# **桥接模式**

功能用途：

构造型设计模式，通过使用封装，聚合，以及继承让不同的类承担不同的责任，将抽象(abstract)与行为(implementation)分离开来，从而保证各自的功能扩展。

角色和职责：

Abstraction：抽象类或接口 Car

维护对行为实现(Implementor)的引用。

Refined Abstraction：Suv, Bus

Abstration的子类

Implementor：Engine

行为实现类接口

Concrete Implementor: V6Engine,V8Engine,V12Engine

Implementor实现类

Client：MainTest

Bridge模式的使用者

优点：

抽象和实现的分离

优秀的扩展能力

实现细节对客户透明

UML：

