Subject: Clang / lld
Posted by Novo on Fri, 19 Apr 2019 20:50:15 GMT
View Forum Message <> Reply to Message

mirek wrote on Sat, 13 April 2019 07:07These are things I would like to have solved, but do not have resources to:

- Fix uldar - old U++ linker for mingw by Tomas Rylek, that is superfast (compared to ld.exe supplied with mingw), but unfortunately stopped working around mingw 2012 version. This would make mingw much more attractive for us (well, there is still that poor TLS handling, but maybe that will get solved over time too).

I personally would take a look at LLD first. Gold linker is supposed to be several times faster than ld, but it doesn't support PE/COFF. LLD is supposed to be several times faster than the gold linker, and it supports PE/COFF. So, there is no need to reinvent the wheel.

Subject: Re: Possible new package or overhaul proposals for U++ in 2019
Posted by mirek on Fri, 19 Apr 2019 22:56:54 GMT
View Forum Message <> Reply to Message

Novo wrote on Fri, 19 April 2019 22:50mirek wrote on Sat, 13 April 2019 07:07These are things I would like to have solved, but do not have resources to:

- Fix uldar - old U++ linker for mingw by Tomas Rylek, that is superfast (compared to ld.exe supplied with mingw), but unfortunately stopped working around mingw 2012 version. This would make mingw much more attractive for us (well, there is still that poor TLS handling, but maybe that will get solved over time too).

I personally would take a look at LLD first. Gold linker is supposed to be several times faster than ld, but it doesn't support PE/COFF. LLD is supposed to be several times faster than the gold linker, and it supports PE/COFF. So, there is no need to reinvent the wheel.

Absolutely! Anyway, the goal is to integrate it with mingw. Will that be possible?

Subject: Re: Possible new package or overhaul proposals for U++ in 2019
Posted by Novo on Fri, 19 Apr 2019 23:05:12 GMT
View Forum Message <> Reply to Message

mirek wrote on Fri, 19 April 2019 18:56Absolutely! Anyway, the goal is to integrate it with mingw. Will that be possible?

I think, yes, this should be possible. I'll try to do that.

Subject: Re: Possible new package or overhaul proposals for U++ in 2019
Posted by mirek on Tue, 23 Apr 2019 14:38:58 GMT

View Forum Message <> Reply to Message

Novo wrote on Sat, 20 April 2019 01:05mirek wrote on Fri, 19 April 2019 18:56Absolutely!
Anyway, the goal is to integrate it with mingw. Will that be possible?

I think, yes, this should be possible. I'll try to do that.

I was going through web investigating this and just to clarify: Even better would be using clang and its linker...

It looks like it should be possible to combine mingw libraries with clang toolchain...

I am really surprised that something like that is not already somewhere for download.

Mirek

Subject: Re: Possible new package or overhaul proposals for U++ in 2019
Posted by Novo on Tue, 23 Apr 2019 15:29:51 GMT

View Forum Message <> Reply to Message

mirek wrote on Tue, 23 April 2019 10:38
I was going through web investigating this and just to clarify: Even better would be using clang and its linker...

It looks like it should be possible to combine mingw libraries with clang toolchain...

I am really surprised that something like that is not already somewhere for download.

Mirek
AFAIK, no one is allowed to distribute mingw. That is the problem ...
clang can be combined with Microsoft SDK as well ... In this case it is called clang-cl ...
I'll try to check clang with mingw later. I need to work on my own stuff now ...

Subject: Re: Possible new package or overhaul proposals for U++ in 2019
Posted by mirek on Wed, 24 Apr 2019 07:04:08 GMT

View Forum Message <> Reply to Message

Novo wrote on Tue, 23 April 2019 17:29mirek wrote on Tue, 23 April 2019 10:38
I was going through web investigating this and just to clarify: Even better would be using clang and its linker...

It looks like it should be possible to combine mingw libraries with clang toolchain...

I am really surprised that something like that is not already somewhere for download.

Mirek
AFAIK, no one is allowed to distribute mingw. That is the problem ...

Nonsense. Mingw (those parts except the GNU compiler) is public domain.

Quote:
clang can be combined with Microsoft SDK as well ... In this case it is called clang-cl ...
I'll try to check clang with mingw later. I need to work on my own stuff now ...

OK, so I have tried... Clang 8.0.0. does not seem to work with mingw.

However, 7.0.0 does, after fixing some issues here and there, with ld.exe. That would be more than enough, 7.0.0 is late 2018 release.

The only problem is that I am so far unable to make ldd.exe work. I am able to compile with "-fuse-ld=lld" and everything seems fine, unfortunately resulting .exe just crashes. So it all leads to nowhere - complete recompile of GUI app is 12 seconds for clang and 17 seconds for that damned slow ld.exe to link results....

I am attaching .bm....

Maybe it would be better to try in Microsoft SDK mode? Perhaps mingw Win32 libraries and headers would import to clang-cl?

File Attachments

1) CLANG.bm, downloaded 357 times

Subject: Re: Possible new package or overhaul proposals for U++ in 2019
Posted by cbpporter on Wed, 24 Apr 2019 07:50:39 GMT
View Forum Message <> Reply to Message

A couple of years ago I did manage to compile normal C++ with windows precompiled clang and Windows SDK.

It had a few bugs though, mainly the feature you use in U++ to check for leaks, did not work.

Subject: Re: Possible new package or overhaul proposals for U++ in 2019
Posted by mirek on Wed, 24 Apr 2019 08:51:08 GMT
View Forum Message <> Reply to Message

cbpporter wrote on Wed, 24 April 2019 09:50A couple of years ago I did manage to compile normal C++ with windows precompiled clang and Windows SDK.

Yeah, thats simple. I can compile with mingw SDK as well. The only problem is that damned linker, which is the whole point of trying.

If it had a faster linker, "mingw" version would be 1000% more attractive. Hell, I might even use it on windows for production development...

Mirek

Subject: Re: Possible new package or overhaul proposals for U++ in 2019
Posted by mirek on Wed, 24 Apr 2019 08:56:26 GMT
View Forum Message <> Reply to Message

I have tried to ask here:

https://stackoverflow.com/questions/55825807/using-llvm-link er-lld-with-mingw

Subject: Re: Possible new package or overhaul proposals for U++ in 2019
Posted by mirek on Wed, 24 Apr 2019 12:45:27 GMT
View Forum Message <> Reply to Message

I have tried snapshot of clang 9.0.0 and it actually works!

Interestingly, even those TLS variables are properly optimized (unlike mingw gcc). Linking GUI app now takes 0.2s.

There are some problems with TLS variables in inline code, but fixing those is possible. Also, <float.h> needed a bit of fixing.

The only problem seems to be that clang is 1.5GB beast. I hope it will be possible to prune it a bit...

Mirek

Subject: Re: Possible new package or overhaul proposals for U++ in 2019
Posted by mirek on Thu, 25 Apr 2019 05:57:21 GMT
View Forum Message <> Reply to Message

Even better: I have tried to simply replace ld.exe in mingw with clang's (9.0.0) lld.exe and everything works!

Here are times:

with original ld.exe

C:\upp\out\examples\MINGWx64.Debug.Debug_Full.Gui\AddressBoo k.exe (39524957 B) linked in (0:19.68)

with lld.exe

C:\upp\out\examples\MINGWx64.Debug.Debug_Full.Gui\AddressBoo k.exe (36620800 B) linked in (0:00.27)

Thank you Novo for pointing me in the right direction. This is really significant step ahead!

Subject: Re: Possible new package or overhaul proposals for U++ in 2019
Posted by Novo on Thu, 25 Apr 2019 14:14:30 GMT

View Forum Message <> Reply to Message

mirek wrote on Thu, 25 April 2019 01:57Even better: I have tried to simply replace ld.exe in mingw with clang's (9.0.0) lld.exe and everything works!

Here are times:

with original ld.exe

C:\upp\out\examples\MINGWx64.Debug.Debug_Full.Gui\AddressBoo k.exe (39524957 B) linked in (0:19.68)

with lld.exe

C:\upp\out\examples\MINGWx64.Debug.Debug_Full.Gui\AddressBoo k.exe (36620800 B) linked in (0:00.27)

Thank you Novo for pointing me in the right direction. This is really significant step ahead!
No problem.
I'm glad to help.

Subject: Re: Possible new package or overhaul proposals for U++ in 2019
Posted by cbpporter on Fri, 03 May 2019 08:27:30 GMT

View Forum Message <> Reply to Message

I upgraded to 13076 and there is no lld.exe inside.

Did you replace the executable with the same name or is this not included for testing yet?

Subject: Re: Possible new package or overhaul proposals for U++ in 2019
Posted by mirek on Fri, 03 May 2019 09:50:28 GMT
View Forum Message <> Reply to Message

cbpporter wrote on Fri, 03 May 2019 10:27I upgraded to 13076 and there is no lld.exe inside.

Did you replace the executable with the same name or is this not included for testing yet?

Yes, it is renamed to ld.exe.

You can easily detect its presence by mingw link times... :)

Mirek

Subject: Re: Possible new package or overhaul proposals for U++ in 2019
Posted by cbpporter on Fri, 03 May 2019 13:11:28 GMT
View Forum Message <> Reply to Message

mirek wrote on Fri, 03 May 2019 12:50cbpporter wrote on Fri, 03 May 2019 10:27I upgraded to 13076 and there is no lld.exe inside.

Did you replace the executable with the same name or is this not included for testing yet?

Yes, it is renamed to ld.exe.

You can easily detect its presence by mingw link times... :)

Mirek
Thanks!

Next, we need to use clang to compile :).

Subject: Re: Possible new package or overhaul proposals for U++ in 2019
Posted by Novo on Fri, 03 May 2019 15:49:47 GMT
View Forum Message <> Reply to Message

cbpporter wrote on Fri, 03 May 2019 09:11
Next, we need to use clang to compile :).

I personally would recommend to be careful with clang. I always thought that clang on Linux is the best C++ compiler ever, but a few days ago I've ported my old app written for msvc to clang and got very unpleasant results. The app itself is big and complicated, written in 99.99% C++98 (keyword "auto" is used only once), but it has a lot of CRTP, static visitors and similar stuff. Now the app won't compile with GCC and MSVC, and it is crashing in the places where it used to work before.

IMHO, ideally, U++ should be tested against GCC, MSVC, and Clang.

Subject: Re: Possible new package or overhaul proposals for U++ in 2019
Posted by mirek on Sat, 04 May 2019 06:48:49 GMT
View Forum Message <> Reply to Message

cbpporter wrote on Fri, 03 May 2019 15:11mirek wrote on Fri, 03 May 2019 12:50cbpporter wrote on Fri, 03 May 2019 10:27I upgraded to 13076 and there is no lld.exe inside.

Did you replace the executable with the same name or is this not included for testing yet?

Yes, it is renamed to ld.exe.

You can easily detect its presence by mingw link times... :)

Mirek
Thanks!

Next, we need to use clang to compile :).

Interestingly, from my tests, mingw / lld might be now ideal combo for Win. Clang is not faster compiler and does not produce better code than GCC 8. It even now seems that mingw/lld might be slightly better than Visual C++. The produced code seems a little bit more refined, link times are virtually zero while Visual C++ linker is fast, but still takes a couple of seconds to link large app (like theide). Compile times are about 2x worse, but in normal development cycle link times might be more important.

The only problem with mingw was horrible TLS implementation, but I have now workaround for that for the next release.

Mirek

Subject: Re: Possible new package or overhaul proposals for U++ in 2019
Posted by Tom1 on Fri, 10 May 2019 09:49:07 GMT
View Forum Message <> Reply to Message

Hi,

Does MinGW support linking against external binary libraries created with MSVC?

Also, are structs binary compatible with MSVC?

Best regards,

Tom

Subject: Re: Possible new package or overhaul proposals for U++ in 2019
Posted by mirek on Sat, 11 May 2019 14:05:23 GMT

View Forum Message <> Reply to Message

Tom1 wrote on Fri, 10 May 2019 11:49Hi,

Does MinGW support linking against external binary libraries created with MSVC?

I guess it depends on a lot of things, but generally, yes. Funny part is that whereas you need import library for MSC, mingw links directly .dll :)

Also, are structs binary compatible with MSVC?

Quote:
Best regards,

As far as I can tell, yes.

Subject: Re: Possible new package or overhaul proposals for U++ in 2019
Posted by Novo on Sat, 11 May 2019 17:29:05 GMT

View Forum Message <> Reply to Message

On compatibility of Clang with MSVC: MSVC compatibility
On C++ ABI compatibility in general: C++ binary compatible API (ABI)

Subject: Re: Possible new package or overhaul proposals for U++ in 2019
Posted by Tom1 on Mon, 13 May 2019 07:37:53 GMT

View Forum Message <> Reply to Message

Hi,

Thanks for the info guys. :) I decided to give MinGW a try. Just downloaded the fresh nightly U++ with MinGW. After some tweaking my external library dependencies were satisfied, but for some reason I cannot get rid of the following:

```
Linking...
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
collect2.exe: error: ld returned 1 exit status

There were errors. (1:47.23)
```

I do not have such libraries defined in my packages.

Any idea how to proceed?

Best regards,

Tom