
Subject: Re: Object epiphany: A new way of building widget applications

Posted by [Ben Tupper](#) on Thu, 05 Apr 2001 14:11:03 GMT

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

Hello,

Martin Schultz wrote:

> Hi all,

>

> With almost a week delay, I finally get around to release the first
> version of a new class of IDL objects: the MGS_GUIObject hierarchy.

> Don't shy away immediately! It's far easier than it sounds, and once

> you will have discovered how easy it now becomes to develop widget

> applications, you will get hooked! Ben Tupper has managed to get

> something running within a day.

>

Now, I could take that as a compliment on my programming prowess or I could take that as testimony of the ease of basing widgets on MGS_GUIObject. I know my programming skills well enough to know that it is the latter.

I have migrated to using objects for all items that are GUIs or serve some form of a databasing function. Perhaps I have been an easy sell for Martin because I got hooked on his earlier MGS_CONTAINER object (I rarely use IDL_CONTAINER... and when I do, I always regret it and change it to Martin's subclass.) I have switched to object coding because the people I work for often change their minds about what is needed. The object style coding is perfect for this situation.

Key items for me (non-programmer/science-lackey/guinea pig) are:

1. It is way too easy. When I think of the hours I have spent building GUIs from scratch that could have been done in hours... ugh. How many times have I written the event handlers for GUIs to handle cleaning up the widget, sending events to the proper procedures/methods. It seemed like an nightmarish repetition that varied only slightly in the details. Here it is done once with a uniform nomenclature and a standard operating procedure.

2. I can treat it as a black box. I don't have to know how it does what it does. I don't need to fiddle-faddle with all of the MODAL, BLOCKING, CLEANUP, etc. knobs that make xMANAGER control the GUIs they way I expect (hope). All of those are handle by the base gui object behind the scenes.

3. I can open the black box and override any of the hidden workings (see

#2 above) with out changing the base object code. I can have a special case without building exception handling into the base (object) code. (#2 and #3 give me the best of both worlds!)

Object-based GUIs are worthy of a serious look by anyone who needs GUIs but hates to (re)write them. And while your at it, there is no point in reinventing the wheel, try MGS_GUIobject.

Ben

--

Ben Tupper
248 Lower Round Pond Road
POB 106
Bristol, ME 04539

Tel: (207) 563-1048
Email: PemaquidRiver@tidewater.net
