Subject: Re: Sacrilegious but genuine question
Posted by Francis Burton on Thu, 10 Jan 2002 10:15:29 GMT
View Forum Message <> Reply to Message

Andy Loughe wrote:
>
> Mark Fardal wrote:
>>
>> Francis Burton <F.Burton@biomed.gla.ac.uk> writes:
>>
>>> How easy would it be to obtain the functionality of IDL by
>>> using Python as the underlying language glue, supplemented
>>> by hard-coded modules for image, signal & file processing,
>>> graphical I/O etc.?
>>
>> Not very easy. But easier than say, writing an open-source
>> operating system, and look what happened there...
>>
>> Just don't try doing it by yourself. The Numpy and SciPy mailing
>> lists are places to start. At least on the Numpy list, the topic of
>> IDL emulation and translation has come up recently.
>>
>> Mark
>
> Mmmm... maybe you should ask someone who has done it!
>
> http://nickbower.com/computer/pydl

Thank you for the pointer to PYDL, and to Liam Gumley who also mentioned it a couple of days ago.

While I think it is interesting and worthwhile project, it is really only a start (it's at version 0.1b2). It offers very limited functionality, the documentation looks nice but is riddled with errors, and some of the comments therein made me wince a bit. E.g. talking about READ_ASCII "If you don't use white-space delimiters, then use the UNIX command tr to create a white-space delimited version. Live with it." There is no reason why trivial misfeatures like that can't be fixed. And they probably will be if enough people who care about quality get involved.

What I think PYDL in its current state is valuable for is to show what is possible. What I have seen so far is encouraging.

Let me make it clear that I have nothing against IDL - apart from its high cost (putting it beyond the reach of individuals like me), the irksome licensing issues, the uncertainty about

which platforms will be supported, the fact that it is rather difficulty to add functionality of one's own (see below), and its rather odd syntax (e.g. the comma between command name and arguments). I =do= like its very respectable performance and the fact that all the functionality is provided in a single package - though I don't see why PYDL (or similar framework) couldn't at some point be provided as a consolidated bundle.

I mentioned extensibility. What I would =really= like to be able to do in IDL is have windows which allow much greater interaction with plotted data. For instance, I have fairly large (50,000-500,000 point) digitized signals which I would like to 1) plot in an efficient manner (i.e. min-max for each x pixel rather than drawing a line between every point) and 2) create and overlay various mousable widgets which return values to IDL code. For example, I'd like to be able to click an arbitrary number of points under the trace to define a baseline for subtraction - and be able to adjust the position of already defined nodes, but constrain them to be monotonically increasing in x. In this case, the widget (if that is the right name for it) would return a list of (x,y) values. Then the IDL code could do the subtraction and replot the adjusted trace.

The trouble is that I have no idea how to do this in IDL. I'm not even 100% sure that it is possible - at least, I believe the effort required would be prohibitive. On the other hand, a well designed framework of the type exemplified by PYDL =could= make such extension quite easy.

Anyway enough rambling. Thanks to everyone who responded.

Francis