

File created: 4-Oct-2022 16:42:36 {DSK}<home>larry>medley>sources>IDLER.;2

changes to: (FNS \IDLE.OUT)

previous date: 28-Sep-2022 19:54:40 {DSK}<home>larry>medley>sources>IDLER.;1

Read Table: INTERLISP

Package: INTERLISP

Format: XCCS

::
:: Copyright (c) 1985-1990, 1992, 2022 by Venue & Xerox Corporation.

(RPAQQ IDLERCOMS

([COMS

:: Basic idling facility

```
(FNS IDLE IDLE.SET.OPTION IDLE.SHOW.OPTIONS IDLE.SHOW.OPTION \IDLER \IDLE.WAIT \OK.TO.IDLE? \IDLE.TIME
 \IDLE.OUT \IDLE.EXIT? \IDLE.PROMPT.WATCHER \IDLE.EXIT.ABORT \IDLE.PROMPTING.WINDOW
 \IDLE.IS.PREVIOUS \IDLE.ISMEMBER \IDLE.AUTHENTICATE \IDLERKEYACTION)
(INITVARS (IDLE.PROFILE '(TIMEOUT 0))
```

; so that it doesn't start idling during the loadup

```
(\IDLING)
(CH.DEFAULT.DOMAIN)
(DEFAULTREGISTRY)
(IDLE.KEYACTIONTABLE))
(ADDVARS (SYSTEMINITVARS (IDLE.PROFILE ALLOWED.LOGINS NIL FORGET NIL TIMEOUT 0 DISPLAYFN IDLE.RANDOM
 SAVEVM 5 LOGOUT 5))
 (IDLE.SUSPEND.PROCESS.NAMES MOUSE)
 (IDLE.RESETVARS (PUPTRACEFLG NIL)
 (XIPTRACEFLG NIL)))
(GLOBALVARS IDLE.PROFILE \IDLING \LASTUSERACTION IDLE.RESETVARS IDLE.SUSPEND.PROCESS.NAMES
 CH.DEFAULT.DOMAIN DEFAULTREGISTRY \AFTERLOGINFNS SAVINGCURSOR \VMEM.INHIBIT.WRITE
 \IDLE.PASSWORD.SET)
(LOCALVARS . T)
(DECLARE%: DONTVAL@LOAD DOCOPY (P (FONTCREATE 'TIMESROMAND 36))
 [ADDVARS (BACKGROUNDFNS \IDLE.OUT)
 (BackgroundMenuCommands
 (Idle '(IDLE)
 "Enter Idle mode"
 (SUBITEMS ("Show Profile" '(IDLE.SHOW.OPTIONS)
 "Print current idle options in prompt window")
 ("Set Timeout" '(IDLE.SET.OPTION 'TIMEOUT)
 "Set how long before idling started"
 (SUBITEMS ("Never" (IDLE.SET.OPTION 'TIMEOUT 0)
 "Never spontaneously enter idle mode"))))
 ("Choose Display" '(IDLE.SET.OPTION 'DISPLAYFN)
 "Choose idle display")
 ("Forget" '(IDLE.SHOW.OPTION 'FORGET)
 "Erase password when leaving idle mode?"
 (SUBITEMS ("Do" '(IDLE.SET.OPTION 'FORGET T)
 "Erase password upon exiting idle mode")
 ("Don't" '(IDLE.SET.OPTION 'FORGET NIL)
 "Retain password through idle mode (unless someone
 logs in)"))))
 ["Allowed Logins" '(IDLE.SHOW.OPTION 'ALLOWED.LOGINS)
 "Who can exit idle mode"
 (SUBITEMS ("Unlocked" '(IDLE.SET.OPTION 'ALLOWED.LOGINS 'UNLOCKED)
 "No login required to exit idle mode")
 ("Locked" '(IDLE.SET.OPTION 'ALLOWED.LOGINS 'T))
 "Only the current user may exit idle mode")
 ("Any Login" '(IDLE.SET.OPTION 'ALLOWED.LOGINS '(*))
 "Any user may exit, but require login")
 ("Group" '(IDLE.SET.OPTION 'ALLOWED.LOGINS 'ADD)
 "Only allow specific users and/or groups to exit"
 (SUBITEMS ("Include Previous User"
 '(IDLE.SET.OPTION 'ALLOWED.LOGINS T)
 "If current user exits, check old
 password")
 ("Add Member" '(IDLE.SET.OPTION
 'ALLOWED.LOGINS
 'ADD)
 "Add a group or username")
 ("Remove Member" '(IDLE.SET.OPTION
 'ALLOWED.LOGINS
 'REMOVE)
 "Remove a group or username"]
 ("Authenticate" '(IDLE.SHOW.OPTION 'AUTHENTICATE)
 "Authenticate user upon exiting idle mode?"
 (SUBITEMS ("Do" '(IDLE.SET.OPTION 'AUTHENTICATE T)
 "User will be authenticated upon exiting idle mode")
 ("Unix" '(IDLE.SET.OPTION 'AUTHENTICATE 'UNIX)
 "User will be authenticated in Unix upon exiting idle
 mode")))
```

```

("NS" '(IDLE.SET.OPTION 'AUTHENTICATE 'NS)
 "User will be authenticated in XNS upon exiting idle
 mode")
("GV" '(IDLE.SET.OPTION 'AUTHENTICATE 'GV)
 "User will be authenticated in Grapevine upon exiting
 idle mode")
("Don't" '(IDLE.SET.OPTION 'AUTHENTICATE NIL)
 "Accept any password--no authentication check"]

```

```

[VAR (BackgroundMenu)
 (\IDLING.OVER (CREATE.EVENT '\IDLING.OVER]
 (P (\DAYTIME0 \LASTUSERACTION]
(COMS ;; Default idle display
(FNS IDLE.BOUNCING.BOX IDLE.BITMAP IDLE.RANDOM)
[INITVARS (IDLE.BOUNCING.BOX (BITMAPCOPY LOGOBITMAP))
 (IDLE.FUNCTIONS '(["Bouncing Box" #' (LAMBDA (W)
 (IDLE.BOUNCING.BOX W (USERNAME NIL NIL T)
 (Random 'IDLE.RANDOM]
 (GLOBALVARS IDLE.FUNCTIONS IDLE.BOUNCING.BOX)))

```

;; Basic idling facility

(DEFINEQ

(IDLE

; Edited 20-Nov-87 11:22 by Snow

```

[LAMBDA (FROMTIMEOUT)
 (COND
 ((NOT \IDLING)
 (OR (FNTYP (LISTGET IDLE.PROFILE 'DISPLAYFN))
 (LISTPUT IDLE.PROFILE 'DISPLAYFN 'IDLE.BOUNCING.BOX))
 (\CARET.DOWN)
 (SETQ \IDLING T)
 (ADD.PROCESS (LIST '\IDLER (KWOTE FROMTIMEOUT))
 'RESTARTABLE T 'NAME 'IDLE 'KEYACTION (IDLERKEYACTION]))

```

(IDLE.SET.OPTION

(* drc%: " 3-Jan-86 11:47")

```

[LAMBDA (OPTION X)
 (CLEARW PROMPTWINDOW)
 (IDLE.SHOW.OPTION OPTION "Old")
 (LET ((OLD.OPTION (LISTGET IDLE.PROFILE OPTION)))
 (LISTPUT IDLE.PROFILE OPTION
 (SELECTQ OPTION
 (DISPLAYFN (OR X (MENU (create MENU
 ITEMS _ IDLE.FUNCTIONS))
 OLD.OPTION))
 (TIMEOUT (LET [(MINS (OR X (if (FGETD 'RNUMBER)
 then (RNUMBER "Idle Timeout (in minutes)" NIL NIL NIL T)
 else (MKATOM (PROMPTFORWORD "Idle Timeout:" NIL NIL
 PROMPTWINDOW NIL 'TTY]
 (if (NULL MINS)
 then OLD.OPTION
 elseif (AND (SMALLP MINS)
 (GREATERP MINS 0))
 then MINS
 else NIL)))
 (ALLOWED.LOGINS
 (SELECTQ X
 (UNLOCKED NIL)
 (T (UNION (LIST T)
 OLD.OPTION))
 (ADD (LET [(GROUP (PROMPTFORWORD "Add to allowed login list:" NIL NIL PROMPTWINDOW
 NIL 'TTY]
 (TERPRI PROMPTWINDOW)
 (COND
 ((NULL GROUP)
 OLD.OPTION)
 ((OR (NOT (LISTGET IDLE.PROFILE 'AUTHENTICATE))
 (STREQUAL GROUP "**")
 (STREQUAL GROUP "T")
 (PROGN (PRINTOUT PROMPTWINDOW "Checking..")
 (COND
 ([OR (AND CH.DEFAULT.DOMAIN (STRPOS ":" GROUP)
 (CH.LOOKUP.OBJECT GROUP))
 (AND DEFAULTREGISTRY (LISTP (GV.READENTRY GROUP)
 (PRINTOUT PROMPTWINDOW "..ok" T)
 T)
 (T (EQ 'Y (RESETFORM (TTYDISPLAYSTREAM PROMPTWINDOW)
 (ASKUSER NIL NIL " no such
 name/group. Add anyway? "])
 (CONS GROUP (LISTP OLD.OPTION)))
 (T OLD.OPTION))))
 (REMOVE (AND OLD.OPTION
 (REMOVE (MENU (create MENU
 TITLE _ "Remove group "
 CENTERFLG _ T
 ITEMS _ OLD.OPTION))

```

OLD.OPTION)))

(OR (LISTP X)
OLD.OPTION)))

X)))

(IDLE.SHOW.OPTION OPTION "New"])

(IDLE.SHOW.OPTIONS

[LAMBDA NIL (* bvm%: "16-Oct-85 00:23")
(FRESHLINE PROMPTWINDOW)
(for TAIL on IDLE.PROFILE by (CDDR TAIL) do (IDLE.SHOW.OPTION (CAR TAIL)

NIL
(COND
((CDDR TAIL)
", ")
(T ".")])

(IDLE.SHOW.OPTION

[LAMBDA (OPTION STRING SEPR) (* bvm%: "16-Oct-85 00:23")
(LET ((VALUE (LISTGET IDLE.PROFILE OPTION)))
(OR SEPR (FRESHLINE PROMPTWINDOW))
(COND

(STRING (printout PROMPTWINDOW STRING " ")))
(OR SEPR (printout PROMPTWINDOW "Idle "))
(printout PROMPTWINDOW (SELECTQ OPTION
(ALLOWED.LOGINS
"Allowed Logins")
(L-CASE OPTION T))

": "

(SELECTQ OPTION
((SAVEVM TIMEOUT)
(COND
[(AND (SMALLP VALUE)
(GREATERP VALUE 0))
(CONCAT VALUE " minute" (COND
((EQ VALUE 1)
"")
(T "s"))
(T "never")))]
(ALLOWED.LOGINS
(COND
((LISTP VALUE)
(SUBPAIR '(T *)
'("<Previous User>" "<Anyone>")
VALUE))
(T "Unlocked")))]
(MKSTRING VALUE)))

(COND
(SEPR (printout PROMPTWINDOW SEPR))
(T (TERPRI PROMPTWINDOW]))

(IDLER

[LAMBDA (FROMTIMEOUT) ; Edited 28-Sep-2022 09:05 by lmm
; Edited 22-Sep-2022 15:04 by lmm
; Edited 29-Jun-88 14:36 by drc:

;; This is the main idling loop.

(RESETLST
(RESETSAVE NIL '(SETTOPVAL \IDLING NIL))
(PROG [(START.TIME (ALTO.TO.LISP.DATE \LASTUSERACTION))
VMEM.SAVED W SAVEVM.TIMER IDLE.PROCESS NO.ERROR EXIT? INTERRUPTED.STRING IDLING.KEYACTIONS
(IDLE.TIMEOUT (LISTGET IDLE.PROFILE 'TIMEOUT))
(SAVEVM (LISTGET IDLE.PROFILE 'SAVEVM))
(LOGOUT (LISTGET IDLE.PROFILE 'LOGOUT))

(COND
((NOT (\OK.TO.IDLE?)) ; 'Somebody in password prompt, better not idle
(RETURN)))
(SETQ \IDLE.PASSWORD.SET NIL)

[COND
((EQ (LISTGET IDLE.PROFILE 'FORGET)
'FIRST)

;; do things like dump cache listings and flush files to servers *before* passwords get smashed

(\USEREVENT 'BEFORESAVEVM)
(\DEVICEEVENT 'BEFORESAVEVM)
(\DEVICEEVENT 'AFTERDOSAVEVM)
(\USEREVENT 'AFTERDOSAVEVM]
(RESETSAVE NIL (LIST (FUNCTION NOTIFY.EVENT)
\IDLING.OVER))

[for X in IDLE.SUSPEND.PROCESS.NAMES bind PROC do ; Turn off things like CROCK, LAFITEMAILWATCH,
; SPACEWINDOW, REMINDERS

(COND
((SETQ PROC (FIND.PROCESS X))
(PROCESS.EVAL PROC '(\IDLE.WAIT)

(RESETSAVE \AFTERLOGINFNS NIL) ; So that SETPASSWORD doesn't trigger any activity


```

        T VMEM.SAVED T)
    (until [NOT (OR (KEYDOWNP 'LSHIFT)
                   (KEYDOWNP 'RSHIFT)
                   do (BLOCK 250))
           (AND IDLE.PROCESS (WAKE.PROCESS IDLE.PROCESS]
           (TTY.PROCESS (THIS.PROCESS)) ; Keep us the tty process, even if someone else tries for it
           (GO WAIT.FOR.CHAR))
    (COND
      ((PROCESSP IDLE.PROCESS)
       (SUSPEND.PROCESS IDLE.PROCESS)))
    [SETQ NO.ERROR (NLSETQ (SETQ EXIT? (IDLE.EXIT?
    (COND
      ((NOT NO.ERROR)
       (SETQ INTERRUPTED.STRING "ERROR while checking Allowed Logins")
       (SETPASSWORD NIL (USERNAME NIL NIL T)
                        ""))
      (SETQ \IDLE.PASSWORD.SET 'CLEAR))
      ((NOT EXIT?)
       [SETQ INTERRUPTED.STRING (CONCAT "Someone tried to use the machine at " (DATE (DATEFORMAT
                                                                                       NO.DATE]
                                         (AND IDLE.PROCESS (WAKE.PROCESS IDLE.PROCESS))
                                         (CLEARBUF T)
                                         (GO WAIT.FOR.CHAR))
    EXIT
      (CLOSEW W)
      (FRESHLINE PROMPTWINDOW)
      (AND INTERRUPTED.STRING (PRINTOUT PROMPTWINDOW INTERRUPTED.STRING T))
      (PRINTOUT PROMPTWINDOW "Idle time " (IDLE.TIME START.TIME))
  ;; should be unnecessary (see RESETSAVE above)
      (NOTIFY.EVENT \IDLING.OVER)))
    (COND
      (\IDLE.PASSWORD.SET ; Notify anyone who cares about login change, since we
                          ; suppressed it earlier
      (MAPC \AFTERLOGINFNS (FUNCTION APPLY*]))

```

(\IDLE.WAIT

```

[LAMBDA NIL
 (AWAIT.EVENT \IDLING.OVER)
 (while \IDLING do (BLOCK 500])

```

(\OK.TO.IDLE?

```

[LAMBDA NIL
 (RESETLST
  (OBTAIN.MONITORLOCK \GETPASSWORD.LOCK T T)))]

```

(* bvm%: " 4-Dec-85 15:05")

(\IDLE.TIME

```

[LAMBDA (START.TIME)
 (LET [(GONE (IDIFFERENCE (IDATE)
                          START.TIME))
       (ONEDAY (CONSTANT (IDIFFERENCE (IDATE "2-Jan-80 00:00:00")
                                       (IDATE "1-Jan-80 00:00:00"))
                   ; Express in hours:min:sec
                   ((ILESSP GONE ONEDAY)
                    (GDATE (IPLUS (IDATE "1-Jan-80 00:00:00")
                                   GONE)
                           (DATEFORMAT NO.DATE)))
                   (T (CONCAT (SETQ GONE (QUOTIENT GONE ONEDAY))
                               " day"
                               (COND
                                 ((GREATERP GONE 1)
                                  "s.")
                                 (T ".")))]

```

(* bvm%: "15-Oct-85 23:35")

(\IDLE.OUT

```

[LAMBDA NIL
 (CL:WHEN (NOT \IDLING)
  (NOTIFY.EVENT \IDLING.OVER)
  (LET [(TIMEOUT (LISTGET IDLE.PROFILE 'TIMEOUT)
           (AND (SMALLP TIMEOUT)
                (GREATERP TIMEOUT 0)
                (\SECONDSCLOCKGREATERP \LASTUSERACTION (TIMES TIMEOUT 60))
                (IDLE T)))))]

```

; Edited 4-Oct-2022 16:41 by lmm
(* bvm%: "16-Sep-85 18:34")

(\IDLE.EXIT?

```

[LAMBDA NIL
 (RESETLST
  (RESETSAVE (TTYDISPLAYSTREAM PROMPTWINDOW))
  (CLEARBUF T)
  [PROG ((GROUP (LISTGET IDLE.PROFILE 'ALLOWED.LOGINS))
         (AUTHTYPE (LISTGET IDLE.PROFILE 'AUTHENTICATE))

```

; Edited 22-Nov-88 15:25 by drc:

```
(TIMEOUT (LISTGET IDLE.PROFILE 'LOGIN.TIMEOUT))
(NAME (USERNAME NIL NIL T))
PWD WATCHER)
(COND
  ((NLISTP GROUP) ; no login check at all
   (COND
    ((LISTGET IDLE.PROFILE 'FORGET)
     (SETPASSWORD NIL NAME ""))
    (RETURN T)))
  (COND
   ((EQ 0 (NCHARS NAME)) ; Not logged in, so don't complain about anything
    (RETURN T))
  (OBTAIN.MONITORLOCK \GETPASSWORD.LOCK NIL T) ; Lock out anyone else trying to prompt for a password
  (CLEARW PROMPTWINDOW) ; prompt for password, maybe new username
  [SETQ PWD (COND
   ((AND (EQUAL GROUP ' (T))
          NAME) ; Only previous user allowed to login
    (PROMPTFORWARD (CONCAT NAME " password:")
                   NIL NIL NIL '* TIMEOUT))
   (T [if TIMEOUT
        then ; spawn process to watch for login. Done this way rather than
           ; timeout in \LOGIN.READ because we want to blow away
           ; timed-out password prompt, too.
        (RESETSAVE NIL (LIST 'DEL.PROCESS
                             (SETQ WATCHER
                               (ADD.PROCESS '\IDLE.PROMPT.WATCHER
                                             ', (THIS.PROCESS)
                                             ', TIMEOUT]
                             (PROG1 [CDR (SETQ NAME (CAR (NLSETQ (\LOGIN.READ NIL NAME NIL NIL 'NS)
                                                                (SETQ NAME (MKSTRING (CAR NAME)))
                                                                (if WATCHER
                                                                    then (DEL.PROCESS WATCHER))))]
                             ; decide whether NAME and PWD are in GROUP
  (RETURN (COND
    ((NULL PWD)
     NIL)
    ([AND (OR (MEMB T GROUP)
              (MEMB '* GROUP))
     (\IDLE.IS.PREVIOUS NAME PWD (EQUAL GROUP ' (T]
    ;; Previous user is allowed to login. Also, if only allowed login is old user, but old password is unknown, allow it
    T)
    (\IDLE.ISMEMBER GROUP NAME PWD)
    (COND
     ((OR (NULL AUTHTYPE)
          (\IDLE.AUTHENTICATE NAME PWD AUTHTYPE (NOT (MEMB T GROUP))
                              PROMPTWINDOW))
      (SETPASSWORD NIL NAME PWD)
      (SETQ \IDLE.PASSWORD.SET T)
      T)
     (T (DISMISS 5000) ; Let the error message be visible
         NIL)))
    (T (PRINTOUT PROMPTWINDOW "login incorrect" T)
      (DISMISS 5000) ; Let the error message be visible
      NIL]))]
```

(\IDLE.PROMPT.WATCHER

```
[LAMBDA (PROC TIMEOUT) ; Edited 3-Apr-87 13:56 by bvm:
  ;; Aborts proc if it goes for longer than TIMEOUT (in seconds) with no user action
  (do [DISMISS (TIMES 1000 (IMAX 1 (- TIMEOUT (- (\DAYTIME0 (create FIXP)
                                                         \LASTUSERACTION)
                                                         ; Dismiss until expected timeout
              (if (\SECONDSCLOCKGREATERP \LASTUSERACTION TIMEOUT)
                  then (PROCESS.EVAL PROC '(\IDLE.EXIT.ABORT))
                  (RETURN)])])]
```

(\IDLE.EXIT.ABORT

```
[LAMBDA NIL ; Edited 3-Apr-87 13:37 by bvm:
  ;; Abort process if still sitting under login reader
  (if (RELSTK (STKPOS '\LOGIN.READ))
      then (ERROR!])]
```

(\IDLE.PROMPTING.WINDOW

```
[LAMBDA (TITLE) (* bvm%: " 5-Nov-85 23:10")
```

;;; Replaces PROVIDE.PROMPTING.WINDOW in \LOGIN.READ while idle is on

```
(RESETSAVE (INTERRUPTCHAR 5 'ERROR)) ; Allow ^E to abort prompt
(COND
  ((NEQ (PROCESSPROP (THIS.PROCESS)
                    'NAME)
        'IDLE)
```

```
(OR \IDLE.PASSWORD.SET (SETQ \IDLE.PASSWORD.SET T))
(RESETSAVE (TTYDISPLAYSTREAM PROMPTWINDOW))
(RESETSAVE (TTY.PROCESS (THIS.PROCESS)))
(RESETSAVE (SUSPEND.PROCESS 'IDLE)
' (WAKE.PROCESS IDLE))
(RESETSAVE (SUSPEND.PROCESS 'IDLE.DISPLAY)
' (WAKE.PROCESS IDLE.DISPLAY])
```

(IDLE.IS.PREVIOUS

```
[LAMBDA (NAME PWD NULLOK)
```

; Edited 26-Jan-89 22:38 by NSato.fx

;;; if the new name is the same as the old name, and the old global password wasn't forgotten, then allow the old password

```
(AND (NEQ \IDLE.PASSWORD.SET 'CLEAR)
(LET* [(PREVIOUS.USERNAME (USERNAME NIL NIL T))
(PASSWORDADDR (EMPASSWORDLOC))
(OLDPWD (if (NEQ PASSWORDADDR 0)
then (GetBcplString (EMPOINTER PASSWORDADDR)
if (ZEROP (NCHARS OLDPWD))
then (SETQ OLDPWD))
if (AND (EQ (MACHINETYPE)
'MAIKO)
(NOT OLDPWD))
then ;; when Maiko is first booted the password is empty but we can check w/ UNIX to see if this is the same user s.t.
;; UNIX only looks at first 8 chars of username, so ignore any extra chars typed.
if (> (NCHARS PREVIOUS.USERNAME)
8)
then (SETQ PREVIOUS.USERNAME (SUBSTRING PREVIOUS.USERNAME 1 8)))
if (> (NCHARS NAME)
8)
then (SETQ NAME (SUBSTRING NAME 1 8)))
(AND (STRING-EQUAL PREVIOUS.USERNAME NAME)
(SUBRCALL CHECKBCPLPASSWORD NAME PWD))
else (AND (STRING-EQUAL PREVIOUS.USERNAME NAME)
(COND
(OLDPWD (STRING-EQUAL OLDPWD PWD))
T)
there was no password
NULLOK])
```

(IDLE.ISMEMBER

```
[LAMBDA (GROUP NAME PWD)
```

; Edited 26-Dec-86 20:31 by cutting

```
(OR [for X in GROUP thereis (COND
((EQ X T)
(STRING-EQUAL NAME (USERNAME)))
((STRPOS "*" X)
T)
((STRPOS ":" X)
(EQUAL.CH.NAMES (PARSE.NSNAME NAME)
(PARSE.NSNAME X)))
(T (STRING-EQUAL X (COND
((OR (NULL DEFAULTREGISTRY)
(STRPOS "." NAME)
(NOT (STRPOS "." X)))
NAME)
(T (CONCAT NAME "." DEFAULTREGISTRY]
for X in GROUP thereis (COND
((EQ X T)
NIL)
((AND DEFAULTREGISTRY (STRPOS "^." X))
(PRINTOUT T "... X "?...")
(SELECTQ (GV.ISMEMBERCLOSURE X (\CHECKNAME NAME))
(T (PRINTOUT T "ok.")
T)
(NIL (PRINTOUT T "no.")
NIL)
(BadRName (PRINTOUT T "not a GV group")
NIL)
T))
((AND CH.DEFAULT.DOMAIN (STRPOS ":" X))
(PRINTOUT T "... X "?...")
(SELECTQ (CH.ISMEMBER (PARSE.NSNAME X)
'MEMBERS
'MEMBERS
(CH.LOOKUP.OBJECT NAME))
(T (PRINTOUT T "ok.")
T)
(NIL (PRINTOUT T "no.")
NIL)
(ERROR (PRINTOUT T "not an NS group")
NIL)
T])
```

(IDLE.AUTHENTICATE

(ADDTOVAR **IDLE.SUSPEND.PROCESS.NAMES** MOUSE)

(ADDTOVAR **IDLE.RESETVARS** (PUPTRACEFLG NIL)
(XIPTRACEFLG NIL))

(DECLARE%: DOEVAL@COMPILE DONTCOPY

(GLOBALVARS IDLE.PROFILE \IDLING \LASTUSERACTION IDLE.RESETVARS IDLE.SUSPEND.PROCESS.NAMES CH.DEFAULT.DOMAIN
DEFAULTREGISTRY \AFTERLOGINFNS SAVINGCURSOR \VMEM.INHIBIT.WRITE \IDLE.PASSWORD.SET)
)

(DECLARE%: DOEVAL@COMPILE DONTCOPY

(LOCALVARS . T)
)

(DECLARE%: DONTEVAL@LOAD DOCOPY

(FONTCREATE 'TIMESROMAND 36)

(ADDTOVAR **BACKGROUNDFNNS** \IDLE.OUT)

(ADDTOVAR **BackgroundMenuCommands**

```
[Idle '(IDLE)
  "Enter Idle mode"
  (SUBITEMS ("Show Profile" '(IDLE.SHOW.OPTIONS)
            "Print current idle options in prompt window")
    ("Set Timeout" '(IDLE.SET.OPTION 'TIMEOUT)
     "Set how long before idling started"
     (SUBITEMS ("Never" (IDLE.SET.OPTION 'TIMEOUT 0)
               "Never spontaneously enter idle mode")))
    ("Choose Display" '(IDLE.SET.OPTION 'DISPLAYFN)
     "Choose idle display")
    ("Forget" '(IDLE.SHOW.OPTION 'FORGET)
     "Erase password when leaving idle mode?"
     (SUBITEMS ("Do" '(IDLE.SET.OPTION 'FORGET T)
               "Erase password upon exiting idle mode"
               ("Don't" '(IDLE.SET.OPTION 'FORGET NIL)
                 "Retain password through idle mode (unless someone logs in)"))))
    ["Allowed Logins" '(IDLE.SHOW.OPTION 'ALLOWED.LOGINS)
     "Who can exit idle mode"
     (SUBITEMS ("Unlocked" '(IDLE.SET.OPTION 'ALLOWED.LOGINS 'UNLOCKED)
               "No login required to exit idle mode")
               ("Locked" '(IDLE.SET.OPTION 'ALLOWED.LOGINS 'T)
                 "Only the current user may exit idle mode")
               ("Any Login" '(IDLE.SET.OPTION 'ALLOWED.LOGINS '(*))
                 "Any user may exit, but require login")
               ("Group" '(IDLE.SET.OPTION 'ALLOWED.LOGINS 'ADD)
                 "Only allow specific users and/or groups to exit"
                 (SUBITEMS ("Include Previous User" '(IDLE.SET.OPTION
                                                       'ALLOWED.LOGINS T)
                           "If current user exits, check old password")
                           ("Add Member" '(IDLE.SET.OPTION 'ALLOWED.LOGINS 'ADD)
                             "Add a group or username")
                           ("Remove Member" '(IDLE.SET.OPTION 'ALLOWED.LOGINS
                                                                'REMOVE)
                             "Remove a group or username"]
                 ("Authenticate" '(IDLE.SHOW.OPTION 'AUTHENTICATE)
                 "Authenticate user upon exiting idle mode?"
                 (SUBITEMS ("Do" '(IDLE.SET.OPTION 'AUTHENTICATE T)
                           "User will be authenticated upon exiting idle mode")
                           ("Unix" '(IDLE.SET.OPTION 'AUTHENTICATE 'UNIX)
                             "User will be authenticated in Unix upon exiting idle mode")
                           ("NS" '(IDLE.SET.OPTION 'AUTHENTICATE 'NS)
                             "User will be authenticated in XNS upon exiting idle mode")
                           ("GV" '(IDLE.SET.OPTION 'AUTHENTICATE 'GV)
                             "User will be authenticated in Grapevine upon exiting idle mode")
                           ("Don't" '(IDLE.SET.OPTION 'AUTHENTICATE NIL)
                             "Accept any password--no authentication check"]
                 ))))
    ))
  )
]
```

(RPAQQ **BackgroundMenu** NIL)

(RPAQ **IDLING.OVER** (CREATE.EVENT '\IDLING.OVER))

(\DAYTIME0 \LASTUSERACTION)
)

:: Default idle display

(DEFINEQ

(IDLE.BOUNCING.BOX

[LAMBDA (WINDOW BOX WAIT)

:: Bounce a window around the screen.

(OR WAIT (SETQ WAIT 1000))

(OR BOX (SETQ BOX IDLE.BOUNCING.BOX))

; Edited 3-Sep-87 18:55 by jds

```
(RESETLST
  [LET ((MAXX (WINDOWPROP WINDOW 'WIDTH))
        (MAXY (WINDOWPROP WINDOW 'HEIGHT))
        ORIGBOX X Y BITMAP)
    [for TAIL on [SETQ BOX (COND
                    ((LISTP BOX) ; don't want to trash user's box
                     (COPY BOX))
                    (T (LIST BOX)
                      ; Precompute everything but windows
                      (RPLACA TAIL (IDLE.BITMAP NIL (CAR TAIL))
                                (SETQ ORIGBOX BOX)
                                (SETQ BITMAP (IDLE.BITMAP BITMAP (CAR BOX)))
                                (SETQ BOX (OR (CDR BOX)
                                             ORIGBOX)) ; rotate it
                                [SETQ X (RAND (IDIFFERENCE MAXX (BITMAPWIDTH BITMAP)
                                             (IDIFFERENCE MAXY (BITMAPHEIGHT BITMAP)
                                             (BITBLT BITMAP 0 0 WINDOW X Y NIL NIL NIL 'INVERT)
                                             (BLOCK WAIT)
                                             (BITBLT BITMAP 0 0 WINDOW X Y NIL NIL NIL 'INVERT)))]))]]
```

(IDLE.BITMAP

; Edited 16-Sep-2022 22:33 by larry (* Imm "18-Jan-86 03:01")

```
[LAMBDA (BITMAP BOX)
  (COND
    ((BITMAPP BOX)
     BOX)
    ((WINDOWP BOX)
     (LET* ((REGION (WINDOWPROP BOX 'REGION))
            (WIDTH (fetch (REGION WIDTH) of REGION))
            (HEIGHT (fetch (REGION HEIGHT) of REGION)))
       (OR (AND (BITMAPP BITMAP)
                (EQ (BITMAPWIDTH BITMAP)
                    WIDTH)
                (EQ (BITMAPHEIGHT BITMAP)
                    HEIGHT))
           (SETQ BITMAP (BITMAPCREATE WIDTH HEIGHT)))
         (TOTOPW BOX)
         (BITBLT (SCREENBITMAP)
                  (fetch (REGION LEFT) of REGION)
                  (fetch (REGION BOTTOM) of REGION)
                  BITMAP))
       BITMAP))
    [(LISTP BOX)
     (OR (BITMAPP (CAR BOX))
         (CAR (RPLACA BOX (IDLE.BITMAP NIL (CAR BOX))
                        (T (LET ((FONT (OR (FONTCREATE 'TIMESROMAND 36 NIL NIL NIL T)
                                         (PROGN ; Shouldn't happen unless somebody flushed TIMESROMAND
                                         ; 36 -- don't want to break
                                         (FONTCREATE 'HELVETICA 12 NIL NIL NIL T))
                        DEFAULTFONT)))
          DSP)
        (COND
          ((NOT (AND (OR (STRINGP BOX)
                       (LITATOM BOX))
                    (NEQ (NCHARS BOX)
                          0))))
           (SETQ BOX "Interlisp.org"))
          (SETQ BITMAP (BITMAPCREATE (STRINGWIDTH BOX FONT)
                                     (FONTHEIGHT FONT)))
          (SETQ DSP (DSPCREATE BITMAP))
          (DSPFONT FONT DSP)
          (MOVETO 0 (DIFFERENCE (FONTHEIGHT FONT)
                                (FONTASCENT FONT)))
          DSP)
        (PRIN3 BOX DSP)
        BITMAP])
```

(IDLE.RANDOM

; Edited 28-Sep-2022 19:46 by Imm

```
[LAMBDA (W)
  (LET ([N (IF (BOUNDP 'LAST.IDLE.FUNCTION)
               THEN [IF (IGREATERP (SETQ LAST.IDLE.FUNCTION (SUB1 LAST.IDLE.FUNCTION))
                                   0)
                       THEN LAST.IDLE.FUNCTION
                       ELSE (SETQ LAST.IDLE.FUNCTION (SUB1 (LENGTH IDLE.FUNCTIONS)
                                                             (SETQ LAST.IDLE.FUNCTION)
                                                             (RAND 1 (SUB1 (LENGTH IDLE.FUNCTIONS)
                                                                     CHOICE))
                   (for FN in IDLE.FUNCTIONS when (NEQ 'Random (CAR FN))
                     do (if (< (SETQ N (SUB1 N))
                              1)
                           then (PROMPTPRINT "Idle display " (CAR FN))
                                (DISMISS 1000)
                                (RETURN (APPLY* (EVAL (CADR FN))
                                                W]))
```

```
{MEDLEY}<sources>IDLER.;1
```

Page 11

```
)
```

```
(RPAQ? IDLE.BOUNCING.BOX (BITMAPCOPY LOGOBITMAP))
```

```
(RPAQ? IDLE.FUNCTIONS '(["Bouncing Box" #' (LAMBDA (W)
                                (IDLE.BOUNCING.BOX W (USERNAME NIL NIL T]
                                (Random 'IDLE.RANDOM)))
```

```
(DECLARE%: DOEVAL@COMPILE DONTCOPY
```

```
(GLOBALVARS IDLE.FUNCTIONS IDLE.BOUNCING.BOX)
```

```
)
```

```
(PUTPROPS IDLER COPYRIGHT ("Venue & Xerox Corporation" 1985 1986 1987 1988 1989 1990 1992 2022))
```

FUNCTION INDEX

IDLE	2	IDLE.SHOW.OPTION	3	\IDLE.IS.PREVIOUS	7	\IDLE.TIME	5
IDLE.BITMAP	10	IDLE.SHOW.OPTIONS	3	\IDLE.ISMEMBER	7	\IDLE.WAIT	5
IDLE.BOUNCING.BOX	9	\IDLE.AUTHENTICATE	7	\IDLE.OUT	5	\IDLER	3
IDLE.RANDOM	10	\IDLE.EXIT.ABORT	6	\IDLE.PROMPT.WATCHER	6	\IDLERKEYACTION	8
IDLE.SET.OPTION	2	\IDLE.EXIT?	5	\IDLE.PROMPTING.WINDOW	6	\OK.TO.IDLE?	5

VARIABLE INDEX

BACKGROUNDFNFS	9	IDLE.BOUNCING.BOX	11	IDLE.SUSPEND.PROCESS.NAMES	9
BackgroundMenu	9	IDLE.FUNCTIONS	11	SYSTEMINITVARS	8
BackgroundMenuCommands	9	IDLE.KEYACTIONTABLE	8	\IDLING	8
CH.DEFAULT.DOMAIN	8	IDLE.PROFILE	8	\IDLING.OVER	9
DEFAULTREGISTRY	8	IDLE.RESETVARS	9		
