File created: 18-Oct-93 14:37:11 **{Pele:mv:envos}<LispCore>Sources>CLTL2>CMLPROGV.;2**

previous date: 3-Sep-91 17:48:59 {Pele:mv:envos}<LispCore>Sources>CLTL2>CMLPROGV.;1

Read Table: INTERLISP

Package: INTERLISP

Format: XCCS

```
;;
;; Copyright (c) 1986, 1987, 1990, 1991, 1993 by Venue & Xerox Corporation.  All rights reserved.

(RPAQQ CMLPROGVCOMS ((FNS \DO.PROGV \DO.PROGV.SETUP.FRAME.AND.EXECUTE)
                     (SPECIAL-FORMS CL:PROGV)
                     (PROP DMACRO CL:PROGV)
                     (PROP FILETYPE CMLPROGV)))

(DEFINEQ

(\DO.PROGV
  [LAMBDA (VARS VALUES FNTOCALL)                                    ; Edited 21-Jan-91 17:10 by jds
    ;; call FNTOCALL after binding VARS to VALUES
    (DECLARE (LOCALVARS . T))
    (LET ((NVARS 0)
          NTSIZE NNILS TMP)
         (for VAR in VARS do ;; Count number of vars to bind, check their validity
                             (CHECK-BINDABLE VAR)
                             (add NVARS 1))
         (.CALLAFTERPUSHINGNILS. (SETQ NNILS (IPLUS NVARS (SETQ NTSIZE (CEIL [ADD1 (UNFOLD NVARS (CONSTANT
                                                                                                  (
                                                                                                  WORDSPERNAMEENTRY
                                                                                                  ]
                                                                                                  WORDSPERQUAD))
                                                                                    (FOLDHI (fetch (FNHEADER OVERHEADWORDS) of T)
                                                                                           WORDSPERCELL)
                                                                                    (SUB1 CELLSPERQUAD)))
                                 (\DO.PROGV.SETUP.FRAME.AND.EXECUTE NNILS NVARS NTSIZE VARS VALUES))
         (CL:FUNCALL FNTOCALL])

(\DO.PROGV.SETUP.FRAME.AND.EXECUTE
  [LAMBDA (NNILS NVARS NTSIZE VARS VALUES)                          ; Edited 30-Jan-91 19:02 by jds
    (DECLARE (LOCALVARS . T))
    (PROG ((CALLER (\MYALINK))
           NILSTART NT HEADER)

;;; Create a nametable inside CALLER where \DO.PROGV pushed all those NILs

          (SETQ HEADER (fetch (FX FNHEADER) of CALLER))             ; The function header of code for \DO.PROGV
          (SETQ NT (ADDSTACKBASE (CEIL (IPLUS (SETQ NILSTART (IDIFFERENCE (fetch (FX NEXTBLOCK) of CALLER)
                                                                          (UNFOLD NNILS WORDSPERCELL)))
                                              (UNFOLD NVARS WORDSPERCELL))
                                       WORDSPERQUAD)))
      ;; Address of our synthesized nametable: beginning of NIL's, not counting additional PVARs we are about to bind, rounded up to quadword
          (for VAR in VARS as VAR# from (FOLDLO (IDIFFERENCE NILSTART (fetch (FX FIRSTPVAR) of CALLER))
                                                WORDSPERCELL)
             as NT1 from (fetch (FNHEADER OVERHEADWORDS) of T) by (CONSTANT (WORDSPERNAMEENTRY)) as NT2
             from (IPLUS (fetch (FNHEADER OVERHEADWORDS) of T)
                         NTSIZE)
             by (CONSTANT (WORDSPERNTOFFSETENTRY)) as VALUEOFF from NILSTART by WORDSPERCELL
             do [PUTBASEPTR \STACKSPACE VALUEOFF (COND
                                                    (VALUES (pop VALUES))
                                                    (T 'NOBIND]
                (SETSTKNAME-RAW NT NT1 (\ATOMVALINDEX VAR))
                (SETSTKNTOFFSET-RAW NT NT2 PVARCODE VAR#))

;;; now fix up header of NT

          (replace (FNHEADER FRAMENAME) of NT with '\PROGV)
          (replace (FNHEADER NTSIZE) of NT with NTSIZE)
          (replace (FX NAMETABLE) of CALLER with NT])

)

(DEFINE-SPECIAL-FORM CL:PROGV (CL::VARIABLES CL:VALUES &REST CL::$$PROGV-FORMS &ENVIRONMENT
                                             CL::$$PROGV-ENVIRONMENT)
    ;; $$PROGV-FORMS and $$PROGV-ENVIRONMENT are named this wierd way because the interpreter is still compiled with the ByteCompiler and
    ;; those variables will eventually be made special by that compiler.  They can get normal names whenever the new compiler starts being used on this
    ;; file.
    [\DO.PROGV (CL:EVAL CL::VARIABLES CL::$$PROGV-ENVIRONMENT)
```

```
          (CL:EVAL CL:VALUES CL::$$PROGV-ENVIRONMENT)
          #'(CL:LAMBDA NIL (\EVAL-PROGN CL::$$PROGV-FORMS CL::$$PROGV-ENVIRONMENT])

(PUTPROPS CL:PROGV DMACRO [(VARIABLES VALUES . FORMS)
                           (\DO.PROGV VARIABLES VALUES #'(LAMBDA NIL . FORMS])

(PUTPROPS CMLPROGV FILETYPE CL:COMPILE-FILE)

(PUTPROPS CMLPROGV COPYRIGHT ("Venue & Xerox Corporation" 1986 1987 1990 1991 1993))
```

FUNCTION INDEX

PROPERTY INDEX

MACRO INDEX

SPECIAL-FORM INDEX
