USER'S GUIDE TO
Computing

Electronic Mail at Wesleyan

28 April 1989

Wesleyan University

Computing Center, Middletown, CT   (203) 347-9411 ext. 2511

Table of Contents

Electronic Mail at Wesleyan

1. OVERVIEW

MINT (Mail INTerface) is the new electronic mail program used on VAX systems at Wesleyan. MINT was written at Wesleyan and is still under development.[1] A second mail program, MM remains in use on the administrative DEC-20, but will soon be replaced by MINT on VAX systems.

Using MINT, members of the Wesleyan community with VAX computing accounts can send and receive messages between users at Wesleyan and hundreds of sites around the world. This guide, intended for both new and experienced VAX users, is organized into five sections:

- A typical MINT session: This section will give you an idea of what your computing interaction will be when you use MINT to read and send messages. Browse through it quickly initially. You may wish to return to this section when you are ready to try the MINT program for the first time.
- Basic Concepts: This section explains several concepts basic to using MINT. The structure of the MINT program, and addressing and referencing messages are among the topics covered.
- Getting Started with MINT: This section explains the MINT commands needed by most users to use the MINT program. Topics include: getting help, listing, reading and sending messages.
- Expanding your capabilities: This section includes topics that will facilitate your use of MINT. For example, it discusses how to create a mailing list that includes abbreviations for addresses you use frequently. It will be most useful to people who use the mail system frequently, or who send/receive many messages.
- Command Dictionary: The dictionary includes a brief explanation of MINT commands, options, and concepts. It is intended primarily as a reference; however, if you browse through it you may discover just the option for which you've been looking.

[1] Please send suggestions and problem reports to Joseph Deck, Wesleyan Computing Center, or via MINT TO: JDECK@EAGLE.

2. A TYPICAL MINT SESSION

This example begins after WGIBBS has logged in and noticed the message:

```
%MINT-I-ONENEWMSG, You have one new mail message
Please use the MINT (Mail INTerface) program to read the message.
```

WGIBBS then begins MINT by typing MINT at the VAX/VMS prompt:

```
PHYS>MINT
```

The following start-up screen is then displayed:

```
RS      1)  4-DEC FRED@LBL.BITNET   ICPEAC: NYC Library         (251)
N       2)  4-DEC KBERKNER@EAGLE    ICPEAC: 2nd mailing         (394)









File: UD:[PHYS.WGIBBS]MINT.TXT;1

MINT>
```

The top part of the screen list two messages numbered 1) and 2). The letters "R" and "S" to the left of the one numbered 1) indicate that this message has been Read and has been flagged by WGIBBS as Special. The letter N beginning the next line, indicates that this is a New message that has been received since the last time WGIBBS used the MINT program.

The bottom part of the screen has a status line and several lines for MINT commands and messages.

To read the new message, WGIBBS types READ at the MINT> prompt which is at the bottom of the screen:

```
MINT>READ
```

and the message is displayed:

```
Date: 4-DEC-1988 17:27:51.75
From: KBERKNER@EAGLE.WESLEYAN.EDU
Subject: ICPEAC: 2nd mailing
To: WGIBBS@EAGLE.WESLEYAN.EDU

Dear Bill,
As you know the 2nd mailing is scheduled to go out on December 15th.
At the last minute we realized that we never reached agreement about
the late registration fee, and we're now taking a straw vote. Please
respond as soon as possible.               -- KLAUS



File: UD:[PHYS.WGIBBS]MINT.TXT;1  CM: 2      CR: 2
READ>
```

WGIBBS decides to send a reply, so types REPLY after the prompt READ> displayed at the bottom of the screen:

```
READ>REPLY
```

As WGIBBS begins to compose the reply to message 2) the screen looks as follows:

```
Date: 4-DEC-1988 17:27:51.75
From: KBERKNER@EAGLE.WESLEYAN.EDU
Subject: ICPEAC: 2nd mailing
To: WGIBBS@EAGLE.WESLEYAN.EDU

Dear Bill,
As you know the 2nd mailing is scheduled to go out on December 15th.
At the last minute we realized that we never reached agreement about
Text of the message to which you are replying
Dear Klaus,




PF1 exits to SEND>; PF2 gets help; PF8 changes windows; PF4 sends msg
```

WGIBBS finishes the message and then mails it by pressing the PF4 key. A second later a line at the bottom of the screen shows that the message has been sent:

```
SEND>
Message queued to KBERKNER@EAGLE.WESLEYAN.EDU
```

The prompt at the bottom of the screen then changes to READ> and WGIBBS types EXIT to exit from the read-level and return to the top-level of MINT:

```
READ>EXIT
```

Finally, to exit from MINT and return to the VAX/VMS prompt, WGIBBS again types EXIT:

```
MINT>EXIT
```

3. BASIC CONCEPTS

3.1. MINT Is Tri-Leveled

The MINT program is organized into three-levels: ***top, read, and send.*** These are used very much as though you were running three different programs. Each level has

- A primary purpose
- A set of commands
- A help facility
- A prompt to remind you which level you are currently using.

When you first begin the MINT program you will be at MINT's ***top-level*** and you will be prompted with the prompt MINT>. The primary purpose of this level to provide you with information about the messages you have received.

When you give a command that involves reading messages, you move to the ***read-level*** and are then prompted READ>. As you read messages, the read level offers numerous options, such as copying, printing, or forwarding the message you have just read.

When you wish to compose and send a message, you move to the ***send-level*** of MINT. At the SEND> command prompt, you will have options such as inserting text from a file or using a spelling checker.

For the most part, switching between the levels of MINT is automatic, but while you are learning the system, its helpful to stay aware of the command prompt. If you see the SEND> prompt, for example, you will recall that you have a message in progress waiting to be sent.

3.2. The MINT Command Language

There are three ways to give commands in MINT:

1. When the cursor is flashing just past a MINT command prompt at the bottom of the screen, either MINT> , READ> , or SEND> , you can type a command name at the prompt. For example, after beginning MINT, you will see the command prompt MINT> and can type DIRECTORY for a list of your messages:

```
MINT>DIRECTORY
```

   Commands can be typed in either upper or lower case and can be abbreviated to two or three characters.

2. Certain function keys can be used to issue commands while composing messages. For example, PF3 can be pressed to insert a file. On VT100+ series terminals the function keys used for MINT are labeled PF1 - PF4. On IBM compatible PC's (using VT100 emulation), these keys are labeled F1 - F4.

3. Control sequences can be used at all three levels of MINT to perform certain commands. For example CTRL/H will provide a list of command options available at the current level.

Command Recall

You can use the up-arrow and down-arrow keys in MINT to recall previously typed commands just as you can recall VAX/VMS commands outside of MINT. Up to your last twenty commands can be recalled in each of the three levels of MINT; i.e., a separate list of old commands is saved for each of the three levels. After recalling a command you can edit it just as you edit VAX/VMS commands.

Command recall saves typing when correcting errors or entering a series of similar commands or addresses.

Command Qualifiers

Command qualifiers offer options to the way commands are used. They have the format:

`command/qualifier`
for example:
`DIRECT/UNSEEN`
	OR	`command/qualifier=value`
		for example:
		`DIRECT/FROM=WGIBBS`

Enclose multiple words in quotes

In some cases you will want to specify more than one word as the value of a qualifier. In this case, enclose the text in quotes; for example:

```
DIRECT/TEXT="last date to apply"
```

Enclose lists in parentheses

You may specify more than one item as the value of a qualifier by enclosing the list in parentheses and separating items with commas; for example:

```
ADD/TO=(MBORN,NBOHR)
```

3.3. MINT.TXT: The file used to store your messages

The messages you receive are stored by the mail system in the *home directory* of your *home account*. (The system uses the logical name SYS$LOGIN for this directory.) This is your default directory when you first login. The filename is MINT.TXT;1. Note that 1) The generation number will remain at ;1, no matter how many messages you receive and 2) *This file should not be edited or printed since it contains some binary information.*

3.4. Referring to a Group of Messages

When each message is stored in your message file, MINT.TXT;1, it is assigned a message number (1,2,3, etc.), and you can use these numbers to refer to specific messages. In MINT terminology, a "group of messages" is called a ***range*** and a key word that specifies a range is called a ***range qualifier***. NEW, ANSWERED, LAST, READ, UNSEEN, SUBJECT, and TEXT are among the many range qualifiers that are used by MINT to classify your messages.

You can refer to a group of messages either by using message numbers or range qualifiers. The following examples show variations of the DIRECTORY command that could be used to list different groups of messages:

```
DIR 1:5
```

This command will list the first 5 messages. The beginning and ending numbers of the range are separated by a colon.

```
DIR 1,2,5,16:20
```

The list of messages can also include several numbers separated by commas. The above command will list messages 1, 2, 5, and 16 through 20.

```
DIR/NEW
```

In this example the range qualifier /NEW is used to specify all NEW messages.

```
DIR/LAST=25
```

In this case, the last twenty-five messages received will be listed.

```
DIR/SUBJECT="APS meeting"
```

The above command refers to the group of messages received with a SUBJECT containing "APS meeting" (any combination of upper and lower case letters).

```
DIR/FROM=LBOLTZMAN
```

This variation refers to all messages in which the "received from" address contains "LBOLTZMAN"; e.g. LBoltzman or LBoltzmankowski.

Illegal Syntax

Note that you cannot use more than one range qualifier in a command, nor can you combine message numbers and range qualifiers on the same command. These examples won't work:

```
DIR/LAST=25/SUBJECT=APS
DIR/FROM=WGIBBS 35:52
```

3.1. Addressing Mail

To communicate with other users you will need to know your own address and the addresses of the users with whom you'd like to communicate. To complicate matters, most users have several different addresses that can be used for communications from various computers or networks.

Your Mailing Address

Your MINT address on the VAX is your username. Addresses for mail to/from users on other computers may also include a *host* name; for example, "EAGLE", a *domain* name; for example, "WESLEYAN.EDU", and a *gateway* name; for example, "WESLEYAN.BITNET". The examples below show the syntax for various types of addresses:

JBARNUM — Jane Barnum's address for local mail on the VAX EAGLE.

JBARNUM@EAGLE
Janes's address for users on other Wesleyan computers.

JBARNUM%EAGLE.WESLEYAN.EDU@WESLEYAN.BITNET
Janes's address for users who are on BITNET.

JBARNUM@EAGLE.WESLEYAN.EDU
Janes's address for users who are on the INTERNET.

4. GETTING STARTED WITH MINT

This section describes the basic operation of MINT. It covers invoking MINT, getting help and the screen setup. It also includes listing, reading, sending, and printing messages.

4.1. Invoking MINT

To begin running MINT, at your VAX/VMS prompt type:

```
MINT
```

Near the bottom of the screen you will see the prompt MINT> indicating that you are at the MINT top-level.

4.2. MINT Screen Setup

The screen is divided into four windows:

1. The top window initially shows a list of any messages you have received since the last time you ran MINT.
2. In the lower part of the screen you will see a highlighted status line. This line shows the name of the mail file you are using and information about your messages.
3. Under the status line you will see a command prompt indicating which level of the MINT program is active.
4. The last two lines of your screen are a message window. Error messages are displayed in this window, as well as special messages relative to your current/last MINT operation. VAX/VMS system broadcast messages also appear in this window.

4.3. Getting HELP In MINT

There are three methods for getting help while running MINT:

- Using the HELP command
- Using CTRL/H control sequence
- Using the PF2 key

The HELP Command

The HELP command can be used at all three levels of MINT whenever the cursor is present after the command prompt MINT> , READ> , or SEND>. Type: HELP to see a list of topics for the current level of MINT, or type: `HELP topic` to receive help for a particular topic, for example:

```
MINT>HELP COPY
```

The MINT HELP facility is similar to the VAX/VMS facility in that it is hierarchical; that is, topics have subtopics, subtopics may have sub-subtopics, etc.

Once you have given a HELP command, you can exit by typing CTRL/Z or by pressing the RETURN key one or more times. The best way to understand this system is to try it!

Control/H

CTRL/H can also be used when you see a command prompt. It provides a list of your current command options but does not provide information about using the commands as the HELP command does.

To restore your screen to its previous state, press RETURN or type CTRL/H again.

PF2

When you are composing (editing) a message, you can get help by pressing the PF2 key. On VT200+ series terminals, the HELP key serves the same purpose. The help screen displayed shows a summary of editing control sequences and function keys that can be used while you are composing a message.

4.4. Listing Your Mail Messages

The MINT top-level command DIRECTORY is used for listing messages. Typing simply DIR, as shown below, will list ***all*** of your messages:

```
MINT>DIR
```

The information listed for each message will include message flags, the message number, the date the message was sent, the first part of the sender’s address, the first part of the subject line, and the size of the message; for example:

```
R   1)  8-NOV KBERKNER@EAGLE.WESLE  superhilac  (1368)
RA  2) 10-NOV WGRAHAM@VAX1.QUEENS-  trouble     (311)
N   3) 13-NOV FRED@LBL.BITNET        varia      (1010)
```

The message flag letters indicate the status of a message. Each message will have one or more flag letters including:

N — Message is new; i.e., received since the last time MINT was run.

R	Message has been read.
U	Message is unseen; i.e., it has not been read.
S	Message has special importance. You can mark a message as "special" by typing: MARK/FLAG=SPECIAL at the READ> prompt after you read the message.
A	Message has been answered.
D	Message has been marked for deletion when you exit MINT.

Using Range Qualifiers to List a Subgroup of Your Messages

Range qualifiers such as /NEW or /SUBJECT can be used with the DIRECTORY command to list a subset of your messages.

Examples:

`DIR/LAST` Lists the last message only.

`DIR/LAST=5` Lists the last five messages.

`DIR/UNSEEN` Lists unseen (unread) messages.

`DIR/FROM=WGIBBS`
Lists all messages from WGIBBS.

`DIR/TEXT="staff meeting"`
Lists all messages including the text "staff meeting".

`DIR/TEXT="STaFF mEEting"`
MINT ignores case when searching for the text you specify, so this example will list the same messages as in the preceding example.

`DIR/SINCE=5-DEC-1988`
Lists messages received since December 5th, 1988. It is important to use the format DD-MMM-YYYY (or D-MMM-YYYY for single digit dates). Months are abbreviated to the first three characters and may be in either upper or lower case.

`DIR/Before=15-apr-1988`
Lists messages received before April 15th, 1988.

Ranges were introduced in Section 3: ***Basic Concepts*** (page 6) and a complete list of range qualifiers can be found in the command dictionary under "ranges" (page 48).

4.5. Reading Mail

Reading and processing the mail you receive is the primary purpose of the read-level of MINT. You enter this level from the top-level by giving a READ command. You can use the READ command to read and process a single message; for example, you might type:

```
READ 12
```

followed by:

```
PRINT
EXIT
```

to read and print message 12 and then exit to the MINT top-level.

However, if you have several messages to read it is most efficient to give a single READ command that will read them all; for example:

```
READ/FROM=NBOHR
```

can be used to read all messages from NBOHR. The most common group of messages you will wish to read is the group consisting of ALL new messages that you have received.

Reading All New Messages

You begin reading all new messages by simply typing READ at the MINT> prompt. Your first new message will appear in the upper window of the screen and the highlighted status line will keep you informed of which messages you are reading; the following status line shows that the current message (CM) is number 3 and the the current range (CR) consists of the new messages 3, 4, 5, 6, and 7:

```
File: UD:[PHYS.WGIBBS]MINT.TXT    CM: 3      CR: 3,4,5,6,7
READ>
```

At the READ> prompt you press RETURN to read the next message in the group, or you can issue one of the following commands to take action on the current message.

READ Command Options

`PRINT`	This will store the file in your MAIL-TO-BE-PRINTED.TXT file to be automatically queued when you exit MINT. (Printing is described in more detail following this list of options.)
`DELETE`	This will mark the message for deletion. It will not be deleted until you exit from MINT or issue the EXPUNGE command at the top-level of MINT.
`KILL`	This option is like DELETE above, but it also begins reading your next message.

`PREVIOUS`	This command will reread the message before the current message.
`TYPE`	The TYPE command will retype the message you have just read; i.e., the current message.
`REREAD`	The REREAD command is equivalent to TYPE.
`REPLY`	Use the REPLY command to send a reply to the sender of the current message. The REPLY command moves you automatically to send-level, but when you mail the reply you will be returned to the read-level and can continue reading messages. REPLY is discussed in more detail later in this section (page 15).
`FORWARD`	The FORWARD command also moves you temporarily to the send-level of MINT. This command is used to forward the current message to one or more users.
`EXIT`	The EXIT command is used to exit from the read-level to the top-level of MINT.

For a complete list of read-level commands see Section 6: ***MINT Dictionary*** (page 49).

4.6. Printing Mail Messages

You can print your mail messages from either the top-level or read-level of MINT using the MINT command PRINT. Each time you issue a PRINT command, the specified messages are written to a file in your home directory called MAIL-TO-BE-PRINTED.TXT. When you EXIT or QUIT the MINT program the file is automatically queued to the system lineprinter. Output can then be picked up in room 520, Science Center.

Note: The file MAIL-TO-BE-PRINTED.TXT will also be printed if if you interrupt MINT by typing CTRL/Y, or if you exit from MINT top-level by typing CTRL/Z.

Printing from the TOP-LEVEL

When you use the PRINT command from MINT's top-level MINT> prompt, you must specify which messages are to be printed. There are several formats that you can use:

PRINT/range_qualifier

You can use range qualifiers with the PRINT command as described previously for the DIRECTORY command.
Example: `PRINT/UNSEEN`

PRINT message_number

Example: `PRINT 13`

PRINT beginning_message_number:ending_message_number

Example: `PRINT 13:20`

Printing from the READ-LEVEL

After reading a message you can have it printed by typing PRINT at the READ> prompt. The message will be stored in the file MAIL-TO-BE-PRINTED.TXT just as it is when the PRINT command is used from top-level. Similarly, when you exit or quit MINT, the file is automatically queued to the lineprinter.

Having Messages Printed Immediately

The qualifier /IMMEDIATE can be used to have messages queued immediately, rather than when you exit or quit MINT. For example, after the MINT> prompt, type:

```
PRINT/IMMEDIATE 16:20
```

or:

```
PRINT 16:32/IMMEDIATE
```

A message in the message window at the bottom of the screen will tell you when the file has been printed.

Having Messages Printed on a Local Printer

If you have a printer connected to your terminal, you can instruct MINT to print files on your printer rather than queue files to the system lineprinter. After either the MINT>, READ>, or SEND> prompts, type:

```
SET PRINT_COMMAND=LOCAL
```

If you wish to make this definition permanent, next type:

```
WRITE
```

The SET and WRITE commands are explained in detail in section 5: ***Expanding Your Capabilities*** (page 53).

4.1. Sending Mail

SEND, REPLY, REMAIL, and FORWARD are the four MINT commands that are used for sending messages. All four of these can be used both from the top-level and the read-level. Briefly, these commands function as follows:

SEND	This is the basic MINT command for sending a message. You are first prompted TO: and SUBJECT: and you then begin composition of the body of the message.
REPLY	This variation is used to REPLY to a message you have already received, most commonly the message you have just read. The address and the subject will automatically be taken from the FROM: and SUBJECT: lines of the message to which you are replying.

FORWARD — The FORWARD command is used to send an existing message, usually the one you have just read, to another address. After the TO: prompt (for the forwarding address) and the SUBJECT: prompt, you are placed into message composition mode to create a cover-letter that will precede the message being forwarded.

REMAIL — REMAIL is similar to FORWARD, except that you are not placed in the composition mode to add your comments to the beginning of the message you are forwarding.

The SEND Command

The TO: Prompt

SEND works similarly whether you invoke it from MINT top-level or from the read-level. In both cases, you are first prompted: TO: for the recipient's address. You can reply by typing a single address, or several addresses separated by commas, or an ***identifier*** that is an abbreviation for one or more addresses. Identifiers will be discussed in Section 5: ***Expanding Your Capabilities*** (page 19). The bottom of your screen will be similar to:

```
File: UD:[PHYS.WGIBBS]MINT.TXT.1
TO: NBOHR,WLAMB
```

The SUBJECT: Prompt

You will next be prompted for the subject of the message. The subject will be displayed in MINT directory listings and in interactive notifications. This prompt also appears below the status line as follows:

```
File: UD:[PHYS.WGIBBS]MINT.TXT.1
SUBJECT: Bevatron Run
```

Composition Mode

After entering the subject and pressing RETURN, the upper screen clears, and the lower screen looks similar to:

```
PF1 exits to Send>;PF2 gets Help;PF3 inserts file;PF4 sends message
Editing session is not being journaled
```

During the composition stage your main task usually is to compose the body of the message you are sending. However, you have many options, four of which are summarized on the highlighted status line. A description of these and other options follows:

PF1 — Pressing this function key makes the SEND> prompt visible so that you can use any of the send-level commands. Send-level commands will be discussed after the send variations: REPLY, FORWARD, and REMAIL.

PF2 — This key provides a help screen with a complete list of the editing and function keys that can be used during message composition. To restore your screen to its previous state just press RETURN. (On VT200+ terminals, the HELP works the same as PF2.)

PF3 — This key is used to have a VAX/VMS file included in your message. A file prompt appears as shown:

```
PF1 exits to Send>;PF2 gets Help;PF3 inserts file;PF4 sends message
File to insert:
```

The file you specify will be placed at the current cursor position in the composition window.

File insertion provides a means of having "enclosures", and it also enables you to deal safely and efficiently with long messages. You may have noted the message:
`Editing session is not being journaled`
displayed two screens above; this means that in the event of a hardware failure, or other unforeseen problem, your partly composed message may be lost. If you create a file outside of MINT using the VAX/VMS editor, your editing session IS journaled; that is, your keystrokes are saved to a file for possible recovery. You can then invoke MINT and include the finished file in a message using PF3.

PF4 — Pressing this key will mail your message.

Control/W — CTRL/W can be pressed to refresh the screen. This is useful if it has become garbled for any reason.

Control/A — Toggles between the insert and overstrike modes. Most control sequences available when using the VAX editor are allowed during composition. Press PF2 to see the list.

The REPLY Command

REPLY can be used at either the top-level or read-level to reply to an existing message. If used at the read-level, MINT will assume you are replying to the current message (the one last read). If used at the top-level, you must supply a message number using the /MESSAGE qualifier as in this example:

```
File: UD:[PHYS.WGIBBS]MINT.TXT.1
MINT>REPLY/MESSAGE=42
```

REPLY works similarly to SEND with the following exceptions:

- You will NOT be prompted for TO: and SUBJECT:. The address will be taken from the FROM line of the message to which you are replying, and the subject will be from the SUBJECT line, but will be prefaced with RE:.

- After issuing the command, the screen clears and the message to which you are replying is displayed in the top window. The next window down is used to compose the reply.
- The minus sign - on the keypad of VT100+ series terminals can be used to switch back and forth between windows. See the "REPLYING WITH STYLE", page 25 for other commands to work with both windows.

The FORWARD Command

FORWARD can be used at either the top-level or read-level to forward an existing message. If used at the read-level, MINT will assume you are forwarding the current message (the one last read). If used at the top-level, you must supply a message number using the /MESSAGE qualifier as in this example:

```
File: UD:[PHYS.WGIBBS]MINT.TXT.1
MINT>FORWARD/MESSAGE=42
```

After you issue this command, you will see a blank composition window as when using the SEND command.

However, when you mail the message, the message to which you are replying will be appended to any new text written. Other than this, FORWARD functions similarly to the SEND command.

The REMAIL Command

The REMAIL command functions like the FORWARD command with this exception:

After issuing the command, and clearing the composition window, the SEND> command prompt appears at the bottom of the screen. You can type: SEND to send the message, COMPOSE to change to composition mode, or you can issue other send-level commands as described in the next section.

Send-level Commands

If you press PF1 any time while composing a message, the SEND> command prompt will appear at the bottom of the screen and you may then give a send-level command. Most send-level commands perform some function on the message you are currently sending; for example, you may want to give a command to check spelling, change or add an address, or execute a VAX/VMS command. Send-level commands include:

COMPOSE This command is used to resume composition mode. You can use this command if you wish to make changes or additions to the body of a message before sending it. (COMPOSE and PF1 toggle you between command and composition modes.)

QUIT	Exits the send-level WITHOUT sending the message. You will be returned to the level from which you entered the send-level.
EXIT	Same as quit.
Control/W	This command refreshes, i.e. redisplays, the screen.
BLANK	The BLANK command clears the screen. (It can be used at MINT> and READ> prompts as well.)
LOGOUT	Logs you off the VAX. Your message will be lost unless a SAVE has been executed.
TYPE	This command can be used if you entered the send-level from the read-level; it types out the last message read.
DCL	This command is useful if you'd like to give a VAX/VMS command; for example,

```
PF1 exits to Send>;PF2 gets Help;PF3 inserts file;PF4 sends message
SEND>DCL TYPE PHONE.NUMBERS
```

After the command is executed, the message:

```
Hit return to continue:
```

will be displayed. After you press RETURN, you will again be in MINT at the SEND> prompt.

If you type DCL alone at the SEND> prompt, you will temporarily leave MINT and will see your VAX prompt. (In VAX terms, you will have spawned a subprocess.) To resume your MINT session type: LOGOUT.

SUBJECT	This command can be used to change the subject of a message. The format is **SUBJECT "text"**, for example:

```
PF1 exits to Send>;PF2 gets Help;PF3 inserts file;PF4 sends message
SEND>SUBJECT "Scattering Calculation"
```

Note that if the subject text is *not enclosed in quotes*, the new subject will be converted to *upper case*.

SAVE	This command saves the current message being created to a file. It is useful if you don't have time to finish a message, but want to save what you have typed so far. Later you can resume where you left off by using the RESTORE command; i.e. type RESTORE at the MINT> prompt.
ADD	This command is used to add addresses. The qualifier /TO will allow you to add an address to the TO list; /CC is used to

add carbon copy addresses and /BCC is used to add blind carbon copy addresses. (The message will be sent to the blind carbon copy addresses, but *unlike* TO addresses and CC addresses, BCC addresses are not listed in the message). In the following example, MBORN is added to the TO list and NBOHR is added to the CC list:

```
PF1 exits to Send>;PF2 gets Help;PF3 inserts file;PF4 sends message
SEND>ADD/TO=MBORN/CC=NBOHR
```

REMOVE — REMOVE is used to remove addresses from the message being sent. The qualifiers /TO, /CC, and /BCC are used as with the ADD command; for example:

```
PF1 exits to Send>;PF2 gets Help;PF3 inserts file;PF4 sends message
SEND>REMOVE/TO=ESCHRODINGER/CC=NBOHR
```

INSERT — Inserts a file into the message. It is similar to pressing PF3 key in message composition mode.

SPELL — Issuing the SPELL command will run the SPELL program on the message you are composing. For more information on the SPELL program, at the VAX/VMS prompt (outside MINT) type:

```
HELP SPELL
```

DISPLAY — This command will display the message being sent. You can use the command to confirm other changes you have made. If you note a mistake in the body, you can use the COMPOSE command to resume composition mode; the ADD, REMOVE, and SUBJECT commands can be used to make changes in the message header.

5. EXPANDING YOUR CAPABILITIES

5.1. Creating a Mailing List File

MINT’s mailing list feature allows you to define abbreviations for addresses and groups of addresses you use frequently. To create a mailing list:

1. Use the VAX editor to create a file named: MAILING-LIST.TXT in the home directory of your home account. Follow the example below. For a

more detailed annotated example, you can also read the system mailing list by typing:

```
TYPE SYS$MINT:MAILING-LIST.TXT
```

2. Next, run the program MLIST by typing:

```
MLIST
```

MLIST converts this file into a format that can be used by MINT. This converted file is called MAILING-LIST.IDX. For more information on MLIST, at your VAX prompt type:

```
HELP MLIST
```

A sample mailing list follows:

```
; sample mailing list
WERNER=WHEISENBERG
FRED=FRED@LBL.BITNET
ALL=BILL,WERNER,WGIBBS
BILL=
    EBJ%VAX1.QUEENS-BELFAST.AC.UC@CUNYVM.EDU
MJJ=BILL,FRED,-
        KWILLIAMS@KLB,-
        AMICHELSON
```

WERNER, FRED, ALL, BILL and MJJ are ***identifiers.*** When running MINT these identifiers can be used in place of addresses at the TO, CC, and BCC prompts, and as values to the qualifiers /TO, /CC, and /BCC.

Syntax notes:

- Comment lines begin with a semicolon (;)
- The general format is ***identifier = list of one or more addresses***
- Continuation lines are indicated with a hyphen (-) at the end of the preceding line; see the definition for MJJ.
- A hyphen (-) is not needed when you end a line with the equal sign (=) following the identifier; see BILL.
- Identifiers will be converted to upper case by MLIST, so case is not significant.
- Addresses are not converted to upper case, since some mail systems are case-sensitive. Note, in particular, that Unix systems are generally case-sensitive and usually use lower case identifiers.

5.2. Customizing MINT to Meet Your Needs

There are over twenty parameters that you can control to customize MINT. By changing the default settings you can have automatic carbon copies generated, change the prompt names, ask for a signature file to be appended to your messages, and much more. This section explains how to change parameters and provides a brief explanation of each.

How to Customize MINT

There are two steps to customizing MINT:

1. Use the SET command to specify a parameter and its new value; for example:

```
File: UD:[PHYS.WGIBBS]MINT.TXT.1
MINT>SET CARBON_COPY=ON
```

   You should then see a success message confirming the new definition:

```
File: UD:[PHYS.WGIBBS]MINT.TXT.1
MINT>
Variable CARBON_COPY has a value of ON
```

2. Use the WRITE command to save your new environment; for example:

```
MINT>WRITE
```

The SET and WRITE commands can be used at all three levels whenever a command prompt is visible. Also, at all three levels you can see a list of parameters and a description of parameters by using the HELP command, and you can show the current setting of a parameter using the SHOW command. Examples:

```
HELP SET
HELP WRITE
HELP SET CARBON_COPY
SET CARBON_COPY=ON
SHOW CARBON_COPY
```

Summary of Parameters

CARBON_COPY determines whether or not there will be a prompt for carbon copy addresses when using the SEND command.

Possible values: ON or OFF
Default: OFF
Shortest abbreviation: CA

BLIND_CARBON_COPY
determines whether or not there will be a prompt for blind carbon copy addresses when using the SEND command. Blind carbon copy addresses DO NOT appear in the message header and are therefore "blind" to receivers of the message.

Possible values: ON or OFF
Default: OFF
Shortest abbreviation: BL

USE_EDITOR_AUTOMATICALLY
determines if the VAX/VMS editor is automatically invoked when using the SEND command. If you DO NOT change this parameter, you will be using the MINT editor to compose your messages. If you set the value of this parameter to ON, you will automatically use the editor that it is invoked when you issue the command EDIT at your VAX/VMS prompt; e.g.,

```
EDIT MAILING-LIST.TXT
```

Possible values: ON or OFF
Default: OFF
Shortest abbreviation: USE

EXPUNGE_ON_EXIT
determines if messages marked deleted are expunged (permanently deleted) when the EXIT command is used.

Possible values: ON or OFF
Default: ON
Shortest abbreviation: EXP

SAVE_LAST_MESSAGE_SENT
determines if the ***last*** message sent is saved to a file. The file name will be MINT.CPY unless you change its name by setting a new value for the parameter LMS_FILE.

Possible values: ON or OFF
Default: ON
Shortest abbreviation: SAVE_L

SAVE_MAIL_SENT determines if ***all*** mail sent is saved to a file. The file name will be MINT.SENT unless you change its name by setting a new value for the parameter SMS_FILE.

Possible values: ON, OFF, and PROMPT
Default: OFF
Shortest abbreviation: SAVE_M

INCLUDE_SIGNATURE_FILE
A signature file is a file that contains one or more lines of text to be appended to the bottom of messages sent.

The INCLUDE SIGNATURE FILE parameter determines whether or not this file should be used. If the parameter has a value of ON, MINT looks for the file MINT.SIGNATURE in the home directory of your home account (your default directory when you first login). You can specify a different name or directory by changing the value of the parameter SIGNATURE_FILE.

Possible values: ON, OFF, PROMPT

Shortest abbreviation: INC

PAGING

This parameter controls whether or not the screen should be automatically paused when full. When the value is OFF (the default), the screen continues to scroll until the end of a message has been reached. To stop the screen it is necessary to press Hold Screen, or to toggle between CTRL/S and CTRL/Q. When the value is ON, scrolling stops when the screen is full, and this message is displayed in the message window:

```
File: UD:[PHYS.WGIBBS]MINT.TXT    CM: 3      CR: 3,4,5,6,7
Press [return] to continue or Q [return] to Quit:
```

Possible values: ON, OFF
Default: OFF
Shortest abbreviation: PA

PRINT_COMMAND

The value to this parameter controls where your mail messages are printed. If you have a local printer attached to your terminal, the following command will instruct MINT to use it for messages you ask to have printed using MINT's PRINT command:

```
File: UD:[PHYS.WGIBBS]MINT.TXT    CM: 3      CR: 3,4,5,6,7
READ>SET PRINT_COMMAND=LOCAL
```

The value can also be any valid VAX/VMS PRINT command, for example:

```
MINT>SET PRINT_COMMAND=PRINT/FORMS=51/DELETE
```

Note: If /DELETE is not used, the file MAIL-TO-BE-PRINTED.TXT will not be deleted after it is printed.

The default setting is:

```
print/delete/queue=sys$print.
```

Shortest abbreviation: PR

DEFAULT_CC_LIST

You can use this parameter to specify a carbon copy address for your mail. For example, at a command prompt type:

```
SET DEFAULT_CC_LIST=WGIBBS
```

WGIBBS will now automatically receive copies of all messages sent.

Shortest abbreviation: DEFAULT_B

DEFAULT_BCC_LIST
This parameter is used to specify blind carbon copy addresses to be used each time mail is sent; for example,

```
SET DEFAULT_BCC_LIST=WGIBBS
```

Specifying your own address as the value to the DEFAULT_BCC_LIST is one method of saving a copy of each message you write.

REPLY_TO
The REPLY TO parameter is used to add a "reply to" line to the header of the message being sent to let the recipients of the message know they should reply to the specified address rather than the address on the "from" line.

If you read a message with a "reply to" line, and then use the REPLY command to reply, your message will automatically be sent to the address on the "reply to" line. A sample message with a "reply to" line follows:

```
DATA: 7-DEC-1988 14:40:04.04
FROM: WGIBBS@EAGLE.WESLEYAN.EDU
SUBJECT: SAMPLE HEADER WITH REPLY LINE
TO: AAVAGODRO
REPLY-TO: GIBBS@LBL.BITNET

HELLO Amadeo...
```

Shortest abbreviation: REP

PERSONAL_NAME
The PERSONAL NAME parameter is used to include a non-computerized name of your choice on your mail. For example, if you set your personal name as follows:

```
SET PERSONAL_NAME="Amadeo Avagodro"
```

recipients of your messages would see the "from" line:

```
From: Amadeo Avagodro <AAVAGODRO@EAGLE.WESLEYAN.EDU>
```

Note that the quotes in the above example, although not required, preserve the case of letters in the personal name.
Shortest abbreviation: PER

DONT_DSPLY_HEADER
Use this parameter if you don't want to read all of the header information that precedes the body of a message. The value of DONT_DISPLY_HEADER can be a list of one or more words (TO, subject, XYZ, etc.) separated by commas. Header lines that contain these items will not be displayed when you read your mail.

```
SET DONT_DISPLY_HEADER=TO,FROM
```

In the above example, the "to" and "from" lines would not be displayed, but any other lines of the header not containing "to" or "from" would be displayed. It is not significant whether you use upper or lower case to define the list; i.e., it is case-insensitive.

Shortest abbreviation: DONT

TOP_PROMPT This parameter allows you to change the top-level prompt MINT> to a prompt of your choice.

Shortest abbreviation: TOP

READ_PROMPT This parameter allows you to change the read-level prompt READ> to a prompt of your choice.

Shortest abbreviation: READ

SEND_PROMPT This parameter allows you to change the send-level prompt SEND> to a prompt of your choice.

Shortest abbreviation: SE

MAILING_LIST This parameter lets you change the name and location of your mailing list file. Normally MINT looks in the home directory of your home account for a file named MAILING-LIST.IDX. For information on how to create a mailing list file, see "CREATING A MAILING LIST FILE", page 18. Include device, directory, and full filename (or logical name) as in this example:

```
MINT>SET MAILING_LIST=UD:[CHEM.LIBRARY]MAILING-LIST.IDX
```

Shortest abbreviation: MAI

LMS_FILE The LMS_FILE parameter is used to specify the file name that contains the last message sent. The default file name is MINT.CPY. This parameter is only used when the parameter SAVE_LAST_MESSAGE_SENT has a value of ON. Include device, directory, and full filename (or logical name); for example, type:

```
SET LMS_FILE=UD:[S91.WGIBBS]LAST.SENT
```

Shortest abbreviation: LMS

SMS_FILE The SMS_FILE parameter is used to specify the file name that contains all messages sent. This parameter is used in conjunction with the parameter SAVE_MAIL_SENT. Include device, directory, and full filename; for example, type:

```
SET SMS_FILE=UD:[S91.WGIBBS]COPIES.TXT
```

Shortest abbreviation: SMS

SIGNATURE_FILE

The SIGNATURE FILE parameter is used to specify a file name for your signature file other than MINT.SIGNATURE in the home directory of your home account. This parameter is used in conjunction with the parameter INCLUDE_SIGNATURE_FILE. Include device, directory, and full filename (or logical name); for example, type:

```
SET SIG=UD:[S91.WGIBBS]MSG.TXT
```

Shortest abbreviation: SIG

5.3. Replying with Style

If you receive many messages a day, and find yourself sending replies such as "YES, YES and NO" to save time, you may find the following technique useful.

How to answer or comment on a series of items

Instructions follow to copy blocks of text from the upper window (containing the message to which you are replying) to the reply window, mark copied lines with the symbols ">>" so that they will stand out, and finally fill in comments or answers after each marked block of text.

1. Begin by issuing the REPLY command.
2. Move the cursor to the upper window that contains the message to which you are replying. PF8 which is the minus sign - on the keypad (VT100+ series terminals) toggles you between screens.
3. Select (highlight) one or more questions or items that you plan to answer. Use the SELECT key or keypad 7 key to begin the selection, move with the arrow keys to highlight text, then press REMOVE or keypad 8. (This saves the text in your "cut/paste" buffer.)
4. Toggle to the reply window by again pressing the minus sign key - on the keypad, and then deposit the text by pressing INSERT or keypad 9.
5. The next step is to mark the questions so that they will stand out from your responses. To do this, move the cursor anyplace on the first paragraph, and then press the ENTER key on the keypad. Repeat this for each paragraph. Each paragraph will be filled and indented with ">>" inserted at the beginning of each line.

Your screen will now be formatted as follows, and you should be able to move quickly to the various items and fill in short replies.

```
Dear Bill,
Are you planning to attend the SIGUSC conference?

Come a day early and present a seminar here.

Any news on the magnetic tape queueng program?

Text of the message to which you are replying
>>Are you planning to attend the SIGUSC conference?
     Yes
>>Come a day early and present a seminar here.
     On one condition,
>>Any news on the magnetic tape queueng program?

PF1 exits to SEND>; PF2 gets help; PF8 changes windows; PF4 sends msg
```

5.4. Organizing Your Messages

If you use electronic mail for several distinct applications, and you send and receive many messages, it may be useful for you to organize your messages into several mail files. Suppose, for example, that you were on the local organizing committee for a conference being hosted at Wesleyan. It might be convenient to have all messages related to the conference stored in a separate message file. You would use the MOVE command to transfer messages from your normal message file MINT.TXT to the conference mail file; for example the following moves message 3 (CM: 3) to file ICPEAC:

```
File: UD:[PHYS.WGIBBS]MINT.TXT    CM: 3      CR: 3,4,5,6,7
READ>MOVE ICPEAC
```

When you chose to work on conference related mail, you would use the GET command to switch to your conference mail file; for example:

```
File: UD:[PHYS.WGIBBS]MINT.TXT.1
MINT>GET ICPEAC
```

Other suggestions that might apply follow:

1. Create a mailing list with an *identifier* for other persons on the local organizing committee. Also include identifiers for persons at other universities with whom you will correspond frequently. See instructions in section 5, page 18. Example:

```
ICP=NBOHR,AMICHELSON,WGRAHAM
```

2. Create a signature file named MINT.SIGNATURE consisting of your BITNET and internet addresses, and any special reminders. For example:

   Bitnet: emorley%eagle.wesleyan.edu@wesleyan.bitnet
   Internet:emorley@eagle.wesleyan.edu
   2nd MAILING: DECEMBER 15, 1986

3. Save a copy of all your messages. There are two techniques that can be used:

   - Set the parameter SAVE_MAIL_SENT to ON:

```
File: UD:[PHYS.WGIBBS]MINT.TXT.1
MINT>SET SAVE_MAIL_SENT=ON
Variable  SAVE_MAIL_SENT has a value of ON
```

   All messages you write will be saved in a file called MAIL.SENT. To review these messages type:

```
GET MAIL.SENT
```

   - Set the parameter DEFAULT_BCC_LIST to your address; for example:

```
File: UD:[PHYS.WGIBBS]MINT.TXT.1
MINT>SET DEFAULT_BCC_LIST=WGIBBS
Variable  DEFAULT_BCC_LIST has a value of WGIBBS
```

   Now you will receive a carbon copy of all your messages. (Your name will not be listed on the CC: header line, however since it is a "bind" carbon copy.) You can use the MOVE command to move messages pertaining to the conference to your conference message file; for example at the MINT> prompt, type:

```
MOVE ICPEAC 13:15,20
```

   This would move messages 13, 14, 15, and 20 to the ICPEAC mail file.

5.5. Dealing with the Unexpected

1. SYSTEM CRASH, or other interruption including premature LOGOUT: In the event that your work is interrupted before you have mailed a partly composed message, you <u>may</u> be able to retrieve the body of the message from the temporary file whose name begins MAIL-BEING-SENT.

   - Before invoking MINT, check for the existance of such a file in the home directory of your home account by typing:

```
V SYS$LOGIN:MAIL-BEING-SENT*
```

- Notice the time, date, and size of any files listed. The most recent one should be the body of your interrupted message. (Write the filename down since it will have disappeared from the screen when you need to use it.)
- Invoke MINT, issue the SEND (REPLY, etc.) command, as when you originally began the message.
- Once in message composition mode, press PF3 and then after the prompt, type the full name of the MAIL-BEING-SENT file; for example:

```
PF1 exits to Send>;PF2 gets Help;PF3 inserts file;PF4 sends message
File to insert: SYS$LOGIN:MAIL-BEING-SENT.57252765
```

2. RECOVERING FROM AN EVE JOURNAL FILE
If you were using the VAX/VMS editor EVE to compose your message, then check for the existance of a journal file in the directory from which you invoked MINT and check for the existance of a file whose name begins MAIL-BEING-SENT in your login directory; e.g.,

```
V  MAIL-BEING-SENT.TJL
V  SYS$LOGIN:MAIL-BEING-SENT*
```

As above, note the time, date, and size of files listed, and write down the name of any likely MAIL-BEING-SENT file. *Procede with this technique if you have both a .TJL (journal file*) and a MAIL-BEING-SENT file.

To attempt recovery of information in the journal file MAIL-BEING-SENT.TJL; follow this example:

```
EDIT/RECOVER SYS$LOGIN:MAIL-BEING-SENT.57252765
```

Chapter 3 of the ***User's Guide to Computing*** discusses recovery procedures in more detail.

3. MISSING MINT.TXT FILE: Hopefully this will never happen to you. However, if it does, try these steps before you invoke MINT:
   - First make sure you are looking for the file in the home directory of your home account. This is your default directory when you first login.
   - Secondly, check for the existance of a file called MAIL-BEING-EXPUNGED.TXT. If it does exist you may rename this file MINT.TXT;1 as follows:

   ```
   RENAME MAIL-BEING-EXPUNGED.TXT MINT.TXT;1
   ```

   - If you do not find such a file, please report the problem to the VAX consultant. Also ask the consultant for a "File Restoration Form". The file MINT.TXT can be restored from the last system backup tape.

4. DAMAGED MINT.TXT FILE: If the file MINT.TXT exists but appears to be damaged, please see the VAX consultant, who will direct you to a WCC staff member who can help. In most cases, your message file can be comletely repaired.

5. GARBLED SCREEN: If your screen becomes garbled, it can be refreshed by typing CTRL/W, or the REFRESH command can be used if a command prompt is visible.

6. MESSAGE SENT PREMATURELY: You can't stop the message you've already mailed from being delivered; however, at the top-level MINT> prompt, you can type: RESTORE to restore the message just as it was before you pressed PF4 or typed SEND. The message will be restored from the file MINT.CPY (or the file you have defined with LMS_File).

5.6. Understanding the MINT File System

MINT.SIGNATURE and MAILING-LIST.TXT are two MINT related files that you might create using the VAX editor. In addition to these, the MINT program creates a variety of files for processing your messages. Most of them, such as MAIL-TO-BE-PRINTED, are automatically deleted by MINT when they are no longer needed. This section will tell you about MINT related files and under what circumstances they can be deleted.

MINT.TXT;1 — Messages you receive from MINT are stored in this file. It appears in your directory when you receive your first MINT message.

MINT.CPY;n — You will see up to 3 files with this name. These are copies of the 3 most recent messages you have sent. You can use the VAX command TYPE to review these files; e.g. `TYPE MINT.CPY;2`. Three possible ways to use these files include:

1. Use the VAX/VMS TYPE command to see the contents.

2. In MINT, use the RESTORE command; for example:

   ```
   RESTORE MINT.CPY;2
   ```

   to first restore the message, then make changes, and then SEND the message.

3. In MINT composition mode, use the INSERT command or PF3 key to include the message in the body of the message you are composing.

DELETION: These files can be deleted if you don't anticipate using them again.

MINT.SIGNATURE

A signature file contains text that MINT automatically appends to the bottom of your messages. The parameter

INCLUDE_SIGNATURE_FILE determines whether or not the file is active. See the section "Customizing MINT to Your Needs" for information on creating and using signature files.

DELETION: This file can be deleted if you do not wish to use it; an alternative is to set the parameter INCLUDE_SIGNATURE_FILE to OFF; for example:

```
SET INCLUDE=OFF
```

MINT.INIT

This is an "initialization" file that MINT reads each time you run MINT. You create this file by using the MINT command WRITE. You change a parameter setting in this file by using the MINT command SET and then WRITE.

DELETION: If you delete this file, MINT will use default values for all parameters.

MAILING-LIST.TXT

This is a mailing list file that you create using the VAX editor. For instructions see Section 5: ***Expanding Your Capabilities*** (page 18).

DELETION: This file is not actively used when in MINT, but it will be needed if you wish to make changes to your mailing lists. See MAILING-LIST.IDX below.

MAILING-LIST.IDX

This file is created by MLIST from MAILING-LIST.TXT. It is the "MINT readable" form of the file MAILING-LIST.TXT. You can select a name other than MAILING-LIST.IDX by changing the parameter MAILING_LIST with the SET command.

DELETION:This file must be present for MINT to access your mailing list identifiers. However, if you have mistakenly deleted it but saved MAILING-LIST.TXT, you can recreate it using MLIST.

MAIL-BEING-SENT.nnnnnnnn

This is a ***temporary*** file created by MINT when you compose a message. It is automatically deleted when your message is mailed so you normally do not see the name listed in your directory. If you DO see this file listed, you may have interrupted MINT by typing CTRL/Y, or you may be logged in on two different terminals.

DELETION: This file should be deleted, UNLESS you are running MINT from two different terminals.

MAIL-BEING-REPLIED.nnnnnnnn

Similar to MAIL-BEING-SENT.nnnnnnnn above, but it is created when the REPLY command is used.

MAIL-TO-BE-PRINTED.TXT

Each time you issue the MINT command PRINT after reading a message, it is stored in this file. When you EXIT from MINT the file is automatically queued to the lineprinter and deleted when printed.

Note: If you change the print command using the SET PRINTER command and don't include the /delete qualifier, this file will not be deleted automatically.

DELETION: Notice the date on the file; if it is more than a few days old, the file will probably never be printed unless you intervene -- the system may have crashed before the file was queued by MINT to the printer -- or you may have interrupted with CNTRL/Y. Usually, files are printed within half hour on normal computer paper, and twice a day on special forms. If you still want the information in the file, type:

```
PRINT MAIL-TO-BE-PRINTED.TXT/DELETE
```

6. MINT COMMAND DICTIONARY

This section summarizes and provides examples for most MINT commands, options and parameters. Each entry includes, a short description, an example and a list of related entries. For more information on a particular item, use the MINT HELP facility. Items are classified as follows:

Usage

MINT>	A top-level MINT command
READ>	A read-level MINT command
SEND>	A send-level MINT command
Compose:	An option while composing (typing in) a message
Concept:	A MINT concept or term
Qualifier:	A keyword used to qualify a MINT command
Parameter:	A keyword that can be set with the SET command to customize the MINT environment.

ADD — Usage: SEND>

The ADD command allows you to add address(s) to the TO, CC, and BCC lines when sending a message.
Example:

```
SEND>Add/cc=Jsmith/bcc=Khillman
```

Jsmith will be added to the list of addresses to receive a carbon copy of the current message; Khillman will receive a blind carbon copy.
See also: REMOVE

Address — Usage: Concept

The computer address of a user with respect to a sender; your address for a user on your host computer is your username; your address for a user on another computer at Wesleyan includes the name of your host computer.
Example 1:

```
WGIBBS@MV2
```

This is an example of an address used to send mail from one Wesleyan computer to another. A user on the "Eagle" would address mail to a user on "MV2" in the Wesleyan Chemistry Department using this format.

Example 2:

```
JSMITH%EAGLE.WESLEYAN.EDU@WESLEYAN.BITNET
```

This example shows JSMITH’s address used by a person at another BITNET university to send mail to Wesleyan.

See also: TO:, CC:, BCC:, SEND command

/ALL Usage: Qualifier

The /ALL range qualifier specifies that a given command be performed on all messages in the file.
Example:

```
PRINT/ALL
```

See also: COPY, DIRECTORY, DELETE

/ANSWERED Usage: Qualifier

The /ANSWERED range qualifier specifies that a given command be performed on all answered messages.
Example:

```
DELETE/ANSWERED
```

See also: PRINT, DIRECTORY

/BCC Usage: Qualifier

The qualifier /BCC allows you to specify a "blind carbon copy address". Blind carbon copy addresses do not appear in the message header at the beginning of a message. This qualifier can be used with the ADD, REMOVE, SEND, REPLY, REMAIL, and FORWARD commands.

To specify more than one address, enclose the list in parentheses and separate addresses with commas.
Example: REMOVE/BCC=TEDISON
See also: /CC, BLIND_CARBON_COPY

/BEFORE Usage: Qualifier

The /BEFORE range qualifier specifies that a given command be performed on all messages delivered before the specified date. Date may be one of the following TODAY, YESTERDAY, or any valid VMS date specification.
Example:

```
DELETE/BEFORE=13-NOV-1988
```

See also: /SINCE, Dates

Note: Date qualifiers are compared to the date that the message is delivered to the recipients MINT.TXT file.

BLANK Usage: MINT>, READ>, SEND>

The blank command clears the display and message windows. The display is the window at the top of the screen where commands such as DIRECTORY, READ, etc. output text. The message window is the window at the bottom of the screen. This window contains success, warning, and error information.
Example:

```
BLANK
```

See also: REFRESH

BLIND_CARBON_COPY Usage: Parameter

The BLIND CARBON COPY parameter determines if blind carbon copy addresses are prompted for when using the SEND command. The parameter accepts either the keyword ON or OFF. The default is set to OFF.
Example:

```
SET BLIND_CARBON_COPY=ON
```

See also: /BCC

CARBON_COPY Usage: Parameter

The CARBON COPY parameter determines if carbon copy addresses are prompted for when using the SEND command. The parameter accepts either the keyword ON or OFF. The default is set to OFF.
Example:

```
SET CARBON_COPY=ON
```

/CC Usage: Qualifier

The /CC qualifier allows you to specify one or more addresses to be added to the CC (Carbon Copy) header.

To specify more than one address, enclose the list in parentheses and separate addresses with commas.
Example:

```
SEND/cc=BRUBBLE
```

See also: CARBON_COPY, BLIND_CARBON_COPY, /BCC

Case Sensitive Usage: Concept

"Case sensitive" and "case insensitive" refer to the way a program such as MINT processes your upper case, lower case, and mixed case text. MINT performs case insensitive searches. When you use the command DIR/TEXT="Staff Meeting", MINT searches for ALL occurrences of "staff meeting" in any combination of upper and lower case letters. If the search had been case sensitive, MINT would have required an exact match of upper and lower case characters.

Command Recall Usage: Concept

Each level of mint has a separate command recall buffer of up to twenty commands. To recall a previous command use the up arrow. You can then modify the command and press RETURN to execute.

COMPOSE Usage: SEND>

The compose command allows you to resume composition mode from the SEND> command prompt. You can toggle between composition mode and send command mode by using COMPOSE at command level and PF1 at composition level.

```
SEND> COMPOSE
```

See also: EDIT

Control/H

Usage: MINT>, READ>, SEND>

CTRL/H lists the currently available commands. CTRL/H is a toggle. Using it once gives you a pop down screen. Using it a second time returns the previous screen. It is useful if you don't know the available commands. For more detailed help on a command use the HELP command.

See also: HELP

Control/W

Usage: MINT>, READ>, SEND>

CTRL/W executes the refresh command. The refresh command refreshes the screen. It is useful if the screen has been garbled.

See also: REFRESH, BLANK

Control/Z

Usage: Control Key

CTRL/Z is a general exit command. At a command prompt it is equivalent to the EXIT command; in message composition mode it is equivalent to typing PF1 to move from composition mode to SEND command level. CTRL/Z can also be typed in response to the TO:, CC:, or SUBJECT: prompts to abort the current SEND command.

See also: EXIT, QUIT

COPY

Usage: MINT>, READ>

The COPY command appends the contents of a range of messages to a file. The file can then be used as a auxiliary mail file, printed or edited by the user.

The COPY command has one parameter and three types of qualifiers.

Format:
COPY filename [/FORMAT=(MAIL,ASCII)] [/NEWVERSION] /[range qualifier]

If the /FORMAT qualifier is omitted, the default file format is MAIL.

Only one range qualifier can be given per command. One of the following range qualifiers is required:

/NEW, /READ, /UNSEEN, /SPECIAL, /ANSWERED, /DELETED, /BEFORE=date, /SINCE=date, /FROM=user, /TEXT=text, /SUBJECT=text, /SEQUENCE=#, /LAST=#, /ALL, /TO=user

Example:

```
COPY NEWMAIL.TXT /ALL
```

Dates

Usage: Concept

Several commands such as /BEFORE and /SINCE require a date in standard VAX/VMS format. The format is dd-mmm-yyyy (or d-mmm-yyyy for single digit dates) as in 12-JAN-1989 and 9-jul-1989.

Months are abbreviated to the first three characters and may be in either upper or lower case. You can also use the keywords TODAY, TOMORROW, and YESTERDAY.

DCL — Usage: MINT>, READ>, SEND>

The DCL command executes the specified DCL command or if no command is given it spawns a new subprocess. DCL stands for Digital Command Language and usually refers to commands typed at your VAX/VMS prompt.

This command is equivalent to SPAWN.

Format: DCL [command]
Example:

```
DCL TYPE MESSAGE.TXT
```

DEFAULT_BCC_LIST — Usage: Parameter

The DEFAULT_BCC_LIST parameter is used to specify blind carbon copy addresses to be used each time mail is sent.

There is no default for the parameter. Note also that addresses are not validated until mail is actually sent.

Example:

```
SET DEFAULT_BCC_LIST=BRUBBLE,FFLINSTONE,CALSO
```

See also:
DEFAULT_CC_LIST, CARBON_COPY, BLIND_CARBON_COPY

DEFAULT_CC_LIST — Usage: Parameter

The DEFAULT_CC_LIST parameter is used to specify carbon copy addresses to be used each time mail is sent.

There is no default for the parameter. Note also that addresses are not validated until mail is actually sent.
Example:

```
SET DEFAULT_CC_LIST=MMOUSE,DDUCK,BCLOWN
```

See also:
DEFAULT_BCC_LIST, CARBON_COPY, BLIND_CARBON_COPY

DELETE — Usage: MINT>, READ>

The DELETE command marks the specified messages deleted. The messages are then removed from the file with the EXPUNGE command, or when you use the EXIT or CTRL/Z command at MINT top-level.

Format: DELETE /range_qualifier

A range qualifier is required at MINT top-level; at the READ> prompt, the range defaults to the current message.
Example:

```
DELETE /ANSWERED
```

See also: UNDELETE, EXPUNGE

/DELETED Usage: Qualifier

The /DELETED range qualifier specifies all deleted messages.
Example:

```
READ /DELETED
```

See also: UNDELETE, DELETE

DIRECTORY Usage: MINT>, READ>

The directory command displays information from the message header lines. The display will have the format:

```
flags  msg #  date  from_address    subject         size
```

For example:

```
RS    2)   4-DEC  FRED@LBL.BITNET  ICPEAC:NYC Library (251)
```

If you send yourself a copy of a messsage, the to_address will replace the from_address.

Format: DIRECTORY/range_qualifier

The default range qualifier is /ALL

Example:

```
DIRECTORY  /FROM=JCOOL
```

See also: range qualifiers

DONT_DSPLY_HEADER Usage: Parameter

The DONT_DSPLY_HEADER parameter is used to filter out mail header lines from being displayed when reading mail. Header lines containing any of the contents of the list given will not be displayed. Thus, in the example below, header lines containing "TO" and "FROM" will not be displayed.

There is no default for the parameter.
Example:

```
SET DONT_DSPLY_HEADER=TO,FROM
```

Dynamic Environment Usage: Concept

A dynamic program environment is one that can be changed while a program is being run. A dynamic environment is achieved in MINT by using the SET command which allows control over MINT options.

See also: WRITE, SET, SHOW

/EVERYONE Usage: Qualifier

The /EVERYONE qualifier is used with the REPLY command to send a reply to all addresses in the message that are on the TO header and CC header as well as the address on the FROM header.
Example:

```
REPLY /EVERYONE /MESSAGE=21
```

EXIT — Usage: MINT>, READ>, SEND>

At MINT top-level, the EXIT command exits MINT, expunging messages that are marked deleted in your current mail file. Also at this time, if you have given the PRINT command, the file MAIL-TO-BE-PRINTED.TXT is queued to the printer.

If you are at the SEND> or READ> prompt, EXIT will move to MINT top-level and you will see the prompt: MINT>

Example:

```
EXIT
```

See also: QUIT

EXPUNGE — Usage: MINT>

The EXPUNGE command expunges your current mail file of any messages marked deleted. By default an expunge command is executed when the EXIT or CTRL/Z command is given at the top level of MINT. This can be turned on and off using the parameter EXPUNGE_ON_EXIT.

Example:

```
EXPUNGE
```

See also: DELETE, EXPUNGE_ON_EXIT

EXPUNGE_ON_EXIT — Usage: Parameter

The EXPUNGE_ON_EXIT parameter determines if messages marked deleted are expunged when the EXIT command is used.

The parameter accepts either the keyword ON or OFF. The default is set to ON.

Example:

```
SET EXPUNGE_ON_EXIT=ON
```

See also: EXPUNGE

File Format — Usage: Concept

File format, in the MINT context, refers to whether a file is created in standard VAX character format called ASCII or in MINT message format called MAIL. Some MINT commands such as COPY and MOVE allow you to specify the type of output file by using the /FORMAT qualifier. Files in ASCII format can be used for purposes unrelated to the mail system; for example as data files. Files in MAIL format are used to store messages that you will use with MINT commands such as GET and RESTORE.

See also: /FORMAT, GET, RESTORE, MOVE, COPY

/FLAG — Usage: Qualifier

The /FLAG qualifier is used with the MARK and UNMARK commands to add or remove a flag associated with a message.

The flags are typically used as follows:

The SPECIAL flag is used for messages of special importance.

The NEW flag is used for messages that have been delivered since the user last read his mail.

The READ flag is used to indicate which messages have been read.

The UNSEEN flag is used to indicate which messages have not been read and were delivered prior to the user last reading his mail.

The ANSWERED flag is used to indicate whether or not you have replied to a message.

The DELETED flag indicates which messages have been marked for deletion using the DELETE command.

Example:

```
MARK/FLAG=SPECIAL
```

See also: message flags, MARK, UNMARK

/FORMAT — Usage: Qualifier

The /FORMAT qualifier is used with the MOVE and COPY commands to specify file format. Possible values include ASCII for standard character format and MAIL for MINT message format. If the /FORMAT qualifier is omitted, MAIL format is used.

Example:

```
COPY filename/FORMAT=ASCII
```

See also: File format, MOVE, COPY

FORWARD — Usage: MINT>, READ>

The FORWARD command forwards a message to one or more specified addresses. This is similar to the REMAIL command except it allows you to insert text at the top of the message. You are placed in composition mode after responding to the TO and SUBJECT prompts and the text you then type is inserted in the message just above of the message header of the message you are forwarding.

Example:

```
FORWARD/MESSAGE=1
```

See also: REMAIL

/FROM — Usage: Qualifier

The /FROM range qualifier specifies all messages sent from the specified user.

To specify more than one address, enclose the list in parentheses and separate addresses with commas.
Example:

```
COPY BILL.TXT/FROM=WGIBBS
```

See also: /SUBJECT, /TEXT

GET Usage: MINT>

The GET command makes the specified file the current mail file. MINT.TXT is the default mail storage file, but additional files in MINT mail format can be created using the COPY and MOVE commands. Also, if the SAVE_MAIL_SENT parameter is set to ON, a mail file called MAIL.SENT is created that contains copies of all messages you send.

If you specify GET without a file name, SYS$LOGIN:MINT.TXT will again become the current mail file. (SYS$LOGIN is a logical name that refers to the home directory of your home account; i.e. your default directory when you first login.)

Format: GET [filename]

If the mail file is not located in your default disk and/or directory, you must also include disk and/or directory names, or you may include a logical name as in the following example:

Example:

```
GET SYS$LOGIN:GROUP.TXT
```

See also: COPY, MOVE

Header Lines Usage: Concept

The lines of information in a message that are displayed above the body of the message are called header lines.

Example:

```
Date: 31 Oct 1988 13:29:30 EST
From: Douglas Chan <DCHAN@EAGLE.WESLEYAN.EDU>
To: JSMITH@EAGLE.WESLEYAN.EDU
CC: jdecker@EAGLE.WESLEYAN.EDU
Subject: Re: internet mail
```

IBM Compatible PCs Usage: Concept

MINT can be run from IBM compatible PC’s using the KERMIT communication program or any VT100/VT200 terminal emulation program. You can obtain a copy from the Computing Center Office, room 501, Science Center. Using this distribution of KERMIT, the keys F1, F2, F3, F4 are used in place of the VT100/VT200+ keys PF1, PF2, PF3, and PF4.

See also: Terminal Control

/IMMEDIATELY Usage: Qualifier

The /IMMEDIATELY qualifier executes the print command when the print command is issued. All queued messages (previous and current) are sent to the specified printer.

Example:

```
PRINT /IMMEDIATELY 34,55,89,144
```

See also: PRINT

INCLUDE_SIGNATURE_FILE Usage: Parameter

The INCLUDE_SIGNATURE_FILE parameter determines if a signature file is appended to mail being sent. The file name is specified by the parameter SIGNATURE_FILE.

The parameter accepts one of the following keywords: ON, OFF or PROMPT. The default is set to OFF.

Example:

```
SET INCLUDE_SIGNATURE_FILE=ON
```

See also: SIGNATURE_FILE

KILL Usage: READ>

The kill command marks the current message deleted and displays the next message in the current range.

Example:

```
KILL
```

See also: DELETE, EXPUNGE

/LAST Usage: Qualifier

The /LAST range qualifier specifies the last n messages in the file.

Format: /LAST=n

Default: /LAST=1
Example:

```
READ /LAST=5
```

LMS_FILE Usage: Parameter

The LMS_FILE parameter is used to specify the file name that contains the last message sent. This parameter is used in conjunction with the parameter SAVE_LAST_MESSAGE_SENT.

The default is SYS$LOGIN:MINT.CPY.

Note: The device, directory, and full filename (or logical name) should be specified.

Example:

```
SET LMS_FILE=UD:[RLANG.EFOX]LAST_FILE.CPY
```

See also: SMS_FILE, SAVE_LAST_MESSAGE_SENT

LOGOUT Usage: MINT>, READ>, SEND>

The LOGOUT command logs you off the system.

Example:

```
LOGOUT
```

LREREAD Usage: READ>

The LREREAD command displays the current message, including header lines, ignoring the setting of the DONT_DISPLAY_HEADERS variable. It is equivalent to LTYPE.

Example:

```
LREREAD
```

See also: TYPE, LTYPE, PREVIOUS

LTYPE Usage: READ>

The LTYPE command displays the current message ignoring the setting of the DONT_DISPLAY_HEADERS variable. It is equivalent to LREREAD.

Example:

```
LTYPE
```

See also: TYPE, LREREAD, PREVIOUS

Macintosh PCs Usage: Concept

MINT can be run from a Macintosh using the KERMIT communication program. You can obtain a copy from the Computing Center Office, room 501, Science Center. Using this distribution of KERMIT, the keypad keys clear, =, /, * are used in place of the VT100/VT200+ keys PF1, PF2, PF3, and PF4.

See also: Terminal Control

MAILING_LIST Usage: Parameter

The MAILING_LIST parameter is used to specify the name of the indexed file which is to be used as your personal mailing list file. For more information on how to create this file see the topic: *Creating a Mailing List File* in section 5, page 18, or type HELP MLIST at your VAX/VMS prompt.

The default is SYS$LOGIN:MAILING-LIST.IDX

Example:

```
SET MAILING_LIST=SYS$LOGIN:PEOPLE_I_KNOW.IDX
```

See also: Mailing lists

Mailing Lists Usage: Concept

Two mailing lists can be used when sending mail. One mailing list is maintained by the system postmaster and the other mailing list can be created by you. A mailing list makes it easier to send mail to users with long addresses and to groups of users. For more information on how to create this file see the topic: *Creating a Mailing List File* in section 5, page 18, and also look at the file SYS$MINT:MAILING-LIST.TXT

See also: MAILING_LIST parameter

Note: Special interest groups can be set up by contacting the postmaster; send mail TO:POSTMASTER.

MARK

Usage: READ>

The MARK command marks the specified messages with the specified flag. Marking messages is one way to categorize messages. Messages marked new and special, for example, are displayed by default on MINT startup.

Format:
MARK [/FLAG=SPECIAL,NEW,READ,UNSEEN] [/range qualifier]

Example:

```
MARK /FLAG=SPECIAL 4,9,16,25,36
```

See also: UNMARK

/MESSAGE

Usage: Qualifier

The /MESSAGE qualifier specifies the message that you want to forward, reply to, or remail. This qualifier is required when you use FORWARD, REPLY, or REMAIL from MINT top-level.

Note that /MESSAGE is not a range qualifier -- it is used to specify only one message, rather than a group of messages.
Example:

```
FORWARD /MESSAGE=49
```

See also: FORWARD, REMAIL, REPLY

MINT

Usage: DCL command, Mail system

The MINT (Mail INTerface) program is the user interface to the mail system.

Example:

```
MINT
```

MOVE

Usage: MINT>, READ>

The MOVE command copies the contents of a range of messages to a file and then marks those messages deleted. The file can then be used as an auxiliary mail file, printed or edited by the user. Note that when you EXIT from the MINT top-level the messages marked as deleted by the MOVE command will be expunged.

The MOVE command has one parameter (filename) and three types of qualifiers.

Format:

MOVE filename [/FORMAT=(MAIL,ASCII)] [/NEWVERSION] /[range qualifier]

The /FORMAT qualifier specifies whether messages will be stored in MAIL or ASCII format. If the file is not to be used with MINT, you should most likely specify /FORMAT=ASCII. If you omit the qualifier, MAIL format is used and the file is not printable.

The /NEWVERSION parameter indicates that a new file should be created. If this is omitted, the messages will be appended to the end of the file you specify.

The *range qualifier* specifies which messages should be moved. It can be omitted at read-level, in which case the current message will be moved. Only one range qualifier can be given per command.

Example:

```
MOVE LOOKIE_HERE.TXT /FORMAT=ASCII /FROM=SCLAUS
```

See also: COPY, File Format

MLIST Usage: VAX/VMS

MLIST is a program used to convert a mailing list file in ASCII format, MAILING-LIST.TXT, into an indexed file in MINT format, MAILING-LIST.IDX. You can run MLIST on a file other than MAILING-LIST.TXT by specifying the /FILE qualifier; e.g., after the VAX/VMS prompt type:

```
MLIST/FILE=EMAIL-ADDRESSES.TXT
```

The MINT parameter MAILING-LIST could then be used to select a mailing list name other than MAILING-LIST.IDX; for example, to correspond with the above example, type:

```
SET MAILING-LIST=EMAIL-ADDRESSES.IDX
```

For more information on MLIST, type MLIST at the VAX/VMS prompt, or read the section in this chapter titled: CREATING A MAILING LIST (Section 5, page 18).
See also: MAILING LISTS

Message Flags Usage: Concept

Messages can be marked with message flags. These flags are used by some of the commands to specify a range of messages.

The message flags include:
N for NEW
R for READ
U for UNSEEN
S for SPECIAL
A for ANSWERED
D for DELETED

N, R, U and A are set by the mail system and S and D are set by the user.

See also: Header lines, ranges, /FLAG, MARK, UNMARK

/NEW Usage: Qualifier

The /NEW range qualifier specifies all new messages

Example:

```
DELETE /NEW
```

See also: /SINCE, /UNSEEN

/NEWVERSION — Usage: Qualifier

The optional /NEWVERSION qualifier is used with the MOVE and COPY commands to specify that a new version of a file is to be created to contain messages being moved or copied. When this qualifier is omitted, messages are appended to the end of the file specified on the MOVE or COPY command.

Example:

```
MOVE RECORDS.TXT/FORMAT=ASCII/NEWVERSION/FROM=FBAGGINS
```

See also: MOVE, COPY

New Features — Usage: Concept

New features will be added to the MINT system while it is under development. Use the MINT HELP command to see a list of new features.

Example: HELP NEW_FEATURES

NEXT — Usage: READ>

The NEXT command is used at read-level to display the next message in the current range. It is equivalent to pressing RETURN.

Example:

```
NEXT
```

See also: PREVIOUS, TYPE

PAGING — Usage: Parameter

The PAGING parameter is used to have the terminal automatically pause when the screen becomes full. After pausing, the message:

```
Press [return] to continue or Q [return] to Quit:
```

is displayed at the bottom of the screen. You can press RETURN to continue viewing the message or Q to quit and return to the READ> prompt.

Ctrl/S-Ctrl/Q and hold screen provide an alternate way of pausing the screen to read a message longer than one page.

The parameter accepts the keywords ON and OFF. The default value is OFF; in other words, if you do not set this parameter ON, the screen will not automatically pause when full.

Example:

```
SET PAGING=ON
```

See also: SET

Parameter — Usage: Concept

Parameters are keywords which describe the MINT environment. You can change the values of parameters by using the SET command, show current settings by using the SHOW command, and save all values you've set by using the WRITE command.

Examples:

```
SET PERSONAL_NAME=Alfred Aho
SHOW PERSONAL_NAME
WRITE
```

See also: SET, WRITE, SHOW, dynamic environment

PERSONAL_NAME Usage: Parameter

The PERSONAL NAME parameter is used to specify a name other than your computer address that will be included on the "from" line of messages you send.

Note that the personal name should not contain a parenthesis as this character has a specific meaning in mail addresses.

Example:

```
SET PERSONAL_NAME=HAL9000
```

The corresponding "from" line in messages sent might be:

```
From: HAL9000 <JSMITH@EAGLE.WESLEYAN.EDU>
```

See also: SET

Pop-Down Menu Usage: Concept

Pop-down menus are a feature of many popular microcomputer programs. They usually appear on a portion of the screen as an overlay, and can be "popped off" to restore the previous contents of the screen.
In MINT, pressing the CTRL/H key toggles a screen that lists the commands currently available.

See also: Control/H

PREVIOUS Usage: READ>

The PREVIOUS command is used at read-level to display the previous message in the current range.

Example:

```
PREVIOUS
```

See also: NEXT, REREAD

PRINT Usage: MINT>, READ>

The PRINT command writes out the specified messages to a temporary file in the directory pointed to by SYS$LOGIN:. The actual print command is executed upon exiting from the mail system. It can be executed immediately if the /IMMEDIATE qualifier is given.

The parameter PRINT_COMMAND can be used to specify a VAX/VMS printer command or to specify that your local printer should be used.
Format: PRINT [/IMMEDIATELY] /range qualifier

Example:

```
PRINT /IMMEDIATELY /FROM=GJETSON
```

See also: PRINT_COMMAND, COPY

PRINT_COMMAND — Usage: Parameter

The PRINT_COMMAND parameter determines whether a local or system printer will be used, and if the system printer is to be used, it determines the command that will be used to actually queue the file.

The keyword LOCAL directs the messages being printed to a local printer connected to the terminal; for example:

```
SET PRINT_COMMAND=LOCAL
```

The default setting is: `print/delete/queue=sys$print.`

If you define a new value (other than LOCAL), include the qualifier /DELETE, to have the file automatically deleted after it is printed; for example:

```
SET PRINT=PRINT/FORM=51/DELETE
```

See also: PRINT

Qualifiers — Usage: Concept

Qualifiers are key words that are used with MINT commands to supply additional information about how a command is to be executed. These rules apply to qualifiers and values of qualifiers:

- Qualifier keywords follow the command name and/or command parameter and are preceded by a slash (/).
- The qualifier keyword is followed by an equal sign (=) if a value is to be given.
- The value of a qualifer may be one item or a list of items enclosed in parentheses and separated by commas.
- If the value of a qualifier consists of more than one word or contains non-alphanumeric characters (e.g. +,=,$), it should be enclosed in quotes.
- Only one qualifier that specifies a range of messages may be used with a command. This type of qualifier is called a ***range qualifier***.
- Only one range qualifier may be used per command.

Examples:

```
READ/SPECIAL
REPLY /MESS=14
MARK 16/FLAG=SPECIAL
MARK/FLAG=SPECIAL 16
MOVE special.mail /special
DIR/SUBJECT="planning meeting"
ADD/TO=(MNORUSIS,NNIE)
COPY/FORMAT=ASCII/SPECIAL
```

See also: Range qualifiers

QUIT

Usage: MINT>, READ>, SEND>

When used from the read or send-level, the QUIT command exits to the top-level. From the top-level, the QUIT command exits the program queuing any messages to be printed.

There is one major difference between EXIT and QUIT: when you use EXIT at MINT top-level, deleted messages are expunged; when you QUIT, deleted messages are not expunged.

Example:

```
QUIT
```

See also: EXIT, EXPUNGE

Quotes

Usage: Concept

Quotes are used in typing MINT commands to maintain the case of a parameter or qualifier value. Quotes must also be used to specify a qualifier value that consists of more than one word or contains special characters such as "/" or "_".

Examples:

```
SEND/TO="JSmith"
DIR/TEXT="end of year"
```

See also: Qualifiers

Range

Usage: Concept

A range is a group of messages. A range can consist of one, some or all of the messages in the mail file. The range can be determined by message number, or some other attribute that messages have in common such as "all having been answered", "all new messages", "all messages from a particular user".
See also: Range Qualifiers

Range Qualifiers

Usage: Concept

A Range Qualifier is a MINT keyword that specifies a particular range of messages in the message file. Qualifiers are used with MINT commands to indicate which messages are to be acted on by the command. A list of range qualifiers follows:

/NEW, /READ, /UNSEEN, /SPECIAL, /ANSWERED, /DELETED, /BEFORE=date, /SINCE=date, /FROM=user, /TEXT=text, /SUBJECT=text, /SEQUENCE=#, /LAST=#, /ALL, /TO=user

Example:

```
PRINT/SINCE=1-DEC-1988
```

This specifies the range of messages received after Dec 1, 1988.

See also: Range

/READ — Usage: Qualifier

The /READ range qualifier specifies all messages that have been read.

Example:

```
DELETE /READ
```

Read-Level — Usage: Concept

Read-level is the level at which you read messages. It is indicated by the READ> prompt. Commands at this level include:

COPY	QUIT	EXIT	REFRESH	ctlW
ctlZ	ctlH	BLANK	DELETE	MARK
UNDELETE	UNMARK	LOGOUT	KILL	LTYPE
TYPE	NEXT	PREVIOUS	DCL	MOVE
WRITE	SET	SHOW	PRINT	SEND
REPLY	REMAIL	FORWARD	REREAD	LREREAD

See also: Send-level, Top-level

READ_PROMPT — Usage: Parameter

The READ_PROMPT parameter is used to modify the read-level prompt of the MINT program.

The default is READ>

Example:

```
SET READ_PROMPT="LookAtThis! "
```

See also: SEND_PROMPT, TOP_PROMPT

REFRESH — Usage: MINT>, READ>, SEND>

The REFRESH command refreshes the screen. It is useful if the screen has been garbled. CTRL/W is equivalent to REFRESH, but can be used even when a command prompt is not visible.

Example:

```
REFRESH
```

See also: Control/W, BLANK

REMAIL — Usage: MINT>, READ>, SEND>

The REMAIL command remails a message to one or more specified addresses. The REMAIL command is similar to the FORWARD command. The difference between the two is that the FORWARD command allows you to insert text at the top of the message.

Format: REMAIL /MESSAGE=(msg number) [/TO/CC/BCC/SUBJECT]

You will be prompted TO: or SUBJECT: if you omit these qualifiers.
Example:

```
REMAIL /MESSAGE=17/TO=JSMITH
```

See also: FORWARD

REMOVE Usage: SEND>

The REMOVE command allows you to remove recipients from the message being sent.

The qualifiers /TO, /CC, and /BCC are used to specify the address list from which the address is to be removed. If only one address exists on the "To:" line it can not be removed.

Note: Addresses used in this command are case sensitive. You must match the case exactly, and you must enclose the address in quotes if it contains lower case letters.

Format: REMOVE [/TO=address /CC=address /BCC=address]

Example:

```
REMOVE /TO=CBROWN /CC="JBrown"
```

See also: ADD

REPLY Usage: READ>, MINT>

The REPLY command sends a reply to the specified message. The default reply goes to the address on the FROM header line in the current message. The qualifiers /EVERYONE, /TO, /CC, and /BCC can be used to send the reply to other addresses.

A subject can be specified by including the /SUBJECT parameter; if not specified, the subject from the current message will be used preceded by "RE:".

A message number can be specified using the qualifier /MESSAGE. This is required at MINT top-level.

Format:
REPLY [/EVERYONE/TO/CC/BCC/SUBJECT] /MESSAGE=number

Example:

```
REPLY /EVERYONE/MESSAGE=7
```

See also: SEND

REPLY_TO Usage: Parameter

The REPLY_TO parameter is used to add a "reply to" message header line to the message being sent to inform the recipient of the message to reply to that address.

Example:

```
SET REPLY_TO=BCLOWN@EAGLE.WESLEYAN.EDU
```

The corresponding message header line would be:

```
Reply to: BCLOWN@EAGLE.WESLEYAN.EDU
```

REREAD Usage: READ>

The REREAD command is used at read-level to redisplay the current message. Any header lines containing strings that have been excluded using DONT_DSPLY_HEADER will not be displayed. It is equivalent to the TYPE command.
Example:

```
REREAD
```

See also: PREVIOUS, LTYPE, NEXT, DONT_DSPLY_HEADER, Header lines

---

RESTORE Usage: MINT>

The RESTORE command restores a previously sent or saved message and places you into message composition mode. The command accepts a filename, but defaults to MAIL.CPY (a copy of the last message sent).

Format: RESTORE [filename]

Example:

```
RESTORE LastMsg.TXT
```

See also: SAVE, LMS_FILE

SAVE Usage: SEND>

The SAVE command allows you to save the message being composed. This command is useful if you are interrupted when composing a message. The command saves information entered so far such as "to" and "subject" lines and whatever text you have written thus far. The RESTORE command is used to resume from the point you issued the SAVE. A filename must be specified; for example:

```
SAVE NewMail.TXT
```

See also: RESTORE

SAVE_LAST_MESSAGE_SENT Usage: Parameter

The SAVE_LAST_MESSAGE_SENT parameter determines if a copy of the last message sent is saved to a file. The default file name is MAIL.CPY, but another name can be specified by using the parameter LMS_FILE.

The SAVE_LAST_MESSAGE_SENT parameter accepts either the keyword ON or OFF. The default is set to ON.
Example:

```
SET SAVE_LAST_MESSAGE_SENT=OFF
```

See also: LMS_FILE, SAVE_MAIL_SENT, SMS_FILE

Note: The file is saved in MAIL format; therefore, you can resend this message by using the RESTORE command.

SAVE_MAIL_SENT Usage: Parameter

The SAVE_MAIL_SENT parameter determines if all mail sent is saved to a file. The file name is specified by the parameter SMS_FILE. The parameter accepts one of the following keywords: ON, OFF or PROMPT. The default is set to OFF.
Examples:

```
SET SAVE_MAIL_SENT=ON
SET SMS_FILE=UD:[PHYS.AJP]MAIL-SENT.TXT
```

See also: SET, SHOW, SMS_FILE

Screen Oriented Usage: Concept

"Screen oriented" implies that a program will use cursor positioning features of a terminal in drawing menus and other displays.

MINT is screen oriented. The screen is divided into four windows. The top window is used to display the output of commands. The next window down is the status window and contains current file, message, and range information. The third window is used for command prompts. The bottom window is used to display information such as error and success messages.

Note: MINT screen displays are supported for VT100, VT200+ video terminals, or microcomputers running emulation programs for VT100 and VT200 terminals. If you wish to run MINT from another terminal, for example, a hardcopy terminal, you can do so by issuing the following VAX/VMS command before you invoke MINT:

```
SET TERMINAL/NOANSI
```

See also: Terminal Support

SEND Usage: MINT>, SEND>

The SEND command is used in two different ways. At the top-level of MINT, the SEND command sends a message to one or more specified addresses. When you type SEND at the MINT> prompt, you are then prompted TO: to enter one or more addresses, and SUBJECT: to enter a subject for the message. Alternatively, you may include one or more of the qualifiers /TO, /CC, /BCC, and /SUBJECT, as in the following example:

```
SEND /TO=UYUTZ/CC=BTCLOWN
```

At the send-level, the SEND command dispatches a message previously created. Typing SEND at the SEND> prompt is equivalent to pressing PF4 while in message composition mode.

See also: Send-level, /TO, /CC, /BCC, /SUBJECT

Send Level Usage: Concept

The MINT program consists of three levels: top, read, and send. Messages are composed and mailed from the send-level.

See also: Top-level,Read-level

SEND_PROMPT Usage: Parameter

The SEND PROMPT parameter is used to modify the send level prompt of the MINT program. The default is "SEND> ".
Example:

```
SET SEND_PROMPT=SEND_IT_OUT>
```

See also: SET, SHOW

/SEQUENCE Usage: Qualifier

The /SEQUENCE range qualifier specifies all messages in the specified sequence.

A sequence can be a single number, a list of numbers separated by commas, a range of numbers specified by a colon, or a combination of a list and range of numbers.

For example, the sequence `1,3,5:10,12` indicates the messages one, three, five through ten, and twelve.
Example:

```
READ/SEQ=1,4,7:10
```

This is equivalent to:

```
READ 1,4,7:10
```

See also: Range qualifiers

SET Usage: MINT>

The SET command is used to set or change the MINT environment. There are 20 possible parameters that can be set. Using the SET command changes the environment only while MINT is currently being executed. The WRITE command can be used to save the current environment to an initialization file (MINT.INIT) which will be read automatically when MINT is invoked.
Example:

```
SET PERSONAL_NAME=Adolph
```

See also: SHOW, WRITE

SHOW Usage: MINT>

The SHOW command is used to show the current MINT environment. The show command has 20 parameters, corresponding to the 20 parameters that can be controlled with the SET command.

Example:

```
SHOW PERSONAL_NAME
```

See also: SET, WRITE

Signature Usage: Concept

In the MINT context, a signature is text which should be automatically appended to mail being sent. This could be information such as your name and address. By default the name of the signature file is MINT.SIGNATURE, but this can be changed via the parameter SIGNATURE_FILE. The parameter INCLUDE_SIGNATURE_FILE determines whether or not the signature file is currently active.

See also: SIGNATURE_FILE, INCLUDE_SIGNATURE_FILE

SIGNATURE_FILE Usage: Parameter

The SIGNATURE_FILE parameter is used to specify the file name that contains the information to be appended to mail being sent. This parameter is used in conjunction with the parameter INCLUDE_SIGNATURE_FILE.

The default is SYS$LOGIN:MINT.SIGNATURE.

Note: The device, directory, and full filename should be specified; for example:

```
SET SIG=UD:[AST.KEPLER]SIG.TXT
```

See also: SIGNATURE, INCLUDE_SIGNATURE_FILE

/SINCE Usage: Qualifier

The /SINCE range qualifier specifies all messages delivered since the specified date

Format: /SINCE=date
Date may be one of the following TODAY, YESTERDAY, or any valid VMS date specification (e.g. 13-sep-1988).

Example:

```
READ /SINCE=YESTERDAY
```

See also: Dates

SMS_FILE Usage: Parameter

The SMS_FILE parameter is used to specify the file name that contains a copy of all messages sent. This parameter is used in conjunction with the parameter SAVE_MAIL_SENT.

The default is SYS$LOGIN:MINT.SENT.

Note: The device, directory, and full filename should be specified.

Example:

```
SET SMS_FILE=UD:[BIO.JSMITH]MINT.SENT-BIO
```

See also: SAVE_MAIL_SENT

/SPECIAL Usage: Qualifier

The /SPECIAL range qualifier specifies all special messages. The MARK command is used to classify messages as special. Messages which have been marked SPECIAL are listed with any new messages each time you invoke MINT.

Example:

```
PRINT /SPECIAL
```

See also: MARK, UNMARK

SPELL — Usage: SEND>

The SPELL command runs the SPELL program to check/correct the spelling of the message being sent. For more information on SPELL type: `HELP SPELL` at the VAX/VMS prompt.

Example:

```
SPELL
```

STATUS — Usage: MINT>

The STATUS command provides information (status) on the current mail file. The information includes the file name, file size, and the number of messages in the file.

Example:

```
STATUS
```

See also: SHOW

/SUBJECT — Usage: Qualifier

The /SUBJECT range qualifier specifies all messages with the subject header containing the specified text

Format: /SUBJECT=text

Example:

```
READ /SUBJECT="Terrorist Squirrels"
```

See also: /TEXT

/TEXT — Usage: Qualifier

The /TEXT range qualifier specifies all messages containing the specified text

Format:
/TEXT=text
or
/TEXT="multi-word text"
Examples:

```
DIRECTORY /TEXT=squirrel
DIRECTORY /TEXT="killed him with an axe"
```

See also: /SUBJECT, /FROM

Terminal Support — Usage: Concept

MINT screen displays are supported for VT100, VT200+ video terminals, or microcomputers running emulation programs for VT100 and VT200 terminals. If you wish to run MINT from another terminal, for example, a hardcopy terminal, you can do so by issuing the following VAX/VMS command before you invoke MINT:

```
SET TERMINAL/NOANSI
```

The status line will not be displayed, but all commands should work except for the following:

BLANK command
REFRESH command
Control/ H (command list display)
Control/ W (refresh)
Control/ Z (abort)
SPELL command
EDIT command

A serious limitation of this mode is that message composition allows editing of the current line only; that is, you cannot edit a line once you have pressed RETURN. *You can improve on this situation by setting the parameter USE_EDITOR_AUTOMATICALLY to ON, and defining the symbol EDIT to be a non-screen oriented editor; for example:*

```
EDIT=="EDIT/EDT"
```

See also: IBM Compatible, Macintosh,USE_EDITOR_AUTOMATICALLY

/TO Usage: Qualifier

The /TO range qualifier specifies all messages sent to the specified user.

To specify more than one address, enclose the list in parentheses and separate addresses with commas.

Examples:

```
DELETE /TO=BTAN
DELETE /TO=(BTAN,CTAN)
```

See also: /FROM

Top Level Usage: Concept

The MINT program consists of three levels: top, read, and send. The top-level provides an avenue to the other levels, and handles many general commands such as DIRECTORY.

See also: Read-level, Send-level

TOP_PROMPT Usage: Parameter

The TOP_PROMPT parameter is used to modify the top level prompt of the MINT program.
Example:

```
SET TOP_LEVEL_PROMPT=MY_MAIL>
```

See also: READ_PROMPT, SEND_PROMPT

TYPE Usage: READ>

The TYPE command is used at read-level to redisplay the current message. Any header lines containing words (strings) that have been excluded using DONT_DSPLY_HEADER will not be displayed.

Example:

```
TYPE
```

See also:
PREVIOUS, LTYPE, NEXT, DONT_DSPLY_HEADER, REREAD

UNDELETE Usage: MINT>

The UNDELETE command removes the "deleted" flag from the specified messages. The UNDELETE command is useful if you accidently marked a message for deletion using the DELETE command.

Example:

```
UNDELETE/ALL
```

See also: DELETE

UNMARK Usage: MINT>

The UNMARK command is used to remove message flags associated with a message or range of messages. The /FLAG qualifier is required to specify which flag should be removed.

Example:

```
UNMARK /FLAG=SPECIAL/FROM=GKHAN
```

UNMARK is used in this example to remove the SPECIAL flag from messages received from GKHAN. These messages will no longer be listed with new messages when MINT is invoked.

/UNSEEN Usage: Qualifier

The /UNSEEN range qualifier specifies that a given command be performed on all unseen messages; i.e.,all messages not yet read.

Example:

```
PRINT/UNSEEN
```

See also: NEW

USE_EDITOR_AUTOMATICALLY Usage: Parameter

The USE EDITOR AUTOMATICALLY parameter determines if the VAX editor is automatically invoked when using the SEND command at the top-level. The parameter accepts either the keyword ON or OFF. The default is set to OFF; in other words, if you do not change this parameter you will be using the MINT editor rather than the standard VAX editor when you compose messages.

Example:

```
SET USE_EDITOR_AUTOMATICALLY=ON
```

Note: The editor used is determined by the VAX/VMS symbol EDIT. By redefining EDIT you can change the editor that will be invoked. Symbol definitions, such as the following example, are typed at the VAX/VMS prompt or included in a LOGIN.COM file:

```
EDIT==EDIT/EDT
```

WRITE — Usage: MINT>

The WRITE command writes out the current environment settings to the file MINT.INIT into the home directory of your home account (your login directory). The file MINT.INIT is read automatically when MINT is started to set up the MINT environment.
Example:

```
WRITE
```

See also: SET, SHOW, dynamic environment

Index