

File created: 16-Jul-99 15:49:37 {DSK}<project>medley3.5>sources>PAINTW.;3

changes to: (FNS PAINTW)

previous date: 17-Jan-94 14:38:09 {DSK}<project>medley3.5>sources>PAINTW.;1

Read Table: XCL

Package: INTERLISP

Format: XCCS

; Copyright (c) 1986, 1990, 1992, 1994, 1999 by Venue & Xerox Corporation. All rights reserved.

(RPAQQ PAINTWCOMS

```
((FNS PAINTW PAINTW.READMODE PAINTW.READBRUSHSHAPE PAINTW.READBRUSHSIZE PAINTW.READBRUSHSHADE
  PAINTW.READBRUSHTEXTURE PAINTW.READ.AND.SAVE.SHADE PAINTW.CACHE.SHADE PAINTW.SHADE.LABEL
  PAINTW.READCOMMAND)
 (COMS (INITVARS (PAINTW.SHADES))
        (GLOBALVARS PAINTW.SHADES))
 (DECLARE\ : DONTEVAL@LOAD DOCOPY (P (PAINTW.CACHE.SHADE BLACKSHADE)
                                       (PAINTW.CACHE.SHADE GRAYSHADE)
                                       (PAINTW.CACHE.SHADE HIGHLIGHTSHADE)))
 (INITVARS (PAINTCOMMANDBRUSH ' (ROUND 2))
            (PAINTCOMMANDMODE ' PAINT)
            (PAINTCOMMANDMENU)
            (PAINTCOMMANDSHADE BLACKSHADE)
            (PAINTSIZEMENU)
            (PAINTSHAPEMENU)
            (PAINTSHADEMENU)
            (PAINTMODEMENU)
            (PAINTWCURSOR)))
```

(DEFINEQ

(PAINTW

(LAMBDA (WINDOW)

; Edited 16-Jul-99 15:49 by rmk:
; Edited 16-Jul-99 15:48 by rmk:
; Edited 17-Jan-94 14:27 by sybalsky:mv:envos

;; allows the user to paint with the cursor

;; should make sure cursor has moved or a button has change before proceeding with the inner loop.

;; has some of the stuff to allow the brush to be an arbitrary bitmap but not all.

```
(SETQ WINDOW (\\INSUREWINDOW WINDOW))
(|printout| PROMPTWINDOW "Left button paints; Middle button erases.
  Right button pops up a command menu.
  To stop, select the QUIT command.")
(RESETLST
 (RESETSAVE NIL (LIST 'CURSOR (CURSOR)))
 (PROG (DS BITSPERPIXEL MASKSHADE BRUSH MASK HOTX HOTY (PREVX -65535)
        (PREVY -65535)
        (PREVBUT -5))
        (TOTOPW WINDOW) ; look for a previously stored brush.
 (COND
  ((SETQ BRUSH (WINDOWPROP WINDOW 'PAINTBRUSH))
   (SETQ PAINTCOMMANDMODE (CAR BRUSH))
   (SETQ PAINTCOMMANDSHADE (CADR BRUSH))
   (SETQ PAINTCOMMANDBRUSH (CADDR BRUSH))))
 (SETQ DS (|fetch| (WINDOW DSP) |of| WINDOW))
 (SETQ BITSPERPIXEL (OR (|fetch| (SCREEN SCDEPTH) |of| (FETCH (WINDOW SCREEN) OF WINDOW))
                       (|fetch| (SCREEN SCBITSPERPIXEL) |of| (FETCH (WINDOW SCREEN) OF WINDOW))))
 (SETQ MASKSHADE (SELECTQ BITSPERPIXEL
                          (1 BLACKSHADE)
                          (MASK.1\S 0 BITSPERPIXEL)))
 BRUSHLP
 (SETQ BRUSH (COND
              ((BITMAP PAINTCOMMANDBRUSH))
              (T (\\GETBRUSH PAINTCOMMANDBRUSH))))
 (SETQ HOTX (HALF (|fetch| (BITMAP BITMAPWIDTH) |of| BRUSH)))
 (SETQ HOTY (HALF (|fetch| (BITMAP BITMAPHEIGHT) |of| BRUSH)))
 (SETQ PAINTWCURSOR
  (|create| CURSOR
           CUIIMAGE _ BRUSH
           CUMASK _ BRUSH
           CUHOTSPOTX _ HOTX
           CUHOTSPOTY _ HOTY
           CUDATA _ NIL |using| PAINTWCURSOR))
 (CURSOR PAINTWCURSOR)
 (COND
  ((NOT (EQ BITSPERPIXEL 1))
   (CURSORCOLOR PAINTCOMMANDSHADE))) ; BRUSH can change if PAINTW is to color screen. *
 (SETQ BRUSH (|fetch| (CURSOR CUIIMAGE) |of| PAINTWCURSOR))
 (SETQ MASK (|fetch| (CURSOR CUMASK) |of| PAINTWCURSOR))
 PAINTLP
 (GETMOUSESTATE)
```

```
(COND
  ((AND (IEQP PREVX LASTMOUSEX)
        (IEQP PREVY LASTMOUSEY)
        (IEQP PREVBUT LASTMOUSEBUTTONS)))
    ;; No movement, and no button changes.
  )
  ((LASTMOUSESTATE RIGHT)
   (SETQ PREVX LASTMOUSEX)
   (SETQ PREVY LASTMOUSEY)
   (SETQ PREVBUT LASTMOUSEBUTTONS))
  (COND
    ((OR (INSIDE? (DSPCLIPPINGREGION NIL DS)
                  (LASTMOUSEX DS)
                  (LASTMOUSEY DS))
         (NOT (WHICHW LASTMOUSEX LASTMOUSEY))) ; inside the interior, give command menu
     (SELECTQ (PAINTW.READCOMMAND)
              (SHADE (SETQ PAINTCOMMANDSHADE (OR (PAINTW.READBRUSHTEXTURE BITSPERPIXEL)
                                                  PAINTCOMMANDSHADE))
                    (GO BRUSHLP))
              (MODE (SETQ PAINTCOMMANDMODE (OR (PAINTW.READMODE)
                                                PAINTCOMMANDMODE))
                    (GO BRUSHLP))
              (SHAPE (RPLACA PAINTCOMMANDBRUSH (OR (PAINTW.READBRUSHSHAPE)
                                                    (CAR PAINTCOMMANDBRUSH)))
                     (GO BRUSHLP))
              (SIZE (RPLACA (CDR PAINTCOMMANDBRUSH)
                            (OR (PAINTW.READBRUSHSIZE)
                                (CADR PAINTCOMMANDBRUSH)))
                    (GO BRUSHLP))
              (QUIT (RETURN)
                    NIL))
     (T ; do the window menu
      (DOWINDOWCOM (WHICHW LASTMOUSEX LASTMOUSEY))))))
  ((AND (LASTMOUSESTATE LEFT)
        (OR (EQ PAINTCOMMANDMODE 'REPLACE)
            (NOT (EQ PAINTCOMMANDSHADE MASKSHADE))))
    ; painting in grey is slightly harder.
    (SETQ PREVX LASTMOUSEX)
    (SETQ PREVY LASTMOUSEY)
    (SETQ PREVBUT LASTMOUSEBUTTONS))
  (COND
    ((EQ PAINTCOMMANDMODE 'REPLACE) ; erase what is there now
     (BITBLT MASK 0 0 DS (IDIFFERENCE (LASTMOUSEX DS)
                                       HOTX)
              (IDIFFERENCE (LASTMOUSEY DS)
                           HOTY)
              NIL NIL 'INPUT 'ERASE) ; put in grey
     (BITBLT BRUSH 0 0 DS (IDIFFERENCE (LASTMOUSEX DS)
                                       HOTX)
              (IDIFFERENCE (LASTMOUSEY DS)
                           HOTY)
              NIL NIL 'MERGE 'PAINT PAINTCOMMANDSHADE))
    (T (BITBLT BRUSH 0 0 DS (IDIFFERENCE (LASTMOUSEX DS)
                                       HOTX)
              (IDIFFERENCE (LASTMOUSEY DS)
                           HOTY)
              NIL NIL 'MERGE PAINTCOMMANDMODE PAINTCOMMANDSHADE))))
  ((LASTMOUSESTATE (OR MIDDLE LEFT))
   (SETQ PREVX LASTMOUSEX)
   (SETQ PREVY LASTMOUSEY)
   (SETQ PREVBUT LASTMOUSEBUTTONS)
   (BITBLT BRUSH 0 0 DS (IDIFFERENCE (LASTMOUSEX DS)
                                       HOTX)
                       (IDIFFERENCE (LASTMOUSEY DS)
                                       HOTY)
                       NIL NIL 'INPUT (COND
                                   ((LASTMOUSESTATE MIDDLE)
                                    'ERASE)
                                   (T PAINTCOMMANDMODE))))
  (T (SETQ PREVX LASTMOUSEX)
     (SETQ PREVY LASTMOUSEY)
     (SETQ PREVBUT LASTMOUSEBUTTONS)))
  (GO PAINTLP))
(WINDOWPROP WINDOW 'PAINTBRUSH (LIST PAINTCOMMANDMODE PAINTCOMMANDSHADE (COPY PAINTCOMMANDBRUSH))))))
```

PAINTW.READMODE

```
(LAMBDA NIL
  (MENU (COND
    ((|type?| MENU PAINTMODEMENU)
     PAINTMODEMENU)
    (T (SETQ PAINTMODEMENU (|create| MENU
                                ITEMS _ ' ((REPLACE 'REPLACE "the screen bits are replaced by the
                                             brush bits")
                                           (INVERT 'INVERT "the screen bits inverted wherever brush
                                             bits are"))
```

(* |rrb| " 1-DEC-82 17:29")

(ADD 'PAINT "the brush bits are added to the bits on the screen")))))))

(PAINTW.READBRUSHSHAPE

(* |rrb| " 1-DEC-82 17:29")

```
(LAMBDA NIL
(MENU (COND
  ((|type?| MENU PAINTSHAPEMENU)
   PAINTSHAPEMENU)
  (T (SETQ PAINTSHAPEMENU (|create| MENU
    ITEMS _ '(DIAGONAL VERTICAL HORIZONTAL SQUARE ROUND)))))))))
```

(PAINTW.READBRUSHSIZE

(* |rrb| " 1-DEC-82 17:30")

```
(LAMBDA NIL
(MENU (COND
  ((|type?| MENU PAINTSIZEMENU)
   PAINTSIZEMENU)
  (T (SETQ PAINTSIZEMENU (|create| MENU
    ITEMS _ '(16 8 4 2 1)))))))))
```

(PAINTW.READBRUSHSHADE

(* |rrb| " 7-Oct-85 14:30")
(* |reads| \a |shade|)

```
(LAMBDA NIL
(PROG (SHADE)
  (* |removed| |the| 16x16 |case| |because| PAINTW |uses| |merge| |mode| |and| |the| |alignment| |of| 16x16 |texture| |is| |off|
  |in| |that| |case.| |When| |fixed| |at| |the| |menu| |item| ("16x16 shade"
  (QUOTE 16X16) "Allows creation of a 16 bits by 16 bits shade"))
  (SELECTQ (SETQ SHADE (MENU (|create| MENU
    CENTERFLG _ T
    TITLE _ "Choose shade"
    ITEMS _ (APPEND (|for| FILLPAT |in| PAINTW.SHADES
      |collect| (LIST (CAR FILLPAT)
        (KWOTE (CADR FILLPAT))
        "changes filling to this pattern"))
      ' ("4 x 4 shade" '4X4 "Allows creation of a 4 bits by
        4 bits shade"))
    MENUBORDERSIZE _ 1)))
    (4X4 (RETURN (PAINTW.READ.AND.SAVE.SHADE)))
    (16X16 (RETURN (PAINTW.READ.AND.SAVE.SHADE T)))
    (RETURN SHADE))))))
```

(PAINTW.READBRUSHTEXTURE

(* |gbn| "25-Jan-86 17:15")

```
(LAMBDA NIL
(SELECTQ (BITS PER PIXEL \COURSE DESTINATION)
  (1 (PAINTW.READBRUSHSHADE))
  (MENU (COLOR MENU (BITS PER PIXEL \COURSE DESTINATION))))))
```

(PAINTW.READ.AND.SAVE.SHADE

(* |rrb| " 4-Oct-85 11:34")
(* |reads| \a |new| |filling.| |confirms| |it| |with| |the| |user| |and|
|adds| |it| |to| PAINTW.SHADES)

```
(LAMBDA (16X16FLG)
(PROG (SHADE)
  (COND
    ((NULL (SETQ SHADE (EDIT SHADE (COND
      (16X16FLG (BITMAPCREATE 16 16))))))
      (* |user| |aborted|)
      (RETURN NIL)))
    (PAINTW.CACHE.SHADE SHADE)
    (RETURN SHADE))))))
```

(PAINTW.CACHE.SHADE

(* |rrb| " 4-Oct-85 11:32")
(* |adds| \a |shade| |to| |the| |global| |cache|.)

```
(LAMBDA (SHADE)
(OR (|for| ENTRY |in| PAINTW.SHADES |when| (EQUAL (CADR ENTRY)
  SHADE)
  |do| (RETURN T))
  (COND
    (PAINTW.SHADES (NCONC1 PAINTW.SHADES (LIST (PAINTW.SHADE.LABEL SHADE)
      SHADE)))
    (T (SETQ PAINTW.SHADES (LIST (LIST (PAINTW.SHADE.LABEL SHADE)
      SHADE))
      'ADDED))))))
```

(PAINTW.SHADE.LABEL

(* |rrb| " 7-Oct-85 14:29")
(* |creates| \a |bitmap| |label| |which| |fills| |it| |with| |the| |texture|

```
(LAMBDA (FILLING)
(FILLING.)
(PROG ((BM (BITMAPCREATE (PLUS 8 (STRINGWIDTH "4 x 4 shade" MENUFONT))
  (FONTPROP MENUFONT 'HEIGHT))))))
```

(BLTSHADE FILLING BM)
(RETURN BM)))

(PAINTW.READCOMMAND

(* [gbn:] "25-Jan-86 16:35")

(LAMBDA NIL
(MENU (COND
 (([type?] MENU PAINTCOMMANDMENU)
 PAINTCOMMANDMENU)
 (T (SETQ PAINTCOMMANDMENU (**[create]** MENU
 ITEMS _ ' ((|SetMode| 'MODE "Allows specification of how new bits
 are merged")
 (|SetShade| 'SHADE "Allows specification of new shade.")
 (|SetShape| 'SHAPE "Allows specification of brush
 shape")
 (|SetSize| 'SIZE "Allows specification of the brush
 size")
 (QUIT 'QUIT "Exits painting mode"))))))))

)

(RPAQ? PAINTW.SHADES)

(DECLARE\ : DOEVAL@COMPILE DONTCOPY

(GLOBALVARS PAINTW.SHADES)
)

(DECLARE\ : DONTEVAL@LOAD DOCOPY

(PAINTW.CACHE.SHADE BLACKSHADE)

(PAINTW.CACHE.SHADE GRAYSHADE)

(PAINTW.CACHE.SHADE HIGHLIGHTSHADE)
)

(RPAQ? PAINTCOMMANDBRUSH ' (ROUND 2))

(RPAQ? PAINTCOMMANDMODE ' PAINT)

(RPAQ? PAINTCOMMANDMENU)

(RPAQ? PAINTCOMMANDSHADE BLACKSHADE)

(RPAQ? PAINTSIZEMENU)

(RPAQ? PAINTSHAPEMENU)

(RPAQ? PAINTSHADEMENU)

(RPAQ? PAINTMODEMENU)

(RPAQ? PAINTWCURSOR)

(PUTPROPS PAINTW COPYRIGHT ("Venue & Xerox Corporation" 1986 1990 1992 1994 1999))

FUNCTION INDEX

PAINW	1	PAINW.READBRUSHSHAPE	3	PAINW.READMODE	2
PAINW.CACHE.SHADE	3	PAINW.READBRUSHSIZE	3	PAINW.SHADE.LABEL	3
PAINW.READ.AND.SAVE.SHADE	3	PAINW.READBRUSHTEXTURE	3		
PAINW.READBRUSHSHADE	3	PAINW.READCOMMAND	4		

VARIABLE INDEX

PAINTCOMMANDBRUSH .4	PAINTCOMMANDMODE ..4	PAINTMODEMENU4	PAINTSHAPEMENU4	PAINW.SHADES4
PAINTCOMMANDMENU ..4	PAINTCOMMANDSHADE .4	PAINTSHADEMENU4	PAINTSIZEMENU4	PAINTWCURSOR4
