Subject: Re: Building DLM on Windows 64-bit
Posted by penteado on Mon, 23 May 2011 19:20:38 GMT
View Forum Message <> Reply to Message

On May 23, 2:10 pm, David Fanning <n...@idlcoyote.com> wrote:
> Folks,
>
> After all these years, I thought it was about time I built a DLM
> file and called it from IDL. To this end, I've purchased Ronn's
> book Calling C and C++ from IDL and I've started in. I installed
> Visual Studio 2010 Express and I am trying to build the very
> first example in the book, simpleExample. (See snapshot_4.png
> below.)
>
> I have a project named simpleExample. This is an empty DLL
> project, as described in Ronn's book. I have installed
> in the source folder two C programs: IDL_Load.c and
> simpleExample.c. I have installed in the header folder
> two files: simpleExample.h and export.h. (I had to copy
> the latter over from the IDL directory because I could
> NOT get VS to find the darn thing! I tried adding it
> in the "Include Directories" list for both the VC++
> Directories and the C/C++ tabs in the project property
> sheet. I tried adding the file name both with and without
> quotes around it. All to no avail! See screenshot_1 and
> screenshot_2 in the examples below.)
>
> In the Linker Input tab in the project property page,
> I have included idl.lib (both with and without quotes, too).
> I have also specified the name of a Module Definition File,
> simpleExample.def, which is written like this:
>
> LIBRARY   simpleExample
> DESCRIPTION   'A simple example program'
> EXPORTS   IDL_Load   @1
>
> You can see the Linker Input tab in screenshot_3.png.
>
> When I try to build this project, I get the following errors.
>
> 1>------ Build started: Project: simpleExample, Configuration: Debug
> Win32 ------
> 1>C:\IDL\dlm\simpleExample.def(2): warning LNK4017: DESCRIPTION
> statement not supported for the target platform; ignored
> 1>   Creating library C:\Users\David\documents\visual studio 2010
> \Projects\simpleExample\Debug\simpleExample.lib and object C:\Users
> \David\documents\visual studio 2010\Projects\simpleExample\Debug
> \simpleExample.exp

> 1>IDL_Load.obj : error LNK2019: unresolved external symbol
> _IDL_SysRtnAdd referenced in function _IDL_Load
> 1>simpleExample.obj : error LNK2019: unresolved external symbol
> _IDL_GettmpLong referenced in function _simpleFunction
> 1>simpleExample.obj : error LNK2019: unresolved external symbol
> _IDL_Message referenced in function _simpleFunction
> 1>C:\Users\David\documents\visual studio 2010\Projects\simpleExample
> \Debug\simpleExample.dll : fatal error LNK1120: 3 unresolved externals
> ========== Build: 0 succeeded, 1 failed, 0 up-to-date, 0 skipped
> ==========
>
> It would appear I am missing an "export" file. Ronn mentions
> that this is needed for building on a Windows platform,
> but nowhere describes it or tells me how to create it.
> (At least in what I have read so far.) Is this what
> I need?
>
> I am at wit's end! Can anyone offer a way forward?
>
>   http://www.idlcoyote.com/misc/snapshot_1.png
>   http://www.idlcoyote.com/misc/snapshot_2.png
>   http://www.idlcoyote.com/misc/snapshot_3.png

These suggest to me that VS is not finding idl.lib because what is
given in the library directories is the file, not the directory. The
same for the include directory. I would check on the compiler command
generated by VS to confirm this.

Subject: Re: Building DLM on Windows 64-bit
Posted by David Fanning on Mon, 23 May 2011 20:37:42 GMT
View Forum Message <> Reply to Message

Paulo Penteado writes:

> These suggest to me that VS is not finding idl.lib because what is
> given in the library directories is the file, not the directory. The
> same for the include directory. I would check on the compiler command
> generated by VS to confirm this.

OK, I changed all "directory" names to names of directories
and not files. I no longer have to include export.h in
my project, so that probably is right.

I still get the same error messages, however, when I try
to build the project. I am still listing the complete
path to idl.lib in the Additional Dependencies path for
Linker Input. Both with and without quotes, but still

no joy.

  C:\Program Files\ITT\IDL\IDL81\bin\bin.x86_64\idl.lib

I even added idl.lib to the project, but that hasn't helped
either.

Where would I find the compiler command generated by VS?

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.idlcoyote.com/
Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: Building DLM on Windows 64-bit
Posted by penteado on Mon, 23 May 2011 21:56:33 GMT

On May 23, 5:37 pm, David Fanning <n...@idlcoyote.com> wrote:
> Where would I find the compiler command generated by VS?

In the version I am looking at (2008), it shows in the project
properties, under Configuration Propertties -> C/C++ -> Command Line,
and Configuration Properties -> Linker -> Command line.

Subject: Re: Building DLM on Windows 64-bit
Posted by David Fanning on Mon, 23 May 2011 22:00:58 GMT

Paulo Penteado writes:

> In the version I am looking at (2008), it shows in the project
> properties, under Configuration Propertties -> C/C++ -> Command Line,
> and Configuration Properties -> Linker -> Command line.

Well, it looks like it is finding idl.lib correctly.

/OUT:"C:\Users\David\documents\visual studio 2010\Projects\simpleExample

---

\Debug\simpleExample.dll" /INCREMENTAL /NOLOGO /DLL "C:\Program Files
\ITT\IDL\IDL81\bin\bin.x86_64\idl.lib" "kernel32.lib" "user32.lib"
"gdi32.lib" "winspool.lib" "comdlg32.lib" "advapi32.lib" "shell32.lib"
"ole32.lib" "oleaut32.lib" "uuid.lib" "odbc32.lib"
"odbccp32.lib" /DEF:"C:\IDL\dlm\simpleExample.def" /MANIFEST
/ManifestFile:"Debug
\simpleExample.dll.intermediate.manifest" /ALLOWISOLATION
/MANIFESTUAC:"level='asInvoker' uiAccess='false'" /DEBUG /PDB:"C:\Users
\David\documents\visual studio 2010\Projects\simpleExample\Debug
\simpleExample.pdb" /SUBSYSTEM:WINDOWS /PGD:"C:\Users\David\documents
\visual studio 2010\Projects\simpleExample\Debug
\simpleExample.pgd" /TLBID:1 /DYNAMICBASE /NXCOMPAT /MACHINE:X86
/ERRORREPORT:QUEUE

Still no joy building this DLL, though. :-(

Thanks,

David

--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.idlcoyote.com/
Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: Building DLM on Windows 64-bit
Posted by David Fanning on Tue, 24 May 2011 00:07:37 GMT
View Forum Message <> Reply to Message

David Fanning writes:

> Still no joy building this DLL, though. :-(

I switched my target from using idl.lib from the
bin.x86_64 directory to the bin.x86 directory
and the darn thing built beautifully!

Do you suppose this has something to do with using
VS 2010 *Express* with a 64-bit architecture?

More questions than answers at this point, but
at least I can turn the page in my book! :-)

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.idlcoyote.com/
Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: Building DLM on Windows 64-bit
Posted by Karl[1] on Tue, 24 May 2011 00:27:11 GMT
View Forum Message <> Reply to Message

On May 23, 6:07 pm, David Fanning <n...@idlcoyote.com> wrote:
> David Fanning writes:
>> Still no joy building this DLL, though. :-(
>
> I switched my target from using idl.lib from the
> bin.x86_64 directory to the bin.x86 directory
> and the darn thing built beautifully!
>
> Do you suppose this has something to do with using
> VS 2010 *Express* with a 64-bit architecture?
>
> More questions than answers at this point, but
> at least I can turn the page in my book! :-)
>
> Cheers,
>
> David
>
> --
> David Fanning, Ph.D.
> Fanning Software Consulting, Inc.
> Coyote's Guide to IDL Programming:http://www.idlcoyote.com/
> Sepore ma de ni thui. ("Perhaps thou speakest truth.")

In your first snapshot PNG file, note that your (active) platform is
Win32. You'll need to visit your configurations and add or use one
for 64-bit. I'm not sure VS Express supports it though.

Another clue is /MACHINE:X86 appearing in your command line listing
above. You'll want to see /MACHINE:X64 if you intend to build it for
64-bit, and that should be the case once you have your platform config
set for 64-bit.

Once you are really building for 64-bit, you can point back at the 64-bit idl.lib.

Another treat that you'll probably encounter is that you'll get to re-do all the include file paths, and other changes you made the first time over again for the 64-bit config, unless you were careful to do them for all configs the first time.

Visual Studio is a complex beast. But once you get it set up, it can be pretty sweet.

Karl

Subject: Re: Building DLM on Windows 64-bit
Posted by penteado on Tue, 24 May 2011 00:29:58 GMT
View Forum Message <> Reply to Message

On May 23, 9:07 pm, David Fanning <n...@idlcoyote.com> wrote:
> David Fanning writes:
> I switched my target from using idl.lib from the
> bin.x86_64 directory to the bin.x86 directory
> and the darn thing built beautifully!
>
> Do you suppose this has something to do with using
> VS 2010 *Express* with a 64-bit architecture?
>
> More questions than answers at this point, but
> at least I can turn the page in my book! :-)

Right. I had been using 32-bit. Maybe Morty's plight would help:

 http://groups.google.com/group/comp.lang.idl-pvwave/browse_thread/thread/a46c01113866688b/

Subject: Re: Building DLM on Windows 64-bit
Posted by David Fanning on Tue, 24 May 2011 00:38:42 GMT
View Forum Message <> Reply to Message

Paulo Penteado writes:

> Right. I had been using 32-bit. Maybe Morty's plight would help:
>
>   http://groups.google.com/group/comp.lang.idl-pvwave/browse_thread/thread/a46c01113866688b/

I'm sure it would, if I could just understand what it
means. Seems to be written in a language I don't fully
understand. :-)

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.idlcoyote.com/
Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: Building DLM on Windows 64-bit
Posted by Wout De Nolf on Tue, 24 May 2011 09:06:44 GMT
View Forum Message <> Reply to Message

On Mon, 23 May 2011 18:38:42 -0600, David Fanning <news@idlcoyote.com>
wrote:

> I'm sure it would, if I could just understand what it
> means. Seems to be written in a language I don't fully
> understand. :-)

Get the latest Windows SDK here:
http://tinyurl.com/33c3mga
You need the 64bit version: GRMSDKX_EN_DVD.iso

Set up VSExpress for 64bit:
http://msdn.microsoft.com/en-us/library/9yb4317s.aspx

Hope this helps.

Subject: Re: Building DLM on Windows 64-bit
Posted by David Fanning on Tue, 24 May 2011 18:09:10 GMT
View Forum Message <> Reply to Message

David Fanning writes:

> Paulo Penteado writes:
>
>> Right. I had been using 32-bit. Maybe Morty's plight would help:
>>

>>   http://groups.google.com/group/comp.lang.idl-pvwave/browse_t
hread/thread/a46c01113866688b/
>
> I'm sure it would, if I could just understand what it
> means. Seems to be written in a language I don't fully
> understand. :-)

Alright, after loading Microsoft SDK software practically
all night long, then re-installing Visual Studio 2010
this morning (another couple of hours), then configuring
everything to use the right libraries (thanks to Paulo
and Mort), I *finally* got simpleExample to build and
work in IDL! Hooray!

Of course, it doesn't do a damn thing, yet. But, hey,
one step at a time. :-)

Cheers,

David

P.S. Let's see, 36 hours to write a 10 line program. That's
a hell of a lot faster than writing a log colorbar using
IDL 8 graphics, I guess. ;-)

--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.idlcoyote.com/
Sepore ma de ni thui. ("Perhaps thou speakest truth.")