
Subject: Re: Storing !NULL in struct
Posted by [Bob\[4\]](#) on Tue, 12 Mar 2013 18:15:57 GMT
[View Forum Message](#) <> [Reply to Message](#)

On Tuesday, March 12, 2013 6:09:37 AM UTC-6, Tom Grydeland wrote:

> Any reason I shouldn't expect this to work?

>

>

>

> IDL> help, { foo: !null }

>

>

>

> --T

Structure fields must be defined at the time of creation and the type (and rank and array sizes) cannot be changed after that. Thus, if you put a !null in the field you could never change it to anything else. Not very useful. If you do not know the detail of whatever you are going to put in the structure you have several options. Traditionally, you would use a PTR. These days you could use a LIST or HASH instead.

Bob

Subject: Re: Storing !NULL in struct
Posted by [tom.grydeland](#) on Wed, 13 Mar 2013 08:17:28 GMT
[View Forum Message](#) <> [Reply to Message](#)

On Tuesday, March 12, 2013 6:15:57 PM UTC, bobnn...@gmail.com wrote:

> Structure fields must be defined at the time of creation and the type (and rank and array sizes) cannot be changed after that. Thus, if you put a !null in the field you could never change it to anything else. Not very useful.

I know IDL won't let me change it, but I will happily rebuild structures to have the same field name and different contents to achieve the dynamism that IDL doesn't have. As long as my structures are anonymous, this works fine.

And it would be useful `_to me_` if I could put a !NULL in a struct. I'm using the struct for UVALUES in widgets, and it would be nice if they allowed `_any_` well-defined IDL value.

> If you do not know the detail of whatever you are going to put in the structure you have several options. Traditionally, you would use a PTR. These days you could use a LIST or HASH instead.

Sure. Thanks!

> Bob

--T

Subject: Re: Storing !NULL in struct
Posted by [Michael Galloy](#) on Wed, 13 Mar 2013 14:46:17 GMT
[View Forum Message](#) <> [Reply to Message](#)

On 3/13/13 2:17 am, Tom Grydeland wrote:
> And it would be useful _to me_ if I could put a !NULL in a struct.
> I'm using the struct for UVALUES in widgets, and it would be nice if
> they allowed _any_ well-defined IDL value.

But !null is undefined:

```
IDL> help, !null  
<Expression>  UNDEFINED = !NULL  
IDL> print, size(!null, /type)  
0
```

Mike

--

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

Subject: Re: Storing !NULL in struct
Posted by [tom.grydeland](#) on Fri, 15 Mar 2013 08:10:45 GMT
[View Forum Message](#) <> [Reply to Message](#)

> On 3/13/13 2:17 am, Tom Grydeland wrote:
>> And it would be useful _to me_ if I could put a !NULL in a struct.
>> I'm using the struct for UVALUES in widgets, and it would be nice if
>> they allowed _any_ well-defined IDL value.

On Wednesday, March 13, 2013 2:46:17 PM UTC, Mike Galloy wrote:
> But !null is undefined:

Not so undefined that I cannot assign it to a variable, or return it from a function, or use it as a good 'missing value' indicator in most cases. It is also pretty damn useful if you want to build an array by parts (although, as Bob pointed out, I could use a LIST for that last case).

See, I was trying to stash away an arbitrary value (provided by the user for their later retrieval) under a given tag in a structure, and I didn't want to restrict what values they were allowed to use.

So instead of using a struct { internal: <whatever>, public: <your value here> }, I thought I could use a List(<whatever>, <your value here>), but that didn't work either, since I'm not allowed to assign to a struct that lives in a List.

```
IDL> c = List({ t: 0})
```

```
IDL> print, c[0].t
```

```
0
```

```
IDL> c[0].t = 1
```

```
% Attempt to store into an expression: Structure reference.
```

```
% Execution halted at: $MAIN$
```

```
:-(
```

```
> Mike
```

```
--T
```
