Subject: Re: calling C++ from IDL - throwing exceptions
Posted by Craig Markwardt on Fri, 09 Feb 2001 21:50:27 GMT
View Forum Message <> Reply to Message

Ugo_DiGirolamo@InVision.iip.com writes:

- > I'm planning to use IDL as a front end for a C++ dll with call_external.
- > However I couldn't find any way of handling in IDL an exception thrown
- > from the C++ code.
- > I was wondering if anyone have an idea about it (also if to confirm
- > that it's impossible!) or if the DLM way would work better.
- > However, I've no real idea about what DLM is and I found really little
- > clue on it in the RSI documentation.
- > cheers

Greetings--

Craig

I believe that IDL is written entirely in C, so you won't get any C++ exception support from it.

Probably the best thing to do perform all exception handling in your wrapper routine.

There is nothing super-special about a DLM routine, except that it provides a standard mechanism for dynamic loading of code into IDL. While I haven't used it myself, I think it's probably the best solution for most applications these days.

Subject: Re: calling C++ from IDL - throwing exceptions Posted by Randall Skelton on Sun, 11 Feb 2001 14:40:00 GMT View Forum Message <> Reply to Message

You may want to consider getting Ron Kling's new book on calling C code from IDL as he does a nice job of describing DLMs and error trapping with them. I think you will end up needing to write C wrappers around your c++ in any case. Ron's book is available at: http://www.rlkling.com/

The DLM documentation for RSI is adequate as a reference for IDL type

structures and internals, but don't expect it to be much help when you are just starting out!

Randall

Ugo_DiGirolamo@InVision.iip.com writes:

- > I'm planning to use IDL as a front end for a C++ dll with call_external.
- > However I couldn't find any way of handling in IDL an exception thrown
- > from the C++ code.
- > I was wondering if anyone have an idea about it (also if to confirm
- > that it's impossible!) or if the DLM way would work better.
- > However, I've no real idea about what DLM is and I found really little
- > clue on it in the RSI documentation.
- > cheers

Subject: Re: calling C++ from IDL - throwing exceptions
Posted by Richard Younger on Mon, 12 Feb 2001 17:22:46 GMT
View Forum Message <> Reply to Message

Randall Skelton wrote:

>

- > You may want to consider getting Ron Kling's new book on calling C code
- > from IDL as he does a nice job of describing DLMs and error trapping with
- > them. I think you will end up needing to write C wrappers around your c++
- > in any case. Ron's book is available at: http://www.rlkling.com/

>

- > The DLM documentation for RSI is adequate as a reference for IDL type
- > structures and internals, but don't expect it to be much help when you are
- > just starting out!

>

> Randall

>

> Ugo_DiGirolamo@InVision.iip.com writes:

>

- >> I'm planning to use IDL as a front end for a C++ dll with call_external.
- >> However I couldn't find any way of handling in IDL an exception thrown
- >> from the C++ code.
- >> I was wondering if anyone have an idea about it (also if to confirm
- >> that it's impossible!) or if the DLM way would work better.
- >> However, I've no real idea about what DLM is and I found really little
- >> clue on it in the RSI documentation.
- >> cheers

C++ works fine with the DLMs. I've tried it out with the MS VC++ compiler (I know, I know, but I'm stuck with it for now), and it works just fine as long as you make sure to set the C calling convention

(cdecl) in the compiler. Now, with a DLM that is limited in scope, many of the _features_ of C++ aren't as useful, but your code that talks to IDL does just fine in C++. You can even write your IDL-callable functions as (static) class members.

I haven't really tried to write cross-language handlers though. In your C++ handler, you can call IDL message routines and pass back debugging variables, but I'm wondering just what you want to handle in IDL that was caused in C++, and how you're thinking of handling it. Just about all the things I can think of should be handled in their original language. If you want user feedback for the handling process, I know you can take a little input back into C++ with IDL_GetKbrd(), but I haven't used that for anything but "press any key" pauses, and it doesn't really return control to IDL. I suppose if you wanted to, you could print data or a menu to the log window and have the user select an option, but that seems to be going backwards.

Regardless, I agree that Ronn's book is a very helpful tutorial for getting started in DLMs, and includes lots of illuminating examples in C. If you can't quite get C++ to work, let me know.

Rich

--

Richard Younger Email: younger@ll.mit.edu MIT Lincoln Laboratory