




第7章 软件设计——面向对象方法 前言


刘其成

计算机与控制工程学院

ytliuqc@163.com

2018-09

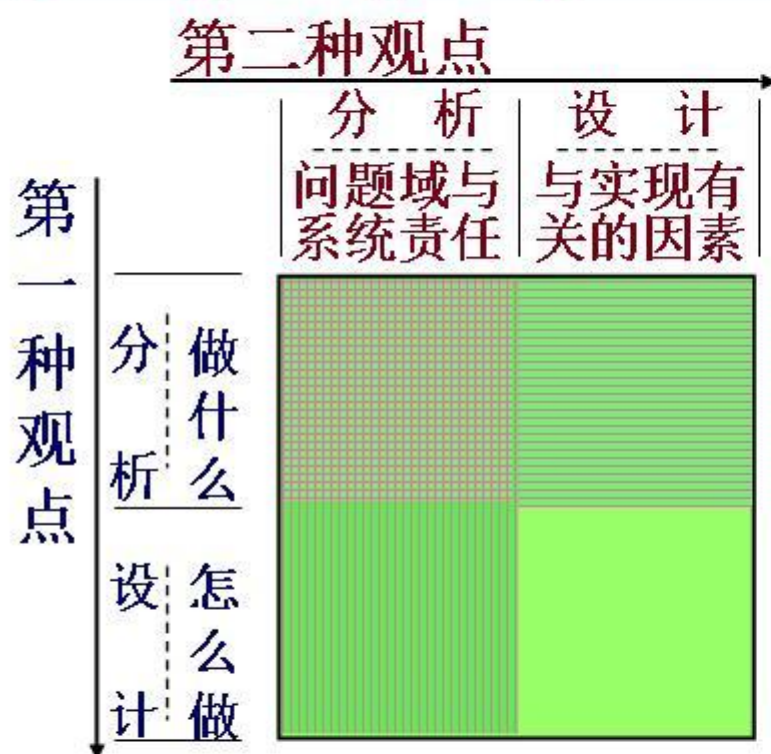
- 
- 面向对象分析
 - 获取用户需求，对问题域进行分析并建立问题域概念模型的过程。
 - 面向对象设计
 - 从计算机技术的角度，将分析阶段得到的概念模型转换成软件系统的技术实现方案，并在功能、性能及成本方面满足用户和质量要求的过程。
 - 一个将问题空间中的面向对象模型转换为求解空间中的技术实现模型的过程。
 - 从面向对象分析到面向对象设计，是一个逐渐扩充模型的过程。

- 
- 面向对象分析和面向对象设计是有明显区别的，但实际的软件开发过程中两者的界限却是模糊的。
 - 许多分析结果可以直接映射成设计结果，而在设计过程中又往往会加深和补充对系统需求的理解，从而进一步完善分析结果。
 - 因此，分析和设计是一个多次反复迭代的过程。

1、什么是面向对象设计

- 面向对象设计（**Object-Oriented Design, OOD**）就是运用面向对象方法进行系统设计。
- 面向对象设计有较长的发展史，在不同的时期有不同的内容和特点。
 - （1）早期的**OOD**
 - （2）基于**OOA**的**OOD**


两种不同的观点



关键问题：对象的特征细节（如属性的数据类型和服务流程图），是在分析时定义还是在设计时定义？

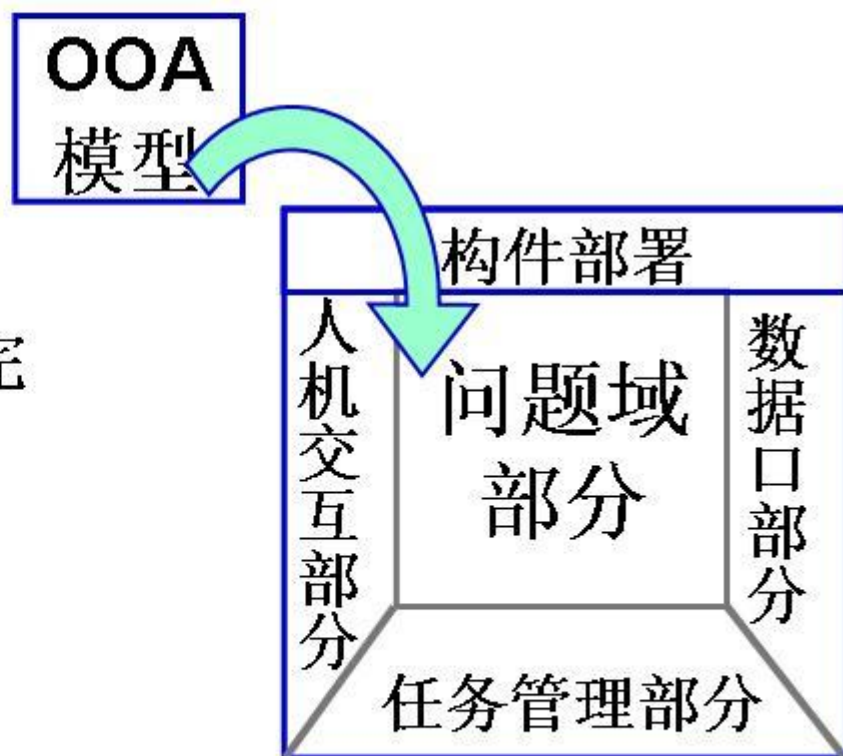
第二种观点的理由：

- (1) 在各种分析/设计方法中“做什么”和“怎么做”实际上没有严格的划分”。
- (2) 过分强调“分析不考虑怎么做”将使某些必须在OOA考虑的问题得不到完整的认识。
- (3) 由于OO方法表示形式的一致，不存在把细化工作留给设计人员的必然理由。
- (4) 避免重复地认识同一事物，并有利于分析结果的复用。

- 
- 现今的**OOD**具有如下特点：
 - (1) 以面向对象的分析为基础，一般不依赖于结构化分析。
 - (2) 与相应的**OOA**方法共同构成一种**OOA&D**方法体系。
 - (3) 较全面地体现了面向对象方法的概念和原则。
 - (4) 大多数**OOD**方法独立于编程语言，但具体应用**OOD**时，要考虑特定的编程语言。

2、OOA与OOD的关系

- (1)从OOA到OOD不是转换
 - 是调整和增补
 - 将OOA模型搬到OOD;
 - 进行必要的调整, 作为OOD模型的问题域部分;
 - 增补其它四个部分, 成为完整的OOD模型。



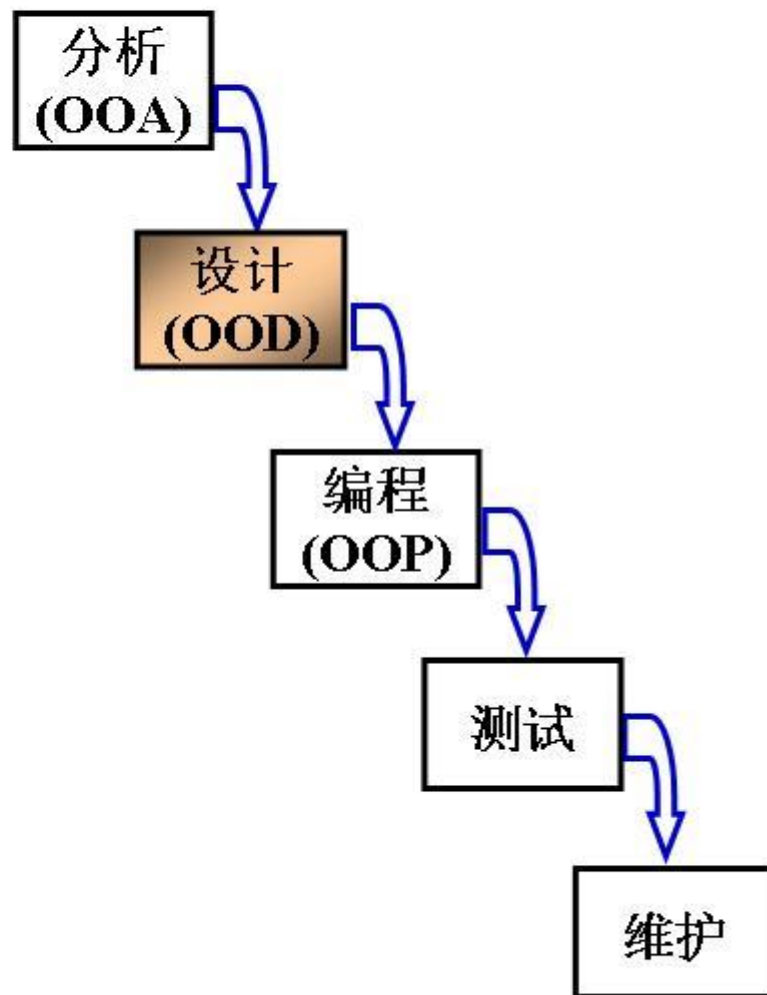
■ (2) 采用一致的概念和表示法 ——不存在分析与设计之间的鸿沟



- (3) 有不同的侧重点和不同的策略
 - **OOA**主要针对问题域，识别有关的对象以及它们之间的关系，产生一个映射问题域，**满足用户需求**，独立于实现的**OOA**模型。
 - **OOD**主要解决与实现有关的问题，基于**OOA**模型，针对具体的软、硬件条件（如机器、网络、**OS**、**GUI**、**DBMS**、编程语言等）产生一个**可实现的OOD**模型。
- (4) **OOA**与**OOD**可适合不同的生命周期模型
 - 瀑布模型、螺旋模型、增量模型、喷泉模型

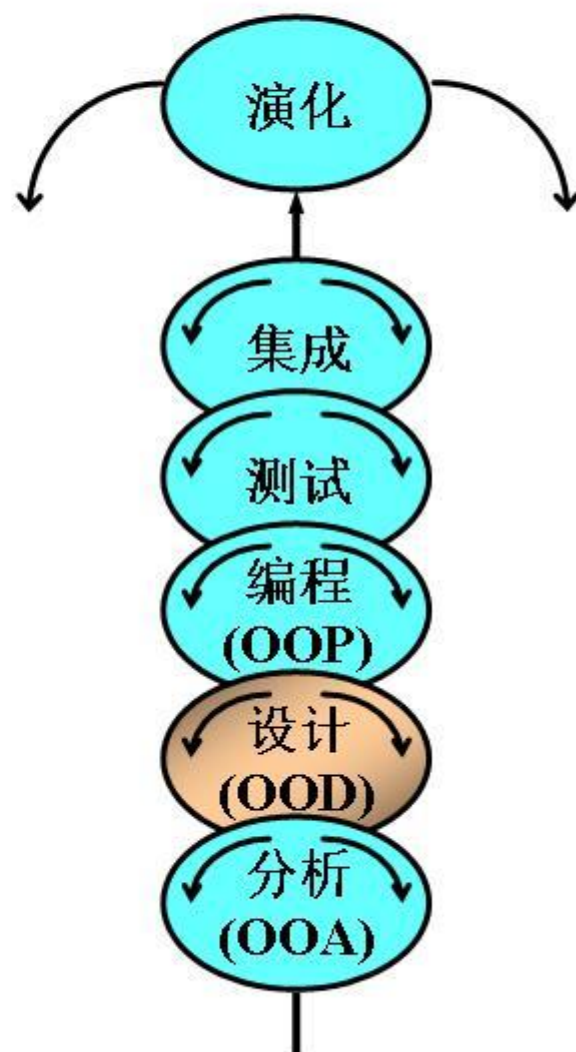
不同过程模型中的OOD——瀑布模型

- 瀑布模型核心思想是按工序将问题简化，将功能的实现与设计分开，便于分工协作。
- 将软件生命周期划分为分析、设计、编程、测试和维护等基本活动，并规定了它们自上而下、相互衔接的**固定次序**，如同瀑布流水，逐级下落。
 - 从上一项活动接收该项活动的工作对象作为输入，利用这一输入实施该项活动应完成的内容给出该项活动的工作成果，并作为输出传给下一项活动
 - 同时评审该项活动的实施，若确认，则继续下一项活动；否则返回前面，甚至更前面的活动。
- 对于经常变化的项目而言，瀑布模型毫无价值。



不同过程模型中的OOD——喷泉模型

- 喷泉模型是一种以用户需求为动力，以对象为驱动的模式。
- 该模型认为软件开发过程自下而上周期的各阶段是**相互迭代**和**无间隙**的特性。
 - 软件的某个部分常常被重复工作多次，相关对象在每次迭代中随之加入渐进的软件成分。
 - 无间隙指在各项活动之间无明显边界，如分析和设计活动之间没有明显的界限，由于对象概念的引入，表达分析、设计、实现等活动只用对象类和关系，从而可以较为容易地实现活动的迭代和无间隙，使其开发自然地包括复用



3、OOD模型和过程

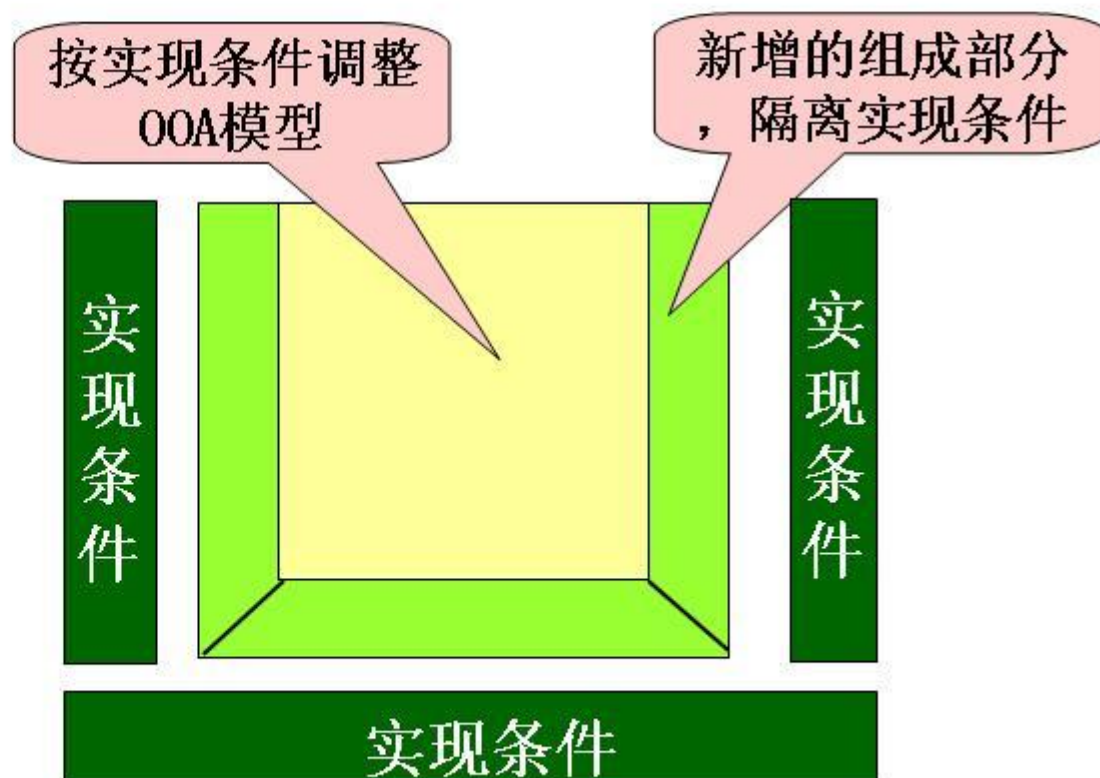
- **OOD——按实现条件对OOA模型进行调整，并补充几个新的组成部分（也是由对象构成）**

- 与实现有关的因素：

- 图形用户界面系统
- 硬件、操作系统及网络
- 数据管理系统
- 其他——编程语言、可复用构件库……

- 基本思想：

- 尽可能隔离实现条件对系统的影响——提供独立的接口。
- 对不可隔离的因素，按实现条件调整OOA模型

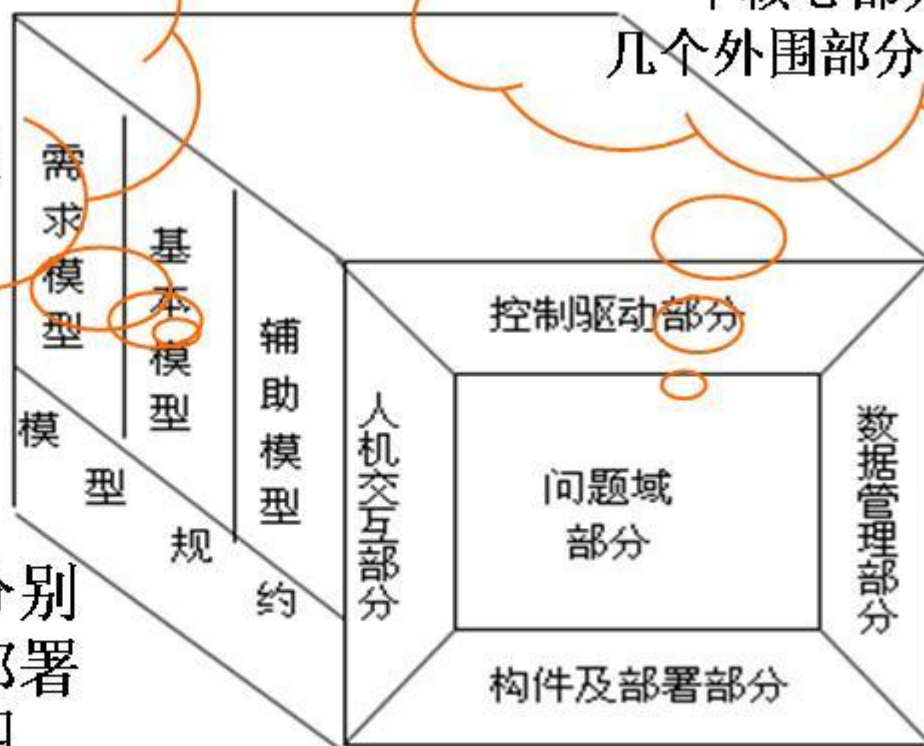


OOD模型——从两个侧面来描述

从另一侧面观察
OOD模型每个部分
如何用OO概念表达
——采用SA的概念及
模型组织方式

从一个侧面观察
OOD模型包括几个主
要部分
——一个核心部分加
几个外围部分

在辅助模型中要增加分别
用于描述构件模型和部署
模型的构件图和部署图



OOD过程

- 设计**OOD**模型的五个部分
 - 问题域部分的设计
 - 人机交互部分的设计
 - 控制流管理部分的设计
 - 数据管理部分的设计
 - 构件部署设计
- **前4项不强调次序**
 - 每个部分均采用与**OOA**一致的概念、表示法及活动，但具有自己独特的策略。
- 进行构件部署设计要在其前面四个部分完成后进行。



谢谢

2018年11月27日