



深入剖析Linux GUI编程与MySQL设计实例
图形化应用程序开发者必备



邱铁 周玉 张民垒 等编著

Linux-Based Qt4 GUI and MySQL Programming

Linux环境下Qt4图形界面 与MySQL编程



机械工业出版社
China Machine Press



Linux-Based Qt4 GUI and MySQL Programming

Linux环境下Qt4图形界面 与MySQL编程

邱铁 周玉 张民垒 等编著



机械工业出版社
China Machine Press

在 Linux 操作系统环境下，图形界面与数据库编程技术越来越受到广泛重视。本书以“图形界面编程控件与数据库编程基础→简单易学的实例→实际工程项目开发与场景分析”为写作主线，以当前最新的 Qt4.7 为依据，采用“深入分析控件+实例解析”的方式，并配合经典的实际工程项目，对 Linux 操作系统下的 Qt4.7 与 MySQL 编程技术进行了全面细致的讲解。

本书注重基础学习，并深入实践。读者可以从零开始学习图形界面编程开发与数据库技术。

本书可作为高等院校计算机、电子、信息类大学生及研究生进行基于 Linux 的图形界面编程开发与数据库技术学习的参考书，也可作为图形界面与数据库编程爱好者开发的参考用书。

封底无防伪标均为盗版

版权所有，侵权必究

本书法律顾问 北京市展达律师事务所

图书在版编目 (CIP) 数据

Linux 环境下 Qt4 图形界面与 MySQL 编程 / 邱铁等编著. —北京: 机械工业出版社, 2012. 3
(原创精品系列)

ISBN 978-7-111-37291-2

I. L… II. 邱… III. ①Linux 操作系统—程序设计 ②软件工具—程序设计 ③关系数据库—数据库管理系统, MySQL—程序设计 IV. ①TP316.89 ②TP311.56 ③TP311.138

中国版本图书馆 CIP 数据核字 (2012) 第 014301 号

机械工业出版社 (北京市西城区百万庄大街 22 号 邮政编码 100037)

责任编辑: 秦 健

北京京师印务有限公司印刷

2012 年 3 月第 1 版第 1 次印刷

186mm × 240mm · 31.25 印张

标准书号: ISBN 978-7-111-37291-2

定价: 79.00 元

凡购本书, 如有缺页、倒页、脱页, 由本社发行部调换

客服热线: (010) 88378991; 88361066

购书热线: (010) 68326294; 88379649; 68995259

投稿热线: (010) 88379604

读者信箱: hzsj@hzbook.com

前 言

Linux 操作系统作为源码开放的自由软件，经过近 20 年的发展与壮大，越来越受到 IT 界的认可，在信息技术领域发挥着重要作用。图形界面是 Linux 走向成熟的重要支撑技术，备受众多开发者的重视。在每个工程项目中，几乎都离不开数据库技术。因此，稳定的 OS、友好的图形界面和完善的数据库技术构成了一个完整的工程项目。

面对 Linux 操作系统图形界面与数据库编程，很多学习者不知道如何下手。本书正是在这种背景下编写的，紧紧围绕本书的写作主线“图形界面编程控件与数据库编程基础→简单易学的实例→实际工程项目开发与场景分析”，以当前最新的 Qt4.7 为依据，采用“深入分析控件+实例解析”的方式，并配合经典的实际工程项目，对 Linux 操作系统下的 Qt4.7 与 MySQL 编程技术进行了全面细致的讲解。本书主要分为以下三大部分：

- Linux 图形界面编程基础。介绍 Qt4.7 的全部控件，并针对每个控件设计了一个简单易学的实例，抛砖引玉，加深读者对 Qt4.7 控件的认识和理解。
- 基于 Linux 操作系统的 MySQL 数据库设计基础。介绍 MySQL 基本操作，并针对每个操作设计了一个简单易学的实例，抛砖引玉，加深读者对 MySQL 操作的认识和理解。
- 基于 Qt4.7 与 MySQL 的经典的实际工程项目案例开发。在本书中设计了列车时刻表查询系统、酒店客房管理系统、房屋租赁系统、书店管理系统、学生上机考试系统、校园点菜系统、餐饮信息服务系统、视频音频播放器、桌面常用软件小助手、俄罗斯方块游戏和局域网聊天系统。这些案例给读者提供了实际工程项目开发参考。

在本书的编写过程中，参考和借鉴了很多资料，它们为本书的编写和实验例程的解决方案提供了重要的指导作用。本书中的范例源代码可以到华章网站（www.hzbook.com）下载。

本书主要由邱铁、周玉、张民垒等完成编著任务。在此，感谢所有参与本书构思、解决方案、编辑和出版工作的同事、同行和为本书编写提供灵感的同志们，其中马利超参加了代码调试和部分文档整理，陈方疏、易磊、刘继伟、王革正、陈坚、宋莉莉、张涛、郝若男、刘晗、张晓彤、王宇辰、高凡等在课程设计中的构思被本书采纳，并被设计成典型案例，向他们表示感谢。

开源项目还在向前飞速发展，Qt 与 MySQL 版本还在不断更新，书中难免存在错误和不妥之处，恳请读者批评指正，并将信息发送到 openlinux2011@gmail.com，我们会尽力及时答复。

编者

2012 年 2 月

目 录

前言

第 1 章 绪论 1

1.1 图形界面设计的必要性 1

1.2 Linux 常用的图形化界面编程工具 1

1.2.1 Qt 1

1.2.2 GTK + 2

1.3 两种常用的 Linux 数据库 3

1.3.1 MySQL 3

1.3.2 SQLite 3

1.4 图形界面与数据库综合编程技术 4

第 2 章 Qt4 图形界面编程基础 5

2.1 Qt Creator 5

2.2 安装 Qt 环境及 Qt Creator 5

2.3 信号和槽机制 8

2.4 一个抛砖引玉的实例 9

第 3 章 Qt4 控件与示例精讲 12

3.1 布局 (Layouts) 12

3.1.1 绝对布局 12

3.1.2 手工布局 12

3.1.3 Qt 布局管理器 13

3.2 间隔器 (Spacers) 14

3.3 按钮 (Buttons) 14

3.3.1 QPushButton 控件 14

3.3.2 RadioButton 控件 15

3.3.3 CheckBox 控件 16

3.3.4 示例 1: QPushButton、RadioButton 和 CheckBox 控件的综合应用 17

3.3.5 ToolButton 控件 21

3.3.6 示例 2: ToolButton 的应用 22

3.3.7 CommandLinkButton 控件 24

3.3.8 示例 3: CommandLinkButton 的 应用 25

3.3.9 ButtonBox 控件 29

3.3.10 示例 4: ButtonBox 的应用 30

3.4 单元视图 (Item Views) 32

3.4.1 ListView 控件 33

3.4.2 示例 5: ListView 的应用 33

3.4.3 TreeView 控件 36

3.4.4 示例 6: TreeView 的应用 37

3.4.5 TableView 控件 40

3.4.6 示例 7: TableView 的应用 42

3.4.7 ColumnView 控件 43

3.5 单元组件 (Item Widgets) 44

3.5.1 ListWidget 控件 44

3.5.2 TreeWidget 控件 46

3.5.3 TableWidget 控件 48

3.5.4 示例 8: TableWidget 的示例 50

3.6 容器 (Containers) 52

3.6.1 GroupBox 控件 52

3.6.2 ScrollArea 控件 53

3.6.3 示例 9: GroupBox 和 ScrollArea 的 示例 53

3.6.4 ToolBox 控件 56

3.6.5 示例 10: ToolBox 的应用 57

3.6.6 TabWidget 控件 60

3.6.7 示例 11: TabWidget 的应用 61

3.6.8 StackedWidget 控件 63

3.6.9 示例 12: StackedWidget 的应用 64

3.6.10 Frame 控件 66

3.6.11	Widget 控件	66	3.8.10	ProgressBar 控件	101
3.6.12	MdiArea 控件	67	3.8.11	示例 23: ProgressBar 的应用	102
3.6.13	示例 13: MdiArea 的应用	68	3.8.12	Line 控件	104
3.6.14	DockWidget 控件	71	第 4 章 Qt4 与 MySQL 数据库编程	105	
3.6.15	示例 14: DockWidget 的示例	71	4.1	数据库编程概述	105
3.7	输入组件 (Input Widgets)	73	4.2	MySQL 的特性	105
3.7.1	ComboBox 控件	74	4.3	Qt 与 MySQL 连接	106
3.7.2	Font ComboBox 控件	75	4.4	Qt 中的数据库查询方法	110
3.7.3	LineEdit 控件	75	4.4.1	基本的查询方法	110
3.7.4	TextEdit 控件	76	4.4.2	一个简单实例	117
3.7.5	PlainTextEdit 控件	76	4.5	MySQL 数据库操作语句	119
3.7.6	示例 15: ComboBox、LineEdit 和 TextEdit 的应用	77	4.5.1	查询	120
3.7.7	SpinBox 控件	79	4.5.2	插入	120
3.7.8	Double SpinBox 控件	80	4.5.3	建立数据库	120
3.7.9	Slider 控件	81	4.5.4	修改	121
3.7.10	示例 16: SpinBox、Double SpinBox 和 Slider 的应用	81	4.5.5	删除	121
3.7.11	Dial 控件	83	第 5 章 基于 Qt4 的列车时刻表查询系统	123	
3.7.12	示例 17: Dial 的应用	84	5.1	任务分析	123
3.7.13	ScrollBar 控件	85	5.2	数据库的建立	123
3.7.14	DateEdit 控件	85	5.2.1	数据库表格说明	123
3.7.15	TimeEdit 控件	86	5.2.2	通过 SQL 语句建立数据库和表	124
3.7.16	DateTimeEdit 控件	87	5.2.3	通过 MySQL Administrator 创建 数据库	124
3.7.17	示例 18: DateEdit、TimeEdit 和 DateTimeEdit 的应用	87	5.2.4	向表中添加原始数据项	129
3.8	显示组件 (Display Widgets)	90	5.3	建立 Qt4 工程及系统界面	130
3.8.1	Label 控件	90	5.3.1	项目创建及数据库连接	130
3.8.2	TextBrowser 控件	91	5.3.2	代码编辑	133
3.8.3	示例 19: TextBrowser 的应用	92	5.4	编译及运行	138
3.8.4	GraphicsView 控件	93	第 6 章 基于 Qt4 与 MySQL 的酒店客房 管理系统设计	140	
3.8.5	示例 20: GraphicsView 的应用	94	6.1	任务分析	140
3.8.6	Calendar 控件	97	6.2	数据库的建立	141
3.8.7	示例 21: Calendar 的应用	98	6.2.1	数据库说明	141
3.8.8	LCDNumber 控件	99	6.2.2	通过 SQL 语句创建数据库和表	142
3.8.9	示例 22: LCDNumber 的应用	100	6.2.3	MySQL Administrator 创建数据库	144
			6.2.4	向表中添加原始数据项	148

6.3	建立工程	149	7.5	本项目中的问题说明与讨论	218
6.4	绘制应用程序界面	152	第8章	基于Qt4的书店管理系统设计	219
6.4.1	登录界面设计 (login.ui)	152	8.1	任务分析	219
6.4.2	密码修改界面 (chgpwd.ui)	153	8.2	数据库的建立	219
6.4.3	系统主界面设计 (mainwindow.ui)	153	8.2.1	数据库表格说明	219
6.5	添加代码	156	8.2.2	通过SQL语句建立数据库和表	222
6.5.1	数据库连接 (databaseconnection.h)	156	8.2.3	通过MySQL Administrator 创建 数据库	223
6.5.2	登录窗体类实现	156	8.2.4	向表中添加原始数据项	226
6.5.3	密码修改窗体类实现	158	8.3	建立Qt4工程及系统界面	227
6.5.4	系统主窗体实现	160	8.3.1	项目创建及数据库连接	227
6.5.5	main.cpp 文件实现	168	8.3.2	代码编辑	233
6.6	编译及运行	168	8.4	编译及运行	245
第7章	基于Qt4与MySQL的房屋租赁 系统设计	171	8.5	开发中的注意事项	247
7.1	任务分析	171	第9章	基于Qt4与MySQL的考试系统 设计与实现	248
7.2	数据库的建立	173	9.1	任务分析	248
7.2.1	数据库表格说明	173	9.2	数据库的建立	249
7.2.2	通过SQL语句建立数据库和表	174	9.2.1	数据库表格说明	249
7.2.3	MySQL Administrator 创建数据库	176	9.2.2	通过SQL语句建立数据库 和表	250
7.2.4	向表中添加原始数据项	176	9.2.3	MySQL Administrator 创建 数据库	252
7.3	建立Qt4工程及系统界面	177	9.2.4	向表中添加原始数据项	252
7.3.1	新建应用程序	177	9.3	建立Qt4工程及系统界面	252
7.3.2	绘制应用程序界面	178	9.3.1	新建应用程序	253
7.3.3	添加代码	186	9.3.2	绘制应用程序界面	253
7.4	编译与运行	213	9.3.3	添加代码	258
7.4.1	登录界面	213	9.4	编译与运行	277
7.4.2	系统主界面	214	9.4.1	登录界面	277
7.4.3	房主信息界面	214	9.4.2	管理员主界面	277
7.4.4	添加房主界面	214	9.4.3	添加教师界面	278
7.4.5	删除房主界面	215	9.4.4	添加学生界面	278
7.4.6	房屋信息界面	215	9.4.5	教师主界面	278
7.4.7	添加房屋界面	216	9.4.6	学生主界面	279
7.4.8	删除房屋界面	216	9.5	本项目中的问题说明与讨论	280
7.4.9	房屋查询界面	216			
7.4.10	入住登记界面	217			
7.4.11	客户查询界面	217			

第 10 章 基于 Qt4 与 MySQL 的校园点菜系统	281	11.4.9 打印订单与账单	350
10.1 任务分析	281	11.5 本项目中的问题说明与讨论	350
10.2 数据库的建立	281	第 12 章 基于 Qt4 的视频音频播放器设计	351
10.2.1 数据库表格说明	281	12.1 任务分析	351
10.2.2 通过 SQL 语句建立数据库和表	282	12.2 建立 Qt4 工程及系统界面	351
10.2.3 通过 MySQL Administrator 创建数据库	283	12.2.1 项目创建	351
10.2.4 向表中添加原始数据项	286	12.2.2 代码编辑	356
10.3 建立 Qt4 工程及系统界面	287	12.3 编译及运行	366
10.3.1 项目创建及数据库连接	287	12.4 开发中的注意事项	367
10.3.2 代码编辑	296	第 13 章 基于 Qt4 的桌面常用软件小助手	368
10.4 编译及运行	312	13.1 任务分析	368
10.5 开发中的注意事项	315	13.2 建立 Qt4 工程及系统界面	370
第 11 章 基于 Qt4 与 MySQL 的餐饮信息服务系统设计	316	13.2.1 新建应用程序	370
11.1 任务分析	316	13.2.2 绘制应用程序界面	371
11.2 数据库的建立	317	13.3 添加代码	373
11.2.1 数据库表格说明	317	13.3.1 建立信号与槽函数映射机制	373
11.2.2 通过 SQL 语句建立数据库和表	317	13.3.2 添加代码	376
11.2.3 MySQL Administrator 创建数据库	318	13.4 编译与运行	414
11.2.4 向表中添加原始数据项	318	13.5 本项目中的问题说明与讨论	416
11.3 建立 Qt4 工程及系统界面	319	第 14 章 基于 Qt4 与 MySQL 的俄罗斯方块设计与实现	417
11.3.1 新建应用程序	319	14.1 任务分析	417
11.3.2 绘制应用程序界面	320	14.2 数据库的建立	418
11.3.3 添加代码	324	14.2.1 数据库表格说明	418
11.4 编译与运行	348	14.2.2 通过 SQL 语句建立数据库和表	419
11.4.1 系统主界面	348	14.2.3 MySQL Administrator 创建数据库	419
11.4.2 菜目管理界面	348	14.2.4 向表中添加原始数据项	419
11.4.3 订餐界面	349	14.3 建立 Qt4 工程及系统界面	420
11.4.4 客户信息查询界面	349	14.3.1 新建应用程序	421
11.4.5 结账界面	349	14.3.2 绘制应用程序界面	421
11.4.6 添加菜目界面	349	14.3.3 添加代码	423
11.4.7 更新/删除菜目界面	350	14.4 编译与运行	444
11.4.8 订单详细界面	350	14.4.1 游戏主界面	444

14.4.2	游戏开始界面	445	15.2.4	向表中添加原始数据项	450
14.4.3	排行榜界面	445	15.3	建立 Qt4 工程及系统界面	450
14.4.4	游戏结束界面	446	15.3.1	新建应用程序	451
14.4.5	进入排行榜界面	446	15.3.2	绘制应用程序界面	451
14.5	本项目中的问题说明与讨论	447	15.3.3	添加代码	455
第 15 章	基于 Qt4 与 MySQL 的局域网聊天		15.4	编译与运行	486
	系统设计与实现	448	15.4.1	登录界面	486
15.1	任务分析	448	15.4.2	账号注册界面	486
15.2	数据库的建立	449	15.4.3	系统主界面	486
15.2.1	数据库表格说明	449	15.4.4	聊天界面	487
15.2.2	通过 SQL 语句建立数据库		15.4.5	发送文件界面	487
	和表	450	15.4.6	接收文件界面	487
15.2.3	MySQL Administrator 创建		15.5	本项目中的问题说明与讨论	488
	数据库	450	参考文献	489

第 1 章 绪 论

本章目的:

- 了解图形界面设计的必要性
- 认识几种常用的图形化界面编程工具
- 认识几种常用的 Linux 数据库技术
- 图形界面与数据库综合编程

1.1 图形界面设计的必要性

在信息技术高速发展的今天,无论是 PC 应用软件还是嵌入式设备,人们对软件的友好支持性要求逐渐提高,因此出现了图形用户界面(Graphical User Interface, GUI)技术。它是一种综合了计算机技术、美学、心理学、行为学以及各商业领域需求分析的人机系统工程,强调“人一机一环境”三者作为一个系统进行总体设计。图形用户界面技术的出现,改变了传统的采用终端命令行显示与控制的方式,从可视化、交互性和友好性等方面带来了极大的优越性。

控件功能的可视化。在图形用户界面编程工具中,将不同的用户需求做成控件的形式,在编程时,可以从控件库中选择相应的控件进行设置,从而达到自己的功能需求。每个控件的功能都能直观地显示出来,具有良好的可视化效果。

图形用户界面建立了与用户的互动交流。采用终端作为控制,可操作性差。当程序执行时,与用户的交互性差。图形用户界面技术将用户与程序执行过程控制紧密结合起来,从而使用户的需求及时地在图形用户界面中得以实施并直观地显示出来。

用户界面友好性。图形用户界面的设计要考虑人的行为学和心理学特点,符合用户的习惯,并在美学方面达到和谐统一。使用户操作方便,给人以舒适的感觉。

1.2 Linux 常用的图形化界面编程工具

本节介绍在 Linux 操作系统中编程时,常用的两种编程工具:Qt 和 GTK+。

1.2.1 Qt

Qt 是一个跨平台的图形界面开发平台,最早由挪威的 TrollTech 公司于 1992 年进行开发。2008 年 6 月, NOKIA 公司获得了 Qt 的开发权,继续对 Qt 平台进行开发。Qt 图形界面开发平台可进行嵌入式系统应用程序和桌面系统应用程序的开发,支持 Windows、Linux/X11、Mac OS X 等操作系统。Qt 官方网页: <http://qt.nokia.com/>, 如图 1-1 所示。

Qt 采用 C++ 语言,包含了丰富的 C++ 类,包括窗口界面设计的接口、IO 控制接口、绘图接口、多媒体接口、数据库操作接口、网络通信接口、XML 接口、模块测试接口等丰富的开发接口。软件开发人员通过使用这些接口,可以方便、高效地完成应用设计与程序开发。由于采用 C++ 语言,Qt 具有较高的执行效率。此外不同平台间的 Qt 开发接口是相同的,因此,可以有效地降低 Qt 应用程序跨平台开发的移植成本。

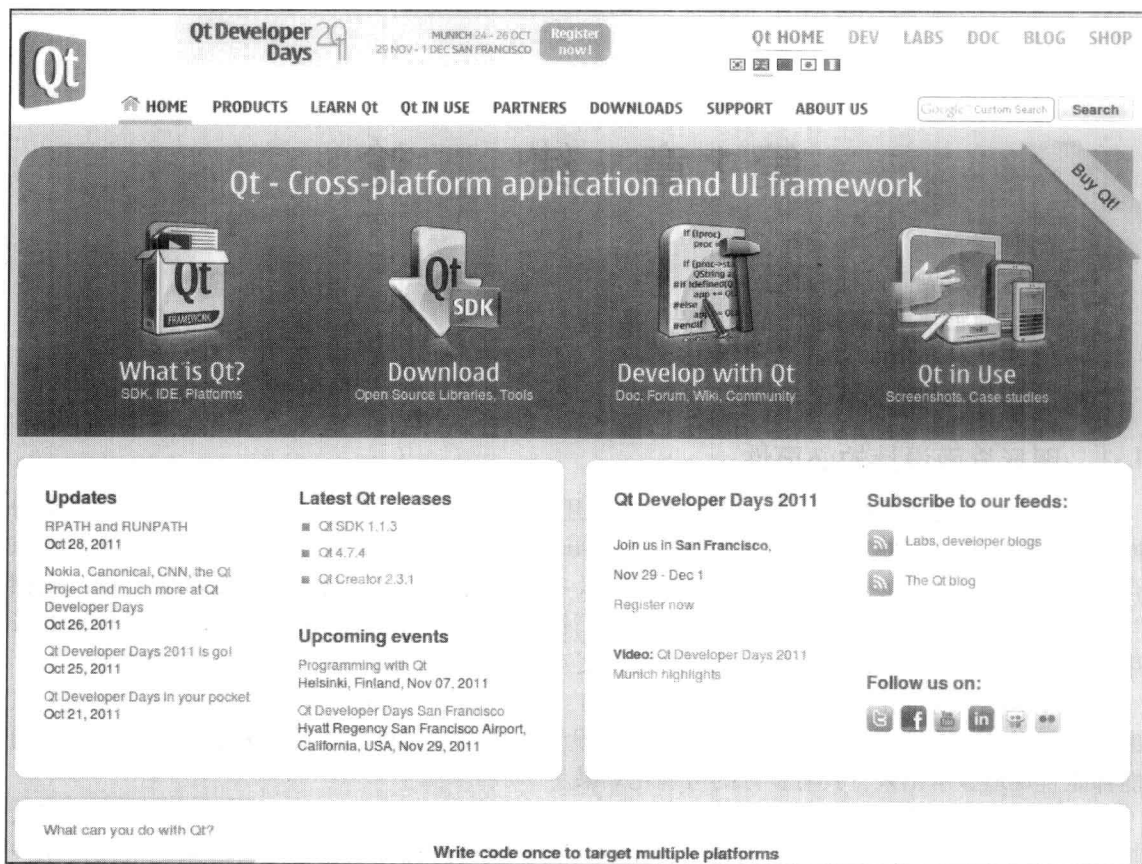


图 1-1 Qt 官方网页

Qt 不仅仅是一个图形界面开发类库，而是一套拥有相对完整的开发环境的开发工具。Qt 提供了丰富的开发工具，用来提高应用设计开发人员的编程效率。这些工具主要包括：界面设计工具 Qt Designer、工程管理工具 qmake 等。在最新的 Qt4.5 及以下的版本中还加入了 Qt 程序开发的 IDE 环境 Qt Creator，以及与其他开发工具的扩展插件，可以支持 Visual Studio 和 Eclipse 等常用开发工具。本书的实例都是基于使用 Qt4.7 的 Creator 集成开发环境而编写的。

NOKIA 公司收购 TrollTech 公司后，推出了两个重要版本：

Qt 商业版：主要面向商业软件开发。这些软件用于商业用途时，需要支付一定的费用，并在有效的时段内得到一定的技术支持。

Qt 开源版：源码开放，遵循 GNU 通用公共许可证（General Public License, GPL），可以进行自由开发。

1.2.2 GTK +

GTK + 来源于 GNU Image Manipulation Program (GIMP)（这是一个 GNU 开放源码项目），由 Peter Mattis 和 Spencer Kimball 共同开发。

确切地说，GTK + 是一个图形用户界面开发工具库。利用这个工具库可以创建基于图形用

户界面的应用程序，也就是说，GTK+集成程序员编程时需要用的控件库，利用这些库可以构建用户所需要的图形界面。GTK+是从GTK中分离出来的，“+”代表了GTK+具有很好的扩展性和支持面向对象技术。关于GTK+的详细内容，感兴趣的读者可以参阅GTK+的官方网站：<http://www.gtk.org/>，如图1-2所示，本书不作过多的讨论。



图 1-2 GTK+ 官方网站

1.3 两种常用的 Linux 数据库

1.3.1 MySQL

MySQL 由瑞典的 MySQL AB 公司开发，是一个中小型数据库管理系统。在 2008 年被 Sun 公司收购，成为 Sun 旗下的产品。2009 年，Sun 又被 Oracle 公司并购。目前，MySQL 是 Oracle 公司的产品之一，其 LOGO 如图 1-3 所示。

MySQL 是一种关系数据库管理系统，它将数据保存在一个个数据表中，大大地提高了数据操作效率。MySQL 支持 SQL 语言访问数据库系统，操作简便。

MySQL 源代码遵循 GNU 通用公共许可证。源代码是开放的，利于设计者根据不同的应用设计专用的解决方案。MySQL 广泛地应用于 Internet 上的中小型网站和嵌入式数据库的开发。

1.3.2 SQLite

SQLite 代码量特别小，大约 3 万行，是一个轻量级的小型数据库系统，占用资源少，处理速度快，最初是为嵌入式产品而设计的。其 LOGO 如图 1-4 所示。



图 1-3 MySQL 的 LOGO



图 1-4 SQLite 的 LOGO

SQLite 能够在多个平台上运行，且使用方便，无须安装和管理配置。能够支持多种编程语言直接创建数据库，灵活自由。2000 年 5 月，SQLite 第一个版本诞生，从此其每一个版本几乎都应用在嵌入式产品开发中。经过近些年嵌入式开发者和自由软件爱好者的完善与改进，SQLite 已经形成了功能相对齐全的嵌入式数据库系统，并在嵌入式产品中得以广泛应用。

1.4 图形界面与数据库综合编程技术

在当前的软件产品和嵌入式开发中，单一的图形界面开发和单一的数据库开发的案例已经很少了，一般都是图形界面配合数据库编程，以获得功能完善的产品设计方案。

典型的软件产品开发分为两个部分：数据库管理员功能模块和用户应用程序模块。

数据库管理员功能模块如图 1-5 所示，数据库中的数据由数据库管理工具通过数据库服务器进程来维护，数据库管理员可以利用数据库管理工具对数据库进行维护操作。

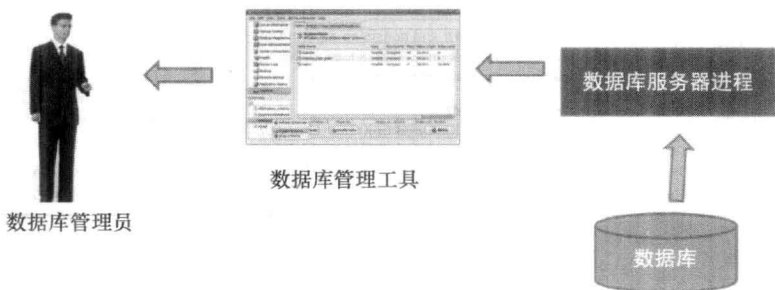


图 1-5 数据库管理员功能模块

用户应用程序功能模块如图 1-6 所示，应用程序是基于图形界面开发工具类库开发的，数据库驱动程序将数据库服务器进程与开发工具类库联系在一起，使它们建立一定的通信机制，数据库服务器进程负责对数据库中的数据进行直接操作。因此，用户可以通过应用程序对数据库进行访问操作。



图 1-6 用户应用程序模块

第 2 章 Qt4 图形界面编程基础

本章目的:

- Qt Creator 的简单介绍
- Qt Creator 的安装
- Qt Creator 的信号槽机制
- Qt Creator 的简单使用

2.1 Qt Creator

Qt Creator 是一个基于 Qt 的进行用户界面设计的可视化集成开发环境,可以跨平台运行,支持 Linux、Mac OS、Windows 等操作系统。Qt Creator 集成了 Qt Designer 的所有特性,方便用户进行界面设计。另外还集成了 Qt Assistant、Qt Linguist、图形化的 GDB 调试前端、qmake 构建工具等。Qt Creator 还包括项目生成向导、高级的 C++ 代码编辑器、浏览文件及类的工具。Qt Creator 使开发人员能够更加便捷地完成开发任务,Qt Creator 界面如图 2-1 所示。



图 2-1 Qt Creator 界面

2.2 安装 Qt 环境及 Qt Creator

本书的所有实例都是在 Ubuntu 环境下进行的,这里只介绍在 Ubuntu 下搭建环境的具体步骤,其他 Linux 系统环境搭建与此相类似。

可以从 Qt 官方网站上直接下载安装 Qt Creator,网址为: <http://qt.nokia.com/downloads>,打开后的网页如图 2-2 所示,我们选择下载 Qt SDK 安装包,其中包含用于 Windows、Linux/X11

32-bit、Linux/X11 64-bit 和 Mac OS 的安装包，笔者系统为 Linux/X11 64-bit，所以选择 Linux/X11 64-bit 的 SDK 安装包。Linux/X11 64-bit 的 SDK 安装包分为在线安装包和本地安装包，根据网速选择，网速较快可选择在线安装。

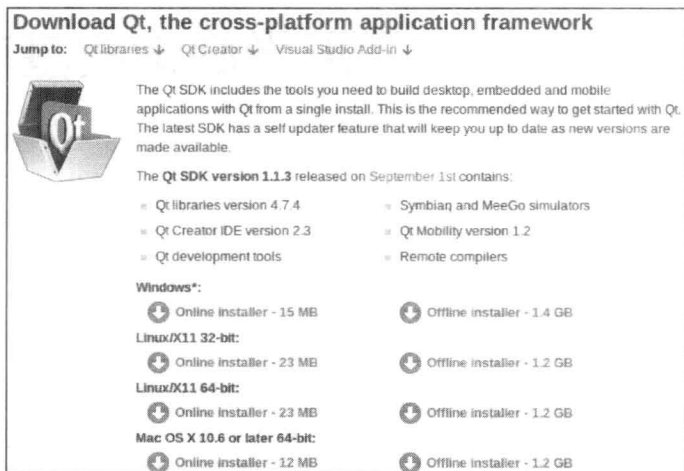


图 2-2 Qt Creator 下载选择

下载完毕后，直接在终端运行安装包，出现图 2-3 所示的界面，单击“Next”按钮，出现图 2-4 所示界面。在图 2-4 所示窗口中设置安装路径，此处使用默认的安装路径，安装类型有两种，可以是默认的安装类型，也可以是用户自定义的安装类型，默认的安装类型安装的东西比较多，占用的空间大，建议初学者选择“Default”安装类型，此处笔者选择“Customer”安装类型，单击“Next”按钮，出现图 2-5 所示的界面。图 2-5 中是一些安装选项，此处将与“Symbian”相关的选项都去掉，单击“Next”按钮，出现图 2-6 所示的界面。图 2-6 是安装协议，选择接受安装协议选项，单击“Next”按钮，出现图 2-7 所示的界面，进行 Qt Creator 的安装。安装完成之后单击“Next”按钮，出现图 2-8 所示的界面，单击“Finish”按钮成功安装 Qt Creator。此时在系统的“应用程序”选项中选择“编程”，可以看到 Qt Creator 选项，说明已成功安装 Qt Creator（见图 2-9）。下面将通过一个简单的实例说明 Qt Creator 的使用方法。

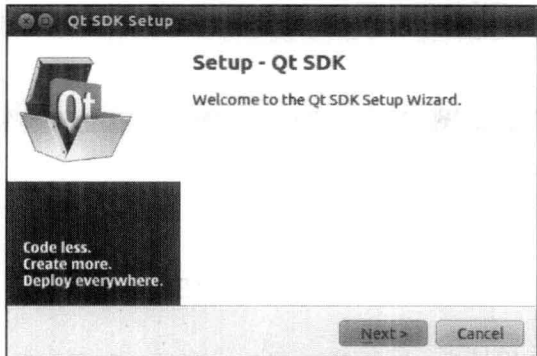


图 2-3 Qt Creator 安装第一步



图 2-4 Qt Creator 安装路径

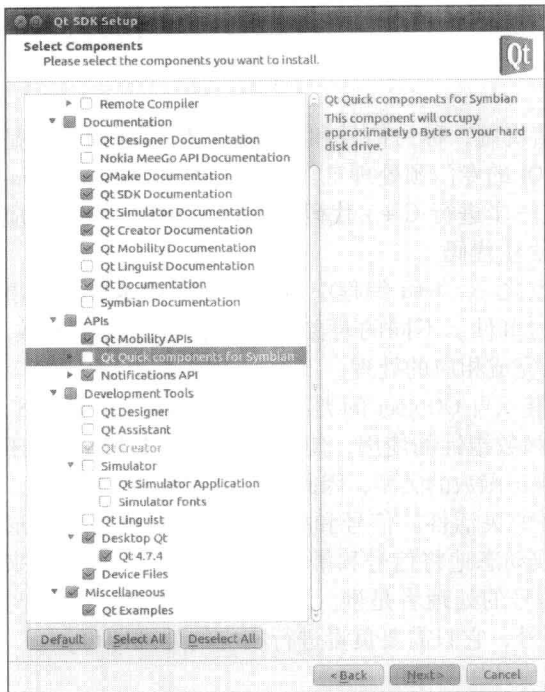


图 2-5 安装组件选择

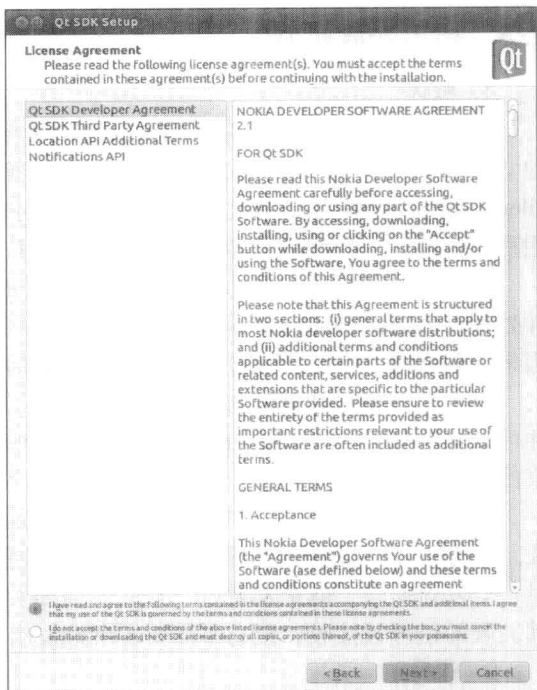


图 2-6 接受安装协议

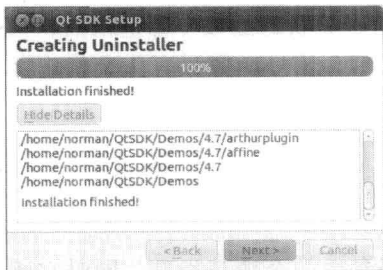


图 2-7 安装完成

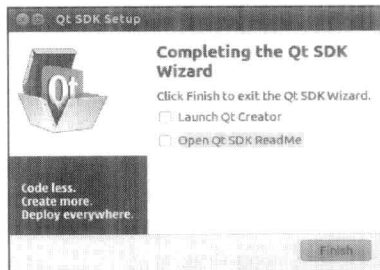


图 2-8 安装成功

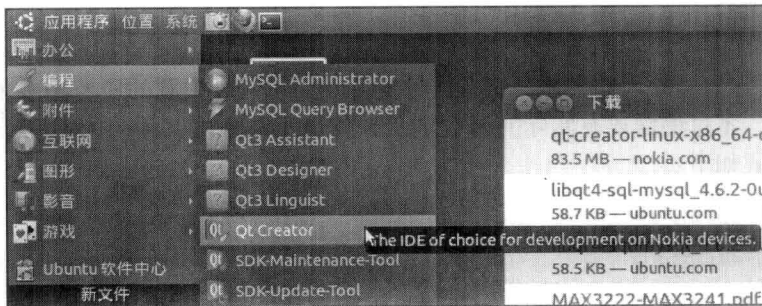


图 2-9 安装成功之后应用程序增加的选项

2.3 信号和槽机制

1. 概述

Qt 采用了一种全新的对象和方法的关联与通信机制，称为信号和槽机制。信号和槽机制是独立于标准 C++ 编译器的，在编译之前需要经过 Qt 的专门预处理工具 MOC (Meta Object Compiler, 元组件编译器) 对代码进行预处理后才能进一步进行 C++ 代码的编译。MOC 会将 Qt 应用程序中特有的代码自动转化为相应的标准 C++ 语法代码。

信号和槽的概念是 Qt 编程中最具代表性的特点之一。GUI 编程中通常使用回调函数进行事件处理，而回调函数通常是一个函数指针，不同的事件、不同的对象都有着各自的回调函数。当事件到来时，系统会通过调用对应的回调函数来完成相应的处理。

Qt 引入了信号和槽机制来取代回调函数。凡是继承自 QObject 的类都可以具有信号和槽成员，并可以使用它们。信号和槽的使用可以有效地减少函数指针的使用，使程序代码清晰简洁，对于事件响应管理更加容易。此外，信号和槽没有严格规定函数的类型，因此在调用过程中是安全的。

信号和槽的使用可以实现信息封装，增加程序的灵活性。信号和槽都采用函数作为存在形式。在 Qt 程序初始化或运行的过程中，可以静态或动态地将信号和槽相关联。当某一事件到来时，会发射信号，但是发射后，它并不需要关心信号的处理者是谁。当触发槽函数时，说明与其关联的信号被接收，但它不需要了解谁发出了信号，它只需要负责进行相应的处理即可。

2. 信号

信号是 Qt 中对事件的一种抽象，当一个事件到来时，会发射信号。所谓的发射信号，就是通过 Qt 中特有的通信机制，调用和信号相关的各个槽函数。因此当发射信号时，将执行与其相关联的槽函数。信号采用函数的形式，因此，当所有与信号关联的函数全部返回后，信号函数才会返回。信号函数在形式上与标准 C++ 中的虚函数类似，信号函数只有头文件中函数的声明，而没有函数的定义（即函数体）。信号的声明形式如下：

```
signal:
    void MySignal();
    void MySignal(int x);
    void MySignalParam(int x, int y);
```

对于信号的声明与普通的 C++ 函数无异，它不限制参数的个数与类型，同时它还支持重载。不过信号的返回值只要求必须为 void 型，因为事件是一种中断，对于突发性的中断，不可能期望其有返回值。

3. 槽

槽是 Qt 中负责信号处理的实体，当有信号发射时，与信号关联的所有槽会依次执行。槽也采用函数的形式，不过槽需要有实际的函数定义，相当于在标准 C++ 中对虚函数的多态实现。槽函数的声明形式如下：

```
public slot:
    void MySlot();
    void MySlot(int x);
    void MySlotParam(int x, int y);
```

与信号相同，槽函数的定义同普通的 C++ 函数无异，支持 C++ 函数的一些特性。同时，槽函数可以采用标准 C++ 函数的使用方式，在代码中直接调用。槽函数具有访问权限的标识，