



链滴

正确在 Windows 上配置 MSVC(2019) 作为 Qt 编译器

作者: [xflcx1991](#)

原文链接: <https://ld246.com/article/1592296047699>

来源网站: 链滴

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

正确在 Windows 上配置 MSVC(2019) 作为 Qt 编译器

本文致力于讲述清楚如何在仅安装 vs 生成工具（不安装完整版本的 visual studio）的情况下配置好环境，使得 Windows+Qt(msvc)+msvc 工具链能正常使用，并能正常单步调试（cdb）。本文极有可能是网上少数讲清楚的教程，反正我自己在解决问题前搜索了大量资料博客。▯

erd_face 不想看废话的话，直接跳至

[具体操作说明](#)

问题由来

早期版本的 vs 和 vs 生成工具，会自动配置好相关环境（后面会讲到底是指哪里），使得安装 Qt（msv 版）及 Qtcreator 后无感直接使用。但是新版本的 vs 生成工具（2019）并不会自动配置导致 Qtcreator 安装后无法使用 msvc c++ 编译器。

传统方法

这使得现在在 Windows 上使用 Qt，在不手动配置下只剩下两种方式：

1. 放弃使用 msvc，使用 Qt with MinGW 版本。这样利用 Qt 安装包自带的 MinGW，可以避免环境配置
2. 安装完整版本的 visual studio，比如 visual studio 2019 community。完整版本的 vs 会自动配环境，所以也可以无需手动配置环境。

网上搜索教程大抵上述两种做法。但是，如果想在 Windows 上使用微软自己的编译工具链，又不想装庞大的完整 vs，就完了，基本上查网上发现可以仅安装 vs 生成工具，这是微软自己提供的单独的 c++ 编译器工具链。但是安装完后发现 Qtcreator 里无法自动发现 msvc 编译器，而现版本的 Qtcreator 又不能手动添加 msvc 类型的编译器，这样导致走不下去了，查阅了无数教程依然无果。直到发现篇讲解 Qtcreator 是如何探测 vs 环境的文章——[link](#)。

本教程原理解析

根据上文指出的答案，答主贴出了 Qtcreator 探测 vs 环境的关键代码，原图请看上述链接的知乎原图，看完代码发现（源码之下无秘密）100 Qtcreator 查找的逻辑是在指定注册表路径下找：

1. 如果是 32 位环境下， `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\VisualStudio\SxS\VS7`
2. 如果是 64 位环境下， `HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Microsoft\VisualStudio\SxS\VS7`

然后遍历 SxS 里面的项，

- 如果键名（其实就是数字）大于14（从15开始，即 msvc2017，那就把该键值（其实就是硬盘里的v安装路径）后面拼接上 `/VC/Auxiliary/Build` 查看此处是不是有 `vcvarsall.bat` 文件
- 否则（即键名小于等于14），那就把该键值后面拼接上 `/VC7` 查看此处是不是有 `vcvarsall.bat` 文件

上述 15 代表 msvc2017，16 代表 msvc2019，14 及更小的数字代表更老版本的 msvc，具体版本号对应关系参见

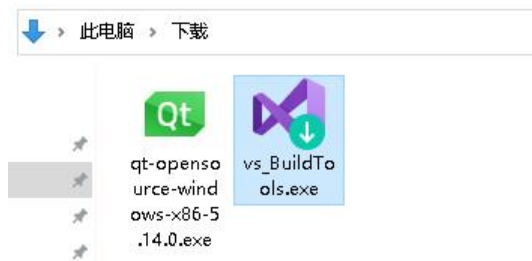
<https://www.cnblogs.com/zjutzz/p/6013783.html> 和 <https://docs.microsoft.com/zh-cn/visualstudio/releases/2019/release-notes> 此处的各版本发行说明

如果上述条件下能找到 **vcvarsall.bat** 文件，那 Qtcreator 就能自动配置好环境，下面是照做说明。

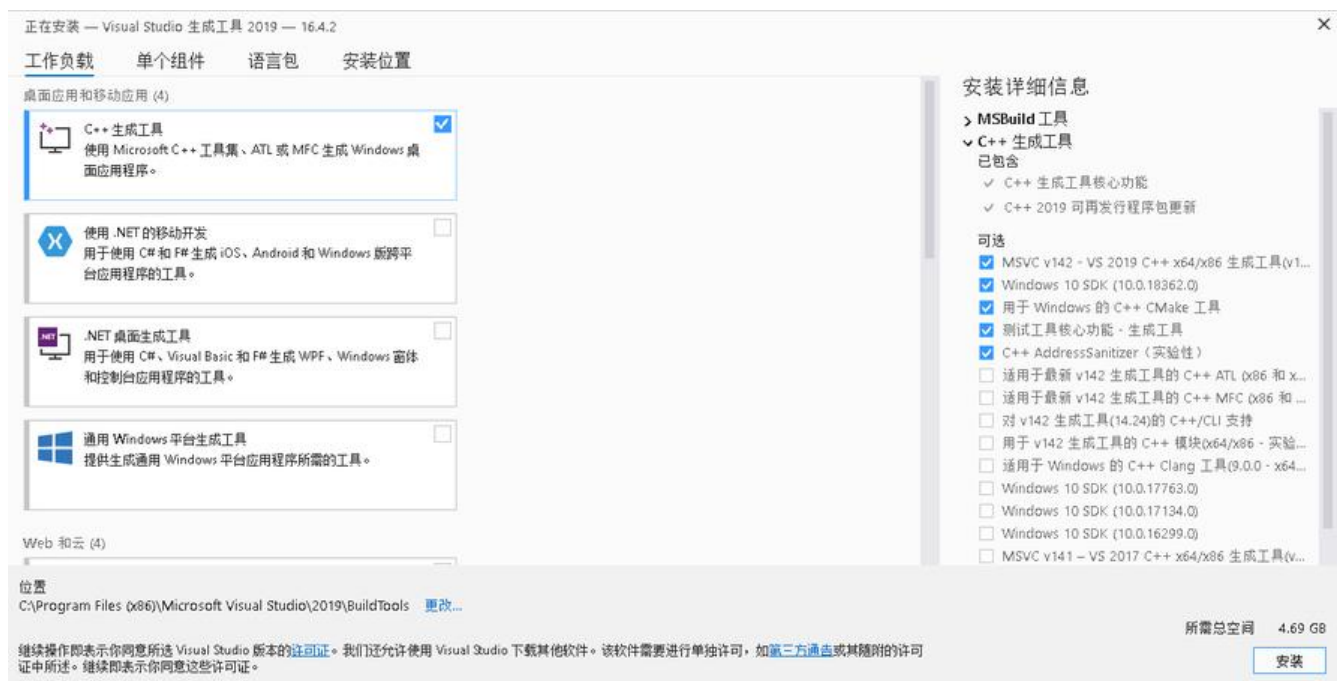
具体操作说明

1. 安装 msvc2019 生成工具

打开 <https://visualstudio.microsoft.com/zh-hans/downloads/>，翻到下面的「所有下载」，展开 Visual Studio 2019 工具」，找到「Visual Studio 2019 生成工具」，点击右边的下载。

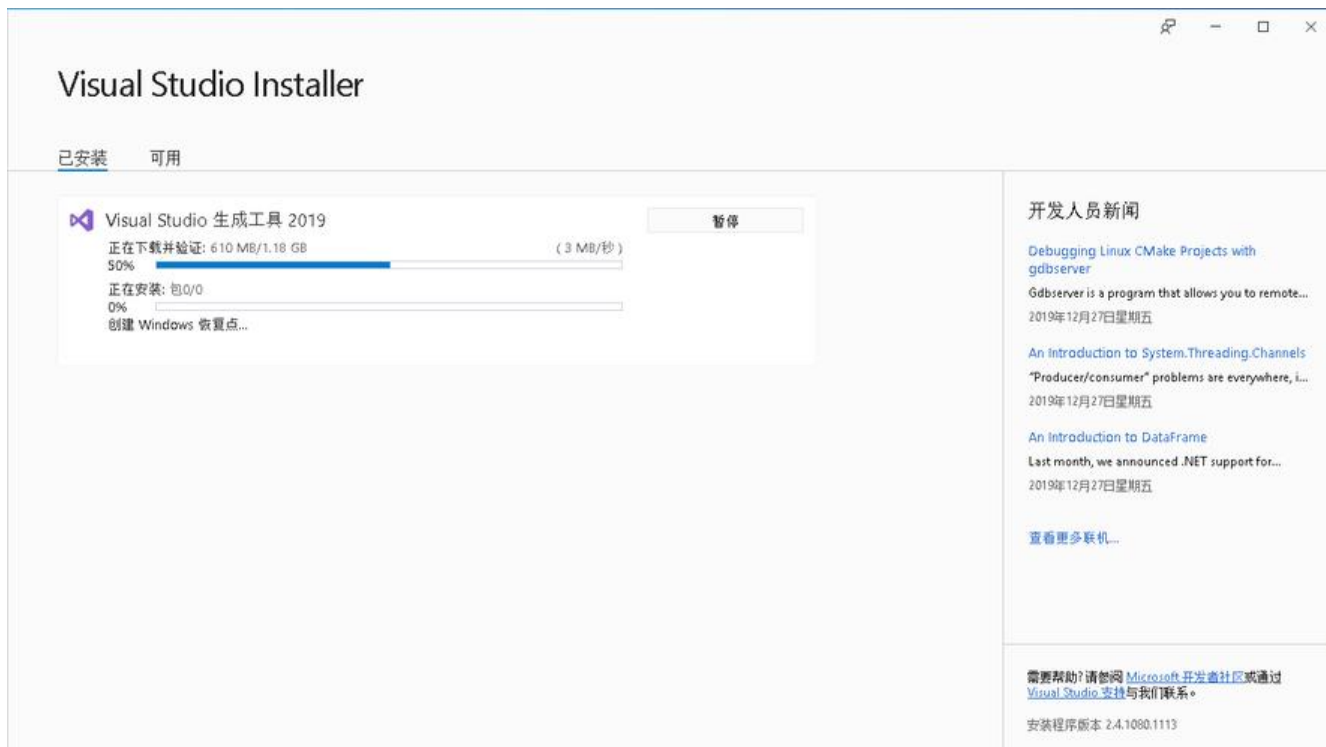


双击打开，我们只需要「C++ 生成工具」



PS. Qt 5.14 推荐采用 msvc 2017，上图右侧也有 2017 的可选项，但是该工具默认链接的是 2019 版并且也能正常使用，所以就先不折腾了，保持默认，只勾图中左侧一个勾，右侧会自动勾选默认的几选项。

然后就耐心等待联网下载和安装。



等安装完就可以关闭了，此后也用不到这里了。

2. 安装 Qt 5.14

基本傻瓜化操作。选择组件时看图即可。红色矩形框必选（至少32位/64位选择一个，对于初学者，勾选32位也行），大括号内容建议勾选。



Qt 5.14.0 安装程序

选择组件

请选择要安装的组件。

默认(A) 全选(S) 取消全选(D)

Qt 5.14.0 Prebuilt Components for MSVC 2017 32-bit

Qt

Qt 5.14.0

☐ MSVC 2015 64-bit

☒ MSVC 2017 32-bit

☒ MSVC 2017 64-bit

☐ MinGW 7.3.0 32-bit

☐ MinGW 7.3.0 64-bit

☐ UWP ARMv7 (MSVC 2015)

☐ UWP x64 (MSVC 2015)

☐ UWP ARMv7 (MSVC 2017)

☐ UWP x64 (MSVC 2017)

☐ UWP x86 (MSVC2017)

☐ Android

☐ Sources

☐ Qt Charts

☐ Qt Quick 3D (Technology Preview)

☐ Qt Data Visualization

☐ Qt Lottie Animation (Technology Preview)

☐ Qt Purchasing

☐ Qt Virtual Keyboard

☐ Qt WebEngine

☐ Qt Network Authorization

☐ Qt WebGL Streaming Plugin

☒ Qt Script (Deprecated)

☐ Qt Quick Timeline

Developer and Designer Tools

Qt Creator 4.11.0

☒ Qt Creator 4.11.0 CDB Debugger Support

☐ MinGW 7.3.0 32-bit

☐ MinGW 7.3.0 64-bit

☐ Strawberry Perl 5.22.1.3

此组件大约占用您硬盘驱动器 515.15 MB 的空间。

下一步(N)

取消

等待一会即可。

3. 配置环境 (关键步骤)

此时打开 Qtcreator 发现空空如也，果然找不到 msvc 编译器。关闭 Qtcreator。

Qt 选项 - Qt Creator

Filter

Kits

环境

文本编辑器

FakeVim

帮助

C++

Kits

构建套件(Kit) Qt Versions 编译器 Debuggers Qbs CMake

Name

Auto-detected

C++

C

Clang (C, x86 32bit in C:\Qt\Qt5.14.0\Tools\QtCreator\bin\clang\bin)

Manual

C++

C

添加

克隆

删除

Remove All

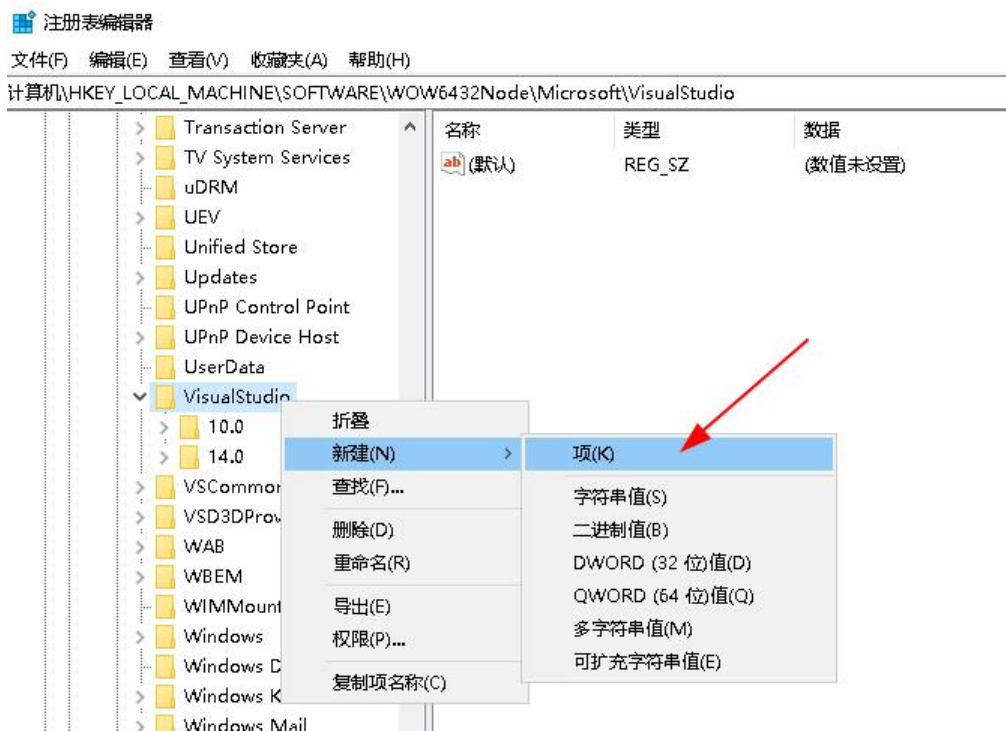
Re-detect

Auto-detection Settings...

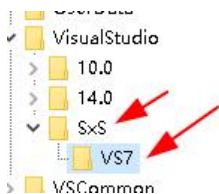
原文链接: [正确在 Windows 上配置 MSVC\(2019\) 作为 Qt 编译器](#)

此时开始做本文的核心操作：

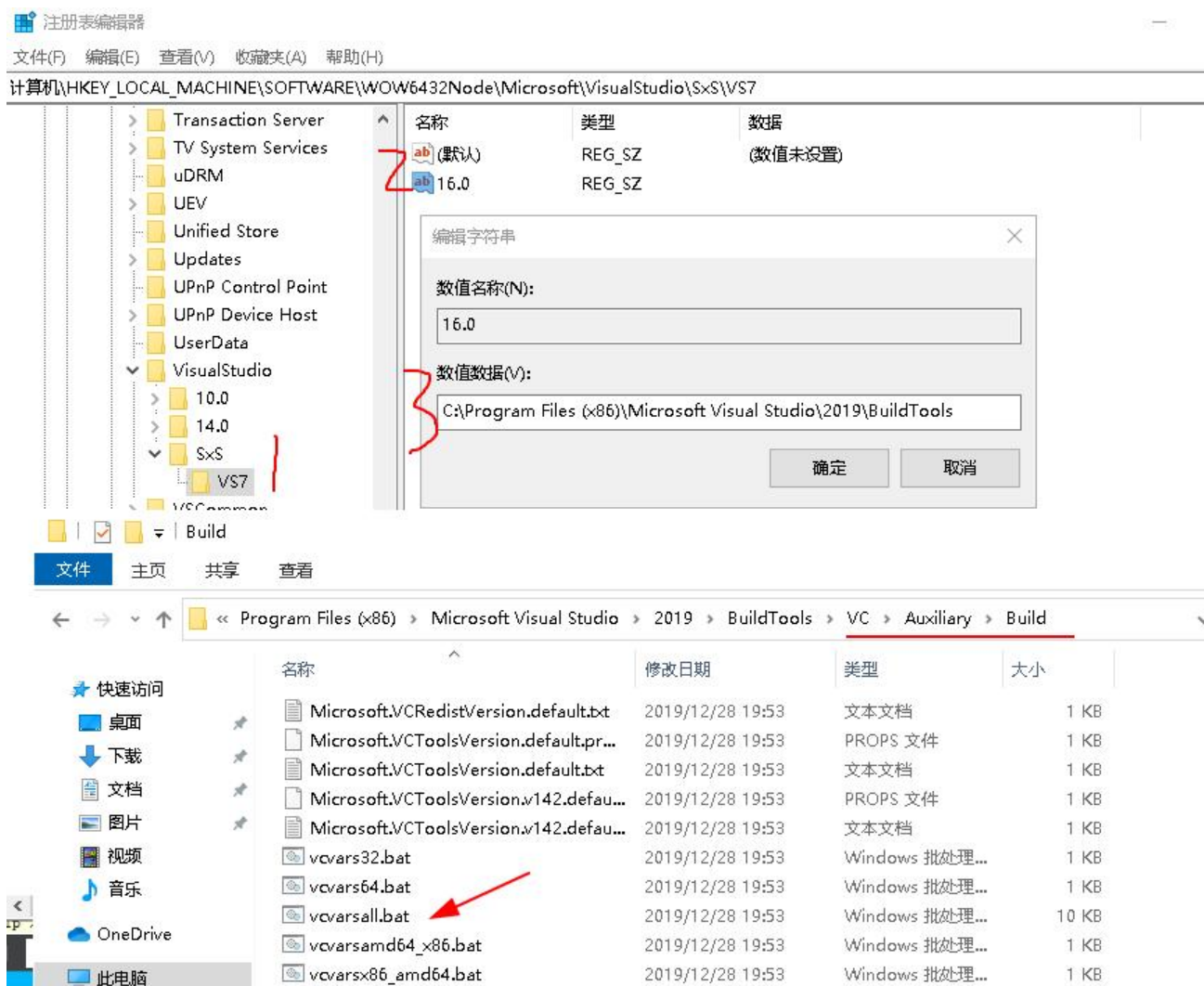
1. 打开注册表`HKEY\_LOCAL\_MACHINE\SOFTWARE\WOW6432Node\Microsoft\VisualStudio`
如果是32位系统，现在估计少见了，打开`HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\VisualStudio\SxS\VS7`)



连续新建两项，见下图，注意层次关系，注意大小写

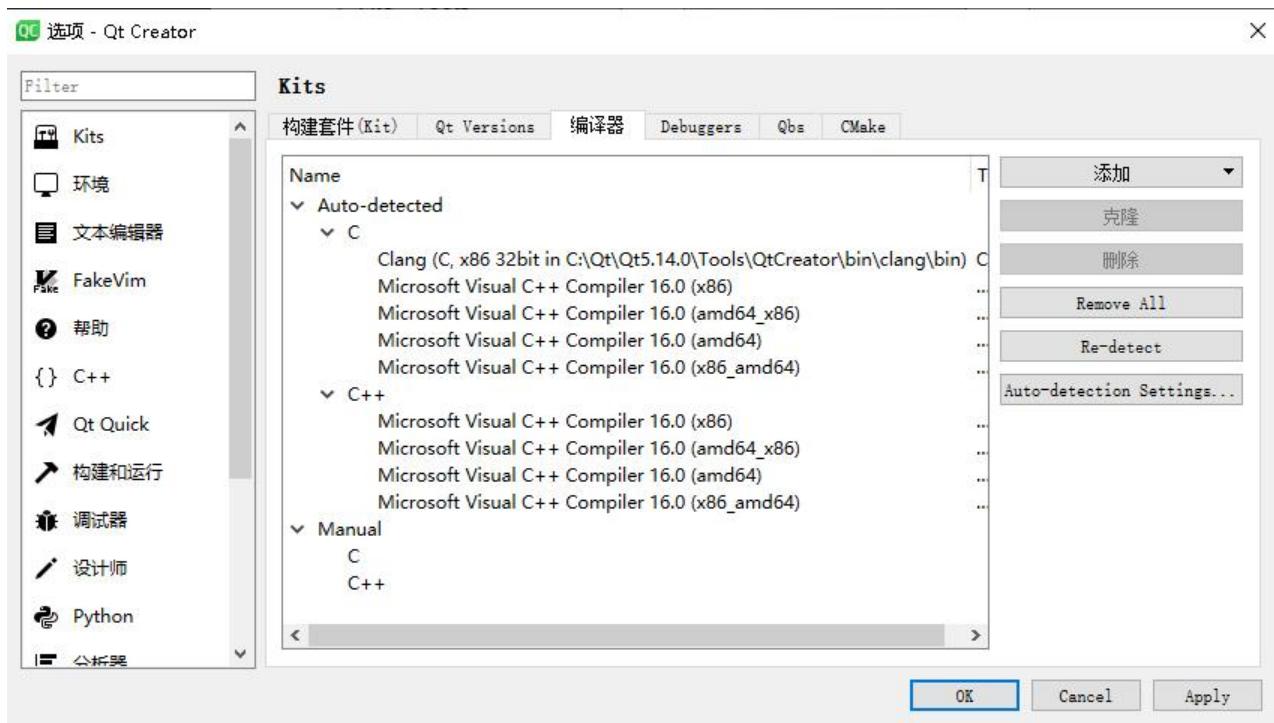


然后在 VS7 上新建字符串值，名称是16.0，值是 vs生成工具的安装目录下的一个子目录，具体看中3处和横线处（仔细看图，我没写错），要保证横线处的路径下有红色箭头处的文件。

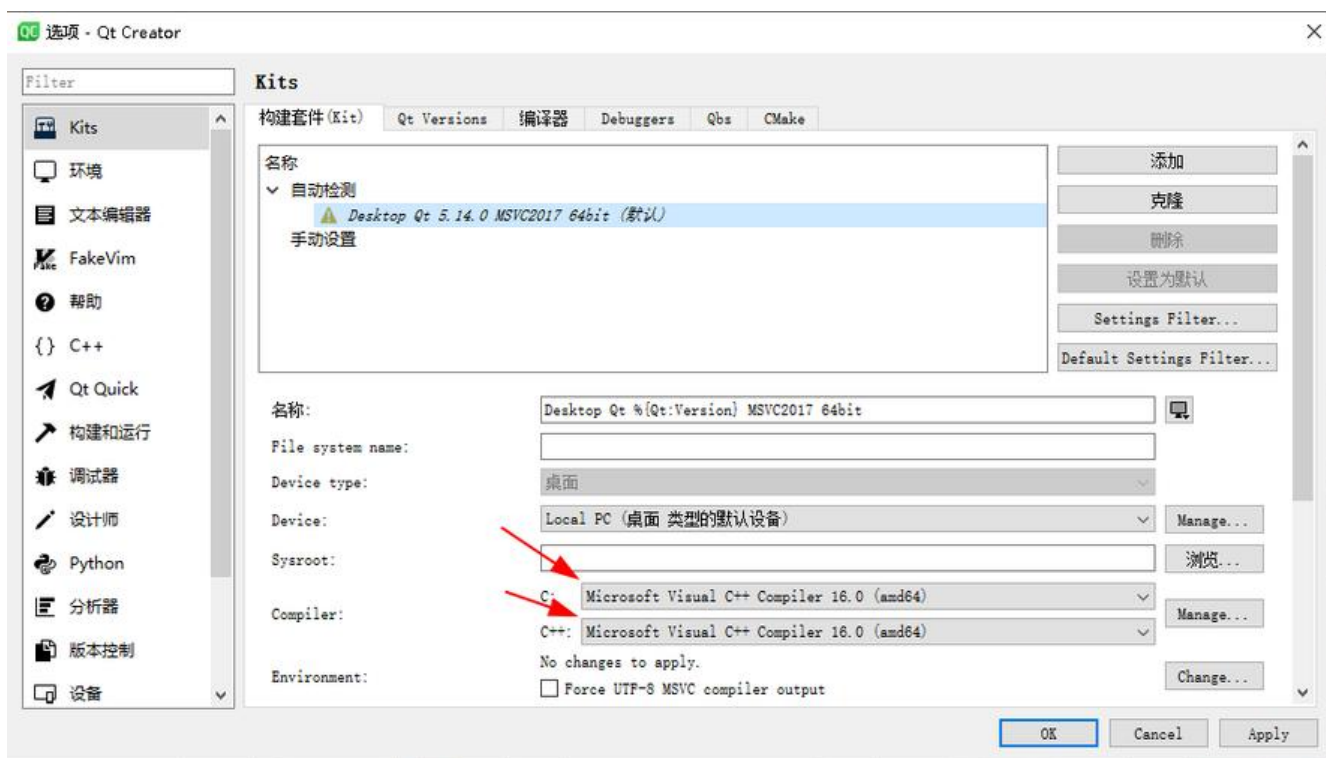


这里解释一下，16.0 代表是 msvc 的2019版，见上面的原理解析（但其实只要大于14即可，原因见上面的原理解析）

2. 再次打开 Qtcreator，神奇的事情发生了，Qtcreator 已自动发现并添加好了。



3. 接下来自然是去「构建套件 (Kit)」里选择相应编译器即可，看红色箭头。



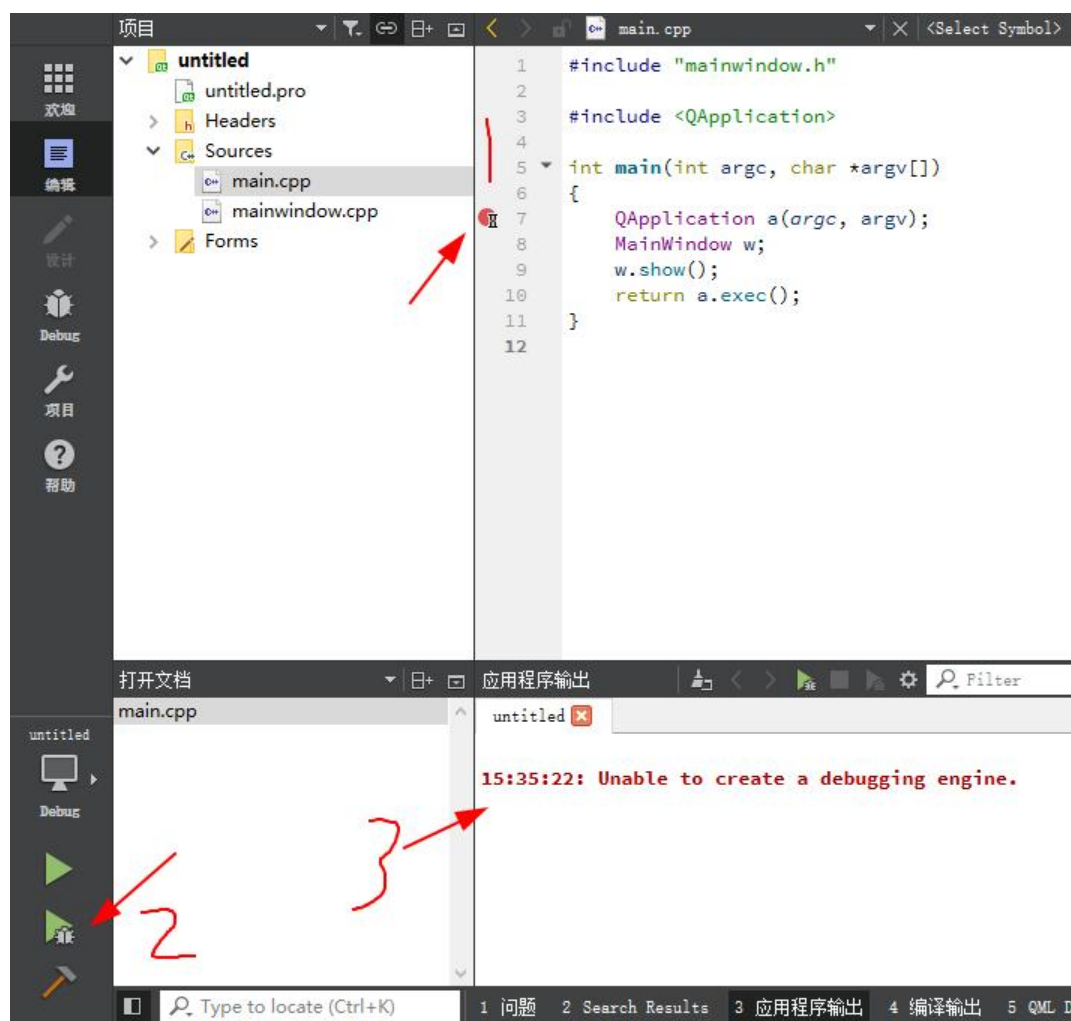
到此结束。如果想编译 32-bit 程序，那自行在这个界面点击「添加」并选择对应编译器，不再赘述。

4. 确认是否能编译程序

新建一个默认项目，看能否正常运行。

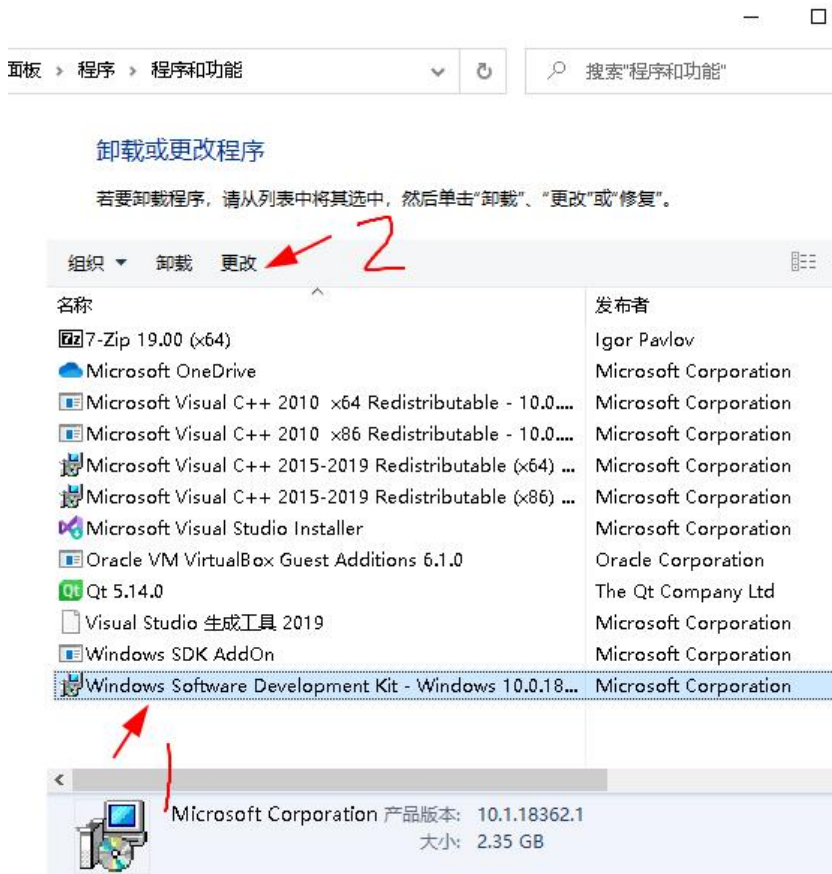
5. 添加单步调试支持。

到目前为止，已经可以运行程序了，但是无法单步调试。

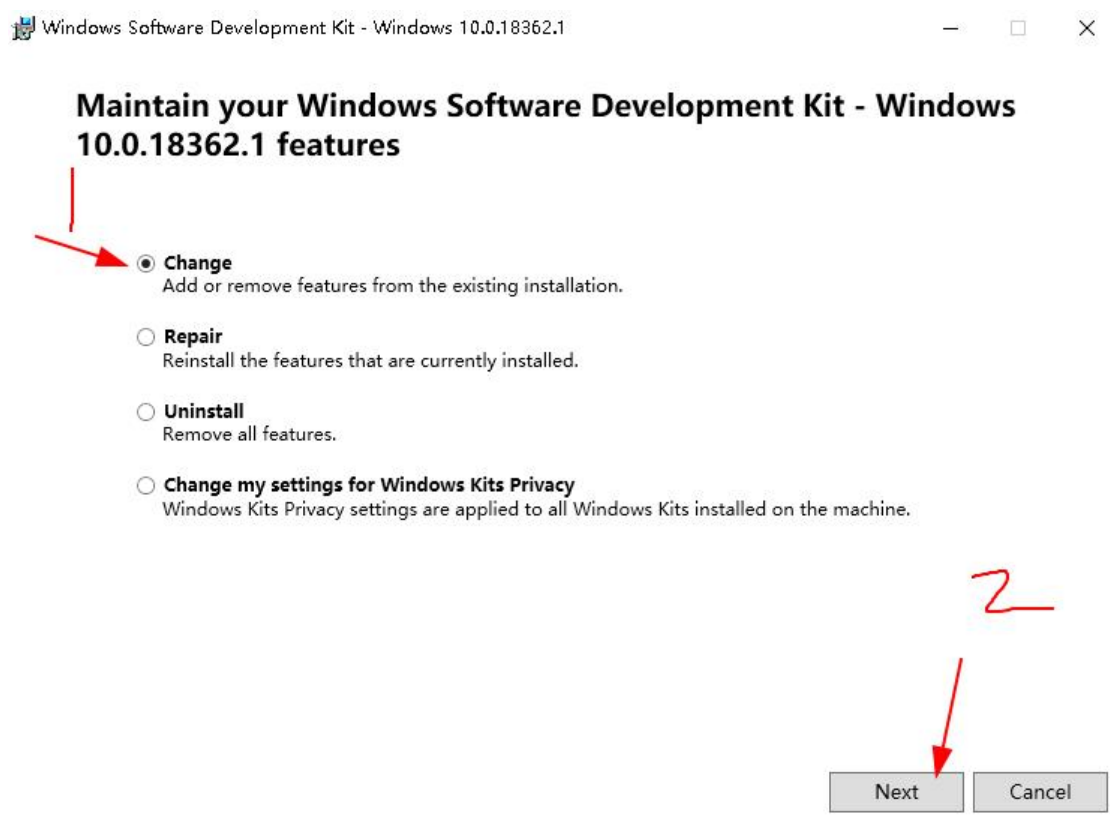


如果去网上一搜，又是一大堆误导、不适用的信息。其实 vs 生成工具自带了 cdb，但是却没有默认启用，这才是关键。

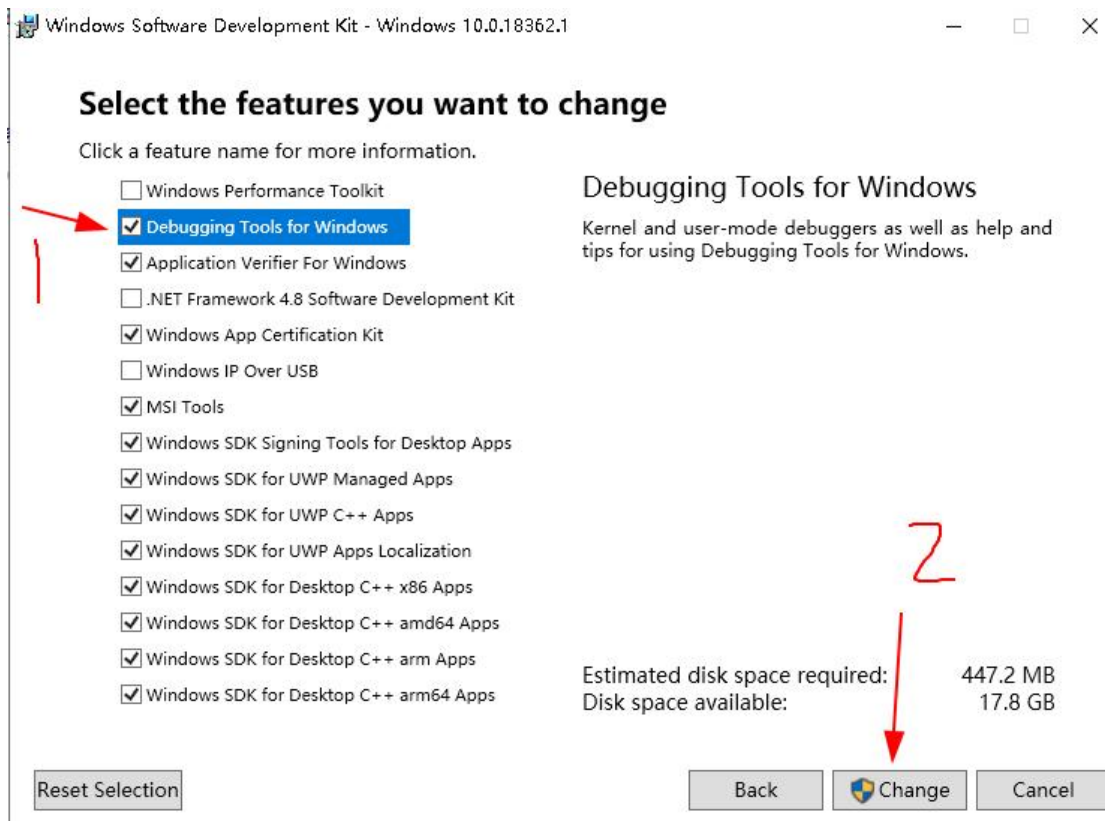
在控制面板——程序里找到`Windows SDK`，点击更改



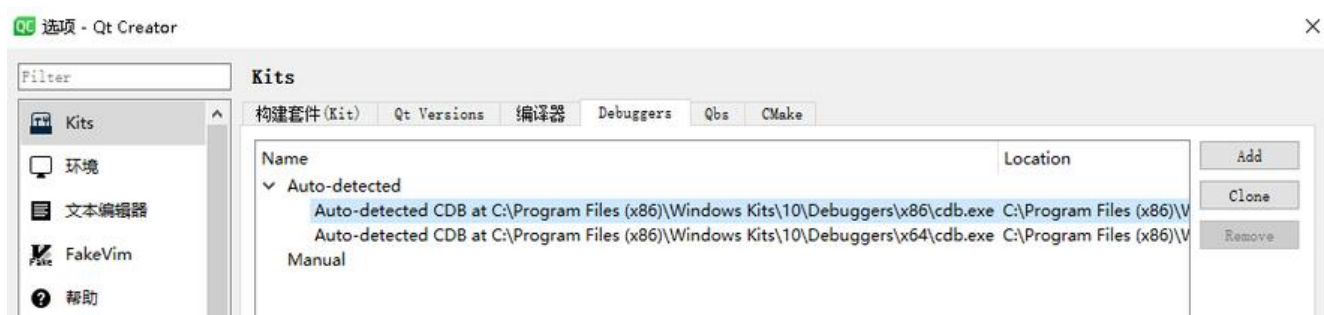
然后，点 Change



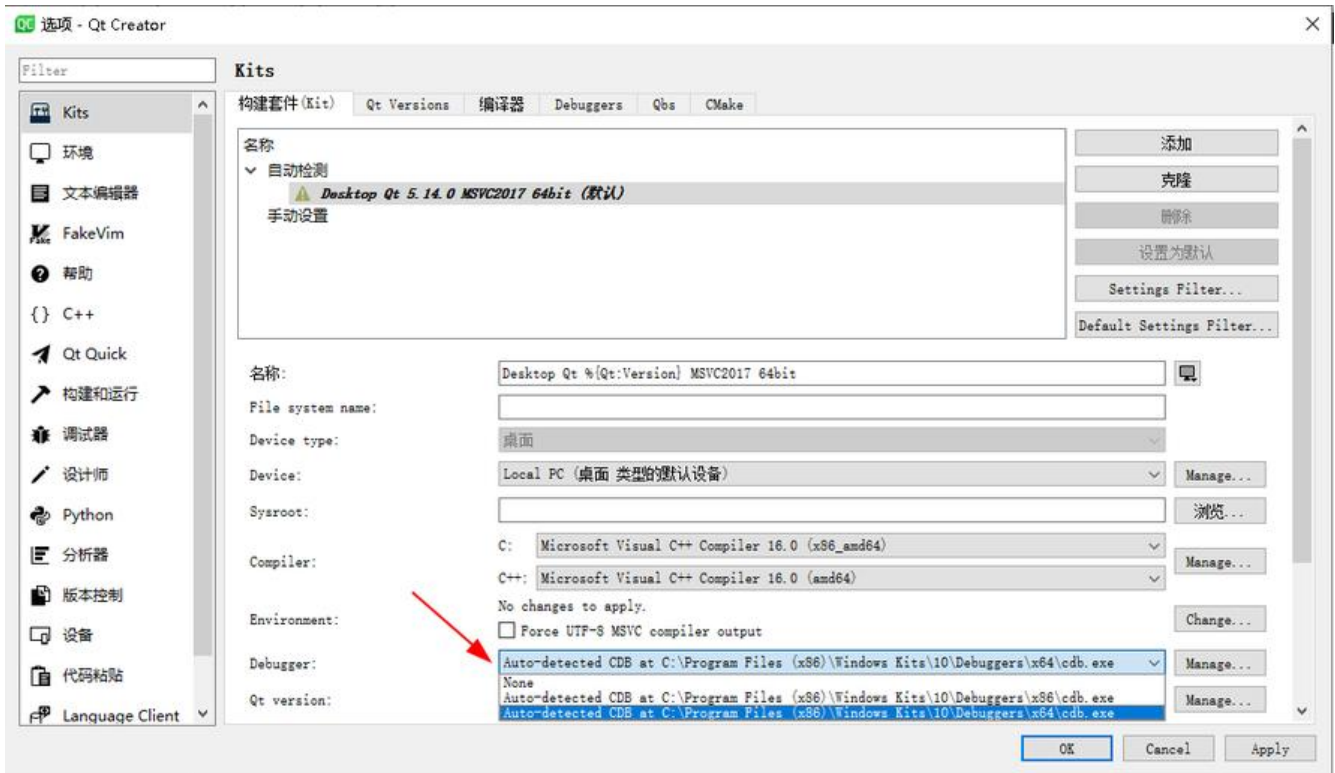
再然后，**勾选 debug tool (这是重点) **



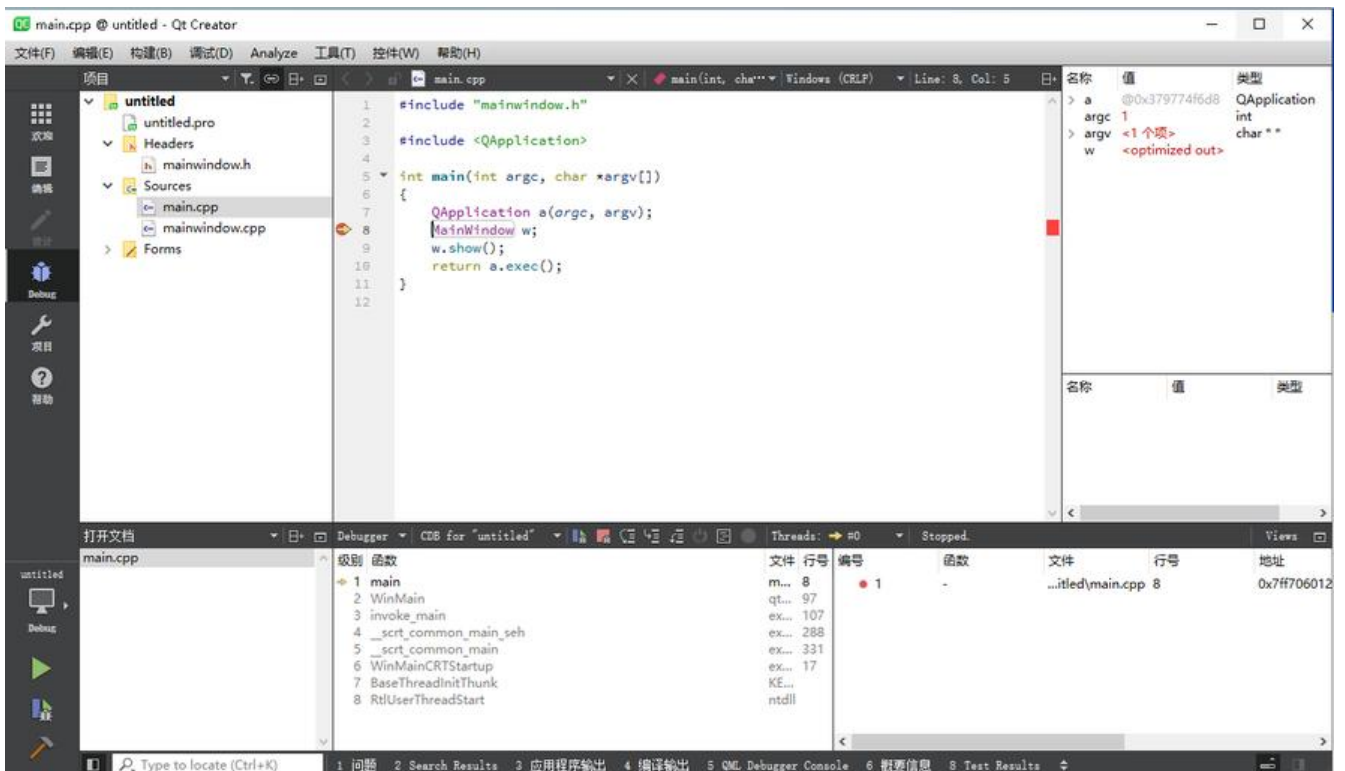
重新打开 Qtcreator, 发现已自动探测到 cdb



到「构建套件 (Kit)」选项卡配置好调试器, 注意选择32位还是64位, 要和编译器保持一致。



6. 确认是否能单步调试
看图，确实已经可以了。



全文终。