

Recherches QMake

Qmake :

L'outil qmake vous fournit un système orienté projet pour gérer le processus de génération d'applications, de bibliothèques et d'autres composants. Cette approche vous donne le contrôle sur les fichiers source utilisés et permet de décrire de manière concise chacune des étapes du processus, généralement dans un seul fichier. qmake étend les informations de chaque fichier de projet à un Makefile qui exécute les commandes nécessaires à la compilation et à la liaison.

Fichiers .pro

Les projets sont décrits par le contenu des fichiers de projet (.pro). qmake utilise les informations contenues dans les fichiers pour générer des Makefiles contenant toutes les commandes nécessaires à la construction de chaque projet. Les fichiers de projet contiennent généralement une liste de fichiers source et d'en-tête, des informations de configuration générales et des détails spécifiques à l'application, tels qu'une liste de bibliothèques supplémentaires à relier, ou une liste de chemins à inclure supplémentaires à utiliser.

Variable	Contents
CONFIG	General project configuration options.
DESTDIR	The directory in which the executable or binary file will be placed.
FORMS	A list of UI files to be processed by the user interface compiler (uic).
HEADERS	A list of filenames of header (.h) files used when building the project.
QT	A list of Qt modules used in the project.
RESOURCES	A list of resource (.qrc) files to be included in the final project. See the The Qt Resource System for more information about these files.
SOURCES	A list of source code files to be used when building the project.
TEMPLATE	The template to use for the project. This determines whether the output of the build process will be an application, a library, or a plugin.

CONFIG

Specifies project configuration and compiler options. The values are recognized internally by qmake and have special meaning.

DESTDIR

Specifies where to put the [target](#) file.

FORMS

Specifies the UI files (see [Qt Designer Manual](#)) to be processed by uic before compiling. All dependencies, headers and source files required to build these UI files will automatically be added to the project.

HEADERS

Defines the header files for the project.

qmake automatically detects whether `moc` is required by the classes in the headers, and adds the appropriate dependencies and files to the project for generating and linking the moc files.

QT

Specifies the `Qt modules` that are used by your project. For the value to add for each module, see the module documentation.

At the C++ implementation level, using a Qt module makes its headers available for inclusion and causes it to be linked to the binary.

By default, QT contains `core` and `gui`, ensuring that standard GUI applications can be built without further configuration.

RESOURCES

Specifies the name of the resource collection files (`qrc`) for the target. For more information about the resource collection file, see [The Qt Resource System](#).

SOURCES

Specifies the names of all source files in the project.