



Community Experience Distilled

Application Development with Qt Creator

A fast-paced guide for building cross-platform applications using
Qt and Qt Quick

Ray Rischpater

[PACKT] open source*
PUBLISHING community experience distilled

Table of Contents

[Application Development with Qt Creator](#)

[Credits](#)

[About the Author](#)

[Acknowledgments](#)

[About the Reviewers](#)

[www.PacktPub.com](#)

[Support files, eBooks, discount offers and more](#)

[Why Subscribe?](#)

[Free Access for Packt account holders](#)

[Preface](#)

[What this book covers](#)

[What you need for this book](#)

[Who this book is for](#)

[Conventions](#)

[Reader feedback](#)

[Customer support](#)

[Downloading the example code](#)

[Errata](#)

[Piracy](#)

[Questions](#)

[1. Getting Started with Qt Creator](#)

[Downloading Qt Creator](#)

[Finding your way around Qt Creator](#)

[Your first application – Hello World](#)

[Hello World using the Qt GUI library](#)

[Hello World using Qt Quick](#)

[Summary](#)

[2. Building Applications with Qt Creator](#)

[Getting started – our sample library](#)

[Learning the landscape – the Build menu and .pro files](#)

[Linking against our sample library](#)

[Getting lost and found again – debugging](#)

[Setting breakpoints and stepping through your program](#)

[Fine-grained control of breakpoints](#)

[Examining variables and memory](#)

[Examining the call stack](#)

[The Projects pane and building your project](#)

[A review – running and debugging your application](#)

[Summary](#)

[3. Designing Your Application with Qt Designer](#)

[Code interlude – signals and slots](#)

[Creating forms in Qt Designer](#)

[Creating the main form](#)

[Using application resources](#)

[Instantiating forms, message boxes, and dialogs in your application](#)

[Wiring the Qt GUI application logic](#)

[Learning more about Qt GUI widgets](#)

[Code interlude – Qt Quick and QML syntax](#)

[Creating Qt Quick applications in Qt Designer](#)

[Creating a reusable button](#)

[The calculator's main view](#)

[Learning more about Qt Quick and QML](#)

[Summary](#)

[4. Localizing Your Application with Qt Linguist](#)

[Understanding the task of localization](#)

[Marking strings for localization](#)

[Localizing your application with Qt Linguist](#)

[Including localized strings in your application](#)

[Localizing special things – currencies and dates with QLocale](#)

[Summary](#)

[5. Performance Optimization with Qt Creator](#)

[The QML performance analyzer](#)

[QtSlowButton – a Qt Quick application in need of performance tuning](#)

[Finding memory leaks with Valgrind](#)

[QtLeakyButton – a Qt C++ application in need of memory help](#)

[Summary](#)

[6. Developing Mobile Applications with Qt Creator](#)

[A mobile software development primer](#)

[User attention is at a premium](#)

[Computational resources are at a premium](#)

[Network resources are at a premium](#)

[Storage resources are at a premium](#)

[To port or not to port?](#)

[A word on testing](#)

[Setting up Qt Creator for Android](#)

[Downloading all the pieces](#)

[Setting up the environment variables](#)

[Finishing the Android SDK installation](#)

[Configuring Qt Creator](#)

[Building and running your application](#)

[Summary](#)

[7. Qt Tips and Tricks](#)

[Writing console applications with Qt Creator](#)

[Integration with version control systems](#)

[Configuring coding style and coding format options](#)

[Building from the command line](#)

[Setting Qt Quick window display options](#)

[Learning more about Qt](#)

[Summary](#)

[Index](#)

Application Development with Qt Creator

Application Development with Qt Creator

Copyright © 2013 Packt Publishing

All rights reserved. No part of this book may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, without the prior written permission of the publisher, except in the case of brief quotations embedded in critical articles or reviews.

Every effort has been made in the preparation of this book to ensure the accuracy of the information presented. However, the information contained in this book is sold without warranty, either express or implied. Neither the author, nor Packt Publishing, and its dealers and distributors will be held liable for any damages caused or alleged to be caused directly or indirectly by this book.

Packt Publishing has endeavored to provide trademark information about all of the companies and products mentioned in this book by the appropriate use of capitals. However, Packt Publishing cannot guarantee the accuracy of this information.

First published: November 2013

Production Reference: 1131113

Published by Packt Publishing Ltd.

Livery Place

35 Livery Street

Birmingham B3 2PB, UK.

ISBN 978-1-78328-231-9

www.packtpub.com

Cover Image by Siddhart Ravishankar (<sidd.ravishankar@gmail.com>)

Credits

Author

Ray Rischpater

Reviewers

Lee Zhi Eng

Niels Holst

Kamakshi Subramaniam

Acquisition Editors

Vinay Argekar

Aarti Kumaraswamy

Commissioning Editor

Sruthi Kutty

Technical Editors

Hardik B. Soni

Krutika Parab

Manan Badani

Pankaj Kadam

Copy Editors

Sayanee Mukherjee

Laxmi Subramanian

Project Coordinator

Sageer Parkar

Proofreader

Linda Morris

Indexers

Mehreen Deshmukh

Tejal R. Soni

Graphics

Ronak Dhruv

Production Coordinator

Conidon Miranda

Cover Work

Conidon Miranda

About the Author

Ray Rischpater is an engineer and author with over 20 years' experience writing about and developing for computing platforms.

During this time, he has participated in the development of Internet technologies and custom applications for Java ME, Qualcomm BREW, Apple iPhone, Google Android, Palm OS, Newton, and Magic Cap, as well as several proprietary platforms. Presently, he's employed as a senior engineer at Microsoft in Mountain View, working on mapping and data visualization.

When not writing for or about mobile platforms, he enjoys hiking and photography with his family and friends in and around the San Lorenzo Valley in central California. When he's able, he also provides a public service through amateur radio as the licensed Amateur Extra station KF6GPE.

The books he's written so far include:

- *Microsoft Mapping: Geospatial Development with Bing Maps and C#* (with Carmen Au, Apress, 2013)
- *Beginning Nokia Apps Development* (with Daniel Zucker, Apress, 2010)
- *Beginning Java ME Platform* (Apress, 2008)
- *Wireless Web Development, Second Edition* (Apress, 2004)
- *eBay Application Development* (Apress, 2004)
- *Software Development for the QUALCOMM BREW Platform* (Apress, 2003)
- *Wireless Web Development, First Edition* (Apress, 2002)
- *Internet Appliances: A Wiley Tech Brief* (John Wiley & Sons, 2001)
- *Advanced Palm Programming* (with Steve Mann, John Wiley & Sons, 2000)
- *Palm Enterprise Applications: A Wiley Tech Brief* (John Wiley & Sons, 2000)

He holds a bachelor's degree in pure mathematics from the University of California, Santa Cruz and is a member of the IEEE, ACM, and ARRL.

Acknowledgments

First, I'd like to thank Sruthi Kutty for approaching me about the idea of writing an introductory book about Qt Creator. Second, I'd like to thank Sageer Parkar for shepherding the project throughout the process at Packt, making my first experience with Packt Publishing a painless one. I was fortunate to have several technical reviewers and editors on the project who gave their time graciously to improve the book. Finally, I'd like to thank my wife and son for their patience with me as I undertook yet another book.

About the Reviewers

Lee Zhi Eng is a 3D artist-turned-programmer who worked as a game artist and game programmer in several local game studios in his country, before becoming a contractor and a part time lecturer at a local university, teaching game development subjects, particularly related to Unity Engine and Unreal Development Kit. You can find more information about him at <http://www.zhieng.com>.

Niels Holst graduated from the University of Copenhagen, Denmark with a PhD in Biology. He currently works at Aarhus University, Denmark where he applies Computer Science to solve problems in Applied Ecology. He is a leader of the Universal Simulator open source project.

Support files, eBooks, discount offers and more

You might want to visit www.PacktPub.com for support files and downloads related to your book.

Did you know that Packt offers eBook versions of every book published, with PDF and ePub files available? You can upgrade to the eBook version at www.PacktPub.com and as a print book customer, you are entitled to a discount on the eBook copy. Get in touch with us at [<service@packtpub.com>](mailto:service@packtpub.com) for more details.

At www.PacktPub.com, you can also read a collection of free technical articles, sign up for a range of free newsletters and receive exclusive discounts and offers on Packt books and eBooks.



<http://PacktLib.PacktPub.com>

Do you need instant solutions to your IT questions? PacktLib is Packt's online digital book library. Here, you can access, read and search across Packt's entire library of books.

Why Subscribe?

- Fully searchable across every book published by Packt
- Copy and paste, print and bookmark content
- On demand and accessible via web browser

Free Access for Packt account holders

If you have an account with Packt at www.PacktPub.com, you can use this to access PacktLib today and view nine entirely free books. Simply use your login credentials for immediate access.

Preface

Whether you're just getting started with programming, or you've settled on Qt as the GUI toolkit for your project, Qt Creator is a great choice for an Integrated Development Environment (IDE)! In this book, we work to help you make the most of Qt Creator, showing you almost every facet of using Qt Creator, from its configuration through compiling and debugging applications, along with numerous tips and tricks. Along the way, you gain valuable experience not just with Qt Creator as an IDE, but with Qt and Qt Quick as well. After reading this book, you'll be able to:

- Edit, compile, debug, and run C++ applications using Qt Creator, opening a path to build state-of-the-art console and GUI applications with Qt and the Standard Template Library (STL)
- Edit, compile, debug, and run Qt Quick applications using Qt Creator, giving you access to one of the most advanced declarative GUI authoring environments anywhere
- Design GUI applications using Qt Designer to build either traditional widget-based or Qt Quick applications
- Analyze the memory and runtime performance of your Qt applications, and make improvements, and fix defects
- Provide localized versions of your application, so that you can deploy it all over the world in different languages
- Use Qt Quick and Qt Widgets to write mobile applications for platforms such as Google Android

What this book covers

This book is divided into seven chapters, which you should plan on reading in order, especially if you're new to Qt Creator and Qt programming in general. These chapters are:

[Chapter 1](#), *Getting Started with Qt Creator*, explains how to download and install Qt Creator, as well as edit simple applications to test your installation.

[Chapter 2](#), *Building Applications with Qt Creator*, explains how to compile, run, and debug your application using Qt Creator. You will learn how Qt Creator integrates with both the GNU debugger and the Microsoft console debugger to provide breakpoints, memory inspection, and other debugging help.

[Chapter 3](#), *Designing Your Application with Qt Designer*, explains how to use the drag-and-drop GUI designer that is part of Qt Creator, to build both Qt widget-based and Qt Quick applications.

[Chapter 4](#), *Localizing Your Application with Qt Linguist*, explains how to manage resource strings for different locales, letting you build your application with different languages in different locales.

[Chapter 5](#), *Performance Optimization with Qt Creator*, explains how to use Qt Creator to examine your Qt Quick application's runtime performance, as well as how to perform memory profiling of your application with Valgrind, an open source diagnostic tool.

[Chapter 6](#), *Developing Mobile Applications with Qt Creator*, gives a look at the exciting arena of mobile software development, and shows how you can use what you've learned in this book about Qt and Qt Creator to write applications for platforms such as Google Android.

[Chapter 7](#), *Qt Tips and Tricks*, covers tricks for using Qt and Qt Creator that will help you use the Qt framework and the Qt Creator IDE efficiently.

What you need for this book

Qt and Qt Creator are cross-platform tools. Whether you're using a Windows machine, a Macintosh using Mac OS X, or a workstation running Linux, you probably have what you need. You should have a reasonable amount of disk space (around 10 gigabytes is plenty) to install the whole Qt Creator IDE and Qt libraries, and as with any software development environment, the more RAM you have, the better (although I've run Qt Creator on netbooks running Ubuntu with a gigabyte of RAM and survived!).

You should have a basic understanding of computer programming, and should be prepared to write code in C++. Basic knowledge of JavaScript is helpful if you're interested in programming with Qt Quick, but you can pick that up along the way with little difficulty.

Who this book is for

I wrote this book for those who have little or no experience with Qt and Qt Creator, who may be using it for the first time as part of a college class, an open source project, or who just want to experiment with the platform and IDE.

I especially want to encourage you to read this book if you're a student using Qt Creator in your university class on C++ programming! You should focus on the first two chapters, and a much of the rest as you need for your course.

Conventions

In this book, you will find a number of styles of text that distinguish between different kinds of information. Here are some examples of these styles, and an explanation of their meaning.

Code words in text, database table names, folder names, filenames, file extensions, pathnames, dummy URLs, user input, and Twitter handles are shown as follows: "For the name, enter `HelloWorldConsole`, and choose a path that makes sense for you (or accept the default)."

A block of code is set as follows:

```
#include <QCoreApplication>
#include <iostream>
using namespace std;
int main(int argc, char *argv[])
{
    QCoreApplication a(argc, argv);
    cout << "Hello world!";
    return a.exec();
}
```

When we wish to draw your attention to a particular part of a code block, the relevant lines or items are set in bold:

```
import QtQuick 2.0
Rectangle {
    width: 360
    height: 360
    Text {
        text: qsTr("Hello World")
        anchors.centerIn: parent
    }
    MouseArea {
        anchors.fill: parent
        onClicked: {
            Qt.quit();
        }
    }
}
```

New terms and **important words** are shown in bold. Words that you see on the screen, in menus or dialog boxes for example, appear in the text like this: "Where it says **Type Here**, right-click and choose **Remove menu bar**."

Note

Warnings or important notes appear in a box like this.

Tip

Tips and tricks appear like this.

Reader feedback

Feedback from our readers is always welcome. Let us know what you think about this book—what you liked or may have disliked. Reader feedback is important for us to develop titles that you really get the most out of.

To send us general feedback, simply send an e-mail to <feedback@packtpub.com>, and mention the book title via the subject of your message.

If there is a topic that you have expertise in and you are interested in either writing or contributing to a book, see our author guide on <www.packtpub.com/authors>.

Customer support

Now that you are the proud owner of a Packt book, we have a number of things to help you get the most from your purchase.

Downloading the example code

You can download the example code files for all Packt books you have purchased from your account at <http://www.packtpub.com>. If you purchased this book elsewhere, you can visit <http://www.packtpub.com/support> and register to have the files e-mailed directly to you.

Errata

Although we have taken every care to ensure the accuracy of our content, mistakes do happen. If you find a mistake in one of our books—maybe a mistake in the text or the code—we would be grateful if you would report this to us. By doing so, you can save other readers from frustration and help us improve subsequent versions of this book. If you find any errata please report them by visiting <http://www.packtpub.com/submit-errata>, selecting your book, clicking on the **errata submission form** link, and entering the details of your errata. Once your errata are verified, your submission will be accepted and the errata will be uploaded on our website, or added to any list of existing errata, under the Errata section of that title. Any existing errata can be viewed by selecting your title from <http://www.packtpub.com/support>.

Piracy

Piracy of copyright material on the Internet is an ongoing problem across all media. At Packt we take the protection of our copyright and licenses very seriously. If you come across any illegal copies of our works, in any form, on the Internet, please provide us with the location address or website name immediately so that we can pursue a remedy.

Please contact us at [<copyright@packtpub.com>](mailto:copyright@packtpub.com) with a link to the suspected pirated material.

We appreciate your help in protecting our authors, and our ability to bring you valuable content.

Questions

You can contact us at [<questions@packtpub.com>](mailto:questions@packtpub.com) if you are having a problem with any aspect of the book, and we will do our best to address it.

Chapter 1. Getting Started with Qt Creator

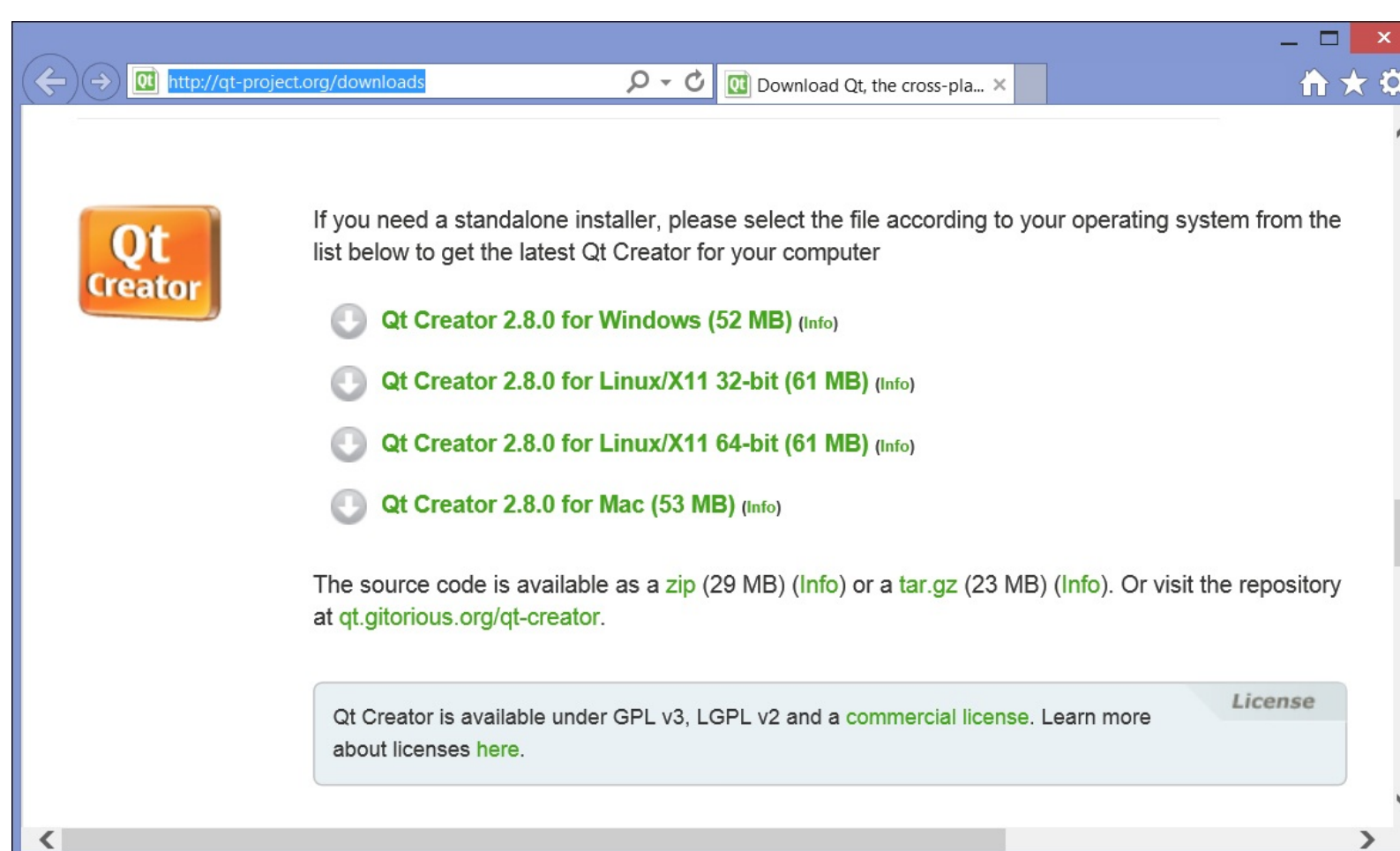
Qt Creator is the integrated software development environment that supports both traditional C++ application development, as well as development using the Qt project's libraries (collectively called "Qt", pronounced "cute"). In this chapter, we will see everything we need to get started with Qt Creator:

- Where to download Qt Creator for Linux, Mac OS X, or Windows
- How to ensure that your basic configuration is running
- A quick look at a simple Qt GUI application, as well as a Qt Quick application

Downloading Qt Creator

Qt, the cross-platform toolkit behind Qt Creator, has had a long and illustrious history. Presently, a project of Digia, it has its own URL at qt-project.org and has both commercial and noncommercial licenses available.

To get started with the noncommercial version for free, head over to <http://bit.ly/13G4Jfr> to see something similar to the following screenshot:



The screenshot shows a web browser window displaying the Qt Creator download page. The address bar shows the URL <http://qt-project.org/downloads>. The page features the Qt Creator logo on the left and a list of download links on the right. The list includes: Qt Creator 2.8.0 for Windows (52 MB), Qt Creator 2.8.0 for Linux/X11 32-bit (61 MB), Qt Creator 2.8.0 for Linux/X11 64-bit (61 MB), and Qt Creator 2.8.0 for Mac (53 MB). Below the list, it states that the source code is available as a zip (29 MB) or a tar.gz (23 MB), and provides a link to the repository at qt.gitorious.org/qt-creator. A license section at the bottom indicates that Qt Creator is available under GPL v3, LGPL v2, and a commercial license, with a link to learn more about licenses.

Downloading Qt Creator

Tip

One of the most popular platforms for application development with Qt is Linux. On many Linux variants—notably Ubuntu, my personal favorite—you can get Qt Creator using the package manager. On my Ubuntu box, Qt Creator is just a `sudo apt-get install qtcreator` command away. You'll get a version of Qt matched with your flavor of Linux, although it might not be the latest and greatest build from Digia.

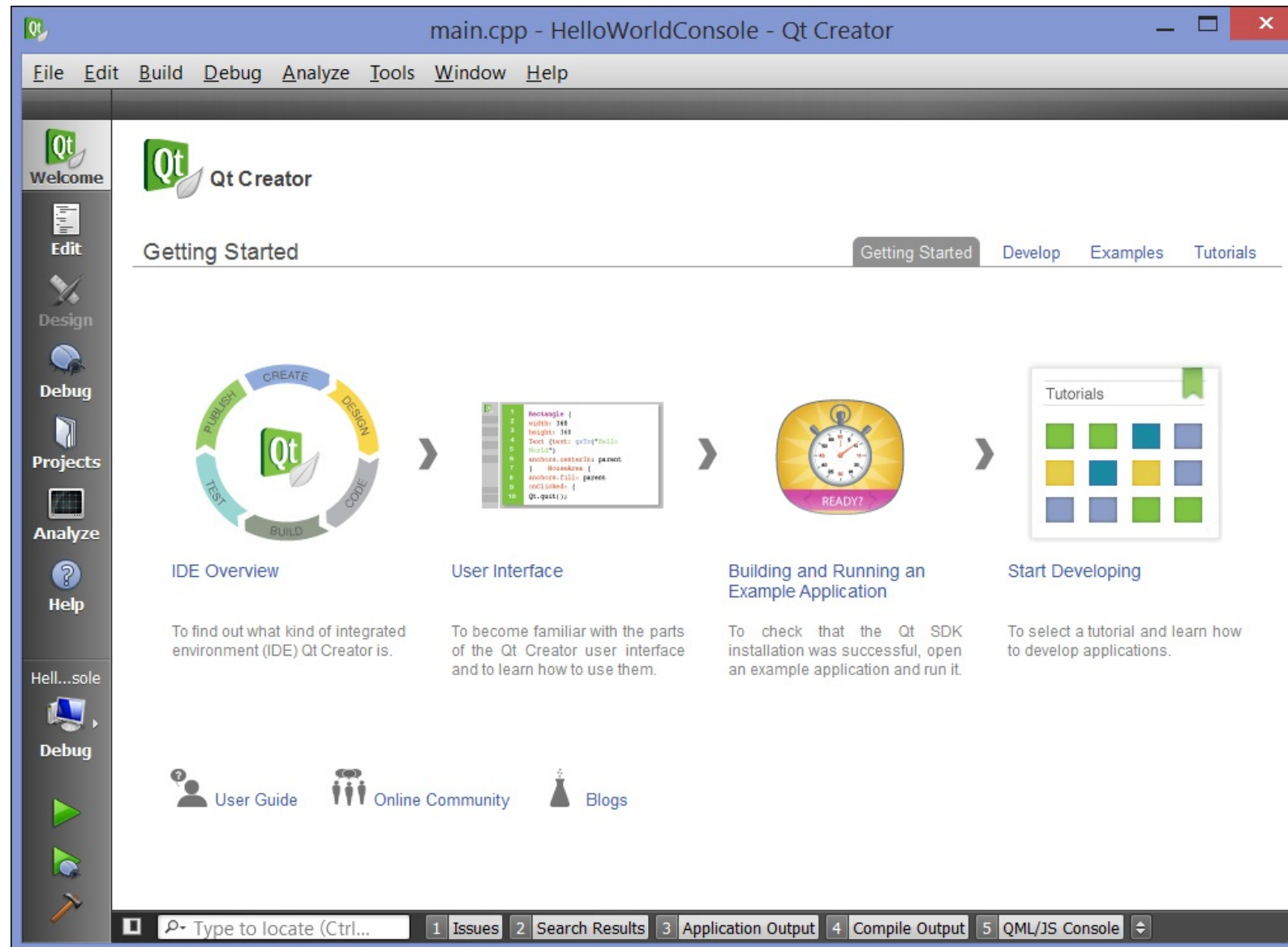
We can also download bits and pieces of Qt, such as just the runtime libraries, or build Qt Creator from source. This typically requires that you already have a compiler and basic development tools installed, and a basic understanding of `qmake` and Qt's build configuration management system.

Some downloads include the C++ compiler and linker you need for your development; others don't. For example, on Windows there's a variant that includes the MinGW tool chain, so you have everything you need to build applications. However, you can also download Qt Creator for Windows that uses the Microsoft Visual Studio compilers, so, if you prefer using Visual Studio for your compilation and Qt Creator as your IDE, that's also an option. On Mac OS X, you'll need to have Xcode and the command-line development tools installed first; you can download Xcode from the Mac OS X App Store, and then use Xcode to download the command-line development tools.

Once the installer is downloaded, run it in the usual way. It'll launch an installation wizard for your platform, and typically the installation takes about three or four minutes. You'll want to have plenty of disk space. Qt Creator doesn't consume that much disk space, but software development typically does; figure at least 500 megabytes for the tools and libraries, and budget a few gigabytes free on your main drive for your source code, intermediate object files, debug symbols, and of course, your compiled application. (This is especially important to plan for if you're running Qt Creator on a virtual machine; make sure that the virtual hard drive for your virtual machine image has plenty of disk space.) You should also ensure that your development box has plenty of RAM; the more, the better. Qt Creator runs happily in 2 GB of RAM, but the compiler and linker used by Qt Creator can run a lot faster if it has more RAM available.

Finding your way around Qt Creator

The following screenshot shows what you see the first time you launch Qt Creator. Let's take a closer look at each portion of the screen:



The landing page of Qt Creator

The main window, which currently shows the icons for **IDE Overview**, **User Interface**, **Building and Running an Example Application**, and **Start Developing**, is your workspace. Under normal conditions, this will be where you'll see the source code for your application. Along the left-hand side are a series of icons that let you select various views into your application. They are:

- The **Welcome** view shows basic information about Qt Creator
- The **Edit** view lets you edit the files that make up your application
- The **Design** view lets you use the Qt Designer to design the user interface for your application
- The **Debug** view lets you debug your application while it's running, including doing things like viewing memory and variables, setting breakpoints, and stepping through your

application

- The **Projects** view lets you adjust the build and link settings for your project
- The **Analyze** view lets you profile your application's runtime performance
- The **Help** view provides documentation about Qt Creator and the Qt Framework

Below the **Help** view button in the previous screenshot you can see the active project; when I took this screenshot, I had already created our first application. Let's do that now.

- **[download SMART Goals Made Simple: 10 Steps to Master Your Personal and Career Goals pdf, azw \(kindle\), epub, doc, mobi](#)**
- [download online Pirate Latitudes: A Novel pdf](#)
- [Mikhail Kuzmin: A Life in Art online](#)
- [Don't Think of an Elephant! Know Your Values and Frame the Debate for free](#)
- **[download Beginning Linux Programming \(4th Edition\) online](#)**
- **[download online Salvador Dal  \(Critical Lives\)](#)**

- <http://xn--d1aboelcb1f.xn--p1ai/lib/Real-World-Digital-Photography--3rd-Edition-.pdf>
- <http://patrickvincitore.com/?ebooks/Pirate-Latitudes--A-Novel.pdf>
- <http://musor.ruspb.info/?library/A-Thousand-Acres--A-Novel.pdf>
- <http://academialanguagebar.com/?ebooks/Don-t-Think-of-an-Elephant--Know-Your-Values-and-Frame-the-Debate.pdf>
- <http://www.celebritychat.in/?ebooks/Beginning-Linux-Programming--4th-Edition-.pdf>
- <http://betsy.wesleychapelcomputerrepair.com/library/I-Live-in-Your-Basement---Goosebumps--Book-61-.pdf>