UNIT V

Application Layer:

An application layer is an abstraction layer that specifies the shared communications protocols and interface methods used by hosts in a communications network. The application layer abstraction is used in both of the standard models of computer networking: the Internet Protocol Suite (TCP/IP) and the OSI model.

The application layer in the OSI model is the closest layer to the end user which means that the application layer and end user can interact directly with the software application. The application layer programs are based on clients and servers.

The Application layer includes the following functions:

- **Identifying communication partners:** The application layer identifies the availability of communication partners for an application with data to transmit.
- **Determining resource availability:** The application layer determines whether sufficient network resources are available for the requested communication.
- **Synchronizing communication:** All the communications occur between the applications requires cooperation which is managed by an application layer.

Services of Application Layers

- **Network Virtual terminal:** An application layer allows a user to log on to a remote host. To do so, the application creates a software emulation of a terminal at the remote host. The user's computer talks to the software terminal, which in turn, talks to the host. The remote host thinks that it is communicating with one of its own terminals, so it allows the user to log on.

- **File Transfer, Access, and Management (FTAM):** An application allows a user to access files in a remote computer, to retrieve files from a computer and to manage files in a remote computer. FTAM defines a hierarchical virtual file in terms of file structure, file attributes and the kind of operations performed on the files and their attributes.
- **Addressing:** To obtain communication between client and server, there is a need for addressing. When a client makes a request to the server, the request contains the server address and its own address. The server response to the client request, the request contains the destination address, i.e., client address. To achieve this kind of addressing, DNS is used.
- **Mail Services:** An application layer provides Email forwarding and storage.
- **Directory Services:** An application contains a distributed database that provides access for global information about various objects and services.

Authentication: It authenticates the sender or receiver's message or both.

Domain Name System :

There are several applications in the application layer of the Internet model that follow the client/server paradigm. The client/server programs can be divided into two categories: those that can be directly used by the user, such as e-mail, and those that support other application programs. The Domain Name System (DNS) is a supporting program that is used by other programs such as e-mail.

1. Need for DNS

To identify an entity, TCP/IP Protocols use the IP address, which uniquely identifies the connection of a host to an internet. In the case of ARPANET, a file named hosts.txt is used to list all hosts and their IP addresses, this work suitable for small network but not for large network due to heavy load and latency. Therefore, people prefer to use names instead of addresses that is, we need a system that can map a name to an address and conversely an address to a name. Thus, the preferred system is called the Domain Name system.

2.Domain Name space:

DNS can be pictured as an inverted hierarchical tree structure with one root node at the top and a maximum of 128 levels.

Labels:

Each node in the tree has a label, which is strong with a maximum of 63 characters.

Domain Name:

Each node in the tree has a domain name. A full domain name is a sequence of labels separated by dots (.).

Fully Qualified Domain Name (FQDN):

A FQDN is a domain name consisting of labels beginning with the host and going back through each level to the root node. Eg. Challenger.atc.fh.da.Edu

Partially Qualified Domain Name (PQDN):

In PQDN is a domain name that does not include all the levels between the host and the root node. Eg. Challenger.

3.Name Server:

In theory at least, a single name server could contain the entire DNS database and respond to all queries about it. In practice, this server would be so overloaded as to be useless. To avoid problems associated with having only a single source of information, the DNS name space is divided into non-overlapping zones. One possible way to divide the name space, where the zone boundaries are placed within a zone is up to that zones administrator. This decision is made in larger part based on how many name servers are desired. To improve reliability, some servers for a zone can be located outside the zone.

The DNS client, called a resolver, maps a name to an address, or an address to a name. When a resolver has a query about the domain name, it passes the query to one of the local name servers. If the domain being sought falls under the jurisdiction of the name server, such as ai.cs.yale.edu falling under cs.yale.edu, it returns the authoritative resource records. An authoritative record is one that comes from the authority that manages the record and it is thus always correct. While, DNS helps in mapping names onto their IP addresses. It does not help locate people, resources, services or objects in general. For locating these things, another directory service has been defined, called LDAP (Light Weight Directory Access protocol).

2. DNS in the Internet:

DNS is a protocol that can be used in different platforms. In the Internet, the domain name space is divided into three sections are

1. Generic domains
2. Country domains and
3. Inverse domain

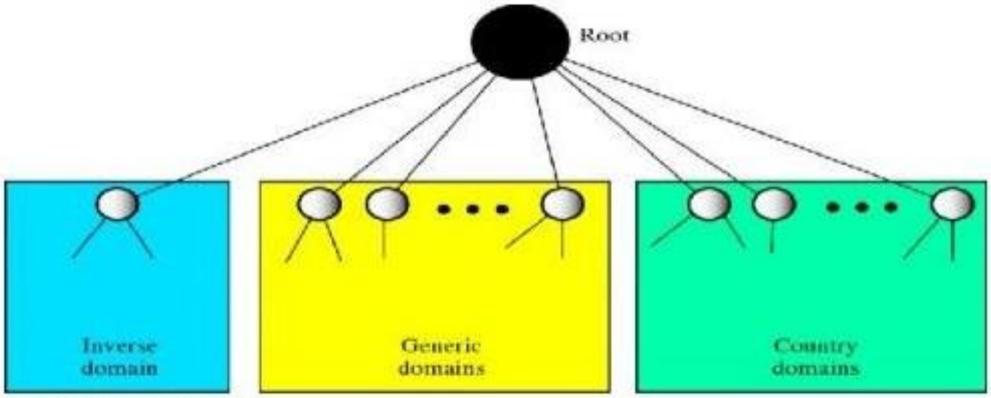


Figure DNS in the Internet

1. Generic domain:

There are 14 generic domains, each specifying an organization type. The generic domain defines registered hosts according to their generic behavior. Each node in the tree defines a domain, which is an index to the domain name space database

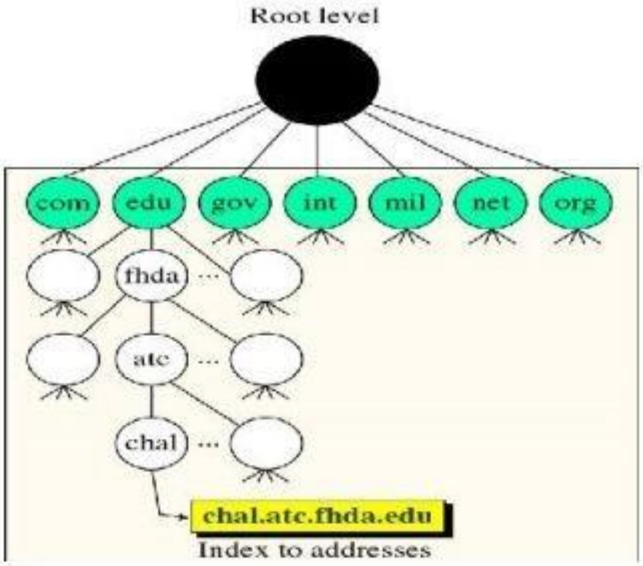
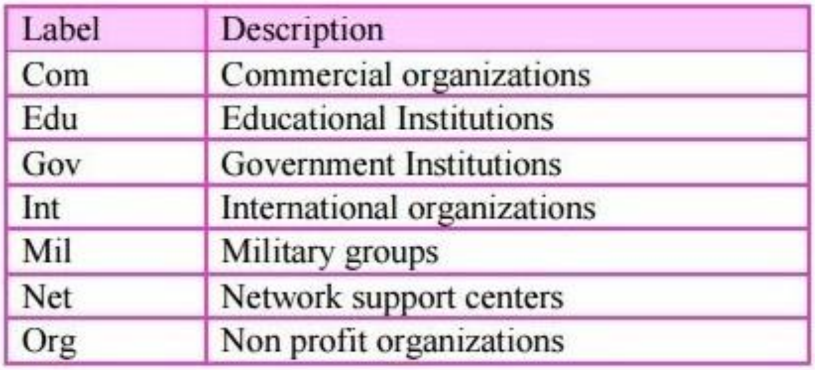


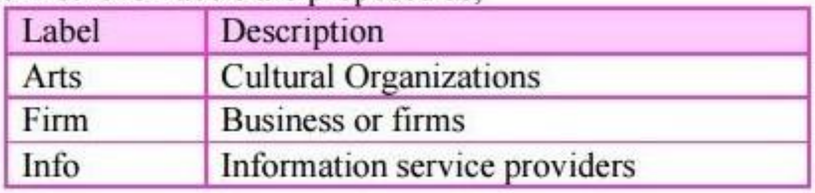
Figure Generic domain

Looking at the tree, we see that the first level in the generic domain section allows seven possible three-character labels. These labels describe the organization types as shown below

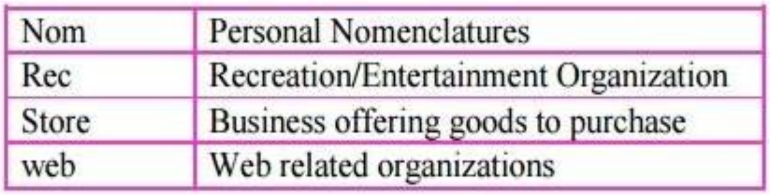


Label	Description
Com	Commercial organizations
Edu	Educational Institutions
Gov	Government Institutions
Int	International organizations
Mil	Military groups
Net	Network support centers
Org	Non profit organizations

Recently a few more first level labels are proposed as,



Label	Description
Arts	Cultural Organizations
Firm	Business or firms
Info	Information service providers



Nom	Personal Nomenclatures
Rec	Recreation/Entertainment Organization
Store	Business offering goods to purchase
web	Web related organizations

2. Country domains

Each country domain specifies a country. This section follows the same format as the generic domains but uses two-character country abbreviations in place of three character organizational abbreviations at the first level. Second level labels can be organizational, or they can be more specific, national designations. The following figure 5.3 shows the country domain section the address cs.Keio.ac.jp refers to the computer science department of Keio University in Japan. To create a new domain, permission is required of the domain in which it will be included.

For example, if a new university is chartered, say the University of Chennai, it must ask the manager of the edu domain to assign it unc.edu, in order to avoid conflicts and each domain can keep track of all its sub-domains

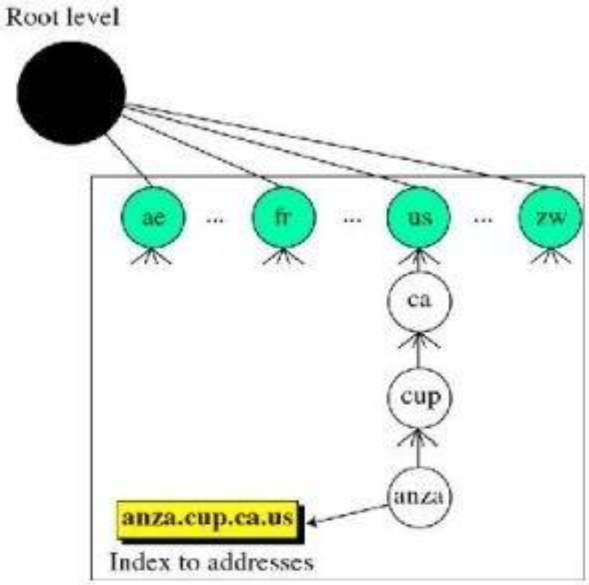


Figure Country domains

Once a new domain has been created and registered, it can create sub-domains, such as cs.unc.edu, without getting permission from anybody higher up the tree.

3. Inverse domain:

The inverse domain finds a domain name for a given IP address. This is called address-to-name resolution. It is used to map an address to a name. This may happen, for example, when a server lists only the IP address of the client. To determine if the client is on the authorized list, it can be send a query to the DNS server and ask for a mapping of address to name in figure.

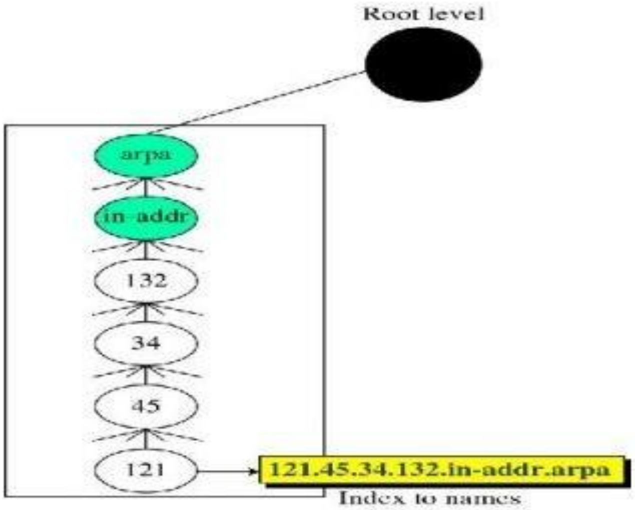


Figure Inverse domains

3. Types of Records:

There are two types of DNS records:

1. Question records
2. Resource records

Question Records:

The question records are used in the question section of the query and response messages. It is used by the client to get information from a server.

Resource Records:

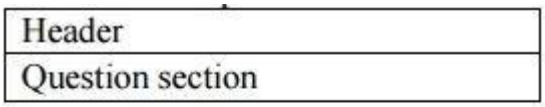
Every domain whether it is a single host or a top level domain, can have a set of resource records associated with it. For a single host, the most common resource record is just its IP address, but many other kinds also exist. When a resolver gives a domain name to DNS, what it gets back are the resource records associated with that name. Thus, the primary function of DNS is to map domain names onto resource records. The server database consists of resource records. This record is used in the answer, authoritative and additional information sections of the response message.

4. DNS Messages:

These are two types of DNS Messages queries and responses. Both types have the same format.

Queries Messages:

The query message consists of a header and question

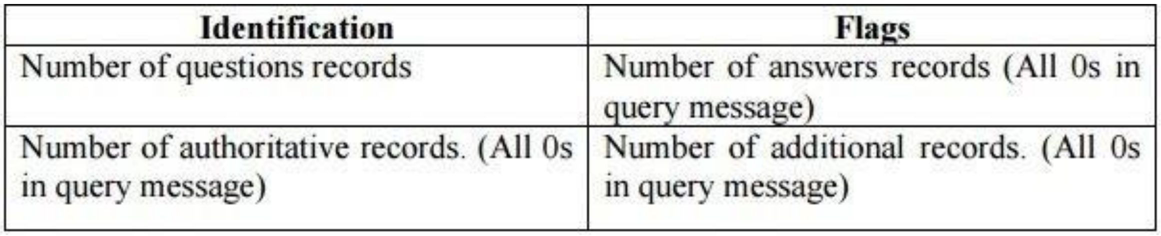


Header
Question section

Figure Query Message

Response Messages:

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



Identification	Flags
Number of questions records	Number of answers records (All 0s in query message)
Number of authoritative records. (All 0s in query message)	Number of additional records. (All 0s in query message)

5. Header Format:

Both have the same header format. The header is 12 bytes.

Identification

1. Number of questions records
2. Number of authoritative records. (All 0s in query message)

Flags

1. Number of answers records (All 0s in query message)
2. Number of additional records. (All 0s in query message)

The **identification subfield** is used by the client to match the response with the query.

The **flag subfield** is a collection of subfields that define the types of the message, the type of answer requested, and the type of desired resolution and so on.

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. Its value is zero in the query message.

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

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

DNS in Internet :

There are 3 components:

Name Space:

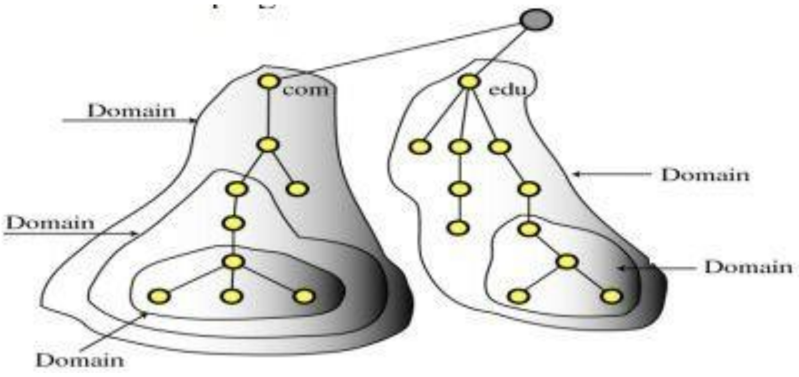
Specifications for a structured name space and data associated with the names

Resolvers:

Client programs that extract information from Name Servers.

Name Servers:

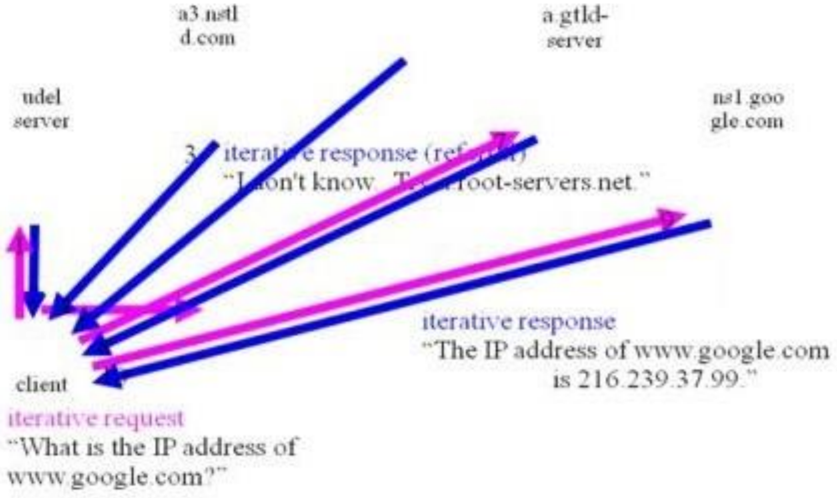
Server programs which hold information about the structure and the names.



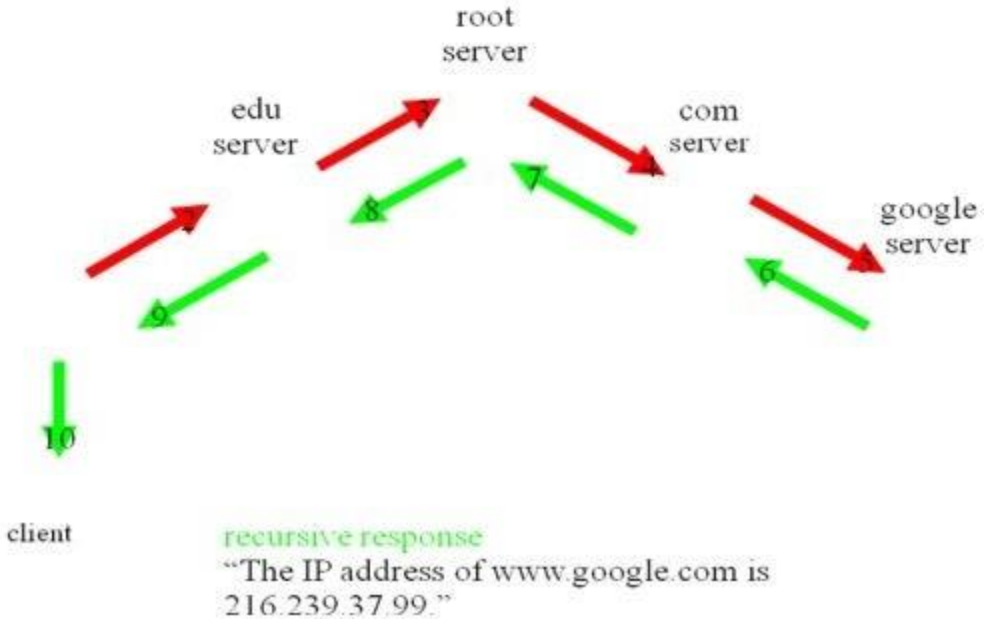
Resolvers

A Resolver maps a name to an address and vice versa.

Iterative Resolution


a3.nstl
d.com
a.gtld-
server
udel
server
ns1.goo
gle.com
3
iterative response (referral)
“I don't know. Try a.root-servers.net.”
7
9
iterative response
“The IP address of www.google.com
is 216.239.37.99.”
client
iterative request
“What is the IP address of
www.google.com?”

Recursive Resolution



Electronic Mail :

What is an Email – an electronic message transmitted over a network from one user to another..

Email can be as simple as a few lines of text, or include attachments such as pictures or documents.

Email made up 75% of network traffic soon after the introduction of the internet.

- The Header

Who sent the email.

To whom the mail is sent.

When the email was sent.

The email subject.

The size of the email.

The Body

Contains the message.

May also contain an attachment.

Attachments Different Architectural Models exist for constructing computer systems.

- Some models include:
  - Peer-Peer
  - Pipe and Filter
  - Implicit Invocation
  - Client-Server

If not embedded within the body, attachments are sent along with the email. How Email Works

Architecture

To explain the architecture of e-mail, we give four scenarios. We begin with the simplest situation and add complexity as we proceed. The fourth scenario is the most common in the exchange of email.

First Scenario

In the first scenario, the sender and the receiver of the email are users (or application programs) on the same system; they are directly connected to a shared system. The administrator has created one mailbox for each user where the received messages are stored. A mailbox is part of a local hard drive, a special file with permission restrictions. Only the owner of the mailbox has access to it. When Alice, a user, needs to send a message to Bob, another user, Alice runs a user agent (VA) program to prepare the message and store it in Bob's mailbox. The message has the sender and recipient mailbox addresses (names of files). Bob can retrieve and read the contents of his mailbox at his convenience, using a user agent.

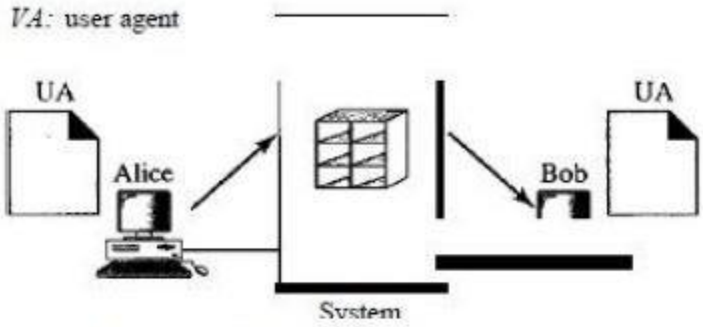


Figure First Scenario in e-mail

Second Scenario

In the second scenario, the sender and the receiver of the email are users (or application programs) on two different systems. The message needs to be sent over the Internet. Here we need user agents (VAs) and message transfer agents (MTAs).

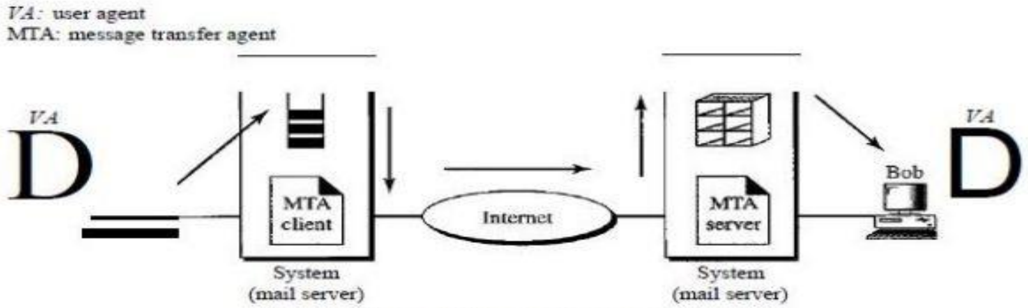


Figure Second Scenario in e-mail.

Third Scenario

In the third scenario, Bob, as in the second scenario, is directly connected to his system. Alice, however, is separated from her system. Either Alice is connected to the system via a point-to-point WAN, such as a dial-up modem, a DSL, or a cable modem; or she is connected toa LAN in an organization that uses one mail server for handling e-mails-all users need to send their messages to this mail server.

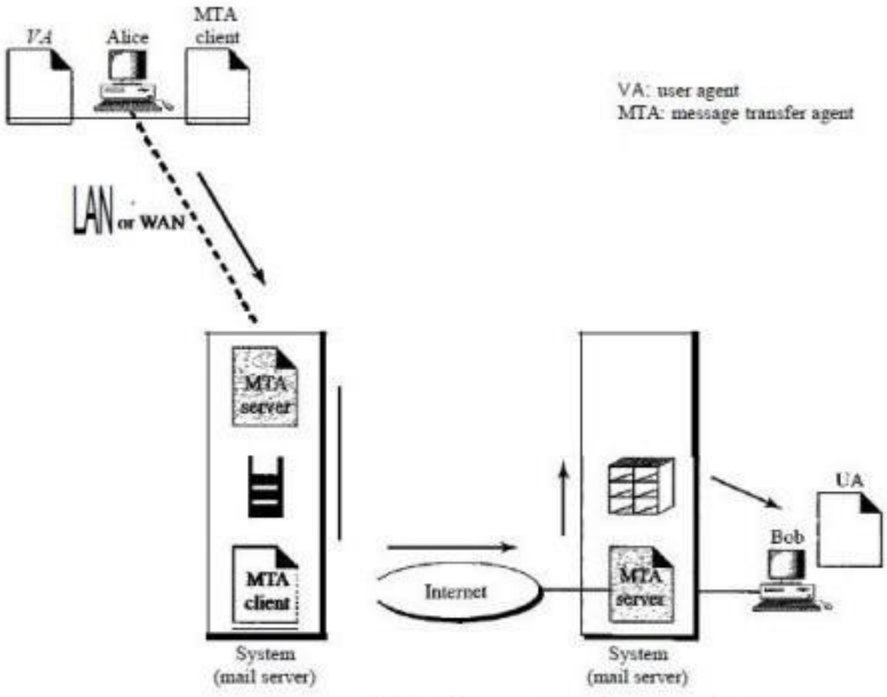


Figure Third Scenario in e-mail

Fourth Scenario

In the fourth and 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. Here, we need another set of client/server agents, which we call message access agents (MAAs). Bob uses an MAA client to retrieve his messages. The client sends a request to the MAA server, which is running all the time, and requests the transfer of the messages.

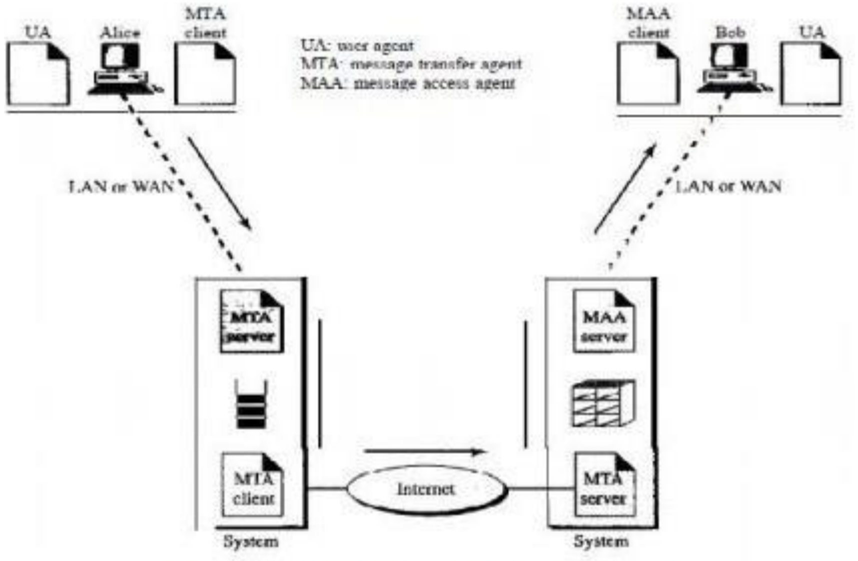


Figure Fourth Scenario in e-mail

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 User Agent

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

Composing Messages A user agent helps the user compose the e-mail message to be sent out. Most user agents provide a template on the screen to be filled in by the user. Some even have a built-in editor that can do spell checking, grammar checking, and other tasks expected from a sophisticated word processor. A user, of course, could alternatively use his or her favourite text editor or word processor to create the message and import it, or cut and paste it, into the user agent template.

Reading Messages The second duty of the user agent is to read the incoming messages. When a user invokes a user agent, it first checks the mail in the incoming mailbox. Most user agents show a one-line summary of each received mail. Each email contains the following fields.

1. A number field.

2. A flag field that shows the status of the mail such as new, already read but not replied to, or read and replied to.

3. The size of the message.

4. The sender.

5. The optional subject field.

Replying to Messages After reading a message, a user can use the user agent to reply to a message. A user agent usually allows the user to reply to the original sender or to reply to all recipients of the message. The reply message may contain the original message (for quick reference) and the new message.

Forwarding Messages Replying is defined as sending a message to the sender a message to the sender or recipients of the copy. Forwarding is defined as sending the message to a third party. A user agent allows the receiver to forward the message, with or without extra comments, to a third party.

3. 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. They are still present as the underlying user agents in servers. A command-driven user agent normally accepts a one-character command from the keyboard to perform its task. 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 and all recipients. Some examples of command-driven user agents are mail, pine, and elm.

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. They have 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, Microsoft's Outlook, and Netscape.

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

4. 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 on the Internet is called the Simple Mail Transfer Protocol (SMTP). As we said before, two pairs of MTA client/server programs are used in the most common situation (fourth scenario).

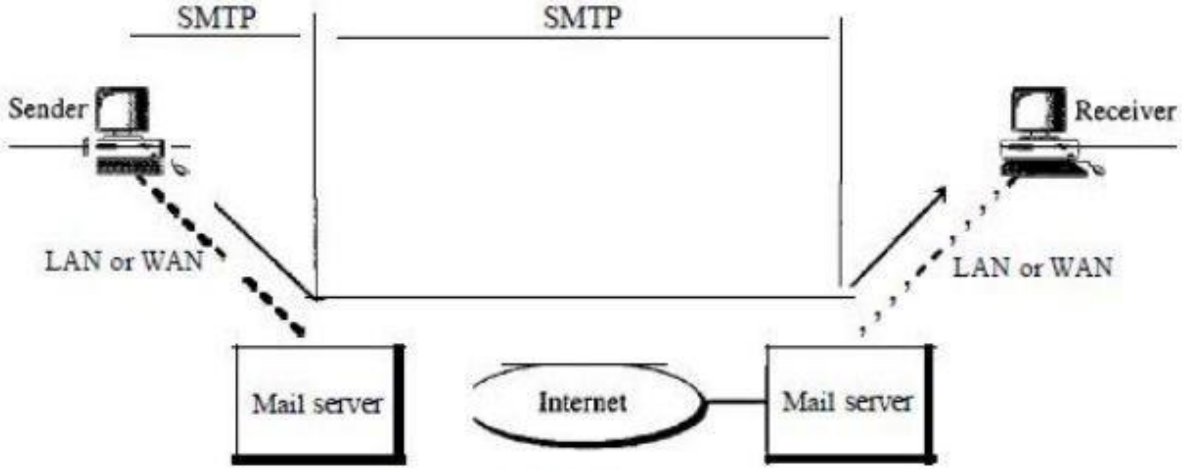


Figure : SMTP Range

SMTP is used two times, between the sender and the sender's mail server and between the two mail servers. As we will see shortly, another protocol is needed between the mail server and the receiver.

SMTP simply defines how commands and responses must be sent back and forth. 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 and an MTA server. Each command or reply is terminated by a two-character (carriage return and line feed) end-of-line token.

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; every implementation must support these five commands. The next three are often used and highly recommended. The last six are seldom used.

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.

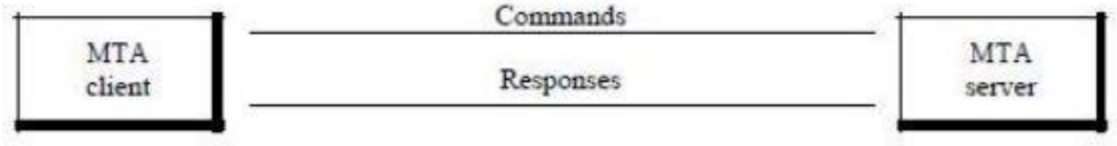


Figure Commands and responses

5. Mail Transfer Phases

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

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. 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. The keep mode is normally used when the user accesses her mail away from her primary computer (e.g., a laptop). The mail is read but kept in the system for later retrieval and organizing..

IMAP4

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

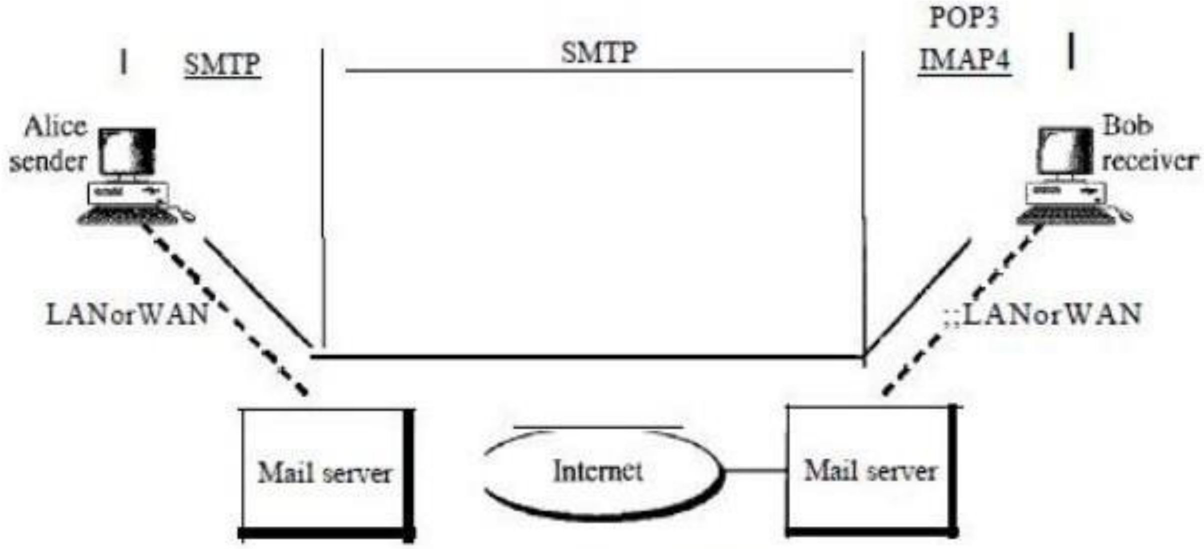
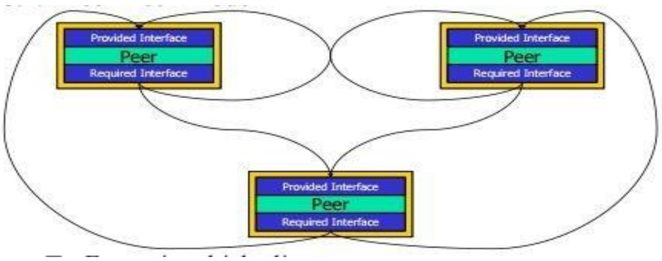


Figure POP3 and IMAP4

Peer-Peer Model



Forms in which clients appear:

- Application based - these are installed onto user's machines and include Microsoft Outlook and the freely available Outlook Express and Eudora.
- Web based - these appear in a web browser's window and include Hotmail, Yahoo and Outlook web client.

Clients vary greatly in functionality, but all provide a basic level of functionality that assists the user.

Basic functions include:

- Ability to create new emails.
- Display and store received emails.
- Hold address lists of contacts, a calendar, journal and other extra functions that help organize the user's working day.
- The client is also configured with the account information and names or IP addresses of the email servers with which it will be communicating.

An email server is typically a combination of processes running on a server with a large storage capacity – a list of users and rules, and the capability to receive, send and store emails and attachments.

These servers are designed to operate without constant user intervention.

Should process emails for months as sending, receiving and maintenance tasks are carried out at scheduled times. The client only has to connect to the email server when it sends and checks/receives new email.

Sometimes it may be permanently connected to the server to allow access to shared address books or calendar information – this is typical of a LAN-based email server.

Most email servers conduct email services by running two separate processes on the same machine.

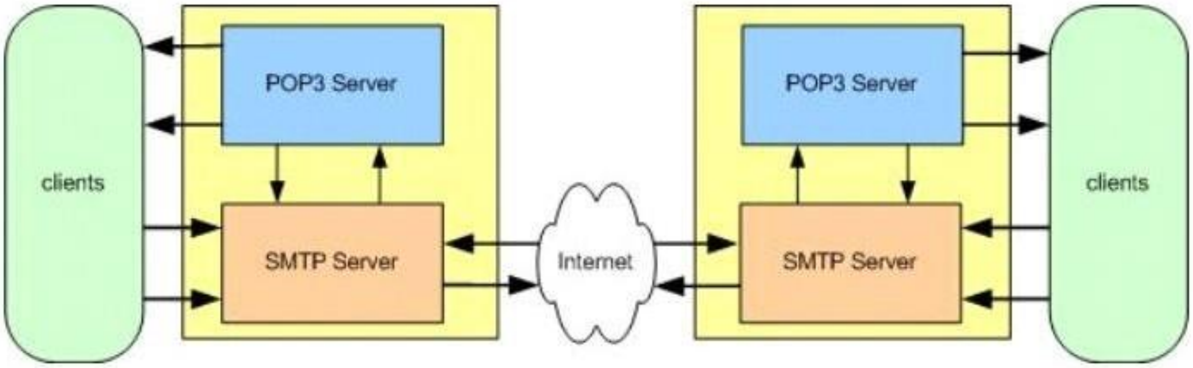
One process is the POP3 (Post Office protocol 3) server, which holds emails in a queue and delivers emails to the client when they are requested.

The other is the SMTP (simple mail transfer protocol) server that receives outgoing emails from clients and sends and receives email from other SMTP servers.

These two processes are linked by an internal mail delivery mechanism that moves mail between the POP3 and SMTP servers.

When the client calls the email server to send or check for mail it connects to the server on certain TCP/IP ports:

- SMTP on port 25
- POP3 on port 110.



File Transfer Protocol :

Transferring files from one computer to another is one of the most common tasks expected from a networking or internetworking 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) 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, some problems must be dealt with first.

FTP differs from other client/server applications in that it establishes two connections between the hosts. One connection is used for data transfer, the other for control information (commands and responses). Separation of commands and 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 types transferred. However, the difference in complexity is at the FTP level, not TCP. For TCP, both connections are treated the same. FTP uses two well-known TCP ports: Port 21 is used for the control connection, and port 20 is used for thedata connection.

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

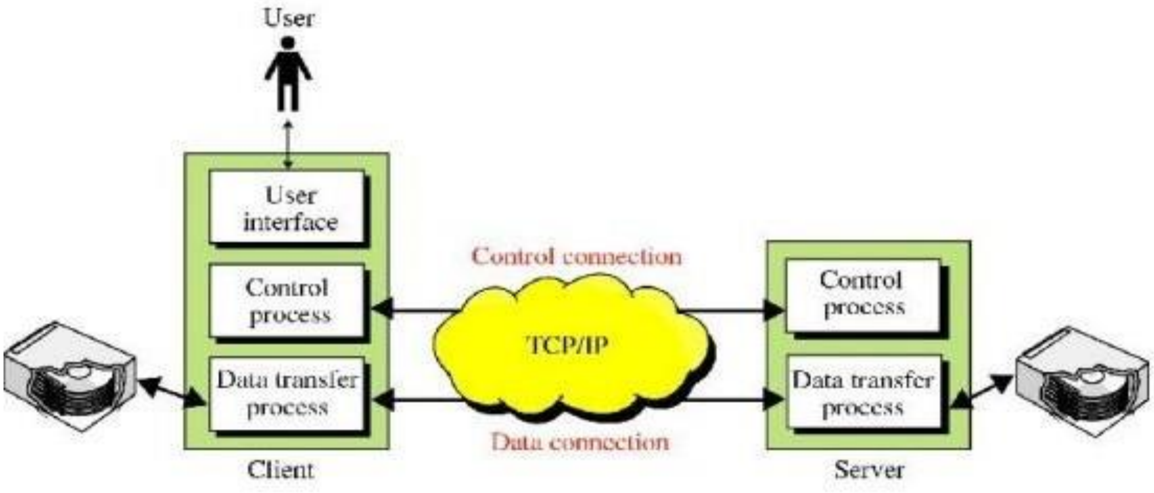


Figure FTP

The control connection remains connected during the entire interactive FTP session. The data connection is opened and then closed for each file transferred. It opens each time commands that involve transferring files are used, and it closes when the file is transferred. In other words, when a user starts an FTP session, the control connection opens. While the control connection is open, the data connection can be opened and closed multiple times if several files are transferred.

1. 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, so we need not worry about file format or file structure. Each line is terminated with a two-character (carriage return and line feed) end-of-line token.

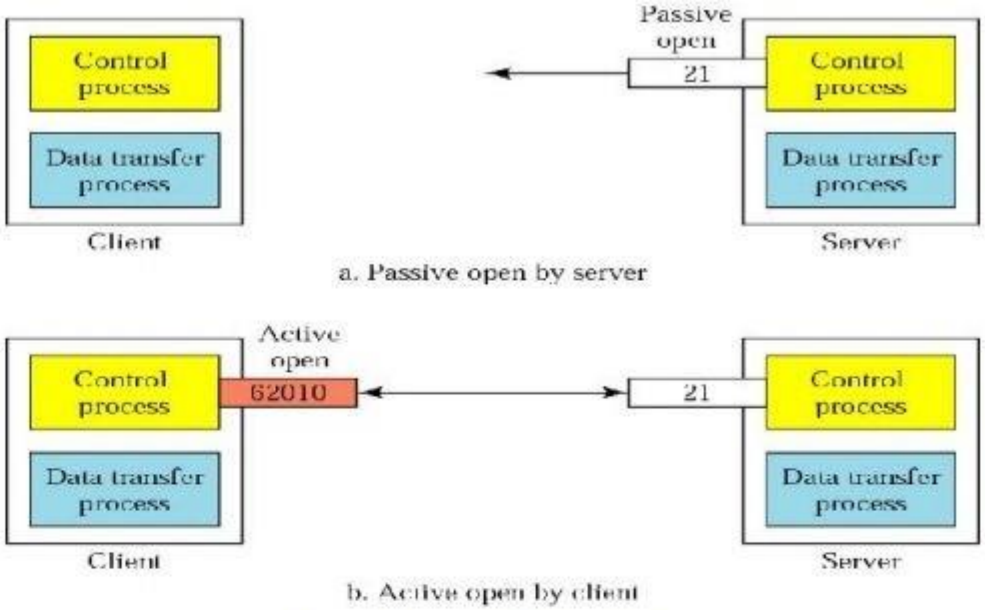


Figure Control Connection

2. 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. This is done under the supervision of the LIST command. Note that FTP treats a list of directory or file names as a file. It is sent over the data connection.

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. The heterogeneity problem is resolved by defining three attributes of communication: file type, data structure, and transmission mode

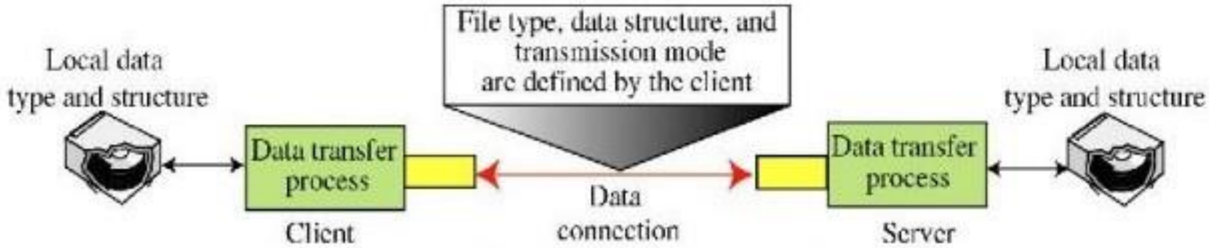

File type, data structure, and transmission mode are defined by the client
Local data type and structure
Data transfer process
Client
Data connection
Data transfer process
Server
Local data type and structure

Figure Data Connection

File Type:

FTP can transfer one of the following file types across the data connection: an ASCII file, EBCDIC file, or image file. The ASCII file is the default format for transferring text files. Each character is encoded using 7-bit ASCII. The sender transforms the file from its own representation into ASCII characters, and the receiver transforms the ASCII characters to its own representation.

Data Structure

FTP can transfer a file across the data connection by using one of the following interpretations about the structure of the data: file structure, record structure, and page structure. In the file structure format, the file is a continuous stream of bytes. In the record structure, the file is divided into records.

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. Data are delivered from FTP to TCP as a continuous stream of bytes. TCP is responsible for chopping data into segments of appropriate size. If the data are simply a stream of bytes (file structure), no end-of-file is needed. End-of-file in this case is the closing of the data connection by the sender.

WWW :

The World Wide Web (WWW) is a repository of information linked together from points all over the world. The WWW has a unique combination of flexibility, portability, and user-friendly features that distinguish it from other services provided by the Internet. The WWW project was initiated by CERN (European Laboratory for Particle Physics) to create a system to handle distributed resources necessary for scientific research.

1. Architecture

The WWW today is a distributed client/server 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. Each Web page can contain a link to other pages in the same site or at other sites. The request, among other information, includes the address of the site and the Web page, called the URL, which we will discuss shortly. The server at site A finds the document and sends it to the client. When the user views the document, she finds some references to other documents, including a Web page at site B. The reference has the URL for the new site. The user is also interested in seeing this document. The client sends another request to the new site, and the new page is retrieved.

Client (Browser)

A variety of vendors offer commercial browsers that interpret and display a Web document, and all 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 controller uses one of the interpreters to display the document on the screen. The client protocol can be one of the protocols such as FTP or HTTP.

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 requested 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.

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. The URL defines four things: protocol, host computer, port, and path.

The protocol is the client/server program used to retrieve the document. Many different protocols can retrieve a document; among them are FTP or 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. Web pages are usually stored in computers, and computers are given alias names that usually begin with the characters "www". This is not mandatory, however, as the host can be any name given to the computer that hosts the Web page. The URL can optionally contain the port number of the server. If the port is included, it is inserted between 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 directories from the subdirectories and files.

2. 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 at which the contents of the document are determined.

1. 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. In other words, the contents of the file are determined when the file is created, not when it is used. Of course, 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.

HTML

Hypertext Markup Language (HTML) is a language for creating Web pages. The term markup language comes from the book publishing industry. Before a book is typeset and printed, a copy editor reads the manuscript and puts marks on it. These marks tell the compositor how to format the text. For example, if the copy editor wants part of a line to be printed in boldface, he or she draws a wavy line under that part. In the same way, data for a Web page are formatted for interpretation by a browser

The two tags <B> and </B> are instructions for the browser. When the browser sees these two marks, it knows that the text must be boldfaced. A markup language such as HTML allows us to embed formatting instructions in the file itself. The instructions are included with the text. In this way, any browser can read the instructions and format the text according to the specific workstation.

A Web page is made up of two parts: the head and the body. The head is the first part of a Web page. The head contains the title of the page and other parameters that the browser will use. The actual contents of a page are in the body, which includes the text and the tags. Whereas the text is the actual information contained in a page, the tags define the appearance of the document. Every HTML tag is a name followed by an optional list of attributes, all enclosed between less-than and greater-than symbols (< and >). An attribute, if present, is followed by an equal's sign and the value of the attribute. Some tags can be used alone; others must be used in pairs. Those that are used in pairs are called beginning and ending tags. The beginning tag can have attributes and values and starts with the name of the tag. The ending tag cannot have attributes orvalues but must have a slash before the name of the tag.

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. Because a fresh document is created for each request, the contents of a dynamic document can vary from one request to another.

Common Gateway Interface (CGI)

The Common Gateway Interface (CGI) is a technology that creates and handles dynamic documents. CGI is a set of standards that defines how a dynamic document is written, how data are input to the program, and how the output result is used. The term common in CO1 indicates that the standard defines a set of rules that is common to any language or platform. The term gateway here means that a COl program can be used to access other resources such as databases, graphical packages, and so on. The term interface here means that there is a set of predefined terms, variables, calls, and so on that can be used in any COl program.

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. It is similar to FTP because it transfers files and uses the services of TCP. However, it is much simpler than FTP because it uses only one TCP connection. There is no separate control connection; only data are transferred between the client and the server. HTTP is like SMTP because the data transferred between 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 interpreted by the HTTP server and HTTP client (browser). SMTP messages are stored and forwarded, but HTTP messages are delivered immediately. The commands from the client to the server are embedded in a request message. The contents of the requested file or other information are embedded in a response message. HTTP uses the services of TCP on well-known port 80.

1. HTTP Transaction

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 are similar. 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. 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 version1.1of HTTP, several request types are defined.

Version: The most current version of HTTP is 1.1.

Status code: This field is used in the response message. The status code field is similar tothose in the FTP and the SMTP protocols. It consists of three digits.

Status phrase: This field is used in the response message. It explains the status code intext form.

Header: The header exchanges additional information between the client and the server.For example, the client can request that the document be sent in a special format, or the server can send extra information about the document. The header can consist of one or more header lines. Each header line has a header name, a colon, a space, and a header value.

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

2. Persistent Versus Non persistent Connection

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

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 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. However, there are some occasions when the sender does not know the length of the data. This is the case when a document is created dynamically or actively.

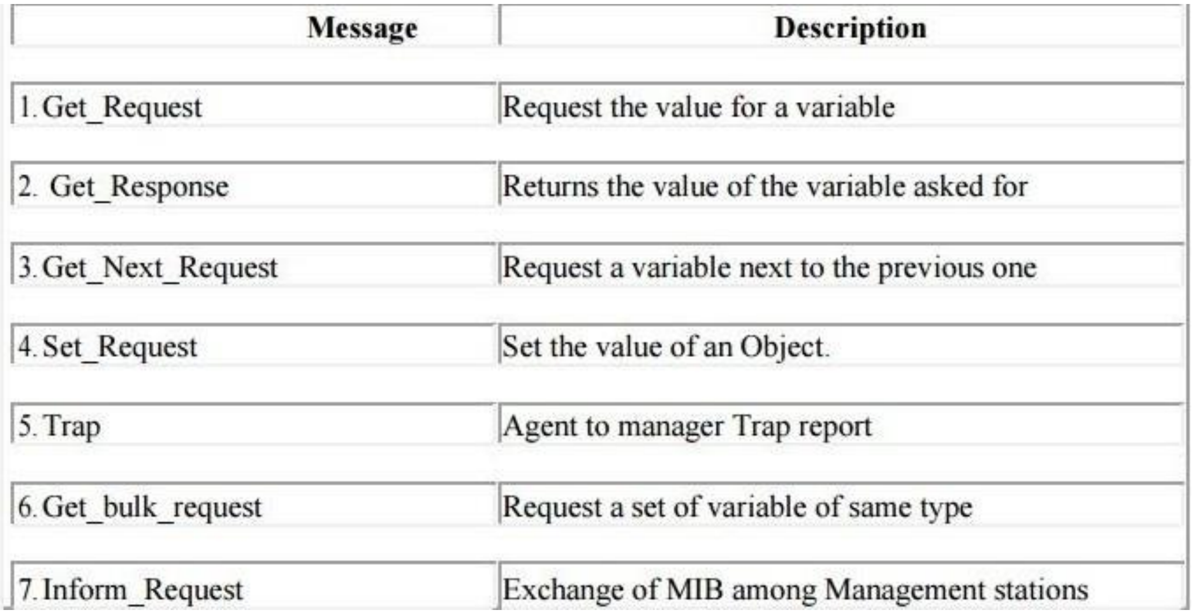
SNMP (SIMPLE NETWORK MANAGEMENT PROTOCOL)

A large network can often get into various kinds of trouble due to routers (dropping too many packets), hosts(going down) etc. One has to keep track of all these occurence and adapt to such situations. A protocol has been defined. Under this scheme all entities in the network belong to 4 classes:

1. Managed Nodes
2. Management Stations
3. Management Information (called Object)
4. A management protocol

The managed nodes can be hosts,routers,bridges,printers or any other device capable of communicating status information to others. To be managed directly by SNMP, a node must be capable of running an SNMP management process, called SNMP agent. Network management is done by management stations by exchanging information with the nodes. These are basically general purpose computers running special management software. The management stations poll the stations periodically. Since SNMP uses unreliable UDP service of UDP the polling is essential to keep in touch with the nodes. Often the nodes send a trap message indicating that it is going to go down. The management stations then periodically check (with an increased frequency) . This type of polling is called trap directed polling. Often a group of nodes are represented by a single node which communicates with the management stations. This type of node is called a proxy agent. The proxy agent can also serve as a security arrangement. All the variables in these schemes are called Objects. Each variable can be referenced by a specific addressing scheme adopted by this system. The entire collection of all objects is called Management Information Base (MIB). The addressing is hierarchical as seen in the picture.

The Internet is addressed as 1.3.61. All the objects under this domain have this string at the beginning. The information is exchanged in a standard and vendor-neutral way . All the data are represented in Abstract Syntax Notation 1 (ASN.1). It is similar to XDR as in RPC but it has a widely different representation scheme. A part of it was actually adopted in SNMP and modified to form the Structure Of Information Base. The Protocol specifies various kinds of messages that can be exchanged between the managed nodes and the management station.



Message	Description
1. Get_Request	Request the value for a variable
2. Get_Response	Returns the value of the variable asked for
3. Get_Next_Request	Request a variable next to the previous one
4. Set_Request	Set the value of an Object.
5. Trap	Agent to manager Trap report
6. Get_bulk_request	Request a set of variable of same type
7. Inform_Request	Exchange of MIB among Management stations

Message : Description

1. Get_Request : Request the value for a variable

2. Get_Response : Returns the value of the variable asked for

3. Get_Next_Request : Request a variable next to the previous one

4. Set_Request : Set the value of an Object.

5. Trap : Agent to manager Trap report

6. Get_bulk_request : Request a set of variable of same type

7. Inform_Request : Exchange of MIB among Management stations

The last two options have been actually added in the SNMPv2. The fourth option need some kind of authentication from the management station.

Addressing Example :

Following is an Example of the kind of address one can refer to when fetching a value in the table :-

```
        (20) IP-Addr-Table = Sequence of IPAddr-Entry (1)
       IPAddrEntry = SEQUENCE {
                 IPADDENTRYADDR      : IPADDR (1)
                     Index           : integer (2)
                     Netmask         :  IPAddr (3)        }
```

So when accessing the netmask of some IP-entity the variable name would be : 1.3.6.1.2.4.20 .1.3.key-value

Here since Ip-address the unique key to index any member of the array the address can be like :- 1.3.6.1.2.4.20.1.3.128.10.2.3

Multimedia :

Recent advances in technology have changed our use of audio and video. In the past, we listened to an audio broadcast through a radio and watched a video program broadcast through a TV. We used the telephone network to interactively communicate with another party. But times have changed. People want to use the Internet not only for text and image communications, but also for audio and video services.

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

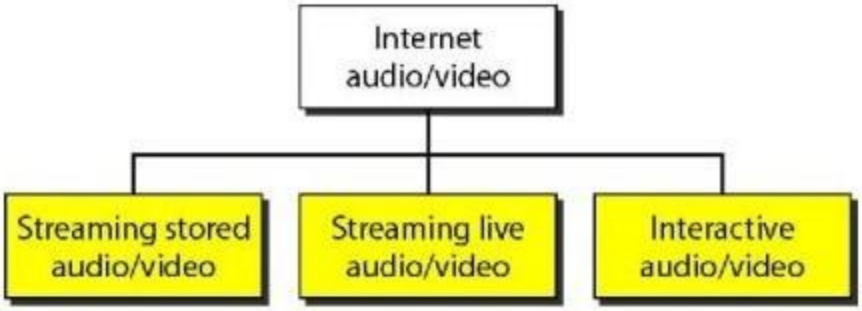


Figure Internet audio/video

1. Streaming stored audio/video, the files are compressed and stored on a server. A client downloads the files through the Internet.

2. Streaming live audio/video, a user listens to broadcast audio and video through the Internet.

3. Interactive audio/video, people use the Internet to interactively communicate with one another.

1. Digitizing Audio and Video

Before audio or video signals can be sent on the Internet, they need to be digitized.

Digitizing Audio

When sound is fed into a microphone, an electronic analog signal is generated which 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. According to the Nyquist theorem, if the highest frequency of the signal is f, we need to sample the signal 21 times per second. There are other methods for digitizing an audio signal, but the principle is the same.

Digitizing Video

A video consists 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. However, to avoid a condition known as flickering, a frame needs to be refreshed. The TV industry repaints each frame twice. This means 50 frames need to be sent, or if there is memory at the sender site, 25 frames with each frame repainted from the memory.

2. Audio and Video Compression

To send audio or video over the Internet requires 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 1.41 I-MHz signal. Two categories of techniques are used for audio compression: predictive encoding and perceptual encoding.

a. 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).

b. Perceptual Encoding: MP3

The most common compression technique that is used to create CD-quality audio is based on the perceptual encoding technique. As we mentioned before, this type of audio needs at least 1.411 Mbps; this cannot be sent over the Internet without compression. MP3 (MPEG audio layer 3), a part of the MPEG standard (discussed in the video compression section), uses this technique.

Video Compression

As we mentioned before, 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. Joint Photographic Experts Group (JPEG) is used to compress images. Moving Picture Experts Group (MPEG) is used to compress video.

a. Image Compression: JPEG

If the picture is not in color (gray scale) then each pixel can be represented by an 8-bit integer (256 levels). If the picture is in color, each pixel can be represented by 24 bits (3 x 8 bits), with each 8 bits representing red, blue, or green (RBG). To simplify the discussion, we concentrate on a grayscale picture.

b. Video Compression: MPEG

The Moving Picture Experts Group 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 spatial 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.

3. Streaming Live Audio/video

Streaming live audio/video is similar to the broadcasting of audio and video by radio and TV stations. Instead of broadcasting to the air, the stations broadcast through the Internet. There are several similarities between streaming stored audio/video and streaming live audio/video. They are both sensitive to delay; neither can accept retransmission. However, there is a difference. In the first application, the communication is unicast and on-demand. In the second, the communication is multicast and live. Live streaming is better suited to the multicast services of IP and the use of protocols such as UDP and RTP.

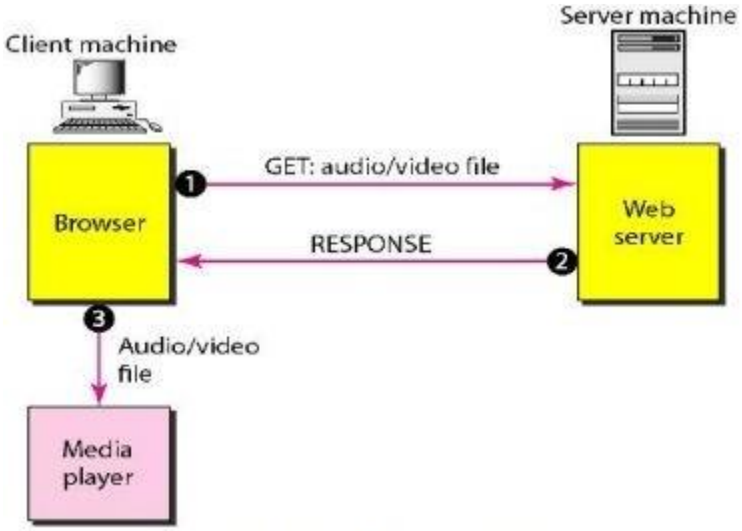


Figure Using a Web server

4. Real-Time Interactive Audio/video

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