Subject: Re: Q: interfacing IDL to DLL
Posted by davidf on Wed, 28 Jul 1999 07:00:00 GMT
View Forum Message <> Reply to Message

Herbert H. Tsang (tsang@vcn.bc.ca) writes:

> Anyone has epxerience in interfacing IDL to C functions' DLL? Is
> there anything special I need to do in order to make a DLL that's works?

No, just write it correctly. :-)

Cheers,

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

Subject: Re: Q: interfacing IDL to DLL
Posted by davidf on Thu, 29 Jul 1999 07:00:00 GMT
View Forum Message <> Reply to Message

Peter Mason (menakkis@my-deja.com) writes:

> I would wager that David means that it's much easier than you might
> think, and that you should just go for it.

Wow, Peter! You thinking of getting into the book business?
I like your quirky style. And I would *definitely* buy the book!

Cheers,

David

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

Subject: Re: Q: interfacing IDL to DLL

Hi,

I feel to some extend the compiler (and esp. linker) You use might play an important role. I can give You a brief report of my experiences with Micro$$$ Visual Studio/VC 5.0:

1.) Suppose You want to create a WIN32 DLL (no MFCs)
2.) Declare the functions You want to export from the DLL as LONG DLLEXPORT

3.) Enclose the prototypes in #ifdef __cplusplus extern "C" .... #endif
4.) The parameter list of the exported functions must be of (LONG argc, void* argv[]) and nothing else; (#define LONG unsigned long)
5.) Take extreme care with the data formats of BYTE, INT, LONG and when passind IDL strings. I advice You strongly to leave the memory management either to the IDL part or to the C part. Don't mix.
6.) After compiling and linking the DLL have a look at its export table. Make sure that the functions You want to export are listed there without signatures (that's important and one reason why it is extremely difficult to interface to C++). Remember the exact names. Put the DLL where You like (in Your LIB search path might bre advantageous)
7.) Do the IDL CALL_EXTERNAL; make sure the path to the DLL is correct; use the exact name of the function to be called. As this name is being looked up in the DLL's export table make sure again the upper and lower cases are correct !
8.) Pray !

Good luck !

Arno R. Schleich, MS, MD

"Herbert H. Tsang" wrote:

> Anyone has epxerience in interfacing IDL to C functions' DLL? Is
> there anything special I need to do in order to make a DLL that's works?
>
> -- Herbert (tsang@vcn.bc.ca)

--
Functional Imaging Technologies GmbH
Siemensstr. 40/41
12247 Berlin
Germany

fon.: +49 (0)30 76 90 24 80
fax.: +49 (0)30 76 90 24 81

mailto:fit@functional-imaging.com
http://www.functional-imaging.com