

建模工具和语言

» 面向对象建模语言

面向对象建模语言在70年代中期到80年代后期之间出现；1989年-1994年间，面向对象建模方法从不到10种发展到50多种。

- Grady Booch's Booch 方法 ---Rational Software Corporation
- Ivar Jacobson's Object-Oriented Software Engineering (OOSE)
- ---Objectory
- James Rumbaugh's Object Modeling Technique (OMT)
- --- General Electric

建模工具和语言

» 面向对象建模语言

- 简而言之
 - The Booch 方法在项目设计和构建阶段极具表达力
- OOSE 对于业务需求分析提供了很好的支持
- OMT-2 对于数据密集型信息系统的分析极具表达力

UML语言

» UML语言适用于对软件密集型系统的物件进行

- 可视化
- 规范化
- 构建
- 文档化



UML语言

- 建模元素
- 关系
- 扩展机制
- 图

建模元素

» 结构化元素

- 类、接口、协作、用例、组件、节点

» 行为元素

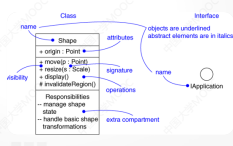
- 交互、状态机

» 分组元素

- 包、子系统

» 其他元素

- 注释



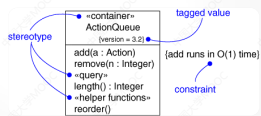
关系

- 依赖
- 关联
- 泛化
- 实现



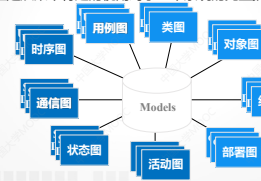
扩展机制

- 构造型
- 标记值
- 约束



模型、视图、图

- 模型是从某个特定的视角对于一个系统的完整描述



图

» 图是模型的一个视角

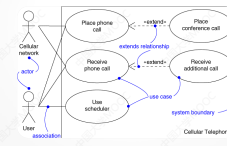
- 从某个特定的利益相关者的视角展示系统
- 提供系统的局部描述
- 与其他图语义一致

» 在UML中，有13种标准图

- 静态视图: 用例图、类图、对象图、组件图、部署图、包图、组合结构图
- 动态视图: 时序图、通信图、状态机图、活动图、定时图、交互概览图

用例图

» 描述系统功能需求



业务用例图

» 从用户视角描述系统功能需求

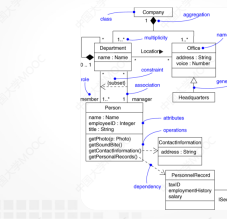
» 开发早期阶段使用

» 目的

- 确定系统上下文
- 捕捉系统需求
- 验证系统架构
- 驱动系统实现和测试用例生成

» 由领域专家和系统分析师完成

类图



类图

» 捕捉系统的词汇表。

» 目的

- 对系统中的概念进行命名和建模
- 确定协作关系
- 确定逻辑数据库结构

» 由分析师、设计师和开发人员完成