Subject: Re: Common block access in DLM
Posted by Rick Towler on Tue, 18 May 2004 20:44:43 GMT
View Forum Message <> Reply to Message

"Haje Korth"  wrote...

> thanks for the example, this is more than I expected. Regarding my
project,
> I ported a FORTRAN library for geophysical coordinate transformations and
> magnetic field calculation from FORTRAN to IDL usind a DLM, basically IDL
> <-> DLM <-> C <-> FORTRAN, taking Ronn's exercises a little further.
> Unfortunately I only have Ronn's first edition book, so I do not know if
he
> expanded on this in the second edition.

I can't say.  I know that he updated the text to include the new keyword API and added the C++ section.  I'm sure he did more, those were just the two most compelling updates for me.

> Anyway, everything works as
> expected, but having to supply year, day of year, hour minute second for
> every call is nerve wrecking, and blows up the call to so many parameters
> that the IDL code looks confusing. I will give the global variables a try.

That's a good place to start.  Then if you need the ability to run multiple instances you can go the C++ route.  I could dig up an example if you like.

Also, have you checked out Stein Vidar Hagfors Haugan's dlmform?  It might be able to handle the tedium for you.

http://www.astro.uio.no/~steinhh/idl/additions.html
http://www.astro.uio.no/~steinhh/idl/dlmform.html

> How do setup such a global variable in
> C? Is this just regular C language, or is there some fancy handling
required
> with the IDL external API? (I am pretty much C newby, but once I now this,
I
> think I can handle the interfacing with FORTRAN.)

Plain old regular C.  Simply define it before your first function and it will be available globally.  I too am fumbling thru the dark when it comes to C/C++ but luckily there are enough people in the group that can help us when we get stuck. :)

Good luck!

-Rick