

A USER'S GUIDE TO LAFITE

XEROX

THE INTERLISP MAIL SYSTEM

Copyright © 1986, Xerox Corporation
All rights reserved

Copyright protection claimed includes all forms and matters of
copyrightable material and information now allowed by statutory or
judicial law or hereinafter granted, including without limitation,
material generated from the software programs which are displayed
on the screen such as icons, screen display looks, etc.

1. Introduction	1
What Lafite Is	1
What This Manual Is About	1
2. Things to Know Before You Start	3
3. Obtaining Lafite	5
What Programs You Need	5
How to Load the Programs You Need	5
4. Entering Lafite	7
The Status Window	7
5. Reading Messages	9
The Browse Command	9
The Browser Window	9
The Get Mail Command	10
The Display Command and Selection Pointer	11
Text Scrolling	12
Text Thumbing	13
6. Deleting Messages and Updating the Mail Folder	15
The Delete Command	15
The Undelete Command	15
The Update Command	16
Other Ways to Update	16
7. Printing Messages	19
8. Filing Messages	21
The Move To Command	21
How to Create New Folders	22
How to View Other Folders	23
The Forget Folder Command	23
9. Writing Messages	25
The Message Composition Window	25
The Message Header	26
The Message Editor	31

The Type-in Point	31
Text Selection	32
Text Deletion	33
Text Moving	33
Text Copying	33
The Include Command	33
The Put Command	34
The Get Command	35
The Hardcopy Command	36
Fancy Message Formats	36
Messages With Bit Maps	36
Message Forms	38
The Standard Form	38
The Answer Form	38
The Forward Form	39
Specialized Report Forms	40
The Forget Message Form Command	42
10. Sending Messages	45
Delivery of Formatted Messages	46
11. Leaving Lafite	49
12. Troubleshooting Lafite Problems	51
Lafite Confuses You With a Previous User	51
File Server Is Slow	51
Mail File Does Not Parse	52
Table of Contents Is Inconsistent With Mail File	53
13. Customizing Lafite	55
Lafite's Global Variables	55
How to Add a New Message Form to Lafite	59
14. Using Lafite-Related Lisp Library Packages	63
Lafite Find	63
How to Load Lafite Find	63
Search Direction Commands	63
Search Type Commands	64
How to Use Lafite Find	65
Mail Scavenger	66
How to Use Mail Scavenger	66

Appendix A: Using Lafite Courteously	69
Foreward	69
Introduction	69
Communication Patterns	70
The Wrong Number	71
Rudeness and Vulgarity	73
Message System Costs	75
Unsolicited Mail	76
The Chain Reaction	77
Off-the-Record Responses	78
Masquerading	78
Wizards Versus Naive Users	79
The Moral of This Tale	79
Appendix B: Using Lafite at Xerox	81
The Grapevine Mail System	81
Grapevine Protocols	81
Grapevine Recipient Addresses	82
Maintain	85
If You're on Both Networks	89
Lafite Modes	89
From Grapevine to NS and Vice Versa	90
Lafite-Related Lisp Users' Packages	91
From Laurel to Lafite	92
The Browse Laurel File Command	92
Glossary of Technical Terms	95
Quick-Reference Section	99
Message Marks	99
Lafite Commands	99
Status Window Commands	99
Browser Window Commands	100
Message Composition Window Commands	101
Index	103

[This page intentionally left blank]

What Lafite Is

Lafite is an easy-to-use, display-oriented electronic message system. Like all electronic message systems, Lafite combines the advantages of the postal service and the telephone system. You can compose messages at leisure and in private, transmit them rapidly, and print copies of the messages you send and receive.

However, Lafite offers more than most electronic message systems. It uses the Interlisp-D window interface, which enables you to read and write many messages simultaneously while you are using other Interlisp systems. It is also fully integrated with the TEdit text editor, which allows you to write messages using all of TEdit's editing and formatting capabilities. In addition, Lafite enables you to create mail folders and file your messages in them and to create, save, and use special message forms.

What This Manual Is About

This manual is designed to help you use Lafite. It assumes you understand the basic principles of using your Xerox Lisp workstation and the Interlisp-D environment. If you need information on operating your workstation, refer to its user's manual. The *Friendly Dandelion Primer* and the *Interlisp-D Reference Manual* describe the Interlisp environment.

If you are a new Lafite user, you will find it helpful to skim through the manual, then return to the beginning and practice using Lafite's features in the order in which they are described. If you are an experienced Lafite user, you should just skim through the manual, then turn to the appropriate section whenever you need help. If you are a technically knowledgeable or adventurous user, you may want to know how to customize the system. Information on modifying Lafite's behavior is provided in chapter 13, "Customizing Lafite."

You can find some useful background information in chapter 2, "Things to Know Before You Start." Chapter 3, "Obtaining Lafite," tells you how to log in and how to load Lafite and the other software you need to use it. In chapter 4, "Entering Lafite," you can find out how to access Lafite and how to use the highest-level Lafite window, the status window. You can learn how to read your mail in chapter 5, "Reading Messages," and how to update your mail files in chapter 6, "Deleting Messages"

and Updating the Mail Folder." Chapter 7, "Printing Messages," tells you how to obtain hard copies of your messages, and chapter 8, "Filing Messages," tells you how to organize them. You can find out how to compose and transmit Lafite mail in chapters 9 and 10, "Writing Messages" and "Sending Messages." Chapter 11, "Leaving Lafite," describes several ways you can exit the system.

If Lafite does not seem to be operating correctly, you can turn to chapter 12, "Troubleshooting Lafite Problems." Chapter 13, "Customizing Lafite," tells you how to change Lafite's global variables (we have tried to make it accessible to nonprogrammers who are willing to do a little extra reading). And chapter 14, "Using Lafite-Related Lisp Library Packages," contains user's guides to the Lafite Find and Mail Scavenger Lisp Library packages.

You will find several different kinds of reference material at the end of this manual. Appendix A, "Using Lafite Courteously," is a guide to message system etiquette. You can look up technical terms used in the manual in the glossary at the end of appendix A. After appendix A, you will find a quick-reference section giving brief summaries of Lafite's commands and a chart of message marks. At the very end, there is an index that shows you where to find all the important information given in this manual.

Lafite is a display-based, interactive program that manipulates a particular class of files called *mail files*. In essence, a mail file is just a set of messages, each of which is a piece of text formatted according to certain conventions. The details of these conventions are not of major interest to most users. Suffice it to say that messages have a *header* and a *body*. The header identifies the *sender* and *recipients* of the message as well as the *time* it was sent, a *subject*, and perhaps other information not directly related to the message's content. The body contains the content of the message. For each mail file, Lafite constructs and maintains a *table of contents* that summarizes the messages residing in the file. Mail files can be stored either on a file server that allows random access or on your own workstation's disk.

Lafite is an interface program that gives you access to the facilities provided by the Xerox Network Systems (NS) Mail Service. This system provides you with *in-boxes*, *mail drops*, and a *name data base*. An in-box is a place where messages sent to you are stored until you retrieve them. Each in-box resides on a *mail server*, which is a machine used for storing and distributing electronic mail. A mail drop is a service to which messages addressed to any user(s) of the system may be sent. The mail transport system sorts out the recipients and delivers messages to their individual in-boxes. The name data base holds the names of *registered users* of the message system along with named lists of such users.

Lafite retrieves inbound mail from your in-boxes on one or more mail servers. The messages are retrieved into a *mail folder*, which is usually called Active.Mail (henceforth referred to as the Active Mail folder). You can then open a *browser* onto the mail folder, which allows its contents to be read, replied to, forwarded, deleted, moved into other folders, printed, and so forth.

[This page intentionally left blank]

What Programs You Need

Before you run Lafite, two Lisp Library programs must be loaded into your workstation's virtual memory. These are Lafite itself and NS Mail. NS Mail contains an implementation of the *protocols*, or information interchange standards, that enable Lafite to communicate with NS mail servers.

In addition, you may want to load two optional but useful programs, Lafite Find and Mail Scavenger. Lafite Find facilitates searching for messages contained in a mail folder. Mail Scavenger helps to restore the internal structure of a mail file in the unlikely event that it becomes damaged.

How to Load the Programs You Need

To load Lafite or another program from a floppy disk, type (LOAD '{FLOPPY}FILENAME.DCOM) at the prompt in your Interlisp-D Executive window. However, if a program is frequently used in your company it is probably stored on a file server rather than on a floppy disk. The following instructions apply to loading packages from a file server, with the words *FILESERVER* and *DIRECTORY* standing for the names of the actual file server and directory on which these programs are stored.

First, you must log into the system. To log in, type (LOGIN) at the prompt in the executive window (see figure 1). On the next line, the system will print the word Login, followed by a colon. Type your log-in name after the colon and press the carriage return. On the same line, the system will prompt you for your password. Type in your password (each letter will appear as an asterisk on the screen) and press the carriage return again. The system will print your name as a confirmation of your log-in.

```

Interlisp-D Executive
J
NIL
5>(LOGIN)
Login: WEISBLATT (password) *****
WEISBLATT
6>(LOAD '{ERIS}\LISP\KOTO\LIBRARY\LAFITE.DCOM)
{ERIS}\LISP\KOTO\LIBRARY\LAFITE.DCOM;1
compiled on 12-Mar-85 03:14:25
FILE CREATED 12-Mar-85 03:12:23
LAFITE.DCOMS

{ERIS}\LISP\KOTO\LIBRARY\LAFITEBROWSE.DCOM;1
compiled on 12-Mar-85 00:56:03
FILE CREATED 12-Mar-85 00:52:59
LAFITEBROWSE.DCOMS

{ERIS}\LISP\KOTO\LIBRARY\LAFITESEND.DCOM;1
compiled on 12-Mar-85 00:56:30
FILE CREATED 12-Mar-85 00:51:52
LAFITESEND.DCOMS

{ERIS}\LISP\KOTO\LIBRARY\LAFITEMAIL.DCOM;1
compiled on 12-Mar-85 00:57:43
FILE CREATED 12-Mar-85 00:52:27
LAFITEMAIL.DCOMS
{ERIS}\LISP\KOTO\LIBRARY\LAFITE.DCOM;1
7>

```

Figure 1. The executive window. The user has logged in and loaded Lafite

You can now load Lafite. Type (LOAD '{FILESERVER} <DIRECTORY>LAFITE.DCOM) at the prompt. (If the DIRECTORIES variable in your initialization file contains a search path to the directory on which Lafite is stored, you can just type (LOAD 'LAFITE.DCOM)). The system will print the file's full name, then give the date the file was created. When it is finished loading the file, it will again print the file name and return you to the top-level prompt.

To load NS Mail, type (LOAD '{FILESERVER}<DIRECTORY> NSMAIL.DCOM).

To load Lafite Find, type (LOAD '{FILESERVER}<DIRECTORY> LAFITEFIND.DCOM).

Finally, to load Mail Scavenger, type (LOAD '{FILESERVER} <DIRECTORY>MAILSCAVENGE.DCOM).

You must log into the network before you can use Lafite. If you have not logged in already, do so now, following the instructions given in chapter 3.

In order to obtain your mail, you must be an *authenticated user*; i.e., your name and password must be accepted by your mail service. When you start up Lafite, it obtains your name and password from the Lisp executive and submits them to the authenticator for verification. If the authentication fails, you are prompted to log in. Lafite will not allow you to send or receive messages until you have been authenticated.

You can start Lafite by typing one of two commands at the prompt in the executive window: (LAFITE) or (LAFITE 'ON). Lafite will automatically supply your Active Mail folder. You can look at another folder, say *MAILFOLDER*, by typing (LAFITE 'ON 'MAILFOLDER). You can also type (LAFITE 'ON NIL) to start Lafite without opening any mail folder, then later select a folder from the Browse menu after you enter Lafite; see chapter 5, "Reading Messages." You can also write and send (but not read, move, or otherwise manipulate) Lafite messages by selecting SendMail from the background menu; see chapter 9, "Sending Messages."

The Status Window

When you start Lafite, two windows will appear on your workstation's screen: the *status window* and the *browser window* (the latter will not appear if you typed (LAFITE 'ON NIL)). The other two important types of Lafite windows are the *message display window* and the *message composition window*. The browser and message display windows are fully discussed in chapter 5, "Reading Messages," and the message composition window is covered in chapter 9, "Writing Messages."

The Lafite status window contains a region for Lafite status information and a small, fixed menu of commands (see figure 2). When you first start Lafite, the status region informs you that Lafite is initializing and displays the time. This message quickly disappears, to be followed by either "New Mail for YourName" or "No New Mail at Time." Your in-boxes are checked for new mail approximately every five minutes; Lafite reports in the status region if you have new mail. If you click in the status region, the background process wakes up and reports status immediately instead of waiting its normal interval.



Figure 2. The status window, showing the status region, the title bar, and the menu of commands

After you have sent a message an additional window, your *out-box*, appears beneath the status window; it lists the dates and subjects of the messages you sent most recently. See chapter 10, "Sending Messages," for more information on the *out-box*.

The status window (and all other Lafite windows except the message display window) has its own fixed menu of commands, which are printed in boldface type. You invoke a command by moving the mouse until the cursor points at that command, then clicking the left mouse button. When you hold down the button, the command will appear *inverted* (i.e., white letters on a black background). When you release the button, the command will appear *grayed* (i.e., black letters on a gray background). If, while holding down the mouse button, you change your mind about the command you intend to select, simply move the cursor off the inverted command name until it is restored to its normal state, then release the button. In some cases, clicking the middle mouse button instead of the left performs some different but related operation or brings up an additional menu; such commands are described below.

The menu region of the Lafite status window contains three commands—Browse, Send Mail, and Quit. The Browse command is discussed in chapter 5, "Reading Messages," and the Send Mail command is discussed in chapter 10, "Sending Messages." The Quit command is discussed in chapter 11, "Leaving Lafite."

If you are using this manual as a tutorial, ask several people in your company to send you messages before going on to chapter 5, "Reading Messages."

The Browse Command

To access your mail from the Lafite status window, click the left mouse button on the Browse command. A menu will appear listing all your mail folders (see figure 3). Your new mail is conventionally stored in your Active Mail folder, and additional folders are added to the menu when you browse them or move messages into them. The creation of mail folders is discussed in chapter 8, "Filing Messages."



Figure 3. A folder menu, showing that the user has only one mail folder (the Other Folder item is used to create new folders)

To read the mail contained in a particular folder, such as Active Mail, click on the name of that mail folder with the left mouse button.

The Browser Window

Clicking on the name of a folder in the Browse menu opens a browser window onto that folder (see figure 4). The main part of the browser window displays a table of contents, which gives a one-line summary of each message, and a horizontal menu above the table of contents that contains commands that apply specifically to that folder. The topmost part of the browser window is the *prompt region*. In the prompt region, the system prints status information related to the browser and sometimes prompts you for information. The black *title bar* just above the table of contents labels the window.

Reading table of contents... done						
Display	Delete	Undelete	Answer	Forward	Hardcopy	Move To
Update Get Mail						
Mail Browser for (ERIS)<GRIMBLE>MAIL>LAFITERG.MAIL						
1	17 Apr	Frances Grimble:	Displaying a selected message [153 chars]			
2	17 Apr	Frances Grimble:	Message marks [135 chars]			
3	17 Apr	Frances Grimble:	"Delete" and "Undelete" [148 chars]			
4	17 Apr	Frances Grimble:	Reshaping message display windows [156 chars]			
5	17 Apr	Frances Grimble:	Thumbing [181 chars]			
6	17 Apr	Frances Grimble:	"Move to" and "Browse" [145 chars]			
7	17 Apr	Frances Grimble:	"New Mail" and "Get Mail" [148 chars]			
8	17 Apr	Frances Grimble:	"Hardcopy" [133 chars]			
9	17 Apr	Frances Grimble:	Composing messages [141 chars]			
10	17 Apr	Frances Grimble:	Recipient names [136 chars]			
11	17 Apr	Frances Grimble:	"Deliver" [132 chars]			
12	17 Apr	Frances Grimble:	Public distribution lists [147 chars]			

Figure 4. A browser window, showing the prompt region, the menu of commands, the title bar, and the table of contents

The table of contents is an index of the messages in the mail folder being viewed. The first time you browse a folder Lafite has to construct the table of contents, which it reports in the prompt region as "Parsing folder." Each summary line in the table of contents is numbered (the numbers are unique to that table of contents) and contains the date sent, the sender's name, the subject, and the number of characters in the message.

Each entry can also have a single-character *message mark*, which is an additional tidbit of information about the message displayed in the summary line. A message originally comes with a question mark, meaning it is unexamined. Some Lafite commands change the mark automatically. For example, displaying a message removes the question mark. You can change the mark directly by selecting with the mouse in the narrow area immediately to the left of the message number, where the mark is printed. Simply click in the mark position and type in the new mark. The message mark is the only piece of information in the table of contents you can directly modify.

The Get Mail Command

The table of contents in your folder does not display your new messages until Lafite retrieves them from your in-box. To instruct Lafite to move the contents of your in-box to your current mail file, point the cursor at the Get Mail command and click the left mouse button. The command then appears on a gray background and the prompt region reports on the activities of your mail server(s). When all the messages have been transferred, the gray background disappears and the prompt window tells you how many new messages you have. The table of contents is updated to include the new messages placed in your mail file; it is scrolled to display the first one.

Lafite automatically places a question mark next to each unread message in the table of contents (except for the ones you sent to yourself).

The Display Command and Selection Pointer

To read a message, you must first select its entry in the table of contents. After obtaining new mail from your in-box, Lafite automatically selects the first new message for you and places a *selection pointer* (a small black triangle pointing to the right) next to it. When you first browse a folder, Lafite selects the first unread message; if there are no unread messages in the folder, Lafite places the selection pointer next to the last message.

To display a selected message, point the cursor at the Display command in the browser window menu and click the left mouse button. A message display window will open and fill up with the message you selected (see figure 5). At this point, the question mark (if any) next to its entry in the table of contents disappears.



Figure 5. A message display window, showing the title bar and the message display region. The message display window lacks the message composition window's prompt region, command menu, and TEdit pop-up menus; you cannot use it for editing.

To examine the next message listed in the table of contents, click on Display again. The selection pointer will move to the next entry, and this message will replace the previous one in the message display window. Lafite will skip over deleted messages when advancing the selection pointer. To display a deleted message you must select it explicitly, then click on Display.

You may explicitly select any entry in the table of contents by moving the cursor into the line desired and clicking the left mouse button. Any existing selection pointers will be removed, and a new one will be placed at the indicated entry. You can have several messages selected at once. If you want to add a message to the current selection, click the middle mouse button instead of the left. To remove a message from the current

selection, hold down the shift key while clicking any mouse button.

If you want to select several entries in a row, place the cursor at the first message you want to read and click the left mouse button. Then move to the last message you want to read and click the right mouse button. Lafite will place selection pointers next to all the messages in between (deleted messages are not included unless the control key—Props on an 1108 or 1186 workstation—is down). You may then read them in succession by clicking on Display each time you finish reading a message.

If you have selected more than one message, clicking on Display after you read the last selected message will cause the cursor to cycle back to the first selected message. The only way to break this cycle is to explicitly select a single message.

If you want several messages displayed on the screen at once, you must create additional message display windows. To do so, click on Display with the middle mouse button. Lafite will prompt you to create a window. After you have done so, the message you selected to display will appear in the new window, and the previously displayed message will remain in the original message display window. You may create as many message display windows as you wish.

Text Scrolling

Lafite browser and message display windows may contain more text than you can see at any one time. (This is also true of message composition windows, which are discussed in chapter 9, "Writing Messages.") You can view this text by using a scroll bar to scroll the window's contents (see figure 6). The Lafite browser window has two scroll bars, one just beyond the left margin and one just below the bottom margin. The message display window has only the left scroll bar. The left scroll bar is used for vertical scrolling, and the lower scroll bar is used for horizontal scrolling.

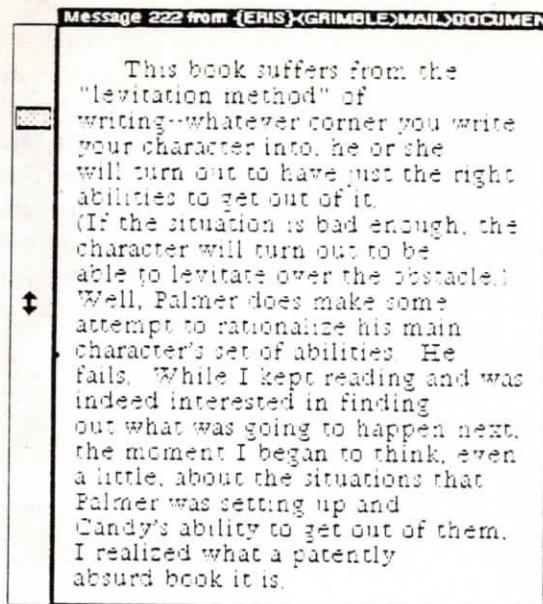


Figure 6. A message display window showing a vertical scroll bar. The gray area represents the position of the text being read in relation to the message as a whole; the double-headed shape of the cursor means that the message is not currently being scrolled.

A scroll bar can be brought into view by moving the mouse cursor onto that area of the screen. The cursor appears as a double-headed arrow until you begin scrolling, when it changes shape to point in the correct direction. The gray rectangle within the scroll bar represents the position of the text you are viewing in relation to the file as a whole. Within a vertical scroll bar, the left mouse button scrolls the contents of the window up, and the right mouse button scrolls them down. Within a horizontal scroll bar, the left mouse button scrolls the contents of the window to the left, and the right mouse button scrolls them to the right.

You can control the amount of text that is scrolled by moving your mouse cursor to different positions on the scroll bar. Scrolling with the cursor at the bottom of the vertical scroll bar moves the text about one window length at a time. Placing the cursor at the top of the bar moves the text about one line at a time. There is less variation in the amount of text you can scroll using a Lafite browser horizontal scroll bar, but placing the cursor at the left end scrolls the text by smaller amounts than placing it on the right. To produce continuous scrolling, hold down the mouse button; release it when the part of the file you want to read is shown.

Text Thumbing

If you want to skip to a distant part of the text quickly, *thumbing* is better than scrolling. Thumbing is analogous to opening a book by placing your thumb at the approximate position of the section you want to read and pulling the book open at that

point. To thumb the contents of a Lafite window, place the cursor in the part of the scroll bar that represents the part of the file you want to read. For instance, to reach the top of a file, place the cursor at the top of the vertical scroll bar. Then press the middle mouse button. The cursor will become a gray triangle pointing toward the window. When you release the mouse button, the text will move to show the part you want to read. If you haven't thumbed to quite the right place, you can find the text you want by scrolling.

The Delete Command

After examining the messages in your mail file, you may wish to delete some of them. Select the messages you want to delete, then invoke the Delete command by pointing the cursor at it and clicking the left mouse button. Lafite will draw a black line through each deleted entry in the table of contents (see figure 7).

	Display	Delete	Undelete	Answer	Forward	Hardcopy	Move To	Update	Get Mail
Mail browser for {ERIS}\GRIMBLE\MAIL\LAFITEFIG.MAIL:									
7	17 Apr	Frances Grimble:	New Mail	"Send " and "Get Mail"	[146 chars]				
8	17 Apr	Frances Grimble:	"Hardcopy"	[133 chars]					
9	17 Apr	Frances Grimble:	Composing messages	[141 chars]					
10	17 Apr	Frances Grimble:	Recipient names	[138 chars]					
11	17 Apr	Frances Grimble:	"Deliver"	[132 chars]					
12	17 Apr	Frances Grimble:	Public distribution lists	[147 chars]					
13	17 Apr	Frances Grimble:	"Get," "Put" and "Save Form"	[151 chars]					

Figure 7. A browser window showing a table of contents from which three messages have been deleted

When processing newly arrived mail, you may wish to delete a message immediately after displaying it. In this case, Lafite will automatically select and display the next message (assuming you have not examined or deleted it).

The Undelete Command

Messages marked for deletion are not actually removed from your mail file at the time the Delete command is given. If you discover that you have inadvertently deleted a message you want to keep, select it, point the cursor at the Undelete command, and click the left mouse button. The line drawn through the table-of-contents entry of the selected message will be removed. You can select several messages and delete or undelete them all at once; see chapter 5, "Reading Messages," for instructions on selecting messages.

The Update Command

With the exception of retrieving new mail, any change you make to a folder being browsed, such as deleting a message or changing a message mark, is not made permanent until you issue the Update command. That is, if you were to boot your machine, start up Lafite, and browse the same folder again, you would find it in the same state as when you last updated (or as when you last browsed it if you never updated). The Update command ensures that your mail file accurately reflects the contents of the browser, so that you could return to the same state if for any reason your machine failed and you had to start up again from scratch. It also provides the opportunity to reclaim the space occupied by deleted messages. Thus, it is important to regularly update your browsers.

When you click on the Update command, a menu appears listing your options: Write Out Changes Only or Expunge Deleted Messages (see figure 8). Ordinarily, you want to choose Expunge Deleted Messages, which, in addition to writing out all changes, permanently removes all deleted messages and rennumbers the remaining ones. However, Expunge can take a long time if your folder is large and there are many "holes" in it caused by deleting messages, since Expunge has to compact the file. Thus, if you are processing a lot of mail and just want to be sure periodically that your changes are saved, with the intention to expunge the folder later, you can choose the Write Out Changes Only option, which writes out your changes without compacting out the deleted messages. This option is usually faster than Expunge.

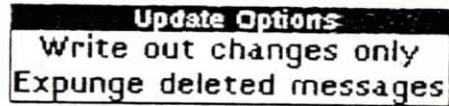


Figure 8. The update options menu

Other Ways to Update

When you attempt to close or shrink a browser window, using the Close or Shrink commands from the standard right-button window menu, Lafite will check whether you have made any changes to the folder, and if so, prompt you with a menu giving your updating options: Write Out Changes Only, Expunge Deleted Messages, or Don't Update File. Choosing either of the first two options is the same as clicking on Update and choosing that option, followed by closing or shrinking the window, as you requested. If you choose Don't Update File, then no update occurs—your changes are not saved.

If the only change you made to the folder was to retrieve new mail to it, or move messages into it from another folder (see chapter 8, "Filing Messages"), then you are instead offered the

options of Update Table of Contents or Just Close. The former is, in this case, all the updating that the folder requires.

You may find it convenient to keep the browser for your Active Mail folder always on your screen, shrinking it down to an icon to save screen space when you are not reading your mail (see figure 9). If you do this regularly, you may never need to use the Update command directly, instead combining the update with the shrinking of the window.

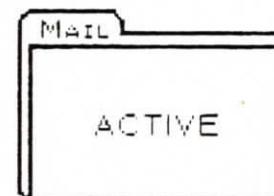


Figure 9. The icon for a shrunken mail folder, in this case Active Mail. You can shrink any Lafite window by placing the mouse cursor on the title bar of the window, pressing the right button, and selecting Shrink from the menu of window commands that appears

[This page intentionally left blank]

To print a copy of a message in your mail file, select it in the table of contents and click the Hardcopy command with the left mouse button. Several status messages will appear in the Interlisp-D prompt window reporting on the activities of your printer. When your message has been printed, its mark is changed to *h*. Lafite considers hard copy to be printed as soon as it has been accepted by the printer. There may be some additional delay before your message is actually printed.

You may select several messages to be printed together. Each message will be printed on a separate page (unless you have customized Lafite to batch printing requests).

You can print an unsent message from a message composition window using the right-button window menu's Hardcopy command; see chapter 9, "Writing Messages."

[This page intentionally left blank]

Most users prefer not to keep all their messages in their one Active Mail folder, but rather disperse them into several different folders, usually sorted by topic. This makes it easier to locate old messages of interest. Lafite allows you to create additional mail folders as you wish and to move your messages freely among them.

The Move To Command

To move messages from one folder to another, select the message(s) that you wish to move. Then click on the Move To command with the left mouse button. A menu will appear listing your existing mail folders, plus the item "Other Folder." If you wish to move the messages into one of the existing folders, simply select the folder's name from the menu. Lafite then prompts you to confirm the move: "Click LEFT to confirm move to *FOLDER*." If you click the left mouse button, Lafite completes the move, deleting the message(s) from the first folder's table of contents and marking them with an *m*. If you click the right mouse button, the move is aborted.

If you wish to move the messages into a folder not listed on the menu or to a new folder, select the "Other Folder" item. Lafite will prompt you with "*FILENAME* (CR to abort):" in the prompt region. Type the name of the folder, followed by a carriage return (or a bare carriage return if you change your mind and want to abort the command). Lafite then prompts you as above to confirm the move: "Click LEFT to confirm move to *FOLDERNAME*," followed by "[*OLD FILE*]" if the file you named already exists, or "[*NEW FILE*]" if it doesn't (see figure 10). If you click the left mouse button, Lafite performs the move, creating a new folder if necessary, and adds the folder's name to the menu for future selection.

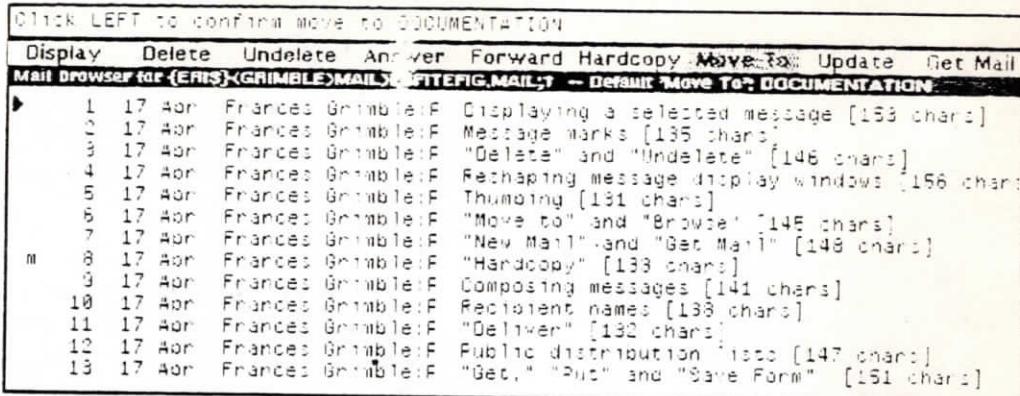


Figure 10. A browser window showing a Move To confirmation prompt and the name of the current default Move To folder

In either case, once the move is complete, the information "Default 'Move To': FOLDERNAME" appears in the title bar of the browser. This is for the benefit of the "accelerated" form of the Move To command. If you select the Move To command with the middle mouse button, Lafite performs the move to the folder named in the title bar without bringing up a menu of choices.

If you are already viewing the destination folder of a Move To in a browser on your screen, the newly moved messages are appended immediately to the browser's table of contents (although the browser is not scrolled to show them, as it would be with the Get Mail command).

How to Create New Folders

A new folder is automatically created if you ask to move mail into a folder that doesn't exist yet. You can also create a new folder without using the Move To command. Click on the Browse command from the Lafite status window and select "Other Folder" from the menu that appears. A small prompt window will appear with the prompt "FILENAME (CR to abort)" (see figure 11). Type the desired name of the folder and press the carriage return. If no folder of that name already exists, Lafite prompts you to confirm that you want a new folder, to make sure that you didn't merely mistype the name of an existing folder: "Click LEFT to confirm creating FOLDERNAME." If you click the left mouse button, Lafite will create the folder and prompt you to shape a browser window for the folder by showing a rectangle of dashed lines. The browser's table of contents will initially be blank, and the prompt region will report "Folder is empty." You can now retrieve new mail to this folder, using the Get Mail command, or move messages into it from another folder, as described above.

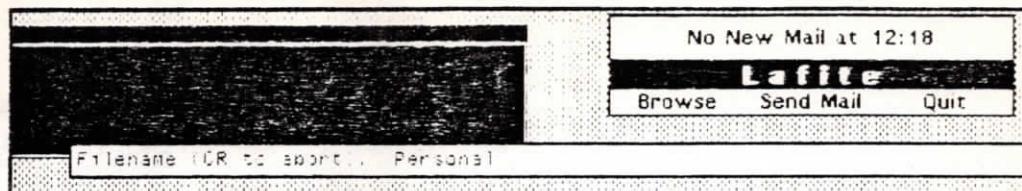


Figure 11. The prompt window for creating a new mail folder from the status window. The user first selected the Browse command, then the Other Folder item from the folder menu

How to View Other Folders

When you start up Lafite without specifying which folder you wish to view, Lafite automatically browses your default folder (usually ACTIVE.MAIL). If you instead wish to view a different mail folder at start-up time, then start Lafite by typing (LAFITE 'ON 'FOLDERNAME) to the executive window. Once Lafite is started, you can view any folder by clicking on the Browse command from the Lafite status window and selecting the folder from the menu offered, or selecting "Other Folder" and typing its name.

The Forget Folder Command

Once you create a folder, it remains in your folder menu (the one offered by the Browse and Move To commands) until you either expunge all messages from the folder, thereby deleting the folder, or you explicitly ask that the folder be removed from the menu. This latter way is needed, for example, if you have browsed various public mail folders that you no longer wish to see in your folder menu.

You can remove a folder from the Browse menu by invoking the Browse command with the middle mouse button. The resulting menu provides four options: Browse, Forget Folder, Forget Message Form, and Browse Laurel File (see figure 12). Browse Laurel File can be used only within Xerox. The Browse command is the same as the one in the Lafite status window, and the Forget Form command is discussed in chapter 9, "Writing Messages." To use the Forget Folder command, simply click on it with the left mouse button and select the folder you want to remove with the same mouse button. The Interlisp-D prompt window will tell you that the folder has been forgotten.

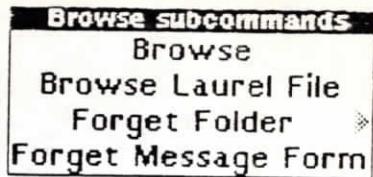


Figure 12. The middle-button Browse menu, showing the options for browsing files and deleting menu items

"Forgetting" a folder only removes the folder from the menu; it does not delete the file. You can always browse the folder again by selecting Other Folder from the menu and typing the folder's name. Within Lafite, you can permanently delete a folder by deleting all its messages, expunging it, and closing its browser window. In this case, the folder's name is automatically removed from the folder menu. You can also delete a folder outside of Lafite by using the File Browser Lisp Library package to delete and expunge the folder and its table-of-contents file (named *FOLDERNAME.MAIL-LAFITE-TOC*), or using the function DELFILE to delete those two files, by typing (DELFILE '{FILESERVER}< DIRECTORY > FOLDERNAME) to the top-level executive. In either case, Lafite is unaware that you have deleted the folder behind its back, so you must use the Forget Folder command if you wish to remove the folder's name from the folder menu.

Writing a message is the process of composing its header and body. *Sending* a message is the process of transmitting the message to its specified recipients. This chapter gives instructions for writing and editing messages, and chapter 10 tells you how to send them.

Lafite provides five kinds of forms you can use for writing messages. You may *compose* a new message, or *answer* one you have received, or *forward* existing messages to a new recipient. You can also use specialized message forms to *report* problems or suggestions regarding Interlisp, Lafite, or TEdit, or use message forms that you have personally composed.

All the message forms have the same essential structure, but they have different kinds and amounts of information already filled in when presented to you for further editing. Some message forms are built by browser commands; others from the Send Mail command. The characteristics common to every type of message form are described first, then the differences between the forms. Once you select a message form, the process of writing and sending each kind of message is the same.

The Message Composition Window

Each type of form is contained in a message composition window (see figure 13). The message composition window contains four regions. The topmost region is the prompt region, where Lafite gives you messages and sometimes prompts you for information. Below it is a menu with two commands, Deliver and Save Form; these are explained in chapter 10. The black title bar informs you that you are in the message editor. At the bottom is the message composition region, which has two parts separated by a blank line: the header and the body.



Figure 13. A message composition window showing a Standard message form. The top three lines of the text compose the skeleton message header; the text `>>Message<<` shows where you can begin the body of your message

The Message Header

The Standard form shown in figure 13 contains a **skeleton message header**. This header has two blank *fields* to be filled in that provide information to your recipient and enable Lafite to deliver your message. These are the **Subject** field and the **To** field. The third field, the **CC** field, is optional; Lafite automatically inserts your name in it.

The skeleton message header contains only the fields that you normally need to fill in yourself to send a message. There are a large number of fields that can possibly appear in a message header, as you can see from viewing the existing mail in your mail folders. Some fields (the date and sender information) are added automatically when Lafite delivers the message; some are added by mail servers along the way; and some are fields that you can add yourself if there is a need.

Each separate field of the header begins on a new line with a word immediately followed by a colon. The word preceding the colon specifies the meaning of the text following the colon. For example, **To:** specifies that the words following the colon are to be interpreted as recipient names, and **Subject:** specifies that the text following the colon is the subject of the message, which is of interest to the recipient but not part of the delivery information.

Lafite identifies items it expects you to replace by supplying keywords between reversed, double angle brackets. To make it easier to fill in these items, the first field is highlighted as a delete selection. Typing any text will cause the keyword and

brackets to be deleted. You can then fill in the field with the text of your choice. Each successive field (except the CC field) can be reached by pressing the Next key on the 1108 or 1186 keyboard (or the middle blank key on the 1132 keyboard). The final item in most message forms is >>Message<<, where you can begin the body of your message. You can also select fields with the mouse.

Because Lafite allows you to edit any part of your outgoing messages, you must observe certain rules to keep them intelligible to Lafite. The header must contain at least one recipient, and the names of multiple recipients in one field must be separated by commas. Fields must be separated by single carriage returns, so that each begins on a different line. The header and body must be separated by a double carriage return, leaving a blank line between them when the message is displayed in the Lafite window. All of Lafite's initial message forms have a blank line following the header. Do not remove it or insert a blank line inside the header; otherwise information contained in the header will be misunderstood by Lafite.

The Subject Field

The Subject field is used to state the topic of your message (see figure 14). The topic should accurately express the content of your message so that interested people will take the time to read the message, but uninterested people can delete it without reading it. An accurate Subject field also facilitates differentiating the message later on from others in the same folder. For example, if your message contains ideas for improving Lafite, the topic might be "Suggestions for new printing features in Lafite," rather than "Lafite" or "Suggestions."

Deliver	Save Form
Message Editor	
Subject: REVIEW: Time's Arrows	
To: Books:PARC:Xerox	
Reply-to: JimDay:PASA:Xerox	
Title: Time's Arrows	
Author: Morris, Richard	
Publisher: Simon & Schuster	
Date: 1984	
Richard Morris holds a Ph.D. in Theoretical Physics and is the author of five previous books on science. In this book he presents a fascinating overview of the attitudes toward time that have influenced Western thought, science, and technology. We all know what time is, don't we? It's what a clock measures. Nothing could be simpler. But considered in detail, the topic of time is seen to be very complex, with many puzzles for philosophers, psychologists, and physicists.	

Figure 14. A message composition window showing several filled-in user-specified fields. Note that, because the message is addressed to a public distribution list, the sender has included a Reply-To field

The To Field

The To field (and the CC field, see below) is where you specify who is to receive your message. Lafite allows you to specify multiple recipients in the To field, provided their names are separated by commas. Conventionally, your main recipients are listed in the To field and the CC field is used to copy the message to additional recipients.

You need to know the registered address of each individual or group to whom you want to send a message. You can often discover these addresses by looking at the headers of incoming messages.

Recipient Addresses. NS recipient addresses have three parts—a *name*, a *domain*, and an *organization*—separated by colons, in the form "Cheryl Jones:PARC:Xerox." A name can represent either a person or a group. Every name belongs to a particular domain, which is usually a small division of an organization that shares common resources, such as file servers and printers. And every domain belongs to an organization, which is usually a single company or a major division within a company. Thus, Cheryl Jones is an employee at PARC, a division of the company Xerox.

Within a domain, a person or group can have several alternative names known as *aliases*. An alias can always be used in place of a person or a group's real name. Thus, if Cheryl Jones's alias is Jones, her mail can be addressed to Jones:PARC:Xerox. If you want to create an alias for your name, see your network administrator.

Lafite allows you to omit the domain and/or organization for recipients who are in your own domain and organization, just as you may omit an area code when telephoning your neighbor across the street. However, you must include the domain and organization for other recipients. If a recipient's domain or organization is left out and is not the same as your own, Lafite will redisplay the message, and the Interlisp prompt window will tell you which recipient's address needs clarification.

Your default domain and organization (the variables CH.DEFAULTDOMAIN and CH.DEFAULTORGANIZATION) are usually set in your initialization file, or are set automatically if your network has only one accessible domain. You can change them by typing (DC INIT) to bring up a DEdit window on your initialization file, resetting the variables, then typing (MAKEFILE '{FILESERVER}<DIRECTORY>INITFILENAME) to save the new version of the file. For more information on DEdit and the MAKEFILE function, see the *Interlisp-D Reference Manual*.

Public Distribution Lists. The mail system provides a way to address messages to groups of recipients called *public distribution lists*. NS public distribution list addresses have the same form as individuals' addresses. Using a distribution list as the recipient of a message causes the message to go to all the

individuals included in the group. For example, the To line "To: AISBU:PARC:Xerox" will cause the message to be delivered to all the members of the Artificial Intelligence Systems Business Unit in the Palo Alto Research Center domain.

Public distribution lists are stored and maintained on a Clearinghouse by a list owner or a network administrator; individual members can add themselves to some lists. If you want to create, add your name to, or remove your name from a public distribution list, contact your list owner or administrator, or use the NS Maintain program (documented in the *Lisp Library Packages Manual*).

While you are permitted to address a message to any public distribution list, you should think very carefully about your choice of message and list so as not to bother recipients with messages they don't care to read. Check with experienced users to see which lists should be used for which kinds of messages (see also appendix A, "Using Lafite Courteously"). Also, add a Reply-To field if appropriate (see below).

You need not worry about a recipient appearing in more than one distribution list. The mail system will detect duplicate names among recipients and ensure that each receives the message only once. Thus, you may use multiple distribution lists freely within the To and CC fields of the message header.

NS Maintain enables you to obtain lists of names, aliases, domains, organizations, and members of groups on your Clearinghouse network. For information, see the *Lisp Library Packages Manual*.

The CC Field

The CC field is used to list persons other than your main recipient(s) who are to receive your message. Names must be separated from each other by commas. When you send your message, these people will automatically receive it along with those specified in the To field. Lafite automatically puts your name in the CC field; you can delete it if you wish. If you wish to send no copies of the message, not even to yourself, delete the entire field.

The From and Sender Fields

Lafite automatically supplies a From field containing your name, which is not visible before the message is delivered. If you want the message to appear to be "from" someone else (e.g., if you are sending the message from somebody else's logged-in Lafite), or from more than one user, you can supply your own From field. In this case, Lafite will supply a Sender field to show who actually sent the message (see figure 15).

Message 84 from {ERIS}<GRIMBLE>MAIL>ACTIVE

Date: 26 Jun 85 13:28
 Sender: Frances
 Grimble:PARC:xerox
 Subject: I need help moving!
 From: Cheryl Jones:PARC:Xerox
 To: Frances
 Grimble:PARC:Xerox

I'm moving from a house in Sunnyvale to an apartment in Mountain View this weekend. Some of my furniture is too heavy for me to handle, so I am having a moving party on Sunday in the hopes of getting some help. The address is 550

Figure 15. A message display window showing the From and Sender fields. In this case, the sender (Cheryl Jones) wrote and sent the message from the recipient's logged-in Lafite

The Date Field

During delivery, Lafite inserts a Date field in the form "Day Month Year Time [in international hours]." Thus, if you send a message on April 1, 1985 at 1:45 p.m., the Date field will read "1 Apr 85 13:45." You cannot supply your own Date field.

The Reply-To Field

When sending a message to a distribution list, or to a large number of individual users, it is important to keep in mind your audience and what you want to have happen if someone wishes to answer your message. Very often it is the case, especially with large distribution lists, that you want replies to be sent only to you, or to some small designated set of people. For example, you send a message announcing a meeting and want people to tell you if the time is bad for them. Or you are taking a poll. Or you are simply making an announcement to which you don't even expect replies. Message system etiquette demands that messages not be inappropriately sent to large numbers of people, as would ordinarily occur if a recipient of your message used the Answer command and didn't take the time to notice that a distribution list was copied on the resulting message.

It is for this reason that the Reply-To field exists. If a message contains a Reply-To field, most message-handling systems, including Lafite, will compose answering messages addressed only to the user(s) named in the Reply-To field, not to the entire recipient list of the original message. To add a Reply-To field to your message, simply insert a line beginning "Reply-to:" followed by one or more addresses (in the same form as the To

field), anywhere in the header (but conventionally at the end of the header). Be sure you still have a blank line between the header and the body of the message.

The In-Reply-To Field

When you use the Answer command (see below) to reply to a message, Lafite automatically adds an In-Reply-To field to the message form. This field contains your name and the date of the message you are answering, in the form "In-reply-to: Cheryl Jones:PARC:Xerox's message of 5 Mar 85 17:29." You can edit it if you wish.

The Message Editor

You have available to you the full power of TEdit when composing messages. Lafite message composition windows are TEdit windows and are used in the same way. Because TEdit has many more features than can be covered here, we will describe only the most basic procedures for writing and editing messages. If you are familiar with TEdit, you can skip this section. If, after reading this section, you want to know more about TEdit, see the TEdit documentation in the *Lisp Library Packages Manual*.

The Type-in Point

The blinking caret in the message composition window is called the *type-in point* (see figure 16). The Interlisp environment has only one type-in point active at a time; you tell the environment you want to type in the message composition window by clicking a mouse button on that window. Now whatever text you type will appear at the type-in point.

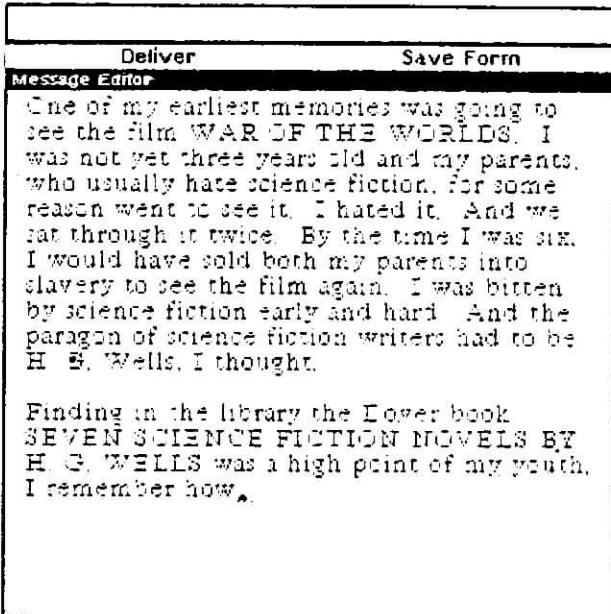


Figure 16. A message composition window showing the type-in point and text wraparound. The caret after "how" is the type-in point. The text will be adjusted automatically when the window is reshaped or the message is printed.

TEdit automatically breaks text between words and sends the overflow to the next line. Don't use a carriage return when you think you have reached the end of a line; use one only when you want to insert blank space between paragraphs or sections. TEdit automatically adjusts the text of your message to fit the width of your recipient's message display window or your printer's requirements, but it leaves blank lines in place.

Text Selection

You may want to change the text of your message after you type it in by deleting it, copying it, or moving it. To change the text of your message, you first say where you want the change made by making a selection. Then you say what you want done by giving a command. Just making a selection has no effect on the message; only commands can change it. Each selection supersedes the previous one; that is, there is only one selection at a time. Deletion and other operations are applied to the currently selected text. You can tell when text is selected because it is highlighted by underlining or reverse video (usually white-on-black); it may have a caret flashing at one end.

There are two regions within an editor window, and which region the mouse cursor is in determines what kind of selection happens. The regions are the area containing text and the line bar that forms the left border of the window (this is not the same as the scroll bar described in chapter 5).

You select text by first pointing with the mouse, then pushing one of its buttons. The left mouse button always selects the smallest units. In the text region, it selects the character you're

pointing at; in the line bar, it selects the single line you're pointing at.

The middle mouse button selects larger units. In the text region, it selects the word the cursor is over, and in the line bar it selects the paragraph the cursor is next to.

The right mouse button always extends a selection. If you select a place in the text with the left or center mouse button, move the mouse cursor somewhere else, and click the right mouse button, all the text in between the two points is selected. If the existing selection is a whole line or paragraph, the extended selection is also a whole line or paragraph.

Text Deletion

There are three ways you can select text for deletion. If you hold down the control key while selecting text, the selected text will be deleted when you release the key. You can also delete text by selecting it and then pressing the Delete key. You can delete text one character at a time by pressing the backspace key; the character to the left of the caret will disappear. Figure 13 shows some text highlighted as a delete selection.

Text Moving

To move text, first click the left mouse button to put the caret in the place you want the text moved to. Then select the text to be moved while holding down the Move key (or the control and shift keys on an 1132 workstation). When you release the key(s), the text will be moved to where the caret is. You can use this method to move text within a message composition window or to move it from one window to another.

Text Copying

If you want to copy text, first put the caret in the place you want the text copied to. Then hold the Copy key down while selecting the text to be copied. When you release the Copy key, the text will be copied to the location of the caret. You can copy text within a window or from one window to another. However, the most efficient way to copy an entire file from one window to another is to use the Include or the Get command; these are described below.

The Include Command

It is often desirable to electronically mail a document, such as a report, that is not a letter. You can conveniently do this by including that file in a Lafite message. First, you open a message composition window and fill in the header information. Next, you type in any covering message you want to send and place the blinking caret where you want to insert the file. Then put the mouse cursor on the black title bar of the window and press the left or middle mouse button.

The basic TEdit command menu will pop up, it contains the commands Put, Get, Include, Find, Looks, Substitute, Quit, and Expanded Menu (see figure 17). While you can use all these commands from within Lafite, only the Include, Put, and Get, commands are described in this manual. For detailed information you should consult the TEdit documentation.



Figure 17. The basic TEdit command menu. Selecting Expanded Menu will bring up a menu containing most of the commands on the basic menu, plus commands that enable you to access additional menus—the Character Looks, Paragraph Looks, and Page Layout menus

To include a file in a message, select Include with the left mouse button. TEdit will ask you for the name of the file you want to load. Type the file name in the prompt region and press the carriage return. After a pause to read the file, your message composition window will fill up with the designated file. The entire file will be highlighted by underlining. To get rid of the underlining, simply click the left mouse button anywhere in the file.

Be careful to give TEdit the exact name of the file you want to include; do not change spacing or punctuation. If TEdit tells you that it can't find a file, check to see if you typed the name correctly.

The Put Command

You can save messages in TEdit files using the Put command and finish or deliver them later on. Bring up the TEdit command menu as described above and select Put with the left mouse button. TEdit will ask you for the name of the file to put to (see figure 18). The syntax for naming files stored on your disk is {DSK}FILENAME. For files stored on a file server, it is {FILESERVER}<DIRECTORY>FILENAME. You don't need to specify a file server or directory if you want TEdit to use your default file server and directory. Make sure that no blank spaces appear in your file name.

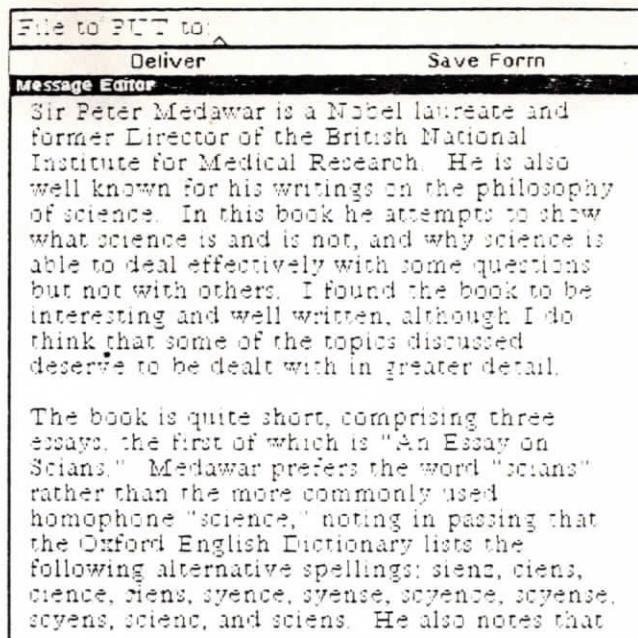


Figure 18. A message composition window after the user has chosen the Put command. The message will be saved and retrieved as a TEdit file

Type the file name in the prompt region and press the carriage return. If you have already saved the file, TEdit will prompt you with the file name you specified earlier. You can either change this file name or confirm the earlier name by pressing the carriage return. TEdit will tell you that it is putting the file, then that it is done. You can now close your message composition window and later retrieve the message with the Get command.

The Put and Get commands are very useful for saving messages that can't be delivered due to network problems. (Or you can shrink the message composition window like any other Interlisp window; the icon looks like an envelope and is labeled "Unsent.") You can also save messages with the Save Form command and retrieve them from the middle-button Send Mail menu; see below.

The Get Command

To retrieve a previously composed, undelivered message, bring up a Lafite message form and select Get from the TEdit menu with the left mouse button. TEdit will prompt you for the file to get. Type its name in the prompt region and press the carriage return. The mouse cursor will change to an hourglass until TEdit finishes getting the file. Once the file is loaded, you can edit or send your message. If you get a Lafite mail folder into TEdit and edit it, change the file's name when you save it. If you continue to use the file as a Lafite folder, Lafite will be unable to parse the folder the next time you open it (see chapter 12, "Troubleshooting Lafite Problems").

The Hardcopy Command

You can use the Hardcopy command in the Interlisp-D window menu to print messages before you send them. You bring up the window menu by placing the mouse cursor on the title bar of a message composition window and pressing the right mouse button. The menu of window manipulation commands shown in figure 19 will appear. Select Hardcopy and wait for TEdit to tell you it is done formatting the message for print. You may have to wait a little longer before your printer has it ready for you.

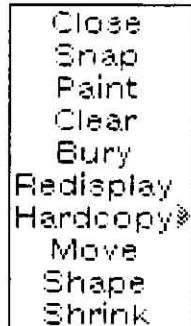


Figure 19. The Interlisp-D right-button window menu. The TEdit documentation in the Lisp Library Packages Manual fully describes how to use this menu with the text editor

Fancy Message Formats

Lafite supplies you with an attractive default message design. If you don't specify otherwise, your messages are displayed in a 12-point Times Roman font. They are printed in a 12-point Classic font on an Interpress printer or a 12-point Times Roman font on a Press printer. They have no line or paragraph leading, and the margins are set flush left and ragged right.

If you wish, you can change this design using TEdit's font and formatting options. (You can also change Lafite's default fonts by resetting global variables; see chapter 13, "Customizing Lafite.") These include several additional fonts and type sizes, boldface and italic type, justified margins, and tabs. Detailed instructions on designing documents with TEdit are contained in the TEdit documentation. You can send specially designed messages either formatted (containing the special format) or unformatted (in the default design). See chapter 10 for instructions on sending formatted messages.

Messages With Bit Maps

Bit maps can be included in messages sent to other Lafite users (see figure 20). To insert a bit map in a Lafite message, first get the picture onto your screen. Place the caret in the place in the message composition window where you want the image to appear. Hold down the shift key and depress the right mouse button in the background. This will bring up a rectangle for Snap, a window menu command that makes a copy of any portion of your screen and allows it to be placed elsewhere on

the screen, for instance in a message composition window. Move the cursor into the rectangle, turning it white-on-black. When you release the key and button, the cursor changes from an arrow to a prompt design. Move the prompt (as if it were the cursor) to the bit map you want to include in the text. Press down the left button and outline the bit map as closely as possible, in the same way you would create a region for a window. When you let up the left button, the bit map will appear where you left the caret flashing in the message composition window.

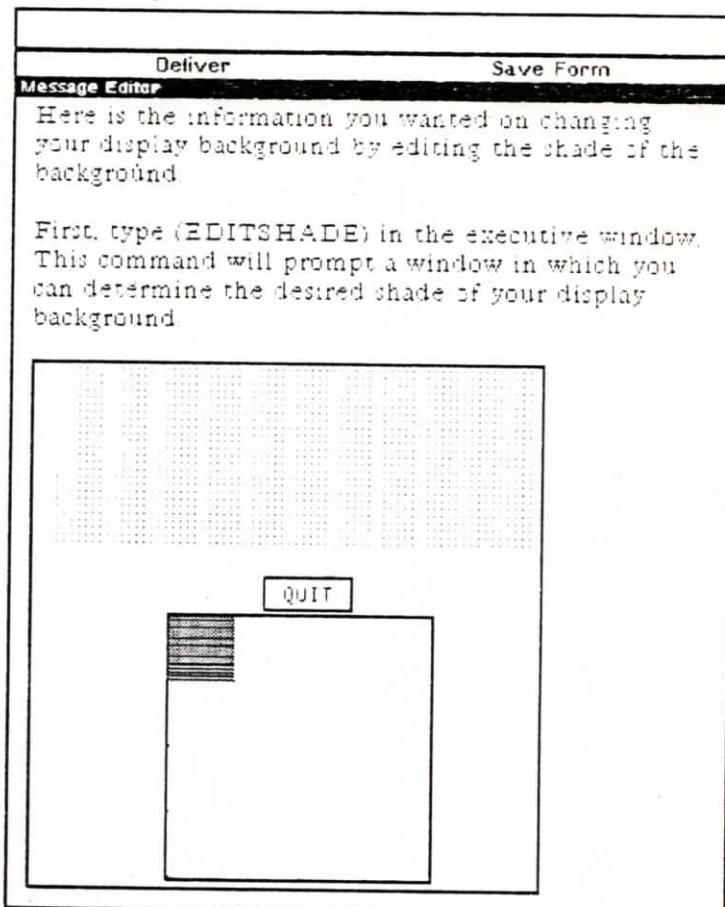


Figure 20. A Lafite message containing a bit map. It is often convenient to include a screen image in a Lafite message when you are giving technical information or reporting problems with the system

You can edit a bit map by clicking inside it with the left mouse button, which brings up a menu of editing operations. You can delete, copy, or move a bit map as if it were a single text character. If you insert a carriage return before and after a bit map, you can adjust its position with the TEdit paragraph-formatting menu (see the TEdit documentation). Bit maps are saved when you save a file, and they are printed when you hard-copy the message (providing your printer can print bit maps).

Messages containing bit maps are automatically sent as formatted messages; see chapter 10.

Message Forms

As described above, Lafite provides five kinds of forms you can use for writing messages, each of which is accessed by a different command or commands. The kinds of forms are the Standard form, the Answer form, the Forward form, the Report form, and the *Personally Created* (or saved) form. This section describes each form and how to use it.

The Standard Form

The Standard message form is a message composition window containing blank To and Subject fields and a CC field with your name in it (see figure 13). You can access the Standard message form in two ways. The first is to use the Send Mail command in the Lafite status window. If you click on this command with the left mouse button, Lafite will open a window containing the standard message form. If you click on Send Mail with the middle mouse button, Lafite will present you with a menu of several forms, including the names of any you have created yourself plus Lisp Report, TEdit Report, Lafite Report, Saved Form, and Standard Form (see figure 21). To access the Standard form, select it from the proffered menu. (Personally Created forms and Report forms are discussed below.)



Figure 21. The middle-button Send Mail menu. In this case the user has no Personally Created forms

You can also access this second menu without otherwise entering Lafite by clicking on your screen background menu with the right mouse button and selecting SendMail from the menu that pops up. If you choose SendMail without having entered Lafite from the executive window, you cannot read, move, or otherwise manipulate messages—only write and send them using the Standard, Personally Created, and Report forms.

The Answer Form

Selecting the Answer command from the browser window menu constructs a message form that is a skeleton reply to the selected message or to the first selected message if more than one is selected (see figure 22). The header of the form contains To and CC fields listing the sender of the message and its recipients, a Subject field of the form "Re: *Subject of selected message*," and an In-Reply-To field containing the name of the sender and the date and time the message was sent. For example, if you are replying to a message sent to you by Cheryl Jones on March 19,

1985 at 10:28 a.m., the field will be "In-Reply-To: Cheryl Jones:Domain:Organization's message of 19 Mar 85 10:28."

>>Message<<'."/>

Figure 22. An Answer form, showing the skeleton message header created by Lafite. The user can edit the header as he or she desires

The form provided by the Answer command is not always exactly what you might want. Lafite takes the viewpoint that it is easier to delete than to add text, and therefore includes all information that seems to be relevant. For example, Lafite often includes your name in the CC field of the answer form, even though you may not want a copy of the message you are composing. You should not expect that the answer form will be exactly right; always examine the header to be certain it contains the desired information.

In particular, if you reply to a message that was directed to a distribution list, the Answer command will copy the distribution list name into the CC field of the Answer form (unless the sender included a Reply-To field). You should consider carefully whether it is appropriate for your reply to be sent to that distribution list, and if not delete the distribution list's name.

After you deliver the answer to a message, an *a* will appear in the browser window as that message's mark.

The Forward Form

Selecting the Forward command from the browser window menu constructs a message form containing the complete text of all the selected messages, and a message header whose subject is the sender and subject of the first selected message, in brackets. You should fill in the desired recipients and a covering message if you desire one. You can also change the subject if you wish.

After you deliver a forwarded message, an *f* will appear as that message's mark.

Specialized Report Forms

The Report forms are used to report problems or make suggestions regarding Interlisp or an application program. Although they always appear on the middle-button Send Mail menu, you cannot use them unless your organization has set up a local support address or addresses to which you can send the reports (see chapter 13, "Customizing Lafite," for instructions). If you choose one of the Report forms and an address has not been set up, a message will be printed in the prompt window saying that the address is not available. If you cannot use the Report forms, please move on to the section below on creating customized message forms.

The following three sections describe the headers of the Report forms, including the information that is provided automatically and the fields you fill in. We have assigned meanings to the fields and made recommendations for reporting problems and requesting features. These guidelines are included because we have found them to be effective and feel they may be helpful to our users. However, they may differ from the policies of your local support organization, so check with that organization before using these forms.

The Lisp Report

The Lisp Report has a header similar to that of the Standard form, plus a section giving certain information about the hardware and software you are using and asking you to supply information about your problem or feature request. The Subject field is filled in with "Lisp: >>Terse summary of problem<<." Replace the words in reverse angle brackets with a subject that accurately identifies your problem or request. Subjects like "Floppy problem" are imprecise; instead, use subjects like "Attempt to write file when floppy door is open causes awful noise." Feature requests generally start with the word "Want," e.g., "Want to make windows circular rather than rectangular."

The To field contains the address of your local Lisp support. You do not need to modify it. The CC field initially contains your name.

Lafite automatically fills in the date your sysout was made, the kind of workstation you are using, the microcode version, and the memory size. In addition, there is a Frequency field containing the words ">>Always, Intermittent, Once<<." The Frequency field is used to report the reproducibility of a problem; it is usually irrelevant for feature requests. If the problem occurs every time you try to perform a task, choose Always and delete the angle brackets and the Intermittent and Once options. (This is most easily done by copy-selecting Always while the bracketed field is delete-selected.) If the problem doesn't always happen, choose Intermittent; if you saw it happen once, choose Once. If the Frequency field is irrelevant, leave it blank or delete the whole line.

Use the Impact field to describe how seriously a problem or the lack of a feature affects your ability to get work done. The names apply to bug reports, but feature requests should be rated along analogous lines. Choose Fatal if the problem causes system crashes, loss of work, etc., or if a feature is required for project completion. Choose Serious if the problem can be worked around but seriously interferes with your work. Choose Moderate if the problem is tolerable but clearly a problem. Choose Annoying for annoying problems and requests for features that "would be nice." Choose Minor for very minor feature requests or problems that may not even be bugs.

Replace the words "detailed problem description" with a description of the problem and what you were doing when you encountered it, plus any other information that might be useful. Always be as precise and detailed as possible; it is extremely difficult to solve a technical problem described only as "Floppy doesn't work."

The Lafite Report

The Lafite Report differs from the Lisp Report in that the Subject field is filled out as "Subject: Lafite: >>Terse summary of problem<<" and the Lafite system date appears above the Lisp system date (see figure 23). You should report problems and requests for new features in the same way as in a Lisp Report.

Message Editor	Deliver	Save Form
Subject: Lafite: >> Terse summary of problem <<		
To: LafiteSupport.pa		
cc: Grimble.pa		
Lafite System Date: 12-Mar-85 03:12:27		
Lisp System Date: 16-Apr-85 13:02:18		
Machine: Dandelion (131#61#)		
Microcode Version: 16.24		
Memory size: 5777		
Frequency: >> Always, Intermittent, Once <<		
Impact: >> Fatal, Serious, Moderate, Annoying, Minor <<		
>>detailed problem description<<		

Figure 23. The Lafite Report form, showing the fields you need to fill in and the ones Lafite fills in automatically

The TEdit Report

The TEdit Report is exactly like the Lafite Report, except that "TEdit" appears in the subject instead of "Lafite," and the TEdit system date is given instead of the Lafite system date.

Personal Message Forms

You can create, save, and reuse Personal message forms or form letters with Lafite's Save Form command. Simply write your message in the message composition window, then choose Save

Form from the window's menu with the left mouse button. Lafite will print the following message in the prompt region: "Save form under name:" Type the name you want the form to have (it cannot contain a space) and press the carriage return. If no extension is given for the file name, Lafite defaults it to Lafite-Form. Lafite will print the full file name in the prompt region, then close the message composition window.

When composing your Personal form, you will probably want to use the same convention as the regular Lafite forms do for items that will need to be filled in when the form is used to send a message. That is, enclose a describing word or phrase in reverse angle brackets, so that TEdit's Next key can be used to easily skip from one such item to the next. In addition, if you include the text ">>Self<<" anywhere in the message, it will be automatically replaced with the name of the logged-in user when the form is requested. The most common use of this is to put the line CC: >>Self<< in the header. This is better than putting your own name there, since your form can be used by other people without having to edit the CC field. Of course, if you edit such a form and save it away again, you must remember to replace your name with >>Self<< again, since the >>Self<< is replaced with your name when you bring up the form.

There are two ways you can access Personally Created forms. After you save a form, its name is added to the middle-button Send Mail menu, and you can select it just as you would any of the other forms on that menu. You can also select Saved Form from the same menu; this brings up a small window asking for the file name. Type in the name of the form and press the carriage return.

When you select, edit, and deliver a Personally Created form, you are only editing a copy of the form; the original remains unchanged. Of course, if you want to change the actual form you can edit it and save it again with the Save Form command.

Note that Save Form is intended for the creation of new message forms. If you merely wish to save a partially composed message for later completion, TEdit's Put command is more appropriate.

The Forget Message Form Command

Once you have saved a form, its name will continue to appear in the middle-button Send Mail menu. If you no longer want that form in the menu, use the Forget Message Form command in the middle-button Browse menu to remove it. Select Browse with the middle mouse button, then Forget Message Form from the proffered menu (see figure 12). This brings up a menu of Personally Created forms; select the one you want to forget with the left mouse button. The Interlisp-D prompt window will tell you the form has been forgotten. Note that, as with the Forget

Folder command, this does not delete the actual file. To delete the file, type (DELFILE '{FILESERVER}<DIRECTORY>*FILENAME.LAFITE-FORM*) in the executive window. You can also use the File Browser Lisp Library package.

[This page intentionally left blank]

Once you have written the message you want to send, you can initiate its delivery to the recipient by pointing the cursor at Deliver and clicking the left mouse button. Lafite will fill in your name and the date (though they won't appear in the window) and proceed to send the message. A gray background will appear behind the Deliver command, then the command will change to Abort (see figure 24). If you decide not to send a message after buttoning Deliver, select the Abort command and Lafite will halt the delivery of the message.

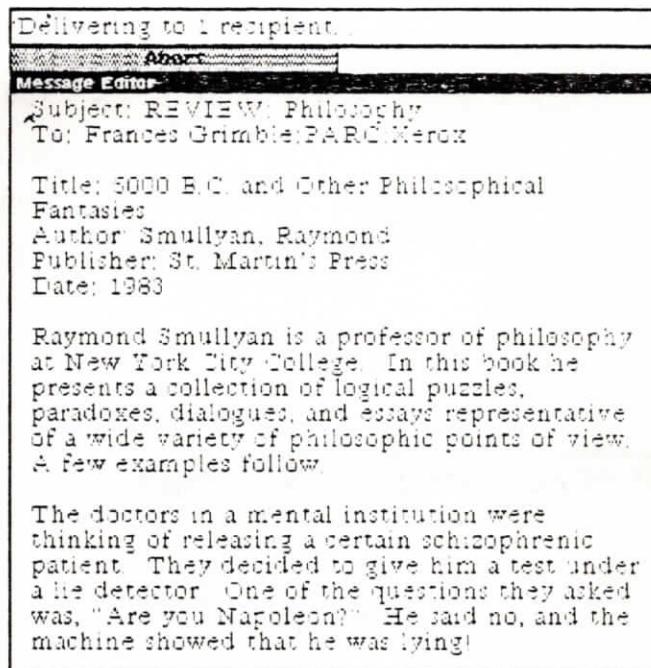


Figure 24. A message composition window after Abort was chosen. After Lafite aborts the delivery, the word "abort" will appear in the prompt region and the message will be redisplayed

Invoking Deliver will send the message to all its intended recipients. Each recipient will receive only one copy of the message, even if his or her name appears more than once (a name might be on several distribution lists specified in the To or CC field).

If Lafite discovers an error in the list of recipients, it will tell you that the "recipient is not understood" and give you an opportunity to correct the mistake and invoke Deliver again. When the list is acceptable (i.e., all specified recipients are known to have in-boxes), Lafite will deliver the message. After successful delivery, the word Abort will disappear. Lafite will tell you that it is done and close the message composition window.

Delivery happens in the background, so you can perform other tasks while delivery proceeds.

After a message is delivered, it is entered into your out-box, a window attached to the bottom of your status window (see figure 25). This window contains one-line descriptions of the two messages you sent most recently. The out-box is treated as a menu—selecting a line in it brings up the corresponding message for further editing and delivery. The out-box can be independently closed if you are no longer interested in the messages displayed therein.



Figure 25. A status window showing the out-box. The number of messages retained can be changed using the LAFITEOUTBOXSIZE variable; see chapter 13, "Customizing Lafite"

If you decide not to do anything with a message you started to compose, simply close the message composition window with the standard right-button window menu. Lafite asks you to confirm flushing the message; its text will not be saved.

Delivery of Formatted Messages

Only Lafite users can read a formatted message; all other mail readers will see only the plain text of the message. If you try to deliver a formatted message, Lafite asks if you want to retain the formatting information, putting up a menu with the choices Send Formatted Message, Send Plain Text, and Abort. Send Formatted Message retains all the formatting information. This choice is made automatically if the message contains a bit map, because there is no way to send the image without the formatting. If you send a message containing a bit map to a non-Lafite user, the content of the bit map will appear as a string of trash, which may disturb some message-reading systems.

Send Plain Text sends only the text of the message. The message will appear to the recipient in whatever font his or her mail reader usually uses, and all paragraph formatting (centering, justification, special tab stops) will vanish. Sometimes a message appears to Lafite to be "formatted" when it really only contains inconsequential variations in fonts or line spacing, typically the result of copy-selecting text from another window. In such cases, where you did not intend special formatting, it is best to choose

Send Plain Text to reduce the message overhead and avoid imposing your default font choice on your recipients.

Abort does not send the message, but returns you to the message editor to continue editing the message.

[This page intentionally left blank]

There are two ways you can leave Lafite. You can select Quit from the status window (or the middle-button Quit menu) which stops Lafite and closes all browser windows. If any of your open mail folders need updating, Lafite prompts you with a menu asking what degree of updating should be performed: Write Out Changes Only, Expunge Deleted Messages, Don't Update File, and Don't Quit; the option you select is applied to all open folders (see figure 26). You must select one of the first three options before Lafite will let you quit. When you do, all your Lafite windows will disappear, indicating that you have successfully quit Lafite.

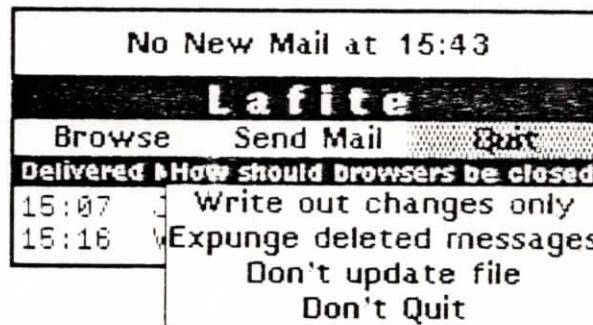


Figure 26. The update menu that appears when you choose Quit and an open mail folder (or folders) needs updating

You can also leave Lafite by typing (LAFITE 'OFF) in the executive window, which has the same effect as selecting Quit.

If you select Quit with the middle mouse button, you get a menu that includes the original Quit command and the Restart command. The Restart command turns Lafite off and then immediately back on again—it is equivalent to typing (LAFITE 'OFF) followed by (LAFITE 'ON NIL). It is mainly useful if you have changed some of your Lafite personalization and you want Lafite to notice the new settings.

You may want to keep the Lafite status window in view instead of quitting Lafite each time you finish reading or sending a batch of mail. The status window keeps you constantly informed about the state of your in-boxes, and it takes up little room on your workstation screen.

[This page intentionally left blank]

Although Lafite is a robust system, you may occasionally encounter minor problems when using it. This chapter describes the most common Lafite problems and how to prevent and/or recover from them.

Lafite Confuses You With a Previous User

If you log into Lafite with the correct name and password but the Browse menu shows no folders or shows someone else's folders, Lafite may be confusing you with a previous user. If you are running in a sysout in which someone else has been using Lafite, you need to take some action to get Lafite to work on your mail files and in your name. When you change the user identity by logging in as yourself, Lafite notices that the current user has changed and attempts to authenticate you. However, Lafite still doesn't know how you want your Lafite customized; in particular, what your known mail folders are.

To get Lafite to recognize your identity, you should first turn Lafite off by choosing Quit from the status window or by calling (LAFITE 'OFF). Then log in again. Type (GREET) in the executive window; you will be asked for the name of your initialization file if there isn't one on your workstation's disk. When this has been loaded, restart Lafite.

File Server Is Slow

If your mail files are stored on a remote file server that is particularly unresponsive, the mail server connection over which new mail is retrieved may *time out* (that is, end communication with your file server) before the file server acknowledges receipt of the messages. The usual consequence of this is that your in-box is not flushed, so your new mail is in two places: your in-box, awaiting retrieval, and your mail file, to which it was just retrieved. A less common occurrence is that the mail server times out partway through the retrieval process, resulting in a Lisp break. You can type ↑ after the prompt in the break window to return to the state before the Get Mail started.

If this is often a problem for you, you may want to adopt the following procedure to maintain the flexibility of remote mail files while utilizing the speed and reliability of the local disk. Keep most of your mail files on the remote server, as usual, but keep your Active Mail file, the one to which you usually retrieve mail, on your local disk. Retrieve mail to this file and dispatch from there to your remote files (using Move To) some or all of the messages you wish to keep. Mail files on disk have very predictable performance during Get Mail, which is good for both you and the mail server. Files on disk are also less subject to other vagaries of remote servers (e.g., sudden crashes) that sometimes cause problems with mail files (see figure 27). And if you tend to delete much of your incoming mail after reading it once, you may find it faster to keep your Active Mail on disk, even if your remote server isn't unreliable.

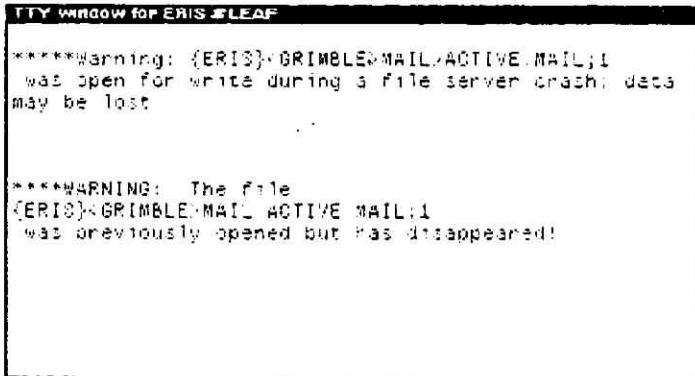


Figure 27. A warning window opened by Lafite after a file server crash. The folder has "disappeared" because the file server is down, therefore the mail folders on it cannot be accessed by Lafite

You can also choose to keep most or all of your mail files on disk, backing them up to a file server periodically. The Lisp Library package CopyFiles is helpful for doing the backup automatically.

Mail File Does Not Parse

The first time Lafite browses a mail folder, or any time that it suspects that the folder's table of contents is obsolete, it tries to parse the folder. A good mail folder consists of a set of messages, each set off by some internal information about the message's length and status. Parsing a folder consists of scanning for this internal information and building a table of contents from it. If a folder has been damaged in some way, the parsing operation fails, most commonly because a message's stated length is inconsistent with what is actually stored in the file (see figure 28). In this case, Lafite aborts the Browse command and prints a message in the browser window describing how far it got. The information Lafite prints includes the header of the last message parsed and a byte pointer

(character position) in the file where the problem was encountered. The mail file is valid up to that pointer position.

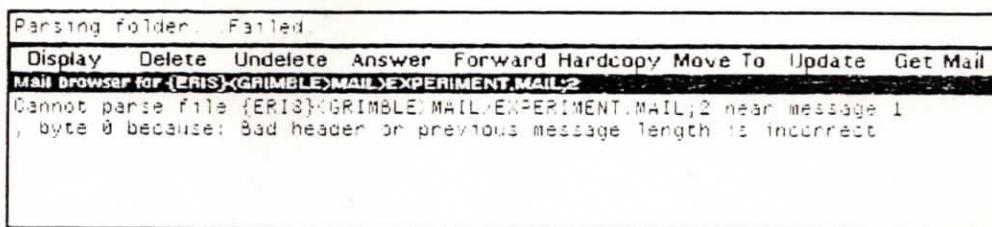


Figure 28. A browser window showing an error message printed by Lafite when it was unable to parse the mail folder

The most common way to get a mail file into an inconsistent state is to abort a Move To or Update command somewhere in the middle (either manually, or because a remote server crashed). In the case of Move To, the problem is usually that the last message in the file is "too short," since it was never completely written. If the first operation you perform on the destination file after the crash is to browse it, Lafite will (usually) detect this situation and let you browse the file anyway, with a warning that the last message is truncated. Updating the file then corrects the length of the last message. If you neglect to browse the destination file before moving additional messages to it, however, your mail file will not parse.

Another way you can damage a mail file is to get it into a TEdit file, edit it, and save it under the original name. The next time you browse that mail folder, Lafite will be unable to parse the file.

The solution to "mail file does not parse" problems is to scavenge the mail file with the Lisp Library package Mail Scavenger, following the instructions in chapter 14, "Using Lafite-Related Lisp Library Packages."

Table of Contents Is Inconsistent With Mail File

Lafite breaks with a message to this effect when it tries to operate on a message that isn't where Lafite thought it was in the file. The most common cause of this is aborting the Expunge command. The appropriate action is to close the browser, select Don't Update, delete the table-of-contents file (the file with -LAFITE-TOC appended to the name) from your directory, and then browse the file again. If this is not successful, you may need to scavenge the file with Mail Scavenger. If you made many changes to the browser (deletions, for example) that you would rather not lose, you can try selecting Write Out Changes Only when you close the browser. This may succeed if the inconsistencies in the table of contents did not intersect with your changes.

[This page intentionally left blank]

Some aspects of Lafite's behavior, such as whether each message of a batch you print starts on a new page and how many messages are shown in your out-box, are controlled by *global variables* set in your initialization file. Each variable is set to a *default value* that most users have found acceptable. However, you may want to reset some global variables to tailor Lafite to your needs. You edit Lafite variables in the same way as any other part of your initialization file, by typing (DC INIT) to bring up a DEdit window on your initialization file, resetting the variables, then typing (MAKEFILE '{FILESERVER} <DIRECTORY> INITFILENAME) to save the new version of your initialization file. For detailed information on DEdit and the MAKEFILE function, see the *Interlisp-D Reference Manual*.

Lafite's Global Variables

LAFITEDEFALTHOST&DIR

[Variable]

is the directory on which Lafite looks for the LAFITE.PROFILE and all mail folders and message forms you access if you don't supply an explicit directory. Lafite uses its own host and directory names for mail folders, etc., rather than the current connected directory because you may want to keep your mail folders someplace special (e.g., the local disk or your log-in directory), and the connected directory can change. LAFITEDEFALTHOST&DIR is provided to tell Lafite where you usually keep your mail. LAFITEDEFALTHOST&DIR should be in the same form as the variable LOGINHOST/DIR (i.e., {FILESERVER}<DIRECTORY>). If LAFITEDEFALTHOST&DIR is NIL, then the value of LOGINHOST/DIR is used—i.e., your log-in host and directory.

LAFITEFORMDIRECTORIES

[Variable]

is a search path for Lafite forms, initially NIL. When you choose the Saved Form command underneath Send Mail, the form name that you enter is first searched for on your default directory (LAFITEDEFALTHOST&DIR), and if it is not found there, Lafite searches the directories in the list LAFITEFORMDIRECTORIES. LAFITEFORMDIRECTORIES is typically set to a list of one or more public directories on which generally useful forms have been collected.

LAFITESTATUSWINDOWPOSITION [Variable]

specifies where the status window appears when Lafite is invoked. It is a POSITION or NIL (in which case you are asked to specify a position when Lafite starts).

LAFITEBROWSERREGION [Variable]

is a REGION used to describe where the primary (i.e., first) browser window is to be placed on the screen. Initially, it is set to something "reasonable" for the standard initial display configuration. If you set it to NIL, you are prompted to specify a region the first time any such window is created.

LAFITEDISPLAYREGION [Variable]

is a REGION used to describe where the primary (i.e., first) message display window is to be placed on the screen. Initially, it is set to something "reasonable" for the standard initial display configuration. If you set it to NIL, you are prompted to specify a region the first time any such window is created.

LAFITEEDITORREGION [Variable]

is a REGION used to describe where the primary (i.e., first) message composition window is to be placed on the screen. Initially, it is set to something "reasonable" for the standard initial display configuration. If you set it to NIL, you are prompted to specify a region the first time any such window is created.

DEFAULTMAILFOLDERNAME's [Variable]

value is used if no mail folder is supplied to the function LAFITE, i.e., you call (LAFITE 'ON). Initially, this value is ACTIVE.MAIL.

LAFITEMAIL.EXT [Variable]

is the default extension for names of mail folders. It is initially MAIL.

LAFITETOC.EXT [Variable]

is the string appended to the name of a mail folder to produce the name of its table-of-contents file. It is initially LAFITE-TOC.

LAFITEFORM.EXT [Variable]

is the default extension for the names of user-defined form files. It is initially LAFITE-FORM.

LISPSUPPORT [Variable]

specifies the address to which Lisp Report messages are sent. Its value is a single string containing one or more addresses, all in the form you would put in the To line of a message header. If LISPSUPPORT is NIL, then the Lisp Report form cannot be used. It is initially NIL. Site administrators may want to set this variable in their site initialization files to the mail address of their local Lisp Support.

LAFITESUPPORT

[Variable]

specifies the address to which Lafite Report messages are sent. Its value is a single string containing one or more addresses, all in the form you would put in the To line of a message header. If LAFITESUPPORT is NIL, then the Lafite Report form cannot be used. It is initially NIL. Site administrators may want to set this variable in their site initialization files to the mail address of their local Lafite Support.

TEDITSUPPORT

[Variable]

specifies the address to which TEdit Report messages are sent. Its value is a single string containing one or more addresses, all in the form you would put in the To line of a message header. If TEDITSUPPORT is NIL, then the TEdit Report form cannot be used. It is initially NIL. Site administrators may want to set this variable in their site initialization files to the mail address of their local TEdit Support.

LAFITEDISPLAYFONT

[Variable]

is a font used for displaying messages. You may change it without changing the other Lafite fonts. It should be a FONTDESCRIPTOR as returned by FONTCREATE. In your initialization file, this is usually specified with a FONTCREATE expression, e.g., (VARS (LAFITEDISPLAYFONT (FONTCREATE 'MODERN12))). Initially, it is Times Roman 12.

LAFITEEDITORFONT

[Variable]

is a font used for composing messages. You may change it without changing the other Lafite fonts. It should be a FONTDESCRIPTOR as returned by FONTCREATE. Initially, it is Times Roman 12.

LAFITEHARDCOPYFONT

[Variable]

is a font used for printing messages. You may change it without changing the other Lafite fonts. It should be a FONTDESCRIPTOR as returned by FONTCREATE. Initially, it is Times Roman 12, which prints as Classic 12 on an Interpress printer.

LAFITEBROWSERFONT

[Variable]

is the font used for displaying the table of contents in the browser window. It is initially Gacha 10.

LAFITEMENUFONT

[Variable]

is the font used for the items in all Lafite menus. It is initially Helvetica 10 Bold.

LAFITETITLEFONT

[Variable]

is the font used for the title bar ("Lafite") in the Lafite status window. It is initially Helvetica 12 Bold.

LAFITEENDOFMESSAGEFONT

[Variable]

is the font in which the LAFITEENDOFMESSAGESTR is displayed (see below). It is initially Times Roman 10 Italic.

LAFITEENDOFMESSAGESTR [Variable]

is a string containing the text of the "End of Message" token displayed at the end of a message. If LAFITEENDOFMESSAGESTR is NIL, then no "End of Message" token appears.

LAFITENEWPAGEFLG [Variable]

is initially T. If so, the Hardcopy command starts each message on a new page. Otherwise it separates each message by a line of dashes.

LAFITEHARDCOPYBATCHFLG [Variable]

allows you to postpone printing messages until several can be done at once. Printing several messages at once helps to keep your short messages from being lost amongst other users' long output, saves paper, and saves time because you don't have to wait for individual messages to be formatted for printing.

When LAFITEHARDCOPYBATCHFLG is T, Lafite batches your hard-copy requests. It doesn't actually print the messages until you do an Update, at which point Lafite sends them all to the printer in one batch. When you have hard copy pending, the Hardcopy command is speckled to remind you of this fact. The Update command has an additional choice, Do Hardcopy Only, in case you want to get your batched hard copy printed without doing an actual Update.

When batching hard copy, LAFITENEWPAGEFLG applies only to the messages selected at each Hardcopy invocation; each new set of messages starts on a new page, independent of the setting of LAFITENEWPAGEFLG. LAFITEHARDCOPYBATCHFLG is initially NIL.

LAFITEHARDCOPY.MIN.TOC [Variable]

is a positive number if non-NIL. Whenever Lafite is instructed to produce hard copy for more than LAFITEHARDCOPY.MIN.TOC messages, it also produces a table of contents as a cover page for the hard copy. Currently, this flag is noticed only if LAFITEHARDCOPYBATCHFLG is NIL. It is initially NIL.

MAILWATCHWAITTIME [Variable]

is the number of minutes between polling for new mail from your mail servers. It is initially set to five.

LAFITEFLUSHMAILFLG [Variable]

is initially T. If it is NIL, Lafite won't flush your in-box when retrieving new mail, so the mail will still be there when you invoke Get Mail again.

LAFITEIFFROMMETHENSEENFLG [Variable]

makes sure, if T, that messages sent from you are considered "seen" (and hence do not have the question mark), even though you have not yet displayed them. It is initially T.

LAFITEDISPLAYAFTERDELETEFLG

[Variable]

can be set to T or ALWAYS. If T, Lafite displays the next message if you delete the message in the message display window and the next message is undeleted and unexamined (i.e., it is marked with a question mark). If ALWAYS, Lafite displays the next undeleted message even if it has already been seen. It is initially T.

LAFITEMOVETOCONFIRMF LG

[Variable]

controls whether Lafite requires confirmation of the Move To command. If ALWAYS, all moves require confirmation; if LEFT, then only left-button moves (selecting the destination from a menu) require confirmation; if MIDDLE, then only middle-button moves (using the default Move To folder) require confirmation; if NIL, then no moves require confirmation. It is initially ALWAYS.

LAFITEBUFFERSIZE

[Variable]

is the number of 512-character buffers used by the stream managing the file behind an open browser window. If you regularly receive very long messages, you might want to increase this to improve performance of Display followed by Hardcopy or Move To. Initially 20, which handles messages up to about 10,000 characters long.

LAFITEOUTBOXSIZE

[Variable]

specifies the number of delivered messages to be retained in your out-box. As you send more messages, older ones fall off the end. Increasing this number gives you a longer "history" from which you can select and reedit old messages, but the desire for a longer history should be balanced with the knowledge that you are tying down the resources used by each of those messages. Setting LAFITEOUTBOXSIZE to zero or NIL disables the out-box feature: after delivery, messages completely vanish. LAFITEOUTBOXSIZE is initially two.

LAFITENEWMAILTUNE

[Variable]

is a list of the form acceptable to the function PLAYTUNE or NIL, in which case it is ignored. It is played when Lafite discovers you have new mail waiting.

LAFITEGETMAILTUNE

[Variable]

is a list of the form acceptable to the function PLAYTUNE or NIL, in which case it is ignored. It is played when a Get Mail command completes.

How to Add a New Message Form to Lafite

The normal way to add a new message form to Lafite is to edit an existing form (or build one from scratch) and save it using the

Save Form menu item (see chapter 9, "Writing Messages"). However, you can also provide message forms that compute the text at the time you request the form, as, for example, the Lisp Report does. To add your own items to the Message Forms menu, add a standard three-element menu item to the variable **LAFITESPECIALFORMS** and then set the variable **LAFITEFORMSMENU** to NIL (this is where the menu is cached). The three-element menu item should yield a LITATOM as its "value," that atom being interpreted as follows:

1. If the atom has a function definition, the function is called (with no arguments) and the returned value (a string or a TEdit text stream) is used.
2. If the atom has a value, its value (a string or a TEdit text stream) is used.
3. Otherwise, a copy of the file by that name is used.

For example, if you wanted to add a message form that contained the date the user's version of TEdit was made (as in the TEdit Report form), you could add

`("TEdit Support" (QUOTE MAKETEDITFORM))`

`"Make a form to report a problem with TEdit")`

to **LAFITESPECIALFORMS**; **MAKETEDITFORM** could be defined to be

```
[LAMBDA NIL
  (PROG (OUTSTREAM)
    (SETQ OUTSTREAM (OPENTEXTSTREAM " "))
    (printout OUTSTREAM "Subject: TEdit: >>Subject<<" T)
    (printout OUTSTREAM "To: " TEDITSUPPORT T)
    (printout OUTSTREAM "cc: " (USERNAME T)
    (printout OUTSTREAM "TEdit-System-Date:"
      TEDITSYSTEMDATE T T)
    (printout OUTSTREAM ">>Message<<" T)
    (RETURN OUTSTREAM)])
```

where **TEDITSUPPORT** and **TEDITSYSTEMDATE** are variables set by TEdit. Lafite supplies one function to make this kind of message form easier to construct.

`(MAKEXXXSUPPORTFORM 'SYSTEMNAME
 'ADDRESS 'SYSTEMDATE) [Function]`

creates a message form (a TEdit stream) to be mailed to the maintainers of **SYSTEMNAME**. **SYSTEMNAME** is the name of the subsystem (a string); **SYSTEMDATE**, if non-NIL, is a date (string) of importance to include in the message; and **ADDRESS** is the mail system address of the intended recipient(s) or an association list in the same form as the variable **LISPSUPPORT**. If the **ADDRESS** is NIL, **MAKEXXXSUPPORTFORM** prints a message in the prompt window that the form is not supported, and returns NIL.

For example, MAKETEDITFORM is actually defined as

```
[LAMBDA NIL  
  (MAKEXXXSUPPORTFORM  
  "TEdit" TEDITSUPPORT TEDITSYSTEMDATE)]
```

and selecting TEdit Report in the Message Forms menu produces a form like the one in figure 29.

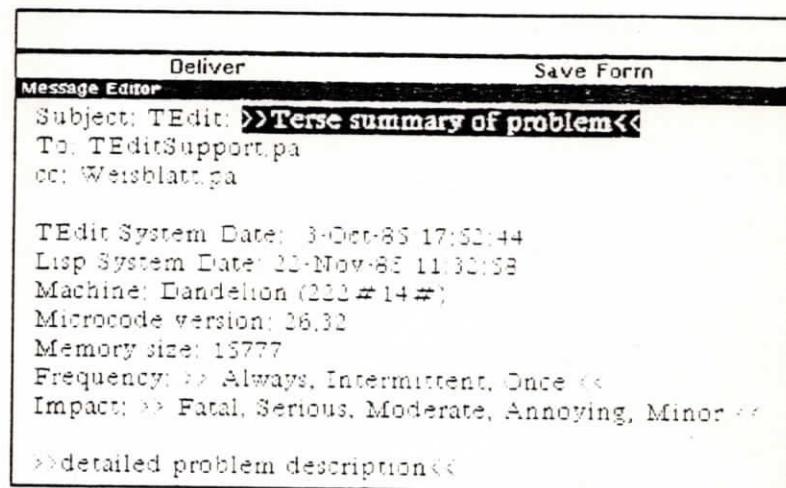


Figure 29. The TEdit report form that was created by following the instructions given above

[This page intentionally left blank]

There are two separate packages that can be used with Lafite: Lafite Find and Mail Scavenger. Lafite Find helps you search for particular messages in a mail folder, and Mail Scavenger helps to restore mail files that have been damaged.

Lafite Find

Lafite Find is a package that helps you search for particular messages in a mail folder. The search is a simple string or keyword search that examines either specific header fields of a message (From, Subject) or the entire message. You can look for either one message or all messages matching a requested pattern.

How to Load Lafite Find

You load Lafite Find by typing (LOAD '{FILESERVER} <DIRECTORY>LAFITEFIND.DCOM). You can then bring up a menu of search commands by clicking on the black title bar of a Lafite browser window with the middle mouse button (see figure 30). Three commands are listed on this menu: Find, Find Related, and Find Again. The Find command is the general search command; the other two are special cases.

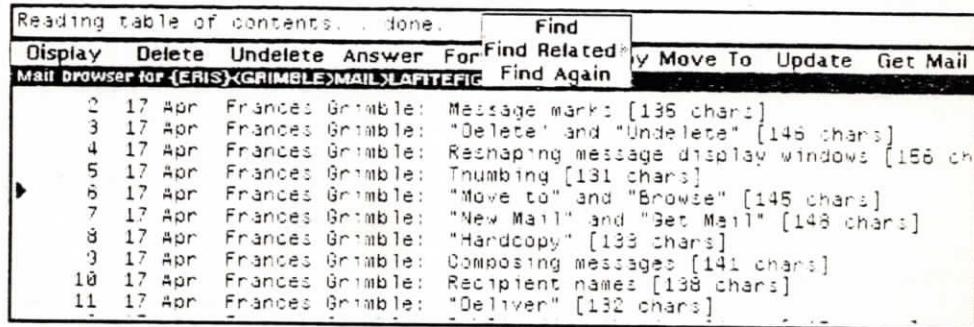


Figure 30. A browser window in which the user has brought up the top-level Lafite Find menu. You can bring up the menu anywhere on the title bar by clicking the middle mouse button

Search Direction Commands

If you select the Find command in this first menu, you are prompted with a menu to determine the direction of search. It lists four options: Find Next One, Find Next All, Find Previous

One, and Find Previous All (see figure 31). To search forward in your mail folder one message at a time, use the Next One command. To search forward for all appropriate messages, use Next All. The Previous One and Previous All commands search backward. The Next searches all search forward from the last selected message in the browser; the Previous searches all search backward from the first selected message.

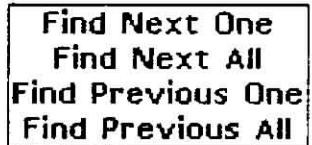


Figure 31. The search direction commands menu

The Find Next All and Find Previous All commands start the search with the currently selected message, unlike the Find Next One and Find Previous One commands, which omit the currently selected message. This is mainly so that the Find Related command includes the original message in the set of messages it selects.

Search Type Commands

Selecting any of the above commands brings up another menu, prompting for the type of search. This menu lists four options: From, Subject, Body, and Related (see figure 32). The From and Subject options enable you to search for messages by the content of their From or Subject fields, the Body which examines the entire message, and the Related search a special kind of Subject search.



Figure 32. The search type commands menu

The From and Subject Searches

The From search looks at the From fields of messages sent by others and the To fields of messages sent by you. Thus, roughly speaking, From searches look at the same names you see in the browser window between the date and subject of each message. The Subject search examines the Subject fields of the messages in your mail folder.

The Body Search

The Body search can be used to locate a message by arbitrary strings in the message itself, rather than just in its From or Subject fields. For example, a Body search can locate messages that are addressed to (or cc'ed to) "Jones," or that were sent in the month of "Feb," or that mention "sugar" in their content. However, such searches are potentially much slower, since they have to search much more territory.

The Related Search

The Related search is a special kind of Subject search. It looks for messages whose subject is the same as the currently selected message, plus or minus the "Re: " that the Answer command adds to a subject (see chapter 9, "Writing Messages"). The easiest way to perform a Related search is to use the Find Related command in the initial menu, which is equivalent to the sequence Find, Find Next All, Related. If you want to do a Related search backward, the Find Related command has a subcommand Find Related Backward, which you can access by rolling the mouse cursor out of the menu to the right.

Find Related does not check that the messages it finds are really related to the selected message, merely that the subject matches the search string formed by taking the subject of the selected message and, if necessary, removing the initial "Re:." Thus, this command may find too many messages.

Find Related is very useful for locating all the messages in an interchange where everyone used the Answer command to reply to the first (or subsequent) message.

How to Use Lafite Find

After you select the type of search (except for Related), you are prompted to type a string to search for. The search is case-insensitive; e.g., "Jones" matches both "JONES" and "jones." The string searches are also substring searches, so you need not type the entire user name in a From search, just enough to uniquely identify the sender.

For example, to search for the next message sent by user Jones, click Find, then Find Next One, then From. Lafite then prompts you in the prompt region with "Find From string:." Type the name of the sender (Jones) and then press the carriage return. If Lafite finds a message matching the string you typed, it selects the message and scrolls the browser, if necessary, to expose that message. If it finds no such message, it tells you so in the prompt region.

If you want to find all the messages from Jones, then follow the same procedure, but choose Find Next All instead of Find Next One. Lafite Find selects all the messages from Jones, and you can then cycle through them with the Display command, delete them all, move them all to a different folder, etc.

If the message found by a Find Next One or Find Previous One command is not the message you were looking for, or you want to locate the next such message, you can repeat the search by selecting the Find Again command in the initial (middle-button title bar) menu.

Mail Scavenger

The Lisp Library package Mail Scavenger is used to rebuild the internal pointers in a mail file that has been damaged (see chapter 12, "Troubleshooting Lafite Problems"). When Lafite detects damage in a file, it usually reports "Can't parse file" and terminates its Browse command. The simplest remedy is to scavenge the file with the Mail Scavenger, then browse the file again.

How to Use Mail Scavenger

To load the Mail Scavenger, type (LOAD '{FILESERVER}<DIRECTORY>MAILSCAVENGE.DCOM) (see figure 33). To call the Mail Scavenger, type (MAILSCAVENGE 'FILENAME). This scavenges, or attempts to fix, the file named *FILENAME*. (*FILENAME* defaults to the extension MAIL and your Lafite directory, exactly as with Lafite's Browse command.) Mail Scavenger will print some information in your executive window about what it is doing to correct the file, such as "Patching length field of header in message number 1." It will then ask, "Do you want to replace {FILESERVER}<DIRECTORY>MAILFILE with the newly scavenged version?" If you answer Yes, the newly scavenged file will replace the damaged mail file. If you answer No, a new version of your mail file will be created. In this case, Mail Scavenger returns the name of the temporary file it wrote, ({DSK}<DIRECTORY>FILENAME.SCAVENGE), which you can then rename or delete as you wish. Ordinarily you should reply Yes, unless you are suspicious about Mail Scavenger's report on what it had to correct.

The finished mail file that Mail Scavenger produces should always be readable; i.e., Lafite will not complain about it. However, it is a good idea to browse the file and check any messages the Mail Scavenger mentioned correcting; these may be missing several characters or be malformed in other ways. You should also check neighboring messages—some of the characters in these messages might really be parts of other messages, and some messages may be duplicates. You may want to delete the damaged messages.

```

Interlisp-D Executive
NIL
17-(LOAD "MAILSCAVENGE.D0OM")
=
{ERIS-LISP}INTERMEZZO:LIBRARY\MAILSCAVENGE.D0OM;1

{ERIS-LISP}INTERMEZZO:LIBRARY\MAILSCAVENGE.D0OM;1
compiled on 21-Feb-85 17:50:30
FILE CREATED 21-Feb-85 17:49:29
MAILSCAVENGE.D0OMS

{ERIS-LISP}INTERMEZZO:LIBRARY\SEARCH.D0OM;1
compiled on 7-May-84 23:20:16
FILE CREATED 7-May-84 23:19:59
SEARCH.D0OMS
{ERIS-LISP}INTERMEZZO:LIBRARY\MAILSCAVENGE.D0OM;1
22-(MAILSCAVENGE 'EXPERIMENT)

Rebuilding header for message number 1
Do you want to replace the mail file {ERIS-GRIMBLE\MAIL\EXPERIMENT.MAIL;2 with the newly-scavenged version? yes
{ERIS-GRIMBLE\MAIL\EXPERIMENT.MAIL;2
23+

```

Figure 33. An executive window in which the user has loaded the Mail Scavenger and scavenged the folder Experiment.Mail

How to Scavenge Files in Place

Calling (MAILSCAVENGE.IN.PLACE '*FILENAME*) is similar to calling MAILSCAVENGE, except that it destructively scavenges the file in place. This is faster than MAILSCAVENGE, but you have to be brave and assume MAILSCAVENGE is not overwriting anything valuable as it scans the file.

Additional Arguments to MAILSCAVENGE

If you wish, you can call the MAILSCAVENGE-function with the arguments *ERRORMSGSTREAM* and *TEMPDIR* as well as *FILENAME*. *ERRORMSGSTREAM* is the stream on which Mail Scavenger writes the information about what it is doing; the default is T. You can also specify the argument *TEMPDIR*, which is the host/directory on which Mail Scavenger writes its intermediate file. *TEMPDIR* defaults to {DSK} unless you are on a workstation without a local disk file system, in which case *TEMPDIR* defaults to the same directory as *FILENAME*.

[This page intentionally left blank]

The great art of living easy and happy in society is to study proper behaviour, and even with our most intimate friends to observe politeness; otherwise we will insensibly treat each other with a degree of rudeness, and each will find himself despised in some measure by the other.

Boswell, *London Journal* (1762)

Foreward

This appendix is an edited version of "Message System Mores," chapter 6 of the Xerox Laurel Manual, by Douglas K. Brotz, and the essay "Message System Mores" that Brotz published in *ACM Transactions in Office Information Systems*, Vol 1, No. 2. The material that appeared only in the ACM journal is copyright 1983 by the Association for Computing Machinery, Inc.

Douglas Brotz was a member of the team at the Palo Alto Research Center (PARC) that designed Laurel, which is an electronic message system similar to Lafite that was written for the Xerox Alto. Through his involvement with Laurel, Brotz discovered patterns of electronic message system behavior that may apply to Lafite users. He also developed some rules for appropriate message system behavior, i.e., message system etiquette. Because many Laurel and Lafite users have found this essay helpful, we have edited it for inclusion here, making the references appropriate for Lafite and deleting information that appears elsewhere in the Lafite manual.

Introduction

This is an essay on manners, in particular, message system manners. Electronic message systems provide a new mode of communication that, while offering convenience, speed, and reliable delivery, also opens channels that may be abused. At the Xerox Palo Alto Research Center, we have designed and implemented an electronic message system that has quickly spread throughout the Xerox Corporation. Through its use, we have discovered many patterns of message system user behavior that appear to apply to electronic message systems in general.

rather than to the particular system that we built. The focus of this essay is not on the features of our system, but on observations of user behavior in the electronic mail environment in an effort to spread understanding of this new medium and to instruct users in proper behavior.

The contents of this essay may be divided into roughly two kinds, objective observations of message system social phenomena and definitely biased suggestions of standards. The opinions expressed herein are solely those of the author. These opinions are not based on scientific studies or samples, but rather on certain intuitive feelings that have evolved through a close association with our system since its inception.

Communication Patterns

Part of the evolution of a society is the structure within which its members communicate. Face-to-face communication, both spoken and through gestures, has been with us for a very long time. Written communication and telephone communication have been employed for a substantially lesser amount of time. Nevertheless, these modes of communication have been around long enough to have developed certain standards of conduct and a framework in which reasonable communication can take place.

The electronic message medium has existed for a much shorter period of time, perhaps 20 or so years. (I am purposely ignoring telegraphic communication, which has very different characteristics due to its long delays and high cost.) Electronic message systems on personal computers have been available for even less time, probably less than 10 years. In this time, standards of electronic communication have not yet had time to mature, so we are still groping toward a workable electronic-messaging society.

In any of the mature communication media, each society places limits on what is considered acceptable behavior. Vulgar language or gestures are generally frowned upon in face-to-face communication, except in smaller sub-societies in which this mode of behavior is necessary for group membership. Shouting at close range is similarly considered to be in bad taste. Methods of dealing with such behavior in face-to-face communication run from mild rejection of the speaker to complete avoidance of that speaker in the future. As the number of human societies is large, and each has had much experience with this means of communication, the means employed for dealing with such situations are quite varied. Within each group, however, the methods used can be quite effective in stifling unwanted behaviors.

I will list several kinds of situations that arise in the electronic message medium and means for dealing with them. Where

possible, I will draw parallels to other more traditional modes of communication to illustrate acceptable manners. In addition, I will try to point out the ways in which communicating via electronic mail is different from the traditional communication media, and how this modifies the problems to be dealt with.

The Wrong Number

We all have dialed wrong numbers and received calls from people who have dialed wrong numbers. The protocol for handling such situations is simple, and arises naturally as a result of the way in which standard phone calls are initiated. A typical wrong number dialog may be as follows:

Callee: Hello.
Caller: Hello. May I speak to John?
Callee: There is no one at this number by that name. I believe you have the wrong number.
Caller: Oh. Isn't this 555-1234?
Callee: No, it isn't. (And sometimes . . .) This is 555-4321!
Caller: Thank you. I'm sorry to have bothered you.

In postal communication, receiving misaddressed mail or mail for a former resident who has moved is akin to the telephone's wrong number. The post office's suggested remedy is for the recipient to line out the address and remail the letter. The post office will then attempt to forward the letter to the correct address, deliver it to the proper address, or return the letter to the sender.

Note that in both of these situations, it was not necessary to begin the actual conversation or open the letter. Enough information is exchanged at the outset to determine if the parties in the communication are the correct ones. This is usually not true when communicating via electronic mail.

In electronic message systems, it is seldom the case that a message sent to a particular name is actually delivered to a recipient with a different name. A different situation is (unfortunately) common when a recipient has a popular name. The problem is that several people may have the same last name, and some electronic message systems do not have convenient facilities for mapping a person's actual name into that person's message system name. Thus, a person named Doe may receive mail for ADoe, BDoe, etc. Here, the original error is committed by the sender, who did not consider that ADoe's message system name was actually ADoe, but just assumed that it was Doe.

The parallel to this situation in the telephone medium is actually a bit more elaborate than the dialog given above. It is more like:

Callee: Hello.
Caller: Hello. Is Johnny there?

Callee: Hold on, I'll get him.
John: Hello?
Caller: Hey Johnny, let's boogie on down to the hoedown.
John: Who is this?
Caller: Come on buddih, this is good old Bodine!
John: I don't know any Bodine.
Caller: Oh. Ain't this 555-1234?

and so on. Notice that in this case a partial name match has occurred, and it is only later in the conversation that one of the parties discovers that something is awry. In the electronic mail case, it is nearly always the case that the message must be at least partially read to determine that it has reached an incorrect recipient.

This situation can be (and has been) handled in several inappropriate ways. First (and worst), the incorrect recipient can just ignore the message. No one gains through such inaction. Second, the incorrect recipient may send a response to the sender of the form "Stop sending me this trash!" This is a bit more helpful, but not quite the best that can be done. Third, the incorrect recipient may send the correct recipient a message of the form "Tell your senders what your name is!" This is not even as good as the previous response, as a message system user cannot know all possible senders.

Proper consideration by all involved can alleviate the "wrong number" syndrome considerably. Senders of messages should know their recipients. When sending a message, if you are not sure of a person's message system name, look it up. At Xerox PARC, the phone list has everyone's message system name correctly listed. Perhaps other organizations should do the same, and eventually a message-system-wide "white pages" may be published. Such lists help, but not if the senders don't use them.

A message addressed to an individual tends to be more important than a message addressed to a distribution list, in that a reply from an individual is expected more than replies from anonymous members of distribution lists. The names contained in distribution lists are usually correct, so there is generally no misdelivery problem. However, senders type the names of individual addressees for important messages directly. Thus, when there is a misaddressed message, it is generally an important one.

When you realize that a message is not for you, use the Forward command to send it back to the sender along with your polite comment that the message has reached a "wrong number." Forwarding the message back is important, as the sender may not have a copy of that message any more. Once you have determined that you have received a "wrong number" message, stop reading it. A message sent through the message system may have personal material, and it is none of your business to peruse the entire message. (Many users who typically dispose of their received mail at a rapid clip take great delight in reading every last character of a misaddressed message—indeed, they consider it their solemn duty to do so.) It is for this reason that I do not

suggest forwarding the message to the proper recipient. Determining who is the proper recipient is the job of the sender. It is presumptuous to believe that you know who the proper recipient is; you may actually forward the message to yet another incorrect recipient. Besides, determining the correct recipient may require reading more of the message than you ought to read. (If you think you know the message system name of the correct recipient by the time you realize that you are not the correct recipient, then you might include that name in your short covering note back to the sender. However, the mistaken sender should not expect correct identification of the intended recipient, just as he or she would not expect it in the telephone or postal mail systems.)

Some further points to consider are these. The "wrong number" mishaps generally happen to people who have common names and whose system names are exactly their last names. The honor of having one's system name be exactly one's last name is generally historical ("I was the first Doe hired here, therefore I'm entitled to be Doe.PA forever!") A reasonable solution would be that our system administrators ensure that no user has a name that is a suffix of another's, so that when ADoe arrives, then Doe has his or her message system name changed to BDoe, or whatever. In this way, the existing message system facilities will catch messages sent to Doe and return them as having been sent to a nonexistent name, at which point the sender can look up the correct message system name.

Rudeness and Vulgarity

The electronic mail medium joins several disparate properties of other communication media in an interesting way. The display of mail on a personal computer is a rather personal experience. Certain feelings of privacy and ownership pervade a personal computer user's relationship with his or her machine. Thus, the process of reading one's own electronic mail includes many of the personal aspects of face-to-face communication.

On the other hand, sending electronic mail is much more impersonal. The recipient is not present, and nearly none of the social strictures that govern one's face-to-face communication are present. The sender is also able to speak his or her piece completely, without any intervening exchanges with the recipients that might moderate the entire business. This situation is enhanced when the recipients are not named directly, but are addressed indirectly through an impersonal distribution list. This imbalance in attitudes between sender and recipient has wide-ranging consequences.

One obvious consequence of this imbalance is that opinions expressed and the language used to express them in messages can be wildly inappropriate to the customs and expectations of

the recipients of such a message. A reader may justifiably feel slapped in the face by a message he or she considers to be in extremely bad taste.

An interesting feature of the most annoying messages is that they tend to come from some "other" part of our messaging community. Julian Orr at PARC has observed that the most serious exchanges of antagonistic messages in our network have occurred shortly after some previously isolated message communities have joined. When the two societies meet and exchange messages, for some period the tone of the "other" community's messages has offended members of "our" community. After an adjustment period, the two communities come to some understanding and establish norms for their intercommunication. This understanding typically involves identification of subjects whose discussion will cease to cross community boundaries.

The development of the junk mail lists in our Palo Alto and El Segundo, California registries illustrates this point. Both communities established Junk ↑ distribution lists for people interested in any for sale ads, announcements, random comments, etc. A slow link between these two locations was replaced by a faster one, and the volume of message traffic between the two communities increased dramatically. The two lists, Junk ↑ .PA and Junk ↑ .ES, were combined into an AllJunk ↑ list. However, even though the stated purpose of both Junk ↑ lists was "anything goes," many PA registrants felt the ES junk mail was beyond the bounds of good taste. In other words, "their junk is much worse than our junk." A majority of the original members of these lists withdrew from the Junk ↑ lists, and many splinter lists developed, ranging from Whimsy ↑ .PA ("lighthearted mail for PA") to @CrankMail.DL (a widely used private list in El Segundo). This is a history of two communities that seem to have similar characteristics. The implications for new message networks linking quite disparate communities are that more serious problems are likely to develop.

When rebuked for inappropriate behavior, errant senders have been known to say "I didn't intend it that way!" This is not good enough. The damage has already been done. The only remedy is for senders to think about what they are saying and to whom they are saying it. The message system to date has been fairly unrestricted. Only as long as the society of message system users practices self-restraint will such a freewheeling communication medium be tolerated. There are several means of applying institutional censorship to the message system traffic, means that we hope will never need to be implemented.

Message System Costs

Many of the problems associated with improper use of the message system are exacerbated (caused?) by the lack of charging for message system usage. In nearly all other modes of communication, "sending a message" implies a certain cost (or risk), which rises with the number of recipients that are being reached. Free speech is, in this sense, not free at all. Certainly in a free society, one can say what one pleases, but not without paying for the means to say it. Let me illustrate this with some examples.

In nearly every communication medium, costs for the use of that medium are borne by the sender of messages. Postal mail requires the sender to pay for a stamp for each copy of a message that is sent. Telephone service is charged to the originator of calls, and each call (in general) goes to only one recipient. Broadcasting messages via radio or television requires a large investment on the part of the sender. The costs of printing handbills or posters are likewise borne by their authors. Public speeches, if they are to reach a large audience, require use of sound systems, etc., that are paid for by the speaker.

It may be argued that recipients do pay some of the costs for using some of these systems. However, these costs (the price of a radio receiver, basic telephone service, etc.) are generally constant; they do not increase as received message usage increases. A receiver's cost for electronic mail is similar in this respect in that the cost of a workstation on which the message system runs is borne by the receiver.

Some other modes of communication do require explicit payment by the receiver. Commercial films, books, magazines, and records fall into this category. However, publication of these materials does involve a substantial financial risk. Material that is not likely to be well received is seldom published, and when it is, large costs are often incurred by the publisher.

Electronic mail as it is usually implemented has a very different cost structure. The cost for a sender is minimal. It essentially consists of the time it takes to compose and send a message. If time is considered the major cost factor, then it is the recipients who pay dearly for the messages they receive. When the amount of time each recipient spends on a message sent to a large distribution list (even if a quick scan of part of the message followed by a Delete), is summed over all recipients, this is easily much more than the time consumed by the sender of that message.

While we would like to keep the free structure of a message system, where any user can send any message to any other users, this freedom must be used with some care. When electronic message systems become widespread, they will undoubtedly change their cost structures to match those of the more traditional communication systems.

Unsolicited Mail

The existence of large public distribution lists in our message system makes it easy for a sender to reach a very wide audience. Each distribution list has a distinct purpose, e.g., lists of people interested in particular topics, lists of employees in certain organizations, lists of members of particular projects, etc. Some lists are used primarily to keep track of all users of the message system. These include such lists as AllPA↑.PA, AllES↑.ES, etc., which contain the names of all individuals in those particular registries. There are also some lists maintained on a purely geographical basis, e.g., PaloAlto↑.PA, which lists all message system users in Palo Alto, California. This is not necessarily the same as AllPA↑.PA, which includes people in the PA registry, but who may not actually work in Palo Alto.

The audiences addressed by these lists should not be considered captive audiences for all users of the message system. At Xerox PARC, the purpose of any distribution list may be discovered by any user by running the Maintain program. Although all lists are (currently) available for use by any message system user, many lists, e.g., AllN↑.N where N is a registry name, should not be used by anyone who doesn't have a very good reason for doing so.

Many distribution lists exist for the enjoyment of their members who wish to receive items of interest to them. One should feel free to send an announcement of an upcoming musical event in Northern California, for instance, to Music↑.PA. Such a message is quite inappropriate to send to AllPA↑.PA, PaloAlto↑.PA, etc. There are lists of message system users who have agreed to suffer through any and all messages. These lists (Junk↑.PA, various @CrankMail.DL files, etc.) are the only lists to which ridiculous messages may be sent without incurring the justifiable wrath of message system users.

A message system user should understand when a message is appropriate to send to all people in his or her work group. Social values are different in different locations, and the members of each group should understand what they are. It has been observed that messages that are sent to audiences wider than the sender's immediate group are the ones that cause the most trouble.

Unfortunately, unsolicited messages have continued to be sent to inappropriate lists. Examples of inappropriate messages for standard organizational or geographic lists are:

"Does anyone know how to get my Alto fixed?"

"This is to let everyone in the message system world know that my phone number has changed."

"I want everyone to know that I really like my roofing contractor."

I'm sure that each user of the message system can recall some other similar gem. The following sections explore some of the consequences of unsolicited mail.

The Chain Reaction

To add insult to injury, after some piece of particularly ridiculous mail has been broadcast to an inappropriate audience, it invariably follows that some recipients cannot control their urge to make even bigger spectacles of themselves by sending their two cents' worth to everyone who received the original nonsense. While the original event is thought by many message system users to be annoying, the latter is considered to be downright stupid. Remember that once you push the Deliver button and watch the last chance to cancel fade away from your screen, there is no way to erase your comments from the collective memory of your peers.

I would like to list some of the typical responses that have been sent not just to the original perpetrator, but to the entire list of victims.

"Your message is inappropriate to send to all these good people."

"If you don't like junk, then get off Junk ↑."

"How do I get off Junk ↑?"

and, my favorite,

"Do you realize that if all of us replied to all of us (as I am doing right now) that the number of messages that would be sent would exceed the number of atoms in the known universe . . ."

It is my opinion that bombarding only the original sender of a ridiculous message with equally nonsensical replies is poetic justice.

Use the Reply-To field to counteract the chain reaction phenomenon. However, there are situations in which replying to the entire list of original recipients is appropriate. In these cases, send the message without a Reply-To field, so that recipients who use Answer will get forms with all recipient names and lists included as recipients.

One final note on this topic. Although Lafite provides these mechanisms to help break chain reactions, the ultimate responsibility for messages sent lies with their senders. Always check the list of recipients in any message you are about to send.

Off-the-Record Responses

There are many situations in which a user submits a question to a wide audience, say to a distribution list of people interested in such questions, and indicates that he or she will collect responses and later make them public. This is a most reasonable thing to do, and it helps to reduce the chain reaction effect. Be sure to include a Reply-To: >>Self<< field when performing such services for your audience.

A note of caution is in order here. Messages should be considered private unless otherwise indicated. If your intention is to publish the responses, then by all means make that intention clear in the same message that poses the original question. If your message did not make that intention clear, and you decide that you would like to publish the responses, then follow up on each response asking whether you may do so.

If the intention to publish responses is clearly indicated in the original message, then publication of any response is fine, as long as that response does not explicitly mention that it should be considered private.

Masquerading

On occasion, people have received messages from fictitious senders, or even worse, from someone masquerading as another real message system user. This is a most serious breach of message system etiquette, and should be considered so by all message system users.

A fictitious From field is legitimate when a valid Sender field is included. For instance, messages that are properly signed with an organization's name, say "The Lafite Group," may be sent by explicitly typing a "From: The Lafite Group" line in the message header. Lafite will notice that a From field is already there, and it will include a Sender: User Name line in the delivered message instead of its usual From: User Name line. Any time you receive a message that has a strange From field, you may check the Sender field for the actual sender.

By a "masquerader" I mean someone who subverts the normal mechanisms embedded in the standard message system programs to send messages of dubious value, without having his or her name appear in such messages. This action is possible not only in electronic message systems, but in other more traditional communication media as well. Masquerading as another may be a criminal act when committed using traditional communication media, with penalties specified in laws that prohibit libel, slander, and fraud. Other situations, such as telephone "breathers," are similarly outlawed.

Masquerading is the most serious social problem that we deal with in our message system. It occurs very rarely (three times to my knowledge), but when it does, our message system administrators make every effort to discover the perpetrator. The consequences may be serious, and the discovered perpetrators have all apologized publicly. I quote without reference from one such apology made by a masquerader who saw the error of his ways: "It is clear that any abuse of the message system, however lighthearted the intent, has repercussions far beyond what one might expect."

At this time, I do not know of any court cases involving libel, slander, etc., in an electronic mail context. Such cases are sure to arise when electronic mail does become more widespread. Masquerading in the message system is not cute or clever. Don't do it.

Wizards Versus Naive Users

This section is addressed mainly to the wizards who should know better. The population of message system users covers a broad range, from those who have knowledge of the most arcane details of a system to those who just barely understand the basics of using that system. When you send a message to a wide audience, be considerate of the naive users, who may be confused by technical jargon.

This admonition extends to those who are using a new, restricted program. It does not help a recipient to hear "Oh you're using that old program. Well, I guess you're stuck." Just don't mention such things to users who cannot take advantage of them.

The Moral of This Tale

The moral of all this is simple: Be considerate. As we strive toward this goal, everyone's use of the message system will become even more of a joy than it already is.

[This page intentionally left blank]

The main part of this manual tells you how to use Lafite at client companies. However, Lafite may be used somewhat differently at the Xerox site where you work. For example, you may use a message transport system known as Grapevine instead of the NS system; or you may use both systems. Or your site may use both Lafite and Laurel (a mail system similar to Lafite that was written for the Xerox Alto). This appendix gives you the specialized information you may need to use Lafite within Xerox.

Lafite use at Grapevine sites is described first. This section includes the documentation for Maintain, the program used to maintain Grapevine public distribution lists. It also tells you how to send Lafite messages to ARPANET recipients. Next described is Lafite use at sites having both the Grapevine and NS systems. Then there is a short description of some Lafite-related packages currently available only within Xerox. The final section describes the differences between Laurel and Lafite.

Note: this appendix was not indexed due to the difficulty of creating and maintaining separate customer and internal indexes.

The Grapevine Mail System

For about five years most Xerox sites have used the Grapevine electronic message transport system; it was for this system that Lafite was originally developed. At the present writing, Xerox is in the process of converting from Grapevine to the NS system. Some sites use only NS mail, some Grapevine, and some both. For the most part, you use Lafite with the Grapevine system in exactly the same way as with the NS system. The four major differences are the mail protocols, the forms of recipient addresses, the ability to send mail to ARPANET recipients, and the way in which public distribution lists are maintained.

Grapevine Protocols

The Grapevine system uses its own set of mail protocols, which you can obtain by loading the Library package file MAILCLIENT.DCOM. Simply type (LOAD '{FILESERVER}< DIRECTORY >MAILCLIENT.DCOM) in the executive window.

Grapevine Recipient Addresses

A Grapevine recipient address has two parts separated by a period, in the form *Recipient.Registry*. The first part is the identifier for the recipient, usually that person's last name. If there are two or more people with the same last name in the same registry, each is usually identified by his or her first initial plus the last name.

The second part of a Grapevine address is the registry name. A registry is no more than a device for grouping related names. The registry name helps the mail system determine which mail server has the in-box for the recipient. Most registry names correspond roughly to "campuses" of activity within Xerox, which should make them easier to remember.

Lafite allows you to omit the registry name for recipients who are in your registry. You can find out what your local registry is by typing DEFAULTREGISTRY at the prompt in your executive window. DEFAULTREGISTRY is the global variable that names your local Grapevine registry; it is used if your log-in name does not include a registry. This variable is normally set in your initialization file.

Grapevine Registries

At present, the following registries exist:

ARPA	A pseudoregistry used to forward mail to people on the ARPANET and points beyond.
AUTO	A registry for software-driven functions requiring automatic access and Grapevine authentication.
CHOLLA	A registry for PARC/ICL, Palo Alto, California (no individuals in this registry).
DC	A registry for the District of Columbia.
DLOS	A registry for Dallas, Texas.
EOSA and EOS	A registry for associates or affiliates.
ES	A registry for El Segundo, California.
FX	A registry for Fuji Xerox, Japan.
GUEST	A registry for Palo Alto, California.
HENR	A registry for Henrietta, New York.
LB	A registry for Leesburg, Virginia.
OGC	A registry for the Office of the General Counsel.
OSDA and OSD	A registry for associates or affiliates.
PA	A registry for Palo Alto, California.
PASA	A registry for Pasadena, California.

RX	A registry for Rank Xerox, Great Britain.
SIEMENS	A registry for the Communications and Information Systems Group, Private and Special-Purpose Networks Division at Xerox XSG, OSD, Palo Alto, California.
STHQ	A registry for the Xerox Corporate Headquarters, Stamford, Connecticut.
SV	A registry for Sunnyvale, California.
WBST	A registry for Webster, New York.
X	A registry to be used by people in many geographic locations.
XRCC	A registry for the Xerox Research Centre of Canada, Toronto, Canada.

Grapevine Distribution Lists

The Grapevine system supports over 1,500 distribution lists, some of which are work related and some of which are purely for fun. You can obtain a complete distribution list directory, giving the name of each list plus a short description of its purpose, by printing {Indigo}<Registrar>GV>DLMap.Txt.

Grapevine distribution list recipient names end with the character "↑," in the form AISBU↑.PA.

The public distribution lists for each registry are stored on mail servers and are maintained by the administrators of that registry, by the owners of the lists, and in some cases by the members of the lists. To create a public distribution list you must contact an administrator for your registry, who will make sure that your proposed name does not conflict with any others, and who will create the list for you. You can get your name added to appropriate lists by contacting the owners of those lists or by using the Maintain program (see below). If you cannot make a desired change to a public list yourself, send a message to the owners of the list. In Grapevine registries, by convention, the owners of a list named *List↑.Registry* are listed in the companion list *Owners-List↑.Registry*.

Delivery to Distribution Lists. If a recipient list includes public distribution lists, you must confirm the delivery by including a Reply-To field in the message (see chapter 9). If you do not include a Reply-To field before selecting Deliver, Lafite will give you a special Reply-To Field menu, asking you to choose between four options. The options are to send the message as is, have recipients reply to yourself, have them reply to someone else, or to abort the message. This is intended to minimize unintentional deliveries to large distribution lists.

ARPANET Messages

The ARPANET (Advanced Research Projects Agency Network) is a network linking ARPA research sites, including some Xerox sites. The Grapevine system is connected to the ARPANET, so you can correspond with individuals at ARPA sites outside Xerox.

As of this writing, ARPANET individual and distribution list addresses are in the form *Name@Host.ARPA*. However, all ARPANET addresses are gradually being changed to domain names. The primary Xerox address, "Xerox.ARPA," will soon become "Xerox.COM"; external ARPANET addresses are also being changed. The name changes are not expected to greatly inconvenience Xerox ARPANET users. The current Xerox address will be supported for some time after the change, so you can receive mail from people who don't yet have your new address. If you send mail to an obsolete address, the ARPANET mail gateway will insert the proper address in the message header, but it is a good idea to keep an eye out for address changes in the correspondence you receive and update your private list of addresses accordingly. You don't have to update ARPANET addresses in any Grapevine distribution lists you own, because ARPANET names on Grapevine distribution lists are kept current by Xerox network administrators (who periodically run a program to update all the names). If you want to verify an ARPANET address, you can search for it in the ARPANET host table stored on {Indigo}<NetInfo>ARPANET>Hosts.Txt. The first name in a host entry list is the primary name and the one you should be using.

ARPANET messages sent via Grapevine undergo some special processing. Because many mail systems receiving the mail you send over the ARPANET cannot process the long lines contained in Lafite messages, carriage returns are automatically inserted to make each line less than 72 characters long if there are any lines of more than 90 characters within the first 5,000 characters of the body of your message. If you have already inserted carriage returns to make your lines slightly longer than 90 characters and you don't want your message to look unaesthetic, include a "Line-Fold: No" line in your header.

ARPANET Distribution Lists. ARPANET distribution lists are not maintained at any one, central location. Instead, they are administered locally by interested sites; for example, SF-Lovers@Rutgers is maintained at Rutgers. When you send a message to an ARPANET distribution list address, it goes to the appropriate host, who sends it to all the members of the list. Some hosts anthologize messages for certain lists (such as SF-Lovers); that is, someone at the host site periodically collects messages from members and sends anthologies of them out to everyone on the list.

Almost every ARPANET distribution list has a Xerox-internal redistribution list, so that all Xerox members are actually members of the redistribution list. For example, the redistribution list for SF-Lovers@Rutgers is XeroxSFLovers↑.PA.

You use Maintain to add yourself to the redistribution list of an ARPANET list in the same way you would add yourself to a Xerox distribution list. If you want to add yourself to (or make another administration request regarding) an ARPANET list that does not have a Xerox redistribution list, send a message to *DistributionList-Request@Host*.

Other Forms of Grapevine and ARPANET Addresses

Lafite recognizes three special forms of address that contain arbitrary text in addition to a valid Grapevine or ARPANET address. The first form is *Human Sensible Name <Actual Address>*. For example, if you wanted to include Cheryl Jones's first name in her address you could write it as Cheryl Jones <Jones.PA>. The area outside the angle brackets can contain any text except for certain prohibited characters: parentheses, angle brackets, commas, colons, semicolons, and quotation marks.

The second form of address is *Actual Address (Comments)*. You could, for instance, send an invitation to Jones.PA (and spouse). You can include any characters within the parentheses except additional parentheses.

The third form of address is "*Comments*" <*Actual Address*>. For example, you could send a message to "Special Attn: Sybalsky, Shih" TEditFriends↑.PA. Any of the prohibited characters can be included within the quote marks except additional quote marks.

Maintain

Maintain is a Lisp Library package used only within Xerox for maintaining Grapevine public distribution lists and changing your Grapevine password. To load Maintain, type (LOAD '{FILESERVER}<DIRECTORY>MAINTAIN.DCOM) in your executive window. To enter Maintain, type (MAINTAIN). Maintain will try to log you in automatically with your current user name and password. If that fails, then you may log in using the Login command in Maintain (see below).

How to Issue Maintain Commands

Maintain has its own prompt sign, in the form GV:. To issue a command in Maintain, you must type after the prompt sign only the initial letters that uniquely identify that command; i.e., as soon as enough of a word (usually one letter) has been typed, Maintain will complete that word for you. For instance, when typing the Add Friend command, you need only type A F. Throughout this section, the letters that Maintain fills in are given in square brackets.

Once you have issued a command for a specific distribution list or individual, Maintain will fill in that same name for the next command you type. If the name is appropriate for that command, press the space bar or carriage return to confirm it. If

the name is not appropriate for the command, begin typing the correct distribution list or individual's name to replace it, ending by hitting the space bar or carriage return.

The Login Command

To log into Maintain, type L[ogin] after Maintain's GV prompt. Now type your name and registry and press the carriage return. Maintain will prompt you for your password, each character of which will appear as an asterisk on the screen. Grapevine maintains its own copy of your password, and the password you give here must be the one Grapevine knows. If you give your password successfully, Maintain will inform you that you are logged in. Once you are logged in, you may proceed to issue other Maintain commands. If you cannot log in, because you have forgotten your password or for some other reason, then you will have to ask an administrator for your registry to issue a Change Password command.

Maintain automatically logs in the current user when it starts up, so you only need to issue the Login command if you wish to change your identity for some reason.

The ? Command

To obtain a list of Maintain commands, type a question mark after the GV prompt. You will see that the commands are Add Friend, Add Member, Add Owner, Change Password, Change Remark, Login, Quit, Remove Friend, Remove Member, Remove Owner, Type Entry, Type Members, and ↑ Y—Enter Lisp. Each of these commands is described below.

The Type Entry Command

The Grapevine system has a *name data base* of entries giving information about groups and individuals. Groups represent distribution lists and some other groups that are unimportant to casual users, such as access control lists. Individuals represent human users and server computers. There are also "abstract users" that are syntactically individuals but actually name a group, such as LafiteSupport.PA.

To read an entry in the data base, type T[type] E[entry] after the GV prompt. Maintain will ask you for the name of the group or individual whose entry you want to read, then print the entry. When it is finished, it will return you to the GV prompt.

An entry for a group has four components: the *remark*, the number of *members*, a list of *owners*, and a list of *friends*. The remark describes the purpose of the group; for example, the remark for the distribution list Books↑.PA is "Resource for Readers (primarily reviews)."

The members are the individuals or other groups contained in the group. To obtain a list of members, use the Type Members command (described below).

The owners are the people empowered to add and remove owners, friends, and members for the group. They can also set and change the remark.

The friends are the people empowered to add and remove their own names to and from the group. If the list of friends is None, no one but the owners can add members to the list. If the list is an asterisk, anyone on the Grapevine system can add himself or herself to the list. If the list is of the form *.Registry, anyone in that registry can add himself or herself to the list, and so on.

An entry for an individual has two components: connect site and mailbox site(s).

The Type Members Command

To find out who the members of a distribution list are, type T[ype] M[embers] after the GV prompt. Maintain will fill in the words "of group" followed by a colon. Type the name of the distribution list after the colon and press the carriage return. Maintain will then print the list of members. If it is a very long list, your executive window may fill up, stop printing the list, and become highlighted in reverse video, a sign for you to hit the space bar to get the listing going again.

Some members of a group may be groups themselves. To see the members of those groups, you can use extra Type Members commands. If you use Type Members for the special group Groups.Registry, you will obtain a list of all distribution lists for that registry.

When Maintain is through printing the members of a distribution list, it will repeat the GV prompt.

The Add Member Command

You can add yourself (or another person if you are an owner) to a distribution list by typing A[dd] M[ember] [name] YourName.Registry after the GV prompt. Maintain will print "to group" after which you should fill in the name of the distribution list to which you want to add your name. If Maintain prints "done," you have successfully added yourself to the list. If it prints "NoChange," you were already a member of the list. If it prints "NotAllowed," you should send a message to the owners of the list, asking them to add your name.

The Remove Member Command

You can remove yourself (or another person if you are an owner) from a distribution list by using the Remove Member command. This works in the same way as the Add Member command.

The Add Owner Command

Use the Add Owner command to add an owner to a distribution list. This is not allowed unless you yourself are an owner.

Otherwise, this works in the same way as the Add Member command.

The Remove Owner Command

Use the Remove Owner command to remove an owner from a distribution list. This is not allowed unless you yourself are an owner. Otherwise, this works in the same way as the Remove Member command.

The Add Friend Command

Use the Add Friend command to add a friend to a distribution list. You cannot do this unless you are an owner. Otherwise, this works in the same way as the Add Member command.

The Remove Friend Command

Use the Remove Friend command to remove a friend from a distribution list. You cannot do this unless you are an owner. Otherwise, this works in the same way as the Remove Member command.

The Change Remark Command

The remark is the description of the distribution list that Maintain prints when you use the Type Entry command. To change the remark, type C[change] R[emark] after the GV prompt. Maintain will prompt you for the name of the group, then ask you to type the new remark followed by a carriage return. You cannot change the remark for a list unless you are an owner.

The Change Password Command

You can change your Grapevine password using Maintain's Change Password command. Type C[change] P[assword] after the GV prompt. Maintain will respond with "for individual." Now type in your name and registry. Maintain will prompt with "to be." Type in your new password, which will appear as a series of asterisks. Maintain will ask you to confirm your new password by typing it again. When you terminate this word with a carriage return or a space, Maintain will store it in the Grapevine data base. Be very careful when typing your new password, because if you make a mistake and can't reproduce the password later you will have to obtain the help of an administrator for your registry to correct it.

After returning to the executive, you will need to reinvoke the Login command (see above) to continue using Lafite to get and send mail.

The ↑ Y Command

If you want to drop into Lisp temporarily, then return to Maintain, type control-Y at the GV prompt. Maintain will fill in

"Enter Lisp" and return you to the regular executive window prompt. To go back to Maintain, type OK.

The Quit Command

To leave Maintain, type Q[uit] after the GV prompt. Maintain will ask you to confirm this command with a carriage return.

If You're on Both Networks

Because the Grapevine and NS systems are totally separate, you must have a registered name and password to send mail on each system you are using. You can have two different passwords, but it is advisable to use the same one because otherwise you will be prompted to log in again each time you switch systems. You can also call (LOGIN 'GV:::) yourself to set your Grapevine log-in, and (LOGIN 'NS:::) to set an NS log-in, apart from your default (Grapevine) log-in.

Lafite Modes

When you use Lafite on the Grapevine system, Lafite is in Grapevine mode. When you use Lafite on the NS system, it is in NS mode. Mail sent to you on a particular system can be accessed, answered, and forwarded only if Lafite is in that system's mode.

If you have loaded both NS and Grapevine protocol handlers (NS Mail and MailClient), Lafite needs to know which mode to operate in. The first time you start Lafite after loading a new Lafite or a full sysout, you must set the mode explicitly. Until you do this, the Lafite status window reads "Mode Not Set." After that, Lafite starts up in the same mode it was in when you quit Lafite. To change modes after you have initialized Lafite, use the middle mouse button to choose Quit from the status menu. Another menu containing several options will appear; one is GV Mode and another is NS Mode. Select the mode you want with the left mouse button. The center region of your status window reflects the mode you select. If you are in Grapevine mode, it says Lafite (GV). If you are in NS mode, it says Lafite (NS). You can also find out what mode you are in by typing (LAFITEMODE) in your executive window, or change modes by typing (LAFITEMODE 'GV) or (LAFITEMODE 'NS). If you always want to start Lafite in the same mode, you can set the global variable LAFITEMODEDEFAULT to the value GV or NS, in which case Lafite will initialize itself in that mode.

At sites with multiple Lafite modes, the values of the global variables LISPSUPPORT, LAFITESUPPORT, and TEDITSUPPORT are association lists in the form ((Mode1 "Address1") (Mode2 "Address2")---). Because the Lisp Report, Lafite Report, and

TEdit Report forms are currently all sent to Grapevine addresses, they cannot be used at all in NS mode.

From Grapevine to NS and Vice Versa

Without switching modes, you can send mail from a Grapevine to an NS address or vice versa through a *mail gateway*. The gateway provides pseudoregistries and pseudodomains for mail addressed to the other kind of network. (A list of the special registries and the domains they correspond to is stored on {Indigo}<Registrar>GV>GV-NS-Mapping.Txt.) A single message can contain both Grapevine and NS addresses.

When Lafite is in Grapevine mode, you can send mail to the NS address *Name:Domain:Organization* by addressing it to "Name:Domain:Organization".NS (the double quotes are needed to keep the colon and any spaces in the name from being misparsed). For convenience, the domains PARC, OSBU North, and OSBU South can also be addressed as the registry of the same name, with the spaces removed. Thus, to send mail to Jones:PARC:Xerox, send it to Jones.PARC; to send mail to Oscar:OSBU North:Xerox, send it to Oscar.OSBUNorth.

When Lafite is in NS mode you can send mail to the GV address *Name.Registry* by addressing it to *Name.Registry:GV*. For convenience, the registries PA and ES can be addressed directly as NS domains. Thus, to send mail to Jones.PA, send it to Jones:PA.

Conversion of Grapevine Distribution Lists for NS Users. Until recently, people who used only NS mail could not add themselves to the many Grapevine public distribution lists. To address this problem, a new naming structure is being applied to general-interest public distribution lists in the PA, ES, and X registries. (Most organization and project lists, or any lists that might be used for access control, will not be converted to the new naming structure.) The new structure allows both NS and Grapevine mail users to add themselves to, broadcast to, and receive messages from these lists. Grapevine users will have to change the way they add themselves to lists. NS users will have to change both the way they send messages to lists that currently reside in Grapevine and the way they add themselves to lists.

To enable the new distribution list structure, new NS domains have been established in a one-to-one correspondence with the ES, PA, and X Grapevine registries. "All Areas:Xerox" is the NS domain analogous to the X registry. "PA Area:Xerox" is the NS analog of the geographically based registry PA, and "ES Area:Xerox" is analogous to ES. After the restructuring is implemented for a distribution list in, for example, the X registry, Grapevine users will continue to broadcast to *List*↑.X, but will add themselves to *List-GV*↑.X. NS users will broadcast to *List:All Areas*, but will add themselves to *List-NS:All Areas*. Either form of broadcast will reach both the Grapevine members (*List-GV*↑.X) and the NS members (*List-NS:All Areas:Xerox*).

The conversion process will take several months. Distribution lists will be converted upon requests from their owners. If you own a distribution list and wish to request conversion, retrieve {USC:OSBU South:Xerox}<Public USC>DLRequest.Form, fill out the form, and send it to the appropriate address, according to the table below.

Registry	Grapevine User	NS User
ES lists	UserAdministration.ESArea	UserAdministration:ES Area:Xerox
PA lists	UserAdministration.PAArea	UserAdministration:PA Area:Xerox
X lists	UserAdministration.AllAreas	UserAdministration>AllAreas:Xerox

As soon as a distribution list of which you are a member is converted to the new naming structure, you will receive notice of the conversion. If you are a Grapevine member of *List↑.X*, you will be added to *List-GV↑.X* and removed from *List↑.X*. If you are a Grapevine user and want to add yourself to a distribution list of which you are not currently a member, use Maintain to attempt to add yourself to *List-GV↑.X*. If Maintain complains that this list does not exist, then add yourself to *List↑.X*. You should continue to address messages to *List↑.X*.

If you are primarily an NS user and are a member of Grapevine *List↑.X*, then when that distribution list is converted to the new structure, you will be moved to *List-GV↑.X*. Then you should ask your system administrator to add you to *List-NS:All Areas:Xerox*. To make sure you don't miss any mail, wait three days after you start receiving two copies of messages sent to *List↑.X*, then remove yourself from *List-GV↑.X* with Maintain. To add yourself to a new distribution list, ask your system administrator to add you to *List-NS:All Areas:Xerox*. If this list does not exist, then add yourself to *List↑.X* using Maintain. You should broadcast messages to *List:All Areas:Xerox*.

One key point to remember is to never send messages to a list with a -GV or -NS suffix. If you do, you will not reach both NS and Grapevine members. Another key point is, never broadcast to both *List↑.X* and *List:AllAreas*, or everybody will get two copies of the message.

A more detailed description of the conversion process is filed on {USC:OSBU South:Xerox}<Public USC>DLConversion.Doc and {Indigo}<Registrar>GV>DLConversion.Doc.

Lafite-Related Lisp Users' Packages

As of this writing, Xerox employees have access to four Lafite-related Lisp Users' packages that have not yet been released to customers. The packages are Mailomat, Mail Sorter, Mail Cards, and Mail Reader. Mailomat enables you to automatically retrieve mail for one or multiple users for whom you have passwords. The program is filed on

{Eris}<Lisp>Koto>LispUsers>Mailomat.DCOM; the documentation is on {Eris}<Lisp>Koto>LispUsers>Mailomat.TEdit. Mail Sorter will automatically sort mail into folders according to a set of rules you have specified. The code and documentation are on {QV}<Pimp>MailSorter.DCOM and {QV}<Pimp>MailSorter.TEdit, respectively. Mail Cards integrates the Note Cards environment with the Lafite mail system; you can place Lafite messages in specialized note cards, which are then placed in a Note Cards network. The code is stored on {QV}<Pimp>MailCards.DCOM and {QV}<Pimp>MailCardsLafiteInterface.DCOM; the documentation, on {QV}<Pimp>MailCards.TEdit. Lastly, the Mail Reader uses a voice synthesizer to read your mail to you over the telephone. You do not need special code to use this package, because you enter all commands by pressing the buttons on a touch-tone telephone. The Mail Reader's documentation is stored on {QV}<Speech>Speech>MailReader>MailReader.TEdit.

All the abovementioned file names are subject to change. If you can't find a specific package, ask a release master or a network administrator for a pointer.

From Laurel to Lafite

Previous users of the Laurel mail program will find Lafite to have a similar user interface. Some of the ways in which it differs from Laurel are the following:

Laurel can only access mail folders on the local disk; Lafite can access folders on remote file servers. Thus, it is not necessary to transfer mail folders back and forth between your local disk and your file servers.

Laurel can only "browse" one mail folder at a time; Lafite can have several mail folders "opened" at the same time. Using the Interlisp window system, you can view the tables of contents of many mail folders and refer to messages in them independently.

You may have multiple windows displaying messages and multiple windows for sending messages and you may move text freely among them.

Messages can be selected in the table of contents by clicking anywhere within the summary line, unlike in Laurel, where you must select at the left end.

The Browse Laurel File Command

Lafite can read mail files written by the Laurel and Hardy mail programs; the files it writes are in Laurel format.

To browse a Laurel file that was produced by the Laurel mail reader version 6.1 or later, use the Browse Laurel File command from the middle-button Browse menu. Laurel 6.1 files are almost the same as Lafite files, but contain some line-formatting information that is stripped out by this command. After you have applied this command once to a file, you can subsequently browse the file with the normal Browse command (unless you use Laurel on it again, of course). This command is not intended for repeated use, but simply to make mail files built by Laurel more pleasing to browse. This command has the side effect of destroying any TEdit-formatted messages, so it should be used with care.

[This page intentionally left blank]

GLOSSARY OF TECHNICAL TERMS

This glossary covers hardware and software terms not specific to Lafite that are used in this manual.

argument Element of a Lisp function that specifies what that function operates on. For example, when you call (LOAD 'FILENAME), FILENAME specifies what file is to be loaded. Note the use of parentheses and the single quote mark—the quote causes FILENAME to be taken literally, rather than as the name of a variable to be evaluated.

association list A list of lists that associates a key (usually an atom) with a value. The first element of each sublist is the key; the rest of the sublist is a value.

atom Any continuous string of characters, numbers, letters, or combinations thereof (except some prohibited characters such as parentheses). An atom is the smallest structure in Lisp.

back up To duplicate files to use in case the originals are destroyed in a hardware or software failure.

bit map A representation of any graphical entity as a sequence of bits.

break A state entered by Lisp during error processing that allows you to recover from the error by typing commands in a break window. If you don't know what to do with a break, type ↑ after the prompt to abort the operation, then start over.

buffer A temporary storage area in a computer.

bug Any error, malfunction, or problem with hardware or software that produces unexpected or unintended results.

cache To save or store intermediate results to avoid having to recompute them.

caret A blinking pointer indicating where keyboard characters will appear when typed.

case sensitive Sensitive to case, that is, upper- and lowercase letters have different meanings.

Clearinghouse A Xerox Network service that provides a directory function within an internetwork, allowing all other system components to locate requested resources and other registered objects. It is implemented as a distributed system.

cursor A small picture (usually an arrow) on the display that tracks the motion of the mouse.

data base A collection of data organized for rapid search and retrieval.

DEdit The Interlisp-D editor for programming code.

default An action taken (or value specified) unless another action is specified by the user.

directory A set of one or more files that are stored together in the same place on a device.

executive window The window the blinking caret is in when you first start up Lisp on the machine. Note that if this and any other window overlap, the executive window dominates (i.e., covers) the other while control is in the executive window. To transfer control to another window (that accepts type-in), click the mouse in it, which usually also brings the window to the top.

extension The second part of a file name, usually used to indicate the type of file. It is separated from the first part of the file name (which indicates what the file contains) by a period. For example, the extension for Lafite mail folders is Mail, so a typical file name might be Active.Mail.

field A part of a message header, preceded by a field name and a colon. Contains information identifying the sender, recipients, subject, or other information of interest either to the users or to the mail system.

file server A computer on the network that provides a file storage and retrieval service.

format To produce in a specified form; the layout of a document, etc.

function A Lisp procedure that carries out a series of steps to produce some result. A function has a name and zero or more arguments on which it does its work.

global variable A variable accessible globally to all loaded programs, and whose value is usually not changed, except explicitly by the user (e.g., in an init file). Global variables are often used to personalize some aspect of the behavior of a program.

hard copy The physical copy of an on-screen document.

host Any machine on a network. Often used to refer specifically to a machine that provides a network service, such as filing.

icon A pictorial representation, usually of a shrunken window. An icon can be moved about on the screen by clicking on it with the left mouse button, and can be expanded by clicking on it with the middle mouse button.

initialization (init) file A file that is loaded when an Interlisp sysout is first started, and which usually customizes your Lisp environment according to your tastes and the idiosyncrasies of your site. The usual arrangement is to have a site initialization file, which supplies information common to all users at your site (e.g., the name of your printer or directory search paths), and a personal initialization file, which supplies information about

how you personally like your environment set up (e.g., where on the screen you like your Lafite windows).

keyword A significant word from a title or document used as an index to content.

mail server A computer that stores and distributes mail.

menu A collection of text strings, buttons, or icons on a display screen generally used to present a set of possible actions.

microcode The microinstructions of a computer, especially a microprocessor.

mode A particular functioning arrangement or condition of a computer.

network An interconnection, by some communications medium, of several computers, allowing them to communicate and share resources. Sometimes also used to refer to the medium itself, e.g., a coaxial cable in the case of the Ethernet.

package An Interlisp-D tool, or program, designed to help you carry out some complex procedure or task. Lafite is a package that enables you to read and send mail. A package must be loaded into your environment before you can use it.

parse To analyze something into its constituent parts, either to expose the structure or extract information. Specifically used to refer to the action of locating the message boundaries in a mail folder, or extracting fields from a message header (e.g., to discover who the message is addressed to).

position As a Lisp data type, a pair of display coordinates, in the form (*XPosition . YPosition*). If you want to know the coordinates of a particular spot on the screen, type (GETPOSITION) and then click the mouse over the desired spot.

protocols Standards specifying how machines exchange information over a network.

random access Access to arbitrary parts of a file in no particular order (contrast with sequential access, which is accessing a file from the start sequentially through to the end).

region As a Lisp data type, a set of numbers describing a rectangular region of the display, in the form (*LEFTCOORDINATE BOTTOMCOORDINATE WIDTH HEIGHT*). If you want to know the region corresponding to any particular rectangular area on the screen, type (GETREGION) and then shape the rectangle as desired.

scavenge To try to fix messages that are not in the proper format.

search path List of directories to be searched for a file in the order given.

stream A programming abstraction that links your display to your message file.

string A sequence of characters. As a type of Lisp data, always entered in double quotes.

sysout A frozen version of an Interlisp-D environment. It contains all the information needed to initialize virtual memory when Interlisp-D is started.

time out To stop waiting for a user response, or to sever a network connection because of inactivity on the part of one of its participants.

value A word or a number that a variable is set to.

virtual memory Large working space on a local disk within which the machine runs Lisp.

Message Marks

? Means that a message has not yet been displayed.

a Means that a message has been answered.

f Means that a message has been forwarded.

h Means that a message has been hard-copied.

m Means that a message has been moved to another folder.

Note: Only one mark is present at a time; each mark replaces the previous one.

Lafite Commands

Status Window Commands

Browse

Pops up a menu of your mail folders. Selecting Browse with the middle button brings up another menu with the additional commands:

Browse Laurel File

Browses a file that was produced by Laurel.

Forget Folder

Removes a folder from the list of known mail folders.

Forget Message Form

Removes a message form from the list of known message forms.

Send Mail

Brings up a message composition window. If Send Mail is selected with the middle button, a menu is presented with the following choices:

Lisp Report

Provides a message template to report an Interlisp bug or make a suggestion.

Lafite Report

Provides a message template similar to the Lisp Report but sent to Lafite maintainers.

TEdit Report

Provides a message template similar to the Lisp Report but sent to TEdit maintainers.

Saved Form

Prompts for a form name.

Standard Form

Provides an empty message template.

Quit

Stops Lafite and closes all browser windows. Selecting the Quit command with the middle button brings up a menu of status-changing commands:

Restart

Turns Lafite off, then on again.

Browser Window Commands

Display

Displays the selected message.

Delete

Deletes the selected message.

Undelete

Undeletes the selected message.

Answer

Constructs an Answer form for the selected message.

Forward

Constructs a Forward form for the selected message.

Hardcopy

Prints the selected message.

Move To

Pops up a menu of known mail folders and moves the selected message to the chosen folder.

Update

Transmits the changes that you made to your mail folder to the physical mail file. Selecting Update with the middle mouse button brings up a menu with the commands:

Write Out Changes Only

Makes the browser and the mail file completely consistent with each other.

Expunge Deleted Messages

Removes all messages marked deleted, in addition to making the browser and the mail file completely consistent with each other.

Get Mail

Brings new mail into the folder.

Message Composition Window Commands

Deliver

Sends your message. During delivery, the menu atop the message window changes into a single item, Abort; if you click on this item, the delivery is aborted.

Save Form

Asks you for a file name on which to save this message for later use as a message form.

[This page intentionally left blank]

A

Abort command, 45, 47, 87
Active Mail folder. *See* folders, Active Mail
 addresses, recipient, 28–29, 71–73
 Answer command, 38–39, 86
 Answer form. *See* forms, Answer
 authentication, of users, 3, 7, 51

B

bit maps, including in messages, 36–37, 46
 Body command, 64
 Browse command, 9, 22–23, 42, 85
 Browse Laurel File command, 23, 85
 browser window. *See* windows, browser

C

CC field, 26, 28–29
 Close command, 16
 commands
 invoking, 8
 Lafite, 85–87. *See also* individual
 command names

D

DEFAULTMAILFOLDERNAME [variable], 56
 Delete command, 15, 86
 Deliver command, 45, 87
 Display command, 11–12, 86
 distribution lists, 28–31, 39, 45, 76–77
 Don't Quit command, 49
 Don't Update File command, 16, 49

E

electronic mail
 cost structure, 75
 need for etiquette, 70–71, 73–74, 79
 Expunge Deleted Messages command, 16, 49, 53, 87

F

fields, header, 26–31, 38–42. *See also* individual
 field names
 files
 including in messages, 33–34
 loading, 5–6
 mail. *See* folders
 necessary, 5
 Find Again command, 63, 65
 Find command, 63–65
 Find Next All command, 63–65
 Find Next One command, 63–65
 Find Previous All command, 64
 Find Previous One command, 63–65
 Find Related Backward command, 65
 Find Related command, 63–65
 folder menu, 9, 21–24
 folders
 Active Mail, 3, 7, 9, 17, 21, 52, 56
 creating, 21–22
 default, 23, 56
 deleting, 23–24
 naming, 21–22, 56
 parsing, 52–53
 repairing, 66–67

- storing, 3, 51–52, 55
 structure of, 3
 updating, 16–17, 49
- f**
 fonts, 36, 57
Forget Folder command, 23–24, 85
Forget Message Form command, 42–43, 85
forms
 Answer, 38–39
 creating, 41–42, 59–61
 deleting, 42–43
 directory search path for, 55
 extension for names, 56
 Forward, 39
 kinds of, 25–26
 menu, 38, 42–43
 Standard, 26–27, 38, 86
 Report, 40–41, 56–57
Forward
 command, 39, 86
 form. *See* forms, Forward
Frequency field, 40
From
 command, 64–65
 field, 29–30
- G**
Get command, 34–35
Get Mail command, 10, 87
- H**
Hardcopy command
 Lafite, 19, 86
 window menu, 19, 36
headers. *See* message headers
- I**
Impact field, 41
- in-box, 3, 7, 10, 51, 58
Include command, 33–34
In-Reply-To field, 31
- J**
Just Close command, 17
- L**
Lafile
 capabilities of, 1, 3
 quitting, 49
 starting, 7
LAFILEBROWSERFONT [variable], 57
LAFILEBROWSERREGION [variable], 56
LAFILEBUFFERSIZE [variable], 59
LAFILEDEFAULTHOST&DIR [variable], 55
LAFILEDISPLAYAFTERDELETEFLG [variable], 59
LAFILEDISPLAYFONT [variable], 57
LAFILEDISPLAYREGION [variable], 56
LAFILEEDITORFONT [variable], 57
LAFILEEDITORREGION [variable], 56
LAFILEENDOFMESSAGEFONT [variable], 57
LAFILEENDOFMESSAGESTR [variable], 58
 Lafite Find, 5–6, 63–65
LAFILEFLUSHMAILFLG [variable], 58
LAFILEFORMDIRECTORIES [variable], 55
LAFILEFORM.EXT [variable], 56
LAFILEFORMSMENU [variable], 60
LAFILEGETMAILTUNE [variable], 59
LAFILEHARDCOPYBATCHFLG [variable], 58
LAFILEHARDCOPYFONT [variable], 57
LAFILEHARDCOPY.MIN.TOC [variable], 58
LAFILEIFFFROMMETHENSEENFLG [variable], 58
LAFILEMAIL.EXT [variable], 56
LAFILEMENUFONT [variable], 57
LAFILEMOVEVETOCONFIRMFGL [variable], 59

LAFITENEWMAILTUNE [variable], 59
 LAFITENEWPAGEFLG [variable], 58
 (LAFITE 'OFF) [function], 49
 (LAFITE 'ON 'MAILFOLDER) [function], 7
 LAFITEOUTBOXSIZE [variable], 59
Lafite Report
 command, 86
 form. *See forms, Report*
 LAFITESPECIALFORMS [variable], 60
 LAFITESTATUSWINDOWPOSITION [variable], 56
 LAFITESUPPORT [variable], 57
 LAFITETITLEFONT [variable], 57
 LAFITETOC.EXT [variable], 56
Lisp Report
 command, 86
 form. *See forms, Report*
 LISPSUPPORT [variable], 56
 log-in procedures, 5

M

(MAILSCAVENGE 'FILENAME 'ERRORMSGSTREAM
 'TEMPDIR) [function], 66-67
 (MAILSCAVENGE.IN.PLACE 'FILENAME) [function], 67
 Mail Scavenger, 5-6, 53, 63, 66-67
 MAILWATCHWAITTIME [variable], 58
 (MAKEXXXSUPPORTFORM 'SYSTEMNAME
 'ADDRESS 'SYSTEMDATE) [function], 60-61
 menus, 8, 34, 36, 63. *See also* individual
 command names

message

body, 25-26
 forms. *See forms*
 headers, 3, 25-31, 38-41, 63-64
 marks, 10-11, 19, 21, 39, 58, 85

messages

answering, 38-39
 deleting, 15

delivering, 45-47
 displaying, 9-14, 59
 expunging, 16
 filing, 21-24. *See also* folders
 forwarding, 39
 misaddressed, 71-73
 moving, 21-22, 59
 printing, 19, 36, 57-58
 and privacy, 71-72, 78
 retrieving, 10, 35
 saving, 34-35, 41-42
 searching for, 63-65
 selecting, 11-12
 structure of, 3
 undeleting, 15
 writing, 25-43

Move To command, 21-22, 53, 86

N

NS Mail
 service, 3
 protocols, 5-6

O

Other Folder command, 21-24
 out-box, 8, 46, 59

P

protocols. *See NS Mail*
 Put command, 34-35

Q

Quit command, 49, 86

R

Related command, 64-65
 Reply-To field, 30-31
 Report forms. *See forms, Report*

Restart command, 49, 86

S

Saved Form command, 38, 42, 86

Save Form command, 41-42, 87

scroll bars *See* windows, scrolling

selection pointer. *See* messages, selecting

Sender field, 29-30

Send Formatted Message command, 46

Send Mail command, 38, 42, 85

Send Plain Text command, 46-47

Shrink command, 16

Standard form. *See* forms, Standard

start-up procedures. *See* Lafite, starting

status window. *See* windows, status

Subject

command, 64

field, 26-27

T

table of contents, Lafite, 3, 9-12, 15-17, 21-22, 24, 52-53, 56, 58

TEdit, 1, 31-37, 53

TEdit Report

command, 38, 86

form. *See* forms, Report

TEDITSUPPORT [variable], 57, 60

TEDITSYSTEMDATE [variable], 60

text

copying, 33

deleting, 33

inserting, 31-32

moving, 33

selecting, 32-33

To field, 26, 28-29

U

Undelete command, 15, 86

Update command, 16-17, 53, 86

Update Table of Contents command, 17

users

authenticating. *See* authentication of users

fictitious, 78-79

V

variables, 55-60

W

window menu, 36

windows

browser, 9-10, 22, 56, 86-87

closing, 16-17, 35, 46

message composition, 25-26, 31-33, 35, 56, 87

message display, 11-14, 56

scrolling, 12-13

shrinking, 16-17, 35

status, 7-8, 46, 49, 56, 85-86

thumbing, 13-14

Write Out Changes Only command, 16, 49, 53, 87