
Subject: Re: U++/Win32 is now using CLANG toolchain

Posted by [Tom1](#) on Wed, 25 Mar 2020 07:57:20 GMT

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

Hi Mirek,

Adding the libs manually

when: WIN32

Libraries: ../lib/libcmt.lib ../lib/liboldnames.lib

introduces a different issue:

Linking...

lld-link: error: duplicate symbol: atexit

>>> defined at ../crt\crtexe.c:438

>>> C:\upp-svn\bin\clang\x86_64-w64-mingw32\lib\crt2.o

>>> defined at libcmt.lib(utility.obj)

lld-link: error: duplicate symbol: __dyn_tls_init

>>> defined at d:\agent_work\4\s\src\vctools\crt\vcstartup\src\tls\tlsdyn.cpp:87

>>> libcmt.lib(tlsdyn.obj)

>>> defined at ../crt\tlssup.c:77

>>> libmingw32.a(lib64_libmingw32_a-tlssup.o)

lld-link: error: duplicate symbol: __dyn_tls_init_callback

>>> defined at libcmt.lib(tlsdyn.obj)

>>> defined at ../crt\tlssup.c:103

>>> libmingw32.a(lib64_libmingw32_a-tlssup.o)

lld-link: error: duplicate symbol: _onexit

>>> defined at d:\agent_work\4\s\src\vctools\crt\vcstartup\src\utility\utility.cpp:256

>>> libcmt.lib(utility.obj)

>>> defined at ../crt\ucrtbase_compat.c:86

>>> libmsvcrt.a(lib64_libucrt_extra_a-ucrtbase_compat.o)

lld-link: error: duplicate symbol: _tls_index

>>> defined at ../crt\tlssup.c:35

>>> libmingw32.a(lib64_libmingw32_a-tlssup.o)

>>> defined at libcmt.lib(tlssup.obj)

lld-link: error: duplicate symbol: _tls_start

>>> defined at ../crt\tlssup.c:41

>>> libmingw32.a(lib64_libmingw32_a-tlssup.o)

>>> defined at libcmt.lib(tlssup.obj)

lld-link: error: duplicate symbol: _tls_end

>>> defined at ../crt\tlssup.c:42

>>> libmingw32.a(lib64_libmingw32_a-tlssup.o)

>>> defined at libcmt.lib(tlssup.obj)

```
lld-link: error: duplicate symbol: __xl_a
>>> defined at ../crt\tlssup.c:44
>>>      libmingw32.a(lib64_libmingw32_a-tlssup.o)
>>> defined at libcmtd.lib(tlssup.obj)
```

```
lld-link: error: duplicate symbol: __xl_z
>>> defined at ../crt\tlssup.c:45
>>>      libmingw32.a(lib64_libmingw32_a-tlssup.o)
>>> defined at libcmtd.lib(tlssup.obj)
```

```
lld-link: error: duplicate symbol: _tls_used
>>> defined at ../crt\tlssup.c:47
>>>      libmingw32.a(lib64_libmingw32_a-tlssup.o)
>>> defined at libcmtd.lib(tlssup.obj)
```

```
lld-link: error: could not open 'liblibvcruntime.a': No such file or directory
lld-link: error: could not open 'liblibucrt.a': No such file or directory
lld-link: error: could not open 'libLIBCMT.a': No such file or directory
lld-link: error: could not open 'libOLDNAMES.a': No such file or directory
clang-10: error: linker command failed with exit code 1 (use -v to see invocation)
```

There were errors. (0:00.50)

So adding the libraries yields duplicate symbols while adding yet more unsatisfied dependencies. Unfortunately, they still do not satisfy the requirement for the strange 'libLIBCMT.a' and 'libOLDNAMES.a'. (I also tried to copy the original files to these names, but that did not work either.)

I wonder if I should create a couple of dummy libraries with no public symbols and give them names 'libLIBCMT.a' and 'libOLDNAMES.a'...

Any suggestions to try?

Best regards,

Tom

[EDIT] Well, those dummy empty libraries with names 'libLIBCMT.a' and 'libOLDNAMES.a' did not help either.
