
Subject: SQL_mySQL reference problems
Posted by [aplatypus](#) on Tue, 07 Mar 2006 13:21:05 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi all ...

I've noted the other posts relating to the SQL_MYSQL reference programme. I have link errors under windows (mysql v5). The errors seem pretty basic for a "reference application".

There are two other points.

The {Setup} / {Automatic Setup} for MYSQL doesn't show the mysql path in te text box. Also I see a duplicate linkage/reference for "__main()". This looks like another linking issue, against mingw this time.

The critical bits of the output are below.

I began this thread to record, (a) how to 'fix' these issues; and (b) To encourage the "reference" samples can be clean builds from user feedback.

Cheers,

Will

Quote:__[output]_____

```
⋮
⋮
----- SQL_MySql ( MT MAIN GCC DEBUG DEBUG_FULL BLITZ WIN32 ) ( 5 / 5 )
Linking...
(option '-O 2' ignored)
__main: duplicate
-> used: libmingw32.a:gccmain.o:1
-> discarded: libgcc.a:__main.o:1
14 undefined symbol(s):
_mysql_affected_rows@4 (referenced from c:\bin\upp\out\mysql\mingw.debug_full.mt\mysql.o:1;
MySQLSession::IsOpen(void)
const; MySQLSession::IsOpen() const; sEmpNull(char const *); sEmpNull(char const*); ...)
_mysql_close@4 (referenced from c:\bin\upp\out\mysql\mingw.debug_full.mt\mysql.o:1;
MySQLSession::IsOpen(void) const; M
ySqlSession::IsOpen() const; sEmpNull(char const *); sEmpNull(char const*); ...)
_mysql_error@4 (referenced from c:\bin\upp\out\mysql\mingw.debug_full.mt\mysql.o:1;
MySQLSession::IsOpen(void) const; M
ySqlSession::IsOpen() const; sEmpNull(char const *); sEmpNull(char const*); ...)
_mysql_fetch_field_direct@8 (referenced from
```

. . . etc

Subject: Re: SQL_mySQL reference problems
Posted by [mirek](#) on Tue, 07 Mar 2006 21:24:19 GMT
[View Forum Message](#) <> [Reply to Message](#)

William wrote on Tue, 07 March 2006 08:21Hi all ...

I've noted the other posts relating to the SQL_MYSQL reference programe. I have link errors under windows (mysql v5). The errors seem pretty basic for a "reference application".

There are two other points.

The {Setup} / {Automatic Setup} for MYSQL doesn't show the mysql path in te text box. Also I see a duplicate linkage/reference for "__main()". This looks like another linking issue, against mingw this time.

The critical bits of the output are below.

I began this thread to record, (a) how to 'fix' these issues; and (b) To encourage the "reference" samples can be clean builds from user feedback.

```
----- SQL_MySql ( MT MAIN GCC DEBUG DEBUG_FULL BLITZ WIN32 ) ( 5 /
```

The sad issue is that MySQL in Win32 does not work with GCC compiler.

The problem is that MySQL does not provide GCC import libraries and there was not enough time to find out how to get it work.

I welcome any suggestions/research in this area.

BTW, U++ "uld" linker (shipped instead original ld in MinGW version) supports direct linking of .dll. Perhaps that is the path?

Mirek

Subject: Re: SQL_mySQL reference problems
Posted by [aplatypus](#) on Wed, 08 Mar 2006 09:50:55 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi Mirek,

I did some investigations on the U++ project. I does attempt to link against the SQL_MySql.dll file 'out of the box'.

The missing symbols are all present in the SQL_MySql.DEF file (in the /lib/ folder). I also took

pointed the linker to "_\MySQL Server 5.0/lib/debug/libmysql.dll" and optimised version "_\MySQL Server 5.0/lib/debug/libmysql.dll".

I'm not sure what could be 'wrong' other 'reference' programs I build work OK. The DEF file is 'correct'. A clue might be available in the mingW docs, concerning the order of the files in the build command, viz.:

<http://www.mingw.org/MinGWiki/index.php/undefined%20references>

Another line to pursue is ...

http://sunsite.mff.cuni.cz/MIRRORS/ftp.mysql.com/doc/en/Windows_client_compiling.html

There are other hints in the mingW materials, it might take some time to check them all. Also it looks like I have to figure how to control the build for U++. Is using ANT or MAKE an option?

Hopefully someone has got a mySQL programme working with mingW and will supply the missing link before I get there. Psst there is not need to prove me wrong on that *lol*

Cheers,
William

Subject: Re: SQL_mySQL reference problems
Posted by [aplatypus](#) on Wed, 08 Mar 2006 14:05:35 GMT
[View Forum Message](#) <> [Reply to Message](#)

recent update ...

The mingW reference material indicates this problem is to do with the sequence of #include files in the sources.

I have seen this problem a lot with the microsoft compilers, when the <windows.h> is included in the wrong place.

Unfortunately I don't (yet) get all the subtleties of the include dependencies. As an aside, it would be really useful for one of the Assits++ tools to manage dependencies (of various kinds).

When I look at the generated preprocessor output, indeed the mySQL.h files are included before the windows.h file. My difficulty so far appears to be the stuff and the order includes are used in <core.h>.

I am not precisely sure why this affects the linkage. The library and object file sequences used to be in the ld command line. My suspicion is that the 'problem' is a side-effect and if I could specify linkage explicitly it might work a-OK. Though I admit to guessing here.

That's what I know. If I'm on the wrong track, tell. File that links with a Microsoft compiler will link with mingw objects and libraries. The question might be about how I set-the-order for searching libraries during linking.

Nice chatting,

... Will.

Subject: Re: SQL_mySQL reference problems
Posted by [mirek](#) on Wed, 08 Mar 2006 21:06:32 GMT
[View Forum Message](#) <> [Reply to Message](#)

I do not think that include order is such a big deal - it would be indicated by C compiler failure (not linking failure).

I have tried simple google: "mysql mingw" and it, to my surprise, revealed

<http://www.dvrsol.com/programming.html>

I have added form the end of page mysql.a to my mingw lib directory, then added path to /lib/opt to my executable paths of build method (because that is added to PATH when executing the code as well) and MySql compiled with mingw.

Please report your success....

Now the question is - are we allowed to ship this library with U++? (legal issues).

Mirek

Subject: Re: SQL_mySQL reference problems
Posted by [mirek](#) on Wed, 08 Mar 2006 21:39:44 GMT
[View Forum Message](#) <> [Reply to Message](#)

Legal issues: I guess, we are not shipping the library, just "import library" created from .dll - I guess we definitely should be able to do so.

Means, it will be in the next release

Mirek

Subject: Re: SQL_mySQL reference problems

Posted by [aplatypus](#) on Sat, 11 Mar 2006 02:10:55 GMT

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

Hi ho Mirek

Thank you for the tip. I applied the fix indicated to make a new library file. So far it has not worked.

I can't find a way to explicitly say "link with this library" in the UPP project. I freely admit there are things I need to learn in that area. At the same time, I have a 'rant' about IDE-s in general. I really think they should consider "parsing" and reflecting good old nmake or ant scripts and stop attempting to re-invent sufficiently good wheels. (~sigh~)

I have yet to find a manual entry for how to specify a specific library.

As far as packaging goes, there are at least two clearly legal ways possible. One is to generate a library file and ship that with Upp. The alternative is to provide a "generate" script to make the library and run that script (when mysql is installed) during Upp installation.

I am no closer to my goal, and I don't really see the problem. Upp's mingw C++ command, finds the libmysql.dll file and still complains that it can't find the symbols.

I hope I will get more time to look for another approach next week.

Cheers,
William

Subject: Re: SQL_mySQL reference problems

Posted by [mirek](#) on Sat, 11 Mar 2006 06:42:27 GMT

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

You add libraries using package organizer - just select MySQL, right click on right pane and choose Add library.

As for "reinventing good wheels", other alternatives do not support modularity and BLITZ is impossible with make....

In fact, modularity was the reason why we started TheIDE. And it works just as we wanted...

Mirek

Subject: Re: SQL_mySQL reference problems

Posted by [aplatypus](#) on Sat, 11 Mar 2006 11:38:53 GMT

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

Once more Mirek,

Many thanks. As the saying goes, I had some good news and some not so good news. Putting the new library removes the unresolved symbols link errors.

This still leaves the SQL_MySql project with a different error that that doesn't make sense to me.

Quote: Linking...

```
c++ -static -mwindows -mconsole -o "c:\bin\upp\out\MINGW.Debug_full.Mt\SQL_MySql.exe"
-ggdb -L"C:\
bin\upp\mingw\lib" -L"C:\bin\upp\SDL-1.2.9\lib" -L"C:\bin\MySQL\MySQL Server 5.0\bin\" -WI,-O,
2 "c:/bin/upp/out/SQL_MySql/MINGW.Debug_full.Main.Mt/main.o"
"c:/bin/upp/out/MySql/MINGW.Debug
_full.Mt/MySql.o" "c:/bin/upp/out/Sql/MINGW.Debug_full.Mt/$blitz.o" "c:/bin/upp/out/plugin/z/
MINGW.Debug_full.Mt/zlibupp.o" "c:/bin/upp/out/plugin/z/MINGW.Debug_full.Mt/adler32.o" "c:/bin
/upp/out/plugin/z/MINGW.Debug_full.Mt/compress.o"
"c:/bin/upp/out/plugin/z/MINGW.Debug_full.Mt
/crc32.o" "c:/bin/upp/out/plugin/z/MINGW.Debug_full.Mt/deflate.o" "c:/bin/upp/out/plugin/z/MIN
GW.Debug_full.Mt/gzio.o" "c:/bin/upp/out/plugin/z/MINGW.Debug_full.Mt/infblock.o" "c:/bin/upp/
out/plugin/z/MINGW.Debug_full.Mt/infcodes.o" "c:/bin/upp/out/plugin/z/MINGW.Debug_full.Mt/inff
ast.o" "c:/bin/upp/out/plugin/z/MINGW.Debug_full.Mt/inflate.o" "c:/bin/upp/out/plugin/z/MINGW.
Debug_full.Mt/inftrees.o" "c:/bin/upp/out/plugin/z/MINGW.Debug_full.Mt/inftutil.o" "c:/bin/upp/
out/plugin/z/MINGW.Debug_full.Mt/trees.o"
"c:/bin/upp/out/plugin/z/MINGW.Debug_full.Mt/uncompr
.o" "c:/bin/upp/out/plugin/z/MINGW.Debug_full.Mt/zutil.o" "c:/bin/upp/out/Core/MINGW.Debug_ful
l.Mt/$blitz.o" "c:/bin/upp/out/Core/MINGW.Debug_full.Mt/OL_Set.o" -WI,--start-group -lUppMySQ
L.lib -ladvapi32 -lshell32 -lwinmm -lole32 -loleaut32 -luuid -WI,--end-group
c++.exe: no input files
Exitcode: 1
```

There were errors. (0:00.59)

The bit that comounds me is the "no input files" message. When i looked in the relevant Upp/out folders all the files I would have expected to find and use for input are there.

Does someone have an idea what "input" is missing?

Slow and steady progress

.../William

Subject: Re: SQL_mySQL reference problems
Posted by [aplatypus](#) on Sat, 11 Mar 2006 12:45:31 GMT
[View Forum Message](#) <> [Reply to Message](#)

Greetings friends ...

I have made one further discovery -- I suspect that the 'next' error will be readily resolved by a

U++ aficionado.

The new library I made was in a directory with spaces in the name. When I moved this file to folder "c:\bin\mysql\UppMySql.a" from "c:\bin\mysql\C:\bin\MySQL\MySQL Server 5.0\bin\" that problem resolved itself. This is the new library described in Mirek's post about mingW and mySQL.

Also the file must use the "libUppMysql.a" extension. Earlier I made the error of calling it with a ".lib"-extension. The "lib"-part and the ".a"-extension are required.

However there is a further linking problem concerning the "_WinMain" entry point. I suspect that this will be a mingW issue. It seems so fundamental, I wondered if someone has already found a way to do this already.

```
cmd output:___main: duplicate
-> used:    libmingw32.a:gccmain.o:1
-> discarded: libgcc.a:___main.o:1
1 undefined symbol(s):
_WinMain@16 (referenced from libmingw32.a:main.o:1; ___main; .text)
linking aborted due to fatal errors
collect2: ld returned 1 exit status
```

This output is from the windows command line. The output window in TheIDE only said, ...

```
TheIDE output:c++.exe: no input files
Exitcode: 1
```

I admit to being mildly confused here. Other GUI reference programs work OK. The SQL_MySql reference looks like it is a console program, may be that is the 'catch'.

In the meantime, I found this pearl. Concerning how ld works for mingW.

web search: <http://www.cygwin.com/ml/cygwin/2004-03/msg00194.html>

To resolve _WinMain@16 with a symbol from your library, it needs to come *after* the library that needs it (/usr/i686-pc-mingw32/lib/libmingw32.a).

Unfortunately (for you), gcc/g++ puts all the default libraries *last* when it calls ld. You need to override that order.

.. link your app using the following command:

```
g++ -mwindows -mno-cygwin -o test.exe test.o -lmingw32 -L./ -lmaintest
```

There are some slow chips at the solution. Wiser heads are welcome. Is there some way to

specify the exact commands to build with minGW? Why do other reference projects work with minGW?

I know everyone loves their own creations. I sincerely recommend that anything called a "reference" program include a comment certifying the "platforms it has been tested on", and the platforms it has not tested for if applicable.

It is an excellent system. Unfortunately my use for U++ at the moment is with mingGW and SQL on Windows. I could go to the microsoft compilers -- Can anyone tell me if they have run the mysql with Microsoft c++? This is such a small problem -- keep up the good work!!

Aloha,
Will.

Subject: Re: SQL_mySQL reference problems
Posted by [mirek](#) on Sat, 11 Mar 2006 14:29:07 GMT
[View Forum Message](#) <> [Reply to Message](#)

```
cmd output:___main: duplicate
-> used: libmingw32.a:gccmain.o:1
-> discarded: libgcc.a:___main.o:1
1 undefined symbol(s):
_WinMain@16 (referenced from libmingw32.a:main.o:1; _main; .text)
linking aborted due to fatal errors
collect2: ld returned 1 exit status
```

Well, this means that you are building GUI program, but your main function is "main" (or CONSOLE_APP_MAIN).

Use GUI_APP_MAIN instead (that contains "WinMain" inside).

"no input files" - I am clearly puzzled by the error.

I am really sorry for causing troubles there. I hope that in the near future (hopefully as soon as next version next week) MySQL/MinGW combo will work out of box.

Mirek

Subject: Re: SQL_mySQL reference problems
Posted by [aplatypus](#) on Sun, 12 Mar 2006 11:50:39 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hy there Mirek

Thanks for your comments, I am not sure of their meaning though. The SQL_MySql demo is a console program. Why do I want to use the "GUI_APP_MAIN" macro here? My question is this, How does a console program come to ask for a '_WinMain' entry point?

And of course, How do I make the console program that isn't looking for the '_WinMain' entry point?

The error: "No input files.", is the only message in TheIDE's output window from the link-command asking for a '_WinMain' entry.

If the new release will correct the issue, then perhaps I will wait for that. It looks to me as if the issue is not with the program as much as in the way the mysql builds are done under minGW.

If you run a build of the SQL_MySql project with a library file made from the DLL with the minGW dlltool program ...

Quote:

```
c:\> dlltool.exe --dllname libmySQL.dll --output-lib libUppMySQL.a -k
```

That will reproduce the "no input" error message. Or if it doesn't, I can go over everything to find my mistake.

Thanks for the support, the tips are helping, there is slow progress. And I'm learning more about Ultimate.

Cheers,
Will.

Subject: Re: SQL_mySQL reference problems
Posted by [aplatypus](#) on Mon, 27 Mar 2006 13:40:41 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi all,

Very good news -- Just as promised the .603 version builds the mySQL reference "out of the box". Thank you very much.

To connect with a database you will probably need to specify a username and password. I put the mysql database in to 'see' how things work.

```
main.cpp  
    if(!session.Connect( "mySelf", "passw", "mysql", "localhost" )) {
```

I must say I am a little befuddled about `_why_` it works now with mySQL v5 when I haven't done anything with mySQL libraries. (I un-did the earlier patches).

The only reason that interests me is because what happens when I want to link another 3rd party library and similar issues pop-up? Fortunately that's something I can cope with later in life.

Well done and thanks again.

Regards,
William

Subject: Re: SQL_mySQL reference problems
Posted by [mirek](#) on Mon, 27 Mar 2006 15:34:20 GMT
[View Forum Message](#) <> [Reply to Message](#)

William wrote on Mon, 27 March 2006 08:40

```
main.cpp
    if(!session.Connect( "mySelf", "passw", "mysql", "localhost" )) {
```

I am no MySql expert so a direct question: Do you propose to put above to reference/SQL_MySql as it is more likely work out of box?

Mirek

Subject: Re: SQL_mySQL reference problems
Posted by [aplatypus](#) on Tue, 28 Mar 2006 00:31:30 GMT
[View Forum Message](#) <> [Reply to Message](#)

```
main.cpp

    if(!session.Connect( "mySelf", "passw", "mysql", "localhost" ))
    {
    }
```

Mirek

I am no MySql expert so a direct question: Do you propose to put above to reference/SQL_MySql as it is more likely work out of box?

Hi Mirek

That's a good question ... The mySQL installation will have some username and password (or 'should') so the code will not work "out of the box" unless the correct username and password is used.

The "mysql" database name is always used.

I can suggest a couple of lines of comment saying to

'substitute your username and password in the code below.'
The same situation should apply to the other database examples. I strongly advise using a test username (not the root username) in an example programme.

Cheers,
Will
