

# 使用预编译头

预编译头（PCH）是某些编译器支持的一种性能功能，用于编译稳定的代码体，并将代码的编译状态存储在二进制文件中。在后续编译期间，编译器将加载存储状态，并继续编译指定的文件。每次后续编译都更快，因为稳定代码不需要重新编译。

qmake 支持在某些平台和构建环境中使用预编译头，包括：

- › 窗户
  - › 恩马克
  - › 可视化工作室项目（VS 2008 及更高版本）
- › 苹果操作系统、苹果系统、电视操作系统和手表操作系统
  - › 生成文件
  - › 断续器
- › 独角蛋白
  - › 海湾合作委员会3.4及更高版本
  - › 铛

## 将预编译头添加到项目中

预编译头必须包含在整个项目中 **稳定且静态** 的代码。典型的预编译头可能如下所示：

```
// Add C includes here

#if defined __cplusplus
// Add C++ includes here
#include <stdlib>
#include <iostream>
#include <vector>
#include <QApplication> // Qt includes
#include <QPushButton>
#include <QLabel>
#include "thirdparty/include/libmain.h"
#include "my_stable_class.h"
...
#endif
```

**注意：** 预编译头文件需要将 C 包含与包含 C++ 分开，因为 C 文件的预编译头文件可能不包含 C++ 代码。

## 项目选项

要使项目使用预编译头，只需在项目文件中定义 `PRECOMPILED_HEADER` 变量：

```
PRECOMPILED_HEADER = stable.h
```

qmake 将处理其余部分，以确保创建和使用预编译的头文件。您不需要在 中包含预编译的头文件，因为如果配置支持预编译头，qmake 将执行此操作。HEADERS

默认情况下，面向窗口的 MSVC 和 g++ 规范处于启用状态。precompile\_header

使用此选项，可以在使用预编译头时触发项目文件中的条件块以添加设置。例如：

```
precompile_header:!isEmpty(PRECOMPILED_HEADER) {  
  DEFINES += USING_PCH  
}
```

若要将预编译头也用于 MSVC nmake 目标上的 C 文件，请添加到 `配置` 变量。如果标头也用于 C++ 并且它包含 C++ 关键字/包含，请用 `)` 括起来。precompile\_header\_c#ifdef \_\_cplusplus

## 关于可能问题的说明

在某些平台上，预编译头文件的文件名后缀与其他对象文件的文件名后缀相同。例如，以下声明可能会导致生成两个具有相同名称的不同对象文件：

```
PRECOMPILED_HEADER = window.h  
SOURCES             = window.cpp
```

为避免此类潜在冲突，请为将要预编译的头文件指定独特的名称。

## Example Project

You can find the following source code in the directory in the Qt distribution: `examples/qmake/precompile`

### mydialog.ui

The following image displays the mydialog.ui file in Qt Creator Design mode. You can view the code in the Edit mode.



Grid Layout

Form Layout

Spacers

Horizontal Spacer

Vertical Spacer

Buttons

Push Button

Tool Button

Radio Button

Check Box

ComboBox

Button Box

Item View (List-Based)

List View

Tree View

Table View

Column View

Item Widget (Form-Based)

Filter

Filter

MyDialog : QDialog

Property	Value
QObject	
objectName	MyDialog
QWidget	
QDialog	
sizeGripEnabled	<input type="checkbox"/>
modal	<input type="checkbox"/>
Layout	
layoutName	
layoutLeftMargin	12
layoutTopMargin	12
layoutRightMargin	12
layoutBottomMargin	12
layoutSpacing	-1
layoutStretch	0,0
layoutSizeConstraint	SetDefaultConstraint

## stable.h

```
/* Add C includes here */

#ifdef __cplusplus
/* Add C++ includes here */

#include <iostream>
#include <QApplication>
#include <QPushButton>
#include <QLabel>
#endif
```

## myobject.h

```
#include <QObject>

class MyObject : public QObject
{
public:
    MyObject();
    ~MyObject();
};
```

## myobject.cpp

```
#include <iostream>
#include <QDebug>
#include <QObject>
#include "myobject.h"
```

```
{
    std::cout << "MyObject::MyObject()\n";
}
```

## util.cpp

```
void util_function_does_nothing()
{
    // Nothing here...
    int x = 0;
    ++x;
}
```

## main.cpp

```
#include <QApplication>
#include <QPushButton>
#include <QLabel>
#include "myobject.h"
#include "mydialog.h"

int main(int argc, char **argv)
{
    QApplication app(argc, argv);

    MyObject obj;
    MyDialog dialog;

    dialog.connect(dialog.aButton, SIGNAL(clicked()), SLOT(close()));
    dialog.show();

    return app.exec();
}
```

## precompile.pro

```
TEMPLATE = app
LANGUAGE = C++
CONFIG += cmdline precompile_header

# Use Precompiled headers (PCH)
PRECOMPILED_HEADER = stable.h

HEADERS = stable.h \
          mydialog.h \
          myobject.h
```

myobject.cpp \

util.cpp

FORMS = mydialog.ui

< Advanced Usage

Configuring qmake >

© 2022 The Qt Company Ltd. Documentation contributions included herein are the copyrights of their respective owners. The documentation provided herein is licensed under the terms of the [GNU Free Documentation License version 1.3](#) as published by the Free Software Foundation. Qt and respective logos are [trademarks](#) of The Qt Company Ltd. in Finland and/or other countries worldwide. All other trademarks are property of their respective owners.



Contact Us

Company

- About Us
- Investors
- Newsroom
- Careers
- Office Locations

Licensing

- Terms & Conditions
- Open Source
- FAQ

Support

- Support Services
- Professional Services
- Partners
- Training

For Customers

- Support Center
- Downloads
- Qt Login
- Contact Us
- Customer Success

Community

- Contribute to Qt
- Forum
- Wiki
- Downloads
- Marketplace

