

数据库系统原理

Database System Principle

李瑞轩

华中科技大学计算机学院
智能与分布计算实验室

rxli[at]sina[.]com
<http://idc.hust.edu.cn/rxli/>

教材及参考书(1)

教材

- 王珊，杜小勇，陈红：数据库系统概论 (第六版)
高等教育出版社，2023.3
- **A First Course in Database Systems (3rd Ed.)**
Prentice Hall, China Machine Press, 2008
Jeffrey D. Ullman, Jennifer Widom
Dept. Of Computer Science, Stanford University

教材及参考书(2)

参考书

- **C. J. Date, An Introduction to Database System** (Ed.8), Addison-Wesley, 2004
- A. Iberschatz, H. F. Korth, S. Sudarshan. **Database System Concepts** (6th Ed.), McGraw-Hill, 2012

上机软件

- **MySQL 6.0/7.0/8.0**
- MS SQL Server 2000/2005/2010/2016/2017/2020
- Oracle 9i/11g/12c
- 达梦数据库 **DM 6.0/7.0/8.0**
- 华为高斯数据库 **Guass**

中文教材

- ❖ **数据库系统概论**（第六版）王珊 杜小勇 陈红
高等教育出版社
- ❖ **数据库系统基础**（第二版）冯玉才
华中理工大学出版社
- ❖ **数据库系统实现**（第二版）杨冬青 吴愈青 包小源 唐世渭（译）
机械工业出版社

English Textbook(s)

- ❖ **A First Course in Database Systems**
(Jeff Ullman, Jennifer Widom)
- ❖ **Database System Concepts**
(Abraham Iberschatz, Henry F.Korth, S.Sudarshan)
- ❖ **An Introduction to Database Systems**
(C.J.Date)
- ❖ **Database System Implementation**
(Hector G. Molina, Jeff Ullman, Jennifer Widom)
- ❖ **The Concepts of Database Management**
(Philip J.Pratt, Joseph J.Adamski)

教师基本信息

- ❖ Instructor: 李瑞轩
- ❖ Office: N1- 442
- ❖ Phone: 8754 4285
- ❖ E-mail: rxli[at]sina[.]com
- ❖ Web: <http://idc.hust.edu.cn/rxli/>
- ❖ TA: TBA

学习方式

听课 (48学时)

(启发式、讨论式)

读书

(预习、复习)

在线课程

(提前学习、课堂提问)

实验 (32学时)

(综合练习、课程设计)

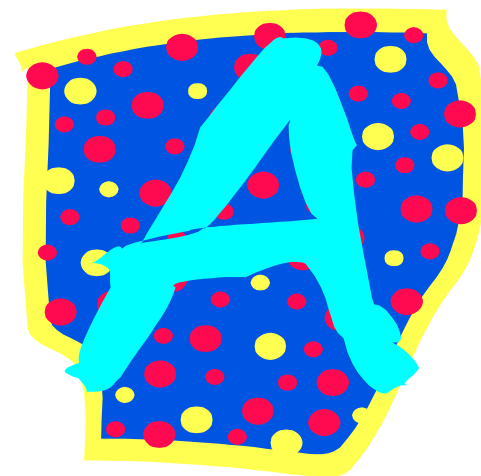


考试成绩

❖ 平时成绩：30%

- 平时作业15%
- 课堂测验15%

❖ 期末考试：70%



课堂辅助

❖ 扫码加入微助教



二维码有效期至: 2025-04-09

刷新

课堂名称: 数据库系统原理 (本硕博2301)

课堂编号: PA059



1、扫码关注公众号: 微助教服务号。

2、点击系统通知: “[点击此处加入【数据库系统原理 \(本硕博2301\)】课堂](#)”，填写学生资料加入课堂。

*如未成功收到系统通知, 请点击公众号下方“学生” - “全部(A)” - “加入课堂” --- “输入课堂编号”手动加入课堂

内容安排(1)

基础篇

- 第1章 绪论 (4)
- 第2章 关系数据库 (4)
- 第3章 关系数据库标准语言SQL (7)
- 第4章 数据库安全性 (3)
- 第5章 数据库完整性 (3)

设计与应用开发篇

- 第6章 关系数据理论 (6)
- 第7章 数据库设计 (4)

内容安排(2)

系统篇

- 第8章 关系数据库引擎基础 (4)
- 第9章 关系查询处理和查询优化 (4)
- 第10章 数据库恢复技术 (4)
- 第11章 并发控制 (5)
- * 第12章 数据库管理系统 (0)

第1至第11章是本科专业的基本教程（书中有*号的部分除外）

第12至第16章是高级教程



数据库系统原理

Database System Principle

第一章 绪论

第一章 绪论

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

1.5 小结

数据库的地位

- ❖ 数据库技术产生于六十年代末，是数据管理的新技术，是计算机科学的重要分支。
- ❖ 数据库技术是信息系统的核心和基础，它的出现极大地促进了计算机应用向各行各业的渗透。
- ❖ 数据库的建设规模、数据库信息量的大小和使用频度已成为衡量一个国家信息化程度的重要标志。

第一章 绪论

1.1 数据库系统概述

1.1.1 四个基本概念

1.1.2 数据管理技术的产生和发展

1.1.3 数据库系统的特点

1.1.1 四个基本概念

- ❖ 数据(Data)
- ❖ 数据库(Database)
- ❖ 数据库管理系统(DBMS)
- ❖ 数据库系统(DBS)

一、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

- 文本、图形、图像、音频、视频、学生的档案记录、货物的运输情况等

❖ 数据的特点

- 数据与其语义是不可分的

数据举例

❖ 数据的含义称为**数据的语义**，数据与其语义是不可分的。

- 例如 100是一个数据

语义1：学生某门课的成绩

语义2：某人的体重

语义3：计算机学院2015级学生人数

语义4：请同学们给出其它解释

数据举例

❖ 学校人事档案中的职工记录

(0005796, 201, 尤政, 1, 1963.11.10, 01)

校长办公室

男

汉族

- 语义：工号, 部门编号, 姓名, 性别, 出生日期, 民族
- 解释：校长尤政, 男, 1963年11月10日出生, 汉族, 在校长办公室办公

二、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、可共享的大量数据的集合。

❖ 数据库的基本特征

- 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

三、数据库管理系统

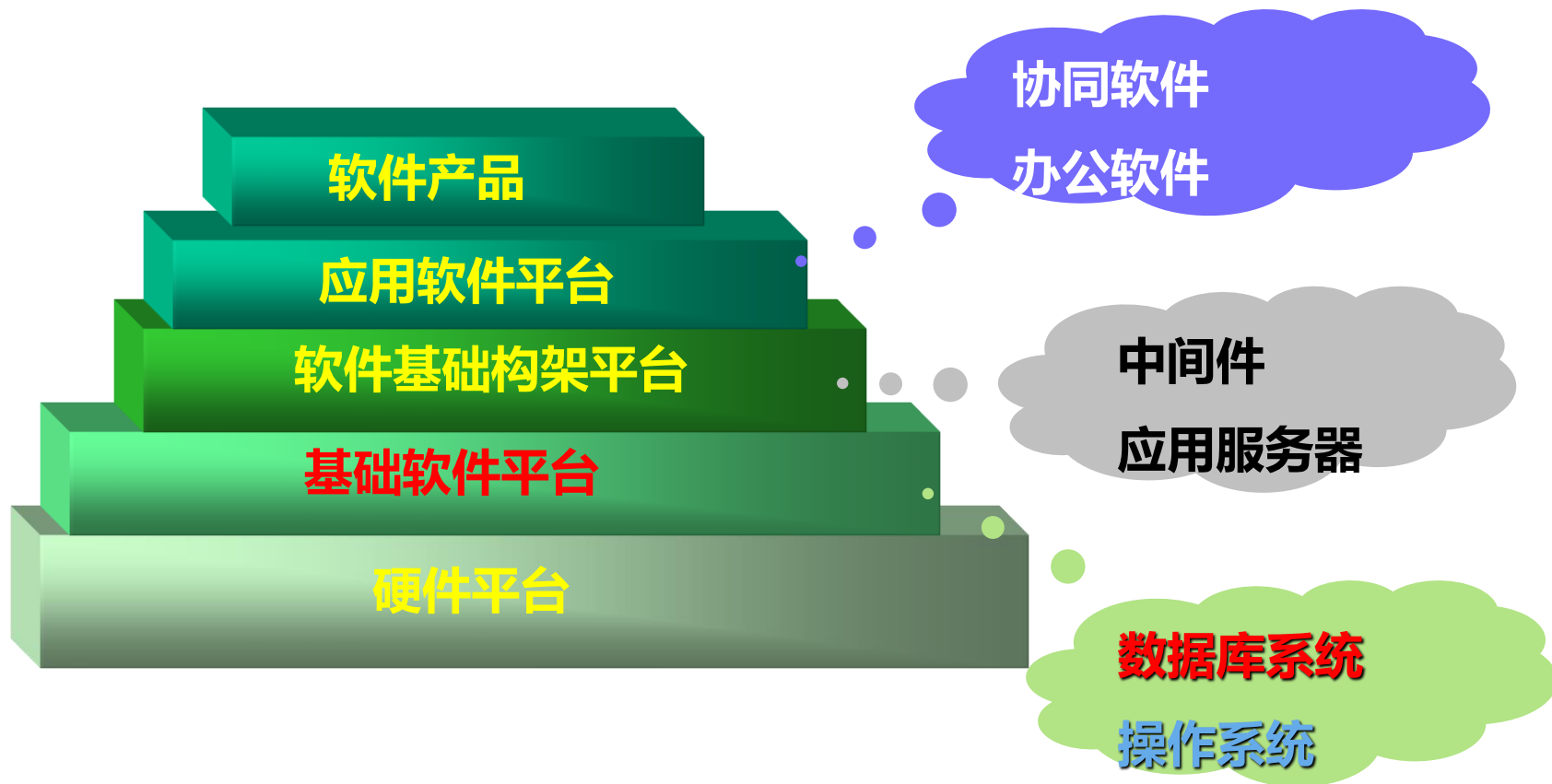
❖ 什么是数据库管理系统（DBMS）

- DBMS: Database Management System
- 位于用户与操作系统之间的一层数据管理软件
- 是基础软件，是一个大型复杂的软件系统

❖ DBMS的用途

- 科学地组织和存储数据、高效地获取和维护数据

数据库在计算机系统中的位置



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

- 数据组织、存储和管理

- 分类组织、存储和管理各种数据
- 确定组织数据的文件结构和存取方式
- 实现数据之间的联系
- 提供多种存取方法提高存取效率

DBMS的主要功能

- 数据操纵功能
 - 提供数据操纵语言(DML)
 - 实现对数据库的基本操作 (查询、插入、删除和修改)
- 数据库的事务管理和运行管理
 - 数据库在建立、运行和维护时由DBMS统一管理和控制
 - 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

DBMS的主要功能

- **数据库的建立和维护功能(实用程序)**
 - 数据库初始数据装载转换
 - 数据库转储
 - 介质故障恢复
 - 数据库的重组织
 - 性能监视分析等
- **其它功能**
 - DBMS与网络中其它软件系统的通信
 - 两个DBMS系统的数据转换
 - 异构数据库之间的互访和互操作

四、数据库系统

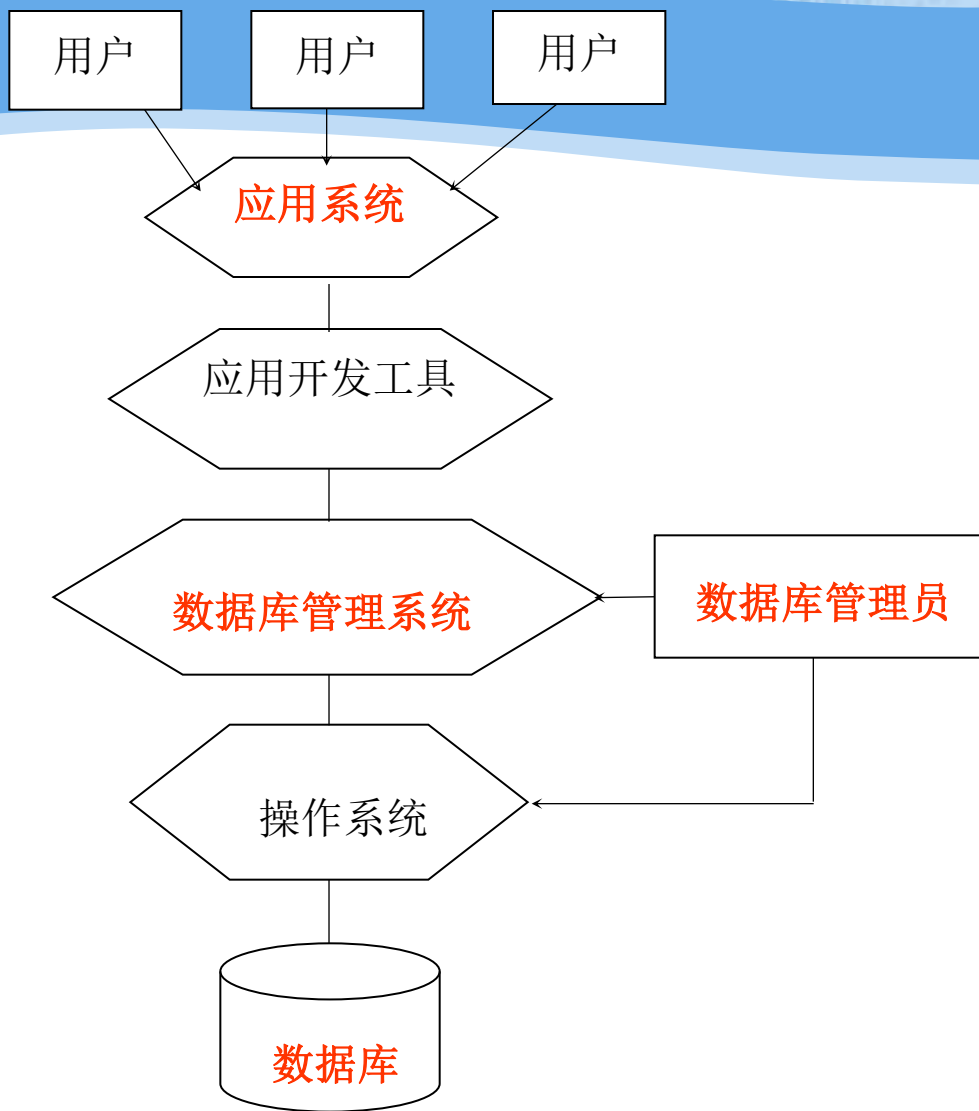
❖ 什么是**数据库系统**（Database System，简称DBS）

在计算机系统中引入数据库后的系统

（数据库系统有时简称为数据库）

❖ 数据库系统的构成

- 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



数据库系统

1.1 数据库系统概述

1.1.1 四个基本概念

1.1.2 数据管理技术的产生和发展

1.1.3 数据库系统的特点

数据管理技术的产生和发展

❖ 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
- 数据处理的中心问题

❖ 数据管理技术的发展过程

- 人工管理阶段(20世纪40年代中—50年代中)
- 文件系统阶段(20世纪50年代末—60年代中)
- 数据库系统阶段(20世纪60年代末—现在)
- 大数据系统阶段(21世纪10年代—现在)

数据管理技术的产生和发展(续)

❖ 数据管理技术的发展动力

- 应用需求的推动
- 计算机硬件的发展
- 计算机软件的发展

数据管理技术的三个发展阶段

手工管理

手工管理（直接书写机器语言）

```
00101011 11100101 11001001  
10101100 11010100 11110000  
01010010 10010000 10000000
```

文件管理

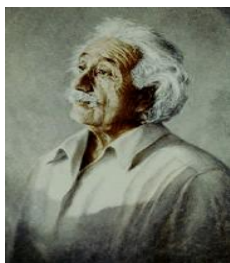
文件管理（程序管理数据）

```
main () {  
    int a, b, c;  
    fopen(... .. );  
}
```

数据库管理

数据库管理（自主管理信息）

```
select * from S  
insert  
delete
```



OS



DBMS



一、人工管理阶段

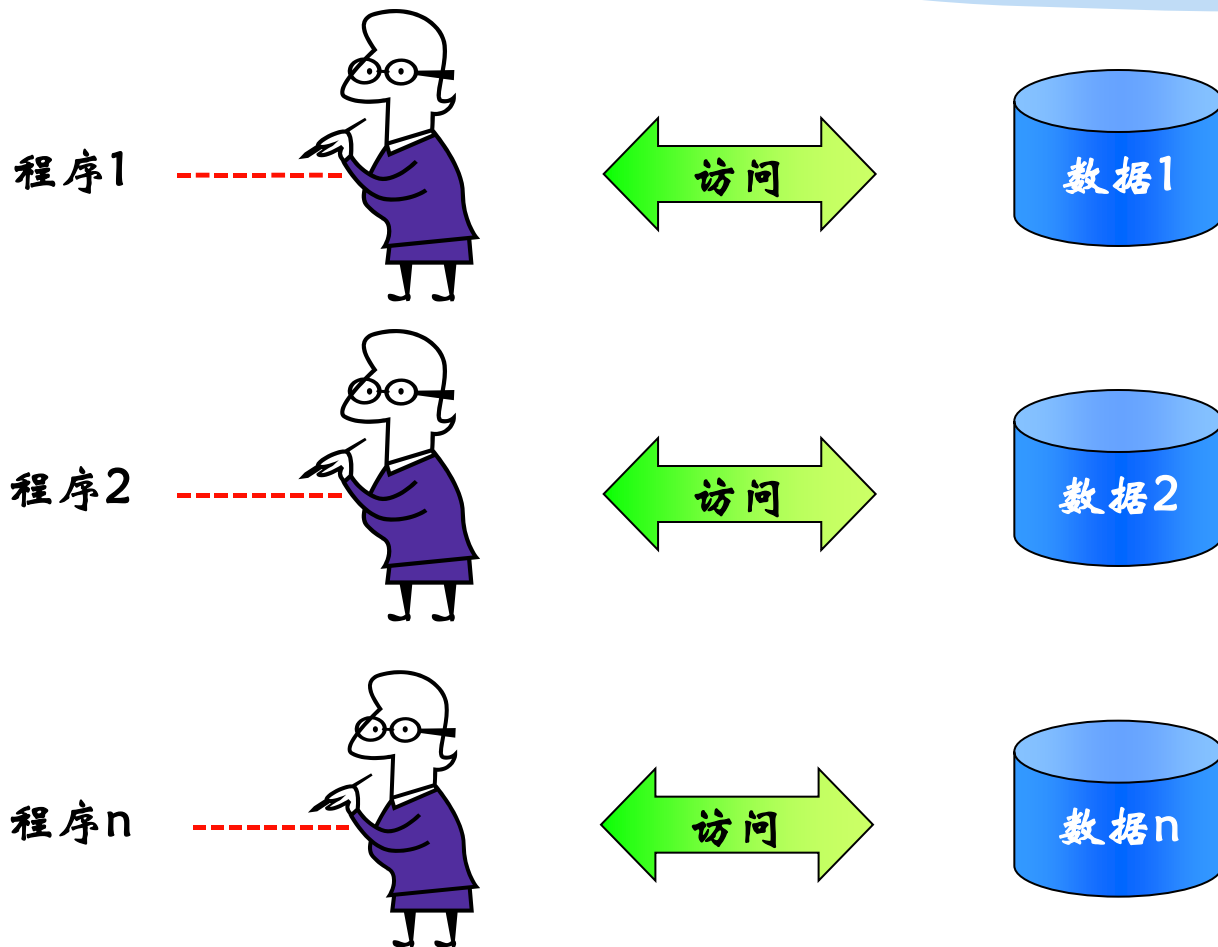
❖ 时期

- 20世纪40年代中——50年代中

❖ 产生的背景

- 应用需求 科学计算（数据量小、结构简单，如高阶方程、曲线拟和等）
- 硬件水平 磁带、卡片、纸带，没有磁盘等直接存取设备
- 软件水平 没有操作系统，没有数据管理软件
- 处理方式 批处理

应用程序与数据的对应关系(人工管理阶段)



人工管理阶段的特点

❖ 特点

- 数据的管理者：用户（程序员），数据不保存
- 数据面向的对象：某一应用程序
- 数据的共享程度：无共享、冗余度极大
- 数据的独立性：不独立，完全依赖于程序
- 数据的结构化：无结构
- 数据控制能力：应用程序自己控制

二、文件系统阶段

❖ 时期

- 20世纪50年代末——60年代中

❖ 产生的背景

- 应用需求 科学计算、数据管理
- 硬件水平 磁盘、磁鼓（直接存取设备）
- 软件水平 有文件系统
- 处理方式 联机实时处理、批处理

文件系统阶段的处理方式

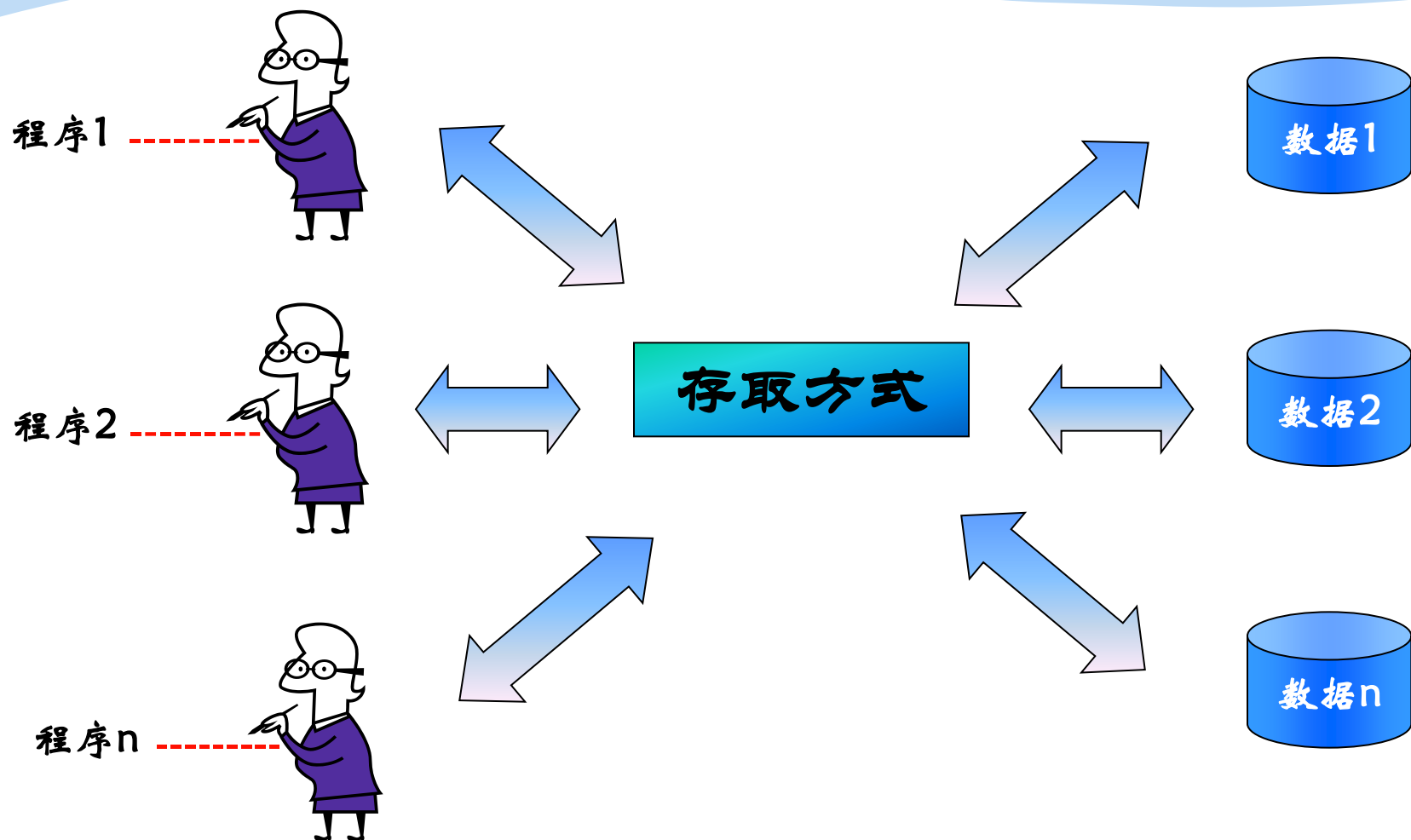
❖ 直接存取设备(磁盘、磁鼓等)的特点

- 无须顺序存取
- 由地址直接访问所需记录

❖ 文件系统的特点

- 文件存储空间的管理
- 目录管理
- 文件读写管理
- 文件保护
- 向用户提供操作接口

应用程序与数据的对应关系(文件系统阶段)



文件管理系统实例



劳资科

学号	姓名	系别	补贴
----	----	----	----



房产科

学号	姓名	性别	系别	住址
----	----	----	----	----



学籍科

学号	姓名	系别	学分	学位
----	----	----	----	----



人事科

学号	姓名	性别	系别	年龄	学位	出身
----	----	----	----	----	----	----

文件系统阶段的缺点

❖ 数据的共享性差，冗余度大

- 数据面向应用
 - 即使不同应用程序所需要的数据有部分相同时，也必须建立各自的文件，而不能共享相同的数据
- 数据孤立
 - 数据分散管理，许多文件，许多数据格式
- 数据的不一致性
 - 由于数据存在很多副本，给数据的修改与维护带来了困难，容易造成数据的不一致性

文件系统阶段的缺点

❖ 数据与程序的独立性差

- 文件系统的出现并没有从根本上改变数据与程序紧密结合的状况，数据的逻辑结构改变则必须修改应用程序
- 文件系统只是解脱了程序员对物理设备存取的负担，它并不理解数据的语义，只负责存储
- 数据的语义信息只能由程序来解释，也就是说，数据收集以后怎么组织，以及数据取出来之后按什么含义应用，只有全权管理它的程序知道
- 一个应用若想共享另一个应用生成的数据，必须同另一个应用沟通，了解数据的语义与组织方式

三、数据库系统阶段

❖ 时期

- 20世纪60年代末以来

❖ 产生的背景

- 应用背景 大规模数据管理
- 硬件背景 大容量磁盘、磁盘阵列
- 软件背景 有数据库管理系统
- 处理方式 联机实时处理，分布处理，批处理

1.1 数据库系统概述

1.1.1 四个基本概念

1.1.2 数据管理技术的产生和发展

1.1.3 数据库系统的特点

1.1.3 数据库系统的特点

- ❖ 数据结构化
- ❖ 数据的共享性高，冗余度低，易扩充
- ❖ 数据独立性高
- ❖ 数据由**DBMS**统一管理和控制

数据结构化

- ❖ 整体数据的结构化是数据库的主要特征之一
- ❖ 整体结构化
 - 不再仅仅针对某一个应用，而是面向全组织
 - 不仅数据内部结构化，而且整体是结构化的，数据之间具有联系
- ❖ 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

数据的共享性高，冗余度低，易扩充

- ❖ 数据库系统从**整体角度**看待和描述数据，数据面向**整个系统**，可以被多个用户、多个应用共享使用
- ❖ 数据共享的好处
 - 减少数据冗余，节约存储空间
 - 避免数据之间的不相容性与不一致性
 - 使系统易于扩充

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。

❖ 数据独立性是由DBMS的二级映像功能来保证的

数据由DBMS统一管理和控制

❖ DBMS提供的数据库控制功能

- (1)数据的**安全性**（Security）保护

保护数据，以防止不合法的使用造成的数据的泄密和破坏。

- (2)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。

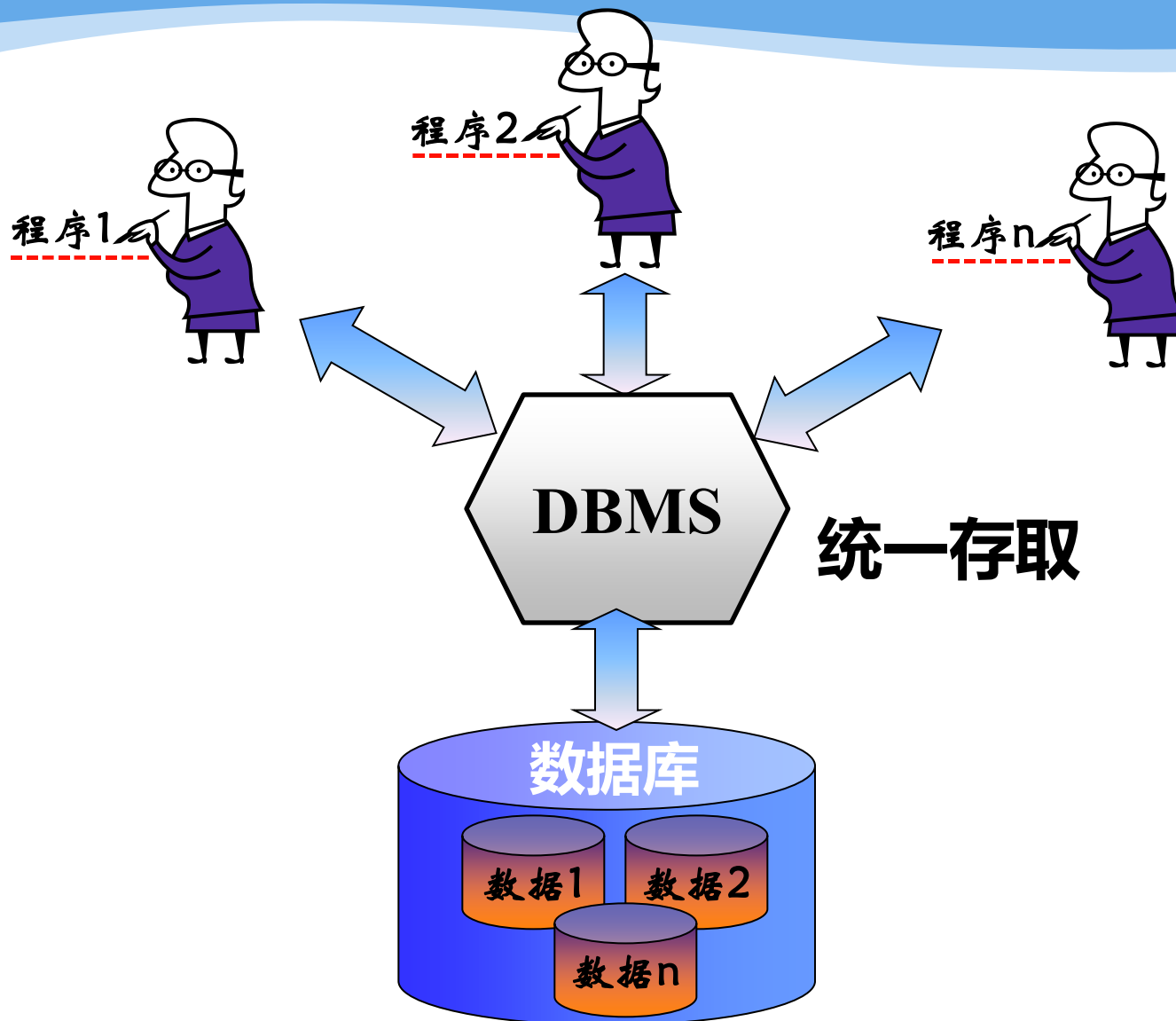
- (3)**并发**（Concurrency）控制

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。

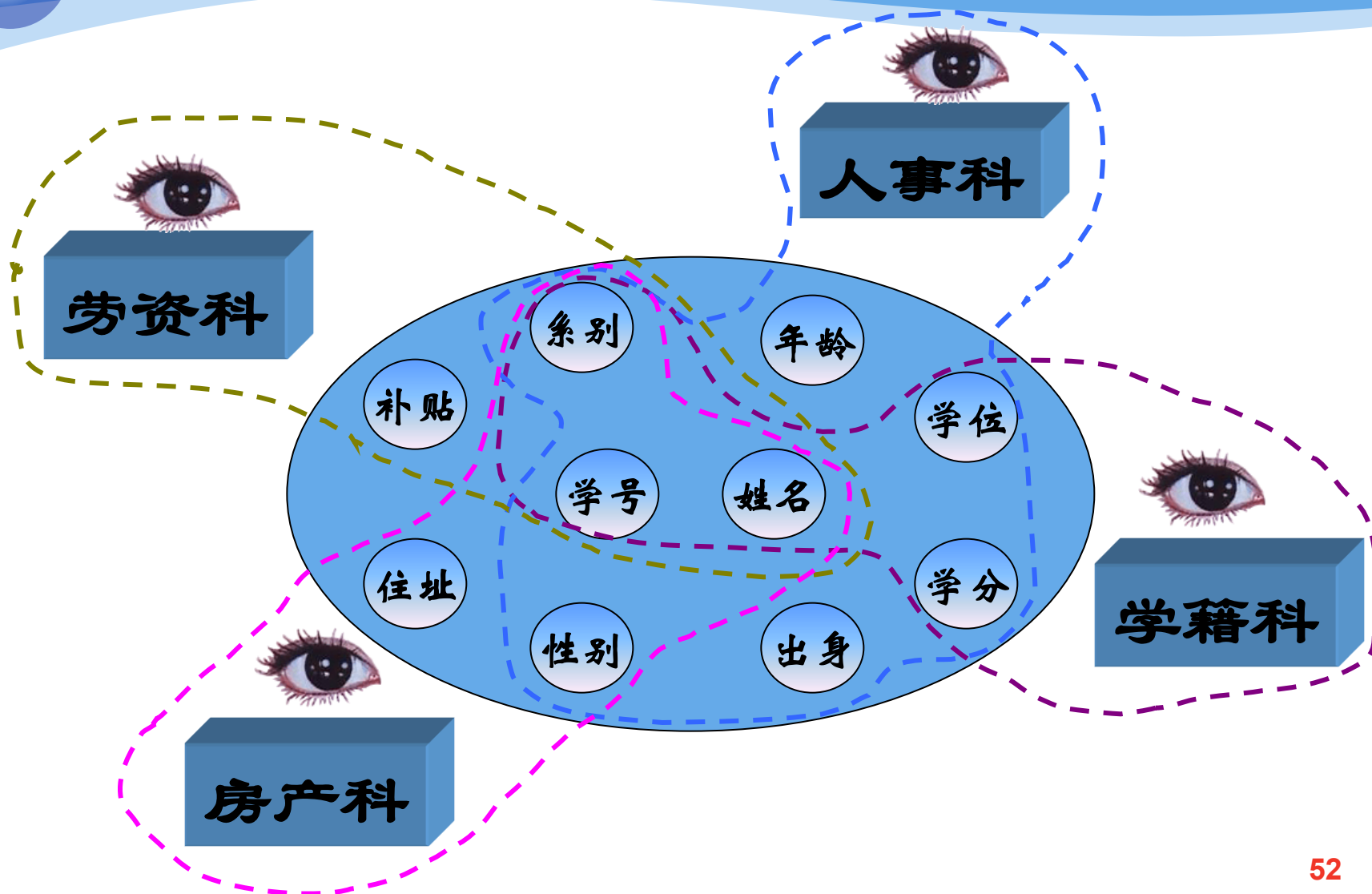
- (4)数据库**恢复**（Recovery）

将数据库从错误状态恢复到某一已知的正确状态。

应用程序与数据的对应关系(数据库系统)



数据库系统管理实例



数据库系统的功能特性

十大功能特性：

- **数据独立性**：物理独立性和逻辑独立性。
- **数据共享性**：数据可被多个用户共享使用。
- **可控冗余性**：同一数据在数据库中只存放一次，所有用户可以共享使用。
- **数据安全性**：身份鉴定、子模式隔离、存取控制。
- **数据完整性**：用完整性约束条件保证。

数据库系统的功能特性（续）

- **数据一致性**：用以避免存有二义性的数据进入数据库。
- **并发控制**：保证完整性、一致性的手段和技术。
- **集中存放、集中控制**：方便管理（对于分布式数据库：物理分布，逻辑集中）
- **故障可恢复性**：日志、事务、副本。
- **数据字典**：系统字典和用户字典。

第一章 绪论

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

1.5 小结

1.2 数据模型

1.2.1 两大类数据模型

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

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。
- ❖ 通俗地讲数据模型就是现实世界的模拟。
- ❖ 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 两大类数据模型

❖ 数据模型分为两类（分属两个不同的层次）

(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，用于数据库设计。

(2) **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、面向对象模型等，按计算机系统的观点对数据建模，用于**DBMS**实现。
- 物理模型是对数据最底层的抽象，描述数据在系统内部的表示方式和存取方法，在磁盘或磁带上的存储方式和存取方法。

三个世界的划分

■ 现实世界

■ 存在于人们头脑之外的客观世界

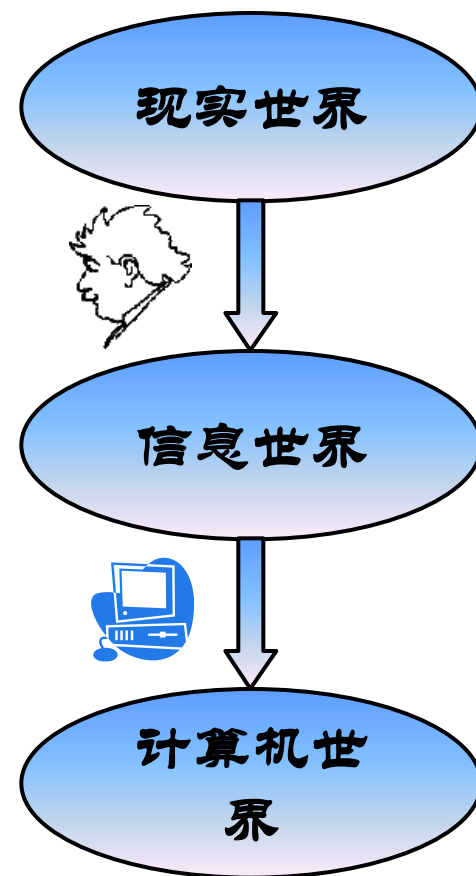
- 例如：仓库管理中涉及的货物管理，货物、货物的进出以及相应的报表、图表、表格、卡片、单据等

■ 信息世界

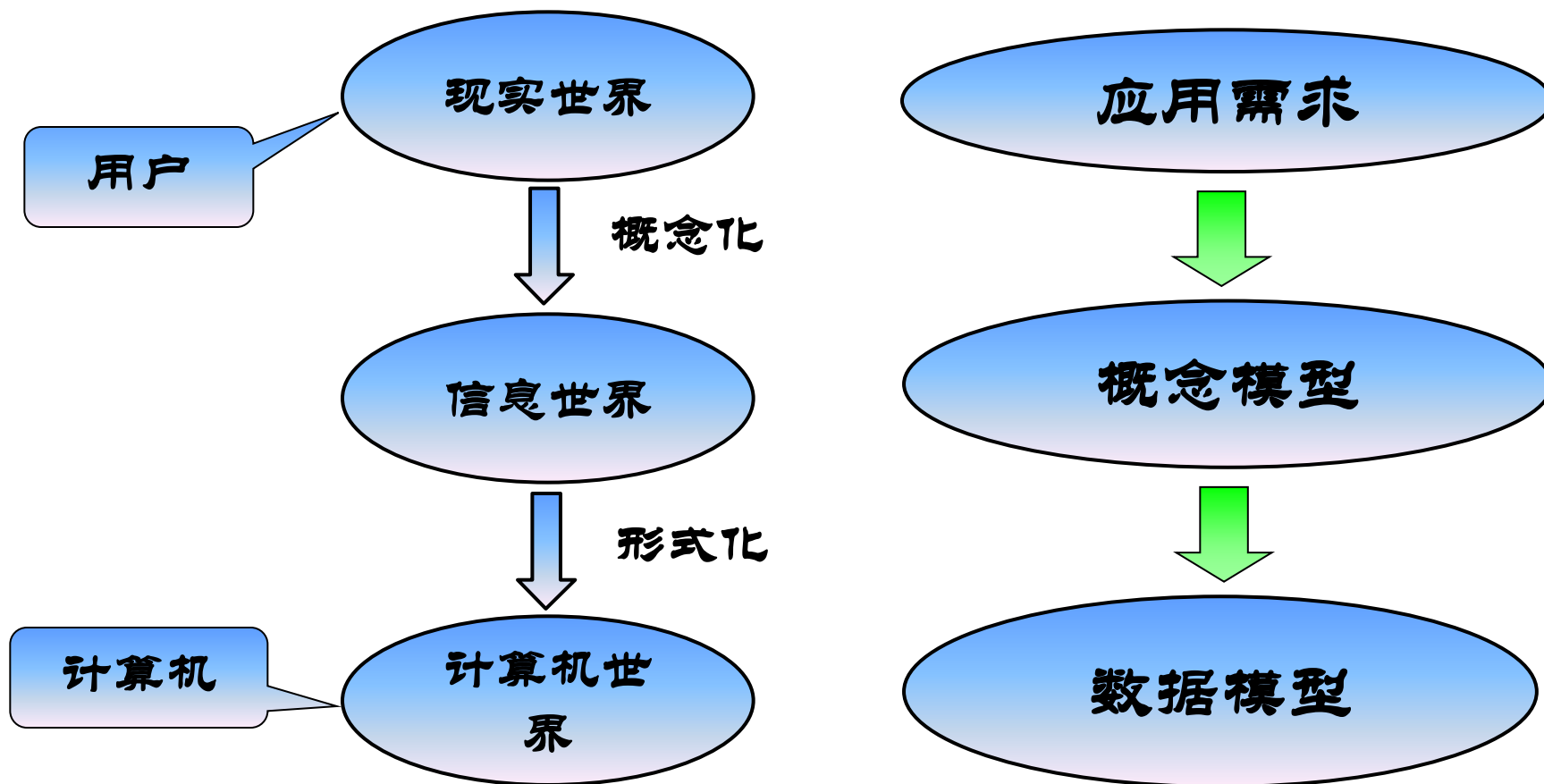
■ 现实世界在人脑中的反映

■ 计算机世界

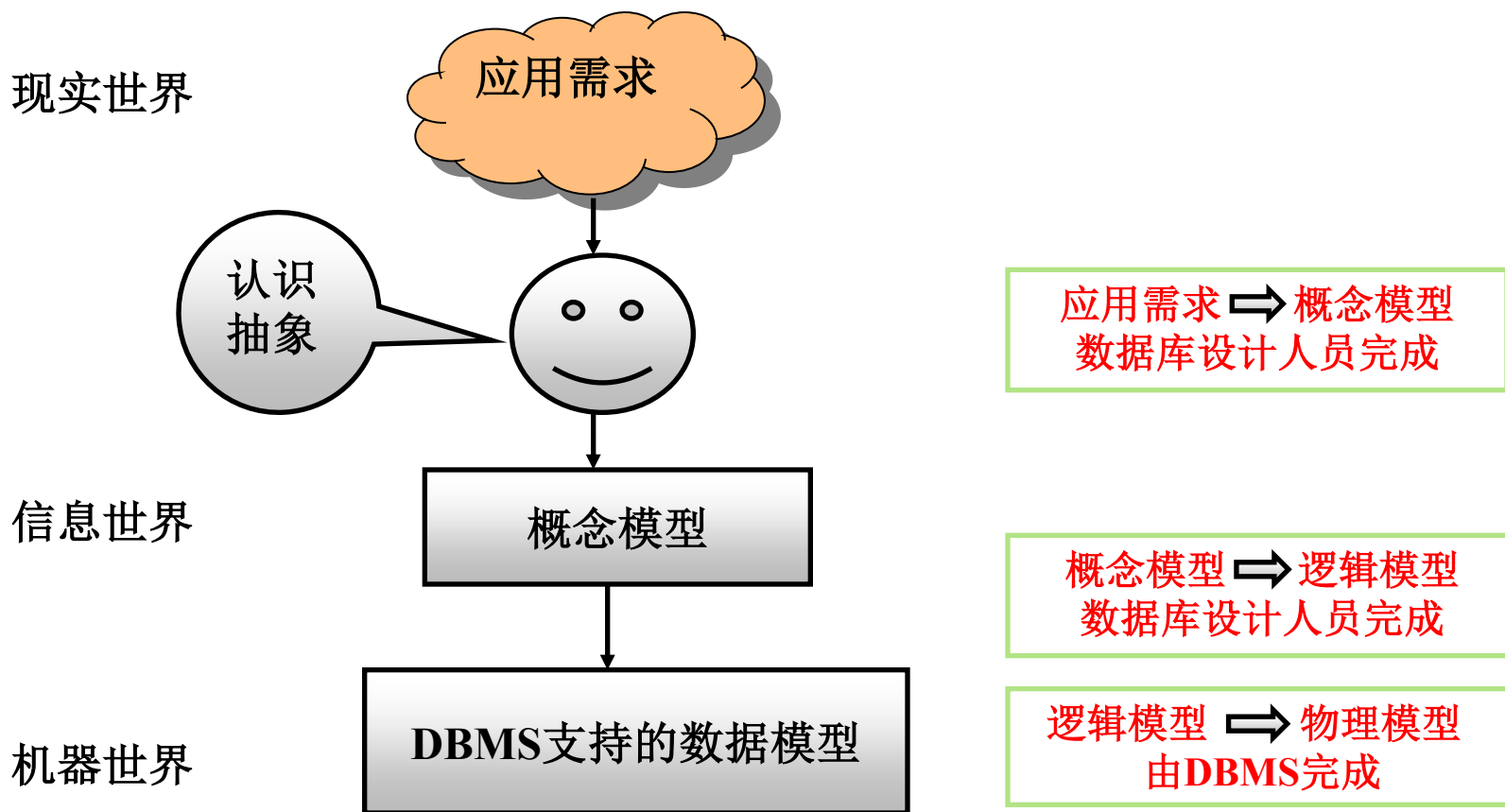
■ 信息在计算机中以数据形式存储



三个世界与两类模型



现实世界中客观对象的抽象过程



1.2 数据模型

1.2.1 两大类数据模型

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

1.2.2 数据模型的组成要素

❖ 数据结构

❖ 数据操作

❖ 完整性约束条件

一、数据结构

❖ 什么是数据结构

- 描述数据库的组成对象，以及对象之间的联系

❖ 描述的内容

- 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

二、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）

数据操作(续)

❖ 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

三、数据的完整性约束条件

❖ 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**：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**。

数据的完整性约束条件(续)

❖ 数据模型对完整性约束条件的定义

- 反映和规定本数据模型必须遵守的基本的通用的完整性约束条件。
 - 例如在关系模型中，任何关系必须满足实体完整性和参照完整性两个条件。
- 提供定义完整性约束条件的机制，以反映具体应用所涉及的数据必须遵守的特定的语义约束条件。

1.2 数据模型

1.2.1 两大类数据模型

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

1.2.3 概念模型

- ❖ 信息世界中的基本概念
- ❖ 概念模型的一种表示方法
- ❖ 一个实例

概念模型

❖ 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

一、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、物或抽象的概念。

(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。

信息世界中的基本概念(续)

(4) 域 (Domain)

属性的取值范围称为该属性的域。

(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

(6) 实体集 (Entity Set)

同一类型实体的集合称为实体集

信息世界中的基本概念(续)

(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内部的联系和实体之间的联系。
- 实体内部的联系通常是指组成实体的各属性之间的联系
- 实体之间的联系通常是指不同实体集之间的联系

二、概念模型的一种表示方法

❖ 实体—联系方法(E-R方法)

- Peter Chen, 1976: **Entity-Relationship Model**
- 用E-R图来描述现实世界的概念模型
- 三种称谓: E-R Model, E-R Diagram, E-R Approach

❖ E-R模型的三要素

- 实体型、属性和联系

E-R图

❖ 实体型

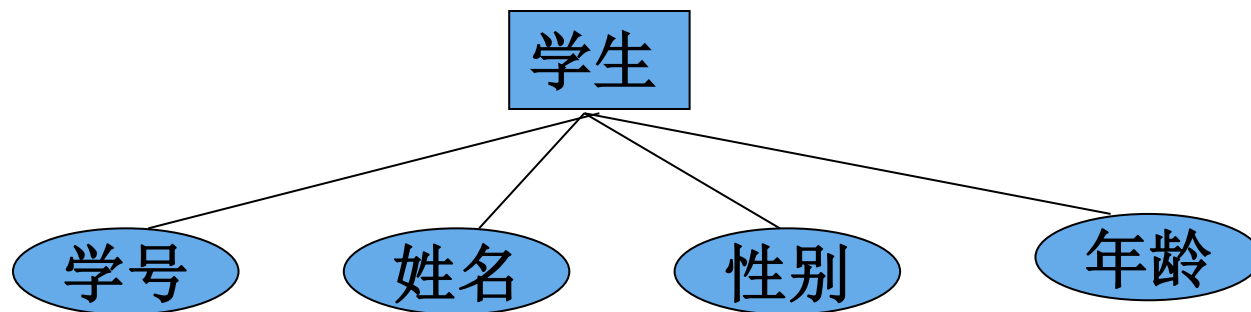
用矩形表示，矩形框内写明实体名。

学生

教师

❖ 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



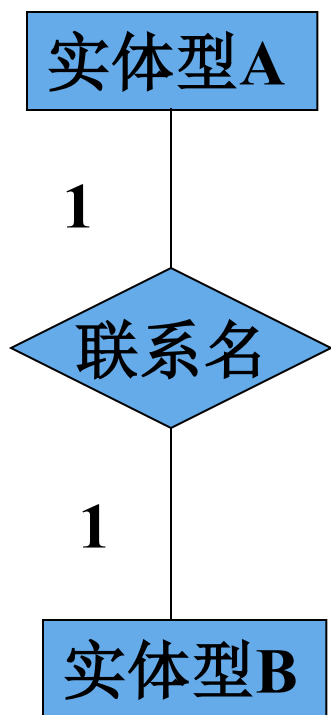
E-R图(续)

❖ 联系

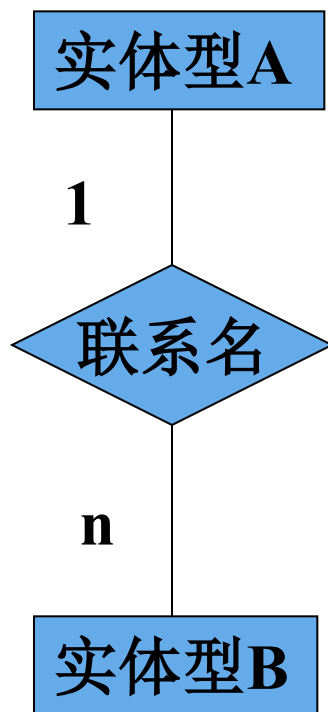
■ 联系本身：

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，在无向边旁标上联系类型（1:1、1:n或m:n）

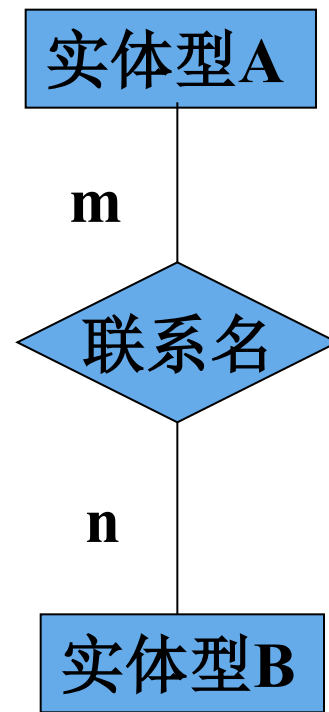
联系的表示方法



1:1联系

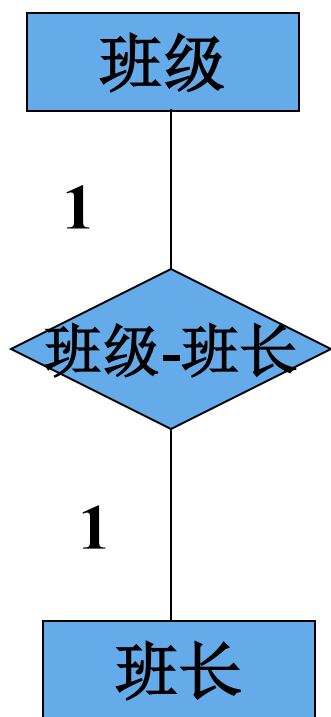


1:n联系

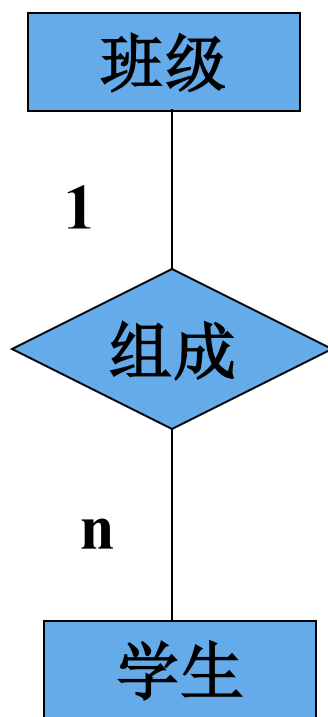


m:n联系

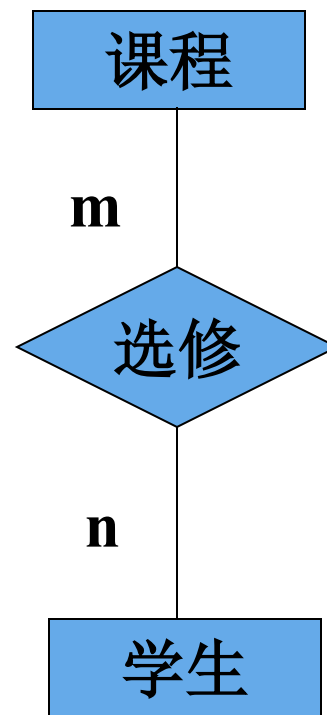
联系的表示方法示例



1:1联系



1:n联系



m:n联系

两个以上实体型之间的多对多联系

- ❖ 多个实体型间的一对一联系
- ❖ 两个以上实体型间的多对多联系

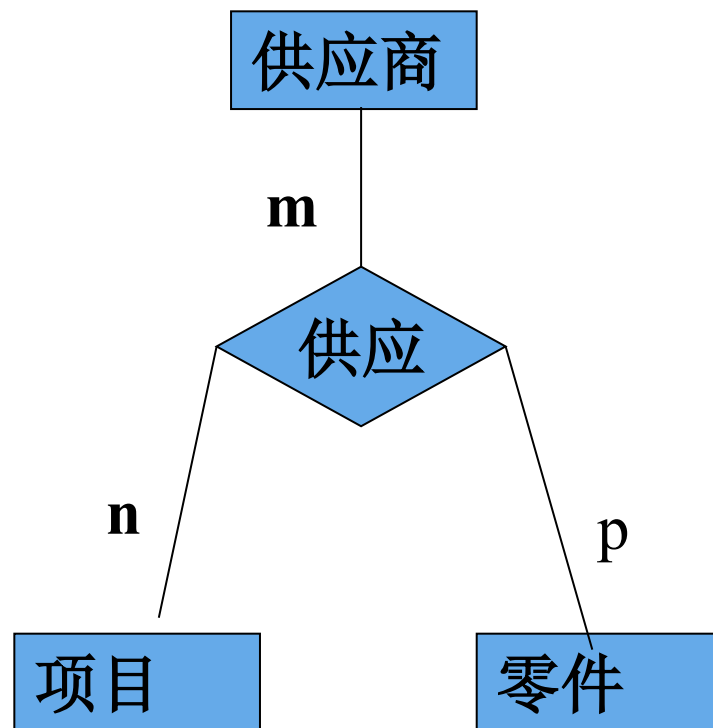
- 实例

供应商、项目、零件三个实体型

一个供应商可以供给多个项目多种零件

每个项目可以使用多个供应商供应的零件

每种零件可由不同供应商供给

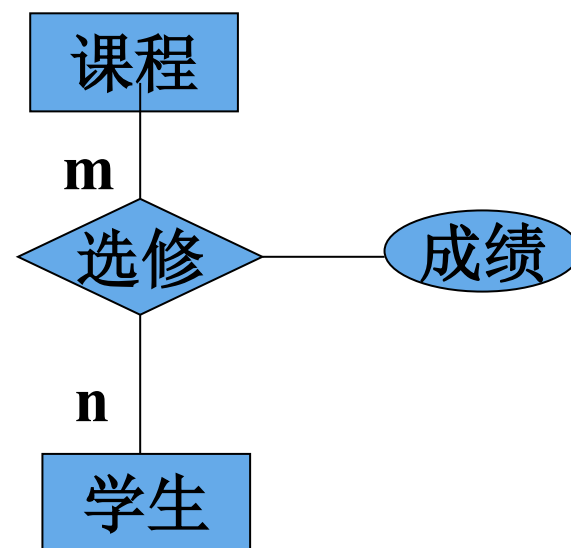


两个以上实体型间m:n联系

联系的属性

❖ 联系的属性:

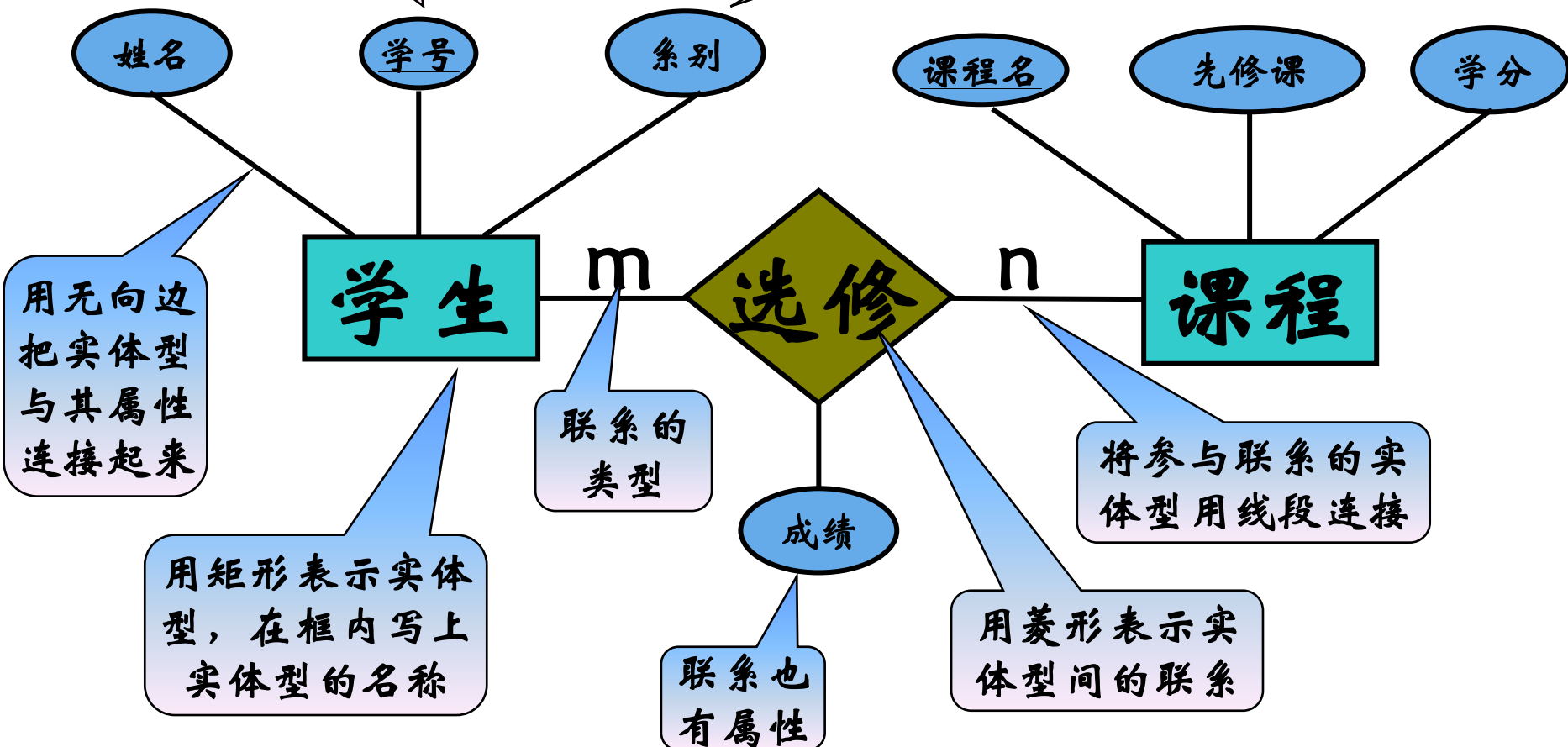
联系本身也是一种实体型，也可以有属性。如果一个联系具有属性，则这些属性也要用无向边与该联系连接起来



例：学生选修课程

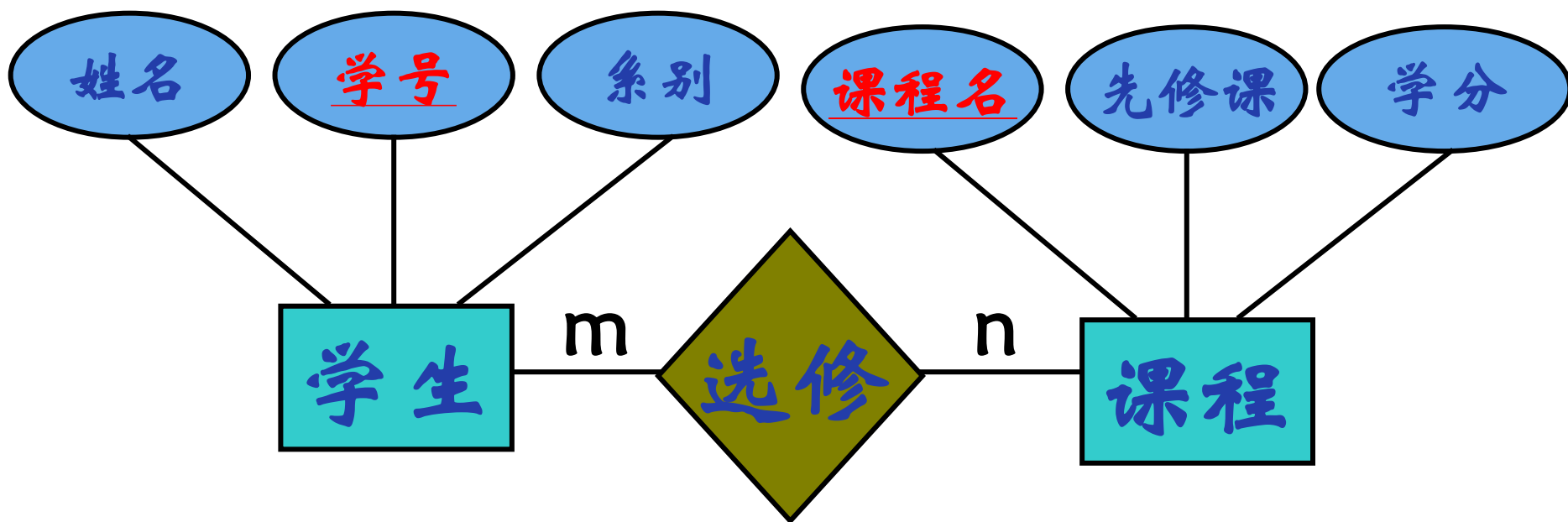
用下划线表示实体的主码

用椭圆表示实体的属性



码在E-R图中的表示

- 用下划线来标明实体型的主码属性，主码可能是单个属性，也可能是多个属性构成的属性集。



三、一个实例（作业）

用E-R图表示某个工厂物资管理的概念模型

❖ 实体

- 仓库： 仓库号、面积、电话号码
- 零件： 零件号、名称、规格、单价、描述
- 供应商： 供应商号、姓名、地址、电话号码、帐号
- 项目： 项目号、预算、开工日期
- 职工： 职工号、姓名、年龄、职称

一个实例（作业）

❖ 实体之间的联系如下：

- 一个仓库可以存放多种零件，一种零件可以存放在多个仓库中。仓库和零件具有多对多的联系。用库存量来表示某种零件在某个仓库中的数量。
- 一个仓库有多个职工当仓库保管员，一个职工只能在一个仓库工作，仓库和职工之间是一对多的联系。职工实体型中具有一对多的联系。
- 职工之间具有领导-被领导关系。即仓库主任领导若干保管员。
- 供应商、项目和零件三者之间具有多对多的联系。

1.2 数据模型

1.2.1 两大类数据模型

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

1.2.4 最常用的数据模型

❖ 非关系模型

- 层次模型(Hierarchical Model)
- 网状模型(Network Model)

❖ 关系模型(Relational Model)

❖ 面向对象模型(Object Oriented Model)

❖ 对象关系模型(Object Relational Model)

1.2.5 层次模型

- ❖ 层次模型是数据库系统中最早出现的数据模型
- ❖ 层次数据库系统的典型代表是IBM公司1968年推出的IMS
(Information Management System) 数据库管理系统
- ❖ 层次模型用树形结构来表示各类实体以及实体间的联系

1.2.6 网状模型

❖ 网状数据库系统采用网状模型作为数据的组织方式

❖ 典型代表是**DBTG**系统：

- 亦称CODASYL系统
- 70年代由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

❖ 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

- ❖ 关系数据库系统采用关系模型作为数据的组织方式
- ❖ 1970年美国IBM公司San Jose研究室的研究员E.F.Codd首次提出了数据库系统的关系模型
- ❖ 计算机厂商新推出的数据库管理系统几乎都支持关系模型

一、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

属性					
学 号	姓 名	年 龄	性 别	系 名	年 级
2005004	王小明	19	女	社会学	2005
2005006	黄大鹏	20	男	商品学	2005
2005008	张文斌	18	女	法律	2005
...

元组

关系数据模型的数据结构（续）

- **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

关系数据模型的数据结构（续）

- **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（Component）**

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）

关系数据模型的数据结构（续）

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，系号，年级）

系（系号，系名，办公地点）

例2

系、系主任、系与系主任间的一对一联系

系（系号，系名，系主任）

关系数据模型的数据结构（续）

例3

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）

关系数据模型的数据结构（续）

❖ 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，
不允许表中还有表

图1.27中工资和扣除是可分的数据项，**不符合关系模型要求**

职工号	姓名	职 称	工 资			扣 除		实 发
			基 本	津 贴	职 务	房 租	水 电	
86051	陈 平	讲 师	1305	1200	50	160	112	2283
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

图1.27 一个工资表(表中有表)实例

关系数据模型的数据结构（续）

表1.2 术语对比

关系术语	一般表格的术语
关系名	表名
关系模式	表头（表格的描述）
关系	（一张）二维表
元组	记录或行
属性	列
属性名	列名
属性值	列值
分量	一条记录中的一个列值
非规范关系	表中有表（大表中嵌有小表）

二、关系数据模型的操纵与完整性约束

- ❖ 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
 - 查询
 - 插入
 - 删除
 - 更新
- ❖ 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**”

关系数据模型的操纵与完整性约束（续）

❖ 关系的完整性约束条件

- 实体完整性
- 参照完整性
- 用户定义的完整性

三、关系数据模型的存储结构

- ❖ 实体及实体间的联系都用**表**来表示
- ❖ 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

四、关系数据模型的优缺点

❖ 优点

- 建立在严格的数学概念的基础上
- 概念单一
 - 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

关系数据模型的优缺点（续）

❖ 缺点

- 存取路径对用户透明导致查询效率往往不如非关系数据模型
- 为提高性能，必须对用户的查询请求进行优化，增加了开发DBMS的难度

第一章 绪论

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

1.5 小结

1.3 数据库系统结构

- ❖ 从数据库管理系统角度看，数据库系统通常采用三级模式结构，是数据库系统内部的系统结构
- ❖ 从数据库最终用户角度看（数据库系统外部的体系结构），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 / 数据库服务器多层结构等

数据库系统结构（续）

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

1.3.1 数据库系统模式的概念

❖ 回忆：面向对象的程序设计

```
class person{  
    public:  
        string name;  
        string address;  
};
```

```
person Tom;
```

person是型(类)，Tom是变量，在某时刻的值是实例

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

学生记录型:

(学号, 姓名, 性别, 系别, 年龄, 籍贯)

一个记录值:

(900201, 李明, 男, 计算机, 22, 江苏)

数据库系统模式的概念（续）

❖ 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

❖ 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

数据库系统模式的概念（续）

例如：在学生选课数据库模式中，包含学生记录、课程记录和学生选课记录

- 2020年的一个学生数据库实例，包含：
 - 2020年学校中所有学生的记录
 - 学校开设的所有课程的记录
 - 所有学生选课的记录
- 2020年度学生数据库模式对应的实例与2021年度学生数据库模式对应的实例是**不同**的

数据库系统结构（续）

1.3.1 数据库系统模式的概念

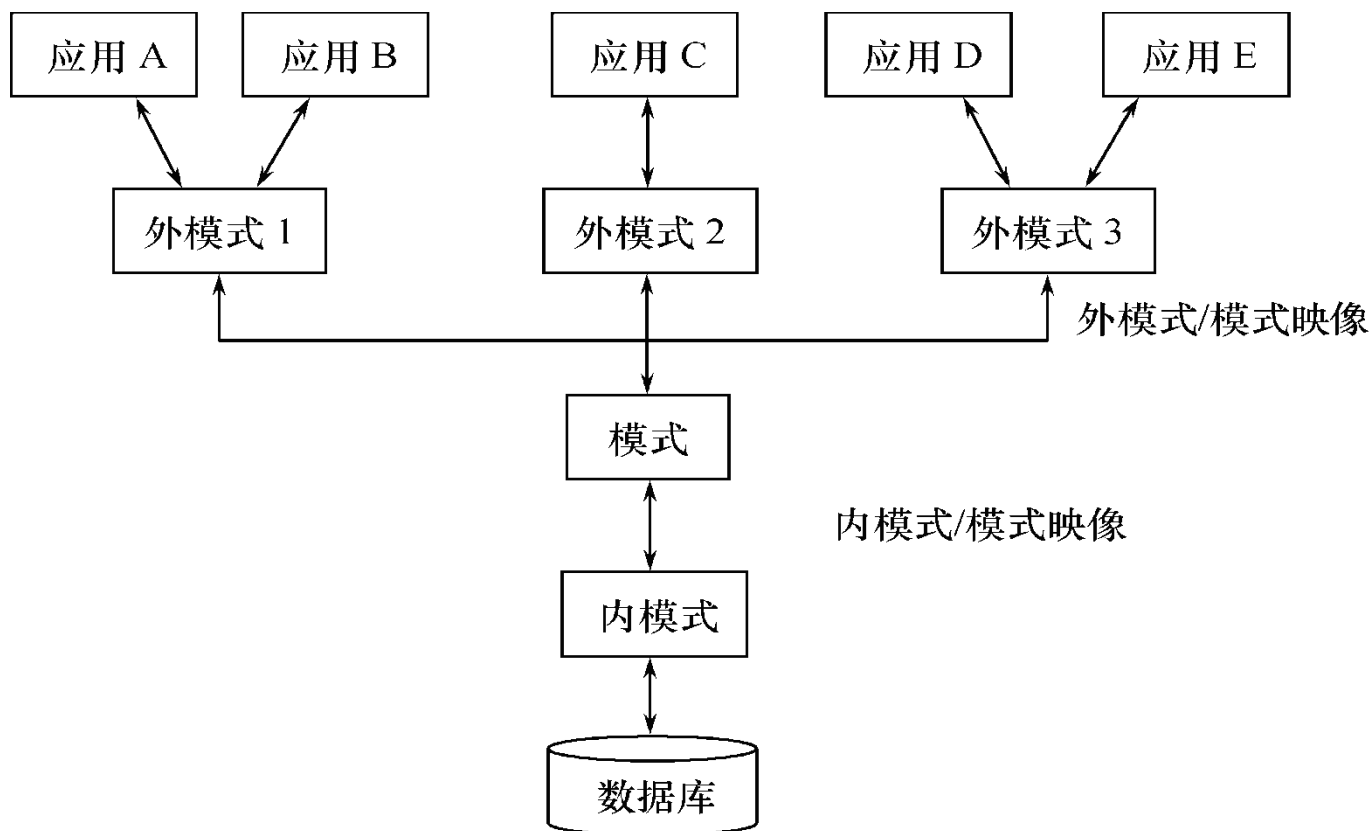
1.3.2 数据库系统的三级模式结构

1.3.3 数据库的二级映像功能与数据独立性

1.3.2 数据库系统的三级模式结构

- ❖ 模式（Schema）
- ❖ 外模式（External Schema）
- ❖ 内模式（Internal Schema）

数据库系统的三级模式结构（续）



一、模式（Schema）

❖ 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

二、外模式（External Schema）

❖ 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

外模式（续）

❖ 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（续）

❖ 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

三、内模式（Internal Schema）

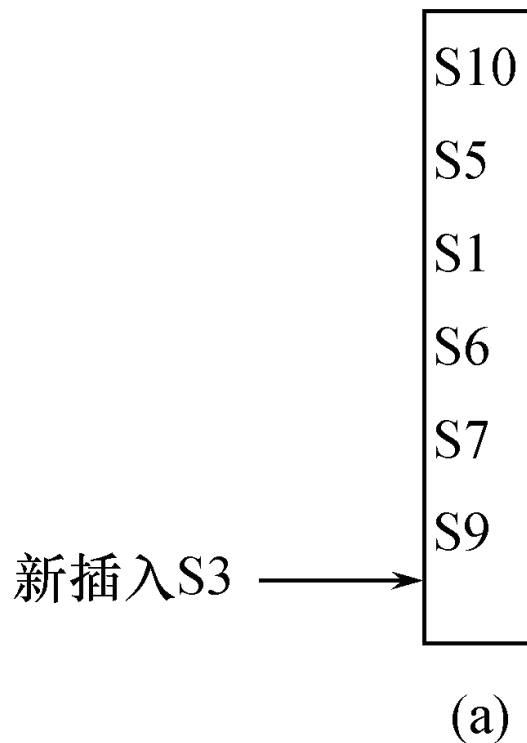
❖ 内模式（也称存储模式）

- 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
（按顺序存储，或B树结构存储，或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

内模式（续）

- ❖ 例如学生记录，如果按堆存储，则插入一条新记录总是放在学生记录存储的最后，如右图所示



内模式（续）

- ❖ 如果按学号升序存储，则插入一条记录就要找到它应在的位置插入，如图1.29（b）所示
- ❖ 如果按照学生年龄聚簇存放，假如新插入的S3是16岁，则应插入的位置如图1.29（c）所示

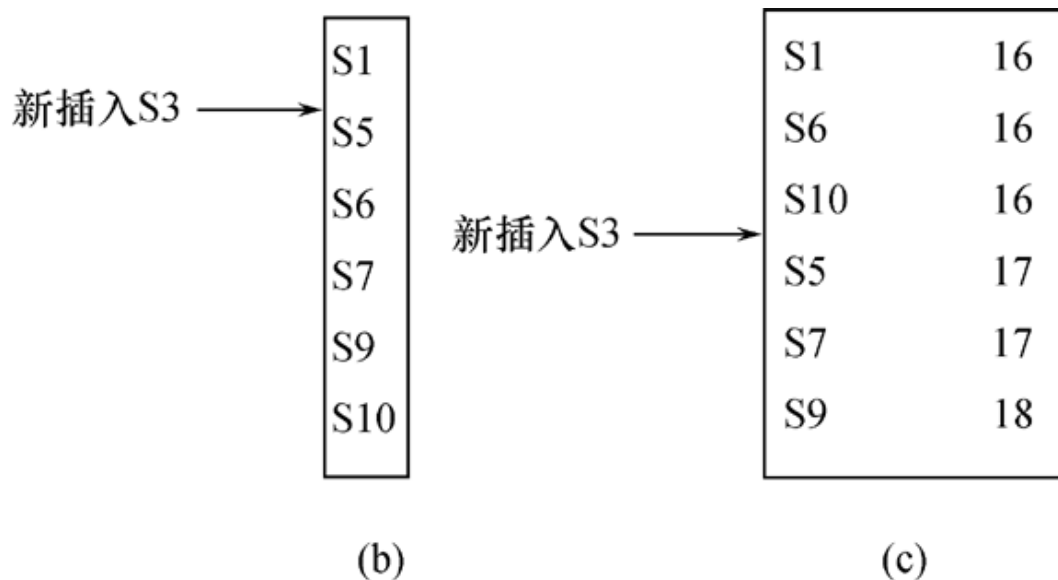


图1.29 记录不同的存储方式示意图

数据库系统结构（续）

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

1.3.3 数据库的二级映像功能与数据独立性

- ❖ 三级模式是对数据的三个抽象级别
- ❖ 二级映像 在DBMS内部实现这三个抽象层次的联系和转换
 - 外模式 / 模式映像
 - 模式 / 内模式映像

一、外模式 / 模式映象

定义外模式与模式之间的对应关系，映象定义通常包含在各自**外模式**的描述中。

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**。

二、模式 / 内模式映象

定义模式与内模式之间的对应关系，该映象是**唯一**的，定义通常包含在**模式**描述中。

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，从而使应用程序不受影响，保证了数据与程序的物理独立性，简称**数据的物理独立性**。

第一章 绪论

1.1 数据库系统概述

1.2 数据模型

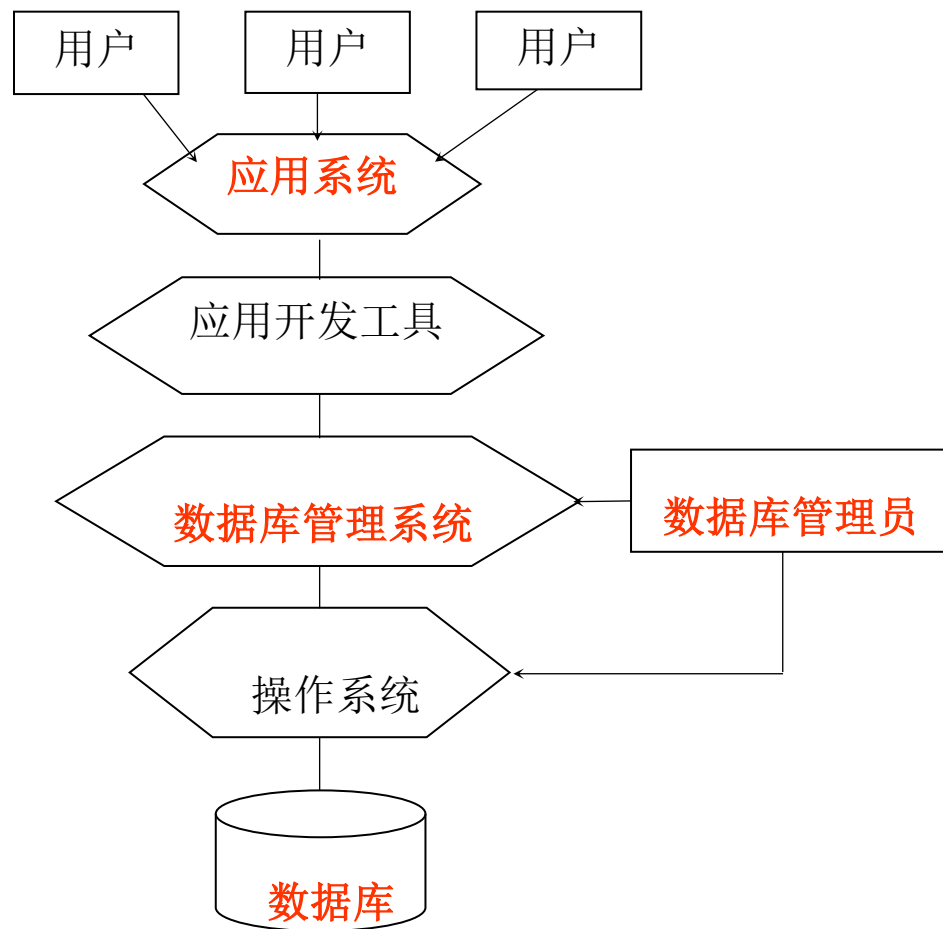
1.3 数据库系统结构

1.4 数据库系统的组成

1.5 小结

1.4 数据库系统的组成

- ❖ 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



数据库系统的组成（续）

另一种分类法：

- ❖ 硬件平台及数据库
- ❖ 软件
- ❖ 人员

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

硬件平台及数据库（续）

(2) 足够大的外存

- 磁盘或磁盘阵列
 - 数据库
- 光盘、磁带
 - 数据备份

(3) 较高的通道能力，提高数据传送率

二、软件

❖ DBMS

- ❖ 支持DBMS运行的操作系统
- ❖ 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

三、人员

- ❖ 数据库管理员
- ❖ 系统分析员和数据库设计人员
- ❖ 应用程序员
- ❖ 用户

人员（续）

❖ 不同的人员涉及不同的数据抽象级别，具有不同的数据视图，如下图所示

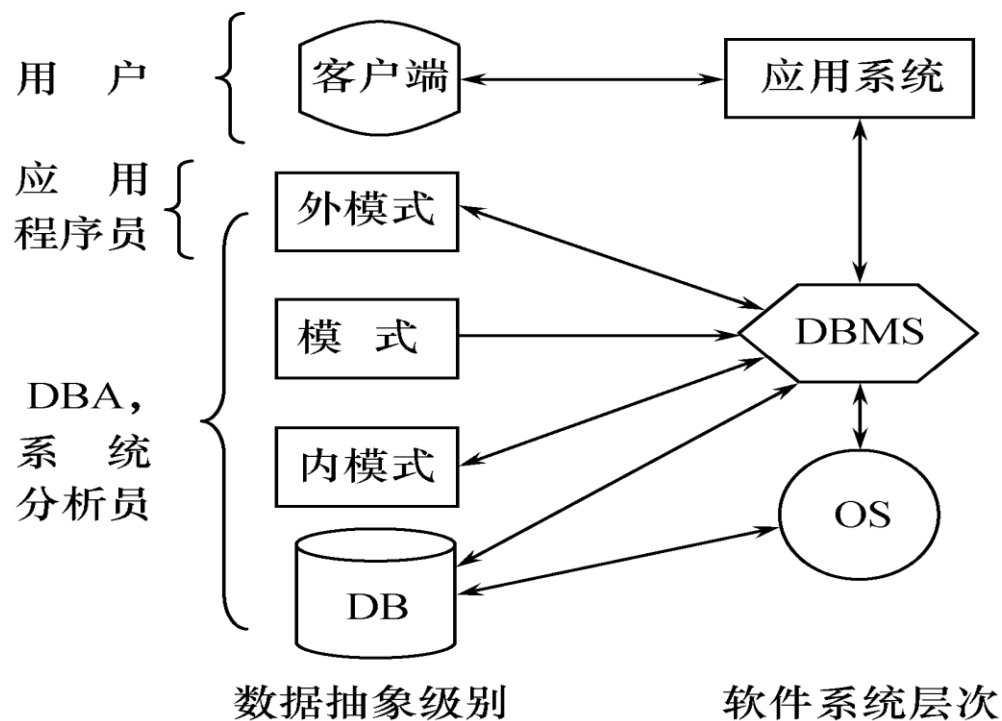


图1.30 各种人员的数据视图

第一章 绪论

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

1.5 小结

1.5 小结

❖ 数据库系统概述

- 数据库的基本概念
- 数据管理的发展过程

❖ 数据模型

- 数据模型的三要素
- 概念模型， E-R 模型
- 三种主要数据库模型

小结(续)

❖ 数据库系统的结构

- 数据库系统三级模式结构
- 数据库系统两层映像系统结构

❖ 数据库系统的组成

数据库领域五位图灵奖获得者

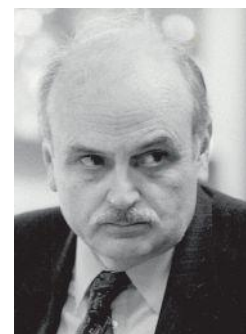
❖ 1973年图灵奖得主

- 查尔斯.巴赫曼 (Charles W. Bachman)
- ——“网状数据库之父”



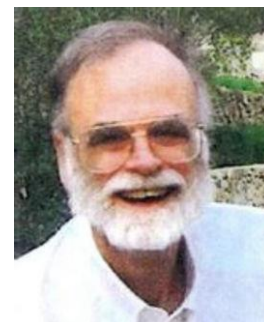
❖ 1981年图灵奖得主

- 埃德加.科德 (Edgar Frank Codd)
- ——“关系数据库之父”



❖ 1998年图灵奖得主

- 詹姆斯.格雷 (James Gray)
- ——数据库技术和“事务处理”专家



数据库领域五位图灵奖获得者

❖ 2014年图灵奖得主

- 迈克尔·斯通布雷克（Michael Stonebraker）
- ——数据库基本概念和实际技术



❖ 2021年图灵奖得主

- 杰弗瑞·大卫·厄尔曼（Jeffrey David Ullman）
- ——数据库理论、编程语言
（编程语言实现的基础算法和理论）



下课了。。。

追求



休息一会儿。。。

