Application Layer:

Client Server Model, Socket Interface, Domain Name System (DNS):

DNS : Domain Name System

Introduction

Where?

DNS is an intricate worldwide network of web servers that collectively comprise a global database of domain names and IP addresses.

Why?

- In data networks, devices are assigned IP addresses so that they can participate in sending and receiving messages over the network.
- Most people have a hard time remembering this numeric address.
- Domain names were created to convert the numeric address into a simple, recognizable name.

What?

- The Domain name system is the central point of the entire internet, and it is directly responsible for the way web addresses are used.
- The Domain Name System (DNS) is a supporting program that is used by other programs such as e-mail.

Figure.15 shows an example of how a DNS client/server program can support an e-mail program to find the IP address of an e-mail recipient. A user of an e-mail program may know the e-mail address of the recipient; however, the IP protocol needs the IP address. The DNS client program sends a request to a DNS server to map the e-mail address to the corresponding IP address.

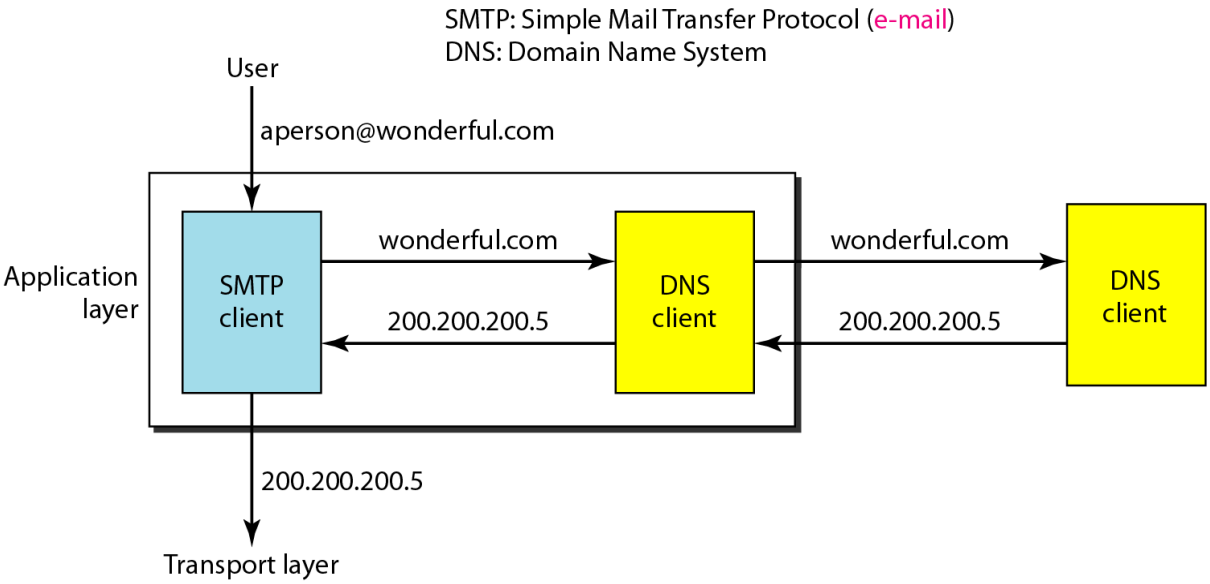


Figure.15 Example of using the DNS services

Concept

How DNS Work?

1) Needs this IPaddress (www.anywebsite.com).

2) This address not in DNS server1 cache will try another DNS server.

3) This address is in DNS Server2 cache. It maps to that IP address.

4) Again DNS server1 will cache that information.

5) DNS client communicate with that website server.

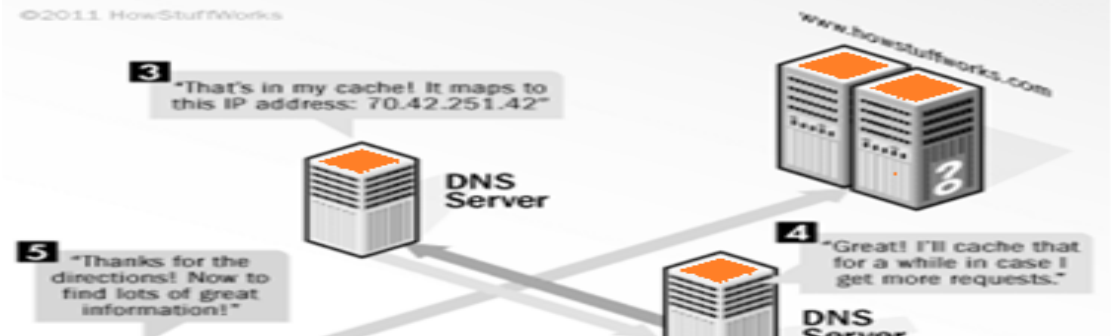


Figure.16 DNS Working

- To identify an entity, TCP/IP protocols use the IP address, which uniquely identifies the connection of a host to the Internet. However, people prefer to use names instead of numeric addresses. Therefore, we need a system that can map a name to an address or an address to a name.
- When the Internet was small, mapping was done by using a host file. The host file had only two columns: name and address. Every host could store the host file on its disk and update it periodically from a master host file. When a program or a user wanted to map a name to an address, the host consulted the host file and found the mapping.
- Today, however, it is impossible to have one single host file to relate every address with a name and vice versa. The host file would be too large to store in every host. In addition, it would be impossible to update all the host files every time there was a change.
- One solution would be to store the entire host file in a single computer and allow access to this centralized information to every computer that needs mapping. But we know that this would create a huge amount of traffic on the Internet.
- Another solution, the one used today, is to divide this huge amount of information into smaller parts and store each part on a different computer. In this method, the host that needs mapping can contact the closest computer holding the needed information. This method is used by the Domain Name System (DNS).

NAME SPACE

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

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 names may or may not have a common section; if they do, it has no meaning. The main disadvantage of a fiat 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.

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. In this case, the authority to assign and control the name spaces can be decentralized. A central authority can assign the part of the name that defines the nature of the organization and the name of the organization.

DOMAIN NAME SPACE

To have a hierarchical name space, a domain name space was designed. 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.

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). DNS requires that children of a node (nodes that branch from the same node) have different labels, which guarantees the uniqueness of the domain names.

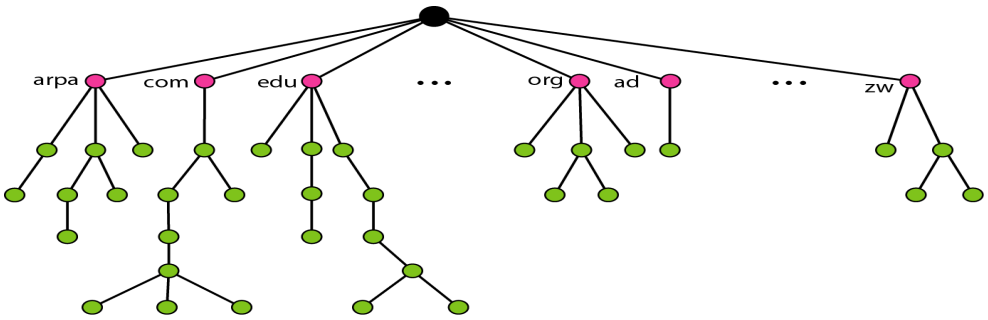


Figure.17 Domain Name Space

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.

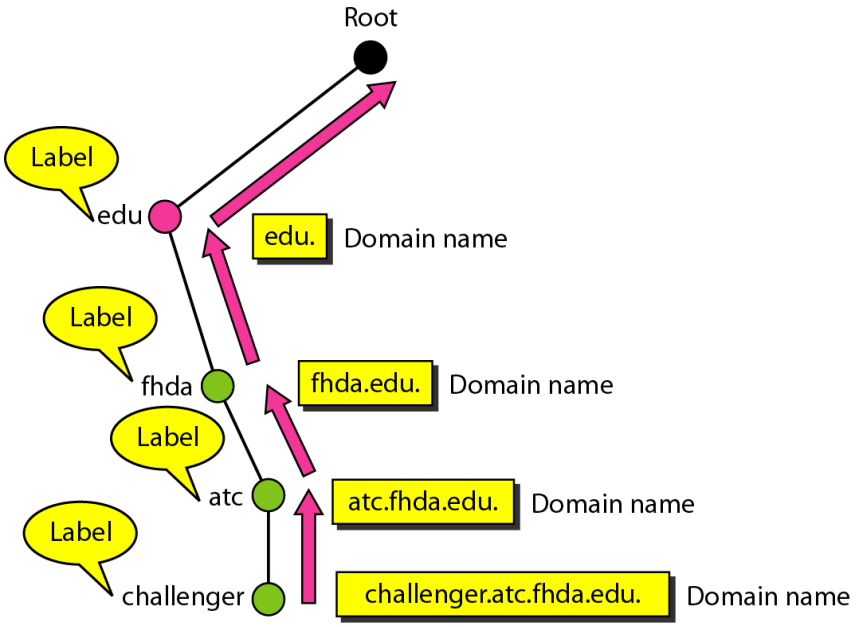


Figure.18 Domain Name and labels

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 domain name that contains the full name of a host. It contains all labels, from the most specific to the most general, that uniquely define the name of the host.

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. It is used when the name to be resolved belongs to the same site as the client. Here the resolver can supply the missing part, called the suffix, to create an FQDN.

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.19 shows some domains. Note that a domain may itself be divided into domains (or subdomains as they are sometimes called).

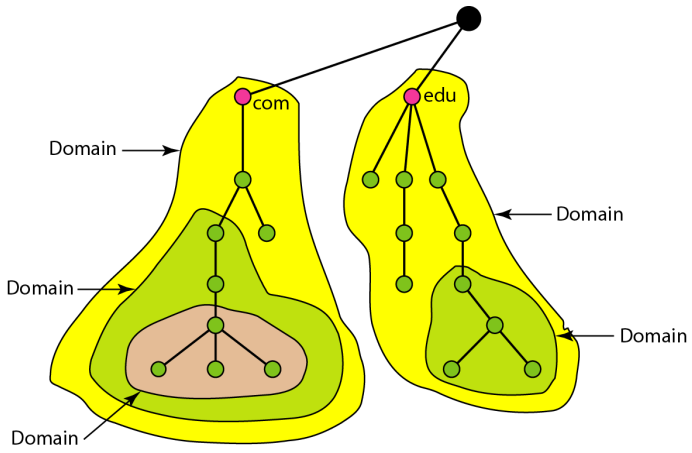


Figure.19 Domain

Hierarchy of Name Servers

- The information contained in the domain name space must be stored. However, it is very inefficient and also unreliable to have just one computer store such a huge amount of information. It is inefficient because responding to requests from all over the world

places a heavy load on the system. It is not unreliable because any failure makes the data inaccessible.

- The solution to these problems is to distribute the information among many computers called DNS servers. One way to do this is to divide the whole space into many domains based on the first level. In other words, we let the root stand alone and create as many domains (subtrees) as there are first-level nodes. Because a domain created in this way could be very large, DNS allows domains to be divided further into smaller domains (subdomains). Each server can be responsible (authoritative) for either a large or a small domain.

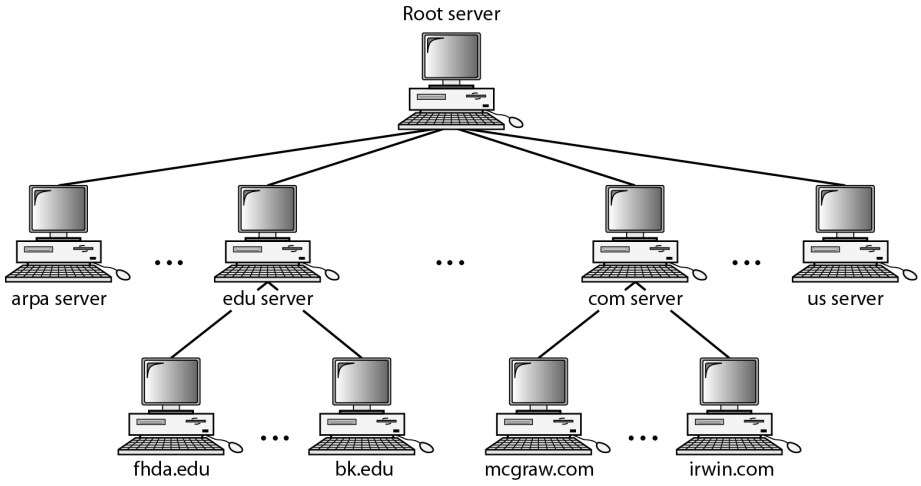


Figure.20 Hierarchy of name servers

Zone

Since the complete domain name hierarchy cannot be stored on a single server, it is divided among many servers. What a server is responsible for or has authority over is called a zone. We can define a zone as a contiguous part of the entire tree. If a server accepts responsibility for a domain and does not divide the domain into smaller domains, the domain and the zone refer to the same thing. The server makes a database called a zone file and keeps all the information for every node under that domain. However, if a server divides its domain into subdomains and delegates part of its authority to other servers, domain and zone refer to different things.

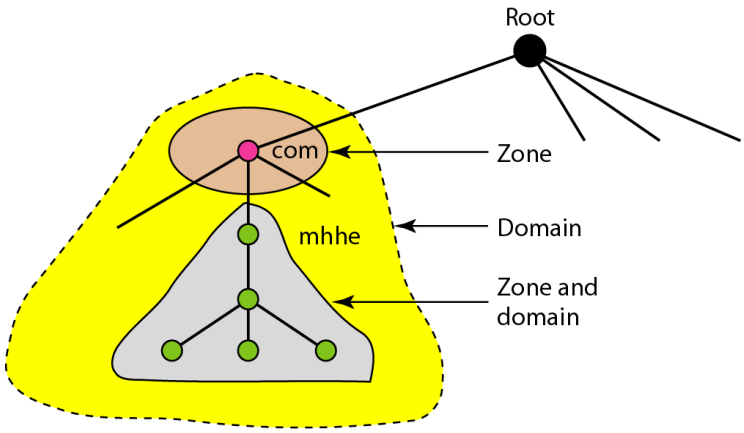


Figure.21 Zones and Domains

Root Server

A root server is a server whose zone consists of the whole tree. A root server usually does not store any information about domains but delegates its authority to other servers, keeping references to those servers. There are several root servers, each covering the whole domain name space. The servers are distributed all around the world.

Primary and Secondary Servers

DNS defines two types of servers: primary and secondary.

- A primary server is a server that stores a file about the zone for which it is an authority. 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. The secondary server neither creates nor updates the zone files. If updating is required, it must be done by the primary server, which sends the updated version to the secondary.
- The primary and secondary servers are both authoritative for the zones they serve.

A primary server loads all information from the disk file; the secondary server loads all information from the primary server. When the secondary downloads information from the primary, it is called zone transfer.

DNS in the Internet

DNS 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.

Generic Domains

The generic domains define registered hosts according to their generic behaviour. Each node in the tree defines a domain, which is an index to the domain name space database.

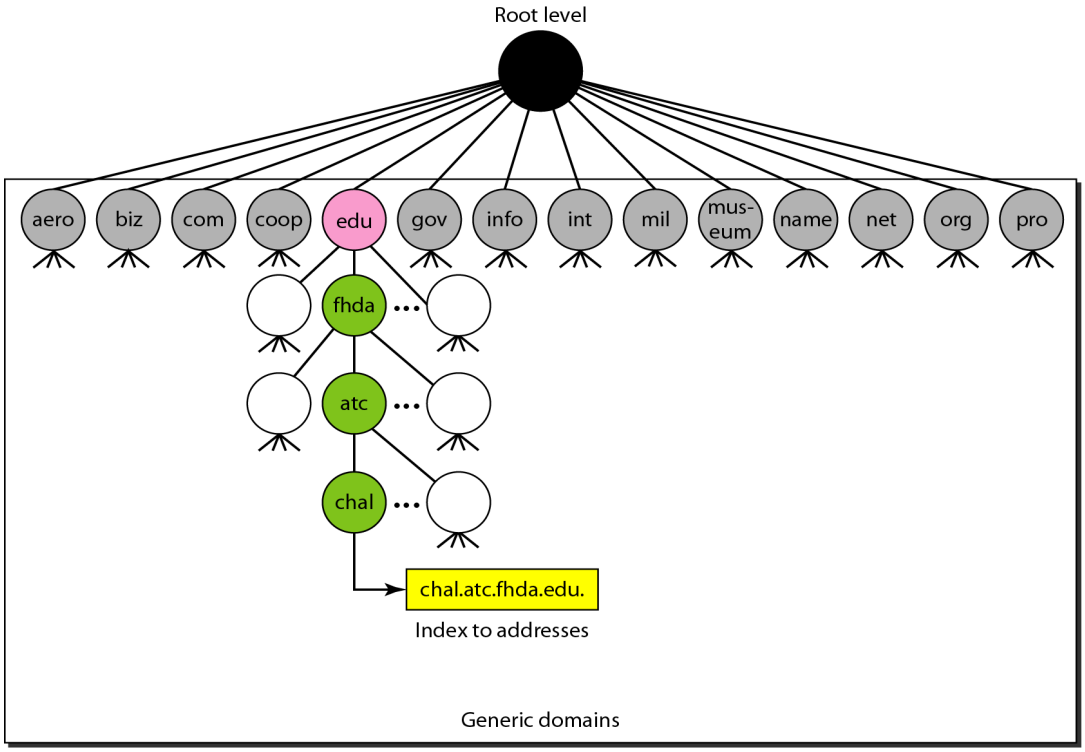


Figure.22 Generic Domain

Looking at the tree, we see that the first level in the generic domains section allows 14 possible labels. These labels describe the organization types as listed in Table.3

Table.3

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

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.).

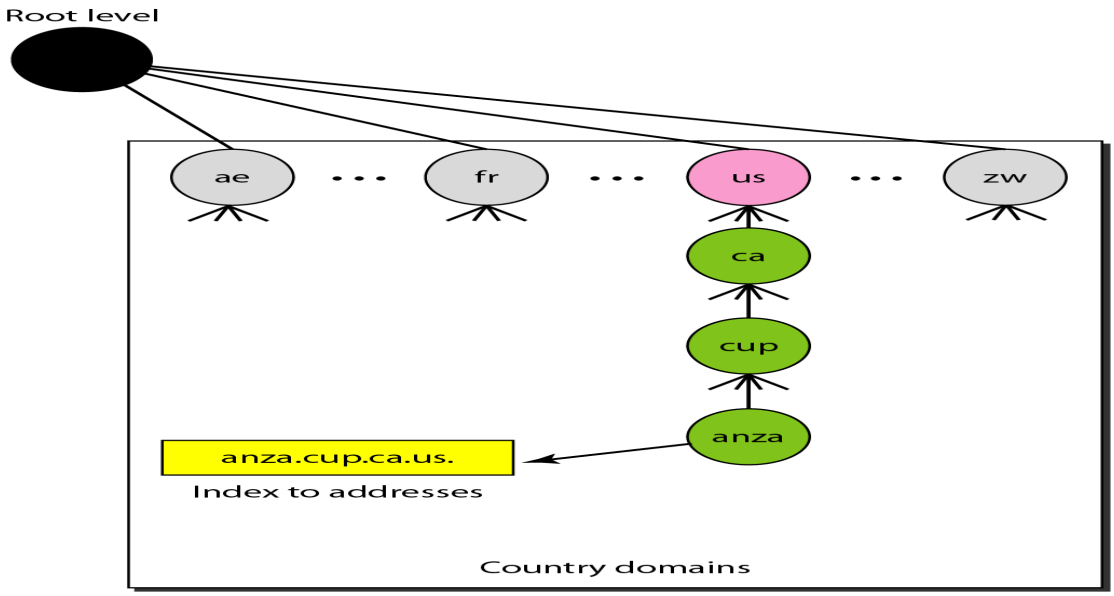


Figure.23 Country Domains

Inverse Domain

- The inverse domain is used to map an address to a name. This may happen, for example, when a server has received a request from a client to do a task. Although the server has a file that contains a list of authorized clients, only the IP address of the client (extracted from the received IP packet) is listed. The server asks its resolver to send a query to the DNS server to map an address to a name to determine if the client is on the authorized list.
- This type of query is called an inverse or pointer (PTR) query. To handle a pointer query, the inverse domain is added to the domain name space with the first-level node called arpa (for historical reasons). The second level is also one single node named in-addr (for inverse address). The rest of the domain defines IP addresses.
- The servers that handle the inverse domain are also hierarchical. This means the net_id part of the address should be at a higher level than the subnet-id part, and the subnet_id part higher than the host_id part. In this way, a server serving the whole site is at a higher level than the servers serving each subnet. This configuration makes the domain look inverted when compared to a generic or country domain. To follow the convention of reading the domain labels from the bottom to the top, an IF address such as 132.34.45.121 (a class B address with netid 132.34) is read as 121.45.34.132.in-addr. arpa.

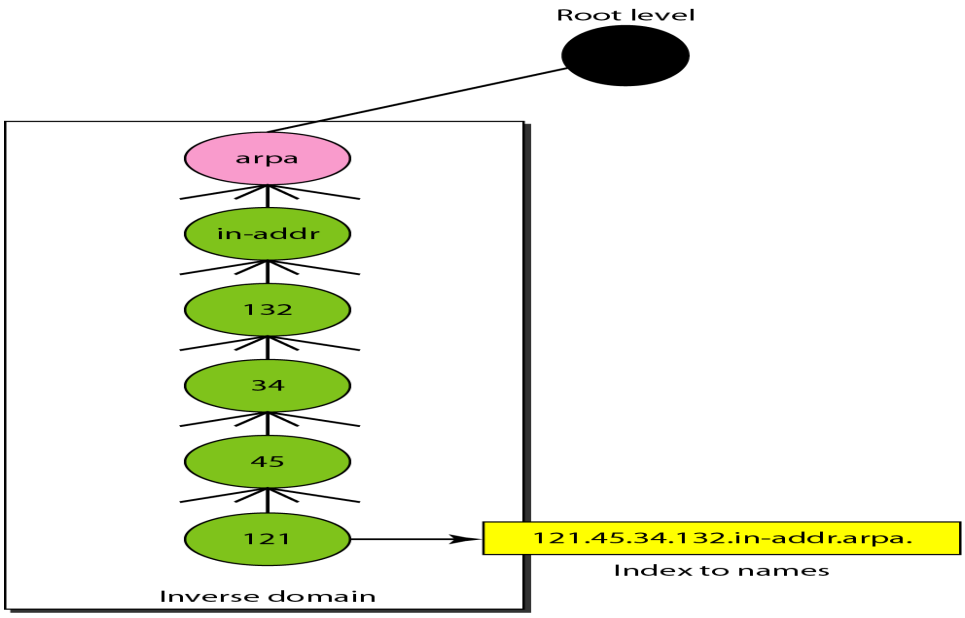


Figure.24 Inverse Domain

Resolving Domain Name to IP address (DNS Operation)

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. The resolver accesses the closest DNS server with a mapping request. If the server has the information, it satisfies the resolver; otherwise, it either refers the resolver to other servers or asks other servers to provide the information.
- After the resolver receives the mapping, it interprets the response to see if it is a real resolution or an error, and finally delivers the result to the process that requested it.

Mapping Names to Addresses

- Most of the time, the resolver gives a domain name to the server and asks for the corresponding address. In this case, the server checks the generic domains or the country domains to find the mapping.
- If the domain name is from the generic domains section, the resolver receives a domain name such as "chal.atc.jhda.edu.". The query is sent by the resolver to the local DNS

server for resolution. If the local server cannot resolve the query, it either refers the resolver to other servers or asks other servers directly.

- If the domain name is from the country domains section, the resolver receives a domain name such as "ch.jhda.cu.ca.us.". The procedure is the same.

Mapping Addresses to Names

A client can send an IP address to a server to be mapped to a domain name. As mentioned before, this is called a PTR query. To answer queries of this kind, DNS uses the inverse domain. However, in the request, the IP address is reversed and the two labels in-addr and arpa are appended to create a domain acceptable by the inverse domain section. For example, if the resolver receives the IF address 132.34.45.121, the resolver first inverts the address and then adds the two labels before sending. The domain name sent is "121.45.34.132.in-addr.arpa." which is received by the local DNS and resolved.

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.

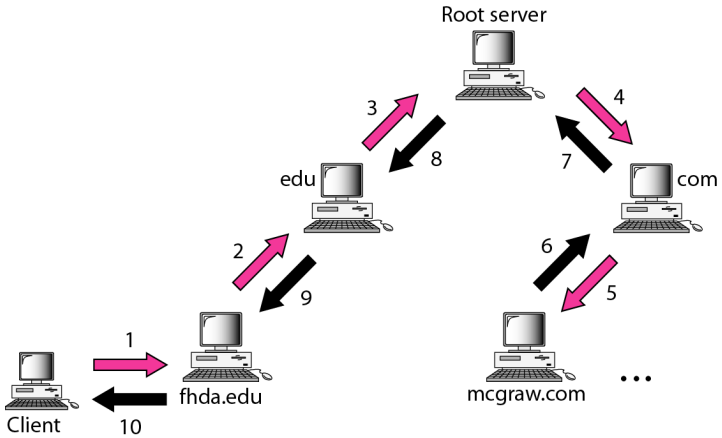


Figure.25 Recursive Resolution

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. If the newly addressed server can resolve the problem, it answers the query with the IP address; otherwise, it returns the IP address of a new server to the client. Now the client must repeat the query to the third server. This process is called iterative resolution because the client repeats the same query to multiple servers.

Caching

Each time a server receives a query for a name that is not in its domain, it needs to search its database for a server IP address. Reduction of this search time would increase efficiency. DNS handles this with a mechanism called caching. When a server asks for a mapping from another server and receives the response, it stores this information in its cache memory before sending it to the client. If the same or another client asks for the same mapping, it can check its cache memory and solve the problem. However, to inform the client that the response is coming from the cache memory and not from an authoritative source, the server marks the response as unauthoritative.

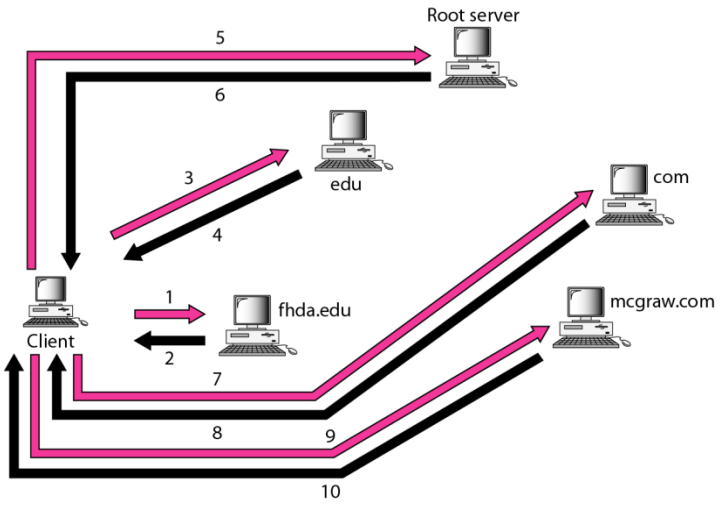


Figure.25 Iterative Resolution

Domain name Disputes

- Domain names have become precious commodities as the internet has no boundaries and no closing hours and unlike trade marks each domain name is unique. This means there can often be some pressure to be the first to register a domain name, and this can lead to disputes on the "right" to register. As a result domain name disputes can arise.
- Domain names and disputes are managed according to policies set by ICANN (Internet Corporation for Assigned Names and Numbers).
- ICANN passes over responsibility for the registration of domain names to registrars. There are over 1,500 accredited registrars each with their own policies and procedures for registering, maintaining and handling domain names. Registrars are determined not to be included or accept liability for any dispute. The courts have maintained this protection for registrars.
- Disputes concerning so called generic Top Level Domains (gTLD) such as .biz, .com, .info, .name, .net, and .org. are managed by an ICANN-authorised dispute resolution service provider.
- Disputes concerning country code Top Level Domains (ccTLD) such as .co.uk, .fr (France), .de (Germany) are managed by the individual country registry. Nominet is the registry for the UK.

Disputes that arise

Concurrent Rights

- Registration is on a first come first serve basis. Simply because you have a registered trade or service mark, have a registered company name, or have been using a trade name for a lengthy period of time does not mean that another person with a legitimate reason for registering the domain and who uses it in good faith must give it up. One example of this is the Prince Sports case in which Prince Sports tried in vain to have the domain www.prince.com transferred from Prince Computers in the UK.
- Companies with trade marks have tried to bully legitimate registrants out of attractive domain names, this has sometimes been called Reverse Domain Name Hijacking and damages can now be awarded in the US under the US Cybersquatting Act for such practice.

Cybersquatting

These are common disputes Cybersquatting involves the registrant having registered a name, or names in most cases, in bad faith to gain some commercial advantage. This can involve trying to sell it back to a party it knows would be interested in having registration of the domain name for an inflated price or more commonly using it to direct traffic to their website or the website of a trade competitor of the trade mark holder in return for payment of a commission.

Gripe sites

Sites such as www.natwestsucks.com or www.stopecg.com have been problematic. Arbitrators and the courts have been inclined to order the transfer of the offending domain name particularly if there is some bad faith or a lack of legitimate use. Reasoning for this has been that a non-native English speaker may not disassociate the "suck" from the trade mark holder's mark.

Some Other Domain name Disputes

Because of the increasing popularity of the Internet, companies have realized that having a domain name that is the same as their company name or the name of one of their products can be an extremely valuable part of establishing an Internet presence. A company wishing to acquire a domain name must file an application with the appropriate agency. Before doing so, a search is done to see if their desired domain name is already taken. A good site for doing such a search is provided by Network Solutions. When a company finds that the domain name corresponding to their corporate name or product trademark is owned by someone else, the company can either choose a different name or fight to get the domain name back from its current owners.

Some well publicized examples of these types of domain names disputes are:

- *candyland.com:* Both Hasbro and an adult entertainment provider desired the candyland.com domain name. Hasbro was too late to register the name itself, but it is never too late to sue (well, almost never). The domain name is now safely in the hands of Hasbro.
- *mcdonalds.com*: This domain name was taken by an author from Wired magazine who was writing a story on the value of domain names. In his article, the author requested that people contact him at ronald@mcdonalds.com with suggestions of what to do with the domain name. In exchange for returning the domain name to McDonalds, the author convinced the company to make a charitable contribution.

- *Micros0ft.com*: The company Zero Micro Software obtained a registration for micros0ft.com (with a zero in place of the second 'o'), but the registration was suspended after Microsoft filed a protest. When the domain name went abandoned for non-payment of fees, the domain name was picked up by someone else: Vision Enterprises of Roanoke, TX.
- *mtv.com*: The MTV domain name was originally taken by MTV video jockey Adam Curry. Although MTV originally showed little interest in the domain name or the Internet, when Adam Curry left MTV the company wanted to control the domain name. After a federal court action was brought, the dispute settled out of court.
- *peta.org*: An organization entitled "People Eating Tasty Animals" obtained the peta.org domain name, much to the disgust of the better know People for the Ethical Treatment of Animals. This domain name was suspended, but as of May 2000 the domain name was still registered in the name of People Eating Tasty Animals.
- *roadrunner.com*: When NSI threatened to suspend the roadrunner.com domain name after a protest by Warner Brothers, the New Mexico Internet access provider who was using the domain name filed suit to prevent the suspension. Although the access provider was able to prevent the suspension, a joint venture company involving Time Warner, MediaOne, Microsoft, Compaq, and Advance/Newhouse eventually obtained the domain name.
- *taiwan.com*: The mainland China news organization Xinhua was allowed to register the domain name taiwan.com, much to the disgust of the government of Taiwan.

Legal Remedies for the disputes

- When a dispute over a domain name occurs, such as those described above, the parties can always turn to the courts. While courts and judges have the authority to award control and ownership over domain names (just as they have authority to award control and ownership over any other property), the judicial process is notoriously slow. Consequently, many parties have avoided the courts and turned to the domain name dispute policies of the domain name registrars.
- Companies that do bring a court action must present legal arguments on why a domain name registered to someone else should be cancelled or transferred to an organization who wasn't fast enough to register the name first. Historically, these arguments were

based on trademark law or dilution law (which are discussed in more detail on the BitLaw section on trademarks on the Internet). It was sometimes difficult to present a strong case under the traditional principals of trademark law, especially when the party seeking to obtain a domain name either could not prove a likelihood of confusion (which is required under trademark law) or was a famous individual who never technically established trademark rights in their name.

- In response to intense lobbying from trademark owners and famous individuals, Congress passed the Anticybersquatting Consumer Protection Act in November of 1999. This act made it easier for individuals and companies to take over domain names that are confusingly similar to their names or valid trademarks. To do so, however, they must establish that the domain name holder acted in bad faith.

The more general portion of the statute protects companies against persons who, in bad faith, register a domain name that is the same or confusingly similar to an existing trademark. The statute the lists the following factors as elements that a court can consider to determine whether the domain name was registered in bad faith.

- ✓ Does the domain name holder have trademark rights in the domain name?
- ✓ Is the domain name the legal name of the domain name holder, or some other name that is otherwise commonly used to identify that person?
- ✓ Has the domain name holder made use (prior to the dispute) of the domain name in connection with a bona fide sale of goods or services?
- ✓ Is the domain name holder using the mark in a bona fide noncommercial or fair use way at a web site accessible at the domain name?
- ✓ Is the domain name holder attempting to divert consumers from the trademark owner's web site in a confusing way, either for commercial gain or in an attempt to tarnish or disparage the trademark mark?
- ✓ Has the domain name holder offered to sell the domain name to the trademark owner (or anyone else) for financial gain without having any intent to use the mark with the sale of goods or services?
- ✓ Has the domain name holder behaved in a pattern of registering and selling domain names without intending to use them in connection with the sale of goods or services?

- ✓ Did the domain name holder provide false information when applying for the registration of the domain name (or do so in connection with other domain names)?
- ✓ Has the domain name holder registered domain names of other parties trademarks?
- ✓ How distinctive and famous is the trademark owner's trademark?

Electronic Mail (SMTP) and

file transfer (FTP)

FTP

1. Introduction:

- **File Transfer Protocol** (**FTP**)is a standard network protocol used to transfer files from one host to another host over a TCP-based network, such as the Internet and used in Application layer of TCP/IP suite.

- The main objective of this protocol is:
  1. To transfer data reliably and efficiently
  2. To promote sharing of files (compute programs and /or data)
  3. To transfer files between FTP client and FTP servers(file download, upload)

- While transferring data over the network, four data representations can be used

  1. ASCII mode
  2. Image mode (commonly called Binary mode)
  3. EBCDIC mode
  4. Local mode

Data transfer can be done in any of three modes:

- Stream mode: Data is sent as a continuous stream,
- Block mode: FTP breaks the data into several blocks
- Compressed mode:

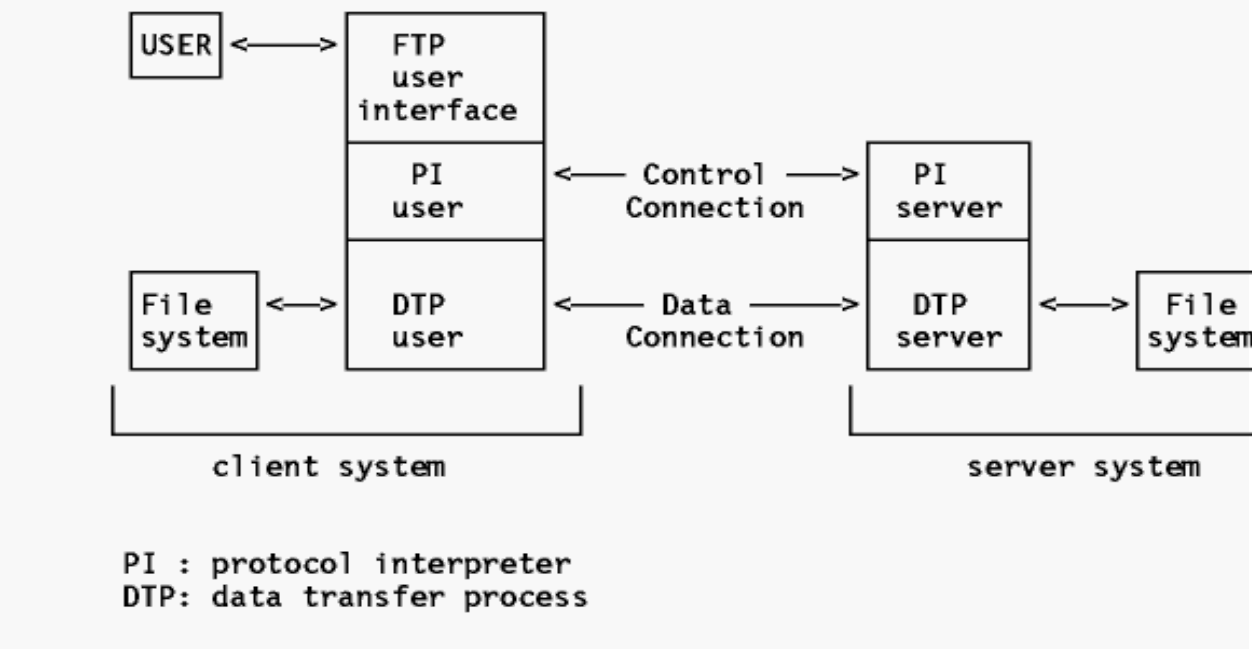
2. History:

- The original specification for the File Transfer Protocol was written by **AbhayBhushan** and published as RFC 114 on 16 April 1971 and later replaced by RFC 765 (June 1980) and RFC 959 (October 1985), the current specification. Several proposed standards amend RFC 959, for example RFC 2228 (June 1997) proposes security extensions and RFC 2428 (September 1998) adds support for IPv6 and defines a new type of passive mode.

- **A Request for Comments** (**RFC**) is a publication of the Internet Engineering Task Force (**IETF**) and the Internet Society, the principal technical development and standards-setting bodies for the Internet.

3. The FTP Model

- FTP uses TCP as transport protocol to provide reliable end-to-end connections.
- Two connections are used: the first is the control connection and the second is the data connection that is managing the data transfer.
- On both sides of the link the FTP application is built with a protocol interpreter (PI) and a data transfer process (DTP). On the client side of the link there exists also a user interface.



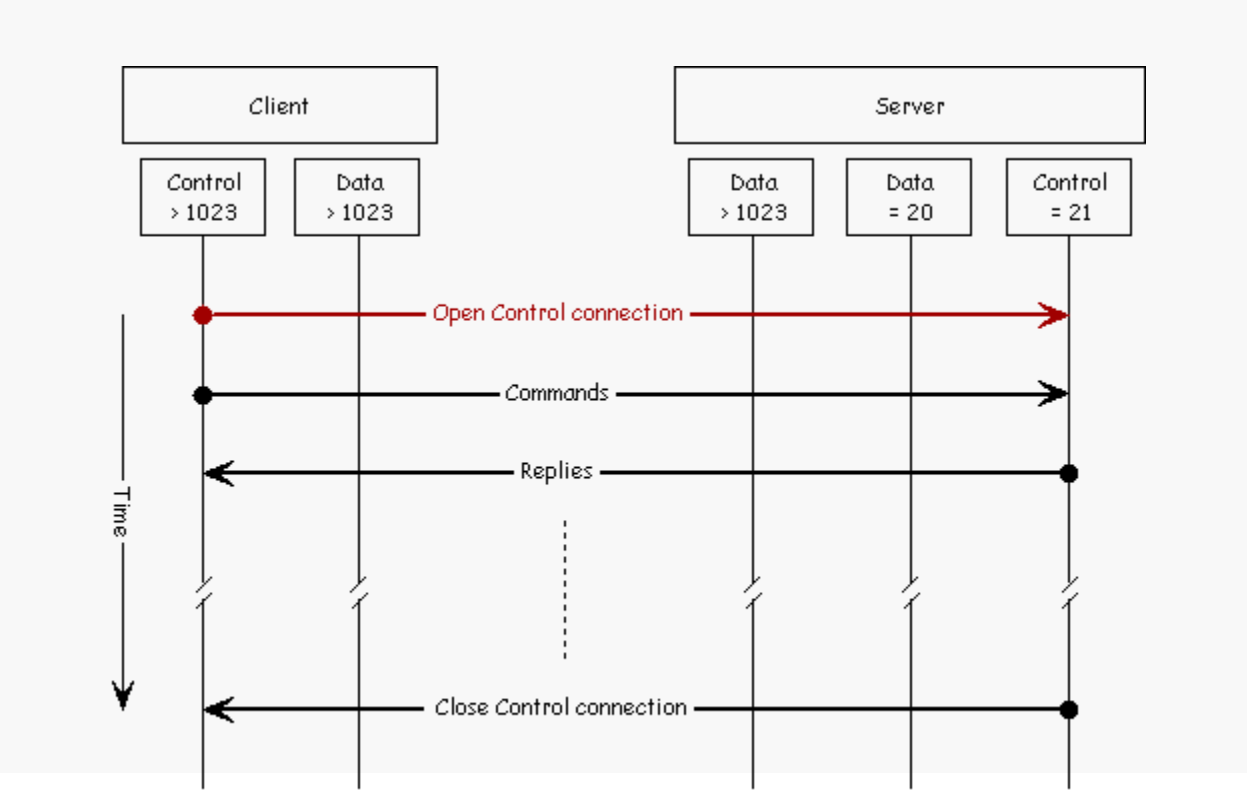
- The user interface communicates with the protocol interpreter, which is in charge of the control connection.
- The protocol interpreter, besides its function of responding to the control protocol, has also to manage the data connection. During the file transfer, the data management is performed by the DTPs.

4. Protocol Overview:

The FTP protocol uses a control connection (the primary connection) and a data connection (the secondary connection).

4.1 The Control connection:

- The control connection is the communication path between the USER-PI and SERVER-PI for the exchange of commands and replies. This connection follows the Telnet Protocol.
- When an FTP client wants to exchange files with an FTP server, the FTP client must first set up the control connection. The client makes a TCP connection from a random unprivileged port N (N > 1023) to the FTP server's well known command port 21 (the IANA assigned port number).

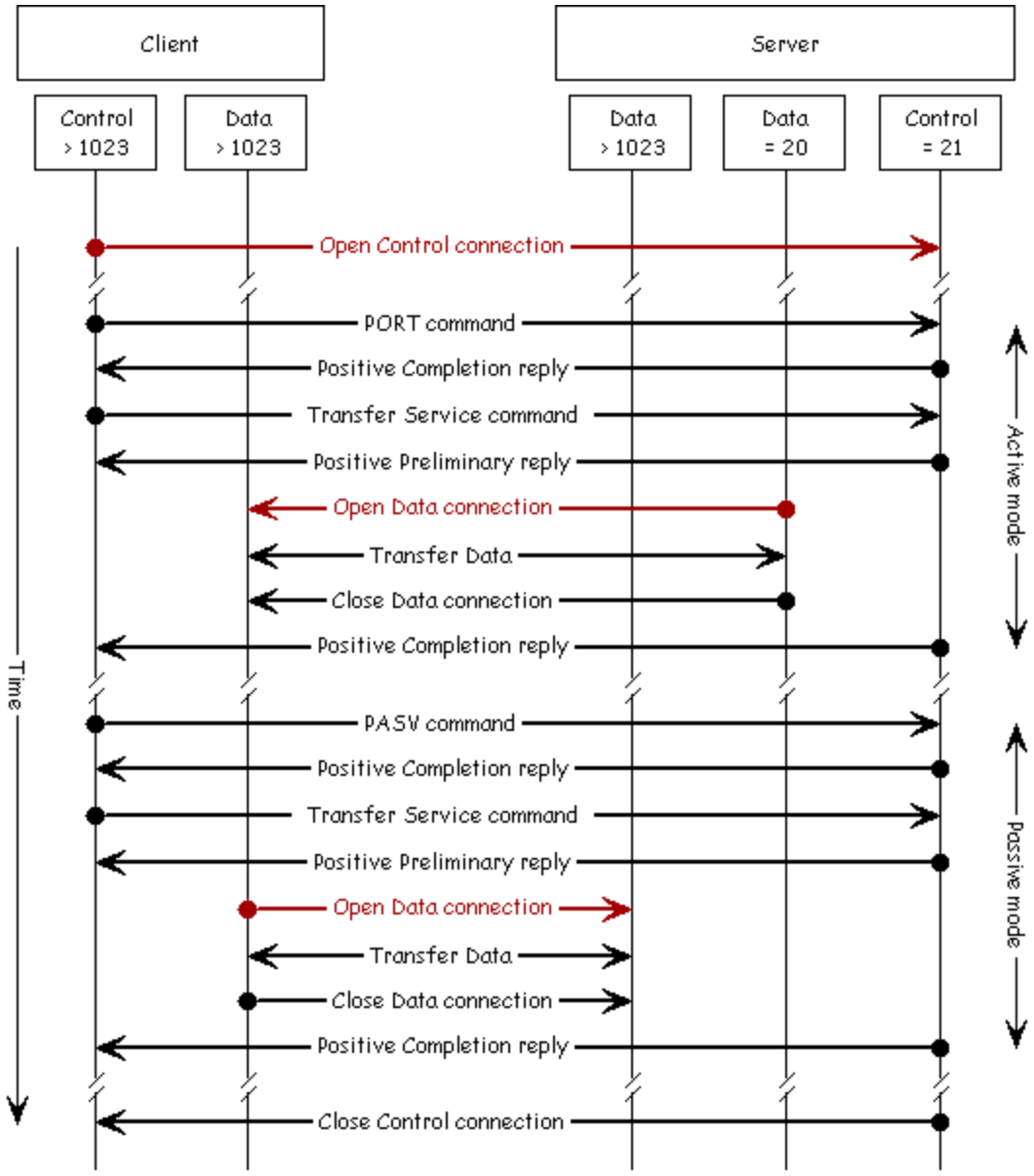


- The protocol requests the control connection to remain open while the data transfer is in progress.
- A data connection cannot exist without an open control connection.

- The data connection doesn't need to exist all of the time and there can be many data connections during the lifetime of a control connection.
- It is the responsibility of the user to request the closing of the control connection when finished using the FTP service. However, it is the server who takes the action to close the control connection.

4.2 The Data connection:

- The data connection is the communication path between the USER-DTP and SERVER-DTP for the exchange of the real data, being directory lists and files. Depending on the chosen FTP mode, the data connection is initiated from the server (active mode) or the client (passive mode).



5. Overview: FTP Basics Operations:

Goal:

Setup control and data connections, transfer data, closed connections.

Topology: A client H1 is connected to a FTP server S1 via Internet.

Steps:

1. H1 requests for a control connection with S1.
2. S1 requests for a data connection with H1.
3. S1 transfers data to H1.

4. When data transfer is done, S1 requests to close data connection and control connection.

H1: Control connection request

At H1, user types: ftp 1.1.2.1. It triggers H1 sending a Control Connection Request packet to S1. When S1 receives this request, it sends an Ack back to H1.Upon receiving Ack, H1 prints "20 FTP Server ready" to indicate that control connection is up.

H1: Get foo, PORT

- User types "get foo" at H1 to ask S1 to send a file foo.
- Then H1 sends a PORT command. Click PORT to see H1's port information: (IP: 1.1.1.1, port: 54705).

Note: Here FTP runs in active mode. It is server that initiates data connection. But server needs to know client's port number first. This is why H1 sends an unsolicited PORT command to S1.

S1: Data connection request

- Upon receiving PORT, S1 sends data_Conn to H1 (source port 20, destination port 54705)
- H1 responds with an Ack_data_Conn. Now data connection is up.
- S1 receives the Ack and sends a message to H1 (not shown in animation)
- H1 receives the message and prints “150 Opening BINARY....” to indicate that data transfer are starting.

S1 transfers foo to H1

- With data connection established, S1 starts to transmit foo data one packet (ftp_Data) at a time.
- When H1 receives a data packet, it responds an Ack_Data.
- When S1 receives Ack, it sends the next data packet.

S1: close data connection

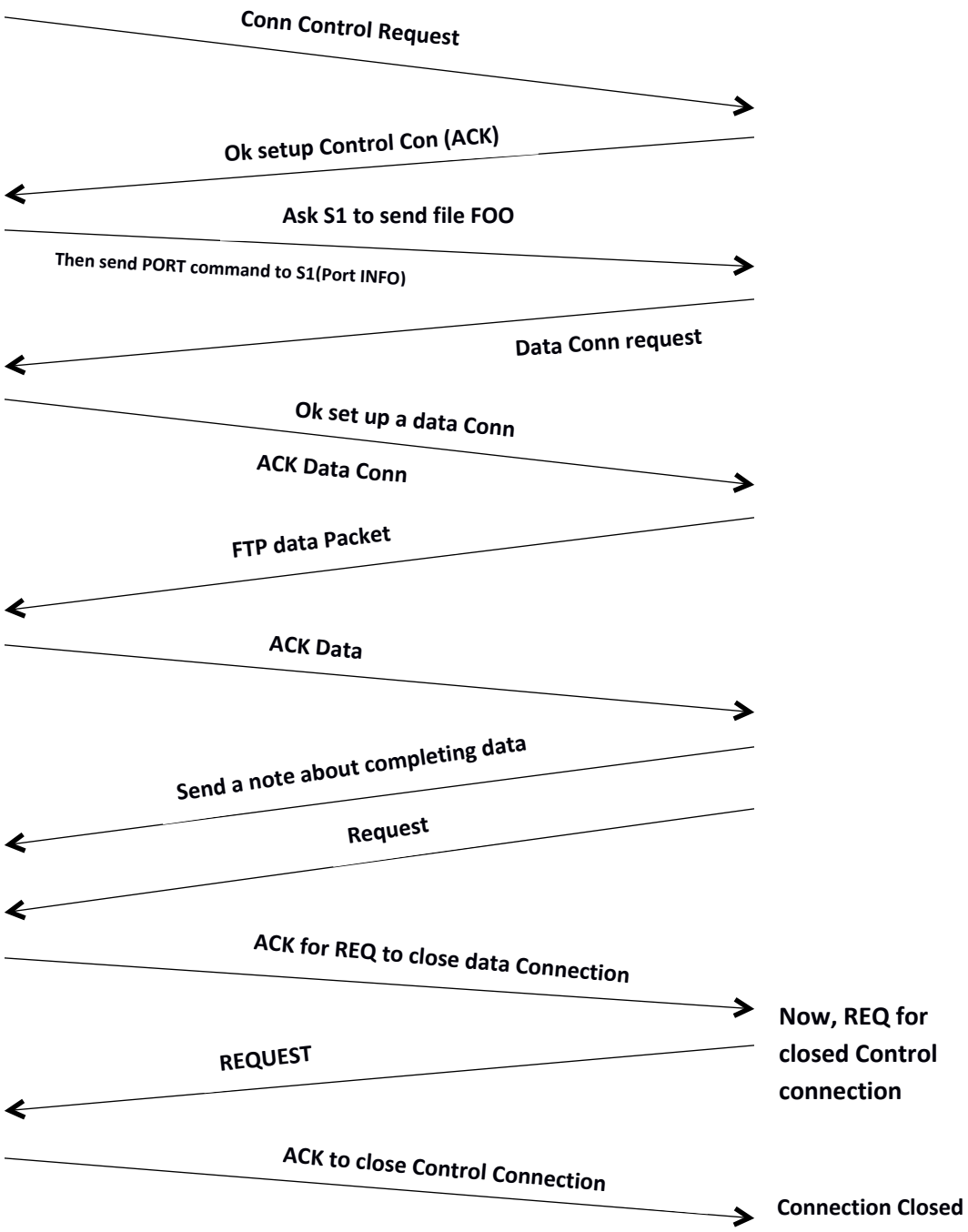
- After S1 has transmitted all data packets, it sends a message to H1
- When H1 receives this message, it print’s"226 Transfer complete" to indicate the file transfer is done.
- S1 closes the data connection and sends Close Data request to H1.
- H1 receives this request and sends an Ack to grant it. This closes FTP data connection.

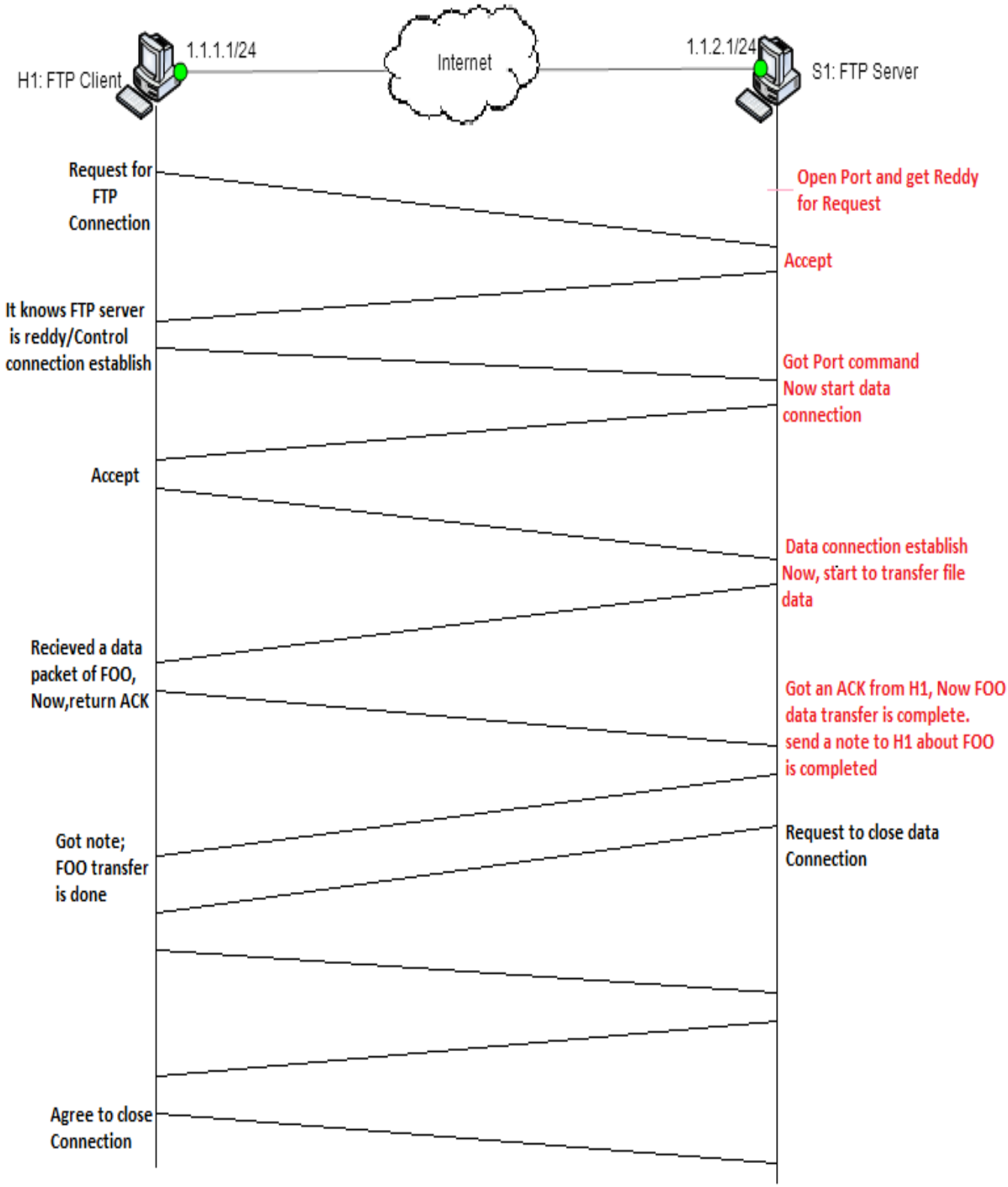
S1: close control connection

- User has no other FTP tasks to do and types "quit." It triggers a message to S1
- When S1 receives the quit message, it sends a goodbye message to H1
- H1 receives this message and prints “221 Goodbye" to tell user that FTP is exited.
- S1 sends Close_Ctrl to close control connection with H1.

- H1 receives the request and sends Ack_Close to confirm. Now FTP control connection is closed.

5.1 File Transfer Protocol (basic Operations):



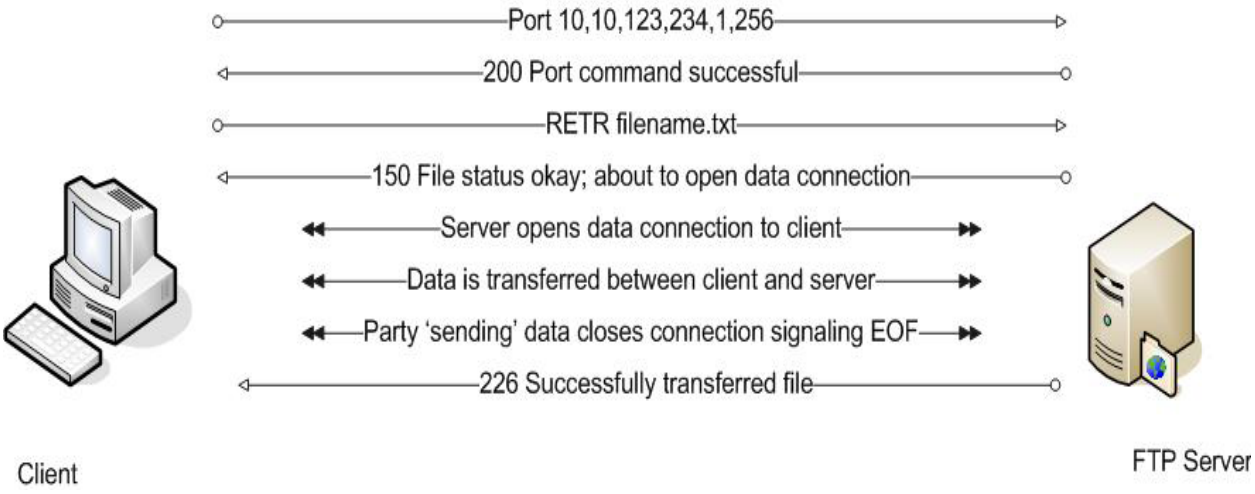


6. The FTP Protocol Modes

FTP may run in active or passive mode, which determines how the data connection is established.

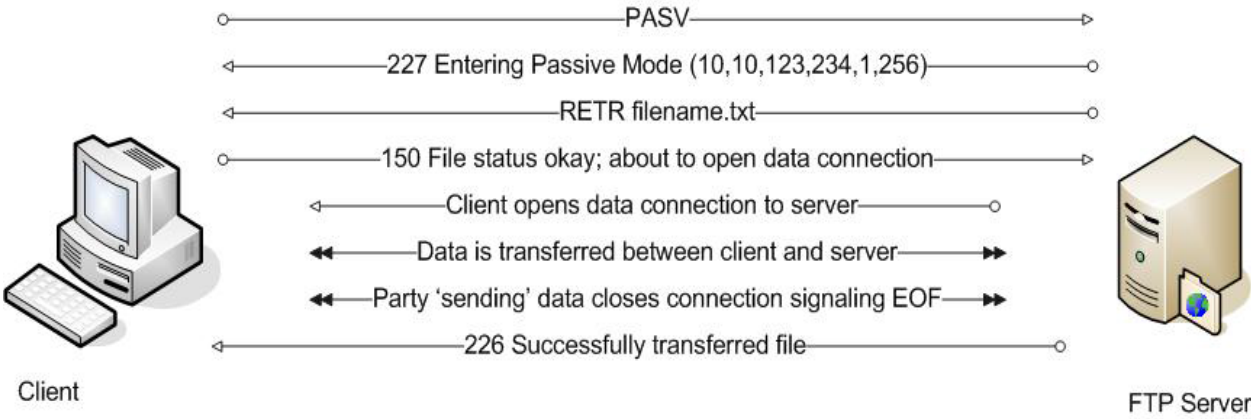
6.1 FTP Active Mode:

- In active mode, the client creates a TCP control connection. While data connection is initiated by FTP server.
- In active mode, the client sends a PORT command to the server. Basically this command tells the server to which host (IP address) and port number (unprivileged port > 1023)
- The server must connect back for the data connection. After accepting the Port command, the server will then establish the data connection from its local data port 20 (the IANA assigned port number) to the IP address and port number learned from the PORT command.



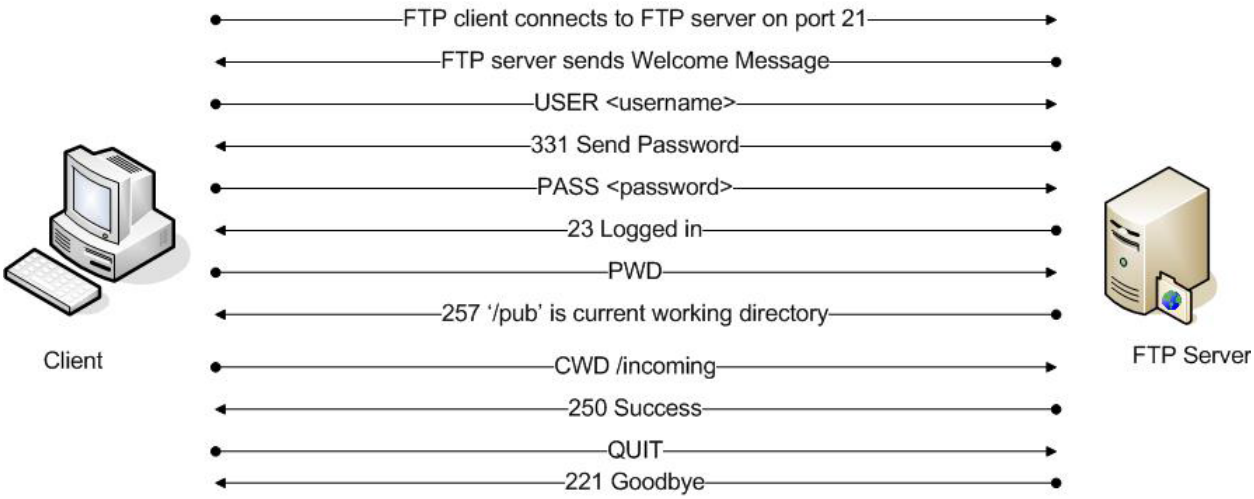
6.2 FTP Passive Mode:

- In Passive mode, the clients are responsible for initiating both the connection control connection as well as data connection.
- In passive mode, the client sends a PASV command to the server. Basically this command asks the server to "listen" on a data port (which is not its default data port 20) and to wait for a connection rather than to initiate one.
- If the server supports the passive mode, it will send a reply to this command including the host (IP address) and port number (unprivileged port > 1023) this server is listening on.
- The client will then establish the data connection from a local random unprivileged port (> 1023) to the IP address and port number learned from the PASV reply.



6.3 Login

FTP login utilizes a normal username and password scheme for granting access. The username is sent to the server using the USER command, and the password is sent using the PASS command. If the information provided by the client is accepted by the server, the server will send a greeting to the client and the session will commence. If the server supports it, users may log in without providing login credentials, but the same server may authorize only limited access for such sessions.



7. List of FTP commands:

These are the FTP **commands** that may be sent to an FTP server, these commands are standardized in RFC 959 by the IETF.

Note that most command-line FTP clients present their own set of commands to users. For example, GET is the common user command to download a file instead of the raw command RETR.

Command	RFC	Description
ABOR		Abort an active file transfer.
ACCT		Account information.
ADAT	RFC 2228	Authentication/Security Data
ALLO		Allocate sufficient disk space to receive a file.
APPE		Append.
AUTH	RFC 2228	Authentication/Security Mechanism
CCC	RFC 2228	Clear Command Channel
CDUP		Change to Parent Directory.
CONF	RFC 2228	Confidentiality Protection Command
CWD		Change working directory.
DELE		Delete file.
ENC	RFC 2228	Privacy Protected Channel
EPRT	RFC 2428	Specifies an extended address and port to which the server should connect.
EPSV	RFC 2428	Enter extended passive mode.
FEAT	RFC 2389	Get the feature list implemented by the server.
HELP		Returns usage documentation on a command if specified, else a general help document is returned.
LANG	RFC 2640	Language Negotiation
LIST		Returns information of a file or directory if specified, else information of the current working directory is returned.
LPRT	RFC 1639	Specifies a long address and port to which the server should connect.
LPSV	RFC 1639	Enter long passive mode.
MDTM	RFC 3659	Return the last-modified time of a specified file.
MIC	RFC 2228	Integrity Protected Command
MKD		Make directory.
MLSD	RFC 3659	Lists the contents of a directory if a directory is named.
MLST	RFC 3659	Provides data about exactly the object named on its command line, and no others.

MODE		Sets the transfer mode (Stream, Block, or Compressed).
NLST		Returns a list of file names in a specified directory.
NOOP		No operation (dummy packet; used mostly on keepalives).
OPTS	RFC 2389	Select options for a feature.
PASS		Authentication password.
PASV		Enter passive mode.
PBSZ	RFC 2228	Protection Buffer Size
PORT		Specifies an address and port to which the server should connect.
PROT	RFC 2228	Data Channel Protection Level.
PWD		Print working directory. Returns the current directory of the host.
QUIT		Disconnect.
REIN		Re initializes the connection.
REST		Restart transfer from the specified point.
RETR		Transfer a copy of the file
RMD		Remove a directory.
RNFR		Rename from.
RNTO		Rename to.
SITE		Sends site specific commands to remote server.
SIZE	RFC 3659	Return the size of a file.
SMNT		Mount file structure.
STAT		Returns the current status.
STOR		Accept the data and to store the data as a file at the server site
STOU		Store file uniquely.
STRU		Set file transfer structure.
SYST		Return system type.
TYPE		Sets the transfer mode (ASCII/Binary).
USER		Authentication username.
XCUP	RFC 775	Change to the parent of the current working directory
XMKD	RFC 775	Make a directory
XPWD	RFC 775	Print the current working directory
XRCP	RFC 743	
XRMD	RFC 775	Remove the directory
XRSQ	RFC 743	
XSEM	RFC 737	Send, mail if cannot
XSEN	RFC 737	Send to terminal

8. Advantages of FTP

- FTP is the fast and efficient way of transferring bulks of data across the internet.
- Allows transferring multiple files as well as directories.
- Many FTP clients have the ability to schedule transfers.
- No size limitation on single transfers (browsers only allow up to 2 GB)
- Many clients have scripting capabilities through command line
- Most clients have a synchronizing utility
- Faster transfers then HTTP
- It has an automatic backup .Whenever you edit your files in your local system you can update the same by copying it to the host system in your site. So in cases where your site has crashed and all the data is lost you have a copy of it in your own local system. It also works the other way round.
- FTP gives you control over transfer. That is, you can choose the mode in which the data is transferred over the network. The data can be transferred either in the ASCII mode (for text files) or in the Binary mode (for executable or compressed files).
- You can work with the directories on the remote systems, delete or rename the remote files while transferring data between 2 hosts.

9. Disadvantages of FTP

- FTP was not designed to be a secure protocol.
- FTP causes the following attacks during the transfer of data.
    1. Bounce Attacks
    2. Spoof Attacks
    3. Brute Force Attacks
    4. Packet Sniffing
    5. User name protection
    6. Port sealing

- Encryption of data is not done in FTP.
- Usernames, passwords and files are sent in clear text.
- Servers can be spoofed to send data to a random port on an unintended computer
- Filtering active FTP connections is difficult on your local machine (passive is preferred)

The World Wide Web (WWW) is a store of data connected together from focuses everywhere throughout the world. The WWW has a remarkable blend of adaptability, convenience, also easy to understand characteristics that recognize it from different administrations gave by the Internet. The WWW undertaking was launched by CERN (European Laboratory for Particle Physics) to make a framework to handle circulated assets vital for logical exploration.

The World Wide Web (WWW, W3) is a data arrangement of interlinked hypertext archives that are gotten to by means of the Internet. It has likewise generally ended up referred to just as the Web. Individual record pages on the World Wide Web are called site pages and are gotten to with a product application running on the client's PC, ordinarily called a web program. Website pages may contain content, pictures, features, and other mixed media parts, and in addition web route gimmicks comprising of hyperlinks.

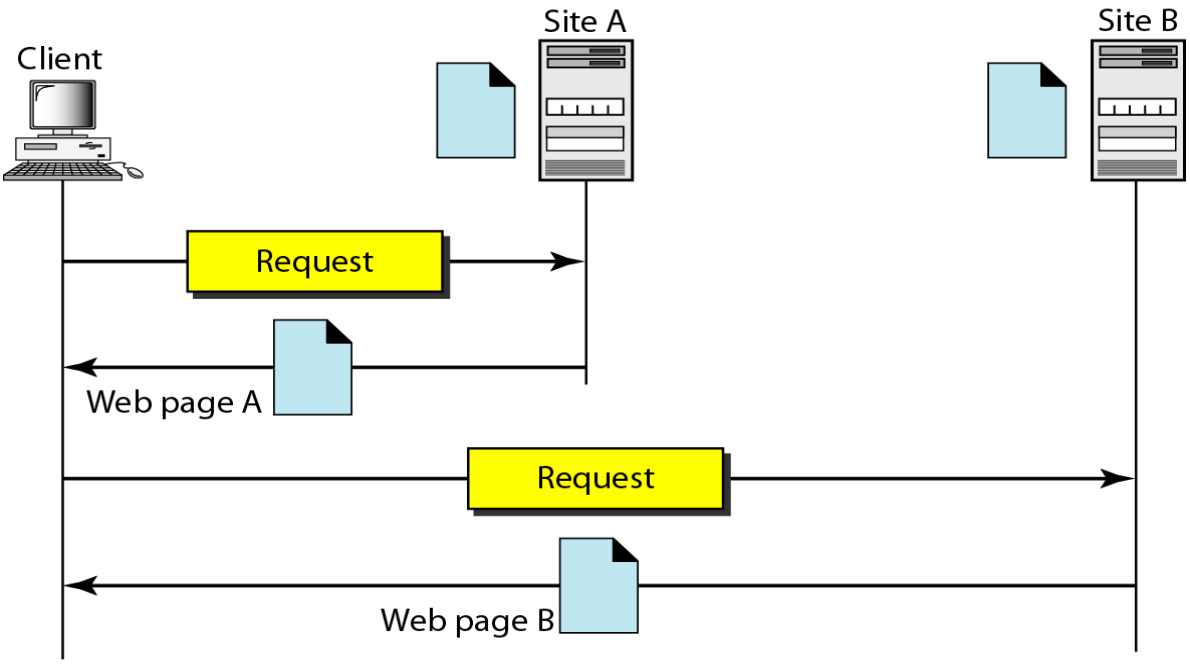
History

Tim Berners-Lee, a British PC researcher and previous CERN representative, is viewed as the innovator of the Web. On 12 March 1989, Berners-Lee composed a proposition for what would in the long run turn into the World Wide Web.

Architecture

The WWW today is a dispersed client server administration, in which a customer utilizing a program

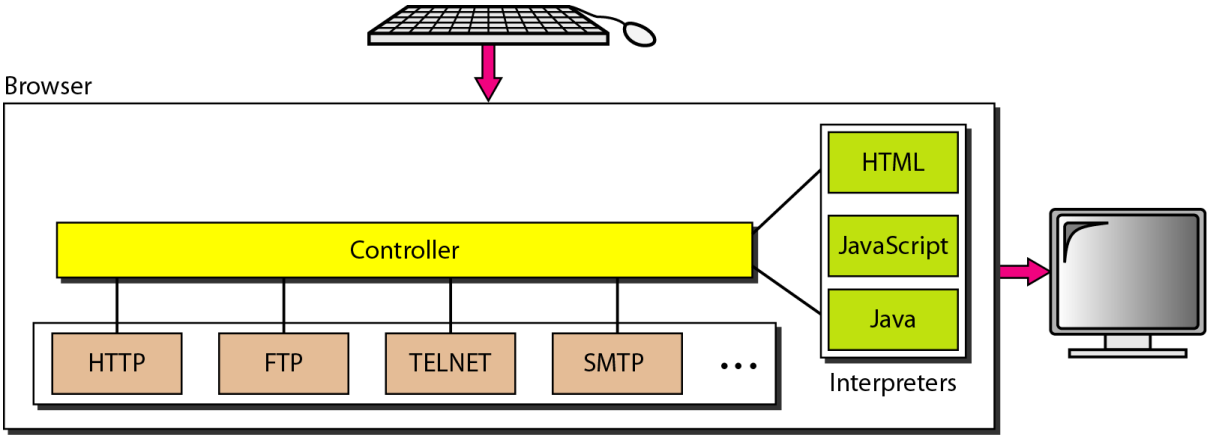
can get to an administration utilizing a server. Notwithstanding, the administration gave is conveyed over numerous areas called sites, as indicated in Figure.



The client needs to see some information that it knows belongs to site A. It sends a request through its browser, a program that is designed to fetch Web documents. The request, among other information, includes the address of the site and the Web page, called the URL. 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)

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 protocol like FTP The interpreter can be HTML, Java,or JavaScript, depending on the type of document.

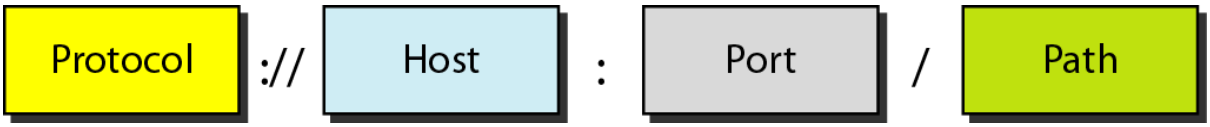


Server

The Webpage is put away at the server. Each time a customer solicitation arrives , the relating record is sent to the client. To enhance efficiency, servers regularly store asked records in a store in memory; memory is speedier to get to than disk. A server can likewise become more productive through multithreading or multiprocessing. In this case, a server can answer more than one appeal at once.

Uniform Resource Locator

An Uniform Resource Locator (abridged URL; otherwise called a web address, especially when utilized with HTTP) is a particular character string that constitutes a reference to an asset. Most web programs show the URL of a website page over the page in an address bar.



The URL defines four things : protocol , host computer , port and path as shown in above figure.

Protocol

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.

Host

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.

Port

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

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

Function of WWW

- The WWW works by establishing hypertext/hypermedia links between documents anywhere on the network.
- A document might include many links to other documents held on many different servers.
- Selecting any one of those links will take you to the related document wherever it is.
  e.g. the references at the end of a paper might have hypertext links to the actual documents held elsewhere.

WWW Hyperlinks

Hyperlinks can link a part of a hypermedia document to

- another part of the same document file.
- another document file on the same server computer.
- another document file on a server computer located elsewhere in the world.

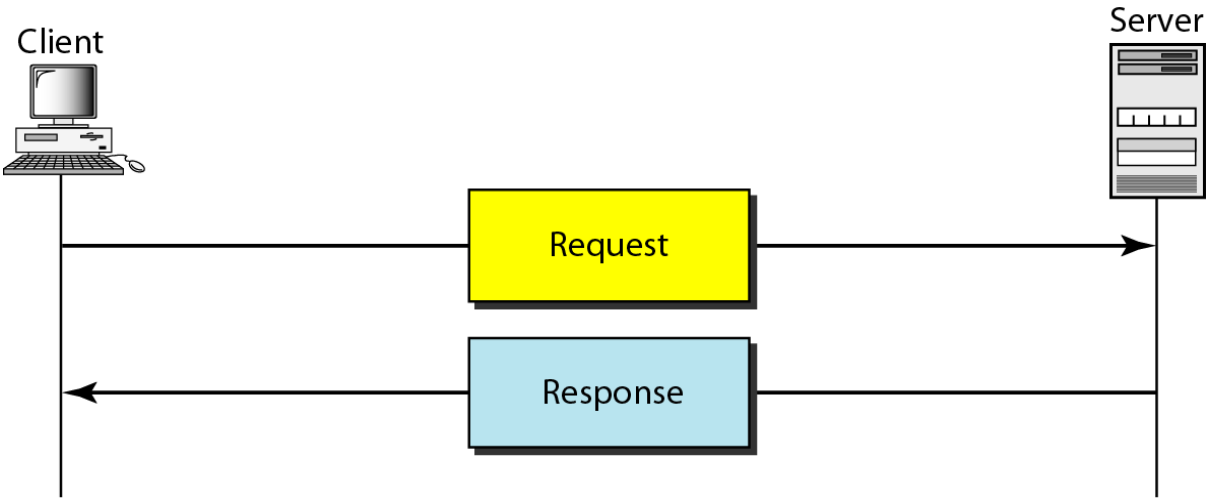
HTTP

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

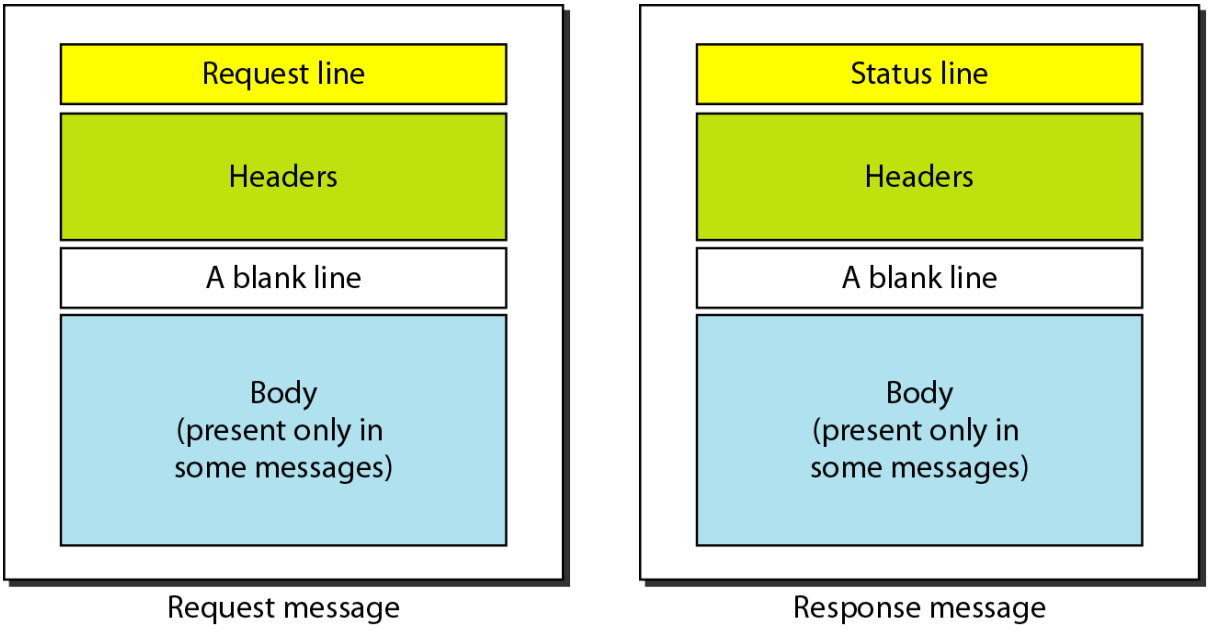
HTTP Trasanction

HTTP itself is a stateless protocol. The client initializes the transaction by sending a request message. The server replies by sending a response.



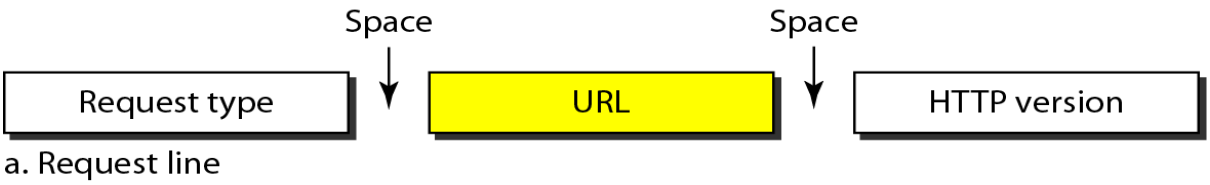
Messages

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.



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.



a. Request line

b. Status line

HTTP Methods

HTTP allows an open-ended set of methods to be used to indicate the purpose of a request. The three most often used methods are GET, HEAD, and POST.

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

The GET Method

The GET method is used to ask for a specific document - when you click on a hyperlink, GET is being used. GET should probably be used when a URL access will not change the state of a database (by, for example, adding or deleting information) and POST should be used when an access will cause a change. The semantics of the GET method changes to a "conditional GET" if the request message includes an "If-Modified-Since:" header field.

The HEAD Method is used to ask only for information about a document, not for the document itself. HEAD is much faster than GET, as a much smaller amount of data is transferred. It's often used by clients who use caching, to see if the document has changed since it was last accessed. If it was not, then the local copy can be reused, otherwise the updated version must be retrieved

with a GET. The meta-information contained in the HTTP headers in response to a HEAD request should be identical to the information sent in response to a GET request.

The POST Method

The POST method is used to transfer data from the client to the server; it's designed to allow a uniform method to cover functions like: annotation of existing resources; posting a message to a bulletin board, newsgroup, mailing list, or similar group of articles; providing a block of data (usually a form) to a data-handling process; extending a database through an append operation.

The Status Code

The status code field used in the response message is similar to those in the FTP and the SMTP protocols. It consists of three digits. Whereas the codes in the 100 range are only informational, the codes in the 200 range indicate a successful request.

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.

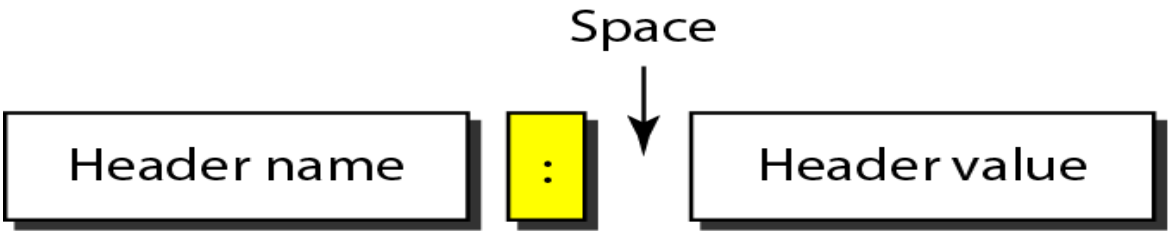
Status Phase

This field is used in the response message. It explains the status code in text form.

HEADER

The header exchanges additional information between the client and the server. 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.

A header line belongs to one of four categories: general header, request header, response header, and entity header. A request message can contain only general, request, and entity headers. A response message, on the other hand, can contain only general, response, and entity headers.



General Header

The general header gives general information about the message and can be present in both a request and a response.

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

Request Header

The request header can be present only in a request message. It specifies the client's configuration and the client's preferred document format.

Header	*Description*
Accept	Shows the medium format the client can accept
Accept-charset	Shows the character set the client can handle
Accept-encoding	Shows the encoding scheme the client can handle
Accept-language	Shows the language the client can accept
Authorization	Shows what permissions the client has
From	Shows the e-mail address of the user
Host	Shows the host and port number of the server
If-modified-since	Sends the document if newer than specified date
If-match	Sends the document only if it matches given tag
If-non-match	Sends the document only if it does not match given tag
If-range	Sends only the portion of the document that is missing
If-unmodified-since	Sends the document if not changed since specified date
Referrer	Specifies the URL of the linked document
User-agent	Identifies the client program

Response Header

The response header can be present only in a response message.It specifies the server's configuration and special information about the request.

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

Entity Header

The entity header gives information about the body of the document. Although it is mostly present in response messages, some request messages, such as POST or PUT methods, that contain a body also use this type of header.

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

Body

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

Search Engines

A **web search engine** is a software system that is designed to search for information on the World Wide Web. The search results are generally presented in a line of results often referred to as search engine results pages (SERPs).

The information may be a mix of web pages, images, and other types of files. Some search engines also mine data available in databases or open directories.

Unlike web directories, which are maintained only by human editors, search engines also maintain real-time information by running an algorithm on a **web crawler**.

A **Web crawler** is an Internet bot that systematically browses the World Wide Web, typically for the purpose of Web indexing. A Web crawler may also be called a **Web spider**, an **ant**, an **automatic indexer**, or (in the FOAF software context) a **Web scutter**.

Web search engines and some other sites use Web crawling or spidering software to update their web content or indexes of others sites' web content. Web crawlers can copy all the pages they visit for later processing by a search engine that indexes the downloaded pages so that users can search them much more quickly