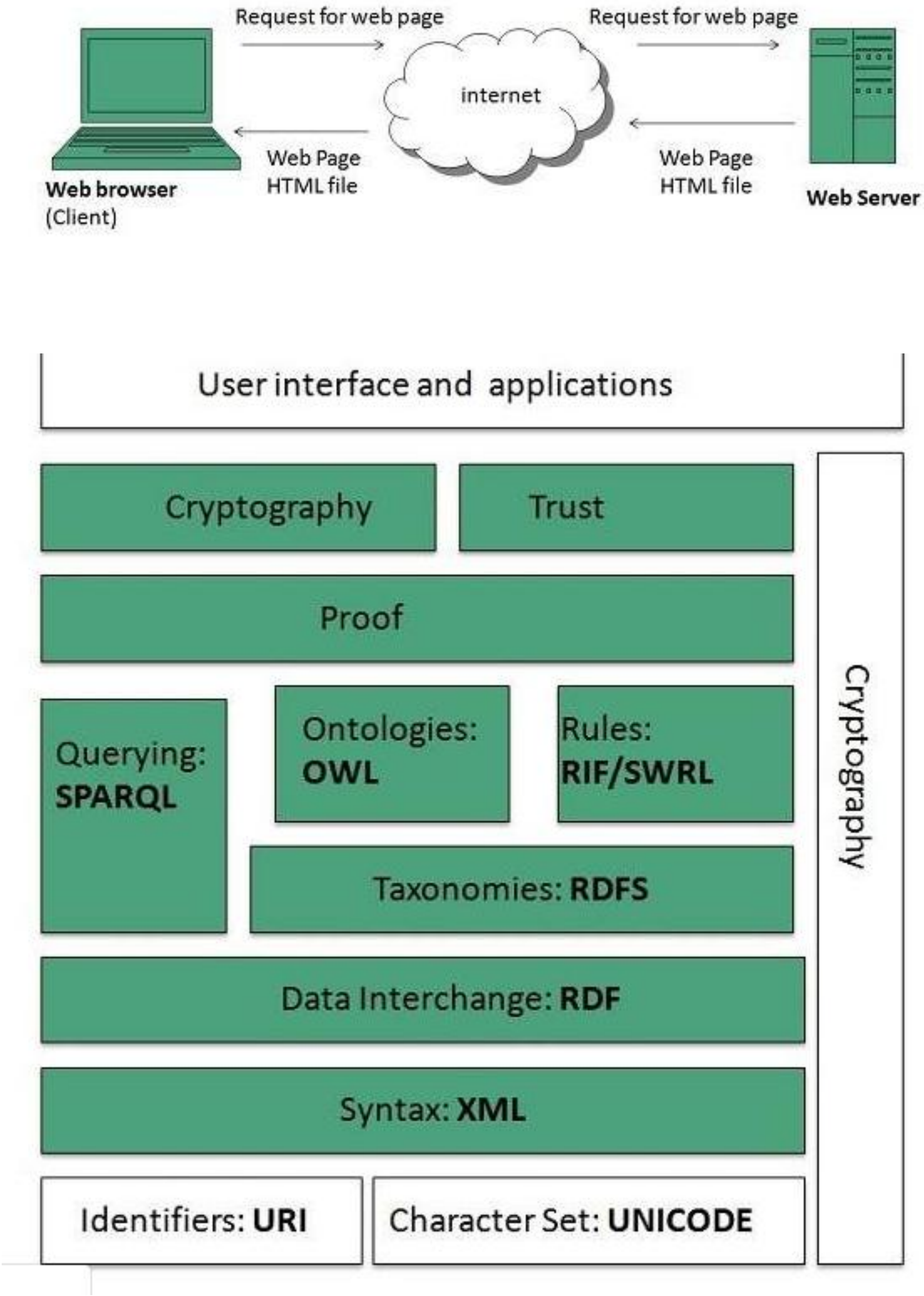


Figure:-Architecture of WWW

A **web page** or **webpage** is a document commonly written in HTML (Hypertext Markup Language) that is accessible through the Internet or other networks using an Internet browser.

A web page is accessed by entering a URL address and may contain text, graphics, and hyperlinks to other web pages and files.