```
    File created:    7-Oct-88 12:26:27  {INDIGO}<GSLWS>LYRIC>LIBRARY>PLOTOBJECTS2.;6

      changes to:   (FNS CLIPPED.FILLPOLYGON CLIPPED.POLYGON FINISH-CLIP-POLYGON CLIP-POLYGON-VERTEX CLIP-INSIDEP
                         CLIP-INTERSECT GETFILLEDPOLYGON DRAWFILLEDPOLYGON CREATEFILLEDPOLYGON DISTANCETOFILLEDPOLYGON
                         ERASEFILLEDPOLYGON EXTENTOFFILLEDPOLYGON HIGHLIGHTFILLEDPOLYGON PLOTFILLEDPOLYGON
                         CLIPPED.FINDTO CLIPPED.FINDLINE COPYFILLEDPOLYGON MOVEFILLEDPOLYGON PUTFILLEDPOLYGON)
                    (VARS PLOTOBJECTS2COMS)
                    (RECORDS CLIPEDGEINFO FILLEDPOLYGONDATA)

    previous date:   5-Oct-88 11:21:10  {INDIGO}<GSLWS>LYRIC>LIBRARY>PLOTOBJECTS2.;1

      Read Table:   XCL

         Package:   INTERLISP

           Format:    XCCS
```

```
(RPAQQ PLOTOBJECTS2COMS
       ((FNS COPYFILLEDPOLYGON CREATEFILLEDPOLYGON DISTANCETOFILLEDPOLYGON DRAWFILLEDPOLYGON ERASEFILLEDPOLYGON
             EXTENTOFFILLEDPOLYGON GETFILLEDPOLYGON HIGHLIGHTFILLEDPOLYGON MOVEFILLEDPOLYGON PLOTFILLEDPOLYGON
             PUTFILLEDPOLYGON)
        (VARS OBJECT2OPSTABLE)
        (RECORDS FILLEDPOLYGONDATA)
        (P (PLOT.SETUP OBJECT2OPSTABLE))
        (FNS CLIPPED.FILLPOLYGON CLIPPED.POLYGON CLIP-POLYGON-VERTEX FINISH-CLIP-POLYGON CLIP-INSIDEP
             CLIP-INTERSECT)
        (RECORDS CLIPEDGEINFO)))

(DEFINEQ

(COPYFILLEDPOLYGON
  (LAMBDA (PLOTOBJECT PLOT)                                        ; Edited  5-Oct-88 10:23 by thh:

    ;; Copyfn for FILLEDPOLYGON objects

    (LET ((OBJECTDATA (FETCH (PLOTOBJECT OBJECTDATA) OF PLOTOBJECT)))
         (CREATE FILLEDPOLYGONDATA
                POLYGONPOINTS _ (COPYALL (FETCH (FILLEDPOLYGONDATA POLYGONPOINTS) OF OBJECTDATA))
                STYLE _ (COPYALL (FETCH (FILLEDPOLYGONDATA STYLE) OF OBJECTDATA))
                TEXTURE _ (FETCH (FILLEDPOLYGONDATA TEXTURE) OF OBJECTDATA)))))

(CREATEFILLEDPOLYGON
  (LAMBDA (POSITIONS LABEL STYLE TEXTURE MENU)                     ; Edited  5-Oct-88 12:49 by thh:
    (CREATEPLOTOBJECT FILLEDPOLYGONFNS 'FILLEDPOLYGON LABEL MENU (|create| FILLEDPOLYGONDATA
                                                                       POLYGONPOINTS _ 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)))
                                                                       TEXTURE _ TEXTURE))))

(DISTANCETOFILLEDPOLYGON
  (LAMBDA (FILLEDPOLYGON STREAMPOSITION PLOT)                      ; Edited  5-Oct-88 10:32 by thh:
    (L1METRIC STREAMPOSITION (|for| POINT |in| (|fetch| (FILLEDPOLYGONDATA STREAMPOINTS) |of| (|fetch| OBJECTDATA
                                                                                                 |of| FILLEDPOLYGON))
                                |smallest| (L1METRIC POINT STREAMPOSITION)))))

(DRAWFILLEDPOLYGON
  (LAMBDA (FILLEDPOLYGON VIEWPORT PLOT)                            ; Edited  5-Oct-88 13:05 by thh:
    (LET* ((STREAM (|fetch| (VIEWPORT PARENTSTREAM) |of| VIEWPORT))
           (STREAMSUBREGION (|fetch| (VIEWPORT STREAMSUBREGION) |of| VIEWPORT))
           (OBJECTDATA (|fetch| (PLOTOBJECT OBJECTDATA) |of| FILLEDPOLYGON))
           (POINTS (|fetch| (FILLEDPOLYGONDATA POLYGONPOINTS) |of| OBJECTDATA))
           (STREAMPOINTS (|for| PT |in| POINTS |collect| (WORLDTOSTREAM PT VIEWPORT)))
           (STYLE (|fetch| (FILLEDPOLYGONDATA 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)))
          (CLIPPED.FILLPOLYGON STREAMSUBREGION STREAMPOINTS (|fetch| (FILLEDPOLYGONDATA TEXTURE) |of| OBJECTDATA)
                 STREAM
                 'REPLACE NIL (< 0 LINEWIDTH)
                 LINEWIDTH
                 'REPLACE COLOR DASHING)
          (COND
             ((EQ STREAM (WINDOWPROP (|fetch| (PLOT PLOTWINDOW) |of| PLOT)
                                'DSP))
              (|replace| (FILLEDPOLYGONDATA STREAMPOINTS) |of| OBJECTDATA |with| STREAMPOINTS))))))

(ERASEFILLEDPOLYGON
  (LAMBDA (FILLEDPOLYGON VIEWPORT PLOT)                          ; Edited  5-Oct-88 13:05 by thh:

    ;; Erase a FILLEDPOLYGONDATA

    (LET* ((STREAM (|fetch| (VIEWPORT PARENTSTREAM) |of| VIEWPORT))
           (STREAMSUBREGION (|fetch| (VIEWPORT STREAMSUBREGION) |of| VIEWPORT))
           (OBJECTDATA (|fetch| (PLOTOBJECT OBJECTDATA) |of| FILLEDPOLYGON))
           (STREAMPOINTS (|fetch| (FILLEDPOLYGONDATA STREAMPOINTS) |of| OBJECTDATA))
           (STYLE (|fetch| (FILLEDPOLYGONDATA STYLE) |of| OBJECTDATA))
           (LINEWIDTH (IPLUS 2 (|fetch| (PLOT.STYLE LINEWIDTH) |of| STYLE)))
           (COLOR (|fetch| (PLOT.STYLE COLOR) |of| STYLE)))
          (CLIPPED.FILLPOLYGON STREAMSUBREGION STREAMPOINTS (|fetch| (FILLEDPOLYGONDATA TEXTURE) |of| OBJECTDATA)
                 STREAM
                 'ERASE NIL (< 0 (|fetch| (PLOT.STYLE LINEWIDTH) |of| STYLE))
                 LINEWIDTH
                 'ERASE COLOR))))

(EXTENTOFFILLEDPOLYGON
  (LAMBDA (FILLEDPOLYGON)                                        ; Edited  5-Oct-88 10:50 by thh:
    (|bind| (MINX _ MAX.FLOAT)
           (MAXX _ MIN.FLOAT)
           (MINY _ MAX.FLOAT)
           (MAXY _ MIN.FLOAT)
          X Y |for| POSITION |in| (|fetch| (FILLEDPOLYGONDATA POLYGONPOINTS) |of| (|fetch| OBJECTDATA |of| FILLEDPOLYGON))
       |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)))))

(GETFILLEDPOLYGON
  (LAMBDA (PROPLST)                                              ; Edited  5-Oct-88 13:22 by thh:
    (LET ((STYLELST (LISTGET PROPLST 'STYLE)))
         (|create| FILLEDPOLYGONDATA
                POLYGONPOINTS _ (LISTGET PROPLST 'POLYGONPOINTS)
                STYLE _ (|create| PLOT.STYLE
                               LINEWIDTH _ (CAR STYLELST)
                               DASHING _ (CADR STYLELST)
                               COLOR _ (CADDR STYLELST))
                TEXTURE _ (LISTGET PROPLST 'TEXTURE)))))

(HIGHLIGHTFILLEDPOLYGON
  (LAMBDA (FILLEDPOLYGON VIEWPORT PLOT)                          ; Edited  5-Oct-88 13:12 by thh:
    (LET* ((STREAM (|fetch| (VIEWPORT PARENTSTREAM) |of| VIEWPORT))
           (STREAMSUBREGION (|fetch| (VIEWPORT STREAMSUBREGION) |of| VIEWPORT))
           (OBJECTDATA (|fetch| (PLOTOBJECT OBJECTDATA) |of| FILLEDPOLYGON))
           (STREAMPOINTS (|fetch| (FILLEDPOLYGONDATA STREAMPOINTS) |of| OBJECTDATA))
           (STYLE (|fetch| (FILLEDPOLYGONDATA STYLE) |of| OBJECTDATA))
           (LINEWIDTH (IPLUS 2 (|fetch| (PLOT.STYLE LINEWIDTH) |of| STYLE)))
           (COLOR (|fetch| (PLOT.STYLE COLOR) |of| STYLE)))
          (CLIPPED.FILLPOLYGON STREAMSUBREGION STREAMPOINTS BLACKSHADE STREAM 'INVERT NIL
                 (< 0 (|fetch| (PLOT.STYLE LINEWIDTH) |of| STYLE))
                 LINEWIDTH
                 'INVERT COLOR))))

(MOVEFILLEDPOLYGON
  (LAMBDA (FILLEDPOLYGON DX DY PLOT)                             ; Edited  5-Oct-88 11:09 by thh:
    (LET ((POINTS (FETCH (FILLEDPOLYGONDATA POLYGONPOINTS) OF (FETCH OBJECTDATA OF FILLEDPOLYGON))))
         (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)))))))

(PLOTFILLEDPOLYGON
  (LAMBDA (PLOT POSITIONS LABEL STYLE TEXTURE MENU NODRAWFLG)    ; Edited  5-Oct-88 11:11 by thh:
    (COND
       ((NOT (|type?| PLOT PLOT))
        (HELP "NOT a PLOT" PLOT)))
    (ADDPLOTOBJECT (CREATEFILLEDPOLYGON POSITIONS LABEL STYLE TEXTURE MENU)
           PLOT NODRAWFLG)))

(PUTFILLEDPOLYGON
  (LAMBDA (PLOTOBJECT PLOT STREAM)                               ; Edited  5-Oct-88 11:13 by thh:
    (PROG ((OBJECTDATA (|fetch| (PLOTOBJECT OBJECTDATA) |of| PLOTOBJECT))
           STYLE)
          (SETQ STYLE (|fetch| (FILLEDPOLYGONDATA STYLE) |of| OBJECTDATA))
          (PRINTOUT STREAM "(" \, "POLYGONPOINTS" \, .P2 (|fetch| (FILLEDPOLYGONDATA POLYGONPOINTS) |of| OBJECTDATA)
                 \, "TEXTURE" \, .P2 (|fetch| (FILLEDPOLYGONDATA TEXTURE) |of| OBJECTDATA)
                 \, "STYLE" \, .P2 (LIST (|fetch| (PLOT.STYLE LINEWIDTH) |of| STYLE)
                                         (|fetch| (PLOT.STYLE DASHING) |of| STYLE)
                                         (|fetch| (PLOT.STYLE COLOR) |of| STYLE))
                 \, ")"))))
)

(RPAQQ OBJECT2OPSTABLE
       ((FILLEDPOLYGON (DRAWFN DRAWFILLEDPOLYGON)
               (ERASEFN ERASEFILLEDPOLYGON)
               (HIGHLIGHTFN HIGHLIGHTFILLEDPOLYGON)
               (MOVEFN MOVEFILLEDPOLYGON)
               (LABELFN LABELGENERIC)
               (EXTENTFN EXTENTOFFILLEDPOLYGON)
               (DISTANCEFN DISTANCETOFILLEDPOLYGON)
               (COPYFN COPYFILLEDPOLYGON)
               (PUTFN PUTFILLEDPOLYGON)
               (GETFN GETFILLEDPOLYGON))))

(DECLARE\: EVAL@COMPILE

(DATATYPE FILLEDPOLYGONDATA (POLYGONPOINTS STREAMPOINTS STYLE TEXTURE)
       STYLE _ 1)
)

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

(PLOT.SETUP OBJECT2OPSTABLE)

(DEFINEQ

(CLIPPED.FILLPOLYGON
  (LAMBDA (CLIPPINGREGION POINTS TEXTURE STREAM OPERATION WINDNUMBER DRAW? WIDTH DRAWOPERATION COLOR DASHING)
                                                                 ; Edited  7-Oct-88 09:03 by thh:

    ;; Clip filled polygon against CLIPPINGREGION. If DRAW? is non-NIL, the clipped perimeter of the polygon is drawn as well using the remaining
    ;; parameters.

    (LET ((CLIPPEDPOINTS (CLIPPED.POLYGON CLIPPINGREGION POINTS)))
                                                                 ; CLIPPEDPOINTS is NIL if polygon doesn't intersect
                                                                 ; CLIPPINGREGION
         (COND
            (CLIPPEDPOINTS

                   ;; fill clipped polygon

                   (FILLPOLYGON CLIPPEDPOINTS TEXTURE STREAM OPERATION WINDNUMBER)

                   ;; draw if requested

                   (AND DRAW? (|bind| (START _ (CAR POINTS)) |first| (MOVETO (|fetch| XCOORD |of| START)
                                                                              (|fetch| YCOORD |of| START)
                                                                              STREAM)
                                 |for| PT |in| (CDR POINTS) |do| (CLIPPED.DRAWTO CLIPPINGREGION
                                                                        (|fetch| XCOORD |of| PT)
                                                                        (|fetch| YCOORD |of| PT)
                                                                        WIDTH DRAWOPERATION STREAM COLOR DASHING)
                                 |finally| (CLIPPED.DRAWTO CLIPPINGREGION (|fetch| XCOORD |of| START)
                                                  (|fetch| YCOORD |of| START)
                                                  WIDTH DRAWOPERATION STREAM COLOR DASHING))))))))

(CLIPPED.POLYGON
  (LAMBDA (CLIPPINGREGION POINTS)                                ; Edited  6-Oct-88 17:10 by thh:

    ;; clips polygon whose vertices are given in POINTS to CLIPPINGREGION using Sutherland-Hodgman algorithm. cf. p.450 of Foley and Van Dam

    (LET* ((LEFT (|fetch| LEFT |of| CLIPPINGREGION))
           (RIGHT (|fetch| RIGHT |of| CLIPPINGREGION))
           (TOP (|fetch| TOP |of| CLIPPINGREGION))
           (BOTTOM (|fetch| BOTTOM |of| CLIPPINGREGION))
           (EDGES (LIST (|create| CLIPEDGEINFO
                               X _ LEFT
                               Y _ BOTTOM
                               END _ TOP
                               VERTICAL? _ T)
                        (|create| CLIPEDGEINFO
                               X _ LEFT
                               Y _ TOP
                               END _ RIGHT
                               VERTICAL? _ NIL)
                        (|create| CLIPEDGEINFO
                               X _ RIGHT
                               Y _ TOP
                               END _ BOTTOM
                               VERTICAL? _ T)
                        (|create| CLIPEDGEINFO
                               X _ RIGHT
                               Y _ BOTTOM
                               END _ LEFT
                               VERTICAL? _ NIL)))
           CLIPPEDPOINTS)
          ;; each edge in EDGES is a pair of points such that on moving from first to second, inside of CLIPPINGREGION is on the right. THESE
          ;; ARE LEFT, TOP, RIGHT AND BOTTOM EDGES RESPECTIVELY.

          (FOR PT IN POINTS DO (SETQ CLIPPEDPOINTS (CLIP-POLYGON-VERTEX PT EDGES CLIPPEDPOINTS)))
          (FINISH-CLIP-POLYGON EDGES CLIPPEDPOINTS))))

(CLIP-POLYGON-VERTEX
  (LAMBDA (POINT EDGES CLIPPEDPOINTS)                               ; Edited  6-Oct-88 16:02 by thh:

;;; implements single step of Sutherland-Hodgman algorithm

    (COND
       (EDGES (LET* ((EDGE (CAR EDGES))
                     (PREVPOINT (|fetch| (CLIPEDGEINFO PREVPT) |of| EDGE))
                     (PREVINSIDE? (|fetch| (CLIPEDGEINFO PREVINSIDE?) |of| EDGE))
                     (INSIDE? (CLIP-INSIDEP POINT EDGE)))

                    ;; update points and check for intersection

                    (COND
                       ((|fetch| (CLIPEDGEINFO FIRSTPT) |of| EDGE)

                        ;; this is not first point of polygon to be clipped with this edge

                        (COND
                           ((NEQ PREVINSIDE? INSIDE?)                ; polygon side crosses edge
                            (SETQ CLIPPEDPOINTS (CLIP-POLYGON-VERTEX (CLIP-INTERSECT PREVPOINT POINT EDGE)
                                                        (CDR EDGES)
                                                        CLIPPEDPOINTS)))))
                       (T ;; this is first point of the polygon for this edge

                          (|replace| (CLIPEDGEINFO FIRSTPT) |of| EDGE |with| POINT)
                          (|replace| (CLIPEDGEINFO FIRSTINSIDE?) |of| EDGE |with| INSIDE?)))
                    (|replace| (CLIPEDGEINFO PREVPT) |of| EDGE |with| POINT)
                    (|replace| (CLIPEDGEINFO PREVINSIDE?) |of| EDGE |with| INSIDE?)

                    ;;

                    ;; check if new point should be included

                    (COND
                       (INSIDE? (SETQ CLIPPEDPOINTS (CLIP-POLYGON-VERTEX POINT (CDR EDGES)
                                                            CLIPPEDPOINTS))))))
       (T                                                            ; nothing to clip against
          (PUSH CLIPPEDPOINTS POINT)))
    CLIPPEDPOINTS))

(FINISH-CLIP-POLYGON
  (LAMBDA (EDGES CLIPPEDPOINTS)                                     ; Edited  6-Oct-88 16:10 by thh:
    (COND
       (EDGES (LET ((EDGE (CAR EDGES)))
                   (COND
                      ((AND CLIPPEDPOINTS (NEQ (|fetch| (CLIPEDGEINFO FIRSTINSIDE?) |of| EDGE)
                                                (|fetch| (CLIPEDGEINFO PREVINSIDE?) |of| EDGE)))
                                                                     ; last side of polygon crosses edge
                       (SETQ CLIPPEDPOINTS (CLIP-POLYGON-VERTEX (CLIP-INTERSECT (|fetch| (CLIPEDGEINFO FIRSTPT)
                                                                                     |of| EDGE)
                                                                         (|fetch| (CLIPEDGEINFO PREVPT)
                                                                            |of| EDGE)
                                                                         EDGE)
                                                   (CDR EDGES)
                                                   CLIPPEDPOINTS))))
                   (|replace| (CLIPEDGEINFO FIRSTPT) |of| EDGE |with| NIL)
                   (FINISH-CLIP-POLYGON (CDR EDGES)
                          CLIPPEDPOINTS)))
       (T CLIPPEDPOINTS))))

(CLIP-INSIDEP
  (LAMBDA (PT EDGE)                                                 ; Edited  6-Oct-88 16:32 by thh:
    ;; T if PT is on or to the right of the directed EDGE (which is the inside of the region of which it is a part)
    (COND
       ((|fetch| (CLIPEDGEINFO VERTICAL?) |of| EDGE)                ; vertical edge
        (COND
           ((GREATERP (|fetch| (CLIPEDGEINFO END) |of| EDGE)
                   (|fetch| (CLIPEDGEINFO Y) |of| EDGE))            ; edge is going up, right is positive x-axis
            (GEQ (|fetch| XCOORD |of| PT)
                 (|fetch| (CLIPEDGEINFO X) |of| EDGE)))
           (T (LEQ (|fetch| XCOORD |of| PT)
                   (|fetch| (CLIPEDGEINFO X) |of| EDGE)))))
       (T                                                            ; horizontal edge
          (COND
             ((GREATERP (|fetch| (CLIPEDGEINFO END) |of| EDGE)
                     (|fetch| (CLIPEDGEINFO X) |of| EDGE))          ; edge is going right, right is negative y-axis
              (LEQ (|fetch| YCOORD |of| PT)
                   (|fetch| (CLIPEDGEINFO Y) |of| EDGE)))
             (T (GEQ (|fetch| YCOORD |of| PT)
                     (|fetch| (CLIPEDGEINFO Y) |of| EDGE))))))))

(CLIP-INTERSECT
  (LAMBDA (P1 P2 EDGE)                                               ; Edited  6-Oct-88 16:42 by thh:

    ;; returns point where segment between P1 and P2 intersect EDGE (the two points are on opposite sides of the edge)

    (COND
       ((|fetch| (CLIPEDGEINFO VERTICAL?) |of| EDGE)                ; vertical edge
        (LET ((X (|fetch| (CLIPEDGEINFO X) |of| EDGE)))
             (|create| POSITION
                    XCOORD _ X
                    YCOORD _ (PLUS (|fetch| YCOORD |of| P1)
                                   (QUOTIENT (TIMES (DIFFERENCE X (|fetch| XCOORD |of| P1))
                                                    (DIFFERENCE (|fetch| YCOORD |of| P2)
                                                           (|fetch| YCOORD |of| P1)))
                                          (DIFFERENCE (|fetch| XCOORD |of| P2)
                                                 (|fetch| XCOORD |of| P1)))))))
       (T                                                            ; horizontal edge
          (LET ((Y (|fetch| (CLIPEDGEINFO Y) |of| EDGE)))
               (|create| POSITION
                      XCOORD _ (PLUS (|fetch| XCOORD |of| P1)
                                     (QUOTIENT (TIMES (DIFFERENCE Y (|fetch| YCOORD |of| P1))
                                                      (DIFFERENCE (|fetch| XCOORD |of| P2)
                                                             (|fetch| XCOORD |of| P1)))
                                            (DIFFERENCE (|fetch| YCOORD |of| P2)
                                                   (|fetch| YCOORD |of| P1))))
                      YCOORD _ Y))))))

)

(DECLARE\: EVAL@COMPILE

(RECORD CLIPEDGEINFO (X Y END VERTICAL? FIRSTPT FIRSTINSIDE? PREVPT PREVINSIDE?))
)

(PUTPROPS PLOTOBJECTS2 COPYRIGHT ("Xerox Corporation" 1988))
```

FUNCTION INDEX

RECORD INDEX

VARIABLE INDEX