

Public Support - Support Request #8864

Visual Studio 2017 Version for ADTF 3

2019-11-05 17:04 - hidden

Status:	Closed	
Priority:	Normal	
Category:		
Customer:	BOSCH	Product Issue Numbers:
Department:	CC-DA/ETV	Affected Products: ADTF 3.6.2
Requester's Priority:	Normal	Platform: Windows 10 64bit
Support Level:	2nd Level	Topic: ADTF::Common
Resolution:	Solved Issue	FAQ Links:

Description

Supportanfrage

Hallo, im ADTF Guide steht, dass man Visual Studio 2017 installieren soll aber ohne eine konkrete Version.

In der OS Freigabe Matrix steht vc141, was der aller ersten Version von VS2017 entsprechen würde.

Kann man auch die neuste V15.9 von Visual Studio 2017 (MSVC++ 14.16) nutzen?

Bei ADTF 2 gab es in der Vergangenheit den Fall, dass bestimmte VS Service Packs nicht empfohlen wurden. Darum frage ich lieber nach.

Hintergrund: Wir hatten eine interne Diskussion welchen C++ Sprachstandard VS2017 unterstützt und wie weit. Dabei kam heraus: Je neuer die Version von VS2017 desto mehr der Features von C++17 sind drin. In dieser Hinsicht würden wir auch die neuste Version nehmen, wenn aus ADTF3-Sicht da nichts gegen spricht.

Ist VS2019 bei Euch auch ein Thema? Habt Ihr es vielleicht sogar schon ausprobiert? Wurde das bereits angefragt?

Lösung

Das passt, siehe [System Requirements](#)

Ebenso https://cmake.org/cmake/help/latest/variable/MSVC_VERSION.html#variable:MSVC_VERSION bzw.

https://cmake.org/cmake/help/latest/variable/MSVC_TOOLSET_VERSION.html#variable:MSVC_TOOLSET_VERSION

Hinsichtlich VS2019 gibt es keine Erfahrung/Bestrebungen aktuell.

History

#1 - 2019-11-05 17:14 - hidden

- Project changed from Public Support to 5
- Status changed from New to Customer Feedback Required
- Topic set to ADTF::Common

Hallo Roland,

Schnellschuss, vlt beantwortet das sogar die Frage:

[System Requirements](#)

Runtime requirements:

Visual C++ 2017 Redistributable

Visual Studio C++ 2017 (for debug libraries and executables)

Development requirements:

Visual Studio C++ 2017 (Version 15.0. at least 15.9.11+28307.586)

VC141 Toolchain

Windows Kit 10 (at least 10.0.17763.0 shipped with Visual Studio C++ 2017)

Bitte Bescheid geben, ob das nützlich war

#2 - 2019-11-06 07:49 - hidden

Danke für die schnelle Rückmeldung.

Jetzt weiß ich auch, wo die Info in der SDK-Doku steht. Gestern habe ich das auf die Schnelle nicht gefunden.
Was ich irritierend finde ist allerdings die Formulierung: Version 15.0 at least 15.9.11 ...
15.0 wäre die erste VS2017 Version und 15.9 die neuste. Wenn mindesten 15.9 vorgeben ist, dann lasst doch bitte 15.0 einfach weg.
Was mir auch nicht klar ist: Was verbirgt sich hinter der VC141 Toolchain?

Bezüglich der VS2017 Version ist meine Frage beantwortet und der Hinweis war nützlich.

Wäre schön noch eine kurze Rückmeldung zu bekommen, ob in Richtung VS2019 bei Euch bereits irgendetwas gelaufen ist und schon Erfahrungen vorliegen. Erwartungen in diese Richtung habe ich aber keine.

#3 - 2019-11-07 09:03 - hidden

- Status changed from Customer Feedback Required to In Progress

@Sebastian kannst du dir das bitte ansehen?

#4 - 2019-11-11 18:04 - hidden

- Status changed from In Progress to Customer Feedback Required

- Platform Windows 10 64bit added

Hi Roland,

Was ich irritierend finde ist allerdings die Formulierung: Version 15.0 at least 15.9.11 ...
15.0 wäre die erste VS2017 Version und 15.9 die neuste. Wenn mindesten 15.9 vorgeben ist, dann lasst doch bitte 15.0 einfach weg.

Ja das ist verwirrend, das hat aber eher was mit der "schönen" Versionierung von Microsoft zu tun, die geben das offiziell so aus...

Also ein VS 2017 bedeutet, ich bin die interne MS Version 15.0 (die gebend das genau so an, leider...)
Vielleicht sollten wir nur 15 ohne 0 schreiben, entscheidend ist ja hier die Angabe, mit welcher Version unsere Compiler bestückt waren.

- Visual Studio C++ 2017 (Version 15 at least 15.9.11+28307.586)

Was mir auch nicht klar ist: Was verbirgt sich hinter der VC141 Toolchain?

VS 2017 bzw. dessen Compiler Toolchain / Build Tools sind in der Version 14.1 (vc141).
Zusätzlich kann man noch den von VS2015 (Version 14.0, also vc140) verwenden als legacy installieren.
VS2019 wäre dann vc142.
Siehe z.B. <https://docs.microsoft.com/de-de/cpp/porting/binary-compat-2015-2017?view=vs-2019>

Hieran sieht man im übrigen auch die erstmalige ABI Kompatibilität zwischen VS2015.3 und VS2017 (nur ein Minor Update in VC Toolchain)

Wäre schön noch eine kurze Rückmeldung zu bekommen, ob in Richtung VS2019 bei Euch bereits irgendetwas gelaufen ist und schon Erfahrungen vorliegen. Erwartungen in diese Richtung habe ich aber keine.

Hier haben wir noch keine Erfahrung gesammelt. Das wäre wieder mit einem Update oder zusätzlichen Plattform verbunden.
Mit dem Upgrade zu VS2017 sind wir erstmal versorgt, auch zahlreiche Libs auf lange Zeit darauf ausgelegt und verwendbar mit MS2019 (bei "alter" VS2017 vc141 Toolchain) natürlich.
Einen Mix würde ich vermeiden an der Stelle, da MS das nicht verspricht.

Beantwortet das deine Fragen ?

#5 - 2019-11-12 13:29 - hidden

Auf jeden Fall habe ich jetzt mehr Klarheit aber doch noch eine abschließende Frage.
Die Informationen über die VC++ Compiler Versionen habe ich von Wikipedia: https://en.wikipedia.org/wiki/Microsoft_Visual_C%2B%2B
Dort steht dass MSVC++14.1 die Version ist, die mit dem ersten Release von VS2017 (V15.0) ausgeliefert wurde. Für die weiteren (Patch-Releases) von VS hat Microsoft noch eine zusätzlich Ziffer angehängt. Zum neusten VS2017 Release 15.9 gehört MSVC++14.16.
Um sicher zu gehen: Wir können VS2017 15.9 mit MSVC++14.16 verwenden es muss nicht explizit 14.1 sein, richtig?
Wenn die Frage beantwortet ist, kann das Ticket geschlossen werden.

#6 - 2019-11-12 13:50 - hidden

- Project changed from 5 to Public Support

- Subject changed from Visual Studio 2017 Version für ADTF 3 to Visual Studio 2017 Version for ADTF 3

- Description updated

- Status changed from Customer Feedback Required to To Be Closed

- *Private changed from Yes to No*

- *Resolution set to Solved Issue*

Hi Roland,

ja genau, das passt, das ist dann lediglich eine Patch Version des Compilers.

Das wichtigste ist, dass ihr im Visual Studio bzw. CMake die Toolchain v141 / vc141 verwendet und nicht v140 / vc140 (der eigentlich zu VS2015 gehört).

Ich gehe noch einmal in mich und versuche das ggf. besser rauszustellen, zumindest so wie bisher im letzten Kommentar.

Diese Übersicht hilft mir immer:

https://cmake.org/cmake/help/latest/variable/MSVC_VERSION.html#variable:MSVC_VERSION

bzw.

https://cmake.org/cmake/help/latest/variable/MSVC_TOOLSET_VERSION.html#variable:MSVC_TOOLSET_VERSION

Entspricht deinen Link, nur etwas kompakter und wie es sich intern (aus binary Entwicklung, bzw. für ADTF relevant "anfühlt")

#8 - 2020-07-07 12:49 - hidden

- *Status changed from To Be Closed to Closed*