

File created: 12-Jun-90 16:23:19 {DSK}<usr>local>lde>lispcore>library>TCPFTP.;3

changes to: (VARS TCPFTPCOMS)

previous date: 20-Jun-89 19:47:44 {DSK}<usr>local>lde>lispcore>library>TCPFTP.;2

Read Table: INTERLISP

Package: INTERLISP

Format: XCCS

::  
:: Copyright (c) 1985, 1986, 1990, 1987, 1988, 1989, 1990 by Venue & Xerox Corporation. All rights reserved.

### (RPAQQ TCPFTPCOMS

[[COMS  
:: FNS from Larry's Interlisp-10 LISPUSERS package

```
(FNS ARPACMD FTPHELP CMDREADCODE CMDREAD DISCARDLINE GETLINE \TCPFTP.INPUT TELNET.EOL)
(INITVARS (\TCPFTP.ARPACMD.LOCK (CREATE.MONITORLOCK "ARPACMD Lock")))
(GLOBALVARS \TCPFTP.ARPACMD.LOCK)
(DECLARE%: DONTEVAL@LOAD DOEVAL@COMPILE DONTCOPY COMPILEVARS (ADDVARS (NLAMA)
                                                                    (NLAML)
                                                                    (LAMA FTPHELP)
```

(COMS  
:: FNS for the Interlisp-D streams facility

```
(FNS \TCPFTP.CONTROL.CLOSED \TCPFTP.GET.OSTYPE \TCPFTP.EVENTFN \TCPFTP.HOSTNAMEP
\GET.TCPFTP.CONNECTION \TCPFTP.OPEN.CONNECTION \TCPFTP.ASSURE.CLEANUP \TCPFTP.CLEANUP
\TCPFTP.RELEASE.CONNECTION \TCPFTP.LOGIN \TCPFTP.DELETEFILE \TCPFTP.DIRECTORYNAMEP
\TCPFTP.ENDOFSTREAMOP \TCPFTP.GENERATEFILES \TCPFTP.GENERATENEXTFILE \TCPFTP.GETFILENAME
\TCPFTP.GETFILEINFO \TCPFTP.SETFILEINFO \TCPFTP.RENAMEFILE \TCPFTP.CONNECT \TCPFTP.OPENFILE
\TCPFTP.CLOSE \TCPFTP.FLUSH \TCPFTP.INIT SET.TCP.EOL.CONVENTION)
(DECLARE%: EVAL@COMPILE DONTCOPY (EXPORT (RECORDS TCPDATASTREAM TCPFTPCON)))
(ADDVARS (TCPFTP.DEFAULT.FILETYPES (NIL . TEXT)
      (DFASL . BINARY)
      (dfasl . BINARY)
      (LCOM . BINARY)
      (lcom . BINARY)
      (DCOM . BINARY)
      (dcom . BINARY)
      (LISP . TEXT)
      (lisp . TEXT)
      (LSP . TEXT)
      (lsp . TEXT)
      (RST . BINARY)
      (rst . BINARY)
      (BIN . BINARY)
      (bin . BINARY)))
(INITVARS (TCP.DEFAULTFILETYPE 'BINARY)
      (TCP.USE.STANDARD.EOL T)
      (\TCPFTP.DEVICES)
      (\TCPFTP.CLEANUP.PROCESS))
(GLOBALVARS \TCPFTP.DEVICES \TCPFTP.CLEANUP.PROCESS TCP.DEFAULTFILETYPE TCP.USE.STANDARD.EOL))
```

(COMS  
:: Data connection handling

```
(FNS \TCP.BYE \TCPFTP.MAYBE.ABORT \TCPFTP.DATA.CLOSED \TCPFTP.OPEN.DATA.CONNECTION
\TCPFTP.PORT.STRING \TCPFTP.SPAWN.DATACONNECTION \TCPFTP.READ.UNTIL.EOF
\TCPFTP.TRANSFER.COMPLETE \TCPFTP.WAIT.FOR.DATACONNECTION \TCPFTP.DELETE.CONNECTION)
(INITVARS (\TCPFTP.DATACONNECTION.LOCK (CREATE.MONITORLOCK "TCPFTP Data Connection Lock"))
      (\TCPFTP.CONNECTION.LOCK (CREATE.MONITORLOCK "TCPFTP Connection Lock"))
      (\TCPFTP.IDLE.TIMEOUT (TIMES 10 60 1000)))
(GLOBALVARS \TCPFTP.DATACONNECTION.LOCK \TCPFTP.CONNECTION.LOCK \TCPFTP.IDLE.TIMEOUT))
(FILE (SYSLOAD)
      TCPNAMES TCP)
(P (\TCPFTP.INIT))
(VARS TCPFTP.DEFAULT.FILETYPES)
(DECLARE%: DONTEVAL@LOAD DOEVAL@COMPILE DONTCOPY COMPILEVARS (ADDVARS (NLAMA)
                                                                    (NLAML)
                                                                    (LAMA]))
```

:: FNS from Larry's Interlisp-10 LISPUSERS package

(DEFINEQ

### (ARPACMD

[LAMBDA (TCPFTP CON CMD ARG WANT DISCARD WANTARG)

(\* ejs%: "15-Nov-86 15:09")  
(\* Imm "16-OCT-78 02:57")

(DECLARE (GLOBALVARS \TCPFTP.ARPACMD.LOCK))

(WITH.MONITOR \TCPFTP.ARPACMD.LOCK

[LET ((INC (fetch (TCPFTP CON TCPIN) of TCPFTP CON))  
(OUTC (fetch (TCPFTP CON TCPOUT) of TCPFTP CON)))

(RESETLST

[RESETSAVE NIL `(COND

(RESETSTATE (AND (OPENP %, INC 'INPUT)  
(CLOSEF %, INC))

(AND (OPENP %, OUTC 'OUTPUT)

```

(CLOSEF %, OUTC]
(PROG NIL
[COND
  (CMD [COND
    (FTPDEBUGFLG (printout FTPDEBUGLOG CMD)
    (COND
      (ARG (printout FTPDEBUGLOG " " ARG]
    (PRIN3 CMD OUTC)
    (COND
      (ARG (PRIN3 " " OUTC)
      (PRIN3 ARG OUTC)))
    (TELNET.EOL OUTC)
    (FORCEOUTPUT OUTC) (* flush)
    (COND
      (FTPDEBUGFLG (TERPRI FTPDEBUGLOG]
  LP (COND
    (FTPDEBUGFLG (printout FTPDEBUGLOG "< ")))
    (SETQ CMD (\TCPFTP.INPUT INC))
    (COND
      (FTPDEBUGFLG (printout FTPDEBUGLOG CMD " ")))
    (COND
      ((EQMEMB CMD WANTARG)
      (AND (EQ (BIN INC)
      (CHARCODE -))
      (FTPHELP CMD))
      (RETURN CMD)))
    [COND
      ((EQ (BIN INC)
      (CHARCODE -))
      (do (DISCARDLINE INC) repeatuntil (EQ (\TCPFTP.INPUT INC)
      CMD]
    (COND
      ((EQMEMB CMD WANT)
      (DISCARDLINE INC)
      (RETURN CMD))
      ((EQMEMB CMD DISCARD)
      (DISCARDLINE INC)
      (GO LP)))
    (SELECTQ (AND (FIXP CMD)
      (IQUOTIENT CMD 100))
      ((2 3)
      (FTPHELP CMD))
      ((4 5)
      (ERROR (GETLINE INC T)))
      NIL)
    (DISCARDLINE INC)
    (GO LP))))])

```

(FTPHELP  
[LAMBDA (ARG) (\* ejs%: "29-Jan-85 17:02")  
(ERROR ARG " unrecognized response from remote FTP server"])

(CMDREADCODE  
[LAMBDA (IN) (\* lmm "31-MAY-78 00:45")  
(PACK\* (CMDREAD IN)  
(CMDREAD IN)  
(CMDREAD IN])

(CMDREAD  
[LAMBDA (IN) (\* ejs%: "12-Jan-85 14:28")  
([LAMBDA (CH)  
(COND  
(FTPDEBUGFLG (BOUT CH FTPDEBUGLOG))  
CH]  
(BIN IN])

(DISCARDLINE  
[LAMBDA (IN) (\* ejs%: "3-Feb-86 16:16")  
(\* lmm "31-MAY-78 00:45")  
(DECLARE (GLOBALVARS FTPDEBUGFLG FTPDEBUGLOG))  
(COND  
(FTPDEBUGFLG (\BACKFILEPTR IN)  
(bind CH until [FMEMB (SETQ CH (BIN IN))  
(CONSTANT (LIST (CHARCODE LF)  
(CHARCODE NULL])  
do (BOUT FTPDEBUGLOG CH) finally (TERPRI FTPDEBUGLOG))  
(T (until (FMEMB (BIN IN)  
(CONSTANT (LIST (CHARCODE LF)  
(CHARCODE NULL])

(GETLINE  
[LAMBDA (IN FLG) (\* ejs%: "12-Jan-85 14:40")

```
(* Imm "31-MAY-78 00:46")
(bind CH (STRING _ (ALLOCSTRING 80)) for POS from 1 while (NEQ (SETQ CH (BIN IN))
(CHARCODE LF))
do (COND
((LEQ POS 80)
(RPLCHARCODE STRING POS CH)))
finally (RETURN (SUBSTRING STRING 1 (SUB1 POS)))
```

(\TCPFTP.INPUT

; Edited 17-Nov-88 15:16 by cdl

```
[LAMBDA (STREAM)
(DECLARE (GLOBALVARS FTPDEBUGFLG FTPDEBUGLOG))
(LET (CCODE (RESULT 0))
(to 3 do (SETQ CCODE (BIN STREAM))
[if (AND (GEQ CCODE (CHARCODE 0))
(LEQ CCODE (CHARCODE 9)))
then (SETQ RESULT (PLUS (TIMES RESULT 10)
(DIFFERENCE CCODE (CHARCODE 0)
repeattuntil (OR (EQ CCODE (CHARCODE SPACE))
(EQ CCODE (CHARCODE -))
(EQ CCODE 0))
finally (if (EQ CCODE (CHARCODE -))
then (if FTPDEBUGFLG
then (printout FTPDEBUGLOG T "< " RESULT))
(DISCARDLINE STREAM)
(\TCPFTP.INPUT STREAM))
RESULT])
```

(\TELNET.EOL

(\* ejs%: " 5-Jan-85 18:44")

```
[LAMBDA (STREAM)
(BOUT STREAM (CHARCODE CR))
(BOUT STREAM (CHARCODE LF))
(FORCEOUTPUT STREAM)]
```

)

(RPAQ? \TCPFTP.ARPACMD.LOCK (CREATE.MONITORLOCK "ARPACMD Lock"))

(DECLARE%: DOEVAL@COMPILE DONTCOPY

(GLOBALVARS \TCPFTP.ARPACMD.LOCK)

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

(ADDTOVAR NLAMA )

(ADDTOVAR NLAML )

(ADDTOVAR LAMA FTPHELP)

:: FNS for the Interlisp-D streams facility

(DEFINEQ

(\TCPFTP.CONTROL.CLOSED

(\* ejs%: "28-Jul-86 14:30")

```
[LAMBDA (INSTREAM OUTSTREAM)
(LET* [(DEVICE (fetch (STREAM DEVICE) of INSTREAM))
(TCPFTPCON (for CONN in (fetch (FDEV DEVICEINFO) of DEVICE) thereis (EQ (fetch (TCPFTPCON TCPIN)
of CONN)
INSTREAM]
(COND
(TCPFTPCON (replace (STREAM ACCESS) of INSTREAM with (replace (STREAM ACCESS) of OUTSTREAM with NIL))
(replace (FDEV DEVICEINFO) of DEVICE with (DREMOVE TCPFTPCON (fetch (FDEV DEVICEINFO)
of DEVICE]))
```

(\TCPFTP.GET.OSTYPE

; Edited 12-May-89 14:10 by welch

```
[LAMBDA (DEVICE)
(LET* ((HOST (fetch (FDEV DEVICENAME) of DEVICE))
ENTRY)
(GETHOSTINFO HOST 'OSTYPE])
```

(\TCPFTP.EVENTFN

(\* ejs%: "23-Apr-85 18:56")

[LAMBDA (FDEV FLG)

(\* \* Called when a major event happens)

```
(SELECTQ FLG
((BEFORELOGOUT BEFORESYSOUT BEFOREMAKESYS)
[bind TCPIN TCPOUT DATASTREAM for TCPFTPCON in (fetch (FDEV DEVICEINFO) of FDEV)
do (SETQ TCPIN (fetch (TCPFTPCON TCPIN) of TCPFTPCON))
(SETQ TCPOUT (fetch (TCPFTPCON TCPOUT) of TCPFTPCON))
(SETQ DATASTREAM (fetch (TCPFTPCON DATASTREAM) of TCPFTPCON))
```



```

                                (READP (fetch (TCPFTPCON TCPIN)
                                                of CON))
                                (\TCPFTP.INPUT (fetch (TCPFTPCON TCPIN)
                                                        of CON]
                                (ARPACMD CON "QUIT" NIL '(221 500))
                                (\TCPFTP.DELETE.CONNECTION CON DEV]
                                TCPFTPCON DEVICE))
[COND
  (INSTREAM [replace (STREAM ENDOFSTREAMOP) of INSTREAM with (FUNCTION (LAMBDA (STREAM)
                                                                    (ZERO]
    (replace (STREAM DEVICE) of INSTREAM with DEVICE)
    (replace (STREAM DEVICE) of OUTSTREAM with DEVICE)
    (replace (TCPFTPCON TCPIN) of TCPFTPCON with INSTREAM)
    (replace (TCPFTPCON TCPOUT) of TCPFTPCON with OUTSTREAM)
    (SELECTQ (\TCPFTP.INPUT INSTREAM)
      (220 [COND
        (FTPDEBUGFLG (printout FTPDEBUGLOG T "< 220 ")
          (DISCARDLINE (fetch (TCPFTPCON TCPIN) of TCPFTPCON]
          (\TCPFTP.LOGIN DEVICE TCPFTPCON)
          (push (fetch (FDEV DEVICEINFO) of DEVICE)
              TCPFTPCON)
          TCPFTPCON)
          (PROGN (\TCPFTP.DELETE.CONNECTION TCPFTPCON DEVICE)
              NIL]))

```

(\TCPFTP.ASSURE.CLEANUP

[LAMBDA NIL (\* ejs%: "27-Apr-85 14:08")

(\* \* Spawn a cleanup function if necessary)

```

[COND
  [(AND (PROCESSP \TCPFTP.CLEANUP.PROCESS)
        (NOT (PROCESS.FINISHEDP \TCPFTP.CLEANUP.PROCESS]
    (T (SETQ \TCPFTP.CLEANUP.PROCESS (ADD.PROCESS '(\TCPFTP.CLEANUP)
        'RESTARTABLE
        'NO])

```

(\TCPFTP.CLEANUP

[LAMBDA NIL (\* ejs%: "28-Jul-86 12:26")

```

(DECLARE (GLOBALVARS \TCPFTP.IDLE.TIMEOUT \TCPFTP.DEVICES \TCPFTP.CONNECTION.LOCK))
(LET
  ((INTERVAL (QUOTIENT \TCPFTP.IDLE.TIMEOUT 4))
   CONNECTIONSP)
  (repeatwhile (NOT (ZEROP CONNECTIONSP))
    do (SETQ CONNECTIONSP 0)
      (for DEVICE in \TCPFTP.DEVICES
        do (for CONNECTION in (APPEND (fetch (FDEV DEVICEINFO) of DEVICE))
          do (add CONNECTIONSP 1)
            (WITH.MONITOR \TCPFTP.CONNECTION.LOCK
              [NLSETQ (COND
                ((AND (NULL (fetch (TCPFTPCON BUSY?) of CONNECTION))
                     (TIMEREXPIRED? (fetch (TCPFTPCON IDLETIMER) of CONNECTION)))
                 (CLOSEF? (fetch (TCPFTPCON TCPIN) of CONNECTION))
                 (CLOSEF? (fetch (TCPFTPCON TCPOUT) of CONNECTION))
                 [COND
                   ((fetch (TCPFTPCON DATASTREAM) of CONNECTION)
                    (CLOSEF? (fetch (TCPFTPCON DATASTREAM) of CONNECTION]
                    (add CONNECTIONSP -1)
                    (\TCPFTP.DELETE.CONNECTION CONNECTION DEVICE T))
                   ([OR (NOT (OPENP (fetch (TCPFTPCON TCPIN) of CONNECTION)
                                   'INPUT))
                      (NEQ 'ESTABLISHED (fetch (TCP.CONTROL.BLOCK TCB.STATE)
                                                of (fetch (TCPSTREAM TCB)
                                                            of (fetch (TCPFTPCON TCPIN) of CONNECTION]
                    (add CONNECTIONSP -1)
                    (\TCPFTP.DELETE.CONNECTION CONNECTION DEVICE])
                (BLOCK)))
              (COND
                ((NOT (ZEROP CONNECTIONSP))
                 (BLOCK INTERVAL]))

```

(\TCPFTP.RELEASE.CONNECTION

[LAMBDA (TCPFTPCON) (\* jmh "11-Oct-85 13:43")

```

[COND
  (TCPFTPCON (replace (TCPFTPCON BUSY?) of TCPFTPCON with NIL)
    (replace (TCPFTPCON DATASTREAM) of TCPFTPCON with NIL)
    (replace (TCPFTPCON IDLETIMER) of TCPFTPCON with (SETUPTIMER \TCPFTP.IDLE.TIMEOUT))
    (\TCPFTP.ASSURE.CLEANUP])

```

(\TCPFTP.LOGIN

[LAMBDA (DEVICE TCPFTPCON)

; Edited 24-Apr-87 16:17 by FS

(\* \* Log us in)

```
(PROG ((OSTYPE (\TCPFTP.GET.OSTYPE DEVICE))
(HOST (fetch (FDEV DEVICENAME) of DEVICE))
(LOGINRETRYCOUNT 0)
INFO)
RETRY
(SETQ INFO (\INTERNAL/GETPASSWORD HOST))

(* * Loop through this label if the server rejected the our name)

(COND
 ([OR (NULL INFO)
 (EQ 0 (NCHARS (CAR INFO)))
 (EQ 0 (NCHARS (CDR INFO)

(* Need to login. Can't send Unix hosts a string of no chars as name or password!)

(LOGIN HOST)
(GO RETRY)))
RETRY1

(* * Loop through this label if the server rejected something else)

(SELECTQ (ARPACMD TCPFTP CON "USER" (COND
 ((AND (EQ OSTYPE 'UNIX)
 (EQ (CAR INFO)
 (U-CASE (CAR INFO)))
 (EQ LOGINRETRYCOUNT 0))
 (L-CASE (CAR INFO)))
 (T (CAR INFO)))
 '(202 230 331 332 500 503 530))
((230 202) (* We're logged in)
 (RETURN T))
(331 (* Needs a password)
 (SELECTQ (ARPACMD TCPFTP CON "PASS" (\DECRYPT.PWD (CDR INFO))
 '(230 331 332 530))
(230 (RETURN T))
(332 (SELECTQ (ARPACMD TCPFTP CON "ACCT" (PROMPTFORWARD (CONCAT "Account for logging
into " HOST))
 '(230 202 530))
(230 (RETURN T))
(GO RETRY1)))
((331 530)
 (LOGIN HOST)
 (add LOGINRETRYCOUNT 1)
 (GO RETRY))
(FTPHELP)))
(332 (SELECTQ (ARPACMD TCPFTP CON "ACCT" (PROMPTFORWARD (CONCAT "Account for logging into " HOST)
)
 '(230 202 530))
(230 (RETURN T))
(GO RETRY1)))
(503 [COND
 ((EQ OSTYPE 'UNIX)
 ;; Well, the sequence of events to get here was probably that the D-machine sent an illegal name/password pair, such
 ;; that the name was not a registered user on the Unix machine. There's a bug in the Unix FTP server which causes it to
 ;; send a 530 error--illegal user name--immediately after it sent a 331 to prompt us for the password. This is blatantly in
 ;; violation of the FTP specification, which states that only 100 class errors can have multiple responses. Now we're out
 ;; of sync with the server, and need somehow to reinitialize our state
 (\PEEKBIN (fetch (TCPFTP CON TCPIN) of TCPFTP CON))
 (\TCPFTP.INPUT (fetch (TCPFTP CON TCPIN) of TCPFTP CON)
 (GO RETRY1))
((500 530) (* No such user?)
 (LOGIN HOST)
 (add LOGINRETRYCOUNT 1)
 (GO RETRY))
(FTPHELP])
```

**(\TCPFTP.DELETEFILE**

[LAMBDA (NAME DEVICE)

(\* ejs%: " 7-Apr-86 11:52")

(\* \* FTP delete request)

```
(LET* [(TCPFTP CON (\GET.TCPFTP.CONNECTION DEVICE))
(OSTYPE (\TCPFTP.GET.OSTYPE DEVICE))
(CODE (CAR (NLSETQ (ARPACMD TCPFTP CON "DELE" (REPACKFILENAME.STRING (PACKFILENAME.STRING
'HOST NIL 'BODY NAME)
OSTYPE)
'(200 226 250 450 550)
(\TCPFTP.RELEASE.CONNECTION TCPFTP CON)
(SELECTQ CODE
((250 226 200)
NAME)
NIL])
```

(\TCPFTP.DIRECTORYNAMEP

(\* ejs%: "27-Apr-85 14:04")

```
[LAMBDA (HOST/DIR DEVICE)
  (LET ((TCPFTPCON (\GET.TCPFTP.CONNECTION DEVICE)))
    (RESETLST
      (RESETSAVE NIL (LIST [FUNCTION (LAMBDA (TCPFTPCON)
        (\TCPFTP.RELEASE.CONNECTION TCPFTPCON)
          (COND
            (RESETSTATE (AND (OPENP (fetch (TCPFTPCON TCPIN) of TCPFTPCON))
              (CLOSEF (fetch (TCPFTPCON TCPIN) of TCPFTPCON))
            )
            (replace (TCPFTPCON TCPIN) of TCPFTPCON with NIL)
            (AND (OPENP (fetch (TCPFTPCON TCPOUT) of TCPFTPCON))
              (CLOSEF (fetch (TCPFTPCON TCPOUT) of TCPFTPCON)))
            (replace (TCPFTPCON TCPOUT) of TCPFTPCON with NIL]
          TCPFTPCON))
    (\TCPFTP.CONNECT DEVICE TCPFTPCON (FILENAMEFIELD HOST/DIR 'DIRECTORY))))])
```

(\TCPFTP.ENDOFSTREAMOP

(\* ejs%: "3-Feb-85 17:01")

```
[LAMBDA (STREAM SILENTLY)
  (\TCPFTP.TRANSFER.COMPLETE STREAM)
  (OR SILENTLY (\EOSERROR STREAM])
```

(\TCPFTP.GENERATEFILES

; Edited 12-May-89 14:00 by welch

```
[LAMBDA (DEVICE PATTERN DESIREDPROPS OPTIONS)
```

(\* \* FTP directory request)

```
(LET* ((TCPFTPCON (\GET.TCPFTP.CONNECTION DEVICE))
  (OSTYPE (\TCPFTP.GET.OSTYPE DEVICE))
  DATASTREAMEVENT DATASTREAM CODE)
  (SETQ DATASTREAMEVENT (\TCPFTP.SPAWN.DATAACONNECTION TCPFTPCON 'INPUT))
  (BLOCK)
  [SETQ CODE (CAR (NLSETQ (ARPACMD TCPFTPCON "NLST" [COND
    [(EQ OSTYPE 'UNIX)
      (COND
        ((AND (EQ (FILENAMEFIELD PATTERN
          'VERSION)
            '*))
          (EQ (FILENAMEFIELD PATTERN
            'EXTENSION)
            '*))
          (EQ (FILENAMEFIELD PATTERN
            'NAME)
            '*))
          (REPACKFILENAME.STRING (
            PACKFILENAME.STRING
            'HOST NIL
            'VERSION NIL
            'EXTENSION NIL
            'NAME "*"
            'BODY PATTERN)
          'UNIX))
        ((EQ (FILENAMEFIELD PATTERN 'VERSION)
            '*))
          (REPACKFILENAME.STRING (
            PACKFILENAME.STRING
            'HOST NIL
            'VERSION NIL
            'BODY PATTERN)
          'UNIX))
        (T (REPACKFILENAME.STRING
          (PACKFILENAME.STRING 'HOST NIL
            'BODY PATTERN)
          'UNIX]
    (T (COND
      ((EQ OSTYPE 'INTERLISP)
        (PACKFILENAME.STRING 'HOST NIL
          'BODY PATTERN))
      (T (REPACKFILENAME.STRING
        (PACKFILENAME.STRING 'HOST NIL
          'BODY PATTERN)
        OSTYPE]
  ]
```

150]

```
(SELECTQ CODE
  (150
```

(\* \* Here we go)

```
(COND
  ((SETQ DATASTREAM (\TCPFTP.WAIT.FOR.DATAACONNECTION DEVICE TCPFTPCON DATASTREAMEVENT
    'INPUT))
    (replace (TCPFTPCON GENERATEFILESDIRECTORY) of TCPFTPCON with (FILENAMEFIELD PATTERN
      'DIRECTORY))
    (create FILEGENOBJ
```

```

NEXTFILEFN _ (FUNCTION \TCPFTP.GENERATENEXTFILE)
FILEINFOFN _ (FUNCTION NIL)
GENFILESTATE _ TCPFTP)
(T (ERROR "Couldn't open data connection to remote TCPFTP server"))
(PROGN (DEL.PROCESS (CAR DATASTREAMEVENT))
(\TCPFTP.RELEASE.CONNECTION TCPFTP)
(\NULLFILEGENERATOR])

```

**(\TCPFTP.GENERATENEXTFILE**

```

[LAMBDA (TCPFTP CON NAMEONLY) ; Edited 8-Mar-89 22:54 by akw:
(PROG ((DATASTREAM (fetch (TCPFTP CON DATASTREAM) of TCPFTP CON))
[OSTYPE (\TCPFTP.GET.OSTYPE (fetch (STREAM DEVICE) of (fetch (TCPFTP CON TCPIN) of TCPFTP CON))
[FILENAMERDTBL (DEFERREDCONSTANT (PROG [(R (COPYREADTABLE 'ORIG)
(SETBRK NIL NIL R)
(SETSYNTAX '%% 'OTHER R)
(SETSEPR '(13 10 31)
NIL R)
(RETURN R)]
CODE NAME)
LOOP
(RETURN (COND
[[AND (OPENP DATASTREAM 'INPUT)
(NOT (EOPF DATASTREAM))
(SETQ NAME (CAR (NLSETQ (READ DATASTREAM FILENAMERDTBL)
(COND
((AND (OR (EQ OSTYPE 'TOPS-20)
(EQ OSTYPE 'TOPS20))
(STRPOS "? Not found" NAME NIL NIL NIL NIL UPPERCASEARRAY))
(NLSETQ (until (EOPF DATASTREAM) do (READ DATASTREAM FILENAMERDTBL)))
(AND (OPENP DATASTREAM)
(CLOSEF DATASTREAM))
(SELECTQ [SETQ CODE (ARPACMD TCPFTP CON NIL NIL '(226 250)
(250 226)
(AND (OPENP DATASTREAM)
(CLOSEF DATASTREAM))
(\TCPFTP.RELEASE.CONNECTION TCPFTP CON)
NIL)
(FTPHELP CODE))])
((AND (EQ OSTYPE 'UNIX)
(STREQUAL ":" (SUBSTRING NAME -1)))
(replace (TCPFTP CON GENERATEFILES DIRECTORY) of TCPFTP CON
with (REPACKFILENAME.STRING (SUBSTRING NAME 1 -2)
'INTERLISP))
(GO LOOP))
(NAMEONLY (REPACKFILENAME.STRING NAME 'INTERLISP))
(T (if (STRPOS "*" (fetch (TCPFTP CON GENERATEFILES DIRECTORY) of TCPFTP CON))
then (PACKFILENAME.STRING 'HOST (fetch (FDEV DEVICENAME)
of (fetch (STREAM DEVICE)
of (fetch (TCPFTP CON TCPIN) of TCPFTP CON))
)
'BODY
(REPACKFILENAME.STRING NAME 'INTERLISP))
else (PACKFILENAME.STRING 'HOST (fetch (FDEV DEVICENAME)
of (fetch (STREAM DEVICE)
of (fetch (TCPFTP CON TCPIN) of TCPFTP CON)))
'DIRECTORY
(fetch (TCPFTP CON GENERATEFILES DIRECTORY) of TCPFTP CON)
'BODY
(REPACKFILENAME.STRING NAME 'INTERLISP)]
(T (AND (OPENP DATASTREAM)
(CLOSEF DATASTREAM))
(SELECTQ [SETQ CODE (ARPACMD TCPFTP CON NIL NIL '(226 250)
(250 226)
(AND (OPENP DATASTREAM)
(CLOSEF DATASTREAM))
(\TCPFTP.RELEASE.CONNECTION TCPFTP CON)
NIL)
(FTPHELP CODE])

```

**(\TCPFTP.GETFILENAME**

```

[LAMBDA (NAME RECOG DEVICE) ; Edited 12-May-89 13:35 by welch
(* * FTP directory request)
(COND
((EQ RECOG 'NEW)
NAME)
(T (PROG ((TCPFTP CON (\GET.TCPFTP.CONNECTION DEVICE))
(OSTYPE (\TCPFTP.GET.OSTYPE DEVICE))
DATASTREAMEVENT DATASTREAM CODE GENERATOR ALLPOSSIBILITIES)
(SETQ DATASTREAMEVENT (\TCPFTP.SPAWN.DATA CONNECTION TCPFTP CON 'INPUT))
(BLOCK)
[SETQ CODE (CAR (NLSETQ (ARPACMD TCPFTP CON "NLST" (COND
(EQ OSTYPE 'INTERLISP)
(PACKFILENAME.STRING 'HOST NIL

```



```

'BODY NAME))
(T (REPACKFILENAME.STRING
(PACKFILENAME.STRING 'HOST NIL
'BODY NAME)
OSTYPE)))
150]
(RETURN (SELECTQ CODE
(150
(* * Here we go)
(COND
((AND (SETQ DATASTREAM (\TCPFTP.WAIT.FOR.DATACONNECTION DEVICE TCPFTPCON
DATASTREAMEVENT 'INPUT))
(SETQ GENERATOR (create FILEGENOBJ
NEXTFILEFN _ (FUNCTION \TCPFTP.GENERATENEXTFILE
)
FILEINFOFN _ (FUNCTION NIL))
GENFILESTATE _ TCPFTPCON)))
(replace (TCPFTPCON GENERATEFILES DIRECTORY) of TCPFTPCON
with (FILENAMEFIELD NAME 'DIRECTORY))
(SETQ ALLPOSSIBILITIES (bind FILE while (SETQ FILE (\GENERATENEXTFILE
GENERATOR))
collect FILE))
(MKATOM (CAR ALLPOSSIBILITIES)))
(T (ERROR "Couldn't open data connection to remote TCPFTP server."))))
(PROGN (DEL.PROCESS (CAR DATASTREAMEVENT))
(\TCPFTP.RELEASE.CONNECTION TCPFTPCON)
NIL])

```

```

(\TCPFTP.GETFILEINFO
[LAMBDA (STREAM ATTRIB DEVICE) (* ejs%: "20-Mar-86 21:01")
(COND
((type? STREAM STREAM)
(STREAMPROP STREAM ATTRIB))
(EQ ATTRIB 'EOL)
'CRLF])

```

```

(\TCPFTP.SETFILEINFO
[LAMBDA (STREAM ATTRIB VALUE DEVICE) (* ejs%: "9-Nov-85 14:20")
(STREAMPROP STREAM ATTRIB VALUE])

```

```

(\TCPFTP.RENAMEFILE
[LAMBDA (OLDDEVICE OLDNAME NEWDEVICE NEWNAME) ; Edited 15-Jun-88 13:41 by atm
(* * FTP delete request)

```

```

(COND
((NEQ OLDDEVICE NEWDEVICE)
(\GENERIC.RENAMEFILE OLDDEVICE OLDNAME NEWDEVICE NEWNAME))
(T (LET ((OSTYPE (\TCPFTP.GET.OSTYPE OLDDEVICE))
(TCPFTPCON (\GET.TCPFTP.CONNECTION OLDDEVICE))
CODE)
(RESETLST
(RESETSAVE NIL (LIST [FUNCTION (LAMBDA (CON)
(\TCPFTP.RELEASE.CONNECTION CON]
TCPFTPCON))
[PROG NIL
RETRY
[SETQ CODE (CAR (NLSETQ (ARPACMD TCPFTPCON "RNFR" (REPACKFILENAME.STRING
(PACKFILENAME.STRING 'HOST NIL
'BODY OLDNAME)
OSTYPE)
(350 450 550]
(SELECTQ CODE
(350 [SETQ CODE (CAR (NLSETQ (ARPACMD TCPFTPCON "RNT0" (REPACKFILENAME.STRING
(PACKFILENAME.STRING
'HOST NIL 'BODY NEWNAME)
OSTYPE)
(200 250 553]
(SELECTQ CODE
((200 250)
(RETURN NEWNAME))
NIL))
(PROGN (SETQ OLDNAME (LISPERROR "FILE NOT FOUND" OLDNAME T))
(GO RETRY]))))

```

```

(\TCPFTP.CONNECT
[LAMBDA (DEVICE TCPFTPCON DIRECTORY) (* ejs%: "24-Jun-85 17:10")
(LET [(DIRECTORYNAME (REPACKFILENAME.STRING (PACKFILENAME.STRING 'DIRECTORY DIRECTORY)
(\TCPFTP.GET.OSTYPE DEVICE)]
(COND
((NEQ 0 (NCHARS DIRECTORYNAME))

```

```
(SELECTQ (ARPACMD TCPFTP CON "CWD" DIRECTORYNAME ' (200 250 450 550))
  ((200 250)
   T)
  NIL))
(T
```

(\* The user specified no connect directory. We'll have to assume he or she meant his or her own login directory, whose name we can't even accurately guess. Thus, we leave it at this)

```
T])
```

(TCPFTP.OPENFILE

[LAMBDA (NAME ACCESS RECOG PARAMETERS DEVICE) ; Edited 22-Mar-89 22:31 by welch

```
(DECLARE (GLOBALVARS TCP.DEFAULTFILETYPE TCP.USE.STANDARD.EOL TCPFTP.EOL.CONVENTION
  TCPFTP.DEFAULT.FILETYPES))
```

```
(LET* ((HOST (fetch (FDEV DEVICENAME) of DEVICE))
  (OSTYPE (\TCPFTP.GET.OSTYPE DEVICE))
```

```
[FILENAME (COND
  ((EQ OSTYPE 'INTERLISP)
   (PACKFILENAME.STRING 'HOST NIL 'BODY NAME))
  (T (REPACKFILENAME.STRING (PACKFILENAME.STRING 'HOST NIL 'BODY NAME)
  OSTYPE]
```

```
(FILENAME.EXTENSION (FILENAMEFIELD FILENAME 'EXTENSION))
(TCPFTP CON (\GET.TCPFTP.CONNECTION DEVICE))
(TYPE (OR (CADR (FASSOC 'TYPE PARAMETERS))
  (CDR (FASSOC FILENAME.EXTENSION TCPFTP.DEFAULT.FILETYPES))
  (CDR (FASSOC (U-CASE FILENAME.EXTENSION)
  TCPFTP.DEFAULT.FILETYPES))
  TCP.DEFAULTFILETYPE))
```

```
DATASTREAMEVENT DATASTREAM CODE FTPCMD STREAMDEV)
```

```
(SELECTQ TYPE
```

```
(TEXT (ARPACMD TCPFTP CON "TYPE" "A N" 200))
(ARPACMD TCPFTP CON "TYPE" "L 8" 200))
```

```
(SETQ DATASTREAMEVENT (\TCPFTP.SPAWN.DATACONNECTION TCPFTP CON (COND
  ((EQ ACCESS 'OUTPUT)
   'APPEND)
  (T ACCESS))
  T))
```

```
(BLOCK)
```

```
(PROG NIL
```

```
  LOOP
```

```
(SETQ FTPCMD (SELECTQ ACCESS
  (INPUT ' "RETR")
  (OUTPUT ' "STOR")
  (APPEND ' "APPE")
  (ERROR "ACCESS must be one of INPUT, OUTPUT, or APPEND" ACCESS)))
```

```
[SETQ CODE (CAR (NLSETQ (ARPACMD TCPFTP CON FTPCMD FILENAME
  '(125 150 226 250 425 426 450 451 550)
```

```
(SELECTQ CODE
  ((125 150)
```

(\* \* Here we go)

```
(COND
  ([SETQ DATASTREAM (\TCPFTP.WAIT.FOR.DATACONNECTION DEVICE TCPFTP CON DATASTREAMEVENT
  (COND
```

```
((EQ ACCESS 'OUTPUT)
  'APPEND)
  (T ACCESS]
```

```
(replace (STREAM ENDOFSTREAMOP) of DATASTREAM with (FUNCTION \TCPFTP.ENDOFSTREAMOP))
```

```
(replace (STREAM FULLFILENAME) of DATASTREAM with NAME)
```

```
[replace (STREAM EOLCONVENTION) of DATASTREAM
```

```
  with (COND
```

```
(TCP.USE.STANDARD.EOL CRLF.EOLC)
  (T (OR TCPFTP.EOL.CONVENTION (SELECTQ OSTYPE
    (UNIX LF.EOLC)
    (TOPS-20 CRLF.EOLC)
    CR.EOLC]
```

```
(STREAMPROP DATASTREAM 'TYPE TYPE)
```

```
(replace (TCPDATASTREAM TCPFTP CON) of DATASTREAM with TCPFTP CON)
```

```
(SETQ STREAMDEV (fetch (STREAM DEVICE) of DATASTREAM))
```

```
(replace (FDEV GETFILENAME) of STREAMDEV with (FUNCTION NIL))
```

```
(replace (FDEV GETFILEINFO) of STREAMDEV with (FUNCTION \TCPFTP.GETFILEINFO))
```

```
(STREAMADDPROP DATASTREAM 'AFTERCLOSE (FUNCTION \TCPFTP.TRANSFER.COMPLETE))
```

```
(STREAMADDPROP DATASTREAM 'BEFORECLOSE (FUNCTION \TCPFTP.READ.UNTIL.EOF))
```

```
(RETURN DATASTREAM))
```

```
(T (ERROR "Couldn't open data connection to remote TCPFTP server"))))
```

```
(425 (* The foreign port is busy)
```

```
(PROMPTPRINT "TCPFTP: Please wait; the remote ftp server is busy.")
```

```
(DEL.PROCESS (CAR DATASTREAMEVENT))
```

```
(DISMISS 5000)
```

```
[SETQ DATASTREAMEVENT (\TCPFTP.SPAWN.DATACONNECTION TCPFTP CON (COND
```

```
((EQ ACCESS
  'OUTPUT)
  'APPEND)
  (T ACCESS]
```

```

        (BLOCK)
        (GO LOOP))
    ((450 550)
     (DEL.PROCESS (CAR DATASTREAMEVENT))
     (\TCPFTP.RELEASE.CONNECTION TCPFTPCON)
     NIL)
    (FTPHELP CODE])

```

**(\TCPFTP.CLOSE**

[LAMBDA (DEVICE)

(\* ejs%: "23-Apr-85 18:41")

(\* \* This needs work)

```

(PROG ((DEVINFO (fetch (FDEV DEVICEINFO) of DEVICE)))
 (AND (OPENP (fetch (TCPFTPCON TCPOUT) of DEVINFO)
        'OUTPUT)
       (CLOSEF (fetch (TCPFTPCON TCPOUT) of DEVINFO)))
 (AND (OPENP (fetch (TCPFTPCON TCPIN) of DEVINFO)
        'INPUT)
       (CLOSEF (fetch (TCPFTPCON TCPIN) of DEVINFO]))

```

**(\TCPFTP.FLUSH**

[LAMBDA (DEVICE)

(\* ejs%: "23-Apr-85 18:56")

(\* \* This needs work)

```

(PROG [(INSTREAM (fetch (TCPFTPCON TCPIN) of (fetch (FDEV DEVICEINFO) of DEVICE)]
 (COND
  ((READP INSTREAM)
   (until (NOT (READP INSTREAM)) do (BIN INSTREAM)]

```

**(\TCPFTP.INIT**

[LAMBDA NIL

(\* ejs%: "10-Apr-85 19:25")

```

(\DEFINEDEVICE NIL (create FDEV
  DEVICENAME _ 'TCPFTP
  HOSTNAMEP _ (FUNCTION \TCPFTP.HOSTNAMEP)
  EVENTFN _ (FUNCTION \TCPFTP.EVENTIFN])

```

**(SET.TCP.EOL.CONVENTION**

[LAMBDA (EOLTYPE)

; Edited 22-Mar-89 22:31 by welch  
; Sets the EOL convention to use

```

(DECLARE (GLOBALVARS TCP.USE.STANDARD.EOL TCPFTP.EOL.CONVENTION))
(SELECTQ EOLTYPE
 (CR (SETQ TCP.USE.STANDARD.EOL NIL)
      (SETQ TCPFTP.EOL.CONVENTION CR.EOLC))
 (LF (SETQ TCP.USE.STANDARD.EOL NIL)
      (SETQ TCPFTP.EOL.CONVENTION LF.EOLC))
 (CRLF (SETQ TCP.USE.STANDARD.EOL NIL)
        (SETQ TCPFTP.EOL.CONVENTION CRLF.EOLC))
 (OS (SETQ TCP.USE.STANDARD.EOL NIL)
      (SETQ TCPFTP.EOL.CONVENTION NIL))
 (SETQ TCP.USE.STANDARD.EOL T])

```

)

(DECLARE%: EVAL@COMPILE DONTCOPY

:: FOLLOWING DEFINITIONS EXPORTED

(DECLARE%: EVAL@COMPILE

```

[ACCESSFNS TCPDATASTREAM ((TCPCONTROLDEVICE (fetch (STREAM F3) of DATUM)
  (replace (STREAM F3) of DATUM with NEWVALUE))
 (SEENEOS (fetch (STREAM F4) of DATUM)
  (replace (STREAM F4) of DATUM with NEWVALUE))
 (TCPFTPCON (fetch (STREAM F5) of DATUM)
  (replace (STREAM F5) of DATUM with NEWVALUE]

```

```

(RECORD TCPFTPCON (TCPIN TCPOUT DATASTREAM BUSY? IDLETIMER GENERATEFILESDIRECTORY))
)
)

```

:: END EXPORTED DEFINITIONS

```

(ADDTOVAR TCPFTP.DEFAULT.FILETYPES (NIL . TEXT)
  (DFASL . BINARY)
  (dfasl . BINARY)
  (LCOM . BINARY)
  (lcom . BINARY)
  (DCOM . BINARY)
  (dcom . BINARY)
  (LISP . TEXT)

```

(lisp . TEXT)
(LSP . TEXT)
(lsp . TEXT)
(RST . BINARY)
(rst . BINARY)
(BIN . BINARY)
(bin . BINARY))

(RPAQ? TCP.DEFAULTFILETYPE 'BINARY)

(RPAQ? TCP.USE.STANDARD.EOL T)

(RPAQ? \TCPFTP.DEVICES )

(RPAQ? \TCPFTP.CLEANUP.PROCESS )

(DECLARE%: DOEVAL@COMPILE DONTCOPY

(GLOBALVARS \TCPFTP.DEVICES \TCPFTP.CLEANUP.PROCESS TCP.DEFAULTFILETYPE TCP.USE.STANDARD.EOL)
)

:: Data connection handling

(DEFINEQ

(\TCP.BYE

[LAMBDA (HOST) ; (\* ejs%: "15-Nov-86 15:05")
(LET\* [(DEVICE (\GETDEVICEFROMNAME HOST NIL T))
(CONNECTIONS (AND DEVICE (fetch (FDEV DEVICEINFO) of DEVICE)
(bind INSTREAM for TCPFTP CON in CONNECTIONS do (SETQ INSTREAM (fetch (TCPFTP CON TCPIN) of TCPFTP CON))
(while (AND (OPENP INSTREAM 'INPUT)
(READP INSTREAM))
do (BIN INSTREAM))
[NLSETQ (ARPACMD TCPFTP CON "QUIT" NIL
'(221 500)
(CLOSEF? (fetch (TCPFTP CON DATASTREAM) of TCPFTP CON))
(CLOSEF? INSTREAM)
(CLOSEF? (fetch (TCPFTP CON TCPOUT) of TCPFTP CON))
T)
(replace (FDEV DEVICEINFO) of DEVICE with NIL])

(\TCPFTP.MAYBE.ABORT

[LAMBDA (DATASTREAM) ; Edited 18-Mar-89 13:43 by welch
(LET\* ((TCPFTP CON (fetch (TCPDATASTREAM TCPFTP CON) of DATASTREAM))
(TCPOUTSTREAM (fetch (TCPFTP CON TCPOUT) of TCPFTP CON))
(STREAMPROP DATASTREAM 'BEFORECLOSE NIL)
(COND
((AND (NOT (fetch (TCPDATASTREAM SEENEOS) of DATASTREAM))
(OPENP DATASTREAM 'INPUT))
(TCP.CLOSE.SENDER (TCP.OTHER.STREAM DATASTREAM))
(BLOCK)
(BOUT TCPOUTSTREAM 244)
(BOUT TCPOUTSTREAM 242)
(TCP.URGENT.MARK TCPOUTSTREAM)
(ARPACMD TCPFTP CON "ABOR" NIL '(226 426 250))

(\TCPFTP.DATA.CLOSED

[LAMBDA (INSTREAM OUTSTREAM) ; (\* ejs%: "28-Jul-86 14:03")
(LET\* ((STREAM (OR INSTREAM OUTSTREAM))
(replace (STREAM ACCESS) of STREAM with NIL))

(\TCPFTP.OPEN.DATA.CONNECTION

[LAMBDA (TCPFTP CON ACCESS EVENT FOR.FILE.TRANSFER) ; (\* ejs%: "26-Sep-86 18:27")
(DECLARE (GLOBALVARS \TCPFTP.DATA.CONNECTION.LOCK))

(\* Tell the FTP control connection on what port we're expecting the data connection to made, and try up to five times to accept a connection. Each time, select a new port (this hopefully a workaround to a Unix bug in which ports sometimes tend to appear busy for 2 minute timeout intervals)

(WITH.MONITOR \TCPFTP.DATA.CONNECTION.LOCK
(bind PORT STREAM for I from 1 to 5
do (SETQ PORT (\TCP.SELECT.PORT))
(ARPACMD TCPFTP CON "PORT" (\TCPFTP.PORT.STRING PORT)
'(200))
[SETQ STREAM (TCP.OPEN NIL NIL PORT 'PASSIVE ACCESS NIL
(COND
(FOR.FILE.TRANSFER (CONSTANT '(MAXSEG %, BYTESPERPAGE WHENCLOSEDFN
\tCPFTP.DATA.CLOSED)
(COND
(STREAM (replace (TCPFTP CON DATASTREAM) of TCPFTP CON with STREAM)
(RETURN)))

finally

(\* We give up. Place a NIL in the datastream field so the client who was trying to accept the data connection will realize we

couldn't succeed)

```

      (replace (TCPFTP.CON DATASTREAM) of TCPFTP.CON with NIL))
(AND (TYPEOF EVENT 'EVENT)
      (NOTIFY.EVENT EVENT))
(fetch (TCPFTP.CON DATASTREAM) of TCPFTP.CON)))

```

(\TCPFTP.PORT.STRING

[LAMBDA (PORT) (\* ejs%: "26-Apr-85 11:54")

(\* Returns "h1,h2,h3,h4,p1,p3" corresponding to bytes of local IP host and PORT for port command)

```

(LET ((IPADDRESS (\LOCAL.IP.ADDRESS))
      (CONCAT (LOADBYTE IPADDRESS 24 8)
              ", "
              (LOADBYTE IPADDRESS 16 8)
              ", "
              (LOADBYTE IPADDRESS 8 8)
              ", "
              (LOADBYTE IPADDRESS 0 8)
              ", "
              (LOADBYTE PORT 8 8)
              ", "
              (LOADBYTE PORT 0 8]))

```

(\TCPFTP.SPAWN.DATA.CONNECTION

[LAMBDA (TCPFTP.CON ACCESS FOR.FILE.TRANSFER) (\* ejs%: "26-Sep-86 19:21")

(\* Called from TCPFTP device methods like \TCPFTP.OPENFILE.  
 Spawns a process to wait for the server program to open a data connection to us.  
 Returns a CONS consisting of the spawned process handle and an event which will be notified when the server has  
 connected to us. This function MUST be called prior to any TCPFTP operations which would cause the server to try to open  
 a data connection to us (otherwise, the server might try to open the connection before we're prepared to accept it))

```

(LET* [(EVENT (CREATE.EVENT))
       (PROCESS (ADD.PROCESS (\TCPFTP.OPEN.DATA.CONNECTION (QUOTE %, ACCESS)
                                                             (QUOTE %, ACCESS)
                                                             %, EVENT %, FOR.FILE.TRANSFER]
                             (RESETSAVE NIL (LIST [FUNCTION (LAMBDA (PROCESS INSTREAM OUTSTREAM)
                                                                (DEL.PROCESS PROCESS) (* CLOSEF? INSTREAM)
                                                                (* CLOSEF? OUTSTREAM)
                                                                NIL]
                                                  PROCESS
                                                  (fetch (TCPFTP.CON TCPIN) of TCPFTP.CON)
                                                  (fetch (TCPFTP.CON TCPOUT) of TCPFTP.CON))))
       (CONS PROCESS EVENT])

```

(\TCPFTP.READ.UNTIL.EOF

[LAMBDA (DATASTREAM) ; Edited 20-Jun-89 19:41 by welch

;; This function is used to avoid possible deadlock in the case where the stream is opened and closed immediately.

```

(PROG ((TCB (fetch (TCPSTREAM TCB) of DATASTREAM))
      (TCPFTP.CON (fetch (TCPDATASTREAM TCPFTP.CON) of DATASTREAM)))
      (WITH.MONITOR \TCPFTP.DATA.CONNECTION.LOCK
        (if (NOT (EOFP DATASTREAM))
            then (while (\QUEUEHEAD (fetch TCB.INPUT.QUEUE of TCB)) do (\TCP.GET.SEGMENT DATASTREAM))
                ;; read to the end of the file.
                (while (NOT (EOFP DATASTREAM)) do (BIN DATASTREAM))))))

```

(\TCPFTP.TRANSFER.COMPLETE

[LAMBDA (DATASTREAM) ; Edited 24-May-89 14:12 by welch

```

(LET ((TCPFTP.CON (fetch (TCPDATASTREAM TCPFTP.CON) of DATASTREAM))
      (STREAMPROP DATASTREAM 'AFTERCLOSE NIL)
      (COND
        ((AND TCPFTP.CON (NOT (fetch (TCPDATASTREAM SEENEOS) of DATASTREAM)))
         [COND
          ((OPEN DATASTREAM 'INPUT)
           (TCP.CLOSE.SENDER (TCP.OTHER.STREAM DATASTREAM)
                             (replace (TCPDATASTREAM SEENEOS) of DATASTREAM with T)
                             (replace (TCPDATASTREAM TCPCONTROLDEVICE) of DATASTREAM with NIL)
                             (replace (TCPDATASTREAM TCPFTP.CON) of DATASTREAM with NIL)
                             (\TCPFTP.RELEASE.CONNECTION TCPFTP.CON]))

```

(\TCPFTP.WAIT.FOR.DATA.CONNECTION

[LAMBDA (DEVICE TCPFTP.CON PROCESS.AND.EVENT ACCESS) (\* ejs%: "26-Sep-86 18:30")

(\* EVENT is a cons of PROCESS and a real event. PROCESS is the process trying to open the connection;  
 EVENT is an event which is notified when the process succeeds or fails to open the connection to the server)

```

(LET (STREAM)

```

```
(AWAIT.EVENT (CDR PROCESS.AND.EVENT)
 120000)
(COND
 ((NULL (SETQ STREAM (fetch (TCPFTPCON DATASTREAM) of TCPFTPCON)))
 (* A NIL in this field means the local client code was unable to open the connection to the server program.)
 NIL)
 ((OPENP STREAM ACCESS)
 (replace (TCPDATASTREAM TCPCONTROLDEVICE) of STREAM with DEVICE)
 STREAM])
```

(\TCPFTP.DELETE.CONNECTION

```
[LAMBDA (TCPFTPCON DEVICE SENDBY) (* ejs%:"15-Nov-86 15:09")
 (LET ((INSTREAM (fetch (TCPFTPCON TCPIN) of TCPFTPCON)))
 [COND
 (SENDBY (NLSETQ (ARPACMD TCPFTPCON "BYE" NIL ' (221 500)
 [COND
 (INSTREAM (DEL.PROCESS (fetch (TCP.CONTROL.BLOCK TCB.PROCESS) of (fetch (TCPSTREAM TCB) of INSTREAM]
 (replace (FDEV DEVICEINFO) of DEVICE with (DREMOVE TCPFTPCON (fetch (FDEV DEVICEINFO) of DEVICE]))
```

```
)
(RPAQ? \TCPFTP.DATACONNECTION.LOCK (CREATE.MONITORLOCK "TCPFTP Data Connection Lock"))
(RPAQ? \TCPFTP.CONNECTION.LOCK (CREATE.MONITORLOCK "TCPFTP Connection Lock"))
(RPAQ? \TCPFTP.IDLE.TIMEOUT (TIMES 10 60 1000))
(DECLARE%: DOEVAL@COMPILE DONTCOPY
(GLOBALVARS \TCPFTP.DATACONNECTION.LOCK \TCPFTP.CONNECTION.LOCK \TCPFTP.IDLE.TIMEOUT)
)
(FILESLoad (SYSLOAD)
 TCPNAMES TCP)
```

(\TCPFTP.INIT)

(RPAQQ TCPFTP.DEFAULT.FILETYPES

```
((NIL . TEXT)
 (DFASL . BINARY)
 (dfasl . BINARY)
 (LCOM . BINARY)
 (lcom . BINARY)
 (DCOM . BINARY)
 (dcom . BINARY)
 (LISP . TEXT)
 (lisp . TEXT)
 (LSP . TEXT)
 (lsp . TEXT)
 (RST . BINARY)
 (rst . BINARY)
 (BIN . BINARY)
 (bin . BINARY)
 (TXT . TEXT)
 (txt . TEXT)
 (TEXT . TEXT)
 (text . TEXT)
 (c . TEXT)
 (h . TEXT)
 (o . BINARY)
 (TEDIT . BINARY)
 (tedit . BINARY)
 (DISPLAYFONT . BINARY)
 (WD . BINARY)))
```

(DECLARE%: DONTEVAL@LOAD DOEVAL@COMPILE DONTCOPY COMPILERVARS

(ADDTovar NLAMA )

(ADDTovar NLAML )

(ADDTovar LAMA )

)

(PUTPROPS TCPFTP COPYRIGHT ("Venue & Xerox Corporation" 1985 1986 1900 1987 1988 1989 1990))

---

**FUNCTION INDEX**

ARPACMD .....	1	\TCPFTP.DATA.CLOSED .....	12	\TCPFTP.LOGIN .....	5
CMDREAD .....	2	\TCPFTP.DELETE.CONNECTION .....	14	\TCPFTP.MAYBE.ABORT .....	12
CMDREADCODE .....	2	\TCPFTP.DELETEFILE .....	6	\TCPFTP.OPEN.CONNECTION .....	4
DISCARDLINE .....	2	\TCPFTP.DIRECTORYNAMEP .....	7	\TCPFTP.OPEN.DATA.CONNECTION .....	12
FTPHELP .....	2	\TCPFTP.ENDOFSTREAMOP .....	7	\TCPFTP.OPENFILE .....	10
GETLINE .....	2	\TCPFTP.EVENTFN .....	3	\TCPFTP.PORT.STRING .....	13
SET.TCP.EOL.CONVENTION .....	11	\TCPFTP.FLUSH .....	11	\TCPFTP.READ.UNTIL.EOF .....	13
TELNET.EOL .....	3	\TCPFTP.GENERATEFILES .....	7	\TCPFTP.RELEASE.CONNECTION .....	5
\GET.TCPFTP.CONNECTION .....	4	\TCPFTP.GENERATENEXTFILE .....	8	\TCPFTP.RENAMEFILE .....	9
\TCP.BYE .....	12	\TCPFTP.GET.OSTYPE .....	3	\TCPFTP.SETFILEINFO .....	9
\TCPFTP.ASSURE.CLEANUP .....	5	\TCPFTP.GETFILEINFO .....	9	\TCPFTP.SPAWN.DATACONNECTION .....	13
\TCPFTP.CLEANUP .....	5	\TCPFTP.GETFILENAME .....	8	\TCPFTP.TRANSFER.COMPLETE .....	13
\TCPFTP.CLOSE .....	11	\TCPFTP.HOSTNAMEP .....	4	\TCPFTP.WAIT.FOR.DATACONNECTION .....	13
\TCPFTP.CONNECT .....	9	\TCPFTP.INIT .....	11		
\TCPFTP.CONTROL.CLOSED .....	3	\TCPFTP.INPUT .....	3		

---

**VARIABLE INDEX**

TCP.DEFAULTFILETYPE .....	12	\TCPFTP.ARPACMD.LOCK .....	3	\TCPFTP.DATACONNECTION.LOCK .....	14
TCP.USE.STANDARD.EOL .....	12	\TCPFTP.CLEANUP.PROCESS .....	12	\TCPFTP.DEVICES .....	12
TCPFTP.DEFAULT.FILETYPES .....	11, 14	\TCPFTP.CONNECTION.LOCK .....	14	\TCPFTP.IDLE.TIMEOUT .....	14

---

**RECORD INDEX**

TCPDATASTREAM .....	11	TCPFTPCON .....	11
---------------------	----	-----------------	----

---