

ADA 042740

ISI/TM-77-4
April 1977
ARPA Order
No. 2223

ARPANET
TENEX

Primer

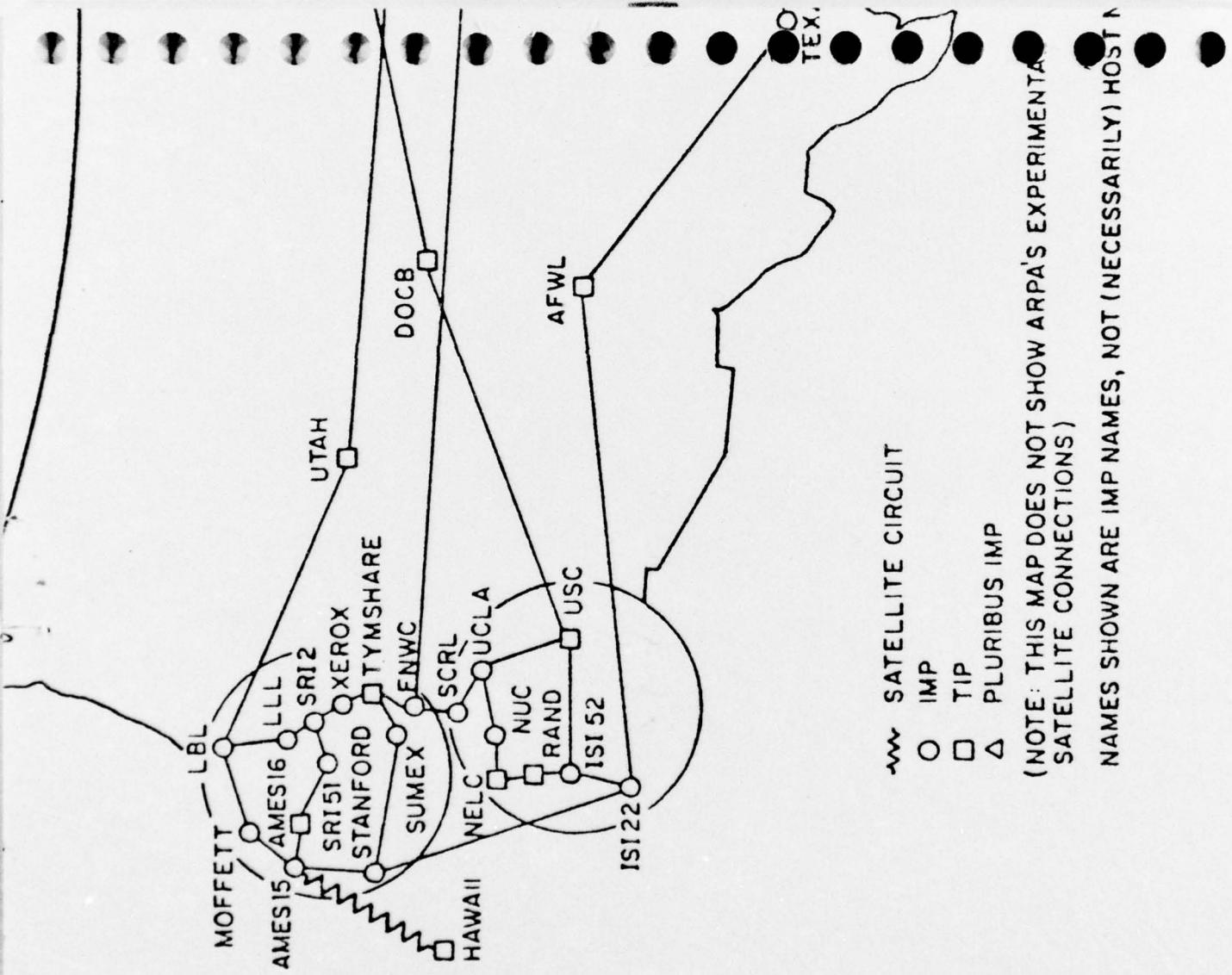
B and
MSG
Handling
Program

Chloe
Holg

DISTRIBUTION STATEMENT A
Approved for public release;
Distribution Unlimited

USC INFORMATION SCIENCES INSTITUTE
4676 Admiralty Way Marina del Rey California 90291

FILE COPY



UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
14. REPORT NUMBER ISI/TM-77-4 ✓	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) The ARPANET/TENEX Primer and the MSG Message Handling Program.	5. TYPE OF REPORT & PERIOD COVERED Technical Manual	
6. AUTHOR(s) Chloe Holg	7. CONTRACT OR GRANT NUMBER DAHCJ5-72-C-0308	
8. PERFORMING ORGANIZATION NAME AND ADDRESS USC/Information Sciences Institute 4676 Admiralty Way Marina del Rey, CA 90291	9. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS ARPA Order No. 2223 Program Code 3D30 & 3P10	
10. CONTROLLING OFFICE NAME AND ADDRESS Defense Advanced Research Projects Agency 1400 Wilson Blvd. Arlington, VA. 22209	11. REPORT DATE April 1977	
12. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) --- 13. NUMBER OF PAGES 3pp.	14. SECURITY CLASS. (of this report) Unclassified	
15a. DECLASSIFICATION/DOWNGRADING SCHEDULE		
16. DISTRIBUTION STATEMENT (of this Report) This document is approved for public release and file, distribution is unlimited		

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

D D C

18. SUPPLEMENTARY NOTES

AUG 9 1977

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

ARPANET, TENEX, MSG, message handling.

C

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This is a basic primer-like manual discussing the operation of the ARPANET, the TENEX operating system, and the MSG message handling program for novice computer users.

DD FORM 1 JAN 73 1473

EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-014-6601

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

407 952

LB

THE ARPANET/TENEX PRIMER
and
THE MSG MESSAGE HANDLING PROGRAM

Chloe Holg
USC/Information Sciences Institute

April 1977

ACCESSION for	
NTIS	White
DDC	Buff
UNANNOUNCED	
JUSTIFICATION	
BY	
DISTRIBUTION/AVAILABILITY	
D.	U
A	

This research is supported by the Advanced Research Projects Agency under Contract No. DAHC15 72 C 0308,
ARPA Order No. 2223, Program Code No. 3D30 and 3P10.

Views and conclusions contained in this manual are the author's.

This document approved for public release and sale; distribution is unlimited.

CONTENTS

FOREWORD	vi
I. INTRODUCTION	1
2. ARPANET: The System and the Language	3
2.1 Terminal	3
2.2 Host Computer	3
2.3 Operating System...TENEX EXECutive	4
2.4 Address on the ARPANET	5
2.5 Password	5
2.6 Account	6
3. TENEX: Accessing the System	7
3.1 The ESCAPE (or ALTMODE) Key	12
3.2 Control Characters	13
3.2.1 \uparrow C (CONTROL C)	14
3.2.2 \uparrow A (CONTROL A)	14
3.2.3 \uparrow R (CONTROL R)	15
3.3 Changing your Password	19
3.4 Logging Out of the System	20



4.	MSG Program: The TYPE Command	22
4.1	How To Read a Message	23
5.	MSG: ANSWER, FORWARD and SEND Commands	31
5.1	How to Answer a Message	31
5.2	How to Send a Message	35
5.3	How to Forward a Message	39
6.	MSG: DELETE/UNDELETE, MOVE, READ and EXIT Commands	41
6.1	The Delete Command	41
6.2	The Undelete Command	42
6.3	The Move Command	43
6.4	The Read Command	44
6.5	The Quit Command	46
6.6	The Exit Command	47

FOREWORD

The use of the ARPANET computer network and its resources by people primarily interested in informal message handling and text processing is impressive. People beginning to use the ARPANET, however, often must operate alone without the help of other users. While many levels of instruction manuals are available, it is apparent that a more basic, primer-like manual addressing the needs of the novice should be made available. Therefore, the ARPANET/TENEX Primer was initiated by Ms. Marie Wyrwa of the Defense Communications Agency, Office of the Chief Scientist. Ms. Wyrwa drafted a document for personal and other DCA use. The original document was rewritten for more general distribution by Ms. Chloe Holg of ISI and is available as an aid to the novice user.

The document is one of a series of user manuals being made available by IPTO through ISI. Great appreciation is extended to Marie Wyrwa.

1. INTRODUCTION

Section 1 Page 1

1. INTRODUCTION

This primer has been designed for those who have never operated a computer terminal, have no knowledge of how a computer works, and are unfamiliar with computer terminology.

A terminal is a device connected to a computer either near or far away by a Terminal Interface Message Processor (TIP). Using the TIP is the way you connect your terminal to a computer. Once your terminal is connected to a computer you are able to perform routine tasks electronically, such as handling messages. To use a terminal to receive or send messages requires an address on the ARPANET (the DEFENSE ADVANCED RESEARCH PROJECTS AGENCY computers that are linked together to form a network). This primer will tell you how to use your address on the ARPANET and how to read, send, file, and delete network mail using a program called MSG.

The ARPANET/TENEX Primer contains six sections:



Section 1 INTRODUCTION

Section 2 ARPANET: The System and the Language

Section 3 TENEX: Accessing the System

Section 4 MSG: The TYPE Command

Section 5 MSG: ANSWER, SEND, and FORWARD Commands

Section 6 MSG: DELETE, UNDELETE, MOVE, READ, QUIT, and EXIT Commands

Section 2 will acquaint you with basic terminology. Section 3 will tell you how to make a terminal/computer connection, how to log in, how to change your password, and how to log out. Section 4 will introduce you to the MSG program and show you how to read your messages. Section 5 will show you how to answer, send, or forward a message; Section 6 will show you how to delete and undelete a message, move a message, read a file, and quit or exit from the program.

CLASSIFIED MATERIAL CANNOT BE SENT OVER THE NETWORK.

2. ARPANET: The System and the Language

Since you are going to learn to use the system you might as well start learning some of the language.

2.1 Terminal

A *terminal* is a device intended for a person to interact directly with a computer. It usually consists of a typewriterlike keyboard and either a typewriterlike printing mechanism or another kind of display device. The computer your *terminal* is connected to (via your TIP) is called a *host*.

2.2 Host Computer

Since there are many *hosts* on the ARPANET (just as there are many post offices), each must have a identifying number (like a ZIP code) so you can connect your terminal to your *host*. When you are given an address on the ARPANET you will be told your *host*

name/number. When you are making a computer connection before logging in on your terminal you identify your host by the number. When you are sending mail to a person you address it to the host name. Each host has an operating system.

2.3 Operating System...TENEX EXECutive

An operating system is just that...a system to supervise a variety of people using a variety of subsystems and programs. The operating system in TENEX is called the EXECutive (or EXEC). TENEX EXEC has its own commands; you are at the command level in TENEX EXEC when the system types an at sign (@) at the beginning of the line. Wherever possible, an operating system or program uses symbols to tell you where you are in the system or program. These symbols are called 'heralds'; in TENEX EXEC @ is the herald telling you that you are at command level and the system is waiting for you to give it a command. When you type a question mark (?) at command level the system will print the commands available to you at TENEX EXEC command level.

Since TENEX EXEC is an operating system, it has within it many subsystems or programs. Each of the subsystems and programs has its own command level and its own commands, as you will see in Section 4 where the MSG program is presented. Since you will be a user of the TENEX operating system and the programs it supports, you need to have the system know who you are - so you are given an address on the ARPANET.

2.4 Address on the ARPANET

Your address on the ARPANET is the name of your directory. The name of your directory is generally your last name or the name of your group. Your directory not only identifies you as a user of the ARPANET but also provides space for you to receive and store mail in the computer, like a drawer in a conventional filing cabinet. Your directory has within it a permanent file named MESSAGE.TXT. Your network mail is automatically stored in this file. You need a key to open your directory; this is called a password.

2.5 Password

When your directory is established you are given a password. If you have not had the opportunity to choose your password when your directory was established, you may change it to any word of not more than 39 characters. Your password is known only to you and those you choose to tell. When you type your password on the terminal it does not print. This is for your protection so it's a good idea to have a short, uncomplicated password. Also a good idea is to make sure to remember exactly what your password is because if you forget it you can't get into your directory without having the people who operate the system do some rather complicated operations. Since you now have a directory and a password, the system needs to know how to keep track of you and gives you an account.



2.6 Account

The group to which you belong needs to be identified on the system; this is done by assigning you and everyone in your group an *account name or number*. You will be told your *account name or number* when you are given a *directory* on the ARPANET. It is necessary to tell the system your *account name or number* or you cannot log in.

You now have a *terminal* (a device) to connect to a *host* (your post office) to use the *operating system* (TENEX EXEC). By using your *directory* (mailbox), your *password* (key) and your *account (group)* you are able to run the MSG program to send and receive network mail.

By following the instructions in this primer you can make it all work for you easily and quickly. You cannot break the system, so relax - welcome to the ARPANET!

3. TENEX: Accessing the System

To connect to the system using a portable hardcopy terminal (i.e., a terminal that prints on paper) you need a telephone to dial the TIP telephone number you were given when your directory was established. Should you have need for a TIP telephone number other than your own, checking the ARPANET Directory Host Names Section (pp. 147-149) will identify the liaison person for the TIP you are interested in accessing. You can either send a network message to that person or check the Network Individuals Section (pp. 7-95) for the liaison person's telephone number or street address.¹

Before attempting to use a terminal the user should review the manual for that particular terminal, as there are many different types of terminal, each with certain characteristics. The procedures we are describing in this primer are for a portable hardcopy terminal.

¹ The ARPANET Directory referred to is the July 1976 issue, NIC No. 36437 and may be obtained from ARPA Network Information Center, Stanford Research Institute, Menlo Park, CA 94025.



For the portable hardcopy terminal check to make sure that:

1. The terminal OFF/ON switch is ON.
2. Speed is set for 30 characters per second operation.
3. Duplex switch is set at FULL DUPLEX.
4. Parity switch is set for EVEN.
5. Interface switch is set for INTERNAL.
6. The ON LINE key is depressed.

You are now ready to connect your terminal to the TIP. Dial the appropriate TIP number on your telephone. You will have completed the call when you hear a high-pitched tone; at this point attach the telephone handset to the acoustic coupler of your terminal in accordance with the directions for your particular type of terminal. Your terminal will give a "ready" indication with a light or the equivalent. When the light comes on type 'e' and a carriage return for synchronization, then type '@r' and a carriage return to allow your TIP to identify itself. You are now ready to make a connection to your host computer and log in to TENEX.

Before you begin the connection/login sequence it is necessary that you know about recovery procedures and the function of typing carriage returns and spaces. If you make an error in

typing before actually logging in to the system, simply type a carriage return. Typing a carriage return tells the system to execute the command you have just given it. The system will not recognize a mistyped command and will do nothing. This gives you the opportunity to retype the command correctly. In the following instructions and examples 'carriage return' appears as (cr). The general rule is that when the system asks you to confirm your command, typing (cr) essentially means "yes, go ahead and do it".

Typing a space after a command tells the system you have more information to give it. The word 'space' appears in the instructions and examples as (sp). The following example demonstrates (sp) and (cr).

Throughout this document, information that the user types or inputs to the system is underlined.

Example: You have checked out your terminal, dialed up your TIP and made a connection. The ready light illuminates and you are ready to go.

e (cr)
er (cr)

Typing e and a carriage return effected synchronization; typing er asked your TIP to identify itself. If you dialed up the DCEC TIP it would look like this:

DCEC TIP 400 *:43

Now you are ready to connect to your host; you type

eo (sp) host number (cr)

When you typed **o** you told the system to 'open a connection'. When you typed a (sp) you told the system you had more information to give it. When you typed the host number (e.g., 86 for ISI-SYSTEM-A) you told the system where to open the connection. When you typed a (cr) you told the system you were finished with the command and the system should go off and do it. At this point the system lets you know it heard your commands and acknowledges by telling you

Trying...

When the system has successfully made the connection it will tell you so and your terminal will print

Open
ISI-TENEX 1.34.8, ISI-SYSTEM-A EXEC 1.54.8
@

The system has opened a connection and the host acknowledges your connection by printing its identification line and then prints **@**. The **@** sign is the TENEX EXEC herald that tells you that the system is at command level and waiting for you to tell it to do something. At this point you must log in to tell the system who you are. You cannot run a program before you log in to the system. When you log in, the system will assign you a job number, tell you your terminal number, the date and time of your login. It will also tell you the date and time of your previous login and give you any information concerning the system (these are called 'system messages'). If your directory has received a message since you last used the system, you will be told you that you have new mail.

If the 'Open' message from the TIP does not appear, it is an indication of trouble on the network; one of the following messages will be typed on your terminal:

NET TROUBLE – means remote site cannot be reached.

REFUSED – means site up but refusing your connection.

HOST SCHEDULED DOWN UNTIL (DAY) (TIME) – means host will be available on the day and time indicated in the message.

HOST UNSCHEDULED DOWN UNTIL (DAY) (TIME) – means host is down and will be available on the day and time indicated in the message.

HOST NOT RESPONDING – means remote site is not up and it is unknown when service will resume.

ICP INTERFERED WITH – means that the TIP has refused to open a connection.

If any of the above messages appear instead of 'Open' you should attempt to make a terminal/computer connection again, after a reasonable lapse of time or when the TIP tells you the system is available by indicating the day and time.

Once you have made a connection to your host and receive the host identification line you are ready to log in. Before we demonstrate the login sequence, however, it is necessary to acquaint you with some of the system aids available to help a user recover from typing errors (control characters), and to find out what information the system wants from the user (the ESCAPE or ALTMODE key).



3.1 The ESCAPE (or ALTMODE) Key

After you have made a connection from your terminal to the host computer and begin the login procedure you can use the key labeled ESC (on some terminals it is labeled ALT) which is generally located at the upper left side of the keyboard. The ESC key does two important things for you in the TENEX system.

1. It will prompt you for input, i.e., if you typed LOGIN and then typed the ESC key, the system would ask you for your user name; when you typed in your user name and typed the ESC key again it would ask you for your password. After you typed in your password and ESC the system would type your account name or number. If after typing your password you typed a (cr), the system would ask for your account name or number. If you now typed ESC the system would print your account name or number. Typing (cr) would complete your login sequence and you would be ready to give another command to the TENEX system.
2. The second important use of the ESC key is to instruct the TENEX EXEC to recognize what has been typed and to type out the rest. Thus the user may type part of a command (or directory name) and ask the system to supply the rest of the command (or directory name) by typing the ESC key. The ESC key will not perform recognition on your password. The ESC does not print on your terminal. The symbol for ESC is always shown in examples or instructions as dollar sign (\$). When you see \$ it means that you should type the ESC key. If the partial type-in is not yet recognizable when you type ESC, then the system will sound a 'bell' (the terminal will

ding, beep, or buzz, — depending on the type of terminal). The user should type a few more characters, then ESC, and the system will recognize and complete your command. If you typed 'LOG\$' the system would complete the command by completing the word: 'LOGIN'.

Remember, typing the ESC (or ALT) key is a prompt or recognition request that the user makes to the system.

3.2 Control Characters

When a command is typed, the system patiently accepts each character typed without doing anything until you type a carriage return. In the system are "control characters" that signal the system to do something immediately. Control characters are not always displayed, but when they are they are shown preceded by an up arrow (^). They are typed by holding down the control key (generally located on the left side of the keyboard and marked CNTL or CTRL) and then typing the character. Thus, Control is like a shift key except that there is no "control lock" key.

The most important control characters are the following:



0 0

3.2.1 tC (CONTROL C)

tC tells the system to "stop what you're doing and get me back to the command level." (You are at command level when the system prints @.) This is useful to start over retyping a command or to interrupt a running program. If you type a tC by mistake and did not really want to interrupt the use of MSG (your running program) you may type

@CONTINUE

Your MSG program will be restored and you may continue your task.

3.2.2 tA (CONTROL A)

If a typing error is made at any time, typing tA will cause the system to forget the last character typed. Usually a backslash (/) will be typed followed by the bad character. Example: if the user types A followed by B followed by tA (CONTROL A) followed by C. the system will type 'AB/BC' and it will be just as if 'AC' had been typed. (You will get used to it.)

3.2.3 ↑R (CONTROL R)

↑R tells the system to retype the current line. After typing several ↑A, the line may become unreadable. ↑R will retype the line cleaned up and you may finish typing your input.

Now, remember, if you make a typing mistake while connecting your terminal to the host computer just type a carriage return and start over. You cannot use the TENEX operating system or programs until you have logged in to the system.

If you make a typing mistake after you have made a connection type ↑A' and then type the correct character. If you make a typing mistake and don't realize that you have made it the system will not recognize what you are telling it and ask you to repeat your command by typing @ ?. All you have to do is repeat your command correctly typed.

If you make a typing error while entering your password you may use ↑A but since your password does not print on your terminal the character you are correcting will not appear. If you suspect you may have erred in entering your password, simply type (cr) and repeat the login sequence.

Remember to use the ESC (or ALT) key if you forget what the system wants you to type or if you want the system to complete the command you partially typed. Remember to use the ↑C if you want to stop what you are doing and return to TENEX EXEC.

So far you have been shown how to connect your terminal to an



ARPANET host via the TIP. The following example recaps this information and demonstrates a login, the system output, notification of new mail; it then stops at the command level in TENEX Exec. Remember that the input the user types is underlined in the example.

Example:

e (cr)
er (cr)

DCEC TIP 400 *: 43

en 86 (cr)

Trying...

Open

ISI-TENEX 1.34.7, ISI-SYSTEM-A EXEC 1.54.3

@LOGIN SMITH 123456 (cr)

JOB 40 ON TTY11 21-SEP-76 04:07

PREVIOUS LOGIN: 20-SEP-76 10:50

TENEX WILL GO DOWN THU 9-23-76 TIL FRI 9-24-76

0500 FOR PREVENTIVE MAINTENANCE

[YOU HAVE NEW MAIL]

e

The @ in the last line of the example above tells you that you are at command level in TENEX EXEC. You are now ready to give commands to TENEX EXEC or ask it to load a program for you.

Since you are logged in to the host via a TIP the @ character cannot be typed as a single character. Typing @ and a carriage return after you are logged in via a TIP will detach your running job. You will sometimes need to type @ even though you are using the TIP. @ means 'at' in a network address, eg., SMITH@ISI. Therefore, you need a way to tell the TIP you do not want to detach your job and you do this by typing @ twice, eg., @@. The system will show @@ as @@@ to tell you it recognized your intent. Should you type one instance of @ and your job is detached you will have to reconnect to your host and attach your job.

Example: You have accidentally detached your job by typing @ as a single character. Since you already have a TIP connection you must connect to your host and attach your job. Instead of logging in to the system (which would create another job) you type the 'WHERE' command and your user name at command level in TENEX EXEC. User input is underlined.

@o 86 (cr)

Trying...

Open

ISI-TENEX 1.34.7, ISI-SYSTEM-A EXEC 1.54.3

@where smith (cr)

DETACHED, JOB 40 EXEC

@

The system tells you that you are detached, your job number and what system you were using when you were detached. You ask the system to attach your job. You type *ATTACH*, your user name, your password, your job number and (cr)'.

@attach smith 40 (cr)

@

Your job is attached and the system is waiting for you to give it a command.

You are now ready to give commands to TENEX EXEC or load a program. For the sake of example we will assume that you do not wish to load a program but rather ask TENEX EXEC to change your password, and then log out.

3.3 Changing your Password

Select a new password which may contain up to 39 characters, although it is recommended that you keep your password short and uncomplicated. Now what you are going to do is tell TENEX EXEC to forget about your old password and to remember the new one. To do this you must be at command level in TENEX EXEC (indicated by @ at the beginning of the line) and use the CHANGE command. You type CHANGE PASSWORD a space and your directory name (dirname), a space and your old password (oldpword) a space and your new password (newpword), another space and type your new password (newpword) again and a carriage return. Your old or new password will not print on the terminal. This is what the CHANGE PASSWORD procedure looks like:

@CHANGE PASSWORD (dirname) (oldpword) (newpword) (newpword) (cr)

and what actually appears on your terminal is

@CHANGE PASSWORD SMITH

because your password NEVER prints on the terminal. Typing your new password a second time is a confirming function and if you mistype your new password the second time the system will not record the change and tell you 'PASSWORD COPIES DO NOT AGREE' and return you to command level where you repeat the procedure, typing your new password correctly both times.



3.4 Logging Out of the System

If you open a connection to the TIP and log in to a host, it is necessary that you log out and close the TIP connection. Turning off the terminal will not do it; you must log out and close the TIP connection. Since you are at command level in TENEX EXEC you give the logout command and type a carriage return.

@logout (cr)

If you typed log\$ the system would tell you that you are already logged in so if you wish to use ESC to complete your command to log out you must type LOGO\$ (cr) and the system will log you out and give you system downtime information, tell you it killed your job, the date and time of your logout and how much computer time you used in real time — something like the following:

TENEX WILL GO DOWN THU 11-11-76 2200 TIL FRI 11-12-76 0500 FOR
PREVENTIVE MAINTENANCE

KILLED JOB 21, USER SMITH, ACCT 12345, TTY 11, AT 11/10/76 0408.
USED 0:0:4 IN 0:1:7

Now you are logged out but you still have a TIP connection so you type

ec (cr)

@c tells the TIP you wish to close your connection and the (cr) says 'do it'. The TIP will give you information on the latest network news and how to access it and then tell you your connection is closed. It would look something like this:

@c (cr)
Latest net news 10 NOV 76
Use "@N<cr>" followed by "netnews<cr>"
Closed

If you wanted to see the latest network news (which probably wouldn't be of great interest to you at this point in your experience on the ARPANET) you would type **@N (cr)** netnews (cr) after the TIP closed the connection. An **@c** command will stop output of the network news immediately.

Turn the terminal OFF/ON switch to OFF and remove the telephone handset from the acoustic coupler on the terminal.

You have so far learned the basic terminology, how to log in, change your password and how to log out. The next section will tell you how to read your mail using the MSG program.

4. MSG Program: The TYPE Command

The MSG program, one of the programs supported by TENEX EXEC, was created at USC/Information Sciences Institute to provide a reliable and robust system for handling network mail.

You were told in Section 2 how to access the TENEX EXEC. You were also told that TENEX EXEC is a system with a command level and how to find out what commands were available to the user. In the definition of your DIRECTORY (Section 1, page 3) you learned that you had a permanent file that received and stored your network mail, a file named MESSAGE.TXT. When you use the MSG program it automatically loads your MESSAGE.TXT file for you and the MSG commands enable you to read, send, move, and delete network messages.

Asking TENEX to load the MSG program and typing a ? at the command level will tell you ALL of the commands that are available. You will note that MSG has a command level herald different from that of TENEX EXEC. The command level herald in TENEX EXEC is @. The command level herald in MSG is <. Since we are concerned with only some of those commands - *TYPE, ANSWER, SEND, FORWARD, DELETE, UNDELETE, MOVE, READ, QUIT, and EXIT* - we will focus on these commands only. In the instruction and examples user input is underlined.

4.1 How To Read a Message

To load the MSG program (or any program supported by TENEX EXEC) simply type the program name and (cr) at the command level in TENEX EXEC.

Example:

eMSG (cr)

TENEX EXEC will get the program for you, let you know that you are now talking to the MSG program, then wait for you to handle your mail. When you finish with the MSG program you are returned to command level in TENEX EXEC. If at any time you type a **tC while using the MSG program, TENEX EXEC will immediately type **@** and wait for you to give it a command. If you type **tC** accidentally and really want to continue using the MSG program, then type **CONTINUE** at TENEX EXEC command level and the MSG program will be restored and you may continue to use it.**

Example: You wish to make a TIP connection, log in to your directory on your ARPANET host to read your mail. User input is underlined.



e (cr)
er (cr)

DCEC TIP 100 0:43

eq 86 (cr)

Trying...

Open

ISI-TENEX 1.34.7, ISI-SYSTEM-A EXEC 1.54.3

eLOGIN SMITH 123456 (cr)

JOB 40 ON TTY11 21-SEP-76 04:07

PREVIOUS LOGIN: 20-SEP-76 10:50

TENEX WILL GO DOWN THU 9-23-76 TIL FRI 9-24-76

0500 FOR PREVENTIVE MAINTENANCE

[YOU HAVE NEW MAIL]

e

The system has told you that you have new mail so you tell TENEX EXEC to load the MSG program for you by typing MSG (cr). As soon as MSG is loaded, your terminal will print out the MSG identification line (program name and the version date of the program), help information, the number, date, sender and subject of your new message. It will also print out the date/time you last read your messages, how many old messages and disk pages you have. It will then print the MSG herald (<) and you are ready to read your new mail.

Example:

eMSG (cr)

MSG -- VERSION of 1 APRIL 1976
TYPE ? FOR HELP,? FOR NEWS

-+ 4 25 SEP JONES at USC-ISI Agenda Items
Last read: 23-SEP-76 04:07:36; 3 old msgs, 2 disk pages.
<-

You want to read your message so you need to know the command to tell MSG to type your message on your terminal. The commands in MSC are single characters and are self-completing. You merely type a single character and MSG completes typing the command and waits for you to type more information. You want to read a message and the command to display the message on your terminal is *TYPE*. The command *TYPE* can be used in many ways; the following describes how the *TYPE* command works. In the instruction and examples the input from the user is shown in lower case, command completion typed by the MSG program in UPPER CASE.

To read a message type t, the message number and (cr).

<-tYPE 4 (cr)

MSG will type message 4 on your terminal.



To read more than one message type, the message numbers in any sequence separated by commas (no spaces), for example, 2,4 (cr).

<-tYPE 2,4 (cr)

MSG will type messages 2 and 4 on your terminal.
You could also type t 4,2 (cr).

<- tYPE 4,2 (cr)

MSG will type messages 4 and 2 on your terminal.

If you want to read messages 1 through 4, type t 1-4 (cr).

<-tYPE 1-4 (cr)

MSG will type messages 1,2,3, and 4 on your terminal.

If you had 10 messages and wanted to read 2,5,6,7,8,9 and 10 you would type t 2,5-10 (cr).

<- tYPE 2,5-10 (cr)

MSG would type messages 2,5,6,7,8,9, and 10 on your terminal.

If you want to read all of your messages type t a (no cr).

<-tYPE aLL MESSAGES

MSG would type all of your messages on your terminal.

The MSG program will tell you if you have messages that are 'not examined'. You may read these messages by typing t n (cr).

<-tYPE nOT EXAMINED (cr)

MSG will type all of your 'not examined' messages on your terminal.

If you do not want to read your messages but just see the 'headers' (headers are the information line telling you message number, date, sender and subject) you type h a (cr).

<-hEADERS aLL (cr)

MSG will type all of the headers of all of your messages on your terminal.



If you wish to check to see if you have received a new message while you are using MSG, type t r (cr).

<-tYPE rECENT MESSAGES (cr)

If you have received a new message while using MSG the header will be typed on your terminal. If you have not received a new message, MSG will return you to the command level and wait for you to type a command.

If you wish to examine old messages only, type t o (cr).

<-tYPE oLD MESSAGES (cr)

MSG will type all of your old messages on your terminal.

If you wish to read your messages in inverse order, type t i (cr).

<-tYPE iNVERSE (cr)

MSG will type your messages beginning with the most recent message and ending with the oldest message.

Suppose you have given MSG a command to type all of your messages and then want to stop the

4. MSG Program: The TYPE Command

Section 4.1 Page 29

typing and return to command level. MSG, like TENEX EXEC, has a control character to accomplish this. The control character in MSG is CONTROL N (tN) which is typed by holding down the control key located on the upper left side of the terminal keyboard and typing N. tN tells MSG to "stop what you are doing and get me back to command level".

Example: You wish to stop output from MSG so you type tN at any place during the output procedure.

tN

MSG tells you it is aborting the ongoing procedure and returns you to command level.

*** ABORTING TYPEOUT ***

<-

NOW, WE HAVE TALKED ABOUT THE TYPE COMMAND AND THE VARIOUS WAYS TO USE IT BUT STILL HAVE NOT READ THE NEW MESSAGE THE SYSTEM MENTIONED DURING LOGIN. WE SHALL DO SO NOW - REMEMBER, THE MESSAGE NUMBER IS 4, SO TYPE t 4 (cr) . IN THE EXAMPLE USER INPUT IS UNDERLINED; THE COMMAND THE USER TYPES IS SHOWN IN LOWER CASE, THE COMMAND COMPLETION BY MSG IN UPPER CASE.



Example:

```
<- TYPE 4 (cr)
(msg. #4, 434 chars)
Mail from USC-ISI rcvd at 23-SEP-76 0911-PST
Date: 23 SEP 1976 0913-PST
Sender: JONES at USC-ISI
Subject: Agenda Item
From: JONES at USC-ISI
TO: SMITH at ISI
Message-ID: USC-ISI 23-SEP-76 09:13:11-PST
```

I think we should include a session on the MSG program.

Sam

<-

You may, perhaps, wish to respond to the message you just received and the next section will show you how to 1) ANSWER a message, 2) SEND a message, and 3) FORWARD a message you received to another user.

5. MSG: ANSWER, FORWARD and SEND Commands

Section 3 described the MSG program, and showed you how to read a message. We will continue with the message you read in Section 3 and show you how to *ANSWER* a message, *SEND* a message, and *FORWARD* a message you received to another user.

5.1 How to Answer a Message

Example: You read your message by typing t 4 (cr) and MSG prints it on your terminal. User input is underlined; user command input is shown in lower case, system command completion in UPPER CASE.



<- TYPE 4 (cr)

(msg. #4, 434 chars)

Mail from USC-ISI rcvd at 23-SEP-76 0911-PST

Date: 23 SEP 1976 0913-PST

Sender: JONES at USC-ISI

Subject: Agenda Item

From: JONES at USC-ISI

To: SMITH at ISI

Message-ID: USC-ISI 23-SEP-76 09:13:11-PST

I think we should include a session on the MSG program.

Sam

<-

You wish to answer the message. The command in MSG is ANSWER. You type a, the number of the message you wish to answer and (cr).

<- aNSWER 4 (cr)

4 23 SEP JONES at USC-ISI Agenda Item

MSG prints the header line and asks you to whom you wish to answer.

Reply to those whom the message is:

You don't understand 'Reply to those whom the message is:' so you type ? and MSG tells you it is reading the HELP file and then prints your options. You wish to direct your answer to the user who sent you the message so you type f (cr); MSG will ask you if you wish to include other users. If yes, type the addresses; if no, type (cr). MSG then tells you the control character (tZ) you type to tell MSG you have finished your message, and the control character (tN) to stop or abort your answer if you change your mind and don't want to answer the message.



Reply to those whom the message is: ?

[Reading HELP file...]

The <ANSWER SUB-COMMAND> can be any of the following:

- F -- From (indicating the sender of the message only)
- T -- To (indicating the sender of the message and those addresses in the To: list)
- C -- CC (indicating all recipients of the original message in addition to the sender in the message)

Reply to those whom the message is: FROM:

Additional addresses (? for help): (cr)

[Complete typing message body, 1Z to finish, 1N to abort]

I agree with you, Sam, and I think we should schedule the session on MSG immediately after the review of the TENEX operating system.

Clyde

1Z

QS,?,carriage-return: (cr)

JONES -- ok

<-

When you typed **tZ**, MSG asked you if you wanted to **Q** (queue your message for regular network traffic), or **S** (send your message immediately), **?** (ask for information), or carriage-return (yes, send the message). You can type **S** or (cr) to send the message immediately. You typed (cr) and MSG told you it sent your message to JONES ok and returned you to command level.

Now you have answered JONES's message. If you had decided to abort your answer and typed a **tN** MSG would have asked if you were sure you wanted to abort and you would type **y** (Yes).

ARE YOU SURE YOU WANT TO ABORT? [CONFIRM] **y**ES
SNDMSG ABORTED

<-

5.2 How to Send a Message

Now you want to **SEND** a message. You need to know the directory (or user) name of the recipient of your message and his host name. If you don't know the address, check the ARPANET



DIRE**C**TORY Network Individuals section (pp. 7-95). For the sake of example you want to send a message to BROWN at a host other than your own, say ISIC, so you type **BROWN@ISIC**. **REMEMBER THAT @ MUST BE TYPED TWICE AND THE SYSTEM WILL PRINT @@@.** SNDMSG is a prompt-driven system for sending messages, which means it will ask you for information automatically.

Example: You are logged in at your host and have loaded the MSG program, read your mail, answered a message, and now want to send a message. You type s (cr). User input is underlined, user command input is in lower case type; MSG command completion is in UPPER CASE type.

```
<- SNDMSG [Confirm] (cr) YES  
[ control-N aborts back to MSG ]
```

To (? for help): brown@~~c~~ISIC (cr)

You typed in an address. You may send the same message to multiple users by typing commas (no spaces) between the addresses. Type (cr) when you finish your input. MSG asks if you wish to send copies of the message to anyone. If yes, type the address or addresses separated by commas (no space) and (cr). You do not wish to send copies so you type (cr). MSG asks you for the subject of the message. You type the subject and a (cr) to tell MSG you are finished and MSG prompts you for the text of the message.

If you make a typing error in the To, Cc, Subject, or Message fields of SNDMSG, control characters are available to aid you in correcting your errors.

Typing a **↑A** says delete the last character you typed.

Typing a **↑W** says delete the last word you typed.

Typing a **↑Q** says delete the last line you typed.

Typing a **↑R** says review the line you have just typed.

Typing a **↑S** says review the entire message that you just typed.

Now you are ready to continue to send the message. We will begin again at the herald and go through the entire procedure.

<- \$NDMSG [Confirm] (cr) YES
[control-N aborts back to MSG]

to (? for help): brown@eisic (cr)

cc (? for help): cr

Subject: New User's Meeting Agenda (cr)

Message (? for help):

Those individuals in your group who wish to attend the New User's Meeting should send a message to SMITH@eISI. The agenda now includes a session on the MSG program.

Clyde

1Z

Q,S,P,carriage-return: (cr)

BROWN at ISIC -- ok

<-

Your message has been sent. As you can see, now that you are getting the hang of it, answering

5. MSG: ANSWER, FORWARD and SEND Commands

Section 5.3 Page 39

messages and sending messages are very similar in procedure. You may, at times, want to have a copy of the message you are sending so you just type your own address in the Cc: line and a copy of the message will be sent to you. MSG will include your name in the message so the recipient will know that you have a copy and will tell you SMITH -- ok after you tell MSG to send the message.

5.3 How to Forward a Message

You decide you wish to *FORWARD* a copy of message number 4 to user WHITE at ISIB. The command for forward is f, the number of the message and (cr). You may forward several messages at the same time by typing the numbers of the messages separated by commas (no spaces) and a (cr). MSG will ask you for the subject of the forwarded message and give you the option to type a message of your own which will appear in the message body. If you do type a message end it with TZ, if you do not wish to type a message also type a TZ. This tells MSG you wish to go on to the next field. MSG asks for an address to forward the message to and if you wish to send copies to anyone. It asks you if you want to queue or send the message and then confirms that the message has been forwarded. User input is underlined; user command input in lower case; MSG command completion output in UPPER CASE.

Example:

<- FORWARD 4 (cr)
4 23 SEP JONES at USC-ISI Agenda Item

Subject of forwarded msg(s): MSG Session (cr)

[Complete typing message body, 1Z to finish, 1N to abort]

1Z

To (? for help) white@isisib (cr)

Cc (? for help): (cr)

Q.S.?carriage-return: (cr)

WHITE -- ok

<-

6. MSG: DELETE/UNDELETE, MOVE, READ and EXIT Commands

You now have a very good idea of how MSG works. Rather than create example scenarios we will assume you are in the MSG program, have read, answered, sent and forwarded mail. Now you wish to **DELETE**, **UNDELETE**, and **MOVE** a message and then **exit** from MSG back to TENEX EXEC.

We will give a brief description of these commands and show you what they look like.

6.1 The Delete Command

You have read message number 4 and wish to delete it. You may delete all (type **d a (cr)**), or some (type **d n,n,n-n, (cr)** where n=the message number), or, as in the example, only one message so type **d 4 (cr)**.

Example:

<- **dELETE 4 (cr)**

<-



MSG has marked message number 4 for deletion. The header remains in your MESSAGE.TXT file but is marked with an asterisk (*) to tell you it is a deleted message. Should you wish to type message 4, MSG will tell you it is a deleted message.

Example:

```
<-TYPE 4 (cr)  
4 IS A DELETED MESSAGE
```

<-

6.2 The Undelete Command

You may, of course, *UNDELETE* a message, but only if you do not *EXIT* from MSG. Once you delete a message and exit from MSG there is no way to retrieve the deleted message. If you *QUIT* the MSG program it will tell you that you have undeleted messages but will not delete them until you exit the program. To undelete the message that you have deleted type **u 4 (cr)**.

Example:

<- **uNDELETE 4 (cr)**

<-

MSG restores the deleted message to its original state and removes the asterisk (*) at the message number in the header list. As in the delete command you may undelete all (type u a (cr)), or some (type u n,n,n-n, (cr) where n=message number), or one message as demonstrated in the example above.

6.3 The Move Command

Suppose you wish to save the message for future reference. You may do so by moving the message to another file (existing or yet to be created). Moving a message is essentially the same as deleting it because it is taken out of your MESSAGE.TXT file and put into another message file that you have created. MSG marks a moved message with an asterisk (*) and it will remain in the header list until you exit your session with MSG. Again, you may *MOVE* all (m a (cr)), or some (m n,n,n-n, (cr) where n=message number) or one message (m 4 (cr)).



Example: You wish to move message number 4 to save it for future reference. You type **m 4 (cr)** and MSG will ask you for the name of the file you want to store the message in.

```
<- mOVE 4 (cr)
INTO FILE NAME: meeting.msgs (cr)
```

<-

The file you created to store your moved message is a "message file format" file and you can read it in MSG at any time by using the **read** command.

6.4 The Read Command

The **READ** command reads into the MSG program any "message file format" file you have created by using the **move** command. A "message file format" file is structured in exactly the same manner as your **MESSAGE.TXT** file, i.e., having a message number, date, author and subject as the first line of each entry. If you tried to read a text file, for instance, into MSG the MSG program would tell you that something was wrong with the file and it could not be read. It is a good idea to use "msgs" in the name of your created message files, e.g., "meeting.msgs", or "old.msgs" or "procedure.msgs", etc.

Example:

```
<- READ FILE NAME: meeting.msgs (cr)
Last read: File never read, 0 old msgs, 2 msgs, 1 disk page.
```

<-

The **READ** command gives you status information on each new file you read into MSG. In this example you have never read the file before.

You are now able to **TYPE** the messages you have stored in your file named "meeting.msgs". After you have finished and wish to continue reading your regular messages you must use the **READ** command to reload your MESSAGE.TXT file.

Example:

<- READ FILE NAME: message.txt (cr)

Last read: 23-SEP-76 04:07:36; 4 old msgs, 2 disk pages.

5

Your MESSAGE.TXT file is restored, you are given a status message, and you may continue with your task.

6.5 The Quit Command

The *QUIT* command tells MSG to quit the program but not to change your file. If you deleted or moved messages they will continue to appear in your message header marked with a asterisk (*) to tell you they are deleted or moved. The *QUIT* command is fine to use if you have not changed your file by deleting or moving messages and you type q (cr).

Example:

```
<-qUIT [Confirm] (cr) Yes  
Good-bye.
```

@

You have been returned to TENEX EXEC and you may now log out and close your TIP connection.

6.6 The Exit Command

The **EXIT** command tells MSG that you are finished using the program and you wish to leave it. If you have not deleted or moved messages and you type e (cr), you will see the following

Example:

```
<- qXIT AND UPDATE OLD FILE <SMITH>MESSAGE.TXT;1 [Confirm] (cr) Yes  
FILE NOT CHANGED, SO NOT UPDATED.  
GOOD-BYE.
```

@

You are now back at TENEX EXEC and you may log out and close your TIP connection.

You have deleted or moved messages and if you type e (cr) you will see

Example:

```
<- qXIT AND UPDATE OLD FILE <SMITH>MESSAGE.TXT;1 [Confirm] (cr) Yes  
Good-bye.
```

@

6. MSG: DELETE/UNDELETE, MOVE, READ and EXIT Commands

Section 6.6 Page 49

You are now back at TENEX EXEC and may now log out and close your TIP connection.²

You now have the basic tools to enable you to use the ARPANET. Many other programs are available on the ARPANET TENEX machines - text editors, other message handling programs, programs to format text, a protocol for transferring files from one ARPANET site to another, and facilities for interhost communication. Documentation for other programs is available to the novice user in the form of TENEX manuals from ISI. A basic package of programs for new users is contained in the JOY of TENEX Manual; MORE JOY of TENEX offers yet more programs - more powerful text editors and message handling programs, a formatting program, a protocol for transferring files between ARPANET hosts, and a facility for interhost communication - for new users who have become familiar with the TENEX operating system and have mastered the basic message handling and text editor programs.

² Logout and closing TIP connection procedures are covered in Section 3.4, pp. 20-21.

