

行为型模式

行为型模式(Behavioral Pattern)是对在不同的对象之间划分责任和算法的抽象化。

行为型模式不仅仅关注类和对象的结构，而且重点关注它们之间的相互作用。

通过行为型模式，可以更加清晰地划分类与对象的职责，并研究系统在运行时实例对象 之间的交互。在系统运行时，对象并不是孤立的，它们可以通过相互通信与协作完成某些复杂功能，一个对象在运行时也将影响到其他对象的运行。

行为型模式分为类行为型模式和对象行为型模式两种：

- 类行为型模式：类的行为型模式使用继承关系在几个类之间分配行为，类行为型模式主要通过多态等方式来分配父类与子类的职责。
- 对象行为型模式：对象的行为型模式则使用对象的聚合关联关系来分配行为，对象行为型模式主要是通过对象关联等方式来分配两个或多个类的职责。根据“合成复用原则”，系统中要尽量使用关联关系来取代继承关系，因此大部分行为型设计模式都属于对象行为型设计模式。

包含模式

- 职责链模式(Chain of Responsibility)
重要程度：3
- 命令模式(Command)
重要程度：4
- 解释器模式(Interpreter)
重要程度：1
- 迭代器模式(Iterator)
重要程度：5
- 中介者模式(Mediator)
重要程度：2
- 备忘录模式(Memento)
重要程度：2
- 观察者模式(Observer)

重要程度：5

- 状态模式(State)

重要程度：3

- 策略模式(Strategy)

重要程度：4

- 模板方法模式(Template Method)

重要程度：3

- 访问者模式(Visitor)

重要程度：1

目录

- 1. 命令模式

- 1.1. 模式动机
- 1.2. 模式定义
- 1.3. 模式结构
- 1.4. 时序图
- 1.5. 代码分析
- 1.6. 模式分析
- 1.7. 实例
- 1.8. 优点
- 1.9. 缺点
- 1.10. 适用环境
- 1.11. 模式应用
- 1.12. 模式扩展
- 1.13. 总结

- 2. 中介者模式

- 2.1. 模式动机
- 2.2. 模式定义
- 2.3. 模式结构
- 2.4. 时序图
- 2.5. 代码分析
- 2.6. 模式分析
- 2.7. 实例
- 2.8. 优点
- 2.9. 缺点
- 2.10. 适用环境
- 2.11. 模式应用
- 2.12. 模式扩展
- 2.13. 总结

- 3. 观察者模式

- 3.1. 模式动机
- 3.2. 模式定义
- 3.3. 模式结构
- 3.4. 时序图
- 3.5. 代码分析
- 3.6. 模式分析
- 3.7. 实例
- 3.8. 优点
- 3.9. 缺点
- 3.10. 适用环境
- 3.11. 模式应用
- 3.12. 模式扩展
- 3.13. 总结
- 4. 状态模式
 - 4.1. 模式动机
 - 4.2. 模式定义
 - 4.3. 模式结构
 - 4.4. 时序图
 - 4.5. 代码分析
 - 4.6. 模式分析
 - 4.7. 实例
 - 4.8. 优点
 - 4.9. 缺点
 - 4.10. 适用环境
 - 4.11. 模式应用
 - 4.12. 模式扩展
 - 4.13. 总结
- 5. 策略模式
 - 5.1. 模式动机
 - 5.2. 模式定义
 - 5.3. 模式结构
 - 5.4. 时序图
 - 5.5. 代码分析
 - 5.6. 模式分析
 - 5.7. 实例
 - 5.8. 优点
 - 5.9. 缺点
 - 5.10. 适用环境
 - 5.11. 模式应用
 - 5.12. 模式扩展
 - 5.13. 总结