安装包名称命名规则

Qt安装包路径http://download.qt.io/official\_releases/qt/5.6/5.6.2/

[qt-opensource-windows-x86-msvc2013-5.6.2.exe](http://download.qt.io/official_releases/qt/5.6/5.6.2/qt-opensource-windows-x86-msvc2013-5.6.2.exe)

1）windows-x86代表使用于windows x86平台

2）msvc2013代表配套msvc2013编译器的配套组件(但是不包含msvc2013编译器)，编译器需要通过安装vs2013来完成安装。

3）5.6.2代表Qt库的版本

[qt-opensource-windows-x86-msvc2013\_64-5.6.2.exe](http://download.qt.io/official_releases/qt/5.6/5.6.2/qt-opensource-windows-x86-msvc2013_64-5.6.2.exe)

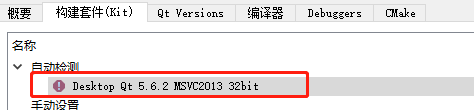
同上，这里的msvc\_64代表msvc2013编译器的64位版本的配套组建，适用于64位项目

[qt-opensource-windows-x86-mingw492-5.6.2.exe](http://download.qt.io/official_releases/qt/5.6/5.6.2/qt-opensource-windows-x86-mingw492-5.6.2.exe)

mingw492代表mingw492的配套组建。

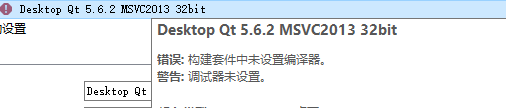
选择编译套件(kit)

选择 工具—选项—构建和运行，打开构建套件选项卡：



可以看到，如果是安装的[qt-opensource-windows-x86-msvc2013-5.6.2.exe](http://download.qt.io/official_releases/qt/5.6/5.6.2/qt-opensource-windows-x86-msvc2013-5.6.2.exe) ，那么构建套件中会自动将与msvc2013编译器相关的配套组件都添加进去。

注意，这里前面有个红色感叹号，悬浮上去可以看到相关提示：



这里可以看到是msvc2013没有设置，我们拉到下面找到编译器一栏：



发现没有匹配的编译器，即msvc2013在档期那机器上没有安装。那么需要先安装vs2013，然后再vs2013的安装目录中找到msvc2013，然后再指定即可。

此外，还可以再配置构建套件的时候，指定当前套件使用哪个版本的Qt库，指定调试器，等等：



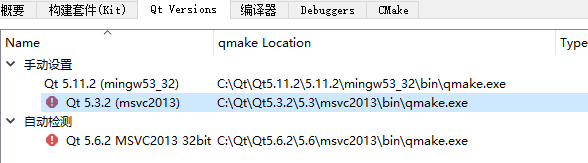
Qt库的版本

在同一台机器上，可以安装多个版本的Qt库及其配套组建(见安装包名称规则)：



这些库是不冲突的，就像上一节中提到的：“每个构建组建够可以指定自己Qt库的版本”。

可以在Qt Version选项卡中增加和删除当前IDE可用Qt库列表：



指定以后，便可以在构建套件中指定相应的Qt版本了。

编译器

Qt 支持很多中编译器：

GCC：linux和OS X系统的编译器

MinGW：GCC的windows版本，注意：这个编译器是和Qt Creator一并安装的，也就是说这个编译器是Qt Creator的内置编译器，只要安装Qt Creator，这个编译器就一并被安装，不需要像msvc那样需要单独安装。

Clang：Object C++ C C++等语言的融合编译器，三大系统中都可用，用的不普及，但是功能很强

QCC：QNX系统的C++编译器

其他编译器：这里就包括msvc

编译器的增加和删除可以在 工具—选项—构建和运行中，找到编译选项卡，然后在其中操作，需要只当编译器的exe文件的具体位置。

小结：

Qt Creator对于工程构建套件的配置是相当灵活的，主要就是几方面：

1. Qt 库版本可配，对于当前工程，可以灵活切换使用哪个版本的Qt库
2. 编译器可配，可以选择使用什么样的编译器来构建当前工程
3. 编译器的配套组建可配，这个一般是跟着编译器走，不然会报错，这里也算不上可配
4. 调试工具可配，比如可以用windbg，gdb等，和编译器的配置类似
5. 对于整个构成的编译套件可配，即对于1，2，3，4的灵活组合

注意，一定要确保工程的其他第三方库支持的位数和编译器的位数要匹配的上