1. 模式定义

桥接模式(Bridge Pattern)：将抽象部分与它的实现部分分离，使它们都可以独立地变化。

对象结构型模式，又称为柄体(Handle and Body)模式或接口(Interface)模式

用抽象关联取代了传统的多层继承

将类之间的静态继承关系转换为动态的对象组合关系



**二、**模式动机

1. 需要在抽象化和具体化之间增加更多的灵活性，避免在两个层次之间建立静态的继承关系
2. 抽象部分和实现部分可以以继承的方式独立扩展而互不影响
3. 一个类存在两个（或多个）独立变化的维度，且这两个（或多个）维度都需要独立地进行扩展
4. 不希望使用继承或因为多层继承导致系统类的个数急剧增加的系统

例如：

蜡笔：颜色和型号两个不同的变化维度（即两个不同的变化原因）耦合在一起，无论是对颜色进行扩展还是对型号进行扩展都势必会影响另一个维度

毛笔：颜色和型号实现了分离，增加新的颜色或者型号对另一方没有任何影响



画笔中存在的两个独立变化维度示意图

三、例子

不同的支付平台有不同的支付方式，如支付宝平台、微信平台支持人脸识别、密码支付、指纹支付。

Java虚拟机不同的应用可以跨不同的平台。

四、优缺点

优点：

* 分离抽象接口及其实现部分
* 可以取代多层继承方案，极大地减少了子类的个数
* 提高了系统的可扩展性，在两个变化维度中任意扩展一个维度，不需要修改原有系统，符合开闭原则

缺点：

* 会增加系统的理解与设计难度，由于关联关系建立在抽象层，要求开发者一开始就要针对抽象层进行设计与编程
* 正确识别出系统中两个独立变化的维度并不是一件容易的事情

五、示例代码