;;

```
(RPAQQ IMAGEOBJCOMS
       ((COMS ;; Bit-map image objects
              (FNS BITMAPTEDITOBJ COERCETOBITMAP WINDOWTITLEFONT \PRINTBINARYBITMAP \READBINARYBITMAP)
              ;; fns for the bitmap tedit object.
              (FNS BMOBJ.BUTTONEVENTINFN BMOBJ.COPYFN BMOBJ.DISPLAYFN BMOBJ.IMAGEBOXFN BMOBJ.PUTFN BMOBJ.INIT
                   BMOBJ.GETFN5 BMOBJ.CREATE.MENU)
              (INITVARS (*SMALLSCREEN* (ILESSP SCREENWIDTH 700))
                        (*SMALLSCREENFACTOR* 0.5))
              (FNS SCALED.BITMAP.GETFN BMOBJ.GETFN BMOBJ.GETFN2 BMOBJ.GETFN3 BMOBJ.GETFN4)
                                                   ; GETFNs for backward compatibility with older objects.
              (RECORDS BITMAPOBJ)
              [INITVARS (DEFAULT.BITMAP.SCALE '(0.75 (DISPLAY 1]
              ;; make ^O be a character that inserts an object read from the user.
              (GLOBALVARS (BITMAP.OBJ.MENU))
              (ADDVARS (BackgroundCopyMenuCommands (SNAP (FUNCTION (BITMAPOBJ.SNAPW))
                                                        "prompts for an area of the screen to insert.")
                       ("Snap 50%%" (FUNCTION (BITMAPOBJ.SNAPW 0.5))
                                "prompts for an area of the screen to insert, scaled down by 50%%.")
                       ("Snap 50%% & Save" (FUNCTION (BITMAPOBJ.SNAPW 0.5 T))
                                "prompts for an area of the screen to save in *INSERT-BITMAP*, scaled down
                                by 50.")
                       ("Insert saved" (FUNCTION (COPYINSERT *INSERT-BITMAP*))
                                "Inserts *INSERT-BITMAP* in a document"))
              (IMAGEOBJGETFNS (BMOBJ.GETFN))
              (IMAGEOBJGETFNS (BMOBJ.GETFN2))
              (IMAGEOBJGETFNS (BMOBJ.GETFN3))
              (IMAGEOBJGETFNS (BMOBJ.GETFN4))
              (IMAGEOBJGETFNS (BMOBJ.GETFN5))
              (IMAGEOBJGETFNS (SCALED.BITMAP.GETFN)))
       (VARS (BackgroundCopyMenu))
       (FNS GET.OBJ.FROM.USER BITMAPOBJ.SNAPW PROMPTFOREVALED)
       (DECLARE%: DONTEVAL@LOAD DOCOPY (P (BMOBJ.INIT)))
       (FILES EDITBITMAP)))))
```

;; Bit-map image objects

```
(DEFINEQ
```

(BITMAPTEDITOBJ

```
  [LAMBDA (BITMAP SCALEFACTOR ROTATION DESCENT)          ; Edited 13-Aug-93 17:17 by rmk:
                                                         ; Edited  6-Jan-89 16:34 by jds
```

;; returns the IMAGEOBJ which gives the functional information for a bitmap object in a tedit file.

```
    (IMAGEOBJCREATE (CREATE BITMAPOBJ
                           BITMAP _ BITMAP
                           BMOBJSCALEFACTOR _ (OR SCALEFACTOR 1)
                           BMOBJROTATION _ (OR ROTATION 0)
                           BMOBJDESCENT _ (OR DESCENT 0))
              BITMAPIMAGEFNS])
```

(COERCETOBITMAP

```
  [LAMBDA (BMSPEC)                                        ; Edited 11-Jun-90 16:28 by mitani
                                                          (* tries to interpret X as a spec for a bitmap.)
    (PROG (BM CR)
          (RETURN (COND
                    ((BITMAPP BMSPEC)
                     BMSPEC)
                    [(LITATOM BMSPEC)                      (* use value.)
                     (COND
                       ((BITMAPP (EVALV BMSPEC 'COERCETOBITMAP]
                       ((REGIONP BMSPEC)                   (* if BMSPEC is a region, treat it as a region of the screen.)
                       [SETQ BM (BITMAPCREATE (fetch (REGION WIDTH) of BMSPEC)
                                             (fetch (REGION HEIGHT) of BMSPEC)
                                             (BITSPERPIXEL (SCREENBITMAP]
                       (BITBLT (SCREENBITMAP)
```

```
                                    (fetch (REGION LEFT) of BMSPEC)
                                    (fetch (REGION BOTTOM) of BMSPEC)
                                    BM 0 0 NIL NIL 'INPUT 'REPLACE)
                          BM)
                        ((WINDOWP BMSPEC)
                         [SETQ BM (BITMAPCREATE (WINDOWPROP BMSPEC 'WIDTH)
                                        (WINDOWPROP BMSPEC 'HEIGHT]
                                                                            (* open the window and bring it to the top.)
                         (TOTOPW BMSPEC)
                         (SETQ CR (DSPCLIPPINGREGION NIL BMSPEC))
                         (BITBLT BMSPEC (fetch (REGION LEFT) of CR)
                                    (fetch (REGION BOTTOM) of CR)
                                    BM 0 0 (fetch (REGION WIDTH) of CR)
                                    (fetch (REGION HEIGHT) of CR))
                         BM])
```

₍**WINDOWTITLEFONT**
```
  [LAMBDA (FONT)                                                (* rrb " 1-Feb-84 15:26")
                                                                (* reset type of function that changes the title font)

    (DSPFONT FONT WindowTitleDisplayStream])
```

₍**PRINTBINARYBITMAP**
```
  [LAMBDA (BITMAP STREAM)                                       (* rrb "23-Jul-84 15:16")

        (* * prints the representation of a bitmap onto STREAM in a form that can be read back by \READBINARYBITMAP.)

    (PROG ((STREAM (GETSTREAM STREAM 'OUTPUT))
            BMH)
         (OR (BITMAPP BITMAP)
             (\ILLEGAL.ARG BITMAP))
         (\WOUT STREAM (BITMAPWIDTH BITMAP))
         (\WOUT STREAM (SETQ BMH (BITMAPHEIGHT BITMAP)))
         (\WOUT STREAM (BITSPERPIXEL BITMAP))
         (\BOUTS STREAM (fetch (BITMAP BITMAPBASE) of BITMAP)
                0
                (ITIMES (fetch (BITMAP BITMAPRASTERWIDTH) of BITMAP)
                     BMH BYTESPERWORD))
         (RETURN BITMAP])
```

₍**READBINARYBITMAP**
```
  [LAMBDA (STREAM)                                              (* rrb "23-Jul-84 15:17")

        (* * reads a bitmap printed on STREAM by \PRINTBINARYBITMAP.)

    (SETQ STREAM (GETSTREAM STREAM 'INPUT))
    (PROG ((BMW (\WIN STREAM))
            (BMH (\WIN STREAM))
            (BPP (\WIN STREAM))
            BITMAP)
         (SETQ BITMAP (BITMAPCREATE BMW BMH BPP))
         (\BINS STREAM (fetch (BITMAP BITMAPBASE) of BITMAP)
                0
                (ITIMES (fetch (BITMAP BITMAPRASTERWIDTH) of BITMAP)
                     BMH BYTESPERWORD))
         (RETURN BITMAP])
)
```

;; fns for the bitmap tedit object.

(DEFINEQ

₍**BMOBJ.BUTTONEVENTINFN**
```
  [LAMBDA (IMAGEOBJ WINDOW SELECTION X Y SELWINDOW TEXTSTREAM BUTTON OPERATION)
                                                             ; Edited 14-Aug-93 19:44 by rmk:
                                                             ; Edited 13-Jan-89 17:41 by jds
```

;;; the user has pressed a button inside the bitmap object IMAGEOBJ. Bring up a menu of bitmap edit operations.

```
    (PROG* ((OBJ (IMAGEOBJPROP IMAGEOBJ 'OBJECTDATUM))
            (OLDSCALE (FETCH (BITMAPOBJ BMOBJSCALEFACTOR) OF OBJ))
             NEW.BITMAP COMMAND.MENU COMMAND PREVIOUS.BITMAP NAME TEMP X Y)
           (COND
              ([OR (EQ BUTTON 'RIGHT)
                   (AND OPERATION (NEQ OPERATION 'NORMAL]    ;  If he's extending a selection, or is selecting for
                                                             ; move/copy/delete, DON'T bring up the bitmap editing menu!
                 (RETURN)))
           (SETQ PREVIOUS.BITMAP (BITMAPCOPY (FETCH (BITMAPOBJ BITMAP) OF OBJ)))
           (SETQ NEW.BITMAP
            (SELECTQ [MENU (COND
                             ((TYPE? MENU BITMAP.OBJ.MENU)
                              BITMAP.OBJ.MENU)
                             (T (SETQ BITMAP.OBJ.MENU (BMOBJ.CREATE.MENU]
                   (CHANGE.SCALE
```

```
                                  ;; Change the scale on the bitmap.  Since scale can be a list, might be better to use list-reading instead of
                                  ;; string-reading functions, but...
                                  (LET [(NEWSCALE (COND
                                                    ((TEDITWINDOWP WINDOW)
                                                     (TEDIT.GETINPUT (TEXTOBJ WINDOW)
                                                        "Scale Factor:  " OLDSCALE))
                                                    (T (PROMPTFORWORD "Scale Factor:  " OLDSCALE NIL PROMPTWINDOW]
                                        (IF [AND NEWSCALE [NLSETQ (SETQ NEWSCALE (READ (OPENSTRINGSTREAM NEWSCALE
                                                                                          'INPUT]
                                                (NOT (EQUAL NEWSCALE OLDSCALE))
                                                (OR (NUMBERP NEWSCALE)
                                                    (AND (NUMBERP (CAR (LISTP NEWSCALE)))
                                                         (FOR X IN (CDR NEWSCALE) ALWAYS (NUMBERP (CADR X]
                                            THEN (REPLACE (BITMAPOBJ BMOBJSCALEFACTOR) OF OBJ WITH NEWSCALE)
                                                              ; Return the prevous bitmap, so we don't change the bits.
                                                    PREVIOUS.BITMAP
                                            ELSE (RETURN NIL))))
                     (HAND.EDIT (EDITBM PREVIOUS.BITMAP))
                     (TRIM (TRIM.BITMAP PREVIOUS.BITMAP))
                     (INVERT.HORIZONTALLY
                          (INVERT.BITMAP.HORIZONTALLY PREVIOUS.BITMAP))
                     (INVERT.VERTICALLY
                          (INVERT.BITMAP.VERTICALLY PREVIOUS.BITMAP))
                     (INVERT.DIAGONALLY
                          (INVERT.BITMAP.DIAGONALLY PREVIOUS.BITMAP))
                     (ROTATE.BITMAP.LEFT
                          (ROTATE.BITMAP.LEFT PREVIOUS.BITMAP))
                     (ROTATE.BITMAP.RIGHT
                          (ROTATE.BITMAP.RIGHT PREVIOUS.BITMAP))
                     (SHIFT.LEFT (INTERACT&SHIFT.BITMAP.LEFT PREVIOUS.BITMAP))
                     (SHIFT.RIGHT (INTERACT&SHIFT.BITMAP.RIGHT PREVIOUS.BITMAP))
                     (SHIFT.DOWN (INTERACT&SHIFT.BITMAP.DOWN PREVIOUS.BITMAP))
                     (SHIFT.UP (INTERACT&SHIFT.BITMAP.UP PREVIOUS.BITMAP))
                     (INTERCHANGE.BLACK/WHITE
                          (INVERT.BITMAP.B/W PREVIOUS.BITMAP))
                     (ADD.BORDER (INTERACT&ADD.BORDER.TO.BITMAP PREVIOUS.BITMAP))
                     (RETURN NIL)))
             (REPLACE (BITMAPOBJ BITMAP) OF OBJ WITH NEW.BITMAP)
             (IMAGEOBJPROP IMAGEOBJ 'CACHED.BITMAP NIL)            ; And clear any cached shrunk bitmaps so the display looks
                                                                   ; reasonable.

             (RETURN 'CHANGED])
```

(**BMOBJ.COPYFN**

```
  [LAMBDA (IMAGEOBJ)                                              ; Edited 13-Aug-93 17:13 by rmk:
                                                                  ; Edited  6-Jan-89 16:19 by jds

      ;; makes a copy of a bitmap image object.

      (LET [(BMOBJ (IMAGEOBJPROP IMAGEOBJ 'OBJECTDATUM]
           (BITMAPTEDITOBJ (BITMAPCOPY (FETCH (BITMAPOBJ BITMAP) OF BMOBJ))
                 (FETCH (BITMAPOBJ BMOBJSCALEFACTOR) OF BMOBJ)
                 (FETCH (BITMAPOBJ BMOBJROTATION) OF BMOBJ)
                 (FETCH (BITMAPOBJ BMOBJDESCENT) OF BMOBJ])
```

(**BMOBJ.DISPLAYFN**

```
  [LAMBDA (IMAGEOBJ IMAGE.STREAM)                                 ; Edited  7-Dec-95 13:20 by
                                                                 ; Edited 13-Aug-93 17:49 by rmk:
                                                                 ; Edited 29-Mar-89 18:38 by snow

      ;; Display a bitmap IMAGEOBJ on IMAGE.STREAM.  Scales and rotates it if appropriate, and moves it down by DESCENT.

      (DECLARE (GLOBALVARS *SMALLSCREEN* *SMALLSCREENFACTOR*))
      (PROG ([FACTOR (fetch (BITMAPOBJ BMOBJSCALEFACTOR) of (IMAGEOBJPROP IMAGEOBJ 'OBJECTDATUM]
             [BITMAP (fetch (BITMAPOBJ BITMAP) of (IMAGEOBJPROP IMAGEOBJ 'OBJECTDATUM]
             (CACHE (IMAGEOBJPROP IMAGEOBJ 'CACHED.BITMAP))
             [DESCENT (fetch (BITMAPOBJ BMOBJDESCENT) of (IMAGEOBJPROP IMAGEOBJ 'OBJECTDATUM]
             (STREAM-SCALE (DSPSCALE NIL IMAGE.STREAM))
             (STREAMTYPE (IMAGESTREAMTYPE IMAGE.STREAM))
             SHRUNK.BITMAP)
          (RELMOVETO 0 [IMINUS (FIXR (FTIMES STREAM-SCALE (OR DESCENT 0]
                 IMAGE.STREAM)
          [IF (NUMBERP FACTOR)
            ELSEIF (LISTP FACTOR)
              THEN (SETQ FACTOR (OR (CADR (ASSOC STREAMTYPE (CDR FACTOR)))
                                    (CAR FACTOR]
          (IF (AND *SMALLSCREEN* *SMALLSCREENFACTOR* (GREATERP FACTOR 0.5)
                  (LEQ FACTOR 1.0)
                  (EQ 'DISPLAY STREAMTYPE))
              THEN  ;; Shrink images on small screens, unless they are already small or specified to be big

                  (SETQ FACTOR *SMALLSCREENFACTOR*))
          (SELECTQ STREAMTYPE
              ((DISPLAY PRESS)  ;; PRESS and  DISPLAY prints the junky shrunk bitmap.  This is strange:  this presumably should be handled
                                ;; in the device's bitblt method.

                  (COND
                     ((NOT (SETQ SHRUNK.BITMAP CACHE))
```

```
                         [COND
                            [(LEQ FACTOR 1.0)                                  ; We're shrinking the bitmap.  Create a shrunk image for display
                               (SETQ SHRUNK.BITMAP (SHRINKBITMAP BITMAP (FQUOTIENT 1.0 FACTOR)
                                                                         (FQUOTIENT 1.0 FACTOR]
                            (T                                                 ; We're expanding it.  Create a bigger one.
                               (SETQ SHRUNK.BITMAP (EXPANDBITMAP BITMAP FACTOR FACTOR]
                            (IMAGEOBJPROP IMAGEOBJ 'CACHED.BITMAP SHRUNK.BITMAP)))
                         [BITBLT SHRUNK.BITMAP NIL NIL IMAGE.STREAM (DSPXPOSITION NIL IMAGE.STREAM)
                                 (DSPYPOSITION NIL IMAGE.STREAM)
                                 (FIXR (FTIMES FACTOR (BITMAPWIDTH BITMAP)))
                                 (FIXR (FTIMES FACTOR (BITMAPHEIGHT BITMAP)])
                   (PROGN
                                 ;; This is the default case--Call SCALEDBITBLT

                                 ;; changed OPERATION from PAINT to REPLACE as PAINT doesn't work for all devices. --was.  From rmk:  if a device
                                 ;; can't implement PAINT properly, then IT should coerce to REPLACE.  Why is that done here?

                          (SCALEDBITBLT BITMAP 0 0 IMAGE.STREAM NIL NIL (BITMAPWIDTH BITMAP)
                                 (BITMAPHEIGHT BITMAP)
                                 'INPUT
                                 'REPLACE NIL NIL FACTOR])
```

(**BMOBJ.IMAGEBOXFN**

```
  [LAMBDA (IMAGEOBJ IMAGE.STREAM CURRENT.X RIGHT.MARGIN)              ; Edited  7-Dec-95 13:20 by
                                                                     ; Edited  6-Dec-95 15:17 by
                                                                     ; Edited  4-Dec-95 13:35 by
                                                                     ; Edited  4-Dec-95 13:29 by
                                                                     ; Edited 13-Aug-93 17:48 by rmk:
                                                                     ; Edited  6-Jan-89 16:35 by jds

      ;; returns an imagebox describing the size of the scaled bitmap
    (DECLARE (GLOBALVARS *SMALLSCREEN* *SMALLSCREENFACTOR*))
    (LET* ((BITMAPOBJ (IMAGEOBJPROP IMAGEOBJ 'OBJECTDATUM))
           (FACTOR (FETCH (BITMAPOBJ BMOBJSCALEFACTOR) OF BITMAPOBJ))
           (BITMAP (FETCH (BITMAPOBJ BITMAP) OF BITMAPOBJ))
           (DESCENT (FETCH (BITMAPOBJ BMOBJDESCENT) OF BITMAPOBJ))
           (SCALE (DSPSCALE NIL IMAGE.STREAM))
           (STREAMTYPE (IMAGESTREAMTYPE IMAGE.STREAM))
           WIDTH HEIGHT)
          [COND
             ((EQ BITMAP 'NoneCached)
              (SETQ WIDTH (SETQ HEIGHT 5)))
             (T [IF (NUMBERP FACTOR)
                    ELSEIF (LISTP FACTOR)
                      THEN (SETQ FACTOR (OR (CADR (ASSOC STREAMTYPE (CDR FACTOR)))
                                            (CAR FACTOR]
                  (IF (AND *SMALLSCREEN* *SMALLSCREENFACTOR* (GREATERP FACTOR 0.5)
                           (LEQ FACTOR 1.0)
                           (EQ 'DISPLAY STREAMTYPE))
                      THEN ;; Shrink images on small screens, unless they are already small or specified to be big

                         (SETQ FACTOR *SMALLSCREENFACTOR*))
                 [SETQ WIDTH (FIXR (FTIMES SCALE (TIMES (BITMAPWIDTH BITMAP)
                                                        FACTOR]
                 (SETQ HEIGHT (FIXR (FTIMES SCALE (TIMES (BITMAPHEIGHT BITMAP)
                                                        FACTOR]
          (CREATE IMAGEBOX
                  XSIZE _ WIDTH
                  YSIZE _ HEIGHT
                  YDESC _ (OR DESCENT 0)
                  XKERN _ 0])
```

(**BMOBJ.PUTFN**

```
  [LAMBDA (BMOBJ STREAM)                                             ; Edited 13-Aug-93 15:41 by rmk:
                                                                     ; Edited 11-Jan-89 17:00 by jds

      ;; Put a description of a bitmap object into the file, including all fields as s-expressions.  To be read by BMOBJ.GETFN5

    (LET* [(BITMAPOBJ (IMAGEOBJPROP BMOBJ 'OBJECTDATUM]
          (\PRINTBINARYBITMAP (fetch (BITMAPOBJ BITMAP) of BITMAPOBJ)
                 STREAM)
          (PRIN2 (fetch (BITMAPOBJ BMOBJSCALEFACTOR) of BITMAPOBJ)
                 STREAM FILERDTBL)
          (SPACES 1 STREAM)
          (PRIN2 (fetch (BITMAPOBJ BMOBJROTATION) of BITMAPOBJ)
                 STREAM FILERDTBL)
          (SPACES 1 STREAM)
          (PRIN2 (fetch (BITMAPOBJ BMOBJDESCENT) of BITMAPOBJ)
                 STREAM FILERDTBL)
          (SPACES 1 STREAM])
```

(**BMOBJ.INIT**

```
  [LAMBDA NIL                                                        ; Edited 13-Aug-93 14:27 by rmk:
                                                                     ; Edited 11-Jan-89 17:01 by jds

      ;; returns the function vector which gives the functional information for a bitmap image object.

    (SETQ BITMAPIMAGEFNS (IMAGEFNSCREATE (FUNCTION BMOBJ.DISPLAYFN)
```

```
                                        (FUNCTION BMOBJ.IMAGEBOXFN)
                                        (FUNCTION BMOBJ.PUTFN)
                                        (FUNCTION BMOBJ.GETFN5)
                                        (FUNCTION BMOBJ.COPYFN)
                                        (FUNCTION BMOBJ.BUTTONEVENTINFN)
                                        (FUNCTION NILL)
                                        (FUNCTION NILL)
                                        (FUNCTION NILL)
                                        (FUNCTION NILL)
                                        (FUNCTION NILL)
                                        (FUNCTION NILL)
                                        (FUNCTION NILL])
```

(**BMOBJ.GETFN5**

```
  [LAMBDA (INPUT.STREAM TEXTSTREAM)                                    ; Edited 13-Aug-93 15:40 by rmk:
                                                                       (* jds "30-Oct-85 11:29")
                                                                       ; reads in a scaled bitmap object with readbitmap and read.
                                                                       ; Gets scale, rotation, and descent as arbitrary s-expressions

    (BITMAPEDITOBJ (\READBINARYBITMAP INPUT.STREAM)
         (READ INPUT.STREAM FILERDTBL)
         (READ INPUT.STREAM FILERDTBL)
         (READ INPUT.STREAM FILERDTBL])
```

(**BMOBJ.CREATE.MENU**

```
  [LAMBDA NIL                                                          ; Edited 30-Jul-87 19:19 by jds

    ;; Creates the menu that comes up when you button in a bitmap image object.

    (create MENU
         TITLE _ "Operations on bitmaps"
         ITEMS _ '((Change% Scale 'CHANGE.SCALE "Changes the scale factor used at output time.")
                   (Hand% Edit 'HAND.EDIT "Starts the bitmap editor on this bitmap.")
                   (Trim 'TRIM "removes the white space from the edges of the bitmap.")
                   (Reflect% Left-to-right 'INVERT.HORIZONTALLY "inverts the bitmap about the vertical
                           midline.")
                   (Reflect% Top-to-bottom 'INVERT.VERTICALLY "inverts the bitmap about the horizontal
                           midline.")
                   (Reflect% Diagonally 'INVERT.DIAGONALLY "inverts the bitmap about the lower left to upper
                           right diagonal.")
                   (Rotate% Left 'ROTATE.BITMAP.LEFT "rotates the bitmap 90 degrees counter-clockwise.")
                   (Rotate% Right 'ROTATE.BITMAP.RIGHT "rotates the bitmap 90 degrees clockwise.")
                   (│Expand on Right│ 'SHIFT.LEFT "prompts for a number of bits to add on the right.")
                   (│Expand on Left│ 'SHIFT.RIGHT "prompts for a number of bits to add on the left.")
                   (│Expand on Bottom│ 'SHIFT.UP "prompts for a number of bits to add on the top.")
                   (│Expand on Top│ 'SHIFT.DOWN "prompts for a number of bits to add on the bottom.")
                   (│Switch Black & White│ 'INTERCHANGE.BLACK/WHITE "changes all black bits to white and all
                           white bits to black.")
                   (Add% Border 'ADD.BORDER "adds an arbitrary border in an arbitrary shade."))
         CENTERFLG _ T
         CHANGEOFFSETFLG _ 'Y
         MENUOFFSET _ (create POSITION
                             XCOORD _ -1
                             YCOORD _ 0])
)
```

```
(RPAQ? *SMALLSCREEN* (ILESSP SCREENWIDTH 700))

(RPAQ? *SMALLSCREENFACTOR* 0.5)

(DEFINEQ
```

(**SCALED.BITMAP.GETFN**

```
  [LAMBDA (INPUT.STREAM TEXTSTREAM)                                    (* jds "30-Oct-85 11:29")
                                                                       (* reads in a scaled bitmap object with readbitmap and read)

    (PROG (FACTOR BITMAP)
         (SETQ BITMAP (READBITMAP INPUT.STREAM))
         (SETQ FACTOR (READ INPUT.STREAM))
         (RETURN (BITMAPEDITOBJ BITMAP (FQUOTIENT 1.0 FACTOR)
                       0])
```

(**BMOBJ.GETFN**

```
  [LAMBDA (STREAM)                                                     (* rrb "17-Jul-84 11:46")


         (* this is an old version of the get function for bitmap image objects.
         It is left around so old tedit documents will still work. |17/7/84|)

    (RESETFORM (INPUT STREAM)
         (PROG ((FIELDS (READ STREAM))
                (BITMAP (READBITMAP)))
              (RETURN (BITMAPEDITOBJ BITMAP (CAR FIELDS)
                            (CADR FIELDS])
```

(**BMOBJ.GETFN2**

```
  [LAMBDA (STREAM)                                               (* rrb "17-Jul-84 11:29")

          (* * reads a bitmap image object from a file. This version stores the binary data rather than the character representation
          used by READBITMAP.)

      (PROG ((SCALE (\WIN STREAM))
             (ROT (\WIN STREAM)))
            (RETURN (BITMAPTEDITOBJ (\READBINARYBITMAP STREAM)
                          SCALE ROT])
```

(**BMOBJ.GETFN3**
```
  [LAMBDA (STREAM)                                               ; Edited 11-Jan-89 17:03 by jds
```

;;; reads a bitmap image object from a file. This version stores the binary data rather than the character representation used by READBITMAP.

```
      (COND
         ((IEQP (\PEEKBIN STREAM)
                (CHARCODE CR))                                   ; This is an old-format sketch with bitmap included.  Skip the
                                                                 ; interfering CR.
          (BIN STREAM)))
      (PROG ((SCALE (FPLUS (\WIN STREAM)
                           (FQUOTIENT (\WIN STREAM)
                                 32768)))
             (DESC (\WIN STREAM)))
            (RETURN (BITMAPTEDITOBJ (\READBINARYBITMAP STREAM)
                          SCALE 0 DESC])
```

(**BMOBJ.GETFN4**
```
  [LAMBDA (STREAM)                                               ; Edited  6-Jan-89 16:33 by jds
```

;;; reads a bitmap image object from a file. This version stores the binary data rather than the character representation used by READBITMAP.

```
      (COND
         ((IEQP (\PEEKBIN STREAM)
                (CHARCODE CR))                                   ; This is an old-format sketch with bitmap included.  Skip the
                                                                 ; interfering CR.
          (BIN STREAM)))
      (LET ((SCALE (FPLUS (\WIN STREAM)
                          (FQUOTIENT (\WIN STREAM)
                                32768)))
            (ROT (\WIN STREAM))
            (DESCENT (\WIN STREAM)))
```

        ;; Dummy words for later expansion:

```
         (\WIN STREAM)
         (\WIN STREAM)
         (\WIN STREAM)
         (\WIN STREAM)
```

        ;; Now read the bitmap itself and construct the object:

```
         (BITMAPTEDITOBJ (\READBINARYBITMAP STREAM)
               SCALE ROT DESCENT])
```

)

;; GETFNs for backward compatibility with older objects.

```
(DECLARE%: EVAL@COMPILE

(RECORD BITMAPOBJ (;; Describes a bitmap imageobj

                BITMAP                                           ; The bitmap itself
                BMOBJSCALEFACTOR                                 ; The factor to scale it by when displaying
                BMOBJROTATION                                    ; A rotation to apply when displaying
                BMOBJDESCENT                                     ; How far below the base line to display it.  NIL => 0.

                ))
)

(RPAQ? DEFAULT.BITMAP.SCALE '(0.75 (DISPLAY 1)))
```

;; make ^O be a character that inserts an object read from the user.

```
(DECLARE%: DOEVAL@COMPILE DONTCOPY

(GLOBALVARS (BITMAP.OBJ.MENU))
)

(ADDTOVAR BackgroundCopyMenuCommands
        (SNAP (FUNCTION (BITMAPOBJ.SNAPW))
              "prompts for an area of the screen to insert.")
        ("Snap 50%%" (FUNCTION (BITMAPOBJ.SNAP 0.5))
               "prompts for an area of the screen to insert, scaled down by 50%%.")
        ("Snap 50%% & Save" (FUNCTION (BITMAPOBJ.SNAPW 0.5 T))
```

```
                     "prompts for an area of the screen to save in *INSERT-BITMAP*, scaled down by 50.")
          ("Insert saved" (FUNCTION (COPYINSERT *INSERT-BITMAP*))
                     "Inserts *INSERT-BITMAP* in a document"))
```

(ADDTOVAR **IMAGEOBJGETFNS** (BMOBJ.GETFN))

(ADDTOVAR **IMAGEOBJGETFNS** (BMOBJ.GETFN2))

(ADDTOVAR **IMAGEOBJGETFNS** (BMOBJ.GETFN3))

(ADDTOVAR **IMAGEOBJGETFNS** (BMOBJ.GETFN4))

(ADDTOVAR **IMAGEOBJGETFNS** (BMOBJ.GETFN5))

(ADDTOVAR **IMAGEOBJGETFNS** (SCALED.BITMAP.GETFN))

(RPAQQ **BackgroundCopyMenu** NIL)

(DEFINEQ

(**GET.OBJ.FROM.USER**
```
  [LAMBDA (TEXTSTREAM TEXTOBJ)                                          ; Edited 26-Apr-91 10:54 by jds

    ;; reads an expression from the user and puts the result into the textstream.

    (ERSETQ (PROG ((VAL (PROMPTFOREVALED "Form to eval:"))
                   (SEL (fetch (TEXTOBJ SEL) of TEXTOBJ))
                  BM)
                  (CL:TYPECASE VAL
                      (STRINGP                                         ; Atoms and strings get inserted as text.
                          (AND VAL (TEDIT.INSERT TEXTSTREAM VAL SEL)))
                      (LITATOM                                         ; Atoms and strings get inserted as text.
                          (AND VAL (TEDIT.INSERT TEXTSTREAM (MKSTRING VAL T)
                                          SEL)))
                      (IMAGEOBJ                                        ; IMAGEOBJs get inserted as is
                          (TEDIT.INSERT.OBJECT VAL TEXTSTREAM (SELECTQ (fetch POINT of SEL)
                                                                  (LEFT (fetch (SELECTION CH#) of SEL))
                                                                  (RIGHT (fetch (SELECTION CHLIM) of SEL))
                                                                  NIL)))
                      (T (COND
                           ((SETQ BM (COERCETOBITMAP VAL))             ; If it can be coerced to a bitmap, do so, then wrap the bitmap up
                                                                       ; as a nobject
                             (TEDIT.INSERT.OBJECT (BITMAPTEDITOBJ BM 1 0)
                                    TEXTSTREAM
                                    (SELECTQ (fetch POINT of SEL)
                                        (LEFT (fetch (SELECTION CH#) of SEL))
                                        (RIGHT (fetch (SELECTION CHLIM) of SEL))
                                        NIL)))
                           (T                                          ; Not a bitmap, nor one of the special cases above;  complain
                             (AND VAL (TEDIT.INSERT TEXTSTREAM (MKSTRING VAL T)
                                          SEL))                        ; (TEDIT.PROMPTPRINT TEXTOBJ (CONCAT 'Not
                                                                       ; implemented to have ' VAL ' in documents yet.') T)

                         ))))])
```

(**BITMAPOBJ.SNAPW**
```
  [LAMBDA (SCALE SAVE)                                                 ; Edited 14-Aug-93 19:54 by rmk:
                                                                       ; Edited 19-Jan-93 16:08 by jds

         (* * makes an image object of a prompted for region of the screen.)

     (PROG ((REG (GETREGION))
            BM)
           [SETQ BM (BITMAPCREATE (fetch (REGION WIDTH) of REG)
                       (fetch (REGION HEIGHT) of REG)
                       (BITSPERPIXEL (SCREENBITMAP]
           (BITBLT (SCREENBITMAP)
                   (fetch (REGION LEFT) of REG)
                   (fetch (REGION BOTTOM) of REG)
                   BM 0 0 NIL NIL 'INPUT 'REPLACE)
           [COND
              (SAVE (SETQ *INSERT-BITMAP* (BITMAPTEDITOBJ BM (OR SCALE (COPY DEFAULT.BITMAP.SCALE))
                                         0)))
              (T (COPYINSERT (BITMAPTEDITOBJ BM (OR SCALE (COPY DEFAULT.BITMAP.SCALE))
                                         0]
           (RETURN])
```

(**PROMPTFOREVALED**
```
  [LAMBDA (MSG WHERE FONT MINWIDTH MINHEIGHT)                          (* jds "26-Sep-85 16:46")

         (* opens a window with MSG in the title and returns the result of evaluating a READ from that window.
         (PROMPTFOREVALED "HOW'S THIS?" (QUOTE (600 . 600)) NIL 100))

     (PROG [NEWVALUE WIN (FONT (OR FONT (FONTCREATE 'HELVETICA 12 'BOLD]
           (RESETFORM (WINDOWTITLEFONT FONT)
                  (SETQ WIN (CREATEW [COND
```

```
                                          ((REGIONP WHERE)
                                           WHERE)
                                          (T (CREATEREGION (COND
                                                             (WHERE (fetch (POSITION XCOORD) of WHERE))
                                                             (T LASTMOUSEX))
                                                          (COND
                                                            (WHERE (fetch (POSITION YCOORD) of WHERE))
                                                            (T LASTMOUSEY))
                                                          (WIDTHIFWINDOW (MAX (STRINGWIDTH MSG FONT)
                                                                              (OR MINWIDTH 0)
                                                                              125)
                                                             8)
                                                          (HEIGHTIFWINDOW (MAX (ITIMES (FONTPROP (DEFAULTFONT
                                                                                                    'DISPLAY)
                                                                                                 'HEIGHT)
                                                                                       3)
                                                                               (OR MINHEIGHT 0)
                                                                               100)
                                                              T 8]
                                           MSG 4))
                    (CLEARW WIN))
            [RESETFORM (TTYDISPLAYSTREAM WIN)
                    (SETQ NEWVALUE (CAR (ERSETQ (LISPX (LISPXREAD T T)
                                                   '>]
            (CLOSEW WIN)
            (RETURN NEWVALUE])
)

(DECLARE%: DONTEVAL@LOAD DOCOPY

(BMOBJ.INIT)
)

(FILESLOAD EDITBITMAP)

(PUTPROPS IMAGEOBJ COPYRIGHT ("Venue & Xerox Corporation" 1984 1985 1986 1987 1988 1989 1990 1991 1993 1995))
```

FUNCTION INDEX

VARIABLE INDEX

RECORD INDEX