UNIT 5- APPLICATION LAYER
EMAIL and FTP

Prepared By

Haseeba Yaseen

26-2 ELECTRONIC MAIL

One of the most popular Internet services is electronic mail (e-mail). Its architecture consists of three main components – user agent, message transfer agent and message access agent.

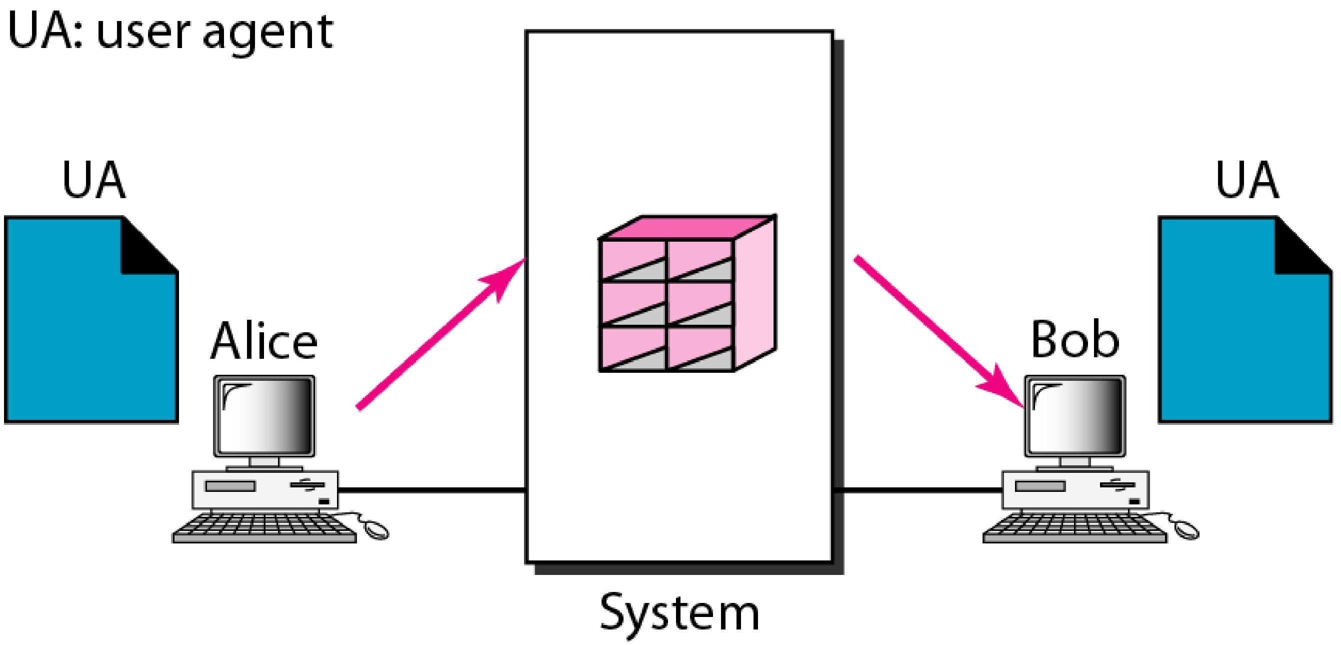
Topics discussed in this section:

Architecture
User Agent
Message Transfer Agent: SMTP
Message Access Agent: POP and IMAP
Web-Based Mail

User Agent:-

1. The first component of an electronic mail system.
2. It provides service to the user to make the process of sending and receiving a message easier.

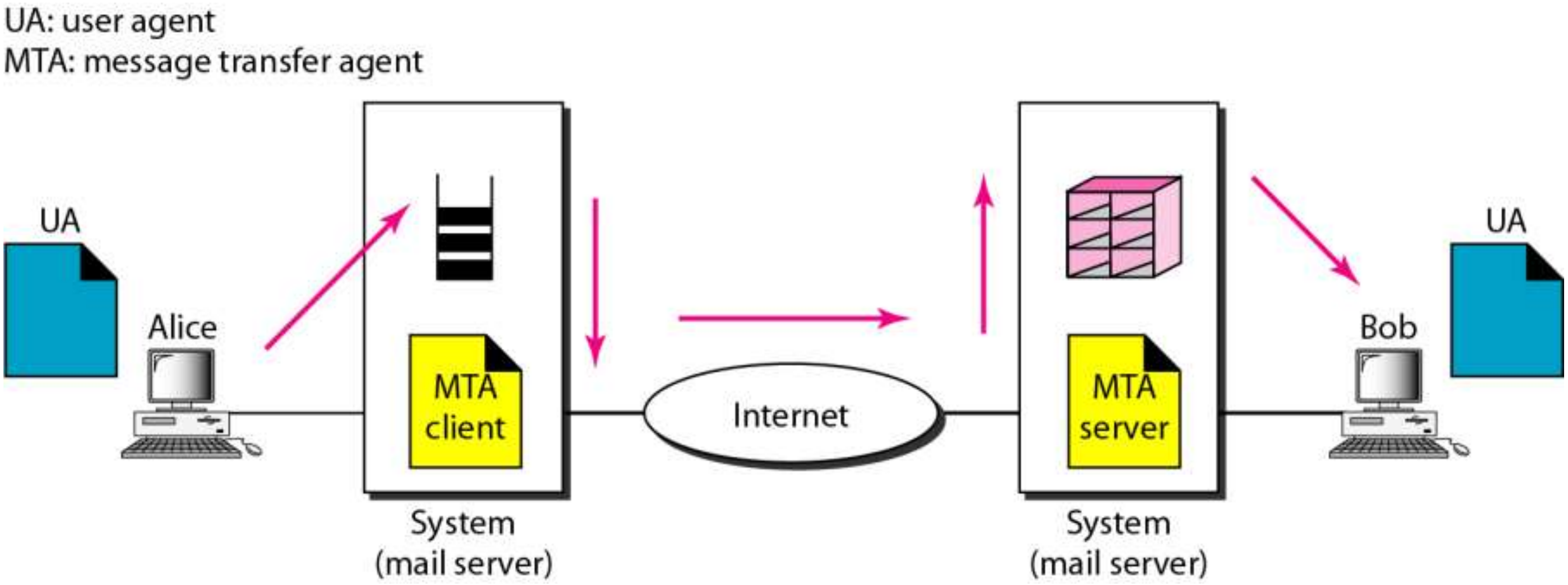
Figure 26.6 Architecture-*First scenario in electronic mail*



Note

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

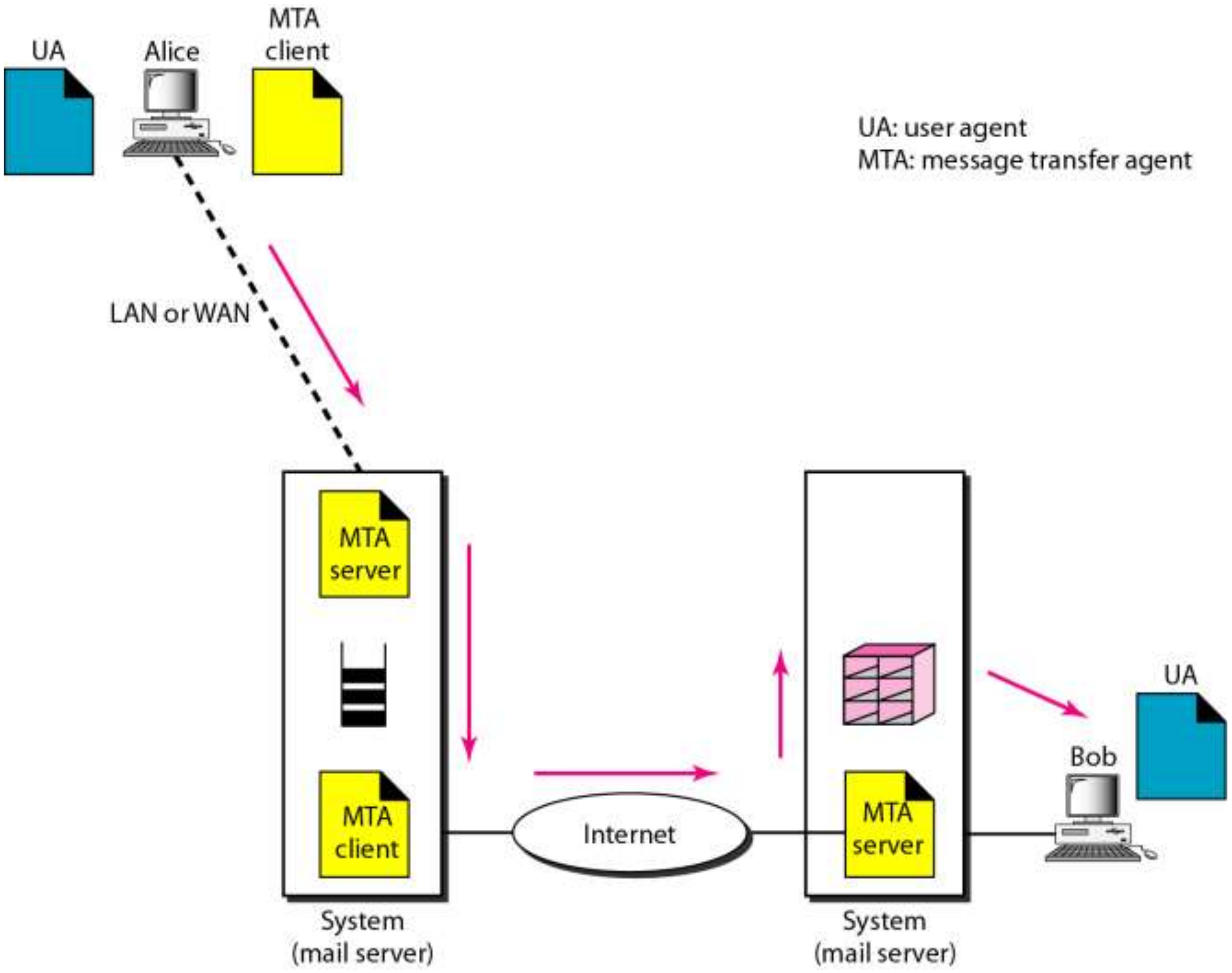
Figure 26.7 *Second scenario in electronic mail*



Note

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

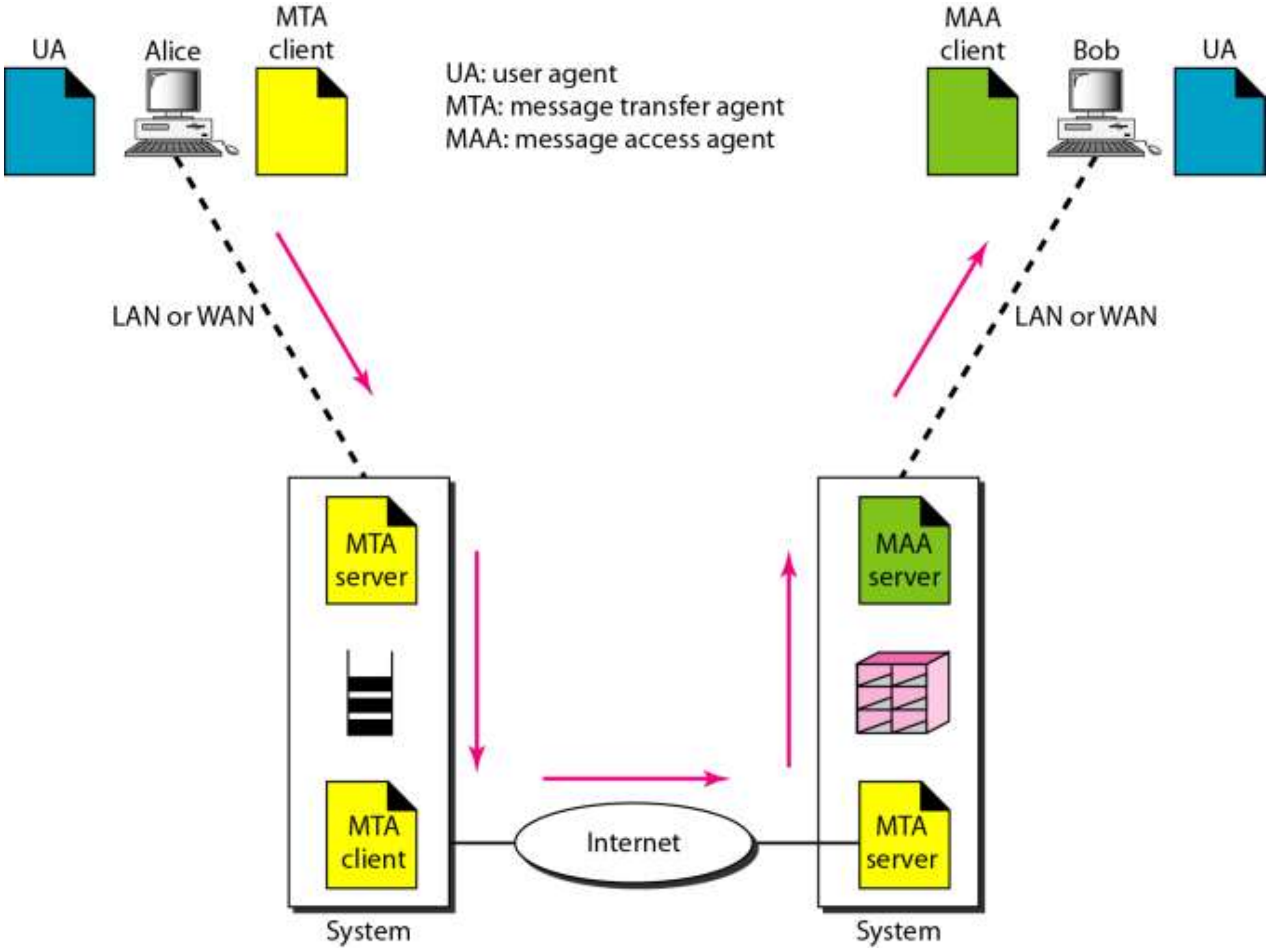
Figure 26.8 *Third scenario in electronic mail*



Note

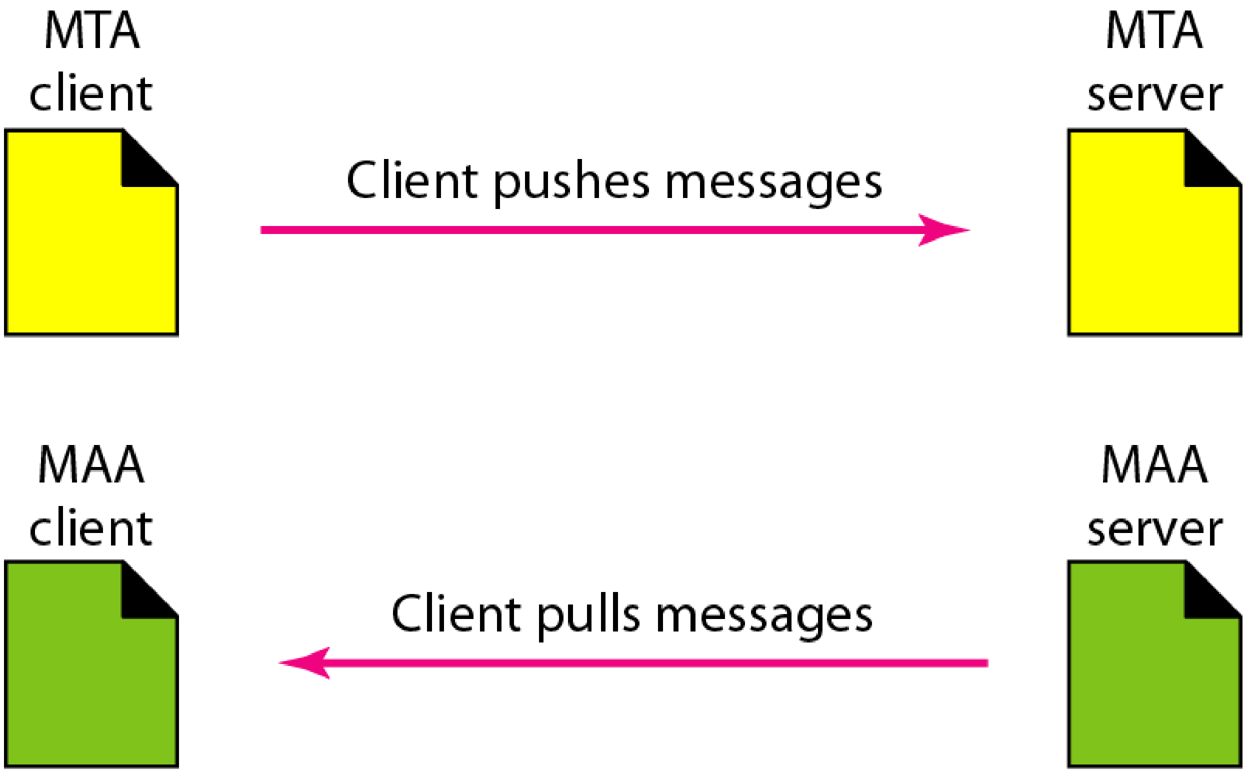
When the sender is connected to the mail server via a LAN or a WAN, we need two UAs and two pairs of MTAs (client and server).

Figure 26.9 *Fourth scenario in electronic mail*



UA: user agent
MTA: message transfer agent
MAA: message access agent

Figure 26.10 *Push versus pull in electronic email*

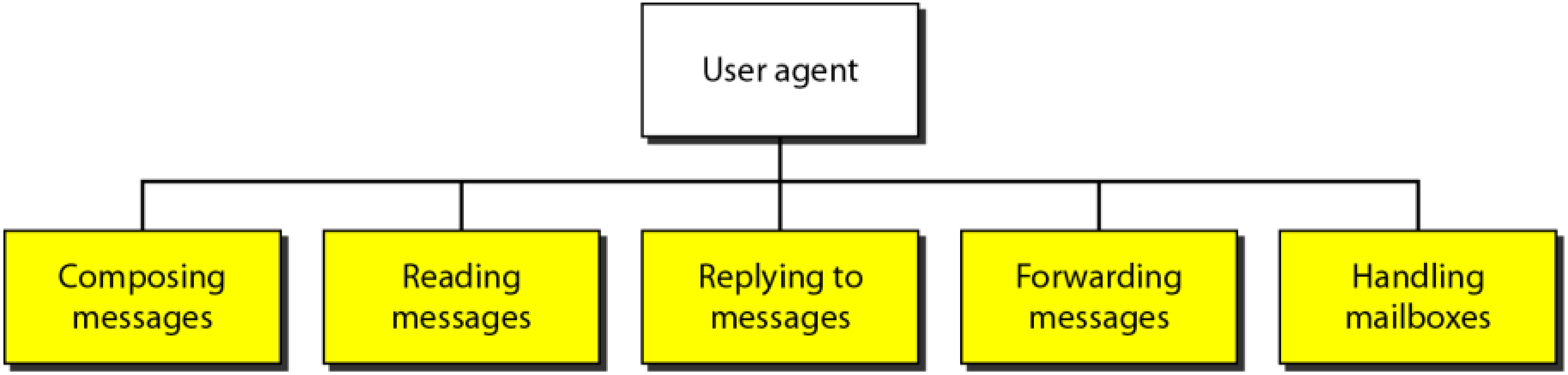


Note

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 and a pair of MAAs.

This is the most common situation today.

Figure 26.11 *Services of user agent*



User Agent Types:-

1. Command- Driven: Accepts a one-character command from the keyboard.

   r- to reply to the sender

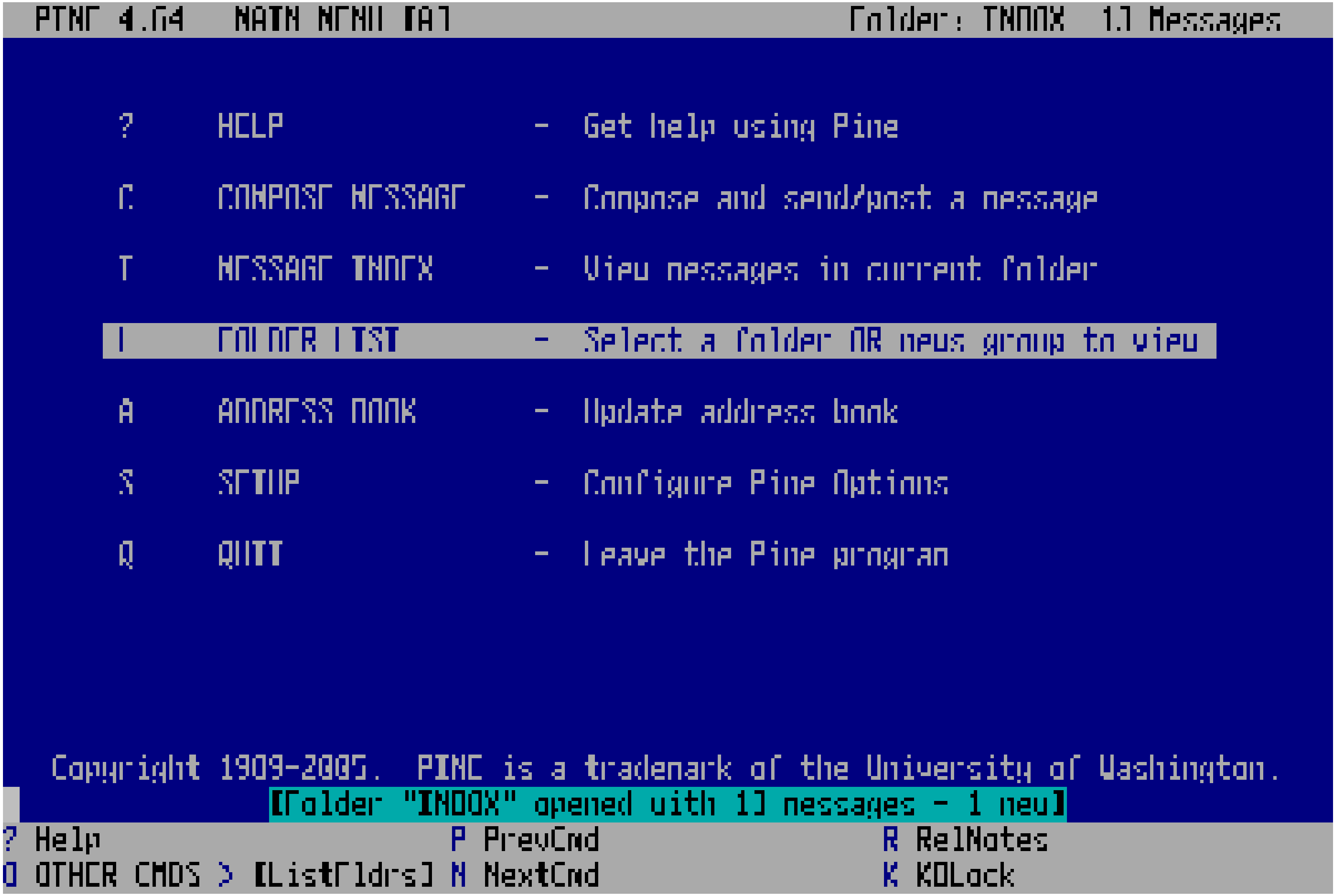
   R- to reply to sender and all recipients.

2. GUI – Based:

They contain GUI components that allow the user to interact with the software by using keyboard and mouse.

Note

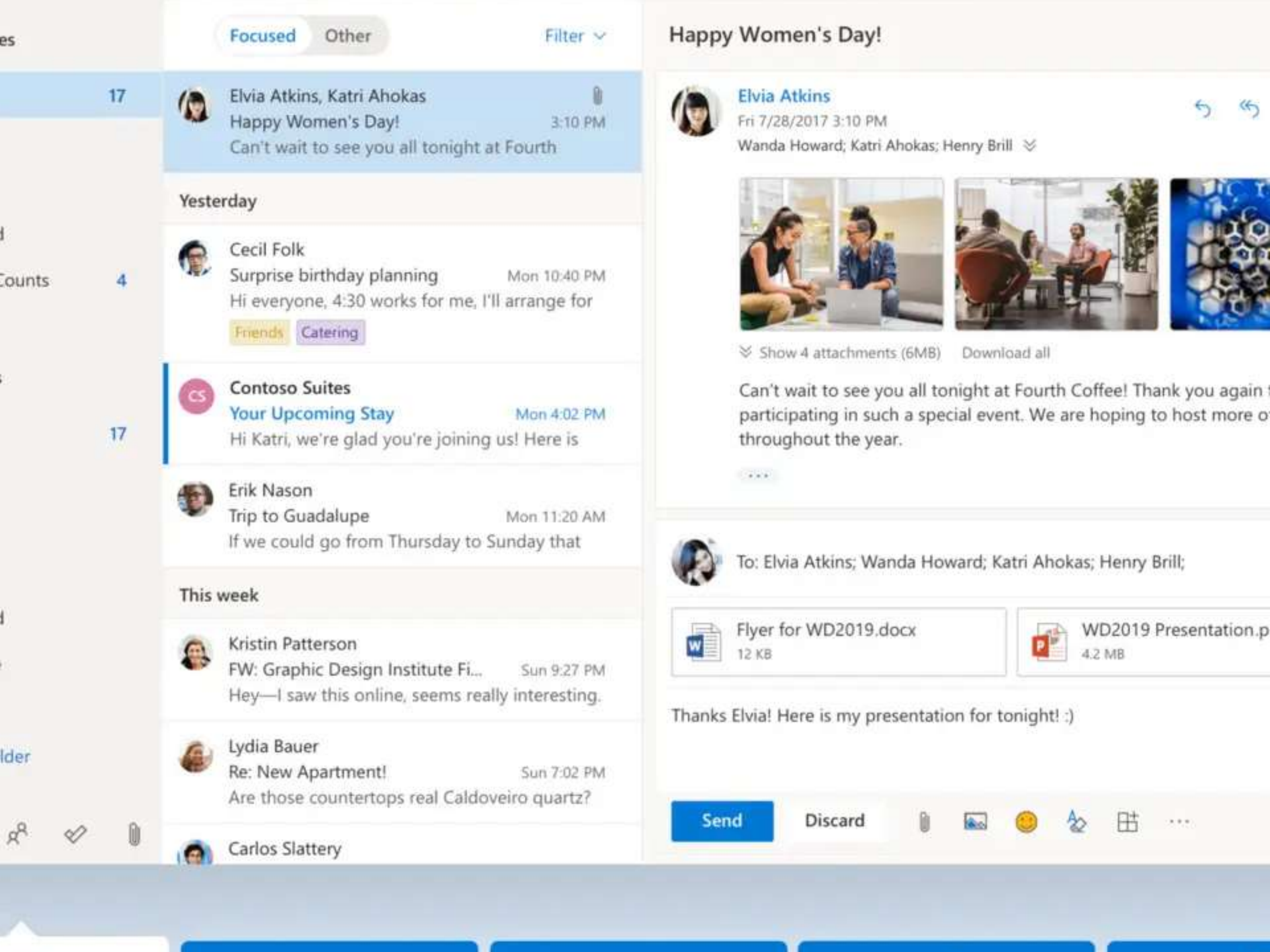
Some examples of command-driven user agents are *mail*, *pine*, and *elm*.



```
  PINE 4.64   MAIN MENU [A]                               Folder: INBOX   11 Messages


      ?     HELP                  -  Get help using Pine

      C     COMPOSE MESSAGE       -  Compose and send/post a message

      I     MESSAGE INDEX         -  View messages in current folder

      L     FOLDER LIST           -  Select a folder OR news group to view

      A     ADDRESS BOOK          -  Update address book

      S     SETUP                 -  Configure Pine Options

      Q     QUIT                  -  Leave the Pine program




   Copyright 1989-2005.  PINE is a trademark of the University of Washington.
                  [Folder "INBOX" opened with 11 messages - 1 new]
? Help                      P PrevCmd                  R RelNotes
O OTHER CMDS > [ListFldrs]  N NextCmd                  K KBLock
```

Note

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



Elvia Atkins, Katri Ahokas
Happy Women's Day! 3:10 PM
Can't wait to see you all tonight at Fourth

Yesterday

Cecil Folk
Surprise birthday planning Mon 10:40 PM
Hi everyone, 4:30 works for me, I'll arrange for
Friends Catering

Contoso Suites
Your Upcoming Stay Mon 4:02 PM
Hi Katri, we're glad you're joining us! Here is

Erik Nason
Trip to Guadalupe Mon 11:20 AM
If we could go from Thursday to Sunday that

This week

Kristin Patterson
FW: Graphic Design Institute Fi... Sun 9:27 PM
Hey—I saw this online, seems really interesting.

Lydia Bauer
Re: New Apartment! Sun 7:02 PM
Are those countertops real Caldoveiro quartz?

Carlos Slattery

Happy Women's Day!

Elvia Atkins
Fri 7/28/2017 3:10 PM
Wanda Howard; Katri Ahokas; Henry Brill

Show 4 attachments (6MB) Download all

Can't wait to see you all tonight at Fourth Coffee! Thank you again f
participating in such a special event. We are hoping to host more o
throughout the year.

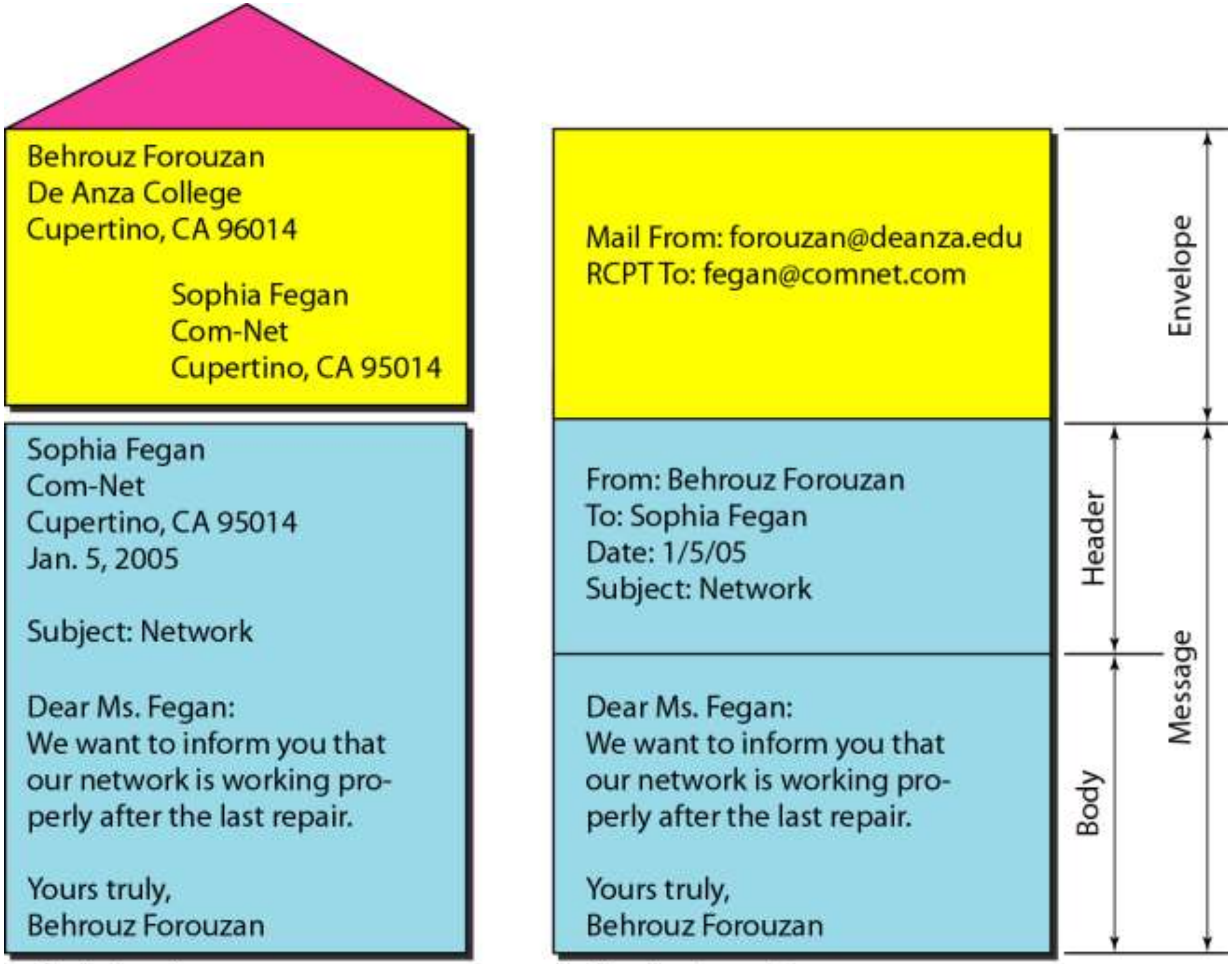
To: Elvia Atkins; Wanda Howard; Katri Ahokas; Henry Brill;

Flyer for WD2019.docx
12 KB

WD2019 Presentation.p
4.2 MB

Thanks Elvia! Here is my presentation for tonight! :)

Figure 26.12 *Format of an e-mail*



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

a. Postal mail

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

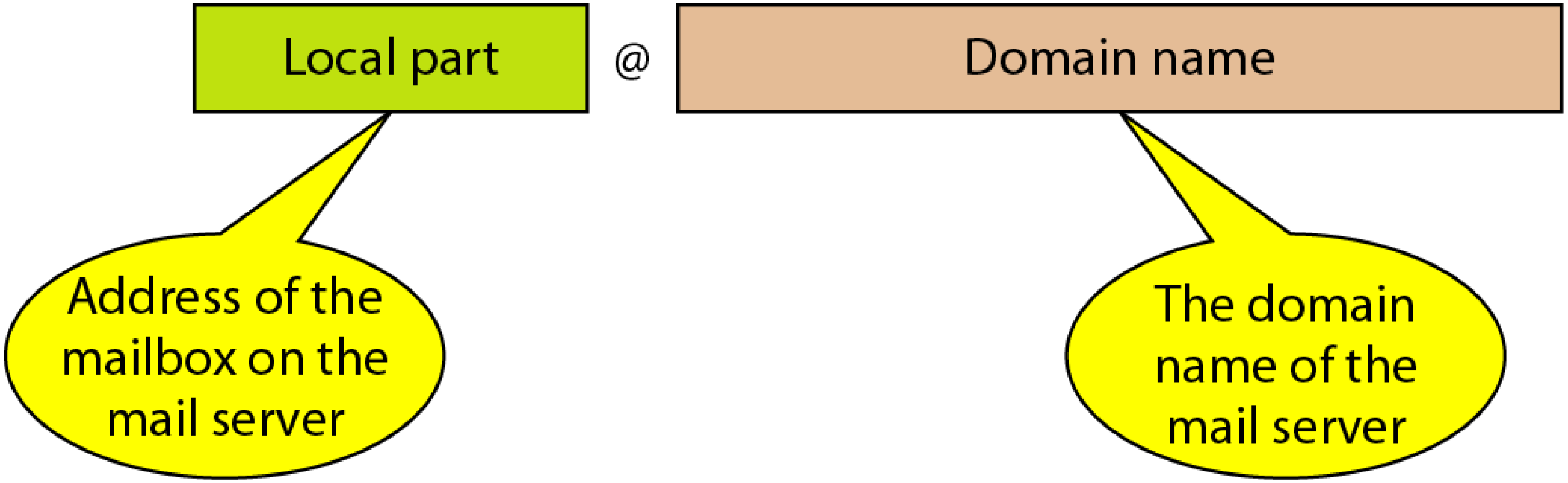
From: Behrouz Forouzan
To: Sophia Fegan
Date: 1/5/05
Subject: Network

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

Yours truly,
Behrouz Forouzan

b. Electronic mail

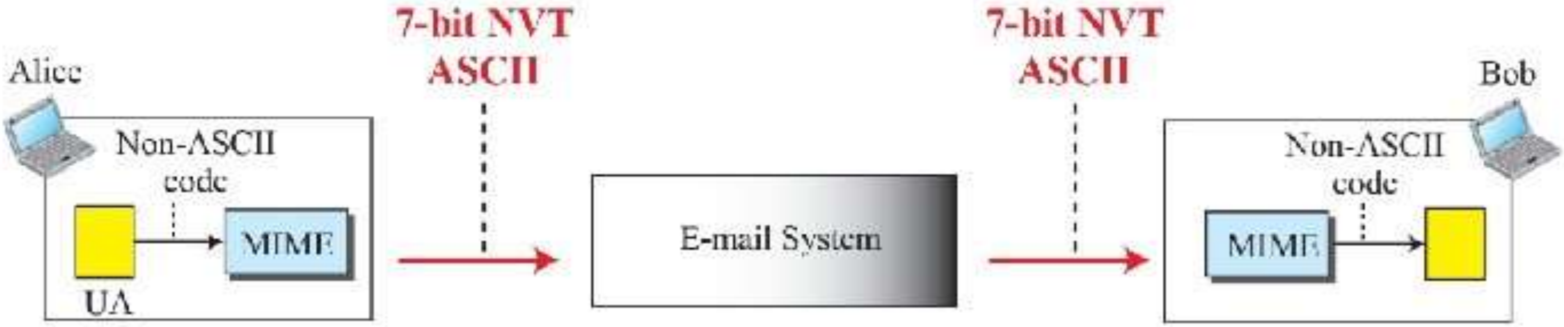
Figure 26.13 *E-mail address*


Local part
@
Domain name
Address of the mailbox on the mail server
The domain name of the mail server

MIME- Multipurpose Internet Mail Extensions

1. It is a supplementary protocol that allows non-ASCII data to be sent through email.
2. Transforms NON ASCII data to ASCII and delivers to client MTA to be sent through Internet.

Figure 26.18: MIME



Electronic mail can send messages only in NVT 7-bit ASCII format. It cannot be used for languages other than English or to send binary files or video or audio data.

Multipurpose Internet Mail Extensions (MIME) is a supplementary protocol that transforms non-ASCII data at the sender site to NVT ASCII data and delivers it to the client MTA to be sent through the Internet. The message at the receiving site is transformed back to the original data.

7-Bit ASCII Code Table

Rightmost			**Leftmost Three Bits**					
Four Bits	**000**	**001**	**010**	**011**	**100**	**101**	**110**	**111**
0000	NUL	DLE	Space	0	@	P	`	p
0001	SOH	DC1	!	1	A	Q	a	q
0010	STX	DC2	"	2	B	R	b	r
0011	ETX	DC3	#	3	C	S	c	s
0100	EOT	DC4	$	4	D	T	d	t
0101	ENQ	NAK	%	5	E	U	e	u
0110	ACK	SYN	&	6	F	V	f	v
0111	BEL	ETB	'	7	G	W	g	w
1000	BS	CAN	(	8	H	X	h	x
1001	HT	EM	)	9	I	Y	i	y
1010	LF	SUB	*	:	J	Z	j	z
1011	VT	ESC	+	;	K	[	k	{
1100	FF	FS	,	<	L	\	l	|
1101	CR	GS	-	=	M	]	m	}
1110	SO	RS	.	>	N	^	n	~
1111	SI	US	/	?	O	_	o	DEL

Figure 26.14 *MIME*

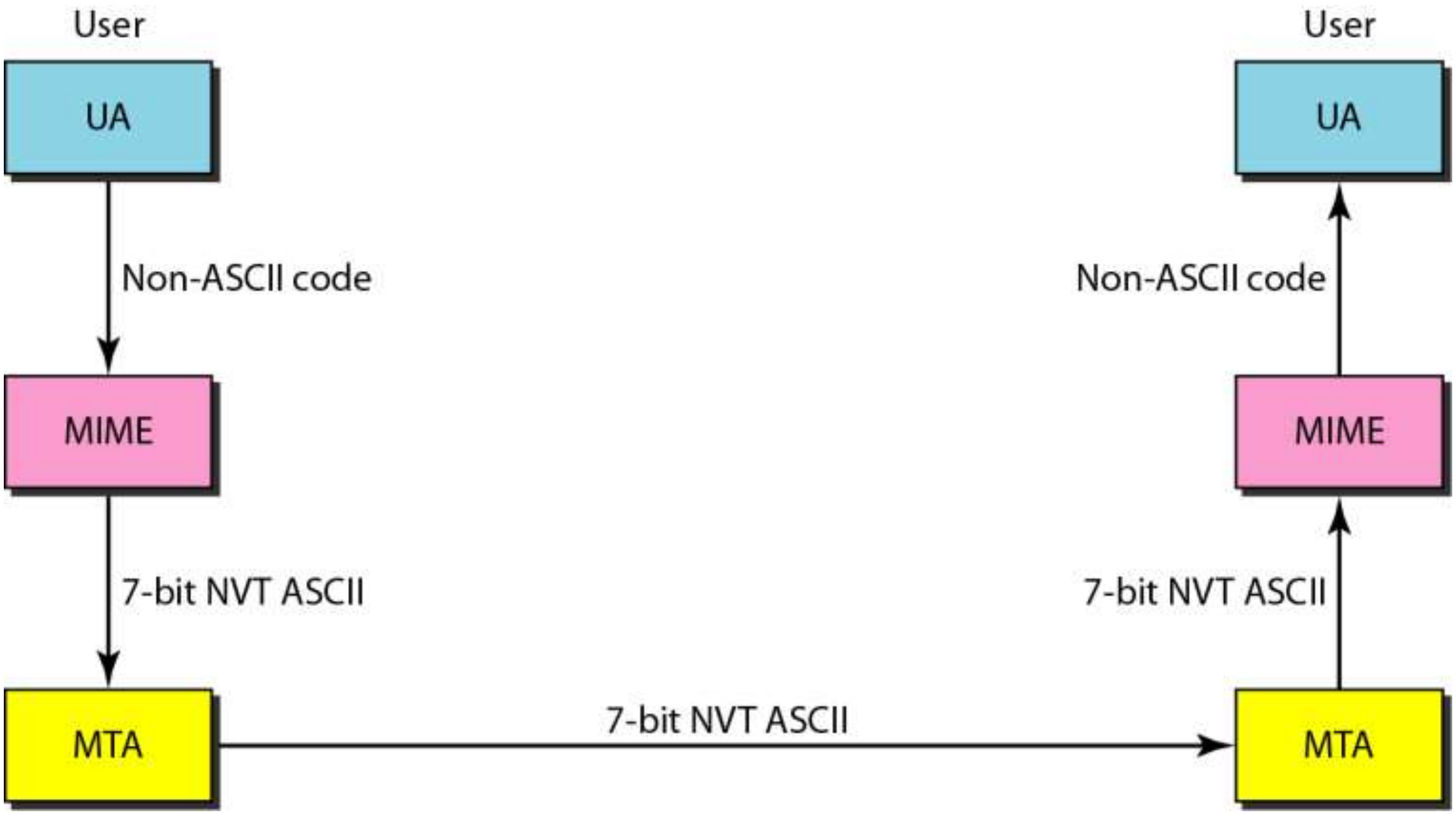
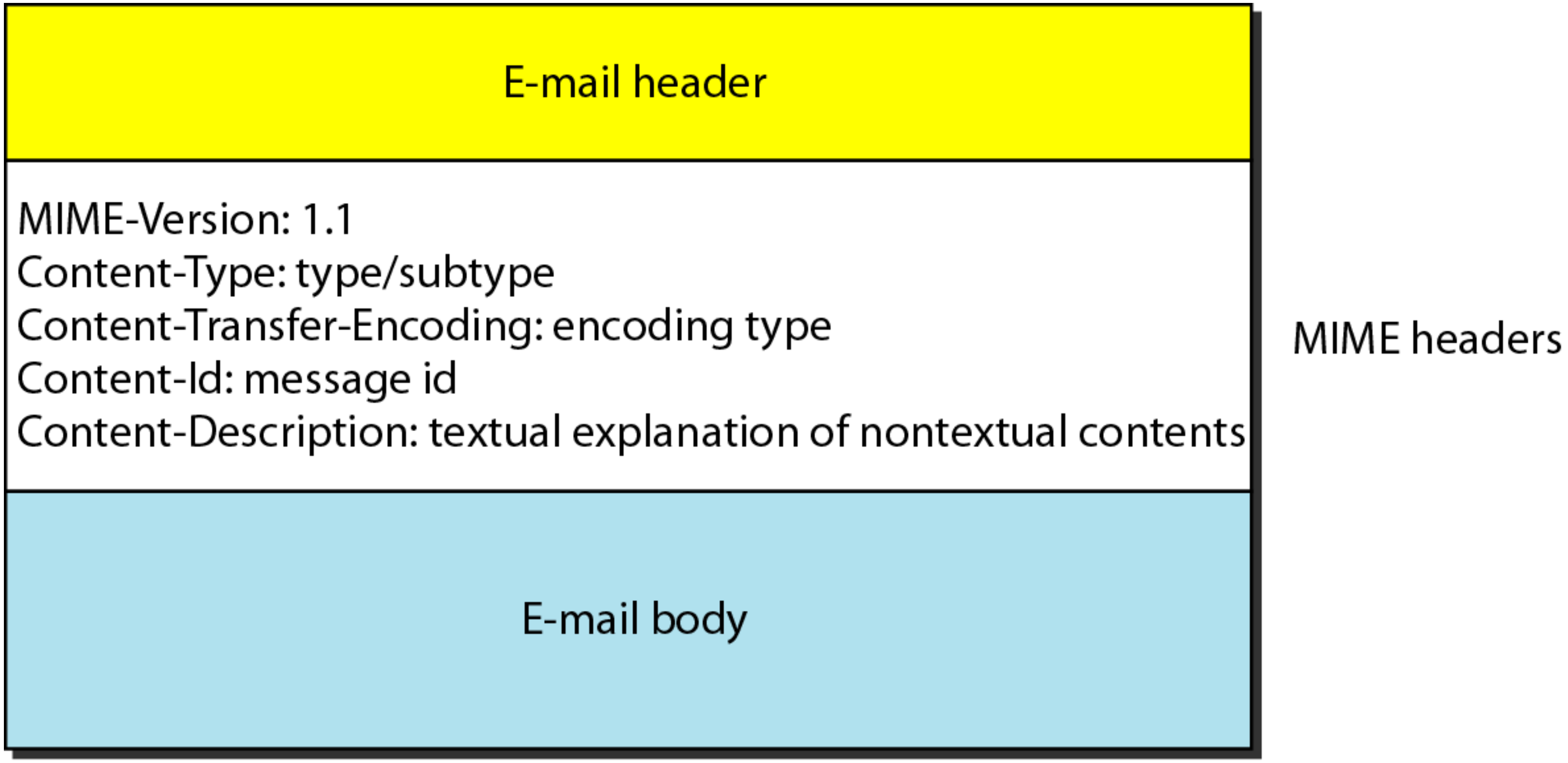


Figure 26.15 *MIME header*



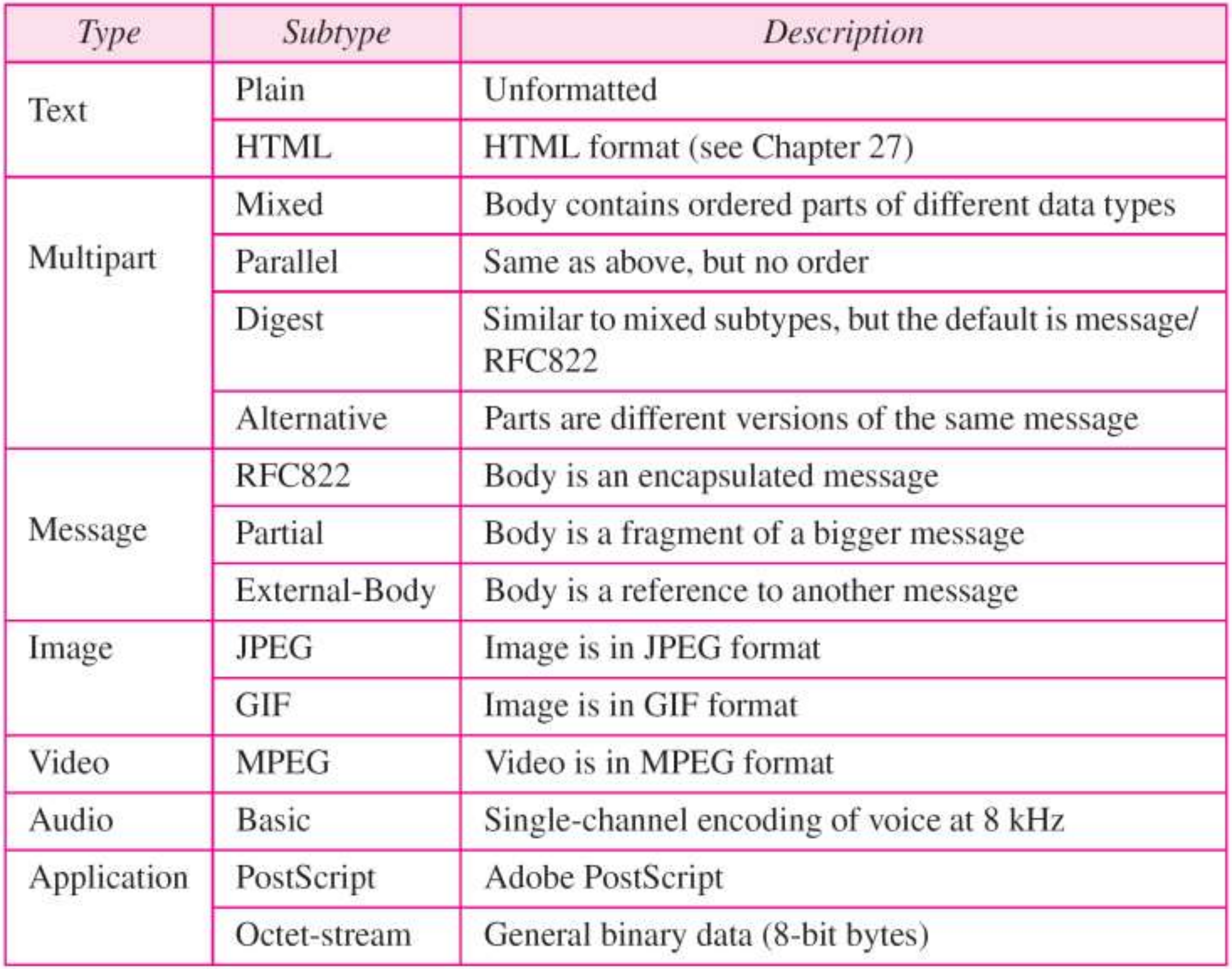
E-mail header

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

MIME headers

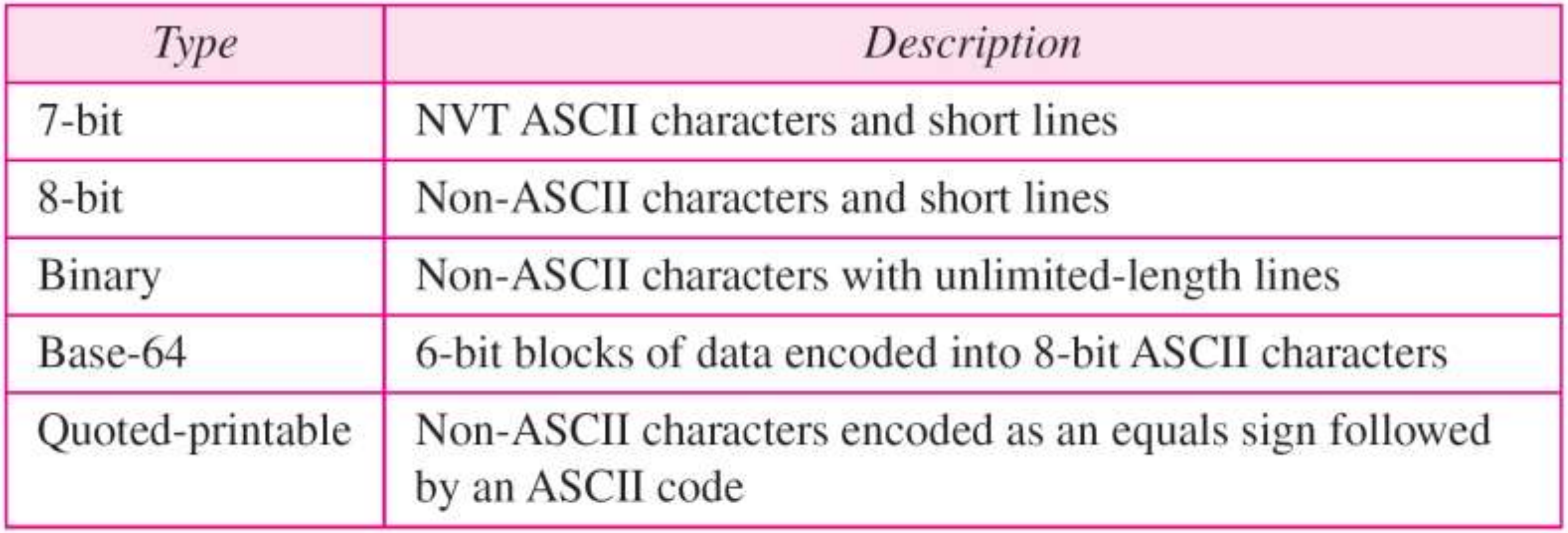
E-mail body

Table 26.5 Content Types- *Data types and subtypes in MIME*

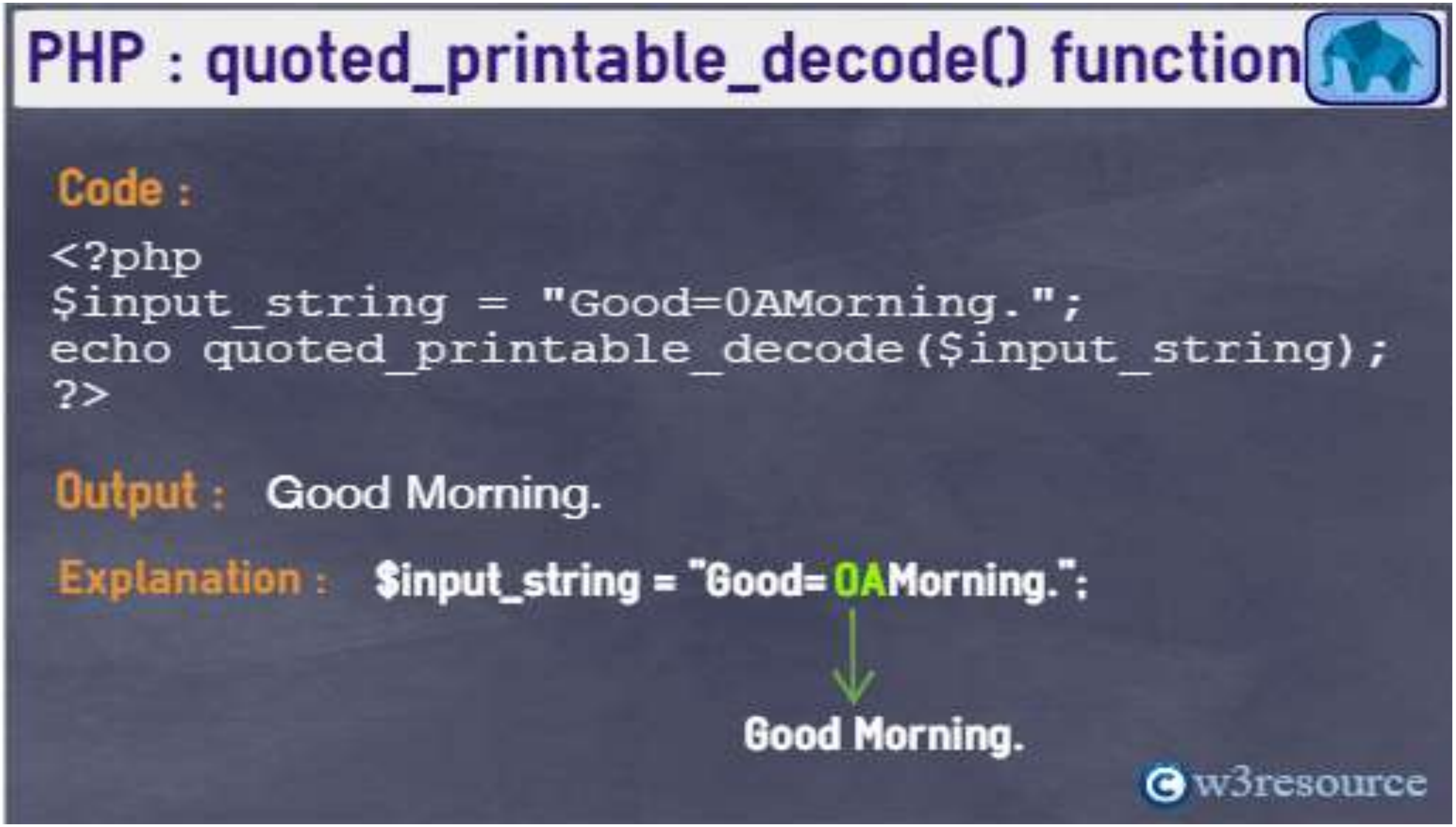


Type	*Subtype*	*Description*
Text	Plain	Unformatted
	HTML	HTML format (see Chapter 27)
Multipart	Mixed	Body contains ordered parts of different data types
	Parallel	Same as above, but no order
	Digest	Similar to mixed subtypes, but the default is message/ RFC822
	Alternative	Parts are different versions 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 (8-bit bytes)

Table 26.6 ***Content-transfer-encoding- Method used to encode the messages into 0s and 1s for transport***



Type	*Description*
7-bit	NVT ASCII characters and short lines
8-bit	Non-ASCII characters and short lines
Binary	Non-ASCII characters with unlimited-length lines
Base-64	6-bit blocks of data encoded into 8-bit ASCII characters
Quoted-printable	Non-ASCII characters encoded as an equals sign followed by an ASCII code



PHP : quoted_printable_decode() function

Code :

```
<?php
$input_string = "Good=0AMorning.";
echo quoted_printable_decode($input_string);
?>
```

Output : Good Morning.

Explanation : $input_string = "Good=0AMorning.";

↓

Good Morning.

Simple Mail Transfer Protocol (SMTP)

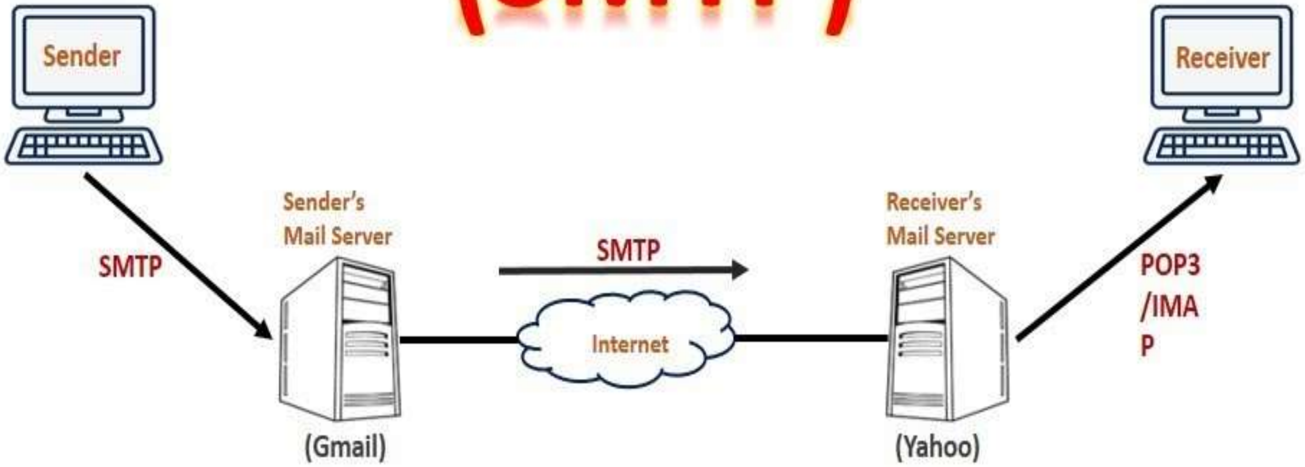


Figure 26.16 *SMTP – Formal protocol that defines MTA Client and MTA Server*

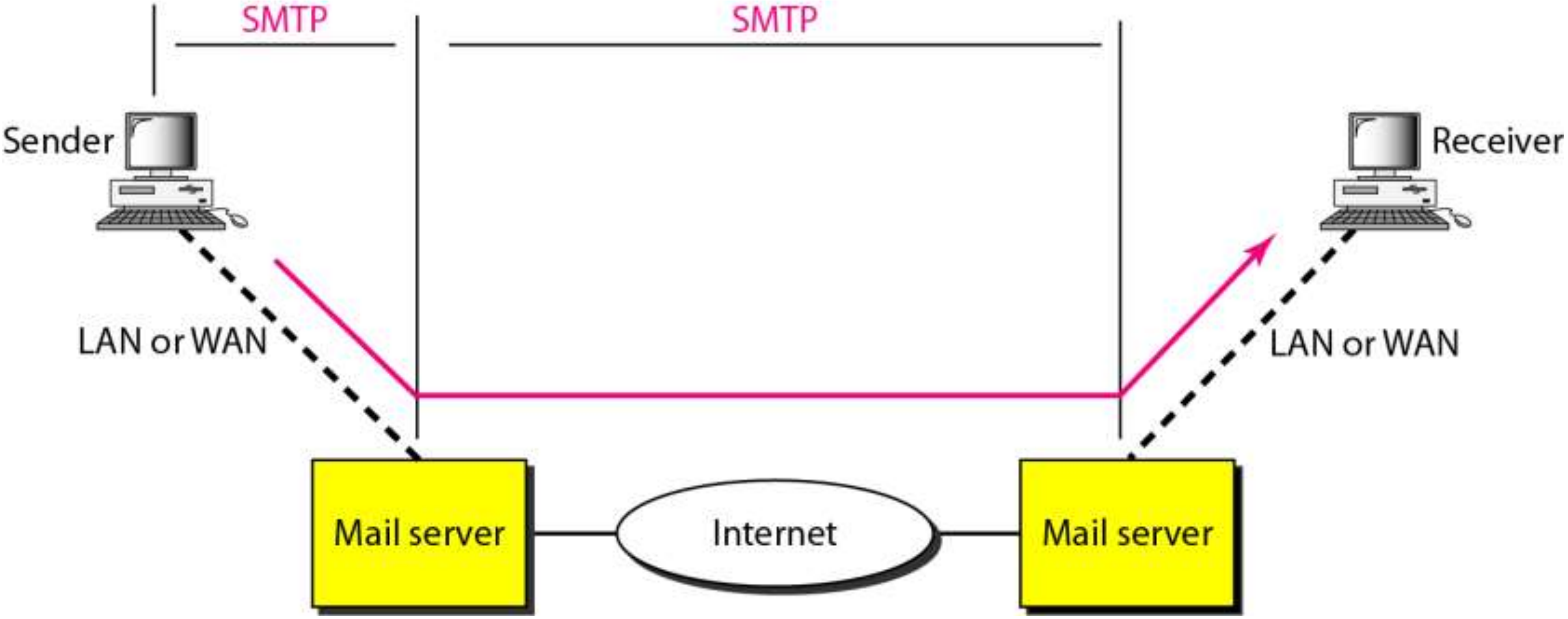


Figure 26.17 *Commands and responses – to transfer messages between MTA client and MTA Server*

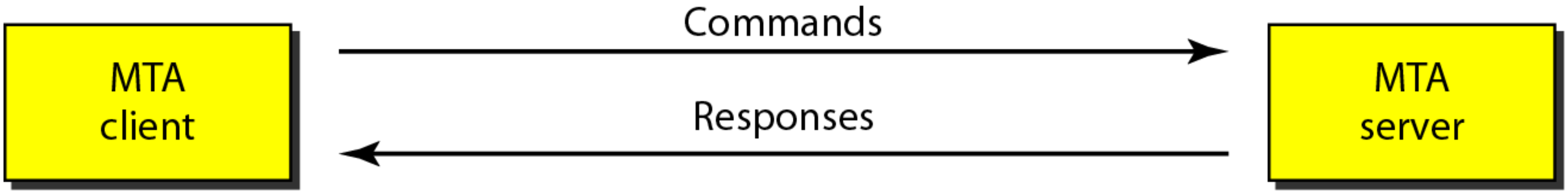
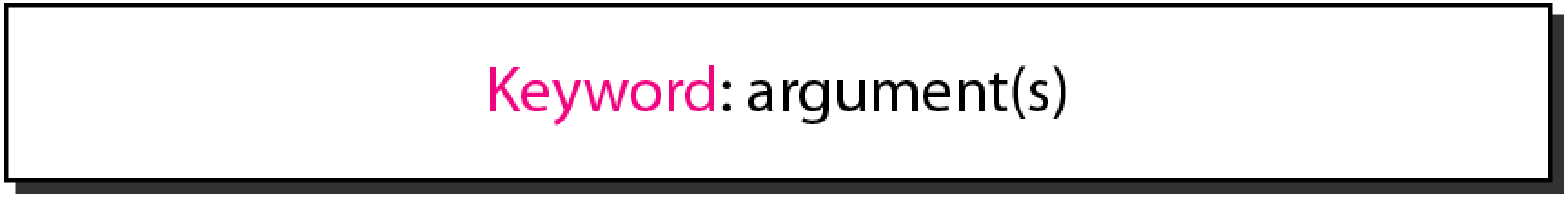
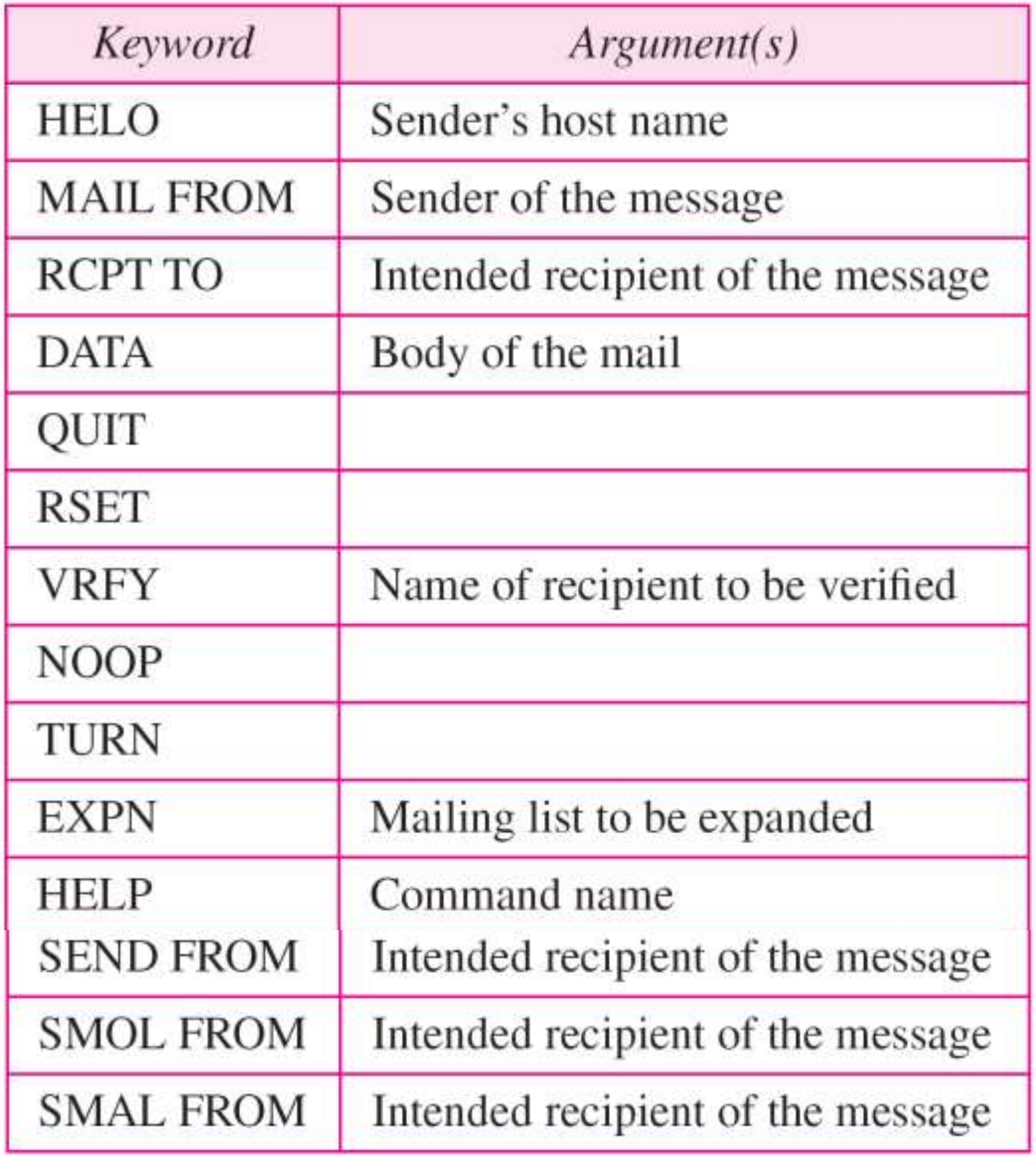


Figure 26.18 *Command format- Commands are sent from Client to Server*



Keyword: argument(s)

Table 26.7 SMTP *Commands – 14*



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

Status
Switch

SEND FROM

This command specifies that the mail is to be delivered to the recipient's terminal and no to the mailbox. If the recipient is offline, then the mail is bounced.

SEND FROM: sender@gmail.com

SMOL FROM

This command specifies that the mail is to be delivered to the terminal or the recipient's mailbox. If the recipient is online, then the mail is sent directly to the terminal. If the recipient is offline, the mail is sent to the mailbox.

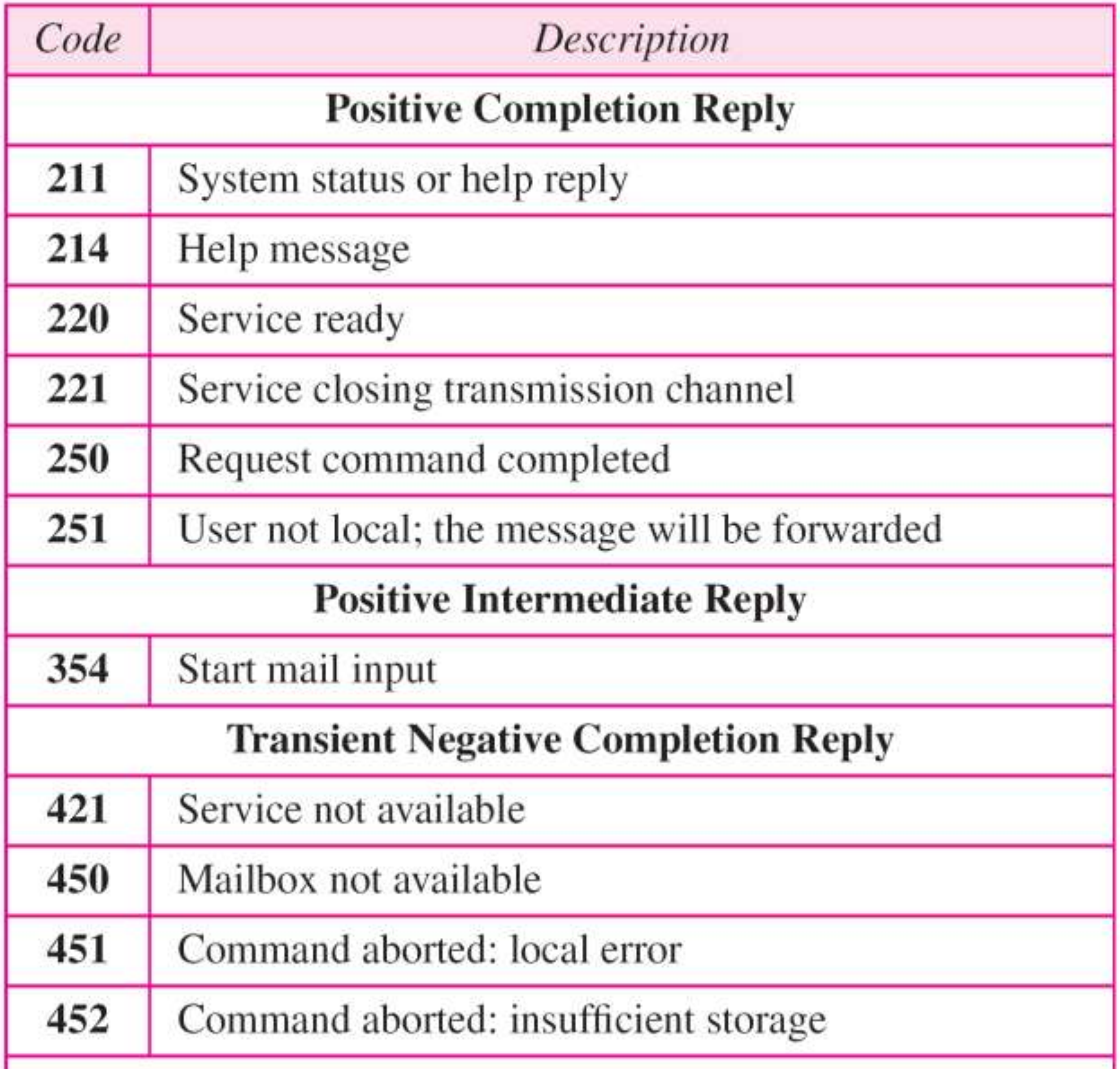
SMOL FROM: sender@gmail.com

SMAL FROM

This command specifies that the mail is to be delivered to the terminal and the recipient’s mailbox. If the recipient is offline, the mail is sent to the terminal and the mailbox. If a recipient is online, the mail is directly sent to the mailbox.

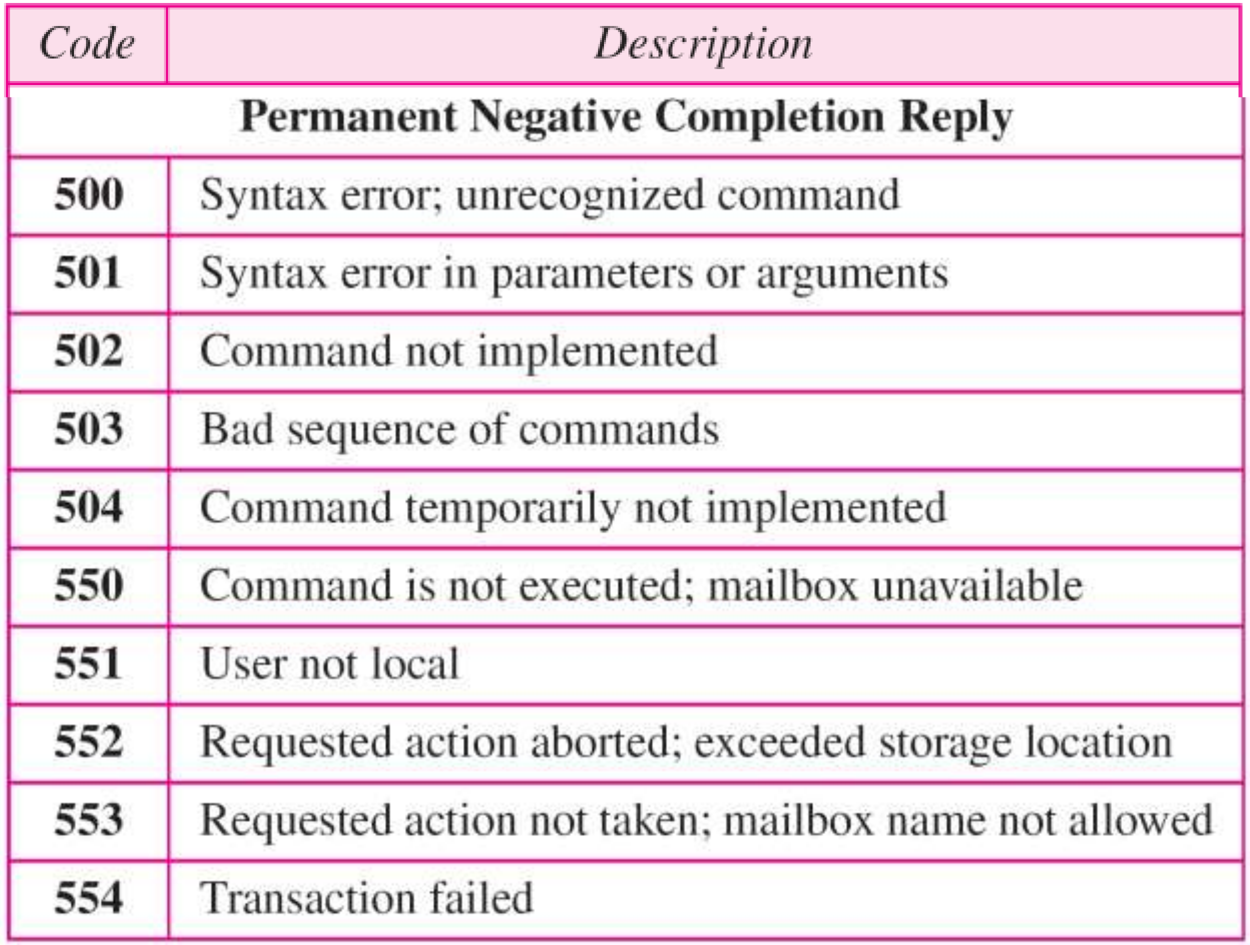
SMAL FROM: sender@gmail.com

Table 26.8 *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

Table 26.8 *Responses* *(continued)*



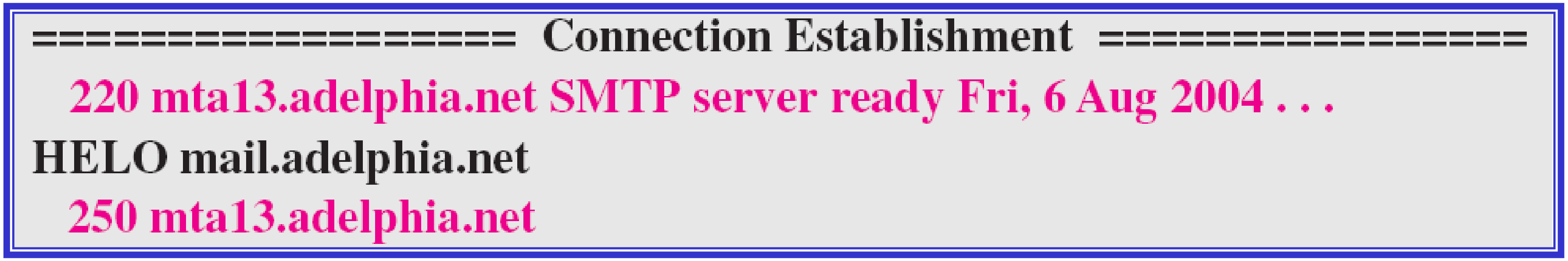
Code	*Description*
Permanent Negative Completion Reply	
500	Syntax error; unrecognized command
501	Syntax error in parameters or arguments
502	Command not implemented
503	Bad sequence of commands
504	Command temporarily not implemented
550	Command is not executed; mailbox unavailable
551	User not local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

Example 26.3

Let us see how we can directly use SMTP to send an e-mail and simulate the commands and responses we described in this section. We use TELNET to log into port 25 (the well-known port for SMTP). We then use the commands directly to send an e-mail. In this example, forouzanb@adelphia.net is sending an e-mail to himself. The first few lines show TELNET trying to connect to the Adelphia mail server. After connection, we can type the SMTP commands and then receive the responses, as shown on the next slide. Note that we have added, for clarification, some comment lines, designated by the "=" signs. These lines are not part of the e-mail procedure.

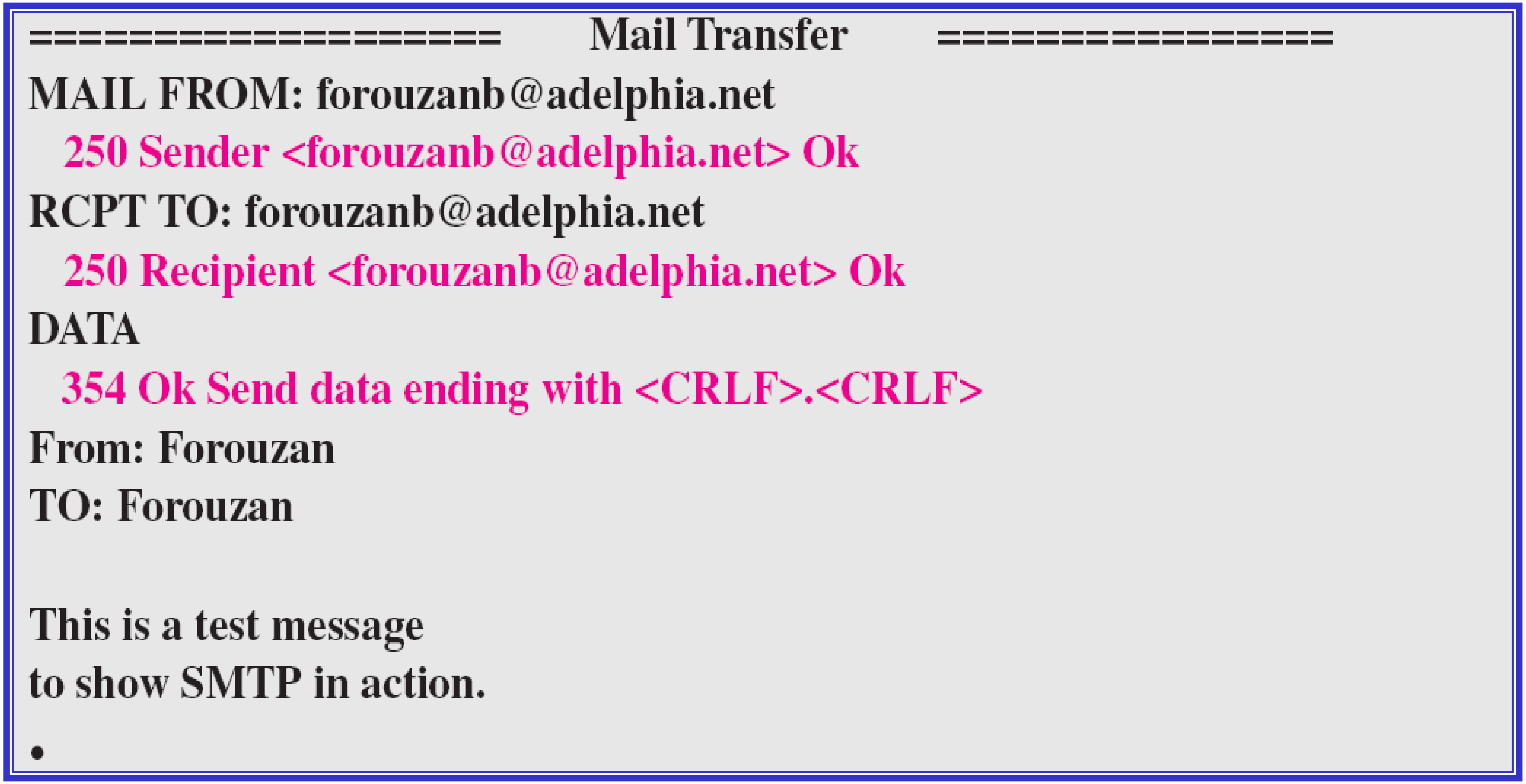
Example 26.3 (continued)

```
$ telnet mail.adelphia.net 25
Trying 68.168.78.100 . . .
Connected to mail.adelphia.net (68.168.78.100).
```



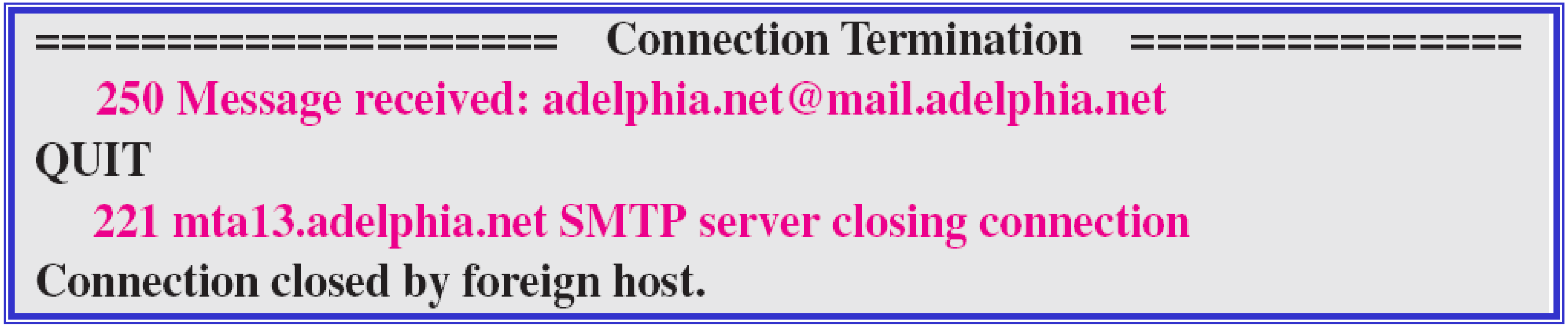
```
===================  Connection Establishment  =================
   220 mta13.adelphia.net SMTP server ready Fri, 6 Aug 2004 . . .
HELO mail.adelphia.net
   250 mta13.adelphia.net
```

Example 26.3 (continued)



```
====================      Mail Transfer      ==================
MAIL FROM: forouzanb@adelphia.net
   250 Sender <forouzanb@adelphia.net> Ok
RCPT TO: forouzanb@adelphia.net
   250 Recipient <forouzanb@adelphia.net> Ok
DATA
   354 Ok Send data ending with <CRLF>.<CRLF>
From: Forouzan
TO: Forouzan

This is a test message
to show SMTP in action.
.
```

Example 26.3 (continued)



```
======================   Connection Termination   ================
    250 Message received: adelphia.net@mail.adelphia.net
QUIT
    221 mta13.adelphia.net SMTP server closing connection
Connection closed by foreign host.
```

POP3 (Post Office Protocol)- PULL PROTOCOL

- Emails are stored on a single device.
- Sent messages are stored on a single device.
- Emails can only be accessed from a single device.
- 2 modes- DELETE and KEEP
- If you want to keep messages on the server, make sure the setting "Keep email on server" is enabled or all messages are deleted from the server once downloaded to the app or software

IMAP 4(Internet Messaging Access Protocol Version 4)

- Emails are stored on the server.
- Sent messages are stored on the server.
- Messages can be synced and accessed across multiple devices.

Figure 26.19 *POP3 and IMAP4*

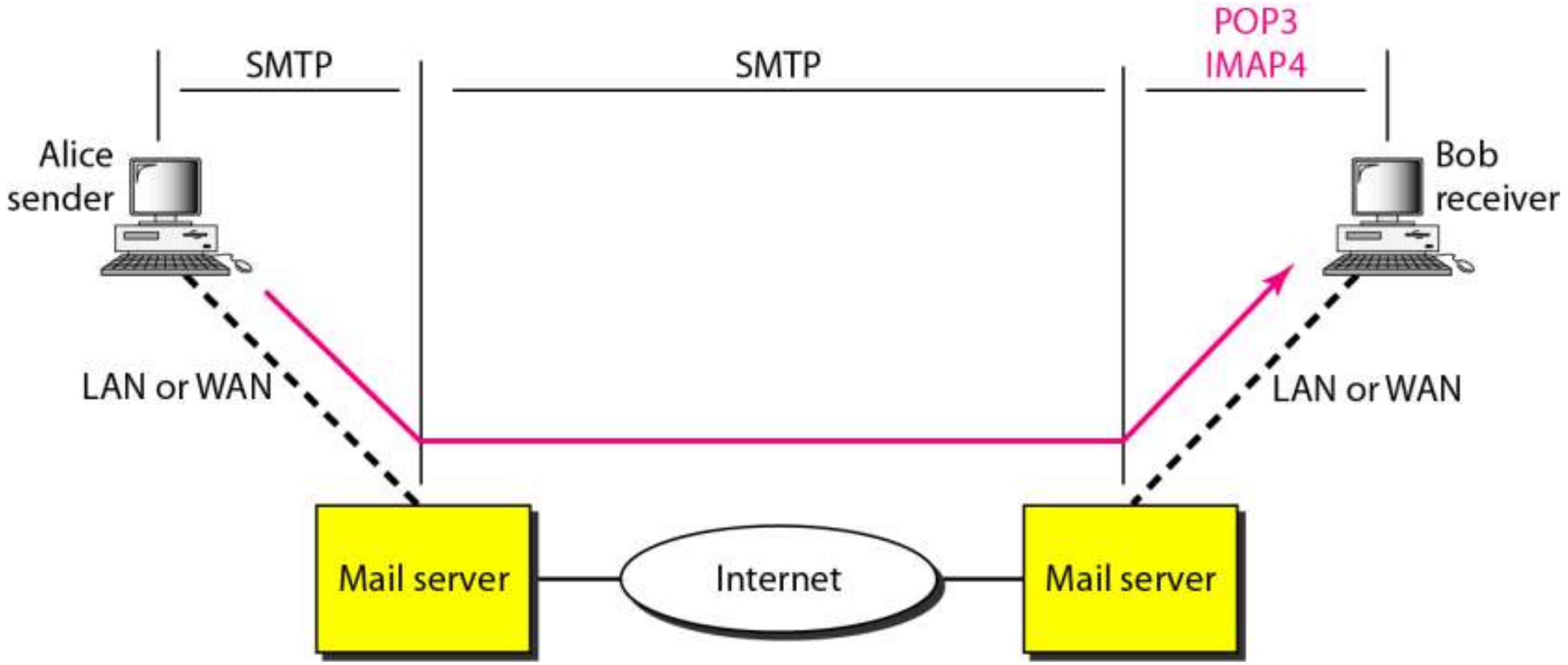
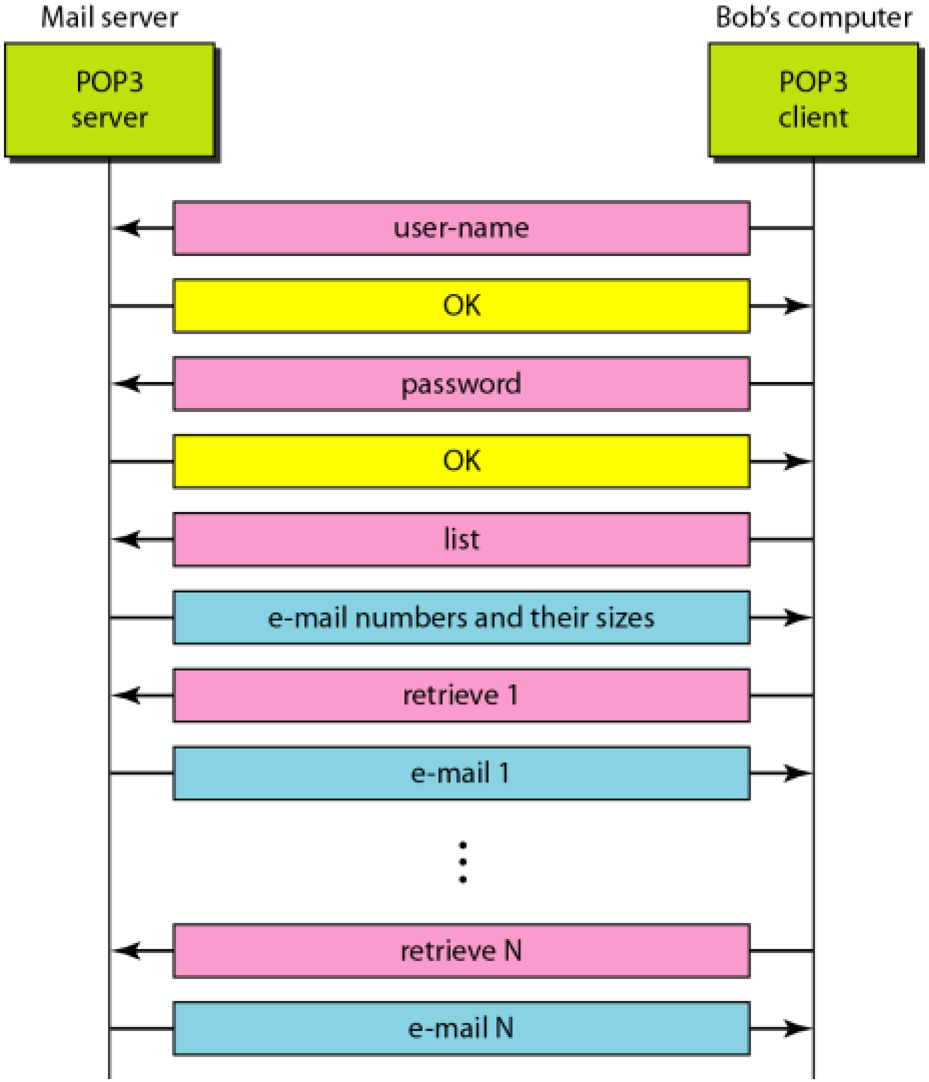
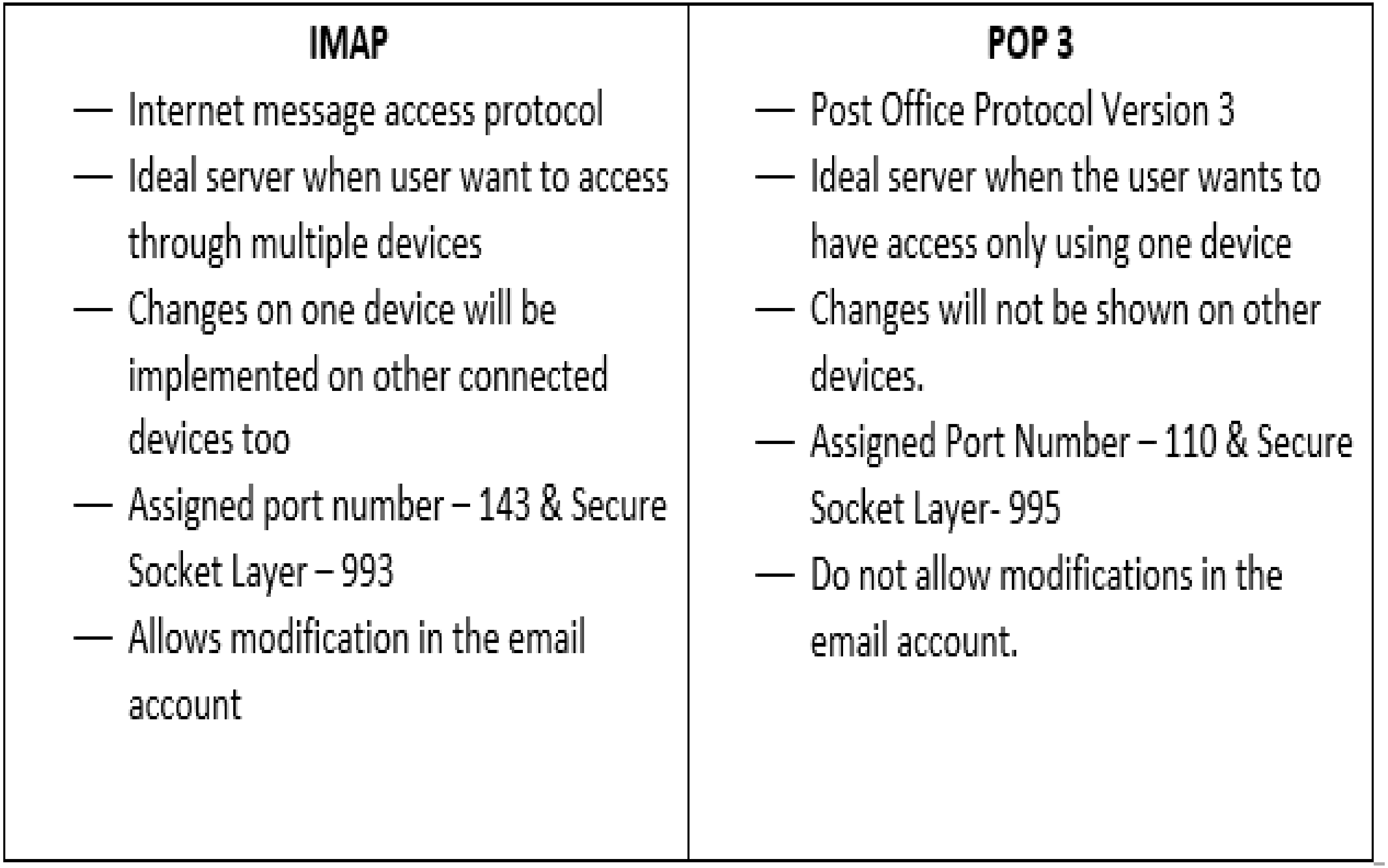


Figure 26.20 *The exchange of commands and responses in POP3*





IMAP	POP 3
— Internet message access protocol — Ideal server when user want to access through multiple devices — Changes on one device will be implemented on other connected devices too — Assigned port number – 143 & Secure Socket Layer – 993 — Allows modification in the email account	— Post Office Protocol Version 3 — Ideal server when the user wants to have access only using one device — Changes will not be shown on other devices. — Assigned Port Number – 110 & Secure Socket Layer- 995 — Do not allow modifications in the email account.

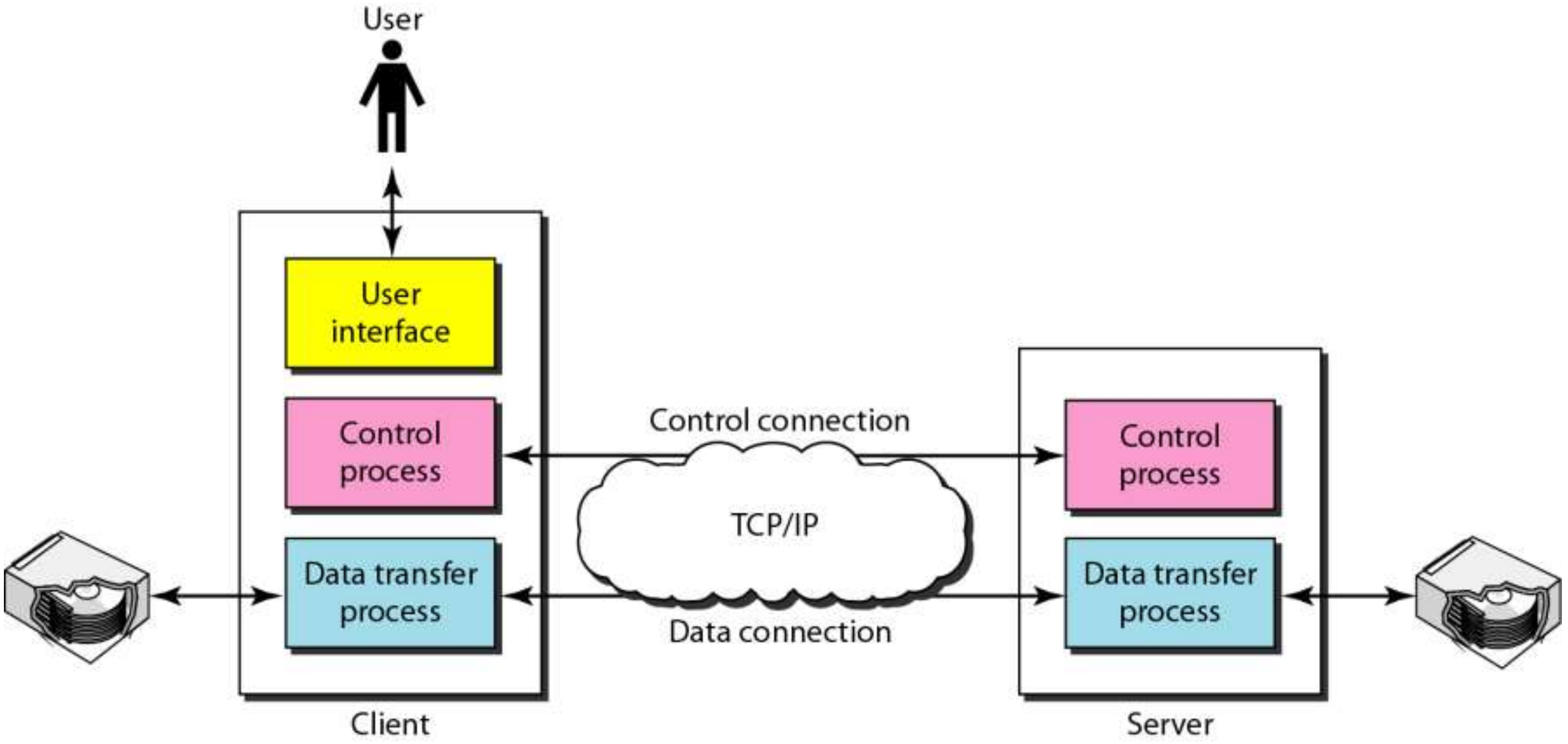
26-3 FILE TRANSFER

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

Topics discussed in this section:

File Transfer Protocol (FTP)
Anonymous FTP

Figure 26.21 *FTP*



Note

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.

Figure 26.22 *Using the control connection*

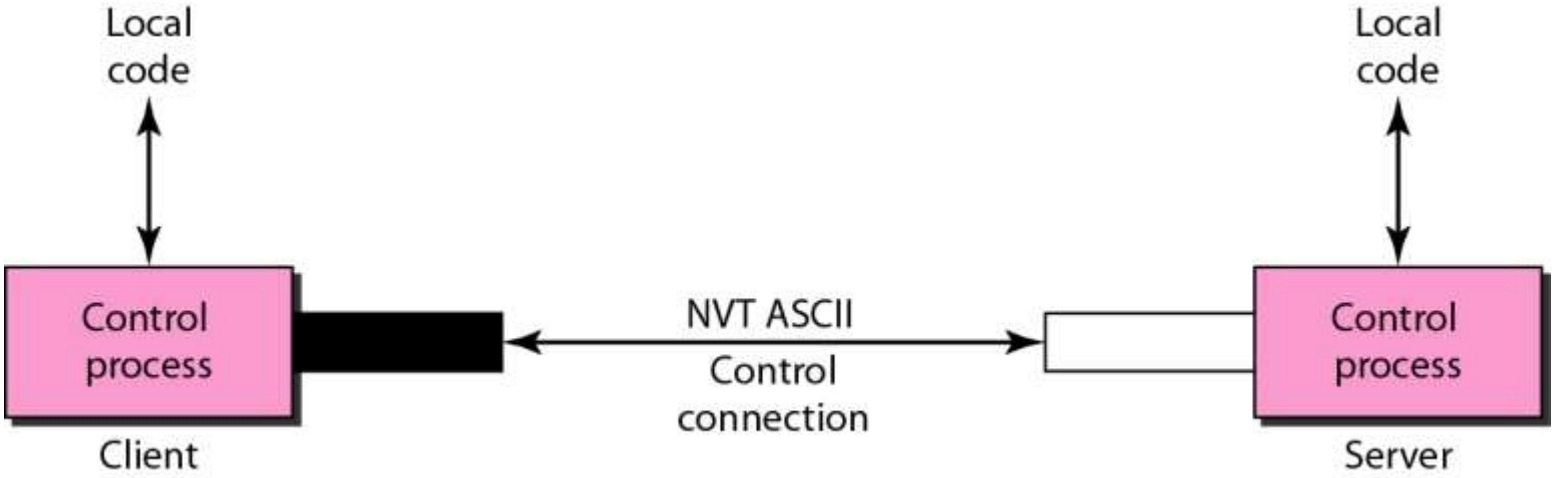
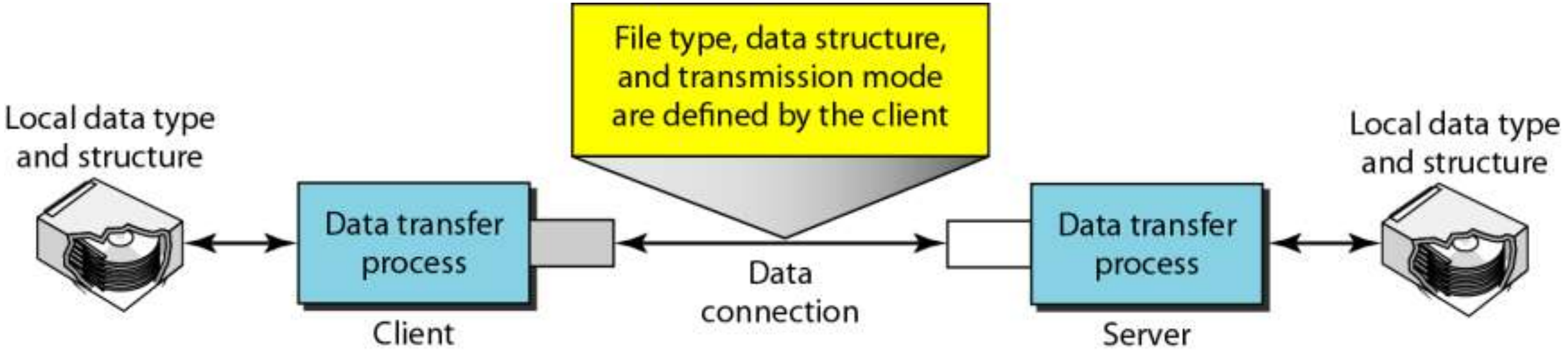


Figure 26.23 *Using the data connection*


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

Example 26.4

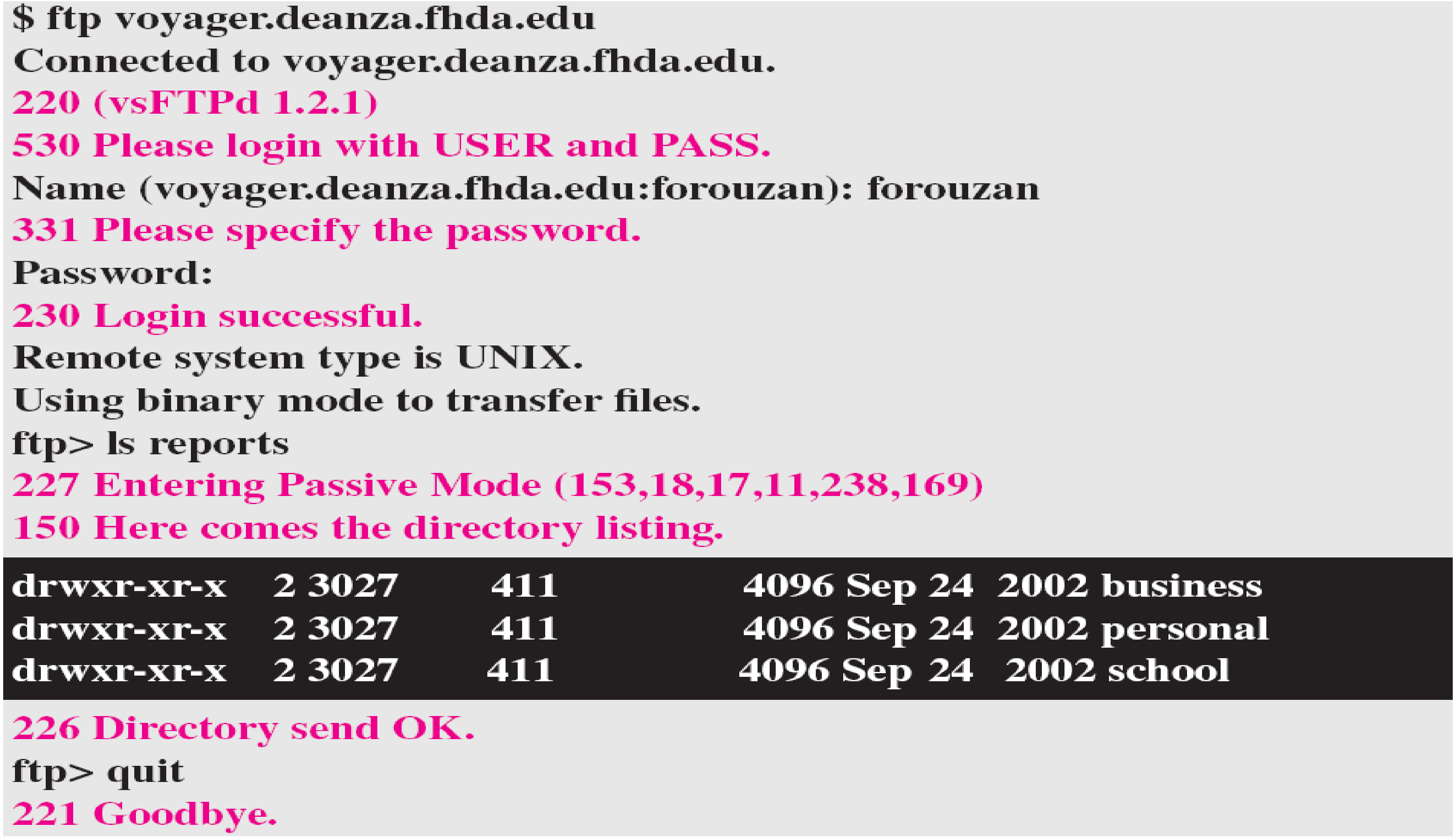
The following shows an actual FTP session for retrieving a list of items in a directory. The colored lines show the responses from the server control connection; the black lines show the commands sent by the client. The lines in white with a black background show data transfer.

1. *After the control connection is created, the FTP server sends the 220 response.*
2. *The client sends its name.*
3. *The server responds with 331.*

Example 26.4 (continued)

4. The client sends the password (not shown).
5. The server responds with 230 (user log-in is OK).
6. The client sends the list command (ls reports) to find the list of files on the directory named report.
7. Now the server responds with 150 and opens the data connection.
8. The server then sends the list of the files or directories on the data connection.
9. The client sends a QUIT command.
10. The server responds with 221.

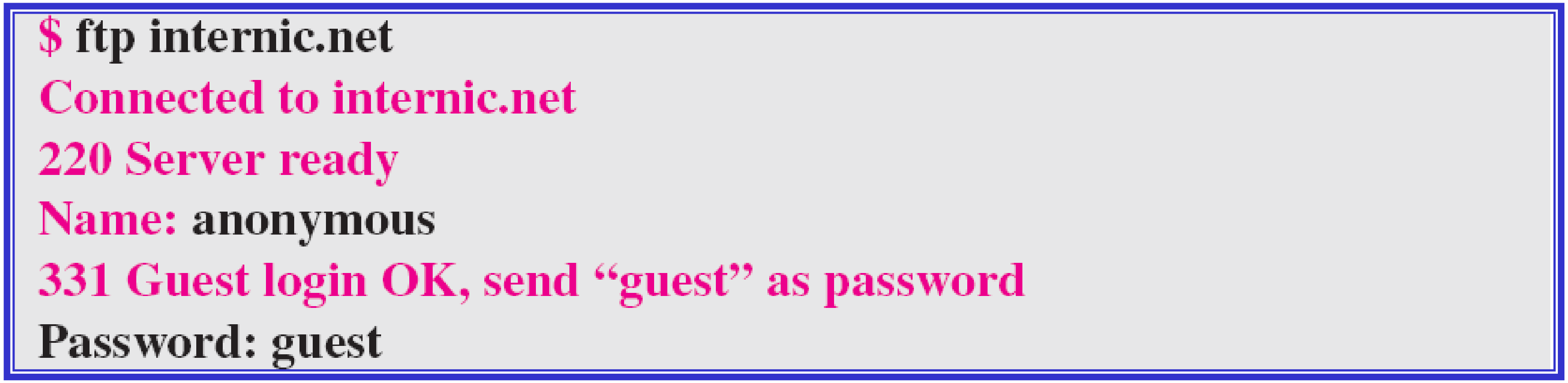
Example 26.4 (continued)



```
$ ftp voyager.deanza.fhda.edu
Connected to voyager.deanza.fhda.edu.
220 (vsFTPd 1.2.1)
530 Please login with USER and PASS.
Name (voyager.deanza.fhda.edu:forouzan): forouzan
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls reports
227 Entering Passive Mode (153,18,17,11,238,169)
150 Here comes the directory listing.
drwxr-xr-x    2 3027      411              4096 Sep 24  2002 business
drwxr-xr-x    2 3027      411              4096 Sep 24  2002 personal
drwxr-xr-x    2 3027      411              4096 Sep 24  2002 school
226 Directory send OK.
ftp> quit
221 Goodbye.
```

Example 26.5

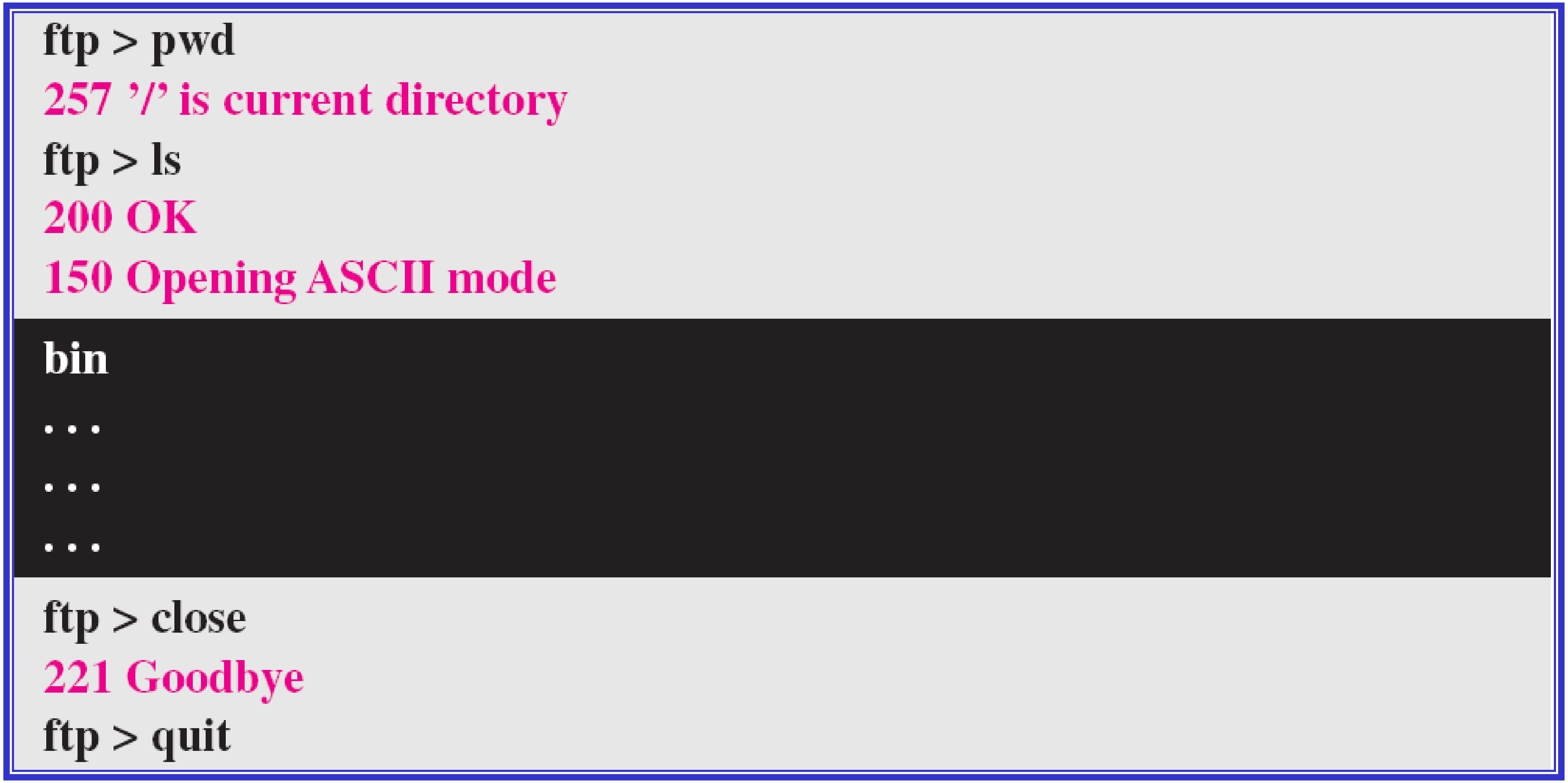
We show an example of anonymous FTP. We assume that some public data are available at internic.net.



```
$ ftp internic.net
Connected to internic.net
220 Server ready
Name: anonymous
331 Guest login OK, send "guest" as password
Password: guest
```

continued on next slide

Example 26.5 (continued)



```
ftp > pwd
257 '/' is current directory
ftp > ls
200 OK
150 Opening ASCII mode
bin
. . .
. . .
. . .
ftp > close
221 Goodbye
ftp > quit
```