



Application Layer

Dr. Firoz Ahmed

Associate Professor

Department of ICE, RU

OutLine

- Client-Server Model
- Domain Name System (DNS)
- Electronic mail (SMTP) and File Transfer (FTP)
- HTTP and WWW

Client-Server Model

- Two remote application process can communicate in mainly two different fashions
 - Peer-to-peer
 - Both remote processes are at same level and exchange data using some shared resource
 - Client-Server
 - One the remote process acts as Client and requests some resource from another application process acting as Server

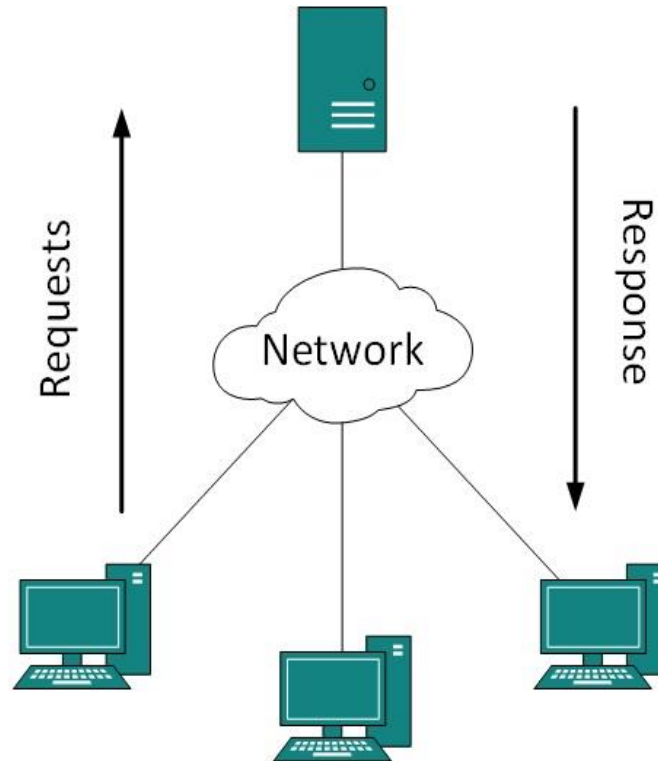
Client-Server Model

- A client-server system consists of a server system and (one or more) client subsystems
- In the client/server model
 - The device requesting the information is called a client and the device responding to the request is called a server
- Client and server processes are considered to be in the Application layer
- The client begins the exchange by requesting data from the server, which responds by sending one or more streams of data to the client
- Application layer protocols describe the format of the requests and responses between clients and servers

Client-Server Model

- In client-server model, any process can act as Server or Client
- This is not the machine or size of the machine or its computing power which makes it server
 - It is the feature of serving request that makes it server
- A system can act as Server and Client simultaneously
- That is, one process is acting as Server and another is acting as a client
- This may also happen that both client and server processes reside on the same machine

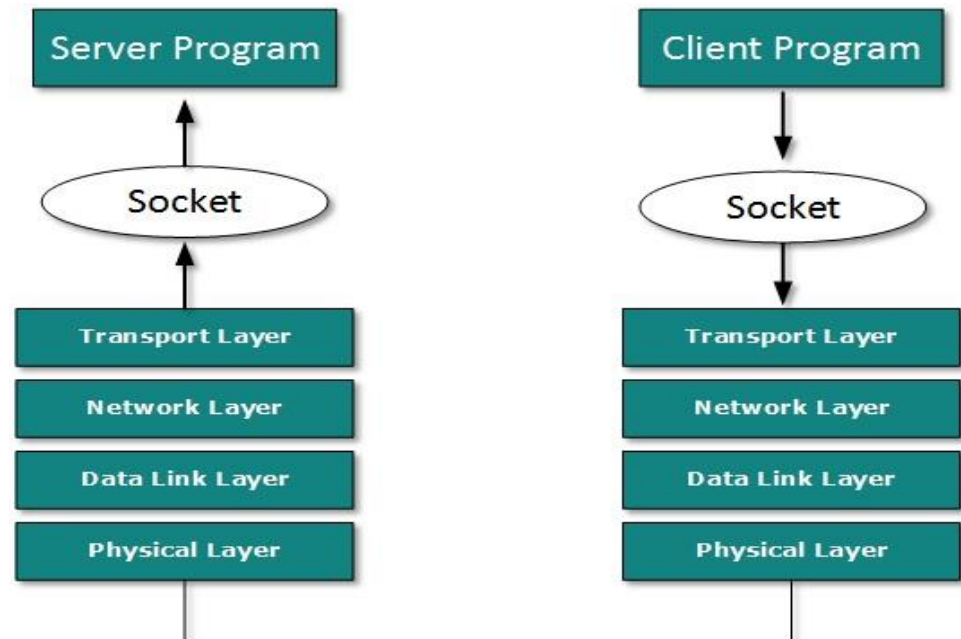
Client-Server Model



- Two processes in client-server model can interact in various ways
 - Sockets
 - Remote Procedure Calls (RPC)

Sockets

- In this paradigm, the process acting as Server opens a socket using a well-known port and waits until some client request comes
- The second process acting as Client also opens a socket but instead of waiting for an incoming request, client process 'requests first'
- When the request is reached to server, it is served. It can either be an information sharing or resource request



Remote Procedure Calls (RPC)

- This is a mechanism where one process interacts with another by means of procedure calls
- One process (client) calls the procedure lying on remote host
- The process on remote host is said to be Server

Domain Name System (DNS)

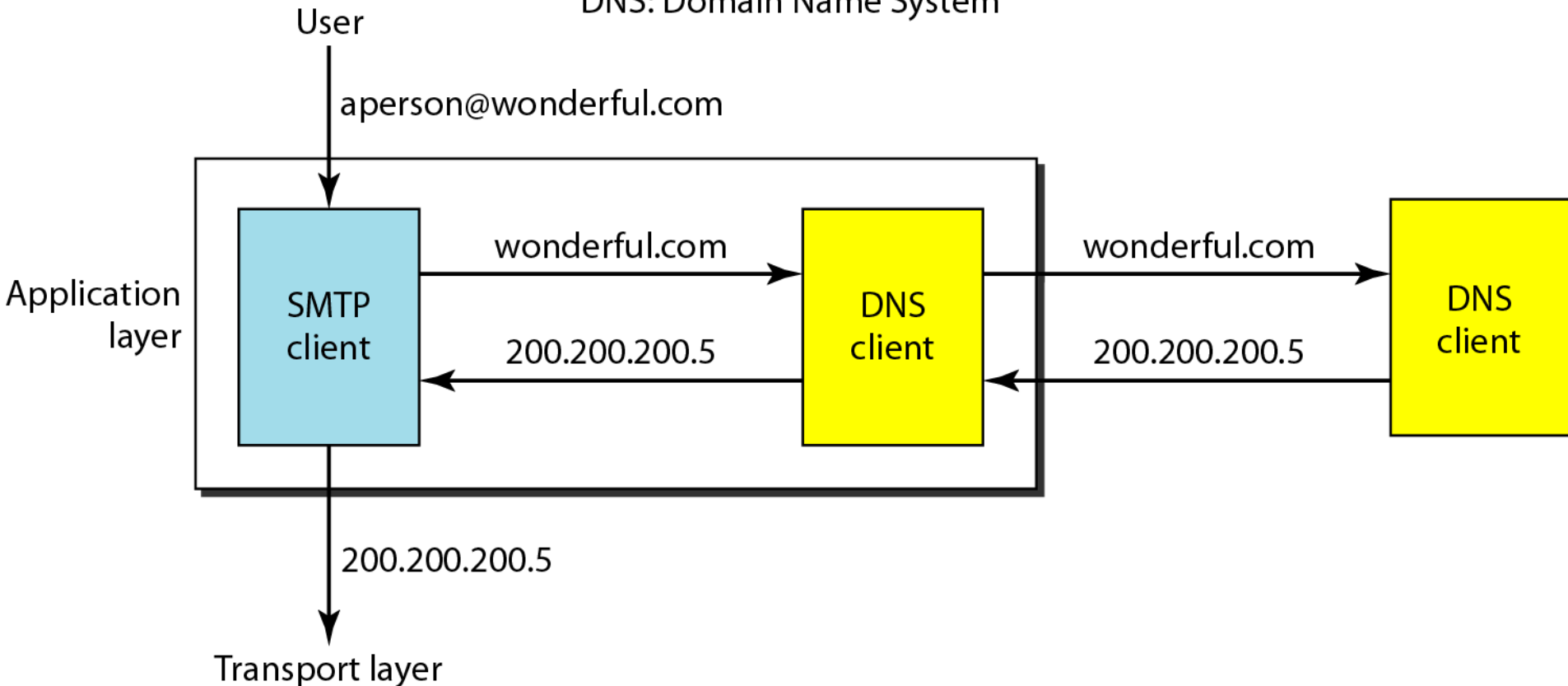
- The Domain Name System (DNS) is a supporting program that is used by other programs such as e-mail
- A DNS client/server program can support an e-mail program to find the IP address of an e-mail recipient
- A user of an e-mail program may know the e-mail address of the recipient
- However, the IP protocol 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

Domain Name System (DNS)

- To identify an entity, TCP/IP protocols use the IP address
- 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

Domain Name System (DNS)

SMTP: Simple Mail Transfer Protocol (e-mail)
DNS: Domain Name System



Example of using the DNS service

Outline

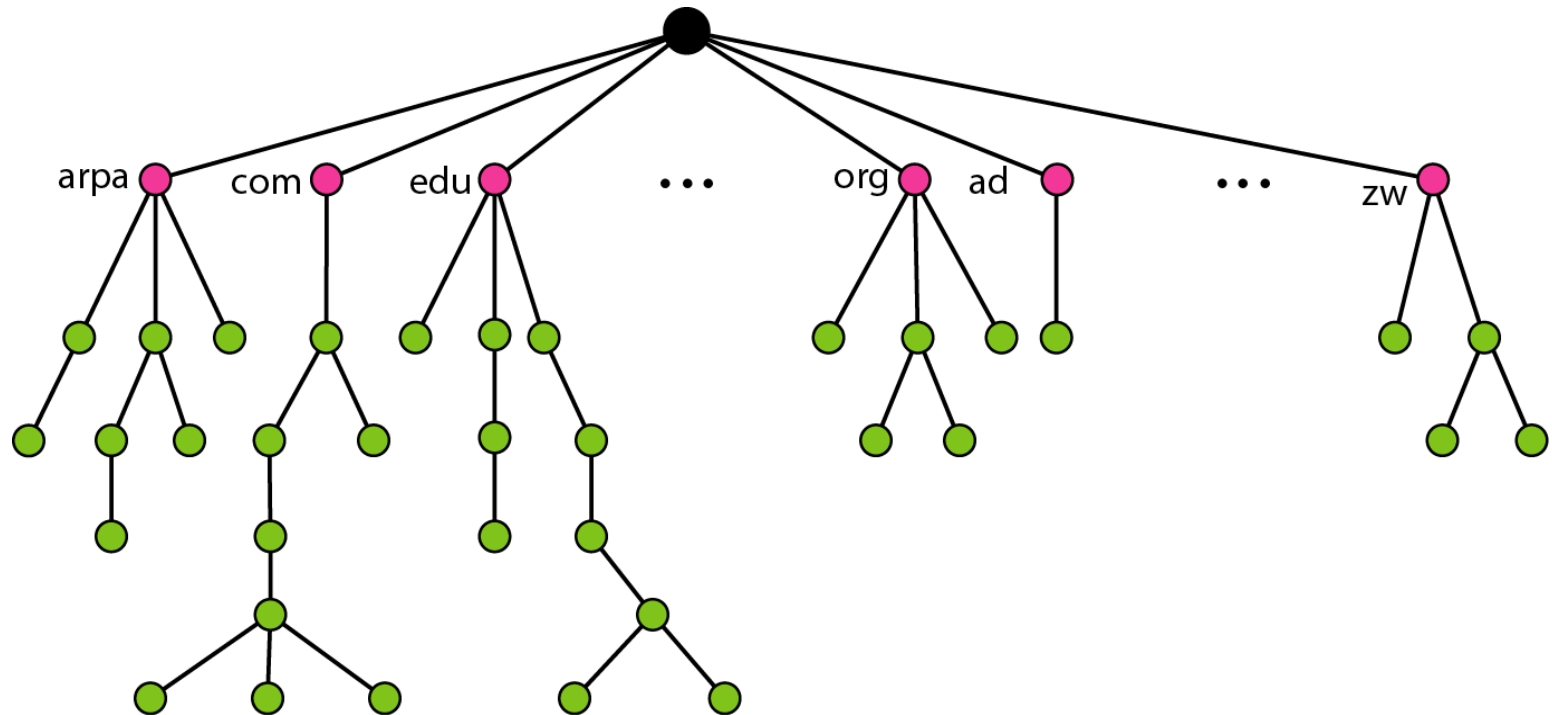
- Name Space
- Domain Name Space
- Distribution of Name Space
- DNS in the Internet
- Resolution
- DNS Messages
- Types of Records
- Registrars
- Dynamic Domain Name System (DDNS)
- Encapsulation

Name Space

- The names must be unique because the addresses are unique
- A name space that maps each address to a unique name can be organized in two ways
 - Flat
 - Hierarchical

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



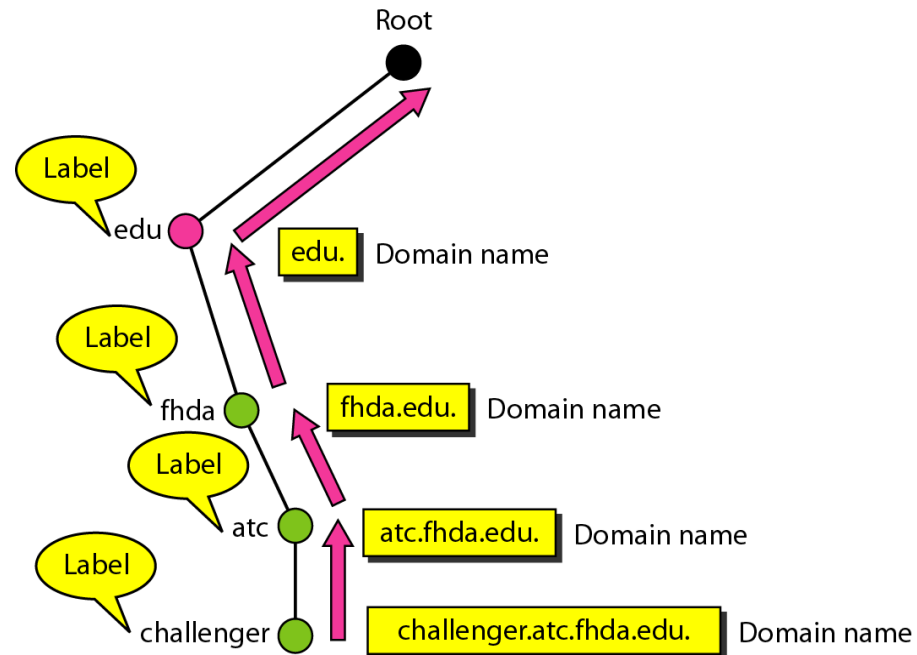
Domain Name Space

■ Label

- Each node in the tree has a label, which is a string with a maximum of 63 characters
- Different labels guarantees the uniqueness of the domain names

■ 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



Domain Name Space

■ Fully Qualified Domain Name (FQDN)

- If a label is terminated by a null string, it is called FQDN
- An FQDN is a domain name that contains the full name of a host
challenger.atc.tbda.edu.

■ Partially Qualified Domain Name (PQDN)

- If a label is not terminated by a null string, it is called PQDN
- A PQDN starts from a node, but it does not reach the root

FQDN

challenger.atc.fhda.edu.
cs.hmme.com.
www.funny.int.

PQDN

challenger.atc.fhda.edu
cs.hmme
www

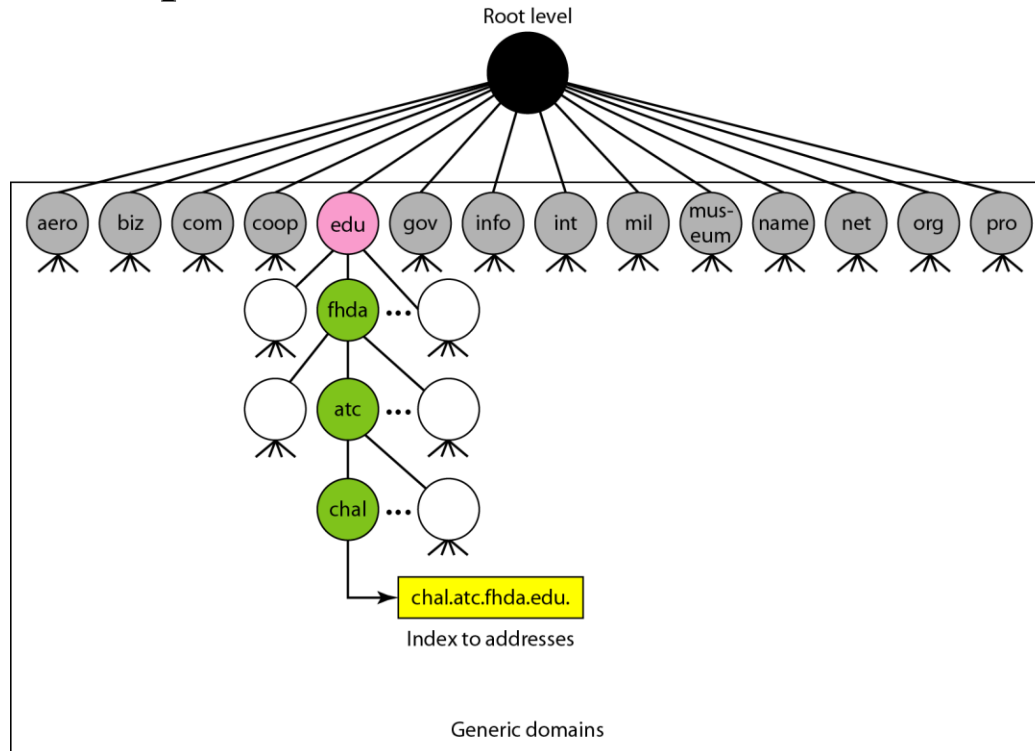
DNS in the Internet

- DNS is a protocol that can be used in different platforms
- In the Internet, the domain name space is divided into three different sections
 - Generic domains
 - Country domains
 - The inverse domain

DNS in the Internet

■ Generic Domains

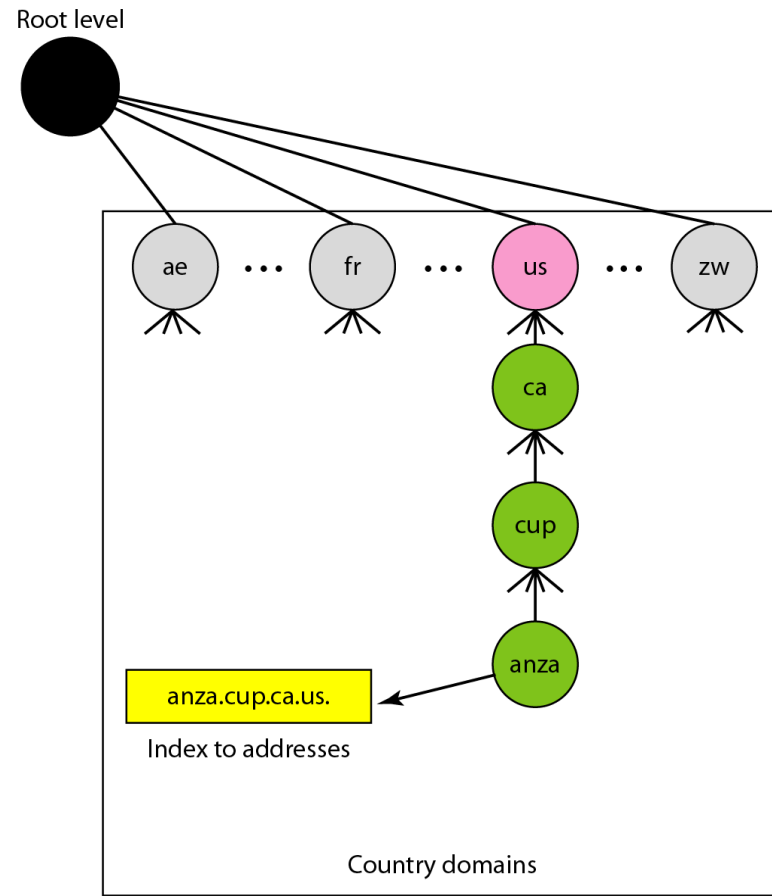
- The generic domains define registered hosts according to their generic behavior
- Each node in the tree defines a domain, which is an index to the domain name space database



DNS in the Internet

■ Country Domains

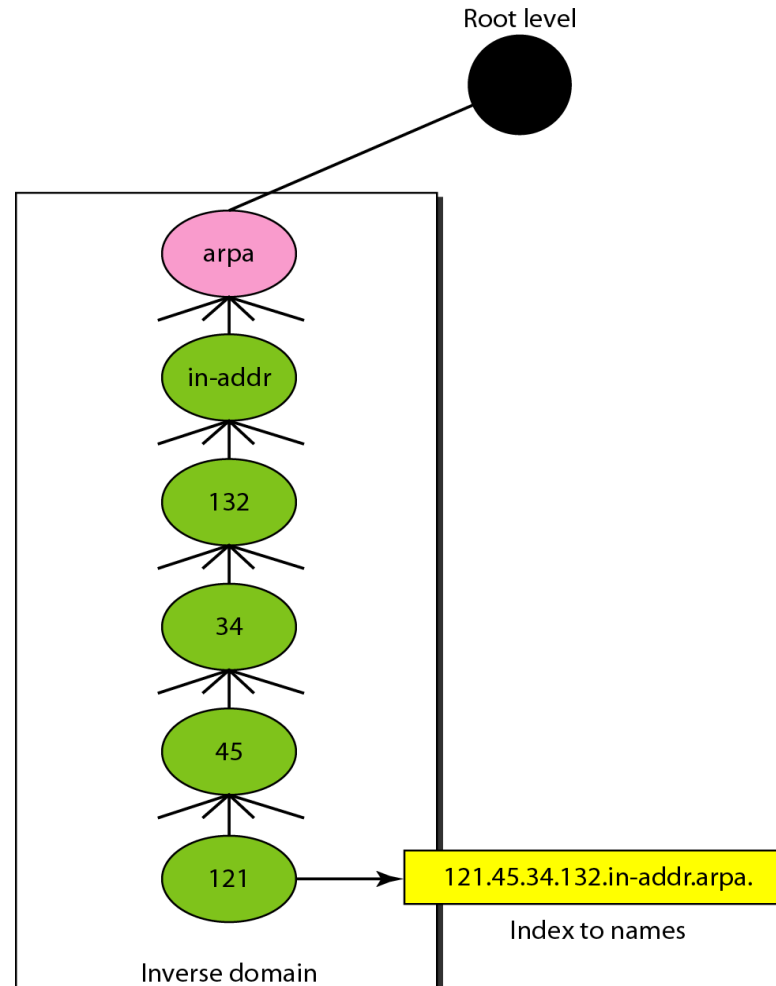
- The country domains section uses two-character country abbreviations (e.g., us for United States)



DNS in the Internet

■ Inverse Domain

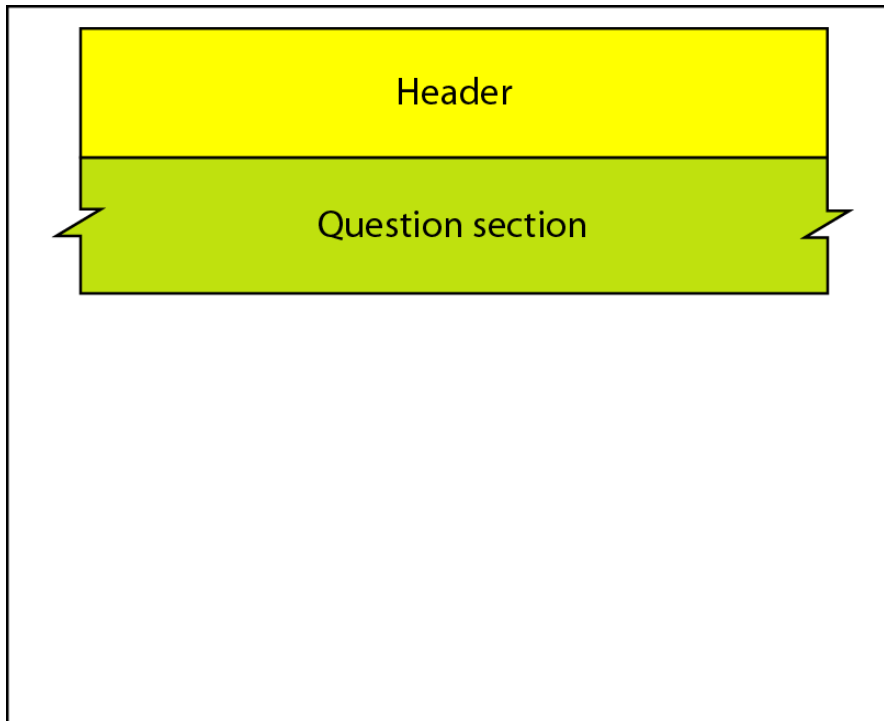
- The inverse domain is used to map an address to a name



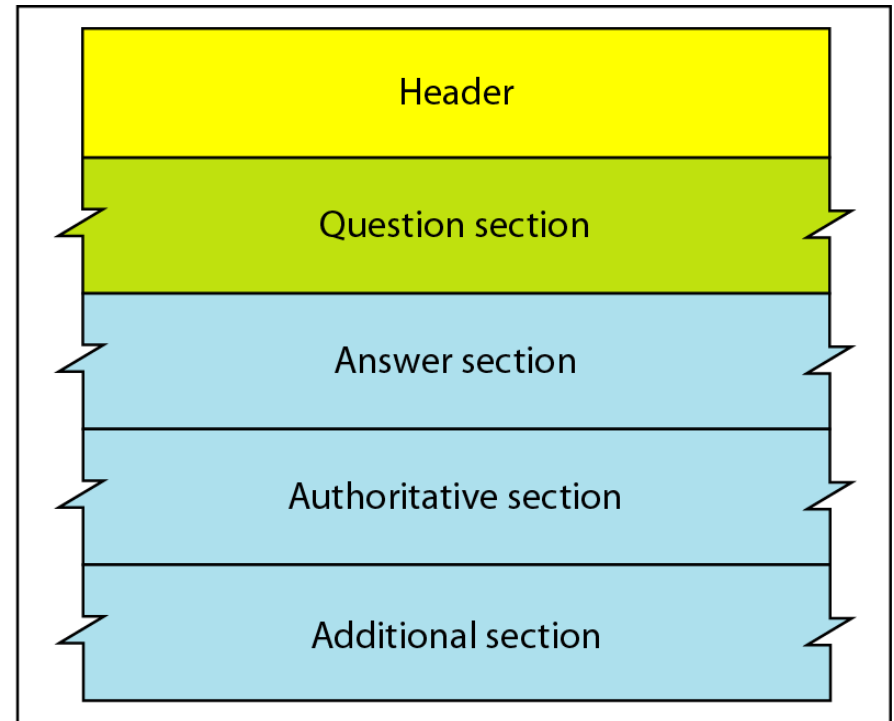
DNS MESSAGES

- DNS has two types of messages
 - Query
 - Response
- Both types have the same format
- The query message consists of a header and question records;
- the response message consists of a header, question records, answer records, authoritative records, and additional records

DNS MESSAGES



a. Query



b. Response

DNS MESSAGES

■ Header

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

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

DNS MESSAGES

■ Identification

- The identification subfield is used by the client to match the response with the query
- The client uses a different identification number each time it sends a query
- The server duplicates this number in the corresponding response

■ Flags

- The flags subfield is a collection of subfields that define the type of the message, answer requested, desired resolution and so on

■ Number of question records

- It contains the number of queries in the question section of the message

DNS MESSAGES

■ Number of answer records

- It contains the number of answer records in the answer section of the response message

■ Number of authoritative records

- It contains the number of authoritative records in the authoritative section of a response message

■ Number of additional records

- It contains The number additional records in the additional section of a response message

DNS MESSAGES

■ Question Section

- It consist of one or more question records
- It is present on both query and response messages

■ Answer Section

- It consist of one or more resource records
- It is present only on response messages
- This section includes the answer from the server to the client

■ Authoritative Section

- It gives information about one or more authoritative server for the query

■ Additional information Section

- It provides additional information that may help the client

Dynamic Domain Name System (DDNS)

- When the DNS was designed, no one predicted that there would be so many addresses changes
- When there is a changes, the change must be made to the DNS master file
- The DNS master file must be updated dynamically
- Therefore DDNS was devised to respond to this need
- To provide security and prevent unauthorized changes in the DNS records, DDNS can use an authentication mechanism

Electronic mail

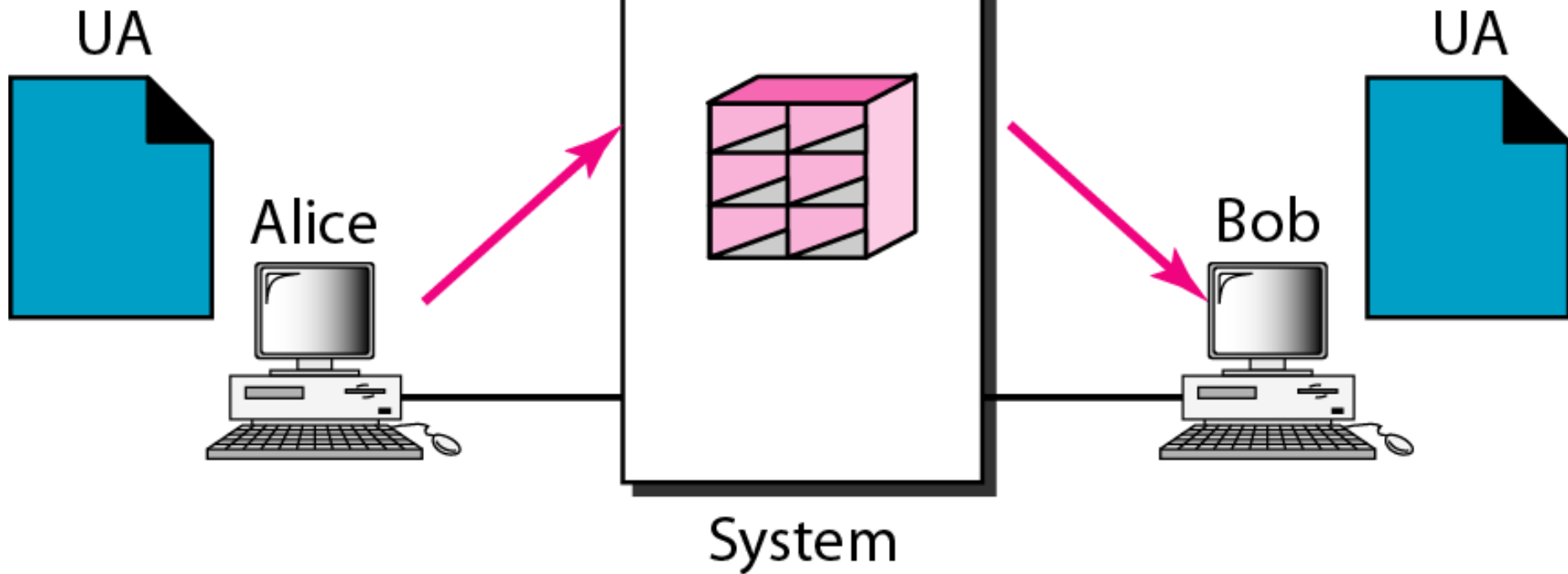
- One of the most popular Internet services is electronic mail (e-mail)
- At the beginning of the Internet era, the messages sent by electronic mail were short and consisted of text only
- Today, electronic mail is much more complex
- It allows a message to include text, audio, and video
- It also allows one message to be sent to one or more recipients

Electronic mail

- The general architecture of an e-mail system including the three main components
 - User agent
 - Message transfer agent
 - Message access agent

Electronic mail (First Scenario)

UA: user agent



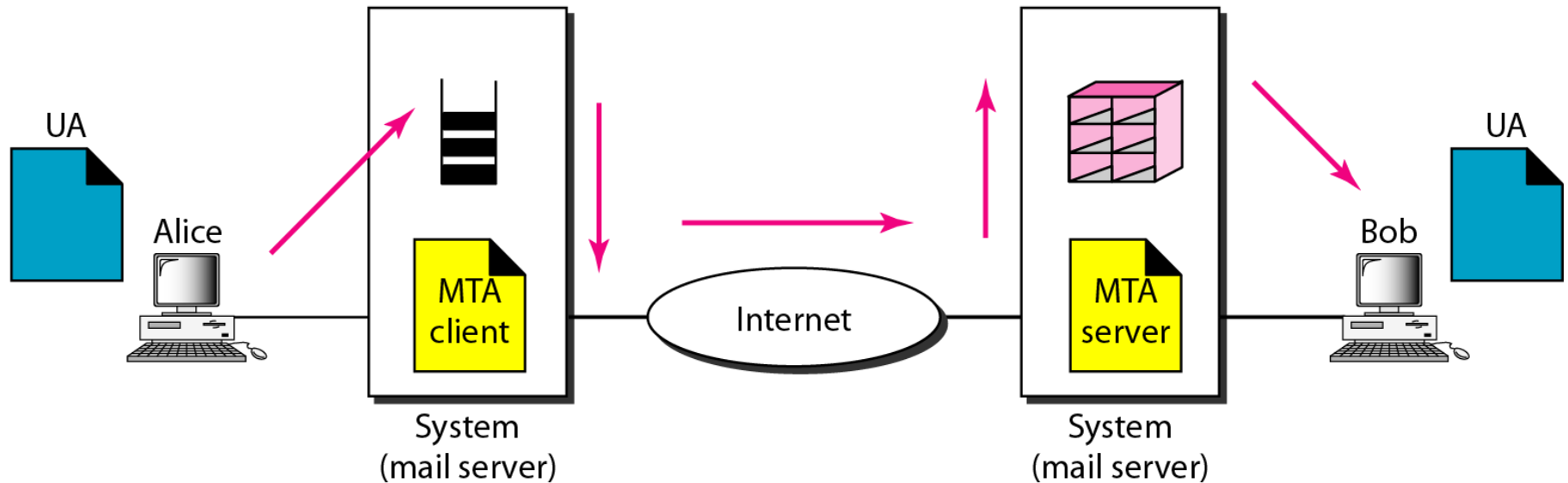
Electronic mail (First Scenario)

- The sender and the receiver of the e-mail are users 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
- Only the owner of the mailbox has access to it
- When Alice needs to send a message to Bob, Alice runs a user agent (*UA*) program to prepare the message and store it in Bob's mailbox
- Bob can retrieve and read the contents of his mailbox, using a user agent

Electronic mail (Second Scenario)

UA: user agent

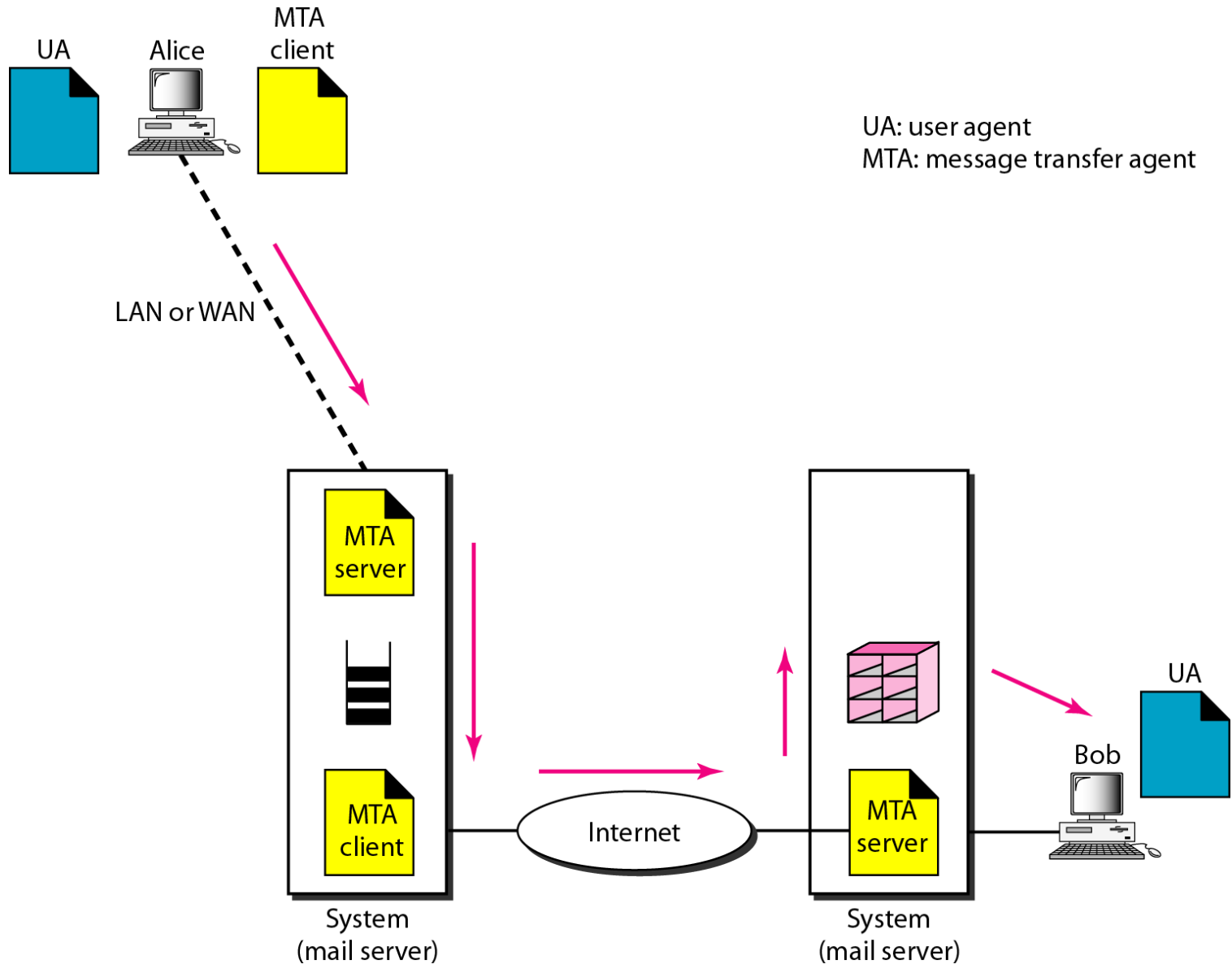
MTA: message transfer agent



Electronic mail (Second Scenario)

- The sender and the receiver of the e-mail are users on two different systems
- The message needs to be sent over the Internet
- We need user agents (UAs) and message transfer agents (MTAs)
- Alice needs to use a user agent program to send her message to the system at her own site
- Bob also needs a user agent program to retrieve messages stored in the mailbox of the system at his site
- The message needs to be sent through the Internet from Alice's site to Bob's site

Electronic mail (Third Scenario)



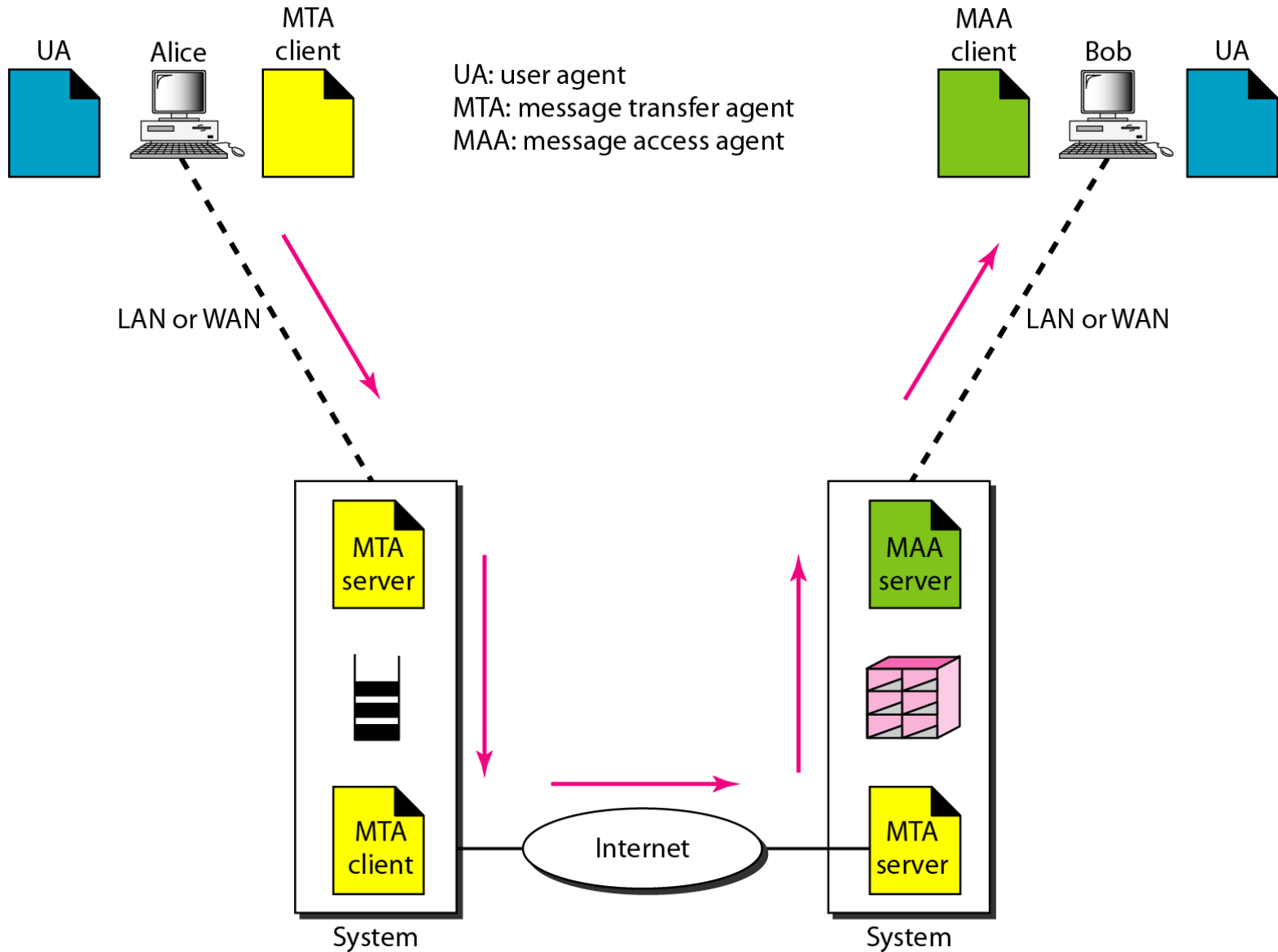
Electronic mail (Third Scenario)

- Bob 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 or LAN
- Alice still needs a user agent to prepare her message
- She then needs to send the message through the LAN or WAN

Electronic mail (Third Scenario)

- Whenever Alice has a message to send, she calls the user agent which, in turn, calls the MTA client
- The MTA client establishes a connection with the MTA server on the system
- The system at Alice's site queues all messages received
- It then uses an MTA client to send the messages to the system at Bob's site
- The system receives the message and stores it in Bob's mailbox

Electronic mail (Fourth Scenario)



Electronic mail (Fourth 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
- 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 and requests the transfer of the messages

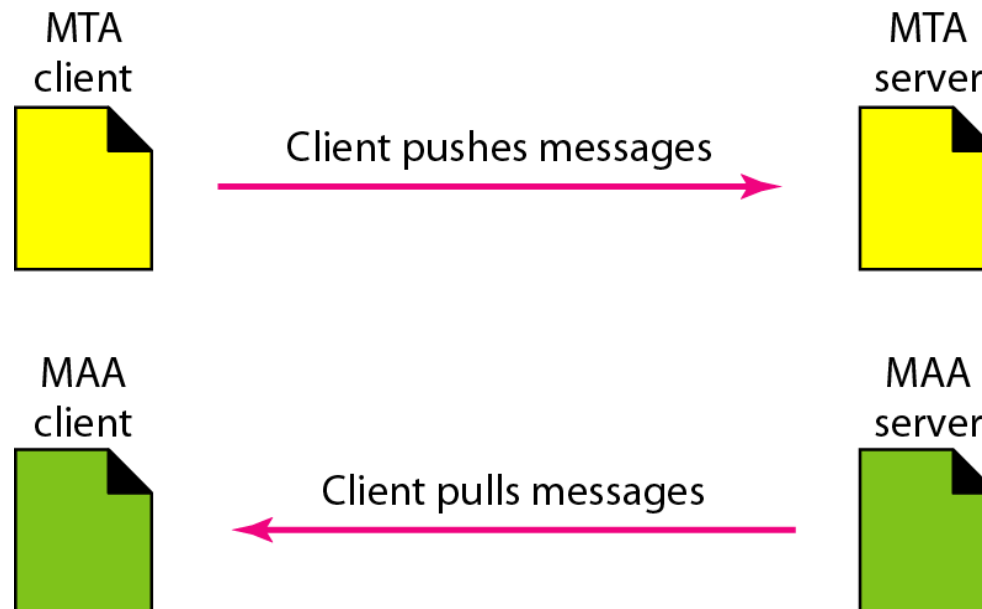
Electronic mail (Fourth Scenario)

■ There are two important points here

- First, Bob cannot bypass the mail server and use the MTA server directly
- To use MTA server directly, Bob would need to run the MTA server all the time because he does not know when a message will arrive
- This implies that Bob must keep his computer on all the time if he is connected to his system through a LAN
- If he is connected through a-WAN, he must keep the connection up all the time

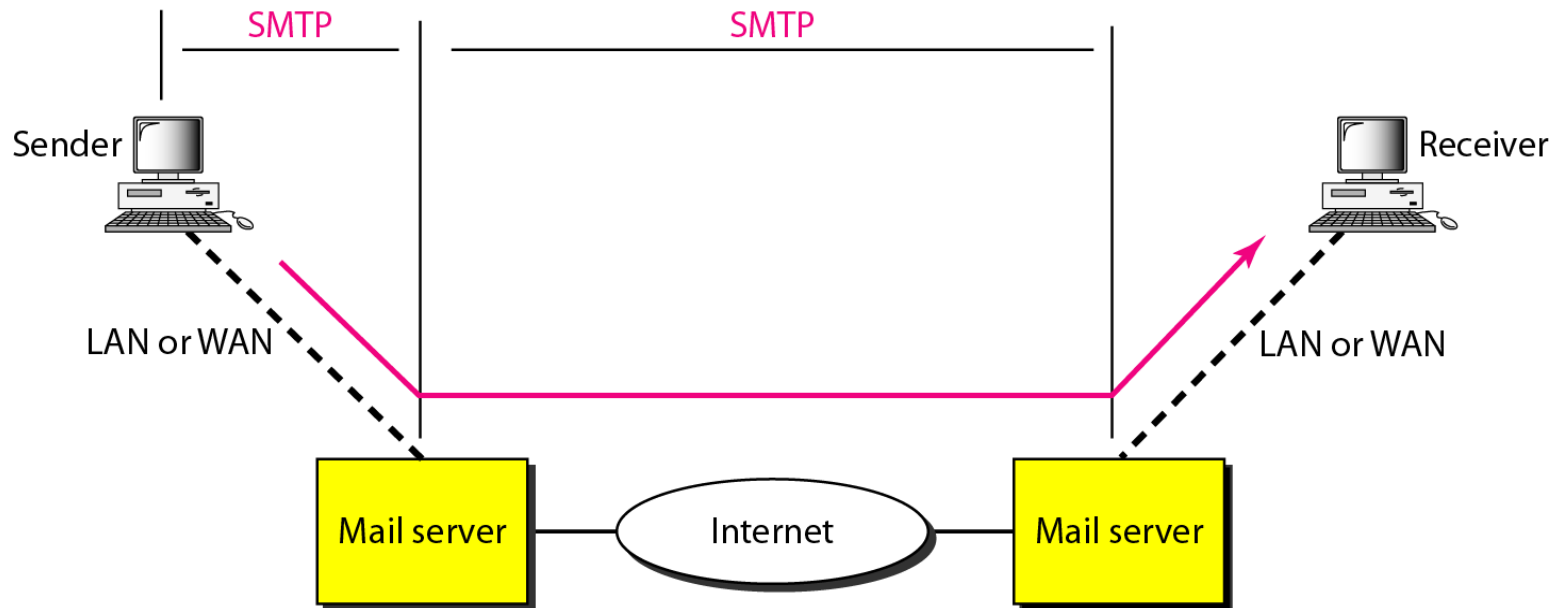
Electronic mail (Fourth Scenario)

- Second, note that Bob needs another pair of client/server programs: message access programs
- This is so because an MTA client/server program is a *push program*: the client pushes the message to the server
- Bob needs a *pull program*. The client needs to pull the message from the server



Message Transfer Agent: SMTP

- The actual mail transfer is done through message transfer agents
- To send mail, a system must have the client MTA, and to receive mail, a system must have a server MTA
- The formal protocol that defines the MTA client and server in the Internet is called the SMTP

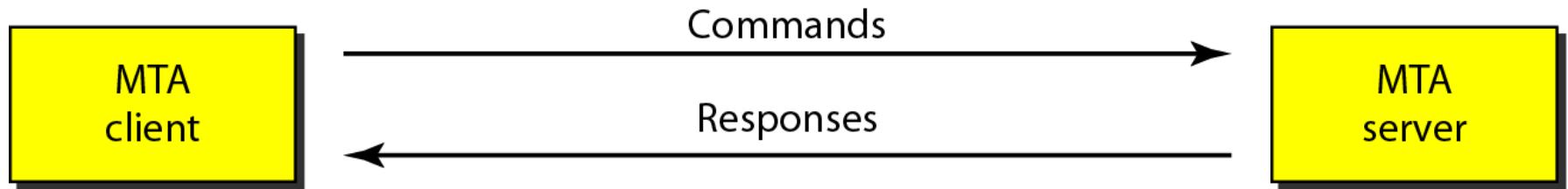


Message Transfer Agent: SMTP

■ Discuss the mechanism of mail transfer by SMTP

■ *Commands and Responses*

- SMTP uses commands and responses to transfer messages between an MTA client and an MTA server



Message Transfer Agent: SMTP

■ 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
- The next three are often used and highly recommended
- The last six are seldom used

Keyword: argument(s)

Message Transfer Agent: SMTP

<i>Keyword</i>	<i>Argument(s)</i>
HELO	Sender's host name
MAIL FROM	Sender of the message
RCPT TO	Intended recipient of the message
DATA	Body of the mail
QUIT	
RSET	
VERFY	Name of recipient to be verified
NOOP	
TURN	
EXPN	Mailing list to be expanded
HELP	Command name
SEND FROM	Intended recipient of the message
SMOL FROM	Intended recipient of the message
SMAL FROM	Intended recipient of the message

Message Transfer Agent: SMTP

■ 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

Message Transfer Agent: SMTP

<i>Code</i>	<i>Description</i>
Permanent Negative Completion Reply	
500	Syntax error; unrecognized command
501	Syntax error in parameters or arguments
502	Command not implemented
503	Bad sequence of commands
504	Command temporarily not implemented
550	Command is not executed; mailbox unavailable
551	User not local
552	Requested action aborted; exceeded storage location
553	Requested action not taken; mailbox name not allowed
554	Transaction failed

Message Transfer Agent: SMTP

■ *Mail Transfer Phases*

- The process of transferring a mail message occurs in three phases
 - Connection establishment
 - Mail transfer
 - Connection termination

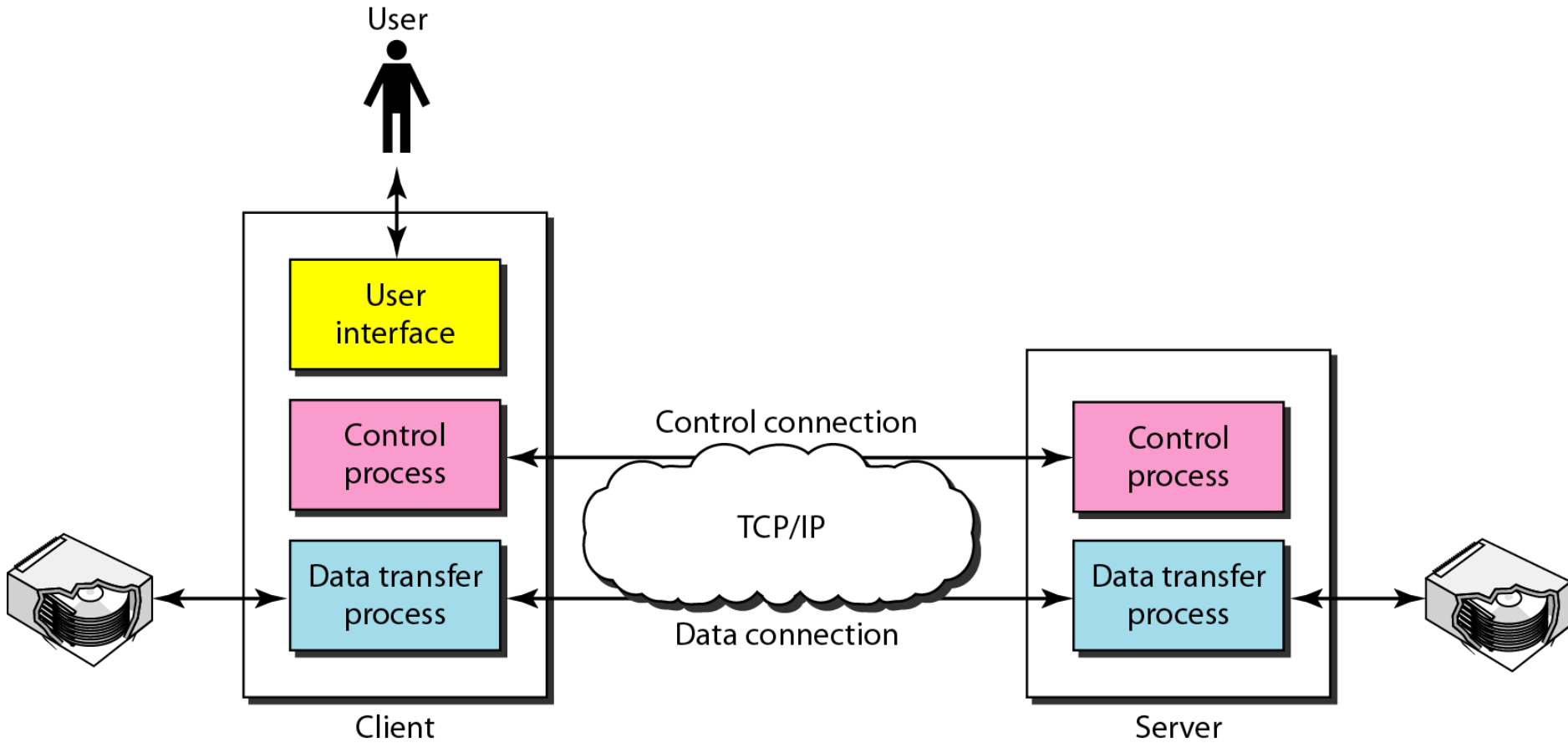
File Transfer

- Transferring files from one computer to another is one of the most common tasks expected from a networking or internetworking environment
- The greatest volume of data exchange in the Internet today is due to file transfer

File Transfer Protocol (FTP)

- FTP is the standard mechanism provided by *TCP/IP* for copying a file from one host to another
- FTP differs from other client/server applications
- It establishes two connections between the hosts
 - One connection is used for data transfer
 - The other for control information (commands and responses)
- FTP uses two well-known TCP ports
 - Port 21 is used for the control connection
 - Port 20 is used for the data connection

File Transfer Protocol (FTP)



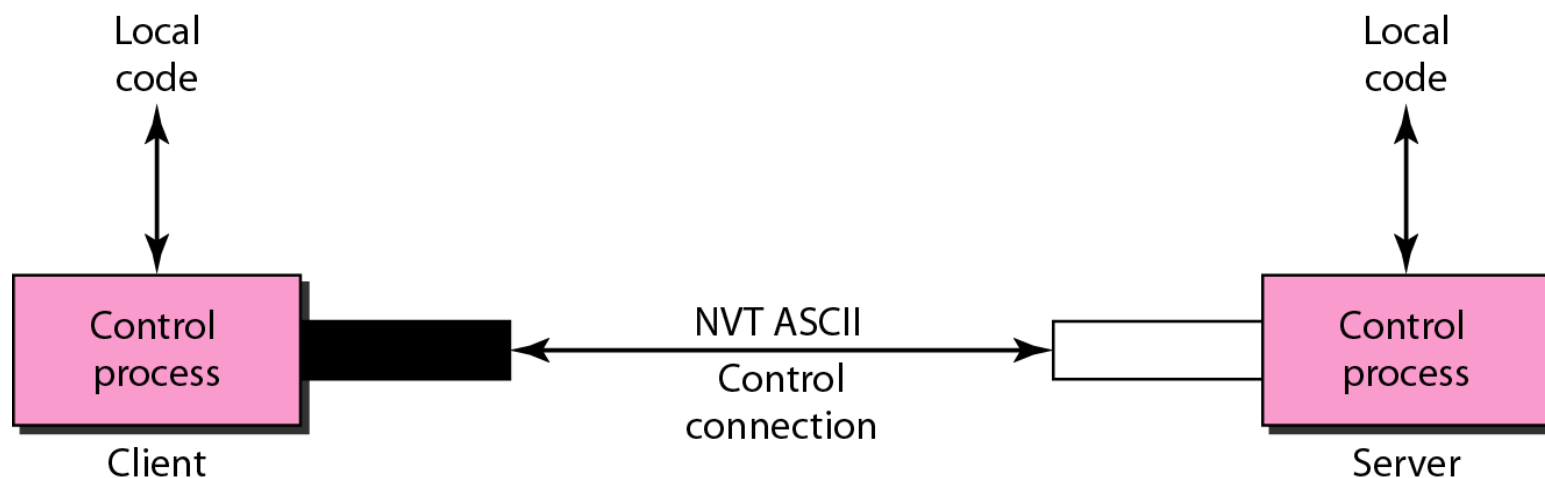
File Transfer Protocol (FTP)

- The client has three components
 - User interface
 - Client control process
 - The client data transfer process
- The server has two components
 - The server control process
 - The server data transfer process
- The control connection is made between the control processes
- The data connection is made between the data transfer processes

File Transfer Protocol (FTP)

■ *Communication over Control Connection*

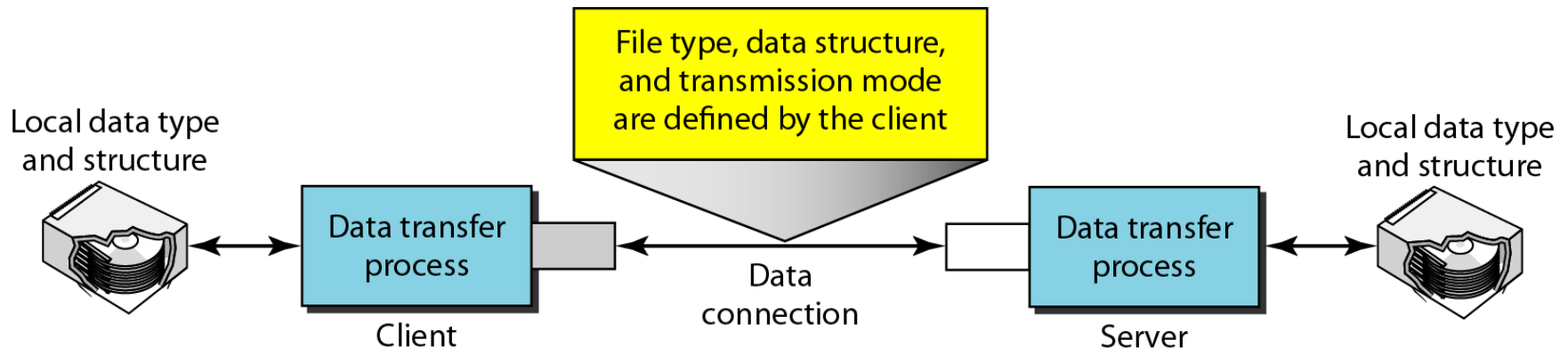
- FTP uses the same approach as SMTP to communicate across the control connection
- It uses the 7-bit ASCII character set
- Communication is achieved through commands and responses



File Transfer Protocol (FTP)

■ *Communication over Data Connection*

- The purpose of the data connection is different from that of the control connection
- File transfer occurs over the data connection under the control of the commands sent over the control connection



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

File Transfer Protocol (FTP)

■ File Type

- FTP can transfer one of the following file types across the data connection
 - An ASCII file
 - The ASCII file is the default format for transferring text files
 - EBCDIC file
 - If one or both ends of the connection use EBCDIC encoding , the file can be transferred using EBCDIC encoding
 - Image file
 - The image file is the default format for transferring binary files

File Transfer Protocol (FTP)

■ 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
 - In the file structure format, the file is a continuous stream of bytes
 - Record structure
 - In the record structure, the file is divided into records.
 - This can be used only with text files
 - Page structure
 - In the page structure, the file is divided into pages, with each page having a page number and a page header

File Transfer Protocol (FTP)

■ Transmission Mode

- FTP can transfer a file across the data connection by using one of the following three transmission modes
 - Stream mode
 - The stream mode is the default mode.
 - Data are delivered from FTP to TCP as a continuous stream of bytes
 - Block mode
 - In block mode, data can be delivered from FTP to TCP in blocks
 - Compressed mode
 - In the compressed mode, if the file is big, the data can be compressed

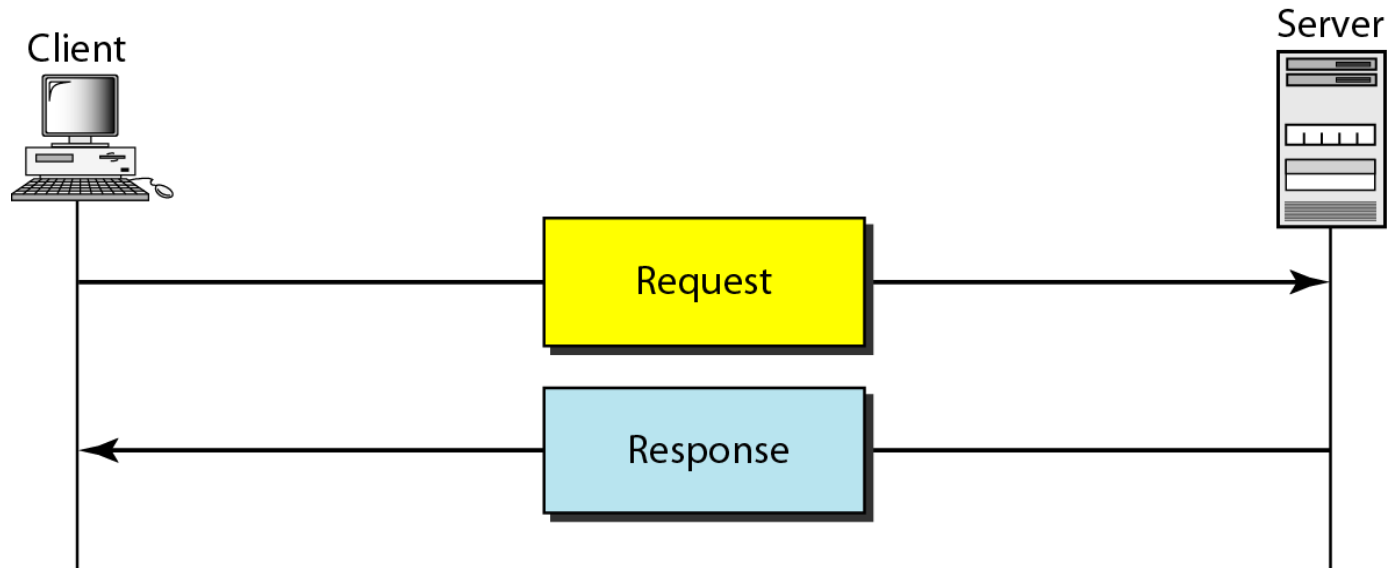
HTTP

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

HTTP (Outline)

- HTTP Transaction
- Persistent Versus Nonpersistent Connection
- Proxy Server

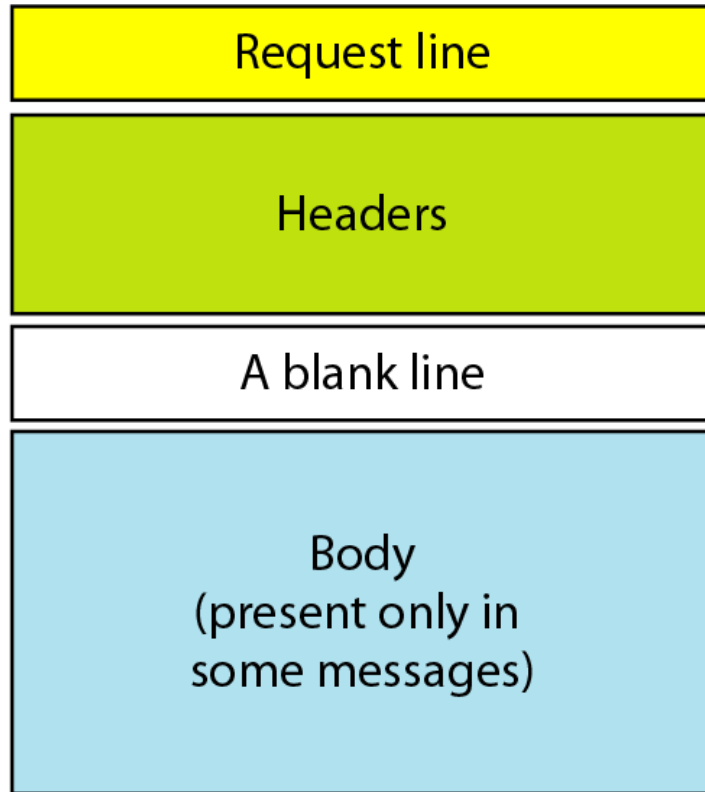
HTTP Transaction



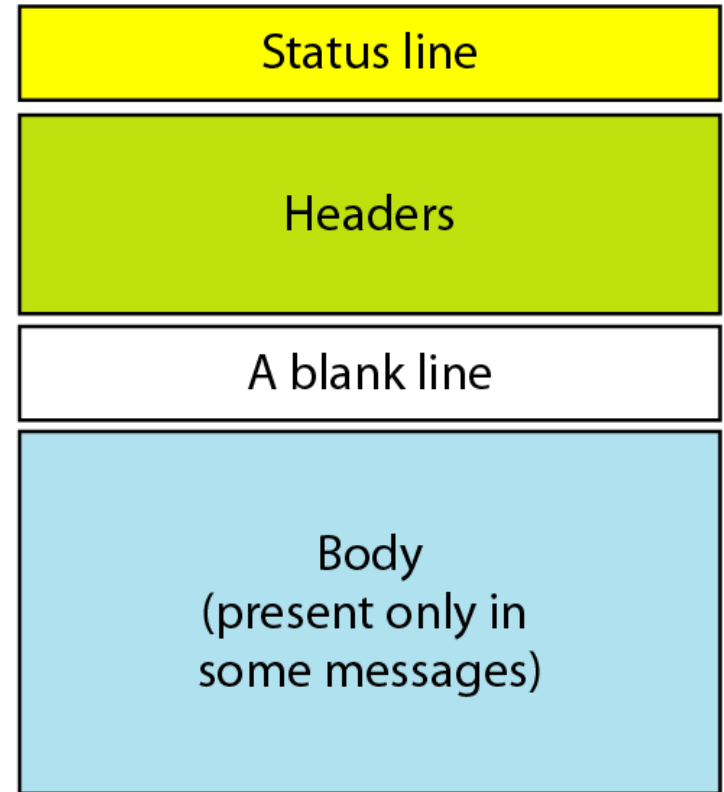
- HTTP transaction between the client and server
- Although HTTP uses the services of TCP, HTTP itself is a stateless protocol
- The client initializes the transaction by sending a request message
- The server replies by sending a response

HTTP Transaction

■ Message



Request message



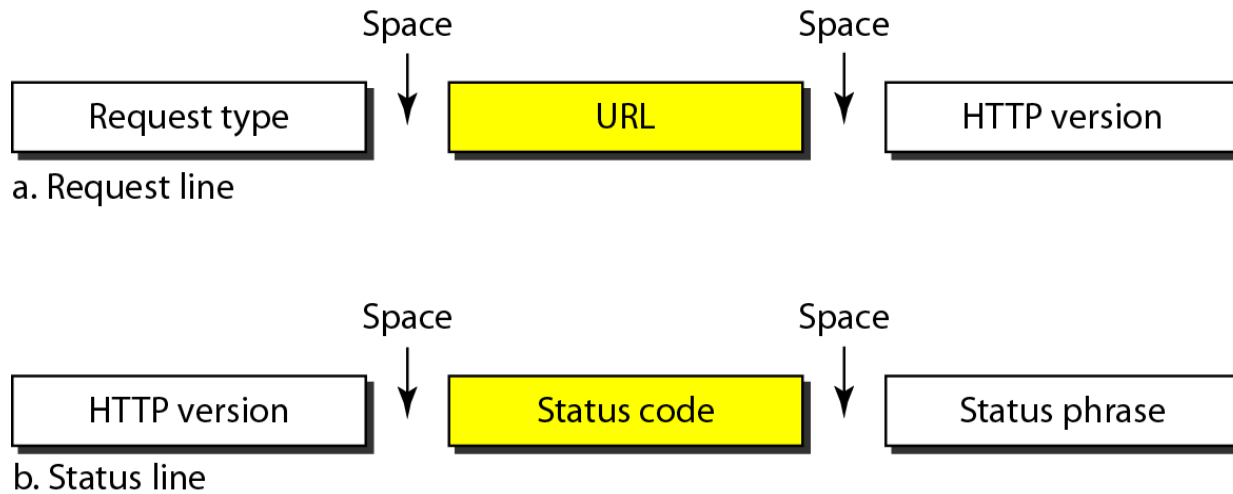
Response message

HTTP Transaction

■ Message

- 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

HTTP Transaction



■ Request 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
- Version
 - The most current version of HTTP is 1.1

HTTP Transaction

■ Status Lines

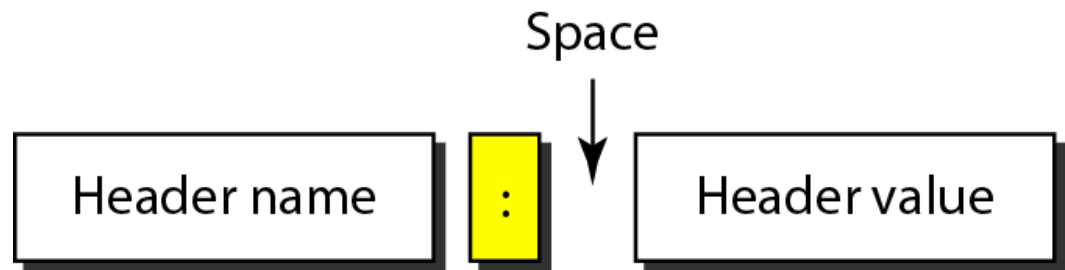
- Status code
 - It consists of three digits
 - Informational (100 range)
 - Success (200 range)
 - Redirection (300 range)
 - Client Error (400 range)
 - Server Error (500 range)
- Status phrase
 - This field is used in the response message
 - It explains the status code in text form

HTTP Transaction

■ 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
- A header value



- A header line belongs to one of four categories
 - General header
 - Request header
 - Response header

HTTP Transaction

■ General header

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

■ 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

■ 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

■ Entity header

- The entity header gives information about the body of the document

HTTP Transaction

■ Body

- The body can be present in a request or response message
- It contains the document to be sent or received

Persistent Versus Nonpersistent Connection

■ *Persistent Connection*

- HTTP version 1.1 specifies a persistent connection by default
- 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

■ *Nonpersistent Connection*

- One TCP connection is made for each request/response
- The following lists the steps in this strategy:
 - The client opens a TCP connection and sends a request
 - The server sends the response and closes the connection
 - The client reads the data until it encounters an end-of-file marker; it then closes the connection

Proxy Server

- HTTP supports proxy servers
- A proxy server is a computer that keeps copies of responses to recent requests
- The HTTP client sends a request to the proxy server
- The proxy server checks its cache
- If the response is not stored in the cache, the proxy server sends the request to the corresponding server
- Incoming responses are sent to the proxy server and stored for future requests from other clients
- The proxy server reduces the load on the original server, decreases traffic, and improves latency

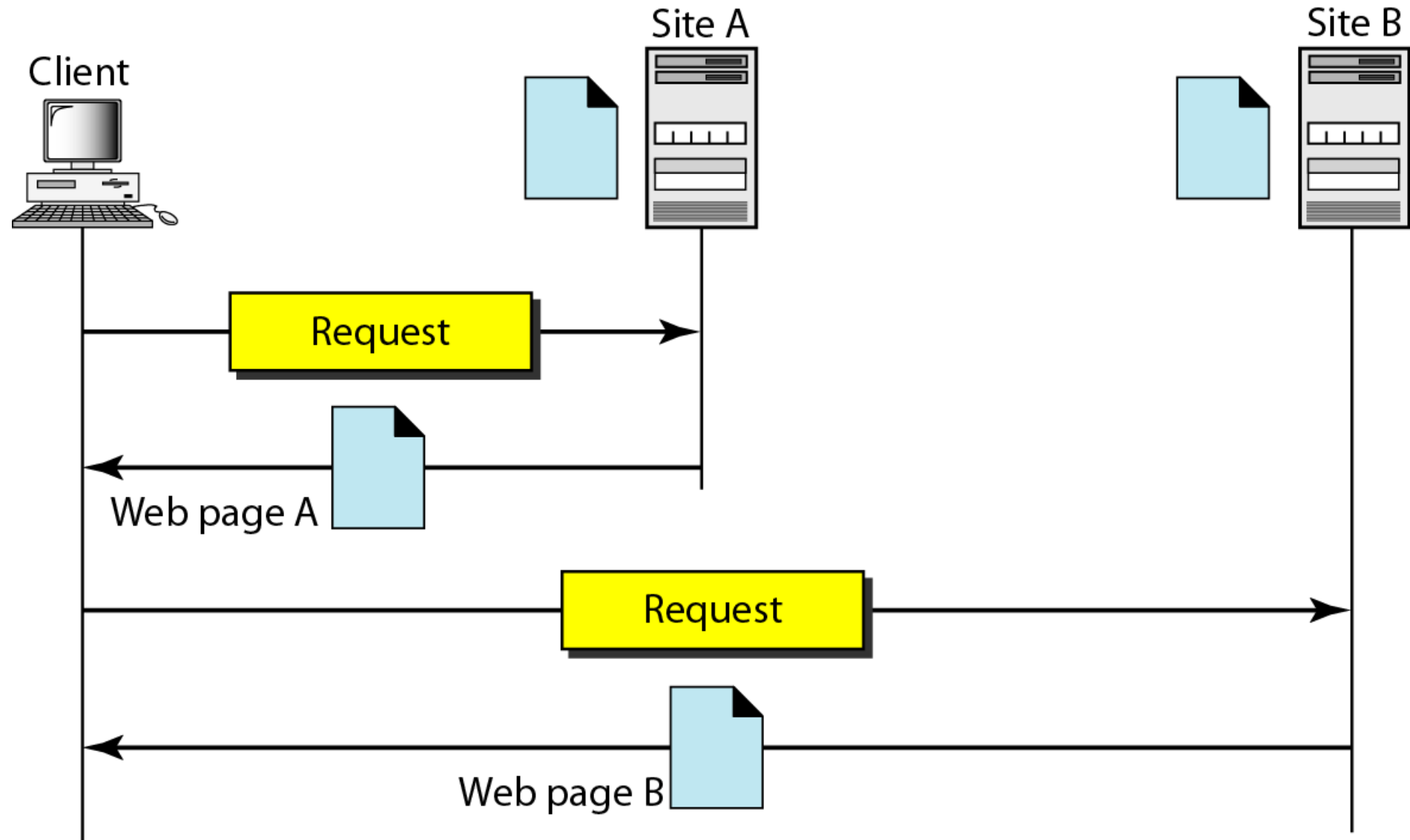
- The World Wide Web (WWW) is a repository of information linked together from points all over the world
- The WWW has a unique combination of flexibility, portability, and user-friendly features that distinguish it from other services provided by the Internet

- Architecture
- Web Documnet

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 pages can be retrieved and viewed by using browsers

Architecture



Architecture

■ Client (Browser)

- Each browser usually consists of three parts
 - A controller
 - The controller receives input from the keyboard or the mouse and uses the client programs to access the document
 - Client protocol
 - The client protocol can be one of the protocols described previously such as FTP or HTTP
 - Interpreters
 - The interpreter can be HTML, Java, or JavaScript, depending on the type of document

■ Server

- The Web page is stored at the server
- Each time a client request arrives, the corresponding document is sent to the client

Architecture

■ Uniform Resource Locator (URL)

- A client that wants to access a Web page needs the address
- URL is a standard for specifying any kind of information on the Internet
- The URL defines four things

- Protocol



- The protocol is the client/server program used to retrieve the document

- Host computer

- The host is the computer on which the information is located

- Port

- The URL can optionally contain the port number of the server

- Path

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

Architecture

■ Cookies

- Cookies are small files that websites put on your computer hard disk drive when you first visit
- A message given to a Web browser by a Web server
- The browser stores the message in a text file
- The message is then sent back to the server each time the browser requests a page from the server
- The main purpose of cookies is to identify users and possibly prepare customized Web pages for them

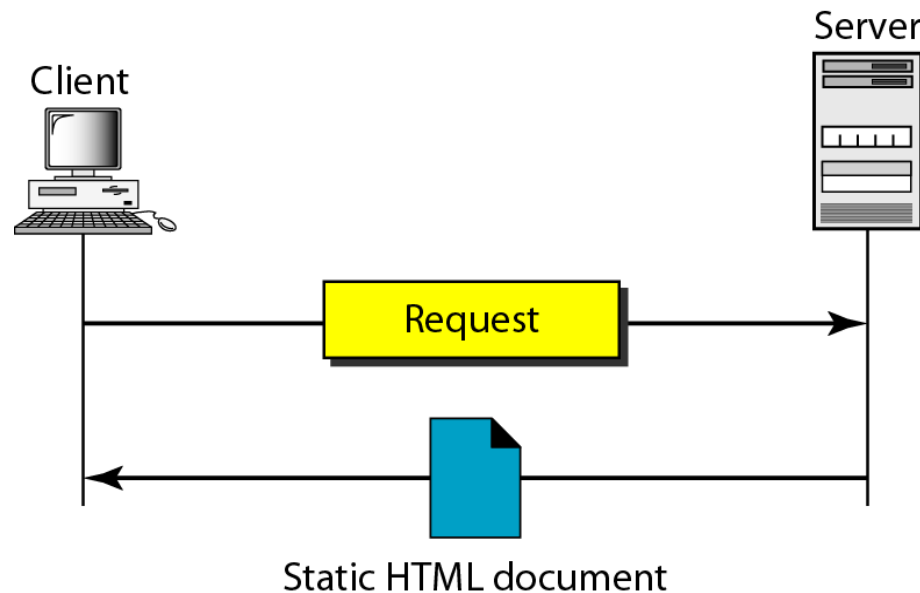
Web Document

- The documents in the WWW can be grouped into three broad categories
 - Static
 - Dynamic
 - Active

Web Document

■ Static Documents

- Static documents are fixed-content documents that are created and stored in a server



■ *HTML*

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

Web Document

■ Dynamic Documents

- A dynamic document is created by a Web server whenever a browser requests the document
- A very simple example of a dynamic document is the retrieval of the time and date from a server
- *Common Gateway Interface (CGI)*
 - CGI is a technology that creates and handles dynamic documents.
- *Scripting Technologies for Dynamic Documents*
 - A few technologies have been involved in creating dynamic documents using scripts
 - Hypertext Preprocessor (pHP), which uses the Perl language
 - Java Server Pages (JSP), which uses the Java language for scripting
 - ActiveServer Pages (ASP), which uses Visual Basic language for scripting
 - ColdFusion, which embeds SQL database queries in the HTML document

Web Document

■ Active Documents

- A program or a script to be run at the client site. These are called active documents
- Example, suppose we want to run a program that creates animated graphics on the screen or a program that interacts with the user
- *Java Applets*
 - One way to create an active document is to use Java applets
- *JavaScript*
 - The idea of scripts in dynamic documents can also be used for active documents