Subject: Re: FORTRAN to IDL translation
Posted by Michael Asten on Tue, 23 Nov 1999 08:00:00 GMT
View Forum Message <> Reply to Message

I was faced with a similar task when I needed a procedure to compute modified bessel functions of complex argument - idl routines happily accept a complex argument, but only operate on the real part (*#!). In the end I did a line-for-line translation from fortran and found it quite straightforward, since idl allows horrible programming features common in F77 such as goto statements and line-labels. Without optimising loops, I found that the speed comparison on the maths was
fortran : idl = 3 : 11 seconds.

The big challenge is likely to be the rewrite to take advantage of array processing - good luck.

Regards,
Michael Asten

Liam Gumley wrote:

> Henrik E. Nilsen <Henrik.Nilsen@fi.uib.no> wrote in message
> news:3836BDF2.9C6F44E2@fi.uib.no...
>> I have a small Fortran program (1000 lines) that I would like to translate
>> to IDL. I've looked arround the net for a translation utility, but can't
>> find one. Does anyone know if there is such a thing? I'm now looking at
>> making a simple utility like this myself. My aim is to make an IDL
>> routine that can translate Fortran code line-by-line. I expect to
>> translate about 80-90% of the code this way, leaving the rest for manual
>> translation. Finally, the code must be restructured some (maually)to take
>> advantage of IDL functions (Array manipulation, objects, etc.) Any Ideas?
>
> There is no utility to transform FORTRAN to IDL. Good luck on constructing
> one yourself. Just make sure you don't translate all those FORTRAN loops to
> IDL loops :-)
>
> Cheers,
> Liam.
> http://cimss.ssec.wisc.edu/~gumley