Subject: Re: General widget programming questions
Posted by Pavel A. Romashkin on Fri, 12 Jan 2001 16:55:55 GMT
View Forum Message <> Reply to Message

I will just say that in those few lines of code I ever wrote, I never
had a problem with events queue slowing me down. On today's computers,
anyway. I guess, I can't click fast enough - about 120 clicks per minute
is all I can dish out, and the CPU apparently keeps up at 500 MHz. The
only slowdown I ever see is the graphics card, and not when I have tons
of events but when those OG 3D transforms are performed on more than
100K objects at the same time. Then I can see performance degradation.
But I have not had to short-cut the old Xmanager. Although I want to
read it out of curiosity :-)

Cheers,
Pavel

P.S. Let's just say I am not advanced enough. David, you'll have to
expand your widget and object sections do move me forward and develop
the need in direct calls to event handlers :-)

David Fanning wrote:
>
> Jason P. Meyers (jpm7934@cis.rit.edu) writes:
>
>> We have recently started writing some widget based programs in our
>> IDL programming class and it has gotten me to thinking about a few
>> issues I would like to better understand before I get too far into our
>> "final" project. For starters, I have read the relevant chapters in
>> Dave Fanning's second edition. So far, that has been the best help in
>> understanding widget based programming. (I have even used some of
>> Dave's info to correct misinformation presented in class!)
>
> Be careful with that book. It has been known
> to turn around and bite you when you least
> expect it. :-(
>
>> I like the ability to create user defined event structures. In my
>> current homework project, I have started using these to pass information
>> from one widget to another. I basically pack the information I want to
>> send to another widget in an event structure and give it a descriptive
>> name and then send it along to the appropriate widget. So far, this
>> seems like a reasonable thing to do. However, I got to thinking about
>> potential problems. First, what happens if other events (i.e. user
>> generated) are waiting in the queue? I assume they will get acted on
>> first and this may or may not cause a problem for the original event
>> handler which sent an event to a fellow widget. If this is the case, is
>> there any way to give a particular event a higher priority over other

>> events?
>
> There is no way to give events priority. *All* events generated
> normally, or sent via SEND_EVENT, are placed on the queue and
> are handled in the order received. This is ordinarily a good
> thing, and is especially advantageous to people just getting
> started in widget programming, because it prevents an awful
> lot of problems. For example, there is never any problem
> with the info structure not being where it is suppose to be.
> If each event handler checks it out, then checks it in before
> exiting, then each event can find it and be assured the information
> in it is current.
>
> But as you get more advanced in widget programming it sooner
> or later occurs to you that with the big bucks you are being
> paid you ought to be a little more clever than to be a slave
> to SEND_EVENT. And you think to yourself, "My God, I'm just
> going to call that event handler directly. After all, it is
> nothing more than another IDL procedure or function."
>
> And you can do that. I've done it. Lot's of times. But it's
> kind of like putting the short side of the board against the
> fence on the table saw. You better be damn careful you know
> what you are doing. And for goodness sake, don't stand directly
> behind the saw!
>
> By calling the event handler directly, you obviously short-circuit
> the event queue. Sometimes this means you have to check the info
> structure in before you make the call. Sometimes you have to do
> other things. (When things get really complicated, you might want
> to put the info structure in a pointer and pass that around,
> since then you don't have to worry about checking out/checking in.)
>
> But I will say that it is quite easy to get too cute with
> all this. I think the simpler you can make event handling,
> the better off you are. Good ol' XManager, one-at-a-time
> event handling is good enough for 99% of your applications,
> I think.
>
>> I haven't seen a good example or even an explanation of how and/or why
>> it would be useful to use the Event.Handler value which is stored in all
>> event structures. I thought that I read somewhere in Dave's book that
>> he was going to address this but, I didn't see any references to using
>> it in the various chapters on widget programming. (Dave, am I blind or
>> did I get confused with something I may have scanned in the IDL online
>> references?)
>
> The handler field is used quite a lot in building compound

> widgets. I intended to add a chapter on compound widgets in
> the book, but I got so sick and tired of reading the darn thing
> that my will collapsed before I could get it written. It's still
> on my list for the "next" book. :-)
>
>> As always, I appreciate any and all insight people might have. Thanks
>> in advance for advice/answers that come flowing. Finally, if Dave (or
>> anyone else) has recommendations for additional high quality
>> educational/tutorial information that picks up where Dave leaves off in
>> his book, please let us all know.
>
> Ronn Kling has a nice book out that explains quite a number
> of useful widget and programming techniques. You might have
> a look at his web page:
>
> http://www.rkling.com/
>
> Cheers,
>
> David
> --
> David Fanning, Ph.D.
> Fanning Software Consulting
> Phone: 970-221-0438 E-Mail: davidf@dfanning.com
> Coyote's Guide to IDL Programming: http://www.dfanning.com/
> Toll-Free IDL Book Orders: 1-888-461-0155
