File created: 18-Jul-2023 23:40:13 **{WMEDLEY}<sources>FILEPKG.;50**

edit by: rmk

changes to: (FNS GATHEREXPORTS)

previous date: 13-Jul-2023 14:56:21 {WMEDLEY}<sources>FILEPKG.;48

Read Table: INTERLISP

Package: INTERLISP

Format: XCCS

```
(RPAQQ FILEPKGCOMS
       [(COMS                                                         ; standard records for accessing file  package type/command
                                                                      ; parts.  Exported for PRETTY
              (VARS FILEPKGTYPEPROPS)
              (EXPORT (DECLARE%: EVAL@COMPILE DONTCOPY (RECORDS * FILEPKGRECORDS)))
              (FNS SEARCHPRETTYTYPELST PRETTYDEFMACROS FILEPKGCOMPROPS)
              (INITRECORDS * FILEPKGRECORDS))
        [DECLARE%: EVAL@COMPILE DOCOPY

               ;; Proclaim SPECIAL those variables that are used freely in a lot of code.

               (P (CL:PROCLAIM '(CL:SPECIAL PRETTYDEFMACROS PRETTYTYPELST FILEPKGTYPES PRETTYPRINTMACROS
                                       *DEFAULT-CLEANUP-COMPILER* MARKASCHANGEDFNS PRETTYFLG FROMEDITOR))
                  (CL:PROCLAIM '(GLOBAL FILELST SYSFILES LOADEDFILELST NOTLISTEDFILES NOTCOMPILEDFILES
                                       MAKEFILEFORMS CLEANUPOPTIONS]
        (INITVARS (MSDATABASELST))
        [COMS ;; making, adding, listing, compiling files

              (FNS CLEANUP COMPILEFILES COMPILEFILES0 CONTINUEDIT MAKEFILE FILECHANGES FILEPKG.MERGECHANGES
                   FILEPKG.CHANGEDFNS MAKEFILE1 COMPILE-FILE? MAKEFILES ADDFILE ADDFILE0 LISTFILES)
              (INITVARS (*DEFAULT-CLEANUP-COMPILER* 'CL:COMPILE-FILE)
                     (FILELST)
                     (LOADEDFILELST)
                     (NOTLISTEDFILES)
                     (NOTCOMPILEDFILES)
                     (MAKEFILEFORMS)
                     (NILCOMS))
              (ADDVARS (MAKEFILEOPTIONS RC C LIST FAST CLISP CLISPIFY NIL REMAKE NEW NOCLISP CLISP%  F ST STF
                              FORMAT (REC . RC)
                              (BREC . RC)
                              (TC . C)
                              (BC . C)
                              (TCOMPL . C)
                              (BCOMPL . C)))
              (INITVARS (MAKEFILEREMAKEFLG T)
                     (CLEANUPOPTIONS '(RC]
        (COMS ;; scanning file coms

              (FNS FILEPKGCHANGES GETFILEPKGTYPE MARKASCHANGED FILECOMS WHEREIS SMASHFILECOMS FILEFNSLST
                   FILECOMSLST UPDATEFILES INFILECOMS? INFILECOMTAIL INFILECOMS INFILECOM INFILECOMSVALS
                   INFILECOMSVAL INFILECOMSPROP IFCPROPS IFCEXPRTYPE IFCPROPSCAN IFCDECLARE INFILEPAIRS
                   INFILECOMSMACRO))
        (COMS ;; adding to a file

              (FNS FILES? FILES?1 FILES?PRINTLST ADDTOFILES? ADDTOFILE WHATIS ADDTOCOMS ADDTOCOM ADDTOCOM1
                   ADDNEWCOM MAKENEWCOM DEFAULTMAKENEWCOM)
              (INITVARS (DEFAULTCOMHASFILEFLG))
              (ADDVARS (MARKASCHANGEDFNS))
              (FNS MERGEINSERT MERGEINSERT1)

              ;; RMK: Changed INITVARS to VARS, so = addition is a synonym for untypable LF, and also suppress appearance of raw CR and LF
              ;; in the file

              (FNS ADDTOFILEKEYLST)
              (INITVARS (ADDTOFILEKEYLST (ADDTOFILEKEYLST))
                     (LASTFILE)))
        (COMS ;; deleting an item from a file

              (FNS DELFROMFILES DELFROMCOMS DELFROMCOM DELFROMCOM1 REMOVEITEM MOVETOFILE)
              (P (MOVD? 'DELFROMFILES 'DELFROMFILE NIL T)
                 (MOVD? 'MOVETOFILE 'MOVEITEM NIL T))
              (ADDVARS (SYSPROPS PROPTYPE VARTYPE)))
        [COMS                                                         ; functions for doing things and marking them changed and
                                                                      ; auxiliary functions
              (FNS SAVEPUT)
              [DECLARE%: DONTEVAL@LOAD DOCOPY (P (OR (CHANGENAME 'PUTPROPS 'PUTPROP 'SAVEPUT)
                                                     (CHANGENAME 'PUTPROPS '/PUT 'SAVEPUT]
              (FNS UNMARKASCHANGED PREEDITFN POSTEDITPROPS POSTEDITALISTS)
              (ADDVARS (LISPXFNS (PUT . SAVEPUT)
                              (PUTPROP . SAVEPUT]
        (COMS                                                         ; sub-functions for file package commands & types
      (FNS ALISTS.GETDEF ALISTS.WHENCHANGED CLEARCLISPARRAY EXPRESSIONS.WHENCHANGED MAKEALISTCOMS
           MAKEFILESCOMS MAKELISPXMACROSCOMS MAKEPROPSCOMS MAKEUSERMACROSCOMS PROPS.WHENCHANGED
           FILEGETDEF.LISPXMACROS FILEGETDEF.ALISTS FILEGETDEF.RECORDS FILEGETDEF.PROPS
           FILEGETDEF.MACROS FILEGETDEF.VARS FILEGETDEF.FNS FILEPKGCOMS.PUTDEF FILES.PUTDEF VARS.PUTDEF
           FILES.WHENCHANGED)
      (ADDVARS (MACROPROPS MACRO BYTEMACRO DMACRO)
             (SYSPROPS PROPTYPE))
      (PROP PROPTYPE I.S.OPR SUBR LIST CODE FILEDATES FILE FILEMAP EXPR VALUE COPYRIGHT FILETYPE)
      (PROP VARTYPE BAKTRACELST BREAKMACROS COMPILETYPELST EDITMACROS ERRORTYPELST FONTDEFS
            LISPXHISTORYMACROS LISPXMACROS PRETTYDEFMACROS PRETTYEQUIVLST PRETTYPRINTMACROS
            PRETTYPRINTYPEMACROS USERMACROS))
(COMS                                                    ; Define the commands below AFTER the various properties
                                                         ; have been established.
      (USERMACROS M))
(COMS                                                    ; GETDEF methods
      (FNS RENAME CHANGECALLERS)
      (FNS SHOWDEF COPYDEF GETDEF GETDEFCOM GETDEFCOM0 GETDEFCURRENT GETDEFERR GETDEFFROMFILE
           GETDEFSAVED PUTDEF EDITDEF DEFAULT.EDITDEF EDITDEF.FILES LOADDEF DWIMDEF DELDEF DELFROMLIST
           HASDEF GETFILEDEF SAVEDEF UNSAVEDEF COMPAREDEFS COMPARE TYPESOF)
      (INITVARS (WHEREIS.HASH)))
(COMS ;; how to dump FILEPKGCOMS. The SPLST must be initialized to contain FILEPKGCOMS in order to get started.

      (FNS FILEPKGCOM FILEPKGTYPE)
      (PROP ARGNAMES FILEPKGCOM)
      (ADDVARS (FILEPKGCOMSPLST FILEPKGCOMS)
             (FILEPKGTYPES FILEPKGCOMS))
      (FILEPKGCOMS FILEPKGCOMS)
      (FILEPKGCOMS ALISTS DEFS EDITMACROS EXPRESSIONS FIELDS FILEPKGTYPES FILES FILEVARS FNS INITRECORDS
             INITVARS LISPXCOMS LISPXMACROS MACROS PRETTYDEFMACROS PROPS RECORDS OLDRECORDS SYSRECORDS
             USERMACROS VARS * CONSTANTS))
(ADDVARS (SHADOW-TYPES (FUNCTIONS FNS)
                (VARIABLES VARS CONSTANTS)))
(INITVARS (SAVEDDEFS))
(COMS                                                    ; EDITCALLERS
      (FNS FINDCALLERS EDITCALLERS EDITFROMFILE FINDATS LOOKIN)
      (FNS SEPRCASE)
      [INITVARS (DEFAULTRENAMEMETHOD '(EDITCALLERS CAREFUL]
      (INITVARS (SEPRCASEARRAYS)
             (CLISPCASEARRAYS))
      (P (MOVD? 'INFILEP 'FINDFILE)
                                                         ; or else from SPELLFILE
         )
      (BLOCKS (EDITFROMFILE EDITFROMFILE LOOKIN (GLOBALVARS EDITLOADFNSFLG)
                   (NOLINKFNS LOADFROM)))
      (GLOBALVARS SYSFILES CLISPCASEARRAYS SEPRCASEARRAYS CLISPCHARS))
(COMS                                                    ; EXPORT
      (FNS IMPORTFILE IMPORTEVAL IMPORTFILESCAN CHECKIMPORTS GATHEREXPORTS \DUMPEXPORTS)
      (FILEPKGCOMS EXPORT)
      [INITVARS (BEGINEXPORTDEFSTRING "* %"FOLLOWING DEFINITIONS EXPORTED%")")
             (ENDEXPORTDEFFORM '(* "END EXPORTED DEFINITIONS"]
      (GLOBALVARS BEGINEXPORTDEFSTRING ENDEXPORTDEFFORM))
(COMS                                                    ; for GAINSPACE
      (FNS CLEARFILEPKG)
      [ADDVARS (GAINSPACEFORMS (FILELST "erase filepkg information" (CLEARFILEPKG RESPONSE)
                                      ((Y "es")
                                       (N "o")
                                       (E . "verything")
                                       (F "ilemaps only
                                          "]
      (GLOBALVARS SMASHPROPSLST1))
(GLOBALVARS %#UNDOSAVES ADDTOFILEKEYLST CLEANUPOPTIONS CLISPIFYPRETTYFLG COMPILE.EXT DECLARETAGSLST
       DEFAULTRENAMEMETHOD FILEPKGCOMSPLST FILERDTBL HISTORYCOMS HISTSTR0 I.S.OPRLST LISPXCOMS
       LISPXHISTORY LISPXHISTORYMACROS LISPXMACROS MACROPROPS MAKEFILEOPTIONS MAKEFILEREMAKEFLG
       MSDATABASELST PRETTYHEADER PRETTYTRANFLG SAVEDDEFS SYSPROPS USERMACROS USERRECLST USERWORDS
       FILEPKGTYPEPROPS)
(BLOCKS (DELFROMCOMS DELFROMCOMS DELFROMCOM DELFROMCOM1 REMOVEITEM (NOLINKFNS . T)
               (SPECVARS COMSNAME))
       (ADDTOFILEBLOCK ADDTOFILES? ADDTOFILE WHATIS ADDTOCOMS ADDTOCOM ADDTOCOM1 ADDNEWCOM (NOLINKFNS
                                                                                                      . T)
              (SPECVARS COMSNAME)
              (ENTRIES ADDTOFILE ADDTOCOMS ADDTOFILES? ADDTOCOM1))
       (INFILECOMS? INFILECOMS? INFILECOMTAIL INFILECOMS INFILECOM INFILECOMSVAL INFILECOMSVALS
              INFILEPAIRS INFILECOMSMACRO IFCPROPS IFCEXPRTYPE IFCPROPSCAN IFCDECLARE
              (LOCALFREEVARS NAME LITERALS VAL TYPE ONFILETYPE ORIGFLG)
              INFILECOMSPROP)
       (NIL MAKEFILE (LOCALVARS . T)
            (SPECVARS FILE OPTIONS REPRINTFNS SOURCEFILE FILETYPE FILEDATES CHANGES))
       (ADDFILE ADDFILE ADDFILE0)
       (FILEPKGCHANGES FILEPKGCHANGES (NOLINKFNS . T))
       (NIL ALISTS.WHENCHANGED CHANGECALLERS CLEANUP CLEARCLISPARRAY COMPARE COMPAREDEFS COMPILEFILES
            COMPILEFILES0 CONTINUEDIT COPYDEF DEFAULTMAKENEWCOM DELFROMFILES EDITDEF
            EXPRESSIONS.WHENCHANGED FILECOMS FILECOMSLST FILEFNSLST FILEPKGCOM FILEPKGCOMPROPS
            FILEPKGTYPE FILES? FILES?1 GETFILEPKGTYPE HASDEF INFILECOMTAIL LOADDEF MAKEALISTCOMS
            MAKEFILE1 MAKEFILES MAKEFILESCOMS MAKELISPXMACROSCOMS MAKENEWCOM MAKEPROPSCOMS
            MAKEUSERMACROSCOMS MARKASCHANGED MOVETOFILE POSTEDITPROPS PREEDITFN PRETTYDEFMACROS
            PROPS.WHENCHANGED PUTDEF RENAME SAVEDEF SAVEPUT SEARCHPRETTYTYPELST SHOWDEF SMASHFILECOMS
            TYPESOF UNMARKASCHANGED UNSAVEDEF UPDATEFILES (GLOBALVARS %#UNDOSAVES SYSFILES
                                                                 MARKASCHANGEDSTATS COMPILE.EXT
                                                                 EDITMACROS EDITLOADFNSFLG LOADOPTIONS)
            (LOCALVARS . T))
       (DELDEF DELDEF DELFROMLIST (NOLINKFNS . T))
       (GETDEF GETDEF DWIMDEF GETDEFCOM GETDEFCOM0 GETDEFERR GETDEFCURRENT GETDEFFROMFILE GETDEFSAVED
              (RETFNS GETDEFCOM)
              (NOLINKFNS . T)
              (GLOBALVARS NOT-FOUNDTAG)))
(DECLARE%: DONTEVAL@LOAD DOEVAL@COMPILE DONTCOPY COMPILERVARS
       (ADDVARS (NLAMA \DUMPEXPORTS MAKEUSERMACROSCOMS MAKEPROPSCOMS MAKELISPXMACROSCOMS MAKEFILESCOMS
                       MAKEALISTCOMS LISTFILES COMPILEFILES CLEANUP FILEPKGCOMPROPS PRETTYDEFMACROS)
              (NLAML)
              (LAMA FILEPKGTYPE FILEPKGCOM FILEPKGCHANGES])
```

;; standard records for accessing file package type/command parts. Exported for PRETTY

```
(RPAQQ FILEPKGTYPEPROPS (NEWCOM WHENFILED WHENUNFILED GETDEF NULLDEF DELDEF PUTDEF WHENCHANGED HASDEF EDITDEF
                                CANFILEDEF FILEGETDEF))
```

;; FOLLOWING DEFINITIONS EXPORTED

```
(DECLARE%: EVAL@COMPILE DONTCOPY

(RPAQQ FILEPKGRECORDS (FILEPKGCOM FILEPKGTYPE FILE FILEDATEPAIR FILEPROP))

(DECLARE%: EVAL@COMPILE

(ACCESSFNS FILEPKGCOM [[ADD (GETPROP DATUM 'ADDTOPRETTYCOM)
                            (UNDOABLE (COND
                                         (NEWVALUE (/PUTPROP DATUM 'ADDTOPRETTYCOM NEWVALUE))
                                         (T (/REMPROP DATUM 'ADDTOPRETTYCOM]
                       [DELETE (GETPROP DATUM 'DELFROMPRETTYCOM)
                              (UNDOABLE (COND
                                           (NEWVALUE (/PUTPROP DATUM 'DELFROMPRETTYCOM NEWVALUE))
                                           (T (/REMPROP DATUM 'DELFROMPRETTYCOM]
                       [PRETTYTYPE (GETPROP DATUM 'PRETTYTYPE)
                              (UNDOABLE (COND
                                           (NEWVALUE (/PUTPROP DATUM 'PRETTYTYPE NEWVALUE))
                                           (T (/REMPROP DATUM 'PRETTYTYPE]
                       [CONTENTS (GETPROP DATUM 'FILEPKGCONTENTS)
                              (UNDOABLE (COND
                                           (NEWVALUE (/PUTPROP DATUM 'FILEPKGCONTENTS NEWVALUE))
                                           (T (/REMPROP DATUM 'FILEPKGCONTENTS]
                       (MACRO [CDR (FASSOC DATUM (GETTOPVAL 'PRETTYDEFMACROS]
                              (STANDARD [COND
                                           [NEWVALUE (PUTASSOC DATUM NEWVALUE (OR (LISTP (GETTOPVAL
                                                                                                'PRETTYDEFMACROS
                                                                                                ))
                                                                                  (SETTOPVAL 'PRETTYDEFMACROS
                                                                                         (LIST (LIST DATUM]
                                           (T (SETTOPVAL 'PRETTYDEFMACROS (REMOVE (FASSOC DATUM
                                                                                         (GETTOPVAL
                                                                                                'PRETTYDEFMACROS
                                                                                                ))
                                                                                 (GETTOPVAL 'PRETTYDEFMACROS]
                                     UNDOABLE
                                     (COND
                                        [NEWVALUE (/PUTASSOC DATUM NEWVALUE (OR (LISTP (GETTOPVAL
                                                                                              'PRETTYDEFMACROS))
                                                                                (/SETTOPVAL 'PRETTYDEFMACROS
                                                                                       (LIST (LIST DATUM]
                                        (T (/SETTOPVAL 'PRETTYDEFMACROS (REMOVE (FASSOC DATUM
                                                                                       (GETTOPVAL
                                                                                              'PRETTYDEFMACROS))
                                                                               (GETTOPVAL 'PRETTYDEFMACROS]
                                                                (* Not an atom record cause want REMPROP on NILs.)
                                                                (* NOTE%: PRETTCOM on PRETTY has open-coded access to
      the MACRO property.)
       (INIT (FILEPKGCOMPROPS ADDTOPRETTYCOM DELFROMPRETTYCOM PRETTYTYPE FILEPKGCONTENTS)))

[ATOMRECORD FILEPKGTYPE (NEWCOM WHENFILED WHENUNFILED GETDEF NULLDEF DELDEF PUTDEF WHENCHANGED HASDEF EDITDEF
                               FILEGETDEF CANFILEDEF)
       (ACCESSFNS FILEPKGTYPE [(CHANGEDLST (CAR (SEARCHPRETTYTYPELST DATUM))
                                      (CAR (SEARCHPRETTYTYPELST DATUM NEWVALUE)))
                               (CHANGED (GETTOPVAL (CAR (SEARCHPRETTYTYPELST DATUM)))
                                      (STANDARD (SETTOPVAL (CAR (SEARCHPRETTYTYPELST DATUM NEWVALUE))
                                                       NEWVALUE)
                                             UNDOABLE
                                             (/SETTOPVAL (CAR (SEARCHPRETTYTYPELST DATUM NEWVALUE))
                                                    NEWVALUE)))
                               (DESCRIPTION (CAR (CDDR (SEARCHPRETTYTYPELST DATUM)))
                                      (CAR (RPLACA (CDDR (SEARCHPRETTYTYPELST DATUM NEWVALUE))
                                                  NEWVALUE)))
                               (ALLFIELDS NIL (/SETTOPVAL 'PRETTYTYPELST (REMOVE (SEARCHPRETTYTYPELST DATUM)
                                                                                (GETTOPVAL 'PRETTYTYPELST]
                                                                (* NOTE%: PRETTYCOM on PRETTY has open-coded access
             to GETDEF property)
```

```
              (INIT [PROGN (SETQ SYSPROPS (UNION FILEPKGTYPEPROPS SYSPROPS))
                           (MAPC FILEPKGTYPEPROPS (FUNCTION (LAMBDA (X)
                                                               (PUT X 'PROPTYPE 'FILEPKGCOMS]
                    (ADDTOVAR PRETTYTYPELST )]

[ATOMRECORD FILE (FILECHANGES FILEDATES FILEMAP)
       (ACCESSFNS FILE ((FILEPROP (GETPROP DATUM 'FILE)
                               (STANDARD (PUTPROP DATUM 'FILE NEWVALUE)
                                      UNDOABLE
                                      (/PUTPROP DATUM 'FILE NEWVALUE]

(RECORD FILEDATEPAIR (FILEDATE . DATEFILENAME))

(RECORD FILEPROP ((COMSNAME . LOADTYPE) . TOBEDUMPED))
)

(FILEPKGCOMPROPS ADDTOPRETTYCOM DELFROMPRETTYCOM PRETTYTYPE FILEPKGCONTENTS)

[PROGN (SETQ SYSPROPS (UNION FILEPKGTYPEPROPS SYSPROPS))
       (MAPC FILEPKGTYPEPROPS (FUNCTION (LAMBDA (X)
                                          (PUT X 'PROPTYPE 'FILEPKGCOMS]

(ADDTOVAR PRETTYTYPELST )
)

;; END EXPORTED DEFINITIONS

(DEFINEQ

(SEARCHPRETTYTYPELST
  [LAMBDA (TYPE FLG)                                             (* rmk%: " 3-JAN-82 22:55")
                                                                 ; access functions used by the records

    (AND (LITATOM TYPE)
         (OR (find X in PRETTYTYPELST suchthat (EQ (CADR X)
                                                 TYPE))
             (COND
                (FLG [/SETTOPVAL 'PRETTYTYPELST (CONS (SETQ FLG (LIST (PACK* 'CHANGED TYPE 'LST)
                                                                      TYPE NIL))
                                                       (GETTOPVAL 'PRETTYTYPELST]
                     (OR (LISTP (GETTOPVAL (CAR FLG)))
                         (/SETTOPVAL (CAR FLG)
                                NIL))
                     FLG])

(PRETTYDEFMACROS
  [NLAMBDA ARGS                                                  (* lmm " 5-SEP-78 16:16")
                                                                 ; included so that old files will continue to load

    (for X in ARGS collect (FILEPKGCOM (CAR X)
                               'MACRO
                               (CDR X])

(FILEPKGCOMPROPS
  [NLAMBDA PROPS
    (MAPC PROPS (FUNCTION (LAMBDA (Y)
                            (OR (MEMB Y SYSPROPS)
                                (SETQ SYSPROPS (CONS Y SYSPROPS)))
                            (PUT Y 'PROPTYPE 'FILEPKGCOMS])
)

(RPAQQ FILEPKGRECORDS (FILEPKGCOM FILEPKGTYPE FILE FILEDATEPAIR FILEPROP))

(FILEPKGCOMPROPS ADDTOPRETTYCOM DELFROMPRETTYCOM PRETTYTYPE FILEPKGCONTENTS)

[PROGN (SETQ SYSPROPS (UNION FILEPKGTYPEPROPS SYSPROPS))
       (MAPC FILEPKGTYPEPROPS (FUNCTION (LAMBDA (X)
                                          (PUT X 'PROPTYPE 'FILEPKGCOMS]

(ADDTOVAR PRETTYTYPELST )

(DECLARE%: EVAL@COMPILE DOCOPY

(CL:PROCLAIM '(CL:SPECIAL PRETTYDEFMACROS PRETTYTYPELST FILEPKGTYPES PRETTYPRINTMACROS
                     *DEFAULT-CLEANUP-COMPILER* MARKASCHANGEDFNS PRETTYFLG FROMEDITOR))

(CL:PROCLAIM '(GLOBAL FILELST SYSFILES LOADEDFILELST NOTLISTEDFILES NOTCOMPILEDFILES MAKEFILEFORMS
                     CLEANUPOPTIONS))
)

(RPAQ? MSDATABASELST )

;; making, adding, listing, compiling files

(DEFINEQ
(CLEANUP
  [NLAMBDA FILES                                                     (* lmm "14-Aug-84 19:17")
    (PROG (TEM1 TEM2 OPTIONS)
           (COND
              ([LISTP (CAR (SETQ FILES (NLAMBDA.ARGS FILES]
               (SETQ OPTIONS (CAR FILES))
               (SETQ FILES (CDR FILES)))
              (T (SETQ OPTIONS CLEANUPOPTIONS)))
           (RETURN (APPEND (MAKEFILES OPTIONS FILES)
                           (COND
                              ((NOT (MEMB 'LIST OPTIONS))
                               NIL)
                              ((NULL FILES)
                               (LISTFILES))
                              ((SETQ TEM1 (INTERSECTION FILES NOTLISTEDFILES))
                                                                     ; Intersection check because LISTFILES applied to NIL means
                                                                     ; list all of NOTLISTEDFILES.
                               (APPLY 'LISTFILES TEM1)))
                           (COND
                              [(NULL (SETQ TEM1 (MEMB 'RC OPTIONS]
                              ((NULL FILES)
                               (COMPILEFILES0 (SETQ TEM2 NOTCOMPILEDFILES)
                                      (CDR TEM1))
                               TEM2)
                              ((SETQ TEM2 (INTERSECTION FILES NOTCOMPILEDFILES))
                               (COMPILEFILES0 TEM2 (CDR TEM1))
                               TEM2])

(COMPILEFILES
  [NLAMBDA FILES                                                     (* lmm "14-Aug-84 19:17")
    (COND
       ([LISTP (CAR (SETQ FILES (NLAMBDA.ARGS FILES]
        (COMPILEFILES0 (CDR FILES)
               (CAR FILES)))
       (T (COMPILEFILES0 FILES])

(COMPILEFILES0
  [LAMBDA (FILES OPTIONS)                                            (* rmk%: "19-FEB-83 21:59")
    (for X OPTS (RCFLG _ T) on (OR FILES NOTCOMPILEDFILES) first (SETQ OPTS (SELECTQ (CAR (LISTP OPTIONS))
                                                                                    (C (SETQ RCFLG NIL)
                                                                                       (CDR OPTIONS))
                                                                                    (RC (CDR OPTIONS))
                                                                                   OPTIONS))
       do (MAKEFILE1 (OR (MISSPELLED? (CAR X)
                                    70 FILELST NIL X)
                           (CAR X))
                  RCFLG OPTS X])

(CONTINUEDIT
  [LAMBDA (FILE)                                                     (* bvm%: "30-Aug-86 15:09")
    (PROG (STREAM FL TEM FC ENV)
           (RESETLST
               [RESETSAVE NIL (LIST 'CLOSEF (SETQ STREAM (OPENSTREAM FILE 'INPUT]
               (SETQ FILE (FULLNAME STREAM))
               (SETFILEPTR STREAM 0)
               (CL:MULTIPLE-VALUE-SETQ (ENV FC)
                      (\PARSE-FILE-HEADER STREAM 'RETURN)))
           (COND
              ([NOT (fetch FILEPROP of (SETQ FL (ROOTFILENAME FILE]
               (LOADFROM FILE)                                       ; also calls addfile to notice the file.

               ))
           (/replace FILECHANGES of FL with (FILECHANGES FC))
           [/replace FILEDATES of FL with (LIST (create FILEDATEPAIR
                                                       FILEDATE _ (CADR FC)
                                                       DATEFILENAME _ FILE)
                                                (create FILEDATEPAIR
                                                       FILEDATE _ [CAR (SETQ TEM (CDR (MEMB 'date%: FC]
                                                       DATEFILENAME _ (CADR TEM]
           (RETURN FILE])

(MAKEFILE
  [LAMBDA (FILE OPTIONS REPRINTFNS SOURCEFILE)                       ; Edited 10-Oct-2021 20:36 by rmk:
                                                                     ; Edited 29-Jun-2021 17:24 by rmk:

    ;; OPTIONS: FAST means dump with PRETTYFLG set to NIL;  LIST means list the FILE;  RC means RECOMPILE, C means COMPILEL;  --- for C
    ;; AND RC assume ST unless next option is F.

    ;; RMK:  OPTIONS can specify external format, either as a pair like (FORMAT :UTF-8) or just :UTF-8 where (FIND-FORMAT :UTF-8) is non NIL.

    [SETQ OPTIONS (for OPT inside OPTIONS collect (CL:IF (FIND-FORMAT OPT T)
                                                         (LIST 'FORMAT OPT))]
    (PROG ((PRETTYFLG (AND [NOT (MEMB 'FAST (SETQ OPTIONS (MKLIST OPTIONS]
                             PRETTYFLG))
```

```
             (*PRINT-BASE* (if (EQ *PRINT-BASE* 8)
                                 then 8
                               else                                       ; make sure radix is either 8 or 10, because all others don’t read
                                                                          ; in like they print.  Maybe obsolete now with makefile
                                                                          ; environments
                                    10))
             FILETYPE ROOTNAME FILEPROP CHANGES FILEDATES (Z (ADDFILE FILE)))
            (DECLARE (CL:SPECIAL PRETTYFLG))
            (SETQ FILE (CAR Z))                                           ; Necessary because FILE might have been misspelled.
            (SETQ ROOTNAME (CADR Z))                                      ; result of (ROOTFILENAME FILE), or if FILE is corrected, result
                                                                          ; of applying ROOTFILENAME to correct value.

            (SETQ FILEPROP (CDDR Z))
            (UPDATEFILES)                                                 ; Want updating done after file is added to filelst, so any
                                                                          ; functions that are being dumped are marked as having been
                                                                          ; dumped.

            (SETQ CHANGES (fetch TOBEDUMPED of FILEPROP))
            (SETQ FILEDATES (LISTP (fetch FILEDATES of ROOTNAME)))
            (SETQ FILETYPE (GETPROP ROOTNAME ’FILETYPE))
       LP0  (if (AND (NULL (fetch LOADTYPE of FILEPROP))
                     (NULL FILEDATES))
                then                                                      ; File has never been loaded and never dumped i.e. user just set
                                                                          ; up COMS in core
              elseif [OR (EQMEMB ’NEW OPTIONS)
                         (AND (NULL MAKEFILEREMAKEFLG)
                              (NOT (MEMB ’REMAKE OPTIONS]
                then (COND
                         ((AND (fetch LOADTYPE of FILEPROP)
                                (NEQ T (fetch LOADTYPE of FILEPROP)))
                          (LISPXPRIN2 FILE T T)
                          (LISPXPRIN1 (SELECTQ (fetch LOADTYPE of FILEPROP)
                                             (LOADCOMP "the file was loaded for compilation purposes only")
                                             ((compiled Compiled COMPILED)
                                                  " -- only the compiled file has been loaded
                                                  ")
                                             ((loadfns LOADFNS)
                                                  " -- only some of its symbolics have been loaded
                                                  ")
                                             (SHOULDNT))
                                  T)
                          (COND
                              ((NEQ (ASKUSER DWIMWAIT ’Y "Go ahead and MAKEFILE anyway? ")
                                    ’Y)                                   ; E.g. user loads a .com file and then resets the COMS or
                                                                          ; defines the functons by hand.
                               (GO OUT)))
                          (/replace LOADTYPE of FILEPROP with NIL)))
                      (SETQ SOURCEFILE NIL)
                      (SETQ REPRINTFNS NIL)
              elseif SOURCEFILE
                then                                                      ; source file given
              elseif [AND FILEDATES (OR [AND (SETQ SOURCEFILE (FINDFILE ROOTNAME T))
                                                (EQUAL (FILEDATE SOURCEFILE)
                                                        (fetch FILEDATE of (CAR FILEDATES]
                                          (AND [NOT (STRING-EQUAL SOURCEFILE (SETQ SOURCEFILE (fetch DATEFILENAME
                                                                                                         of (CAR FILEDATES]
                                                (INFILEP SOURCEFILE)
                                                (EQUAL (FILEDATE SOURCEFILE)
                                                        (fetch FILEDATE of (CAR FILEDATES]
                then (/replace DATEFILENAME of (CAR FILEDATES) with SOURCEFILE)
                      (OR REPRINTFNS (SETQ REPRINTFNS (FILEPKG.CHANGEDFNS CHANGES)))
              elseif [AND (CDR FILEDATES)
                          [SETQ SOURCEFILE (INFILEP (fetch DATEFILENAME of (CADR FILEDATES]
                          (EQUAL (FILEDATE SOURCEFILE)
                                  (fetch FILEDATE of (CADR FILEDATES]
                then ;; prevous version file is gone, drop back to original daddy file and dump everything that has been changed.
                      (SETQ CHANGES (FILEPKG.MERGECHANGES (fetch TOBEDUMPED of FILEPROP)
                                              (fetch FILECHANGES of ROOTNAME)))
                      (SETQ REPRINTFNS (FILEPKG.CHANGEDFNS CHANGES))
              else (LISPXPRIN1 ’"can’t find either the previous version or the original version of
                                  " T)
                   (LISPXPRIN2 FILE T T)
                   (LISPXPRIN1 ’", so it will have to be written anew
                                  " T)
                   (SETQ SOURCEFILE NIL)
                   (SETQ REPRINTFNS NIL)
                   (push OPTIONS ’NEW)
                   (SETQ CHANGES (fetch FILECHANGES of ROOTNAME))
                   (GO LP0))
            (COND
               ((AND SOURCEFILE (SETQ Z (SELECTQ (fetch LOADTYPE of FILEPROP)
                                                 (LOADCOMP                ; only loaded via LOADCOMP.  Need to do LOADFROM
                                                             (LIST ’N SOURCEFILE "was loaded with LOADCOMP" ’-
                                                                   "LOADFROM it to obtain VARS/COMS"))
                                                 (Compiled (AND (INFILECOMS? ’DONTCOPY ’DECLARE%:
                                                                           (fetch COMSNAME of FILEPROP))
                                                                  (LIST ’Y "only compiled version of" ROOTNAME "was
                                                                         loaded; LOADVARS the (DECLARE .. DONTCOPY )
```

```
                                                                 expressions")))
                                           ((compiled loadfns)
                                                (LIST 'N "Only some functions from" SOURCEFILE "loaded via
                                                      LOADFNS. Load all other expressions from it"))
                                           NIL)))
               (SELECTQ [ASKUSER DWIMWAIT (CAR Z)
                               (CDR Z)
                               '((Y "es
                                    ")
                                 (N "o
                                    ")
                                 (A "bort MAKEFILE
                                    "]
                   (Y (SELECTQ (fetch LOADTYPE of FILEPROP)
                          (LOADCOMP                                   ; file was never actually loaded, just loadcomped.  thus no
                                                                      ; filecoms
                                     (LOADFROM SOURCEFILE))
                          (Compiled
                                     ;; This is going to be a remake.  If it was originally loaded as a compiled file, must first do a
                                     ;; LOADFROM in order to get the properties set up by declare: etc.

                                     (LOADVARS 'DONTCOPY SOURCEFILE)
                                     (/replace LOADTYPE of FILEPROP with 'COMPILED)
                                                                      ; So wont have to be done again.

                                     ;; These are the only DECLARE:'s that are not also on the compiled file.  Note that a DECLARE:
                                     ;; DONTEVAL@LOAD will be found and evaluated, but the corresponding expressions won't be
                                     ;; evaluated from within the DECLARE:  Not worthwhile to bother setting up a complicated edit pattern
                                     ;; to screen these out, especially if you consider expressions like (DECLARE: -- DONTEVAL@LOAD --
                                     ;; DOEVAL@LOAD --)

                                     )
                          ((loadfns compiled)

                                           ;; This is going to be a remake, but the original call to LOADFNS didnt specify all the VARS, so
                                           ;; some expressions may not have been loaded.

                               (LOADVARS T SOURCEFILE))
                          NIL))
                   (A (GO OUT))
                   NIL)))
           (RESETLST
               [COND
                  ((MEMB 'NOCLISP OPTIONS)
                   (RESETSAVE PRETTYTRANFLG T))
                  ((MEMB 'CLISP%  OPTIONS)
                   (RESETSAVE PRETTYTRANFLG 'BOTH]
               (RESETSAVE %#UNDOSAVES)
               [COND
                  ((OR (MEMB 'CLISPIFY OPTIONS)
                       (MEMB 'CLISP OPTIONS))
                   (RESETSAVE CLISPIFYPRETTYFLG T))
                  ((OR (EQ FILETYPE 'CLISP)
                       (MEMB 'CLISP (LISTP FILETYPE)))
                   (RESETSAVE CLISPIFYPRETTYFLG 'CHANGES]
               (for X in MAKEFILEFORMS do (ERSETQ (EVAL X)))
               [SETQ FILE (PRETTYDEF NIL FILE (fetch COMSNAME of FILEPROP)
                                 REPRINTFNS SOURCEFILE CHANGES (CADR (ASSOC 'FORMAT OPTIONS])
           (SETQ LASTFILE ROOTNAME)
           (/replace TOBEDUMPED of FILEPROP with NIL)
           (COND
              ((NOT (EQMEMB 'DON'TLIST FILETYPE))
               (pushnew NOTLISTEDFILES ROOTNAME)))
           (COND
              ((NOT (EQMEMB 'DON'TCOMPILE FILETYPE))
               (pushnew NOTCOMPILEDFILES ROOTNAME)))
           [for TAIL OPT on OPTIONS do (SETQ OPT (CAR TAIL))
                                      (SELECTQ OPT
                                           (RC (AND (MEMB ROOTNAME NOTCOMPILEDFILES)
                                                    (MAKEFILE1 FILE T (CDR TAIL))))
                                           (C (AND (MEMB ROOTNAME NOTCOMPILEDFILES)
                                                   (MAKEFILE1 FILE NIL (CDR TAIL))))
                                           (LIST (AND (MEMB ROOTNAME NOTLISTEDFILES)
                                                      (APPLY 'LISTFILES (LIST FILE))))
                                           (COND
                                              ((MEMB (CL:IF (LISTP OPT)
                                                            (CAR OPT)
                                                            OPT)
                                                     MAKEFILEOPTIONS))
                                              ((FIXSPELL OPT NIL MAKEFILEOPTIONS NIL OPTIONS)
                                               (GO $$LP))
                                              (T (ERROR "Unrecognized MAKEFILE option" OPT]
           (RETURN FILE)
      OUT (RETURN (LIST FILE "-- MAKEFILE not performed."])
```

(FILECHANGES

```
  [LAMBDA (FILE TYPE)                                                 ; Edited  2-Mar-2022 15:43 by larry
                                                                      (* bvm%: "30-Aug-86 15:08")
```

```
    ;; If FILE is a list, it is assumed to be a file-created expressions;  otherwise, the filecreated expression is read from FILE.  If TYPE, returns the list of
    ;; changed items of that type from the changes expression.  If TYPE=NIL, returns the whole list of typed change-lists

    (PROG ([FCEXPR (OR (LISTP FILE)
                       (AND FILE (RESETLST
                                     (LET (OLDPTR STREAM)
                                          [if (SETQ STREAM (OPENP FILE 'INPUT))
                                              then (SETQ OLDPTR (GETFILEPTR STREAM))
                                                   (SETFILEPTR STREAM 0)
                                            else (RESETSAVE NIL (LIST 'CLOSEF (SETQ STREAM (OPENSTREAM
                                                                                           FILE
                                                                                           'INPUT]
                                          (CL:MULTIPLE-VALUE-BIND (ENV FC)
                                              (\PARSE-FILE-HEADER STREAM 'RETURN)
                                            (if OLDPTR
                                                then (SETFILEPTR STREAM OLDPTR))
                                            FC)))]
           FNS CHANGES FOUND)
          (while FCEXPR do [SELECTQ (CAR FCEXPR)
                               ((to%: :CHANGES-TO)
                                   (SETQ FOUND T))
                               ((previous :PREVIOUS-DATE)
                                   (RETURN))
                               (CL:WHEN FOUND
                                   (push CHANGES (CAR FCEXPR)))]
                          (SETQ FCEXPR (CDR FCEXPR)))
          [if (AND TYPE (NEQ TYPE 'FNS))
              then (RETURN (CDR (ASSOC TYPE CHANGES]
          (SETQ FNS (SUBSET CHANGES (FUNCTION LITATOM)))           ; Old style changes expression listed FNS by name and other
                                                                  ; things by type
          (RETURN (if TYPE
                      then                                        ; TYPE=FNS cause of test above.
                           (NCONC FNS (CDR (ASSOC 'FNS CHANGES)))
                    elseif FNS
                      then (CONS (CONS 'FNS FNS)
                                 (SUBSET CHANGES (FUNCTION LISTP)))
                    else CHANGES])

(FILEPKG.MERGECHANGES
  [LAMBDA (C1 C2)                                                 (* rmk%: "24-MAY-82 23:09")

    ;; Merges 2 changes lists into a single one.  Treat LITATOM's as FNS, to accomodate old-style format on files.

    (for E2 TEMP (VAL _ (for E1 in C1 when (CDR (LISTP E1)) collect (APPEND E1))) in C2
       do [COND
             ((SETQ TEMP (ASSOC (CAR E2)
                                VAL))
              (NCONC TEMP (for X in (CDR E2) unless (MEMBER X (CDR TEMP)) collect X)))
             (T (SETQ VAL (NCONC1 VAL (APPEND E2]
       finally (RETURN VAL])

(FILEPKG.CHANGEDFNS
  [LAMBDA (CHANGES)                                               (* rmk%: "20-MAY-82 22:00")

    ;; Returns list of function names from a file-changes list.  Interprets old format (functions are atoms) and new format (with explicit type headers)

    (CDR (ASSOC 'FNS CHANGES])

(MAKEFILE1
  [LAMBDA (FILE RECOMPFLG OPTIONS OTHERFILES)                     ; Edited 27-Oct-2020 15:40 by rmk:
                                                                  ; Edited 29-Aug-89 11:46 by bvm

    ;; RMK:  Call COMPILE-FILE? with FILE instead of (ROOTFILENAME FILE)

    (PROG ((ROOTNAME (ROOTFILENAME FILE))
           (COMPILER (COMPILE-FILE? FILE))
           GROUP)
          (COND
             ((AND (OR (EQ COMPILER 'BCOMPL)
                       (EQ COMPILER 'TCOMPL))
                   (NOT (FILEFNSLST ROOTNAME)))                   ; No FNS on this file, and we're told to use Interlisp compiler, so
                                                                  ; nothing to do.
              (/SETTOPVAL 'NOTCOMPILEDFILES (REMOVE ROOTNAME NOTCOMPILEDFILES))
              (RETURN NIL)))
          (COND
             ([find X in (SETQ GROUP (GETPROP ROOTNAME 'FILEGROUP))
                 suchthat (AND (NEQ X ROOTNAME)
                               (OR (fetch TOBEDUMPED of (fetch FILEPROP of X))
                                   (MEMB X OTHERFILES]

              ;; The file in question must be recompiled with other files, and one of the remaining files still needs to be dumped, or else one of the
              ;; other file is further down the list of files being compiled.  Wait.

              (RETURN)))
          (LISPXPRIN1 '"
                       compiling " T)
          (LISPXPRINT (OR GROUP FILE)
                 T T)
          (LISPXPRINT (LET [[REDEFINE? (OR (EQ (CAR OPTIONS)
                                                'ST)
                                            (EQ (CAR OPTIONS)
                                                'STF]
                             (FORGET-EXPRS? (EQ (CAR OPTIONS)
                                                'STF]
                            (SELECTQ COMPILER
                                ((FAKE-COMPILE-FILE)              ; The old CommonLispy interface to the ByteCompiler.
                                     (FAKE-COMPILE-FILE FILE :REDEFINE REDEFINE? :SAVE-EXPRS (AND REDEFINE?
                                                                                               (NOT
                                                                                                 FORGET-EXPRS?
                                                                                                    ))))
                                ((CL:COMPILE-FILE)                ; The new, improved (?) compiler
                                     (CL:COMPILE-FILE FILE :LOAD (COND
                                                                    ((AND REDEFINE? (NOT FORGET-EXPRS?))
                                                                     :SAVE)
                                                                    (REDEFINE? T)
                                                                    (T NIL))))
                                ((TCOMPL BCOMPL)                  ; The old ByteCompiler
                                     [if (MEMB (CAR OPTIONS)
                                              '(ST F S STF))
                                         then (LISPXUNREAD (LIST (CAR OPTIONS]
                                     [if GROUP
                                         then ;; File contained in FILEGROUP.  Therefore must be blockcompiled.

                                             (if RECOMPFLG
                                                 then (BRECOMPILE GROUP)
                                               else (BCOMPL GROUP))
                                       elseif (EQ COMPILER 'TCOMPL)
                                         then (if RECOMPFLG
                                                 then (RECOMPILE FILE)
                                               else (TCOMPL (LIST FILE)))
                                       else (if RECOMPFLG
                                               then (BRECOMPILE FILE)
                                             else (BCOMPL (LIST FILE])
                                (SHOULDNT "Non-existent compiler returned from COMPILE-FILE?...")))
                 T T])


(COMPILE-FILE?
  [LAMBDA (FILE)                                                  ; Edited 27-Oct-2020 15:39 by rmk:
                                                                  ; Edited 19-Jan-87 21:12 by Pavel

    ;; RMK: Argument is FILE instead of ROOTFILENAME, maybe more information

;;; Which compiler should CLEANUP use?

    (LET ((TYPE (GET (ROOTFILENAME FILE)
                     'FILETYPE))
          (UNKNOWN NIL))
         (for X inside TYPE do (SELECTQ X
                                  ((TCOMPL :TCOMPL)
                                       (RETURN 'TCOMPL))
                                  ((BCOMPL :BCOMPL)
                                       (RETURN 'BCOMPL))
                                  ((:FAKE-COMPILE-FILE CL:COMPILE-FILE COMPILE-FILE)
                                       (RETURN 'FAKE-COMPILE-FILE))
                                  ((:COMPILE-FILE :XCL-COMPILE-FILE)
                                       (RETURN 'CL:COMPILE-FILE))
                                  ((CLISP)
                                      NIL)
                                  (SETQ UNKNOWN T))
            finally (if UNKNOWN
                      then (CL:FORMAT T "~2%%**Warning: unknown FILETYPE value ~S~2%%" TYPE))
                  (RETURN *DEFAULT-CLEANUP-COMPILER*])


(MAKEFILES
  [LAMBDA (OPTIONS FILES)                                         (* rmk%: "23-FEB-83 21:20")
    (RESETVARS (%#UNDOSAVES)                                      ; Willing to save arbitrary amounts of undo info
               (UPDATEFILES)
               [COND
                  ((NULL FILES)
                   (for TYPE FLG in FILEPKGTYPES when [FILES?1 TYPE (COND
                                                                       ((NULL FLG)
                                                                  ; Gets printed the first time

                                                                        '"****NOTE: the following are not
                                                                         contained on any file:
                                                                             ")
                                                                       (T '"    "]
                      do (SETQ FLG T) finally (AND FLG (ADDTOFILES?]
               (SETQ OPTIONS (MKLIST OPTIONS))
               (RETURN (for FILE inside (OR FILES FILELST) when [fetch TOBEDUMPED of (LISTP (fetch FILEPROP
                                                                                             of (ROOTFILENAME
                                                                                                 FILE]
                          collect (LISPXPRIN2 FILE T T)
                                 (LISPXPRIN1 '|...| T)
                                 (PROG1 (MAKEFILE FILE OPTIONS)
                                        (LISPXTERPRI T])

(ADDFILE
  [LAMBDA (FILE LOADTYPE PRLST FCLST)                               (* bvm%: "29-Aug-86 12:22")
    ;; PRLST is the FILEPKGCHANGES prior to this file operation, FCLST is a list of file-created arguments, a singleton for a symbolic file, and a list
    ;; whose car represents the compiled file and whose cdr represent symbolic files compiled into it, for compiled files.

    (PROG ((ROOTNAME (ROOTFILENAME FILE))
           FLST VAL)
          [COND
             ((NOT FCLST)
              (SETQ VAL (ADDFILE0 ROOTNAME LOADTYPE FILE)))
             [(NULL (CDR FCLST))                                    ; A simple symbolic file
              (SETQ FCLST (CAR FCLST))
              (SETQ VAL (ADDFILE0 (COND
                                      ((LITATOM (CADR FCLST))
                                       (ROOTFILENAME (CADR FCLST)))
                                      (T ROOTNAME))
                               LOADTYPE FILE (CAR FCLST]
             (T ;; A compiled file, skip the first expression representing the compiled file itself, look at the cdr representing the symbolic files.

                (SELECTQ LOADTYPE
                     ((T LOADFNS)
                          (SETQ LOADTYPE 'Compiled))
                     (loadfns (SETQ LOADTYPE 'compiled))
                     (LOADCOMP                                      ; loadcomp on compiled file.  Don't notice since we don't know
                                                                    ; what its state is
                               NIL)
                     (SHOULDNT))
                (for X in (CDR FCLST) when (LITATOM (CADR X)) do (push FLST (CADR X))
                                                                  (OR (EQ LOADTYPE 'LOADCOMP)
                                                                      (ADDFILE0 (ROOTFILENAME (CADR X))
                                                                             LOADTYPE
                                                                             (CADR X)
                                                                             (CAR X]
          (UPDATEFILES PRLST (OR FLST (LIST FILE)))
          [AND LOADTYPE (for TYPE CHANGED in FILEPKGTYPES when (AND (LITATOM TYPE)
                                                                   (SETQ CHANGED (fetch CHANGED of TYPE)))
                           do (/replace CHANGED of TYPE with (INTERSECTION (CDR (ASSOC TYPE PRLST))
                                                                   CHANGED]
          (AND ADDSPELLFLG (ADDSPELL ROOTNAME USERWORDS))
          (RETURN VAL])
```

```
(ADDFILE0
  [LAMBDA (ROOTNAME LOADTYPE FULLNAME DAT)                          (* lmm "28-Nov-84 16:47")
    (PROG (COMS X FILEPROP FLG TEM)
       TOP (SETQ COMS (FILECOMS ROOTNAME))
          [COND
             ((SETQ FILEPROP (fetch FILEPROP of ROOTNAME))
              (COND
                 ([AND LOADTYPE (FMEMB LOADTYPE (CDR (FMEMB (fetch LOADTYPE of FILEPROP)
                                                            '(LOADCOMP loadfns compiled Compiled LOADFNS
                                                                      COMPILED NIL T]
                  (/replace LOADTYPE of FILEPROP with LOADTYPE)

                  ;; This call to ADDFILE reflects a 'higher' degree of loading, so upgrade property.  'loadfns' means just some information from
                  ;; file, if go to do makefile, must do loadfrom, 'compiled' is like 'loadfns' but for compiled files e.g. user does LOADFNS on
                  ;; compiled file.  'Compiled' means all but DECLARE: expressions are in.  e.g. user does LOAD of a compiled file.  COMPILED
                  ;; means everything is in, e.g. user does LOADDFROM a compiled file.  LOADFNS means everything in, e.g. user des
                  ;; LOADFROM symbolic file.  COMPILED and LOADFNS are equivalent in that means dont have to do any more loading when
                  ;; go to do a makefile but makefile NEW isnt permitted.  NIL is a makefile when coms were set up in core.  T is full load of
                  ;; symbolic file.  The check on TYPE=NIL is bcause dont want to upgrade as result of call from makefile, i.e. no new information
                  ;; there.

                  ;; LOADCOMP means file was loadcomp'ed.  note that the actual structure is a tree, not a list, and the above is only an
                  ;; approximation.  if you do a loadcomp, and then load the compiled file, the state will be left with latter, but then loadcomp? will
                  ;; loadcomp again because compiled files might not contain all the declare: EVAL@COMPILE expressions, e.g. macros, records
                  ;; etc.  however, in most cases, loadcomp is used independently of other loading, e.g. for compilation purposes only, so this will
                  ;; at least permit loadcomp? to work.

                  (GO OUT))
                 (T (GO OUT1]
          (COND
             [(OR LOADTYPE (LISTP (GETTOPVAL COMS)))
              (SETQ FILEPROP (/replace FILEPROP of ROOTNAME with (create FILEPROP
                                                                      COMSNAME _ COMS
                                                                      LOADTYPE _ LOADTYPE]
             (FLG (GO ERROR))
             ((AND DWIMFLG (EQ ROOTNAME FULLNAME)
                   (SETQ ROOTNAME (MISSPELLED? ROOTNAME 70 FILELST T)))

              ;; The EQ check is so as not to try correcting if the user has specified a version number or directory, as it is too messy trying to take
              ;; them out, and then put them back in on the corrected root name.

              (SETQ FULLNAME ROOTNAME)
              (SETQ FLG T)                                          ; so wont try to spelling correct again if file isnt there
              (GO TOP))
             (T (GO ERROR)))
```

```
      OUT [AND LOADTYPE DAT (/replace FILEDATES of ROOTNAME
                                with (LIST (create FILEDATEPAIR
                                                   FILEDATE _ DAT
                                                   DATEFILENAME _ FULLNAME]
          (AND (EQ LOADTYPE T)
               (/replace TOBEDUMPED of FILEPROP with NIL))
      OUT1
          [COND
             ([AND (LISTP (GETTOPVAL COMS))
                   (NOT (FMEMB ROOTNAME (GETTOPVAL 'FILELST]     ; coms wuld not be set up on a loadccomp.
              (/SETTOPVAL 'FILELST (CONS ROOTNAME (GETTOPVAL 'FILELST]
          (RETURN (COND
                     ((NULL LOADTYPE)                              ; call from makefile.
                      (CONS FULLNAME (CONS ROOTNAME FILEPROP)))
                     (T FILEPROP)))
      ERROR
          (ERROR FULLNAME "not file name." T])


(LISTFILES
  [NLAMBDA FILES                                                   (* rmk%: " 3-Dec-84 08:58")
    (DECLARE (GLOBALVARS NOTLISTEDFILES))                          ; LISTFILES1 is machinedependent
    (for FILE FULLNAME OPTIONS in (COND
                                     (FILES (SETQ FILES (NLAMBDA.ARGS FILES)))
                                     (T NOTLISTEDFILES))
       when (COND
               ((LISTP FILE)
                (SETQ OPTIONS (APPEND FILE OPTIONS))
                NIL)
               ((SETQ FULLNAME (FINDFILE FILE))
                FULLNAME)
               (T (printout T FILE " not found." T)
                  NIL))
       collect [COND
                  ((LISTFILES1 FULLNAME OPTIONS)
                   (SETQ NOTLISTEDFILES (REMOVE (NAMEFIELD FULLNAME T)
                                                NOTLISTEDFILES]
             FULLNAME])

)

(RPAQ? *DEFAULT-CLEANUP-COMPILER* 'CL:COMPILE-FILE)

(RPAQ? FILELST )

(RPAQ? LOADEDFILELST )

(RPAQ? NOTLISTEDFILES )

(RPAQ? NOTCOMPILEDFILES )

(RPAQ? MAKEFILEFORMS )

(RPAQ? NILCOMS )

(ADDTOVAR MAKEFILEOPTIONS RC C LIST FAST CLISP CLISPIFY NIL REMAKE NEW NOCLISP CLISP%  F ST STF FORMAT
                          (REC . RC)
                          (BREC . RC)
                          (TC . C)
                          (BC . C)
                          (TCOMPL . C)
                          (BCOMPL . C))

(RPAQ? MAKEFILEREMAKEFLG T)

(RPAQ? CLEANUPOPTIONS '(RC))

;; scanning file coms

(DEFINEQ

(FILEPKGCHANGES
  [LAMBDA N                                                        (* Pavel " 7-Oct-86 19:22")
    (COND
       [(EQ N 0)
        (PROG (TEM)
              (RETURN (for X in FILEPKGTYPES when (AND (LITATOM X)
                                                       (SETQ TEM (FILEPKGCHANGES X)))
                         collect (CONS X TEM]
       [(EQ (ARG N 1)
            T)
        (for X in FILEPKGTYPES when (LITATOM X) collect (CONS X (FILEPKGCHANGES X]
       [(EQ N 1)
        (COND
           [(LISTP (ARG N 1))
            (for X in (ARG N 1) when (FMEMB (CAR X)
                                            FILEPKGTYPES)
               do (/replace CHANGED of (CAR X) with (CDR X]
           (T (for Y on (fetch CHANGED of (ARG N 1)) when [AND (CAR Y)
                                                              (NOT (for Z in (CDR Y)
                                                                      thereis (CL:EQUAL (CAR Y)
                                                                                    Z]
                 collect (CAR Y]
       (T (/replace CHANGED of (ARG N 1) with (ARG N 2])

(GETFILEPKGTYPE
  [LAMBDA (TYPE ONLY NOERROR NAME)                              (* lmm "20-Nov-86 23:10")

    ;; Coerce TYPE to a well defined definition type (FILEPKG type) or a command. ONLY is an indicator of which is acceptable; if NIL, either one is
    ;; acceptable, if COMS, only commands are acceptable, and if TYPES, only types should be returned. If none is found, will signal an error if
    ;; NOERROR is NIL, otherwise return NIL.

    (COND
       [(LISTP TYPE)

        ;; given a list of types, coerce them all or return NIL

        (for X in TYPE collect (OR (GETFILEPKGTYPE X ONLY NOERROR NAME)
                                    (RETURN]
       ((EQ TYPE '?)

        ;; odd case, may be obsolete: if given IL:?, return all known types of NAME. Maybe used by EDITDEF(NAME ?)??

        (AND NAME (TYPESOF NAME)))
       [(AND (NEQ ONLY 'COMS)
             (OR (SELECTQ TYPE
                      (NIL 'FNS)
                      (T 'VARS)
                      NIL)
                 (for X in FILEPKGTYPES do (if (EQ TYPE X)
                                             then ;; type matched exactly

                                                  (RETURN TYPE)
                                           elseif (AND (LISTP X)
                                                      (EQ TYPE (CAR X)))
                                             then (RETURN (CDR X]
       [(AND (NEQ ONLY 'TYPE)
             (LITATOM TYPE)
             (PROG1 (CAR (FMEMB TYPE FILEPKGCOMSPLST))                 ; Prefer an exact match quickly

                    ]
       [(AND (NEQ ONLY 'COMS)
             (LITATOM TYPE)
             (for X in FILEPKGTYPES bind NAME do (SETQ NAME (if (NLISTP X)
                                                               then X
                                                             else (CAR X)))

                                                ;; see if spelled the same or 1 char shorter; assume all FILEPKGTYPE names end with S. This
                                                ;; handles package conversions and also pluralization

                                                       (AND (<= 0 (- (NCHARS NAME)
                                                                     (NCHARS TYPE))
                                                                1)
                                                            (STRPOS TYPE NAME)
                                                            (RETURN (if (EQ X NAME)
                                                                        then X
                                                                      else (CDR X]
       [(FIXSPELL TYPE NIL (SELECTQ ONLY
                                (TYPE FILEPKGTYPES)
                                (COMS FILEPKGCOMSPLST)
                                (UNION FILEPKGTYPES FILEPKGCOMSPLST]
       ((NOT NOERROR)
        (ERROR (SELECTQ ONLY
                    (TYPE "unrecognized manager definition type")
                    (COMS "unrecognized manager command")
                    "unrecognized manager definition-type/command")
               TYPE])

(MARKASCHANGED
  [LAMBDA (NAME TYPE REASON)                                    ; Edited 25-May-88 15:37 by drc:
    (COND
       (FILEPKGFLG (SETQ REASON (SELECTQ REASON
                                     ((CLISP LOAD CHANGED DEFINED DELETED)
                                         REASON)
                                     (NIL 'CHANGED)
                                     (T 'DEFINED)
                                     (ERROR "bad REASON in MARKASCHANGED" REASON)))
              (SETQ TYPE (GETFILEPKGTYPE TYPE 'TYPE))
              (for FN inside (fetch WHENCHANGED of TYPE) do (APPLY* FN NAME TYPE REASON))
              (for FN in MARKASCHANGEDFNS do (APPLY* FN NAME TYPE REASON))
              [COND
                 ((EQ REASON 'DELETED)
                  (for L on (fetch CHANGED of TYPE) when (EQUAL (CAR L)
                                                              NAME)
                     do (/RPLACA L NIL))                                ; unmark as changed and remove from files
                  (DELFROMFILES NAME TYPE))
                 (T (LET ((LST (push (fetch CHANGED of TYPE)
                                     NAME)))
                         (AND LISPXHIST (UNDOSAVE (LIST '/RPLACA LST)
                                                  LISPXHIST))           ; UNDO by smashing with NIL;  makes calls to
                                                                        ; MARKASCHANGED independent

                         ]
              NAME])

(FILECOMS
  [LAMBDA (FILE X)                                                      (* rmk%: "19-FEB-83 13:55")
    (COND
       ((AND (NULL FILE)
             (NULL X))
        'NILCOMS)
       [(AND (OR (NULL X)
                 (EQ X 'COMS))
             (fetch COMSNAME of (LISTP (fetch FILEPROP of FILE]
       (T (PACK* (NAMEFIELD FILE)
                 (OR X 'COMS])

(WHEREIS
  [LAMBDA (NAME TYPE FILES FN)                                          ; Edited 19-Aug-2022 20:13 by lmm
                                                                        ; Edited 12-Jul-88 17:14 by MASINTER

    ;; T as a NAME has a special meaning to INFILECOMS?  so don't pass through.

    (CL:UNLESS (EQ NAME T)
        (LET [(IN-FILES (UNION [SUBSET (OR (LISTP FILES)
                                           FILELST)
                                      (FUNCTION (LAMBDA (FILE)
                                                  (INFILECOMS? NAME TYPE (FILECOMS FILE]
                               (AND (EQ FILES T)
                                    (CL:FBOUNDP 'XCL::HASH-FILE-WHERE-IS)
                                    (LET ((FILES NIL))
                                         (for TY inside TYPE do (for FILE-NAME in (XCL::HASH-FILE-WHERE-IS
                                                                                   NAME
                                                                                   (GETFILEPKGTYPE TY))
                                                                 do (CL:PUSHNEW (MKATOM (U-CASE FILE-NAME))
                                                                               FILES)))
                                         (REVERSE FILES]
             (CL:IF FN
                 [MAPC IN-FILES (FUNCTION (LAMBDA (FILE)
                                             (APPLY* FN NAME FILE]
                 IN-FILES))))])

(SMASHFILECOMS
  [LAMBDA (FILE)                                                        (* rmk%: "19-FEB-83 22:15")
    (for X in (FILECOMSLST FILE 'FILEVARS) when (LITATOM X) do (SETTOPVAL X 'NOBIND))
    FILE])

(FILEFNSLST
  [LAMBDA (FILE)                                                        ; Edited 14-Jun-90 19:30 by jds
    (FILECOMSLST FILE '(FUNCTIONS FNS])

(FILECOMSLST
  [LAMBDA (FILE TYPE FLG)                                               (* JonL "24-Jul-84 19:48")
                                                                        ; TYPE is coerced in the innards of INFILECOMS?
    (COND
       ((EQ FLG 'UPDATE)
        (CDR (INFILECOMS? NIL TYPE (FILECOMS FILE)
                    FLG)))
       (T (INFILECOMS? NIL TYPE (FILECOMS FILE)
                 FLG])

(UPDATEFILES
  [LAMBDA (PRLST FLST)                                                  (* rmk%: "19-FEB-83 14:27")

    ;; PRLST may be the value of FILEPKGCHANGES before some operation (e.g.  LOAD, LOADFNS) involving the files in FLST began.

    (for TYPE CHANGED in FILEPKGTYPES when (SETQ CHANGED (fetch CHANGED of TYPE))
      do (COND
             ((NULL (SETQ CHANGED (FILEPKGCHANGES TYPE)))               ; FILEPKGCHANGES eliminates duplicates
              (/replace CHANGED of TYPE with NIL))
             (T (for FILE FOUND FILEPROP COMS LST TYPEDPROP PCHANGES (PREVITEMS _ (CDR (ASSOC TYPE PRLST)))
                   in FILELST first (SETQ LST (INFILECOMS? CHANGED TYPE 'NILCOMS 'UPDATE))

                                  ;; First check NIL=Nowhere.  LST:1 contains variables whose values are on the file literally.  These are
                                  ;; 'found' but not marked.  LST::1 contains all other items.

                                  (SETQ FOUND (NCONC (CAR LST)
                                                     (CDR LST)
                                                     FOUND))
                   do (SETQ PCHANGES (COND
                                         ((FMEMB (fetch DATEFILENAME of (CAR (fetch FILEDATES of FILE)))
                                                 FLST)

                                      ;; PREVITEMS are changed items that were previously on the changed list, before PRLST was
                                      ;; computed as this LOAD/LOADFNS began.  Thus, by this intersection we only worry about items
                                      ;; that were previously changed;  any items that were only changed during this operation are ignored.

                                          (INTERSECTION CHANGED PREVITEMS))
                                         (T CHANGED)))
                       [COND
                          ([AND PCHANGES [SETQ COMS (fetch COMSNAME of (SETQ FILEPROP (LISTP (fetch FILEPROP
                                                                                                  of FILE]
                                (SETQ LST (INFILECOMS? PCHANGES TYPE COMS 'UPDATE]

                           ;; LST:1 is a list of the times that literally appear on this file, LST::1 is a list of those whose literal values are not in the
                           ;; coms

                           [COND
                              ((CDR LST)                                    ; CDR items must be distributed
                               [COND
                                  ((NULL (fetch TOBEDUMPED of FILEPROP))

                                   ;; Only finagle global lists the first time an item is added to PROP, when PROP::1 goes from NIL to
                                   ;; non-NIL

                                   [/SETTOPVAL 'NOTLISTEDFILES (REMOVE FILE (GETTOPVAL 'NOTLISTEDFILES]
                                   (/SETTOPVAL 'NOTCOMPILEDFILES (REMOVE FILE (GETTOPVAL 'NOTCOMPILEDFILES]
                                                                            ; Get the (possibly new) TYPE item list to smash
                               [COND
                                  [(SETQ TYPEDPROP (ASSOC TYPE (fetch TOBEDUMPED of FILEPROP]
                                  (T (/NCONC1 FILEPROP (SETQ TYPEDPROP (CONS TYPE]
                                                                            ; Now distribute items to the file property
                               (for Y in (CDR LST) unless (MEMBER Y (CDR TYPEDPROP)) do (/NCONC1 TYPEDPROP Y]
                           (SETQ FOUND (NCONC (CAR LST)
                                              (CDR LST)
                                              FOUND]
                   finally (/replace CHANGED of TYPE with (LDIFFERENCE CHANGED FOUND])
```

```
(INFILECOMS?
  [LAMBDA (NAME TYPE COMS ONFILETYPE)                               ; Edited 12-Jul-88 17:42 by MASINTER

    ;; Returns T if NAME is 'CONTAINED' in COMS.  If NAME is NIL, then value is a list of all of the functions contained in COMS.  If NAME=T, value
    ;; is T if there are any elements of type TYPE, otherwise NIL (this feature is used for deciding whether or not (and how) to compile files.)  Called by
    ;; FILEFNSLST (which is used by BRECOMPILE) and by NEWFILE1.  while elements are the subset of NAME which are on the file in other case

    ;; if ONFILETYPE is UPDATE, then NAME is a list of elements, and INFILECOMS?  returns the dotted pair of (literals  . elements) where literals
    ;; are those which are 'literally' on the file (e.g.  (VARS (X 3))) --- if ONEFILETYPE is EDIT, then NAME is interpreted as for ONFILETYPE=NIL, but
    ;; only those elements which are not on the file literally and which are not subparts of other types are returned

    ;; if ONFILETYPE is TYPESOF, type can be a list of types, and returns a list of types suitable for EDITDEF

    (PROG (VAL LITERALS ORIGFLG)
          (SETQ TYPE (GETFILEPKGTYPE TYPE))
          (SELECTQ ONFILETYPE
               (EDIT (SELECTQ TYPE
                          (FILEVARS (RETURN))
                          NIL))
               NIL)
          [COND
             ((LITATOM COMS)
              (SELECTQ TYPE
                   ((VARS FILEVARS)                                         ; the COMS of a file are also on it
                        (INFILECOMSVAL COMS))
                   NIL)
              (SETQ COMS (EVALV COMS]
          (INFILECOMS COMS)
          (SETQ VAL (DREVERSE VAL))
          (RETURN (COND
                     ((EQ ONFILETYPE 'UPDATE)
                      (CONS LITERALS VAL))
                     (T VAL])
```

```
(INFILECOMTAIL
  [LAMBDA (COM FLG)                                                 ; Edited  2-Aug-88 02:15 by masinter
    [SETQ COM (COND
                 ((EQ (CADR COM)
                      '*)
                  (COND
                     [(LITATOM (CADDR COM))
                      (LISTP (EVALV (CADDR COM]
                     (T [RESETVARS (DWIMLOADFNSFLG)
                                   (NLSETQ (SETQ COM (EVAL (CADDR COM]
                         COM)))
                 (T (CDR COM]
    (if (NOT FLG)
        then (for X in COM do [if (AND (LISTP X)
                                     (EQ (CAR X)
                                         COMMENTFLG))
                                then (RETURN (SUBSET COM (FUNCTION (LAMBDA (X)
                                                                      (OR (NLISTP X)
                                                                          (NEQ (CAR X)
                                                                               COMMENTFLG]
               finally (RETURN COM))
      else COM])

(INFILECOMS
  [LAMBDA (COMS)                                                 (* rmk%: "19-FEB-83 22:17")
    (for X in COMS do (INFILECOM X])

(INFILECOM
  [LAMBDA (COM)                                                  ; Edited  2-Aug-88 02:27 by masinter
    (COND
       [(NLISTP COM)
        (COND
           ((EQ TYPE 'VARS)
            (INFILECOMSVAL COM]
       ((EQ (CAR COM)
            COMMENTFLG)

        ;; must be special case'd first so that (* * values) doesn't make it look like 'values' is a variable
                                                                 ; don't know why I should bother, but someone might want to
                                                                 ; know all of the comments on a file???

        (COND
           ((EQ TYPE COMMENTFLG)
            (INFILECOMSVAL COM T)))
        NIL)
       (T (PROG ((COMNAME (CAR COM))
                 (TAIL (CDR COM))
                 CFN TEM)
                (COND
                   [[COND
                       ((SETQ CFN (fetch (FILEPKGCOM CONTENTS) of COMNAME))
                        (SETQ TEM (APPLY* CFN COM (COND
                                                      ((AND (NULL ONFILETYPE)
                                                            (NOT (CL:SYMBOLP NAME)))
                                                                 ; call from WHEREIS of a name which is not a symbol
                                                       (LIST NAME))
                                                      (T NAME))
                                         TYPE ONFILETYPE)))
                       ((SETQ CFN (fetch (FILEPKGCOM PRETTYTYPE) of COMNAME))
                                                                 ; for compatability
                        (SETQ TEM (APPLY* CFN COM TYPE NAME]
                    (COND
                       [(NLISTP TEM)
                        (COND
                           ((EQ TEM T)
                            (COND
                               ((OR (EQ NAME T)
                                    (NULL ONFILETYPE))
                                (RETFROM 'INFILECOMS? T]
                       (T (INFILECOMSVALS TEM]
                   ((LISTP TAIL)

                    ;; this SELECTQ handles the 'exceptional cases' for the built in types.  There is an explicit RETURN in the SELECTQ clause
                    ;; if the default is handled

                    (SELECTQ COMNAME
                         ((PROP IFPROP)
                              (SETQ TAIL (CDR TAIL)))
                         NIL)
                    [COND
                       ((EQ (CAR TAIL)
                            '*)
                        (COND
                           ((LITATOM (CADR TAIL))
                            (SELECTQ TYPE
                                 ((VARS FILEVARS)
                                      (INFILECOMSVAL (CADR TAIL)))
                                 NIL))
                           ((AND (LISTP (CADR TAIL))
                                 (EQ ONFILETYPE 'UPDATE)
                                 (EQ TYPE 'VARS)
                                 (EQ (CAADR TAIL)
                                     'PROGN)
                                 (FMEMB (CAR (LAST (CADR TAIL)))
                                        NAME))
                            (SETQ VAL (CONS (CADR TAIL)
                                            VAL]
                    (SELECTQ COMNAME
                         ((COMS EXPORT)
                              (INFILECOMS (INFILECOMTAIL COM T)))
                         (CL:EVAL-WHEN (INFILECOMS (INFILECOMTAIL (CDR COM)
                                                                  T)))
                         (DECLARE%:                              ; skip over DECLARE: tags
                                    [RETURN (AND (NOT (FMEMB 'COMPILERVARS COM))
                                                 (IFCDECLARE (INFILECOMTAIL COM)
                                                        (EQ TYPE 'DECLARE%:])
                         (ORIGINAL                               ; dont expand macros
                                   (PROG ((ORIGFLG T))
                                         (INFILECOMS (INFILECOMTAIL COM T))))
                         ((PROP IFPROP)                          ; this currently does not handle ‘pseudo-types’ of PROPNAMES
                              (SELECTQ TYPE
                                  (PROPS (IFCPROPSCAN (INFILECOMTAIL (CDR COM)
                                                                     T)
                                                (CADR COM)))
                                  (MACROS (INFILECOMSMACRO (INFILECOMTAIL (CDR COM))
                                                 (CADR COM)))
                                 NIL))
                         (PROPS (RETURN (IFCPROPS COM)))
                         (MACROS (RETURN (SELECTQ TYPE
                                              (PROPS (IFCPROPSCAN (INFILECOMTAIL COM T)
                                                            MACROPROPS))
                                              (MACROS (INFILECOMSVALS (INFILECOMTAIL COM T)))
                                             NIL)))
                         (ALISTS                                 ; sigh.  This should probably also ‘coerce’ when asking for
                                                                 ; LISPXMACROS, etc.
                                 (RETURN (SELECTQ TYPE
                                              (ALISTS (INFILEPAIRS (INFILECOMTAIL COM)))
                                             NIL)))
                         (P [RETURN (SELECTQ TYPE
                                         ((EXPRESSIONS P)
                                             (INFILECOMSVALS (INFILECOMTAIL COM T)
                                                    T))
                                         (COND
                                            ((NULL ONFILETYPE)   ; for WHEREIS and FILECOMSLST
                                             (SELECTQ TYPE
                                                 (I.S.OPRS (IFCEXPRTYPE COM 'I.S.OPR))
                                                 (TEMPLATES (IFCEXPRTYPE COM 'SETTEMPLATE))
                                                 NIL])
                         ((ADDVARS APPENDVARS)
                              (SELECTQ TYPE
                                 (VARS [RETURN (AND (NULL ONFILETYPE)
                                                    (for X in (INFILECOMTAIL COM T)
                                                       do (INFILECOMSVAL (CAR X)
                                                                 T])
                                 (ALISTS [RETURN (for X in (INFILECOMTAIL COM T)
                                                    when (EQMEMB 'ALIST (GETPROP (CAR X)
                                                                             'VARTYPE))
                                                    do (for Z in (CDR X) do (INFILECOMSVAL (LIST (CAR X)
                                                                                                 (CAR Z))
                                                                                  T])
                                 (OR (EQ TYPE COMNAME)
                                     (RETURN))))
                         ((VARS INITVARS FILEVARS UGLYVARS HORRIBLEVARS CONSTANTS ARRAY)
                              [RETURN (COND
                                         ((EQ TYPE 'EXPRESSIONS)
                                          (for X in (INFILECOMTAIL COM T) when (AND (LISTP X)
                                                                                  (NEQ (CAR X)
                                                                                       COMMENTFLG))
                                            do (INFILECOMSVAL (CONS 'SETQ X)
                                                      T)))
                                         ((OR (EQ TYPE 'VARS)
                                              (EQ TYPE COMNAME)) ; either want all VARS, or else want all FILEVARS and this is a
                                                                 ; FILEVARS command
                                          (for X in (INFILECOMTAIL COM T)
                                            do (COND
                                                  ((LISTP X)
                                                    (AND (CAR X)
                                                         (NEQ (CAR X)
                                                              COMMENTFLG)
                                                         (INFILECOMSVAL (CAR X)
                                                                T)))
                                                  (X (INFILECOMSVAL X (EQ COMNAME 'INITVARS])
                         (DEFS [RETURN (for X in (INFILECOMTAIL COM T) when (EQ TYPE (CAR X))
                                          do (INFILECOMSVALS (CDR X])
                         (FILES (RETURN))
                         NIL)

                    ;; Exceptional cases now handled.  If TYPE matches (CAR COM) then scan the tail as usual.  Else expand the com's
                    ;; MACRO, if it has one, unless there was a CONTENTS function

                    (COND
                       ((EQ COMNAME TYPE)
                         (INFILECOMSVALS (INFILECOMTAIL COM T)))
                       [(AND (LISTP TYPE)
                              (FMEMB COMNAME TYPE))
                         (LET ((TYPE COMNAME))
                              (INFILECOMSVALS (INFILECOMTAIL COM T]
                       ((AND (OR (NULL CFN)
                                 (AND (EQ CFN T)
                                      (NULL ONFILETYPE)))
                              (NULL ORIGFLG)
                              (SETQ TEM (fetch (FILEPKGCOM MACRO) of COMNAME)))
                         (INFILECOMS (SUBPAIR (CAR TEM)
                                            (INFILECOMTAIL COM T)
```

```
                                            (CDR TEM])

(INFILECOMSVALS
  [LAMBDA (X FLG)                                                ; Edited  2-Aug-88 02:21 by masinter
    (for Y in X when (NOT (AND (LISTP Y)
                              (EQ (CAR Y)
                                  COMMENTFLG)))
       do (INFILECOMSVAL Y FLG])

(INFILECOMSVAL
  [LAMBDA (X FLG)                                                ; Edited 12-Jul-88 17:56 by MASINTER
    (COND
       [(EQ ONFILETYPE 'UPDATE)
        (AND (OR (NULL NAME)
                 (MEMBER X NAME))
             (COND
                (FLG (SETQ LITERALS (CONS X LITERALS)))
                (T (SETQ VAL (CONS X VAL]
       ((AND (EQ ONFILETYPE 'EDIT)
             FLG)                                                ; literals should not be edited as they are on the fileCOMS
        NIL)
       ((EQ ONFILETYPE 'TYPESOF)
        (AND (COND
                ((LITATOM NAME)
                 (EQ NAME X))
                (T (EQUAL NAME X)))
             (CL:PUSHNEW TYPE VAL)))
       ([OR (EQ NAME T)
            (COND
               ((LITATOM NAME)
                (EQ NAME X))
               (T (EQUAL NAME X]
        (RETFROM (FUNCTION INFILECOMS?)
               T))
       ((NULL NAME)
        (SETQ VAL (CONS X VAL])

(INFILECOMSPROP
  [LAMBDA (AT PROP)                                              (* lmm "25-SEP-81 17:15")
    (COND
       [(EQ ONFILETYPE 'UPDATE)
        (AND [OR (NULL NAME)
                 (find X in NAME suchthat (AND (EQ (CAR X)
                                                   AT)
                                               (EQ (CADR X)
                                                   PROP]
             (SETQ VAL (CONS (LIST AT PROP)
                             VAL]
       ((OR (EQ NAME T)
            (AND (EQ (CAR NAME)
                     AT)
                 (EQ (CADR NAME)
                     PROP)))
        (RETFROM (FUNCTION INFILECOMS?)
               T))
       ((NULL NAME)
        (SETQ VAL (CONS (LIST AT PROP)
                        VAL])

(IFCPROPS
  [LAMBDA (COM)                                                  (* bvm%: " 2-Dec-83 14:24")

;;; Examine a PROPS com for objects of specified TYPE

    (SELECTQ TYPE
         (PROPS                                                  ; the PROPS command can actually take (PROPNAME at1 at2
                                                                 ; ...)
                (INFILEPAIRS (INFILECOMTAIL COM)))
         (PROP                                                   ; return the atoms which have any properties at all
               (for PAIR in (INFILECOMTAIL COM) do (for ATNAME inside (CAR PAIR) do (INFILECOMSVAL ATNAME))))
         (MACROS                                                 ; only MACRO properties
                 (for PAIR in (INFILECOMTAIL COM) do (INFILECOMSMACRO (CAR PAIR)
                                                          (CDR PAIR))))
        NIL])

(IFCEXPRTYPE
  [LAMBDA (COM FN)                                               ; Edited  6-Apr-87 20:20 by Pavel

;;; Recognizes expressions in COM (a P com) that are calls to function FN

    (for SUBCOM in (INFILECOMTAIL COM) when (AND (EQ (CAR SUBCOM)
                                                     FN)
```

```
                                                 (EQ (CAR (LISTP (CADR SUBCOM)))
                                                     'QUOTE))
       do (INFILECOMSVAL (CADR (CADR SUBCOM))
                 T])

(IFCPROPSCAN
  [LAMBDA (ATOMS PROPNAMES)                                       ; Edited  2-Aug-88 02:20 by masinter

;;; Recognizes members of ATOMS as being names (atom prop) of type PROPS for any prop in PROPNAMES

    (for AT in ATOMS when (LITATOM AT) unless [COND
                                                [(EQ ONFILETYPE 'UPDATE)
                                                 (COND
                                                    (NAME (NOT (ASSOC AT NAME]
                                                ((LISTP NAME)
                                                 (NEQ AT (CAR NAME]
       do (COND
             ((EQ PROPNAMES 'ALL)
              (for PROP in (GETPROPLIST AT) by (CDDR PROP) when (NOT (FMEMB PROP SYSPROPS))
                 collect (INFILECOMSPROP AT PROP)))
             (T (for PROP inside PROPNAMES do (INFILECOMSPROP AT PROP])

(IFCDECLARE
  [LAMBDA (TAIL WANTDECLARE)                                      ; Edited  8-Jun-90 18:11 by teruuchi
    (PROG ((TAIL TAIL))
      LP   (COND
             ((LISTP TAIL)
              [SELECTQ (CAR TAIL)
                   ((EVAL@LOADWHEN EVAL@COMPILEWHEN COPYWHEN)
                       [AND WANTDECLARE (INFILECOMSVAL (LIST (CAR TAIL)
                                                                       (CADR TAIL]
                       (SETQ TAIL (CDR TAIL)))
                   (DONTEVAL@LOAD
                       [COND
                          ((OR (\STKSCAN 'DOFILESLOAD)
                               (\STKSCAN 'LOAD))                  ; Edited by TT (8-June-90 : for AR#9376) In loading, discard the
                                                                  ; following contents in DECLARE tag "DONTEVAL@LOAD"
                           (RETURN))
                          (WANTDECLARE (INFILECOMSVAL (CAR TAIL])
                   (COMPILERVARS (RETURN))
                   (COND
                      [(FMEMB (CAR TAIL)
                              DECLARETAGSLST)
                       (COND
                          (WANTDECLARE (INFILECOMSVAL (CAR TAIL]
                      (T (INFILECOM (CAR TAIL]
              (SETQ TAIL (CDR TAIL))
              (GO LP])

(INFILEPAIRS
  [LAMBDA (LST)                                                   (* lmm " 4-DEC-78 09:51")
    (for LL in LST do (for X inside (CAR LL) do (for Y inside (CDR LL) do (INFILECOMSVAL (LIST X Y])

(INFILECOMSMACRO
  [LAMBDA (ATS PROPS)                                             (* lmm "28-SEP-78 18:35")
    ;; this function is used, given a PROP or PROPS command, to tell which MACROS are contained in it.  --- Normally (e.g.  for WHEREIS and
    ;; FILECOMSLST) it wants to return if the command contains any of the MACROPROPS for the given atom.  However, for UPDATE, it only wants a
    ;; 'hit' if the command contains ALL of the macro properties

    (for AT inside ATS do (AND [OR (NEQ ONFILETYPE 'UPDATE)
                                  (EVERY (PROPNAMES AT)
                                         (FUNCTION (LAMBDA (X)
                                                     (OR (NOT (FMEMB X MACROPROPS))
                                                         (EQMEMB X PROPS]
                              [SOME MACROPROPS (FUNCTION (LAMBDA (PROP)
                                                           (EQMEMB PROP PROPS]
                              (INFILECOMSVAL AT])
)

;; adding to a file

(DEFINEQ

(FILES?
  [LAMBDA NIL                                                     (* bvm%: "27-Oct-86 18:14")

;;; Display each file needing dumping, etc.  For files needing dumping, display details of why.

    (UPDATEFILES)
    (LET (FILES CHANGES PRINTED)
         (for FILE in FILELST when [SETQ CHANGES (fetch TOBEDUMPED of (LISTP (fetch FILEPROP of FILE]
            do (if (NOT PRINTED)
                   then (LISPXPRIN1 "To be dumped:
                             " T)
                        (SETQ PRINTED T))
               (LISPXPRIN2 FILE T)
               (LISPXPRIN1 " ...changes to " T)
               [for CH in CHANGES bind TB do (COND
                                                ((LISTP CH)
                                                 [COND
                                                    (TB (LISPXTAB TB NIL T))
                                                    (T (SETQ TB (POSITION T]
                                                 (LISPXPRIN2 (CAR CH)
                                                        T)
                                                 (FILES?PRINTLST (CDR CH)))
                                                (T                 ; old style
                                                   (LISPXPRIN2 CH T)
                                                   (LISPXSPACES 1 T]
               (LISPXTERPRI T))
         (for TYPE FLG in FILEPKGTYPES when (FILES?1 TYPE (AND PRINTED "    plus ")) do (SETQ FLG T)
            finally (if FLG
                          then (OR PRINTED (LISPXPRIN1 "...to be dumped.  " T))
                               (ADDTOFILES?)))
         (if (SETQ FILES NOTCOMPILEDFILES)
             then (FILES?PRINTLST FILES "To be compiled: ")
                  (LISPXTERPRI T))
         (if (SETQ FILES NOTLISTEDFILES)
             then (FILES?PRINTLST FILES "To be listed: ")
                  (LISPXTERPRI T))
         (CL:VALUES])
```

```
(FILES?1
  [LAMBDA (TYPE FIRST)                                              ; Edited 13-Jun-2021 10:18 by rmk:

    ;; If there are changed objects of TYPE, then print them out, preceded by FIRST (if given) plus a descriptive string, and return T.

    (LET (STR LST)
         (COND
            ([AND (LITATOM TYPE)
                  (SETQ STR (fetch (FILEPKGTYPE DESCRIPTION) of TYPE))
                  (LISTP (SETQ LST (fetch (FILEPKGTYPE CHANGED) of TYPE]
             (AND FIRST (LISPXPRIN1 FIRST T))
             (LISPXPRIN1 '"the " T)
             (LISPXPRIN1 STR T)
             (FILES?PRINTLST LST)
             (LISPXTERPRI T)
             T])
```

```
(FILES?PRINTLST
  [LAMBDA (LST STR)                                                 (* bvm%: "27-Oct-86 18:15")

    ;; Print elements of LST separated by commas and indenting new lines a bunch.  If MAPRINT had a left margin arg, this would be simpler.

    (MAPRINT LST T (OR STR ": ")
           NIL ", " [FUNCTION (LAMBDA (STR)
                                 (COND
                                    ((> (+ (POSITION T)
                                           (NCHARS STR T T)
                                           3)
                                        (LINELENGTH NIL T))
                                     (LISPXTERPRI T)
                                     (LISPXPRIN1 "          " T)))
                                 (LISPXPRIN2 STR T T]
           T])
```

```
(ADDTOFILES?
  [LAMBDA (NOASKSTR)                                                ; Edited 13-Jun-2021 10:22 by rmk:
                                                                    ; Edited 21-Aug-91 10:13 by jds

    ;; ask user about all of the things that need to be dumped, and distribute them to the files that he says

    ;; RMK: Eliminated literal CR's in the key list.

    (ERSETQ
     (PROG [BUFS (VARSCHANGES (fetch (FILEPKGTYPE CHANGED) of 'VARS]

      ;; Save VARS list at the beginning, so that changes that might occur from adding things to files (e.g.  changing NILCOMS) will not be processed
      ;; differently depending on the order of elements in FILEPKGTYPES

           [COND
              (NOASKSTR (PRIN1 NOASKSTR T))
              (T (DOBE)
                 (SETQ BUFS (READP T))
                 (SELECTQ (ASKUSER DWIMWAIT 'N '("want to say where the above go")
                                  `([Y ,(CONCAT "es" (CHARACTER (CHARCODE EOL]
                                    [N ,(CONCAT "o" (CHARACTER (CHARCODE EOL]
                                    (%] ,(CONCAT "Nowhere" (CHARACTER (CHARCODE EOL)))
                                        EXPLAINSTRING "] - nowhere, all items will be marked as dummy
                                        " NOECHOFLG T))
                                 T)
                      (N (RETURN))
           (%]                                        ; Nowhere
               (for TYPE in FILEPKGTYPES do (for NAME in (fetch (FILEPKGTYPE CHANGED) of TYPE)
                                             do (ADDTOFILE NAME TYPE NIL)))
               (RETURN))
          NIL)                                        ; if there was type-ahead BEFORE the askuser, then don't allow
                                                      ; it now
      (COND
         (BUFS (SETQ BUFS (COND
                             ((READP T)
                              (LINBUF)
                              (SYSBUF)
                              (SETQ BUFS (CLBUFS NIL T READBUF]
[for TYPE STR LST in FILEPKGTYPES when [AND (SETQ STR (fetch (FILEPKGTYPE DESCRIPTION) of TYPE))
                                           (LISTP (SETQ LST (COND
                                                               ((EQ TYPE 'VARS)
                                                                VARSCHANGES)
                                                               (T (fetch (FILEPKGTYPE CHANGED)
                                                                     of TYPE]
   do
   (printout T "(" STR ")" T)
   (for NAME TEM FILE in LST when NAME
      do (PROG NIL
           LP  (PRIN2 NAME T)
               (SPACES 2 T)

          ;; if user typed ahead before entering addtofiles??  then dont allow typeahead here, because it will justgobble his earlier
          ;; typeahead.

          ;; SELCHARQ to avoid literal CR

               (SELCHARQ (CHCON1 (SETQ TEM (ASKUSER NIL NIL NIL ADDTOFILEKEYLST T)))
                    (%[ (ERSETQ (PROGN (SHOWDEF NAME TYPE T)

                          ;; the DOBE is so that if the user control-E's after the printout is done but before it appears on the
                          ;; screen that the control-E will merely clear output buffer

                                       (DOBE)))
                        (GO LP))
                    (%]                               ; Nowhere
                        (SETQ FILE))
                    (SPACE                            ; No action
                           (RETURN))
                    ((LF =)
                         (PRINT (OR (SETQ FILE LASTFILE)
                                    'Nowhere)
                                T))
                    (SETQ FILE TEM))
               (OR (ERSETQ (PROG (TEM COMSNAME PLACE LISTNAME NEAR)
                                 (SETQ PLACE (WHATIS FILE NIL TYPE))
                                 [COND
                                    ((LITATOM PLACE)  ; file name
                                     (SETQ FILE PLACE)
                                     (OR (ADDTOCOMS (SETQ COMSNAME (FILECOMS FILE))
                                                NAME TYPE NEAR LISTNAME)
                                         (ADDNEWCOM COMSNAME NAME TYPE NIL FILE))
                                     (for F in (fetch (FILEPKGTYPE WHENFILED) of TYPE)
                                        do (APPLY* F NAME TYPE FILE))

                                     ;; This isn't factored to the end, cause ADDTOLISTNAME might have to deal with a set
                                     ;; of old elements on the listname.

                                     )
                                    ((EQ (CAR PLACE)
                                         'Near%:)
                                     (SETQ NEAR (CADR PLACE))
                                     (COND
                                        ([SOME FILELST (FUNCTION (LAMBDA (FL)
                                                                   (ADDTOCOMS (FILECOMS
                                                                                (SETQ FILE FL))
                                                                          NAME TYPE NEAR LISTNAME]
                                         (PRINT (LIST 'on FILE)
                                                T T))
                                        (T (PRINT (LIST (CADR PLACE)
                                                        'not
                                                        'found)
                                                  T T)
                                           (ERROR!)))
                                     (for F in (fetch (FILEPKGTYPE WHENFILED) of TYPE)
                                        do (APPLY* F NAME TYPE FILE)))
                                    ([OR [UNDONLSETQ (PROGN (SAVESET (SETQ LISTNAME (CAR PLACE))
                                                                   (MERGEINSERT NAME
                                                                          (LISTP (GETTOPVAL LISTNAME)
                                                                                 )
                                                                          T)
                                                                   T
                                                                   'NOPRINT)
                                                            (OR (SETQ FILE
                                                                     (CAR (WHEREIS NAME TYPE FILELST)))
                                                                    (ERROR!]
                                         (SOME FILELST (FUNCTION (LAMBDA (X)
                                                                                (ADDTOCOMS (FILECOMS
                                                                                              (SETQ FILE X))
                                                                                      NAME TYPE NEAR LISTNAME]
                                                 (PRIN1 "  value is filed on " T)
                                                 (PRINT FILE T T)
                                                 (for F in (fetch (FILEPKGTYPE WHENFILED) of TYPE)
                                                    do (APPLY* F NAME TYPE FILE))
                                      ;; Only have to notice the single new item here, unlike the case in ADDNEWCOM below, cause other
                                      ;; items on the list already belong and were previously noticed

                                                 )
                                                (T (PRIN1 "    put list " T)
                                                   (PRIN2 (CAR PLACE)
                                                          T T)
                                                   (SETQ FILE (WHATIS (ASKUSER NIL NIL " on file: "
                                                                              '((""  "" EXPLAINSTRING "a file
                                                                                    name" KEYLST (Ÿ)))
                                                                              T)
                                                                     'FILE))
                                                   (SAVESET (CAR PLACE)
                                                          (MERGEINSERT NAME (LISTP (GETTOPVAL (CAR PLACE)))
                                                                 T)
                                                          T
                                                          'NOPRINT)
                                                 ;; Add new item before new command, so that user's new command function can
                                                 ;; inspect (CAR PLACE) and see all the items involved.
                                                   (ADDNEWCOM (FILECOMS FILE)
                                                          NAME TYPE (CAR PLACE)
                                                          FILE)
                                                   (for F in (fetch (FILEPKGTYPE WHENFILED) of TYPE)
                                                      do (for I in (GETTOPVAL (CAR PLACE))
                                                            do (APPLY* F I TYPE FILE]
                                             (AND FILE (ADDFILE FILE))
                                             (SETQ LASTFILE PLACE)))
                              (GO LP]
           (AND BUFS (BKBUFS BUFS))
           (UPDATEFILES])

(ADDTOFILE
  [LAMBDA (NAME TYPE FILE NEAR LISTNAME)                        (* lmm "21-Nov-84 11:43")
                                                                 ; adds NAME to the file FILE
    (PROG (TEM COMSNAME)
          [SETQ TYPE (OR (GETFILEPKGTYPE TYPE NIL T)
                         (COND
                            ((FMEMB TYPE FILELST)
                             (GETFILEPKGTYPE (swap TYPE FILE)))
                            (T (GETFILEPKGTYPE TYPE]
          (SETQ FILE (WHATIS FILE 'FILE))
          (OR (ADDTOCOMS (SETQ COMSNAME (FILECOMS FILE))
                     NAME TYPE NEAR LISTNAME)
              (ADDNEWCOM COMSNAME NAME TYPE NIL FILE))
          (for F in (fetch WHENFILED of TYPE) do (APPLY* F NAME TYPE FILE))
          (AND FILE (NOT (FMEMB FILE FILELST))
               (ADDFILE FILE))
          (RETURN FILE])

(WHATIS
  [LAMBDA (USERINPUT ONLY)                                      (* lmm "28-Nov-84 16:49")
    ;; decides whether USERINPUT is a file or a list name --- if ONLY is nil, means either a listname or a filename is accepatble;  if ONLY is LIST then
    ;; only a listname is acceptable and if ONLY is FILE then only a file name is acceptable
    (PROG (TEM UCASE)
          (RETURN (COND
                     ((NULL USERINPUT)                          ; nowhere
                      NIL)
                     [(LISTP USERINPUT)
                      (COND
                         (ONLY (ERROR!))
                         (T (SELECTQ (CAR USERINPUT)
                                ((@ Near%:)
                                   (CONS 'Near%: (CDR USERINPUT)))
                                (WHATIS (CAR USERINPUT)
                                       'LIST]
                     ([AND (NEQ ONLY 'LIST)
                           (OR (FMEMB (SETQ TEM (SETQ UCASE (U-CASE USERINPUT)))
                                      FILELST)
                               (LISTP (GETTOPVAL (FILECOMS UCASE)))
                               (SETQ TEM (FIXSPELL UCASE NIL FILELST T]
                      TEM)
                     ((AND (NEQ ONLY 'FILE)
                           (LISTP (GETTOPVAL USERINPUT)))
                      (LIST USERINPUT))
                     ((AND (NEQ ONLY 'LIST)
                           (EQ (ASKUSER NIL NIL (LIST "create new file" UCASE)
                                      NIL T)
                               'Y))
                      UCASE)
                     ((AND (NEQ ONLY 'FILE)
                           (EQ (ASKUSER NIL NIL (LIST "create new list" USERINPUT)
                                      NIL T)
                               'Y))
                      (LIST USERINPUT))
                     (T                                          ; none of above
                        (ERROR!])

(ADDTOCOMS
  [LAMBDA (COMS NAME TYPE NEAR LISTNAME)                         (* rmk%: "10-JUN-82 22:53")
    ;; try to insert NAME of type TYPE command list COMS (either a coms name, or a just a list of coms);  return NIL if unsuccessful.  If LISTNAME is
    ;; given, then only insert by adding to LISTNAME.  If NEAR is given, only insert near it
    (COND
       ((NULL COMS)
        NIL)
       [(LITATOM COMS)                                           ; given a name of a command;  rebind COMSNAME to current
                                                                 ; variable and try to add to its value
        (OR [PROG ((COMSNAME COMS))
                  (RETURN (ADDTOCOMS (LISTP (GETTOPVAL COMSNAME))
                                 NAME TYPE NEAR (AND (NEQ COMS LISTNAME)
                                                     LISTNAME]
            (AND (EQ COMS LISTNAME)
                 (ADDNEWCOM COMS NAME TYPE]
       (T (SETQ TYPE (GETFILEPKGTYPE TYPE))
          (for TAIL on COMS do (COND
                                  [(LISTP (CAR TAIL))
                                   (COND
                                      ((ADDTOCOM (CAR TAIL)
                                              NAME TYPE NEAR LISTNAME)
                                       (RETURN T]
                                  (T (SELECTQ (CAR TAIL)
                                         ((EVAL@LOADWHEN EVAL@COMPILEWHEN COPYWHEN)
                                              (SETQ TAIL (CDR TAIL)))
                                         NIL])

(ADDTOCOM
  [LAMBDA (COM NAME TYPE NEAR LISTNAME)                          ; Edited  2-May-87 19:04 by Pavel
                                                                 ; tries to insert NAME into the prettycom COM;  returns NIL if
                                                                 ; unsuccessful
    (PROG (TEM)
          (COND
             ([AND NEAR (NOT (INFILECOMS? NEAR TYPE (LIST COM]
              (RETURN)))
          [COND
             ((SETQ TEM (fetch ADD of (CAR COM)))
              (RETURN (COND
                         ((OR (NULL LISTNAME)
                              (INFILECOMS? LISTNAME 'FILEVARS (LIST COM)))
                          (AND (SETQ TEM (APPLY* TEM COM NAME TYPE NEAR))
                               (MARKASCHANGED COMSNAME 'VARS))
                          TEM]
          (RETURN (SELECTQ (CAR COM)
                      (FNS (AND (EQ TYPE 'FNS)
                                (ADDTOCOM1 COM NAME NEAR LISTNAME)))
                      ((VARS INITVARS)
                           (COND
                              ((OR (EQ (CAR COM)
                                       'VARS)
                                   NEAR LISTNAME)                ; Don't stick on INITVARS unless NEAR or LISTNAME says we
                                                                 ; should.
                               (SELECTQ TYPE
                                   (EXPRESSIONS (COND
                                                   ((EQ (CAR NAME)
                                                        'SETQ)
                                                    (ADDTOCOM1 COM (CDR NAME)
                                                           NEAR LISTNAME))))
                                   (VARS (ADDTOCOM1 COM NAME NEAR LISTNAME))
                                   NIL))))
                      (COMS (ADDTOCOMS (COND
                                          [(EQ (CADR COM)
                                               '*)
                                           (COND
                                              ((LITATOM (CADDR COM))
                                               (CADDR COM))
                                              (T (RETURN]
                                          (T (CDR COM)))
                                   NAME TYPE NEAR LISTNAME))
                      (DECLARE%: (AND (OR LISTNAME NEAR)
                                      (ADDTOCOMS (COND
                                                    [(EQ (CADR COM)
                                                         '*)
                                                        (COND
                                                           ((LITATOM (CADDR COM))
                                                            (CADDR COM))
                                                           (T (RETURN]
                                                       (T (CDR COM)))
                                              NAME TYPE NEAR LISTNAME)))
                       (CL:EVAL-WHEN (AND (OR LISTNAME NEAR)
                                          (ADDTOCOMS (COND
                                                          [(EQ (CL:THIRD COM)
                                                               '*)
                                                           (COND
                                                              ((LITATOM (CL:FOURTH COM))
                                                               (CL:FOURTH COM))
                                                              (T (RETURN]
                                                          (T (CDDR COM)))
                                                 NAME TYPE NEAR LISTNAME)))
                       ((PROP IFPROP)
                           (SELECTQ TYPE
                               (PROPS (COND
                                          ((EQ (CADR COM)
                                               (CADR NAME))
                                           (ADDTOCOM1 (CDR COM)
                                                  (CAR NAME)
                                                  NEAR LISTNAME))
                                          ((AND (EQ (CAR NAME)
                                                    (CADDR COM))
                                                (NULL (CDDDR COM)))
                                           [/RPLACA (CDR COM)
                                                  (UNION (MKLIST (CDR NAME))
                                                         (MKLIST (CADR COM]
                                           (MARKASCHANGED COMSNAME 'VARS)
                                           T)))
                               (MACROS (COND
                                           ([AND (for PROP inside (CADR COM) always (EQMEMB PROP MACROPROPS))
                                                 (for PROP in MACROPROPS
                                                    always (OR (EQMEMB PROP (CADR COM))
                                                               (NOT (GETPROP NAME PROP]

                                            ;; every property in the command is a macro prop and, either this is an IFPROP or else the
                                            ;; MACROS are changed

                                            (ADDTOCOM1 (CDR COM)
                                                   NAME NEAR LISTNAME))))
                               NIL))
                       ((PROPS ALISTS)
                           (AND (EQ TYPE (CAR COM))
                                (ADDTOCOM1 COM (/NCONC1
                                                  (OR [ASSOC (CAR NAME)
                                                             (COND
                                                                [(EQ (CADR COM)
                                                                     '*)
                                                                 (COND
                                                                    [(LITATOM (CADDR COM))
                                                                     (AND (OR (NULL LISTNAME)
                                                                              (EQ (CADDR COM)
                                                                                  LISTNAME))
                                                                          (GETTOPVAL (CADDR COM]
                                                                    (T (RETURN]
                                                                (T (CDR COM]
                                                      (LIST (CAR NAME)))
                                                  (CADR NAME))
                                       NEAR LISTNAME)))
                       (P (COND
                             ((AND (EQ TYPE 'EXPRESSIONS)
                                   (NEQ (CAR NAME)
                                        'SETQ))
                              (ADDTOCOM1 COM NAME NEAR LISTNAME))))
                       (AND (EQ (CAR COM)
                                TYPE)
                            (ADDTOCOM1 COM NAME NEAR LISTNAME])

(ADDTOCOM1
  [LAMBDA (COM NAME NEAR LISTNAME)                               (* rmk%: " 3-JAN-82 22:53")
    (COND
       [(EQ (CADR COM)
            '*)                                                  ; add to list name
        (AND [COND
                (LISTNAME (EQ (CADDR COM)
                              LISTNAME))
                (T (LITATOM (CADDR COM]
             (SAVESET (CADDR COM)
                    [PROGN [SETQ COM (LISTP (GETTOPVAL (CADDR COM]
                           (COND
                              ((AND NEAR (SETQ NEAR (MEMBER NEAR COM)))
                               (/RPLACD NEAR (CONS NAME (CDR NEAR)))
                               COM)
                              (T (MERGEINSERT NAME COM T]
                    T
                    'NOPRINT]
       ((NULL LISTNAME)                                          ; add to standard com
        [AND (NOT (MEMBER NAME (CDR COM)))
             (COND
                [(SETQ NEAR (MEMBER NEAR COM))
                 (/RPLACD NEAR (CONS NAME (CDR NEAR]
                (T (/RPLACD COM (MERGEINSERT NAME (CDR COM]
        (MARKASCHANGED COMSNAME 'VARS)
        T])

(ADDNEWCOM
  [LAMBDA (COMSNAME NAME TYPE LISTNAME FILE)                     (* rmk%: " 3-JAN-82 22:53")
   ;; Adds to COMSNAME a new command that will dump NAME as a TYPE on FILE.  --- if LISTNAME is given, then use it as the listname
    (PROG (NEWCOM OLDCOM TAIL)
          (SETQ NEWCOM (MAKENEWCOM NAME TYPE LISTNAME FILE))
          [COND
             ((NLISTP (SETQ TAIL (GETTOPVAL COMSNAME)))
              (RETURN (SAVESET COMSNAME (LIST NEWCOM)
                             T
                             'NOPRINT]
      LP  [COND
             ((OR (NLISTP (SETQ OLDCOM (CAR TAIL)))
                  (SELECTQ (CAR OLDCOM)
                      ((LOCALVARS SPECVARS BLOCKS)
                           T)
                      (DECLARE%: (FMEMB 'COMPILERVARS (CDR OLDCOM)))
                      NIL))
              (/ATTACH NEWCOM TAIL))
             ((LISTP (CDR TAIL))
              (SETQ TAIL (CDR TAIL))
              (GO LP))
             (T (/RPLACD TAIL (LIST NEWCOM]
          (MARKASCHANGED COMSNAME 'VARS])

(MAKENEWCOM
  [LAMBDA (NAME TYPE LISTNAME FILE)                              ; Edited  8-Apr-87 14:55 by Pavel
    (SETQ TYPE (GETFILEPKGTYPE TYPE))
    (PROG (TEM)
     ;; the user function MUST (a) check if FILE = T and not do anything destructive (since this is only for showdef) and (b) if LISTNAME is given, then
     ;; use it rather than generating a different listname
          (AND (LISTP NAME)
               (SETQ NAME (COPY NAME)))
          (RETURN (OR (AND (SETQ TEM (fetch NEWCOM of TYPE))
                          (APPLY* TEM NAME TYPE LISTNAME FILE))
                      (SELECTQ TYPE
                           (PROPS [AND (NULL LISTNAME)
                                       (CONS 'PROP (CONS (COND
                                                            ((AND (LISTP (CDR NAME))
                                                                  (NULL (CDDR NAME)))
                                                             (CADR NAME))
                                                            (T (CDR NAME)))
                                                         (OR (LISTP (CAR NAME))
                                                             (LIST (CAR NAME])
                           (EXPRESSIONS [COND
                                           ((EQ (CAR NAME)
                                                'SETQ)
                                            (MAKENEWCOM (CDR NAME)
                                                   'VARS LISTNAME FILE))
                                           (T (CONS 'P (COND
                                                          (LISTNAME (LIST '* LISTNAME))
                                                          (T (LIST NAME])
                          NIL)
                      (DEFAULTMAKENEWCOM NAME TYPE LISTNAME FILE])

(DEFAULTMAKENEWCOM
  [LAMBDA (NAME TYPE LISTNAME FILE)                              ; Edited 13-Jun-2021 10:24 by rmk:
    (COND
       ((NOT (OR (FMEMB TYPE FILEPKGCOMSPLST)
                 (fetch (FILEPKGCOM MACRO) of TYPE)
                 (fetch (FILEPKGTYPE GETDEF) of TYPE)))
        (ERROR "no defined way to dump or obtain the definition of " (OR (fetch (FILEPKGTYPE DESCRIPTION)
                                                                            of TYPE)
                                                                         TYPE)
               T))
       ((NULL DEFAULTCOMHASFILEFLG)                              ; disable FOOFNS FOOVARS junk
        (LIST TYPE NAME))
       ((EQ FILE T)                                              ; FILE=T only when called from SHOWDEF
        (LIST TYPE NAME))
       ([OR LISTNAME (AND FILE (SAVESET (SETQ LISTNAME (FILECOMS FILE TYPE))
                                      (MERGEINSERT NAME (LISTP (GETTOPVAL LISTNAME)))
                                             T)
                                      T
                                      'NOPRINT]                  ; The check (AND FILE --) is so that it will not bother with making
                                                                 ; listnames just for deleting items
        (LIST TYPE '* LISTNAME))
       (T (LIST TYPE NAME])

)

(RPAQ? DEFAULTCOMHASFILEFLG )

(ADDTOVAR MARKASCHANGEDFNS )

(DEFINEQ

(MERGEINSERT
  [LAMBDA (NEW LST ONEFLG)                                       (* lmm "30-Jun-86 18:11")
    ;; searches LST to find the most reasonable place to insert NEW.  Does nothing if ONEFLG is T and NEW is already a member of LST
    (COND
       ((AND ONEFLG (MEMBER NEW LST))
        LST)
       ((LISTP NEW)
        (/NCONC1 LST NEW))
       (T (PROG ((N 0)
                 LST1 PLACE TEM)
                (SETQ LST1 LST)
            LP
           ;; finds the function with the longest leading common substring.  The idea is that if the list is only paatially sorted, want to insert the new
           ;; thing in among those function that look like they are related.
                (COND
                   ((NULL LST1)
                    (GO OUT))
                   ((OR (LISTP (CAR LST1))
                        (SETQ TEM (STRPOS (CAR LST1)
                                         NEW 1 NIL T T)))
                    ;; this takes precedence over even a longer string so that for example in the list (ADDTOFILES? ADDTOFILE), ADDTOFILE1
                    ;; will be inserted aater ADDTOFILE
                    (SETQ PLACE LST1)
                    (GO OUT))
                   ((IGREATERP (SETQ TEM (MERGEINSERT1 (CAR LST1)
                                                    NEW))
                           N)
                    (SETQ N TEM)
                    (SETQ PLACE LST1)))
                (SETQ LST1 (CDR LST1))
                (GO LP)
            OUT (SETQ TEM (CAR PLACE))
                (OR [SOME (OR PLACE LST)
                          (FUNCTION (LAMBDA (X LST)
                                      (COND
                                         ([OR (ALPHORDER NEW X)
                                              (AND PLACE (NOT (ALPHORDER TEM X]
                                      ;; for example, if the FNS list is something like (...  FOO FOO1 ...) where the ...  may or may not be in
                                      ;; order, e.g.  (ZAP FOO FOO1 BLAH), then want to insert FOO2 after FOO1, i.e. before BLAH, even
                                      ;; though FOO2 wold not come before BLAH in a sorted list.
                                          (/ATTACH NEW LST))
                                         (T (SETQ TEM X)
                                            NIL]
                    (SETQ LST (/NCONC1 LST NEW)))
                (RETURN LST])

(MERGEINSERT1
  [LAMBDA (X Y)                                                  (* rmk%: "24-MAY-82 00:05")
    ;; value is the number of leading characters of X and Y that agree.
    (PROG ((N 1)
           C1 C2)
      LP  [COND
             ((OR (NULL (SETQ C1 (NTHCHARCODE X N)))
                  (NULL (SETQ C2 (NTHCHARCODE Y N)))
                  (NEQ C1 C2))
              (RETURN (SUB1 N]
          (SETQ N (ADD1 N))
          (GO LP])
)

;; RMK: Changed INITVARS to VARS, so = addition is a synonym for untypable LF, and also suppress appearance of raw CR and LF in the file

(DEFINEQ

(ADDTOFILEKEYLST
  [LAMBDA NIL                                                    ; Edited 12-Feb-2021 17:15 by larry
```

```
    `(("[" "" EXPLAINSTRING "[ -- prettyprint the item to terminal and then ask again" NOECHOFLG T)
      (= "" EXPLAINSTRING "= - same as previous response" NOECHOFLG T)
      (,(CHARACTER (CHARCODE LF))
       "" EXPLAINSTRING "{line-feed} - same as previous response" NOECHOFLG T)
      (" " ,(CONCATCODES (LIST (CHARCODE SPACE)
                               (CHARCODE EOL)))
           EXPLAINSTRING "{space} - no action" NOECHOFLG T)
      ("]" ,(CONCAT "Nowhere" (CHARACTER (CHARCODE EOL)))
           EXPLAINSTRING
           ,(CONCAT "] - nowhere, item is marked as a dummy" (CHARACTER (CHARCODE EOL)))
           NOECHOFLG T)
      ["(" "List:  (" EXPLAINSTRING "(list name)" NOECHOFLG T KEYLST ((Ÿ "" CONFIRMFG
                                                                        [%) %] ,(CHARACTER (CHARCODE SPACE))
                                                                            ,(CHARACTER (CHARCODE EOL]
                                                                        RETURN
                                                                        (CDR ANSWER]
      (@ "Near: " EXPLAINSTRING "@ other-item   -- put the item near the other item" NOECHOFLG T KEYLST
         ((Ÿ "" CONFIRMFLG [,(CHARACTER (CHARCODE EOL]
            RETURN ANSWER)))
      [,(CHARACTER (CHARCODE CR))
       "" RETURN ,(CHARACTER (CHARCODE SPACE]
      ("" "File name: " EXPLAINSTRING "a file name" KEYLST (Ÿ])

)

(RPAQ? ADDTOFILEKEYLST (ADDTOFILEKEYLST))

(RPAQ? LASTFILE )
```

;; deleting an item from a file

```
(DEFINEQ

(DELFROMFILES
  [LAMBDA (NAME TYPE FILES)                                         (* rmk%: " 6-MAR-82 13:16")

    ;; Eliminates NAME as an item of type TYPE in COMS.

    (PROG (COMS)
          (SETQ TYPE (GETFILEPKGTYPE TYPE))
          (RETURN (for FILE inside (OR FILES FILELST)
                      when (PROG1 (DELFROMCOMS (SETQ COMS (FILECOMS FILE))
                                         NAME TYPE)
                                (COND
                                   ((INFILECOMS? NAME TYPE COMS)
                                    (printout T "(could not delete " NAME " from " FILE ")" T))))
                      collect (for FN in (fetch WHENUNFILED of TYPE) do (APPLY* FN NAME TYPE FILE))
                            FILE])

(DELFROMCOMS
  [LAMBDA (COMS NAME TYPE)                                          (* bvm%: " 1-Oct-86 22:02")

    ;; delete NAME of type TYPE from the coms COMS (either the name of some coms or a list).  Returns T if it does anything

    ;; If COMS is not a symbol, caller is required to bind COMSNAME to the symbol whose value we are deleting from, for benefit of marking it
    ;; changed.

    (COND
       [(LITATOM COMS)
        (LET ((COMSNAME COMS))
             (DECLARE (SPECVARS COMS))
             (AND (LISTP (SETQ COMS (GETTOPVAL COMSNAME)))
                  (DELFROMCOMS COMS NAME TYPE]
       (T (PROG (DONE)
                (SETQ TYPE (GETFILEPKGTYPE TYPE))
            LP  (COND
                   ((NLISTP COMS)
                    (RETURN DONE)))
                [COND
                   ((LISTP (CAR COMS))
                    (SELECTQ (DELFROMCOM (CAR COMS)
                                     NAME TYPE)
                         (ALL (/RPLNODE2 COMS (CDR COMS))
                              (SETQQ DONE ALL)
                              (GO LP))
                         (NIL)
                         (SETQ DONE T)))
                   (T (SELECTQ (CAR COMS)
                          ((EVAL@LOADWHEN EVAL@COMPILEWHEN COPYWHEN)
                               (SETQ COMS (CDR COMS)))
                          (COND
                             ((AND (EQ TYPE 'VARS)
                                   (EQ NAME (CAR COMS)))
                              (/RPLNODE2 COMS (CDR COMS))
                              (SETQ DONE T)
                              (GO LP]
                (SETQ COMS (CDR COMS))
                (GO LP])
```

```
(DELFROMCOM
  [LAMBDA (COM NAME TYPE)                                  ; Edited  2-May-87 19:02 by Pavel
                                                           ; Tries to delete NAME from COM
    (PROG (TEM VAR NEW)
          (COND
             ((SETQ TEM (fetch DELETE of (CAR COM)))
              (AND (SETQ TEM (APPLY* TEM COM NAME TYPE))
                   (MARKASCHANGED COMSNAME 'VARS))
              (RETURN TEM)))
          (RETURN (SELECTQ (CAR COM)
                      ((DECLARE%: COMS)
                           (DELFROMCOMS (COND
                                              [(EQ (CADR COM)
                                                   '*)
                                               (COND
                                                  ((LITATOM (CADDR COM))
                                                   (CADDR COM))
                                                  (T (RETURN]
                                              (T (CDR COM)))
                                 NAME TYPE))
                      ((CL:EVAL-WHEN)
                           (DELFROMCOMS (COND
                                              [(EQ (CL:THIRD COM)
                                                   '*)
                                               (COND
                                                  ((LITATOM (CL:FOURTH COM))
                                                   (CL:FOURTH COM))
                                                  (T (RETURN]
                                              (T (CDDR COM)))
                                 NAME TYPE))
                      ((ALISTS PROPS)
                           (AND (EQ TYPE (CAR COM))
                                (COND
                                   ((EQ (CADR COM)
                                        '*)
                                    (COND
                                       ([AND (LITATOM (SETQ VAR (CADDR COM)))
                                             (SETQ TEM (ASSOC (CAR NAME)
                                                                (GETTOPVAL VAR)))
                                             (NEQ (CDR TEM)
                                                  (SETQ TEM (REMOVEITEM (CADR NAME)
                                                                     (CDR TEM]
                                        (SAVESET VAR TEM T 'NOPRINT)
                                        T)))
                                   ([AND [CDR (SETQ TEM (ASSOC (CAR NAME)
                                                                (CDR COM]
                                         (NEQ (CDR TEM)
                                              (SETQ NEW (REMOVEITEM (CADR NAME)
                                                                (CDR TEM]
                                    (/RPLACD TEM NEW)
                                    (MARKASCHANGED COMSNAME 'VARS)
                                    T))))
                      (BLOCKS ;; Remove function name from blocks declarations.  This isn't entirely correctly, since in removing the name
                              ;; from the block variables, it will hit homonyms in globalvars, specvars, etc.

                              [AND (EQ TYPE 'FNS)
                                   (for BLOCK in (INFILECOMTAIL COM T)
                                      do (AND (MEMB NAME BLOCK)
                                              (/DREMOVE NAME BLOCK))
                                         (for X in BLOCK when (AND (LISTP X)
                                                                   (MEMB NAME (CDR X)))
                                            do (/RPLACD X (REMOVE NAME (CDR X])
                      ((PROP IFPROP)
                           [SELECTQ TYPE
                               (PROPS (RETURN (COND
                                                  ((EQ (CADR COM)
                                                       (CADR NAME))
                                                   (DELFROMCOM1 (CDR COM)
                                                          (CAR NAME)))
                                                  ((AND (EQMEMB (CADR NAME)
                                                                (CADR COM))
                                                        [NULL (CDR (SETQ TEM (PRETTYCOM1 (CDR COM]
                                                        (EQ (CAR TEM)
                                                            (CAR NAME)))
                                                   [/RPLACA (CDR COM)
                                                          (REMOVE (CADR NAME)
                                                                 (MKLIST (CADR COM]
                                                   (MARKASCHANGED COMSNAME 'VARS)
                                                   T))))
                               (COND
                                  ([for PROP inside (CADR COM) always (EQ TYPE (GETPROP PROP 'PROPTYPE]
                                   (DELFROMCOM1 (CDR COM)
                                          NAME])
                      ((RECORDS INITRECORDS SYSRECORDS)
                           (AND (EQ TYPE 'RECORDS)
                                 (DELFROMCOM1 COM NAME)))
                       (P (AND (EQ TYPE ’EXPRESSIONS)
                               (DELFROMCOM1 COM NAME)))
                       ((VARS INITVARS)
                            (AND (EQ TYPE ’VARS)
                                 (DELFROMCOM1 COM NAME T)))
                       (AND (EQ TYPE (CAR COM))
                            (DELFROMCOM1 COM NAME])

(DELFROMCOM1
  [LAMBDA (COM NAME FLG)                                         (* rmk%: "10-JUN-82 22:44")
    ;; FLG is passed on to REMOVEITEM, determines whether lists whose CAR is NAME will be removed
    (LET (TEM VAL)
         (COND
            ((EQ (CADR COM)
                 ’*)
             (COND
                ([AND (LITATOM (SETQ TEM (CADDR COM)))
                      (NEQ (SETQ VAL (GETTOPVAL TEM))
                           (SETQ VAL (REMOVEITEM NAME VAL FLG]
                 (SAVESET TEM VAL T ’NOPRINT)
                 T)))
            ((NEQ (CDR COM)
                  (SETQ TEM (REMOVEITEM NAME (CDR COM)
                                   FLG)))
             (/RPLACD COM TEM)
             (MARKASCHANGED COMSNAME ’VARS)
             T])

(REMOVEITEM
  [LAMBDA (X LST FLG)                                            ; Edited 25-May-88 17:52 by drc:
                                                                 (* lmm "10-FEB-78 17:29")
    ;; returns a subset of LST with X deleted;  if FLG is set, also remove elements whose CAR is X
    (COND
       [[OR (MEMBER X LST)
            (AND FLG (SOME LST (FUNCTION (LAMBDA (Y)
                                            (EQUAL (CAR (LISTP Y))
                                                   X]
        (SUBSET LST (FUNCTION (LAMBDA (Y)
                                 (AND (NOT (EQUAL Y X))
                                      (OR (NOT FLG)
                                          (NLISTP Y)
                                          (NOT (EQUAL (CAR Y)
                                                      X]
       (T LST])

(MOVETOFILE
  [LAMBDA (TOFILE NAME TYPE FROMFILE)                            (* rmk%: "18-OCT-79 19:51")
                                                                 ; To move items between files
    (SETQ TYPE (GETFILEPKGTYPE TYPE))
    [COND
       ((OR (EQ TYPE ’FNS)
            FROMFILE)                                            ; FNS definition can reside on file if LOADFNS was done.  This
                                                                 ; guarantees that it is loaded.
        (PUTDEF NAME TYPE (GETDEF NAME TYPE FROMFILE ’(NOCOPY NODWIM]
    (AND (EQ TYPE ’FNS)
         (MARKASCHANGED NAME TYPE))                              ; FNS won’t get dumped unless they are ‘changed’
    (DELFROMFILES NAME TYPE FROMFILE)
    (ADDTOFILE NAME TYPE TOFILE])
)

(MOVD? ’DELFROMFILES ’DELFROMFILE NIL T)

(MOVD? ’MOVETOFILE ’MOVEITEM NIL T)

(ADDTOVAR SYSPROPS PROPTYPE VARTYPE)

;; functions for doing things and marking them changed and auxiliary functions

(DEFINEQ

(SAVEPUT
  [LAMBDA (ATM PROP VAL)                                         (* lmm " 7-May-84 16:56")
    ;; analogous to SAVESET but also marks changed property lists;  LISPXFNS are marked to change PUT and PUTPROP to SAVEPUT
    [COND
       ((NOT (LITATOM ATM))
        (ERRORX (LIST 14 ATM]
    (PROG ((X (GETPROPLIST ATM))
           X0 TEM OLDFLG)
      LOOP
```

```
          (COND
             ((NLISTP X)
              (COND
                 ((AND (NULL X)
                       X0)                                       ; typical case.  property list ran out on an even parity position.
                                                                 ; e.g.  (A B C D)
                  (SETQ TEM (LIST PROP VAL))
                  (AND LISPXHIST (UNDOSAVE (LIST '/PUT-1 ATM TEM)
                                          LISPXHIST))
                  (FRPLACD (CDR X0)
                         TEM)
                  (GO RET)))
              ;; property list was initially NIL or a non-list, or else it ended in a non-list following an even parity position, e.g. (A B . C) fall through
              ;; and add new property at beginning

              )
             ((NLISTP (CDR X))
              ;; property list runs out on an odd parity, or ends in an odd list following an odd parity, e.g.  (A B C) or (A B C . D) fall through and add
              ;; at beginning.

              )
             [(EQ (CAR X)
                  PROP)
              (SETQ OLDFLG (NEQ (EQUALN (CADR X)
                                       VAL 400)
                                T))                              ; i.e. it probably changed
              (/RPLACA (CDR X)
                     VAL)
              (COND
                 ((NOT OLDFLG)
                  (GO RET1))
                 (T (OR (EQ DFNFLG T)
                        (LISPXPRINT (LIST 'new PROP 'property 'for ATM)
                               T T))
                    (GO RET]
             (T (SETQ X (CDDR (SETQ X0 X)))
                (GO LOOP)))
          [SETQ TEM (CONS PROP (CONS VAL (GETPROPLIST ATM]
          (SETPROPLIST ATM TEM)
          (AND LISPXHIST (UNDOSAVE (LIST '/PUT-1 ATM TEM)
                                LISPXHIST))
      RET (MARKASCHANGED (LIST ATM PROP)
                 'PROPS
                 (NOT OLDFLG))
      RET1
          (AND ADDSPELLFLG (ADDSPELL ATM 0))
          (RETURN VAL])

)

(DECLARE%: DONTEVAL@LOAD DOCOPY

(OR (CHANGENAME 'PUTPROPS 'PUTPROP 'SAVEPUT)
    (CHANGENAME 'PUTPROPS '/PUT 'SAVEPUT))
)

(DEFINEQ

(UNMARKASCHANGED
  [LAMBDA (NAME TYPE)                                            (* JonL "24-Jul-84 19:59")
    ;; says to remove NAME from TYPE's changedlst, and also to remove it from any FILE properties.  Value is name if anything is done

    (PROG (ANYFLG)
          (bind TAIL [CHANGED _ (fetch CHANGED of (SETQ TYPE (GETFILEPKGTYPE TYPE 'TYPE]
             while (SETQ TAIL (MEMBER NAME CHANGED)) do (/RPLACA TAIL)
                                                        (SETQ ANYFLG T))
          [for F TAIL PROP TYPEDPROP in FILELST
             when [SETQ TAIL (MEMBER NAME (CDR (SETQ TYPEDPROP (ASSOC TYPE (fetch TOBEDUMPED
                                                                               of (SETQ PROP
                                                                                     (fetch FILEPROP of F]
             do (SETQ ANYFLG T)
                (COND
                   ((SETQ TAIL (REMOVE (CAR TAIL)
                                      (CDR TYPEDPROP)))
                    (/RPLACD TYPEDPROP TAIL))
                   (T (/replace TOBEDUMPED of PROP with (REMOVE TYPEDPROP (fetch TOBEDUMPED of PROP]
          (RETURN (AND ANYFLG NAME])

(PREEDITFN
  [LAMBDA (ATM TYPE EDITCHANGES)                                 (* rmk%: "18-FEB-82 21:49")
                                                                 ; EDITL is advised to call this before editing something
    (AND FILEPKGFLG (SELECTQ TYPE
                        (PROPLST [AND (OR CLISPARRAY (PROGN (CLISPTRAN (CONS)
                                                                     (CONS))
                                                             CLISPARRAY))
                                       (for X in (GETPROPLIST ATM) do (OR (NLISTP X)
                                                                         (GETHASH X CLISPARRAY)
                                                                         (PUTHASH X (CONS (CAR X)
                                                                                           (CDR X))
                                                                                 CLISPARRAY]

                                  ;; note that if CLISPARRAY is disabled that ALL properties of an edited prop list will get marked as
                                  ;; changed if any destructive edit is made

                                  [RESETSAVE NIL (LIST (FUNCTION POSTEDITPROPS)
                                                       EDITCHANGES
                                                       (APPEND (GETPROPLIST ATM])
                         (VARS [COND
                                  ((EQMEMB 'ALIST (GETPROP ATM 'VARTYPE))
                                   [AND (OR CLISPARRAY (PROGN (CLISPTRAN (CONS)
                                                                         (CONS))
                                                              CLISPARRAY))
                                        (for X in (EVALV ATM) do (OR (NLISTP X)
                                                                     (GETHASH X CLISPARRAY)
                                                                     (PUTHASH X (CONS (CAR X)
                                                                                      (CDR X))
                                                                            CLISPARRAY]
                                   (RESETSAVE NIL (LIST (FUNCTION POSTEDITALISTS)
                                                        EDITCHANGES
                                                        (for X in (EVALV ATM) collect (CAR X])
                         NIL])

(POSTEDITPROPS
  [LAMBDA (EDITCHANGES OLDPROPS)                                  (* rmk%: "18-FEB-82 21:50")
                                                                  ; was RESETSAVE'd from PREEDITFN
    (PROG (OV FOUNDCHANGE)
          (OR FILEPKGFLG (RETURN))
          (COND
             ((CADR EDITCHANGES)
              (for NEWPROP on (GETPROPLIST (CAR EDITCHANGES)) by (CDDR NEWPROP)
                 when (for OLDPROP on OLDPROPS by (CDDR OLDPROP)
                         do (COND
                               ((EQ (CAR OLDPROP)
                                    (CAR NEWPROP))                ; Found the property
                                [AND (EQ (CADR OLDPROP)
                                         (CADR NEWPROP))
                                     (COND
                                        ((NLISTP (CADR OLDPROP))  ; value is same
                                         (RETURN))
                                        ((AND CLISPARRAY (SETQ OV (GETHASH (CADR NEWPROP)
                                                                          CLISPARRAY))
                                              (EQ (CAADR NEWPROP)
                                                  (CAR OV))
                                              (EQ (CDADR NEWPROP)
                                                  (CDR OV)))
                                         (PUTHASH (CADR NEWPROP)
                                                NIL CLISPARRAY)   ; value has been edited (CLISPARRAY translation went away)
                                         (RETURN]
                                (RETURN T)))
                         finally                                  ; didn't find the property
                               (RETURN T))
                 do (MARKASCHANGED (LIST (CAR EDITCHANGES)
                                         (CAR NEWPROP))
                           'PROPS NIL)
                    (SETQ FOUNDCHANGE T))
              (AND FOUNDCHANGE (RPLACA (CDR EDITCHANGES)
                                       NIL])

(POSTEDITALISTS
  [LAMBDA (EDITCHANGES OLDTOKENS)                                 (* rmk%: " 4-JAN-82 10:14")
    (PROG [OV FOUNDCHANGE (NEWENTRIES (GETTOPVAL (CAR EDITCHANGES]
                                                                  ; called after an ALIST has been edited
          (OR FILEPKGFLG (RETURN))
          (COND
             ((CADR EDITCHANGES)
              (for X in OLDTOKENS when (NOT (FASSOC X NEWENTRIES)) do (MARKASCHANGED (LIST (CAR EDITCHANGES)
                                                                                            X)
                                                                             'ALISTS NIL)
                                                                      (SETQ FOUNDCHANGE T))
              [for NEWENTRY in NEWENTRIES do (COND
                                                ([AND (LISTP NEWENTRY)
                                                      (NOT (AND CLISPARRAY (SETQ OV (GETHASH NEWENTRY CLISPARRAY)
                                                                                    )
                                                                (EQ (CAR NEWENTRY)
                                                                    (CAR OV))
                                                                (EQ (CDR NEWENTRY)
                                                                    (CDR OV]
                                                 (PUTHASH NEWENTRY NIL CLISPARRAY)
                                                 (MARKASCHANGED (LIST (CAR EDITCHANGES)
                                                                        (CAR NEWENTRY))
                                                       'ALISTS NIL)
                                                 (SETQ FOUNDCHANGE T]
              (AND FOUNDCHANGE (RPLACA (CDR EDITCHANGES)
                                      NIL])

)

(ADDTOVAR LISPXFNS (PUT . SAVEPUT)
                   (PUTPROP . SAVEPUT))

;; sub-functions for file package commands & types

(DEFINEQ

(ALISTS.GETDEF
  [LAMBDA (NAME TYPE OPTIONS)                                    (* Pavel " 7-Oct-86 17:24")
    (AND (LISTP NAME)
         (CL:SYMBOLP (CAR NAME))
         (LET [(ASSOCIATION (ASSOC (CADR NAME)
                                   (GETTOPVAL (CAR NAME]
              (AND ASSOCIATION (LIST 'ADDTOVAR (CAR NAME)
                                     ASSOCIATION])

(ALISTS.WHENCHANGED
  [LAMBDA (NAME TYPE NEWFLG)                                     (* lmm "16-OCT-78 20:02")
                                                                 ; called by MARKASCHANGED when an ALIST entry has
                                                                 ; changed
    (PROG [(VARTYPE (GETPROP (CAR NAME)
                           'VARTYPE]
          (AND (LISTP VARTYPE)
               (EQ (CAR VARTYPE)
                   'ALIST)
               (RETFROM 'MARKASCHANGED (MARKASCHANGED (CADR NAME)
                                              (CADR VARTYPE)
                                              NEWFLG])

(CLEARCLISPARRAY
  [LAMBDA (NAME TYPE REASON)
    (DECLARE (SPECVARS NAME TYPE REASON))                        (* lmm "14-Aug-84 15:03")
    (AND CLISPARRAY (MAPHASH CLISPARRAY (COND
                                          [(EQ TYPE 'I.S.OPRS)
                                            (FUNCTION (LAMBDA (TRAN FORM)
                                                        (AND (MEMB NAME FORM)
                                                             (PUTHASH FORM NIL CLISPARRAY]
                                           (T                    ; MACRO changed
                                              (FUNCTION (LAMBDA (TRAN FORM)
                                                          (COND
                                                             ((OR (EQ NAME (CAR FORM))
                                                                  (EQ (CAR (GETPROP (CAR FORM)
                                                                                   'CLISPWORD))
                                                                      'CHANGETRAN))
                                                              (PUTHASH FORM NIL CLISPARRAY])

(EXPRESSIONS.WHENCHANGED
  [LAMBDA (EXPR)                                                 ; Edited  6-Apr-87 20:21 by Pavel
    (SELECTQ (CAR EXPR)
        ((SETQ SETQQ)
             (UNMARKASCHANGED (CADR EXPR)
                   'VARS))
        ((PROGN PROG)
             (for X in (CDR EXPR) do (EXPRESSIONS.WHENCHANGED X)))
        NIL])

(MAKEALISTCOMS
  [NLAMBDA X                                                     (* rmk%: "14-OCT-83 13:34")

    ;; make command to dump prettydefmacros

    (LIST (CONS 'ADDVARS
                (for PR in X
                   join (for ALISTNAME inside (CAR PR)
                          collect (CONS ALISTNAME (for ATNAME inside (CDR PR) bind ENTRY
                                                     when (SETQ ENTRY (OR (SASSOC ATNAME (GETTOPVAL ALISTNAME))
                                                                          (PROGN (LISPXPRINT (LIST 'no ATNAME
                                                                                                   'entry
                                                                                                   'on ALISTNAME)
                                                                                        T T)
                                                                                 NIL)))
                                                     collect ENTRY])

(MAKEFILESCOMS
  [NLAMBDA FILES                                                 (* JonL "12-FEB-83 19:02")

    ;; This scans the command just to warn the user about any errors.  Must match up with the big SELECTQ in FILESLOAD NIL
    [for FILE in FILES
       do (OR (LITATOM FILE)
              (while (LISTP FILE)
                 do (SELECTQ (CAR (OR (LISTP FILE)
                                      (RETURN)))
                         ((LOADCOMP LOADFROM))
                         (FROM (pop FILE)
                               (if (OR (EQ (CAR FILE)
                                           'VALUEOF)
                                      (if (AND (EQ (CAR FILE)
                                                   'VALUE)
                                               (EQ (CADR FILE)
                                                   'OF))
                                          then (pop FILE)))
                                   then (pop FILE)))
                         ((COMPILED LOAD EXTENSION EXT SOURCE SYMBOLIC IMPORT NOERROR))
                         (OR (FMEMB (CAR FILE)
                                    LOADOPTIONS)
                             (PRINT (CONS (CAR FILE)
                                          '(-- unrecognized FILES option))
                                    T)))
                    (pop FILE]
    (CONS 'FILESLOAD FILES])
```

(MAKELISPXMACROSCOMS

```
  [NLAMBDA X                                                       (* lmm " 5-SEP-78 23:15")
    (PROG (TEM TEM2)
          (RETURN (CONS [CONS 'ALISTS (SETQ TEM (NCONC (AND [SETQ TEM (SUBSET X (FUNCTION (LAMBDA (Z)
                                                                                             (FASSOC Z
                                                                                               LISPXHISTORYMACROS
                                                                                                    ]
                                                             (LIST (CONS 'LISPXHISTORYMACROS TEM)))
                                                        (AND [SETQ TEM (SUBSET X (FUNCTION (LAMBDA (Z)
                                                                                             (FASSOC Z
                                                                                                    LISPXMACROS]
                                                             (LIST (CONS 'LISPXMACROS TEM]
                        (SETQ TEM2 (NCONC [AND [SETQ TEM2 (SUBSET X (FUNCTION (LAMBDA (Z)
                                                                                 (FMEMB Z LISPXCOMS]
                                               (LIST (LIST 'ADDVARS (CONS 'LISPXCOMS TEM2]
                                          (AND [SETQ TEM2 (SUBSET X (FUNCTION (LAMBDA (Z)
                                                                                 (FMEMB Z HISTORYCOMS]
                                               (LIST (LIST 'ADDVARS (CONS 'HISTORYCOMS TEM2])
```

(MAKEPROPSCOMS

```
  [NLAMBDA X                                                       (* lmm "26-FEB-78 17:10")

    ;; make command to dump PROPS

    (for PAIR in X collect (CONS 'PROP (CONS (COND
                                                 ((AND (LISTP (CDR PAIR))
                                                       (NULL (CDDR PAIR)))
                                                  (CADR PAIR))
                                                 (T (CDR PAIR)))
                                              (OR (LISTP (CAR PAIR))
                                                  (LIST (CAR PAIR])
```

(MAKEUSERMACROSCOMS

```
  [NLAMBDA X                                                       (* rmk%: " 3-JAN-82 23:20")
    (PROG (TEM)
          [COND
             [X (for Y in X do (OR (FASSOC Y USERMACROS)
                                  (FASSOC Y EDITMACROS)
                                  (LISPXPRINT (CONS Y '(-- no entry on USERMACROS))
                                         T T]
             (T (SETQ X (INTERSECTION (SETQ X (MAPCAR USERMACROS 'CAR))
                               X]
          (RETURN (LIST (CONS 'ADDVARS (NCONC (for VAR in '(USERMACROS EDITMACROS)
                                                 when (SETQ TEM (for Y in (GETTOPVAL VAR)
                                                                   when (FMEMB (CAR Y)
                                                                               X)
                                                                   collect Y))
                                                 collect (CONS VAR TEM))
                                              (for LST in '(EDITCOMSA EDITCOMSL COMPACTHISTORYCOMS
                                                                  DONTSAVEHISTORYCOMS)
                                                 when [SETQ TEM (SUBSET (GETTOPVAL LST)
                                                                       (FUNCTION (LAMBDA (Y)
                                                                                   (OR
                                                                                    (FMEMB Y X)
                                                                                    (AND (LISTP Y)
                                                                                         (FMEMB (CAR Y)
                                                                                                X]
                                                 collect (CONS LST TEM])
```

(PROPS.WHENCHANGED

```
  [LAMBDA (NAME TYPE NEWFLG)                                     (* lmm " 7-SEP-78 22:08")
    (PROG [(PROPTYPE (GETPROP (CADR NAME)
                            'PROPTYPE]
          (COND
             [PROPTYPE (RETFROM 'MARKASCHANGED (COND
                                                   ((NEQ PROPTYPE 'IGNORE)
                                                    (MARKASCHANGED (CAR NAME)
                                                           PROPTYPE NEWFLG]
             (T (SELECTQ (CADR NAME)
                     (CLISPWORD (CLEARCLISPARRAY (CAR NAME)))
                     NIL])

(FILEGETDEF.LISPXMACROS
  [LAMBDA (NAME TYPE SOURCE OPTIONS)                             (* lmm " 4-Jul-85 15:12")
    (MKPROGN (for X in [LOADFNS NIL SOURCE 'GETDEF '(LAMBDA (FIRST SECOND)
                                                      (AND (EQ FIRST 'ADDTOVAR)
                                                           (MEMB SECOND '(LISPXMACROS LISPXCOMS))
                                                          T]
                when (SELECTQ (CADR X)
                          (LISPXMACROS                           ; Rebuild the expressions cause there might be other elements in
                                                                 ; the ADDTOVAR
                                        (AND (SETQ X (ASSOC NAME (CDDR X)))
                                             (SETQ X (LIST 'ADDTOVAR 'LISPXMACROS X))))
                          (LISPXCOMS [COND
                                         ((MEMB NAME (CDDR X))
                                          (SETQ X (LIST 'ADDTOVAR 'LISPXCOMS NAME)))
                                         ((SETQ X (ASSOC NAME (CDDR X)))
                                                                 ; For synonym pairs
                                          (SETQ X (LIST 'ADDTOVAR 'LISPXCOMS X])
                          NIL)
                collect X])

(FILEGETDEF.ALISTS
  [LAMBDA (NAME TYPE SOURCE OPTIONS)                             (* lmm " 4-Jul-85 15:13")
    (for X in [LOADFNS NIL SOURCE 'GETDEF '(LAMBDA (FIRST SECOND)
                                             (AND (EQ FIRST 'ADDTOVAR)
                                                  (EQ SECOND (CAR NAME]
       when (SETQ X (ASSOC (CADR NAME)
                           (CDDR X)))
       collect X finally (RETURN (COND
                                   ($$VAL (CONS 'ADDTOVAR (CONS (CAR NAME)
                                                                $$VAL])

(FILEGETDEF.RECORDS
  [LAMBDA (NAME TYPE SOURCE OPTIONS NOTFOUND)                    (* lmm "26-Jun-86 15:56")
    (LET [(VAL (LOADFNS NIL SOURCE 'GETDEF '(LAMBDA (FIRST SECOND)
                                               (AND (MEMB FIRST CLISPRECORDTYPES)
                                                    (OR (EQ SECOND NAME)
                                                        (AND (MEMB SECOND '(%( %[))
                                                             (PROGN (RATOM)
                                                                    (RATOM)
                                                                    (RATOM)
                                                                    (EQ NAME (RATOM]
         (if (EQ (CAAR VAL)
                 'NOT-FOUND%:)
             then NOTFOUND
           elseif (CDR VAL)
             then (CONS 'PROGN VAL)
           else (CAR VAL])

(FILEGETDEF.PROPS
  [LAMBDA (NAME TYPE SOURCE OPTIONS)                             (* lmm " 4-Jul-85 15:13")
    (for X in [LOADFNS NIL SOURCE 'GETDEF '(LAMBDA (FIRST SECOND)
                                             (AND (EQ FIRST 'PUTPROPS)
                                                  (EQ SECOND (CAR NAME]
       join (for TAIL on (CDDR X) by (CDDR TAIL) when (EQ (CAR TAIL)
                                                         (CADR NAME))
              join (LIST (CAR TAIL)
                        (CADR TAIL)))
       finally (RETURN (COND
                        ($$VAL (CONS 'PUTPROPS (CONS (CAR NAME)
                                                      $$VAL])

(FILEGETDEF.MACROS
  [LAMBDA (NAME TYPE SOURCE OPTIONS)                             (* lmm "28-May-86 09:51")
    (MKPROGN (for X in [LOADFNS NIL SOURCE 'GETDEF '(LAMBDA (FIRST SECOND)
                                                      (AND (FMEMB FIRST '(PUTPROPS DEFMACRO ))
                                                           (EQ SECOND NAME]
                join (if (EQ (CAR X)
                           'DEFMACRO)
                        then (LIST X)
                    else (for TAIL on (CDDR X) by (CDDR TAIL) when (FMEMB (CAR TAIL)
                                                                            MACROPROPS)
                             collect (LIST 'PUTPROPS (CADR X)
                                           (CAR TAIL)
                                           (CADR TAIL])

(FILEGETDEF.VARS
  [LAMBDA (NAME TYPE SOURCE OPTIONS)                              (* lmm " 4-Jul-85 15:14")
    (for X in (LOADFNS NIL SOURCE 'GETDEF NAME) do (SELECTQ (CAR X)
                                                       ((RPAQQ SETQQ )
                                                            (RETURN (CADDR X)))
                                                       ((RPAQ SETQ RPAQ?)
                                                            (RETURN (EVAL (CADDR X))))
                                                       NIL)
       finally (RETURN 'NOBIND])

(FILEGETDEF.FNS
  [LAMBDA (NAME TYPE SOURCE OPTIONS)                              (* bvm%: "29-Aug-86 22:30")
    (LET (MAP ENV)
         (COND
            [(AND (EQMEMB 'FAST OPTIONS)
                  (PROGN (CL:MULTIPLE-VALUE-SETQ (ENV MAP)
                                (GET-ENVIRONMENT-AND-FILEMAP SOURCE))
                         MAP))
             (for PAIR MAPLOC in (CDR MAP) when [SETQ MAPLOC (CADR (ASSOC NAME (CDDR PAIR]
                do [OR (OPENP SOURCE)
                       (RESETSAVE NIL (LIST 'CLOSEF? (SETQ SOURCE (OPENSTREAM SOURCE 'INPUT 'OLD]
                   (SETFILEPTR SOURCE MAPLOC)
                   (RETURN (WITH-READER-ENVIRONMENT ENV
                                  [COND
                                     ((EQMEMB 'ARGLIST OPTIONS)
                                      (RATOM SOURCE)
                                      (READ SOURCE)
                                      (RATOM SOURCE)
                                      (LIST (READ SOURCE)
                                            (READ SOURCE)))
                                     (T (CADR (READ SOURCE])]
            (T (CADR (FASSOC NAME (LOADEFS NAME SOURCE])

(FILEPKGCOMS.PUTDEF
  [LAMBDA (NAME TYPE DEFINITION REASON)                           (* lmm "15-Jul-85 11:29")
    (PROG (COM TYP)
          [SELECTQ (CAR (LISTP DEFINITION))
               (COM (SETQ COM (CDR DEFINITION)))
               (TYPE (SETQ TYP (CDR DEFINITION)))
               (PROGN (SETQ COM (CDR (ASSOC 'COM DEFINITION)))
                      (SETQ TYP (CDR (ASSOC 'TYPE DEFINITION]

     ;; Check properties first, so that we don't smash some and then get an error in a later call to FILEPKGCOM/TYPE

          (for I in COM by (CDDR I) do (SELECTQ I
                                           ((ADD DELETE MACRO CONTENTS CONTAIN COM))
                                           (ERROR I "not file package command property")))
                                                                  ; COM merely adds to spelling list, for builtins
          [FILEPKGCOM NAME 'CONTENTS (OR (LISTGET COM 'CONTENTS)
                                           (LISTGET COM 'CONTAIN] ; Until CONTAIN is de-documented.
          (for PROP in '(ADD DELETE MACRO COM) do (FILEPKGCOM NAME PROP (LISTGET COM PROP)))
          [for I in TYP by (CDDR I) do (OR (FMEMB I FILEPKGTYPEPROPS)
                                          (SELECTQ I
                                               ((DESCRIPTION TYPE))
                                               (ERROR I "not file package type/command property"]
                                                                  ; TYPE merely adds to spelling list, for builtins
          (for PROP in (UNION '(DESCRIPTION TYPE)
                              FILEPKGTYPEPROPS)
             do (FILEPKGTYPE NAME PROP (LISTGET TYP PROP])

(FILES.PUTDEF
  [LAMBDA (NAME TYPE DEFINITION REASON)                           (* lmm "15-Jul-85 17:13")
    (PROGN (PUTDEF (FILECOMS NAME)
                  'VARS
                  (CAR DEFINITION)
                  REASON)                                         ; DEFINE THE COMS
           (ADDFILE NAME)                                         ; MAKE SURE IT IS A FILE PACKAGE ENTITY
           [/replace TOBEDUMPED of (fetch FILEPROP of NAME)
                                  (FILEPKG.MERGECHANGES (CADR DEFINITION)
                                         (fetch TOBEDUMPED of (fetch FILEPROP of NAME]
           (OR (fetch FILEDATES of NAME)
               (/replace FILEDATES of NAME with (CADDR DEFINITION])

(VARS.PUTDEF
  [LAMBDA (NAME TYPE DEFINITION REASON)                           (* lmm "29-Jul-85 20:59")
    (/SETTOPVAL NAME DEFINITION T])
```

```
(FILES.WHENCHANGED
  [LAMBDA (NAME TYPE REASON)
    (MARKASCHANGED (FILECOMS NAME)
           'VARS REASON])

)

(ADDTOVAR MACROPROPS MACRO BYTEMACRO DMACRO)

(ADDTOVAR SYSPROPS PROPTYPE)

(PUTPROPS I.S.OPR PROPTYPE I.S.OPRS)

(PUTPROPS SUBR PROPTYPE IGNORE)

(PUTPROPS LIST PROPTYPE IGNORE)

(PUTPROPS CODE PROPTYPE IGNORE)

(PUTPROPS FILEDATES PROPTYPE IGNORE)

(PUTPROPS FILE PROPTYPE IGNORE)

(PUTPROPS FILEMAP PROPTYPE IGNORE)

(PUTPROPS EXPR PROPTYPE FNS)

(PUTPROPS VALUE PROPTYPE VARS)

(PUTPROPS COPYRIGHT PROPTYPE FILES)

(PUTPROPS FILETYPE PROPTYPE FILES)

(PUTPROPS BAKTRACELST VARTYPE ALIST)

(PUTPROPS BREAKMACROS VARTYPE ALIST)

(PUTPROPS COMPILETYPELST VARTYPE ALIST)

(PUTPROPS EDITMACROS VARTYPE (ALIST USERMACROS))

(PUTPROPS ERRORTYPELST VARTYPE ALIST)

(PUTPROPS FONTDEFS VARTYPE ALIST)

(PUTPROPS LISPXHISTORYMACROS VARTYPE (ALIST LISPXMACROS))

(PUTPROPS LISPXMACROS VARTYPE (ALIST LISPXMACROS))

(PUTPROPS PRETTYDEFMACROS VARTYPE (ALIST FILEPKGCOMS))

(PUTPROPS PRETTYEQUIVLST VARTYPE ALIST)

(PUTPROPS PRETTYPRINTMACROS VARTYPE ALIST)

(PUTPROPS PRETTYPRINTYPEMACROS VARTYPE ALIST)

(PUTPROPS USERMACROS VARTYPE (ALIST USERMACROS))

;; Define the commands below AFTER the various properties have been established.

(ADDTOVAR USERMACROS
          (M (X . Y)
             (E (MARKASCHANGED (COND ((LISTP 'X)
                                      (CAR 'X))
                                     (T 'X))
                       'USERMACROS)
                T)
             (ORIGINAL (M X . Y)))
          (M NIL (MAKE FILE FILE)))

(ADDTOVAR EDITMACROS
          (M (X . Y)
             (E (MARKASCHANGED (COND ((LISTP 'X)
                                      (CAR 'X))
                                     (T 'X))
                       'USERMACROS)
                T)
             (ORIGINAL (M X . Y))))

(ADDTOVAR EDITCOMSA M)

(ADDTOVAR EDITCOMSL M)

;; GETDEF methods
```

```
(DEFINEQ

(RENAME
  [LAMBDA (OLD NEW TYPES FILES METHOD)                              (* JonL "24-Jul-84 20:01")
    (PROG ((TYPES (GETFILEPKGTYPE TYPES 'TYPE NIL OLD)))

     ;; special kludge: change the callers BEFORE if we are changing a field;  this is so the CHANGECALLERS won't get an UNABLE TO DWIMIFY
     ;; message

          [for TYPE inside TYPES when (NEQ TYPE 'FIELDS) do (COPYDEF OLD NEW TYPE NIL (COND
                                                                                             ((EQ TYPE 'VARS)
                                                                                              'NOERROR]
          (CHANGECALLERS OLD NEW TYPES FILES METHOD)
          [for TYPE inside TYPES do (COND
                                       ((AND (EQ TYPE 'FIELDS)
                                             (HASDEF OLD 'FIELDS))

                                       ;; The HASDEF test is because the rename might already have been done in EDITFROMFILE in the
                                       ;; CHANGECALLERS, if it found a record with the field on a file.  Otherwise, COPYDEF essentially
                                       ;; will just do the necessary substitution in the existing record declarations, given that definitions for
                                       ;; FIELDS are mutually exclusive.

                                        (COPYDEF OLD NEW 'FIELDS))
                                       (T (DELDEF OLD TYPE]
          (RETURN NEW])

(CHANGECALLERS
  [LAMBDA (OLD NEW AS-TYPES FILES METHOD)                           ; Edited  6-Dec-86 01:25 by lmm
    (PROG ((AS-TYPES (GETFILEPKGTYPE AS-TYPES))
           REL TEM EDITCOMS FNS)
          (OR METHOD (SETQ METHOD DEFAULTRENAMEMETHOD))
          [SETQ EDITCOMS
           (LIST (COND
                    [(OR (EQMEMB 'CAREFUL METHOD)
                         (PROGN (SETQ TEM (TYPESOF OLD NIL AS-TYPES))
                                (printout T "Warning --" OLD " is also defined as " TEM T)))

                     ;; This creates a 'command' that searches like EXAM, but interrogates the user about whether to do the Rename.  Y means
                     ;; do it, No means skip, anything else goes into TTY.

                     (SUBPAIR '(OLD NEW)
                            (LIST OLD NEW)
                            '(BIND (LPQ (F OLD N)
                                        (MARK %#1)
                                        (ORR (1 !0 P)
                                             NIL)
                                        (MARK %#2)
                                        (COMS (SELECTQ (ASKUSER NIL NIL "   Replace ? " '((Y "Yes
                                                                                               ")
                                                                                            (N "No
                                                                                               ")
                                                                                            (%
"")
                                                                                            (%  "")
                                                                                            (%
"")
                                                                                            (& ""))
                                                                 NIL NIL '(NOECHOFLG T))
                                                    (Y '(R1 OLD NEW))
                                                    (N NIL)
                                                    'TTY%:))
                                        (MARK %#3)
                                        (IF (EQ (%## (\ %#3))
                                                (%## (\ %#2)))
                                            ((\ %#1))
                                            NIL]
                    (T (LIST 'R OLD NEW]
          (SELECTQ (COND
                      ((AND (EQMEMB 'MASTERSCOPE METHOD)
                            MSDATABASELST
                            (for TYPE inside AS-TYPES do [COND
                                                              ((SETQ TEM (SELECTQ TYPE
                                                                                ((FNS FUNCTIONS SPECIAL-FORMS
                                                                                      OPTIMIZERS)
                                                                                     'CALL)
                                                                                (MACROS '(CALL DIRECTLY))
                                                                                ((VARS VARIABLES)
                                                                                     '(USE OR BIND))
                                                                                ((RECORDS FIELDS I.S.OPRS)
                                                                                     (LIST 'USE 'AS TYPE))
                                                                                (RETURN NIL)))
                                                               (COND
                                                                  (REL (SETQ REL (LIST TEM 'OR REL)))
                                                                  (T (SETQ REL TEM]
                               finally (RETURN REL)))

                       ;; can only use masterscope if (a) we say to, (b) something's been analyzed, and (c) the types the function is are known

                       'MASTERSCOPE)
                      ((EQMEMB 'EDITCALLERS METHOD)
                       'EDITCALLERS)
                      (T 'SEARCH))
               (MASTERSCOPE (MAPC [SETQ FNS (NCONC [COND
                                                      ((NULL FILES)
                                                       (UPDATEFILES)
                                                       (FILEPKGCHANGES 'FNS]
                                                   (for FILE inside (OR FILES FILELST) join (FILEPKGFNSLST FILE]
                                  (FUNCTION UPDATEFN))
                            (SETQ FNS (INTERSECTION (GETRELATION OLD (SETQ REL (PARSERELATION REL))
                                                               T)
                                            FNS)))
               (EDITCALLERS (SETQ FILES (for X inside (OR FILES FILELST) when (SETQ TEM (EDITCALLERS OLD X T))
                                          collect (PROGN (SETQ FNS (NCONC FNS (CDR TEM)))
                                                         X))))
               (SEARCH (SETQ FNS (for X inside (OR FILES FILELST) join (FILEFNSLST X))))
               (ERROR "UNRECOGNIZED RENAME METHOD" METHOD))
          (AND (EQMEMB 'FNS AS-TYPES)
               (FMEMB OLD FNS)
               (SETQ FNS (REMOVE OLD FNS)))
          (EDITFROMFILE FNS FILES OLD EDITCOMS)
          [for TYPE inside AS-TYPES do (for FILE in (WHEREIS OLD TYPE FILES)
                                        do (AND (ADDTOFILE NEW TYPE FILE)
                                                (DELFROMFILES OLD TYPE FILE)
                                                (printout T OLD " changed to " NEW " on " FILE)))
                                     (COND
                                        ((SETQ TEM (WHEREIS OLD TYPE FILES))
                                         (printout T "Couldn't change " OLD " to " NEW " as " TYPE " on " TEM]
          (COND
             (REL (UPDATECHANGED)
                  (COND
                     ((AND (SETQ TEM (GETRELATION OLD REL T))
                           (WHEREIS TEM 'FNS FILES))
                      (printout T "Couldn't find where " OLD " is referenced in " TEM T])

)

(DEFINEQ

(SHOWDEF
  [LAMBDA (NAME TYPE FILE)                                      ; Edited 26-Oct-2021 09:21 by rmk:
                                                                ; Edited 16-Apr-2018 21:35 by rmk:
                                                                ; prettyprint NAME as it would be dumped as a TYPE (in the
                                                                ; current reader environment)
    (RESETLST
        (PROG (ORIGFLG FNSLST FL PRETTYCOMSLST NEWFILEMAP (SOURCEFILENV (MAKE-READER-ENVIRONMENT
                                                                               *DEFAULT-MAKEFILE-ENVIRONMENT*)))
              (DECLARE (SPECVARS . T))
              [AND FILE (NEQ FILE (OUTPUT))
                   (if (SETQ FL (OPENP FILE 'OUTPUT))
                       then (RESETSAVE (OUTPUT FL))
                     else (OUTFILE FILE)
                          (RESETSAVE NIL (LIST (FUNCTION CLOSEF?)
                                               (OUTPUT]
              (PRETTYCOM (MAKENEWCOM NAME TYPE))))])

(COPYDEF
  [LAMBDA (OLD NEW TYPE SOURCE OPTIONS)                         (* lmm "14-Aug-84 18:38")
                                                                ; like MOVD, but takes a type.
    (PROG (TEM DEF)
          (SETQ TYPE (GETFILEPKGTYPE TYPE 'TYPE))
          [SETQ DEF (GETDEF OLD TYPE SOURCE (COND
                                                ((EQ OPTIONS 'NOCOPY)
                                                 NIL)
                                                (T (REMOVE 'NOCOPY (MKLIST OPTIONS]
                                                                ; The default is for GETDEF to return a COPY.  Make sure that
                                                                ; NOCOPY isn't in options though.
          (SELECTQ TYPE
               (VARS)
               (FILES [for X in (CAR DEF) do                    ; change all the listnames which are of form filenameTYPE
                                           (SELECTQ (CAR X)
                                                ((PROP IFPROP)
                                                    (SETQ X (CDR X)))
                                                NIL)
                                           (COND
                                              ((EQ (CADR X)
                                                   '*)
                                               (SETQ X (CDDR X))
                                               (COND
                                                  ((AND (LITATOM (CAR X))
                                                        (SETQ TEM (STRPOS OLD (CAR X)
                                                                          1 NIL T T)))
                                                   (SAVESET (SETQ TEM (PACK* NEW (SUBATOM (CAR X)
                                                                                          TEM -1)))
                                                          (COPY (GETTOPVAL (CAR X)))
                                                          T)
                                                   (FRPLACA X TEM])
              ((PROPS ALISTS)
                  (OR (EQ (CAR NEW)
                          (CAR OLD))
                      (DSUBST (CAR NEW)
                             (CAR OLD)
                             DEF))
                  (OR (EQ (CADR NEW)
                          (CADR OLD))
                      (DSUBST (CADR NEW)
                             (CADR OLD)
                             DEF)))
              (DSUBST NEW OLD DEF))
          (PUTDEF NEW TYPE DEF)
          (RETURN NEW])
```

```
(GETDEF
  [LAMBDA (NAME TYPE SOURCE OPTIONS)                        ; Edited  3-Sep-2022 16:43 by larry
                                                            (* lmm "13-Jul-85 04:10")

    ;; returns the definition of NAME as a TYPE from SOURCE;  cause ERROR if not found unless OPTIONS is NOERROR --- usually returns a copy
    ;; unless OPTIONS is NOCOPY in which case it tries not to return a copy --- FLG=NOCOPY is currently only used from SAVEDEF where
    ;; SOURCE is always 0 --- If options is or contains a string, returns that string instead of causing error if no def found.  The caller can figure out
    ;; what happened, even for types for which NIL/NOBIND might have defs.

    ;;  FROMEDITOR is used if the editing form is different from the defining form (as in LOOPS)

    (PROG (DEF TEM (EDIT (EQMEMB 'EDIT OPTIONS))
               (NOCOPY (EQMEMB 'NOCOPY OPTIONS)))
          (DECLARE (SPECVARS NOCOPY EDIT))
          (SELECTQ OPTIONS
              (0 (SETQQ OPTIONS (NOERROR NODWIM))
                 (SETQ NOCOPY T))
              (1 (SETQQ OPTIONS (NOERROR NODWIM FAST ARGLIST))
                 (SETQ NOCOPY T))
              (T (SETQQ OPTIONS SPELL))
              NIL)
          (SETQ TYPE (GETFILEPKGTYPE TYPE 'TYPE))
          (SELECTQ SOURCE
              (0 (SETQQ SOURCE CURRENT))
              (T (SETQQ SOURCE SAVED))
              (NIL (SETQQ SOURCE ?))
              NIL)
          [SELECTQ SOURCE
              (CURRENT (SETQ DEF (GETDEFCURRENT NAME TYPE OPTIONS)))
              (? [LET [(NOERROR (CONS 'NOERROR (MKLIST OPTIONS]
                      (OR (NEQ (SETQ DEF (GETDEFCURRENT NAME TYPE NOERROR))
                               (fetch NULLDEF of TYPE))
                          (NEQ (SETQ DEF (GETDEFSAVED NAME TYPE NOERROR))
                               (fetch NULLDEF of TYPE))
                          (SETQ DEF (GETDEFFROMFILE NAME TYPE 'FILE OPTIONS])
              (SAVED (SETQ DEF (GETDEFSAVED NAME TYPE OPTIONS)))
              (COND
                 ((AND (LISTP SOURCE)
                       (EQ (CAR SOURCE)
                           '=))
                  (SETQ DEF (CDR SOURCE)))
                 (T (SETQ DEF (GETDEFFROMFILE NAME TYPE SOURCE OPTIONS))
                    (SETQ NOCOPY T]
          (OR NOCOPY (SETQ DEF (COPY DEF)))
          (COND
             ((AND (EQ TYPE 'FNS)
                   (NOT (EQMEMB 'NODWIM OPTIONS)))
              (DWIMDEF DEF NAME SOURCE)))
          (RETURN DEF])
```

```
(GETDEFCOM
  [LAMBDA (X)                                               (* lmm " 4-Jul-85 13:31")

    ;; In the case where GETDEF doesn't know how to get the definition of something, it resorts to asking the file package to print it out to a file and
    ;; then reading the file back in.  Actually, though, that is a two stage process where the 'command' to print out the datum is first macro expanded
    ;; and then executed.  --- In some cases, you can tell what would be printed without printing it by looking at the prettydef-macro expansion.  That is
    ;; what GETDEFCOM does: it takes a list of prettydef commands and returns what Would be printed by those commands (or NIL if it is 'too hard' to
    ;; figure out.) --- A few of the commands are special-cased inside GETDEFCOM0 because they occur frequently or are simple.
                                                            ; a RETFROM point
    (for Y in X join (GETDEFCOM0 Y])
```

```
(GETDEFCOM0
  [LAMBDA (COM)                                             (* wt%: " 7-FEB-79 23:28")
    (PROG (TEM)
          (RETURN (COND
                     ((SETQ TEM (fetch MACRO of (CAR COM)))    (* COND ((fetch CONTENTS of (CAR COM))
                                                            (* ; "if it has a CONTENTS function, generally means it is not
                                                            safe to evaluate") (RETFROM (QUOTE GETDEFCOM))))
                      (for Y in (SUBPAIR (CAR TEM)
                                       (PRETTYCOM1 COM)
                                       (CDR TEM))
```

```
                         join (GETDEFCOM0 Y)))
                     (T (SELECTQ (CAR COM)
                             (COMS (for X in (PRETTYCOM1 COM) join (GETDEFCOM0 X)))
                             (ADDVARS (for Y in (PRETTYCOM1 COM) collect (CONS 'ADDTOVAR Y)))
                             (APPENDVARS (for Y in (PRETTYCOM1 COM) collect (CONS 'APPENDTOVAR Y)))
                             (P (APPEND (PRETTYCOM1 COM)))
                             (RETFROM 'GETDEFCOM])
```

```
(GETDEFCURRENT
  [LAMBDA (NAME TYPE OPTIONS)                                       ; Edited 2-May-87 19:00 by Pavel
                                                                    ; Gets the current definition--source=0
    (LET
     (DEF)
     (COND
        ((AND (SETQ DEF (fetch GETDEF of TYPE))
              (NEQ DEF T))

         ;; We assign T to types whose GETDEF is normally handled in the SELECTQ below but whose MACRO is to be defaulted to the
         ;; PUTDEF/GETDEF in PRETTYCOM.

         (OR (NEQ (SETQ DEF (APPLY* DEF NAME TYPE OPTIONS))
                  (fetch NULLDEF of TYPE))
             (GETDEFERR NAME TYPE OPTIONS))
         DEF)
        (T
         (OR
          (NEQ
           [SETQ DEF
            (SELECTQ TYPE
                 (FNS (AND (LITATOM NAME)
                           (EXPRP (SETQ DEF (VIRGINFN NAME)))
                           DEF))
                 (VARS (if (LITATOM NAME)
                           then (GETTOPVAL NAME)
                         else 'NOBIND))
                 ((FIELDS RECORDS)
                      (if (LITATOM NAME)
                          then [SETQ DEF (SELECTQ TYPE
                                             (RECORDS (RECLOOK NAME))
                                             (MKPROGN (FIELDLOOK NAME]
                               (if (EQMEMB 'EDIT OPTIONS)
                                   then (COPY DEF)
                                 else DEF)))
                 (FILES                                             ; what is the 'definition' of a file? -- I guess the COMS which say
                                                                    ; what it contains
                        [if (LITATOM NAME)
                            then (if (SETQ DEF (GETFILEDEF NAME))
                                     then (UPDATEFILES)
                                          (LIST (LISTP (GETTOPVAL (FILECOMS DEF)))
                                                (fetch TOBEDUMPED of (fetch FILEPROP of DEF))
                                                (LISTP (fetch FILEDATES of DEF])
                 (TEMPLATES (if (AND (LITATOM NAME)
                                     (SETQ DEF (GETTEMPLATE NAME)))
                                then (LIST 'SETTEMPLATE (KWOTE NAME)
                                           (KWOTE DEF))))
                 (MACROS [if [AND (LITATOM NAME)
                                  (SETQ DEF (for X on (GETPROPLIST NAME) by (CDDR X)
                                               when (FMEMB (CAR X)
                                                           MACROPROPS)
                                               join (LIST (CAR X)
                                                          (CADR X]
                             then `(PUTPROPS (\, NAME) ,@DEF)])
                 (EXPRESSIONS (LISTP NAME))
                 (PROPS [AND (LISTP NAME)
                             (AND (SETQ DEF (SOME (GETPROPLIST (CAR NAME))
                                                  [FUNCTION (LAMBDA (X)
                                                              (EQ X (CADR NAME]
                                                  (FUNCTION CDDR)))
                                  (LIST 'PUTPROPS (CAR NAME)
                                        (CADR NAME)
                                        (CADR DEF])
                 (FILEPKGCOMS [AND (LITATOM NAME)
                                   (PROG ((COM (FILEPKGCOM NAME))
                                          (TYP (FILEPKGTYPE NAME)))
                                         (RETURN (COND
                                                    ((AND COM TYP)
                                                     (LIST (CONS 'COM COM)
                                                           (CONS 'TYPE TYP)))
                                                    (COM (LIST (CONS 'COM COM)))
                                                    (TYP (LIST (CONS 'TYPE TYP])
                 (FILEVARS (COND
                              ((AND (LITATOM NAME)
                                    (LISTP (SETQ DEF (GETTOPVAL NAME)))
                                    (WHEREIS NAME 'FILEVARS))
                               DEF)
                              (T 'NOBIND)))
                 (LET
                  ((COMS (LIST (MAKENEWCOM NAME TYPE)))
                   FILE)
                  [COND
                     ((NOT (SETQ DEF (GETDEFCOM COMS)))
                      (WITH-READER-ENVIRONMENT *OLD-INTERLISP-READ-ENVIRONMENT*
                          (RESETLST
                              (RESETSAVE PRETTYFLG)
                              (RESETSAVE FONTCHANGEFLG)
                              [RESETSAVE (OUTPUT (SETQ FILE (OPENSTREAM '{NODIRCORE} 'BOTH]
                              (PRETTYDEFCOMS COMS)
                              (SETFILEPTR FILE 0)
                              [SETQ DEF
                               (for X in (READFILE FILE)
                                  join (SELECTQ (CAR X)
                                                  ((*)
                                                      NIL)
                                                  (DECLARE%: (for Y on (CDR X) unless (SELECTQ (CAR Y)
                                                                                                       ((COPYWHEN EVAL@LOADWHEN
                                                                                                               EVAL@COMPILEWHEN)
                                                                                                           (RETURN (LIST Y)))
                                                                                                       (FMEMB (CAR Y)
                                                                                                              DECLARETAGSLST))
                                                                 collect (CAR Y)))
                                                  (CL:EVAL-WHEN (CDDR X))
                                                  (PROGN (CDR X))
                                                  (LIST X]
                              (SETQ NOCOPY T)))]
                  (MKPROGN DEF]
           (fetch NULLDEF of TYPE))
          (GETDEFERR NAME TYPE OPTIONS))
         DEF])

(GETDEFERR
  [LAMBDA (NAME TYPE OPTIONS MSG)                              (* lmm "13-Jul-85 04:11")
    (DECLARE (USEDFREE NODEF))                                 ; Message non-null if looking for saved or filed definition.
    (PROG (TEM)
          (RETURN (COND
                     ((EQMEMB 'NOERROR OPTIONS)                ; We want to do the string search in the HASDEF case
                      (RETURN (fetch NULLDEF of TYPE)))
                     [(AND (NULL MSG)
                           (EQMEMB 'SPELL OPTIONS)
                           (SETQ TEM (HASDEF NAME TYPE NIL (OR (LISTGET1 (LISTP OPTIONS)
                                                                         'SPELL)
                                                               T)))
                           (NEQ TEM NAME))
                      (RETFROM 'GETDEF (GETDEF TEM TYPE '? (CONS 'NOERROR (MKLIST OPTIONS]
                     (T (for O inside OPTIONS when (STRINGP O) do (RETFROM 'GETDEF O)
                           finally (ERROR NAME (CONS TYPE '(definition not found))
                                        T])

(GETDEFFROMFILE
  [LAMBDA (NAME TYPE SOURCE OPTIONS)                           (* bvm%: " 1-Oct-86 22:10")

    ;; Tries to get definition from source file.  If successful, returns the definition.  Otherwise returns the NULLDEF of the type if OPTIONS contains
    ;; NOERROR.

    (DECLARE (SPECVARS NAME))
    (bind (NOTFOUND _ "not found")
         DEF SOURCE TEM2 for FILE inside (COND
                                          ((EQ SOURCE 'FILE)
                                           (WHEREIS NAME TYPE T))
                                          (T SOURCE))
       when
       (AND
        (SETQ SOURCE (FINDFILE FILE T))
        (NEQ
         [SETQ DEF
          (COND
             ((SETQ TEM2 (fetch FILEGETDEF of TYPE))
              (APPLY* TEM2 NAME TYPE SOURCE OPTIONS NOTFOUND))
             (T (SELECTQ TYPE
                    (FNS (FILEGETDEF.FNS NAME TYPE SOURCE OPTIONS NOTFOUND))
                    ((VARS FILEVARS)
                         (FILEGETDEF.VARS NAME TYPE SOURCE OPTIONS NOTFOUND))
                    (MACROS (FILEGETDEF.MACROS NAME TYPE SOURCE OPTIONS NOTFOUND))
                    (PROPS (FILEGETDEF.PROPS NAME TYPE SOURCE OPTIONS NOTFOUND))
                    (RECORDS (FILEGETDEF.RECORDS NAME TYPE SOURCE OPTIONS NOTFOUND))
                    (ALISTS (FILEGETDEF.ALISTS NAME TYPE SOURCE OPTIONS NOTFOUND))
                    (LISPXMACROS (FILEGETDEF.LISPXMACROS NAME TYPE SOURCE OPTIONS NOTFOUND))
                    (COND
                       [(SETQ DEF (GET TYPE 'DEFINERS))
                        (LET [(VAL (LOADFNS NIL SOURCE 'GETDEF
                                          `(LAMBDA (FIRST SECOND)
                                             (AND (MEMB FIRST ',DEF)
                                                  (OR (EQ SECOND NAME)
                                                      (AND (MEMB SECOND '(%( %[))
                                                            (PROGN (RATOM)
                                                                    (RATOM)
                                                                    (RATOM)
                                                                    (EQ NAME (RATOM]
                                                                 ; ick!  Should use real closure
                              (if (EQ (CAAR VAL)
                                     'NOT-FOUND)
                                  then NOTFOUND
                                elseif (CDR VAL)
                                  then (CONS 'PROGN VAL)
                                else (CAR VAL]
                        (T (RESETLST
                               (RESETSAVE (RESETUNDO))
                               [LET (LOAD-VERBOSE-STREAM)
                                    (DECLARE (SPECVARS LOAD-VERBOSE-STREAM))
                                                                 ; just in case we get a PRETTYCOMPRINT in here
                                    (LOADFNS NIL SOURCE 'PROP (COND
                                                                 ((LITATOM NAME)
                                                                 ; If an atom, only bother with expressions that contain it
                                                                  (CONS (LIST '& '|..| NAME)))
                                                                 (T T]
                               (GETDEFCURRENT NAME TYPE (CONS 'NOERROR (MKLIST OPTIONS)))))]
         NOTFOUND))
       do (AND (EQ SOURCE 'FILE)
               (OR (FMEMB FILE FILELST)
                  (CL:FORMAT T "(from ~A)~%%" SOURCE)))          ; Copying and dwimifying are done in GETDEF
          (RETURN DEF))
       finally (RETURN (GETDEFERR NAME TYPE OPTIONS (APPEND '(no definition on)
                                                           (MKLIST SOURCE])

(GETDEFSAVED
  [LAMBDA (NAME TYPE OPTIONS)                                    ; Edited 11-Aug-87 18:14 by cutting
                                                                 ; Gets the 'saved' definition--source=T
    (SELECTQ TYPE
        (FNS (OR (GETPROP NAME 'EXPR)
                  (GETDEFERR NAME TYPE OPTIONS "no saved definition for")))
        (VARS                                                    ; The value of a variable is never substituted into and never
                                                                 ; COPIED
               (for X on (GETPROPLIST NAME) by (CDDR X) when (EQ (CAR X)
                                                                'VALUE)
                  do (RETURN (CADR X)) finally (RETURN (GETDEFERR NAME TYPE OPTIONS "no saved value for "))))
        (OR (CDR (SASSOC NAME (FASSOC TYPE SAVEDDEFS)))
            (GETDEFERR NAME TYPE OPTIONS "no saved definition for "])

(PUTDEF
  [LAMBDA (NAME TYPE DEFINITION REASON)                          ; Edited  8-Apr-87 12:52 by Pavel
    (SETQ TYPE (GETFILEPKGTYPE TYPE 'TYPE))
    (LET ((PUTDEF.METHOD (fetch PUTDEF of TYPE)))
         (COND
            (PUTDEF.METHOD (APPLY* PUTDEF.METHOD NAME TYPE DEFINITION REASON))
            (T (SELECTQ TYPE
                       (FNS (FNS.PUTDEF NAME TYPE DEFINITION REASON))
                       (VARS (VARS.PUTDEF NAME TYPE DEFINITION REASON))
                       (FILES (FILES.PUTDEF NAME TYPE DEFINITION REASON))
                       (FILEPKGCOMS (FILEPKGCOMS.PUTDEF NAME TYPE DEFINITION REASON))
                       (EVAL DEFINITION))
               NAME])

(EDITDEF
  [LAMBDA (NAME TYPE SOURCE EDITCOMS OPTIONS)
    (DECLARE (LOCALVARS . T)
           (SPECVARS SOURCE))                                    ; Edited 27-Jul-87 11:04 by cutting

    ;; lets you edit anything.  Given name and type, call editor on the definition (loading it in from SOURCE if necessary).  If you change it, then the
    ;; definition gets unsaved.  OPTIONS is passed through from ED to the editor.

    (SETQ TYPE (GETFILEPKGTYPE TYPE 'TYPE))
    (COND
       ((AND (fetch EDITDEF of TYPE)
             (APPLY* (fetch EDITDEF of TYPE)
                    NAME TYPE SOURCE EDITCOMS OPTIONS)))
       ((AND (EQ TYPE 'FNS)
             (NULL SOURCE))                                      ; special hack for EDITDEF of FNS because of ability to
                                                                 ; EDITLOADFNS
        (EDITDEF.FNS NAME EDITCOMS OPTIONS))
       (T (DEFAULT.EDITDEF NAME TYPE SOURCE EDITCOMS OPTIONS)))
    NAME])

(DEFAULT.EDITDEF
  [LAMBDA (NAME TYPE SOURCE EDITCOMS OPTIONS)                    ; Edited 11-Jun-92 16:26 by cat
    (PROG [(DEF (COND
                   [SOURCE (GETDEF NAME TYPE SOURCE '(EDIT NOCOPY]
                   [(GETDEF NAME TYPE 'CURRENT '(EDIT NOCOPY NOERROR]
                   [(GETDEF NAME TYPE 'SAVED '(EDIT NOCOPY NOERROR]
                   (T (LET ((FILES (WHEREIS NAME TYPE T)))
                           (CL:IF (NULL FILES)
                               (CL:FORMAT T "~S has no ~A definition.~%%" NAME TYPE)
                               [LET [(FILE (PROGN (CL:FORMAT T "~S is contained on~{ ~S~}.~%%" NAME FILES)
                                                  (CL:IF (CL:ENDP (CDR FILES))
                                                      (CL:IF (CL:Y-OR-N-P "Shall I load this file PROP? ")
                                                             (CAR FILES))
                                                      (ASKUSER NIL NIL "indicate which file to load PROP: "
                                                             (MAKEKEYLST FILES)
                                                             T))]
                                    (CL:WHEN FILE
                                        (LOAD FILE 'PROP)
                                        (GETDEF NAME TYPE '? '(EDIT NOCOPY))))])]
     ;; the EDIT option says to return a COPY if editing this structure isn't enough, and some installation is necessary.
          (DECLARE (SPECVARS RETRY))
     ;; what is RETRY ???
          (SETQ RETRY)
          (CL:WHEN DEF
              (EDITE DEF EDITCOMS NAME TYPE [FUNCTION (LAMBDA (NAME DEF TYPE EXITFLG)
                                                                 ; this function is called when there were changes made
                                                        (FIXEDITDATE DEF)
                                                                 ; fix the edit date first - jtm
                                                        (PUTDEF NAME TYPE DEF)
                                                        (MARKASCHANGED NAME TYPE 'CHANGED)
                                      ;; woz 1/25/91 MARKASCHANGED must be called after PUTDEF, so sedit's markaschangedfn will
                                      ;; see the new definition.  doc for PUTDEF says it calls MARKASCHANGED, but it doesn't always, so
                                      ;; do it here.  this sometimes results in MARKASCHANGED getting called twice.

                                                        ]
                     OPTIONS))])

(EDITDEF.FILES
  [LAMBDA (NAME TYPE SOURCE EDITCOMS OPTIONS)                    ; Edited 18-Mar-87 16:07 by woz
    (EDITDEF (FILECOMS NAME)
           'VARS SOURCE EDITCOMS OPTIONS])

(LOADDEF
  [LAMBDA (NAME TYPE SOURCE)                                     (* lmm "13-SEP-78 01:34")
    (PUTDEF NAME TYPE (GETDEF NAME TYPE SOURCE '(NODWIM NOCOPY])

(DWIMDEF
  [LAMBDA (DEF FN SOURCE)                                        (* lmm " 6-Jun-86 17:23")
    (AND [OR (EQ DWIMIFYCOMPFLG T)
             (EQ CLISPIFYPRETTYFLG T)
             (EQ (CAR (CADDR DEF))
                 'CLISP%:)
             (SELECTQ SOURCE
                 ((CURRENT SAVED FILE ?)
                      NIL)
                 (AND (LITATOM SOURCE)
                      (EQMEMB 'CLISP (GETPROP SOURCE 'FILETYPE]
         (LET ((NOSPELLFLG T)
               (DWIMESSGAG T)
               FILEPKGFLG LISPXHIST)
              (DECLARE (CL:SPECIAL NOSPELLFLG DWIMESSGAG FILEPKGFLG LISPXHIST))
              (DWIMIFY0 DEF (COND
                                ((OR (LISTP FN)
                                     (NULL FN))
                                 '?)
                                (T FN))
                     NIL DEF])

(DELDEF
  [LAMBDA (NAME TYPE)                                            ; Edited  5-Dec-86 06:20 by lmm
    (PROG (TEM)
          (SETQ TYPE (GETFILEPKGTYPE TYPE 'TYPE))
      LP  [COND
             ((SETQ TEM (fetch DELDEF of TYPE))
              (APPLY* TEM NAME TYPE))
             (T (SELECTQ TYPE
                     (FNS                                        ; special because GETDEF of a FNS is only its EXPR definition,
                                                                 ; and DELDEF should only remove such
                          (AND (EXPRP NAME)
                               (/PUTD NAME))
                          (REMPROP NAME 'EXPR)
                          [AND MSDATABASELST (MASTERSCOPE (LIST 'ERASE (KWOTE NAME])
                     (VARS (/SETTOPVAL NAME 'NOBIND))
                     (FILES [for LST in '(FILELST NOTCOMPILEDFILES NOTLISTEDFILES)
                               do (/SETTOPVAL LST (REMOVE NAME (GETTOPVAL LST]
                            (/replace FILEPROP of NAME with NIL)
                            (/replace FILECHANGES of NAME with NIL)
                            (/replace FILEDATES of NAME with NIL)
                            (FLUSHFILEMAPS NAME))
                     (FILEPKGCOMS (DELFROMLIST 'FILEPKGCOMSPLST NAME)
                                  (DELFROMLIST 'FILEPKGTYPES NAME)
                                  (for FIELD on (FILEPKGCOM NAME) by (CDDR FIELD)
                                     do (FILEPKGCOM NAME (CAR FIELD)
                                                NIL))
                                  (for FIELD on (FILEPKGTYPE NAME) by (CDDR FIELD)
                                     do (FILEPKGTYPE NAME (CAR FIELD)
                                                NIL))
                                  (/replace ALLFIELDS of NAME with NIL))
                     (ALISTS [AND (LISTP NAME)
                                  (DELFROMLIST (CAR NAME)
                                         (FASSOC (CADR NAME)
                                                (GETTOPVAL (CAR NAME])
                     (MACROS (for P in MACROPROPS do (/REMPROP NAME P)))
                     (PROPS (AND (LISTP NAME)
                                 (/REMPROP (CAR NAME)
                                        (CADR NAME))))
                     (LISPXMACROS (DELFROMLIST 'LISPXMACROS (FASSOC NAME LISPXMACROS))
                                  (DELFROMLIST 'LISPXHISTORYMACROS (FASSOC NAME LISPXHISTORYMACROS))
                                  (DELFROMLIST 'LISPXCOMS NAME)
                                  (DELFROMLIST 'HISTORYCOMS NAME))
                     (PRIN1 (LIST "Note: deleting" TYPE "not implemented yet")
                            T]
          (MARKASCHANGED NAME TYPE 'DELETED)
          (RETURN NAME])

(DELFROMLIST
  [LAMBDA (VAR VAL)                                             (* rmk%: " 3-JAN-82 23:22")
    (AND (FMEMB VAL (GETTOPVAL VAR))
         (/SETTOPVAL VAR (SUBSET (GETTOPVAL VAR)
                                 (FUNCTION (LAMBDA (X)
                                             (AND (NEQ X VAL)
                                                  (OR (NLISTP X)
                                                      (NEQ (CDR X)
                                                           VAL])

(HASDEF
  [LAMBDA (NAME TYPE SOURCE SPELLFLG)                           ; Edited 31-Aug-87 18:02 by drc:
    ;; is NAME the name of something of type TYPE?  NIL SOURCE means 0, not ?
    (DECLARE (SPECVARS TYPE))
    (COND
       [[OR (LISTP TYPE)
            (LISTP (SETQ TYPE (GETFILEPKGTYPE TYPE 'TYPE]       ; ignore SPELLFLG
        (for TY in TYPE do (AND (SETQ $$VAL (HASDEF NAME TY SOURCE))
                               (RETURN $$VAL]
       (T
        (PROG [(NODEF (fetch NULLDEF of TYPE))
               (OPTS '(NODWIM NOCOPY NOERROR HASDEF]
              (COND
                 ((NULL SOURCE)
                  (SETQQ SOURCE CURRENT)))
              (RETURN (SELECTQ SOURCE
                          ((CURRENT 0)
                               [COND
                                  ([OR (MEMBER NAME (fetch CHANGED of TYPE))
                                       (LET ((TM (fetch HASDEF of TYPE)))
                                            (COND
                                               (TM (APPLY* TM NAME TYPE SOURCE))
                                               [(NOT (LITATOM NAME))
                                                (SELECTQ TYPE
                                                     (PROPS (AND (LISTP NAME)
                                                                 (GETPROP (CAR NAME)
                                                                        (CADR NAME))))
                                                     ((FILES TEMPLATES MACROS LISPXMACROS VARS I.S.OPRS FNS
                                                             FIELDS USERMACROS FILEVARS FILEPKGCOMS)
                                                         NIL)
                                                     (NEQ NODEF (GETDEF NAME TYPE 'CURRENT OPTS]
                                               (T ;; symbol definitions
                                                  (SELECTQ TYPE
                                                      (FILES (LET ((SYMBOL (CL:FIND-SYMBOL (CONCAT NAME "COMS")
                                                                                  "INTERLISP")))
                                                                  (AND SYMBOL (BOUNDP SYMBOL))))
                                                      (TEMPLATES (GETTEMPLATE NAME))
                                                      (MACROS (GETLIS NAME MACROPROPS))
                                                      (LISPXMACROS (OR (FASSOC NAME LISPXMACROS)
                                                                       (FASSOC NAME LISPXHISTORYMACROS)))
                                                      (VARS (AND (NOT (CL:KEYWORDP NAME))
                                                                 (NEQ (GETTOPVAL NAME)
                                                                      'NOBIND)))
                                                      (RECORDS (RECLOOK NAME))
                                                      (I.S.OPRS [PROG [(TEM (GETPROP NAME 'CLISPWORD]
                                                                      (RETURN (AND TEM (EQ (CAR TEM)
                                                                                           'FORWORD)
                                                                                   (GETPROP (CDR TEM)
                                                                                          'I.S.OPR])
                                                      (FNS (AND (OR (AND (GETD NAME)
                                                                         (EXPRP (GETD NAME)))
                                                                    (GET NAME 'EXPR))
                                                                (NOT (HASDEF NAME 'FUNCTIONS SOURCE))))
                                                      (FIELDS (RECORDFIELD? NAME))
                                                      (USERMACROS (FASSOC NAME USERMACROS))
                                                      (FILEVARS)
                                                      ((PROPS ALISTS DEFS EXPRESSIONS)
                                                          NIL)
                                                      (FILEPKGCOMS (OR (FMEMB NAME FILEPKGCOMSPLST)
                                                                       (FMEMB NAME FILEPKGTYPES)))
                                                      (NEQ NODEF (GETDEF NAME TYPE 'CURRENT OPTS]
                                  (OR NAME T))
                                 (SPELLFLG (CL:WHEN (CL:SYMBOLP NAME)
                                               (FIXSPELL NAME NIL
                                                      (SELECTQ TYPE
                                                           (FILES FILELST)
                                                           (FILEPKGCOMS (UNION FILEPKGCOMSPLST FILEPKGTYPES))
                                                           (FIELDS (for X in USERRECLST
                                                                     join (APPEND (RECORDFIELDNAMES X))))
                                                           (RECORDS (for X in USERRECLST
                                                                       when (LITATOM (CADR X))
                                                                       collect (CADR X)))
                                                           (LISPXMACROS LISPXCOMS)
                                                           (I.S.OPRS I.S.OPRLST)
                                                           (USERMACROS (MAPCAR USERMACROS
                                                                              (FUNCTION CAR)))
                                                           USERWORDS)
                                                      NIL
                                                      (LISTP SPELLFLG)
                                                      [FUNCTION (LAMBDA (X)
                                                                  (HASDEF X TYPE 'CURRENT]
                                                      NIL T))]])
                          (? (OR (HASDEF NAME TYPE 'CURRENT)
                                 (AND (LITATOM NAME)
                                      (HASDEF NAME TYPE 'SAVED SPELLFLG))
                                 (WHEREIS NAME TYPE T)))
                          ((SAVED T)
                               (NEQ NODEF (GETDEF NAME TYPE 'SAVED OPTS)))
                          (NEQ NODEF (GETDEF NAME TYPE SOURCE OPTS])
```

```
(GETFILEDEF
  [LAMBDA (FILENAME)                                              (* lmm " 4-Jul-85 13:25")
    ;; returns the official file name from a file name if NAME is FOO, look for FOO.LSP on FILELST
    (COND
       ((FMEMB FILENAME FILELST)
        FILENAME)
       (T (for FILE in FILELST when (STRPOS FILENAME FILE 1 NIL T) do (COND
                                                                        ((EQ (FILENAMEFIELD FILE 'NAME)
                                                                             FILENAME)
                                                                         (RETURN FILE])
```

```
(SAVEDEF
  [LAMBDA (NAME TYPE DEFINITION)                                  (* JonL "24-Jul-84 20:11")
    (COND
       [(AND (LISTP NAME)
             (NULL TYPE))
        (MAPCAR NAME (FUNCTION (LAMBDA (I)
                                 (SAVEDEF I 'FNS]
       (T [SELECTQ (SETQ TYPE (GETFILEPKGTYPE TYPE 'TYPE))
               (FNS (AND (OR DEFINITION (SETQ DEFINITION (GETD NAME)))
                         (/PUT NAME [SETQ TYPE (COND
                                                  ((SUBRP DEFINITION)
                                                   'SUBR)
                                                  ((EXPRP DEFINITION)
                                                   'EXPR)
                                                  ((CCODEP DEFINITION)
                                                   'CODE)
                                                  (T 'LIST]
                              DEFINITION)))
               (VARS (AND (NEQ (OR DEFINITION (SETQ DEFINITION (GETTOPVAL NAME)))
                               'NOBIND)
                          (EQ DEFINITION (GETTOPVAL NAME))
                          (/PUT NAME (SETQ TYPE 'VALUE)
                                DEFINITION)))
               (AND [OR DEFINITION (SETQ DEFINITION (GETDEF NAME TYPE 'CURRENT '(NOCOPY NOERROR NODWIM]
                    (/PUTASSOC NAME DEFINITION (OR (CDR (FASSOC TYPE SAVEDDEFS))
                                                   (CAR (SETQ SAVEDDEFS (CONS (LIST TYPE (CONS NAME))
                                                                              SAVEDDEFS]
          TYPE])
```

```
(UNSAVEDEF
  [LAMBDA (NAME TYPE DEF)                                       (* lmm " 6-Jun-86 17:24")
    (SELECTQ TYPE
        ((NIL EXPR CODE SUBR LIST)
             (COND
                [(LISTP NAME)                                   ; for compatibility
                 (MAPCAR NAME (FUNCTION (LAMBDA (X)
                                          (UNSAVED1 X TYPE]
                (T (UNSAVED1 NAME TYPE))))
        (PROG NIL
              [OR DEF (SETQ DEF (GETDEF NAME (SETQ TYPE (GETFILEPKGTYPE TYPE 'TYPE))
                                       'SAVED 0))
                  (RETURN (CONS TYPE '(not found]
              (COND
                 ((NEQ DFNFLG T)
                  (SAVEDEF NAME TYPE)
                  (LET ((DFNFLG T))
                       (PUTDEF NAME TYPE DEF)))
                 (T (PUTDEF NAME TYPE DEF)))
              (RETURN TYPE])

(COMPAREDEFS
  [LAMBDA (NAME TYPE SOURCES)                                   ; Edited  8-Nov-2021 10:52 by rmk:
                                                                ; Edited 30-Oct-2021 20:01 by rmk:
                                                                (* lmm " 4-Jul-85 14:37")
    (COND
       ((AND (LISTP TYPE)
             (GETFILEPKGTYPE SOURCES NIL T))
        (swap TYPE SOURCES)))
    (SETQ TYPE (GETFILEPKGTYPE TYPE 'TYPE))
    (PROG [DEF DEFS (SRCS (OR SOURCES (WHEREIS NAME TYPE T]
          [COND
             ((NULL SOURCES)
              (AND [OR (MEMBER NAME (FILEPKGCHANGES TYPE))
                       (SOME SRCS (FUNCTION (LAMBDA (FILE)
                                              (MEMBER NAME (CDR (ASSOC TYPE (fetch TOBEDUMPED
                                                                                of (fetch FILEPROP of FILE]
                   (push SRCS 'CURRENT]
          (SETQ SRCS (for SRC in SRCS when (COND
                                             ((NEQ [SETQ DEF (GETDEF NAME TYPE SRC '(NOERROR NOCOPY]
                                                   (fetch NULLDEF of TYPE))
                                              (OR [SOME DEFS (FUNCTION (LAMBDA (DP)
                                                                         (COMPARELST DEF (CDR DP]
                                                  (push DEFS (CONS SRC DEF)))
                                              T)
                                             (T (PRINTOUT T "No " SRC " definition found for " NAME T)
                                               NIL))
                        collect SRC))
          (RETURN (COND
                     ((NULL SRCS)
                      '(no definitions found))
                     ((NULL (CDR SRCS))
                      '(only one definition found))
                     ((CDR DEFS)
                      [for S1 [FILECOL _ (IPLUS (NCHARS NAME)
                                                 (CONSTANT (NCHARS " from "]
                         on (DREVERSE DEFS) do (for S2 on (CDR S1) do (PRIN2 NAME T T)
                                                                     (AND (CAAR S1)
                                                                          (PRIN1 " from " T)
                                                                          (PRIN2 (CAAR S1)
                                                                                 T T))
                                                                     (TAB (IDIFFERENCE FILECOL
                                                                                 (CONSTANT (NCHARS " and ")))
                                                                          NIL T)
                                                                     (PRIN1 " and " T)
                                                                     (COND
                                                                        ((CAAR S2)
                                                                         (PRIN2 (CAAR S2)
                                                                                T T)))
                                                                     (TERPRI T)
                                                                     (COMPARELISTS (CDAR S1)
                                                                            (CDAR S2]
                      'DIFFERENT)
                     (T 'SAME])

(COMPARE
  [LAMBDA (NAME1 NAME2 TYPE SOURCE1 SOURCE2)                    (* lmm " 5-SEP-78 13:37")
    (PROG [[DEF1 (GETDEF NAME1 TYPE SOURCE1 '(NOERROR NOCOPY]
           (DEF2 (GETDEF NAME2 TYPE SOURCE2 '(NOERROR NOCOPY]
          (COND
             ((COMPARELST DEF1 DEF2)
              (RETURN)))
          (PRIN2 NAME1 T T)
          (COND
             (SOURCE1 (PRIN1 " from " T)
                    (PRIN2 SOURCE1 T T)))
          (PRIN1 " and " T)
          (PRIN2 NAME2 T T)
          (COND
             (SOURCE2 (PRIN1 " from " T)
                    (PRIN2 SOURCE2 T T)))
          (PRIN1 " differ:" T)
          (TERPRI T)
          (COMPARELISTS DEF1 DEF2)
          (RETURN T])

(TYPESOF
  [LAMBDA (NAME POSSIBLETYPES IMPOSSIBLETYPES SOURCE FILTER)      ; Edited  2-Aug-88 02:08 by masinter
    ;; return list of all known types which NAME names
    (LET (FOUND SHADOWED)
         (if (FMEMB SOURCE '(? NIL))
             then (CL:FLET [(RSHADOW NIL (for X in FOUND do (for Y in (CDR (FASSOC X SHADOW-TYPES))
                                                                do (if (FMEMB Y FOUND)
                                                                        then
                                                                    ; shadower found before shadowed
                                                                             (SETQ FOUND (REMOVE Y FOUND]
                          (LET (NOTFOUND NEWTYPES)
                               (for TYPE inside (OR POSSIBLETYPES FILEPKGTYPES)
                                  when [AND (LITATOM TYPE)
                                            (NOT (EQMEMB TYPE IMPOSSIBLETYPES))
                                            (OR (NULL FILTER)
                                                (CL:FUNCALL FILTER TYPE))
                                            (NOT (find X in FOUND suchthat (FMEMB TYPE (CDR (FASSOC X SHADOW-TYPES]
                                  do (if [OR (HASDEF NAME TYPE 'CURRENT)
                                             (AND (LITATOM NAME)
                                                  (HASDEF NAME TYPE 'SAVED]
                                         then (push FOUND TYPE)
                                       else (push NOTFOUND TYPE)))
                               (RSHADOW)
                               [for FILE in FILELST while NOTFOUND when [NEQ T (fetch LOADTYPE
                                                                                   of (GETPROP FILE 'FILE]
                                  do (if (SETQ NEWTYPES (INFILECOMS? NAME NOTFOUND (FILECOMS FILE)
                                                                  'TYPESOF))
                                         then [bind X for TYPE in NEWTYPES when (FMEMB TYPE NOTFOUND)
                                                 do (push FOUND TYPE)
                                                     (if (SETQ X (FASSOC TYPE SHADOW-TYPES))
                                                         then (SETQ NOTFOUND (LDIFFERENCE NOTFOUND X))
                                                       else (SETQ NOTFOUND (REMOVE TYPE NOTFOUND]
                                              (SETQ NOTFOUND (LDIFFERENCE NOTFOUND NEWTYPES]
                               (if (AND NOTFOUND (GETD 'XCL::HASH-FILE-TYPES-OF))
                                   then (SETQ NEWTYPES (XCL::HASH-FILE-TYPES-OF NAME NOTFOUND))
                                        (SETQ FOUND (UNION NEWTYPES FOUND)))
                               (RSHADOW)
                               FOUND))
           else (for TYPE inside (OR POSSIBLETYPES FILEPKGTYPES) when (AND (LITATOM TYPE)
                                                                             (NOT (EQMEMB TYPE IMPOSSIBLETYPES))
                                                                             (OR (NULL FILTER)
                                                                                 (CL:FUNCALL FILTER TYPE))
                                                                             (HASDEF NAME TYPE SOURCE))
                   do (push FOUND TYPE)))
         FOUND])
)

(RPAQ? WHEREIS.HASH )

;; how to dump FILEPKGCOMS. The SPLST must be initialized to contain FILEPKGCOMS in order to get started.

(DEFINEQ

(FILEPKGCOM
  [LAMBDA N                                                           (* JonL "10-Jul-84 19:38")
    (PROG (TEM (COM (ARG N 1)))
          (RETURN (COND
                      [(EQ N 1)
                       (OR (for FIELD in '(MACRO CONTENTS DELETE ADD) when (SETQ TEM (FILEPKGCOM COM FIELD))
                              join (LIST FIELD TEM))
                           (AND (FMEMB COM (GETTOPVAL 'FILEPKGCOMSPLST))
                                (LIST 'COM T))
                           (AND [SETQ TEM (CDR (ASSOC COM (GETTOPVAL 'FILEPKGCOMSPLST]
                                (LIST 'COM TEM]
                      ((EQ N 2)
                       (SELECTQ (ARG N 2)
                            (ADD (fetch ADD of COM))
                            (DELETE (fetch DELETE of COM))
                            (MACRO (fetch MACRO of COM))
                            ((CONTENTS CONTAIN)
                                 [OR (fetch (FILEPKGCOM CONTENTS) of COM)
```

```
                                    (COND
                                       ((SETQ COM (fetch (FILEPKGCOM PRETTYTYPE) of COM))
                                        (COND
                                           ((EQ COM 'NILL)
                                            COM)
                                           [(EQ (CAR COM)
                                                'LAMBDA)
                                            (CONS (CAR COM)
                                                  (CONS [CONS (CAADR COM)
                                                              (CONS (OR (CADDR (CADR COM))
                                                                        'NAME)
                                                                    (CONS (CADR (CADR COM))
                                                                          (CDDDR (CADR COM]
                                                        (SUBST 'INFILECOMTAIL 'PRETTYCOM1 (CDDR COM]
                                           (T (LIST 'LAMBDA '(COM TYPE NAME)
                                                    (CONS COM '(COM TYPE NAME])
                           (COM [OR (AND (FMEMB COM (GETTOPVAL 'FILEPKGCOMSPLST))
                                         T)
                                    (CDR (ASSOC COM (GETTOPVAL 'FILEPKGCOMSPLST])
                           (ERROR (ARG N 2)
                                  "not file package command property")))
                      (T [for I TEM2 from 2 to N by 2
                           do (SETQ TEM (ARG N (ADD1 I)))
                              (COND
                                 [(EQ (ARG N I)
                                      'COM)
                                  (SELECTQ TEM
                                      (NIL)
                                      (T [OR (FMEMB COM (GETTOPVAL 'FILEPKGCOMSPLST))
                                             (/SETTOPVAL 'FILEPKGCOMSPLST (CONS COM (GETTOPVAL
                                                                                     'FILEPKGCOMSPLST])
                                      (COND
                                         ([SETQ TEM2 (ASSOC COM (GETTOPVAL 'FILEPKGCOMSPLST]
                                          (/RPLACD TEM2 TEM))
                                         (T (/SETTOPVAL 'FILEPKGCOMSPLST (CONS (CONS COM TEM)
                                                                               (GETTOPVAL 'FILEPKGCOMSPLST]
                                 (T [AND TEM (OR (FMEMB COM (GETTOPVAL 'FILEPKGCOMSPLST))
                                                 (/SETTOPVAL 'FILEPKGCOMSPLST (CONS COM (GETTOPVAL
                                                                                         'FILEPKGCOMSPLST]
                                    (SELECTQ (ARG N I)
                                        (ADD (/replace (FILEPKGCOM ADD) of COM with TEM))
                                        (DELETE (/replace (FILEPKGCOM DELETE) of COM with TEM))
                                        (MACRO (/replace (FILEPKGCOM MACRO) of COM with TEM))
                                        ((CONTENTS CONTAIN)
                                             (/replace (FILEPKGCOM CONTENTS) of COM with TEM))
                                        (ERROR (ARG N I)
                                               "not file package command property"]
                         (MARKASCHANGED COM 'FILEPKGCOMS])

(FILEPKGTYPE
  [LAMBDA N                                                         ; Edited 13-Jun-2021 10:20 by rmk:
    (PROG ((TYPE (ARG N 1))
           TEM)
          (RETURN (COND
                      [(EQ N 1)
                       (OR (for FIELD in (UNION '(DESCRIPTION)
                                                FILEPKGTYPEPROPS)
                              when (SETQ TEM (FILEPKGTYPE TYPE FIELD)) join (LIST FIELD TEM))
                           (AND (FMEMB TYPE (GETTOPVAL 'FILEPKGTYPES))
                                (LIST 'TYPE T))
                           (AND [SETQ TEM (CDR (ASSOC TYPE (GETTOPVAL 'FILEPKGTYPES]
                                (LIST 'TYPE TEM]
                      [(EQ N 2)
                       (if (FMEMB (ARG N 2)
                                  FILEPKGTYPEPROPS)
                          then (GETPROP TYPE (ARG N 2))
                        else (SELECTQ (ARG N 2)
                                  (DESCRIPTION (fetch (FILEPKGTYPE DESCRIPTION) of TYPE))
                                  (TYPE [OR (AND (FMEMB TYPE (GETTOPVAL 'FILEPKGTYPES))
                                                 T)
                                            (CDR (ASSOC TYPE (GETTOPVAL 'FILEPKGTYPES])
                                  (ERROR (ARG N 2)
                                        "not file package type property"]
                      (T [for I TEM2 from 2 to N by 2
                           do (SETQ TEM (ARG N (ADD1 I)))
                              (COND
                                 [(EQ (ARG N I)
                                      'TYPE)
                                  (SELECTQ TEM
                                      (NIL)
                                      (T (OR (FMEMB TYPE FILEPKGTYPES)
                                             (/SETTOPVAL 'FILEPKGTYPES (CONS TYPE FILEPKGTYPES))))
                                      (COND
                                         ((SETQ TEM2 (ASSOC TYPE FILEPKGTYPES))
                                          (/RPLACD TEM2 TEM))
                                         (T (/SETTOPVAL 'FILEPKGTYPES (CONS (CONS TYPE TEM)
                                                                              FILEPKGTYPES]
                                  (T [AND TEM (OR (FMEMB TYPE FILEPKGTYPES)
                                                  (/SETTOPVAL 'FILEPKGTYPES (CONS TYPE FILEPKGTYPES]
                                     (if (FMEMB (ARG N I)
                                               FILEPKGTYPEPROPS)
                                         then (if TEM
                                                  then (/PUTPROP TYPE (ARG N I)
                                                                  TEM)
                                                else (/REMPROP TYPE (ARG N I)))
                                       else (SELECTQ (ARG N I)
                                                (DESCRIPTION (/replace (FILEPKGTYPE DESCRIPTION) of TYPE
                                                                with TEM))
                                                (ERROR (ARG N I)
                                                       "not file package command/type property"]
                         (MARKASCHANGED TYPE 'FILEPKGCOMS])

)

(PUTPROPS FILEPKGCOM ARGNAMES (COMMANDNAME (KEYWORDS%: MACRO ADD DELETE CONTENTS)))

(ADDTOVAR FILEPKGCOMSPLST FILEPKGCOMS)

(ADDTOVAR FILEPKGTYPES FILEPKGCOMS)

[PUTDEF 'FILEPKGCOMS 'FILEPKGCOMS '([COM CONTENTS (LAMBDA (COM NAME TYPE)
                                                          (* Revert to NILL when no longer coercing
                                                             PRETTYDEFMACROS to FILEPKGCOMS)
                                                          (AND (EQ TYPE 'FILEPKGCOMS)
                                                               (INFILECOMTAIL COM]
                                     (TYPE DESCRIPTION "file package commands/types" GETDEF T PUTDEF
                                           FILEPKGCOMS.PUTDEF]

[PUTDEF 'ALISTS 'FILEPKGCOMS '([COM MACRO (X (COMS * (MAKEALISTCOMS . X]
                                (TYPE DESCRIPTION "alist entries" GETDEF ALISTS.GETDEF WHENCHANGED (
                                                                                                ALISTS.WHENCHANGED
                                                                                                     ]

[PUTDEF 'DEFS 'FILEPKGCOMS '((COM MACRO (X (COMS . X]

[PUTDEF 'EDITMACROS 'FILEPKGCOMS '((TYPE TYPE USERMACROS]

[PUTDEF 'EXPRESSIONS 'FILEPKGCOMS '((TYPE DESCRIPTION "expressions" WHENCHANGED (EXPRESSIONS.WHENCHANGED)
                                           EDITDEF NILL]

[PUTDEF 'FIELDS 'FILEPKGCOMS '((TYPE EDITDEF NILL]

[PUTDEF 'FILEPKGTYPES 'FILEPKGCOMS '((COM COM FILEPKGCOMS)
                                      (TYPE TYPE FILEPKGCOMS]

[PUTDEF 'FILES 'FILEPKGCOMS '([COM MACRO [X (P * (CONS (MAKEFILESCOMS . X]
                                   CONTENTS
                                   (LAMBDA (COM NAME TYPE)
                                           (AND (EQ TYPE 'FILES)
                                                (SUBSET (INFILECOMTAIL COM)
                                                       (FUNCTION LITATOM]
                               (TYPE PUTDEF FILES.PUTDEF WHENCHANGED (FILES.WHENCHANGED)
                                     EDITDEF EDITDEF.FILES]

[PUTDEF 'FILEVARS 'FILEPKGCOMS '((COM MACRO (X (VARS . X)))
                                  (TYPE NULLDEF NOBIND EDITDEF NILL]

[PUTDEF 'FNS 'FILEPKGCOMS '((COM MACRO (X [E (MAPC 'X (FUNCTION (LAMBDA (FN)
                                                                         (AND (GETPROP FN 'FUNCTIONS)
                                                                              (CL:WARN "~A has a FUNCTIONS
                                                                                     definition" FN]
                                           (ORIGINAL (FNS . X)))
                                  CONTENTS NILL)
                             (TYPE DESCRIPTION "functions" PUTDEF FNS.PUTDEF CANFILEDEF T]

[PUTDEF 'INITRECORDS 'FILEPKGCOMS '((COM MACRO (X (P * (RECORDALLOCATIONS . X)))
                                          CONTENTS
                                          (LAMBDA (COM NAME TYPE ONFILETYPE)
                                                 (AND (NULL ONFILETYPE)
                                                      (EQ TYPE 'RECORDS)
                                                      (INFILECOMTAIL COM]

[PUTDEF 'INITVARS 'FILEPKGCOMS '((COM COM T]

[PUTDEF 'LISPXCOMS 'FILEPKGCOMS '((TYPE TYPE LISPXMACROS]

[PUTDEF 'LISPXMACROS 'FILEPKGCOMS '((COM MACRO (X (COMS * (MAKELISPXMACROSCOMS . X)))
                                          CONTENTS NILL)
                                     (TYPE DESCRIPTION "LISPX commands"]

[PUTDEF 'MACROS 'FILEPKGCOMS
       '((COM MACRO [X (DECLARE%: EVAL@COMPILE (P * (MAPCAR 'X (FUNCTION
                                                                 (LAMBDA
                                                                  (Y)
                                                                  (LET [[FNDEF (GETDEF Y 'FUNCTIONS 'CURRENT
                                                                                      '(NOCOPY NOERROR]
                                                                        (MACDEF (GETDEF Y 'MACROS 'CURRENT
                                                                                       '(NOCOPY NOERROR]
                                                                       (COND ((AND FNDEF (EQ (CAR FNDEF)
                                                                                             'DEFMACRO))
                                                                              (CL:WARN "Need to change MACROS to
                                                                                     FUNCTIONS for writing out
                                                                                     Common Lisp macro ~S." FNDEF
                                                                                     )
                                                                              (LIST 'PROGN FNDEF MACDEF))
                                                                             (T (OR MACDEF (CL:CERROR "Go ahead
                                                                                                  and finish
                                                                                                  writing out the
                                                                                                  file." "No
                                                                                                  MACROS
                                                                                                  definition for
                                                                                                  ~A." Y)
                                                                                    (GETDEF Y 'MACROS
                                                                                           'CURRENT]
              CONTENTS NILL)
         (TYPE DESCRIPTION "Interlisp macros" GETDEF MACROS.GETDEF WHENCHANGED (CLEARCLISPARRAY]

[PUTDEF 'PRETTYDEFMACROS 'FILEPKGCOMS '((COM COM FILEPKGCOMS]

[PUTDEF 'PROPS 'FILEPKGCOMS '([COM MACRO (X (COMS * (MAKEPROPSCOMS . X]
                              (TYPE DESCRIPTION "property lists" WHENCHANGED (PROPS.WHENCHANGED]

[PUTDEF 'RECORDS 'FILEPKGCOMS '([COM MACRO (X [E (MAPC 'X (FUNCTION (LAMBDA (RECORD)
                                                                            (AND (GETPROP RECORD 'STRUCTURES)
                                                                                 (CL:WARN "~A has a STRUCTURES
                                                                                        definition" RECORD]
                                              (E (RECORDECLARATIONS . X))
                                              (INITRECORDS . X))
                                     CONTENTS
                                     (LAMBDA (COM NAME TYPE ONFILETYPE)
                                            (AND (EQ TYPE 'FIELDS)
                                                 (NULL ONFILETYPE)
                                                 (MAPCONC (INFILECOMTAIL COM)
                                                        (FUNCTION (LAMBDA (X)
                                                                    (APPEND (RECORDFIELDNAMES X]
                                (TYPE DESCRIPTION "records" DELDEF (LAMBDA (X)
                                                                     (/SETTOPVAL 'USERRECLST
                                                                            (REMOVE (RECLOOK X)
                                                                                   USERRECLST]

[PUTDEF 'OLDRECORDS 'FILEPKGCOMS '((COM COM T]

[PUTDEF 'SYSRECORDS 'FILEPKGCOMS '((COM MACRO (X (E (SAVEONSYSRECLST . X)))
                                        CONTENTS
                                        (LAMBDA (COM NAME TYPE ONFILETYPE)
                                               (AND (NULL ONFILETYPE)
                                                    (EQ TYPE 'RECORDS)
                                                    (INFILECOMTAIL COM]

[PUTDEF 'USERMACROS 'FILEPKGCOMS '((COM MACRO (X (COMS * (MAKEUSERMACROSCOMS . X)))
                                        CONTENTS NILL)
                                   (TYPE DESCRIPTION "edit macros"]

[PUTDEF 'VARS 'FILEPKGCOMS '((COM MACRO (X [E (MAPC 'X (FUNCTION (LAMBDA (VAR)
                                                                         (AND (GETPROP VAR 'VARIABLES)
                                                                              (CL:WARN "~A also has a VARIABLES
                                                                                     definition" VAR]
                                           (ORIGINAL (VARS . X)))
                                  CONTENTS NILL)
                             (TYPE DESCRIPTION "variables" NULLDEF NOBIND PUTDEF VARS.PUTDEF]

[PUTDEF '* 'FILEPKGCOMS '((COM CONTENTS NILL]

[PUTDEF 'CONSTANTS 'FILEPKGCOMS '((COM MACRO (X (DECLARE%: EVAL@COMPILE (VARS . X)
                                                        (P (CONSTANTS . X]

(ADDTOVAR SHADOW-TYPES (FUNCTIONS FNS)
                         (VARIABLES VARS CONSTANTS))

(RPAQ? SAVEDDEFS )

;; EDITCALLERS

(DEFINEQ

(FINDCALLERS
  [LAMBDA (ATOMS FILES DEPTH)                                  ; Edited 13-Jul-2023 14:55 by rmk
                                                               (* lmm "30-SEP-78 01:36")
    (PROG ((X (EDITCALLERS ATOMS FILES T DEPTH)))
          (RETURN (NCONC (DREVERSE (CDR X))
                          (AND (CAR X)
                               (LIST (CONS (COND
                                              ((CDR X)
                                               '"plus other places on")
                                              (T 'on))
                                           (CAR X])

(EDITCALLERS
  [LAMBDA (ATOMS FILES COMS DEPTH)
    ;; Edited 13-Jul-2023 14:56 by rmk
    ;; Edited 31-Oct-2022 16:04 by rmk
    ;; Edited 24-Jul-2022 15:45 by rmk
    ;; Edited 21-Jul-2022 21:51 by rmk
    ;; Edited 19-Jul-2022 22:33 by rmk
    ;; Edited  9-Jul-2022 22:09 by rmk
    ;; Edited 30-Jun-2022 20:18 by rmk: Make confirmation of separator boundaries as a posttest on successful matches rather than a case-array test.
    ;; This means this can run at byte-level FFILEPOS speed for arbitrary external formats.
    ;; Edited 28-Mar-2022 20:32 by rmk: FILDIR with depth 2, reopen stream after LOADFILEMAP
    ;; Edited 24-Mar-2022 16:38 by rmk: If FILES contains *, use FILDIR
    ;; Edited 28-Jun-2021 09:50 by rmk:
                                                                  (* bvm%: " 3-Nov-86 17:30")
    (LET
     (FFILEPOSPATTERNS FNS OTHERSFILES EDITPATTERN)
     [SETQ EDITPATTERN (EDITFPAT (CONS '*ANY* (SETQ ATOMS (MKLIST ATOMS]
     [for FILE FULL in (COND
                         ((NULL FILES)
                          FILELST)
                         ((EQ FILES T)
                          (UNION SYSFILES FILELST))
                         ((LISTP FILES)
                          FILES)
                         ((STRPOS "*" FILES)                      ; Depth 2 for TMAX>TMAX
                          (FILDIR FILES (OR DEPTH 2)))
                         (T (LIST FILES)))
        unless (DIRECTORYNAMEP FILE)
        do
        (RESETLST
            [PROG (PATTERNS CA RDTBL MAP FILESTREAM PRINTFLG ENV TOP I CASEINSENSITIVE)
                  (OR (SETQ FULL (FINDFILE FILE))
                      (RETURN (LISPXPRINT (CONS FILE '(not found))
                                     T T)))
                  [RESETSAVE NIL (LIST (FUNCTION CLOSEF?)
                                       (SETQ FILESTREAM (OPENSTREAM FULL 'INPUT]
                  (CL:FORMAT T "~A: " FULL)
                  (CL:MULTIPLE-VALUE-SETQ (ENV MAP TOP)
                         (OR (GET-ENVIRONMENT-AND-FILEMAP FILESTREAM)
                             (\PARSE-FILE-HEADER FILESTREAM)))
             ;; Get reader environment of file.  The call to GET-ENVIRONMENT-AND-FILEMAP with the filename will get cached info if it exists.
             ;; Otherwise, read the top of the file
                  (CL:WHEN ENV
                      (SETQ RDTBL (fetch (READER-ENVIRONMENT REREADTABLE) of ENV))
                      (\EXTERNALFORMAT FILESTREAM ENV))
                  (SETQ CASEINSENSITIVE (READTABLEPROP RDTBL 'CASEINSENSITIVE))
                  (SETQ SEPRCA (SEPRCASE DWIMIFYCOMPFLG RDTBL))
             ;; Escape matches .*, match need not be bracketed by seprs.  We test brackets after each hit, if needed. This avoids the slow case of a
             ;; CASEARRAY in FFILEPOS. Also more accurate, we don't match any string-internal sepr character to any other string internal sepr
             ;; character.
                  (CL:UNLESS (SETQ PATTERNS (CDR (ASSOC RDTBL FFILEPOSPATTERNS)))
                      [push FFILEPOSPATTERNS (CONS RDTBL
                                                   (SETQ PATTERNS
                                                    (for ATOM PREESCAPE POSTESCAPE in ATOMS
                                                       collect (CL:WHEN (SETQ PREESCAPE (EQ (CHCON1 ATOM)
                                                                                            (CHARCODE ESCAPE)))
                                                                   (SETQ ATOM (SUBSTRING ATOM 2)))
                                                             (CL:WHEN (SETQ POSTESCAPE
                                                                       (STRPOS (CONSTANT (CHARACTER (CHARCODE
                                                                                                     ESCAPE)))
                                                                               ATOM))
                                                                 (SETQ ATOM (SUBSTRING ATOM 1 (SUB1 POSTESCAPE))
                                                                  ))
                                                             (CL:WHEN (LITATOM ATOM)
                                                                 (LET ((*PACKAGE* (CL:SYMBOL-PACKAGE ATOM)))
                                                                ; Keep MKSTRING from putting a prefix on
                                                                      (SETQ ATOM (MKSTRING ATOM T RDTBL))))
                                                             (LIST ATOM PREESCAPE POSTESCAPE])
                  (for PATTERN CODE HIT (SEPRBASE _ (FETCH (ARRAYP BASE) OF SEPRCA))
                       (SEPRSIZE _ (FETCH (ARRAYP LENGTH) OF SEPRCA))
                       (SEPRFAT _ (EQ \ST.POS16 (FETCH (ARRAYP TYP) OF SEPRCA))) in PATTERNS
                     do
                     (SETFILEPTR FILESTREAM (SETQ I (OR TOP 0)))
                     ;; If the pattern characters  match, the pre and post characters must be seprs, unless escapes as noted and stripped above
                     (while [SETQ I (CDR (SETQ HIT (FFILEPOS (CAR PATTERN)
                                                        FILESTREAM I NIL NIL 'BOTH (CL:WHEN CASEINSENSITIVE
                                                                                          UPPERCASEARRAY]
                        when [AND [OR (CADR PATTERN)
                                      (PROGN (SETFILEPTR FILESTREAM (CAR HIT))
                                             (PROG1 (OR (NOT (SETQ CODE (\BACKCCODE FILESTREAM)))
                                                        (ZEROP (\CATRANSLATE SEPRBASE SEPRSIZE SEPRFAT CODE)))
                                                    (\SETFILEPTR FILESTREAM I]
                                  (OR (CADDR PATTERN)
                                      (NOT (SETQ CODE (\PEEKCCODE FILESTREAM T)))
                                      (ZEROP (\CATRANSLATE SEPRBASE SEPRSIZE SEPRFAT CODE]
                        do ;; The next search begins after the last search, since I is the tail of a match, even if the fileptr is set to 0 to get the
                           ;; map
                           (CL:UNLESS PRINTFLG                                  ; cause the printing of the filename to be saved on history list
                               (SETQ PRINTFLG T)
                               (LISPXPRIN2 FULL T T T)
                               ;; print with NODOFLG=T means just to record the printing;  the idea is that only those files in which something
                               ;; is found will be remembered on the history list
                               (LISPXPRIN1 ": " T NIL T))
                           (CL:UNLESS MAP
                               ;; After the first hit, use LOADFNS to try harder, perhaps scanning to create a map. Guard against a
                               ;; LOADFNS error
                               [NLSETQ (SETQ MAP (LOADFNS NIL FILESTREAM NIL 'FILEMAP]
                               ;; LOADFNS may implicitly close the file, so reopen for next hit.  Depending on the file device, we may not get
                               ;; the exact same stream, so make sure we close this one too.
                               [RESETSAVE NIL (LIST (FUNCTION CLOSEF?)
                                                    (SETQ FILESTREAM (OPENSTREAM FILESTREAM 'INPUT 'OLD
                                                                            `((EXTERNALFORMAT ,ENV]
                               (CL:UNLESS MAP                                   ; Set to T so only try and print once
                                   (LISPXPRIN1 " no filemap!" T)
                                   (SETQ MAP T)))
                           [OR [for X in (CDR (LISTP MAP))
                                  thereis (AND (ILESSP (CAR X)
                                                      I)
                                               (IGREATERP (CADR X)
                                                      I)
                                               (for Z in (CDDR X) thereis (COND
                                                                             ((AND (ILESSP (CADR Z)
                                                                                          I)
                                                                                   (IGREATERP (CDDR Z)
                                                                                          I))
                                                                              [COND
                                                                                 ((NOT (FMEMB (CAR Z)
                                                                                              FNS))
                                                                                  (SETQ FNS
                                                                                   (CONS (LISPXPRIN2 (CAR Z)
                                                                                                T T)
                                                                                         FNS]
                                                                              (SETQ I (CDDR Z))
                                                                              T]
                               (PROGN (LISPXPRIN2 I T T)
                                      (OR (FMEMB FILE OTHERSFILES)
                                          (SETQ OTHERSFILES (CONS FILE OTHERSFILES]
                           (LISPXSPACES 1 T)))
                  (COND
                     (PRINTFLG (LISPXTERPRI T))
                     (T (TERPRI T)))
                  (COND
                     ((NEQ COMS T)
                      (COND
                         ((OR FNS OTHERSFILES)
                          (EDITFROMFILE (OR (EQ MAP T)
                                            (DREVERSE FNS))
                                 FULL EDITPATTERN COMS (NULL OTHERSFILES))
                          (SETQ OTHERSFILES)
                          (SETQ FNS])]
     (COND
        ((EQ COMS T)
         (CONS OTHERSFILES FNS])

(EDITFROMFILE
  [LAMBDA (FNS FILES EDITPATTERN EDITCOMS ONLYTYPES)                    ; Edited 13-Jun-2021 10:24 by rmk:
    (RESETVARS [(EDITLOADFNSFLG (COND
                                   ((EQ EDITLOADFNSFLG T)
                                    '(T . NO))
                                   (T EDITLOADFNSFLG]
               (PROG NIL
                     [OR EDITCOMS (SETQ EDITCOMS (LIST (LIST 'EXAM EDITPATTERN]
                     (AND
                      (SETQ FILES (for FILE inside (OR FILES FILELST)
                                     when (OR (AND EDITLOADFNSFLG (FMEMB (ROOTFILENAME FILE)
                                                                         FILELST))
                                              (COND
                                                 ((EQ 'Y (ASKUSER DWIMWAIT 'Y (LIST "load from" FILE)
                                                                NIL T))
                                                  (LOADFROM FILE FNS 'ALLPROP)
                                                  T)))
                                     collect FILE))
                      (for TYPE in [COND
                                     ((LISTP ONLYTYPES))
                                     (ONLYTYPES '(FNS))
                                     (T  ;; Move FNS to the front.  This means that all the fns will be dwimified and edited before anything
                                         ;; else (like a rename of fields) is done.

                                        (CONS 'FNS (REMOVE 'FNS FILEPKGTYPES]
                         when (AND (LITATOM TYPE)
                                   (NEQ (fetch (FILEPKGTYPE EDITDEF) of TYPE)
                                        'NILL))
                         do
                         (PROG (SEEN)
                               (for FILE inside FILES
                                  do
                                  (for NAME in [COND
                                                 ((AND (EQ TYPE 'FNS)
                                                       (NEQ FNS T))
                                                                ; for this type, we are given the list of items
                                                  (PROG1 FNS (SETQ FNS NIL)))
                                                 (T             ; only want the values of 'TYPE' which are not part of some other
                                                                ; type
                                                    (FILECOMSLST FILE TYPE 'EDIT]
                                     unless (MEMBER NAME SEEN)
                                     do
                                     (ERSETQ
                                      (PROG [(DEF (OR (GETDEF NAME TYPE 'CURRENT '(NOCOPY NOERROR))
                                                      (GETDEF NAME TYPE 'SAVED '(NOCOPY NOERROR]

                                     ;; If definition has been loaded, it may have been editted.  Work on that explicitly instead of bringing in
                                     ;; a file definition to smash the users previous changes.  Perhaps we should query the user about this,
                                     ;; but until the interaction is worked out, it is better to avoid trashing his in core edits, given that he can
                                     ;; always get the file definition from permanent storage with LOADFNS.  --- We might also be more
                                     ;; discriminating about this: if the user specified a root file name, then he means the definition from the
                                     ;; definition group, not the physical file.  But ...  rmk

                                            (COND
                                               ((OR (AND (EQ TYPE 'FNS)
                                                         (NEQ FNS T))
                                                    (AND (LISTP DEF)
                                                         (LOOKIN DEF EDITPATTERN)))
                                                (COND
                                                   ((NULL SEEN)
                                                    (LISPXPRIN1 "editing the " T)
                                                    (LISPXPRIN1 (OR (fetch (FILEPKGTYPE DESCRIPTION)
                                                                       of TYPE)
                                                                    TYPE)
                                                           T)
                                                    (LISPXSPACES 1 T)))
                                                (SETQ SEEN (CONS NAME SEEN))
                                                (LISPXPRIN2 NAME T T)
                                                (LISPXPRIN1 ":
                                                       " T)
                                                (COND
                                                   ((NOT (ERSETQ (EDITDEF NAME TYPE
                                                                        (OR (AND DEF (CONS '= DEF))
                                                                            FILE)
                                                                        EDITCOMS)))
                                                    (LISPXPRIN1 "failed" T)))
                                                (LISPXTERPRI T])

(FINDATS
  [LAMBDA (X L)                                                          (* lmm "11-FEB-78 16:03")
    (COND
       ((NLISTP X)
        (FMEMB X L))
       (T (OR (FINDATS (CAR X)
                     L)
              (FINDATS (CDR X)
                     L])

(LOOKIN
  [LAMBDA (X PAT)                                                        (* lmm "11-MAR-78 14:20")
    (COND
       ([AND (EQ (CAR PAT)
                 '*ANY*)
             (EVERY (CDR PAT)
                    (FUNCTION (LAMBDA (X)
                                (AND (LITATOM X)
                                     (NOT (STRPOS 'Ÿ X]
        (FINDATS X (CDR PAT)))
       (T (EDITFINDP X PAT T])
)
(DEFINEQ

(SEPRCASE
  [LAMBDA (CLFLG RDTBL)
    ;; Edited 24-Jul-2022 15:39 by rmk

    ;; Edited 22-Jul-2022 20:54 by rmk: Ensured that quote and comma are seen as delimiters for every readtable and whether or not CLFLG.  This
    ;; increases recall, precision is not so important.

    ;; Edited 22-Jul-2022 20:51 by rmk                            (* bvm%: "24-Oct-86 18:16")

    ;; make a case array for FFILEPOS in which all of the seprs, breaks, and (possibly) clisp chars are all equivalent.

    (OR RDTBL (SETQ RDTBL FILERDTBL))
    (OR [ARRAYP (CDR (ASSOC RDTBL (COND
                                     (CLFLG CLISPCASEARRAYS)
                                     (T SEPRCASEARRAYS]
        (LET ((CA (CASEARRAY)))
             (for X in (NCONC (AND CLFLG (for Y in CLISPCHARS collect (CHCON1 Y)))
                             (GETSEPR RDTBL)
                             (GETBRK RDTBL))
                do (SETCASEARRAY CA X 0))
             (SETCASEARRAY CA (CHARCODE %')
                    0)
             (SETCASEARRAY CA (CHARCODE %,)
                    0)
             (if *PACKAGE*
                 then                                               ; symbols qualified with package prefix will otherwise be
                                                                    ; unfindable
                      (SETCASEARRAY CA (READTABLEPROP RDTBL 'PACKAGECHAR)
                             0))
             (SETQ CA (CONS RDTBL CA))
             (COND
                (CLFLG (push CLISPCASEARRAYS CA))
                (T (push SEPRCASEARRAYS CA)))
             (CDR CA])
)

(RPAQ? DEFAULTRENAMEMETHOD '(EDITCALLERS CAREFUL))

(RPAQ? SEPRCASEARRAYS )

(RPAQ? CLISPCASEARRAYS )

(MOVD? 'INFILEP 'FINDFILE)

;; or else from SPELLFILE

(DECLARE%: DONTEVAL@LOAD DOEVAL@COMPILE DONTCOPY

(BLOCK%: EDITFROMFILE EDITFROMFILE LOOKIN (GLOBALVARS EDITLOADFNSFLG)
       (NOLINKFNS LOADFROM))
)

(DECLARE%: DOEVAL@COMPILE DONTCOPY

(GLOBALVARS SYSFILES CLISPCASEARRAYS SEPRCASEARRAYS CLISPCHARS)
)

;; EXPORT

(DEFINEQ

(IMPORTFILE
  [LAMBDA (FILE RETURNFLG)                                          (* lmm " 6-Jun-86 17:43")
    (RESETLST
        [RESETSAVE NIL (LIST 'CLOSEF (SETQ FILE (OPENSTREAM FILE 'INPUT]
        (RESETSAVE (INPUT FILE))                                    ; Reset INPUT in case some form on the file's action is to read
                                                                    ; the next expression
        (NCONC [COND
                  ((EQ RETURNFLG T)                                 ; Just creating EXPORTS.ALL, don't side-effect the world
                   (IMPORTFILESCAN FILE RETURNFLG))
                  (T (LET (FILEPKGFLG DFNFLG)
                          (IMPORTFILESCAN FILE RETURNFLG]
               (IMPORTEVAL [LIST 'PUTPROP (KWOTE (ROOTFILENAME FILE))
                                  ''IMPORTDATE
                                  (LIST 'IDATE (GETFILEINFO FILE 'CREATIONDATE]
                      RETURNFLG)))])
```

```
(IMPORTEVAL
  [LAMBDA (FORM RETURNFLG)                                           ; Edited  2-May-87 18:57 by Pavel

    ;; Ignore DONTEVAL@LOAD'S --- If RETURNFLG is on, return list of forms

    (AND (LISTP FORM)
         (SELECTQ (CAR FORM)
              (DECLARE%: (for Z in (CDR FORM) join (IMPORTEVAL Z RETURNFLG)))
              (CL:EVAL-WHEN (for Z in (CDDR FORM) join (IMPORTEVAL Z RETURNFLG)))
              (/DECLAREDATATYPE                                      ; Ignore datatype initializations -- we only need the record
                                                                     ; declaration itself

                   NIL)
              (PROGN                                                 ; default: eval and/or return it
                     (AND (NEQ RETURNFLG T)
                          (EVAL FORM))
                     (AND RETURNFLG (LIST FORM])

(IMPORTFILESCAN
  [LAMBDA (FILE RETURNFLG)                                           (* bvm%: "24-Oct-86 19:31")
    (WITH-READER-ENVIRONMENT (GET-ENVIRONMENT-AND-FILEMAP FILE)
        (while (FFILEPOS BEGINEXPORTDEFSTRING FILE NIL NIL NIL T) bind DEF
           join (until (EQUAL (SETQ DEF (READ FILE))
                              ENDEXPORTDEFFORM)
                   join (IMPORTEVAL DEF RETURNFLG)))))])

(CHECKIMPORTS
  [LAMBDA (FILES NOASKFLG)                                           (* rmk%: "19-FEB-83 16:31")
                                                                     ; Loads exported definitions from new versions of FILES.
    (COND
       ((AND (SETQ FILES (for FILE inside FILES bind FULLFILENAME DATE
                              when [AND (SETQ FULLFILENAME (FINDFILE FILE T))
                                         (OR [NOT (SETQ DATE (GETPROP (ROOTFILENAME FILE)
                                                                     'IMPORTDATE]
                                             (NOT (IEQP DATE (GETFILEINFO FULLFILENAME 'ICREATIONDATE]
                              collect (LIST FILE FULLFILENAME)))
             (OR NOASKFLG (SELECTQ (ASKUSER 5 'Y (LIST "load new exports from " (MAPCAR FILES
                                                                                              (FUNCTION CAR)))
                                            '((Y "es
                                                  ")
                                              (N "o
                                                  "))
                                            T)
                                 (N NIL)
                                T)))
        (for FILE in FILES do (IMPORTFILE (CADR FILE])

(GATHEREXPORTS
  [LAMBDA (FROMFILES TOFILE FLG)                                     ; Edited 18-Jul-2023 23:39 by rmk
                                                                     ; Edited 22-May-2021 00:01 by rmk:

    ;; Use PRETTYDEF so that EXPORTS.ALL is registered with the normal file properties, reader environment, etc.

    (CL:UNLESS FLG (SETQ FLG T))
    (SETQ TOFILE (MKATOM TOFILE))                                    ; PRETTYDEF doesn't like strings--why?
    (RESETLST
        [PRETTYDEF NIL TOFILE `((E (MAPC (MKLIST FROMFILES)
                                            (FUNCTION (LAMBDA (F)
                                                         (MAPC (IMPORTFILE F FLG)
                                                               (FUNCTION PRINT])])

(\DUMPEXPORTS
  [NLAMBDA COMS                                                      (* bvm%: "24-Oct-86 19:42")

;;; Dumps an EXPORT form.  IMPORTFILE looks for a string announcing imports, but we must print it in a way that lets the file be loaded ok.

    (PRIN1 "(")
    (PRIN2 '*)
    (PRIN1 (SUBSTRING BEGINEXPORTDEFSTRING 2))                       ; BEGINEXPORTDEFSTRING starts with a * for benefit of
                                                                     ; IMPORTFILE
    (for TAIL on COMS do (PRETTYCOM (CAR TAIL)))
    (TERPRI)
    (PRINT ENDEXPORTDEFFORM)
    (TERPRI])
)

[PUTDEF 'EXPORT 'FILEPKGCOMS '((COM MACRO (X (E (\DUMPEXPORTS . X]

(RPAQ? BEGINEXPORTDEFSTRING "* %"FOLLOWING DEFINITIONS EXPORTED%")")

(RPAQ? ENDEXPORTDEFFORM '(* "END EXPORTED DEFINITIONS"))

(DECLARE%: DOEVAL@COMPILE DONTCOPY

(GLOBALVARS BEGINEXPORTDEFSTRING ENDEXPORTDEFFORM)
```

```
)

;; for GAINSPACE

(DEFINEQ

(CLEARFILEPKG
  [LAMBDA (FLG)                                                                 (* bvm%: "29-Aug-86 13:02")
    (PROG NIL
          (COND
             ((SELECTQ FLG
                    ((E T)
                        T)
                    (Y (TERPRI T)
                       (PRIN1 "you can delete just the filemaps -
                              " T)
                       (PROG1 [ASKUSER NIL NIL "are you sure you want to delete EVERYTHING ? "
                                           '((Y "es - everything" RETURN T)
                                             (N "o - just the filemaps" RETURN NIL)
                                             (E "verything" RETURN T)
                                             (F "ilemaps only" RETURN NIL]
                              (TERPRI T)))
                    NIL)
              (UPDATEFILES)
              [SETQ FILELST (SUBSET FILELST (FUNCTION (LAMBDA (FILE)
                                                         (COND
                                                            ((fetch TOBEDUMPED of (fetch FILEPROP of FILE))
                                                             (PRINT FILE T T)
                                                             (PRIN1 " has changes, not wiped." T)
                                                             (TERPRI T)
                                                             T)
                                                            (T (replace FILEPROP of FILE with NIL)
                                                               (replace FILECHANGES of FILE with NIL)
                                                               (SMASHFILECOMS FILE)
                                                               (NCONC1 SYSFILES FILE)
                                                              NIL]
              (SETQ LOADEDFILELST)))
          (SELECTQ FLG
               ((NIL T))
               (CLRHASH *FILEMAP-HASH*])
)

(ADDTOVAR GAINSPACEFORMS (FILELST "erase filepkg information" (CLEARFILEPKG RESPONSE)
                                  ((Y "es")
                                   (N "o")
                                   (E . "verything")
                                   (F "ilemaps only
                                      "))))

(DECLARE%: DOEVAL@COMPILE DONTCOPY

(GLOBALVARS SMASHPROPSLST1)
)

(DECLARE%: DOEVAL@COMPILE DONTCOPY

(GLOBALVARS %#UNDOSAVES ADDTOFILEKEYLST CLEANUPOPTIONS CLISPIFYPRETTYFLG COMPILE.EXT DECLARETAGSLST
       DEFAULTRENAMEMETHOD FILEPKGCOMSPLST FILERDTBL HISTORYCOMS HISTSTR0 I.S.OPRLST LISPXCOMS LISPXHISTORY
       LISPXHISTORYMACROS LISPXMACROS MACROPROPS MAKEFILEOPTIONS MAKEFILEREMAKEFLG MSDATABASELST PRETTYHEADER
       PRETTYTRANFLG SAVEDDEFS SYSPROPS USERMACROS USERRECLST USERWORDS FILEPKGTYPEPROPS)
)

(DECLARE%: DONTEVAL@LOAD DOEVAL@COMPILE DONTCOPY

(BLOCK%: DELFROMCOMS DELFROMCOMS DELFROMCOM DELFROMCOM1 REMOVEITEM (NOLINKFNS . T)
       (SPECVARS COMSNAME))

(BLOCK%: ADDTOFILEBLOCK ADDTOFILES? ADDTOFILE WHATIS ADDTOCOMS ADDTOCOM ADDTOCOM1 ADDNEWCOM (NOLINKFNS . T)
       (SPECVARS COMSNAME)
       (ENTRIES ADDTOFILE ADDTOCOMS ADDTOFILES? ADDTOCOM1))

(BLOCK%: INFILECOMS? INFILECOMS? INFILECOMTAIL INFILECOMS INFILECOM INFILECOMSVAL INFILECOMSVALS INFILEPAIRS
       INFILECOMSMACRO IFCPROPS IFCEXPRTYPE IFCPROPSCAN IFCDECLARE (LOCALFREEVARS NAME LITERALS VAL TYPE
                                                                           ONFILETYPE ORIGFLG)
       INFILECOMSPROP)

(BLOCK%: NIL MAKEFILE (LOCALVARS . T)
       (SPECVARS FILE OPTIONS REPRINTFNS SOURCEFILE FILETYPE FILEDATES CHANGES))

(BLOCK%: ADDFILE ADDFILE ADDFILE0)

(BLOCK%: FILEPKGCHANGES FILEPKGCHANGES (NOLINKFNS . T))

(BLOCK%: NIL ALISTS.WHENCHANGED CHANGECALLERS CLEANUP CLEARCLISPARRAY COMPARE COMPAREDEFS COMPILEFILES
       COMPILEFILES0 CONTINUEDIT COPYDEF DEFAULTMAKENEWCOM DELFROMFILES EDITDEF EXPRESSIONS.WHENCHANGED FILECOMS
       FILECOMSLST FILEFNSLST FILEPKGCOM FILEPKGCOMPROPS FILEPKGTYPE FILES? FILES?1 GETFILEPKGTYPE HASDEF
       INFILECOMTAIL LOADDEF MAKEALISTCOMS MAKEFILE1 MAKEFILES MAKEFILESCOMS MAKELISPXMACROSCOMS MAKENEWCOM
       MAKEPROPSCOMS MAKEUSERMACROSCOMS MARKASCHANGED MOVETOFILE POSTEDITPROPS PREEDITFN PRETTYDEFMACROS
       PROPS.WHENCHANGED PUTDEF RENAME SAVEDEF SAVEPUT SEARCHPRETTYTYPELST SHOWDEF SMASHFILECOMS TYPESOF
       UNMARKASCHANGED UNSAVEDEF UPDATEFILES (GLOBALVARS %#UNDOSAVES SYSFILES MARKASCHANGEDSTATS COMPILE.EXT
                                                  EDITMACROS EDITLOADFNSFLG LOADOPTIONS)
       (LOCALVARS . T))

(BLOCK%: DELDEF DELDEF DELFROMLIST (NOLINKFNS . T))

(BLOCK%: GETDEF GETDEF DWIMDEF GETDEFCOM GETDEFCOM0 GETDEFERR GETDEFCURRENT GETDEFFROMFILE GETDEFSAVED
       (RETFNS GETDEFCOM)
       (NOLINKFNS . T)
       (GLOBALVARS NOT-FOUNDTAG))
)

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

(ADDTOVAR NLAMA \DUMPEXPORTS MAKEUSERMACROSCOMS MAKEPROPSCOMS MAKELISPXMACROSCOMS MAKEFILESCOMS MAKEALISTCOMS
                      LISTFILES COMPILEFILES CLEANUP FILEPKGCOMPROPS PRETTYDEFMACROS)

(ADDTOVAR NLAML )

(ADDTOVAR LAMA FILEPKGTYPE FILEPKGCOM FILEPKGCHANGES)
)

(PUTPROPS FILEPKG COPYRIGHT ("Venue & Xerox Corporation" 1982 1983 1984 1985 1986 1987 1988 1989 1990 1991 1992
                                    1993 1995 2018 2022))
```

{MEDLEY}<sources>FILEPKG.;1 28-Jun-2024 18:34:03
-- Listed on 30-Jun-2024 13:15:47 --

FUNCTION INDEX

VARIABLE INDEX

PROPERTY INDEX

RECORD INDEX