```
(RPAQQ COMMON-MAKECOMS [;; FUNCTIONS TO HANDLE WRITING STANDARD COMMONLISP SOURCE FILES

                        (FNS COMMON-FILE-COMMAND COMMON-MAKEFILE)
                        (PROP MAKEFILE-ENVIRONMENT COMMON-MAKE)
                        (DECLARE%: DONTCOPY (ALISTS (EDITHISTALIST COMMON-MAKE])
```

```
;; FUNCTIONS TO HANDLE WRITING STANDARD COMMONLISP SOURCE FILES

(DEFINEQ

(COMMON-FILE-COMMAND
  [LAMBDA (COMMAND)                                        ; Edited 11-Dec-87 14:46 by DJVB
                                                           (* THE NEW COMMONLISP COMMANDS ARE MOSTLY
                                                           MACROS TO THINGS THIS HANDLES)

    (SELECTQ (SETQ TYPE (GETFILEPKGTYPE (CAR COMMAND)
                                        'COMMAND))
        (FNS [for FN in (PRETTYCOM1 COMMAND T T) bind DEF
                do (SETQ DEF (GETDEF FN 'FNS))
                   (CL:PPRINT (SELECTQ (CAR DEF)
                                  (CL:LAMBDA `(CL:DEFUN ,FN ,@(CDR DEF)))
                                  (LAMBDA `(CL:DEFUN ,FN (&OPTIONAL ,@(CADR DEF))
                                                    ,@(CDDR DEF)))
                                  (HELP "UNSUPPORTED LAMBDA" (CAR DEF])
        (DECLARE%: [FOR DEC IN (PRETTYCOM1 COMMAND T T) BIND (CND _ '(CL:LOAD CL:EVAL))
                     DO (SELECTQ DEC
                            ((EVAL@LOADWHEN EVAL@COMPILEWHEN COPYWHEN)
                                (HELP))
                            ((FIRST NOTFIRST))
                            (COMPILERVARS (RETURN))
                            ((COPY DOCOPY)
                                (SETQ CND (CL:ADJOIN 'CL:LOAD CND)))
                            ((DOEVAL@COMPILE EVAL@COMPILE)
                                (SETQ CND (CL:ADJOIN 'CL:COMPILE CND)))
                            ((DOEVAL@LOAD EVAL@LOAD)
                                (SETQ CND (CL:ADJOIN 'CL:LOAD CND)))
                            (DONTCOPY (SETQ CND (CL:REMOVE 'CL:LOAD CND)))
                            (DONTEVAL@COMPILE
                                (SETQ CND (CL:REMOVE 'CL:COMPILE CND)))
                            (DONTEVAL@LOAD
                                (SETQ CND (CL:REMOVE 'CL:EVAL CND)))
                            (PROGN (CL:FORMAT T "~&(EVAL-WHEN ~S " CND)
                                   (COMMON-FILE-COMMAND DEC)
                                   (CL:FORMAT T ")"])
        (SPECVARS [CL:PPRINT `(PROCLAIM (SPECIAL ,@(PRETTYCOM1 COMMAND T T])
        (GLOBALVARS [CL:PPRINT `(PROCLAIM (USER::GLOBAL ,@(PRETTYCOM1 COMMAND T T])
        (LOCALVARS [CL:PPRINT `(PROCLAIM (USER::LEXICAL ,@(PRETTYCOM1 COMMAND T T])
        ((PROP IFPROP)
            [LET ((IFFLG (EQ (CAR COMMAND)
                             PROP))
                  (PROP (CADR COMMAND))
                  (ATMS (PRETTYCOM1 (CDR COMMAND)
                                    T T)))
                 (IF (LISTP PROP)
                     THEN [FOR PRP IN PROP
                             DO (for ATM in ATMS when (OR IFFLG (GET ATM PRP))
                                   do (CL:PPRINT `(CL:SETF (GET ',ATM ',PRP)
                                                          ',(GET ATM PRP]
                     ELSEIF (EQ PROP 'ALL)
                        THEN                                (* ALL PROPERTIES)
                             [FOR ATM IN ATMS
                                DO (FOR PAIR ON (GETPROPLIST ATM) BY (CDDR PAIR)
                                      UNLESS (MEMB (CAR PAIR)
                                                   SYSPROPS)
                                      DO (CL:PPRINT `(CL:SETF [GET ',ATM ',(CAR PAIR]
                                                             ',(CADR PAIR]
                     ELSE (for ATM in ATMS when (OR (NOT IFFLG)
                                                    (GET ATM PROP))
                             do (CL:PPRINT `(CL:SETF (GET ',ATM ',PROP)
                                                    ',(GET ATM PROP])
        (PROPS [FOR AP in (PRETTYCOM1 (CDR COMMAND)
                                      T T)
                  do (CL:PPRINT `(CL:SETF [GET ',(CAR AP)
                                                ',(CADR AP]
```

```
                                                         ',(GET (CAR AP)
                                                                (CADR AP])
            (P (for PTHIS in (PRETTYCOM1 COMMAND T) do (CL:PPRINT PTHIS)))
            (MACROS (HELP "I THOUGHT YOU TRANSORED ALL THOSE MACROS" COMMAND))
            ((VARS ARRAY)
                [for VAR in (PRETTYCOM1 COMMAND T T)
                    do (CL:PPRINT (if (LITATOM VAR)
                                      then '(CL:DEFPARAMETER ,VAR ',(GETTOPVAL VAR))
                                      else '(CL:DEFPARAMETER ,(CAR VAR) ',(CADR VAR))])
            (INITVARS [FOR VAR IN (PRETTYCOM1 COMMAND T T)
                        DO (CL:PPRINT (IF (LITATOM VAR)
                                          THEN '(CL:DEFVAR ,VAR NIL)
                                          ELSE (IF (SUPERPRINTEQ (CAR VAR)
                                                                 COMMENTFLG)
                                                   THEN VAR
                                                   ELSE '(CL:DEFVAR ,@VAR)])
            (CONSTANTS [VARS (FOR VAR IN (PRETTYCOM1 COMMAND T T)
                               DO (CL:PPRINT (IF (LITATOM VAR)
                                                 THEN '(CL:DEFCONSTANT ,VAR ',(GETTOPVAL VAR))
                                                 ELSE '(CL:DEFCONSTANT ,(CAR VAR) ',(CADR VAR))])
            ((UGLYVARS HORRIBLEVARS)
                [LET ((*PRINT-CIRCLE* T))
                     (DECLARE (SPECVARS *PRINT-CIRCLE*))
                     (FOR VAR IN (PRETTYCOM1 COMMAND T T)
                       DO (CL:PPRINT '(DEFPARAMETER ,VAR ',(GETTOPVAL VAR])
            (ADDVARS [for AV in (PRETTYCOM1 COMMAND T T)
                        do (CL:PPRINT (if (CDDR AV)
                                          then '[SETQ ,(CAR AV)
                                                      (UNION ',(CDR AV)
                                                             ,(CAR AV]
                                          else '(CL:PUSHNEW ',(CADR AV)
                                                            ,(CAR AV])
            (APPENDVARS [FOR AV IN (PRETTYCOM1 COMMAND T T)
                           DO (CL:PPRINT '(SETQ ,(CAR AV)
                                               (CL:APPEND ,(CAR AV)
                                                          ',(CDR AV])
            (E (HELP "I HOPE THIS %"E%"KNOWS WHAT ITS DOING" COMMAND)
               (FOR EXP IN (PRETTYCOM1 COMMAND T) DO (EVAL EXP)))
            ((FILEPKGCOMS I.S.OPRS TEMPLATES BLOCKS EXPORT EDITHIST)
                                                            (* JUST IGNORE THESE)
                NIL)
            ((RECORDS INITRECORDS SYSRECORDS)
                (HELP "I THOUGHT YOU TRANSORED ALL THOSE RECORDS" COMMAND))
            (COMS (FOR COM IN (PRETTYCOM1 COMMAND T) DO (COMMON-FILE-COMMAND COM)))
            (ORIGINAL                                       (* COMS, BUT WITHOUT ANY USER DEFINED COMMANDS)
                    (LET* ((PRTTYTEM (PRETTYCOM1 COMMAND T))
                           (ORIGFLG T))
                          (DECLARE (SPECVARS ORIGFLG))
                          (for COM in PRTTYTEM do (COMMON-FILE-COMMAND COM))))
            (FILES                                          (* INSIDE LISTP%: FROM dir SOURCE COMPILED LOAD
                                                            LOADCOMP LOADFROM SYSLOAD PROP ALLPROP)
                                                            (* REQUIRE IS NOT IDENTICAL, BUTS IS AS CLOSE AS CL
                                                            GETS)
                [for F in (PRETTYCOM1 COMMAND T T) bind DIR PLACE
                    do (if (LISTP F)
                           then (if (SETQ PLACE (MEMB 'FROM F))
                                    then (SETQ DIR (LIST (CADR PLACE)))
                                    else (HELP "FILES OPTION?" F))
                           else (CL:PPRINT '(CL:REQUIRE ,F ,@DIR])
            (* (IF (EQ (CADR COMMAND)
                       '*)
                   THEN (BOUT *STANDARD-OUTPUT* (CHARCODE FORM))
                   ELSE (TERPRI)
                        (TERPRI)
                        (TERPRI))
               (PRINTDEF COMMAND NIL T)
               (TERPRI)
               (TERPRI))
            (LET (MACRO)
                 (if (SETQ MACRO (CDR (ASSOC (CAR COMMAND)
                                             PRETTYDEFMACROS)))
                     then (for COM in (SUBPAIR (CAR MACRO)
                                               (PRETTYCOM1 COMMAND T T)
                                               (CDR MACRO))
                               do (COMMON-FILE-COMMAND COM))
                     else (HELP "CAN'T HANDLE" (CAR COMMAND])


(COMMON-MAKEFILE
  [LAMBDA (FILE DEBUG)                                      ; Edited 11-Dec-87 13:25 by DJVB
    (PROG ((*PRINT-SEMICOLON-COMMENTS* 'ALL)
           (*PRINT-ARRAY* T)
           (*PRINT-STRUCTURE* T)
           **COMMENTFLG** FONTCHANGEFLG *PRINT-LENGTH* *PRINT-LEVEL* %#RPARS)
          (DECLARE (SPECVARS *PRINT-SEMICOLON-COMMENTS* *PRINT-ARRAY* *PRINT-STRUCTURE* **COMMENTFLG**
                             FONTCHANGEFLG *PRINT-LENGTH* *PRINT-LEVEL* %#RPARS))
          (RETURN (PROG [(*STANDARD-OUTPUT* (OPENSTREAM (PACKFILENAME 'EXTENSION 'LISP 'BODY FILE)
```

```
                                            'OUTPUT]
                    (DECLARE (SPECVARS *STANDARD-OUTPUT*))
                    (RETURN (CL:UNWIND-PROTECT
                                (PROG (DATES FILEILNAME PKGNAME BASE (*PACKAGE* *PACKAGE*)
                                             (*PRINT-BASE* *PRINT-BASE*)
                                             (*READTABLE* (FIND-READTABLE "LISP")))
                                    (DECLARE (SPECVARS *PACKAGE* *PRINT-BASE* *READTABLE*))
                                    (SETQ DATES (GETPROP (SETQ FILEILNAME (CL:INTERN (STRING FILE)
                                                                                      "IL"))
                                                         'FILEDATES))
                                    (SETQ PKGNAME (OR (LISTGET (GETPROP FILEILNAME 'MAKEFILE-ENVIRONMENT)
                                                               :PACKAGE)
                                                      "USER"))
                                    (SETQ BASE (OR (LISTGET (GETPROP FILEILNAME 'MAKEFILE-ENVIRONMENT)
                                                            :BASE)
                                                   10))
                                    (CL:FORMAT T ";;; -*- Mode: LISP; Syntax: Common-lisp; Package: ~A;
                                           Base: ~A -*-" PKGNAME BASE)
                                    (SETQ *PACKAGE* (CL:FIND-PACKAGE PKGNAME))
                                    (SETQ *PRINT-BASE* BASE)
                                    (CL:FORMAT T "~%%;;; File converted ~A from source ~A" (DATE)
                                           FILE)
                                    (AND DATES (CL:FORMAT T "~&;;; Original source ~A created ~A"
                                                      (CDAR DATES)
                                                      (CAAR DATES)))
                                    (for P in (LISTP (GETTOPVAL (FILECOMS FILE))) do (COMMON-FILE-COMMAND
                                                                                       P))
                                    (RETURN (FULLNAME *STANDARD-OUTPUT*)))
                                (CLOSEF *STANDARD-OUTPUT*))])
)

(PUTPROPS COMMON-MAKE MAKEFILE-ENVIRONMENT (:READTABLE "INTERLISP" :PACKAGE "INTERLISP"))

(DECLARE%: DONTCOPY

(ADDTOVAR EDITHISTALIST
         (COMMON-MAKE ("11-Dec-87 12:54:22" DJVB {DSK}<XAVIER>COMMON-MAKE.;1 (COMMON-FILE-COMMAND
                                                                                COMMON-MAKEFILE))
                 ("11-Dec-87 13:35:35" DJVB {DSK}<XAVIER>COMMON-MAKE.;2 (COMMON-FILE-COMMAND COMMON-MAKEFILE)
                     (GETTING DETAILS RIGHT))
                 ("11-Dec-87 13:40:48" DJVB {DSK}<XAVIER>COMMON-MAKE.;3 (COMMON-FILE-COMMAND))
                 ("11-Dec-87 14:09:04" DJVB {DSK}<XAVIER>COMMON-MAKE.;4 (COMMON-FILE-COMMAND))
                 ("11-Dec-87 14:48:44" DJVB {DSK}<XAVIER>COMMON-MAKE.;5 (COMMON-FILE-COMMAND)
                     (FIXED FILE COMMENTS AND CL:DEFVAR ET AL))))
)
```

FUNCTION INDEX

VARIABLE INDEX

PROPERTY INDEX