Subject: Re: God save me!
Posted by Heinz Stege on Thu, 16 Jan 2014 22:08:19 GMT
View Forum Message <> Reply to Message

On Thu, 16 Jan 2014 05:58:28 -0800 (PST), ??? wrote:

> Thank a lot, Heinz. Adding compile_opt idl2 works. But I wonder why sometimes the compilation passes even without the option. And later when I remove the line (compile_opt idl2) the compilation also passes.
>
Zhibo, it is nothing wrong with not using the compile option. However, if you allow round brackets for array indexing (i.e. no compile option), the IDL compiler or interpreter has to find out if "colorbar(...)" is a variable or a function.

I have never read (or investigated) how IDL is doing this. (Because for me it is convenient, to use square brackets for arrays.) I think, IDL makes it's decision on basis of it's lists of defined variables and compiled functions. But I don't really know.

You can try to find it out. Does the compilation pass without an error message, when the colorbar function is already compiled? (You can check this with "help,/routines".)

If you allow round brackets for arrays, then you must not use the same name (in this case "colorbar") for a variable as well for a function. (This is meant for inside of one routine. You may use "colorbar" for a function within one routine and for a variable within another routine.)

If you use the said compile option, everything is simple: "y=colorbar(...)" is a function call, and "y=colorbar[...]" is array subscripting. Note that "y=colorbar[0]" and "y=colorbar[*]" are allowed also for scalar variables.

Hope this helps.

Cheers, Heinz