

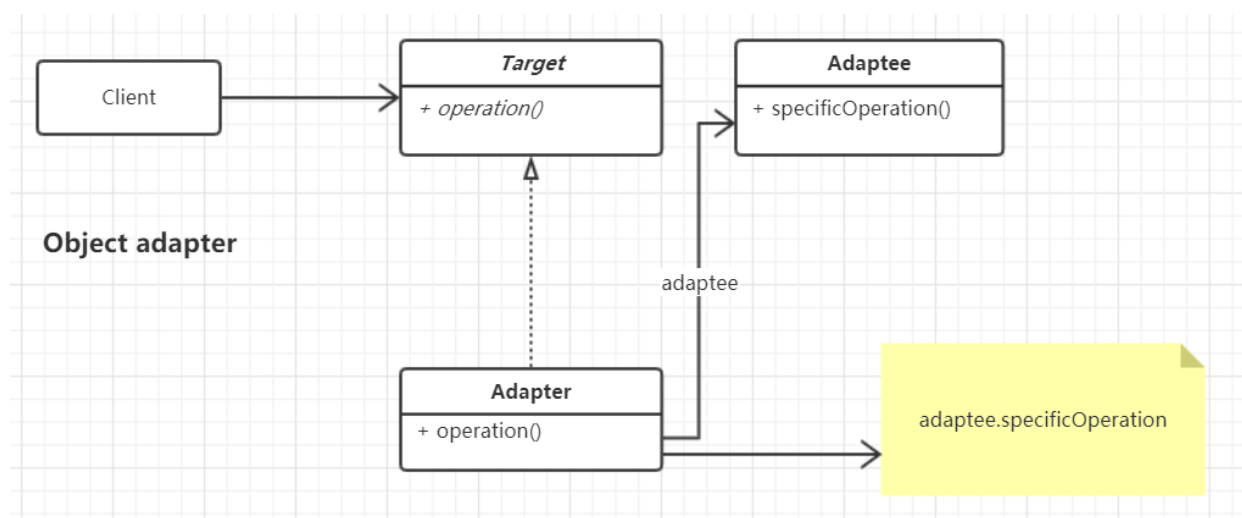
# 适配器模式

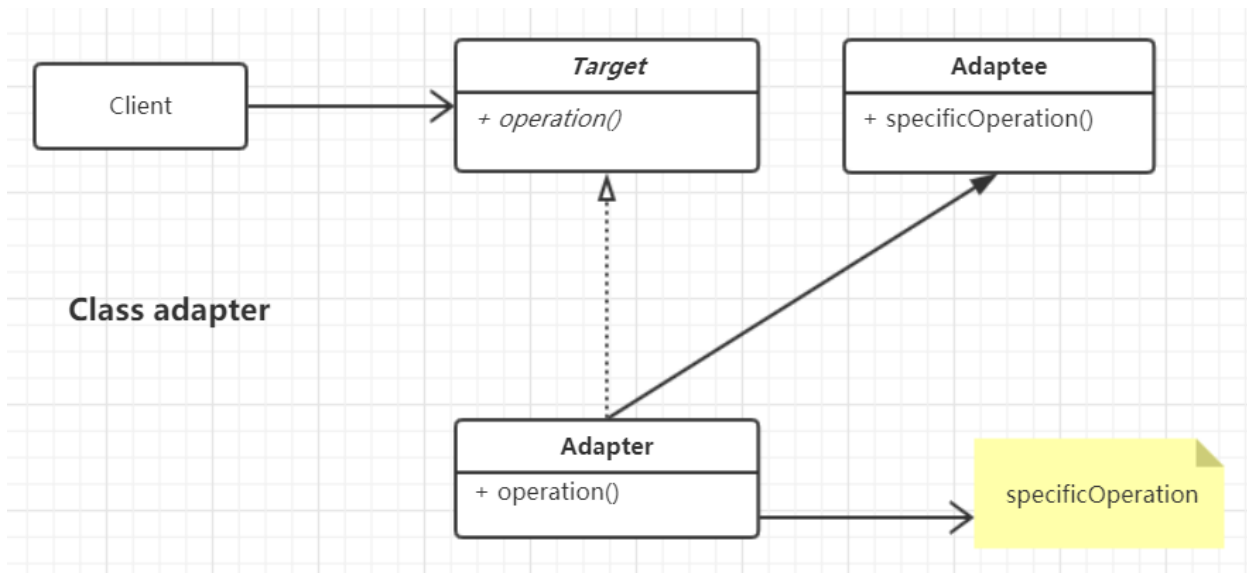
## Adapter

郭嘉

模式定义：

将一个类的接口转换成客户希望的另一个接口。Adapter模式使得原本由于接口不兼容而不能一起工作的那些类可以一起工作





## 案例

### 对象适配器模式

```
1 package com.tuling.designpattern.adapter.v1;
2
3 /**
4  * @author 腾讯课堂-图灵学院 郭嘉
5  * @Slogan 致敬大师，致敬未来的你
6  */
7 public class AdapterTest1 {
8     public static void main(String[] args) {
9         Adaptee adaptee=new Adaptee();
10
11         Target target = new Adapter( adaptee );
12         target.output5v();
13     }
14 }
15
16 class Adaptee{
17     public int output220v(){
18         return 220;
19     }
20 }
21 interface Target{
22     int output5v();
23 }
```

```

24 // Object Adapter
25 class Adapter implements Target{
26
27     private Adaptee adaptee;
28
29     public Adapter(Adaptee adaptee) {
30         this.adaptee=adaptee;
31     }
32
33     @Override
34     public int output5v() {
35         int i=adaptee.output220v();
36         // .....
37         System.out.println(String.format( "原始电压:  %d v - > 输出电压:  %d v
",i,5 ));
38
39         return 5;
40     }
41 }
42
43
44

```

## 类适配器模式

```

1 package com.tuling.designpattern.adapter.v2;
2
3 /**
4  * @author 腾讯课堂-图灵学院 郭嘉
5  * @Slogan 致敬大师，致敬未来的你
6  */
7 public class AdapterTest2 {
8     public static void main(String[] args) {
9         Adpater adpater=new Adpater();
10        adpater.output5v();
11
12    }
13 }
14
15 class Adaptee{
16     public int output220v(){

```

```

17     return 220;
18 }
19 }
20 interface Target {
21     int output5v();
22 }
23 // Class Adapter
24 class Adapter extends Adaptee implements Target{
25
26     @Override
27     public int output5v() {
28         int i= output220v();
29         // .....
30         System.out.println(String.format( "原始电压:  %d v - > 输出电压:  %d v
",i,5 ));
31         return 5;
32     }
33 }

```

## 应用场景

- 1.当你希望使用某些现有类，但其接口与您的其他代码不兼容时，请使用适配器类。
- 2.当你希望重用几个现有的子类，这些子类缺少一些不能添加到超类中的公共功能时，请使用该模式。

## 优点：

1. 符合单一职责原则
2. 符合开闭原则

## JDK & Spring源码中的应用

```
1 JDK:  
2 java.util.Arrays#asList()  
3 java.util.Collections#list()  
4 Spring:  
5 org.springframework.context.event.GenericApplicationListenerAdapter
```