



# 北京交通大学

## 面向对象建模

——一图顶千言

苟娟琼

[jqgou@bjtu.edu.cn](mailto:jqgou@bjtu.edu.cn)

北京交通大学



# 目标

1. 了解UML建模的主要特点
2. 理解UML中主要模型图的含义
3. 初步掌握用例图、类图、序列图的绘制方法



## 建模：抽象与多视图

- A *model* is an **abstraction** (or **simplification**) of something. It is developed for a specific **purpose**.
  - 抽取系统中的基本特性而忽略非基本的特性，以便充分地注意与当前**目标**有关的方面。

### 需要哪些视图？ 如何描述？

- 面向对象认为**现实世界**由一系列彼此**相关**并能够**相互通信**的**实体组成**。



# 统一建模语言(Unify Modeling Language简称UML)

- 由Rumbaugh、Booch、Jacobson提出的统一建模语言(Unify Modeling Language简称UML)。
- UML是一种定义良好，易于表达，功能强大且普遍实用的建模语言。一图顶得千行字：
  - 更好地理解问题
  - 加强人员之间的沟通
  - 更早地发现错误或疏漏的地方
  - 可以获取设计结果
  - 为最后的代码生成提供依据



# Rational三剑客



Grady Booch



Ivar Jacobson

Jim Rumbaugh





# UML的主要目标

1. 用对象的概念来构造系统模型（不仅是针对软件系统）。
2. 建立直观的从模型直至可执行体之间的对应关系。（多层次）
3. 为复杂的系统建立衡量标准。
4. 创建一种对人和机器都适合的语言，即可以用人工来描述，又可以用机器实现计算机辅助建模。



# UML概念范围

- UML概念可以划分为以下范围：
  - 系统需求——对外呈现的功能
  - 静态结构——组成元素及其关系
  - 动态行为——对象自身的活动
  - 交互行为——内部交互行为，运行机理
  - 物理实现
  - 各种图之间的关系
  - 模型组织
  - 扩展机制



# 模型图

— 静态视图:

— 类图

— 对象图

— 部署图

— 组件图

○ 动态视图:

○ 用例图: 从外界对系统的感知

○ 状态图: 对象自身的活动

○ 活动图:

○ 序列图: }

○ 协作图: }

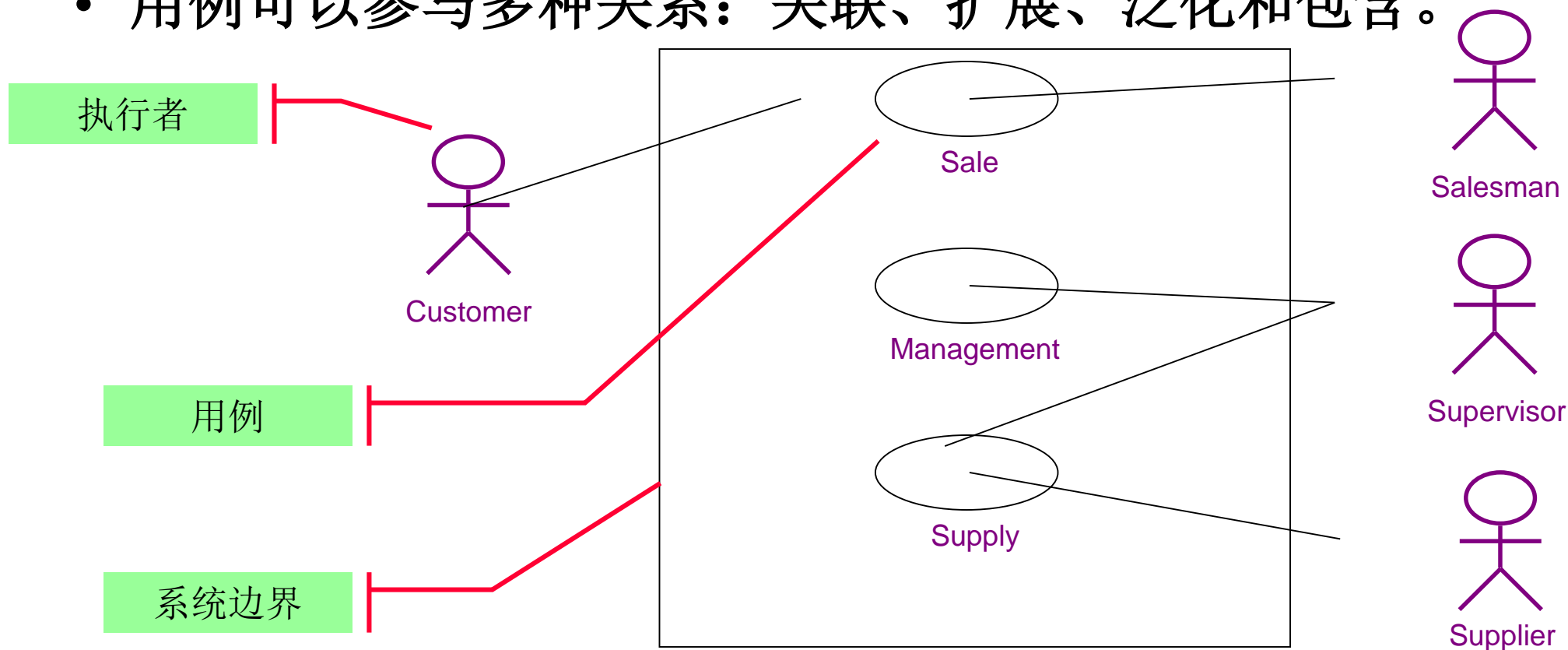
对象之间的关系





## 动态建模[1]—用例图

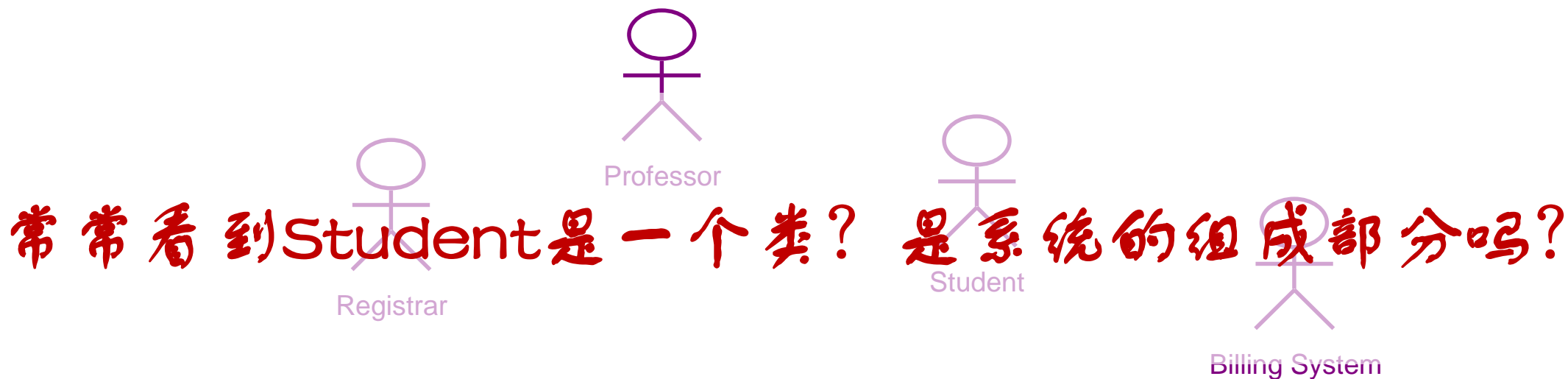
- 用例图描述执行者在各个用例中的参与情况。
- 软件开发过程是用例驱动的。
- 用例可以参与多种关系：关联、扩展、泛化和包含。





## 基本图素[1]—角色

1. 角色是与系统、子系统或类交互的**外部**人员，进程或事务。
2. 在运行时，具体人员会充当系统的多个执行者，不同用户可能会成为一个执行者。
3. 参与者实际上**并不是**系统的一部分。



根据上下文的不同而具有不同的含义



## 基本图素[2]—用例

- 用例是系统提供的外部可感知的功能单元，用例的目的是定义清晰的系统行为，但不解释系统的内部结构。
- 用例可以与角色关联，也可以参与其他的多种关系，比如扩展、泛化和包含等。
- 用户的动态部分用交互视图来描述，比如顺序图、协作图。
- 用例用椭圆来表示，用例名标在椭圆下方，用实线与同自身通信的用户相连。



求一元二次方程根



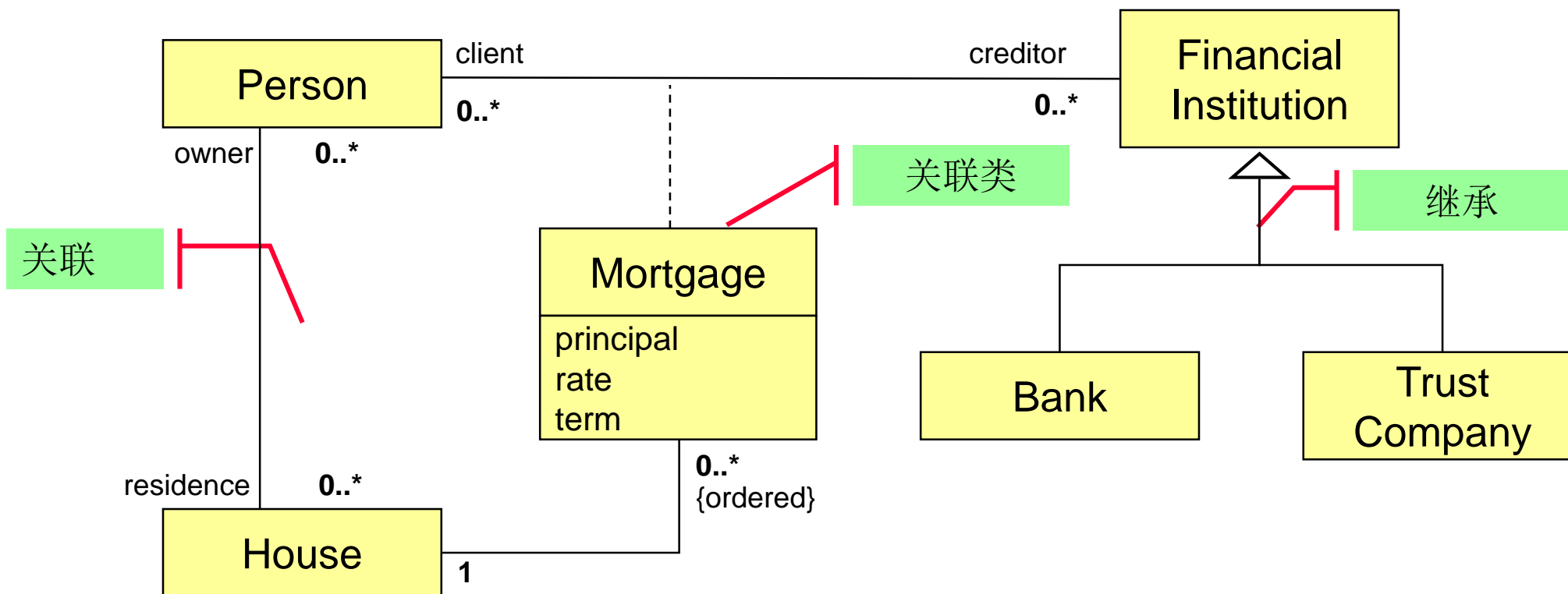
## 动态建模[1]—用例图

- **用例图**描述各个执行者在各个用例中的参与情况，描述系统为用户所感知的外部视图。
- 用例图的功能：
  - 捕获系统用户需求
  - 描述系统边界
  - 指明系统外部行为
  - 指导系统开发者的功能开发
  - 系统建模的起点，指导所有的类图和交互图的设计
  - 产生测试用例，用户文档
  - 估计项目大小和进度。



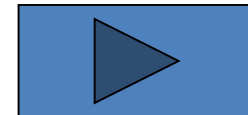
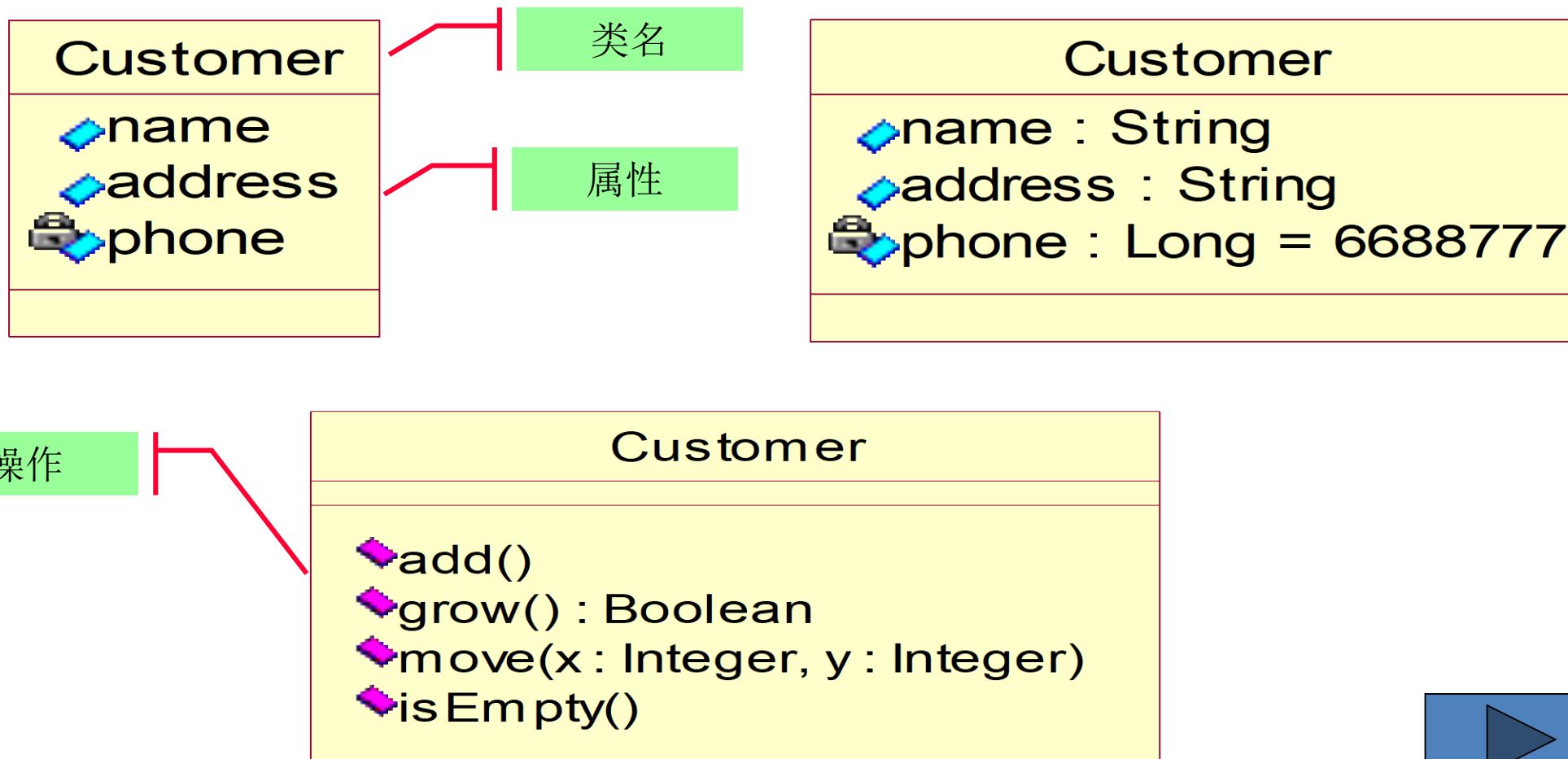
## 静态建模[1]—类图

- 静态视图是UML的基础，静态视图表示为类图，主要是描述类和类之间的关系。



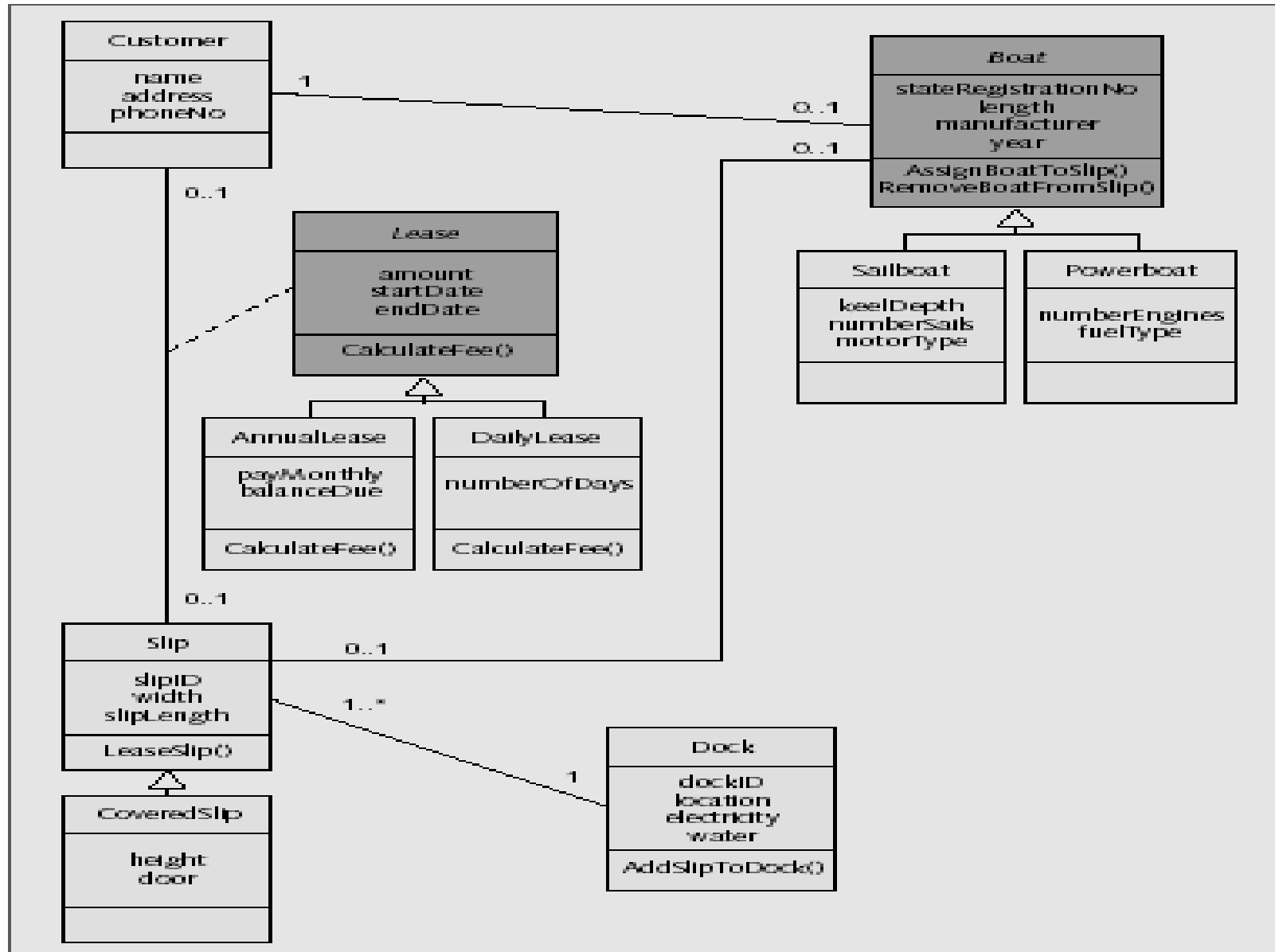


## 基本图素[4]—类





## 基本图素[4]—类之间的关系





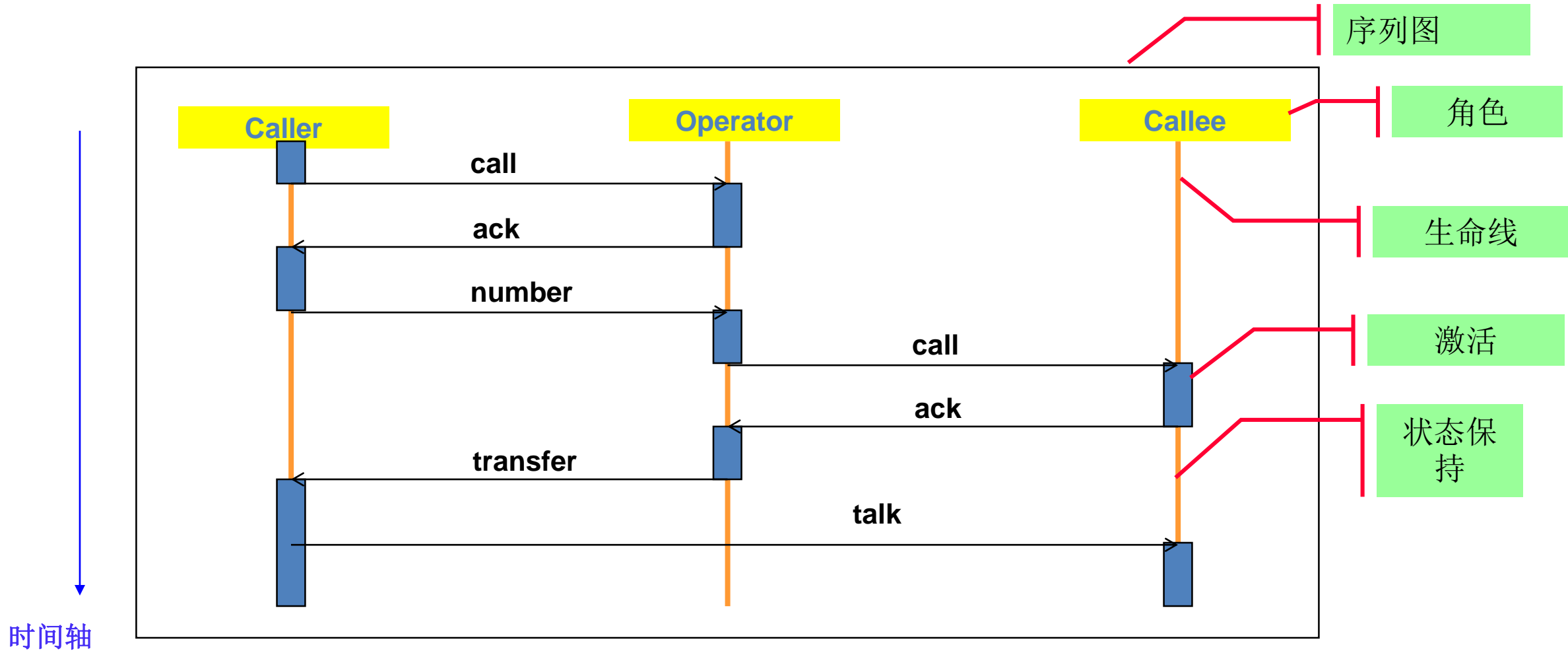
# 动态建模[4]—序列图

- 对象行为是通过交互来实现的，**交互**是对象间为完成某一目的而进行的一系列**消息**交换。
- **消息**是对象间的单向通信，从发送者到接受者的携带信息的控制流。消息可能带有**值参**。
- 消息序列可用两种图表示：**序列图**（重点在消息的时间顺序）和**协作图**（重点在交换消息的对象间的关系）。对协作图来说，时间顺序可以从顺序号获得。



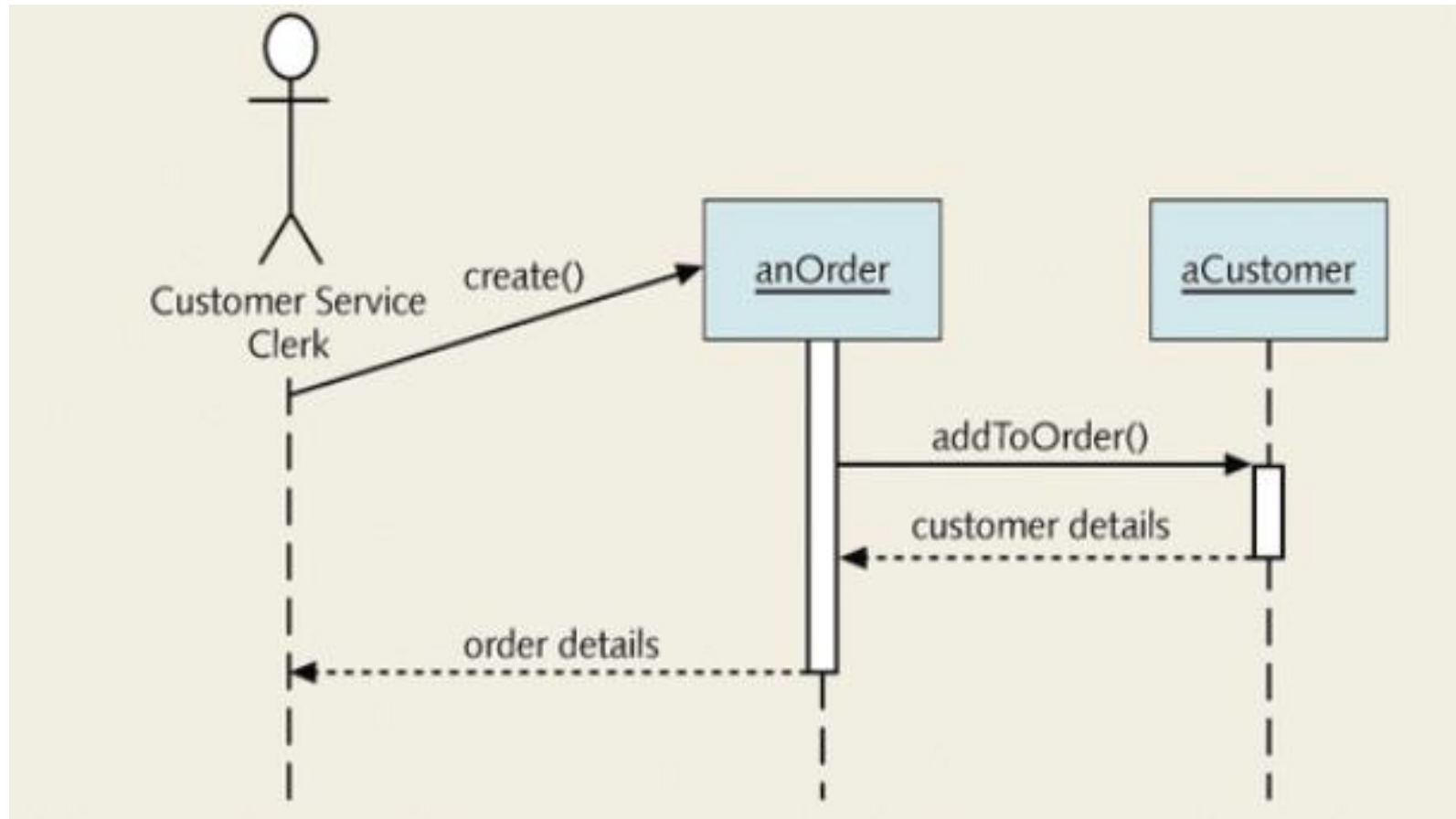


# 动态建模[4]—序列/顺序图





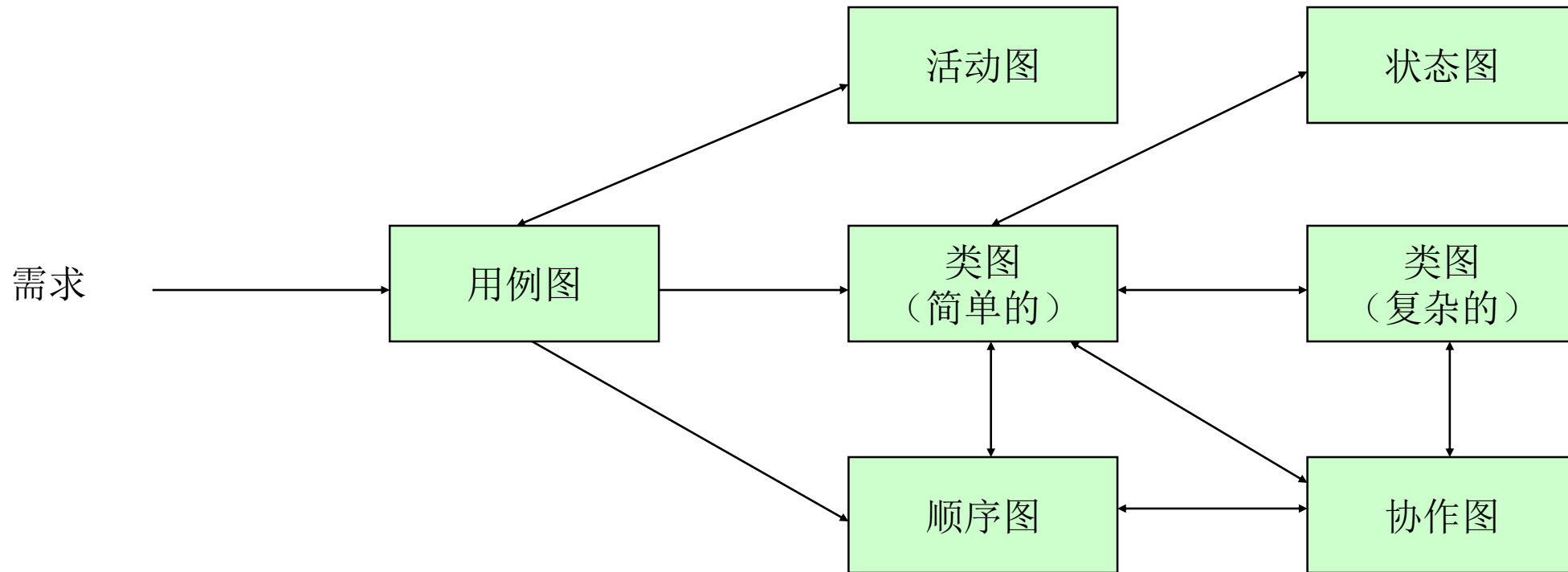
# 程序设计中的序列图



- 描述上图的交互过程



# UML中几个图之间的关系



UML中的模型图绘制的依据是什么?  
如何知道有哪些类?



# UML工具

CASE: Computer Aided Software Engineering

- IBM Rational **Rose** <http://www.rational.com/>
- Borland **Together**
- Jude 5.1 Community版免费 (Java)
- Trufun楚凡科技 Plato/**Kant2007** (集成eclipse)



- **Rational Rose**是美国**Rational**公司开发的一个**UML**工具，它全面支持**UML**模型中的元素，提供给开发者一个强大的可视化建模工具。
- 采用**Rose**进行面向对象系统开发的好处在于：
  - 通用的**UML**标准建模语言可以改善开发团队内部和团队与顾客之间的沟通；
  - 正向和逆向工程支持软件快速开发和对企业旧软件系统的集成。
  - 以用例为关注点的模型设计和系统开发模式可以提高软件的质量





## 练习

- 思考：
  - 基于面向对象思想，分析各模型图的设计意图。
  - **UML**中的这些图是在系统开发的哪个阶段绘制？
  - 绘制的依据和方法是什么？