

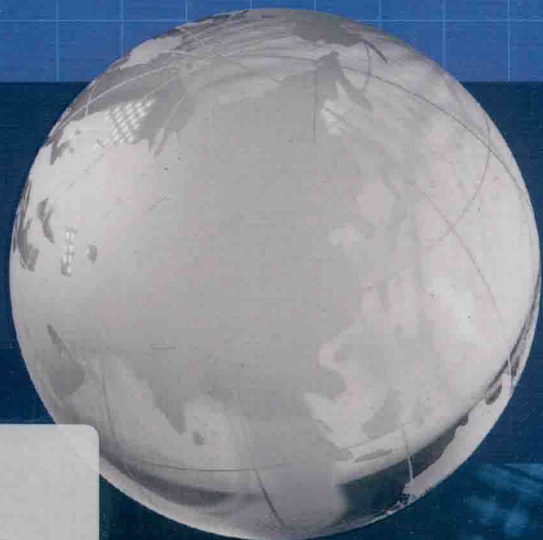


IT新技术丛书

Qt编程 快速入门

北京跟踪与通信技术研究所

鲍忠贵 王涛 陈凌晖 编著



清华大学出版社

IT新技术丛书

Qt编程 快速入门

鲍忠贵 王涛 陈凌晖 编著



清华大学出版社
北京

内 容 简 介

Qt 是一种跨操作系统平台的 C++ 语言开发工具,集成了丰富的界面设计窗口部件和系统功能模块,它具有“一次编写,随处编译运行”的特点,为广大软件开发者所喜爱。

本书采用循序渐进、由浅入深的编撰方法,通过大量的示例介绍了 Qt 的窗口、信号和槽、键盘和鼠标、窗口部件、对话框、文件与打印、多文档窗口和切分窗口、2D 与 3D 绘图、进程和线程、网络、数据库和简单网络管理 SNMP 等内容的软件开发设计技术。

本书适合具有 C 或 C++ 语言基础的软件开发人员、中高等学校师生和工程技术人员使用。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

Qt 编程快速入门/鲍忠贵,王涛,陈凌晖编著. --北京:清华大学出版社,2016
(IT 新技术丛书)

ISBN 978-7-302-41649-4

I. ①Q… II. ①鲍… ②王… ③陈… III. ①C 语言—程序设计 IV. ①TP312

中国版本图书馆 CIP 数据核字(2015)第 230662 号

责任编辑:石磊 赵从棉

封面设计:傅瑞学

责任校对:赵丽敏

责任印制:沈露

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座 邮 编:100084

社 总 机:010-62770175 邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质量反馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

印 刷 者:清华大学印刷厂

装 订 者:北京市密云县京文制本装订厂

经 销:全国新华书店

开 本:185mm×230mm 印 张:13.75 字 数:298 千字

版 次:2016 年 6 月第 1 版 印 次:2016 年 6 月第 1 次印刷

印 数:1~2500

定 价:35.00 元

产品编号:065787-01

前 言

随着操作系统日益多样化,采用跨平台的开发工具编程越来越受到程序员的重视,一次开发,随处编译运行,不仅能减少重复性,而且可适应不同用户和操作系统的需求。Qt就是这样一种跨操作系统平台的可视化开发工具,事实上,也日益成为 UNIX、Linux、Windows 甚至国产操作系统中标麒麟等平台上开发 C++ /C 程序的首选。

Qt 从第 4 版后,推出了 Qt Creator 可视化集成开发环境,极大地方便了开发者对界面所见即所得的要求,而且,Qt 不仅是界面开发工具,其内置的软件模块也日益丰富,逐步支持了对系统操作、进程管理、数据库、网络、3D 绘图等功能的开发,基本满足程序员对软件开发集成环境的要求。

Qt 的类库和模块很多,本书不是一本面面俱到的参考书,而是面向具有一定 C/C++ 基础的 Qt 初学者。内容安排循序渐进,由简入难,首先从程序窗口的创建、信号和槽的机制和应用、键盘和鼠标事件的响应与处理、各种窗口部件的实现等基础性的编程讲起,逐步过渡到多文档与切分窗口、2D/3D 绘图、进程与线程的应用、UDP 和 TCP 网络协议的应用、数据库的连接和使用等有一定难度的内容,最后结合第三方软件包,介绍了网管管理站的示例。这些内容都是程序员开发软件经常用到的技术。掌握了这些内容,可以更好地理解并使用 Qt 的开发语言和环境。

从书稿的编排上,开门见山,力求像使用说明书一样,一步一步地讲述,使读者能快速掌握 Qt 的编程语言和开发技术。书中的例子简单实用,读者参考例子稍作修改就可以建立满足自己需求的程序。因编者水平有限,书中定有不足甚至错误之处,欢迎读者批评指正。

编 者

2015 年 4 月于北京

目 录

第 1 章 开始 Qt 的第一个程序	1
1.1 认识 Qt	1
1.2 建立 Qt 的工程项目	3
1.3 了解 Qt 生成的工程文件	6
1.4 了解 Qt 的程序结构	10
第 2 章 窗口	13
2.1 主窗口	13
2.1.1 主窗口的框架	13
2.1.2 主窗口的标题	14
2.1.3 主窗口的图标	16
2.1.4 主窗口的基本操作	18
2.2 菜单	19
2.2.1 设计菜单	19
2.2.2 编写菜单代码	22
2.2.3 弹出式菜单	23
2.3 工具条	24
2.3.1 设计工具条	24
2.3.2 编写工具条代码	25
2.3.3 在工具条添加部件	25
2.3.4 工具条窗口	27
2.4 状态条	29
2.4.1 在状态条显示信息	29
2.4.2 在状态条添加部件	29
2.5 系统拖盘图标	30

第 3 章 信号和槽	33
3.1 基本概念	33
3.2 在 Qt Designer 中建立信号和槽	34
3.3 自定义信号和槽	36
第 4 章 键盘和鼠标的操作	41
4.1 键盘和鼠标的类	41
4.2 键盘事件的响应	41
4.3 鼠标事件的响应	43
第 5 章 窗口部件	44
5.1 按钮	44
5.1.1 按钮的种类	44
5.1.2 按钮的设置	45
5.1.3 按钮的编码实现	47
5.2 输入窗口部件	48
5.2.1 设计输入窗口	48
5.2.2 值的设置	50
5.2.3 值的读取	52
5.2.4 值的输入检验	52
5.3 显示窗口部件	55
5.3.1 设计显示窗口	55
5.3.2 值的设置和读取	56
5.4 浏览器	56
5.4.1 文本浏览器	56
5.4.2 图像浏览器	58
5.4.3 Web 浏览器	59
5.5 基本布局	61
5.5.1 布局的设计	62
5.5.2 布局的编码实现	63
5.6 项的显示部件	63
5.6.1 设计项显示部件	64
5.6.2 编写项的代码	65
5.6.3 在项中加入窗口部件	68
5.6.4 在项中加入弹出菜单	68

5.7	列项的显示视图	69
5.7.1	文件系统模型	70
5.7.2	字符串列表模型	71
5.7.3	标准项模型	72
5.7.4	委托的建立	73
5.8	容器部件	78
5.9	自定义窗口部件	81
第6章	对话框	86
6.1	使用内置标准对话框	86
6.1.1	颜色对话框	86
6.1.2	文件对话框	87
6.1.3	字体对话框	88
6.1.4	输入对话框	89
6.1.5	消息对话框	90
6.1.6	页面设置对话框	91
6.1.7	打印对话框	92
6.1.8	打印预览对话框	92
6.1.9	进度显示对话框	94
6.2	内置标准对话框中的文字替换	94
6.3	建立自己的对话框	96
第7章	文件与打印	100
7.1	文件	100
7.1.1	读取格式化文本	100
7.1.2	保存文本文件	101
7.1.3	读取二进制文件	102
7.1.4	使用结构	104
7.1.5	读取XML文件	105
7.2	打印	110
7.2.1	打印文本文件	110
7.2.2	程序打印控制	112
7.2.3	打印图像	113

第 8 章 多文档界面与切分窗口	115
8.1 多文档界面	115
8.2 切分窗口	120
第 9 章 绘图	123
9.1 2D 绘图	123
9.1.1 绘图的程序框架	123
9.1.2 绘制图形	123
9.1.3 在内存中绘制图形	127
9.2 3D 绘图	129
9.2.1 从 OpenGL 到 Qt	129
9.2.2 使用 QGLWidget	130
9.2.3 使用贴图	134
第 10 章 进程与线程	136
10.1 进程	136
10.1.1 进程的启动	136
10.1.2 进程的关闭	137
10.1.3 与进程通信	137
10.2 线程	139
10.2.1 线程的创建与启动	139
10.2.2 线程间的同步操作	141
10.2.3 线程综合例子	143
10.3 自定义函数库	147
10.3.1 动态链接库	147
10.3.2 静态链接库	151
10.4 共享内存	152
第 11 章 网络	154
11.1 获取网络信息	154
11.1.1 获取主机 IP 地址	154
11.1.2 获取主机 IP 地址和子网掩码	156
11.1.3 获取主机 MAC 地址	157
11.2 UDP 通信	158
11.2.1 基于主窗口的实现	159


11.2.2 基于线程的实现	162
11.3 TCP 通信	166
第 12 章 数据库	172
12.1 Qt 数据库基础	172
12.1.1 Qt 数据库的类库结构	172
12.1.2 Hello QtDatabase	173
12.2 连接数据库	175
12.3 数据库操作	177
12.3.1 执行 SQL 语句	177
12.3.2 使用占位符	177
12.3.3 数据检索	178
12.3.4 读取字段名	179
12.3.5 自定义数据模型	179
12.4 SQL 数据模型	180
12.4.1 使用 QSqlTableModel	181
12.4.2 使用 QSqlRelationalTableModel	183
第 13 章 简单网络管理 SNMP	185
13.1 简单网络管理协议	185
13.1.1 基本概念	185
13.1.2 使用 Net-SNMP 软件包	187
13.2 Qt 开发管理站	187
第 14 章 字符串、列表与时间类	199
14.1 QString 类	199
14.1.1 创建字符串	199
14.1.2 格式化输出字符串	200
14.1.3 类型转换	201
14.1.4 字符串操作	202
14.2 QList 类	203
14.2.1 定义类型列表	203
14.2.2 列表的操作	204
14.3 日期、时间和定时器类	204
14.3.1 QDate 类	204

14.3.2	QTime 类	205
14.3.3	QDateTime 类	206
14.3.4	QTimer 类	206
参考文献		208

开始 Qt 的第一个程序


1.1 认识 Qt


Qt 是 1991 年奇趣科技开发的一个跨平台的 C++ 图形用户界面应用程序框架。随着不断扩充和发展,Qt 集成了更多系统级的功能组件,现在的 Qt 不仅用于开发用户界面,也可以开发控制台工具和服务器程序。2008 年 Qt 被 Nokia 公司收购,2012 年 8 月 Qt 软件业务出售给芬兰 IT 服务公司 Digia。虽然经历了这些变化,Qt 还是以它强大易用的功能获得了技术和商业上的成功。Qt 是自由且开放源代码的软件,支持广泛的编译器,包括 GCC 的 C++ 编译器和 Visual Studio。用户可以直接从 Qt 官方网站(<http://qt-project.org/downloads>)下载安装包或源代码包,在 Windows 或 Linux 系统上可以直接运行安装,有的 Linux 系统还将 Qt 打包在系统安装包中,在安装时选择添加开发工具即可,非常方便。


像其他可视化软件开发工具一样,Qt4 以后提供了 Qt Creator 集成开发环境,我们可以在集成开发环境中设计程序界面、编辑代码、调试编译构建可执行程序等。在程序菜单中,选择  选项,启动 Qt Creator,出现 Qt 开发环境的主窗口,如图 1-1 所示。本书以 Qt4.7.x 的集成开发环境为例介绍 Qt 的开发技术,其他版本如 Qt4.8.x 和 Qt5.2.x 的界面类似,读者可以进行对应的操作。

在 Qt Creator 界面中,顶层是菜单栏,包括文件、编辑、建立、调试、工具集、窗口和帮助操作菜单。

左上角是模式选择区,以下进行介绍。

: 欢迎界面模式,包括 Qt 的学习资料、参考例子,新建或打开 Qt 工程文件、最近的开发项目等,可以快速地进入工作界面。

: 编辑模式,从其他模式或界面快速切换到编辑状态。

: 设计模式,启动 Qt Designer,在该模式下通过所见即所得的可视化显示,快速设计图形用户界面,如图 1-2 所示。

设计界面中,左边是设计部件面板,包括布局、按钮、容器、输入输出等部件;界面中间是主窗口样式;右上角是样式中的对象面板;右下角是对象的属性面板。

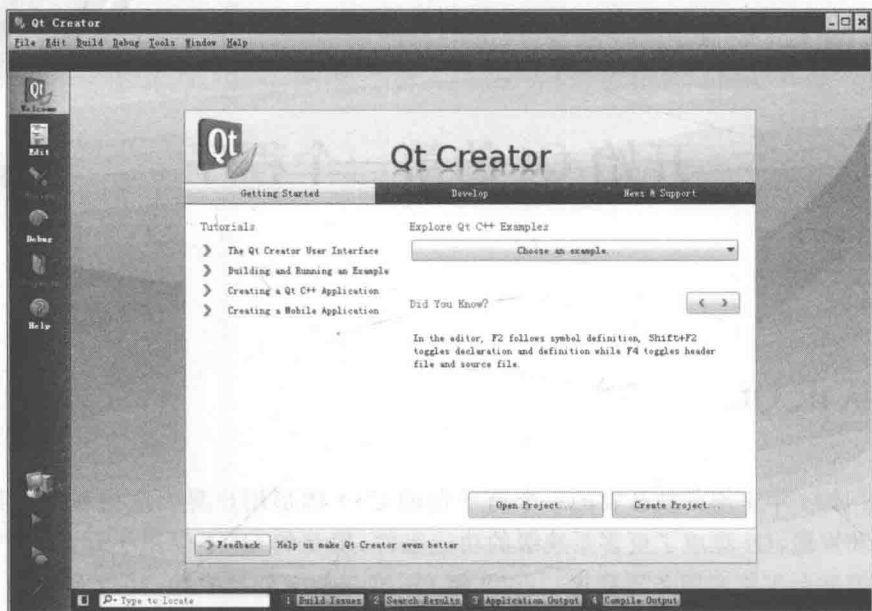


图 1-1 Qt 开发环境的主窗口

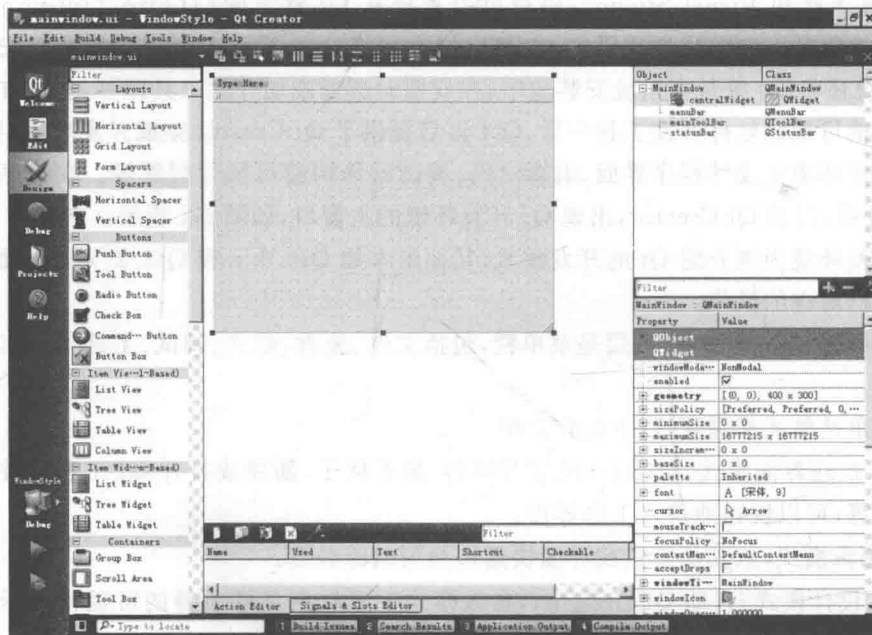


图 1-2 Qt 开发环境的界面设计窗口

: 调试模式, 在该界面下提供了调试面板, 如图 1-3 所示。

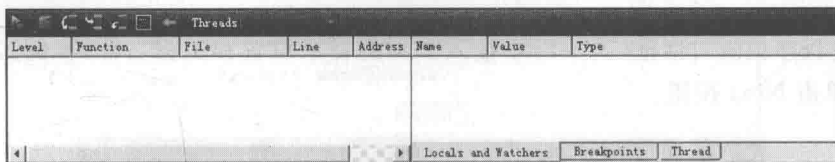



图 1-3 程序调试面板

: 工程模式, 提供了快速设置建立(编译)、运行和编辑等工作环境。

: 帮助模式, 快速检索和查看在线帮助系统。

在左下角是建立运行区, 分别是目标系统选择、运行、调试和建立快捷选项。

在 Qt Creator 界面中, 最下层是定位和输出面板快捷选项。

本节暂不对 Qt Creator 界面中的细节进行展开介绍, 我们将在后续介绍程序开发的过程中逐步了解 Qt 的使用。

1.2 建立 Qt 的工程项目

像所有的集成开发工具一样, Qt 集成开发环境需要建立工程文件, 把生成软件所需的所有文件通过工程文件统一管理。

新建工程文件可以从文件菜单中选择新建文件或工程(New File or Project)菜单项, 或从欢迎界面中选择创建工程(Create Project)开始, 选择后出现“新建”对话框, 见图 1-4。

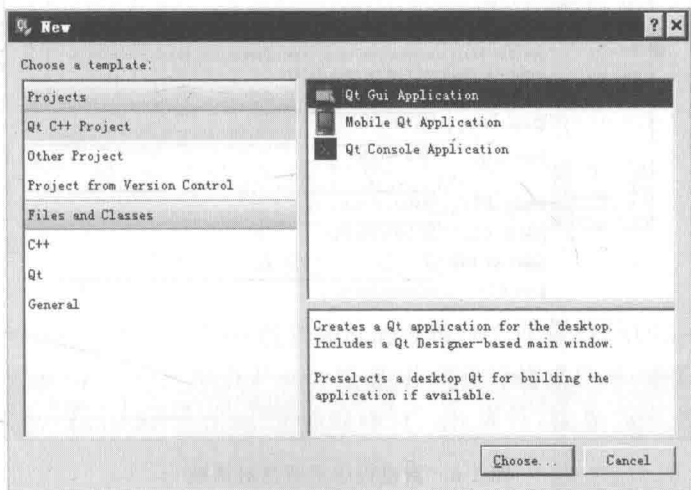


图 1-4 “新建”对话框

依次选择 Qt C++ Project 和 Qt Gui Application 选项,单击 Choose 按钮进入下一步操作。

在新建程序位置对话框(图 1-5)中输入工程的目录位置和工程名称,本例工程名称为 Ch1,然后单击 Next 按钮。

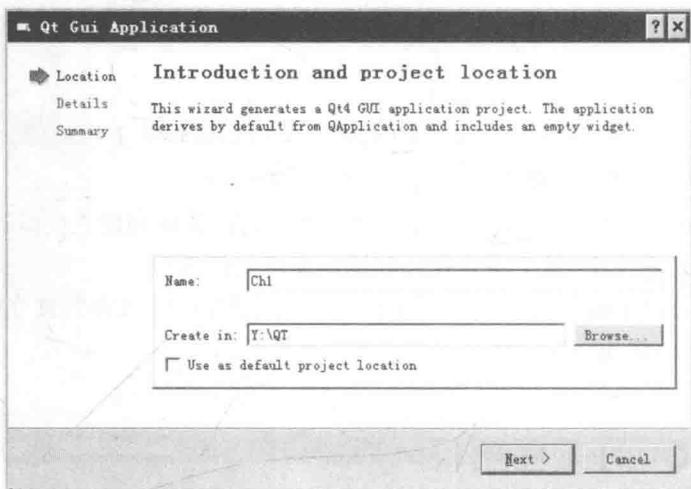


图 1-5 新建程序位置对话框

在新建程序类信息对话框(图 1-6)中填写类的信息,包括主窗口的类名、基类、头文件、源文件和主窗口界面文件等,单击 Next 按钮。

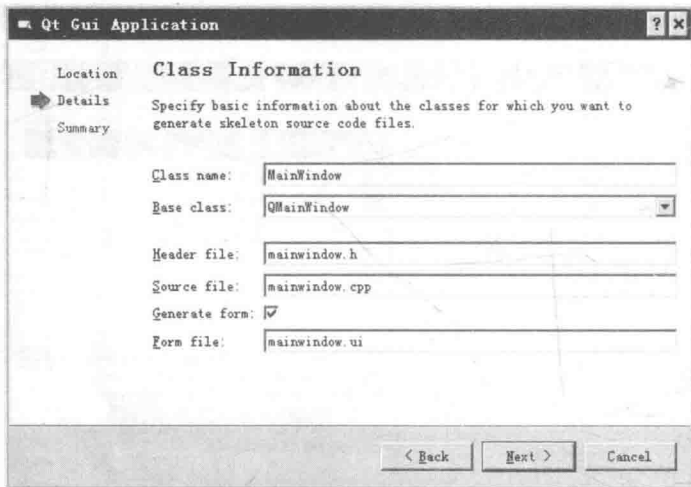


图 1-6 新建程序类信息对话框

在新建程序完成信息对话框(图 1-7)中单击 Finish 按钮,在下一步设置生成目标的目

录,如图 1-8 所示。

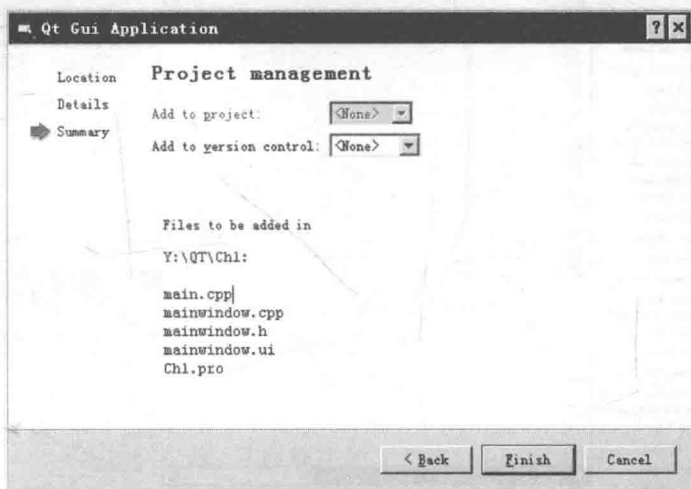


图 1-7 新建程序完成信息对话框

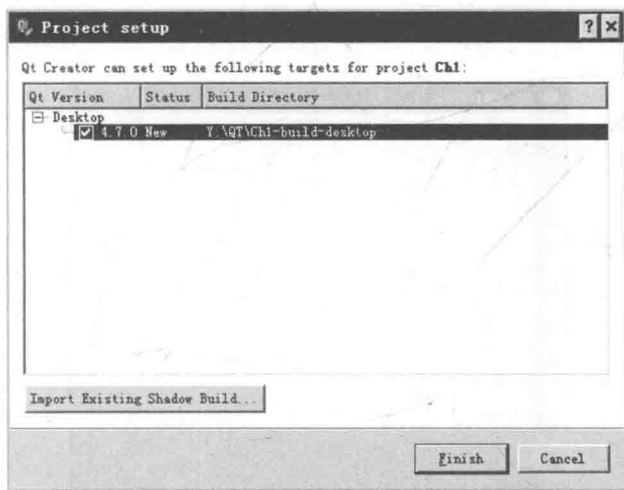


图 1-8 新建程序目标目录设置对话框

这样,源文件在 QT\Ch1 目录中,建立完成的可执行文件在 QT\Ch1-build-desktop 目录的 Debug 和 Release 目录中。单击 Finish 按钮,出现设计模式界面,如图 1-9 所示。

单击左下方运行按钮,Qt Creator 进行编译链接和运行,生成一个空框架的窗口,如图 1-10 所示。

至此,我们用 Qt 创建了第一个应用程序。这个程序用户不用编写一行代码,完全是系统默认设置的。下一节,我们将了解 Qt 生成的工程文件。

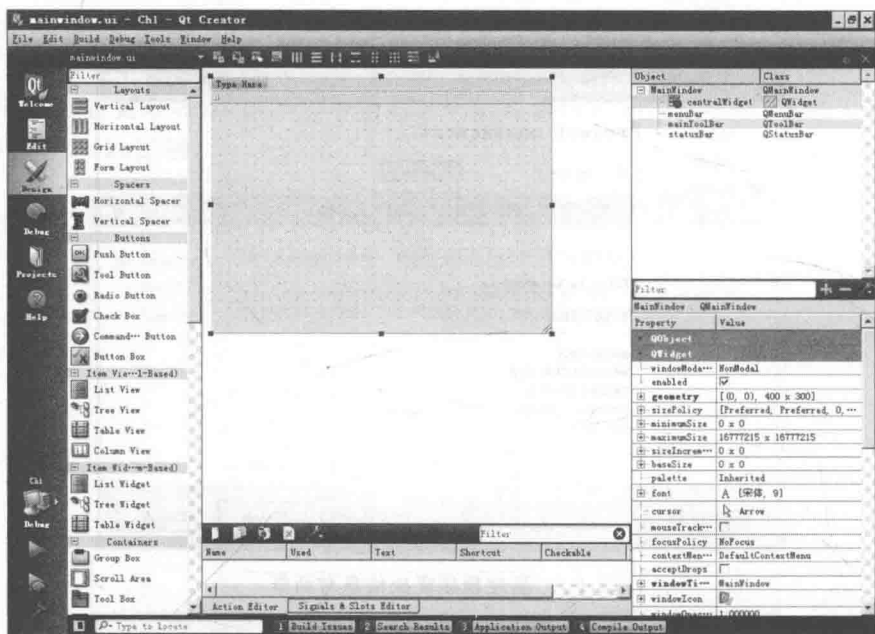


图 1-9 用户界面设计窗口

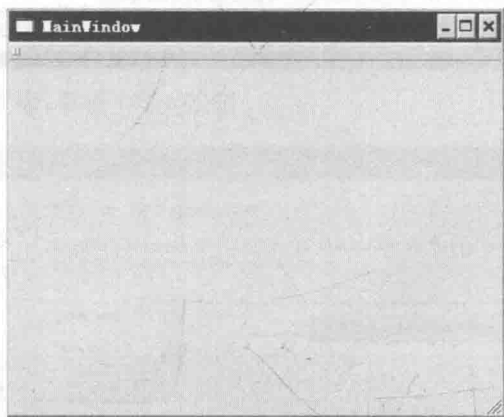


图 1-10 第一个应用程序运行界面

1.3 了解 Qt 生成的工程文件

Qt 生成的工程文件包括工程文件、样式文件、头文件和源文件，在 Qt Creator 集成环境编辑模式中可以分类看到，双击相应的文件，就会在编辑窗口中打开，如图 1-11 所示。这些

文件存放在 QT\Ch1 目录中。



图 1-11 编辑窗口快捷面板

1. 工程文件: *.pro

工程文件定义了 Qt 用的类库、目标程序的文件名,采用模板、C++ 源文件、C++ 头文件和样式文件,如下所示:

```
#-----  
#  
# Project created by QtCreator 2013-11-27T15:34:33  
#  
#-----  
QT      +=core gui  
TARGET=Ch1  
TEMPLATE=app  
SOURCES+=main.cpp\  
        mainwindow.cpp  
HEADERS +=mainwindow.h  
FORMS   +=mainwindow.ui
```

2. 样式文件: *.ui

样式文件定义了人机交互界面,在设计模式下是所见即所得的窗口和控件,在编辑模式下是 XML 定义文件,如下所示:

```
<?xml version="1.0" encoding="UTF-8"?>  
<ui version="4.0">  
<class>MainWindow</class>  
<widget class="QMainWindow" name="MainWindow">  
<property name="geometry">  
<rect>  
<x>0</x>
```