

File created: 28-Apr-92 17:29:44 {PELE:MV:ENVOS}<LISPCORE>SOURCES>NSPRINT.;3

changes to: (FNS \NSPRINT.INTERNAL)

previous date: 16-May-90 20:54:31 {PELE:MV:ENVOS}<LISPCORE>SOURCES>NSPRINT.;2

Read Table: INTERLISP

Package: INTERLISP

Format: XCCS

::
:: Copyright (c) 1984, 1985, 1986, 1987, 1990, 1992 by Venue & Xerox Corporation. All rights reserved.

(RPAQQ **NSPRINTCOMS**

```
[ (COMS (COURIERPROGRAMS PRINTING)
  (DECLARE%: DONTCOPY (RECORDS NSPRINTER)
    (GLOBALVARS NS.DEFAULT.PRINTER NSPRINT.DEFAULT.MEDIUM NSPRINT.WATCHERFLG))
  (INITVARS (NS.DEFAULT.PRINTER NIL)
    (NSPRINT.DEFAULT.MEDIUM)
    (NSPRINT.WATCHERFLG T))
  (FNS GETNSPRINTER NSPRINT \NSPRINT.INTERNAL \NSPRINT.MEDIUM.CHECK \NSPRINT.UNSUPPORTED
    NSPRINTER.HOSTNAMEP NSPRINTER.STATUS NSPRINTER.PROPERTIES NSPRINTREQUEST.STATUS
    \NSPRINT.ENQUIRE \NSPRINT.COURIER.OPEN))
  (COMS ; Printer watcher
    (FNS \NSPRINT.WATCHDOG \NSPRINT.WATCH.JOB \NSPRINT.FULL.REQUEST.STATUS)
    (GLOBALVARS *PRINT-JOBS-IN-PROGRESS*)
    (INITVARS (*PRINT-JOBS-IN-PROGRESS*)))
  (COMS ; FAX
    (FNS FAX.SEND.FILE FAX.STATUS FAX.PROPERTIES FAX.HOSTNAMEP \FAX.PARSE.NAME)
    (INITVARS (DEFAULTFAXHOST)
      (FAXADDRESSES)
      (FAX.NO.WATCHER T))
    (GLOBALVARS DEFAULTFAXHOST FAXADDRESSES FAX.NO.WATCHER)
    (ADDVARS (PRINTERTYPES ((FAX TELECOPIER)
      (CANPRINT (INTERPRESS))
      (HOSTNAMEP FAX.HOSTNAMEP)
      (STATUS FAX.STATUS)
      (PROPERTIES FAX.PROPERTIES)
      (SEND FAX.SEND.FILE)
      (BITMAPSCALE INTERPRESS.BITMAPSCALE)
      (BITMAPFILE (INTERPRESSBITMAP FILE BITMAP SCALEFACTOR REGION ROTATION
        TITLE]))
```

(COURIERPROGRAM **PRINTING** (4 3)
TYPES

```
[ (REQUEST.ID (ARRAY 5 UNSPECIFIED))
  [PRINT.ATTRIBUTES (SEQUENCE (CHOICE (PRINT.OBJECT.NAME 0 STRING)
    (PRINT.OBJECT.CREATE.DATE 1 TIME)
    (SENDER.NAME 2 STRING))
  [PRINT.OPTIONS (SEQUENCE (CHOICE (PRINT.OBJECT.SIZE 0 LONGCARDINAL)
    (RECIPIENT.NAME 1 STRING)
    (MESSAGE 2 STRING)
    (COPY.COUNT 3 CARDINAL)
    (PAGES.TO.PRINT 4 (RECORD (BEGINNING.PAGE.NUMBER CARDINAL)
      (ENDING.PAGE.NUMBER CARDINAL)))
    (MEDIUM.HINT 5 MEDIUM)
    (PRIORITY.HINT 6 (ENUMERATION (HOLD 0)
      (LOW 1)
      (NORMAL 2)
      (HIGH 3)))
    (RELEASE.KEY 7 HASHED.PASSWORD)
    (STAPLE 8 BOOLEAN)
    (TWO.SIDED 9 BOOLEAN)
  [PRINTER.PROPERTIES (SEQUENCE (CHOICE (MEDIA 0 MEDIA)
    (STAPLE 1 BOOLEAN)
    (TWO.SIDED 2 BOOLEAN)
  [PRINTER.STATUS (SEQUENCE (CHOICE (SPOOLER 0 (ENUMERATION (Available 0)
    (Busy 1)
    (Disabled 2)
    (Full 3)))
    (FORMATTER 1 (ENUMERATION (Available 0)
      (Busy 1)
      (Disabled 2)))
    (PRINTER 2 (ENUMERATION (Available 0)
      (Busy 1)
      (Disabled 2)
      (NeedsAttention 3)
      (NeedKeyOperator 4)))
    (MEDIA 3 MEDIA)
  [REQUEST.STATUS (SEQUENCE (CHOICE (STATUS 0 (ENUMERATION (Pending 0)
    (InProgress 1)
    (Completed 2)
    (Unknown 3)
    (Rejected 4)
```

```

                                (Aborted 5)
                                (Cancelled 6)
                                (Held 7)))
                                (STATUS.MESSAGE 1 STRING]
(MEDIA (SEQUENCE MEDIUM))
(MEDIUM (CHOICE (PAPER 0 PAPER)))
[PAPER (CHOICE (UNKNOWN 0 NIL)
              (KNOWN.SIZE 1 (ENUMERATION ("US.LETTER" 1)
                                           ("US.LEGAL" 2)
                                           ("A0" 3)
                                           ("A1" 4)
                                           ("A2" 5)
                                           ("A3" 6)
                                           ("A4" 7)
                                           ("A5" 8)
                                           ("A6" 9)
                                           ("A7" 10)
                                           ("A8" 11)
                                           ("A9" 12)
                                           ("A10" 35)
                                           ("ISO.B0" 13)
                                           ("ISO.B1" 14)
                                           ("ISO.B2" 15)
                                           ("ISO.B3" 16)
                                           ("ISO.B4" 17)
                                           ("ISO.B5" 18)
                                           ("ISO.B6" 19)
                                           ("ISO.B7" 20)
                                           ("ISO.B8" 21)
                                           ("ISO.B9" 22)
                                           ("ISO.B10" 23)
                                           ("JIS.B0" 24)
                                           ("JIS.B1" 25)
                                           ("JIS.B2" 26)
                                           ("JIS.B3" 27)
                                           ("JIS.B4" 28)
                                           ("JIS.B5" 29)
                                           ("JIS.B6" 30)
                                           ("JIS.B7" 31)
                                           ("JIS.B8" 32)
                                           ("JIS.B9" 33)
                                           ("JIS.B10" 34))))
              (OTHER.SIZE 2 (RECORD (WIDTH CARDINAL)
                                     (LENGTH CARDINAL]
(CONNECTION.PROBLEM (ENUMERATION (NoRoute 0)
                                 (NoResponse 1)
                                 (TransmissionHardware 2)
                                 (TransportTimeout 3)
                                 (TooManyLocalConnections 4)
                                 (TooManyRemoteConnections 5)
                                 (MissingCourier 6)
                                 (MissingProgram 7)
                                 (MissingProcedure 8)
                                 (ProtocolMismatch 9)
                                 (ParameterInconsistency 10)
                                 (InvalidMessage 11)
                                 (ReturnTimedOut 12)
                                 (Other 65535)))
(TRANSFER.PROBLEM (ENUMERATION (Aborted 0)
                                (ChecksumIncorrect 1)
                                (FormatIncorrect 2)
                                (NoRendezvous 3)
                                (WrongDirection 4]

```

PROCEDURES

```

((PRINT 0 (BULK.DATA.SOURCE PRINT.ATTRIBUTES PRINT.OPTIONS)
  RETURNS
  (REQUEST.ID)
  REPORTS
  (BUSY CONNECTION.ERROR INSUFFICIENT.SPOOL.SPACE INVALID.PRINT.PARAMETERS MASTER.TOO.LARGE
   MEDIUM.UNAVAILABLE SERVICE.UNAVAILABLE SPOOLING.DISABLED SPOOLING.QUEUE.FULL SYSTEM.ERROR
   TOO.MANY.CLIENTS TRANSFER.ERROR UNDEFINED.ERROR))
(GET.PRINTER.PROPERTIES 1 NIL RETURNS (PRINTER.PROPERTIES)
  REPORTS
  (SERVICE.UNAVAILABLE SYSTEM.ERROR UNDEFINED.ERROR))
(GET.PRINT.REQUEST.STATUS 2 (REQUEST.ID)
  RETURNS
  (REQUEST.STATUS)
  REPORTS
  (SERVICE.UNAVAILABLE SYSTEM.ERROR UNDEFINED.ERROR))
(GET.PRINTER.STATUS 3 NIL RETURNS (PRINTER.STATUS)
  REPORTS
  (SERVICE.UNAVAILABLE SYSTEM.ERROR UNDEFINED.ERROR)))

```

ERRORS

```

((BUSY 0)
 (INSUFFICIENT.SPOOL.SPACE 1)
 (INVALID.PRINT.PARAMETERS 2)
 (MASTER.TOO.LARGE 3)

```



```

(CLOSEF INSTREAM))
' INPUT NIL NIL
' ((SEQUENTIAL T])
(COPYBYTES INSTREAM DATASTREAM)
NIL]
(if PRINTRESULTS
  then [COND
    ([AND NSPRINT.WATCHERFLG (NOT (LISTGET OPTIONS 'NO.WATCHER)
      (NSPRINT.WATCH.JOB PRINTRESULTS PRINTER (OR JOBNAME (UNPACKFILENAME.STRING
        DOCUMENT.NAME
        ' NAME)
        DOCUMENT.NAME])
      DOCUMENT.NAME))
    ;; Be sure to close stream on the way out
    (AND INSTREAM (CLOSEF? INSTREAM)))]])

```

(NSPRINT.INTERNAL

[LAMBDA (PRINTER OPTIONS TRANSFERFN)

; Edited 9-Aug-89 14:54 by bvm

;;; Calls the PRINT program for PRINTER, interpreting OPTIONS as a plist of print options. TRANSFERFN is a function applied to the transfer stream to
 ;;; actually send the Interpress master

```

(LET (COURIERSTREAM)
  (CL:UNWIND-PROTECT
    (PROG* ((MEDIUM (OR (LISTGET OPTIONS 'MEDIUM)
      NSPRINT.DEFAULT.MEDIUM))
      (STAPLE? (LISTGET OPTIONS 'STAPLE?))
      (TWO.SIDED? (EQ 2 (OR (LISTGET OPTIONS '%#SIDES)
        EMPRESS#SIDES)))
      (SENDER.NAME (OR (LISTGET OPTIONS 'SENDER.NAME)
        (USERNAME NIL NIL T)))
      (DOCNAME (OR (LISTGET OPTIONS 'DOCUMENT.NAME)
        "Document"))
      [ATTRIBUTES `( (PRINT.OBJECT.NAME ,DOCNAME)
        [PRINT.OBJECT.CREATE.DATE ,(OR (LISTGET OPTIONS 'DOCUMENT.CREATION.DATE)
          (IDATE)
          (SENDER.NAME ,SENDER.NAME)]
      [PRINTOPTIONS `( (COPY.COUNT ,(FIX (OR (LISTGET OPTIONS '%#COPIES)
        1]
      PROPERTIES VALUE STATUS LASTSTATUS) ; #copies 'option' seems to be required
    [COND
      ((SETQ VALUE (LISTGET OPTIONS 'RECIPIENT.NAME))
        (push PRINTOPTIONS (LIST 'RECIPIENT.NAME (OR (STRINGP VALUE)
          (MKSTRING VALUE))
      [COND
        ((SETQ VALUE (LISTGET OPTIONS 'PRIORITY))
          (push PRINTOPTIONS (LIST 'PRIORITY.HINT (SELECTQ VALUE
            ((HOLD LOW NORMAL HIGH)
            VALUE)
            (\ILLEGAL.ARG VALUE)
      [COND
        ((SETQ VALUE (LISTGET OPTIONS 'MESSAGE))
          (push PRINTOPTIONS (LIST 'MESSAGE (OR (STRINGP VALUE)
            (MKSTRING VALUE))
      [COND
        ((SETQ VALUE (LISTGET OPTIONS 'PAGES.TO.PRINT))
          ; A page range to print, (first# last#)
          (COND
            ((AND (LISTP VALUE)
              (LISTP (CDR VALUE))
              (NULL (CDDR VALUE))
              (SMALLPOSP (CAR VALUE))
              (SMALLPOSP (CADR VALUE)))
              (push PRINTOPTIONS (LIST 'PAGES.TO.PRINT VALUE)))
            (T (\ILLEGAL.ARG VALUE)
      RETRY
      (COND
        ((NOT (SETQ COURIERSTREAM (NSPRINT.COURIER.OPEN PRINTER)))
          (printout PROMPTWINDOW .TAB0 0 "No response from printer " (fetch NSPRINT.INTERNAME
            of PRINTER))
          (DISMISS 5000)
          (GO RETRY)))
          ; Check the status of the printer. No point sending to busy
          ; spooler, only to get a "too many clients" reject.
      (bind SS PS do (SETQ STATUS (COURIER.CALL COURIERSTREAM 'PRINTING 'GET.PRINTER.STATUS
        'RETURNERRORS))
      [COND
        ((EQ (CAR STATUS)
          'ERROR)
          (COND
            ((NOT (EQUAL STATUS LASTSTATUS))
              (printout PROMPTWINDOW T "[From " (fetch NSPRINT.INTERNAME of PRINTER)
                " Error: "
                (SUBSTRING (CDR STATUS)
                  2 -2)
                "; will retry")

```

```

        (SETQ LASTSTATUS STATUS)))
                                ; Wait longer for this problem
        (DISMISS 30000))
        ((NEQ (SETQ SS (CADR (ASSOC 'SPOOLER STATUS)))
              LASTSTATUS)
         (SELECTQ SS
          (Available                ; All is hunky dory
           (RETURN))
          (Busy (CL:FORMAT PROMPTWINDOW "~%[Spooler on ~A is busy~@[ (printer
~A~)]; will retry]" (fetch NSPRINTERNAME of PRINTER)
              (SELECTQ (SETQ PS (CADR (ASSOC 'PRINTER STATUS)))
               ((Available Busy)
                ; Printer status is uninteresting
                NIL)
               (STRING PS)))
          (SETQ LASTSTATUS SS))
         (PROGN
          ; Full or Disabled--these are not transient errors, so may want to
          ; interact with user
          (SETQ STATUS (CONS NIL (ASSOC 'SPOOLER STATUS)))
          (GO HANDLE.ERROR]
        (DISMISS 5000))
[COND
  ((OR MEDIUM STAPLE? TWO.SIDED?) ; Check that the printer supports these options.
   (SETQ PROPERTIES (COURIER.CALL COURIERSTREAM 'PRINTING 'GET.PRINTER.PROPERTIES
'RETURNERRORS))
   (COND
    ((EQ (CAR PROPERTIES)
         'ERROR)
     (SETQ STATUS PROPERTIES)
     (GO HANDLE.ERROR)))
   (COND
    (MEDIUM [COND
              ((SETQ VALUE (\NSPRINT.MEDIUM.CHECK MEDIUM (CADR (ASSOC 'MEDIA PROPERTIES)
                                                                    )
                                                                    PRINTER))
               (push PRINTOPTIONS (LIST 'MEDIUM.HINT VALUE]
              (SETQ MEDIUM)))
    (COND
     (STAPLE? (COND
                ((CADR (ASSOC 'STAPLE PROPERTIES))
                 (push PRINTOPTIONS (LIST 'STAPLE T)))
                (T (\NSPRINT.UNSUPPORTED PRINTER "stapled copies")))
              (SETQ STAPLE?)))
    (COND
     (TWO.SIDED? (COND
                  ((CADR (ASSOC 'TWO.SIDED PROPERTIES))
                   (push PRINTOPTIONS (LIST 'TWO.SIDED T)))
                  ((LISTGET OPTIONS '#SIDES)
                   ; Only warn if user explicitly asked for 2-sided
                   (\NSPRINT.UNSUPPORTED PRINTER "two-sided copies")))
                (SETQ TWO.SIDED?])
    ;; Finally, send the print document
    (SETQ STATUS (COURIER.CALL COURIERSTREAM 'PRINTING 'PRINT TRANSFERFN ATTRIBUTES PRINTOPTIONS
'RETURNERRORS))
    (COND
     ((NEQ (CAR STATUS)
          'ERROR)
      (CLOSEF COURIERSTREAM)
      (SETQ COURIERSTREAM NIL)
      (RETURN STATUS)))
    HANDLE.ERROR
    ;; Come here with STATUS = a courier error form.
    (CLOSEF COURIERSTREAM)
    (SELECTQ (CADR STATUS)
     ((TOO.MANY.CLIENTS TRANSFER.ERROR CONNECTION.ERROR)
      ; Transient errors, quietly try again
      (if (NEQ LASTSTATUS (SETQ LASTSTATUS 'Busy))
       then (printout PROMPTWINDOW T "[From " (fetch NSPRINTERNAME of PRINTER)
" -- "
              (CL:STRING-CAPITALIZE (CADR STATUS))
              "; will retry]"))
      (DISMISS 10000))
     (CL:CERROR "Try to send the document again" "Unexpected error from ~A while attempting
to print ~A -- ~A" (fetch NSPRINTERNAME of PRINTER)
              DOCNAME
              (CDR STATUS)))
    (GO RETRY))
    (if COURIERSTREAM
     then
     ; Abort the stream
     (SPP.CLOSE COURIERSTREAM T))])

```

(\NSPRINT.MEDIUM.CHECK
[LAMBDA (MEDIUM MEDIA PRINTER)

; Edited 11-Dec-87 14:31 by bvm:

```
(if (EQ MEDIUM T)
  then (CAR MEDIA)
  else (for X in MEDIA when [OR (EQUAL X MEDIUM)
                              (AND (EQ (CAR X)
                                       'PAPER)
                                   (STRPOS MEDIUM (CADR (CADR X))
                                               NIL NIL NIL NIL (UPPERCASEARRAY)
                                               (RETURN X) finally (\NSPRINT.UNSUPPORTED PRINTER "print medium" MEDIUM)
                                               (RETURN (CAR MEDIA])
```

(\NSPRINT.UNSUPPORTED

```
[LAMBDA (PRINTER FEATURE VALUE) ; Edited 11-Dec-87 14:27 by bvm:
;; Mention that this printer does not support some feature, with optional value.
(CL:FORMAT PROMPTWINDOW "~%[Printer ~A does not support ~A~@[[: ~A~]]]" (fetch NSPRINTERNAME of PRINTER)
  FEATURE VALUE])
```

(\NSPRINTER.HOSTNAMEP

```
[LAMBDA (PRINTERNAME) (* bvm%: "16-Sep-85 22:49")
;; True if PRINTERNAME names an NS printer. Do stupid test for now. Later on might want to test that random NS name really is a printer
(AND (STRPOS ":" PRINTERNAME)
      'INTERPRESS])
```

(\NSPRINTER.STATUS

```
[LAMBDA (PRINTER) (* bvm%: "29-Jun-84 17:02")
(\NSPRINT.ENQUIRE PRINTER 'GET.PRINTER.STATUS])
```

(\NSPRINTER.PROPERTIES

```
[LAMBDA (PRINTER) (* bvm%: "29-Jun-84 17:02")
(\NSPRINT.ENQUIRE PRINTER 'GET.PRINTER.PROPERTIES])
```

(\NSPRINTREQUEST.STATUS

```
[LAMBDA (REQUESTID PRINTER) (* bvm%: "29-Jun-84 16:38")
(\NSPRINT.ENQUIRE PRINTER (LIST 'GET.PRINT.REQUEST.STATUS REQUESTID])
```

(\NSPRINT.ENQUIRE

```
[LAMBDA (PRINTER OP) (* bvm%: "20-Jul-84 17:56")
```

;;; Perform a printing Courier op to PRINTER. OP is (FN . ARGS) to perform a COURIER.CALL on

```
(SETQ PRINTER (GETNSPRINTER PRINTER))
(PROG ((STREAM (\NSPRINT.COURIER.OPEN PRINTER)))
  (RETURN (COND
            (STREAM (RESETLST
                    (RESETSAVE NIL (LIST (FUNCTION \SPP.RESETCLOSE)
                                         STREAM))
                    [APPLY (FUNCTION COURIER.CALL)
                          (CONS STREAM (CONS 'PRINTING (APPEND (OR (LISTP OP)
                                                                    (LIST OP))
                                                                    (LIST 'NOERROR))]))
```

(\NSPRINT.COURIER.OPEN

```
[LAMBDA (PRINTER) (* bvm%: "20-Jul-84 10:31")
(COURIER.OPEN (fetch NSPRINTERADDRESS of PRINTER)
  NIL T (PACK* (fetch NSOBJECT of (fetch NSPRINTERNAME of PRINTER))
             "#Printing"])
```

;; Printer watcher

(DEFINEQ

(\NSPRINT.WATCHDOG

```
[LAMBDA (ID PRINTER JOBNAME) ; Edited 11-Dec-87 14:38 by bvm:
;; Run the single process for a given printer.
;; *PRINT-JOBS-IN-PROGRESS* is a list of quads (JOB-ID PRINTER JOBNAME LASTSTATUS).
(BLOCK 15000)
(bind MSG STATUS do [for TAIL on *PRINT-JOBS-IN-PROGRESS*
  do (DESTRUCTURING-BIND (ID PRINTER JOBNAME GIVEUPCNT . LASTSTATUS)
    (CAR TAIL)
    (COND
      ((NOT (EQUAL (SETQ STATUS (\NSPRINT.FULL.REQUEST.STATUS ID PRINTER))
                  LASTSTATUS))
       (printout PROMPTWINDOW .TAB0 0)
       (COND
         (JOBNAME (printout PROMPTWINDOW JOBNAME " on "))))
```

```

(printout PROMPTWINDOW (fetch NSPRINTENAME of PRINTER)
  " -- "
  (OR (CAR STATUS)
    "No response"))
(COND
  ((SETQ MSG (CADR STATUS))
    (printout PROMPTWINDOW " (" MSG ")"))))
(COND
  ((SETQ MSG (CADDR STATUS))
    (printout PROMPTWINDOW " (" MSG ")"))))
(if (SELECTQ (CAR STATUS)
  ((Pending InProgress)
    (SETQ GIVEUPCNT 0)
    NIL)
  (NIL ; No response
    (> (add GIVEUPCNT 1)
      5))
  T)
  then ; Can stop watching. This awful construction is because
      ; DREMOVE can disrupt the iteration.
  (RPLACA TAIL NIL)
  else ; Note status for next time
  (RPLACD (RPLACA (CDDDR (CAR TAIL))
    GIVEUPCNT)
    STATUS)
  (if (NULL (SETQ *PRINT-JOBS-IN-PROGRESS* (DREMOVE NIL *PRINT-JOBS-IN-PROGRESS*)))
    then (RETURN))
  (BLOCK 30000])

```

(\NSPRINT.WATCH.JOB

[LAMBDA (PRINTRESULTS PRINTER JOBNAME) ; Edited 11-Dec-87 12:38 by bvm:

:: Set up a 'watchdog' process to keep the guy informed of the print job's status

```

(CL:PUSH (LIST PRINTRESULTS PRINTER JOBNAME 0)
  *PRINT-JOBS-IN-PROGRESS*)
(OR (FIND.PROCESS 'Printer% Watcher)
  (ADD.PROCESS (LIST (FUNCTION \NSPRINT.WATCHDOG)
    'NAME
    'Printer% Watcher
    'AFTEREXIT
    'DELETE]))

```

(\NSPRINT.FULL.REQUEST.STATUS

[LAMBDA (ID PRINTER) (* bvm%: "26-Jul-85 12:38")

:: Returns a triple (RequestStatus StatusMessage PrinterStatus), with the last two items being NIL when they are uninteresting

```

(SETQ PRINTER (GETNSPRINTER PRINTER))
(CAR (NLSETQ (RESETLST
  (LET ((STREAM (\NSPRINT.COURIER.OPEN PRINTER))
    RESULT STATUS)
    (COND
      ((AND STREAM [PROGN (RESETSAVE NIL (LIST (FUNCTION \SPP.RESETCLOSE)
        STREAM))
        (SETQ RESULT (COURIER.CALL STREAM 'PRINTING
          'GET.PRINT.REQUEST.STATUS ID 'NOERROR])
        (LIST (CADR (ASSOC 'STATUS RESULT))
          (AND (SETQ STATUS (CADR (ASSOC 'STATUS.MESSAGE RESULT)))
            (NOT (STREQVAL STATUS ""))
            STATUS)
          (SELECTQ [SETQ STATUS (CADR (ASSOC 'PRINTER (COURIER.CALL STREAM
            'PRINTING
            'GET.PRINTER.STATUS
            'NOERROR])
            ((NIL Busy Available)
              ; Expected status values
              NIL)
            STATUS]))))
  )

```

(DECLARE%: DOEVAL@COMPILE DONTCOPY

(GLOBALVARS *PRINT-JOBS-IN-PROGRESS*)

(RPAQ? *PRINT-JOBS-IN-PROGRESS*)

:: FAX

(DEFINEQ

(FAX.SEND.FILE

[LAMBDA (HOST FILE PRINTOPTIONS) (* bvm%: "17-Sep-85 15:52")

;;; Sends Interpress document FILE to a FAX server specified by HOST, which is of the form person@place. Simple front end to NSPRINT

```
(LET ((HOST&OPTIONS (\FAX.PARSE.NAME HOST))
      (NSPRINT (CAR HOST&OPTIONS)
               FILE
               (APPEND (CDR HOST&OPTIONS)
                       PRINTOPTIONS
                       (AND FAX.NO.WATCHER (LIST 'NO.WATCHER T))
```

(FAX.STATUS

```
[LAMBDA (HOST) (* bvm%: "16-Sep-85 23:29")
```

;;; Tests status of FAX server specified by HOST, which is of the form person@place. Simple front end to NSPRINTER.STATUS

```
(NSPRINTER.STATUS (CAR (\FAX.PARSE.NAME HOST]))
```

(FAX.PROPERTIES

```
[LAMBDA (HOST) (* bvm%: "16-Sep-85 23:33")
```

;;; Returns properties of FAX server specified by HOST, which is of the form person@place. Simple front end to NSPRINTER.PROPERTIES

```
(NSPRINTER.PROPERTIES (CAR (\FAX.PARSE.NAME HOST]))
```

(FAX.HOSTNAMEP

```
[LAMBDA (PRINTERNAME) (* bvm%: "16-Sep-85 22:51")
```

;;; True if PRINTERNAME is something that looks like a FAX spec, i.e., person@place, where place is a phone number or something registered as a fax address. Stupid for now

```
(AND (STRPOS "@" PRINTERNAME)
      'FAX])
```

(\FAX.PARSE.NAME

```
[LAMBDA (PLACE) (* bvm%: "17-Sep-85 15:58")
```

;;; Parse a Fax spec 'Person@Place' and return a dotted pair (FaxServer . PrintOptions)

```
(PROG (AT PERSON DESTINATION PHONE HOST MSG INFO)
  RETRY
  (SETQ AT (STRPOS "@" PLACE))
  [COND
    ([SETQ PERSON (AND (NEQ AT 1)
                      (SUBSTRING PLACE 1 (SUB1 AT)
                                (SETQ PERSON (LIST 'RECIPIENT.NAME PERSON]
                      (SETQ DESTINATION (SUBSTRING PLACE (ADD1 AT)))
    (COND
      ([for CH instring DESTINATION always (OR (DIGITCHARP CH)
                                                (EQ CH (CHARCODE -))
                                                (EQ CH (CHARCODE *))
                                                (EQ CH (CHARCODE %#])
                                               ; Looks like a phone number
      (SETQ PHONE DESTINATION))
      ((AND (SETQ INFO (CDR (ASSOC (MKATOM (U-CASE DESTINATION))
                                  FAXADDRESSES)))
            (SETQ PHONE (CAR INFO)))
      (SETQ HOST (CADR INFO)))
      (T (SETQ MSG (CONCAT "The FAX destination %" DESTINATION "%" is unknown.
                          Edit the list FAXADDRESSES"))
         (GO FAIL)))
    (COND
      ((AND (NULL HOST)
            (NULL (SETQ HOST DEFAULTFAXHOST)))
       (SETQ MSG "Don't know the name of your local FAX server.
                 Set the variable DEFAULTFAXHOST")
        (GO FAIL)))
      [RETURN (CONS HOST (CONS 'MESSAGE (CONS PHONE PERSON]
  FAIL
  (ERROR (CONCAT "Don't understand " PLACE " because:"
                (CONCAT MSG " appropriately, then say OK.
                Alternatively, RETURN %"name@CorrectPhoneOrDestination%"))
```

```
)
(RPAQ? DEFAULTFAXHOST )
(RPAQ? FAXADDRESSES )
(RPAQ? FAX.NO.WATCHER T)
```

```
(DECLARE%: DOEVAL@COMPILE DONTCOPY
(GLOBALVARS DEFAULTFAXHOST FAXADDRESSES FAX.NO.WATCHER)
)
```


(ADDTOVAR **PRINTERTYPES**
((FAX TELECOPIER)
(CANPRINT (INTERPRESS))
(HOSTNAMEP FAX.HOSTNAMEP)
(STATUS FAX.STATUS)
(PROPERTIES FAX.PROPERTIES)
(SEND FAX.SEND.FILE)
(BITMAPSCALE INTERPRESS.BITMAPSCALE)
(BITMAPFILE (INTERPRESSBITMAP FILE BITMAP SCALEFACTOR REGION ROTATION TITLE))))
(PUTPROPS **NSPRINT COPYRIGHT** ("Venue & Xerox Corporation" 1984 1985 1986 1987 1990 1992))

FUNCTION INDEX

FAX.HOSTNAMEP	8	NSPRINTER.PROPERTIES	6	\NSPRINT.INTERNAL	4
FAX.PROPERTIES	8	NSPRINTER.STATUS	6	\NSPRINT.MEDIUM.CHECK	5
FAX.SEND.FILE	7	NSPRINTREQUEST.STATUS	6	\NSPRINT.UNSUPPORTED	6
FAX.STATUS	8	\FAX.PARSE.NAME	8	\NSPRINT.WATCH.JOB	7
GETNSPRINTER	3	\NSPRINT.COURIER.OPEN	6	\NSPRINT.WATCHDOG	6
NSPRINT	3	\NSPRINT.ENQUIRE	6		
NSPRINTER.HOSTNAMEP	6	\NSPRINT.FULL.REQUEST.STATUS	7		

VARIABLE INDEX

PRINT-JOBS-IN-PROGRESS 7	FAX.NO.WATCHER	8	NS.DEFAULT.PRINTER	3	NSPRINT.WATCHERFLG	3
DEFAULTFAXHOST	FAXADDRESSES	8	NSPRINT.DEFAULT.MEDIUM ..	3	PRINTERTYPES	9

RECORD INDEX

NSPRINTER	3
-----------------	---

COURIERPROGRAM INDEX

PRINTING	1
----------------	---
