

File created: 29-Sep-88 01:18:17 {PHYLUM}<BURWELL>LISP>LYRIC>PROGRAMCHAT.;1

changes to: (FNS OPENCHATSTREAM PROGRAMCHAT PROGRAMCHAT.LOGIN)

previous date: 12-May-88 21:46:25 {QV}<BURWELL>LISP>PROGRAMCHAT.;1

Read Table: INTERLISP

Package: INTERLISP

Format: XCCS

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

(RPAQQ PROGRAMCHATCOMS

```
[(FNS OPENCHATSTREAM PROGRAMCHAT PROGRAMCHAT.LOGIN PROGRAMCHAT.OUTPUT)
 (P [if (ASSOC 'TENEX NETWORKLOGINFO)
      then
        (PUTASSOC 'LOGOUT ' ("LOGOUT" CR)
                  (CDR (ASSOC 'TENEX NETWORKLOGINFO))
        [if (ASSOC 'TOPS20 NETWORKLOGINFO)
            then
              (PUTASSOC 'LOGOUT ' ("LOGOUT" CR)
                        (CDR (ASSOC 'TOPS20 NETWORKLOGINFO))
        [if (ASSOC 'UNIX NETWORKLOGINFO)
            then
              (PUTASSOC 'LOGOUT ' (WAIT CR "logout" CR)
                        (CDR (ASSOC 'UNIX NETWORKLOGINFO))
        [if (ASSOC 'VMS NETWORKLOGINFO)
            then
              (PUTASSOC 'LOGOUT ' ("LOGOUT" CR)
                        (CDR (ASSOC 'VMS NETWORKLOGINFO))
        [if (ASSOC 'IFS NETWORKLOGINFO)
            then
              (PUTASSOC 'LOGOUT ' ("Quit" CR)
                        (CDR (ASSOC 'IFS NETWORKLOGINFO))
        (if (ASSOC 'NS NETWORKLOGINFO)
            then
              (PUTASSOC 'LOGOUT ' ("Quit" CR)
                        (CDR (ASSOC 'NS NETWORKLOGINFO))])
```

(DEFINEQ

(OPENCHATSTREAM

```
[LAMBDA (HOST) ; Edited 23-Sep-88 16:57 by bbb
 (PROG (OPENFN SLASH PROTOCOL)
 (COND
 ((BOUNDP 'CHAT.PROTOCOLTYPES)
 [if [AND (SETQ SLASH (STRPOS "/" HOST))
          (SETQ PROTOCOL (CDR (CL:ASSOC (SUBSTRING HOST (ADD1 SLASH))
                                       CHAT.PROTOCOL.ABBREVS :TEST 'STRING-EQUAL))
          ; Caller explicitly specified protocol to use
          then
            [if [NOT (SETQ OPENFN (CDR (ASSOC PROTOCOL CHAT.PROTOCOLTYPES))
                then (printout PROMPTWINDOW T (CONCAT "The " PROTOCOL " Chat protocol is not loaded.")
                )
                (RETURN NIL)
            elseif [NOT (SETQ OPENFN (CL:FUNCALL OPENFN (SETQ HOST (SUBSTRING HOST 1 (SUB1 SLASH))
                then (printout PROMPTWINDOW T (CONCAT HOST " is not a recognized " PROTOCOL " host.")
                (RETURN NIL)
            else (RETURN (APPLY* (CADR OPENFN)
                                (CAR OPENFN))
            elseif (AND [SETQ PROTOCOL (CDR (CL:ASSOC HOST CHAT.HOST.TO.PROTOCOL :TEST 'STRING-EQUAL))
                       (SETQ OPENFN (CDR (ASSOC PROTOCOL CHAT.PROTOCOLTYPES)))
                       (SETQ OPENFN (CL:FUNCALL OPENFN HOST))]
            then ; use protocol that worked the last time. Clear PROTOCOL to
                ; skip the test below for remembering it
                (SETQ PROTOCOL NIL)
                (RETURN (APPLY* (CADR OPENFN)
                                (CAR OPENFN)))
            else ; Try all protocols
                (for PAIR in CHAT.PROTOCOLTYPES when (SETQ OPENFN (CL:FUNCALL (CDR PAIR)
                                                                              HOST))
                do ; Value returned is (CanonicalHostName OpenFn)
                    (SETQ PROTOCOL (CAR PAIR))
                    (RETURN (APPLY* (CADR OPENFN)
                                    (CAR OPENFN)))
                (* old code (if (for PROTOCOL in CHAT.PROTOCOLTYPES
                    thereis (SETQ OPENFN (APPLY*
                        (CDR PROTOCOL) HOST))) then
                        (RETURN (APPLY* (CADR OPENFN)
                                        (CAR OPENFN))))))])
```

])

(PROGRAMCHAT

```
[LAMBDA (HOST CMDSTREAM LOGSTREAM) ; Edited 29-Sep-88 00:48 by bbb
```

```
(PROG ((STREAMPAIR (OPENCHATSTREAM HOST))
      INCHAT OUTCHAT)
  [if (STRPOS "/" HOST)
    then (SETQ HOST (SUBSTRING HOST 1 (SUB1 (STRPOS "/" HOST))
      (COND
        (STREAMPAIR (SETQ INCHAT (CAR STREAMPAIR))
          (SETQ OUTCHAT (CDR STREAMPAIR))
          (SETFILEINFO OUTCHAT 'ENDOFSTREAMOP (FUNCTION CHAT.ENDOFSTREAMOP))
          (SETFILEINFO INCHAT 'ENDOFSTREAMOP (FUNCTION CHAT.ENDOFSTREAMOP))
          [ADD.PROCESS `(PROGRAMCHAT.OUTPUT (QUOTE %, INCHAT)
            (QUOTE %, LOGSTREAM)

          (BLOCK)
          (PROGRAMCHAT.LOGIN HOST INCHAT OUTCHAT)
          [COND
            ((STRINGP CMDSTREAM)
              (SETQ CMDSTREAM (OPENSTRINGSTREAM CMDSTREAM 'INPUT)
            [COND
              ((NULL LOGSTREAM)
                (SETQ LOGSTREAM (OPENSTREAM '{NULL} 'OUTPUT)
              (while (AND (OPENP OUTCHAT 'OUTPUT)
                (NOT (EOF? CMDSTREAM)))
                do (BOUT OUTCHAT (BIN CMDSTREAM))
                (BLOCK)
              finally (COND
                ((EOF? CMDSTREAM)
                  (CLOSEF CMDSTREAM)
                  (BOUT OUTCHAT (CHARCODE CR))
                  (PROGRAMCHAT.LOGIN HOST INCHAT OUTCHAT 'LOGOUT)
                  (FORCEOUTPUT OUTCHAT T)
                (until (NOT (OPENP INCHAT 'INPUT)) do (BLOCK) finally (CLOSEF? OUTCHAT]))


```

(PROGRAMCHAT.LOGIN

[LAMBDA (HOST INSTREAM OUTSTREAM OPTION) ; Edited 29-Sep-88 01:08 by bbb

(* Login to HOST. If a job already exists on HOST, Attach to it unless OPTION overrides.)

```
(PROG ((LOGININFO (CDR (ASSOC (OR (GETOSTYPE HOST)
      'IFS)
      NETWORKLOGININFO)))
      NAME/PASS COM)
  (OR LOGININFO (RETURN))
  (SETQ NAME/PASS (\INTERNAL/GETPASSWORD HOST))
  [SETQ COM (COND
    (OPTION)
    ((ASSOC 'ATTACH LOGININFO)
      (OR (CHAT.LOGININFO INSTREAM HOST (CAR NAME/PASS))
        'LOGIN))
    (T
      (* Don't know how to do anything but login, so silly to try
        anything else)
      'LOGIN)
  (COND
    ((NULL (SETQ LOGININFO (ASSOC COM LOGININFO)))
      (printout PROMPTWINDOW T "Login option " COM " not implemented for this type of host"))
    (T (for X in (CDR LOGININFO) do (SELECTQ X
      ((CR LF)
        (BOUT OUTSTREAM (CHARCODE CR))
        (FORCEOUTPUT OUTSTREAM))
      (USERNAME (PRIN3 (CAR NAME/PASS)
        OUTSTREAM))
      (PASSWORD (PRIN3 (\DECRYPT.PWD (CDR NAME/PASS))
        OUTSTREAM))
      (WAIT
        (* Some systems do not permit typeahead)
        (COND
          ((NOT (CHAT.FLUSH&WAIT INSTREAM))
            (* Couldn't sync, so wait longer.)
            (DISMISS CHAT.WAIT.TIME)))
          (DISMISS CHAT.WAIT.TIME))
      (PRIN3 X OUTSTREAM)))
    (FORCEOUTPUT OUTSTREAM))


```

(PROGRAMCHAT.OUTPUT

[LAMBDA (INCHATSTREAM LOGSTREAM) (* ejs%: "23-Feb-85 19:18")

```
(bind CH while (AND (NEQ CH -1)
  (OPENP INCHATSTREAM 'INPUT))
do (SETQ CH (BIN INCHATSTREAM))
[COND
  ((NEQ CH -1)
    (COND
      (LOGSTREAM (BOUT LOGSTREAM CH)
    finally (COND
      ((OPENP INCHATSTREAM)
        (CLOSEF INCHATSTREAM]))


```

)

```
{MEDLEY}<lispusers>PROGRAMCHAT.;1

[if (ASSOC 'TENEX NETWORKLOGINFO)
  then (PUTASSOC 'LOGOUT '("LOGOUT" CR)
         (CDR (ASSOC 'TENEX NETWORKLOGINFO)]

[if (ASSOC 'TOPS20 NETWORKLOGINFO)
  then (PUTASSOC 'LOGOUT '("LOGOUT" CR)
         (CDR (ASSOC 'TOPS20 NETWORKLOGINFO)]

[if (ASSOC 'UNIX NETWORKLOGINFO)
  then (PUTASSOC 'LOGOUT '(WAIT CR "logout" CR)
         (CDR (ASSOC 'UNIX NETWORKLOGINFO)]

[if (ASSOC 'VMS NETWORKLOGINFO)
  then (PUTASSOC 'LOGOUT '("LOGOUT" CR)
         (CDR (ASSOC 'VMS NETWORKLOGINFO)]

[if (ASSOC 'IFS NETWORKLOGINFO)
  then (PUTASSOC 'LOGOUT '("Quit" CR)
         (CDR (ASSOC 'IFS NETWORKLOGINFO)]

[if (ASSOC 'NS NETWORKLOGINFO)
  then (PUTASSOC 'LOGOUT '("Quit" CR)
         (CDR (ASSOC 'NS NETWORKLOGINFO)]

(PUTPROPS PROGRAMCHAT COPYRIGHT ("Xerox Corporation" 1985 1986 1987 1988))
```

FUNCTION INDEX

OPENCHATSTREAM1 PROGRAMCHAT1 PROGRAMCHAT.LOGIN2 PROGRAMCHAT.OUTPUT2
