Subject: U++ and MinGW 4.7 (-std=c++11) issues
Posted by Ptomaine on Sat, 21 Jul 2012 19:40:52 GMT
View Forum Message <> Reply to Message

First, I want to say that U++ compiles very well when used with MinGW 4.7 even when using c++11 compiler option for U++'s native libraries. But there are some issues pertaining to compilation.

To resolve these issues I've had to do the following changes:

1. Right before inclusion of the <shlobj.h> file, the CY definition must be undefined:

```
#undef CY
#include <shlobj.h>
```

I found two files that need to be fixes this way: App.cpp and Path.cpp

2. WINVER definition must be corrected in <Core.h>:
from
```
 #ifdef COMPILER_MINGW
  #define WINVER 0xFFFF
 #endif
```

to
```
 #if defined(COMPILER_MINGW) && !defined(WINVER)
  #define WINVER 0xFFFF
 #endif
```

3. The compiler asks to include <winsock2.h> before <windows.h>, so the definition block with <winsock2.h> need to be placed higher than the first <windows.h> inclusion in the <Core.h> file.

4. The line in the App.cpp source file must be chaged:
from
```
if (int(ShellExecuteW(NULL, L"open", wurl, NULL, L".", SW_SHOWDEFAULT)) <= 32) {
```

to
```
if (reinterpret_cast<std::ptrdiff_t>(ShellExecuteW(NULL, L"open", wurl, NULL, L".",
SW_SHOWDEFAULT)) <= 32) {
```

5. In the Defs.h header file, the code must be changed:
from
```
#ifdef PLATFORM_WIN32
inline bool IsNaN(double d)       { return _isnan(d); }
```

to
```
#ifdef PLATFORM_WIN32
inline bool IsNaN(double d)       { return std::isnan(d); }
```

So, if I make all these changes, the compilation goes smooth and fine even with c++11 compiler option.

Subject: Re: U++ and MinGW 4.7 (-std=c++11) issues
Posted by Sender Ghost on Sun, 22 Jul 2012 06:08:19 GMT
View Forum Message <> Reply to Message

Hello, Arlen.
Ptomaine wrote on Sat, 21 July 2012 21:40First, I want to say that U++ compiles very well when used with MinGW 4.7 even when using c++11 compiler option for U++'s native libraries. But there are some issues pertaining to compilation.
I saw diagnostic messages you mentioned, but for MinGW-w64 builds only. They not related to "-std=c++11" compiler option. I saw them even for 4.6.3 or some below versions, not only for 4.7.1. For example, the nuwen.net MinGW builds haven't such issues.

But I could agree with WINVER define changes (because it is possible to define WINVER before including U++ headers or as compiler option).

Will see what the main developers say.

Subject: Re: U++ and MinGW 4.7 (-std=c++11) issues
Posted by Ptomaine on Sun, 22 Jul 2012 19:13:45 GMT
View Forum Message <> Reply to Message

Hello, Mirek.

Yes, you're right! I've used the MinGW package that can compile x86-64 and x86-32 binaries using just the -m64 and -m32 compiler flags. But, by default, the compiler produce x86-64 binaries since I've downloaded the x64 compiler version from this URL.

So, anyway, why not to make some x64 related fixes to let compile U++ in x86-64 mode just smooth and flawless?

Subject: Re: U++ and MinGW 4.7 (-std=c++11) issues
Posted by mirek on Mon, 23 Jul 2012 07:04:46 GMT
View Forum Message <> Reply to Message

Applied.

Mirek

Subject: Re: U++ and MinGW 4.7 (-std=c++11) issues
Posted by unodgs on Mon, 23 Jul 2012 11:50:26 GMT
View Forum Message <> Reply to Message

Thanks! I had similar changes but in another branch Speaking of MingGW64. I'm unable to start debugger in the new experimental mode. Ide says 'Failded to run app'. I have gdb 7.4.1 and python 2.7.3 (on PATH). Does it only work in 32 bit mode?

Subject: Re: U++ and MinGW 4.7 (-std=c++11) issues
Posted by mirek on Mon, 23 Jul 2012 11:56:10 GMT
View Forum Message <> Reply to Message

unodgs wrote on Mon, 23 July 2012 07:50Thanks! I had similar changes but in another branch Speaking of MingGW64. I'm unable to start debugger in the new experimental mode. Ide says 'Failded to run app'. I have gdb 7.4.1 and python 2.7.3 (on PATH). Does it only work in 32 bit mode?

I dare to say that you are the first testing new debugger with mingw.

Mirek

Subject: Re: U++ and MinGW 4.7 (-std=c++11) issues
Posted by unodgs on Mon, 23 Jul 2012 13:08:26 GMT
View Forum Message <> Reply to Message

That would explain that I'll try to investigate on my own then.

Subject: Re: U++ and MinGW 4.7 (-std=c++11) issues
Posted by mdelfede on Mon, 23 Jul 2012 15:58:11 GMT
View Forum Message <> Reply to Message

unodgs wrote on Mon, 23 July 2012 13:50Thanks! I had similar changes but in another branch Speaking of MingGW64. I'm unable to start debugger in the new experimental mode. Ide says 'Failded to run app'. I have gdb 7.4.1 and python 2.7.3 (on PATH). Does it only work in 32 bit mode?

I use it on gcc on ubuntu 64 bit and it works well... never tested on mingw, either.

I don't have a mingw setup here.... could you please try to "debug the debugger" with the old one to see why it can't start ?
Usually I have the "installed" ide named "theide" and the testing one named (automatically...) ide, so you can start theide in debug mode with different settings than the installed one, so it's quite easy to debug it.
If you add a couple of DLOG in Gdb_MI2.cpp :

```
MIValue Gdb_MI2::MICmd(const char *cmdLine)
{
   DLOG("CMDLINE : " << cmdLine);
......
```

and

```
MIValue Gdb_MI2::ParseGdb(String const &output, bool wait)
{
   DLOG("OUTPUT : " << output);
.......
```

and send me the ide log file, I could probably find the problem.

Ciao

Max
