

附录 A 常见问题

正文未及之处或需导引之点，列于此处。

该怎么学习 Qt Quick

先说我的经历吧，因为写作本书时市面上还没有系统介绍 Qt Quick 的书籍，我是跟着 Qt 帮助文档和示例一路走过来的，不明白的就仔细研读、试验，为了博闻强记，我甚至放着支持语法高亮的 Qt Creator 不用，故意使用 Notepad++ 和 UltraEdit 来编辑 QML 文档，为的就是多记忆一些类库啊。如果帮助和试验不解决问题，我就阅读 Qt 5.3 的源代码，最后总会找到办法的。虽然身边没有人与我同行，但有帮助与源码在，也不孤独。凡事问帮助，还不清楚问源码，这是我的不二法门。

本书面世后，再学习 Qt Quick 就不用那么费劲了。只要将本书内容通读一遍，相信你就能入门了。虽然我没有覆盖 Qt Quick 的所有主题，比如粒子特效、着色器、数据库等，但核心的概念，比如 QML 语法、ECMAScript、信号与槽、QML 与 C++ 混合编程、Component、Canvas、Model-View-Delegate 等都讲得很明白，有了这些基石，再加上 Qt 帮助，再加上你无敌的聪明才智，横扫一切障碍不成问题。

如何发布 Qt Quick 应用

我们只说动态库版本，即 Windows 和 Android 平台。

(1) Windows 平台

Qt SDK 里有个工具叫 `windeployqt.exe`，你可以使用它找到你的应用所依赖的动态库。打开 Qt SDK 命令行环境（参考 2.5.2 节），输入 `windeployqt`，敲下回车键，应该就可以看到说明了。一般使用下面的命令就可以把依赖都搬运到你的项目目录下：

```
windeployqt --qmldir <path-to-app-qml-files> <path-to-app-binary>
```

举个例子吧。

将 13.8 节的找图看实例的可执行程序 `ImageViewer.exe` 放到 `d:\projects\qtquick\Image`

Viewer 目录下。初始效果如图 A-1 所示。

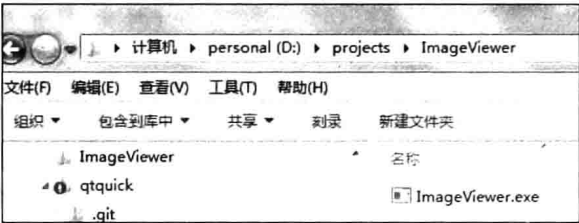


图 A-1 找图看安装目录初始情况

执行下面的命令（我把 qml 放到 qrc 内了，所以没有指定应用的 qml 目录）：

```
windeployqt D:\projects\ImageViewer\ImageViewer.exe
```

可以看到如图 A-2 所示的效果，windeployqt 开始干活了。



图 A-2 windeployqt 在复制依赖文件

稍后你会看到如图 A-3 所示的效果。

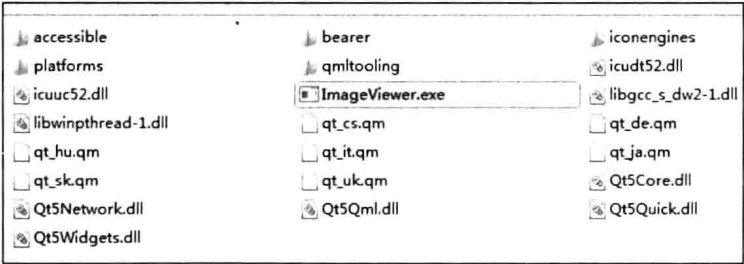


图 A-3 windeployqt 运行过后

其实 windeployqt 复制了很多内容，比如 Qt 应用框架本身的 DLL，如 Qt5Core.dll；如翻译文件 qm，如 Qt 库依赖的第三方动态库 libstdc++-6.dll 等。

这个有没有多余的？可能有，比如你可以删除一些 qm 文件，或者其他的库，自己试试就知道了，只要不影响应用运行的就是多余的。

这个已经全了吗？非也！你还需要把 Qt SDK 目录下的 qml 目录（如 C:\Qt\Qt5.3.1\5.3\

mingw482_32\qml) 都复制过来。有人说这忒多了吧, 是的, 很多, 你可以只复制你用到的目录, 比如 QtQuick.2、QtQuick 等。其实 QtQuick 下面有很多你没用到的, 追求完美的话, 可以一一剔除掉, 只保留 *.qml、*.qmltypes、*.js 即可。

到此为止, 你的应用依赖的那些幺蛾子们基本复制齐了。如果你不放心, 可以把 Qt SDK 的目录改个名字 (以防你设置了系统环境变量, 干扰你的发布), 再运行一下你的应用试试看, 如果报错, 说明还差点儿东西; 如果正常运行, 恭喜你, 快通关了。

其实还有一个东西, 那就是 VC 运行时库, 以我的开发环境为例, 它在 C:\Qt\Qt5.3.1\vcredist 目录下, 名字是 “vcredist_spl_x86.exe”, 这个需要你制作安装包时带进去, 并在用户环境未安装 VC 运行时库时自动把它装上。

好啦, 万事俱备, 该去找个安装包制作工具了, 建议使用 NSIS。虽然 Winamp 已成昨日黄花, 可与它有裙带关系的 NSIS, 依然应用广泛。如何使用 NSIS 制作安装包, 超出了本书的范畴, 找“度娘”问一下吧。当然, 你也可以到我的博客 (<http://blog.csdn.net/foruok>) 上面查看, 那里有一篇文章, 完整地介绍了怎样为 Qt 应用制作 Windows 安装包。

(2) Android 版本

其实这个似乎没有必要说了, 你安装了支持 Android 平台的 Qt SDK, 创建支持 Android 的应用, 选择运行, 选择部署策略 (use Ministro 或者 Bundle Qt libraries in APK), Qt Creator 会自动为你生成一个 APK, 那个 APK 就是安装包了。项目的 Android 选项设置请参考 2.3.3 节或者笔者的另外一本书《Qt on Android 核心编程》。

怎样保持 Android 手机屏幕常亮

最简单的办法是更改手机显示设置中的屏幕待机选项, 让它像万里长城长又长。啊, 什么, 你要在代码中控制, 打开应用时才常亮? 好吧, 你需要从 QtActivity 继承, 重写 onCreate 方法, 加入下面的代码:

```
getWindow().addFlags(android.view.WindowManager.LayoutParams.FLAG_KEEP_SCREEN_ON);
```

如果你还想友好一些, 在你的活动停止时关闭常亮标记, 请重写 onStop 方法, 加入下面的代码:

```
getWindow().removeFlags(android.view.WindowManager.LayoutParams.FLAG_KEEP_SCREEN_ON);
```

不知道 QtActivity 是什么东西, 不知道如何在 Qt 项目中添加 Java 源文件, 不知道如何使用 JNI, 好吧, 请参考笔者的《Qt on Android 核心编程》。

我想创建可扩展的 UI 来适应各种屏幕

使用 BorderImage, 结合 PPI、DPI 等可以实现这种需求, 请参考本书第 18 章和 KDAB 的系列文章:

- <http://www.kdab.com/scalable-uis-qml/>
- <http://www.kdab.com/pixels-trust-scalable-uis-qml-part-2/>
- <http://www.kdab.com/borderimage-scaling-scalable-uis-2-2/>

Qt Quick 应用如何国际化

请参考 13.5 节，亲，那里有详细说明和示例，包教包会，不会就再看再学，不另外收费。

Qt Quick 应用体积太大了

我没有好办法。我只有两个建议：

- 只链接你需要的模块，泛指 Qt Quick 模块和 Qt C++ 模块。
- 请尝试自己编译 Qt 库，裁剪不需要的功能。不知道怎么编译？请看下一个问题，有链接有真相。

有没有静态库版本的 Qt

先参考这些链接：

- http://qt-project.org/wiki/How_to_build_a_static_Qt_version_for_Windows_with_gcc
- http://qt-project.org/wiki/Build_Qt_For_Linux
- <http://qt-project.org/wiki/Category:Developing-with-Qt::General>，这个页面 Build 开头的链接都可以参考
- <http://blog.qt.digia.com/blog/2014/08/27/qt-weekly-18-static-linking-with-qt/>

也许你想看中文的，请找“度娘”，让它帮你搜“qt 静态编译”或者“静态编译 qt”，它会给你很多选择。

我不想让别人看见我的 QML 文档

好吧，有两个办法：

- 把所有的 QML 文档都放在 qrc 中。
- 花点儿钱，使用 Qt 企业版，可以把 QML 文档预编译为二进制格式。

Android 上的 Qt Quick 控件看起来和系统风格不一样

Qt 5.3 及以前的版本，只有使用 Ministro 服务时，Qt Quick 以及 Qt Widgets 才有 Android Native 风格，使用 Bundle Qt Libraries in APK 这种部署策略时是没有 Android 本地风格提供的。

期待 5.4 吧，2014 年 9 月 8 日，Alpha 版本发布。链接在这里：<http://blog.qt.digia.com/blog/2014/09/08/qt-5-4-alpha-available/>。

关于 Qt 5.4 的重大改进说明，也在这个文章里了。想了解更详细的，看下面两个链接：

- <http://qt-project.org/wiki/New-Features-in-Qt-5.4>
- <http://qt-project.org/wiki/Qt-5.4-release>

本书以 Qt SDK 5.3.1 为基础哈，接近完稿时 5.4 Alpha 放出来了。

自己做一些 Qt Quick 控件怎么样

Good!

Qt Quick 给我们的最大的彩蛋，就是定制已有控件的自由和创建新控件的便利。

要定制已有控件，参考笔者讲解常见控件时对 XXXStyle 的介绍，如 ButtonStyle、TableViewStyle 等，在第 9 章、第 13 章中，这样的例子随处可见。

要创建新控件，建议以已有的 Qt Quick 图元为基础，通过聚合不同的图元来实现自己的新控件，请参考 13.8.3 节，那里有个 FlatButton 示例。

你还可以使用 Canvas，自己从头绘制控件，请参考第 10 章来了解 Canvas 的细节。当然，你还要处理事件，请参考第 6 章来了解在 Qt Quick 中如何处理事件。最后呢，请参考 13.5.4 节，看如何将 Canvas 和事件结合起来，这节使用 Canvas 实现了 TableView 的单元格内的删除按钮。

Qt Quick 不支持文件操作吗

谁说的？支持！

对于 UTF-8 编码的文本文件，可以曲线救国啊，就用 XMLHttpRequest 了，请参看第 17 章。

对于其他的，也可以写 C++ 代码来访问文件，再混合 QML 编程。请参考 13.3 节和 13.5 节，QFile 很强悍哦，配上 QTextStream、QDataStream，天下无双。

想用数据库来着……

Qt Quick 支持，不过本书没有涉及。你可以到 Qt 帮助中，在索引模式下，键入“Local Storage”查询，可以看到“Qt Quick Examples - Local Storage”这个示例。还可以看到“Qt Quick Local Storage QML Types”页面，专讲支持数据库的 Qt Quick 类型。

说真的，本书示例代码在哪儿可以下载

访问笔者的 github 主页：<https://github.com/foruok>，都在这里了，连截图都有呢。

参 考 文 献

在本书写作过程中参考了诸多书籍或在线文档，笔者要真心感谢他们。下面是参考的书籍和文档等资料列表，再次感谢。

- Qt 帮助文档
- Qt 5.3.1 源代码
- Qt SDK 里的 QML 源码和示例
- <http://qmlbook.org/>，这是本书问世之前为数不多的较为系统的 QML 资料了，英文的，很好很强大
- 《Qt on Android 核心编程》，本人的另一本书，内容经典，深入浅出，值得拥有，也是本书非常非常不错的搭档
- <https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference>
- 《JavaScript 高级程序设计》(Professional JavaScript for Web Developers)，第 5 章参考了此书相关章节，特此说明
- <http://w3school.com.cn/jsref/>，第 5 章多有参考相关页面，特此说明
- <http://www.w3school.com.cn/xpath/>
- <http://www.w3.org/TR/xpath20/>
- <http://www.kdab.com/>上的部分文章
- <http://qt-project.org>
- <http://planet.qt-project.org/>
- Qt 官网 (<http://qt.digia.com/>)
- Qt 官方博客 (<http://blog.qt.digia.com/>)
- <http://blog.sina.com.cn/liang19890820>，这个博客的博主，有时叫“‘☆奋斗 ing♥孩子’”，有时叫“‘☆一去♥二三里’”，有趣
- Android 在线 SDK (<http://developer.android.com/sdk/index.html>)
- winorlose2000 博客 (<http://vaero.blog.51cto.com/>) 中关于图像处理算法的博文
- ian 的个人博客 (<http://www.icodelogic.com/>) 中关于图像处理算法的博文
- “度娘” (<http://www.baidu.com>)，在失去 Google 的世界里，只有你伴我度过漫漫长夜，虽是备胎，可我还是爱死你了

业界热评

Qt以跨平台和高性能著称，只要你在这方面有需求，我相信你一定评估过Qt。但Qt毕竟是一个小众的开发框架，很多人担心它的学习资源不够丰富而难以掌握，害怕真正应用起来会被“坑”，因此望而却步，敬而远之。本书的出版足以解决你的后顾之忧！读一本书，洞察一门有用的技术；一次愉快的对话，在“悦享品质”的过程中交一位有共同语言的朋友——此书值得拥有！

爱奇艺技术总监 陆其明

自己学一个开发框架可能不难，但是要系统地整理出来著书远非易事。本书全面、系统化地介绍了Qt Quick的相关知识，是能给广大从业人员以帮助的好作品。

淘宝技术部总监 曾宪杰（花黎）

Qt Quick是Qt新推的UI框架，快速、简洁、灵活、强大，但相关书籍太少，此书正好填补了空白，全面、基础而且有一定的深度，每个例子都由作者精心选择和测试，可见作者是用心的在写这本书。

浙江力太科技有限公司技术总监、IPMA项目经理 谷伟年

foruok的技术博客写的不错，其博文行云流水、逻辑清晰，我个人一直是其忠实粉丝。《Qt Quick核心编程》别出机杼，在细致、扎实地讲述QML技术的同时，融入了作者多年的一线软件开发实战经验以及文艺程序员特有的幽默，非常适合对编程技术精益求精的开发者阅读。

西安OpenParty组织者之一、葡萄城控件社区经理 王文刚

你想让你的软件拥有直观、现代、如行云流水般的界面吗？你想为其添加动画、音效、过场等效果吗？本书幽默诙谐的语言，生动直观的例子，将带你逐步领略Qt Quick编程的美妙所在。

Nokia (Digia) Certified Qt Specialist, CSDN博客认证专家
张德保(dbzhang800)

《Qt Quick核心编程》花了大量篇幅讲解Controls、Styles、Events、Model-View、Animation、Canvas等主题，甚至连JavaScript都讲了！不仅如此，每个章节还都提供相关示例并详细介绍，充分体现出foruok对读者的关心与细心。foruok的这本书无论是广度上还是深度上都做到了极致，不管是对想入门的朋友还是想进阶提升的朋友来讲，都是不可多得的佳作。

QTCN开发网QML开发版版主，资深Qt/QML开发者
寒山-居士(toby520)

我很荣幸向大家推荐foruok的新书。Qt是一款非常优秀的开发框架，如今Qt的安卓版、iOS版和WP版都可供开发者使用了，如果大家能够掌握Qt的开发，那么就拥有了全平台开发的能力以及对未来软件开发平台的优势。有很多开发者经常向我们询问移动开发和QML的点点滴滴。幸运的是，这本书填补了这方面的空白，你再也不用在黑暗中摸索了！这是真的吗？这是真的。

Qt移动开发先驱、游戏开发者 彩阳

QML和C++混合编程是Qt的趋势，描述性界面和逻辑运算分开也是发展的趋势。此书既有丰富的QML内容，又将如何与Qt C++结合讲述得很到位，对我们一直全部用Qt Widgets (C++) 开发者来说，是打开Qt Quick的一把金钥匙。

QTCN开发网Qt基础编程版版主 渡世白玉

《Qt Quick核心编程》是一本由浅入深、系统实用的Qt Quick编程书籍。作者深谙开发人员的学习曲线，在内容的安排上，使人有“走到哪都水到渠成，学到哪都柳暗花明”的感觉，既适合初触Qt Quick编程的读者，可供有经验的开发者参考进行深入系统的学习。

QTCN开发网Qt代码秀讨论区版主 realfan

上架建议：计算机>程序设计>Qt

ISBN 978-7-121-24684-5



定价：79.00元



博文视点Broadview



@博文视点Broadview



策划编辑：高洪霞
责任编辑：葛娜
封面设计：侯士卿