Virtual Reality Explored

The magazine and
support group
exclusively for users
of the Archimedes

MouseTrap

Confining Your Mouse

Peter Arnold presents an application to confine your mouse pointer to vertical and horizontal directions; guaranteed harmless to rodents.

Drawing perfectly straight lines or right angles in Draw or Paint can be quite a tricky affair. There are ways round the problem (like locking to a grid), but wouldn't it be nice if you could confine your mouse pointer to move only up and down or left and right? Well, the answer's yes, and just to prove it here is an application that does all this and more.

When this application is run, a little window appears containing two radio icons: (*Vertical only* and *Horizontal only*) and a yes/no icon (*Confined*). Clicking on the radio icons defines which type of confinement is applied to the pointer (i.e. confinement to either the vertical or horizontal direction), and clicking on the yes/no icon turns confinement on or off.

This sounds great, but isn't much use if you want to change the confinement direction half way through an operation (for example if you're drawing a rectangle), or if you want to confine the pointer away from the window. This is where the

The MouseTrap window

key shortcuts come in: F1 will confine movement to the vertical direction only, F2 will switch to the horizontal direction, and F3 toggles confinement on or off. So, to draw a perfect right angle in Draw, position the mouse at one corner, press F1, move the mouse up to the second corner and click, press F2, and then move to the third point and click: 90° precisely.

Note that if a window is open for an application which traps the F1-F3 keys (like Edit or DeskEdit for instance) then the application will not work correctly. Closing the window or removing the input focus from it should solve the problem.

Another handy feature of the program is that it allows the pointer to be moved using the cursor keys, assuming again that these are not trapped by a window that owns the caret. The cursor keys on their own move the pointer slowly for accuracy, and with the Shift key held down the pointer moves much more quickly.

CREATING THE APPLICATION

To enter the application, you must create a directory called *!MouseTrap*, and open it up by double-clicking on it while holding down Shift. Next use Edit to create an Obey file called *!Run* containing the following three lines, and save inside the *!MouseTrap* directory:

```
WimpSlot -min 16K -max 16K
Run <Obey$Dir>.!RunImage
```
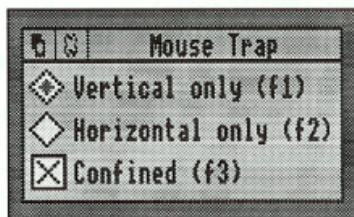
Next, type in the listing given here and save it as *!RunImage* inside *!MouseTrap*. Finally, use Paint to design a sprite called *!mousetrap* (mode 12, 34 pixels wide by 17 pixels high), and save this file along with the others (of course, you can borrow a sprite from another application to save designing a new one, but don't forget to rename it).

So there you have it - mouse confinement made easy.

PROGRAM NOTES

The program is a good example of how to use the standard radio and yes/no icons in windows. Basically you must create the sprite icon with a validation string "Sradiooff,radioon" for radio icons, and "Sno,yes" for yes/no icons (option icons would have a validation string "Soptoff/opton"), and then to change the

state of the icon you simply set or unset the *Icon Selected* bit in the icon flags (using the SWI call *Wimp_SetIconState*).

This then "highlights" the icon, but because you have provided two sprite names in the validation string, the sprite is changed to the second sprite name given, rather than just being inverted (which is what normally happens to highlighted sprites, as in Filer windows). In the program, the icons are set up by *PROCCreateIcons* in line 1260 onwards, and the icon states are changed in *PROCSetIcons* from line 800.

```
 10 REM            >!RunImage
 20 REM Program    MouseTrap
 30 REM Author     Peter Arnold
 40 REM Version    A 1.10
 50 REM RISC User July 1992
 60 REM Program    Subject to Copyright
 70 REM            Not Public Domain
 80 :
 90 PROCInit
100 ON ERROR PROCError(ERR,REPORT$+" a
t line "+STR$(ERL))
110 REPEAT
120 SYS "Wimp_Poll",1,Event% TO Action
%
130 CASE Action% OF
140 WHEN 2:SYS "Wimp_OpenWindow",,Even
t%
150 WHEN 3:SYS "Wimp_CloseWindow",,Eve
nt%:PROCAllOver
160 WHEN 6:PROCMouse(Event%!16)
170 WHEN 8:PROCKeys(Event%!24)
180 WHEN 17,18:PROCReceive(Event%!16)
190 ENDCASE
200 UNTIL FALSE
210 PROCAllOver
220 END
230 :
240 DEF PROCError(err%, report$)
250 Size%=LEN(report$)+4
260 DIM Block% Size%
270 !Block% = err%
280 $(Block%+4)=report$
290 SYS "Wimp_ReportError", Block%, 1,
"MouseTrap"
300 PROCAllOver
310 ENDPROC
320 :
330 DEF PROCInit
340 Confined%=FALSE:Vertical%=TRUE
350 SYS "OS_Byte",161,194 TO ,,Step%
360 DIM q% 256, Indirect% 140, Event%
256
370 $q%="TASK"
380 SYS "Wimp_Initialise",200,!q%,"Mou
seTrap"
390 PROCOpenWindow
400 PROCSetIcons
410 ENDPROC
420 :
430 DEF PROCMouse(icon%)
440 IF icon%=0 THEN Vertical%=TRUE
450 IF icon%=1 THEN Vertical%=FALSE
460 IF icon%=2 THEN Confined%=NOT Conf
ined%
470 PROCConfine
480 PROCSetIcons
490 ENDPROC
500 :
510 DEF PROCKeys(Key%)
520 LOCAL x%,y%,z%
530 MOUSE x%,y%,z%
540 CASE Key% OF
550 WHEN &181 : Vertical%=TRUE: PROCCo
nfine: PROCSetIcons
560 WHEN &182 : Vertical%=FALSE: PROCC
onfine: PROCSetIcons
570 WHEN &183 : Confined%=NOT Confined
%: PROCConfine: PROCSetIcons
580 WHEN &18C : MOUSE TO x%-2,y%
590 WHEN &18D : MOUSE TO x%+2,y%
600 WHEN &18E : MOUSE TO x%,y%-2
610 WHEN &18F : MOUSE TO x%,y%+2
620 WHEN &19C : MOUSE TO x%-16,y%
630 WHEN &19D : MOUSE TO x%+16,y%
640 WHEN &19E : MOUSE TO x%,y%-16
650 WHEN &19F : MOUSE TO x%,y%+16
660 OTHERWISE SYS "Wimp_ProcessKey",Ke
y%
670 ENDCASE
680 ENDPROC
690 :
700 DEF PROCReceive(Num%)
710 IF Num%=0 THEN PROCAllOver
720 ENDPROC
730 :
740 DEF PROCAllOver
```

```
 750 Confined%=FALSE
 760 PROCConfine
 770 SYS "Wimp_CloseDown"
 780 END
 790 :
 800 DEF PROCSetIcons
 810 IF Vertical% THEN
 820 PROCSetIconState(0,2^21)
 830 PROCSetIconState(1,0)
 840 ELSE
 850 PROCSetIconState(1,2^21)
 860 PROCSetIconState(0,0)
 870 ENDIF
 880 IF Confined% THEN PROCSetIconState
(2,2^21) ELSE PROCSetIconState(2,0)
 890 ENDPROC
 900 :
 910 DEF PROCSetIconState(icon%, eor%)
 920 !q%=Handle%
 930 q%!4=icon%
 940 q%!8=eor%
 950 q%!12=2^21
 960 SYS "Wimp_SetIconState",,q%
 970 ENDPROC
 980 :
 990 DEF PROCConfine
1000 IF Confined% THEN
1010 IF Vertical% THEN MOUSE STEP 0,Ste
p%
1020 IF NOT Vertical% THEN MOUSE STEP S
tep%,0
1030 ELSE
1040 MOUSE STEP Step%,Step%
1050 ENDIF
1060 ENDPROC
1070 :
1080 DEF PROCOpenWindow
1090 PROCCreateWindow
1100 PROCCreateIcons
1110 q%!0=Handle%
1120 SYS "Wimp_GetWindowState",0,q%
1130 SYS "Wimp_OpenWindow",0,q%
1140 ENDPROC
1150 :
1160 DEF PROCCreateWindow
1170 LOCAL loop%
1180 FOR loop%=q% TO q%+84 STEP 4
1190 !loop%=FNGetNextNumber
1200 NEXT
1210 SYS "Wimp_CreateWindow",0,q% TO Ha
ndle%
```

```
1220 ENDPROC
1230 :
1240 DATA 2,812,392,980,0,-128,-1,&8703
1012,&1070207,&20E0C,0,-1024,1280,0,61,0
,1,0,&73756F4D,&72542065,&D7061,0
1250 :
1260 DEF PROCCreateIcons
1270 LOCAL pointer%,temp$
1280 pointer%=Indirect%
1290 FOR loop%=0 TO 5
1300 !q%=Handle%
1310 q%!4 =FNGetNextNumber
1320 q%!8 =FNGetNextNumber
1330 q%!12=FNGetNextNumber
1340 q%!16=FNGetNextNumber
1350 q%!20=FNGetNextNumber
1360 READ temp$
1370 q%!24=pointer%
1380 pointer%=FNStoreString(temp$,point
er%)
1390 q%!28=pointer%
1400 q%!32=LEN(temp$)+1
1410 READ temp$
1420 pointer%=FNStoreString(temp$,point
er%)
1430 SYS "Wimp_CreateIcon",,q%
1440 NEXT
1450 ENDPROC
1460 :
1470 DATA 10,-180,48,-132,&700311B,"","
Sradiooff,radioon"
1480 DATA 10,-232,48,-184,&700311B,"","
Sradiooff,radioon"
1490 DATA 11,-284,50,-240,&700311B,"","
Sno,yes"
1500 DATA 54,-188,354,-132,&C7000119,"V
ertical only (f1)",""
1510 DATA 54,-236,386,-188,&C7000119,"H
orizontal only (f2)",""
1520 DATA 54,-280,274,-248,&C7000119,"C
onfined (f3)",""
1530 :
1540 DEF FNStoreString(s$,s%)
1550 $s%=s$
1560 =s%+LEN(s$)+1
1570 :
1580 DEFFNGetNextNumber
1590 LOCAL a$
1600 READ a$
1610 =EVAL(a$)
```

RU