

GeekBand 极客班

互联网人才加油站!

C++设计模式

www.geekband.com

GeekBand 极客班 互联网人才+加油站：

极客班携手 网易云课堂，针对热门IT互联网岗位，联合业内专家大牛，紧贴企业实际需求，量身打造精品实战课程。

专业课程

+

项目碾压

+

习题&辅导

- | | | |
|------------|----------------|----------|
| • 顶尖大牛亲授 | • 紧贴课程内容 | • 学前导读 |
| • 贴合企业实际需求 | • 全程实战操练 | • 周末直播答疑 |
| • 找对重点深挖学习 | • 作品就是最好的PASS卡 | • 定期作业点评 |
| | | • 多项专题辅导 |



www.geekband.com

C++设计模式

Singleton 单件模式

李建忠

“对象性能” 模式

➤面向对象很好地解决了“抽象”的问题，但是不可避免地要付出一定的代价。对于通常情况来讲，面向对象的成本大都可以忽略不计。但是某些情况，面向对象所带来的成本必须谨慎处理。

- 典型模式
 - Singleton
 - Flyweight

Singleton 单件模式

动机 (Motivation)

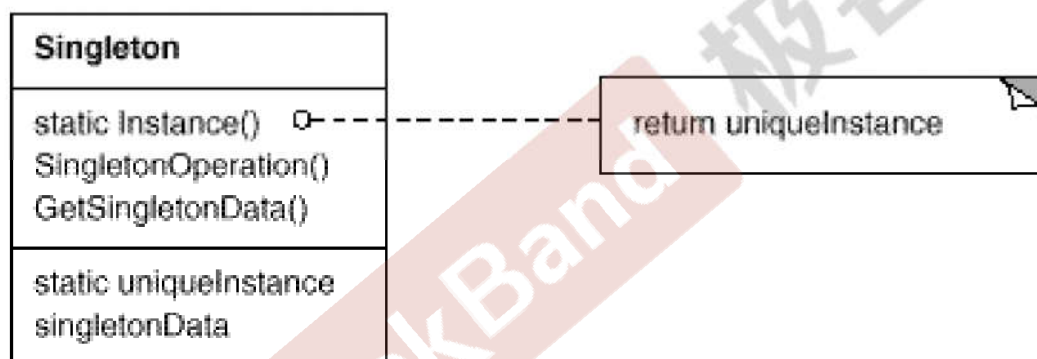
- 在软件系统中，经常有这样一些特殊的类，必须保证它们在系统中只存在一个实例，才能确保它们的逻辑正确性、以及良好的效率。
- 如何绕过常规的构造器，提供一种机制来保证一个类只有一个实例？
- 这应该是类设计者的责任，而不是使用者的责任。

模式定义

保证一个类仅有一个实例，并提供一个该实例的全局访问点。

——《设计模式》GoF

结构 (Structure)



要点总结

- Singleton模式中的实例构造器可以设置为protected以允许子类派生。
- Singleton模式一般不要支持拷贝构造函数和Clone接口，因为这有可能导致多个对象实例，与Singleton模式的初衷违背。
- 如何实现多线程环境下安全的Singleton？注意对双检查锁的正确实现。