```
;;
;; Copyright (c) 1985-1988, 1991, 2022 by Xerox Corporation.


(RPAQQ IDLEHAXCOMS
       [(COMS [ADDVARS (IDLE.FUNCTIONS (Lines 'LINES)
                               (Warp-Out 'WARP)
                               (Radar 'WALKINGSPOKE)
                               [Triangles (FUNCTION (LAMBDA (W)
                                                            (LINES W 3 1 40]
                               [RandAngles (FUNCTION (LAMBDA (W)
                                                            (LINES W (RAND 3 7)
                                                                   (RAND 1 16)
                                                                   (RAND 25 100]
                               (Polygons 'POLYGONS)
                               (Bubbles 'BUBBLES)
                               (Kaleidoscope 'KALDEMO)
                               (Windows 'IDLE-WINDOWS]
              (VARS (IDLE.DEFAULTFN 'LINES)
                    (POLYGONWAIT3 250)))
        (COMS                                                  ; for drawing polygons
              (FNS POLYGONSDEMO POLYGONS CONNECTPOLYS DRAWPOLY1 RANDOMPT)
              (INITVARS (POLYGONSWINDOW))
              (VARS POLYGONWAIT2 POLYGONMINPTS POLYGONMAXPTS POLYGONSTEPS POLYGONWAIT)
              (RECORDS NPOINT))
        (COMS                                                  ; and dots
              (DECLARE%: DONTCOPY (RECORDS KALSTATE KALFIXP)
                     (CONSTANTS KAL.MASK))
              (FNS KALDEMO KAL.ADVANCE KAL.SPOTS KAL.BMS KAL.ORAND))
        (COMS                                                  ; Fun with circles...
              (FNS BUBBLES BUBBLE.CREATE)
              (VARS (BUBBLECNT 20))
              (FNS IDLE-WINDOWS))
        (COMS                                                  ; line drawing demo
              (FNS LINES LINES1 LINES2 LINES3)
              (VARS LINECNT))
        (COMS                                                  ; circles and lines
              (FNS WALKINGSPOKE WARP))
        [COMS                                                  ; melting
              (FNS IDLE-MELT IDLE-SLIDE)
              (VARS MELT-BLOCK-SIZE)
              (ADDVARS (IDLE.FUNCTIONS ("Melt screen" 'IDLE-MELT)
                              ("Slide screen" 'IDLE-SLIDE]
        (COMS                                                  ; utilities
              (FNS DEMOWINDOW)
              (GLOBALVARS BLOCKTIMER)
              (MACROS PERIODIC.BLOCK))
        (COMS [ADDVARS (IDLE.FUNCTIONS ("Drain" 'IDLE-DRAIN]
              (FNS IDLE-DRAIN))
        (COMS (INITVARS (IDLE-SWAP-SIZE 64))
              (FNS IDLE-SWAP)
              (ADDVARS (IDLE.FUNCTIONS ("Swap" 'IDLE-SWAP])

(ADDTOVAR IDLE.FUNCTIONS
          (Lines 'LINES)
          (Warp-Out 'WARP)
          (Radar 'WALKINGSPOKE)
          [Triangles (FUNCTION (LAMBDA (W)
                                     (LINES W 3 1 40]
          [RandAngles (FUNCTION (LAMBDA (W)
                                     (LINES W (RAND 3 7)
                                            (RAND 1 16)
                                            (RAND 25 100]
          (Polygons 'POLYGONS)
          (Bubbles 'BUBBLES)
          (Kaleidoscope 'KALDEMO)
          (Windows 'IDLE-WINDOWS))

(RPAQQ IDLE.DEFAULTFN LINES)

(RPAQQ POLYGONWAIT3 250)

;; for drawing polygons

(DEFINEQ
```

(POLYGONSDEMO
```
[LAMBDA NIL                                                           (* hts%: "20-AUG-83 22:55")
    (POLYGONS [OR POLYGONSWINDOW (SETQ POLYGONSWINDOW (CREATEW '(200 150 600 500]
           T
           (SETUPTIMER 10000))
```

(POLYGONS
```
[LAMBDA (W)                                                          (* lmm "30-Jul-85 20:31")
    (SETQ W (DEMOWINDOW W))
    (LET [(OP (if (VIDEOCOLOR)
                    then 'PAINT
                  else 'ERASE]
        (bind NPOINTS do (SETQ NPOINTS (RAND POLYGONMINPTS POLYGONMAXPTS))
                          (CONNECTPOLYS (for I from 1 to NPOINTS collect (RANDOMPT W))
                            (for I from 1 to NPOINTS collect (RANDOMPT W))
                              POLYGONSTEPS W OP)
                          (DISMISS POLYGONWAIT])
```

(CONNECTPOLYS
```
[LAMBDA (FROMS TOS NSTEPS W OPERATION)                     ; Edited 23-Aug-2022 08:10 by larry
                                                           (* lmm "30-Jul-85 17:19")
    (PROG (DIFFS)
          (CLEARW W)
          (LINES2 FROMS 3 W OPERATION)
          (SETQ DIFFS (for FPT in FROMS as TPT in TOS bind DX DY
                        collect (SETQ DX (IQUOTIENT (IDIFFERENCE (fetch XC of TPT)
                                                    (fetch XC of FPT))
                                            POLYGONSTEPS))
                                (SETQ DY (IQUOTIENT (IDIFFERENCE (fetch YC of TPT)
                                                    (fetch YC of FPT))
                                            POLYGONSTEPS))
                                (replace XC of TPT with (IPLUS (fetch XC of FPT)
                                                            (ITIMES POLYGONSTEPS DX)))
                                (replace YC of TPT with (IPLUS (fetch YC of FPT)
                                                            (ITIMES POLYGONSTEPS DY)))
                                (CONS DX DY)))
          (LINES2 TOS 3 W OPERATION)
          (for FPT in FROMS as TPT in TOS do (DRAWLINE (fetch XC of FPT)
                                                       (fetch YC of FPT)
                                                       (fetch XC of TPT)
                                                       (fetch YC of TPT)
                                                       1 OPERATION W)
                                             (DISMISS POLYGONWAIT2))
          (CLEARW W)
          (for I from 1 to POLYGONSTEPS do (DISMISS POLYGONWAIT3)
                                           (LINES2 FROMS 1 W OPERATION)
                                           (for PT in FROMS as DIF in DIFFS
                                              do (add (fetch XC of PT)
                                                        (CAR DIF))
                                                 (add (fetch YC of PT)
                                                        (CDR DIF)))
             finally (LINES2 FROMS 1 W OPERATION])
```

(DRAWPOLY1
```
[LAMBDA (PTLIST WIDTH OPERATION W NOBLOCK)                 (* edited%: "19-AUG-83 04:14")

        (* draws a closed polygon of the points given If OPERATION is not given, use the one from the default DS.)

    [COND
      (PTLIST (OR OPERATION (SETQ OPERATION (DSPOPERATION NIL W)))
            (PROG ((PTS PTLIST))
                  (while (CDR PTS) do (DRAWLINE (fetch XC of (CAR PTS))
                                               (fetch YC of (CAR PTS))
                                               (fetch XC of (CADR PTS))
                                               (fetch YC of (CADR PTS))
                                               WIDTH OPERATION W)
                                      (pop PTS)
                     finally (DRAWLINE (fetch XC of (CAR PTS))
                                       (fetch YC of (CAR PTS))
                                       (fetch XC of (CAR PTLIST))
                                       (fetch YC of (CAR PTLIST))
                                       WIDTH OPERATION W]
    (COND
      (NOBLOCK (ALLOW.BUTTON.EVENTS))
      (T (BLOCK])
```

(RANDOMPT
```
[LAMBDA (DS)                                               (* edited%: "18-AUG-83 16:22")
    (PROG ((REG (DSPCLIPPINGREGION NIL DS)))
          (RETURN (create NPOINT
                    XC _ (RAND (fetch LEFT of REG)
                               (fetch RIGHT of REG))
```

```
                          YC _ (RAND (fetch BOTTOM of REG)
                                     (fetch TOP of REG])
)

(RPAQ? POLYGONSWINDOW )

(RPAQQ POLYGONWAIT2 25)

(RPAQQ POLYGONMINPTS 3)

(RPAQQ POLYGONMAXPTS 9)

(RPAQQ POLYGONSTEPS 35)

(RPAQQ POLYGONWAIT 2000)

(DECLARE%: EVAL@COMPILE

(DATATYPE NPOINT ((XC XPOINTER)
                  (YC XPOINTER)))
)

(/DECLAREDATATYPE 'NPOINT '(XPOINTER XPOINTER)
```

        ;; ---field descriptor list elided by lister---

```
        '4)
```

;; and dots

```
(DECLARE%: DONTCOPY

(DECLARE%: EVAL@COMPILE

(RECORD KALSTATE (A B C PERIODCOUNT PERIOD))

(BLOCKRECORD KALFIXP ((VALUE FIXP)))
)

(DECLARE%: EVAL@COMPILE

(RPAQQ KAL.MASK 65535)

(CONSTANTS KAL.MASK)
)
)

(DEFINEQ
```

(**KALDEMO**
```
  [LAMBDA (W PERIOD PERSISTENCE)                            ; Edited 23-Aug-2022 08:49 by lmm
                                                           (* lmm " 5-Aug-85 22:16")

    (OR PERIOD (SETQ PERIOD (RAND 16 128)))
    [OR PERSISTENCE (SETQ PERSISTENCE (LSH 1 (RAND 14 23]
    (SETQ W (DEMOWINDOW W))
    (LET ((XSTATEB (create KALSTATE
                           A _ 1
                           B _ -1849
                           C _ (RAND 2 4)
                           PERIOD _ PERIOD
                           PERIODCOUNT _ 1))
          (XSTATEE (create KALSTATE))
          (YSTATEB (create KALSTATE
                           A _ 1
                           B _ -1809
                           C _ (RAND 0 20)
                           PERIOD _ PERIOD
                           PERIODCOUNT _ 1))
          (YSTATEE (create KALSTATE))
          [WINDOWSIDE (MIN (WINDOWPROP W 'HEIGHT)
                           (WINDOWPROP W 'WIDTH]
          (TIMER (SETUPTIMER 0 NIL 'TICKS))
          (BLACK (NOT (VIDEOCOLOR)))
          XOFFSET)
         (SETQ XOFFSET (QUOTIENT (MAX (DIFFERENCE (WINDOWPROP W 'WIDTH)
                                                  WINDOWSIDE)
                                      0)
                                 2))
         (SETQ XSTATEE (COPY XSTATEB))
         (SETQ YSTATEE (COPY YSTATEB))
         (from 1 to PERSISTENCE do (KAL.ADVANCE XSTATEB)
                                   (KAL.ADVANCE YSTATEB)
                                   (KAL.SPOTS (ffetch A of XSTATEB)
                                              (ffetch A of YSTATEB)
                                              WINDOWSIDE W BLACK XOFFSET)
                                   (BLOCK 100 TIMER))
         (do (KAL.ADVANCE XSTATEE)
```

```
                    (KAL.ADVANCE YSTATEE)
                    [PROG ((X0 (LOGAND (LRSH (ffetch A of XSTATEE)
                                            7)
                                      KAL.MASK))
                           (Y0 (LOGAND (LRSH (ffetch A of YSTATEE)
                                            7)
                                      KAL.MASK))
                          X1 Y1)
                          (COND
                             ((ILESSP X0 Y0)
                              (SETQ X1 (IDIFFERENCE (SUB1 WINDOWSIDE)
                                            X0))
                              (SETQ Y1 (IDIFFERENCE (SUB1 WINDOWSIDE)
                                            Y0))
                              (KAL.BMS W X0 Y0 X1 Y1 (if BLACK
                                                         then 1
                                                         else 0)
                                    XOFFSET]
                    (KAL.ADVANCE XSTATEB)
                    (KAL.ADVANCE YSTATEB)
                    (KAL.SPOTS (ffetch A of XSTATEB)
                           (ffetch A of YSTATEB)
                           WINDOWSIDE W BLACK XOFFSET)
                    (PERIODIC.BLOCK TIMER])
```

(**KAL.ADVANCE**
```
  [LAMBDA (STATE)                                                ; Edited 26-Jun-2022 18:20 by rmk
                                                                 (* lmm "30-Jul-85 20:16")
      (freplace (KALSTATE A) of STATE with (KAL.ORAND (ffetch (KALSTATE A) of STATE)
                                                  (ffetch (KALSTATE B) of STATE)))
      (freplace (KALSTATE PERIODCOUNT) of STATE with (SUB1 (ffetch (KALSTATE PERIODCOUNT) of STATE)))
      (COND
         ((EQ (ffetch (KALSTATE PERIODCOUNT) of STATE)
              0)
          (freplace (KALSTATE B) of STATE with (KAL.ORAND (ffetch (KALSTATE B) of STATE)
                                                  (ffetch (KALSTATE C) of STATE)))
          (freplace (KALSTATE PERIODCOUNT) of STATE with (ffetch (KALSTATE PERIOD) of STATE])
```

(**KAL.SPOTS**
```
  [LAMBDA (X Y WINDOWSIDE W BLACK XOFFSET)                       (* lmm " 3-Aug-85 21:59")
      (PROG ((X0 (LRSH X 7))
             (Y0 (LRSH Y 7))
            X1 Y1 C)
            (COND
               ((ILESSP X0 Y0)
                (SETQ X1 (IDIFFERENCE (SUB1 WINDOWSIDE)
                              X0))
                (SETQ Y1 (IDIFFERENCE (SUB1 WINDOWSIDE)
                              Y0))
                (SETQ C (LOGAND X 1))
                (KAL.BMS W X0 Y0 X1 Y1 (if BLACK
                                            then (DIFFERENCE 1 C)
                                            else C)
                      XOFFSET])
```

(**KAL.BMS**
```
  [LAMBDA (W X0 Y0 X1 Y1 C XOFFSET)                              (* lmm "30-Jul-85 19:38")
      (UNINTERRUPTABLY
          (if (EQUAL (GETWINDOWPROP W 'REGION)
                     WHOLESCREEN)
              then (TOTOPW W)
                   (SETQ W (SCREENBITMAP)))
          (BITMAPBIT W (PLUS XOFFSET X0)
                 Y0 C)
          (BITMAPBIT W (PLUS XOFFSET Y0)
                 X0 C)
          (BITMAPBIT W (PLUS XOFFSET X1)
                 Y0 C)
          (BITMAPBIT W (PLUS XOFFSET Y0)
                 X1 C)
          (BITMAPBIT W (PLUS XOFFSET X1)
                 Y1 C)
          (BITMAPBIT W (PLUS XOFFSET Y1)
                 X1 C)
          (BITMAPBIT W (PLUS XOFFSET X0)
                 Y1 C)
          (BITMAPBIT W (PLUS XOFFSET Y1)
                 X0 C))])
```

(**KAL.ORAND**
```
  [LAMBDA (A B)                                                  ; Edited 26-Sep-91 14:34 by jds
      (LET [(BOX (CONSTANT (NCREATE 'FIXP]
           (replace (KALFIXP VALUE) of BOX with A)
```

```
            (\BOXIPLUS BOX B)
            (LOGXOR (LOGAND BOX KAL.MASK)
                    (LOGAND B KAL.MASK])
)
```

;; Fun with circles...

```
(DEFINEQ
```

(**BUBBLES**
```
  [LAMBDA (W)                                                      ; Edited 23-Aug-2022 08:14 by larry
                                                                   (* lmm "30-Jul-85 20:35")

    [WINDOWPROP (SETQ W (DEMOWINDOW W))
            'RESHAPEFN
            (FUNCTION (LAMBDA (W)
                          (DSPFILL NIL (if (VIDEOCOLOR)
                                           then WHITESHADE
                                           else BLACKSHADE)
                                   'REPLACE W]
    (DSPFILL NIL (if (VIDEOCOLOR)
                     then WHITESHADE
                     else BLACKSHADE)
            'REPLACE W)
    (bind (ARRAY _ (ARRAY BUBBLECNT 'POINTER))
          (I _ 1)
        CIRCLE eachtime (SETQ I (if (EQ I BUBBLECNT)
                                    then 1
                                    else (ADD1 I)))
      do
        (* * first erase the circle at I in array)

        (SETQ CIRCLE (ELT ARRAY I))
        (DSPOPERATION (if (VIDEOCOLOR)
                          then 'ERASE
                          else 'PAINT)
              W)                                                   (* there will be no circle at I the first time through)
        (AND CIRCLE (DRAWCIRCLE (CAR CIRCLE)
                                (CADR CIRCLE)
                                (CADDR CIRCLE)
                                NIL NIL W))

        (* * now put a new circle in array at I and draw it)

        (SETQ CIRCLE (SETA ARRAY I (BUBBLE.CREATE W)))
        (DSPOPERATION 'REPLACE W)                                  (* fill center w/ black so it ocludes ones under)
        (FILLCIRCLE (CAR CIRCLE)
               (CADR CIRCLE)
               (SUB1 (CADDR CIRCLE))
               (if (VIDEOCOLOR)
                   then WHITESHADE
                   else BLACKSHADE)
               W)
        (DSPOPERATION 'INVERT W)
        (DRAWCIRCLE (CAR CIRCLE)
               (CADR CIRCLE)
               (CADDR CIRCLE)
               NIL NIL W)
        (BLOCK 100])
```

(**BUBBLE.CREATE**
```
  [LAMBDA (W)                                                      (* drc%: "29-Jul-85 13:51")
    (LET* [(REGION (WINDOWPROP W 'REGION))
           (WIDTH (SUB1 (fetch WIDTH of REGION)))
           (HEIGHT (SUB1 (fetch HEIGHT of REGION)))
           (CENTERX (RAND 1 (SUB1 WIDTH)))
           (CENTERY (RAND 1 (SUB1 HEIGHT]
        (LIST CENTERX CENTERY (RAND 1 (IMIN (IDIFFERENCE WIDTH CENTERX)
                                            CENTERX
                                            (IDIFFERENCE HEIGHT CENTERY)
                                            CENTERY])
)
```

```
(RPAQQ BUBBLECNT 20)
```

```
(DEFINEQ
```

(**IDLE-WINDOWS**
```
  [LAMBDA (W DELAY)                                                ; Edited 23-Aug-2022 08:35 by lmm
                                                                   (* lmm " 7-Jun-86 22:21")

    (SETQ W (DEMOWINDOW W))
    (PROG [(D (WINDOWPROP W 'WIDTH))
           (H (WINDOWPROP W 'HEIGHT]
          (LET [(TITLE (WINDOWPROP (CREATEW (LIST 0 0 D (HEIGHTIFWINDOW 0 T))
                                            "Yet another window" NIL T)
```

```
                                           'IMAGECOVERED]
                       (while T do (PROG [[X (RAND 0 (- D (+ 2 2 100]
                                         (Y (RAND 0 (- H 8 100]
                                       (PROG [[D0 (MAX 100 (RAND 100 (- D X]
                                              (H0 (MAX 100 (RAND 100 (- H Y]
                                             (BITBLT NIL NIL NIL W X Y D0 2 'TEXTURE 'REPLACE BLACKSHADE)
                                             (BITBLT NIL NIL NIL W X Y 2 H0 'TEXTURE 'REPLACE BLACKSHADE)
                                             (BITBLT NIL NIL NIL W (+ X (- D0 2))
                                                     Y 2 H0 'TEXTURE 'REPLACE BLACKSHADE)
                                             (BITBLT TITLE NIL (+ WBorder (QUOTIENT WBorder 2))
                                                     W X (+ Y H0)
                                                     D0 NIL NIL 'REPLACE)
                                             (BITBLT NIL NIL NIL W (+ X 2)
                                                     (+ Y 2)
                                                     (- D0 (+ 2 2))
                                                     (- H0 2)
                                                     'TEXTURE
                                                     'ERASE BLACKSHADE)))
                               (BLOCK (OR DELAY 500]
)


;; line drawing demo

(DEFINEQ

(LINES
  [LAMBDA (W N LCNT STEPS ODDSTEP)                           (* lmm "27-Sep-85 00:50")
    (SETQ W (DEMOWINDOW W))
    (OR STEPS (SETQ STEPS POLYGONSTEPS))
    (OR N (SETQ N 2))
    (RESETLST
        (PROG ((LINES (to (OR LCNT (if (NEQ N 2)
                                       then (ADD1 (QUOTIENT LINECNT N))
                                     else LINECNT))
                          collect NIL))
               (CNT 0)
               FROMPTS TOPTS DXS (ODDSTART))
              (RESETSAVE NIL (LIST (FUNCTION RPLACD)
                                   LINES))
              (NCONC LINES LINES)
              (SETQ FROMPTS (to N collect (RANDOMPT W)))
              (bind (TIMER _ (SETUPTIMER 0 NIL 'TICKS)) while T
                 do [COND
                       ((ILEQ CNT 0)
                        [SETQ TOPTS (bind (ODDP _ (SETQ ODDSTART (NOT ODDSTART))) for X in FROMPTS
                                          collect (if (AND ODDSTEP (SETQ ODDP (NOT ODDP)))
                                                      then X
                                                    else (RANDOMPT W]
                        [SETQ DXS (for TP in TOPTS as FP in FROMPTS
                                     collect (create NPOINT
                                                     XC _ (QUOTIENT (DIFFERENCE (fetch XC TP)
                                                                                (fetch XC FP))
                                                                    STEPS)
                                                     YC _ (QUOTIENT (DIFFERENCE (fetch YC TP)
                                                                                (fetch YC FP))
                                                                    STEPS]
                        (SETQ CNT STEPS))
                       (T (SETQ CNT (SUB1 CNT]
                    (LINES1 FROMPTS LINES W)
                    (for X in FROMPTS as D in DXS do (add (fetch XC X)
                                                          (fetch XC D))
                                                     (add (fetch YC X)
                                                          (fetch YC D)))
                    (SETQ LINES (CDR LINES))
                    (PERIODIC.BLOCK TIMER)))]])


(LINES1
  [LAMBDA (ENDPOINTS LINES DSP)                              ; Edited 23-Aug-2022 07:59 by larry
                                                             (* lmm "30-Jul-85 17:33")
    (PROG (PTS)
          [COND
             ((SETQ PTS (CAR LINES))                         (* ERASE OLD)
              (LINES3 (CAR LINES)
                      1 DSP 'INVERT ENDPOINTS))
             (T [RPLACA LINES (SETQ PTS (in ENDPOINTS collect (create NPOINT]
                (LINES2 ENDPOINTS 1 DSP 'INVERT]
          (BLOCK 75)
          (for PT in PTS as EP in ENDPOINTS do (replace XC of PT with (fetch XC of EP))
                                               (replace YC of PT with (fetch YC of EP])


(LINES2
  [LAMBDA (ENDPOINTS WIDTH WINDOW OPERATION)                 (* lmm "30-Jul-85 17:14")
    (for (X _ ENDPOINTS) while (OR (CDR X)
                                   (if (CDDR ENDPOINTS)
```

```
                                              then X))
          do (DRAWLINE (fetch XC (CAR X))
                       (fetch YC (CAR X))
                       (fetch XC (CAR (OR (SETQ X (CDR X))
                                          ENDPOINTS)))
                       (fetch YC (CAR (OR X ENDPOINTS)))
                       WIDTH OPERATION WINDOW])
```

(**LINES3**
```
  [LAMBDA (ENDPOINTS WIDTH WINDOW OPERATION EP2)              (* lmm "30-Jul-85 17:33")
    (for (X _ ENDPOINTS) while (OR (CDR X)
                                   (if (CDDR ENDPOINTS)
                                       then X))
        bind (Y _ EP2) do (DRAWLINE (fetch XC (CAR X))
                                    (fetch YC (CAR X))
                                    (fetch XC (CAR (OR (SETQ X (CDR X))
                                                       ENDPOINTS)))
                                    (fetch YC (CAR (OR X ENDPOINTS)))
                                    WIDTH OPERATION WINDOW)
                          (DRAWLINE (fetch XC (CAR Y))
                                    (fetch YC (CAR Y))
                                    (fetch XC (CAR (OR (SETQ Y (CDR Y))
                                                       EP2)))
                                    (fetch YC (CAR (OR Y EP2)))
                                    WIDTH OPERATION WINDOW])
)
```

(RPAQQ **LINECNT** 100)

;; circles and lines

(DEFINEQ

(**WALKINGSPOKE**
```
  [LAMBDA (W)                                                 (* lmm "19-Mar-86 17:49")
    (LET
     ((W (DEMOWINDOW W))
      (SINARRAY (ARRAY 360 NIL NIL 0))
      (MARGIN (QUOTIENT MAX.SMALLP SCREENWIDTH)))
     [for N from 0 to 359 do (SETA SINARRAY N (FIXR (TIMES MARGIN (SIN N]
     (CLEARW W)
     (do (PROG [(WIDTH (WINDOWPROP W 'WIDTH))
                (HEIGHT (WINDOWPROP W 'HEIGHT]                (* for YY from 0 to 298 do (DRAWLINE 0 YY 298 YY 1
                                                              (QUOTE INVERT) RADARWINDOW))
               (LET ((R (QUOTIENT (RAND (MIN 100 WIDTH HEIGHT)
                                        (MIN WIDTH HEIGHT))
                                  2)))
                    (LET [(X (RAND R (DIFFERENCE WIDTH R)))
                          (Y (RAND R (DIFFERENCE HEIGHT R]
                         (RPTQ 2 (for N from 0 to 359
                                     do (DRAWLINE X Y (PLUS X (QUOTIENT (TIMES R (ELT SINARRAY
                                                                                      (IMOD (PLUS N 90)
                                                                                            360)))
                                                                       MARGIN))
                                                      (PLUS Y (QUOTIENT (TIMES R (ELT SINARRAY N))
                                                                        MARGIN))
                                                      2
                                                      'INVERT W)
                                     (BLOCK))
                         (RECLAIM])
```

(**WARP**
```
  [LAMBDA (W)                                                 ; Edited 23-Aug-2022 08:01 by larry
                                                              (* hdj " 1-Apr-86 14:22")
     (do (CLEARW W)
         (LET ((OLDOP (DSPOPERATION 'INVERT W)))
              [LET [(WIDTH (WINDOWPROP W 'WIDTH))
                    (HEIGHT (WINDOWPROP W 'HEIGHT]
                   (LET ((CENTERX (RAND 0 WIDTH))
                         (CENTERY (RAND 0 HEIGHT)))
                        (for RADIUS from (RAND 5 250) to 5 by -2 do (FILLCIRCLE (PLUS CENTERX (RAND 0 2))
                                                                               (PLUS CENTERY (RAND 0 2))
                                                                               RADIUS BLACKSHADE W)
                                                                   (BLOCK 75]
              (DSPOPERATION OLDOP W])
)
```

;; melting

(DEFINEQ

(**IDLE-MELT**

```
  [LAMBDA (WINDOW SIZE INITIAL PATH)                              ; Edited 23-Aug-2022 08:20 by larry
                                                                  ; Edited 10-Jun-88 17:15 by MASINTER

    (OR SIZE (SETQ SIZE MELT-BLOCK-SIZE))
    (SETQ WINDOW (DEMOWINDOW WINDOW))
    (PROG ((W (WINDOWPROP WINDOW 'WIDTH))
           (H (WINDOWPROP WINDOW 'HEIGHT))
           BM
           (TAIL INITIAL)
           TIMER)
      REPAINT
          (CLEARW WINDOW)
          [SETQ BM (OR (CAR TAIL)
                       (WINDOWPROP WINDOW 'IMAGECOVERED]
          (for BITMAP inside BM do (BITBLT (SETQ BITMAP (if (BITMAPP BITMAP)
                                                           then BITMAP
                                                         elseif (CL:SYMBOLP BITMAP)
                                                           then (CAR (READBRUSHFILE BITMAP))
                                                           else (IDLE.BITMAP NIL BITMAP)))
                                      NIL NIL WINDOW (RAND 0 (- W (BITMAPWIDTH BITMAP)))
                                      (RAND 0 (- H (BITMAPHEIGHT BITMAP)))
                                      NIL NIL (if (VIDEOCOLOR)
                                                then NIL
                                                else 'INVERT)
                                      'REPLACE))
          (if INITIAL
              then [SETQ TIMER (AND (CADR TAIL)
                                    (SETUPTIMER (CADR TAIL)
                                           TIMER
                                           'SECONDS
                                           'SECONDS]
                   (SETQ TAIL (OR (CDDR TAIL)
                                  INITIAL)))
          [do (LET [(X (RAND 0 (- W SIZE)))
                    (Y (RAND 0 (- H SIZE]
                   (BITBLT WINDOW X Y WINDOW (+ X (RAND -1 1))
                          (+ Y (RAND -1 1))
                          SIZE SIZE NIL 'REPLACE))
              (BLOCK 100) repeatuntil (AND TIMER (TIMEREXPIRED? TIMER 'SECONDS]
          (GO REPAINT])
```

(**IDLE-SLIDE**
```
  [LAMBDA (W SIZE SPEED COUNT SOURCE)                             ; Edited 10-Jun-88 17:12 by MASINTER
    (OR SIZE (SETQ SIZE 128))
    (OR COUNT (SETQ COUNT 120))
    (OR SPEED (SETQ SPEED 2))
    (SETQ W (DEMOWINDOW W))
    (BITBLT [OR SOURCE (SETQ SOURCE (WINDOWPROP W 'IMAGECOVERED]
           NIL NIL W NIL NIL NIL NIL (if (VIDEOCOLOR)
                                       then NIL
                                       else 'INVERT)
           'REPLACE)
    (LET [(D (WINDOWPROP W 'WIDTH))
          (H (WINDOWPROP W 'HEIGHT]
         (LET [(XMAX (- D SIZE))
               (YMAX (- H SIZE))
               X Y DX DY (CNT 1)
               DDX DDY (TIMER (SETUPTIMER 0 NIL 'TICKS]
              (do (COND
                    ((OR (EQ (add CNT -1)
                             0)
                         (< X 0)
                         (> X XMAX)
                         (< Y 0)
                         (> Y YMAX))
                     (SETQ X (RAND 0 XMAX))
                     (SETQ Y (RAND 0 YMAX))
                     (SETQ DX (RAND (- SPEED)
                                    SPEED))
                     (SETQ DY (RAND (- SPEED)
                                    SPEED))
                     (BITBLT SOURCE X Y W X Y SIZE SIZE NIL 'REPLACE)
                     (SETQ DDX DY)
                     (SETQ DDY DX)
                     (SETQ CNT COUNT)))
                  (BITBLT W X Y W (+ X DDX)
                         (+ Y DDY)
                         SIZE SIZE NIL 'REPLACE)
                  (add X DX)
                  (add Y DY)
                  (PERIODIC.BLOCK TIMER])

)

(RPAQQ MELT-BLOCK-SIZE 32)

(ADDTOVAR IDLE.FUNCTIONS ("Melt screen" 'IDLE-MELT)
```

```
                              ("Slide screen" 'IDLE-SLIDE))
```

;; utilities

```
(DEFINEQ
```

(DEMOWINDOW
```
  [LAMBDA (W)                                                    (* lmm "30-Jul-85 20:34")
    [OR W (SETQ W (OR POLYGONSWINDOW (SETQ POLYGONSWINDOW (CREATEW]
    (DSPTEXTURE (if (VIDEOCOLOR)
                    then WHITESHADE
                  else BLACKSHADE)
          W)
    (DSPOPERATION 'INVERT W)
    (CLEARW W)
    W])
)
```

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

```
(GLOBALVARS BLOCKTIMER)
)
```

```
(DECLARE%: EVAL@COMPILE
```

```
(PUTPROPS PERIODIC.BLOCK MACRO ((TIMER)
                                (BLOCK 100 TIMER)))
)
```

```
(ADDTOVAR IDLE.FUNCTIONS ("Drain" 'IDLE-DRAIN))
```

```
(DEFINEQ
```

(IDLE-DRAIN
```
  [LAMBDA (WINDOW)                                               ; Edited 23-Aug-2022 07:52 by larry
                                                                 (* hdj "28-May-86 11:52")

    (do (BITBLT (WINDOWPROP WINDOW 'IMAGECOVERED)
               NIL NIL WINDOW NIL NIL NIL NIL 'INVERT 'REPLACE)
        (LET* ((WIDTH (WINDOWPROP WINDOW 'WIDTH))
               (HEIGHT (WINDOWPROP WINDOW 'HEIGHT))
               (HALF-WIDTH (IQUOTIENT WIDTH 2))
               (HALF-HEIGHT (IQUOTIENT HEIGHT 2)))
             (for EDGE from 0 to (MIN HALF-WIDTH HALF-HEIGHT) do (BLOCK 100)
                                                                 (BITBLT WINDOW EDGE 0 WINDOW (PLUS 1 EDGE)
                                                                        0
                                                                        (- HALF-WIDTH EDGE)
                                                                        HEIGHT
                                                                        'INPUT
                                                                        'REPLACE)
                                                                 (BLTSHADE WHITESHADE WINDOW EDGE 0 1 HEIGHT
                                                                        'REPLACE)
                                                                 (BITBLT WINDOW (PLUS 1 HALF-WIDTH)
                                                                        0 WINDOW HALF-WIDTH 0 (- HALF-WIDTH EDGE
                                                                                               )
                                                                        HEIGHT
                                                                        'INPUT
                                                                        'REPLACE)
                                                                 (BLTSHADE WHITESHADE WINDOW (- WIDTH EDGE)
                                                                        0 1 HEIGHT 'REPLACE)
                                                                 (BITBLT WINDOW 0 EDGE WINDOW 0 (PLUS 1 EDGE)
                                                                        WIDTH
                                                                        (- HALF-HEIGHT EDGE)
                                                                        'INPUT
                                                                        'REPLACE)
                                                                 (BLTSHADE WHITESHADE WINDOW 0 EDGE WIDTH 1
                                                                        'REPLACE)
                                                                 (BITBLT WINDOW 0 (+ HALF-HEIGHT EDGE 1)
                                                                        WINDOW 0 (+ HALF-HEIGHT EDGE)
                                                                        WIDTH
                                                                        (- HALF-HEIGHT EDGE)
                                                                        'INPUT
                                                                        'REPLACE)
                                                                 (BLTSHADE WHITESHADE WINDOW 0 (- HEIGHT EDGE)
                                                                        WIDTH 1 'REPLACE])
)
```

```
(RPAQ? IDLE-SWAP-SIZE 64)
```

```
(DEFINEQ
```

(IDLE-SWAP
```
  [LAMBDA (WINDOW)                                               ; Edited 28-Sep-2022 19:48 by lmm
                                                                 (* hdj "29-May-86 23:41")

    (DECLARE (GLOBALVARS IDLE-SWAP-SIZE))
    (BITBLT (WINDOWPROP WINDOW 'IMAGECOVERED)
```

```
          NIL NIL WINDOW NIL NIL NIL NIL 'INVERT 'REPLACE)
    (LET [(WIDTH (WINDOWPROP WINDOW 'WIDTH))
          (HEIGHT (WINDOWPROP WINDOW 'HEIGHT]
         (do (BLOCK 250)
             (LET [[RAND-X-1 (TIMES IDLE-SWAP-SIZE (RAND 0 (IQUOTIENT WIDTH IDLE-SWAP-SIZE]
                   [RAND-Y-1 (TIMES IDLE-SWAP-SIZE (RAND 0 (IQUOTIENT HEIGHT IDLE-SWAP-SIZE]
                   [RAND-X-2 (TIMES IDLE-SWAP-SIZE (RAND 0 (IQUOTIENT WIDTH IDLE-SWAP-SIZE]
                   (RAND-Y-2 (TIMES IDLE-SWAP-SIZE (RAND 0 (IQUOTIENT HEIGHT IDLE-SWAP-SIZE]
                  (if (AND (NEQ RAND-X-1 RAND-X-2)
                           (NEQ RAND-Y-1 RAND-Y-2))
                      then
```

(* * swap the two regions of the window using BITBLT only)

```
                             (BITBLT WINDOW RAND-X-1 RAND-Y-1 WINDOW RAND-X-2 RAND-Y-2 IDLE-SWAP-SIZE
                                 IDLE-SWAP-SIZE 'INPUT 'INVERT)
                             (BITBLT WINDOW RAND-X-2 RAND-Y-2 WINDOW RAND-X-1 RAND-Y-1 IDLE-SWAP-SIZE
                                 IDLE-SWAP-SIZE 'INPUT 'INVERT)
                             (BITBLT WINDOW RAND-X-1 RAND-Y-1 WINDOW RAND-X-2 RAND-Y-2 IDLE-SWAP-SIZE
                                 IDLE-SWAP-SIZE 'INPUT 'INVERT])
```

)

(ADDTOVAR **IDLE.FUNCTIONS** ("Swap" 'IDLE-SWAP))

(PUTPROPS **IDLEHAX COPYRIGHT** ("Xerox Corporation" 1985 1986 1987 1988 1991 2022))

FUNCTION INDEX

VARIABLE INDEX

RECORD INDEX

MACRO INDEX

CONSTANT INDEX