
Subject: Re: Locking graphics in GUI / disable resize, rotate or translate

Posted by [PMan](#) on Tue, 10 Mar 2015 15:00:41 GMT

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

On Tuesday, March 10, 2015 at 10:03:13 AM UTC-4, Paul Mallas wrote:

> On Tuesday, March 10, 2015 at 9:42:52 AM UTC-4, Paul Mallas wrote:

>> On Monday, March 9, 2015 at 5:40:31 PM UTC-4, Chris Torrence wrote:

>>> On Monday, March 9, 2015 at 1:09:07 PM UTC-6, Paul Mallas wrote:

>>>> Hi Chris,

>>>>

>>>> I have been working creating a class for handling events. Things are moving along slowly but I think I can do all the stuff I need with this. But I have one question - I am trying to override the selectChange method. How do I change the selection of the graphic directly? I can't do graphic.select (or graphic.select, /clear) since this calls the method I am trying to override and I end up with an infinite loop. Does this make any sense?

>>>>

>>>> Thanks,

>>>> Paul

>>>

>>> Would it be possible to have some sort of state variable in your class, like "self.selecting=1"? Then call graphic.Select, and check for this variable within your handler, and then return,1 if that were true (after turning the variable back off).

>>> -Chris

>>

>> I tried something similar to this - managing the selection at a higher level (sort of above where IDL considers a graphic selected). But I had no luck. But let me take another look. Thanks.

>

> Part of problem here is that for one 'select' event, my selectChange method gets called twice. Seems to be a bug (I reported this to the support folks) or perhaps some shortfall in my understanding.

Here is my barest bones example. Run the code below, the graphics name gets printed twice per one click:

```
FUNCTION ExWid2Win::Init
```

```
    self.select = 0
```

```
    return, 1
```

```
END
```

```
FUNCTION ExWid2Win::SelectChange, oWin, graphic, mode, wasSelected
```

```
    print, graphic.name
```

```
    return, 0
```

```
END
```

PRO ExWid2Win__define

```
void = {ExWid2Win, $  
  inherits GraphicsEventAdapter, $  
  select: 0L}
```

END

PRO ExWidget2WindowEvents_event, event

```
w = WIDGET_EVENT(/NOWAIT)
```

```
print, 'do nothing'
```

END

PRO ExWidget2WindowEvents

```
wBase = WIDGET_BASE(/COLUMN, /TLB_RESIZE_NODRAW, MAP=0)
```

```
wDraw = WIDGET_WINDOW(wBase)
```

```
WIDGET_CONTROL, wBase, /REALIZE
```

```
WIDGET_CONTROL, wDraw, GET_VALUE=win
```

```
win.Select
```

```
p = PLOT(/TEST, /CURRENT)
```

```
handler = OBJ_NEW('ExWid2Win')
```

```
win.EVENT_HANDLER = handler
```

```
WIDGET_CONTROL, wBase, /MAP
```

```
XMANAGER, 'ExWidget2WindowEvents', wBase, /NO_BLOCK
```

END
