Subject: Visual Studio (Community) 2017 released
Posted by Tom1 on Wed, 15 Mar 2017 14:11:53 GMT
View Forum Message <> Reply to Message

Hi,

It seems Microsoft Visual Studio (Community) 2017 released has been released and is available for download now. It includes MSC14 too. U++ 2017.1 can automatically find and configure that one. However, the MSC15 compiler included is not automatically detected, but can be manually added to U++ build methods. Any plans to support auto detection for that one soon?

I must admit that installation and configuration of VS2017 (for desktop application workload) was easier than before.

Best regards,

Tom

Subject: Re: Visual Studio (Community) 2017 released
Posted by mirek on Thu, 16 Mar 2017 18:20:22 GMT
View Forum Message <> Reply to Message

Tom1 wrote on Wed, 15 March 2017 15:11Hi,

It seems Microsoft Visual Studio (Community) 2017 released has been released and is available for download now. It includes MSC14 too. U++ 2017.1 can automatically find and configure that one. However, the MSC15 compiler included is not automatically detected, but can be manually added to U++ build methods. Any plans to support auto detection for that one soon?

Definitely!

I am thinking that next release should be 2-3 months away. Will probably contain Debian 9 compatibility fixes, this one, and some GIT support.

Mirek

Subject: Re: Visual Studio (Community) 2017 released
Posted by deep on Sat, 25 Mar 2017 04:30:04 GMT
View Forum Message <> Reply to Message

Hi

Have anyone installed VS2017 Community.

I need BM file.

I have installed VS2017 community. Install dir is not in C:. UPP is not finding it automatically. Also I have MinGW in other drive. It is also not auto detected. but could set up manually by copying BM file from other location and modifying it.

Can some one share BM file for VS2015 or VS2017.

Subject: Re: Visual Studio (Community) 2017 released
Posted by cbpporter on Mon, 27 Mar 2017 06:51:22 GMT
View Forum Message <> Reply to Message

Sorry, don't have one for 15. I attached the 14 one, don't know if it will help.

But I never bothered with BM files.

I use the build method navigator and if I point TheIDE in the right direction, I can make it work with any version, probably VS2017 to.

Try something like:
c:/program files (x86)/microsoft visual studio 15.0/
c:/program files (x86)/windows kits/10/lib/

Basically the latest version of VS and Windows Kits.

I have my own detection code, so I guess I need to update it for VS2017.

File Attachments
1) MSC14.bm, downloaded 546 times

Subject: Re: Visual Studio (Community) 2017 released
Posted by deep on Mon, 27 Mar 2017 08:46:07 GMT
View Forum Message <> Reply to Message

Hi cbpporter

Thank you for your file.

Now I have generated MSC17.bm file.

But one strange thing happening.

if Builder name not available in dropdown of build methods it gives invalid builder.
So I selected MSC15X64. All other relevant files for VS2017 selected. It is working.

If I select Builder without x64 then it is giving linking error.
Any Builder with X64 Linking is working. I want to build for 64 bit system.

BUILDER selected from droplist is changing some flag for linking.

BUILDER = "MSC15X64";

My working MSC17.bm

Any suggestions for the options in .bm file

File Attachments

1) MSC17.bm, downloaded 568 times

Subject: Re: Visual Studio (Community) 2017 released
Posted by cbpporter on Wed, 29 Mar 2017 07:23:36 GMT
View Forum Message <> Reply to Message

Well, I'll get back to you when I get to install the latest VS. This will take at least one month. I have two post release bi-weekly updates to prepare so I have no time to install new VS. When I'll install I'll also test auto-detecting it.

Until then, as Mirek said, we might get official support for auto-detection. If you managed to get it to work with that BM file it means that the process is similar to the rest of MSC so auto-detection should be easy.

Subject: Re: Visual Studio (Community) 2017 released
Posted by wimpie on Sat, 01 Apr 2017 22:52:52 GMT
View Forum Message <> Reply to Message

deep wrote on Mon, 27 March 2017 10:46Hi cbpporter

If I select Builder without x64 then it is giving linking error.
Any Builder with X64 Linking is working. I want to build for 64 bit system.

my thanks also. "Builder without x64" needs other paths.. to x86 stuff.

I just installed UPP after some time, and installed VS community 2017 also, and tried your BM file after some path adjustments.

next thing to try... clang/c2???

File Attachments

1) MSC17.bm, downloaded 556 times

Subject: Re: Visual Studio (Community) 2017 released
Posted by Tom1 on Mon, 03 Jul 2017 10:40:08 GMT
View Forum Message <> Reply to Message

Hi,

It seems the Microsoft Visual C++ compilers' version naming on U++ is not entirely accurate. For quite a while U++ used MSC15 for what is now called MSC14. Later, on top of this thread I made a mistake by calling the new compiler version MSC15. Well, according to this, it was a mistake:

https://en.wikipedia.org/wiki/Microsoft_Visual_C%2B%2B#Inter nal_version_numbering

This sort of implies we should have used MSC14.1 for the VS 2017 compiler. Or maybe we should have used MSC19.1 instead: Namely, when we open the VS 2017 64-bit build environment and call cl.exe and link.exe we will get:

```
**********************************************************************
** Visual Studio 2017 Developer Command Prompt v15.0.26403.7
** Copyright (c) 2017 Microsoft Corporation
**********************************************************************
[vcvarsall.bat] Environment initialized for: 'x64'

C:\Program Files (x86)\Microsoft Visual Studio\2017\Community>cl
Microsoft (R) C/C++ Optimizing Compiler Version 19.10.25019 for x64
Copyright (C) Microsoft Corporation.  All rights reserved.

usage: cl [ option... ] filename... [ /link linkoption... ]

C:\Program Files (x86)\Microsoft Visual Studio\2017\Community>link
Microsoft (R) Incremental Linker Version 14.10.25019.0
Copyright (C) Microsoft Corporation.  All rights reserved.

 usage: LINK [options] [files] [@commandfile]
```

So, in effect we have compiler version 19.10 and linker version 14.10 running on top of Developer command prompt 15.0...

This all adds up to a very confusing versioning. Would it be clearer to start using the Visual Studio versioning and call the build methods along the lines of VS2015, VS2017 to keep track of what compiler product to install to get specific build system? Or would it be better to go along with the version number of the compiler (e.g. 19.10), which is defined with _MSC_VER as e.g. 1910? Anyway, the MSC15 is not correct and I do not feel 14.10 is a strong identifier for the compiler either.

Best regards,

Tom

Subject: Re: Visual Studio (Community) 2017 released
Posted by mirek on Mon, 03 Jul 2017 12:54:22 GMT
View Forum Message <> Reply to Message

Tom1 wrote on Mon, 03 July 2017 12:40
This all adds up to a very confusing versioning. Would it be clearer to start using the Visual Studio versioning

Well, that is why I was using MSC15 initially... (but the community did not liked that...)

Quote:
and call the build methods along the lines of VS2015, VS2017 to keep track of what compiler product to install to get specific build system? Or

Might work. I was a little bit hesitant to have "VS" moniker, as we are using just compiler.

Quote:
would it be better to go along with the version number of the compiler (e.g. 19.10), which is defined with _MSC_VER as e.g. 1910? Anyway, the MSC15 is not correct and I do not feel 14.10 is a strong identifier for the compiler either.

Another option is to keep MSC14 for 2017 too - in reality, if I remember well, nothing has changed except autodetection process...

Best regards,

Tom[/quote]

Subject: Re: Visual Studio (Community) 2017 released

Posted by cbpporter on Mon, 03 Jul 2017 13:26:19 GMT
View Forum Message <> Reply to Message

Visual Studio 2015 = MSC14 = CL 19.00
Visual Studio 2017 = MSC15 = CL 19.10

Tom1, is this not what you expect to see?

Subject: Re: Visual Studio (Community) 2017 released
Posted by Tom1 on Mon, 03 Jul 2017 13:32:40 GMT
View Forum Message <> Reply to Message

Mirek,

Just out of interest I briefly installed something called "Visual Studio Build Tools 2017". The idea was to drop out the VS IDE. After manually adjusting the paths (auto detection does not work for this), the compilation and linking worked with that too (same as VS2017 Community) at least for 32 bit targets. However, there was no debugger bundled -- or I just could not find it. Anyway, I removed it and reinstalled VS2017 Community. The reason for trying was to avoid the unnecessarily large installation of the whole VS2017.

The build method names, in my opinion, should point out the actual toolset installed in a way that it can be easily identified, and therefore found and reinstalled, if needed.

--

cbpporter: MSC15 was what I expected to see, before I found the link and thereafter some other material pointing out that it might actually be MSC14.1. Indeed the toolset referred by the Visual Studio 2017 installer refers to v141 for the latest toolset.

I just wanted to point out that I may have made a mistake by assuming it would have been version 15 whereas it may have been 14.1 in question.

Best regards,

Tom

Subject: Re: Visual Studio (Community) 2017 released
Posted by cbpporter on Mon, 03 Jul 2017 13:36:25 GMT
View Forum Message <> Reply to Message

14.1 is not something I can find in the main listing.

Might it be related to the pre-release version? I think it is a RC or something.

Subject: Re: Visual Studio (Community) 2017 released
Posted by Tom1 on Mon, 03 Jul 2017 14:11:11 GMT
View Forum Message <> Reply to Message

You see the link below. It is 14.1 there:

https://en.wikipedia.org/wiki/Microsoft_Visual_C%2B%2B#Inter nal_version_numbering

Additionally, when you run Visual Studio 2017 installer and select Visual Studio Community 2017 and Desktop development with C++ workload, you can see "VC++ 2017 v141 toolset (x86,x64)" as a selected component. Further on, there is an option to install "VC++ 2015.3 v140 toolset (x86,x64)".

When you search the web for microsoft visual c++ compiler versions, you will find additional discussions of the confusion over the version numbers.

I started this discussion here for U++ users to recognize the problem and possibly overcome the current situation by changing the naming to something that easily relates to the Microsoft compiler product being used. For me personally, it does not matter what they are called as far as I know what to install and that the result will work correctly. However, to avoid confusion and therefore wasted time for many other people, it would be easier to use names and numbers that can be used to pick up a specific compiler product on the web.

Best regards,

Tom

Subject: Re: Visual Studio (Community) 2017 released
Posted by Tom1 on Mon, 03 Jul 2017 14:21:48 GMT
View Forum Message <> Reply to Message

Hi,

Please take into account the possibility that Microsoft too is confused here; Microsoft VC CRT redistributable components are in a folder called Microsoft.VC150.CRT but contain files with names *140.dll ...

Best regards,

Tom

Subject: Re: Visual Studio (Community) 2017 released
Posted by mirek on Mon, 03 Jul 2017 15:40:35 GMT
View Forum Message <> Reply to Message

Tom1 wrote on Mon, 03 July 2017 15:32Mirek,

Just out of interest I briefly installed something called "Visual Studio Build Tools 2017".

Do they release this once again? Cool! Where is the link?

Quote:
worked with that too (same as VS2017 Community) at least for 32 bit targets. However, there was no debugger bundled -- or I just could not find

Debugger in theide is sort of independent of it, as long as build tools are able to produce debug info / .pdb.

Quote:
The build method names, in my opinion, should point out the actual toolset installed in a way that it can be easily identified, and therefore found and reinstalled, if needed.

Personally, I would prefer MSCyear scheme. It is really unlikely that they would change compiler commandline options twice in single year and with all the confusion about MSC version it is as good as it gets. That would make current MSC: MSC17.

BTW:

https://blogs.msdn.microsoft.com/vcblog/2016/11/16/introduci ng-the-visual-studio-build-tools/

It seems like MS 'screenname' is VC++ 2017 v141, which is inline with MSC17 (IMO).

Subject: Re: Visual Studio (Community) 2017 released
Posted by Tom1 on Mon, 03 Jul 2017 18:51:13 GMT
View Forum Message <> Reply to Message

Mirek,

I installed Visual Studio 2017 build tools using this link:

https://www.visualstudio.com/thank-you-downloading-visual-st udio/?sku=BuildTools&rel=15

Visual Studio 2015 build tools download button can be found on:

http://landinghub.visualstudio.com/visual-cpp-build-tools

MSCyear would be fine with me. However, since the convention would then change from what it was until now, should it use four digit year instead? Like MSC2017 to reflect the fact that the logic has changed and it is no longer the toolset version that is being referred to by the number.

If you look at the Visual Studio installation folders' names in C:\Program Files (x86)\ carefully, you can see that the old ones used to be like "Microsoft Visual Studio 9.0" and "Microsoft Visual Studio 14.0". But now for the first time it is just "Microsoft Visual Studio" and now there is a sub folder called "2017".

Best regards,

Tom

Subject: Re: Visual Studio (Community) 2017 released
Posted by koldo on Tue, 04 Jul 2017 09:26:14 GMT
View Forum Message <> Reply to Message

Hello all

After installing MSVC, I have created a MSC17x64.bm file based in MSC17.bm file uploaded in this post.

I get these errors:
C:\Program Files (x86)\Microsoft Visual Studio\2017\BuildTools\VC\Tools\MSVC\14.10.25017\include\xtr1common(100): error C2737: 'std::is_same_v': 'constexpr' object must be initialized
Include folders are:
C:\Program Files (x86)\Microsoft Visual Studio\2017\BuildTools\VC\Tools\MSVC\14.10.25017\include
C:\Program Files (x86)\Windows Kits\10\Include\10.0.10586.0\um
C:\Program Files (x86)\Windows Kits\10\Include\10.0.10586.0\ucrt
C:\Program Files (x86)\Windows Kits\10\Include\10.0.10586.0\shared

---

Subject: Re: Visual Studio (Community) 2017 released
Posted by Tom1 on Tue, 04 Jul 2017 10:47:08 GMT
View Forum Message <> Reply to Message

Hi Koldo,

Not sure if this helps, but I tested the VS2017 BuildTools by duplicating the build method I already had for VS2017 community edition (which was automatically generated by theide). Then I just renamed it and changed the include, lib and bin folders to point at BuildTools instead of the VS community.

Best regards,

Tom

Subject: Re: Visual Studio (Community) 2017 released
Posted by koldo on Wed, 05 Jul 2017 14:16:50 GMT
View Forum Message <> Reply to Message

Thank you Tom. However in my case TheIDE Automatic Build Setup does not generate any new file.

I have moved all .bm files to other folder, opened TheIDE and called Automatic Build Setup. Then it finds old install 14, but it does not find "Microsoft Visual Studio\2017".

I installed 2017 from here: https://www.visualstudio.com/thank-you-downloading-visual-studio/?sku=BuildTools&rel=15. Maybe do I need to install anything else?

Subject: Re: Visual Studio (Community) 2017 released
Posted by mirek on Wed, 05 Jul 2017 16:31:30 GMT
View Forum Message <> Reply to Message

koldo wrote on Wed, 05 July 2017 16:16Thank you Tom. However in my case TheIDE Automatic Build Setup does not generate any new file.

I have moved all .bm files to other folder, opened TheIDE and called Automatic Build Setup. Then it finds old install 14, but it does not find "Microsoft Visual Studio\2017".

I installed 2017 from here: https://www.visualstudio.com/thank-you-downloading-visual-studio/?sku=BuildTools&rel=15. Maybe do I need to install anything else?

Are you using U++ trunk version?

(Current stable was released about the same time as VS2017, so there is not a support in stable yet).

Subject: Re: Visual Studio (Community) 2017 released
Posted by mirek on Wed, 05 Jul 2017 17:19:30 GMT
View Forum Message <> Reply to Message

Tom1 wrote on Tue, 04 July 2017 12:47Hi Koldo,

Not sure if this helps, but I tested the VS2017 BuildTools by duplicating the build method I already had for VS2017 community edition (which was automatically generated by theide). Then I just renamed it and changed the include, lib and bin folders to point at BuildTools instead of the VS community.
Tom

BTW, could you post that .bm here? Just to get me started...

Subject: Re: Visual Studio (Community) 2017 released
Posted by Tom1 on Wed, 05 Jul 2017 20:38:19 GMT
View Forum Message <> Reply to Message

Hi Koldo and Mirek,

I had already trashed that installation, but I reinstalled the Visual Studio 2017 Build Tools and reconfigured TheIDE with a suitable BM. Please find it attached. Also, note that this is a 32-bit BM. It uses debugger from the parallel Visual Studio 2017 Community installation because I think there is none included with Build Tools installation.

Best regards,

Tom

File Attachments
1) VSBT2017.bm, downloaded 536 times

Subject: Re: Visual Studio (Community) 2017 released
Posted by koldo on Thu, 06 Jul 2017 06:14:26 GMT
View Forum Message <> Reply to Message

Thank you Mirek. This is my BM.

Thank you Tom. A difference with your MSVC install is that your is under c:/program files and mine is under C:\Program Files (x86). Maybe is your OS 32 bits?

File Attachments
1) MSC17x64.bm, downloaded 531 times

Subject: Re: Visual Studio (Community) 2017 released
Posted by Tom1 on Thu, 06 Jul 2017 06:41:54 GMT
View Forum Message <> Reply to Message

Hi Koldo,

You are absolutely right. I made it under Windows 10 Pro 32 bit.

Please note that current U++ can auto detect the VS2017 Community installation, while it cannot auto detect VS2017 Build Tools installation at this time. I have a strong feeling that this will change pretty soon now since Mirek indicated interest in this. I think the compiler is exactly the same, but Build Tools lacks debugger and IDE compared to a full VS2017 Community installation. (So installing VS2017 Community already includes everything needed and is also readily auto detected with current U++.)

Best regards,

Tom

Subject: Re: Visual Studio (Community) 2017 released
Posted by cbpporter on Thu, 06 Jul 2017 07:30:22 GMT
View Forum Message <> Reply to Message

I see MS are as bad as ever with consistency and their naming schemes. I have not yet bothered with 2017, but I guess I need to install it.

A bit of a pain since I use and regularly test for MSC10, MSC12 and MSC15. Now I'll have MSC17 too .

Subject: Re: Visual Studio (Community) 2017 released
Posted by Tom1 on Thu, 06 Jul 2017 08:04:54 GMT
View Forum Message <> Reply to Message

Hi,

Just for completeness, please find attached both 32 and 64 bit BM files for Visual Studio 2017 Build Tools created on Windows 10 Professional 64 bit OS.

Best regards,

Tom

File Attachments
1) VSBT2017BMs.zip, downloaded 486 times

Subject: Re: Visual Studio (Community) 2017 released
Posted by Tom1 on Thu, 06 Jul 2017 08:11:35 GMT
View Forum Message <> Reply to Message

Hi Koldo,

I just took a closer look and found that your MSC17x64.bm file also had PATH and LIB references pointing at VS2017 Community. Only INCLUDE was pointing to BuildTools. This may explain if you had issues with your BM when trying to use BuildTools instead of Community.

Best regards,

Tom

Subject: Re: Visual Studio (Community) 2017 released
Posted by koldo on Thu, 06 Jul 2017 14:15:35 GMT
View Forum Message <> Reply to Message

Tom1 wrote on Thu, 06 July 2017 10:11I just took a closer look and found that your MSC17x64.bm file also had PATH and LIB references pointing at VS2017 Community. Only INCLUDE was pointing to BuildTools. This may explain if you had issues with your BM when trying to use BuildTools instead of Community.Thank you Tom, you are right!

The only problem is that I have not found any devenv.exe under Microsoft Visual Studio\2017.

However TheIDE seems to compile and debug

Subject: Re: Visual Studio (Community) 2017 released
Posted by mirek on Thu, 06 Jul 2017 17:07:29 GMT
View Forum Message <> Reply to Message

MS Build Tools now supported by auto setup.

BM names changed to MSVS15, MSVS17, MSBT17

Subject: Re: Visual Studio (Community) 2017 released
Posted by koldo on Fri, 07 Jul 2017 07:16:16 GMT
View Forum Message <> Reply to Message

Perfect.

In my case, with BuildTools, _MSC_FULL_VER reports
- 191025019 for MSC17x64
- 190023026 for MSC15x64

Subject: Re: Visual Studio (Community) 2017 released
Posted by Tom1 on Fri, 07 Jul 2017 08:14:39 GMT
View Forum Message <> Reply to Message

Hi,

I can confirm that the 64 bit TheIDE 11218 correctly autoconfigured both x86 and x64 build methods for VS 2015, VS 2017 and BT 2017. This was on Windows 10 Pro 64.

Thanks and best regards,

Tom

Subject: Re: Visual Studio (Community) 2017 released
Posted by rafiwui on Fri, 07 Jul 2017 15:45:39 GMT
View Forum Message <> Reply to Message

Well I can't confirm it on my home PC but I can confirm it on my work PC.

But I have to say my setup at home is a bit lets say "special"

Windows 10 64bit is installed on C:\
Visual Studio 2017 Community is installed on a secondary SSD (E:\Programs\Visual Studio 2017 Community)
Finally U++ is unzipped on another secondary HDD (D:\Programs\U++\upp)

When I start TheIDE the first time you see for a short time that it is doing sth about build methods but then continues and nothing is created. When I tried installing the new nightly at work today this was longer and the correct .bm files were created.

So I guess the installer just can't connect my 3 drives Or is it windows and the methods from the WinAPI which are getting called? (I did not take a deeper look inside the files so I can't tell)

Can someone upload a newly created MSVS2017.bm/MSVS2017x64.bm for me so I can set the paths correct?

Greetings
Daniel

Subject: Re: Visual Studio (Community) 2017 released
Posted by Tom1 on Tue, 11 Jul 2017 07:06:20 GMT
View Forum Message <> Reply to Message

Hi,

Sorry for the delay. Here are the BMs (auto-generated by TheIDE 11218) for Visual Studio 2017 Community and also Visual Studio 2017 Build Tools.

Best regards,

Tom

File Attachments

1) upp-11218-VS2017BMs.zip, downloaded 486 times

Subject: Re: Visual Studio (Community) 2017 released
Posted by rafiwui on Tue, 11 Jul 2017 17:00:14 GMT
View Forum Message <> Reply to Message

Thanks for the files. But it does not work. TheIDE tells me when I try to compile:
c:\program files (x86)\windows kits\10\include\10.0.15063.0\um\winsock2.h (882): fatal error C1083: Cannot open include file: 'qos.h': No such file or directory

There is no file 'qos.h' in the folder c:\program files (x86)\windows kits\10\include\10.0.15063.0\um. Should it be there?

And another thing: In your MSVS17x64 there is this line:
PATH = "...;c:/program files (x86)/windows kits/10/bin/x86;...;
Shouldn't it be:
PATH = "...;c:/program files (x86)/windows kits/10/bin/x64;...;
??

Subject: Re: Visual Studio (Community) 2017 released
Posted by Tom1 on Wed, 12 Jul 2017 09:19:47 GMT
View Forum Message <> Reply to Message

Hi,

The files I uploaded are just the ones my U++ installation have auto-created on my platform. I wouldn't necessarily expect them to work on your system without modification. I just uploaded them for you as a template. If you find similar paths on your system, you can then tune these files accordingly to match your system configuration.

I have no knowledge of the automatic build method generation, so I cannot help on that part. Certainly having unusual installation paths can affect such code.

The reference to x86 in the PATH does not appear to cause any trouble on my 64-bit system. The reason for it is unknown to me.

I wish you the best of luck!

Best regards,

Tom

Subject: Re: Visual Studio (Community) 2017 released
Posted by rafiwui on Wed, 12 Jul 2017 19:39:15 GMT
View Forum Message <> Reply to Message

Ok I found the problem. Somehow in the include folder of the newest windows sdk (c:/program files (x86)/windows kits/10/include/10.0.15063.0/...) some files are missing (qos.h is one of them). So I guess your files are working (thanks for giving me them) but I have to check my visual studio for updates or repairing my sdk.

Subject: Re: Visual Studio (Community) 2017 released
Posted by rafiwui on Mon, 17 Jul 2017 13:47:28 GMT
View Forum Message <> Reply to Message

Ok the problem was my VS2017. It wasn't up to date with the newest Win10 SDK. Working perfectly fine now.

Subject: Re: Visual Studio (Community) 2017 released
Posted by forlano on Sun, 13 Aug 2017 19:57:51 GMT
View Forum Message <> Reply to Message

Tom1 wrote on Tue, 11 July 2017 09:06
Sorry for the delay. Here are the BMs (auto-generated by TheIDE 11218) for Visual Studio 2017 Community and also Visual Studio 2017 Build Tools.

Hello,

today I have installed U++ 11300 and VS2017 community on a 64 bit machine with windows 7.
I looked for a bm file because my theide was not able to produce one via automatic build method setup.
Then I used your MSVS17x64.bm file but I had to rename it MSC17x64.bm to avoid an "invaild build method".
Not it seems to compile as usual.

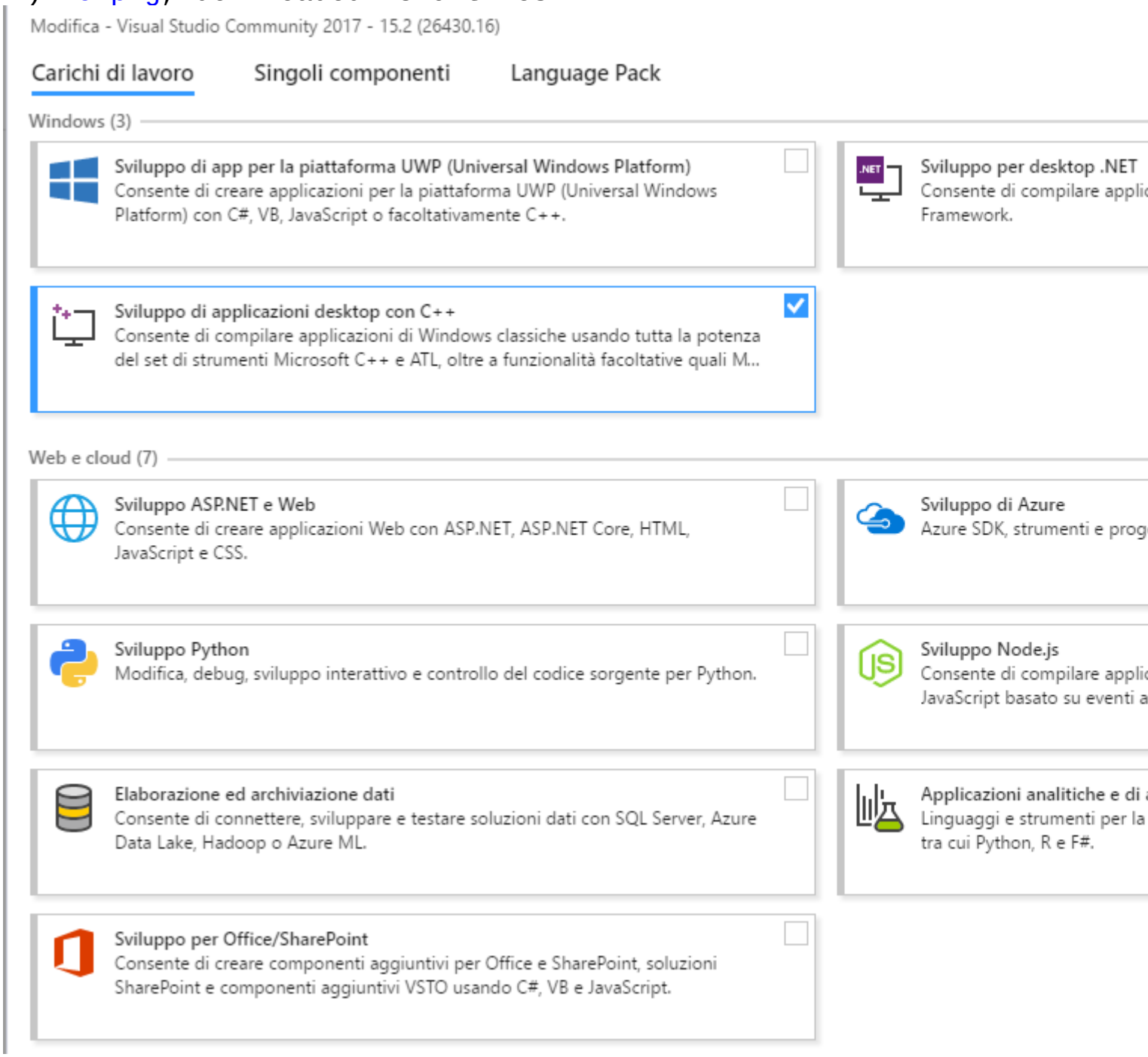
However it didn't produce an exe file.
It stopped at processing a .rc file.
Reading the forum I have understood my installation has not the rc.exe program, so I missed to download something.
Does anybody know (see picture) what have I to download to have a U++ working setup.

Thanks,

Luigi

File Attachments

1) rc.png, downloaded 1829 times



Subject: Re: Visual Studio (Community) 2017 released
Posted by deep on Mon, 14 Aug 2017 05:03:01 GMT
View Forum Message <> Reply to Message

Hi Luigi,

I think you will require this additionally.

https://developer.microsoft.com/en-US/windows/downloads/wind ows-10-sdk

---

Subject: Re: Visual Studio (Community) 2017 released
Posted by rafiwui on Mon, 14 Aug 2017 06:58:30 GMT
View Forum Message <> Reply to Message

That is correct. You need the Win10 SDK. But you can select it in the VS installer directly. If you can't find it on the first tab. Search it in the single components tab and select it.

Subject: Re: Visual Studio (Community) 2017 released
Posted by forlano on Mon, 14 Aug 2017 09:50:52 GMT
View Forum Message <> Reply to Message

deep wrote on Mon, 14 August 2017 07:03Hi Luigi,

I think you will require this additionally.

https://developer.microsoft.com/en-US/windows/downloads/wind ows-10-sdk

Thanks for the answer.
I installed it despite I checked it in the visual studio download.
Again it compiles but no exe. It stops at a .rc file processing.
I cannot find a rc.exe in my distro.
Where this file should be? Have you tried to compile a program with a .rc file?

Thanks,
Luigi

Subject: Re: Visual Studio (Community) 2017 released
Posted by deep on Mon, 14 Aug 2017 10:10:49 GMT
View Forum Message <> Reply to Message

Hi Luigi,

Check your "build methods" settings in MSCxx method

Add in "PATH - executable directories" folder path for "Windows Kits" where rc.exe is located.

In my case it is
"C:\Program Files (x86)\Windows Kits\10\bin\x64"

Subject: Re: Visual Studio (Community) 2017 released
Posted by forlano on Mon, 14 Aug 2017 10:26:15 GMT
View Forum Message <> Reply to Message

deep wrote on Mon, 14 August 2017 12:10Hi Luigi,

Check your "build methods" settings in MSCxx method

Add in "PATH - executable directories" folder path for "Windows Kits" where rc.exe is located.

In my case it is
"C:\Program Files (x86)\Windows Kits\10\bin\x64"

Now it works!

for some reason the search method does not work on my system and the rc.exe is located in C:\Program Files (x86)\Windows Kits\10\bin\10.0.15063.0\x64

Thanks again!
Luigi

Subject: Re: Visual Studio (Community) 2017 released
Posted by Tom1 on Mon, 14 Aug 2017 10:57:14 GMT
View Forum Message <> Reply to Message

Hi Luigi,

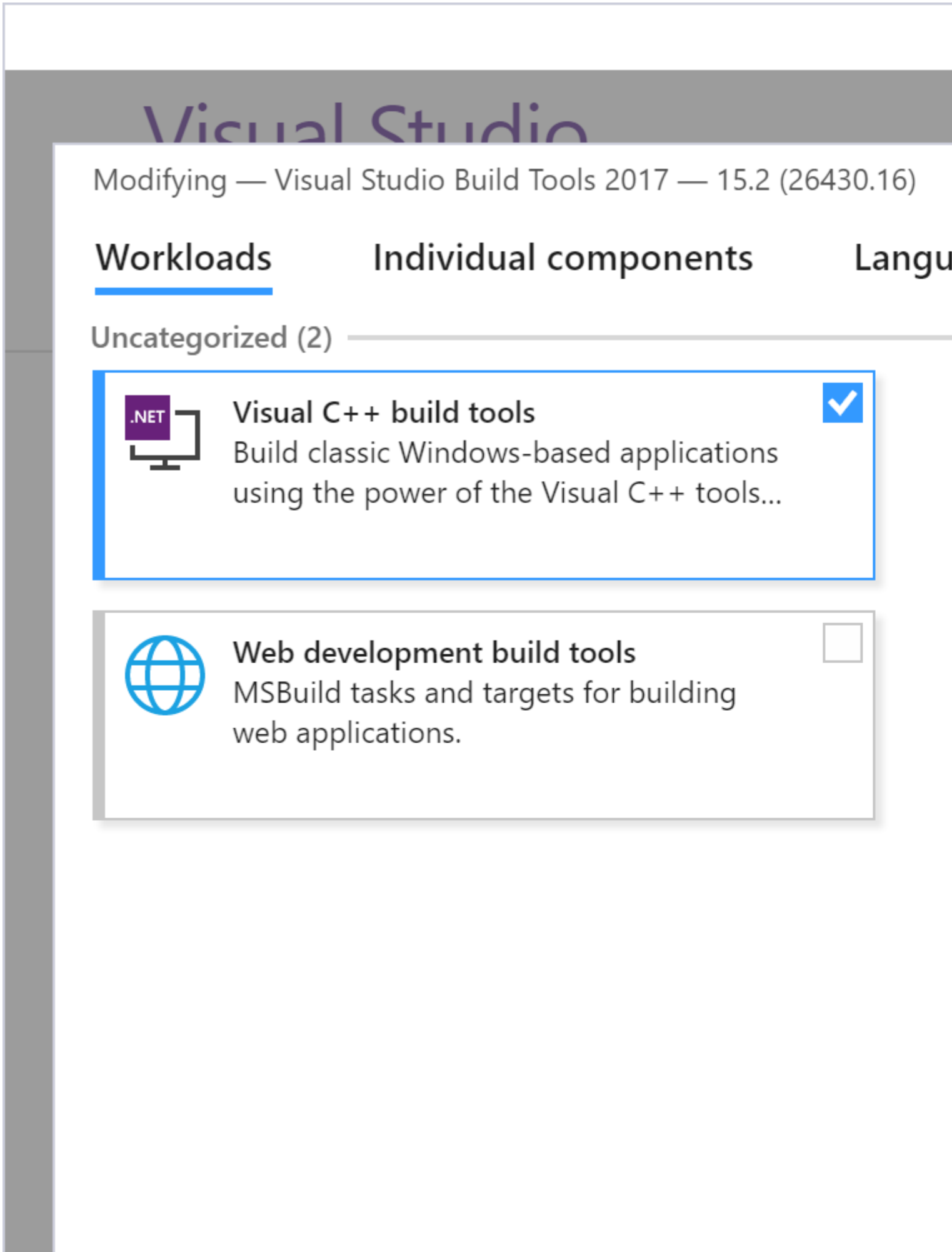
I see the problem was solved already. That's good.

Just for reference, please find attached an image of my Visual Studio Build Tools setup. This was completely automatically detected with TheIDE and works (with .rc's too).

Best regards,

Tom

File Attachments
1) VSBT2017.png, downloaded 1068 times


Visual Studio
Modifying — Visual Studio Build Tools 2017 — 15.2 (26430.16)
Workloads
Individual components
Langu
Uncategorized (2)
.NET
Visual C++ build tools
Build classic Windows-based applications using the power of the Visual C++ tools...
Web development build tools
MSBuild tasks and targets for building web applications.

Subject: Re: Visual Studio (Community) 2017 released
Posted by forlano on Mon, 14 Aug 2017 14:50:40 GMT
View Forum Message <> Reply to Message

Hi Tom,

Thanks. It seems I downloaded several Gigabyte of useless code and the good one are at bottom page here

https://www.visualstudio.com/downloads/

Best regards,
Luigi

Subject: Re: Visual Studio (Community) 2017 released
Posted by JeyCi on Wed, 11 Jan 2023 15:58:25 GMT
View Forum Message <> Reply to Message

deep wrote on Mon, 14 August 2017 12:10
Add in "PATH - executable directories" folder path for "Windows Kits" where rc.exe is located.
In my case it is
"C:\Program Files (x86)\Windows Kits\10\bin\x64"
helped with MSBT2019 (VS Build Tools 2019) too. Auto-detected when first IDE Open. Thanks.
Though I wonder for the same reason:
koldo wrote on Thu, 06 July 2017 16:15
The only problem is that I have not found any devenv.exe under Microsoft Visual Studio\2017.
However TheIDE seems to compile and debug
Hope, nothing changed! If you experienced some problems - please give your feedback here, but now I just leave everything as-is.
And myBM attached

Perhaps, to PATH : [i]C:\Program Files (x86)\Microsoft Visual Studio\2019\BuildTools\MSBuild\Current\Bin[\i] or any folder where VSBT was installed e.g. C:\VSBT\MSBuild\Current\Bin

p.s.
Happy 2023 New Year for everybody with best wishes in your Dev. path

File Attachments

1) MSBT19.bm, downloaded 305 times