

## 14. USING LAFITE-RELATED LISP LIBRARY PACKAGES

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.

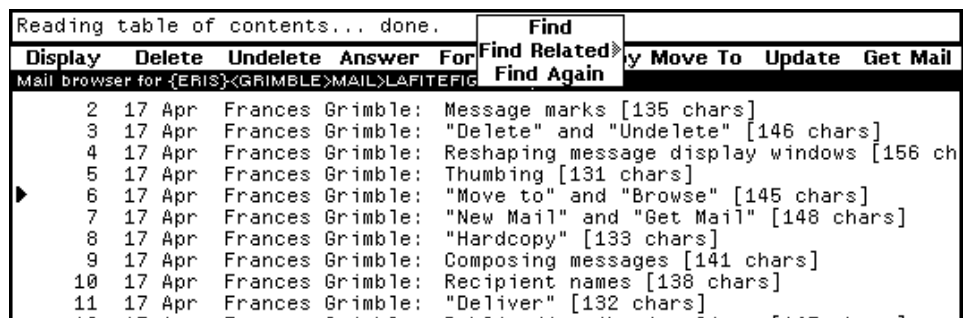
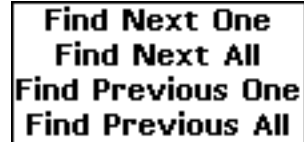


Figure 30. A browser window in which the user has brought up the top-level Lafite Find menu. You can bring up the menu anywhere on the title bar by clicking the middle mouse button

### 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.



**Find Next One**  
**Find Next All**  
**Find Previous One**  
**Find Previous All**

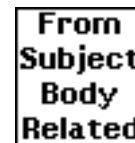
*Figure 31. The search direction commands menu*

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 search examines the entire message, and the Related search is a special kind of Subject search.



**From**  
**Subject**  
**Body**  
**Related**

*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.DCOM)
=
{ERIS}<LISP>INTERMEZZO>LIBRARY>MAILSCAVENGE.
DCOM;1

{ERIS}<LISP>INTERMEZZO>LIBRARY>MAILSCAVENGE.
DCOM;1
compiled on 21-Feb-85 17:50:30
FILE CREATED 21-Feb-85 17:49:29
MAILSCAVENGECOMS

{ERIS}<LISP>INTERMEZZO>LIBRARY>BSEARCH.DCOM;
1
compiled on 7-May-84 23:20:16
FILE CREATED 7-May-84 23:19:59
BSEARCHCOMS
{ERIS}<LISP>INTERMEZZO>LIBRARY>MAILSCAVENGE.
DCOM;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 newl
y-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*.