
Subject: Re: RSI's Priorities (was: GUI Builder...)
Posted by [Brian Jackel](#) on Sat, 14 Nov 1998 08:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Robert S. Mallozzi wrote:

- > I agree - objects and pointers greatly increased IDL's
- > functionality and ease of use. They, as widgets did, take
- > IDL to a new level. I often find myself thinking,
- > what's the best way to handle this data...use a pointer, of
- > course. In fact, it's interesting to think of how I ever
- > got along without them :-)
- >
- > Also, I feel there is no doubt about it - object programming
- > is, in most cases, far superior to the "old" procedural method.
- > Code reuse is one of the great advantages - write the object once,
- > then *forget* about the internals, and just use it. That said,
- > of course there is still the need for fast development and
- > analysis, and objects might not be useful for this type of
- > "quick and dirty" work.

Pointers are definitely handy, especially for getting large amounts of data between widgets. As for objects, well...

Perhaps the problem is that most of my work is "quick and dirty"! For example, I've just finished deblurring some satellite images. This required

- 1) Figuring out how to read in a peculiar data format
- 2) Cleaning the data and calculating statistics
- 3) Trying several different deblurring algorithms
- 4) Creating a bunch of figures for a report

The whole process was very interactive and iterative. Now I'm (sort of) done and have a set of working routines and some example scripts. It might be nice to go back and clean everything up, but time is short. Sure it could be nicely packaged with objects, but I can't see how objects would have helped me during the development phase.*

Another important issue is that most of us researchers don't have much experience with object oriented programming. If I give some IDL code to someone who knows Fortran/Pascal/C they can probably see what's going on and make whatever changes they need. But objects? Forget it. Again, life is short and procedural programming works well enough for most of us.

Hmmm, I'm sounding like an old codger now ("In my day,

all we had were zeros and ones, and sometimes we didn't even have ones!"). Ten years ago as an undergrad I'd learn new programming and scripting languages just because they were cool. These days I'm more interested in physics, and just want tools that work.

There does seem to be a clear distinction between people involved in

- 1) Research/analysis
- 2) Development of ``production" level code

For what it's worth, I'd say every one of the half-dozen IDL users in our group fall into category 1.

Brian Jackel

*A general ``image" object might have been useful, but I'm not sure how a sufficiently generic thing could be created. A satellite image is a completely different beast from the image produced by an all-sky camera. I'll just stick with a 2D array, and do whatever pruning is required on a case-by case basis.
