Subject: Re: Storing !NULL in struct
Posted by Michael Galloy on Mon, 18 Mar 2013 19:33:47 GMT

View Forum Message <> Reply to Message

On 3/18/13 12:26 PM, Yngvar Larsen wrote:

```
>
>  This is not the case for lists, as already discussed:
>
>  IDL> larr = list(length=10) & for n=0, 9 do larr[n] = {foo: 0}
>  IDL> print, larr[0].foo
>         0
>  IDL> larr[0].foo = 4
>  % Attempt to store into an expression: Structure reference.
>  % Execution halted at: $MAIN$
>
>  I fail to see why the latter isn't allowed.
```

I agree.

```
>  This means that if you still want to modify structures within your list, you need to do something like this
>
>  IDL> tmp = larr[0]
>  IDL> tmp.foo = 4
>  IDL> larr[0] = tmp
>  IDL> print, larr[0].foo
>         4
>
>  This seems silly to me. What is the purpose of not allowing this?
>
>  Switching to hash tables instead of structures as general purpose data structure is of course a very good  strategy in 2013, but I have at least 10 years worth of legacy code which already heavily (mis-)uses the anonymous structure as a "hash table light" with case insensitive keys.
```

I agree here as well. But, I do find it difficult to use new IDL
features for long after they are no longer new since so many users do
not have the latest version of IDL.

Mike
--
Michael Galloy
www.michaelgalloy.com
Modern IDL: A Guide to IDL Programming (http://modernidl.idldev.com)
Research Mathematician
Tech-X Corporation