

File created: 21-May-2024 18:45:54 {LU}MANAGER.;4

edit by: mth

changes to: (FNS Manager.DO.COMMAND)

previous date: 20-May-2024 11:16:10 {LU}MANAGER.;3

Read Table: INTERLISP

Package: INTERLISP

Format: XCCS

## (RPAQQ MANAGERCOMS

```
[;; The Manager : a menu based interface to the file manager.
;; Originally written by: Jay Ferguson of Ford Aerospace & Communications Corp and Robert Noble of Intellicorp.
;; Rewritten by Larry Masinter, winter of 1986.
;; Further modifications & significant enhancements by Andrew J. Cameron III, summer of 1987.
;; Ongoing maintenance and performance tuning by Ron Fischer at Xerox AI Systems.
;;
;; There are two patches in here that should be removed if Xerox Lisp is fixed. The first is the advice (MARKASCHANGED :IN
;; DEFAULT.EDITDEFA0001) that removes a (mostly) redundant call to MARKASCHANGED in the editor interface, which otherwise slows
;; down manager updates. Somehow this call is not redundant when editing FILELST (perhaps there's a special case for FILELST or when
;; items are not in any existing files). The second is the fns Manager.REMOVE.DUPLICATE.ADVISE called by the advice on LOAD and
;; LOADFNS, which removes redundant advice which would otherwise pile up and cause massive slow downs in manager updates!
;;
;; The edit history is now kept in the file MANAGER.HISTORY.
;;
;; Known bugs and feature requests are now kept in the documentation file MANAGER.TEDIT.
;;
(SPECVARS Manager.ACTIVEFLG MANAGER-CASES MANAGER-ADDTOFILES?)
(GLOBALVARS LAMBDAFONT DEFAULTFONT MENUFONT BOLDMENUFONT MANAGER-WINDOWS LOADDBFLG SAVEDBFLG
  MANAGER-ITEM-OPERATION-COMMANDS MANAGER-ITEM-FILE-RELATION-COMMANDS
  MANAGER-FILE-OPERATIONS-COMMANDS MANAGER-FILE-FILE-RELATION-COMMANDS MANAGER-MAIN-MENU-ITEMS
  MANAGER-ACTIVITY-WINDOW-TITLE MANAGER-MAIN-WINDOW MANAGER-MAIN-ICONW Manager.WINDOW-ANCHOR
  MANAGER.BM MANAGER.BM.MASK BackgroundMenuCommands BackgroundMenu)
(VARS *UNMANAGED-TYPES* MANAGER-ACTIVITY-WINDOW-TITLE (MANAGER-CASES)
  (MANAGER-ADDTOFILES?)
  MANAGER-FILE-FILE-RELATION-COMMANDS MANAGER-FILE-OPERATIONS-COMMANDS
  MANAGER-ITEM-FILE-RELATION-COMMANDS MANAGER-ITEM-OPERATION-COMMANDS MANAGER-MAIN-MENU-ITEMS
  MANAGER.BM MANAGER.BM.MASK)
(INITVARS (Manager.ACTIVEFLG NIL)
  (Manager.SORTFILELSTFLG T)
  (Manager.WINDOW-ANCHOR 'ANCHOR-BL)
  (Manager.MENURROWS 20)
  (Manager.DATASPACE NIL)
  (MANAGER-WINDOWS NIL)
  (MANAGER-MAIN-WINDOW NIL)
  (MANAGER-MAIN-ICONW (ICONW MANAGER.BM MANAGER.BM.MASK (create POSITION XCOORD _ 0 YCOORD _ 0)
    T))
  (MANAGER-OPEN-WINDOWS NIL)
  (MANAGER-FILE-MENU NIL)
  (MANAGER-FILELST-MENU NIL)
  (MANAGER-FILE-OPERATIONS-MENU NIL)
  (MANAGER-FILE-FILE-RELATION-MENU NIL)
  (MANAGER-MARKED-SHADE BOLDMENUFONT))
(FILES DATABASEFNS FILEBROWSER (FROM LISPUSERS)
  COMMON-MAKE)
; FILEBROWSER for SEE command
(FNS MANAGER MANAGER.RESET Manager.ADDADV Manager.ADDTOFILES? Manager.ALTERMARKING
  Manager.ANCHORED-SET-POSITION Manager.DO.COMMAND Manager.DO.COMMAND.PROCFN Manager.HIGHLIGHT
  Manager.PROMPT Manager.WINDOW Manager.insurefilehighlights Manager.CHANGED? Manager.CHECKFILE
  Manager.COLLECTCOMS Manager.COMS.WSF Manager.COMSOPEN Manager.COMSUPDATE Manager.HIGHLIGHTED
  Manager.INSUREHIGHLIGHTS Manager.FILECHANGES Manager.FILELSTCHANGED? Manager.FILESUBTYPES
  Manager.GET.ENVIROMENT Manager.GETFILE Manager.INTITLE? Manager.MAIN.WSF Manager.MAINCLOSE
  Manager.MAINMENUITEMS Manager.MAINOPEN Manager.MAINUPDATE Manager.MAKEFILE.ADV Manager.MENUCOLUMNS
  Manager.MENUHASITEM Manager.MENUITEMS Manager.REMOVE.DUPLICATE.ADVISE Manager.RESETSUBITEMS
  Manager.SET-ANCHOR Manager.SORT.COMS Manager.SORTBYCOLUMN)
(ADVISE ADDFILE ADDTOFILES? MAKEFILE MARKASCHANGED UNMARKASCHANGED UPDATEFILES ADDTOCOMS DELFROMCOMS
  \ADDTOFILEBLOCK/ADDNEWCOM LOAD LOADFNS (MARKASCHANGED :IN DEFAULT.EDITDEFA0001))
(MACROS GETDATUM PUTDATUM Manager.TTYCOMMAND)
(PROP MANAGER-DEFINITION-TYPE-COMMANDS ADVISE FNS RECORDS VARS FUNCTIONS)
(ADDVARS (BackgroundMenuCommands (File% Manager (MANAGER)
  "Starts the menu driven file manager"))
  (P (LSUBST 'Manager NIL BackgroundMenuCommands)
    ; remove old manager entry if it exists
    (SETQ BackgroundMenu NIL)
    ; cause the backGround menu to be rebuilt
    (MANAGER.RESET (CL:SYMBOL-VALUE 'Manager.ACTIVEFLG))
    ; Shutdown any old manager windows and restart if we're
    ; already running.
```

```

    (if (STREQUAL MANAGER-ACTIVITY-WINDOW-TITLE (WINDOWPROP NIL 'TITLE))
        then
            ; If we're in the manager activity window, close it, since we
            ; dropped the pointer to it in MANAGER.RESET.
            (CLOSEW NIL)))
    (PROP (MAKEFILE-ENVIRONMENT FILETYPE)
        MANAGER)
    (DECLARE%: DONTVAL@LOAD DOEVAL@COMPILE DONTCOPY COMPILEVAR (ADDVARS (NLAMA)
                                                                    (NLAML)
                                                                    (LAMA)))

;; The Manager : a menu based interface to the file manager.
;; Originally written by: Jay Ferguson of Ford Aerospace & Communications Corp and Robert Noble of Intellicorp.
;; Rewritten by Larry Masinter, winter of 1986.
;; Further modifications & significant enhancements by Andrew J. Cameron III, summer of 1987.
;; Ongoing maintenance and performance tuning by Ron Fischer at Xerox AI Systems.
;;
;; There are two patches in here that should be removed if Xerox Lisp is fixed. The first is the advice (MARKASCHANGED :IN
;; DEFAULT.EDITDEFA0001) that removes a (mostly) redundant call to MARKASCHANGED in the editor interface, which otherwise slows down
;; manager updates. Somehow this call is not redundant when editing FILELST (perhaps there's a special case for FILELST or when items are not in
;; any existing files). The second is the fns Manager.REMOVE.DUPLICATE.ADVISE called by the advice on LOAD and LOADFNS, which removes
;; redundant advice which would otherwise pile up and cause massive slow downs in manager updates!
;;
;; The edit history is now kept in the file MANAGER.HISTORY.
;;
;; Known bugs and feature requests are now kept in the documentation file MANAGER.TEDIT.
;;

(DECLARE%: DOEVAL@COMPILE DONTCOPY

(SPECVARS Manager.ACTIVEFLG MANAGER-CASES MANAGER-ADDTOFILES?)
)

(DECLARE%: DOEVAL@COMPILE DONTCOPY

(GLOBALVARS LAMBDAFONT DEFAULTFONT MENUFONT BOLDMENUFONT MANAGER-WINDOWS LOADDBFLG SAVEDBFLG
MANAGER-ITEM-OPERATION-COMMANDS MANAGER-ITEM-FILE-RELATION-COMMANDS MANAGER-FILE-OPERATIONS-COMMANDS
MANAGER-FILE-FILE-RELATION-COMMANDS MANAGER-MAIN-MENU-ITEMS MANAGER-ACTIVITY-WINDOW-TITLE
MANAGER-MAIN-WINDOW MANAGER-MAIN-ICONW Manager.WINDOW-ANCHOR MANAGER.BM MANAGER.BM.MASK
BackgroundMenuCommands BackgroundMenu)
)

(RPAQQ *UNMANAGED-TYPES* (EXPRESSIONS FILES FIELDS FILEVARS-ARE-NOW-OK))

(RPAQQ MANAGER-ACTIVITY-WINDOW-TITLE "Manager Command Activity")

(RPAQQ MANAGER-CASES NIL)

(RPAQQ MANAGER-ADDTOFILES? NIL)

(RPAQQ MANAGER-FILE-FILE-RELATION-COMMANDS
((" Delete " 'DELETE "Delete this file")
 ("Rename" 'RENAME "Rename this file")
 ("Copy" 'COPY "Copy this item to another file")
 ("Mark" 'CHANGED "Mark this file as being changed")
 ("Unmark" 'UNMARK "Unmark this file as being changed"))))

(RPAQQ MANAGER-FILE-OPERATIONS-COMMANDS
[("See" 'SEE "Show file in a window" (SUBITEMS ("Fast" 'SEE "Show file in a window"
                                                (" Scrollable " 'TEDIT-SEE "Show file in a scrollable
                                                window"))))
 ("(Re)Load" 'LOAD "Load the source of this file" (SUBITEMS ("Load" 'LOAD "Load the source of this file"
                                                            (" SysLoad " 'SYSLOAD "SysLoad the file: smashes
                                                            everything on the way in and is not
                                                            UNDOable"))))
 ("MakeFile" 'MAKEFILE "Dump the source of this file" (SUBITEMS ("MakeFile" 'MAKEFILE "Dump the source of
                                                                this file, by remaking it")
                                                                ("New" 'NEW "Don't copy any definitions from
                                                                old version")
                                                                ("Fast" 'FAST "Dump the source without
                                                                pretty printing")
                                                                (" CommonLisp " 'COMMON-MAKEFILE "Create a
                                                                .LSP file containing plain CommonLisp
                                                                source
                                                                Will load Common-MakeFile if
                                                                necessary"))))
 ("List" 'LIST "List this file on the default printer")
 ("CleanUp" 'CLEANUP "Dump, list and recompile this file" (SUBITEMS ("CleanUp" 'CLEANUP "Dump, list and
                                                                recompile this file, using the
                                                                default cleanup compiler")
                                                                (" Set default: compile-file " CLEANUPC

```

```

                                "Change the default cleanup
                                compiler to compile-file;
                                yeilding .dfasl files")
                                ("Set default: TCOMPL" CLEANUPT
                                "Change the default cleanup
                                compiler to TCOMPL; yeilding
                                .LCOM files
                                This compiler will be going away
                                soon"))
["MasterScope" 'ANALYZE "Analyze the FNS on the selected file with MasterScope"
 (SUBITEMS ("Analyze" 'ANALYZE "Analyze the FNS on the selected file with MasterScope")
  ("Check" 'CHECK "Check the file for problems through MasterScope")
  ("Show Paths" 'SHOWPATHFILE "Show all functions called by functions in this file")
  (" DataBaseFNS " 'DBFILE "Display DATABASE property for this file
    Will load DataBaseFNS if necessary" (SUBITEMS ("Set to Ask" 'DBFILEASK "Ask about
      disposition of MasterScope
      information when loading and
      storing this file")
      ("Set to On" 'DBFILEON "Automatically
        maintain the MasterScope
        information for this file")
      ("Set to Off" 'DBFILEOFF "Do not
        automatically maintain the
        MasterScope information for this
        file")
      (" Load DB " 'LOADDB "Load this file's
        MasterScope information, if it
        exists and make it's upkeep
        automatic")
      ("Dump DB" 'DUMPDB "Dump this file's
        MasterScope information, if it
        exists and make it's upkeep
        automatic"]
  ("Compile" 'COMPILE "Compile this file" (SUBITEMS ("Compile" 'COMPILE "InterLisp compiler")
    (" CL:COMPILE-FILE " 'CL:COMPILE-FILE "CommonLisp
      compiler"))))
  ("Changes" 'CHANGES "Show the changes that have been made to this file." (SUBITEMS ("Brief" 'CHANGES
    "Show the
    changes that
    have been made
    to this
    file.")
    (" Everything "
    'PL "Display everything
    on this file's property
    list")
    ("Edit PL" 'EDIT
    "Edit this file's
    property list"])))

```

(RPAQQ **MANAGER-ITEM-FILE-RELATION-COMMANDS**

```

((" Delete " 'DELETE "Delete this item")
 ("EditAll" 'EDITCALLERS "Edit occurances of this item's name in its file")
 ("Rename" 'RENAME "Rename this item and update its file with new name" (SUBITEMS ("Rename" 'RENAME
  "Rename this
  item locally and
  update its file
  with new name")
  ("CopyDef" 'COPYDEF
  "Make a copy with a
  new name")
  (" Rename All "
  'RENAME-ALL "Rename this
  item in *ALL* loaded
  files"))))

("Move" 'MOVE "Move this item to another file")
("Copy" 'COPY "Copy this item to another file")
("Mark" 'CHANGED "Mark this item as being changed" (SUBITEMS ("Changed" 'CHANGED "Mark item as being
  CHANGED")
  (" Defined " 'DEFINED "Mark item as being
  DEFINED")
  ("Deleted" 'DELETED "Mark item as being
  DELETED"))))

("Unmark" 'UNMARK "Unmark this item as being changed"))

```

(RPAQQ **MANAGER-ITEM-OPERATION-COMMANDS**

```

[("Edit" 'EDIT "Edit this item")
 (" PrettyPrint " 'SHOWDEF "Show how this item would be written to a file" (SUBITEMS ("Show" 'SHOWDEF
  "Show how
  this item
  would be
  written to a
  file")
  ("Value" 'PV
  "Display
  (Pretty-Print)
  this item's

```



```

("LoadFrom"
 'LOADFROMLATER "Notice a file using
LOADFROM" (SUBITEMS
  (" LoadFrom Later "
   'LOADFROMLATER
    "Notice a file with
    side-effects, but don't
    load the function defs
    until needed")
  ("LoadFrom Now"
   'LOADFROMNOW "Notice a
   file with side-effects
   and load all it's
   function defs"))
("Load" 'LOAD "Notice a file by
  actually LOADING it")
("AddFile" 'ADDFILE "Notices a file
  via ADDFILE (buggy)")
("Edit FILELST" 'EDIT "Edit the
  variable which lists the files
  noticed by the file package"))

```

```

)
("Advice" 'SHOWADVICE "Display the list of advised or traced fns and functions.")
("Set Window Anchor" 'ANCHOR-BL "Set the anchor corner for window growth to Bottom Left (default)"
 (SUBITEMS (" Top Left " 'ANCHOR-TL "Set the anchor corner to Top Left")
  (" Top Right " 'ANCHOR-TR "Set the anchor corner to Top Right")
  (" Bottom Left " 'ANCHOR-BL "Set the anchor corner to Bottom Left")
  (" Bottom Right " 'ANCHOR-BR "Set the anchor corner to Bottom Right")))
("Quit" 'QUIT "Shut down all manager windows" (SUBITEMS ("Quit" 'QUIT "Shut down all manager windows")
  (" Reset " 'RESET "Reset the manager, leaving only
  the main window open"])
```

(RPAQQ MANAGER.BM



(RPAQQ MANAGER.BM.MASK



(RPAQQ? Manager.ACTIVEFLG NIL)

(RPAQQ? Manager.SORTFILELSTFLG T)

(RPAQQ? Manager.WINDOW-ANCHOR 'ANCHOR-BL)

(RPAQQ? Manager.MENUROWS 20)

(RPAQQ? Manager.DATASPACE NIL)

(RPAQQ? MANAGER-WINDOWS NIL)

(RPAQQ? MANAGER-MAIN-WINDOW NIL)

```

(RPAQQ? MANAGER-MAIN-ICONW (ICONW MANAGER.BM MANAGER.BM.MASK (create POSITION XCOORD _ 0 YCOORD _ 0)
T))

```

(RPAQQ? MANAGER-OPEN-WINDOWS NIL)

(RPAQQ? MANAGER-FILE-MENU NIL)

(RPAQQ? MANAGER-FILELST-MENU NIL)

(RPAQQ? MANAGER-FILE-OPERATIONS-MENU NIL)

(RPAQQ? MANAGER-FILE-FILE-RELATION-MENU NIL)

(RPAQQ? MANAGER-MARKED-SHADE BOLDMENUFONT)

```

(FILESLoad DATABASEFNS FILEBROWSER (FROM LISPUSERS)
COMMON-MAKE)

```

;; FILEBROWSER for SEE command

(DEFINEQ

(MANAGER

[LAMBDA (POSITION)

; Edited 3-Sep-87 13:58 by raf

;;; Turns manager on if its not already on

(if (OR (NULL Manager.ACTIVEFLG)

```

(NULL MANAGER-MAIN-WINDOW)
(Manager.FILELSTCHANGED?))
then ;; If either the manager was off or FILELST changed, rebuild main menu.
  (if Manager.ACTIVEFLG
    then (Manager.MAINCLOSE))
  (LET ((Manager.ACTIVEFLG NIL))
    (UPDATEFILES))
  (if FILELST
    then (Manager.MAINOPEN POSITION)
    else (PROMPTPRINT "FILELST is empty; there are no files to manage."))
else (TOTOPW MANAGER-MAIN-WINDOW])

```

**(MANAGER.RESET**

[LAMBDA (RESTARTFLG)

; Edited 21-Aug-87 11:41 by raf

```

;;; Remove all cached menu info, close the main window, clear the global data space. If the RESTARTFLG is true, turn everything on again.

```

```

;; Delete all of the menu caches
(for X in FILEPKGCOMSPLST when (LITATOM X) do (REMPROP X 'MANAGER-ITEM-OPERATION-MENU))
(SETQ MANAGER-MAIN-MENU NIL)
(SETQ MANAGER-FILE-OPERATIONS-MENU NIL)
(SETQ MANAGER-ITEM-FILE-RELATION-MENU NIL)
(SETQ MANAGER-ITEM-OPERATION-MENU NIL)
(LET [(REGION (AND RESTARTFLG (WINDOWP MANAGER-MAIN-WINDOW)
  (WINDOWPROP MANAGER-MAIN-WINDOW 'REGION) ; Save away the old region (if there was one.

;; Close the main window and all subwindows.
(Manager.MAINCLOSE T)
;; Clear the data space.
[SETQ Manager.DATASPACE (COPY ' (NIL]
(if RESTARTFLG
  then ;; Now turn it all on again.
    [MANAGER (AND REGION (create POSITION
                                XCOORD _ (fetch (REGION LEFT) of REGION)
                                YCOORD _ (fetch (REGION BOTTOM) of REGION)
  else (SETQ Manager.ACTIVEFLG NIL))

```

**(Manager.ADDADV**

[LAMBDA (!VALUE FILECOMS NAME COMSTYPE)

; Edited 16-Aug-87 22:38 by raf

```

;;; Called when any file's COMS are added to or deleted from. For each open subitem window of that file, if we're under ADDTOFILES? save the
;;; change, otherwise update the window.

```

```

(PROG (FILE SUBITEMS ITEMS)
  (if (OR (NULL !VALUE)
    (LISTP FILECOMS))
    then (RETURN)
    else (if [SETQ FILE (for F in FILELST thereis (EQ FILECOMS (FILECOMS F)
      then (for WINDOW in MANAGER-OPEN-WINDOWS bind STUFF
        when (AND (OPENWP WINDOW)
          (EQ [CDR (SETQ STUFF (GETDATUM (CAR (WINDOWPROP WINDOW 'MENU]
            COMSTYPE)
          (EQ (CAR STUFF)
            FILE))
        do (if MANAGER-ADDTOFILES?
          then (pushnew MANAGER-CASES STUFF)
          else (Manager.COMSOPEN FILE COMSTYPE)))
      (Manager.RESETSUBITEMS FILE COMSTYPE])

```

**(Manager.ADDTOFILES?**

[LAMBDA NIL

(\* Imm "16-Nov-86 23:16")

```

(for CASE in MANAGER-CASES do (Manager.COMSOPEN (CAR CASE)
  (CDR CASE)))
(SETQ MANAGER-CASES NIL])

```

**(Manager.ALTERMARKING**

[LAMBDA (ITEM TYPE MARKING?)

; Edited 3-Sep-87 16:39 by raf

```

;;; Called from MARKSCHANGED or UNMARKASCHANGED.

```

```

(COND
  ((EQ MARKING? 'CLISP) ; ignore
  )
  ((AND (EQ ITEM 'FILELST)
    (EQ TYPE 'VARS)) ; FILELST has been edited.
    (MANAGER))
  ((EQ TYPE 'FILES) ; A whole file has been marked.
    (UPDATEFILES))
  (T ;; For each manager menu window that's open we look to see if it contains the named definition. We can only update a menu if the window

```

;; is expanded (and can't see the menu when its window is shrunk).

```
(for WINDOW in MANAGER-OPEN-WINDOWS bind MENU (UPDATEFILES _ NIL)
  when [AND (OPENWP WINDOW)
            (SETQ MENU (CAR (WINDOWPROP WINDOW 'MENU))
            do (if [AND (Manager.MENUHASITEM ITEM MENU)
                      (EQ TYPE (CDR (GETDATUM MENU))
                      then (SELECTQ MARKING?
                                ((DELETED DEFINED)
                                (SETQ UPDATEFILES T)
                                (Manager.COMSOPEN (CAR (GETDATUM MENU))
                                TYPE NIL))
                                (Manager.HIGHLIGHT ITEM MENU MARKING?)))
                      finally (Manager.MAINUPDATE UPDATEFILES])
```

**(Manager.ANCHORED-SET-POSITION**

; Edited 10-Oct-2023 11:22 by mth

```
[LAMBDA (IW IH)
  (LET (WREGION XPOS YPOS TEMP)
    (SETQ WREGION (WINDOWPROP MANAGER-MAIN-WINDOW 'REGION))
    (SETQ YPOS (fetch (REGION BOTTOM) of WREGION))
    (if (FMEMB Manager.WINDOW-ANCHOR ' (ANCHOR-TL ANCHOR-TR))
      then (SETQ YPOS (- (+ YPOS (fetch (REGION HEIGHT) of WREGION))
                        IH)))
    (SETQ TEMP (+ YPOS IH))
    (if (>= TEMP SCREENHEIGHT)
      then (SETQ YPOS (- SCREENHEIGHT 1)))
    (SETQ XPOS (fetch (REGION LEFT) of WREGION))
    (if (FMEMB Manager.WINDOW-ANCHOR ' (ANCHOR-TR ANCHOR-BR))
      then (SETQ XPOS (- (+ XPOS (fetch (REGION WIDTH) of WREGION))
                        IW)))
    (SETQ TEMP (+ XPOS IW))
    (if (>= TEMP SCREENWIDTH)
      then (SETQ XPOS (- SCREENWIDTH 1)))
    (create POSITION
      XCOORD _ XPOS
      YCOORD _ YPOS])
```

**(Manager.DO.COMMAND**

[LAMBDA (COMMAND ITEM COMSTYPE FILE MENU)

; Edited 21-May-2024 17:56 by mth

; Edited 17-May-2024 09:41 by mth

; Edited 13-Oct-2023 16:28 by mth

```
(if (EQ COMSTYPE 'FILEVARS)
  then (SETQ COMSTYPE 'VARS)
```

; The Manager currently does unnatural things with the  
; FILEVARS type, this is a hack to compensate for it. E.g., editing  
; a FILEVARS = editing the VARS, etc.

```
)
(SELECTQ COMMAND
  (NIL
  )
```

; Do nothing.

```
(EDIT (WITH-READER-ENVIRONMENT (if FILE
                                then (Manager.GET.ENVIRONMENT FILE)
                                else (MAKE-READER-ENVIRONMENT *PACKAGE* *READTABLE* *READ-BASE*)))
```

;; SEdit does not use \*package\*.

```
[COND
  ((EQ COMSTYPE 'FILES)
   (ED ITEM 'PROPERTY-LIST))
  ((NULL COMSTYPE)
   (EDITDEF 'FILELST 'VARS))
  (T (EDITDEF ITEM COMSTYPE NIL NIL '(:DONTWAIT)))
(ADD.PROCESS '[CL:APPLY #'Manager.DO.COMMAND.PROCFN '(,COMMAND ,ITEM ,COMSTYPE ,FILE ,MENU)
              'NAME
              'MANAGER-COMMAND))
NIL])
```

**(Manager.DO.COMMAND.PROCFN**

[LAMBDA (COMMAND ITEM COMSTYPE FILE MENU)

; Edited 20-May-2024 11:15 by mth

```
(WITH-READER-ENVIRONMENT (if FILE
                              then (Manager.GET.ENVIRONMENT FILE)
                              else (MAKE-READER-ENVIRONMENT *PACKAGE* *READTABLE* *READ-BASE*)))
```

```
[LET ((ACTIVITY-WINDOW NIL)
      (ACTIVITY-WINDOW-WAS-SHRUNK NIL))
  (RESETLST
   (RESETSAVE (TTY.PROCESS (THIS.PROCESS)))
   [if [NOT (FMEMB COMMAND ' (BREAK TRACE UNBREAK CHANGED DELETED DEFINED UNMARK SEE LIST HARDCOPY
                               REMOVE NIL]
     then
      (TTYDISPLAYSTREAM (SETQ ACTIVITY-WINDOW (Manager.WINDOW)))
      (SETQ ACTIVITY-WINDOW-WAS-SHRUNK (NOT (OPENWP ACTIVITY-WINDOW))
      (CL:BLOCK NIL
       (CL:ECASE COMMAND
         (READWISE (APPLY* (FUNCTION READWISE)
                           ITEM))
         (UNADVISE (APPLY* (FUNCTION UNADVISE)
```

; steal the TTY, if we really need it (there are also further  
; complementary lists at the bottom of the following BLOCK).

```

ITEM))
(SHOWADVICE
  (printout T .FONT LAMBDAFONT "Advised and traced fns and functions:" .FONT
    DEFAULTFONT T)
  (for ITEM in ADVISEDFNS do (printout T 10 ITEM T)))
(RESET (COND
  ((MOUSECONFIRM "Reset the Manager destroying all the menus? " NIL T)
    (CL:FORMAT T "Expunging and reconstructing the Manager's menus~%%Please
      Stand By.")
    (MANAGER.RESET T)
    (CL:FORMAT T "~&Done.~%%-----")
    (CLOSEW T))))
(QUIT (COND
  ((MOUSECONFIRM "Quit the Manager? " NIL T)
    (Manager.MAINCLOSE T)
    (CLOSEW T))))
(RELOAD
  (CL:FORMAT T "~&Loading ~A definition of ~S from ~A." ITEM COMSTYPE FILE)
  (LOADDEF ITEM COMSTYPE FILE))
(SHOWDEF
  (printout T .FONT LAMBDAFONT COMSTYPE " definition of " ITEM .FONT DEFAULTFONT "
    (source file format):" T)
  (SHOWDEF ITEM COMSTYPE))
(BREAK (APPLY* 'BREAK ITEM))
(TRACE (EVAL (LIST 'TRACE ITEM)))
(UNBREAK (EVAL (LIST 'UNBREAK ITEM)))
(DISASSEMBLE
  (printout T .FONT LAMBDAFONT "Compiled code for " ITEM ":" .FONT DEFAULTFONT T)
  (INSPECTCODE ITEM))
(PV (printout T .FONT LAMBDAFONT "Value of " ITEM ":" .FONT DEFAULTFONT T
  (if (BOUNDP ITEM)
    then (EVAL ITEM)
    else "Not bound!"))))
(PF
  (printout T .FONT LAMBDAFONT "Function definition of " ITEM ":" .FONT DEFAULTFONT T)
  (APPLY* #'PF ITEM))
(PL
  (printout T .FONT LAMBDAFONT "Property list for " ITEM ":" .FONT DEFAULTFONT T)
  (PRINTPROPS (if (EQ COMSTYPE 'PROPS)
    then (CAR ITEM)
    else ITEM)))
(CLDESCRIBE
  (printout T .FONT LAMBDAFONT "Description of " ITEM ":" .FONT DEFAULTFONT T)
  (CL:DESCRIBE ITEM))
(CLDOC (printout T .FONT LAMBDAFONT "Documentation for " ITEM ":" .FONT DEFAULTFONT T
  (CL:DOCUMENTATION ITEM)))
(FIELDS (printout T .FONT LAMBDAFONT "Fields of " ITEM ":" .FONT DEFAULTFONT T
  (REVERSE (RECORDFIELDNAMES ITEM))))
(ARGS (printout T .FONT LAMBDAFONT "Arguments of " ITEM ":" .FONT DEFAULTFONT T 10
  (SMARTARGLIST ITEM)
  T))
(EDITCALLERS (EDITCALLERS ITEM FILE))
(COPYDEF (LET [(FILENAME (Manager.PROMPT (CONCAT "Copy " ITEM " as name: ")
  (if FILENAME
    then (COPYDEF ITEM FILENAME COMSTYPE))
  (RENAME (LET [(FILENAME (Manager.PROMPT (CONCAT "Rename " ITEM " to: ")
  (if FILENAME
    then (RENAME ITEM FILENAME COMSTYPE FILE)))
  (RENAME-ALL (LET [(FILENAME (Manager.PROMPT (CONCAT "Rename (in ALL loaded files) "
    ITEM " to: ")
    (if FILENAME
      then (RENAME ITEM FILENAME COMSTYPE FILEST)))
  (DELETE (if (MOUSECONFIRM (CONCAT "DELETE the " COMSTYPE " " ITEM " from " FILE "?")
    then (DELFROMFILES ITEM COMSTYPE FILE)))
  (LOAD (LET [(FILENAME (Manager.PROMPT "Filename: "))
  (if FILENAME
    then (LOAD FILENAME)))
  (LOADFNSLATER [LET [(FILENAME (Manager.PROMPT "Filename: "))
  (if FILENAME
    then (LOADFNS NIL FILENAME 'ALLPROP 'VARS])
  (LOADFNSNOW [LET [(FILENAME (Manager.PROMPT "Filename: "))
  (if FILENAME
    then (LOADFNS T FILENAME 'ALLPROP 'VARS])
  (LOADFROMLATER (LET [(FILENAME (Manager.PROMPT "Filename: "))
  (if FILENAME
    then (LOADFROM FILENAME)))
  (LOADFROMNOW (LET [(FILENAME (Manager.PROMPT "Filename: "))
  (if FILENAME
    then (LOADFROM FILENAME T)))
  (ADDFILE (LET [(FILENAME (Manager.PROMPT "Filename: "))
  (if FILENAME
    then (ADDFILE FILENAME)))
  (SYSLOAD [COND
    ((MOUSECONFIRM (CONCAT "Do you really want to SYSLOAD " FILE "?" NIL T))
      NIL
      (LOAD FILE 'SYSLOAD])
  (MOVE (LET [(ANSWER (Manager.GETFILE (CONCAT "File to move " COMSTYPE " " ITEM " to")

```



```

        (AND ANSWER (MOVETOFILE ANSWER ITEM COMSTYPE FILE))))
(COPY (LET [(ANSWER (Manager.GETFILE (CONCAT "File to copy " COMSTYPE " " ITEM " to")
        (AND ANSWER (ADDTOTFILE ITEM COMSTYPE ANSWER))))
((CHANGED DELETED DEFINED) (if COMSTYPE
        then (MARKASCHANGED ITEM COMSTYPE COMMAND)
        else (MARKASCHANGED (FILECOMS ITEM)
        'VARS COMMAND)
        (UPDATEFILES)
        ; This is needed because the main menu is a special case. Its
        ; not in the open windows list, nor does it carry "type" information
        ; (like that it contains filevars).
        ))
(UNMARK (if (EQ COMSTYPE 'FILES)
        then ; whole file
        (COND
        ((MOUSECONFIRM (CONCAT "Unmark entire contents of " FILE "?" NIL T)
        )
        (/RPLACD (GETPROP FILE 'FILE)
        NIL)
        (Manager.insurefilehighlights FILE)
        (Manager.HIGHLIGHT FILE MENU)))
        else ; single item
        (UNMARKASCHANGED ITEM COMSTYPE)))
(SEE (LET ((FULLNAME (OR (CDAR (GETPROP FILE 'FILEDATES))
        FILE)))
        ;; I'm assuming that the CAR of the FILEDATES list is the most recent...
        (FB.FASTSEE.ONEFILE NIL FULLNAME (LET [(W (CREATEW NIL (CONCAT "Seeing "
        FULLNAME "...")
        (DSPSCROLL 'ON W)
        (WINDOWPROP W 'PAGEFULLFN
        'FB.SEEFULLFN)
        (TTYDISPLAYSTREAM W)
        W))))
(TEDIT-SEE (TEDIT-SEE (OR (CDAR (GETPROP FILE 'FILEDATES))
        FILE)))
(LOAD
        (printout T .FONT LAMBDAFONT "Loading file " FILE "." .FONT DEFAULTFONT T)
        (LOAD FILE))
((MAKEFILE NEW FAST) (if FILE
        then (printout T .FONT LAMBDAFONT "Writing file " FILE "."
        .FONT DEFAULTFONT T)
        (PRINT (MAKEFILE FILE (if (EQ COMMAND 'MAKEFILE)
        then NIL
        else COMMAND))
        T)
        else (printout T .FONT LAMBDAFONT "Writing files ")
        (PRINT (MAKEFILES (if (EQ COMMAND 'MAKEFILE)
        then NIL
        else (LIST COMMAND)
        (printout T .FONT DEFAULTFONT T)))
(COMMON-MAKEFILE (if FILE
        then (printout T .FONT LAMBDAFONT "Writing CommonLisp source into
" FILE ".LSP" .FONT DEFAULTFONT T)
        (PRINT (COMMON-MAKEFILE FILE)
        T)
        else (CL:FORMAT T "~&CommonLispify must be selected separately for
        each file")))
((LIST HARDCOPY) (LISTFILES1 FILE))
((ANCHOR-TL ANCHOR-TR ANCHOR-BL ANCHOR-BR) (Manager.SET-ANCHOR COMMAND))
(CLEANUP
        (printout T .FONT LAMBDAFONT "Cleanup..." .FONT DEFAULTFONT T)
        ; These are different, presumably because CLEANUP is an
        ; NLAMBDA.
        (if FILE
        then (APPLY* (FUNCTION CLEANUP)
        FILE)
        else (CLEANUP)))
(CLEANUPT (printout T .FONT LAMBDAFONT "Changing default cleanup compiler:" .FONT
        DEFAULTFONT T "Old value " *DEFAULT-CLEANUP-COMPILER* T "New value: "
        (SETQ *DEFAULT-CLEANUP-COMPILER* 'TCOMPL)
        T))
(CLEANUPC (printout T .FONT LAMBDAFONT "Changing default cleanup compiler:" .FONT
        DEFAULTFONT T "Old value " *DEFAULT-CLEANUP-COMPILER* T "New value: "
        (SETQ *DEFAULT-CLEANUP-COMPILER* 'COMPILE-FILE)
        T))
;; Masterscope stuff
(ANALYZE
        (printout T .FONT LAMBDAFONT "Analyzing the file " FILE " with MasterScope..." .FONT
        DEFAULTFONT T)
        (MASTERSCOPE `(ANALYZE FNS ON %, FILE)))
(CHECK
        (printout T .FONT LAMBDAFONT "Checking the file " FILE " with MasterScope..." .FONT
        DEFAULTFONT T)
        (MASTERSCOPE `(CHECK %, FILE)))
(DESCRIBE (SELECTQ COMSTYPE

```

```

(VARS [CL:FORMAT T "~&~a is used by:~%~a" ITEM
      (MASTERSCOPE `(WHO USES ',ITEM)])
(PROGN NIL (printout T .FONT LAMBDAFONT "MasterScope analysis of " ITEM
      ":" .FONT DEFAULTFONT T)
      (MSDESCRIBE ITEM))))
(SHOWPATHTO
  (printout T .FONT LAMBDAFONT "Showing who calls " ITEM " with MasterScope..." .FONT
    DEFAULTFONT T)
  (MASTERSCOPE `(SHOW PATHS TO %, ITEM)))
(SHOWPATHFROM
  (printout T .FONT LAMBDAFONT "Showing who is called by " ITEM " with
    MasterScope..." .FONT DEFAULTFONT T)
  (MASTERSCOPE `(SHOW PATHS FROM %, ITEM)))
(SHOWPATHFILE
  (printout T .FONT LAMBDAFONT "Showing who is called by functions in the file " ITEM
    " with MasterScope..." .FONT DEFAULTFONT T)
  (MASTERSCOPE `(SHOW PATHS FROM ON %, FILE)))
;; DATABASEFNS stuff
(DB (CL:FORMAT T "~&Global DataBaseFNS Flags:~%~a"SAVEDBFLG = ~a, LOADDDBFLG = ~a"
  SAVEDBFLG LOADDDBFLG))
(DBFILE
  (CL:FORMAT T "~&The DATABASE prop for ~a is: ~a" FILE (GETPROP FILE 'DATABASE))
  (CL:FORMAT T "~&Global DataBaseFNS Flags:~%~a"SAVEDBFLG = ~a, LOADDDBFLG = ~a"
    SAVEDBFLG LOADDDBFLG))
(DBON
  (SETQ LOADDDBFLG 'ON)
  (SETQ SAVEDBFLG 'ON))
(DBOFF
  (SETQ LOADDDBFLG 'NO)
  (SETQ SAVEDBFLG 'NO))
(DBASK
  (SETQ LOADDDBFLG 'ASK)
  (SETQ SAVEDBFLG 'ASK))
(DBLOADON (SETQ LOADDDBFLG 'YES))
(DBSAVEON (SETQ SAVEDBFLG 'YES))
(DBLOADOFF (SETQ LOADDDBFLG 'NO))
(DBSAVEOFF (SETQ SAVEDBFLG 'NO))
(DBLOADASK (SETQ LOADDDBFLG 'ASK))
(DBSAVEASK (SETQ SAVEDBFLG 'ASK))
(DBFILEON (PUTPROP FILE 'DATABASE 'YES))
(DBFILEOFF (PUTPROP FILE 'DATABASE 'NO))
(DBFILEASK (PUTPROP FILE 'DATABASE 'ASK))
(DUMPDB
  (printout T .FONT LAMBDAFONT "Dumping the Masterscope Database for file " FILE .FONT
    DEFAULTFONT T)
  (DUMPDB FILE))
(LOADDB
  (printout T .FONT LAMBDAFONT "Loading the Masterscope Database for file " FILE .FONT
    DEFAULTFONT T)
  (LOADDB FILE))
(COMPILE
  (printout T .FONT LAMBDAFONT "Compiling..." .FONT DEFAULTFONT T)
  (if (EQ COMSTYPE 'FILES)
    then (APPLY* (FUNCTION COMPILEFILES)
      FILE)
    (Manager.REMOVE.DUPLICATE.ADVISE FILE)
    else (PRINT (CL:COMPILE ITEM
      T))))
(CL:COMPILE-FILE
  (printout T .FONT LAMBDAFONT "Compiling using compile-file..." .FONT DEFAULTFONT T)
  (CL:COMPILE-FILE FILE)
  (Manager.REMOVE.DUPLICATE.ADVISE FILE))
(REMOVE (DELDEF FILE 'FILE))
(CHANGES
  (Manager.CHANGED? FILE)) ; FILE is NIL from main menu
(FILES?
  (printout T .FONT LAMBDAFONT "Files and their changes:" .FONT DEFAULTFONT T)
  (FILES?)))
;; Relase the window now, but get ready to shrink it back down unless another manager command comes along and need
;; the window.
(if [NOT (FMEMB COMMAND
  '(BREAK TRACE UNBREAK CHANGED DELETED DEFINED UNMARK SEE LIST HARDCOPY
    REMOVE QUIT RESET RENAME COPY NIL]
  then (CL:FORMAT T "~&-----"))))
;; Shink the dialog window after ten seconds so long as its not in use by another manager command.
(if ACTIVITY-WINDOW-WAS-SHRUNK
  then (if (FMEMB COMMAND
    '(SHOWDEF SHOWADVISE PV PF PL CLDESCRIBE CLDOC FIELDS ARGS DB DBFILE MAKEFILE
      NEW FAST COMMON-MAKEFILE CLEANUPT CLEANUPC CLEANUP ANALYZE CHECK
      DESCRIBE CHANGES FILES? COMPILE CL:COMPILE NIL))
    then (DISMISS 10000)
    else (DISMISS NIL))
  (if (EQ ACTIVITY-WINDOW (CAR MANAGER-WINDOWS))
    then (SHRINKW T]))))

```

**(Manager.HIGHLIGHT**

```
[LAMBDA (ITEM MENU ON)
  (SHADEITEM (SASSOC ITEM (fetch ITEMS of MENU))
    MENU
    (if ON
      then MANAGER-MARKED-SHADE
      else 0))]
```

; Edited 31-Jul-87 17:33 by raf

**(Manager.PROMPT**

```
[LAMBDA (PROMPT)
  (LET (W (Manager.WINDOW))
    (RESETFORM (TTY.PROCESS (THIS.PROCESS))
      (PROG1 (MKATOM (PROMPTFORWORD PROMPT NIL NIL W))
        (printout W T)))]
```

; Edited 17-Aug-87 14:31 by raf

**(Manager.WINDOW**

```
[LAMBDA NIL
```

; Edited 21-Aug-87 12:04 by raf

```
;;; Make a window for manager activity, and set TTYDISPLAYSTREAM into it.
```

```
(LET [(W (OR (pop MANAGER-WINDOWS)
  (CREATEW NIL MANAGER-ACTIVITY-WINDOW-TITLE]
  (RESETSAVE NIL (LIST [FUNCTION (LAMBDA (W)
    (AND (OPENWP W)
      (TERPRI W))
    (push MANAGER-WINDOWS W]
    W))
  (TTYDISPLAYSTREAM W)
  W)])
```

**(Manager.insurefilehighlights**

```
[LAMBDA (FILE)
```

; Edited 26-Jun-87 16:30 by andyiii

; insures open menus of a file are correctly highlighted

```
(SETQ FILE (ROOTFILENAME FILE))
(for WINDOW in MANAGER-OPEN-WINDOWS bind MENU
  when (AND (OPENWP WINDOW)
    (EQ [CAR (GETDATUM (SETQ MENU (CAR (WINDOWPROP WINDOW 'MENU]
      FILE))
    do [if (EQUAL (Manager.COLLECTCOMS FILE (CDR (GETDATUM MENU)))
      (fetch (MENU ITEMS) of MENU))
      then (Manager.COMSUPDATE WINDOW) ; no change in contents
      else (Manager.COMSOPEN FILE (CDR (GETDATUM MENU]) ; contents changed
    ])
```

**(Manager.CHANGED?**

```
[LAMBDA (FILES)
```

; Edited 26-Jun-87 03:42 by andyiii

```
(bind CHANGES for FILE inside (OR FILES FILELST) first (TERPRI T) when [SETQ CHANGES (CDR (GETPROP FILE
  'FILE]
  do (printout T .FONT LAMBDALFONT "Changes to " FILE .FONT DEFAULTFONT T)
    (for CHANGE in CHANGES do (printout T (CAR CHANGE)
      ":" 10 .PARA 10 0 (CDR CHANGE)
      T))
```

**(Manager.CHECKFILE**

```
[LAMBDA (FILE)
```

; Edited 17-Aug-87 14:26 by raf

```
;;; If called from ADDTOFILES? (special flag indicates this) and the file being checked is on the main menu, checks all of a particular FILE's submenus,
;;; otherwise rebuilds the main (FILELST) menu. Called from advice on ADDFILE, ADDTOFILES? and LOAD.
```

```
(if (AND (NULL MANAGER-ADDTOFILES?)
  (Manager.MENUHASITEM FILE MANAGER-FILE-MENU))
  then (SETQ FILE (ROOTFILENAME FILE))
  [for WINDOW in MANAGER-OPEN-WINDOWS bind MENU
    when [AND (OPENWP WINDOW)
      (EQ [CAR (GETDATUM (SETQ MENU (CAR (WINDOWPROP WINDOW 'MENU]
        FILE)
      (NOT (EQUAL (Manager.COLLECTCOMS FILE (CDR (GETDATUM MENU)))
        (fetch (MENU ITEMS) of MENU])
      do (Manager.COMSOPEN FILE (CDR (GETDATUM MENU])
    else (MANAGER)
      (Manager.RESETSUBITEMS FILE)])
```

**(Manager.COLLECTCOMS**

```
[LAMBDA (FILE TYPE)
```

; Edited 16-Aug-87 22:13 by raf

```
;;; Collect all the names of a particular type in a file, returning them in correct menu item format.
```

```
(PROG ((COMSLST (FILECOMSLST FILE TYPE)))
```

```

(RETURN (Manager.SORTBYCOLUMN (COND
  ((NULL COMSLST)
   (RETURN))
  ((EQ TYPE 'VARS)
   (for VAR in COMSLST bind (FILEVARS _ (FILECOMSLST FILE 'FILEVARS))
    when (NOT (FMEMB VAR FILEVARS)) collect
      ;; List of item to get around menu feature that list's first
      ;; item is used for display.
      (LIST VAR)))
  (T ;; List of item to get around menu feature that list's first item is used for display.
   (MAPCAR (INTERSECTION COMSLST COMSLST)
    (FUNCTION LIST]))

```

**(Manager.COMS.WSF**

```

[LAMBDA (ITEM MENU KEY)
  (SETQ ITEM (CAR ITEM))

  (PROG (FILE COMSTYPE COMSLST FILECOMS COMMAND)
    (DECLARE (SPECVARS ITEM COMSTYPE))
    (if (NULL ITEM)
      then (RETURN))
    (if (.COPYKEYDOWNP.)
      then (RETURN (BKSYSEBUF.GENERAL ITEM)))
    (SETQ COMSLST (GETDATUM MENU))
    (SETQ FILE (CAR COMSLST))
    (SETQ COMSTYPE (CDR COMSLST))
    [SETQ COMMAND
      (MENU (SELECTQ KEY
        (LEFT [OR (GETPROP COMSTYPE 'MANAGER-ITEM-OPERATION-MENU)
          [AND (GETPROP COMSTYPE 'MANAGER-DEFINITION-TYPE-COMMANDS)
            (PUTPROP COMSTYPE 'MANAGER-ITEM-OPERATION-MENU
              (create MENU
                ITEMS _ (APPEND MANAGER-ITEM-OPERATION-COMMANDS
                  (GETPROP COMSTYPE
                    'MANAGER-DEFINITION-TYPE-COMMANDS))
                CENTERFLG _ T
                TITLE _ (CONCAT COMSTYPE " operations")
                CHANGEOFFSETFLG _ 'Y]
              MANAGER-ITEM-OPERATION-MENU
            (SETQ MANAGER-ITEM-OPERATION-MENU
              (create MENU
                ITEMS _ MANAGER-ITEM-OPERATION-COMMANDS
                CENTERFLG _ T
                TITLE _ (CONCAT COMSTYPE " operations")
                CHANGEOFFSETFLG _ 'Y])
            (MIDDLE (OR MANAGER-ITEM-FILE-RELATION-MENU
              (create MENU
                ITEMS _ MANAGER-ITEM-FILE-RELATION-COMMANDS
                CENTERFLG _ T
                TITLE _ "Other operations"
                CHANGEOFFSETFLG _ 'Y)))
            (SHOULDNT])
          (if COMMAND
            then (Manager.DO.COMMAND COMMAND ITEM COMSTYPE FILE))

```

; Edited 25-Jun-87 02:00 by andyiii

; Menu items handed in are list of item to get around menu  
; feature that list has first item used to display!**(Manager.COMSOPEN**

```

[LAMBDA (FILE TYPE FLASHFLG)

```

; Edited 16-Aug-87 22:30 by raf

```

;;; Open a subitems window. If it already exists, and all subitems of this type have been deleted, close the window, otherwise open it, flash and check
;;; the highlights. If the file's subitems for this type have changed, then rebuild the menu.

```

```

(PROG ((COMSLST (Manager.COLLECTCOMS FILE TYPE))
  (COMSTYPE (FILECOMS FILE TYPE))
  (MENU WINDOW POSITION)
  (COND
    [COMSLST (COND
      ([AND (SETQ MENU (GETDATUM COMSTYPE))
        (EQUAL (fetch (MENU ITEMS) of MENU)
          COMSLST)
        (SETQ WINDOW (OR (WFROMMENU MENU)
          (for W in MANAGER-OPEN-WINDOWS
            thereis (EQ (WINDOWPROP W 'COMSTYPE)
              COMSTYPE)]
        (COND
          (FLASHFLG (FLASHWINDOW WINDOW 2))
          (T (TOTOPW WINDOW)))
        (Manager.INSUREHIGHLIGHTS MENU (Manager.FILECHANGES FILE TYPE)))
      (T ;; make sure all the title is visible. This is hard since the menu does not exist yet.
       (SETQ MENU (create MENU
         ITEMS _ COMSLST
         MENU COLUMNS _ (Manager.MENUCOLUMNS COMSLST)
         WHENSELECTEDFN _ (FUNCTION Manager.COMS.WSF)

```

```

                                MENUOUTLINESIZE _ 0))
(COND
  ((SETQ WINDOW (WFROMMENU (GETDATUM COMSTYPE)))
   (SETQ POSITION (with REGION (WINDOWPROP WINDOW 'REGION)
                             (create POSITION
                                      XCOORD _ LEFT
                                      YCOORD _ BOTTOM)))
   (CLOSEW WINDOW)))
[ADDMENU MENU
  (SETQ WINDOW
    (CREATEW [with MENU MENU
              (LET ((IW (WIDTHIFWINDOW IMAGEWIDTH))
                    (IH (HEIGHTIFWINDOW IMAGEHEIGHT T)))
                (with POSITION (OR POSITION (GETBOXPOSITION IW IH))
                  (create REGION
                          LEFT _ XCOORD
                          WIDTH _ IW
                          BOTTOM _ YCOORD
                          HEIGHT _ IH)
                  (CONCAT TYPE " on " FILE]
                [WINDOWPROP WINDOW 'CLOSEFN (FUNCTION (LAMBDA (WINDOW)
                                                         [PUTDATUM (CAR (WINDOWPROP WINDOW
                                                         'MENU]
                                                         (PUTDATUM (WINDOWPROP WINDOW 'COMSTYPE))
                                                         (SETQ MANAGER-OPEN-WINDOWS (DREMOVE WINDOW
                                                         MANAGER-OPEN-WINDOWS
                                                         ]
                                                         (for ITEM in (Manager.FILECHANGES FILE TYPE) do (Manager.HIGHLIGHT ITEM MENU T))
                                                         (PUTDATUM COMSTYPE MENU)
                                                         (PUTDATUM MENU (CONS FILE TYPE))
                                                         (push MANAGER-OPEN-WINDOWS WINDOW)
                                                         (WINDOWPROP WINDOW 'COMSTYPE COMSTYPE]
                ((SETQ WINDOW (WFROMMENU (GETDATUM COMSTYPE)))
                 (CLOSEW WINDOW]))

```

**(Manager.COMSUPDATE**

[LAMBDA (WINDOW FLASHFLG)

; Edited 16-Aug-87 22:30 by raf

;;; Open a window (if closed) and update the coms listed on the menu therein. If items are the same, their highlighting is checked, otherwise the menu is rebuilt.

```

(PROG ([MENU (CAR (WINDOWPROP WINDOW 'MENU]
      CHANGELST FILE TYPE)
      (SETQ FILE (CAR (GETDATUM MENU)))
      (SETQ TYPE (CDR (GETDATUM MENU)))
      (SETQ CHANGELST (Manager.FILECHANGES FILE TYPE))
[COND
  [(NULL (OPENWP WINDOW))
   (if (OPENWP (WINDOWPROP WINDOW 'ICONWINDOW))
       then (EXPANDW (WINDOWPROP WINDOW 'ICONWINDOW)
   (T (if FLASHFLG
         then (FLASHWINDOW WINDOW 2)
         else (TOTOPW WINDOW]
(COND
  ((EQUAL (fetch (MENU ITEMS) of MENU)
          (Manager.COLLECTCOMS FILE TYPE))
   (Manager.INSUREHIGHLIGHTS MENU CHANGELST))
  (T (Manager.COMSOPEN FILE TYPE))

```

**(Manager.HIGHLIGHTED**

[LAMBDA (MENU)

; Edited 9-Jul-87 13:57 by raf

```

  (for X in (fetch (MENU SHADEDITEMS) of MENU) collect (CAR (CAR (NTH (fetch (MENU ITEMS) of MENU)
                                                                    (CAR X]))

```

**(Manager.INSUREHIGHLIGHTS**

[LAMBDA (MENU SHOULD-BE-HIGHLIGHTED)

; Edited 26-Jun-87 18:10 by andyiii

```

  (LET ((HIGH (Manager.HIGHLIGHTED MENU)))
    (if (WFROMMENU MENU)
        then (REDISPLAYW (WFROMMENU MENU)))
    (for ITEM in HIGH when (NOT (FMEMB ITEM SHOULD-BE-HIGHLIGHTED)) do (Manager.HIGHLIGHT ITEM MENU NIL))
    (for ITEM in SHOULD-BE-HIGHLIGHTED when (NOT (FMEMB ITEM HIGH)) do (Manager.HIGHLIGHT ITEM MENU T))

```

**(Manager.FILECHANGES**

[LAMBDA (FILE COMSTYPE)

; Edited 26-Jun-87 04:35 by andyiii

```

  (CDR (FASSOC (if (EQ COMSTYPE 'FILEVARS)
                  then 'VARS
                  else COMSTYPE)
              (CDR (GETPROP FILE 'FILE))

```

**(Manager.FILELSTCHANGED?**

[LAMBDA NIL

; Edited 17-Aug-87 14:16 by raf

```

  (NOT (EQUAL (if Manager.SORTFILELSTFLG

```

```

      then (SORT (COPY FILELST))
    else FILELST)
(Manager.MENUITEMS MANAGER-FILE-MENU])

```

**(Manager.FILESUBTYPES**

[LAMBDA (FILE)

; Edited 16-Aug-87 22:05 by raf

;; Gather the names of all subtypes in a file's coms.

```

(for TYPE in FILEPKGTYPES bind COMSLST
  when (AND (NOT (FMEMB TYPE *UNMANAGED-TYPES*))
    (SETQ COMSLST (FILECOMSLST FILE TYPE))
    (if (EQ TYPE 'VARS)
      then (for VAR in COMSLST bind (FILEVARS _ (FILECOMSLST FILE 'FILEVARS))
        thereis (NOT (FMEMB VAR FILEVARS)))
      else T))
  collect TYPE])

```

**(Manager.GET.ENVIRONMENT**

[LAMBDA (FILE)

; Edited 26-Jun-87 18:53 by andyiii

```

;; Get's a file's environment, either from the cache in the makefile-environment property (which we initialize here if it hasn't been already) or as per
;; the defaulting described in the Lyric release notes:

```

;; cache property exists? use it,

;; new file? use \*DEFAULT-MAKEFILE-ENVIRONMENT\*,

;; old file which has environment in it? use environment from old file,

;; otherwise use an interlisp style environment.

```

(LET [(ENVIRONMENT (OR (GETPROP FILE 'MAKEFILE-ENVIRONMENT)
  (PUTPROP FILE 'MAKEFILE-ENVIRONMENT
    (LET ((DATE (FILEDATE FILE)))
      (if (NULL DATE)
        then *DEFAULT-MAKEFILE-ENVIRONMENT*
        else (LET [(FORM (CL:WITH-OPEN-FILE (STREAM (OR (FINDFILE FILE)
          (CL:ERROR "Can't find
            file ~s to get
            its environment"
            FILE)))
            (LET ((*READTABLE* (FIND-READTABLE
              "OLD-INTERLISP-FILE"))
              (*PACKAGE* (CL:FIND-PACKAGE "INTERLISP"))
              (CL:READ STREAM)
              (if (EQ 'DEFINE-FILE-INFO (CAR FORM))
                then (CDR FORM)
                else '(:READTABLE "INTERLISP" :PACKAGE "INTERLISP" :BASE 10)
              (LET [(VALUE (EVAL (CL:GETF ENVIRONMENT KEY)
                (if (TYPEP VALUE TYPE)
                  then VALUE
                  else (CL:FUNCALL COERCE VALUE)
                '(:PACKAGE :READTABLE :BASE)
                '(:PACKAGE CL:READTABLE INTEGER)
                '(:CL:FIND-PACKAGE FIND-READTABLE CL:IDENTITY]))
            (APPLY #'MAKE-READER-ENVIRONMENT (CL:MAPCAR #'[LAMBDA (KEY TYPE COERCE)
              (LET [(VALUE (EVAL (CL:GETF ENVIRONMENT KEY)
                (if (TYPEP VALUE TYPE)
                  then VALUE
                  else (CL:FUNCALL COERCE VALUE)
                '(:PACKAGE :READTABLE :BASE)
                '(:PACKAGE CL:READTABLE INTEGER)
                '(:CL:FIND-PACKAGE FIND-READTABLE CL:IDENTITY]))

```

**(Manager.GETFILE**

[LAMBDA (PROMPT PASSED-IN-FILE-LIST)

; Edited 17-Aug-87 14:32 by raf

```

(LET ((FILE-LIST (OR PASSED-IN-FILE-LIST FILELST)))
  [COND

```

```

    ((OR (NULL (CAR MANAGER-FILELST-MENU))
      (Manager.FILESTCHANGED?))

```

; what is this doing ???

```

      (SETQ MANAGER-FILELST-MENU (create MENU

```

```

        TITLE _ PROMPT
        ITEMS _ (CONS '*newfile* FILE-LIST)
        WHENSELECTEDFN _ (FUNCTION (LAMBDA (ITEM MENU KEY)
          (PROG (ANSWER FILECOMS)
            (COND

```

```

              ((EQ ITEM '*newfile*)

```

```

                (SETQ ANSWER (

```

```

                  Manager.PROMPT

```

```

                    "New file

```

```

                    Name: ")

```

```

                (ADDFILE ANSWER)

```

```

                (RETURN ANSWER)

```

```

              (T (RETURN ITEM)

```

```

            (MENU MANAGER-FILELST-MENU]))

```

**(Manager.INTITLE?**

[LAMBDA (WINDOW)

(\* edited%: "31-Dec-00 16:40")

```

  (PROG (INTERIOR.HEIGHT REGION MENU)

```

```

    [SETQ MENU (CAR (WINDOWPROP WINDOW 'MENU)

```

```

    (SETQ INTERIOR.HEIGHT (FONTPROP (OR (fetch MENUTITLEFONT of MENU)

```

```

      (fetch MENUFONT of MENU))

```

```

      'HEIGHT))
  (with REGION (WINDOWPROP WINDOW 'REGION)
    (SETQ REGION (CREATEREGION LEFT (IDIFFERENCE TOP INTERIOR.HEIGHT)
      WIDTH INTERIOR.HEIGHT)))
  (RETURN (INSIDEP REGION LASTMOUSEX LASTMOUSEY))

```

**(Manager.MAIN.WSF**

```

[LAMBDA (ITEM MENU KEY)
  (DECLARE (SPECVARS ITEM)) ; Edited 31-Jul-87 18:25 by raf
  (PROG NIL
    [if (.COPYKEYDOWNP.)
      then (RETURN (COPYINSERT (CAR ITEM)
        (LET ((SLIDEOFFITEM (EQLLENGTH ITEM 3))) ; A slideoff subitem was selected.
          (SETQ ITEM (CADR ITEM))
          (SELECTQ KEY
            (MIDDLE (COND
              ((NOT SLIDEOFFITEM)
                (Manager.DO.COMMAND [MENU (OR MANAGER-FILE-FILE-RELATION-MENU
                  (create MENU
                    ITEMS _
                    MANAGER-FILE-FILE-RELATION-COMMANDS
                    CENTERFLG _ T
                    TITLE _ "Other operations"
                    CHANGEOFFSETFLG _ 'Y]
                    (CAR ITEM)
                    'FILES
                    (CAR ITEM)
                    MENU))
                (T (FLASHWINDOW (WFROMMENU MENU)
                  1))))
              (LEFT (COND
                (SLIDEOFFITEM (Manager.COMSOPEN (CAR ITEM)
                  (CDR ITEM)
                  T))
                (T
                  (Manager.DO.COMMAND [MENU (OR MANAGER-FILE-OPERATIONS-MENU
                    (SETQ MANAGER-FILE-OPERATIONS-MENU
                      (create MENU
                        ITEMS _ MANAGER-FILE-OPERATIONS-COMMANDS
                        CENTERFLG _ T
                        TITLE _ "File operations"]
                        (CAR ITEM)
                        'FILES
                        (CAR ITEM)
                        MENU))))
                    (FLASHWINDOW (WFROMMENU MENU)
                      1))
                  ]))

```

**(Manager.MAINCLOSE**

```

[LAMBDA (SHUTDOWNFLG) ; Edited 20-Aug-87 16:18 by raf
  (PROG (MENU ICON.WINDOW)
    (for WINDOW in (APPEND MANAGER-OPEN-WINDOWS)
      when (OR SHUTDOWNFLG (NOT (FMEMB [CAR (GETDATUM (CAR (WINDOWPROP WINDOW 'MENU)
        FILELST))))
        do (CLOSEW WINDOW))
    (if (SETQ ICON.WINDOW (WINDOWPROP MANAGER-MAIN-WINDOW 'ICON))
      then (CLOSEW ICON.WINDOW))
    (if SHUTDOWNFLG
      then (if (SETQ ICON.WINDOW (WINDOWPROP MANAGER-MAIN-WINDOW 'ICON))
        then (CLOSEW ICON.WINDOW))
        (AND MANAGER-MAIN-WINDOW (CLOSEW MANAGER-MAIN-WINDOW))
        (SETQ MANAGER-MAIN-WINDOW NIL))
    (for WINDOW in MANAGER-WINDOWS do (if SHUTDOWNFLG
      then (EXPANDW WINDOW))
      (CLOSEW WINDOW))
    (if SHUTDOWNFLG
      then (SETQ MANAGER-WINDOWS NIL]))

```

**(Manager.MAINMENUITEMS**

```

[LAMBDA NIL ; Edited 17-Aug-87 14:14 by raf

```

;;; Returns the menu 'items' for the main manager file menu. This is, for each file, the menu element and the subitems which contain all of the 'types' . If there is already a file menu, we reuse the subitems rather than recomputing them.

```

(for FILE in (if Manager.SORTFILELSTFLG
  then (SORT (COPY FILELST))
  else FILELST)
  collect `(% FILE (% FILE . FILEVARS)
    %,
    (CONCAT "Brings up a File Operations menu for the file " FILE)
    (SUBITEMS %,@ (for TYPE in (SORT (Manager.FILESUBTYPES FILE))
      collect `(% TYPE (% FILE %, @ TYPE)
        %,
        (CONCAT "Creates a " TYPE " submenu for the file " FILE]))

```

**(Manager.MAINOPEN**

[LAMBDA (POSITION)

; Edited 10-Oct-2023 11:23 by mth

;;; Builds the manager main (FILELST) menu at the indicated position.

```

(SETQ MANAGER-FILE-MENU (create MENU
                                ITEMS _ (Manager.MAINMENUITEMS)
                                WHENSELECTEDFN _ (FUNCTION Manager.MAIN.WSF)
                                MENCOLUMNS _ 1
                                MENUOUTLINESIZE _ 0))

(LET (IW IH)
  ;; some of the complexity here is so that, in the odd case that there are more files than will fit on the screen, the result will be a scrollable
  ;; window
  (ADDMENU MANAGER-FILE-MENU
    (SETQ MANAGER-MAIN-WINDOW
      (CREATEW (with POSITION (with MENU MANAGER-FILE-MENU (SETQ IW (MIN (WIDTHIFWINDOW IMAGEWIDTH)
                                                                    SCREENWIDTH))
                                                                    ; width of file menu. Actually unlikely to be wider than
                                                                    ; screenwidth (!)
                                                                    (SETQ IH (MIN (HEIGHTIFWINDOW IMAGEHEIGHT T)
                                                                    SCREENHEIGHT))
                                                                    ; height of window; could possibly be higher than screen if lots of
                                                                    ; files
                                                                    (if (POSITIONP POSITION)
                                                                      then
                                                                    ; gave an initial position for the manager file menu
                                                                      POSITION
                                                                    elseif (WINDOWP MANAGER-MAIN-WINDOW)
                                                                      then
                                                                    ; if there was a window, put the new one in the same place (and
                                                                    ; close the old one)
                                                                      (PROG1 (Manager.ANCHORED-SET-POSITION IW IH)
                                                                    ;; (with REGION (WINDOWPROP MANAGER-MAIN-WINDOW (QUOTE
                                                                    ;; REGION)) (create POSITION XCOORD _ LEFT YCOORD _ BOTTOM))
                                                                      (CLOSEW MANAGER-MAIN-WINDOW))
                                                                    ; let user say where to put the menu
                                                                      (GETBOXPOSITION IW IH)))
                                                                    else
                                                                    (create REGION
                                                                      LEFT _ XCOORD
                                                                      WIDTH _ IW
                                                                      BOTTOM _ YCOORD
                                                                      HEIGHT _ IH))
                                                                    "Manager"))
      [WINDOWPROP MANAGER-MAIN-WINDOW 'BUTTONEVENTFN
        (FUNCTION (LAMBDA (WINDOW)
          (if (Manager.INTITLE? WINDOW)
            then [Manager.DO.COMMAND (MENU (OR MANAGER-MAIN-MENU
                                              (SETQ MANAGER-MAIN-MENU
                                                (create MENU
                                                  TITLE _ "Manager operations"
                                                  ITEMS _ MANAGER-MAIN-MENU-ITEMS
                                                  CENTERFLG _ T)
                                                  else (MENUBUTTONFN WINDOW)
            ;; Shrink to the manager icon, and remember to update when the expanding
            [WINDOWPROP MANAGER-MAIN-WINDOW 'ICONFN (FUNCTION (LAMBDA (WIN OICON)
              (SETQ MANAGER-MAIN-ICONW
                (if (NULL OICON)
                  then (OR MANAGER-MAIN-ICONW (ICONW MANAGER.BM
                                                                    MANAGER.BM.MASK
                                                                    ))
                  else OICON))
              [WINDOWPROP MANAGER-MAIN-ICONW 'EXPANDFN
                (FUNCTION (LAMBDA NIL
                  (Manager.MAINUPDATE NIL)
                  MANAGER-MAIN-ICONW]
              (SETQ Manager.ACTIVEFLG T)
              (Manager.MAINUPDATE T))
            )
          )
        )
      )
    )
  )

```

**(Manager.MAINUPDATE**

[LAMBDA (FROMUPDATE)

; Edited 16-Aug-87 21:34 by raf

;;; Updates the highlighting of the main (FILELST) menu. Does not handle adding or removing of names from FILELST. Typically called after  
 ;;; Manager.ALTERMARKING.

```

(if (NOT FROMUPDATE)
  then (LET ((Manager.ACTIVEFLG NIL))
    (UPDATEFILES)))
(Manager.INSUREHIGHLIGHTS MANAGER-FILE-MENU (for ITEM in (Manager.MENUITEMS MANAGER-FILE-MENU)
  when (CDR (GETPROP ITEM 'FILE)) collect ITEM))

```

**(Manager.MAKEFILE.ADV**



; Edited 20-Aug-87 15:04 by raf

; Edited 27-May-87 17:26 by raf

; Edited 9-Jul-87 14:06 by raf

; Edited 20-Aug-87 13:45 by raf

; Edited 16-Aug-87 22:06 by raf

```
(AND FILE (PROG (ITEMS MENU.ITEMS SUBTYPES)
  (SETQ MENU.ITEMS (FASSOC FILE (fetch (MENU ITEMS) of MANAGER-FILE-MENU)))
  (SETQ ITEMS (NTH MENU.ITEMS 4))
  (SETQ SUBTYPES (Manager.FILESUBTYPES FILE))
  (if [AND ITEMS (OR (NULL COMSTYPE)
    (if (FASSOC COMSTYPE (CDAR ITEMS))
      then (NULL (FMEMB COMSTYPE SUBTYPES))
      else (FMEMB COMSTYPE SUBTYPES)]
    then (RPLACA ITEMS `(SUBITEMS %,@ (for TYPE in SUBTYPES
      collect `(% , TYPE (% , FILE %,@ TYPE)
      %,
      (CONCAT "Creates a " TYPE " submenu for
              the file " FILE]))
```

**(Manager.SET-ANCHOR**

```
[LAMBDA (NEWANCHOR)
  (if (AND (FMEMB NEWANCHOR ' (ANCHOR-TL ANCHOR-TR ANCHOR-BL ANCHOR-BR))
        (NEQ Manager.WINDOW-ANCHOR NEWANCHOR))
    then (SETQ Manager.WINDOW-ANCHOR NEWANCHOR))
; Edited 10-Oct-2023 11:24 by mth
```

**(Manager.SORT.COMS**

```
[LAMBDA (A B)
; Edited 18-Nov-87 15:12 by raf
```

```
;;; This allows CLOS method definitions to display in a sorted fashion.
```

```
;;; They are stored on the fileCOMS variable as:
```

```
;;; (method-name (required-arg-type-specifiers))
```

```
(ALPHORDER (COND
  ((LITATOM A)
   A)
  (T (CONCAT A)))
(COND
  ((LITATOM B)
   B)
  (T (CONCAT B)))
```

**(Manager.SORTBYCOLUMN**

```
[LAMBDA (ITEMS)
  (PROG ((LENGTH (FLENGTH ITEMS))
         COLUMNCOUNT COLUMNLENGTH EXTRAITEMCOLUMNS RESULT)
    (if (NULL ITEMS)
        then (RETURN))
    (SORT ITEMS 'Manager.SORT.COMS)
    (SETQ COLUMNCOUNT (Manager.MENUCOLUMNS ITEMS))
    (SETQ COLUMNLENGTH (IQUOTIENT LENGTH COLUMNCOUNT))
    (SETQ EXTRAITEMCOLUMNS (IREMAINDER LENGTH COLUMNCOUNT))
    [SETQ RESULT (for I to COLUMNCOUNT collect (for J to (COND
      ((ILEQ I EXTRAITEMCOLUMNS)
       (ADD1 COLUMNLENGTH))
      (COLUMNLENGTH))
      collect (pop ITEMS)]
    (RETURN (while (CAR RESULT) join (DREMOVE NIL (for LST on RESULT collect (PROG1 (CAAR LST)
      (RPLACA LST (CDAR LST))))))
)
; Edited 19-Jun-87 20:58 by andyiii
```

```
[XCL:REINSTALL-ADVICE 'ADDFILE :AROUND ' ((:LAST (PROG1 (LET ((Manager.ACTIVEFLG NIL))
  *)
  (AND Manager.ACTIVEFLG (Manager.CHECKFILE FILE))))]

[XCL:REINSTALL-ADVICE 'ADDTOFILES? :AROUND ' ((:LAST (PROG1 (LET ((MANAGER-ADDTOFILES? T))
  *)
  (AND Manager.ACTIVEFLG (Manager.ADDTOFILES?))))]

[XCL:REINSTALL-ADVICE 'MAKEFILE :AROUND ' ((:LAST (PROG1 (LET ((Manager.ACTIVEFLG NIL))
  *)
  (AND Manager.ACTIVEFLG (Manager.MAKEFILE.ADV FILE OPTIONS))))]

[XCL:REINSTALL-ADVICE 'MARKASCHANGED :AROUND ' ((:LAST (PROG1 (LET ((Manager.ACTIVEFLG NIL))
  *)
  (AND Manager.ACTIVEFLG (Manager.ALTERMARKING
    NAME TYPE (OR REASON T))))))

[XCL:REINSTALL-ADVICE 'UNMARKASCHANGED :AROUND ' ((:LAST (LET (!VALUE)
  (PROG1 (LET ((Manager.ACTIVEFLG NIL))
    (SETQ !VALUE *))
    (AND Manager.ACTIVEFLG !VALUE
      (Manager.ALTERMARKING NAME TYPE NIL))))))

[XCL:REINSTALL-ADVICE 'UPDATEFILES :AROUND ' ((:LAST (PROG1 (LET ((Manager.ACTIVEFLG NIL))
  *)
  (AND Manager.ACTIVEFLG (Manager.MAINUPDATE T))))]

[XCL:REINSTALL-ADVICE 'ADDTOCOMS :AROUND ' ((:LAST (LET (!VALUE)
  (PROG1 (LET ((Manager.ACTIVEFLG NIL))
    (SETQ !VALUE *))
    (AND Manager.ACTIVEFLG (Manager.ADDADV !VALUE COMS
      NAME TYPE))))))

[XCL:REINSTALL-ADVICE 'DELFROMCOMS :AROUND ' ((:LAST (LET (!VALUE)
  (PROG1 (LET ((Manager.ACTIVEFLG NIL))
    (SETQ !VALUE *))
    (AND Manager.ACTIVEFLG (Manager.ADDADV !VALUE COMS
      NAME TYPE))))))

[XCL:REINSTALL-ADVICE '\ADDTOFILEBLOCK/ADDNEWCOM :AROUND ' ((:LAST (PROG1 (LET ((Manager.ACTIVEFLG NIL))
```

```

*)
(AND Manager.ACTIVEFLG (
  Manager.RESETSUBITEMS
  FILE TYPE))))]

[XCL:REINSTALL-ADVICE 'LOAD :AROUND '(:LAST (PROG1 (LET ((Manager.ACTIVEFLG NIL))
  *)
  (if Manager.ACTIVEFLG
    then (Manager.REMOVE.DUPLICATE.ADVISE FILE)
    (Manager.CHECKFILE FILE))))]

[XCL:REINSTALL-ADVICE 'LOADFNS :AROUND '(:LAST (PROG1 (LET ((Manager.ACTIVEFLG NIL))
  *)
  (if Manager.ACTIVEFLG
    then (Manager.REMOVE.DUPLICATE.ADVISE FILE)
    (Manager.CHECKFILE FILE))))]

[XCL:REINSTALL-ADVICE ' (MARKASCHANGED :IN DEFAULT.EDITDEFA0001)
:AROUND
' (:LAST (AND (AND (EQ NAME 'FILELST)
  (EQ TYPE 'VARS))
  *)]

(READWISE ADDFILE ADDTOFILES? MAKEFILE MARKASCHANGED UNMARKASCHANGED UPDATEFILES ADDTOCOMS DELFROMCOMS
\ADDTOFILEBLOCK/ADDNEWCOM LOAD LOADFNS (MARKASCHANGED :IN DEFAULT.EDITDEFA0001))

(DECLARE%: EVAL@COMPILE

(PUTPROPS GETDATUM MACRO ((KEY)
  (CDR (FASSOC KEY Manager.DATASPACE))))

(PUTPROPS PUTDATUM MACRO ((KEY VALUE)
  (PUTASSOC KEY VALUE Manager.DATASPACE)))

(PUTPROPS Manager.TTYCOMMAND MACRO ((X . Y)
  (PROGN (ALLOW.BUTTON.EVENTS)
    X . Y)))

)

(PUTPROPS ADVISE MANAGER-DEFINITION-TYPE-COMMANDS (("ReAdvise" 'READWISE "Enable all advice under this name")
  ("UnAdvise" 'UNADVISE "Disable all advice under this name")
  ))

(PUTPROPS FNS MANAGER-DEFINITION-TYPE-COMMANDS (("Break" 'BREAK "Break this function")
  ("Trace" 'TRACE "Trace this function")
  ("UNBreak" 'UNBREAK "UnBreak this function")
  ("Compile" 'COMPILE "Compile this function"
    (SUBITEMS ("Compile" 'COMPILE "Compile this function")
      (DISASSEMBLE 'DISASSEMBLE "Print the compiled
        code of the function"))))
  [" MasterScope " 'DESCRIBE "Invoke MasterScope to DESCRIBE the
    function" (SUBITEMS ("Describe" 'DESCRIBE "Invoke
      MasterScope to describe this
        function")
        ("Show Paths"
          'SHOWPATHTO "Invoke MasterScope to
            show who calls this function"
            (SUBITEMS ("To" 'SHOWPATHTO
              "Invoke MasterScope
                to show who calls
                  this function")
                ("From" 'SHOWPATHFROM
                  "Invoke MasterScope to
                    show who is called by
                      this function"]
          ("?" 'ARGS "The function's argument list"))))

(PUTPROPS RECORDS MANAGER-DEFINITION-TYPE-COMMANDS (("Fields" 'FIELDS "List the field names"))

(PUTPROPS VARS MANAGER-DEFINITION-TYPE-COMMANDS [(" MasterScope " 'DESCRIBE "Who uses this?"
  (SUBITEMS ("Who uses?" 'DESCRIBE "Who uses this?"))

(PUTPROPS FUNCTIONS MANAGER-DEFINITION-TYPE-COMMANDS (("Break" 'BREAK "Break this function")
  ("Trace" 'TRACE "Trace this function")
  ("UNBreak" 'UNBREAK "UnBreak this function")
  ("Compile" 'COMPILE "Compile this function"
    (SUBITEMS ("Compile" 'COMPILE "Compile this
      function")
        ("Disassemble" 'DISASSEMBLE "Print the
          compiled code of the function"))))
  ("?" 'ARGS "The function's argument list"))

(ADDTOVAR BackgroundMenuCommands (File% Manager (MANAGER)
  "Starts the menu driven file manager"))

(LSUBST 'Manager NIL BackgroundMenuCommands)

```

```

{MEDLEY}<lispusers>MANAGER.;1

;; remove old manager entry if it exists

(SETQ BackgroundMenu NIL)

;; cause the backGround menu to be rebuilt
(MANAGER.RESET (CL:SYMBOL-VALUE 'Manager.ACTIVEFLG))

;; Shutdown any old manager windows and restart if we're already running.

(if (STREQUAL MANAGER-ACTIVITY-WINDOW-TITLE (WINDOWPROP NIL 'TITLE))
    then                                     ; If we're in the manager activity window, close it, since we
                                           ; dropped the pointer to it in MANAGER.RESET.
        (CLOSEW NIL))

(PUTPROPS MANAGER MAKEFILE-ENVIRONMENT (:PACKAGE "INTERLISP" :READTABLE "INTERLISP"))

(PUTPROPS MANAGER FILETYPE :COMPILE-FILE)

(DECLARE%: DONTEVAL@LOAD DOEVAL@COMPILE DONTCOPY COMPILERVERS

(ADDTOPVAR NLAMA )

(ADDTOPVAR NLAML )

(ADDTOPVAR LAMA )
)

(PUTPROPS MANAGER COPYRIGHT (NONE))

```

---

## FUNCTION INDEX

MANAGER .....	5	Manager.FILELSTCHANGED? .....	13	Manager.MAKEFILE.ADV .....	16
Manager.ADDADV .....	6	Manager.FILESUBTYPES .....	14	Manager.MENUCOLUMNS .....	17
Manager.ADDTOFILES? .....	6	Manager.GET.ENVIRONMENT .....	14	Manager.MENUHASITEM .....	17
Manager.ALTERMARKING .....	6	Manager.GETFILE .....	14	Manager.MENUITEMS .....	17
Manager.ANCHORED-SET-POSITION ....	7	Manager.HIGHLIGHT .....	11	Manager.PROMPT .....	11
Manager.CHANGED? .....	11	Manager.HIGHLIGHTED .....	13	Manager.REMOVE.DUPLICATE.ADVISE ..	17
Manager.CHECKFILE .....	11	Manager.insurefilehighlights ....	11	MANAGER.RESET .....	6
Manager.COLLECTCOMS .....	11	Manager.INSUREHIGHLIGHTS .....	13	Manager.RESETSUBITEMS .....	17
Manager.COMS.WSF .....	12	Manager.INTITLE? .....	14	Manager.SET-ANCHOR .....	18
Manager.COMSOPEN .....	12	Manager.MAIN.WSF .....	15	Manager.SORT.COMS .....	18
Manager.COMSUPDATE .....	13	Manager.MAINCLOSE .....	15	Manager.SORTBYCOLUMN .....	18
Manager.DO.COMMAND .....	7	Manager.MAINMENUITEMS .....	15	Manager.WINDOW .....	11
Manager.DO.COMMAND.PROCFN .....	7	Manager.MAINOPEN .....	16		
Manager.FILECHANGES .....	13	Manager.MAINUPDATE .....	16		

---

## VARIABLE INDEX

*UNMANAGED-TYPES* .....	2	MANAGER-MAIN-ICONW .....	5
BackgroundMenuCommands .....	19	MANAGER-MAIN-MENU-ITEMS .....	4
MANAGER-ACTIVITY-WINDOW-TITLE .....	2	MANAGER-MAIN-WINDOW .....	5
MANAGER-ADDTOFILES? .....	2	MANAGER-MARKED-SHADE .....	5
MANAGER-CASES .....	2	MANAGER-OPEN-WINDOWS .....	5
MANAGER-FILE-FILE-RELATION-COMMANDS .....	2	MANAGER-WINDOWS .....	5
MANAGER-FILE-FILE-RELATION-MENU .....	5	Manager.ACTIVEFLG .....	5
MANAGER-FILE-MENU .....	5	MANAGER.BM .....	5
MANAGER-FILE-OPERATIONS-COMMANDS .....	2	MANAGER.BM.MASK .....	5
MANAGER-FILE-OPERATIONS-MENU .....	5	Manager.DATASPACE .....	5
MANAGER-FILELST-MENU .....	5	Manager.MENUROWS .....	5
MANAGER-ITEM-FILE-RELATION-COMMANDS .....	3	Manager.SORTFILELSTFLG .....	5
MANAGER-ITEM-OPERATION-COMMANDS .....	3	Manager.WINDOW-ANCHOR .....	5

---

## ADVICE INDEX

ADDFILE .....	18	MAKEFILE .....	18
ADDTOCOMS .....	18	MARKASCHANGED .....	18
ADDTOFILES? .....	18	UNMARKASCHANGED .....	18
DELFROMCOMS .....	18	UPDATEFILES .....	18
LOAD .....	19	\ADDTOFILEBLOCK/ADDNEWCOM .....	18
LOADFNS .....	19	(MARKASCHANGED :IN DEFAULT.EDITDEFA0001) .....	19

---

## PROPERTY INDEX

ADVICE .....	19	FNS .....	19	FUNCTIONS ....	19	MANAGER .....	20	RECORDS .....	19	VARS .....	19
--------------	----	-----------	----	----------------	----	---------------	----	---------------	----	------------	----

---

## MACRO INDEX

GETDATUM .....	19	Manager.TTYCOMMAND .....	19	PUTDATUM .....	19
----------------	----	--------------------------	----	----------------	----

---