Subject: structure array changed in 5.5
Posted by Craig Markwardt on Wed, 12 Dec 2001 22:37:57 GMT
View Forum Message <> Reply to Message

Here's another interesting tidbit about subscripting, this one only applying to IDL 5.5.

I always complained that extracting array values from structures can be dangerous. Consider the following structure:

```
s = {x: [1]}
```

The tag X is a one-element vector. Under IDL 5.4 and earlier, retrieving the value of S.X returned a *scalar*.

```
IDL> help, s.x
<Expression>    INT       =        1
```

This is a problem if you don't know ahead of time whether X should be an array or scalar, because there is no way to find out.

In IDL 5.5, this behavior has changed, for the better I think.

```
IDL>  help, s.x
<Expression>    INT       = Array[1]
```

This also works for higher dimensional arrays, as they should work. The only problem is if you depend on the former behavior. Aaron Barth (et al.) found a problem with my published CMPS_FORM. It relied on the following code to work

```
configs     = predefined(*).config
```

However, the CONFIG tag is a structure itself, and since structures are always arrays in IDL, the returned value is different under pre-5.5 and post-5.5.

```
pre 5.5:    CONFIGS         STRUCT    = -> CMPS_FORM_INFO Array[6]
post 5.5:   CONFIGS         STRUCT    = -> CMPS_FORM_INFO Array[1, 6]
```

You see one is a 1x6 array and one is a 6 element array. This fouled things up down the line (now corrected).

The short answer is that if you depend on the behavior of arrays of structures with arrays (!), then you should double check your code to avoid the above snag.

Yours,

Craig

--

---------- ----------
Craig B. Markwardt, Ph.D. EMAIL: craigmnet@cow.physics.wisc.edu
Astrophysics, IDL, Finance, Derivatives | Remove "net" for better response
---------- ----------