## 简介

不破坏封装的前提下，保存对象的当前状态，并外部化，存储起来，从而在将来需要的时候把对象恢复到当前状态。常与迭代器模式和命令模式一起使用。

## 角色

1. 备忘录角色（Memento）：
   1. 保存对象的状态，备忘录可以根据发起人的判断来决定存储多少发起人对象的内部状态。
   2. 备忘录可以保护其内容不被发起人以外的任何对象读取到。
   3. 备忘录有两个等效的接口：
      1. 窄接口：发起人之外的任何对象看到的是备忘录的窄接口，这个接口只允许它把备忘录对象传递给其他对象
      2. 宽接口：发起人看到的接口，允许读取所有数据，以便根据数据恢复发起人对象内部状态。
2. 发起人角色（originator）：角色责任如下：
   1. 创建含有一个当前内部状态的备忘录对象
   2. 使用备忘录对象存储其内部状态
3. 负责人角色（CareTaker）：责任如下：
   1. 负责保存备忘录对象
   2. 不检查备忘录对象的内容

## 分类

1. 白箱备忘录模式：备忘录角色对所有对象都存储一个宽接口，备忘录对象内部存储状态对多有对象都公开
2. 黑箱备忘录模式：备忘录角色对发起人对象提供一个宽接口，对其他对象提供窄接口

## 优缺点

优点

1. 窄接口保证发起者数据的安全性
2. 保存发起者历史状态

缺点：

1. 资源占用多

## 使用场景

需要记录和维护属性历史的功能

## 参考

<http://www.cnblogs.com/java-my-life/archive/2012/06/06/2534942.html>