

Bridge Pattern

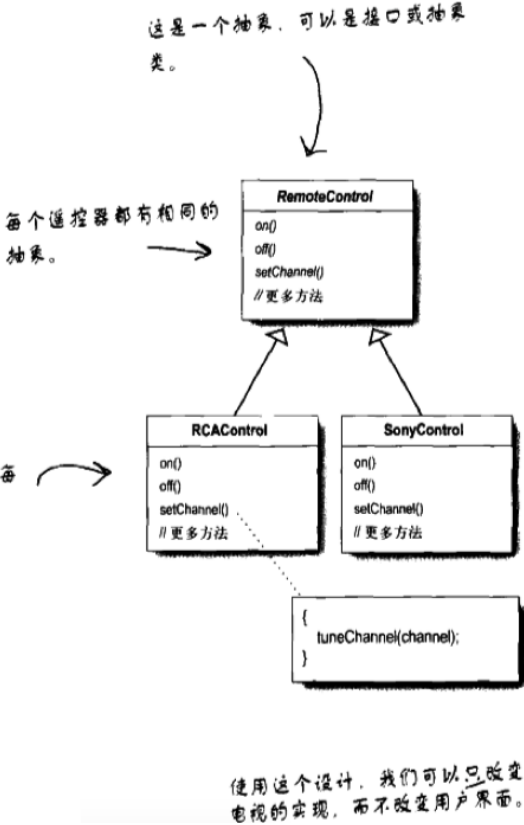
Decouple an abstraction from its implementation so that the two can vary independently.

场景

对电视，
开发遥控器

所有遥控器基于相同抽象

而对此抽象又做出许多不同实现



两难

都会改变

电视机
(实现)

抽象出界面

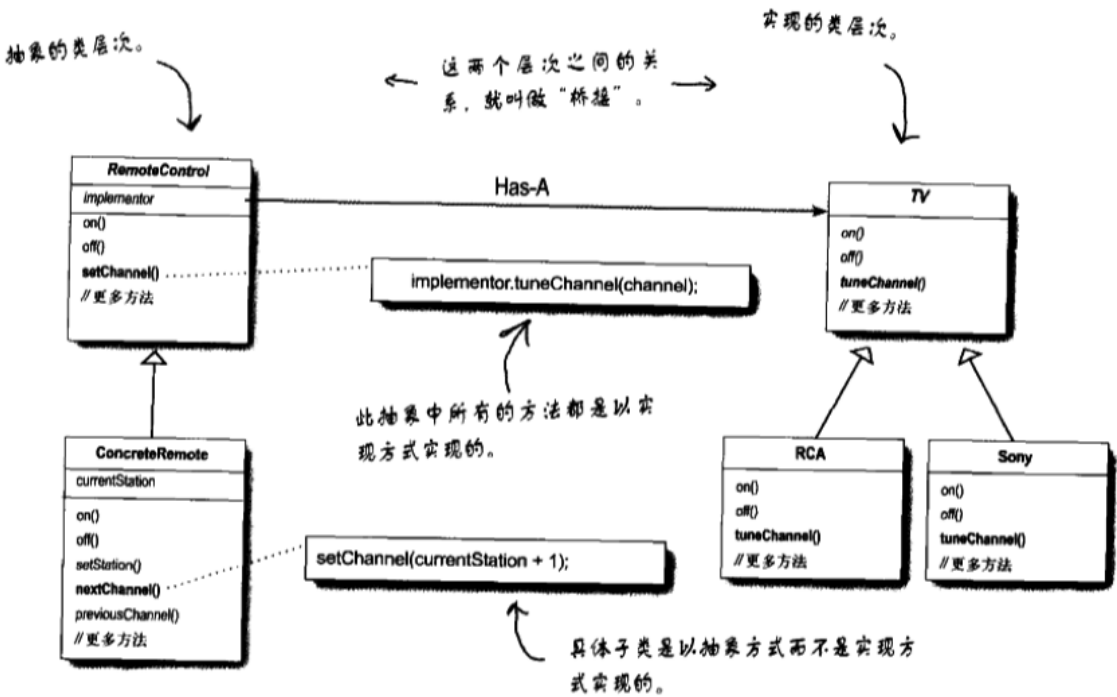
对不同对电视机，做不同的实现

遥控器
(抽象)

用户对界面提出想法

应对用户反馈，对抽象做出改变

实现和抽象
放在两个不同的类层次中，
使它们可以独立变化



优缺点
用途

优点

将实现解耦，它和实现不在永久绑定

抽象和实现，独立扩展，不影响对方

对ConcreteRemote改变，不回影响客户

缺点

增加类复杂度

用途

适合使用在跨多个平台对图像和窗口

当需要以不同的方式改变接口和实现时