

Introduction

1. Qt, pourquoi faire ?

Qt est un ensemble d'outils d'aide à la création de logiciels.

L'objectif de Qt est de faciliter le travail des développeurs de différentes manières :

- en étant multiplateforme : les applications créées avec Qt sont compatibles avec différents systèmes d'exploitation et le développeur peut se concentrer sur son applicatif plutôt que sur les détails spécifiques des plateformes visées ;
- en étant natif : les applications ne subissent pas de pénalité en terme de performances et d'intégration à la plateforme (ergonomie, design, accès au système) ;
- en étant complet : tous les ingrédients de base pour la réalisation d'applications sont intégrés, y compris des couches graphiques pour les interfaces utilisateurs.

En dehors de ces éléments quantitatifs, l'aspect qualitatif est également une composante très importante de Qt. Il s'agit du soin apporté aux API et à la documentation, et d'une réflexion permanente sur la "bonne façon" de faire les choses, en masquant autant que possible la complexité sous-jacente.

Le paysage applicatif est en évolution rapide depuis 2007, qui est l'année de la sortie du premier iPhone : les nouvelles plateformes mobiles (smartphones et tablettes) gagnent du terrain au détriment des plateformes bureautiques et des consoles de jeu ; un autre phénomène majeur est l'extension du domaine des applications web et des outils de développement associés. Sans être contradictoires, ces deux tendances ne sont pas convergentes : les besoins d'intégration au système, ainsi que les problèmes de continuité de la connexion internet rendent l'usage d'applications 100 % web difficile dans le contexte mobile.

Qt s'est adapté à ce paysage récent en proposant ses propres solutions aux nouveaux problèmes posés par les appareils mobiles et tactiles, non seulement en terme d'adaptation de l'interface utilisateur, mais aussi en terme d'adaptation aux cycles de développement de ces interfaces souvent beaucoup plus courts. Ces solutions sont basées essentiellement sur le module [Qt Quick](#), qui sera présenté en détail dans le livre.

2. Objectifs du livre

Ce livre a pour ambition de vous présenter les outils de base de Qt 5, en mettant l'accent sur ses nouveautés. Le framework et ses usages étant très variés, il nous a semblé plus pertinent de les aborder de manière thématique plutôt que de se confiner dans une structure d'apprentissage progressive classique. Par ailleurs, nous souhaitons aussi bien intéresser le lecteur débutant en Qt que le développeur familier de Qt 4 désireux de se former à Qt 5. Il en résulte un ouvrage un peu particulier, organisé sous forme de modules plus ou moins autonomes – un peu à l'image du framework.

Les essentiels constituent une première série de modules, visant à introduire le framework dans son ensemble et les fonctionnalités Qt 5 les plus attendues. Par la force des choses, la partie traitant de Qt Quick et du QML est la plus importante : c'est là que résident les principales nouveautés de Qt 5 et un débutant devra nécessairement commencer par s'initier à leur emploi.

Si vous êtes un adepte de Qt Widgets, vous serez certainement un peu surpris du peu de place que nous lui avons accordé. Non pas que nous jugeons ce module majeur de Qt comme secondaire, mais du fait de sa présence historique au sein du framework, il est déjà très documenté. Nous avons préféré nous concentrer sur des aspects moins connus.

Malgré ses 600 pages, le présent livre a fait l'objet de choix draconiens. Et nous avons le sentiment de n'avoir abordé qu'une petite partie de Qt. C'est pourquoi, nous prévoyons de publier à l'avenir d'autres modules spécialisés, qui viendront compléter cette première base.

3. Contenu du livre

Attention > *Ce livre n'est pas construit de manière linéaire. Si vous débutez, vous serez sans doute tenté de lire les chapitres dans l'ordre, or ce n'est pas forcément la meilleure façon ! Vous pourriez être un peu perdu. Commencez par les chapitres les plus débutants en vous aidant de la carte graphique de l'ouvrage et des parcours proposés à la Section 4, Par où commencer ?.*

Vous trouverez dans *Les essentiels* les modules suivants :

- **Tour d'horizon**, qui introduit les nouveautés de Qt 5 au regard des versions antérieures, présente la philosophie de son développement et ses perspectives ;
- **Migration de Qt 4 à Qt 5**, qui comme son nom l'indique vous aidera à migrer. Ces deux premiers chapitres, de par leur nature, s'adressent plutôt aux développeurs déjà familiers de Qt ;
- **Installation de Qt et introduction à Qt Creator**, qui fournit aux plus débutants d'entre vous des informations relatives à l'installation du SDK et un premier aperçu

- de Qt Creator, l'environnement de développement de référence. Vous y trouverez également des informations avancées pour des installations de Qt personnalisées ;
- [Premiers programmes \(avec Qt Widgets\)](#), qui passe en revue un ensemble de concepts importants de Qt 5 au travers de l'analyse de trois exemples. L'interface graphique de ces applications utilise le module Qt Widgets, ce qui peut constituer une introduction à ce module. Si vous avez déjà pratiqué Qt 4, vous y trouverez une illustration de fonctions nouvelles, ainsi qu'une petite introduction à Enginio, un service web disposant de fonctions d'intégration aux applications Qt ;
 - [Aller plus loin avec Qt Creator](#), qui présente différentes fonctions de Qt Creator fort utiles pour travailler efficacement et prendre en main l'environnement de développement dédié à Qt ;
 - [Qt Quick](#), qui vous initie en onze chapitres à l'utilisation de Qt Quick, le nouveau module de Qt permettant de développer facilement des interfaces graphiques en QML. Il se décompose en plusieurs chapitres allant du moins technique (à destination des graphistes et développeurs débutants) au plus technique (pour les développeurs C++).
 - [Graphics View](#), qui vous initie en trois chapitres (Les bases, notions avancées et exemple d'application) à l'emploi du framework QGraphicsView.

À ces modules s'ajoutent deux petits compléments : [Choix de l'interface graphique](#) qui vous aidera à choisir le module Qt dont vous avez besoin pour développer votre interface et [Déploiement d'applications](#) qui fournit quelques informations essentielles sur le déploiement d'une application avec Qt 5.

4. Par où commencer ?

Fondamentalement aucun ordre de lecture n'est imposé, vous pouvez lire les chapitres dans l'ordre que vous voulez, selon vos besoins et votre intérêt. Toutefois certains chapitres supposent que vous ayez acquis certaines connaissances de base, et d'autres s'adressent plus spécifiquement aux débutants. D'une manière générale, assurez-vous à l'aide de la carte graphique que le niveau du module qui vous intéresse corresponde bien à votre niveau.

Parcours Débutants infographistes

Commencez par le chapitre [Installation de Qt et introduction à Qt Creator](#) en y lisant que la section portant sur l'[installation de Qt par l'installateur](#) et l'[introduction à Qt Creator](#). Une fois Qt installé sur votre ordinateur, allez directement à la [Partie Qt Quick](#), et commencez par la première partie du chapitre [Créer des interfaces](#)

avec [Qt Quick Designer](#). Appropriiez-vous le mode Designer en créant vos propres exemples, et seulement après, si vous en sentez le besoin, essayez de comprendre ce qui se cache derrière vos objets : lisez [l'introduction](#) de la partie, puis le chapitre [Démarrer en QML](#). Progressez ensuite dans l'ordre des chapitres aussi loin que vous suivez.

Parcours Débutants développeurs

Commencez par le chapitre [Installation de Qt et introduction à Qt Creator](#), sans entrer dans les détails de la compilation, puis allez directement à la [Partie Qt Quick](#), lisez [l'introduction](#) puis passez directement à [Démarrer en QML](#). Progressez ensuite dans l'ordre des chapitres ou attaquez directement par le chapitre [Étude d'une première application avec Qt Quick](#) en vous référant aux chapitres plus théoriques au fur et à mesure de sa lecture pour approfondir votre connaissance des outils utilisés. Une fois les notions de base acquises, la découverte du [Designer](#) pourra vous faire gagner du temps. Après [Qt Quick](#), étendez votre connaissance de Qt avec les autres modules.

Parcours Développeurs débutant en Qt

Vous développez déjà en C++ et souhaitez prendre en main Qt, commencez par [Choix de l'interface graphique](#) et [Premiers programmes \(avec Qt Widgets\)](#), puis laissez-vous guider par votre curiosité et vos besoins. N'oubliez pas le chapitre [Aller plus loin avec Qt Creator](#), qui vous aidera à configurer votre environnement de développement selon vos habitudes.

Parcours Développeurs familiers de Qt 4

Commencez par le [Tour d'horizon](#) afin de repérer les nouveaux éléments de Qt 5 qui vous intéressent. Si vous devez faire migrer vos applications, poursuivez avec [Migration de Qt 4 à Qt 5](#). Puis poursuivez votre lecture selon vos centres d'intérêts. N'oubliez pas le chapitre [Aller plus loin avec Qt Creator](#), qui vous aidera à configurer votre environnement de développement selon vos habitudes. Ni le chapitre [Créer des interfaces avec Qt Quick Designer](#) qui apportera des pistes pour améliorer la collaboration entre designers et développeurs.

5. Prérequis

Le module Qt Quick ayant été conçu pour permettre à des non-développeurs de créer des interfaces en Qt, les premiers chapitres ayant trait à son emploi n'exigent aucun prérequis. C'est dans cette perspective qu'un chapitre spécifique introduisant JavaScript pour Qt Quick vous est proposé. Les autres parties du livre supposent que le lecteur possède des compétences de base en programmation C++ et en programmation orientée objet. Aucune connaissance préalable du framework Qt n'est nécessaire, cependant l'ouvrage est consacré à Qt 5 et ne couvre que partiellement les fondations de Qt 4.


Pour utiliser les exemples de ce livre, il vous faudra créer un environnement de développement en téléchargeant le framework Qt 5.x (de préférence Qt 5.1 ou plus), un environnement de développement intégré (EDI) et un compilateur C++. Le développement Qt est supporté sous Windows, Mac OS et Linux avec les composants cités ci-dessus. L'installateur Qt fournit Qt Creator pour chacune de ces plateformes comme EDI de base, la [Section 3.2, Introduction à Qt Creator](#) vous présente comment bien débuter avec cette application et le chapitre [Aller plus loin avec Qt Creator](#) vous permet d'aller plus loin. Il est également possible d'utiliser d'autres EDI comme Visual Studio sous Windows ou Xcode sous Mac OS.

Il vous faudra également un compilateur C++ pour compiler les exemples du livre et les applications que vous allez créer. Visual Studio Express 2010 et 2012 sont disponibles gratuitement pour Windows ou vous pouvez télécharger [MinGW](#) pour utiliser le compilateur GCC sur cette même plateforme. Les packages Linux et Mac OS proposent généralement une version récente de GCC. Depuis la version Qt 5.0.1, un installateur du framework pour Windows inclut les binaires de MinGW dans sa version 32 bits qui contient GCC 4.7.2. Qt 5.1 est depuis livré avec MinGW 4.8.

Si vous désirez faire du développement mobile, vous n'avez pas besoin d'un téléphone pour développer des applications Qt Quick. Des simulateurs pour les systèmes MeeGo et MeeGo sont disponibles avec Qt Simulator inclus dans Qt Creator. Le support de Android et iOS sont également disponibles en Technology Preview dans Qt 5.1. Il est cependant possible de brancher votre téléphone possédant un de ces systèmes d'exploitation avec un câble USB et de tester vos applications créées avec Qt Creator directement sur votre téléphone. Pour le développement mobile, des outils supplémentaires, comme un SDK spécifique à la plateforme, peuvent être nécessaires. Reportez-vous au chapitre [Installation de Qt et introduction à Qt Creator](#) pour plus d'informations.

6. Codes sources des exemples

Vous trouverez sur la [page](#) du site des éditions D-Booker consacrée au livre, onglet Compléments, une archive réunissant l'ensemble des codes sources des exemples du livre. Ceux-ci sont classés par projet et chapitre.

Toutefois pour vous simplifier l'accès à ces codes au fur et à mesure de votre lecture, nous les avons aussi hébergés sur Github dans un [dossier dédié](#). Chaque fois que vous rencontrerez l'icône  , un simple clic dessus vous renverra vers le dossier contenant le code complet de l'exemple.

7. Accès aux vidéos

La version numérique du livre contient quelques illustrations sous forme de vidéos .

- Si vous lisez ce livre en ligne, elles sont intégrées à votre page, et votre navigateur ira chercher de lui-même le format de vidéo qu'il supporte.
- Si vous lisez une version téléchargée, un clic sur l'image vous redigera vers la vidéo en ligne au format MP4. Si votre navigateur par défaut ne supporte pas nativement le MP4, modifiez à la main l'extension du fichier dans l'url en remplaçant `.mp4` par `.webm`.

Note > Avant de cliquer, assurez-vous que le pointeur de votre souris s'est changé en main.