

File created: 23-Jun-93 20:39:54 {DSK}<python>release>loops>2.0>library-src>LOTSMAP.;1

changes to: (VARS LOTSMAPCOMS)  
(FNS MAP-EQUAL)

previous date: 14-Jun-93 14:27:03 {DSK}<users>sybalsky>lots>phase-2>LOTSMAP.;21

Read Table: XCL

Package: INTERLISP

Format: XCCS

; Copyright (c) 1993 by Venue. All rights reserved.

(RPAQQ **LOTSMAPCOMS**

```
((COMS (RECORDS MAPMARK)
        (FNS MAPMARK-REMOVE MAPMARK-PLACE MAPMARK-UNPLACE MAKE-MAPMARK MAPW-DISPLAY-MARK)
        (FNS MAPMARKS-UPDATEY MAPMARKS-UPDATEX))
```

;;; This file implements map windows for LOTS.

```
(COMS ;; The map/map window itself
      (RECORDS MAP)
      (BITMAPS MAPW-ICON-IMAGE MAPW-ICON-MASK)
      (CONSTANTS (MAP-SIDE-BEARING 50)
                  (MAP-BOTTOM-BEARING 60)
                  (MAP-TOP-BEARING 30)
                  (MAP-2SIDE-BEARING 100)
                  (MAP-2BOTTOM-BEARING 90))
      (FNS MAP-CREATE)
      ;; Creation and basic window operations
      (FNS MAPW-CREATE MAPW-P MAP-EQUAL MAPW-REPAINTFN MAPW-SCROLLFN MAPW-BUTTONFN MAPW-RESHAPEFN
        MAPW-ICONCREATE)
      ;; Drawing of lat/long. marks, the map legend, etc.
      (FNS MAPW-DRAWLEGEND MAP-DRAWTICKS MAP-DRAWTICKS-LEFT MAP-DRAWTICKS-RIGHT MAP-DRAWTICKS-TOP
        MAP-DRAWTICKS-BOTTOM MAPW-PRINT-COORD)
      ;; Coordinate-display subwindow support
      ;; Generic COordinate-window functions:
      (FNS COORDW-CREATE COORDW-UPDATE COORDW-REPAINTFN)
      ;; Latitude/longitude window
      (FNS COORDW-LAT-CREATE COORDW-LAT-UPDATE COORDW-LAT-REPAINTFN)
      ;; Bit-offset window for mapmaking.
      (FNS COORDW-BIT-CREATE COORDW-BIT-UPDATE COORDW-BIT-REPAINTFN)
      ;; Functions called to ADD and DELETE items from the map
      (FNS MAPW-ADD-OBJECT MAPW-DELETE-OBJECT)
      (FNS MAPW-MAKE-LEFTMENU MAPW-MAKE-MIDDLEMENU MAPW-MENU-WHENSELECTEDFN)
      (VARS (*MAPW-LEFTBUTTON-ITEMS* '(|Lat/Long Window|))
            (*MAPW-MIDDLEBUTTON-ITEMS* '(|Bit-offset Window|))
            (*MAPW-LEFTBUTTON-MENU*)
            (*MAPW-MIDDLEBUTTON-MENU*))
      (P (MAPW-MAKE-LEFTMENU)
        (MAPW-MAKE-MIDDLEMENU)))
(COMS ;; Map references. Each reference is a map and coordinates within the map (in pixels) of the location.
      ;; You can get ranges and bearings between references.
      (RECORDS MAPREF)
      (FNS MAPREF-BEARING MAPREF-RANGE))
(COMS ;; The window for unplaced assets.
      (FNS UNPLACEDW-CREATE UNPLACEDW-BUTTONFN))
(COMS ;; Utility functions
      (FNS SECONDS)
      ;; HOLDS THE OBJECT BEING MOVED AROUND BY MOUSE ON THE MAP.
      (INITVARS (*MAPW-MOVING-OBJECT* NIL))
      ;; A LIST OF ALL THE MAPS KNOWN TO THE SYSTEM:
      (INITVARS *MAPS* NIL))
(COMS ;; Object support for mappable items.
      (BITMAPS MAPITEM-CURSOR-IMAGE GENERIC-MAP-IMAGE AMPHIB-CURSOR-IMAGE RORO-CURSOR-IMAGE
        ELCAS-CURSOR-IMAGE CRANE-CURSOR-IMAGE)
      (CLASSES |LotsMappableItem|)
      (METHODS |LotsMappableItem.BuildMarker| |LotsMappableItem.Display|
        |LotsMappableItem.DisplayUnplaced| |LotsMappableItem.HotX| |LotsMappableItem.HotY|
        |LotsMappableItem.ImageH| |LotsMappableItem.ImageW| |LotsMappableItem.SetLocation|
        |LotsMappableItem.UnDisplay| |LotsMappableItem.UnDisplayUnplaced|)))
```

(DECLARE\ : EVAL@COMPILE

(DATATYPE MAPMARK (PIXLAT PIXLONG MARKIMAGE MARKSAVE LAT LONG MARKREGION HOTSPOTX HOTSPOTY OBJECT))  
)(/DECLAREDATATYPE 'MAPMARK ' (POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER)  
;; ---field descriptor list elided by lister---  
' 20)

(DEFINEQ

**(MAPMARK-RELMOVE**(LAMBDA (WINDOW MARK DX DY MAPXOFFSET MAPYOFFSET)  
 (LET ((X (|fetch| (MAPMARK PIXLONG) |of| MARK))  
 (Y (|fetch| (MAPMARK PIXLAT) |of| MARK))  
 (MAPMARK-UNPLACE WINDOW MARK MAPXOFFSET MAPYOFFSET)  
 (MAPMARK-PLACE WINDOW MARK (+ X DX)  
 (+ Y DY)  
 MAPXOFFSET MAPYOFFSET)))

; Edited 7-Jun-93 01:16 by sybalsky:mv:envos

**(MAPMARK-PLACE**

(LAMBDA (WINDOW MARK NEWX NEWY MAPXOFFSET MAPYOFFSET)

; Edited 10-Jun-93 14:14 by bane

;; Move MARK's notion of where it is to NEWX,NEWY, and display it at that place, offset by the two MAP offsets (to allow for scrolling, etc.)

(LET ((REG (|fetch| (MAPMARK MARKREGION) |of| MARK))  
 (|replace| (MAPMARK PIXLAT) |of| MARK |with| (+ NEWY MAPYOFFSET))  
 (|replace| (MAPMARK PIXLONG) |of| MARK |with| (+ NEWX MAPXOFFSET))  
 (SETQ LEFT (|replace| (REGION LEFT) |of| REG |with| (- NEWX (|fetch| (MAPMARK HOTSPOTX) |of| MARK))))  
 (SETQ BOTTOM (|replace| (REGION BOTTOM) |of| REG |with| (- NEWY (|fetch| (MAPMARK HOTSPOTY) |of| MARK))))  
 (!  
 (|fetch| (MAPMARK OBJECT) |of| MARK)  
 (CL:IF (MAPW-P WINDOW)  
 ' |Display|  
 ' |DisplayUnplaced|)  
 WINDOW NEWX NEWY)(\* BITBLT WINDOW LEFT BOTTOM  
(|fetch| (MAPMARK MARKSAVE) |of| MARK) 0 0 NIL NIL  
(QUOTE INPUT) (QUOTE REPLACE))  
(\* BITBLT (|fetch| (MAPMARK MARKIMAGE) |of| MARK) 0 0  
WINDOW LEFT BOTTOM NIL NIL  
(QUOTE INPUT) (QUOTE PAINT))

)))

**(MAPMARK-UNPLACE**

(LAMBDA (WINDOW MARK NEWX NEWY MAPXOFFSET MAPYOFFSET)

; Edited 10-Jun-93 14:14 by bane

;; UNdisplay MARK off of WINDOW, leaving behind the saved image.

(LET\* ((REG (|fetch| (MAPMARK MARKREGION) |of| MARK))  
 (X (|fetch| (REGION LEFT) |of| REG))  
 (Y (|fetch| (REGION BOTTOM) |of| REG))  
 (!  
 (|fetch| (MAPMARK OBJECT) |of| MARK)  
 (CL:IF (MAPW-P WINDOW)  
 ' |UnDisplay|  
 ' |UnDisplayUnplaced|)  
 WINDOW  
 (+ X (|fetch| (MAPMARK HOTSPOTX) |of| MARK))  
 (+ Y (|fetch| (MAPMARK HOTSPOTY) |of| MARK)))(\* BITBLT (|fetch| (MAPMARK MARKSAVE) |of| MARK) 0 0  
WINDOW X Y NIL NIL (QUOTE INPUT)  
(QUOTE REPLACE))

)))

**(MAKE-MAPMARK**(LAMBDA (OBJECT)  
 (LET\* ((WIDTH (|\_ OBJECT |ImageW|))  
 (HEIGHT (|\_ OBJECT |ImageH|))  
 (HOTX (|\_ OBJECT |HotX|))  
 (HOTY (|\_ OBJECT |HotY|))  
 (REGION (LIST (IMINUS HOTX)  
 (IMINUS HOTY)  
 WIDTH HEIGHT)))  
 (|create| MAPMARK  
 PIXLAT \_ 0  
 PIXLONG \_ 0  
 HOTSPOTX \_ HOTX  
 HOTSPOTY \_ HOTY  
 MARKREGION \_ REGION  
 MARKSAVE \_ (|\_@  
 OBJECT : |map-save| (BITMAPCREATE WIDTH HEIGHT))  
 OBJECT \_ OBJECT)))

; Edited 10-Jun-93 19:53 by bane

**(MAPW-DISPLAY-MARK**

; Edited 10-Jun-93 16:23 by bane

```

(LAMBDA (WINDOW MARK)
  (LET ((REG (|fetch| (MAPMARK MARKREGION) |of| MARK))
        (MAPX (WINDOWPROP WINDOW 'MAPX))
        (MAPY (WINDOWPROP WINDOW 'MAPY))
        (CLP (DSPCLIPPINGREGION (WINDOWPROP WINDOW 'MAPCLIP)
                                WINDOW)))
    (|_!|
     (|fetch| (MAPMARK OBJECT) |of| MARK)
     ' |Draw| WINDOW (+ (|fetch| (MAPMARK HOTSPOTX) |of| MARK)
                        (|fetch| (REGION LEFT) |of| REG))
     (+ (|fetch| (MAPMARK HOTSPOTY) |of| MARK)
        (|fetch| (REGION BOTTOM) |of| REG)))

    (DSPCLIPPINGREGION CLP WINDOW))))

```

```

(* BITBLT (|fetch| (MAPMARK MARKIMAGE) |of| MARK) 0 0
WINDOW (|fetch| (REGION LEFT) |of| REG)
(|fetch| (REGION BOTTOM) |of| REG) NIL NIL
(QUOTE INPUT) (QUOTE PAINT))

```

(DEFINEQ

**(MAPMARKS-UPDATEY**

; Edited 4-Jun-93 19:04 by sybalsky:mv:envos

```

(LAMBDA (MARKS DY)
  (|for| MARK |in| MARKS |do| (|add| (|fetch| (REGION BOTTOM) |of| (|fetch| (MAPMARK MARKREGION) |of| MARK))
                                DY)))

```

**(MAPMARKS-UPDATEx**

```

(LAMBDA (MARKS DX)
  (|for| MARK |in| MARKS |do| (|add| (|fetch| (REGION LEFT) |of| (|fetch| (MAPMARK MARKREGION) |of| MARK))
                                DX)))

```

;;; This file implements map windows for LOTS.

;; The map/map window itself

(DECLARE\ : EVAL@COMPILE

```

(DATATYPE MAP (MAPNAME MAPSERIES MAPSCALE LATITUDE LONGITUDE IMAGE PIXPERSECLAT PIXPERSECLONG PIXPERYARD MAGDEC
              TRUEDEC OTHERMAPS LATORG LONGORG)
  MAGDEC _ 0 TRUEDEC _ 0 LATORG _ 0 LONGORG _ 0)

```

```

(/DECLAREDATATYPE 'MAP ' (POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER
                           POINTER POINTER POINTER)

```

;; ---field descriptor list elided by lister---

'28)

(RPAQQ MAPW-ICON-IMAGE



(RPAQQ MAPW-ICON-MASK



(DECLARE\ : EVAL@COMPILE

(RPAQQ MAP-SIDE-BEARING 50)

(RPAQQ MAP-BOTTOM-BEARING 60)

(RPAQQ MAP-TOP-BEARING 30)

(RPAQQ MAP-2SIDE-BEARING 100)

(RPAQQ MAP-2BOTTOM-BEARING 90)

```

(CONSTANTS (MAP-SIDE-BEARING 50)
  (MAP-BOTTOM-BEARING 60)
  (MAP-TOP-BEARING 30)
  (MAP-2SIDE-BEARING 100)
  (MAP-2BOTTOM-BEARING 90))

```

)

(DEFINEQ

**(MAP-CREATE**

(LAMBDA (NAME SERIES SCALE LAT LONG IMAGE PIX/LAT PIX/LONG PIX/YARD MAGDEC TRUEDEC OTHERMAPS)  
; Edited 9-Jun-93 15:32 by bane

;; Create a MAP from its constituent parts.

```
(|create| MAP
  MAPNAME _ NAME
  MAPSERIES _ SERIES
  MAPSCALE _ SCALE
  LATITUDE _ LAT
  LONGITUDE _ LONG
  IMAGE _ IMAGE
  PIXPERSECLAT _ PIX/LAT
  PIXPERSECLONG _ PIX/LONG
  PIXPERYARD _ PIX/YARD
  MAGDEC _ MAGDEC
  TRUEDEC _ TRUEDEC
  OTHERMAPS _ OTHERMAPS))
```

)

;; Creation and basic window operations

(DEFINEQ

**(MAPW-CREATE**

(LAMBDA (REGION MAP SCENARIO) ; Edited 10-Jun-93 23:48 by bane

;; Create a map window in REGION. The window will display the map MAP, and any mappable objects from LOTS scenario SCENARIO.

```
(LET ((WINDOW (CREATEW (AND REGION (|create| REGION |using| REGION BOTTOM _ (+ 60 (|fetch| (REGION BOTTOM)
                                                                                               |of| REGION))
                        HEIGHT _ (- (|fetch| (REGION HEIGHT) |of| REGION)
                                   60))))
      (|fetch| (MAP MAPNAME) |of| MAP)))
```

REG MAPCLIP UNPLACED)

;; Start out with the lower-left corner of the map showing in the window.

```
(WINDOWPROP WINDOW 'MAPX 0)
(WINDOWPROP WINDOW 'MAPY 0)
(WINDOWPROP WINDOW 'MAP MAP)
```

;; Connect the scenario, so the map can find it, if need be.

(WINDOWPROP WINDOW 'MAPSCENARIO SCENARIO)

;; Set up MAPCLIP, the clipping region for the actual map display within the window.

```
(SETQ REG (COPY (DSPCLIPPINGREGION NIL WINDOW)))
(|add| (|fetch| (REGION LEFT) |of| REG)
  MAP-SIDE-BEARING)
(|add| (|fetch| (REGION BOTTOM) |of| REG)
  MAP-BOTTOM-BEARING)
(|add| (|fetch| (REGION WIDTH) |of| REG)
  (IMINUS MAP-2SIDE-BEARING))
(|add| (|fetch| (REGION HEIGHT) |of| REG)
  (IMINUS MAP-2BOTTOM-BEARING))
(WINDOWPROP WINDOW 'MAPCLIP REG)
(SETQ MAPCLIP REG)
```

;; Set up the window support functions for a Map window.

```
(WINDOWPROP WINDOW 'SCROLLFN (FUNCTION MAPW-SCROLLFN))
(WINDOWPROP WINDOW 'REPAINTFN (FUNCTION MAPW-REPAINTFN))
(WINDOWPROP WINDOW 'BUTTONEVENTFN (FUNCTION MAPW-BUTTONFN))
(WINDOWPROP WINDOW 'RESHAPEFN (FUNCTION MAPW-RESHAPEFN))
(WINDOWPROP WINDOW 'ICONFN (FUNCTION MAPW-ICONCREATE))
(REDISPLAYW WINDOW)
(DSOPERATION 'PAINT WINDOW)
```

;; Set up the window for unplaced assets

```
(SETQ REG (WINDOWPROP WINDOW 'REGION))
(ATTACHWINDOW (SETQ UNPLACED (UNPLACEDW-CREATE (LIST (|fetch| (REGION LEFT) |of| REG)
                                                       (- (|fetch| (REGION BOTTOM) |of| REG)
                                                           60)
                                                       (|fetch| (REGION WIDTH) |of| REG)
                                                       60))))
```

```
WINDOW
'BOTTOM
'JUSTIFY)
```

```
(WINDOWPROP WINDOW 'MAPUNPLACED UNPLACED)
(WINDOWPROP WINDOW 'EXTENT (CREATEREGION 0 0 (FIXR (TIMES (|fetch| (REGION WIDTH) |of| REG)
                                                         (FQUOTIENT (BITMAPWIDTH (|fetch| (MAP IMAGE)
                                                                                               |of| MAP))
                                                         (|fetch| (REGION WIDTH) |of| MAPCLIP))))
  (FIXR (TIMES (|fetch| (REGION HEIGHT) |of| REG)
            (FQUOTIENT (BITMAPHEIGHT (|fetch| (MAP IMAGE) |of| MAP))
```

(|fetch| (REGION HEIGHT) |of| MAPCLIP))))))

;; Set us up a prompt window attached to this window.

(GETPROMPTWINDOW WINDOW 1)  
(WINDOW))**(MAPW-P**(LAMBDA (WINDOW)  
(WINDOWPROP WINDOW 'MAP)))**(MAP-EQUAL**

(LAMBDA (MAP1 MAP2)

; Edited 23-Jun-93 20:04 by sybalsky:mv:envos

;; Equality predicate for two maps. Returns NIL unless they represent the same region, with the same name, etc.

(AND (STRING-EQUAL (|fetch| (MAP MAPNAME) |of| MAP1)  
(|fetch| (MAP MAPNAME) |of| MAP2))  
(STRING-EQUAL (|fetch| (MAP MAPSERIES) |of| MAP1)  
(|fetch| (MAP MAPSERIES) |of| MAP2))  
(= (|fetch| (MAP MAPSCALE) |of| MAP1)  
(|fetch| (MAP MAPSCALE) |of| MAP2))  
(= (|fetch| (MAP LATITUDE) |of| MAP1)  
(|fetch| (MAP LATITUDE) |of| MAP2))  
(= (|fetch| (MAP LONGITUDE) |of| MAP1)  
(|fetch| (MAP LONGITUDE) |of| MAP2))  
(= (|fetch| (MAP PIXPERSECLAT) |of| MAP1)  
(|fetch| (MAP PIXPERSECLAT) |of| MAP2))  
(= (|fetch| (MAP PIXPERSECLONG) |of| MAP1)  
(|fetch| (MAP PIXPERSECLONG) |of| MAP2))  
(= (|fetch| (MAP PIXPERYARD) |of| MAP1)  
(|fetch| (MAP PIXPERYARD) |of| MAP2))))))**(MAPW-REPAINTFN**

(LAMBDA (WINDOW REGION)

; Edited 10-Jun-93 23:44 by bane

(LET ((L (|fetch| (REGION LEFT) |of| REGION))  
(B (|fetch| (REGION BOTTOM) |of| REGION))  
(W (|fetch| (REGION WIDTH) |of| REGION))  
(H (|fetch| (REGION HEIGHT) |of| REGION))  
(MAP (WINDOWPROP WINDOW 'MAP))  
(MAPX (WINDOWPROP WINDOW 'MAPX))  
(MAPY (WINDOWPROP WINDOW 'MAPY))  
(MAPCLIP (WINDOWPROP WINDOW 'MAPCLIP))  
(MAPMARKS (WINDOWPROP WINDOW 'MAPMARKS))  
IMAGE-L IMAGE-B IMAGE-W IMAGE-H SCALE SEC/TICK TICKS/PRINT)

;; Compute tick frequency, in (A) seconds per tick, and (B) ticks per printing of lat/long markings

(SETQ SCALE (|fetch| (MAP MAPSCALE) |of| MAP))  
(COND  
((<= 18000 SCALE 30000)  
(SETQ SEC/TICK 30)  
(SETQ TICKS/PRINT 5))  
((<= 30001 SCALE 50000)

;; SCALE 80000 NAUTICAL CHARTS &amp; DOUBLE SIZE VERSIONS big tick every 5 min.

(SETQ SEC/TICK 30)  
(SETQ TICKS/PRINT 2))  
(T (SETQ SEC/TICK 5)  
(SETQ TICKS/PRINT 2)))  
(BITBLT (|fetch| (MAP IMAGE) |of| MAP)  
MAPX MAPY WINDOW MAP-SIDE-BEARING MAP-BOTTOM-BEARING 10000 10000 'INPUT 'REPLACE NIL  
(INTERSECTREGIONS REGION (WINDOWPROP WINDOW 'MAPCLIP))  
MAPCLIP)  
(CL:WHEN (REGIONSINTERSECTP REGION (LIST 0 0 MAP-SIDE-BEARING 10000))  
(MAP-DRAWTICKS-LEFT WINDOW MAP MAPX MAPY L B W H SEC/TICK TICKS/PRINT))  
(CL:WHEN (REGIONSINTERSECTP REGION (LIST 0 0 10000 MAP-BOTTOM-BEARING))  
(MAP-DRAWTICKS-BOTTOM WINDOW MAP MAPX MAPY L B W H SEC/TICK TICKS/PRINT)  
(MAPW-DRAWLEGEND WINDOW MAP))  
(CL:WHEN (REGIONSINTERSECTP REGION (LIST (|fetch| (REGION RIGHT) |of| MAPCLIP)  
0 MAP-SIDE-BEARING 10000))  
(MAP-DRAWTICKS-RIGHT WINDOW MAP MAPX MAPY L B W H SEC/TICK TICKS/PRINT))  
(CL:WHEN (REGIONSINTERSECTP REGION (LIST 0 (|fetch| (REGION TOP) |of| MAPCLIP)  
10000 MAP-BOTTOM-BEARING))  
(MAP-DRAWTICKS-TOP WINDOW MAP MAPX MAPY L B W H SEC/TICK TICKS/PRINT))  
(|for| MARK |in| MAPMARKS |when| (REGIONSINTERSECTP MAPCLIP (|fetch| (MAPMARK MARKREGION) |of| MARK))  
|do| (MAPW-DISPLAY-MARK WINDOW MARK))))))**(MAPW-SCROLLFN**

(LAMBDA (WINDOW DX DY CONTINUOUSFLG)

; Edited 10-Jun-93 23:39 by bane

;; First, do X scrolling

(LET ((OLDMAPX (WINDOWPROP WINDOW 'MAPX))  
(OLDMAPY (WINDOWPROP WINDOW 'MAPY))  
(MAP (WINDOWPROP WINDOW 'MAP))  
(MAPCLIP (WINDOWPROP WINDOW 'MAPCLIP)))

```

(MAPMARKS (WINDOWPROP WINDOW 'MAPMARKS))
MAPX MAPY)
(COND
  ((FIXP DX)
    ;; Scroll a fixed amount
    (SETQ MAPX (IPLUS OLDMAPX (IMINUS DX)))
    (SETQ MAPX (IMAX 0 (IMIN MAPX (- (BITMAPWIDTH (|fetch| (MAP IMAGE) |of| MAP))
                                      (|fetch| (REGION WIDTH) |of| MAPCLIP)))))
    (CL:WHEN (NOT (IEQP OLDMAPX MAPX))
      (WINDOWPROP WINDOW 'MAPX MAPX)
      (MAPMARKS-UPDTEX MAPMARKS (- OLDMAPX MAPX))
      (MAPW-REPAINTFN WINDOW ` (,MAP-SIDE-BEARING 0 , (SUB1 (|fetch| (REGION WIDTH) |of| MAPCLIP))
                              1000)))
    ((FLOATP DX)
      ;; Thumb-scrolling in X
      (SETQ MAPX (FIXR (FTIMES DX (BITMAPWIDTH (|fetch| (MAP IMAGE) |of| MAP)))))
      (SETQ MAPX (IMAX 0 (IMIN MAPX (- (BITMAPWIDTH (|fetch| (MAP IMAGE) |of| MAP))
                                      (|fetch| (REGION WIDTH) |of| MAPCLIP)))))
      (CL:WHEN (NOT (IEQP OLDMAPX MAPX))
        (WINDOWPROP WINDOW 'MAPX MAPX)
        (MAPMARKS-UPDTEX MAPMARKS (- OLDMAPX MAPX))
        (MAPW-REPAINTFN WINDOW ` (,MAP-SIDE-BEARING 0 , (SUB1 (|fetch| (REGION WIDTH) |of| MAPCLIP))
                                1000))))
    ;; Now, do Y scrolling
    (COND
      ((AND (FIXP DY)
            (NOT (ZEROP DY)))
        ;; Scroll a fixed amount
        (SETQ MAPY (IPLUS OLDMAPY (IMINUS DY)))
        (SETQ MAPY (IMAX 0 (IMIN MAPY (- (BITMAPHEIGHT (|fetch| (MAP IMAGE) |of| MAP))
                                          (|fetch| (REGION HEIGHT) |of| MAPCLIP)))))
        (CL:WHEN (NOT (IEQP OLDMAPY MAPY))
          (MAPMARKS-UPDTEY MAPMARKS (- OLDMAPY MAPY))
          (WINDOWPROP WINDOW 'MAPY MAPY)
          (MAPW-REPAINTFN WINDOW ` (0 ,MAP-BOTTOM-BEARING 1000 , (|fetch| (REGION HEIGHT) |of| MAPCLIP)))))
        ((FLOATP DY)
          ;; Thumb-scrolling in Y
          (SETQ MAPY (FIXR (FTIMES (FDIFFERENCE 1.0 DY)
                                   (BITMAPHEIGHT (|fetch| (MAP IMAGE) |of| MAP)))))
          (SETQ MAPY (IMAX 0 (IMIN MAPY (- (BITMAPHEIGHT (|fetch| (MAP IMAGE) |of| MAP))
                                          (|fetch| (REGION HEIGHT) |of| MAPCLIP)))))
          (CL:WHEN (NOT (IEQP OLDMAPY MAPY))
            (WINDOWPROP WINDOW 'MAPY MAPY)
            (MAPMARKS-UPDTEY MAPMARKS (- OLDMAPY MAPY))
            (MAPW-REPAINTFN WINDOW ` (0 ,MAP-BOTTOM-BEARING 1000 , (|fetch| (REGION HEIGHT) |of| MAPCLIP)))))
          (LET ((WID (|fetch| (REGION WIDTH) |of| (WINDOWPROP WINDOW 'REGION)))
                (HGT (|fetch| (REGION HEIGHT) |of| (WINDOWPROP WINDOW 'REGION)))
                (MAPX (WINDOWPROP WINDOW 'MAPX))
                (MAPY (WINDOWPROP WINDOW 'MAPY)))
            (WINDOWPROP WINDOW 'EXTENT (CREATEREGION (FIXR (TIMES WID (FQUOTIENT (BITMAPWIDTH
                                                                              (|fetch| (MAP IMAGE)
                                                                              |of| MAP))
                                                                              (|fetch| (REGION WIDTH)
                                                                              |of| MAPCLIP))
                                                    (- 0 (FQUOTIENT MAPX
                                                                    (BITMAPWIDTH (|fetch| (MAP IMAGE)
                                                                    |of| MAP)))))
              (FIXR (TIMES HGT (FQUOTIENT (BITMAPHEIGHT (|fetch| (MAP IMAGE)
                                                                    |of| MAP))
              (|fetch| (REGION HEIGHT) |of| MAPCLIP))
              (- 0 (FQUOTIENT MAPY (BITMAPHEIGHT
              (|fetch| (MAP IMAGE)
              |of| MAP)))))
              (FIXR (TIMES WID (FQUOTIENT (BITMAPWIDTH (|fetch| (MAP IMAGE)
                                                                    |of| MAP))
              (|fetch| (REGION WIDTH) |of| MAPCLIP)))))
              (FIXR (TIMES HGT (FQUOTIENT (BITMAPHEIGHT (|fetch| (MAP IMAGE)
                                                                    |of| MAP))
              (|fetch| (REGION HEIGHT) |of| MAPCLIP)))))
            ))
          ))

```

**(MAPW-BUTTONFN**

(LAMBDA (WINDOW)

; Edited 11-Jun-93 00:06 by bane

;; BUTTONEVENTFN for map windows.

```

(LET ((X (LASTMOUSEX WINDOW))
      (Y (LASTMOUSEY WINDOW))
      (MAPX (WINDOWPROP WINDOW 'MAPX))
      (MAPY (WINDOWPROP WINDOW 'MAPY))
      (MAP (WINDOWPROP WINDOW 'MAP))
      (MAPMARKS (WINDOWPROP WINDOW 'MAPMARKS)))

```

```

(MAPCLIP (WINDOWPROP WINDOW 'MAPCLIP))
(COORDW (WINDOWPROP WINDOW 'COORDW))
(COND
  ((MOUSESTATE (NOT (OR LEFT MIDDLE)))) ; No buttons down.
)
((NOT (INSIDE? (DSPCLIPPINGREGION NIL WINDOW)
  X Y))
  (COND
    ((MOUSESTATE LEFT)
      (MAPW-MENU-WHENSELECTEDFN WINDOW (MENU (OR *MAPW-LEFTBUTTON-MENU* (MAPW-MAKE-LEFTMENU))))
    )
    ((MOUSESTATE MIDDLE)
      (MAPW-MENU-WHENSELECTEDFN WINDOW (MENU (OR *MAPW-MIDDLEBUTTON-MENU* (MAPW-MAKE-MIDDLEMENU
        ))))))
  ((MOUSESTATE LEFT) ; Left button went down
    (|while| (MOUSESTATE LEFT) |do|)
    (SETQ X (LASTMOUSEX WINDOW))
    (SETQ Y (LASTMOUSEY WINDOW))
    (CL:WHEN (INSIDE? MAPCLIP X Y)
      (AND (WINDOWPROP WINDOW 'COORDW)
        (COORDW-UPDATE (WINDOWPROP WINDOW 'COORDW)
          X Y))
      (COND
        (*MAPW-MOVING-OBJECT* ; There's a mark being moved; This mouse click puts it in place.
          (MAPMARK-PLACE WINDOW *MAPW-MOVING-OBJECT* X Y (- MAPX MAP-SIDE-BEARING)
            (- MAPY MAP-BOTTOM-BEARING))
          (WINDOWPROP WINDOW 'MAPMARKS (NCONC1 MAPMARKS *MAPW-MOVING-OBJECT*))
          (|_!|
            (|fetch| (MAPMARK OBJECT) |of| *MAPW-MOVING-OBJECT*)
            '|SetLocation|
            (|create| MAPREF
              MAP _ (WINDOWPROP WINDOW 'MAP)
              PIXLAT _ (|fetch| (MAPMARK PIXLAT) |of| *MAPW-MOVING-OBJECT*)
              PIXLONG _ (|fetch| (MAPMARK PIXLONG) |of| *MAPW-MOVING-OBJECT*))
            (SETQ *MAPW-MOVING-OBJECT* NIL)
            (CURSOR T))
          (T (SETQ MARK (|for| AMARK |in| MAPMARKS |suchthat| (INSIDE? (|fetch| (MAPMARK MARKREGION)
            |of| AMARK)
              X Y)))
            (CL:WHEN MARK
              (MAPMARK-UNPLACE WINDOW MARK (- MAPX MAP-SIDE-BEARING)
                (- MAPY MAP-BOTTOM-BEARING))
              (SETQ *MAPW-MOVING-OBJECT* MARK)
              (|_!|
                (|fetch| (MAPMARK OBJECT) |of| *MAPW-MOVING-OBJECT*)
                '|SetLocation| NIL)
                (CURSOR (@ (|fetch| (MAPMARK OBJECT) |of| MARK)
                  |::move-cursor|))
                (SETQ MAPMARKS (REMOVE MARK MAPMARKS))
                (WINDOWPROP WINDOW 'MAPMARKS MAPMARKS))))))
  ((MOUSESTATE MIDDLE) ; Middle button went down
  ))))
)))))

```

**(MAPW-RESHAPEFN**

```

(LAMBDA (WINDOW) ; Edited 7-Jun-93 13:13 by sybalsky:mv:envos
  ;; Aftermath of reshaping a map window -- fix the clipping region. ; Edited 7-Jun-93 13:12 by sybalsky:mv:envos
  (LET (REG MAPCLIP)
    (SETQ REG (COPY (DSPCLIPPINGREGION NIL WINDOW)))
    (|add| (|fetch| (REGION LEFT) |of| REG)
      MAP-SIDE-BEARING)
    (|add| (|fetch| (REGION BOTTOM) |of| REG)
      MAP-BOTTOM-BEARING)
    (|add| (|fetch| (REGION WIDTH) |of| REG)
      (MINUS MAP-2SIDE-BEARING))
    (|add| (|fetch| (REGION HEIGHT) |of| REG)
      (MINUS MAP-2BOTTOM-BEARING))
    (WINDOWPROP WINDOW 'MAPCLIP REG)
    (REDISPLAYW WINDOW)
    (WINDOW)))

```

**(MAPW-ICONCREATE**

```

(LAMBDA (W ICON ICON-POSITION) ; Edited 7-Jun-93 13:37 by sybalsky:mv:envos
  ;; Create the icon that represents this window.
  (PROG ((ICON (WINDOWPROP W 'ICON)))
    (COND
      (ICON) ; install a new icon
      (T
        (WINDOWPROP W 'ICON (TITLEDICONW (|create| TITLEDICON
          ICON _ MAPW-ICON-IMAGE
          MASK _ MAPW-ICON-MASK
          TITLEREG _ '(3 3 57 33))
        (WINDOWPROP W 'TITLE)

```

```

(FONTCREATE 'MODERN 10)
ICON-POSITION T NIL 'FILE))))))

(WINDOWPROP W 'ICON)))

)

```

;; Drawing of lat/long. marks, the map legend, etc.

```
(DEFINEQ
```

## (MAPW-DRAWLEGEND

```

(LAMBDA (WINDOW)
  (LET ((MAP (WINDOWPROP WINDOW 'MAP))
        STRING)
    (DSPFONT ' (MODERN 12)
      WINDOW)
    (SETQ STRING (CONCAT "1:" (MKSTRING (|fetch| (MAP MAPSCALE) |of| MAP))))
    (MOVETO (- (/ (|fetch| (REGION WIDTH) |of| (DSPCLIPPINGREGION NIL WINDOW))
      2)
      (/ (STRINGWIDTH STRING WINDOW)
      2)))
    10 WINDOW)
    (PRIN1 STRING WINDOW))))

```

## (MAP-DRAWTICKS

```
(LAMBDA (WINDOW MAP MAPX MAPY L B W H)
```

; Edited 4-Jun-93 14:13 by sybalsky:mv:envos

;; Draw the latitude/longitude indicia in a map window

```

(LET ((R (IPLUS L W -1))
      (TOP (IPLUS B H -1))
      (PIXLAT (|fetch| (MAP PIXPERSECLAT) |of| MAP))
      (PIXLONG (|fetch| (MAP PIXPERSECLONG) |of| MAP))
      (LATBOT (|fetch| (MAP LATITUDE) |of| MAP))
      (LONGBOT (|fetch| (MAP LONGITUDE) |of| MAP))
      (P5LAT (FIXR (FTIMES (|fetch| (MAP PIXPERSECLAT) |of| MAP)
        5)))
      (P5LONG (FIXR (FTIMES (|fetch| (MAP PIXPERSECLONG) |of| MAP)
        5)))
      FIRSTTICK LASTTICK)

```

;; Draw ticks at 5" intervals in latitude, labelling every 10"

```

(SETQ FIRSTTICK (CL:CEILING (+ MAPY B)
  P5LAT))
(SETQ LASTTICK (CL:FLOOR (+ MAPY TOP)
  P5LAT))
(BITBLT NIL NIL NIL WINDOW 0 0 MAP-SIDE-BEARING 10000 'TEXTURE 'REPLACE 0)
(|for| TICK |from| FIRSTTICK |to| LASTTICK |do| (SETQ Y (+ MAP-BOTTOM-BEARING (- (ITIMES P5LAT TICK)
  MAPY)))
  (CL:WHEN (<= B Y TOP)
    (DRAWLINE (+ L MAP-SIDE-BEARING -8)
      Y
      (+ L MAP-SIDE-BEARING)
      Y 1 'PAINT WINDOW)
    (DRAWLINE (+ R (IMINUS MAP-SIDE-BEARING)
      8)
      Y
      (- R MAP-SIDE-BEARING)
      Y 1 'PAINT WINDOW)
    (CL:WHEN (EVENP TICK)
      (MOVETO 0 Y WINDOW)
      (MAPW-PRINT-COORD WINDOW (IPLUS LATBOT
        (ITIMES TICK 5))))))
  )))

```

## (MAP-DRAWTICKS-LEFT

```
(LAMBDA (WINDOW MAP MAPX MAPY L B W H SEC/TICK TICKS/PRINT)
```

; Edited 10-Jun-93 22:49 by bane

;; Draw the latitude/longitude indicia in a map window

```

(LET ((R (IPLUS L W -1))
      (TOP (IPLUS B H -1))
      (PIXLAT (|fetch| (MAP PIXPERSECLAT) |of| MAP))
      (PIXLONG (|fetch| (MAP PIXPERSECLONG) |of| MAP))
      (LATBOT (|fetch| (MAP LATITUDE) |of| MAP))
      (LONGBOT (|fetch| (MAP LONGITUDE) |of| MAP))
      (LATORG (|fetch| (MAP LATORG) |of| MAP))
      (P5LAT (FIXR (FTIMES (|fetch| (MAP PIXPERSECLAT) |of| MAP)
        SEC/TICK)))
      (P5LONG (FIXR (FTIMES (|fetch| (MAP PIXPERSECLONG) |of| MAP)
        SEC/TICK)))
      FIRSTTICK LASTTICK Y)

```

;; Draw ticks at 15" intervals in latitude, labelling every 2'30"

```

(SETQ FIRSTTICK (CL:CEILING (- (+ MAPY B)
  MAP-BOTTOM-BEARING LATORG)
  P5LAT))

```



```

    (SETQ LASTTICK (CL:FLOOR (- (+ MAPY TOP)
                                MAP-BOTTOM-BEARING LATORG)
                                P5LAT))
    (BITBLT NIL NIL NIL WINDOW 0 0 MAP-SIDE-BEARING 10000 'TEXTURE 'REPLACE 0)
    (|for| TICK |from| FIRSTTICK |to| LASTTICK |do| (SETQ Y (+ LATORG MAP-BOTTOM-BEARING (- (ITIMES P5LAT TICK)
                                                                                          MAPY)))
              (CL:WHEN (<= B Y TOP)
                (DRAWLINE (+ L MAP-SIDE-BEARING -8)
                           Y
                           (+ L MAP-SIDE-BEARING)
                           Y 1 'PAINT WINDOW)
                (CL:WHEN (EVENP TICK TICKS/PRINT)
                  (MOVETO 0 Y WINDOW)
                  (MAPW-PRINT-COORD WINDOW (IPLUS LATBOT
                                                    (ITIMES TICK
                                                                SEC/TICK))))))
  ))))

```

**(MAP-DRAWTICKS-RIGHT**

(LAMBDA (WINDOW MAP MAPX MAPY L B W H SEC/TICK TICKS/PRINT) ; Edited 10-Jun-93 23:00 by bane

;; Draw the latitude/longitude indicia in a map window

```

(LET ((R (|fetch| (REGION RIGHT) |of| (DSPCLIPPINGREGION NIL WINDOW)))
      (TOP (IPLUS B H -1))
      (PIXLAT (|fetch| (MAP PIXPERSECLAT) |of| MAP))
      (PIXLONG (|fetch| (MAP PIXPERSECLONG) |of| MAP))
      (LATBOT (|fetch| (MAP LATITUDE) |of| MAP))
      (LONGBOT (|fetch| (MAP LONGITUDE) |of| MAP))
      (LATORG (|fetch| (MAP LATORG) |of| MAP))
      (P5LAT (FIXR (FTIMES (|fetch| (MAP PIXPERSECLAT) |of| MAP)
                           SEC/TICK)))
      (P5LONG (FIXR (FTIMES (|fetch| (MAP PIXPERSECLONG) |of| MAP)
                           SEC/TICK)))
      (FIRSTTICK LASTTICK Y))

```

;; Draw ticks at 5" intervals in latitude, labelling every 10"

```

(SETQ FIRSTTICK (CL:CEILING (- (+ MAPY B)
                                MAP-BOTTOM-BEARING LATORG)
                                P5LAT))
(SETQ LASTTICK (CL:FLOOR (- (+ MAPY TOP)
                              MAP-BOTTOM-BEARING LATORG)
                              P5LAT))
(BITBLT NIL NIL NIL WINDOW (- R MAP-SIDE-BEARING)
  MAP-BOTTOM-BEARING MAP-SIDE-BEARING 10000 'TEXTURE 'REPLACE 0)
(|for| TICK |from| FIRSTTICK |to| LASTTICK |do| (SETQ Y (+ LATORG MAP-BOTTOM-BEARING (- (ITIMES P5LAT TICK)
                                                                                          MAPY)))
          (CL:WHEN (<= B Y TOP)
            (DRAWLINE (+ R (IMINUS MAP-SIDE-BEARING)
                           8)
                       Y
                       (- R MAP-SIDE-BEARING)
                       Y 1 'PAINT WINDOW)
            (CL:WHEN (EVENP TICK TICKS/PRINT)
              (MOVETO (+ R (IMINUS MAP-SIDE-BEARING)
                           8)
                      Y WINDOW)
              (MAPW-PRINT-COORD WINDOW (IPLUS LATBOT
                                              (ITIMES TICK
                                                          SEC/TICK))))))
  ))))

```

**(MAP-DRAWTICKS-TOP**

(LAMBDA (WINDOW MAP MAPX MAPY L B W H SEC/TICK TICKS/PRINT) ; Edited 6-Jun-93 11:14 by sybalsky:mv:envos

;; Draw the latitude/longitude indicia in a map window

```

(AND NIL (LET ((R (IPLUS L W -1))
               (TOP (IPLUS B H -1))
               (PIXLAT (|fetch| (MAP PIXPERSECLAT) |of| MAP))
               (PIXLONG (|fetch| (MAP PIXPERSECLONG) |of| MAP))
               (LATBOT (|fetch| (MAP LATITUDE) |of| MAP))
               (LONGBOT (|fetch| (MAP LONGITUDE) |of| MAP))
               (P5LAT (FIXR (FTIMES (|fetch| (MAP PIXPERSECLAT) |of| MAP)
                                    SEC/TICK)))
               (P5LONG (FIXR (FTIMES (|fetch| (MAP PIXPERSECLONG) |of| MAP)
                                    SEC/TICK)))
               (FIRSTTICK LASTTICK))

```

;; Draw ticks at 5" intervals in latitude, labelling every 10"

```

(SETQ FIRSTTICK (CL:CEILING (+ MAPY B)
                              P5LAT))
(SETQ LASTTICK (CL:FLOOR (+ MAPY TOP)
                            P5LAT))
(BITBLT NIL NIL NIL WINDOW 0 0 MAP-SIDE-BEARING 10000 'TEXTURE 'REPLACE 0)
(|for| TICK |from| FIRSTTICK |to| LASTTICK
  |do| (SETQ Y (+ MAP-BOTTOM-BEARING (- (ITIMES P5LAT TICK)

```

```

MAPY)))
(CL:WHEN (<= B Y TOP)
  (DRAWLINE (+ L MAP-SIDE-BEARING -8)
    Y
    (+ L MAP-SIDE-BEARING)
    Y 1 'PAINT WINDOW)
  (DRAWLINE (+ R (IMINUS MAP-SIDE-BEARING)
    8)
    Y
    (- R MAP-SIDE-BEARING)
    Y 1 'PAINT WINDOW)
  (CL:WHEN (EVENP TICK TICKS/PRINT)
    (MOVETO 0 Y WINDOW)
    (MAPW-PRINT-COORD WINDOW (IPLUS LATBOT (ITIMES TICK SEC/TICK))))))

```

**(MAP-DRAWTICKS-BOTTOM**

(LAMBDA (WINDOW MAP MAPX MAPY L B W H SEC/TICK TICKS/PRINT) ; Edited 8-Jun-93 10:57 by sybalsky:mv:envos

;; Draw the latitude/longitude indicia across the bottom of a map window

```

(LET ((R (IPLUS L W -1))
      (TOP (IPLUS B H -1))
      (PIXLONG (|fetch| (MAP PIXPERSECLONG) |of| MAP))
      (LONGBOT (|fetch| (MAP LONGITUDE) |of| MAP))
      (LONGORG (FETCH (MAP LONGORG) OF MAP))
      (P5LONG (FIXR (FTIMES (|fetch| (MAP PIXPERSECLONG) |of| MAP)
        SEC/TICK)))
      (FIRSTTICK LASTTICK X))
  ;; Draw ticks at 5" intervals in latitude, labelling every 10"
  (SETQ FIRSTTICK (CL:CEILING (- (+ MAPX L)
    MAP-SIDE-BEARING LONGORG)
    P5LONG))
  (SETQ LASTTICK (CL:FLOOR (- (+ MAPX R)
    MAP-SIDE-BEARING LONGORG)
    P5LONG))
  (BITBLT NIL NIL NIL WINDOW 0 0 10000 MAP-BOTTOM-BEARING 'TEXTURE 'REPLACE 0)
  (|for| TICK |from| FIRSTTICK |to| LASTTICK |do| (SETQ X (+ LONGORG MAP-SIDE-BEARING (- (ITIMES P5LONG TICK)
    MAPX)))
    (CL:WHEN (<= L X R)
      (DRAWLINE X (- MAP-BOTTOM-BEARING 8)
        X MAP-BOTTOM-BEARING 1 'PAINT WINDOW)
      (CL:WHEN (EVENP TICK TICKS/PRINT)
        (MOVETO X (- MAP-BOTTOM-BEARING 8)
          WINDOW)
        (MAPW-PRINT-COORD WINDOW (IPLUS LONGBOT
          (ITIMES TICK
            SEC/TICK))
          NIL T))))))

```

**(MAPW-PRINT-COORD**

(LAMBDA (STR SEC SHOWS CENTER? FONT) ; Edited 9-Jun-93 16:02 by bane

```

(DSPFONT (OR FONT ' (MODERN 8))
  STR)
(LET ((ORIGSEC SEC)
      (SETQ SEC (IABS SEC))
      (COND
        (CENTER?
          ;; Center it UNDER the current point.
          (LET ((STRING (CL:FORMAT NIL "~D°~D'~d\" (IQUOTIENT SEC 3600)
            (IQUOTIENT (IREMAINDER SEC 3600)
              60)
            (IREMAINDER SEC 60))))
            (SELECTQ SHOWS
              (NS (SETQ STRING (CONCAT STRING (CL:IF (IGREATERP SEC 0)
                (PRIN1 "N" STR)
                (PRIN1 "S" STR)))))
              (EW (SETQ STRING (CONCAT STRING (CL:IF (IGREATERP SEC 0)
                (PRIN1 "E" STR)
                (PRIN1 "W" STR)))))
              NIL)
            (SETQ WIDTH (STRINGWIDTH STRING STR))
            (MOVETO (- (DSPXPOSITION NIL STR)
              (LRSH WIDTH 1))
              (- (DSPYPOSITION NIL STR)
                (FONTPROP (DSPFONT NIL STR)
                  'ASCENT))
              STR)
            (PRIN1 STRING STR)))
        (T
          ;; Print at the current point.
          (CL:FORMAT STR "~D°~D'~d\" (IQUOTIENT SEC 3600)
            (IQUOTIENT (IREMAINDER SEC 3600)
              60)
            (IREMAINDER SEC 60))

```

```

      (SELECTQ SHOWNS
        (NS (CL:IF (IGREATERP ORIGSEC 0)
          (PRIN1 "N" STR)
          (PRIN1 "S" STR)))
        (EW (CL:IF (IGREATERP ORIGSEC 0)
          (PRIN1 "E" STR)
          (PRIN1 "W" STR)))
        NIL)))
)

```

```
;; Coordinate-display subwindow support
```

```
;; Generic COordinate-window functions:
```

```
(DEFINEQ
```

### (COORDW-CREATE

```
(LAMBDA (MAINWINDOW TYPE)
```

```
; Edited 10-Jun-93 22:13 by bane
```

```
;; Create the coordinate-display window that goes with a map window.
```

```

  (SELECTQ TYPE
    (LATLONG (COORDW-LAT-CREATE MAINWINDOW))
    (BITS (COORDW-BIT-CREATE MAINWINDOW))
    NIL))
)

```

### (COORDW-UPDATE

```
(LAMBDA (MAINWINDOW X Y)
```

```
; Edited 10-Jun-93 22:17 by bane
```

```

  (SELECTQ (WINDOWPROP MAINWINDOW 'COORDWTYPE)
    (LATLONG (COORDW-LAT-UPDATE MAINWINDOW X Y))
    (BITS (COORDW-BIT-UPDATE MAINWINDOW X Y))
    NIL))
)

```

### (COORDW-REPAINTFN

```
(LAMBDA (WINDOW)
  NIL)
```

```
; Edited 10-Jun-93 22:04 by bane
```

```
)
```

```
;; Latitude/longitude window
```

```
(DEFINEQ
```

### (COORDW-LAT-CREATE

```
(LAMBDA (MAINWINDOW)
```

```
; Edited 10-Jun-93 22:01 by bane
```

```
;; Create the coordinate-display window that goes with a map window.
```

```

  (LET* ((FONT (FONTCREATE 'MODERN 10))
    (BOLD (FONTCREATE 'MODERN 10 'BOLD))
    (WIDTH (+ (STRINGWIDTH "Lat " BOLD)
      (STRINGWIDTH "180o99'99\" " FONT)
      (STRINGWIDTH "Long: " BOLD)
      (STRINGWIDTH "180o99'99\" " FONT)))
    (HEIGHT (FONTPROP BOLD 'HEIGHT))
    WINDOW BOT)
    (SETQ WINDOW (CREATEW (LIST 0 0 (WIDTHIFWINDOW WIDTH)
      (HEIGHTIFWINDOW HEIGHT))
      NIL NIL T))
    (WINDOWPROP WINDOW 'COORDWTYPE 'LATLONG)
    (ATTACHWINDOW WINDOW MAINWINDOW 'TOP 'LEFT)
    (WINDOWPROP WINDOW 'LATBOT (SETQ BOT (FONTPROP BOLD 'DESCENT)))
    (WINDOWPROP WINDOW 'LATLEFT (STRINGWIDTH "Lat " BOLD))
    (WINDOWPROP WINDOW 'LONLEFT (+ (STRINGWIDTH "Lat " BOLD)
      (STRINGWIDTH "180o99'99\" " FONT)
      (STRINGWIDTH "Long: " BOLD)))
    (DSPFONT BOLD WINDOW)
    (MOVETO 0 BOT WINDOW)
    (PRIN1 "Lat: " WINDOW)
    (MOVETO (+ (STRINGWIDTH "Lat " BOLD)
      (STRINGWIDTH "180o99'99\" " FONT))
      BOT WINDOW)
    (PRIN1 "Long: " WINDOW)
    (DSPFONT FONT WINDOW)
    WINDOW))
)

```

### (COORDW-LAT-UPDATE

```
(LAMBDA (WINDOW X Y)
```

```
; Edited 10-Jun-93 22:22 by bane
```

```

  (LET* ((LATBOT (WINDOWPROP WINDOW 'LATBOT))
    (LATLEFT (WINDOWPROP WINDOW 'LATLEFT))
    (LONLEFT (WINDOWPROP WINDOW 'LONLEFT))
    (MAINWINDOW (WINDOWPROP WINDOW 'MAINWINDOW))
    (MAP (WINDOWPROP MAINWINDOW 'MAP))
    (MAPX (WINDOWPROP MAINWINDOW 'MAPX))
    (MAPY (WINDOWPROP MAINWINDOW 'MAPY))

```

```

(LAT (+ (|fetch| (MAP LATITUDE) |of| MAP)
  (/ (- (+ MAPY Y)
    MAP-BOTTOM-BEARING
    (|fetch| (MAP LATORG) |of| MAP))
    (|fetch| (MAP PIXPERSECLAT) |of| MAP))))
(LONG (+ (|fetch| (MAP LONGITUDE) |of| MAP)
  (/ (- (+ MAPX X)
    MAP-SIDE-BEARING
    (|fetch| (MAP LONGORG) |of| MAP))
    (|fetch| (MAP PIXPERSECLONG) |of| MAP))))
(BITBLT NIL NIL NIL WINDOW LATLEFT LATBOT (LOADTIMECONSTANT (STRINGWIDTH "180o99'99\" "
  (FONTCREATE 'MODERN 10)))
  100
  'TEXTURE
  'REPLACE WHITESHADE)
(BITBLT NIL NIL NIL WINDOW LONGLEFT LATBOT (LOADTIMECONSTANT (STRINGWIDTH "180o99'99\" "
  (FONTCREATE 'MODERN 10)))
  100
  'TEXTURE
  'REPLACE WHITESHADE)
(MOVETO LATLEFT LATBOT WINDOW)
(MAPW-PRINT-COORD WINDOW LAT NIL NIL ' (MODERN 10))
(MOVETO LONGLEFT LATBOT WINDOW)
(MAPW-PRINT-COORD WINDOW LONG NIL NIL ' (MODERN 10))))

```

**(COORDW-LAT-REPAINTFN**

```

(LAMBDA (WINDOW)
  ASDF))

```

```

)

```

```

;; Bit-offset window for mapmaking.

```

```

(DEFINEQ

```

**(COORDW-BIT-CREATE**

```

(LAMBDA (MAINWINDOW)

```

```

; Edited 10-Jun-93 22:18 by bane

```

```

;; Create the coordinate-display window that goes with a map window.

```

```

(LET* ((FONT (FONTCREATE 'MODERN 10))
  (BOLD (FONTCREATE 'MODERN 10 'BOLD))
  (WIDTH (+ (STRINGWIDTH "X: " BOLD)
    (STRINGWIDTH "00000 " FONT)
    (STRINGWIDTH "Y: " BOLD)
    (STRINGWIDTH "00000 " FONT)))
  (HEIGHT (FONTPROP BOLD 'HEIGHT))
  WINDOW BOT)
  (SETQ WINDOW (CREATEW (LIST 0 0 (WIDTHIFWINDOW WIDTH)
    (HEIGHTIFWINDOW HEIGHT))
    NIL NIL T))
  (WINDOWPROP WINDOW 'COORDWTYPE 'BITS)
  (ATTACHWINDOW WINDOW MAINWINDOW 'TOP 'LEFT)
  (WINDOWPROP WINDOW 'LATBOT (SETQ BOT (FONTPROP BOLD 'DESCENT)))
  (WINDOWPROP WINDOW 'LONGLEFT (STRINGWIDTH "X: " BOLD))
  (WINDOWPROP WINDOW 'LATLEFT (+ (STRINGWIDTH "X: " BOLD)
    (STRINGWIDTH "00000 " FONT)
    (STRINGWIDTH "Y: " BOLD)))
  (DSPFONT BOLD WINDOW)
  (MOVETO 0 BOT WINDOW)
  (PRIN1 "X: " WINDOW)
  (MOVETO (+ (STRINGWIDTH "X: " BOLD)
    (STRINGWIDTH "00000 " FONT))
    BOT WINDOW)
  (PRIN1 "Y: " WINDOW)
  (DSPFONT FONT WINDOW)
  WINDOW)))

```

**(COORDW-BIT-UPDATE**

```

(LAMBDA (WINDOW X Y)

```

```

; Edited 10-Jun-93 22:22 by bane

```

```

(LET* ((LATBOT (WINDOWPROP WINDOW 'LATBOT))
  (LATLEFT (WINDOWPROP WINDOW 'LATLEFT))
  (LONGLEFT (WINDOWPROP WINDOW 'LONGLEFT))
  (MAINWINDOW (WINDOWPROP WINDOW 'MAINWINDOW))
  (MAP (WINDOWPROP MAINWINDOW 'MAP))
  (MAPX (WINDOWPROP MAINWINDOW 'MAPX))
  (MAPY (WINDOWPROP MAINWINDOW 'MAPY))
  (LAT (- (+ MAPY Y)
    MAP-BOTTOM-BEARING))
  (LONG (- (+ MAPX X)
    MAP-SIDE-BEARING)))
  (BITBLT NIL NIL NIL WINDOW LATLEFT LATBOT (LOADTIMECONSTANT (STRINGWIDTH "00000 " (FONTCREATE
    'MODERN 10)))
    100
    'TEXTURE
    'REPLACE WHITESHADE)

```

```

        (BITBLT NIL NIL NIL WINDOW LONGLEFT LATBOT (LOADTIMECONSTANT (STRINGWIDTH "00000 "
                                                                                   (FONTCREATE 'MODERN 10)))
          100
          'TEXTURE
          'REPLACE WHITESHADE)
(MOVETO LATLEFT LATBOT WINDOW)
(PRIN1 LAT WINDOW)
(MOVETO LONGLEFT LATBOT WINDOW)
(PRIN1 LONG WINDOW)))

```

**(COORDW-BIT-REPAINTFN**

```

(LAMBDA (WINDOW)
  ASDF))

```

```

)

```

```

;; Functions called to ADD and DELETE items from the map

```

```

(DEFINEQ

```

**(MAPW-ADD-OBJECT**

```

(LAMBDA (WINDOW OBJECT) ; Edited 10-Jun-93 19:50 by bane
  (LET ((MAPMARKS (WINDOWPROP WINDOW 'MAPMARKS))
        (UNPLACEDW (WINDOWPROP WINDOW 'MAPUNPLACED))
        UNPLACEDMARKS LASTMARK MARK MARKX MARKY)
    (COND
      ((|for| MARK |in| MAPMARKS |thereis| (EQ OBJECT (|fetch| (MAPMARK OBJECT) |of| MARK)))
        (ERROR "Adding object to a map twice!")))
    (COND
      ((|for| MARK |in| (SETQ UNPLACEDMARKS (WINDOWPROP UNPLACEDW 'MAPMARKS))
        |thereis| (EQ OBJECT (|fetch| (MAPMARK OBJECT) |of| MARK)))
        (ERROR "Adding object to a map twice!")))
      (SETQ LASTMARK (CAR (FLAST UNPLACEDMARKS)))
      (SETQ MARK (MAKE-MAPMARK OBJECT))
      (COND
        (LASTMARK (SETQ MARKX (IPLUS (|fetch| (REGION RIGHT) |of| (|fetch| (MAPMARK MARKREGION) |of| LASTMARK))
                                     8
                                     (|fetch| (MAPMARK HOTSPOTX) |of| MARK)))
          (SETQ MARKY 20)
          (MAPMARK-PLACE UNPLACEDW MARK MARKX MARKY 0 0))
        (T (SETQ MARKX (IMAX 50 (|fetch| (MAPMARK HOTSPOTX) |of| MARK)))
          (SETQ MARKY 20)
          (MAPMARK-PLACE UNPLACEDW MARK MARKX MARKY 0 0)))
      (WINDOWPROP UNPLACEDW 'MAPMARKS (NCONC1 UNPLACEDMARKS MARK))))))

```

**(MAPW-DELETE-OBJECT**

```

(LAMBDA (WINDOW OBJECT) ; Edited 10-Jun-93 16:52 by bane
  (LET ((UNPLACEDW (WINDOWPROP WINDOW 'MAPUNPLACED))
        MARKS)
    (|for| MARK |in| (SETQ MARKS (WINDOWPROP UNPLACEDW 'MAPMARKS)) |when| (EQ OBJECT (|fetch| (MAPMARK OBJECT)
                                                                                               |of| MARK))
      |do| (MAPMARK-UNPLACE UNPLACEDW MARK 0 0)
        (_ OBJECT |SetLocation| NIL)
        (WINDOWPROP UNPLACEDW 'MAPMARKS (REMOVE MARK MARKS)))
    (|for| MARK |in| (SETQ MARKS (WINDOWPROP WINDOW 'MAPMARKS)) |when| (EQ OBJECT (|fetch| (MAPMARK OBJECT)
                                                                                               |of| MARK))
      |do| (MAPMARK-UNPLACE WINDOW MARK 0 0)
        (_ OBJECT |SetLocation| NIL)
        (WINDOWPROP WINDOW 'MAPMARKS (REMOVE MARK MARKS))))))

```

```

)

```

```

(DEFINEQ

```

**(MAPW-MAKE-LEFTMENU**

```

(LAMBDA NIL ; Edited 10-Jun-93 21:20 by bane
  ;; Set up the MAPW left-button menu
  (SETQ *MAPW-LEFTBUTTON-MENU* (|create| MENU
                                     ITEMS _ *MAPW-LEFTBUTTON-ITEMS*)))

```

**(MAPW-MAKE-MIDDLEMENU**

```

(LAMBDA NIL ; Edited 10-Jun-93 22:15 by bane
  ;; Set up the MAPW middle-button menu
  (SETQ *MAPW-MIDDLEBUTTON-MENU* (|create| MENU
                                     ITEMS _ *MAPW-MIDDLEBUTTON-ITEMS*)))

```

**(MAPW-MENU-WHENSELECTEDFN**

```

(LAMBDA (MAINWINDOW ITEM) ; Edited 10-Jun-93 22:11 by bane
  ;; Handle menu requests for a map window.
  (SELECTQ ITEM

```

```

(|Lat/Long Window|                                     ; Toggle the existence of the latitude/longitude display window.
  (LET ((COORDW (WINDOWPROP MAINWINDOW 'COORDW))
        (REG (WINDOWPROP MAINWINDOW 'REGION)))
    (COND
      (COORDW (DETACHWINDOW COORDW MAINWINDOW)
               (CLOSEW COORDW)
               (WINDOWPROP MAINWINDOW 'COORDW NIL))
      (T (SETQ COORDW (COORDW-CREATE MAINWINDOW 'LATLONG))
          (WINDOWPROP MAINWINDOW 'COORDW COORDW))))))
(|Bit-offset Window|                                   ; Toggle the existence of the latitude/longitude display window.
  (LET ((COORDW (WINDOWPROP MAINWINDOW 'COORDW))
        (REG (WINDOWPROP MAINWINDOW 'REGION)))
    (COND
      (COORDW (DETACHWINDOW COORDW MAINWINDOW)
               (CLOSEW COORDW)
               (WINDOWPROP MAINWINDOW 'COORDW NIL))
      (T (SETQ COORDW (COORDW-CREATE MAINWINDOW 'BITS))
          (WINDOWPROP MAINWINDOW 'COORDW COORDW))))))
NIL)))
)

```

```
(RPAQQ *MAPW-LEFTBUTTON-ITEMS* (|Lat/Long Window|))
```

```
(RPAQQ *MAPW-MIDDLEBUTTON-ITEMS* (|Bit-offset Window|))
```

```
(RPAQQ *MAPW-LEFTBUTTON-MENU* NIL)
```

```
(RPAQQ *MAPW-MIDDLEBUTTON-MENU* NIL)
```

```
(MAPW-MAKE-LEFTMENU)
```

```
(MAPW-MAKE-MIDDLEMENU)
```

;; Map references. Each reference is a map and coordinates within the map (in pixels) of the location.

;; You can get ranges and bearings between references.

```
(DECLARE\ : EVAL@COMPILE
```

```
(DATATYPE MAPREF (MAP PIXLAT PIXLONG))
)
```

```
(/DECLAREDATATYPE 'MAPREF ' (POINTER POINTER POINTER)
```

```
;; ---field descriptor list elided by lister---
```

```
' 6)
```

```
(DEFINEQ
```

```
(MAPREF-BEARING
```

```
(LAMBDA (MAPREF1 MAPREF2)
```

; Edited 4-Jun-93 16:57 by sybalsky:mv:envos

;; Compute the range from one map reference to another, regardless of bearing.

;; For now, the two map references must be on the same map -- there's no provision for cross-map ranging

```

(LET ((MAP1 (|fetch| (MAPREF MAP) |of| MAPREF1))
      (MAP2 (|fetch| (MAPREF MAP) |of| MAPREF2))
      ANGLE DX DY)
  (COND
    ((NEQ MAP1 MAP2)
     (ERROR "Bearings between refs on different maps not supported")))
    (SETQ DX (- (|fetch| (MAPREF PIXLONG) |of| MAPREF2)
                (|fetch| (MAPREF PIXLONG) |of| MAPREF1)))
    (SETQ DY (- (|fetch| (MAPREF PIXLAT) |of| MAPREF2)
                (|fetch| (MAPREF PIXLAT) |of| MAPREF1)))
    (COND
      ((ZEROP DX)
       (COND
         (((< DY 0)
          180)
          (T 0)))
      ((ZEROP DY)
       (COND
         (((< DX 0)
          270)
          (T 90)))
      ((< DX 0)
       (COND
         ((< DY 0)
          (+ 270 (ARCTAN (/ DY DX)))))
         (T (- 270 (ARCTAN (/ DY DX))))))
      (T (- 90 (ARCTAN (/ DY DX))))))

```

```
(MAPREF-RANGE
```

```
(LAMBDA (MAPREF1 MAPREF2)
```

; Edited 4-Jun-93 17:00 by sybalsky:mv:envos

;; Compute the range from one map reference to another, regardless of bearing.

;; For now, the two map references must be on the same map -- there's no provision for cross-map ranging

```
(LET ((MAP1 (|fetch| (MAPREF MAP) |of| MAPREF1))
      (MAP2 (|fetch| (MAPREF MAP) |of| MAPREF2))
      (PIXPERYARD (|fetch| (MAP PIXPERYARD) |of| (|fetch| (MAPREF MAP) |of| MAPREF1)))
      DX DY)
  (COND
    ((NEQ MAP1 MAP2)
     (ERROR "Range between refs on different maps not supported")))
  (SETQ DX (- (|fetch| (MAPREF PIXLONG) |of| MAPREF1)
              (|fetch| (MAPREF PIXLONG) |of| MAPREF2)))
  (SETQ DY (- (|fetch| (MAPREF PIXLAT) |of| MAPREF1)
              (|fetch| (MAPREF PIXLAT) |of| MAPREF2)))
  (FQUOTIENT (SQRT (+ (TIMES DX DX)
                      (TIMES DY DY)))
              PIXPERYARD)))

)
```

;; The window for unplaced assets.

(DEFINEQ

### (UNPLACEDW-CREATE

(LAMBDA (REGION)

; Edited 10-Jun-93 16:41 by bane

;; Create a window for holding the unplaced assets of a scenario.

```
(LET ((WINDOW (CREATEW REGION "Unplaced Assets for this Scenario")))
      (* WINDOWPROP WINDOW (QUOTE REPAINTFN)
        (FUNCTION UNPLACEDW-REPAINTFN))
      (WINDOWPROP WINDOW 'BUTTONEVENTFN (FUNCTION UNPLACEDW-BUTTONFN))
      (DSOPERATION 'PAINT WINDOW)
      WINDOW)))
```

### (UNPLACEDW-BUTTONFN

(LAMBDA (WINDOW)

; Edited 10-Jun-93 21:38 by bane

;; BUTTONEVENTFN for map windows.

```
(LET ((X (LASTMOUSEX WINDOW))
      (Y (LASTMOUSEY WINDOW))
      (MAPMARKS (WINDOWPROP WINDOW 'MAPMARKS))
      MARK)
  (COND
    ((MOUSESTATE (NOT (OR LEFT MIDDLE)))
     ; No buttons down.
    )
    ((MOUSESTATE LEFT)
     ; Left button went down
     (|while| (MOUSESTATE LEFT) |do|)
     (SETQ X (LASTMOUSEX WINDOW))
     (SETQ Y (LASTMOUSEY WINDOW))
     (COND
       (*MAPW-MOVING-OBJECT*
        (LET ((LASTMARK (CAR (FLAST MAPMARKS)))
              MARKX MARKY)
          (COND
            (LASTMARK (SETQ MARKX (IPLUS (|fetch| (REGION RIGHT) |of| (|fetch| (MAPMARK MARKREGION)
                                                                                   |of| LASTMARK)))
                                         8
              (|fetch| (MAPMARK HOTSPOTX) |of| *MAPW-MOVING-OBJECT*)))
            (SETQ MARKY 20)
            (MAPMARK-PLACE WINDOW *MAPW-MOVING-OBJECT* MARKX MARKY 0 0))
          (T (SETQ MARKX (IMAX 50 (|fetch| (MAPMARK HOTSPOTX) |of| *MAPW-MOVING-OBJECT*)))
            (SETQ MARKY 20)
            (MAPMARK-PLACE WINDOW *MAPW-MOVING-OBJECT* MARKX MARKY 0 0)))
          (WINDOWPROP WINDOW 'MAPMARKS (NCONC1 MAPMARKS *MAPW-MOVING-OBJECT*))
          (SETQ *MAPW-MOVING-OBJECT* NIL)
          (CURSOR T)))
        (T (LET ((MAPMARKS (WINDOWPROP WINDOW 'MAPMARKS)))
              (SETQ MARK (|for| AMARK |in| MAPMARKS |suchthat| (INSIDE? (|fetch| (MAPMARK MARKREGION)
                                                                                   |of| AMARK)
                                                                           X Y)))
              (CL:WHEN MARK
                (MAPMARK-UNPLACE WINDOW MARK 0 0)
                (SETQ *MAPW-MOVING-OBJECT* MARK)
                (CURSOR (@ (|fetch| (MAPMARK OBJECT) |of| MARK)
                           |::move-cursor|))
                (SETQ MAPMARKS (REMOVE MARK MAPMARKS))
                (|for| AMARK |in| MAPMARKS |when| (> (|fetch| (REGION LEFT) |of| (|fetch| (MAPMARK MARKREGION)
                                                                                   |of| AMARK))
                                                       X)
                  |do| (MAPMARK-RELMOVE WINDOW AMARK (- -8 (|fetch| (REGION WIDTH)
                                                                                   |of| (|fetch| (MAPMARK MARKREGION)
                                                                                   |of| MARK))
                                                       0 0 0))
                  (WINDOWPROP WINDOW 'MAPMARKS MAPMARKS)))))))
```

```
{MEDLEY}<loops>obsolete>LOTSMAP.;1 (UNPLACEDW-BUTTONFN cont.)
```

Page 16

```
((MOUSESTATE MIDDLE) ; Middle button went down  
)))))
```

```
)
```

```
:: Utility functions
```

```
(DEFINEQ
```

```
(SECONDS
```

```
(LAMBDA (DEG MIN SEC)  
  (+ SEC (TIMES 60 (+ MIN (TIMES 60 DEG))))))
```

```
)
```

```
:: HOLDS THE OBJECT BEING MOVED AROUND BY MOUSE ON THE MAP.
```

```
(RPAQ? *MAPW-MOVING-OBJECT* NIL)
```

```
:: A LIST OF ALL THE MAPS KNOWN TO THE SYSTEM:
```


```
(RPAQ? *MAPS* NIL)
```


```
(RPAQ? NIL NIL)
```


```
:: Object support for mappable items.
```

```
(RPAQQ MAPITEM-CURSOR-IMAGE )
```

```
(RPAQQ GENERIC-MAP-IMAGE )
```

```
(RPAQQ AMPHIB-CURSOR-IMAGE )
```

```
(RPAQQ RORO-CURSOR-IMAGE )
```

```
(RPAQQ ELCAS-CURSOR-IMAGE )
```

```
(RPAQQ CRANE-CURSOR-IMAGE )
```

```
(DEFCLASSES |LotsMappableItem|)
```