
Subject: WIDGET_DRAW event.press/release values for mouse wheel events..undocumented?

Posted by [Paul Van Delst\[1\]](#) on Thu, 16 Aug 2007 14:02:32 GMT

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

Hi All,

IDL> print, !version

{ x86 linux unix linux 6.2 Jun 20 2005 32 64}

I have the following widget_draw setup:

```
dw_id = widget_draw( tlb_id, $
                    /button_events, $
                    event_pro = 'wplot_draw_event', $
                    xsize = dw_xsize, $
                    ysize = dw_ysize )
```

When I press/release the left/middle/right button I get the usual:

	press/release value
left	1
middle	2
right	4

Now, I have one of those mice where the middle "button" is also a wheel. Being a fat-fingered type, sometimes when I press the middle button in my apps, I sometimes also move the wheel up or down. For those cases the event.press/release values are:

	press/release value
roll up	8
roll down	16

and the event.type values are still 0 or 1 (for mouse button press and release) even though I'm not depressing the wheel/button (event.clicks always remains 0)

Is this behaviour documented anywhere? The wheel_events keyword to widget_draw is not set (and it's ignored for non-windows machines anyway). I googled the clip archives but didn't find anything. The widget_draw docs didn't

Is this just a poor mans method of emulating wheel events for non-windows systems?

cheers,

paulv

Subject: Re: WIDGET_DRAW event.press/release values for mouse wheel events..undocumented?

Posted by [Paul Van Delst\[1\]](#) on Thu, 16 Aug 2007 20:09:51 GMT

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

Richard Clark wrote:

> In version 6.4 I think you need to explicitly enable wheel events to
> have them passed through to your event handler.
>
> In my case the wheel was just a source of noise to be ignored. If you
> will be using the wheel then you may also need to guard against
> accidental button presses on the middle button when you meant the
> wheel:-) But then I think the TYPE tag on the event structure may have
> a distinct value for wheel events too... I was only interested in
> silencing the thing.

That was my initial need too. But now I'm wondering if I can (should? :o) use the wheel to enable some sort of centred in-and-out zooming of my current data plot (e.g. like in google maps).

My particular app doesn't (yet) use the middle button, just the Left (click and drag zoombox) and Right (return to original zoom) ones.

cheers,

paulv

>
> Richard
>
> In article <fa253r\$8er\$1@news.nems.noaa.gov>,
> Paul van Delst <Paul.vanDelst@noaa.gov> wrote:
>> Richard Clark wrote:
>>> Yes, I have encountered this too. We use intel linux systems with pc
>>> type wheel mice.
>>>
>>> Neither the bu... I mean feature, nor its fix (in version 6.4) seems
>>> to be documented.
>>>
>>> After checking event.type for UP, DOWN, or MOTION mouse events, check
>>> for PRESS or RELEASE values >4.
>>>
>>> Here is a code snippet from a heavily modified version of Fanning's
>>> DRAWBOX_WIDGET example procedure that I use to get around it.
>>> It looks like you've already found the same fix.
>> Similar. I now use the following:
>>
>> FUNCTION WPLOT_Which_Button, Value
>> CASE Value OF

```
>> 1: e='LEFT'
>> 2: e='MIDDLE'
>> 4: e='RIGHT'
>> 8: e='UP'
>> 16: e='DOWN'
>> ELSE: e='NONE'
>> ENDCASE
>> RETURN, e
>> END
>>
>> and call it thusly:
>>
>> IF ( WPLOT_Which_Button(event.RELEASE) EQ 'LEFT' ) THEN BEGIN
>> ....
>>
>>
>> I decided to use the CASE statement since one can't rely on the 8 and 16 values remaining
>> valid (and apparently, in v6.4, they're not)
>>
>> cheers,
>>
>> paulv
>
>
```
