

软件体系结构

Zhenyan Ji

— Beijing Jiaotong University —

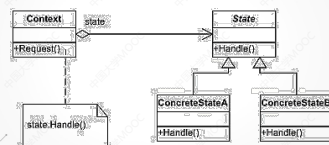
行为型模式--状态模式

状态模式

目的:

- 允许对象在其内部状态改变时改变其行为，看起来像改变了它的类。

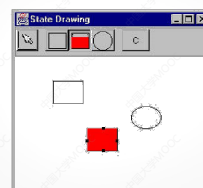
状态模式



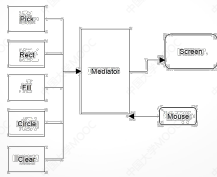
状态模式：参与者

- Context**
 - 定义client感兴趣的接口
 - 维护定义当前状态的ConcreteState子类的实例。
- State**
 - 定义用于封装与Context的特定状态相关联的行为的接口。
- Concrete State**
 - 每个子类实现与Context状态相关联的行为。

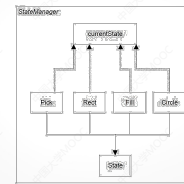
状态模式：例子



状态模式：例子



状态模式：例子2



状态模式：适用性

» 状态模式的应用场景：

- 对象的行为取决于它的状态，并且它必须根据该状态在运行时改变其行为。
- 对象方法中具有依赖对象状态的大量if-else或Switch-case语句。状态模式将条件的每个分支置于单独的类中。

结果

» 好处

- 将与状态相关的所有行为都封装于一个对象中
- 允许状态迁移逻辑封装到状态对象中而不是用大量的if-else或switch-case语句

» 坏处

- 对象数量增加