
Subject: Re: Convert a byte array to a structure?

Posted by [Pavel A. Romashkin](#) on Thu, 17 Jan 2002 21:39:08 GMT

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

I am sorry, today I can't read IDL with the usual ease.

Could you simply tell what you wanted to happen?, Like, not how you do it, but what do you want first, then - how to do it?

Pavel

Mike Fitzgibbon wrote:

```
>
> Does IDL have a simple and/or elegant way to convert a byte array
> into a structure?
>
> I could write a brute force approach like:
>
> proc arr_to_struct_1, arr, struct, idx
>   snipped ...n
```

Subject: Re: Convert a byte array to a structure?

Posted by [Bhautik Joshi](#) on Thu, 17 Jan 2002 22:57:34 GMT

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

```
> Does IDL have a simple and/or elegant way to convert a byte array
> into a structure?
```

```
> ..
```

```
> But is there a better, platform independent, way?
```

```
> (Windows NT/2K, Linux, and Solaris)
```

I wouldn't mind finding this out myself (its a nifty trick). However, from what I can see, a simple method is not possible? Please, correct me if I'm wrong.

There is a newsgroup post from a while back here (from JD Smith):

<http://groups.google.com/groups?q=converting+type+struct+group:comp.lang.idl-pvwave++group:comp.lang.idl-pvwave&hl=en&selm=3BAF6CAD.CFB19104%40astro.cornell.edu&num=3>

The structure of a structure in memory is something like:

```
typedef struct {          /* Reference to a structure */
  IDL_ARRAY *arr;         /* ^ to array block containing data */
  struct _idl_structure *sdef; /* ^ to structure definition */
} IDL_SREF;
```

So what you have is data, stored as bytes in an array, that is the data that fills out the structure fields, all concatenated together. Then there is the 'context', which defines the data types of the fields and

how many bytes each field takes up. So, if you could extract the data and context blocks (I haven't tried this before, but using pointers and stuff, just like good old C) and then play with it from there...

Another option would be to use the READS command and use explicitly formatted I/O to format the data into fields. I'm having a look at that now, and I'll get back to you on that one :)

cheers,
Bhautik

```
--  
/-----(\)-----\  
| nbj@imag.wsahs.nsw.gov.au | phone: 0404032617 |..|--\ -moo |  
| ICQ #: 2464537           | http://cow.mooh.org | |--| |  
|-----+-----\OO//| -----/  
| international           |  
| roast. my sanity has gone |  
| its lost forever         |  
|-----/
```

Subject: Re: Convert a byte array to a structure?
Posted by [Bhautik Joshi](#) on Thu, 17 Jan 2002 23:43:12 GMT
[View Forum Message](#) <> [Reply to Message](#)

> Does IDL have a simple and/or elegant way to convert a byte array
> into a structure?
> ..
> But is there a better, platform independent, way?
> (Windows NT/2K, Linux, and Solaris)
I've had no luck with READS and the FORMAT keyword either :(

Problem is this:

Lets say we have an arbitrary array of bytes, of length n. We also have a field, of n bytes length, that needs to be filled with that data as the concatenation of those values. As an example:

['ff'x, '01'x] -> 'ff01'x -> 65281

BYTE[2] -> intermediate step (?) -> FIX[1]

Would it be possible to do this with other size fields (like converting 4 bytes to a floating point number etc.)? Simply 'gluing' the elements of they byte array together, like 255*byte[0]+ byte[1] won't work with floating point numbers (?). I suspect the solution may involve pointers somehow, but I'm not sure how it might be done.

Then again, I might be missing something blindingly obvious, so please

let me know :)

cheers,
Bhautik

--

```
/-----(\_)------\
| nbj@imag.wsahs.nsw.gov.au | phone: 0404032617 |..|--\ -moo |
| ICQ #: 2464537           | http://cow.mooh.org | |--|   |
|-----+-----\OO//| -----/
| international           |
| roast. my sanity has gone |
| its lost forever         |
|-----/
```

Subject: Re: Convert a byte array to a structure?
Posted by [Craig Markwardt](#) on Fri, 18 Jan 2002 00:10:33 GMT
[View Forum Message](#) <> [Reply to Message](#)

mikef@grsgds6.lpl.arizona.edu (Mike Fitzgibbon) writes:

> Does IDL have a simple and/or elegant way to convert a byte array
> into a structure?
>
> I could write a brute force approach like:
... brute force approach which converts each data type by hand ..

I think you are stuck with the brute force approach. If you have an array of structures, then you may save some time by slicing out all of the field "A" tags first, then "B" and so on.

I believe that if you want to go "native" there are C functions to import C structures into IDL.

Good luck!
Craig

--

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

Subject: Re: Convert a byte array to a structure?
Posted by [sven](#) on Sat, 26 Jan 2002 08:06:37 GMT

> Problem is this:
> Lets say we have an arbitrary array of bytes, of length n. We also have
> a field, of n bytes length, that needs to be filled with that data as
> the concatenation of those values. As an example:
>
> ['ff'x,'01'x] -> 'ff01'x -> 65281
>
> BYTE[2] -> intermediate step (?) -> FIX[1]
>
> Would it be possible to do this with other size fields (like converting
> 4 bytes to a floating point number etc.)? Simply 'gluing' the elements
> of they byte array together, like 255*byte[0]+ byte[1] won't work with
> floating point numbers (?). I suspect the solution may involve pointers
> somehow, but I'm not sure how it might be done.

I am quite possibly missing something here, but why don't you use the standard typecasting functions like fix() and float() ?

```
IDL> b = ['66'xb,'66'xb,'8a'xb,'41'xb]
IDL> help,b
B      BYTE      = Array[4]
IDL> print,b
102 102 138 65
IDL> print,float(b,0)
17.3000
IDL> print,fix(b[0:1],0)
26214
```

Isn't that what they're there for?

--
/''' Sven Geier

c-OO This .signature is under the GNU public license. You may copy, use
\ > and distribute verbatim copys of it, as long as this notice and the
_v name of the author are left intact.
