

File created: 8-Dec-2023 21:42:20 {WMEDLEY}<sources>IMAGEIO.;8

edit by: rmk

changes to: (FNS \IMAGEIOINIT)  
(RECORDS IMAGEOPS)

previous date: 30-Oct-2021 19:09:48 {WMEDLEY}<sources>IMAGEIO.;7

Read Table: INTERLISP

Package: INTERLISP

Format: XCCS

## (RPAQQ IMAGEIOCOMS

```
[ (FNS IMAGESTREAMP IMAGESTREAMTYPE IMAGESTREAMTYPEP OPENIMAGESTREAM \GOOD.DASHLST)
  (INITVARS (IMAGESTREAMTYPES NIL))
  (FNS DRAWDASHEDLINE)
  (FNS DSPBACKCOLOR DSPBOTTOMMARGIN DSPCOLOR DSPCLIPPINGREGION DSPRESET DSPFONT DSPLEFTMARGIN DSPLINEFEED
    DSPOPERATION DSPRIGHTMARGIN DSPTOPMARGIN DSPSCALE DSPSPACEFACTOR DSPXPOSITION DSPYPOSITION
    DSPROTATE DSPPUSHSTATE DSPPOPSTATE DSPDEFAULTSTATE DSPSCALE2 DSPTRANSLATE)
  (FNS DSPNEWPAGE DRAWBETWEEN DRAWCIRCLE DRAWARC DRAWCURVE DRAWELLIPSE DRAWLINE DRAWPOLYGON DRAWPOINT
    FILLPOLYGON DRAWTO FILLCIRCLE MOVETO RELDRAWTO BITMAPIMAGESIZE SCALEDBITBLT)
  (FNS \DRAWPOINT.GENERIC \DRAWPOLYGON.GENERIC \DRAWCIRCLE.GENERIC \DRAWELLIPSE.GENERIC)
  (FNS \IMAGEIOINIT \NOIMAGE.DSPFONT \UNIMPIMAGEOP)
[COMS
  ;; stuff to support the checking and defaulting of arguments in the device independent drawing functions.
  (FNS INSURE.BRUSH BRUSHP \POSSIBLECOLOR NEGSHADE)
  (DECLARE%: DONTCOPY EVAL@COMPILE (RESOURCES SYSTEMBRUSH))
  (INITRESOURCES SYSTEMBRUSH)
  (FNS DASHINGP INSURE.DASHING)
  (DECLARE%: DONTCOPY (EXPORT (RECORDS BRUSH)))
  (DECLARE%: DONTCOPY (CONSTANTS (MICASPERPT (FQUOTIENT 635 18))
    (RECORDS IMAGEOPS)
    (GLOBALVARS \NOIMAGEOPS)))
  (INITRECORDS IMAGEOPS)
  (SYSRECORDS IMAGEOPS)
  (DECLARE%: DONTEVAL@LOAD DOCOPY (P (\IMAGEIOINIT)))
[COMS
  ;; Implementation of display stream resident 'files.' Done here cause it might matter that the display device get defined early so that its
  ;; event fn will be evaluated as the last thing before logout
  (INITVARS (\COLORDISPLAYSTREAMTYPES '(4DISPLAY 8DISPLAY 24DISPLAY))
    (\DISPLAYSTREAMTYPES (CONS 'DISPLAY \COLORDISPLAYSTREAMTYPES)))
  (FNS \DisplayEventFn \DISPLAYINIT \4DISPLAYINIT \8DISPLAYINIT \24DISPLAYINIT \DISPLAYSTREAMTYPEBPP
    )
  (ALISTS (IMAGESTREAMTYPES DISPLAY 4DISPLAY 8DISPLAY 24DISPLAY))
  (GLOBALVARS DisplayFDEV \4DISPLAYFDEV \8DISPLAYFDEV \24DISPLAYFDEV)
  (DECLARE%: DONTEVAL@LOAD DOCOPY (P (\DISPLAYINIT
    (\4DISPLAYINIT)
    (\8DISPLAYINIT)
    (\24DISPLAYINIT)
  (DECLARE%: DONTEVAL@LOAD DOEVAL@COMPILE DONTCOPY COMPILERVERS (ADDVARS (NLAMA)
    (NLAML)
    (LAMA IMAGESTREAMP))
```

(DEFINEQ

## (IMAGESTREAMP

```
[LAMBDA (NARGS)
  (PROG ([STREAM (AND (IGREATERP NARGS 0)
    (SELECTQ (ARG NARGS 1)
      (T \TERM.OFD)
      (NIL *STANDARD-OUTPUT*)
      (ARG NARGS 1)
    )
    STYPE)
  (OR (type? STREAM STREAM)
    (RETURN))
  (SETQ STYPE (fetch (IMAGEOPS IMAGETYPE) of (fetch (STREAM IMAGEOPS) of STREAM)))
  (RETURN (AND (COND
    ((EQ NARGS 2)
      (for X inside (ARG NARGS 2) always (EQMEMB X STYPE)))
    (T STYPE))
    STREAM])
```

## (IMAGESTREAMTYPE

```
[LAMBDA (STREAM)
  (fetch (IMAGEOPS IMAGETYPE) of (fetch (STREAM IMAGEOPS) of (\STREAMARG STREAM))) (* rmk%: "20-AUG-83 17:28")
```

## (IMAGESTREAMTYPEP

```
[LAMBDA (STREAM STYPE)
  (* AJB "16-Jul-85 15:31")
```

;;; Returns T if STREAM is an imagestream of type STYPE

```
(LET ((S (SELECTQ STREAM
```

```

      ((T NIL)
       (\GETSTREAM STREAM 'OUTPUT T))
      STREAM)))
(AND (type? STREAM S)
     (for X inside STYPE always (EQMEMB X (fetch (IMAGEOPS IMAGETYPE) of (fetch (STREAM IMAGEOPS)
                                             of S)))

```

**(OPENIMAGESTREAM**

```
[LAMBDA (FILE IMAGETYPE OPTIONS)
```

```

; Edited 1-Jun-93 12:32 by rmk:
; Edited 11-Jan-91 16:05 by jds

```

```

;; Opens an IMAGETYPE imagestream, or if NIL, an imagestream of a type that FILE (perhaps from DEFAULTPRINTINGHOST) can print directly.
;; If FILE is an the LPT device, then the type of the corresponding printer is used. If FILE is NIL, then an LPT file on a printer from default
;; printinghost is used, so the file will be printed on closing.

```

```

(DECLARE (GLOBALVARS IMAGESTREAMTYPES))
(LET (LPTNAME LPTP (DEFPRINTER (OR (CAR (LISTP DEFAULTPRINTINGHOST))
                                   DEFAULTPRINTINGHOST)))

```

```

  (SETQ FILE (\CONVERT-PATHNAME FILE))
  [COND

```

```

    ((AND (NULL FILE)
          (NEQ IMAGETYPE 'DISPLAY))

```

```

; YUCK! TAKE THIS OUT WHEN WE FIGURE OUT DISPLAY
; IMAGESTREAMS BETTER

```

```

      (SETQ LPTP T)
      (SETQ FILE '{LPT}))
      ((STREAMP FILE))
      ((EQ (FILENAMEFIELD FILE 'HOST)
           'LPT)
       (SETQ LPTP T)
       (LET (POS)

```

```

;; This should be (FILENAMEFIELD FILE 'NAME) except that FILENAMEFIELD won't accept : as part of the name, thinks it
;; marks a device field. This code is borrowed from PRINTERDEVICE

```

```

      (AND (SETQ POS (STRPOS "]" FILE))
           (SETQ LPTNAME (SUBATOM FILE (ADD1 POS)
                                   (SUB1 (OR (STRPOS "." FILE (ADD1 POS))
                                           0))

```

```

    ]
  [ (NULL IMAGETYPE)

```

```

;; Get the image type from FILE if it is an LPT file, otherwise choose the image type from the first printer on DEFAULTPRINTINGHOST
;; Assume that it will be printed on the defaultprintinghost if it is an ordinary filename. If defaultprinter is a list, chooses the
;; preferred-file-type if it is specified, otherwise uses the first of the printer type's CANPRINT property.

```

```

      (SETQ IMAGETYPE (COND
        ((PRINTFILETYPE.FROM.EXTENSION FILE))
        [(AND (NOT LPTNAME)
              (CADDR (LISTP DEFPRINTER)
                    [(CAR (MKLIST (PRINTERPROP (PRINTERTYPE (OR LPTNAME DEFPRINTER))
                                                'CANPRINT)
              (T (ERROR "Can't determine IMAGETYPE for " FILE]
[LPTNAME (OR (EQMEMB IMAGETYPE (PRINTERPROP (PRINTERTYPE LPTNAME)
                                                'CANPRINT))
              (ERROR (CONCAT "Printer " LPTNAME " can't print " IMAGETYPE " files")
; This includes the NIL FILE case, cause of initial coercion
(LPTP (FOR P INSIDE DEFAULTPRINTINGHOST WHEN (EQMEMB IMAGETYPE (PRINTERPROP (PRINTERTYPE P)
                                                'CANPRINT))

```

```

      DO (SETQ LPTNAME (PRINTERNAME P))
         (SETQ FILE (PACKFILENAME 'HOST 'LPT 'NAME LPTNAME))
         (RETURN)

```

```

      FINALLY (ERROR (CONCAT "Can't find a printer on DEFAULTPRINTINGHOST that can print "
                             IMAGETYPE " files")

```

```

(LET ((STREAM (APPLY* (OR [CADR (ASSOC 'OPENSTREAM (CDR (ASSOC IMAGETYPE IMAGESTREAMTYPES)]
                    (ERROR "No open function for " IMAGETYPE " streams"))

```

```

      [COND
        ((OR LPTP (STREAMP FILE)
              (EQ IMAGETYPE 'DISPLAY))
         FILE)

```

```

      (T ; Stick on default extension from PRINTFILETYPES

```

```

        (PACKFILENAME 'BODY FILE 'EXTENSION
          (OR [CAR (CADR (ASSOC 'EXTENSION (CDR (ASSOC IMAGETYPE PRINTFILETYPES)

```

```

              IMAGETYPE]

```

```

        OPTIONS)))

```

```

      (IF LPTNAME
        THEN (STREAMPROP STREAM 'PRINTERNAME LPTNAME))
      STREAM])

```

**(GOOD.DASHLST**

```
[LAMBDA (DASHING BRUSH)
```

```
(* rrb " 9-Sep-86 16:16")
```

```
;;: massage the DASHING parameter to mesh well with the size of the BRUSH
```

```

(PROG [(DASHLST (INSURE.DASHING DASHING))
      (BRUSHSIZE (COND
        ((LITATOM BRUSH)
         1)

```

```
; handles NULL and function name case.
```

```

((BITMAPP BRUSH)
 (IQUOTIENT (IPLUS 2 (BITMAPHEIGHT BRUSH)
 (BITMAPWIDTH BRUSH))
 2))
((NUMBERP BRUSH)
 (FIXR BRUSH))
(T (fetch (BRUSH BRUSHSIZE) of BRUSH]

```

; brush can be a number meaning ROUND and it hasn't been ; coerced yet.

```

[COND
 ((AND DASHLST (GREATERP BRUSHSIZE 1))
 [COND
 ((ODDP (LENGTH DASHLST))

```

; adjust the dashing to take into account the brush size.

; even out the DASHLST because on and off are handled ; differently.

```

 (SETQ DASHLST (APPEND DASHLST DASHLST]
 (SETQ DASHLST (bind NOWOFF for NDASH in DASHLST collect (COND

```

(NOWOFF (SETQ NOWOFF NIL) (TIMES NDASH BRUSHSIZE)) ((SETQ NOWOFF T) ; make the on case be 1 for the first one and brushsize for every ; one after that.

```

 (ADD1 (TIMES (SUB1 NDASH) BRUSHSIZE]

```

```
(RETURN DASHLST])
```

)

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

```
(DEFINEQ
```

**{DRAWDASHEDLINE**

```

[LAMBDA (X1 Y1 X2 Y2 WIDTH OPERATION STREAM COLOR DASHING)
 [COND

```

; Edited 26-Jul-90 16:24 by matsuda

```

 ((NOT (EQ WIDTH 0))
 (PROG ((DASHON T)
 DASHTAIL DASHCNT (ADJACENT (IDIFFERENCE X2 X1))
 (OPPOSITE (IDIFFERENCE Y2 Y1))
 (LENGTHDRAWN 0)
 DASHLST NEWX NEWY LINELENGTH SINE COSINE)
 [SETQ LINELENGTH (FIX (SQRT (IPLUS (ITIMES ADJACENT ADJACENT)
 (ITIMES OPPOSITE OPPOSITE)

```

; expand the dashing by the width.

```

 (SETQ DASHLST (bind NOWOFF for NDASH in DASHING collect (TIMES NDASH WIDTH)))
 (SETQ DASHTAIL DASHLST)
 (SETQ SINE (FQUOTIENT OPPOSITE LINELENGTH))
 (SETQ COSINE (FQUOTIENT ADJACENT LINELENGTH))
 (while (ILESSP (PLUS LENGTHDRAWN (CAR DASHTAIL))
 LINELENGTH)

```

```

 do (SETQ DASHCNT (CAR DASHTAIL))
 (SETQ DASHTAIL (CDR DASHTAIL))
 (add LENGTHDRAWN DASHCNT)
 (SETQ NEWX (FPLUS X1 (FTIMES COSINE DASHCNT)))
 (SETQ NEWY (FPLUS Y1 (FTIMES SINE DASHCNT)))

```

;; Old code incorrect: (COND (DASHON (DRAWLINE X1 Y1 NEWX NEWY WIDTH OPERATION STREAM COLOR)) (T ; (RELMOVETO NEWX NEWY STREAM)))

```

 (if DASHON
 then (DRAWLINE X1 Y1 NEWX NEWY WIDTH OPERATION STREAM COLOR))
 (SETQ DASHON (NOT DASHON))
 (SETQ X1 NEWX)
 (SETQ Y1 NEWY)
 (COND
 (NULL DASHTAIL)
 (SETQ DASHTAIL DASHLST)))

```

finally ; do last partial segment

```

 (if DASHON
 then (DRAWLINE X1 Y1 X2 Y2 WIDTH OPERATION STREAM COLOR]

```

```
(MOVETO X2 Y2 STREAM])
```

)

```
(DEFINEQ
```

**{DSPBACKCOLOR**

```

[LAMBDA (COLOR STREAM)
 (IMAGEOP 'IMBACKCOLOR (SETQ STREAM (\OUTSTREAMARG STREAM)
 STREAM COLOR])

```

(\* rmk%: "12-Sep-84 09:53") ; Switches background color on stream

**{DSPBOTTOMMARGIN**

```

[LAMBDA (YPOSITION STREAM)
 (IMAGEOP 'IMBOTTOMMARGIN (SETQ STREAM (\OUTSTREAMARG STREAM)
 STREAM YPOSITION])

```

(\* rmk%: "26-Jun-84 13:56") ; Sets the Y position that forces a new page

**(DSPCOLOR**

[LAMBDA (COLOR STREAM)  
  
(IMAGEOP 'IMCOLOR (SETQ STREAM (\OUTSTREAMARG STREAM))  
STREAM COLOR])

(\* rmk%: "12-Sep-84 09:53")  
; Switches foreground color on stream

**(DSPCLIPPINGREGION**

[LAMBDA (REGION STREAM)  
  
(AND REGION (NOT (type? REGION REGION))  
(\ILLEGAL.ARG REGION))  
(COND  
(STREAM  
(IMAGEOP 'IMCLIPPINGREGION (SETQ STREAM (\OUTSTREAMARG STREAM))  
STREAM REGION))  
(T (\ILLEGAL.ARG STREAM]))

(\* bvm%: " 4-Sep-85 20:57")  
; Set the clipping region for an imagestream  
  
; special check done for NIL to stop default to primary output file.

**(DSPRESET**

[LAMBDA (STREAM)  
  
(IMAGEOP 'IMRESET (SETQ STREAM (\OUTSTREAMARG STREAM))  
STREAM])

(\* jds "11-Jan-85 16:54")  
; resets a display stream

**(DSPFONT**

[LAMBDA (FONT STREAM)  
  
(IMAGEOP 'IMFONT (SETQ STREAM (\OUTSTREAMARG STREAM))  
STREAM FONT])

(\* rmk%: " 2-SEP-83 10:50")  
; sets the font that an image stream uses to print characters.

**(DSPLEFTMARGIN**

[LAMBDA (XPOSITION STREAM)  
  
(IMAGEOP 'IMLEFTMARGIN (SETQ STREAM (\OUTSTREAMARG STREAM))  
STREAM XPOSITION])

(\* rmk%: " 2-SEP-83 10:50")  
; Sets the position that a carriage return returns to

**(DSPLINEFEED**

[LAMBDA (DELTAY STREAM)  
  
(IMAGEOP 'IMLINEFEED (SETQ STREAM (\OUTSTREAMARG STREAM))  
STREAM DELTAY])

(\* rmk%: " 2-SEP-83 10:50")  
; Sets the Xposition of STREAM

**(DSPOPERATION**

[LAMBDA (OPERATION STREAM)  
  
(IMAGEOP 'IMOPERATION (SETQ STREAM (\OUTSTREAMARG STREAM))  
STREAM OPERATION])

(\* rmk%: "12-Sep-84 09:56")  
; sets the operation field of a stream

**(DSPRIGHTMARGIN**

[LAMBDA (XPOSITION STREAM)  
  
(IMAGEOP 'IMRIGHTMARGIN (SETQ STREAM (\OUTSTREAMARG STREAM))  
STREAM XPOSITION])

(\* rmk%: " 2-SEP-83 10:51")  
; Sets the right margin that determines when a cr is inserted by  
; print.

**(DSPTOPMARGIN**

[LAMBDA (YPOSITION STREAM)  
  
(IMAGEOP 'IMTOPMARGIN (SETQ STREAM (\OUTSTREAMARG STREAM))  
STREAM YPOSITION])

(\* rmk%: "26-Jun-84 13:55")  
; Sets the Y position that a newpage starts at

**(DSPSCALE**

[LAMBDA (SCALE STREAM)  
  
(IMAGEOP 'IMSCALE (SETQ STREAM (\OUTSTREAMARG STREAM))  
STREAM SCALE])

(\* rmk%: "16-Jun-84 14:48")  
; Returns (and eventually will set) the current scale of STREAM.

**(DSPSPACEFACTOR**

[LAMBDA (FACTOR STREAM)  
  
(AND FACTOR (OR (GREATERP FACTOR 0)  
(\ILLEGAL.ARG FACTOR)))  
(IMAGEOP 'IMSPACEFACTOR (SETQ STREAM (\OUTSTREAMARG STREAM))  
STREAM FACTOR])

(\* rmk%: "27-Nov-84 18:57")  
; Sets the space factor of STREAM

**(DSPXPOSITION**

[LAMBDA (XPOSITION STREAM)

(\* rmk%: " 2-SEP-83 10:51")

; Sets the Xposition of STREAM

```
(IMAGEOP 'IMXPOSITION (SETQ STREAM (\OUTSTREAMARG STREAM))
  STREAM XPOSITION])
```

**(DSPYPOSITION**

```
[LAMBDA (YPOSITION STREAM)
```

(\* rmk%: " 2-SEP-83 10:51")  
; Sets the Yposition of STREAM

```
(IMAGEOP 'IMYPOSITION (SETQ STREAM (\OUTSTREAMARG STREAM))
  STREAM YPOSITION])
```

**(DSPROTATE**

```
[LAMBDA (ROTATION STREAM)
```

(\* hdj "22-Oct-85 12:15")  
; Sets the rotation of STREAM

```
(IMAGEOP 'IMROTATE (SETQ STREAM (\OUTSTREAMARG STREAM))
  STREAM ROTATION])
```

**(DSPPUSHSTATE**

```
[LAMBDA (STREAM)
```

(\* hdj "25-Nov-85 11:49")

;;; push a new graphics context for STREAM

```
(IMAGEOP 'IMPUSHSTATE (SETQ STREAM (\OUTSTREAMARG STREAM))
  STREAM])
```

**(DSPPOPSTATE**

```
[LAMBDA (STREAM)
```

(\* hdj "25-Nov-85 11:50")

;;; pop a the graphics context for STREAM

```
(IMAGEOP 'IMPOPSTATE (SETQ STREAM (\OUTSTREAMARG STREAM))
  STREAM])
```

**(DSPDEFAULTSTATE**

```
[LAMBDA (STREAM)
```

(\* hdj "30-Dec-85 17:39")

;;; push a new graphics context for STREAM

```
(IMAGEOP 'IMDEFAULTSTATE (SETQ STREAM (\OUTSTREAMARG STREAM))
  STREAM])
```

**(DSPSCALE2**

```
[LAMBDA (Sx Sy STREAM)
```

(\* hdj " 2-Jan-86 18:38")  
; Sets the scaling of STREAM

```
(IMAGEOP 'IMSCALE2 (SETQ STREAM (\OUTSTREAMARG STREAM))
  STREAM Sx Sy])
```

**(DSPTRANSLATE**

```
[LAMBDA (Tx Ty STREAM)
```

(\* hdj " 2-Jan-86 18:37")  
; Sets the translation of STREAM

```
(IMAGEOP 'IMTRANSLATE (SETQ STREAM (\OUTSTREAMARG STREAM))
  STREAM Tx Ty])
```

)

(DEFINEQ

**(DSPNEWPAGE**

```
[LAMBDA (STREAM)
```

(\* jds " 9-Feb-86 17:18")

;;; Start a new page on the image stream STREAM.

```
(AND (STREAMPROP (SETQ STREAM (\OUTSTREAMARG STREAM))
  'BEFORENEWPAGEFN)
  (APPLY* (STREAMPROP STREAM 'BEFORENEWPAGEFN)
  STREAM))
```

; Let the stream's creator get control before and after the page  
; break, if he wants it.

```
(IMAGEOP 'IMNEWPAGE (SETQ STREAM (\OUTSTREAMARG STREAM))
  STREAM)
```

```
(AND (STREAMPROP STREAM 'AFTERNEWPAGEFN)
  (APPLY* (STREAMPROP STREAM 'AFTERNEWPAGEFN)
  STREAM))
```

**(DRAWBETWEEN**

```
[LAMBDA (PT1 PT2 WIDTH OPERATION STREAM COLOR DASHING)
```

; Edited 14-Feb-94 11:06 by nilsson  
; draws a line bewteen two points

```
(OR (POSITIONP PT1)
  (ERROR "Point1 not POSITIONP"))
(OR (POSITIONP PT2)
  (ERROR "Point2 not POSITIONP"))
```

```
(IMAGEOP 'IMDRAWLINE (SETQ STREAM (\OUTSTREAMARG STREAM))
  STREAM
  (fetch XCOORD of PT1)
  (fetch YCOORD of PT1)
  (fetch XCOORD of PT2)
  (fetch YCOORD of PT2)
  WIDTH OPERATION COLOR DASHING])
```

**(DRAWCIRCLE**

```
[LAMBDA (CENTERX CENTERY RADIUS BRUSH DASHING STREAM) (* rrb "30-Oct-85 14:22")
; Generic DRAWCIRCLE

(COND
  ((LESSP RADIUS 0)
   (\ILLEGAL.ARG RADIUS))
  ((EQP RADIUS 0)
   NIL)
  (T (IMAGEOP 'IMDRAWCIRCLE (SETQ STREAM (\OUTSTREAMARG STREAM))
    STREAM CENTERX CENTERY RADIUS (INSURE.BRUSH BRUSH STREAM)
    (INSURE.DASHING DASHING]))
```

**(DRAWARC**

```
[LAMBDA (CENTERX CENTERY RADIUS STARTANGLE NDEGREES BRUSH DASHING STREAM)
(* rrb "31-Oct-85 09:18")

;; Draws an arc of a given brush and dashing. NDEGREES can be either positive (counterclockwise) or negative (clockwise).

(IMAGEOP 'IMDRAWARC (SETQ STREAM (\OUTSTREAMARG STREAM))
  STREAM CENTERX CENTERY RADIUS STARTANGLE NDEGREES (INSURE.BRUSH BRUSH STREAM)
  (INSURE.DASHING DASHING])
```

**(DRAWCURVE**

```
[LAMBDA (KNOTS CLOSED BRUSH DASHING STREAM) (* edited%: "31-Mar-86 20:07")
; draws a spline curve with a given brush.

(LET ((VALIDBRUSH BRUSH))
  (if (NOT (BRUSHP BRUSH))
    then (SETQ VALIDBRUSH (INSURE.BRUSH BRUSH STREAM)))
  (IMAGEOP 'IMDRAWCURVE (SETQ STREAM (\OUTSTREAMARG STREAM))
    STREAM KNOTS CLOSED VALIDBRUSH (INSURE.DASHING DASHING))
  (if (NEQ VALIDBRUSH BRUSH)
    then (FREERESOURCE SYSTEMBRUSH VALIDBRUSH]))
```

**(DRAWELLIPSE**

```
[LAMBDA (CENTERX CENTERY SEMIMINORRADIUS SEMIMAJORRADIUS ORIENTATION BRUSH DASHING STREAM)
(* rrb "30-Oct-85 14:26")

;; Draws an ellipse. At ORIENTATION 0, the semimajor axis is horizontal, the semiminor axis vertical. Orientation is positive in the
;; counterclockwise direction. The current location in the stream is left at the center of the ellipse.

(DECLARE (LOCALVARS . T))
(IMAGEOP 'IMDRAWELLIPSE (SETQ STREAM (\OUTSTREAMARG STREAM))
  STREAM CENTERX CENTERY SEMIMINORRADIUS SEMIMAJORRADIUS ORIENTATION (INSURE.BRUSH BRUSH STREAM)
  (INSURE.DASHING DASHING])
```

**(DRAWLINE**

```
[LAMBDA (X1 Y1 X2 Y2 WIDTH OPERATION STREAM COLOR DASHING) ; Edited 6-Feb-87 15:06 by FS

;; Some streams allow WIDTH to be a BRUSH, display currently does not

(IMAGEOP 'IMDRAWLINE (SETQ STREAM (\OUTSTREAMARG STREAM))
  STREAM X1 Y1 X2 Y2 WIDTH OPERATION COLOR DASHING])
```

**(DRAWPOLYGON**

```
[LAMBDA (POINTS CLOSED BRUSH DASHING STREAM) ; Edited 13-Jan-88 21:00 by FS

;; draws a polygon with a given brush. Change so BRUSH can be just number, and passed through? Then display can you better drawline. Other
;; streams?

(IMAGEOP 'IMDRAWPOLYGON (SETQ STREAM (\OUTSTREAMARG STREAM))
  STREAM POINTS CLOSED (INSURE.BRUSH BRUSH STREAM)
  (INSURE.DASHING DASHING])
```

**(DRAWPOINT**

```
[LAMBDA (X Y BRUSH STREAM OPERATION) ; Edited 24-Aug-87 16:25 by FS

;; draws a brush point at position X Y. Doc says brush can be a BM (only fn so documented).

(IMAGEOP 'IMDRAWPOINT (SETQ STREAM (\OUTSTREAMARG STREAM))
  STREAM X Y (OR (BITMAPP BRUSH)
  (INSURE.BRUSH BRUSH STREAM))
  OPERATION])
```

**(FILLPOLYGON**

```
[LAMBDA (POINTS TEXTURE STREAM OPERATION WINDNUMBER) (* rrb " 5-Mar-86 15:39")
; fills a polygon with a given texture
```

```
(COND
  ((NOT (OR (EQUAL WINDNUMBER 0)
            (EQUAL WINDNUMBER 1))))
  (SETQ WINDNUMBER 1))
(IMAGEOP 'IMFILLPOLYGON (SETQ STREAM (\OUTSTREAMARG STREAM))
  STREAM POINTS TEXTURE (OR OPERATION (DSOPERATION NIL STREAM))
  WINDNUMBER])
```

**(DRAWTO**

```
[LAMBDA (X Y WIDTH OPERATION STREAM COLOR DASHING) (* hdj "7-Nov-84 14:03")
  ;; draws a line fro the current position of STREAM to absolute position X,Y.
  (IMAGEOP 'IMDRAWLINE (SETQ STREAM (\OUTSTREAMARG STREAM))
    STREAM
    (IMAGEOP 'IMXPOSITION STREAM STREAM)
    (IMAGEOP 'IMYPOSITION STREAM STREAM)
    X Y WIDTH OPERATION COLOR DASHING])
```

**(FILLCIRCLE**

```
[LAMBDA (CENTERX CENTERY RADIUS TEXTURE STREAM) (* rmk%: "2-SEP-83 10:54")
  (IMAGEOP 'IMFILLCIRCLE (SETQ STREAM (\OUTSTREAMARG STREAM))
    STREAM CENTERX CENTERY RADIUS TEXTURE])
```

**(MOVETO**

```
[LAMBDA (X Y STREAM) (* rmk%: "17-Sep-84 17:59")
  ; sets both the X and Y positions in a Stream
  (IMAGEOP 'IMMOVETO (SETQ STREAM (\OUTSTREAMARG STREAM))
    STREAM X Y])
```

**(RELDRAWTO**

```
[LAMBDA (DX DY WIDTH OPERATION STREAM COLOR DASHING) ; Edited 22-Apr-87 12:43 by rrb
  ; Draws a vector from the current position
  (PROG (ORIGX ORIGY (STRM (\OUTSTREAMARG STREAM)))
    (RETURN (COND
      ((NOT (AND (ZEROP DX)
                (ZEROP DY)))) ; documented to not draw anything if DX and DY are both 0
      (IMAGEOP 'IMDRAWLINE STRM STRM (SETQ ORIGX (IMAGEOP 'IMXPOSITION STRM STRM))
        (SETQ ORIGY (IMAGEOP 'IMYPOSITION STRM STRM))
        (IPLUS ORIGX DX)
        (IPLUS ORIGY DY)
        WIDTH OPERATION COLOR DASHING))
```

**(BITMAPIMAGESIZE**

```
[LAMBDA (BITMAP DIMENSION STREAM) (* hdj "19-Dec-84 11:57")
  (IMAGEOP 'IMBITMAPSIZE STREAM STREAM BITMAP DIMENSION])
```

**(SCALEDIBITBLT**

```
[LAMBDA (SOURCE SOURCELEFT SOURCEBOTTOM DESTINATION DESTINATIONLEFT DESTINATIONBOTTOM WIDTH HEIGHT SOURCTYPE
  OPERATION TEXTURE CLIPPINGREGION SCALE) ; Edited 29-Mar-89 18:32 by snow
  ;; Changed to pass thru the DESTINATIONLEFT and DESTINATIONBOTTOM arguments as NIL is significantly different than 0. NIL means "put
  ;; the bitmap at the current position." --was
  (IMAGEOP 'IMSCALEDIBITBLT DESTINATION SOURCE SOURCELEFT SOURCEBOTTOM DESTINATION DESTINATIONLEFT
    DESTINATIONBOTTOM WIDTH HEIGHT SOURCTYPE OPERATION TEXTURE CLIPPINGREGION
    (if CLIPPINGREGION
      then (fetch (REGION LEFT) of CLIPPINGREGION)
      else 0)
    (if CLIPPINGREGION
      then (fetch (REGION BOTTOM) of CLIPPINGREGION)
      else 0)
    (OR SCALE 1])
```

**(DEFINEQ**

**(DRAWPOINT.GENERIC**

```
[LAMBDA (STREAM X Y BRUSH OPERATION) (* hdj "19-Nov-86 15:12")
  ;; generic version of drawpoint that calls drawline. Used as the default.
  (DRAWLINE X Y X Y (fetch (BRUSH BRUSHSIZE) of BRUSH)
    OPERATION STREAM (fetch (BRUSH BRUSHCOLOR) of BRUSH])
```

**(DRAWPOLYGON.GENERIC**

```
[LAMBDA (STREAM POINTS CLOSED BRUSH DASHING) ; Edited 31-Mar-88 18:35 by FS
  ;; generic version of drawpolygon that calls drawline. Used as the default.
  (if POINTS
    then (bind (COLOR_ (fetch (BRUSH BRUSHCOLOR) of BRUSH)) for PTAIL on POINTS while (CDR PTAIL)
      do (DRAWLINE (fetch (POSITION XCOORD) of (CAR PTAIL))
```

```

(ffetch (POSITION YCOORD) of (CAR PTAIL))
(fetch (POSITION XCOORD) of (CADR PTAIL))
(fetch (POSITION YCOORD) of (CADR PTAIL))
BRUSH NIL STREAM COLOR DASHING)
finally (COND
  ((NULL (CDR POINTS)) ; only one point
    (DRAWPOINT (fetch (POSITION XCOORD) of (CAR POINTS))
      (ffetch (POSITION YCOORD) of (CAR POINTS))
      BRUSH STREAM NIL))
  ((AND CLOSED (CDDR POINTS)) ; draw the closing line.
    (DRAWLINE (fetch (POSITION XCOORD) of (CAR PTAIL))
      (ffetch (POSITION YCOORD) of (CAR PTAIL))
      (fetch (POSITION XCOORD) of (CAR POINTS))
      (ffetch (POSITION YCOORD) of (CAR POINTS))
      BRUSH NIL STREAM COLOR DASHING]))

```

**(DRAWCIRCLE.GENERIC**

[LAMBDA (STREAM CENTERX CENTERY RADIUS BRUSH DASHING) ; Edited 13-Apr-88 14:03 by FS

:: Approximate ellipse with cubic spline. Generic in the sense that if the stream supports splines, then this code will work (only as good as the approximation).. -FS.

::  
 :: Could have instead provided Pitteway's algorithm, but would have had to handle dashing, brushes, etc.  
 ::  
 :: Could also have simply called \DRAWELLIPSE.GENERIC.

```

(PROG [(R2RAD (FIXR (FTIMES RADIUS (CONSTANT (FQUOTIENT (SQRT 2)
  2)
  (DRAWCURVE (LIST (CREATEPOSITION (IPLUS CENTERX RADIUS)
    CENTERX)
    (CREATEPOSITION (IPLUS CENTERX R2RAD)
      (IPLUS CENTERY R2RAD))
    (CREATEPOSITION CENTERX (IPLUS CENTERY RADIUS))
    (CREATEPOSITION (IDIFFERENCE CENTERX R2RAD)
      (IPLUS CENTERY R2RAD))
    (CREATEPOSITION (IDIFFERENCE CENTERX RADIUS)
      CENTERX)
    (CREATEPOSITION (IDIFFERENCE CENTERX R2RAD)
      (IDIFFERENCE CENTERY R2RAD))
    (CREATEPOSITION CENTERX (IDIFFERENCE CENTERY RADIUS))
    (CREATEPOSITION (IPLUS CENTERX R2RAD)
      (IDIFFERENCE CENTERY R2RAD)))
  T BRUSH DASHING STREAM])

```

**(DRAWELLIPSE.GENERIC**

[LAMBDA (STREAM CENTERX CENTERY SEMIMINORRADIUS SEMIMAJORRADIUS ORIENTATION BRUSH DASHING) ; Edited 13-Apr-88 14:03 by FS

:: Approximate ellipse with cubic spline. Generic in the sense that if the stream supports splines, then this code will work (only as good as the approximation).. -FS.

:: not a great approximation for degenerate ellipses (e.g. minorrad. 1, majorrad 200), but seems to be more numerically stable than Pitteway's algorithm (in \DrawEllipse.Display)

```

(PROG ((SINOR (COND
  (ORIENTATION (SIN ORIENTATION))
  (T 0.0)))
  (COSOR (COND
  (ORIENTATION (COS ORIENTATION))
  (T 1.0)))
  (ROOT2DIV2 (CONSTANT (FQUOTIENT (SQRT 2)
  2))))
  MAJORXOFFSET MAJORYOFFSET MINORXOFFSET MINORYOFFSET)
  (SETQ MAJORXOFFSET (FTIMES COSOR SEMIMAJORRADIUS))
  (SETQ MAJORYOFFSET (FTIMES SINOR SEMIMAJORRADIUS))
  (SETQ MINORXOFFSET (FTIMES SINOR SEMIMINORRADIUS))
  (SETQ MINORYOFFSET (FTIMES COSOR SEMIMINORRADIUS))
  (SETQ EXTRAXOFFSET (CL:* ROOT2DIV2 (- MAJORXOFFSET MINORXOFFSET)))
  (SETQ EXTRAYOFFSET (CL:* ROOT2DIV2 (+ MAJORYOFFSET MINORYOFFSET)))
  (SETQ VERSOXOFFSET (CL:* ROOT2DIV2 (+ MAJORXOFFSET MINORXOFFSET)))
  (SETQ VERSOYOFFSET (CL:* ROOT2DIV2 (- MAJORYOFFSET MINORYOFFSET)))
  (DRAWCURVE (LIST (CREATEPOSITION (+ CENTERX MAJORXOFFSET)
    (+ CENTERY MAJORYOFFSET))
    (CREATEPOSITION (+ CENTERX EXTRAXOFFSET)
      (+ CENTERY EXTRAYOFFSET))
    (CREATEPOSITION (- CENTERX MINORXOFFSET)
      (+ CENTERY MINORYOFFSET))
    (CREATEPOSITION (- CENTERX VERSOXOFFSET)
      (- CENTERY VERSOYOFFSET))
    (CREATEPOSITION (- CENTERX MAJORXOFFSET)
      (- CENTERY MAJORYOFFSET))
    (CREATEPOSITION (- CENTERX EXTRAXOFFSET)
      (- CENTERY EXTRAYOFFSET))
    (CREATEPOSITION (+ CENTERX MINORXOFFSET)
      (- CENTERY MINORYOFFSET))
    (CREATEPOSITION (+ CENTERX VERSOXOFFSET)

```



(+ CENTERX VERSOYOFFSET))
T BRUSH DASHING STREAM)
(MOVETO CENTERX CENTERX STREAM)]

)

(DEFINEQ

(\IMAGEIOINIT

[LAMBDA NIL

; Edited 8-Dec-2023 21:38 by rmk
(\* rrb "17-Sep-86 15:09")
; most of the functions are filled with NIL from the record
; declaration for IMAGEOPS

(DECLARE (GLOBALVARS \NOIMAGEOPS))

(SETQ \NOIMAGEOPS (create IMAGEOPS
IMAGETYPE \_ NIL
IMXPOSITION \_ [FUNCTION (LAMBDA (STREAM POS)
(LET ((OPOS (POSITION STREAM)))
(PROG1 OPOS
(COND
(POS (SPACES (DIFFERENCE POS OPOS)
STREAM))))])
IMYPOSITION \_ [FUNCTION (LAMBDA (STREAM N)
(PROG1 (AND \#DISPLAYLINES (NEQ \CURRENTDISPLAYLINE -1)
(DIFFERENCE \#DISPLAYLINES
\CURRENTDISPLAYLINE))
[COND
(N (\UNIMPIMAGEOP STREAM 'DSPYPOSITION))]
IMFONT \_ (FUNCTION \NOIMAGE.DSPFONT)
IMLEFTMARGIN \_ (FUNCTION ZERO)
IMRIGHTMARGIN \_ [FUNCTION (LAMBDA (STREAM N)
(LINELENGTH N STREAM]
IMLINEFEED \_ [FUNCTION (LAMBDA (STREAM DY)
(PROG1 -1
[AND DY (COND
((NEQ DY -1)
(ERROR DY "Illegal DSPLINEFEED for
terminal"))]
IMSPACEFACTOR \_ [FUNCTION (LAMBDA (STREAM)
(\UNIMPIMAGEOP STREAM 'DSPSPACEFACTOR]
IMFONTCREATE \_ [FUNCTION (LAMBDA (STREAM)
(\UNIMPIMAGEOP STREAM 'FONTCREATE]
IMSTRINGWIDTH \_ [FUNCTION (LAMBDA (STREAM STR RDTBL)
(NCHARS STR RDTBL RDTBL]
IMCHARWIDTH \_ [FUNCTION (LAMBDA NIL 1]
IMDRAWPOLYGON \_ (FUNCTION NIL)
IMDRAWPOINT \_ (FUNCTION NIL)]

(\NOIMAGE.DSPFONT

[LAMBDA (STREAM FONT)

; Edited 30-Oct-2021 19:09 by rmk:
; Edited 28-Oct-87 20:10 by jds

:: DSPFONT method for non-image streams: Put out font-change characters.
:: RMK: Save and restore CHARPOSITION

(LET ((OLDFONT (ffetch (STREAM IMAGEDATA) of STREAM)))
(PROG1 OLDFONT
[AND (NEQ OLDFONT 0)
(LET ([FONTN (OR (SMALLP FONT)
(AND (type? FONTCLASS FONT)
(fetch (FONTCLASS PRETTYFONT#) of FONT]
CHARPOS)
(COND
((AND FONTN (NEQ FONTN OLDFONT))
;; must be an outchar so that if the file is run-coded, the font change characters will come out in charset 0.
(COND
((NEQ FONTN 0)
(SETQ CHARPOS (FFETCH (STREAM CHARPOSITION) OF STREAM))
(\OUTCHAR STREAM (CONSTANT (CHCON1 FONTESCAPECHAR)))
(\OUTCHAR STREAM FONTN)
(FREPLACE (STREAM CHARPOSITION) OF STREAM WITH CHARPOS)))
(freplace (STREAM IMAGEDATA) of STREAM with FONTN]]))

(\UNIMPIMAGEOP

[LAMBDA (STREAM OP)
(ERROR STREAM (CONCAT "does not support " OP))

(\* rmk%: "26-Jun-84 13:28")

)

:: stuff to support the checking and defaulting of arguments in the device independent drawing functions.

(DEFINEQ

(\INSURE.BRUSH

[LAMBDA (BRUSH STREAM NOERRORFLG)

; Edited 13-Jan-88 20:59 by FS



**(DASHINGP**

```
[LAMBDA (DASHING) (* rrb "30-Oct-85 11:33")
;; return DASHING if it is a legal DASHING Note that NIL is a legal dashing and this will return NIL.
(AND (LISTP DASHING)
      (for X in DASHING always (NUMBERP X))
      DASHING])
```

**(INSURE.DASHING**

```
[LAMBDA (DASHING NOERRORFLG) (* rrb "30-Oct-85 11:35")
;; checks to make sure DASHING is a legal dashing spec.
(COND
  (DASHING (COND
    ((DASHINGP DASHING))
    (NOERRORFLG NIL)
    (T (\ILLEGAL.ARG DASHING))
  )
)
```

(DECLARE%: DONTCOPY

;; FOLLOWING DEFINITIONS EXPORTED

```
(DECLARE%: EVAL@COMPILE
(RECORD BRUSH (BRUSHSHAPE BRUSHSIZE BRUSHCOLOR)
  BRUSHSHAPE _ 'ROUND BRUSHSIZE _ 1)
)
```

;; END EXPORTED DEFINITIONS

```
(DECLARE%: DONTCOPY
(DECLARE%: EVAL@COMPILE
(RPAQ MICASPERPT (FQUOTIENT 635 18))
(CONSTANTS (MICASPERPT (FQUOTIENT 635 18)))
)
```

(DECLARE%: DONTCOPY

;; FOLLOWING DEFINITIONS EXPORTED

```
(DECLARE%: EVAL@COMPILE
(PUTPROPS IMAGEOP MACRO [ARGS (CONS 'SPREADAPPLY* (CONS (COND
  [(EQ (CAR (LISTP (CAR ARGS)))
        'QUOTE)
    (LIST 'fetch (LIST 'IMAGEOPS (CADAR ARGS))
          'of
          (LIST 'fetch '(STREAM IMAGEOPS)
                'of
                (CADR ARGS])
          (CADR ARGS])
    (T (HELP "IMAGEOP - OPNAME not quoted:" ARGS))]
  (CDDR ARGS])
)
```

(DECLARE%: EVAL@COMPILE

```
(DATATYPE IMAGEOPS
  (IMAGETYPE IMCLOSEFN IMXPOSITION IMYPOSITION IMFONT IMLEFTMARGIN IMRIGHTMARGIN IMLINEFEED IMDRAWLINE
    IMDRAWCURVE IMDRAWCIRCLE IMDRAWELLIPSE IMFILLCIRCLE IMBLTSHADE IMBITBLT IMNEWPAGE IMMOVETO IMSCALE
    IMTERPRI IMTOPMARGIN IMBOTTOMMARGIN IMSPACEFACTOR IMFONTCREATE IMOPERATION IMCOLOR IMSTRINGWIDTH
    IMCHARWIDTH IMCHARWIDTHY IMBACKCOLOR IMBITMAPSIZE IMCLIPPINGREGION IMRESET IMDRAWPOLYGON
    IMFILLPOLYGON IMSCALEDBITBLT IMWRITEPIXEL (NIL POINTER
      ; Was IMCHARSET
    )
  )
  IMROTATE IMDRAWARC IMTRANSLATE IMSCALE2 IMPUSHSTATE IMPOPSTATE IMDEFAULTSTATE IMDRAWPOINT
  IMBLTCHAR IMXOFFSET IMYOFFSET)
IMCLOSEFN _ (FUNCTION NIL)
IMTERPRI _ [FUNCTION (LAMBDA (STREAM)
  (\OUTCHAR STREAM (CHARCODE EOL)
  (\OUTCHAR STREAM (CHARCODE ^L)
  )
)]
IMNEWPAGE _ [FUNCTION (LAMBDA (STREAM)
  (\OUTCHAR STREAM (CHARCODE ^L)
  )
)]
IMOPERATION _ (FUNCTION NIL)
IMCOLOR _ (FUNCTION NIL)
IMCLIPPINGREGION _ (FUNCTION NIL)
IMRESET _ (FUNCTION NIL)
IMBACKCOLOR _ (FUNCTION NIL)
IMSTRINGWIDTH _ [FUNCTION (LAMBDA (STREAM STR RDTBL)
```

```

        (STRINGWIDTH STR (DSPFONT NIL STREAM)
          RDTBL RDTBL)
IMCHARWIDTH _ [FUNCTION (LAMBDA (STREAM CHARCODE)
  (CHARWIDTH CHARCODE (DSPFONT NIL STREAM)
    (IMAGEOP 'IMXPOSITION STREAM STREAM X)
    (IMAGEOP 'IMYPOSITION STREAM STREAM Y)
  )
)]
IMBITMAPSIZE _ [FUNCTION (LAMBDA (STREAM BITMAP DIMENSION)
  (SELECTQ DIMENSION
    (WIDTH (TIMES (DSPSCALE NIL STREAM)
      (BITMAPWIDTH BITMAP)))
    (HEIGHT (TIMES (DSPSCALE NIL STREAM)
      (BITMAPHEIGHT BITMAP)))
    (NIL (CONS (TIMES (DSPSCALE NIL STREAM)
      (BITMAPWIDTH BITMAP))
      (TIMES (DSPSCALE NIL STREAM)
      (BITMAPHEIGHT BITMAP))))
    (\ILLEGAL.ARG DIMENSION]
)]
IMWRITEPIXEL _ (FUNCTION NIL)
IMXPOSITION _ (FUNCTION NIL)
IMYPOSITION _ (FUNCTION NIL)
IMFONT _ (FUNCTION NIL)
IMLEFTMARGIN _ (FUNCTION NIL)
IMRIGHTMARGIN _ (FUNCTION NIL)
IMLINEFEED _ (FUNCTION NIL)
IMDRAWLINE _ (FUNCTION NIL)
IMDRAWCURVE _ (FUNCTION NIL)
IMDRAWCIRCLE _ (FUNCTION NIL)
IMDRAWELLIPSE _ (FUNCTION NIL)
IMFILLCIRCLE _ (FUNCTION NIL)
IMBLTSHADE _ (FUNCTION NIL)
IMBITBLT _ (FUNCTION NIL)
IMSCALE _ (FUNCTION NIL)
IMTOPMARGIN _ (FUNCTION NIL)
IMBOTMARGIN _ (FUNCTION NIL)
IMSPACEFACTOR _ (FUNCTION NIL)
IMFONTCREATE _ (FUNCTION NIL)
IMCHARWIDTHY _ (FUNCTION NIL)
IMDRAWPOLYGON _ (FUNCTION \DRAWPOLYGON.GENERIC)
IMDRAWPOINT _ (FUNCTION \DRAWPOINT.GENERIC)
IMFILLPOLYGON _ (FUNCTION NIL)
IMSCALEDBITBLT _ (FUNCTION NIL)
IMROTATE _ (FUNCTION NIL)
IMDRAWARC _ (FUNCTION NIL)
IMTRANSLATE _ (FUNCTION NIL)
IMPUSHSTATE _ (FUNCTION NIL)
IMPOPSTATE _ (FUNCTION NIL)
IMSCALE2 _ (FUNCTION NIL)
IMDEFAULTSTATE _ (FUNCTION NIL)
IMBLTCHAR _ (FUNCTION \MEDW.BLTCHAR)
IMXOFFSET _ (FUNCTION \MEDW.XOFFSET)
IMYOFFSET _ (FUNCTION \MEDW.YOFFSET)
)
(/DECLAREDATATYPE 'IMAGEOPS
  ( POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER
    POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER
    POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER )
  ;; ---field descriptor list elided by lister---
  ' 96)
(DECLARE%: DOEVAL@COMPILE DONTCOPY
(GLOBALVARS \NOIMAGEOPS)
)
)
;; END EXPORTED DEFINITIONS
(/DECLAREDATATYPE 'IMAGEOPS
  ( POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER
    POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER
    POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER POINTER )
  ;; ---field descriptor list elided by lister---
  ' 96)
(ADDOVAR SYSTEMRECLST
  (DATATYPE IMAGEOPS
    (IMAGETYPE IMCLOSEFN IMXPOSITION IMYPOSITION IMFONT IMLEFTMARGIN IMRIGHTMARGIN IMLINEFEED
      IMDRAWLINE IMDRAWCURVE IMDRAWCIRCLE IMDRAWELLIPSE IMFILLCIRCLE IMBLTSHADE IMBITBLT
      IMNEWPAGE IMMOVETO IMSCALE INTERPRI IMTOPMARGIN IMBOTTOMMARGIN IMSPACEFACTOR
      IMFONTCREATE IMOPERATION IMCOLOR IMSTRINGWIDTH IMCHARWIDTH IMCHARWIDTHY IMBACKCOLOR
      IMBITMAPSIZE IMCLIPPINGREGION IMRESET IMDRAWPOLYGON IMFILLPOLYGON IMSCALEDBITBLT

```

```

IMWRITEPIXEL (NIL POINTER ) ; Was IMCHARSET
IMROTATE IMDRAWARC IMTRANSLATE IMSCALE2 IMPUSHSTATE IMPOPSTATE IMDEFAULTSTATE
IMDRAWPOINT IMBLTCHAR IMXOFFSET IMYOFFSET))

```

```
(DECLARE%: DONTEVAL@LOAD DOCOPY
```

```
(\IMAGEIOINIT)
)
```

;; Implementation of display stream resident 'files.' Done here cause it might matter that the display device get defined early so that its event fn will be  
 ;; evaluated as the last thing before logout

```
(RPAQ? \COLORDISPLAYSTREAMTYPES ' (4DISPLAY 8DISPLAY 24DISPLAY))
```

```
(RPAQ? \DISPLAYSTREAMTYPES (CONS 'DISPLAY \COLORDISPLAYSTREAMTYPES))
```

```
(DEFINEQ
```

**(\DisplayEventFn**

```

[LAMBDA (FDEV EVENT) (* bvm%: "25-MAY-83 12:32")
  (SELECTQ EVENT
    (BEFORELOGOUT (DISPLAYBEFOREEXIT 'LOGOUT))
    (AFTERLOGOUT (DISPLAYAFTERENTRY 'LOGOUT))
    (BEFOREMAKESYS
      (DISPLAYBEFOREEXIT 'MAKESYS))
    (AFTERMAKESYS (DISPLAYAFTERENTRY 'MAKESYS))
    ((BEFORESYSOUT BEFORESAVEVM
      (DISPLAYBEFOREEXIT 'SYSOUT))
     ((AFTERSYSOUT AFTERSAVEVM
      (DISPLAYAFTERENTRY 'SYSOUT))
      NIL])

```

**(\DISPLAYINIT**

```
[LAMBDA NIL ; Edited 25-Sep-2021 20:57 by rmk:
```

;; Initializes global variables for the Display device  
 ;; Display Streams are referred to only by themselves so they do not need directory operations. Most of the fields in the DisplayDevice are empty  
 ;; to avoid something bad happening.

```
(DECLARE (GLOBALVARS DisplayFDEV \DISPLAYIMAGEOPS \DisplayDeviceMethods \DisplayDeviceData))
```

```
(SETQ \DisplayDeviceMethods (create WSOPS))
```

```
(SETQ \DisplayDeviceData (create WSDATA
  WSDESTINATION _ "Destination"
  WSREGION _
  (create REGION
    LEFT _ 0
    BOTTOM _ 0
    WIDTH _ 1024
    HEIGHT _ 808)))
```

```
(MAKE-EXTERNALFORMAT :DISPLAY NIL NIL NIL (FUNCTION \DSPPRINTCHAR)
  NIL CR.EOLC)
```

```
(SETQ \DISPLAYIMAGEOPS (create IMAGEOPS
  IMAGETYPE _ 'DISPLAY
  IMFONT _ (FUNCTION \DSPFONT.DISPLAY)
  IMLEFTMARGIN _ (FUNCTION \DSPLEFTMARGIN.DISPLAY)
  IMRIGHTMARGIN _ (FUNCTION \DSPRIGHTMARGIN.DISPLAY)
  IMLINEFEED _ (FUNCTION \DSPLINEFEED.DISPLAY)
  IMXPOSITION _ (FUNCTION \DSPXPOSITION.DISPLAY)
  IMYPOSITION _ (FUNCTION \DSPYPOSITION.DISPLAY)
  IMCLOSEFN _ (FUNCTION NIL)
  IMDRAWCURVE _ (FUNCTION \DRAWCURVE.DISPLAY)
  IMFILLCIRCLE _ '\FILLCIRCLE.DISPLAY
  IMDRAWLINE _ (FUNCTION \DRAWLINE.DISPLAY)
  IMDRAWELLIPSE _ (FUNCTION \DRAWELLIPSE.DISPLAY)
  IMDRAWCIRCLE _ (FUNCTION \DRAWCIRCLE.DISPLAY)
  IMFILLPOLYGON _ (FUNCTION \POLYSHADE.DISPLAY)
  IMBITBLT _ (FUNCTION \BITBLT.DISPLAY)
  IMSCALEDBITBLT _ (FUNCTION \SCALEDBITBLT.DISPLAY)
  IMBLTSHADE _ (FUNCTION \BLTSHADE.DISPLAY)
  IMNEWPAGE _ (FUNCTION \NEWPAGE.DISPLAY)
  IMSCALE _ [FUNCTION (LAMBDA NIL 1)
  IMSPACEFACTOR _ (FUNCTION NIL)
  IMFONTCREATE _ 'DISPLAY
  IMCOLOR _ (FUNCTION NIL)
  IMBACKCOLOR _ (FUNCTION \BACKCOLOR.DISPLAY)
  IMOPERATION _ (FUNCTION \DSPOPERATION.DISPLAY)
  IMSTRINGWIDTH _ (FUNCTION \STRINGWIDTH.DISPLAY)
  IMCHARWIDTH _ (FUNCTION \CHARWIDTH.DISPLAY)
  IMCLIPPINGREGION _ (FUNCTION \DSPCLIPPINGREGION.DISPLAY)
  IMRESET _ (FUNCTION \DSPRESET.DISPLAY)
  IMDRAWARC _ (FUNCTION \DRAWARC.DISPLAY)
  IMDRAWPOLYGON _ (FUNCTION \DRAWPOLYGON.DISPLAY)
  IMDRAWPOINT _ (FUNCTION \DRAWPOINT.DISPLAY)
  IMBLTCHAR _ (FUNCTION \MEDW.BLTCHAR)
  IMXOFFSET _ (FUNCTION \MEDW.XOFFSET)
```

```

      IMYOFFSET _ (FUNCTION \MEDW.YOFFSET))
(SETQ DisplayFDEV (create FDEV
  DEVICENAME _ 'DISPLAY
  RESETABLE _ NIL
  RANDOMACCESSP _ NIL
  PAGEMAPPED _ NIL
  CLOSEFILE _ (FUNCTION NIL)
  DELETEDFILE _ (FUNCTION NIL)
  GETFILEINFO _ (FUNCTION NIL)
  OPENFILE _ [FUNCTION (LAMBDA (NAME ACCESS RECOG OTHERINFO FDEV)
    NAME)]
  READPAGES _ (FUNCTION \ILLEGAL.DEVICEOP)
  SETFILEINFO _ (FUNCTION NIL)
  GENERATEFILES _ (FUNCTION \GENERATENOFILES)
  TRUNCATEFILE _ (FUNCTION NIL)
  WRITEPAGES _ (FUNCTION \ILLEGAL.DEVICEOP)
  GETFILENAME _ [FUNCTION (LAMBDA (NAME RECOG FDEV)
    NAME)]
  REOPENFILE _ [FUNCTION (LAMBDA (NAME)
    NAME)]
  EVENTFN _ (FUNCTION \DisplayEventFn)
  DIRECTORYNAMEP _ (FUNCTION NIL)
  HOSTNAMEP _ (FUNCTION NIL)
  BIN _ (FUNCTION \ILLEGAL.DEVICEOP)
  BOUT _ (FUNCTION \DSPPRINTCHAR)
  PEEKBIN _ (FUNCTION \ILLEGAL.DEVICEOP)
  BACKFILEPTR _ (FUNCTION \PAGEDBACKFILEPTR)
  BLOCKIN _ (FUNCTION \ILLEGAL.DEVICEOP)
  BLOCKOUT _ (FUNCTION \NONPAGEDBOUTS)
  WINDOWOPS _ \DisplayDeviceMethods
  WINDOWDATA _ \DisplayDeviceData
  DEVICEINFO _ (create DISPLAYSTATE)
  DEFAULTEXTERNALFORMAT _ :DISPLAY))
(\DEFINEDEVICE 'LFDISPLAY DisplayFDEV)

```

(\4DISPLAYINIT

; Edited 25-Sep-2021 18:42 by rmk:

```

[LAMBDA NIL
(DECLARE (GLOBALVARS \4DISPLAYIMAGEOPS \4DISPLAYFDEV))
(SETQ \4DISPLAYIMAGEOPS (create IMAGEOPS
  IMAGETYPE _ '4DISPLAY
  IMFONT _ (FUNCTION \DSFFONT.DISPLAY)
  IMLEFTMARGIN _ (FUNCTION \DSPLEFTMARGIN.DISPLAY)
  IMRIGHTMARGIN _ (FUNCTION \DSPRIGHTMARGIN.DISPLAY)
  IMLINEFEED _ (FUNCTION \DSPLINEFEED.DISPLAY)
  IMXPOSITION _ (FUNCTION \DSPXPOSITION.DISPLAY)
  IMYPOSITION _ (FUNCTION \DSPYPOSITION.DISPLAY)
  IMCLOSEFN _ (FUNCTION NIL)
  IMDRAWCURVE _ (FUNCTION \DRAWCURVE.DISPLAY)
  IMFILLCIRCLE _ 'FILLCIRCLE.DISPLAY
  IMDRAWLINE _ (FUNCTION \DRAWLINE.DISPLAY)
  IMDRAWELLIPSE _ (FUNCTION \DRAWELLIPSE.DISPLAY)
  IMDRAWCIRCLE _ (FUNCTION \DRAWCIRCLE.DISPLAY)
  IMBITBLT _ (FUNCTION \BITBLT.DISPLAY)
  IMBLTSHADE _ (FUNCTION \BLTSHADE.DISPLAY)
  IMNEWPAGE _ (FUNCTION \NEWPAGE.DISPLAY)
  IMSCALE _ [FUNCTION (LAMBDA NIL 1)]
  IMSPACEFACTOR _ (FUNCTION \DSPSPACEFACTOR.DISPLAY)
  IMFONTCREATE _ '4DISPLAY
  IMCOLOR _ (FUNCTION \DSPCOLOR.DISPLAY)
  IMBACKCOLOR _ (FUNCTION \DSPBACKCOLOR.DISPLAY)
  IMOPERATION _ (FUNCTION \DSPOPERATION.DISPLAY)
  IMSTRINGWIDTH _ (FUNCTION \STRINGWIDTH.DISPLAY)
  IMCHARWIDTH _ (FUNCTION \CHARWIDTH.DISPLAY)
  IMCLIPPINGREGION _ (FUNCTION \DSPCLIPPINGREGION.DISPLAY)
  IMRESET _ (FUNCTION \DSPRESET.DISPLAY)
  IMDRAWARC _ (FUNCTION \DRAWARC.DISPLAY)
  IMDRAWPOLYGON _ (FUNCTION \DRAWPOLYGON.DISPLAY)
  IMDRAWPOINT _ (FUNCTION \DRAWPOINT.DISPLAY)
  IMBLTCHAR _ (FUNCTION \MEDW.BLTCHAR)
  IMXOFFSET _ (FUNCTION \MEDW.XOFFSET)
  IMYOFFSET _ (FUNCTION \MEDW.YOFFSET))
(SETQ \4DISPLAYFDEV (create FDEV
  DEVICENAME _ '4DISPLAY
  RESETABLE _ NIL
  RANDOMACCESSP _ NIL
  PAGEMAPPED _ NIL
  CLOSEFILE _ (FUNCTION NIL)
  DELETEDFILE _ (FUNCTION NIL)
  GETFILEINFO _ (FUNCTION NIL)
  OPENFILE _ [FUNCTION (LAMBDA (NAME ACCESS RECOG OTHERINFO FDEV)
    NAME)]
  READPAGES _ (FUNCTION \ILLEGAL.DEVICEOP)
  SETFILEINFO _ (FUNCTION NIL)
  GENERATEFILES _ (FUNCTION \GENERATENOFILES)
  TRUNCATEFILE _ (FUNCTION NIL)
  WRITEPAGES _ (FUNCTION \ILLEGAL.DEVICEOP)

```

```

GETFILENAME _ [FUNCTION (LAMBDA (NAME RECOG FDEV)
NAME)]
REOPENFILE _ [FUNCTION (LAMBDA (NAME)
NAME)]
EVENTFN _ (FUNCTION NIL)
DIRECTORYNAMEP _ (FUNCTION NIL)
HOSTNAMEP _ (FUNCTION NIL)
BIN _ (FUNCTION \ILLEGAL.DEVICEOP)
BOUT _ (FUNCTION \DSPPRINTCHAR)
PEEKBIN _ (FUNCTION \ILLEGAL.DEVICEOP)
BACKFILEPTR _ (FUNCTION \PAGEDBACKFILEPTR)
BLOCKIN _ (FUNCTION \ILLEGAL.DEVICEOP)
BLOCKOUT _ (FUNCTION \NONPAGEDBOUTS)
DEVICEINFO _ (create DISPLAYSTATE)
WINDOWOPS _ NIL
DEFAULTEXTERNALFORMAT _ :DISPLAY))

```

(\DEFINEDEVICE NIL \4DISPLAYFDEV)

(\8DISPLAYINIT

; Edited 25-Sep-2021 18:43 by rmk:

```

[LAMBDA NIL
(DECLARE (GLOBALVARS \8DISPLAYIMAGEOPS \8DISPLAYFDEV))
(SETQ \8DISPLAYIMAGEOPS (create IMAGEOPS
IMAGETYPE _ '8DISPLAY
IMFONT _ (FUNCTION \DSPFONT.DISPLAY)
IMLEFTMARGIN _ (FUNCTION \DSPLEFTMARGIN.DISPLAY)
IMRIGHTMARGIN _ (FUNCTION \DSPRIGHTMARGIN.DISPLAY)
IMLINEFEED _ (FUNCTION \DSPLINEFEED.DISPLAY)
IMXPOSITION _ (FUNCTION \DSPXPOSITION.DISPLAY)
IMYPOSITION _ (FUNCTION \DSPYPOSITION.DISPLAY)
IMCLOSEFN _ (FUNCTION NIL)
IMDRAWCURVE _ (FUNCTION \DRAWCURVE.BIGBM)
IMFILLCIRCLE _ (FUNCTION \FILLCIRCLE.BIGBM)
IMDRAWLINE _ (FUNCTION \DRAWLINE.DISPLAY)
IMDRAWELLIPSE _ (FUNCTION \DRAWELLIPSE.BIGBM)
IMDRAWCIRCLE _ (FUNCTION \DRAWCIRCLE.BIGBM)
IMBITBLT _ (FUNCTION \BITBLT.DISPLAY)
IMBLTSHADE _ (FUNCTION \BLTSHADE.DISPLAY)
IMNEWPAGE _ (FUNCTION \NEWPAGE.DISPLAY)
IMSCALE _ [FUNCTION (LAMBDA NIL 1)]
IMSPACEFACTOR _ (FUNCTION \DSPSPACEFACTOR.DISPLAY)
IMFONTCREATE _ '8DISPLAY
IMCOLOR _ (FUNCTION \DSPCOLOR.DISPLAY)
IMBACKCOLOR _ (FUNCTION \DSPBACKCOLOR.DISPLAY)
IMOPERATION _ (FUNCTION \DSPOPERATION.DISPLAY)
IMSTRINGWIDTH _ (FUNCTION \STRINGWIDTH.DISPLAY)
IMCHARWIDTH _ (FUNCTION \CHARWIDTH.DISPLAY)
IMCLIPPINGREGION _ (FUNCTION \DSPCLIPPINGREGION.DISPLAY)
IMRESET _ (FUNCTION \DSPRESET.DISPLAY)
IMDRAWARC _ (FUNCTION \DRAWARC.DISPLAY)
IMDRAWPOLYGON _ (FUNCTION \DRAWPOLYGON.DISPLAY)
IMDRAWPOINT _ (FUNCTION \DRAWPOINT.DISPLAY)
IMBLTCHAR _ (FUNCTION \MEDW.BLTCHAR)
IMXOFFSET _ (FUNCTION \MEDW.XOFFSET)
IMYOFFSET _ (FUNCTION \MEDW.YOFFSET)))
(SETQ \8DISPLAYFDEV (create FDEV
DEVICENAME _ '8DISPLAY
RESETABLE _ NIL
RANDOMACCESSP _ NIL
PAGEMAPPED _ NIL
CLOSEFILE _ (FUNCTION NIL)
DELETEFILE _ (FUNCTION NIL)
GETFILEINFO _ (FUNCTION NIL)
OPENFILE _ [FUNCTION (LAMBDA (NAME ACCESS RECOG OTHERINFO FDEV)
NAME)]
READPAGES _ (FUNCTION \ILLEGAL.DEVICEOP)
SETFILEINFO _ (FUNCTION NIL)
GENERATEFILES _ (FUNCTION \GENERATENOFILES)
TRUNCATEFILE _ (FUNCTION NIL)
WRITEPAGES _ (FUNCTION \ILLEGAL.DEVICEOP)
GETFILENAME _ [FUNCTION (LAMBDA (NAME RECOG FDEV)
NAME)]
REOPENFILE _ [FUNCTION (LAMBDA (NAME)
NAME)]
EVENTFN _ (FUNCTION NIL)
DIRECTORYNAMEP _ (FUNCTION NIL)
HOSTNAMEP _ (FUNCTION NIL)
BIN _ (FUNCTION \ILLEGAL.DEVICEOP)
BOUT _ (FUNCTION \DSPPRINTCHAR)
PEEKBIN _ (FUNCTION \ILLEGAL.DEVICEOP)
BACKFILEPTR _ (FUNCTION \PAGEDBACKFILEPTR)
BLOCKIN _ (FUNCTION \ILLEGAL.DEVICEOP)
BLOCKOUT _ (FUNCTION \NONPAGEDBOUTS)
DEVICEINFO _ (create DISPLAYSTATE)
WINDOWOPS _ NIL
DEFAULTEXTERNALFORMAT _ :DISPLAY))
(\DEFINEDEVICE NIL \8DISPLAYFDEV])

```

(24DISPLAYINIT

; Edited 25-Sep-2021 18:44 by rmk:

```

[LAMBDA NIL
  (DECLARE (GLOBALVARS \24DISPLAYIMAGEOPS \24DISPLAYFDEV)
    (SETQ \24DISPLAYIMAGEOPS (create IMAGEOPS
      IMAGETYPE _ '24DISPLAY
      IMFONT _ (FUNCTION \DSPFONT.DISPLAY)
      IMLEFTMARGIN _ (FUNCTION \DSPLEFTMARGIN.DISPLAY)
      IMRIGHTMARGIN _ (FUNCTION \DSPRIGHTMARGIN.DISPLAY)
      IMLINEFEED _ (FUNCTION \DSPLINEFEED.DISPLAY)
      IMXPOSITION _ (FUNCTION \DSPXPOSITION.DISPLAY)
      IMYPOSITION _ (FUNCTION \DSPYPOSITION.DISPLAY)
      IMCLOSEFN _ (FUNCTION NIL)
      IMDRAWCURVE _ (FUNCTION \DRAWCURVE.DISPLAY)
      IMFILLCIRCLE _ '\FILLCIRCLE.DISPLAY
      IMDRAWLINE _ (FUNCTION \DRAWLINE.DISPLAY)
      IMDRAWELLIPSE _ (FUNCTION \DRAWELLIPSE.DISPLAY)
      IMDRAWCIRCLE _ (FUNCTION \DRAWCIRCLE.DISPLAY)
      IMBITBLT _ (FUNCTION \BITBLT.DISPLAY)
      IMBLTSHADE _ (FUNCTION \BLTSHADE.DISPLAY)
      IMNEWPAGE _ (FUNCTION \NEWPAGE.DISPLAY)
      IMSCALE _ [FUNCTION (LAMBDA NIL 1]
      IMSPACEFACTOR _ (FUNCTION \DSPSPACEFACTOR.DISPLAY)
      IMFONTCREATE _ '24DISPLAY
      IMCOLOR _ (FUNCTION \DPCOLOR.DISPLAY)
      IMBACKCOLOR _ (FUNCTION \DSPBACKCOLOR.DISPLAY)
      IMOPERATION _ (FUNCTION \DSPOPERATION.DISPLAY)
      IMSTRINGWIDTH _ (FUNCTION \STRINGWIDTH.DISPLAY)
      IMCHARWIDTH _ (FUNCTION \CHARWIDTH.DISPLAY)
      IMCLIPPINGREGION _ (FUNCTION \DSPCLIPPINGREGION.DISPLAY)
      IMRESET _ (FUNCTION \DSPRESET.DISPLAY)
      IMDRAWARC _ (FUNCTION \DRAWARC.DISPLAY)
      IMDRAWPOLYGON _ (FUNCTION \DRAWPOLYGON.DISPLAY)
      IMDRAWPOINT _ (FUNCTION \DRAWPOINT.DISPLAY)
      IMBLTCHAR _ (FUNCTION \MEDW.BLTCHAR)
      IMXOFFSET _ (FUNCTION \MEDW.XOFFSET)
      IMYOFFSET _ (FUNCTION \MEDW.YOFFSET)))
    (SETQ \24DISPLAYFDEV (create FDEV
      DEVICENAME _ '24DISPLAY
      RESETABLE _ NIL
      RANDOMACCESSP _ NIL
      PAGEMAPPED _ NIL
      CLOSEFILE _ (FUNCTION NIL)
      DELETEFILE _ (FUNCTION NIL)
      GETFILEINFO _ (FUNCTION NIL)
      OPENFILE _ [FUNCTION (LAMBDA (NAME ACCESS RECOG OTHERINFO FDEV)
        NAME]
      READPAGES _ (FUNCTION \ILLEGAL.DEVICEOP)
      SETFILEINFO _ (FUNCTION NIL)
      GENERATEFILES _ (FUNCTION \GENERATENOFILES)
      TRUNCATEFILE _ (FUNCTION NIL)
      WRITEPAGES _ (FUNCTION \ILLEGAL.DEVICEOP)
      GETFILENAME _ [FUNCTION (LAMBDA (NAME RECOG FDEV)
        NAME]
      REOPENFILE _ [FUNCTION (LAMBDA (NAME)
        NAME]
      EVENTFN _ (FUNCTION NIL)
      DIRECTORYNAMEP _ (FUNCTION NIL)
      HOSTNAMEP _ (FUNCTION NIL)
      BIN _ (FUNCTION \ILLEGAL.DEVICEOP)
      BOUT _ (FUNCTION \DSPPRINTCHAR)
      PEEKBIN _ (FUNCTION \ILLEGAL.DEVICEOP)
      BACKFILEPTR _ (FUNCTION \PAGEDBACKFILEPTR)
      BLOCKIN _ (FUNCTION \ILLEGAL.DEVICEOP)
      BLOCKOUT _ (FUNCTION \NONPAGEDBOUTS)
      DEVICEINFO _ (create DISPLAYSTATE)
      WINDOWOPS _ NIL
      DEFAULTTEXTINTERNALFORMAT _ :DISPLAY))
    (\DEFINEDEVICE NIL \24DISPLAYFDEV])

```

(DISPLAYSTREAMTYPEBPP

(\* kbr%: " 6-Feb-86 18:14")

```

[LAMBDA (DISPLAYSTREAMTYPE)
  (SELECTQ DISPLAYSTREAMTYPE
    (DISPLAY 1)
    (4DISPLAY 4)
    (8DISPLAY 8)
    (24DISPLAY 24)
    (SHOULDNT])
)

```

(ADDTOVAR IMAGESTREAMTYPES

```

(DISPLAY (OPENSTREAM OPENDISPLAYSTREAM)
  (FONTCREATE \CREATEDISPLAYFONT)
  (FONTSAVAILABLE \SEARCHDISPLAYFONTFILES))

```



```
(4DISPLAY (OPENSTREAM OPENDISPLAYSTREAM)
  (FONTCREATE \CREATEDISPLAYFONT)
  (FONTSAVAILABLE \SEARCHDISPLAYFONTFILES))
(8DISPLAY (OPENSTREAM OPENDISPLAYSTREAM)
  (FONTCREATE \CREATEDISPLAYFONT)
  (FONTSAVAILABLE \SEARCHDISPLAYFONTFILES))
(24DISPLAY (OPENSTREAM OPENDISPLAYSTREAM)
  (FONTCREATE \CREATEDISPLAYFONT)
  (FONTSAVAILABLE \SEARCHDISPLAYFONTFILES)))
```

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

```
(GLOBALVARS DisplayFDEV \4DISPLAYFDEV \8DISPLAYFDEV \24DISPLAYFDEV)
)
```

```
(DECLARE%: DONTEVAL@LOAD DOCOPY
```

```
(\DISPLAYINIT)
```

```
(\4DISPLAYINIT)
```

```
(\8DISPLAYINIT)
```

```
(\24DISPLAYINIT)
```

```
)
```

```
(DECLARE%: DONTEVAL@LOAD DOEVAL@COMPILE DONTCOPY COMPILERVARS
```

```
(ADDTOVAR NLAMA )
```

```
(ADDTOVAR NLAML )
```

```
(ADDTOVAR LAMA IMAGESTREAMP)
```

```
)
```

---

**FUNCTION INDEX**

BITMAPIMAGESIZE .....	7	DSPCOLOR .....	4	DSPTRANSLATE .....	5	\4DISPLAYINIT .....	14
BRUSHP .....	10	DSPDEFAULTSTATE .....	5	DSPXPOSITION .....	4	\8DISPLAYINIT .....	15
DASHINGP .....	11	DSPFONT .....	4	DSPYPOSITION .....	5	\DisplayEventFn .....	13
DRAWARC .....	6	DSPLEFTMARGIN .....	4	FILLCIRCLE .....	7	\DISPLAYINIT .....	13
DRAWBETWEEN .....	5	DSPLINEFEED .....	4	FILLPOLYGON .....	6	\DISPLAYSTREAMTYPEBPP ..	16
DRAWCIRCLE .....	6	DSPNEWPAGE .....	5	IMAGESTREAMP .....	1	\DRAWCIRCLE.GENERIC .....	8
DRAWCURVE .....	6	DSPOPERATION .....	4	IMAGESTREAMTYPE .....	1	\DRAWELLIPSE.GENERIC .....	8
DRAWDASHEDLINE .....	3	DSPPOPSTATE .....	5	IMAGESTREAMTYPEP .....	1	\DRAWPOINT.GENERIC .....	7
DRAWELLIPSE .....	6	DSPPUSHSTATE .....	5	INSURE.BRUSH .....	9	\DRAWPOLYGON.GENERIC .....	7
DRAWLINE .....	6	DSPRESET .....	4	INSURE.DASHING .....	11	\GOOD.DASHLST .....	2
DRAWPOINT .....	6	DSPRIGHTMARGIN .....	4	MOVETO .....	7	\IMAGEIOINIT .....	9
DRAWPOLYGON .....	6	DSPROTATE .....	5	NEGSHADE .....	10	\NOIMAGE.DSPFONT .....	9
DRAWTO .....	7	DSPSCALE .....	4	OPENIMAGESTREAM .....	2	\POSSIBLECOLOR .....	10
DSPBACKCOLOR .....	3	DSPSCALE2 .....	5	RELDRAWTO .....	7	\UNIMPIMAGEOP .....	9
DSPBOTTOMMARGIN .....	3	DSPSPACEFACTOR .....	4	SCALEDBITBLT .....	7		
DSPCLIPPINGREGION .....	4	DSPTOPMARGIN .....	4	\24DISPLAYINIT .....	16		

---

**VARIABLE INDEX**

IMAGESTREAMTYPES .....	3,16	\COLORDISPLAYSTREAMTYPES .....	13
SYSTEMRECLST .....	12	\DISPLAYSTREAMTYPES .....	13

---

**RECORD INDEX**

BRUSH .....	11	IMAGEOPS .....	11
-------------	----	----------------	----

---

**MACRO INDEX**

IMAGEOP .....	11
---------------	----

---

**CONSTANT INDEX**

MICASPERPT .....	11
------------------	----

---