
Subject: Re: Problem with sensitivity of a Draw widget (also in XROI.pro)

Posted by [David Fanning](#) on Mon, 08 Mar 2004 14:57:36 GMT

[View Forum Message](#) <> [Reply to Message](#)

Ahammer Helmut writes:

> I'm working on an interactive widget with several buttons, text and a
> draw widget for image display, scrolling and curser information. The
> last thing I implemented was the curser information (with
> keyword/motion_event). Now the curser position and the grey value of
> the image are displayed if the curser moves over the image. But now
> there emerged a problem I had not before. If the scrolling bar is
> moved with the mouse and after the mouse button is released and if the
> mouse thereafter is moved instantly over the image display, then the
> scrolling event is not finished readily. The viewport is not scrolled
> instantly. The viewport is scrolled step by step and the whole scroll
> event need much more time (up to 10-20 seconds). It seems that the
> curser motion events are handled before the scroll event has finished.
> I used the sensitive keyword, but it didn't help.
> Furthermore, XROI.pro does the same. I`m using IDL 6.0 with W2k

With XROI I can stop this from happening by turning draw widget MOTION events OFF while I process the viewport event and then turning them back on. Here is the code I changed:

```
pro xroi__Viewport, sEvent

    compile_opt idl2, hidden

    ; Handle viewport move (scroll) event in the draw area.

    WIDGET_CONTROL, sEvent.top, GET_UVALUE=pState
Widget_control>(*pState).wDraw, Draw_Motion_Events=0, /Clear_Events
;~~~~~
    draw_geom = WIDGET_INFO((*pState).wDraw, /GEOM)

    ; On Motif, the geom.xsize may include room for scrollbar even if
    ; no scrollbar present. Restrict size to <= to the virtual canvas.
    draw_xsize = draw_geom.xsize < draw_geom.draw_xsize
    draw_ysize = draw_geom.ysize < draw_geom.draw_ysize

    (*pState).oView->SetProperty, $
        VIEWPLANE_RECT=[sEvent.x, sEvent.y, draw_xsize, draw_ysize]

    ; If we know our draw time is long, then set the hourglass.
    if ((*pState).draw_time gt 0.1) then $
        WIDGET_CONTROL, /HOURLASS
```

```
(*pState).oWindow->Draw, (*pState).oView  
Widget_control, (*pState).wDraw, Draw_Motion_Events=1, /Clear_Events  
;~~~~~  
end
```

Cheers,

David

--

David Fanning, Ph.D.

Fanning Software Consulting

Coyote's Guide to IDL Programming: <http://www.dfanning.com/>
