Subject: Re: How to specify which exact version of a subroutine to use....
Posted by David Fanning on Thu, 17 Mar 2011 16:40:17 GMT
View Forum Message <> Reply to Message

George Millward writes:

> I frequently encounter the problem where I am running a program that
> utilizes a number of IDL libraries
> (ie, coyote, solarsoft - etc.) . The problem that crops up constantly
> is with the program addressing the wrong
> version of a given subroutine. ie, right now I am trying to use a
> certain coyote routine and my code is crashing because it is picking
> up an older version of the routine that is attached to the solarsoft
> library - rather than the new version that is located locally on my
> own machine.
>
> Can I specify the exact version of the routine that I want the code to
> use - by providing the full path or something ?

I spent a couple of days trying to get my software
compatible with older versions in SolarSoft. The problem
is that SolarSoft used a number of my programs, then fixed
bugs and added features to the programs without notifying me
about it. So the programs drifted apart.

This is a real problem, and the same thing is likely
to happen now with the Astronomy Library routines,
which also contain most of the Coyote Graphics routines
from the Coyote Library.

Several things can help. First, better communication!
(And, by the way, Wayne Landsman is terrific to work with.)
I don't mind people fixing bugs and adding features to my
programs, but please let me know about it. I'm all for
writing better programs!

Second, we REALLY need a standard repository for all
these libraries, so we can work together on this. Or,
at least, when we build programs we can get the right
dependencies.

I don't know, it's a hard problem. But at least half of
the problems people report to me are caused by old programs
somewhere in their path.

Sigh... OK, had to get that out. ;-)

The program that is compiled is going to get used. Normally,

the "program that is compiled" is the first one found on the
IDL path. You can affect this by choosing your IDL path order.
In the case of Coyote and Solarsoft, you would be better off
with the Coyote directory listed first, since I have at least
*tried* to make my routines compatible with the older Solarsoft
routines. (If you report the problems, I'll try to fix them.
Please don't suggest firearms as a solution, because I would
take you seriously.)

This is a pain-in-the-kiester solution, of course. So, maybe
you need something better. You could create a list of the
conflicting routines and write a program that compiled one
set or the other. Maybe I should write this myself and add
it to the Coyote library. You could use RESOLVE_ROUTINE
to do the compilation for you. I envision something that works
like this:

```
IDL> Resolve_Conflicts, /Coyote
IDL> Resolve_Conflicts, /SolarSoft
```

If you want to use Coyote Graphics routines in conjunction with
SolarSoft routines, I'm not sure what you do. Maybe hand-compile
a routine just before you use it.

```
IDL> .compile /usr/lib/coyote/linkedlist__define.pro
IDL> cgContour, ...., /Window
```

Tough, situation. Maybe a letter to the SolarSoft people
will help. :-)

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.idlcoyote.com/
Sepore ma de ni thui. ("Perhaps thou speakest truth.")