

[Qt 6.4](#) > [使用 CMake 构建](#) > [在命令行上生成项目](#)

在命令行上生成项目

本页介绍如何配置和生成现有项目。如果您想知道如何创建基于 Qt 的 CMake 项目，请参阅[有关如何开始使用 CMake](#) 的文档。

要构建Qt项目，CMake需要知道Qt安装的位置。通常，这是通过将CMake变量`CMAKE_PREFIX_PATH`设置为Qt的安装前缀来完成的。如果要交叉编译，请参阅[交叉编译](#)，了解有关需要设置的其他变量的详细信息。

如果使用在线安装程序安装Qt，请在顶级安装目录中选择Qt版本。例如，以下命令显示了如何在 Windows 上完成此操作：

```
cmake -DCMAKE_PREFIX_PATH=C:\Qt\6.4.0\msvc2019_64 -S <source-dir> -B <build-dir>
```

[Topics](#) >

中天发电机

CMake 生成必要的构建系统文件，使 GNU Make 或 Ninja 等构建工具能够构建您的项目。

CMake 的默认生成器取决于平台和构建环境。例如，在Windows上，如果检测到Visual Studio环境，CMake会生成Visual Studio项目文件。

为了在所有平台上获得一致的开发人员体验，请使用理论生成器。NinjaNinja Multi-Config

您可以通过设置环境变量或使用参数来选择 CMake 生成器：CMAKE_GENERATOR-G

```
cmake -G Ninja ...
```

Qt-cmake

脚本是配置项目的便捷替代方法。它消除了您指定的需要。您可以在Qt安装前缀的目录中找到它。该脚本将所有参数传递给 CMake，因此您可以像使用它一样使用它：qt-cmakeCMAKE_PREFIX_PATHbincmake\

```
C:\Qt\6.4.0\msvc2019_64\bin\qt-cmake -G Ninja -S <source-dir> -B <build-dir>
```

生成生成系统文件后，即可生成项目：

```
cd <build-dir>
ninja
```

您还可以使用独立于生成器的 CMake 命令：

交叉编译

为与开发计算机不同的平台构建项目称为交叉编译。一个例子是在Windows机器（主机平台）上为Android（目标平台）构建。

对于大多数平台，使用 CMake 进行交叉编译需要一个[工具链文件](#)。除了目标平台的Qt版本外，它还需要开发主机的Qt版本。例如，您需要安装 Qt for Windows 和 Qt for Android 才能在 Windows 上为 Android 交叉编译。

从目标平台的Qt安装中使用，为该平台交叉编译您的项目：qt-cmake

```
<target-qt>/bin/qt-cmake -S <source-dir> -B <build-dir>
```

这将为目标平台配置项目。工具链文件会自动传递，并可能设置其他特定于平台的变量。

指定自定义工具链文件

该脚本将一个 Qt 内部[工具链文件](#)传递给 CMake。该工具链文件设置了几个特定于Qt目标平台的变量。qt-cmake

如果您使用的是尚未在计算机上构建的Qt安装，则需要知道目标平台的CMake工具链文件的位置。qt-cmake

在这种情况下，您可以通过设置变量来指示链加载自定义工具链文件：qt-cmakeQT_CHAINLOAD_TOOLCHAIN_FILE

```
~/Qt/6.4.0/android_armv7/bin/qt-cmake -DQT_CHAINLOAD_TOOLCHAIN_FILE=<file-path> -S <source-dir> -B <build-dir>
```

这将指示Qt的内部工具链文件也加载您的自定义工具链文件。

< CMake 入门

导入的目标 >

©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/或其他国家/地区的[商标](#) 全球。所有其他商标均为其各自所有者的财产。



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

对于客户

- 支持中心
- 下载



客户成功案例

社区

- 为Qt做贡献
- 论坛
- 维基
- 下载
- 市场

© 2022 Qt公司

[反馈](#) [登录](#)