

Please visit website: <http://cxyroad.com>

接口隔离原则，到底什么需要隔离？

=====

你好，我是猿java。

在 [单一职责到底是什么？十分钟带你掌握!](http://cxyroad.com/https://link.juejin.cn/?target=https%3A%2F%2Fmp.weixin.qq.com%2Fs%3F__biz%3DMzlwNDAYOTI2Nw%3D%3D%26mid%3D2247497957%26idx%3D1%26sn%3D1027f3881f3f95c5ed1a3e17d56ab991%26chksm%3D96c4d4d9a1b35dcf9092dd80b0068e446d26b97151332ca07dd53f8fa163cd7cf51f296c228e%26token%3D2051507942%26lang%3Dzh_CN%23rd) 这篇文章中，我们分析了单一职责原则，今天要分析的接口隔离原则应该是 SOLID原则中最简单的一种，但是在实际工作中，发现很多人容易混淆两种原则，因此，今天我们就来详细的聊一聊。

什么是接口隔离？

=====

接口隔离原则，Interface segregation principle (ISP)，也是 Robert C. Martin提出的 SOLID原则中的一种，老规矩，还是先看看作者 Robert C. Martin 对接口隔离原则是如何定义的：

...

Clients should not be forced to depend upon interfaces that they do not use.

...

在作者对接口隔离原则的定义中强调：不应强迫客户依赖他们不使用的接口。

在 Java中，我们一直都强调要面向接口编程，足以看出接口在 Java中的重要性。其实，与单一职责原则类似，接口隔离原则的目标是通过将软件拆分为多个独立的部分来减少所需更改的副作用和频率。

这里的“不应强迫”该如何理解？通常来讲“不应强迫”有2种理解：

1. 第一种理解是用户不能被强迫使用整个接口。
2. 第二种理解是用户只使用接口中的部分方法，其余的方法不能被强迫使用。

显然，第二种理解比较合理，所以接口隔离原则可以更直白一点的表达成：在接口中，不要放置接口使用者不需要的方法。

站在接口使用者的角度，这样的设计更加人性化，为什么要增加一些我不需要的依赖负担呢？

如何实现接口隔离？

=====

假如有一个业务场景，需要定义一个交通工具的 Transportation类，类中包含设置基本信息（价格，颜色），启停以及飞行等方法：

```
...  
public interface Transportation{  
    void setPrice(double price);  
    void setColor(String color);  
    void start();  
    void stop();  
    void fly();  
}
```

...

汽车属于一种交通工具，因此我们可以定义一个 Car类去实现 Transportation类，代码如下：

...

```
public class Car implements Transportation {  
    @Override  
    public void setPrice(double price) {  
        // 价格设置逻辑  
    }  
    @Override  
    public void setColor(String color) {  
        // 颜色设置逻辑  
    }  
    @Override
```

```

    public void start(){
        // 启动逻辑
    }
    @Override
    public void stop(){
        // 停止逻辑
    }
    @Override
    public void fly(){
        // 飞行逻辑
    }
}

```

...

从上面的代码可以发现一个问题：Car不能飞行却要实现 fly()方法，为什么？显然 fly()这个方法是 Car这种交通工具不需要的，这就违反了接口隔离原则。

如何解决这个问题呢？

首先，我们将交通工具接口分成多个角色接口，每个角色接口用于特定的行为，在这里我们可以将 Transportation分成 BasicFeature、Movable、Flyable 三类行为接口。

...

```

// 基本属性， 价格， 颜色
public interface BasicFeature{
    void setPrice(double price);
    void setColor(String color);
}

```

```

// Movable 行为， 行驶和停止
public interface Movable {
    void start();
    void stop();
}

```

```

// 飞行 行为
public interface Flyable {
    void fly();
}

```

...

而 Car只需要基本属性和 Movable行为，代码如下：

...

```
public class Car implements BasicFeature, Movable {
    @Override
    public void setPrice(double price) {
        // 价格设置逻辑
    }
    @Override
    public void setColor(String color) {
        // 颜色设置逻辑
    }
    @Override
    public void start(){
        // 启动逻辑
    }
    @Override
    public void stop(){
        // 停止逻辑
    }
}
```

...

Airplane飞机需要基本属性，Movable行为和飞行行为，代码如下：

...

```
public class Airplane implements BasicCFeature, Movable, Flyable {
    @Override
    public void setPrice(double price) {
        // 价格设置逻辑
    }
    @Override
    public void setColor(String color) {
        // 颜色设置逻辑
    }
    @Override
    public void start(){
        // 启动逻辑
    }
    @Override
    public void stop(){
        // 停止逻辑
    }
}
```

```
@Override
public void fly(){
    // 飞行逻辑
}
}
...

```

通过上面的拆解，我们可以看到每种交通工具只需要自己需要的接口就好了，自己不需要的接口就不会被强迫，更加不会造成 Car能 fly()这样不常见的误区。

接口隔离和单一职责的比较

=====

接口隔离原则和单一职责原则都是 SOLID设计原则中的重要组成部分，虽然它们有一些相似之处，但它们的重点和应用的范围有所不同，在实际开发中，很容易搞混淆，因此，这里对这两个原则做详细比较。

1. 点不同

单一职责原则（SRP）：类的职责划分，确保每个类只有为一类行为负责，它主要解决的是类内部职责过多导致的复杂性问题。

接口隔离原则（ISP）：接口的设计，确保客户端只依赖于它们实际需要的方法。它主要解决的是接口过于庞大导致的依赖问题。

2. 作用范围不同

单一职责原则（SRP）：作用于类的设计和实现层面，通过分离职责提高类的内聚性。

接口隔离原则（ISP）：作用于接口的设计层面，通过细化接口减少客户端的依赖，提高系统的灵活性。

3. 实现方法不同

单一职责原则（SRP）：通过将一个类的多种职责分离成多个独立的类来实现。

接口隔离原则（ISP）：通过将一个大的接口分解为多个小接口，让不同的客户端依赖于不同的小接口来实现。

因此，接口隔离原则是在遵守单一职责原则的前提下，将接口更加细化。

总结

==

接口隔离可以提高代码的可读性、可维护性和灵活性，减少系统的耦合度，在实际开发中，合理应用接口隔离原则，可以帮助我们创建高质量的代码和系统

。然而，在应用时需要注意适度细化和明确职责，避免过度设计和接口混乱。

学习交流

=====

文章总结不易，感谢帮忙点赞，收藏，或者公众号：猿java，持续为你呈现更多干货！

原创好文

=====

* [单一职责到底是什么？十分钟带你掌握!](<http://cxyroad.com/>
”https://link.juejin.cn?target=https%3A%2F%2Fmp.weixin.qq.com%2Fs%3F__biz%3DMzlwNDAYOTI2Nw%3D%3D%26mid%3D2247497957%26idx%3D1%26sn%3D1027f3881f3f95c5ed1a3e17d56ab991%26chksm%3D96c4d4d9a1b35dcf9092dd80b0068e446d26b97151332ca07dd53f8fa163cd7cf51f296c228e%26token%3D2051507942%26lang%3Dzh_CN%23rd”)

* [Google二面：聊聊系统设计思路！（推荐阅读）](<http://cxyroad.com/>
”https://link.juejin.cn?target=https%3A%2F%2Fmp.weixin.qq.com%2Fs%3F__biz%3DMzlwNDAYOTI2Nw%3D%3D%26mid%3D2247497924%26idx%3D1%26sn%3D3753221633c74869366ccd5079146432%26chksm%3D96c4d4f8a1b35deec7fa79d0fbac4c2dd934da3b446fdefc4b1f80aebf7a32ce78db0b62749%26token%3D356778831%26lang%3Dzh_CN%23rd”)

* [美团一面：Git 是如何工作的？（推荐阅读）](<http://cxyroad.com/>
”https://link.juejin.cn?target=https%3A%2F%2Fmp.weixin.qq.com%2Fs%3F__biz%3DMzlwNDAYOTI2Nw%3D%3D%26mid%3D2247491037%26idx%3D1%26sn%3D2351e0e54dcbf16a2575e353ed41734d%26chksm%3D96c731e1a1b0b8f73027d20dc84e60760a195d46e106fcbddf98a945d125c453990f805a2769%26token%3D94106720%26lang%3Dzh_CN%23rd”)

* [阿里 P7二面：Redis 执行 Lua，能保证原子性吗？](<http://cxyroad.com/>
”https://link.juejin.cn?target=https%3A%2F%2Fmp.weixin.qq.com%2Fs%3F__biz%3DMzlwNDAYOTI2Nw%3D%3D%26mid%3D2247485640%26idx%3D1%26sn%3D68b6a1bf4873a394b7a903d6a15ba697%26chksm%3D96c724f4a1b0ade21c40da5b73dbf36851de848ed7b3d0af203ef7fead81cb850e3242351402%26token%3D189563748%26lang%3Dzh_CN%23rd”)

* [当下环境，程序员需要修炼的 3项技能](<http://cxyroad.com/>
”https://link.juejin.cn?target=https%3A%2F%2Fmp.weixin.qq.com%2Fs%3F__biz%3DMzlwNDAYOTI2Nw%3D%3D%26mid%3D2247485548%26idx%3D1%26sn%3D9fbabe3280e88810859975f44140e15d%26chksm%3D96c72450a1b0ad46a0741e0f4369424fda76d8906750ef87e58839ce9”)

7ffd0381c84c4492f6d%26token%3D595465147%26lang%3Dzh_CN%23rd”)

* [AI是打工人的下一个就业风口吗?](<http://cxyroad.com/>
”[* \[和斯坦福博士写代码的一个月\]\(<http://cxyroad.com/>
”\[* \\[肝了一周，这下彻底把 MySQL 的锁搞懂了\\]\\(<http://cxyroad.com/>
”\\[原文链接: <https://juejin.cn/post/7385776238788427785>\\]\\(https://link.juejin.cn?target=https%3A%2F%2Fmp.weixin.qq.com%2Fs%3F__biz%3DMzlwNDAYOTI2Nw%3D%3D%26mid%3D2247484561%26idx%3D1%26sn%3D2d021a42fd74163367b121eb0b98b53f%26chksm%3D96c728ada1b0a1bb7660c3918a684b288405da848bf9602dfaaaa8c6554d6c3148d864ee4f75%26token%3D595465147%26lang%3Dzh_CN%23rd”\\)</p></div><div data-bbox=\\)\]\(https://link.juejin.cn?target=https%3A%2F%2Fmp.weixin.qq.com%2Fs%3F__biz%3DMzlwNDAYOTI2Nw%3D%3D%26mid%3D2247485310%26idx%3D1%26sn%3D98111850b5aab60f129b2414f355fef6%26chksm%3D96c72b42a1b0a254ae6a6bc4ffafbc9a11b6b93a22044fd51ce647b9aef618b89e710d1eb59d%26token%3D595465147%26lang%3Dzh_CN%23rd”\)</p></div><div data-bbox=\)](https://link.juejin.cn?target=https%3A%2F%2Fmp.weixin.qq.com%2Fs%3F__biz%3DMzlwNDAYOTI2Nw%3D%3D%26mid%3D2247485571%26idx%3D1%26sn%3De00066ca6bd2f5df2bcf0fe2b5a1f340%26chksm%3D96c724bfa1b0ada9ae2b3116cf0805f48edd2aef81c9e4ebfca386a8014cc81d800cc21283fe%26token%3D595465147%26lang%3Dzh_CN%23rd”)</p></div><div data-bbox=)