

# 用 Qt 进行面向对象软件的设计与开发

万长胜, 金 革, 王 坚, 李昔华, 郝黎凯, 黄 鲲, 李 峰, 任 间,

(中国科技大学近代物理系 安徽合肥 230027)

**摘要:** Qt 是目前 Linux 下比较流行的面向对象的 C++ 类库。介绍了 Qt 类库结构和具体应用, 并结合“九五”大工程项目 LAMOST 望远镜的 GUI 的开发作进一步的说明。

**关键词:** Qt; 面向对象; 类库

**中图分类号:** TP319 **文献标识码:** A **文章编号:** 0258-0934(2004)03-0311-03

## 0 引言

软件设计的目标在于形成软件的一种层次的可对底层节点交叉引用的模块化结构(图 1)<sup>[1]</sup>, 软件设计方法考虑的就是如何分解出这样的层次模块结构。面向对象的软件设计基于对数据的抽象, 对模块的分解以“信息隐藏”为准则, 而以数据操作作为模块界面。不管使用什么样的语言, 只要软件系统结构以数据为中心进行组织, 遵循模块化的一般准则并具有数据抽象, 属性继承等模块特征, 都认为是面向对象设计的。面向对象的三个基本特征<sup>[2]</sup>: 封装, 继承和多态。

Qt 类库是一个跨平台的面向对象的 C++ 类库, 目前主要用于 Linux 下的开发。普通的 X-Window 编程是极其底层和繁琐的, 而 Qt 的高度面向对象和模块化的特征将开发人员从这种繁琐中解脱了出来。Qt 不像 Motif 那样只有商业版本, 它还提供了免费的版本, 这对于非商业的科学项目如我们目前正在开发国家“九五”大科学工程“大天区面积多目标光纤光谱天文

望远镜”的观测控制系统特别适合, 同时 Qt 还是跨平台的, 这对于要求移植性很强的项目也是很好的选择。

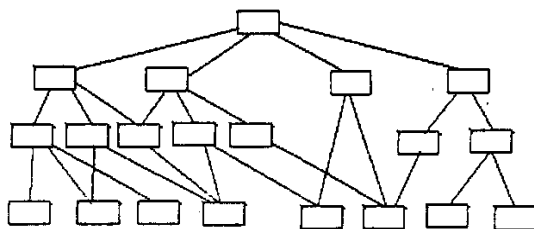


图 1 软件设计树

## 1 Qt 类库

Qt 类库大致可以分为三个部分: 控件, 框架和工具。

### 1.1 控件

控件部分包括环境控件, 主窗口控件, 标准对话框, 基本的 GUI 控件, 扩展 GUI 控件, GUI 组织控件, 以及帮助系统控件。

环境控件为应用程序提供全局服务的类, 包括系统设定、国际化等。例如 QTranslator, QApplication。主窗口类为应用程序提供界面框架, 可以在上面添加菜单、工具条等, 为应用程序提供集装箱的功能, 如 QMainWindow。标准对话框类是打开/关闭文件, 选择颜色等预先设计 的 标 准 对 话 框, 如 QColorDialog, QFileDialog。基本控件包括所有的 GUI 控件, 如按钮、组合框等, 如 QPushButton。扩展的控

收稿日期: 2003-01-03

作者简介: 万长胜(1976-), 男, 湖北随州人, 中国科技大学近代物理系博士生, 从事物理电子学方面的研究, 现正参与国家“九五”大工程项目 LAMOST 的开发工作。

件包括树状视图、进度条等,如 `QListView`。  
GUI 组织类负责对各种控件进行组织以构成复杂的对话框,如 `QGroupBox`。帮助系统是为应用程序提供在线帮助的类,如 `QStatusBar`,  
`QToolBar`。

1.2 框架

框架部分包括的是一些抽象的类,通常不可见,如对象模型、抽象控件、绘图、拖放、控件外观。

对象模型是框架的基础,如 `QObject`。可见的控件一般从抽象控件派生,如 `QPushButton`。绘图类控制绘图,如 `QBrush`。拖放类控制拖放操作,如 `QDragObject`。控件外观类控制控件的外观如颜色、字体等。例如 `QColor`。

1.3 工具

工具部分包括时间日期和链表树等数据结构,它们和 GUI 无关。

普通工具包括链表、堆栈、队列、树等常见数据结构,如 `QArray`。图形处理工具控制图像的编码/解码算法。如 `QImageIO`。I/O 控制工具处理 I/O 的一些类,如 `QFile`。时间和日期工具类处理时间和日期,如 `QDate`, `QTime`。另外还有其他杂类如 `QSignal`, `QIconSet` 等。

2 Qt 对象间通讯机制

对象间通讯是面向对象程序设计的一个极其重要的内容,类似于 `Microsoft MFC` 的消息映射 (`Message Mapping`) 和事件循环,Qt 的对象间通讯采用的是 `signal-slot` 机制, `signal` 就好像是事件,而 `slot` 则是响应事件的方法,如果需要对对象间的通讯,只需要把一个对象的 `slot` 和另外一个对象的 `signal` 连接起来,就可以实现“事件驱动”。

在 Qt 中 `signal` 是一个全新的东西,它既不属于成员函数也不是变量,用户可以自定义 `signal`,下面是自定义 `signal` 的例子:

```
signals:
    void created();
```

自定义的 `signal` 可以在需要的时候发送,要发送上面的 `signal`,可以用下面的语句:

```
emit created();
```

3 Qt 应用程序框架及开发工具

Qt 的应用程序框架 (`framework`): Qt 的应

用程序框架是自顶向下的,首先是在 `main` 函数中产生一个 `QApplication` 类的实例 `a` 和一个主窗口类 `w`,主窗口可以是任何 `QWidget` 类的派生类,这里用的是 `QMainWindow` 类,调用该类的方法来设置主窗口的位置和大小,调用 `a` 的方法来把窗口 `w` 设置成主窗口并显示,剩下的工作就是按照项目对面板的要求来编辑主窗口类 `w` 了。下面是一个最简单的程序框架:

```
# include "mainwindow.h"
int main(int argc, char * * argv)
{
    QApplication a(argc, argv);
    QMainWindow w;
    w.setGeometry(150, 150, 300, 200);
    a.setMainWindow(&w);
    w.show();
    return a.exec();
}
```

Linux 下开发 Qt 应用程序比较好的 IDE 是 `KDevelop`。`KDevelop` 是 X 系统中的一个快速开发工具。它集成了很多其他实用工具,如 `make` 和 `GNU C++ Compilers`,它还可以为工程的每个类自动生成 HTML 格式的 API 文档,同时有着很好的项目管理、编辑环境和调试环境,另外它还有一个所见即所得的对话框编辑器。

4 用 Qt 进行 LAMOST GUI 的开发

4.1 LAMOST 的 GUI

大天区面积多目标光纤光谱天文望远镜<sup>[3]</sup> (LAMOST) 是我国正在建造的一个世界上光谱观测能力最强的大口径兼备大视场的反射施密特光谱望远镜。LAMOST 的控制系统由观测战略系统 (SSS)、观测控制系统 (OCS)、望远镜控制系统 (TCS)、焦面仪器控制系统 (ICS) 和数据处理系统 (DHS) 部分组成,其中观测控制系统 (OCS) 是控制系统的核心 (图 2),它将控制和管理各子系统实现各自的功能,完成观测任务。

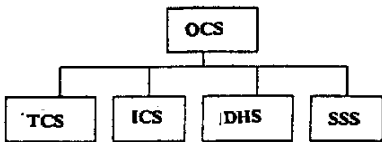


图2 OCS 系统的结构

对于整个控制系统来说,它需要接受终端

用户的请求, 并把用户需要的信息显示出来。这就需要一个人机交互的界面(GUI)。整个UI的结构框图如图3所示, 分为3部分: 第一部分是UI面板, 该部分负责信息的收集和显示, 各子系统是不同的; 第二部分是UI控制器, 该部分负责处理UI面板收集的信息和从UI接口来的信息, 并把它们或交给UI面板去显示或提交给模拟内核去处理, 各子系统的UI控制器都是类似的; 第三部分是模拟内核, 它实际上就是GUI与OCS的接口, 该部分处理UI控制器来的数据, 并把处理结果送回UI控制器, 这部分不仅要有一个事件处理器, 还要有接口模块负责与各个系统的UI联系在整个控制系统中, 模拟内核是UI系统的核心, 它把各子系统的UI联系起来并利用观测控制系统OCS提供的工具实现用户的要求。

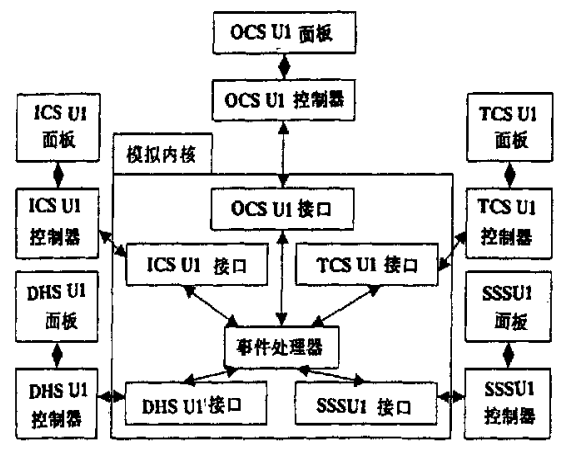


图3 UI结构图

4.2 使用Qt库开发LAMOST的GUI

我们使用KDevelop的应用程序向导来创建整个项目。在各子系统及OCS的面板中, 窗

口类从QMainWindow类派生, 这样我们能方便地使用Qt的QMenuBar类来添加菜单, 并使用Qt的控件类来摆放面板控件, 整个程序以事件驱动, 通过Qt的信号-slot机制与终端用户交互, 在LAMOST的UI结构图(图3)中可以看出, UI面板不仅要与用户交互, 还要与UI控制器交互, Qt的QThread类解决了这种程序的并行性, 与线程紧密相连的就是现成的同步问题, 我们采用Qt库的QMutex来对共享资源加锁。

UI控制器的设计, 采用消息队列机制, 并行的处理各条消息。由于涉及到信息缓存的问题, 需要消息队列。Qt类库也为我们提供了相应的消息队列类。

图3中的事件处理器, 由于是并行的处理不同的消息, 也同样需要多个Qt的消息队列对象和线程对象。

5 结束语

Qt库是目前Linux下很流行的一个面向对象的C++类库, 而KDevelop则是目前Linux较流行的一个IDE。随着时间的推移, 使用Qt库来开发Linux下的面向对象应用软件将会变得越来越广泛。

参考文献

[1]冯玉琳, 等. 软件工程[M]. 合肥: 中国科学技术大学出版社, 1998  
[2]李宋琛. Linux面向对象窗口高级编程[M]. 北京: 科学出版社, 2001  
[3]LAMOST工程指挥部. LAMOST可行性研究报告[R]. 北京: LAMOST工程指挥部, 1997.

Design and development of object-oriented software based on Qt

WAN Chang-sheng, JIN Ge, WANG Jian, LI Xi-hua, HAO Li-kai,  
HUANG Kun, LI Feng, REN Jian

(Department of Modern Physics, University of Science and Technology of China,  
Hefei of Anhui, Prov. 230027, China)

**Abstract** Qt is a popular object-oriented C++ class library under Linux environment. In this paper we will introduce the object-oriented technology, the Qt library and the IDE to develop Qt software. At last an example of Qt software will be shown in detail.

**Key words:** Qt; object-oriented; class library