```
;;
;; Copyright (c) 1989-1991, 1993, 2018 by Venue & Xerox Corporation.


(RPAQQ COPYFILESCOMS
       ((FNS COPYFILES MAPFILES MAPFILES1 COPIEDFILENAME COPIEDFILEPATTERN COPIEDFILEMATCH COPIEDFROMSPEC
             COPIEDTOSPEC ESPATTERN NOHOST COMPAREFILES)
        (COMS  ;; For concatenating a list of files into one file.

               (FNS CONCATFILES))
        (COMS  ;; For splitting a big file into several files.

               (FNS SPLITFILE))
        (COMS  ;; For making DOS file systems

               (FNS DOSLINKER SHORTEN))
        (I.S.OPRS INFILES)))

(DEFINEQ

(COPYFILES
  [LAMBDA (FROMSPEC TOSPEC OPTIONS)
    (DECLARE (SPECVARS FROMSPEC TOSPEC))                            ; Edited 27-Sep-89 15:07 by bvm

    ;; Copies the files specified in FROMSPEC to the destination in TOSPEC.  Which versions get copied, whether to copy old files, etc.  is controlled
    ;; by OPTIONS.

    (SETQ TOSPEC (\ADD.CONNECTED.DIR TOSPEC))
    (SETQ FROMSPEC (\ADD.CONNECTED.DIR FROMSPEC))
    (LET
     ((*UPPER-CASE-FILE-NAMES* NIL)
      (COPYFILESOUTPUT T)
      (COPYFILES.WHENTOSKIP (FUNCTION ILEQ))
      (COPYFILESALWAYS T)
      (COPYFILESVERSIONS NIL)
      (COPYFILESASK NIL)
      (COPYFILESASKDEFAULT)
      (COPYFILESREPLACE NIL)
      (COPYFILESPURGESOURCE NIL)
      (COPYFILESPURGE NIL)
      (COPYFILESTERSE)
      (COPYFILESTOSPEC (COPIEDTOSPEC TOSPEC))
      (SECONDARYSPEC)
      (COPYFILESFN 'COPYFILE)
      (COPYFILESSKIPFN 'NILL)
      (DONTCOPY)
      COPYFILESFROMSPEC)
     (DECLARE (SPECVARS . T))
     [for X inside OPTIONS
        do ;; Run thru the options, turning them into internal flag settings and functional specifications.

           (SELECTQ X
               ((QUIET :QUIET)                                  ; Don't want to hear about files as they're copied.  Set the output
                                                                ; file to NIL to suppress printing.

                   (SETQ COPYFILESOUTPUT NIL))
               ((TERSE :TERSE)                                  ; Only print a . per file copied.  Set the TERSE flag.
                   (SETQ COPYFILESOUTPUT NIL)
                   (SETQ COPYFILESTERSE T))
               ((RENAME MOVE :RENAME :MOVE)                     ; He wants the files moved, not copied.
                   (SETQ COPYFILESFN 'RENAMEFILE))
               ((ALWAYS :ALWAYS)                                ; ALWAYS copy the files specified.
                   (SETQ COPYFILESALWAYS T)                     ; Tell it so
                   (SETQ COPYFILES.WHENTOSKIP (FUNCTION NILL))  ; And say never to skip a potential file

                   )
               (>                                               ; Only copy if the source has a newer version than the
                                                                ; destination.

                   (SETQ COPYFILES.WHENTOSKIP (FUNCTION ILEQ))
                   (SETQ COPYFILESALWAYS NIL))
               ((= =A)                                          ; = without ALWAYS doesn't make a lot of sense
                   (SETQ COPYFILES.WHENTOSKIP (FUNCTION TRUE))
                   (SETQ COPYFILESALWAYS T))
               ((%# /=)                                         ; Skip files that are the same on the destination
                   (SETQ COPYFILES.WHENTOSKIP (FUNCTION EQUAL))
                   (SETQ COPYFILESALWAYS NIL))
               (ALLVERSIONS (SETQ COPYFILESVERSIONS T))
```

```
                    ((%#A /=A)
                          (SETQ COPYFILES.WHENTOSKIP (FUNCTION EQUAL)))
                          (SETQ COPYFILESALWAYS T))
                    (>A (SETQ COPYFILES.WHENTOSKIP (FUNCTION ILEQ))
                          (SETQ COPYFILESALWAYS T))
                    ((ASK :ASK)
                          (SETQ COPYFILESASK T))
                    ((PURGE :PURGE)
                          (SETQ COPYFILESPURGE T))
                    ((PURGESOURCE :PURGESOURCE)
                          (SETQ COPYFILESPURGESOURCE T)
                          (SETQ COPYFILESALWAYS NIL))
                    ((REPLACE :REPLACE)
                          (SETQ COPYFILESREPLACE T))
                    (SELECTQ (CAR (LISTP X))
                          (OUTPUT (SETQ COPYFILESOUTPUT (OPENSTREAM (CADR X)
                                                             'OUTPUT
                                                             'NEW)))
                          (ASK (SETQ COPYFILESASK T)
                                (SETQ COPYFILESASKDEFAULT (CADR X)))
                          (DONTCOPY (SETQ DONTCOPY (CDR X)))
                          (COPYFN                                      ; Use this instead of COPYFILE
                                (SETQ COPYFILESFN (CADR X)))
                          (SECONDARY  ;; Use FROMSPECT/TOSPEC to decide what files to copy, but actually do the copying to this secondary spec
                                      ;; (if the file's not there already).  Also, if we skip a file because the date comparison failed or the destination
                                      ;; doesn't exist, we delete the corresponding file(s) on the secondary location.  This is pretty strange --bvm.

                                (SETQ SECONDARYSPEC (COPIEDTOSPEC (CADR X)))
                                [SETQ COPYFILESFN
                                 (FUNCTION (LAMBDA (SOURCE DEST)

                                        ;; This gets called when we're to "copy" from SOURCE to DEST

                                        (LET (DST FULLDST)
                                              (if [CAR (ERSETQ (AND [SETQ FULLDST
                                                                       (INFILEP (SETQ DST (COPIEDFILENAME
                                                                                                 SOURCE
                                                                                                 COPYFILESFROMSPEC
                                                                                                 SECONDARYSPEC]
                                                                    (= (GETFILEINFO SOURCE 'ICREATIONDATE)
                                                                       (GETFILEINFO FULLDST 'ICREATIONDATE]
                                                  then (PRINTOUT COPYFILESOUTPUT "[backed up on " FULLDST
                                                                   "]" T)
                                                        "!"
                                                  else (COPYFILE SOURCE DST]
                                [SETQ COPYFILESSKIPFN (FUNCTION (LAMBDA (SOURCE)

                                                                    ;; This gets called when we skip SOURCE

                                                                    (LET ((BACKUP (COPIEDFILENAME SOURCE
                                                                                       COPYFILESFROMSPEC
                                                                                       SECONDARYSPEC)))
                                                                         (bind BK while (SETQ BK (INFILEP BACKUP))
                                                                            do (PRINTOUT COPYFILESOUTPUT "
                                                                                   [deleting " BK "...")
                                                                                (COND
                                                                                   ((NOT (DELFILE BK))
                                                                                    (PRINTOUT COPYFILESOUTPUT
                                                                                        "couldn't!]")
                                                                                    (RETURN))
                                                                                   (T (PRINTOUT COPYFILESOUTPUT
                                                                                          "ok]"])
                          (ERROR X "unrecognized option"]
            (if (AND COPYFILESASK (NOT COPYFILESOUTPUT))
                then (SETQ COPYFILESOUTPUT T))
            (if (OR (LISTP COPYFILESTOSPEC)
                    (LISTP SECONDARYSPEC))
               then                                                        ; copiedfilename will want the from spec broken down to do
                                                                           ; pattern matching.
                  (SETQ COPYFILESFROMSPEC (COPIEDFROMSPEC FROMSPEC)))
            (MAPFILES FROMSPEC [FUNCTION (LAMBDA (FILENAME DT1)
                                    [PROG (NEWFILENAME NF CF DT2 HELPFLAG)
                                          (DECLARE (SPECVARS HELPFLAG))
                                                                           ; So that errors don't cause breaks
                                          (if DONTCOPY
                                              then (if (CL:MEMBER (UNPACKFILENAME.STRING FILENAME 'EXTENSION)
                                                              DONTCOPY :TEST 'STRING.EQUAL)
                                                       then (AND COPYFILESOUTPUT (PRINTOUT COPYFILESOUTPUT
                                                                                      FILENAME " ignored." T))
                                                            (RETURN)))
                                          (AND COPYFILESOUTPUT (printout COPYFILESOUTPUT .TAB0 0 FILENAME))
                                                                           ; List the candidate file's name
                                          [OR (ERSETQ (SETQ NEWFILENAME (COPIEDFILENAME FILENAME
                                                                             COPYFILESFROMSPEC COPYFILESTOSPEC
                                                                             COPYFILESVERSIONS)))
                                              (RETURN (AND COPYFILESOUTPUT (printout COPYFILESOUTPUT " illegal
                                                                   file name "]
                                                                           ; Find out what the file's name would be at the destination.
                                          [if (OR (NOT COPYFILESALWAYS)
```

```
                                                 (NEQ COPYFILES.WHENTOSKIP 'NILL))
                               then                              ; We aren't ALWAYS copying.  So have to check this file to see if
                                                                ; it meets the copy criteria.
                                           (COND
                                              [(SETQ NF (INFILEP NEWFILENAME))
                                                                ; There is a file of the same name at the destination.  CHeck it
                                                                ; out.
                                               (SETQ DT2 (GETFILEINFO NF 'ICREATIONDATE))
                                                                ; The destination file's create date
                                               (AND COPYFILESOUTPUT (printout COPYFILESOUTPUT " ["
                                                                           (GDATE DT1)
                                                                           "]" " vs. " NF "["
                                                                           (if DT2
                                                                               then (GDATE DT2)
                                                                             else "no date?")
                                                                           "]"))
                                                                ; Tell the user we're comparing dates
                                               (COND
                                                  ((AND DT2 (CL:FUNCALL COPYFILES.WHENTOSKIP DT1 DT2))
                                                                ; If the file has a create date, and it meets the SKIP criteria, then
                                                                ; skip over this file
                                                   (CL:FUNCALL COPYFILESSKIPFN FILENAME)
                                                   (AND COPYFILESOUTPUT (printout COPYFILESOUTPUT "
                                                                           skipped."))
                                                   (RETURN]
                                              (COPYFILESPURGESOURCE
                                                                ; We're to purge the source directory of non-corresponding files
                                                   (AND COPYFILESOUTPUT (printout COPYFILESOUTPUT "
                                                                           (no corresponding "
                                                                           NEWFILENAME "), "))
                                               [COND
                                                  ((OR (NOT COPYFILESASK)
                                                       (EQ 'Y (ASKUSER DWIMWAIT
                                                                      COPYFILESASKDEFAULT
                                                                      "delete? " NIL T)))
                                                   ([LAMBDA (STR)
                                                      (AND COPYFILESOUTPUT (printout
                                                                                 COPYFILESOUTPUT
                                                                                 STR]
                                                     (if (DELFILE FILENAME)
                                                         then " deleted."
                                                       else " couldn't delete."]
                                                   (RETURN))
                                              ((NOT COPYFILESALWAYS)
                                                                ; file doesn't exist on destination
                                                   (CL:FUNCALL COPYFILESSKIPFN FILENAME)
                                                   (AND COPYFILESOUTPUT (printout COPYFILESOUTPUT " does not
                                                                           exist on destination --
                                                                           skipped"))
                                                   (RETURN]
                               (if (AND COPYFILESREPLACE NF)
                                   then (SETQ NEWFILENAME NF))
                               (if COPYFILESOUTPUT
                                   then                              ; Write out the file's new name, and tell him we're copying or
                                                                ; moving it.
                                           (printout COPYFILESOUTPUT (SELECTQ COPYFILESFN
                                                                           (COPYFILE " copy")
                                                                           (RENAMEFILE " rename")
                                                                           " process"))
                                           (if (NOT NF)
                                               then (printout COPYFILESOUTPUT " to (new file) "
                                                           NEWFILENAME)))
                               (COND
                                  ((AND COPYFILESASK (NEQ (ASKUSER DWIMWAIT COPYFILESASKDEFAULT "? "
                                                                 NIL T)
                                                         'Y))
                                   (RETURN)))
                               [OR (ERSETQ (SETQ CF (CL:FUNCALL COPYFILESFN FILENAME NEWFILENAME)))
                                   (RETURN (AND COPYFILESOUTPUT (printout COPYFILESOUTPUT " failed."]
                               (if COPYFILESOUTPUT
                                   then (if (AND (NOT COPYFILESASK)
                                                 (NOT NF)
                                                 (STRPOS NEWFILENAME CF 1 NIL 1 NIL (UPPERCASEARRAY)))
                                            then (printout COPYFILESOUTPUT
                                                           (OR (SUBSTRING CF (ADD1 (NCHARS NEWFILENAME)))
                                                               "."))
                                          else (printout COPYFILESOUTPUT " => " CF]
                               (AND COPYFILESTERSE (PRIN1 "." COPYFILESTERSE]
                '(ICREATIONDATE)
                "*"
                (if COPYFILESVERSIONS
                    then "*"
                  else "")
                NIL
                (SELECTQ (fetch (FDEV GENERATEFILES) of (\GETDEVICEFROMNAME FROMSPEC))
                   ((\FTP.GENERATEFILES \LEAF.GENERATEFILES \NSFILING.GENERATEFILES \TCPFTP.GENERATEFILES)
                                                                ; If source is PupFtp, TCP, or NS, enumerate the whole directory
                                                                ; first, lest some awkward copy cause the source enumeration
```

```
                                                                                   ; stream to die.
                        T)
                    NIL))
        [if COPYFILESPURGE
            then                                                                   ; delete from source if doesn't exist on destination
                    (AND COPYFILESOUTPUT (printout COPYFILESOUTPUT T "Deleting files on destination but not on
                                    source" T))
                    (COPYFILES TOSPEC FROMSPEC (APPEND (if COPYFILESOUTPUT
                                                          then (LIST (LIST 'OUTPUT COPYFILESOUTPUT)))
                                                       (if COPYFILESASK
                                                          then (LIST (LIST 'ASK COPYFILESASKDEFAULT)))
                                                       '(= PURGESOURCE]
        (COND
            (COPYFILESOUTPUT (TAB 0 0 COPYFILESOUTPUT))
                (NEQ COPYFILESOUTPUT T)
                (CLOSEF COPYFILESOUTPUT])
```

(**MAPFILES**
```
  [LAMBDA (FILESPEC FN ATTRIBUTES DEFAULTEXT DEFAULTVERS INCLUDE-DIRECTORIES ENUMERATE-FIRST)
```
  ;; Edited 26-Mar-2022 11:43 by rmk: Respect DEFAULTEXT/VERS in singleton no-stars case

  ;; Edited 6-Apr-2018 21:14 by rmk:

  ;; Run thru all the files that match FILESPEC, calling FN on each such file name, with remaining args being the value of each of the ATTRIBUTES
  ;; of the file

```
    (if (LISTP FILESPEC)
        then (for X in FILESPEC do (MAPFILES X FN DEFAULTEXT DEFAULTVERS ATTRIBUTES INCLUDE-DIRECTORIES
                                        ENUMERATE-FIRST))
      elseif [OR (STRPOS "*" FILESPEC)
                 (FMEMB (NTHCHARCODE FILESPEC -1)
                        (CHARCODE (/ > %) %] } %:]
        then                                                                   ; Pattern or directory spec
            (SETQ FILESPEC (DIRECTORY.FILL.PATTERN FILESPEC DEFAULTEXT DEFAULTVERS))
            (if ENUMERATE-FIRST
                then                                                           ; Generate all the files first, then apply fn
                    (for PAIR in [XCL:WITH-COLLECTION (MAPFILES1 FILESPEC ATTRIBUTES INCLUDE-DIRECTORIES
                                                        (FUNCTION (CL:LAMBDA (NAME &REST ATTRS)
                                                                    (XCL:COLLECT (CONS NAME ATTRS]
                        do (CL:APPLY FN (CAR PAIR)
                                (CDR PAIR)))
                else                                                           ; Call on each one as we go
                    (MAPFILES1 FILESPEC ATTRIBUTES INCLUDE-DIRECTORIES FN))
      elseif (SETQ FILESPEC (INFILEP (PACKFILENAME.STRING 'BODY FILESPEC 'EXTENSION DEFAULTEXT 'VERSION
                                        DEFAULTVERS)))
        then
            ;; rmk: Singleton, no stars.  We don't want to coerce NIL DEFAULTVERS/EXT to *, but still we want to pay attention to them.  Hence,
            ;; do the packfilename

            (CL:APPLY FN FILESPEC (for ATTR inside ATTRIBUTES collect (GETFILEINFO FILESPEC ATTR]
```

(**MAPFILES1**
```
  [LAMBDA (FILESPEC ATTRIBUTES INCLUDE-DIRECTORIES FN)                    ; Edited 27-Sep-89 14:49 by bvm
```
  ;; Enumerate FILESPEC (pattern must already be filled) and apply FN to each file and its ATTRIBUTES

```
    (RESETLST
        [LET ([FILEGROUP (\GENERATEFILES FILESPEC (SETQ ATTRIBUTES (MKLIST ATTRIBUTES))
                            '(SORT RESETLST]
              NAME LEN)
            (while (SETQ NAME (\GENERATENEXTFILE FILEGROUP))
                unless [PROGN                                              ; Skip IFS's <dir>.;1 file.  Also other dir files unless
                                                                          ; INCLUDE-DIRECTORIES is true.
                            (OR (AND (>= (SETQ LEN (NCHARS NAME))
                                        4)
                                    (STRING-EQUAL NAME ".;1" :START1 (- LEN 4)))
                                (AND (NOT INCLUDE-DIRECTORIES)
                                    (FMEMB (NTHCHARCODE NAME LEN)
                                        (CHARCODE (/ >]
                do (if (NULL (CDR ATTRIBUTES))
                        then                                              ; Optimize slightly for the case of one attribute
                            (CL:FUNCALL FN NAME (\GENERATEFILEINFO FILEGROUP (CAR ATTRIBUTES)))
                        else (CL:APPLY FN NAME (for ATTR in ATTRIBUTES collect (\GENERATEFILEINFO FILEGROUP ATTR]])
```

(**COPIEDFILENAME**
```
  [LAMBDA (FILENAME COPIEDFROMSPEC COPIEDTOSPEC PRESERVEVERSION) ; Edited 27-Sep-89 15:03 by bvm
```
  ;; FILENAME is the file produced by the directory enumeration. COPIEDFROMSPEC is the parsed 'FROM' specification, and COPIEDTOSPEC is
  ;; either a directory specification (string) or else a list, CDR of which is a list of character atoms.

```
    [SETQ FILENAME (COND
                      [(LISTP COPIEDTOSPEC)                               ; destination is a pattern, so have to be fancy.  NOHOST strips
                                                                         ; off the HOST field
                          (CONCATLIST (LET [(FROMCHARS (NOHOST (UNPACK FILENAME]
                                          [if (NOT PRESERVEVERSION)
                                              then                       ; Discard the version
                                                  (RPLACD (NLEFT FROMCHARS 1 (FMEMB '; FROMCHARS]
                                          (COPIEDFILEPATTERN COPIEDFROMSPEC (CDR COPIEDTOSPEC))
```

```
                                                     FROMCHARS]
                          (T (CL:APPLY (FUNCTION PACKFILENAME.STRING)
                                  'DIRECTORY COPIEDTOSPEC 'HOST NIL 'DEVICE NIL 'DIRECTORY NIL
                                  '(,@(AND (NOT PRESERVEVERSION)
                                           '(VERSION NIL))
                                     BODY
                                     ,FILENAME]
       (COND
          ((EQ (NTHCHARCODE FILENAME -1)
               (CHARCODE %.))
```

;; this is a terrible kludge, to get around the problem that for some devices, (INFILEP 'FOO.') fails while (INFILEP 'FOO') doesn't. This
;; stripping off of a terminal '.' doesn't hurt, but doesn't belong here. Necessary for getting a working version for the harmony release.

```
           (SUBSTRING FILENAME 1 -2 FILENAME))
          (T FILENAME)])
```

(**COPIEDFILEPATTERN**
```
  [LAMBDA (FRPAT TOPAT CHARS)                                          ; Edited 27-Sep-89 15:58 by bvm
    (while [AND FRPAT CHARS (EQ (U-CASE (CAR FRPAT))
                                (U-CASE (CAR CHARS]
       do                                                              ; Skip to the first place where pattern and actual name differ
           (pop FRPAT)
           (pop CHARS))
    (NCONC (on old TOPAT while (NEQ (CAR TOPAT)
                                    '*)
                 collect (CAR TOPAT))
           (COND
              ((AND FRPAT (NEQ (CAR FRPAT)
                               '*))                                    ; Ran out of pattern before getting to a *
                (if [AND (NULL CHARS)
                         [OR [if (AND (EQ (CAR FRPAT)
                                          '%.)
                                      (EQ (CADR FRPAT)
                                          '*))
                                 then (NULL (SETQ FRPAT (CDDR FRPAT]
                             (EQUAL FRPAT '(; *]
                         (OR (NULL TOPAT)
                             (EQUAL TOPAT '                            (*)]
                    then NIL
                  else (ERROR "--From Spec doesn't match generated file")))
              (TOPAT                                                   ; both TOPAT and FRPAT start with *
                (NCONC [LDIFF CHARS (SETQ CHARS (for X on CHARS when (COPIEDFILEMATCH X (CDR FRPAT))
                                                   do                  ; Find the last tail of CHARS that matches the pattern
                                                       (SETQ $$VAL X]
                       (COPIEDFILEPATTERN (CDR FRPAT)
                                          (CDR TOPAT)
                                          CHARS)))
              (T (OR (COPIEDFILEMATCH CHARS FRPAT)
                     (ERROR "file pattern doesn't match"))
                 NIL]])
```

(**COPIEDFILEMATCH**
```
  [LAMBDA (CHARS FRPAT)                                                ; Edited 27-Sep-89 15:44 by bvm
    (PROG [(SEMI* '(; *]
      LP   (if (NULL FRPAT)
                then (RETURN (NULL CHARS))
              elseif (EQ (CAR FRPAT)
                         '*)
                then                                                   ; Match arbitrarily many CHARS
                     [RETURN (OR (NULL (SETQ FRPAT (CDR FRPAT)))
                                 (EQUAL FRPAT SEMI*)
                                 (find X on CHARS suchthat (COPIEDFILEMATCH X FRPAT]
              elseif (NULL CHARS)
                then (RETURN (EQUAL FRPAT SEMI*))
              elseif (EQ (U-CASE (pop FRPAT))
                         (U-CASE (pop CHARS)))
                then (GO LP)
              else (RETURN)])
```

(**COPIEDFROMSPEC**
```
  [LAMBDA (FROMSPEC)                                                   ; Edited 27-Sep-89 15:52 by bvm
```

;; Return something for copiedfilepattern to work on

```
    (SETQ FROMSPEC (MKSTRING FROMSPEC))
    [SETQ FROMSPEC (NOHOST (ESPATTERN (LET ([DIREND (for I from (SUB1 (OR (CL:POSITION #\* FROMSPEC)
                                                                          (NCHARS FROMSPEC)))
                                                       to 0 by -1 do (CASE (CL:CHAR FROMSPEC I)
                                                                        ((#\/ #\> #\}) (RETURN I)))]
                                            CAN)
                                       (if [AND DIREND (SETQ CAN (DIRECTORYNAME (CL:SUBSEQ FROMSPEC 0
                                                                                  (ADD1 DIREND]
                                           then                        ; Canonicalize the directory before we proceed, so that files
                                                                       ; coming back have a hope of matching
                                                (CONCAT CAN (CL:SUBSEQ FROMSPEC (ADD1 DIREND)))
```

```
                                                          else FROMSPEC]
        (if (NOT (FMEMB '; FROMSPEC))
           then                                                        ; Add ;* to match gracefully against real file names like foo.baz;3
                (NCONC FROMSPEC (LIST '; '*))
         else FROMSPEC])
```

(**COPIEDTOSPEC**
```
  [LAMBDA (SPEC)                                                       ; Edited 29-Oct-87 16:23 by jds

    ;; Create the spec for what file(s) are to be copied TO:  If there's a * in the spec or there's more than just directory specified, then return the pattern
    ;; for filename matching; if a directory got specified, return that.

    (COND
       ((STRPOS "*" SPEC)                                              ; There are wildcards in the name.
        (CONS 'PATTERN (ESPATTERN SPEC)))
       ((UNPACKFILENAME.STRING SPEC 'NAME)                             ; There's more than just a directory spec.
        (CONS 'PATTERN (ESPATTERN SPEC)))
       (T                                                             ; It's a directory (or had better be!)
          (OR (DIRECTORYNAME SPEC NIL T)
              (ERROR SPEC "not a valid directory"]
```

(**ESPATTERN**
```
  [LAMBDA (X)                                                          ; Edited 27-Sep-89 15:50 by bvm
    (SETQ X (UNPACK X))
    (for Y on X do (SELECTQ (CAR Y)
                       ((*)                                            ; Turn *.*; into *;
                           (if [AND (EQ (CADR Y)
                                        '%.)
                                    (EQ (CADDR Y)
                                        '*)
                                    (FMEMB (CADDDR Y)
                                           '(NIL ;]
                              then (RPLACD Y (CDDDR Y))))
                       NIL))
    X])
```

(**NOHOST**
```
  [LAMBDA (UP)
    (SELECTQ (CAR UP)
        (({ %( %[)
            (do (pop UP)
                (SELECTQ (CAR UP)
                    (NIL (RETURN))
                    ((} %) %])
                        (RETURN (pop UP)))
                    (%' (pop UP))
                    NIL)))
        NIL)
    UP])
```

(**COMPAREFILES**
```
  [LAMBDA (OLDFILE NEWFILE OSTART OEND)                                ; Edited  9-Apr-91 16:42 by jds
                                                                       ; Compare two files to see if their contents are the same.

    ;; If OSTART & OEND are specified, they are as for COPYBYTES--I.E., fileptrs into the OLD file to compare the new file to.

    [CL:WITH-OPEN-STREAM [OSTREAM (OPENSTREAM OLDFILE 'INPUT 'OLD '(SEQUENTIAL T]
        (CL:WITH-OPEN-STREAM [NSTREAM (OPENSTREAM NEWFILE 'INPUT 'OLD '(SEQUENTIAL T]
            (LET (OLEN NLEN NBYTES OBYTE NBYTE)
                (SETQ OLEN (GETFILEINFO OSTREAM 'LENGTH))
                (SETQ NLEN (GETFILEINFO NSTREAM 'LENGTH))
                [COND
                    [OSTART                                            ; He specified a starting fileptr or a char count.
                        (COND
                            ((NOT OEND)                                ; It was a char count (no ending point specified)
                             (SETQ NBYTES OSTART)
                             (SETQ OSTART 0))
                            (T                                         ; He specified an ending point.
                                (SETFILEPTR OSTREAM OSTART)
                                (SETQ NBYTES (IDIFFERENCE OEND OSTART))
                                (SETQ OSTART 0]
                    (T                                                 ; Nothing specified; run thru the whole file.
                        (SETQ OSTART 0)
                        (SETQ NBYTES OLEN)
                        (COND
                            ((NOT (EQP OLEN NLEN))                     ; If they files are of different lengths, they aren't the same.
                             (ERROR "File lengths differ:  " (CONCAT OLEN " vs " NLEN]
                [COND
                    (OLEN                                              ; FTP returns NIL for the length of an empty file!
                        (for BYTEPOS from OSTART to (SUB1 NBYTES)
                           do (COND
                                  ((NEQ (SETQ OBYTE (BIN OSTREAM))
                                        (SETQ NBYTE (BIN NSTREAM)))
                                   (PRINTOUT T "Files differ at byte " BYTEPOS ", old-file has " OBYTE
                                          " but new file has " NBYTE "." T)
```

```
                                              (OR (ASKUSER NIL NIL "Continue comparing?")
                                                  (RETURN]
                             (CLOSEF? OSTREAM)
                             (CLOSEF? NSTREAM]
        T])
)
```

;; For concatenating a list of files into one file.

```
(DEFINEQ
```

(**CONCATFILES**
```
   [LAMBDA (INPUT-FILES OUTPUT-FILE)
      (CL:WITH-OPEN-STREAM [OUT (OPENSTREAM OUTPUT-FILE 'OUTPUT 'NEW '(SEQUENTIAL T]
             (for FILE in INPUT-FILES do (CL:WITH-OPEN-STREAM [IN (OPENSTREAM FILE 'INPUT 'OLD '(SEQUENTIAL T]
                                                (COPYBYTES IN OUT])
)
```

;; For splitting a big file into several files.

```
(DEFINEQ
```

(**SPLITFILE**
```
   [LAMBDA (FILE SPLIT-SIZE)                                        ; Edited 26-Jan-93 20:46 by jds
      (CL:WITH-OPEN-STREAM (INSTR (OPENSTREAM FILE 'INPUT 'OLD))
           (for I from 1 as START from 0 by SPLIT-SIZE while (ILESSP START (GETEOFPTR INSTR))
              do (CL:WITH-OPEN-STREAM (OUTSTR (OPENSTREAM (PACKFILENAME 'VERSION NIL 'NAME
                                                    (CONCAT (UNPACKFILENAME.STRING FILE
                                                                  'NAME)
                                                         I)
                                                    'BODY FILE)
                                                 'OUTPUT
                                                 'NEW))
                       (COPYBYTES INSTR OUTSTR START (IMIN (GETEOFPTR INSTR)
                                                           (+ START SPLIT-SIZE])
)
```

;; For making DOS file systems

```
(DEFINEQ
```

(**DOSLINKER**
```
   [LAMBDA (FILES OLDDIR STREAM)                                   ; Edited 23-Mar-93 02:38 by jds
      (for FILE in FILES do (PRINTOUT STREAM "ln -s " OLDDIR FILE "  " (L-CASE (SHORTEN (UNPACKFILENAME.STRING
                                                                                          FILE
                                                                                          'NAME)
                                                                                    8))
                              "."
                              (L-CASE (SHORTEN (UNPACKFILENAME.STRING FILE 'EXTENSION)
                                            3))
                              T])
```

(**SHORTEN**
```
   [LAMBDA (STRING LEN)
      (SUBSTRING STRING 1 (IMIN LEN (NCHARS STRING])
)

(DECLARE%: EVAL@COMPILE

(I.S.OPR 'INFILES NIL '[SUBST (GENSYM)
                              'GENVAR
                              '(BIND GENVAR _ (\GENERATEFILES BODY NIL '(SORT))
                                  EACHTIME (PROGN (OR (SETQ I.V. (\GENERATENEXTFILE GENVAR))
                                                      (GO $$OUT))
                                                (IF (LISTP I.V.)
                                                    THEN (SETQ I.V. (CONCATCODES I.V.]
      T)
)

(PUTPROPS COPYFILES COPYRIGHT ("Venue & Xerox Corporation" 1989 1990 1991 1993 2018))
```

FUNCTION INDEX

I.S.OPR INDEX