Subject: Re: Top 10 for old farts
Posted by davidf on Sun, 30 Jul 2000 07:00:00 GMT
View Forum Message <> Reply to Message

Since I enjoy (as Brian Jackel puts it) a "pointless, but interesting discussion" almost as much as I do writing IDL object programs, let me just weigh in with one more response.

Pete Riley (uk2@linuxfan.com) writes:

> Why not replace all the direct graphics calls with object
> equivalents? You should be able to say:
>
> PLOT, findgen(10)
>
> and you get an object window (maybe plus the print, modify,etc...options
> that are in insight). There should have been a transparent migration to
> object graphics.

This was exactly RSI's original intention. But, alas, things don't always go the way we plan. And with object graphics the vagaries of programming projects were even more pronounced than usual. It was almost as if Coyote himself were put in charge of it.

Let me recount the story as I remember it. (Which is to say, this may not be how it *really* happened. But it is the way I tell the story.)

Some four or five years ago now RSI realized that their graphic system was out of date. It was wonderful for the early 1980's, but these were the 90's and the millennium was fast approaching. And in particular, their 2.5D way of rendering surfaces (the Z axis must always be vertical) was hopelessly out of date. Three orthogonal axes should be completely rotatable. The decision was made to completely overhaul the entire graphic system and bring it up to date.

One of the best 3D graphic guys in the country (I forget his name) was hired to design the system and the RSI programmers set to work. A year was set aside for the effort, during which time there would be no work done on bug fixes etc., as everyone was tied up in building this new thing. Customers would get jumpy about paying maintenance dollars during this time, but the new system would be so much better than the old that this problem was judged worth the risk.

To make a long and painful story short, it took a LOT longer than one year to do the work. In fact, it was nearly two years by the time IDL 5.0 left the barn. Customers were screaming, maintenance revenue was dwindling, new license sales were off as customers were waiting for the big new release that was right around the corner. And expenses were up sharply as more programmers were hired to keep up with the larger and larger work load. I think everyone in the building was putting in 60+ hour weeks. In short, it was one of those really tense, trying times that all businesses go though occasionally.

But about 6 months before the actual release the graphics system was built sufficiently (most of the effort up to then had gone into coding the graphics primitives that could be assembled into line plots, surface plots, etc.) that simple "high level" graphics commands could be tested. The idea had always been (I think because this is what the graphics guy told us) that 2D graphics would "fall out of" the 3D graphic system. In other words, 2D graphics were just a specific subset of the 3D system.

But that turned out to be a bit of an oversimplification. Because when the first line plot was tested in the new graphics system it rendered about 10 times slower than the equivalent direct graphics command. In fact, *all* of the 2D commands were about 10 times slower than the equivalent direct graphics commands. The problem, as it turned out, is inherent in 3D systems, and cannot be easily solved even today, I think. It is the reason you have never been able to make a decent line plot in a fabulous 3D graphics system like AVS.

Oh, oh. We were going to go to angry customers and tell them "OK, you paid us a lot of maintenance money for two years and didn't receive anything for it, but here is your great new system. And by the way, it is 10 times slower than the old one." It was going to be a hard sell, to say the least. :-)

So the decision was made (the correct one, I think) to keep the old direct graphics system, but to add the new object graphics system to IDL. Customers could use the one that made the most sense to them. It was expected that most line plots and image displays would be done in the old direct graphics system (since it was a lot faster than the object graphics system), but that

visualizations that required a robust 3D capability would use the object graphics system.

That is pretty much where we are today.

Where RSI has fallen short, I think, is in not making more high-level object graphics commands. It is still pretty much a system that has to be built from scratch. (I know there is IDL Insight. But have you used it!? I find it infinitely frustrating and completely non-intuitive. Plus, aesthetically it is awful. I'm embarrassed by it whenever I have to show it to someone.) And no one else has stepped forward really to make the kinds of tools that people need. (Partly, I should tell you, because no one seems to want to pay you for making the effort. :-)

My sense of what is happening in object graphics is that RSI is moving on to other things (I don't see much in the way of updates on the list of features for IDL 5.4.) Perhaps they are locked into Insight for political reasons, I don't know. Or perhaps there are more people using it than I realize. (Lord knows I couldn't abide a Top 10 Things I Like About Insight thread.)

But, in any case, objects (which had to be invented to create the object graphics class library) are NOT object graphics. Too many people fail to make the distinction. When I talk about object programming I am very rarely talking about using the object graphics class library. I'm taking about using a new way of constructing programs that has the potential to make your programs much more useful to you.

This ability to create your own objects is the major story, I think, in the IDL 5.0 release. Not the object graphics systems. I don't know why it has fallen on me to be the spokesman for the Crusade to Get the Object Story Out, but it appears to be the case. But then Coyote has always liked his little jokes. :-)

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