
Subject: Re: Q on widget programming style
Posted by [davidf](#) on Tue, 28 Jul 1998 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Mirko Vukovic (mvukovic@my-dejanews.com) writes:

- > In my widget programming I tend to assign an event handler to each widget
- > (well, almost), since that eliminates testing on the origin of the event,
- > and produces somewhat cleaner code.
- >
- > But are there any pitfalls to that practice?

Uh, it produces a lot of code. :-)

I tend, these days, to use a lot of event handlers too, but I tend to group them by logical function or utility. For example, all of the buttons in the "Processing" pull-down menu will be handled by the same event handler. Or all the "color tools" will be handled by the same event handler, etc.

This tends to cut down on the amount of code a little bit and makes it easier to find the part of the code I want to work on.

I'm also very big these days on writing compound widgets as objects. Normally this will mean a single event handler, but many methods for the compound widget. Maybe it's my imagination, but it seems very elegant to me. :-)

Cheers,

David

--

David Fanning, Ph.D.
Fanning Software Consulting
E-Mail: davidf@dfanning.com
Phone: 970-221-0438
Coyote's Guide to IDL Programming: <http://www.dfanning.com/>

Subject: Re: Q on widget programming style
Posted by [mvukovic](#) on Wed, 29 Jul 1998 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

In article <MPG.1027c928665bc771989818@news.frii.com>,
davidf@dfanning.com (David Fanning) wrote:
> Mirko Vukovic (mvukovic@my-dejanews.com) writes:

>
 >> In my widget programming I tend to assign an event handler to each widget
 >> (well, almost), since that eliminates testing on the origin of the event,
 >> and produces somewhat cleaner code.
 >>
 >> But are there any pitfalls to that practice?
 >
 > Uh, it produces a lot of code. :-)
 >
 > I tend, these days, to use a lot of event handlers too, but
 > I tend to group them by logical function or utility. For example,
 > all of the buttons in the "Processing" pull-down menu will be
 > handled by the same event handler. Or all the "color tools" will
 > be handled by the same event handler, etc.

I should have been more explicit. This is my approach too. The main reason why I don't like to dump too many widgets into one event handler is the complexity of the testing which event occurred, and possible hassle if I want to add another widget, that I'll have to re-work the whole testing tree.

On the other hand, I could simply note all the widget ID's, and have a case statement parse which code to execute in the event handler.

I would still vote for the logical grouping (i.e., not the dump-in-one-event-handler approach noted in the previous paragraph), as variable naming and such is much easier.

> This tends to cut down on the amount of code a little bit
 > and makes it easier to find the part of the code I want to
 > work on.
 >
 > I'm also very big these days on writing compound widgets as
 > objects. Normally this will mean a single event handler, but
 > many methods for the compound widget. Maybe it's my imagination,
 > but it seems very elegant to me. :-)

hmm, a widget is an object, so why not. But can you expand on this a bit (please)?

mirko (I never thought I would be looking forward to 100F weather to cool-off. This comment comes after a couple of weeks in T>100F in beautiful and sunny Arizona)

-----== Posted via Deja News, The Leader in Internet Discussion ==-----
http://www.dejanews.com/rg_mkgrp.xp Create Your Own Free Member Forum