

# 《Qt及Linux操作系统窗口设计》

## 图书基本信息

书名：《Qt及Linux操作系统窗口设计》

13位ISBN编号：9787121024344

10位ISBN编号：7121024349

出版时间：2006-4

出版社：电子工业出版社

作者：倪继利

页数：449

版权说明：本站所提供下载的PDF图书仅提供预览和简介以及在线试读，请支持正版图书。

更多资源请访问：[www.tushu000.com](http://www.tushu000.com)

# 《Qt及Linux操作系统窗口设计》

## 内容概要

如何在Linux内核上建立窗口系统，这是嵌入式设备软件开发工程师必须了解的。Qt / Embedded是高端嵌入式设备PDA及手机的主流开发工具。作为一项成熟的技术，KDE桌面系统对于嵌入式窗口系统的设计有很大参考价值：Qttopia给嵌入式窗口系统提供了很好的框架。本书不仅阐述了KDE及X Window的机制，分析了Qt及Qt / Embedded开发工具的核心技术，而且还详细介绍了如何在嵌入式设备上建立Qttopia窗口系统。掌握了本书中所介绍的技能，读者可以在Linux内核上建立窗口系统，开发应用程序并能够在PC上模拟运行嵌入式设备开发的应用程序。本书主要针对从事开发Linux应用程序的软件工程师，也很适合作为大学教材和参考书。

第1章“X Window”阐述了X的体系分层结构、显卡驱动程序、X协议，说明了建立在X上的各种编程方法，还分析了X客户端应用程序Xlib函数如何发出请求以及X服务器的工作原理及源代码。

第2章“Qt编程核心技术”介绍了Qt对象模型、国际化方法、元对象及代码生成、进程间通信、窗口部件的基类、模板库和集合类、Qt线程、鼠标拖放、键盘焦点、会话管理，以及调试等方面的技术。

第3章“KDE窗口系统”介绍了KDE的MIME类型处理、服务、KIO框架、组件框架和国际化等，还分析了KDE的启动过程及kicker应用程序。

第4章“Qt / Embedded客户 / 服务器”讲述了将应用程序下载到嵌入式设备运行的方法，介绍了各种嵌入式GUI窗口系统，阐述了Qt / Embedded的客户 / 服务器模型工作过程。本章还分析了设备输入事件在客户与服务器之间的传递机制、应用程序窗口的显示原理，并说明了运行Qt / Embedded程序的方法。

第5章“Qttopia核心技术”介绍了Qttopia开放源代码的核心技术及类库，其中着重分析了插件的装载过程、应用程序快速启动器的机制、类MimeType，以及编写输入法的方法等。

第6章“Qttopia服务器”主要说明组成Qttopia服务器的各个类的功能，介绍了Qttopia服务器的启动过程，分析了Qttopia服务器如何启动应用程序及插件，还分析了电源监控、外观设置、应用程序安装与卸载以及AppServices等与系统管理相关的小应用程序。

第7章“Qttopia的sysinfo应用程序分析”详细分析了sysinfo应用程序的源代码。

第8章“Qttopia编译及系统集成”说明了Qttopia的定制与集成，阐述了，pro项目文件的语法。并且详细分析了由pro项目文件生成Makefile文件的过程以及描述了Qttopia的打包、安装和调试方法。

- 第1章 X Window1.1 X Window的体系结构1.1.1 X Server1.1.2 X Client1.1.3 X Protocol1.2窗口管理器1.3 X Window启动过程1.4 XFree86配置文件分析1.5 X Window程序设计简介1.5.1 Xlib编程1.5.2 Motif编程1.5.3 GTK / GNOME1.5.4 KDE1.6显示驱动程序1.6.1显卡驱动方式1.6.2帧缓冲1.6.3在台式机上使用通用帧缓冲1.6.4帧缓冲设备驱动程序1.7 X Window协议1.7.1 X协议1.7.2 x传输接口1.7.3 XDMCP协议1.8 x client应用程序源代码分析1.8.1 xinit分析1.8.2 XOpenDisplay1.8.3函数XCreateWindow分析1.9 x Server应用程序分析1.9.1 X Server概述1.9.2 DIX层1.9.3 OS层1.9.4 DDX层1.9.5 Screen第2章 Qt编程核心技术2.1 Qt概述2.2 Qt对象模型2.2.1信号和槽2.2.2元对象系统2.2.3元对象编译器限制2.2.4属性2.3 QObject类2.3.1对象树2.3.2事件处理过程2.3.3事件运行机制2.3.4事件过滤器2.3.5定时器2.3.6连接函数connect2.3.7字符串翻译函数2.4 Qt国际化2.4.1软件中字符串国际化方法2.4.2创建译本2.4.3编码支持2.5 QMetaobject元对象类2.5.1相关的数据结构2.5.2 QMetaObject对象2.5.3 Q.OBJECT宏及moc生成代码分析2.6进程间通信——QCopChannel2.7窗口部件类2.7.1窗口部件的基类QWidget2.7.2 QFrame类2.7.3 QScrollView类2.8 Qt风格机制2.8.1风格类2.8.2窗口系统风格更新机制2.9布局类2.10 Qt插件2.11 Qt模板库2.11.1迭代器2.11.2算法2.11.3数据流串行化2.12集合类2.12.1基于指针的容器的结构2.12.2管理集合条目2.12.3迭代器2.13 Qt线程2.13.1线程类QThread2.13.2线程安全的事件传递2.14鼠标拖放2.15键盘焦点2.15.1焦点移动的方式2.15.2焦点策略及操作函数2.16会话管理2.16.1会话管理2.16.2测试和调试会话管理2.17调试技术2.17.1命令行参数2.17.2打印警告和调试消息2.17.3调试宏第3章 KDE窗口系统3.1 KDE 3.5源代码说明3.2 KDE类库3.3系统资源访问3.3.1标准资源目录3.3.2 KDE系统配置缓冲3.4图形支持3.5用户界面3.5.1 Action模式3.5.2用XML定义菜单和工具栏3.5.3在线帮助3.5.4复杂窗口部件3.6 MIME类型处理3.6.1一个应用MIME类型的例子3.6.2映射MIME类型到一个应用程序或服务3.7 KDE服务3.7.1定义服务类型3.7.2定义共享库服务3.7.3定义DCOP服务3.8 KIO框架3.8.1同步使用KIO3.8.2同步使用KIO3.8.3 MetaData3.8.4调度3.8.5定义一个ioslave3.9 DCOP3.9.1 DCOP工具3.9.2 DCOP收发数据3.10 KDE应用程序启动其他应用程序的方法3.11国际化和本地化3.12编译方法3.13桌面框架3.13.1 Panel Applets3.13.2控制中心模块3.14组件框架3.14.1 KParts组件3.14.2系统托盘3.14.3集成Java3.15 KDE协议说明3.16 KDE启动3.16.1 startkde脚本分析3.16.2 startkde中的应用程序说明3.16.3桌面组件3.17 kicker应用程序3.17.1 kicker应用程序的类继承关系3.17.2桌面配置文件3.17.3 kicker窗口的建立过程3.17.4 K菜单第4章 Qt / Embedded客户 / 服务器4.1 各种嵌入式GUI4.1.1 X Window4.1.2 MICROWindows4.1.3 MiniGL4.1.4 QT / Embedded4.2 Qt/Embedded的应用示例4.2.1编译Linux内核4.2.2 root文件系统4.2.3裁剪Qt / Embedded库4.2.4编译应用程序4.2.5拷贝Qt库及应用程序4.2.6打包4.2.7下载及运行4.3帧缓冲4.3.1 Direct1.1B介绍4.3.2虚拟帧缓冲qvfb4.4创建应用程序4.4.1 QApplication类4.4.2 QApplication构造函数4.5构建服务器4.5.1 QWSServer类说明4.5.2 QWSServer类构造函数分析4.6 Qt / Embedded客户端与服务器通信4.6.1服务器相关的类说明4.6.2客户与服务器的接收事件处理4.6.3服务器对窗口的协调管理4.7输入设备与应用程序通信4.7.1鼠标键盘相关类4.7.2打开鼠标设备的过程4.7.3设备文件与socket连接4.7.4服务器捕获设备输入信号4.7.5服务器发送事件给客户4.8应用程序显示4.8.1与显示相关的类4.8.2与字体相关的类4.8.3函数init display4.8.4 QWSDisplay构造函数分析4.8.5 QScreen4.8.6与图形显示相关的类4.8.7图形显示4.9运行Qt / Embedded应用程序4.10字符输入4.11增加一个加速显示卡驱动程序到Qt / Embedded4.12 Qt / Embedded作为VNC服务器第5章 Qtopia的核心技术5.1 Qtopia简介5.1.1 Qtopia PDA5.1.2 Qtopia手机版5.1.3手机库5.1.4 Qtopia Desktop5.2信道和消息5.3国际化5.3.1翻译操作步骤5.3.2.qm文件位置5.3.3 Desktop文件国际化5.3.4文档国际化5.4 Qtopia文件系统标准5.5 Qtopia插件5.5.1建立插件的步骤5.5.2装载插件的方法5.6输入法插件5.6.1创建弹出输入方法5.6.2创建复合输入法5.7快速启动应用程序5.7.1改进主窗口创建时间5.7.2 Quick Launcher5.7.3应用Quick Launcher5.7.4 Quick Launcher的宏定义5.7.5运行Quick Launcher分析5.8 Qtopia核心类库说明5.8.1 QPEApplication类5.8.2应用程序配置与连接类5.8.3 PIM管理类5.8.4桌面相关类5.8.5字体与输入法相关类5.8.6 StorageInfo和FileSystem类5.8.7 DeviceButtonManager和DeviceButton类5.8.8图像处理相关类说明5.8.9多媒体播放相关类说明5.8.10网络相关类说明5.8.11软件模块接口类5.8.12 Service和ServiceRequest类5.8.13 MimeTVpe类5.8.14其他类说明第6章 Qtopia服务器6.1 Qtopia服务器特定的类6.1.1 ServerApplication类6.1.2 Server类6.1.3 Launcher类6.1.4 LauncherTabWidget、LaunchernTabBar和LauncherTab类6.1.5 LauncherView和LauncherItem类6.1.6 AppLauncher和QuickLauncher类6.1.7 LoadingWidget和DocumentList类6.1.8 TaskBar类6.1.9 StartMenu和StartPopupMenu类6.1.10

# 《Qt及Linux操作系统窗口设计》

InputMethodSelector和InputMethods类6.1.11 RunningAppBar和SysTray类6.1.12服务器的其他类说明6.2  
Qtopia服务器启动过程6.2.1 Qtopia服务器的main函数6.2.2 ServerApplication类构造函数6.2.3 Server类构造函数6.2.4创建服务器主窗口界面UI6.2.5创建应用程序图标视图6.3应用程序启动过程6.4文档的打开过程6.5电池监控小应用程序6.5.1电池状态窗口类6.5.2电源状态管理器类6.5.3插件装载6.6外观设置6.6.1  
main函数6.6.2 SampleWindow类6.6.3 AppearanceSettings类6.6.4 AppearanceSettings类的构造函数分析6.7安  
装与卸载应用程序6.8 AppServices应用程序6.8.1 AppServices类6.8.2 ASCheckListItem类6.9插件管理器第7  
章 Qtopia的sys ; nfo应用程序分析7.1 SystemInfo类7.2 VersionInfo类7.3 StorageInfoView类7.4负载信  
息LoadInfo7.5内存信息7.6 DataView类第8章 Qtopia编译及系统集成8.1 qmake介绍8.1.1 qmake编译过  
程8.1.2.pro文件的语法分析8.2 Qtopia编译方法8.2.1 Qtopia编译步骤8.2.2 Qtopia Desktop编译步骤8.2.3编  
译一个应用程序8.2.4使用NFS运行应用程序8.2.5应用程序调试8.3为设备定制Qtopia8.3.1定制Qtopia启动  
器用户界面8.3.2设备硬件因素8.3.3不安装及不支持的组件8.3.4安全模式需要的插件8.3.5电源管理8.3.6可  
移去的存储卡8.3.7 118N8.3.8访问权限和仪读文件系统8.3.9 MMS客户端8.3.10 GSM模块集成8.3.11定制键  
盘8.3.12配置硬件按钮8.4系统集成8.4.1创建设备特定交叉编译工具配置文件8.4.2定制电源、背景灯控制和  
键盘扫描码8.4.3在Qt / Embedded中创建键盘处理函数8.4.4配置指针设备8.4.5配置适合设备的Qt  
 / Embedded库8.4.6配置适合设备的Qtopia8.4.7为目标设备创建和安装Qtopia image文件8.4.8在设备上配  
置适合于Qtopia的环境变量8.4.9安装另外的字体8.4.10集成Java 8.5 Qtopia编译系统8.5.1内部的编译  
系统项目文件说明8.5.2项目文件生成：Makefile8.5.3 src—componentspro和Srcpro文件8.5.4 configure脚本  
生成Makefile8.6 Qtopia的编译过程8.6.1编译库8.6.2相互独立的应用程序及插件的编译8.6.3文件的安  
装8.6.4打包分发8.7生成img文件并烧录主要参考文献

# 《Qt及Linux操作系统窗口设计》

## 媒体关注与评论

本书特点：  
实际：作者在实践中使用QT开发嵌入式设备的就用程序界面，在本书中，他与读者分离自己在外企从事Linux研发的经验。  
实用：具有专业人士所需的深度，可以直接拿来运用到开发实战中去。本书介绍的KDE及X Windows机制、Qt及Qt/Embedded核心技术，有助于读者在嵌入式设备的Linux内核上创建自己的窗口系统。  
实用：重点介绍了一些对实际研发很有用的设计方法，包括窗口分层设计、接口标准化、菜单通用化、应用程序查询硬件信息等。

# 《Qt及Linux操作系统窗口设计》

## 编辑推荐

如何在Linux内核上建立窗口系统，这是嵌入式设备软件开发工程师必须了解的。Qt / Embedded是高端嵌入式设备PDA及手机的主流开发工具。作为一项成熟的技术，KDE桌面系统对于嵌入式窗口系统的设计有很大参考价值：Qt/Embedded给嵌入式窗口系统提供了很好的框架。本书不仅阐述了KDE及X Window的机制，分析了Qt及Qt / Embedded开发工具的核心技术，而且还详细介绍了如何在嵌入式设备上建立Qt/Embedded窗口系统。掌握了本书中所介绍的技能，读者可以在Linux内核上建立窗口系统，开发应用程序并能够在PC上模拟运行为嵌入式设备开发的应用程序。本书主要针对从事开发Linux应用程序的软件工程师，也很适合作为大学教材和参考书。

# 《Qt及Linux操作系统窗口设计》

## 精彩短评

- 1、很大，很厚，字小，前后关联介绍不清楚。
- 2、有点过时了.....  
不过前面xlib、以及中间关于Qt的原理部分，讲的还是不错的。
- 3、随着QT版本的更新，这本书过时了

# 《Qt及Linux操作系统窗口设计》

## 版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:[www.tushu000.com](http://www.tushu000.com)