File created: 9-Nov-90 20:59:31 **{DSK}<medley>project3>color>SOURCES>COLOR.;2**

changes to: (FNS MAPOFACOLOR)

previous date: 27-Jan-87 15:56:46 {DSK}<medley>release>alpha>library>COLOR.;1

Read Table: INTERLISP

Package: INTERLISP

Format: XCCS

;;


```
(RPAQQ COLORCOMS
       [(FNS DISPLAYCOLORLEVELS DISPLAYHLSLEVELS HLSLEVEL HLSTORGB HLSVALUEFN HLSVALUEFROMLEVEL
             LEVELFROMHLSVALUE RAINBOWMAP RGBTOHLS)
        (FNS OVERPAINT BITMAPFROMSTRING SHADEBITMAP)
        (INITVARS (EDITCOLORMAP.WINDOW NIL))
        (FNS EDITCOLORMAP EDITCOLORMAP.BUTTONEVENTFN EDITCOLORMAP.REDISPLAYFN EDITCOLORMAP.VALUELEVEL
             EDITCOLORMAP.WINDOWLEVEL CHANGECOLORLEVELS GETCOLOR#FROMUSER GETCOLOR#FROMSCREEN DISPLAYCOLORLEVEL
             FILLINREGION AREAFILL CENTEREDLEFT OUTLINEAREA OUTLINEREGION)
        (FNS ADJUSTCOLORMAP SHOWCOLORBLOCKS MAPOFACOLOR COLORHEXPATTERN)
        (VARS EditColorMapHeight EditColorMapWidth (COLOR#MENUSAVE)
              (CONTROLMENUSAVE)
              (EDIT8BITCOLORMAPMENU)
              (EDIT8BITCOLORMAPNUMBERREADER))
        (GLOBALVARS COLOR#MENUSAVE CONTROLMENUSAVE EDIT8BITCOLORMAPMENU EDIT8BITCOLORMAPNUMBERREADER
               EditColorMapHeight EditColorMapWidth)
        (COMS
```

;;; support for global naming and querying of colors.

```
              (FNS CNSMENUINIT CNSTOCSL CNSTORGB CSLTOCNS DICOLOR.FROM.USER GETCNS HLSTOCSL CSLTOHLS RGBTOCNS)
              (VARS DICOLOR.hueMapping DICOLOR.lightnessMapping DICOLOR.saturationMapping NEWCOLORITEM)
              (INITVARS (COLORNAMEMENU))
              (FNS DICOLOR.hueN DICOLOR.hueNvalue DICOLOR.hueNname DICOLOR.lightnessN DICOLOR.lightnessNvalue
                   DICOLOR.lightnessNname DICOLOR.saturationN DICOLOR.saturationNvalue DICOLOR.saturationNname)
              (DECLARE%: EVAL@LOAD DONTCOPY (*)
                     (RECORDS hueRecord lightnessRecord saturationRecord)
                     (CONSTANTS * DICOLOR.hueConstants)
                     (CONSTANTS * DICOLOR.saturationConstants)
                     (CONSTANTS * DICOLOR.lightnessConstants))
              (P (CNSMENUINIT)))
        (FILES LLCOLOR READNUMBER)
        (P (SETQ EDITBMMENU NIL)
           (MOVD 'ARRAYP 'COLORMAPP])

(DEFINEQ
```

(DISPLAYCOLORLEVELS

```
  [LAMBDA (WINDOW RGB)                                          (* kbr%: " 3-Jun-86 19:45")
    (PROG (HLS)
          (DISPLAYCOLORLEVEL WINDOW 'RED (fetch (RGB RED) of RGB)
                 (fetch (RGB RED) of RGB))
          (DISPLAYCOLORLEVEL WINDOW 'GREEN (fetch (RGB GREEN) of RGB)
                 (fetch (RGB GREEN) of RGB))
          (DISPLAYCOLORLEVEL WINDOW 'BLUE (fetch (RGB BLUE) of RGB)
                 (fetch (RGB BLUE) of RGB))
          (SETQ HLS (RGBTOHLS RGB))
          (DISPLAYCOLORLEVEL WINDOW 'HUE (fetch (HLS HUE) of HLS)
                 (EDITCOLORMAP.WINDOWLEVEL 'HUE (fetch (HLS HUE) of HLS)))
          (DISPLAYCOLORLEVEL WINDOW 'LIGHTNESS (fetch (HLS LIGHTNESS) of HLS)
                 (EDITCOLORMAP.WINDOWLEVEL 'LIGHTNESS (fetch (HLS LIGHTNESS) of HLS)))
          (DISPLAYCOLORLEVEL WINDOW 'SATURATION (fetch (HLS SATURATION) of HLS)
                 (EDITCOLORMAP.WINDOWLEVEL 'SATURATION (fetch (HLS SATURATION) of HLS])
```

(DISPLAYHLSLEVELS

```
  [LAMBDA (HLS WIN)                                             (* rrb "25-OCT-82 14:08")
                                                                (* displays a hue lightness saturation triple in the edit window.)
    (DISPLAYHLSLEVEL HLS 'HUE NIL WIN)
    (DISPLAYHLSLEVEL HLS 'LIGHTNESS NIL WIN)
    (DISPLAYHLSLEVEL HLS 'SATURATION NIL WIN])
```

(HLSLEVEL

```
  [LAMBDA (HLS FIELD NEWLEVEL)                                  (* rrb "25-OCT-82 13:29")
                                                                (* returns the value of the named field from a hue lightness
                                                                saturation record.)
    (SELECTQ FIELD
         (HUE (PROG1 (fetch (HLS HUE) of HLS)
                     (AND NEWLEVEL (replace (HLS HUE) of HLS with NEWLEVEL))))
         (LIGHTNESS (PROG1 (fetch (HLS LIGHTNESS) of HLS)
                           (AND NEWLEVEL (replace (HLS LIGHTNESS) of HLS with NEWLEVEL))))
```

```
        (SATURATION (PROG1 (fetch (HLS SATURATION) of HLS)
                           (AND NEWLEVEL (replace (HLS SATURATION) of HLS with NEWLEVEL))))
        (SHOULDNT])
```

(HLSTORGB

```
  [LAMBDA (HLS LIGHTNESS SATURATION)                                    (* kbr%: " 3-Jun-86 21:16")
```

(* Converts from a hue saturation lightness triple into red green blue triple.
HUE is in range 0 to 360, lightness and saturation are in the range 0 to 1.0 *)

(* This algorithm was taken from siggraph vol 13 number 3 August 1979%: Status report on graphics standards planning committee. *)

```
    (PROG (HUE M1 M2 RGB)
          (COND
             ((LISTP HLS)
              (SETQ HUE (fetch (HLS HUE) of HLS))
              (SETQ LIGHTNESS (fetch (HLS LIGHTNESS) of HLS))
              (SETQ SATURATION (fetch (HLS SATURATION) of HLS)))
             (T (SETQ HUE HLS)))
          [SETQ M1 (COND
                      ((FGREATERP 0.5 LIGHTNESS)
                       (FTIMES LIGHTNESS (FPLUS 1.0 SATURATION)))
                      (T (FDIFFERENCE (FPLUS LIGHTNESS SATURATION)
                                (FTIMES LIGHTNESS SATURATION]
          (SETQ M2 (FDIFFERENCE (FTIMES 2.0 LIGHTNESS)
                          M1))
          [SETQ RGB (create RGB
                           RED _ (HLSVALUEFN M1 M2 HUE)
                           GREEN _ (HLSVALUEFN M1 M2 (IDIFFERENCE HUE 120))
                           BLUE _ (HLSVALUEFN M1 M2 (IDIFFERENCE HUE 240]
          (RETURN RGB])
```

(HLSVALUEFN

```
  [LAMBDA (M1 M2 HUE)                                                   (* kbr%: " 3-Jun-86 20:45")
                                                                        (* Internal value function for converting from HLS to RGB.
                                                                        *)
    (SETQ HUE (IMOD HUE 360))
    (FIX (FTIMES (COND
                    ((ILESSP HUE 60)
                     M1)
                    [(ILESSP HUE 120)
                     (FPLUS M1 (FTIMES (FQUOTIENT (FDIFFERENCE HUE 60)
                                                  60)
                                       (FDIFFERENCE M2 M1]
                    ((ILESSP HUE 240)
                     M2)
                    [(ILESSP HUE 300)
                     (FPLUS M2 (FTIMES (FQUOTIENT (FDIFFERENCE HUE 240)
                                                  60)
                                       (FDIFFERENCE M1 M2]
                    (T M1))
                 255])
```

(HLSVALUEFROMLEVEL

```
  [LAMBDA (HLS LEVEL)                                                   (* rrb "25-OCT-82 13:26")
```

(* returns the scaled value of the hls marker on a scale from 0 to 255)

```
    (SELECTQ HLS
         (HUE (IQUOTIENT (ITIMES LEVEL 360)
                         255))
         (FQUOTIENT LEVEL 255])
```

(LEVELFROMHLSVALUE

```
  [LAMBDA (HLS LEVEL)                                                   (* rrb "25-OCT-82 14:06")
```

(* returns the level on a scale from 0 to 255 that this value would have.)

```
    (SELECTQ HLS
         (HUE (IQUOTIENT (ITIMES LEVEL 255)
                         360))
         (FIX (FTIMES LEVEL 255])
```

(RAINBOWMAP

```
  [LAMBDA (NBITS)                                                       (* rrb "21-OCT-82 18:14")
    [OR NBITS (NULL (COLORDISPLAYP))
        (SETQ NBITS (COLORMAPBITS (SCREENCOLORMAP]
    (COLORMAPCREATE (COND
                       [(EQ NBITS 8)
                        (PROG (MAXINTENSITY MINVISIBLERED MINVISIBLEBLUE MINVISIBLEGREEN NSTEPS REDSTEPSIZE
                                    GREENSTEPSIZE BLUESTEPSIZE)
```

```
                              (SETQ MAXINTENSITY 255)
                              (SETQ MINVISIBLERED 69)
                              (SETQ MINVISIBLEBLUE 38)
                              (SETQ MINVISIBLEGREEN 38)
                              (SETQ NSTEPS (IQUOTIENT (EXPT 2 NBITS)
                                                      8))
```

(* determine how many steps are available for each transition from one color to the next.
There are 8 such transitions. red up, green up, red down, blue up, green down, red up, green up, all down)

(* minimum visible intensity values were emperically determined but will differ depending upon the brightness setting of the individual display. They are also diddled to make the numer of steps come out right.)

```
                              (RETURN (NCONC (for I from MINVISIBLERED to MAXINTENSITY
                                                by (SETQ REDSTEPSIZE (IQUOTIENT (IPLUS (IDIFFERENCE MAXINTENSITY
                                                                                                    MINVISIBLERED)
                                                                                       NSTEPS -2)
                                                                                NSTEPS))
                                                collect                 (* red up)
                                                        (LIST I 0 0))
                                             (for I from MINVISIBLEGREEN to MAXINTENSITY
                                                by (SETQ GREENSTEPSIZE (IQUOTIENT (IPLUS (IDIFFERENCE
                                                                                            MAXINTENSITY
                                                                                            MINVISIBLEGREEN)
                                                                                          -1 NSTEPS)
                                                                                   NSTEPS))
                                                collect                 (* GREEN UP)
                                                        (LIST 255 I 0))
                                             (for I from REDSTEPSIZE to (IDIFFERENCE MAXINTENSITY MINVISIBLERED)
                                                by REDSTEPSIZE collect
                                                                        (* red down)
                                                             (LIST (IDIFFERENCE MAXINTENSITY I)
                                                                   255 0))
                                             (CONS '(0 255 0))
                                             (for I from MINVISIBLEBLUE to MAXINTENSITY
                                                by (SETQ BLUESTEPSIZE (IQUOTIENT (IPLUS (IDIFFERENCE
                                                                                            MAXINTENSITY
                                                                                            MINVISIBLEBLUE)
                                                                                        -1 NSTEPS)
                                                                                  NSTEPS))
                                                collect                 (* BLUE UP)
                                                        (LIST 0 255 I))
                                             (for I from GREENSTEPSIZE to (IDIFFERENCE MAXINTENSITY
                                                                                       MINVISIBLEGREEN)
                                                by GREENSTEPSIZE collect
                                                                        (* GREEN down)
                                                        (LIST 0 (IDIFFERENCE MAXINTENSITY I)
                                                              255))
                                             (CONS '(0 0 255))
                                             (for I from MINVISIBLERED to MAXINTENSITY by REDSTEPSIZE
                                                collect                 (* red up)
                                                        (LIST I 0 255))
                                             (for I from MINVISIBLEGREEN to MAXINTENSITY by GREENSTEPSIZE
                                                collect                 (* GREEN UP)
                                                        (LIST 255 I 255))
                                             (for I from GREENSTEPSIZE to (IDIFFERENCE MAXINTENSITY
                                                                                       MINVISIBLEGREEN)
                                                by GREENSTEPSIZE collect
                                                                        (* all down)
                                                        (LIST (IDIFFERENCE MAXINTENSITY I)
                                                              (IDIFFERENCE MAXINTENSITY I)
                                                              (IDIFFERENCE MAXINTENSITY I)))
                                             (CONS '(0 0 0]
                                    (T RAINBOWINTENSITIES))
                     NBITS])
```

(**RGBTOHLS**

```
  [LAMBDA (RGB GREEN BLUE)                                  (* kbr%: " 3-Jun-86 20:13")
```

(* Converts from a red green blue triple of color information into a hue lightness saturation triple.
*)

(* This algorithm was taken from Procedural Elements for Computer Graphics 1985 page 405 by David F. Rogers *)

```
    (PROG (RED CR CG CB M1 M2 LIGHTNESS HLS)
          (COND
             ((LISTP RGB)
              (SETQ RED (fetch (RGB RED) of RGB))
              (SETQ GREEN (fetch (RGB GREEN) of RGB))
              (SETQ BLUE (fetch (RGB BLUE) of RGB)))
             (T (SETQ RED RGB)))
          (SETQ M1 (MAX RED GREEN BLUE))
          (SETQ M2 (MIN RED GREEN BLUE))
          (SETQ LIGHTNESS (FQUOTIENT (FPLUS (FQUOTIENT M1 255)
                                            (FQUOTIENT M2 255))
```

```
                                2))
          [SETQ HLS (COND
                       ((EQ M1 M2)
                        (create HLS
                               HUE _ 0
                               LIGHTNESS _ LIGHTNESS
                               SATURATION _ 0.0))
                       (T (SETQ CR (FQUOTIENT (IDIFFERENCE M1 RED)
                                              (IDIFFERENCE M1 M2)))
                          (SETQ CG (FQUOTIENT (IDIFFERENCE M1 GREEN)
                                              (IDIFFERENCE M1 M2)))
                          (SETQ CB (FQUOTIENT (IDIFFERENCE M1 BLUE)
                                              (IDIFFERENCE M1 M2)))
                          (create HLS
                                 HUE _ (IMOD (FIX (FTIMES [COND
                                                             ((EQ M1 RED)
                                                              (FDIFFERENCE CB CG))
                                                             ((EQ M1 GREEN)
                                                              (FPLUS 2.0 (FDIFFERENCE CR CB)))
                                                             (T (FPLUS 4.0 (FDIFFERENCE CG CR]
                                                          60.0))
                                             360)
                                 LIGHTNESS _ LIGHTNESS
                                 SATURATION _ (COND
                                                  ((FGREATERP 0.5 LIGHTNESS)
                                                   (FQUOTIENT (IDIFFERENCE M1 M2)
                                                              (IPLUS M1 M2)))
                                                  (T (FQUOTIENT (IDIFFERENCE M1 M2)
                                                                (IDIFFERENCE (ITIMES 2 255)
                                                                             (IPLUS M1 M2]
          (RETURN HLS])

)

(DEFINEQ

(OVERPAINT
  [LAMBDA (BM1 BM2 X Y TXT SCR)                                    (* kbr%: " 2-Sep-85 20:30")

          (* Uses BM1 as a mask thru which it paints the INVERSE of texture onto BM2 at position X Y)

    (PROG (BMW BMH)
          (SETQ BMW (BITMAPWIDTH BM1))
          (SETQ BMH (BITMAPHEIGHT BM1))
          (OR SCR (SETQ SCR (BITMAPCOPY BM1)))                     (* We need a scratch BM. Most demos cache one)
          (BITBLT BM1 0 0 SCR 0 0 BMW BMH 'INPUT 'REPLACE)
          (BITBLT NIL NIL NIL SCR 0 0 BMW BMH 'TEXTURE 'ERASE TXT)
          (BITBLT BM1 0 0 BM2 X Y BMW BMH 'INPUT 'ERASE)
          (BITBLT SCR 0 0 BM2 X Y BMW BMH 'INPUT 'PAINT])

(BITMAPFROMSTRING
  [LAMBDA (STRING FONT BITSPERPIXEL)                               (* kbr%: "11-Aug-85 16:14")
    (PROG (BITMAP DS)
          (SETQ BITMAP (BITMAPCREATE (STRINGWIDTH STRING FONT)
                                     (FONTPROP FONT 'HEIGHT)
                                     BITSPERPIXEL))
          (SETQ DS (DSPCREATE BITMAP))
          (DSPFONT FONT DS)
          (MOVETO 0 (FONTPROP FONT 'DESCENT)
                  DS)
          (PRIN3 STRING DS)
          (RETURN BITMAP])

(SHADEBITMAP
  [LAMBDA (BM T0 T1)                                               (* bas%: "25-APR-82 15:02")
                                                                   (* Shades bitmap BM with T0 into 0 areas and T1 into 1 areas)
    (BITBLT NIL NIL NIL BM NIL NIL NIL NIL 'TEXTURE 'INVERT (LOGAND T0 (LOGXOR T0 T1)))
    (BITBLT NIL NIL NIL BM NIL NIL NIL NIL 'TEXTURE 'PAINT (LOGAND T0 T1))
    (BITBLT NIL NIL NIL BM NIL NIL NIL NIL 'TEXTURE 'ERASE (LOGXOR (LOGOR T0 T1)
                                                                   65535])

)

(RPAQ? EDITCOLORMAP.WINDOW NIL)

(DEFINEQ

(EDITCOLORMAP
  [LAMBDA NIL                                                      (* kbr%: " 5-Jun-86 22:49")
                                                                   (* Colormap Editor. Let's user interactively adjust colormap.
                                                                   *)
    (PROG (XPOS REDREGION GREENREGION BLUEREGION HUEREGION LIGHTNESSREGION SATURATIONREGION BOTTOM)
          (COND
             ((NULL EDITCOLORMAP.WINDOW)
              (SETQ EDITCOLORMAP.WINDOW (CREATEW (GETBOXREGION EditColorMapWidth EditColorMapHeight NIL NIL NIL
```

```
                                                    "Select location of Colormap Editor window.")
                                                "Colormap Editor"))
            (CLRPROMPT)
            (WINDOWPROP EDITCOLORMAP.WINDOW 'BUTTONEVENTFN 'EDITCOLORMAP.BUTTONEVENTFN)
            (WINDOWPROP EDITCOLORMAP.WINDOW 'REPAINTFN 'EDITCOLORMAP.REDISPLAYFN)
            (WINDOWPROP EDITCOLORMAP.WINDOW 'COLOR 0))
           (T (CLEARW EDITCOLORMAP.WINDOW)))
         (REDISPLAYW EDITCOLORMAP.WINDOW])
```

(EDITCOLORMAP.BUTTONEVENTFN

```
[LAMBDA (WINDOW)                                                    (* kbr%: " 4-Jun-86 21:21")

         (* Colormap editor. Displays a colormap in a window and allows the user to change it.
         *)

   (PROG (REDREGION GREENREGION BLUEREGION HUEREGION LIGHTNESSREGION SATURATIONREGION BOTTOM COLOR COLORMAP
                LEVEL LASTX LASTY HLS OLDLEVEL COMPONENT)
         (PROGN (SETQ REDREGION (WINDOWPROP WINDOW 'REDREGION))
                (SETQ GREENREGION (WINDOWPROP WINDOW 'GREENREGION))
                (SETQ BLUEREGION (WINDOWPROP WINDOW 'BLUEREGION))
                (SETQ HUEREGION (WINDOWPROP WINDOW 'HUEREGION))
                (SETQ LIGHTNESSREGION (WINDOWPROP WINDOW 'LIGHTNESSREGION))
                (SETQ SATURATIONREGION (WINDOWPROP WINDOW 'SATURATIONREGION))
                (SETQ BOTTOM (fetch (REGION BOTTOM) of REDREGION)))
         (SETQ COLOR (WINDOWPROP WINDOW 'COLOR))
         (SETQ COLORMAP (SCREENCOLORMAP))
         (COND
            [(LASTMOUSESTATE MIDDLE)
             (COND
                ((NUMBERP (SETQ LEVEL (GETCOLOR#FROMUSER)))
                 (WINDOWPROP WINDOW 'COLOR LEVEL)
                 (REDISPLAYW WINDOW]
            ((LASTMOUSESTATE LEFT)
             (SETQ LASTX (LASTMOUSEX WINDOW))
             (SETQ LASTY (LASTMOUSEY WINDOW))
             (COND
                ([SETQ COMPONENT (COND
                                    ((INSIDEP REDREGION LASTX LASTY)
                                     'RED)
                                    ((INSIDEP GREENREGION LASTX LASTY)
                                     'GREEN)
                                    ((INSIDEP BLUEREGION LASTX LASTY)
                                     'BLUE)
                                    ((INSIDEP HUEREGION LASTX LASTY)
                                     'HUE)
                                    ((INSIDEP LIGHTNESSREGION LASTX LASTY)
                                     'LIGHTNESS)
                                    ((INSIDEP SATURATIONREGION LASTX LASTY)
                                     'SATURATION]
                 (SETQ OLDLEVEL (WINDOWPROP WINDOW COMPONENT))
                 (until (MOUSESTATE (NOT LEFT)) do          (* As long as LEFT is down, adjust the color.
                                                            *)
                                                 [SETQ LEVEL (IMIN 255 (IMAX 0 (IDIFFERENCE (LASTMOUSEY WINDOW
                                                                                                )
                                                                                            BOTTOM]
                                                 (COND
                                                    ((NOT (EQ LEVEL OLDLEVEL))
                                                     (CHANGECOLORLEVELS WINDOW COMPONENT LEVEL)
                                                     [SCREENCOLORMAPENTRY COLOR (create RGB
                                                                                        RED _
                                                                                        (WINDOWPROP WINDOW
                                                                                               'RED)
                                                                                        GREEN _
                                                                                        (WINDOWPROP WINDOW
                                                                                               'GREEN)
                                                                                        BLUE _
                                                                                        (WINDOWPROP WINDOW
                                                                                               'BLUE]
                                                 (SETQ OLDLEVEL LEVEL])
```

(EDITCOLORMAP.REDISPLAYFN

```
[LAMBDA (WINDOW)                                                    (* kbr%: " 4-Jun-86 20:46")
                                                                    (* Colormap Editor. Let's user interactively adjust colormap.
                                                                    *)
   (PROG (XPOS REDREGION GREENREGION BLUEREGION HUEREGION LIGHTNESSREGION SATURATIONREGION BOTTOM COLORMAP
               COLOR)
         (CLEARW WINDOW)
         (PROGN (MOVETO 35 4 WINDOW)
                (PRIN1 "RED" WINDOW)
                (SETQ REDREGION '(40 16 10 256))
                (OUTLINEREGION REDREGION 2 NIL WINDOW)
                (WINDOWPROP WINDOW 'REDREGION REDREGION))
         (PROGN (MOVETO 70 4 WINDOW)
                (PRIN1 "GREEN" WINDOW)
                (SETQ GREENREGION '(82 16 10 256))
```

```
                (OUTLINEREGION GREENREGION 2 NIL WINDOW)
                (WINDOWPROP WINDOW 'GREENREGION GREENREGION))
          (PROGN (MOVETO 119 4 WINDOW)
                 (PRIN1 "BLUE" WINDOW)
                 (SETQ BLUEREGION '(128 16 10 256))
                 (OUTLINEREGION BLUEREGION 2 NIL WINDOW)
                 (WINDOWPROP WINDOW 'BLUEREGION BLUEREGION))
          (PROGN (MOVETO 181 4 WINDOW)
                 (PRIN1 "HUE" WINDOW)
                 (SETQ HUEREGION '(186 16 10 256))
                 (OUTLINEREGION HUEREGION 2 NIL WINDOW)
                 (WINDOWPROP WINDOW 'HUEREGION HUEREGION))
          (PROGN (MOVETO 216 4 WINDOW)
                 (PRIN1 "LIGHTNESS" WINDOW)
                 (SETQ LIGHTNESSREGION '(242 16 10 256))
                 (OUTLINEREGION LIGHTNESSREGION 2 NIL WINDOW)
                 (WINDOWPROP WINDOW 'LIGHTNESSREGION LIGHTNESSREGION))
          (PROGN (MOVETO 300 4 WINDOW)
                 (PRIN1 "SAT" WINDOW)
                 (SETQ SATURATIONREGION '(305 16 10 256))
                 (OUTLINEREGION SATURATIONREGION 2 NIL WINDOW)
                 (WINDOWPROP WINDOW 'SATURATIONREGION SATURATIONREGION))
          (PROGN (SETQ COLORMAP (SCREENCOLORMAP))
                 (SETQ COLOR (WINDOWPROP WINDOW 'COLOR))
                 (MOVETO 8 250 WINDOW)
                 (printout WINDOW .I3 COLOR)
                 (DISPLAYCOLORLEVELS WINDOW (ELT COLORMAP COLOR])
```

(EDITCOLORMAP.VALUELEVEL

```
[LAMBDA (COMPONENT WINDOWLEVEL)                                    (* kbr%: " 3-Jun-86 19:55")

          (* * Value that would be stored in an RGB or HLS corresponding to WINDOWLEVEL.
          *)

   (SELECTQ COMPONENT
        (HUE (IQUOTIENT (ITIMES WINDOWLEVEL 360)
                        255))
        ((LIGHTNESS SATURATION)
            (FQUOTIENT WINDOWLEVEL 255))
        ((RED GREEN BLUE)
            WINDOWLEVEL)
        (SHOULDNT])
```

(EDITCOLORMAP.WINDOWLEVEL

```
[LAMBDA (COMPONENT VALUELEVEL)                                     (* kbr%: " 3-Jun-86 19:55")

          (* * Given VALUELEVEL of an RGB or HLS, what WINDOWLEVEL should be used to display it? *)

   (SELECTQ COMPONENT
        (HUE (IQUOTIENT (ITIMES VALUELEVEL 255)
                        360))
        ((LIGHTNESS SATURATION)
            (FIX (FTIMES VALUELEVEL 255)))
        ((RED GREEN BLUE)
            VALUELEVEL)
        (SHOULDNT])
```

(CHANGECOLORLEVELS

```
[LAMBDA (WINDOW COMPONENT WINDOWLEVEL)                             (* kbr%: " 3-Jun-86 19:55")
   (PROG (RGB HLS)
         (DISPLAYCOLORLEVEL WINDOW COMPONENT (EDITCOLORMAP.VALUELEVEL COMPONENT WINDOWLEVEL)
                WINDOWLEVEL)
         (SELECTQ COMPONENT
              ((RED GREEN BLUE)
                  [SETQ HLS (RGBTOHLS (WINDOWPROP WINDOW 'RED)
                                      (WINDOWPROP WINDOW 'GREEN)
                                      (WINDOWPROP WINDOW 'BLUE]
                  (DISPLAYCOLORLEVEL WINDOW 'HUE (fetch (HLS HUE) of HLS)
                         (EDITCOLORMAP.WINDOWLEVEL 'HUE (fetch (HLS HUE) of HLS)))
                  (DISPLAYCOLORLEVEL WINDOW 'LIGHTNESS (fetch (HLS LIGHTNESS) of HLS)
                         (EDITCOLORMAP.WINDOWLEVEL 'LIGHTNESS (fetch (HLS LIGHTNESS) of HLS)))
                  (DISPLAYCOLORLEVEL WINDOW 'SATURATION (fetch (HLS SATURATION) of HLS)
                         (EDITCOLORMAP.WINDOWLEVEL 'SATURATION (fetch (HLS SATURATION) of HLS))))
              ((HUE LIGHTNESS SATURATION)
                  [SETQ RGB (HLSTORGB (EDITCOLORMAP.VALUELEVEL 'HUE (WINDOWPROP WINDOW 'HUE))
                                      (EDITCOLORMAP.VALUELEVEL 'LIGHTNESS (WINDOWPROP WINDOW 'LIGHTNESS))
                                      (EDITCOLORMAP.VALUELEVEL 'SATURATION (WINDOWPROP WINDOW 'SATURATION]
                  (DISPLAYCOLORLEVEL WINDOW 'RED (fetch (RGB RED) of RGB)
                         (fetch (RGB RED) of RGB))
                  (DISPLAYCOLORLEVEL WINDOW 'GREEN (fetch (RGB GREEN) of RGB)
                         (fetch (RGB GREEN) of RGB))
                  (DISPLAYCOLORLEVEL WINDOW 'BLUE (fetch (RGB BLUE) of RGB)
                         (fetch (RGB BLUE) of RGB)))
              (SHOULDNT])
```

(GETCOLOR#FROMUSER

```
[LAMBDA NIL                                                    (* edited%: " 8-SEP-82 21:44")
                                                               (* reads a color number from the user.)

  (PROG (RESPONSE)
        (MOVEW [COND
                  ((TYPENAMEP EDIT8BITCOLORMAPNUMBERREADER 'WINDOW)
                   EDIT8BITCOLORMAPNUMBERREADER)
                  (T (SETQ EDIT8BITCOLORMAPNUMBERREADER (CREATE.NUMBERPAD.READER '(Enter color number to
                                                                                      edit%:)
                                                                (create POSITION
                                                                        XCOORD _ LASTMOUSEX
                                                                        YCOORD _ LASTMOUSEY]
               (create POSITION
                       XCOORD _ LASTMOUSEX
                       YCOORD _ LASTMOUSEY))
    LP  (COND
           ([NULL (ERSETQ (SETQ RESPONSE (NUMBERPAD.READ EDIT8BITCOLORMAPNUMBERREADER]

        (* currently there is no way NIL can be returned from NUMBERPAD.READ but there should be a way to quit.)

            (RETURN NIL))
           ((OR (ILESSP RESPONSE 0)
                (IGREATERP RESPONSE 255))
            (PROMPTPRINT "Color numbers must be between 0 and 255.")
            (GO LP))
           (T (RETURN RESPONSE])
```

(GETCOLOR#FROMSCREEN

```
[LAMBDA NIL                                                    (* rrb " 3-NOV-82 13:57")
                                                               (* returns the color number of a point selected by the user.)

  (RESETFORM (CHANGECURSORSCREEN (COLORSCREENBITMAP))
         (PROG (POS)
               (SETQ POS (GETPOSITION))
               (RETURN (AND POS (BITMAPBIT (COLORSCREENBITMAP)
                                           (fetch (POSITION XCOORD) of POS)
                                           (fetch (POSITION YCOORD) of POS])
```

(DISPLAYCOLORLEVEL

```
[LAMBDA (WINDOW COMPONENT NEWLEVEL WINDOWLEVEL)                (* kbr%: " 4-Jun-86 20:23")
  (PROG (REGION)
        (WINDOWPROP WINDOW COMPONENT WINDOWLEVEL)
        (SETQ REGION (SELECTQ COMPONENT
                              (RED (WINDOWPROP WINDOW 'REDREGION))
                              (BLUE (WINDOWPROP WINDOW 'BLUEREGION))
                              (GREEN (WINDOWPROP WINDOW 'GREENREGION))
                              (HUE (WINDOWPROP WINDOW 'HUEREGION))
                              (LIGHTNESS (WINDOWPROP WINDOW 'LIGHTNESSREGION))
                              (SATURATION (WINDOWPROP WINDOW 'SATURATIONREGION))
                              (SHOULDNT)))
        [PROGN                                                 (* Print out new level of COMPONENT.
                                                               *)
               (MOVETO (IDIFFERENCE (fetch (REGION LEFT) of REGION)
                                    12)
                       (IPLUS 8 (fetch (REGION TOP) of REGION))
                       WINDOW)                                 (* Overstrike extra digits in case the old value was larger.
                                                               *)
               (COND
                  ((FIXP NEWLEVEL)
                   (printout WINDOW " " .I3 NEWLEVEL))
                  (T (printout WINDOW .F5.3 NEWLEVEL]
        (FILLINREGION REGION WINDOWLEVEL GRAYSHADE WINDOW])
```

(FILLINREGION

```
[LAMBDA (REGION HEIGHT GRAY WINDOW)                            (* rrb "23-FEB-82 12:26")
                                                               (* fills part of a region with gray.)

  (DSPFILL REGION WHITESHADE 'REPLACE WINDOW)
  (AREAFILL (fetch (REGION LEFT) of REGION)
            (fetch (REGION BOTTOM) of REGION)
            (fetch (REGION WIDTH) of REGION)
            HEIGHT GRAY 'REPLACE WINDOW])
```

(AREAFILL

```
[LAMBDA (LFT BTM WDTH HGTH SHADE OPERATION WINDOW)             (* fills an area of a window with shade.)
  (BITBLT NIL NIL NIL WINDOW LFT BTM WDTH HGTH 'TEXTURE OPERATION SHADE])
```

(CENTEREDLEFT

```
[LAMBDA (WIDTH LEFT RIGHT)                                     (* rrb "16-FEB-82 14:58")
                                                               (* returns the left point that would leave WIDTH centered
                                                               between LEFT and RIGHT)

  (IQUOTIENT (IDIFFERENCE (IPLUS LEFT RIGHT)
```

```
                    WIDTH)
          2])
```

(OUTLINEAREA

```
  [LAMBDA (LFT BTM WDTH HGHT LINEWIDTH OPERATION WIN)          (* rrb "17-FEB-82 10:59")
                                                               (* outlines an area of a window.)
    (PROG (LEFTPLUSWIDTH RIGHTLINELEFT VERTLINETOP TOPY LINEWIDTH)
          (SETQ LINEWIDTH (OR (NUMBERP LINEWIDTH)
                              1))
          (SETQ LFT (IDIFFERENCE LFT LINEWIDTH))
          (SETQ BTM (IDIFFERENCE BTM LINEWIDTH))
          (SETQ WDTH (IPLUS WDTH (ITIMES LINEWIDTH 2)))
          (SETQ HGHT (IPLUS HGHT (ITIMES LINEWIDTH 2)))
          (DRAWLINE LFT BTM LFT (SETQ VERTLINETOP (SUB1 (IPLUS BTM HGHT)))
                 LINEWIDTH OPERATION WIN)
          (DRAWLINE (SETQ RIGHTLINELEFT (IDIFFERENCE (IPLUS LFT WDTH)
                                                     LINEWIDTH))
                 BTM RIGHTLINELEFT VERTLINETOP LINEWIDTH OPERATION WIN)
          (DRAWLINE (SETQ LEFTPLUSWIDTH (IPLUS LFT LINEWIDTH))
                 BTM
                 (SETQ RIGHTLINELEFT (SUB1 RIGHTLINELEFT))
                 BTM LINEWIDTH OPERATION WIN)
          (DRAWLINE LEFTPLUSWIDTH (SETQ TOPY (ADD1 (IDIFFERENCE VERTLINETOP LINEWIDTH)))
                 RIGHTLINELEFT TOPY LINEWIDTH OPERATION WIN])
```

(OUTLINEREGION

```
  [LAMBDA (REGION OUTLINEWIDTH OPERATION WIN)                  (* rrb "17-FEB-82 10:58")
                                                               (* outlines the region REGION with a width wide line)
    (OUTLINEAREA (fetch (REGION LEFT) of REGION)
           (fetch (REGION BOTTOM) of REGION)
           (fetch (REGION WIDTH) of REGION)
           (fetch (REGION HEIGHT) of REGION)
           OUTLINEWIDTH OPERATION WIN])

)

(DEFINEQ
```

(ADJUSTCOLORMAP

```
  [LAMBDA (PRIMARY DELTA)                                      (* kbr%: " 5-Jun-86 19:41")
                                                               (* Adds DELTA points of intensity to all values of PRIMARY
                                                               color in SCREENCOLORMAP *)
    (PROG NIL
          (for COLOR from 0 to (MAXIMUMCOLOR (BITSPERPIXEL (SCREENCOLORMAP)))
             do (COLORLEVEL COLOR PRIMARY (IMIN 255 (IMAX 0 (IPLUS (COLORLEVEL COLOR PRIMARY)
                                                                   DELTA])
```

(SHOWCOLORBLOCKS

```
  [LAMBDA (DESTINATION)                                        (* kbr%: "17-Aug-85 21:44")
                                                               (* Puts shade blocks onto DESTINATION.
                                                               *)
    (PROG (BITSPERPIXEL MAXSHADE N WIDTH HEIGHT SHADE)
          (SETQ BITSPERPIXEL (BITSPERPIXEL DESTINATION))
          (SETQ MAXSHADE (MAXIMUMSHADE BITSPERPIXEL))
          [SETQ N (FIXR (SQRT (ADD1 MAXSHADE]
          (SETQ WIDTH (IQUOTIENT (IPLUS (BITMAPWIDTH DESTINATION)
                                        N -1)
                                 N))
          (SETQ HEIGHT (IQUOTIENT (IPLUS (BITMAPHEIGHT DESTINATION)
                                         N -1)
                                  N))
          (SETQ SHADE 0)
          (for Y from (SUB1 N) to 0 by -1 do (for X from 0 to (SUB1 N) do (BLTSHADE SHADE DESTINATION
                                                                                 (ITIMES X WIDTH)
                                                                                 (ITIMES Y HEIGHT)
                                                                                 WIDTH HEIGHT 'REPLACE)
                                                                          (SETQ SHADE (ADD1 SHADE))
                                                                          (COND
                                                                             ((IGREATERP SHADE MAXSHADE)
                                                                              (SETQ SHADE 0])
```

(MAPOFACOLOR

```
  [LAMBDA (RGB BITSPERPIXEL)                                   ; Edited  9-Nov-90 20:45 by TS
                                                               (* creates a gray color map *)
    (DECLARE (GLOBALVARS \COLORSCREEN))
    (PROG (MAXCOLOR RED GREEN BLUE OPRED OPGREEN OPBLUE COLORMAP)
          [SETQ MAXCOLOR (MAXIMUMCOLOR (OR BITSPERPIXEL (SETQ BITSPERPIXEL (FETCH (SCREEN SCBITSPERPIXEL)
                                                                                   OF \COLORSCREEN]
          (SETQ RED (fetch (RGB RED) of RGB))
          (SETQ GREEN (fetch (RGB GREEN) of RGB))
          (SETQ BLUE (fetch (RGB BLUE) of RGB))
          (SETQ OPRED (IDIFFERENCE MAXCOLOR RED))
          (SETQ OPGREEN (IDIFFERENCE MAXCOLOR GREEN))
```

```
          (SETQ OPBLUE (IDIFFERENCE MAXCOLOR BLUE))
          (SETQ COLORMAP (COLORMAPCREATE (for I from 0 to MAXCOLOR as OPI from MAXCOLOR to 0 by -1
                                            collect (create RGB
                                                            RED _ (IQUOTIENT (IPLUS (ITIMES OPI OPRED)
                                                                                    (ITIMES I RED))
                                                                             MAXCOLOR)
                                                            GREEN _ (IQUOTIENT (IPLUS (ITIMES OPI OPGREEN)
                                                                                      (ITIMES I GREEN))
                                                                               MAXCOLOR)
                                                            BLUE _ (IQUOTIENT (IPLUS (ITIMES OPI OPBLUE)
                                                                                     (ITIMES I BLUE))
                                                                              MAXCOLOR)))
                                  BITSPERPIXEL))
          (RETURN COLORMAP])
```

(COLORHEXPATTERN

```
  [LAMBDA (LIGHTNESS)                                                    (* kbr%: " 3-Jun-86 22:36")
                                                                         (* Put a color hex pattern on the color display.
                                                                         *)
    (PROG (DESTINATION WIDTH HEIGHT BITSPERPIXEL N HEXWIDTH HEXHEIGHT LEFT BOTTOM COLOR MAXI JDIST IDIST)
          (COND
             ((NULL LIGHTNESS)
              (SETQ LIGHTNESS 0.5)))
          (SETQ DESTINATION (COLORSCREENBITMAP))
          (SETQ WIDTH (BITMAPWIDTH DESTINATION))
          (SETQ HEIGHT (BITMAPHEIGHT DESTINATION))
          (SETQ BITSPERPIXEL (BITSPERPIXEL DESTINATION))
          (SETQ N (SELECTQ BITSPERPIXEL
                       (4 1)
                       (8 8)
                       (RETURN)))
          (SETQ HEXWIDTH (IQUOTIENT WIDTH (IPLUS (ITIMES 2 N)
                                                 1)))
          (SETQ HEXHEIGHT (IQUOTIENT HEIGHT (IPLUS (ITIMES 2 N)
                                                   1)))
          (BLTSHADE MINIMUMSHADE DESTINATION)
          (SETQ COLOR 0)
          [for J from N to 0 by -1 do (SETQ BOTTOM (ITIMES (IPLUS J N)
                                                           HEXHEIGHT))
                                      (SETQ MAXI (IDIFFERENCE (IPLUS (ITIMES 2 N)
                                                                     1)
                                                              J))
                                      (for I from 0 to MAXI
                                         do (SETQ LEFT (IQUOTIENT (ITIMES (IPLUS (ITIMES 2 I)
                                                                                 J)
                                                                          HEXWIDTH)
                                                                  2))
                                            (SETQ COLOR (ADD1 COLOR))
                                            (BLTSHADE COLOR DESTINATION LEFT BOTTOM HEXWIDTH HEXHEIGHT)
                                            (SETQ JDIST (FQUOTIENT J N))
                                            (SETQ IDIST (FDIFFERENCE (FTIMES 2.0 (FQUOTIENT I MAXI))
                                                                     1.0))
                                            (SCREENCOLORMAPENTRY COLOR
                                                   (HLSTORGB (ATAN JDIST IDIST)
                                                          LIGHTNESS
                                                          (SQRT (FQUOTIENT (FPLUS (FTIMES IDIST IDIST)
                                                                                  (FTIMES JDIST JDIST))
                                                                           2.0]
          (for J from -1 to (IMINUS N) by -1
             do (SETQ BOTTOM (ITIMES (IPLUS J N)
                                     HEXHEIGHT))
                (SETQ MAXI (IPLUS (IPLUS (ITIMES 2 N)
                                         1)
                                  J))
                (for I from 0 to MAXI do (SETQ LEFT (IQUOTIENT (ITIMES (IPLUS (ITIMES 2 I)
                                                                              (IMINUS J))
                                                                       HEXWIDTH)
                                                               2))
                                         (SETQ COLOR (ADD1 COLOR))
                                         (BLTSHADE COLOR DESTINATION LEFT BOTTOM HEXWIDTH HEXHEIGHT)
                                         (SETQ JDIST (FQUOTIENT J N))
                                         (SETQ IDIST (FDIFFERENCE (FTIMES 2.0 (FQUOTIENT I MAXI))
                                                                  1.0))
                                         (SCREENCOLORMAPENTRY COLOR (HLSTORGB (ATAN JDIST IDIST)
                                                                              LIGHTNESS
                                                                              (SQRT (FQUOTIENT (FPLUS (FTIMES IDIST
                                                                                                              IDIST)
                                                                                                      (FTIMES JDIST
                                                                                                              JDIST))
                                                                                               2.0])
)

(RPAQQ EditColorMapHeight 315)

(RPAQQ EditColorMapWidth 380)
```

```
(RPAQQ COLOR#MENUSAVE NIL)

(RPAQQ CONTROLMENUSAVE NIL)

(RPAQQ EDIT8BITCOLORMAPMENU NIL)

(RPAQQ EDIT8BITCOLORMAPNUMBERREADER NIL)

(DECLARE%: DOEVAL@COMPILE DONTCOPY

(GLOBALVARS COLOR#MENUSAVE CONTROLMENUSAVE EDIT8BITCOLORMAPMENU EDIT8BITCOLORMAPNUMBERREADER EditColorMapHeight
       EditColorMapWidth)
)
```

;;; support for global naming and querying of colors.

```
(DEFINEQ
```

(CNSMENUINIT

```
  [LAMBDA NIL                                                    (* gbn " 9-Aug-85 03:11")
    [SETQ CNSHUEMENU (create MENU
                             ITEMS _ (for I in DICOLOR.hueMapping collect (CAR I]
    [SETQ CNSSATURATIONMENU (create MENU
                                    ITEMS _ (for I in DICOLOR.saturationMapping collect (CAR I]
    (SETQ CNSLIGHTNESSMENU (create MENU
                                   ITEMS _ (for I in DICOLOR.lightnessMapping collect (CAR I])
```

(CNSTOCSL

```
  [LAMBDA (hue saturation lightness)                             (* hdj "12-Apr-85 19:01")
    (PROG ((hueAtom (MKATOM hue))
           (saturationAtom (MKATOM saturation))
           (lightnessAtom (MKATOM lightness))
           c s l)
          (if [NOT (SETQ c (fetch (hueRecord ordering) of (ASSOC hueAtom DICOLOR.hueMapping]
              then (SETQ c DICOLOR.achromatic))
          (if (EQ c DICOLOR.achromatic)
              then (SETQ s DICOLOR.noSaturation)
            else (if [NOT (SETQ s (fetch (saturationRecord ordering) of (ASSOC saturationAtom
                                                                               DICOLOR.saturationMapping]
                     then (SETQ s DICOLOR.vivid)))
          (SELECTQ hueAtom
               (Black (SETQ l DICOLOR.black))
               (White (SETQ l DICOLOR.white))
               (if [NOT (SETQ l (fetch (lightnessRecord ordering) of (ASSOC lightnessAtom DICOLOR.lightnessMapping]
                   then (SETQ l DICOLOR.medium)))
          (RETURN (LIST c s l])
```

(CNSTORGB

```
  [LAMBDA (saturation lightness hue)                             (* hdj "15-Jul-85 12:33")
    (LET ((CSL (CNSTOCSL hue saturation lightness)))
         (HLSTORGB (APPLY (FUNCTION CSLTOHLS)
                          CSL])
```

(CSLTOCNS

```
  [LAMBDA (c s l)                                                (* hdj "15-Jul-85 12:37")
    (PROG (hue saturation lightness)
          [if (EQ c DICOLOR.achromatic)
              then (SETQ saturation "")
                   [SELECTC l
                        (DICOLOR.black
                             (SETQ hue "Black")
                             (SETQ lightness ""))
                        (DICOLOR.white
                             (SETQ hue "White")
                             (SETQ lightness ""))
                        (PROGN (SETQ hue "Gray")
                               (SETQ lightness (MKSTRING (fetch (lightnessRecord name) of (DICOLOR.lightnessN l]
            else (SETQ hue (fetch (hueRecord name) of (DICOLOR.hueN c)))
                 (SETQ saturation (fetch (saturationRecord name) of (DICOLOR.saturationN s)))
                 (SETQ lightness (fetch (lightnessRecord name) of (DICOLOR.lightnessN l]
          (RETURN (LIST saturation lightness hue])
```

(DICOLOR.FROM.USER

```
  [LAMBDA NIL                                                    (* gbn "30-Oct-85 11:28")

          (* * Returns a color, either by its name (which can then be looked up on colornames) or as an RGB triple if it is not named.
          Prompts the user first with the global color name menu. She can then choose NEWCOLOR which can be specified as RGB
          or CNS)

    (PROG (NAME RGB)                                             (* first try to get a color name)
          [SETQ NAME (MENU (OR COLORNAMEMENU (SETQ COLORNAMEMENU (create MENU
```

```
                                                                ITEMS _
                                                                (CONS NEWCOLORITEM
                                                                      (for ENTRY in COLORNAMES
                                                                         collect (CAR ENTRY]
        (if (NOT NAME)
            then                                                (* the user clicked outside the menu)
                 (RETURN))
        (SETQ RGB (SELECTQ NAME
                       (RGB (READCOLOR1 "specify new color"))
                       (CNS (APPLY (FUNCTION CNSTORGB)
                                   (GETCNS)))
                       (RETURN NAME)))
        (if (NOT (SETQ NAME (TTYIN "New color name? ")))
            then                                                (* user decided that she didn't want to name the color)
                 (RETURN RGB))
        (push COLORNAMES (CONS (SETQ NAME (CAR NAME))
                               RGB))
        (SETQ COLORNAMEMENU NIL)                                (* invalidate the menu)
        (RETURN NAME])
```

(GETCNS

```
  [LAMBDA NIL                                                   (* gbn " 9-Aug-85 03:13")
    (LIST (MENU CNSLIGHTNESSMENU)
          (MENU CNSSATURATIONMENU)
          (MENU CNSHUEMENU])
```

(HLSTOCSL

```
  [LAMBDA (hue lightness saturation)                            (* hdj "15-Jul-85 12:14")
    (LET ((ISLHue (FQUOTIENT (MOD (PLUS hue 240)
                                  360)
                             360)))
         (PROG (c s l)
               (for old s from DICOLOR.noSaturation to DICOLOR.vivid
                  do (if (EQ s DICOLOR.vivid)
                         then (RETURN))
                     (if (LEQ saturation (PLUS (DICOLOR.saturationNvalue s)
                                               (QUOTIENT (DIFFERENCE (DICOLOR.saturationNvalue (ADD1 s))
                                                                     (DICOLOR.saturationNvalue s))
                                                         2)))
                         then (RETURN)))
               [if (EQ s DICOLOR.noSaturation)
                   then (SETQ c DICOLOR.achromatic)
                        (for old l from DICOLOR.black to DICOLOR.white
                           do (if (EQ l DICOLOR.white)
                                  then (RETURN))
                              (if (LEQ lightness (PLUS (DICOLOR.lightnessNvalue l)
                                                       (QUOTIENT (DIFFERENCE (DICOLOR.lightnessNvalue (ADD1 l))
                                                                             (DICOLOR.lightnessNvalue l))
                                                                 2)))
                                  then (RETURN)))
                 else (for old c from DICOLOR.red to DICOLOR.purplishRed
                         do                                     (* (HELP c))
                            (if (EQ c DICOLOR.purplishRed)
                                then (if (GREATERP ISLHue (PLUS (DICOLOR.hueNvalue c)
                                                                (QUOTIENT (DIFFERENCE 1 (DICOLOR.hueNvalue c))
                                                                          2)))
                                         then (SETQ c DICOLOR.red))
                                     (RETURN))
                            (if (LEQ ISLHue (PLUS (DICOLOR.hueNvalue c)
                                                  (QUOTIENT (DIFFERENCE (DICOLOR.hueNvalue (ADD1 c))
                                                                        (DICOLOR.hueNvalue c))
                                                            2)))
                                then (RETURN)))
                      (for old l from DICOLOR.veryDark to DICOLOR.veryLight
                         do (if (EQ l DICOLOR.veryLight)
                                then (RETURN))
                            (if (LEQ lightness (PLUS (DICOLOR.lightnessNvalue l)
                                                     (QUOTIENT (DIFFERENCE (DICOLOR.lightnessNvalue (ADD1 l))
                                                                           (DICOLOR.lightnessNvalue l))
                                                               2)))
                                then (RETURN]
               (RETURN (LIST c s l])
```

(CSLTOHLS

```
  [LAMBDA (c s l)                                               (* hdj "15-Jul-85 12:23")
    (PROG (hue saturation lightness)
          (if (EQ c DICOLOR.achromatic)
              then (SETQ hue 0.0)
                   (SETQ saturation 0.0)
                   (SETQ lightness (DICOLOR.lightnessNvalue l))
            else (SETQ hue (DICOLOR.hueNvalue c))
                 (SETQ saturation (DICOLOR.saturationNvalue s))
                 (SETQ lightness (DICOLOR.lightnessNvalue l)))
          (RETURN (LIST (MOD (FPLUS 120 (FTIMES hue 360))
```

```
                            360)
                        lightness saturation])
```

(RGBTOCNS

```
  [LAMBDA (Red Green Blue)                                    (* hdj "15-Jul-85 12:36")
    (APPLY (FUNCTION CSLTOCNS)
           (APPLY (FUNCTION HLSTOCSL)
                  (RGBTOHLS Red Green Blue])

)

(RPAQQ DICOLOR.hueMapping
       ((Achromatic 0.0 -1)
        (Red 0.0 0)
        (OrangishRed 0.01 1)
        (RedOrange 0.02 2)
        (ReddishOrange 0.03 3)
        (Orange 0.04 4)
        (YellowishOrange 0.07 5)
        (OrangeYellow 0.1 6)
        (OrangishYellow 0.13 7)
        (Yellow 0.1673 8)
        (GreenishYellow 0.2073 9)
        (YellowGreen 0.2473 10)
        (YellowishGreen 0.2873 11)
        (Green 0.3333 12)
        (BluishGreen 0.4133 13)
        (GreenBlue 0.4933 14)
        (GreenishBlue 0.5733 15)
        (Blue 0.6666 16)
        (PurplishBlue 0.6816 17)
        (BluePurple 0.6966 18)
        (BluishPurple 0.7116 19)
        (Purple 0.73 20)
        (ReddishPurple 0.8 21)
        (PurpleRed 0.87 22)
        (PurplishRed 0.94 23)
        (BrownishRed 0.01 24)
        (RedBrown 0.02 25)
        (ReddishBrown 0.03 26)
        (Brown 0.04 27)
        (YellowishBrown 0.07 28)
        (BrownYellow 0.1 29)
        (BrownishYellow 0.13 30)))

(RPAQQ DICOLOR.lightnessMapping ((Black 0.0 0)
                                 (VeryDark 0.1666 1)
                                 (Dark 0.3333 2)
                                 (Medium 0.5 3)
                                 (Light 0.6666 4)
                                 (VeryLight 0.8333 5)
                                 (White 1.0 6)))

(RPAQQ DICOLOR.saturationMapping ((NoSaturation 0.0 0)
                                  (Grayish 0.25 1)
                                  (Moderate 0.5 2)
                                  (Strong 0.75 3)
                                  (Vivid 1.0 4)))

(RPAQQ NEWCOLORITEM (New% Color 'CNS "Allows specification of a new color" (SUBITEMS (RGB 'RGB "Specify a new
                                                                                        color using Red,
                                                                                        Green, Blue
                                                                                        sliders")
                                                                                (CNS 'CNS "Specify a new
                                                                                     color using English")))

)

(RPAQ? COLORNAMEMENU )

(DEFINEQ
```

(DICOLOR.hueN

```
  [LAMBDA (N)                                                 (* hdj "17-Apr-85 13:38")
    (DECLARE (GLOBALVARS DICOLOR.hueMapping))
    (for ELT in DICOLOR.hueMapping suchthat (EQ (fetch (hueRecord ordering) of ELT)
                                                N])
```

(DICOLOR.hueNvalue

```
  [LAMBDA (N)                                                 (* hdj "18-Apr-85 09:58")
    (fetch (hueRecord value) of (DICOLOR.hueN N])
```

(DICOLOR.hueNname

```
  [LAMBDA (N)                                                 (* hdj "18-Apr-85 10:07")
    (fetch (hueRecord name) of (DICOLOR.hueN N])
```

```
(DICOLOR.lightnessN
  [LAMBDA (N)                                                    (* hdj "17-Apr-85 13:40")
    (DECLARE (GLOBALVARS DICOLOR.lightnessMapping))
    (for ELT in DICOLOR.lightnessMapping suchthat (EQ (fetch (lightnessRecord ordering) of ELT)
                                                      N])

(DICOLOR.lightnessNvalue
  [LAMBDA (N)                                                    (* hdj "17-Apr-85 13:36")
    (fetch (lightnessRecord value) of (DICOLOR.lightnessN N])

(DICOLOR.lightnessNname
  [LAMBDA (N)                                                    (* hdj "17-Apr-85 14:02")
    (fetch (lightnessRecord name) of (DICOLOR.lightnessN N])

(DICOLOR.saturationN
  [LAMBDA (N)                                                    (* hdj "17-Apr-85 13:39")
    (DECLARE (GLOBALVARS DICOLOR.saturationMapping))
    (for ELT in DICOLOR.saturationMapping suchthat (EQ (fetch (saturationRecord ordering) of ELT)
                                                       N])

(DICOLOR.saturationNvalue
  [LAMBDA (N)                                                    (* hdj "17-Apr-85 13:36")
    (fetch (saturationRecord value) of (DICOLOR.saturationN N])

(DICOLOR.saturationNname
  [LAMBDA (N)                                                    (* hdj "17-Apr-85 14:02")
    (fetch (saturationRecord name) of (DICOLOR.saturationN N])

)

(DECLARE%: EVAL@LOAD DONTCOPY

(DECLARE%: EVAL@COMPILE

(RECORD hueRecord (name value ordering))

(RECORD lightnessRecord (name value ordering))

(RECORD saturationRecord (name value ordering))
)

(RPAQQ DICOLOR.hueConstants
       (DICOLOR.achromatic DICOLOR.blue DICOLOR.bluePurple DICOLOR.bluishGreen DICOLOR.bluishPurple
              DICOLOR.brown DICOLOR.brownYellow DICOLOR.brownishRed DICOLOR.brownishYellow DICOLOR.green
              DICOLOR.greenBlue DICOLOR.greenishBlue DICOLOR.greenishYellow DICOLOR.orange DICOLOR.orangeYellow
              DICOLOR.orangishRed DICOLOR.orangishYellow DICOLOR.purple DICOLOR.purpleRed DICOLOR.purplishBlue
              DICOLOR.purplishRed DICOLOR.red DICOLOR.redBrown DICOLOR.redOrange DICOLOR.reddishBrown
              DICOLOR.reddishOrange DICOLOR.reddishPurple DICOLOR.yellow DICOLOR.yellowGreen
              DICOLOR.yellowishBrown DICOLOR.yellowishGreen DICOLOR.yellowishOrange))

(DECLARE%: EVAL@COMPILE

(RPAQQ DICOLOR.achromatic -1)

(RPAQQ DICOLOR.blue 16)

(RPAQQ DICOLOR.bluePurple 18)

(RPAQQ DICOLOR.bluishGreen 13)

(RPAQQ DICOLOR.bluishPurple 19)

(RPAQQ DICOLOR.brown 27)

(RPAQQ DICOLOR.brownYellow 29)

(RPAQQ DICOLOR.brownishRed 24)

(RPAQQ DICOLOR.brownishYellow 30)

(RPAQQ DICOLOR.green 12)

(RPAQQ DICOLOR.greenBlue 14)

(RPAQQ DICOLOR.greenishBlue 15)

(RPAQQ DICOLOR.greenishYellow 9)

(RPAQQ DICOLOR.orange 4)
```

```
(RPAQQ DICOLOR.orangeYellow 6)

(RPAQQ DICOLOR.orangishRed 1)

(RPAQQ DICOLOR.orangishYellow 7)

(RPAQQ DICOLOR.purple 20)

(RPAQQ DICOLOR.purpleRed 22)

(RPAQQ DICOLOR.purplishBlue 17)

(RPAQQ DICOLOR.purplishRed 23)

(RPAQQ DICOLOR.red 0)

(RPAQQ DICOLOR.redBrown 25)

(RPAQQ DICOLOR.redOrange 2)

(RPAQQ DICOLOR.reddishBrown 26)

(RPAQQ DICOLOR.reddishOrange 3)

(RPAQQ DICOLOR.reddishPurple 21)

(RPAQQ DICOLOR.yellow 8)

(RPAQQ DICOLOR.yellowGreen 10)

(RPAQQ DICOLOR.yellowishBrown 28)

(RPAQQ DICOLOR.yellowishGreen 11)

(RPAQQ DICOLOR.yellowishOrange 5)

(CONSTANTS DICOLOR.achromatic DICOLOR.blue DICOLOR.bluePurple DICOLOR.bluishGreen DICOLOR.bluishPurple
       DICOLOR.brown DICOLOR.brownYellow DICOLOR.brownishRed DICOLOR.brownishYellow DICOLOR.green
       DICOLOR.greenBlue DICOLOR.greenishBlue DICOLOR.greenishYellow DICOLOR.orange DICOLOR.orangeYellow
       DICOLOR.orangishRed DICOLOR.orangishYellow DICOLOR.purple DICOLOR.purpleRed DICOLOR.purplishBlue
       DICOLOR.purplishRed DICOLOR.red DICOLOR.redBrown DICOLOR.redOrange DICOLOR.reddishBrown
       DICOLOR.reddishOrange DICOLOR.reddishPurple DICOLOR.yellow DICOLOR.yellowGreen DICOLOR.yellowishBrown
       DICOLOR.yellowishGreen DICOLOR.yellowishOrange)
)

(RPAQQ DICOLOR.saturationConstants (DICOLOR.noSaturation DICOLOR.grayish DICOLOR.moderate DICOLOR.strong
                                          DICOLOR.vivid))

(DECLARE%: EVAL@COMPILE

(RPAQQ DICOLOR.noSaturation 0)

(RPAQQ DICOLOR.grayish 1)

(RPAQQ DICOLOR.moderate 2)

(RPAQQ DICOLOR.strong 3)

(RPAQQ DICOLOR.vivid 4)

(CONSTANTS DICOLOR.noSaturation DICOLOR.grayish DICOLOR.moderate DICOLOR.strong DICOLOR.vivid)
)

(RPAQQ DICOLOR.lightnessConstants (DICOLOR.black DICOLOR.veryDark DICOLOR.dark DICOLOR.medium DICOLOR.light
                                         DICOLOR.veryLight DICOLOR.white))

(DECLARE%: EVAL@COMPILE

(RPAQQ DICOLOR.black 0)

(RPAQQ DICOLOR.veryDark 1)

(RPAQQ DICOLOR.dark 2)

(RPAQQ DICOLOR.medium 3)

(RPAQQ DICOLOR.light 4)

(RPAQQ DICOLOR.veryLight 5)

(RPAQQ DICOLOR.white 6)

(CONSTANTS DICOLOR.black DICOLOR.veryDark DICOLOR.dark DICOLOR.medium DICOLOR.light DICOLOR.veryLight
       DICOLOR.white)
)
)
```

(**CNSMENUINIT**)

```
(FILESLOAD LLCOLOR READNUMBER)

(SETQ EDITBMMENU NIL)

(MOVD 'ARRAYP 'COLORMAPP)

(PUTPROPS COLOR COPYRIGHT ("Xerox Corporation" 1982 1983 1985 1986 1987 1990))
```

FUNCTION INDEX

CONSTANT INDEX

VARIABLE INDEX

RECORD INDEX