FILEBROWSER

FileBrowser provides a convenient user interface for manipulating files stored on a workstation or file server. It enables you to see, edit, delete, print, load, copy, move, rename, compile, sort, and get several types of information about groups of files. You can also customize FileBrowser by adding your own commands.

Requirements

TABLEBROWSER

In addition, the `HARDCOPY` commands require printer drivers and fonts, and the EDIT command requires one or more editors (TEdit, SEdit, DEdit).

Installation

Load `FILEBROWSER.LCOM` and the required `.LCOM` modules from the library.

User Interface

Starting FileBrowser

Once you have loaded FileBrowser, there are two ways to open a browser on a set of files:

1. Select the FileBrowser command from the background menu, in which case you are prompted for a file name pattern, or

2. Type the command FB *FILEPATTERN* in your Executive window.

In either case, FileBrowser will prompt you to create a window by presenting you with a dashed rectangle with the mouse cursor and a small geometric design at the lower right corner.

1. Move your mouse until the upper left corner of the rectangle is where you want it on the screen.

2. Hold down the left mouse button and move your mouse down and to the right, thus expanding the window diagonally, until the window is the right size.

3. Release the mouse button. This creates a window group on your screen in the outlined area.

Next, if you did not specify a pattern by using the FB command, FileBrowser prompts you for a file pattern. Type a pattern, as described in the Specifying What Files to Browse section below.

FileBrowser enumerates the set of files matching the pattern you requested to see. While the enumeration is in progress, the `RECOMPUTE` command is grayed out. When the enumeration is finished, you may select files and issue commands.   You can scroll the window at any time, even while the browser is busy.

If FileBrowser cannot find any files matching the pattern you specified, or you decide you specified the wrong pattern and want to try again, you can specify a new file name pattern from within the browser using the NEW PATTERN command; see RECOMPUTE in the FileBrowser Commands section below.

You can have as many active FileBrowsers open at once as you like.

Specifying What Files to Browse

A full file name in Lisp consists of a device or host (such as your local disk, a file server, or a floppy disk), a principal directory and zero or more subdirectories, a file name (possibly including an extension), and a version number. These fields are put together in the form

```
{HOST}<DIRECTORY>SUBDIRECTORY>FILENAME.EXTENSION;VERSION
```

A file name pattern, as specified to FileBrowser, consists of a file name with one or more pieces omitted or filled with wild cards (*). All the files matching the pattern are listed by FileBrowser. Thus, you can browse all the files in a particular directory, all the files in a subdirectory of that directory, all the files in a directory with a particular extension, and so forth. The wild card * can be used to stand for zero or more consecutive characters in the file name. You can use as many wild cards in a pattern as you wish.

If you leave out some of the fields in a file name pattern, the missing fields are defaulted by the system. Omitted fields in the front of the pattern, i.e., the host, device, or directory fields, are filled in by consulting your connected directory. Other omitted fields are filled in with wild cards unless they are explicitly omitted; i.e., the field is empty, but the preceding punctuation is still present. In more detail, some of the cases are as follows:

If you leave out the name of the host/device, specifying *<DIRECTORY>FILENAME*, FileBrowser will use the name of the host/device for the directory to which you are currently connected.

If you leave out both the device and directory names, specifying *FILENAME*, FileBrowser will use the device and directory to which you are currently connected.

If you do not specify a file name, FileBrowser lists all the files in the specified directory (or the connected directory if you also omitted the host and directory).

If you leave out the extension of a file name, FileBrowser lists all the files with the specified file name and any extension. If you omit the extension but include the period that usually precedes the extension, FileBrowser lists only the files with the specified name and *no* extension.

If you omit the version number of the file name, FileBrowser lists all versions of the matching files. If you omit the version number but include the semicolon that usually precedes the version, FileBrowser lists only the highest version of the matching files.

Thus, the minimal pattern you can type is * (asterisk—enumerate all files in the connected directory) or ; (semicolon—enumerate just the highest version of all files). If you press the RETURN key without giving a pattern, FileBrowser aborts the prompt for a pattern, leaving you with an empty browser in which the only things you can do are change some FileBrowser parameters (see the subcommands of RECOMPUTE in the FileBrowser Commands section below) and then use the RECOMPUTE command to be prompted for a pattern again.

Example

The pattern `*` specifies all files in the connected directory. It is equivalent to `*.*` or `*.*;*`.

The pattern `<FOO>BAR` specifies all files in directory `FOO` with name `BAR` and any extension. It is equivalent to `<FOO>BAR.*;*`.

The pattern `<FOO>BAR.` specifies all files in directory `FOO` with name `BAR` and *no* extension. It is equivalent to `<FOO>BAR.;*`.

The pattern `*.TEdit` specifies all files in the connected directory with the extension TEdit. It is equivalent to `*.TEdit;*`.

The pattern `*.TEdit;` specifies only the newest version of all files in the connected directory with the extension `.TEdit`.

The pattern `<FOO>A*E` specifies all files in directory `FOO` whose names begin with `A` and end with `E` and have any extension.

The pattern `{TOAST}<FOO>*MY*` specifies all files in directory `{TOAST}<FOO>` whose names contain the substring `MY` and any extension.

Using the FileBrowser Window

The FileBrowser window has six major subwindows, which from top to bottom are as follows:

PROMPT window

> This topmost subwindow is where FileBrowser prints messages about what it is doing and receives input from you. Its contents are cleared before every command.

TALLY window

> This subwindow immediately below the prompt window keeps a running tally of the total number of files listed in the window and the number of files that you have marked for deletion. In addition, if one of the attributes you are displaying is a size attribute (Pages or Length, as in the INFO menu, described below), this window maintains a tally of the total number of pages in the files listed and the files marked for deletion.

> This window also has a title bar across the top identifying the pattern you specified and the time at which the directory enumeration was performed.

> The window is blank while the files are being enumerated.

```
File group description: {erinyes}<Lisp>Lyric>Library>*.lcom

Enumerating {erinyes}<Lisp>Lyric>Library>*.lcom;* ...done
```

{ERINYES}<Lisp>Lyric>Library>*.lcom;* at 11:13 Wed 3-Feb-			FB Commands
Total: 99 / 4803 pages		Deleted: 0 / 0 pages	Delete ▶
Name	Pages	Created	Undelete ▶
			Copy ▶
4045XLPSTREAM.LCOM;1	101	5-Mar-87 17:52:40 PST	Rename
BROWSER.LCOM;1	19	24-Apr-87 10:59:38 PDT	Hardcopy ▶
CENTRONICS.LCOM;1	8	29-Nov-86 17:22:04 PST	See ▶
CHARCODETABLES.LCOM;1	8	29-Nov-86 17:22:28 PST	Edit ▶
CHAT.LCOM;1	59	26-Jan-87 22:28:05 PST	Load ▶
CHATTERMINAL.LCOM;1	20	27-Nov-86 13:27:14 PST	Compile ▶
CMLFLOATARRAY.LCOM;1	23	9-Apr-87 16:32:31 PDT	Expunge
COPYFILES.LCOM;1	16	27-Nov-86 15:21:53 PST	Recompute ▶
DATABASEFNS.LCOM;1	14	3-Dec-86 11:50:59 PST	Sort
DEDIT.LCOM;1	96	29-Nov-86 17:31:58 PST	
DEDITPP.LCOM;1	30	16-Dec-86 17:56:57 PST	
DES.LCOM;1	29	22-Dec-86 11:41:51 PST	
DLRS232C.LCOM;1	109	3-Jun-87 10:18:19 PDT	

Figure 1. Tally Window

BROWSER window

> This is the principal subwindow, in which the files matching the specified
> pattern are listed. Each file's name appears at the left, and various attributes
> of the file are displayed in columns to the right. A title bar across the top of the
> browser window identifies the contents of each column (e.g., Name, Pages,
> Created). The files are listed in alphabetical order, with multiple versions of the
> same file listed in decreasing version order; i.e., the newest version appears
> first. The width of the column listing the file names is initially chosen to be
> appropriate for average-sized file names. If the files you asked to browse have
> particularly long names, then when FileBrowser has finished listing all the files
> it may choose to redraw the browser window with the attribute columns moved
> farther to the right to accommodate the longer file names.

COMMAND menu

> This menu appears vertically along the right side of FileBrowser window (under
> a title bar "FB Commands") and lists the commands that you may select to
> perform operations on the files in the browser, or to change the appearance of
> the browser. Most of the commands operate on the set of currently selected files
> (see the Selecting Files section below). Some commands have subcommands, as
> indicated by the small triangle alongside them, which can be selected by holding
> down the left mouse button and sliding the mouse to the right over the triangle.

INFO menu

> An additional subwindow, the Info menu, is not normally displayed. It is used
> to change the set of file information (attributes) displayed in the browser (see
> the Getting Information About Files section below).

SCROLL bar

> If there are more files in the listing than fit at one time in the browser window,
> you can scroll the browser window to view more files. Slide the mouse cursor
> out the left side of the browser window to get the scroll bar and press the left
> mouse button to scroll the region up and the right mouse button to scroll it
> down. Pressing a mouse button when the cursor is near the bottom of the scroll
> bar scrolls the region by larger increments than when the cursor is at the top.

> You can also press the middle mouse button in the scroll bar to move the listing
> to the place that corresponds to that position in the scroll bar.   For example,

pressing the middle mouse button when the cursor is at the bottom of the scroll bar displays the end of the listing. This quick-scrolling technique is called thumbing. The gray box in the scroll region indicates where the currently displayed contents are, relative to the entire contents of the browser.

Similarly, if there is more attribute information than fits in the browser window, you can scroll the browser window horizontally to view the rest of the attribute information. To do this, slide the mouse out the bottom of the browser window to get the horizontal scroll bar. The left button scrolls to the left, the right button to the right.

Selecting Files

Most FileBrowser operations are performed by selecting a single file or set of files, then giving a command that specifies what you want to do with the selected files. The current selection is indicated by a small triangle in the left margin of the browser next to each selected file.

```
  CENTRONICS.LCOM;1            8   29-Nov-86 17:22:04 PST      See      »
▶ CHARCODETABLES.LCOM;1        8   29-Nov-86 17:22:28 PST      Edit     »
▶ CHAT.LCOM;1                 59   26-Jan-87 22:28:05 PST      Load     »
▶ CHATTERMINAL.LCOM;1         20   27-Nov-86 13:27:14 PST    Compile    »
  CMLFLOATARRAY.LCOM;1        23    9-Apr-87 16:32:31 PDT    Expunge
  COPYFILES.LCOM;1            16   27-Nov-86 15:21:53 PST   Recompute  »
  DATABASEFNS.LCOM;1          14    3-Dec-86 11:50:59 PST      Sort
▶ DEDIT.LCOM;1                96   29-Nov-86 17:31:58 PST
▶ DEDITPP.LCOM;1              30   16-Dec-86 17:56:57 PST
  DES.LCOM;1                  29   22-Dec-86 11:41:51 PST
```

Figure 2. FileBrowser Window with Files Selected

To select one file, point to any part of the line (which lists the file name and its attributes) and press the left mouse button. If other files are already selected, this unselects them; thus, a file selected with the left mouse button is always the only selection.

To add a single file to the current selection, press the middle mouse button at any place in the line. The file is selected without unselecting any other file.

To remove a single file from the current selection, hold down the control key and press the middle mouse button at any place in the line. The file is unselected without affecting any other file.

To extend the selection to include a group of contiguous files, that is, to select all the files between a file and the nearest already selected file, press the right mouse button on any part of the line. You can only extend the selection from the first selected file upward, or the last selected file downward. In addition, files marked for deletion are not normally selected when you extend.

If you want to include all files, both deleted and undeleted, hold down the control key while extending the selection.

Some lines in a FileBrowser display are directory-only lines. These lines are slightly indented and name the directory and subdirectory to which the files listed below that line belong. You cannot select in these lines, though you can copy-select them (see the Copy-Selecting Files section below).

Commands that Require Input

Some FileBrowser commands require input from you. For example, the COPY command requires that you supply a destination file name. When a command requires

input, FileBrowser prints a prompt message in its prompt window. This is usually followed by a default answer. If you want the default answer, you can just press the carriage return to finish the input. If you want to specify a different answer, simply start typing it; the default answer is erased and your answer replaces it.

Alternatively, you can modify the default answer by backspacing over individual letters, or typing control-W to back up over complete words. Typing control-Q erases the entire answer. You can also use the mouse to edit your answer, using the same rules as followed by the Executive (see the documentation of TTYIN). Briefly, the left mouse button positions the caret at a character boundary; the middle mouse button positions the caret at the nearest word boundary; and the right mouse button deletes the characters between the caret and the mouse.

When you have finished, position the caret at the end of your answer, if it isn't there already, and press the carriage return. You can also type control-X to finish your answer even if the caret isn't at the end.

If you change your mind and want to abort the command, supply an empty input; i.e., if there is an answer in progress, backspace over it or type control-Q to erase it, then press the carriage return. FileBrowser prints "aborted" and aborts the command. In most situations, the control-E interrupt can also be used to abort your answer.

While you are typing an answer, you can copy-select file names out of the browser (or any other browser), as described below in the Copy-Selecting Files section. This can be useful, for example, if you wish to rename a file to a similar name in the same directory, or move a file into a subdirectory listed in the browser.

Aborting Commands

During commands of indefinite duration, such as RECOMPUTE or COPY, FileBrowser adds another command to the browser, "Abort".
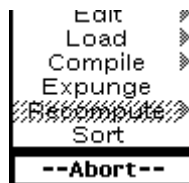

Figure 3. Command Menu with "Abort" Added at Bottom

Clicking on the Abort command will immediately abort the current operation. Aborting some commands can take a little while, as FileBrowser may need to do some cleaning up, so the Abort command is greyed out during this time to show you that it is doing something.

Quitting the FileBrowser

To quit a FileBrowser, simply close the browser window. If any files have been deleted but not expunged, a small menu will pop up listing two options: "Expunge Deleted Files" and "Don't Expunge." If you choose "Expunge Deleted Files", the files will be expunged before the window closes. If you choose the "Don't Expunge" command, your deletions are ignored. If you click outside the menu, no action is taken, and the Close command is aborted.
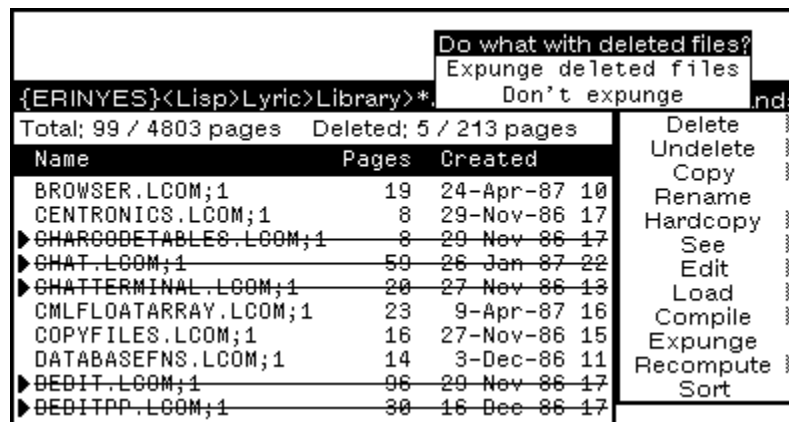
Figure 4. FileBrowser Window with Files Selected for Deletion

If you have finished with FileBrowser only temporarily and want to put it aside to work on later, you can shrink the browser (by selecting the SHRINK command from the right-button background menu). The browser shrinks to an icon which displays the file pattern inside the browser. If any files are marked for deletion, you will be prompted with the same menu of EXPUNGE options as when you close a browser.

Figure 5. Shrink Icon with File Pattern

Copy-Selecting Files

You can copy-select file names from a FileBrowser into other windows, such as Executive and TEdit windows, by holding down the Copy (or Shift) key while selecting a name in the window. The full name of the file is inserted as if you had typed it where the input caret is flashing. You can also copy-select in a directory-only line, in which case the full directory name is inserted in your type-in.

Note: Most file names contain characters, in particular colon and semi-colon, that have special meaning to the Lisp reader. Thus, if your type-in point is in an Executive window, you probably want to type a double-quote character before and after the file name, so that the file name is presented to Lisp as a string.

Getting Hardcopy Directory Listings

You can get a hardcopy listing of the directory displayed in a FileBrowser by using the regular window Hardcopy command. Press the right button in FileBrowser's prompt window or tally window and select HARDCOPY from the menu. FileBrowser will produce a hardcopy listing of the files and the attributes displayed in the browser.

If the browser displays a large number of attributes, or your default printer font is too large, the listing may not accommodate all the attributes on one line, making the listing less readable. You may want to make the listing with fewer attributes, or use a smaller font for the listing (see description of FB.HARDCOPY.FONT in the Customizing FileBrowser and Using the Programmer Interface section).

FileBrowser Commands

DELETE, UNDELETE

Removing a file from the file system using FileBrowser is a two-step process. First, mark the file or files for deletion. Then issue the Expunge command. Any time between the deletion and the expunge you can change your mind and undelete any of the files.

To mark a file or files for deletion, select them, then choose the DELETE command. FileBrowser draws a line through the deleted files. It also adjusts the numbers in the tally window to show how many files are marked deleted and how many pages they contain. It is thus easy to see how much file space you will regain when you issue the EXPUNGE command.

```
  CENTRONICS.LCOM;1        8  29-Nov-86 17    Hardcopy  ▶
▶ CHARCODETABLES.LCOM;1    8  29-Nov-86 17      See     ▶
▶ CHAT.LCOM;1             59  26-Jan-87 22      Edit    ▶
▶ CHATTERMINAL.LCOM;1     20  27-Nov-86 13      Load    ▶
  CMLFLOATARRAY.LCOM;1    23   9-Apr-87 16    Compile   ▶
  COPYFILES.LCOM;1        16  27-Nov-86 15    Expunge
  DATABASEFNS.LCOM;1      14   3-Dec-86 11   Recompute  ▶
▶ DEDIT.LCOM;1            96  29-Nov-86 17      Sort
▶ DEDITPP.LCOM;1          30  16-Dec-86 17
  DES.LCOM;1              29  22-Dec-86 11
```

To undelete a file or files (i.e., to remove the deletion mark), select them, then choose the UNDELETE command. The lines through the files are removed, and the tally of deleted files is updated. The UNDELETE command has a single subcommand, UNDELETE ALL FILES, which undeletes all the files in the browser, independently of whether they are selected. This is useful if you completely change your mind about deleting any files.

```
FB Commands
   Delete    ▶
   Undelete  ▶ Undelete ALL Files
   Copy      ▶
   Rename
   Hardcopy  ▶
```
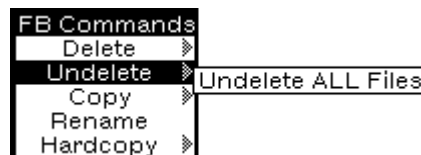
The DELETE command has a useful subcommand, DELETE OLD VERSIONS. When you have been editing a file in the text editor and performing repeated PUT commands, or you the programmer have done many MAKEFILEs of the same file, multiple versions of the file accumulate, each more recent version denoted by a higher version number. The DELETE OLD VERSIONS command is used to delete excess versions of the files displayed in the browser.

```
FB Commands Delete Selected Files
   Delete    ▶ Delete Old Versions
   Undelete  ▶
   Copy      ▶
   Rename
   Hardcopy  ▶
```

To use this command, press the mouse down on the DELETE command and slide the cursor out to the right, choosing the DELETE OLD VERSIONS command. Unlike the DELETE command (or DELETE SELECTED FILES, the equivalent subcommand), the DELETE OLD VERSIONS command operates on all the files in the browser. FileBrowser prompts you for the number of versions of each file that you wish to retain. It offers the default of one version. You can accept the default or you can type a different number of your choosing, followed by a carriage return. FileBrowser then marks for deletion all but the most recent N versions of all the files in the browser, where N is the number you specified. Before issuing the EXPUNGE command, you can, if you wish, scroll

through the browser, undeleting any particular files for which you wish to retain more versions than you specified.

The `DELETE OLD VERSIONS` command is sometimes useful even when you are not planning to actually expunge the files. This is because of the way extending the selection avoids deleted files (see the section "Selecting Files," above).

For example, if you wanted to copy only the most recent version of all the files in the browser to another location, you could do the following:

1. Use the `DELETE OLD VERSIONS` command, retaining just one version. This marks deleted all files but the newest version of each.

2. Go to the start of the browser and select the first file, then scroll to the end of the browser and press the right mouse button to extend the selection to the end of the browser. You have selected exactly the newest version of each file.

3. Use the `COPY` command to copy those files.

4. Finally, use the `UNDELETE ALL FILES` command to undelete all the old versions.

COPY

The `COPY` command is used to copy an entire file or set of files to another file system location; for example, from your disk to a file server. Select the file(s) you wish to copy, then select the `COPY` command. FileBrowser prompts you to supply a destination.

If you selected just one file, FileBrowser prints the old name and offers a default, which consists of the same file name and either the same directory that was last used in a COPY or RENAME in this FileBrowser, or the connected directory if this is the first use of COPY in this FileBrowser. You can accept the default or supply your own destination file name. If you supply just a directory specification, e.g., `{SERVER}<DIRECTORY>`, the file is copied to that directory under its current name. If you supply a complete name, the file is copied to that exact name.

```
Copy file {ERINYES}<LISP>LYRIC>LIBRARY>CHAT.LCOM;1 to new file
name: {DSK}CHAT.LCOM

{ERINYES}<Lisp>Lyric>Library>*.lcom;* at 11:13  FB Commands
Total: 99 / 4803 pages    Deleted: 0 / 0 pages       Delete      »
                                                      Undelete    »
 Name                     Pages   Created            ////Copy////»
  CHARCODETABLES.LCOM;1      8    29-Nov-86 17        Rename
▶ CHAT.LCOM;1               59    26-Jan-87 22        Hardcopy    »
  CHATTERMINAL.LCOM;1       20    27-Nov-86 13        See         »
  CMLFLOATARRAY.LCOM;1      23     9-Apr-87 16        Edit        »
  COPYFILES.LCOM;1          16    27-Nov-86 15        Load        »
```

Note: Unless you specify a version number in the destination file name, the version number of the new file will be 1, or one higher than the highest existing version of the file in the destination directory, independent of the version number of the old name.

Even files marked for deletion can be copied.

If you selected several files, FileBrowser notes how many files you wish to copy and offers as a default destination the connected directory. You can accept the default or supply a different directory. All the files are copied to that directory under the names they currently have.

You must supply a directory specification, e.g., `{SERVER}<YOURDIRECTORY>`, rather than a complete file name, since you can't copy multiple files to the same name. If you

mistakenly type a file name, rather than a directory specification, FileBrowser will complain and abort the command.

If you want to copy files from different subdirectories, FileBrowser will ask, via a message in its prompt window, if you want to preserve the subdirectory structure at the destination. If you answer YES, then the names at the destination will include not just the root name of each source file, but also all the subdirectory names below the greatest subdirectory prefix common to all the selected files (this common prefix is displayed as part of the question). If you answer NO, then the names at the destination are formed solely from the root name of each file (the name displayed in the browser), ignoring any directory information each name might have. This can cause multiple files with the same root name to be copied into the same destination name (but with different version numbers, of course).

```
Copy 15 files to which directory? {ERINYES}<Medley>Library>▲

{ERINYES}<Lisp>Lyric>Library>*.lcom;* at 11:13  FB Commands
Total: 99 / 4803 pages    Deleted: 0 / 0 pages      Delete    ≫
                                                     Undelete  ≫
  Name                   Pages   Created           //////Copy//////≫
▶KERMIT.LCOM;1             83    8-Jan-87  19        Rename
▶KERMITMENU.LCOM;1         17    8-Jan-87  19        Hardcopy  ≫
▶KEYBOARDEDITOR.LCOM;1     56    8-Jan-87  19         See      ≫
▶MASTERSCOPE.LCOM;1       124   11-Feb-87  15         Edit     ≫
▶MATCH.LCOM;1             71    8-Jan-87  18         Load     ≫
▶MATMULT.LCOM;1           35   22-Apr-87  10         Compile  ≫
 MINISERVE.LCOM;1         10   17-Apr-87  11        Expunge
▶MSANALYZE.LCOM;1         40   10-Dec-86  16        Recompute ≫
▶MSPARSE.LCOM;1           41   12-Jan-87  17         Sort
 NSCHAT.LCOM;1            32   27-Nov-86  13
 NSMAINTAIN.LCOM;1        24   28-Jan-87  11       --Abort--
 PRESS.LCOM;1             80   25-Mar-87  11
```
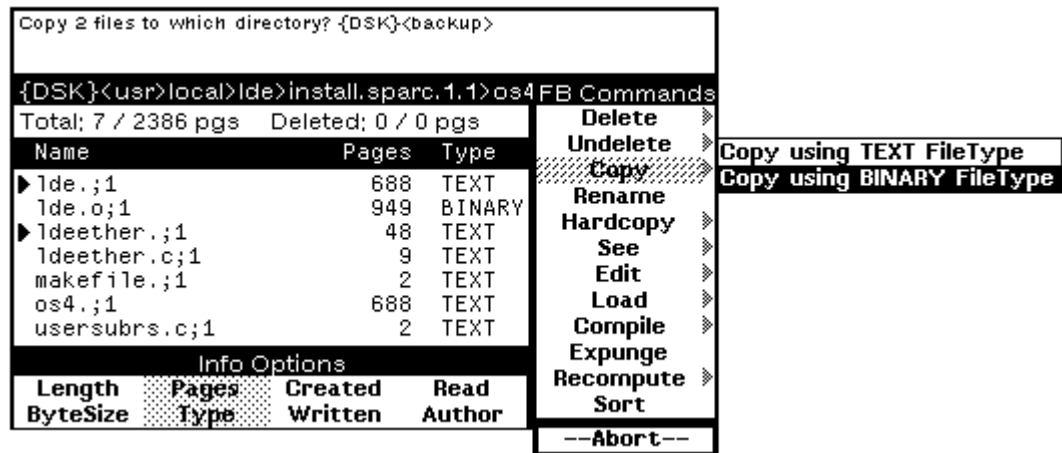
When copying (or renaming) multiple versions of the same file, FileBrowser does the copying in order of increasing version number, so that the versions at the destination are in the same relative order as at the source.

As each file is copied, FileBrowser prints a message giving the full name of the new file. If a file with the chosen name already exists, the new file's version number will be one higher; otherwise it will be version 1 (one). The new file will have the same creation date as the original file. If the destination file happens to be one that matches the pattern of the files in the browser, the new file is inserted in the appropriate place in the browser display. However, if it matches the pattern of some other FileBrowser, it is not inserted in that other browser's display (in other words, FileBrowsers do not know about each other). You would have to RECOMPUTE the destination FileBrowser to see that the file was copied into it.

Copying files from filesystems that do not normally distinguish between **TEXT** and **BINARY** file types , like UNIX, can sometimes cause problems. On these systems the type of a file is inferred from its extension, or by using the Lisp variables **DEFAULTFILETYPE** and **DEFAULTFILETYPELIST**. At times, there will be files where the system cannot properly infer the type. By selecting the submenu of COPY you can copy files with the right attributes.

```
Copy 2 files to which directory? {DSK}<backup>


{DSK}<usr>local>lde>install.sparc.1.1>os4  FB Commands
Total: 7 / 2386 pgs    Deleted: 0 / 0 pgs        Delete      ⟫
Name                    Pages   Type          Undelete    ⟫    ┌──────────────────────────────┐
▶ lde.;1                  688   TEXT          ▒▒Copy▒▒▒  ⟫  │ Copy using TEXT FileType    │
  lde.o;1                 949   BINARY        Rename        │ Copy using BINARY FileType  │
▶ ldeether.;1              48   TEXT          Hardcopy    ⟫    └──────────────────────────────┘
  ldeether.c;1              9   TEXT            See       ⟫
  makefile.;1               2   TEXT            Edit      ⟫
  os4.;1                  688   TEXT            Load      ⟫
  usersubrs.c;1             2   TEXT          Compile     ⟫
                                              Expunge
        Info Options                        Recompute   ⟫
  Length   ▒Pages▒  Created     Read          Sort
  ByteSize ▒Type▒   Written    Author      ──Abort──
```

In this example, since the chosen files had no extension, the system incorrectly inferred
their type from the value of **DEFAULTFILETYPE**. By selecting the **BINARY** copy option,
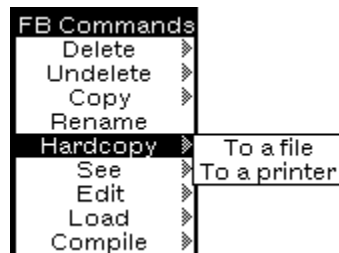the correct type of the file will be preserved after the operation.

RENAME

The RENAME command is used for changing the name of a file or group of files, or for moving a file or group of files to a different directory.

The RENAME command is used in exactly the same way as the COPY command. If you rename a single file, you can supply a complete new name or just a directory; if you rename several files, you must specify a directory. As each file is renamed, FileBrowser prints a message giving the file's new name and removes the file from the browser display. If the new name belongs in the same browser, it is inserted in the appropriate place. If for some reason a file could not be renamed, this is noted in the FileBrowser prompt window. The reasons for the failure of a renaming operation are roughly the same as for the failure of an EXPUNGE; the file is open, or you do not have the access rights needed to rename the file.

Note: If the destination of the rename is on a different file system than the original file, changing its name is equivalent to copying the file to its new name and then deleting the original file.

HARDCOPY

You can print text files, TEdit files, Interpress or Press files, and Lisp files from FileBrowser. Select the appropriate file or files, then select the HARDCOPY command. The HARDCOPY command will determine what type of file you are printing and call the appropriate function for printing that file. Then the files will be printed one at a time on your default printer. The prompt window will display status messages telling you when files are being printed and when they are done (if your printer is one that provides this status service).

```
FB Commands
   Delete    »
 Undelete    »
    Copy     »
  Rename
 Hardcopy    »|   To a file
    See      »|To a printer
    Edit     »
    Load     »
  Compile    »
```

You may specify printing to a file or to a printer other than the default printer by means of a submenu from the HARDCOPY command. This menu is the same as the one on the HARDCOPY command in the background menu. Selecting TO A PRINTER presents you with a choice of printers from a menu. Selecting TO A FILE prompts you to supply a file name. If you selected a single file, you must specify a single hardcopy file name (or accept the offered default). If you selected multiple files, then you must specify a pattern with a single asterisk somewhere in the "name" field, for example, *.INTERPRESS or Hardcopy-*.IP. The output file names are constructed by merging the pattern with each selected file name. If the name includes an extension that implies the type of print format (e.g., .IP or .INTERPRESS implies the InterPress print format), then a file of the specified type is made automatically. Otherwise, you are prompted to supply a print format type.

Note: For files stored on servers not supporting random access, FileBrowser is currently unable to determine that a file is in TEdit format unless the file has the extension .TEDIT. Therefore you should use TEdit to hardcopy TEdit files with other extensions. Use FileBrowser's EDIT command (to call TEdit), then the HARDCOPY command either from the TEdit Expanded Menu or from the right-button menu. As of the Lyric release, TEdit files written directly to an NS

file server are identifiable as TEdit files, so this restriction does not apply to them.

Note: To obtain a hardcopy of the directory itself, use the Hardcopy command from the right-button window menu. See the Getting Hardcopy Directory Listings section.

SEE

When you browse a directory you sometimes want to see a file before printing or performing some other operation on it. To do this, select the file, then select the SEE command from the command menu. FileBrowser will prompt you to open a window by presenting you with a dashed rectangle and printing a message in the system prompt window. The window will be blank until FileBrowser starts printing the contents of the file in it.

There are actually four different SEE commands, as shown in the submenu for the SEE command. The two FAST SEE commands are provided to let you quickly see the contents of a file, but not do anything fancy, such as scroll around at random in the file. The slower SCROLLABLE & PRETTY command does let you scroll, and if the file contains formatting information of a kind that FileBrowser knows about (via the editors you have loaded), you will see the file formatted. However, this command does much more work, and may take a bit longer to show you even the first line of the file. The FILEBROWSE command is for use on "files" that are actually directories; it is described in the next section.
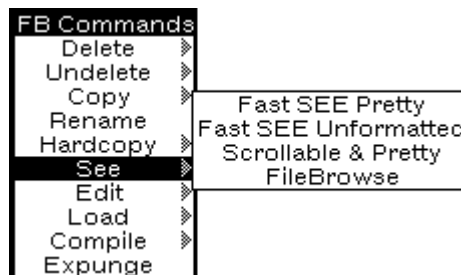
The two FAST SEE commands display the selected file in the display window one windowfull at a time. When the file fills the window, a small menu appears at the bottom-left corner of the window (or top-left if your display window is at the bottom of the screen) giving you the option of seeing more of the file or aborting the SEE command. If you issued the SEE command with more than one file selected, you also have the choice of aborting just the display of this file or the entire SEE command.

```
Viewing {ERIS}<Lisp>Koto>Lispusers>ACTIVEREGIONS,;1
(FILECREATED " 5-JUL-84 17:48:34" {SDRVX1}DISKD:<DOLPHIN.LISPUS
ERS>ACTIVEREG.;8 7633
    changes to: (FNS ACTIVEREGIONS/MULTIPLEREGIONS? FINDACTIVEREGIO
N
            ACTIVEREGIONS/DEFAULTHIGHLIGHTFN)
        (VARS ACTIVEREGIONSFNS ACTIVEREGIONSHIDDENFNS)
    previous date: "15-MAR-84 08:43:50" {SDRVX1}DISKD:<DOLPHIN.LISP
USERS>ACTIVEREG.;5)  **COMMENT**
(PRETTYCOMPRINT ACTIVEREGIONSCOMS)
(RPAQQ ACTIVEREGIONSCOMS (  **COMMENT**
            (RECORDS ACTIVEREGION)
            (FNS * ACTIVEREGIONSFNS)  **COMMENT**
            (FNS * ACTIVEREGIONSHIDDENFNS)))
  **COMMENT**
[DECLARE: EVAL@COMPILE
(RECORD ACTIVEREGION (REGION HELPSTRING DOWNFN UPFN HIGHLIGHTFN LO
WLIGHTFN DATA))
]
```

```
  More    Next File    Abort
```

If you select More, the SEE command displays another windowful of the file. If you select Next File, the SEE command closes this file and goes on to display the next file in the current selection. If you select Abort, the entire SEE command is aborted. You can also abort the SEE command by closing the display window.

The next time you give a FAST SEE command, the same window will be reused.

The only difference between the FAST SEE PRETTY and the FAST SEE UNFORMATTED commands is the manner in which the characters of the file are processed as they are displayed.

The pretty (formatted) version interprets certain control characters found in Lisp source files to be font change commands, and interprets certain multibyte sequences as representing characters in the Xerox extended character set (see *XSIS Character Code Standard,* version *XC1-2-2-0*). It also squeezes out blank lines and shrinks the indentation of indented lines in order to better fit the text in a window that is generally much narrower than the standard file width. The formatted version is thus most appropriate for viewing source files and files containing plain text.

The unformatted version of the SEE command does no special processing on the characters whatsoever. It simply displays each eight-bit byte as a single character, uninterpreted. This means that bytes that do not represent normal printing characters may be displayed as black boxes, in the form ^x or #x, or as a flashing of the window (for the byte that represents the ASCII "bell" character). The Unformatted version is thus most appropriate for viewing binary files that also contain text portions that might be worth seeing; e.g., compiled files (those with extension .MCOM) or Interpress masters (extension .IP or .INTERPRESS).

The SEE SCROLLABLE & PRETTY command views a file in a different way. This command brings up a new read-only TEdit window for viewing a file (only if TEdit is loaded in your system; otherwise, SCROLLABLE & PRETTY reverts to FAST). You can scroll and copy-select the file's contents at will, as with any TEdit window. If the file is a Lisp source file, its contents are first formatted into a TEdit document, so that all the font information is retained. This formatting, however, can take a long time for a large file. For other kinds of files, the SEE SCROLLABLE & PRETTY command is exactly like viewing the file in a regular TEdit window, except that you can't edit it. If you want to edit a file, use the Edit command instead of the See command.

You can keep the display window used by the SEE SCROLLABLE & PRETTY command open as long as you like. The command uses a different window for each file

you select. Simply close the window with the standard right-button window menu when you are finished with it.

FILEBROWSE

The FILEBROWSE command is a subcommand of SEE used to view a subdirectory in its own FileBrowser window. The selected file must be a (sub)directory. Subdirectory files appear in browsers on XNS file servers when the depth is finite (see the SET DEPTH command), and their names always end in ">".

```
Name (depth 1)        Pages  Created        Undelete  ⚑
  Examples>             139   4-Feb-88 12    Copy      ⚑ ┌─────────────────────┐
  Lisp>                   1  15-Sep-84 15    Rename    ⚑ │   Fast SEE Pretty   │
▶ Mail>                 685  26-May-87 11    Hardcopy  ⚑ │Fast SEE Unformatted │
  Misc>                   1  17-Jul-87 17    See       ⚑ │  Scrollable & Pretty│
  Star>                 273   1-Jul-87 17    Edit      ⚑ │     FileBrowse      │
                                             Load      ⚑ └─────────────────────┘
```

On UNIX servers, subdirectories are not syntactically distinguishable from ordinary files, nor is Lisp able to distinguish them internally; you simply have to know. The FILEBROWSE command prompts you for a region for a new FileBrowser window group, in which it proceeds to enumerate the contents of the selected subdirectory, to the same depth as the main browser used, if any.

EDIT

The EDIT command invokes an editor on the selected file. To specify an editor explicitly, use one of the commands on the submenu.

```
Hardcopy  ⚑
See       ⚑ ┌──────────┐
Edit      ⚑ │  TEdit   │
Load      ⚑ │ Lisp Edit│
Compile   ⚑ └──────────┘
```

To start up a TEdit editor on a selected text file, select EDIT with the left mouse button. If you have recently closed a TEdit window, then TEdit will probably reuse that window; otherwise, you will be prompted to create an editor window. TEdit only remembers the most recently abandoned window, however, so if you issue the EDIT command when you have several files selected, you will be prompted to create windows for all but the first file.

The subcommand LISP EDIT is appropriate for Lisp source files produced by the file manager. It calls the Lisp structure editor on the file's coms. If the file is not yet known to the file manager, you will be asked whether you want to load it first (using LOAD PROP). If not, the operation is aborted. The editor used is the default structure editor (SEdit or DEdit, depending on your setting of *EDITMODE*).

If you select the main Edit command, without sliding off to the submenu, then FileBrowser's default editor is called. This editor is initially TEdit, but you can change the default behavior by setting the variable FB.DEFAULT.EDITOR (see the section "Customizing FileBrowser and Using the Programmer Interface," below).
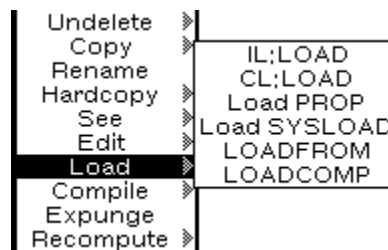
LOAD

FileBrowser's LOAD command can be used to load both source (interpreted) and compiled files into your workstation's virtual memory. First select the file or files you want to load, then select Load with the left mouse button.

A special display window is opened to give information about the files as they are loaded. When the load is complete, FileBrowser closes the load window.

```
TTY window for LOAD

{ERINYES}<LYRIC>LIBRARY>KEYBOARDEDITOR.LCOM;1
compiled on  8-Jan-87 19:03:57
File created 21-Sep-85 08:03:04
KEYBOARDEDITORCOMS

{ERINYES}<LYRIC>LIBRARY>VIRTUALKEYBOARDS.LCOM;1
compiled on  6-Jan-87 22:41:02
File created  5-Nov-86 16:55:40
VIRTUALKEYBOARDSCOMS
```

The LOAD command also has subcommands that enable you to load files in different ways. These commands are described briefly here; see the *IRM* for further details. Only the LOAD and LOAD SYSLOAD commands are of interest to nonprogrammers. All load commands are placed on the history list. With the exception of LOAD SYSLOAD, all are undoable using Lisp history list commands.

```
Undelete  ▶
  Copy    ▶
 Rename   ▶      IL:LOAD
Hardcopy  ▶      CL:LOAD
   See    ▶     Load PROP
  Edit    ▶   Load SYSLOAD
  Load    ▶     LOADFROM
 Compile  ▶     LOADCOMP
 Expunge
Recompute ▶
```

LOAD (same as the Lisp LOAD command) loads the file with LDFLG = NIL. If any functions or variables in the file redefine ones that are already in memory, messages such as "(FOO redefined)" are printed. This command is used for loading source files that you as a programmer plan to make modifications to, or for loading any file that you have doubts about, in which case you might want to be able to undo the load.

CL:LOAD differs from IL:LOAD in that it calls the Common Lisp LOAD function, rather than that of Interlisp. In the present implementation, these two commands are essentially identical.

LOAD PROP loads the file(s) with LDFLG = PROP. Interlisp functions are loaded onto property lists, rather than redefining the functions already in memory. Common Lisp functions, macros, etc. are loaded into a definitions table, again without changing the definitions currently in effect. This command is used for loading Lisp source files for which the compiled version already exists in memory, but which you plan to edit.

LOAD SYSLOAD loads the file(s) with LDFLG=SYSLOAD. Function and variable redefinitions occur quietly (i.e., without printing (FOO redefined), (BAR reset), etc.). The file manager is not informed of this file. This is the fastest loading command and consumes the fewest resources, but it is not undoable. It is the best way to load compiled (extensions LCOM or DFASL) files that you are certain you want to load into your environment and are not planning to edit.

LOADFROM calls the file manager's function LOADFROM. This loads variables and other expressions but not Interlisp functions, and does so in a way that informs the file manager, so that the editor knows where to find the functions.

Note: The LOADFROM command is not appropriate for Common Lisp files—it is better to use LOAD PROP for them.

The LOADCOMP subcommand calls the file manager's function LOADCOMP. This loads from the file all the expressions that the compiler would evaluate if compiling the

file—macros, records, and any other expressions enclosed in an EVAL@COMPILE declaration.

COMPILE

The COMPILE command is used to compile a selected Lisp source file or files. The files do not have to be loaded. The COMPILE command uses the same compiler as CLEANUP does (the value of *DEFAULT-CLEANUP-COMPILER*), unless you select a different compiler from the COMPILE submenu.
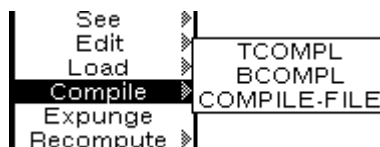
Note that this command is placed on the history list, so that it and its subcommands are undoable.

A special Executive window is opened for each file to display information about the code being compiled. When the compilation is finished, the window is closed. In the case of TCOMPL and BCOMPL, which invoke the Interlisp compiler, each compiled file is saved on your connected directory with the original file name and the extension MCOM. In the case of COMPILE-FILE, which invokes the Xerox Common Lisp compiler, the compiled file is saved on the same directory as the source file with the original file name and the extension MFASL.

Note that this command compiles files found on a storage device, not the functions defined in the Lisp image. If you have made changes to any of the functions on a loaded file, you must perform a MAKEFILE to write an updated version before compiling it. For more information on making files and compiling, see the *IRM*.

EXPUNGE

If you are sure you want to delete files permanently, choose the EXPUNGE command. The EXPUNGE command is grayed while FileBrowser expunges the files that were marked for deletion by the DELETE command. As each file is removed from the system, it is removed as well from the browser display, and the tally of total number of files and number of deleted files is updated, so you can see the progress of the command.

If for some reason a file can not be expunged, FileBrowser prints a message saying so in its prompt window, but continues to expunge the other files. The main reasons that prevent a file from being expunged are its being opened, either by you or some other user, or your not having the access rights required to delete it (if it is on a file server). See the section "Troubleshooting Problems with FileBrowser," below.

Note: The EXPUNGE command is not affected by the current selection; it operates only on files marked for deletion, whether currently selected or not.

RECOMPUTE

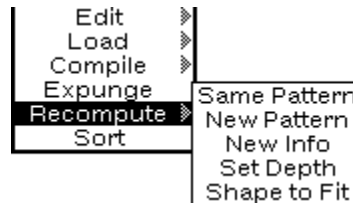FileBrowser's display shows those files that existed and matched the specified pattern at the time you created the browser. If you want the browser to reflect the latest state of the file system, use the RECOMPUTE command.

For example, if you open a FileBrowser on your directory, then save several versions of a TEdit file on that directory, the file listing will not display the new versions until you RECOMPUTE.

The RECOMPUTE command operates exactly as when you started up FileBrowser initially: it clears the display and tally windows, then enumerates the files matching the pattern. The RECOMPUTE command in the menu is grayed until the enumeration is finished. During this time you cannot scroll or perform any other operations on the browser. However, you can close the window if you want to abort the command and throw away the browser.

If any files are marked for deletion at the time you request a RECOMPUTE, FileBrowser will present the choice of expunging or undeleting the files, just as it does when you want to quit the browser (see the section "Quitting the FileBrowser," above).

The RECOMPUTE command also has a menu of subcommands that allow you to list different files or different information for the same set of files.

```
Edit        »
Load        »
Compile     »
Expunge      ┌──────────────┐
Recompute » │ Same Pattern │
Sort        │ New Pattern  │
            │ New Info     │
            │ Set Depth    │
            │ Shape to Fit │
            └──────────────┘
```

SAME PATTERN is the same as the main RECOMPUTE command, i.e., it enumerates the files matching the same pattern as before.

NEW PATTERN lets you change the pattern, i.e., browse a new set of files. FileBrowser prompts you to supply a new file name pattern and offers the old pattern as an initial default. You can either type an entirely new pattern, replacing the one offered, or delete the old pattern one character at a time by backspacing. Press the carriage return when you have finished specifying the pattern. FileBrowser then enumerates the files matching this pattern, just as with the RECOMPUTE command. You can abort the command with the Abort button, or by erasing the whole pattern (by backspacing or using control-Q) and then pressing the carriage return.

NEW INFO lets you change which attributes the browser displays. It is described in the next section.

SET DEPTH lets you change the depth to which FileBrowser enumerates a directory on an XNS file server. It is described in the section "SET DEPTH".

SHAPE TO FIT reshapes the FileBrowser window so that all the attributes in the display are visible at once, eliminating the need to horizontally scroll the window to get at all the information.

NEW INFO

FileBrowser displays some file attributes, or information about the file, alongside each file in the browser display. Ordinarily, the attributes displayed are the size of the file in pages, its creation date, and its author. You can change which attributes are displayed for all new FileBrowsers by changing FB.DEFAULT.INFO (see the section "Customizing FileBrowser and Using the Programmer Interface," below). You can change the attributes displayed in a particular browser window by using the NEW INFO command.

To use the NEW INFO command, select it from the submenu of the RECOMPUTE command. FileBrowser opens up an additional subwindow, the Info Options window, below the display window. This subwindow contains a menu of attributes, with the current defaults shaded. Selecting a shaded item unshades it; selecting an unshaded item shades it. When you have selected all the attributes you wish to see displayed, issue either the RECOMPUTE or the NEW PATTERN command. The files will be

listed with the new information you requested. The Info Options window stays open—you can close it at any time.

```
Select from the lower menu which attributes are to be displayed,
then click Recompute

{DSK}*.*;* browser                          FB Commands
Total: 48 / 2838 pgs    Deleted: 0 / 0 pgs       Delete    ≫
                                                 Undelete  ≫
  Name                     Pages   Create        Copy      ≫
                                                 Rename
  EDITBITMAP.TEDIT;1          76   20-Ma         Hardcopy  ≫
  ETHERRECORDS.TEDIT;1        15   20-Ma         See       ≫
▶ FILEBROWSER.TEDIT;2        388    3-Ap         Edit      ≫
  FONTSAMPLE.TEDIT;1          17   20-Ma         Load      ≫
  FTPSERVER.TEDIT;1           13   20-Ma         Compile   ≫
  FX-80DRIVER.TEDIT;2         88    8-Ap         Expunge
  FX-80DRIVER.TEDIT;1         88   20-Ma         Recompute ≫
  GCHAX.TEDIT;1               40   20-Ma         Sort
  GRAPHER.TEDIT;1             67   20-Ma
                    Info Options
  Length    Pages    Created      Read
  ByteSize  Type     Written      Author
```

The Info Options items have the following meanings:

CREATED   The date and time that the content of the file was created. This date changes whenever the file is modified, but does not change when a file is copied or renamed.

WRITTEN   The date and time the file was last written to the file system. This date is never older than the Created date, but it can be newer if the file is copied, unmodified, from one file system to another.

READ   The date and time the file was last read. This attribute may be blank if the file has never been read.

AUTHOR   The login name of the person who wrote the file, or last modified it.

LENGTH   The length of the file in (usually eight-bit) bytes.

PAGES   The number of 512-byte pages in the file. On some servers, this attribute is blank if the file is empty.

BYTESIZE   The size (in bits) of the bytes in the file. In Xerox Lisp this is always eight, but some older computers and file servers allow other sizes.

TYPE   A value indicating what kind of data the file contains. The usual values of this attribute are TEXT, meaning the file contains just characters, or BINARY, meaning the file contains arbitrary data. Some servers have additional types, such as INTERPRESS for files in Interpress format.

SET DEPTH

XNS file servers support a feature that allows enumerating a directory to a user-specifiable depth. The "depth" of a file reflects the number of subdirectories between it and the root of the enumeration, i.e., the directory or subdirectory you gave in the pattern to FileBrowser, not counting any containing wildcards (asterisks). The immediate descendants of the root are at depth 1, files in subdirectories of depth 1 are at depth 2, and so on.

Ordinarily, FileBrowser enumerates a directory to the default depth, which is usually unlimited. To enumerate a directory to a different default, use the FB command with argument :DEPTH *n*, for some positive integer *n*, or T for unlimited depth. To change

the depth in an existing FileBrowser, use the SET DEPTH command, a subcommand under the RECOMPUTE command. The command offers you a menu of choices:

```
Global default
   Infinite
      1
      2
    Other
```

"Global default" means use the default depth, overriding the depth at which this browser was last enumerated. "Infinite" means use no depth limit (same as depth T). "1" and "2" are common depth choices; to choose some other numeric value, select "Other" and enter the value via the displayed keypad.

The SET DEPTH command does not affect the current display. It takes effect the next time you use the RECOMPUTE or FILEBROWSE commands from the same browser.

During a RECOMPUTE, if a subdirectory appears at the specified maximum depth, its descendants are not enumerated; rather, the subdirectory itself appears as an entry in the browser display. This entry can be selected, just like a file, but only a small number of commands can be used on it: you can RENAME it, you can DELETE it if it has no descendants, and you can FILEBROWSE it. It has attributes, just as ordinary files do. Its page size is the size of the entire subtree rooted at the subdirectory.

Note: Depth currently affects only XNS servers; all other devices ignore it and enumerate to their own default depths. In addition, due to a bug in XNS Services 10, depth is ignored for nontrivial patterns, i.e., anything but "*.*".

SORT

The SORT command allows you to sort the files in the browser by any attribute of the files displayed in the browser.   Selecting SORT brings up a menu of attributes by which to sort. This menu includes all the attributes currently displayed in the browser (such as Creation Date, Author), plus the choice Name. For some attributes you can sort forward or backwards; the choice is on a submenu, and the default is generally in the order of numerically greatest (e.g., size) or most recent (e.g., creation date) first.

If the attribute you select is not Name, then the file names displayed in the browser will be reformatted to include their directory portion (if there are any subdirectories below the browser's main pattern), as the subdirectory information is no longer implicit in a file's position in the browser.

The sort order Name, Decreasing Version is the default order in which browsers initially are created.

Customizing FileBrowser and Using the Programmer Interface

FileBrowsers are created programmatically by the function FILEBROWSER, or by type-in from an Executive window with the FB command.

Note: All functions, variables and literals in this section are in the Interlisp package. You'll have to use the prefix IL: if you are using another Executive.

FileBrowser Functions

(FILEBROWSER *FILESPEC ATTRIBUTES OPTIONS*)   [Function]

Creates a FileBrowser on files matching the pattern *FILESPEC* and displaying the values of the specified *ATTRIBUTES* for each file.   Returns the main window of the browser.

If *ATTRIBUTES* is missing or `NIL`, it defaults to the value of FB.DEFAULT.INFO (below). See `FB.INFO.MENU.ITEMS` for the complete set of allowable attributes.

OPTIONS is a list in property-list format. The currently implemented properties and their values are as follows:

`:REGION`	A screen region in which FILEBROWSER will open the browser; if this option is omitted, FileBrowser will prompt you for a region.
`:DEPTH`	The depth to which the enumeration should be performed. Affects only XNS servers (see SET DEPTH).
`:TITLE`	The title for the main browser. If this is omitted, the title derives from *FILESPEC*, and is updated whenever the RECOMPUTE command is used.
`:MENU-TITLE`	The title for the command menu. Defaults to "FB Commands".
`:MENU-ITEMS`	The set of ITEMS composing the command menu. The default is the value of FB.MENU.ITEMS.

`FB` *FILESPEC ATTR1 . . . ATTRN*   [Command]

This is an Executive command for creating FileBrowsers. FB creates a FileBrowser on files matching the pattern *FILESPEC* and displaying the attributes *ATTR1* through *ATTRN*, or the value of `FB.DEFAULT.INFO` if no attributes are specified. It prompts you for a window region.   If a keyword appears in the command line, the remainder of the line from that point on is interpreted as the *OPTIONS* argument to FILEBROWSER.

For example, the Executive command

```
FB *.MCOM LENGTH CREATIONDATE
```

browses all files on the connected directory with extension MCOM, displaying the length in bytes and creation date for each.   The command

```
FB * :DEPTH 1
```

browses the connected directory to depth 1, displaying the default attributes.

FB always returns `NIL`.

FileBrowser Variables

There are several global variables that can be altered to affect FileBrowser's behavior. You can set them by typing

```
(SETQ VARIABLENAME NEWVALUE)
```

to your Executive window, and can save their values with a VARS command in your initialization file.

`FB.DEFAULT.INFO`   [Variable]

A list specifying which attributes should be displayed for each file. The elements of this list are the Lisp names for the attributes you want displayed.

The choices are CREATIONDATE, WRITEDATE, READDATE, LENGTH, SIZE, BYTESIZE, AUTHOR, and TYPE. The attribute SIZE corresponds to the info item "Pages".

For example,

```
(SETQ FB.DEFAULT.INFO '(CREATIONDATE LENGTH))
```

would cause all new FileBrowsers to display exactly the attributes creation date and length in bytes for each file.

FB.INFO.MENU.ITEMS   [Variable]

The list of items in the menu used by the NEW INFO command. Each element of the list is of the form (*LABEL ATTRIBUTE* "*DOCUMENTATION*"). If you add new attributes to this variable, you should also add corresponding entries to FB.INFO.FIELDS.

FB.INFO.FIELDS   [Variable]

A list describing, for each attribute, the format in which it is displayed. In addition, the order of attributes in this list is the order in which they are displayed in a browser window. Each element is of the form (*ATTRIBUTE HEADER WIDTH FORMAT PROTOTYPE*), where

> *ATTRIBUTE* is the name of the attribute, which must be a valid attribute for GETFILEINFO
>
> *HEADER* is a string displayed in the header line above the main browser window
>
> *WIDTH* is the total width in pixels to allocate for printing the values of this attribute, including trailing space
>
> *FORMAT* is NIL for ordinary values, DATE for attributes whose value is a date, or (FIX *N*) for integer values
>
> *PROTOTYPE* is a string describing the widest value you expect the field to have

All values are printed left-justified in the allotted space, except for format (FIX *N*), used for integer values, which are right-justified in a field *N* pixels wide, with *WIDTH–N* pixels left for trailing space.

If *PROTOTYPE* is present, then the *WIDTH* and *N* fields for the item are ignored, and the width is taken to be the width of the prototype string in the browser's font (FB.BROWSERFONT), plus two characters' worth of space between columns.

FB.DEFAULT.NAME.WIDTH   [Variable]

The amount of space, in points, to use for displaying file names in a browser, initially 140. The name column is automatically expanded if enough names are too wide. You can set this larger if you routinely browse directories of long file names.

FB.ICONFONT   [Variable]

The font in which the file pattern is displayed on the browser icon, initially eight-point Helvetica. The value of this variable should be a font descriptor, as returned by FONTCREATE.

For example,

```
(SETQ FB.ICONFONT (FONTCREATE 'MODERN 10 'BOLD))
```

FB.BROWSERFONT   [Variable]

The font in which the information in the main display window is printed, initially 10-point Gacha.

FB.PROMPTFONT   [Variable]

The font in which prompt messages are printed, initially eight-point Gacha.

FB.PROMPTLINES   [Variable]

The number of lines in the prompt window, initially three.

FB.HARDCOPY.FONT   [Variable]

Specifies the font to use when producing hardcopy directory listings. Initially NIL, which means to use the font class DEFAULTFONT.

FB.HARDCOPY.DIRECTORY.FONT   [Variable]

Specifies the font to use for subdirectory names, if there are any, in hardcopy directory listings. Initially NIL, which means to use the font class ITALICFONT.

FB.DEFAULT.EDITOR   [Variable]

Specifies the editor to call by default when the main Edit command is selected. Its value is one of the following:

TEDIT   Use the TEdit text editor.

LISP   Use the Lisp structure editor, as in the Lisp Edit subcommand.

NIL   Use the Lisp structure edior if the selected file is a File Manager Lisp source file, TEdit otherwise.

Other   Names the entrypoint function of the editor of choice. The editor is called with a single argument, the file name.

The initial value of FB.DEFAULT.EDITOR is TEDIT.

FILING.ENUMERATION.DEPTH   [Variable]

The system variable that controls the default depth to which a directory is enumerated. The value is a positive integer, or T for unlimited depth. The initial value is T.

Adding FileBrowser Commands

You can add your own commands to FileBrowser by adding items to FB.MENU.ITEMS and writing functions to handle the commands. This section describes the format of menu commands and a set of functions that are useful for the implementation of FileBrowser commands.

FB.MENU.ITEMS   [Variable]

A list of the items that appear on FileBrowser's command menu. You can add new FileBrowser commands by adding new elements to the end of this list. After your change, any new FileBrowser will have the added commands.

Each element of `FB.MENU.ITEMS` is of the form (*LABEL YOURFN "EXPLANATION"*), where

> *LABEL* is the name of the command, as it is to appear in the menu;

> *YOURFN* is the name of the function to be called when the command is selected

> *EXPLANATION* is the explanation to be printed when the mouse cursor is held over the command

While *YOURFN* is Executing, the menu command is grayed out, and FileBrowser is "locked" so that no other commands or processes can access it.

You can have subcommands as well if you make the menu command be of the form (*LABEL YOURFN "EXPLANATION"* (*SUBITEMS item1. . . itemN*)), where each *itemJ* is recursively of the same form as a menu command.

FileBrowser calls *YOURFN* with four arguments: (*YOURFN BROWSER KEY ITEM MENU*), as follows:

> *BROWSER* is FileBrowser object in control of this browser window.

> *KEY* is the mouse key pressed (left or middle).

> *ITEM* is the menu item that was selected.

> *MENU* is the command menu.

YOURFN can also be a list of two elements, (*FN ARG*), where *ARG* is an arbitrary value that is passed, unevaluated, as the fifth argument to *FN*. This is useful for writing one function that implements several subcommands that are similar to the original command.

Any additions to `FB.MENU.ITEMS` should be saved with the file manager command `APPENDVARS`, so that the new items are added to the end of the menu, and your changes will not interfere with any changes to built-in FileBrowser commands in new FileBrowser releases.

(`FB.TABLEBROWSER` *BROWSER*)   [Function]

Returns the TABLEBROWSER object belonging to FileBrowser *BROWSER*. See the documentation for the module TABLEBROWSER for further operations you might perform on one of these browsers.

(`FB.SELECTEDFILES` *BROWSER NOERRORFLG*)   [Function]

Returns a list of table items representing the files currently selected in *BROWSER*. If there are no selected files, this prints

> **No files are selected**

in the prompt window, unless *NOERRORFLG* is true, in which case this function quietly returns `NIL`.

(`FB.FETCHFILENAME` *ITEM*)   [Function]

Returns the full name of the file denoted by *ITEM*, one of the table items returned by `FB.SELECTEDFILES`.

Troubleshooting Problems with FileBrowser

When FileBrowser returns the message "No files in group *FILENAMEPATTERN*" when you know those files exist, the file server is probably down or rejecting connections. If this is so, your only option is to wait until the server is functioning again, and then give the Recompute command. In the case of an NS file server, the enumeration of files can also fail if you do not have sufficient access privileges; this condition is usually noted by a message in the system prompt window.

When you try to expunge a file and FileBrowser displays the message `"Can't expunge FILENAME,"` it may be because you don't have write access to the file, or someone else is reading the file. However, the most common reason is that the file is still open. Be sure to close any TEdit windows in which you may still be viewing the file. If you have recently issued a HARDCOPY command for the file, a background process may still be working on the file. If that's not the problem, you can get a list of open streams by typing `(OPENP)` at the prompt in an Interlisp Executive window. If one of them is open on the file you want to delete, you can close it by passing the stream to the function CLOSEF. To close all open streams (not recommended unless you're sure it is okay), you can type `(MAPCAR (OPENP) 'CLOSEF)`. If you don't find an open stream, and the server is a Leaf or XNS file server, you may have a disagreement between Lisp and the server on what files are open, something that can occur if you had aborted an OPENSTREAM operation. Call `(BREAKCONNECTION "servername")` to reset the connection, then try again.

[This page intentionally left blank]