Subject: Re: Passing Arrays BY REFERENCE
Posted by Craig Markwardt on Sun, 23 Aug 2015 22:30:17 GMT
View Forum Message <> Reply to Message

On Monday, August 10, 2015 at 11:46:22 AM UTC-4, g.na...@gmail.com wrote:
> I was trying to find out how IDL can pass arrays BY REFERENCE instead of BY VALUE. I found out this but was not very helpful.
>
> http://www.exelisvis.com/docs/PassingParameterArraysByRefere nce.html
>
> I wanted to return an array BY REFERENCE. Does anyone know how to do that?

Not exactly. All IDL variables are "by reference." If a parameter is passed by reference then it is returned by reference.

In this example,
```
a = [[1d,-0.01d],[.1d,1d]]
choldc, a, p
```
the variable A is passed "by reference" to CHOLDC. If you consult the documentation, CHOLDC modifies the array so that upon return, A is changed. So in that case it is returned by reference as well. The variable P does not need to be defined before calling CHOLDC, but upon return, P will contain new values. So in a sense the variable P was returned by reference.

The only case where "by value" would play a role is in this modified example,
```
a = [[1d,-0.01d],[.1d,1d]]
choldc, a[*,*], p
```
Here the changed part is A[*,*] instead of just A. The A[*,*] instructs IDL to create a new temporary array with all elements of the original array, and pass that to CHOLDC. When CHOLDC returns, that temporary array is discarded, so A does not change. We say that A[*,*] is passed by value. But P would still be passed (and returned) by reference.

That's because all IDL variables by themselves contain a reference to the actual data. They are a name, "A" and a reference to the data, in this case a square array of data. Passing a variable by name in a function call is always by reference.

Craig