
Subject: Re: Happy Reasons to use IDL 8.1
Posted by [pgrigis](#) on Wed, 25 May 2011 14:34:18 GMT
[View Forum Message](#) <> [Reply to Message](#)

On May 25, 5:58 am, ameigs <andyme...@gmail.com> wrote:

> On May 25, 9:42 am, FÖLDY Lajos <fo...@rmki.kfki.hu> wrote:

>
>
>
>
>
>
>
>
>
>

>> On Tue, 24 May 2011, David Fanning wrote:

>>> FÖLDY Lajos writes:

>

>>>> Probably IDL does not know about your image function and tries to
>>>> subscript the image variable. Does compile_opt idl2 help?

>

>>> Well, this occurs in a fresh IDL session, so image
>>> is undefined. I've also checked to be sure no procedures
>>> or functions named "image" are defined.

>

>>> IDL> .reset

>>> IDL> .compile -v 'C:\IDL\default\junk.pro'

>

>>> img = image(image, /current)

>>> ^

>>> % Syntax error.

>>> At: C:\IDL\default\junk.pro, Line 4

>>> % 1 Compilation error(s) in module \$MAIN\$.

>

>>> Yes, Compile_Opt idl2 helps and seems to solve
>>> the problem. Yet one more reason (if we need one!)
>>> that this should finally be the default behavior!

>

>>> The fact that I know how to work-around the problem
>>> (naming my image "bob" also solves the problem!) doesn't
>>> give me much satisfaction. What do you suppose the
>>> problem is?

>

>> The problem is that image() is ambiguous, it can be a function call or a
>> variable subscript. IDL must choose after reading the tokens 'image' '('.
>> I guess the rules are:

>

>> - if the image function is defined, it is a function call

>
>> - if the image variable is defined, it is a subscript
>
>> - if the image function can be resolved using !PATH, it is a function call
>
>> - it is a subscript
>
>> (here defined means defined in the internal tables of the compiler)
>
>> I think we have to live with it. "bob" or compile_opt idl2 (strictarray)
>> removes the ambiguity. Probably IDL should add an -idl2 command line
>> switch to make it default globally.
>
>> regards,
>> Lajos
>
> Know you're trying to help, but ever heard of the "teaching an old dog
> to suck eggs". David is finding unexpected problems and well he is one
> of 5 experts that I know of in IDL. Sorry, I can't test this
> behaviour; my lab is stuck currently at IDL 7.1.1....
>
> Andy

Andy, Lajos wrote an IDL clone from scratch and therefore is probably more experienced then anybody else about rules regarding how IDL statements are compiled (or how they *should* be....)

Ciao,
Paolo
