Subject: Re: legend not working with symcat Posted by wlandsman on Thu, 05 Nov 2009 00:48:24 GMT

View Forum Message <> Reply to Message

On Nov 4, 6:19 pm, David Fanning <da...@dfanning.com> wrote:

- > Wayne Landsman writes:
- >> That would be true if all users had symcat.pro in their !path. But
- >> in case there are some inexperienced users who don't have the Coyote
- >> library installed yet, I've slightly modified your update to
- >> legend.pro to only call symcat.pro for non-standard PSYM values.

>

>> http://idlastro.gsfc.nasa.gov/ftp/pro/plot/legend.pro

>

- > The problem now is that LEGEND won't compile *unless* you
- > have SYMCAT in your path. :-(

Well, I don't think compilation is a problem, e.g. the program

pro test,dum

if keyword_set(dum) then a = some_program(3) else print, 'AOK' return end

will run fine so long as one doesn't try to access Some_program().

IDL> test

% Compiled module: TEST.

AOK

IDL> test,1

% Variable is undefined: SOME PROGRAM.

% Execution halted at: TEST 2 /Volumes/Apps_and_Docs/

landsman/codev/test.pro

(If I had added a compile_opt idl2 at the beginning, it would also know that SOME_PROGRAM is an underined function/procedure rather than a variable.)

Similarly, legend.pro should now run fine without symcat.pro in one's! path, so long as one doesn't try to use a value of PSYM>=11.

But I agree the more general question is difficult. Currently, I keep JHUAPL routines I need in a separate directory, and tell users to delete the directory if they have the entire JHUAPL library installed.

--Wayne