

Reimar Bauer (R.Bauer@fz-juelich.de) writes:

- > I like very much the I of idl it stands for interactive.
- > For my opinion objects can't be interactive they are always well planned.
- >
- > The people starting with idl are almost very different in learning and
- > working with idl. In most cases my feeling is if they can theireselfs work
- > a bit on a solution without reading many books or asking a lot of people
- > for help they like idl more.

I'm not sure how this discussion got turned in the direction of objects. I'm pretty sure it started out as a discussion of how to write a modal dialog widget. But the fact that it did encourages me. I've learned in my programming therapy sessions that talking about something is the first step in the process of accepting it. :-)

Objects are definitely well planned. But the fact that they aren't interactive stems entirely from the nature of the objects we have been given in IDL, not from an inherent property of objects. The point is (and I know this for a fact) that if you have the right object library, objects can be as interactive--and as easy to build things with--as any other tool we have in IDL.

In fact, Dave and I have been building object tools in the past several days (CTV, CWINDOW, CWDELETE, etc.) that mimic basic IDL interactive commands, but which put the full power of objects into the hands of people who really need to know nothing about objects. We find the problem with objects is not the lack of interactivity, but the great temptation to be too clever with objects. They can, in fact, be TOO interactive!

Let me give you an example. Our basic image class, IMAGE2D, creates an image class that basically combines all the functionality of my programs TVIMAGE, FSC_COLOR, SELECTIMAGE, and XCOLORS. Thus, the image can size itself into any window, with or without preserving its aspect ratio, can read and load new image files, can manipulate its own color table, can draw and maintain its own overlays (map coordinates, for example), can tell you the actual data value for any point you select on the image, can automatically apply any image processing object to itself (including those you might write in the future), and can support an unlimited (and

user-defined) number of undos and redos.

This all comes for free when you type "CTV, image".

And, as a bonus, if you don't have a window open when you type the command, a GRAPHICSWINDOW will open so that you have a completely resizable graphics window with your image inside it.

That's pretty interactive. And it can also be surprising.

Yesterday I noticed that when I loaded a PNG image that had a built-in color table that the image was drawing itself before I was ready for it to be drawn. It turned out that it was drawing itself because I was loading the PNG's color table in the image's color tool, and the color tool was smart enough to send a message to the image to redraw itself whenever its colors changed. So I was fooling around adding the capability to unregister a message.

The point is that a LOT of work goes into the first layer of objects that you write (what we call the Catalyst Library). But that as you work on the library, you spend less and less time on the low-level routines where I was working yesterday and more and more time building more complex objects and applications.

In the large satellite observation system we built with our first attempt at this library, we spent about 60% of our time designing and building the 25-30 core objects for our library and about 40% of our time building the actual application. I think everyone can appreciate that if your library is basic and powerful enough to be re-usable, that application building is going to be MUCH faster.

And in the meantime, even the object neophytes can take advantage of some of the object's really neat interactive functionality. :-)

Cheers,

David

--

David W. Fanning, Ph.D.

Fanning Software Consulting, Inc.

Phone: 970-221-0438, E-mail: david@dfanning.com

Coyote's Guide to IDL Programming: <http://www.dfanning.com/>

Toll-Free IDL Book Orders: 1-888-461-0155