

# 数据库系统概论

## Introduction to Database Systems

---

中国人民大学信息学院

School of Information  
Renmin University of China  
2023

# 教材及参考书

## 教材

王珊,杜小勇,陈红. 数据库系统概论 (第6版)  
北京:高等教育出版社,2023

## 参考书

Abraham Silberschatz, Henry F. Korth, S.Sudarsha  
DATABASE SYSTEM CONCEPTS, 7th Edition, 2020

## 上机软件

推荐 KingBase ES

从北京人大金仓信息技术股份有限公司的网站

<http://www.kingbase.com.cn/> 上免费 下载



# 学习方式

## 听课/MOOC课程

(启发式、讨论式)

## 读书

(预习、复习)

## 报告

(综合练习)



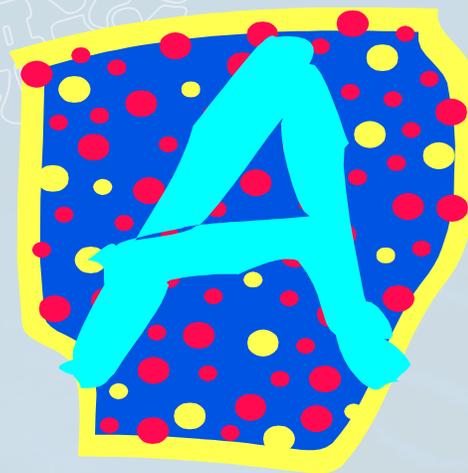
# 考试成绩

## ❖ 平时成绩

(书面作业、上机练习、综合练习)

## ❖ 期中考试

## ❖ 期末考试



# 内容安排 (1)

## 基础篇

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

## 设计与应用开发篇

- 第6章 关系数据理论
- 第7章 数据库设计
- 第8章 数据库编程



# 内容安排 (2)

## 系统篇

- 第9章 关系数据库存储管理
- 第10章 关系查询处理和查询优化
- 第11章 数据库恢复技术
- 第12章 并发控制
- \* 第13章 数据库管理系统概述

第1至第13章是本科专业的基本教程  
(书中有\*号的部分除外)

第14至第18章是新技术篇 (本科生、研究生选读)



# \*内容安排 (3)

## 新技术篇

- 第14章 数据库发展概述
- 第15章 大数据管理系统
- 第16章 数据仓库与联机分析处理
- 第17章 内存数据库系统
- 第18章 区块链与数据库



# 数据库系统概论

## 第1章 绪论



# 第1章 绪论

## 1.1 数据库系统概述

## 1.2 数据模型

## 1.3 数据库系统的三级模式结构

## 1.4 数据库系统的组成

## \* 1.5 数据库系统的体系结构

## 本章小结



# 数据库的地位

- 数据库技术产生于**1960**年代，是数据管理的核心技术
- 数据库管理系统是大型复杂基础软件，是现代信息系统核心和基础
- 数据库技术是形成**良性循环的典范**：应用需求→理论研究→技术创新→产品开发→广泛应用
- 数据库技术是计算机科学与技术的重要分支，计算机科学与技术中发展最快领域之一，应用最广的技术之一，极大地促进了计算机应用向各行各业的渗透。
- 数据库已经成为每个人生活中不可缺少的部分，大数据应用、云计算技术的迅猛发展，越来越凸显出数据库技术的重要性。



# 二维码 1.1

数据库的辉煌成就：

数据库领域图灵奖获得者简介



# 本章导读

## ❖ “盲人摸象”勾画出数据库系统的轮廓

- 数据库若干基本概念
- 数据库发展历史
- 数据建模和数据模型
- 数据库系统的模式结构
- 数据库系统组成
- 了解学习数据库系统的必要性和重要性



# 1.1 数据库系统概述

## 1.1.1 数据库的4个基本概念

## 1.1.2 数据管理技术的产生和发展



# 1.1.1 数据库的4个基本概念

- ❖ 数据 (**Data**)
- ❖ 数据库 (**DataBase, DB**)
- ❖ 数据库管理系统 (**DataBase Management System, DBMS**)
- ❖ 数据库系统 (**DataBase System, DBS**)



# 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

■ 文本、图形、图像、音频、视频、互联网上的博客、微信中的聊天记录、学生的档案记录、个人的网购记录、医院病历等



# 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 **93**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：计算机科学与技术专业**2018**级学生人数

语义4：请同学给出.....



# 数据举例（续）

## ❖ 学生档案中的学生记录

**（20180002，刘晨，女，1999-9-1，计算机科学与技术）**

■ 语义：学号、姓名、性别、出生日期、主修专业

■ 解释：学号为**20180002**的同学，姓名**刘晨**，性别**女**，**1999年9月1日**出生，**计算机科学与技术**专业



## 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、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
- 较小的冗余度
- 较高的数据独立性
- 可扩展性



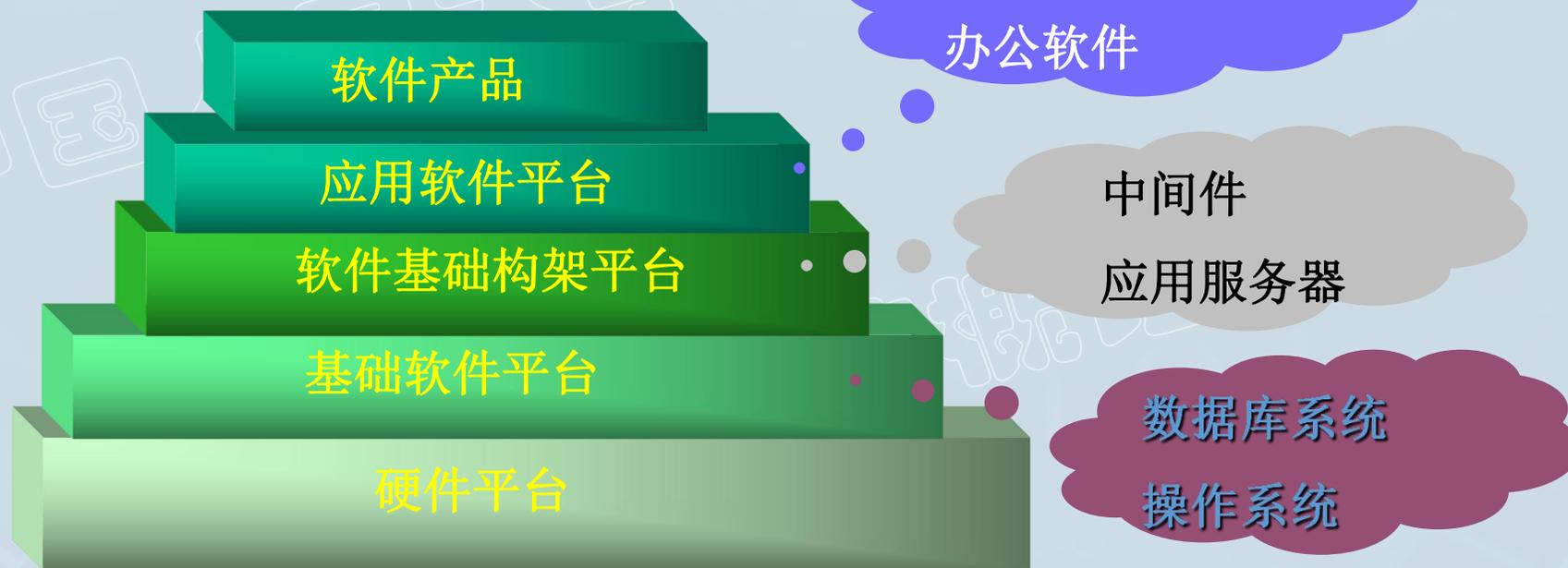
# 3.数据库管理系统

## ❖ 什么是数据库管理系统

- 位于用户与操作系统之间的一层数据管理软件
- 计算机的基础软件，是一个大型复杂的软件系统



# 数据库在计算机系统的位置



# 数据库管理系统的主要功能

## ■①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的组成与结构

## ■②数据组织、存储和管理功能

- 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



# 数据库管理系统的主要功能（续）

## ■③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）

## ■④数据库的事务管理和运行管理功能

- 数据库在建立、运行和维护时由数据库管理系统统一管理和控制
- 保证数据的安全性、完整性
- 多用户对数据的并发使用及发生故障后的系统恢复



# 数据库管理系统的主要功能（续）

## ■⑤数据库的建立和维护功能

- 数据库初始数据的输入和转换
- 数据库转储和恢复功能
- 数据库的重组、性能监视和数据分析等

## ■⑥其它功能

- 数据库管理系统与网络中其它软件系统的通信
- 数据库管理系统系统之间或与文件系统的数据库转换
- 异构数据库之间的互访和互操作



# 4.数据库系统

❖ 数据库系统（**database system**，简称**DBS**）

❖ 数据库系统的构成

■ 数据库

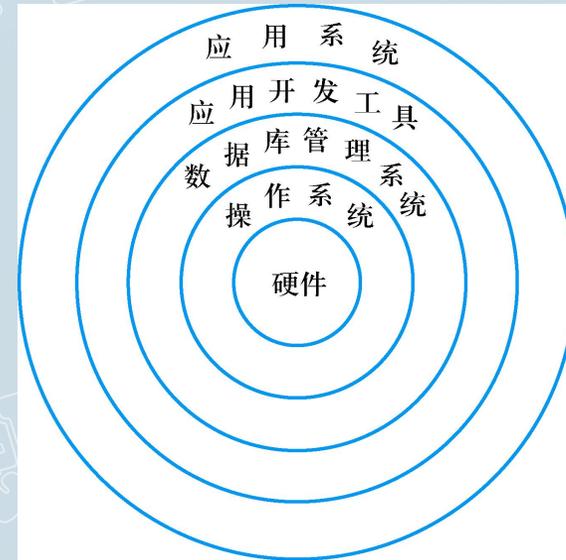
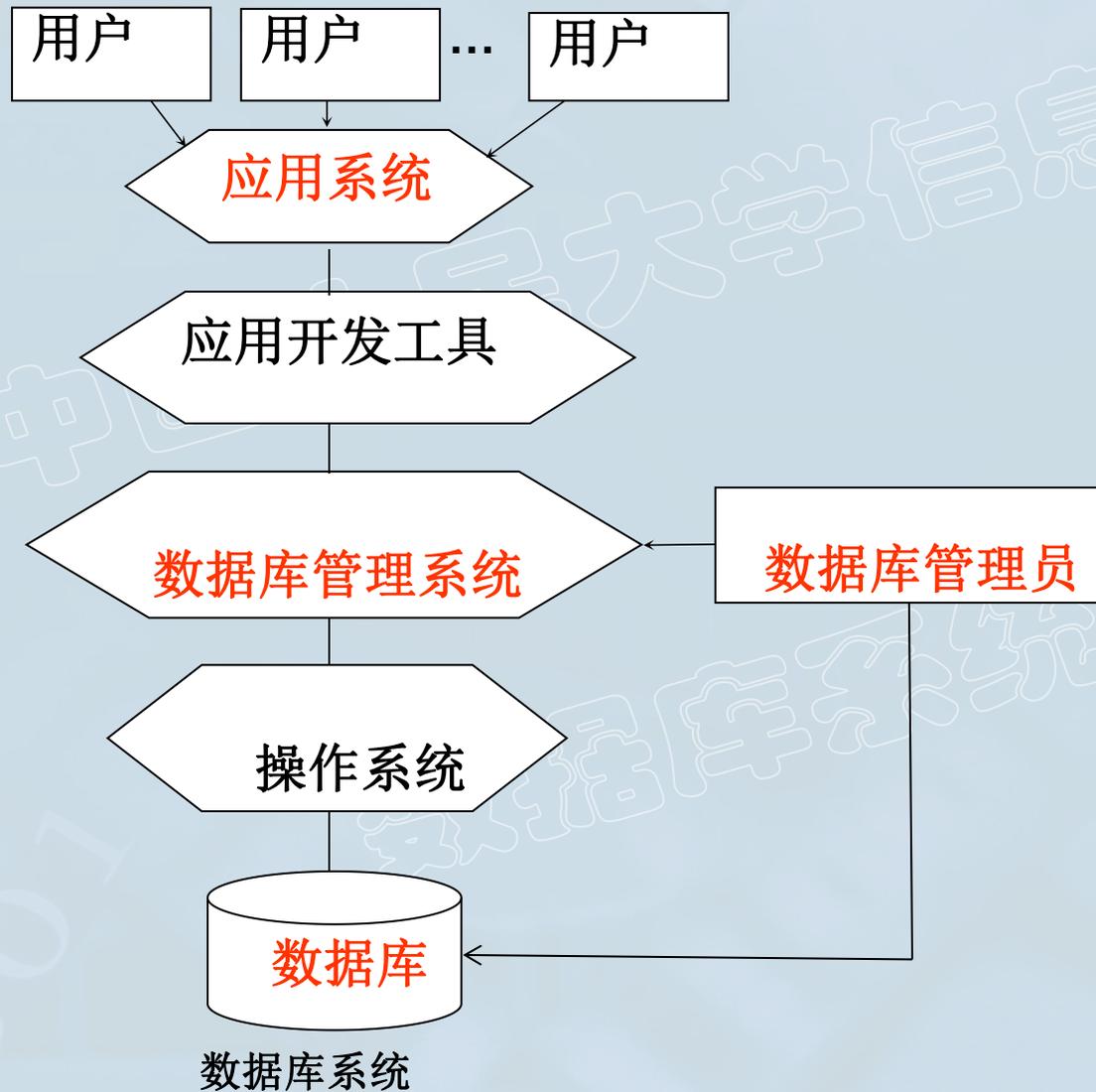
■ 数据库管理系统（及外围的应用开发工具）

■ 应用程序

■ 数据库管理员



# 数据库系统（续）



引入数据库后计算机系统的层次机构



# 1.1 数据库系统概述

1.1.1 数据库的4个基本概念

1.1.2 数据管理技术的产生和发展



# 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50年代中期以前）
- 文件系统阶段（20世纪50年代后期-- 20世纪50年代中期）
- 数据库系统阶段（20世纪50年代后期--）



# 数据管理技术的产生和发展（续）

表 1.1 数据管理技术三个阶段的比较

对比项	人工管理阶段	文件系统阶段	数据库系统阶段
应用领域	科学计算	科学计算、数据管理	大规模数据管理
主要硬件	无直接存取存储设备	磁盘、磁鼓	大容量磁盘、磁盘阵列
主要软件	没有操作系统，没有管理数据的专门软件	有文件系统	有数据库管理系统
处理方式	批处理	联机实时处理、批处理	联机实时处理、分布处理、批处理
数据的管理者	人(程序员)	文件系统	数据库管理系统
数据面向的对象	某一应用程序	某一应用	现实世界(部门、企业、跨国组织等)
数据的共享程度	不共享，冗余度极大	共享性弱，冗余度高	共享性强，冗余度低且易扩充
数据的独立性	不独立，完全依赖应用程序	独立性弱	具有较强的物理独立性和一定的逻辑独立性
数据的结构化	无结构	记录内有结构、整体无结构	整体结构化，用数据模型描述
数据控制能力	应用程序自己控制	应用程序自己控制	由数据库管理系统提供数据安全性、完整性、并发性控制和数据库恢复功能



# 1.人工管理阶段

## ❖ 时期

- 20世纪50年代中期以前

## ❖ 产生的背景

- 应用背景            科学计算
- 硬件背景            无直接存取的存储设备
- 软件背景            没有操作系统
- 处理方式            批处理



# 人工管理阶段（续）

## ❖ 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



# 应用程序与数据的对应关系（人工管理阶段）



人工管理阶段应用程序与数据之间的一一对应关系



## 2. 文件系统阶段

### ❖ 时期

■ 20世纪50年代后期-- 20世纪60年代中期

### ❖ 产生的背景

- 应用背景      科学计算、数据管理
- 硬件背景      磁盘、磁鼓
- 软件背景      有文件系统
- 处理方式      联机实时处理、批处理



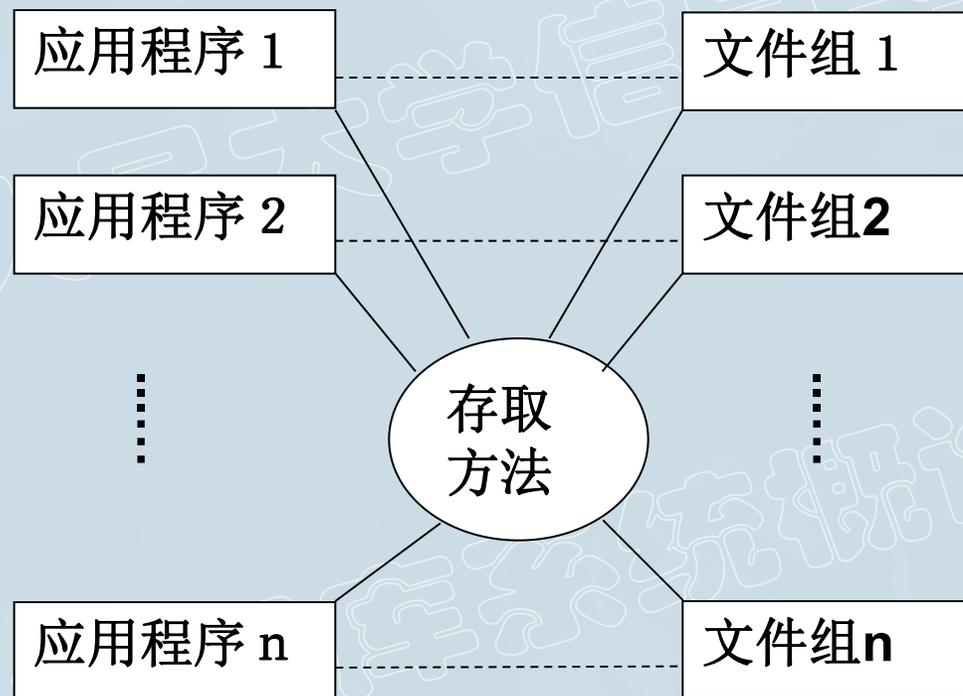
# 文件系统阶段（续）

## ❖ 特点

- 数据的管理者：文件系统，数据可长期保存
- 数据面向的对象：某一应用
- 数据的共享程度：共享性差、冗余度大
- 数据的结构化：记录内有结构，整体无结构
- 数据的独立性：独立性差
- 数据控制能力：应用程序自己控制



# 应用程序与数据的对应关系（文件系统阶段）



文件系统阶段应用程序与数据之间的对应关系



# 3. 数据库系统阶段

## ❖ 时期

- 20世纪60年代后期以来

## ❖ 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



# 数据库系统管理数据的特点

- ❖ (1) 整体数据的结构化
- ❖ (2) 数据的共享性强，冗余度低且易于扩充
- ❖ (3) 数据的独立性强
- ❖ (4) 数据由数据库管理系统统一管理和控制



# (1) 整体数据的结构化

❖ 整体数据的结构化是数据库的主要特征之一

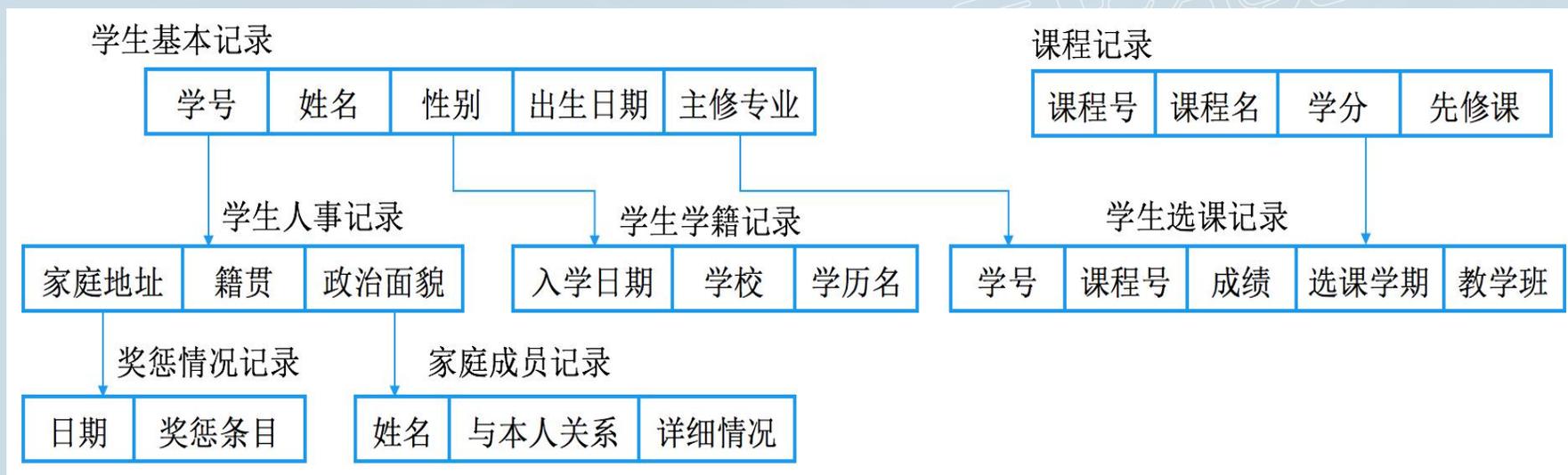
❖ 整体数据的结构化

- 不仅仅针对某一个应用，而是面向整个组织或企业的多种应用需求
- 不仅数据本身结构化，整体是结构化的，数据之间具有联系
- 数据记录可以变长
- 数据的最小存取单位是数据项

数据用数据模型描述，无需应用程序定义



# 数据结构化（续）



“高校本科教务管理”信息系统中与学生有关的数据结构

- 为各部门的应用提供必要的记录，整体数据结构化
- 描述数据时不仅描述数据本身，还要描述数据之间的联系



二维码 1.2

# 文件系统与数据库系统

## 操作比较示例



# 二维码 1.2

## 问题描述

- 分别用文件系统（C++编写程序）和数据库系统 构建“学生-获奖”应用系统
- 描述：学生信息表，学号唯一；获奖最多列出5项。

学号 Sno	姓名 Sname	性别 Ssex	出生日期 Sbirthdate	主修专业 Smajor	奖励1	奖励2	奖励3	奖励4	奖励5
20180001	李勇	男	2000-3-8	信息安全	2019年全国大学生信息安全竞赛一等奖	2020年萨师焯精英基金特等奖	2020年中国大学生计算机设计大赛一等奖		
20180002	刘晨	女	1999-9-1	计算机科学与技术	2019年萨师焯精英基金一等奖	2020年中国“互联网+”大学生创新创业大赛一等奖	2021年CCSP大学生计算机系统与程序设计竞赛一等奖		
20180005	陈新奇	男	2001-11-1	信息管理与信息系统	2019年北京市三好学生	2021中国大学生计算机设计大赛一等奖			
20180006	赵明	男	2000-6-12	数据科学与大数据技术	2019中国大学生计算机设计大赛一等奖	2021年萨师焯精英基金二等奖	2021年CCSP大学生计算机系统与程序设计竞赛一等奖		

- 完成数据插入功能。
- 实现查询功能，例如根据学号查找学生基本信息，查找获得的奖励情况。



## (2) 数据的共享性强，冗余度低且易于扩充

- ❖ 数据面向整个系统，可以被多个用户、多个应用、使用不同的接口、不同的编程语言共享使用。
- ❖ 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



### (3) 数据的独立性强

#### ❖ 物理独立性

- 用户的应用程序与数据库中数据的物理存储是相互独立的
- 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，应用程序不用改变。

#### ❖ 数据独立性由数据库管理系统的两级映像功能来保证



## (4) 数据由数据库管理系统统一管理和控制

### ❖ 数据库管理系统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数据的泄密和破坏

#### ■ ② 数据的完整性 (**integrity**) 检查

保证数据的正确性、有效性和相容性

#### ■ ③ 数据的并发控制

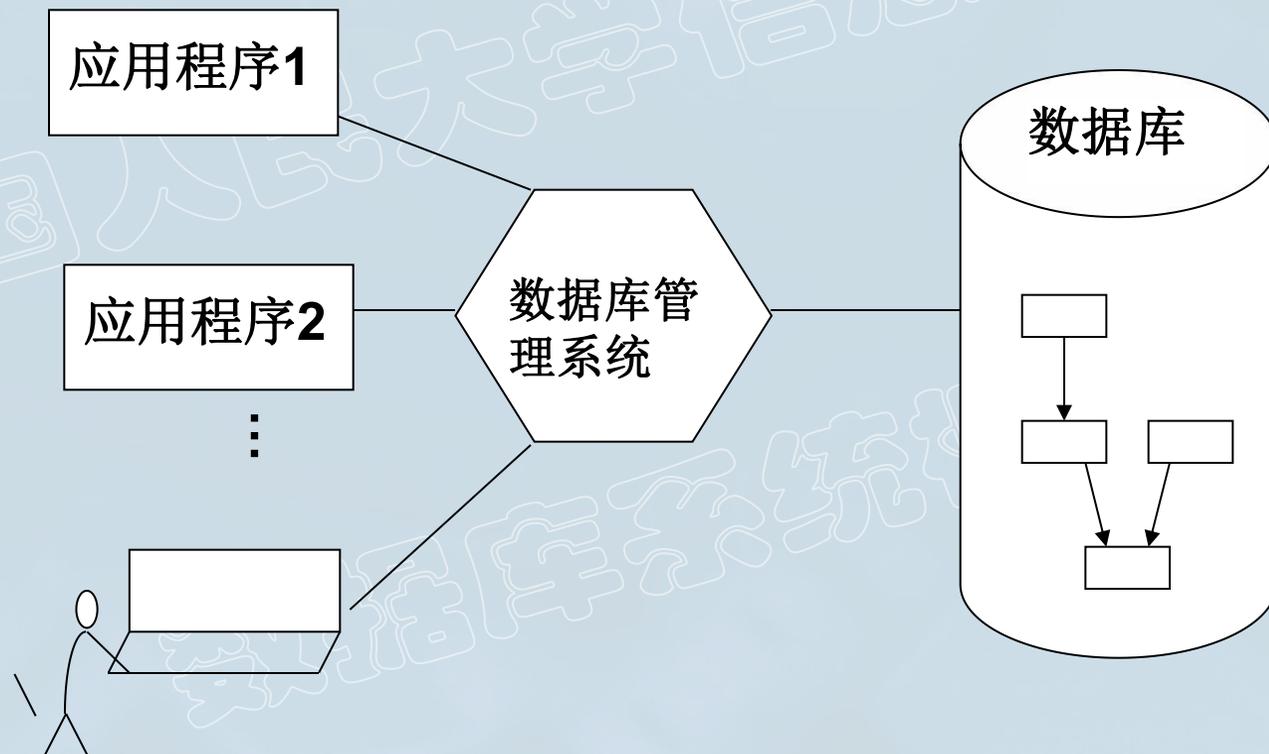
对多用户的并发操作加以控制和协调，防止相互干扰而得到错误的结果

#### ■ ④ 数据库的恢复 (**recovery**)

将数据库从错误状态恢复到某一已知的正确状态（完整状态或一致状态）



# 应用程序与数据的对应关系（数据库系统阶段）



数据库系统阶段应用程序与数据之间的对应关系



# 数据库概念小结

- ❖ 数据库是长期存储在计算机内有组织、可共享的大量数据的集合
- ❖ 可以供各种用户共享，具有最小冗余度和较高的数据独立性
- ❖ 数据库管理系统在数据库建立、运维护时对数据库进行统一控制，以保证数据的完整性、安全性，并在多用户同时使用数据库时进行并发控制，在发生故障后对数据库进行恢复。



# 第1章 绪论

1.1 数据库系统概述

1.2 数据模型

1.3 数据库系统的三级模式结构

1.4 数据库系统的组成

\*1.5 数据库系统的体系结构

本章小结



## 1.2 数据模型

- ❖ 数据模型是对现实世界数据特征的抽象，是现实世界的模拟
- ❖ 数据模型是用来描述数据、组织数据和对数据进行操作的
- ❖ 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
- ❖ 数据模型是数据库系统的核心和基础



# 1.2 数据模型

## 1.2.1 数据建模

## 1.2.2 概念模型

## 1.2.3 数据模型的三要素

## 1.2.4 层次模型

## 1.2.5 网状模型

## 1.2.6 关系模型

## 1.2.7 数据库领域中不断涌现的数据模型

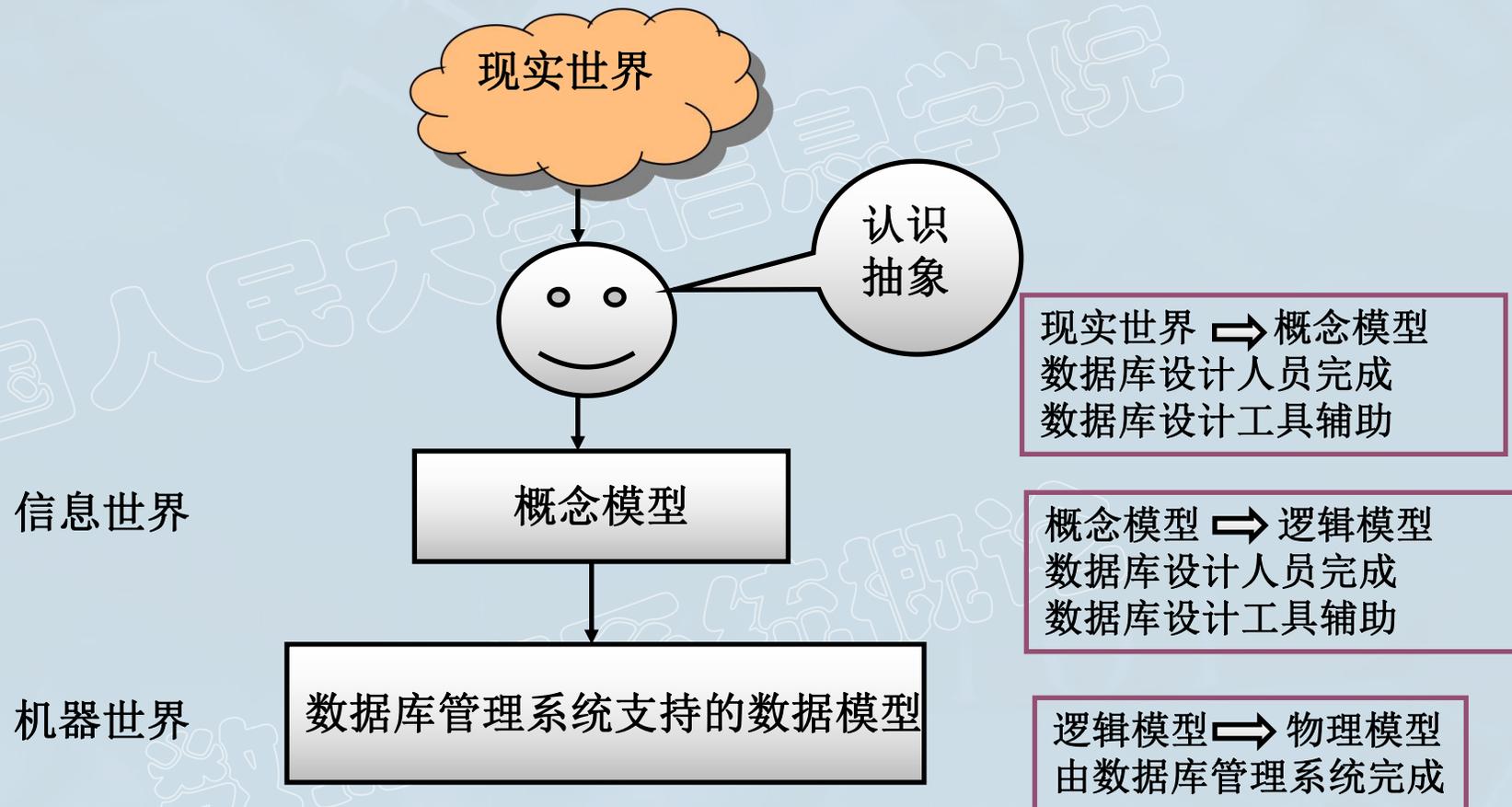


# 数据建模

- ❖ 把现实世界中的具体事物抽象、组织为某一数据库管理系统支持的数据模型
- ❖ 数据建模过程---**两步抽象**
  - 现实世界中的客观对象抽象为概念模型
    - 将现实世界抽象为信息世界
  - 把概念模型转换为某一数据库管理系统支持的数据模型
    - 将信息世界转换为机器世界



# 数据建模（续）



现实世界中客观对象的抽象过程



# 1.2 数据模型

1.2.1 数据建模

1.2.2 概念模型

1.2.3 数据模型的三要素

1.2.4 层次模型

1.2.5 网状模型

1.2.6 关系模型

1.2.7 数据库领域中不断涌现的数据模型



## 1.2.2 概念模型

### ❖ 概念模型的用途

- 是现实世界到机器世界的一个中间层次
- 用于信息世界的建模
- 现实世界到信息世界的第一层抽象
- 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
- 简单、清晰、易于用户理解



# 概念模型（续）

1. 信息世界中的基本概念
2. 概念模型的一种表示方法：实体-联系模型



# 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、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



# 信息世界中的基本概念（续）

## （4）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

## （5）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）

## （6）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。
- **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
- 实体之间的联系有**一对一**、**一对多**和**多对多**等多种类型



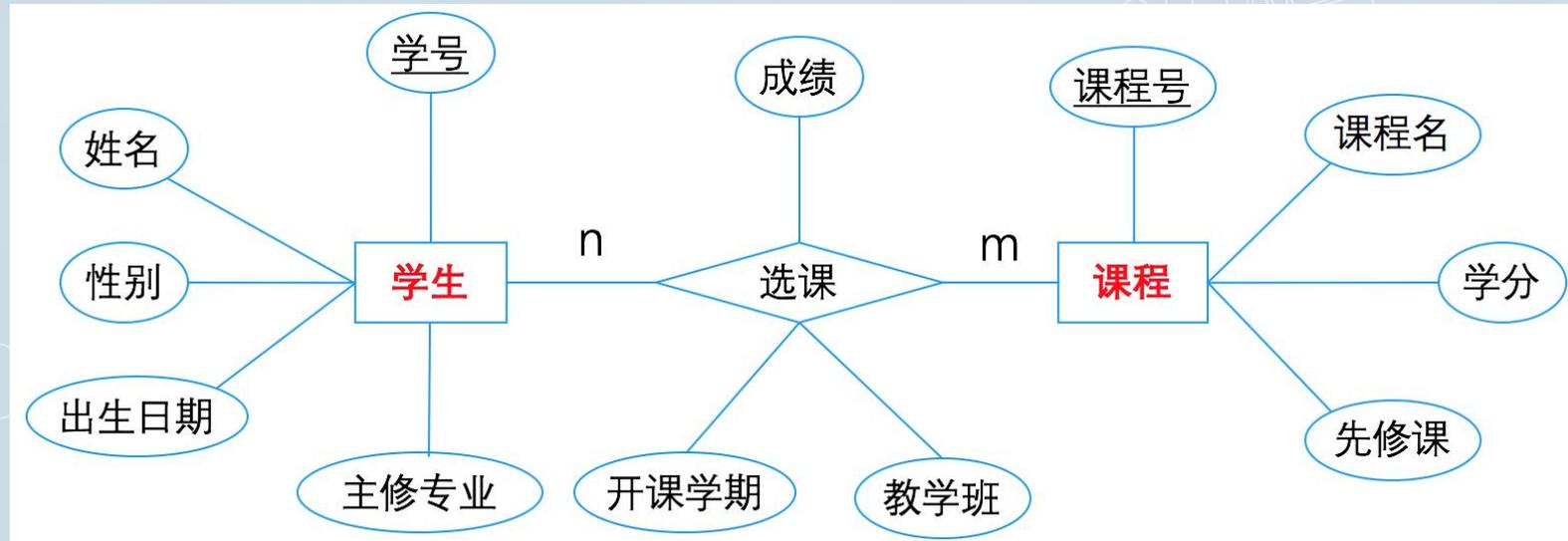
## 2.概念模型的一种表示方法： 实体-联系模型

### ❖ 实体-联系方法（**Entity-Relationship Approach**）

- 用**E-R**图来描述现实世界的概念模型
- **E-R**方法或**E-R**模型



# 实体-联系模型（续）



学生选课E-R图示例

- ❖ 抽象了学校中的学生和课程两个客观事物：学生实体和课程实体
- ❖ 抽象了现实世界中事物之间的联系：
  - 一门课程可以有多名学生选修，一个学生可以选修多门课程
  - 用课程实体与学生实体多对多（m:n）联系来描述



## 1.2 数据模型

1.2.1 数据建模

1.2.2 概念模型

1.2.3 数据模型的三要素

1.2.4 层次模型

1.2.5 网状模型

1.2.6 关系模型

1.2.7 数据库领域中不断涌现的数据模型



# 数据模型的三要素

- ❖ 1. 数据结构
- ❖ 2. 数据操纵
- ❖ 3. 完整性约束



# 1. 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

- 与对象的类型、内容、性质有关
- 与数据之间联系有关

❖ 数据结构是对系统静态特性的描述



## 2. 数据操纵

### ❖ 数据操纵

- 对数据库中各种对象（型）的实例（值）允许执行的操作的集合，包括操作及有关的操作规则

### ❖ 数据库主要操作

- 查询
- 更新（包括插入、删除、修改）



# 数据操纵（续）

## ❖ 数据模型必须定义

- 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

## ❖ 数据操纵是对系统动态特性的描述



# 3. 完整性约束

## ❖ 完整性约束

- 一组完整性规则
- 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依存规则
- 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、有效和相容



# 完整性约束（续）

## ❖ 数据模型对完整性约束条件的定义

### ■ 反映和规定必须遵守的**基本的**和**通用的**完整性约束

- 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

## ❖ 数据模型都是逻辑上的

## ❖ 物理模型：数据模型以一定的组织方式存储在数据库管理系统中，是数据模型在数据库管理系统内部的物理存储结构

- 描述数据在数据库管理系统内部的数据组织和存取方法的实现，包括存储结构和索引结构的物理实现



## 1.2 数据模型

1.2.1 数据建模

1.2.2 概念模型

1.2.3 数据模型的三要素

1.2.4 层次模型

1.2.5 网状模型

1.2.6 关系模型

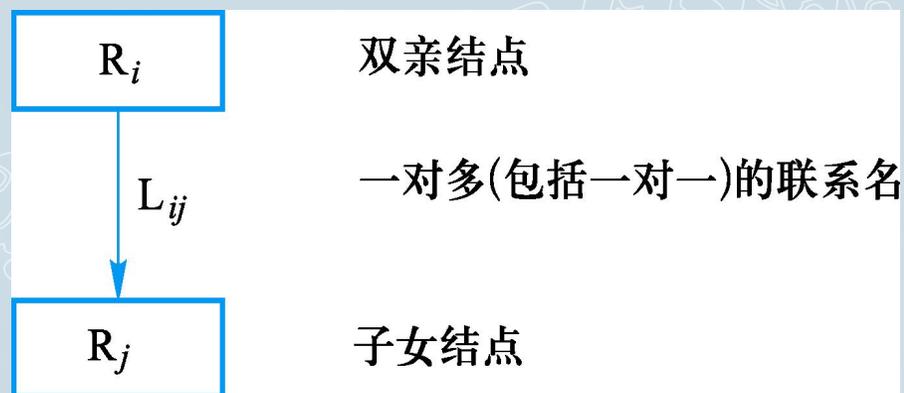
1.2.7 数据库领域中不断涌现的数据模型



## 1.2.4 层次模型

### ❖ 层次模型

- 实体用记录表示
- 实体的属性对应记录的数据项（或字段）
- 实体之间的联系转换成记录之间的两两联系
- 数据结构的单位是基本层次联系
  - 是指两个记录以及它们之间的一对多（包括一对一）的联系



# 层次模型（续）

- ❖ 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公司的IMS（**Information Management System**）
- ❖ 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. 层次模型的数据结构

## ❖ 层次模型

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



# 层次模型的数据结构（续）

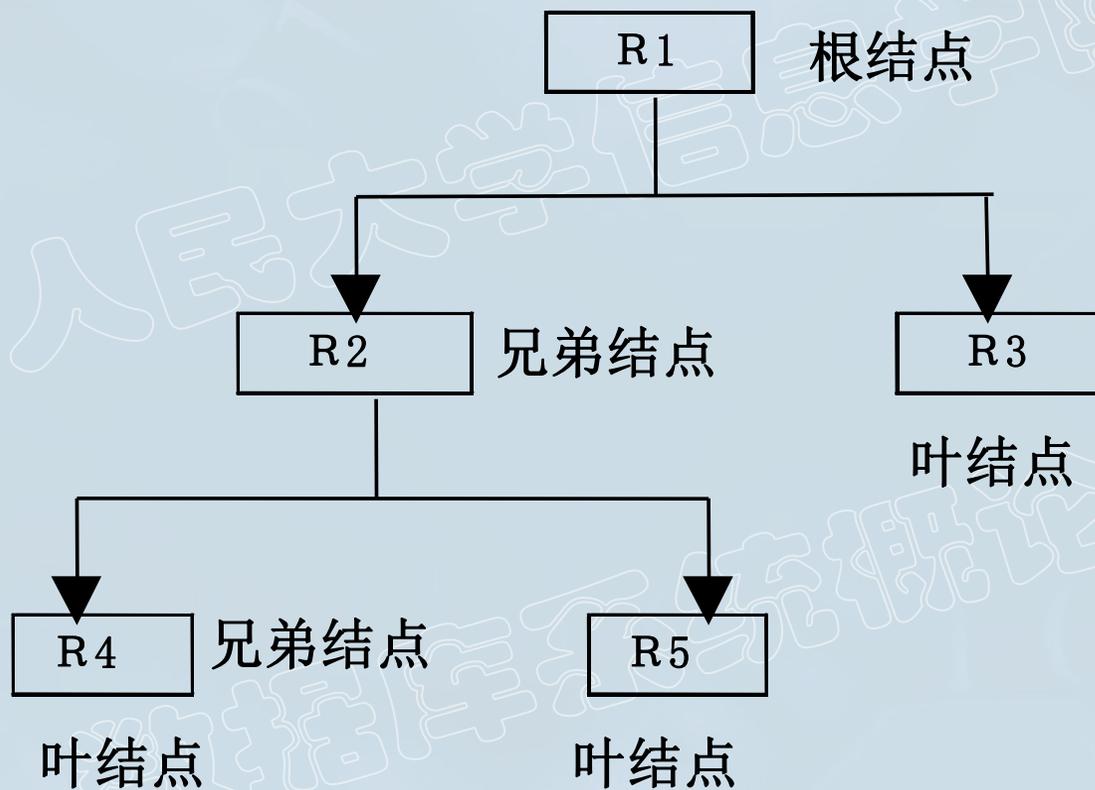


图1.10 一个层次模型的示例



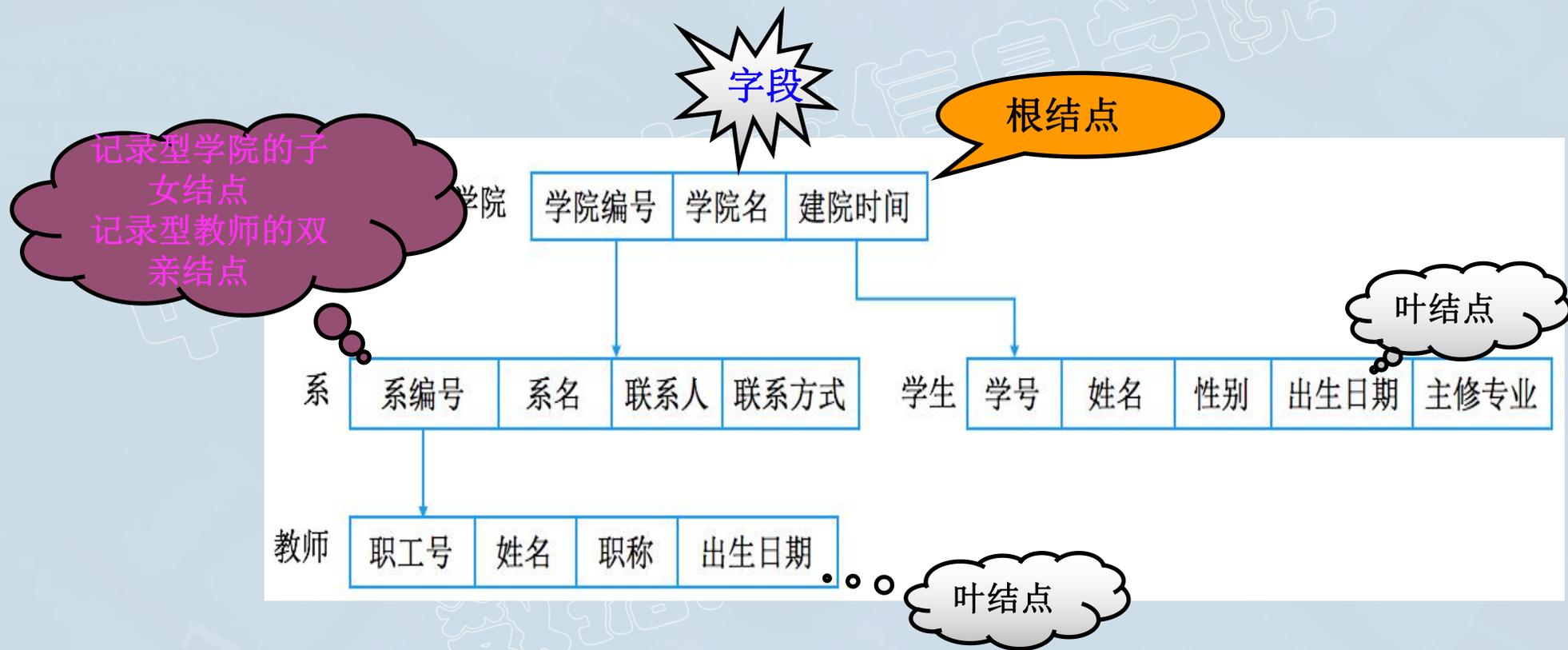
# 层次模型的数据结构（续）

## ❖ 层次模型的特点：

- 结点的双亲是唯一的
- 只能直接处理一对多的实体联系
- 每个结点表示一个记录类型，记录类型之间的联系用结点之间的连线（有向边）表示
- 各个记录类型、同一记录类型中各个字段不能同名
- 每个记录类型可以定义一个排序字段，也称为码字段
- 任何记录值只有按其路径查看，没有一个子女记录值能够脱离双亲记录值而独立存在



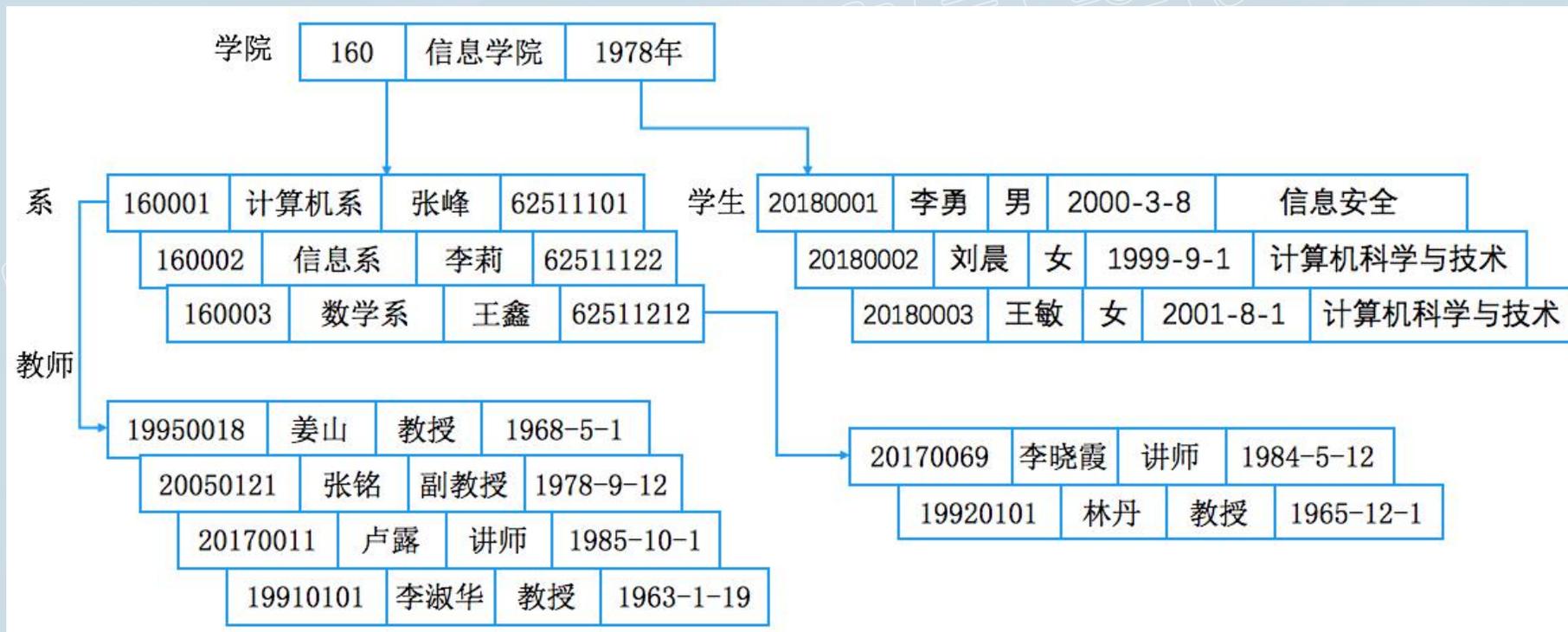
# 层次模型的数据结构（续）



“学生学籍管理”子系统的层次模型示意图



# 层次模型的数据结构（续）



“学生学籍管理”子系统层次模型的一个值



## 2. 层次模型的数据操纵与完整性约束

### ❖ 层次模型的数据操纵

■ 查询

■ 插入

■ 删除

■ 更新



# 层次模型的数据操纵与完整性约束（续）

## ❖ 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



# 3.层次模型的优缺点

## ❖ 优点

- 层次模型的数据结构比较简单清晰
- 查询效率高
  - 层次模型中记录之间的联系用有向边表示，就是记录之间的存取路径
- 层次数据模型提供了良好的完整性约束支持

## ❖ 缺点

- 很多联系是非层次性，不适合用层次模型表示
- 一个结点具有多个双亲结点，只能通过冗余数据（易产生不一致性）或创建非自然的数据结构（虚拟结点）来解决
- 对插入和删除操作的限制多，应用程序的编写比较复杂
- 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 数据模型

1.2.1 数据建模

1.2.2 概念模型

1.2.3 数据模型的三要素

1.2.4 层次模型

1.2.5 网状模型

1.2.6 关系模型

1.2.7 数据库领域中不断涌现的数据模型



## 1.2.5 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
  - 20世纪70年代由**DBTG**提出的一个系统方案
- ❖ 实际系统
  - **Cullinet Software**公司的 **IDMS**
  - **Univac**公司的 **DMS1100**
  - **Honeywell**公司的**IDS/2**
  - **HP**公司的**IMAGE**



# 1. 网状模型的数据结构

## ❖ 网状模型

满足下面两个条件的基本层次联系集合：

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



# 网状模型的数据结构（续）

## ❖ 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（实体）

**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（实体）之

间的一对多的父子联系



# 网状模型的数据结构（续）

## ❖ 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描述现实世界
- 层次模型实际上是网状模型的一个特例



# 网状模型的数据结构（续）

- ❖ 网状模型中子女结点与双亲结点的联系可以不唯一，要为每个联系命名，并指出与该联系有关的双亲记录和子女记录

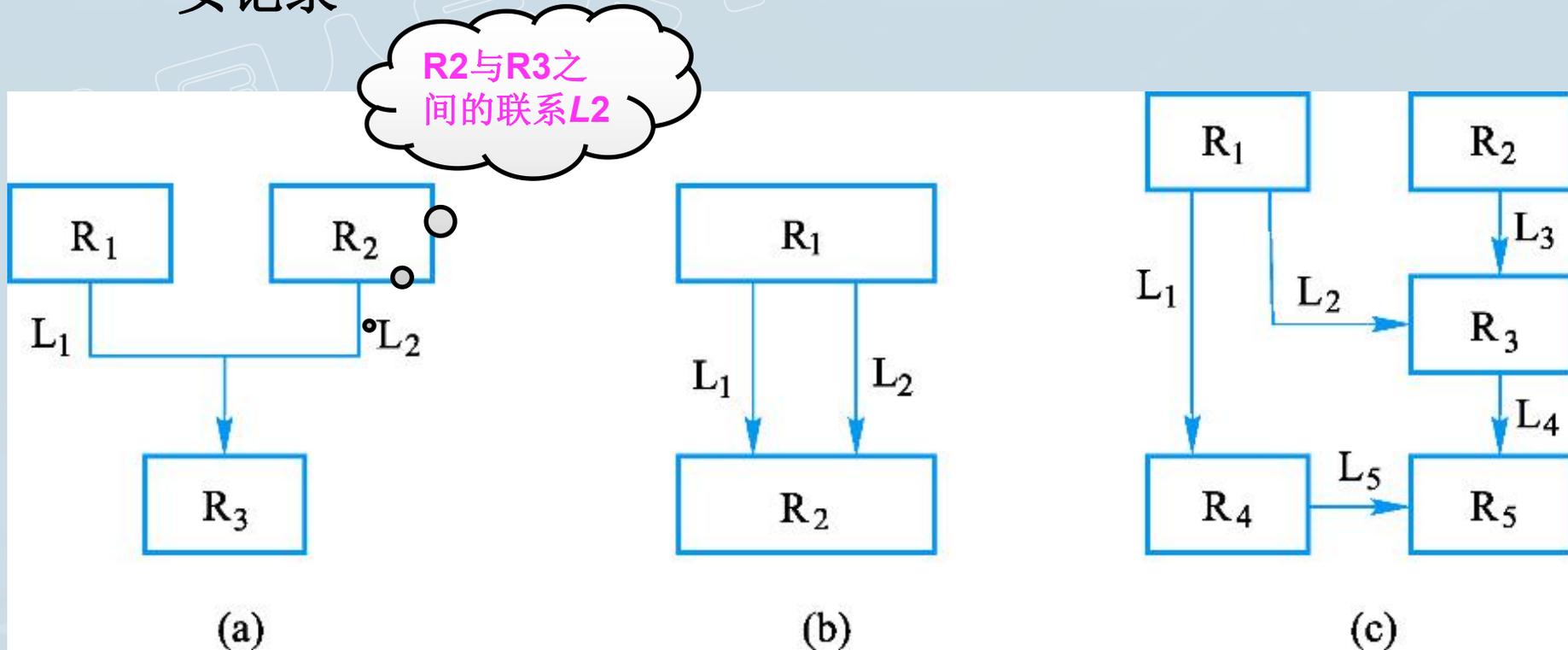


图1.13 网状模型示例



# 网状模型的数据结构（续）

## 多对多联系在网状模型中的表示

- 用网状模型**间接**表示多对多联系

- 方法：

  - 将多对多联系**直接**分解成一对多联系



# 网状模型的数据结构（续）

例如：一个学生可以选修若干门课程，某一课程可以被多个学生选修，学生与课程之间是多对多联系

■ 引进一个学生选课的连接记录，由5个数据项组成

- 学号
- 课程号
- 成绩
- 选课学期
- 教学班



图1.14 “学生选课”子系统的网状模型



## 2. 网状模型的数据操纵与完整性约束

❖ 网状数据库系统（如**DBTG**）对数据操纵加限制，提供了一定的完整性约束

- 码：唯一标识记录的数据项的集合
- 一个联系中双亲记录与子女记录之间是一对多联系
- 支持双亲记录和子女记录之间某些约束条件



# 3. 网状模型的优缺点

## ❖ 优点

- 能够更直接地描述现实世界，如一个结点可以有多个双亲，结点之间可以有多种联系
- 具有良好的性能，存取效率较高

## ❖ 缺点

- 结构比较复杂，随着应用环境的扩大，数据库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，要嵌入某一种高级语言中，用户不容易使用
- 记录之间联系是通过存取路径实现，用户必须了解系统结构的细节



## 1.2 数据模型

1.2.1 数据建模

1.2.2 概念模型

1.2.3 数据模型的三要素

1.2.4 层次模型

1.2.5 网状模型

1.2.6 关系模型

1.2.7 数据库领域中不断涌现的数据模型



## 1.2.6 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
- ❖ 1980年代以来，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



# 1. 关系模型的数据结构

- 从用户观点看，关系模型由一组关系组成
- 每个关系的数据结构是一张规范化的二维表

学生登记表

属性

学号 Sno	姓名 Sname	性别 Ssex	出生日期 Sbirthdate	主修专业 Smajor
20180001	李勇	男	2000-3-8	信息安全
20180002	刘晨	女	1999-9-1	计算机科学与技术
20180003	王敏	女	2001-8-1	计算机科学与技术
20180004	张立	男	2000-1-8	计算机科学与技术
20180005	陈新奇	男	2001-11-1	信息管理与信息系统
20180006	赵明	男	2000-6-12	数据科学与大数据技术
20180007	王佳佳	女	2001-12-7	数据科学与大数据技术

元组

表1.2 关系模型的数据结构示例：学生表



# 关系模型的数据结构（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

## ■码（**key**）

- 又称码键或键。表中的某一个属性或一组属性，它的值可以唯一确定一个元组



# 关系模型的数据结构（续）

## ■域（domain）

- 是一组具有相同数据类型的值的集合。属性的取值范围来自某个域。

## ■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，性别，出生日期，主修专业）



# 关系模型的数据结构（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

表1.3中联系方式是可分的数据项，不符合关系模型要求

学号	姓名	性别	出生日期	主修专业	联系方式		
					手机号	Email	微信号
20180001	李勇	男	2000-3-8	信息安全	18301200745	liyong@ qq. com	liyong@ ruc
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

表1.3 非规范化的表示例：表中有表



# 关系模型的数据结构（续）

表1.4 关系术语与现实生活中表格使用的术语对比

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）



## 2. 关系模型的数据操纵与完整性约束

❖ 数据操作是集合操作，操作对象和操作结果都是关系

■ 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”



# 关系模型的数据操纵与完整性约束（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



# 3. 关系模型的优缺点

## ❖ 优点

- 建立在严格的数学概念的基础上

- 概念单一

- 实体和实体之间联系都用关系来表示
- 对数据的检索和更新结果也是关系

- 关系模型的存取路径对用户是隐蔽的

- 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



# 关系模型的优缺点（续）

## ❖ 缺点

- 存取路径对用户隐蔽，查询效率往往不如层次模型和网状模型
- 为提高性能，必须对用户的查询请求进行优化，增加了开发数据库管理系统的难度



## 1.2 数据模型

1.2.1 数据建模

1.2.2 概念模型

1.2.3 数据模型的三要素

1.2.4 层次模型

1.2.5 网状模型

1.2.6 关系模型

1.2.7 数据库领域中不断涌现的数据模型



# 1.2.7 数据库领域中不断涌现的数据模型

## ❖ 数据模型的发展进程

- 层次模型、网状模型

- 关系模型

- 面向对象数据模型、对象关系数据模型

- 半结构化的XML数据模型等

- 新型数据模型

- 键值对数据模型(Key - Value)、文档数据模型、图数据模型、时序数据模型、时空数据模型、流数据模型、多媒体数据模型等



# 数据库领域中不断涌现的数据模型（续）

- ❖ 相比于关系模型，这些模型按照数据模型应该具备的三个基本要素来衡量不严格
  - 例如对这些数据的完整性约束就需要研究和发



## 1.2 小结

❖ 数据模型的初步概念

❖ 数据建模的过程划分为两个层次

■ 概念模型：按照用户的观点对数据和信息建模

■ 数据模型：按计算机系统的观点对数据建模

❖ 数据模型是对现实世界客观对象的抽象

❖ 数据模型应该满足三方面要求：

■ 比较真实地模拟世界、容易为人所理解、便于在计算机上实现



# 第1章 绪论

1.1 数据库系统概述

1.2 数据模型

1.3 数据库系统的三级模式结构

1.4 数据库系统的组成

\* 1.5 数据库系统的体系结构

本章小结



# 1.3 数据库系统的三级模式结构

- ❖ 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体系结构
- ❖ 根据计算机的系统结构，从**数据库最终用户角度**看，数据库系统的外部体系结构分为：
  - 集中式结构
  - 客户机/服务器（浏览器/应用服务器/数据库服务器）
  - 并行结构
  - 分布式结构
  - 云结构



# 数据库系统的三级模式结构（续）

**1.3.1 数据库系统中模式的概念**

**1.3.2 数据库系统的三级模式结构**

**1.3.3 数据库的两级映像与数据独立性**



# 1.3.1 数据库系统中模式的概念

## ❖ “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：

(学号，姓名，性别，出生日期，主修专业)

一个记录值：

(**20180003**，王敏，女，**2001-8-1**，计算机科学与技术)



# 数据库系统中模式的概念（续）

## ❖ 模式（**schema**）

- 数据库中全体数据的逻辑结构和特征的描述
- 是型的描述，不涉及具体值
- 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
- 实例随数据库中数据的更新而变动



# 数据库系统中模式的概念（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“学生选课”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

- “2019年学生选课”数据库实例与“2020年学生选课”数据库实例不同



# 数据库系统结构（续）

1.3.1 数据库系统中模式的概念

1.3.2 数据库系统的三级模式结构

1.3.3 数据库的两级映像与数据独立性



## 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）

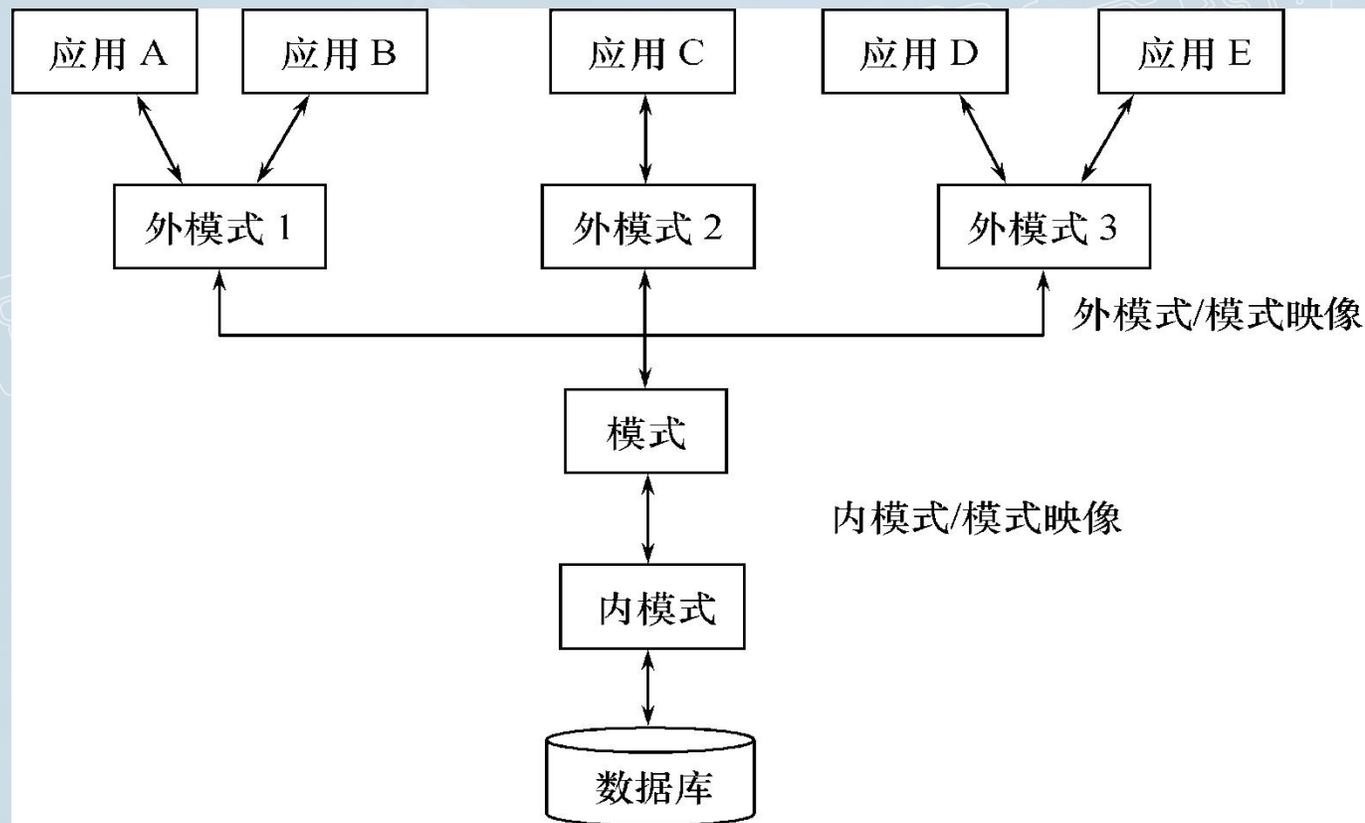


图1.15 数据库系统的三级模式结构



# 1. 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

## ❖ 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（续）

## ❖ 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

## ❖ 数据库管理系统提供模式数据定义语言（模式**DDL**）来严格地定义模式



## 2. 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) 能够看见和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）

- ❖ 一个数据库可以有多个外模式
  - 不同在用户的应用需求、看待数据的方式、对数据保密的要求等方面存在差异
- ❖ 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
- ❖ 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

❖ 数据库管理系统提供外模式数据定义语言（外模式DDL）来严格地定义外模式。



# 3. 内模式 (internal schema)

## ❖ 内模式 (也称物理模式或存储模式)

- 是数据物理结构和组织方式的描述

- 是数据在数据库内部的表示方式

- 记录的存储方式 (堆存储, 按照某个 (些) 属性值的升 (降) 序存储, 或按照属性值聚簇 (cluster) 存储)
- 索引的组织方式 (是B+树索引还是hash索引)
- 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



# 数据库系统结构（续）

1.3.1 数据库系统中模式的概念

1.3.2 数据库系统的三级模式结构

1.3.3 数据库的两级映像与数据独立性



# 数据库的两级映像与数据独立性

- ❖ 三级模式是对数据的三个抽象级别
- ❖ 两级映像 in 数据库管理系统内部实现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



# 1. 外模式 / 模式映像

- ❖ 模式：描述的是数据的全局逻辑结构
- ❖ 外模式：描述的是数据的局部逻辑结构
- ❖ 同一个模式可以有任意多个外模式
- ❖ 每一个外模式，数据库系统都有一个外模式 / 模式映像，  
定义外模式与模式之间的对应关系
- ❖ 映像定义通常包含在各自外模式的描述中



# 外模式 / 模式映像（续）

## 保证数据的逻辑独立性

- 当模式改变时，数据库管理员对外模式 / 模式映像作相应改变，使外模式保持不变
- 应用程序是依据数据的外模式编写的，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



## 2. 模式 / 内模式映像

- ❖ 模式 / 内模式映像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，说明逻辑记录和字段在内部是如何表示的
- ❖ 数据库中模式 / 内模式映像唯一的
- ❖ 该映像定义通常包含在模式描述中



# 模式 / 内模式映像（续）

## ❖ 保证数据的物理独立性

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



# 3.小结

## ❖ 数据库模式（即全局逻辑结构）

- 是数据库的中心与关键
- 独立于数据库的其他层次
- 设计数据库模式结构时应首先确定数据库的逻辑模式



# 小结（续）

## ❖ 数据库的内模式

- 依赖于它的全局逻辑结构
- 独立于数据库的用户视图，即外模式
- 独立于具体的存储设备
- 将全局逻辑结构中所定义的数据结构及其联系按照一定的物理存储策略进行组织，以达到较好的时间与空间效率



# 小结（续）

## ❖ 数据库的外模式

- 面向具体的应用程序
- 定义在逻辑模式之上
- 独立于存储模式和存储设备
- 当应用需求发生较大变化，相应外模式不能满足其视图要求时，该外模式就得做相应改动
- 设计外模式时应充分考虑到应用的扩充性



# 小结（续）

## ❖ 特定的应用程序

- 在外模式描述的数据结构上编制的
- 依赖于特定的外模式
- 与数据库的模式和存储结构独立
- 不同的应用程序有时可以共用同一个外模式



# 小结（续）

## ❖ 数据库的二级映像

- 保证了数据库外模式的稳定性

- 从底层保证了应用程序的稳定性，除非应用需求本身发生变化，否则应用程序一般不需要修改

## ❖ 数据与程序之间的独立性，使得数据的定义和描述可以从应用程序中分离出去

## ❖ 数据的存取由数据库管理系统管理

- 简化了应用程序的编制

- 大大减少了应用程序的开发和维护成本



# 第1章 绪论

1.1 数据库系统概述

1.2 数据模型

1.3 数据库系统的三级模式结构

1.4 数据库系统的组成

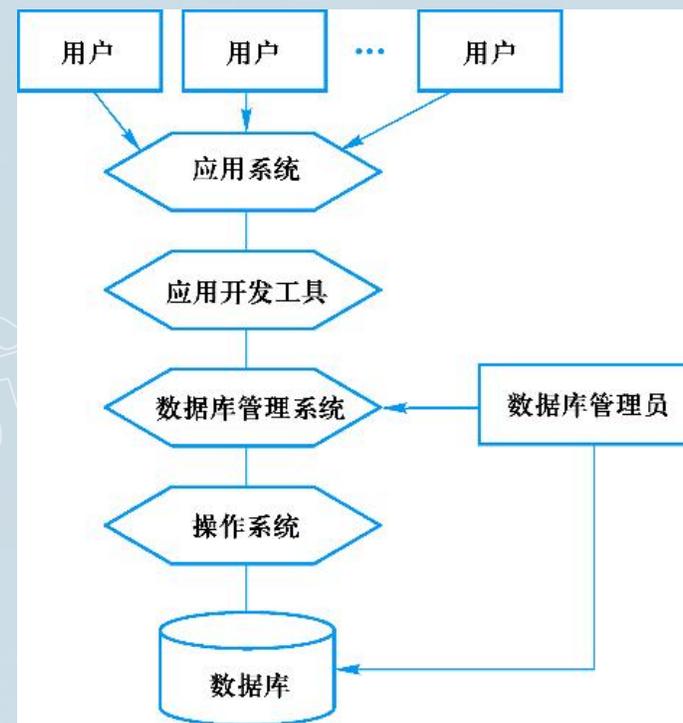
\* 1.5 数据库系统的体系结构

本章小结



# 1.4 数据库系统的组成

- ❖ 数据库
- ❖ 数据库管理系统（及其应用开发工具）
- ❖ 应用系统
- ❖ 数据库管理员



# 数据库系统的组成（续）

❖ 1. 硬件平台

❖ 2. 软件平台

❖ 3. 人员



# 1. 硬件平台

- 数据库管理系统建立在计算机硬件平台和操作系统之上，数据库存放在计算机存储设备中。
- 硬件平台中**存储器**与**处理器技术**的升级推动了数据库技术从磁盘数据库到内存数据库的技术升级
- **海量存储设备**与**高速处理器**的硬件特性成为新型数据库存储引擎和查询处理引擎设计的重要因素



## 2. 软件平台

- 支持数据库管理系统运行的操作系统
- 数据库管理系统
- 开发应用系统的高级语言及其编译系统
- 应用开发工具
- 为特定应用背景开发的数据库应用系统



# 3. 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
- 最终用户



### 3.人员（续）

- 不同的人员各司其职，涉及不同数据抽象级别，具有不同的数据视图

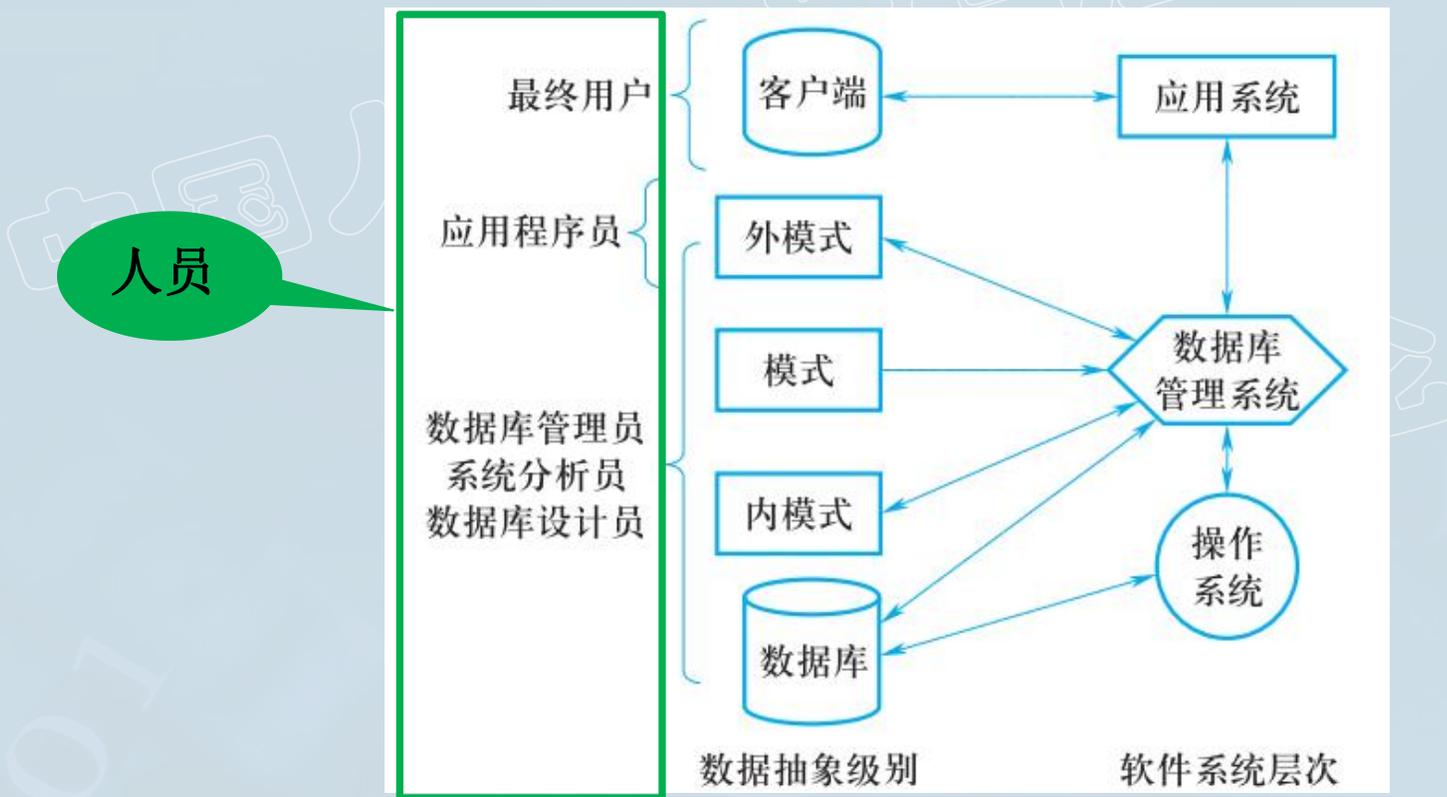


图1.16 数据库系统各种人员的数据视图



# (1) 数据库管理员 (DBA)

主要职责:

## ■①设计与定义数据库

- DBA必须参与数据库设计的全过程
- 与用户、应用开发人员、系统分析员密切结合
- 设计概念模式、数据库模式以及各个应用的外模式
- 熟悉DBMS产品，决定数据库的存储结构和存取策略，设计数据库的内模式

## ■②帮助最终用户使用数据库系统



# 数据库管理员（续）

## 主要职责：

### ■③负责数据库系统的运维工作

- **DBA**负责监视数据库系统的运行情况
- 及时处理运行过程中出现的问题
- 控制不同用户访问数据库的权限
- 收集数据库的审计信息，保证数据库的安全性和完整性



# 数据库管理员（续）

## 主要职责：

- ④改进和重组数据库系统，调优数据库系统的性能
  - **DBA**负责监视、分析数据库系统的性能，包括空间利用率和处理效率。根据实际应用环境不断改进数据库设计
  - 数据库运行过程中不断地插入、删除、修改数据，**DBA**要定期地或按一定的策略对数据库进行重组



# 数据库管理员（续）

## 主要职责：

### ■ ⑤ 转储与恢复数据库

- 为减少硬件、软件或人为故障对数据库系统的破坏，**DBA**必须定义和实施适当的后援和恢复策略
- 一旦系统故障，**DBA**必须能够在最短时间内把数据库恢复到某一正确状态

### ■ ⑥ 重构数据库

- 用户应用需求改变时，**DBA**需要重新构造数据库，包括修改内模式或模式



## (2) 系统分析员和数据库设计人员

- 负责应用系统的需求分析与规范说明，进行总体设计
- 与用户及DBA结合，进行数据库各级模式的设计
- 确定系统的硬软件配置



## (3) 应用程序员

- ❖ 以外模式为基础开发应用系统，编制具体的应用程序
- ❖ 不必考虑数据的存储细节



## (4) 最终用户

- ❖ 操作应用系统，通过应用系统客户端的用户界面使用数据库，完成其业务活动



# 第1章 绪论

1.1 数据库系统概述

1.2 数据模型

1.3 数据库系统的三级模式结构

1.4 数据库系统的组成

\*1.5 数据库系统的体系结构

本章小结



# 1.5 数据库系统的体系结构

## ❖ 数据库系统

- 集中式数据库系统
- 客户-服务器数据库系统
- 并行数据库系统
- 分布式数据库系统
- 云数据库系统



# (1) 集中式数据库系统

- ❖ 数据库管理系统、数据库和应用程序都在一台计算机上
- ❖ 小型机和大型机的集中式数据库系统一般是多用户系统
  - 多个用户通过各自的终端运行不同的应用系统
  - 共享数据库
- ❖ 微型计算机上的数据库系统一般是单用户的



## (2) 客户-服务器数据库系统

- ❖ 数据库管理系统、数据库驻留在服务器上
- ❖ 应用程序放置在客户机上（微型计算机或工作站）
- ❖ 客户机和服务器通过网络进行通信
  - 客户机负责业务数据处理流程和应用程序的界面
  - 要存取数据库中的数据时就向服务器发出请求
  - 服务器接收客户机的请求后进行处理
  - 将客户要求的数据返回给客户机



# 客户-服务器数据库系统（续）

- ❖ 客户-服务器两层结构发展为三层或多层结构
- ❖ 三层结构：浏览器/应用服务器/数据库服务器结构
- ❖ 用户界面采用统一的浏览器方式
- ❖ 应用服务器上安装应用系统或应用模块
- ❖ 数据库服务器上安装数据库管理系统和数据库
- ❖ 两层或三层结构优点：
  - 数据库管理系统的功能进行合理分配
  - 减轻数据库服务器的负担
  - 服务器有更多能力完成事务处理和数据访问控制
  - 支持更多的用户，提高系统的性能



## (3) 并行数据库系统

❖ 在并行计算机机上运行的具有并行处理能力的数据库系统

❖ 并行计算机系统有：

■ 共享内存型

■ 共享磁盘型

■ 非共享型

■ 混合型等



# 并行数据库系统（续）

## ❖ 并行数据库系统优势

- 发挥多处理机优势，采用并行查询处理技术
- 采用并行数据分布与管理技术
- 具有高性能、高可用性、高扩展性等优点



## (4) 分布式数据库系统

- ❖ 数据库中的数据在逻辑上是一个整体
- ❖ 物理地分布在计算机网络的各个不同结点上
- ❖ 网络中的每个结点独立处理本地数据库中的数据（称为场地自治），执行局部应用
- ❖ 也可以执行全局应用，即通过网络通信系统同时存取和处理多个结点上数据库的数据



# 分布式数据库系统（续）

## ❖ 分布式数据库系统优点

- 适应企业部门分布的组织结构
- 降低费用
- 提高系统的可靠性和可用性
- 具有良好可扩展性



## (5) 云数据库系统

### ❖ 云数据库系统产生背景:

- 大数据应用飞速发展，数据量激增，并发用户数在峰值和低谷时差距明显
- 按峰值配置设备，平时造成很大的浪费
- 按低谷值配置设备，又会无法应对峰值

### ❖ 云数据库系统

- 把数据库部署或虚拟化在云计算环境下
- 通过计算机网络以服务的形式提供数据库的功能
  - 包括数据存储、数据更新、查询处理、事务管理等



# 云数据库系统（续）

## ❖ 云数据库系统优点

- 现在的云数据库是运行在机群上的并行数据库系统，能够较好地进行动态伸缩、按需分配计算资源和存储资源。

## ❖ 云数据库系统缺点

- 云数据库存储的安全可信、隐私保护等问题亟待研究解决



# 第1章 绪论

**1.1 数据库系统概述**

**1.2 数据模型**

**1.3 数据库系统的结构**

**1.4 数据库系统的组成**

**\*1.5 数据库系统的体系结构**

**本章小结**



# 本章小结

## ❖ 数据库系统概述

- 数据库的基本概念
- 数据库技术产生和发展背景
- 数据库系统的优点

## ❖ 数据模型

- 数据模型的三个要素
- 三种数据模型



# 本章小结（续）

## ❖ 数据库系统内部的体系结构

- 数据库系统三级模式结构
- 数据库的两级映像

## ❖ 数据库系统的组成

## ❖ 数据库系统的体系结构

- 集中式数据库系统
- 客户-服务器数据库系统
- 并行数据库系统
- 分布式数据库系统
- 云数据库系统

