
Subject: Re: GUI in IDL or PV-WAVE

Posted by [David Fanning](#) on Wed, 25 Nov 2009 17:02:01 GMT

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

John writes:

> Thanks for the quick reply and you are quite correct, I'm not a
> programmer. I have some knowledge of programming from the "good old
> days" but no OO experience. However, I am an expert user of this type
> of software and I will be heavily involved in the design process. My
> programmer colleagues will actually do the spade work and I would like
> to find them a development environment that they are happy to use and
> that produces good results efficiently and quickly.

Yes, this is known in the trade as the "holy grail". :-)

I think you will find IDL somewhere near the top of the pack when it comes to ease of use and producing results quickly. But, of course, "results" is a relative term and your mileage may vary depending upon the skill of the programmer and how well he or she knows their tools. Producing large and complicated application programs has never been "quick" or "easy" in my personal experience, but then I'm still on the holy quest, too.

> One very important feature of the new software that myself and other
> expert users need is a way of customising the application whilst we
> are using it, e.g. via a macro or a scripting language and this has
> helped to prompt our interest in both PV-WAVE and IDL.

It is not clear to me what this means, exactly, but it sounds ominous. If I were bidding on a job like this, this is the paragraph that I would underline in red and put three stars in the margin. I'm not saying it is impossible. I'm just saying I would need to know a LOT more about what you have in mind.

> Our GUI would need to be quite complicated though, particularly when
> analysing the data. We like to graphically interact with the data and
> would need rubber band rectangles and lasso tools to select data and
> then various other tools to carry out operations on the data. As all
> of our data is time based, we also like to use "data brushing". This
> is where we might have three or four X-Y plots open for viewing with
> one of them in focus for editing. If we highlight data on the main X-Y
> plot between certain times, we would like to see a similar highlight
> automatically appear on the background plots between the same times.
> If this sort of thing can be done relatively easily and can be made to
> look ?sexy? in Windows, then IDL might be the way to go for us.

"Sexy" is easy enough, I guess. (Coyote could help out.) And

communicating between plots or windows is not particularly difficult. I've done that any number of times in building applications like this. It will help, tremendously, if you write your IDL application using objects, however. Otherwise, you will quickly become nearly overwhelmed keeping track of things. It will help a lot if you keep program intelligence de-centralized.

Since I've written these kinds of programs over and over, I have developed an object library, named Catalyst, to help me build these kinds of applications. (We actually thought we had found the holy grail with this library, until we realized we would have to draw a map (i.e., write documentation) if we wanted others to find it, too. That discouraged us.)

In our hands, the Catalyst Library does allow us to write applications "quickly and efficiently". Or, at least in about half the time it used to take us, since so much of the infrastructure of the program is built in. For example, object to object communication. One object simply registers an interest in another object's actions. When that action is performed, the interested object is notified. This would make it, for example, trivial to implement the highlighting action you wish to see in your program. The library is available for free and is well documented internally. What is missing is a User's Guide that might provide a broad overall perspective.

But I think people have been able to look at the example programs that come with the library and pretty quickly figure out how to make it work. It helps, obviously, to know both IDL and object programming. Anyway, it might be something to look at while you are doing your evaluation.

<http://www.dfanning.com/tips.html#Catalyst>

Cheers,

David

--

David Fanning, Ph.D.

Fanning Software Consulting, Inc.

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

Sepore ma de ni thui. ("Perhaps thou speakest truth.")
