



豆子空间

<http://devbean.blog.51cto.com> [【复制】](#) [【订阅】](#)

[博 客](#) | [图 库](#) | [写 博 文](#) | [帮 助](#)

[首页](#) | [Java](#) | [JavaScript](#) | [C/C++](#) | [Qt](#) | [Flex](#) | [Photoshop](#) | [Firefox](#) | [杂谈](#)

FinderCheng 的BLOG

相关视频课程

更多



[写留言](#)

[去学院学习](#)

[发消息](#)

[加友情链接](#)

[进家园](#) [加好友](#)

博客统计信息

[51CTO推荐博客](#)

用户名: FinderCheng

文章数: 123

评论数: 989

访问量: 3542454

无忧币: 2042

博客积分: 3388

博客等级: 7

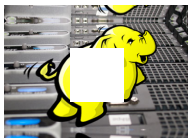
注册日期: 2008-08-11

热门专题

[更多>>](#)



精通Spark内核系列课程5: 核心源码剖析 (共309人学习)



从源码角度分析Hadoop中Eclipse开发的代码 (2519人学习)



Spark源码完整解析和系统定制系列课程5: (63人学习)

[博主的更多文章>>](#)

原创 Qt核心剖析：信息隐藏(3)

2010-06-19 12:49:56

标签: [源码](#) [Qt](#) [剖析](#) [休闲](#) [核心](#)

原创作品，允许转载，转载时请务必以超链接形式标明文章 [原始出处](#)、作者信息和本声明。否则将追究法律责任。
任。<http://devbean.blog.51cto.com/448512/335550>

前面我们已经看到了怎样使用标准的 C++ 代码以及 Qt 提供的 API 来达到信息隐藏这一目标。下面我们来看一下 Qt 是如何实现的。

还是以 QObject 的源代码作为例子。先打开 qobject.h, 找到 QObjectData 这个类的声明。具体代码如下所示:

```
01.  QObjectData {
02.  public:
03.      virtual ~QObjectData() = 0;
04.      // others
05.  };
```

然后在下面就可以找到 QObject 的声明:

```
01.  class QObject
02.  {
03.      Q_DECLARE_PRIVATE(QObject)
04.  public:
05.      Q_INVOKABLE explicit QObject(QObject *parent=0);
06.  protected:
07.      QObject(QObjectPrivate &dd, QObject *parent = 0);
08.      QScopedPointer<QObjectData> d_ptr;
09.      // others
10.  }
```

注意，这里我们只是列出了我们所需要的代码，并且我的 Qt 版本是 2010.03。这部分代码可能会随着不同的 Qt 版

热门文章

- Qt学习之路(17): Qt标准..
- Qt学习之路(2): Hello, w..
- Qt学习之路(1): 前言
- Qt学习之路(50): QString
- Qt学习之路(7): 创建一个..
- Qt学习之路(8): 创建一个..
- Qt学习之路(15): Qt标准..
- 深入Java单例模式

搜索BLOG文章

搜索

最近访客



10552..



72974..



15192..



jim003



qinzy212



61940..



sucha..



肇庆..



一心永成



a9341..



loserfly



gaoku..

最新评论

飘雪夏娜样: 书上说的都说了, 书上没有讲清楚的..

本所有不同。

首先先了解一下 Qt 的设计思路。既然每个类都应该把自己的数据放在一个 private 类中, 那么, 为什么不把这个操作放在几乎所有类的父类 QObject 中呢? 所以, Qt 实际上是用了这样一个思路, 实现了我们前面介绍的数据隐藏机制。

首先回忆一下, 我们前面说的 D-Pointer 需要有一个 private 或者 protected 的指向自己数据类的指针。在 QObject 中,

```
01. QScopedPointer<QObjectData> d_ptr;
```

就扮演了这么一个角色。或许, 你可以把它理解成

```
01. QObjectData *d_ptr;
```

这不就和我们前面说的 D-Pointer 技术差不多了? QScopedPointer 是 Qt 提供的一个辅助类, 这个类保存有一个指针, 它的行为类似于一种智能指针: 它能够保证在这个作用域结束后, 里面的所有指针都能够被自动 delete 掉。也就是说, 它其实就是一个比普通指针更多功能的指针。而这个指针被声明成 protected 的, 也就是只有它本身以及其子类才能够访问到它。这就提供了让子类不必声明这个 D-Pointer 的可能。

那么, 前面我们说, QObjectData 这种数据类不应该放在公开的头文件中, 可 Qt 怎么把它放进来了呢? 这样做的用途是, QObject 的子类的数据类都可能继承自这个 QObjectData。这个类有一个纯虚的析构函数。没有实现代码, 保证了这个类不能被初始化; 虚的析构函数, 保证了其子类都能够被正确的析构。

回到我们前面说明的 Q_DECLARE_PRIVATE 这个宏:

```
01. #define Q_DECLARE_PRIVATE(Class) \
02.     inline Class##Private* d_func() { return reinterpret_cast(qGetPtrHelper(d_ptr)); } \
03.     inline const Class##Private* d_func() const { return reinterpret_cast(qGetPtrHelper(d_ptr)); } \
04.     friend class Class##Private;
```

我们把代码中的 Q_DECLARE_PRIVATE(QObject) 展开看看是什么东西:

```
01. inline QObjectPrivate* d_func() { return reinterpret_cast(qGetPtrHelper(d_ptr)); }
02. inline const QObjectPrivate* d_func() const { return reinterpret_cast(qGetPtrHelper(d_ptr)); }
03. friend class QObjectPrivate;
```

清楚是清楚, 只是这个 QObjectPrivate 是哪里来的? 既然是 Private, 那么它肯定不会在公开的头文件中。于是我们立刻想到到 qobject.cpp 或者是 qobject_p.h 中寻找。终于, 我们在 qobject_p.h 中找到了这个类的声明:

```
01. class Q_CORE_EXPORT QObjectPrivate : public QObjectData
02. {
03.     Q_DECLARE_PUBLIC(QObject)
04.
05. public:
06.     QObjectPrivate(int version = QObjectPrivateVersion);
07.     virtual ~QObjectPrivate();
08.     // others
09. }
```

这个类是继承 QObjectData 的! 想想也是, 因为我们说过, QObjectData 是不能被实例化的, 如果要使用, 必须创

aguai1617: 请问有没有从应用程序拖动文件到本..
wx275271279: “这里其实是一个冗余的操作, 因为..
wx275271279: 另外还有一点, 在一些必须精确操作..
wx275271279: 讲的深入透彻, 楼主厉害, 十分感谢
51CTO推荐博文
更多>>
运维角度浅谈MySQL数据库优化
Oracle Study之一--AMD CPU安装Orac..
svn利用钩子脚本功能实现代码同步..
【UNITY3D 游戏开发之八】Unity编..
PL/SQL中如何让程序每隔几秒插入..
Keepalived+LVS+MariaDB Galera C..
Ubuntu14. 04快速搭建SVN服务器及..
整理的部分Java和C#不同点
Apache Httpd服务器之虚拟机详解
将Uhost上的MySQL迁移到UDB
mysql分布式中间件cobar
友情链接
石榴石网
趣闻网
汉堡店加盟
奶茶店加盟
IT精品课程
我的主页
PicWorks
Qt 文档翻译
Exchange 2010
51CTO博客开发

建它的一个子类。显然, QObjectPrivate 就扮演了这么一个角色了。不仅如此, 我们还在这里看到了熟悉的 Q_DECLARE_PUBLIC 宏。好在我们已经知道它的含义了。

在 qobject.cpp 中, 我们看一下 QObject 的构造函数:

```
01.  QObject::QObject(QObject *parent)
02.      : d_ptr(new QObjectPrivate)
03.  {
04.      // others
05.  }
06.
07.  QObject::QObject(QObjectPrivate &dd, QObject *parent)
08.      : d_ptr(&dd)
09.  {
10.      // others
11.  }
```

第一个构造函数就是我们经常见到的那个。它使用自己创建的 QObjectPrivate 指针对 d_ptr 初始化。第二个构造函数使用传入的 QObjectPrivate 对象, 但它是 protected 的, 也就是说, 你不能在外部类中使用这个构造函数。

那么这个构造函数有什么用呢? 我们来看一下 QWidget 的代码:

```
01.  class QWidget : public QObject, public QPaintDevice
02.  {
03.      Q_OBJECT
04.      Q_DECLARE_PRIVATE(QWidget)
05.      // others
06.  }
```

QWidget 是 QObject 的子类, 然后看它的构造函数:

```
01.  QWidget::QWidget(QWidget *parent, Qt::WindowFlags f)
02.      : QObject(*new QWidgetPrivate, 0), QPaintDevice()
03.  {
04.      QT_TRY {
05.          d_func()->init(parent, f);
06.      } QT_CATCH(...) {
07.          QWidgetExceptionCleaner::cleanup(this, d_func());
08.          QT_RETHROW;
09.      }
10.  }
```

它调用了那个QObject 的 protected 构造函数, 并且传入一个 QWidgetPrivate ! 这个 QWidgetPrivate 显然继承了 QObjectPrivate。于是我们已经明白, 为什么 QWidget 中找不到 d_ptr 了, 因为所有的 d_ptr 都已经在父类 QObject 中定义好了! 尝试展开一下 Q_DECLARE_PRIVATE 宏, 你就能够发现, 它实际上把父类的 QObjectPrivate 指针偷偷地转换成了 QWidgetPrivate 的指针。这个就是前面说的 Qt 的设计思路。

本文出自 “豆子空间” 博客, 请务必保留此出处http://devbean.blog.51cto.com/448512/335550

分享至:

收藏 +

yushaolin、skiney、zhanghuhu0419 3人 了这篇文章

类别: Qt | 阅读 (6084) | 评论 (5) | 返回博主首页 | 返回博客首页



微信



关注51CTO博客微信
有机会赢下载VIP会员

微信号：blog51cto

[上一篇 Qt 文档翻译项目](#) [下一篇 Qt核心剖析：moc](#)



相关文章

- [Qt核心剖析：寻找 QObject 的源代码](#)
- [Qt核心剖析：信息隐藏\(2\)](#)
- [Qt学习之路\(14\)：状态栏](#)
- [qt翻译---QSemaphore](#)
- [晒晒Pimp My Widgets大赛的优胜作品](#)
- [Qt核心剖析：信息隐藏\(1\)](#)
- [基于Qt/Embedded和Qtopia的GUI设计](#)
- [菜鸟玩qt（3）---qt工具介绍](#)
- [Qt学习之路\(2\)：Hello, world!](#)
- [由函数调用约定引起的问题](#)

文章评论

[1楼]  anjing6066 回复

2010-06-23 21:52:36

豆子先生你好，
我按照c++ GUI Programming Qt4上的第二章 Shape-Changing Dialogs编写的成绩，出现了如下1两类的错误，百度之后无法解决，特像您求教：
invalid use of incomplete type 'struct Ui::sortdialog'
forward declaration of 'struct Ui::sortdialog'
sortdialog 是我定义的类名。

谢谢，麻烦您了！

[2楼]  anjing6066 回复

2010-06-24 08:29:50

我找到问题出在哪来，用Designer设计对话框的时候没有保存成功！

[3楼]  河豚 回复

2010-06-29 16:33:58

你刚毕业一年就那么牛B啊, 看来中国的IT将来还有希望.

[4楼] 楼主  FinderCheng 回复

2010-06-29 18:42:58

回复 河豚:[3楼]
过奖哦，这个我也是学习的啊，也不都是我自己研究出来的，呵呵

非常感谢

发表评论

[51CTO学院2周年庆，分享故事赢大礼包](#)

昵 称:

验证码: 请点击后输入验证码 [博客过2级，无需填写验证码](#)

内 容: