GLA UNIVERSITY

COMPUTER NETWORK

By:

Dr. Ankush Agarwal

APPLICATION LAYER

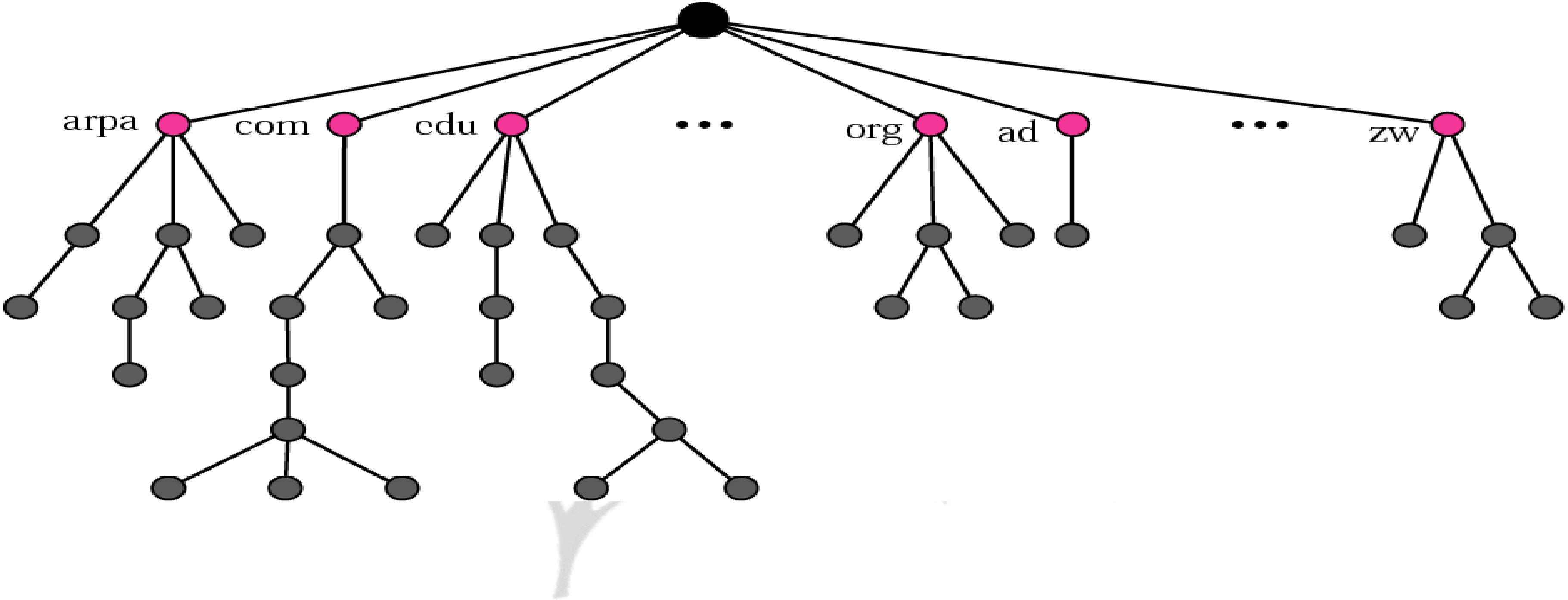
Name space

- The names assigned to machines 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

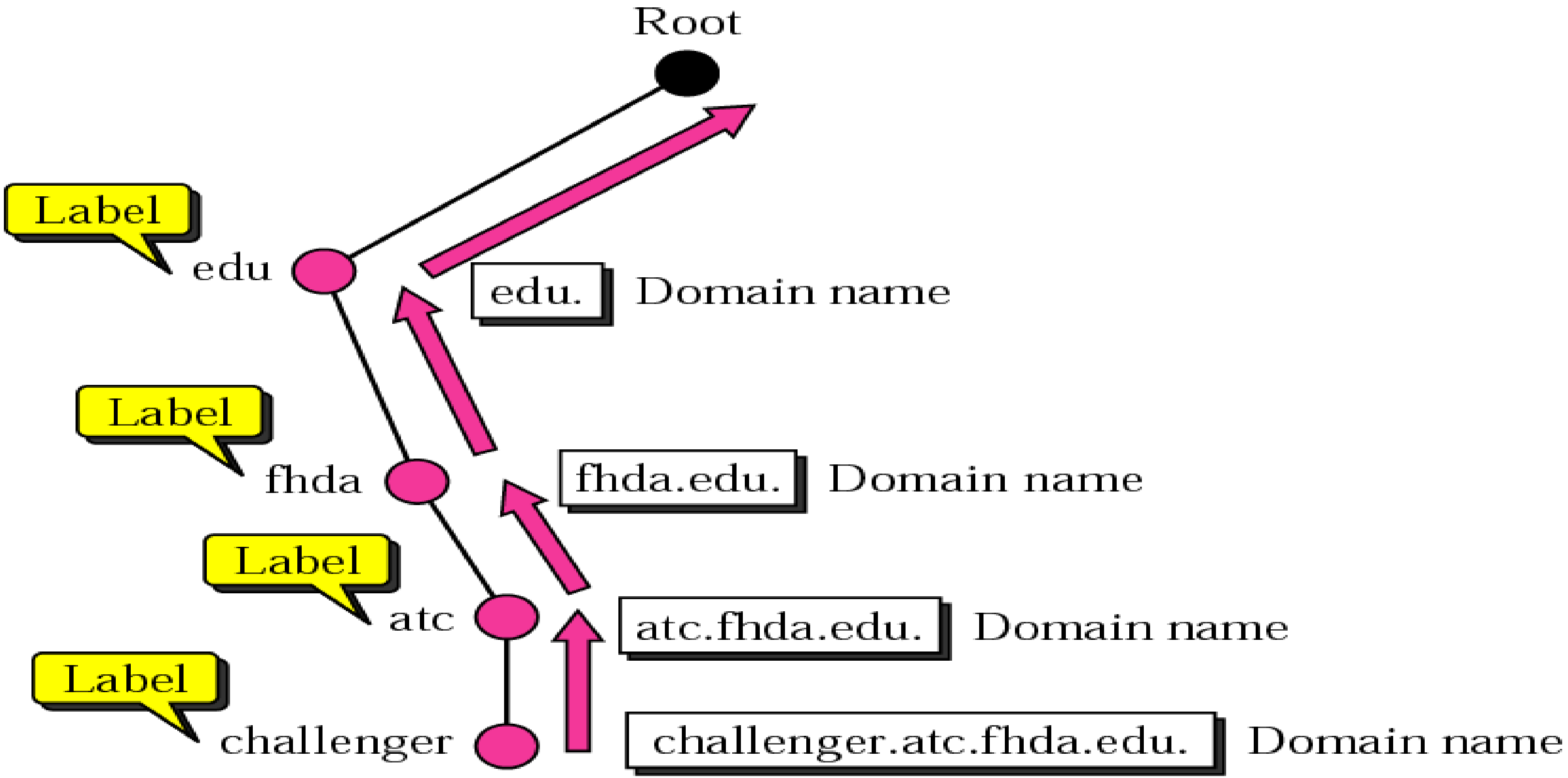
Domain Name Space (DNS)

- The domain name space is hierarchical in design
- The names are defined in an inverted-tree structure with the root at the top
- The tree can have 128 levels: level 0 (root) to level 127

Domain Name Space (DNS)



Domain Name and Labels



Distribution of name space

- The information contained in the domain name space is distributed among many computers called DNS servers

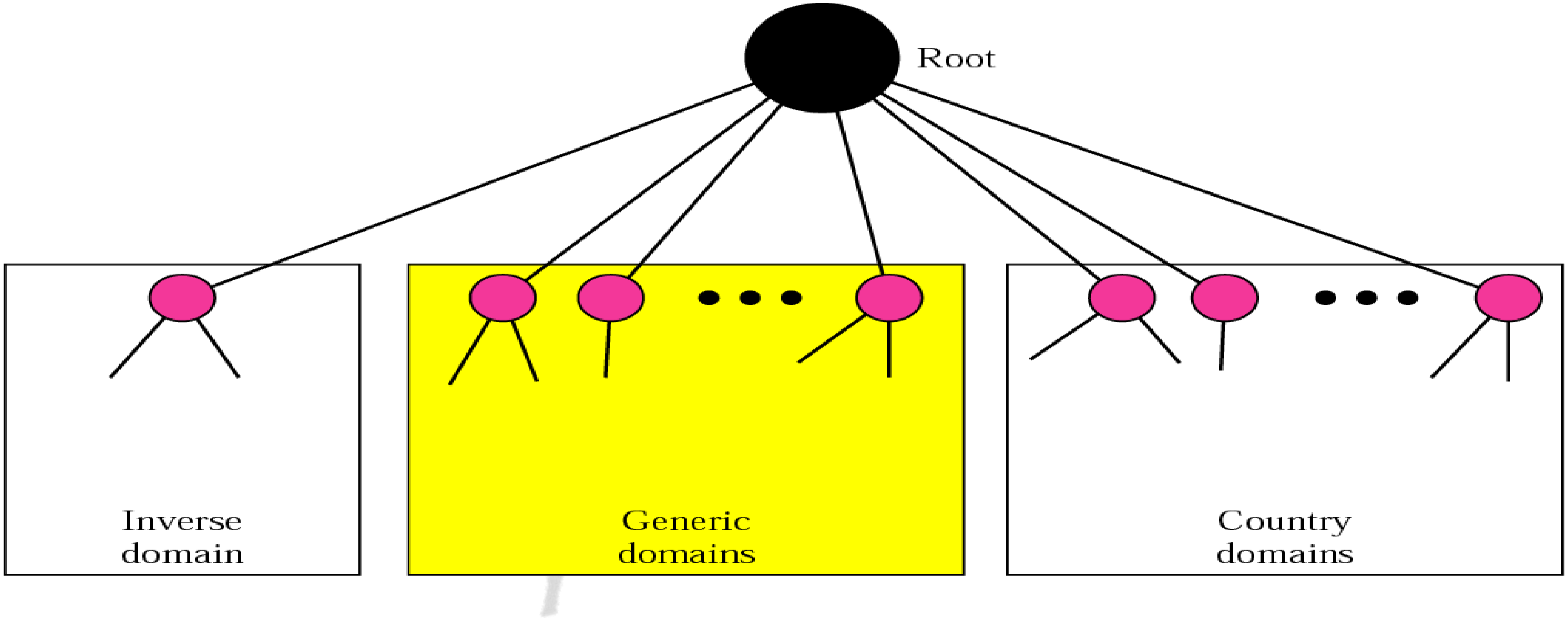
Zones and domains

- 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

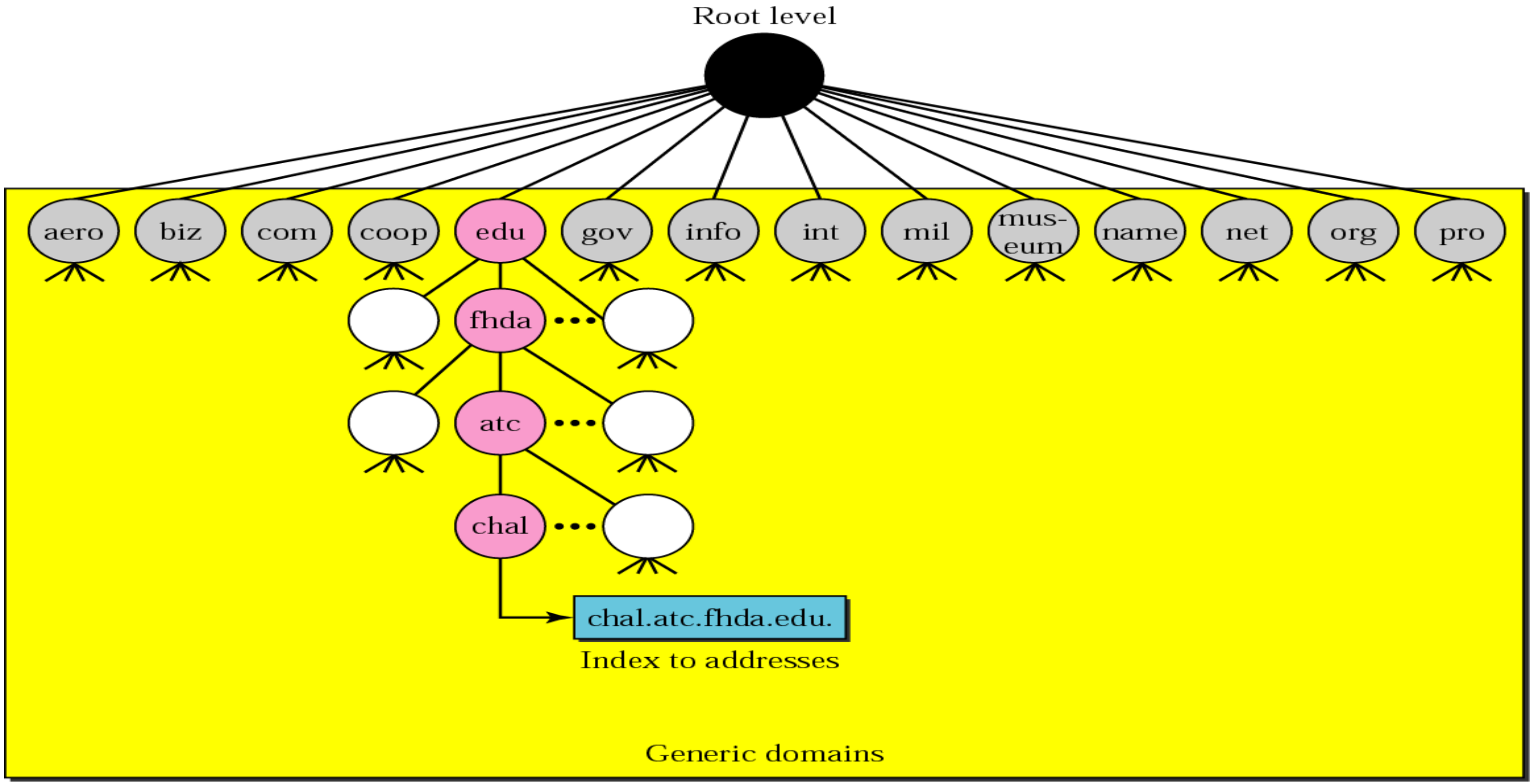
DNS in the internet

- The domain name space (tree) is divided into three different sections
  - generic domains
  - country domains
  - inverse domain

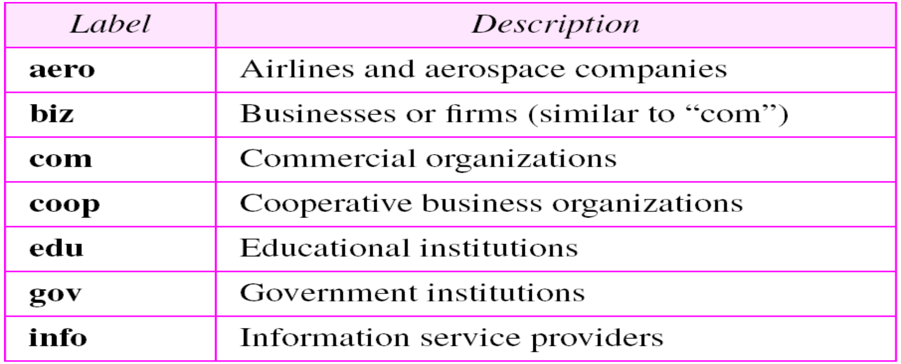
DNS used in the Internet



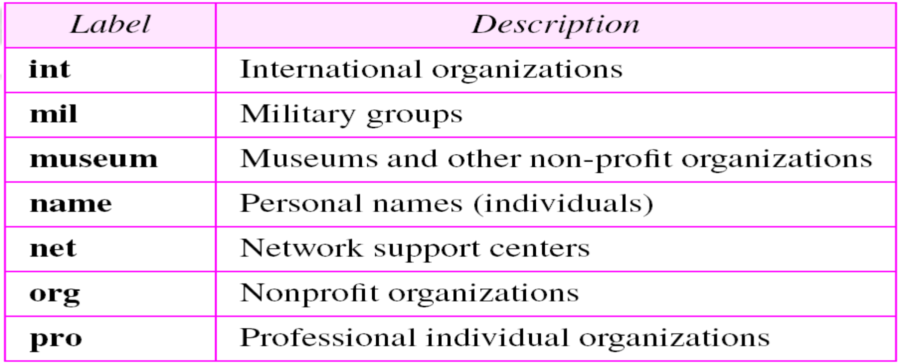
Generic domain



Generic domain labels

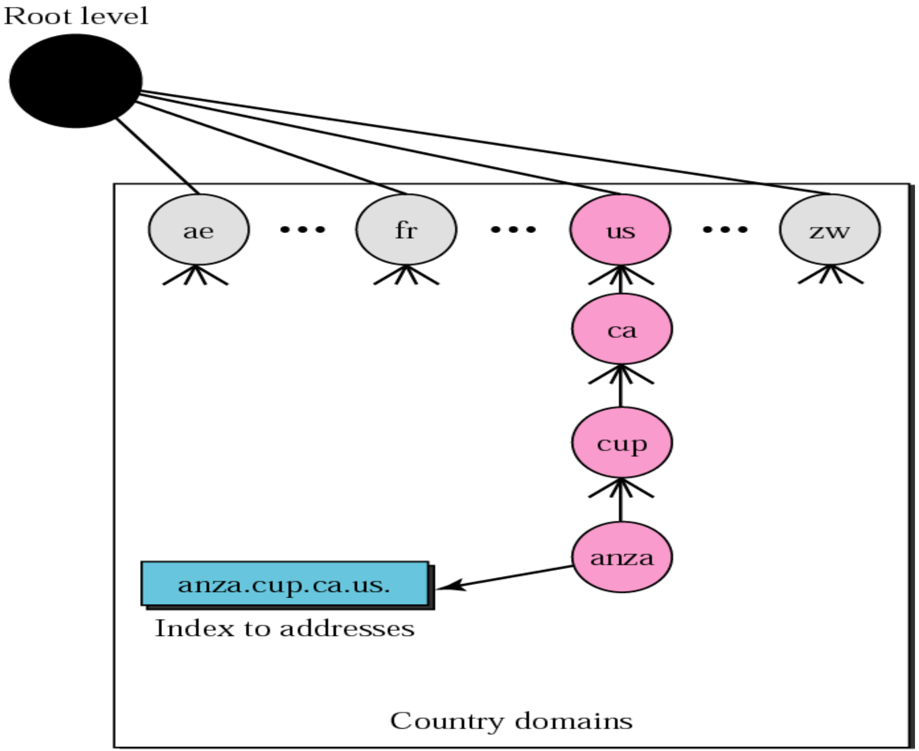


Label	*Description*
aero	Airlines and aerospace companies
biz	Businesses or firms (similar to “com”)
com	Commercial organizations
coop	Cooperative business organizations
edu	Educational institutions
gov	Government institutions
info	Information service providers

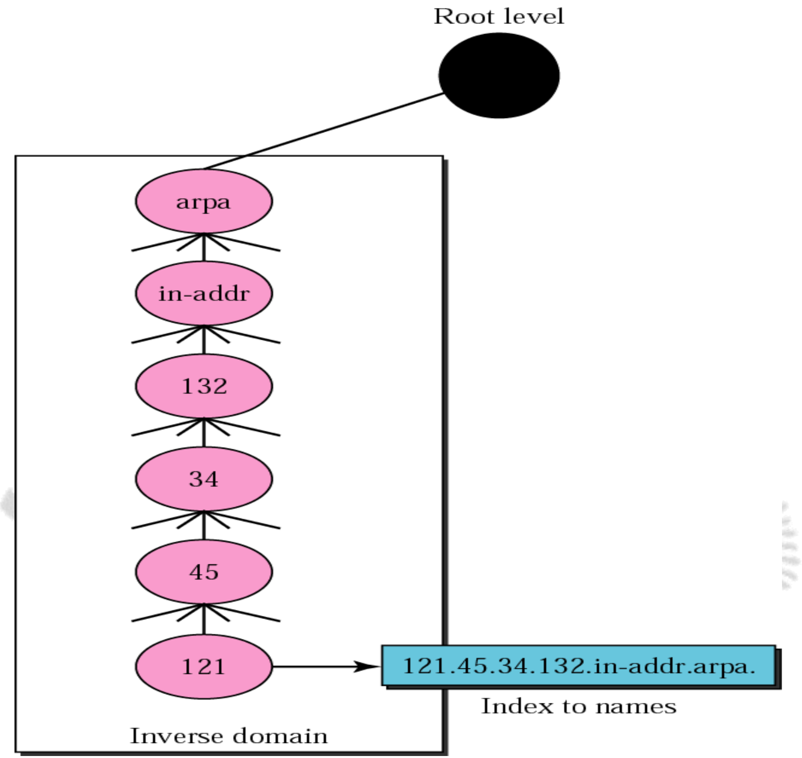


Label	*Description*
int	International organizations
mil	Military groups
museum	Museums and other non-profit organizations
name	Personal names (individuals)
net	Network support centers
org	Nonprofit organizations
pro	Professional individual organizations

Country domain



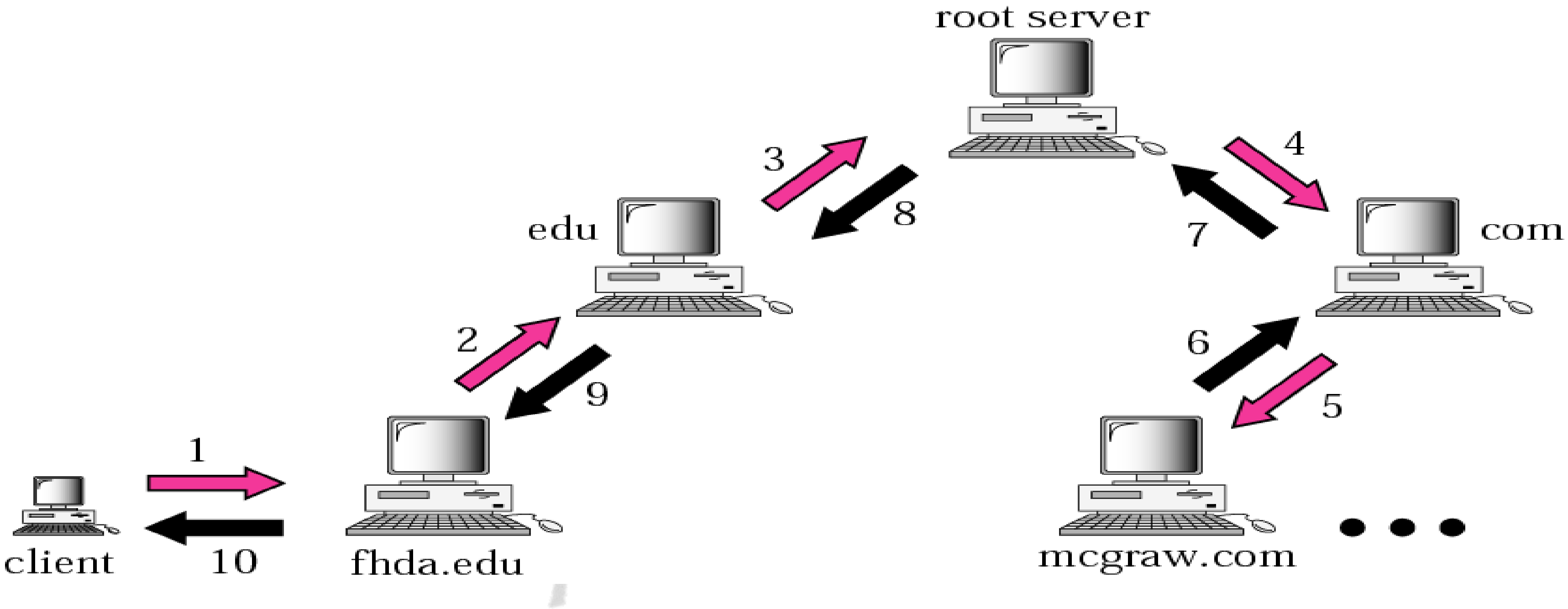
Inverse domain



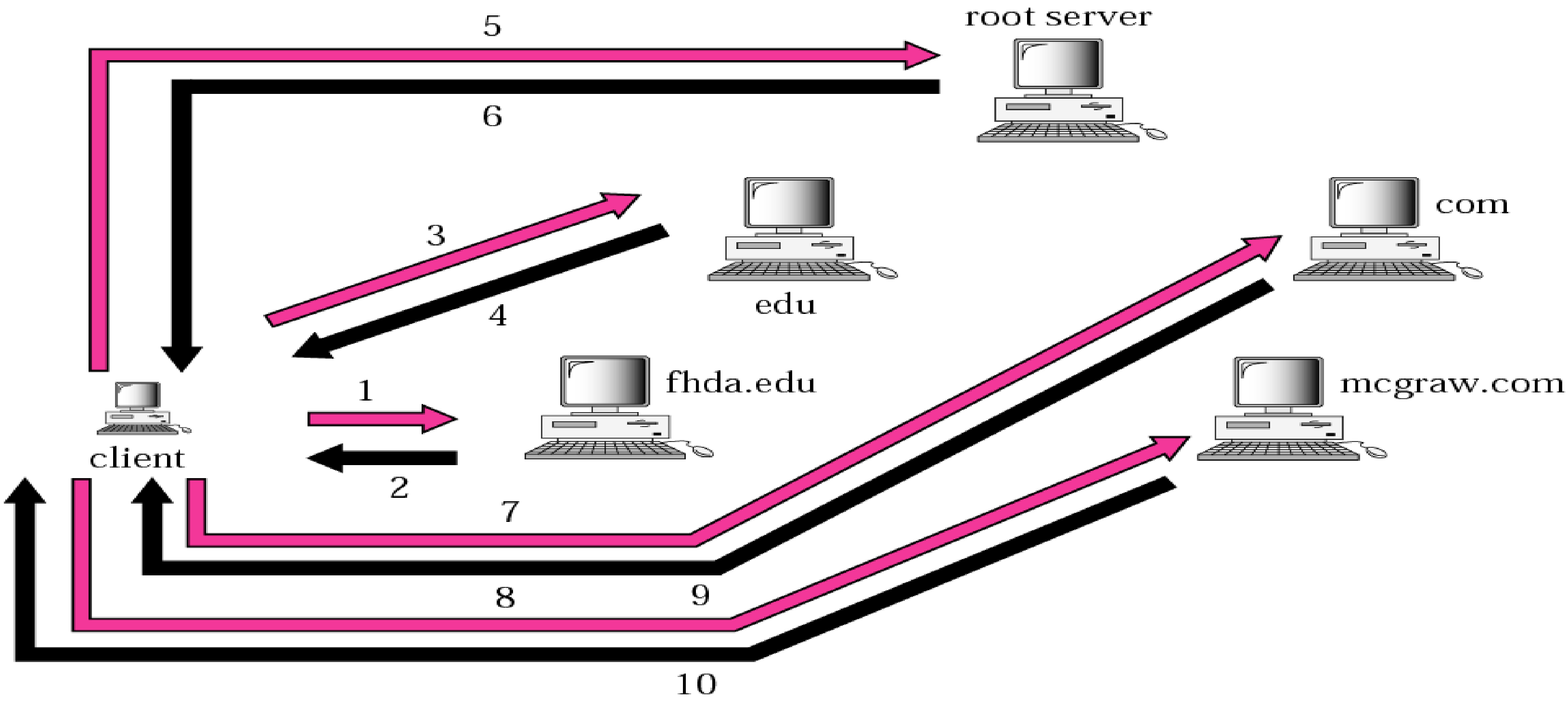
Resolution

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

Recursive resolution



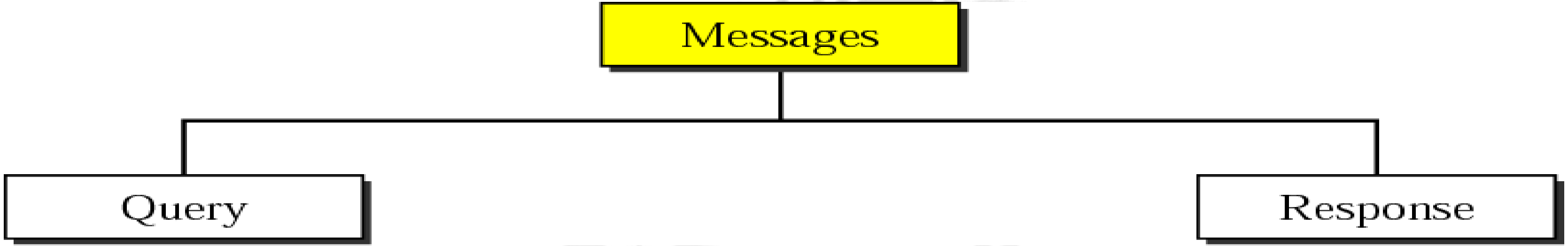
Iterative resolution



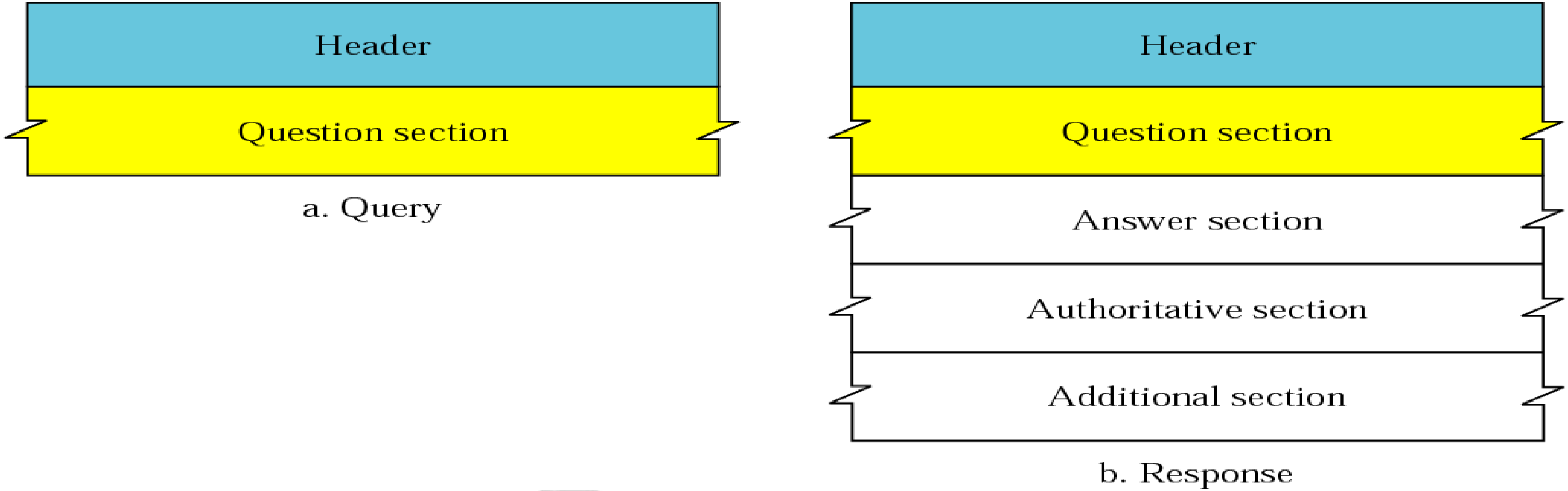
DNS messages

- The DNS query message consists of a
  - Header
  - Question records
- The DNS response message consists of a
  - Header
  - question records
  - answer records
  - authoritative records
  - additional records

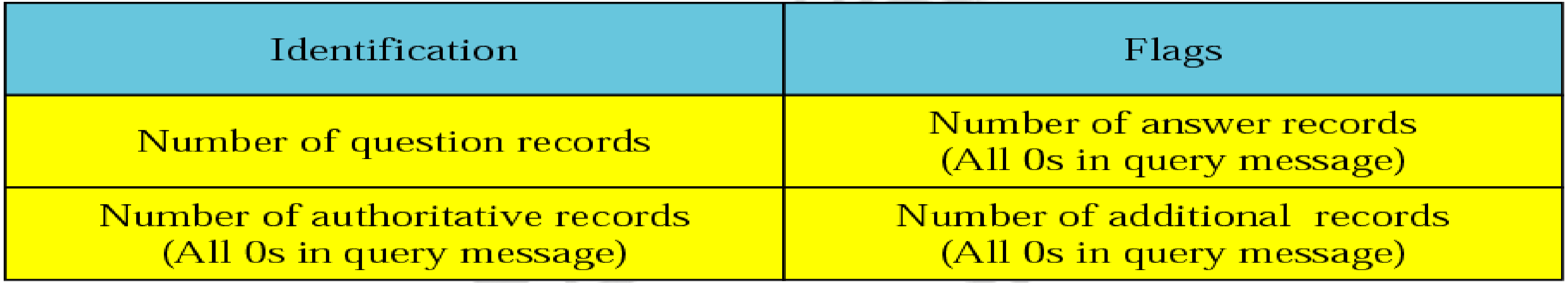
DNS messages



Query and response messages

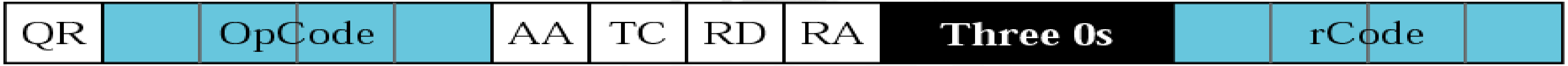


Header format



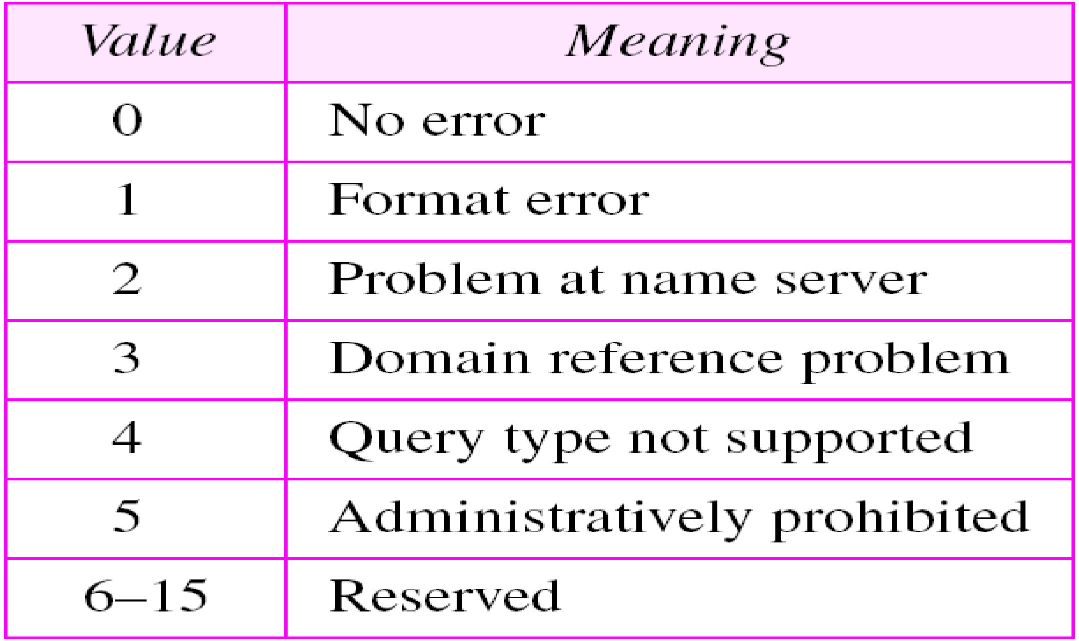
Identification	Flags
Number of question records	Number of answer records (All 0s in query message)
Number of authoritative records (All 0s in query message)	Number of additional records (All 0s in query message)

Flag fields



QR	OpCode	AA	TC	RD	RA	Three 0s	rCode

Values of rCode

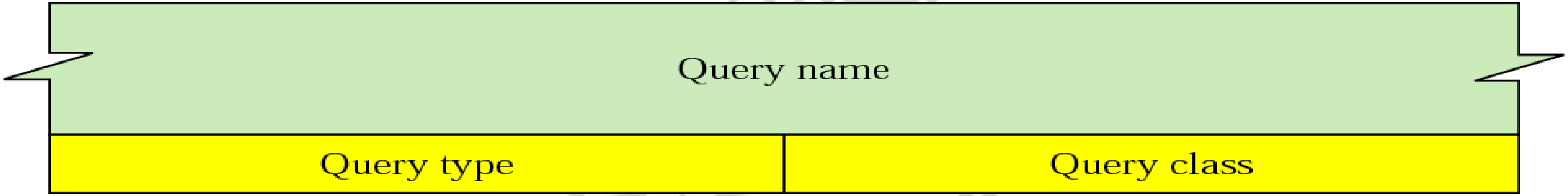


Value	*Meaning*
0	No error
1	Format error
2	Problem at name server
3	Domain reference problem
4	Query type not supported
5	Administratively prohibited
6–15	Reserved

Types of records

- Two types of records are used in DNS
- The question records are used in the question section of the query and response messages
- The resource records are used in the answer, authoritative, and additional information sections of the response message

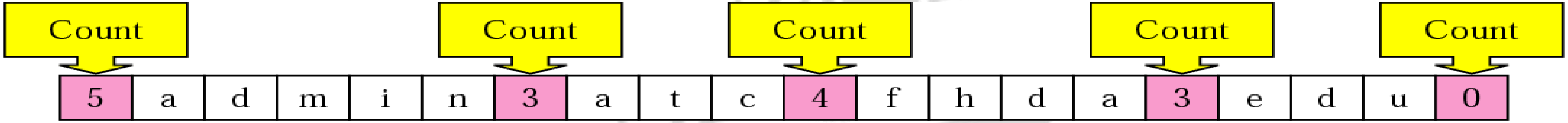
Question record format


Query name

Query type

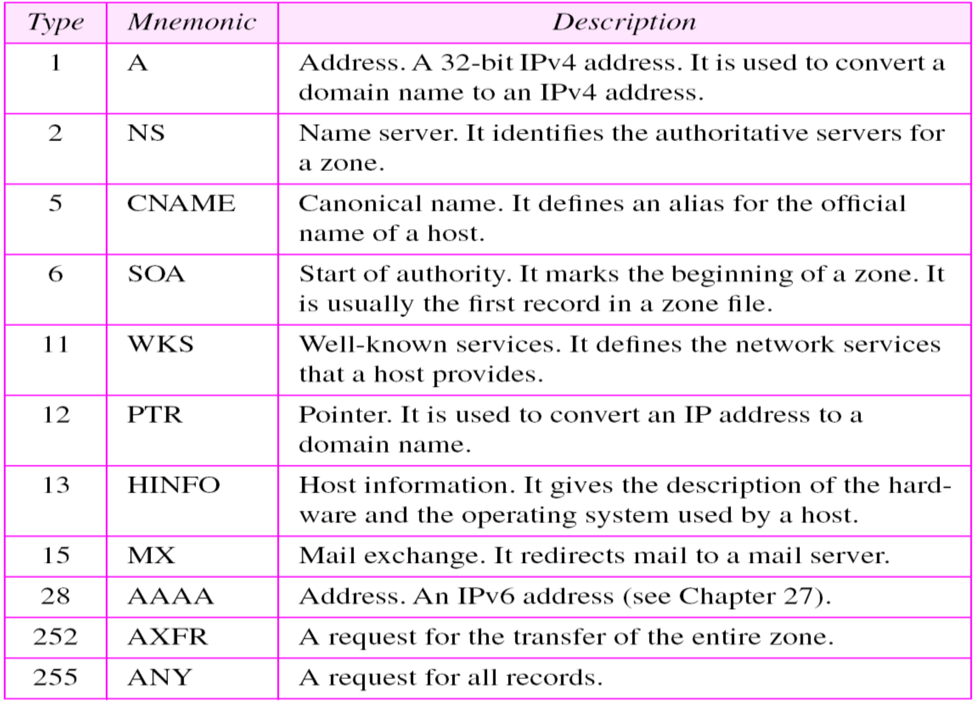
Query class

Query name format



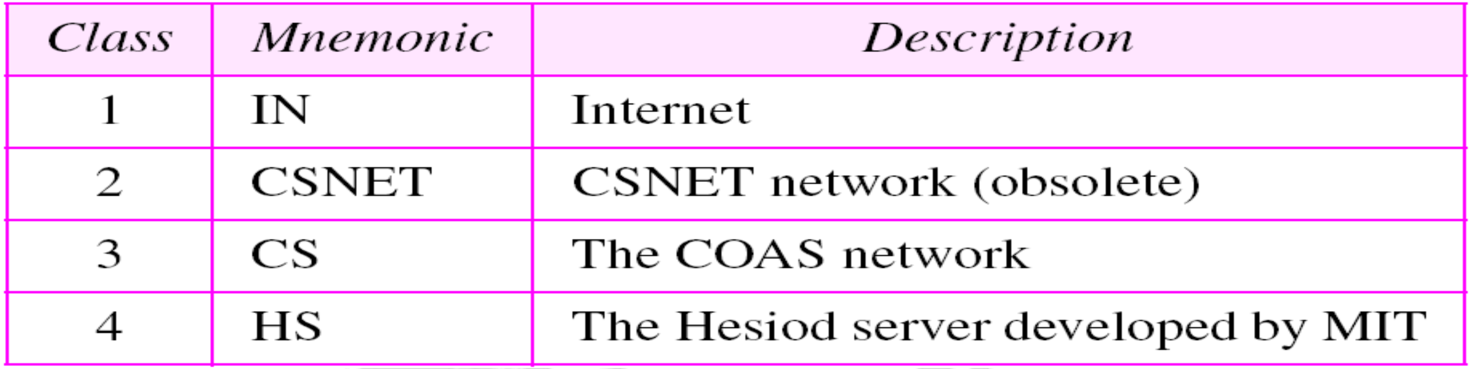
5	a	d	m	i	n	3	a	t	c	4	f	h	d	a	3	e	d	u	0

Types



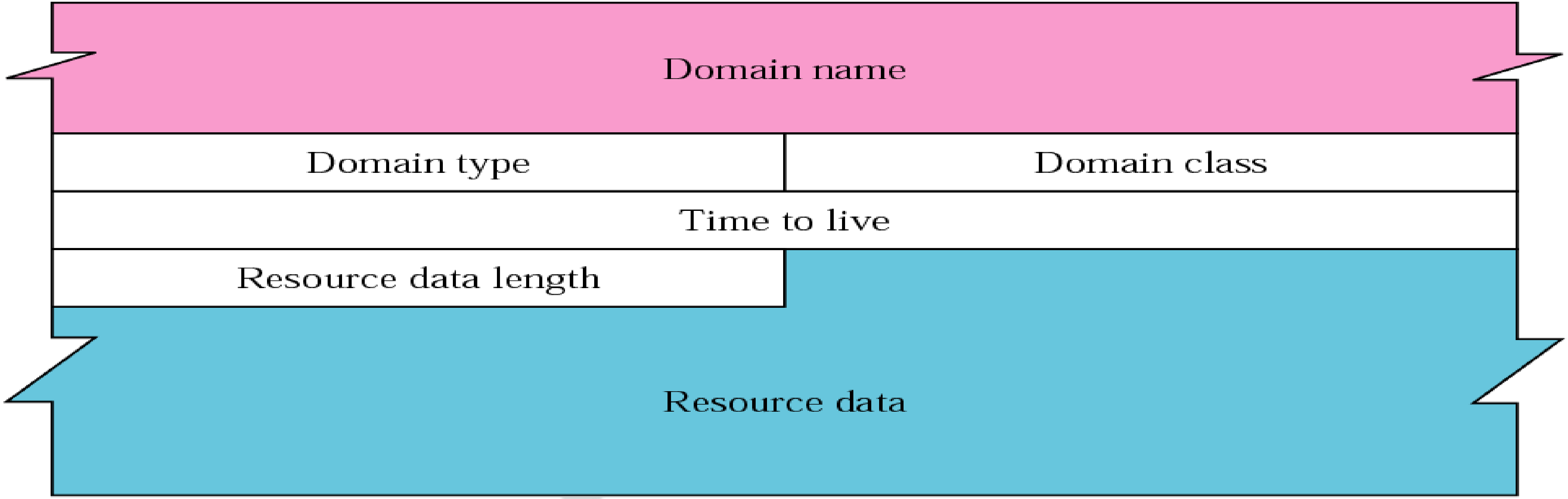
Type	*Mnemonic*	*Description*
1	A	Address. A 32-bit IPv4 address. It is used to convert a domain name to an IPv4 address.
2	NS	Name server. It identifies the authoritative servers for a zone.
5	CNAME	Canonical name. It defines an alias for the official name of a host.
6	SOA	Start of authority. It marks the beginning of a zone. It is usually the first record in a zone file.
11	WKS	Well-known services. It defines the network services that a host provides.
12	PTR	Pointer. It is used to convert an IP address to a domain name.
13	HINFO	Host information. It gives the description of the hardware and the operating system used by a host.
15	MX	Mail exchange. It redirects mail to a mail server.
28	AAAA	Address. An IPv6 address (see Chapter 27).
252	AXFR	A request for the transfer of the entire zone.
255	ANY	A request for all records.

Classes



Class	*Mnemonic*	*Description*
1	IN	Internet
2	CSNET	CSNET network (obsolete)
3	CS	The COAS network
4	HS	The Hesiod server developed by MIT

Resource record format



Domain name

Domain type | Domain class

Time to live

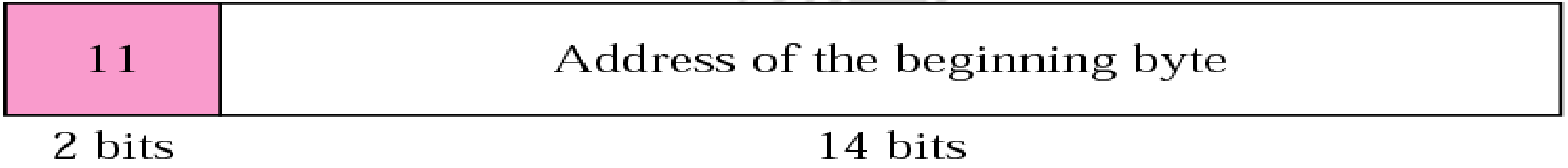
Resource data length

Resource data

Compression

- DNS requires that a domain name be replaced by an offset pointer if it is repeated
- DNS defines a 2-byte offset pointer that points to a previous occurrence of the domain name or part of it

Format of an offset pointer



11	Address of the beginning byte
2 bits	14 bits

DDNS

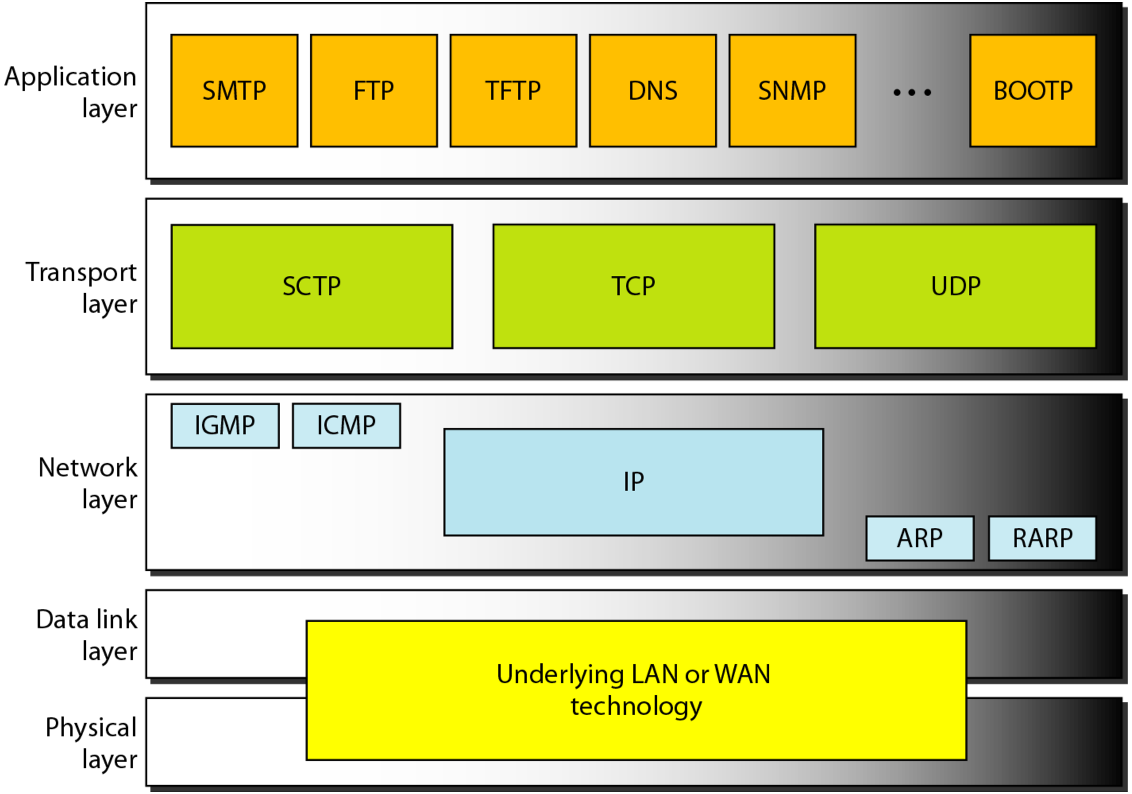
- The Dynamic Domain Name System (DDNS) updates the DNS master file dynamically

Encapsulation

- DNS uses UDP as the transport protocol when the size of the response message is less than 512 bytes
- If the size of the response message is more than 512 bytes, a TCP connection is used

- DNS can use the services of UDP or TCP using the well-known port 53

FTP

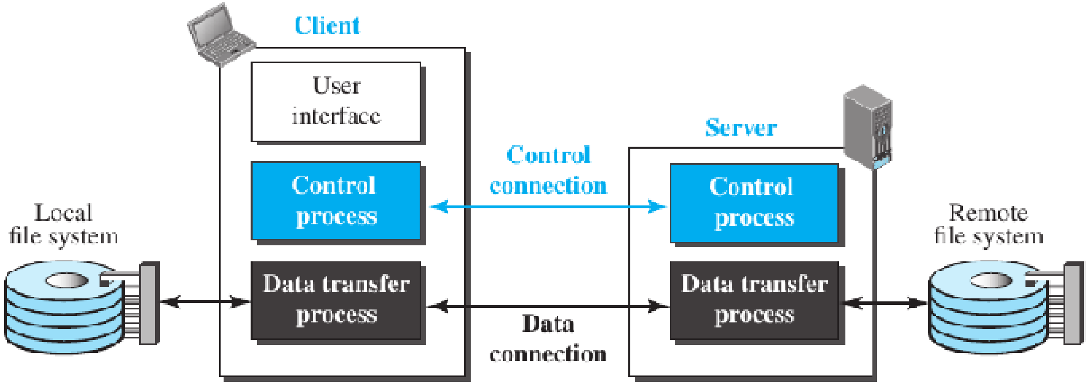

Application layer
SMTP
FTP
TFTP
DNS
SNMP
• • •
BOOTP
Transport layer
SCTP
TCP
UDP
Network layer
IGMP
ICMP
IP
ARP
RARP
Data link layer
Physical layer
Underlying LAN or WAN technology

FTP

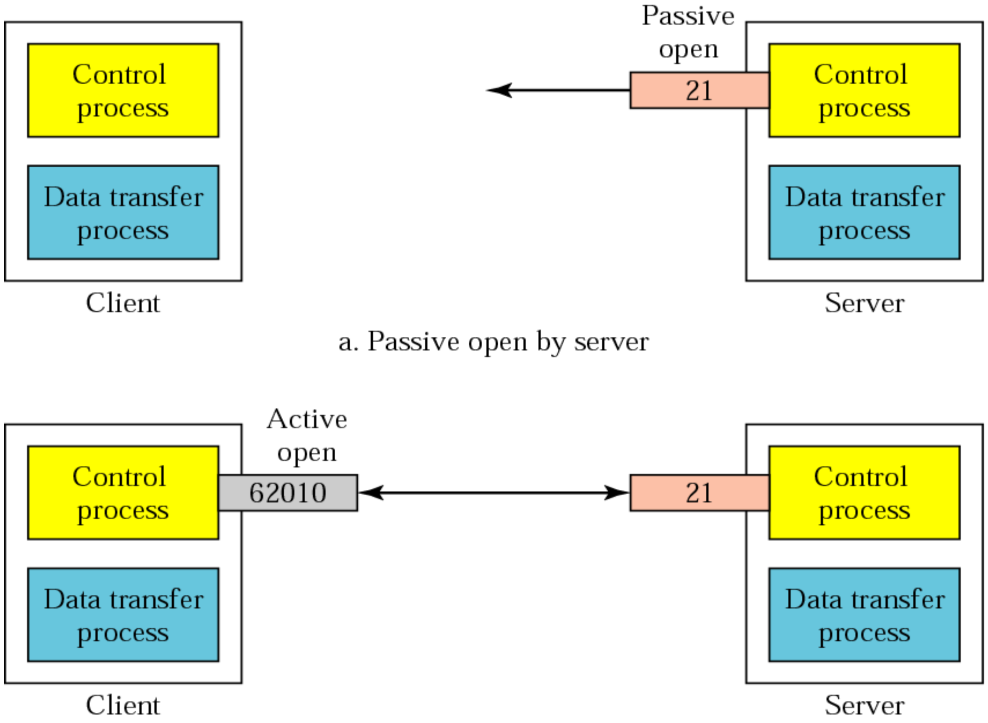
- File Transfer Protocol (FTP) is the standard mechanism provided by TCP/IP for copying a file from one host to another

FTP

- FTP uses the services of TCP
- It needs two TCP connections
- The well-known port 21 is used for the control connection and the well-known port 20 for the data connection



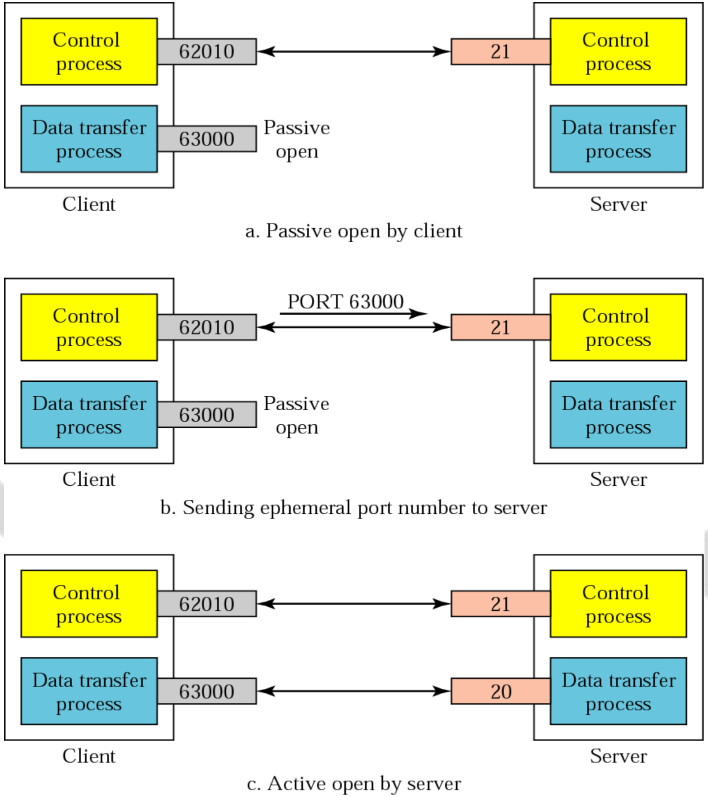
Opening the control connection



a. Passive open by server

b. Active open by client

Creating the data connection

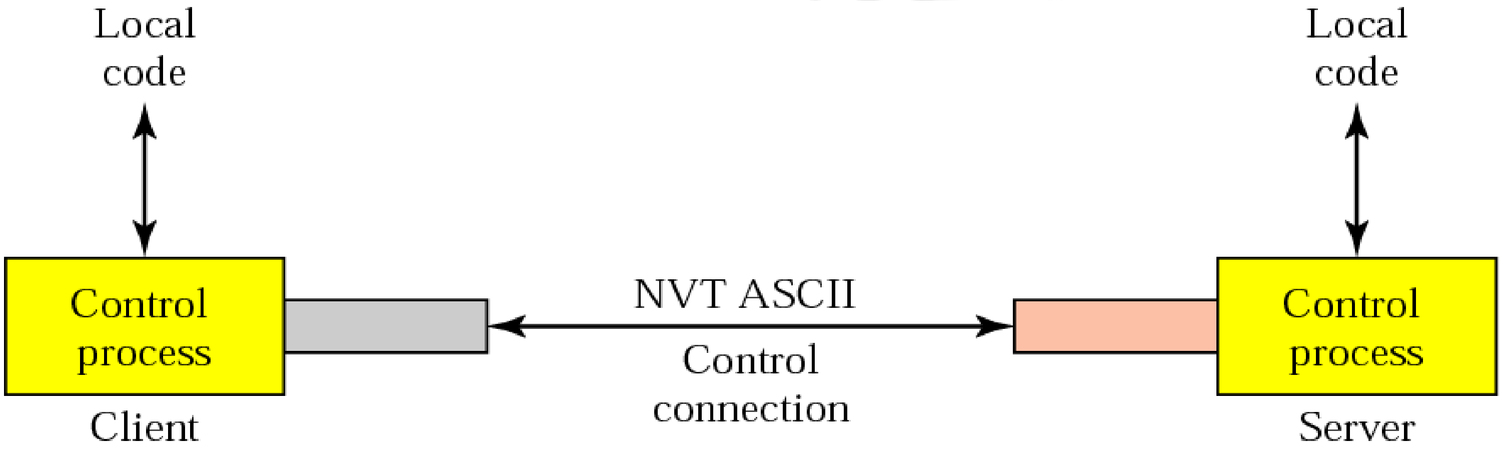


a. Passive open by client

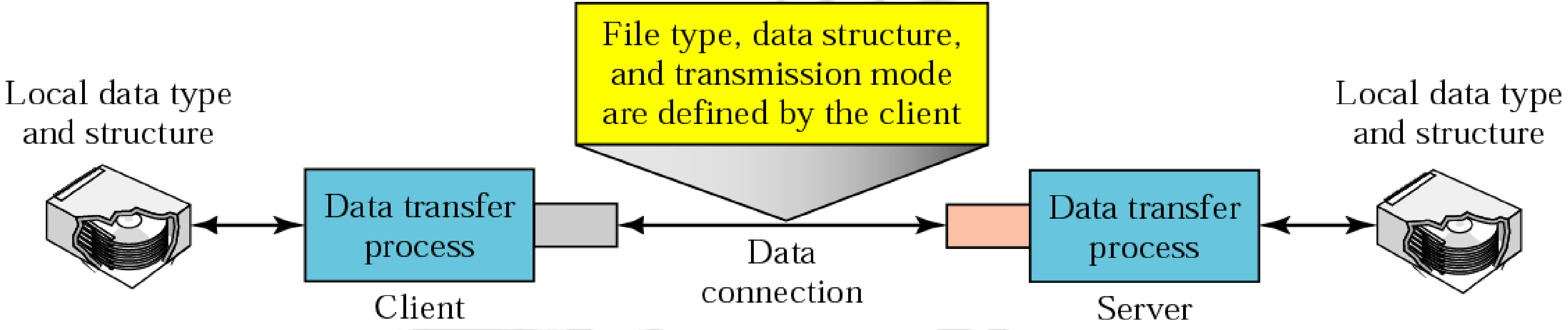
b. Sending ephemeral port number to server

c. Active open by server

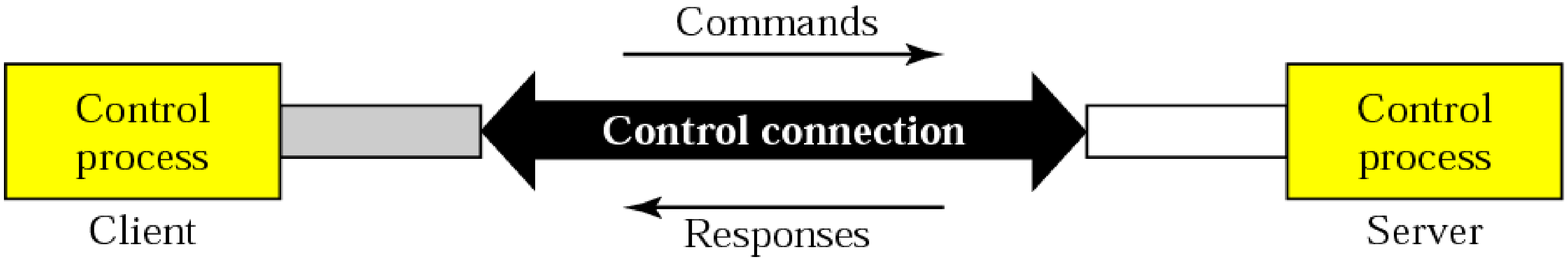
Using the control connection



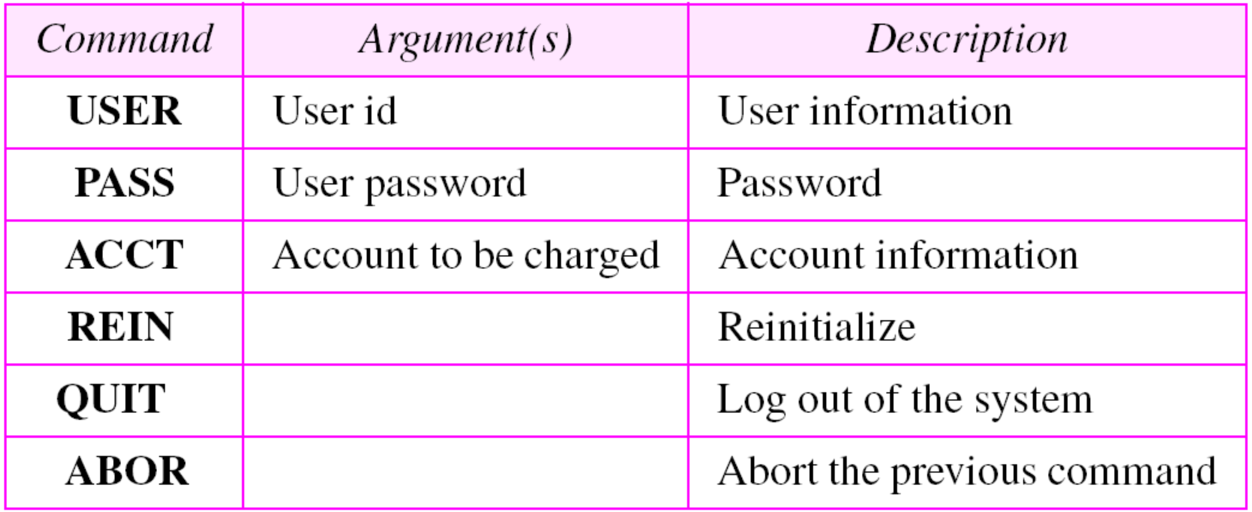
Using the data connection


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

Command processing

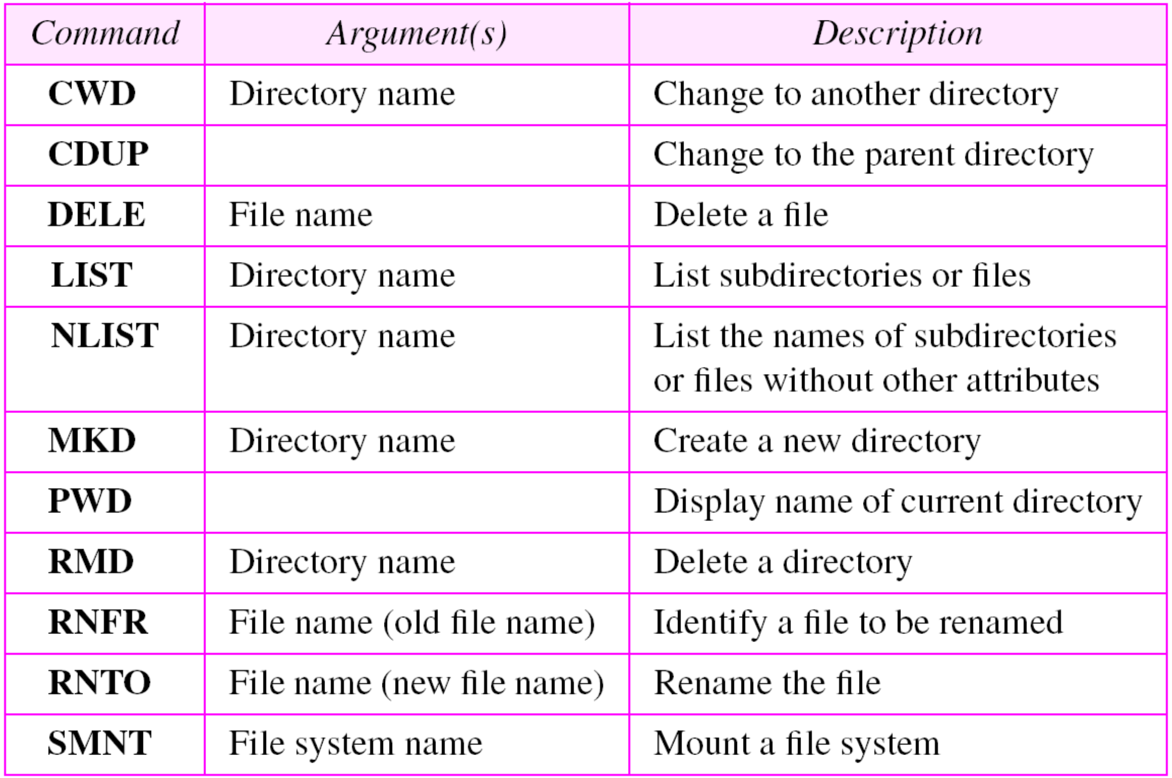


Access commands



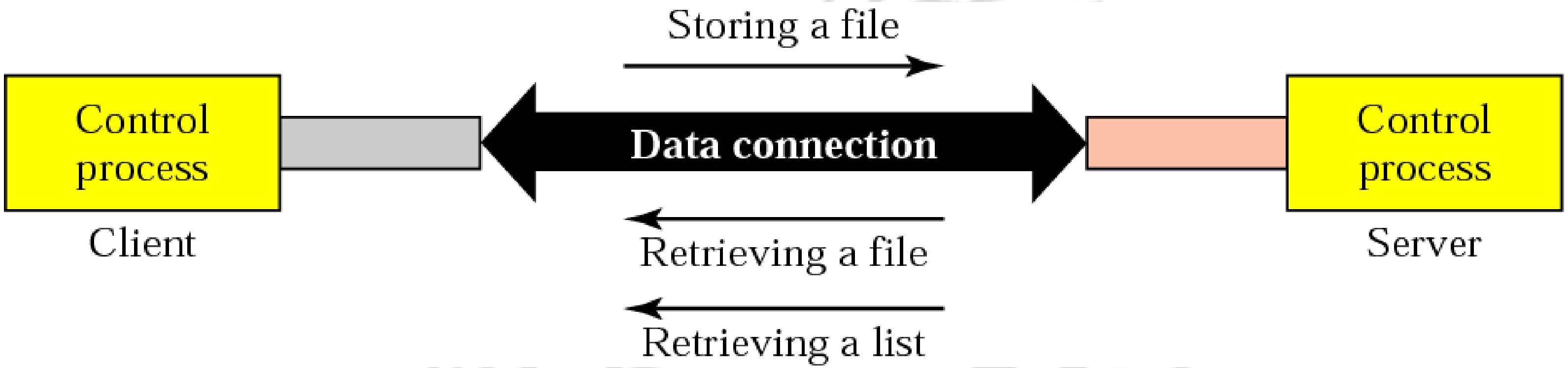
Command	*Argument(s)*	*Description*
USER	User id	User information
PASS	User password	Password
ACCT	Account to be charged	Account information
REIN		Reinitialize
QUIT		Log out of the system
ABOR		Abort the previous command

File management commands

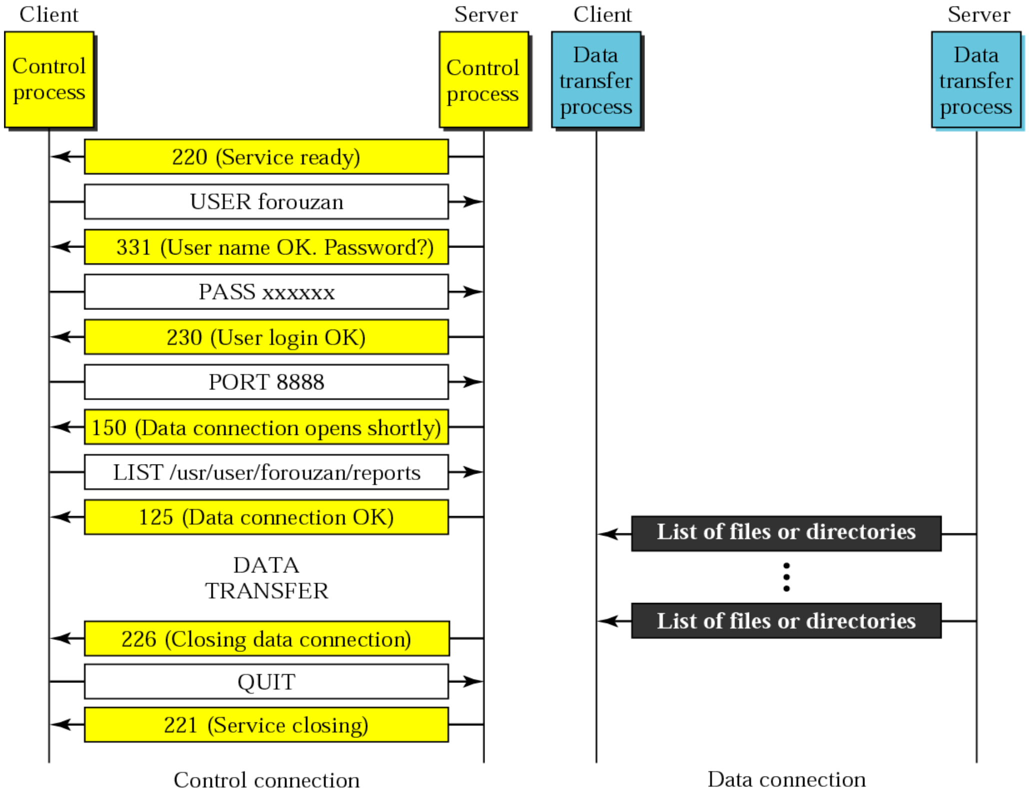


Command	*Argument(s)*	*Description*
CWD	Directory name	Change to another directory
CDUP		Change to the parent directory
DELE	File name	Delete a file
LIST	Directory name	List subdirectories or files
NLIST	Directory name	List the names of subdirectories or files without other attributes
MKD	Directory name	Create a new directory
PWD		Display name of current directory
RMD	Directory name	Delete a directory
RNFR	File name (old file name)	Identify a file to be renamed
RNTO	File name (new file name)	Rename the file
SMNT	File system name	Mount a file system

File transfer



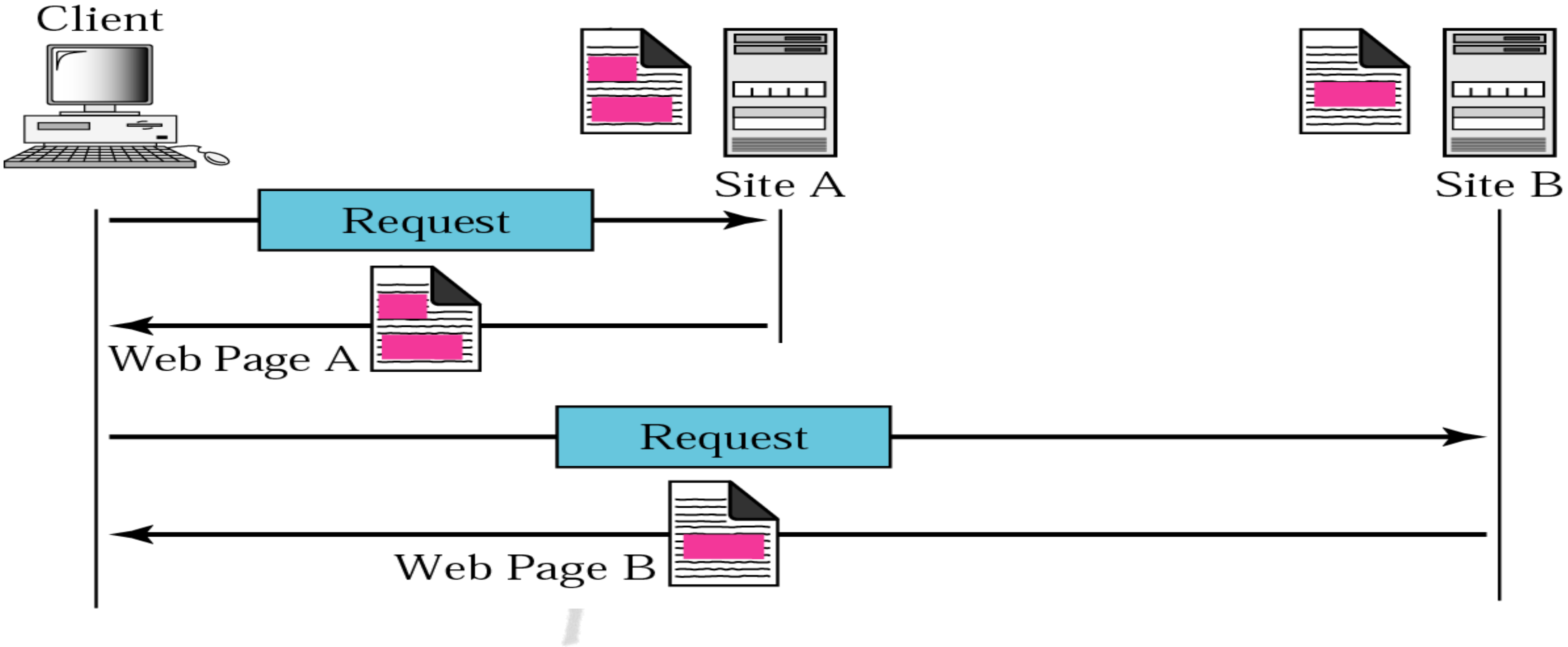
Example


Client
Control process
Server
Control process
Client
Data transfer process
Server
Data transfer process
220 (Service ready)
USER forouzan
331 (User name OK. Password?)
PASS xxxxxx
230 (User login OK)
PORT 8888
150 (Data connection opens shortly)
LIST /usr/user/forouzan/reports
125 (Data connection OK)
DATA TRANSFER
226 (Closing data connection)
QUIT
221 (Service closing)
List of files or directories
List of files or directories
Control connection
Data connection

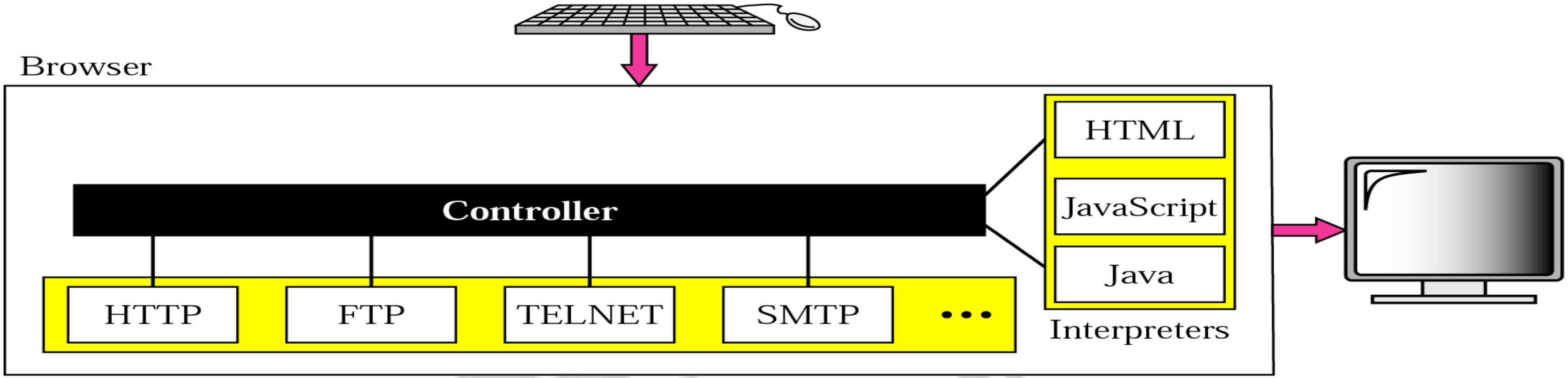
WWW

- The WWW is a distributed client-server service, in which a client using a browser can access a service using a server
- The service provided is distributed over many locations called sites

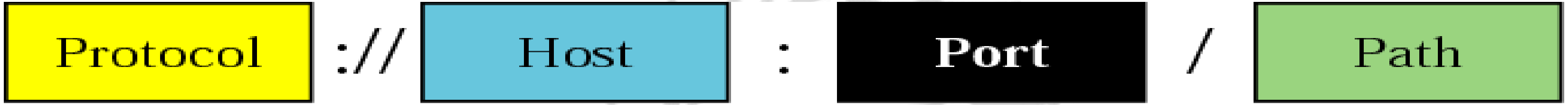
Architecture of WWW



Browser


Browser
Controller
HTTP
FTP
TELNET
SMTP
• • •
HTML
JavaScript
Java
Interpreters

URL

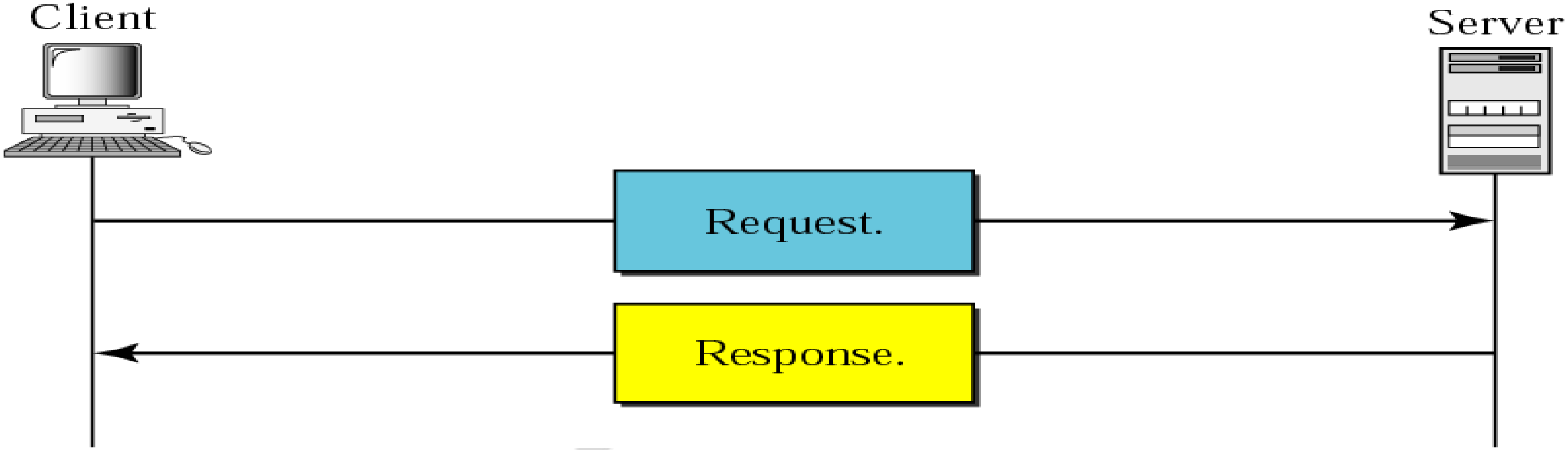


Protocol :// Host : Port / Path

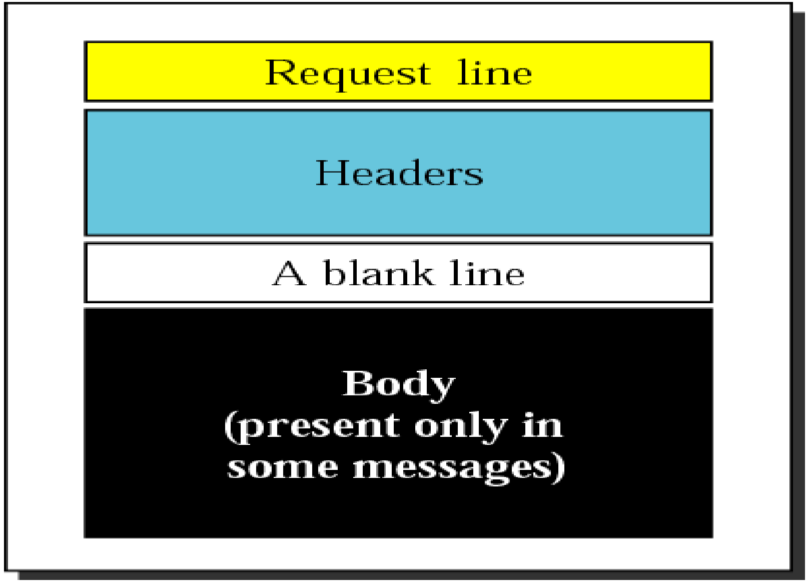
WWW

- The Hypertext Transfer Protocol (HTTP) is a protocol used mainly to access data on the World Wide Web
- HTTP functions like a combination of FTP and SMTP
- HTTP uses the services of TCP on well-known port 80

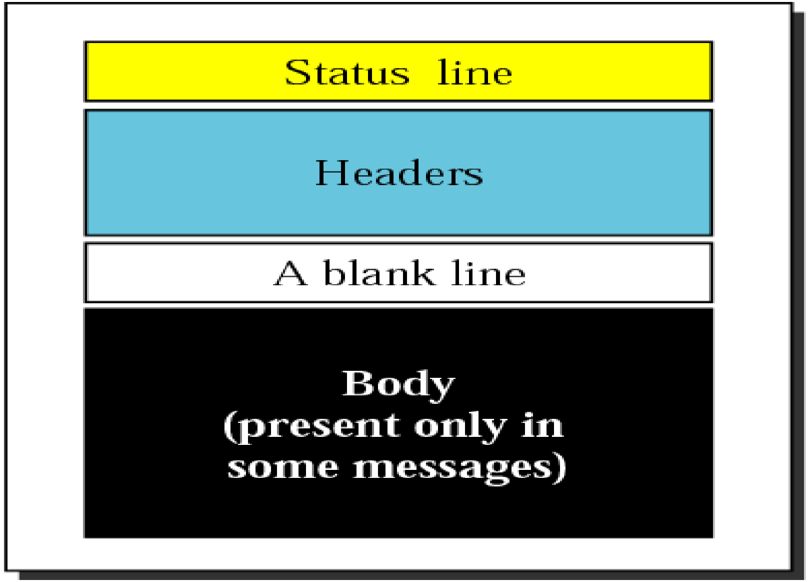
HTTP transaction



Request and response messages

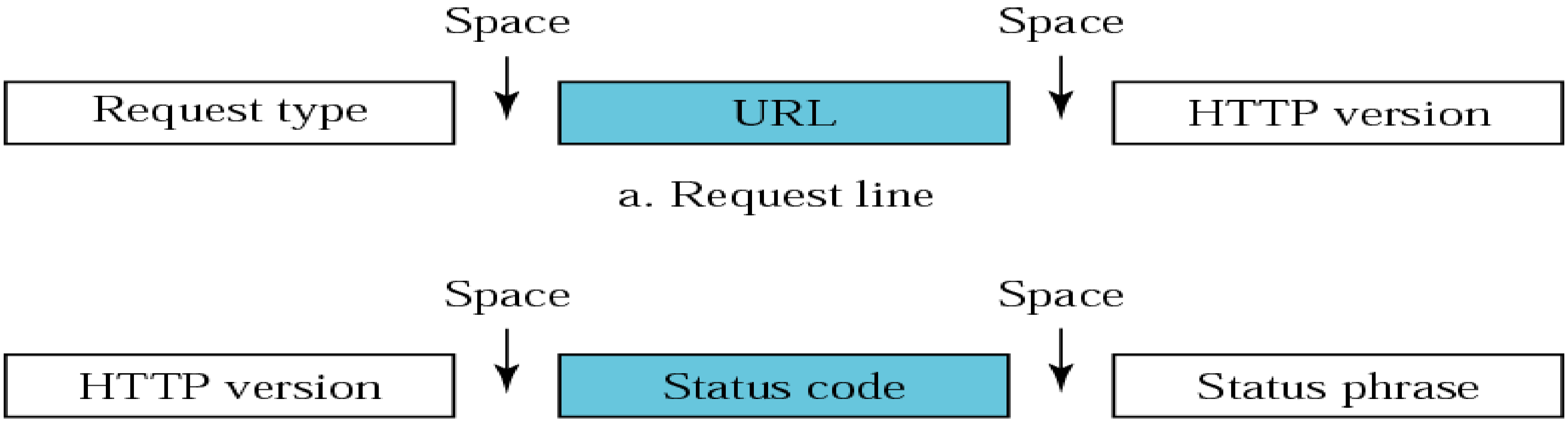


Request message



Response message

Request and status lines



Space

Space

Request type

URL

HTTP version

a. Request line

Space

Space

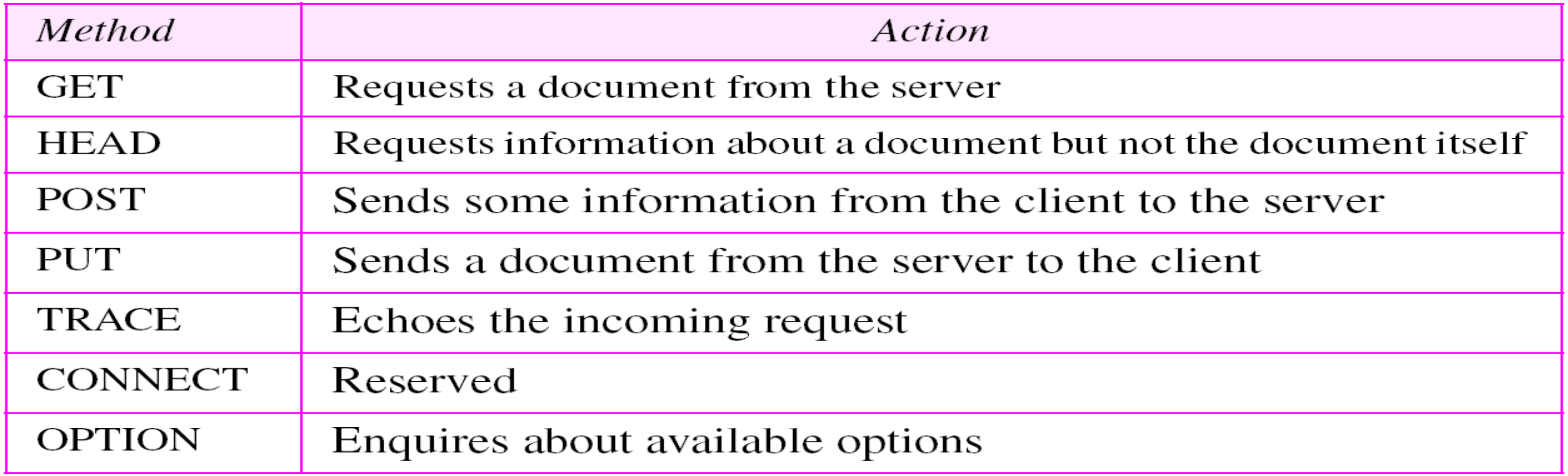
HTTP version

Status code

Status phrase

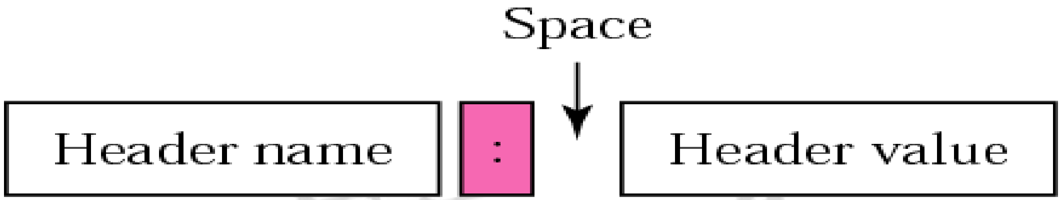
b. Status line

Methods



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

Header format



Space

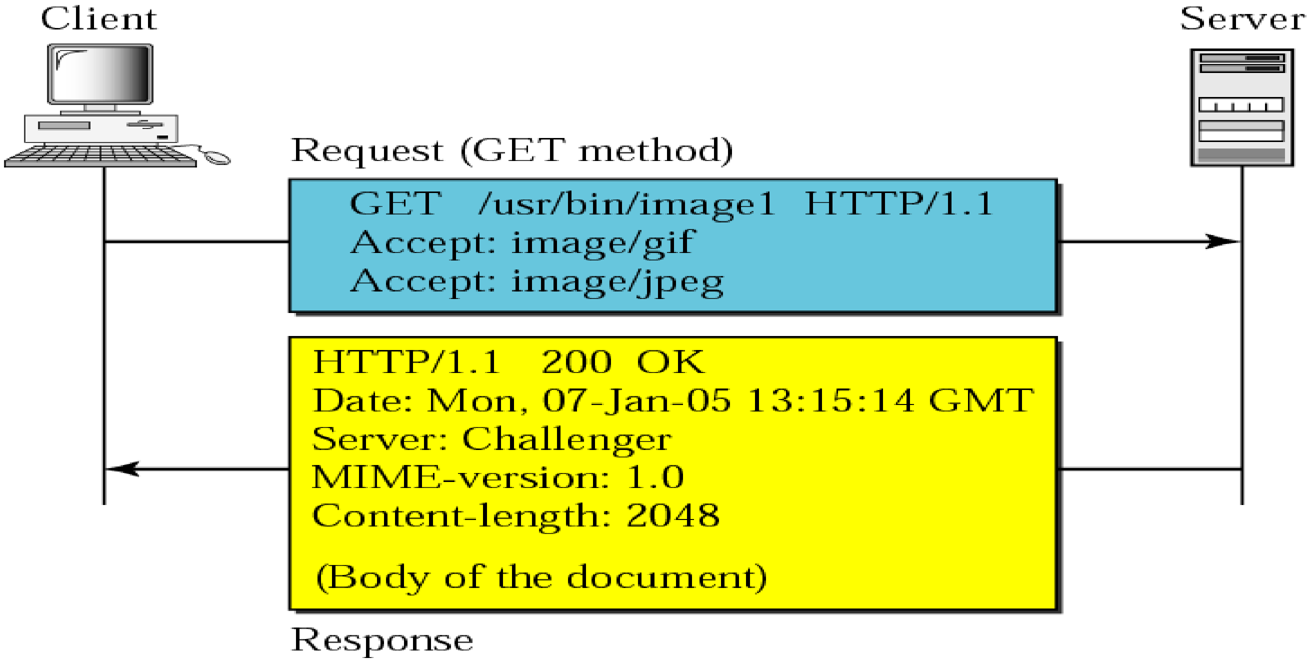
Header name

:

Header value

Example

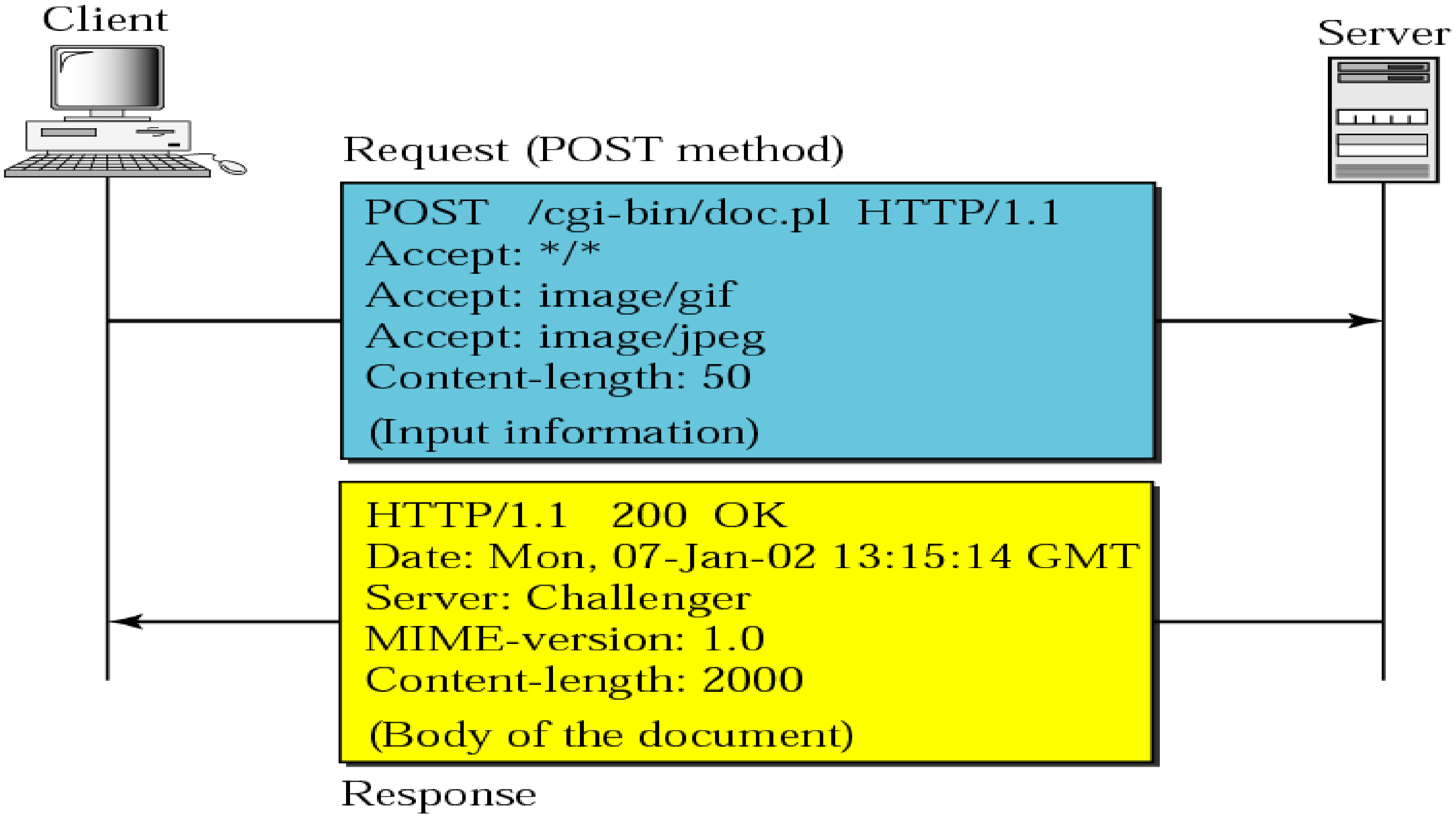
- HTTP version 1.1 specifies a persistent connection by default



```
GET   /usr/bin/image1   HTTP/1.1
Accept: image/gif
Accept: image/jpeg
```

```
HTTP/1.1   200  OK
Date: Mon, 07-Jan-05 13:15:14 GMT
Server: Challenger
MIME-version: 1.0
Content-length: 2048

(Body of the document)
```

Example



```
POST   /cgi-bin/doc.pl  HTTP/1.1
Accept: */*
Accept: image/gif
Accept: image/jpeg
Content-length: 50

(Input information)
```

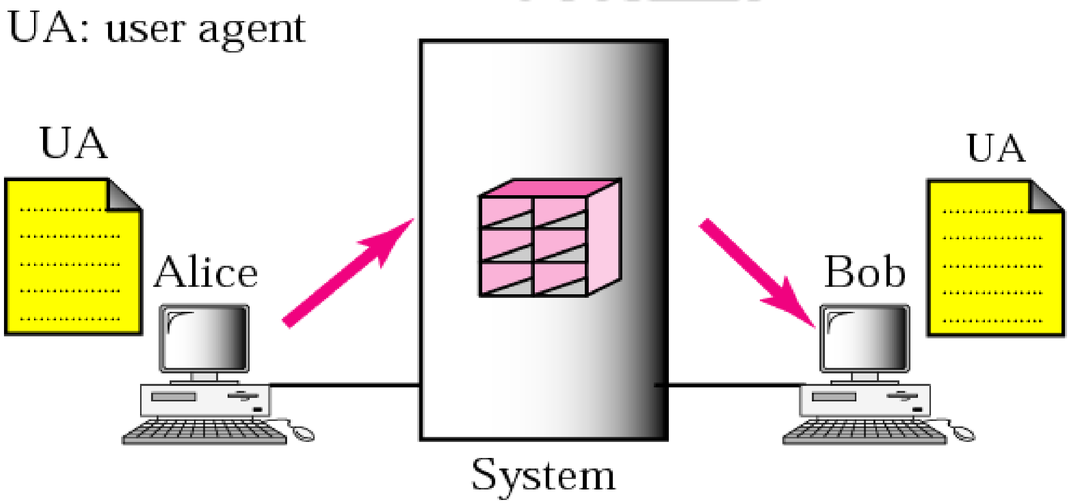
```
HTTP/1.1   200  OK
Date: Mon, 07-Jan-02 13:15:14 GMT
Server: Challenger
MIME-version: 1.0
Content-length: 2000

(Body of the document)
```

SMTP

- To explain the architecture of email, we give few 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

- When the sender and the receiver of an email are on the same system, we need only two user agents

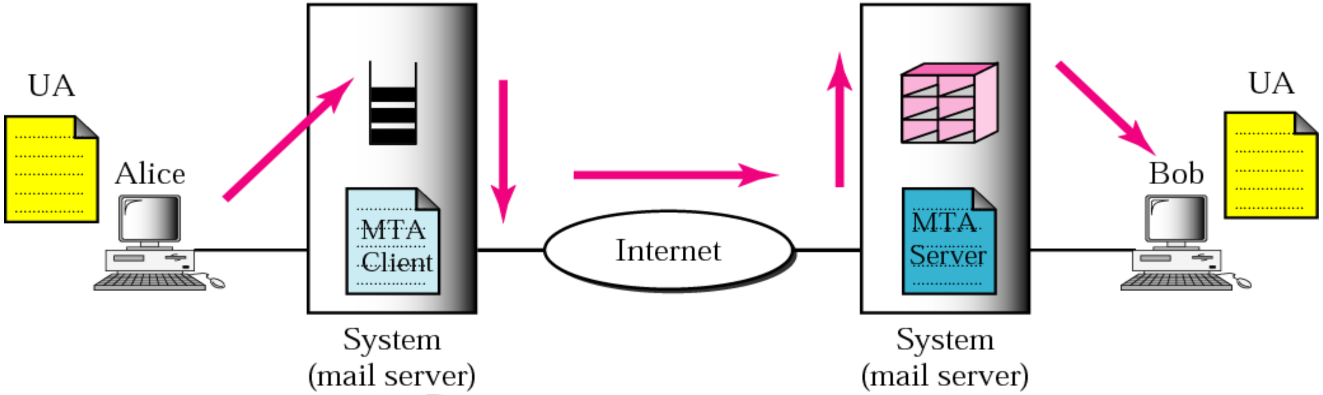


Second scenario

- When the sender and the receiver of an email are on different systems, we need two UAs and a pair of MTAs (client and server)

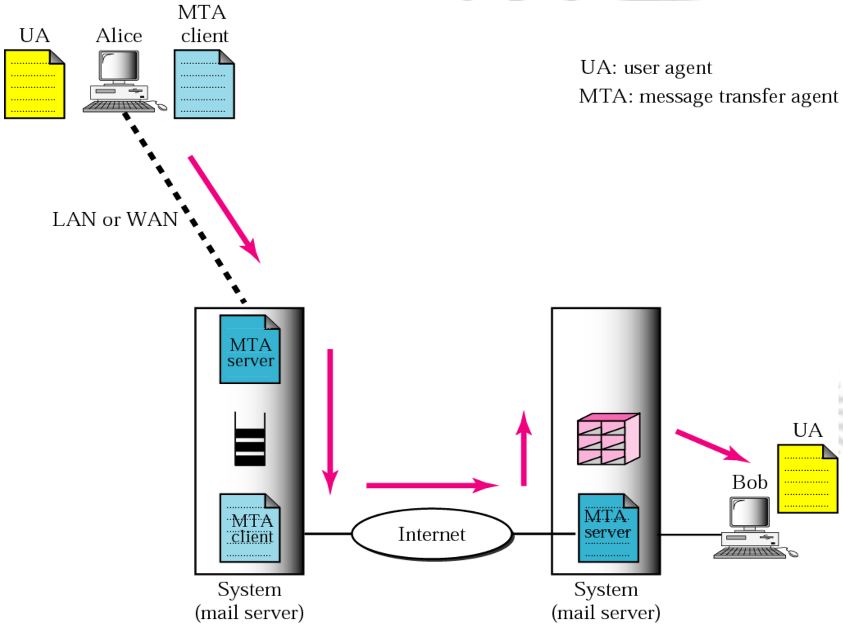
UA: user agent

MTA: message transfer agent



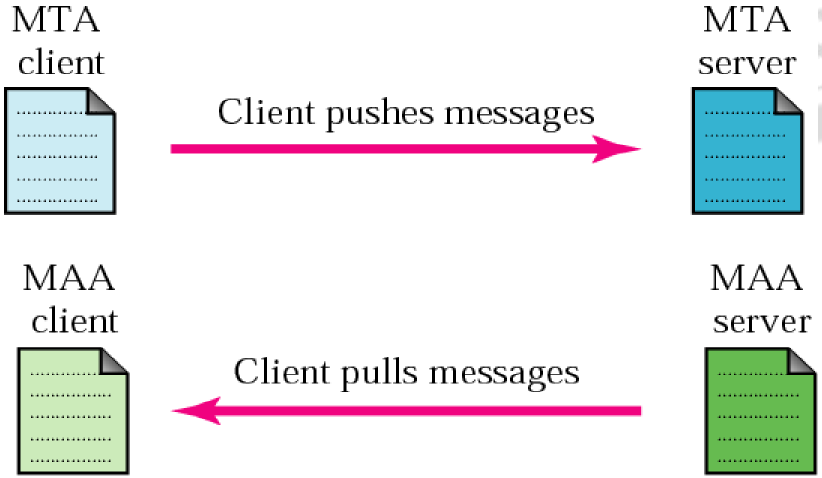
Third scenario

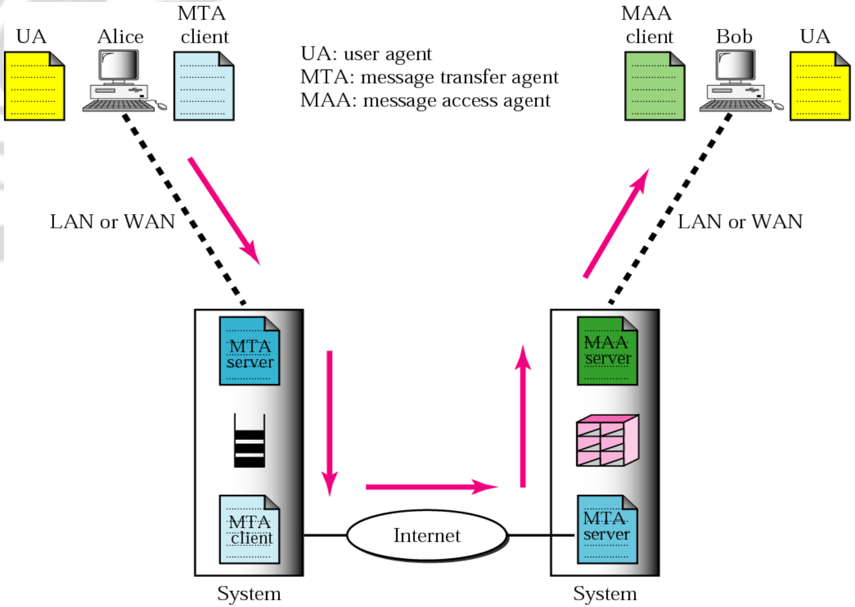
- When the sender is connected to the mail server via a LAN or a WAN, we need two UAs and two pairs of MTAs (client and server)



Fourth scenario

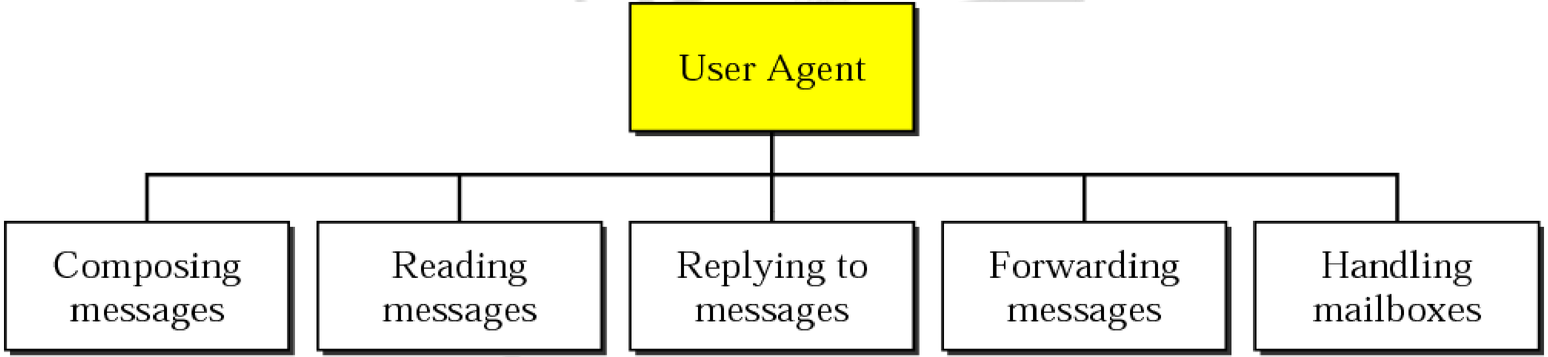
- When both sender and receiver are connected to the mail server via a LAN or a WAN, we need two UAs, two pairs of MTAs (client and server), and a pair of MAAs (client and server)
- This is the most common situation



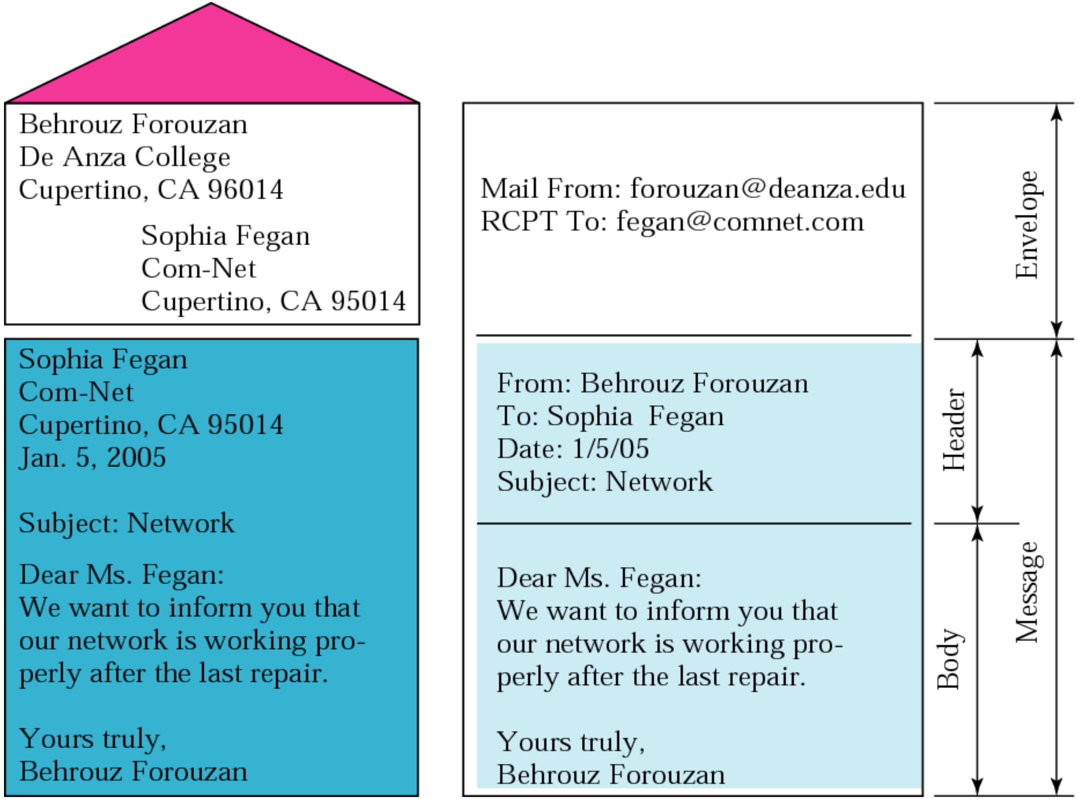


User Agent

- The user agent (UA) provides service to the user to make the process of sending and receiving a message easier
- Some examples of command-driven user agents are mail, pine, and elm
- Some examples of GUI-based user agents are Eudora, Outlook, and Netscape



Format of an email



Behrouz Forouzan
De Anza College
Cupertino, CA 96014

Sophia Fegan
Com-Net
Cupertino, CA 95014

Sophia Fegan
Com-Net
Cupertino, CA 95014
Jan. 5, 2005

Subject: Network

Dear Ms. Fegan:
We want to inform you that our network is working properly after the last repair.

Yours truly,
Behrouz Forouzan

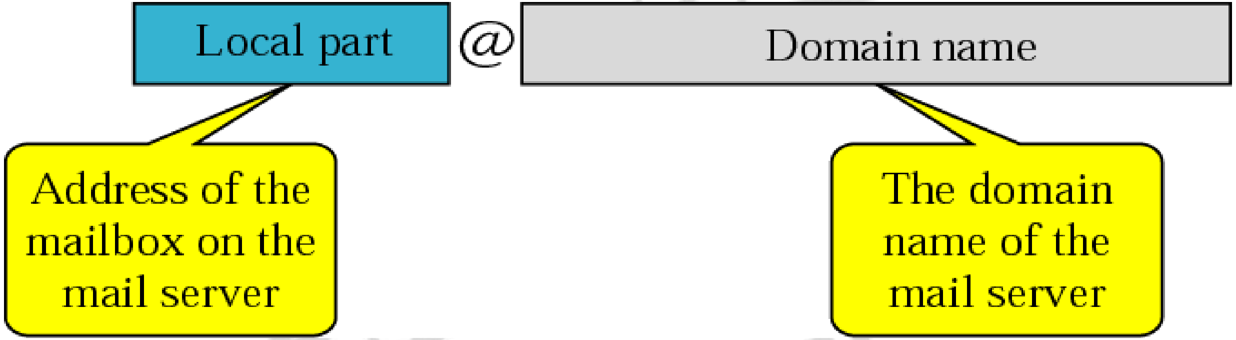
Mail From: forouzan@deanza.edu
RCPT To: fegan@comnet.com

From: Behrouz Forouzan
To: Sophia Fegan
Date: 1/5/05
Subject: Network

Dear Ms. Fegan:
We want to inform you that our network is working properly after the last repair.

Yours truly,
Behrouz Forouzan

Email address



Local part

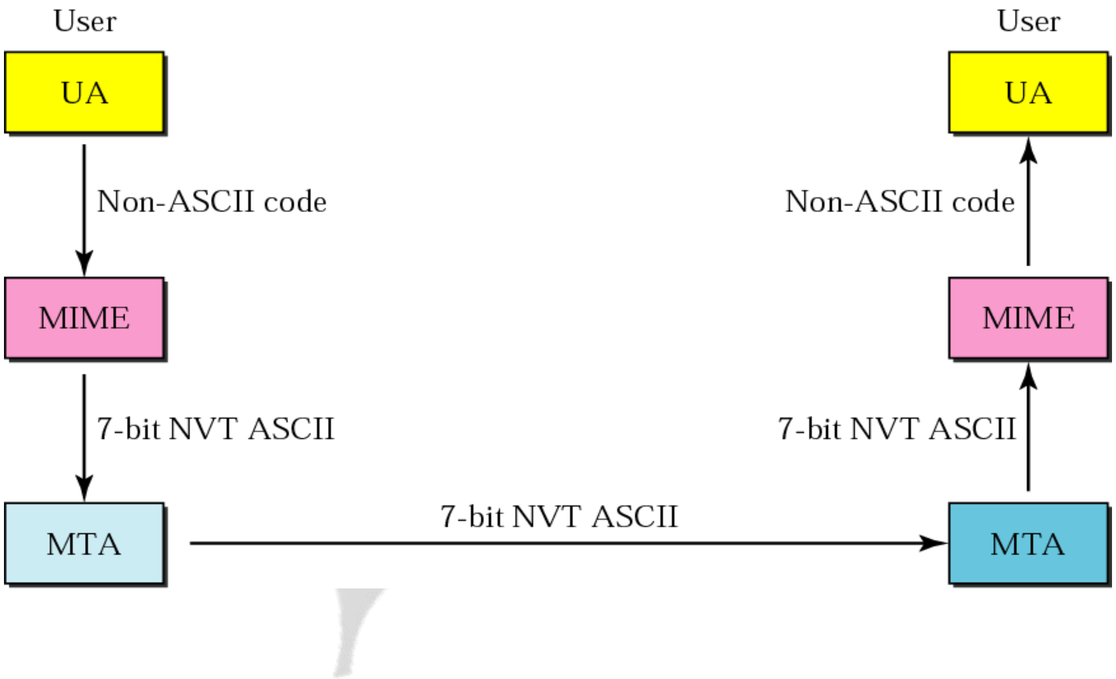
@

Domain name

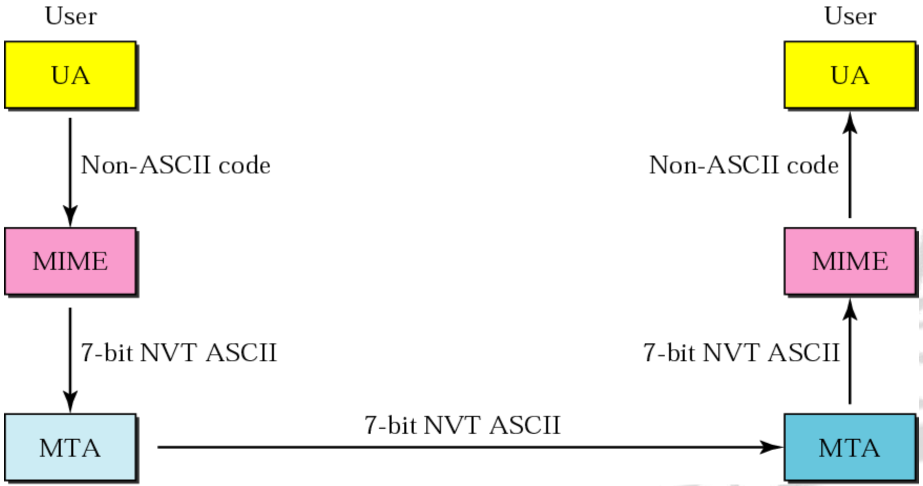
Address of the mailbox on the mail server

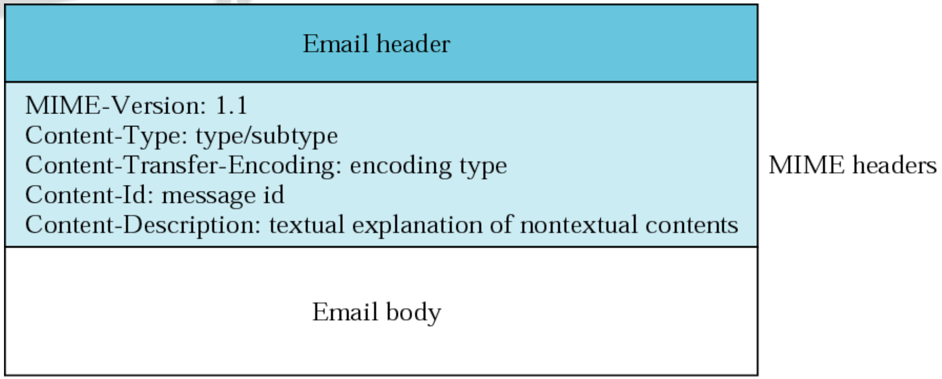
The domain name of the mail server

MIME


User
UA
Non-ASCII code
MIME
7-bit NVT ASCII
MTA
7-bit NVT ASCII
User
UA
Non-ASCII code
MIME
7-bit NVT ASCII
MTA

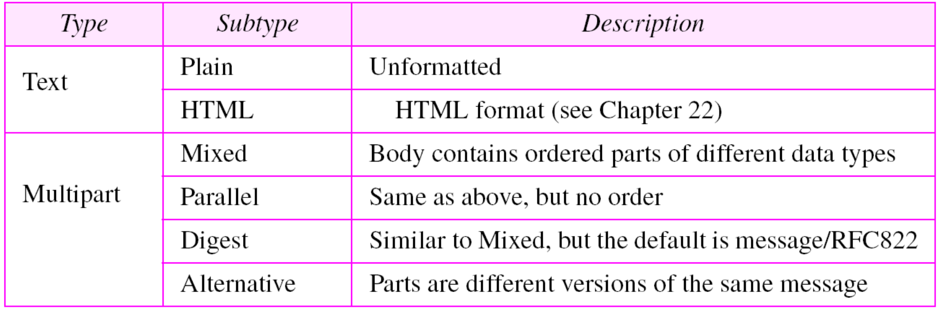
MIME



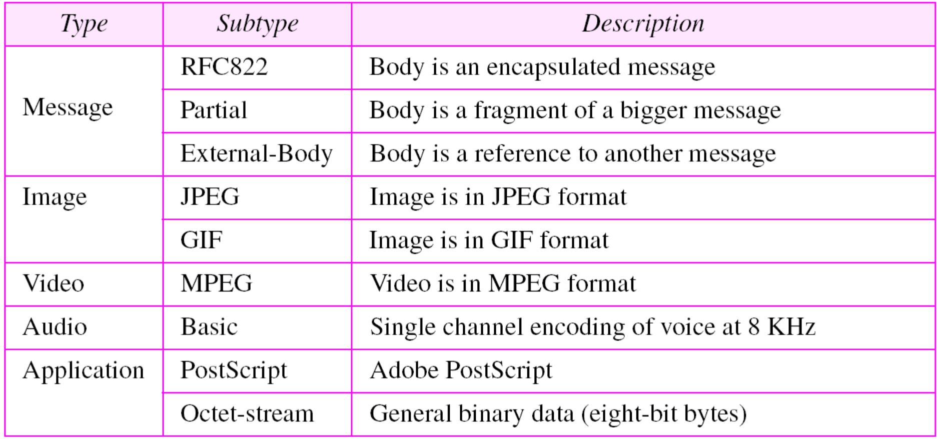


MIME-Version: 1.1
Content-Type: type/subtype
Content-Transfer-Encoding: encoding type
Content-Id: message id
Content-Description: textual explanation of nontextual contents

Data types in sub types in MIME

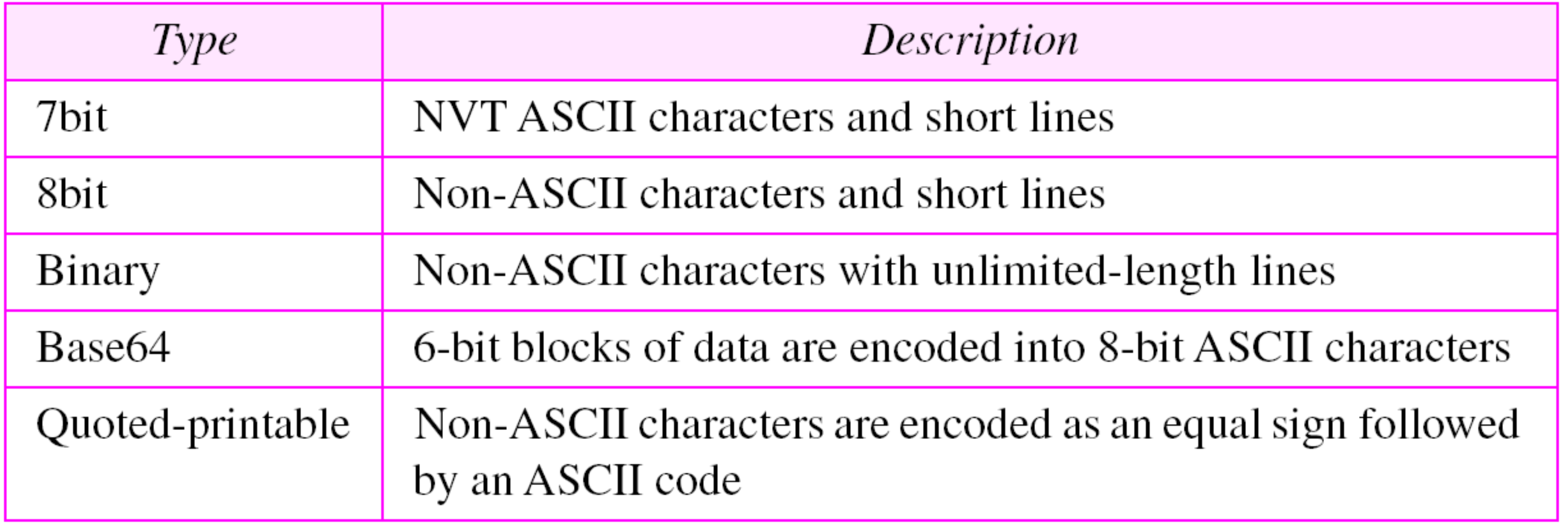


Type	*Subtype*	*Description*
Text	Plain	Unformatted
	HTML	HTML format (see Chapter 22)
Multipart	Mixed	Body contains ordered parts of different data types
	Parallel	Same as above, but no order
	Digest	Similar to Mixed, but the default is message/RFC822
	Alternative	Parts are different versions of the same message



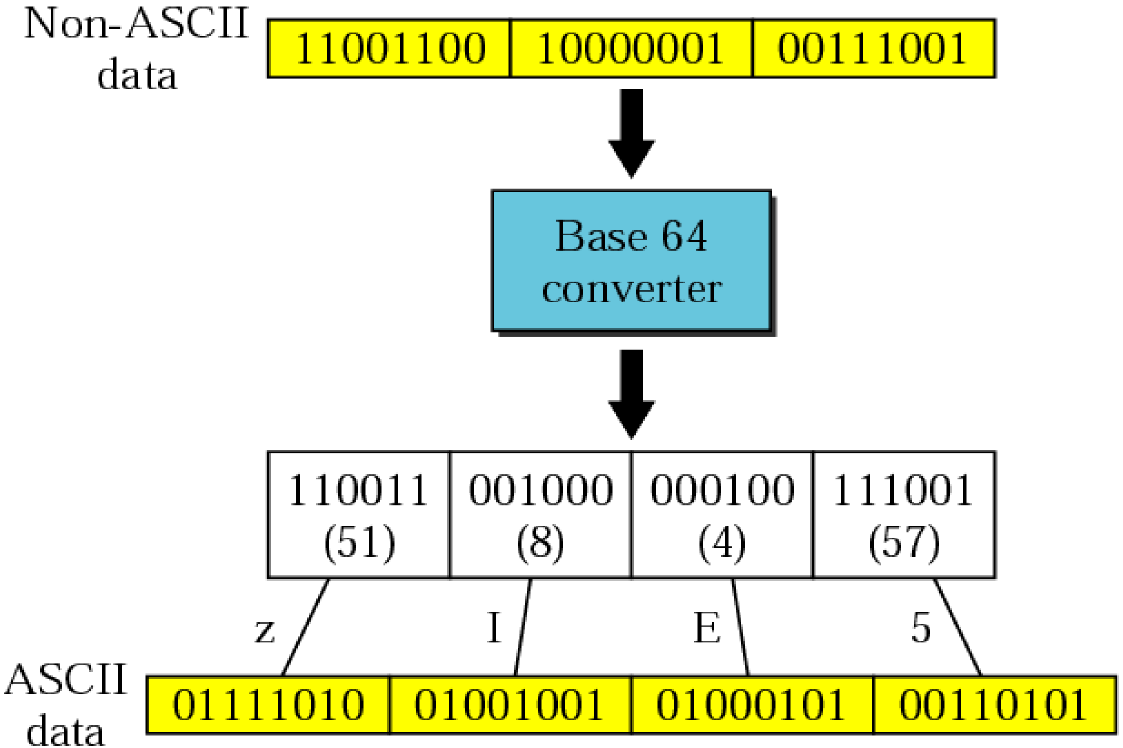
Type	*Subtype*	*Description*
Message	RFC822	Body is an encapsulated message
	Partial	Body is a fragment of a bigger message
	External-Body	Body is a reference to another message
Image	JPEG	Image is in JPEG format
	GIF	Image is in GIF format
Video	MPEG	Video is in MPEG format
Audio	Basic	Single channel encoding of voice at 8 KHz
Application	PostScript	Adobe PostScript
	Octet-stream	General binary data (eight-bit bytes)

Content transfer encoding

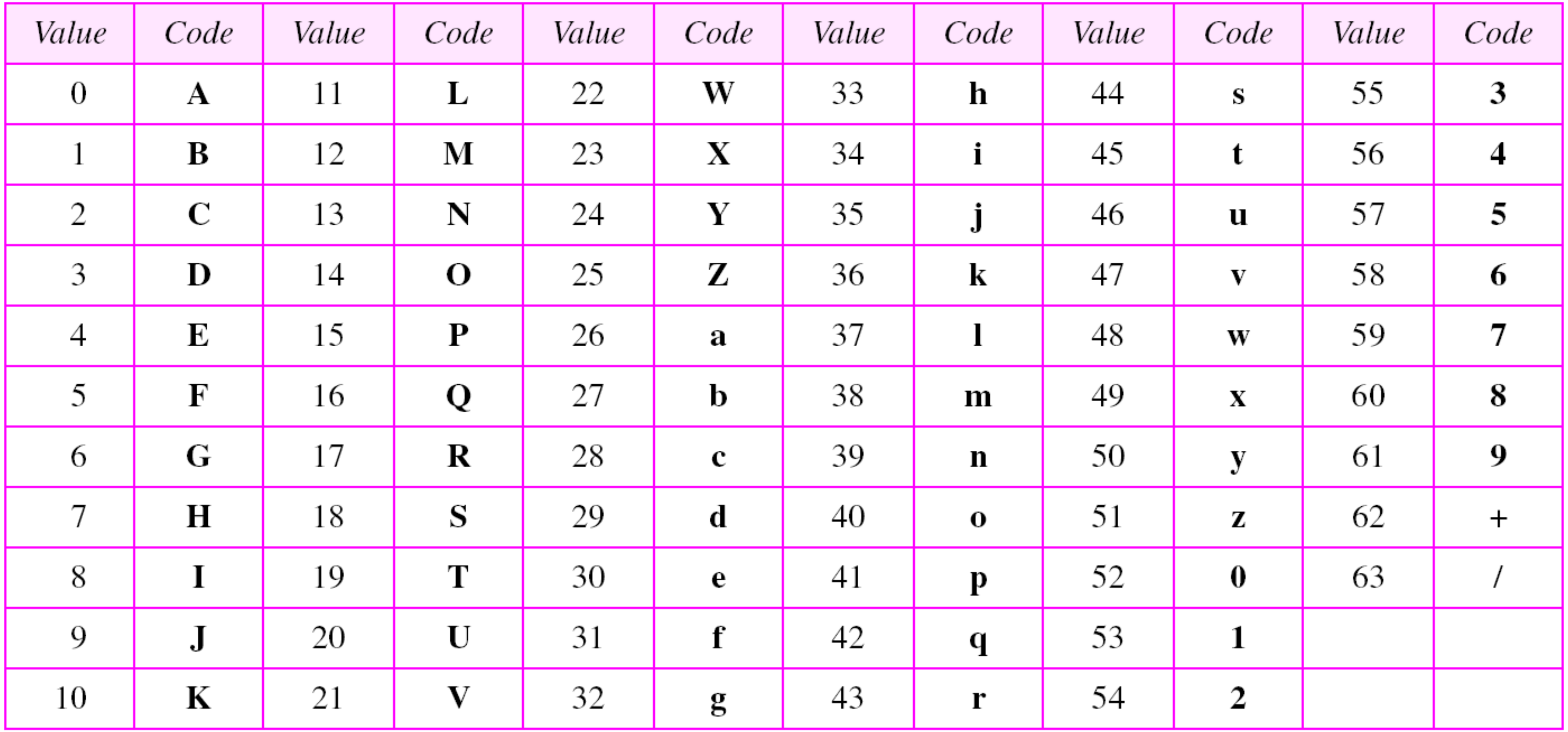


Type	*Description*
7bit	NVT ASCII characters and short lines
8bit	Non-ASCII characters and short lines
Binary	Non-ASCII characters with unlimited-length lines
Base64	6-bit blocks of data are encoded into 8-bit ASCII characters
Quoted-printable	Non-ASCII characters are encoded as an equal sign followed by an ASCII code

Base64

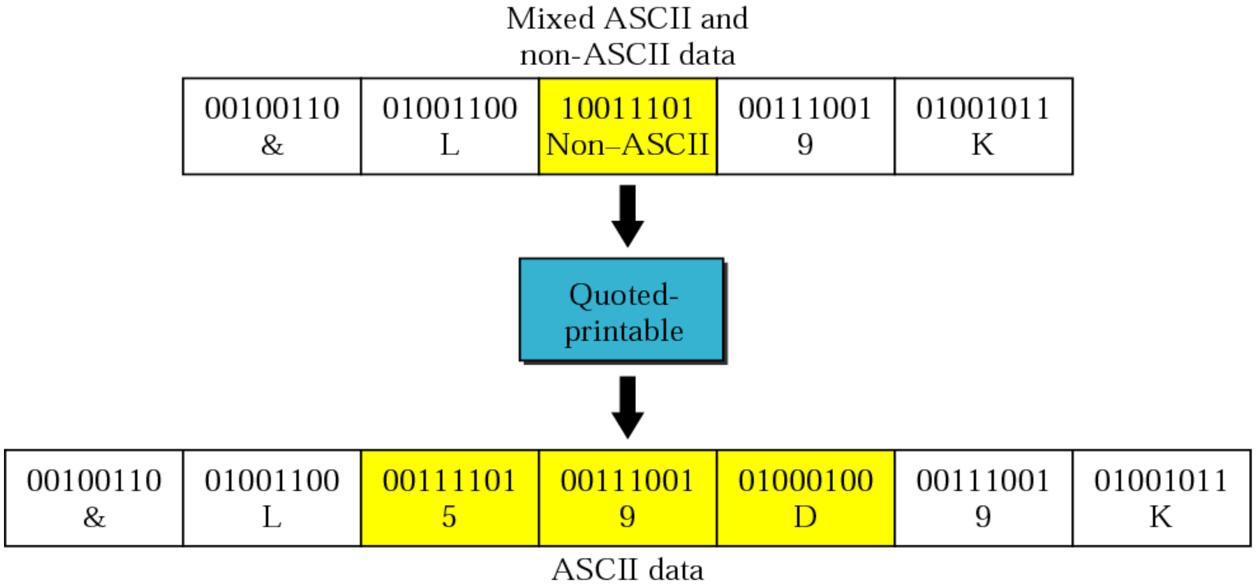

Non-ASCII data
11001100
10000001
00111001
Base 64 converter
110011 (51)
001000 (8)
000100 (4)
111001 (57)
z
I
E
5
ASCII data
01111010
01001001
01000101
00110101

Base64 encoding table



Value	Code	Value	Code	Value	Code	Value	Code	Value	Code	Value	Code
0	**A**	11	**L**	22	**W**	33	**h**	44	**s**	55	**3**
1	**B**	12	**M**	23	**X**	34	**i**	45	**t**	56	**4**
2	**C**	13	**N**	24	**Y**	35	**j**	46	**u**	57	**5**
3	**D**	14	**O**	25	**Z**	36	**k**	47	**v**	58	**6**
4	**E**	15	**P**	26	**a**	37	**l**	48	**w**	59	**7**
5	**F**	16	**Q**	27	**b**	38	**m**	49	**x**	60	**8**
6	**G**	17	**R**	28	**c**	39	**n**	50	**y**	61	**9**
7	**H**	18	**S**	29	**d**	40	**o**	51	**z**	62	+
8	**I**	19	**T**	30	**e**	41	**p**	52	**0**	63	/
9	**J**	20	**U**	31	**f**	42	**q**	53	**1**		
10	**K**	21	**V**	32	**g**	43	**r**	54	**2**		

Quoted-printable



Mixed ASCII and non-ASCII data

00100110 &	01001100 L	10011101 Non–ASCII	00111001 9	01001011 K

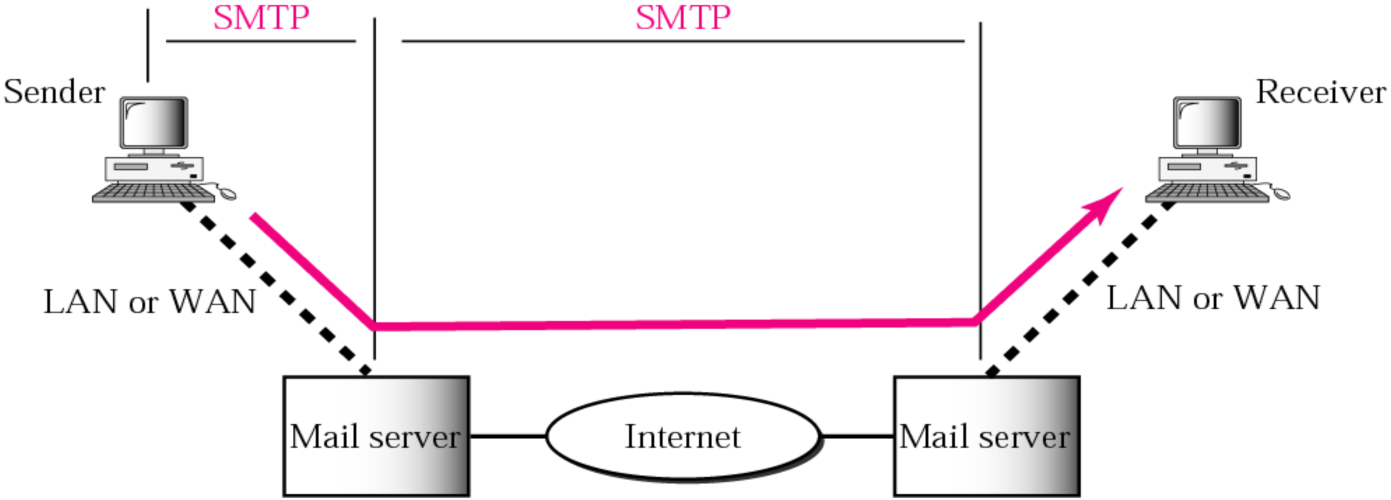
Quoted-printable

00100110 &	01001100 L	00111101 5	00111001 9	01000100 D	00111001 9	01001011 K

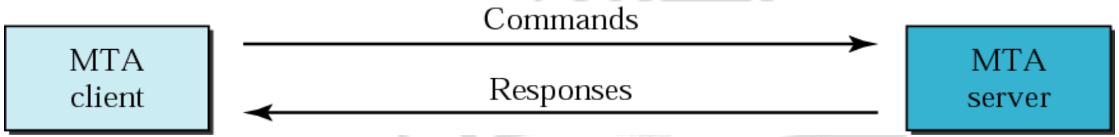
ASCII data

MTA

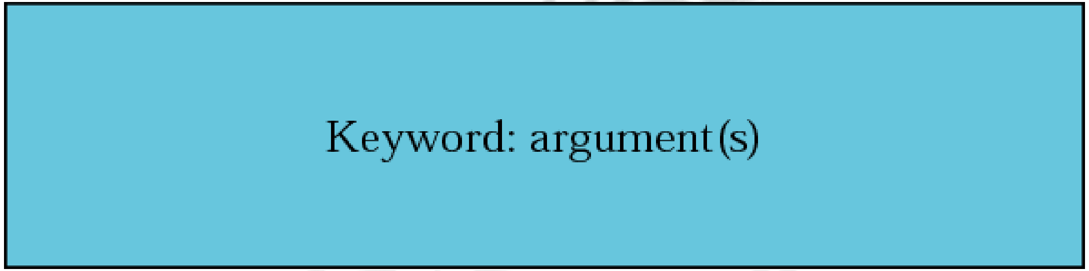
- The actual mail transfer requires message transfer agents (MTAs)
- The protocol that defines the MTA client and server in the Internet is called Simple Mail Transfer Protocol (SMTP)



Commands and responses

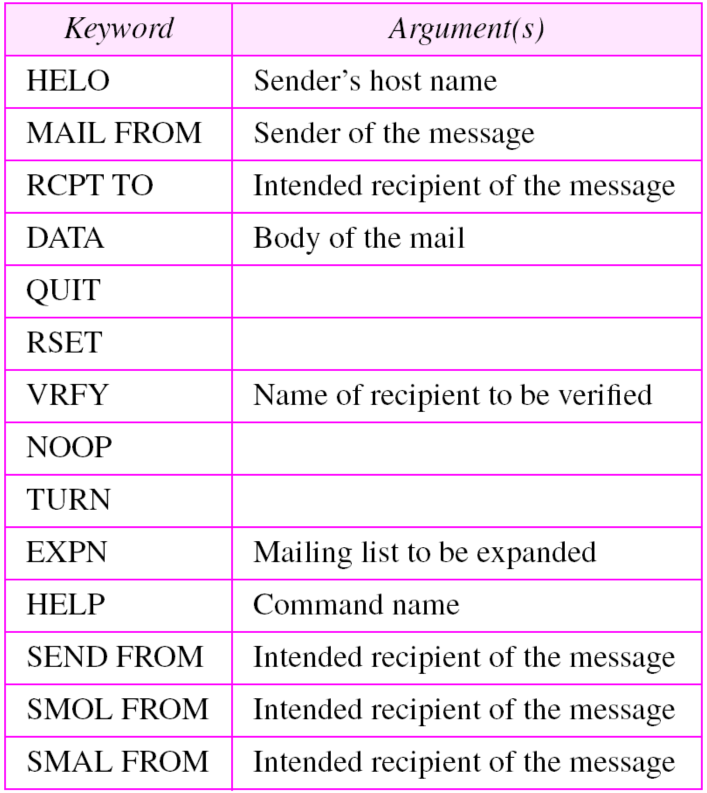


Command format



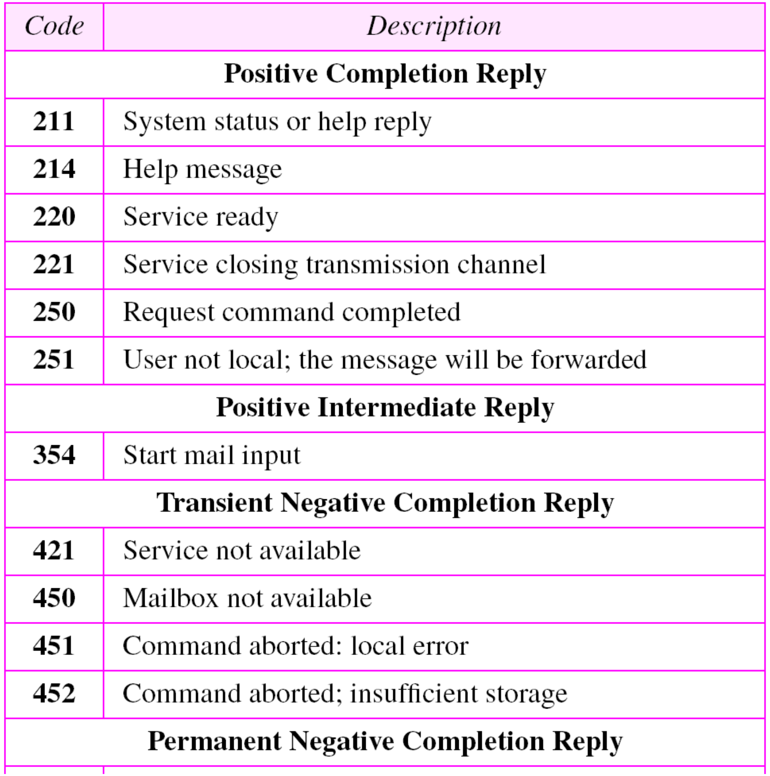
Keyword: argument(s)

Commands

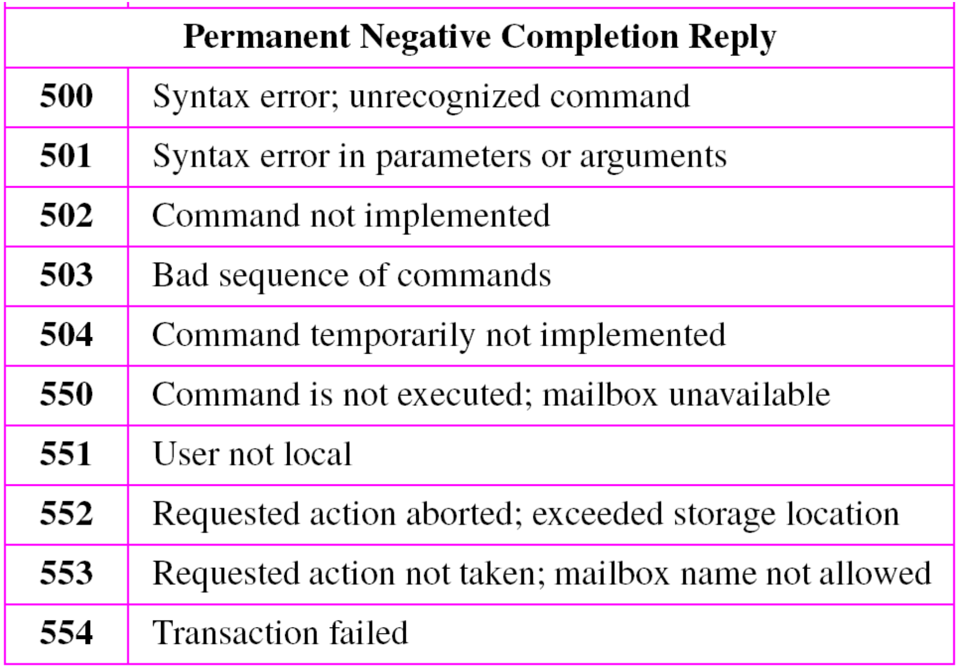


Keyword	*Argument(s)*
HELO	Sender’s host name
MAIL FROM	Sender of the message
RCPT TO	Intended recipient of the message
DATA	Body of the mail
QUIT	
RSET	
VRFY	Name of recipient to be verified
NOOP	
TURN	
EXPN	Mailing list to be expanded
HELP	Command name
SEND FROM	Intended recipient of the message
SMOL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

Responses

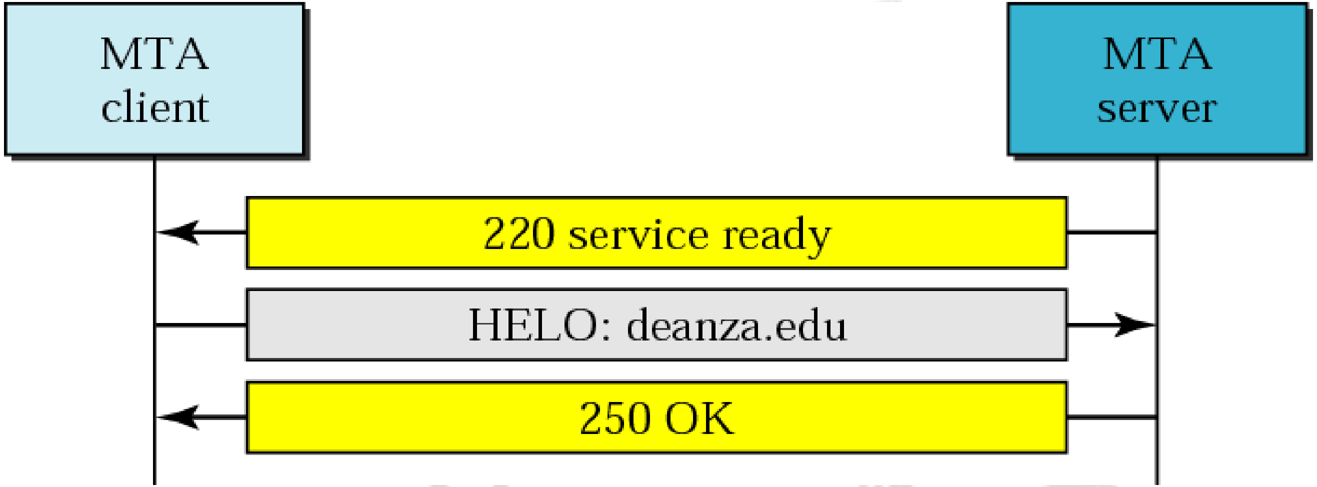


Code	*Description*
Positive Completion Reply	
211	System status or help reply
214	Help message
220	Service ready
221	Service closing transmission channel
250	Request command completed
251	User not local; the message will be forwarded
Positive Intermediate Reply	
354	Start mail input
Transient Negative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

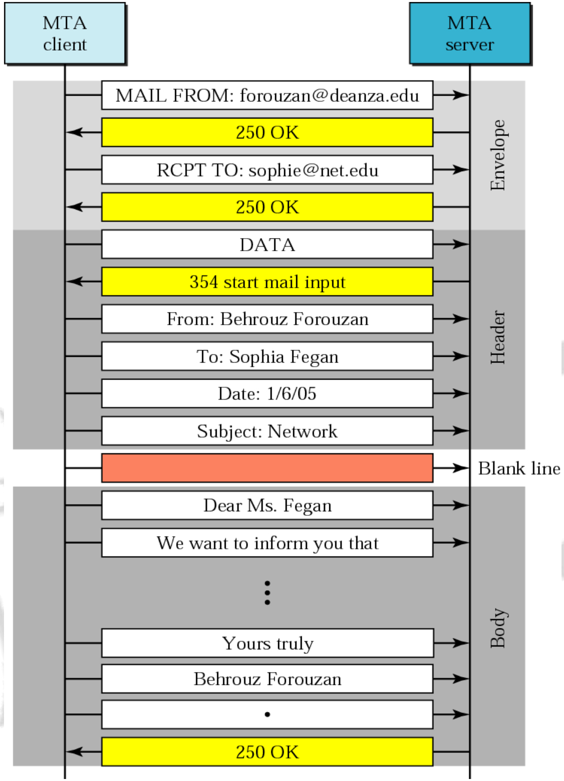


Permanent Negative Completion Reply	
500	Syntax error; unrecognized command
501	Syntax error in parameters or arguments
502	Command not implemented
503	Bad sequence of commands
504	Command temporarily not implemented
550	Command is not executed; mailbox unavailable
551	User not local
552	Requested action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

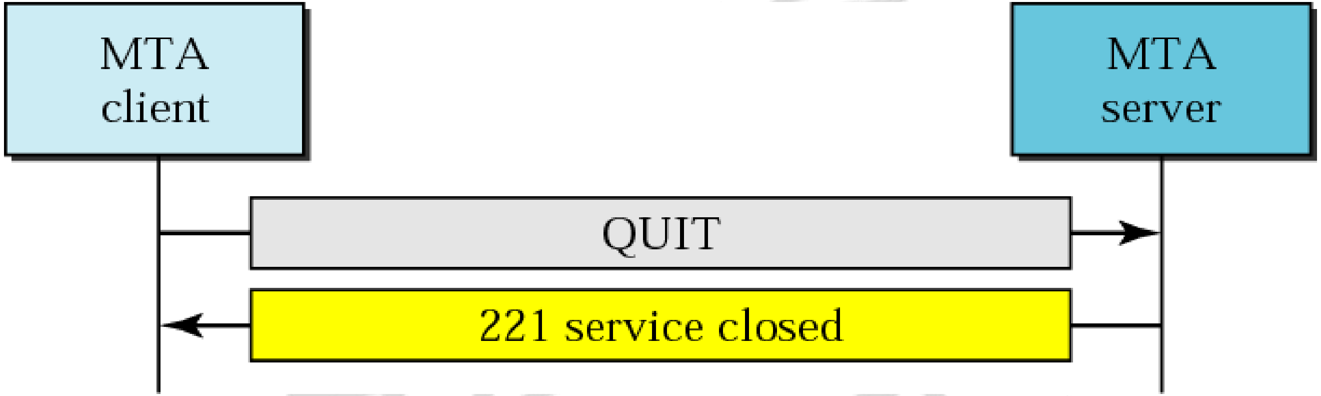
Connection establishment



Message transfer



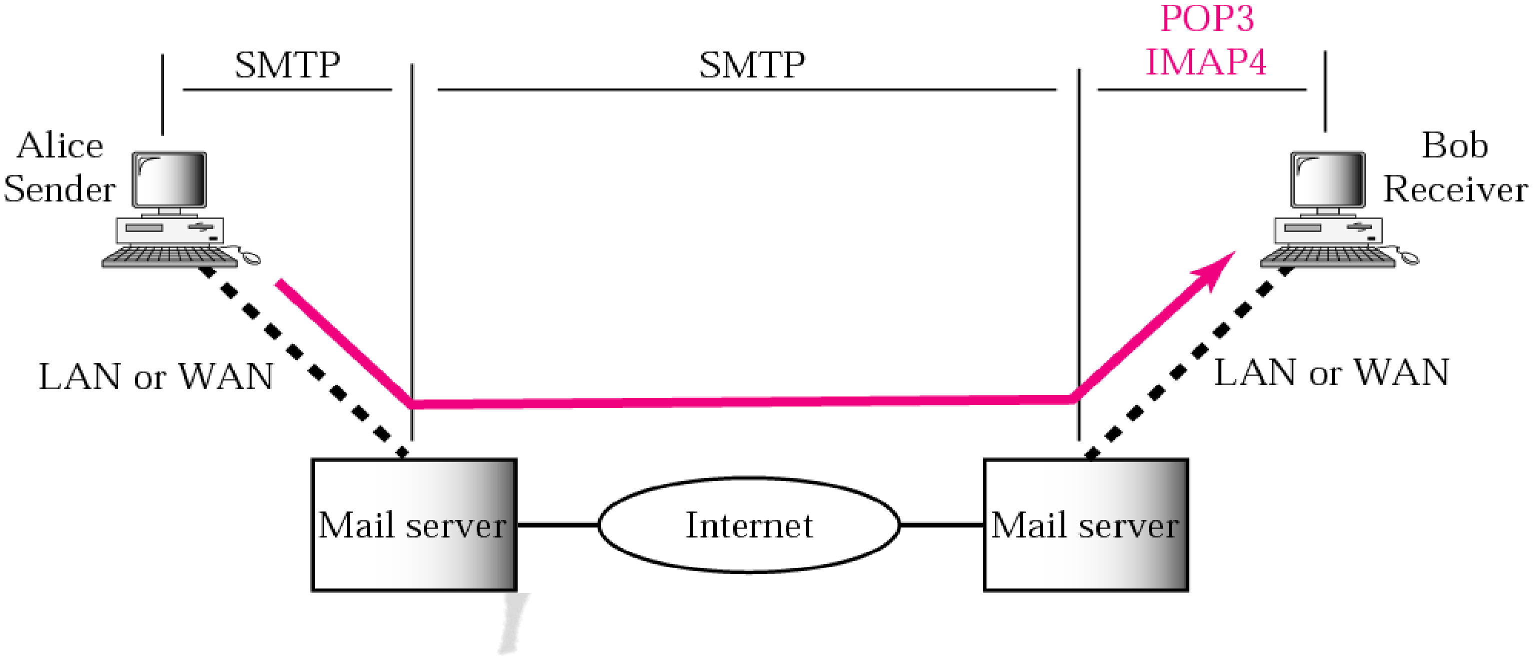
Connection termination



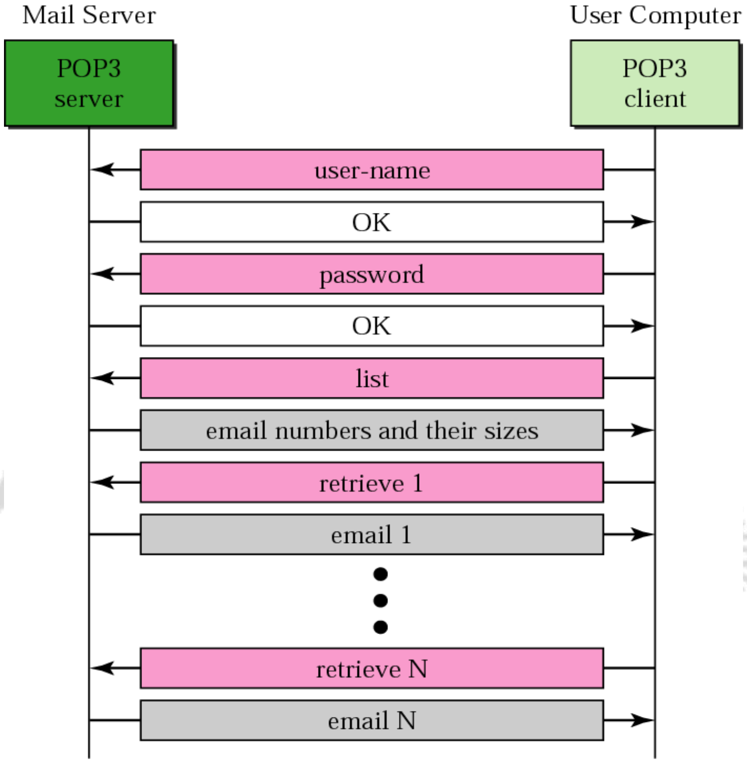
Message Access Agent: POP and IMAP

- The third stage of mail delivery uses a message access agent; the client must pull messages from the server
- Currently two message access protocols are available
  - Post Office Protocol, version 3 (POP3)
  - Internet Mail Access Protocol, version 4

POP3 and IMAP4



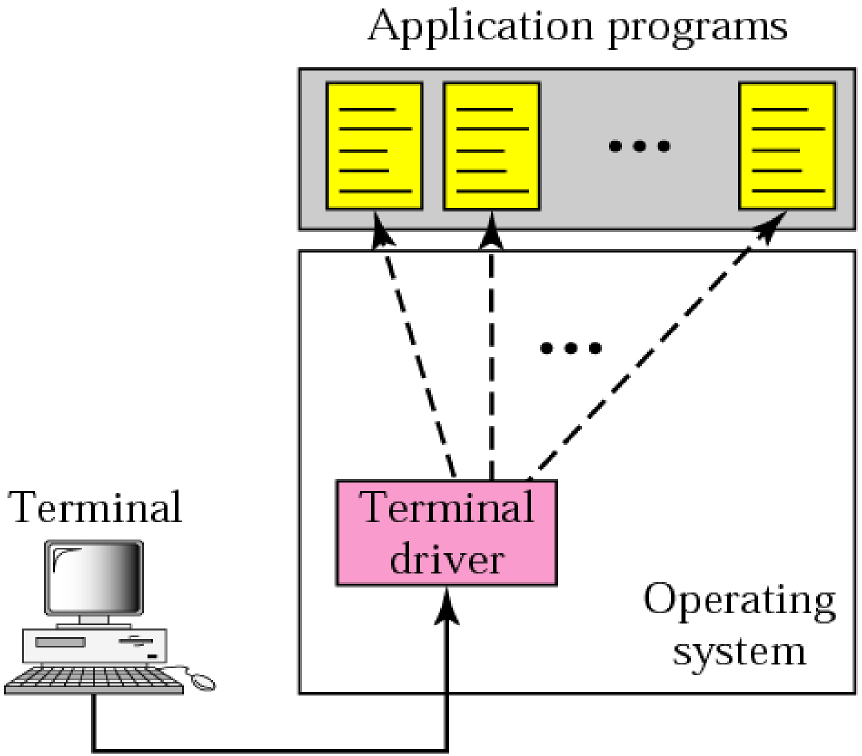
POP3


Mail Server
User Computer
POP3 server
POP3 client
user-name
OK
password
OK
list
email numbers and their sizes
retrieve 1
email 1
retrieve N
email N

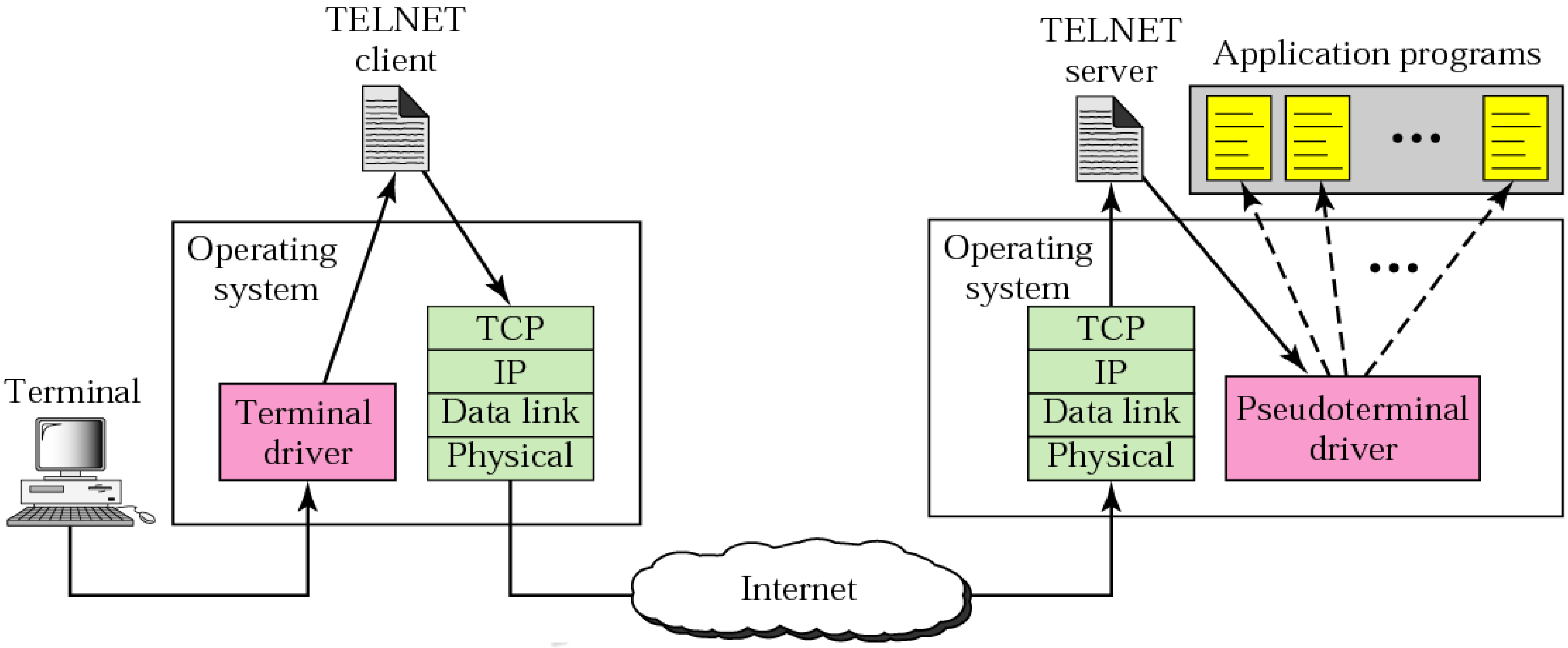
Telnet

- TELNET enables the establishment of a connection to a remote system in such a way that the local terminal appears to be a terminal at the remote system
- TELNET is a general-purpose client-server application program

Local login


Application programs
• • •
• • •
Terminal
Terminal driver
Operating system

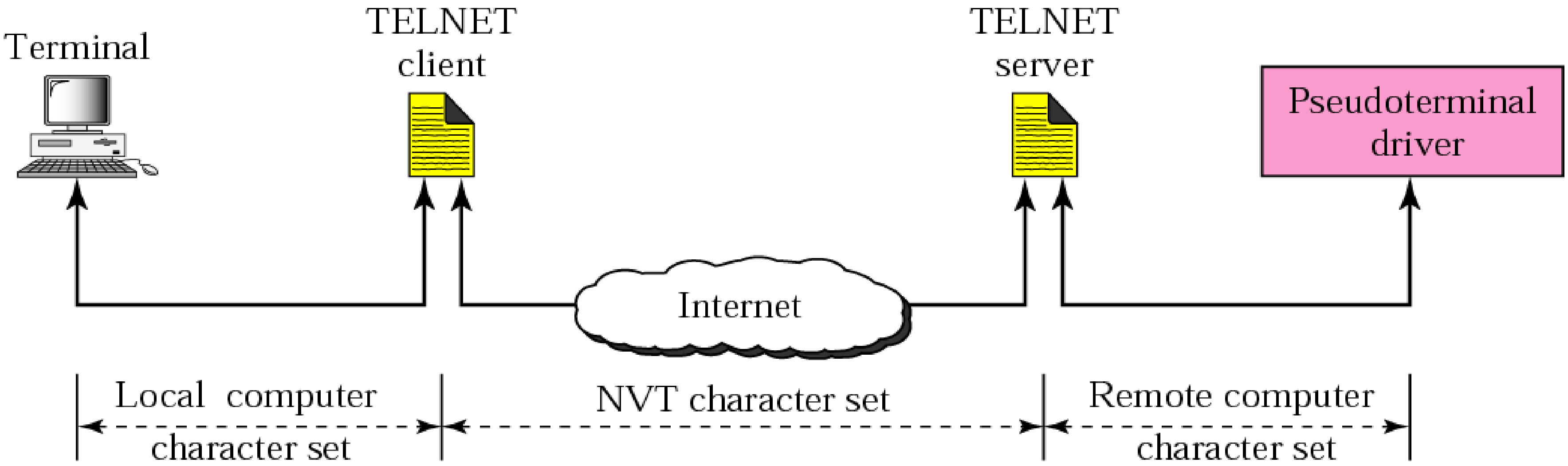
Remote login



Network Virtual Terminal (NVT)

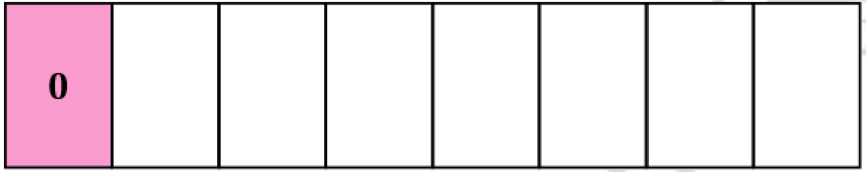
- Via a universal interface called the Network Virtual Terminal (NVT) character set, the TELNET client translates characters (data or commands) that come from the local terminal into NVT form and delivers them to the network
- TELNET server translates data and commands from NVT form into the form acceptable by the remote computer.

Concept of NVT

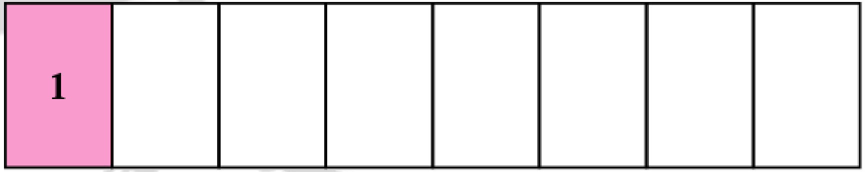


NVT character set

- NVT uses two sets of characters, one for data and one for control. Both are 8-bit bytes

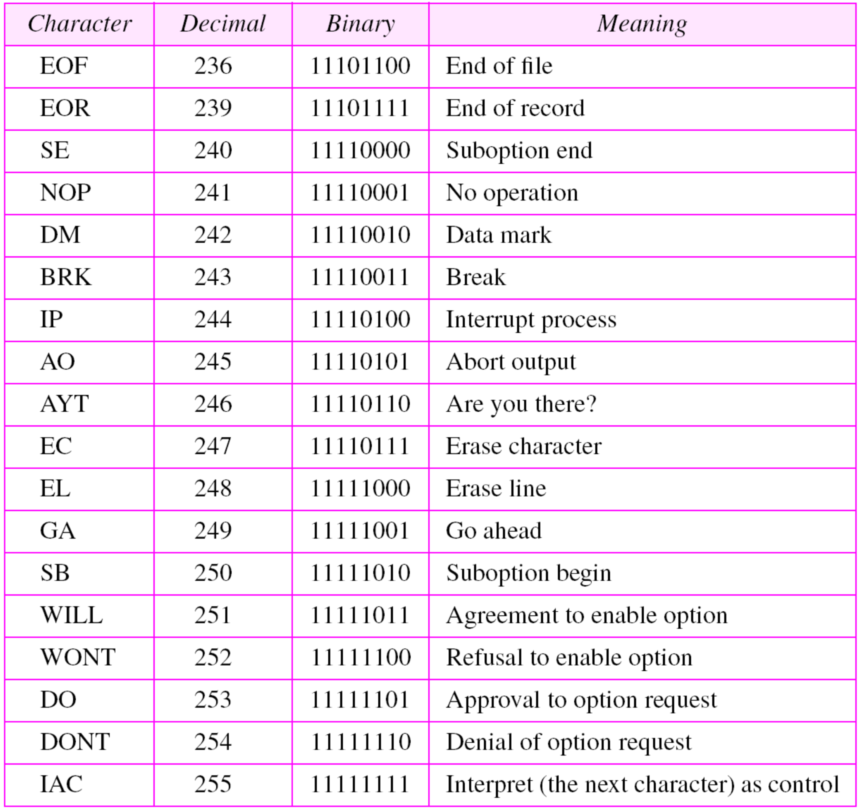


Format of data characters



Format of control characters

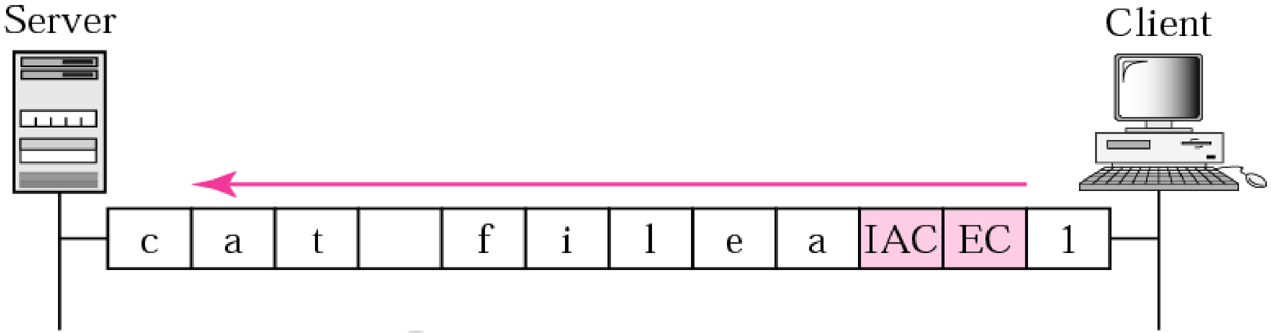
NVT control characters



Character	*Decimal*	*Binary*	*Meaning*
EOF	236	11101100	End of file
EOR	239	11101111	End of record
SE	240	11110000	Suboption end
NOP	241	11110001	No operation
DM	242	11110010	Data mark
BRK	243	11110011	Break
IP	244	11110100	Interrupt process
AO	245	11110101	Abort output
AYT	246	11110110	Are you there?
EC	247	11110111	Erase character
EL	248	11111000	Erase line
GA	249	11111001	Go ahead
SB	250	11111010	Suboption begin
WILL	251	11111011	Agreement to enable option
WONT	252	11111100	Refusal to enable option
DO	253	11111101	Approval to option request
DONT	254	11111110	Denial of option request
IAC	255	11111111	Interpret (the next character) as control

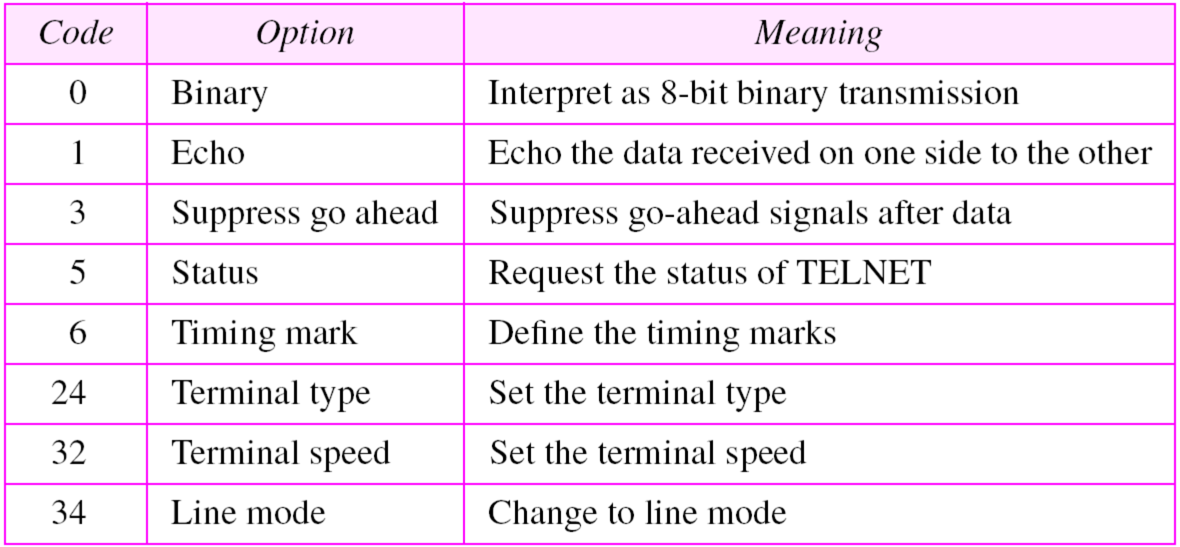
Embedding

- The same connection is used by TELNET for sending both data and control characters
- TELNET accomplishes this by embedding the control characters in the data stream



Options

- TELNET lets the client and server negotiate options before or during the use of the service
- Options are extra features available to a user with a more sophisticated terminal

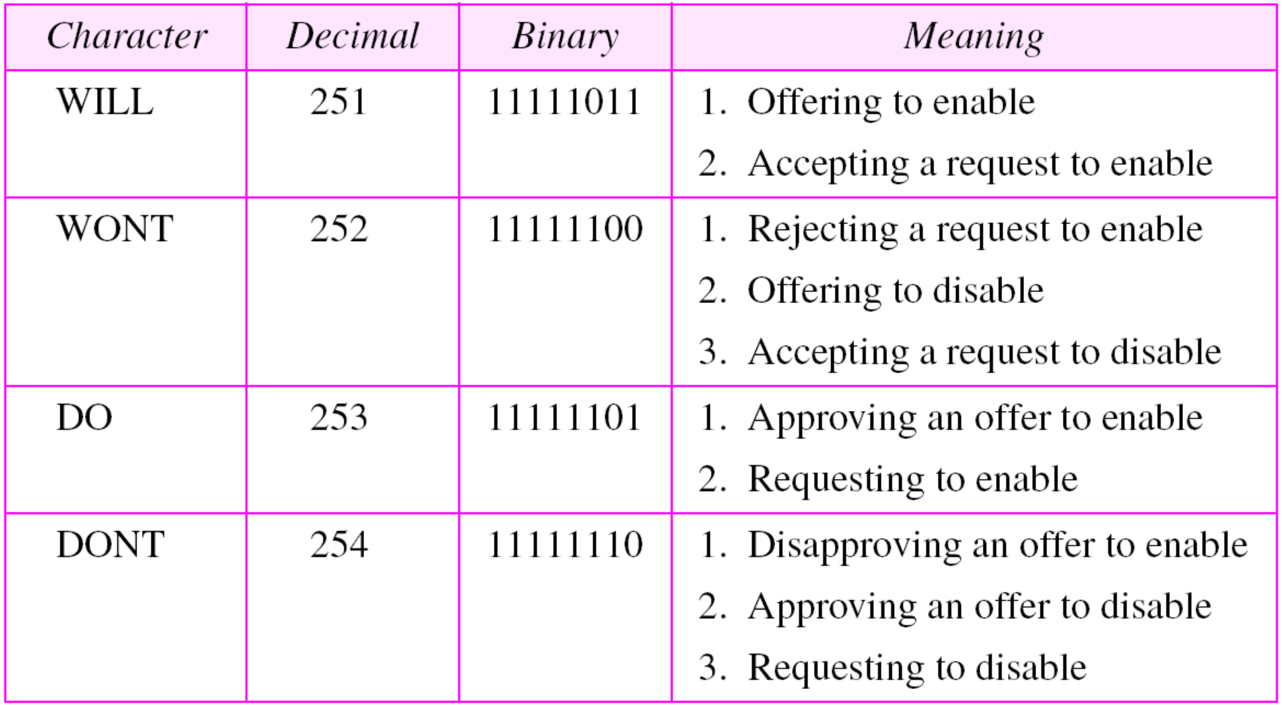


Code	*Option*	*Meaning*
0	Binary	Interpret as 8-bit binary transmission
1	Echo	Echo the data received on one side to the other
3	Suppress go ahead	Suppress go-ahead signals after data
5	Status	Request the status of TELNET
6	Timing mark	Define the timing marks
24	Terminal type	Set the terminal type
32	Terminal speed	Set the terminal speed
34	Line mode	Change to line mode

Option negotiation

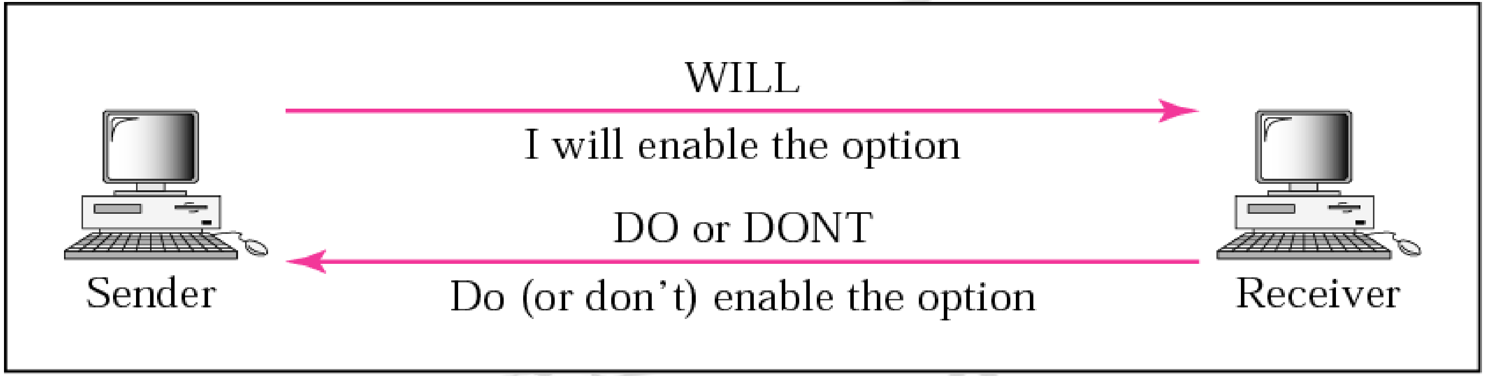
- To use any of the options first requires option negotiation between the client and the server
- Four control characters are used for this purpose

NVT character set for option negotiation

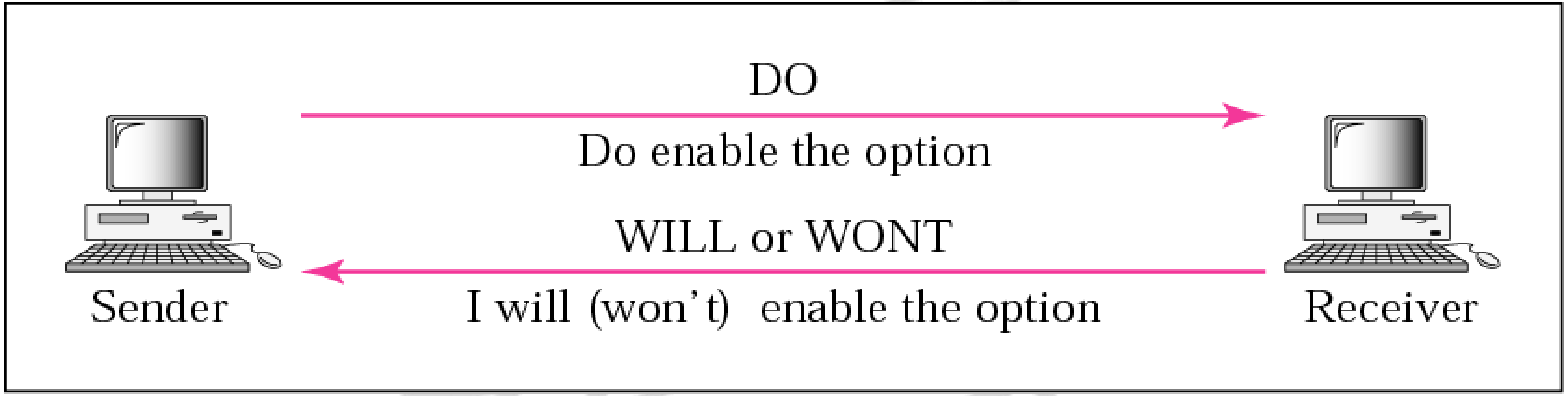


Character	*Decimal*	*Binary*	*Meaning*
WILL	251	11111011	1. Offering to enable 2. Accepting a request to enable
WONT	252	11111100	1. Rejecting a request to enable 2. Offering to disable 3. Accepting a request to disable
DO	253	11111101	1. Approving an offer to enable 2. Requesting to enable
DONT	254	11111110	1. Disapproving an offer to enable 2. Approving an offer to disable 3. Requesting to disable

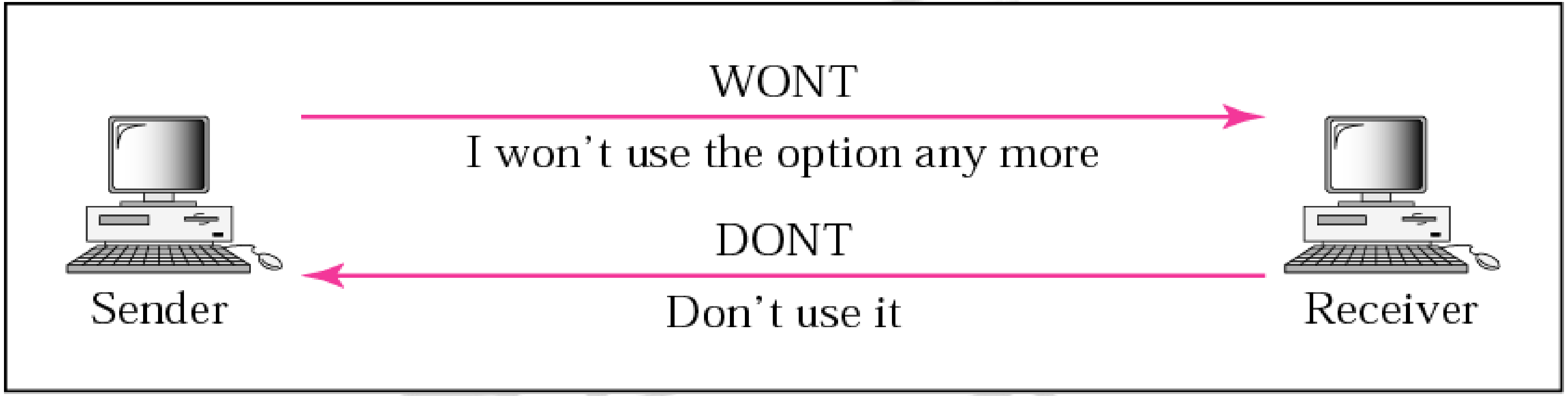
Offer to enable an option



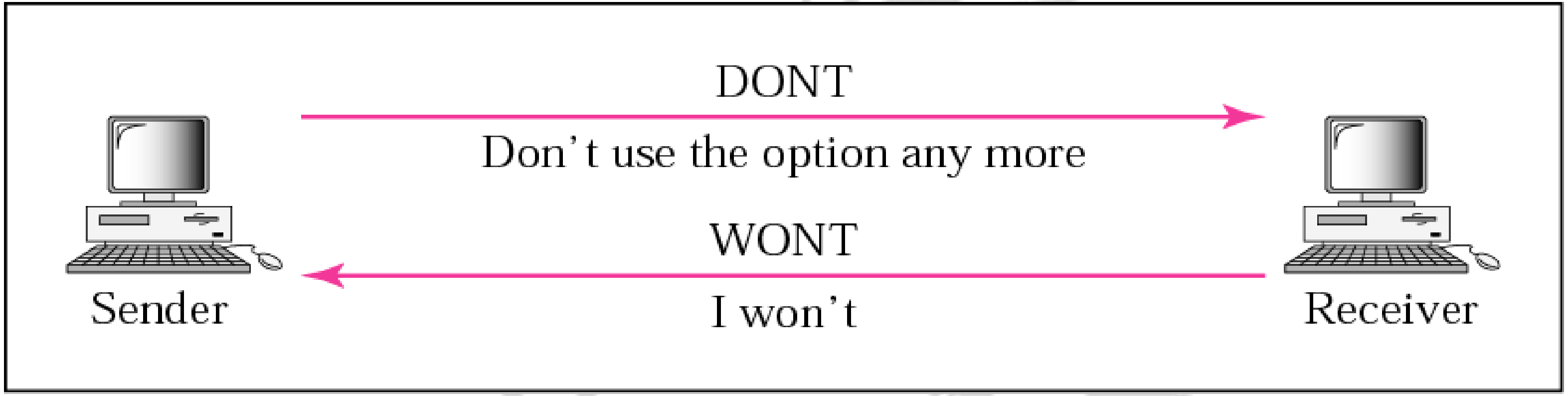
Request to enable an option



Offer to disable an option



Request to disable an option



CRYPTOGRAPHY

Cryptography

- Information needs to be secured from attacks
- To be secured, information needs to be hidden from
  - unauthorized access (confidentiality)
  - protected from unauthorized change (integrity)
  - available to an authorized entity when it is needed (availability)

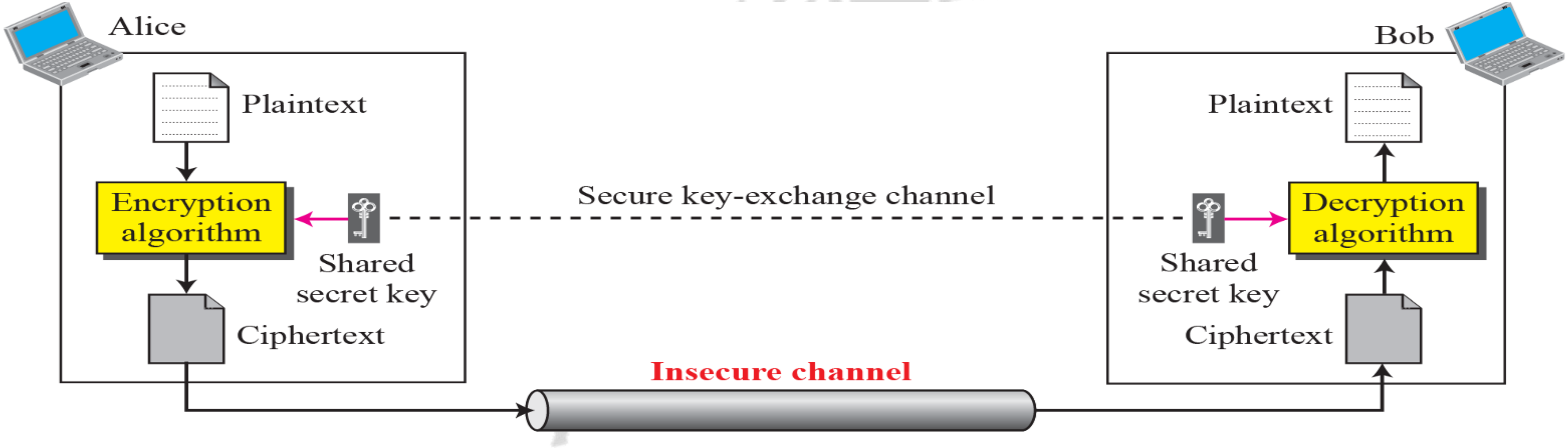
Traditional Ciphers

- We now look at the first goal of security, confidentiality
- Confidentiality can be achieved using ciphers
- Traditional ciphers are called symmetric-key ciphers (or secret-key ciphers) because the same key is used for encryption and decryption and the key can be used for bidirectional communication

- Encryption:
  - plain text (original data) to cipher text
- Decryption
  - cipher text to plain text

General idea of traditional cipher

- A substitution cipher replaces one symbol with another



Representation of characters in modulo 26

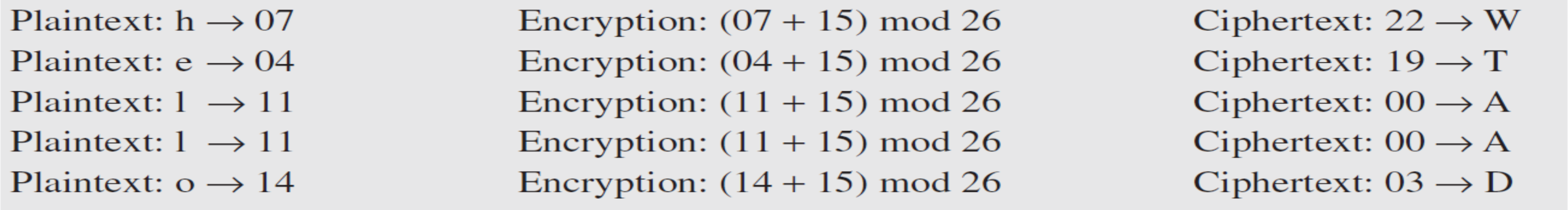
- In additive cipher, the plaintext, ciphertext, and key are integers in modulo 26

Example

- Use the additive cipher with key = 15 to encrypt the message “hello”

Solution

- The cipher is mono-alphabetic because two instances of the same plaintext character (ls) are encrypted as the same character (A). The result is “WTAAD”



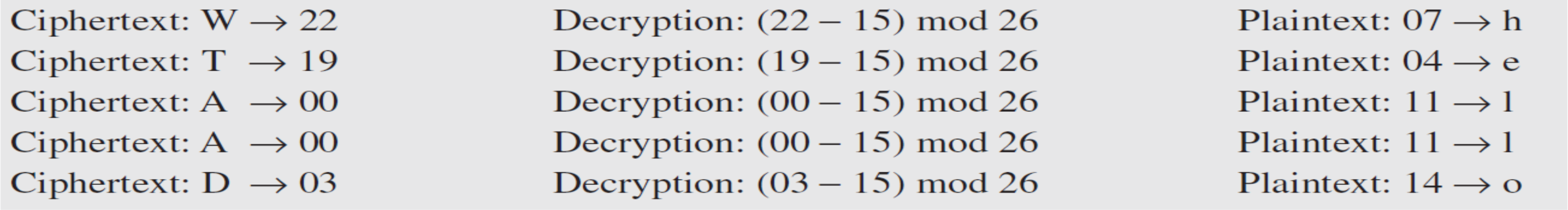
Plaintext: h → 07	Encryption: (07 + 15) mod 26	Ciphertext: 22 → W
Plaintext: e → 04	Encryption: (04 + 15) mod 26	Ciphertext: 19 → T
Plaintext: l → 11	Encryption: (11 + 15) mod 26	Ciphertext: 00 → A
Plaintext: l → 11	Encryption: (11 + 15) mod 26	Ciphertext: 00 → A
Plaintext: o → 14	Encryption: (14 + 15) mod 26	Ciphertext: 03 → D

Example

- Use the additive cipher with key = 15 to decrypt the message “WTAAD”.

Solution

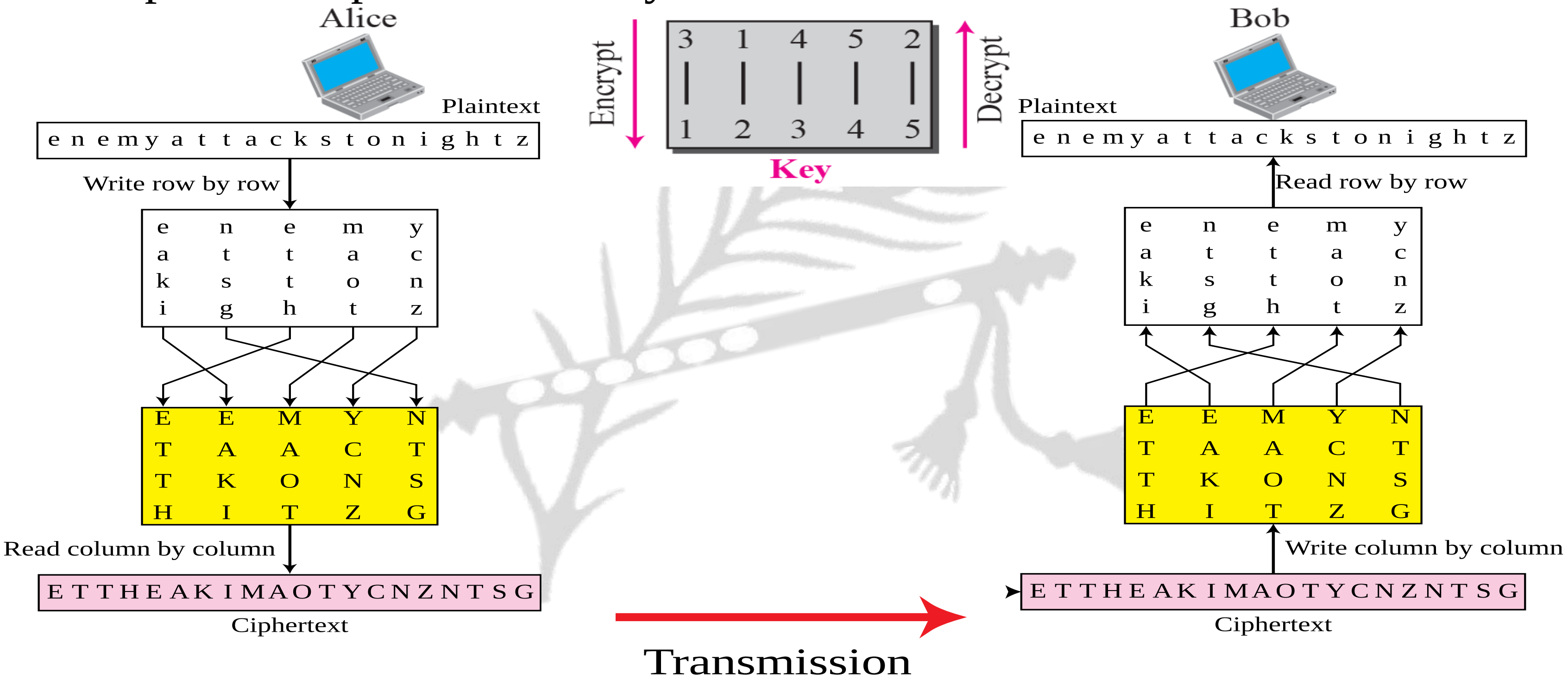
- We apply the decryption algorithm to the plaintext character by character. The result is “hello”. Note that the operation is in modulo 26, which means that we need to add 26 to a negative result (for example -15 becomes 11).



Ciphertext: W → 22	Decryption: (22 – 15) mod 26	Plaintext: 07 → h
Ciphertext: T → 19	Decryption: (19 – 15) mod 26	Plaintext: 04 → e
Ciphertext: A → 00	Decryption: (00 – 15) mod 26	Plaintext: 11 → l
Ciphertext: A → 00	Decryption: (00 – 15) mod 26	Plaintext: 11 → l
Ciphertext: D → 03	Decryption: (03 – 15) mod 26	Plaintext: 14 → o

Transposition cipher

- A transposition cipher reorders symbols



enemyattackstonightz

3	1	4	5	2
1	2	3	4	5

e	n	e	m	y
a	t	t	a	c
k	s	t	o	n
i	g	h	t	z

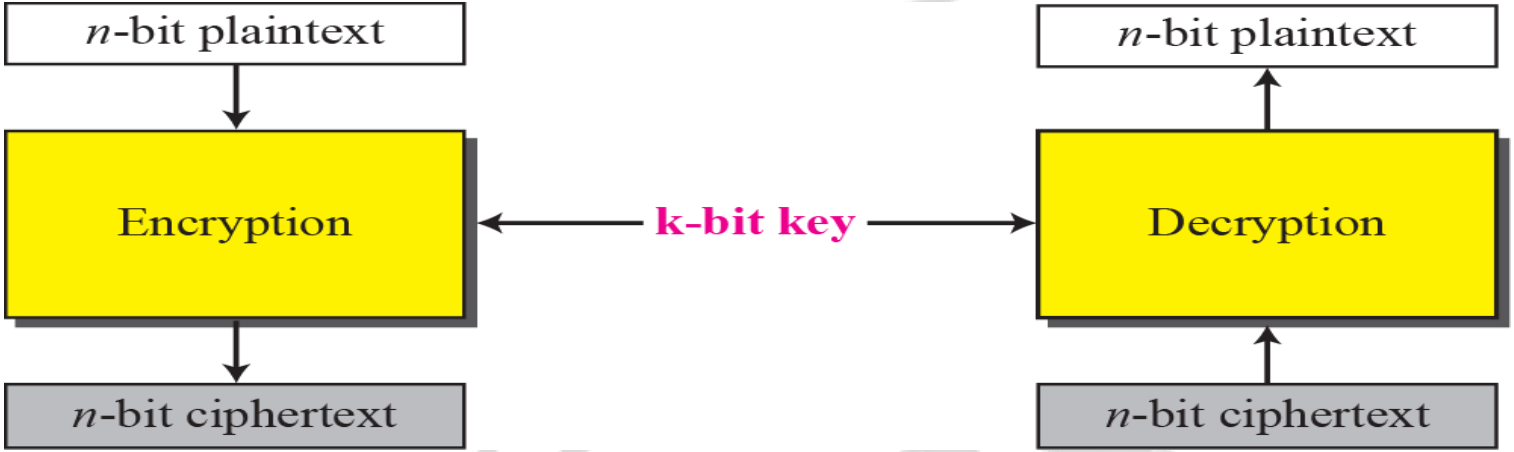
E	E	M	Y	N
T	A	A	C	T
T	K	O	N	S
H	I	T	Z	G

ETTHEAKIMAOTYCNZNTSG

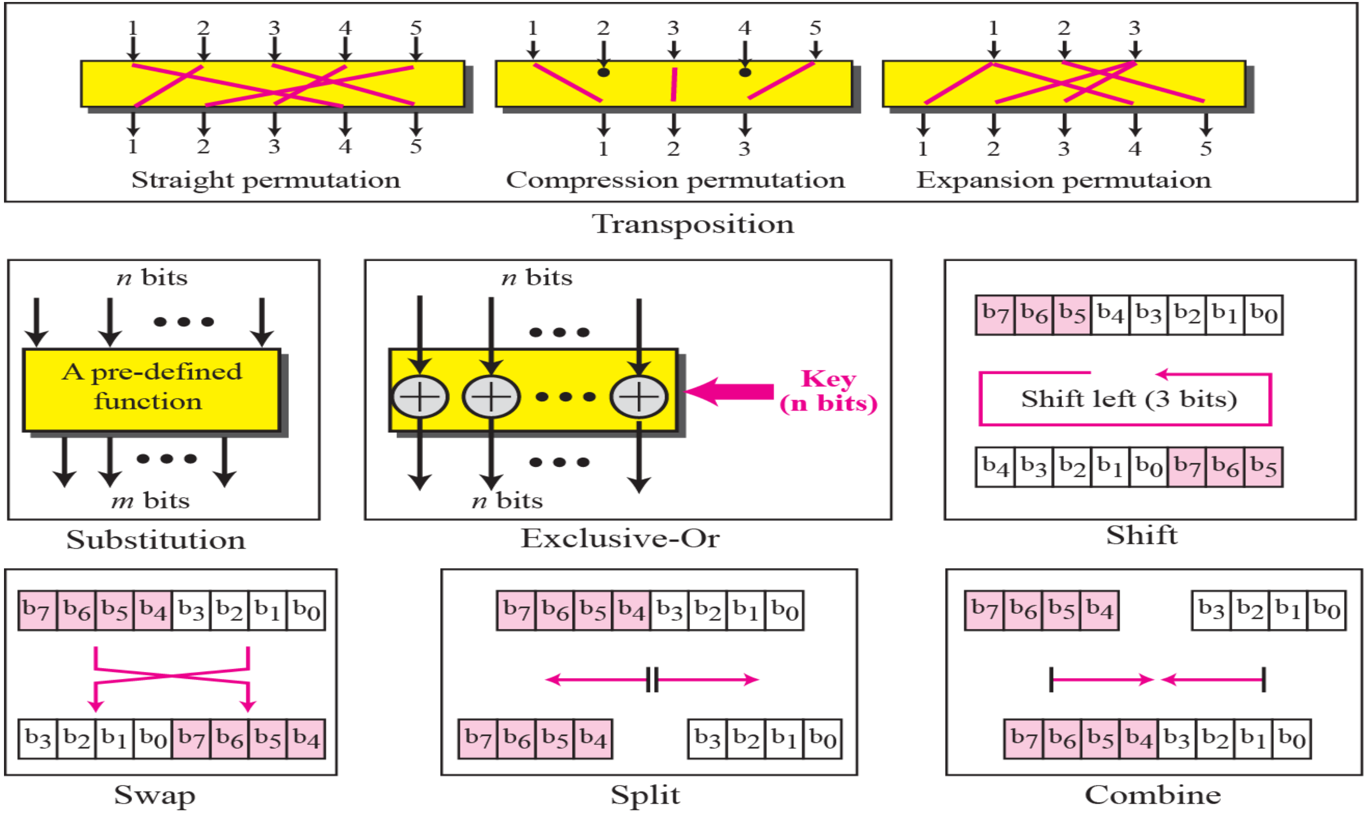
Modern ciphers

- The traditional symmetric-key ciphers that we have studied so far are character-oriented ciphers
- With the advent of the computer, we need bit-oriented ciphers
- This is because the information to be encrypted is not just text; it can also consist of numbers, graphics, audio, and video data
- It is convenient to convert these types of data into a stream of bits, to encrypt the stream, and then to send the encrypted stream
- A modern block cipher can be either a block cipher or a stream cipher

Modern ciphers



Components of Modern ciphers



Straight permutation

Compression permutation

Expansion permutaion

Transposition

Substitution

Exclusive-Or

Shift

Swap

Split

Combine

Asymmetric key ciphers

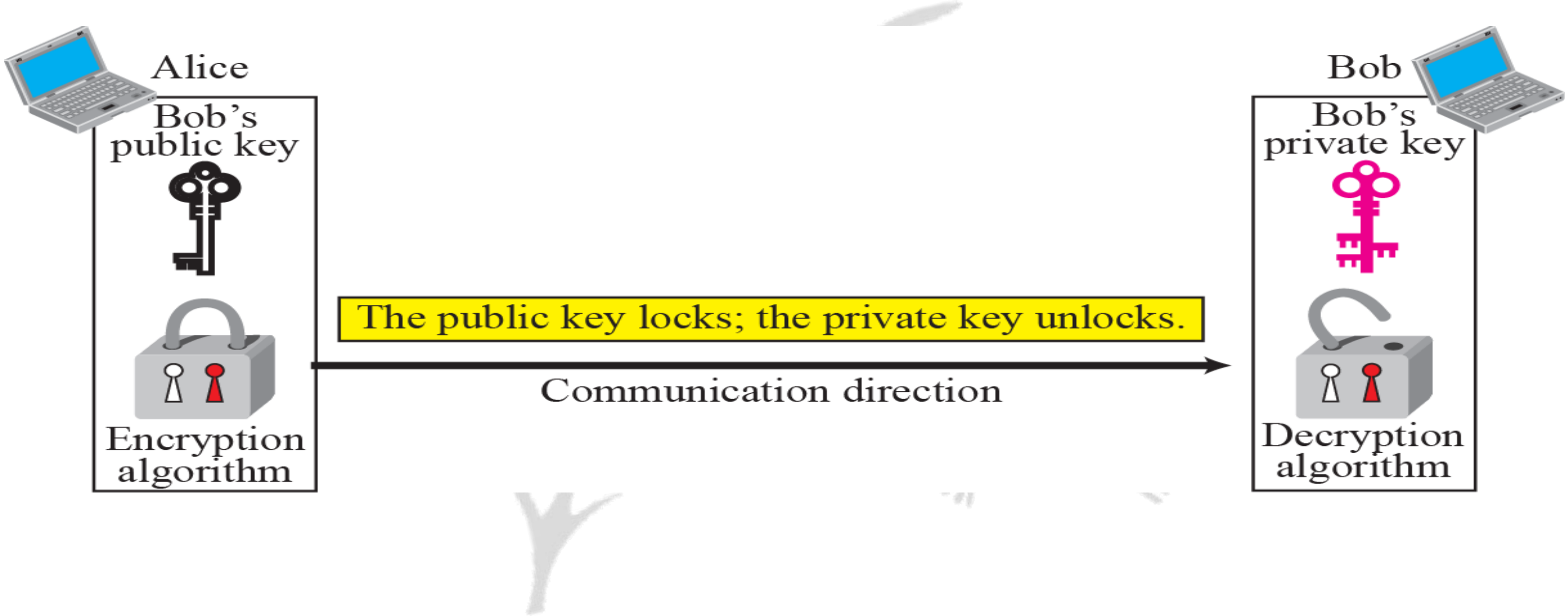
- Symmetric-key and asymmetric-key ciphers will exist in parallel and continue to serve the community
- We actually believe that they are complements of each other
- The advantages of one can compensate for the disadvantages of the other

- Symmetric-key cryptography is based on sharing secrecy
- Asymmetric-key cryptography is based on personal secrecy

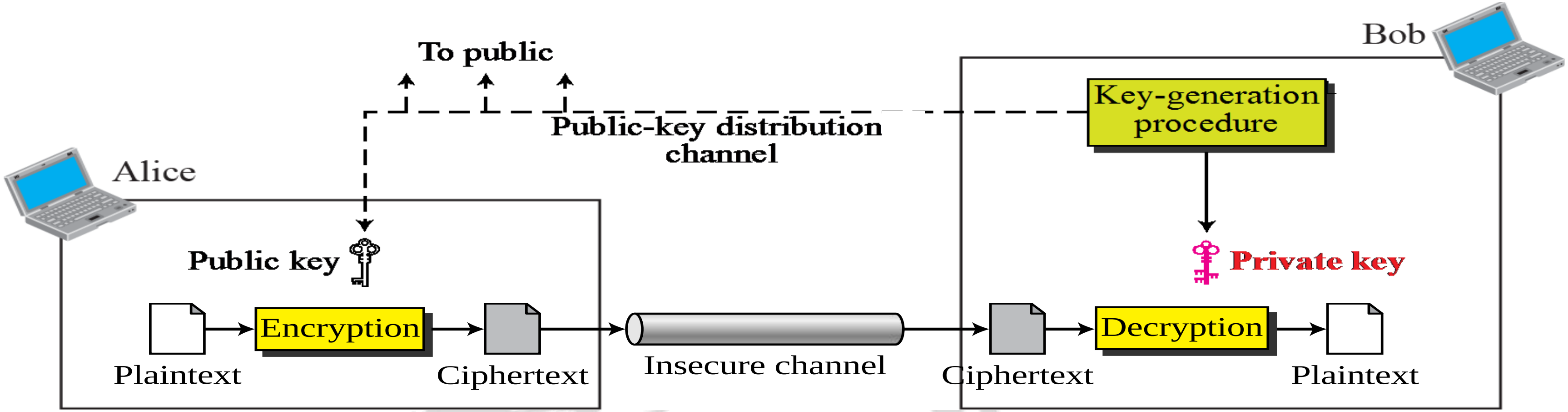
- In symmetric-key cryptography, symbols are permuted or substituted
- in asymmetric-key cryptography, numbers are manipulated

Asymmetric key ciphers

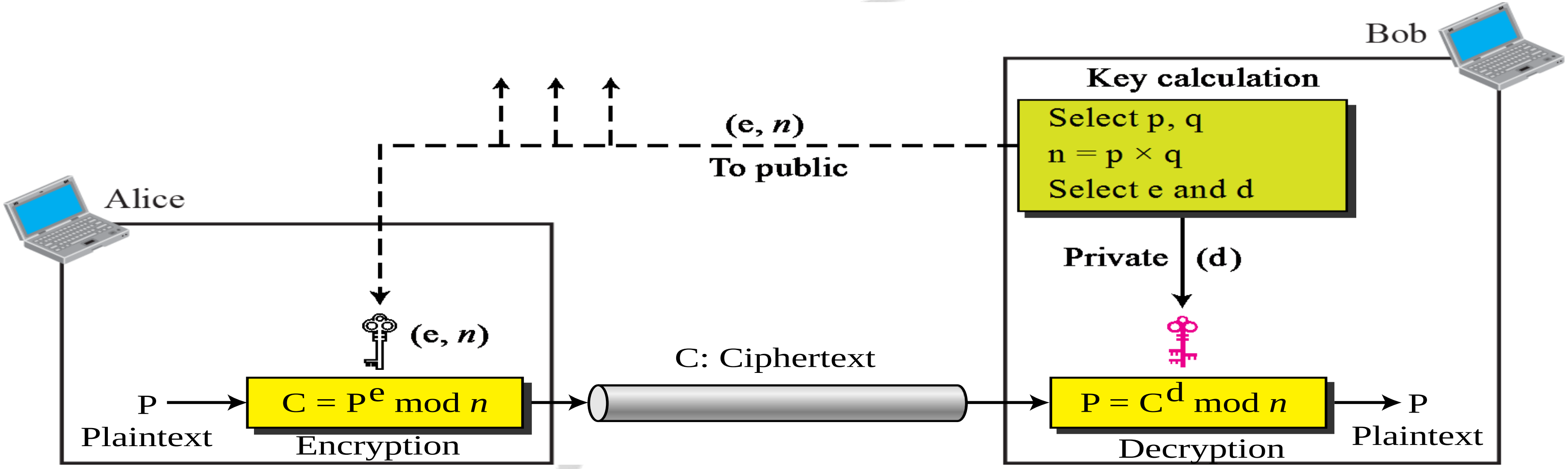
- Asymmetric-key ciphers are sometimes called public-key ciphers



General idea of asymmetric key ciphers

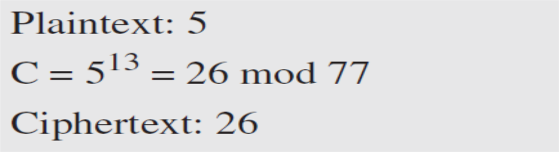


Encryption, decryption and key in RSA



Example

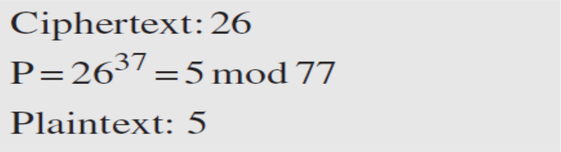
- For the sake of demonstration, let Bob choose 7 and 11 as p and q and calculate n = 7 × 11 = 77
- The value of φ(n) = (7 − 1)(11 − 1), or 60. If he chooses e to be 13, then d is 37. Note that e × d mod 60 = 1
- Now imagine that Alice wants to send the plaintext 5 to Bob. She uses the public exponent 13 to encrypt 5. This system is not safe because p and q are small



Plaintext: 5

$C = 5^{13} = 26 \bmod 77$

Ciphertext: 26



Ciphertext: 26

$P = 26^{37} = 5 \bmod 77$

Plaintext: 5