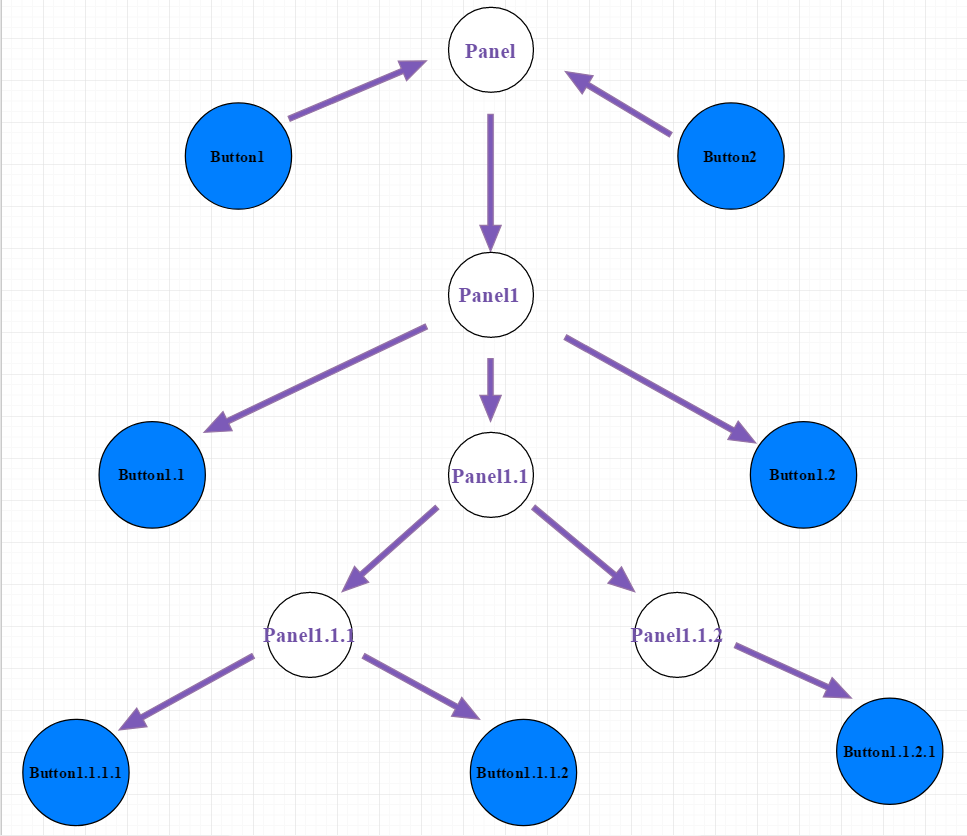
1. 定义

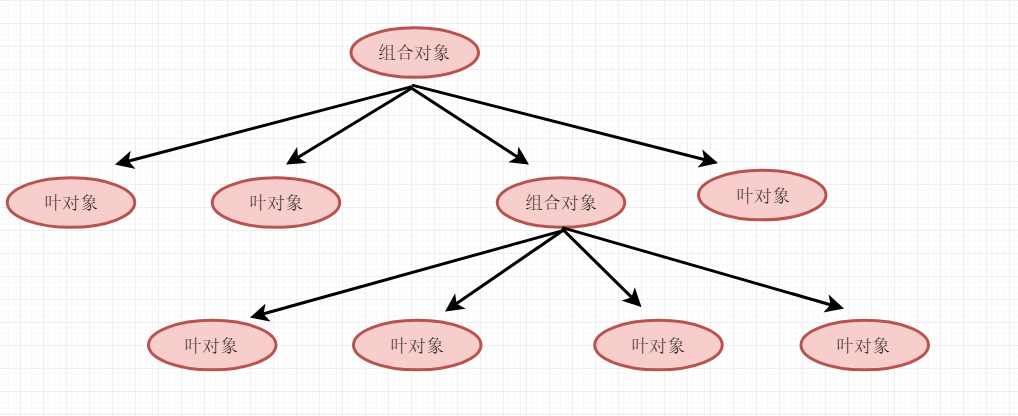
表示整体与部分的关系的数据结构，类似于树形结构。例如容器里可以包含组件，比如Panel面板中可以包含Button组件和子面板Panel,子Panel中同样又可以包含Panel和Button



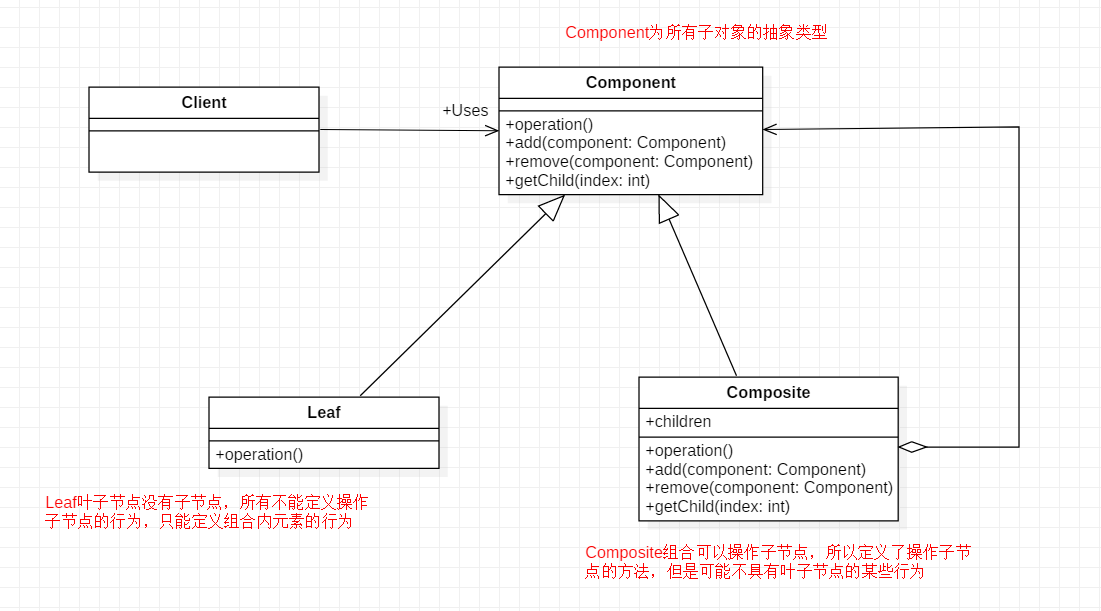
组合模式的出现就是为了让我们以一致的方式来处理个别对象亦或组合对象，而不必区分对象类型。

也就是说我们要对树上的节点和叶子进行操作时，它能够提供一种一致的方式，不必纠结他是叶子类型还是节点类型。

当我们的案例牵扯到树形结构或者类似的”部分-整体”关系时候，就可以使用组合模式。



1. 组合模式的UML类图



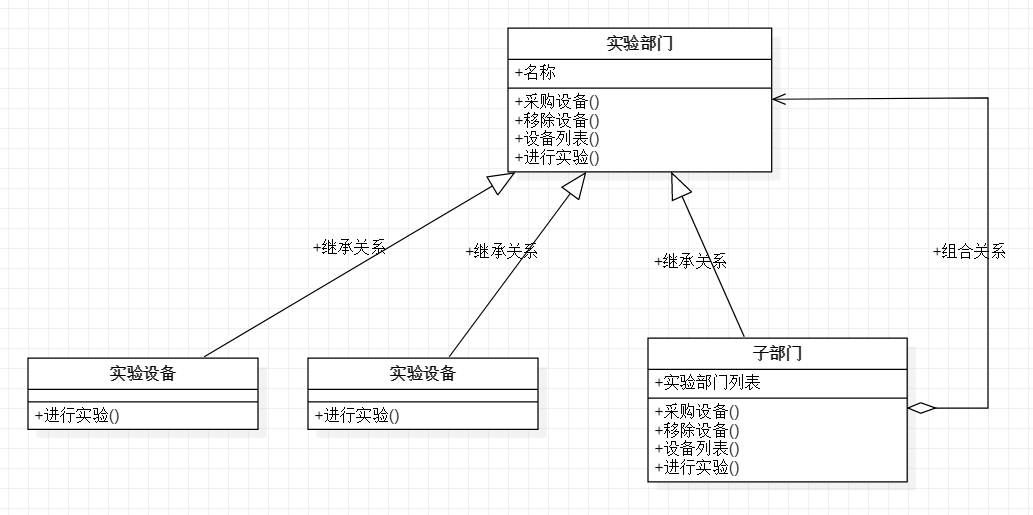
Component 为容器和内容的抽象类型，除了最基本的add,remove,getChild方法外，还可以定义其他组件公共的方法，比如getSize,pringList等。Leaf为叶子节点对象，该对象由于没有叶子节点，所以不能定义类似add这种针对容器的方法，但是可以定义叶子节点自身行为的一些方法，比如getName等;Composite为容器类，一般包含容器元素Children、操作容器的部分方法比如给容器添加元素的add方法。

1. 组成角色

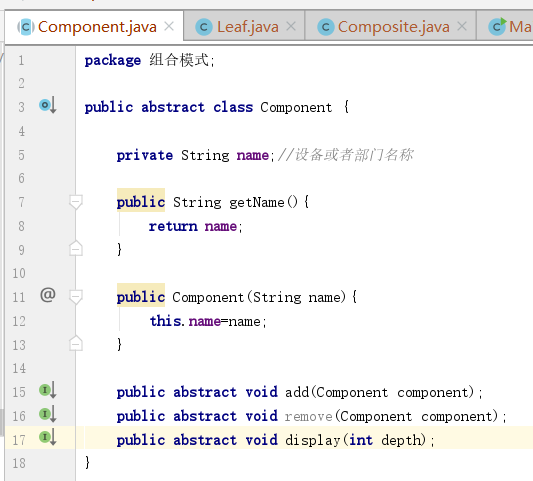


1. 使用实例

我们有很多实验部门，每个部门地下会管理多台实验设备，每个部门底下又会存在多个子部门。



抽象构件,用来定义组件共同的方法和属性(设备和部门的抽象)



叶子结构



复合构件



Client

