
Subject: Many procedures, what to do?

Posted by [Craig Markwardt](#) on Thu, 11 Jan 2001 23:47:28 GMT

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

Greetings all!

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?

Craig

--

Craig B. Markwardt, Ph.D. EMAIL: craigmnet@cow.physics.wisc.edu
Astrophysics, IDL, Finance, Derivatives | Remove "net" for better response

Subject: Re: Many procedures, what to do?

Hi Craig:

I had a similar problem some time ago. My solution was to have a number of different source files with routines and functions grouped together:

```
easy.pro (main)
easy_define.pro (10+ functions)
easy_fit_proc.pro (3 procedures)
....
(about 20 source files with up to 20(?) single funcs)
```

Secondly, I have a procedure to make the savfile (The procedure also writes some status, user and date information but I cut that) :

easy_make_sav.pro

```
PRO easy_make_sav, SHORT=short

; compile the procedures
RESOLVE_ROUTINE, 'easy'
RESOLVE_ROUTINE, 'easy_utilities'
RESOLVE_ROUTINE, 'easy_operate'
....
; compile all functions
RESOLVE_ROUTINE, 'easy_diagnostic_plot', /IS_FUNCTION
RESOLVE_ROUTINE, 'easy_error', /IS_FUNCTION
.....
RESOLVE_ROUTINE, 'easy_model2userfile', /IS_FUNCTION

IF NOT KEYWORD_SET(short) THEN BEGIN
    RESOLVE_ALL, SKIP_ROUT=excluded
ENDIF

SAVE, /ROUTINE, FILENAME=savfile, COMPRESS=compress ; , /VERBOSE

RETURN
END
```

Finally I wrote a routine easy_startup which loads the sav file for correct version of IDL and does some talking to the user if it was not found.

The concept works fine since then and most of the users in our group do just now that there is easy_startup which loads the whole thing if necessary and starts the easy environment. I normally distribute

easy_startup.pro and easy_54.sav (easy_53.sav etc.) only.

If you have questions or if you want to have a closer look drop me a note.

Best regards Theo

Craig Markwardt wrote:

>
> Greetings all!
>
> 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?
>
> Craig
>
> --
> -----
> Craig B. Markwardt, Ph.D. EMAIL: craigmnet@cow.physics.wisc.edu

> Astrophysics, IDL, Finance, Derivatives | Remove "net" for better response
> -----

--

Dr. Theo Brauers
Institut fuer Atmosphaerische Chemie (ICG-3)
Forschungszentrum Juelich
52425 JUELICH, Germany
Tel. +49-2461-61-6646 Fax. +49-2461-61-5346
<http://www.kfa-juelich.de/icg/icg3/MITARBEITER/th.brauers.ht ml>

Subject: Re: Many procedures, what to do?
Posted by [davidf](#) on Mon, 15 Jan 2001 15:14:49 GMT
[View Forum Message](#) <> [Reply to Message](#)

Theo Brauers (th.brauers@fz-juelich.de) writes:

> The concept works fine since then and most of the users in our group
> do just now that there is easy_startup which loads the whole thing if
> necessary and starts the easy environment. I normally distribute
> easy_startup.pro and easy_54.sav (easy_53.sav etc.) only.

I can tell you that my great idea to distribute IDL save files lasted about 15 minutes. Long enough for me to discover a bug in one of my files and realize I needed to resave every file on every platform every time I needed an update.

No thanks. :-(

Cheers,

David

--

David Fanning, Ph.D.
Fanning Software Consulting
Phone: 970-221-0438 E-Mail: davidf@dfanning.com
Coyote's Guide to IDL Programming: <http://www.dfanning.com/>
Toll-Free IDL Book Orders: 1-888-461-0155

Subject: Re: Many procedures, what to do?
Posted by [Theo Brauers](#) on Mon, 15 Jan 2001 15:59:54 GMT
[View Forum Message](#) <> [Reply to Message](#)

David Fanning wrote:

>
> I can tell you that my great idea to distribute IDL save
> files lasted about 15 minutes. Long enough for me to
> discover a bug in one of my files and realize I needed
> to resave every file on every platform every time I needed
> an update.

>
> No thanks. :-(

>
Bugs suck anyway. The same savfile works on Windows and Unix :-)
only versions matter ;-(. In order to patch a saved program
just mail a new version of easy_startup which compiles the
de-bugged source file after restoring the savfile. I am not going
to recommend savfiles for any set of routines but I recommend
it if you want to provide exactly the same software for
different users. In my case the savfile (400+ routines)
identifies itself when it writes output, so that there is a
version control even if the user crashed his/her IDL. I update
every 3 month after having tested the new version some weeks.

Cheers,
Theo

Subject: Re: Many procedures, what to do?
Posted by [Craig Markwardt](#) on Mon, 15 Jan 2001 16:09:32 GMT
[View Forum Message](#) <> [Reply to Message](#)

davidf@dfanning.com (David Fanning) writes:

> I can tell you that my great idea to distribute IDL save
> files lasted about 15 minutes. Long enough for me to
> discover a bug in one of my files and realize I needed
> to resave every file on every platform every time I needed
> an update.

I have to agree on this one.

But your FSC_PSCONFIG lasted in SAVE form longer than 15 minutes
didn't it? :-)

Craig

--

Craig B. Markwardt, Ph.D. EMAIL: craigmnet@cow.physics.wisc.edu
Astrophysics, IDL, Finance, Derivatives | Remove "net" for better response

Subject: Re: Many procedures, what to do?

Posted by [Alex Schuster](#) on Mon, 15 Jan 2001 17:45:14 GMT

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

Craig Markwardt wrote:

- > 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.

That's what I do for large projects, too.

- > 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.

I tell all users that they have to call the big routine once before doing anything else, when they want to use it. Well, maybe short after DEVICE, DECOMPOSED=0 and such.

- > 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...

What about a startup file (the one that gets executed when IDL is started) containing all the FORWARD_FUNCTIONS. It could as well compile your CMSVLIB, but that would take some time, probably too long. That's another disadvantage of the many files, whenever IDL is started, it scans the whole \$IDL_PATH for files. This is fast, but well noticeable here (2000 files).

Alex

--

Alex Schuster Wonko@weird.cologne.de
alex@pet.mpin-koeln.mpg.de

PGP Key available

Subject: Re: Many procedures, what to do?

Posted by [Craig Markwardt](#) on Mon, 15 Jan 2001 21:28:17 GMT

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

Hi Alex--

Thanks for your support. Finally somebody who agrees with me! :-)
Actually I appreciate where David is coming from, and agree that jumping through too many hoops just makes it harder for everyone.

I can offer some real numbers for consideration. My SAVE library would have a similar number of files to the MPFIT package if the low-level files were compacted into one routine. Going over about 1.3 years worth of download logs, I see that 725 people download a tar or zip file of MPFIT, while 815 people downloaded the individual MPFIT.PRO file. So yes, David, people do like zip files, but they also like to download the individual PRO file too!

I should mention that I converted a bunch of my functions to procedures, so that makes it a little easier to support. There are really just two functions now, and they are **quite** low-level, so almost noone will need to FORWARD_FUNCTION them. So in principle I could put all the files into one master library file with little or no repercussions, as long as I force programmers to call CMSVLIB to initialize the library. I do still like this idea.

I will try it separate files in a zip archive to begin with. I'm getting close!

Craig

Alex Schuster <alex@pet.mpin-koeln.mpg.de> writes:

```
> Craig Markwardt wrote:
>
>> 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.
>
> That's what I do for large projects, too.
>
>> 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.
>
> I tell all users that they have to call the big routine once before
> doing anything else, when they want to use it. Well, maybe short after
```

> DEVICE, DECOMPOSED=0 and such.
>
>> 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...
>
> What about a startup file (the one that gets executed when IDL is
> started) containing all the FORWARD_FUNCTIONS. It could as well compile
> your CMSVLIB, but that would take some time, probably too long.
> That's another disadvantage of the many files, whenever IDL is started,
> it scans the whole \$IDL_PATH for files. This is fast, but well
> noticeable here (2000 files).
>
> Alex
> --
> Alex Schuster Wonko@weird.cologne.de PGP Key available
> alex@pet.mpin-koeln.mpg.de

--

Craig B. Markwardt, Ph.D. EMAIL: craigmnet@cow.physics.wisc.edu
Astrophysics, IDL, Finance, Derivatives | Remove "net" for better response

Subject: Re: Many procedures, what to do?
Posted by [davidf](#) on Mon, 15 Jan 2001 21:46:10 GMT
[View Forum Message](#) <> [Reply to Message](#)

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

> I will try it separate files in a zip archive to begin with. I'm
> getting close!

I was thinking we are all getting pretty worn out
with the anticipation. Talk about a marketing
build-up! :-)

Cheers,

David

P.S. Let's just say with software, I've found it
better to set expectations low. :-(

--

David Fanning, Ph.D.
Fanning Software Consulting

Phone: 970-221-0438 E-Mail: davidf@dfanning.com
Coyote's Guide to IDL Programming: <http://www.dfanning.com/>
Toll-Free IDL Book Orders: 1-888-461-0155
