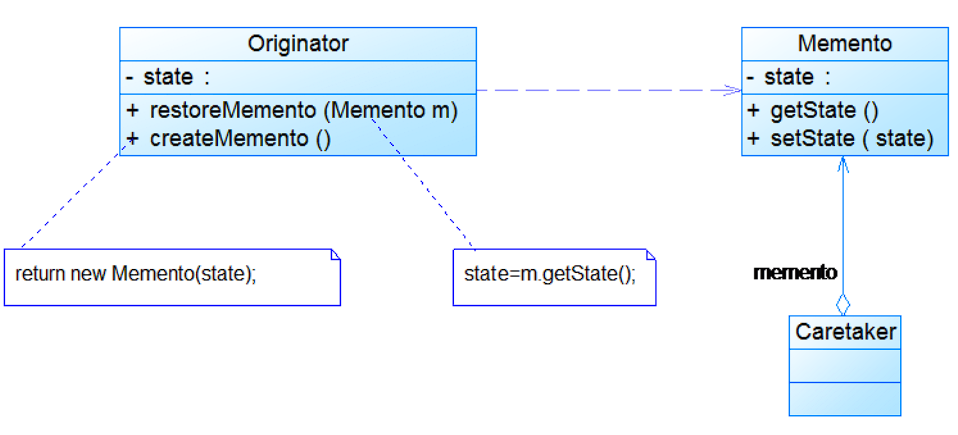
1. 模式定义

备忘录模式（Memento尸attern)：在不破坏封装的前提下，捕获一个对象的内部状态，并在该对象之外保存这个状态，这样可以在以后将对象恢复到原先保存的状态。它是一种对象行为型模式，其别名为Token。

**二、**模式动机

在应用软件的开发过程中，很多时候我们都需要记录一个对象的内部状态。

在具体实现过程中，为了允许用户取消不确定的操作或从错误中恢复过来，需要实现备份点和撤销机制，而要实现这些机制，必须事先将状态信息保存在某处，这样才能将对象恢复到

它们原先的状态。

备忘录模式是一种给我们的软件提供后悔药的机制，通过它可以使系统恢复到某一特定的历史。

三、例子

软件撤销操作、数据库的事务失败回滚

四、优缺点

优点：

·提供了一种状态恢复的实现机制，使得用户可以方便地回到一个特定的历史步骤，当新的状态无效或者存在问题时，可以使用先前存储起来的备忘录将状态复原。

·实现了信息的封装，一个备忘录对象是一种原发器对象的表示，不会被其他代码改动，这种模式简化了原发器对象，备忘录只保存原发器的状态，采用堆栈来存储备忘录对象可以实现多次撤销操作，可以通过在负责人中定义集合对象来存储多个备忘录。

缺点：

·资源消耗过大，如果类的成员变量太多，就不可避免占用大量的内存，而且每保存一次对象的状态都需要消耗内存资源，如果知道这一点大家就容易理解为什么一些提供了撤销功能的软件在运行时所需的内存和硬盘空间比较大了。

五、适用环境

·保存一个对象在某一个时刻的状态或部分状态，这样以后需要时它能够恢复到先前的状态。

·如果用一个接口来让其他对象得到这些状态，将会暴露对象的实现细节并破坏对象的封装性，一个对象不希望外界直接访问其内部状态，通过负责人可以间接访问其内部状态。

六、示例代码