COMP3027J
Software Architecture
Modifiability and its Tactics

DENG, YONGJIAN

Faculty of Computer Science, BJUT

Data Mining & Security Lab (DMS Lab)

Outline

1. The Meaning of modifiability

2. Tactics to Improve Modifiability

Outline

1. The Meaning of modifiability

2. Tactics to Improve Modifiability

The Meaning of Modifiability

Concerns

- Cost of Modification
- Which parts of the system are modified
- When the modification occurs
- Who performs the modification

The Meaning of Modifiability

Measurement Metrics

- Time Taken to Complete Modification
- Human Resource Cost of Modification
- Economic Cost of Modification
- … …

The Meaning of Modifiability - Scenarios

Source of Stimulus

- Who Performs the Modification (Developers/Administrators/Users)

Stimulus

- Specific Modifications to Be Made

The Meaning of Modifiability - Scenarios

Artifacts

- Modifying the system's functionality or UI or other systems for interaction?

Environment

- When does the modification perform? During design or development or operation?
- The later the modification, the more disadvantageous it is

The Meaning of Modifiability - Scenarios

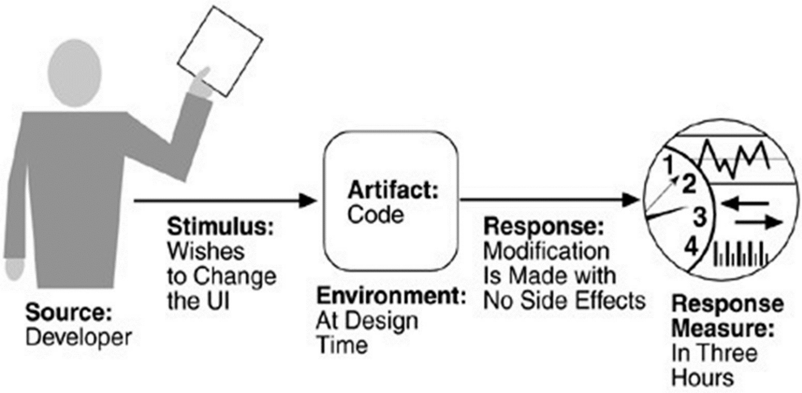
Response

- Operators must understand how to modify, perform modification operations, test, and deploy

Response Measure

- Time & Cost

Modifiability Scenario Example



Source: Developer

Stimulus: Wishes to Change the UI

Artifact: Code

Environment: At Design Time

Response: Modification Is Made with No Side Effects

Response Measure: In Three Hours

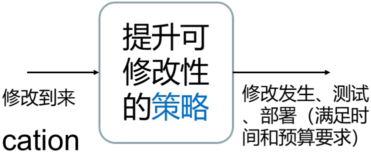
Outline

1. The Meaning of modifiability

2. Tactics to Improve Modifiability

Tactics to Improve Modifiability

-Overview



Target

Reduce the time and cost of modification

Direction 1: Limit the Scope of Modification

Keep the software scope affected by modification as small as possible

Direction 2: Delay Binding Time

Allow software to be flexibly modified during running

Tactics to Improve Modifiability

-Limiting the Scope

High cohesion, low coupling in modules

- Try to control the modification of the program within a single module
- Can leverage frameworks, middleware

Consider potential modifications

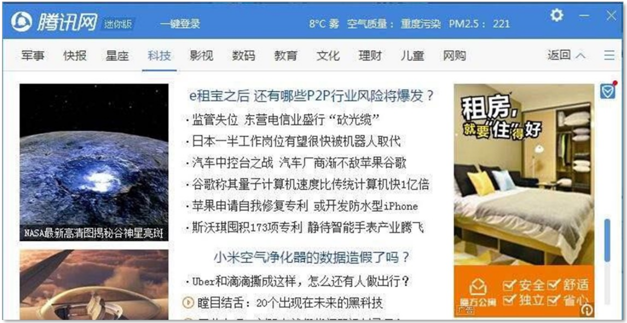
- Helps assess the division of responsibilities between modules
- Ensure that a modification at one point only affects one module
- Avoid unrelated multiple modifications affecting the same module

Tactics to Improve Modifiability

-Limiting the Scope

Make modules generic

- "Interpreter" style approach



腾讯网 迷你版
一键登录
8°C 雾 空气质量：重度污染 PM2.5：221
军事 快报 星座 科技 影视 数码 教育 文化 理财 儿童 网购 返回
NASA最新高清图揭秘谷神星亮斑
租房，就要"住"得好
☑安全☑舒适 ☑独立☑省心
广告

e租宝之后 还有哪些P2P行业风险将爆发？

- 监管失位 东营电信业盛行"砍光缆"
- 日本一半工作岗位有望很快被机器人取代
- 汽车中控台之战 汽车厂商渐不敌苹果谷歌
- 谷歌称其量子计算机速度比传统计算机快1亿倍
- 苹果申请自我修复专利 或开发防水型iPhone
- 斯沃琪囤积173项专利 静待智能手表产业腾飞

小米空气净化器的数据造假了吗？

- Uber和滴滴撕成这样，怎么还有人做出行？

瞠目结舌：20个出现在未来的黑科技

Tactics to Improve Modifiability

-Limiting the Scope

Hide information

- Accessibility in object-oriented mechanisms (public/private)

Maintain a consistent interface

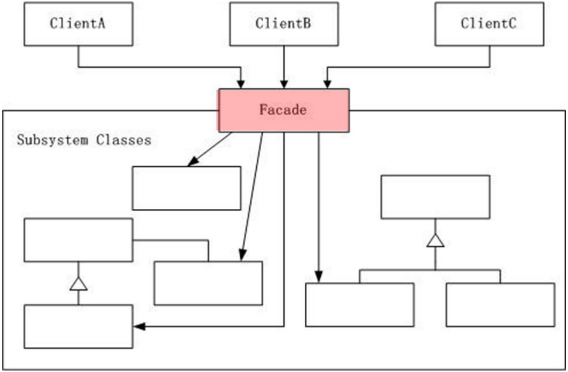
- Allow independent changes on both sides of the interface without changing the interface itself

Tactics to Improve Modifiability

-Limiting the Scope

Limit communication paths

- Façade pattern in design patterns



Tactics to Improve Modifiability

-Limiting the Scope

Use intermediaries

- Data intermediaries: Style for sharing data
- Service intermediaries: Design patterns such as bridge, factory method, etc.

Tactics to Improve Modifiability

-Limiting the Scope

Name server

- Query the location of required resources/objects, resolve location dependencies

Create instances on demand

- Leverage creational patterns in design patterns

Tactics to Improve Modifiability

-Delaying Binding Time

Configuration files

- Modify configuration files without modifying code

```
<?xml version="1.0" encoding="utf-8"?>
<!-- The configuration file for SMSvcHost.exe -->
<configuration>
   <runtime>
      <gcConcurrent enabled="false" />
   </runtime>
   <system.serviceModel>
      <!-- SMSvcHost ETW traces are redirected by
default to an etwProviderId different from WCF's default.
-->
      <diagnostics performanceCounters="Off"
etwProviderId="{f18839f5-27ff-4e66-bd2d-
639b768cf18b}"/>
   </system.serviceModel>
</configuration>
```

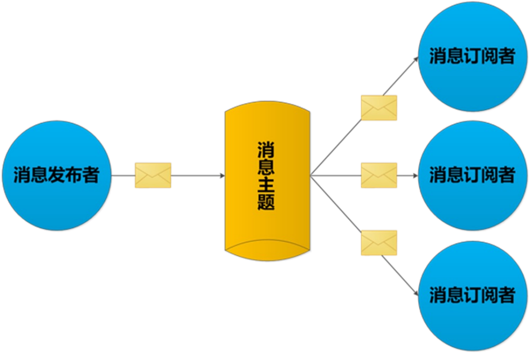
```
[Global]
MessageTitle=智能自制内容工具
Button[0]=确定(&O)
Button[1]=取消(&C)
Button[2]=关闭(&C)
[Project]
Title=新建项目
Button[0]=名称:
Button[1]=位置:
Button[2]=浏览(&B)...
Text[0]=当前目录已经存在，是否覆盖此目录?
[PutCode]
Title=请输入相应数字
Button[0]=编号:
Button[1]=如果存在相同编号直接覆盖
```

Tactics to Improve Modifiability

-Delaying Binding Time

Publish-subscribe style

- Already introduced in the software architecture style part (event-based system)
- Observer Pattern

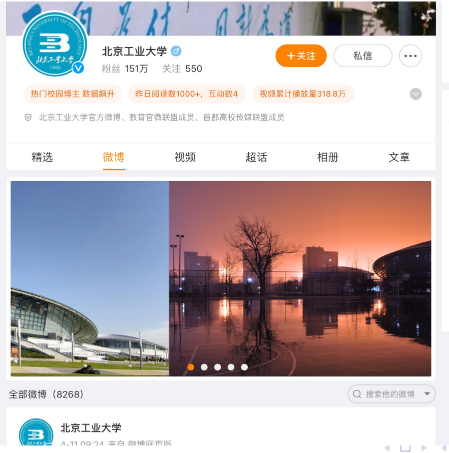


Tactics to Improve Modifiability

-Delaying Binding Time

Publish-subscribe style

- Example of Weibo



北京工业大学

粉丝 151万　关注 550

热门校园博主 数据飙升　昨日阅读数1000+，互动数4　视频累计播放量318.8万

北京工业大学官方微博、教育官微联盟成员、首都高校传媒联盟成员

全部微博（8268）

北京工业大学

4-11 09:24 来自 微博网页版

Tactics to Improve Modifiability

-Delaying Binding Time

Polymorphism

- Use different subclasses to implement different functionalities

```
class Animal {
  public void eat() {System.out.println(“进食");}
}
class Dog extends Animal {
  public void eat(){System.out.println("狗吃肉");}
}
class Cat extends Animal {
  public void eat(){System.out.println("猫吃鱼");}
}
```

```
public class Test {
  public static void main(String[] args){
    //定义为狗
    Animal a = new Dog();
    a.eat();
    //变成猫
    a = new Cat();
    a.eat();
```

Modifiability- Summary

Concerns of Modifiability

- Cost of Modification

Tactics to improve Modifiability

- Limiting the Scope of Modification
- Delaying Binding Time

Thank you!