
Subject: Scope_VarFetch Bug or Feature?

Posted by [David Fanning](#) on Tue, 14 Feb 2006 21:46:29 GMT

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

Folks,

Here is a short test program and main-level program:

```
.*****  
,  
PRO test  
  
  names = Scope_Varname(Level=1,Count=varcount)  
  FOR j=0,varcount-1 DO BEGIN  
    Print, 'Fetching variable ' + names[j] + '...'  
    mainvar = Scope_VarFetch(names[j], Level=1, /Enter)  
    Help, mainvar  
  ENDFOR
```

END

```
a = 5  
b = c  
test
```

END

```
.*****  
,
```

Compile and run the main-level program. Yes, it will cause an error. Don't worry about it. :-)

When the error occurs, you will have two variables at the main level (LEVEL=1 in SCOPE_ parlance), a and b. The variable a is equal to 5 and b is undefined.

Now, just type "test" and the test program runs. It will find both variables at the main level. But, as far as I can tell there is no way to learn anything more about the variables unless I retrieve them. BUT, I can't retrieve an undefined variable. It causes an error. :-)

(According to the documentation the ENTER keyword should cause a variable with the name I am trying to fetch (b, in this case) to be created at the main-level, but this doesn't seem to work either. At least I still get the error. This DOES seem like a bug to me.)

This is all causing me some heartburn, because I need to

retrieve variables at a particular level and save them.
All works well unless I try to retrieve undefined variables,
and in this particular application, I cannot assume there
won't be undefined variables.

Does this seem like a bug to you, or a feature? Is the
only alternative to CATCH the error?

Cheers,

David

--

David Fanning, Ph.D.

Fanning Software Consulting, Inc.

Coyote's Guide to IDL Programming: <http://www.dfanning.com/>
