

---

Subject: Re: Many procedures, what to do?

Posted by [davidf](#) on Fri, 12 Jan 2001 00:35:36 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

Craig Markwardt ([craigmnet@cow.physics.wisc.edu](mailto:craigmnet@cow.physics.wisc.edu)) writes:

- > I have about twenty five functions that go into a single larger
- > library. They are all pretty much inextricably linked together. This
- > is the SAVE file library I have mentioned once or twice, with the
- > ability to read and write and interrogate save files yourself.
- >
- > Anyway, my question is what to do with all these files? I would
- > imagine that for most people, including myself, this many files is a
- > pain in the neck. Also they share a lot of the same parameters so it
- > would be difficult to keep the documentation up to date.
- >
- > My other option is to merge them into a single file called, say,
- > CMSVLIB. There are a couple of problems with that.
- >
- > First, how to get them compiled. That's easy, I just require every
- > program which calls the library to invoke CMSVLIB first. As long as
- > there is actually a procedure called CMSVLIB at the end of the file,
- > this should force all the other routines in the file to be compiled.
- >
- > The other problem is more subtle. Since none of the individual files
- > are compiled when the invoking procedure is compiled, IDL won't know
- > about the functions. It will see the round parenthesis of
- > "cmsv\_rlong(block, pointer)" and think it's an array subscript.
- > Arghh.
- >
- > Okay, that can be solved by forcing everybody to declare the functions
- > they use with FORWARD\_FUNCTION. Now it's starting to get annoying
- > again. I guess I could rewrite everything to be procedures...
- >
- > Does anybody else have suggestions, or experiences with something like
- > this?

I have to admit, I'm always confused when I read articles like this. I've been writing IDL programs for over 12 years now, and I've maybe used FORWARD\_FUNCTION three times. I have gotten the wrong FOO program from some other directory maybe a few times more than that. But I have never, in all those years, had difficulty getting programs to compile and run when they are suppose to.

What have I been doing wrong! :-(

My rigid rule of thumb is that if there is a "command" I am

trying to build (procedure or function), the file is given that name and that module goes at the end of the file. Any support routines that are NOT meant to be "commands", in the sense that I want someone to use them from the IDL command line, are placed in front of the command module. If something becomes "command important", I yank it out of the file and put it in a file of its own, with the file given the command name, as before.

I just can't see how IDL can fail to get it right if each "command" is in a separate file, and the files are in a directory on the path. It works every time for me.

Cheers,

David

--

David Fanning, Ph.D.

Fanning Software Consulting

Phone: 970-221-0438 E-Mail: [davidf@dfanning.com](mailto:davidf@dfanning.com)

Coyote's Guide to IDL Programming: <http://www.dfanning.com/>

Toll-Free IDL Book Orders: 1-888-461-0155

---