

qq 交流群：894294771

第十章 合成复用原则

合成复用原则是面向对象设计原则的 7 条原则中剩下的最后一条，下面我们将对其进行详细地介绍。

1 合成复用原则的定义

合成复用原则（Composite Reuse Principle，CRP）又叫组合/聚合复用原则（Composition/Aggregate Reuse Principle，CARP）。它要求在软件复用时，要尽量先使用组合或者聚合等关联关系来实现，其次才考虑使用继承关系来实现。

如果要使用继承关系，则必须严格遵循里氏替换原则。合成复用原则同里氏替换原则相辅相成，两者都是开闭原则的具体实现规范。

2 合成复用原则的重要性

通常类的复用分为继承复用和合成复用两种，继承复用虽然有简单和易实现的优点，但它也存在以下缺点：

- a) 继承复用破坏了类的封装性。因为继承会将父类的实现细节暴露给子类，父类对子类是透明的，所以这种复用又称为“白箱”复用。
- b) 子类与父类的耦合度高。父类的实现的任何改变都会导致子类的实现发生变化，这不利于类的扩展与维护。
- c) 它限制了复用的灵活性。从父类继承而来的实现是静态的，在编译时已经定义，所以在运行时不可能发生变化。

采用组合或聚合复用时，可以将已有对象纳入新对象中，使之成为新对象的一部分，新对象可以调用已有对象的功能，它有以下优点：

- a) 它维持了类的封装性。因为成分对象的内部细节是新对象看不见的，所以这种复用又称为“黑箱”复用。
- b) 新旧类之间的耦合度低。这种复用所需的依赖较少，新对象存取成分对象的唯一方法是通过成分对象的接口。
- c) 复用的灵活性高。这种复用可以在运行时动态进行，新对象可以动态地引用与成分对象类型相同的对象。

3 合成复用原则的实现方法

合成复用原则是通过将已有的对象纳入新对象中，作为新对象的成员对象来实现的，新对象可以调用已有对象的功能，从而达到复用。

下面以汽车分类管理程序为例来介绍合成复用原则的应用。

【例 1】汽车分类管理程序。

分析：汽车按“动力源”划分可分为汽油汽车、电动汽车等；按“颜色”划分可分为白色汽车、黑色汽车和红色汽车等。如果同时考虑这两种分类，其组合就很多。图 1 所示是用继承关系实现的汽车分类的类图。

qq 交流群：894294771

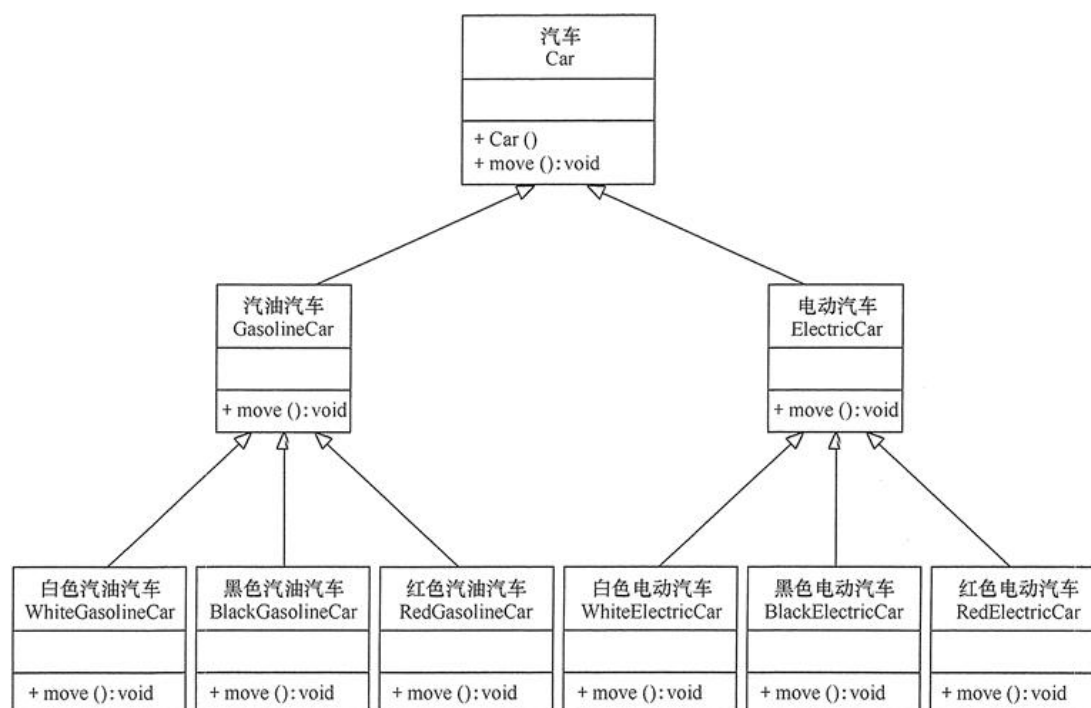


图 1 用继承关系实现的汽车分类的类图

从图 1 可以看出用继承关系实现会产生很多子类，而且增加新的“动力源”或者增加新的“颜色”都要修改源代码，这违背了开闭原则，显然不可取。但如果改用组合关系实现就能很好地解决以上问题，其类图如图 2 所示。

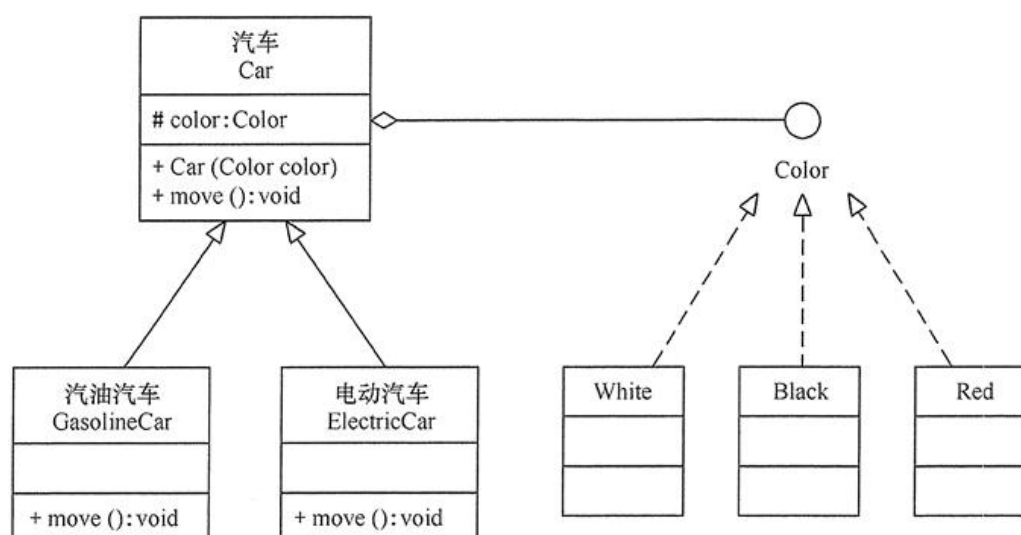


图 2 用组合关系实现的汽车分类的类图

结合前几节的内容，我们一共介绍了 7 种设计原则，它们分别为开闭原则、里氏替换原则、依赖倒置原则、单一职责原则、接口隔离原则、迪米特法则和本节所介绍的合成复用原则。

这 7 种设计原则是软件设计模式必须尽量遵循的原则，各种原则要求的侧重点不同。其中，**开闭原则是总纲**，它告诉我们要对扩展开放，对修改关闭；**里氏替换原则**告诉我们要破坏继承体系；**依赖倒置原则**告诉我们要面向接口编程；**单一职责原则**告诉我们实现类要职责单一；**接口隔离原则**告诉我们在设计接口的时候要精简单一；**迪米特法则**（最少知识）告诉我们要降低耦合度；**合成复用原则**告诉我们要优先使用组合或者聚合关系复用，少用继承关系复用。