## Subject: Re: Using Windows IDL to access a UNIX database using DLM's Posted by Haje Korth on Tue, 13 Jun 2006 17:51:02 GMT

View Forum Message <> Reply to Message

## Ryan,

maybe I am blind but the file pgsqltoidl.c which contains IDL\_Load is not even contained in your compiler call, right? Once you include it you should be in good shape.

Cheers. Haje

cl -D\_DLL -DMSWIN -DWIN32 -D\_MT /nologo /I"C:\RSI\IDL62\external\include" /c "Z:\Database\_Access\PGSQLfunctions.c" /Fo"PGSQLfunctions\_4040\_ACEDESK3.obj" PGSQLfunctions.c link /out:"PgsglToIdl 4040 ACEDESK3.dll" /nologo /nodefaultlib /dll "PGSQLfunctions 4040 ACEDESK3.obj" /def: "PgsqlToldl 4040 ACEDESK3.def"

## rchughes@gmail.com wrote:

> Hey All,

>

- > I am relatively new to using IDL with very little knowledge of in-depth
- > computer stuff (compilers/linkers/etc.). I am trying to access a

"C:\RSI\IDL62\bin\bin.x86\idl32.lib" msvcrt.lib kernel32.lib

- > database using DLM's written by someone else (in C). I have spent way
- > too long trying to solve the problem by downloading different
- > compilers, looking into different linking issues and I still haven't
- > accomplished much. My major issue is trying to get IDL to load the DLM
- > and use it. I have added a IDL DLM PATH to my environment variables so
- > that when I type "help, /dlm" at the command line it shows that my DLM
- > is there. I then try to call a DLM it gives me an error that it can't
- > find the desired functions.

>

- > If you have had success using DLM's from Windows IDL please respond
- > because there is something that I'm doing incorrectly. I think that my
- > problem is that i am unable to generate the sharable library (dll file)
- > but I'm not exactly sure why. I have tried to follow the example of
- > for 'testmodule' given in the External Development Guide and that works
- > fine. I have followed build testmodule.pro as an example to create my
- > own build\_pgsql\_functions.pro file but it has problems with generating
- > the sharable library.

>

- > The error that I get is:
- > cl -D\_DLL -DMSWIN -DWIN32 -D\_MT /nologo
- > /I"C:\RSI\IDL62\external\include" /c
- > "Z:\Database Access\PGSQLfunctions.c"

```
> /Fo"PGSQLfunctions 4040 ACEDESK3.obj"
> PGSQLfunctions.c
> link /out: "PgsqlToIdl_4040_ACEDESK3.dll" /nologo /nodefaultlib /dll
"PGSQLfunctions_4040_ACEDESK3.obj" /def:"PgsqlToIdl_4040_ACEDESK3.def"
"C:\RSI\IDL62\bin\bin.x86\idl32.lib" msvcrt.lib kernel32.lib
> PgsqlToldl_4040_ACEDESK3.def : error LNK2001: unresolved external
> symbol IDL Load
> PgsqlToIdl_4040_ACEDESK3.lib : fatal error LNK1120: 1 unresolved
> externals
> LINK: fatal error LNK1141: failure during build of exports file
>
>
> Here is the code for build_pgsql_functions.pro:
> pro build_PGSQL_functions, VERBOSE=verbose
> ;Locating the PgsqlToldl files
> rootdir = 'Z:\'
> subdirs = ['Database Access']
> pgsqltoidl_dir = filepath(", ROOT_DIR=rootdir, SUBDIRECTORY=subdirs)
> dlm file = filepath('PgsqlToldl.dlm', ROOT DIR=rootdir,
> SUBDIRECTORY=subdirs)
  new dlm file = filepath('PgsqlToldl.dlm',
 ROOT_DIR=!make_dll.compile_directory)
>
  ;Build the PostGresSQL DLM in the !make_dll.compile_directory
> make dll, 'PGSQLfunctions', 'PgsglToldl', 'IDL Load',
> INPUT_DIR=pgsqltoidl_dir, $
  VERBOSE=verbose, SHOW ALL OUTPUT=verbose, /NOCLEANUP
>
  ;Copy the DLM file into the director with the sharable library
  file copy, dlm file, new dlm file, /OVERWRITE
> end
>
> Here is a sample of the pgsqlToldl.dlm file:
> MODULE PGSQLTOIDL
> DESCRIPTION IDL Interface to libpg (Postgres)
> VERSION 1.2beta
> SOURCE RS
> BUILD DATE May 17, 2004
> FUNCTION PGSQL CONNECT 1 1
> PROCEDURE PGSQL DISCONNECT 1 1 KEYWORDS
> FUNCTION PGSQL SELECT 3 3
> FUNCTION PGSQL COMMAND 2 2
> FUNCTION PGSQL_PQSTATUS 1 1
> FUNCTION PGSQL PQRESSTATUS 1 1
> FUNCTION PGSQL_PQCONNECTDB 1 1
> FUNCTION PGSQL PQSETDBLOGIN 7 7
> FUNCTION PGSQL PQSETDB 5 5
```

```
> ...
>
> Here is the code for the pgsqltoidl.c file:
> /* ANSI */
> #include <stdio.h>
> #include <stdlib.h>
> #include <math.h>
> #include <string.h>
>
> /* IDL */
> #include "idl export.h"
>
> /* Postgresql */
> #include "libpq-fe.h"
> /* Local */
> #include "PgsqlToldl.h"
> /* Define message codes and their corresponding printf(3) format
> strings.
  * Note that message codes start at zero and each one is one less than
  * previous one Codes must be monotonic and contiguous. Must match to
  * corrisponding entries in PgsqlToldl.h */
>
 static IDL_MSG_DEF msg_arr[] =
   { "PgsqlToldl_Error", "%NError: %s." },
   { "PgsqlToIdl_NOSTRINGARRAY", "%Nstring arrays not allowed %s"},
>
>
> /* The load function fills this message block handle with the opaque
> handle to
  * the message block used for this model. The other routines can then
  * to throw errors from this block. */
> IDL_MSG_BLOCK msg_block;
>
> int IDL Load(void)
> /* Define the message block */
> if (!(msg_block = IDL_MessageDefineBlock("PgsqlToldl",
> ARRLEN(msg_arr),
    msg_arr))) {
  return IDL FALSE;
  }
>
```

```
>
  /* Call the startup function to add the routines to IDL. */
>
>
  /* Routines */
  if (!PgsqlToldl_Startup()) {
   IDL_MessageFromBlock(msg_block, PgsqlToIdl_ERROR,
    IDL_MSG_RET, "Unable to initialize PostgreSQL to IDL interface");
  }
>
>
  return IDL_TRUE;
> }
>
> Here is my PgsqlToldl.def file that is needed for windows:
> LIBRARY "PGSQLfunctions"
> EXPORTS IDL_Load @1
>
> All the routines are in a different file called PgsqlToldlfunctions.c.
> I tried copying these functions into the PgsqlToldl.c file so that they
> compile at the same time but that doesn't work.
>
> FYI: the PgsqlToldl_Startup function is found in the
> PgsglToldlfunctions.c file, which is partly why I think it cannot load
> it.
>
> Any help determining where I am going wrong would be greatly
> appreciated.
>
> Ryan.
```