

基于 Qt/Embedded 的电子词典的设计与实现

田莺¹, 杨中华²

(1. 鞍山师范学院 教育科学与技术学院, 辽宁 鞍山 114007;

2. 航天一院 15 所, 北京 100076)

摘要: 随着嵌入式技术的不断进步以及智能终端如 PDA、手机、车载设备市场需求的日新月异, 人们对手持设备的应用功能多样化以及 GUI 的美观提出了更高的要求。为了增加智能终端的应用功能以适应市场需求, 设计并实现了一款具有英语词汇查询、学习记忆、用户管理和帮助等功能的电子词典。

关键词: 嵌入式 Linux; Qt; Qt/Embedded; Qtopia; 信号 / 槽

中图分类号: TP311 **文献标识码:** A

Design and realization of electronic dictionary based on Qt/Embedded

TIAN Ying¹, YANG Zhong Hua²

(1.College of Education Science and Technology, Anshan Normal University, Anshan 114007,China;

2. No.15 Institute, First Academy of Aerospace, Beijing 100076,China)

Abstract: With the rapid progress of intelligent terminals such as PDA, cell phone, on-board equipment, as a significant branch of consumer electronic products, more and more requirements of the function and delight GUI appearance are needed. The paper designed an intelligent application of the terminal to meet market, and operated an english vocabulary with a query function, learning and memory function, users' management functions on it.

Key words: embedded Linux; Qt; Qt/Embedded; Qtopia ; signal/slot

1 嵌入式 GUI 的发展概况

普通 Linux 一般提供基于 X-Window 的 KDE/GNOME 作为图形用户界面, 但对于嵌入式系统来说, 这些都太庞大臃肿。因此, 一批基于嵌入式 Linux 的 GUI 系统应运而生, 目前发展比较成熟的主要包括以下几种:

(1)MiniGUI^[1]

在 Linux 控制台上运行, 是基于 SVGALib 和 Linux Thread 库的多窗口图形用户界面支持系统。MiniGUI 采用了类 Win32 的 API 接口, 实现了简化的类 Windows 98 风格的图形用户界面。但是 API 没有封装, 不便移植;

(2)Open GUI^[2]

采用 LGPL 条款发布, 比较适合基于 X86 平台的实时系统。基于汇编实现内核, 利用 MMX 指令提高运行速度, 但可移植性稍差;

(3)Micro Windows/Nano X^[3]

一个采用 MPL 条款的开放源码的项目, 其主要特色在于提供了类似 X 的客户 / 服务器体系结构以及相对完

善的图形功能。但它的图形引擎存在一些问题, 无任何硬件加速能力、算法低效和代码质量较差;

(4)Qt/Embedded

跨平台 C++ 图形用户界面工具包。因其面向对象、跨平台、界面设计更美观而得到广泛的应用。由于 KDE 等项目使用 Qt 作为支持库, 所以由许多基于 Qt 的 X-Windows 程序可以非常方便地移植到 Qt/Embedded 版本上。Qtopia 是由 TrollTech 公司基于 Qt/Embedded 开发的第一个嵌入式窗口环境和应用程序, 广泛应用于 PDA、掌上设备和网络设施等。

2 电子词典的开发环境

Qt/Embedded 开放了源代码, 使开发人员可以在 GPL 许可协议下自由地使用, 以开发嵌入式 Linux 应用系统。本设计使用了 TrollTech 公司的免费版: Qt /Embedded 2.3.7 版本、Qtopia 1.7.0 版本和 Qt /X112.3.2 版本。由于程序要运行在三星 S3C2410 的微处理器上, 所以在 Linux/X86 上经过正确设置环境变量后要要进行如下配置和编译: ./configure

xplatform linux-arm-g++ -no-xft no-qvfb depths 4,8,16,32, 编译后将得到如下创建应用程序所需的工具包:

(1)designer 是设计窗口组件的应用程序。使用它将最后生成 file.ui 文件和 main.cpp 文件;

(2)uic 是从 XML 文件生成代码的用户界面编译器。其主要作用是将 file.ui 文件生成 file.h 和 file.cpp 文件。

(3)qmake 工具用来生成 file.pro 文件; tmake 是跨平台 Makefile 生成器。

利用这些工具如编译器就可以进行电子词典的开发、编译和调试。

3 电子词典的设计与实现

对于应用软件设计工具 Qt, 在设计过程中有 2 种常用的设计方法。一种方法采用纯粹的面向对象的语言实现^[4], 这种方法要求完全编写 main 文件、头文件以及应用文件, 相对于 Qt 来说, 就是要实现 *.cpp 和 *.h 文件; 另一种设计方法采用图形化界面设计与源代码编写相结合的方式, 可以用 Qt designer(Qt 设计器)设计要求的界面, 这 2 种方法都有其优缺点, 比较如下:

(1)面向对象的语言设计方法使用方便, 不受 Qt designer 的限制, 而使用程序中包含的库或文件, 对于信号/槽的实现也比较容易。最大特点就是可以利用更多资源根据自己的需要设计^[5]。另一方面, 由于这种方法要求编写所有的源程序, 特别对于一些库中已存在的组件的使用需要用编程语言设置其属性, 使工作量增加, 而且由于只有在整个程序完全运行后才可以看到界面, 从而影响了工作中的交互性和开发的速度;

(2)图形化界面设计方法可以根据需要通过改变组件属性来实现。界面中涉及到的信号/槽也同样利用信号/槽属性窗口来设置。当界面设计好之后, 就会自动产生不完整的 main 文件和 *.cpp 源文件, 需要做的就是编写源文件中缺少部分的内容, 每个应用程序名称在信号槽设置中已定义。这种方法的缺点就是受到 Qt designer 的限制, 源程序与图形界面的改变相互影响。

本文研究的电子词典采用了 2 种方法相结合的手段。实际开发经验表明, 善用 Qt designer 与纯编码相结合可大大提高开发速度。

3.1 界面设计与实现

GUI 界面的编码是件相当烦琐的事情, 但 Qt/Embedded 提供了很好的工具, 其功能虽然比不上 Ms Windows 下的界面开发工具, 但也有其自己的特色。Qt Designer 是 Qt 系统专用的界面开发工具, 它不包含任何编译器, 而仅仅是提供一个可视化界面编辑器。Qt Designer 将编辑完成的界面通过 XML 保存为 .ui 文件, 然后由专用的 uic 界面编译器将其转换为标准 c++ 的源文件^[6]。在 Qt Designer 中, 可以通过拖拉或点击的方式, 在一张空白

表单的适当位置上添加一些输入框和按钮等窗口组件。这时 Designer 工具会自动编写和维护代码。

使用 Qt Designer 进行 C++ 程序编制的基本步骤是:

(1)建立窗体, 并根据应用的需要在窗体中添加控件, Qt 会将建立的窗体保存为 .ui 文件; (2)使用 Qt 提供的 uic 工具将 .ui 文件转换为 .h 和 .cpp 文件, 手动添加控制动作; (3)使用 program 工具为该应用程序建立 .pro 工程文件, 并通过 tmake 工具为该工程建立 Makefile 文件; (4)运行 make 即可生成可执行文件。

为了用户操作简洁方便, 在设计电子词典的界面时使用一个窗体设计了 4 个 tab, 每个 tab 是一个功能界面, 其中包括按钮、行编辑框、文本框等。search 编译运行后的截图如图 1 所示。



图 1 search 的界面设计截图

3.2 功能的设计与实现

具体功能的实现需要具备单词词库、Qt/Embedded 开发环境、移植成功的 SQLite。其中的单词词库是以 Excel 表存放。主要实现步骤如下:

- (1)创建数据库文件;
- (2)Qt/Embedded 与 SQLite 连接;
- (3)Qt/Embedded 应用程序的实现。

篇幅所限, 本文将主要介绍 Qt/Embedded 应用程序的实现。

数据库 SQLite 和 Qt/Embedded 连接成功后, 可以通过 Qt/Embedded 的信号与槽机制实现英语词典的基本功能。

在 search 中定义了 3 个成员函数 connect(), 分别是:

```
connect( PbtSearch, SIGNAL( clicked() ), this, SLOT( PbtSearchPress( ) ));
```

```
connect(LineEditInput, SIGNAL( textChanged(const QString& ) ), this, SLOT( lineEditChangeSlot(const QString& ) ));
```

```
connect(ListBoxWords, SIGNAL( highlighted(const QString& ) ), this, SLOT( ListBoxWordsClickSlot(const QString& ) ));
```

第 1 个连接函数是点击图形界面中的 search 按钮组件发送信号, 由自定义的槽 PbtSearchPress() 接收信号。槽的设计中是在组件行编辑框中输入单词点击 search 按

按钮组件，将对应单词的意思和音标显示在组件文本编辑框中，整个槽函数的实现思想是执行 SQL 语句得到数据的行和列，用行和列的乘积设置循环查找结束条件，如果输入的内容与数据库找到的内容匹配，就将其词义和音标追加到文本框中。限于篇幅，槽函数的实现代码在这里不一一说明。第 2 个连接函数是在组件行编辑框中输入，即发送一个信号给槽 `lineEditChangeSlot(const QString&)`。槽的设计中是列表编辑框的内容随着行编辑框的内容变化而变化。设计中使用了 Qt/Embedded 库的 `QListBox` 类中的 `insertItem()` 函数，此函数的主要功能是插入项目到列表中。其实现思想类似前一个槽函数的实现，不再详述。第 3 个连接函数是在组建列表编辑框中点击某个单词即发送一个信号给槽 `ListboxWords-ClickSlot(const QString&)`。槽的设计中是点击列表框的单词，只在文本框显示其解释和词性。其实现思想类似前面设计的槽。利用上述思想就可以完整地实现单词查询的基本功能。图 2 是输入单词后点击 search 的一个运行效果图。图中左边显示了词库(CET-4)中以“fail”为索引的项，右边是单词的音标、中文解释和词性。

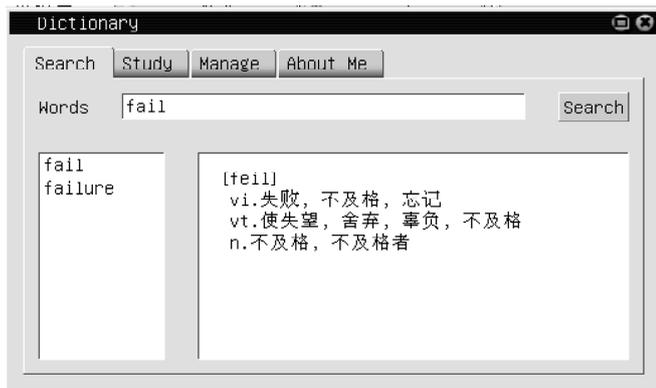


图 2 点击 search 按钮查询单词的截图

为了方便用户查询，设计了词典的单词索引功能，可以通过下拉索引列寻找要找的单词。点击索引列的单词进行词义查询。图 3 是输入“c”后点击索引列的“catch”后的截图。



图 3 索引列的单词显示截图

3.3 编译并添加应用程序到 Qtopia

编译 Qt/Embedded 应用程序大致需要 3 步:(1)生成工程文件.pro;(2)生成 Makefile 文件;(3)使用 Linux 自带的 make 生成一个二进制的可执行文件，具体编译过程请参考在线帮助文档。下面介绍如何在已经安装好的 Qtopia 里添加编写的应用程序(Dictionary)：

(1)建立程序 Dictionary 的图标文件

制作一个 32 × 32 大小的 PNG 格式的图标文件，将此文件存放在 Qtopia/pic/inline 目录下，然后利用 qt-x11-free-3.3.3 里的一个工具 `qembed`，将 Qtopia/pics/inline 下所有的图形文件转换成一个 C 语言的头文件，此头文件包含了该目录下的图形文件的 rgb 信息；

(2)重新交叉编译 Qtopia(具体编译过程可参考在线文档^[7])；

(3)建立应用启动器 (.desktop) 文件,将其保存在 \$QPEDIR/apps/applications 目录下,具体可参考 Qtopia 自带应用的.desktop 文件；

(4)建立根文件系统

利用原有的 `qtopia.cramfs` 的根文件系统映象，把新建的应用的相关文件添加到这个根文件系统中。首先要将 `qtopia.cramfs` 的根文件系统 mount 到工作机器上，然后复制这个文件系统的内容到一个临时目录下，这时可以看到根文件系统里的 `qpe` 安装目录，接着把新建的应用的相关文件（包括启动器文件，包含了图标的库文件 `libqte.so.*` 和应用程序的可执行文件）复制到 `qpe` 的对应的目录下。将生成的新的根文件系统烧写到 `s3c2410` 的 Flash 根文件系统区；

(5)运行

重新运行 Qtopia,就可以看到所添加的应用图标，如图 4 所示。点击“Dictionary”图标,就可以运行电子词典。



图 4 点击“Dictionary”图标的截图

嵌入式产品已经成为新的技术热点,而嵌入式 GUI 是嵌入式 Linux 不可缺少的组成部分。Qt/Embedded 和 Qtopia 在智能终端等嵌入式系统中得到广泛应用。结合

(下转第 27 页)

(上接第 23 页)

电子词典的开发实例介绍了基于 Qt/Embedded 的 GUI 应用程序开发过程, 并编译调试实现。Qt/Embedded 是比较理想的 GUI 开发环境。

参考文献

- [1] 任善全, 吕强, 钱培德, 等. 一个基于Qt/Embedded的嵌入式Linux应用程序的实现[J]. 计算机应用与软件, 2006, 23(2): 106-108.
- [2] 张鹏, 张雪兰. 基于嵌入式Linux的GUI应用程序的实现[J]. 计算机应用, 2003, 23(4): 115-117.
- [3] 王策, 赵宇. 基于Linux的嵌入式系统开发[J]. 计算机应用,

《信息化纵横》2009年第14期

2002, 22(7): 54-56.

- [4] Xteam(中国)软件技术有限公司.Qt程序设计[M], 北京:清华大学出版社, 2002: 79-83.
- [5] 倪继利. Qt及Linux操作系统窗口设计[M]. 北京:电子工业出版社, 2006.
- [6] 齐亮. C++ GUI Qt3编程[M]. 北京:北京航空航天大学出版社, 2006.
- [7] Trolltech online reference documentation[M]. <http://doc.trolltech.com>. 2003.3.

(收稿日期: 2009-03-23)

欢迎网上投稿 www.pcachina.com

27