

# 厦门大学计算机科学系本科生课程

## 《数据库系统原理》



## 第1章 绪论 (2018版)

林子雨

厦门大学计算机科学系

E-mail: [ziyulin@xmu.edu.cn](mailto:ziyulin@xmu.edu.cn) ▶▶

扫一扫访问班级网站  
支持手机浏览

主页: <http://www.cs.xmu.edu.cn/linziyu>





# 第1章 绪论

- 1.0 课程介绍
- 1.1 数据库系统概述
- 1.2 数据模型
- 1.3 数据库系统结构
- 1.4 数据库系统的组成



# 1.0 课程介绍

- 1.0.0 主讲教师和助教
- 1.0.1 学科知识点与本课程的关系
- 1.0.2 本课程的主要学习内容
- 1.0.3 班级网站



# 1.0.0 主讲教师



主讲教师：林子雨

中国高校首个“数字教师”提出者和建设者

2009年7月从事教师职业以来

累计**免费**网络发布超过**500万**字高价值教学和科研资料

网络浏览量超过**500万**次



数字教师LOGO



打造了国内高校大数据教学知名品牌

大数据教材



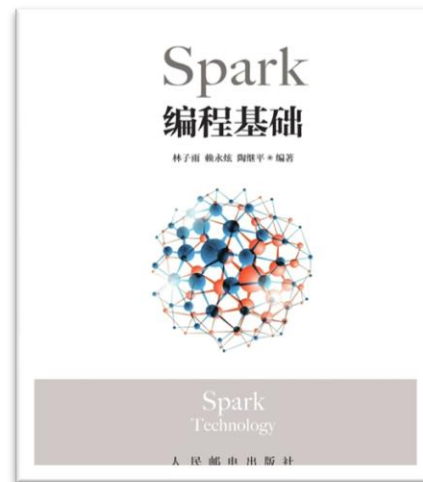
2015年7月第1版  
2017年2月第2版

配套实验指导书



2017年8月出版

  
1+1黄金组合  
厦门大学林子雨编著

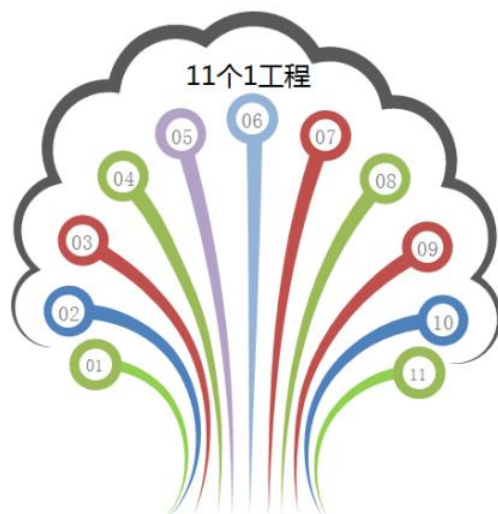


2018年3月出版



# 高校大数据课程

公共 服务 平台



中国高校大数据课程公共服务平台  
厦门大学数据库实验室全力打造

- 01 1本教材
- 02 1个教师服务站
- 03 1个学生服务站
- 04 1个公益项目
- 05 1堂巡讲公开课
- 06 1个示范班级
- 07 1门在线课程
- 08 1个交流群
- 09 1个保障团队
- 10 1个培训基地
- 11 1个实验平台

- 国内高校第一个大数据课程公共服务平台
- 为教师教学和学生学习大数据课程，提供全方位、一站式服务
- 平台所有资源全部通过网络**免费共享（无需注册）**、采用WORD和PPT可编辑格式



制作精美的平台宣传图册





宣传片观看地址: <http://dblab.xmu.edu.cn/post/xuanchuanpian/>

中国高校大数据课程公共服务平台

# 3分钟FLASH 动画宣传片



扫一扫手机观看





# 1.0.0 课程助教



## 助教：程璐

单位：厦门大学计算机科学系数据库实验室2017级硕士研究生  
E-mail: luchengxm@outlook.com



## 助教：林哲

单位：厦门大学计算机科学系数据库实验室2017级硕士研究生  
E-mail: linzhe@stu.xmu.edu.cn



# 1.0.1 学科知识点与本课程的关系

## 1. 课程分类（三大类）

### (1) 计算机数学类

数学分析      线性代数  
概率与统计    数值分析  
离散数学      数理逻辑

### (2) 硬件与体系结构类

普通物理    电路原理    电子技术  
数字逻辑                      汇编语言  
计算机组成原理                      微型计算机技术  
计算机网络与通信    计算机体系结构

### (3) 计算机软件类

计算机导论  
高级语言程序设计      面向对象程序设计      程序设计方法学导论  
数据结构                      算法设计与分析  
操作系统                      编译原理                      **数据库系统原理**  
    UNIX系统程序设计      软件体系结构与开发环境  
计算机图形学                      计算机图象处理  
视窗程序设计                      人机界面设计技术  
人工智能                      形式语言与自动机  
软件工程      管理信息系统      电子商务      信息安全      嵌入式系统



# 1.0.1 学科知识点与本课程的关系

## 2. 与《数据库系统原理》相关的课程

### (1) 前导课程

数学

高级语言程序设计

程序设计方法学

离散数学

数据结构

算法设计与分析

数理逻辑

### (2) 相关课程

软件工程

面向对象程序设计

计算机网络 .....

### (3) 互为补充（提供基础与工具）

软件体系结构与开发环境 .....

### (4) 后续课程

管理信息系统

电子商务

决策支持系统

数据仓库

数据挖掘

人工智能

大数据技术原理与应用

毕业设计



## 1.0.2 本课程的主要学习内容

- 1.0.2.1 本课程的研究范畴和教学内容
- 1.0.2.2 本课程的教学思想与目标
- 1.0.2.3 参考书和学习方法
- 1.0.2.4 考核方法



# 1.0.2.1 本课程的研究范畴和教学内容

## (1) 本课程的研究范畴

- **数据库理论**

数据库理论的研究主要集中于关系演算理论、关系数据库理论等。近年来，随着云计算与大数据的兴起，非关系型数据库（NoSQL）得到了迅速发展，成为关系数据库重要的补充。

- **数据库管理系统软件的研制**

数据库管理系统DBMS是位于用户与操作系统之间的一层数据管理软件。它实现对共享数据的有效组织、管理和存取。DBMS的研制包括研制DBMS本身以及以DBMS为核心的一组相互联系的软件系统。研制的目标是扩大功能、提高性能和提高用户的生产率。

- **数据库设计**

数据库设计的主要任务是在DBMS的支持下，按照应用的要求，为某一部门或组织设计一个结构合理、使用方便、效率较高的数据库及其应用系统。



# 1.0.2.1 本课程的研究范畴和教学内容

## (2) 本课程的教学内容

- 本课程较全面介绍数据库系统的基本概念、基本原理和基本技术
- 着重介绍关系模型系统，包括关系模型的基本概念、关系代数、关系演算、关系数据库标准语言--SQL的简要介绍、查询优化等
- 详细讨论关系数据库设计理论，包括函数依赖、多值依赖、Armstrong公理、关系模式的分解、关系模式的规范化等
- 简要介绍数据库设计方法
- 简要介绍如何使用编程方法对数据库进行操作的技术



## 1.0.2.2 本课程的教学思想与目标

本课程将注重对同学如下能力的培养

1) 分析、归纳和设计能力：深刻理解基本概念和方法，学会数据库系统的分析方法，归纳各种概念，权衡各方因素，设计合理系统。

2) 开发、调试能力：能够动手开发成功的数据库应用系统。

3) 自学能力：数据库软件很多，发展快，绝大多数要求学生实验或课余自学掌握，课堂重在讲方法和个案剖析。

4) 系统集成能力：数据库与其他部分，数据库与数据库之间，异构环境和系统之间，如达到全局优化，要能把握较复杂较系统的整体结构。

5) 研究创新能力：数据库技术发展快，新内容层出不穷，注意学习类比，观察提炼，提出新问题，新概念，新方法。



## 1.0.2.3 参考书和学习方法

- **学习方法.....**
- 《数据库系统概论》（第四版），王珊 萨师焯 著
- 《数据库系统教程》，施伯乐等著
- 《数据库系统基础》（第二版），冯玉才著
- 《DATABASE PROCESSING (数据库处理-基础、设计与实现)》（第七版），[美] David M.Kroenke 著
- 《Database Design》，[美] Ryan K.Stephens Ronald R.Plew 著
- 《WEB数据库开发与应用》（可以任选）
- 相关的介绍大型数据库系统的书籍，如：
  - 《SQL Server 数据库开发指南》
  - 《Oracle 数据库管理员指南》
  - 《BD2应用技术》
- 最近还有一些新的数据库系统原理的教材，外文书见下页
- 到后期还会推荐一些相关的数据库系统理论和实践的书籍





## 1.0.2.3 参考书和学习方法

### 外文参考书

- 《THE CONCEPTS OF DATABASE MANAGEMENT, 2E》, (数据库管理系统基础第2版) Philip J. Pratt, Joseph J. Adamski
- 《DATABASE SYSTEM CONCEPTS, 4E》(数据库系统原理第4版), Abraham Silberschatz, Henry F. Korth, S. Sudarshan
- 《DATABASE PROCESSING (数据库处理——基础、设计与实现)》(第七版), [美] David M. Kroenke
- 《DATABASE SYSTEM IMPLEMENTATION》(数据库系统实现), Hector Garcia-Molina, Jeffrey D. Ullman, Jennifer Widom (Stanford University)
- 《DATABASE: PRINCIPLES, PROGRAMMING, AND PERFORMANCE, 2E》(数据库: 原理、编程和性能第2版), Patrick O'Neil, Elizabeth O'Neil
- 《Database Design》, [美] Ryan K. Stephens Ronald R. Plew 著



# 1.0.2.4 教学进度表

章（或节）	主要内容	学时安排
第一章 绪论	数据库的基本概念	6
第二章 关系数据库	关系数据库的模型和操作语言	6
第三章 关系数据库操作语言-SQL	数据库操作语言SQL的原理及使用	8
第四章 数据库安全性	数据库安全控制的原理及控制策略	4
第五章 数据库完整性	数据库完整性控制的原理和控制操作	4
第六章 关系数据库理论	数据库设计的范式理论及模式分解算法	12
第七章 数据库设计	数据库设计的过程控制及各阶段的设计方法	6
第八章 数据库编程	数据库过程语言机存储过程的原理和编程方法	4
第九章 数据库新技术	数据库领域最新前沿技术	6



## 1.0.2.4 考核方法

考核方法：闭卷笔试

期末成绩：平时成绩（30%）+笔试成绩（70%）



## 1.0.3 班级网站

林子雨主讲《数据库系统原理》2018班级主页

<http://dblab.xmu.edu.cn/post/10147/>

提供课件下载，发布教学信息，保存班级相册



扫一扫访问班级网站  
支持手机浏览



# 1.1 数据库系统概述

**1.1.0 数据库发展史上的杰出贡献者**

**1.1.1 数据、数据库、数据库管理系统、数据库系统**

**1.1.2 数据管理技术的产生与发展**

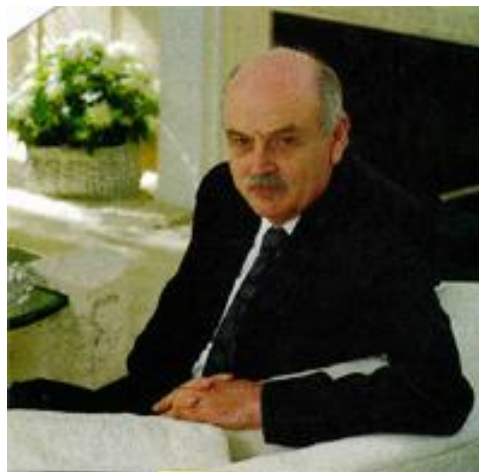
**1.1.3 数据库系统的特点**



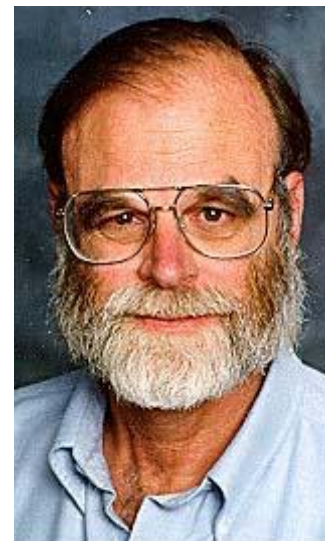
# 1.1.0 数据库发展史上的杰出贡献者



查尔斯·巴赫曼  
Charles W. Bachman  
网状数据库之父  
1973年图灵奖获得者



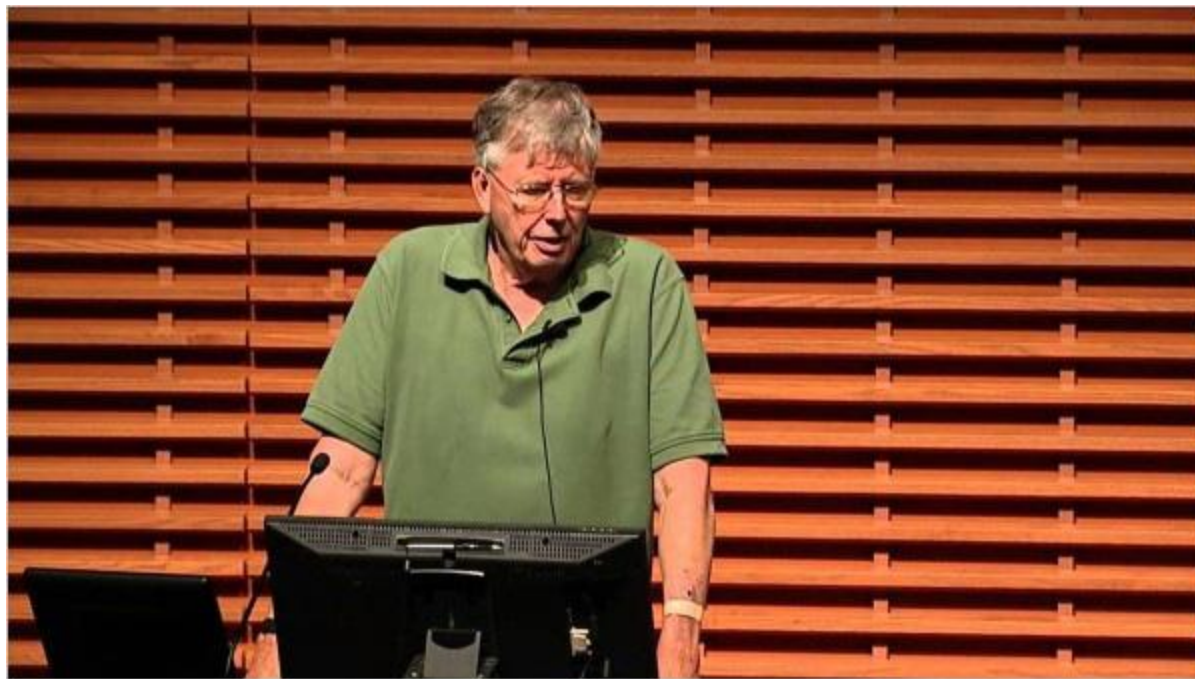
埃德加·科德  
Edgar Frank Codd  
关系数据库之父  
1981年图灵奖获得者



詹姆斯·格雷  
James Gray  
1998年图灵奖获得者  
数据库事务处理专家



# 1.1.0 数据库发展史上的杰出贡献者



- 2015年3月25日，主办方美国计算机协会（ACM）宣布，因“对现代数据库系统底层的概念与实践所做出的基础性贡献”，来自麻省理工学院的教授**Michael Stonebraker**（迈克尔·斯通布雷克）最终摘得被誉为“计算机界诺贝尔奖”的2015年度图灵奖。
- 在1992年提出对象关系数据库模型，更是众多数据库公司的创始人之一，其中包括Ingres、Illustra、Cohera、StreamBase Systems和Vertica等。  
Stonebraker也是SQL Server/Sysbase的奠基人



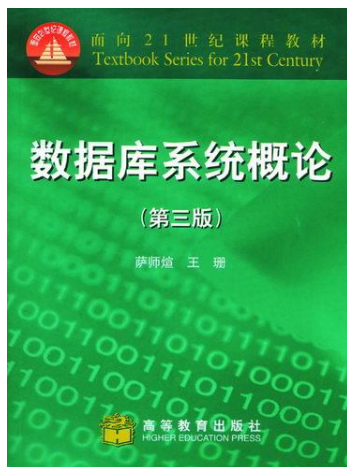
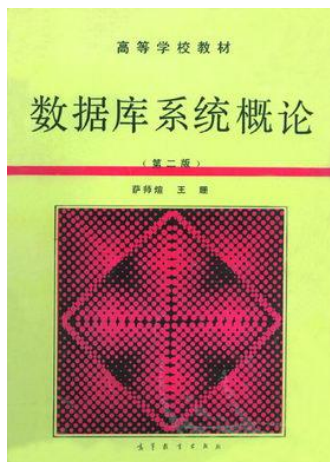
# 1.1.0 数据库发展史上的杰出贡献者



萨师煊，1922年12月27日出生于福州。福建闽侯萨氏家族，源于山西雁门萨氏先祖，已有六七百年的历史。1941年9月考入厦门大学数理系。1950年中国人民大学成立，萨师煊随着华北大学的全体教员一起成为中国人民大学教师。萨师煊是中国人民大学经济信息管理系的创建人，是我国数据库学科的奠基人之一，数据库学术活动的积极倡导者和组织者。萨师煊以他的人格魅力和渊博学识，团结了全国数据库工作者，成为我国数据库界有口皆碑的组织者和带头人，为我国数据库学科的人才培养和技术发展作出了开创性的贡献。

萨师煊(1922.12-2010.7)

中国数据库学科的奠基人之一







## 1.1.2 数据、数据库、数据库管理系统、数据库系统

1.1.2.1 数据 (Data)

1.1.2.2 数据库 (Database)

1.1.2.3 数据库管理系统 (DBMS)

1.1.2.4 数据库系统 (DBS)



## 1.1.2.1 数据

- 数据(Data)是数据库中存储的基本对象
- 数据的定义
  - 描述事物的符号记录
- 数据的种类
  - 文字、图形、图象、声音
- 数据的特点
  - 数据与其语义是不可分的



## 1.1.2.1 数据

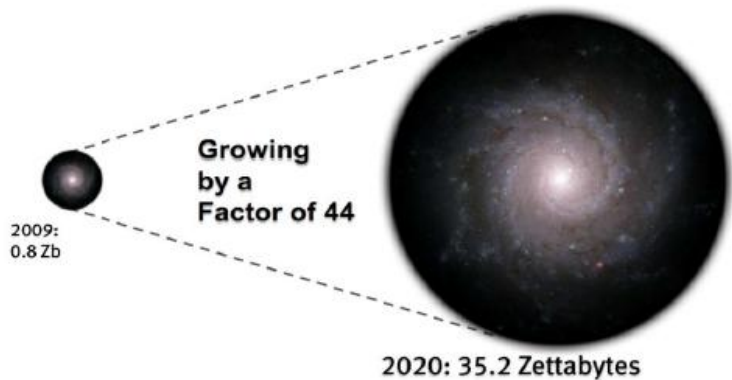
### 数据实例

- 学生成绩表中的学生记录  
(李明, **90**, **95**, **26**, **5**)
- 数据的形式不能完全表达其内容
- 数据的解释
  - 语义：学生姓名、名次、总人数、平均成绩、不及格门数)
  - 解释：李明班级排名第**90**，全班共**95**人，平均成绩**26**，有**5**门不及格



# 1.1.2.1 数据

- 根据IDC作出的估测，数据一直都在以每年50%的速度增长，也就是说每两年就增长一倍（大数据摩尔定律）
- 人类在最近两年产生的数据量相当于之前产生的全部数据量
- 预计到2020年，全球将总共拥有35ZB的数据量，相较于2010年，数据量将增长近30倍



大数据时代

TERABYTE	10 的 12 次方	一块 1TB 硬盘		200,000 照片或 mp3 歌曲
PETABYTE	10 的 15 次方	两个数据中心机柜		16 个 Blackblaze pod 存储单元
EXABYTE	10 的 18 次方	2,000 个机柜		占据一个街区的 4 层数据中心
ZETTABYTE	10 的 21 次方	1000 个数据中心		纽约曼哈顿的 1/5 区域
YOTTABYTE	10 的 24 次方	一百万个数据中心		特拉华州和罗德岛州



## 1.1.2.2 数据库

- 数据库的定义
  - 数据库(Database, 简称DB)是长期储存在计算机内、有组织的、可共享的大量数据集合
- 数据库的特征
  - 数据按一定的数据模型组织、描述和储存
  - 可为各种用户共享
  - 冗余度较小
  - 数据独立性较高
  - 易扩展



## 1.1.2.3 数据库管理系统

- 什么是DBMS

- 数据库管理系统（**Database Management System**，简称**DBMS**）是位于用户与操作系统之间的一层数据管理软件。

- **DBMS**的用途

- 科学地组织和存储数据、高效地获取和维护数据

拓展阅读：2015图领奖获得者迈克尔.斯通布雷克《数据库系统体系结构》  
厦门大学数据库实验室翻译

<http://dblab.xmu.edu.cn/post/architecture-of-a-database-system/>



## 1.1.2.3 数据库管理系统

### DBMS的主要功能

- 数据定义功能

提供数据定义语言(DDL)

定义数据库中的数据对象

- 数据操纵功能:提供数据操纵语言(DML)

操纵数据实现对数据库的基本操作

(查询、插入、删除和修改)



# 1.1.2.3 数据库管理系统

## DBMS的主要功能

### – 数据库的运行管理

保证数据的安全性、完整性

多用户对数据的并发使用

发生故障后的系统恢复

### – 数据库的建立和维护功能(实用程序)

数据库数据批量装载

数据库转储

介质故障恢复

数据库的重组

性能监视等



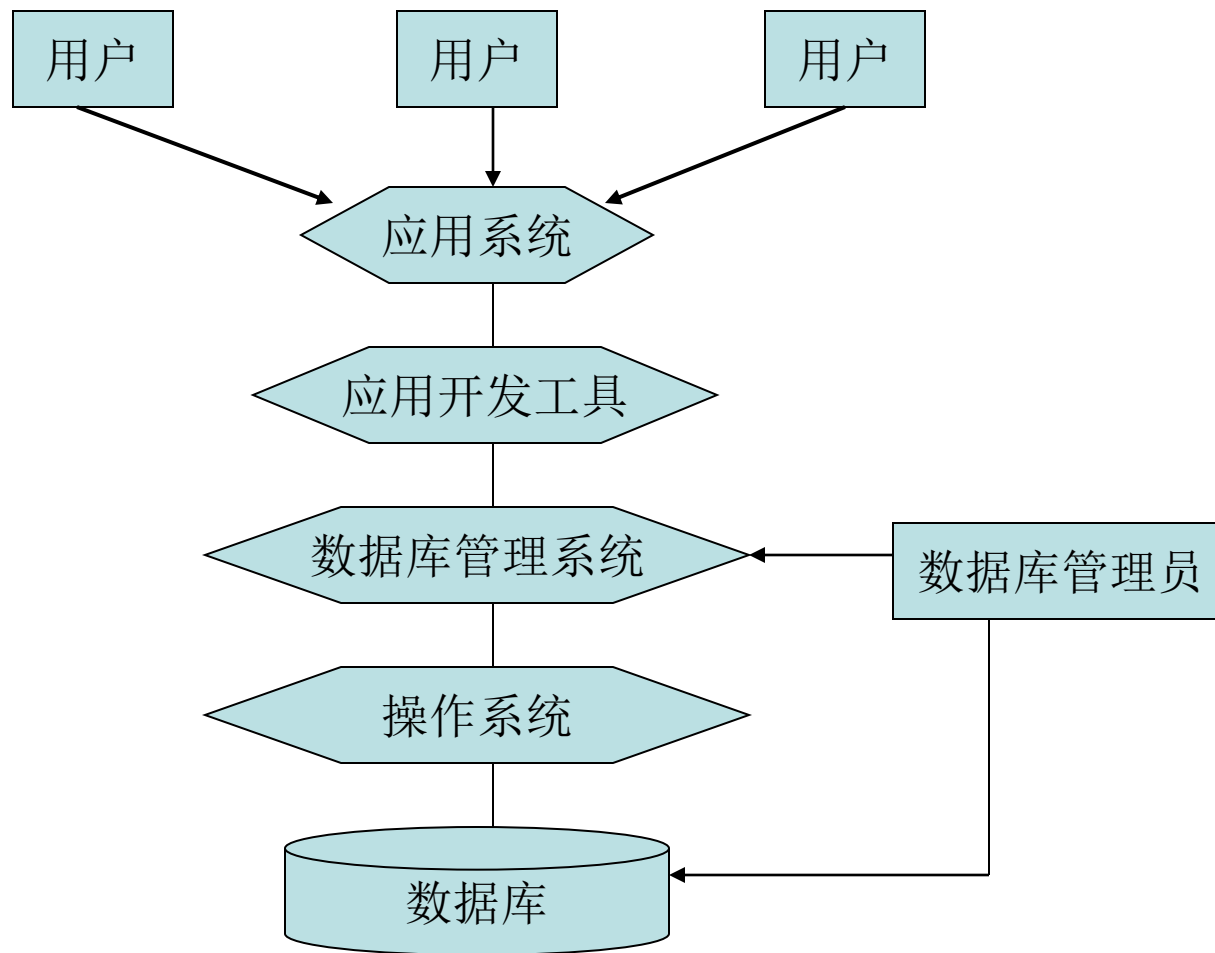


## 1.1.2.4 数据库系统

- 什么是数据库系统
  - 数据库系统（**Database System**，简称**DBS**）是指在计算机系统中引入数据库后的系统。
  - 在不引起混淆的情况下常常把数据库系统简称为数据库。
- 数据库系统的构成
  - 由数据库、数据库管理系统（及其开发工具）、应用系统、数据库管理员（和用户）构成。



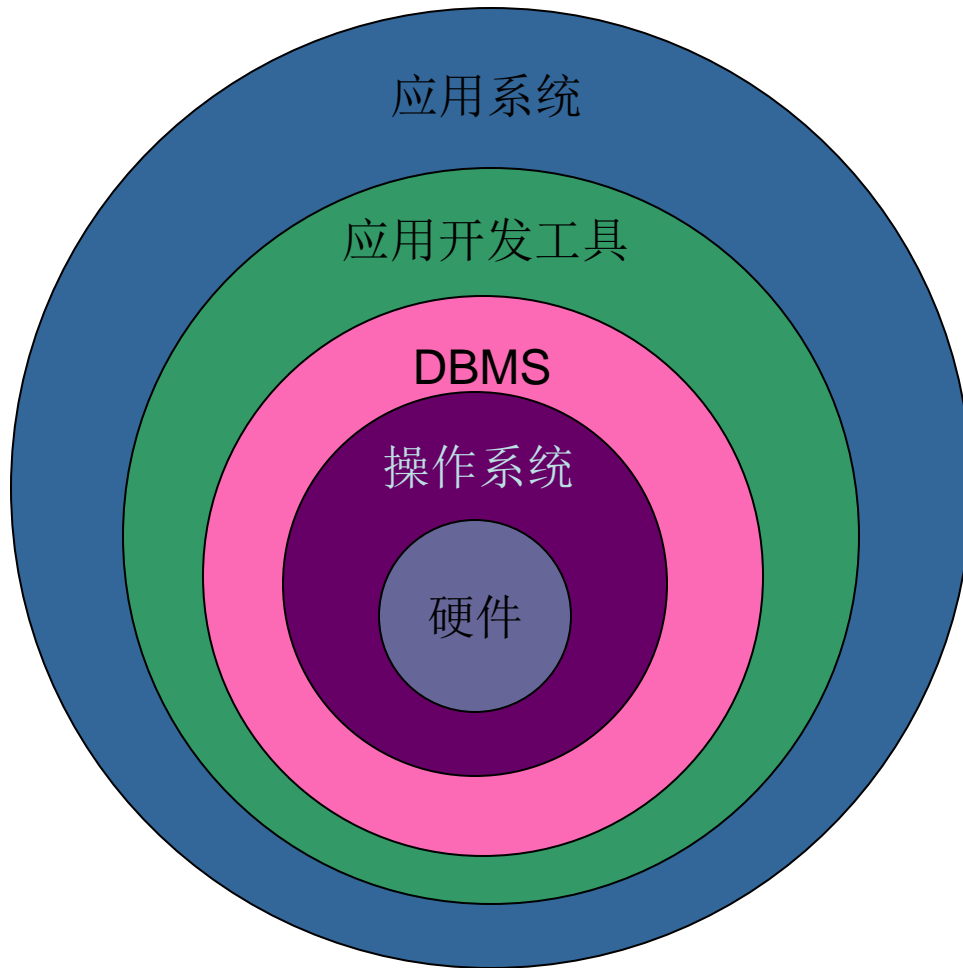
# 1.1.2.4 数据库系统





# 1.1.2.4 数据库系统

## 数据库系统在计算机系统的位置





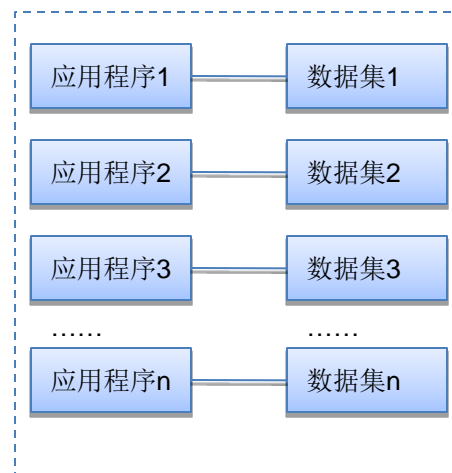
## 1.1.3 数据管理技术的产生与发展

- 什么是数据管理
  - 对数据进行分类、组织、编码、存储、检索和维护，是数据处理的中心问题
- 数据管理技术的发展过程
  - 1.1.3.1 人工管理阶段(40年代中--50年代中)
  - 1.1.3.2 文件系统阶段(50年代末--60年代中)
  - 1.1.3.3 数据库系统阶段(60年代末--现在)



# 1.1.3.1 人工管理阶段

## ■人工管理阶段(40年代中期--50年代中期)



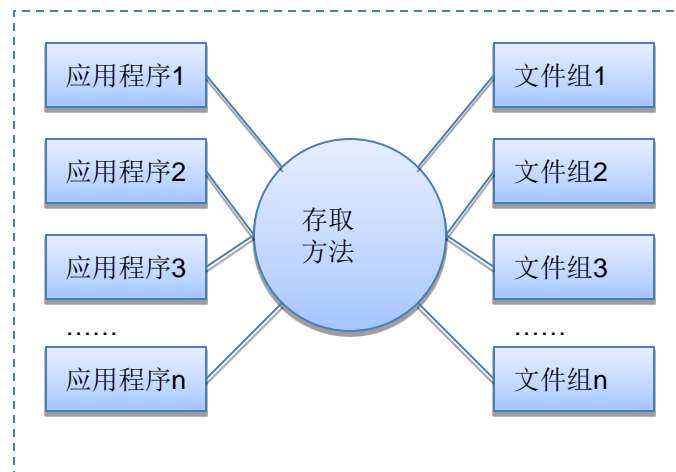
应用程序与数据的对应关系

- 数据不保存
- 应用程序管理数据
- 数据不共享
- 数据不具有独立性



## 1.1.3.2 文件系统阶段

### ■ 文件系统阶段(50年代末期--60年代中期)



应用程序与数据的对应关系

优点:

- 数据可以长期保存
- 由文件系统管理数据

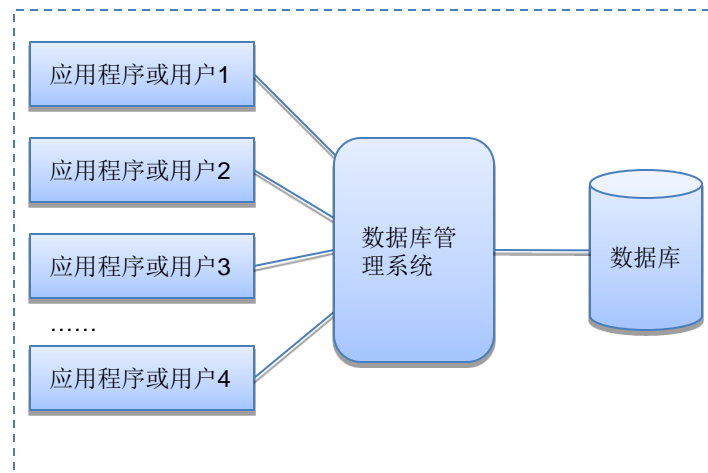
缺点:

- 数据共享性差
- 数据独立性差



# 1.1.3.3 数据库系统阶段

## ■数据库系统阶段(60年代末期--现在)



应用程序与数据的对应关系



## 1.1.4 数据库系统的特点

1.1.4.1 数据结构化

1.1.4.2 数据的共享性高，冗余度低，易扩充

1.1.4.3 数据独立性高

1.1.4.4 数据由**DBMS**统一管理和控制





## 1.1.4.1 数据结构化

- 整体数据的结构化是数据库的主要特征之一
- 数据库中实现的是数据的真正结构化
  - 数据的结构用数据模型描述，无需程序定义和解释
  - 数据可以变长
  - 数据的最小存取单位是数据项



## 1.1.4.2 数据的共享性高，冗余度低，易扩充

- 降低数据的冗余度，节省存储空间
- 避免数据间的不一致性
- 使系统易于扩充



## 1.1.4.3 数据独立性高

- 物理独立性
  - 指用户的应用程序与存储在磁盘上的数据库中数据是相互独立的。当数据的物理存储改变了，应用程序不用改变
- 逻辑独立性
  - 指用户的应用程序与数据库的逻辑结构是相互独立的。数据的逻辑结构改变了，用户程序也可以不变



## 1.1.4.4 数据由DBMS统一管理和控制

- 数据的安全性 (Security) 保护
  - 使每个用户只能按指定方式使用和处理指定数据，保护数据以防止不合法的使用造成的数据的泄密和破坏。
- 数据的完整性 (Integrity) 检查
  - 将数据控制在有效的范围内，或保证数据之间满足一定的关系。
- 并发 (Concurrency) 控制
  - 对多用户的并发操作加以控制和协调，防止相互干扰而得到错误的结果。
- 数据库恢复 (Recovery)
  - 将数据库从错误状态恢复到某一已知的正确状态。



# 1.2 数据模型

**1.2.1 两类数据模型**

**1.2.2 数据模型的组成要素**

**1.2.3 概念模型**

**1.2.4 最常用的数据模型**

**1.2.5 层次模型**

**1.2.6 网状模型**

**1.2.7 关系模型**



## 1.2.1 两类数据模型

- 在数据库中用数据模型这个工具来抽象、表示和处理现实世界中的数据和信息。
- 数据模型应满足三方面要求：
  - 能比较真实地模拟现实世界
  - 容易为人所理解
  - 便于在计算机上实现

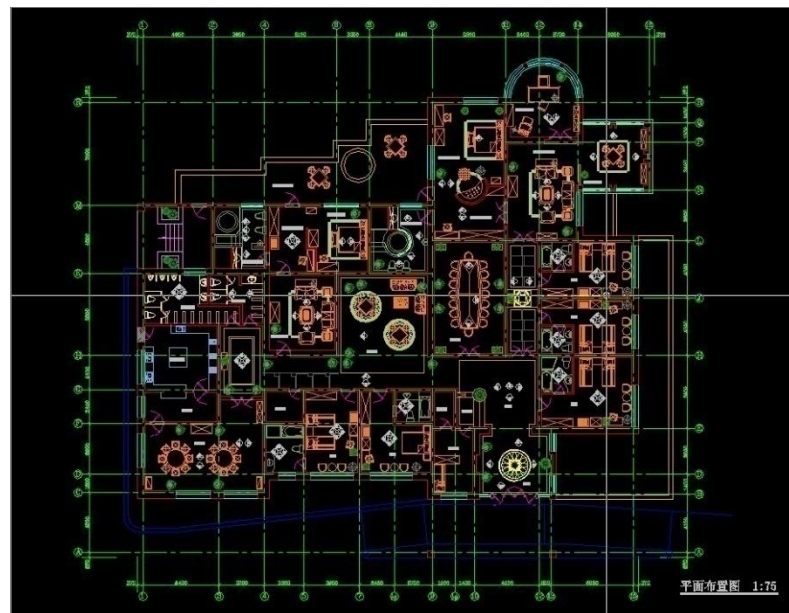


## 1.2.1 两类数据模型

如同在建筑设计和施工的不同阶段需要不同的图纸，在开发实施数据库应用系统中也需要使用不同的数据模型



建筑效果图



施工图



## 1.2.1 两类数据模型

- 数据模型分成两个不同的层次
  - (1) **概念模型**：也称信息模型，它是按用户的观点来对数据和信息建模。
  - (2) **数据模型**：包括逻辑模型和物理模型。
    - 逻辑模型主要包括网状模型、层次模型、关系模型等，它是按计算机系统的观点对数据建模，主要用于**DBMS**的实现。
    - 物理模型是对数据最低层的抽象，描述数据在系统内部的表示方式和存取方法，是面向计算机系统的。



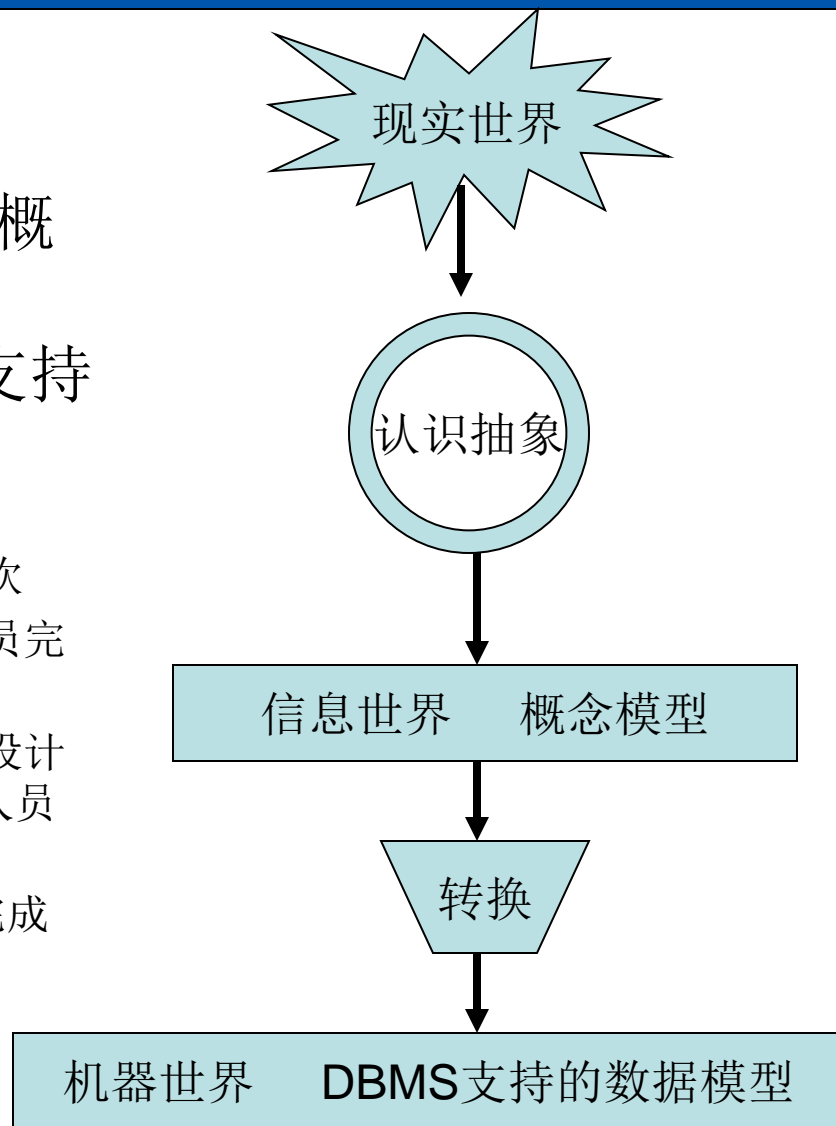


# 1.2.1 两类数据模型

## 客观对象的抽象过程---两步抽象

- 现实世界中的客观对象抽象为概念模型；
- 把概念模型转换为某一DBMS支持的数据模型。

- 概念模型是现实世界到机器世界的一个中间层次
- 从现实世界到概念模型的转换由数据库设计人员完成
- 从概念模型到逻辑模型的转换，可以由数据库设计人员完成，也可以由数据库设计工具协助设计人员完成
- 从逻辑模型到物理模型的转换一般是由DBMS完成的





## 1.2.2 数据模型的组成要素

**1.2.2.1 数据结构**

**1.2.2.2 数据操作**

**1.2.2.3 数据的完整性约束条件**



## 1.2.2.1 数据结构

- 什么是数据结构
  - 对象类型的集合
- 两类对象
  - 与数据类型、内容、性质有关的对象
  - 与数据之间联系有关的对象
- 数据结构是对系统静态特性的描述



## 1.2.2.2 数据操作

- 数据操作
  - 对数据库中各种对象（型）的实例（值）允许执行的操作及有关的操作规则
- 数据操作的类型
  - 查询
  - 更新（包括插入、删除、修改）



## 1.2.2.2 数据操作

- 数据模型对操作的定义
  - 操作的确切含义
  - 操作符号
  - 操作规则（如优先级）
  - 实现操作的语言
- 数据操作是对系统动态特性的描述。



## 1.2.2.3 数据的完整性约束条件

一组完整性规则的集合。完整性规则是给定的数据模型中数据及其联系所具有的制约和储存规则，用以限定符合数据模型的数据库状态以及状态的变化，以保证数据的正确、有效、相容。



## 1.2.3 概念模型

1.2.3.0 概念模型介绍

1.2.3.1 信息世界中的基本概念

1.2.3.2 两个实体型之间的联系

1.2.3.3 两个以上的实体型之间的联系

1.2.3.4 单个实体型内的联系

1.2.3.5 概念模型的一种表示方法：实体-联系方法

1.2.3.6 一个实例



## 1.2.3.0 概念模型介绍

- 概念模型的用途
  - 概念模型用于信息世界的建模
  - 是现实世界到机器世界的一个中间层次
  - 是数据库设计的有力工具
  - 数据库设计人员和用户之间进行交流的语言
- 对概念模型的基本要求
  - 较强的语义表达能力，能够方便、直接地表达应用中的各种语义知识
  - 简单、清晰、易于用户理解。





## 1.2.3.1 信息世界中的基本概念

### (1) 实体 (Entity)

客观存在并可相互区别的事物称为实体

可以是具体的人、事、物，也可以是抽象的概念或联系

### (2) 属性 (Attribute)

实体所具有的某一特性称为属性。

一个实体可以由若干个属性来刻画。

### (3) 码 (Key)

唯一标识实体的属性集称为码。

### (4) 域 (Domain)

属性的取值范围称为该属性的域



## 1.2.3.1 信息世界中的基本概念

### (5) 实体型 (Entity Type)

用实体名及其属性名集合来抽象和刻画同类实体称为实体型  
学生 (学号, 姓名, 性别, 年龄, 系, 年级)

(1001, 张三, 男, 20, 计算机, 2002)

### (6) 实体集 (Entity Set)

同型实体的集合称为实体集

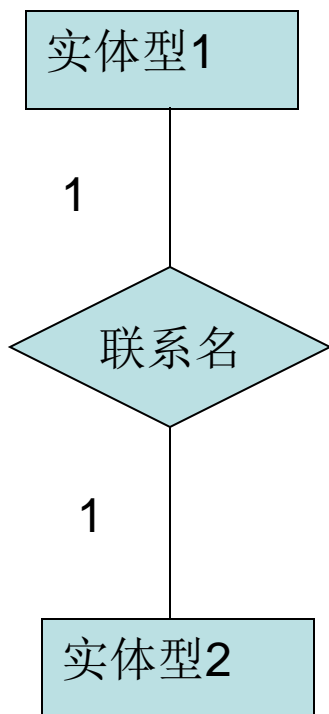
全体学生, 女学生

### (7) 联系 (Relationship)

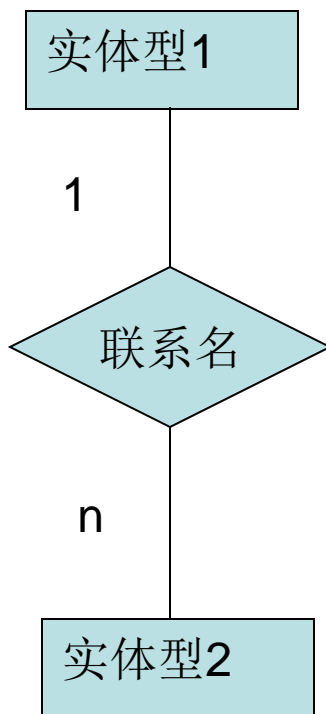
现实世界中事物内部以及事物之间的联系, 在信息世界中反映为实体 (型) 内部的联系和实体 (型) 之间的联系



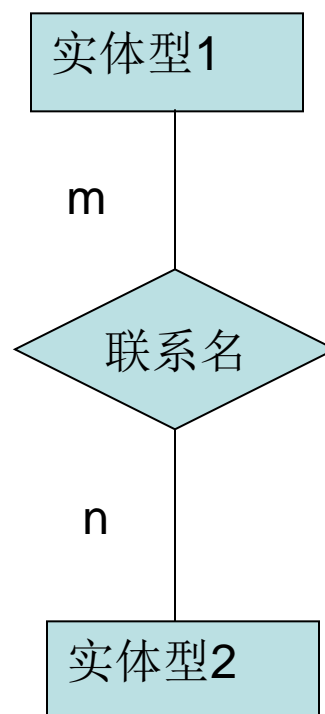
## 1.2.3.2 两个实体型之间的联系



1:1联系



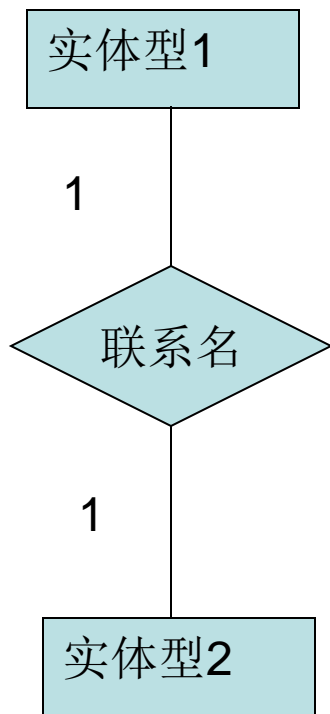
1:n联系



m:n联系



## 1.2.3.2 两个实体型之间的联系

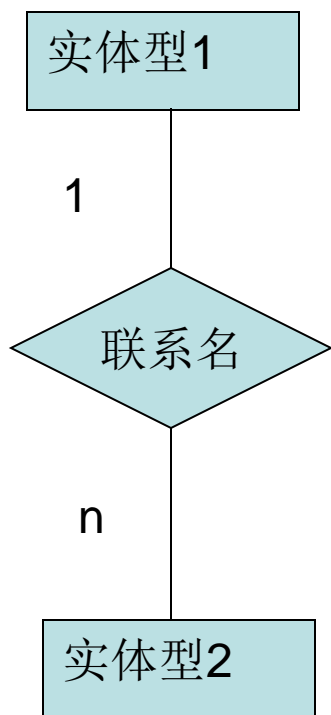


1:1联系

- 一对一联系
  - 如果对于实体集A中的每一个实体，实体集B中至多有一个实体与之联系，反之亦然，则称实体集A与实体集B具有一对一联系。记为1:1。
  - 实例
    - 班级与班长之间的联系：
      - 一个班级只有一个班长
      - 一个班长只在一个班中任职



## 1.2.3.2 两个实体型之间的联系



1:n联系

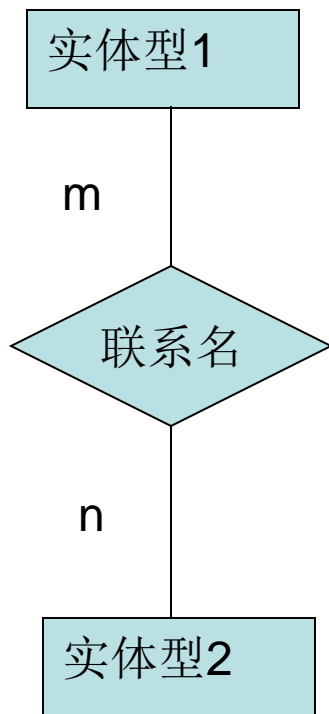
- 一对多联系
  - 如果对于实体集A中的每一个实体，实体集B中有n个实体 ( $n \geq 0$ ) 与之联系，反之，对于实体集B中的每一个实体，实体集A中至多只有一个实体与之联系，则称**实体集A与实体集B**有一对多联系

记为1:n

- 实例
  - 班级与学生之间的联系：  
一个班级中有若干名学生，  
每个学生只在一个班级中学习



## 1.2.3.2 两个实体型之间的联系



m:n联系

- 多对多联系 (m:n)
  - 如果对于实体集**A**中的每一个实体，实体集**B**中有**n**个实体 ( $n \geq 0$ ) 与之联系，反之，对于实体集**B**中的每一个实体，实体集**A**中也有**m**个实体 ( $m \geq 0$ ) 与之联系，则称实体集**A**与实体**B**具有多对多联系。记为**m:n**
  - 实例
    - 课程与学生之间的联系：
      - 一门课程同时有若干个学生选修
      - 一个学生可以同时选修多门课程



# 1.2.3.3 两个以上的实体型之间的联系

- 多个实体型间的一对多联系

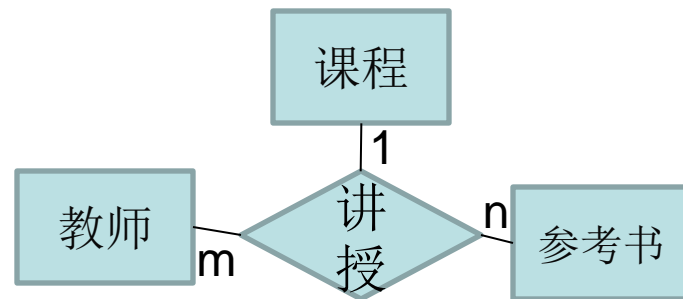
- 若实体集 $E_1, E_2, \dots, E_n$ 存在联系, 对于实体集 $E_j$  ( $j=1, 2, \dots, i-1, i+1, \dots, n$ ) 中的给定实体, 最多只和 $E_i$ 中的一个实体相联系, 则我们说 $E_i$ 与 $E_1, E_2, \dots, E_{i-1}, E_{i+1}, \dots, E_n$ 之间的联系是一对多的。

- 实例

课程、教师与参考书三个实体型

如果一门课程可以有若干个教师讲授, 使用若干本参考书, 每一个教师只讲授一门课程, 每一本参考书只供一门课程使用

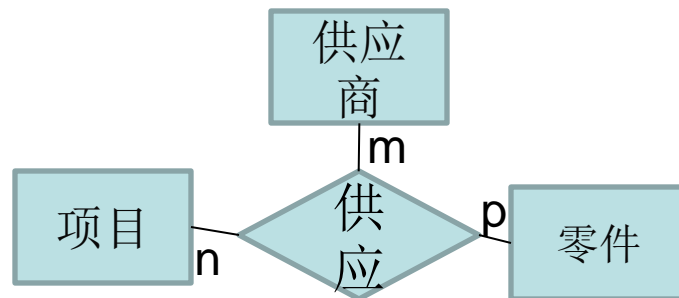
课程与教师、参考书之间的联系是一对多的





# 1.2.3.3 两个以上的实体型之间的联系

多个实体型之间的多对多联系  
有三个实体型：供应商、项目和零件。一个供应商可以供给多个项目零件，每个项目可以使用多个供应商供应的零件，每种零件可以由不同供应商供给







# 1.2.3.4 单个实体型内的联系

- 一对多联系

- 实例

职工实体集内部具有领导与被领导的联系

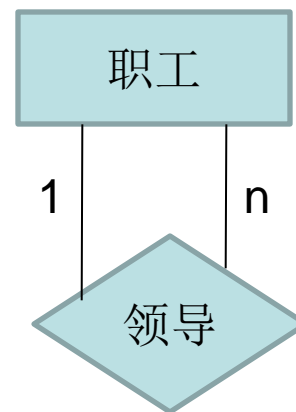
某一职工（干部）“领导”若干名职工

一个职工仅被另外一个职工直接领导

这是一对多的联系

- 一对一联系

- 多对多联系





## 1.2.3.5 概念模型的一种表示方法：实体-联系方法

- 概念模型的表示方法很多
- 实体—联系方法(**E-R方法**)
  - 用**E-R图**来描述现实世界的概念模型
  - **E-R方法**也称为**E-R模型**



## 1.2.3.5 概念模型的一种表示方法：实体-联系方法

- 实体型
  - 用矩形表示，矩形框内写明实体名。

学生

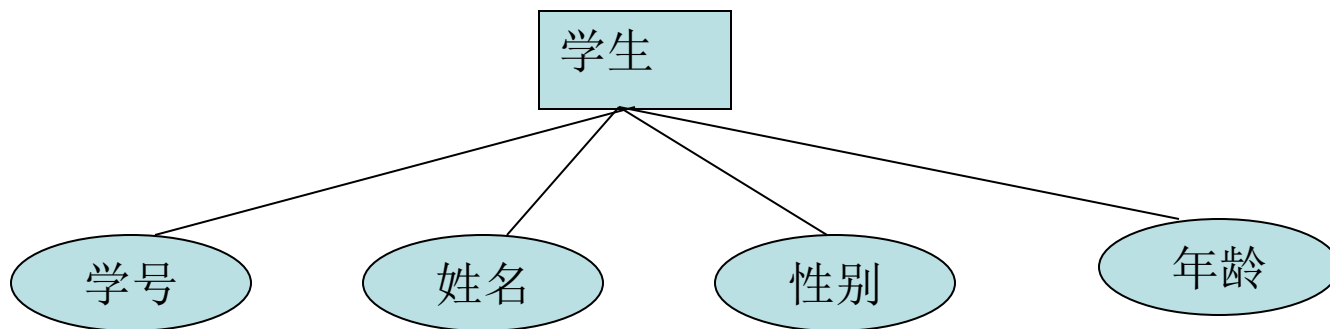
教师



## 1.2.3.5 概念模型的一种表示方法：实体-联系方法

- 属性

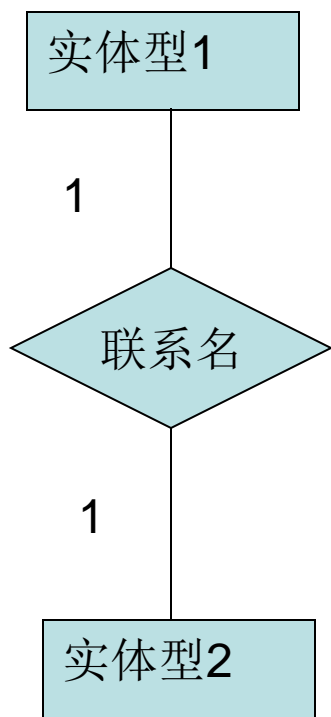
- 用椭圆形表示，并用无向边将其与相应的实体连接起来



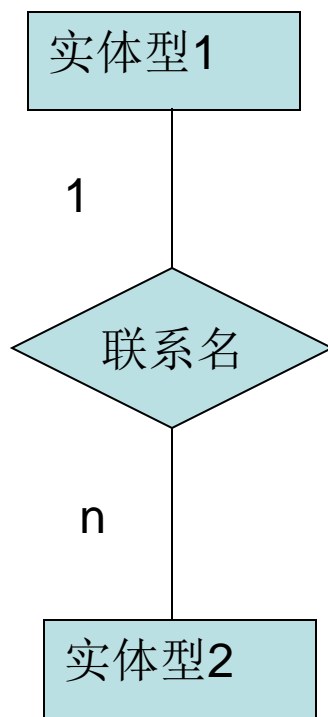


## 1.2.3.5 概念模型的一种表示方法：实体-联系方法

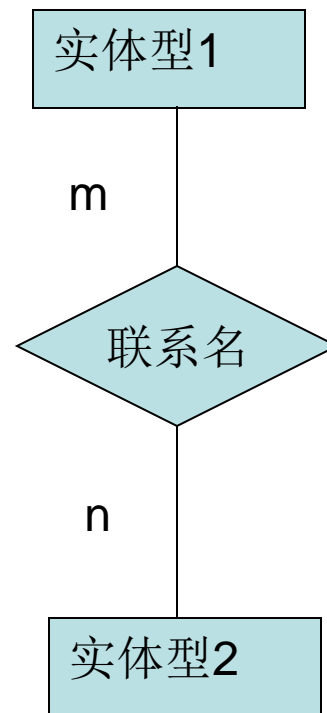
- 联系：用菱形表示，菱形框内写明联系名，并用无向边分别与有关实体连接起来，同时无向边旁标上联系的类型（1:1、1:n或m:n）。下面是两个实体型之间的联系：



1:1联系



1:n联系

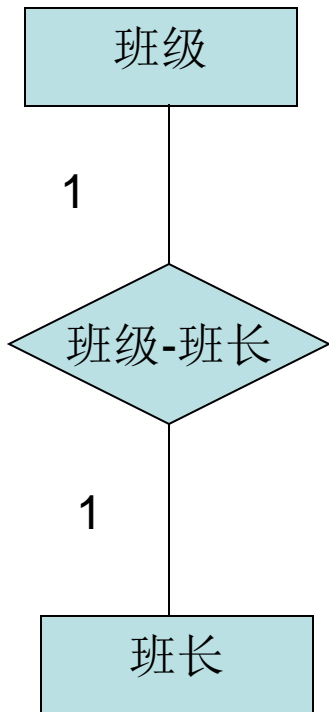


m:n联系

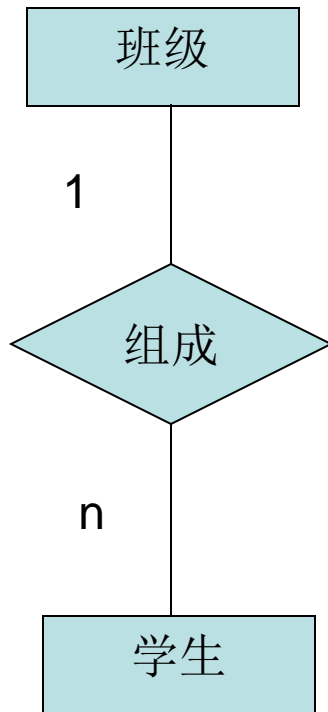


# 1.2.3.5 概念模型的一种表示方法：实体-联系方法

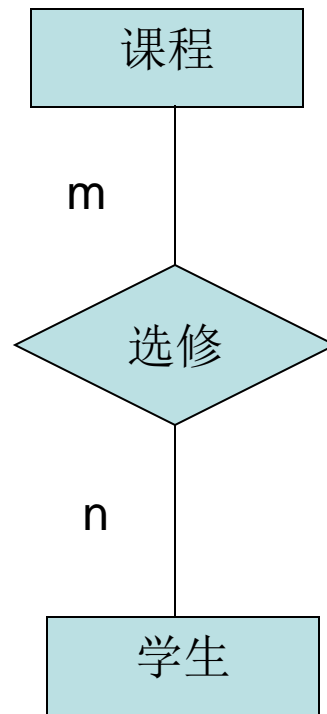
两个实体型之间的联系的代表方法实例



1:1联系



1:n联系

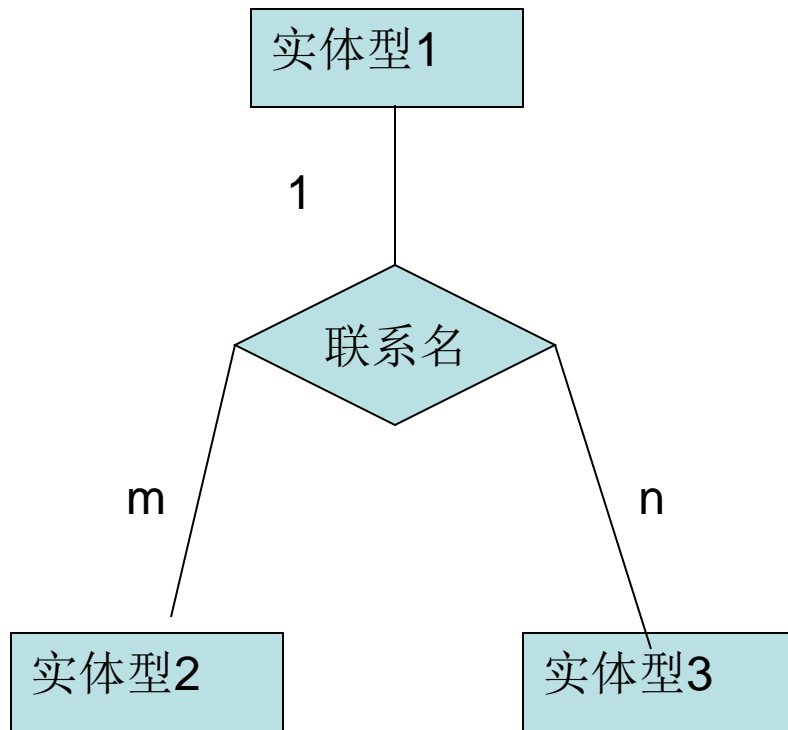


m:n联系

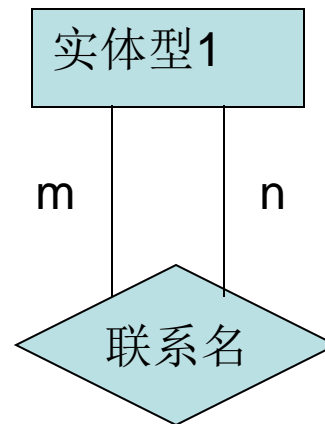


# 1.2.3.5 概念模型的一种表示方法：实体-联系方法

## 多个实体型之间的联系



## 单个实体型内部的联系



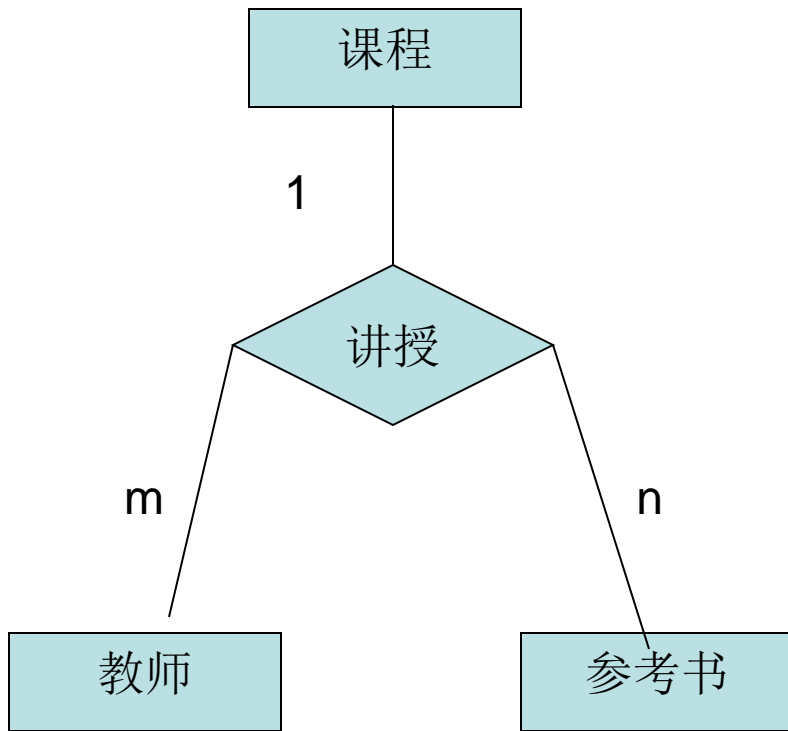
同一实体型内部的  
m:n联系

## 多个实体型间的1:n联系



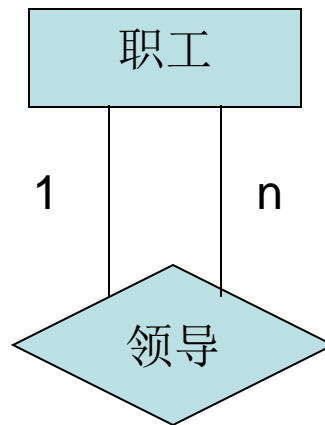
# 1.2.3.5 概念模型的一种表示方法：实体-联系方法

多个实体型之间的联系实例



多个实体型间的1:n联系

单个实体型内部的联系实例



同一实体型内部的  
1:n联系

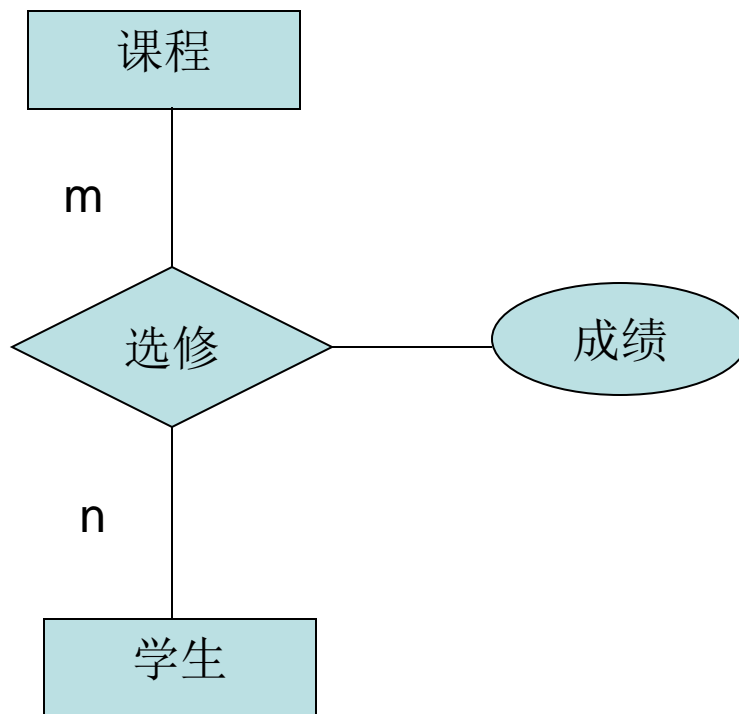




## 1.2.3.5 概念模型的一种表示方法：实体-联系方法

- 联系

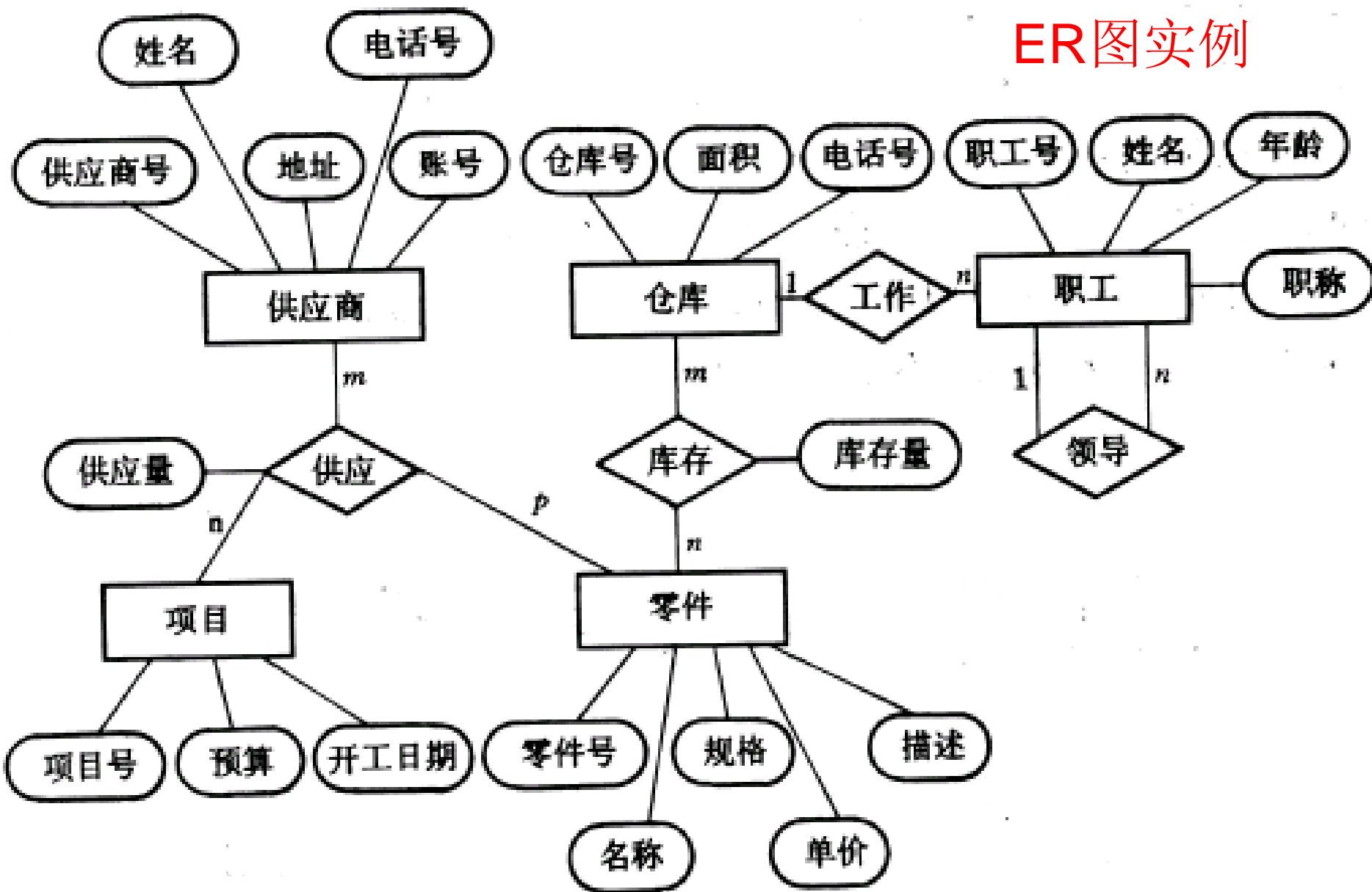
- 联系的属性：联系本身也是一种实体型，也可以有属性。如果一个联系具有属性，则这些属性也要用无向边与该联系连接起来





# 1.2.3.6 一个实例

## ER图实例





## 1.2.4 最常用的数据模型

- 层次模型
- 网状模型
- 关系模型
- 面向对象模型
- 对象关系模型



## 1.2.5 层次模型

- 层次模型数据结构

满足下面两个条件的基本层次联系的集合为层次模型。

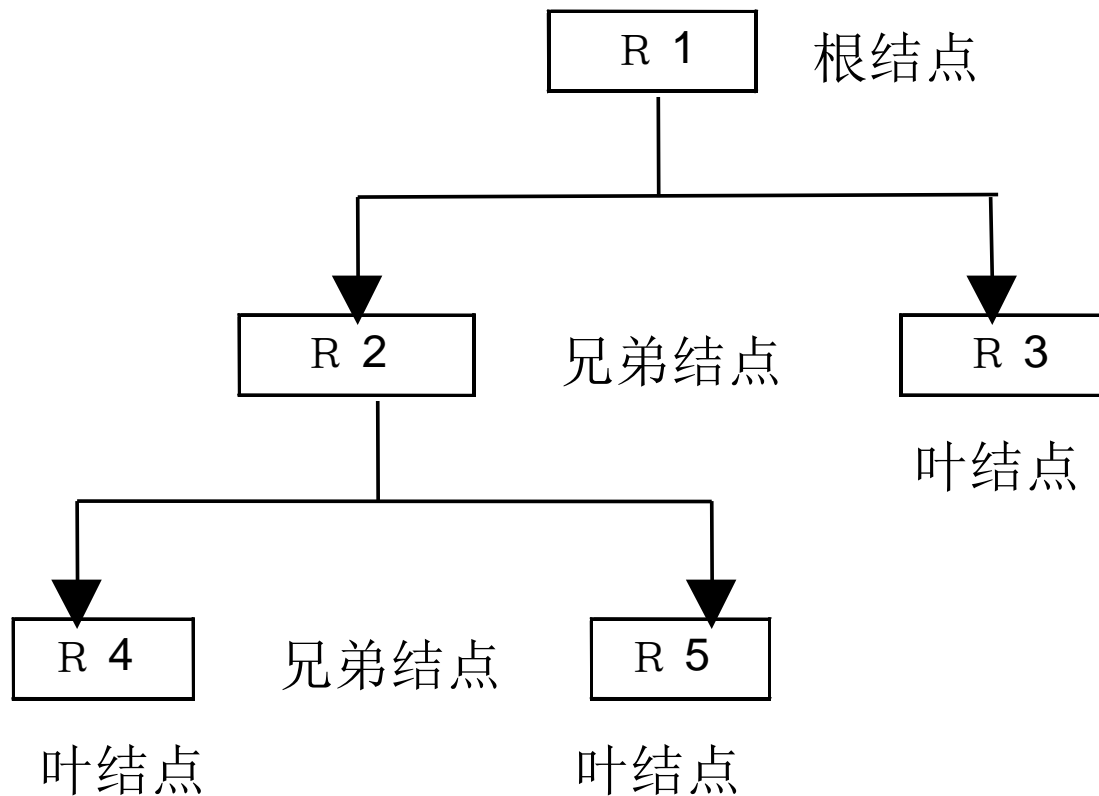
1. 有且只有一个结点没有双亲结点，这个结点称为根结点
2. 根以外的其它结点有且只有一个双亲结点

- 层次模型中的几个术语

- 根结点，双亲结点，兄弟结点，叶结点



# 1.2.5 层次模型





## 1.2.5 层次模型

- 表示方法

**实体型：**用记录类型描述。

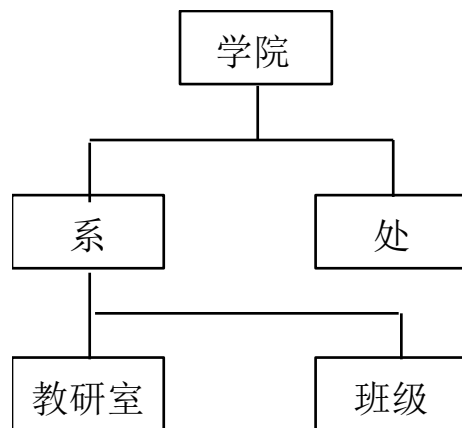
每个结点表示一个记录类型。

**属性：**用字段描述。每个记录类型可包含若干个字段。

**联系：**用结点之间的连线表示记录（类）型之间的

**一对多的联系**

实例：





## 1.2.5 层次模型

- 特点
  - 结点的双亲是唯一的
  - 只能直接处理一对多的实体联系
  - 每个记录类型定义一个排序字段，也称为码字段
  - 任何记录值只有按其路径查看时，才能显出它的全部意义
  - 没有一个子女记录值能够脱离双亲记录值而独立存在



## 1.2.5 层次模型

- 多对多联系在层次模型中的表示
  - 用层次模型**间接**表示多对多联系
  - 方法
    - 将多对多联系**分解**成一对多联系
  - 分解方法
    - 冗余结点法
    - 虚拟结点法





## 1.2.5 层次模型

### 层次模型的数据操纵

- 查询
- 插入
- 删除
- 更新



## 1.2.5 层次模型

### 层次模型的完整性约束

- 无相应的双亲结点值就不能插入子女结点值
- 如果删除双亲结点值，则相应的子女结点值也被同时删除
- 更新操作时，应更新所有相应记录，以保证数据的一致性



## 1.2.5 层次模型

- 优点
  - 层次数据模型简单，对具有一对多的层次关系的部门描述自然、直观，容易理解
  - 性能优于关系模型，不低于网状模型
  - 层次数据模型提供了良好的完整性支持
- 缺点
  - 多对多联系表示不自然
  - 对插入和删除操作的限制多
  - 查询子女结点必须通过双亲结点
  - 层次命令趋于程序化



## 1.2.5 层次模型

### 典型的层次数据库系统

#### IMS数据库管理系统

- 第一个大型商用**DBMS**
- **1968**年推出
- **IBM**公司研制



## 1.2.6 网状模型

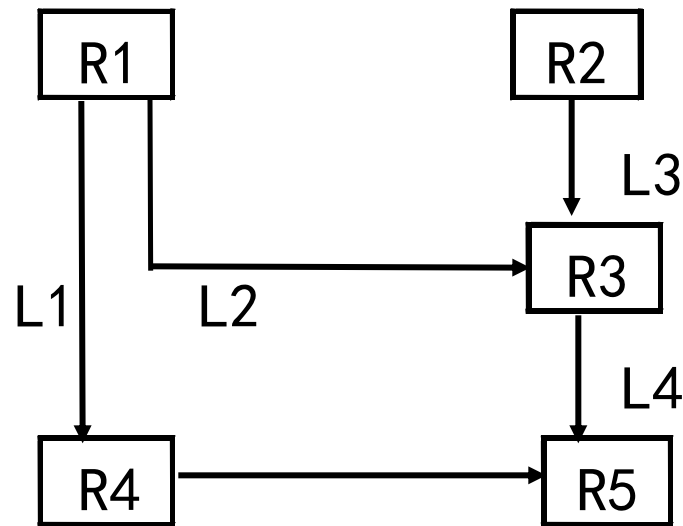
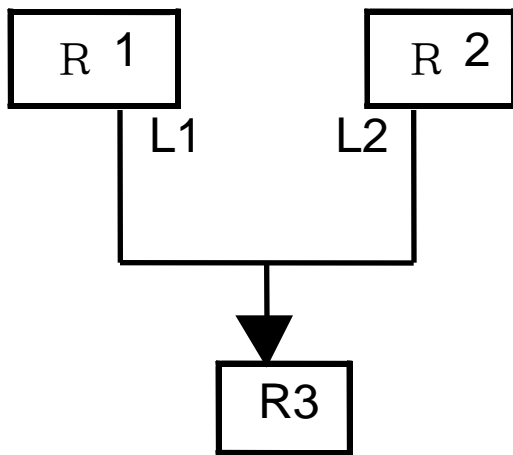
- 网状模型数据结构

满足下面两个条件的基本层次联系的集合为网状模型。

1. 允许一个以上的结点无双亲；
2. 一个结点可以有多个的双亲。



# 1.2.6 网状模型





## 1.2.6 网状模型

- 表示方法（与层次数据模型相同）

**实体型：**用记录类型描述。

每个结点表示一个记录类型。

**属性：**用字段描述。

每个记录类型可包含若干个字段。

**联系：**用结点之间的连线表示记录（类）类型之间的一对多的父子联系。



## 1.2.6 网状模型

- 网状模型与层次模型的区别
  - 网状模型允许多个结点没有双亲结点
  - 网状模型允许结点有多个双亲结点
  - 网状模型允许两个结点之间有多种联系（复合联系）
  - 网状模型可以更直接地去描述现实世界
  - 层次模型实际上是网状模型的一个特例





## 1.2.6 网状模型

### 网状模型的数据操纵

- 查询
- 插入
- 删除
- 更新



## 1.2.6 网状模型

- 完整性约束条件
  - 允许插入尚未确定双亲结点值的子女结点值
  - 允许只删除双亲结点值
- 优点
  - 能够更为直接地描述现实世界，如一个结点可以有多个双亲
  - 具有良好的性能，存取效率较高
- 缺点
  - 结构比较复杂，而且随着应用环境的扩大，数据库的结构就变得越来越复杂，不利于最终用户掌握
  - **DDL、DML**语言复杂，用户不容易使用



## 1.2.6 网状模型

### 典型的网状数据库系统

- DBTG系统，亦称CODASYL系统
  - 由**DBTG**提出的一个系统方案
  - 奠定了数据库系统的基本概念、方法和技术
  - **70年代**推出
- 实际系统
  - **Cullinet Software Inc.**公司的 **IDMS**
  - **Univac**公司的 **DMS1100**
  - **Honeywell**公司的**IDS/2**
  - **HP**公司的**IMAGE**



## 1.2.7 关系模型

- 最重要的一种数据模型。也是目前主要采用的数据模型
- **1970**年由美国**IBM**公司**San Jose**研究室的研究员**E.F.Codd**提出
- 本课程的重点



## 1.2.7 关系模型

### 关系数据模型的数据结构

- 在用户观点下，关系模型中数据的逻辑结构是一张二维表，它由行和列组成。

学生登记表

学号	姓名	年龄	性别	系名	年级
95004	王小明	19	女	社会学	95
95006	黄大鹏	20	男	商品学	95
95008	张文斌	18	女	法律学	95
...	...	...	...	...	...



## 1.2.7 关系模型

### 关系模型的基本概念

#### – 关系 (**Relation**)

一个关系对应通常说的一张表。

#### – 元组 (**Tuple**)

表中的一行即为一个元组。

#### – 属性 (**Attribute**)

表中的一列即为一个属性，给每一个属性起一个名称即属性名。



## 1.2.7 关系模型

### 关系模型的基本概念

- 主码 (**Key**)

表中的某个属性组，它可以唯一确定一个元组。



- 域 (**Domain**)

属性的取值范围。

- 分量

元组中的一个属性值。

- 关系模式

对关系的描述

关系名 (属性1, 属性2, ..., 属性n)

学生 (学号, 姓名, 年龄, 性别, 系, 年级)



## 1.2.7 关系模型

- 实体及实体间的联系的表示方法
  - 实体型：直接用关系（表）表示。
  - 属性：用属性名表示。
  - 一对一联系：隐含在实体对应的关系中。
  - 一对多联系：隐含在实体对应的关系中。
  - 多对多联系：直接用关系表示。





## 1.2.7 关系模型

### 例1

学生、系、系与学生之间的一对多联系：

学生 (学号, 姓名, 年龄, 性别, 年级 **系号**)

系 (系号, 系名, 办公地点)



## 1.2.7 关系模型

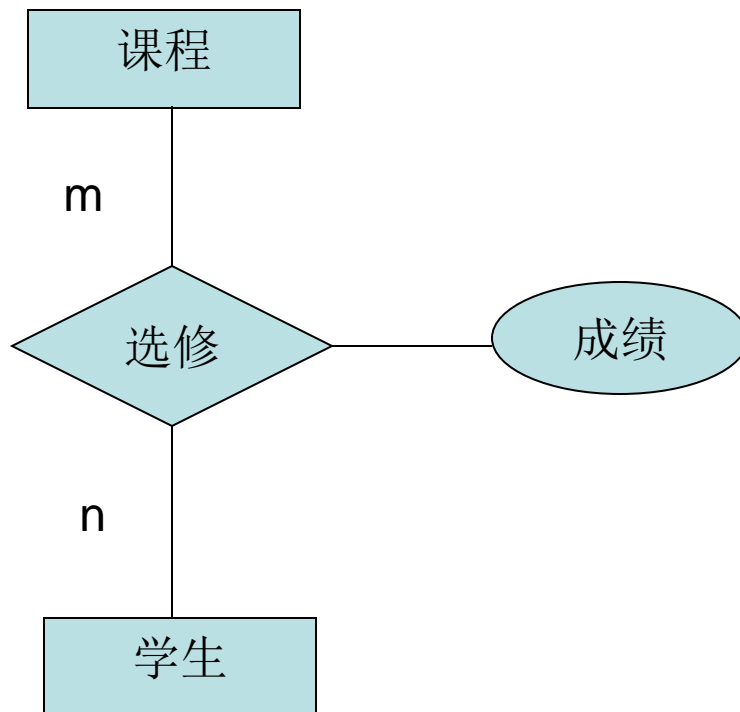
### 例2

学生、课程、学生与课程之间的多对多联系：

学生（学号，姓名，年龄，性别，系号，年级）

课程（课程号，课程名，学分）

选修（学号，课程号，成绩）





## 1.2.7 关系模型

- 关系必须是规范化的，满足一定的规范条件

最基本的规范条件：关系的每一个分量必须是一个不可分的数据项。

职工号	姓名	职称	工资			扣除		实发
			基本	工龄	职务	房租	水电	
86051	陈平	讲师	105	9.5	15	6	12	11.5
.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.



## 1.2.7 关系模型

### 关系模型的数据操纵

- 查询、插入、删除、更新
- 数据操作是集合操作，操作对象和操作结果都是关系，即若干元组的集合
- 存取路径对用户隐蔽，用户只要指出“干什么”，不必详细说明“怎么干”



## 1.2.7 关系模型

### 关系模型的完整性约束

- 实体完整性
- 参照完整性
- 用户定义的完整性



## 1.2.7 关系模型

### 关系数据模型的存储结构

- 表以文件形式存储
- 有的**DBMS**一个表对应一个操作系统文件
- 有的**DBMS**自己设计文件结构



## 1.2.7 关系模型

- 优点
  - 建立在严格的数学概念的基础上
  - 概念单一。数据结构简单、清晰，用户易懂易用
    - 实体和各类联系都用关系来表示。
    - 对数据的检索结果也是关系。
  - 关系模型的存取路径对用户透明
    - 具有更高的数据独立性，更好的安全保密性
    - 简化了程序员的工作和数据库开发建立的工作



## 1.2.7 关系模型

### 典型的关系数据库系统

- **ORACLE**
- **SYBASE**
- **INFORMIX**
- **DB2**
- **SQL SERVER**
- **PBASE**
- **EasyBase**
- **DM/2**
- **OpenBase**





# 第1章 作业

## • 课后作业

教材P38 习题12、13

再将E-R模型转换为关系模型



# 1.3 数据库系统结构

- 1.3.0 数据库系统结构概述
- 1.3.1 数据库系统模式的概念
- 1.3.2 数据库系统的三级模式结构
- 1.3.3 数据库的二级映像功能与数据独立性



## 1.3.0 数据库系统结构概述

- 从**数据库管理系统**角度看：三级模式结构
- 从**数据库开发人员**角度看，数据库系统的结构分为单用户结构、主从式结构、分布式结构、客户/服务器、浏览器/应用服务器/数据库服务器



## 1.3.1 数据库系统模式的概念

- “型” 和 “值” 的概念

- 型 (Type)

- 对某一类数据的结构和属性的说明

- 值 (Value)

- 是型的一个具体赋值

- 例如：学生记录

- 记录型：

- (学号，姓名，性别，系别，年龄，籍贯)

- 该记录型的一个记录值：

- (**900201**，李明，男，计算机，**22**，江苏)

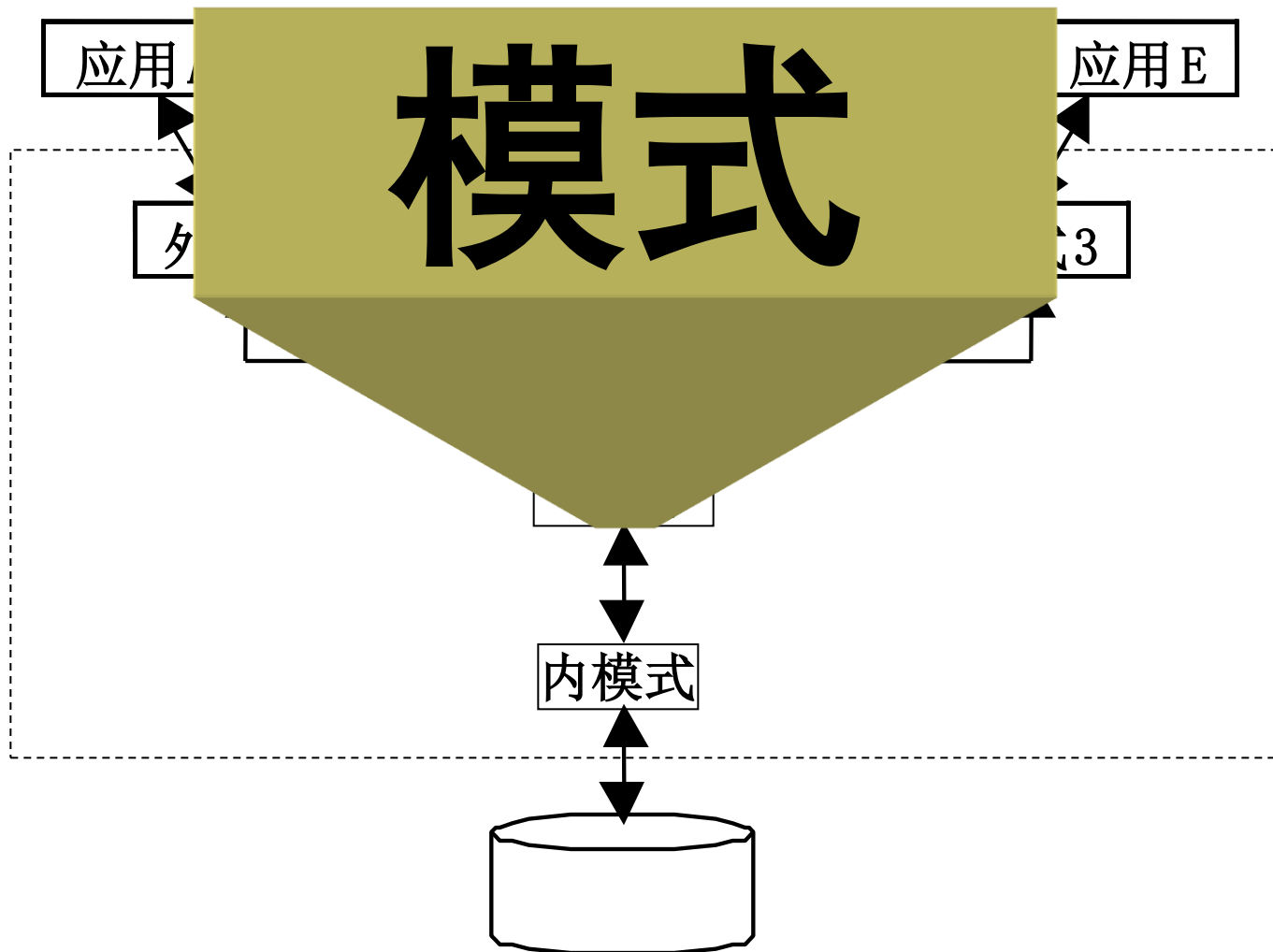


# 1.3.1 数据库系统模式的概念

- 模式（Schema）
  - 数据库逻辑结构和特征的描述
  - 是型的描述
  - 反映的是数据的结构及其联系
  - 模式是相对稳定的
- 模式的一个实例（Instance）
  - 模式的一个具体值
  - 反映数据库某一时刻的状态
  - 同一个模式可以有很多实例
  - 实例随数据库中的数据更新而变动



# 1.3.2 数据库系统的三级模式结构





## 1.3.2 数据库系统的三级模式结构

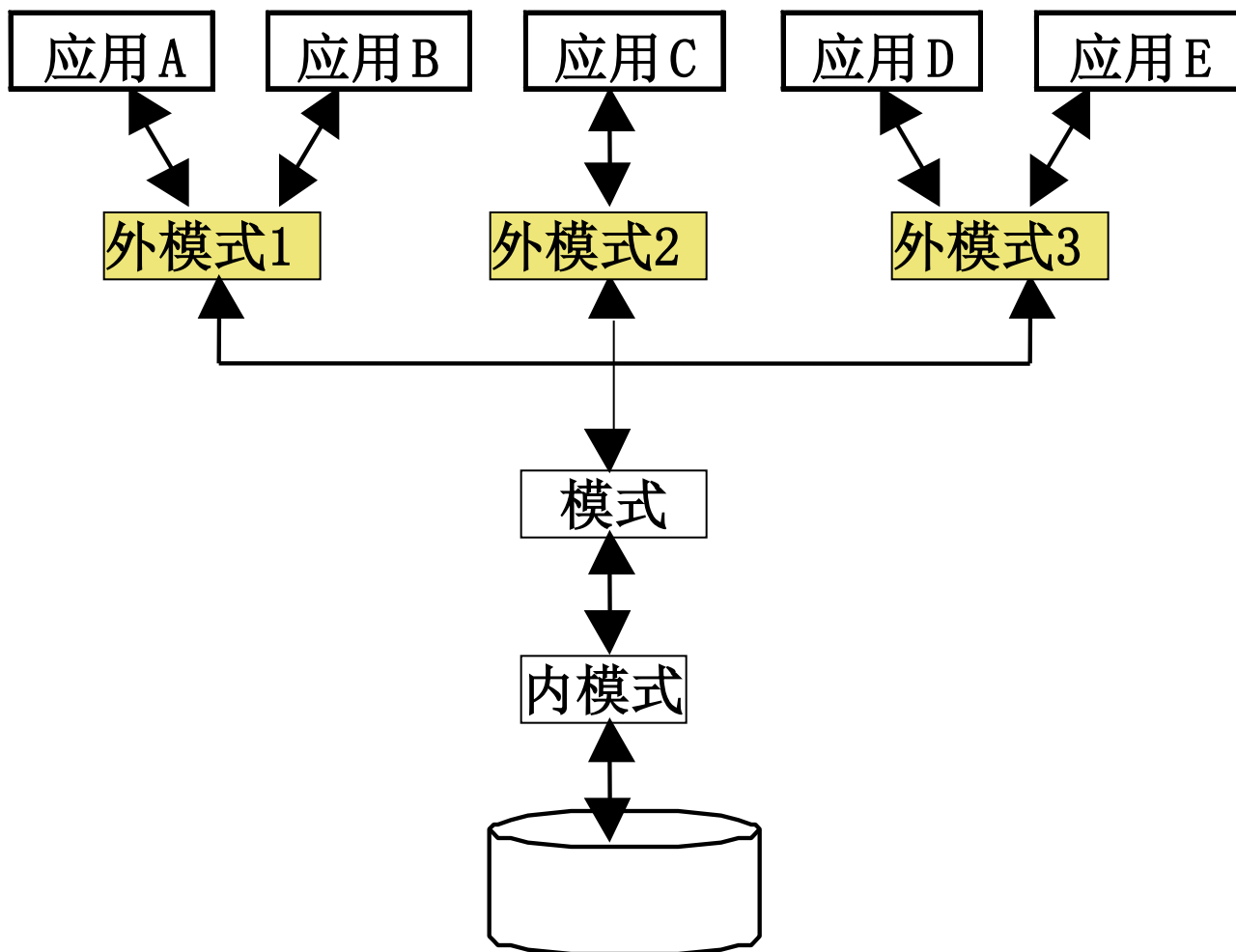
### 一、模式

- 模式（也称逻辑模式）
  - 数据库中全体数据的逻辑结构和特征的描述
  - 所有用户的公共数据视图，综合了所有用户的需求
- 一个数据库只有一个模式
- 模式的地位：是数据库系统模式结构的中间层
  - 与数据的物理存储细节和硬件环境无关
  - 与具体的应用程序、开发工具及高级程序设计语言无关
- 模式的定义
  - 数据的逻辑结构（数据项的名字、类型、取值范围等）
  - 数据之间的联系
  - 数据有关的安全性、完整性要求



# 1.3.2 数据库系统的三级模式结构

## 二、外模式







## 1.3.2 数据库系统的三级模式结构

### 二、外模式

- 外模式（也称子模式或用户模式）
  - 数据库用户（包括应用程序员和最终用户）使用的局部数据的逻辑结构和特征的描述
  - 数据库用户的数据视图，是与某一应用有关的数据的逻辑表示



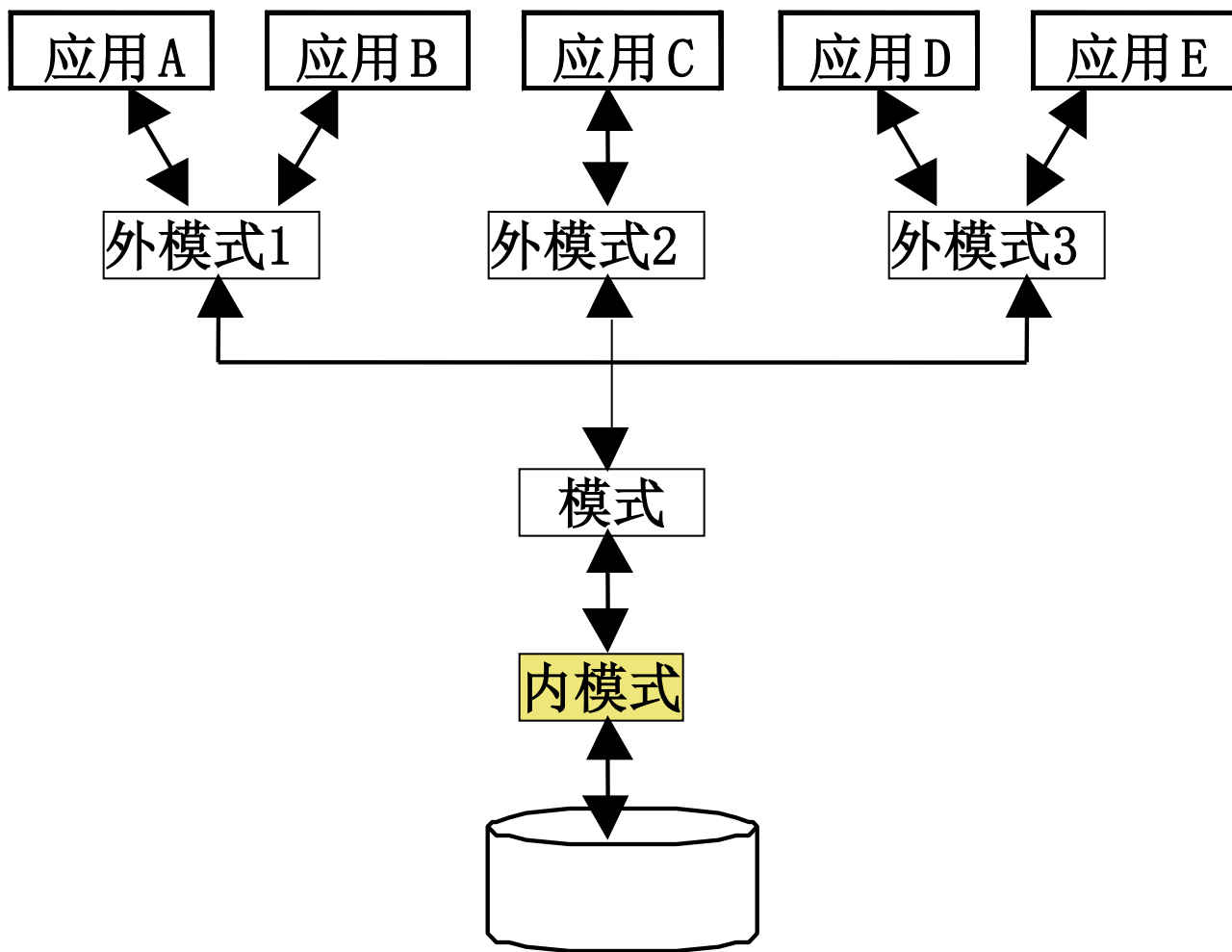
## 1.3.2 数据库系统的三级模式结构

- 外模式的地位：介于模式与应用之间
  - 模式与外模式的关系：一对多
    - 外模式通常是模式的子集
    - 一个数据库可以有多个外模式。反映了不同的用户的应用需求、看待数据的方式、对数据保密的要求
    - 对模式中同一数据，在外模式中的结构、类型、长度、保密级别等都可以不同
  - 外模式与应用的关系：一对多
    - 同一外模式也可以为某一用户的多个应用系统所使用
    - 但一个应用程序只能使用一个外模式。
- 外模式的用途
  - 保证数据库安全性的一个有力措施。
  - 每个用户只能看见和访问所对应的外模式中的数据



# 1.3.2 数据库系统的三级模式结构

## 三、内模式





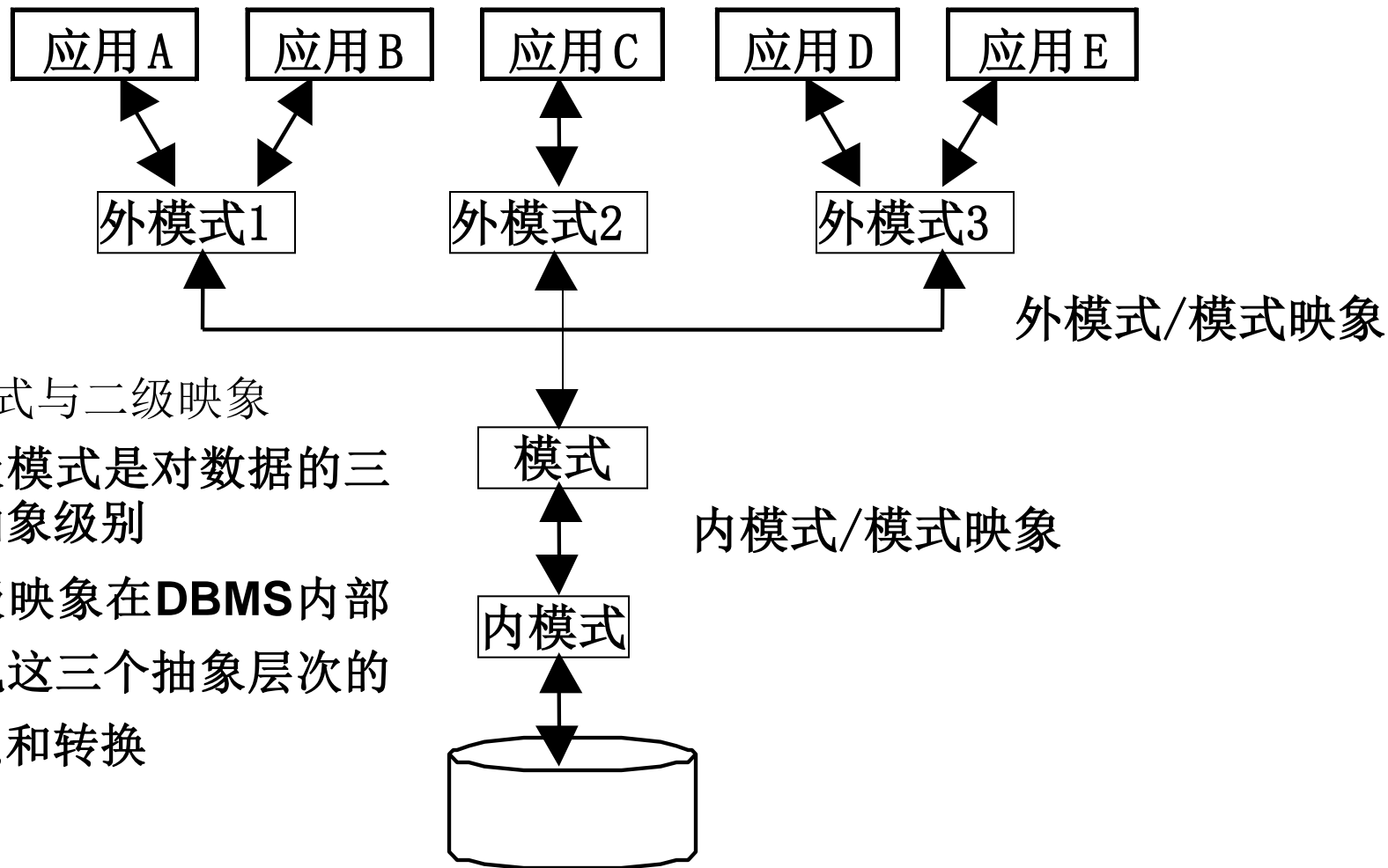
## 1.3.2 数据库系统的三级模式结构

### 三、内模式

- 内模式（也称存储模式）
  - 是数据物理结构和存储方式的描述
  - 是数据在数据库内部的表示方式
    - 记录的存储方式（顺序存储，按照**B**树结构存储，按**hash**方法存储）
    - 索引的组织方式
    - 数据是否压缩存储
    - 数据是否加密
    - 数据存储记录结构的规定
- 一个数据库只有一个内模式



# 1.3.3 数据库的二级映像功能与数据独立性



## 三级模式与二级映象

- 三级模式是对数据的三个抽象级别
- 二级映象在DBMS内部实现这三个抽象层次的联系和转换



## 1.3.3 数据库的二级映像功能与数据独立性

### 一、外模式 / 模式映像

- 定义外模式与模式之间的对应关系
- 每一个外模式都对应一个外模式 / 模式映像
- 映像定义通常包含在各自外模式的描述中



## 1.3.3 数据库的二级映像功能与数据独立性

### 外模式 / 模式映象的用途

#### 保证数据的逻辑独立性

- 当模式改变时，数据库管理员修改有关的外模式 / 模式映象，使外模式保持不变
- 应用程序是依据数据的外模式编写的，从而应用程序不必修改，保证了数据与程序的逻辑独立性，简称数据的逻辑独立性。



## 1.3.3 数据库的二级映像功能与数据独立性

### 二、模式 / 内模式映象

- 模式 / 内模式映象定义了数据全局逻辑结构与存储结构之间的对应关系。例如，说明逻辑记录 and 字段在内部是如何表示的
- 数据库中模式 / 内模式映象是唯一的
- 该映象定义通常包含在模式描述中





## 1.3.3 数据库的二级映像功能与数据独立性

### 模式 / 内模式映象的用途

#### 保证数据的物理独立性

- 当数据库的存储结构改变了（例如选用了另一种存储结构），数据库管理员修改模式 / 内模式映象，使模式保持不变
- 应用程序不受影响。保证了数据与程序的物理独立性，简称数据的物理独立性。



# 1.4 数据库系统的组成

## 一、硬件平台及数据库

- 数据库系统对硬件资源的要求

### (1) 足够大的内存

- 操作系统
- **DBMS**的核心模块
- 数据缓冲区
- 应用程序

### (2) 足够大的外存

- 磁盘
  - 操作系统
  - **DBMS**
  - 应用程序
  - 数据库及其备份
- 光盘、磁带、软盘
  - 数据备份

### (3) 较高的通道能力，提高数据 传送率



# 1.4 数据库系统的组成

## 二、软件

- **DBMS**
- 操作系统
- 与数据库接口的高级语言及其编译系统
- 以**DBMS**为核心的应用开发工具
- 为特定应用环境开发的数据库应用系统



## 1.4 数据库系统的组成

### 三、人员

- 数据库管理员
- 系统分析员
- 数据库设计人员
- 应用程序员
- (最终用户)



# 1.4 数据库系统的组成

## 1. 数据库管理员(DBA)

- 决定数据库中的信息内容和结构
- 决定数据库的存储结构和存取策略
- 定义数据的安全性要求和完整性约束条件
- 监控数据库的使用和运行
  - 周期性转储数据库
    - 数据文件
    - 日志文件
  - 系统故障恢复
  - 介质故障恢复
  - 监视审计文件
- 数据库的改进和重组
  - 性能监控和调优
  - 数据重组
- 数据库重构



# 1.4 数据库系统的组成

## 2. 系统分析员

- 负责应用系统的需求分析和规范说明
- 与用户及**DBA**协商，确定系统的硬软件配置
- 参与数据库系统的概要设计



## 1.4 数据库系统的组成

### 3. 数据库设计人员

- 参加用户需求调查和系统分析
- 确定数据库中的数据
- 设计数据库各级模式



## 1.4 数据库系统的组成

### 4. 应用程序员

- 设计和编写应用系统的程序模块
- 进行调试和安装





# 1.4 数据库系统的组成

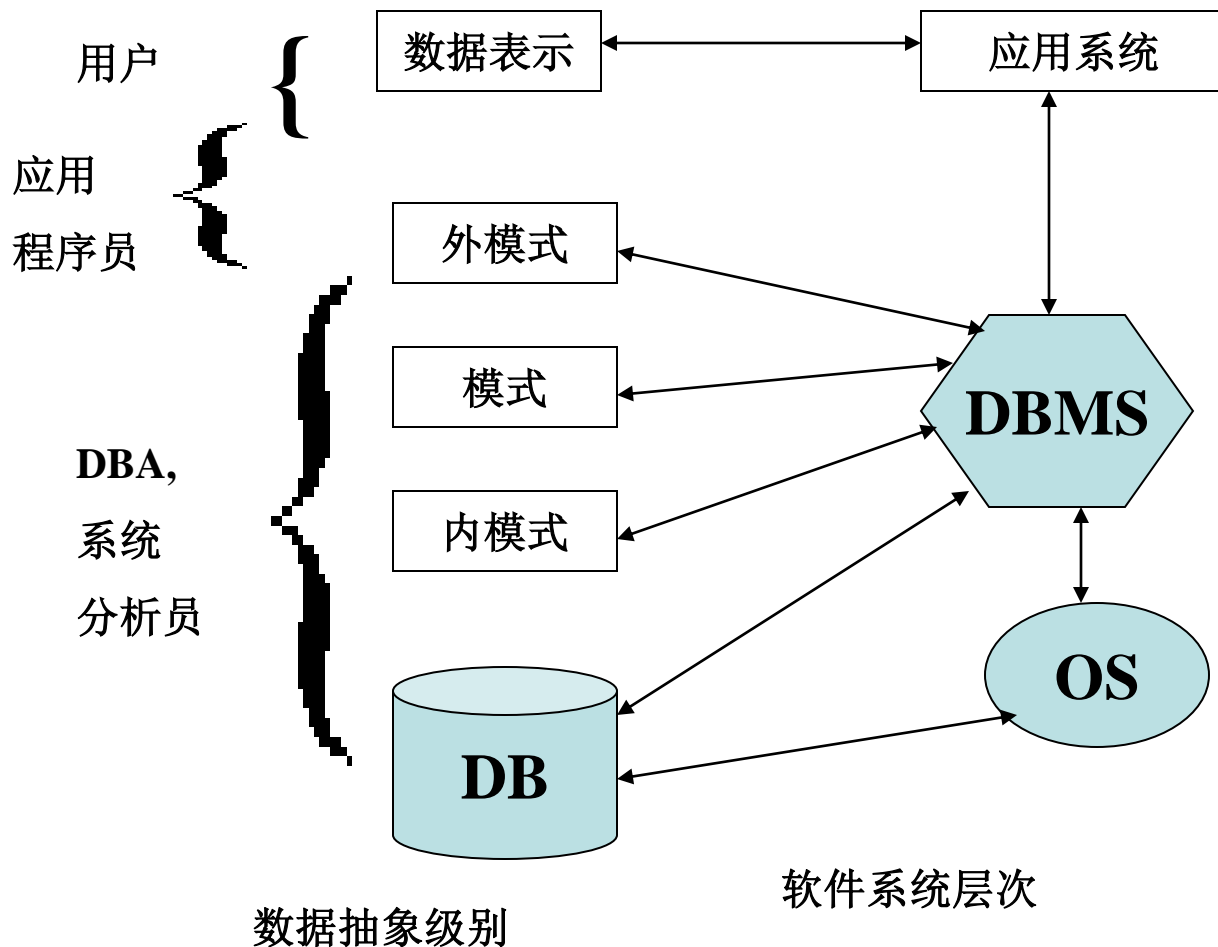
## 5. 用户

- 偶然用户
  - 企业或组织机构的高中级管理人员
- 简单用户
  - 银行的职员、机票预定人员、旅馆总台服务员
- 复杂用户
  - 工程师、科学家、经济学家、科技工作者等
  - 直接使用数据库语言访问数据库，甚至能够基于数据库管理系统的**API**编制自己的应用程序



# 1.4 数据库系统的组成

## 各类人员对应的数据视图





# 附录：主讲教师



## 主讲教师：林子雨

单位：厦门大学计算机科学系

E-mail: ziyulin@xmu.edu.cn

个人网页: <http://www.cs.xmu.edu.cn/linziyu>

数据库实验室网站: <http://dblab.xmu.edu.cn>



扫一扫访问个人主页

林子雨，男，1978年出生，博士（毕业于北京大学），现为厦门大学计算机科学系助理教授（讲师），曾任厦门大学信息科学与技术学院院长助理、晋江市发展和改革局副局长。中国计算机学会数据库专业委员会委员，中国计算机学会信息系统专业委员会委员。国内高校首个“数字教师”提出者和建设者，厦门大学数据库实验室负责人，厦门大学云计算与大数据研究中心主要建设者和骨干成员，2013年度和2017年度厦门大学教学类奖教金获得者，荣获2017年福建省精品在线开放课程和2017年厦门大学高等教育成果二等奖。主要研究方向为数据库、数据仓库、数据挖掘、大数据、云计算和物联网，并以第一作者身份在《软件学报》《计算机学报》和《计算机研究与发展》等国家重点期刊以及国际学术会议上发表多篇学术论文。作为项目负责人主持的科研项目包括1项国家自然科学基金青年基金项目(No.61303004)、1项福建省自然科学基金项目(No.2013J05099)和1项中央高校基本科研业务费项目(No.2011121049)，主持的教改课题包括1项2016年福建省教改课题和1项2016年教育部产学协作育人项目，同时，作为课题负责人完成了国家发改委城市信息化重大课题、国家物联网重大应用示范工程区域试点泉州市工作方案、2015泉州市互联网经济调研等课题。中国高校首个“数字教师”提出者和建设者，2009年至今，“数字教师”大平台累计向网络免费发布超过500万字高价值的研究和教学资料，累计网络访问量超过500万次。打造了中国高校大数据教学知名品牌，编著出版了中国高校第一本系统介绍大数据知识的专业教材《大数据技术原理与应用》，并成为京东、当当网等网店畅销书籍；建设了国内高校首个大数据课程公共服务平台，为教师教学和学生学大数据课程提供全方位、一站式服务，年访问量超过100万次。



# 附录：课程助教



**助教：程璐**

单位：厦门大学计算机科学系数据库实验室2017级硕士研究生  
E-mail: luchengxm@outlook.com



**助教：林哲**

单位：厦门大学计算机科学系数据库实验室2017级硕士研究生  
E-mail: linzhe@stu.xmu.edu.cn



# 附录：班级网站

林子雨主讲《数据库系统原理》2018班级主页

<http://dblab.xmu.edu.cn/post/10147/>



扫一扫访问班级网站  
支持手机浏览

The background of the slide features several faint, light-blue silhouettes of people. At the top, there are two groups of people standing and holding hands. On the right side, a person is shown in profile, looking towards the center. On the left side, two people are shown in profile, one appearing to be speaking or gesturing towards the other. The overall scene suggests a group of people in a meeting or presentation setting.

**Thank You!**

Department of Computer Science, Xiamen University, 2018