

3

Installation de Qt et introduction à Qt Creator

Il existe plusieurs manières d'installer Qt : par des installateurs de binaires précompilés ou en compilant soi-même les sources, principalement. Enfin, les distributions Linux proposent généralement des paquets, comme pour bien d'autres bibliothèques, frameworks et logiciels. La méthode recommandée est de passer par un installateur, mais nous vous donnerons également un aperçu global des autres méthodes (pour ceux notamment qui ont des besoins spécifiques).

Après une comparaison des avantages et inconvénients de ces méthodes, nous aborderons l'utilisation de base de l'EDI Qt (environnement de développement intégré) de référence : [Qt Creator](#). Son utilisation n'est pas compliquée, mais cela pourra aider les plus débutants d'entre vous à démarrer.

Note > Des SDK spécialisés existent pour des utilisations plus spécifiques de Qt, comme Native SDK pour BlackBerry 10 ou V-Play, un framework de développement de jeux multiplateformes. Ils sont dévolus entièrement à ces usages, apportant des binaires pour les plateformes cibles et des outils plus pointus pour les besoins visés (des débogueurs ou des bibliothèques supplémentaires, par exemple). Cette section se concentre sur l'installation de Qt pour des plateformes desktop généralistes.

Également, Nokia proposait précédemment un SDK Qt à jour (avant le rachat de Qt par Digia), ressemblant très fort aux installateurs actuels mais ne supportant que Qt 4. Il est prévu tant pour une utilisation desktop que mobile (Symbian et MeeGo Harmattan).

Le remplaçant de ce SDK est l'installateur, qui permet depuis Qt 5.1 d'effectuer tant l'installation que la mise à jour des composants Qt.

3.1. Installation de Qt

Note > Qt ne supporte pas toutes les plateformes existantes. [La documentation](#) décrit la liste des configurations de référence, régulièrement testées, ainsi que la liste des plateformes testées moins régulièrement : tous les systèmes d'exploitation majeurs sont supportés, dans leurs versions relativement récentes (Windows depuis XP, OS X depuis la 10.6, Ubuntu depuis la 10.04, actuellement). D'autres plateformes fournissent elles-mêmes un support de Qt, comme FreeBSD.

Par l'installateur

Les installateurs fournis par le Qt Project représentent la manière la plus simple d'installer Qt, clé sur porte : le fichier téléchargé contient tous les fichiers nécessaires à l'utilisation de Qt (fichiers de bibliothèque dynamique, fichiers d'import, documentation, en-têtes, outils, etc.), rarement un compilateur (MinGW sous Windows).

Deux types d'installateurs sont disponibles : en ligne et hors ligne. Les seconds fournissent tout ce qui est nécessaire dans le fichier téléchargé depuis le site du Qt Project (pratique pour une installation sur un ordinateur qui n'a pas accès à Internet), tandis que les premiers téléchargent juste les parties nécessaires à l'exécution de l'installateur.

On les trouve généralement à deux endroits : soit sur [la page des téléchargements de Qt](#), soit depuis [la page des préversions](#).

Qt Project propose une série d'installateurs à destination des plateformes les plus courantes sur [la page des téléchargements de Qt](#). Vous en trouverez également pour certaines versions en cours de développement à la [la page des préversions](#).

ANGLE ou OpenGL sous Windows ?

Pour les installateurs sous Windows, en plus du compilateur, vous devrez choisir entre ANGLE ou OpenGL.

Pour fonctionner avec de bonnes performances, l'environnement Qt Quick 2 utilise beaucoup l'accélération matérielle du processeur graphique par le biais d'OpenGL. Il s'agit d'une bibliothèque standardisée pour l'utilisation du processeur graphique à des fins d'affichage à l'écran (*open graphics library*). Qt Quick 2 requiert OpenGL 3.0 ou OpenGL ES 2.0, soit la version pour ordinateurs de bureau et pour l'embarqué (*embedded systems*). Cette dernière API n'a pas toute la richesse fonctionnelle de sa grande sœur, à cause des environnements visés.

Sous Windows, seul OpenGL 1.0 est supporté de base : pour les versions ultérieures, il peut falloir installer des pilotes plus récents pour la ou les cartes graphiques. Même si des pilotes sont préinstallés sur l'ordinateur, la qualité du support OpenGL n'est pas garantie (voir, par exemple, [l'état des lieux dressé par les développeurs de Dolphin, un émulateur GameCube/Wii](#)). Par contre, DirectX, une API concurrente à OpenGL et développée par Microsoft, a un support en standard bien plus élevé. ANGLE (*almost native graphics layer engine*) est une implémentation d'OpenGL ES 2.0 sur DirectX.

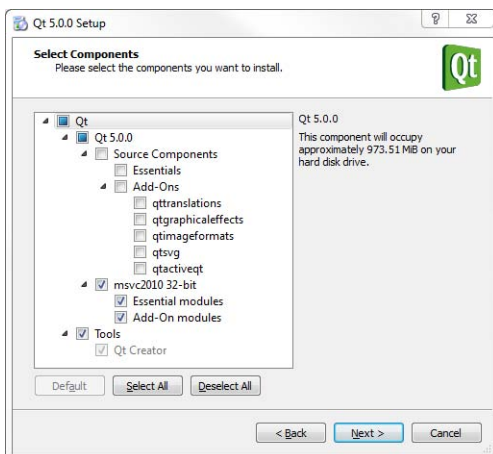
Le choix entre ANGLE et OpenGL peut se faire selon plusieurs critères :

- si l'application utilise des fonctionnalités non disponibles dans OpenGL ES 2.0, il faudra utiliser OpenGL, sinon ANGLE suffit ;
- s'il est important de garantir qu'une application utilisant OpenGL ou Qt Quick 2 fonctionne sur toutes les machines, sans dépendre des pilotes installés, ANGLE est préférable ; par contre, s'il faut viser Windows XP, il faut utiliser OpenGL, ANGLE n'étant pas supporté sur cette plateforme ;
- si ni Qt Quick 2, ni Qt OpenGL ne sont utilisés, la version OpenGL est alors préférable, puisqu'elle a moins d'implications sur le déploiement (voir [Section 6.3, Spécificités de Windows](#)).

L'installation est relativement simple : une fois l'installateur lancé, il suffit de suivre les instructions jusqu'au bout et Qt est installé. Pour utiliser les outils en ligne de commande sous Windows, une invite spécifique est disponible dans le menu DÉMARRER (les variables d'environnement nécessaires pour utiliser `qmake` et les autres outils en ligne de commande sont alors définies) ; sous OS X et Linux, les variables globales sont mises à jour.

Les options proposées pour l'installation sont assez limitées mais suffisent généralement à la majorité des besoins : il installe toujours Qt Creator ; il propose d'installer les modules essentiels et les add-ons de Qt compilés (uniquement pour le compilateur choisi lors du téléchargement) ; également, les sources sont proposées, tant pour les modules essentiels qu'add-ons. Les sources n'étant pas utiles à tous, la configuration par défaut (montrée à la [Figure 3.1](#)) devrait convenir.

Figure 3.1 : Options proposées par l'installateur



Attention > *Installation d'un compilateur : à aucun moment un compilateur n'est installé sur la machine, il faut donc en avoir un (le même que celui qui correspond au téléchargement), sauf sous Windows avec MinGW.*

Sous Windows, Visual C++ 10 (2010) et 11 (2012) sont les seules versions du compilateur commercial pour lesquelles des binaires précompilés sont disponibles. Il faut donc l'installer : l'édition Express (2010 – choisir l'édition C++ – ou 2012 – choisir l'édition Windows Desktop) suffit largement, pas besoin d'utiliser une des éditions payantes (sauf si vous avez besoin des fonctionnalités supplémentaires telles que l'analyse statique du code de l'édition 2010¹ ou les add-ins – sachant que, si vous êtes étudiant, ces éditions vous sont proposées gratuitement). Pour les autres versions, il faudra compiler Qt soi-même (Visual C++ 2008 et 2013 ne devraient pas poser de problème).

Depuis Qt 5.0.1, une version pour MinGW 4.7 (le port de GCC sous Windows) est disponible, c'est la seule à inclure un compilateur (elle est donc préférable pour les plus débutants). Qt 5.1 est depuis livré avec MinGW 4.8.

Sous OS X, il faut installer Xcode (disponible sur le DVD de l'OS jusqu'OS X Lion 10.7 ou sur l'App Store depuis lors). Selon la version d'OS X, Xcode vient avec GCC ou LLVM/Clang : Qt 5 supporte les deux (GCC sous OS X 10.6, Clang depuis lors).

Sous les autres UNIX, seul GCC est actuellement activement supporté. Il est toujours disponible dans le gestionnaire de paquets de la distribution utilisée (par exemple, sous Arch Linux, par `pacman -S gcc`).

Par les dépôts

Les systèmes de type UNIX disposent généralement de leur système de gestion de paquets : Yum pour Fedora, Synaptic pour Ubuntu et Mint, APT pour Debian, MacPorts pour OS X, etc. Comme la majorité des distributions propose une version de KDE, Qt est aussi disponible, bien que pas toujours dans ses dernières versions² ; leurs mainteneurs peuvent aussi décider d'intégrer certaines bibliothèques supplémentaires (Qwt, QxOrm, etc.) dans leurs dépôts.

Le principal avantage d'utiliser cette méthode est que l'on développe directement avec les mêmes bibliothèques que l'utilisateur final, ce qui peut résoudre quelques problèmes épineux au déploiement. Également, on évite d'utiliser un gestionnaire de paquets supplémentaire, ce qui garantit la cohérence du système (pas d'installation de Qt en double, par exemple).

La méthode d'installation varie grandement avec les systèmes ; en général, on utilise une commande comme `pacman -S qt5` sous Arch Linux. Il faut bien vérifier que les paquets de développement sont installés, car ils contiennent les fichiers requis pour développer avec Qt (les fichiers d'en-tête et d'import, en plus des binaires). Cependant, au vu de

¹Ces outils sont disponibles gratuitement avec l'édition Express 2012.

²Par exemple, Fedora propose Qt 4.8.4 – la version stable la plus récente de Qt 4 lors de la rédaction – et Qt Creator, alors que MacPorts a proposé très longtemps uniquement Qt 4.4.3 sans Qt Creator (ce qui est maintenant réglé).