
Subject: Re: Defining Integer Field in a Structure

Posted by [John-David T. Smith](#) on Thu, 01 Nov 2001 21:24:12 GMT

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

Brian Bell wrote:

>
> Is there a way to define a field within a structure whose type is
> integer? In my case, I want the field to be any integer from 0 to 4.
> I know you can assign an actual integer to the field, but I don't want
> to do this because I want the field value to change within certain
> procedures and to return that value to the procedures calling it. I
> would appreciate any help you might be able to offer.
>

This is illegal:

```
IDL> a={0:"this", 1:"is",2:"a",3:"funny",4:"test"}
```

so the short answer is no. However, if you just want 4 numbered "slots", try an array:

```
IDL> a={vals:['this','is','a','funny','test']}
```

```
IDL> print,a.vals[1]
```

is

if you'd like the types of each slot to be different, try at a pointer array. If you'd like the number of "slots" to be variable, try a pointer to a pointer array.

JD

Subject: Re: Defining Integer Field in a Structure

Posted by [David Fanning](#) on Thu, 01 Nov 2001 22:05:42 GMT

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

JD Smith (jdsmith@astro.cornell.edu) writes:

> Brian Bell wrote:
>>
>> Is there a way to define a field within a structure whose type is
>> integer? In my case, I want the field to be any integer from 0 to 4.
>> I know you can assign an actual integer to the field, but I don't want
>> to do this because I want the field value to change within certain
>> procedures and to return that value to the procedures calling it. I
>> would appreciate any help you might be able to offer.
>>
>

```

> This is illegal:
>
> IDL> a={0:"this", 1:"is",2:"a",3:"funny",4:"test"}
>
> so the short answer is no. However, if you just want 4 numbered
> "slots", try an array:
>
> IDL> a={vals:['this','is','a','funny','test']}
> IDL> print,a.vals[1]
> is
>
> if you'd like the types of each slot to be different, try at a pointer
> array. If you'd like the number of "slots" to be variable, try a
> pointer to a pointer array.

```

I have a feeling Brian was looking for something a LOT simpler than this. :-)

I think he is confused between a specific *instance* of a structure and the *definition* of a structure. Especially because he has probably seen most structured both defined and populated with the same statement. For example, like this:

```
IDL> struct = {MYSTRUCT, a:3, b:5.4}
```

Here the variable struct is a specific *instance* of the structure whose definition is defined in IDL as "MYSTRUCT". For example, you can create another structure like this:

```

IDL> struct2 = {MYSTRUCT}
IDL> Help, struct2, /Structure
** Structure MYSTRUCT, 2 tags, length=8, data length=6:
A          INT          0
B          FLOAT        0.000000

```

The fields in the second structure are defined *by* the fields that appeared in the original structure definition statement. These are the "null" field values, if you like.

You can assign any value you like to them:

```
IDL struct2.a = 4
```

If you wish to create a structure definition without actually creating an instance of a structure, you put the structure definition in a file, like this:

```
PRO newstruct__define.pro
  struct = {NEWSTRUCT, a:0, b:0.0, c:fltarr(10)}
END
```

Then you can use the structure definition whenever you like:

```
IDL> var = {NEWSTRUCT}
IDL> var.a = 5
IDL> var.c = Findgen(10)*5
```

Cheers,

David
Cheers,

David

--

David W. Fanning, Ph.D.
Fanning Software Consulting
Phone: 970-221-0438, E-mail: david@dfanning.com
Coyote's Guide to IDL Programming: <http://www.dfanning.com/>
Toll-Free IDL Book Orders: 1-888-461-0155

Subject: Re: Defining Integer Field in a Structure
Posted by [Pavel A. Romashkin](#) on Thu, 01 Nov 2001 22:27:25 GMT
[View Forum Message](#) <> [Reply to Message](#)

I am getting confused by David and JD. Isn't all you want is this:

```
my_str = {field_1: 0s, field_2: 'Something_else'}
```

The only problem I see is that FIELD_1 is not going to be limited to 1...4 range, any valid integer (or any numeric type, for that matter) can be used to assign the value of FIELD_1. If you want it to be in 1...4 range, a check for that will have to be implemented.

When you pass the structure from one pro to another, they can read and change FIELD_1 as required, as long as the entire MY_STR was passed.

Cheers,
Pavel

Brian Bell wrote:

```
>
> Is there a way to define a field within a structure whose type is
> integer? In my case, I want the field to be any integer from 0 to 4.
```

> I know you can assign an actual integer to the field, but I don't want
> to do this because I want the field value to change within certain
> procedures and to return that value to the procedures calling it. I
> would appreciate any help you might be able to offer.
>
> Thanks,
>
> Brian

Subject: Re: Defining Integer Field in a Structure
Posted by [David Fanning](#) on Thu, 01 Nov 2001 22:58:29 GMT
[View Forum Message](#) <> [Reply to Message](#)

Pavel A. Romashkin (pavel.romashkin@noaa.gov) writes:

> I am getting confused by David and JD. Isn't all you want is this:

I'm going to have a class on How to Write Questions
that Normal People Can Answer. I should know better
than to guess what someone *really* means. :-)

But questions, many times, are symptoms of confusion,
not actual entities that should be answered specifically.
Any good technical support person knows that, but fewer
have time to do it right.

Cheers,

David

--

David W. Fanning, Ph.D.
Fanning Software Consulting
Phone: 970-221-0438, E-mail: david@dfanning.com
Coyote's Guide to IDL Programming: <http://www.dfanning.com/>
Toll-Free IDL Book Orders: 1-888-461-0155

Subject: Re: Defining Integer Field in a Structure
Posted by [Craig Markwardt](#) on Thu, 01 Nov 2001 23:36:38 GMT
[View Forum Message](#) <> [Reply to Message](#)

JD Smith <jdsmith@astro.cornell.edu> writes:

> Brian Bell wrote:

```

>>
>> Is there a way to define a field within a structure whose type is
>> integer? In my case, I want the field to be any integer from 0 to 4.
>> I know you can assign an actual integer to the field, but I don't want
>> to do this because I want the field value to change within certain
>> procedures and to return that value to the procedures calling it. I
>> would appreciate any help you might be able to offer.
>>
>
> This is illegal:
>
> IDL> a={0:"this", 1:"is",2:"a",3:"funny",4:"test"}
>
> so the short answer is no. However, if you just want 4 numbered
> "slots", try an array:
>
> IDL> a={vals:['this','is','a','funny','test']}
> IDL> print,a.vals[1]
> is
>
> if you'd like the types of each slot to be different, try at a pointer
> array. If you'd like the number of "slots" to be variable, try a
> pointer to a pointer array.

```

Following up on this diversion, it was in fact possible in IDL 5.2 and earlier to make structures whose tag names were not valid IDL variable names. For example, this statement was perfectly valid and executable:

```
x = create_struct('1',"this",'2',"is",'3',"funny",'4',"test ")
```

and it made exactly the structure you were expecting. However, I'm not sure it would be useful, and more importantly these types of antics are now forbidden by the IDL runtime.

Craig

--

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

Subject: Re: Defining Integer Field in a Structure
 Posted by [Logan Lindquist](#) on Fri, 02 Nov 2001 20:06:05 GMT
[View Forum Message](#) <> [Reply to Message](#)

"David Fanning" <david@dfanning.com> wrote in message

news:MPG.164b817bd6ec291d989743@news.frii.com...

> Pavel A. Romashkin (pavel.romashkin@noaa.gov) writes:

- > I'm going to have a class on How to Write Questions
- > that Normal People Can Answer.

I think the real problem is that people are just too different. In the phrasing of a question, terminology and the logic of the problem can be confused too easily.

- > But questions, many times, are symptoms of confusion,
- > not actual entities that should be answered specifically.
- > Any good technical support person knows that, but fewer
- > have time to do it right.

Everyone is busy these days, our lives continually find ways to move our interaction with the world on to new things. If the person genuinely needs help and the 'expert' genuinely wants to give assistance than there can be no bad questions and no bad answers. The communication received on both ends is the result of each persons intentions.

Regards,
Logan

Subject: Re: Defining Integer Field in a Structure
Posted by [Harvey Rarback](#) on Wed, 07 Nov 2001 01:02:30 GMT
[View Forum Message](#) <> [Reply to Message](#)

"David Fanning" <david@dfanning.com> wrote

- > If you wish to create a stucture definition without
- > actually creating an instance of a structure, you
- > put the structure definition in a file, like this:
- >
- > PRO newstruct__define.pro
- > struct = {NEWSTRUCT, a:0, b:0.0, c:fltarr(10)}
- > END

This is something that has always confused me. It seems like the variable "struct" is created (at least help will tell you that it exists). So, does IDL "actually create an instance of a structure" with this mechanism or not?

Thanks (in advance) for helping clear up my confusion.

--Harvey
