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

Posted by [Martin Schultz](#) on Thu, 05 Apr 2001 08:43:54 GMT

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

JD Smith wrote:

>

>> 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. ...

>

> I think it only fair to let people know that I tend to shy away from
> distributed code with people's initials in the name. I know, it sounds
> stupid, but I'm not sure I'm the only one. It seems to be a reasonably
> common practice here (Craig, you listening?), but one which I think
> might be best to avoid, for the following reasons:

>

Well, to be honest, it makes me a little sad that there are several
articles in this thread, but none dealing with the software itself,
and all sidetracking on a somewhat futile discussion about initials
and namespace. (but thanks Craig for your email)

But I feel I need to take up this topic anyhow, and I disagree with
you, JD.

> 1. It conveys a sense of ownership or heavy expectations that are
> perhaps unjustified, and not intended. (Can I *change* such a routine,
> should I feel guilty, etc...).

I think you have the wrong feelings here. Mine are rather along the
lines of "I am using fsc_...; hey - that's a great piece of code, this
guy is worth remembering". And when I want to change things, I just
do, because he explicitly allows me to do so in his license. And then
I can rename it to mgs_... because then I know it's my version, and if
something goes wrong, I am to blame first.

>

> 2. It takes up space in a name which could perfectly well have been
> used for more descriptive characters.

I'm not sure of this. I think, Mark hit the ball here saying that
"descriptive" prefixes tend to produce namespace conflicts. And in a
way, the author's initials are also descriptive: they tell you who
wrote the piece even if you are currently not in idlwave mode and
simply look at a directory listing.

>

> 3. If the routine/class/function/widget name following, e.g., JDS, is
> so ambiguous as to require the initials to discriminate it from another
> of the same name, either the routine/class/function/widget isn't that
> useful, or its name is entirely too inspecific. And the way I think

> about it, since IDL doesn't do any shadow checking (but cf. idlwave!),
> the *best* routine with a given generic name will rise to the top.
That's completely idealistic. Hey, JD, I wouldn't have expected such
an unreflected thought from you. Besides the create_struct ...
problem, just think of the famous colorbar routine. Do you know which
one I am referring to? I know of at least 4 of these, and they all
have their specific merits. The only "solution" in that sense would
probably be that RSI selected one of the conflicting routines to
incorporate it in the IDL distribution and maintain it. But I know for
sure they are not up to it (and I feel sympathetic).

Also, if I see mgs_colorbar, ... in my routine, I know where to
find the source code: it's in my library. If I see fsc_color, it's in
lib/david_fanning. This often proves very valuable - even considering
the fact that idlwave has these nice features. But idlwave is not the
only editor that people are using (unfortunately).

>
> 4. The author(s) can always be found in a proper documentation header.
True, too.

Another thought about this software and the ownership: I had given it
several thoughts how to properly acknowledge David for the invaluable
"contribution" he made to these programs. Well,

(1) I definitively wanted to change the name, because mgs_field just
doesn't work like fsc_field, and you can't use it as a drop-in
replacement.

(2) I considered writing David and me down together as co-authors, but
that could mean that he will receive the emails from people asking for
help, and I am not sure he wants to do it ;-)

(3) I felt that, although significant portions of David's code have
entered my programs verbatim (especially in mgs_field and
mgs_drawcolor), there have been changes that are substantial enough to
claim that these are new programs (several of them are completely new
anyway).

(4) I am trying to make sure that David's name appears at least twice
in every of these programs ;-)

Please tell me if it doesn't.

I encourage everyone of you to do the same (exploiting software via
copy and paste) with my programs: that's why they are open software!
In fact, Ben came up with this twolist object only a day after I sent
a preliminary version to him; and I certainly don't feel bad about him
taking pieces of my code. Rather the opposite: I am convinced that
this is the fastest and most efficient way to write (object widget)
programs: Take pieces from available software and build it together to

