Subject: Re: IDL/Wave alternatives
Posted by Jarle Brinchmann on Wed, 25 Oct 2000 07:00:00 GMT
View Forum Message <> Reply to Message

hcp@newsread.ed.ac.uk (H C Pumphrey) writes:
>
> (5) PerlDL
> ----------
> PROS: It _is_ perl, with add-ons.
> CONS: It _is_ perl, with add-ons. 2-D graphics done via pgplot, which is
> not true free software. No geographical maps.

Since I do some development for PDL (as well as being a regular IDL user) I thought I'd pitch in here with a couple of extra points.

Re: Graphics
- A package to do geographical maps was just created for PDL.
- There is also a (constantly developing) interface to OpenGL which at least as long as you use Mesa is free.

I use PDL a lot because it is free, which is nice for my home machine. It also has some distinct advantages of IDL in particular for multi-dimensional array manipulations, as PDL has a concept of 'threading' over extra dimensions so that most functions that can deal with a 2D array knows what to do when presented with a 7D hypercube etc. In comparison with IDL this has allowed me to cut down on loops significantly.

It is also (IMHO) much easier to interface external libraries/C/Fortran code to PDL than to IDL. Of course it all depends on how used you are to doing this in IDL but hey .. :)

On the negative side there are two points that stand out:

- First of course the point about it being Perl. If you love Perl, you are likely to like PDL. If you hate Perl, you probably will lose your hair very fast with PDL... I love the flexibility and elegance of PDL which is why I use it extensively despite being a long-term IDL user, but sometimes Perl just makes your programs unreadable. It is possible to counter that with OO programming for instance, but it requires some constrains on the programmer.

- The second is the small code-base. At the moment the code is actively developed by people that have spent time on the core of the language and very few packages of "useful" code have been made publicly available. If your main interest is image analysis,

esp. astronomical, we have a reasonable set of code ready but for many other applications we are not comparable with IDL.

In addition for many IDL users it might be of interest that there is no interface to HDF (although I am writing one), but there is one for netCDF.

I would advise any Perl friends to have a look, there is also a real chance that the PDL code-base will end up in perl6 when (if? :) this surfaces, and it is at the moment a reasonably stable data language.

Anyway, let me close with the warning that unless you are prepared to a lot of coding do not expect PerlDL to be a replacement for IDL - it certainly isn't yet for me.

Jarle.
