桥接模式

也叫桥模式 桥梁模式

桥接模式 用于的场景：



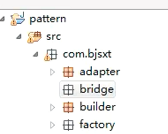
对电脑进行了分类

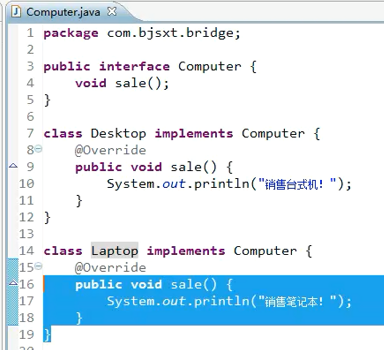
这个是一个多层次的继承结构

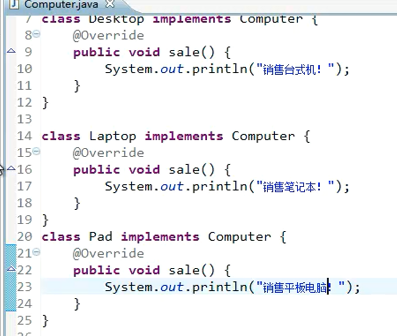
如果使用简单的继承来实现 是非常的繁琐

每次加一个品牌 就要在每一个种类下面加

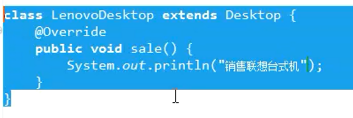
我们有没有办法去避免这个问题？



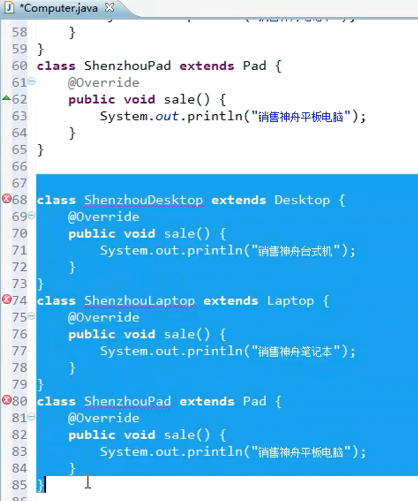




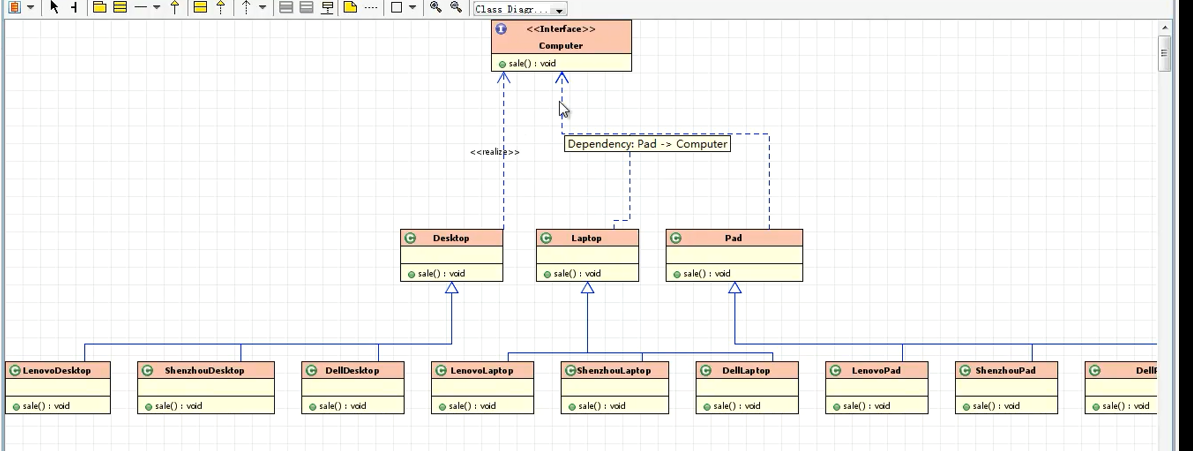
现在 我要增加相关的子类 比如 联想的台式机 笔记本 Pad



这样流程非常繁琐



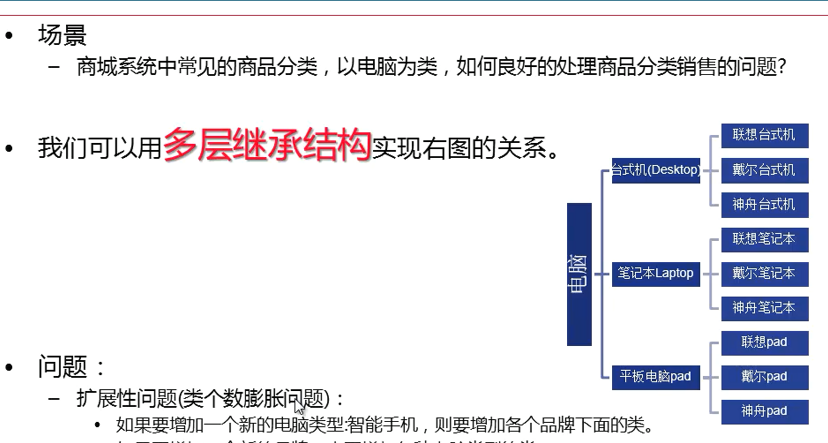
以后添加新的品种 就会非常麻烦



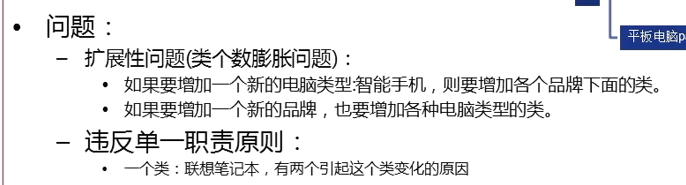
这样 类图非常复杂 繁琐

这种结构 按照现在套路 是非常有问题的

可扩展性有问题 非常繁琐

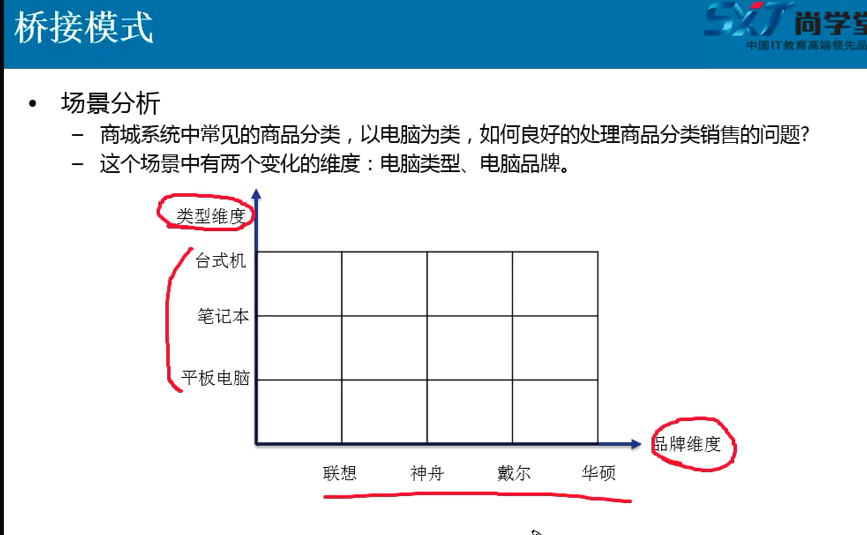


有没有办法避免这样的问题？



添加新类 要么曾艾电脑类型 要么增加品牌 原因就是  承担了品牌 和 类型 一个类有两个引起类变化的原因

现在干脆分开

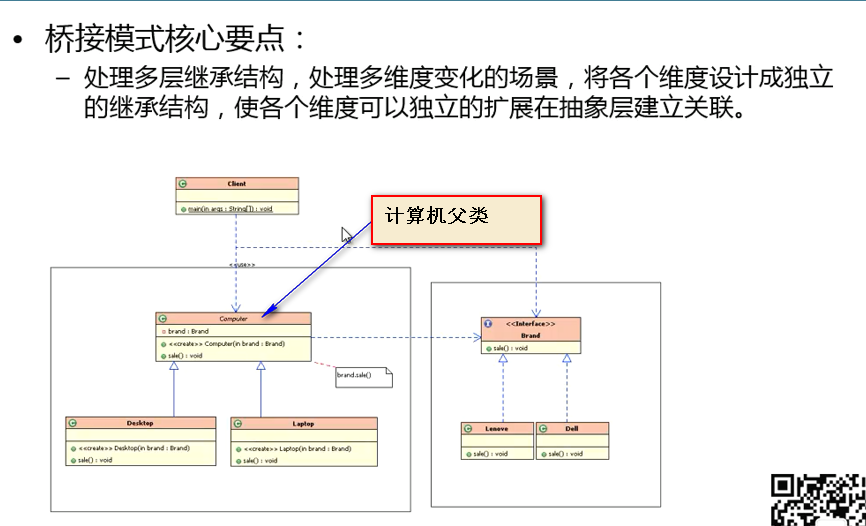


分开以后 类型维度 和 品牌维度两个不同的维度 通过某一个类 把这两个维度关联起来 这个就是桥梁模式

这样 每一个维度变化的时候 和 另一个维度没有关系

分析 有两种 或者多种变化的维度 可以考虑桥接模式

这样 你这个类UML会越来越复杂



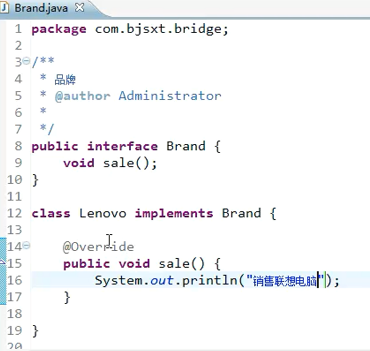
Computer里面有一个brand 持有了Brand接口的对象

这就是一个桥接模式

另一个寄存结构   
这个连接了Computer和Brand这两个类

现在扩展一个新的子类 Computer 因为是继承 就天然有了Brand这个属性

同样 Brand扩充 不影响Computer这个类

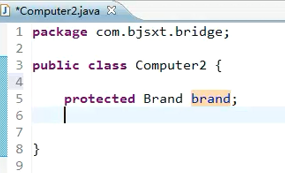
 我仅仅知道什么品牌 不知道什么类型

这样品牌的类层次结构就完事

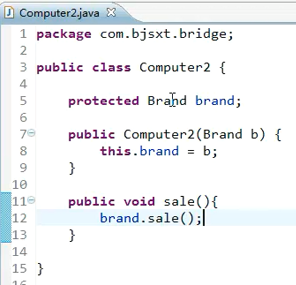
我们现在构建Computer类

这里不能使用接口 要使用抽象类 因为有一个公用的属性brand

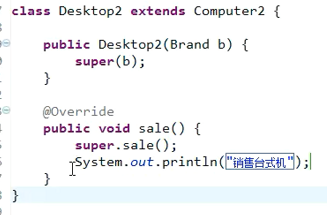
Protected 让子类使用



这样电脑就天然具备了品牌



现在定义我们的桌面类



这样就是另一个维度的结构 --- 电脑类型

这样品牌什么的都写好了

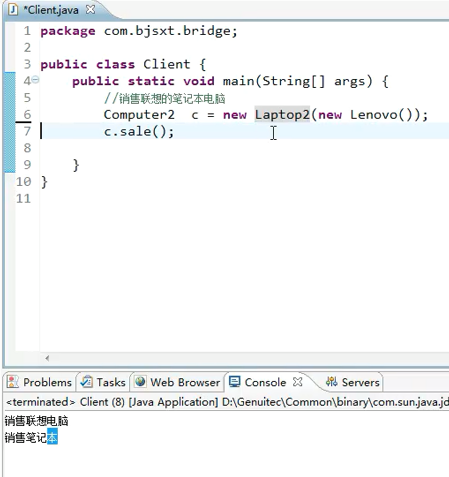
这样关系梳理好了

现在新加品牌 或者类型 各自添加即可

然后做一个测试Client

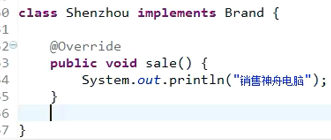
现在 我要销售联想电脑的笔记本电脑

 用组合代替了继承关系

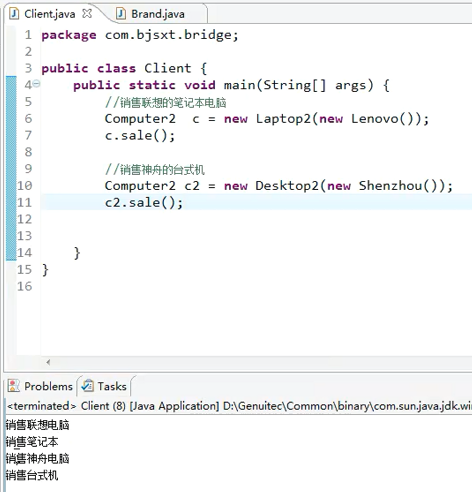


现在新家品牌

神州



添加一个类别

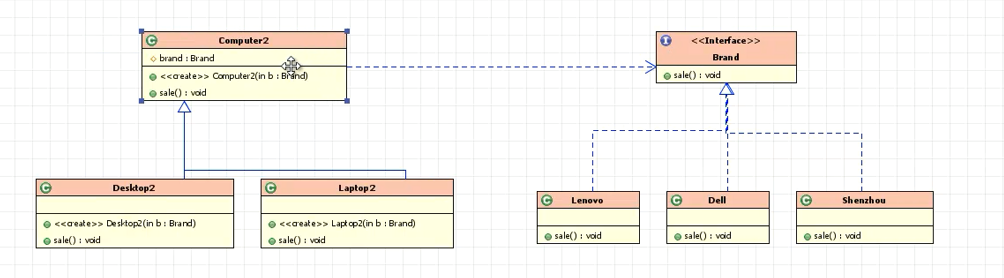


【分工之后 聚合即可 --- 桥接模式】

这样就达成了一样的效果

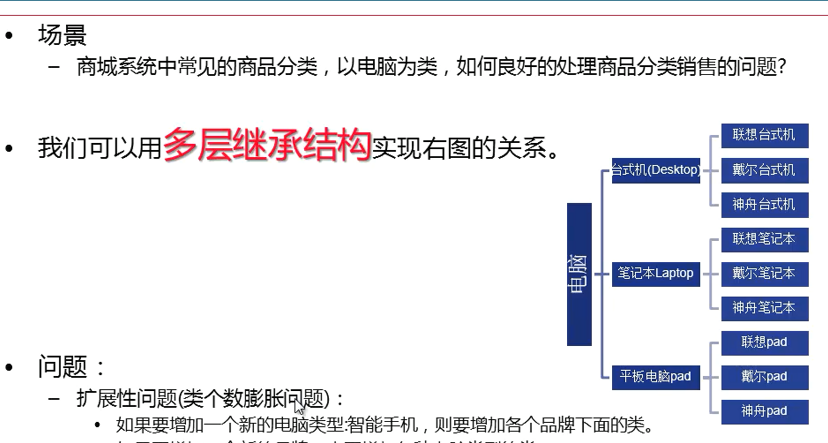
后期的扩展非常方便

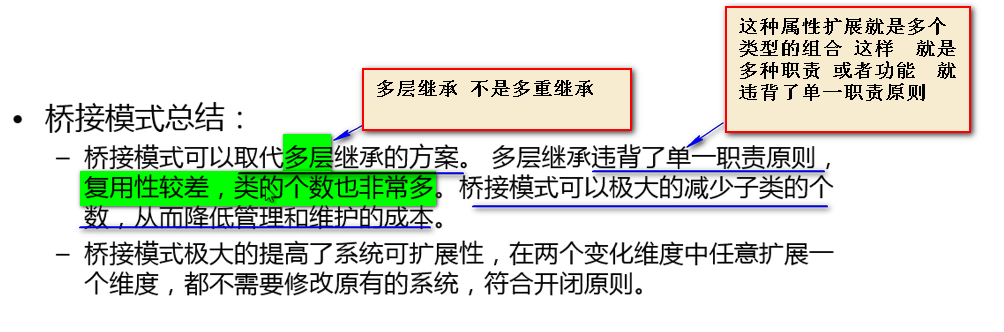
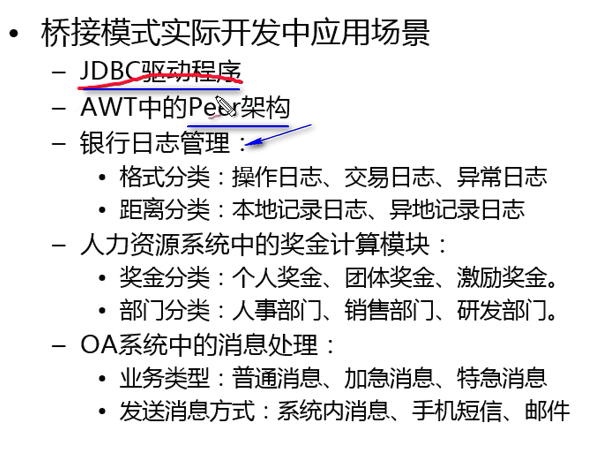
顺便把这个图画一下



这个是桥接模式的

Java中没有多重继承 这样 只能树形结构



银行日志 格式分类和距离分类 两个维度 两个变化的维度 就非常适合我们的桥接模式

HR部门奖金的计算 ----- 这个时候 我们可以使用桥接模式来做 这也是两个变化的维度 也要使用桥接模式 否则类的结构非常多

OA系统的消息处理 有两类 业务类型 和 消息发送方式 两个维度

总之 只要是具备多个变化维度 ---- 桥接模式就可以派上用场