

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

Command 命令模式

李建忠

GeekBald 极客班

“行为变化” 模式

➤在组件的构建过程中，组件行为的变化经常导致组件本身剧烈的变化。“行为变化”模式将组件的行为和组件本身进行解耦，从而支持组件行为的变化，实现两者之间的松耦合。

➤典型模式

- Command
- Visitor

Command 命令模式

动机 (Motivation)

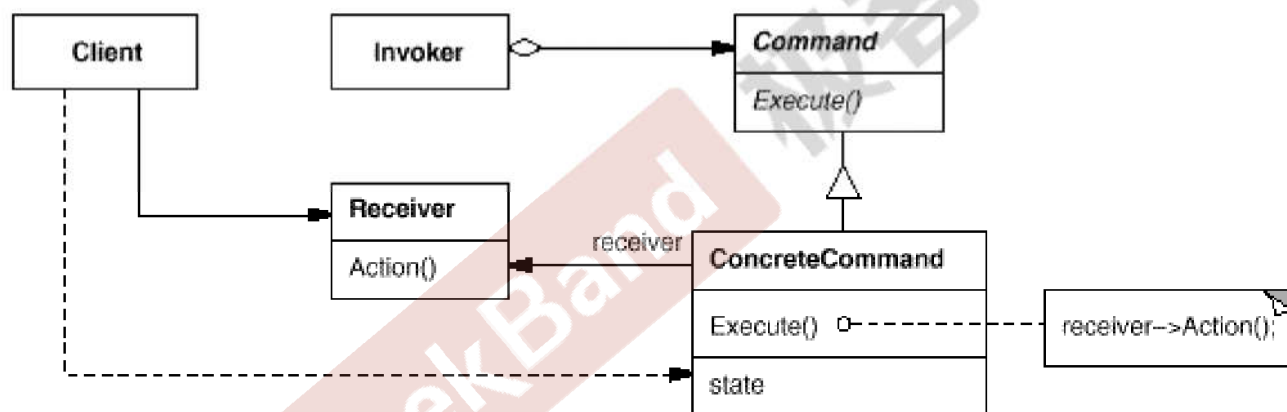
- 在软件构建过程中，“行为请求者”与“行为实现者”通常呈现一种“紧耦合”。但在某些场合——比如需要对行为进行“记录、撤销/重(undo/redo)、事务”等处理，这种无法抵御变化的紧耦合是不合适的。
- 在这种情况下，如何将“行为请求者”与“行为实现者”解耦？将一组行为抽象为对象，可以实现二者之间的松耦合。

模式定义

将一个请求(行为)封装为一个对象，从而使你可用不同的请求对客户进行参数化；对请求排队或记录请求日志，以及支持可撤销的操作。

——《设计模式》GoF

结构 (Structure)



要点总结

- Command模式的根本目的在于将“行为请求者”与“行为实现者”解耦，在面向对象语言中，常见的实现手段是“将行为抽象为对象”。
- 实现Command接口的具体命令对象ConcreteCommand有时候根据需要可能会保存一些额外的状态信息。通过使用Composite模式，可以将多个“命令”封装为一个“复合命令”MacroCommand。
- Command模式与C++中的函数对象有些类似。但两者定义行为接口的规范有所区别：Command以面向对象中的“接口-实现”来定义行为接口规范，更严格，但有性能损失；C++函数对象以函数签名来定义行为接口规范，更灵活，性能更高。