

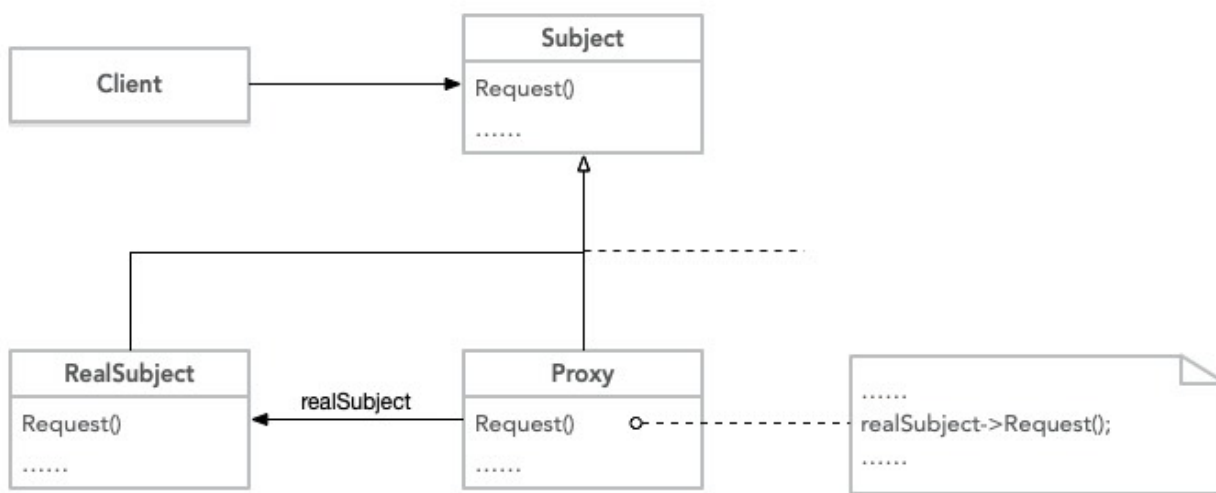
PHP设计模式之代理模式

代理人这个职业在中国有另外一个称呼，房产经济人、保险经济人，其实这个职业在国外都是叫做房产代理或者保险代理。顾名思义，就是由他们来帮我们处理这些对我们大部分人来说都比较生疏的专业领域的工作。代理模式也是一样的道理，同时，在这篇文章中还会简单的介绍正向代理和反向代理是怎么回事。

Gof类图及解释

GoF定义：为其它对象提供一种代理以控制对这个对象的访问

GoF类图



代码实现

```
1 interface Subject
2 {
3     public function Request();
4 }
5
6 class RealSubject implements Subject
7 {
8     function Request()
9     {
10         echo "真实的操作", PHP_EOL;
11     }
12 }
13
14 class Proxy implements Subject
```

```

15 {
16     private $realSubject;
17
18     public function __construct()
19     {
20         $this->realSubject = new RealSubject();
21     }
22
23     public function Request()
24     {
25         echo "代理的操作", PHP_EOL;
26         $this->realSubject->Request();
27     }
28 }
29
30 $proxy = new Proxy();
31 $proxy->Request();

```

- 代理模式的实现其实非常简单，或许你在不经意间经常会用到
- 请注意代理模式与装饰器、适配器的区别，另外，模板方法模式也和它很像
- 装饰器，一般是对对象进行装饰，其中的方法行为会有增加，以修饰对象为主
- 适配器，一般会改变方法行为，目的是保持接口的统一但得到不同的实现
- 模板方法模式，我们后面会讲，这里只要知道，模板方法是在抽象类中去组合调用子类的方法
- 代理模式有几种形式：远程代理（例如：第三方接口SDK）、虚代理（例如：异步加载图片）、保护代理&智能指引（例如：权限保护），而我们代码实现的最普通的代理，其实就是让代理类来代替真实类的操作

**

完整代码： <https://github.com/zhangyue0503/designpatterns-php/blob/master/12.proxy/source/proxy.php>

实例

短信功能不能停，这回用简单的代理模式来增加一些前后的提示吧！

短信发送类图

完整源码： <https://github.com/zhangyue0503/designpatterns-php/blob/master/12.proxy/source/proxy-msg.php>

```

1  <?php
2
3  interface SendMessage
4  {
5      public function Send();
6  }
7
8  class RealSendMessage implements SendMessage
9  {
10     public function Send()
11     {
12         echo '短信发送中...', PHP_EOL;
13     }
14 }
15
16 class ProxySendMessage implements SendMessage
17 {
18     private $realSendMessage;
19
20     public function __construct($realSendMessage)
21     {
22         $this->realSendMessage = $realSendMessage;
23     }
24
25     public function Send()
26     {
27         echo '短信开始发送', PHP_EOL;
28         $this->realSendMessage->Send();
29         echo '短信结束发送', PHP_EOL;
30     }
31 }
32
33 $sendMessage = new ProxySendMessage(new RealSendMessage());
34 $sendMessage->Send();

```

说明

- 例子非常简单，还是普通代理的实现，这里我们讲下正向代理和反向代理

- 通常我们所说的科学上网（fanqiang），就是正向代理，由我们使用软件或者自己配置代理网关上网，实际就是我们把请求发送到指定的网关，再由这个网关代替我们去访问其他网站，这种由我们指定选择的代理就是正向代理
- PHPer们大多还是比较清楚反向代理的，毕竟现在Nginx已经代替Apache成为了PHP标配了。当我们访问一个网站的时候，某些路径或者域名并不一定是在这台服务器上，他们在服务器上直接代理到了其他的服务器甚至是别人家的站点。对于这个我们这些浏览网站的人是不知道的，这种我们不知道就被莫名其妙代理了的情况就是反向代理，一般在服务后台运维中是必备知识！