
Subject: Re: widgets and objects

Posted by [davidf](#) on Thu, 13 Aug 1998 07:00:00 GMT

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

Mirko Vukovic (mirko_vukovic@notes.mrc.sony.com) writes:

- > this morning, while I was [...] thinking [about] widgets, and what
- > is to me the biggest pain, i.e.
- > remembering information from one event to the other.

Indeed. But then I'm informed by recent reading that life itself is pain. Having just had another birthday in the midst of a week-long tennis tournament, I am prepared to believe it. :-)

- > The currently recommended procedure is to use a pointer to a structure which
- > contains all the pertinent info. The "aesthetic" problem is that
- > you need to recall the pointer from some UVALUE, and then access the contents.

> [Much stuff snipped...]

- > Now, the combination of widgets and objects has been mentioned but not
- > described by several folks.

Yes, I am working on an example in my "spare" time. :-)

- > Can one define a widget to be an object, and the
- > events to be methods? Then self is naturally defined, and at least I do not
- > have to worry about accessing it. The code does not get much cleaner, but at
- > least one level of pointer access is eliminated. However, it is not clear to
- > me that the event handlers can be methods. In that sense, one needs the
- > event handler to call the method, yet another layer of complexity.

Yes. You can define a widget program to be an object, although there are some limitations. One is, as you point out, that event handlers cannot currently be methods. To some extent this is handled by catching the event in a normal event handler and calling the event method from there. Not elegant, perhaps, but it works.

I find the greatest utility of widgets-as-objects to be in writing compound widgets. The problem with compound widgets is that it is always difficult to interact with them. For example, if a compound widget has 15 simple widgets, what does it mean to "set a value", which is the way you must communicate with a compound widget.

With a compound widget-as-object, you simply "get the value" of the compound widget, which is the self reference object,

and you can call any method you like to do anything you like.
Very slick and *very* useful.

I think RSI is aware that widgets as objects are desirable.
Probably if they were redesigning how widgets work that
capability would be built in. As it is now, it is a bit of
a retrofit that could be difficult I imagine. Having just
come off of a major project (object graphics) I can't imagine
there are too many engineers who are ready for another big
push just yet.

In the meantime, I am encouraged by the reports of object
programming reported here and in my own experiments with
it. It would be wonderful if we could start to collect some
of these objects into a library we could share.

Cheers,

David

--

David Fanning, Ph.D.
Fanning Software Consulting
E-Mail: davidf@dfanning.com
Phone: 970-221-0438, Toll Free Book Orders: 1-888-461-0155
Coyote's Guide to IDL Programming: <http://www.dfanning.com/>
