File created: 12-Sep-91 18:26:50 **{PELE:MV:ENVOS}<LISPCORE>LIBRARY>NSCHAT.;3**

changes to: (FNS NSCHAT.SERVICES NSCHAT.OPEN)

previous date: 12-Jun-90 10:26:34 {PELE:MV:ENVOS}<LISPCORE>LIBRARY>NSCHAT.;2

Read Table: INTERLISP

Package: INTERLISP

Format: XCCS

;;

```
(RPAQQ NSCHATCOMS
       [(COURIERPROGRAMS GAP)
        (FNS NSCHAT.3270.HOST NSCHAT.ATTENTIONFN NSCHAT.ERRORHANDLER NSCHAT.HOST.FILTER NSCHAT.OPEN
             NSCHAT.OPEN.3270 NSCHAT.OPEN.DIALOUT NSCHAT.OPEN.TTYHOST NSCHAT.SERVICES SPP.INPUT.EVENT)
        (INITVARS (NSCHAT.OPTIONS.MENU))
        (DECLARE%: EVAL@COMPILE DONTCOPY (FILES (SOURCE)
                                                SPPDECLS)
               (RECORDS NSCHAT.SERVICE)
               (CONSTANTS (\NS.WKS.Courier 5)
                      (GAP.RemoteSystemAdministration 1)
                      (GAP.RemoteSystemExecutive 2)
                      (GAP.InteractiveTerminalService 3))
               (GLOBALVARS NSCHAT.OPTIONS.MENU))
        (DECLARE%: DONTEVAL@LOAD DOCOPY

               ;; Tell Chat we exist.

               (ADDVARS (CHAT.PROTOCOLTYPES (NS . NSCHAT.HOST.FILTER))
                      (CHAT.PROTOCOL.ABBREVS (X . NS])

(COURIERPROGRAM GAP (3 3)
    TYPES
      [(WaitTime CARDINAL)
       (CharLength (ENUMERATION (five 0)
                          (six 1)
                          (seven 2)
                          (eight 3)))
       (Parity (ENUMERATION (none 0)
                      (odd 1)
                      (even 2)
                      (one 3)
                      (zero 4)))
       (StopBits (ENUMERATION (one 0)
                        (two 1)))
       (FlowControl (RECORD (type (ENUMERATION (none 0)
                                         (xOnXOff 1)))
                           (xOn UNSPECIFIED)
                           (xOff UNSPECIFIED)))
       (SessionHandle (ARRAY 2 UNSPECIFIED))
       [SessionParameterObject (CHOICE (xerox800 0 NIL)
                                      (xerox850 1 UNSPECIFIED)
                                      (xerox860 2 UNSPECIFIED)
                                      (system6 3 (RECORD (sendBlockSize CARDINAL)
                                                        (receiveBlockSize CARDINAL)))
                                      (cmcll 4 (RECORD (sendBlockSize CARDINAL)
                                                      (receiveBlockSize CARDINAL)))
                                      (ibm2770 5 (RECORD (sendBlockSize CARDINAL)
                                                        (receiveBlockSize CARDINAL)))
                                      (ibm2770Host 6 (RECORD (sendBlockSize CARDINAL)
                                                            (receiveBlockSize CARDINAL)))
                                      (ibm6670 7 (RECORD (sendBlockSize CARDINAL)
                                                        (receiveBlockSize CARDINAL)))
                                      (ibm6670Host 8 (RECORD (sendBlockSize CARDINAL)
                                                            (receiveBlockSize CARDINAL)))
                                      (ibm3270 9 NIL)
                                      (ibm3270Host 10 NIL)
                                      (OldTtyHost 11 (RECORD (charLength CharLength)
                                                            (parity Parity)
                                                            (stopBits StopBits)
                                                            (frameTimeout CARDINAL)))
                                      (OldTty 12 (RECORD (charLength CharLength)
                                                        (parity Parity)
                                                        (stopBits StopBits)
                                                        (frameTimeout CARDINAL)))
                                      (other 13 NIL)
                                      (unknown 14 NIL)
                                      (ibm2780 15 (RECORD (sendBlockSize CARDINAL)
                                                         (receiveBlockSize CARDINAL)))
                                      (ibm2780Host 16 (RECORD (sendBlockSize CARDINAL)
                                                             (receiveBlockSize CARDINAL)))
                                      (ibm3780 17 (RECORD (sendBlockSize CARDINAL)
                                                         (receiveBlockSize CARDINAL)))
                                      (ibm3780Host 18 (RECORD (sendBlockSize CARDINAL)
                                                              (receiveBlockSize CARDINAL)))
                                      (siemens9750 19 NIL)
                                      (siemens9750Host 20 NIL)
                                      (ttyHost 21 (RECORD (charLength CharLength)
                                                          (parity Parity)
                                                          (stopBits StopBits)
                                                          (frameTimeout CARDINAL)
                                                          (flowControl FlowControl)))
                                      (tty 22 (RECORD (charLength CharLength)
                                                      (parity Parity)
                                                      (stopBits StopBits)
                                                      (frameTimeout CARDINAL)
                                                      (flowControl FlowControl]
       [TransportObject (CHOICE [rs232c 0 (RECORD (CommParams CommParamObject)
                                                  (preemptOthers ReserveType)
                                                  (preemptMe ReserveType)
                                                  (phoneNumber STRING)
                                                  (line (CHOICE (alreadyReserved 0 (RECORD (resource Resource)))
                                                               (reserveNeeded 1 (RECORD (lineNumber CARDINAL]
                                (bsc 1 (RECORD (localTerminalID STRING)
                                              (localSecurityID STRING)
                                              (lineControl LineControl)
                                              (authenticateProc UNSPECIFIED)))
                                (teletype 2 NIL)
                                (polledBSCController 3 (RECORD (hostControllerName STRING)
                                                              (controllerAddress ControllerAddress)
                                                              (portsOnController CARDINAL)))
                                (sdlcController 4 (RECORD (hostControllerName STRING)
                                                         (controllerAddress ControllerAddress)
                                                         (portsOnController CARDINAL)))
                                (polledBSCTerminal 5 (RECORD (hostControllerName STRING)
                                                            (terminalAddress TerminalAddress)))
                                (sdlcTerminal 6 (RECORD (hostControllerName STRING)
                                                       (terminalAddress TerminalAddress)))
                                (service 7 (RECORD (id LONGCARDINAL)))
                                (unused 8 NIL)
                                (polledBSCPrinter 9 (RECORD (hostControllerName STRING)
                                                           (printerAddress TerminalAddress)))
                                (sdlcPrinter 10 (RECORD (hostControllerName STRING)
                                                       (printerAddress TerminalAddress]
       (Sequence.TransportObject (SEQUENCE TransportObject))
       (BidReply (ENUMERATION (wack 0)
                 (nack 1)
                 (default 2)))
       (ExtendedBoolean (ENUMERATION (true 0)
                        (false 1)
                        (default 2)))
       (DeviceType (ENUMERATION (undefined 0)
                   (terminal 1)
                   (printer 2)))
       [AccessDetail (CHOICE (directConn 0 (RECORD (duplex (ENUMERATION (full 0)
                                                                  (half 1)))
                                                  (lineType LineType)
                                                  (lineSpeed LineSpeed)))
                             (dialConn 1 (RECORD (duplex (ENUMERATION (full 0)
                                                                (half 1)))
                                               (lineType LineType)
                                               (lineSpeed LineSpeed)
                                               (dialMode (ENUMERATION (manual 0)
                                                                 (auto 1)))
                                               (dialerNumber CARDINAL)
                                               (retryCount CARDINAL]
       (CommParamObject (RECORD (accessDetail AccessDetail)))
       (LineType (ENUMERATION (bitSynchronous 0)
                 (byteSynchronous 1)
                 (asynchronous 2)
                 (autoRecognition 3)))
       (LineSpeed (ENUMERATION (bps50 0)
                  (bps75 1)
                  (bps110 2)
                  (bps135p5 3)
                  (bps150 4)
                  (bps300 5)
                  (bps600 6)
                  (bps1200 7)
                  (bps2400 8)
                  (bps3600 9)
                  (bps4800 10)
                  (bps7200 11)
                  (bps9600 12)
                  (bps19200 13)
                  (bps28800 14)
                  (bps38400 15)
                  (bps48000 16)
                  (bps56000 17)
                  (bps57600 18)))
```

```
(LineControl (ENUMERATION (primary 0)
                    (secondary 1)))
(ControllerAddress CARDINAL)
(TerminalAddress CARDINAL)
(credentials (AUTHENTICATION . CREDENTIALS))
(verifier (AUTHENTICATION . VERIFIER))
(Duplexity (ENUMERATION (full 0)
                  (half 1)))
(PortClientType (ENUMERATION (unassigned 0)
                       (outOfService 1)
                       (its 2)
                       (irs 3)
                       (gws 4)
                       (ibm3270Host 5)
                       (ttyEmulation 6)
                       (rbs 7)
                       (fax 8)
                       (mailGateway 9)
                       (phototypesetter 10)))
(PortDialerType (ENUMERATION (none 0)
                       (vadic 1)
                       (hayes 2)
                       (ventel 3)
                       (rs366 4)))
(PortEchoingLocation (ENUMERATION (application 0)
                            (ciu 1)
                            (terminal 2)))
(ReserveType (ENUMERATION (preemptNever 0)
                    (preemptAlways 1)
                    (preemptInactive 2)))
[RS232CData (RECORD (cIUPort BOOLEAN)
                   (owningClientType PortClientType)
                   (preemptionAllowed BOOLEAN)
                   (lineNumber CARDINAL)
                   (dialerNumber CARDINAL)
                   (duplexity Duplexity)
                   (dialingHardware PortDialerType)
                   (charLength CharLength)
                   (echoing PortEchoingLocation)
                   (flowControl FlowControl)
                   (lineSpeed LineSpeed)
                   (parity Parity)
                   (stopBits StopBits)
                   (portActsAsDCE BOOLEAN)
                   (accessControl NSNAME)
                   (validLineSpeeds (SEQUENCE LineSpeed]
(RS232CBack (RECORD (owningCIU STRING)
                   (owningECS STRING)
                   (owningClient STRING)
                   (portNumber CARDINAL)))
(IBMDeviceType (ENUMERATION (unused 0)
                      (model1 1)
                      (model2 2)
                      (model3 3)
                      (model4 4)
                      (model5 5)
                      (printer 6)
                      (other 7)))
(IBM3270Languages (ENUMERATION (USenglish 0)
                         (Austrian 1)
                         (AustrianAlt 2)
                         (German 3)
                         (GermanAlt 4)
                         (Belgian 5)
                         (Brazilian 6)
                         (CanadianFrench 7)
                         (Danish 8)
                         (DanishAlt 9)
                         (Norwegian 10)
                         (NorwegianAlt 11)
                         (Finnish 12)
                         (FinnishAlt 13)
                         (Swedish 14)
                         (SwedishAlt 15)
                         (French 16)
                         (International 17)
                         (Italian 18)
                         (JapaneseEnglish 19)
                         (JapaneseKana 20)
                         (Portuguese 21)
                         (Spanish 22)
                         (SpanishAlt 23)
                         (SpanishSpeaking 24)
                         (UKenglish 25)
                         (unused1 26)
                         (unused2 27)
                         (unused3 28)
```

```
                                (unused4 29)
                                (unused5 30)
                                (unused6 31)))
       (ControllerLinkType (ENUMERATION (sdlc 0)
                                  (bsc 1)))
       (IBM3270Device (RECORD (model IBMDeviceType)
                             (accessControl NSNAME)))
       [IBM3270Controller (RECORD (controllerAddress CARDINAL)
                                 (portsOnController CARDINAL)
                                 (linkType ControllerLinkType)
                                 (language IBM3270Languages)
                                 (devices (SEQUENCE IBM3270Device]
       (IBM3270HostData (SEQUENCE IBM3270Controller))
       (IBM3270HostBack (RECORD (path NSNAME]
    PROCEDURES
      ((Reset 0)
       (Create 2 (SessionParameterObject Sequence.TransportObject WaitTime credentials verifier)
              RETURNS
              (SessionHandle)
              REPORTS
              (badAddressFormat controllerAlreadyExists controllerDoesNotExist dialingHardwareProblem
                     illegalTransport inconsistentParams mediumConnectFailed noCommunicationHardware
                     noDialingHardware terminalAddressInUse terminalAddressInvalid tooManyGateStreams
                     transmissionMediumUnavailable serviceTooBusy userNotAuthenticated userNotAuthorized
                     serviceNotFound registeredTwice transmissionMediumHardwareProblem
                     transmissionMediumUnavailable transmissionMediumNotReady noAnswerOrBusy noRouteToGAPService
                     gapServiceNotResponding courierProtocolMismatch gapVersionMismatch)))
    ERRORS
      ((unimplemented 0)
       (noCommunicationHardware 1)
       (illegalTransport 2)
       (mediumConnectFailed 3)
       (badAddressFormat 4)
       (noDialingHardware 5)
       (dialingHardwareProblem 6)
       (transmissionMediumUnavailable 7)
       (inconsistentParams 8)
       (tooManyGateStreams 9)
       (bugInGAPCode 10)
       (gapNotExported 11)
       (gapCommunicationError 12)
       (controllerAlreadyExists 13)
       (controllerDoesNotExist 14)
       (terminalAddressInUse 15)
       (terminalAddressInvalid 16)
       (serviceTooBusy 17)
       (userNotAuthenticated 18)
       (userNotAuthorized 19)
       (serviceNotFound 20)
       (registeredTwice 21)
       (transmissionMediumHardwareProblem 22)
       (transmissionMediumUnavailable 23)
       (transmissionMediumNotReady 24)
       (noAnswerOrBusy 25)
       (noRouteToGAPService 26)
       (gapServiceNotResponding 27)
       (courierProtocolMismatch 28)
       (gapVersionMismatch 29)))

(DEFINEQ

(NSCHAT.3270.HOST
  [LAMBDA (IBM.VIRTUAL.HOST)                                           (* ejs%: " 3-May-86 18:24")

;;; Retrieves the NSNAME of the physical host on which the virtual host lives

    (COURIER.FETCH (GAP . IBM3270HostBack)
           path
           (CADR (CH.RETRIEVE.ITEM IBM.VIRTUAL.HOST 'IBM3270.HOST.BACK '(GAP . IBM3270HostBack])

(NSCHAT.ATTENTIONFN
  [LAMBDA (STREAM ATTNBYTE)                                            (* bvm%: "11-Mar-85 13:00")
    [SELECTQ ATTNBYTE
         (208 -1)
         (209 (NOTIFY.EVENT (STREAMPROP STREAM 'MediumUpEvent)))
         (COND
            ((OR XIPTRACEFLG NSWIZARDFLG)
             (printout PROMPTWINDOW T "Attention byte: " .I1.8 ATTNBYTE]
    T])

(NSCHAT.ERRORHANDLER
  [LAMBDA (STREAM ERRCODE)                                             (* ejs%: "18-Dec-84 20:43")
    (SELECTQ ERRCODE
         (EOM (SPP.CLEAREOM STREAM))
         (ATTENTION (SPP.CLEARATTENTION STREAM)
                    (NSCHAT.ATTENTIONFN STREAM (BIN STREAM)))
         (END (ADD.CHAT.MESSAGE STREAM "[Connection closed by remote host]")
              -1)
         (COND
            ((SPP.OPENP STREAM)                                   ; non-fatal error?
             (ADD.CHAT.MESSAGE STREAM (CONCAT "[SPP error " ERRCODE "]"))
             (BIN STREAM))
            (T (\EOF.ACTION STREAM])

(NSCHAT.HOST.FILTER
  [LAMBDA (NAME EXPLICIT)                                         ; Edited 15-Feb-90 12:50 by bvm

;;; Return NSCHAT.OPEN if NAME is an NS host with (potentially) a telnet server.

    ;; Unless we were explicitly invoked, we ignore hosts with no colon in name

    (AND \NSFLG (OR EXPLICIT (STRPOS ":" NAME))
         (SETQ NAME (CAR (LOOKUP.NS.SERVER NAME NIL T)))
         (LIST (MKATOM NAME)
               (FUNCTION NSCHAT.OPEN])

(NSCHAT.OPEN
  [LAMBDA (HOST)                                                  ; Edited 12-Sep-91 17:48 by jds

;;; Return a pair of SPP streams for a chat connection, or NIL.  Add CHAT specific operations to the STREAM via STREAMPROP.

    (PROG ((MediumUpEvent (CREATE.EVENT))
           SERVICE.OPTIONS PORT SERVICE OUTSTREAM STREAM HANDLE FAILURE)
      LP  (SETQ SERVICE.OPTIONS (NSCHAT.SERVICES HOST SERVICE))
          [COND
             ([NOT (SETQ SERVICE (COND
                                    ((LITATOM SERVICE.OPTIONS)

                                      ;; The clearinghouse didn't respond, so return T to move to the (NULL SERVICE.OPTIONS) clause.

                                      T)
                                    ((EQLENGTH SERVICE.OPTIONS 1)
                                     (printout PROMPTWINDOW T "Connecting to " (CAAR SERVICE.OPTIONS)
                                            " on " HOST T)
                                     (CADAR SERVICE.OPTIONS))
                                    (T (PROG1 [MENU (create MENU
                                                           ITEMS _ SERVICE.OPTIONS
                                                           TITLE _ "Specific NS Service?"
                                                           WHENSELECTEDFN _ (FUNCTION (LAMBDA (ITEM MENU MOUSE)
                                                                                         (CADR ITEM]
                                               (CLRPROMPT]
              (COND
                 (STREAM (CLOSEF? STREAM)))
              (RETURN NIL))
             ((LITATOM SERVICE.OPTIONS)
              (SETQ FAILURE (CONCAT "Clearinghouse not responding - " SERVICE.OPTIONS)))
             ([NULL (SETQ PORT (LOOKUP.NS.SERVER (CL:FUNCALL (fetch (NSCHAT.SERVICE PhysicalHostFunction)
                                                                  of SERVICE)
                                                        HOST]
              (SETQ FAILURE "Name not found"))
             ([NULL (SETQ STREAM (COURIER.OPEN PORT NIL T 'NSCHAT NIL '(EOM.ON.FORCEOUT T ERRORHANDLER
                                                                              NSCHAT.ERRORHANDLER ATTENTIONFN
                                                                              NSCHAT.ATTENTIONFN]
                                                                  ; No response

              )
             (T (STREAMPROP STREAM 'MediumUpEvent MediumUpEvent) ; the EOFPFN is udf - (STREAMPROP STREAM 'EOFPFN
                                                                  ; (FUNCTION NSCHAT.EOFPFN))
                (SPP.DSTYPE STREAM 192)
                (COND
                   [(NLISTP (SETQ HANDLE (CL:FUNCALL (fetch (NSCHAT.SERVICE OpenFunction) of SERVICE)
                                                 STREAM HOST (fetch (NSCHAT.SERVICE ServiceID) of SERVICE]
                   [(EQ (CAR HANDLE)
                        'ERROR)
                    (SETQ FAILURE (SELECTQ (CADR HANDLE)
                                       ((REJECT serviceNotFound)
                                            [PRINTOUT PROMPTWINDOW T HOST " does not support "
                                                   (CAR (find S in SERVICE.OPTIONS
                                                           suchthat (EQ SERVICE (CADR S]
                                            (GO LP))
                                       (SUBSTRING (CDR HANDLE)
                                              2 -2]
                   (T (STREAMPROP STREAM 'SETDISPLAYTYPE (FUNCTION NILL))
                      (STREAMPROP STREAM 'LOGINFO (FUNCTION NILL))
                      (STREAMPROP STREAM 'FLUSH&WAIT (FUNCTION NILL))
                      (STREAMPROP STREAM 'SENDSCREENPARAMS (FUNCTION NILL))
                      (STREAMPROP STREAM 'READPEVENT (SPP.INPUT.EVENT STREAM))
                      (SETQ OUTSTREAM (SPPOUTPUTSTREAM STREAM))
                      (AWAIT.EVENT MediumUpEvent 15000)
                      (COND
                         ((NEQ SERVICE GAP.InteractiveTerminalService)

                          ;; This is always true.  Do you suppose he means (fetch ServiceID of service)? --bvm
                          (SPP.SENDATTENTION OUTSTREAM 209)))
                      (RETURN (CONS STREAM OUTSTREAM]
          (CLOSEF? STREAM)
          (printout PROMPTWINDOW T "Could not chat to " HOST " because: " (OR FAILURE "No Response"))
          (RETURN NIL])

(NSCHAT.OPEN.3270
  [LAMBDA (STREAM IBM3270.VIRTUAL.HOST)                              (* bvm%: " 7-Oct-86 21:24")
    (LET [(AUTHENTICATOR (CH.GETAUTHENTICATOR T))
          (CONTROLLER (CAR (CADR (CH.RETRIEVE.ITEM IBM3270.VIRTUAL.HOST 'IBM3270HOSTDATA '(GAP
                                                                                               . IBM3270HostData]
         (COND
            ((AND CONTROLLER STREAM)
             (COURIER.CALL STREAM 'GAP 'Create `(ibm3270Host)
                    (SELECTQ (COURIER.FETCH (GAP . IBM3270Controller)
                                    linkType CONTROLLER)
                         (sdlc `((sdlcTerminal (,(CONCAT (fetch (NSNAME NSOBJECT) of (PARSE.NSNAME
                                                                                           IBM3270.VIRTUAL.HOST))
                                                        "#"
                                                        (OCTALSTRING (COURIER.FETCH (GAP . IBM3270Controller)
                                                                            controllerAddress CONTROLLER))
                                                        "B")
                                                1))))
                         (bsc `((polledBSCTerminal (,(CONCAT (fetch (NSNAME NSOBJECT) of (PARSE.NSNAME
                                                                                             IBM3270.VIRTUAL.HOST
                                                                                               ))
                                                            "#"
                                                            (OCTALSTRING (COURIER.FETCH (GAP
                                                                                             . IBM3270Controller)
                                                                                controllerAddress CONTROLLER))
                                                            "B")
                                                    5))))
                         (ERROR "Unknown 3270 controller link type" (COURIER.FETCH (GAP . IBM3270Controller)
                                                                           linkType CONTROLLER)))
                    15000
                    (CAR AUTHENTICATOR)
                    (CADR AUTHENTICATOR)
                    'RETURNERRORS])

(NSCHAT.OPEN.DIALOUT
  [LAMBDA (STREAM HOST)                                              ; Edited  2-Apr-90 16:38 by gadener
    ;; From {PHYLUM}<LISP>MEDLEY>PATCHES>NSCHAT-DIALOUT-PATCH.;1
    (DECLARE (GLOBALVARS CHAT.PHONE.NUMBERS))
    (LET ([RS232CDATA (CADR (CH.RETRIEVE.ITEM HOST 'RS232CDATA '(GAP . RS232CData]
          (AUTHENTICATOR (CH.GETAUTHENTICATOR T))
          PHONENUMBER)
         (COND
            ((NULL RS232CDATA)
             (printout PROMPTWINDOW T HOST " does not appear support dialout service" T)
             (CLOSEF STREAM)
             (ERROR!))
            (T (RESETLST
                   (RESETSAVE NIL (LIST [FUNCTION (LAMBDA (CSTREAM)
                                                     (AND RESETSTATE (CLOSEF CSTREAM]
                                        STREAM))
                   (OR (EQ (COURIER.FETCH (GAP . RS232CData)
                                  dialingHardware of RS232CDATA)
                           'none)
                       (SETQ PHONENUMBER (CHAT.CHOOSE.PHONE.NUMBER)))
                   (COURIER.CALL STREAM 'GAP 'Create `[tty ,(LIST (COURIER.FETCH (GAP . RS232CData)
                                                                         charLength of RS232CDATA)
                                                                   (COURIER.FETCH (GAP . RS232CData)
                                                                         parity of RS232CDATA)
                                                                   (COURIER.FETCH (GAP . RS232CData)
                                                                         stopBits of RS232CDATA)
                                                                   100
                                                                   (COURIER.FETCH (GAP . RS232CData)
                                                                         flowControl of RS232CDATA]
                          [COND
                             [PHONENUMBER `([rs232c (((dialConn (full asynchronous ,(COURIER.FETCH (GAP
                                                                                                    . RS232CData)
                                                                                            lineSpeed of
                                                                                            RS232CDATA)
                                                                      auto
                                                                      ,(COURIER.FETCH (GAP . RS232CData)
                                                                              dialerNumber of RS232CDATA)
                                                                      1)))
                                                     preemptInactive preemptInactive ,PHONENUMBER
                                                      (reserveNeeded (,(COURIER.FETCH (GAP . RS232CData)
                                                                              lineNumber of RS232CDATA]
                                            (teletype]
                             (T `([rs232c ([(directConn (full asynchronous ,(COURIER.FETCH (GAP . RS232CData)
                                                                                    lineSpeed of RS232CDATA]
```

```
                                           preemptInactive preemptInactive NIL
                                           (reserveNeeded (,(COURIER.FETCH (GAP . RS232CData)
                                                                   lineNumber of RS232CDATA]
                                  (teletype]
                          15000
                          (CAR AUTHENTICATOR)
                          (CADR AUTHENTICATOR)
                          'RETURNERRORS))])

(NSCHAT.OPEN.TTYHOST
  [LAMBDA (STREAM HOST SERVICE)                                  (* bvm%: " 7-Oct-86 21:12")
    (LET ((AUTHENTICATOR (CH.GETAUTHENTICATOR)))
         (COURIER.CALL STREAM 'GAP 'Create '(ttyHost (seven even two 100 (none 0 0)))
                `((service (,SERVICE))
                  (teletype))
                15000
                (CAR AUTHENTICATOR)
                (CADR AUTHENTICATOR)
                'RETURNERRORS])

(NSCHAT.SERVICES
  [LAMBDA (HOST ALLFLG)                                          ; Edited 12-Sep-91 17:43 by jds
    (LET [(SERVICES (CONSTANT `((,(LIST (CH.PROPERTY 'FILE.SERVICE)
                                        (CH.PROPERTY 'PRINT.SERVICE)
                                        (CH.PROPERTY 'CLEARINGHOUSE.SERVICE)
                                        (CH.PROPERTY 'EXTERNAL.COMMUNICATION.SERVICE)
                                        (CH.PROPERTY 'GATEWAY.SERVICE)
                                        (CH.PROPERTY 'INTERNET.ROUTING.SERVICE)
                                        (CH.PROPERTY 'MAIL.SERVICE)
                                        (CH.PROPERTY 'REMOTE.BATCH.SERVICE)
                                        10024)
                                 "Remote System Administration"
                                 ,(create NSCHAT.SERVICE
                                         ServiceID _ GAP.RemoteSystemAdministration
                                         PhysicalHostFunction _ (FUNCTION CL:IDENTITY)
                                         OpenFunction _ (FUNCTION NSCHAT.OPEN.TTYHOST))
                                 "Connect to a server executive")
                                (,(CH.PROPERTY 'WORKSTATION)
                                 "Remote System Executive"
                                 ,(create NSCHAT.SERVICE
                                         ServiceID _ GAP.RemoteSystemExecutive
                                         PhysicalHostFunction _ (FUNCTION CL:IDENTITY)
                                         OpenFunction _ (FUNCTION NSCHAT.OPEN.TTYHOST))
                                 "Connect to a remote exec on another workstation")
                                (,(CH.PROPERTY 'INTERACTIVE.TERMINAL.SERVICE)
                                 "Interactive Terminal Service"
                                 ,(create NSCHAT.SERVICE
                                         ServiceID _ GAP.InteractiveTerminalService
                                         PhysicalHostFunction _ (FUNCTION CL:IDENTITY)
                                         OpenFunction _ (FUNCTION NSCHAT.OPEN.TTYHOST))
                                 "Connect to a terminal-based mail reader")
                                (,(CH.PROPERTY 'RS232CDATA)
                                 "RS232 Dialout"
                                 ,(create NSCHAT.SERVICE
                                         ServiceID _ 'DIALOUT
                                         PhysicalHostFunction _ (FUNCTION CL:IDENTITY)
                                         OpenFunction _ (FUNCTION NSCHAT.OPEN.DIALOUT))
                                 "Connect to a dialout facility")
                                (,(CH.PROPERTY 'IBM3270.HOST)
                                 "IBM3270 Emulation Service"
                                 ,(create NSCHAT.SERVICE
                                         ServiceID _ 'IBM3270
                                         PhysicalHostFunction _ (FUNCTION NSCHAT.3270.HOST)
                                         OpenFunction _ (FUNCTION NSCHAT.OPEN.3270))
                                 "Connect to an IBM3270 Emulator Port"]
         (if [AND (NOT ALLFLG)
                 (for S in SERVICES bind (PROPERTIES _ (CADR (CH.LIST.PROPERTIES HOST)))
                    first (COND
                             ((LITATOM PROPERTIES)                       ; CH.LIST.PROPERTIES returned an atom, indicating a failure
                                                                         ; mode.  Return NIL to indicate that we had trouble.
                              (RETFROM 'NSCHAT.SERVICES PROPERTIES)))
                    collect (CDR S) when (for X inside (CAR S) thereis (FMEMB X PROPERTIES]
           else (if (NOT ALLFLG)
                   then (printout PROMPTWINDOW T HOST " does not have any registered NSCHAT services."))
               (printout PROMPTWINDOW T "Please choose a service from the menu.")
                                                                         ; Return them all
               (MAPCAR SERVICES (FUNCTION CDR])

(SPP.INPUT.EVENT
  [LAMBDA (STREAM)                                               (* ejs%: " 2-Jul-85 13:38")

;;; Returns the SPPINPUTEVENT of the associated connection for STREAM, if STREAM is open for INPUT

    (COND
       ((EQ (fetch (STREAM ACCESS) of STREAM)
            'INPUT)
        (fetch (SPPCON SPPINPUTEVENT) of (fetch (SPPSTREAM SPP.CONNECTION) of STREAM)))
       (T (ERROR "FILE NOT OPEN" STREAM])

)

(RPAQ? NSCHAT.OPTIONS.MENU )

(DECLARE%: EVAL@COMPILE DONTCOPY

(FILESLOAD (SOURCE)
       SPPDECLS)

(DECLARE%: EVAL@COMPILE

(RECORD NSCHAT.SERVICE (ServiceID PhysicalHostFunction OpenFunction))
)

(DECLARE%: EVAL@COMPILE

(RPAQQ \NS.WKS.Courier 5)

(RPAQQ GAP.RemoteSystemAdministration 1)

(RPAQQ GAP.RemoteSystemExecutive 2)

(RPAQQ GAP.InteractiveTerminalService 3)

(CONSTANTS (\NS.WKS.Courier 5)
       (GAP.RemoteSystemAdministration 1)
       (GAP.RemoteSystemExecutive 2)
       (GAP.InteractiveTerminalService 3))
)

(DECLARE%: DOEVAL@COMPILE DONTCOPY

(GLOBALVARS NSCHAT.OPTIONS.MENU)
)
)

(DECLARE%: DONTEVAL@LOAD DOCOPY

(ADDTOVAR CHAT.PROTOCOLTYPES (NS . NSCHAT.HOST.FILTER))

(ADDTOVAR CHAT.PROTOCOL.ABBREVS (X . NS))
)

(PUTPROPS NSCHAT COPYRIGHT ("Venue & Xerox Corporation" 1984 1985 1986 1987 1989 1990 1991))
```

FUNCTION INDEX

CONSTANT INDEX

VARIABLE INDEX

RECORD INDEX

COURIERPROGRAM INDEX