

设计模式（第2版）

第17章 职责链模式

刘伟

大 纲

1

模式动机与定义

2

模式结构与分析

3

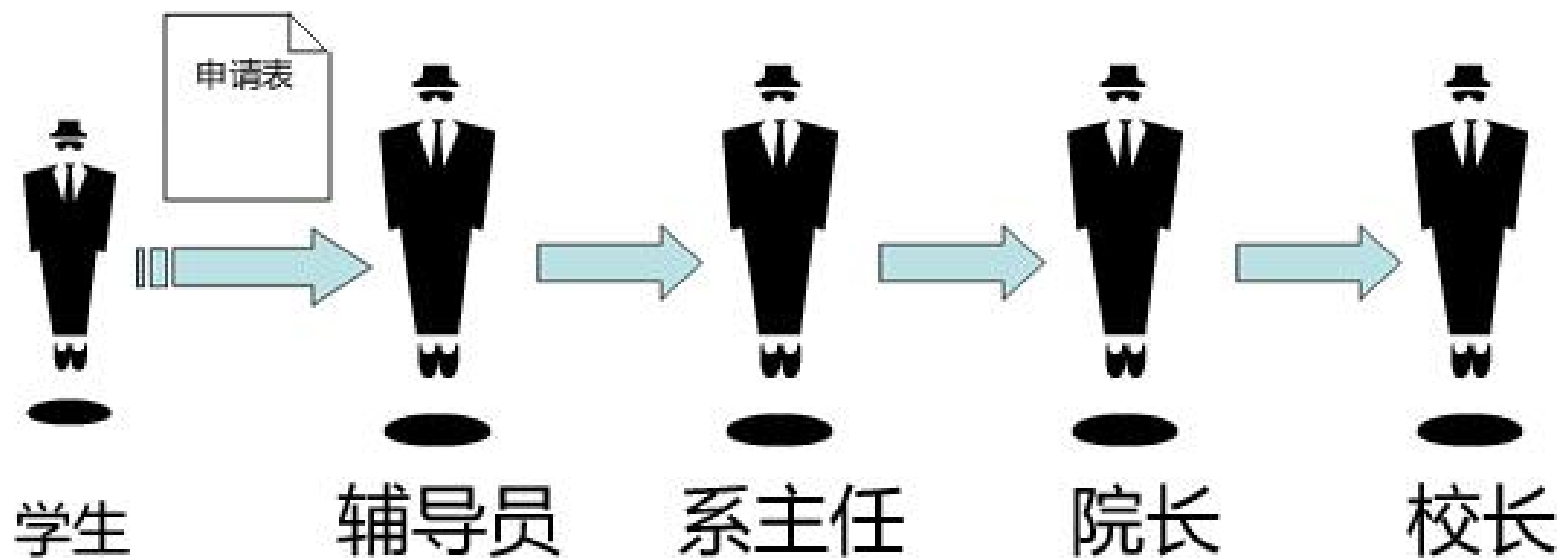
模式实例与解析

4

模式效果与应用

● 模式动机

✓ 奖学金审批



● 模式动机

- ✓ 辅导员、系主任、院长、校长都可以处理奖学金申请表，他们构成一个处理申请表的链式结构，申请表沿着这条链进行传递，这条链就称为职责链
- ✓ 职责链可以是一条直线、一个环或者一个树形结构，最常见的职责链是直线型，即沿着一条单向的链来传递请求

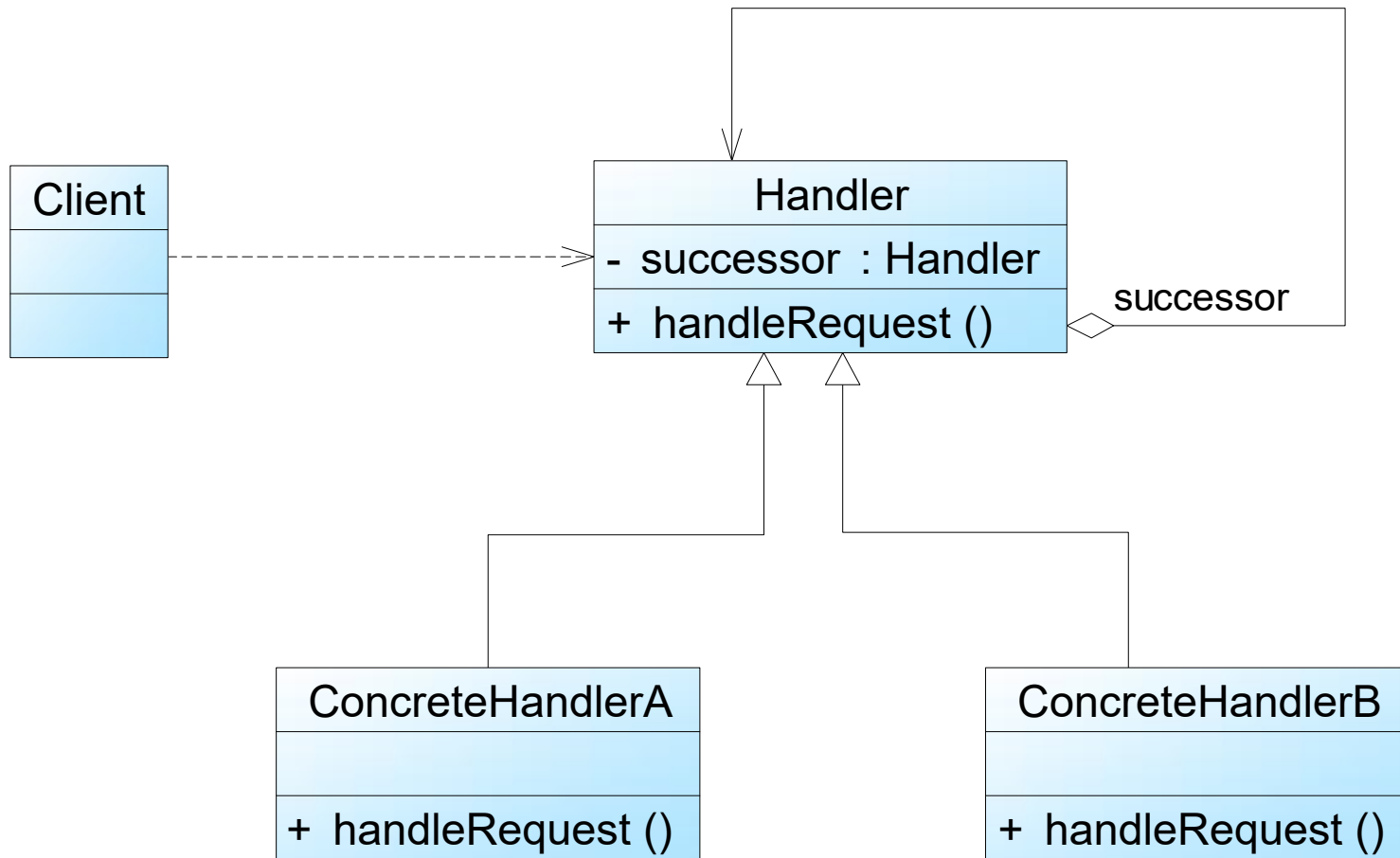


● 模式定义

- ✓ 职责链模式(Chain of Responsibility Pattern): 避免请求发送者与接收者耦合在一起, 让多个对象都有可能接收请求。将这些对象连接成一条链, 并且沿着这条链传递请求, 直到有对象处理它为止。
- ✓ 对象行为型模式



● 模式结构



● 模式结构

- ✓ 职责链模式包含如下角色：
 - Handler: 抽象处理者
 - ConcreteHandler: 具体处理者



● 模式分析

- ✓ 将请求的处理者组织成一条链，并让请求沿着链传递，由链上的处理者对请求进行相应的处理
- ✓ 客户端无须关心请求的处理细节以及请求的传递，只需将请求发送到链上，将请求的发送者和请求的处理者解耦



● 模式分析

✓ 抽象处理者示例代码：

```
public abstract class Handler {  
    //维持对下家的引用  
    protected Handler successor;  
  
    public void setSuccessor(Handler successor) {  
        this.successor=successor;  
    }  
  
    public abstract void handleRequest(String request);  
}
```



● 模式分析

✓ 具体处理者示例代码：

```
public class ConcreteHandler extends Handler {  
    public void handleRequest(String request) {  
        if (请求满足条件) {  
            //处理请求  
        }  
        else {  
            this.successor.handleRequest(request); //转发请求  
        }  
    }  
}
```



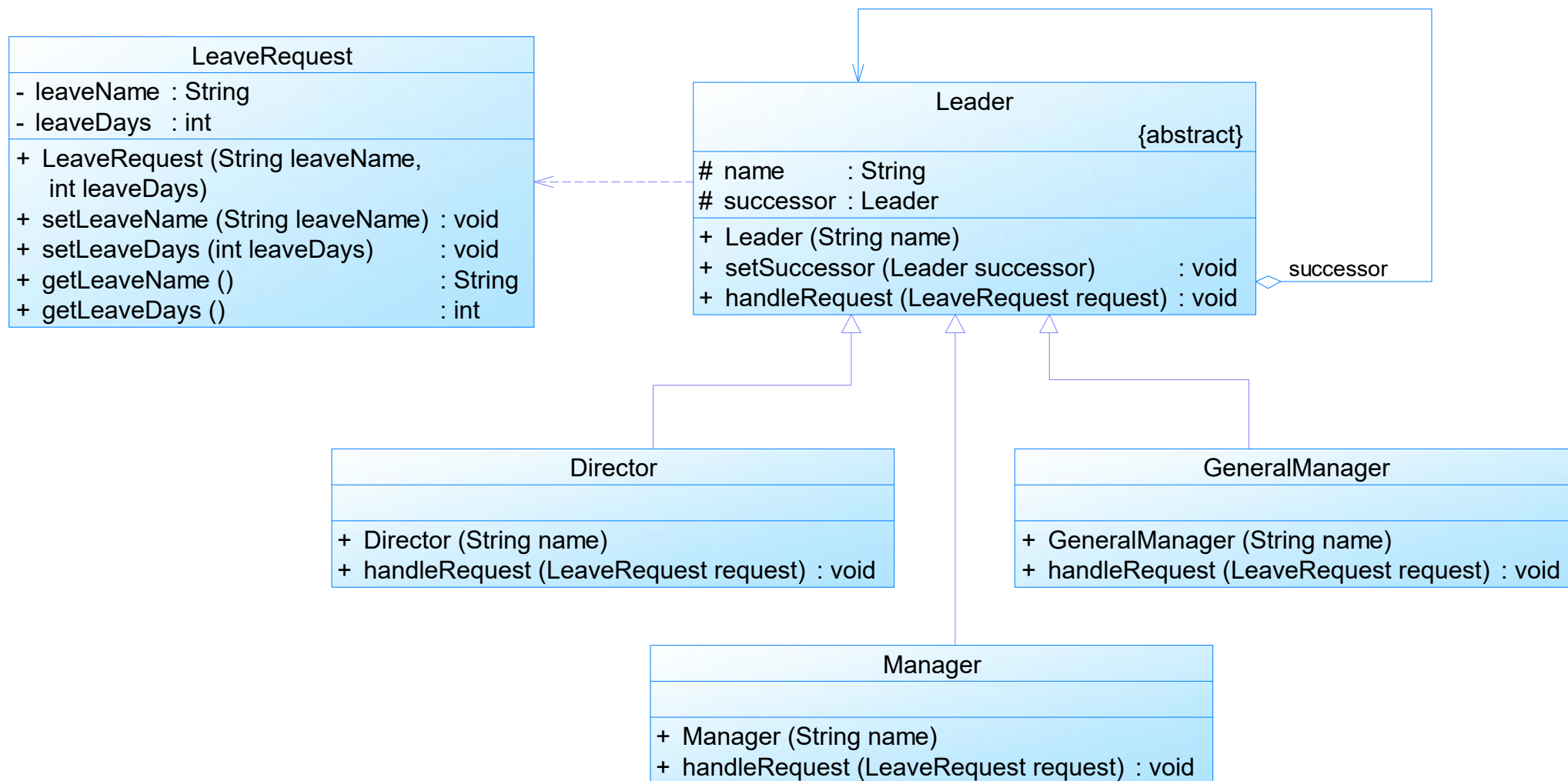
● 模式实例

✓ 审批假条：实例说明

- 某OA系统需要提供一个假条审批的模块，如果员工请假天数小于3天，主任可以审批该假条；如果员工请假天数大于等于3天，小于10天，经理可以审批；如果员工请假天数大于等于10天，小于30天，总经理可以审批；如果超过30天，总经理也不能审批，提示相应的拒绝信息。

● 模式实例

✓ 审批假条：参考类图



● 模式实例

- ✓ 审批假条：参考代码
 - DesignPatterns之cor包



● 职责链模式优点：

- ✓ 使得一个对象无须知道是其他哪一个对象处理其请求，降低了系统的耦合度
- ✓ 可简化对象之间的相互连接
- ✓ 给对象职责的分配带来更多的灵活性
- ✓ 增加一个新的具体请求处理者时无须修改原有系统的代码，只需要在客户端重新建链即可



● 职责链模式缺点：

- ✓ 不能保证请求一定会被处理
- ✓ 对于比较长的职责链，系统性能将受到一定影响，在进行代码调试时不太方便
- ✓ 如果建链不当，可能会造成循环调用，将导致系统陷入死循环



- 在以下情况下可以使用职责链模式：

- ✓ 有多个对象可以处理同一个请求，具体哪个对象处理该请求待运行时刻再确定
- ✓ 在不明确指定接收者的情况下，向多个对象中的一个提交一个请求
- ✓ 可动态指定一组对象处理请求





THANKS