

# 软件工程

## 第12讲 面向对象分析与设计（4）

贾西平

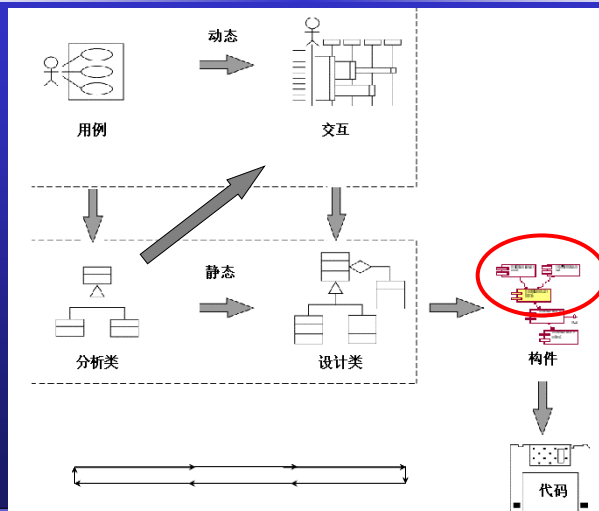
Email: [jiaxp@126.com](mailto:jiaxp@126.com)

# 架构

Think

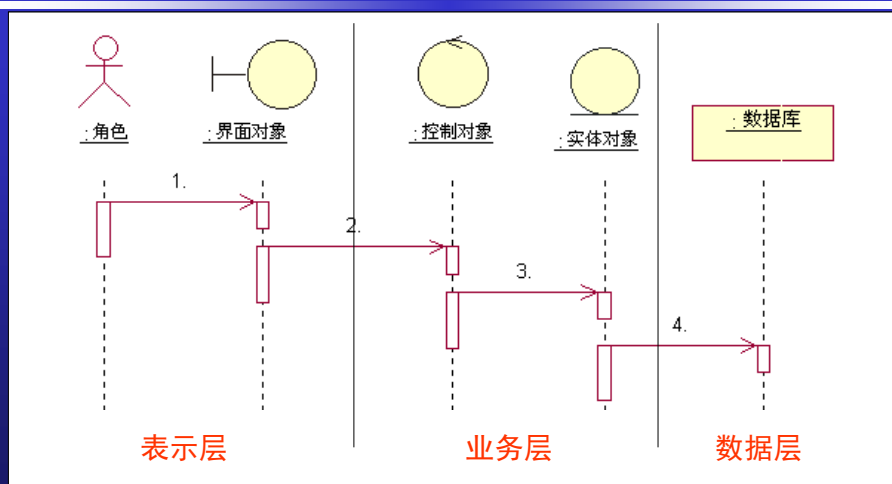


## 开发流程



3

## 三层



4

## 三层架构

表示层

Select Department		
Dept ID	Department Name	Department Manager
100	R & D	David Scott
200	Sales	Meiwa Kelly
300	Finance	Mary Anne Shea
400	Marketing	Scott Evans
500	Shipping	Jose Martinez

随着界面表达方式的变化而变化

业务逻辑层



随着业务的变化而变化

数据存取层



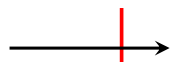
随着数据存取的方式变化而变化



5

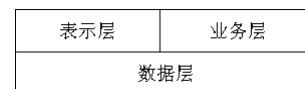
## 架构实现上的变迁

❖ 单机



三层同在一台机器上

❖ 客户/服务器



❖ 三层分开



6

## 不同系统的运行架构



电梯



收款机



Outlook Express



局域网内运行的酒店管理系统



特大型商业网站



7

## 进一步分层

表示层	Window窗体，HTML页面，报表
业务环境层	信息进入系统之前过滤和处理信息
业务规则层	实施业务规则
数据转换层	将业务层的要求翻译成合适的语言（如SQL）
数据访问层	执行数据访问API（如：ADO）
存储	提供物理存储服务（如：SQL Server，Oracle）



8

## 表示层



9

## 业务环境层

☒ 男
 ☐ 女

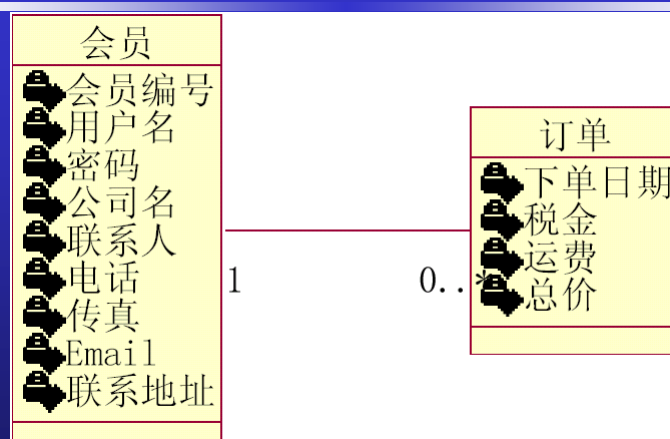
引体向上

仰卧起坐

检测代码、JavaScript...

10

## 业务规则层



实施业务规则——就是前面我们说的实体类和控制类



11

## 存储：对象持久化

- ❖ 文件
  - ❖ 各种格式的文件(.txt, .ini...)
- ❖ 关系数据库 (RDBMS) (最常用)
- ❖ 面向对象数据库 (OODBMS)
  - ❖ Jasmine (多媒体, 大规模集成电路)

关系数据库正在向对象-关系数据库发展  
Oracle的可变数组、嵌套表



12

## 用关系数据库来存储

- ❖ 你想把车停在一个**面向对象的**车库里。把车开进车库，下车，关上车门，然后回到你的房间。当你想出去的时候，只要走进车库，钻进汽车，启动，然后开走。

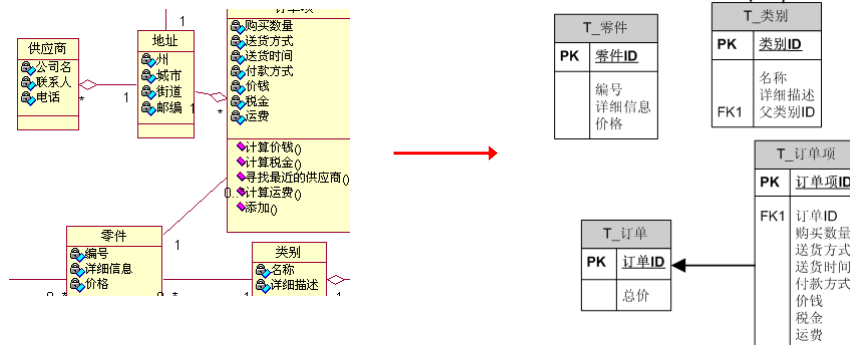


- ❖ 你想把车停在一个**关系数据库**的车库里。把车开进车库，下车，卸下车门，将它们放在地上；卸下所有的车轮，将它们放到地上；卸下保险杠及其它的东西。然后回到你的房间。当你想出去的时候，走进车库，先安上车门，再安上保险杠，然后是车轮等等，都安完了，钻进汽车，点火，然后开走。



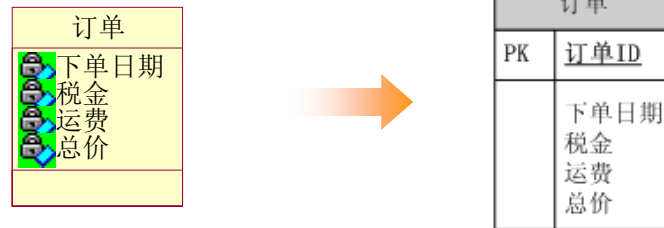
13

## 把实体类映射到关系数据库



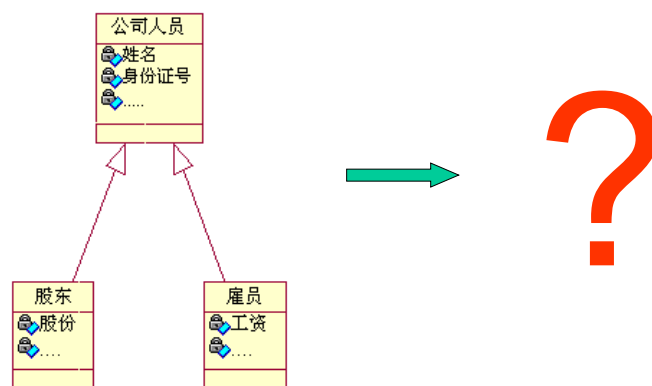
14

## 映射类和属性



15

## 映射泛化关系



16



## 映射泛化关系（1）

T_公司人员	
PK	<u>公司人员ID</u>
	姓名 身份证号 股份 工资

### ❖ 优点：

- ❖ 只有一张表
- ❖ 能实现角色变化
- ❖ 报表操作简单

### ❖ 缺点：

- ❖ 子类的修改会影响到整个结构
- ❖ 数据库存在大量空值，浪费空间

只为超类建一张表



17

## 映射泛化关系（2）

T_股东		T_雇员	
PK	<u>股东ID</u>	PK	<u>雇员ID</u>
	姓名 身份证号 股份		姓名 身份证号 工资

每个子类映射一张表

### ❖ 优点：

- ❖ 表中包含了具体子类的所有信息

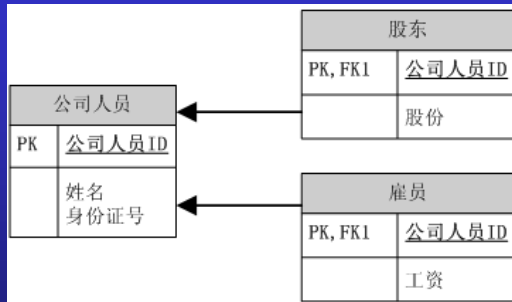
### ❖ 缺点：

- ❖ 超类的修改会影响到所有子类表
- ❖ 角色变化时，会造成ID的重新赋值
- ❖ 支持多重角色时，数据完整性难以维护



18

## 映射泛化关系（3）



❖ 优点：

❖ 弹性最好

❖ 缺点：

❖ 表的数量多

❖ 访问数据的时间稍长

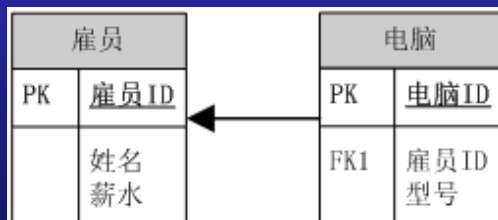
超类子类都映射成表，超类主键作为所有类的主键



19

## 映射关联关系（1）

——1对0..1



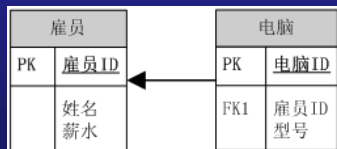
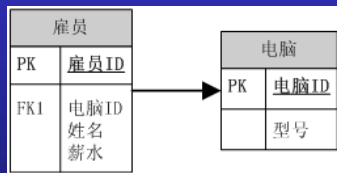
外键放在0..1一端



20

## 映射关联关系（2）

——1对1



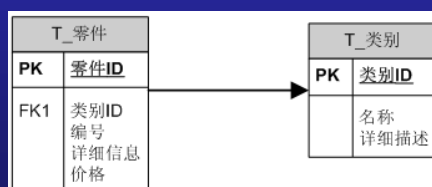
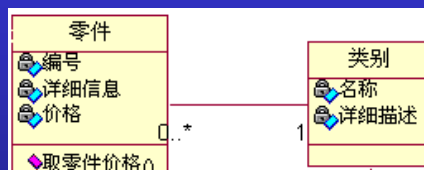
外键放在任意一端



21

## 映射关联关系（3）

——1对多



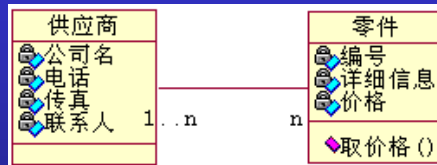
外键放在“多”方



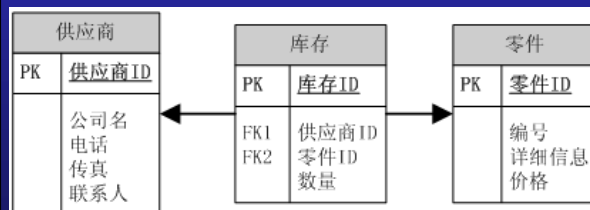
22

## 映射关联关系（4）

——多对多



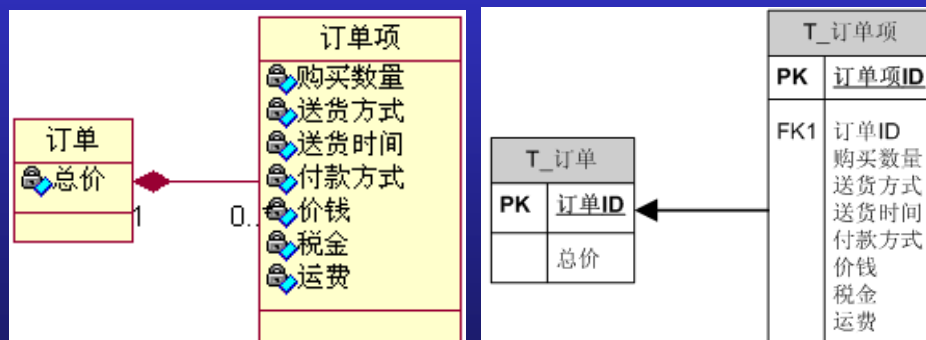
添加第三个表



23

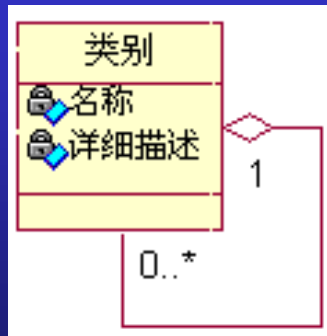
## 映射聚合/组合关系

● 映射规则同二元关联



24

## 映射反身关联（聚合）



25

## 主键的选择

- 在能单一标识记录的字段中挑选有意义的字段作为主键（学号，工号）
- 另外增加无意义字段作为主键（代理主键）



26

## 讨论：主键的选择

❖ 一个企业组织，“职员”应该用什么作为主键？

- 姓名
- 工号（03012045）
- 身份证号（340205740801203）
- 系统添加的ID



27

## 主键的作用

- ❖ 唯一标识记录
- ❖ 被其他表引用为外键



28

## 主键不应有业务含义

- ❖ 有业务含义，意味着可能潜伏着变化
- ❖ 任何对主键的修改都可能导致巨大的工作量



29

## 代理主键的好处

- ❖ 每个表的主键都是相同的数据类型
- ❖ 表间连接被限定在单个列上，SQL语句的书写不复杂
- ❖ 更稳定的设计

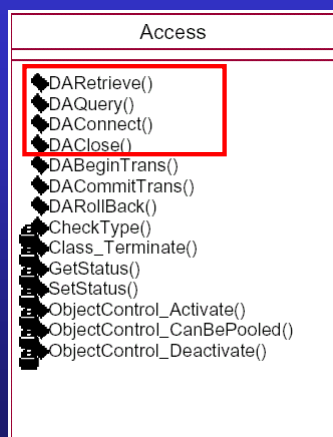


30





## 数据访问层



整个系统只需一个  
执行数据访问API（如：ADO）



33

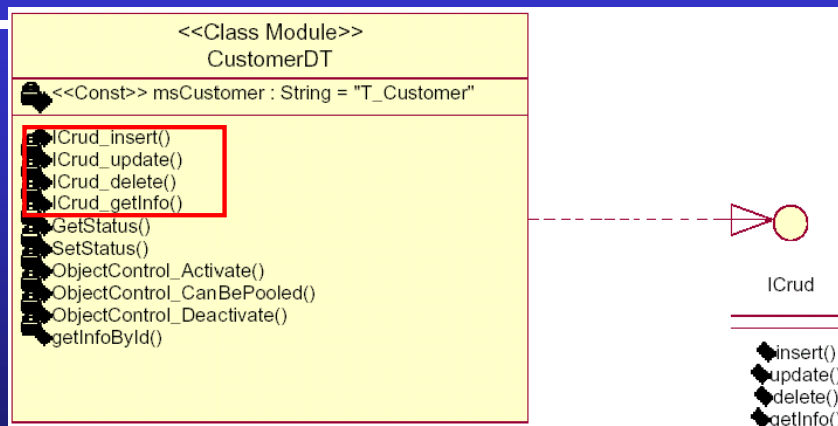
## 数据访问层

- ❖ 代码解析（ VB ）
- ❖ DAConnect： 连接数据库
- ❖ DARetrieve： 取数据
- ❖ DAQuery： 执行SQL语句



34

## 数据转换层



每个实体类对应一个  
将业务层的要求翻译成合适的语言（如SQL）



35

## 推荐视频

- <https://www.bilibili.com/video/BV1Q741157ve?p=64>
- <https://www.bilibili.com/video/BV1Q741157ve?p=65>
- <https://www.bilibili.com/video/BV1Q741157ve?p=66>
- <https://www.bilibili.com/video/BV1Q741157ve?p=67>
- <https://www.bilibili.com/video/BV1Q741157ve?p=68>
- <https://www.bilibili.com/video/BV1Q741157ve?p=69>
- Web系统架构设计  
<https://www.bilibili.com/video/BV1Q741157ve?p=70>
- 数据库选择策略  
<https://www.bilibili.com/video/BV1Q741157ve?p=71>



36

谢谢大家！  
感谢清华大学刘强老师的视频资源！

