
Subject: Re: xmanager, /managed
Posted by [David Fanning](#) on Fri, 08 Jul 2005 06:27:52 GMT
[View Forum Message](#) <> [Reply to Message](#)

Benjamin Hornberger writes:

- > Two months later, I want to report another experience with this
- > issue. By the way, David seems to have removed the article from his
- > website --he doesn't seem to like the thing any more :-).

With the exception of articles written by JD, I try not to publish articles on my web page that I don't understand. People have the most embarrassing habit of asking questions... :-(

- > Even though I don't really understand why (see the previous messages
- > in this thread), the "widget_control, /managed" did solve one issue I found
- > quite disturbing before: If you have a non-blocking widget program
- > with a draw widget on the display, but then want to plot something
- > from the command line, it will be plotted in the draw widget unless
- > you manually open a new plot window first. With the statement above
- > (see code attached), a new plot window pops up automatically as
- > desired.

I'll have to explain it to you in a private e-mail (I wouldn't *think* of risking it here), but that's why I use it, too.

- > However, by pure chance I discovered now that this statement caused
- > another problem I had been chasing for weeks now: I am using a GUI
- > program with a draw widget to display images. The user can mark a
- > rectangle in the image by dragging the mouse with the left mouse
- > button held down. On Windows, everything worked fine, but on Linux
- > the rectangle was not shown while dragging, only the final rectangle
- > appeared after releasing the left mouse button. After commenting out
- > the "widget_control, tlb_id, /managed" line, it works in Linux as
- > well.

Humm. I think I saw this behavior, too, just shortly after I released that Catalyst save file for public consumption. I think I solved the problem in another way, but I can't think now how I did it. Smoke and mirrors and pixmaps, I expect.

- > Attached is a sample program with the relevant code extracted from my
- > real application, for the ones who care. It requires David's
- > fsc_color(). Try to drag a rectangle in the image. On Windows,
- > everything should be fine, but on Linux the rectangle doesn't show up
- > while dragging (at least that's how it behaves for me). Commenting
- > out the "widget_control, /managed" near the end solves this (but
- > introduces the problem mentioned in the beginning).

Yes, well, I'll have Karsten chase this problem down. He has his brand new Mac configured as a Mac, a PC, a UNIX machine, a German Umpah Band, and I don't know what else! If he can't get to the bottom of this, no one can. :-)

- > I should mention that besides keeping track of colors in the GUI
- > program, as David recommends in his book, I also try not to mess with
- > the command line's color vectors and decomposition state. Even while
- > the widget program is running, I want to be able to quickly tvscl
- > another array from the command line with a nice b-w color table
- > rather than the GUI's color table with some plot line colors loaded
- > at the top. Unfortunately, the code becomes quite ugly. Maybe object
- > graphics is the way to go, but I haven't had the time to learn that.

Object graphics would certainly solve your problem (as well as putting your marriage at risk with the months you will spend learning it), but it seems like overkill to me. I think your main problem is all those damn TV commands. That is what is causing you so much pain. You have to live in an 8-bit environment in a 24-bit world. I would set that DECOMPOSED keyword permanently to 1 and get a sensible replacement for TV (TVIMAGE, TVSCALE, IMGDISP, PLOTIMAGE). I've lived for nearly two years now neither knowing or caring much about my color decomposition state. (Except for filled contour plots, of course. Sigh...)

My advice is to write your programs so they are totally indifferent to the DECOMPOSED keyword. Life is *much* simpler that way. :-)

Cheers,

David

P.S. And if you *are* forced to go the object route (always a good idea, in my opinion), direct graphics objects are a LOT simpler to learn than object graphics and are more useful in a lot of circumstances.