Subject: Re: IDL as programming language?
Posted by n9140397 on Tue, 30 Aug 1994 23:51:45 GMT
View Forum Message <> Reply to Message

In article <1994Aug28.122441.32497@waikato.ac.nz> abz@waikato.ac.nz writes:
>
>
> OK. One the lecturers in our department recently went to Hawaii & America for a
> few weeks (lucky him) and came back with the idea of using IDL as a complete
> programming language rather than as just a plotter for FORTRAN generated data.
IMHO, this is a very bad idea... Unfortunately, I have only a few things to
back this up. Mostly it's a feeling, which doesn't hold much water, but,
see below...

> (i) Accuracy. Our current version of IDL seems to prefer doing calculations
> in single precision, while we prefer double. Has this been improved in the
> latest version? (e.g. in our current version, routines like LUDCMP work in
> s.p., despite being passed d.p. arguments.)

My first reason for not liking IDL is this. You have to very explicit
about double precision variables. This has not been improved to my
knowledge, but, I don't have the latest version, either.

> (ii) Speed. Some of us Grads are running some really time consuming programs
> (large arrays, large loops). How does IDL compare with (say) FORTRAN in
> general, speedwise? (my impression is that it's pretty slow, but I could be
> wrong...)

This is my impression as well... it's slow, even though code-wise, it's really
nice to do large array computations because you can often skip explicitly
writing loops.

> (iii) Memory. How does IDL's memory management compare? Again, some of our
> programs (FORTRAN) have a tendency to gobblelarge chunks of memory (probably
> bad programming, but still...)

Really bad. IDL is always sucking up lots of memory, in my experience.

> (iv) What is a large IDL code like to debug?

UGH! Again, my opinion. Some people will probably like debugging idl code
over FORTRAN code (or some other legitamate language). But, I would venture
that they do not know how to use a real debugger like dbx...

> (v) How 'robust' is IDL as a programming language? We have a variety of
> different programming styles here -- some prefer 'quick and dirty' programmin
> others a more structured approach. Forgive my possible ignorance, but I have
> the impression that IDL as a language is more suited to the 'quick and dirty'

> approach. Is this true? Does IDL as a programming language have many
> glitches or inconveniences from a mathematical programmers point of view?

IDL is the epitome of quick and dirty. Yet another reason I don't like it.
It's almost impossible to make idl code organized, it seems.

Other things you should consider: IDL does not, to my knowlege, create
executable code which you store on disk or which runs independently.
If you want to share something you wrote with a friend, they have to have
idl also, and maybe even the same version. Actualy languages like
C and FORTRAN are much more "portable".

Also, a good optomizing FORTRAN compiler is probably still the fastest
thing around as far as an executable goes. This from a C advocate...

IDL is nice for making plots, and you can write quick and dirty, but,
I sure wouldn't want to write a model in it...

My 2 cents....

Mike

******Opinions are my own and not necessarily those of my employer*********