

---

Subject: Re: DLM returning a pointer...

Posted by [Randall Skelton](#) on Tue, 24 Apr 2001 21:04:38 GMT

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

---

Wow... I am now well on my way with this little project thanks to Craig, Martin, Richard (and a few emails from Jim). You guys are great!

All of the suggestions given have been very helpful and I have incorporated many of them into my code. I must admit to have been rather confused regarding the need for allocating memory in IDL, but Craig's last email and code has resolved much of this.

Thanks again,  
Randall

On 24 Apr 2001, Craig Markwardt wrote:

> Randall, if I'm not mistaken, it's the job of the DBconset function to  
> allocate the memory for the DBcon structure, so in principle you don't  
> have to worry about that. What I am suggesting is to maintain a list  
> of such pointers in your C wrapper. So it could be something like the  
> below. Here I have made a static list of 10 DBcon pointers, which can  
> hold up to ten simultaneous connections. If you want to get smarter,  
> you could dynamically allocate memory if the number of simultaneous  
> connections is unknown and > 10.  
>  
> Of course I haven't done any error checking, and the dealing with IDL  
> <-> C translation isn't handled here. I'm just showing how a C  
> pointer can be converted to and from an integer value using a lookup  
> table (the table is dbcons[] in this case).  
>  
> The question of allocating memory is really a separate question  
> altogether. My personal feeling is that there is no harm to use  
> malloc(), but you can use the IDL\_MemAlloc() too. Both of these are  
> for raw memory. The other functions are for allocating IDL variables,  
> which carry along other baggage which you don't need here, and as you  
> say, IDL can trample on that data. Of course it will persist across  
> function calls. [ If your DLM is unloaded with  
> .IDL\_SUPER\_SESSION\_RESET or whatever I have no idea what happens. ]  
>  
> Good luck,  
> Craig  
>  
>  
[snip]

---