Subject: IDL Style (was Re: IDL 5.2 GUI Builder Tutorial ?)
Posted by fireman on Thu, 27 May 1999 07:00:00 GMT
View Forum Message <> Reply to Message

Mike Schienle (mgs@ivsoftware.com) wrote:
: dealing with a very disgusting usage of WIDED. Our tax dollars (in the
: U.S.) have been used to create this:
: base1 = WIDGET_BASE(base0, /ROW)
: label1 = WIDGET_LABEL(base1, VALUE = 'Input L0 file   ')
: gwid.ifile = WIDGET_TEXT(base1, /EDIT, XSIZE = 40, $
:   VALUE = '')
: gwid.select = WIDGET_BUTTON(base1, VALUE='Select')
etc.

Mike -

What exactly is so bad about this? Aside from the fact that the input
windows don't line up. It looks like my widget code - well, the last
I wrote anyhow, which was well before GUI Builder, before WIDED, and even
before compound widgets.

: ...let me suggest an IDL Style Guide that I put together earlier this year.
: <http://www.ivsoftware.com/IDL_Style.html>.

This style guide is enforceable within an organization, but there is
too much variation in personal style for the IDL community to adopt in
en masse. For example, I reject the idea that variable names should
start with a letter corresponding to data type.

--
-- Gwyn F. Fireman
-- General Sciences Corporation / MODIS Characterization Support Team
-- Gwyn.Fireman@gsfc.nasa.gov   301-352-2118

Subject: Re: IDL Style (was Re: IDL 5.2 GUI Builder Tutorial ?)
Posted by davidf on Tue, 01 Jun 1999 07:00:00 GMT
View Forum Message <> Reply to Message

Mike Schienle (mgs@ivsoftware.com) writes:

> I think David's planning on a book to handle that as soon as he finishes
> the one he's working on :-)

Heh, heh, heh!

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

Subject: Re: IDL Style (was Re: IDL 5.2 GUI Builder Tutorial ?)
Posted by mgs on Tue, 01 Jun 1999 07:00:00 GMT
View Forum Message <> Reply to Message

In article <375386C3.3338D770@mail.earth.monash.edu.au>, Michael Asten
<masten@mail.earth.monash.edu.au> wrote:

>>
>>>
>>> Mike -
>>>
>>> What exactly is so bad about this?
>>
>> Hi Gwyn -
>>
>> No clue what base0, base1, label1, etc. indicate, and they're repeated
...
>
> Hi guys,
> arent we confusing a couple of different issues here?

Who me? Confused? Nahh, couldn't be.

> Firstly, thanks to Mike S. for the earlier posting on the style guide. I like
> it. I tend to use a default fortran-style naming convention, but the
additional
> rigor suggested by Mike makes sense. In hindsight it would have saved me a few
> "what did I mean when I wrote this 3 months ago" experiences.

I wish I could have had the same hindsight sooner. It wasn't until about 4
years ago that I finally started getting the bigger picture. Of course I
resisted having that picture shoved down my throat by a couple managers
before I realized maybe they had a point. The Style Guide is an adaptation
of two different guides from two different companies, both originally
written with C in mind. Again, it's a pain to start using, but very
helpful in the long run.

> Secondly, regarding undocumented widget code generated by WIDED or
GUIBUILDER, I

> don't accept the argument that poor documentation is an argument
> against their use. Their use saves a great deal of programming time for
some of
> us,   and before we give away, (or archive for ourselves!) the code, it is
> commonsense professsional practice to add blocks of comment code. This will be
> "what the widget does" rather than individual lines of "how the code does it".
> Mikes angst with the busy person who failed to document before putting
code into
> the public domain, is justifiable, but keep in mind that a lot more tax
dollars
> might have been spent if the programmer had worked from a lower level.

This makes perfect sense when you're dealing with a simple interface (just
had an interesting typo/freudian slip: face turned into feca - I digress).
My experience has been that the interface builders fall short when it
comes to iterative changes to complex widgets. I haven't been part of a
major GUI development team for about 4 years now, so maybe things have
improved, but I doubt it. Each time we got part way through the project
there was a mad scramble when we hit the builder's limits and had to
figure out what the hell it was thinking. We fell back to hand-coding each
time and the irritation with the builder was very high.

You get lulled into a comfortable feeling, then everything falls apart.
The schedules get shredded when you have to backtrack that much. Projects
and morale go downhill. I feel sorry for the folks that have to deal with
the SeaDAS code on a daily basis, it has to be a nightmare.

> I believe that arguments against the use of guibuilders are akin to arguments
> against the use of 4GLs generally (amazing the number of professional
programmers
> who think that programming in any language higher-level than c++ is for wimps).
>
> One of the beauties of machine generated widget code, apart from initial time
> savings, is that it is predictable and systematic and requires less
> documentation than a block of code written by a person with a personal style.
> However one has to resist the temptation to hack the generated code to pieces.

Assuming the GUI code is doing what you really want. Again, simple GUI's
will likely run without a problem.

> I
> think the guibuilder is a good attempt in the right direction, since it
encourages
> separation of event handling code from gui code. I also find that the
> much-maligned WIDED is superior when it comes to making a gui to handle text
> fields (especially real mumbers) . I'd love to see rsi produce a clear
tutorial
> on how to get the most out of the guibuilder, to the same standard as say David

> Fannings tutorials on how to avoid it.

I think David's planning on a book to handle that as soon as he finishes
the one he's working on :-)

--
Mike Schienle   Interactive Visuals, Inc.
mgs@ivsoftware.com   Remote Sensing and Image Processing
http://www.ivsoftware.com/   Analysis and Application Development