

🔍 搜索

Qt创作者手册 > [设置调试器](#)

设置调试器

主调试器设置与生成和运行项目的[工具包](#)相关联。若要指定要用于每个工具包的调试器和编译器，请选择“**编辑>首选项>工具包**”。

仅当自动安装失败时，才需要设置调试器，因为缺少本机调试器（通常 Windows 上的 CDB 调试器就是这种情况，您始终必须自己安装）或因为安装的版本不受支持（例如，当您的系统不包含 GDB 或过时版本，并且您希望改用本地安装的替换版本时）。

注意：如果需要更改调试器以用于自动检测的工具包，可以[克隆工具包](#)并更改克隆中的参数。确保为项目选择克隆的工具包。

Qt创建者手册8.0.2
Topics >

注意：若要使用 Windows 的调试工具，必须安装它们并将 Microsoft 提供的符号服务器添加到调试器的符号搜索路径中。有关详细信息，请参阅[在 Windows 上设置 CDB 路径](#)。

注意：要在 macOS 上使用 GDB 的自由软件基金会（FSF）版本，您必须对其进行签名并修改[套件](#)设置。

本节介绍用于调试C++代码的选项，并提供受支持的本机调试器的安装说明。它也适用于其他编译语言的代码，如C，FORTRAN，Ada。

有关调试器模式的更多信息，请参见[在不同模式下启动调试器](#)。

支持的本机调试器版本

Qt Creator在处理编译代码时支持本机调试器。在大多数支持的平台上，可以使用 GNU 符号调试器 GDB。在 Microsoft Windows 上，使用 Microsoft 工具链时，需要 Microsoft Console Debugger CDB。在macOS和Linux上，可以使用LLDB调试器。

下表总结了对调试C++代码的支持：

平台	编译器	本机调试器
Linux目录	海湾合作委员会，国际商会	GDB，LLDB
Unix	海湾合作委员会，国际商会	GNU CDB

视窗/MinGW	海湾合作委员会	广东元
视窗/MSVC	Microsoft Visual C++ Compiler	适用于 Windows/CDB 的调试工具

支持的 GDB 版本

从3.1版本开始，Qt Creator需要Python脚本扩展。没有 Python 脚本的 GDB 构建不再受支持，并且无法正常工作。支持的最低版本是使用 Python 版本 2.7、3.3 或更高版本的 GDB 7.5。

对于使用 GDB 和 GDB 服务器的远程调试，目标[设备上](#)支持的最低 GDB 服务器版本为 7.0。

支持的 CDB 版本

Qt支持的所有版本的CDB定位平台都由Qt Creator支持。

支持的 LLDB 版本

LLDB 本机调试器具有与 GDB 调试器类似的功能。LLDB 是 macOS 上 Xcode 中的默认调试器，用于支持桌面上的C++。LLDB通常与Clang编译器一起使用（即使您也可以将其与GCC一起使用）。

在 macOS 上，您可以使用 Xcode 附带的 LLDB 版本或从源代码构建。支持的最低版本为 LLDB 320.4。

在Linux上，支持的最低版本是LLDB 3.8。

安装本机调试器

以下部分提供有关安装本机调试器的信息。

广东元

在Windows上，使用与Qt软件包捆绑在一起或最新版本的MinGW附带的支持Python的GDB版本。在大多数 Linux 发行版上，系统附带的 GDB 版本就足够了。

您还可以按照[构建 GDB](#) 中的说明构建自己的 GDB。

macOS 上不再支持 Xcode 附带的 GDB 版本。

适用于 Windows 的调试工具

若要使用 CDB 调试器，必须安装[适用于 Windows 的调试工具](#)。您可以从 Windows SDK 中的“[下载并安装适用于 Windows 的调试工具](#)”下载它们。

注意： Visual Studio 不包括所需的调试工具，因此，您必须单独安装它们。

此外，在安装 Qt 或独立的 Qt Creator 时，您必须选择 **QtCreator CDB 调试器支持**（在 Qt>Tools>Qt Creator 中）。

使用Microsoft Visual C++ Compiler手动构建Qt Creator时，构建过程会检查所需的文件。"%ProgramFiles%\Debugging Tools for Windows"

强烈建议您将 Microsoft 提供的符号服务器添加到调试器的符号搜索路径中。符号服务器提供用于调试 Windows 应用程序的操作系统库的调试信息。有关详细信息，请参阅[在 Windows 上设置 CDB 路径](#)。

Qt二进制发行版包含库的调试和发布变体。但是，即使应用程序编译为debug，也必须显式告诉运行时链接器要使用调试库，因为发布是默认库。

如果在Qt Creator中使用基于qmake的项目，则可以在**项目**模式下的**运行配置**中设置一个标志。在运行配置中，选择**使用框架的调试版本**。

有关在 macOS 上进行调试的更多详细信息，请参阅：[Mac OS X 调试魔法](#)。

LLDB

我们建议使用随最新 Xcode 一起提供的 LLDB 版本。

为 macOS 设置 FSF GDB

要在 macOS 上使用 FSF GDB，您必须对其进行签名并将其添加到 Qt Creator **工具包** 中。

1. 要创建用于对 FSF GDB 进行签名的密钥，请选择**“钥匙串访问”>证书助手>创建证书**：

1. 在**“名称”**字段中，输入fsfgdb以替换现有内容。
2. 在**“证书类型”**字段中，选择**“代码签名”**。
3. 选中**“让我覆盖默认值”**复选框。
4. 选择**“继续”**，然后按照向导的说明进行操作（使用默认设置），直到**“指定证书的位置”**对话框打开。
5. 在**“钥匙串”**字段中，选择**“系统”**。
6. 选择**“钥匙串访问”>系统**，然后找到证书。
7. 双击证书以查看证书信息。
8. 在**“信任”**部分，选择**“使用此证书时”**字段中的**“始终信任”**，然后关闭对话框。

2. 要对二进制文件进行签名，请在终端中输入以下命令：

```
codesign -f -s "fsfgdb" $INSTALL_LOCATION/fsfgdb
```

3. 在Qt Creator中，选择Qt Creator>Preferences>Kits>Add以创建使用FSF GDB的工具包。
4. 在**“调试器”**字段中，指定 FSF GDB 的路径（但具有显式值）。\$HOME/gdb72/bin/fsfgdb\$HOME
5. 若要使用调试器，请在项目的**“生成设置”**中添加工具包。

[< 调试](#)

[启动调试器 >](#)

©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

支持

- 支持服务
- 专业服务
- 合作伙伴
- 训练

社区

- 为Qt做贡献
- 论坛
- 维基
- 下载
- 市场

发牌

- 条款和条件
- 开源
- 常见问题

对于客户

- 支持中心
- 下载
- Qt登录
- 联系我们
- 客户成功案例