
Subject: Re: catching of wheel button / wheel scroll events?

Posted by [Jeff Guerber](#) on Fri, 16 Apr 2004 22:14:01 GMT

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

> "Stefan Tuchschnid" wrote in message...

>> Hi Guys

>> For a medical application the client requested that he can scroll
>> through a stack of images with a Logitech Wheel Mouse. Only one is
>> shown at a given time. That means I'll have to come up with an
>> solution to catch the wheel event. I have not found any documentation
>> how this can be done in IDL.

>>

>> It might be possible to have an empty IDLgrWindow with scroll bars
>> invisible in back of my Draw Object, and then catch the OnScroll
>> Events. However, this seems to be a rather awkward solution.

>>

>> I know that there's a OS issue with that and all, but we're simply
>> trying to make it as fast as possible for the radiologist.

>>

>> Solution anyone?

X-Windows has something of a "de-facto" standard that rotation of the mouse wheel generates Mouse-4 and Mouse-5 events, as if the mouse had two extra buttons. I just tried it in IDL 6.0.3 with a draw widget set to return button events, and upward rotation generated {widget_draw} events with event.press (or release) = 8; downward, 16. (The press and release fields are bit flags, one bit per button.) The events came in pairs, the first with event.press=8 (or 16) and event.release=0, then vice versa. This was using IDL 6.0.3 on HP-UX and X-Win32 5.4.4 on MS-Windows 2000Pro.

Note that not every X server supports mouse wheel events, however. On PC's, I know StarNet X-Win32 does, but Hummingbird Exceed does not (or at least, not the last time I tried it). Also, you may need to fiddle with the X server's configuration settings to turn them on.

For IDL running directly under MS-Windows, I can't say.

Jeff Guerber
