

File created: 20-May-2000 10:31:55 {DSK}<project>medley3.5>lispusers>PLOT OBJECTS.;3

changes to: (VARS PLOT OBJECTS COMS CIRCLE CROSS)

previous date: 4-Nov-93 14:59:31 {DSK}<project>medley3.5>lispusers>PLOT OBJECTS.;2

Read Table: INTERLISP

Package: INTERLISP

Format: XCCS

::
:: Copyright (c) 1985, 1986, 1987, 1993, 2000 by Xerox Corporation. All rights reserved.

(RPAQQ PLOT OBJECTS COMS

```
[ (FNS COPYCOMPOUND COPYCURVE COPYFILLEDRECTANGLE COPYGENERIC COPYGRAPH OBJECT COPYLINE COPYPOINT
COPYPOLYGON COPYTEXT CREATECOMPOUND CREATECURVE CREATEFILLEDRECTANGLE CREATEGRAPH CREATELINE
CREATEPOINT CREATEPOLYGON CREATETEXT DISTANCETOCOMPOUND DISTANCETOCURVE DISTANCETOFILLEDRECTANGLE
DISTANCETOGRAPH DISTANCETOLINE DISTANCETOPOINT DISTANCETOPOLYGON DISTANCETOTEXT DRAWCOMPOUNDOBJECT
DRAWCURVEOBJECT DRAWFILLEDRECTANGLEOBJECT DRAWGRAPH OBJECT DRAWLINEOBJECT DRAWPOINT OBJECT
DRAWPOLYGON OBJECT DRAWTEXT OBJECT ERASECOMPOUNDOBJECT ERASECURVEOBJECT ERASEFILLEDRECTANGLEOBJECT
ERASEGRAPH OBJECT ERASELINE OBJECT ERASEPOINT OBJECT ERASEPOLYGON OBJECT ERASETEXT OBJECT
EXTENTOF COMPOUND EXTENTOF CURVE EXTENTOFFILLEDRECTANGLE EXTENTOF GRAPH EXTENTOF LINE EXTENTOF POINT
EXTENTOF POLYGON EXTENTOF TEXT GETCOMPOUND GETCURVE GETFILLEDRECTANGLE GETGENERIC GETGRAPH GETLINE
GETPOINT GETPOLYGON GETTEXT HIGHLIGHTCOMPOUND HIGHLIGHTCURVE HIGHLIGHTFILLEDRECTANGLE
HIGHLIGHTGRAPH HIGHLIGHTLINE HIGHLIGHTPOINT HIGHLIGHTPOLYGON HIGHLIGHTTEXT LABELGENERIC LABELPOINT
LABELTEXT LOWLIGHTCOMPOUND MOVECOMPOUND MOVECURVE MOVEFILLEDRECTANGLE MOVELINE MOVEPOINT
MOVEPOLYGON MOVETEXT PLOTCOMPOUND PLOTCURVE PLOTFILLEDRECTANGLE PLOTGRAPH PLOTLINE PLOTPOINT
PLOTPOINTS PLOTPOLYGON PLOTTEXT PUTCOMPOUND PUTCURVE PUTFILLEDRECTANGLE PUTGENERIC PUTGRAPH PUTLINE
PUTPOINT PUTPOLYGON PUTTEXT)
(MACROS L1METRIC L2METRIC)
(VARS CIRCLE CROSS DASH DOT DOTDASH SHADE1 SHADE2 SHADE3 SHADE4 SHADE5 SHADE6 SHADE7 SHADE8 STAR)
(RECORDS COMPOUNDDATA CURVEDATA FILLEDRECTANGLEDATA GRAPHDATA LINEDATA PLOT.STYLE POINTDATA POLYGONDATA
TEXTDATA)
(PROP ARG NAMES PLOTCOMPOUND)
(DECLARE%: DONTEVAL@LOAD DOEVAL@COMPILE DONTCOPY (FILES (LOADCOMP)
PLOT TWODGRAPHICS))
(DECLARE%: DONTEVAL@LOAD DOEVAL@COMPILE DONTCOPY (LOCALVARS . T))
(DECLARE%: DONTEVAL@LOAD DOEVAL@COMPILE DONTCOPY COMPILERVARS (ADDVARS (NLAMA)
(NLAML)
(LAMA PLOTCOMPOUND]))
```

(DEFINEQ

(COPYCOMPOUND

```
[LAMBDA (PLOT OBJECT PLOT) ; Edited 5-May-87 17:45 by jop
;; Copyfn for COMPOUND objects
(PROG ((OBJECTDATA (fetch (PLOT OBJECT OBJECTDATA) of PLOT OBJECT))
(RETURN (create COMPOUNDDATA
COMPONENTS _ (for OBJECT in (fetch (COMPOUNDDATA COMPONENTS) of OBJECTDATA)
collect (COPY PLOT OBJECT OBJECT PLOT))
COMPOUNDTYPE _ (fetch (COMPOUNDDATA COMPOUNDTYPE) of OBJECTDATA))
```

(COPYCURVE

```
[LAMBDA (PLOT OBJECT PLOT) ; Edited 5-May-87 17:46 by jop
;; Copyfn for CURVE objects
(PROG ((OBJECTDATA (fetch (PLOT OBJECT OBJECTDATA) of PLOT OBJECT))
(RETURN (create CURVEDATA
CURVEPOINTS _ (COPYALL (fetch (CURVEDATA CURVEPOINTS) of OBJECTDATA))
STYLE _ (COPYALL (fetch (CURVEDATA STYLE) of OBJECTDATA))
```

(COPYFILLEDRECTANGLE

```
[LAMBDA (PLOT OBJECT PLOT) ; Edited 5-May-87 17:46 by jop
;; Copyfn for FILLEDRECTANGLE objects
(PROG ((OBJECTDATA (fetch (PLOT OBJECT OBJECTDATA) of PLOT OBJECT))
(RETURN (create FILLEDRECTANGLEDATA
OBJECTLEFT _ (fetch (FILLEDRECTANGLEDATA OBJECTLEFT) of OBJECTDATA)
OBJECTBOTTOM _ (fetch (FILLEDRECTANGLEDATA OBJECTBOTTOM) of OBJECTDATA)
OBJECTWIDTH _ (fetch (FILLEDRECTANGLEDATA OBJECTWIDTH) of OBJECTDATA)
OBJECTHEIGHT _ (fetch (FILLEDRECTANGLEDATA OBJECTHEIGHT) of OBJECTDATA)
BORDERWIDTH _ (fetch (FILLEDRECTANGLEDATA BORDERWIDTH) of OBJECTDATA)
TEXTURE _ (fetch (FILLEDRECTANGLEDATA TEXTURE) of OBJECTDATA))
```

(COPYGENERIC

```
[LAMBDA (PLOT OBJECT PLOT) ; Edited 5-May-87 17:46 by jop
;; Default COPYFN
(HCOPYALL (fetch OBJECTDATA of PLOT OBJECT))
```

(COPYGRAPHOBJECT

```
[LAMBDA (PLOTOBJECT PLOT) ; Edited 5-May-87 17:46 by jop
  (PROG ((OBJECTDATA (fetch (PLOTOBJECT OBJECTDATA) of PLOTOBJECT)))
    (RETURN (create GRAPHDATA
      GRAPHFN _ (fetch (GRAPHDATA GRAPHFN) of OBJECTDATA)
      NSAMPLES _ (fetch (GRAPHDATA NSAMPLES) of OBJECTDATA)
      STYLE _ (COPYALL (fetch (GRAPHDATA STYLE) of OBJECTDATA))
```

(COPYLINE

```
[LAMBDA (PLOTOBJECT PLOT) ; Edited 5-May-87 17:46 by jop
  ;; Copyfn for LINE objects
  (PROG ((OBJECTDATA (fetch (PLOTOBJECT OBJECTDATA) of PLOTOBJECT)))
    (RETURN (create LINEDATA
      INFINITESLOPE? _ (fetch (LINEDATA INFINITESLOPE?) of OBJECTDATA)
      SLOPE _ (fetch (LINEDATA SLOPE) of OBJECTDATA)
      CONSTANT _ (fetch (LINEDATA CONSTANT) of OBJECTDATA)
      STYLE _ (COPYALL (fetch (LINEDATA STYLE) of OBJECTDATA))
```

(COPYPPOINT

```
[LAMBDA (PLOTOBJECT PLOT) ; Edited 5-May-87 17:46 by jop
  ;; Copyfn for POINT objects
  (PROG ((OBJECTDATA (fetch (PLOTOBJECT OBJECTDATA) of PLOTOBJECT)))
    (RETURN (create POINTDATA
      POINTPOSITION _ (COPYALL (fetch (POINTDATA POINTPOSITION) of OBJECTDATA))
      SYMBOL _ (fetch (POINTDATA SYMBOL) of OBJECTDATA))
```

(COPYPOLYGON

```
[LAMBDA (PLOTOBJECT PLOT) ; Edited 5-May-87 17:46 by jop
  ;; Copyfn for POLYGON objects
  (PROG ((OBJECTDATA (fetch (PLOTOBJECT OBJECTDATA) of PLOTOBJECT)))
    (RETURN (create POLYGONDATA
      POLYGONPOINTS _ (COPYALL (fetch (POLYGONDATA POLYGONPOINTS) of OBJECTDATA))
      STYLE _ (COPYALL (fetch (POLYGONDATA STYLE) of OBJECTDATA))
```

(COPYTEXT

```
[LAMBDA (PLOTOBJECT PLOT) ; Edited 5-May-87 17:47 by jop
  ;; Copyfn for TEXT objects
  (PROG ((OBJECTDATA (fetch (PLOTOBJECT OBJECTDATA) of PLOTOBJECT)))
    (RETURN (create TEXTDATA
      TEXTPOSITION _ (COPYALL (fetch (TEXTDATA TEXTPOSITION) of OBJECTDATA))
      TEXT _ (COPYALL (fetch (TEXTDATA TEXT) of OBJECTDATA))
      FONT _ (fetch (TEXTDATA FONT) of OBJECTDATA))
```

(CREATECOMPOUND

```
[LAMBDA (COMPOUNDTYPE COMPONENTS LABEL MENU) ; Edited 5-May-87 17:47 by jop
  ;; create a compound plot object. First is the required Compoundtype, then the components, a list of plotobjects, then the optional label, and menu
  (CREATEPLOTOBJECT COMPOUNDFNS 'COMPOUND LABEL MENU (create COMPOUNDDATA
    COMPONENTS _ COMPONENTS
    COMPOUNDTYPE _ COMPOUNDTYPE])
```

(CREATECURVE

```
[LAMBDA (POSITIONS LABEL STYLE MENU) ; Edited 5-May-87 17:47 by jop
  ;; Create a curve plot object
  (CREATEPLOTOBJECT CURVEFNS 'CURVE LABEL MENU (create CURVEDATA
    CURVEPOINTS _ POSITIONS
    STYLE _ (COND
      ((FIXP STYLE)
        (create PLOT.STYLE
          LINEWIDTH _ STYLE))
      ((LISTP STYLE)
        (create PLOT.STYLE
          LINEWIDTH _ (CAR STYLE)
          DASHING _ (CADR STYLE)
          COLOR _ (CADDR STYLE)))
      (T (create PLOT.STYLE
          LINEWIDTH _ 1))
```

(CREATEFILLEDRECTANGLE

```
[LAMBDA (LEFT BOTTOM WIDTH HEIGHT LABEL TEXTURE BORDERWIDTH MENU) ; Edited 5-May-87 17:47 by jop
  ;; Create a filledrectangle plot object
  (if (NULL TEXTURE)
```

```

then (SETQ TEXTURE 'SHADE3))
(CREATEPLOT OBJECT FILLEDRECTANGLEFNS 'FILLEDRECTANGLE LABEL MENU
  (create FILLEDRECTANGLEDATA
    OBJECTLEFT _ LEFT
    OBJECTBOTTOM _ BOTTOM
    OBJECTWIDTH _ WIDTH
    OBJECTHEIGHT _ HEIGHT
    BORDERWIDTH _ (OR BORDERWIDTH 1)
    TEXTURE _ TEXTURE])

```

(CREATEGRAPH

```

[LAMBDA (GRAPHFN NSAMPLES LABEL STYLE MENU) ; Edited 5-May-87 17:47 by job
  (CREATEPLOT OBJECT GRAPHFNS 'GRAPH LABEL MENU (create GRAPHDATA
    GRAPHFN _ GRAPHFN
    NSAMPLES _ (OR (FIXP NSAMPLES)
      100)
    STYLE _ (if (FIXP STYLE)
      then (create PLOT.STYLE
        LINWIDTH _ STYLE)
      elseif (LISTP STYLE)
        then (create PLOT.STYLE
          LINWIDTH _ (CAR STYLE)
          DASHING _ (CADR STYLE)
          COLOR _ (CADDR STYLE))
      else (create PLOT.STYLE
        LINWIDTH _ 1)))

```

(CREATELINE

```

[LAMBDA (SLOPE CONSTANT LABEL STYLE MENU) ; Edited 5-May-87 17:47 by job
  ;; Create a line plot object
  (CREATEPLOT OBJECT LINEFNS 'LINE LABEL MENU (create LINEDATA
    INFINITESLOPE? _ (NOT SLOPE)
    SLOPE _ (OR SLOPE 0.0)
    CONSTANT _ CONSTANT
    STYLE _ (COND
      ((FIXP STYLE)
        (create PLOT.STYLE
          LINWIDTH _ STYLE))
      ((LISTP STYLE)
        (create PLOT.STYLE
          LINWIDTH _ (CAR STYLE)
          DASHING _ (CADR STYLE)
          COLOR _ (CADDR STYLE)))
      (T (create PLOT.STYLE
        LINWIDTH _ 1)))

```

(CREATEPOINT

```

[LAMBDA (POSITION LABEL SYMBOL MENU) ; Edited 5-May-87 17:48 by job
  ;; Create a point plot object
  (if (NULL SYMBOL)
    then (SETQ SYMBOL STAR))
  (CREATEPLOT OBJECT POINTFNS 'POINT LABEL MENU (create POINTDATA
    POINTPOSITION _ POSITION
    SYMBOL _ SYMBOL])

```

(CREATEPOLYGON

```

[LAMBDA (POSITIONS LABEL STYLE MENU) ; Edited 5-May-87 17:48 by job
  ;; Create a polygon Plot object
  (CREATEPLOT OBJECT POLYGONFNS 'POLYGON LABEL MENU (create POLYGONDATA
    POLYGONPOINTS _ POSITIONS
    STYLE _ (if (FIXP STYLE)
      then (create PLOT.STYLE
        LINWIDTH _ STYLE)
      elseif (LISTP STYLE)
        then (create PLOT.STYLE
          LINWIDTH _ (CAR STYLE)
          DASHING _ (CADR STYLE)
          COLOR _ (CADDR STYLE))
      else (create PLOT.STYLE
        LINWIDTH _ 1)))

```

(CREATETEXT

```

[LAMBDA (POSITION TEXT LABEL FONT MENU) ; Edited 5-May-87 17:48 by job
  ;; Create a Text Plot object
  (CREATEPLOT OBJECT TEXTFNS 'TEXT LABEL MENU (create TEXTDATA
    TEXTPOSITION _ POSITION
    TEXT _ TEXT
    FONT _ FONT])

```

(DISTANCETOCOMPOUND

```
[LAMBDA (COMPOUNDDATA STREAMPOSITION PLOT) (* edited%: "27-Mar-86 21:25")
  (PROG [(COMPONENTS (fetch (COMPOUNDDATA COMPONENTS) of (fetch OBJECTDATA of COMPOUNDDATA)
    (RETURN (bind (CMIN _ (DISTANCETOPLOT OBJECT (CAR COMPONENTS)
      STREAMPOSITION PLOT))
        PMIN for PART in (CDR COMPONENTS) do (SETQ PMIN (DISTANCETOPLOT OBJECT PART STREAMPOSITION
          PLOT))
          (if (LESSP PMIN CMIN)
            then (SETQ CMIN PMIN))
        finally (RETURN CMIN])])])])
```

(DISTANCETOCURVE

```
[LAMBDA (CURVEDATA STREAMPOSITION PLOT) (* edited%: "21-May-85 15:28")
  (L1METRIC STREAMPOSITION (for POINT in (fetch (CURVEDATA STREAMPOINTS) of (fetch OBJECTDATA of CURVEDATA))
    smallest (L1METRIC POINT STREAMPOSITION]))
```

(DISTANCETOFILLEDRECTANGLE

```
[LAMBDA (FILLEDRECTANGLE STREAMPOSITION PLOT) ; Edited 5-May-87 17:48 by jop
  (PROG ((OBJECTDATA (fetch OBJECTDATA of FILLEDRECTANGLE))
    (CLOSEST (CONSTANT (create POSITION)))
    (STREAMX (fetch XCOORD of STREAMPOSITION))
    (STREAMY (fetch YCOORD of STREAMPOSITION))
    STREAMLEFT STREAMBOTTOM STREAMRIGHT STREAMTOP INSIDEFLG)
  (SETQ STREAMLEFT (fetch (FILLEDRECTANGLEDATA STREAMLEFT) of OBJECTDATA))
  (SETQ STREAMBOTTOM (fetch (FILLEDRECTANGLEDATA STREAMBOTTOM) of OBJECTDATA))
  (SETQ STREAMRIGHT (fetch (FILLEDRECTANGLEDATA STREAMRIGHT) of OBJECTDATA))
  (SETQ STREAMTOP (fetch (FILLEDRECTANGLEDATA STREAMTOP) of OBJECTDATA))
  [replace XCOORD of CLOSEST
    with (if (GREATERP STREAMX STREAMRIGHT)
      then STREAMRIGHT
      elseif (LESSP STREAMX STREAMLEFT)
      then STREAMLEFT
      else (if (OR (GREATERP STREAMY STREAMTOP)
        (LESSP STREAMY STREAMBOTTOM))
        then STREAMX
        else (SETQ INSIDEFLG T)
          ;; Hack to deal with the case of adjacent filledrectangles. Bonus subtracted from metric if cursor inside
          ;; rectangle
          (if (LESSP (IMIN (IDIFFERENCE STREAMTOP STREAMY)
            (IDIFFERENCE STREAMY STREAMBOTTOM))
              (IMIN (IDIFFERENCE STREAMRIGHT STREAMX)
                (IDIFFERENCE STREAMX STREAMLEFT)))
            then STREAMX
            else (if (LESSP (IDIFFERENCE STREAMRIGHT STREAMX)
              (IDIFFERENCE STREAMX STREAMLEFT))
              then STREAMRIGHT
              else STREAMLEFT))
    [replace YCOORD of CLOSEST
      with (if (GREATERP STREAMY STREAMTOP)
        then STREAMTOP
        elseif (LESSP STREAMY STREAMBOTTOM)
        then STREAMBOTTOM
        else (if (OR (GREATERP STREAMX STREAMRIGHT)
          (LESSP STREAMX STREAMLEFT))
          then STREAMY
          else (if (LESSP (IMIN (IDIFFERENCE STREAMRIGHT STREAMX)
            (IDIFFERENCE STREAMX STREAMLEFT))
              (IMIN (IDIFFERENCE STREAMTOP STREAMY)
                (IDIFFERENCE STREAMY STREAMBOTTOM)))
            then STREAMY
            else (if (LESSP (IDIFFERENCE STREAMTOP STREAMY)
              (IDIFFERENCE STREAMY STREAMBOTTOM))
              then STREAMTOP
              else STREAMBOTTOM))
    (RETURN (if INSIDEFLG
      then (IDIFFERENCE (L1METRIC STREAMPOSITION CLOSEST)
        2)
      else (L1METRIC STREAMPOSITION CLOSEST]))]
```

(DISTANCETOGRAPH

```
[LAMBDA (GRAPHOBJECT STREAMPOSITION PLOT) (* jop%: "12-Dec-85 13:15")
  (L1METRIC STREAMPOSITION (for POINT in (fetch (GRAPHDATA STREAMPOSITIONS) of (fetch OBJECTDATA of GRAPHOBJECT))
    smallest (L1METRIC POINT STREAMPOSITION]))
```

(DISTANCETOLINE

```
[LAMBDA (LINEOBJECT STREAMPOSITION PLOT) ; Edited 4-Nov-93 14:59 by rmk:
  ; Edited 5-May-87 17:49 by jop
  (PROG ((X0 (fetch XCOORD of STREAMPOSITION))
    (Y0 (fetch YCOORD of STREAMPOSITION))
    (STREAMSLOPE (fetch STREAMSLOPE of (fetch OBJECTDATA of LINEOBJECT))))]
```



```

STREAMBOTTOM))
(SETQ STREAMRIGHT (PLUS STREAMLEFT STREAMWIDTH))
(SETQ STREAMTOP (PLUS STREAMBOTTOM STREAMHEIGHT))
(CLIPPED.BITBLT STREAMSUBREGION NIL NIL NIL STREAM STREAMLEFT STREAMBOTTOM STREAMWIDTH STREAMHEIGHT
 'TEXTURE
 'PAINT TEXTURE)
(MOVETO STREAMLEFT STREAMBOTTOM STREAM)
(CLIPPED.DRAWTO STREAMSUBREGION STREAMRIGHT STREAMBOTTOM BORDERWIDTH 'REPLACE STREAM)
(CLIPPED.DRAWTO STREAMSUBREGION STREAMRIGHT STREAMTOP BORDERWIDTH 'REPLACE STREAM)
(CLIPPED.DRAWTO STREAMSUBREGION STREAMLEFT STREAMTOP BORDERWIDTH 'REPLACE STREAM)
(CLIPPED.DRAWTO STREAMSUBREGION STREAMLEFT STREAMBOTTOM BORDERWIDTH 'REPLACE STREAM)
(if (EQ STREAM (WINDOWPROP (fetch (PLOT PLOTWINDOW) of PLOT)
 'DSP))
 then (replace (FILLEDRECTANGLEDATA STREAMLEFT) of OBJECTDATA with STREAMLEFT)
 (replace (FILLEDRECTANGLEDATA STREAMBOTTOM) of OBJECTDATA with STREAMBOTTOM)
 (replace (FILLEDRECTANGLEDATA STREAMWIDTH) of OBJECTDATA with STREAMWIDTH)
 (replace (FILLEDRECTANGLEDATA STREAMHEIGHT) of OBJECTDATA with STREAMHEIGHT])

```

(DRAWGRAPHOBJECT

```

[LAMBDA (GRAPHOBJECT VIEWPORT PLOT) ; Edited 5-May-87 17:50 by jop
 (LET* ((STREAM (fetch (VIEWPORT PARENTSTREAM) of VIEWPORT))
 (STREAMSUBREGION (fetch (VIEWPORT STREAMSUBREGION) of VIEWPORT))
 (OBJECTDATA (fetch (PLOT OBJECT OBJECTDATA) of GRAPHOBJECT))
 (XUPPER (fetch (PLOT XUPPER) of PLOT))
 (XLOWER (fetch (PLOT XLOWER) of PLOT))
 (YUPPER (fetch (PLOT YUPPER) of PLOT))
 (YLOWER (fetch (PLOT YLOWER) of PLOT))
 (GRAPHFN (fetch (GRAPHDATA GRAPHFN) of OBJECTDATA))
 (NSAMPLES (fetch (GRAPHDATA NSAMPLES) of OBJECTDATA))
 (STYLE (fetch (GRAPHDATA STYLE) of OBJECTDATA))
 (LINEWIDTH (TIMES (DSPSCALE NIL STREAM)
 (fetch (PLOT.STYLE LINEWIDTH) of STYLE)))
 (DASHING (fetch (PLOT.STYLE DASHING) of STYLE))
 (COLOR (fetch (PLOT.STYLE COLOR) of STYLE))
 STREAMPOSITIONS)
 [SETQ STREAMPOSITIONS (NCONC1 (bind (INC _ (FQUOTIENT (FDIFFERENCE XUPPER XLOWER)
 (SUB1 NSAMPLES)))
 for I from 1 to (SUB1 NSAMPLES) as X from XLOWER by INC
 collect (CREATEPOSITION (WORLDTOSTREAMX X VIEWPORT)
 (WORLDTOSTREAMY (APPLY* GRAPHFN X)
 VIEWPORT)))
 (CREATEPOSITION (WORLDTOSTREAMX XUPPER VIEWPORT)
 (WORLDTOSTREAMY (APPLY* GRAPHFN XUPPER)
 VIEWPORT))
 (first (MOVETO (fetch XCOORD of (CAR STREAMPOSITIONS))
 (fetch YCOORD of (CAR STREAMPOSITIONS))
 STREAM)
 for PT in (CDR STREAMPOSITIONS) do (CLIPPED.DRAWTO STREAMSUBREGION (fetch XCOORD of PT)
 (fetch YCOORD of PT)
 LINEWIDTH
 'REPLACE STREAM COLOR DASHING))
 (if (EQ STREAM (WINDOWPROP (fetch (PLOT PLOTWINDOW) of PLOT)
 'DSP))
 then (replace (GRAPHDATA STREAMPOSITIONS) of OBJECTDATA with STREAMPOSITIONS])

```

(DRAWLINEOBJECT

```

[LAMBDA (LINEOBJECT VIEWPORT PLOT) ; Edited 5-May-87 17:50 by jop
 (LET* ((STREAM (fetch (VIEWPORT PARENTSTREAM) of VIEWPORT))
 (STREAMSUBREGION (fetch (VIEWPORT STREAMSUBREGION) of VIEWPORT))
 (OBJECTDATA (fetch (PLOT OBJECT OBJECTDATA) of LINEOBJECT))
 (XUPPER (fetch (PLOT XUPPER) of PLOT))
 (XLOWER (fetch (PLOT XLOWER) of PLOT))
 (YUPPER (fetch (PLOT YUPPER) of PLOT))
 (YLOWER (fetch (PLOT YLOWER) of PLOT))
 (STYLE (fetch (LINEDATA STYLE) of OBJECTDATA))
 (LINEWIDTH (TIMES (DSPSCALE NIL STREAM)
 (fetch (PLOT.STYLE LINEWIDTH) of STYLE)))
 (DASHING (fetch (PLOT.STYLE DASHING) of STYLE))
 (COLOR (fetch (PLOT.STYLE COLOR) of STYLE))
 (INFINITESLOPE? (fetch (LINEDATA INFINITESLOPE?) of OBJECTDATA))
 (SLOPE (fetch (LINEDATA SLOPE) of OBJECTDATA))
 (CONSTANT (fetch (LINEDATA CONSTANT) of OBJECTDATA))
 STREAMSLOPE STREAMCONSTANT STREAMPT1 STREAMPT2 X1 Y1 X2 Y2)
 [SETQ X1 (COND
 (INFINITESLOPE? CONSTANT)
 (T XLOWER)))
 [SETQ Y1 (COND
 (INFINITESLOPE? YLOWER)
 (T (FPLUS CONSTANT (FTIMES SLOPE X1))
 (INFINITESLOPE? CONSTANT)
 (T XUPPER)))
 [SETQ X2 (COND
 (INFINITESLOPE? YUPPER)
 (T (FPLUS CONSTANT (FTIMES SLOPE X2))

```

```
[SETQ STREAMSLOPE (AND (NOT INFINITESLOPE?)
  (FTIMES SLOPE (FQUOTIENT (fetch (VIEWPORT WORLDSTOSTREAMMY) of VIEWPORT)
    (fetch (VIEWPORT WORLDSTOSTREAMMX) of VIEWPORT))
  (INFINITESLOPE? (WORLDSTOSTREAMX CONSTANT VIEWPORT))
  (T (FDIFFERENCE (WORLDSTOSTREAMY CONSTANT VIEWPORT)
    (FTIMES STREAMSLOPE (fetch (VIEWPORT WORLDSTOSTREAMAX) of VIEWPORT))
  (SETQ STREAMPT1 (CREATEPOSITION (WORLDSTOSTREAMX X1 VIEWPORT)
    (WORLDSTOSTREAMY Y1 VIEWPORT)))
  (SETQ STREAMPT2 (CREATEPOSITION (WORLDSTOSTREAMX X2 VIEWPORT)
    (WORLDSTOSTREAMY Y2 VIEWPORT)))
  (CLIPPED.DRAWBETWEEN STREAMSUBREGION STREAMPT1 STREAMPT2 LINETHICKNESS 'REPLACE STREAM COLOR DASHING)
  (COND
    ((EQ STREAM (WINDOWPROP (fetch (PLOT PLOTWINDOW) of PLOT)
      'DSP))
    (COND
      (STREAMSLOPE (replace (replace (LINEDATA STREAMSLOPE) of OBJECTDATA with STREAMSLOPE))
        (T (replace (LINEDATA STREAMSLOPE) of OBJECTDATA with 0.0)))
      (replace (LINEDATA STREAMCONSTANT) of OBJECTDATA with STREAMCONSTANT)
      (replace (LINEDATA STREAMPT1) of OBJECTDATA with STREAMPT1)
      (replace (LINEDATA STREAMPT2) of OBJECTDATA with STREAMPT2]))
```

(DRAWPOINTOBJECT

[LAMBDA (POINT VIEWPORT PLOT) ; Edited 5-May-87 17:50 by jop

:: Draw a glyph at POINTPOSITION. SYMBOL is the glyph to be drawn.

```
(LET* ((STREAM (fetch (VIEWPORT PARENTSTREAM) of VIEWPORT))
  (STREAMSUBREGION (fetch (VIEWPORT STREAMSUBREGION) of VIEWPORT))
  (OBJECTDATA (fetch (PLOT OBJECT OBJECTDATA) of OBJECTDATA))
  (SYMBOL (fetch (POINTDATA SYMBOL) of OBJECTDATA))
  (PT (fetch (POINTDATA POINTPOSITION) of OBJECTDATA))
  (STREAMPT (WORLDSTOSTREAM PT VIEWPORT)))
  (CLIPPED.PLOTAT STREAMSUBREGION STREAMPT SYMBOL STREAM)
  (if (EQ STREAM (WINDOWPROP (fetch (PLOT PLOTWINDOW) of PLOT)
    'DSP))
    then (replace (POINTDATA STREAMPOSITION) of OBJECTDATA with STREAMPT]))
```

(DRAWPOLYGONOBJECT

[LAMBDA (POLYGONOBJECT VIEWPORT PLOT) ; Edited 5-May-87 17:50 by jop

:: Draws a polygon in VIEWPORT.

```
(LET* ((STREAM (fetch (VIEWPORT PARENTSTREAM) of VIEWPORT))
  (STREAMSUBREGION (fetch (VIEWPORT STREAMSUBREGION) of VIEWPORT))
  (OBJECTDATA (fetch (PLOT OBJECT OBJECTDATA) of POLYGONOBJECT))
  (POINTS (fetch (POLYGONDATA POLYGONPOINTS) of OBJECTDATA))
  (STREAMPOINTS (for PT in POINTS collect (WORLDSTOSTREAM PT VIEWPORT)))
  (STYLE (fetch (POLYGONDATA STYLE) of OBJECTDATA))
  (LINETHICKNESS (TIMES (DSPSCALE NIL STREAM)
    (fetch (PLOT.STYLE LINETHICKNESS) of STYLE)))
  (DASHING (fetch (PLOT.STYLE DASHING) of STYLE))
  (COLOR (fetch (PLOT.STYLE COLOR) of STYLE))
  (bind (START _ (CAR STREAMPOINTS)) first (MOVETO (fetch XCOORD of START)
    (fetch YCOORD of START)
    STREAM)
    for PT in (CDR STREAMPOINTS) do (CLIPPED.DRAWTO STREAMSUBREGION (fetch XCOORD of PT)
    (fetch YCOORD of PT)
    LINETHICKNESS
    'REPLACE STREAM COLOR DASHING)
    finally (CLIPPED.DRAWTO STREAMSUBREGION (fetch XCOORD of START)
    (fetch YCOORD of START)
    LINETHICKNESS
    'REPLACE STREAM COLOR DASHING))
  (if (EQ STREAM (WINDOWPROP (fetch (PLOT PLOTWINDOW) of PLOT)
    'DSP))
    then (replace (POLYGONDATA STREAMPOINTS) of OBJECTDATA with STREAMPOINTS]))
```

(DRAWTEXTOBJECT

[LAMBDA (TEXTOBJECT VIEWPORT PLOT) ; Edited 5-May-87 17:51 by jop

```
(LET* ((STREAM (fetch (VIEWPORT PARENTSTREAM) of VIEWPORT))
  (STREAMSUBREGION (fetch (VIEWPORT STREAMSUBREGION) of VIEWPORT))
  (OBJECTDATA (fetch (PLOT OBJECT OBJECTDATA) of TEXTOBJECT))
  (TEXT (fetch (TEXTDATA TEXT) of OBJECTDATA))
  (FONT (fetch (TEXTDATA FONT) of OBJECTDATA))
  (PT (fetch (TEXTDATA TEXTPOSITION) of OBJECTDATA))
  STREAMX STREAMY)
  (SETQ STREAMX (WORLDSTOSTREAMX (fetch XCOORD of PT)
  VIEWPORT))
  (SETQ STREAMY (WORLDSTOSTREAMY (fetch YCOORD of PT)
  VIEWPORT))
  (RESETLST
    (RESETSAVE (DSPFONT FONT STREAM)
      (LIST 'DSPFONT (DSPFONT NIL STREAM)
        STREAM))
    (MOVETO STREAMX STREAMY STREAM))
```

```

(CLIPPED.PRINT STREAMSUBREGION TEXT STREAM))
(COND
  ((EQ STREAM (WINDOWPROP (fetch (PLOT PLOTWINDOW) of PLOT)
    'DSP))
    (replace (TEXTDATA STREAMPOSITION) of OBJECTDATA with (CREATEPOSITION STREAMX STREAMY]))

```

(ERASECOMPOUNDOBJECT

```

[LAMBDA (COMPOUNDOBJECT VIEWPORT PLOT) ; (* edited%: "27-Mar-86 21:26")
  (for OBJECT in (fetch (COMPOUNDDATA COMPONENTS) of (fetch (PLOT OBJECT OBJECTDATA) of COMPOUNDOBJECT))
    do (ERASEPLOT OBJECT PLOT]))

```

(ERASECURVEOBJECT

```

[LAMBDA (CURVEOBJECT VIEWPORT) ; Edited 5-May-87 17:51 by jop

```

;; Erase the CURVEOBJECT, using the cached stream coordinates

```

(LET* ((STREAM (fetch (VIEWPORT PARENTSTREAM) of VIEWPORT))
  (STREAMSUBREGION (fetch (VIEWPORT STREAMSUBREGION) of VIEWPORT))
  (OBJECTDATA (fetch (PLOT OBJECT OBJECTDATA) of CURVEOBJECT))
  (STREAMPOINTS (fetch (CURVEDATA STREAMPOINTS) of OBJECTDATA))
  (STYLE (fetch (CURVEDATA STYLE) of OBJECTDATA))
  (LINEWIDTH (IPLUS 2 (fetch (PLOT.STYLE LINEWIDTH) of STYLE)))
  (COLOR (fetch (PLOT.STYLE COLOR) of STYLE)))
  (first (MOVETO (fetch XCOORD of (CAR STREAMPOINTS))
    (fetch YCOORD of (CAR STREAMPOINTS))
    STREAM)
    for PT in (CDR STREAMPOINTS) do (CLIPPED.DRAWTO STREAMSUBREGION (fetch XCOORD of PT)
      (fetch YCOORD of PT)
      LINEWIDTH
      'ERASE STREAM COLOR]))

```

(ERASEFILLEDRECTANGLEOBJECT

```

[LAMBDA (FILLEDRECTANGLE VIEWPORT PLOT) ; Edited 5-May-87 17:51 by jop

```

```

(LET* ((STREAM (fetch (VIEWPORT PARENTSTREAM) of VIEWPORT))
  (STREAMSUBREGION (fetch (VIEWPORT STREAMSUBREGION) of VIEWPORT))
  (OBJECTDATA (fetch (PLOT OBJECT OBJECTDATA) of FILLEDRECTANGLE))
  (TEXTURE (fetch (FILLEDRECTANGLEDATA TEXTURE) of OBJECTDATA))
  (BORDERWIDTH (TIMES (DSPSCALE NIL STREAM)
    (fetch (FILLEDRECTANGLEDATA BORDERWIDTH) of OBJECTDATA)))
  (STREAMLEFT (fetch (FILLEDRECTANGLEDATA STREAMLEFT) of OBJECTDATA))
  (STREAMBOTTOM (fetch (FILLEDRECTANGLEDATA STREAMBOTTOM) of OBJECTDATA))
  (STREAMWIDTH (fetch (FILLEDRECTANGLEDATA STREAMWIDTH) of OBJECTDATA))
  (STREAMHEIGHT (fetch (FILLEDRECTANGLEDATA STREAMHEIGHT) of OBJECTDATA))
  (STREAMRIGHT (fetch (FILLEDRECTANGLEDATA STREAMRIGHT) of OBJECTDATA))
  (STREAMTOP (fetch (FILLEDRECTANGLEDATA STREAMTOP) of OBJECTDATA)))
  (MOVETO STREAMLEFT STREAMBOTTOM STREAM)
  (CLIPPED.BITBLT STREAMSUBREGION NIL NIL NIL STREAM STREAMLEFT STREAMBOTTOM STREAMWIDTH STREAMHEIGHT
    'TEXTURE
    'INVERT TEXTURE)
  (CLIPPED.DRAWTO STREAMSUBREGION STREAMRIGHT STREAMBOTTOM BORDERWIDTH 'ERASE STREAM)
  (CLIPPED.DRAWTO STREAMSUBREGION STREAMRIGHT STREAMTOP BORDERWIDTH 'ERASE STREAM)
  (CLIPPED.DRAWTO STREAMSUBREGION STREAMLEFT STREAMTOP BORDERWIDTH 'ERASE STREAM)
  (CLIPPED.DRAWTO STREAMSUBREGION STREAMLEFT STREAMBOTTOM BORDERWIDTH 'ERASE STREAM]))

```

(ERASEGRAPHOBJECT

```

[LAMBDA (GRAPHOBJECT VIEWPORT) ; Edited 5-May-87 17:51 by jop

```

```

(LET* ((STREAM (fetch (VIEWPORT PARENTSTREAM) of VIEWPORT))
  (STREAMSUBREGION (fetch (VIEWPORT STREAMSUBREGION) of VIEWPORT))
  (OBJECTDATA (fetch (PLOT OBJECT OBJECTDATA) of GRAPHOBJECT))
  (STREAMPOSITIONS (fetch (GRAPHDATA STREAMPOSITIONS) of OBJECTDATA))
  (STYLE (fetch (GRAPHDATA STYLE) of OBJECTDATA))
  (LINEWIDTH (IPLUS 2 (fetch (PLOT.STYLE LINEWIDTH) of STYLE)))
  (COLOR (fetch (PLOT.STYLE COLOR) of STYLE)))
  (first (MOVETO (fetch XCOORD of (CAR STREAMPOSITIONS))
    (fetch YCOORD of (CAR STREAMPOSITIONS))
    STREAM)
    for PT in (CDR STREAMPOSITIONS) do (CLIPPED.DRAWTO STREAMSUBREGION (fetch XCOORD of PT)
      (fetch YCOORD of PT)
      LINEWIDTH
      'ERASE STREAM COLOR]))

```

(ERASELINEOBJECT

```

[LAMBDA (LINEOBJECT VIEWPORT PLOT) ; Edited 5-May-87 17:51 by jop

```

```

(LET* ((STREAM (fetch (VIEWPORT PARENTSTREAM) of VIEWPORT))
  (STREAMSUBREGION (fetch (VIEWPORT STREAMSUBREGION) of VIEWPORT))
  (OBJECTDATA (fetch (PLOT OBJECT OBJECTDATA) of LINEOBJECT))
  (STYLE (fetch (LINEDATA STYLE) of OBJECTDATA))
  (LINEWIDTH (IPLUS (fetch (PLOT.STYLE LINEWIDTH) of STYLE)
    2)))
  (COLOR (fetch (PLOT.STYLE COLOR) of STYLE))
  (STREAMPT1 (fetch (LINEDATA STREAMPT1) of OBJECTDATA))
  (STREAMPT2 (fetch (LINEDATA STREAMPT2) of OBJECTDATA))
  (CLIPPED.DRAWBETWEEN STREAMSUBREGION STREAMPT1 STREAMPT2 LINEWIDTH 'ERASE STREAM COLOR]))

```


(ERASEPOINTOBJECT

[LAMBDA (POINT VIEWPORT PLOT) ; Edited 5-May-87 17:51 by jop

;; Erase POINT, using cached stream coordinates

```
(LET* ((STREAM (fetch (VIEWPORT PARENTSTREAM) of VIEWPORT))
      (STREAMSUBREGION (fetch (VIEWPORT STREAMSUBREGION) of VIEWPORT))
      (OBJECTDATA (fetch (PLOT OBJECT OBJECTDATA) of POINT))
      (SYMBOL (fetch (POINTDATA SYMBOL) of OBJECTDATA))
      (STREAMPT (fetch (POINTDATA STREAMPOSITION) of OBJECTDATA))
      (CLIPPED.PLOTAT STREAMSUBREGION STREAMPT SYMBOL STREAM 'ERASE))
```

(ERASEPOLYGONOBJECT

[LAMBDA (POLYGONOBJECT VIEWPORT) ; Edited 5-May-87 17:52 by jop

;; Erase a POLYGONDATA

```
(LET* ((STREAM (fetch (VIEWPORT PARENTSTREAM) of VIEWPORT))
      (STREAMSUBREGION (fetch (VIEWPORT STREAMSUBREGION) of VIEWPORT))
      (OBJECTDATA (fetch (PLOT OBJECT OBJECTDATA) of POLYGONOBJECT))
      (STREAMPOINTS (fetch (POLYGONDATA STREAMPOINTS) of OBJECTDATA))
      (STYLE (fetch (POLYGONDATA STYLE) of OBJECTDATA))
      (LINEWIDTH (IPLUS 2 (fetch (PLOT.STYLE LINEWIDTH) of STYLE)))
      (COLOR (fetch (PLOT.STYLE COLOR) of STYLE)))
  (bind (START _ (CAR STREAMPOINTS)) first (MOVETO (fetch XCOORD of START)
                                                    (fetch YCOORD of START)
                                                    STREAM)
        for PT in (CDR STREAMPOINTS) do (CLIPPED.DRAWTO STREAMSUBREGION (fetch XCOORD of PT)
                                                                           (fetch YCOORD of PT)
                                                                           LINEWIDTH
                                                                           'ERASE STREAM COLOR)
        finally (CLIPPED.DRAWTO STREAMSUBREGION (fetch XCOORD of START)
                                                  (fetch YCOORD of START)
                                                  LINEWIDTH
                                                  'ERASE STREAM COLOR])
```

(ERASETEXTOBJECT

[LAMBDA (TEXTOBJECT VIEWPORT PLOT) ; Edited 5-May-87 17:52 by jop

;; ERASE the TEXTDATA

```
(LET* ((STREAM (fetch (VIEWPORT PARENTSTREAM) of VIEWPORT))
      (STREAMSUBREGION (fetch (VIEWPORT STREAMSUBREGION) of VIEWPORT))
      (OBJECTDATA (fetch (PLOT OBJECT OBJECTDATA) of TEXTOBJECT))
      (TEXT (fetch (TEXTDATA TEXT) of OBJECTDATA))
      (FONT (fetch (TEXTDATA FONT) of OBJECTDATA))
      (STREAMPOSITION (fetch (TEXTDATA STREAMPOSITION) of OBJECTDATA))
      (STREAMX (fetch XCOORD of STREAMPOSITION))
      (STREAMY (fetch YCOORD of STREAMPOSITION))
      (BLANCREGION)
      (RESETLST
        (RESETSAVE (DSPFONT FONT STREAM)
                   (LIST 'DSPFONT (DSPFONT NIL STREAM)
                          STREAM))
        (MOVETO STREAMX STREAMY STREAM)
        (SETQ BLANCREGION (STRINGREGION TEXT STREAM))
        (with REGION BLANCREGION (CLIPPED.BITBLT STREAMSUBREGION NIL NIL NIL STREAM LEFT BOTTOM WIDTH
                                                HEIGHT 'TEXTURE))))]
```

(EXTENTOFCOMPOUND

[LAMBDA (COMPOUNDOBJECT) ; Edited 5-May-87 17:52 by jop

```
(bind (CMINX _ MAX.FLOAT)
      (CMAXX _ MIN.FLOAT)
      (CMINY _ MAX.FLOAT)
      (CMAXY _ MIN.FLOAT)
  PEXTENT for PART in (fetch (COMPOUNDDATA COMPONENTS) of (fetch OBJECTDATA of COMPOUNDOBJECT))
  declare (TYPE FLOATING CMINX CMAXX CMINY CMAXY) do (SETQ PEXTENT (EXTENTOFPLOBJECT PART))
  (if (LESSP (fetch MINX of PEXTENT)
           CMINX)
      then (SETQ CMINX (fetch MINX of PEXTENT)))
  (if (GREATERP (fetch MAXX of PEXTENT)
           CMAXX)
      then (SETQ CMAXX (fetch MAXX of PEXTENT)))
  (if (LESSP (fetch MINY of PEXTENT)
           CMINY)
      then (SETQ CMINY (fetch MINY of PEXTENT)))
  (if (GREATERP (fetch MAXY of PEXTENT)
           CMAXY)
      then (SETQ CMAXY (fetch MAXY of PEXTENT)))

  finally (RETURN (create EXTENT
                          MINX _ CMINX
                          MAXX _ CMAXX
                          MINY _ CMINY
                          MAXY _ CMAXY]))
```

(EXTENTOFCURVE

```

[LAMBDA (CURVEOBJECT) ; Edited 5-May-87 17:52 by jop
  (bind (MINX _ MAX.FLOAT)
        (MAXX _ MIN.FLOAT)
        (MINY _ MAX.FLOAT)
        (MAXY _ MIN.FLOAT)
        X Y for POSITION in (fetch (CURVEDATA CURVEPOINTS) of (fetch OBJECTDATA of CURVEOBJECT))
  declare (TYPE FLOATING MINX MAXX MINY MAXY X Y) do
    (SETQ X (fetch XCOORD of POSITION))
    (SETQ Y (fetch YCOORD of POSITION))
    (COND
      ((FLESSP X MINX)
       (SETQ MINX X))
      (COND
        ((FGREATERP X MAXX)
         (SETQ MAXX X))
        (COND
          ((FLESSP Y MINY)
           (SETQ MINY Y))
          (COND
            ((FGREATERP Y MAXY)
             (SETQ MAXY Y))

  finally (RETURN (create EXTENT
                        MINX _ MINX
                        MAXX _ MAXX
                        MINY _ MINY
                        MAXY _ MAXY]))

```

(EXTENTOFFILLEDRECTANGLE

```

[LAMBDA (FILLEDRECTANGLE) (* edited%: "21-May-85 15:29")
  (create EXTENT
    MINX _ (fetch (FILLEDRECTANGLEDATA OBJECTLEFT) of (fetch OBJECTDATA of FILLEDRECTANGLE))
    MAXX _ (fetch (FILLEDRECTANGLEDATA OBJECTRIGHT) of (fetch OBJECTDATA of FILLEDRECTANGLE))
    MINY _ (fetch (FILLEDRECTANGLEDATA OBJECTBOTTOM) of (fetch OBJECTDATA of FILLEDRECTANGLE))
    MAXY _ (fetch (FILLEDRECTANGLEDATA OBJECTTOP) of (fetch OBJECTDATA of FILLEDRECTANGLE))

```

(EXTENTOFGRAPH

```

[LAMBDA (GRAPHOBJECT) ; Edited 5-May-87 17:53 by jop
  (create EXTENT
    MINX _ MAX.FLOAT
    MAXX _ MIN.FLOAT
    MINY _ MAX.FLOAT
    MAXY _ MIN.FLOAT])

```

(EXTENTOFFLINE

```

[LAMBDA (LINEOBJECT) (* jop%: " 5-Mar-85 14:03")
  (create EXTENT
    MINX _ MAX.FLOAT
    MAXX _ MIN.FLOAT
    MINY _ MAX.FLOAT
    MAXY _ MIN.FLOAT])

```

(EXTENTOFPOINT

```

[LAMBDA (POINT) (* edited%: "21-May-85 15:28")
  (PROG [(POSITION (fetch (POINTDATA POINTPOSITION) of (fetch OBJECTDATA of POINT))
        (RETURN (create EXTENT
                    MINX _ (fetch XCOORD of POSITION)
                    MAXX _ (fetch XCOORD of POSITION)
                    MINY _ (fetch YCOORD of POSITION)
                    MAXY _ (fetch YCOORD of POSITION))])

```

(EXTENTOFFPOLYGON

```

[LAMBDA (POLYGONOBJECT) ; Edited 5-May-87 17:53 by jop
  (bind (MINX _ MAX.FLOAT)
        (MAXX _ MIN.FLOAT)
        (MINY _ MAX.FLOAT)
        (MAXY _ MIN.FLOAT)
        X Y for POSITION in (fetch POLYGONPOINTS of (fetch OBJECTDATA of POLYGONOBJECT))
  declare (TYPE FLOATING MINX MAXX MINY MAXY X Y) do
    (SETQ X (fetch XCOORD of POSITION))
    (SETQ Y (fetch YCOORD of POSITION))
    (if (FLESSP X MINX)
      then (SETQ MINX X))
    (if (FGREATERP X MAXX)
      then (SETQ MAXX X))
    (if (FLESSP Y MINY)
      then (SETQ MINY Y))
    (if (FGREATERP Y MAXY)
      then (SETQ MAXY Y))

  finally (RETURN (create EXTENT
                        MINX _ MINX
                        MAXX _ MAXX
                        MINY _ MINY

```

MAYY _ MAYY])

(EXTENTOFTEXT

[LAMBDA (TEXTOBJECT) ; Edited 5-May-87 17:53 by jop
(PROG [(POSITION (fetch TEXTPOSITION of (fetch OBJECTDATA of TEXTOBJECT)
(RETURN (create EXTENT
MINX _ (fetch XCOORD of POSITION)
MAXX _ (fetch XCOORD of POSITION)
MINY _ (fetch YCOORD of POSITION)
MAXY _ (fetch YCOORD of POSITION])

(GETCOMPOUND

[LAMBDA (PROPLST) ; Edited 5-May-87 17:53 by jop
;; GETFN for COMPOUND objects
(create COMPOUNDDATA
COMPOUNDTYPE _ (LISTGET PROPLST 'COMPOUNDTYPE)
COMPONENTS _ (LISTGET PROPLST 'COMPONENTS])

(GETCURVE

[LAMBDA (PROPLST) ; Edited 5-May-87 17:54 by jop
;; GETFN for CURVE objects
(PROG [(STYLELST (LISTGET PROPLST 'STYLE]
(RETURN (create CURVEDATA
CURVEPOINTS _ (LISTGET PROPLST 'CURVEPOINTS)
STYLE _ (create PLOT.STYLE
LINewidth _ (CAR STYLELST)
DASHING _ (CADR STYLELST)
COLOR _ (CADDR STYLELST])

(GETFILLEDRECTANGLE

[LAMBDA (PROPLST) ; Edited 5-May-87 17:54 by jop
;; GETFN for FILLEDRECTANGLE objects
(create FILLEDRECTANGLEDATA
OBJECTLEFT _ (LISTGET PROPLST 'OBJECTLEFT)
OBJECTBOTTOM _ (LISTGET PROPLST 'OBJECTBOTTOM)
OBJECTWIDTH _ (LISTGET PROPLST 'OBJECTWIDTH)
OBJECTHEIGHT _ (LISTGET PROPLST 'OBJECTHEIGHT)
BORDERWIDTH _ (LISTGET PROPLST 'BORDERWIDTH)
TEXTURE _ (LISTGET PROPLST 'TEXTURE])

(GETGENERIC

[LAMBDA (EXPR) (* jop%: "27-Aug-85 17:11")
EXPR])

(GETGRAPH

[LAMBDA (PROPLST) ; Edited 5-May-87 17:54 by jop
(PROG [(STYLELST (LISTGET PROPLST 'STYLE]
(RETURN (create GRAPHDATA
GRAPHFN _ (LISTGET PROPLST 'GRAPHFN)
NSAMPLES _ (LISTGET PROPLST 'NSAMPLES)
STYLE _ (create PLOT.STYLE
LINewidth _ (CAR STYLELST)
DASHING _ (CADR STYLELST)
COLOR _ (CADDR STYLELST])

(GETLINE

[LAMBDA (PROPLST) ; Edited 5-May-87 17:54 by jop
;; GETFN for LINE objects
(PROG [(STYLELST (LISTGET PROPLST 'STYLE]
(RETURN (create LINEDATA
INFINITESLOPE? _ (LISTGET PROPLST 'INFINITESLOPE?)
SLOPE _ (LISTGET PROPLST 'SLOPE)
CONSTANT _ (LISTGET PROPLST 'CONSTANT)
STYLE _ (create PLOT.STYLE
LINewidth _ (CAR STYLELST)
DASHING _ (CADR STYLELST)
COLOR _ (CADDR STYLELST])

(GETPOINT

[LAMBDA (PROPLST) ; Edited 5-May-87 17:54 by jop
;; Putfn for POINT objects
(create POINTDATA
POINTPOSITION _ (LISTGET PROPLST 'POINTPOSITION)
SYMBOL _ (LET [(SYMBOL (LISTGET PROPLST 'SYMBOL]

```
(if (LITATOM SYMBOL)
    then (EVAL SYMBOL)
    else SYMBOL])
```

(GETPOLYGON

```
[LAMBDA (PROPLST) ; Edited 5-May-87 17:55 by jop
  ;; GETFN for POLYGON objects
  (PROG [(STYLELST (LISTGET PROPLST 'STYLE)
    (RETURN (create POLYGONDATA
      POLYGONPOINTS _ (LISTGET PROPLST 'POLYGONPOINTS)
      STYLE _ (create PLOT.STYLE
        LINETHICKNESS _ (CAR STYLELST)
        DASHING _ (CADR STYLELST)
        COLOR _ (CADDR STYLELST))
```

(GETTEXT

```
[LAMBDA (PROPLST) ; Edited 5-May-87 17:55 by jop
  ;; GETFN for TEXT objects
  (create TEXTDATA
    TEXTPOSITION _ (LISTGET PROPLST 'TEXTPOSITION)
    TEXT _ (LISTGET PROPLST 'TEXT)
    FONT _ (LISTGET PROPLST 'FONT])
```

(HIGHLIGHTCOMPOUND

```
[LAMBDA (COMPOUNDOBJECT VIEWPORT PLOT) (* edited%: "27-Mar-86 21:26")
  (for OBJECT in (fetch (COMPOUNDDATA COMPONENTS) of (fetch (PLOT OBJECT OBJECTDATA) of COMPOUNDOBJECT))
    do (HIGHLIGHTPLOT OBJECT OBJECT PLOT])
```

(HIGHLIGHTCURVE

```
[LAMBDA (CURVEOBJECT VIEWPORT PLOT) ; Edited 5-May-87 17:55 by jop
  ;; Highlight the CURVEOBJECT, by redrawing in invert mode with fatter lines
  (LET* ((STREAM (fetch (VIEWPORT PARENTSTREAM) of VIEWPORT))
    (STREAMSUBREGION (fetch (VIEWPORT STREAMSUBREGION) of VIEWPORT))
    (OBJECTDATA (fetch (PLOT OBJECT OBJECTDATA) of CURVEOBJECT))
    (STREAMPOINTS (fetch (CURVEDATA STREAMPOINTS) of OBJECTDATA))
    (STYLE (fetch (CURVEDATA STYLE) of OBJECTDATA))
    (LINETHICKNESS (IPLUS 2 (fetch (PLOT.STYLE LINETHICKNESS) of STYLE)))
    (COLOR (fetch (PLOT.STYLE COLOR) of STYLE)))
    (first (MOVETO (fetch XCOORD of (CAR STREAMPOINTS))
      (fetch YCOORD of (CAR STREAMPOINTS))
      STREAM)
    for PT in (CDR STREAMPOINTS) do (CLIPPED.DRAWTO STREAMSUBREGION (fetch XCOORD of PT)
      (fetch YCOORD of PT)
      LINETHICKNESS
      'INVERT STREAM COLOR))
```

(HIGHLIGHTFILLEDRECTANGLE

```
[LAMBDA (FILLEDRECTANGLE VIEWPORT PLOT) ; Edited 5-May-87 17:55 by jop
  (LET* ((STREAM (fetch (VIEWPORT PARENTSTREAM) of VIEWPORT))
    (STREAMSUBREGION (fetch (VIEWPORT STREAMSUBREGION) of VIEWPORT))
    (OBJECTDATA (fetch (PLOT OBJECT OBJECTDATA) of FILLEDRECTANGLE))
    (BORDERWIDTH (IPLUS 2 (OR (fetch (FILLEDRECTANGLEDATA BORDERWIDTH) of OBJECTDATA)
      1)))
    (STREAMLEFT (fetch (FILLEDRECTANGLEDATA STREAMLEFT) of OBJECTDATA))
    (STREAMBOTTOM (fetch (FILLEDRECTANGLEDATA STREAMBOTTOM) of OBJECTDATA))
    (STREAMWIDTH (fetch (FILLEDRECTANGLEDATA STREAMWIDTH) of OBJECTDATA))
    (STREAMHEIGHT (fetch (FILLEDRECTANGLEDATA STREAMHEIGHT) of OBJECTDATA))
    (STREAMRIGHT (fetch (FILLEDRECTANGLEDATA STREAMRIGHT) of OBJECTDATA))
    (STREAMTOP (fetch (FILLEDRECTANGLEDATA STREAMTOP) of OBJECTDATA)))
    (CLIPPED.BITBLT STREAMSUBREGION NIL NIL NIL STREAM STREAMLEFT STREAMBOTTOM STREAMWIDTH STREAMHEIGHT
      'TEXTURE
      'INVERT BLACKSHADE)
    (MOVETO STREAMLEFT STREAMBOTTOM STREAM)
    (CLIPPED.DRAWTO STREAMSUBREGION STREAMRIGHT STREAMBOTTOM BORDERWIDTH 'INVERT STREAM)
    (CLIPPED.DRAWTO STREAMSUBREGION STREAMRIGHT STREAMTOP BORDERWIDTH 'INVERT STREAM)
    (CLIPPED.DRAWTO STREAMSUBREGION STREAMLEFT STREAMTOP BORDERWIDTH 'INVERT STREAM)
    (CLIPPED.DRAWTO STREAMSUBREGION STREAMLEFT STREAMBOTTOM BORDERWIDTH 'INVERT STREAM])
```

(HIGHLIGHTGRAPH

```
[LAMBDA (GRAPHOBJECT VIEWPORT PLOT) ; Edited 5-May-87 17:55 by jop
  (LET* ((STREAM (fetch (VIEWPORT PARENTSTREAM) of VIEWPORT))
    (STREAMSUBREGION (fetch (VIEWPORT STREAMSUBREGION) of VIEWPORT))
    (OBJECTDATA (fetch (PLOT OBJECT OBJECTDATA) of GRAPHOBJECT))
    (STREAMPOSITIONS (fetch (GRAPHDATA STREAMPOSITIONS) of OBJECTDATA))
    (STYLE (fetch (GRAPHDATA STYLE) of OBJECTDATA))
    (LINETHICKNESS (IPLUS 2 (fetch (PLOT.STYLE LINETHICKNESS) of STYLE)))
    (COLOR (fetch (PLOT.STYLE COLOR) of STYLE)))
    (first (MOVETO (fetch XCOORD of (CAR STREAMPOSITIONS))
```

```

      (fetch YCOORD of (CAR STREAMPOSITIONS))
      STREAM)
  for PT in (CDR STREAMPOSITIONS) do (CLIPPED.DRAWTO STREAMSUBREGION (fetch XCOORD of PT)
                                     (fetch YCOORD of PT)
                                     LINEWIDTH
                                     'INVERT STREAM COLOR])

```

(HIGHLIGHTLINE

```

[LAMBDA (LINEOBJECT VIEWPORT PLOT) ; Edited 5-May-87 17:55 by jop
  (LET* ((STREAM (fetch (VIEWPORT PARENTSTREAM) of VIEWPORT))
         (STREAMSUBREGION (fetch (VIEWPORT STREAMSUBREGION) of VIEWPORT))
         (OBJECTDATA (fetch (PLOT OBJECT OBJECTDATA) of LINEOBJECT))
         (STYLE (fetch (LINEDATA STYLE) of OBJECTDATA))
         (LINEWIDTH (IPLUS (fetch (PLOT.STYLE LINEWIDTH) of STYLE)
                             2))
         (COLOR (fetch (PLOT.STYLE COLOR) of STYLE))
         (STREAMPT1 (fetch (LINEDATA STREAMPT1) of OBJECTDATA))
         (STREAMPT2 (fetch (LINEDATA STREAMPT2) of OBJECTDATA)))
    (CLIPPED.DRAWBETWEEN STREAMSUBREGION STREAMPT1 STREAMPT2 LINEWIDTH 'INVERT STREAM COLOR])

```

(HIGHLIGHTPOINT

```

[LAMBDA (POINT VIEWPORT PLOT) ; Edited 5-May-87 17:56 by jop
  ;; Highlight POINT
  (LET* [(STREAM (fetch (VIEWPORT PARENTSTREAM) of VIEWPORT))
         (STREAMSUBREGION (fetch (VIEWPORT STREAMSUBREGION) of VIEWPORT))
         (OBJECTDATA (fetch (PLOT OBJECT OBJECTDATA) of POINT))
         (SYMBOL (fetch (POINTDATA SYMBOL) of OBJECTDATA))
         (STREAMPT (fetch (POINTDATA STREAMPOSITION) of OBJECTDATA))
         (WIDTHGLYPH (BITMAPWIDTH SYMBOL))
         (HEIGHTGLYPH (BITMAPHEIGHT SYMBOL))
         (OFFSETX (IDIFFERENCE (fetch XCOORD of STREAMPT)
                               (IQUOTIENT WIDTHGLYPH 2)))
         (OFFSETY (IDIFFERENCE (fetch YCOORD of STREAMPT)
                               (IQUOTIENT HEIGHTGLYPH 2))]
    (CLIPPED.BITBLT STREAMSUBREGION NIL NIL NIL STREAM OFFSETX OFFSETY WIDTHGLYPH HEIGHTGLYPH 'TEXTURE
                    'INVERT BLACKSHADE])

```

(HIGHLIGHTPOLYGON

```

[LAMBDA (POLYGONOBJECT VIEWPORT PLOT) ; Edited 5-May-87 17:56 by jop
  ;; Highlight a Polygon
  (LET* ((STREAM (fetch (VIEWPORT PARENTSTREAM) of VIEWPORT))
         (STREAMSUBREGION (fetch (VIEWPORT STREAMSUBREGION) of VIEWPORT))
         (OBJECTDATA (fetch (PLOT OBJECT OBJECTDATA) of POLYGONOBJECT))
         (STREAMPOINTS (fetch (POLYGONDATA STREAMPOINTS) of OBJECTDATA))
         (STYLE (fetch (POLYGONDATA STYLE) of OBJECTDATA))
         (LINEWIDTH (IPLUS 2 (fetch (PLOT.STYLE LINEWIDTH) of STYLE)))
         (COLOR (fetch (PLOT.STYLE COLOR) of STYLE)))
    (bind (START _ (CAR STREAMPOINTS)) first (MOVETO (fetch XCOORD of START)
                                                      (fetch YCOORD of START)
                                                      STREAM)
          for PT in (CDR STREAMPOINTS) do (CLIPPED.DRAWTO STREAMSUBREGION (fetch XCOORD of PT)
                                                                            (fetch YCOORD of PT)
                                                                            LINEWIDTH
                                                                            'INVERT STREAM COLOR)
          finally (CLIPPED.DRAWTO STREAMSUBREGION (fetch XCOORD of START)
                                                    (fetch YCOORD of START)
                                                    LINEWIDTH
                                                    'INVERT STREAM COLOR])

```

(HIGHLIGHTTEXT

```

[LAMBDA (TEXTOBJECT VIEWPORT PLOT) ; Edited 5-May-87 17:56 by jop
  ;; HIGHLIGHT the TEXTDATA
  (LET* ((STREAM (fetch (VIEWPORT PARENTSTREAM) of VIEWPORT))
         (STREAMSUBREGION (fetch (VIEWPORT STREAMSUBREGION) of VIEWPORT))
         (OBJECTDATA (fetch (PLOT OBJECT OBJECTDATA) of TEXTOBJECT))
         (TEXT (fetch (TEXTDATA TEXT) of OBJECTDATA))
         (FONT (fetch (TEXTDATA FONT) of OBJECTDATA))
         (STREAMPOSITION (fetch (TEXTDATA STREAMPOSITION) of OBJECTDATA))
         (STREAMX (fetch XCOORD of STREAMPOSITION))
         (STREAMY (fetch YCOORD of STREAMPOSITION))
         BLANCREGION)
    (RESETLST
     (RESETSAVE (DSPFONT FONT STREAM)
                (LIST 'DSPFONT (DSPFONT NIL STREAM)
                      STREAM))
     (MOVETO STREAMX STREAMY STREAM)
     (SETQ BLANCREGION (STRINGREGION TEXT STREAM))
     (with REGION BLANCREGION (CLIPPED.BITBLT STREAMSUBREGION NIL NIL NIL STREAM LEFT BOTTOM WIDTH
                                              HEIGHT 'TEXTURE 'INVERT BLACKSHADE))))])

```

(LABELGENERIC

```
[LAMBDA (OBJECT PLOT) ; Edited 5-May-87 17:56 by jop
;; Generic label routine. Intended for interactive use only
(PROG ((LABEL (fetch OBJECTLABEL of OBJECT))
(VIEWPORT (fetch PLOTWINDOWVIEWPORT of PLOT))
(TEXTOBJECT (PLOTOBJECTPROP OBJECT 'LABEL))
LABELPOSITION)
(COND
(TEXTOBJECT (DRAWPLOTOBJECT TEXTOBJECT VIEWPORT PLOT))
(T (PLOTSPROMPT (CONCAT "SELECT A POSITION FOR LABEL " LABEL)
PLOT)
(SETQ LABELPOSITION (STREAMTOWORLD (GETPOSITION (fetch PLOTWINDOW of PLOT))
VIEWPORT))
(SETQ TEXTOBJECT (CREATETEXT LABELPOSITION LABEL NIL SMALLPLOTFONT))
(DRAWPLOTOBJECT TEXTOBJECT VIEWPORT PLOT)
(PLOTOBJECTPROP OBJECT 'LABEL TEXTOBJECT])
```

(LABELPOINT

```
[LAMBDA (POINT PLOT) ; Edited 5-May-87 17:56 by jop
;; Label a POINT
(PROG ((OBJECTDATA (fetch (PLOTOBJECT OBJECTDATA) of POINT))
(VIEWPORT (fetch (PLOT PLOTWINDOWVIEWPORT) of PLOT))
(LABEL (fetch (PLOTOBJECT OBJECTLABEL) of POINT))
(TEXTOBJECT (PLOTOBJECTPROP POINT 'LABEL))
SYMBOL LABELPOSITION)
(SETQ LABELPOSITION (create POSITION using (fetch (POINTDATA POINTPOSITION) of OBJECTDATA)))
(SETQ SYMBOL (fetch (POINTDATA SYMBOL) of OBJECTDATA)) ; Displace Label to right of point object
(if TEXTOBJECT
then (DRAWPLOTOBJECT TEXTOBJECT VIEWPORT PLOT)
else [replace XCOORD of LABELPOSITION with (PLUS (fetch XCOORD of LABELPOSITION)
(TIMES 2 (STREAMTOWORLDDXLENGTH (BITMAPWIDTH SYMBOL)
VIEWPORT))
(SETQ TEXTOBJECT (CREATETEXT LABELPOSITION LABEL NIL SMALLPLOTFONT))
(DRAWPLOTOBJECT TEXTOBJECT VIEWPORT PLOT) ; CACHE LABEL ON PROP LIST OF OBJECT
(PLOTOBJECTPROP POINT 'LABEL TEXTOBJECT])
```

(LABELTEXT

```
[LAMBDA (TEXTOBJECT PLOT) (* jop%: "20-Feb-86 17:56")
(PLOTSPROMPT "Cannot label text" PLOT)]
```

(LOWLIGHTCOMPOUND

```
[LAMBDA (COMPOUNDOBJECT VIEWPORT PLOT) (* edited%: "27-Mar-86 21:27")
(for OBJECT in (fetch (COMPOUNDDATA COMPONENTS) of (fetch (PLOTOBJECT OBJECTDATA) of COMPOUNDOBJECT))
do (LOWLIGHTPLOTOBJECT OBJECT PLOT))
```

(MOVECOMPOUND

```
[LAMBDA (COMPOUNDOBJECT DX DY PLOT) (* edited%: "27-Mar-86 21:27")
(for OBJECT in (fetch (COMPOUNDDATA COMPONENTS) of (fetch OBJECTDATA of COMPOUNDOBJECT))
do (MOVEPLOTOBJECT OBJECT DX DY PLOT))
```

(MOVECURVE

```
[LAMBDA (CURVEOBJECT DX DY PLOT) (* jop%: "8-Dec-85 18:35")
(PROG [(POINTS (fetch (CURVEDATA CURVEPOINTS) of (fetch OBJECTDATA of CURVEOBJECT))
(for POINT in POINTS do (replace XCOORD of POINT with (PLUS DX (fetch XCOORD of POINT)))
(replace YCOORD of POINT with (PLUS DY (fetch YCOORD of POINT))
```

(MOVEFILLEDRECTANGLE

```
[LAMBDA (FILLEDRECTANGLEOBJECT DX DY PLOT) (* edited%: "18-May-85 16:32")
(PROG ((OBJECTDATA (fetch OBJECTDATA of FILLEDRECTANGLEOBJECT))
(replace OBJECTLEFT of OBJECTDATA with (PLUS DX (fetch OBJECTLEFT of OBJECTDATA)))
(replace OBJECTBOTTOM of OBJECTDATA with (PLUS DY (fetch OBJECTBOTTOM of OBJECTDATA))
```

(MOVELINE

```
[LAMBDA (LINEOBJECT DX DY PLOT) ; Edited 4-Nov-93 14:59 by rmk:
(* edited%: "18-May-85 16:58")
(PROG ((OBJECTDATA (fetch OBJECTDATA of LINEOBJECT)))
(replace (LINEDATA CONSTANT) of OBJECTDATA
with (if (fetch INFINITESLOPE? of OBJECTDATA)
then (PLUS DX (fetch (LINEDATA CONSTANT) of OBJECTDATA))
else (DIFFERENCE (PLUS (fetch (LINEDATA CONSTANT) of OBJECTDATA)
(TIMES DX (fetch SLOPE of OBJECTDATA)))
DY]))
```

(MOVEPOINT

```
[LAMBDA (POINT DX DY PLOT) (* jop%: "24-Feb-86 14:43")
(PROG [(POSITION (fetch (POINTDATA POINTPOSITION) of (fetch (PLOTOBJECT OBJECTDATA) of POINT)
```

(replace XCOORD of POSITION with (PLUS DX (fetch XCOORD of POSITION)))
(replace YCOORD of POSITION with (PLUS DY (fetch YCOORD of POSITION)))

(MOVEPOLYGON

[LAMBDA (POLYGONOBJECT DX DY PLOT) (* edited%: "18-May-85 16:16")
(PROG [(POINTS (fetch POLYGONPOINTS of (fetch OBJECTDATA of POLYGONOBJECT)
(for POINT in POINTS do (replace XCOORD of POINT with (PLUS DX (fetch XCOORD of POINT)))
(replace YCOORD of POINT with (PLUS DY (fetch YCOORD of POINT)))]

(MOVETEXT

[LAMBDA (TEXTOBJECT DX DY PLOT) (* edited%: "18-May-85 17:05")
(PROG [(POSITION (fetch TEXTPOSITION of (fetch OBJECTDATA of TEXTOBJECT)
(replace XCOORD of POSITION with (PLUS DX (fetch XCOORD of POSITION)))
(replace YCOORD of POSITION with (PLUS DY (fetch YCOORD of POSITION)))]

(PLOTCOMPOUND

[LAMBDA ARGS ; Edited 5-May-87 17:57 by jop
;; ADD A COMPOUND OBJECT with an unknown number of COMPONENTS. First arg must be a PLOT. Second arg must be the compound
;; object type. Next are the Nospread COMPONENTS, then the optional LABEL, MENU, and NODRAWFLG
(if (LESSP ARGS 3)
then (HELP "Must have at least 3 args. Plot, compound type, and one component"))
(PROG ((PLOT (ARG ARGS 1))
(COMPOUNDTYPE (ARG ARGS 2))
COMPONENTS STARTRESTARGS)
(if (NOT (type? PLOT PLOT))
then (HELP "NOT a PLOT " PLOT))
(SETQ COMPONENTS (for I from 3 to ARGS while (type? PLOT OBJECT (ARG ARGS I))
collect (ARG ARGS I)))
(SETQ STARTRESTARGS (PLUS 3 (LENGTH COMPONENTS)))
(RETURN (ADDPLOT OBJECT [CREATECOMPOUND COMPOUNDTYPE COMPONENTS (if (GEQ ARGS STARTRESTARGS)
then (ARG ARGS STARTRESTARGS))
(if (GEQ ARGS (PLUS 1 STARTRESTARGS))
then (ARG ARGS (PLUS 1 STARTRESTARGS))
PLOT
(if (GEQ ARGS (PLUS 2 STARTRESTARGS))
then (ARG ARGS (PLUS 2 STARTRESTARGS))

(PLOTCURVE

[LAMBDA (PLOT POSITIONS LABEL STYLE MENU NODRAWFLG) ; Edited 5-May-87 17:57 by jop
;; User Entry Point. Draw a piecewise linear curve in a Plotting WINDOW. Style is either the line width to use or a list (width dashing color) or an
;; instance of PLOT.STYLE. POSITIONS is a list of positions to be connected.
(COND
((NOT (type? PLOT PLOT))
(HELP "NOT a PLOT " PLOT))
(ADDPLOT OBJECT (CREATECURVE POSITIONS LABEL STYLE MENU)
PLOT NODRAWFLG])

(PLOTFILLEDRECTANGLE

[LAMBDA (PLOT LEFT BOTTOM WIDTH HEIGHT LABEL TEXTURE BORDERWIDTH MENU NODRAWFLG) ; Edited 5-May-87 17:57 by jop
;; User Entry Point. Draw a FILLEDRECTANGLE in a Plotting WINDOW. Style is the line width to use.
(if (NOT (type? PLOT PLOT))
then (HELP "NOT a PLOT " PLOT))
(if (NULL TEXTURE)
then (SETQ TEXTURE SHADE3))
(ADDPLOT OBJECT (CREATEFILLEDRECTANGLE LEFT BOTTOM WIDTH HEIGHT LABEL TEXTURE BORDERWIDTH MENU)
PLOT NODRAWFLG])

(PLOTGRAPH

[LAMBDA (PLOT GRAPHFN NSAMPLES LABEL STYLE MENU NODRAWFLG) ; Edited 5-May-87 17:58 by jop
;; User Entry Point.
(if (NOT (type? PLOT PLOT))
then (HELP "NOT a PLOT " PLOT))
(ADDPLOT OBJECT (CREATEGRAPH GRAPHFN NSAMPLES LABEL STYLE MENU)
PLOT NODRAWFLG])

(PLOTLINE

[LAMBDA (PLOT SLOPE CONSTANT LABEL STYLE MENU NODRAWFLG) ; Edited 5-May-87 17:58 by jop
;; User Entry Point.
(COND
((NOT (type? PLOT PLOT))
(HELP "NOT a PLOT " PLOT))
(ADDPLOT OBJECT (CREATELINE SLOPE CONSTANT LABEL STYLE MENU)
PLOT NODRAWFLG])

(PLOTPOINT

```
[LAMBDA (PLOT POSITION LABEL SYMBOL MENU NODRAWFLG) ; Edited 5-May-87 17:58 by jop
;; User entry point. Add a point to the plotwindow WINDOW, at world position POSITION, with Label LABEL and plotting symbol SYMBOL
(if (NOT (type? PLOT PLOT))
    then (HELP "NOT a PLOT " PLOT))
(ADDPLOT OBJECT (CREATEPOINT POSITION LABEL SYMBOL MENU)
    PLOT NODRAWFLG])
```

(PLOTPOINTS

```
[LAMBDA (PLOT POSITIONS LABELS SYMBOL MENU NODRAWFLG) ; Edited 5-May-87 17:58 by jop
;; User Entry Point. Draw the POINTs at POSITIONS in a Plotting WINDOW. Symbol is a LITATOM which Describes the glyph to use.
(if (NOT (type? PLOT PLOT))
    then (HELP "NOT a PLOT " PLOT))
(PROG (EXTENT NEWSCALES OBJECTS)
    [SETQ EXTENT
    (bind (MINX _ MAX.FLOAT)
        (MAXX _ MIN.FLOAT)
        (MINY _ MAX.FLOAT)
        (MAXY _ MIN.FLOAT)
        for PT in POSITIONS do (if (LESSP (fetch XCOORD of PT)
            MINX)
                then (SETQ MINX (fetch XCOORD of PT)))
            (if (GREATERP (fetch XCOORD of PT)
                MAXX)
                then (SETQ MAXX (fetch XCOORD of PT)))
            (if (LESSP (fetch YCOORD of PT)
                MINY)
                then (SETQ MINY (fetch YCOORD of PT)))
            (if (GREATERP (fetch YCOORD of PT)
                MAXY)
                then (SETQ MAXY (fetch YCOORD of PT))))
        finally (RETURN (create EXTENT
            MINX _ MINX
            MAXX _ MAXX
            MINY _ MINY
            MAXY _ MAXY)
            (ADJUSTSCALE? EXTENT PLOT) ; Scale up the plot so that each ADDOBJECT need not rescale
            [SETQ OBJECTS (bind (LABEL _ LABELS) for POSITION in POSITIONS collect (PROG1 (CREATEPOINT POSITION
                (CAR LABEL)
                SYMBOL MENU)
                (SETQ LABEL (CDR LABEL)))]
            ; Do surgery on the display list
            (replace (PLOT PLOT OBJECTS) of PLOT with (APPEND OBJECTS (fetch (PLOT PLOT OBJECTS) of PLOT)))
            (if (NULL NODRAWFLG)
                then (REDRAWPLOTWINDOW PLOT))
            (RETURN OBJECTS])
```

(PLOT POLYGON

```
[LAMBDA (PLOT POSITIONS LABEL STYLE MENU NODRAWFLG) ; Edited 5-May-87 17:58 by jop
;; User Entry Point. Draw a POLYGON in a Plotting WINDOW. Style is the line width to use. POSITIONS is a list of positions to be connected.
(if (NOT (type? PLOT PLOT))
    then (HELP "NOT a PLOT " PLOT))
(ADDPLOT OBJECT (CREATEPOLYGON POSITIONS LABEL STYLE MENU)
    PLOT NODRAWFLG])
```

(PLOTTEXT

```
[LAMBDA (PLOT POSITION TEXT LABEL FONT MENU NODRAWFLG) (* edited%: "27-Mar-86 21:22")
(COND
    ((NOT (type? PLOT PLOT))
        (HELP "NOT A PLOT " PLOT))
    (COND
        ((NULL FONT)
            (SETQ FONT SMALLPLOTFONT))
        (ADDPLOT OBJECT (CREATETEXT POSITION TEXT LABEL FONT MENU)
            PLOT NODRAWFLG])
```

(PUTCOMPOUND

```
[LAMBDA (PLOT OBJECT PLOT STREAM) ; Edited 5-May-87 17:59 by jop
;; PUTFN for COMPOUND objects
(PROG ((OBJECTDATA (fetch (PLOT OBJECT OBJECTDATA) of PLOT OBJECT))
    (PRINTOUT STREAM "(" % ,)
    (PRINTOUT STREAM "COMPOUNDTYPE" % , .P2 (fetch (COMPOUNDDATA COMPOUNDTYPE) of OBJECTDATA)
        % ,)
    (PRINTOUT STREAM "COMPONENTS (" % ,) ; THIS ASSUMES APPROPRIATE HPRINT MACROS
    (for OBJECT in (fetch (COMPOUNDDATA COMPONENTS) of OBJECTDATA) do (HPRINT OBJECT STREAM T))
    (PRINTOUT STREAM ")"))]
```

(PUTCURVE


```
[LAMBDA (PLOT OBJECT PLOT STREAM) ; Edited 5-May-87 17:59 by jop
;; Putfn for CURVE objects
(PROG ((OBJECTDATA (fetch (PLOT OBJECT OBJECTDATA) of PLOT OBJECT))
STYLE)
(SETQ STYLE (fetch (CURVEDATA STYLE) of OBJECTDATA))
(PRINTOUT STREAM "(" % "CURVEPOINTS" % .P2 (fetch (CURVEDATA CURVEPOINTS) of OBJECTDATA)
% "STYLE" % .P2 (LIST (fetch (PLOT.STYLE LINEWIDTH) of STYLE)
(fetch (PLOT.STYLE DASHING) of STYLE)
(fetch (PLOT.STYLE COLOR) of STYLE))
% " " ]])
```

(PUTFILLEDRECTANGLE

```
[LAMBDA (PLOT OBJECT PLOT STREAM) ; Edited 5-May-87 17:59 by jop
;; PUTFN for FILLEDRECTANGLE objects
(PROG ((OBJECTDATA (fetch (PLOT OBJECT OBJECTDATA) of PLOT OBJECT))
(PRINTOUT STREAM "(" % ,)
(PRINTOUT STREAM "OBJECTLEFT" % .P2 (fetch (FILLEDRECTANGLEDATA OBJECTLEFT) of OBJECTDATA)
% ,)
(PRINTOUT STREAM "OBJECTBOTTOM" % .P2 (fetch (FILLEDRECTANGLEDATA OBJECTBOTTOM) of OBJECTDATA)
% ,)
(PRINTOUT STREAM "OBJECTWIDTH" % .P2 (fetch (FILLEDRECTANGLEDATA OBJECTWIDTH) of OBJECTDATA)
% ,)
(PRINTOUT STREAM "OBJECTHEIGHT" % .P2 (fetch (FILLEDRECTANGLEDATA OBJECTHEIGHT) of OBJECTDATA)
% ,)
(PRINTOUT STREAM "BORDERWIDTH" % .P2 (fetch (FILLEDRECTANGLEDATA BORDERWIDTH) of OBJECTDATA)
% ,)
(PRINTOUT STREAM "TEXTURE" % .P2 (fetch (FILLEDRECTANGLEDATA TEXTURE) of OBJECTDATA)
% ,)
(PRINTOUT STREAM " " ]])
```

(PUTGENERIC

```
[LAMBDA (OBJECT PLOT STREAM) (* jop%: "27-Aug-85 17:10")
(HPRINT (fetch OBJECTDATA of OBJECT)
STREAM NIL T])
```

(PUTGRAPH

```
[LAMBDA (PLOT OBJECT PLOT STREAM) ; Edited 5-May-87 17:59 by jop
;; Putfn for CURVE objects
(PROG ((OBJECTDATA (fetch (PLOT OBJECT OBJECTDATA) of PLOT OBJECT))
STYLE)
(SETQ STYLE (fetch (GRAPHDATA STYLE) of OBJECTDATA))
(PRINTOUT STREAM "(" % "GRAPHFN" % .P2 (fetch (GRAPHDATA GRAPHFN) of OBJECTDATA)
% "NSAMPLES" % .P2 (fetch (GRAPHDATA NSAMPLES) of OBJECTDATA)
% "STYLE" % .P2 (LIST (fetch (PLOT.STYLE LINEWIDTH) of STYLE)
(fetch (PLOT.STYLE DASHING) of STYLE)
(fetch (PLOT.STYLE COLOR) of STYLE))
% " " ]])
```

(PUTLINE

```
[LAMBDA (PLOT OBJECT PLOT STREAM) ; Edited 5-May-87 17:59 by jop
;; Putfn for LINE objects
(PROG ((OBJECTDATA (fetch (PLOT OBJECT OBJECTDATA) of PLOT OBJECT))
STYLE)
(SETQ STYLE (fetch (LINEDATA STYLE) of OBJECTDATA))
(PRINTOUT STREAM "(" % "INFINITESLOPE?" % .P2 (fetch (LINEDATA INFINITESLOPE?) of OBJECTDATA)
% "SLOPE" % .P2 (fetch (LINEDATA SLOPE) of OBJECTDATA)
% "CONSTANT" % .P2 (fetch (LINEDATA CONSTANT) of OBJECTDATA)
% "STYLE" % .P2 (LIST (fetch (PLOT.STYLE LINEWIDTH) of STYLE)
(fetch (PLOT.STYLE DASHING) of STYLE)
(fetch (PLOT.STYLE COLOR) of STYLE))
" " ]])
```

(PUTPOINT

```
[LAMBDA (PLOT OBJECT PLOT STREAM) ; Edited 5-May-87 18:00 by jop
;; Putfn for POINT objects
(PROG ((OBJECTDATA (fetch (PLOT OBJECT OBJECTDATA) of PLOT OBJECT))
SYMBOL LAB)
(SETQ SYMBOL (fetch (POINTDATA SYMBOL) of OBJECTDATA))
(SETQ LAB (if (EQ SYMBOL STAR)
then 'STAR
elseif (EQ SYMBOL CROSS)
then 'CROSS
elseif (EQ SYMBOL CIRCLE)
then 'CIRCLE))
(PRINTOUT STREAM "(" % "POINTPOSITION" % .P2 (fetch (POINTDATA POINTPOSITION) of OBJECTDATA)
% "SYMBOL" % ,)
(if LAB
```

```

    then (PRINTOUT STREAM .P2 LAB %,)
    else (HPRINT SYMBOL STREAM T T)
(PRINTOUT STREAM " ")

```

(PUTPOLYGON

[LAMBDA (PLOT OBJECT PLOT STREAM)

; Edited 5-May-87 18:00 by jop

;; Putfn for POLYGON objects

```

(PROG ((OBJECTDATA (fetch (PLOT OBJECT OBJECTDATA) of PLOT OBJECT))
      STYLE)
      (SETQ STYLE (fetch (POLYGONDATA STYLE) of OBJECTDATA))
      (PRINTOUT STREAM "(" % "POLYGONPOINTS" % .P2 (fetch (POLYGONDATA POLYGONPOINTS) of OBJECTDATA)
        % "STYLE" % .P2 (LIST (fetch (PLOT.STYLE LINewidth) of STYLE)
                              (fetch (PLOT.STYLE DASHING) of STYLE)
                              (fetch (PLOT.STYLE COLOR) of STYLE))
        % " ")

```

(PUTTEXT

[LAMBDA (PLOT OBJECT PLOT STREAM)

; Edited 5-May-87 18:00 by jop

;; Putfn for TEXT objects

```

(PROG ((OBJECTDATA (fetch (PLOT OBJECT OBJECTDATA) of PLOT OBJECT))
      FONT)
      (SETQ FONT (fetch (TEXTDATA FONT) of OBJECTDATA))
      (PRINTOUT STREAM "(" % "TEXTPOSITION" % .P2 (fetch (TEXTDATA TEXTPOSITION) of OBJECTDATA)
        % "TEXT" % .P2 (fetch (TEXTDATA TEXT) of OBJECTDATA)
        % "FONT" %)
      (HPRINT FONT STREAM T T)
      (PRINTOUT STREAM " ")

```

; Assumes FONT has an HPRINTMACRO

)

(DECLARE%: EVAL@COMPILE

(PUTPROPS **L1METRIC MACRO** [OPENLAMBDA (POINT1 POINT2) (* jop%: "17-Jan-85 15:27")

;; Computes the L 1 metric between POINT1 and POINT2

```

(PLUS (IABS (DIFFERENCE (fetch XCOORD of POINT1)
                        (fetch XCOORD of POINT2)))
      (IABS (DIFFERENCE (fetch YCOORD of POINT1)
                        (fetch YCOORD of POINT2])))

```

(PUTPROPS **L2METRIC MACRO** [OPENLAMBDA (POINT1 POINT2 PLOT) (* jop%: "17-Jan-85 15:27")

;; Computes the L 2 metric between POINT1 and POINT2

```

(FPLUS (FTIMES (FDIFFERENCE (fetch XCOORD of POINT1)
                             (fetch XCOORD of POINT2))
          (FDIFFERENCE (fetch XCOORD of POINT1)
                        (fetch XCOORD of POINT2)))
        (FTIMES (FTIMES (fetch NORMCONSTANT of PLOT)
                        (FDIFFERENCE (fetch YCOORD of POINT1)
                                       (fetch YCOORD of POINT2)))
          (FTIMES (fetch NORMCONSTANT of PLOT)
                  (FDIFFERENCE (fetch YCOORD of POINT1)
                                (fetch YCOORD of POINT2))))

```

)

(RPAQQ **CIRCLE** ○)

(RPAQQ **CROSS** +)

(RPAQQ **DASH** (5))

(RPAQQ **DOT** (1 5))

(RPAQQ **DOTDASH** (5 5 1 5))

(RPAQQ **SHADE1** 64)

(RPAQQ **SHADE2** 576)

(RPAQQ **SHADE3** 4680)

(RPAQQ **SHADE4** 37449)

(RPAQQ **SHADE5** 55899)

(RPAQQ **SHADE6** 31710)

(RPAQQ **SHADE7** 64479)

(RPAQQ **SHADE8** 65023)

(RPAQQ **STAR** *)

```

(DECLARE%: EVAL@COMPILE

(DATATYPE COMPOUNDDATA (COMPOUNDTYPE COMPONENTS))

(DATATYPE CURVEDATA (CURVEPOINTS STREAMPOINTS STYLE))

(DATATYPE FILLEDRECTANGLEDATA ((OBJECTLEFT FLOATING)
                                (OBJECTBOTTOM FLOATING)
                                (OBJECTWIDTH FLOATING)
                                (OBJECTHEIGHT FLOATING)
                                STREAMLEFT STREAMBOTTOM STREAMWIDTH STREAMHEIGHT BORDERWIDTH TEXTURE)
                                BORDERWIDTH _ 1 (ACCESSFNS ((OBJECTRIGHT (PLUS (fetch (FILLEDRECTANGLEDATA OBJECTLEFT) of DATUM)
                                                                              (fetch (FILLEDRECTANGLEDATA OBJECTWIDTH) of DATUM)))
                                (OBJECTTOP (PLUS (fetch (FILLEDRECTANGLEDATA OBJECTBOTTOM) of DATUM)
                                                  (fetch (FILLEDRECTANGLEDATA OBJECTHEIGHT) of DATUM)))
                                (STREAMRIGHT (PLUS (fetch (FILLEDRECTANGLEDATA STREAMLEFT) of DATUM)
                                                    (fetch (FILLEDRECTANGLEDATA STREAMWIDTH) of DATUM)))
                                (STREAMTOP (PLUS (fetch (FILLEDRECTANGLEDATA STREAMBOTTOM) of DATUM)
                                                 (fetch (FILLEDRECTANGLEDATA STREAMHEIGHT) of DATUM)]))

(DATATYPE GRAPHDATA (GRAPHFN NSAMPLES STYLE STREAMPOSITIONS))

(DATATYPE LINEDATA (STYLE INFINITESLOPE? (SLOPE FLOATING)
                    (CONSTANT FLOATING)
                    (STREAMSLOPE FLOATING)
                    (STREAMCONSTANT FLOATING)
                    STREAMPT1 STREAMPT2)
                    STYLE _ 1)

(DATATYPE PLOT.STYLE (LINewidth DASHING COLOR)
                     LINewidth _ 1)

(DATATYPE POINTDATA (POINTPOSITION STREAMPOSITION SYMBOL)
                     SYMBOL _ STAR)

(DATATYPE POLYGONDATA (POLYGONPOINTS STREAMPOINTS STYLE)
                       STYLE _ 1)

(DATATYPE TEXTDATA (TEXTPOSITION STREAMPOSITION TEXT FONT)
                    FONT _ SMALLPLOTFONT)
)

(/DECLAREDATATYPE 'COMPOUNDDATA ' (POINTER POINTER)
  ;; ---field descriptor list elided by lister---
  ' 4)

(/DECLAREDATATYPE 'CURVEDATA ' (POINTER POINTER POINTER)
  ;; ---field descriptor list elided by lister---
  ' 6)

(/DECLAREDATATYPE 'FILLEDRECTANGLEDATA ' (FLOATP FLOATP FLOATP FLOATP POINTER POINTER POINTER POINTER POINTER
                                           POINTER)
  ;; ---field descriptor list elided by lister---
  ' 20)

(/DECLAREDATATYPE 'GRAPHDATA ' (POINTER POINTER POINTER POINTER)
  ;; ---field descriptor list elided by lister---
  ' 8)

(/DECLAREDATATYPE 'LINEDATA ' (POINTER POINTER FLOATP FLOATP FLOATP FLOATP POINTER POINTER)
  ;; ---field descriptor list elided by lister---
  ' 16)

(/DECLAREDATATYPE 'PLOT.STYLE ' (POINTER POINTER POINTER)
  ;; ---field descriptor list elided by lister---
  ' 6)

(/DECLAREDATATYPE 'POINTDATA ' (POINTER POINTER POINTER)
  ;; ---field descriptor list elided by lister---
  ' 6)

(/DECLAREDATATYPE 'POLYGONDATA ' (POINTER POINTER POINTER)
  ;; ---field descriptor list elided by lister---
  ' 6)

(/DECLAREDATATYPE 'TEXTDATA ' (POINTER POINTER POINTER POINTER)
  ;; ---field descriptor list elided by lister---
  ' 8)

```

```
(PUTPROPS PLOTCOMPOUND ARGNAMES (NIL (PLOT COMPOUNDTYPE COMPONENT1 |...| LABEL MENU NODRAWFLG) . COMPOUNDARGS)
)
(DECLARE%: DONTEVAL@LOAD DOEVAL@COMPILE DONTCOPY
(FILESLoad (LOADCOMP)
  PLOT TWODGRAPHICS)
)
(DECLARE%: DONTEVAL@LOAD DOEVAL@COMPILE DONTCOPY
(DECLARE%: DOEVAL@COMPILE DONTCOPY
(LOCALVARS . T)
)
)
(DECLARE%: DONTEVAL@LOAD DOEVAL@COMPILE DONTCOPY COMPILERVERS
(ADDTovar NLAMA )
(ADDTovar NLAML )
(ADDTovar LAMA PLOTCOMPOUND)
)
(PUTPROPS PLOTOBJECTS COPYRIGHT ("Xerox Corporation" 1985 1986 1987 1993 2000))
```

FUNCTION INDEX

| | | | | | |
|---------------------------|---|----------------------------|----|---------------------|----|
| COPYCOMPOUND | 1 | DRAWTEXT OBJECT | 7 | HIGHLIGHTPOLYGON | 13 |
| COPYCURVE | 1 | ERASECOMPOUNDOBJECT | 8 | HIGHLIGHTTEXT | 13 |
| COPYFILLEDRECTANGLE | 1 | ERASECURVEOBJECT | 8 | LABELGENERIC | 14 |
| COPYGENERIC | 1 | ERASEFILLEDRECTANGLEOBJECT | 8 | LABELPOINT | 14 |
| COPYGRAPH OBJECT | 2 | ERASEGRAPH OBJECT | 8 | LABELTEXT | 14 |
| COPYLINE | 2 | ERASELINEOBJECT | 8 | LOWLIGHTCOMPOUND | 14 |
| COPYPOINT | 2 | ERASEPOINT OBJECT | 9 | MOVECOMPOUND | 14 |
| COPYPOLYGON | 2 | ERASEPOLYGON OBJECT | 9 | MOVECURVE | 14 |
| COPYTEXT | 2 | ERASETEXT OBJECT | 9 | MOVEFILLEDRECTANGLE | 14 |
| CREATECOMPOUND | 2 | EXTENTOF COMPOUND | 9 | MOVELINE | 14 |
| CREATECURVE | 2 | EXTENTOF CURVE | 10 | MOVEPOINT | 14 |
| CREATEFILLEDRECTANGLE | 2 | EXTENTOFFILLEDRECTANGLE | 10 | MOVEPOLYGON | 15 |
| CREATEGRAPH | 3 | EXTENTOF GRAPH | 10 | MOVETEXT | 15 |
| CREATELINE | 3 | EXTENTOF LINE | 10 | PLOTCOMPOUND | 15 |
| CREATEPOINT | 3 | EXTENTOF POINT | 10 | PLOT CURVE | 15 |
| CREATEPOLYGON | 3 | EXTENTOF POLYGON | 10 | PLOTFILLEDRECTANGLE | 15 |
| CREATETEXT | 3 | EXTENTOF TEXT | 11 | PLOTGRAPH | 15 |
| DISTANCETOCOMPOUND | 4 | GETCOMPOUND | 11 | PLOTLINE | 15 |
| DISTANCETOCURVE | 4 | GETCURVE | 11 | PLOTPOINT | 16 |
| DISTANCETOFILLEDRECTANGLE | 4 | GETFILLEDRECTANGLE | 11 | PLOTPOINTS | 16 |
| DISTANCETOGRAPH | 4 | GETGENERIC | 11 | PLOTPOLYGON | 16 |
| DISTANCETOLINE | 4 | GETGRAPH | 11 | PLOTTEXT | 16 |
| DISTANCETOPOINT | 5 | GETLINE | 11 | PUTCOMPOUND | 16 |
| DISTANCETOPOLYGON | 5 | GETPOINT | 11 | PUTCURVE | 16 |
| DISTANCETOTEXT | 5 | GETPOLYGON | 12 | PUTFILLEDRECTANGLE | 17 |
| DRAWCOMPOUNDOBJECT | 5 | GETTEXT | 12 | PUTGENERIC | 17 |
| DRAWCURVEOBJECT | 5 | HIGHLIGHTCOMPOUND | 12 | PUTGRAPH | 17 |
| DRAWFILLEDRECTANGLEOBJECT | 5 | HIGHLIGHTCURVE | 12 | PUTLINE | 17 |
| DRAWGRAPH OBJECT | 6 | HIGHLIGHTFILLEDRECTANGLE | 12 | PUTPOINT | 17 |
| DRAWLINEOBJECT | 6 | HIGHLIGHTGRAPH | 12 | PUTPOLYGON | 18 |
| DRAWPOINT OBJECT | 7 | HIGHLIGHTLINE | 13 | PUTTEXT | 18 |
| DRAWPOLYGON OBJECT | 7 | HIGHLIGHTPOINT | 13 | | |

RECORD INDEX

| | | | | | |
|---------------------|----|------------|----|-------------|----|
| COMPOUNDDATA | 19 | GRAPHDATA | 19 | POINTDATA | 19 |
| CURVEDATA | 19 | LINEDATA | 19 | POLYCONDATA | 19 |
| FILLEDRECTANGLEDATA | 19 | PLOT.STYLE | 19 | TEXTDATA | 19 |

VARIABLE INDEX

| | | | | | | | | | | | | | |
|--------|----|------|----|---------|----|--------|----|--------|----|--------|----|--------|----|
| CIRCLE | 18 | DASH | 18 | DOTDASH | 18 | SHADE2 | 18 | SHADE4 | 18 | SHADE6 | 18 | SHADE8 | 18 |
| CROSS | 18 | DOT | 18 | SHADE1 | 18 | SHADE3 | 18 | SHADE5 | 18 | SHADE7 | 18 | STAR | 18 |

MACRO INDEX

| | | | |
|----------|----|----------|----|
| L1METRIC | 18 | L2METRIC | 18 |
|----------|----|----------|----|

PROPERTY INDEX

| | |
|--------------|----|
| PLOTCOMPOUND | 20 |
|--------------|----|
