

File created: 5-May-2023 22:14:14 {WMEDLEY}<lispusers>CALENDAR.;4

edit by: rmk

changes to: (FNS PACKDATE CALTEDITSTRING)

previous date: 1-Feb-2022 17:14:32 {WMEDLEY}<lispusers>CALENDAR.;2

Read Table: INTERLISP

Package: INTERLISP

Format: XCCS

::  
:: Copyright (c) 1985-1990 by Xerox Corporation.

(RPAQQ **CALENDARCOMS**

```
(( (VARS (CALCIRCLEDAY)
      (CALCIRCLEMONTH)
      (CALENDARVERSION "Calendar Version 2.1")
      CALOPTIONSDESC CALOPTIONSDESCLYRIC (LAFITE.AFTER.GETMAIL.FN 'CALPEEKNEWMAIL))
  (INITVARS (CALALERTFLG T)
            (CALCURBROWSER ' (NIL))
            (CALCURDAY)
            (CALDAYBROWSERS)
            (CALDAYDEFAULTREGION ' (32 200 362 100))
            (CALDAYSTART 900)
            (CALDEFAULTALERTDELTA 0)
            (CALDEFAULTHOST&DIR)
            (CALDIRTYREMLST NIL)
            (CALFILELST)
            (CALFLASHTIMES 0)
            (CALFLASHTYPE 'None)
            (CALFONT)
            (CALHARDCOPYPOMFLG T)
            (CALHASH (HARRAY 200))
            (CALHILITETODAY 'CIRCLE)
            (CALKEEPEXPIREDREMSFLG)
            (CALMAINMENU)
            (CALMONLOCK)
            (CALMONTHDEFAULTREGION ' (32 32 500 400))
            (CALMONTHICON)
            (CALMONTHLST)
            (CALNEEDSUPDATE)
            (CALREMCREATEREGION ' (400 400 400 300))
            (CALREMDISPLAYREGION ' (200 400 400 300))
            (CALREMINDEERS)
            (CALREMSLOADED)
            (CALTEDITWINDOW)
            [CALTUNE ' ((750 . 20000)
                      (650 . 20000))
            (CALUPDATEONSHRINKFLG 'Never)
            (CALWATCHMAILFLG 'TEXT)
            (CALYEARICON)
            (PBIGFONT)
            (PCALFONT)
            (PLITTLEFONT))
  (DECLARE%: EVAL@LOAD EVAL@COMPILE DONTCOPY (FILES (LOADCOMP)
                                                    TABLEBROWSER))
  (FNS CALADDEVENT CALCREATEREM CALDELETEREM CALDISPEVENT CALDOPTIONS CALENDAR CALENDARWATCHER
      CALEXTENSEL CALLOADFILE CALMAKEKEY CALMONTHBEF CALMONTHICONFN CALMONTHRBF CALOPTIONMENU
      CALPEEKNEWMAIL CALPRINTREM CALREMDEF CALTBCLOSEFN CALTBCOPYFN CALTBNULLENFN CALTBSELECTEDFN
      CALTEDITEXIT CALTEDITSTRING CALUPDATEFILE CALUPDATEINIT CALYEARICONFN CALYEARINRANGE CIRCLETODAY
      CLEARDAY CLOSEMONTH DAYABBR DAYNAME DAYOF DAYPLUS DAYSIN DERIVENEWDATE DOREMINDER FMNWAYITEM
      GETREMDEF INVERTGROUP LISPDATEDAY LISPDATEMONTH LISPDATYEAR MDMENUITEMREGION MENUITEM
      MENUREGIONITEM MONTHABBR MONTHNAME MONTHNUM MONTHOFDAYPLUS MONTHPLUS MONTHYEARPLUS NEWPARSETIME
      NEXTMDISPLAYREGION PACKDATE PARSETIME PICKFONTSIZE POM POMDAYS PRINTMONTH REMINDERSOF REMINDERTIME
      REMINDERTIMELT REMSINMONTH REPAINTMONTH REPAINTYEAR SAMEDAYAS SAMEMONTHAS SCALEBITMAP SHOWDAY
      SHOWMONTH SHOWMONTHSMALL SHOWMOON SHOWREMSINDAY SHOWREMSINMONTH SHOWYEAR SHRINKMONTH SHRINKYEAR
      TIMEDREMP TPLUS WEEKOF YNCONVERT)
  (BITMAPS CALDAYICON CALMONTHICONMAP CALYEARICONMAP FQMAP FMMAP LQMAP NMMAP)
  (FILES (SYSLOAD FROM VALUEOF DIRECTORIES)
        FREEMENU TABLEBROWSER)))
```

(RPAQQ **CALCIRCLEDAY** NIL)

(RPAQQ **CALCIRCLEMONTH** NIL)

(RPAQQ **CALENDARVERSION** "Calendar Version 2.1")

(RPAQQ **CALOPTIONSDESC**

```
(( (TYPE TITLE LABEL Alert%: FONT (HELVETICA 10 BOLD))
  (TYPE NWAY ID CALALERTFLG LABEL Yes MESSAGE "Reminders will alert you when they fire.")
  (TYPE NWAY ID CALALERTFLG LABEL No MESSAGE "Reminders will not alert you when they fire.")
  ((TYPE TITLE LABEL "Keep expired rems.:" FONT (HELVETICA 10 BOLD))
```

```

(TYPE NWAY ID CALKEEEXPIREDREMSFLG LABEL Yes MESSAGE "Expired reminders will not be deleted.")
(TYPE NWAY ID CALKEEEXPIREDREMSFLG LABEL No MESSAGE "Reminders are deleted automatically when they
fire.")
((TYPE TITLE LABEL "Auto. file update:" FONT (HELVETICA 10 BOLD))
(TYPE NWAY ID CALUPDATEONSHRINKFLG LABEL Always MESSAGE "Update after each reminder is created.")
(TYPE NWAY ID CALUPDATEONSHRINKFLG LABEL Shrink MESSAGE "Update only when you shrink a month window.")
(TYPE NWAY ID CALUPDATEONSHRINKFLG LABEL Never MESSAGE "No automatic updates - use Update in day
browser menu.))
((TYPE EDITSTART LABEL "Alert delta:" ITEMS (CALDEFAULTALERTDELTA)
FONT
(HELVETICA 10 BOLD)
MESSAGE "Default alert time offset in minutes: - for before, + for after.")
(TYPE EDIT ID CALDEFAULTALERTDELTA LABEL 0))
((TYPE EDITSTART LABEL "Host & dir.:" ITEMS (CALDEFAULTHOST&DIR)
FONT
(HELVETICA 10 BOLD))
(TYPE EDIT ID CALDEFAULTHOST&DIR LABEL ""))
((TYPE MOMENTARY LABEL Apply! FONT (HELVETICA 10 BOLD)
SELECTEDFN CALDOPTIONS MESSAGE "Puts the selected options into effect and closes this window.")
(WINDOWPROPS TITLE "Calendar Options"))

```

(RPAQQ CALOPTIONSDESCLYRIC

```

((GROUP (PROPS ID ALERTGROUP)
((TYPE DISPLAY LABEL "Alert:" FONT (HELVETICA 10 BOLD))
(TYPE NWAY COLLECTION CALALERTFLG LABEL Yes MESSAGE "Reminders will alert you when they fire.")
(TYPE NWAY COLLECTION CALALERTFLG LABEL No MESSAGE "Reminders will not alert you when they
fire.")]
[ (GROUP (PROPS ID XGROUP)
((TYPE DISPLAY LABEL "Keep expired rems.:" FONT (HELVETICA 10 BOLD))
(TYPE NWAY COLLECTION CALKEEEXPIREDREMSFLG LABEL Yes MESSAGE "Expired reminders will not be
deleted.")
(TYPE NWAY COLLECTION CALKEEEXPIREDREMSFLG LABEL No MESSAGE "Reminders are deleted
automatically when they fire.")]
[ (GROUP (PROPS ID UPGROUP)
((TYPE DISPLAY LABEL "Auto. file update:" FONT (HELVETICA 10 BOLD))
(TYPE NWAY COLLECTION CALUPDATEONSHRINKFLG LABEL Always MESSAGE "Update after each reminder is
created.")
(TYPE NWAY COLLECTION CALUPDATEONSHRINKFLG LABEL Shrink MESSAGE "Update only when you shrink a
month window.")
(TYPE NWAY COLLECTION CALUPDATEONSHRINKFLG LABEL Never MESSAGE "No automatic updates - use
Update in day browser menu.")]
((TYPE EDITSTART LABEL "Alert delta:" ITEMS (CALDEFAULTALERTDELTA)
FONT
(HELVETICA 10 BOLD)
MESSAGE "Default alert time offset in minutes: - for before, + for after.")
(TYPE EDIT ID CALDEFAULTALERTDELTA LABEL 0))
((TYPE EDITSTART LABEL "Host & dir.:" ITEMS (CALDEFAULTHOST&DIR)
FONT
(HELVETICA 10 BOLD))
(TYPE EDIT ID CALDEFAULTHOST&DIR LABEL ""))
((TYPE MOMENTARY LABEL Apply! FONT (HELVETICA 10 BOLD)
SELECTEDFN CALDOPTIONS MESSAGE "Puts the selected options into effect and closes this window.")
))

```

(RPAQQ LAFITE.AFTER.GETMAIL.FN CALPEEKNEWMAIL)

(RPAQ? CALALERTFLG T)

(RPAQ? CALCURBROWSER ' (NIL) )

(RPAQ? CALCURDAY )

(RPAQ? CALDAYBROWSERS )

(RPAQ? CALDAYDEFAULTREGION ' (32 200 362 100))

(RPAQ? CALDAYSTART 900)

(RPAQ? CALDEFAULTALERTDELTA 0)

(RPAQ? CALDEFAULTHOST&DIR )

(RPAQ? CALDIRTYREMLST NIL)

(RPAQ? CALFILELST )

(RPAQ? CALFLASHTIMES 0)

(RPAQ? CALFLASHTYPE 'None)

(RPAQ? CALFONT )

(RPAQ? CALHARDCOPYPOMFLG T)

(RPAQ? CALHASH (HARRAY 200))

(RPAQ? CALHILITETODAY 'CIRCLE)

```
(RPAQ? CALKEEEXPIREDREMSFLG )
(RPAQ? CALMAINMENU )
(RPAQ? CALMONLOCK )
(RPAQ? CALMONTHDEFAULTREGION '(32 32 500 400))
(RPAQ? CALMONTHICON )
(RPAQ? CALMONTHLST )
(RPAQ? CALNEEDSUPDATE )
(RPAQ? CALREMCREATEREGION '(400 400 400 300))
(RPAQ? CALREMDISPLAYREGION '(200 400 400 300))
(RPAQ? CALREMINDERS )
(RPAQ? CALREMSLOADED )
(RPAQ? CALTEDITWINDOW )
(RPAQ? CALTUNE '((750 . 20000)
                (650 . 20000)))
(RPAQ? CALUPDATEONSHRINKFLG 'Never)
(RPAQ? CALWATCHMAILFLG 'TEXT)
(RPAQ? CALYEARICON )
(RPAQ? PBIGFONT )
(RPAQ? PCALFONT )
(RPAQ? PLITTLEFONT )
(DECLARE%: EVAL@LOAD EVAL@COMPILE DONTCOPY
(FILESLoad (LOADCOMP)
            TABLEBROWSER)
)
```

(DEFINEQ

**(CALADDEVENT**

```
[LAMBDA (M D YR W BROWSER INITMSG MSGSTREAM)
; Edited 14-Oct-88 13:25 by MJD
(* MJD "2-Jul-86 14:10")
(PROG (ANS MSGTITLE DATELST REMDATE REMTIME ALERTFLG ALERTTIME PARSEDALERTTIME PARSEDREMTIME
      AMBIGUOUSTIMEFLG HOUR PMFLG ASTARTPOS TSTARTPOS)
(OBTAIN.MONITORLOCK CALMONLOCK)
[OR MSGSTREAM (WITH.MONITOR CALMONLOCK
                (SETQ MSGSTREAM (CALTEDITSTRING INITMSG M D YR)))]
(if (NOT MSGSTREAM)
  then (printout PROMPTWINDOW T "Reminder aborted")
        (RELEASE.MONITORLOCK CALMONLOCK)
        (RETURN NIL))
(SETQ ANS (COERCETEXTOBJ MSGSTREAM 'STRINGP))
(if (NOT D)
  then ;; This is a rem. coming in via mail, so find its date from the rem. text:
        [SETQ DATELST (\UNPACKDATE (IDATE (CONCAT (SUBSTRING ANS 7 15)
                                                  " 12:00:00"]
        (SETQ D (CADDR DATELST))
        (SETQ M (ADD1 (CADR DATELST)))
        (SETQ YR (CAR DATELST)))
(if (NOT (STRPOS (CONCAT (CHARACTER 13)
                        "Event time: ")
                  ANS))
  then (printout PROMPTWINDOW T "Error parsing event time: reminder aborted")
        (RELEASE.MONITORLOCK CALMONLOCK)
        (RETURN NIL))
(SETQ TSTARTPOS (IPLUS (STRPOS (CONCAT (CHARACTER 13)
                                      "Event time: ")
                                  ANS)
                       13))
[SETQ REMTIME (SUBSTRING ANS TSTARTPOS (SUB1 (STRPOS (CHARACTER 13)
                                                    ANS TSTARTPOS))
(if (STRING-EQUAL REMTIME ">>Time<<")
  then (SETQ REMTIME NIL))
(SETQ REMDATE (PACKDATE (SETQ PARSEDREMTIME (NEWPARSETIME REMTIME))
                        M D YR))
(if (NOT (STRPOS (CONCAT (CHARACTER 13)
                        "Alert time: ")
```

```

      (ANS))
    then (printout PROMPTWINDOW T "Error parsing alert time: reminder aborted")
          (RELEASE.MONITORLOCK CALMONLOCK)
          (RETURN NIL))
  (SETQ ASTARTPOS (IPLUS (STRPOS (CONCAT (CHARACTER 13)
                                     "Alert time: ")
                               ANS)
                        13))
[SETQ ALERTTIME (SUBSTRING ANS ASTARTPOS (SUB1 (STRPOS (CHARACTER 9)
                                                       ANS ASTARTPOS)
                                               ; Alert time field ends with a TAB
                                               )
                                (STRING-EQUAL ALERTTIME ">>Time<<"))
  then (SETQ ALERTTIME NIL)
  (SETQ PARSEDALERTTIME (NEWPARSETIME ALERTTIME))
  (if (NULL PARSEDREMTIME)
    then (printout PROMPTWINDOW T "Sorry - I couldn't parse that time.")
          (CALADDEVENT M D YR W BROWSER ANS)
          (RETURN T)
    elseif (IGREATERP PARSEDREMTIME 2359)
    then (SHOULDNT "Illegal time: must be <= 23:59")
    elseif (AND REMTIME (ILEQ (IDATE REMDATE)
                              (IDATE)))
    then (printout PROMPTWINDOW T "Warning: you have added a reminder with a time in the past.))
                                               ; If user gave an alert time w/o an event time, assume event time
                                               ; = alert time.

  (if (AND (EQ PARSEDREMTIME 0)
          (NEQ PARSEDALERTTIME 0))
    then (SETQ PARSEDREMTIME PARSEDALERTTIME))
;; If user didn't give an alert time, but has a default delta, then derive an alert time from that plus the event time.

  (if (AND (NEQ CALDEFAULTALERTDELTA 0)
          (EQ PARSEDALERTTIME 0))
    then (SETQ PARSEDALERTTIME (TPLUS PARSEDREMTIME CALDEFAULTALERTDELTA)))
  (if (NOT (STRPOS (CONCAT (CHARACTER 9)
                          "Alert: ")
                  ANS))
    then (printout PROMPTWINDOW T "Error parsing alert option: reminder aborted")
          (RELEASE.MONITORLOCK CALMONLOCK)
          (RETURN NIL))
  (SETQ ASTARTPOS (IPLUS (STRPOS (CONCAT (CHARACTER 9)
                                     "Alert: ")
                               ANS)
                        8))
[SETQ ALERTFLG (SUBSTRING ANS ASTARTPOS (SUB1 (STRPOS (CHARACTER 13)
                                                       ANS ASTARTPOS)
                                               )
                                (STRING-EQUAL ALERTFLG "Yes")
                                T)
                                (STRING-EQUAL ALERTFLG "No")
                                NIL)
                                (T CALALERTFLG)))
  (if (AND (IGREATERP (HARRAYPROP CALHASH 'NUMKEYS)
                    0)
          (NOT CALNEEDSUPDATE))
    then (SETQ CALREMSLOADED T))
  (if AMBIGUOUSTIMEFLG
    then (SETQ HOUR (QUOTIENT PARSEDREMTIME 100))
        [if (IGEQ HOUR 12)
          then (SETQ PMFLG T)
              (if (IGEQ HOUR 12)
                then (SETQ HOUR (IDIFFERENCE HOUR 12])
                (printout PROMPTWINDOW "Assuming " HOUR ":" .I2.10.0 (IMOD PARSEDREMTIME 100)
                    (if PMFLG
                      then " p.m."
                      else " a.m."
                    T))
                    ; tell user translated time
                (RELEASE.MONITORLOCK CALMONLOCK)
                [SETQ MSGTITLE (SUBSTRING ANS 24 (SUB1 (STRPOS (CONCAT (CHARACTER 13)
                                                                    "Event time")
                                                                    ANS]
                                                            ; This needs fixing for groups
                                                            )
                                (EQ (WINDOWPROP W 'GROUPEND)
                                    '% )
                                then (WINDOWPROP W 'GROUPEND NIL))
          (for RDAY from D to (OR (WINDOWPROP W 'GROUPEND)
                                D)
            do (CALCREATEREM (LIST MSGTITLE MSGSTREAM)
                            PARSEDREMTIME PARSEDALERTTIME ALERTFLG M RDAY YR BROWSER)
                (AND W (SHOWREMSINDAY W M RDAY YR)))
            (SETQ CALNEEDSUPDATE T)
            (if (NOT CALUPDATEONSHRINKFLG)
              then (CALUPDATEINIT]))

```

(CALCREATEREM

[LAMBDA (MSG REMTIME ALERTTIME ALERTFLG M D YR BROWSER) (\* MJD "23-Feb-88 15:53")

;; MSG is a list of the form (title-string TEdit-stream), REMTIME is a number representing the time

;; ALERTTIME is either a time if > 0, a Timer if < 0, or not used if = 0 (note that this disallows times of 0000, ie. midnight, and should eventually be fixed). ALERTFLG if NIL means do not fire this reminder.

;; Timed reminders are stored on the list CALREMINDERS as (timer-object date-string TB-pointer) The message itself is stored in the data field of the browser item.

;; BROWSER is always supplied, unless the user clicked Middle in the month window to go startight to CALADDEVENT w/o calling Add from a browser menu.

```
(PROG (R REMDATE ITEM)
  (SETQ REMDATE (PACKDATE REMTIME M D YR))
  (SETQ ITEM (create TABLEITEM))
  (SETQ R (LIST (COND
    ((LESSP ALERTTIME 0)
     ALERTTIME)
    ((GREATERP ALERTTIME 0)
     (SETUPTIMER.DATE (PACKDATE ALERTTIME M D YR)))
    ((NEQ REMTIME 0)
     (SETUPTIMER.DATE REMDATE))
    (T NIL))
    REMDATE ITEM))
  (replace TIDATA of ITEM with (APPEND R MSG))
  (if BROWSER
    then (TB.INSERT.ITEM BROWSER ITEM)
         (if (ILESSP (TB.NUMBER.OF.ITEMS BROWSER 'SELECTED)
                    1)
             then (TB.SELECT.ITEM BROWSER ITEM)))
  [if (AND (NEQ REMTIME 0)
           ALERTFLG
           (IGREATERP (IDATE REMDATE)
                      (IDATE)))
    then ;; It's a timed reminder. If he wants an alert AND this rem. is not in the past (we now allow this for historical purposes), then put
          ;; it on CALREMINDERS.
          (if CALREMINDERS
            then (MERGE (LIST R)
                       CALREMINDERS T)
            else (SETQ CALREMINDERS (LIST R)
                    (pushnew CALDIRTYREMLST ITEM)
                    (PUTHASH (CALMAKEKEY M D YR)
                             (SORT (NCONC1 (GETHASH (CALMAKEKEY M D YR)
                                                    CALHASH)
                                           ITEM)
                                   'REMINDERTIMELT)
                             CALHASH]))
  CALHASH])
```

(CALDELETEREM

[LAMBDA (BROWSER ITEM) (\* MJD "10-Dec-87 15:44")

;; ITEM can be either a timed list-form reminder (timer-obj date-str TI-pointer) from CALREMINDERS if this is being called by DOREMINDER, or a TABLEITEM if this is being called by the user via the browser menu.

```
(PROG (M D YR R RTIMESTR DAYBROWSER ITEMKEY)
  (if BROWSER
    then (TB.DELETE.ITEM BROWSER ITEM))
  (SETQ CALDIRTYREMLST (REMOVE ITEM CALDIRTYREMLST))
  (SETQ R (if (EQ (TYPENAME ITEM)
                 'TABLEITEM)
             then (if (EQ (TYPENAME (fetch TIDATA of ITEM))
                         'TABLEITEM)
                     then (fetch TIDATA of (fetch TIDATA of ITEM))
                     else (fetch TIDATA of ITEM))
             else ITEM))
  (SETQ RTIMESTR (CADR R))
  (SETQ M (LISPDATEMONTH RTIMESTR))
  (SETQ D (LISPDATEDAY RTIMESTR))
  (SETQ YR (LISPDATEYEAR RTIMESTR))
  ;; See if this rem. has a browser open so it can be marked as deleted. However, if it has an alert time earlier than its event time, leave it be
  ;; (looks bad to have a rem. crossed out before the event time.)
  [if (AND (NOT BROWSER)
           (LESSP (IDATE RTIMESTR)
                  (IDATE)))
    then (SETQ DAYBROWSER (for B in CALDAYBROWSERS thereis (AND (EQ D (CADR (TB.USERDATA B)))
                                                                (EQ M (CAR (TB.USERDATA B)))
                                                                (EQ YR (CADDR (TB.USERDATA B))))
  [if DAYBROWSER
    then (TB.DELETE.ITEM DAYBROWSER (CAR (LAST ITEM))
  ;; If this is a timed rem and it hasn't expired yet, remove it form CALREMINDERS.
  (if (AND (TIMEDREMP R)
           (IGREATERP (IDATE RTIMESTR)
                      (IDATE)))
    then (SETQ CALREMINDERS (REMOVE [for REM in CALREMINDERS thereis (EQ ITEM (CAR (NTH REM 3)
                                          CALREMINDERS)))
  ;; If this reminder is periodic, its hash key is stored in its 6th slot. If nothing is found there, compute the key the usual way.
  (SETQ ITEMKEY (OR (CAR (NTH R 6))
                   (CALMAKEKEY M D YR))) ; Now that we have the key, we can remove it from the list of
```



```

      (SETQ DSTR (CADR (GETREMDEF I)))
      (SETQ D (LISPDATE DAY DSTR))
      (SETQ M (LISPDATE MONTH DSTR))
      (SETQ YR (LISPDATE YEAR DSTR))
; First, remove the item from its original slot...
      (PUTHASH (CALMAKEKEY M D YR)
        (REMOVE I
          (GETHASH (CALMAKEKEY M D YR)
            )
          CALHASH))
      CALHASH)

```

:: Hash key period codes: 0 = daily; 1-31 = monthly; 32-38 = weekly (32 + day no.); yearly is set by  
:: using 1900 for YR (because of the way calmakekey works.

```

      [SETQ ITEMKEY
        (COND
          ((EQ CHOICE 'Daily)
            0)
          ((EQ CHOICE 'Weekly)
            (IPLUS (DAYOF M D YR)
              32))
          ((EQ CHOICE 'Monthly)
            D)
          ((EQ CHOICE 'Yearly)
            (CALMAKEKEY M D 1900))
; ...and move it to the appropriate periodic slot:
        (replace TIDATA of I
          with (NCONC1 (GETREMDEF I)
            ITEMKEY))

```

:: Note that we save the access key to this item in the rem. itself so that 1. we'll be able to find it if we  
:: need to delete it, and 2. when it fires we can tell it's periodic, figure out its next firing time and put it  
:: back on CALREMINDERS.

```

      (PUTHASH ITEMKEY
        (SORT (NCONC1 (GETHASH ITEMKEY
          CALHASH)
            I)
          'REMINDERTIMELT)
        CALHASH]

```

```

      'CALTBNULLFN)
      (PRINTOUT (GETPROMPTWINDOW (TB.WINDOW BROWSER)
        1)
        T "OK"]
      (TOTOPW (TB.WINDOW BROWSER])

```

### CALDOPTIONS

```

[LAMBDA (ITEM WINDOW BUTTON) (* MJD "9-Dec-87 10:24")

```

:: The conversion to Lyric has turned this routine into a disgusting mess, in particular that whole TYPEP clause.

```

      (PROG [VALLIST (OPTLIST (if (EQ MAKESYSNAME 'KOTO)
        then (FM.READSTATE WINDOW)
        else (FM.GETSTATE WINDOW)
      (SETQ VALLIST (CDR OPTLIST))
      [for ITEM in OPTLIST by (CDDR OPTLIST) as VAL in VALLIST by (CDDR VALLIST) when (NEQ VAL T)
        do (SET ITEM (COND
          ((EQ VAL 'Yes)
            T)
          ((EQ VAL 'No)
            NIL)
          [(AND (NEQ MAKESYSNAME 'KOTO)
            (TYPEP VAL 'FREEMENUITEM))
            (if (AND (EQ (FM.ITEMPROP VAL 'TYPE)
              'NWAY)
                (NEQ (FM.ITEMPROP VAL 'LABEL)
              'Yes)
                (NEQ (FM.ITEMPROP VAL 'LABEL)
              'No))
              then (FM.ITEMPROP VAL 'LABEL)
            else (FM.ITEMPROP VAL 'STATE)]
          (T (MKATOM VAL)]
      (CLOSEW WINDOW)
      (PRINTOUT PROMPTWINDOW T "OK"]])

```

### CALENDAR

```

[LAMBDA (M D YR) (* MJD "9-Dec-87 10:21")

```

; Top-level entry to the program, and public programming  
; interface.

::: If you use any part of Calendar code in your own programs, I would appreciate it if you would include credit to the original author. Thanks.

```

      (pushnew BACKGROUNDFNs 'CALENDARWATCHER)
      [OR (EQ (TYPENAME CALMONLOCK)
        'MONITORLOCK)
      (SETQ CALMONLOCK (CREATE.MONITORLOCK 'CALLOCKNAME)]
      [if (NOT CALDEFAULTHOST&DIR)

```

```

then (SETQ CALDEFAULTHOST&DIR (PROMPTFORWARD "Please enter a default host & directory for reminder
files:" NIL NIL PROMPTWINDOW NIL NIL (CHARCODE EOL]
(COND
  ((type? MENU CALMAINMENU)
   (DELETEMENU CALMAINMENU))
  (SETQ CALMAINMENU (create MENU
    ITEMS _ [APPEND (for YR from (IDIFFERENCE (LISPDATEYEAR (DATE))
      1)
      to (IPLUS (LISPDATEYEAR (DATE))
      3)
      collect (LIST YR YR "Will make a calendar for this year.")
      (LIST '(Other 'OTHER "Lets you choose another year"]
      TITLE _ "Year"
      CENTERFLG _ T
      CHANGEOFFSETFLG _ T
      WHENSELECTEDFN _ 'SHOWYEAR))
(COND
  ((NOT CALFONT)
   (if (AND (NOT M)
            (NOT D)
            (NOT YR))
       then (printout PROMPTWINDOW T "Looking for font TimesRoman 36 - one moment please ...")
            (SETCURSOR WAITINGCURSOR))
   (SETQ CALFONT (FONTCREATE 'TIMESROMAN 36))
   (CURSOR T)))
(COND
  ((AND (NOT M)
        (NOT D)
        (NOT YR))
   (printout T CALENDARVERSION T)
   (printout T "See the Prompt Window for Calendar messages." T)
   (printout PROMPTWINDOW T "Select a year for calendar.")
   (MENU CALMAINMENU))
  [ (EQ M 'TODAY)
    (SHOWDAY (LIST (LISPDATEDAY (DATE))
                  (LISPDATEMONTH (DATE))
                  (LISPDATEYEAR (DATE))
  [ (EQ M 'THISMONTH)
    (SHOWMONTH (LIST NIL (LISPDATEMONTH (DATE))
                       (LISPDATEYEAR (DATE))
  [ (EQ M 'THISYEAR)
    (SHOWYEAR (LIST (LISPDATEYEAR (DATE))
  ((AND (NUMBERP M)
        (NUMBERP D)
        (NUMBERP YR))
   (SHOWDAY (LIST D M YR)))
  ((AND (NOT M)
        (NUMBERP YR))
   (SHOWYEAR (LIST YR)))
  [ (NUMBERP M)
    (SHOWMONTH (LIST NIL M (OR YR (LISPDATEYEAR (DATE))
  (T NIL])

```

**(CALENDARWATCHER**

```

[LAMBDA NIL (* MJD "23-Jun-87 15:53")
  (if (AND CALREMINDERS (TIMEREXPIRED? (CAAR CALREMINDERS)
    'SECONDS))
      then (DOREMINDER (CAR CALREMINDERS]))

```

**(CALEXTENDSEL**

```

[LAMBDA (CALMONTHWINDOW) (* MJD "29-Jan-88 11:06")
  ; Changes the length of a day group selection.
  (PROG [DEND NEWEND [CALMONTHMENU (CAR (WINDOWPROP CALMONTHWINDOW 'MENU)
    [CALCURMONTH (WINDOWPROP CALMONTHWINDOW 'MONTH#))
    [CALCURYEAR (WINDOWPROP CALMONTHWINDOW 'YEAR#]
    (while (MOUSESTATE (ONLY RIGHT))
      do (SETQ DEND (CAR (MENUREGIONITEM CALMONTHWINDOW CALMONTHMENU)))
        (if (EQ DEND '% )
            then (RETURN (TOTOPW CALMONTHWINDOW)) ; He clicked Right in a blank box.
            (OR CALCURDAY (RETURN (TOTOPW CALMONTHWINDOW)) ; Happens if he clicked right before selecting any day -
              ; (SHOWDAY sets CALCURDAY)
            (OR DEND (SETQ DEND CALCURDAY))
            (INVERTGROUP CALCURMONTH CALCURDAY CALCURYEAR CALCURMONTH DEND CALCURYEAR BLACKSHADE
              CALMONTHMENU)
            (SETQ NEWEND (CAR (MENUREGIONITEM CALMONTHWINDOW CALMONTHMENU)))
            ;; At this point we have to check NEWEND for two possibilities: user wandered into a blank box (which makes it a blank), or out of
            ;; the menu entirely (which makes it NIL). Either way, skip it.
            (if (AND NEWEND (NEQ NEWEND '% )
                (ILESSP NEWEND DEND))
                then (INVERTGROUP CALCURMONTH NEWEND CALCURYEAR CALCURMONTH DEND CALCURYEAR WHITESHADE
                  CALMONTHMENU)
                (SETQ DEND NEWEND)))
    (WINDOWPROP CALMONTHWINDOW 'GROUPEND DEND])

```



(CALLOADFILE

[LAMBDA (F)

; Edited 20-Feb-90 16:13 by MJD

:: Each reminder on the file has the format:

:: (timer-value date-string hash-key title-string) [TEdit-text] <CR> \*start\*<CR>. Note that the text may be omitted. The file ends in STOP<CR>.

```
(PROG ((*readtable* (FIND-READTABLE "OLD-INTERLISP-T"))
  (*package* (CL:FIND-PACKAGE "INTERLISP"))
  (R# 0)
  (R#SKIP 0)
  FILE FNAME FILESTREAM TEMPSTREAM RSTREAM R REMLIST ITEM TIMER REMDATE REMSTARTPTR REMENDPTR
  SAMETITLES TITLE ITEMKEY)
  (SETCURSOR WAITINGCURSOR)
  [SETQ FILE (OR F (U-CASE (PROMPTFORWARD "File to load:" NIL NIL PROMPTWINDOW NIL NIL (CHARCODE EOL))
```

:: First see if he typed in a full file name. If not, make it one, using the value of CALDEFAULTHOST&DIR:

```
(if (NOT (MEMBER 'HOST (UNPACKFILENAME FILE)))
  then (SETQ FILE (PACKFILENAME 'NAME FILE 'DIRECTORY CALDEFAULTHOST&DIR)))
```

:: Now that we have a complete name, see if it's really out there:

```
(if (NOT (INFILEP FILE))
  then (PRINTOUT PROMPTWINDOW T FILE " not found. No reminders loaded."
        (CURSOR T)
        (RETURN NIL))
  (if (AND (INFILEP FILE)
          (GETPROP (CAR (MEMBER FILE CALFILELST))
                   'CONTENTS))
      then (if (NOT (MOUSECONFIRM (CONCAT FILE " is already loaded. " "Are you sure you want to do
                                   this?")))
                then (PRINTOUT PROMPTWINDOW T "OK. No reminders loaded."
                      (CURSOR T)
                      (RETURN NIL)))
          (SETQ FILESTREAM (OPENSTREAM FILE 'INPUT 'OLD))
```

:: Open a stream on it and verify that it's a valid file:

```
(if (NEQ (CAR (READ FILESTREAM *readtable*))
        '$$CALREMINDEERS)
  then (PRINTOUT PROMPTWINDOW T FILE " is not a valid reminders file. No reminders loaded."
        (CLOSEF FILESTREAM)
        (CURSOR T)
        (RETURN NIL))
```

:: Looks OK - let's read it:

```
(PRINTOUT PROMPTWINDOW T "Loading " FILE "...")
(until (EQ (SETQ R (READ FILESTREAM *readtable*))
           'STOP)
  do (SETQ TIMER (CAR R)) ; eg. -1558614616
      (SETQ REMDATE (CADR R)) ; eg. "12-Oct-87"
      (SETQ ITEMKEY (CADDR R)) ; eg. 29271
      (SETQ TITLE (CADDRR R)) ; eg. "FOO"
      (* SETQ RSTREAM (OPENTEXTSTREAM))
```

:: 9/28/88: Attempt to fix formatted rem. read-in bug (it's not clear this is all really needed - the old way also seems to work. The only problem may have been just in selecting the right start and end points to copy out of the file):

```
(SETQ TEMPSTREAM (OPENSTREAM '{NODIRCORE} 'BOTH 'NEW))
(SETQ REMSTARTPTR (GETFILEPTR FILESTREAM))
(SETQ REMENDPTR (FILEPOS (CONCAT (CHARACTER 13)
                                 "**start*"
                                 FILESTREAM)))
```

:: Check for Unix-style converted files with LF's instead o CR's:

```
(OR REMENDPTR (SETQ REMENDPTR (FILEPOS (CONCAT (CHARACTER 10)
                                                "**start*"
                                                FILESTREAM))))
(SETFILEPTR FILESTREAM REMSTARTPTR)
(COPYCHARS FILESTREAM TEMPSTREAM (ADD1 REMSTARTPTR)
  REMENDPTR)
(SETFILEPTR TEMPSTREAM 0)
(SETQ RSTREAM (OPENTEXTSTREAM TEMPSTREAM)) ; the reminder text stream.
(* TEDIT.INCLUDE (TEXTOBJ RSTREAM) FILESTREAM
  (ADD1 REMSTARTPTR) REMENDPTR)
```

:: Move past the separator: always skip 8 for the string "<CR>\*start\*". If the just-read rem. had no text, it ended with <CR><CR> and reading stopped with the first <CR>, so we have to skip one more char - that is what the IF tests for.

```
(SETFILEPTR FILESTREAM (IPLUS (GETFILEPTR FILESTREAM)
  8
  (if (GREATERP REMENDPTR (ADD1 REMSTARTPTR))
      then 0
      else 1))) ; Move past the separator.
```

:: Only load this rem. if it hasn't been loaded before. Tests are ordered from easy to hard to minimize performance hit. First see if there are already any rems already in this day. If not, this one must be new. Then compare titles. If any matches there, compare message lengths. Note that this still doesn't \*guarantee\* the rems. are different.

```
(if (OR (NOT (GETHASH ITEMKEY CALHASH))
        (NOT (SETQ SAMETITLES (for ENTRY in (GETHASH ITEMKEY CALHASH)
                                         when (STRING-EQUAL TITLE (CAR (NTH (fetch TIDATA of ENTRY)
```

```

4)))
      collect ENTRY)))
    (NOT (for ENTRY in SAMETITLES
          when [EQ (NCHARS RSTREAM)
                  (NCHARS (CAR (NTH (fetch TIDATA of ENTRY)
                                       5)
                                collect ENTRY)))]
          then (SETQ ITEM (create TABLEITEM)
                (if (AND (ILEQ ITEMKEY 38)
                        (TIMEREXPIRED? TIMER 'SECONDS))
                    then
                        ;; It's a periodic rem. with an expired timer, so we need to find the next future time it will come up so we can
                        ;; add it to CALREMINDEERS.
                        (SETQ REMDATE (DERIVENEWDATE REMDATE ITEMKEY))
                        (SETQ TIMER (SETUPTIMER.DATE REMDATE)))
                    (replace TIDATA of ITEM with (LIST TIMER REMDATE ITEM TITLE RSTREAM ITEMKEY))
                    [if [AND TIMER (NOT (TIMEREXPIRED? TIMER 'SECONDS))
                        then
                            ;; It's a timed reminder. Note that we don't put already expired timers on the list, as might happen when an
                            ;; old file containing timed-keep's is reloaded.
                            (if CALREMINDEERS
                                then (MERGE (LIST (LIST TIMER REMDATE ITEM ITEMKEY))
                                             CALREMINDEERS T)
                                else (SETQ CALREMINDEERS (LIST (LIST TIMER REMDATE ITEM ITEMKEY)
                                                                'REMINDEERTIMELT))
                            ;; Stuff it into the hash array:
                            (PUTHASH ITEMKEY (SORT (NCONC1 (GETHASH ITEMKEY CALHASH)
                                                           ITEM)
                                                  'REMINDEERTIMELT))
                            CALHASH)
                            ;; Keep track of the rems. we're making so we can save it at the end:
                            (SETQ REMLIST (NCONC1 REMLIST ITEM))
                            (add R# 1)
                            (if (EQ (REMAINDER R# 5)
                                    0)
                                then (PRINTOUT PROMPTWINDOW R# ",")
                                else (add R#SKIP 1)))
                ;; Wrap up: First, close the file:
                (CLOSEF FILESTREAM)
                ;; Add the file to the list of known files:
                (pushnew CALFILELST (SETQ FNAME (MKATOM FILE)))
                ;; Save the list of rems. in this file as a prop on the name. You have to do the MEMBER in the case that the user is reloading an already loaded
                ;; file, where the name isn't at the CAR of the list.
                (PUTPROP (CAR (MEMBER FNAME CALFILELST))
                        'CONTENTS REMLIST)
                (SETQ CALREMSLOADED T)
                (OR (EQ (REMAINDER R# 5)
                        0)
                    (printout PROMPTWINDOW R# ",")
                    (PRINTOUT PROMPTWINDOW " done."))
                (if (GREATERP R#SKIP 0)
                    then (PRINTOUT PROMPTWINDOW " " R#SKIP " duplicate" (if (GREATERP R#SKIP 1)
                                                                            then "s"
                                                                            else ""))
                    " skipped."))
                (CURSOR T)
                (RETURN T])

```

```

(CALMAKEKEY
 [LAMBDA (M D YR)
 (BLOCK)
 (LOGOR (LLSH M 12)
        (LLSH D 7)
        (IDIFFERENCE YR 1900])
 (* MJD "20-Nov-86 15:48")

```

```

(CALMONTHBEF
 [LAMBDA (CALMONTHWINDOW)
 (PROG (CALMONTHSTREAM FILE)
 (SETQ CALMONTHSTREAM (WINDOWPROP CALMONTHWINDOW 'DSP))
 (if [AND (MOUSESTATE MIDDLE)
          (NOT (INSIDEP (CREATEREGION 0 0 (WINDOWPROP CALMONTHWINDOW 'WIDTH)
                                                       (WINDOWPROP CALMONTHWINDOW 'HEIGHT))
                       (LASTMOUSEX CALMONTHSTREAM)
                       (LASTMOUSEY CALMONTHSTREAM))
          then (SETQ FILE (MENU (create MENU
                                     ITEMS _ (APPEND CALFILELST (LIST 'Other))
                                     TITLE _ "Load file:"))
                (if (EQ FILE 'Other)
                    then (SETQ FILE (PROMPTFORWORD "File name:" NIL NIL PROMPTWINDOW))
                    (if FILE
                        then (CALLOADFILE FILE)

```

```

      else (PRINTOUT PROMPTWINDOW T "No file given.")
    else (MENUBUTTONFN CALMONTHWINDOW])

```

(CALMONTHICONFN

(\* MJD "17-Jun-87 15:47")

```

[LAMBDA (W ICON)
  (if ICON
    then [ICONW.TITLE ICON (MONTHNAME (WINDOWPROP W 'MONTH#)
      ICON
    else [SETQ CALMONTHICON (create TITLEDICON
      ICON _ CALMONTHICONMAP
      TITLEREG _ '(3 51 56 9)
      (TITLEDICONW CALMONTHICON (MONTHNAME (WINDOWPROP W 'MONTH#))
        LITTLEFONT)])

```

(CALMONTHRBF

(\* MJD "17-Nov-87 16:53")

```

[LAMBDA (CALMONTHWINDOW)
  ;; User clicked in a month window. If inside menu area with left, pass on to menu. If inside menu area with right, extend a selection. If outside
  ;; menu area, do standard window menu.
  (PROG [(CALMONTHSTREAM (WINDOWPROP CALMONTHWINDOW 'DSP)
    (if (INSIDEP [MENUREGION (CAR (WINDOWPROP CALMONTHWINDOW 'MENU)
      (LASTMOUSEX CALMONTHSTREAM)
      (LASTMOUSEY CALMONTHSTREAM))
    then (if (MOUSESTATE LEFT)
      then (MENUBUTTONFN CALMONTHWINDOW)
      else (CALEXTENDSEL CALMONTHWINDOW))
    else (DOWINDOWCOM CALMONTHWINDOW)])

```

(CALOPTIONMENU

; Edited 5-Nov-87 16:58 by MJD

```

[LAMBDA NIL
  (if (EQ MAKESYSNAME 'KOTO)
    then (SETQ CALOPTIONWINDOW (FM.FORMATMENU CALOPTIONSDESC))
      (FM.CHANGELABEL (FM.ITEMFROMID CALOPTIONWINDOW 'CALDEFAULTHOST&DIR)
        CALOPTIONWINDOW CALDEFAULTHOST&DIR)
      (FM.CHANGELABEL (FM.ITEMFROMID CALOPTIONWINDOW 'CALDEFAULTALERTDELTA)
        CALOPTIONWINDOW CALDEFAULTALERTDELTA)
      (FM.CHANGESTATE (FMNWAYITEM CALOPTIONWINDOW 'CALALERTFLG (YNCONVERT CALALERTFLG))
        CALOPTIONWINDOW)
      (FM.CHANGESTATE (FMNWAYITEM CALOPTIONWINDOW 'CALKEEPEXPIREDREMSFLG (YNCONVERT
        CALKEEPEXPIREDREMSFLG))
        CALOPTIONWINDOW)
      (FM.CHANGESTATE (FM.ITEMFROMID CALOPTIONWINDOW CALUPDATEONSHRINKFLG)
        CALOPTIONWINDOW)
      (MOVEW CALOPTIONWINDOW LASTMOUSEX LASTMOUSEY)
      (OPENW CALOPTIONWINDOW)
    else (SETQ CALOPTIONWINDOW (FREEMENU CALOPTIONSDESCLYRIC "Calendar Options"))
      (FM.CHANGELABEL (FM.GETITEM 'CALDEFAULTHOST&DIR NIL CALOPTIONWINDOW)
        CALDEFAULTHOST&DIR CALOPTIONWINDOW)
      (FM.CHANGELABEL (FM.GETITEM 'CALDEFAULTALERTDELTA NIL CALOPTIONWINDOW)
        CALDEFAULTALERTDELTA CALOPTIONWINDOW)
      (FM.CHANGESTATE 'CALALERTFLG (FM.GETITEM (YNCONVERT CALALERTFLG)
        'ALERTGROUP CALOPTIONWINDOW)
        CALOPTIONWINDOW)
      (FM.CHANGESTATE 'CALKEEPEXPIREDREMSFLG (FM.GETITEM (YNCONVERT CALKEEPEXPIREDREMSFLG)
        'XGROUP CALOPTIONWINDOW)
        CALOPTIONWINDOW)
      (FM.CHANGESTATE 'CALUPDATEONSHRINKFLG (FM.GETITEM CALUPDATEONSHRINKFLG 'UPGROUP CALOPTIONWINDOW)
        CALOPTIONWINDOW)
      (MOVEW CALOPTIONWINDOW LASTMOUSEX LASTMOUSEY)
      (OPENW CALOPTIONWINDOW])

```

(CALPEEKNEWMAIL

; Edited 29-Feb-88 16:21 by DENBER  
 ; This is a LAFITE.AFTER.GETMAIL.FN. It checks your mail for  
 ; msgs. that start with "\$CALENDAR" in the subject. These get  
 ; added to your calendar automatically.

;; The decls for this is on {Erinyes}<Lisp>Lyric>Internal>Library>LAFITEDECLS.

```

(PROG (FSTREAM RSTREAM OLDPTR MSTRING MSTARTPOS MSGTEXT (TOT# 0))
  (if CALWATCHMAILFLG
    then (SETQ FSTREAM (fetch FOLDERSTREAM of FOLDER))
      (for MSG in MSGLST when (STRING.EQUAL (SUBSTRING (fetch SUBJECT of MSG)
        1 9)
        "$CALENDAR")
      do (SETQ OLDPTR (GETFILEPTR FSTREAM))
        (TEDIT.INCLUDE (TEXTOBJ (SETQ RSTREAM (OPENTEXTSTREAM)))
          FSTREAM
          (PROGN (SETFILEPTR FSTREAM (fetch START of MSG))
            (IPLUS (FILEPOS (CONCAT (CHARACTER 13)
              (CHARACTER 13))
            FSTREAM)
          2))
        (fetch END of MSG))

```

;; All this stuff is to see if the msg. is a list. If so, see if posting it is allowed before adding it (guards against possible  
;; Trojan horses):

```
(SETQ MSTRING (COERCETEXTOBJ RSTREAM 'STRINGP))
(SETQ MSTARTPOS (IPLUS (OR (STRPOS (CONCAT (CHARACTER 13)
                                           "Message: ")
                                           MSTRING)
                          -9)
                      9))
(SETFILEPTR RSTREAM MSTARTPOS)
(if (IGREATERP (IDIFFERENCE (NCHARS MSTRING)
                            MSTARTPOS)
      1)
    then (SETQ MSGTEXT (READ RSTREAM)))
      (if [OR (NOT (LISTP MSGTEXT))
            (AND (LISTP MSGTEXT)
                 (EQ CALWATCHMAILFLG 'ANY])
          then (CALADDEVENT NIL NIL NIL NIL NIL NIL RSTREAM)
              (add TOT# 1))
          (SETFILEPTR FSTREAM OLDPTR))
      (if (IGREATERP TOT# 0)
          then (PLAYTUNE CALTUNE)
              (PRINTOUT PROMPTWINDOW T TOT# " reminder" (if (EQ TOT# 1)
                                                            then ""
                                                            else "s")
              " posted to Calendar from new mail.]))
```

**(CALPRINTREM**

[LAMBDA (B ITEM STREAM)

(\* MJD " 7-Oct-87 13:52")

; Prints reminder in day box of month window. Caller must set  
; x,y position in STREAM first.

```
(PROG (REMINDER (XOFFSET 0))
      (SETQ REMINDER (fetch TIDATA of ITEM))
      (if (EQ (TYPENAME REMINDER)
              'TABLEITEM)
          then (SETQ REMINDER (fetch TIDATA of REMINDER)))
      (if (TIMEDREMP REMINDER)
          then (PRIN1 (REMINDERTIME REMINDER)
                    STREAM)
              (SPACES 1 STREAM)
              (if (NEQ (IMAGESTREAMTYPE STREAM)
                      'DISPLAY)
                  then (SETQ XOFFSET -10)))
```

;; This kludge is required because IP streams currently do not support clipping regions (SHOWREMSINMONTH sets the clipping region that limits  
;; the line length automatically):

```
(PRIN1 (if (EQ (IMAGESTREAMTYPE STREAM)
                'DISPLAY)
          then (CALREMDEF REMINDER)
          else (OR (SUBSTRING (CALREMDEF REMINDER)
                              1
                              (IPLUS 26 XOFFSET))
                  (CALREMDEF REMINDER)))
      STREAM)
(TERPRI STREAM])
```

; The OR above hinges on the fact that SUBSTRING returns NIL  
; if its arg is too big.

**(CALREMDEF**

[LAMBDA (REMINDER)

(\* MJD " 5-Jun-87 12:48")

(\* Return reminder message title text.)

(CAR (NTH REMINDER 4])

**(CALTBCLOSEFN**

[LAMBDA (BROWSER W TYPE)

(\* MJD "16-Nov-87 12:50")

; Before closing a day browser, remove it from the list of active  
; browsers.

```
(if (EQ TYPE 'CLOSE)
    then (SETQ CALDAYBROWSERS (REMOVE BROWSER CALDAYBROWSERS)))
NIL])
```

**(CALTBCOPYFN**

[LAMBDA (BROWSER ITEM)

(\* MJD "23-Feb-88 17:00")

; Copy a rem. from BROWSER into previously selected browser.

```
(PROG (DDATE DBROWSER M D YR R REMTIME MSG ALERTTIME ALERTFLG DATELST)
```

;; CALCURBROWSER is a dotted pair containing (source-browser . dest.-browser). It is set by CALTBSELECTEDFN every time you click in a  
;; day browser.

```
(if (NOT (CDR CALCURBROWSER))
    then (PRINTOUT PROMPTWINDOW T "Please select a destination for copy first.")
        (RETURN NIL)
    else (SETQ DBROWSER (CDR CALCURBROWSER))
        (SETQ DDATE (TB.USERDATA DBROWSER))
        (SETQ M (CAR DDATE))
```

```

(SETQ D (CADR DDATE))
(SETQ YR (CADDR DDATE))
(SETQ R (fetch TIDATA of ITEM))
(SETQ REMTIME (OR (REMINDERTIME R)
0))
[SETQ MSG (LIST (CALREMDEF R)
(CAR (NTH (GETREMDEF (CAR (NTH R 3)))
5]
;; Extract the actual remind-time from the old Timer, so CALCREATEREM will know the time for the new date. \UNPACKDATE
;; returns a list in the form (YR M D HR MIN SEC x x). The PROGN turns the hr and min ints. into a single 24-hr. time integer.
(SETQ ALERTTIME (if (TIMEDREMP R)
then [PROGN [SETQ DATELST (\UNPACKDATE (IPLUS (IDATE)
(TIME.UNTIL (TIMEDREMP R)
'SECONDS]
(IPLUS (ITIMES (CAR (NTH DATELST 4))
100)
(CAR (NTH DATELST 5]
else 0))
(SETQ ALERTFLG (TIMEDREMP R))
(CALCREATEREM MSG REMTIME ALERTTIME ALERTFLG M D YR DBROWSER])

```

(CALTBNULFN

```

[LAMBDA (BROWSER) ; (* MJD "22-Jun-87 14:49")
(PRINTOUT (GETPROMPTWINDOW (TB.WINDOW BROWSER)
1)
T "No reminders selected.")]

```

(CALTBSELECTEDFN

```

[LAMBDA (W) ; (* MJD "23-Feb-88 13:07")
; Makes this browser be the source for rem. copies.
(RPLACD CALCURBROWSER (CAR CALCURBROWSER))
(RPLACA CALCURBROWSER (WINDOWPROP W 'TABLEBROWSER])

```

(CALTEDITEXIT

```

[LAMBDA (ITEM MNAME BUTTON) ; (* MJD "17-Jun-87 12:38")
(COND
((EQ ITEM 'Save)
(TEDIT.QUIT (TEXTSTREAM CALTEDITWINDOW)))
((EQ ITEM 'Abort)
(TEDIT.QUIT (TEXTSTREAM CALTEDITWINDOW)
'Abort])

```

(CALTEDITSTRING

```

[LAMBDA (STRING M D YR) ; Edited 5-May-2023 21:56 by rmk
; Edited 1-Feb-2022 17:13 by rmk
; Edited 14-Oct-88 12:48 by MJD
(* T.Bigham "12-Nov-84 11:03")
;; this may not be needed in Carol. In harmony, this makes tedit put the value into the item editor without the confirmation that always pops up
;; when changes have been made without saving the file.
(PROG ((*readtable* (FIND-READTABLE "INTERLISP"))
(*package* (CL:FIND-PACKAGE "INTERLISP"))
STREAM)
(if (NOT (WINDOWP CALTEDITWINDOW))
then (SETQ CALTEDITWINDOW (CREATEW CALREMCREATEREGION "" NIL T))
(ATTACHMENU (create MENU
ITEMS _ '(Save Abort)
ITEMWIDTH _ 199
CENTERFLG _ T
MENUROWS _ 1
MENUFONT _ (FONTCREATE 'HELVETICA 12 'BOLD)
MENUBORDERSIZE _ 1
WHENSELECTEDFN _ 'CALTEDITEXIT)
CALTEDITWINDOW
'TOP
'LEFT))
(WINDOWPROP CALTEDITWINDOW 'TITLE (CONCAT "Calendar message editor for " (MKSTRING (MONTHNAME M))
" " D ", " (MKSTRING YR)))
(RETURN (EVAL.IN.TTY.PROCESS
` (PROGN [SETQ STREAM (OPENTEXTSTREAM (OPENSTRINGSTREAM
(OR ,STRING (CONCAT "Date: "
(GDATE (\PACKDATE
,YR
(SUB1 ,M)
,D 0 0 0)
(DATEFORMAT NO.TIME))
(CHARACTER 13)
"Title: >>One line<<"
(CHARACTER 13)
"Event time: >>Time<<"
(CHARACTER 13)
"Alert time: >>Time<<"

```

```

                                (CHARACTER 9)
                                "Alert: >>Yes No<<"
                                (CHARACTER 13)
                                "Duration: >>hh:mm<<"
                                (CHARACTER 13)
                                "Message: >>Any text<<"))
                                NIL NIL NIL '(QUITFN T]
(TEDIT.NEXT STREAM)
(SPAWN.MOUSE)
[SETQ RESULT (TEDIT STREAM CALTEDITWINDOW T '(QUITFN T]
(IF (EQ RESULT 'Abort)
    THEN NIL
    ELSE STREAM))
T])

```

(CALUPDATEFILE

; Edited 24-Oct-88 16:09 by MJD

[LAMBDA (FILE)

```

;; Each reminder on the file has the form:
;; (timer-value date-string hash-key title-string) TEdit-stream <CR> *start*
;; File updates work like this: The file to be updated will contain all still-valid reminders that were on it when it was loaded (this info. was cached in
;; the hash array under the file name when it was loaded or initially created), plus any new reminders that have not yet been saved (this comes from
;; CALDIRTYREMLST).
;; 9/23/88: A long-standing bug involving a break when reading in TEdit-formatted rems. is hopefully fixed. See the fns. NC.PutTextSubstance and
;; NC.GetTextSubstance in {QV}<NOTECARDS>1.3L>NCTEXTCARD for the model. A change was made to CALLOADFILE for this also. I'm still
;; not sure *what* the problem was. At the moment, the old way seems to be working here - the only change was in CALLOADFILE where we start
;; reading one byte later.

```

```

(PROG ((*readtable* (FIND-READTABLE "OLD-INTERLISP-T"))
      (*package* (CL:FIND-PACKAGE "INTERLISP")))
      FSTREAM REMSTREAM RDATA REMLIST WRITTENREMS (R# 0))
(OBTAIN.MONITORLOCK CALMONLOCK)
(WITH.MONITOR CALMONLOCK
 [OUTPUT (SETQ FSTREAM (OPENSTREAM FILE 'BOTH 'OLD/NEW)
 (printout PROMPTWINDOW T "Updating reminder file " FILE "...")
 (pushnew CALFILELST FILE)

```

;; A list of all the reminders that were in this file when it was loaded (or NIL if this is a new file to be written):

```

(SETQ REMLIST (GETPROP (CAR (MEMBER FILE CALFILELST))
                      'CONTENTS))
(PRINT (LIST '$$CALREMINDEERS CALENDARVERSION)
      FSTREAM)

```

;; The hash array contains both lists of items and items in files.

```

[MAPHASH
 CALHASH
 (FUNCTION (LAMBDA (VLIST KEY)

```

```

;; This is ugly, but we need the key of each item being written - it's the only way to tell if it's periodic. We sweep
;; through the entire hash array looking for items that are on REMLIST. This test isn't done when creating a new file
;; from scratch. In this case, there are no previously loaded rems., and so REMLIST is NIL.

```

```

(SETQ WRITTENREMS
 (APPEND
  WRITTENREMS
  (for VAL in VLIST when (OR (MEMBER VAL REMLIST)
                            (MEMBER VAL CALDIRTYREMLST))
    collect (PROGN (SETQ RDATA (fetch TIDATA of VAL))

```

;; Now put out the timer (CAR), the date-string (CADR), the hash key (KEY), and the title (CADDR):

```

(PRINT (LIST (CAR RDATA)
            (CADR RDATA)
            KEY
            (CADDR RDATA))
      FSTREAM)

```

;; Finally, write the reminder text:

```

(if [STREAMP (SETQ REMSTREAM (CAR (LAST (fetch TIDATA of VAL]
    then
      ;; Proposed fix for fmt bug
      (SETQ STARTPTR (GETFILEPTR FSTREAM))
      (* SETQ TEXTLEN (fetch (TEXTOBJ TEXTLEN) of
        (TEXTOBJ REMSTREAM)))
      (* TEDIT.PUT.PCTB (TEXTOBJ REMSTREAM) FSTREAM)
      (* SETFILEPTR FSTREAM (DIFFERENCE
        (SETQ EOFPTR (GETEOFPTR FSTREAM)) 8))
      (* SETQ STARTFORMATPTR (DWIN FSTREAM))
      (* SETFILEPTR FSTREAM (DIFFERENCE EOFPTR 8))
      (* DWOUT FSTREAM (DIFFERENCE STARTFORMATPTR
        STARTPTR))
      ; Set file ptr to eof:
      (* SETFILEPTR FSTREAM -1)

```

;; Old way:

```

(COPYCHARS (OPENSTREAM (COERCETEXTOBJ
                      (CAR (LAST (fetch TIDATA
                                of VAL))

```

```

                                'FILE)
                                'INPUT)
                                FSTREAM))
;; and the separator:
( TERPRI FSTREAM)
( PRINT '*start* FSTREAM)
( add R# 1)
;; User feedback - print N every 5:
( if (EQ (REMAINDER R# 5)
        0)
      then (PRINTOUT PROMPTWINDOW R# ",")
    VAL]

;; Wrap-up:
( PRINT 'STOP FSTREAM)
( CLOSEF FSTREAM)
( SETQ CALDIRTYREMLST NIL)

;; Make sure the entry for this file knows what rems. are on it so that the next Update of it will work right.:
( PUTPROP (CAR (MEMBER FILE CALFILELST))
          'CONTENTS WRITTENREMS)
( SETQ CALREMSLOADED T)
( OR (EQ (REMAINDER R# 5)
        0)
      (printout PROMPTWINDOW R# ",")
      (printout PROMPTWINDOW " done."))
( RELEASE.MONITORLOCK CALMONLOCK])

```

(CALUPDATEINIT

[LAMBDA NIL

; Edited 24-Oct-88 11:31 by MJD  
; Handles file update preliminaries - getting and checking name,  
; adding to known file list, then calls CALUPDATEFILE.

```

( PROG (FILE)
  (SETQ FILE (MENU (create MENU
                    ITEMS _ (APPEND CALFILELST (LIST 'Other 'Abort))
                    TITLE _ "File to update:")))
  (if (OR (NOT FILE)
          (EQ FILE 'Abort))
      then (PRINTOUT PROMPTWINDOW T "Update aborted.")
          (RETURN NIL))
      [if (EQ FILE 'Other)
          then (SETQ FILE (U-CASE (PROMPTFORWARD "File name:" NIL NIL PROMPTWINDOW)))
              (if (NOT FILE)
                  then (PRINTOUT PROMPTWINDOW T "No file given - update aborted.")
                      (RETURN NIL))

          ;; Now see if he typed in a full file name. If not, make it one, using the value of CALDEFAULTHOST&DIR:
          (if (NOT (MEMBER 'HOST (UNPACKFILENAME FILE)))
              then (SETQ FILE (PACKFILENAME 'NAME FILE 'DIRECTORY
                                           (OR CALDEFAULTHOST&DIR
                                               (SETQ CALDEFAULTHOST&DIR
                                                    (U-CASE (PROMPTFORWARD "Please enter a host & directory for
                                                                    the reminders file:" NIL NIL PROMPTWINDOW NIL
                                                                    NIL (CHARCODE EOL))

          ;; Apparently, we were passing a string filename to calupdatefile sometimes. This was causing the putprop there to croak. This should fix that:
          (SETQ FILE (MKATOM FILE))
          (if (AND (NOT (GETPROP (CAR (MEMBER FILE CALFILELST))
                              'CONTENTS))
                  (INFILEP FILE))
              then ;; If there's already a file out there with this name but we can't find it in the hash array, it hasn't been loaded - this could be
                  ;; trouble (typo in name, forgot to load, etc.) so warn user:
                  (if (MOUSECONFIRM (CONCAT FILE " already exists but hasn't been loaded into this Calendar
                                             yet." " Should I overwrite it?"))
                      then (CALUPDATEFILE FILE)
                      else (PRINTOUT PROMPTWINDOW T "File not updated.")
                          (RETURN NIL))
                  else ;; If the file's not on disk, make sure he really wants to create it:
                  (if (NOT (INFILEP FILE))
                      then (if (MOUSECONFIRM (CONCAT "Should I create " FILE "?"))
                              then (CALUPDATEFILE FILE)
                              else (PRINTOUT PROMPTWINDOW T "File not updated.")
                                  (RETURN NIL))
                      else (CALUPDATEFILE FILE) ;_ The normal case.
                  ])

```

(CALYEARICONFN

[LAMBDA (W ICON)

(\* MJD "22-Jun-87 14:40")

```

( if ICON
  then [ICONW.TITLE ICON (MONTHNAME (WINDOWPROP W 'YEAR#)
                                     ICON)

```

```

else [SETQ CALYEARICON (create TITLEDICON
                               ICON _ CALYEARICONMAP
                               TITLEREG _ '(6 26 50 9]
      (TITLEDICONW CALYEARICON (WINDOWPROP W 'YEAR#)
                               LITTLEFONT])

```

**(CALYEARINRANGE**

```

[LAMBDA (YR)
  (AND YR (ILESSP YR 2100)
         (IGREATERP YR 1700])

```

(\* MJD " 7-Jan-86 12:33" "Actual range is 3/1/1700 - 2/28/2100")

**(CIRCLETODAY**

```

[LAMBDA (CALMONTHWINDOW)

```

; Edited 16-May-90 15:51 by MJD

:: Put a circle around today. Only do this if: 1: the current month is this month, 2: the current year is this year (don't want circle around 3/12/87 if it's 3/12/86), and 3: today is different from the day already circled.

```

(PROG ([CALMONTHMENU (CAR (WINDOWPROP CALMONTHWINDOW 'MENU)
                          (CENTERFACTOR 0.62)
                          RADIUS DAYREGION)
      (OR CALHILITETODAY (RETURN NIL))
      (COND

```

; Don't if not wanted.

```

  ((AND (NEQ CALCIRCLEDAY (LISPDATEDAY (DATE)))
        (EQ (WINDOWPROP CALMONTHWINDOW 'MONTH#)
            (LISPDATEMONTH (DATE)))
        (EQ (WINDOWPROP CALMONTHWINDOW 'YEAR#)
            (LISPDATEYEAR (DATE)
              (TOTOPW CALMONTHWINDOW T)
              (SETQ DAYREGION (MDMENUITEMREGION CALCIRCLEDAY CALMONTHMENU)

```

:: Bug! The following doesn't work for circles on SPARC's:

```

(DSOPERATION 'INVERT CALMONTHWINDOW)

```

:: Can the topfn to avoid loops (drawcircle calls totopw):

```

(WINDOWPROP CALMONTHWINDOW 'TOTOPFN NIL) ; Erase the old circle, if any
[AND CALCIRCLEDAY (EQ CALCIRCLEMONTH (LISPDATEMONTH (DATE)))
  (SETQ RADIUS (TIMES (MAXMENUITEMWIDTH CALMONTHMENU)
                     CENTERFACTOR))

```

```

  (COND
    ((EQ CALHILITETODAY 'CIRCLE)
     (DRAWCIRCLE (IPLUS (CAR DAYREGION)
                       RADIUS)
                 (IPLUS (CADR DAYREGION)
                       RADIUS)
                 RADIUS 1 NIL CALMONTHWINDOW))
    ((EQ CALHILITETODAY 'BOX)
     (BITBLT NIL 0 0 CALMONTHWINDOW (CAR DAYREGION)
              (CADR DAYREGION)
              (MAXMENUITEMWIDTH CALMONTHMENU)
              (MAXMENUITEMHEIGHT CALMONTHMENU)
              'TEXTURE NIL 32800]

```

:: Then reset the circle to today, and draw a new circle:

```

(SETQ CALCIRCLEDAY (LISPDATEDAY (DATE)))
(SETQ RADIUS (TIMES (MAXMENUITEMWIDTH CALMONTHMENU)
                  CENTERFACTOR)) ; Figure out the new location:
(SETQ DAYREGION (MDMENUITEMREGION CALCIRCLEDAY CALMONTHMENU))

```

```

(COND
  ((EQ CALHILITETODAY 'CIRCLE)
   (DRAWCIRCLE (IPLUS (CAR DAYREGION)
                     RADIUS)
               (IPLUS (CADR DAYREGION)
                     RADIUS)
               RADIUS 1 NIL CALMONTHWINDOW))
  ((EQ CALHILITETODAY 'BOX)
   (BITBLT NIL 0 0 CALMONTHWINDOW (CAR DAYREGION)
          (CADR DAYREGION)
          (PLUS (MAXMENUITEMWIDTH CALMONTHMENU)
                4)
          (PLUS (MAXMENUITEMHEIGHT CALMONTHMENU)
                4)
          'TEXTURE NIL 32800)))
(DSOPERATION 'REPLACE CALMONTHWINDOW)
(WINDOWPROP CALMONTHWINDOW 'TOTOPFN 'CIRCLETODAY)
(SETQ CALCIRCLEMONTH (LISPDATEMONTH (DATE)])

```

**(CLEARDAY**

```

[LAMBDA (D CALMONTHWINDOW CALMONTHMENU)

```

(\* MJD "29-Jan-88 12:04")

; Erase the contents of this day box so it can be rewritten.

```

(PROG ((DAYREGION (MDMENUITEMREGION D CALMONTHMENU)))

```

:: Fool CIRCLETODAY into erasing the circle before clearing the box. Then we'll be OK when we redraw the circle. We have to do this since the circle overlaps into the text area and its top part would get lopped off otherwise.

```

  (if (EQ D CALCIRCLEDAY)
      then (SETQ CALCIRCLEDAY NIL)

```



```

(CIRCLETODAY CALMONTHWINDOW) ; Second const. was .08714
(if (IGREATERP (WINDOWPROP CALMONTHWINDOW 'HEIGHT)
    100)
    then (BITBLT NIL NIL NIL CALMONTHWINDOW (CAR DAYREGION)
        (IPLUS (CADDR DAYREGION)
            (TIMES (WINDOWPROP CALMONTHWINDOW 'HEIGHT)
                0.053))
        (CADDR DAYREGION)
        (SUB1 (TIMES (WINDOWPROP CALMONTHWINDOW 'HEIGHT)
            0.0868))
        'TEXTURE
        'ERASE BLACKSHADE))
    (if (EQ D CALCIRCLEDAY)
        then (SETQ CALCIRCLEDAY NIL)
            (CIRCLETODAY CALMONTHWINDOW]))

```

(CLOSEMONTH

```

[LAMBDA (W) ; (* MJD " 1-Dec-87 17:29")
  (PROG [ (M (WINDOWPROP W 'MONTH#))
        (YR (WINDOWPROP W 'YEAR#))
        [AND (NEQ CALUPDATEONSHRINKFLG 'Never)
            CALNEEDSUPDATE
            (ADD.PROCESS ' (CALUPDATEINIT)
            (SETQ CALMONTHLST (REMOVE W CALMONTHLST))
            (for B in CALDAYBROWSERS when [AND (EQ M (CAR (TB.USERDATA B)))
                (EQ YR (CADDR (TB.USERDATA B))
            do (CLOSEW (TB.WINDOW B])

```

(DAYABBR

```

[LAMBDA (D SCALE) ; (* MJD " 7-Aug-87 14:15")
  (if (GEQ SCALE 0.2)
    then (CAR (NTH '(Sun Mon Tue Wed Thu Fri Sat %)
        (ADD1 D)))
    else (CAR (NTH '(S M T W T F S %)
        (ADD1 D]))

```

(DAYNAME

```

[LAMBDA (D) ; (* MD " 2-Feb-84 17:15")
  (CAR (NTH '(Sunday Monday Tuesday Wednesday Thursday Friday Saturday %)
    (ADD1 D]))

```

(DAYOF

```

[LAMBDA (M D Y) ; (* MD " 2-Feb-84 15:39")
  (PROG (N)
    (SETQ N (FQUOTIENT (IDIFFERENCE (IPLUS [FIX (FTIMES 365.25 (COND
        ((IGREATERP M 2)
        Y)
        (T (SUB1 Y)
        [FIX (FTIMES 30.6 (COND
        ((IGREATERP M 2)
        (ADD1 M))
        (T (IPLUS M 13]
        D)
        621049)
        7))
    (RETURN (FIX (FPLUS (FTIMES (FDIFFERENCE N (FIX N))
        7)
        0.5]))

```

(DAYPLUS

```

[LAMBDA (M D YR N) ; (* MJD " 4-Jan-88 12:02")
  (if (ILEQ (IPLUS D N)
    (DAYSIN M YR))
    then (IPLUS D N)
    else (IDIFFERENCE N (IDIFFERENCE (DAYSIN M YR)
        D]))

```

(DAYSIN

```

[LAMBDA (M Y) ; Edited 2-Jul-90 09:23 by MJD
; Returns number of days in month M of year Y.

```

```

(COND
  ((EQ M 2)
    ;; K&R put it this way: "...a year is a leap year if it is divisible by 4 but not by 100, except that years divisible by 400 ARE leap years."
    (COND
      ((OR (AND (EQ (IREMAINDER Y 4)
        0)
        (NOT (EQ (IREMAINDER Y 100)
        0))))
        (EQ (IREMAINDER Y 400)
        0))

```

```

29)
(T 28))
(T (CAR (NTH ' (31 NIL 31 30 31 30 31 31 30 31 30 31)
M])

```

(DERIVENEWDATE

```

[LAMBDA (DSTRING ITEMKEY)

```

(\* MJD " 6-Jan-88 13:35")  
; Previous edit by Sybalsky

:: Taken a date string for some expired periodic reminder and returns a new date representing the next scheduled firing time for the reminder.  
:: ITEMKEY is the rem's. hash key. This is used to tell what kind of periodic rem. it is.

```

(PROG (M D YR DNEW NEWM NEWDATESTR)
  (SETQ M (LISPDATEMONTH DSTRING))
  (SETQ D (LISPDATEDAY DSTRING))
  (SETQ YR (LISPDATEYEAR DSTRING))

```

:: Start incrementing the day, month, or year, as appropriate until we create some date in the future from now:

```

(repeatwhile (LESSP (IDATE NEWDATESTR)
  (IDATE (DATE)))

```

do (COND

```

  ((EQ ITEMKEY 0) ; Daily Item
    (SETQ DNEW (DAYPLUS M D YR 1))
    (SETQ NEWM (MONTHOFDAYPLUS M D YR 1))
    (COND
      ((LESSP NEWM M) ; Ran into a new year when we bumped the month: Up the year.
        (ADD YR 1))
      (SETQ D DNEW)
      (SETQ M NEWM))
    ((ILESSP ITEMKEY 32) ; Monthly Item
      (SETQ NEWM (MONTHPLUS M 1))
      (COND
        ((LESSP NEWM M) ; Ran into a new year when we bumped the month: Up the year.
          (ADD YR 1))
        (SETQ M NEWM))
      ((GEQ ITEMKEY 32) ; Weekly Item
        (SETQ DNEW (DAYPLUS M D YR 7))
        (SETQ NEWM (MONTHOFDAYPLUS M D YR 7))
        (SETQ D DNEW)
        (COND
          ((LESSP NEWM M) ; Ran into a new year when we bumped the month: Up the year.
            (ADD YR 1))
          (SETQ M NEWM)))
      (SETQ NEWDATESTR (PACKDATE (GDATE (IDATE DSTRING)
        (DATEFORMAT NO.DATE))
        M D YR)))

```

```

(RETURN NEWDATESTR])

```

(DOREMINDER

```

[LAMBDA (REM)

```

; Edited 14-May-90 14:21 by MJD

```

(PROG ((*readtable* (FIND-READTABLE "OLD-INTERLISP-T"))
  (*package* (CL:FIND-PACKAGE "INTERLISP"))
  RSTREAM RDATESTR MSG MSTARTPOS MSGTEXT ITEM ITEMKEY DNEW R NEWTIMER REMDATE)

```

:: Sometimes a machine boots with no time set, which makes the time be '31-Dec-00' causing all pending reminders to fire at once. This prevents  
:: that - until the year 2000, I guess.

```

(if (LESSP (IDATE)
  (IDATE "31-Dec-86 00:00"))
  then (RETURN NIL))
(pop CALREMINDERS)

```

; Get rid of it before CALENDARWATCHER sees it again

:: REM is an instance of what goes on CALREMINDERS, ie. (timer-integer date-string tableitem-obj). The tableitem-obj has the form  
:: (timer-integer date-string tableitem-obj title-string text-stream hashkey). Hashkey is present only if this is a periodic reminder (we need it to  
:: figure out when the next firing time will be).

```

(SETQ RSTREAM (CAR (NTH (GETREMDEF (CAR (NTH REM 3)))
  5)))
(SETQ RDATESTR (CADR REM))
[if (LISTP RSTREAM)
  then (EVAL RSTREAM)
  else (SETQ MSG (COERCETEXTOBJ RSTREAM 'STRINGP))
  (SETQ MSTARTPOS (IPLUS (OR (STRPOS (CONCAT (CHARACTER 13)
    "Message: ")
    MSG)
    -9)
    9))
(SETFILEPTR RSTREAM MSTARTPOS)

```

:: This check is to catch rems. whose message is a lone CR (the READ causes a break otherwise):

```

(if (IGREATERP (IDIFFERENCE (NCHARS MSG)
  MSTARTPOS)
  1)
  then (SETQ MSGTEXT (READ RSTREAM *readtable*))
  (if (LISTP MSGTEXT)
    then (EVAL MSGTEXT)
    else (if (STRING-EQUAL (CL:MACHINE-TYPE)
      "sparc")

```

```

      then (CLOSEF (CREATE-PROCESS-STREAM (CONCAT "cat " CALTUNE "> /dev/audio")))
    else (PLAYTUNE CALTUNE)
  (if (EQ CALFLASHTYPE 'SCREEN)
      then (FLASHWINDOW NIL CALFLASHTIMES))
  (TEDIT.SETSEL RSTREAM 1 0)
  (TEDIT.SHOWSEL RSTREAM NIL)
  (TEDIT RSTREAM (PROG1 (CREATEW CALREMDISPLAYREGION "Reminder Display Window")
                        (if (EQ CALFLASHTYPE 'WINDOW)
                            then (FLASHWINDOW RSTREAM CALFLASHTIMES))))
    NIL
    ' (QUITFN T LEAVETTY T SEL DON'T] ; Let's see if this one is periodic:
  (SETQ ITEM (CAR (NTH REM 3)))
  (SETQ ITEMKEY (CAR (NTH (GETREMDEF (CAR (NTH REM 3)))
                             6)))
  [if (AND ITEMKEY (ILEQ ITEMKEY 38))
      then ; Yup, so figure out its next scheduled firing time and put it back
           ; on CALREMINDERS
      (SETQ REMDATE (DERIVENEWDATE RDATESTR ITEMKEY))
      (SETQ NEWTIMER (SETUPTIMER.DATE REMDATE))
      (SETQ R (LIST NEWTIMER REMDATE ITEM))
      (if CALREMINDERS
          then (MERGE (LIST R)
                      CALREMINDERS T)
          else (SETQ CALREMINDERS (LIST R)
                (if (NOT CALKEEPEXPIREDREMSFLG)
                    then (CALDELETEREM NIL REM])

```

```

(FMNWAYITEM
 [LAMBDA (W ID LABEL) ; (* MJD "22-Jul-87 12:01")
  (for I in (WINDOWPROP W 'FM.ITEMS) thereis (AND (EQ (FM.ITEMPROP I 'ID)
                                                    ID)
                                                    (EQ (FM.ITEMPROP I 'LABEL)
                                                        LABEL]))

```

```

(GETREMDEF
 [LAMBDA (ITEM) ; (* MJD "21-May-87 16:49")
  (if (EQ (TYPENAME (fetch TIDATA of ITEM))
          'TABLEITEM)
      then (fetch TIDATA of (fetch TIDATA of ITEM))
      else (fetch TIDATA of ITEM])

```

```

(INVERTGROUP
 [LAMBDA (M1 D1 YR1 M2 D2 YR2 SHADE CALMONTHMENU) ; (* MJD "9-Dec-87 10:54")
  (AND D2 (for D from D1 to D2 do (SHADEITEM (MENUITEM D CALMONTHMENU)
                                             CALMONTHMENU SHADE]))

```

```

(LISPDATEDAY
 [LAMBDA (LD) ; (* MJD "10-Jul-86 12:54")
  (SUBATOM LD (COND
              ((STREQUAL (SUBSTRING LD 1 1)
                          " ")
               2)
              (T 1))
  2])

```

```

(LISPDATEMONTH
 [LAMBDA (LD) ; (* MD "14-Feb-84 15:56")
  (MONTHNUM (SUBATOM LD 4 6))

```

```

(LISPDATEYEAR
 [LAMBDA (LD) ; (* MJD "24-Jun-87 10:55")
              ; (* Returns the year of a date in Lisp date format.
              ; eg.%. "26-Nov-86 15:30:00")
  (if (EQ (SUBATOM LD 10 10)
          '%)
      then (IPLUS 1900 (SUBATOM LD 8 9))
      else (SUBATOM LD 8 11])

```

```

(MDMENUITEMREGION
 [LAMBDA (ITEM MNAME SCALE) ; (* MJD "12-Feb-86 16:00")
  (for I in (fetch ITEMS of MNAME) until (EQ ITEM (CAR I)) do NIL
  finally (RETURN (if SCALE
                    then (for J in (MENUITEMREGION I MNAME) collect (TIMES J SCALE))
                    else (MENUITEMREGION I MNAME]))

```

```

(MENUITEM
 [LAMBDA (ITEM MNAME) ; (* MJD "25-Jun-86 12:03")
  (for I in (fetch ITEMS of MNAME) thereis (EQ ITEM (CAR I))

```

**(MENUREGIONITEM**

```
[LAMBDA (W MNAME) (* MJD "22-May-87 13:44")
  (GETMOUSESTATE)
  (for I in (fetch ITEMS of MNAME) thereis (INSIDEP (MENUITEMREGION I MNAME)
    (LASTMOUSEX W)
    (LASTMOUSEY W]))
```

**(MONTHABBR**

```
[LAMBDA (M) (* MD "15-Feb-84 12:19")
  (CAR (NTH ' (Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec)
    M))
```

**(MONTHNAME**

```
[LAMBDA (M) (* MJD "28-Jan-88 16:23")
  (CAR (NTH ' (January February March April May June July August September October November December)
    M))
```

**(MONTHNUM**

```
[LAMBDA (MNAME) (* MD "14-Feb-84 16:01")
  (LISTGET ' (Jan 1 Feb 2 Mar 3 Apr 4 May 5 Jun 6 Jul 7 Aug 8 Sep 9 Oct 10 Nov 11 Dec 12)
    MNAME)]
```

**(MONTHOFDAYPLUS**

```
[LAMBDA (M D YR N) (* MJD "23-Jun-87 16:27")
  (if (ILEQ (DAYPLUS M D YR N)
    D)
    then (MONTHPLUS M 1)
    else M)]
```

**(MONTHPLUS**

```
[LAMBDA (M N) (* MD "19-Oct-84 13:57")
  (COND
    ((ILEQ (IPLUS M N)
    0)
    (IPLUS M N 12))
    ((AND (EQ M 12)
    (IGREATERP N 0))
    1)
    (T (IREMAINDER (IPLUS M N)
    13]))
```

**(MONTHYEARPLUS**

```
[LAMBDA (M YR N) (* MD " 5-Nov-84 14:48")
  (IPLUS YR (IQUOTIENT (IPLUS M N)
    13)
  (if (ILEQ (IPLUS M N)
    0)
    then -1
    else 0))
```

**(NEWPARSETIME**

```
[LAMBDA (TSTRING) (* MJD " 9-Dec-87 11:07")
```

;;; This function converts the string TSTRING into an atom doing all the error checking to insure the time is valid. An a.m. or p.m. specifier is allowed as well as time in the 12 or 24 hour format. If the 12 hour format is allowed then the routine tries to deduce what the user meant. The global variable CALDAYSTART is an atom which represents the time that the user's day starts. Typically CALDAYSTART might be set to 900. The user's day when goes from 9:00 am to 8:59 pm. If TSTRING is 1:00 then this means 1:00 p.m. or 13:00. If TSTRING is 9:00 this translates to 9:00 am. If TSTRING is 8:00 this translates to 8:00 pm or 20:00

```
(LET* ([TempCleanedString (PACK (LDIFFERENCE (UNPACK TSTRING)
  ' (%. %: - % A P M a p m)
  (CleanedString (if (AND (NOT (STRPOS "." TSTRING 1))
    (NOT (STRPOS ":" TSTRING 1))
    (NOT (STRPOS "-" TSTRING 1))
    (NUMBERP TempCleanedString)
    (IGEQ TempCleanedString 0)
    (ILEQ TempCleanedString 23))
    then ;; handle the cases where the user says n meaning n:00
    (TIMES TempCleanedString 100)
    else TempCleanedString))
  (TwelveHours 1200)
  (TwentyFourHours (TIMES 2 TwelveHours))
  Start End Time NewTime AMBIGUOUSTIMEFLG)
  (if (NULL TSTRING)
    then 0
    elseif (NOT (NUMBERP CleanedString))
    then NIL
    else (if (OR (STRPOS "A" TSTRING 1)
```

```

      (STRPOS "a" TSTRING 1))
    then ;; am specified
      (if (AND (IGEQL CleanedString 0)
              (ILEQL CleanedString 1200))
          then CleanedString
          else (printout PROMPTWINDOW " - time greater than 12:00 plus am doesn't make sense")
              NIL)
    elseif (OR (STRPOS "P" TSTRING 1)
              (STRPOS "p" TSTRING 1))
    then ;; pm specified
      (if (AND (IGEQL CleanedString 0)
              (ILEQL CleanedString 1200))
          then (IPLUS CleanedString 1200)
          elseif (AND (IGREATERP CleanedString 1200)
                    (ILEQL CleanedString 2400))
              then CleanedString
              else (printout PROMPTWINDOW " - time greater than 23:59 doesn't make sense")
                  NIL)
    elseif (AND (IGREATERP CleanedString 1259)
              (ILEQL CleanedString 2359))
    then ;; In 24 hour mode between 12:59 and 23:59
      CleanedString
    elseif (IGEQL CleanedString 2400)
    then ;; time greater than 23:59
      (printout PROMPTWINDOW " - time greater than 23:59 doesn't make sense")
      NIL
    else (* ambiguous time)
      (SETQ AMBIGUOUSTIMEFLG T)
      (if (OR (NOT (SMALLP CALDAYSTART))
              (ILESSP CALDAYSTART 0)
              (IGREATERP CALDAYSTART 2359))
          then (printout PROMPTWINDOW "- invalid variable CALDAYSTART " CALDAYSTART T)
              NIL
          else (SETQ Time CleanedString)
              (SETQ Start CALDAYSTART)
              (if (EQ Start TwelveHours)
                  then (* Special case when we are starting at 12%:00)
                  (SETQ End TwentyFourHours)
                  else (SETQ End (IMOD (IPLUS Start TwelveHours)
                                       TwentyFourHours)))
              (SETQ NewTime (IMOD (IPLUS Time TwelveHours)
                                 TwentyFourHours))
              (if (GREATERP Start TwelveHours)
                  then (if (OR (IGEQL NewTime Start)
                              (ILESSP NewTime End))
                          then (* the time is the new time)
                          else (SETQ NewTime Time))
                  else (if (AND (IGEQL NewTime Start)
                              (ILESSP NewTime End))
                          then (* the time is the new time)
                          else (SETQ NewTime Time)))
              NewTime])

```

(NEXTMDISPLAYREGION

[LAMBDA (W H)

(\* MJD " 2-Dec-87 10:34")

; Handles tiling of month windows given the location of the  
; previous one (in CALMONTHLST) if any. Otherwise use  
; defaults.

```

(PROG (REG WWIDTH WHEIGHT WXLOC WYLOC)
  (SETQ REG (if CALMONTHLST
    then (WINDOWPROP (CAR CALMONTHLST)
      'REGION)
    else CALMONTHDEFAULTREGION))
  ;; If the month we're keying off is shrunken, find the position of the window itself, not the icon. If this isn't the case, we've got the xloc in REG:
  (SETQ WXLOC (if (AND CALMONTHLST (WINDOWPROP (CAR CALMONTHLST)
    'ICONFOR))
    then (CAR (WINDOWPROP (WINDOWPROP (CAR CALMONTHLST)
      'ICONFOR)
      'REGION))
    else (CAR REG)))
  (SETQ WYLOC (if (AND CALMONTHLST (WINDOWPROP (CAR CALMONTHLST)
    'ICONFOR))
    then (CADR (WINDOWPROP (WINDOWPROP (CAR CALMONTHLST)
      'ICONFOR)
      'REGION))
    else (CADR REG)))
  (SETQ WWIDTH (CADR REG))
  (SETQ WHEIGHT (CADDR REG))
  (RETURN (LIST (if (AND CALMONTHLST (IGREATERP (IPLUS WXLOC WWIDTH W)
    SCREENWIDTH))
    then (CAR CALMONTHDEFAULTREGION)

```

```

else (if (AND CALMONTHLST (ILEQ (IPLUS WXLOC WWIDTH W)
                                SCREENWIDTH))
         then (IPLUS WXLOC WWIDTH 1)
         else (CAR CALMONTHDEFAULTREGION)))
(if (IGREATERP (IPLUS WXLOC WWIDTH W)
            SCREENWIDTH)
    then (if (IGREATERP (IPLUS WYLOC WHEIGHT H)
                        SCREENHEIGHT)
             then (CADR CALMONTHDEFAULTREGION)
             else (IPLUS WYLOC WHEIGHT 1))
    else WYLOC)
W H])

```

**(PACKDATE**

[LAMBDA (M TIME M D YR)

; Edited 5-May-2023 22:10 by rmk  
(\* MJD "15-May-87 09:38")

(\* Takes a time, M, D, and YR, and packs them into a formatted date which is returned.)

(\* If MTIME = 0, then this is an untimed rem., so store NIL in the time field.)

```

(CONCAT (if (IGE Q D 10)
           then D
           else (CONCAT " " D))
        "-")
(MONTHABBR M)
"- " YR " " (if (EQ MTIME 0)
                then NIL
                else MTIME])

```

**(PARSETIME**

[LAMBDA (TSTRING)

(\* MJD "22-Oct-85 12:06")

```

(COND
  ([AND TSTRING (NOT (NUMBERP (PACK (LDIFFERENCE (UNPACK TSTRING)
                                                  '(% . %: - % A P M a p m)
                                                  NIL)
          [TSTRING (IPLUS (ITIMES [PACK (LDIFFERENCE (UNPACK TSTRING)
                                                  '(% . %: - % A P M a p m)
          (COND
            ([OR (AND (NUMBERP (MKATOM TSTRING))
                     (ILEQ (MKATOM TSTRING)
                           24))
              (AND (NOT (NUMBERP (MKATOM TSTRING)))
                   (NOT (MEMBER '%: (UNPACK TSTRING]
                       100)
                   (T 1)))
            (COND
              ((STRPOS "P" TSTRING 1)
               1200)
              ((STRPOS "p" TSTRING 1)
               1200)
              (T 0)
            (T 0])

```

**(PICKFONTSIZE**

[LAMBDA (W H)

(\* MJD "4-Jan-88 13:45")

```

(PROG ((KEYSIZE (MIN W H)))
  (RETURN (COND
    ((LEQ KEYSIZE 40)
     8)
    ((LEQ KEYSIZE 50)
     10)
    ((LEQ KEYSIZE 60)
     12)
    ((LEQ KEYSIZE 70)
     14)
    ((LEQ KEYSIZE 80)
     18)
    ((LEQ KEYSIZE 90)
     24)
    ((LESSP KEYSIZE 100)
     30)
    (T 36])

```

**(POM**

[LAMBDA (M D YR)

(\* MD "4-Apr-84 13:38")

```

(PROG [GOLDEN CENTURY GREGCORRECTION CLAVCORRECTION EXTRADAYS EPACT SECSFROMNEWTTHISYEAR SECSTHISMOON
      DAYINYEAR (SECSPERMIN 60)
      SECSPERHR SECSPERDAY SECSPERMOON
      (MONTHTABLE '(0 31 60 91 121 152 182 213 244 274 305 335 366]
      (SETQ SECSPERHR (ITIMES SECSPERMIN 60))
      (SETQ SECSPERDAY (ITIMES SECSPERHR 24))
      (SETQ SECSPERMOON (IPLUS (ITIMES SECSPERDAY 29)

```

```

        (ITIMES SECSPERHR 12)
        (ITIMES SECSPERMIN 44)
        3))
    (SETQ GOLDEN (ADD1 (IREMAINDER YR 19)))
    (SETQ CENTURY (ADD1 (IQUOTIENT YR 100)))
    (SETQ GREGCORRECTION (IDIFFERENCE (IQUOTIENT (ITIMES 3 CENTURY)
        4)
        12))
    (SETQ CLAVCORRECTION (IQUOTIENT (IDIFFERENCE (IDIFFERENCE CENTURY 16)
        (IQUOTIENT (IDIFFERENCE CENTURY 18)
        25))
        3))
    (SETQ EXTRADAYS (IDIFFERENCE (IDIFFERENCE (IQUOTIENT (ITIMES 5 YR)
        4)
        GREGCORRECTION)
        10))
    (SETQ EPACT (ADD1 (IREMAINDER (IPLUS (ITIMES 11 GOLDEN)
        19 CLAVCORRECTION (IMINUS GREGCORRECTION))
        30)))
    (COND
      ((OR (AND (EQ EPACT 25)
        (IGREATERP GOLDEN 11))
        (EQ EPACT 24))
        (add EPACT 1)))
    (SETQ DAYINYEAR (IPLUS (CAR (NTH MONTHTABLE M)
        D))
    [COND
      ((IGREATERP M 2)
        (COND
          ((EQ (IREMAINDER YR 4)
            0)
            (COND
              [(NEQ (IREMAINDER YR 100)
                0)
                (COND
                  ((EQ (IREMAINDER YR 400)
                    0)
                    (add DAYINYEAR 1)
                    (T (add DAYINYEAR 1)
                    (SETQ SECSFROMNEWTTHISYEAR (IPLUS (ITIMES DAYINYEAR SECSPERDAY)
                    (ITIMES EPACT SECSPERDAY)))
                    (SETQ SECSTHISMOON (IREMAINDER SECSFROMNEWTTHISYEAR SECSPERMOON))
                    (RETURN (IQUOTIENT SECSTHISMOON (IQUOTIENT SECSPERMOON 8))

```

**(POMDAYS**

[LAMBDA (M YR)

(\* MJD "13-Mar-86 15:47")

(\* PLIST is list of phase of each day. Then return list of first days of phases NM, FQ, Full, LQ in that order.)

(\* The COND is complicated because the first phase may be split between the beginning and end of the month. Since we want the first day of the phase (which might not be the first time it appears on the list) we have to check for this.)

```

(PROG (PLIST)
  (SETQ PLIST (for D from 1 to (DAYSIN M YR) collect (POM M D YR)))
  (RETURN (for D in '(0 2 4 6)
    collect (COND
      ((EQ D (CAR PLIST))
        (if (EQ D (CAR (LAST PLIST)))
          then [ADD1 (IDIFFERENCE (DAYSIN M YR)
            (COUNT (MEMBER (CAR PLIST)
            (NLEFT PLIST 15))
            else 1))
          (T (ADD1 (IDIFFERENCE (DAYSIN M YR)
            (COUNT (MEMBER D PLIST))

```

**(PRINTMONTH**

[LAMBDA (W STREAM)

; Edited 21-Aug-90 09:16 by MJD

;; Prints a month calendar on paper. Fully cut over for IP printers.

```

(PROG [CALPRINTSTREAM (M (WINDOWPROP W 'MONTH#)
  (YR (WINDOWPROP W 'YEAR#)
  (SETCURSOR WAITINGCURSOR)
  (PRINTOUT PROMPTWINDOW T "Formatting for print...")

```

;; First, bag the stupid portrait stream we got sent (thanks, BVM):

```

(LET ((RESETSTATE 'ERROR)
  (DECLARE (SPECVARS RESETSTATE))
  (DELFILE (CLOSEF STREAM)))

```

;; Now open our own landscape stream:

;; NIL used to be (PACKFILENAME 'VERSION NIL 'BODY (FULLNAME STREAM)):

```

[SETQ CALPRINTSTREAM (OPENIMAGESTREAM NIL 'INTERPRESS '(LANDSCAPE T)
[OR PBIGFONT (SETQ PBIGFONT (FONTCREATE 'HELVETICA 14 NIL 0 'INTERPRESS)
[OR PCALFONT (SETQ PCALFONT (FONTCREATE 'TIMESROMAN 24 NIL 0 'INTERPRESS)
[OR PLITTLEFONT (SETQ PLITTLEFONT (FONTCREATE 'HELVETICA 8 NIL 0 'INTERPRESS)

```

```

(DSPFONT PCALFONT CALPRINTSTREAM)
[PROG (X Y CT)
  (SETQ CT 0)
  (DSPRESET CALPRINTSTREAM)
  (MOVETO 9500 20400 CALPRINTSTREAM)
  (PRIN1 (MONTHNAME M)
    CALPRINTSTREAM)
  (PRIN1 " " CALPRINTSTREAM) ; Leaves room for 3-ring binder hole
  (PRIN1 YR CALPRINTSTREAM)
  (SETQ X 550)
  (SETQ Y 16700)
  (for I in (APPEND (for N from 1 to (DAYOF M 1 YR) collect '% )
    (for N from 1 to (DAYSIN M YR) collect N))
    do (MOVETO X Y CALPRINTSTREAM)
      (PRIN1 I CALPRINTSTREAM) ; Print day numbers
      (add X 3750)
      (add CT 1)
      (COND
        ((EQ (IREMAINDER CT 7)
          0)
          (SETQ X 600)
          (add Y -3166]
  (for X from 300 to 26800 by 3750 do (DRAWLINE X 600 X 19600 40 'PAINT CALPRINTSTREAM))
  ; Print vertical lines
  (DSPFONT PBIGFONT CALPRINTSTREAM)
  (for X from 800 to 25600 by 3750 as D from 0 to 6 do (MOVETO X 19800 CALPRINTSTREAM)
    (PRIN1 (DAYNAME D)
      CALPRINTSTREAM))
  ; Print day names
  (for Y from 600 to 19600 by 3166 do (DRAWLINE 300 Y 26550 Y 40 'PAINT CALPRINTSTREAM))
  ; Print horizontal lines
  (if CALHARDCOPYPOMFLG
    then (SHOWMOON M YR 32.0 (CAR (WINDOWPROP W 'MENU))
      CALPRINTSTREAM))
  (DSPFONT PLITTLEFONT CALPRINTSTREAM)
  (SHOWMONTHSMALL (MONTHPLUS M -1)
    (MONTHYEARPLUS M YR -1)
    19300 950 28.0 CALPRINTSTREAM)
  (SHOWMONTHSMALL (MONTHPLUS M 1)
    (MONTHYEARPLUS M YR 1)
    23100 950 28.0 CALPRINTSTREAM)
  (SHOWREMSINMONTH M YR 1 W (CAR (WINDOWPROP W 'MENU))
    CALPRINTSTREAM)
  (CLOSEF CALPRINTSTREAM)
  (PRINTOUT PROMPTWINDOW "done." T)
  (CURSOR T])

```

(REMINDERSOF

```

[LAMBDA (M D YR) ; Edited 12-Dec-88 16:37 by MJD
  ; Returns all rems. for this day.

```

;; This day's reminders are the union of one shot rems. explicitly stored on this day; yearlies, keyed by M and D with YR = 1900; weeklies (in the 32-38 range); monthlies, keyed by D; and dailies, keyed by 0 (always applies).

;; See CALDISPEVENT for period setup.

```

(SORT (APPEND (GETHASH (CALMAKEKEY M D YR)
  CALHASH)
  (GETHASH (CALMAKEKEY M D 1900)
  CALHASH)
  (GETHASH (IPLUS (DAYOF M D YR)
    32)
  CALHASH)
  (GETHASH D CALHASH)
  (GETHASH 0 CALHASH))
  'REMINDERTIMELT])

```

(REMINDERTIME

```

[LAMBDA (R) ; (* MJD "15-May-87 11:16")
  (if (CAR R)
    then (MKATOM (GDATE (IDATE (CADR R))
      (DATEFORMAT NO.DATE NO.SECONDS]))

```

(REMINDERTIMELT

```

[LAMBDA (R1 R2) ; Edited 19-Jan-89 15:15 by MJD

```

;; Returns T if R1's time is earlier than R2 (the AND clause), or if R2 is untimed (the NOT clause). (That has the effect of putting all the untimed's at the end of the list). R1 and R2 are TABLEITEM's. Their TIDATA is a list of the form (timer# date-string tableitem-obj stream-obj).

```

(OR (NOT (CAR (fetch TIDATA of R2)))
  (AND (CAR (fetch TIDATA of R1))
    (CAR (fetch TIDATA of R2))
    (ILESSP (IDATE (CADR (fetch TIDATA of R1)))
      (IDATE (CADR (fetch TIDATA of R2)))))

```

(REMSINMONTH



[LAMBDA (M YR)
(for D to (DAYSIN M YR) collect (REMINDERSOF M D YR]) (\* MJD "16-May-86 11:57")

(REPAINTMONTH
[LAMBDA (W REG)
(DSPCLIPPINGREGION (CREATEREGION 0 0 (WINDOWPROP W 'WIDTH)
(IPLUS (WINDOWPROP W 'HEIGHT)
3))
(WINDOWPROP W 'DSP))
(SHOWMONTH (LIST W (WINDOWPROP W 'MONTH#)
(WINDOWPROP W 'YEAR#])) (\* MJD "28-Jan-88 17:27")

(REPAINTYEAR
[LAMBDA (W REG)
(DSPCLIPPINGREGION (CREATEREGION 0 0 (WINDOWPROP W 'WIDTH)
(WINDOWPROP W 'HEIGHT))
(WINDOWPROP W 'DSP))
(SHOWYEAR (LIST (WINDOWPROP W 'YEAR#))
NIL NIL W]) (\* MJD "22-May-87 13:24")

(SAMEDAYAS
[LAMBDA (LD M D YR)
(COND
((AND (EQ (LISPDATE DAY LD)
D)
(EQ (LISPDATE MONTH LD)
M)
(OR (EQ (LISPDATE YEAR LD)
YR)
(EQ (LISPDATE YEAR LD)
2034))))
T)
(T NIL)]) (\* MD "12-Oct-84 14:23")

(SAMEMONTHAS
[LAMBDA (LD M YR)
(AND (EQ (LISPDATE MONTH LD)
M)
(OR (EQ (LISPDATE YEAR LD)
YR)
(EQ (LISPDATE YEAR LD)
2034)])) (\* MD "10-May-85 10:50")

(SCALEBITMAP
[LAMBDA (BITMAP FACTOR) (\* PmT "18-Mar-85 14:34")

(\* SCALES BITMAPS BY AN ARBITRARY AMOUNT OF 2 DECIMAL PLACES.
FACTOR CAN BE OF THE FOLLOWING FORMS%: I (AN INTEGER REPRESENTING A PERCENTAGE AMOUNT;
E.G. I=67 MEANS REDUCE THE X AND Y AXIS TO 67% OF THEIR ORIGINAL);
R (A REAL; E.G. R=1.3 MEANS INCREASE THE X AND Y AXIS BY A FACTOR OF 1.3);
(I X . I Y) (A DOTTED PAIR OF INTEGERS; E.G. (75 . 125) MEANS REDUCE THE X AXIS TO 75% OF ORIGINAL;
INCREASE Y TO 125% OF ORIGINAL); (R X . R Y) (A DOTTED PAIR OF REALS;
E.G. (2.3 . 0.81) MEANS 2.3 TIMES ORIGINAL X AXIS, 0.81 TIMES ORIGINAL Y))

(PROG (XFACTOR YFACTOR DELTAX DELTAY XROUND YROUND BITMAPWIDTH BITMAPHEIGHT HEIGHT-1 RASTERWIDTH BITMAPBASE
NEWBITMAP NEWHEIGHT-1 NEWBITMAPBASE NEWRASTERWIDTH ORIGBASE NEWBASE ORIGWORD NEWWORD XSTART
YSTART ENDX ENDY ONLINE)
(OR (type? BITMAP BITMAP)
(\ILLEGAL.ARG BITMAP))
(SETQ BITMAPWIDTH (fetch (BITMAP BITMAPWIDTH) of BITMAP))
(SETQ BITMAPHEIGHT (fetch (BITMAP BITMAPHEIGHT) of BITMAP))
(COND
((NUMBERP FACTOR)
(SETQ XFACTOR FACTOR)
(SETQ YFACTOR FACTOR))
((POSITIONP FACTOR)
(SETQ XFACTOR (CAR FACTOR))
(SETQ YFACTOR (CDR FACTOR)))
(T (\ILLEGAL.ARG FACTOR)))
[AND (FLOATP XFACTOR)
(SETQ XFACTOR (FIX (FTIMES XFACTOR 100))
[AND (FLOATP YFACTOR)
(SETQ YFACTOR (FIX (FTIMES YFACTOR 100))
(SETQ XFACTOR (IMIN SCREENWIDTH XFACTOR))
(SETQ YFACTOR (IMIN SCREENHEIGHT YFACTOR))
(COND
((ILESSP XFACTOR 101)
(SETQ DELTAX 100)
(SETQ XROUND (IQUOTIENT XFACTOR 2)))
(T (SETQ DELTAX XFACTOR)
(SETQ XROUND 50)))
(COND

```

((ILESSP YFACTOR 101)
 (SETQ DELTAY 100)
 (SETQ YROUND (IQUOTIENT YFACTOR 2)))
(T (SETQ DELTAY YFACTOR)
 (SETQ YROUND 50)))
(SETQ NEWBITMAP (BITMAPCREATE (IQUOTIENT (IPLUS XROUND DELTAX (ITIMES (SUB1 BITMAPWIDTH)
                                          XFACTOR)))
                              100)
              (IQUOTIENT (IPLUS YROUND DELTAY (ITIMES (SUB1 BITMAPHEIGHT)
                                                        YFACTOR)))
              100)
              1))
(* MAKE ALL VALUES QUICKLY AVAILABLE)
(SETQ HEIGHT-1 (SUB1 BITMAPHEIGHT))
(SETQ RASTERWIDTH (fetch (BITMAP BITMAPRASTERWIDTH) of BITMAP))
(SETQ BITMAPBASE (fetch (BITMAP BITMAPBASE) of BITMAP)) (* AND THE NEW BITMAP VALUES)
(SETQ NEWHEIGHT-1 (SUB1 (fetch (BITMAP BITMAPHEIGHT) of NEWBITMAP)))
(SETQ NEWRASTERWIDTH (fetch (BITMAP BITMAPRASTERWIDTH) of NEWBITMAP))
(SETQ NEWBITMAPBASE (fetch (BITMAP BITMAPBASE) of NEWBITMAP))
(* OK, CRANK IT OUT)
(* ORIGWORD AND NEWWORD ARE SORTA CACHED FOR
SPEED PURPOSES)
[for Y from 0 to HEIGHT-1
 do [SETQ ORIGBASE (\ADDBASE BITMAPBASE (ITIMES RASTERWIDTH (IDIFFERENCE HEIGHT-1 Y)
 (SETQ ONLINE NIL)
 [for X from 0 to (SUB1 BITMAPWIDTH)
 do [AND (ZEROP (IMOD X 16))
 (SETQ ORIGWORD (\GETBASE ORIGBASE (LRSH X 4)
 (* LOOK FOR STRINGS OF "ON" BITS;
 THEN TREAT AS A LINE FOR TRANSLATIONAL PURPOSES)
(COND
 [(BITTEST ORIGWORD (\WORDELT BITMASKARRAY (IMOD X 16)))
 (OR ONLINE (AND (SETQ ONLINE T)
 (SETQ XSTART X)
 (SETQ YSTART Y]
 (NULL ONLINE)
 )
 (* JUST SKIP OVER BLANKS)
(T
 (* SPELL THIS ALL OUT SO I CAN SEE WHAT'S GOIN' ON
 HERE)
 (SETQ XSTART (IQUOTIENT (IPLUS XROUND (ITIMES XSTART XFACTOR))
 100))
 (SETQ ENDY (IQUOTIENT (IPLUS (ITIMES YSTART YFACTOR)
 YROUND DELTAY)
 100))
 (SETQ YSTART (IQUOTIENT (IPLUS YROUND (ITIMES YSTART YFACTOR))
 100))
 (SETQ ENDX (IQUOTIENT (IPLUS XROUND (ITIMES (SUB1 X)
 XFACTOR))
 100))
 (for NY from YSTART to (SUB1 ENDY)
 do (SETQ NEWWORD (\GETBASE [SETQ NEWBASE (\ADDBASE NEWBITMAPBASE
 (ITIMES NEWRASTERWIDTH
 (IDIFFERENCE NEWHEIGHT-1
 NY]
 (LRSH XSTART 4)))
 (for NX from XSTART to ENDX do [AND (ZEROP (IMOD NX 16))
 (SETQ NEWWORD (\GETBASE NEWBASE
 (LRSH NX 4)
 [SETQ NEWWORD (LOGOR NEWWORD
 (\WORDELT BITMASKARRAY
 (IMOD NX 16]
 (AND (ZEROP (IMOD (ADD1 NX)
 16))
 (\PUTBASE NEWBASE (LRSH NX 4)
 NEWWORD)))
 (\PUTBASE NEWBASE (LRSH ENDX 4)
 NEWWORD))
 (SETQ ONLINE NIL]
(COND
 (ONLINE
 (* GOTTA CLEANUP AFTER THE LAST CASE)
 (* THIS IN CASE WORKING ON A LINE THAT GOES TO END
 (* GAWD! WHAT A WASTE O SPACE THIS IS.
 FIX LATER)
 (SETQ XSTART (IQUOTIENT (IPLUS XROUND (ITIMES XSTART XFACTOR))
 100))
 (SETQ ENDY (IQUOTIENT (IPLUS (ITIMES YSTART YFACTOR)
 YROUND DELTAY)
 100))
 (SETQ YSTART (IQUOTIENT (IPLUS YROUND (ITIMES YSTART YFACTOR))
 100))
 (SETQ ENDX (IQUOTIENT (IPLUS XROUND (ITIMES (SUB1 BITMAPWIDTH)
 XFACTOR))
 100))
 (for NY from YSTART to (SUB1 ENDY)
 do (SETQ NEWWORD (\GETBASE [SETQ NEWBASE (\ADDBASE NEWBITMAPBASE
 (ITIMES NEWRASTERWIDTH
 (IDIFFERENCE NEWHEIGHT-1 NY]
 (LRSH XSTART 4)))

```



```

                                                    (ITIMES (DAYOF M D YR)
                                                    DFHEIGHT))
                    else CALDAYDEFAULTXLOC
                    (IPLUS (if CALMONTHMENU
                               then (CADR (MDMENUITEMREGION D
                                           CALMONTHMENU))
                               else CALDAYDEFAULTYLOC)
                            180)
                    (CADDR CALDAYDEFAULTREGION)
                    (CADDR CALDAYDEFAULTREGION))
    (LIST 'PRINTFN (FUNCTION CALPRINTREM)
          'CLOSEFN
          (FUNCTION CALTBCLOSEFN)
          'COPYFN
          (FUNCTION CALTBCOPYFN)
          'USERDATA
          (LIST M D YR CALMONTHWINDOW)
          'TITLE
          (CONCAT "Day browser for " (MKSTRING (MONTHNAME M))
                  " " D ", " (MKSTRING YR)

(push CALDAYBROWSERS DAYBROWSER)
(SETQ CALTBITEMS (REMINDERSOF M D YR))
(for ITEM in CALTBITEMS do (TB.INSERT.ITEM DAYBROWSER ITEM))
(WINDOWPROP (TB.WINDOW DAYBROWSER)
  'TOTOPFN
  'CALTBSELECTEDFN)
(SETQ CALDISPMENU (create MENU
  ITEMS _ '( (Add CALADD "Add a new message in this day."
              (Display CALDISPLAY "Displays the contents of the selected
              reminder.")
              (Delete CALDELETE "The selected messages will be deleted
              immediately.")
              (Update CALUPDATE "Write out all reminder changes to disk
              file.")
              (SendMail CALMAIL "The selected messages will be mailed to
              the recipients of your choice.")
              (Period CALPERIOD "Makes the selected messages periodic."))
  MENUROWS _ 1
  CENTERFLG _ T
  WHENSELECTEDFN _ 'CALDISPEVENT))
(PUTMENUPROP CALDISPMENU 'BROWSER DAYBROWSER)
(ATTACHMENU CALDISPMENU (TB.WINDOW DAYBROWSER)
  'TOP
  'LEFT))
(OR (TB.WINDOW DAYBROWSER)
    (SHOULDNT "Browser window is NIL: please inform author"))
(TB.REDISPLAY.ITEMS DAYBROWSER)
(AND CALTBITEMS (TB.SELECT.ITEM DAYBROWSER (CAR CALTBITEMS)))
(TOTOPW (TB.WINDOW DAYBROWSER))
(RETURN (SETQ CALCURDAY D))

```

**(SHOWMONTH**

[LAMBDA (ITEM)

; Edited 19-Jan-89 14:36 by MJD  
; Both displays new and redisplay existing month windows.

```

(PROG ((CALLTYPE (CAR ITEM))
      (M (CADR ITEM))
      (YR (CAR (LAST ITEM))))
  MLOC CALMONTHWINDOW CALMONTHSTREAM CALMONTHMENU TOFFSETX TOFFSETY NMOFFSETX LMOFFSETX LMOFFSETY
  OOFFSETX OOFFSETY DHEIGHT DOFFSET MOFFSET MWIDTH MHEIGHT FONTUSED TEMP (WWIDTH (CADDR
                                                                                   CALMONTHDEFAULTREGION
                                                                                   ))
  (WHEIGHT (CADDR CALMONTHDEFAULTREGION)))
(LET* ((TITLETEXT (CONCAT (MKSTRING (MONTHNAME M))
                          " "
                          (MKSTRING YR)))
       (TITLETEXTWITHVERSION (CONCAT TITLETEXT "
                                     " CALENDARVERSION)))
  (if (NOT (CALYEARINRANGE YR))
      then (RETURN NIL)) ; Can it be done?
  (SETCURSOR WAITINGCURSOR)
  (SETQ CALMONTHWINDOW (WINDOWP CALLTYPE))
  (if (NOT CALMONTHWINDOW)
      then ;; Magic numbers:
        (SETQ MWIDTH (FIX (FQUOTIENT WWIDTH 7.15))) ; Menu item width
        (SETQ MHEIGHT (IQUOTIENT WHEIGHT 7)) ; Menu item height
        (SETQ MOFFSET (IQUOTIENT WHEIGHT 60)) ; Menu offset w/in window
        (SETQ DHEIGHT (FIX (FQUOTIENT WHEIGHT 1.096))) ; Height of day names
        (SETQ DOFFSET (FIX (FQUOTIENT WWIDTH 36.1667))) ; Offset in x of day names
        (SETQ OOFFSETX (FIX (FQUOTIENT WWIDTH 1.66))) ; Option item offset in x
        (SETQ OOFFSETY (FIX (FQUOTIENT WHEIGHT 12.1))) ; Was 12.069
        ; Option item offset in y

```

```

(SETQ LMOFFSETX (FIX (FQUOTIENT WWIDTH 1.39)))
; Little last mo. x offset
(SETQ LMOFFSETY (FIX (FQUOTIENT WHEIGHT 23.0)))
; Little last mo. y offset
(SETQ NMOFFSETX (FIX (FQUOTIENT WWIDTH 1.165)))
; Little next mo. x offset
(SETQ TOFFSETX (FIX (FQUOTIENT WWIDTH 3.472)))
(SETQ TOFFSETY (FIX (FQUOTIENT WHEIGHT 1.045)))

;; Do we want to redo an existing month window?
(if CALMONTHWINDOW
  then
    ;; Yes, so save the menu and delete it from the window (has the effect of clearing the window. Then the ADDMENU
    ;; below will redraw the menu items for us.)
    [SETQ CALMONTHMENU (CAR (WINDOWPROP CALMONTHWINDOW 'MENU))
    ;; CALMONTHMENU could be NIL (eg. if the window being passed in is newly created):
    (AND CALMONTHMENU (DELETEMENU CALMONTHMENU NIL CALMONTHWINDOW))
    ;; If this call is due to a window reshape, we'll have to remake the menu anyway.
    (if (OR (NEQ (WINDOWPROP CALMONTHWINDOW 'WIDTH)
                868)
            (NEQ (WINDOWPROP CALMONTHWINDOW 'HEIGHT)
                700))
      then (SETQ CALMONTHMENU NIL))
    ;; If this call is from a Next or Prev, the window passed in ITEM is there only to tell us to reuse this window --- it's
    ;; month# and menu must be changed.
    (if (NEQ (WINDOWPROP CALMONTHWINDOW 'MONTH#)
            M)
      then (WINDOWPROP CALMONTHWINDOW 'MONTH# M)
          (WINDOWPROP CALMONTHWINDOW 'YEAR# YR)
          (SETQ CALMONTHMENU NIL))
      (SETQ WWIDTH (WINDOWPROP CALMONTHWINDOW 'WIDTH))
      (SETQ WHEIGHT (WINDOWPROP CALMONTHWINDOW 'HEIGHT))
      (SETQ MWIDTH (FIX (FQUOTIENT WWIDTH 7.15)))
      (SETQ MHEIGHT (IQUOTIENT WHEIGHT 7))
      (SETQ MOFFSET (SUB1 (IQUOTIENT WHEIGHT 61)))
      ; Was 60
      (SETQ DHEIGHT (FIX (FQUOTIENT WHEIGHT 1.096)))
      (SETQ DOFFSET (FIX (FQUOTIENT WWIDTH 36.1667)))
      (SETQ OOFFSETX (FIX (FQUOTIENT WWIDTH 1.66)))
      [SETQ OOFFSETY (SUB1 (FIX (FQUOTIENT WHEIGHT 15.0)
                              ; WAS 12.069
                              ))
      (SETQ LMOFFSETX (FIX (FQUOTIENT WWIDTH 1.39)))
      (SETQ LMOFFSETY (FIX (FQUOTIENT WHEIGHT 23.0)))
      (SETQ NMOFFSETX (FIX (FQUOTIENT WWIDTH 1.165)))
      (SETQ TOFFSETX (IQUOTIENT [IMAX 1 (DIFFERENCE (WINDOWPROP CALMONTHWINDOW 'WIDTH)
                                                    (STRINGWIDTH TITLETEXT (WINDOWPROP
                                                                    CALMONTHWINDOW
                                                                    'DSP]
                                                    2)
                                                    ))
      ;; (setq toffsetx (iquotient (imax 1 (difference (windowprop calmonthwindow 'width) (stringwidth tittletext (windowprop
      ;; calmonthwindow 'dsp)))) 2))
      (* SETQ TOFFSETX (FIX (FQUOTIENT
                            (WINDOWPROP CALMONTHWINDOW
                            (QUOTE WIDTH)) 3.472)))
      (SETQ TOFFSETY (FIX (FQUOTIENT WHEIGHT 1.045)))
    else (SETQ CALMONTHWINDOW (CREATEW (NEXTDISPLAYREGION (WIDTHIFWINDOW WWIDTH)
                                                         (HEIGHTIFWINDOW WHEIGHT T))
                                     TITLETEXTWITHVERSION NIL T))
        (WINDOWPROP CALMONTHWINDOW 'HARDCOPYFN 'PRINTMONTH)
        (WINDOWPROP CALMONTHWINDOW 'CLOSEFN 'CLOSEMONTH)
        (WINDOWPROP CALMONTHWINDOW 'SHRINKFN 'SHRINKMONTH)
        (WINDOWPROP CALMONTHWINDOW 'ICONFN 'CALMONTHICONFN)
        (WINDOWPROP CALMONTHWINDOW 'TOTOPFN 'CIRLETODAY)
        (WINDOWPROP CALMONTHWINDOW 'RIGHTBUTTONFN 'CALMONTHRBF)
        (WINDOWPROP CALMONTHWINDOW 'PROCESS (FIND.PROCESS 'BACKGROUND))
        (WINDOWPROP CALMONTHWINDOW 'MINSIZE '(77 . 77))
        (WINDOWPROP CALMONTHWINDOW 'BORDER 2)
        (WINDOWPROP CALMONTHWINDOW 'MONTH# M)
        (WINDOWPROP CALMONTHWINDOW 'YEAR# YR))
    (if (ILEQ WWIDTH 100)
      then (SETQ TEMPFONT (WINDOWTITLEFONT))
          (WINDOWTITLEFONT LITTLEFONT))
      (SETQ CALMONTHSTREAM (DECODE/WINDOW/OR/DISPLAYSTREAM CALMONTHWINDOW NIL TITLETEXTWITHVERSION))
      (WINDOWPROP CALMONTHWINDOW 'TITLE (if (ILESSP (STRINGWIDTH TITLETEXTWITHVERSION DEFAULTFONT)
                                                  WWIDTH)
      then TITLETEXTWITHVERSION
      else TITLETEXT))
      ; Month name in title bar.
      (CLEARW CALMONTHWINDOW)
      (WINDOWPROP CALMONTHWINDOW 'GROUPEND NIL)
      [OR CALMONTHMENU (SETQ CALMONTHMENU
                          (create MENU
                              ITEMS _ [APPEND (for I from 1 to (DAYOF M 1 YR)
                                              collect (LIST '% '% "Does nothing."))

```

```

(for I from 1 to (DAYSIN M YR)
 collect (LIST I M "Left opens a day browser; middle
         adds a reminder" YR))
(for I from 1
 to (IDIFFERENCE 38 (IPLUS (DAYOF M 1 YR)
 (DAYSIN M YR)))
 collect (LIST '% '% "Does nothing.))
(LIST (LIST '% M " " YR 'OUCH))
(LIST (LIST '% M "Opens a menu for setting options." YR
 'OPTIONS))
(LIST (LIST '% M "Left shows last month in this window;
         middle creates a new window." YR
 'PREV))
(LIST (LIST '% M "Left shows next month in this window;
         middle creates a new window." YR
 'NEXT])

MENUCOLUMNS _ 7
MENUFONT _ (FONTCREATE 'TIMESROMAN (PICKFONTSIZE MWIDTH MHEIGHT))
ITEMHEIGHT _ (MAX MHEIGHT 10)
ITEMWIDTH _ (MAX MWIDTH 10)
MENUBORDERSIZE _ (if (GEQ WWIDTH 100)
 then 1
 else 0)
MENUOUTLINESIZE _ (if (GEQ WWIDTH 100)
 then 1
 else 0)
WHENSELECTEDFN _ 'SHOWDAY]
(ADDMENU CALMONTHMENU CALMONTHWINDOW (CONS MOFFSET MOFFSET))
(WINDOWPROP CALMONTHWINDOW 'RESHAPEFN 'REPAINTMONTH)
(WINDOWPROP CALMONTHWINDOW 'REPAINTFN 'REPAINTMONTH)
(WINDOWPROP CALMONTHWINDOW 'SCROLLFN NIL)
(WINDOWPROP CALMONTHWINDOW 'BUTTONEVENTFN 'CALMONTHBEF)
; WINDOWPROP CALMONTHWINDOW (QUOTE
; BUTTONEVENTFN) (QUOTE CALMONTHBEF)

;; Trailing blanks help erase previous name if this mo. is a display in an existing window (but this causes wrap-around problems
;; with small month window sizes):
;; FIX: Just simply clear the window, as done above. andyiii
(DSPFONT (SETQ FONTUSED (FONTCREATE 'TIMESROMAN (PICKFONTSIZE MWIDTH MHEIGHT)))
 CALMONTHSTREAM)
(SETQ TOFFSETX (IQUOTIENT (IMAX 1 (DIFFERENCE (WINDOWPROP CALMONTHWINDOW 'WIDTH)
 (STRINGWIDTH TITLETEXT FONTUSED)))
 2))
(if (GEQ (IPLUS TOFFSETX 6)
 (IPLUS DHEIGHT (FONTHEIGHT FONTUSED)))
 then ; Big month name at top
 (MOVETO TOFFSETX TOFFSETX CALMONTHSTREAM)
 (PRIN3 TITLETEXT CALMONTHSTREAM)
 ;; Can't use this 'cause we have to use PRIN3 to fix the "split text" bug:
 (CENTERPRININREGION TITLETEXT
 (CREATEREGION 0 TOFFSETX WWIDTH
 (FONTHEIGHT FONTUSED)) CALMONTHSTREAM)
 )

;; Pick font for day names across the top:
(DSPFONT (SETQ FONTUSED (if (GEQ (FQUOTIENT (WINDOWPROP CALMONTHWINDOW 'WIDTH)
 700)
 0.6)
 then BIGFONT
 else (if (GEQ (FQUOTIENT (WINDOWPROP CALMONTHWINDOW 'WIDTH)
 700)
 0.4)
 then DEFAULTFONT
 else LITTLEFONT)))
 CALMONTHWINDOW)

;; (lif| (geq toffsety (iplus dheight (fontheight fontused))) |then| (moveto toffsetx toffsety calmonthstream) (printout calmonthstream
;; titletext))
(if (GEQ WHEIGHT 100)
 then (for X from MOFFSET to WWIDTH by MWIDTH as D from 0 to 6
 do ; Day names across the top:
 (MOVETO (IPLUS X DOFFSET)
 DHEIGHT CALMONTHSTREAM)
 (PRIN1 (if (GEQ (FQUOTIENT (WINDOWPROP CALMONTHWINDOW 'WIDTH)
 868)
 0.7)
 then (DAYNAME D)
 else (DAYABBR D (FQUOTIENT (WINDOWPROP CALMONTHWINDOW 'WIDTH)
 868)))
 CALMONTHSTREAM)))
 (if (GEQ (WINDOWPROP CALMONTHWINDOW 'WIDTH)
 175)
 then (SHOWMOON M YR 1 CALMONTHMENU CALMONTHWINDOW)
 ; Phases of moon
 (DSPFONT (if (GEQ (FQUOTIENT (WINDOWPROP CALMONTHWINDOW 'HEIGHT)
 700)
 0.6)

```

```

      then DEFAULTFONT
      else LITTLEFONT)
    CALMONTHWINDOW)
(MOVETO OOFFSETX OOFFSEY CALMONTHSTREAM)
(PRINTOUT CALMONTHSTREAM (if (GEQ (FQUOTIENT (WINDOWPROP CALMONTHWINDOW 'WIDTH)
      868)
      0.6)
      then "Options"
      else (if (GEQ (FQUOTIENT (WINDOWPROP CALMONTHWINDOW 'WIDTH)
      868)
      0.2)
      then "Opt"
      else "o")))
(DSPFONT LITTLEFONT CALMONTHWINDOW)
(if (GEQ WHEIGHT 150)
  then (SHOWREMSINMONTH M YR 1 CALMONTHWINDOW CALMONTHMENU CALMONTHSTREAM))
[for DELTA in '(-1 1) as MOFFSETX in (LIST LMOFFSETX NMOFFSETX)
  do
    (if (GEQ (FQUOTIENT WHEIGHT 700)
      0.9)
      then (SHOWMONTHSMALL (MONTHPLUS M DELTA)
        (MONTHYEARPLUS M YR DELTA)
        MOFFSETX LMOFFSEY 1 CALMONTHWINDOW)
      else (MOVETO MOFFSETX OOFFSEY CALMONTHSTREAM)
        (PRINTOUT CALMONTHSTREAM
          (SUBSTRING (MONTHNAME (MONTHPLUS M DELTA))
            [SETQ TEMP (STRPOS ' (J F M A S O N D)
              (MONTHNAME (MONTHPLUS M DELTA))
            (if (GEQ (FQUOTIENT WWIDTH 868)
              0.6)
              then NIL
              else (if (GEQ (FQUOTIENT WWIDTH 868)
                0.2)
                then (IPLUS TEMP 2)
                else (IPLUS TEMP 0]
              ; Little next month
            (DSPFONT (FONTCREATE 'TIMESROMAN (PICKFONTSIZE MWIDTH MHEIGHT))
              CALMONTHWINDOW)
            (SETQ CALCIRCLEDAY NIL)
            (CIRCLETODAY CALMONTHWINDOW)
            (pushnew CALMONTHLST CALMONTHWINDOW)
            (if (ILEQ WWIDTH 100)
              then (WINDOWTITLEFONT TEMPFONT))
            (CURSOR T)
            (RETURN M])

```

(SHOWMONTHSMALL

(\* MJD " 2-Feb-88 13:09")

```

[LAMBDA (M YR XLOC YLOC SCALE WINDOW)
  (PROG [(CT 0)
    (X XLOC)
    (Y (IPLUS YLOC (TIMES 48 SCALE)
      (MOVETO (IPLUS X (TIMES SCALE 24))
        (IPLUS Y (TIMES SCALE 12))
        WINDOW)
      (PRIN1 (if (OR (NEQ (IMAGESTREAMTYPE WINDOW)
        'DISPLAY)
        (GEQ (WINDOWPROP WINDOW 'WIDTH)
          280))
          then (MONTHNAME M)
          else (MONTHABBR M))
        WINDOW)
    (for I in (APPEND (for N from 1 to (DAYOF M 1 YR) collect '% )
      (for N from 1 to (DAYSIN M YR) collect N))
    do (MOVETO X Y WINDOW)
      (PRIN1 I WINDOW)
      (add X (TIMES SCALE 16))
      (add CT 1)
      (COND
        ((EQ (IREMAINDER CT 7)
          0)
          (SETQ X XLOC)
          (add Y (TIMES SCALE -10])

```

(SHOWMOON

```

[LAMBDA (M YR SCALE CALMONTHMENU STREAM)
  ; Edited 20-Aug-90 15:39 by MJD
  ; SCALE here is 1 for screen res. Other than that, it's obsolete -
  ; should be removed. Currently only supports IP for hardcopy.
  ;; The month window is 23550 x 18996 in 300 spi. landscape printer coordinates. Each day is 3750 x 3166. The origin is a 600,300. (The printer
  ;; clause contains magic numbers that should be parameterized - sometime.)
  ;; Show each moon making sure they have the proper sense (depends on background color).
  (for P in (POMDAYS M YR) as PMAP in (if (AND (EQ (IMAGESTREAMTYPE STREAM)
    'DISPLAY)
    (VIDEOCOLOR))
    then ' (NMMAP FQMAP FMMAP LQMAP)

```

```

else ' (FMMAP LQMAP NMMAP FQMAP))
do (if (EQ (IMAGESTREAMTYPE STREAM)
' INTERPRESS)
then ;; Print on paper: the first factor in the PLUS corrects for window offset on the page; the second factor adjusts position within
;; the day box.
(\MOVETO.IP STREAM (PLUS (TIMES (DAYOF M P YR)
3750)
600 1500)
(PLUS (TIMES (IDIFFERENCE 5 (WEEKOF M P YR))
3166)
300 500))
(SHOWBITMAP.IP STREAM (EVAL PMAP)
NIL 0.5) ; Thanks, Dinh!

else ;; Write to display:
(BITBLT (if (GEQ (MIN (WINDOWPROP STREAM 'WIDTH)
(WINDOWPROP STREAM 'HEIGHT))
600)
then (EVAL PMAP)
else (SCALEBITMAP (EVAL PMAP)
(FQUOTIENT (MIN (WINDOWPROP STREAM 'WIDTH)
(WINDOWPROP STREAM 'HEIGHT))
900)))
NIL NIL STREAM (IPLUS (CAR (MDMENUITEMREGION P CALMONTHMENU SCALE))
(FQUOTIENT (WINDOWPROP STREAM 'WIDTH)
16.6))
(IPLUS (CADR (MDMENUITEMREGION P CALMONTHMENU SCALE))
(FQUOTIENT (WINDOWPROP STREAM 'HEIGHT)
350.0))
34 34 'INPUT 'INVERT])

```

(SHOWREMSINDAY

```

[LAMBDA (CALMONTHWINDOW M D YR) (* MJD "10-Aug-87 13:35")
;; This code is similar to SHOWREMSINMONTH except that it is optimized for picking out the reminders for only one particular day, rather than all
;; reminders in a month. Changes here may need to be done to SHOWREMSINMONTH also.
(PROG [(CALMONTHSTREAM (WINDOWPROP CALMONTHWINDOW 'DSP))
[CALMONTHMENU (CAR (WINDOWPROP CALMONTHWINDOW 'MENU))
(NREMS (FIX (TIMES (WINDOWPROP CALMONTHWINDOW 'HEIGHT)
0.01))
(CLEARDAY D CALMONTHWINDOW CALMONTHMENU)
(DSPFONT LITTLEFONT CALMONTHWINDOW)
(MOVETOUPPERLEFT CALMONTHSTREAM (MDMENUITEMREGION D CALMONTHMENU))
(DSPYPOSITION (IDIFFERENCE (DSPYPOSITION NIL CALMONTHSTREAM)
2)
CALMONTHSTREAM)
(SETQ DAYREGION (MDMENUITEMREGION D CALMONTHMENU))
(DSPCLIPPINGREGION DAYREGION CALMONTHSTREAM)
(for REMINDER in (REMINDERSOF M D YR) as I to NREMS do (DSPXPOSITION (CAR (MDMENUITEMREGION D
CALMONTHMENU))
CALMONTHSTREAM)
(CALPRINTREM NIL REMINDER CALMONTHSTREAM))
(DSPCLIPPINGREGION (CREATEREGION 0 0 (WINDOWPROP CALMONTHWINDOW 'WIDTH)
(WINDOWPROP CALMONTHWINDOW 'HEIGHT))
CALMONTHSTREAM)])

```

(SHOWREMSINMONTH

```

[LAMBDA (M YR SCALE CALMONTHWINDOW CALMONTHMENU STREAM) ; Edited 20-Aug-90 16:05 by MJD
;; Handles printing of all reminders in a month both for screen and on paper. Changes here may need to be done to SHOWREMSINDAY also.
;; SCALE is now obsolete (8/20/90).
(PROG [D REMLIST DAYREGION NREMS (TOSCREEN (EQ (IMAGESTREAMTYPE STREAM)
'DISPLAY]
(SETQ REMLIST (REMSINMONTH M YR))
;; Set the max. number of rems. to show in each day:
(SETQ NREMS (if TOSCREEN
then (FIX (TIMES (WINDOWPROP CALMONTHWINDOW 'HEIGHT)
0.01))
else 8))
(for REMINDER in REMLIST as D to (DAYSIN M YR) when REMINDER
do (SETQ DAYREGION (if TOSCREEN
then (MDMENUITEMREGION D CALMONTHMENU SCALE)
else (LIST (PLUS (TIMES (DAYOF M D YR)
3750)
600)
(PLUS (TIMES (IDIFFERENCE 5 (WEEKOF M D YR))
3166)
300)
3750 3166)))
(MOVETOUPPERLEFT STREAM DAYREGION)
;; Provide a little clearance off the top edge:

```



```

(DSPYPOSITION (IDIFFERENCE (DSPYPOSITION NIL STREAM)
 2)
  STREAM)
(DSPCLIPPINGREGION DAYREGION STREAM)
(for R in REMINDER as I to NREMS do (DSPXPOSITION (CAR DAYREGION)
  STREAM)
 (CALPRINTREM NIL R STREAM)))
(if TOSCREEN
 then (DSPCLIPPINGREGION (CREATEREGION 0 0 (WINDOWPROP CALMONTHWINDOW 'WIDTH)
 (WINDOWPROP CALMONTHWINDOW 'HEIGHT))
  STREAM])

```

**(SHOWYEAR**

(\* MJD "22-Jan-88 16:52")

```

[LAMBDA (ITEM MNAME BUTTON CALYEARWINDOW)
 (PROG ((YR (CAR ITEM))
 (CALLTYPE (LENGTH ITEM))
 (MHEIGHT 70)
 MLOC CALYEARSTREAM CALYEARMENU)
 [if (EQ YR 'Other)
 then (TERPRI PROMPTWINDOW)
 (SETQ YR (MKATOM (PROMPTFORWORD "Year: " NIL NIL PROMPTWINDOW NIL NIL (CHARCODE EOL))
 (COND
 [(CALYEARINRANGE YR)
 (if CALYEARWINDOW
 then (CLEARW CALYEARWINDOW)
 else [SETQ CALYEARWINDOW (CREATEW (if (NEQ CALLTYPE 1)
 then (PROGN (SETQ MLOC (GETBOXPOSITION 364 324 NIL NIL NIL
 "Please position the Year
 Window."))
 (create REGION
 LEFT _ (CAR MLOC)
 BOTTOM _ (CDR MLOC)
 WIDTH _ 364
 HEIGHT _ 324))
 else '(32 400 364 324))
 (CONCAT CALENDARVERSION " " (MKSTRING YR)
 [SETQ CALYEARSTREAM (DECODE/WINDOW/OR/DISPLAYSTREAM CALYEARWINDOW NIL (CONCAT
 CALENDARVERSION
 " "
 (MKSTRING YR)
 (WINDOWPROP CALYEARWINDOW 'ICON CALYEARICON)
 (WINDOWPROP CALYEARWINDOW 'ICONFN 'CALYEARICONFN)
 (WINDOWPROP CALYEARWINDOW 'YEAR# YR)
 (ATTACHMENU CALMAINMENU CALYEARWINDOW 'RIGHT 'TOP))
 (SETQ CALYEARMENU (create MENU
 ITEMS _ (for I from 1 to 12 collect (LIST '% I YR))
 MENCOLUMNS _ 3
 ITEMHEIGHT _ (IPLUS MHEIGHT 2)
 ITEMWIDTH _ 118
 WHENSELECTEDFN _ 'SHOWMONTH))
 (ADDMENU CALYEARMENU CALYEARWINDOW '(0 . 0))
 (WINDOWPROP CALYEARWINDOW 'RESHAPEFN 'DON'T)
 (WINDOWPROP CALYEARWINDOW 'REPAINTFN 'REPAINTYEAR)
 (WINDOWPROP CALYEARWINDOW 'SCROLLFN NIL)
 (DSPFONT DEFAULTFONT CALYEARWINDOW)
 (MOVE TO 157 294 CALYEARSTREAM)
 (PRIN1 YR CALYEARSTREAM)
 (DSPFONT LITTLEFONT CALYEARWINDOW)
 (for Y from 0 to 3 do (for X from 0 to 2 do (SHOWMONTHSMALL (IPLUS (ADD1 X)
 (ITIMES Y 3))
 YR
 (IPLUS (ITIMES X 120)
 4)
 (IPLUS (ITIMES (IDIFFERENCE 3 Y)
 MHEIGHT)
 8)
 1 CALYEARWINDOW]
 (T (printout PROMPTWINDOW T "Sorry - I can only handle years between 1700 and 2100."))

```

**(SHRINKMONTH**

(\* MJD "20-Jul-87 14:10")

```

[LAMBDA (X)
 [AND (EQ CALUPDATEONSHRINKFLG 'Shrink)
 CALNEEDSUPDATE
 (ADD.PROCESS ' (CALUPDATEINIT)
 (OR CALMONTHICON (SETQ CALMONTHICON (create TITLEDICON
 ICON _ CALMONTHICONMAP
 TITLEREG _ '(3 51 56 9))

```

**(SHRINKYEAR**

(\* MJD "19-Jun-87 12:09")

```

[LAMBDA (X)
 [OR (WINDOWPROP CALYEARWINDOW 'ICONPOSITION)
 (WINDOWPROP CALYEARWINDOW 'ICONPOSITION (GETBOXPOSITION (BITMAPWIDTH CALYEARICON)
 (BITMAPHEIGHT CALYEARICON))
 (OR CALYEARICON (SETQ CALYEARICON (create TITLEDICON

```

ICON \_ CALYEARICONMAP  
TITLEREG \_ '(6 26 50 9))

(TIMEDREMP

[LAMBDA (REM)  
(CAR REM)]

(\* MJD "30-Jun-87 16:15")

(TPLUS

[LAMBDA (TIME MINS)

; Edited 16-Dec-88 11:36 by MJD

;; Adds a time number and minute number, returning a time number. E.g. 1300 + -10 = 1250. The "1987" is just to make PACKDATE happy - the  
;; date itself is ignored.

(PACK (LDIFFERENCE (UNPACK (GDATE (PLUS (IDATE (PACKDATE TIME 7 1 1987))  
(TIMES MINS 60))  
(DATEFORMAT NO.DATE NO.SECONDS)))

' (%:))

(WEEKOF

[LAMBDA (M D YR)

; Edited 20-Aug-90 14:49 by MJD

;; First week of month is number 0.

(IQUOTIENT (IPLUS (SUB1 D)  
(DAYOF M 1 YR))  
7))

(YNCONVERT

[LAMBDA (X)

(\* MJD "22-Jul-87 12:07")

(if X  
then 'Yes  
else 'No))

(RPAQQ CALDAYICON



(RPAQQ CALMONTHICONMAP

S	M	T	W	T	F	S
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30		

(RPAQQ CALYEARICONMAP



(RPAQQ FQMAP



(RPAQQ FM MAP



(RPAQQ LQMAP



(RPAQQ NMMAP



{MEDLEY}<lispusers>CALENDAR.;1

(FILESLOAD (SYSLOAD FROM VALUEOF DIRECTORIES)  
FREEMENU TABLEBROWSER)

(PUTPROPS **CALENDAR COPYRIGHT** ("Xerox Corporation" 1985 1986 1987 1988 1989 1990))

---

FUNCTION INDEX

CALADDEVENT .....	3	CALTEDITEXIT .....	13	LISPDATEYEAR .....	19	REMSINMONTH .....	24
CALCREATEREM .....	4	CALTEDITSTRING .....	13	MDMENUITEMREGION .....	19	REPAINTMONTH .....	25
CALDELETEREM .....	5	CALUPDATEFILE .....	14	MENUITEM .....	19	REPAINTYEAR .....	25
CALDISPEVENT .....	6	CALUPDATEINIT .....	15	MENUREGIONITEM .....	20	SAMEDAYAS .....	25
CALDOPTIONS .....	7	CALYEARICONFN .....	15	MONTHABBR .....	20	SAMEMONTHAS .....	25
CALENDAR .....	7	CALYEARINRANGE .....	16	MONTHNAME .....	20	SCALEBITMAP .....	25
CALENDARWATCHER .....	8	CIRCLETODAY .....	16	MONTHNUM .....	20	SHOWDAY .....	27
CALEXTENDSEL .....	8	CLEARDAY .....	16	MONTHOFDAYPLUS .....	20	SHOWMONTH .....	28
CALLOADFILE .....	9	CLOSEMONTH .....	17	MONTHPLUS .....	20	SHOWMONTHSMALL .....	31
CALMAKEKEY .....	10	DAYABBR .....	17	MONTHYEARPLUS .....	20	SHOWMOON .....	31
CALMONTHBEF .....	10	DAYNAME .....	17	NEWPARSETIME .....	20	SHOWREMSINDAY .....	32
CALMONTHICONFN .....	11	DAYOF .....	17	NEXTMDISPLAYREGION .....	21	SHOWREMSINMONTH .....	32
CALMONTHRBF .....	11	DAYPLUS .....	17	PACKDATE .....	22	SHOWYEAR .....	33
CALOPTIONMENU .....	11	DAYSIN .....	17	PARSETIME .....	22	SHRINKMONTH .....	33
CALPEEKNEWMAIL .....	11	DERIVENEWDATE .....	18	PICKFONTSIZE .....	22	SHRINKYEAR .....	33
CALPRINTREM .....	12	DOREMINDER .....	18	POM .....	22	TIMEDREMP .....	34
CALREMDEF .....	12	FMNWAYITEM .....	19	POMDAYS .....	23	TPLUS .....	34
CALTBCLSEFN .....	12	GETREMDEF .....	19	PRINTMONTH .....	23	WEEKOF .....	34
CALTBCOPYFN .....	12	INVERTGROUP .....	19	REMINDESOFF .....	24	YNCONVERT .....	34
CALTBNULLFN .....	13	LISPDATEDAY .....	19	REMINDERTIME .....	24		
CALTBSELECTEDFN .....	13	LISPDATEMONTH .....	19	REMINDERTIMELT .....	24		

---

VARIABLE INDEX

CALALERTFLG .....	2	CALENDARVERSION .....	1	CALMONTHICON .....	3	CALUPDATEONSHRINKFLG .....	3
CALCIRCLEDAY .....	1	CALFILELST .....	2	CALMONTHICONMAP .....	34	CALWATCHMAILFLG .....	3
CALCIRCLEMONTH .....	1	CALFLASHTIMES .....	2	CALMONTHLST .....	3	CALYEARICON .....	3
CALCURBROWSER .....	2	CALFLASHTYPE .....	2	CALNEEDSUPDATE .....	3	CALYEARICONMAP .....	34
CALCURDAY .....	2	CALFONT .....	2	CALOPTIONSDESC .....	1	FMMAP .....	34
CALDAYBROWSERS .....	2	CALHARDCOPYPOMFLG .....	2	CALOPTIONSDESCLYRIC .....	2	FQMAP .....	34
CALDAYDEFAULTREGION .....	2	CALHASH .....	2	CALREMCREATEREGION .....	3	LAFITE.AFTER.GETMAIL.FN .....	2
CALDAYICON .....	34	CALHILITETODAY .....	2	CALREMDISPLAYREGION .....	3	LQMAP .....	34
CALDAYSTART .....	2	CALKEEPEXPIREDREMSFLG .....	3	CALREMINDEERS .....	3	NMMAP .....	34
CALDEFAULTALERTDELTA .....	2	CALMAINMENU .....	3	CALREMSLOADED .....	3	PBIGFONT .....	3
CALDEFAULTHOST&DIR .....	2	CALMONLOCK .....	3	CALTEDITWINDOW .....	3	PCALFONT .....	3
CALDIRTYREMLST .....	2	CALMONTHDEFAULTREGION .....	3	CALTUNE .....	3	PLITTLEFONT .....	3

---