



# 数据库技术与应用

---

北京邮电大学计算机学院 肖达

xiaoda99@gmail.com



# 上节课回顾

## ■ 1.0 引言

- 1.0.1 数据库技术的发展历史及趋势
- 1.0.2 当前流行的数据库产品

## ■ 1.1 数据库系统概述

- 1.1.1 四个基本概念
  - Data、DB、DBMS、DBS
- 1.1.2 数据管理技术的产生和发展
  - 人工管理 → 文件系统 → 数据库系统
- 1.1.3 数据库系统的特点
  - 数据结构化、共享性高、独立性高、由DBMS统一管理



# 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

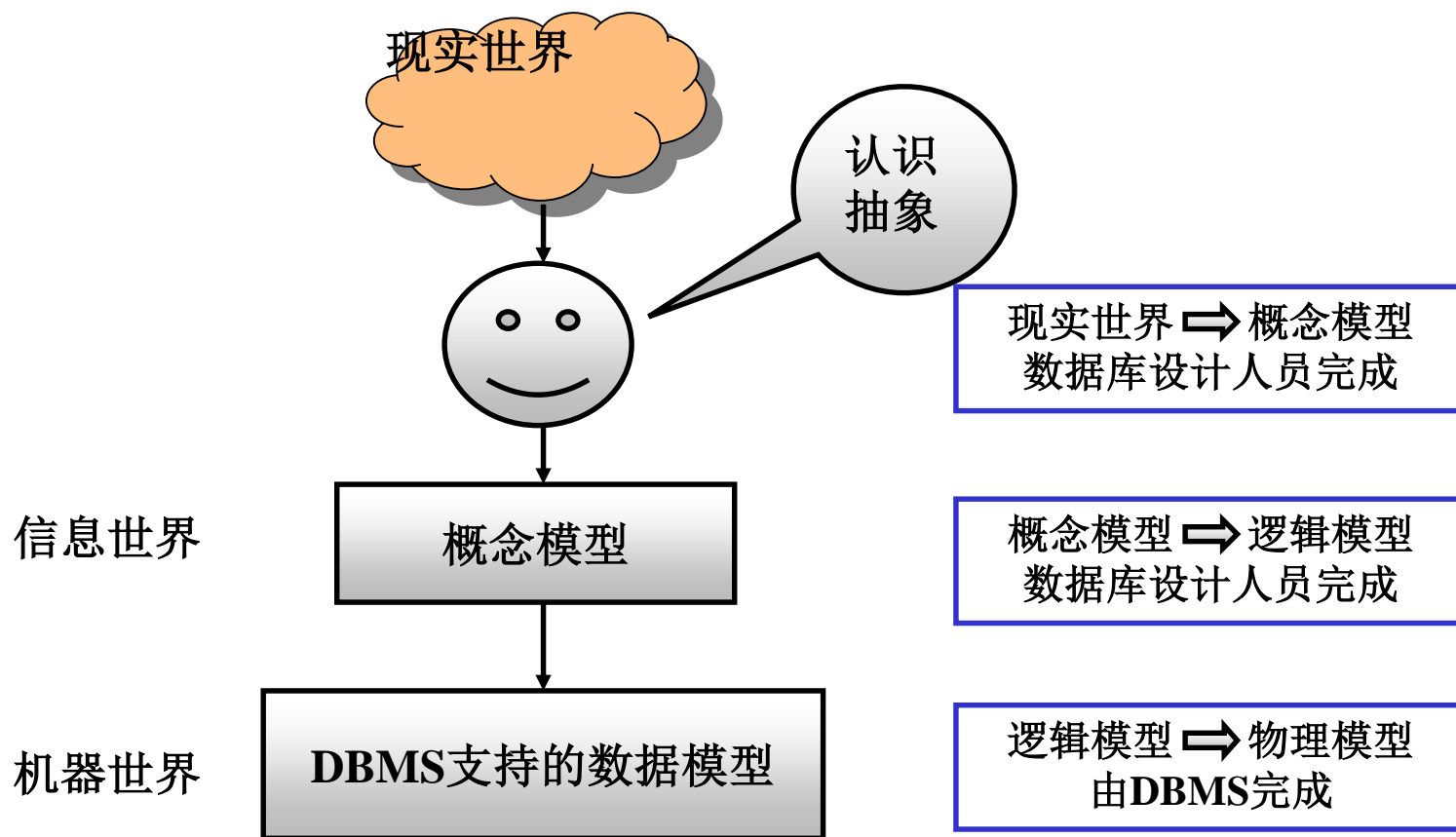
- 在数据库中用数据模型这个工具来抽象、表示和处理现实世界中的事物。
- 通俗地讲数据模型就是现实世界的抽象和模拟。
- 数据模型应满足三方面要求
  - 能比较真实地模拟现实世界
  - 容易为人所理解
  - 便于在计算机上实现



## 1.2.1 两大类数据模型

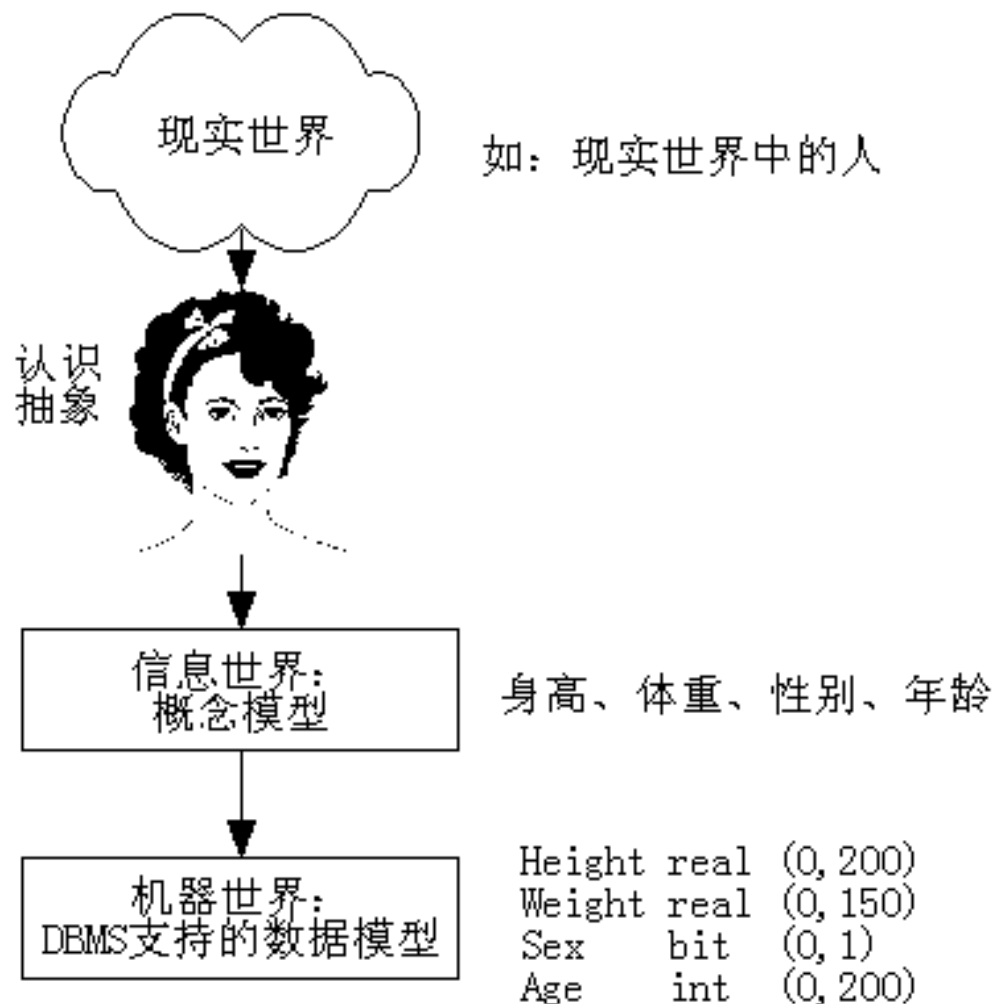
- 数据模型分为两类（分属两个不同的层次）
  - (1) **概念模型** 也称信息模型，它是按用户的观点来对数据和信息建模，用于数据库设计。
  - (2) **逻辑模型和物理模型** 按计算机系统的观点对数据建模，用于DBMS的实现。
    - **逻辑模型**主要包括网状模型、层次模型、关系模型、面向对象模型等，按计算机系统的观点对数据建模，用于DBMS实现。
    - **物理模型**是对数据最底层的抽象，描述数据在系统内部的表示方式和存取方法，在磁盘或磁带上的存储方式和存取方法。

# 两大类数据模型 (续)



现实世界中客观对象的抽象过程

# 两大类数据模型 (续)





# 1.2 数据模型

---

## 1.2.1 两大类数据模型

## 1.2.2 概念模型

## 1.2.3 数据模型的组成要素

## 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





# 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。

## (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

例子：学生（学号、姓名、性别、出生年月）



# 信息世界中的基本概念(续)

## (6) 实体集 (Entity Set)

同一类型实体的集合称为实体集

学号	姓名	性别	出生日期
20021001	张三	男	1978-5-6
20021003	李四	女	1980-1-24
20021004	王五	男	1979-11-12

## (7) 联系 (Relationship)

- 现实世界：事物内部以及事物之间的联系
- → 信息世界：实体内部的联系和实体之间的联系

## 二、两个实体型之间的联系

### ■ 一对一联系（1:1）

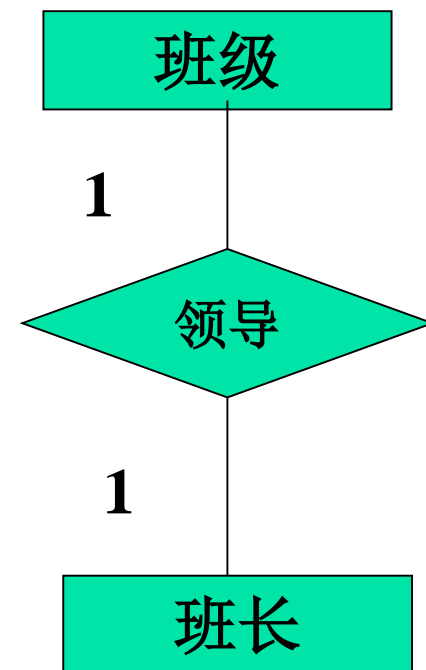
#### ➤ 实例

一个班级只有一个正班长

一个班长只在一个班中任职

#### ➤ 定义：

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（也可以没有）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**



**1:1联系**

# 两个实体型之间的联系 (续)

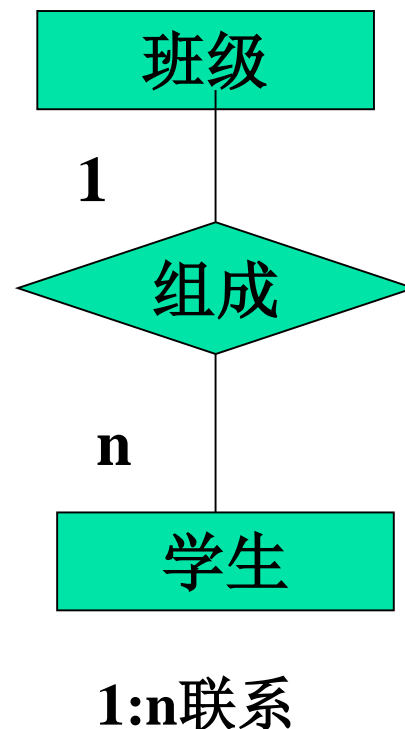
## ■ 一对多联系 (1: n)

### ➤ 实例

一个班级中有若干名学生，  
每个学生只在一个班级中学习

### ➤ 定义：

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**



# 两个实体型之间的联系 (续)

## ■ 多对多联系 (m:n)

### ➤ 实例

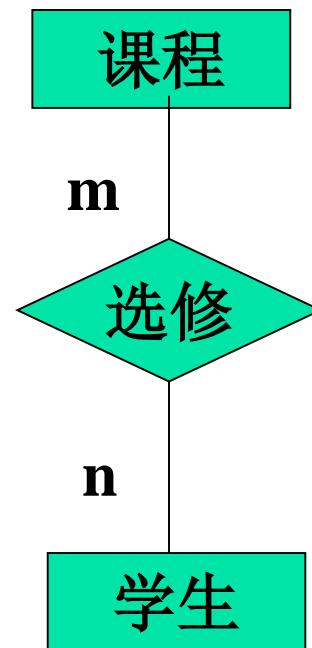
课程与学生之间的联系：

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

### ➤ 定义：

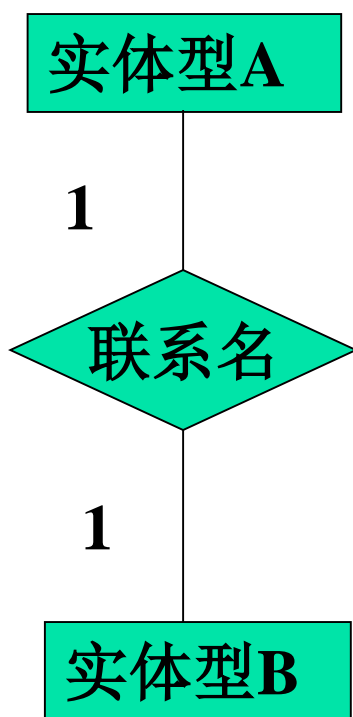
如果对于实体集**A**中的每一个实体，实体集**B**中有**n**个实体 ( $n \geq 0$ ) 与之联系，反之，对于实体集**B**中的每一个实体，实体集**A**中也有**m**个实体 ( $m \geq 0$ ) 与之联系，则称实体集**A**与实体**B**具有多对多联系，记为**m:n**



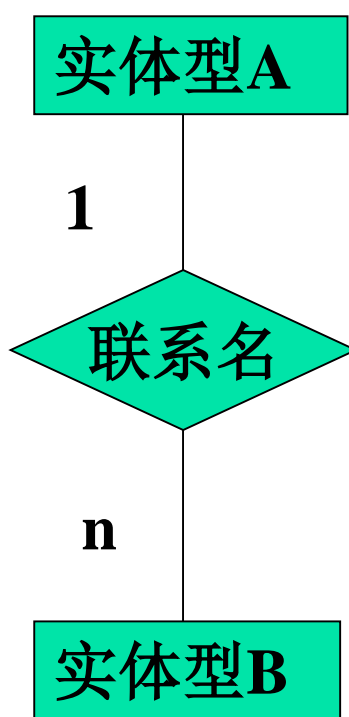
**m:n**联系

## 二、两个实体型之间的联系

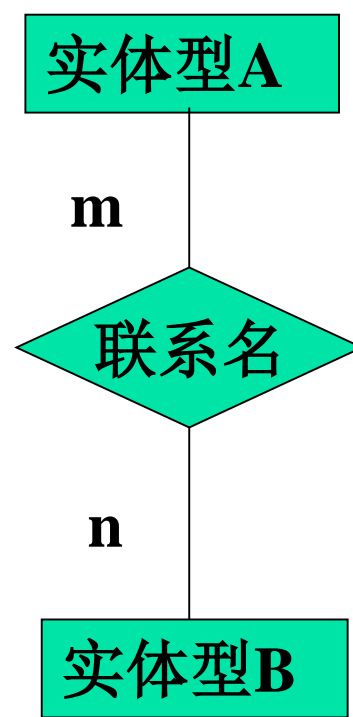
用图形来表示两个实体型之间的三类联系



1:1联系



1:n联系



m:n联系



# 三、两个以上实体型之间的联系

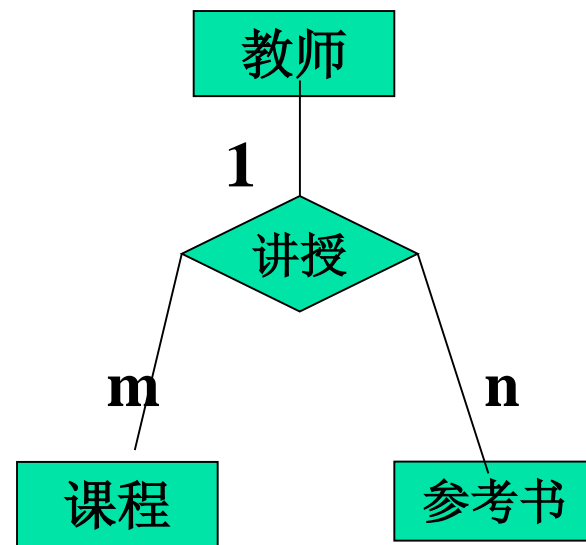
## ■ 两个以上实体型之间一对多联系

- 若实体型 $E_1, E_2, \dots, E_n$ 存在联系, 对于实体型 $E_j$  ( $j=1, 2, \dots, i-1, i+1, \dots, n$ ) 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

每一个教师可以讲授若干门课程, 每门课程可以使用若干本参考书



两个以上实体型间1:n联系



# 两个以上实体型之间的联系(续)

- 两个以上实体型间的一对一联系
- 两个以上实体型间的多对多联系

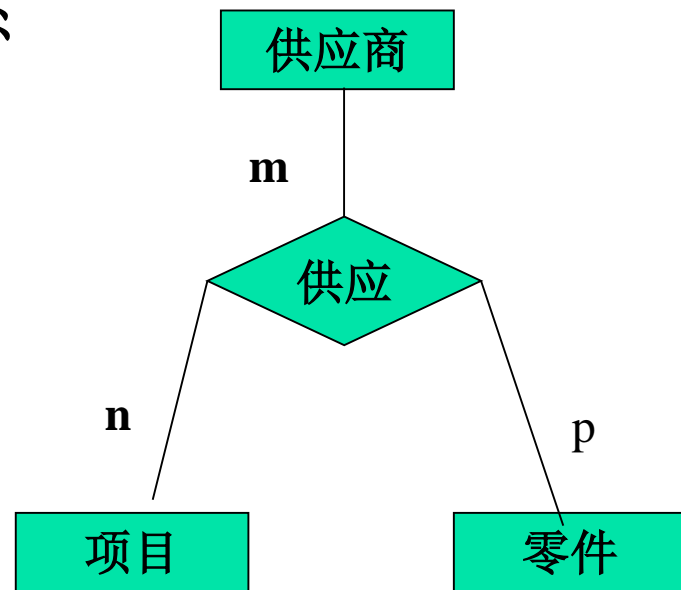
## ➤ 实例

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

## 四、单个实体型内的联系

- 同一实体集内的各实体之间的关系
- 一对多联系

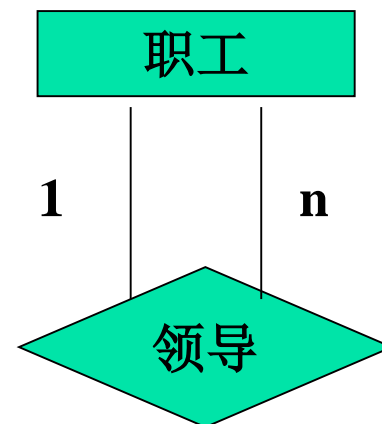
- 实例

职工实体型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



单个实体型内部1:n联系

- 一对一联系和多对多联系

请举例



# 练习

- 下列实体型之间有何种联系？
  - 电影和演员
  - 电影和导演
  - 淘宝网的卖家和物品
  - 淘宝网的卖家、买家和物品
  - QQ的用户

# 五、 概念模型的一种表示方法



## ■ 实体—联系方法(E-R方法)

- Peter Chen于1976年提出
- 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

# E-R图

## ■ 实体型

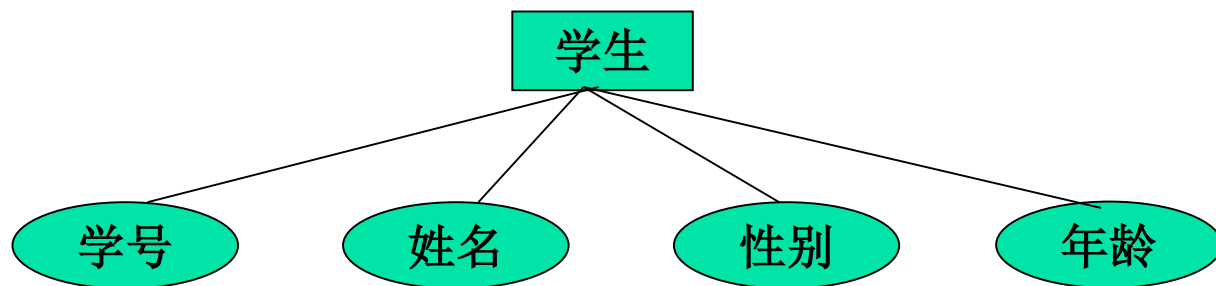
用矩形表示，矩形框内写明实体名。

## ■ 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

学生

教师

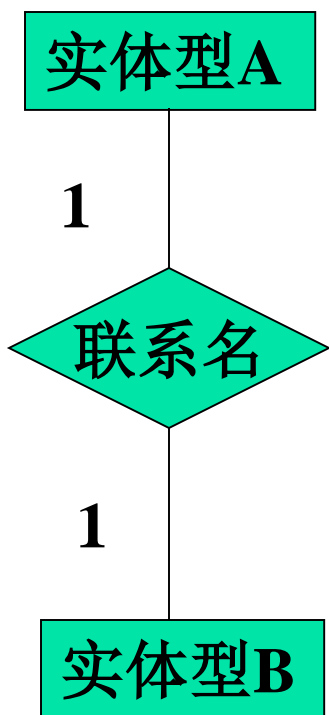


## ■ 联系

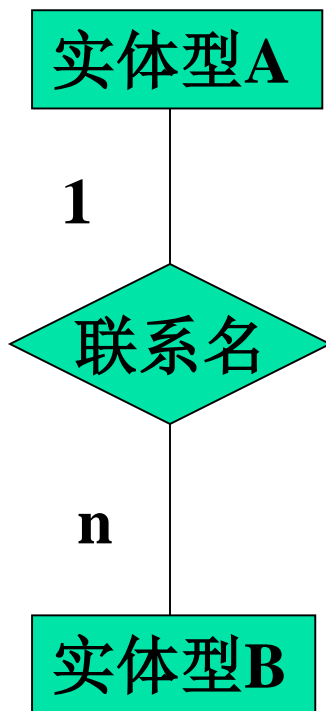
### ➤ 联系本身:

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）

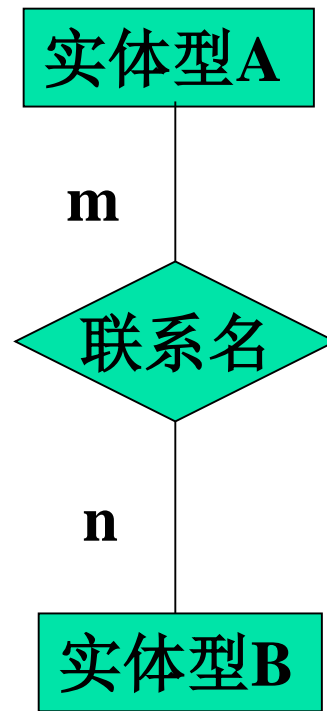
# 联系的表示方法



1:1联系

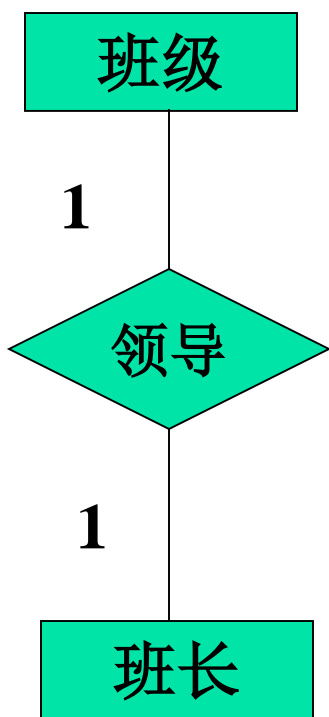


1:n联系

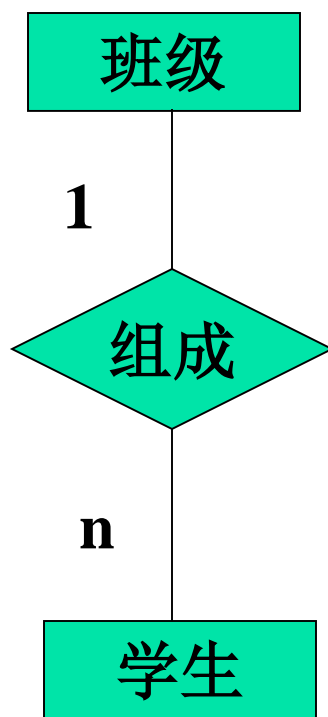


m:n联系

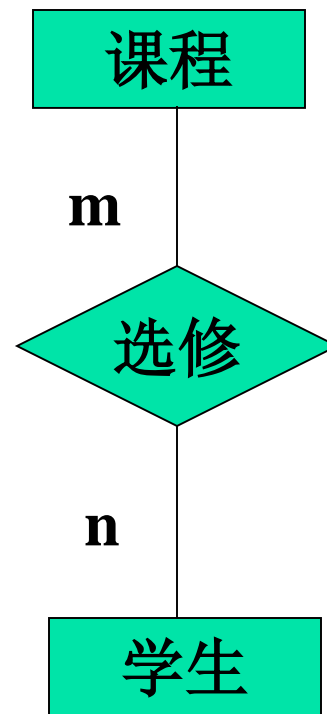
# 联系的表示方法示例



1:1联系



1:n联系

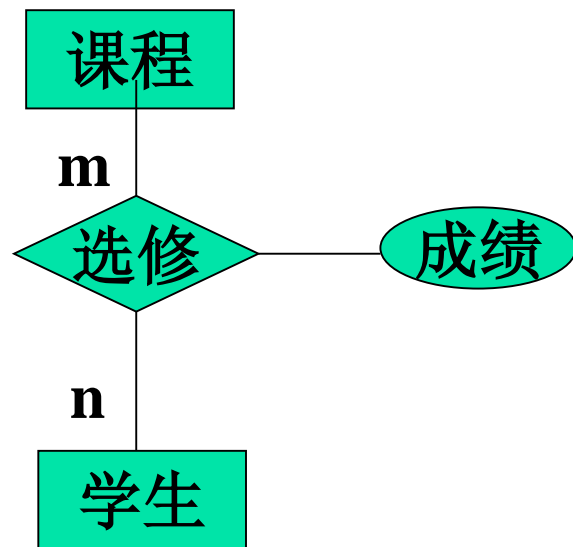


m:n联系

# 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







# 六、一个实例

## 用E-R图表示某个工厂物资管理的概念模型

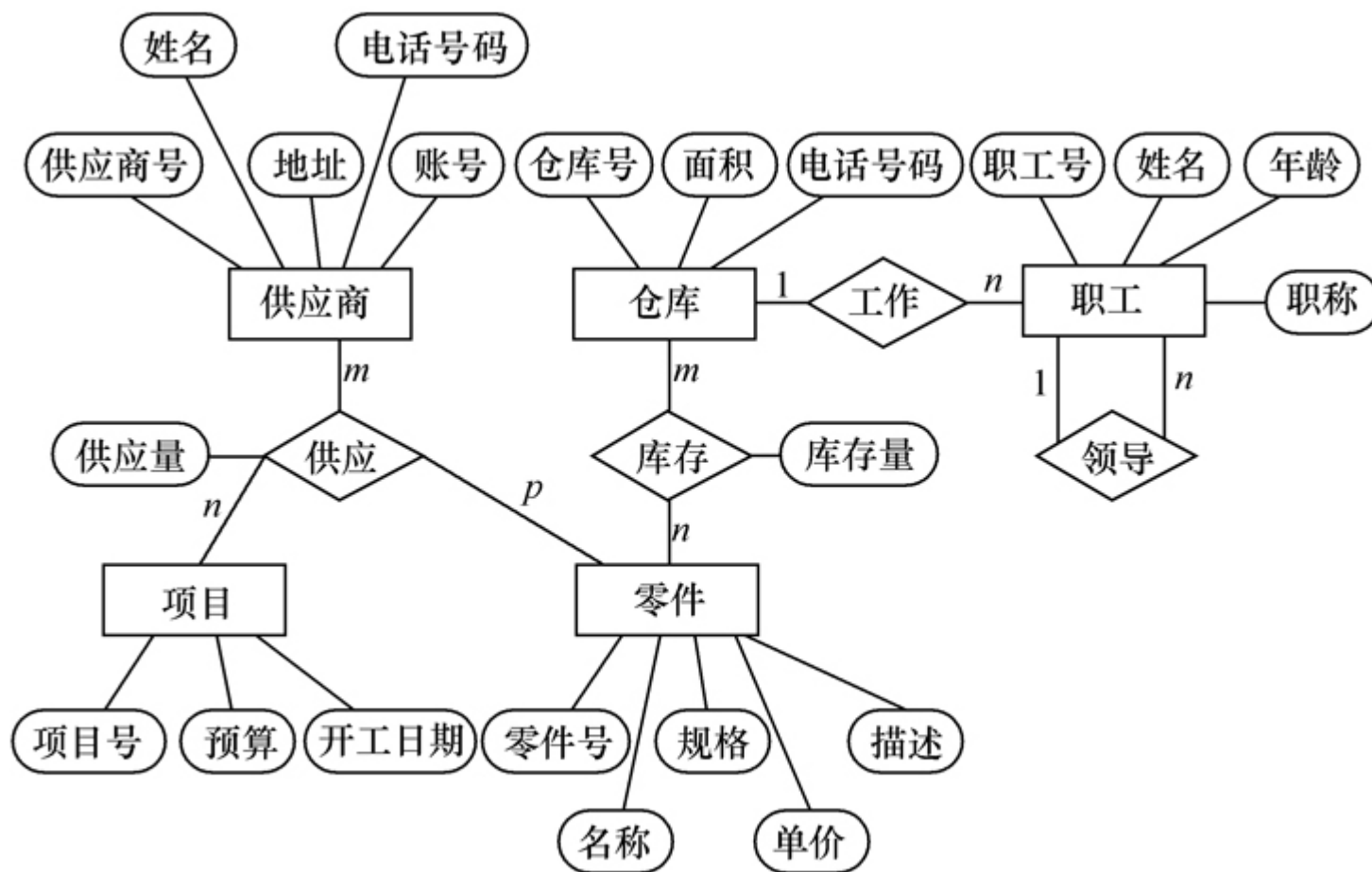
### ■ 实体

- **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

### ■ 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。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
- 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

# 一个实例



(c) 完整的实体-联系图



# 练习：画ER图

- **练习1：**某工厂生产若干种**产品**，每种产品由不同**零件**组成，有的零件可用在不同的产品上。这些零件由不同的**原料**制成，不同零件所用的原料可以相同。这些零件按所属的不同产品分别放在不同的**仓库**中，原料按照类别放在不同的仓库中。
- **练习2：**学校有若干个**系**，每个系有若干个**教研室**，每个教研室有若干**教师**。同时每个系有若干**学生**。





# 1.2 数据模型

---

## 1.2.1 两大类数据模型

## 1.2.2 概念模型

## 1.2.3 数据模型的组成要素

## 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。
- 数据模型对完整性约束条件的定义
  - 必须遵守的基本的通用的完整性约束条件。
    - 例如，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。
  - 反映具体应用所涉及的特定的约束条件。
    - 例子：学校数据库中规定博士学生的年龄必须小于45岁
    - 例子：银行的系统中规定帐号的余额不能小于1元



# 1.2 数据模型

---

## 1.2.1 两大类数据模型

## 1.2.2 概念模型

## 1.2.3 数据模型的组成要素

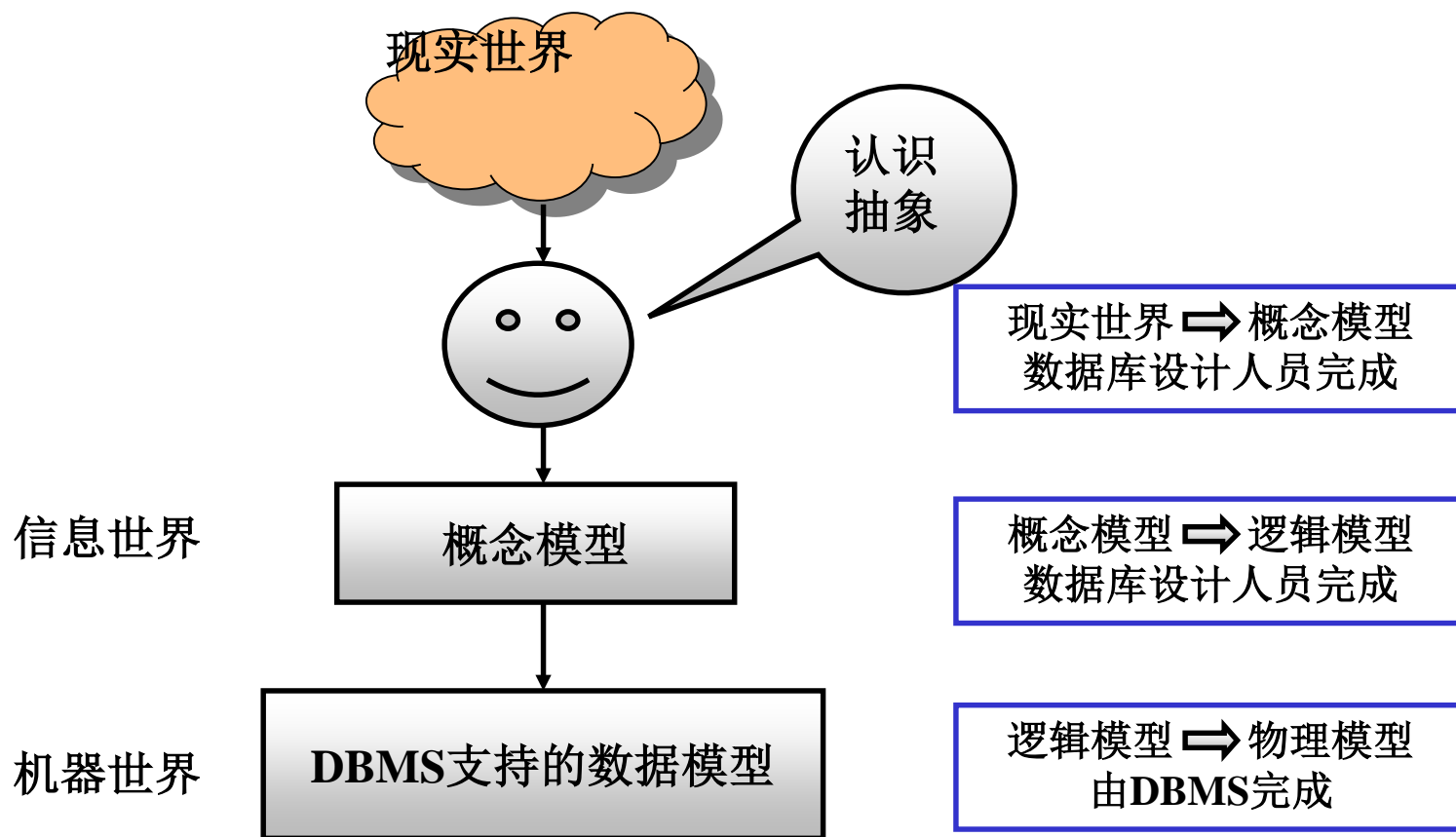
## 1.2.4 最常用的数据模型

## 1.2.5 层次模型

## 1.2.6 网状模型

## 1.2.7 关系模型

# 两大类数据模型 (续)



现实世界中客观对象的抽象过程



## 1.2.4 最常用的数据模型

---

- 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**)



## 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



# 一、层次数据模型的数据结构

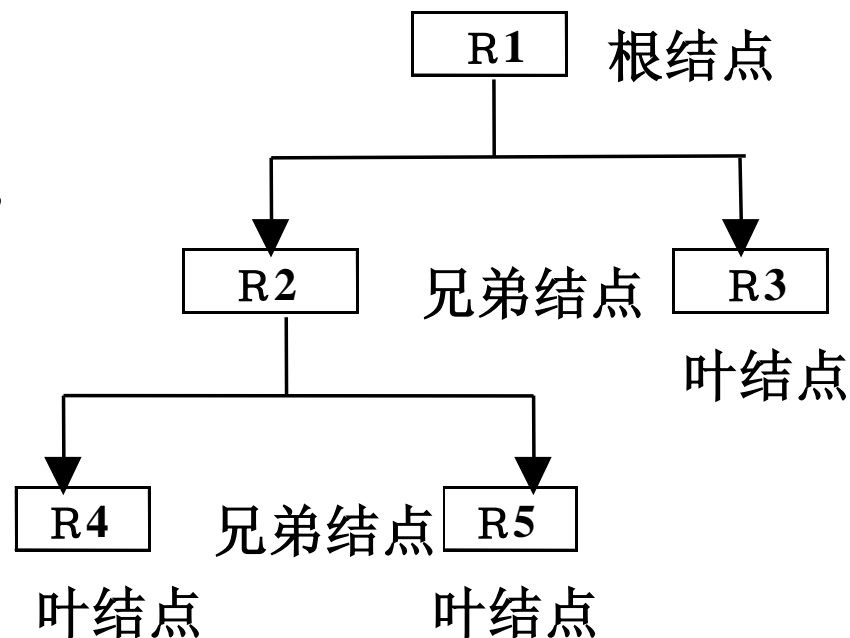
## ■ 层次模型

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

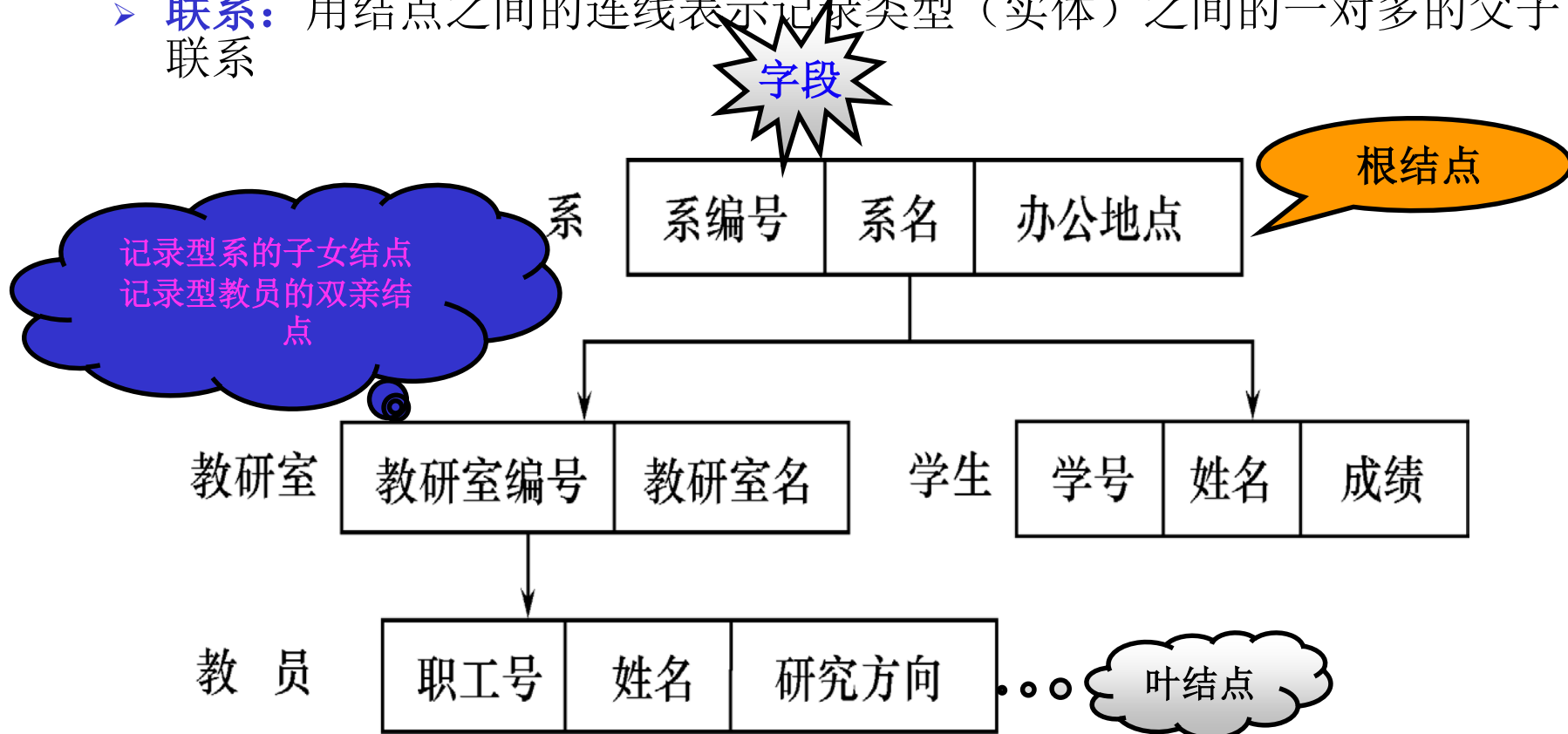
- 根结点，双亲结点，兄弟结点，叶结点



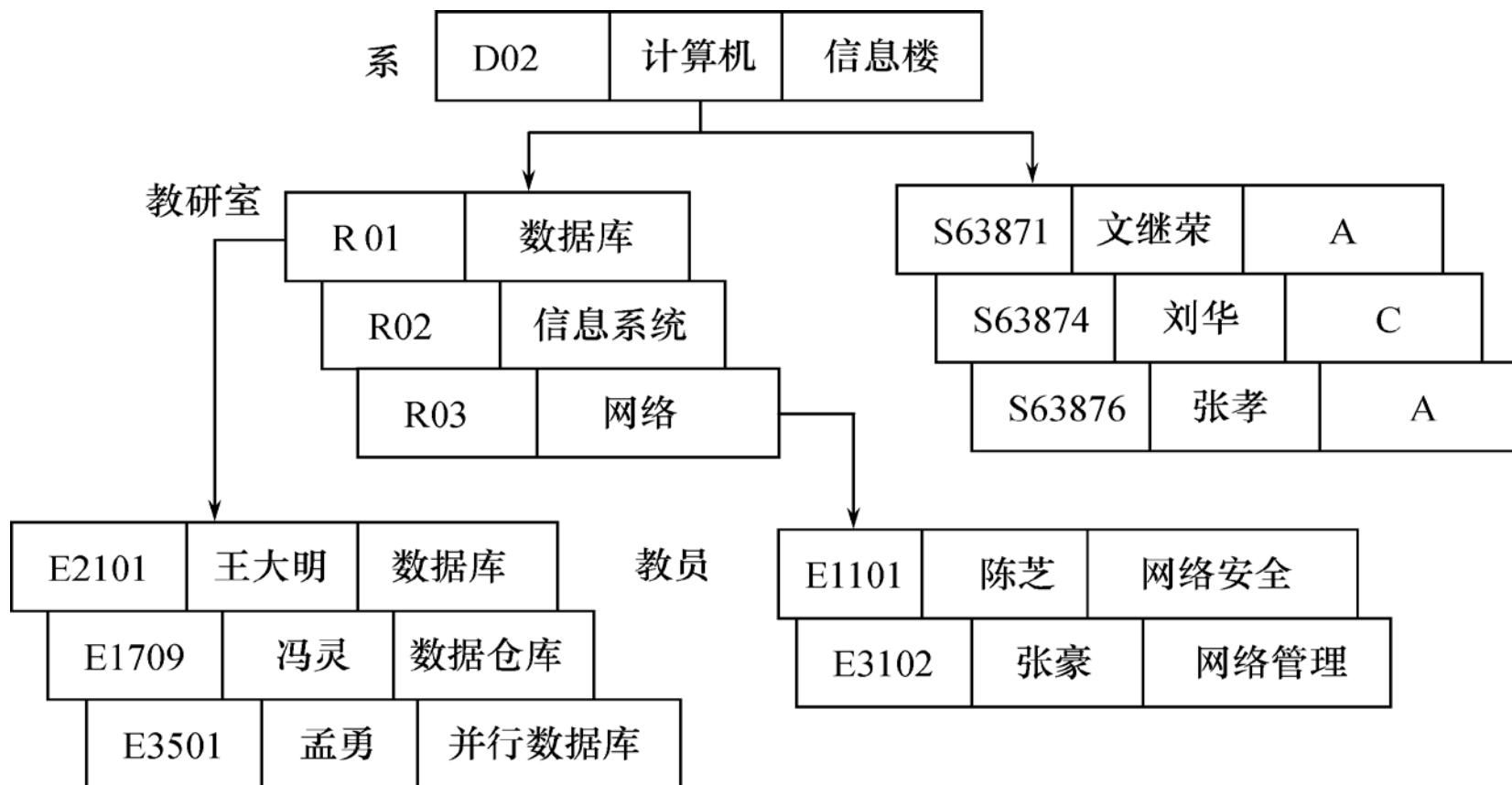
# 层次数据模型的数据结构(续)

## ■ 表示方法

- **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



# 层次数据模型的数据结构(续)



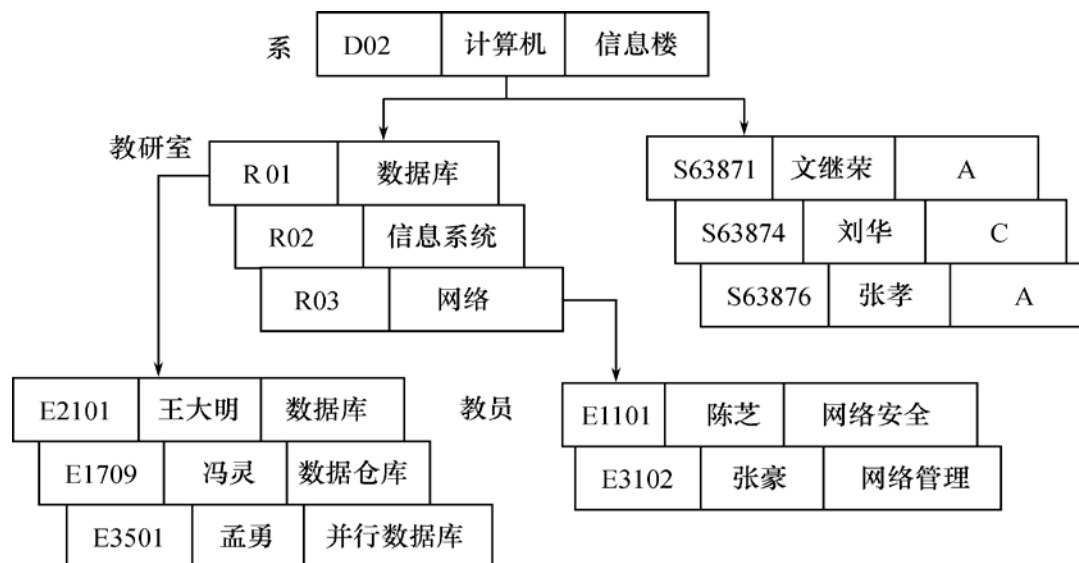
教员学生层次数据库的一个值



# 层次数据模型的数据结构(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
- 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





# 多对多联系在层次模型中的表示

- 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
  - 例子：学生选课数据库

# 多对多联系在层次模型中的表示

## ■ 例子：学生选课数据库

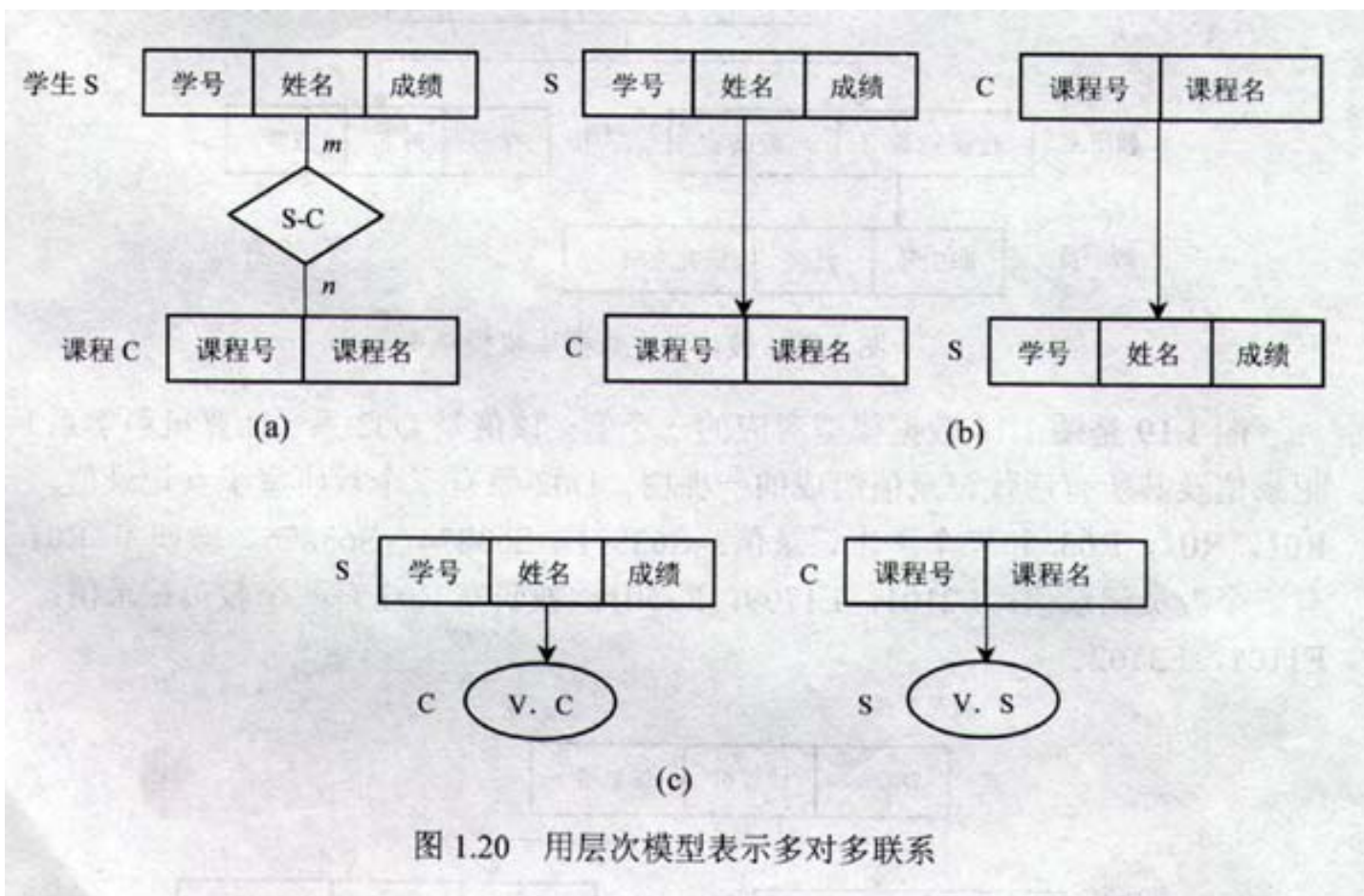


图 1.20 用层次模型表示多对多联系



# 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

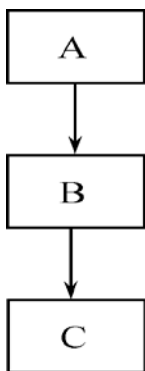
## ■ 层次模型的完整性约束条件

- 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

# 四、层次数据模型的存储结构

## ■ 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



(a)

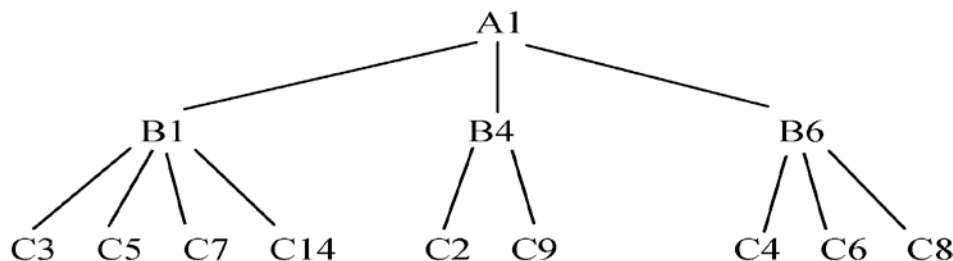


图1.20 层次数据库及其**(b)**实例

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

图1.21 邻接法



# 层次数据模型的存储结构（续）

---

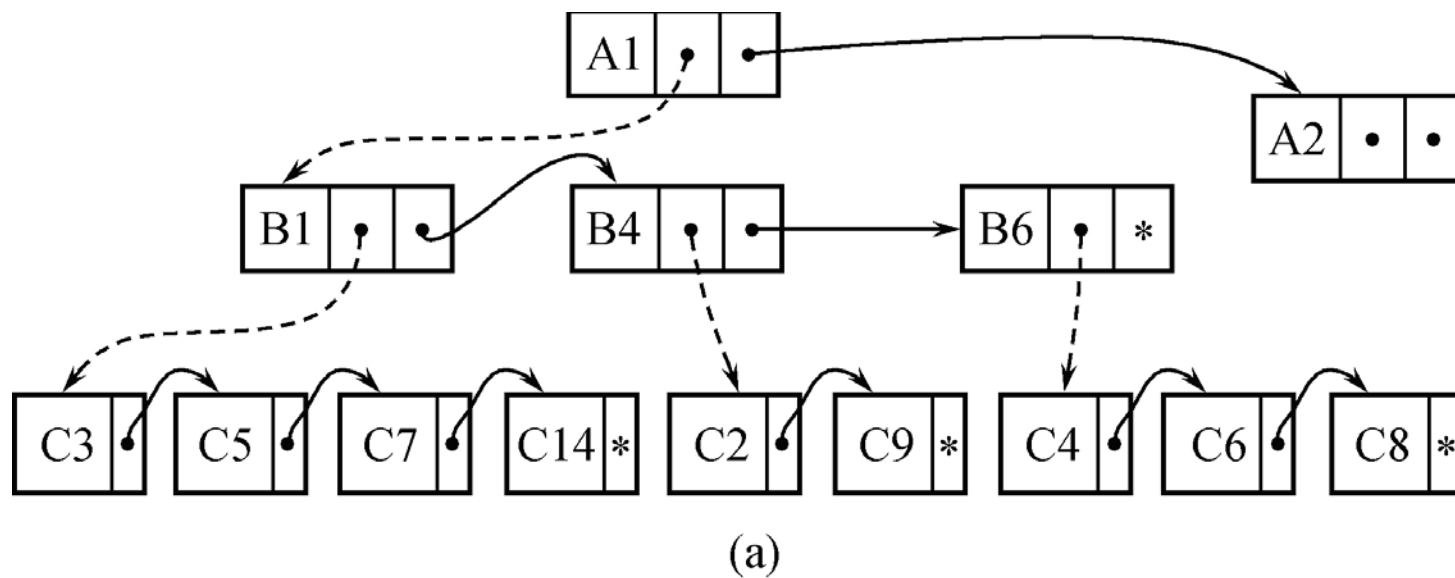
## ■ 链接法

- 用指引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

# 层次数据模型的存储结构（续）

## ■ 子女-兄弟链接法

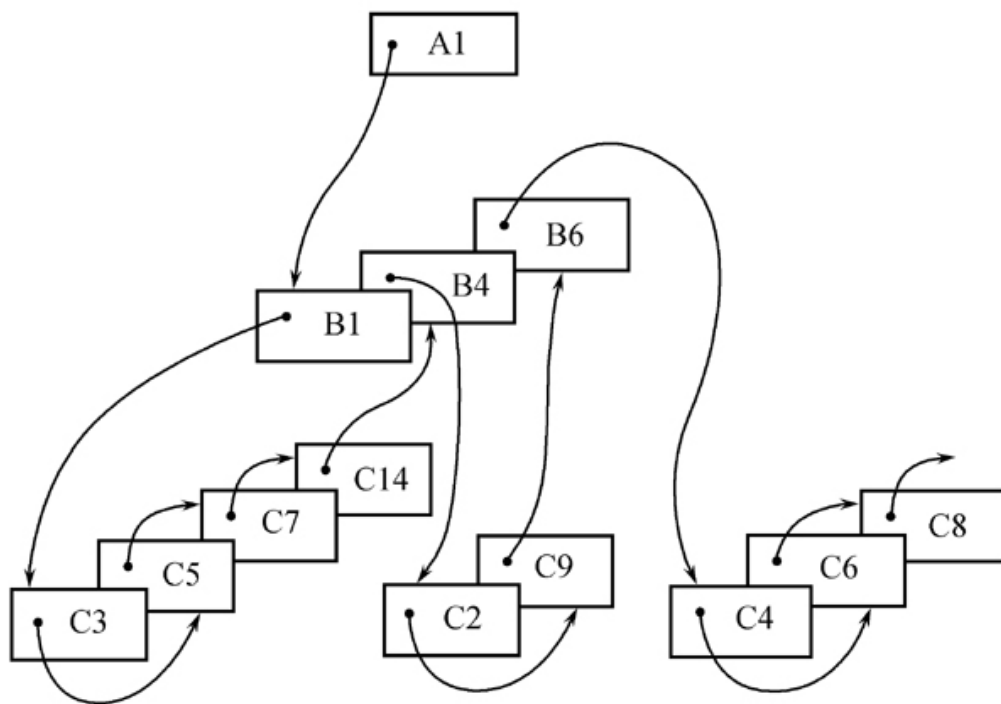
每个记录设两类指针，分别指向最左边的子女（每个记录型对应一个）和最近的兄弟



# 层次数据模型的存储结构（续）

## ► 层次序列链接法

按树的前序穿越顺序链接各记录值



(b)



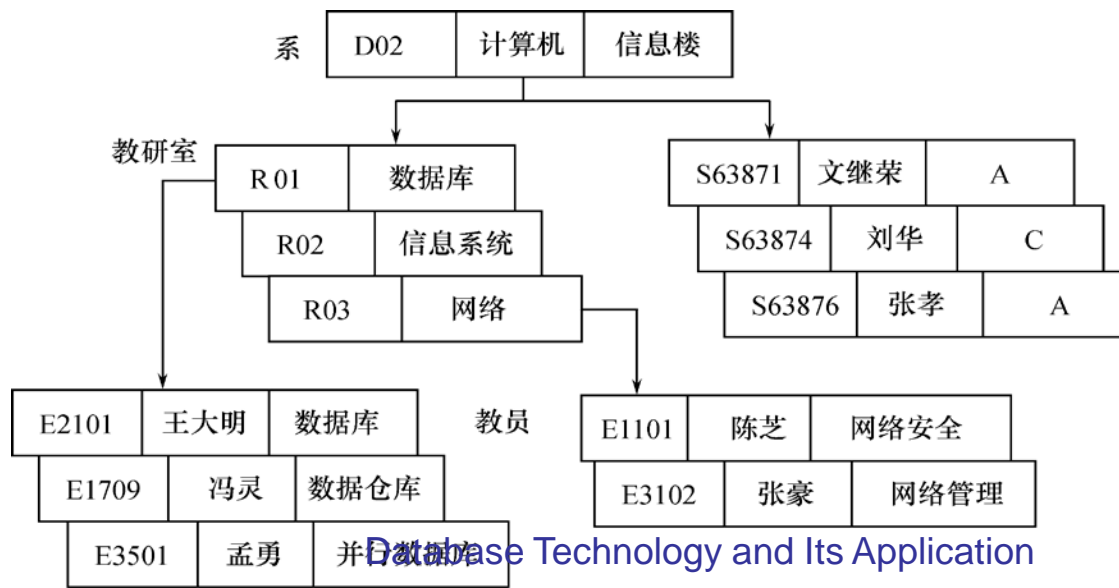
# 五、层次模型的优缺点

## ■ 优点

- 层次模型的数据结构比较简单清晰
- 查询效率高，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，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





# 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



## 1.2.6 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
  - 通用电气公司的IDS
  - Cullinet Software Inc.公司的 IDMS
  - Univac公司的 DMS1100
  - Honeywell公司的IDS/2
  - HP公司的IMAGE



# 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：

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。

## ■ 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）

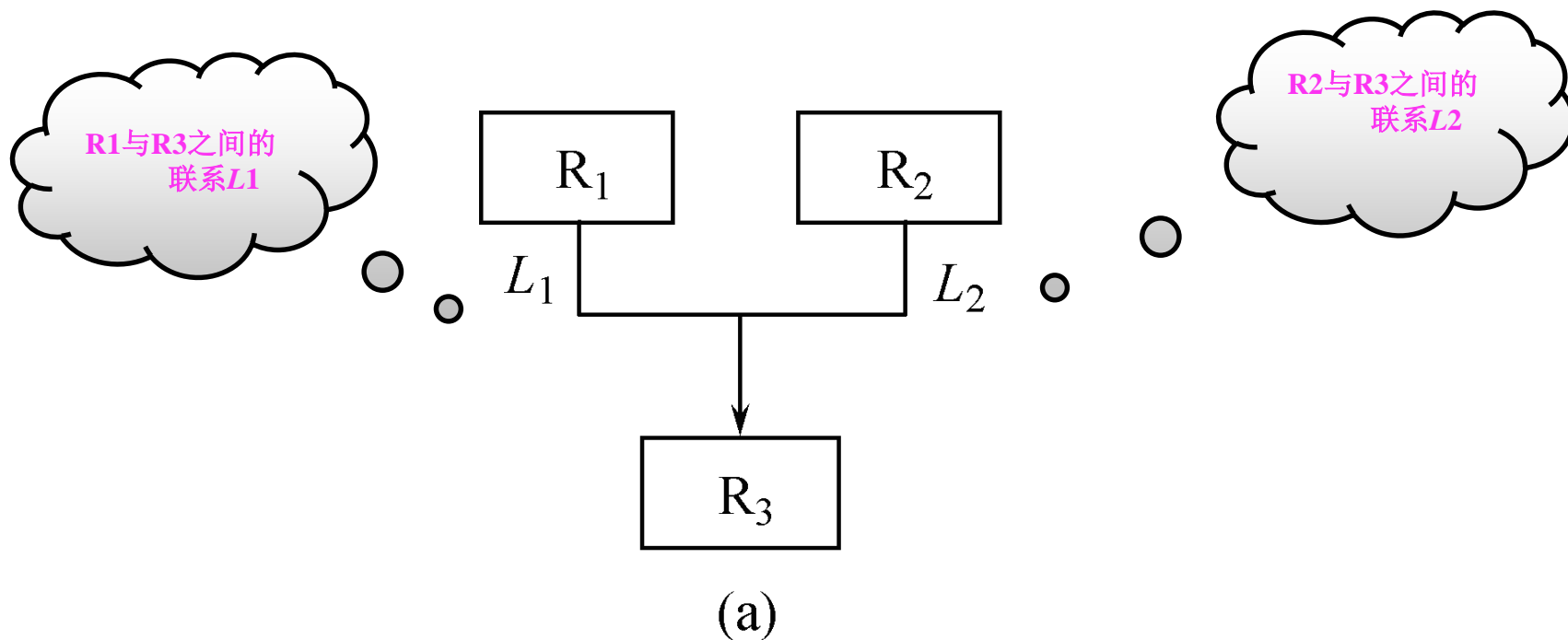


- 网状模型与层次模型的区别
  - 网状模型允许多个结点没有双亲结点
  - 网状模型允许结点有多个双亲结点
  - 网状模型允许两个结点之间有多种联系（复合联系）
  - 网状模型可以更直接地去描述现实世界
  - 层次模型实际上是网状模型的一个特例

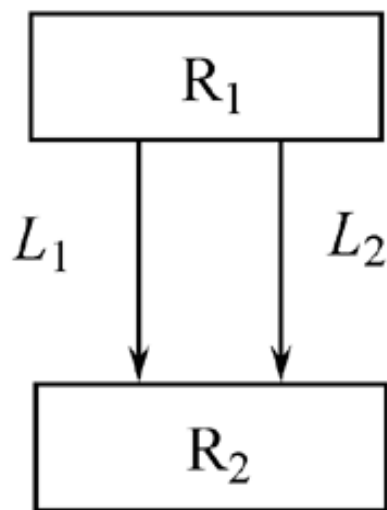
# 网状数据模型的数据结构（续）



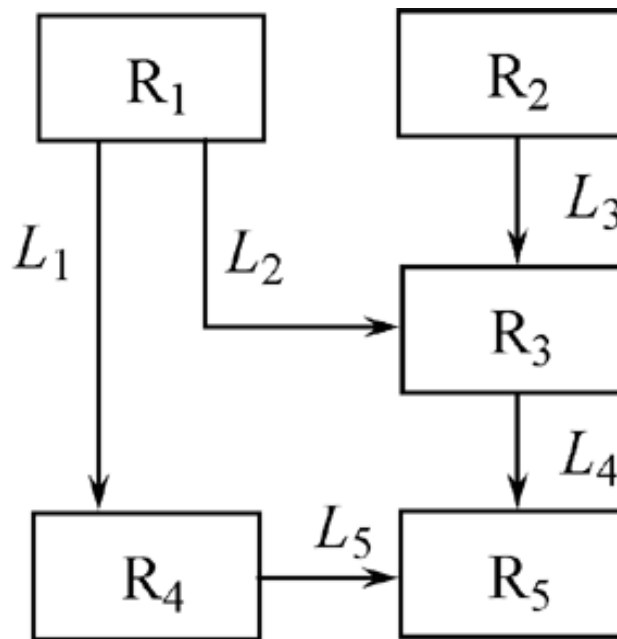
- ❖ 网状模型中子女结点与双亲结点的联系可以不唯一  
要为每个联系命名，并指出与该联系有关的双亲记录和子女记录



# 网状数据模型的数据结构（续）



(b)



(c)

网状模型的例子

# 网状数据模型的数据结构（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

- 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联系记录，由**3**个数据项组成

**学号、课程号、成绩**

表示某个学生选修某一门课程及其成绩





# 网状数据模型的数据结构（续）

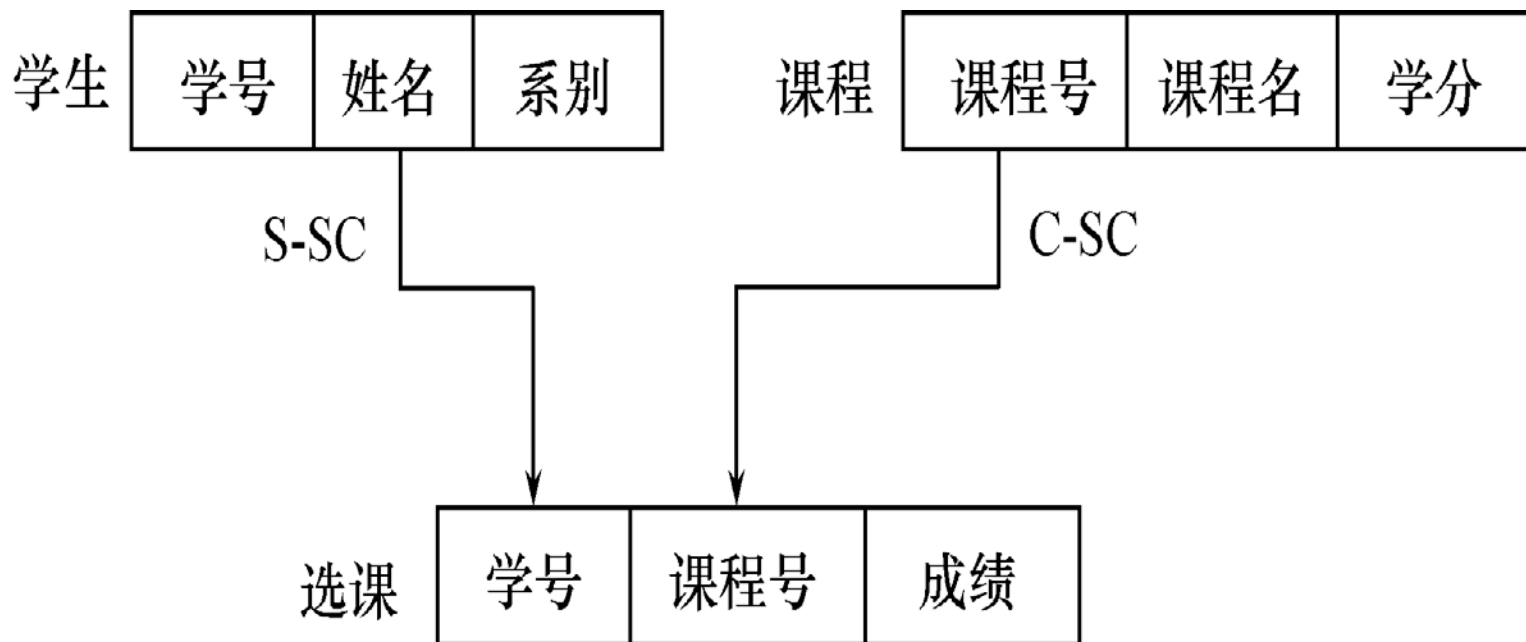


图1.24 学生/选课/课程的网状数据模型



# 网状数据模型的操纵与完整性约束（续）

- 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：唯一标识记录的数据项的集合
  - 支持双亲记录和子女记录之间某些约束条件
    - 有些子女记录要求双亲记录存在才能插入，双亲记录删除时也连同删除。例如学生选课记录。



# 三、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
  - 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（续）

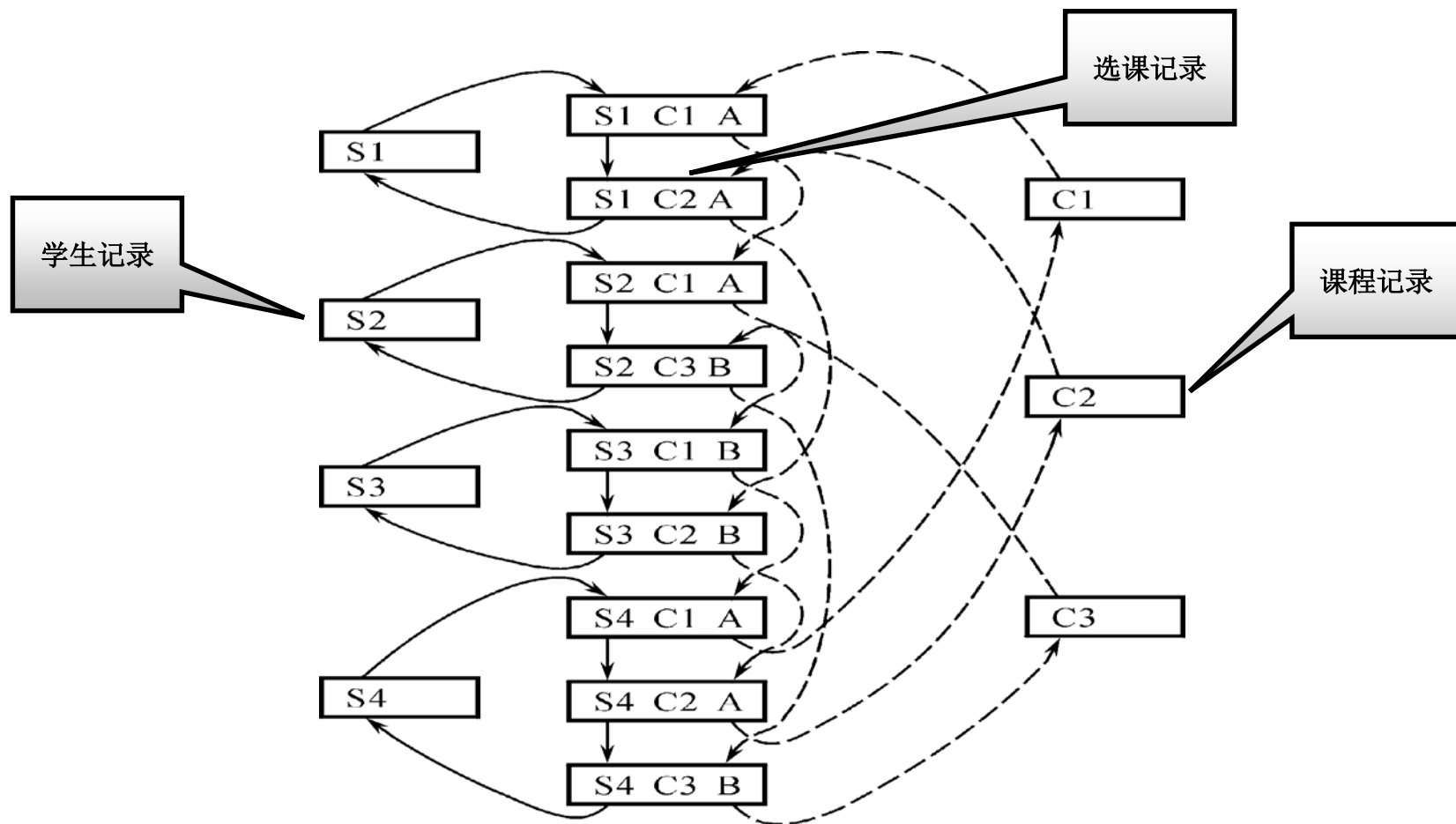


图1.25 学生/选课/课程的网状数据库实例



# 四、网状数据模型的优缺点

## ■ 优点

- 能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
- 记录类型联系变动后涉及链接指针的调整，扩充和维护都比较复杂

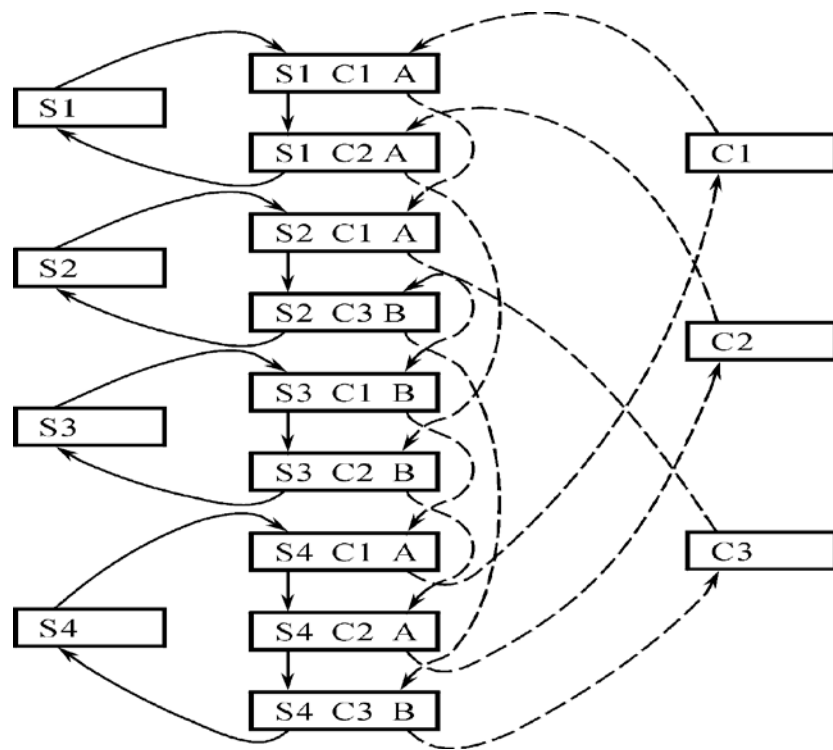
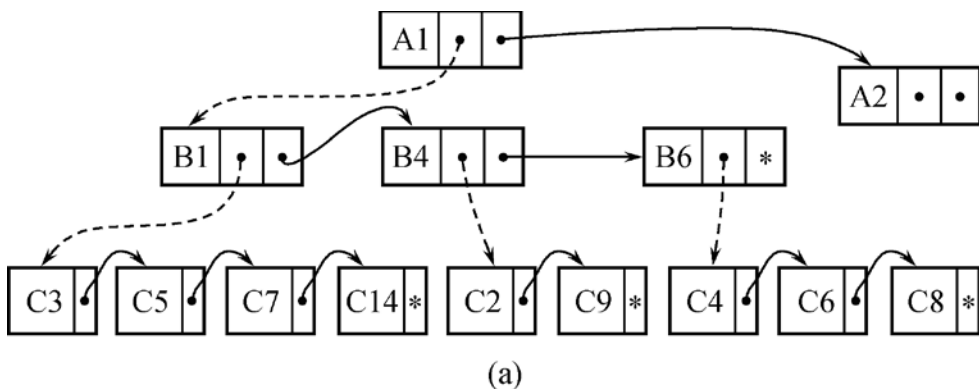


# 格式化模型的共同缺点

- 记录之间的联系是通过存取路径实现的，应用程序在访问数据时必须选择适当的存取路径，用户必须了解系统结构的细节，加重了编写应用程序的负担。
- 不支持集合处理，即未提供一次处理多个记录的功能。

# 格式化模型的共同缺点

## ■ 通过存储路径访问数据





# 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





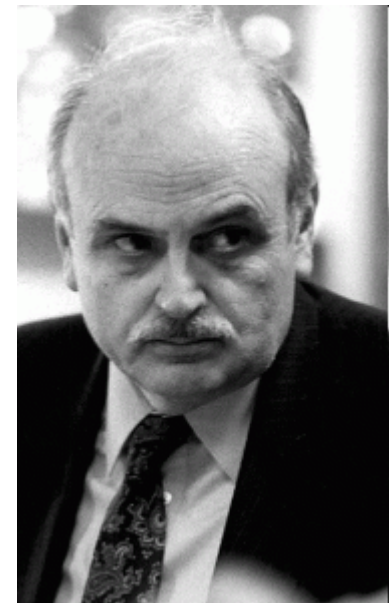
## 1.2.7 关系模型

---

- 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

# 小插曲：Edgar F. Codd 生平

- 1923年生于英国多塞郡波特兰岛，曾就读于牛津大学，主修数学和化学。
- 二战中，作为一名机长在英国皇家空军服役。
- 战争结束后的1948年，来到纽约成为IBM公司的一名程序员。
- 50年代末，为IBM STRETCH计算机发明了“多道程序设计”技术。
- 60年代初重返校园，在密歇根大学深造，1963年获得硕士学位，1965年取得计算机科学博士学位。
- 毕业后调到IBM公司San Jose研究中心工作，开始从事关系型数据管理模型的研究。
- 1970年发表关系数据模型的开创性论文。



# 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

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

属性

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

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

主码

分量

属性

元组

# 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

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）

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

元组

主码

分量

属性

# 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

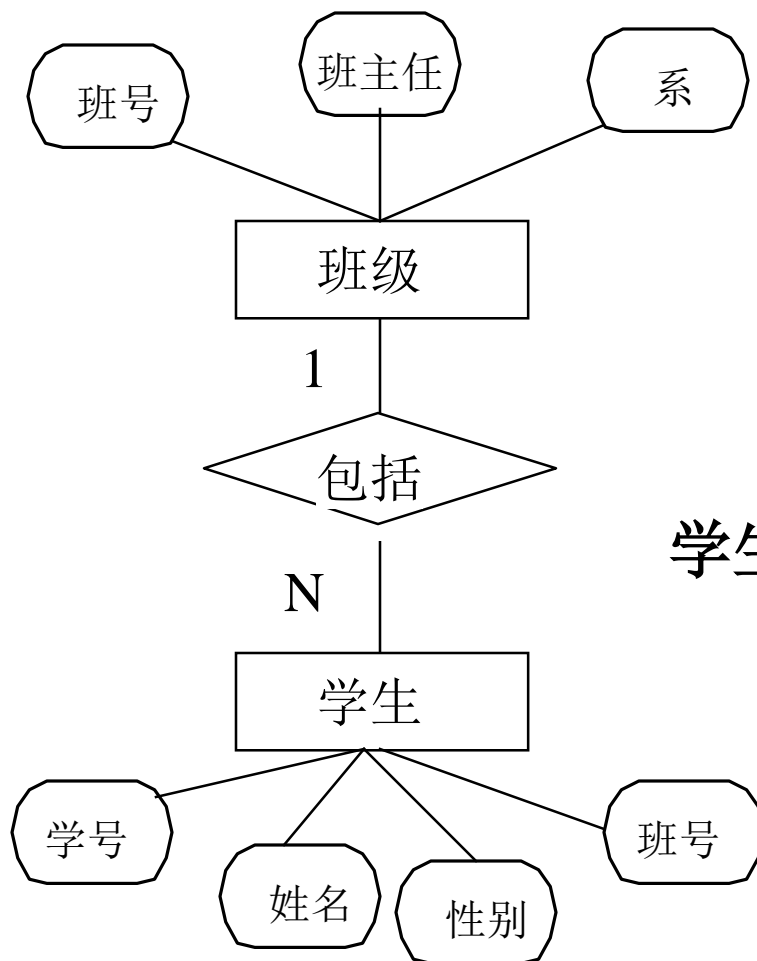
图1.27中工资是可分的数据项，不符合关系模型要求

班号	组名	工资		扣除	实发
		基本	补助		
004	甲组	3200	20	100	3120
	乙组	1500		50	1450
008	甲组	2200	150	100	2250
...	...	...	...	...	...

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）



班级（**班号**，班主任，系）

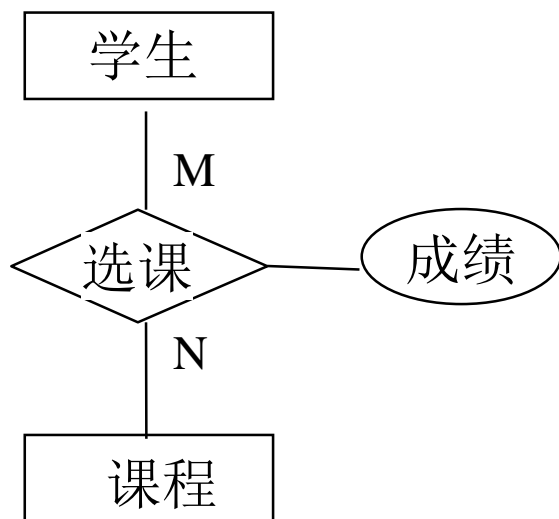
学生（学号，姓名，性别，**班号**）

1 : N

# 关系数据模型的数据结构（续）



学生（**学号**，姓名，性别，班号）



选课（**学号**，**课程号**，成绩）

课程（**课程号**，课程名，学分）

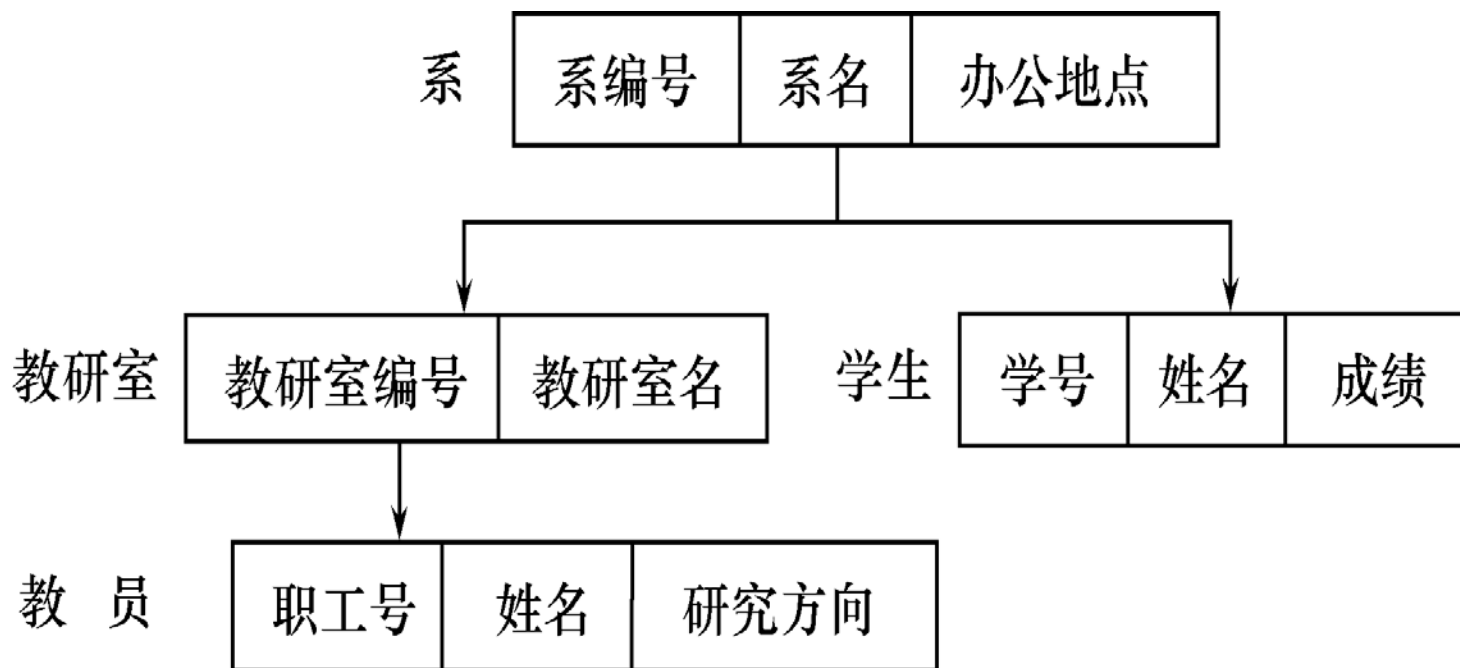
**M : N**



# 关系数据模型的数据结构（续）



## ■ 例子：教员学生数据库





# 关系数据模型的操纵与完整性约束

- 数据操作是集合操作，操作对象和结果都是关系
  - 查询
  - 插入
  - 删除
  - 更新
  
- 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*

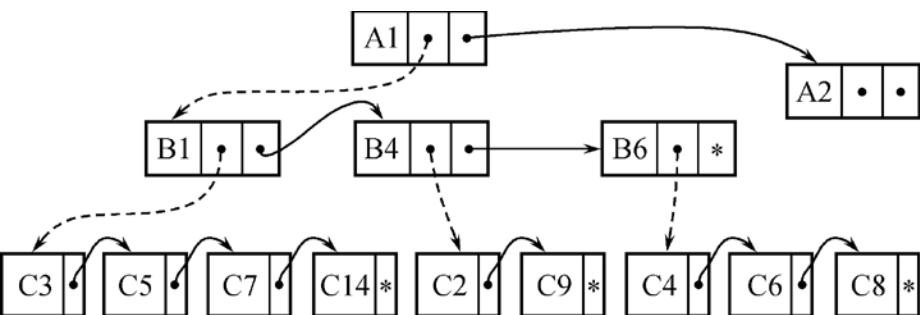


# 三、关系数据模型的存储结构

---

- 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

# 关系数据模型的优缺点

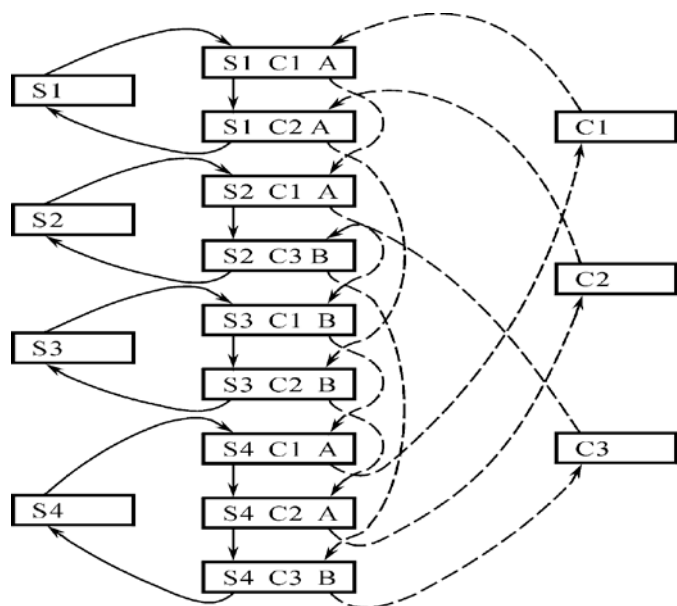


(a)

系 (系编号, 系名, 办公地点)

教研室 (教研室编号, 教研室名, 系编号)

教员 (职工号, 姓名, 研究方向, 教研室编号)



学生 (学号, 姓名, 性别, 班号)

选课 (学号, 课程号, 成绩)

课程 (课程号, 课程名, 学分)



# 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
- 可以描述一对一、一对多和多对多的联系
- 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
- 存取路径对用户透明
  - 用户只要指出“干什么”，不必详细说明“怎么干”
  - 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

## ■ 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



# 小结

---

- 数据库设计的一般过程
- 数据模型的组成要素
  - 数据结构、数据操作、数据的完整性约束条件
- 概念模型
  - 实体、属性、联系
  - E-R图
- 层次模型
- 网状模型
- 关系模型



# 课堂练习：画E-R图

---

- 某航班管理系统中有如下基本信息：
  - **乘机人**：身份证号、姓名、住址、电话；
  - **航班**：航班号、起飞日期、起飞时间、飞机型号
- 顾客可以预定某天的某个航班生成订单，订单中包含座位号信息。



# 作业：画E-R图

---

- 学校有若干系，每个系有若干班级，每个班级有若干本科生或研究生；每个系有若干教研室，每个教研室有若干教授、副教授和讲师，教授和副教授可以指导研究生。学生可以选修课程。