

# 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

# 1.1 数据库系统概述

## 1.1.1 数据库的地位

数据库系统是信息系统的核心和基础

## 1.1.2 四个基本概念

数据 数据库 DBMS DBS

## 1.1.3 数据管理技术的产生与发展

人工管理 - 文件系统 - 数据库系统

习题：

3. 试述文件系统与数据库系统的区别和联系。
4. 举出适合用文件系统而不是数据库系统的应用例子；再举出适合用数据库系统的应用例子。

补充习题      第一章 绪论补充习题.docx

问答题

其他类型题各选 5 道

# 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 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 常用数据模型

## 1.2.5 层次模型

## 1.2.6 网状模型

## 1.2.7 关系模型

## 1.2.1 两大类数据模型

---

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** 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。

# 数据模型（续）

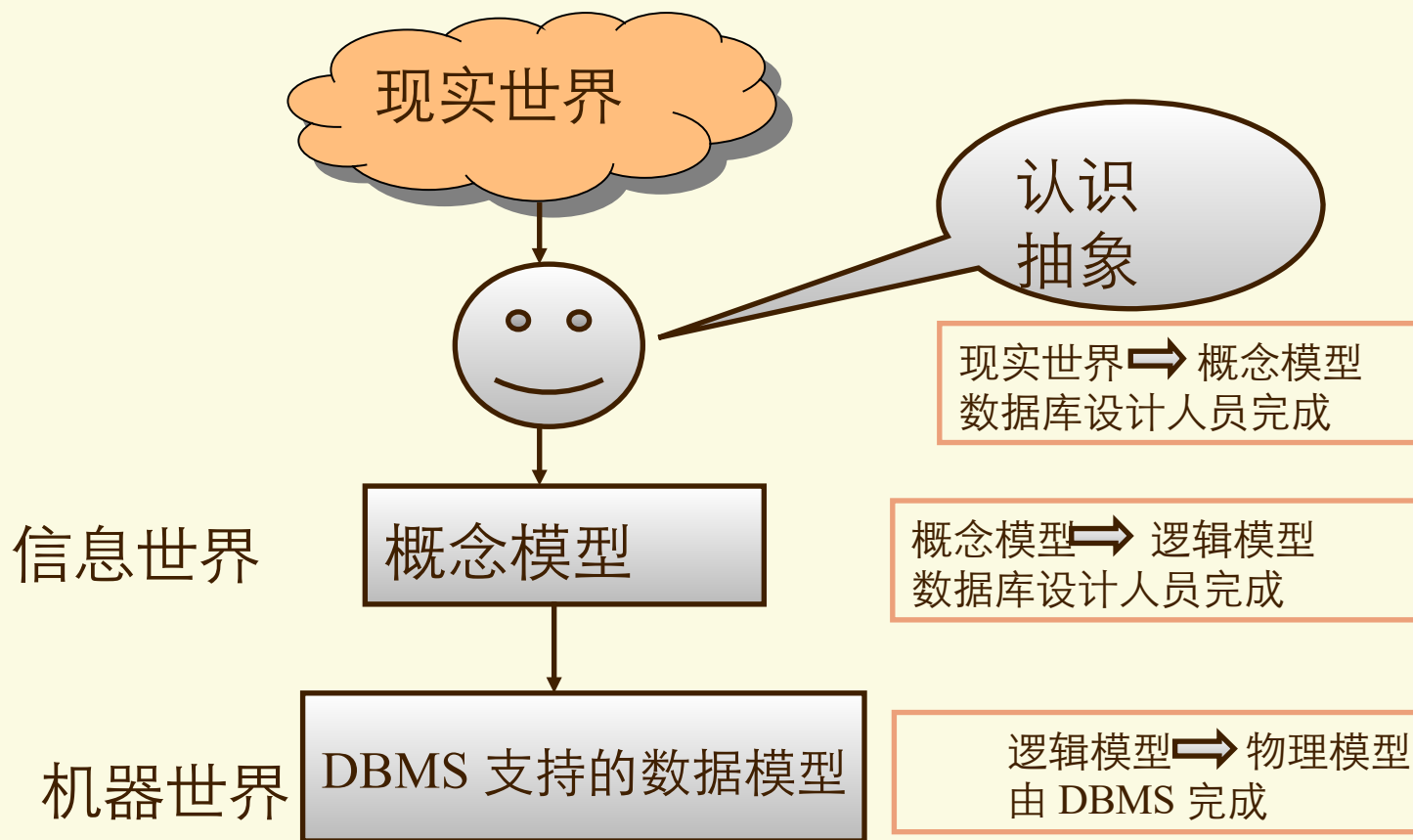
---

## ∞ 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。

# 客观对象的抽象过程 - 两步抽象



现实世界中客观对象的抽象过程

An Introduction to Database Systems





# 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 常用数据模型

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

# 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

## ∞ 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

## 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。
- 完整性规则是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

- 数据模型遵守的基本的通用的完整性约束条件

例如关系模型必须满足实体完整性和参照完整性

- 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

# 概念模型

## 5.0 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

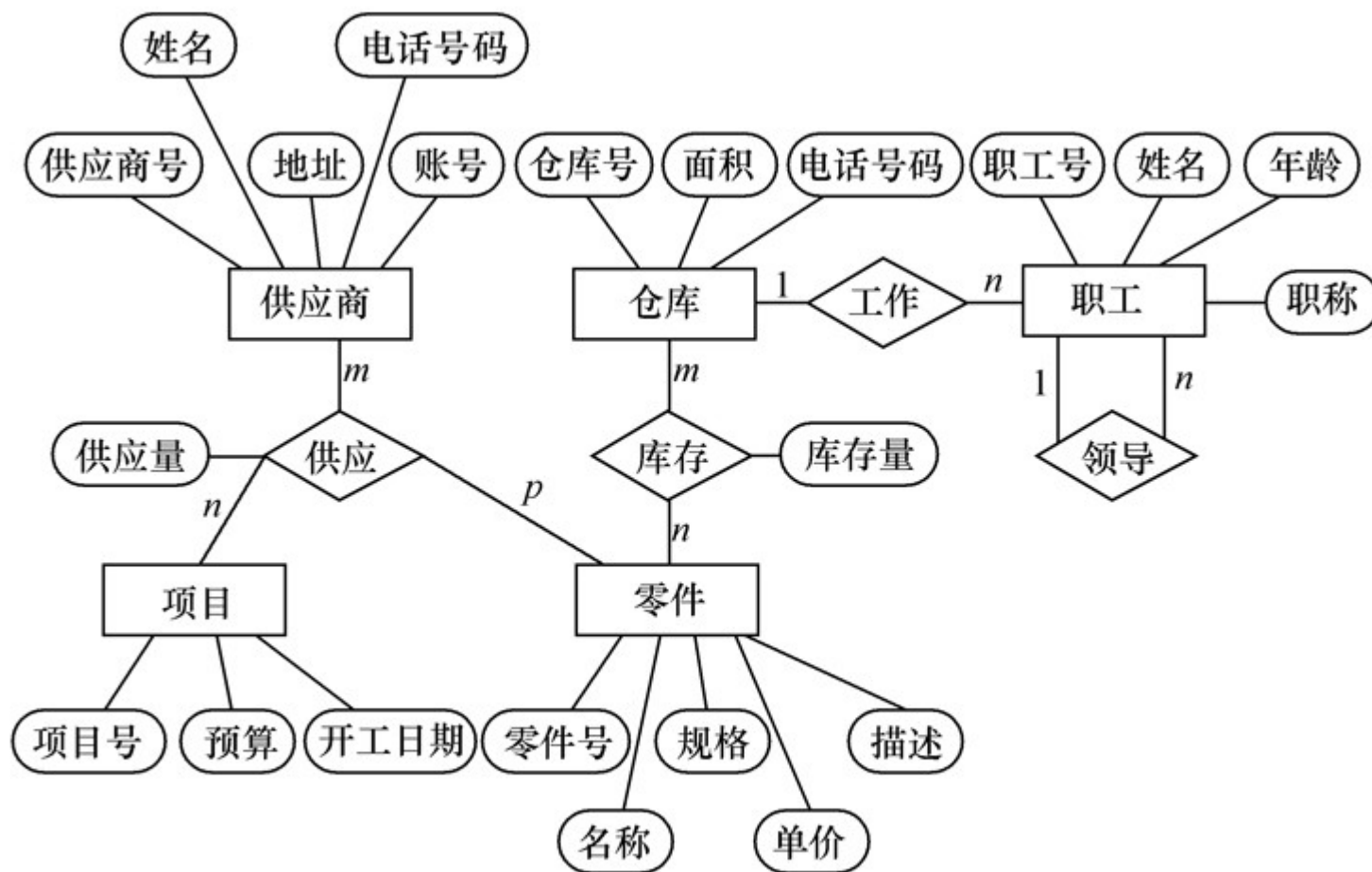
## 5.0 对概念模型的基本要求

- 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



# 实例：工厂物质管理 E-R

模型



(c) 完整的实体-联系图

# 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

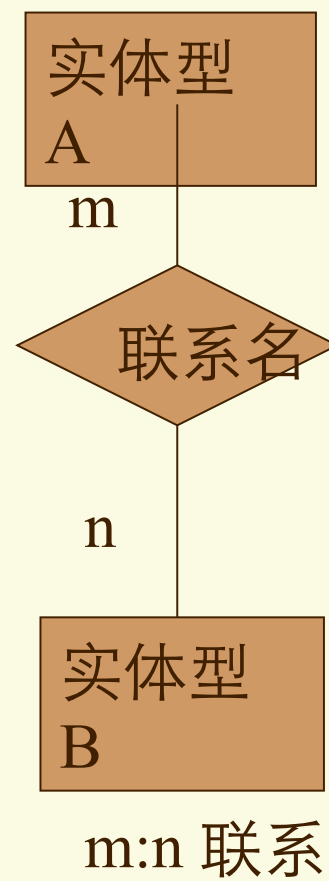
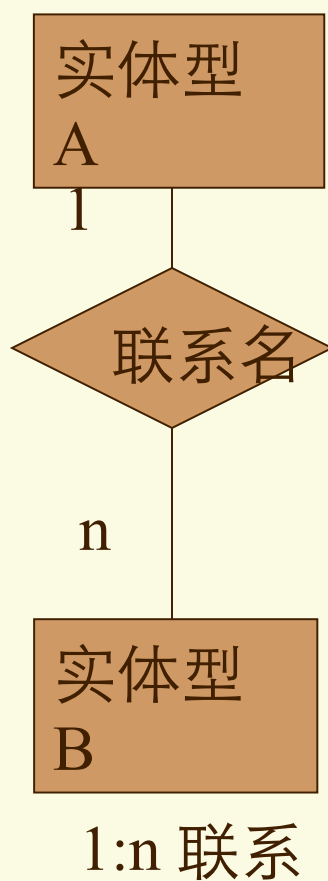
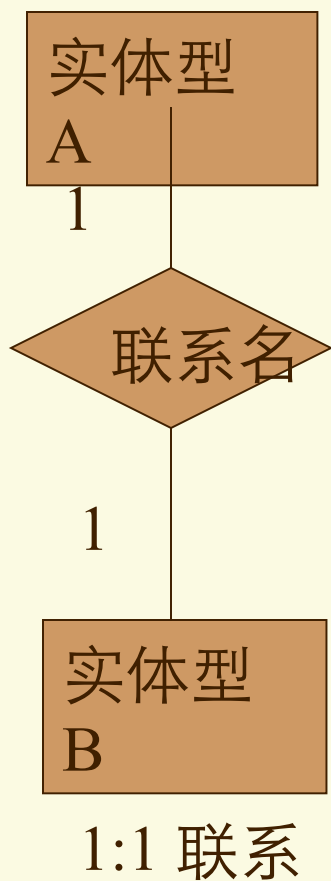
# 信息世界中的基本概念（续）

---

## (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、两个实体型之间的联系



# 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 最常用数据模型

---

## ∞ 非关系模型（格式化模型）

- 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)

## ∞ XML 数据模型 (XML Data Model)

## ∞ 等等

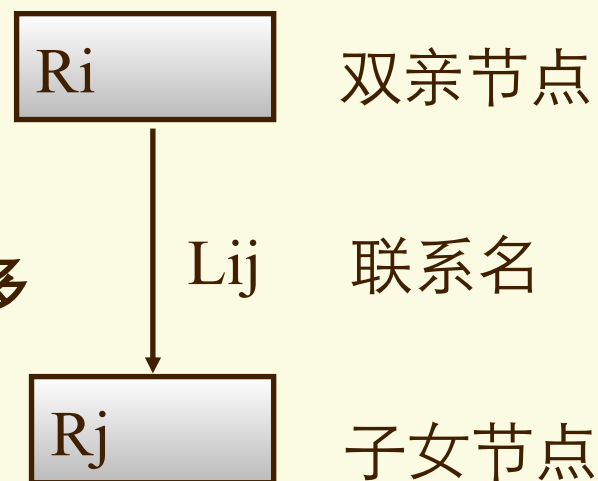
# 非关系模型（格式化模型）

- 实体—记录表示
- 实体的属性—记录的数据项（或字段）
- 实体间联系—记录之间的两两联系

数据结构：基本层次联系为单位

基本层次联系：

两个记录以及它们之间的一对多  
（包括一对一）的联系



基本层次关系



# 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 常用数据模型

**1.2.5 层次模型**

1.2.6 网状模型

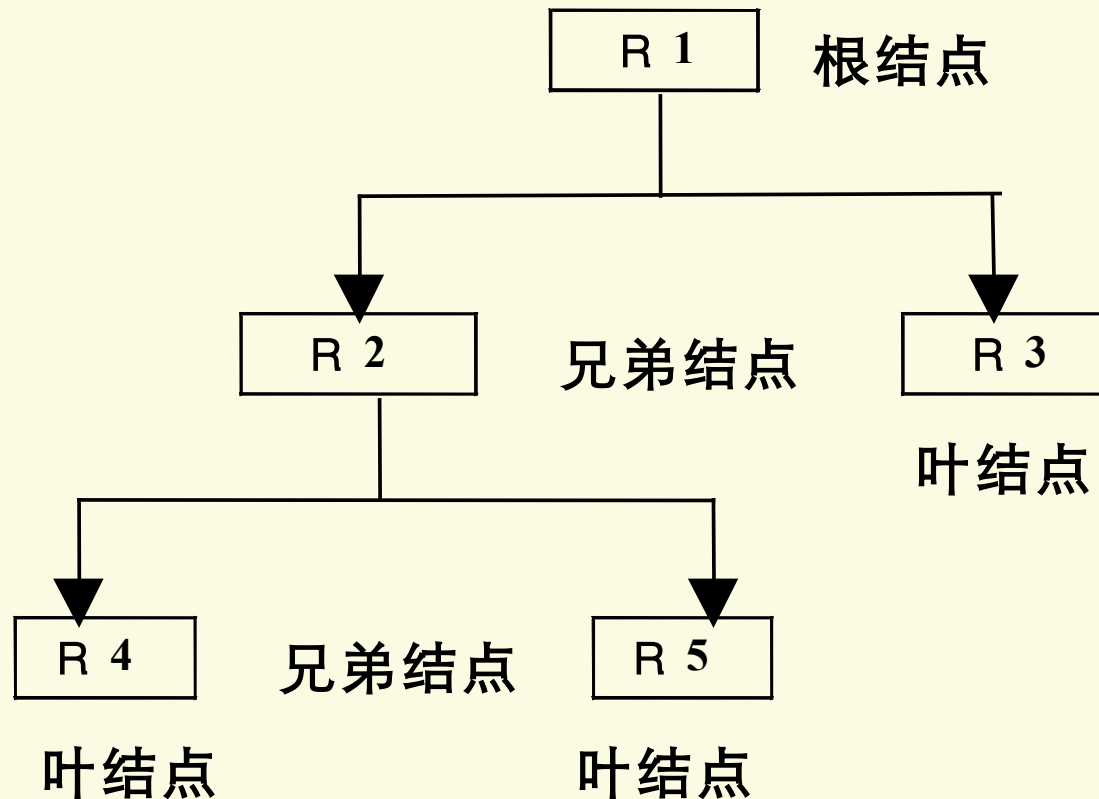
1.2.7 关系模型

## 1.2.4 层次模型

---

1. 层次数据模型的数据结构
2. 层次数据模型的数据操纵
3. 层次数据模型的完整性约束
4. 层次数据模型的存储结构
5. 层次数据模型的优缺点
6. 典型的层次数据库系统

# 1. 层次数据模型的数据结构 - 树



# 1. 层次数据模型的数据结构

## 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

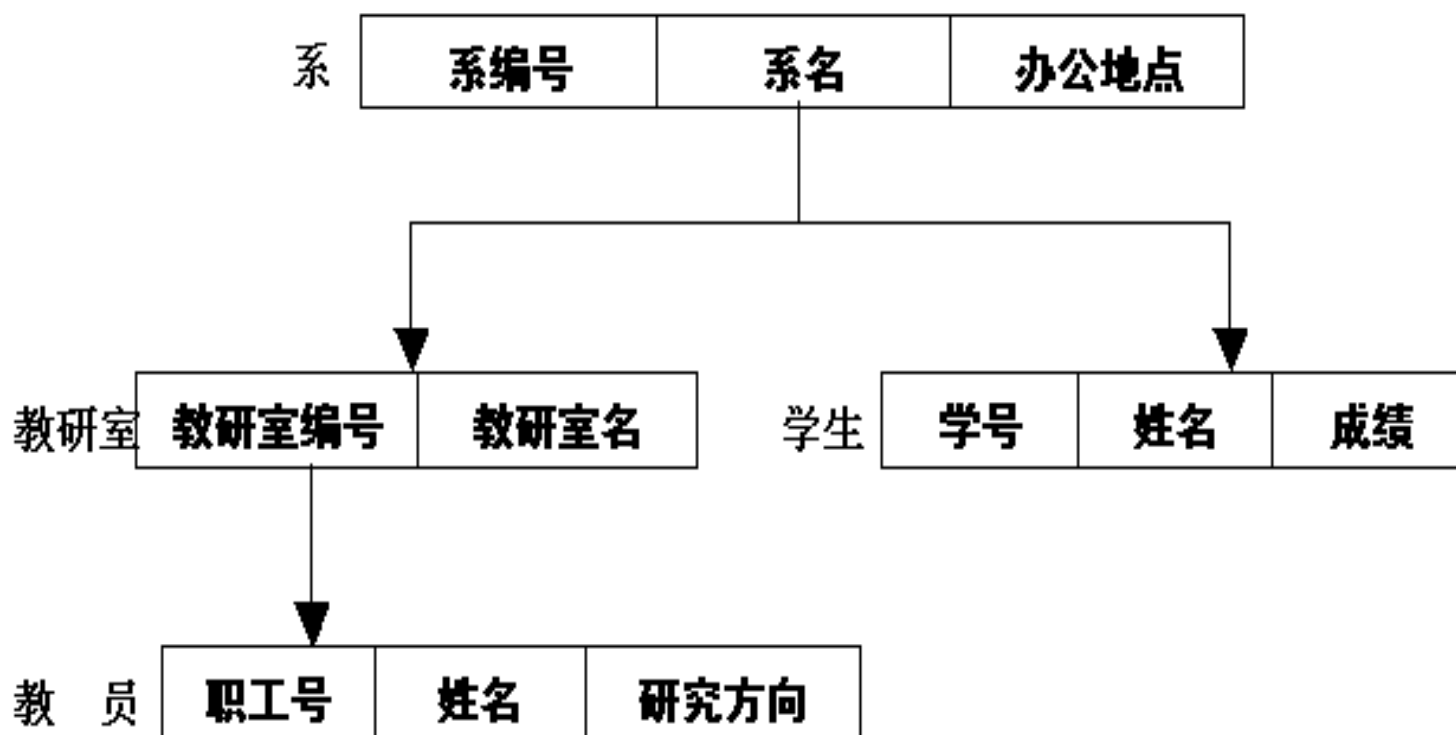
# 层次数据模型的数据结构

## (续)

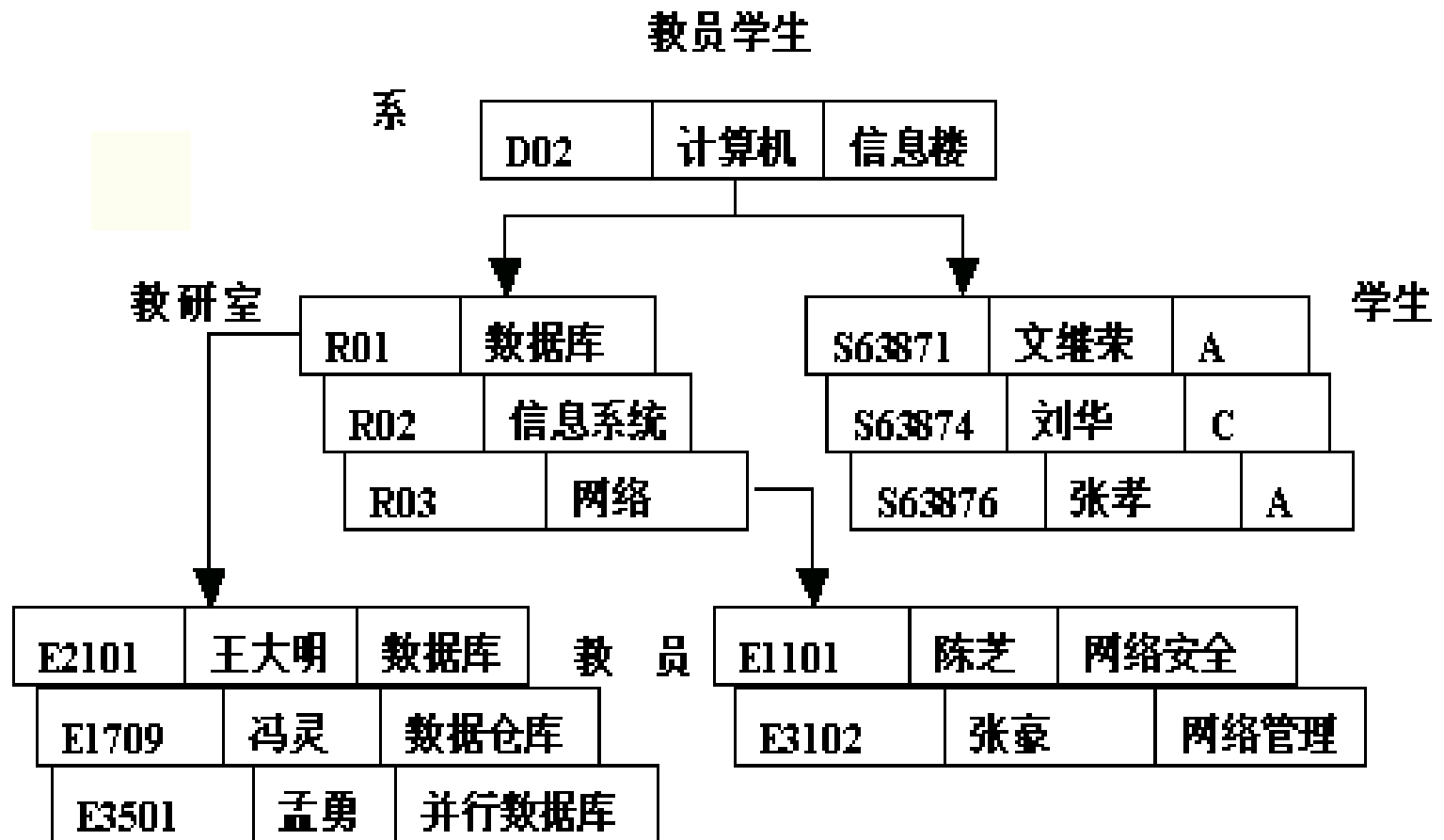
### 表示方法

- **记录类型**：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。
- 每个记录类型可以定义一个排序字段，也称为码字段

# 教员学生数据库模型



# 教员学生数据库的一个值



# 层次数据模型的数据结构 (续)

## 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
- 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



## 2. 层次模型的数据操纵

---

∞ 查询

∞ 插入

∞ 删除

∞ 更新

### 3. 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

## 4. 层次数据模型的存储结构

---

### 层次数据模型存储结构的特点

- 数据与数据间的联系结合在一起存储
- 数据间的联系用位置相邻表达或用指针表达

### 两种常用的实现方法

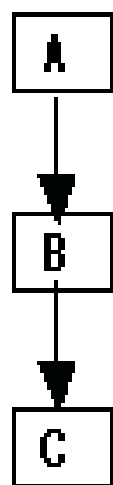
- 邻接法
- 链接法

# 层次数据模型的存储结构 (续)

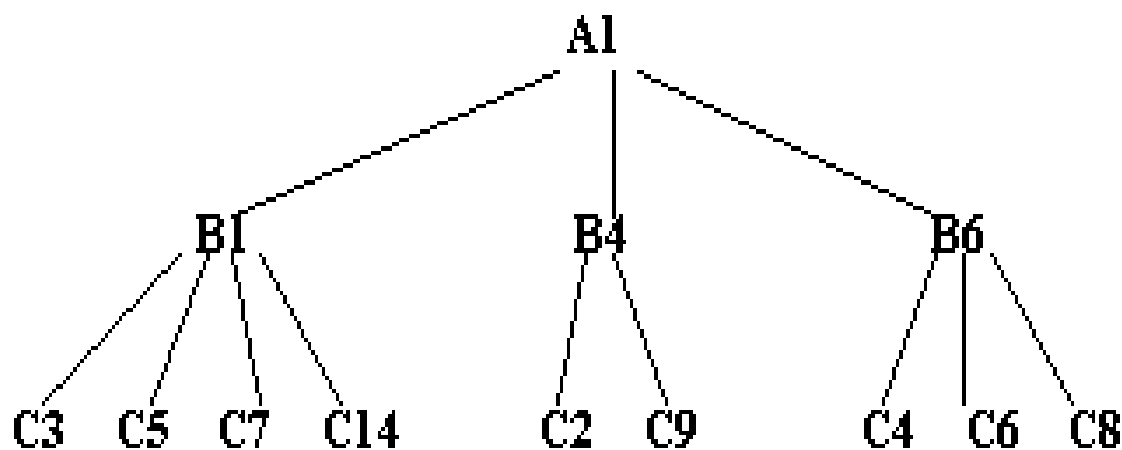
## ∞邻接法

- 按照层次树前序遍历的顺序把所有记录值依次邻接存放，即通过物理空间的位置相邻来实现层次顺序。

# 层次数据模型的存储结构示例



(a)



(b)

图 1.21

# 邻接法示例

A1	B1	C3	C5	C7	C14	B4	C2	C9	B6	C4	C6	C8	A2	...
----	----	----	----	----	-----	----	----	----	----	----	----	----	----	-----

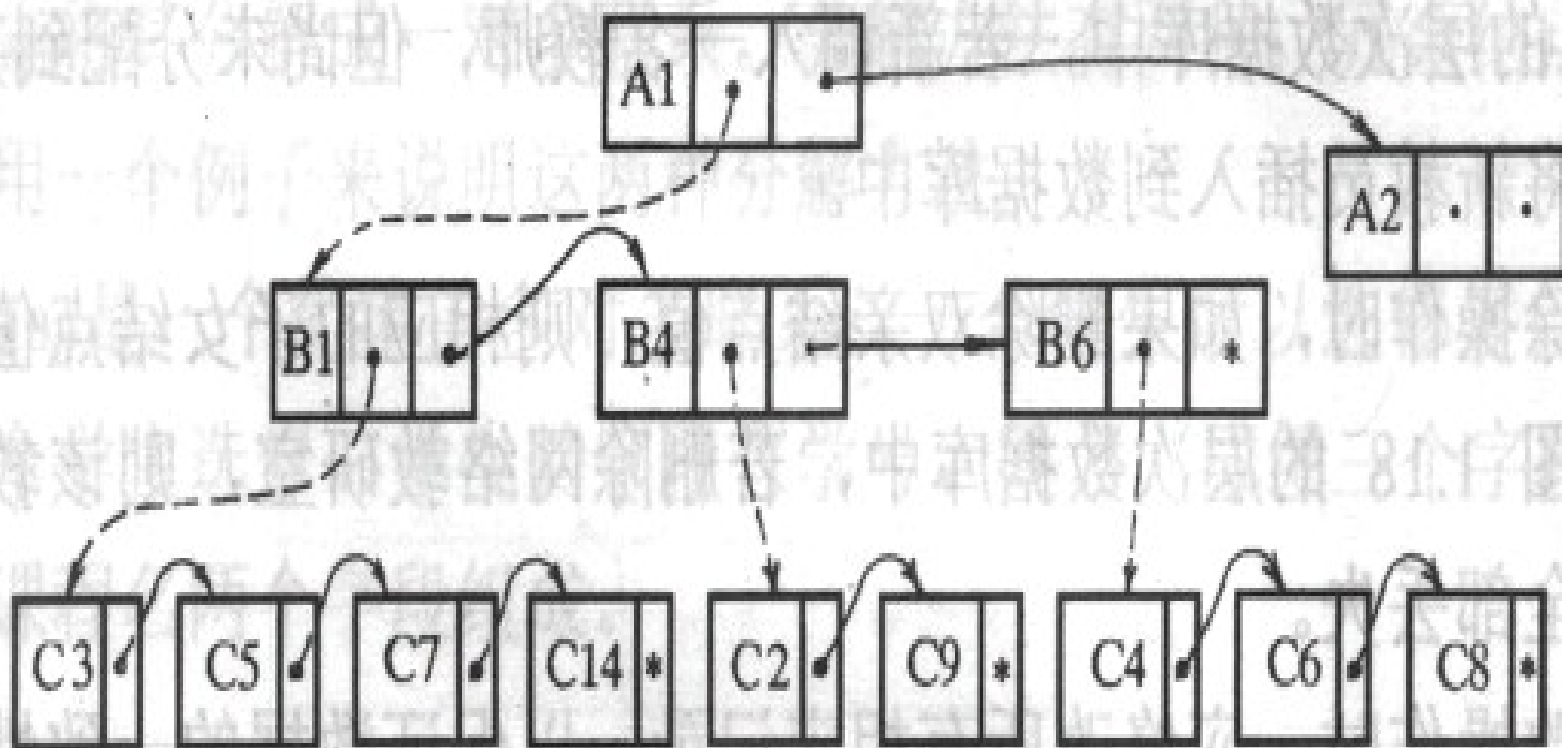
图1.22 邻接法

# 层次数据模型的存储结构 (续)

## ∞ 链接法

- 用指针来反映数据之间的层次联系
  - 子女 - 兄弟链接法
  - 层次序列链接法

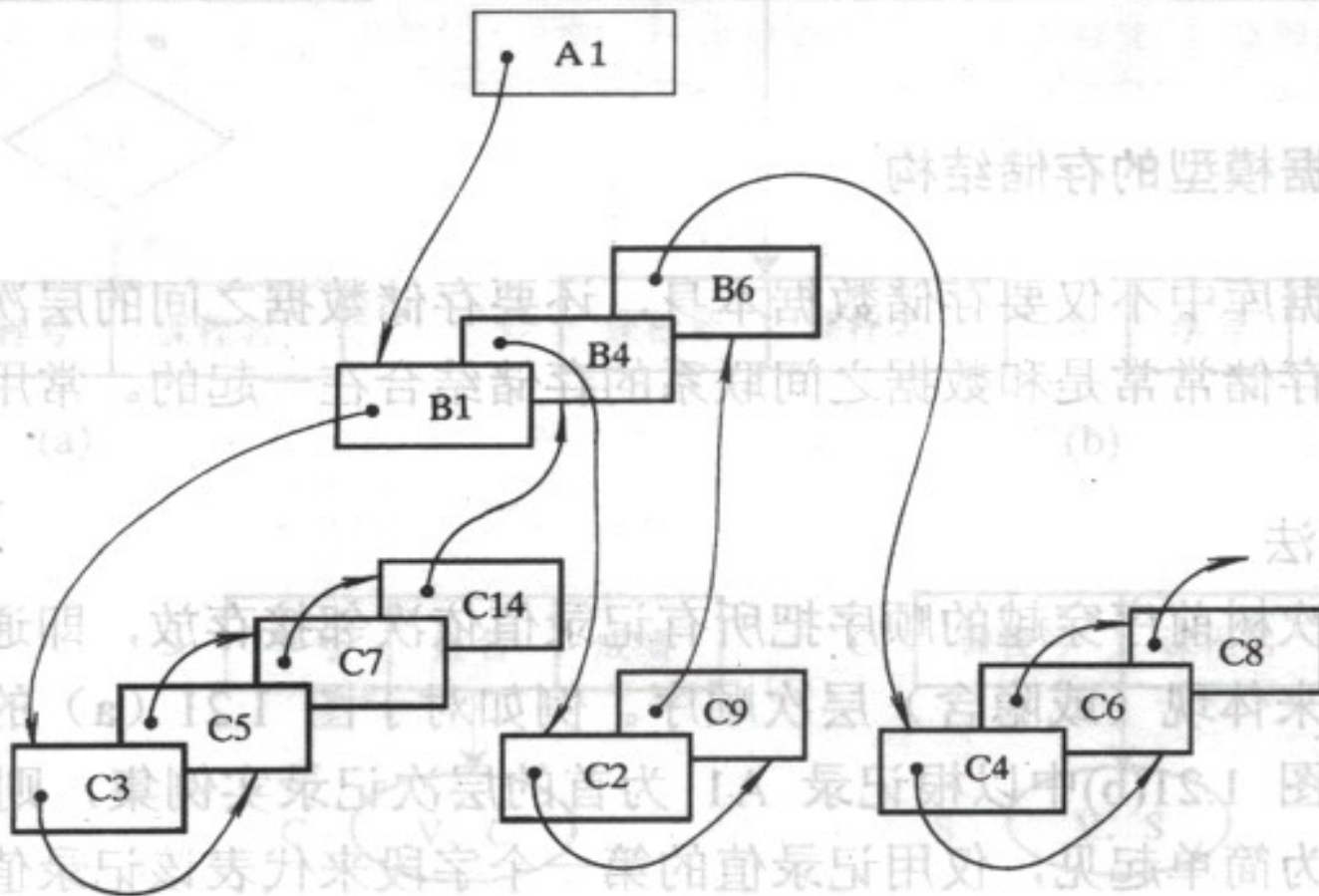
# 子女 - 兄弟链接法示例



(a)



# 层次序列链接法示例



(b)

## 5. 层次模型的优缺点

---

### ☞ 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

# 层次模型的优缺点（续）

---

## ❧ 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

## 6. 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

☞IMS 为保证阿波罗飞船 1969 年顺利登月作出了贡献

# 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 常用数据模型

1.2.5 层次模型

1.2.6 网状模型

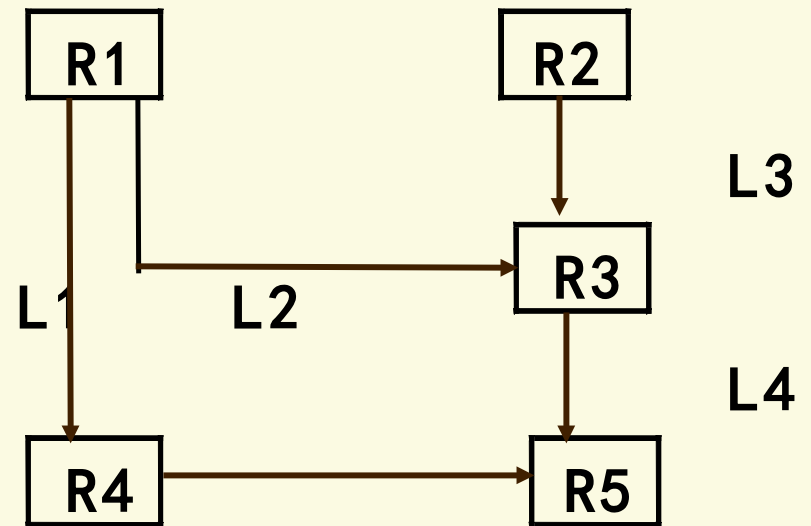
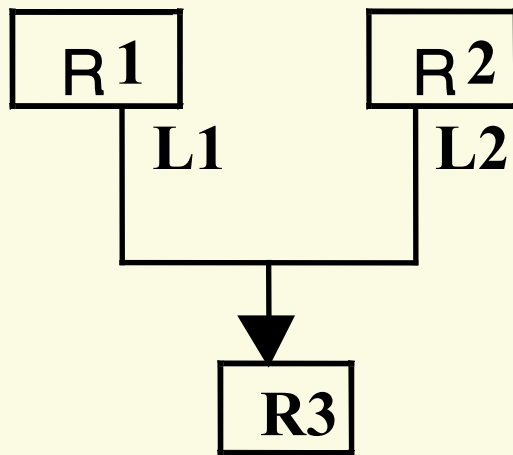
1.2.7 关系模型

## 1.2.6 网状模型

---

1. 网状数据模型的数据结构
2. 网状数据模型的数据操纵
3. 网状数据模型的完整性约束
4. 网状数据模型的存储结构
5. 网状数据模型的优缺点
6. 典型的网状数据库系统

# 1. 网状数据模型的数据结构



# 1. 网状数据模型的数据结构

---

## ∞ 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。



# 网状数据模型的数据结构 (续)

## 表示方法（与层次数据模型相同）

- **记录类型**：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）型之间的一对多的父子联系。
- 每个记录类型定义一个排序字段，也称为码字段

# 网状数据模型的数据结构 (续)

## ∞ 特点

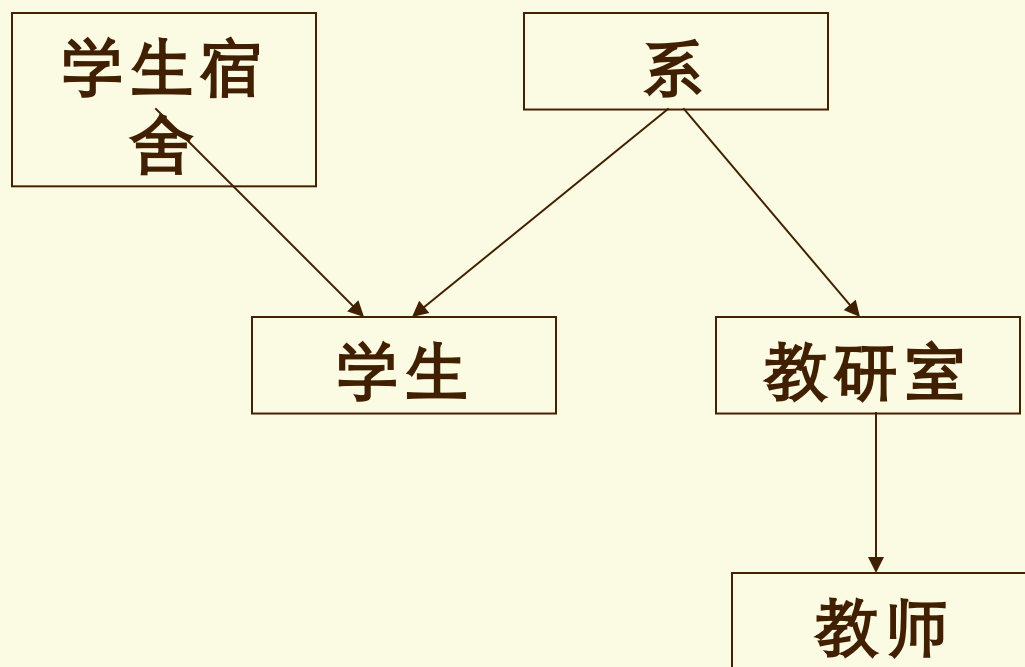
- 只能直接处理一对多的实体联系
- 任何记录值只有按其路径查看时，才能显出它的全部意义

# 网状数据模型的数据结构 (续)

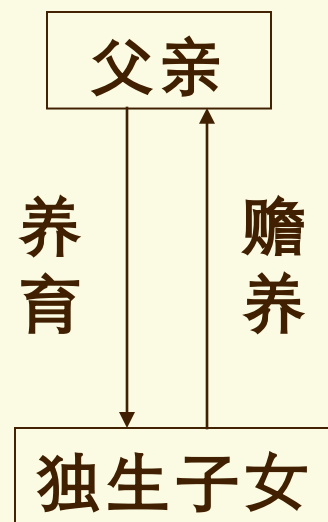
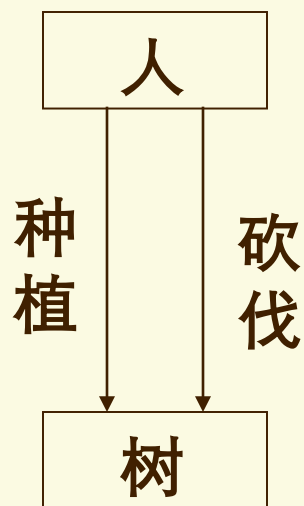
## ∞ 网状模型与层次模型的区别

- 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去描述现实世界
- 层次模型实际上是网状模型的一个特例

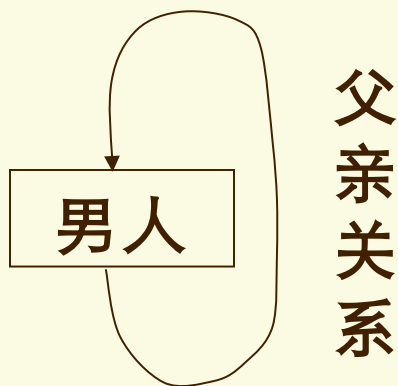
# 网状数据模型的数据结构 (续)



# 网状数据模型的数据结构 (续)



# 网状数据模型的数据结构 (续)



## 2. 网状模型的数据操纵

---

- 查询：导航式的查询语言
- 插入
- 删除
- 更新

### 3. 网状数据模型的完整性约束

---

#### ∞ 完整性约束条件不严格

- 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



# 网状数据模型的完整性约束 (续)

具体的网状数据库系统（如 DBTG）对数据操纵加了一些限制，提供了一定的完整性约束。

- 码

- 属籍类别

- 加入类别（自动的，手工的）
- 移出类别（固定的，必须的，随意的）

## 4. 网状数据模型的存储结构

---

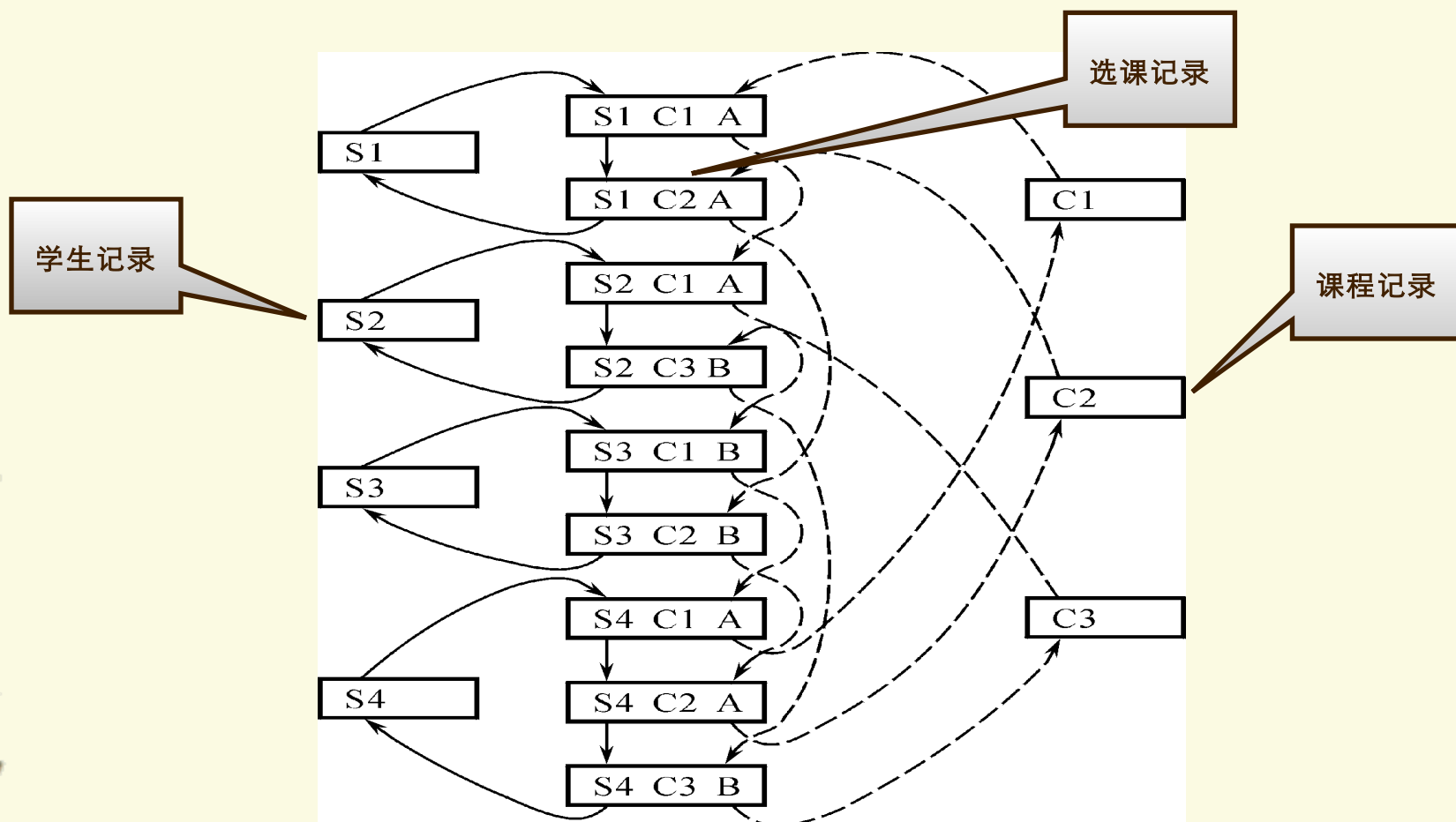
### ∞ 关键

- 实现记录之间的联系

### ∞ 常用方法：用指针实现记录之间的联系

- 单向链接
- 双向链接
- 环状链接
- 向首链接

# 网状数据模型的存储结构示例



## 5. 网状模型的优缺点

---

### ∞ 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

# 网状模型的优缺点（续）

---

## 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

## 6. 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
- 奠定了网状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 数据模型

---

1.2.1 概念模型

1.2.2 数据模型的组成要素

1.2.3 最常用的数据模型

1.2.4 层次模型

1.2.5 网状模型

1.2.6 关系模型

## 1.2.6 关系模型

---

1. 关系数据模型的数据结构
2. 关系数据模型的操纵
3. 关系数据模型的完整性约束
4. 关系数据模型的存储结构
5. 关系数据模型的优缺点
6. 典型的关系数据库系统



# 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. 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

元组

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
...	...	...	...	...	...

# 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。

。

∞ 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）

# 关系数据模型的数据结构

## (续)

### 500 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。

# 关系数据模型的数据结构 (续)

例 1

学生 (学号, 姓名, 年龄, 性别, 系号, 年级)  
系与学生之间的一对多联系隐含在学生关系中。

例 2

系 (系号, 系名, 系主任、办公地点)  
系与系主任间的一对一联系隐含在系关系中。

例 3

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)

学生与课程之间的多对多联系用选修关系表示。

# 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		□ □
			基 本	津 贴	职 务	房 租	水 电	
86051	陈 平	讲 师	1305	1200	50	160	112	2283
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	

# 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）

## 2. 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”



# 3. 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

## 4. 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

# 5. 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

- 实体和各类联系都用关系来表示
- 对数据的检索结果也是关系

数据结构简单清晰，用户易懂易用

— 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格式化模型
- 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

# 6. 典型的关系数据库系统

---

## ☞ 典型实验系统

- System R、University INGRES、。。。
- COBASE、PBASE。。。

## ☞ 典型商用系统

- ORACLE
- DB/2、INFORMIX
- MS SQL Server
- SYBASE
- kingbaseES（金仓数据库）
- DM
- 等等

