Subject: IDL as programming language?
Posted by abz on Sun, 28 Aug 1994 00:24:41 GMT
View Forum Message <> Reply to Message

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. Apparently this is being done overseas. Our department (mathematics) has been considering this possibility, but nmaking this step would be fairly costly for us, as we'd have to extend our license, so we want to be sure that we would be doing the right thing. We have a number of questions, concerning comparisons between IDL and other programming languages (particularly FORTRAN). We are currently running an older version of IDL (2.2.2) on a Sun SPARC station.

(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.)

(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...)

(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...)

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

(v) How 'robust' is IDL as a programming language? We have a variety of different programming styles here -- some prefer 'quick and dirty' programming, 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?

Any info/advice would be much appreciated. The types of stuff we do here are generally large numerical (finite difference) codes on 2D and 3D grids. I'll wait a while before sending this to see if there is/has been any discussion on this topic in the newsgroup.

Please email me, or post to this newsgroup if you think anyone else will be interested.... Thanks

Alec.