

File created: 9-Sep-88 12:35:44 {DSK}<LISPFIL2>SYSTEM>MSPATCH.;5

changes to: (VARS MSPATCHCOMS)  
(RECORDS MSANALYZABLE MSSETPHRASE)  
(FNS MSEDITF MSSHOWUSE PatchCCode)

previous date: 9-Sep-88 11:57:36 {DSK}<LISPFIL2>SYSTEM>MSPATCH.;4

Read Table: INTERLISP

Package: INTERLISP

Format: XCCS

;;  
;; Copyright (c) 1988 by Savoir, Inc.. All rights reserved.

(RPAQO **MSPATCHCOMS**

```
[ (RECORDS MSANALYZABLE MSSETPHRASE)
  (FNS MSEDITF MSSHOWUSE PatchCCode)
  [P (PatchCCode '\MASTERSCOPEBLOCK/MSINTERPRET 530 ' (104)
      ' (105 104)]
  (PROP FILETYPE MSPATCH)
  (DECLARE%: DONTEVAL@LOAD DOEVAL@COMPILE DONTCOPY COMPILERVERS (ADDVARS (NLAMA)
                                                                    (NLAML)
                                                                    (LAMA PatchCCode))
```

(DECLARE%: EVAL@COMPILE

(RECORD MSANALYZABLE (FILEPKGNAME SETNAME GETDEF-FN MARKCHANGED-FN))

(RECORD MSSETPHRASE (DET (TYPE KNOWN . DEFAULTTYPE) . REP)  
(**TYPE?** (EQLLENGTH (CADR DATUM)  
2)))

)

(DEFINEQ

(**MSEDITF**

```
[LAMBDA ARGCOUNT ; Edited 9-Sep-88 12:24 by TAL
  (LET [(FNAME (ARG ARGCOUNT 1))
        (FEDITCOMS (for x from 2 to ARGCOUNT collect (ARG ARGCOUNT X)
        (DECLARE (SPECVARS MSFNTPES))
        (for FPTYPE in MSFNTPES bind FPNAME when (HASDEF FNAME (SETQ FPNAME (ffetch (MSANALYZABLE FILEPKGNAME)
                                                                    of FPTYPE)))
          do [if (EQ FPTYPE 'FNS)
                then (APPLY 'EDITF (CONS FNAME FEDITCOMS))
                else (EDITE (GETDEF FNAME FPNAME NIL ' (NOERROR NOCOPY EDIT))
                           FEDITCOMS FNAME FPNAME (FUNCTION (LAMBDA (NAME DEF TYPE EXITFLG)
                                                                ; this function is called when there were changes made
                                                                (MARKASCHANGED NAME TYPE 'CHANGED)
                                                                (PUTDEF NAME TYPE DEF]
          (RETURN FNAME]))
```

(**MSSHOWUSE**

```
[LAMBDA (SHOWFN SHOWTYPE SHOWSET SHOWEDIT IFCANT EDITCOMS) ; Edited 9-Sep-88 11:51 by TAL
  (PROG (DEF ANYFOUND)
    (DECLARE (LOCALVARS . T)
      (SPECVARS DEF ANYFOUND SHOWFN))
    (COND
      ([SETQ DEF (OR (MSGETDEF SHOWFN (AND (fetch (MSSETPHRASE KNOWN) of SHOWSET)
                                                (fetch (MSSETPHRASE TYPE) of SHOWSET))
        (COND
          ((EQ SHOWEDIT 'SHOW)
            '?)
            (T 'CURRENT))
          ' (NOERROR NODWIM NOCOPY))
        (AND (EQ SHOWEDIT 'EDIT)
          (LET ((FILE (EDITLOADFNS? SHOWFN)))
            (COND
              (FILE (LOADFNS SHOWFN FILE 'PROP)
                    (GETPROP SHOWFN 'EXPR]
                    ; was (MSGETDEF SHOWFN IFCANT (EQ SHOWEDIT
                    ; (QUOTE SHOW)))
                    ; The SHOW command does not need to save
          (\MASTERSCOPEBLOCK/MSUPDATEFN1 SHOWFN DEF
            (LIST SHOWTYPE [FUNCTION (LAMBDA (ITEM SS SE PRNT INCLISP)
              (COND
                ((MSMEMBSET ITEM SS)
                  (COND
                    ((NOT ANYFOUND)
                     (TAB 0 0 T)
                     (PRIN2 SHOWFN)
                     (PRIN1 " :
                          ")
                    ))
```

```

      (SETQ ANYFOUND (CONS (CONS PRNT
                                (AND INCLISP
                                      (NOT (MSFIND INCLISP PRNT))
                                      INCLISP))
                            ANYFOUND))
      (COND
        ([AND (EQ SE 'SHOW)
              (NOT (FASSOC PRNT (CDR ANYFOUND))
                (SPACES 3)
                (LVLPRINT PRNT (OUTPUT)
                          2)
                (COND
                  ((CDAR ANYFOUND)
                   ; This is under a clisp
                   (PRIN1 " {under " )
                   (LVLPRIN2 INCLISP (OUTPUT)
                               2)
                   (PRIN1 " }
                           "])
                SHOWSET SHOWEDIT)))
        (T (printout T "Can't find a definition for " SHOWFN "!" T)
          (RETURN)))
      (COND
        ((NOT ANYFOUND)
         (RETURN))
        ((EQ SHOWEDIT 'EDIT)
         [MAPC ANYFOUND (FUNCTION (LAMBDA (X)
                                   ;; Definer definitions get copied, so == pattern loses.
                                   ;; (FRPLNODE X (QUOTE ==) (OR (CDR X) (CAR X)))
                                   (FRPLNODE2 X (OR (CDR X)
                                                    (CAR X]
                                   (SETQ ANYFOUND (CONS '*ANY* ANYFOUND))
                                   (PRINT [APPLY* 'MSEDITF SHOWFN (LIST 'BIND ' (E (SETQ %#1)
                                          T)
                                          (LIST 'F ANYFOUND T)
                                          (LIST 'LPQ (LIST 'IF ' (NEQ (%##)
                                                                    %#1)
                                                                    [LIST ' (ORR (P)
                                                                    NIL)
                                                                    ' (S %#1)
                                                                    (COND
                                                                      (EDITCOMS (CONS 'BIND EDITCOMS))
                                                                      (T 'TTY%:]
                                                                    NIL)
                                          (LIST 'F ANYFOUND 'N]
                                          T T)))
        (RETURN T])

```

**(PatchCCode**

; Edited 7-Sep-88 15:24 by TAL

```

[CL:LAMBDA (Fn Offset NewBytes &OPTIONAL OldBytes)
  (LET ((tmp (GETD Fn)))
    (COND
      ((AND (CCODEP tmp)
            (SETQ tmp (\GETBASEPTR tmp 0))
            (for off from Offset as x inside OldBytes always (EQ (\GETBASEBYTE tmp off)
                                                                    x)))
       (for off from Offset as x inside NewBytes do (\PUTBASEBYTE tmp off x))
      T])
)

```

```

(PatchCCode '\MASTERSCOPEBLOCK/MSINTERPRET 530 '(104)
 '(105 104))

```

```

(PUTPROPS MSPATCH FILETYPE :COMPILE-FILE)

```

```

(DECLARE%: DONTVAL@LOAD DOEVAL@COMPILE DONTCOPY COMPILEVAR%

```

```

(ADDTTOVAR NLAMA )

```

```

(ADDTTOVAR NLAML )

```

```

(ADDTTOVAR LAMA PatchCCode)

```

```

)

```

```

(RPAQQ MSPATCHCOMS

```

```

  [(RECORDS MSANALYZABLE MSSETPHASE)
   (FNS MSEDITF MSSHOWUSE PatchCCode)
   [P (PatchCCode '\MASTERSCOPEBLOCK/MSINTERPRET 530 '(104)
                 '(105 104]
   (PROP FILETYPE MSPATCH)

```

```
{MEDLEY}<loops>obsolete>from1.1>patches>MSPATCH.;1  (MSPATCHCOMS cont.)
```

Page 3

```
(DECLARE%: DONTEVAL@LOAD DOEVAL@COMPILE DONTCOPY COMPILERVERS (ADDVARS (NLAMA)
                                                                    (NLAML)
                                                                    (LAMA PatchCCode MSEDITF]))
```

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

```
(ADDTOVAR NLAMA )
```

```
(ADDTOVAR NLAML )
```

```
(ADDTOVAR LAMA PatchCCode MSEDITF)
)
```

```
(PUTPROPS MSPATCH COPYRIGHT ("Savoir, Inc." 1988))
```

---

FUNCTION INDEX

MSEDITF .....1      MSSHOWUSE .....1      PatchCCode .....2

---

RECORD INDEX

MSANALYZABLE .....1      MSSETPHRASE .....1

---

PROPERTY INDEX

MSPATCH .....2

---