

OOAD第十章

1. GRASP

可见性

是对象“看到”或引用其他对象的能力

实现对象A到对象B的可见性通常有四种方式：

属性可见性—B是A的属性

参数可见性—B是A中的方法的参数

局部可见性—B是A中方法的局部对象

全局可见性—B具有某种方式的全局可见性

为了使对象A能够向对象B发送消息，对于A而言，B必须是可见的

多态

当相关选择或行为随类型有所不同时，使用多态操作作为变化的行为类型分配职责

将不同的行为指定给不同的子类，用多态的方法来 判断具体应该使用那个类，而不是用

if instance of 来判断该类是什么、执行什么

面向对象三大特征（封装、继承、多态）之一

纯虚构

对人为制造的类分配一组高内聚的职责，该类并不代表问题领域的概念——虚构的事物，用以支持高内聚、低耦合和复用

非问题域的职责分配给人工 生成、在业务逻辑以外加的类。

间接性

将职责分配给中介对象，使其作为其他构件或服务之间的媒介，以避免它们之间的直接耦合。中介实现了其他构件之间的间接性

防止变异

识别预计变化或不稳定之处，分配职责用以在这些变化之外创建稳定接口

防止变异是一个根本原则，用来提供灵活性和防止变化——变化包括数据、行为、硬件、软件构件、操作系统等变化

2. 使用面向对象语言将设计制品映射为代码

用OO语言创建代码并不是OOAD的一部分，它是最终的目标

在设计模型中创建的制品为生成代码提供了必要的信息

用例、OOAD和OO编程结合使用可以提供从需求到代码的端到端路线图

迭代和进化方法关键：对编程中存在的变化和偏差有所预计和计划

面向对象语言中的实现需要为以下元素编写源代码：

类和接口的定义

方法的定义

类的实现要按照从耦合度最低到耦合度最高的顺序来完成