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

In article <MPG.10b6b35a5dca0660989722@news.frii.com>, davidf@dfanning.com (David Fanning) writes:

- > Brian Jackel (jackel@danlon.physics.uwo.ca) writes:
- >
- >> However, I'm watching recent developments with objects and
- >> object graphics with some dismay. All of the data analysis
- >> and display that I've needed to do has been easy to accomplish
- >> without any of these newfangled object thingies. Seriously,
- >> I've managed to maintain some moderate sized programs without
- >> using objects. Given the existence of objects it is quite
- >> likely that I'll get around to using them eventually, but I
- >> can't help but see this as a choice between
- >>
- >> New bells and whistles at \$1500 a pop or
- >> Freeze development and deliver an ``IDL Classic" at
- >> \$500 per head.
- >
- > Objects have absolutely revolutionized the way I write
- > programs. I think they will have as big an effect on me
- > as widgets did when they were first introduced. I don't
- > think I've written a program for pay in the last six
- > months that didn't have at least one object in it. And
- > I especially like them with direct graphics. They are
- > fast, they print nicely, and they are unbelievably
- > powerful.
- >
- > Could I do the same thing without objects? Probably, if
- > I were clever enough. But objects just make it so darn
- > easy. :-)
- >
- > I cast my vote more more bells and whistles like this
- > one!

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.

Regarding the recent discussions of the cost of IDL, it is a bit disconcerting that the price of the software really prohibits me from getting it for home use, and I believe is restricting the possible widespread use that IDL might enjoy if it was lower cost.

I think stand-alone IDL executables would certainly be a step in the right direction for increasing the user base and thus (hopefully) reducing the cost of the software. If people could see the power of IDL without having to buy the interpreter, it seems to me that there would be some developers that might say "Gee, now that I see how powerful the language is, maybe I should think about buying it to do some development...". So, how about it, RSI - give us the capability to distribute IDL executables for the world to see!

—

Robert S. Mallozzi 256-544-0887
Mail Code ES 84
Work: <http://www.batse.msfc.nasa.gov/> Marshall Space Flight Center
Play: <http://cspar.uah.edu/~mallozzir/> Huntsville, AL 35812
~~~~~