

# *ELECTRONIC MAIL*

## *SMTPL, POP, IMAP, MIME*

Prof. Sruti S. Singh

# Electronic Mail (e-mail):

- It allows users to exchange messages.
- Messaging services typically fall into one of **two models**, **Push messaging**, or **Pull messaging**. The major difference is the method of message retrieval.

## **Push messaging**

- New messages are delivered automatically to the client by the server.

## **Pull messaging**

- New messages are delivered only when requested by the client.
- In practice, a “Send/Receive” or “Check Mail” function is called by the user manually, or by the client application per a set schedule, to pull in new messages.

# Electronic mail (E-mail) cont...

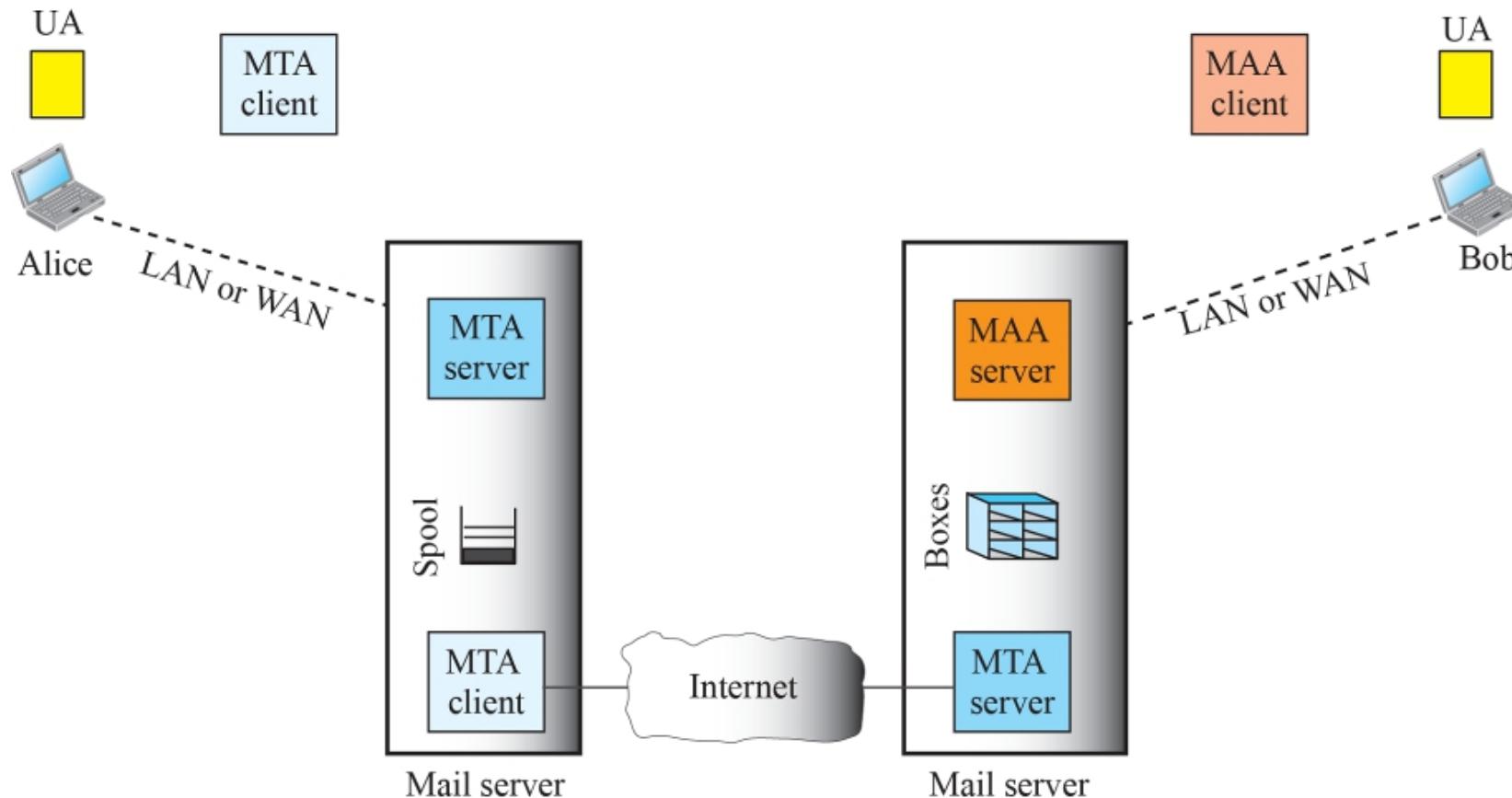
- In HTTP or FTP, the server program is running all the time, waiting for a request from a client. When the request arrives, the server provides the service.
- In E-mail, the **scenario is different**:
  - e-mail is considered a one-way transaction. **Sender** may expect a response, but this is not a mandate.
  - it is neither feasible nor logical for the **receiver** to run a server program and wait until someone sends an e-mail to him.
  - the idea of client/server programming should be implemented in another way: using some **intermediate servers**.
  - both the end **users run only client programs** when they want, and the intermediate servers apply the client/server paradigm

# Architecture:

UA: user agent

MTA: message transfer agent

MAA: message access agent



- In the example, the **sender** and the **receiver** of the e-mail, Alice and Bob respectively, are **connected via a LAN or a WAN to two mail servers.**
- The administrator has created **one mailbox for each user** where the received messages are stored.
- The administrator has also created a queue (spool) to store **messages waiting to be sent.**
- Alice and Bob use three different agents:
  - a user agent (UA),
  - a message transfer agent (MTA),
  - a message access agent (MAA).

- 1) When Alice needs to send a message to Bob, she **runs a UA program** to prepare the message and send it to her mail server.
  - 2) The mail server at her site uses a queue (spool) to **store messages** waiting to be sent.
- 
- 3) The message, however, needs to be sent through the Internet **from Alice's site to Bob's site** using an MTA.
    - Here **two MTAs** are needed: one client and one server.
    - the **server needs to run** all the time because it does not know when a client will ask for a connection.
    - The client, on the other hand, can be triggered by the system when there is a message in the queue to be sent.
- 
- 4) Bob uses an **MAA client** to retrieve the message from an **MAA server** running on its mail server.
  - 5) The **UA program** at the Bob site allows Bob to read the received message
- Two important points:
    - Bob **cannot bypass the mail server** and use the MTA server directly. To use the MTA server directly, Bob would need to run the MTA server all the time because he does not know when a message will arrive.
    - Bob **needs another pair of client-server programs**: message access programs. This is because an MTA client-server program is a *push* program. But Bob needs a *pull* program.
  - So, **the e-mail system needs two UAs, two pairs of MTAs (client and server), and a pair of MAAs (client and server)**.

# User Agent (UA)

- It provides service to the user to make the process of sending and receiving a message easier.
- A user agent is a **software package** (program) that composes, reads, replies to, and forwards messages.
- It also **handles local mailboxes** on the user computers.
- There are **two types of user agents**:
  - **command-driven** : e.g., *mail*
  - **GUI-based** : e.g., *Outlook*
- To **send mail**, the user, through the UA, creates mail that looks very similar to postal mail.

Behrouz Forouzan  
20122 Olive Street  
Bellbury, CA 91000



Firouz Mosharraf  
1400 Los Gatos Street  
San Louis, CA 91005

Behrouz Forouzan  
20122 Olive Street  
Bellbury, CA 91000  
Jan. 10, 2011

Subject: Network

Dear Mr. Mosharraf  
We want to inform you that  
our network is working pro-  
perly after the last repair.

Yours truly,  
Behrouz Forouzan

## Postal mail

**Mail From:** forouzan@some.com  
**RCPT To:** mosharraf@aNetwork.com

From: Behrouz Forouzan  
To: Firouz Mosharraf  
Date: 1/10/2011  
Subject: Network

Dear Mr. Mosharraf  
We want to inform you that  
our network is working pro-  
perly after the last repair.

Yours truly,  
Behrouz Forouzan

Envelope

Header

Body

Message

## Electronic mail

- The UA is triggered by the user (or a timer).
- If a user has mail, the UA informs the user with a notice.
- If the user is ready to read the mail, a list is displayed.
- The user can select any of the messages and display its contents on the screen.
- To deliver mail, a mail handling system must use an addressing system with unique addresses.

Local part

Mailbox address of the recipient

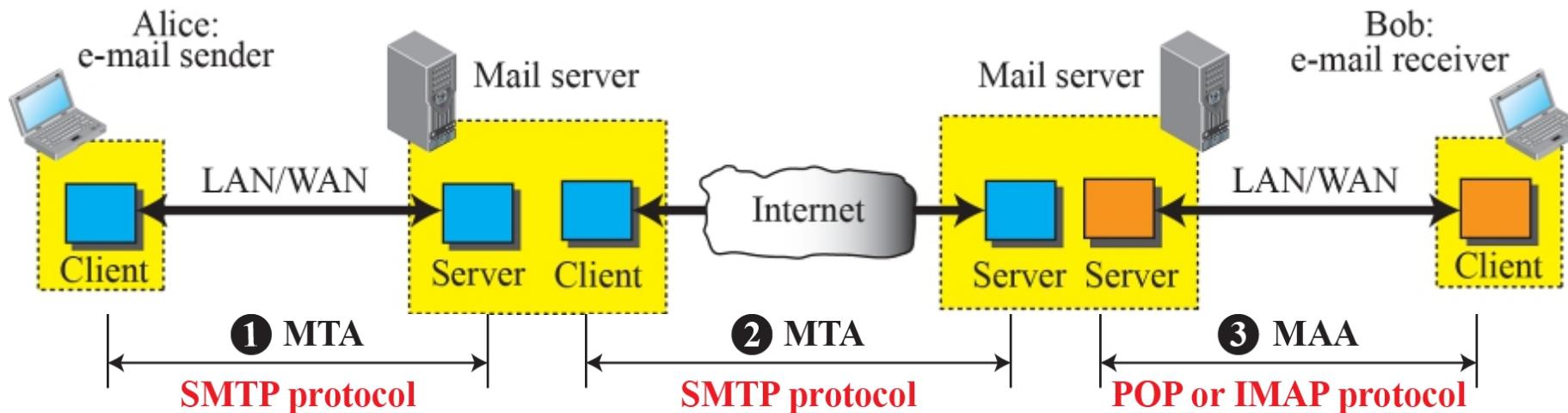


Domain name

The domain name of the mail server

# Message Transfer Agent: SMTP

- The formal protocol that defines the MTA client and server in the Internet is called *Simple Mail Transfer Protocol (SMTP)*.
- SMTP is used two times:
  - Between the sender and sender's mail server
  - Between the two mail servers
- POP or IMAP is used between the receiver and receiver's mail server



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

### SMTP Commands

<i>Keyword</i>	<i>Argument(s)</i>	<i>Description</i>
HELO	Sender's host name	Identifies itself
MAIL FROM	Sender of the message	Identifies the sender of the message
RCPT TO	Intended recipient	Identifies the recipient of the message
DATA	Body of the mail	Sends the actual message
QUIT		Terminates the message
RSET		Aborts the current mail transaction
VRFY	Name of recipient	Verifies the address of the recipient
NOOP		Checks the status of the recipient
TURN		Switches the sender and the recipient
EXPN	Mailing list	Asks the recipient to expand the mailing list.
HELP	Command name	Asks the recipient to send information about the command sent as the argument
SEND FROM	Intended recipient	Specifies that the mail be delivered only to the terminal of the recipient, and not to the mailbox
SMOL FROM	Intended recipient	Specifies that the mail be delivered to the terminal <i>or</i> the mailbox of the recipient
SMAL FROM	Intended recipient	Specifies that the mail be delivered to the terminal <i>and</i> the mailbox of the recipient

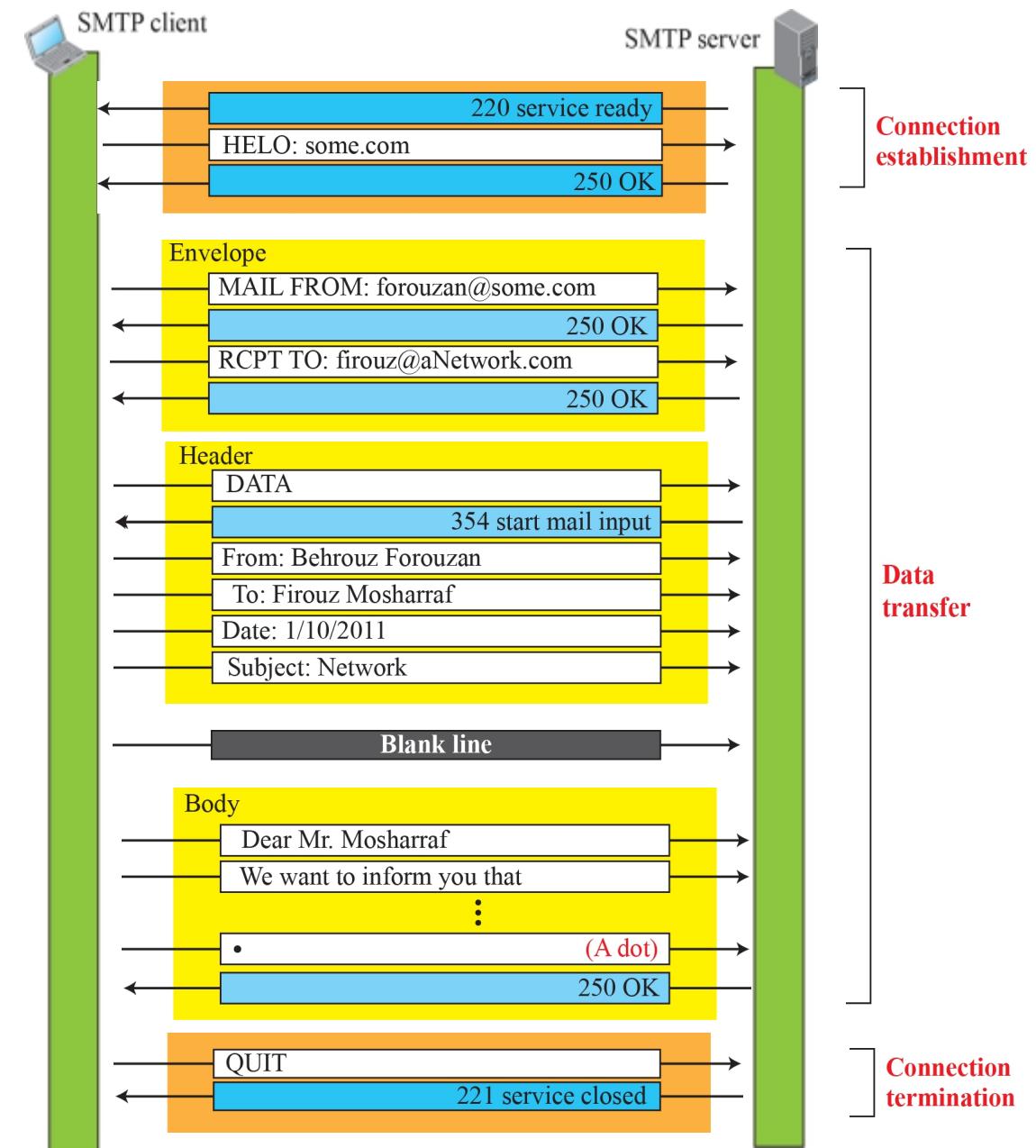
<i>Code</i>	<i>Description</i>
<b>Positive Completion Reply</b>	
<b>211</b>	System status or help reply
<b>214</b>	Help message
<b>220</b>	Service ready
<b>221</b>	Service closing transmission channel
<b>250</b>	Request command completed
<b>251</b>	User not local; the message will be forwarded
<b>Positive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

*SMTP responses*

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

# Mail Transfer Phases

- The process of transferring a mail message occurs in **three phases**:
  - connection establishment,
  - mail transfer,
  - connection termination.
- After a client has made a **TCP connection** to the well-known **port 25**, the SMTP protocol starts its connection phase.

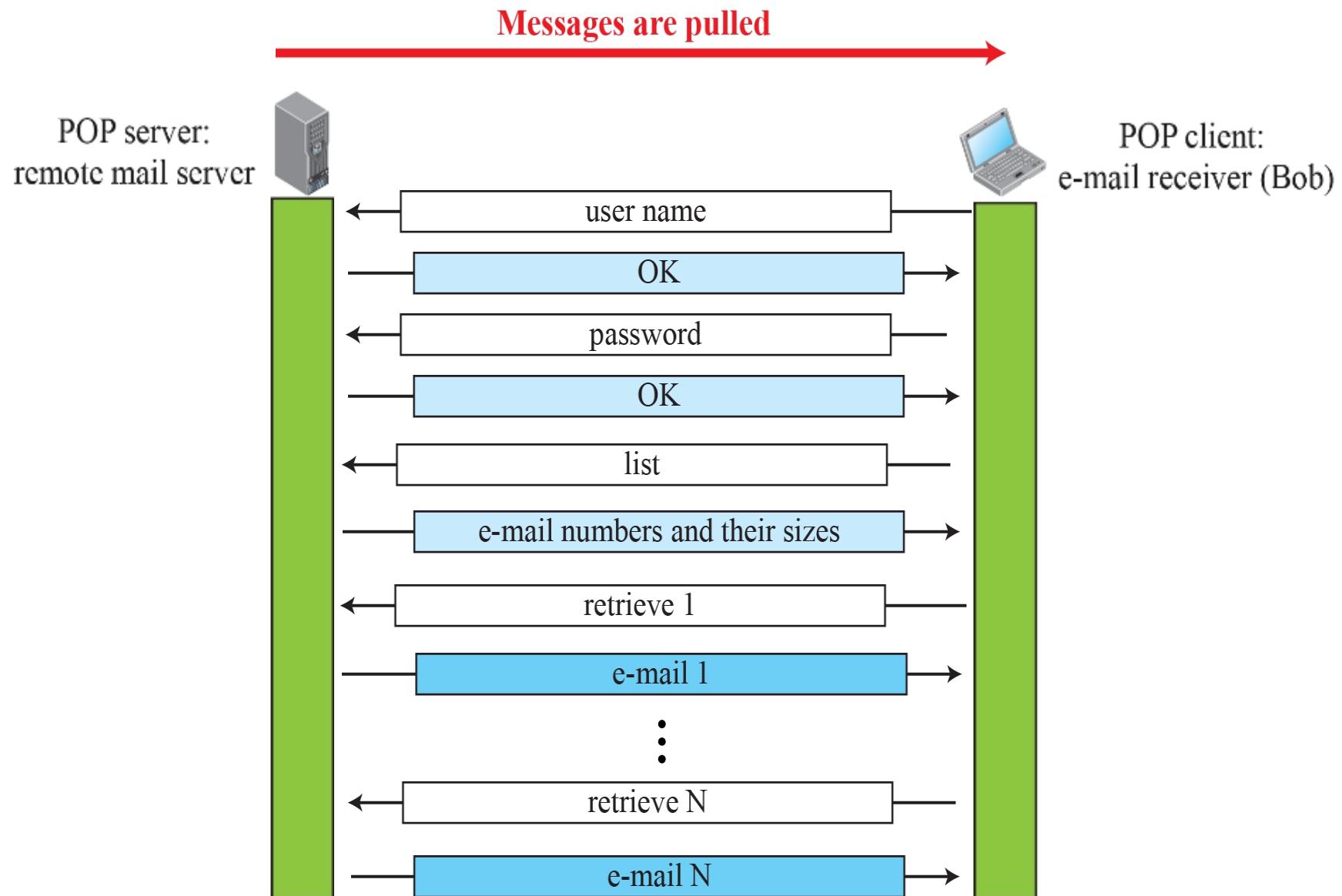


# Message Access Agent: POP/IMAP

- In the third stage we needs a *pull protocol*; the client must pull messages from the server.
- Two message access protocols are available:
  - Post Office Protocol (POP)
  - Internet Mail Access Protocol (IMAP)

# POP3:

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



## Post Office Protocol (POP3)

POP is a simple protocol that only allows downloading messages from your Inbox to your local computer.

In POP3 the mail can only be accessed from a single device at a time.

To read the mail it has to be downloaded on the local system.

The user can not organize mails in the mailbox of the mail server.

The user can not create, delete or rename email on the mail server.

A user can not search the content of mail before downloading to the local system.

It has two modes : delete mode and keep mode.

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

In keep mode , the mail remains in the mail box after retrieval.

Changes in the mail can be done using local email software.

All the message are downloaded at once.

# IMAP4:

- IMAP4 is similar to POP3, but it has more features; IMAP4 is more powerful and more complex.
- IMAP4 provides the following extra functions:
  - A user can check the e-mail header prior to downloading.
  - A user can search the contents of the e-mail for a specific string of characters prior to downloading.
  - A user can partially download e-mail. This is especially useful if bandwidth is limited and the e-mail contains multimedia with high bandwidth requirements.
  - A user can create, delete, or rename mailboxes on the mail server.
  - A user can create a hierarchy of mailboxes in a folder for e-mail storage.

## Internet Message Access Protocol (IMAP)

IMAP is much more advanced and allows you the user to see all the folders on the mail server.

Messages can be accessed across multiple devices

The mail content can be read partially before downloading.

The user can organize the emails directly on the mail server.

The user can create, delete or rename email on the mail server.

A user can search the content of mail for specific string before downloading.

Multiple redundant copies of the message are kept at the mail server, in case of loss of message of a local server, the mail can still be retrieved

Changes made web interface or email software stay in sync with the server.

Message header can be viewed prior to downloading.

# Multipurpose Internet Mail Extension(MIME):

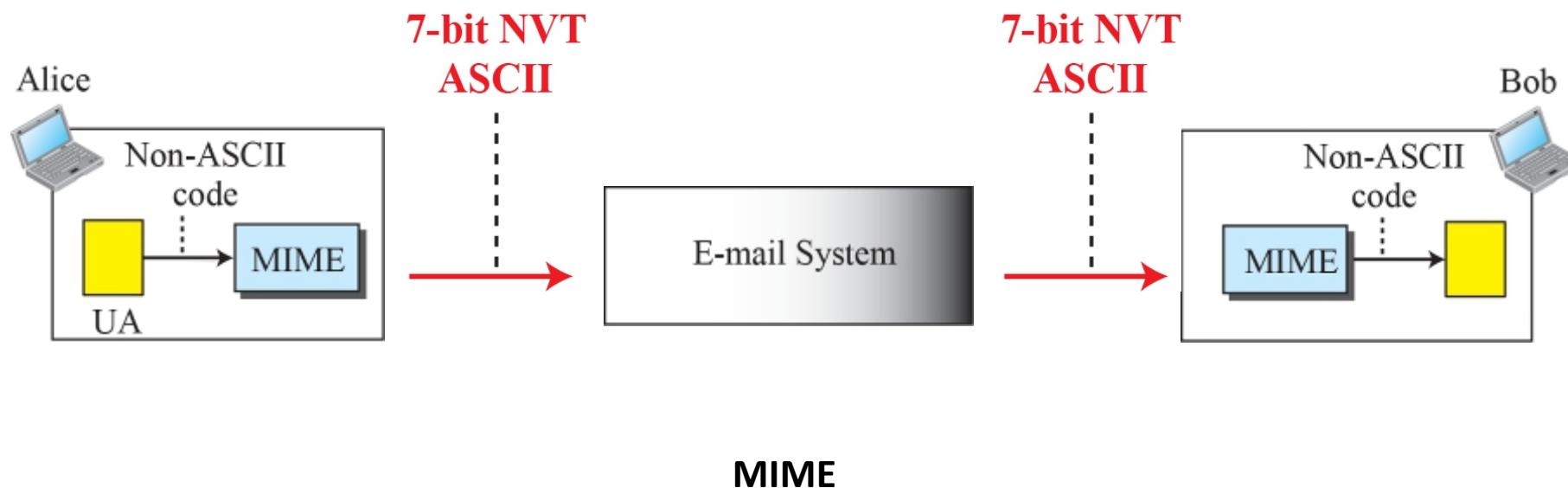
- MIME is a kind of add on or a supplementary protocol which **allows non-ASCII data to be sent through SMTP**.
- It allows the users to exchange different kinds of data files on the Internet: audio, video, images, application programs as well.

## **Features of MIME –**

- It is able to send multiple attachments with a single message.
- Unlimited message length.
- Binary attachments (executables, images, audio, or video files) which may be divided if needed.
- MIME provided support for varying content types and multi-part messages.

## Working of MIME –

- Suppose a user wants to send an email through user agent and it is in a non-ASCII format so there is a MIME protocol which converts it into 7-bit NVT ASCII format.
- Message is transferred through e-mail system to the other side in 7-bit format, now MIME protocol again converts it back into non-ASCII code and then the user agent of receiver side reads it and then information is finally read by the receiver.
- MIME header is basically inserted at the beginning of any e-mail transfer.



# MIME Header:

There are five headers which we add to the original header:

MIME Version – Defines version of MIME protocol. It must have the parameter Value 1.1, which indicates that message is formatted using MIME.

Content Type – Type of data used in the body of message. They are of different types like text data (plain, HTML), audio content or video content.

Content Transfer Encoding – It defines the method used for encoding the message. Like 7-bit encoding, 8-bit encoding, etc.

Content Id – It is used for uniquely identifying the message.

Content description – It defines whether the body is actually image, video or audio.

**MIME headers**

E-mail header

MIME-Version: 1.1

Content-Type: type/subtype

Content-Transfer-Encoding: encoding type

Content-ID: message ID

Content-Description: textual explanation of nontextual contents

E-mail body

# Data Types and Subtypes in MIME

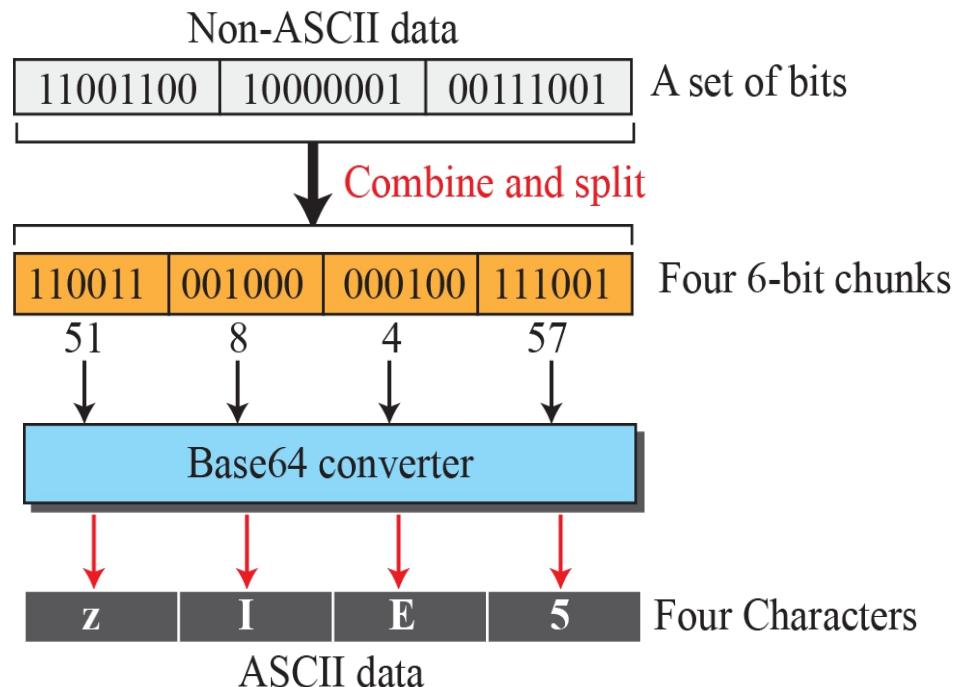
Type	Subtype	Description
Text	Plain	Unformatted
	HTML	HTML format (see Appendix C)
Multipart	Mixed	Body contains ordered parts of different data types
	Parallel	Same as above, but no order
	Digest	Similar to Mixed, but the default is message/RFC822
	Alternative	Parts are different versions of the same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

## Methods for Content-Transfer-Encoding

Type	Description
7-bit	NVT ASCII characters with each line less than 1000 characters
8-bit	Non-ASCII characters with each line less than 1000 characters
Binary	Non-ASCII characters with unlimited-length lines
Base64	6-bit blocks of data encoded into 8-bit ASCII characters
Quoted-printable	Non-ASCII characters encoded as an equal sign plus an ASCII code

## Base64 conversion:

- In the Base64 encoding, data, as a string of bits, is first divided into 6-bit chunks as shown in figure.
- Each 6-bit section is then converted into an ASCII character according to Base64 converting table.
- It is a redundant encoding scheme which implies every six bits become one ASCII character and are sent as eight bits.
- Here we have an overhead of 25 percent.



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 encoding:

- If the data contains mostly ASCII characters with a small non-ASCII portion, we can use Quoted-printable encoding.
- In this encoding, if a character is ASCII then it is sent as it is, but if a character is non-ASCII then it is sent as three characters.
- First character is equal sign (=). The next two characters are hexadecimal representations of the byte.

