
Subject: Re: explicit redraw does nothing until expose(?) event - IDLitComponent
propertysheets

Posted by [David Fanning](#) on Fri, 23 Apr 2004 03:30:40 GMT

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

Sean Dettrick writes:

> The handle_all_events routine inspects the UVALUE of the widget
> causing the event, and uses CALL_METHOD to call my ReDraw method on
> the object (David Fanning credits Stein Vidar for this idea)

Really!? I guess I thought I had "invented" it myself.
Maybe I re-discovered it independently after Stein Vidar
showed it to me and I forgot all about it. :-(

> I can't see why that should change anything. But could CALL_METHOD
> somehow slow things down?

Well, probably. But what you are seeing isn't a slow
down, it is a screeching halt.

>> Why not simply add a win->draw at the end of your property sheet event
>> handler? You know that if a property changes, you want to redraw.

>>

>

> Well I tried that too. It doesn't change the behaviour.

Wait a minute. You called the method *directly* from
the event handler module you *know* you are in, and
it didn't change anything!? Now, that is truly strange.

It looks to me as if something is blocking events.
I can't imagine it is the PropertySheet, since I've
managed to interact with PropertySheets before and
I get updates directly and immediately. And there
would be no reason for a property sheet to block,
I don't think.

> I think that
> for some reason the events are just not making it to the event handler
> until I interact with the draw window. Could they be trying to go
> somewhere else, e.g. into a non-existent iTool?

In RSI supplied code!? No, I don't think so. Now, if
you thought there was an error that was being handled
silently...well, let's just say I wouldn't dismiss it
out of hand.

- > It's a little frustrating but I haven't a clue on how to resolve it.
- > Any other ideas?

Not really, but I have risked \$10 in favor of a programmer error rather than a software bug. :-)

There is always a danger in programming object-widgets that events are not occurring in exactly the way you *think* they are occurring. In fact, programming widget-objects is a good way to find yourself embracing a mystical religious tradition. You might want to step through your program to figure out where you are coming from and going to when you pop up that dialog. I have occasionally found myself at the end of a procedure I thought I had exited eons ago. Something like this might be occurring. (Fortunately, we built a "verbose" switch into the 2nd version of the Catalyst Library, so it can show us graphically which program modules we enter and leave. This has saved us from just this kind of frustration several times.)

Do you happen to be in Boulder, Sean? I'm not sure why I think that, but I'm going to be up that way tomorrow. I'd be curious to see what this looked like in real life.

Cheers,

David

--

David Fanning, Ph.D.
Fanning Software Consulting
Coyote's Guide to IDL Programming: <http://www.dfanning.com/>
