

File created: 2-Jun-87 14:51:46 {DSK}<LISPFILERS>WORK>EVAL-WHEN-PATCH.;1

changes to: (VARS EVAL-WHEN-PATCHCOMS)  
(FNS ADDTOCOM DELFROMCOM GETDEFCURRENT IMPORTEVAL INFILECOM)

Read Table: XCL

Package: INTERLISP

Format: XCCS

; Copyright (c) 1987 by Xerox Corporation. All rights reserved.

(RPAQQ EVAL-WHEN-PATCHCOMS ((FNS ADDTOCOM DELFROMCOM GETDEFCURRENT IMPORTEVAL INFILECOM)))

(DEFINEQ

(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 (CADDR 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 (CADDR COM))))
(MARKASCHANGED COMSNAME 'VARS)
T)))
(MACROS (COND
((AND (|for| PROP |inside| (CADDR COM) |always| (EQMEMB PROP MACROPROPS))
(|for| PROP |in| MACROPROPS
|always| (OR (EQMEMB PROP (CADDR 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 (CADDR 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))))))

```

**(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 (CADDR 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 (CDDDR COM)))
NAME TYPE))
(ALISTS PROPS)
(AND (EQ TYPE (CAR COM))
(COND
(EQ (CADDR 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))))))

```

**(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)))
    (|if| (EQMEMB 'EDIT OPTIONS)
          |then| (COPY DEF)
          |else| DEF)))

```

```

(FILEVAR (LITATOM NAME) ; 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 (CDDR X))
                (LIST X))))))
            (SETQ NOCOPY T))))))
    (MKPROGN DEF))))
  (|fetch| NULLDEF |of| TYPE))
  (GETDEFERR NAME TYPE OPTIONS))
  DEF))))

```

**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 (CDDR 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))))))

```

**(INFILECOM**

```

(LAMBDA (COM)                               ; Edited 2-May-87 19:03 by Pavel
  (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)
                                (LISTP NAME))
                           ; call from WHEREIS of a name which is a list
                           (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 IFFPROP)
   (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)))
  (CL:EVAL-WHEN (INFILECOMS (INFILECOMTAIL (CDR COM))))
  (DECLARE\ : ; skip over DECLARE: tags
   (RETURN (AND (NOT (FMEMB 'COMPILERVERS COM))
                (IFCDECLARE (INFILECOMTAIL COM)
                            (EQ TYPE 'DECLARE\ :))))))
  (ORIGINAL ; dont expand macros
   (PROG ((ORIGFLG T)
          (INFILECOMS (INFILECOMTAIL COM))))
  ((PROP IFFPROP) ; this currently does not handle 'pseudo-types' of PROP NAMES
   (SELECTQ TYPE

```

```

      (PROPS (IFCPROPSCAN (INFILECOMTAIL (CDR COM))
                (CADR COM)))
      (MACROS (INFILECOMSMACRO (INFILECOMTAIL (CDR COM))
                (CADR COM)))
      NIL))
(PROPS (RETURN (IFCPROPS COM)))
(MACROS (RETURN (SELECTQ TYPE
                  (PROPS (IFCPROPSCAN (INFILECOMTAIL COM)
                                      MACROPROPS))
                  (MACROS (INFILECOMSVALS (INFILECOMTAIL COM))
                          NIL))))
(ALISTS
      (RETURN (SELECTQ TYPE
                (ALISTS (INFILEPAIRS (INFILECOMTAIL COM))
                        NIL)))
      ; sigh. This should probably also 'coerce' when asking for
      ; LISPXMACROS, etc.
(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)
                                     |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) |when| (LISTP X)
         |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)
         |do| (COND
          ((LISTP X)
           (AND (CAR X)
                (INFILECOMSVAL (CAR X)
                              T)))
          (X (INFILECOMSVAL X (EQ COMNAME 'INITVARS))))))))
      (DEFS (RETURN (|for| X |in| (INFILECOMTAIL COM) |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)))
  ((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)
                        (CDR TEM))))))

```

)

**FUNCTION INDEX**

ADDTOCOM .....1    DELFROMCOM .....2    GETDEFCURRENT .....3    IMPORTEVAL .....4    INFILECOM .....5

---