Subject: testcase
Posted by Ted Cary on Wed, 03 Apr 2002 22:56:24 GMT
View Forum Message <> Reply to Message

Since my original posting I've discovered that the GET_DRAW_VIEW keyword to widget control will return the viewport coordinates and that SET_DRAW_VIEW will set them, so I've answered some of my own questions. But using GET_DRAW_VIEW has only proven that the X field of the draw widget expose event structure has nothing to do with the x-coordinate of the viewport, at least not on my computer.

At the end of this posting is some code which demonstrates the behavior I'm talking about. The program is called App_Scroll_Testcase, although the problem may occur on all scrollable draw widgets, not just those using APP_SCROLL.

The program creates a resizable scrollable draw widget. With every expose event, it prints three values to the command line for comparison purposes:

(1) EVENT.X: the expose event structure's X field
(2) VIEWPORT X : the viewport x-coordinate as returned in the GET_DRAW_VIEW keyword to WIDGET_CONTROL.
(3) SCR_YSIZE: the draw widget's y screen size as returned using the GEOMETRY switch with WIDGET_INFO.

Just run the program, resize the window a few times, and watch the command line. If your computer is like mine, TLB_SIZE events will always generate WIDGET_DRAW expose events, and the X field in the expose event structures will not depend on the viewport position at all, but will always be equal to the draw widget's screen y size. What am I doing wrong?

Program listing below.

```
PRO App_Scroll_TestCase_Event, ev

CASE Tag_Names(ev, /Structure_Name) OF

   'WIDGET_BASE' : BEGIN
      Print, 'TLB_SIZE_EVENT!!__________'
      Widget_Control, ev.top, Get_UValue=info
      Widget_Control, ev.top, Update=0 ;
      Widget_Control, info.drawID, $
         Scr_XSize=ev.x, Scr_YSize=ev.y
      Widget_Control, ev.top, /Update
         ; Turning screen updates off during resizing
         ; eliminates ugly leftover pieces of the old
         ; draw widget that otherwise shown up on my Mac.
   ENDCASE

   'WIDGET_DRAW' : BEGIN
      CASE ev.type OF
         3 : BEGIN
           Print, 'VIEWPORT MOTION EVENT!!__________'
         ENDCASE
         4 : BEGIN
           Print, 'EXPOSE EVENT!!__________'
           Widget_Control, ev.id, Get_Draw_View = drawView
           drawGeo = Widget_Info(ev.id, /Geometry)
           Print, 'EVENT.X:  ', ev.x
           Print, 'VIEWPORT X: ', drawView[0]
           Print, 'SCR_YSIZE: ', drawGeo.scr_ysize
         ENDCASE
         ELSE :
      ENDCASE
   ENDCASE
   ELSE:
ENDCASE
END;---------------------------------------------


PRO App_Scroll_Testcase

tlbID = Widget_Base(/TLB_Size_Events)
drawID = Widget_Draw( $
   tlbID, $
   /App_Scroll, $
   X_Scroll_Size = 400, $
   Y_Scroll_Size = 400, $
   XSize=1500, $
   YSize=1500 $
   )
info = {drawID : drawID}
Widget_Control, tlbID, Set_UValue=info

Widget_Control, tlbID ,/Realize

XManager,'app_scroll_testcase', tlbID, /No_Block
END;---------------------------------------------
```