



# 对象图

# 主要内容

- 概述
- 基本概念
- 建模方法



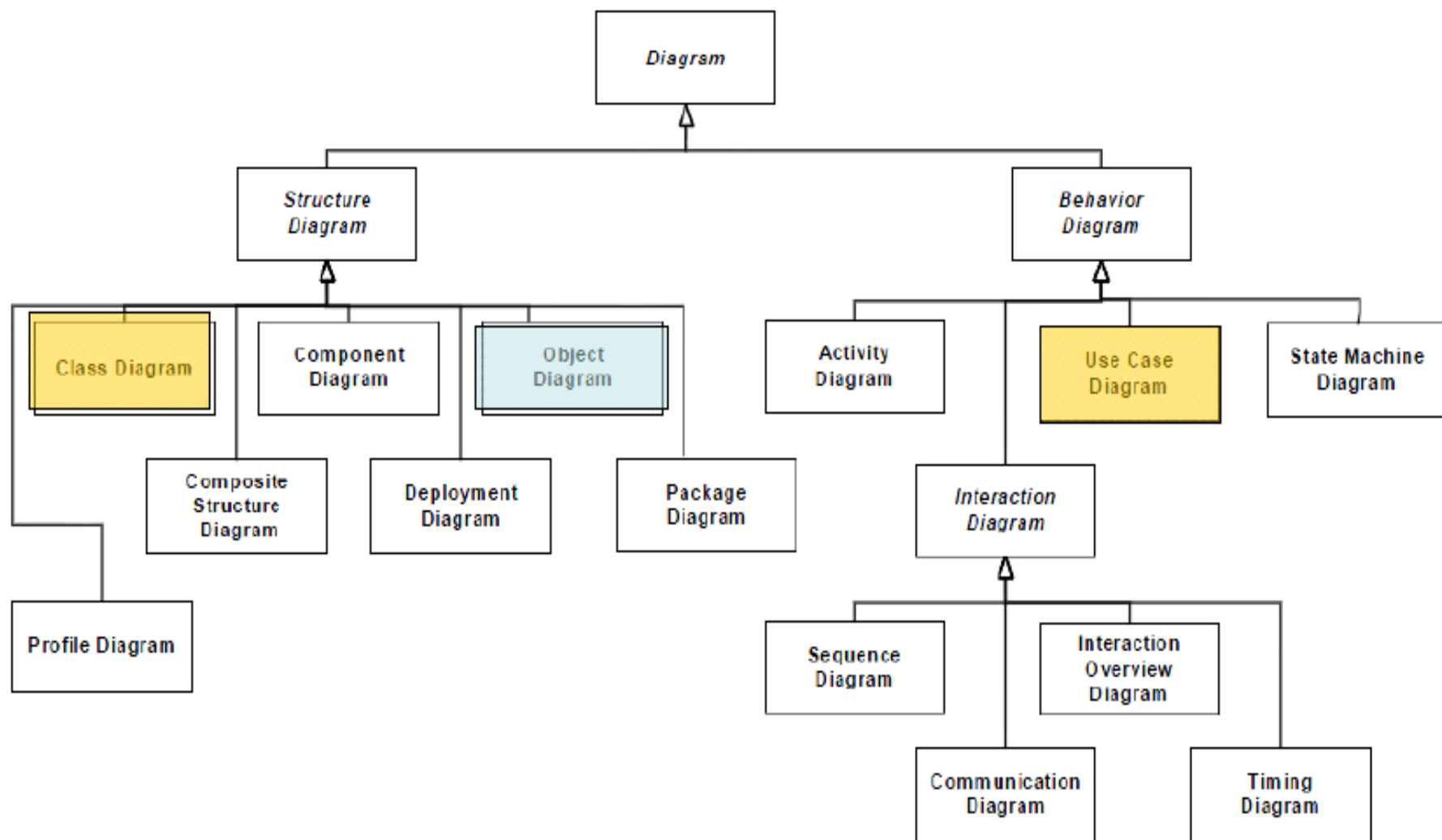
# 主要内容

## ➤ 概述

- 基本概念
- 建模方法



# 对象图



# 对象图

- A object diagram is a diagram that shows objects and their relationships at a point in time
- 对象图表示一组对象和它们之间的关系
- 对象图是一个系统的详细状态在某时刻的快照
- 对象图可以看作是类图在某个时刻的实例
- 对象图的建模元素
  - 对象
  - 对象之间的关系



# 主要内容

- 概述
- 基本概念
- 建模方法



# 对象的定义

在面向对象技术中，对象定义为：封装了状态和行为的、有明确边界和标识的实体。  
结构和行为类似的对象定义在它们共同的类中。因此，类的“实例”和“对象”是等价的术语



# 对象的特性

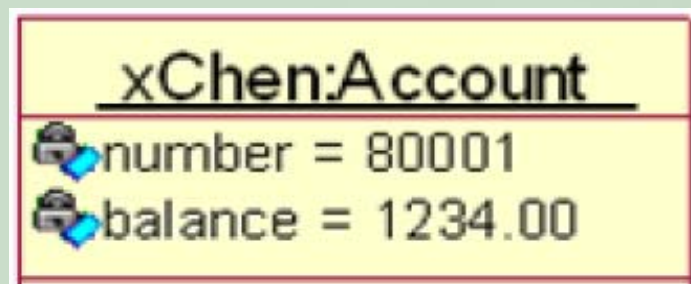
- 状态：用属性和关系表示
- 行为：用操作、方法和状态机表示
- 标识：在生命周期内它有唯一的名





# 对象的表示方法

- 对象名
  - “对象名:类名”
  - “:类名”
  - “对象名”
- 属性
  - “属性名=属性值”



# 对象和类的关系

- 对象是一个存在于时间和空间中的实体，类仅代表一种抽象，抽象出对象的本质
- 类是静态的，对象是动态的
- 类是一般化，对象是个性化
- 类是定义，对象是实例
- 类是抽象，对象是具体

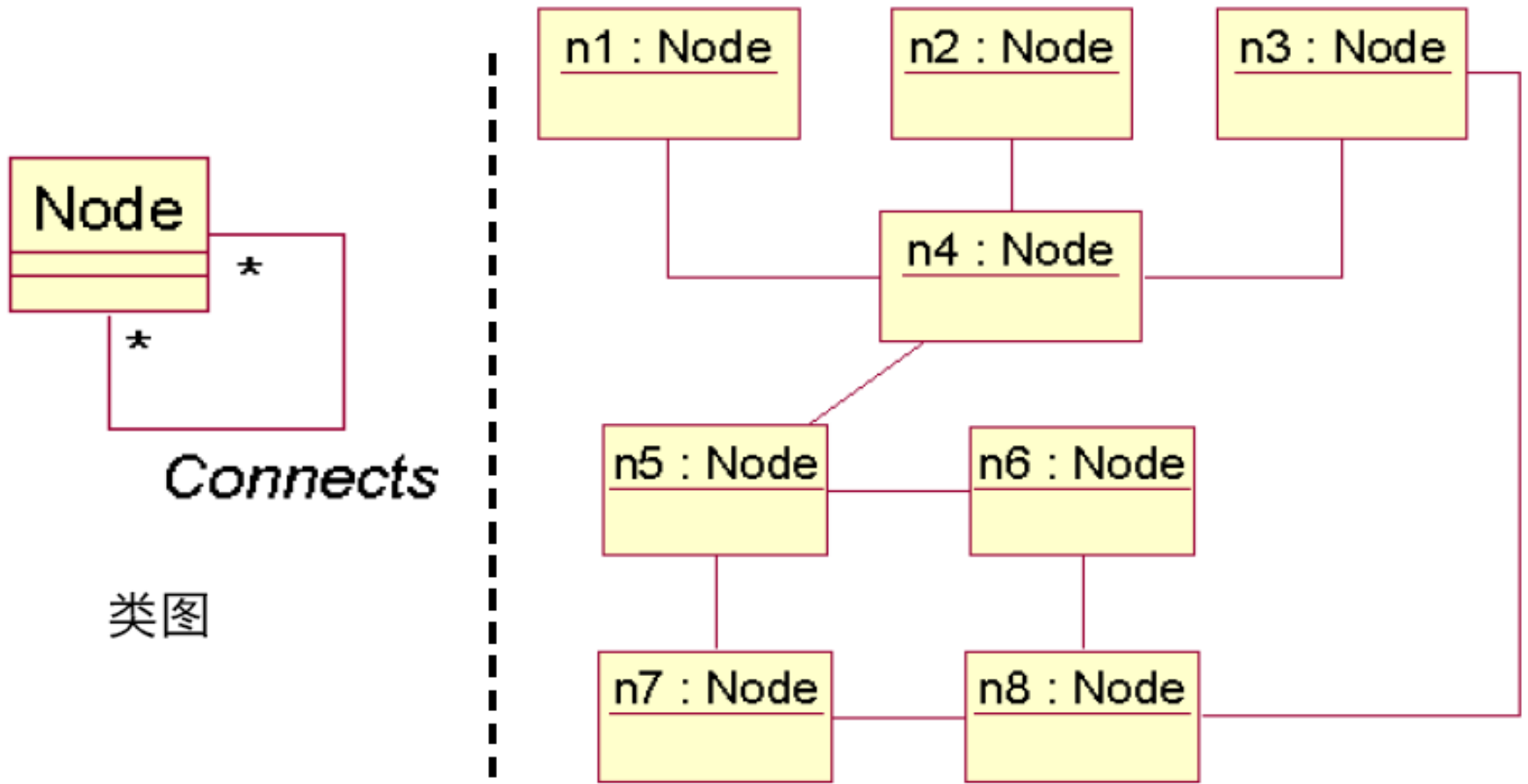


# 对象之间的关系

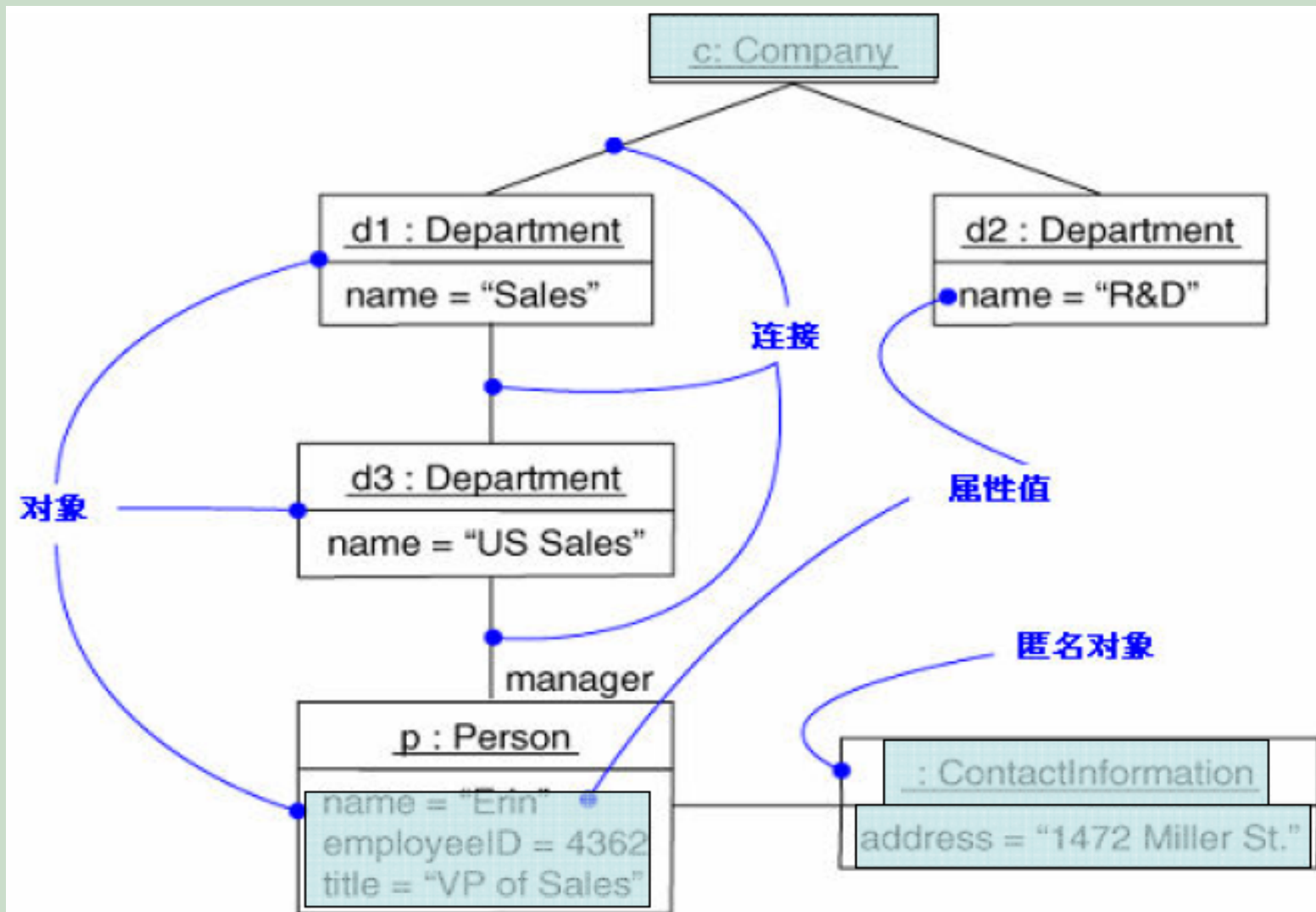
- 对象之间通过协作，体现系统的行为
- 链接（link）是关联的一个实例



# 对象图实例



# 对象图实例



# 对象图的用途

- 论证类模型的设计
  - 当设计了类模型时，可以通过对象图来模拟出一个运行时的状态，以验证运行时设计的合理性
  - 同时，也可以作为开发人员讨论的一个基础
- 分析和说明源代码
  - 在分析源代码时，可以通过对象图来细化分析
  - 而对于开发人员，对于逻辑较复杂的类交互时，可以考虑画出一些对象图来做补充说明



# 对象图的说明

- 对象与类的图形表示相似
- 对象图实质上是类图的实例
- 对象图常用于表示复杂的类图的一个实例
- 对象图的使用较为有限，主要用于表达数据结构示例，以及表示系统在某个特定时刻的具体情况





# 类图和对象图的比较

- 类图可以完整地描述系统的对象结构；对象图则不能。
- 类图中的一个类可能对应对象图中的多个对象，相互之间存在关系的类，对应的对象配置可能很多。
- 类有三个分栏；对象只有两个。
- 类的属性分栏定义属性的特征；对象定义属性当前值。
- 类中列出操作；对象中不包含操作。
- 类间的关联可指定多重性；对象间的链接没有多重性，都是一对一的。





# 主要内容

- 概述
- 基本概念
- 建模方法



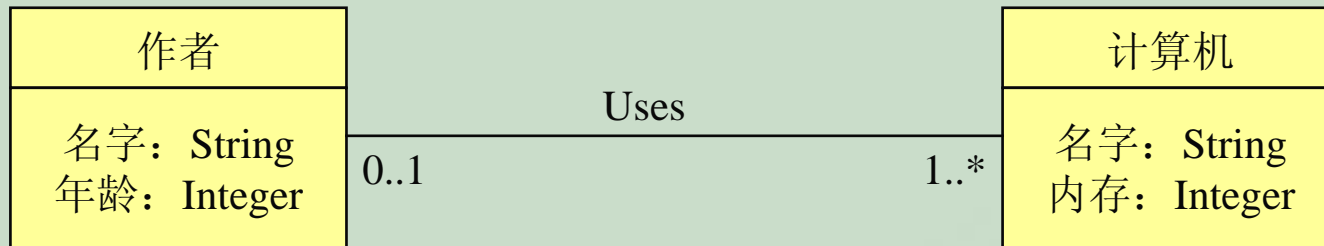
# 对象图建模

- 识别相关的类、接口和其它元素及其之间的关系
- 针对某个场景，在特定时刻冻结，识别相关的对象
- 描述对象的状态和属性值
- 描述对象之间的链接



# 对象图示例

类图



对象图

