
Subject: Re: xmanager, /managed

Posted by [Benjamin Hornberger](#) on Thu, 12 May 2005 14:57:55 GMT

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

David Fanning wrote:

> Benjamin Hornberger writes:

>

>

>> I am trying this out, but I can't get it to work. I tried that line of
>> code right before and right after the XMANAGER call, and right after the
>> TLB was created. None of them worked.

>

>

> Humm. I wonder if this is a Windows thing?

>

> The code I use to test this with is XCOLORS:

>

> <http://www.dfanning.com/programs/xcolors.pro>

>

> The "managed" line is line 1104. Comment it out.

> Then make sure there are no open windows in your IDL
> session. Run XCOLORS. While it is on the display, type
> this:

>

> Plot, findgen(11)

>

> The plot should show up in the XColors colorbar window.

>

> Exit XColors and uncomment the managed line. Compile and run
> it again, as before. This time the PLOT command should open
> its own plot window.

>

Now I can confirm this behaviour on IDL 6.1.1 on Windows. But you have to mention that you store the original !d.window at the beginning of your widget definition procedure and restore it after the "widget_control, /managed" call. Maybe you should link to xcolors.pro as example code.

It seems to work in my program as well now, even though I don't really understand it. If I store my original !d.window and reset it at the end of my widget definition procedure, and do the same in each event handler which plots something, I would logically expect that it works as described even without the "widget_control, /managed". But it doesn't.

>

>> Also, does it mean when I finally want to plot something in my draw
>> widget, I first have to save the content of !d.window, WSET to the draw

>> widget, plot my stuff and then WSET back to the original value?
>
>
> Well, I figure anyone who wants to draw into a window *always*
> knows where they are drawing, so I don't ordinarily WSET back
> to the original. Every man for himself, is my motto. But in a
> widget program you must ALWAYS do a WSET to the window you
> expect to draw into, yes.
>

Well, if you follow the "every man for himself" motto, what are we talking about then anyway? You wouldn't need the whole thing at all. I agree that every widget program should take care of itself, but if you work on the command line while a non-blocking widget program is open, it's kind of annoying if the widget program messes up the system variables, color table etc. all the time (I am following your advice to restore the display each time the widget program gets the keyboard focus). That's why I found your article very useful, and I'm thinking of a strategy to do the same with color vectors.

Cheers,
Benjamin
