

File created: 14-Sep-2023 22:53:09 {WMEDLEY}<sources>PRINTFN.;35

edit by: rmk

changes to: (FNS PF)

previous date: 19-Jun-2022 00:02:19 {WMEDLEY}<sources>PRINTFN.;34

Read Table: INTERLISP

Package: INTERLISP

Format: XCCS

::
:: Copyright (c) 1986-1987, 1990, 1999, 2018, 2021 by Venue & Xerox Corporation.

(RPAQQ **PRINTFNCOMS**

```
[ (* * PRINTFN)
 (FNS PF PF* PRINTFN PRINTFNDEF FINDFNDEF PFCOPYBYTES DISPLAYP)
 (INITVARS (PFDEFAULT 'PFCOPYBYTES))
 (DECLARE%: DONTCOPY (MACROS PFPRINCHAR PFOUTCHAR))
 (P (MOVD? 'COPYBYTES 'PFCOPYBYTES))
 (USERMACROS PF)
 (GLOBALVARS **COMMENT**FLG LASTWORD PFDEFAULT FILERDTBL USEMAPFLG)
 (DECLARE%: DONTEVAL@LOAD DOEVAL@COMPILE DONTCOPY COMPILERVARS (ADDVARS (NLAMA PF* PF)
 (NLAML)
 (LAMA]))
```

(* * PRINTFN)

(DEFINEQ

(**PF**

[NLAMBDA FN

; Edited 14-Sep-2023 22:52 by rmk
; Edited 4-Apr-2018 11:13 by rmk:

:: RMK; Fixed to skip compiled files, also to use FUNCTIONS as well as FNS. That might not help, if FUNCTIONS are not included in the filemap.
:: Print from files known to masterscope database before looking at whereis database. Note, however, that it also prefers the masterscope
:: database to incore files
:: If FN is NIL, prints the function named by LASTWORD
:: If FN is a list, then extra args are interpreted as:
:: OUTPUT FILE
:: ...

(RESETLST

```
(PROG (OUT OTHERARGS IFILES)
 (SETQ FN (NLAMBDA.ARGS FN))
 [COND
 ((LISTP FN)
 (SETQ OTHERARGS (CDR FN))
 (SETQ FN (CAR FN))
```

; Grab the args as a list
; If it's a list, take the first element as the function name.

```
(COND
 (FN
 (SETQ LASTWORD FN))
 (T
 (SETQ FN LASTWORD)))
```

; FN name specified; use it.
; Not specified, use LASTWORD

```
[SETQ IFILES (OR (CAR OTHERARGS)
 (APPEND (WHEREIS FN 'FNS T)
 (WHEREIS FN 'FUNCTIONS T))
```

```
[RESETSAVE (OUTPUT (COND
 ((CADR OTHERARGS) ; An output file was specified; if not open for output, open it.
 (OR (OPENP (CADR OTHERARGS)
 'OUTPUT)
 (WINDOWP (CADR OTHERARGS))
 (PROGN [RESETSAVE (SETQ OUT (OPENSTREAM (CADR OTHERARGS)
 'OUTPUT))
 ' (PROGN (CLOSEF? OLDVALUE)
 OUT))))
 (T ; otherwise, use primary output.
 T) ; skip compiled files
```

```
(FOR FILE INSIDE IFILES UNLESS (MEMB (FILENAMEFIELD FILE 'EXTENSION)
 *COMPILED-EXTENSIONS*))
 DO (PRINTFN FN FILE))))]
```

(**PF***

[NLAMBDA FN

; Edited 10-Jun-87 11:09 by jds

::: Print the function FN (or LASTWORD), with comments visible to the user.

```
(RESETVARS (**COMMENT**FLG)
 (APPLY (FUNCTION PF)
 FN])
```

(PRINTFN

```
[LAMBDA (FN FROMFILE TOFILE)
  (PROG ((LOC (FINDFDEF FN FROMFILE)))
    (COND
      ((LISTP LOC)
        (PRINTFDEF (CAR LOC)
          TOFILE
          (CADR LOC)
          (CADDR LOC)
          (CADDR LOC))
        (RETURN FN))
      ((EQ LOC 'FILE.NOT.FOUND)
        (printout TOFILE "file " FROMFILE " not found." T))
      (T (printout TOFILE FN " not found on " LOC "." T]))
```

; Edited 17-Oct-2021 18:00 by rmk:

(PRINTFDEF

```
[LAMBDA (SRCFIL DSTFIL START END TYPE)
  ;; RMK: It wasn't clear what PFDEFAULT was doing, or why. I've assigned it a meaning here: the name of the function to call to print a function
  ;; on a display stream. Initialized to PFCOPYBYTES
  (RESETLST
    (LET (TEM)
      [COND
        ((SETQ TEM (GETSTREAM DSTFIL 'OUTPUT T))
          (SETQ DSTFIL TEM))
        (T (RESETSAVE (SETQ DSTFIL (OPENSTREAM DSTFIL 'OUTPUT))
          '(PROGN (CLOSEF? OLDVALUE))
          [COND
            ((SETQ TEM (GETSTREAM SRCFIL 'INPUT T))
              (RESETSAVE NIL (LIST 'SETFILEPTR TEM (GETFILEPTR TEM)))
              (SETQ SRCFIL TEM))
            (T (RESETSAVE (SETQ SRCFIL (OPENSTREAM SRCFIL 'INPUT))
              '(PROGN (CLOSEF? OLDVALUE))
              (PRINTOUT DSTFIL "{from " .P2 (FULLNAME SRCFIL)
                "}" T))
            (APPLY* (CL:IF (DISPLAYP DSTFIL)
              PFDEFAULT
              (FUNCTION COPYBYTES))
              SRCFIL DSTFIL START END)
            (TERPRI DSTFIL))]]
```

; Edited 7-Oct-2021 20:51 by rmk:

(FINDFDEF

```
[LAMBDA (FN FROMFILE)
```

; Edited 15-Mar-2022 00:18 by rmk: Changed FINDFILE to FINDFILE-WITH-EXTENSIONS (bvm%:"27-Aug-86 16:27")

;;; Locates FNS definition of FN on FROMFILE. If found, returns a list (file start end type); if file not found, returns symbol FILE.NOT.FOUND; if file found but not fn, returns full name of file that was found

(* * "Locates FNS definition of FN on FROMFILE. If found, returns a list (file start end type); if file not found, returns symbol FILE.NOT.FOUND; if file found but not fn, returns full name of file that was found")

```
(LET (FULL MAP VALUE)
  (COND
    ((NOT (SETQ FULL (FINDFILE-WITH-EXTENSIONS FROMFILE)))
      'FILE.NOT.FOUND)
    [(COND
      ((SETQ MAP (OR (GETFILEMAP FULL)
        (LOADFILEMAP FULL)))
        ;; First clause is quick check when the file already has a map. LOADFILEMAP will find file map, rebuild if necessary and rewrite it
        ;; on file if updatemapflg is T.
        (AND (for GROUP in (CDR MAP) thereis (SETQ VALUE (FASSOC FN GROUP)))
          (LIST FULL (CADR VALUE)
            (CDDR VALUE)
            'MAP)
          (T FULL])
```

(PFCOPYBYTES

```
[LAMBDA (SRCFIL DSTFIL START END NOTERPRI)
```

; Edited 19-Jun-2022 00:01 by rmk: Changed #CHARS to NBYTES, to be clear about what we are counting

; Edited 2-Dec-2021 13:27 by rmk:

; Edited 8-Oct-2021 00:17 by rmk:

; Edited 24-Mar-93 14:16 by rmk:

; RMK: Added NOTERPRI to at least give caller control over whether a TERPRI is done just in the case of copying the whole file.

; RMK: What does FLG do? It isn't referenced. It seems to be passed as the value of PFDEFAULT from PRINTFDEF, and that variable is initialized to NIL. I'm removing it.

; copy from SRCFIL to DSTFIL, paying attention to font changes. Other stuff about truncating lines gone away. Interprets all possible EOL conventions as EOL. Has to call \INCHAR-\INCCODE macros in order to keep track of character count--READDCCODE doesn't do that.

:: If END is NIL and START is given, then START is the number of characters to copy from the current file position. Otherwise, copy to the end of the file.

```
(DECLARE (GLOBALVARS CHANGECHAR COMMENTFLG **COMMENT**FLG))
(RESETLST
  (PROG ((SSTRM (\INSTREAMARG SRCFIL))
        (DSTRM (\OUTSTREAMARG DSTFIL))
        FONTARRAY CHARCODE NBYTES MAXFONT)
    (DECLARE (SPECVARS . T)) ; In particular, NBYTES must be a specvar for \INCCODE
    (COND
      ((IMAGESTREAMAMP DSTRM)
       (SETQ FONTARRAY (FONTMAPARRAY))
       (SETQ MAXFONT (ARRAYSIZE FONTARRAY))
       (RESETSAVE NIL (LIST (FUNCTION DSPFONT)
                            (DSPFONT NIL DSTRM)
                            DSTRM))
       (DSPFONT (ELT FONTARRAY 1)
                DSTRM)))
      [SETQ NBYTES (COND
                   (END (SETFILEPTR SSTRM START)
                        ;; Doesn't call \SETFILEPTR cause START has to be checked
                        (IDIFFERENCE (COND
                                     ((EQ END -1)
                                      (GETEOFPTR SSTRM))
                                     (T END))
                                     START))
                   (START)
                   (T
                    (SETQ START (GETFILEPTR SSTRM))
                    (IDIFFERENCE (GETEOFPTR SSTRM)
                                 (GETFILEPTR SSTRM))
                    ; Copy everything from here to the end-of-file
                    ))
      (COND
        ((ILEQ NBYTES 0)
         (RETURN T)) ; Nothing to do
      LP (COND
        ((ILEQ NBYTES 0)
         (CL:WHEN (AND (EQ START 0)
                       (EOF SSTRM))
                  ; RMK: We copied the whole file, why should we do a TERPRI
                  (OR NOTERPRI (TERPRI DSTRM)))
         (RETURN T))
        (SETQ CHARCODE (\INCCODE.EOLC SSTRM ANY.EOLC 'NBYTES NBYTES))
        (IF (EQ CHARCODE (CONSTANT (CHARCODE.DECODE FONTESCAPECHAR)))
            THEN ; No EOL check on font character, otherwise we would be limited to 9 fonts
              (SETQ CHARCODE (\INCCODE SSTRM 'NBYTES NBYTES))
              (CL:WHEN (AND (IGEQ MAXFONT CHARCODE)
                            (NEQ CHARCODE 0))
                       (DSPFONT (ELT FONTARRAY CHARCODE)
                                 DSTRM))
            ELSE (\OUTCHAR DSTRM CHARCODE))
        (GO LP))))]
```

(DISPLAYP

```
[LAMBDA (STREAM) ; (* AJB "23-Sep-85 14:53")
  (LET ((STRM (\OUTSTREAMARG STREAM T))
        (AND STRM (OR (DISPLAYSTREAMAMP STRM)
                      (IMAGESTREAMAMPTYPEP STRM 'TEXT))
  )
```

(RPAQ? PFDEFAULT 'PFCOPYBYTES)

(DECLARE%: DONTCOPY

(DECLARE%: EVAL@COMPILE

```
(PUTPROPS PFPRINCHAR MACRO ((CC)
  (COND
    (EOLFLG (TERPRI DSTRM)
             (SETQ EOLFLG NIL)
             (SETQ HPOS LMAR)))
    (COND
      ((NOT (ZEROP %#SPACES))
       (FRPTQ (COND
               ((OR FLG STRFLG)
                %#SPACES)
               (T (FOLDHI %#SPACES 2)))
              (PFOUTCHAR (CHARCODE SPACE)))
              (SETQ %#SPACES 0))
       (PFOUTCHAR CC)))
  (PFOUTCHAR CC)))
```

```
(PUTPROPS PFOUTCHAR MACRO ((CC)
  ([LAMBDA (WIDTH)
   (COND
     ((AND WIDTH (IGREATERP (add HPOS WIDTH)
```

```

                                RMAR)) (* past RIGHT margin, force eol)
                                (TERPRI DSTRM)
                                (SETQ HPOS WIDTH))
                                (\OUTCHAR DSTRM CC]
                                (\STREAMCHARWIDTH CC DSTRM \PRINTERMTABLE)))
)
)
(MOVD? 'COPYBYTES 'PFCOPYBYTES)
(ADDTOVAR EDITMACROS [PF NIL (ORR [(E (APPLY* 'PF (FIRSTATOM (%##]
                                (E 'PF?])
(ADDTOVAR EDITCOMSA PF)
(DECLARE%: DOEVAL@COMPILE DONTCOPY
(GLOBALVARS **COMMENT**FLG LASTWORD PFDEFAULT FILERDTBL USEMAPFLG)
)
(DECLARE%: DONTEVAL@LOAD DOEVAL@COMPILE DONTCOPY COMPILERVERS
(ADDTOVAR NLAMA PF* PF)
(ADDTOVAR NLAML )
(ADDTOVAR LAMA )
)
(PUTPROPS PRINTFN COPYRIGHT ("Venue & Xerox Corporation" 1986 1987 1990 1999 2018 2021))
```

FUNCTION INDEX

DISPLAYP3 FINDFNDEF ...2 PF1 PF*1 PFCOPYBYTES .2 PRINTFN2 PRINTFNDEF ..2

VARIABLE INDEX

EDITCOMSA ...4 EDITMACROS ..4 PFDEFAULT ...3

MACRO INDEX

PFOUTCHAR ...3 PFPRINCHAR ..3
