Subject: Re: read a C written binary file with IDL
Posted by manodeep@gmail.com on Fri, 10 Feb 2012 04:56:23 GMT
View Forum Message <> Reply to Message

Hi,

This is because C pads the structure to produce alignments. Under 'normal' operations, you would expect MyStruct to be 20 bytes, however, if you do a sizeof(struct MyStruct), you will probably see that the size is 24. (And you can enable the warning for gcc by using the compile time option -Wpadded).

In general, the padding is compiler specific -- so there is no standard way of reading in those binary files into IDL/other codes. The best bet would be to write out the individual fields of the structure and then read them back into IDL.

HTH,
Manodeep

On Feb 9, 9:44 pm, bing999 <thibaultga...@gmail.com> wrote:
> Hi,
>
> I am having a problem with reading a C written binary file with IDL.
> It may come from differences of type definitions between C and IDL but
> I could not really figure out from Google...
>
> In C, it writes a structure containing the following variable types:
>
> struct MyStruct
> {
> int a;
> long long b;
> int c;
> float d;
>
> };
>
> Then, in IDL, I read this with:
>
> MyStruct = {$
> a : 0L, $
> b : 0LL, $
> c : 0L, $
> d : 0.0 $
> }
>
> openr, 1, filename, /SWAP_IF_BIG_ENDIAN

> readu, 1, MyStruct
> close, 1
>
> but this gives me wrong values.
>
> Did I miss something about the type conversion??
>
> If someone could please clarify this, it would really help!
> Thanks !
