Subject: Re: Coyote graphics and resizeable draw widgets
Posted by wlandsman on Fri, 23 Mar 2012 15:52:34 GMT
View Forum Message <> Reply to Message

On Friday, March 23, 2012 10:34:29 AM UTC-4, David Fanning wrote:

> If your interface is resized, you determine what
> size your graphics window should be, and you call
> the Resize method with the new sizes.
>
> windowObj -> Resize, event.x, event.y

Enlightenment comes slowly, Sensei.

I assume that I still need to compute the new size of the graphics window, i.e. if there are other buttons on the widget, I need to get their sizes and subtract them from the new size reported by TLB_GET_SIZE keyword for the resizing event.

Where cgcmdwindow does save me (a lot) is having to redraw the plot (and keep track of the data needed to redraw the plot) when the window is resized.

So instead of my current widget_draw() call, I create a cgcmdwindow object with the id of the parent widget. I currently use a single giant event handler, that distributes to the method name and object reference stored in a UVALUE. Its not yet clear to me how to store a UVALUE (that can be fetched with a GET_UVALUE) or even how to get the widget draw ID with the cgcmdwindow object, though there are things I can try.

Another thing that cgcmdwindow might help with are graphics overlays. For example, I want a circle to appear (only) at the position where the user has clicked with his cursor. Currently, I keep a separate pixmap window of the plot which gets copied back (to erase any old circles), and then a circle drawn at the cursor position. (The pixmap window also needs to be resized when the widget is resized.) But now I could use a DeleteCommand method for the old circle, and an AddCommand method for the new circle. (This might be slower since the plot would be redrawn rather than copied.)

Thanks, --Wayne