搜索

配置 qmake

性能

qmake 有一个持久配置系统，它允许您在 qmake 中设置一次属性，并在每次调用 qmake 时查询它。您可以在 qmake 中设置属性，如下所示：

```
qmake -set PROPERTY VALUE
```

Topics >

您可以从 qmake 检索此信息，如下所示：

```
qmake -query PROPERTY
qmake -query #queries all current PROPERTY/VALUE pairs
```

> **注：**除了设置的属性外，还列出了内置属性。qmake -queryqmake -set PROPERTY VALUE

此信息将被保存到QSettings对象中（这意味着它将存储在不同平台的不同位置）。

以下列表总结了属性：built-in

- QMAKE_SPEC - 主机的短名称在主机构建期间解析并存储在QMAKESPEC变量中mkspec
- QMAKE_VERSION - QMAKE 的当前版本
- QMAKE_XSPEC - 目标的短名称在目标构建期间解析并存储在QMAKESPEC变量中mkspec
- QT_HOST_BINS - 主机可执行文件的位置
- QT_HOST_DATA - qmake 使用的主机可执行文件的数据位置
- QT_HOST_LIBS - 主机库的位置
- QT_HOST_LIBEXECS - 运行时主机库所需的可执行文件的位置
- QT_HOST_PREFIX - 所有主机路径的默认前缀

- QT_INSTALL_CONFIGURATION - Qt设置的位置。不适用于视窗
- QT_INSTALL_DATA - 与架构无关的常规Qt数据的位置
- QT_INSTALL_DOCS - 文档位置
- QT_INSTALL_EXAMPLES - 示例的位置
- QT_INSTALL_HEADERS - 所有头文件的位置
- QT_INSTALL_LIBEXECS - 库在运行时所需的可执行文件的位置
- QT_INSTALL_LIBS - 图书馆的位置
- QT_INSTALL_PLUGINS - Qt插件的位置
- QT_INSTALL_PREFIX - 所有路径的默认前缀
- QT_INSTALL_QML - QML 2.x 扩展的位置
- QT_INSTALL_TESTS - Qt测试用例的位置
- QT_INSTALL_TRANSLATIONS - Qt 字符串的翻译信息的位置
- QT_SYSROOT - 目标构建环境使用的 sysroot
- QT_VERSION - Qt版本。我们建议您改用 $$QT.<module>.version 变量查询 Qt 模块特定的版本号。

例如，您可以使用以下属性查询此版本的qmake的Qt安装：QT_INSTALL_PREFIX

```
qmake -query "QT_INSTALL_PREFIX"
```

可以按如下方式查询项目文件中的属性值：

```
QMAKE_VERS = $$[QMAKE_VERSION]
```

QMAKESPEC

qmake 需要一个平台和编译器描述文件，其中包含许多用于生成适当 Makefile 的默认值。标准的Qt发行版附带了许多这些文件，位于Qt安装的子目录中。mkspecs

环境变量可以包含以下任何内容：QMAKESPEC

- 包含文件的目录的完整路径。在这种情况下，qmake 将从该目录中打开文件。如果该文件不存在，qmake 将退出并显示错误。qmake.confqmake.conf
- 平台编译器组合的名称。在这种情况下，qmake 将在编译 Qt 时指定的数据路径的子目录指定的目录中搜索（参见QLibraryInfo：:D ataPath）。mkspecs

> **注意：**路径将自动添加到生成的 Makefile 中，在INCLUDEPATH系统变量的内容之后。QMAKESPEC

缓存文件

如果 qmake 找到文件，那么它将在处理项目文件之前先处理此文件。 .qmake.cache

文件扩展名

在正常情况下，qmake会尝试为您的平台使用适当的文件扩展名。但是，有时需要覆盖每个平台的默认选项，并显式定义 qmake 要使用的文件扩展名。这是通过重新定义某些内置变量来实现的。例如，可以使用项目文件中的以下分配重新定义用于moc文件的扩展名：

```
QMAKE_EXT_MOC = .mymoc
```

以下变量可用于重新定义 qmake 识别的常见文件扩展名：

- QMAKE_EXT_MOC修改放置在包含的 MOC 文件上的扩展名。
- QMAKE_EXT_UI修改用于*Qt Designer*UI文件的扩展名（通常在FORMS中）。
- QMAKE_EXT_PRL修改放置在库依赖项文件上的扩展名。
- QMAKE_EXT_LEX更改 Lex 文件（通常在LEXSOURCES 中）中使用的后缀。
- QMAKE_EXT_YACC更改Yacc文件中使用的后缀（通常在YACCSOURCES中）。
- QMAKE_EXT_OBJ更改生成的对象文件上使用的后缀。

以上所有值仅接受第一个值，因此必须仅为其分配一个将在整个项目文件中使用的值。有两个变量接受值列表：

- QMAKE_EXT_CPP会导致 qmake 将所有带有这些后缀的文件解释为C++源文件。
- QMAKE_EXT_H会导致 qmake 将所有带有这些后缀的文件解释为 C 和 C++ 头文件。

‹ 使用预编译标头 参考 ›

职业
办公地点

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

社区

为Qt做贡献
论坛
维基
下载
市场

反馈 登录