

File created: 22-Jul-88 15:43:07 {MCS:MCS:STANFORD}<LANE>TALK.;10

previous date: 16-Jun-88 09:25:17 {MCS:MCS:STANFORD}<LANE>TALK.;9

Read Table: INTERLISP

Package: INTERLISP

Format: MCCS

;;
;; Copyright (c) 1987, 1988 by Stanford University. All rights reserved.

```
(RPAQ TALKCOMS
(* TALK client/server code)
(LOCALVARS . T)
(FNS TALK)
(FNS TALK.RECONNECT TALK.PROCESS TALK.DISPLAY TALK.LISTEN TALK.CLOSEFN TALK.ANSWER TALK.ANSWER.WINDOW
  TALK.ANSWER.USERNAME TALK.GET.NAME TALK.ADD.NAME TALK.FLASH.CARET TALK.WHENSELECTEDFN
  TALK.RINGBELLS TALK.START.SERVER)
(FNS TALK.ICON.BUTTONEVENTFN TALK.ICON.CLOSEFN)
(* TALK data)
(DECLARE%: DONTCOPY (RECORDS TALK.SERVICETYPE TALK.PROTOCOLTYPE))
(VARS TALK.MENU.ITEMS TALK.USER.MESSAGES)
(INITVARS TALK.SERVICETYPES TALK.PROTOCOLTYPES TALK.GAG TALK.HOSTNAMES TALK.ICON.WINDOWS (
  TALK.ANSWER.WAIT
  15)

  (TALK.READTABLE (COPYREADTABLE 'ORIG))
  (TALK.DEFAULT.REGION (CREATEREGION 0 0 500 500))
  (TALK.CLOSED.STRING " -- Connection Closed")
  (TALK.ICON.FONT LITTLEFONT))
(GLOBALVARS TALK.MENU.ITEMS TALK.USER.MESSAGES TALK.SERVICETYPES TALK.PROTOCOLTYPES TALK.GAG
  TALK.HOSTNAMES TALK.ICON.WINDOWS TALK.ANSWER.WAIT TALK.READTABLE TALK.DEFAULT.REGION
  TALK.CLOSED.STRING TALK.ICON.FONT)
(ALISTS (BackgroundMenuCommands TALK))
(VARS (BackgroundMenu))
(APPENDVARS (BACKGROUNDFN TALK.START.SERVER)
  (AFTERMAKESYSFORMS (TALK.START.SERVER NIL T)))
(BITMAPS TALK.ICON.BITMAP)
(GLOBALVARS TALK.ICON.BITMAP)
(P (SETSYNTAX (CHARCODE SPACE)
  (CHARCODE A)
  TALK.READTABLE))))

(* * TALK client/server code)

(DECLARE%: DOEVAL@COMPILE DONTCOPY

(LOCALVARS . T)
)

(DEFINEQ

TALK
[LAMBDA (USER.OR.HOSTNAME SERVICE PROTOCOL)
; Edited 9-Jun-88 12:32 by cdl
(* DECLARATIONS%: (RECORD RESULT
  (SERVICETYPE INPUTSTREAM . OUTPUTSTREAM)))
(PROG (USER PROTOCOLTYPE PROTOCOLTYPES SERVICETYPE SERVICETYPES RESULT ADDRESSABLE?)
  (if (NULL USER.OR.HOSTNAME)
    then (if (SETQ USER.OR.HOSTNAME (TALK.GET.NAME))
      then (if (LISTP USER.OR.HOSTNAME)
        then (RETURN (TALK.RECONNECT USER.OR.HOSTNAME)))
        else (RETURN)))
    (if SERVICE
      then (if [SETQ SERVICETYPE (for SERVICETYPE in TALK.SERVICETYPES thereis (with
        TALK.SERVICETYPE
        SERVICETYPE
        (STRING-EQUAL
        SERVICE
        TALK.SERVICENAME
        ]
        then (SETQ SERVICETYPES (LIST SERVICETYPE))
        else (RETURN (LIST "Unknown service type!" SERVICE)))
      else (if (NULL (SETQ SERVICETYPES TALK.SERVICETYPES))
        then (RETURN "No services available!"))
    (if PROTOCOL
      then (if (SETQ PROTOCOLTYPE (ASSOC PROTOCOL TALK.PROTOCOLTYPES))
        then (SETQ PROTOCOLTYPES (LIST PROTOCOLTYPE))
```

```

    else (RETURN (LIST "Unknown protocol!" PROTOCOL))
  else (if (NULL (SETQ PROTOCOLTYPES TALK.PROTOCOLTYPES))
    then (RETURN "No protocols available!"))
  (if [SETQ PROTOCOLTYPE (bind ADDRESS for PROTOCOLTYPE in PROTOCOLTYPES
    when (with TALK.PROTOCOLTYPE PROTOCOLTYPE (SETQ ADDRESS
      (APPLY*
        TALK.HOSTNAMEFN
          USER.OR.HOSTNAME
        )))
    thereis (PROGN (TALK.ADD.NAME USER.OR.HOSTNAME ADDRESS (with
      TALK.PROTOCOLTYPE
        PROTOCOLTYPE
          TALK.PROTOCOLNAME
        )))
      (SETQ ADDRESSABLE? T)
      (SELECTQ (SETQ RESULT (with TALK.PROTOCOLTYPE
        PROTOCOLTYPE (APPLY*
          TALK.CONNECTFN
            ADDRESS
              SERVICETYPES
            )))
        (ANSWER (RETURN))
        (LISTP RESULT)
      then (with RESULT RESULT (RETURN (TALK.PROCESS INPUTSTREAM OUTPUTSTREAM SERVICETYPE
        PROTOCOLTYPE 'CLIENT USER.OR.HOSTNAME T)))
    else (RETURN (if ADDRESSABLE?
      then (SELECTQ RESULT
        (ANSWER "No answer from TALK service!")
        (LIST "Can't connect to host!" USER.OR.HOSTNAME))
      else (LIST "Host not found!" USER.OR.HOSTNAME)]
  )

```

(DEFINEQ

(TALK.RECONNECT
[LAMBDA (DESTINATION)

; Edited 10-Jun-88 14:59 by cdl
 (* DECLARATIONS%: (RECORD RESULT
 (SERVICETYPE INPUTSTREAM . OUTPUTSTREAM))
 (RECORD DESTINATION (NAME . ENTRIES))
 (RECORD ENTRY (PROTOCOL . ADDRESS)))

(DECLARE (SPECVARS DESTINATION))
 (if TALK.SERVICETYPES

then [LET (PROTOCOLTYPE RESULT ENTRY ADDRESS)

(* try all the protocols but prefer those that have already
 succeeded)

(if [SETQ PROTOCOLTYPE (for PROTOCOLTYPE in [SORT (APPEND TALK.PROTOCOLTYPES)
 (FUNCTION (LAMBDA (PROTOCOLTYPE)

(* DECLARATIONS%: (RECORD DESTINATION
 (NAME . ENTRIES)))

(with TALK.PROTOCOLTYPE
 PROTOCOLTYPE
 (with DESTINATION
 DESTINATION
 (ASSOC
 TALK.PROTOCOLNAME
 ENTRIES]

when [with TALK.PROTOCOLTYPE PROTOCOLTYPE
 (AND [SETQ ADDRESS (with DESTINATION DESTINATION
 (if (SETQ ENTRY (ASSOC
 TALK.PROTOCOLNAME
 ENTRIES
)))
 then (with ENTRY ENTRY
 ADDRESS)
 else (APPLY*
 TALK.HOSTNAMEFN

```

NAME]
(SETQ RESULT (APPLY* TALK.CONNECTFN ADDRESS
TALK.SERVICETYPES]
thereis (SELECTQ RESULT
(ANSWER (RETURN))
(LISTP RESULT)
then (with RESULT RESULT (TALK.PROCESS INPUTSTREAM OUTPUTSTREAM SERVICETYPE
PROTOCOLTYPE 'CLIENT (with DESTINATION
DESTINATION NAME)
T))
else (SELECTQ RESULT
(ANSWER "No answer from TALK service!")
(LIST "Can't connect to host!" (with DESTINATION DESTINATION NAME]
else "No services available!")])

```

```

(TALK.PROCESS
[LAMBDA (INPUTSTREAM OUTPUTSTREAM SERVICETYPE PROTOCOLTYPE MODE USER SPAWN?)
; Edited 9-Jun-88 12:35 by cdl
(if (LITATOM SERVICETYPE)
then (SETQ SERVICETYPE (ASSOC SERVICETYPE TALK.SERVICETYPES)))
(if (LITATOM PROTOCOLTYPE)
then (SETQ PROTOCOLTYPE (ASSOC PROTOCOLTYPE TALK.PROTOCOLTYPES)))
(LET ((DISPLAYSTREAM (TALK.DISPLAY INPUTSTREAM OUTPUTSTREAM SERVICETYPE PROTOCOLTYPE
MODE USER)))
(if SPAWN?
then [ADD.PROCESS '(TALK.LISTEN ,INPUTSTREAM ,OUTPUTSTREAM ,(KWOTE SERVICETYPE)
,(KWOTE PROTOCOLTYPE)
,DISPLAYSTREAM]
else (TALK.LISTEN INPUTSTREAM OUTPUTSTREAM SERVICETYPE PROTOCOLTYPE DISPLAYSTREAM]))

```

```

(TALK.DISPLAY
[LAMBDA (INPUTSTREAM OUTPUTSTREAM SERVICETYPE PROTOCOLTYPE MODE USER)
; Edited 9-Jun-88 14:46 by cdl
(* DECLARATIONS%: (ASSOCRECORD MESSAGES
(GREETING)))
(LET (MAINWINDOW WINDOW REGION GREETING)
(DECLARE (SPECVARS GREETING))
(SETQ USER (with TALK.PROTOCOLTYPE PROTOCOLTYPE (APPLY* TALK.USERNAMEFN INPUTSTREAM
OUTPUTSTREAM SERVICETYPE
MODE USER)))
(with REGION (SETQ REGION (if (REGIONP TALK.DEFAULT.REGION)
then (with REGION TALK.DEFAULT.REGION (GETBOXREGION WIDTH
HEIGHT))
else (GETREGION)))
(SETQ HEIGHT (QUOTIENT HEIGHT 2)))
(SETQ MAINWINDOW (CREATEW (with REGION REGION (create REGION
BOTTOM ← (PLUS BOTTOM HEIGHT) using
REGION
))
(PACK* "TALK (" (with TALK.SERVICETYPE SERVICETYPE
TALK.SERVICENAME)
")))))
(SETQ WINDOW (CREATEW REGION (CONCAT "(" (with TALK.PROTOCOLTYPE PROTOCOLTYPE
TALK.PROTOCOLNAME)
") Talk from " USER)))
(WINDOWPROP MAINWINDOW 'STREAMS (CONS INPUTSTREAM OUTPUTSTREAM))
(WINDOWADDPROP MAINWINDOW 'CLOSEFN (FUNCTION TALK.CLOSEFN))
(ATTACHWINDOW WINDOW MAINWINDOW 'BOTTOM)
(ATTACHMENU (create MENU
ITEMS ← TALK.MENU.ITEMS
CENTERFLG ← T
MENUBORDERSIZE ← 1
WHENSELECTEDFN ← (FUNCTION TALK.WHENSELECTEDFN))
WINDOW
'BOTTOM)
(with TALK.SERVICETYPE SERVICETYPE (APPLY* TALK.DISPLAYFN MAINWINDOW WINDOW
INPUTSTREAM OUTPUTSTREAM PROTOCOLTYPE
USER))

```

else 1]

```

      USER)
    then
      (WINDOWPROP WINDOW 'TALK (LIST USER (CONS PROTOCOL ADDRESS)))
      (WINDOWPROP WINDOW 'EVENT NIL)
      (INVERTW WINDOW)
    else (WINDOWPROP WINDOW 'EVENT NIL)
      (CLOSEW WINDOW))
(WINDOWPROP WINDOW 'RESULT])

```

```

(TALK.ANSWER.WINDOW
[LAMBDA (USER)
  (PROG (WINDOW REGION)
    [if TALK.ICON.WINDOWS
      then [if [AND USER (SETQ WINDOW (for WINDOW in TALK.ICON.WINDOWS thereis (EQUAL
        USER
        (CAR (WINDOWPROP
          WINDOW
          'TALK]
        then (RETURN WINDOW)
      else (SETQ REGION (with REGION (WINDOWPROP (CAR TALK.ICON.WINDOWS)
        'REGION)
        (if (LESSP (PLUS PRIGHT WIDTH)
          SCREENWIDTH)
          then (CREATEREGION PRIGHT BOTTOM WIDTH HEIGHT)
          else (CREATEREGION (OR (fetch (REGION LEFT) of (REGIONP
            TALK.DEFAULT.REGION
            ))
            0)
            (if (LESSP (PLUS PTOP HEIGHT)
              SCREENHEIGHT)
              then PTOP
              else (OR (fetch (REGION BOTTOM) of (REGIONP
                TALK.DEFAULT.REGION
                ))
                0))
            WIDTH HEIGHT]
        else (SETQ REGION (with BITMAP TALK.ICON.BITMAP (if (REGIONP TALK.DEFAULT.REGION)
          then (with REGION TALK.DEFAULT.REGION
            (CREATEREGION LEFT BOTTOM
              BITMAPWIDTH
              BITMAPHEIGHT))
          else (CREATEREGION 0 0 BITMAPWIDTH
            BITMAPHEIGHT]
        (push TALK.ICON.WINDOWS (SETQ WINDOW (CREATEW REGION NIL 0 T)))
        (WINDOWPROP WINDOW 'BUTTONEVENTFN (FUNCTION TALK.ICON.BUTTONEVENTFN))
        (WINDOWPROP WINDOW 'CLOSEFN (FUNCTION TALK.ICON.CLOSEFN))
        (RETURN WINDOW])

```

```

(TALK.ANSWER.USERNAME
[LAMBDA (USER REGION WINDOW)
  (LET (PTR FONTHEIGHT (FONT (DSPFONT NIL WINDOW)))
    (if (AND (GREATERP (NCHARS USER)
      (QUOTIENT (BITMAPWIDTH TALK.ICON.BITMAP)
        (CHARWIDTH (CHARCODE A)
          FONT)))
      (SETQ PTR (STRPOS (CONSTANT (CHARACTER (CHARCODE SPACE)))
        USER)))
    then (DSPFONT TALK.ICON.FONT WINDOW)
      (SETQ FONTHEIGHT (QUOTIENT (FONTPROP TALK.ICON.FONT 'HEIGHT)
        2))
      (CENTERPRINTINREGION (SUBSTRING USER 1 (SUB1 PTR))
        (with REGION REGION (CREATEREGION LEFT (PLUS BOTTOM FONTHEIGHT)
          WIDTH HEIGHT))
        WINDOW)
      (CENTERPRINTINREGION (SUBSTRING USER (ADD1 PTR)
        -1)
        (with REGION REGION (CREATEREGION LEFT (DIFFERENCE BOTTOM FONTHEIGHT)
          WIDTH HEIGHT))

```

(* cdl "10-Jun-87 08:38")

```

      WINDOW)
    (DSPFONT FONT WINDOW)
  else (CENTERPRINTINREGION USER REGION WINDOW])

```

```

(TALK.GET.NAME
[LAMBDA NIL

```

```

; Edited 16-Jun-88 09:24 by cdl
(* DECLARATIONS%: (RECORD ENTRY
(NAME . PAIRS)) (RECORD PAIR
(PROTOCOL . ADDRESS)))

```

```

(LET [HOSTNAME HOSTNAMES MENU (ITEM '("" NIL "")]
  (if (SETQ HOSTNAMES (for ENTRY in TALK.HOSTNAMES
    collect (if (LISTP ENTRY)
      then [with ENTRY ENTRY
        '(,NAME ,(KWOTE ENTRY)
          NIL
          (SUBITEMS ,@(for PAIR in PAIRS
            collect (with PAIR PAIR
              '(, (CONCAT PROTOCOL
                " " ADDRESS)
              ,(KWOTE (LIST NAME
                PAIR]
            else ENTRY)))
        then (push HOSTNAMES ITEM))
    [SETQ MENU (create MENU
      TITLE ← "TALK"
      ITEMS ← '(("Prompt for User/Host" 'PROMPT "Prompt for a new user or hostname.")
        (, (if TALK.GAG
          then "Turn TALK On"
          else "Turn TALK Off")
        (PROGN (SETQ TALK.GAG (NOT TALK.GAG))
          NIL)
        "Toggle TALK connection accept/refuse switch.")
        ,@HOSTNAMES]
    [if HOSTNAMES
      then (SHADEITEM ITEM MENU BLACKSHADE)
        (* Kludge to make entire line of menu inverted, not just up to
        subitem arrows)
      (with REGION (MENUITEMREGION ITEM MENU)
        (with MENU MENU (BLTSHADE BLACKSHADE (with WINDOW IMAGE SAVE)
          (PLUS LEFT MENUOUTLINESIZE)
          (PLUS BOTTOM MENUOUTLINESIZE)
          WIDTH HEIGHT]
        (SELECTQ (SETQ HOSTNAME (MENU MENU))
          (PROMPT (SETQ HOSTNAME (MKATOM (PROMPTFORWARD "User or host?" NIL NIL PROMPTWINDOW)))
            (TERPRI PROMPTWINDOW))
          NIL)
        HOSTNAME])

```

```

(TALK.ADD.NAME
[LAMBDA (NAME ADDRESS PROTOCOL)

```

```

; Edited 9-Jun-88 12:39 by cdl
(* DECLARATIONS%: (RECORD ENTRY
(NAME . PAIRS)))

```

```

(LET (ENTRY)
  (if (NOT (EQUAL NAME ADDRESS))
    then (if (SETQ ENTRY (bind HOSTNAME (NCHARS ← (NCHARS NAME)) for ENTRY in TALK.HOSTNAMES
      everytime (SETQ HOSTNAME (if (LISTP ENTRY)
        then (with ENTRY ENTRY NAME)
        else ENTRY))
      thereis (STRING-EQUAL HOSTNAME NAME)))
    then (if (NLISTP ENTRY)
      then (SETQ TALK.HOSTNAMES (DREMOVE ENTRY TALK.HOSTNAMES))
      (push TALK.HOSTNAMES (LIST NAME (CONS PROTOCOL ADDRESS)))
      else (PUTASSOC PROTOCOL ADDRESS (with ENTRY ENTRY PAIRS)))
    else (push TALK.HOSTNAMES (LIST NAME (CONS PROTOCOL ADDRESS))])

```

```

(TALK.FLASH.CARET
[LAMBDA (WINDOW POSITION FLG)
  (DECLARE (GLOBALVARS DEFAULTCARET))
  (if (OPENWP WINDOW)

```

```

; Edited 2-Jun-88 15:17 by cdl

```

```

then (SELECTQ FLG
      (OFF [with POSITION POSITION (if XCOORD
                                     then (with CURSOR DEFAULTCARET
                                             (BITBLT CUIMAGE NIL NIL WINDOW XCOORD
                                                       YCOORD NIL NIL 'INVERT))
                                     (ON [with POSITION POSITION (with CURSOR DEFAULTCARET (BITBLT CUIMAGE NIL NIL WINDOW
                                                                 (SETQ XCOORD
                                                                  (DIFFERENCE (DSPXPOSITION
                                                                    NIL WINDOW)
                                                                    CUHOTSPOTX))
                                                                  (SETQ YCOORD
                                                                    (DIFFERENCE (DSPYPOSITION
                                                                    NIL WINDOW)
                                                                    CUHOTSPOTY))
                                                                  NIL NIL NIL 'INVERT])
                                     NIL]))

```

(TALK.WHENSELECTEDFN
[LAMBDA (ITEM FROMMENU BUTTON)

; Edited 9-Jun-88 14:50 by cdl
(* DECLARATIONS%: (RECORD STREAMS
(INPUTSTREAM . OUTPUTSTREAM)))

```

(LET [MAINWINDOW TEXTSTREAM STREAMS (WINDOW (MAINWINDOW (WFROMMENU FROMMENU)
      (DECLARE (SPECVARS WINDOW MAINWINDOW TEXTSTREAM STREAMS))
      (SETQ TEXTSTREAM (WINDOWPROP (SETQ MAINWINDOW (MAINWINDOW WINDOW))
                                   'TEXTSTREAM))
      (if (AND (SETQ STREAMS (WINDOWPROP MAINWINDOW 'STREAMS))
              (OPENP (with STREAMS STREAMS OUTPUTSTREAM)))
          then (ERRORSET (CADR ITEM))

```

(TALK.RINGBELLS
[LAMBDA (WINDOW)
 (DECLARE (GLOBALVARS RINGBELLS.L1 RINGBELLS.L2))
 (PLAYTUNE RINGBELLS.L1)

 (FLASHWINDOW WINDOW)
 (PLAYTUNE RINGBELLS.L2))

(* cdl "16-Mar-87 08:01")

(* Dorados and Dolphins can't do PLAYTUNE but let
BEEPON/BEEPOFF handle that)

(TALK.START.SERVER
[LAMBDA (PROTOCOL RESTART)
 (DECLARE (SPECVARS RESTART))
 (if PROTOCOL

; Edited 8-Jun-88 15:06 by cdl

```

    then (LET ((PROTOCOLTYPE (ASSOC PROTOCOL TALK.PROTOCOLTYPES)))
            (DECLARE (SPECVARS PROTOCOLTYPE))
            (if PROTOCOLTYPE
                then [with TALK.PROTOCOLTYPE PROTOCOLTYPE (if TALK.STARTSERVERFN
                                                                then (CAR (NLSETQ (APPLY*
                                                                TALK.STARTSERVERFN
                                                                RESTART]
                                                                else (ERROR PROTOCOL "Unknown protocol!"))))
            else (for PROTOCOLTYPE declare%: (SPECVARS PROTOCOLTYPE) in TALK.PROTOCOLTYPES
                  do (with TALK.PROTOCOLTYPE PROTOCOLTYPE (if TALK.STARTSERVERFN
                                                                then (NLSETQ (APPLY* TALK.STARTSERVERFN
                                                                RESTART])

```

)

(DEFINEQ

(TALK.ICON.BUTTONEVENTFN
[LAMBDA (WINDOW)

; Edited 9-Jun-88 10:02 by cdl
(* DECLARATIONS%: (RECORD DESTINATION
(NAME (PROTOCOL . ADDRESS))))

```

  (RESETFORM (INVERTW WINDOW)
              (until (MOUSESTATE UP) do))
  (ALLOW.BUTTON.EVENTS)
  (if (WINDOWPROP WINDOW 'EVENT)
      then (WINDOWPROP WINDOW 'RESULT T)

```

{MEDLEY}<lispusers>talk>TALK.;1 (TALK.ICON.BUTTONEVENTFN cont.)

Page 8

```
(NOTIFY.EVENT (WINDOWPROP WINDOW 'EVENT NIL)
T)
else (LET ((DESTINATION (WINDOWPROP WINDOW 'TALK))
RESULT)
(if (MOUSECONFIRM (CONCAT "(Re)Connect to " (with DESTINATION DESTINATION NAME)
"?")))
then (if (PROCESSP (SETQ RESULT (TALK.RECONNECT DESTINATION)))
then (CLOSEW WINDOW)
else (FLASHWINDOW WINDOW)
(PROMPTPRINT RESULT))
```

```
(TALK.ICON.CLOSEFN
[LAMBDA (WINDOW)
(LET ((EVENT (WINDOWPROP WINDOW 'EVENT NIL)))
(if EVENT
then (NOTIFY.EVENT EVENT T)))
(SETQ TALK.ICON.WINDOWS (DREMOVE WINDOW TALK.ICON.WINDOWS])
)
(* cdl "10-May-87 10:07")
```

(* * TALK data)

(DECLARE%: DONTCOPY

(DECLARE%: EVAL@COMPILE

(RECORD TALK.SERVICETYPE (TALK.SERVICENAME TALK.DISPLAYFN TALK.LISTENFN))

```
(RECORD TALK.PROTOCOLTYPE (TALK.PROTOCOLNAME TALK.HOSTNAMEFN TALK.USERNAMEFN TALK.CONNECTFN TALK.EVENTFN
TALK.STARTSERVERFN TALK.CASEARRAY))
)
```

```
(RPAQQ TALK.MENU.ITEMS
((Disconnect (TALK.CLOSEFN MAINWINDOW)
"Close TALK connection and keep window open.")
(RingBells (PROGN (PRINTCCODE (CHARCODE ↑G)
(CDR STREAMS))
(FORCEOUTPUT (CDR STREAMS))
(FLASHWINDOW MAINWINDOW))
"Execute a (RINGBELLS) on the remote machine.")
(Message (LET [(MESSAGE (MENU (create MENU ITEMS ← TALK.USER.MESSAGES]
(if [AND MESSAGE (TTY.PROCESSP (WINDOWPROP MAINWINDOW 'PROCESS]
then
(BKSYSBUF MESSAGE)))
"Insert a generic message.")))
```

```
(RPAQQ TALK.USER.MESSAGES (("One moment please" "One moment please..." NIL (SUBITEMS ("the phone's ringing" "One moment
please, the phone's ringing..."
("there's someone at the door" "One moment
please, there's someone at the door..."
("someone is trying to TALK to me" "One
moment please, someone is trying to
TALK to me...")))
```

```
(DATE (DATE)
"The current date and time.")
"Bye."))
```

(RPAQ? TALK.SERVICETYPES NIL)

(RPAQ? TALK.PROTOCOLTYPES NIL)

(RPAQ? TALK.GAG NIL)

(RPAQ? TALK.HOSTNAMES NIL)

(RPAQ? TALK.ICON.WINDOWS NIL)

(RPAQ? TALK.ANSWER.WAIT 15)

(RPAQ? TALK.READTABLE (COPYREADTABLE 'ORIG))

(RPAQ? TALK.DEFAULT.REGION (CREATEREGION 0 0 500 500))

(RPAQ? TALK.CLOSED.STRING " -- Connection Closed")

(RPAQ? TALK.ICON.FONT LITTLEFONT)

(DECLARE%: DOEVAL@COMPILE DONTCOPY

(GLOBALVARS TALK.MENU.ITEMS TALK.USER.MESSAGES TALK.SERVICETYPES TALK.PROTOCOLTYPES TALK.GAG TALK.HOSTNAMES
TALK.ICON.WINDOWS TALK.ANSWER.WAIT TALK.READTABLE TALK.DEFAULT.REGION TALK.CLOSED.STRING TALK.ICON.FONT)
)

(ADDTOVAR BackgroundMenuCommands (Talk '(PRINTOUT PROMPTWINDOW T (TALK)
T)
"Start a TALK session with another user/host."))

(RPAQQ BackgroundMenu NIL)

(APPENDTOVAR BACKGROUNDFNFS TALK.START.SERVER)

(APPENDTOVAR AFTERMAKESYSFORMS (TALK.START.SERVER NIL T))

(RPAQQ TALK.ICON.BITMAP

TALK FROM:
TIME:
MODE:

)

(DECLARE%: DOEVAL@COMPILE DONTCOPY

(GLOBALVARS TALK.ICON.BITMAP)
)

(SETSYNTAX (CHARCODE SPACE)
(CHARCODE A)
TALK.READTABLE)

(PUTPROPS TALK COPYRIGHT ("Stanford University" 1987 1988))

FUNCTION INDEX

TALK	1	TALK.DISPLAY	3	TALK.PROCESS	3
TALK.ADD.NAME	6	TALK.FLASH.CARET	6	TALK.RECONNECT	2
TALK.ANSWER	4	TALK.GET.NAME	6	TALK.RINGBELLS	7
TALK.ANSWER.USERNAME	5	TALK.ICON.BUTTONEVENTFN	7	TALK.START.SERVER	7
TALK.ANSWER.WINDOW	5	TALK.ICON.CLOSEFN	8	TALK.WHENSELECTEDFN	7
TALK.CLOSEFN	4	TALK.LISTEN	4		

VARIABLE INDEX

BackgroundMenu	9	TALK.GAG	8	TALK.MENU.ITEMS	8
BackgroundMenuCommands	9	TALK.HOSTNAMES	8	TALK.PROTOCOLTYPES	8
TALK.ANSWER.WAIT	8	TALK.ICON.BITMAP	9	TALK.READTABLE	8
TALK.CLOSED.STRING	9	TALK.ICON.FONT	9	TALK.SERVICETYPES	8
TALK.DEFAULT.REGION	8	TALK.ICON.WINDOWS	8	TALK.USER.MESSAGES	8

RECORD INDEX

TALK.PROTOCOLTYPE	8	TALK.SERVICETYPE	8
-------------------------	---	------------------------	---
