Subject: Re: function graphics event handling
Posted by Helder Marchetto on Mon, 02 Mar 2015 21:44:31 GMT
View Forum Message <> Reply to Message

On Monday, March 2, 2015 at 10:30:58 PM UTC+1, Helder wrote:

> Hi,

> I find my self often a bit puzzled by how the fg event handling works. Events such as mouse pressed or so, are handled with functions that with a call to "return, 1" continue the processing by the fg.

> But what if I would want do insert some code after the processing of the object?
>
> Let me make a demo example, first in words further down as code.

> Say you have a widget_window with an image and an ellipse inside. The window event mouse_down_handler calls a function called MouseDownEvent. I would like to know if the user clicked on the image or on the ellipse. For that I have the getSelect() function. However, this will give you the previous object selected... the new one is selected after the "return,1" is called.
>
> So here is an example:
>

```
>  function MouseDownEvent, oWin, xPos, yPos, Button, KeyMods, Clicks
>  ;here I would like to know which object has been selected
>  help, oWin->getSelect()
>  ;but this will always be the last one selected!!!
>  return, 1
>  end
>
>  pro testGetSelect
>  img = dist(500)
>  tlb = widget_base(/column)
>  wWindow = widget_window(tlb, xsize=500, ysize=500,mouse_down_handler
='MouseDownEvent')
>  widget_control, tlb, /realize
>  widget_control, wWindow, get_value=oWin
>  io = image(img, image_dimensions=[500,500], current=oWin, margin=0)
>  ;make some objects on top
>  el   = ellipse(0.25,0.75, major=0.125, '-r2', fill_background=0, /normal, target=io)
>  sq   = polygon([0.2,0.3,0.3,0.2], [0.2,0.2,0.3,0.3], '-y2', fill_background=0, /normal, target=io)
>  rect = polygon([0.7,0.8,0.8,0.7], [0.2,0.2,0.4,0.4], '-b2', fill_background=0, /normal, target=io)
>  end
>
```

> I will make my final question in the next few lines, but first I will say that one can use hitTest() to check which element "could" be selected.

> However, hitTest() seems to be non-consistent with getSelect(). When I have three objects intersecting, it's not clear which will be returned by hit test and which will be returned by getSelect(). I had an object selected (corners highlighted) that was the second in the array of three returned from hitTest(). Figure that.

> To check the effect of hit set, substitute or add in MouseDownEvent function these lines:

```
>  res = oWin->hitTest(xPos, yPos)
```

> for i=0,n_elements(res)-1 do help, res[i]
>
> And here is the question: Why not giving the possibility to act/interact *after* fg processing?
>
> Is this a goofy idea?
>
> Thanks,
> Helder

Ok, maybe for such things I should just use the mouseUp instead of mouseDown.

...mmm...

Helder