

[Qt创作者手册](#) > [常见问题](#)

常见问题

本节包含有关Qt Creator的一些常见问题的解答。您还可以在“[已知问题](#)”和“[操作方法](#)”部分或特殊区域（如[调试](#)）的“[疑难解答](#)”部分找到问题的答案。

一般问题

如何重置所有Qt Creator设置？

删除Qt Creator创建的设置文件。

有关文件在每个受支持的平台上的位置的详细信息，请参阅[设置文件的位置](#)。

Qt Creator 自带 MinGW，我应该在 Qt 中使用这个版本吗？

使用针对 Qt 版本构建的版本。

Qt Creator找不到辅助应用程序，例如Git或编译器。我该怎么办？

启动Qt Creator时，请确保应用程序位于系统路径中。此外，选择[编辑](#)>[首选项](#)以检查为应用程序指定的设置。许多插件指定了所需工具的路径或运行环境。

这对于macOS尤其重要，当Qt Creator启动时，macOS可能不在路径中。`/usr/local/bin`

如何更改Qt Creator的界面语言？

Qt Creator已经本地化为多种语言。如果系统语言是受支持的语言之一，则会自动选择它。要更改语言，请选择[编辑](#)>[首选项](#)>[环境](#)，然后在[语言](#)字段中选择一种语言。选择[立即重启](#)以重启Qt Creator并使更改生效。

报告的问题是否已得到解决？

您可以在[Qt项目错误跟踪器](#)中查找任何问题。

Qt设计器集成问题

为什么自定义小部件没有在设计模式下加载，即使它在独立的Qt Designer中工作？

Qt Designer从标准位置获取插件，并加载与其构建键匹配的插件。独立和集成Qt Designer的位置不同。

有关更多信息，请参阅[添加 Qt 设计器插件](#)。

为什么即使我有模块，QML导入路径也有一条红线：

默认情况下，Qt Creator 会查找 Qt 的 QML 导入路径以获取 QML 模块。有时，它没有得到正确的处理，你需要告诉它模块在哪里。使用 qmake 作为构建系统时，请在应用程序的文件中指定。使用 CMake 时，请将命令添加到 CMakeLists.txt 文件中。QML_IMPORT_PATH.proset

这还可以实现 QML 代码的代码完成并删除错误消息。

以下示例说明了如何指定 qmake 项目的导入路径，以便在不同目标平台的生成和运行工具包之间切换时使其正常工作：

```
TEMPNAME = ${QMAKE_QMAKE}
MY_QTPATH = ${dirname(TEMPNAME)}
QML_IMPORT_PATH += ${MY_QTPATH}/../qml
message("my QML Import Path: "${QML_IMPORT_PATH})
```

有关如何设置 CMake 项目的导入路径的更多信息，请参阅[导入 QML 模块](#)。

当Qt Creator抱怨缺少OpenGL支持时，我该怎么办？

Qt Creator的某些部分，如QML Profiler，使用Qt Quick 2，它依赖于OpenGL API进行绘制。不幸的是，使用 OpenGL 可能会导致问题，尤其是在远程设置和过时的驱动程序中。在这些情况下，Qt Creator会在控制台上显示与OpenGL 相关的错误消息，或将它们记录在Windows调试器日志中。

修复和解决方法因设置而异。作为最后的手段，您可以禁用受影响的插件。

虚拟机

尝试在虚拟机的设置中启用3D 加速。对于VirtualBox，还要确保已安装来宾插件，包括实验性的Direct3D支持。

窗户

检查Qt Creator是否已经用OpenGL/Desktop或ANGLE作为后端编译。官方二进制文件总是使用 ANGLE（一个将 OpenGL ES API 映射到 DirectX 的库）构建的。

- › ANGLE后端：这需要比Windows XP更新的Windows版本。如果遇到问题，请尝试更新图形驱动程序或更新 DirectX 版本。运行以检查是否确实启用了Direct3D 加速。dxdiag.exe
- › OpenGL 后端：确保您的图形驱动程序支持 OpenGL 2.1 或更高版本。尝试更新图形驱动程序。

Unix

运行以快速检查OpenGL是否正常工作。检查输出以获取更多详细信息，例如 OpenGL 驱动程序和渲染器（在应用程序的输出中搜索OpenGL）。glxgears|glxinfo

如果您使用的是Mesa驱动程序，则可以通过设置环境变量来强制OpenGL在软件中呈现。
LIBGL_ALWAYS_SOFTWARE

禁用插件

您可以禁用Qt Creator插件，但代价是失去功能：

- › 从命令行启动Qt Creator，带有参数。-noload QmlProfiler -noload QmlDesigner
- › 通过选择帮助>关于插件永久禁用插件。

帮助问题

Qt Creator与Qt文档和使用Qt帮助插件的示例元集成。集成的Qt参考文档适用于Qt 4.4及更高版本。Qt Creator和其他Qt可交付成果包含.qch文件形式的文档。所有文档都可以在**帮助**模式下访问。

要查看可用的文档并添加文档，请选择**编辑>首选项>帮助>文档**。有关更多信息，请参阅[添加外部文档](#)。

调试器问题

有关调试器疑难解答的信息，请参阅[调试器疑难解答](#)。

Qt创建者手册3.0.2

Topics >

在Linux和Windows上，使用安装Qt Creator和Qt时安装的启用Python的GDB版本。在macOS上，GDB不再受官方支持。要构建您自己的支持Python的GDB，请按照[构建GDB](#)中的说明进行操作。

您必须使用Python版本2.6或2.7。

有关设置调试器的详细信息，请参阅[设置调试器](#)。

如何在Qt Creator中生成核心文件？

若要触发在调试时生成核心文件的GDB命令，请选择**查看>视图>调试器日志**。在**命令**字段中，键入并按Enter。核心文件在当前工作目录中创建。您可以指定文件的另一个位置，包括相对路径或绝对路径，作为命令的参数。
gcore

编译器问题

如何通过Qt Creator使用我的多核CPU？

在Linux和macOS上，转到**项目**模式，在**构建设置**中选择您的配置，找到**构建步骤**，然后添加以下值，其中CPU中的内核数量：`<num>-j <num>`

在Windows上，nmake不支持该参数。相反，我们提供了一个名为jom的直接替代品。您可以从[Qt Downloads](#)下载jom的预编译版本。将jom.exe放在%PATH中的某个位置。转到**生成设置**并将jom.exe设置为make命令。`-j`

注意：与GNU make不同，jom会自动检测您的内核，并生成与CPU拥有的内核一样多的并行进程。您可以使用上述参数覆盖此行为。`-j`

Qt安装问题

我不能将QSslSocket与我从二进制包安装的Qt一起使用。我该怎么办？

二进制包中的Qt是用定义QT_NO_OPENSSL构建的。重建它是可能的。有关详细信息，请参阅<http://www.qtcentre.org/threads/19222-Qssl>。

在Ubuntu或Debian上需要发行版中的哪些开发包？

```
sudo apt-get install libglib2.0-dev libSM-dev libxrender-dev libfontconfig1-dev libxext-dev
```

如果你使用QtOpenGL，你还需要：

平台相关问题

Qt Creator中的应用程序输出显示在哪里？

在Unix (Linux和macOS) 上，相关函数使用标准输出和错误输出。运行或调试应用程序时，可以在“[应用程序输出](#)”中查看输出。qDebug()

对于需要输入的控制台应用程序，请选择“**项目>运行设置>在终端中运行**”。要指定要使用的终端，请选择**编辑>首选项>环境>系统**。

在视窗上：控制台应用程序和GUI 应用程序的输出显示方式不同。

对于 qmake 项目，.pro 文件中的设置指定使用其他运行时将应用程序构建为控制台应用程序。CONFIG += console

这是 CMake 项目的标准行为。若要在 Windows 上创建 GUI 应用程序或在 macOS 上创建应用程序捆绑包，必须在 CMakeLists.txt 文件中指定qt_add_executable命令的 theorproperty 。WIN32MACOSX_BUNDLE

运行控制台应用程序时，可以在调用应用程序的控制台窗口中查看输出。如果调用应用程序是GUI应用程序（例如，Qt Creator的发布构建版本），则会打开一个新的控制台窗口。对于这种类型的应用程序，相关功能使用标准输出和错误输出。qDebug()

建议为控制台应用程序选择“**项目>运行设置>在终端中运行**”。

对于GUI应用程序，相关功能使用Windows API功能。输出显示在“**应用程序输出**”中。但是，Qt Creator一次只能显示一个源的输出，以便正确显示。可以使用外部调试输出查看器（如[Windows 调试视图](#)）来显示 GUI 应用程序的输出。qDebug()OutputDebugString()

有关新功能的问题

请求的功能是否会实现？

如果是计划功能，您可以在任务跟踪器中看到此信息。如果某个功能已经实现，则会在即将发布的版本的[更改文件](#)中提及该功能。

为什么Qt Creator不使用编辑器的标签？

这个问题不时出现，所以我们仔细考虑过。以下是我们不使用选项卡的主要原因：

- › 选项卡不缩放。如果您打开了 5 到 6 个编辑器，它们就可以正常工作，使用 10 个编辑器会变得很麻烦，如果您需要比选项卡栏更多的水平空间，则界面根本不起作用。
- › 选项卡不适应您的工作集。
- › 常见的解决方案是让用户能够对选项卡重新排序。现在用户必须管理选项卡而不是编写代码。
- › 选项卡迫使您限制打开的编辑器的数量，否则您会感到困惑。

考虑以下用例：*开发人员想要切换编辑器。*

事实上，开发人员不想切换编辑器，但可能必须这样做才能完成他们的任务。我们需要弄清楚任务是什么，以便为开发人员提供更好的导航方法，同时执行任务。

许多用例中的一个常见因素是在处理一组打开的文件时切换编辑器。在处理文件 A 和 B 时，用户有时需要查看文件 C。他们可以按Ctrl+Tab在文件之间移动，并根据文件类型在正确的编辑器中打开文件。列表按上次使用时间排序。

通常，用户还处理多个相关的类或函数，即使它们是在不同的文件中定义或声明的。Qt Creator为此提供了两个快捷方式：[跟随光标下的符号](#)（Ctrl+Shift+J）和[查找对它的引用](#)（Ctrl+Shift+F）。

- › 按F4在标头和源之间切换。
- › 按Alt+Left在导航历史记录中向后移动。
- › 使用定位器 (Ctrl + K) 简单地告诉Qt Creator去哪里。

定位器可用于打开文件，但打开文件也只是完成任务的一个步骤。例如，考虑以下用例：修复来自 `someclass.cpp/someclass.h` 的 `SomeClass` 中的 `AFunction`。

使用选项卡式用户界面，开发人员将在选项卡栏中搜索某个类.cpp然后搜索，只是发现该功能不在该文件中。然后他们会在标签栏中搜索someclass.h，找到我们的函数是内联的，解决问题，然后忘记它们来自哪里。::AFunction

使用Qt Creator，开发人员可以键入以查找函数。通常，他们只需要键入函数名称的 3 到 4 个字符。然后，他们可以解决问题并按Alt+Back返回原来的位置。Ctrl+K m AFun

其他定位器过滤器包括 for 类、用于所有符号和（感谢社区贡献）当前文件中的符号。c:.

‹ 使用帮助模式

操作方法 ›

©2022 Qt有限公司 此处包含的文档贡献的版权归 他们各自的所有者。此处提供的文档根据自由软件基金会发布的GNU 自由文档许可证版本 1.3的条款进行许可。Qt和相应的徽标是Qt有限公司在芬兰和/或全球其他国家的商标。所有其他商标均为财产其各自所有者。



联系我们

公司

关于我们
投资者
编辑部
职业
办公地点

发牌

条款和条件
开源
常见问题

支持

支持服务
专业服务
合作伙伴
训练

对于客户

支持中心
下载
Qt登录
联系我们
客户成功案例

为Qt做贡献
论坛
维基
下载
市场

© 2022 Qt公司

[反馈](#) [登录](#)