

Preface

Over the past five years, we at Klarälvdalens Datakonsult AB¹ have presented almost one hundred weeklong Qt courses, which has brought us enormous insight into the kinds of problems people experience in the real world. We also have lots of practical experience ourselves from programming millions of lines of code in Qt consultancy projects and from spending our spare time programming in the KDE project.²

During our courses, we have often answered questions you would not find in a normal Qt programming book, questions such as, “How do I change the label of the OK button in a `QFileDialog`?”, “How do I make certain table items read-only?”, or “How do I get a `QSpinBox` to show floating-point numbers?”. Sometimes we were not even capable of answering the question off the top of our heads, but had to sit down and carefully read the reference documentation (or the source code for Qt), and, of course, apply our knowledge about how Qt normally works.

This book will answer a lot of common questions and some *odd* ones that bring an interesting approach to solving problems in Qt. Once you get started reading the book, you will learn to think in nontraditional ways, ways that will help you solve problems that might previously have been very hard for you to solve.

The layout of this book ensures that you can extend your Qt knowledge during a lengthy compilation. You may also use the book as a dictionary to common problems many people face with Qt. You may read a chapter each evening before going to sleep, but be prepared to get up and rush to the computer to try out your new knowledge with a problem you might have been struggling with for a long time.

No matter, how you choose to use this book, we wish you a lot of joy, and hope to get some feedback from you at our email address: practicalqt@klaralvdalens-datakonsult.se

Who Should Read This Book?

This book is for those who do daily programming in the trenches, so basic Qt programming knowledge is assumed. If you feel unsure of the basics, we (not quite unbiasedly) recommend *Programming with Qt* by Matthias Kalle Dalheimer, published by O'Reilly & Associates.

1 <http://www.kdab.net>

2 K Desktop Environment-<http://www.kde.org>

However, you do not have to be an experienced Qt programmer to benefit from this book. This book is designed to help you become one or, if you are already quite experienced, to become even more productive.

Programmer productivity is one of the main themes of this book, which is set up to help you find answers to common and not-so-common problems quickly and easily. If you save a few hours here and there by following the tips, you will have more than saved the purchase price of the book, and we will have fulfilled our goal.

Another benefit you should get from this book is increased programmer creativity. Many of the tips presented here are not difficult, but they might require some creativity to find the right solution. Hopefully, reading this book will make you not only more productive but also more creative and ultimately let you have even more fun programming.

Downloading the Examples

There is an old saying that goes like this: *Checking that a crossword is solved correctly is easy; it's much harder to fill it in.* In the same way, reading about programming will only get you so far: You need to play with the examples, try to change them, and see the effects of doing so. For that reason, we have made the examples from the book available on the web. Go to <http://www.kdab.net>, choose Publications in the side bar, and then choose Examples from Practical Qt.

All the code examples are copyrighted by Klarälvdalens Datakonsult AB, the authors' employer, and you can use them freely in your own (commercial or non-commercial) software projects. You may not use the examples, however, for teaching or publishing without further written authorization.

Acknowledgements

First of all, we would like to thank all our co-workers who have contributed ideas, discussed solutions to problems, and done technical reviews of the chapters. A special acknowledgement goes to Steffen Hansen, who wrote the chapter on shared libraries, and to Karl-Heinz Zimmer, who wrote the chapter on matching printer and screen coordinate systems. David Faure also needs a special thank-you up front, because he, more than anyone else, has taken the challenge to discuss solutions to hard-to-solve problems for this book. Tobias Larsson has helped preparing the example code and the screenshots. Steffen, Karl-Heinz, David, Tobias, and our other colleagues, Marc Mutz, Tanja Dalheimer, Michel Boyer de la Giroday, Bo Thorsen, Lutz Rogowski, and Daniel Molkenntin, have provided a tremendous working environment in which Qt ideas flow at an amazing speed.

We would also like to thank the people in our Qt classes, who have made our life much harder by asking questions we never ever had thought about; without those questions this book would never have taken off.

All the people who continuously answer questions on the qt-interest and #qt IRC channel also deserve a thank-you. Many sections of this book started from questions in the above forums, and sometimes we were lucky to get ideas from the answers.

A big thank-you also goes to our editor René Schönfeldt, who pushed us toward the deadline. To Christoph Kaeder from the Lehmanns bookstore, who got us in contact with René. To Manfred Krüger, who processed our home-grown bunch of XML files into something actually printable and type-set the book. To Cyndi Marsico, who skillfully transformed our crimes against the English language into something that hopefully even native English speakers can read without fainting. And to all the staff at dpunkt, who were involved in the production of the book.

Custom tells us to send thank-you's to our significant others, children, other relatives, friends, pets, and so on, and Who are we to break with tradition? So here we go: This book was to a large extent written during working hours, so that in itself did not keep us away from our loved ones, but work in general takes up many hours, for sure. As the workaholics that we clearly are, we just have to continue remembering that no one on his or her deathbed has ever wished to have spent more time at work. We promise to try to spend more time with all of you in the future.