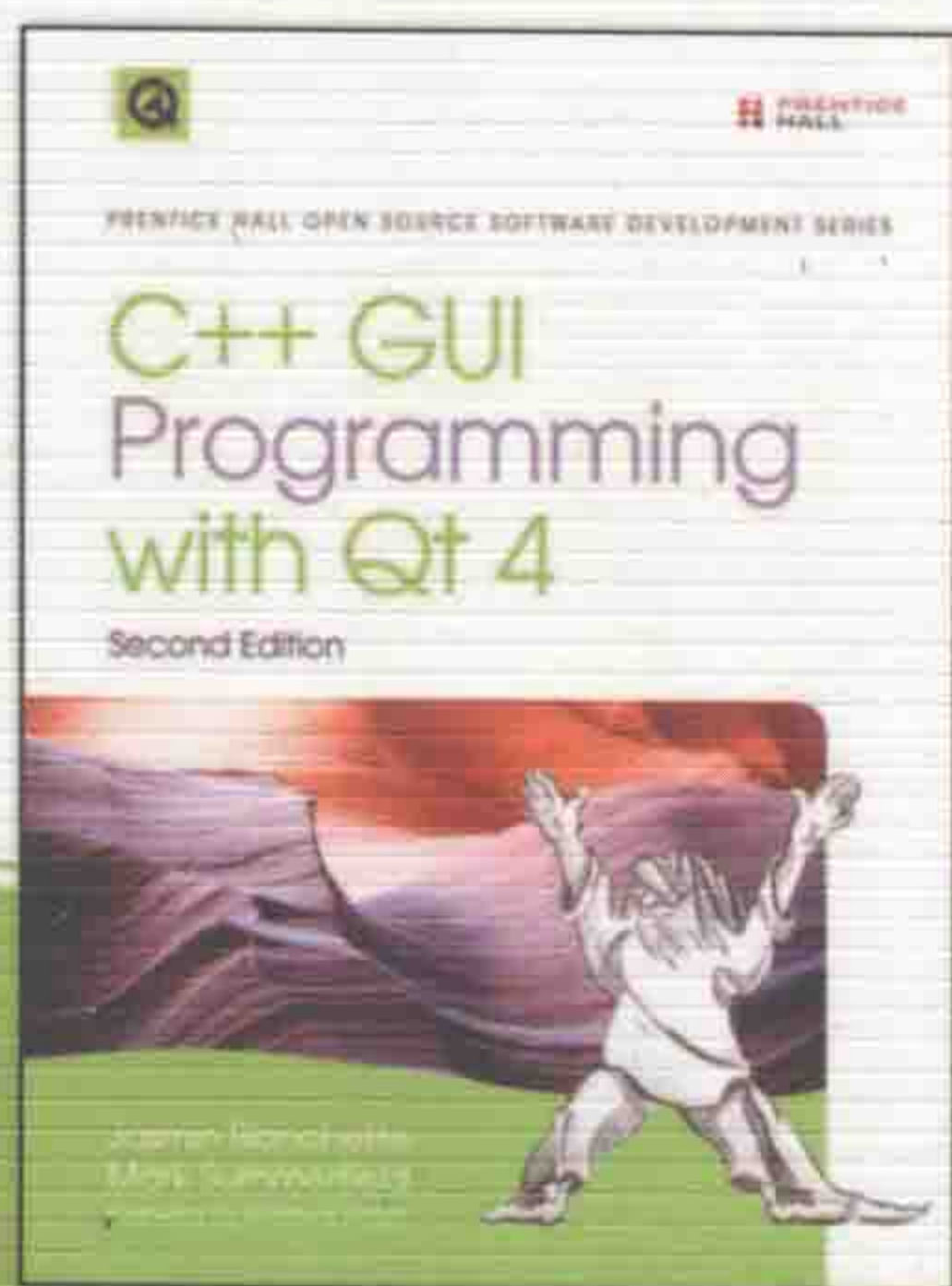


C++ GUI Qt 4 编程 (第二版)

C++ GUI Programming with Qt 4, Second Edition



[加] Jasmin Blanchette

著

[英] Mark Summerfield

闫锋欣 曾泉人 张志强 等译

周莉娜 赵延兵 审校

C++ GUI Qt 4 编程

(第二版)

C++ GUI Programming with Qt 4
Second Edition

[加] Jasmin Blanchette

[英] Mark Summerfield

著

闫锋欣 曾泉人 张志强 等译

周莉娜 赵延兵 审校

電子工業出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

本书详细讲述了使用更新的 Qt 版本进行图形用户界面应用程序开发的各个方面。全书分为四个部分,共 24 章及 4 个附录:第一部分介绍 Qt 的基础知识和编写图形用户界面应用程序时所需的基本概念;第二和第三部分主要讲解 Qt 的中级、高级编程,包括布局管理、事件处理、二维/三维图形、拖放、项视图类、容器类、输入与输出、数据库、多线程、网络、XML、国际化、嵌入式编程等内容;第四部分为附录,分别介绍了 Qt 的获取和安装、Qt 应用程序的构建、Qt Jambi 和 C++ 的基础知识。对于本书讲授的大量 Qt 4 编程原理和实践,都可以轻易将其应用于 Qt 4、Qt 5 以及后续版本的 Qt 程序的开发过程中。

本书适合对 Qt 编程感兴趣的程序员及广大计算机编程爱好者阅读,也可作为相关机构的培训教材。

Authorized translation from the English language edition, entitled C++ GUI Programming with Qt 4, Second Edition, 9780132354165 by Jasmin Blanchette and Mark Summerfield, published by Pearson Education, Inc., publishing as Prentice Hall, Copyright © 2008 Trolltech ASA.

All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage retrieval system, without permission from Pearson Education, Inc.

CHINESE SIMPLIFIED language edition published by PEARSON EDUCATION ASIA LTD., and PUBLISHING HOUSE OF ELECTRONICS INDUSTRY Copyright © 2018.

本书中文简体字版专有出版权由 Pearson Education(培生教育出版集团)授予电子工业出版社。未经出版者预先书面许可,不得以任何方式复制或抄袭本书的任何部分。

本书贴有 Pearson Education(培生教育出版集团)激光防伪标签,无标签者不得销售。

版权贸易合同登记号 图字:01-2008-2499

图书在版编目(CIP)数据

C++ GUI Qt 4 编程:第二版/(加)杰思敏·布兰切特(Jasmin Blanchette),(英)马克·萨默菲尔德(Mark Summerfield)著;闫锋欣等译. —北京:电子工业出版社,2018.5

书名原文:C++ GUI Programming with Qt 4, Second Edition

ISBN 978-7-121-34162-5

I. ①C… II. ①杰… ②马… ③闫… III. ①软件工具-程序设计 IV. ①TP311.56

中国版本图书馆 CIP 数据核字(2018)第 088234 号

策划编辑:冯小贝

责任编辑:冯小贝

印 刷:辽宁泰阳广告彩色印刷有限公司

装 订:辽宁泰阳广告彩色印刷有限公司

出版发行:电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本:787×1092 1/16 印张:31.75 字数:875 千字

版 次:2018 年 5 月第 1 版(原著第 2 版)

印 次:2018 年 5 月第 1 次印刷

定 价:89.00 元

凡所购买电子工业出版社图书有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系,联系及邮购电话:(010)88254888, 88258888。

质量投诉请发邮件至 zlts@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

本书咨询联系方式: fengxiaobei@phei.com.cn。

译者序

“The future is written with Qt.”

Qt 在近 30 年的时间里,实现了从小小接口库到业界主流跨平台开发框架的华丽转变,已成为众多 C/C++ 跨平台应用程序开发人员的首要选择。跨平台、时效快、易应用、开放扩展等优点,使得 Qt 能够为烦琐的多平台应用程序开发提供一站式解决方案,赢得了家用产品、汽车行业、消费电子、航空航天、军工民用等众多世界 500 强企业的信赖,更为这些企业的开发人员搭建起了完善的无缝开发平台。

“这的确是一本好书”,Prentice Hall 开源软件开发系列丛书主编之一的 Arnold Robbins 评价道。该书在讲述 Qt 程序开发中的很多关键技术时,不仅详细介绍了基本概念和底层背景知识,还对编程实践中的技巧和理论做了充分说明,为开发其他程序提供了宝贵经验。同时,脉络清晰的主线、循序渐进的内容体系,使得不同技术层次的读者都可以从中寻求到自己的解决方案。此外,该书一直作为 Qt 公司新员工的培训教材,更足以说明其经典性。

近年来,从桌面程序转向移动应用是软件开发行业面临的重要转变之一,开发人员也从未像今天这般需要跨平台技术。Qt“一次编码,随处编译”的跨平台开发框架,全面地解决了 Windows/Linux 等桌面平台、Android/iOS 等移动计算平台、Raspberry/WinCE 等嵌入式架构和 Web 环境的一体化开发需求。然而,随着 The Qt Company 对 Qt 技术的推陈出新,该书首版中的技术缺陷被放大,不少随书源码都已无法直接运行于 Qt 5.x 环境,极有必要依据 Qt 5 技术的发展更新书中的相应内容。

十余位 Qt 译者历时半年有余,在闫锋欣、曾泉人和张志强三人完成的翻译版的基础上,以随书源码全面升级兼容 Qt 5.x 最新技术为目标,完成了:(1)测试更新了全书的所有链接,是读者学习 Qt 技术的强力支撑;(2)基于 Qt 5.10 及 QtCreator 4.5.0 (Community),调试完成了随书源码,方便读者练习使用;(3)添加了大量译者注,介绍从 Qt 4.x 到 5.x 的技术发展;(4)整合了已知的内容勘误;(5)重新译校了部分内容。

本次翻译修订工作的具体分工是:西北农林科技大学的闫锋欣负责第 1~4 章,陕西师范大学的孟永宏负责第 5、17 章和附录 A~D、前言、致谢等部分;江苏金智科技股份有限公司的宋康负责第 6 章;徐景亮负责第 7、23、24 章;李湛负责第 8、20、21 章;王德刚负责第 9、14 章;陶兴龙、马俊阳、王祥凯分别负责第 10、11、12 章;李平负责第 13、15 章;赵益欢负责第 16、18、22 章;张涛负责第 19 章。此外,参与本书文字校对工作的还有西北农林科技大学的杜烁炜、苗港、安长青和西安交通大学的尹帅。全书最后由闫锋欣统稿。

本书的中文站点是 <http://www.qtcn.org/gpq4/>。在这里,读者可以下载随书示例程序源码或与专业人士探讨 Qt 的相关技术问题,也可留下您的宝贵建议助力后续修订工作。

感谢中央高校基本科研业务费项目(Z109021423)和农业部现代农业装备重点实验室开放课题(201603002)的资助支持,让译者能够全心投入译审校的繁重工作中。

由于译者水平有限,加之时间仓促,译稿中难免有曲解或误解作者原意的地方,望您谅解并批评指正。

闫锋欣

2018 年 2 月 5 日

丛书编者序

亲爱的读者：

作为一名职业程序员，我每天都在使用 Qt，而且我对 Qt 的组织结构、设计及其为 C++ 程序员所带来的动力印象深刻。

自 Qt 作为跨平台的图形用户界面工具包诞生以来，它已经扩展到了包括便携式设备在内的几乎当今程序设计的所有领域，比如文件、进程、网络和数据库的存取访问等。由于 Qt 的广泛适应性及良好的可移植性，只要您编写过一次代码，那么在其他不同的操作系统平台中，只需重新将其编译一遍即可重复使用。特别是当客户要求您的应用程序能同时运行于不同的平台时，Qt 的这一优势就显得格外重要了。

当然，使用开源许可协议也可以获得 Qt。如果您是一名开源程序开发人员，那么，从 Qt 那里也将获益无穷。

尽管 Qt 提供了大量的在线帮助文档，但这些帮助文档以参考性内容为主。示例程序非常有用，但仅仅通过阅读这些示例程序就希望能够在自己的程序中正确使用 Qt，显然是一件非常困难的事情。而这一点正是本书引人入胜的地方所在。

这的确是一本好书。首先，这是一本由奇趣科技公司推出的关于 Qt 的官方书籍，这说起来可能有些多余了。同时，它也是一部巨著：组织巧妙、文笔优雅，并且易于根据该书进行学习。与阐述伟大技术的巨著相结合，更易于造就真正的成功者，而这也就是为什么让我感到非常自豪和非常兴奋并乐于将此书作为 Prentice Hall 开源软件开发系列丛书之一的原因。

我希望您能享受到阅读这本书并从中获益良多的那种乐趣，而我，的确已经体会到了这一点。

Arnold Robbins

Nof Ayalon, 以色列

序 言

为什么会是 Qt? 为什么像我这样的程序员会选择 Qt? 这个问题的答案显而易见: Qt 单一程序的兼容性、丰富的特性、C++ 方面的性能、源代码的可用性、它的文档、高质量的技术支持, 以及在奇趣科技公司那些精美的营销材料中所涉及的其他优势等。这些答案看起来确实都不错, 但是遗漏了最为重要的一点: Qt 的成功缘于程序员们对它的喜爱。

那么, 是什么让程序员喜欢某种技术而放弃另外一种呢? 就我而言, 我认为软件工程师们喜欢某种技术, 是因为他们觉得这种技术是合适的, 但是这也会让他们讨厌所有那些他们觉得不合适的其他技术。除此之外, 我们还能解释下面的这些情况吗? 例如, 一些最出众的程序员需要在其他人的帮助之下才能编写出一个复杂的程序, 或者又比如, 似乎大多数工程师在操作本公司的电话系统时总会遇到麻烦。我虽然善于记住随机数字和指令的序列, 但是如果将其比作用于控制我的应答系统所需要的条件来说, 则可能一条也不具备。在奇趣科技公司, 我们的电话系统要求在拨打其他人的分机号码前, 一定要按住“*”键 2 秒后才允许开始拨号。如果忘记了这样做而是直接拨打分机号码, 那么就不得不再重新拨一遍全部的号码。为什么是“*”键而不是“#”键、“1”键或者“5”键? 或者为什么不是 20 个电话键盘中的其他任何一个呢? 又为什么是 2 秒, 而不是 1 秒、3 秒或者 1.5 秒呢? 问题到底出在哪里? 我发现打电话很烦人, 所以我尽可能不去使用它。没有人喜欢总是去做一些不得不做的随机事情, 特别是当这些随机事情显然只出现在同样随机的情况下的时候, 真希望自己从来都没有听到过它。

编程很像我们正在使用的电话系统, 并且比它还糟糕。而这正是 Qt 所要解决的问题。Qt 与众不同。一方面, Qt 很有意义; 另一方面, Qt 颇具趣味性。Qt 可以让您把精力集中在您的任务上。当 Qt 的首席体系结构设计师面对一个问题的时候, 他们不是寻求一个好的、快速的或者最简便的解决方案, 而是寻求一个恰当的解决方案, 然后将其记录在案。应当承认, 他们犯下了一些错误, 并且还要承认的是, 他们的一些设计决策没有通过时间的检验, 但是他们确实做出了很多正确的设计, 并且那些错误的设计应当而且也是能够改正的。看一看最初设计用于构建 Windows 95 和 UNIX Motif 之间的桥梁系统, 到后来演变为跨越 Windows Vista、Mac OS X 和 GNU/Linux 以及那些诸如移动电话等小型设备在内的统一的现代桌面系统, 这些事实就足以证明这一点。

早在 Qt 大受欢迎并且被广泛使用以前, 正是 Qt 的开发人员为寻求恰当的解决方案所做出的贡献才使 Qt 变得与众不同。其贡献之大, 至今仍然影响着每一个对 Qt 进行开发和维护的人。对我们而言, 研发 Qt 是一种使命和殊荣。能够使您的职业生涯和开源生活变得更为轻松和更加有趣, 这让我们倍感自豪。

人们乐于使用 Qt 的诸多原因之一是它的在线帮助文档, 但是该帮助文档的主要目的是集中介绍个别的类, 而很少讲述应当如何构建现实世界中那些复杂的应用程序。这本好书填补了这一缺憾, 它展示了 Qt 所提供的东西, 如何使用“Qt 的方式”进行 Qt 编程, 以及如何充分地利用 Qt。本书将指导 C++、Java 或者 C# 程序员进行 Qt 编程, 并且提供了丰富的资料来使他们成长为老练的 Qt 程序员。本书包含了很多很好的例子、建议和说明——并且, 本书也是我们对那些新加入公司的程序员们进行培训的入门教材。

如今,已有大量的商业或者免费的 Qt 应用程序可以购买或者下载,其中的一些专门用于特殊的高端市场,其他则面向大众市场。看到如此多的应用程序都是基于 Qt 构建而成的,这使我们充满了自豪感,并且还激励我们要让 Qt 变得更好。相信在这本书的帮助下,将会前所未有地出现更多的、质量更高的 Qt 应用程序。

Matthias Ettrich

柏林,德国

前言

Qt 使用“一次编写,随处编译”的方式为开发跨平台的图形用户界面应用程序提供了一个完整的 C++ 应用程序开发框架。Qt 允许程序开发人员使用应用程序的单一源码树来构建可以运行在不同平台下的应用程序的不同版本,这些平台包括从 Windows 98 到 Vista、Mac OS X、Linux、Solaris、HP-UX 以及其他很多基于 X11 的 UNIX。许多 Qt 库和工具也都是 Qt/Embedded Linux 的组成部分。Qt/Embedded Linux 是一个可以在嵌入式 Linux 上提供窗口系统的产品。

本书的目标就是教您如何使用 Qt 4 来编写图形用户界面程序。本书从“Hello Qt”开始,然后很快地转移到更高级的话题中,如自定义窗口部件的创建和拖放功能的提供等。通过本书的网站 (<http://www.informit.com/title/0132354160>),您可以下载到一些作为本书文字补充材料的示例程序。附录 A 说明了如何下载和安装这些软件,其中包括一个用于 Windows 的 C++ 免费编译器。

本书分为四部分。第一部分涵盖了在使用 Qt 编写图形用户界面应用程序时所必需的全部基本概念和练习。仅掌握这一部分中所蕴含的知识就足以写出实用的图形用户界面应用程序。第二部分进一步深入介绍了 Qt 的一些重要主题,第三部分则提供了更为专业和高级的内容。读者可以按任意顺序阅读第二部分和第三部分中的章节,但这是建立在对第一部分中的内容非常熟悉的基础之上的。第四部分包括数个附录,附录 B 说明了如何构建 Qt 应用程序,附录 C 则介绍了 Qt Jambi,它是 Java 版的 Qt。

本书的第一版建立在 Qt 3 版本的基础上,尽管已通过全书修订来反映那些很好的 Qt 4 编程技术,但本书还是根据 Qt 4 的模型/视图结构、新的插件框架、使用 Qt/Embedded Linux 进行嵌入式编程等内容而引入了一些新的章节和新的附录。作为第二版,本书充分利用了 Qt 4.2 和 Qt 4.3 中引入的新特性对其进行了彻底更新,并包含“自定义外观”和“应用程序脚本”两个新的章以及两个新的附录。原有的“图形”一章已经拆分为“二维图形”和“三维图形”两章,在它们中间,涵盖了新的图形视图类和 QPainter 的 OpenGL 后端实现。此外,在数据库、XML 和嵌入式编程等几章中,还添加了许多新内容。

与本书的前一版一样,这一版的重点放在如何进行 Qt 编程的说明和许多真实例子的提供上,而不是对丰富的 Qt 在线文档的简单拼凑和总结。因为本书纯粹讲授的是 Qt 4 编程中的原理和实践知识,因而读者能够轻松学会将要出现在 Qt 4.4、Qt 4.5 以及 Qt 4.x 等后续版本中的 15 个 Qt 新模块。如果您正在使用的 Qt 版本恰好是这些后续版本中的一个,那么当然要阅读一下参考文档中的“*What's New in Qt 4.x*”一章,以便可以对那些可用的新特性有一个总体把握。

在写作本书的时候,假定读者已经具备了 C++、Java 或者 C# 的基本知识。本书中的例子代码使用的是 C++ 中的一个子集,从而避免了很多在 Qt 编程中极少使用的 C++ 特性。在某些不可避免而必须使用 C++ 高级结构的地方,会在使用时对其做出必要的解释。如果您对 Java 或者 C# 已经非常熟悉但是对 C++ 还知之不多甚至一无所知,那么建议您先阅读附录 D。附录 D 提供了对 C++ 较为充分的介绍,从而能够让您掌握使用本书所必备的 C++ 知识。对于 C++ 中的面向对象编程更为全面的介绍,建议您阅读由 P. J. Deitel 和 H. M. Deitel 撰写的 *C++ How to Program*,以及由 Stanley B. Lippman、Josée Lajoie 和 Barbara E. Moo 撰写的 *C++ Primer* 这两本书。

Qt 以其作为一个跨平台框架而著称,但由于 Qt 拥有直观、强大的 API(应用程序编程接口),很多公司更愿意把 Qt 用于单一平台的软件开发上。Adobe PhotoShop Album 就是用 Qt 编写的面

向大众市场的 Windows 应用程序中的一个例子。纵向市场中很多功能完善的软件系统,如三维动画工具、数字电影处理软件、自动化电路设计系统(用于芯片设计)、油气资源勘探、金融服务以及医学成像等,都可以基于 Qt 构建而成。如果您正是一名通过 Qt 成功编写 Windows 产品来谋求发展的人,那么只需通过重新编译您的产品,就可以轻松地在 Mac OS X 和 Linux 世界中开拓出新的市场。

可以基于多种许可协议获得 Qt 的使用权。如果想构建商业应用程序,那么必须从奇趣科技公司购买一个 Qt 的商业许可协议。但如果只想构建一些开源程序,那么就可以使用基于 GPL 的 Qt 开源版本。KDE 和多数开源应用程序都是基于这种模式并使用 Qt 构建起来的。

除了 Qt 的数百个类之外,还有很多扩展 Qt 应用范围和功能的软件。其中的一些产品,像 Qt Solutions 中的一些组件,都可以从该公司获得。同时,还有其他很多软件由另外一些公司或者开源社区提供。对于可用的 Qt 额外软件的清单,可以查阅 <https://www1.qt.io/partners/>。奇趣科技公司的开发人员也有他们自己的网站,这就是 Trolltech 实验室(Trolltech Labs,现已改版为 Qtblog, <https://blog.qt.io>),他们会把自己编写的一些用于娱乐方面的、有趣的或者是有用的非官方代码放在那里。Qt 还建立了一个维护良好并且内容丰富的用户社区,用户可以通过 Qt 兴趣邮件列表来进行交流,详细情况请参阅 lists.qt-project.org/mailman/listinfo。

如果您在本书中发现了任何错误、对下一版有任何建议或者想反馈意见,我们将非常高兴收到您的邮件,邮件请发送到 qt-book@trolltech.com^①。

^① 由于该邮件地址已经失效,建议将意见反馈到本书中文网站 www.qtcn.org/gpq4/。

致 谢

首先要感谢 Eirik Chambe-Eng,他是奇趣科技公司的总裁,也是该公司两位创始人之一。Eirik 不仅热情地鼓励我们编写了本书的第一版(基于 Qt 3 版本),他还允许我们在写作本书时占用大量的工作时间。Eirik 和公司的 CEO——Haavard Nord 都阅读了本书的初稿并给出了许多宝贵的意见。他们的慷慨和远见来自于 Matthias Ettrich,他是该公司的首席程序员,他欣然同意我们把编著本书放在工作的首位,并且给我们提出了很多关于 Qt 编程的良好风格习惯的建议。

在本书的 Qt 3 版本中,我们邀请了 Qt 的两位客户:Paul Curtis 和 Klaus Schmidinger,由他们作为我们的外部审稿人。他们都是 Qt 专家,非常关注技术细节,他们在初稿中发现了一些小的错误,并且提出了许多改进的建议。此外,在公司中,除了 Matthias 以外,Reginald Stadlbauer 也是我们最忠实的审稿人,他的技术洞察力是无价的,他还教会了我们如何在 Qt 中做一些我们甚至认为是不可能的事情。

对基于 Qt 4 的本书第二版,我们继续从 Eirik、Haavard 和 Matthias 他们那里得到了无私的帮助和大力的支持。Klaus Schmidinger 继续给出了颇有价值的反馈意见,并且我们也继续从 Qt 的客户 Paul Floyd 对于一些新材料内容的详细审阅中获益。还要感谢 David García Garzón 在附录 B 中关于 SCons 的内容的帮助。在公司里,其他的主要审稿人还有 Carlos Manuel Duclos Vergara、Andreas Aardal Hanssen、Henrik Hartz、Martin Jones、Vivi Glückstad Karlsen、Trond Kjernåsen、Trenton Schulz、Andy Shaw、Gunnar Sletta 和 Pål de Vibe。

除了以上提到的几位审稿人,我们还得到了一些专家的帮助,他们分别是:Eskil Abrahamsen Blomfeldt(Qt Jambi)、Frans Englich(XML)、Harald Fernengel(数据库)、Kent Hansen(应用程序脚本)、Volker Hilsheimer(ActiveX)、Bradley Hughes(多线程)、Lars Knoll(二维图形和国际化)、Anders Larsen(数据库)、Sam Magnuson(qmake)、Marius Bugge Monsen(项视图类)、Dimitri Papadopoulos(Qt/X11)、Girish Ramakrishnan(样式表)、Samuel Rødal(三维图形)、Rainer Schmid(网络和 XML)、Amrit Pal Singh(C++ 简介)、Paul Olav Tvette(自定义窗口部件和嵌入式编程)、Geir Vattekar(Qt Jambi)和 Thomas Zander(编译系统)。

写作本书占用了我们大量的时间。同时,还要对公司的文档管理人员以及与文档处理相关的技术支持团队表示感谢,并且也要对公司的系统管理员们表示感谢,他们让我们的机器始终稳定运行并可以随时与整个项目的网络保持联系。

在本书的制作方面,感谢 Lout 排版工具的作者 Jeff Kingston,他不断地增强了该工具的许多功能,并且对我们的许多问题给予了及时的回复。也要感谢 James Cloos 所提供的 DejaVu Mono 字体的压缩版,该字体是我们使用的 monospaced 字体的基础。感谢奇趣科技公司的 Cathrine Bore 代表我们处理了合同和法律上的事务。还要感谢 Nathan Clement 所做的系列插图,以及 Audrey Doyle 对本书所做的详细校对。最后,感谢我们的编辑 Debra Williams-Cauley,既要感谢她的支持,又要感谢她为我们提供的轻松自在的写作过程。还要感谢 Lara Wysong 编辑,它使得本书这么实用。

Qt 简史

Qt 框架首度为公众可用是在 1995 年 5 月。它最初由 Haavard Nord(奇趣科技公司的 CEO)和 Eirik Chambe-Eng(公司总裁)开发而成。Haavard 和 Eirik 在位于挪威特隆赫姆的挪威科技学院相识,在那里,他们都获得了计算机科学的硕士学位。

Haavard 对 C++ 图形用户界面开发的兴趣始于 1988 年,当时一家瑞典公司委托他开发一套 C++ 图形用户界面框架。几年后,在 1990 年的夏天,Haavard 和 Eirik 因为一个超声波图像方面的 C++ 数据库应用程序而在一起工作。这个系统需要一个能够在 UNIX、Macintosh 和 Windows 上都能运行的图形用户界面。在那个夏天中的某天,Haavard 和 Eirik 一起出去散步,享受阳光,当他们坐在公园的一条长椅上时,Haavard 说:“我们需要一个面向对象的显示系统。”由此引发的讨论,为他们即将创建的面向对象的、跨平台的图形用户界面框架奠定了技术基础。

1991 年,Haavard 和 Eirik 开始一起合作设计、编写最终成为 Qt 的那些类。在随后的一年中,Eirik 提出了“信号和槽”的设想——一个简单并且有效的强大的图形用户界面编程规范,而现在,它已经可以被多个工具包实现。Haavard 实践了这一想法,并且据此创建了一个手写代码的实现系统。到 1993 年,Haavard 和 Eirik 已经开发出了 Qt 的第一套图形内核程序,并且能够利用它实现他们自己的一些窗口部件。同年末,为了创建“世界上最好的 C++ 图形用户界面框架”,Haavard 提议一起进军商业领域。

1994 年成为两位年轻程序员不幸的一年,他们没有客户,没有资金,只有一个未完成的产品,但是他们希望能够闯进一个稳定的市场。幸运的是,他们的妻子都有工作并且愿意为他们的丈夫提供支持。在这两年里,Haavard 和 Eirik 认为,他们需要继续开发产品并且从中赚得收益。

之所以选择字母“Q”作为类的前缀,是因为该字母在 Haavard 的 Emacs 字体中看起来非常漂亮。随后添加的字母“t”代表“工具包”(toolkit),这是从“Xt”——一个 X 工具包的命名方式中获得的灵感。公司于 1994 年 3 月 4 日成立,最初的名字是“Quasar Technologies”,随后更名为“Troll Tech”,而公司今天的名字则是“Trolltech”。

1995 年 4 月,通过 Haavard 就读过的大学的一位教授的联系,挪威的 Metis 公司与他们签订了一份基于 Qt 进行软件开发的合同。大约在同一时间,公司雇用了 Arnt Gulbrandsen,在公司工作的 6 年时间里,他设计并实现了一套独具特色的文档系统,并且对 Qt 的代码也做出了不少贡献。

1995 年 5 月 20 日,Qt 0.90 被上传到 sunsite.unc.edu。6 天后,在 comp.os.linux.announce 上发布。这是 Qt 的第一个公开发行版本。Qt 既可以用于 Windows 上的程序开发,又可以用于 UNIX 上的程序开发,而且在这两种平台上,都提供了相同的应用程序编程接口。从第一天起,Qt 就提供了两个版本的软件许可协议:一个是进行商业开发所需的商业许可协议版,另一个则是适用于开源开发的自由软件许可协议版。Metis 的合同确保了公司的发展,然而,在随后长达 10 个月的时间内,再没有任何人购买 Qt 的商业许可协议。

1996 年 3 月,欧洲航天局(European Space Agency)购买了 10 份 Qt 的商业许可协议,它成为第二位 Qt 客户。凭着坚定的信念,Eirik 和 Haavard 又雇用了另外一名开发人员。Qt 0.97 在同年的 5 月底正式发布,随后在 1996 年 9 月 24 日,Qt 1.0 正式面世。到了这一年的年底,Qt 的版本已经发展到了 1.1,共有来自 8 个不同国家的客户购买了 18 份 Qt 的商业许可协议。也就是在这一年,在 Matthias Ettrich 的带领下,创立了 KDE 项目。

Qt 1.2 于 1997 年 4 月发布。Matthias Ettrich 利用 Qt 建立 KDE 的决定,使 Qt 成为 Linux 环境下开发 C++ 图形用户界面的事实标准。Qt 1.3 于 1997 年 9 月发布。

Matthias 在 1998 年加入公司,并且在当年 9 月发布了 Qt 1 系列的最后一个版本——V 1.40。1999 年 6 月,Qt 2.0 发布,该版本拥有一个新的开源许可协议——Q 公共许可协议(QPL, Q Public License),它与开源的定义一致。1999 年 8 月,Qt 赢得了 LinuxWorld 的最佳库/工具奖。大约在这个时候,Trolltech Pty Ltd. (澳大利亚)成立了。

2000 年,公司发布了 Qt/Embedded Linux,它用于 Linux 嵌入式设备。Qt/Embedded Linux 提供了自己的窗口系统,并且可以作为 X11 的轻量级替代产品。现在,Qt/X11 和 Qt/Embedded Linux 除了提供商业许可协议之外,还提供了广为使用的 GNU 通用公共许可协议(GPL, General Public License)。2000 年底,成立了 Trolltech Inc. (美国),并发布了 Qtopia 的第一版,它是一个用于移动电话和掌上电脑(PDA)的环境平台。Qt/Embedded Linux 在 2001 年和 2002 年两次获得了 LinuxWorld 的“Best Embedded Linux Solution”奖,Qtopia Phone 也在 2004 年获得了同样的荣誉。

2001 年,Qt 3.0 发布。现在,Qt 已经可用于 Windows、Mac OS X、UNIX 和 Linux(桌面和嵌入式)平台。Qt 3 提供了 42 个新类和超过 500 000 行代码。Qt 3 是自 Qt 2 以来前进历程中最为重要的一步,它主要在诸多方面进行了众多改良,包括本地化和统一字符编码标准的支持、全新的文本查看和编辑窗口部件,以及一个类似于 Perl 正则表达式的类等。2002 年,Qt 3 赢得了 Software Development Times 的“Jolt Productivity Award”^①。

2005 年夏,Qt 4.0 发布,它大约有 500 个类和 9000 多个函数,Qt 4 比以往的任何一个版本都要全面和丰富,并且它已经裂变成多个函数库,从而使开发人员可以根据自己的需要只连接所需要的 Qt 部分。相比于以前的所有 Qt 版本,Qt 4 的进步是巨大的,它不仅彻底地对高效易用的模板容器、高级的模型/视图功能、快速而灵活的二维绘图框架和强大的统一字符编码标准的文本查看和编辑类进行了大量改进,更不必说对那些贯穿整个 Qt 类中的成千上万个小的改良了。现如今,Qt 4 具有如此广泛的特性,以至于 Qt 已经超越了作为图形用户界面工具包的界限,逐渐成长为一个成熟的应用程序开发框架。Qt 4 也是第一个能够在其所有可支持的平台上既可用于商业开发又可用于开源开发的 Qt 版本。

同样在 2005 年,公司在北京开设了一家办事处,以便为中国及其销售区域内的用户提供服务 and 培训,并且为 Qt/Embedded Linux 和 Qtopia 提供技术支持。

通过获取一些非官方的语言绑定件(language bindings),非 C++ 程序员也已早就开始使用 Qt,特别是用于 Python 程序员的 PyQt 语言绑定件。2007 年,公司发布了用于 C#程序员的非官方语言绑定件 Qyoto。同一年,Qt Jambi 投放市场,它是一个官方支持的 Java 版 Qt 应用程序编程接口。附录 C 提供了对 Qt Jambi 的介绍。

自奇趣科技公司诞生以来,Qt 的声望经久不衰,而且至今依旧持续高涨。取得这样的成绩不但说明了 Qt 的质量可信,而且也说明了人们都喜欢使用它。在过去的 10 年中,Qt 已经从一个只被少数专业人士所熟悉的“秘密”产品,发展了到如今遍及全世界拥有数以千计的客户和数以万计的开源开发人员的产品。

^① Jolt 奖素有“软件业界的奥斯卡”之美誉,共设通用类图书、技术类图书、语言和开发环境、框架库和组件、开发者网站等十余个分类,每个分类设有一个“震撼奖”(Jolt Award)和三个“生产力奖”(Productivity Award)。一项技术产品只有在获得了 Jolt 奖之后才能真正成为行业的主流,一本技术书籍只有在获得了 Jolt 奖之后才能真正奠定其作为经典的地位。虽然 Jolt 奖项并不起决定作用,但它代表了某种技术趋势与潮流。——译者注

目 录

第一部分 Qt 基础

第 1 章 Qt 入门	2
1.1 Hello Qt	2
1.2 建立连接	4
1.3 窗口部件的布局	4
1.4 使用参考文档	7
第 2 章 创建对话框	10
2.1 子类化 QDialog	10
2.2 深入介绍信号和槽	15
2.3 快速设计对话框	17
2.4 改变形状的对话框	23
2.5 动态对话框	29
2.6 内置的窗口部件类和对话框类	30
第 3 章 创建主窗口	35
3.1 子类化 QMainWindow	35
3.2 创建菜单和工具栏	39
3.3 设置状态栏	43
3.4 实现 File 菜单	44
3.5 使用对话框	49
3.6 存储设置	54
3.7 多文档	55
3.8 程序启动画面	57
第 4 章 实现应用程序的功能	59
4.1 中央窗口部件	59
4.2 子类化 QTableWidgetItem	60
4.3 载入和保存	64
4.4 实现 Edit 菜单	66
4.5 实现其他菜单	70
4.6 子类化 QTableWidgetItem	73
第 5 章 创建自定义窗口部件	80
5.1 自定义 Qt 窗口部件	80
5.2 子类化 QWidget	81

5.3	在 Qt 设计师中集成自定义窗口部件	89
5.4	双缓冲	92

第二部分 Qt 中级

第 6 章	布局管理	108
6.1	在窗体中摆放窗口部件	108
6.2	分组布局	113
6.3	切分窗口	114
6.4	滚动区域	117
6.5	停靠窗口和工具栏	118
6.6	多文档界面	121
第 7 章	事件处理	129
7.1	重新实现事件处理器	129
7.2	安装事件过滤器	133
7.3	处理密集时的响应保持	135
第 8 章	二维图形	138
8.1	用 QPainter 绘图	138
8.2	坐标系统变换	142
8.3	用 QImage 高质量绘图	148
8.4	基于项的图形视图	150
8.5	打印	166
第 9 章	拖放	173
9.1	使拖放生效	173
9.2	支持自定义拖动类型	177
9.3	剪贴板处理技术	181
第 10 章	项视图类	182
10.1	使用项视图的简便类	183
10.2	使用预定义模型	187
10.3	实现自定义模型	191
10.4	实现自定义委托	203
第 11 章	容器类	207
11.1	连续容器	207
11.2	关联容器	213
11.3	通用算法	216
11.4	字符串、字节数组和变量	217
第 12 章	输入与输出	223
12.1	读取和写入二进制数据	223
12.2	读取和写入文本	228
12.3	遍历目录	232

12.4	嵌入资源	233
12.5	进程间通信	234
第 13 章	数据库	238
13.1	连接和查询	238
13.2	查看表	243
13.3	使用窗体编辑记录	245
13.4	在表中显示数据	249
第 14 章	多线程	255
14.1	创建线程	255
14.2	同步线程	258
14.3	与主线程通信	263
14.4	在次线程中使用 Qt 的类	268
第 15 章	网络	270
15.1	写 FTP 客户端	270
15.2	写 HTTP 客户端	277
15.3	写 TCP 客户/服务器应用程序	279
15.4	发送和接收 UDP 数据报	287
第 16 章	XML	291
16.1	使用 QDomStreamReader 读取 XML	291
16.2	用 DOM 读取 XML	297
16.3	使用 SAX 读取 XML	300
16.4	写入 XML	304
第 17 章	提供在线帮助	306
17.1	工具提示、状态提示和“ What’s This?”帮助	306
17.2	利用 Web 浏览器提供在线帮助	307
17.3	将 QTextBrowser 作为简单的帮助引擎	309
17.4	使用 Qt Assistant 提供强大的在线帮助	311

第三部分 Qt 高级

第 18 章	国际化	314
18.1	使用 Unicode	314
18.2	让应用程序感知翻译	317
18.3	动态切换语言	322
18.4	翻译应用程序	326
第 19 章	自定义外观	329
19.1	使用 Qt 样式表	329
19.2	子类化 QStyle	340
第 20 章	三维图形	352
20.1	使用 OpenGL 绘图	352

20.2	OpenGL 和 QPainter 的结合	356
20.3	使用帧缓存对象生成叠加	361
第 21 章	创建插件	366
21.1	利用插件扩展 Qt	366
21.2	使应用程序感知插件	374
21.3	编写应用程序的插件	377
第 22 章	应用程序脚本	379
22.1	ECMAScript 语言概述	379
22.2	使用脚本扩展 Qt 应用程序	386
22.3	使用脚本实现 GUI 扩展	389
22.4	使用脚本自动化处理任务	394
第 23 章	平台相关特性	403
23.1	连接本地的应用程序编程接口	403
23.2	在 Windows 上使用 ActiveX	406
23.3	处理 X11 会话管理	416
第 24 章	嵌入式编程	421
24.1	从 Qt/Embedded Linux 开始	421
24.2	自定义 Qt/Embedded Linux	423
24.3	Qt 应用程序与 Qtopia 的集成	424
24.4	使用 Qtopia 的 API	427

第四部分 附录

附录 A	Qt 的获取和安装	436
A.1	协议说明	436
A.2	Qt/Windows 的安装	436
A.3	Qt/Mac 的安装	437
A.4	Qt/X11 的安装	437
附录 B	编译 Qt 应用程序	439
B.1	使用 qmake	439
B.2	使用第三方编译工具	443
附录 C	Qt Jambi 简介	447
C.1	Qt Jambi 入门	447
C.2	在 Eclipse IDE 中使用 Qt Jambi	451
C.3	在 Qt Jambi 中集成 C++ 组件	455
附录 D	面向 Java 和 C# 程序员的 C++ 简介	461
D.1	C++ 入门	461
D.2	主要语言之间的差异	465
D.3	C++ 标准库	489

第一部分 Qt 基础

第1章 Qt 入门

第2章 创建对话框

第3章 创建主窗口

第4章 实现应用程序的功能

第5章 创建自定义窗口部件