上海交通大学
SHANGHAI JIAO TONG UNIVERSITY

SJTU

数 据 库 技 术

Database System Technology

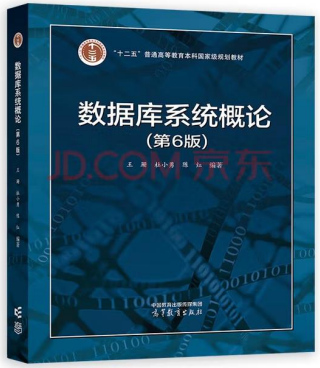
郭 捷

(guojie@sjtu.edu.cn)

教学参考书

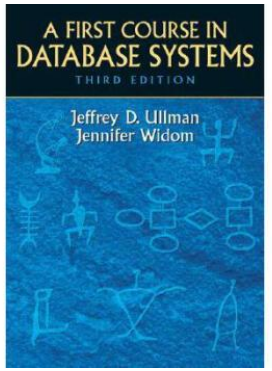
教材：

- 王珊，萨师煊，《数据库系统概论(第6版)》，高等教育出版社，2023年3月

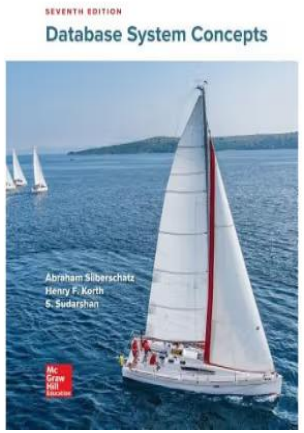


参考书：

- A First Course in Database Systems (Third Edition), Jeffrey. D. Ullman, Jennifer Widom, Dept. of Computer Science, Stanford University



- Database System Concepts (Seventh Edition), Abraham Silberschatz, Henry F. Korth, S. Sudarshan,



教学安排和考核方法

- 48课时，2024年2月 ～ 2024年6月（1 ～ 16周）

  1 ～ 16周　　周一下午7-8节

  1 ～ 8周　　周四上午3-4节

考核方法：

1）期末考试（总成绩的60%）

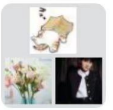
2）动手实践大作业（总成绩的20%）

3）最新进展报告大作业（总成绩的10%）

4）课堂讨论、课后作业和出勤（总成绩的10%）

课件查阅和作业上载

- 课件查阅：
  1. 登录：oc.sjtu.edu.cn
  2. 预习过程中有问题，可以发邮件 guojie@sjtu.edu.cn
     或 班级微信群发布 或 微信私信（请提前几个小时）；



群聊：2024 数据库技术 郭捷

该二维码7天内(2月22日前)有效，重新进入将更新

- 课后作业：
  1. 登录：oc.sjtu.edu.cn；
  2. 作业截止时间：下次上课前4个小时（批阅时间），课上会针对学生的作业问题进行批讲；

学习方式

- **听课**

（启发式、讨论式）

- **读书（资料）**

（预习、复习、课外文献阅读）

- **作业**

（课后习题、综合练习、大作业、动手能力训练）

第一章 数据库系统概论

上海交通大學
SHANGHAI JIAO TONG UNIVERSITY

目 录

数据库的地位

- 数据库技术产生于六十年代末，是数据管理的最新技术，是计算机科学的重要分支；
- 数据库技术是信息系统的核心技术和重要基础设施：
  - 联机事务处理（On-Line Transaction Processing，OLTP)
  - 联机分析处理（On-Line Analysis Processing，OLAP)
  - 计算机辅助设计与制造（CAD/CAM)、计算机集成制造系统（CIMS)
  - 电子政务（e-Government)、电子商务（e-Commerce)
  - 地理信息系统（GIS)

01

四个基本概念

四个基本概念

- 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 (Data)

- 数据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：
  - ✓ 数值、文字、图形、图象、声音、视频等；
- 数据的特点：
  - ✓ 数据与其语义是不可分的；

例如：（李明，男，199505，江苏，计算机系，2013）

- 语义：学生姓名、性别、出生年月、籍贯、所在系别、入学时间

2. 数据库 (DataBase，DB)

- 人们收集并抽取出一个应用所需要的大量数据之后，将其保存起来，以供进一步加工处理，抽取有用信息，转换为有价值的知识。

- **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、可共享的大量数据集合。

数据库 举例

学生登记表

学　号	姓　名	年　令	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
…	…	…	…	…	…

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；

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；

数据库管理系统的主要功能

- 数据定义功能
  - ✓ 提供数据定义语言(Data Definition Language, DDL)，定义数据库中的数据对象的组成与结构。
- 数据组织、存储、管理功能
  - ✓ 文件结构和存取方式
  - ✓ 数据如何联系
  - ✓ 提高存储空间利用率、方便存取

数据库管理系统的主要功能

数据操纵功能

- ✓ 提供数据操纵语言（Data Manipulation Language, DML）
- ✓ 操纵数据实现基本操作，如查询、插入、删除和修改

数据库的事务管理和运行管理

- ✓ 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

数据库管理系统的主要功能

- 数据库的建立和维护功能(实用程序或管理工具)
  - ✓ 数据库数据批量装载和转储
  - ✓ 介质故障恢复
  - ✓ 数据库的重组织
  - ✓ 性能监视、分析
- 其他功能
  - ✓ 数据库管理系统与网络中其它软件系统的通信
  - ✓ 数据库管理系统各系统之间的数据转换
  - ✓ 异构数据库之间的互访和互操作

4. 数据库系统

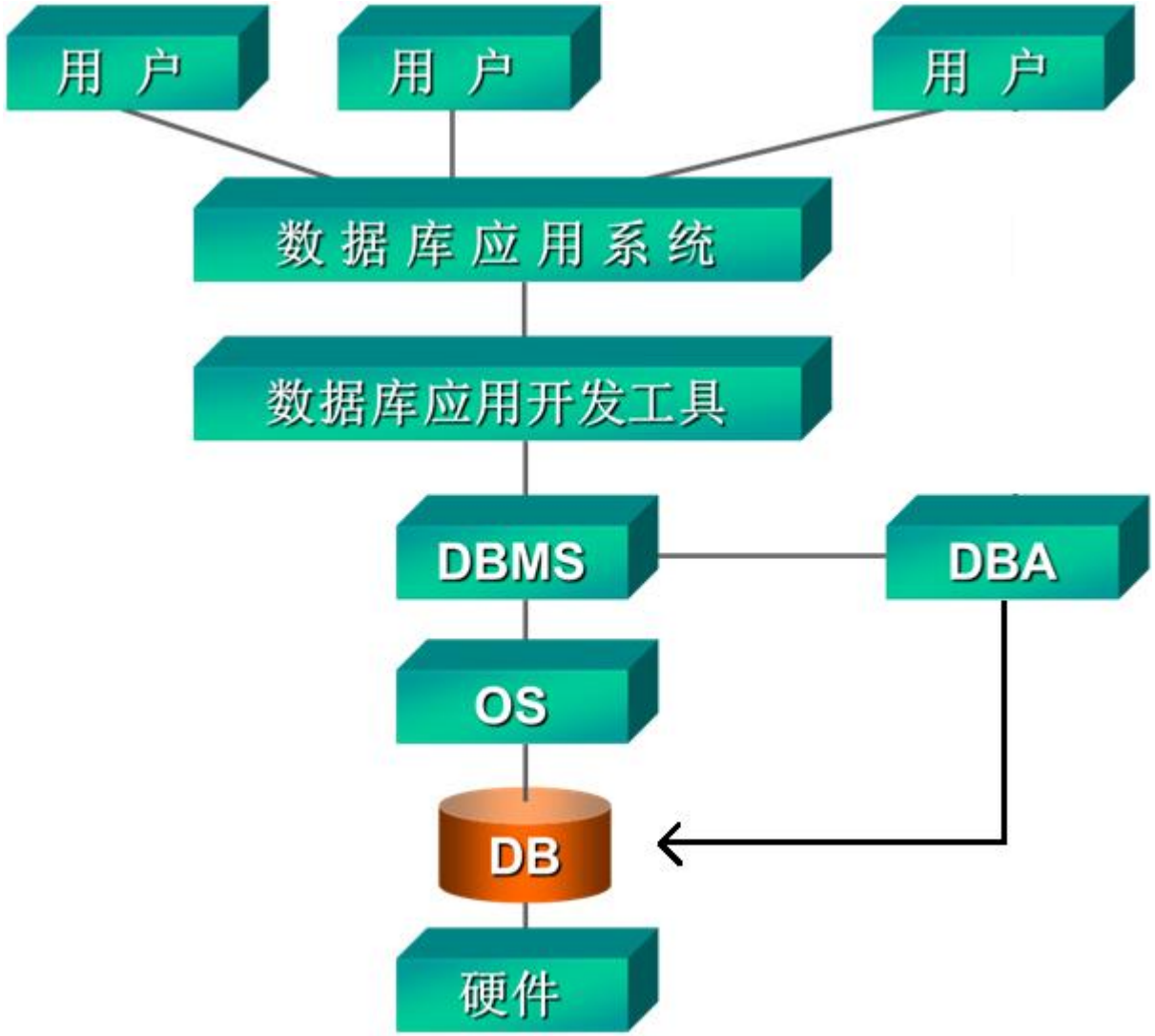
数据库系统

- 数据库系统（Database System，简称DBS），是指在计算机系统中引入数据库和DBMS后的系统构成。
- 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（及其应用开发工具）、应用系统、数据库管理员（DataBase Administrator，DBA）构成。

数据库系统结构图



02

数据库管理技术的产生与发展

数据管理技术

数据管理技术

- 对数据进行分类、组织、编码、存储、检索和维护，是数据处理和数据分析的中心问题。

数据管理技术的发展过程

✓人工管理阶段(40年代--50年代中)

✓文件系统阶段(50年代末--60年代中)

✓数据库系统阶段(60年代末--现在)

1. 人工管理阶段（40年代中--50年代中）

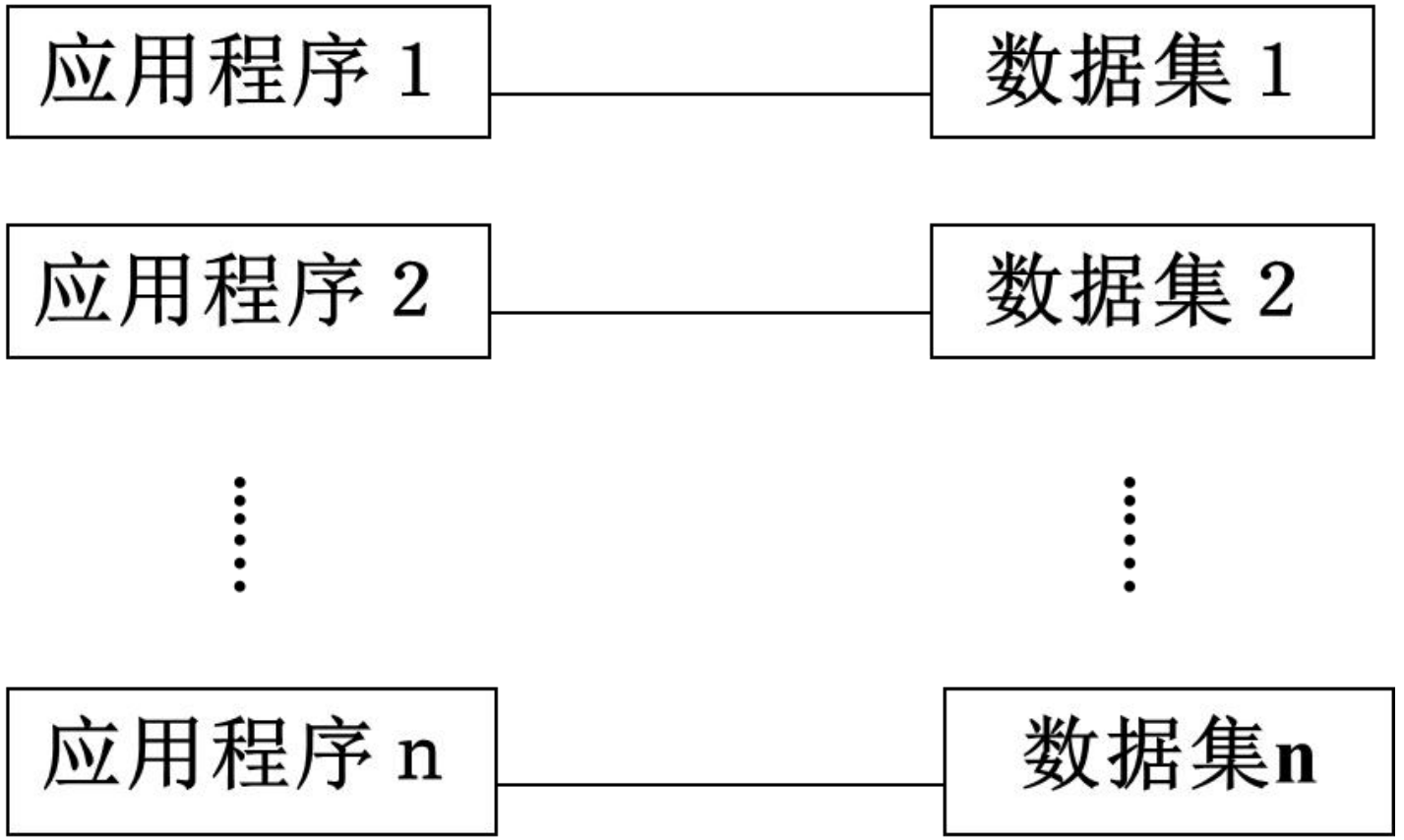
背景

- 应用背景：计算机主要用于科学计算；
- 硬件背景：外存只有磁带、卡片、纸带，没有直接存储设备；
- 软件背景：没有操作系统、没有管理数据的软件；
- 处理方式：批处理；

特点

- 数据不保存，没有文件的概念；
- 应用程序管理数据，程序员负担很重；
- 数据面向某个应用程序，无共享，冗余度大；
- 应用程序与数据的关系：一一对应，数据不具有独立性；

应用程序与数据对应关系（人工管理阶段）



一一对应关系

2. 文件系统阶段

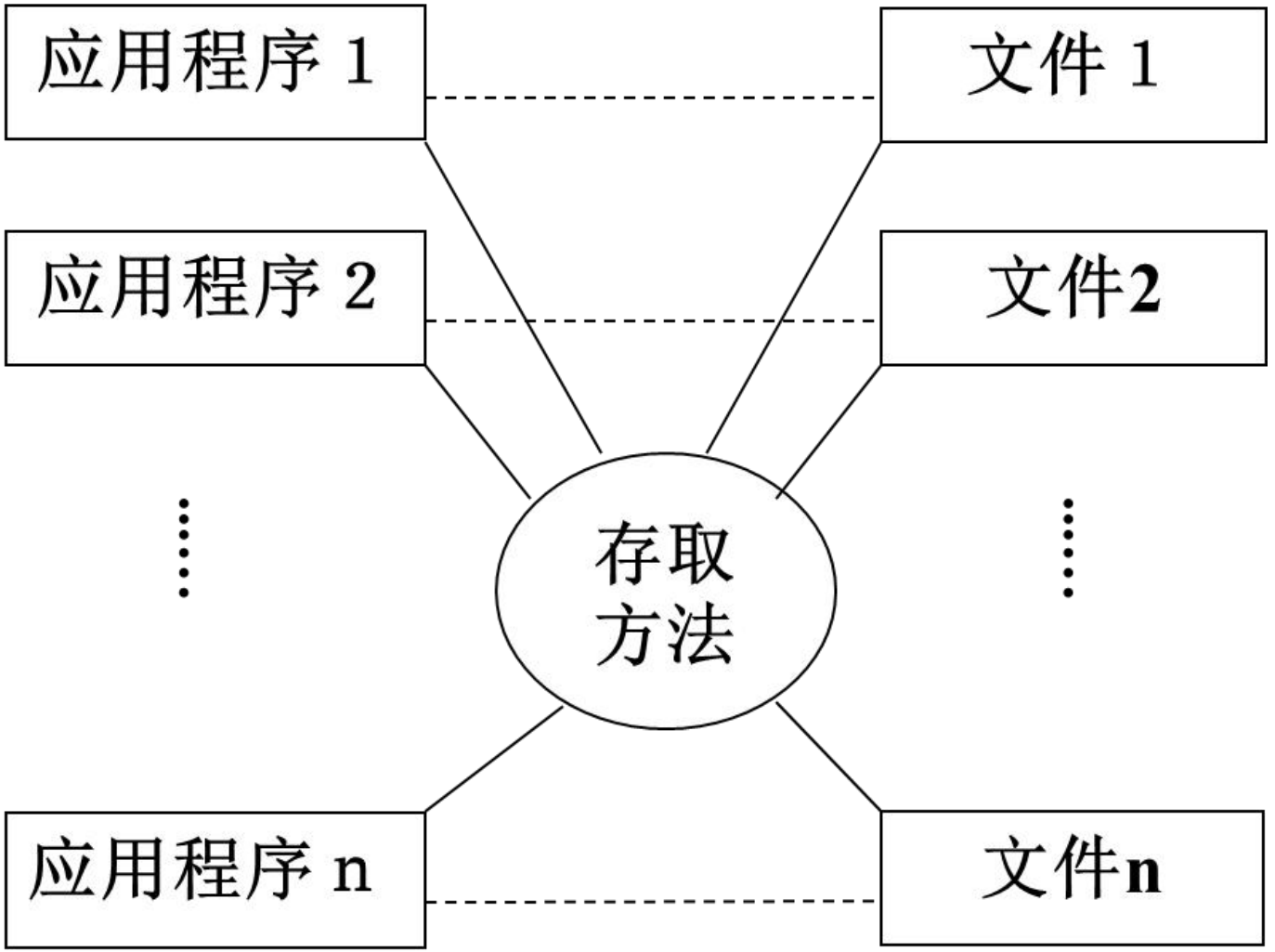
- 时期
  - 50年代末--60年代中

- 产生的背景
  - 应用需求：科学计算、管理
  - 硬件水平：磁盘、磁鼓等存储设备
  - 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：某一应用程序
- 数据的共享程度：共享性差、冗余度极大
- 数据的独立性：独立性差，数据的逻辑结构改变必须修改应用程序
- 数据的结构化：记录内有结构，整体无结构数据
- 控制能力：应用程序自己控制

应用程序与数据对应关系（文件系统阶段）



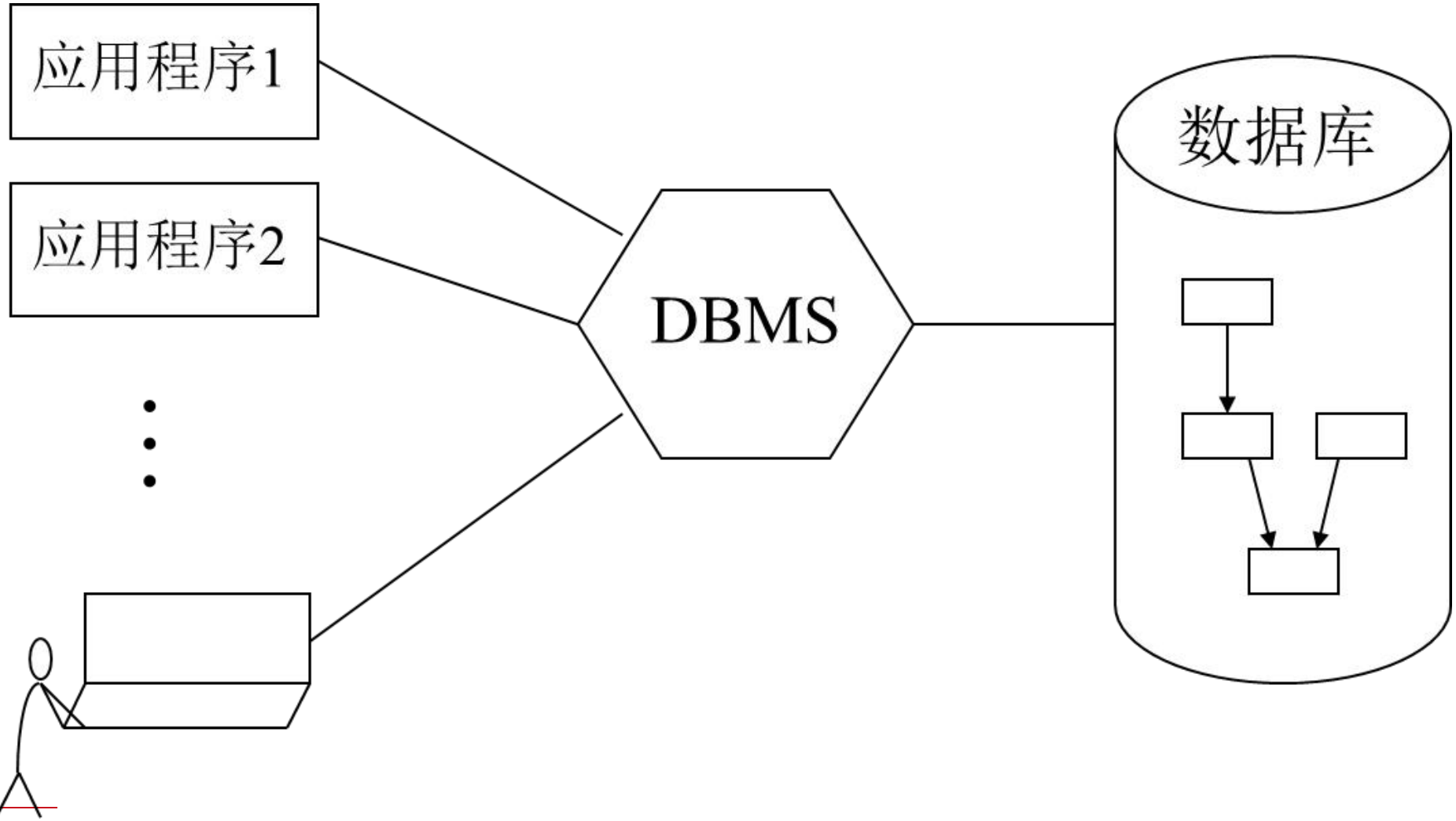
3. 数据库系统阶段

- 时期
  - 60年代末以来
- 产生的背景
  - 应用需求：大规模管理
  - 硬件水平：大容量磁盘、磁盘列阵
  - 软件水平：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

数据库系统阶段的特点

- 数据的管理者：DBMS
- 数据面向的对象：现实世界
- 数据的共享程度：共享性高，冗余度小
- 数据的独立性：高度的物理独立性和一定的逻辑独立性
- 数据的结构化：整体结构化，用数据模型来表示
- 控制能力：由DBMS统一管理和控制

应用程序与数据对应关系（数据库系统阶段）



03

数据库系统的特点

一个例子：用数据库系统实现学籍管理

存储数据：用CREAT命令建立两张表

```
CREATE TABLE Student (    //存放学生基本信息
    Sno CHAR(9),
    Sname CHAR(20),
    Ssex CHAR(2),
    Major CHAR(20) );
CREATE TABLE Award (      //存放学生奖励情况
    Sno CHAR(9),
    Details VARCHAR(2000));
```

查询数据：用SELECT-FROM-WHERE查询

数据录入：用INSERT命令

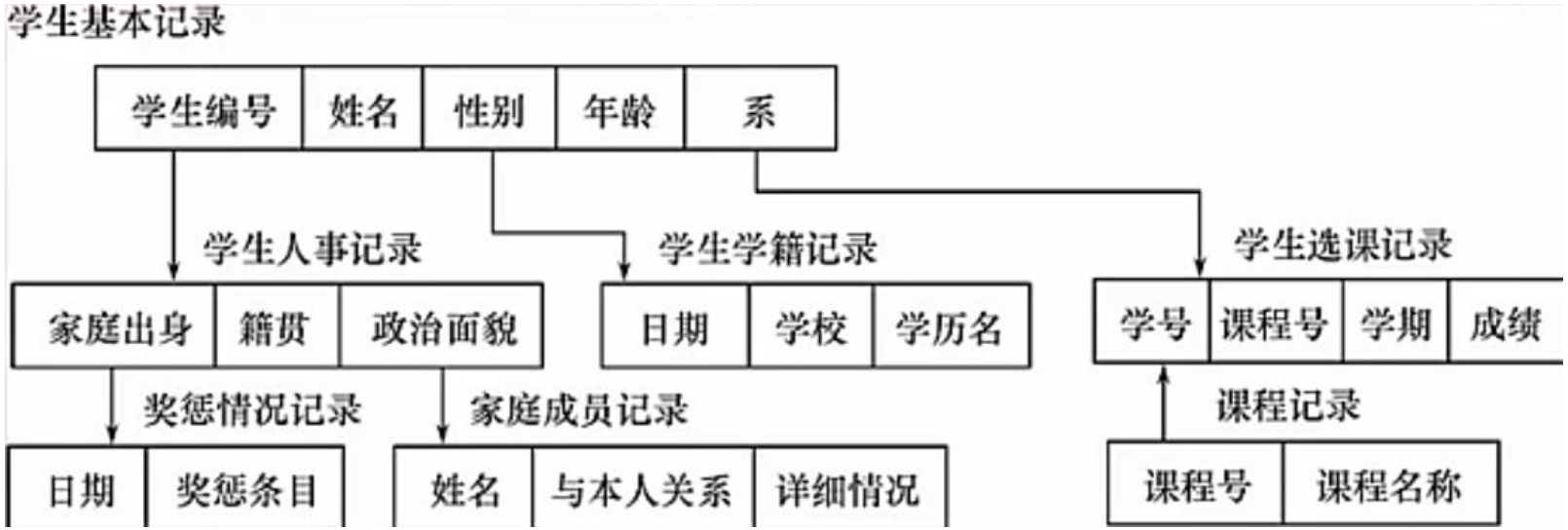
```
INSERT INTO Student
 ( Sno, Sname, Ssex, Major)
VALUES
('801', '张三', '女','计算机');

INSERT INTO Award
 ( Sno, Details)
VALUES
('801', '2013年校奖学金，2015年国
    家奖学金');
```

优点：用户不用关注记录的存储和不同表之间的联系，不要编程，开发速度快。

一、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。
- 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。
  - ✓ 不再仅仅针对某一应用，而是面向整个企业或组织。



二、数据的独立性

物理独立性

- 指用户的应用程序与存储在物理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。

三、数据的高共享性

- 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：
  - ✓ 降低数据的冗余度，节省存储空间。
  - ✓ 避免数据间的不一致性和不相容性。
  - ✓ 数据库系统弹性大，易于扩充。

四、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
  - 保持数据的正确性、有效性、相容性。将数据控制在有效的范围内，保证数据之间满足一定的关系。

四、数据由DBMS统一管理和控制

- 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。

第一章 数据库系统概论

上海交通大學
SHANGHAI JIAO TONG UNIVERSITY

目 录

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

01

两类数据模型

两类数据模型

- 数据模型分为两类（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、关系模型等，它是按计算机系统的观点对数据建模，用于DBMS的实现。
- 物理模型　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，面向计算机系统。

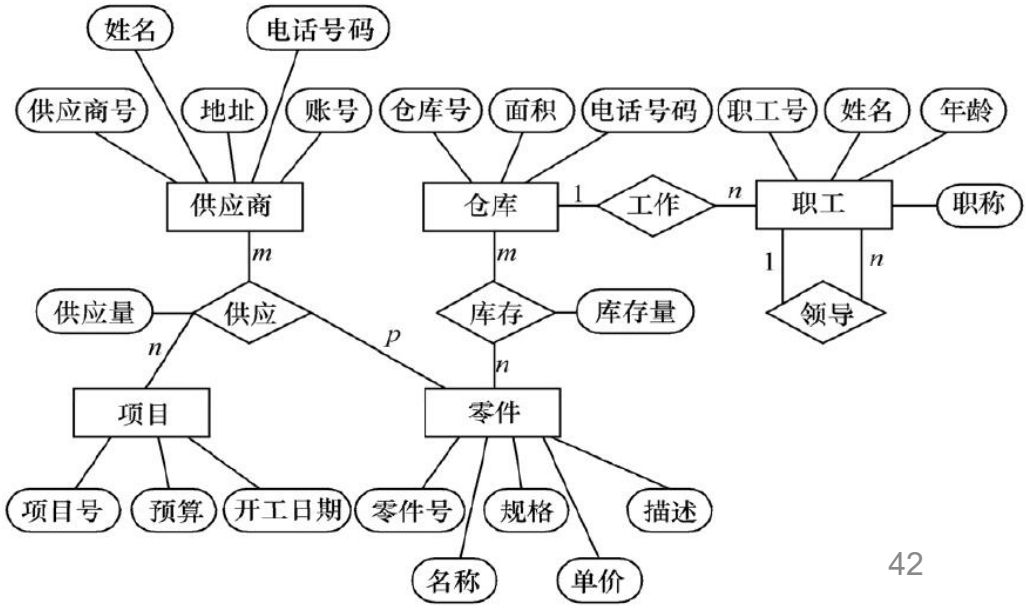
02

概念模型

SJTU

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；
  - 简单、清晰、易于用户理解。



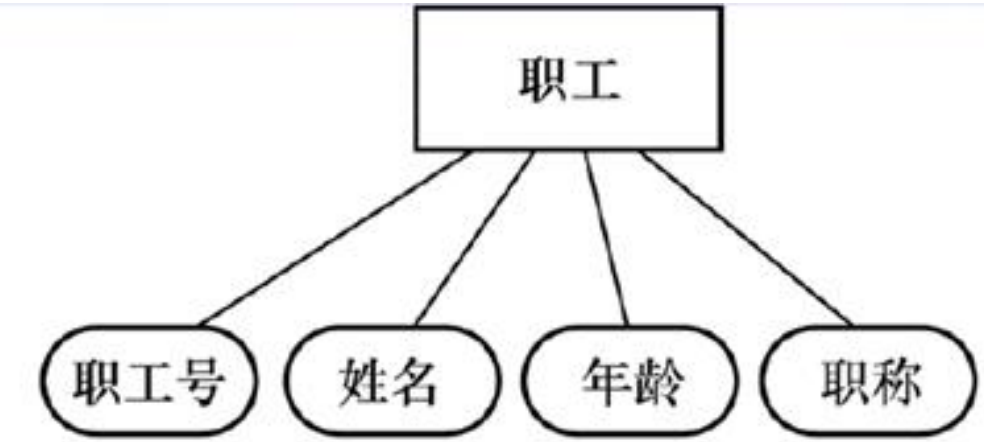
信息世界的基本概念

(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。

信息世界的基本概念

(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）

- ✓同型实体的集合称为实体集

信息世界的基本概念

(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。
- ✓ 实体之间的联系：

两个实体型
三个实体型
一个实体型

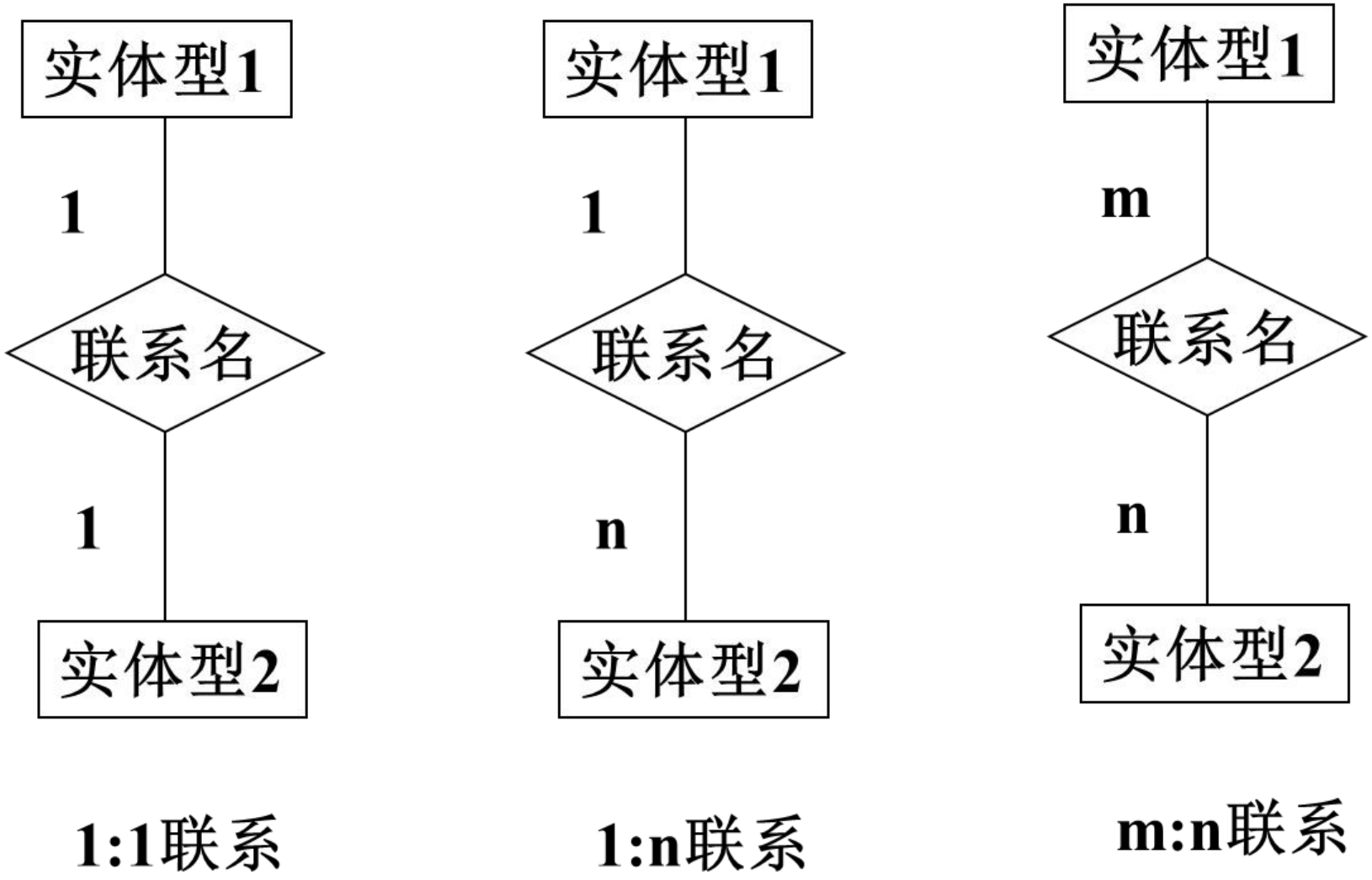
或

一对一联系（1:1）
一对多联系（1:n）
多对多联系（m:n）

实体之间的联系

- 参与联系的实体型的数目，称为**联系的度**
  - 两个实体型之间的联系度为2，称为二元联系；
  - 三个实体型之间的联系度为3，称为三元联系；
  - N个实体型之间的联系度为N，称为N元联系；

两个实体型之间的联系



1:1联系　　1:n联系　　m:n联系

两个实体型之间的联系

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：

  一个班级只有一个正班长

  一个班长只在一个班中任职

1:1联系

两个实体型之间的联系

一对多联系：

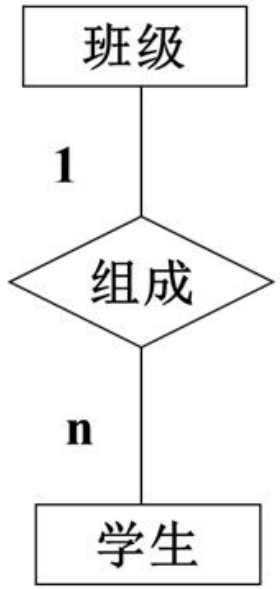
- 如果对于实体集A中的每一个实体，实体集B中有n个实体（n≥0）与之联系，反之，对于实体集B中的每一个实体，实体集A中至多只有一个实体与之联系，则称实体集A与实体集B有一对多联系，记为1:n。

- 实例

  班级与学生之间的联系：

  一个班级中有若干名学生，

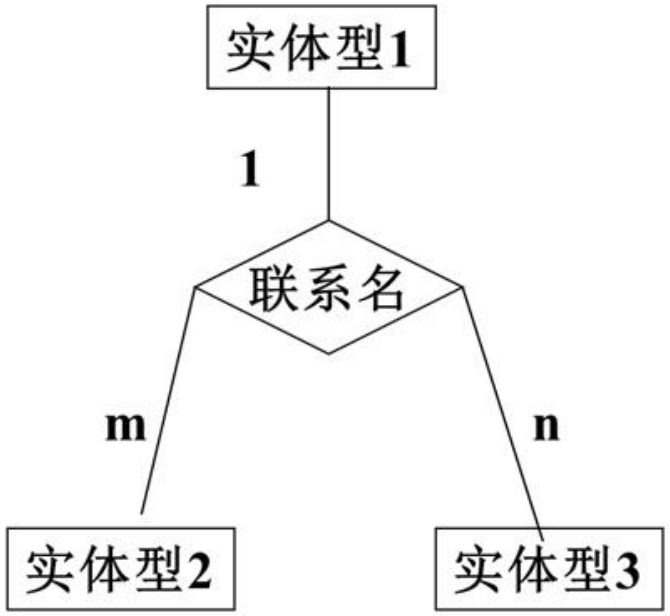
  每个学生只在一个班级中学习



1:n联系

多个实体型之间的联系

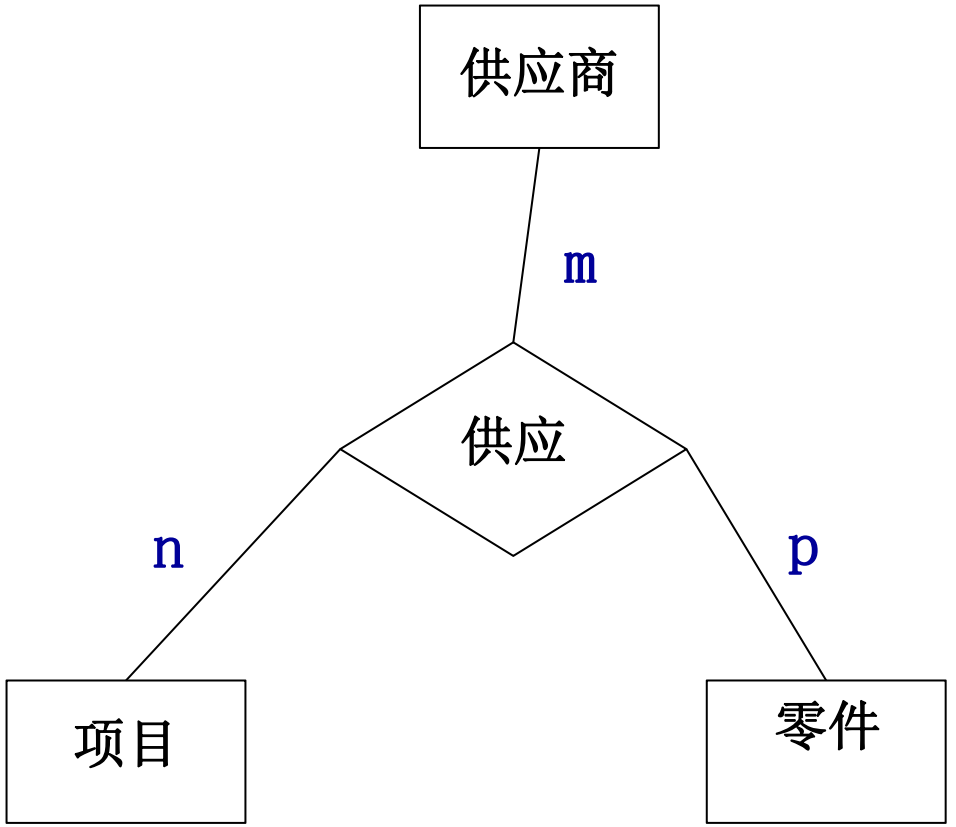
- 多个实体型间的一对多联系：
  - 若实体型E_1，E_2，…，E_n存在联系，对于实体型E_j（j=1，2，…，i-1，i+1，…，n）中的给定实体，最多只和E_i中的一个实体相联系，则我们说E_i与E_1，E_2，…，E_{i-1}，E_{i+1}，…，E_n之间的联系是一对多的。



多个实体型间的1:n联系

多个实体型之间的联系

- 多个实体型间的多对多联系



- 多个实体型间的一对一联系

同一实体集内各实体间的联系

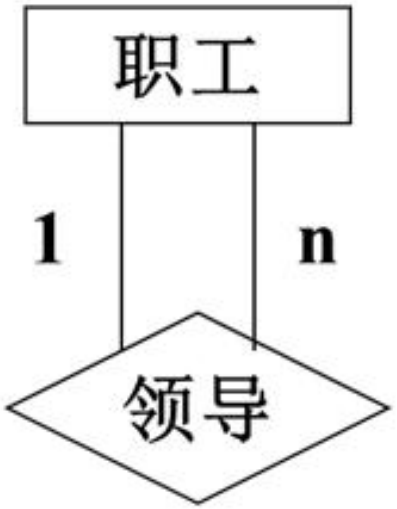
- 一对多联系
  - 实例

    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



同一实体型内部的1:n联系

- 一对一联系
- 多对多联系

概念模型的表示方法

- 概念模型的表示方法很多

- 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
  - ✓ 提供了表示实体型、属性和联系的方法
  - ✓ E-R方法也称为E-R模型

E-R图

实体型

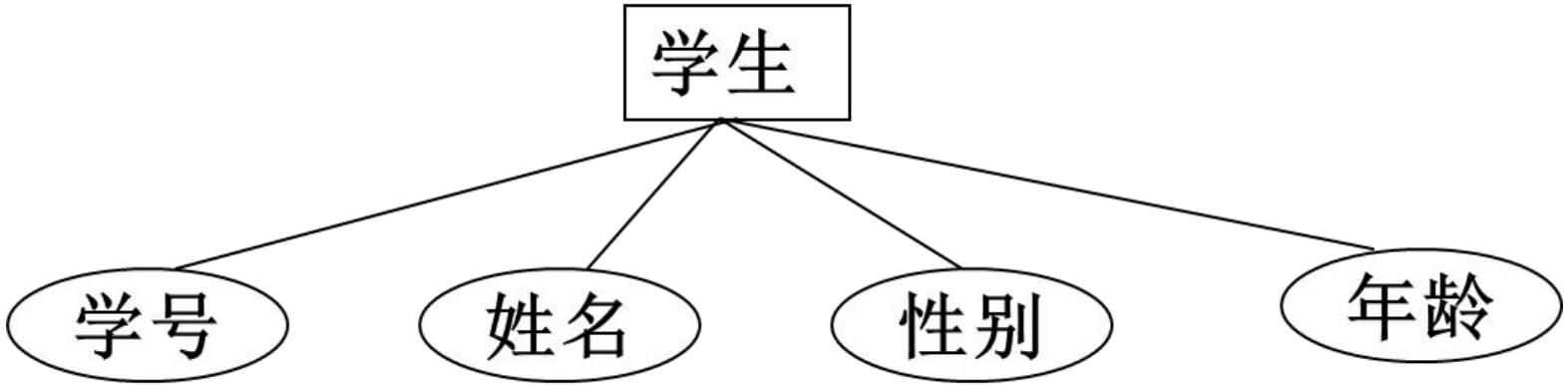
- 用矩形表示，矩形框内写明实体名。

学生　　教师

E-R图

属性

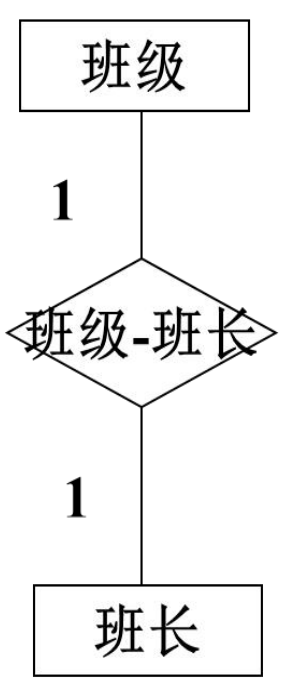
- 用椭圆形表示，并用无向边将其与相应的实体连接起来



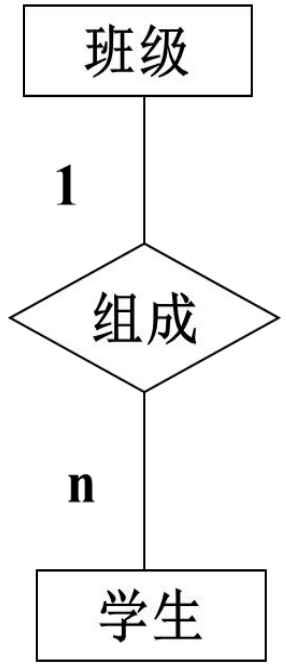
E-R图

联系

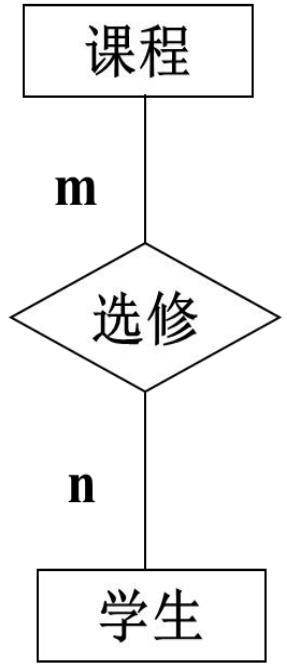
- 联系本身：用菱形表示，菱形框内写明联系名，并用无向边分别与有关实体连接起来，同时在无向边旁标上联系的类型（1:1、1:n或m:n）



1:1联系



1:n联系

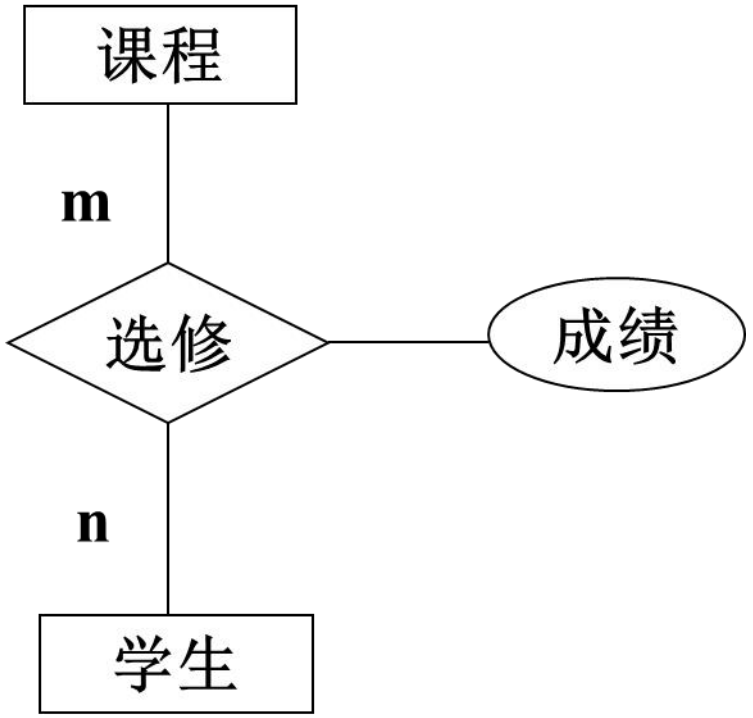


m:n联系

E-R图

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起来。



03

数据模型的组成要素

数据模型的组成要素

- 数据结构 —— 静态特性
- 数据操作 —— 动态特性
- 数据的完整性约束条件

一、数据结构

- 什么是数据结构？
  - 描述数据库的组成对象及对象之间的联系。
  - 经常用数据结构的类型来命名数据模型：

    例：层次结构—层次模型；关系结构—关系模型
- 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）
- 数据操作是对系统动态特性的描述。

三、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；
- 完整性规则可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；

数据的约束条件

数据模型对约束条件的定义

- 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。

04

常用的数据模型

SJTU

常用的逻辑数据模型

格式化模型（第一代数据库）

- 层次模型（Hierarchical Model）
- 网状模型(Network Model)
- 数据结构：以基本层次联系为基本单位。
- 典型事件：
  - ✓ IBM公司的IMS系统（层次模型） —— 1968年
  - ✓ 美国通用电气公司的IDS系统（网状模型） —— 1961年
  - ✓ DBTG报告(数据库标准) —— 1969年
  - ✓ 1973年，网状数据库之父 Charles W. Bachman 获图灵奖

常用的逻辑数据模型

关系模型(Relational Model)（第二代数据库）

- 数据结构：表
- 典型事件：
  - ✓ 1970年，Codd发表关系模型文章；
  - ✓ 1981年,关系数据库之父 Edgar F. Codd 获图灵奖

常用的逻辑数据模型

新一代数据库

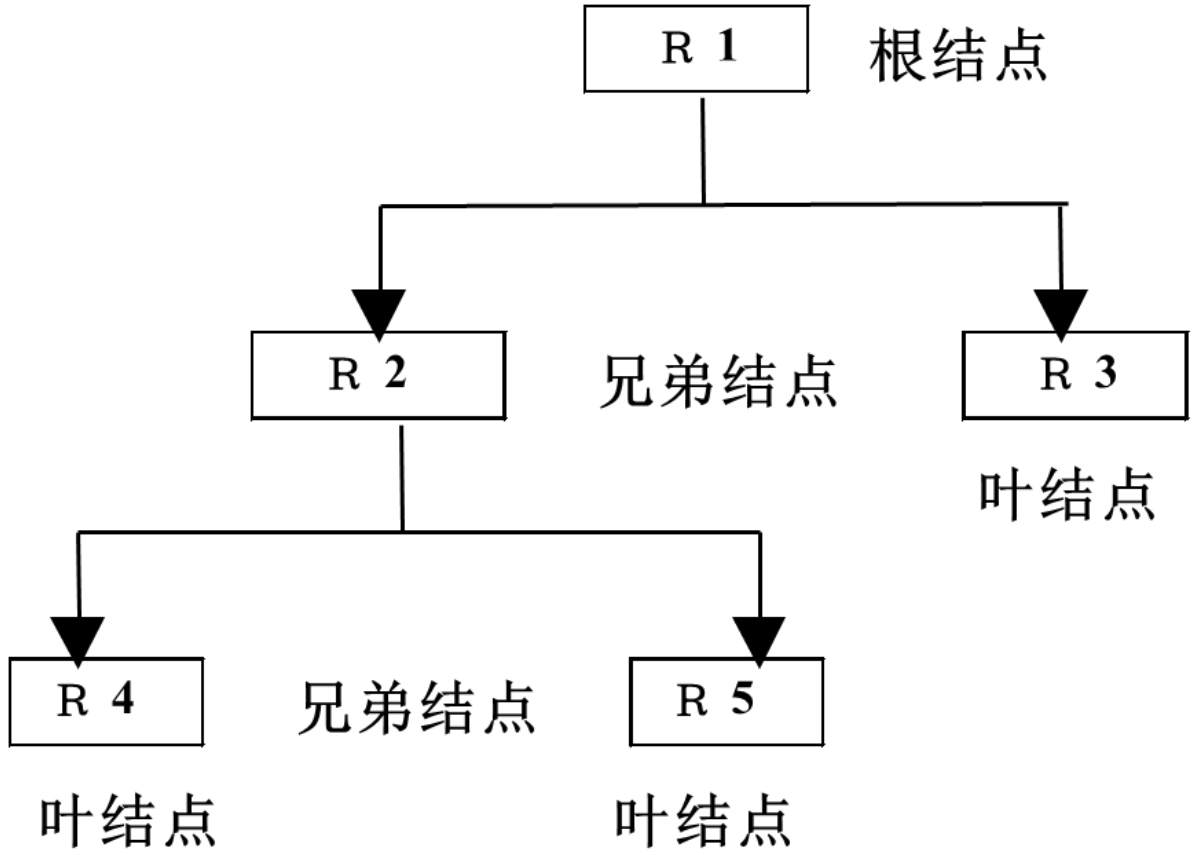
- 面向对象模型(Object Oriented Data Model)
- 对象关系模型(Object Relational Model)
- 半结构化数据模型 (Semi-structure Data Model)
- 非结构化数据模型
- 图模型
- 其他

05

层次模型

层次数据模型的数据结构

- 层次模型中的几个术语
  - 双亲结点，子女结点
  - 根结点，叶结点
  - 兄弟结点

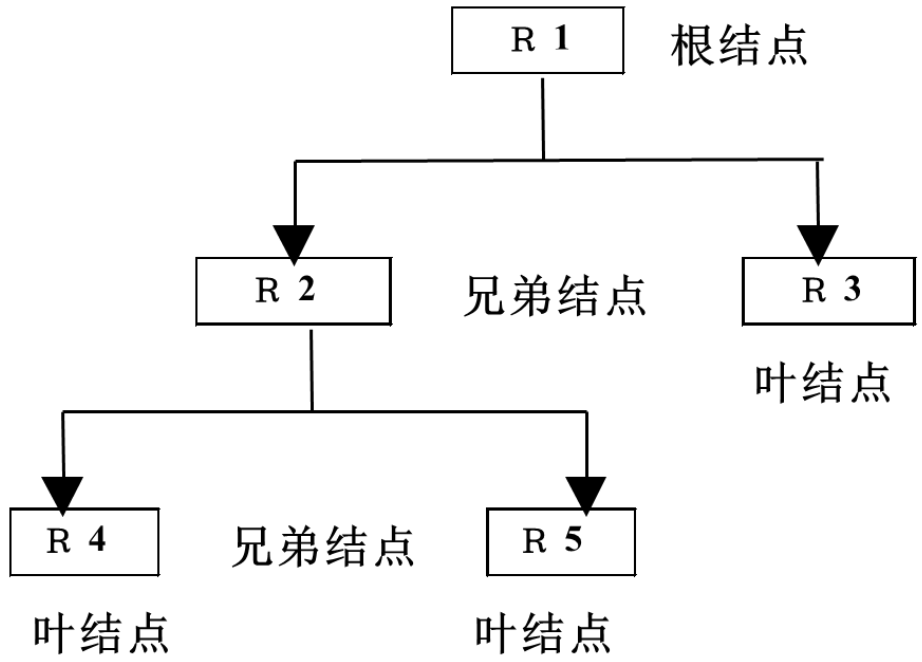


层次数据模型的数据结构

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。

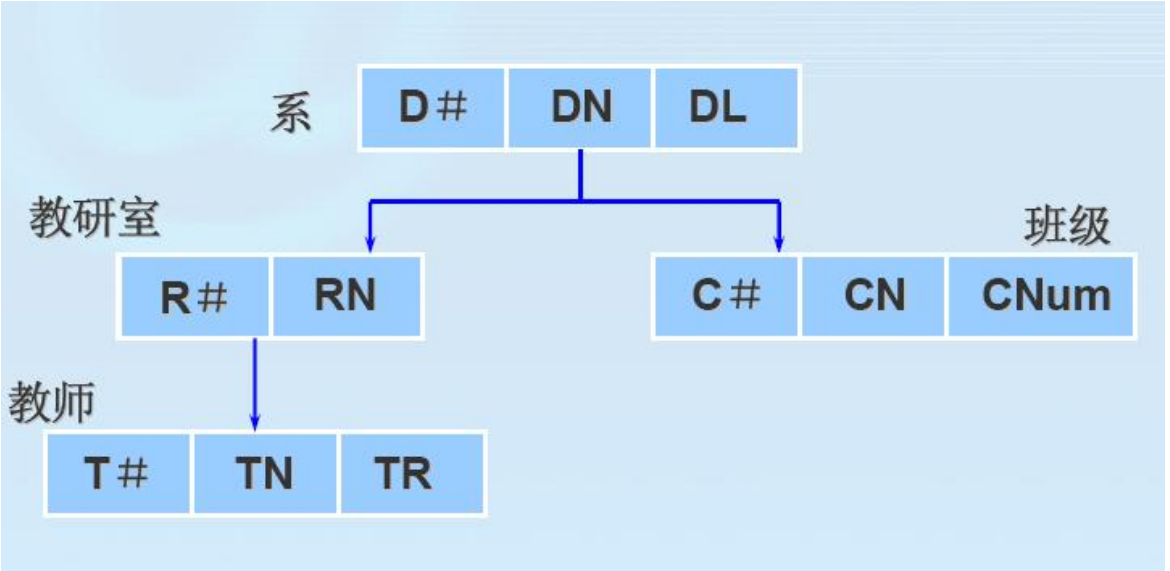


层次数据模型的数据结构

- 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（有向边）表示记录（类型）之间的一对多的父子联系。

层次数据模型的数据结构

- 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；

层次数据模型的数据结构

- 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

层次数据模型的数据操纵

- 查询
- 插入
- 删除
- 更新

层次数据模型的完整性约束

- 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；

层次数据模型的优缺点

- 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；
  - 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；
  - 查询子女结点必须通过双亲结点；
  - 查询及更新操作必须给出完整路径；

06

网状模型

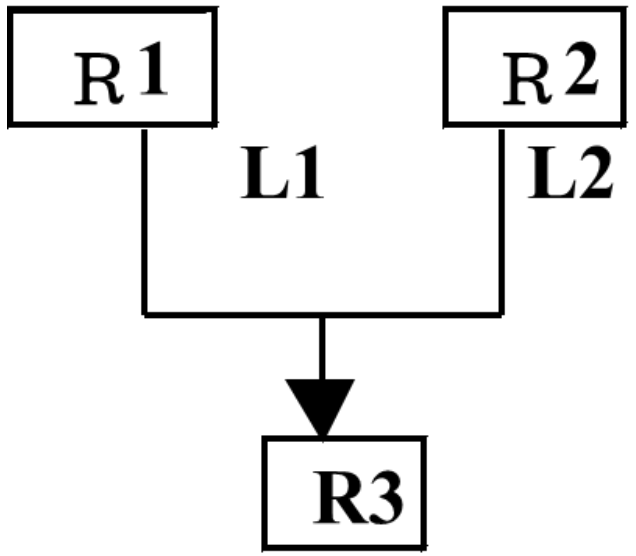
SJTU

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：

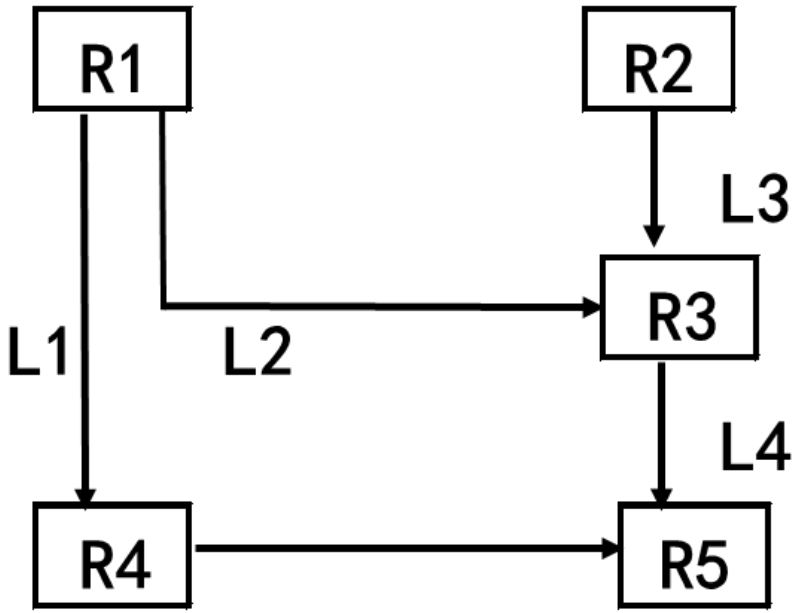
1. 允许一个以上的结点无双亲；



网状数据模型的数据结构

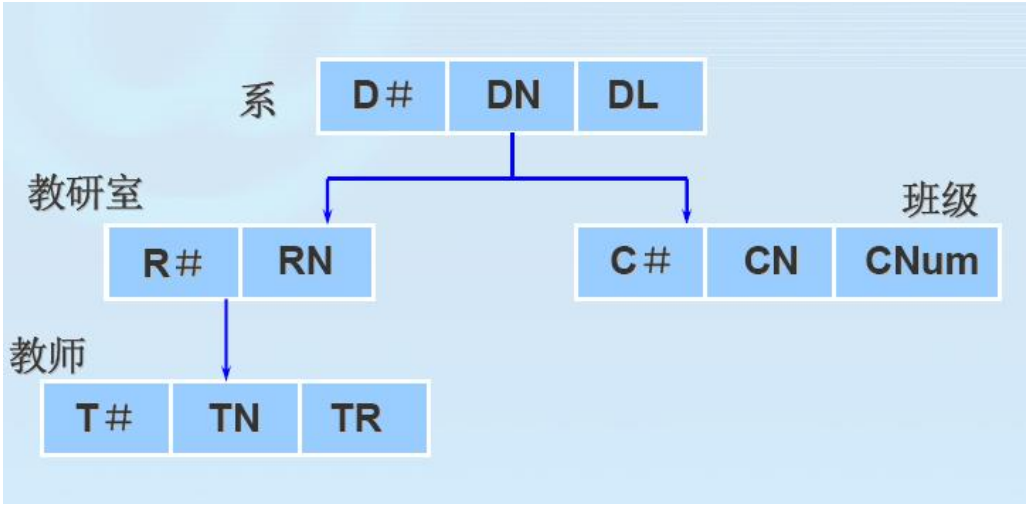
- 网状模型

2. 一个结点可以有多于一个的双亲；



网状数据模型的数据结构

- 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。

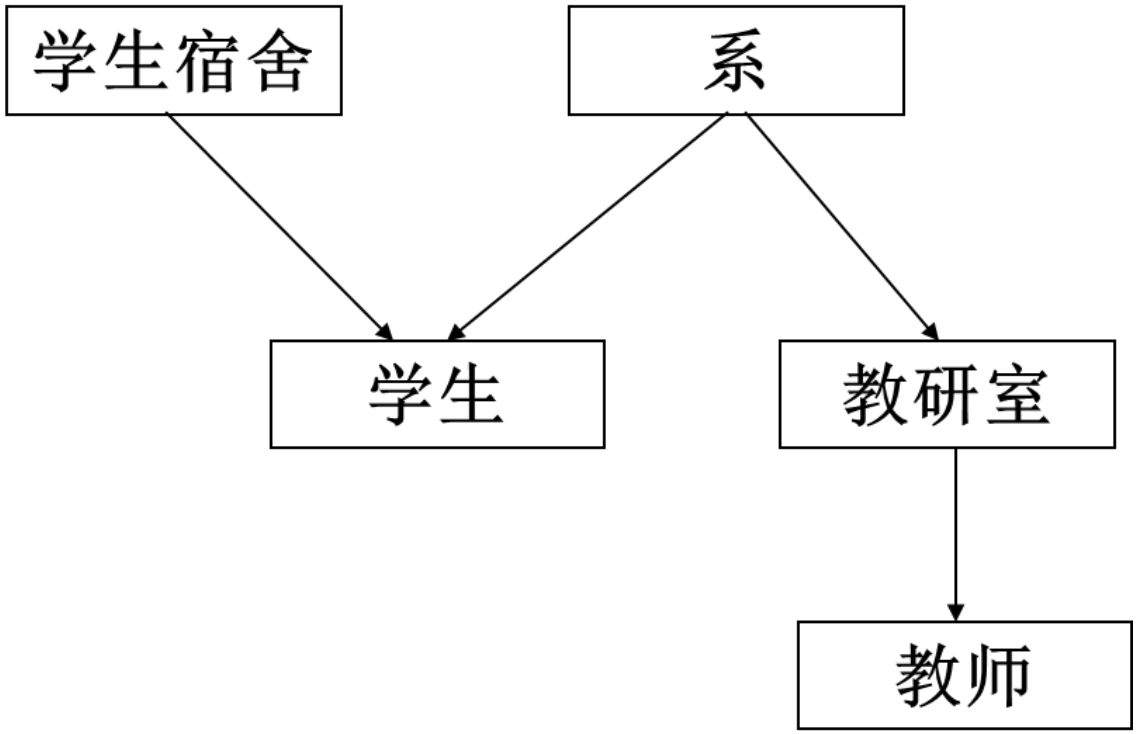
每个记录类型可包含若干个字段。

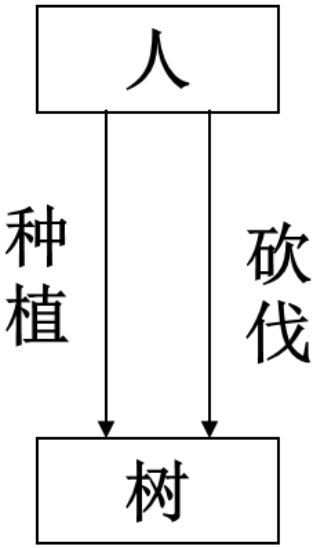
✓联系：用结点之间的连线表示记录（类）型之间的一对多的父子联系。

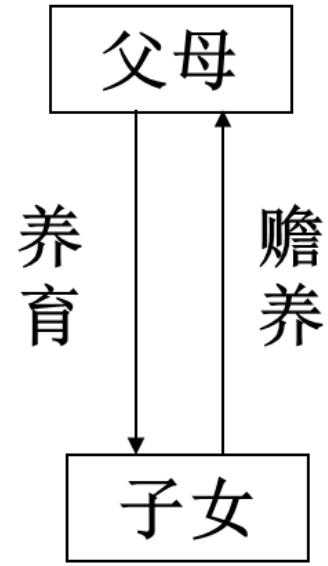
网状模型与层次模型的区别

- 网状模型允许多个结点没有双亲结点，允许一个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







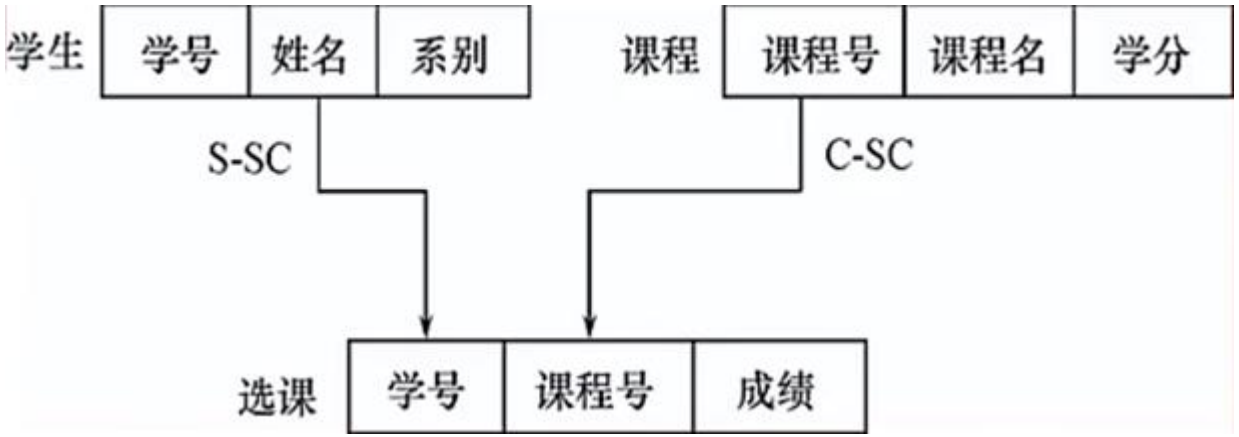
网状数据模型的数据结构

- 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；

引进一个学生选课的联结记录

选课（学号，课程号，成绩）



网状数据模型的数据操纵

- 查询
- 插入
- 删除
- 更新

网状数据模型的完整性约束

- 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。
  - ✓ 支持记录码的概念（唯一标识记录的数据项集合）；
  - ✓ 双亲结点与子女结点之间是一对多联系；
  - ✓ 可以支持属籍类别：
    - 加入类别：双亲记录在，子女记录才可以加入；
    - 移出类别：双亲记录删除，子女记录删除；

网状数据模型的优缺点

- 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；
- 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；

07

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　令	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
…	…	…	…	…	…

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。

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，年龄，性别，系，年级）

关系模型的数据结构

- 关系模型建立在集合代数的基础上；

学 号	姓 名	性 别	专 业 号	年 龄
801	张 三	女	01	19
802	李 四	男	01	20
803	王 五	男	01	20
804	赵 六	女	02	20
805	钱 七	男	02	19

学生（学号，姓名，性别，专业号，年龄）

关系模型的数据结构

- 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。

    例：学生（学号，姓名，年龄，性别，系号，年级）

    课程（课程号，课程名，学分）
  - 一对一联系：隐含在实体对应的关系中。

    例：班级（班级号，班级人数，班长学号）
  - 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。

    例：选修（学号，课程号，成绩）

关系模型的数据结构

- 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！

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	115.5
.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.

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”；

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

关系模型的存储结构

- 实体及实体间的联系用表来表示；
- 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，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

关系模型的优缺点

- 缺点：
  - 存取路径对用户隐蔽，导致查询效率往往不如格式化数据模型；
  - 为提高性能，必须对用户的查询请求进行优化，增加了开发数据库管理系统的难度；

第一章 数据库系统概论

上海交通大學
SHANGHAI JIAO TONG UNIVERSITY

目 录

数据库系统的结构

- 从数据库应用开发人员角度：
  - 数据库采用三级模式结构，是数据库系统内部的系统结构。
- 从数据库最终用户角度：
  - 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

01

数据库系统模式的概念

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）

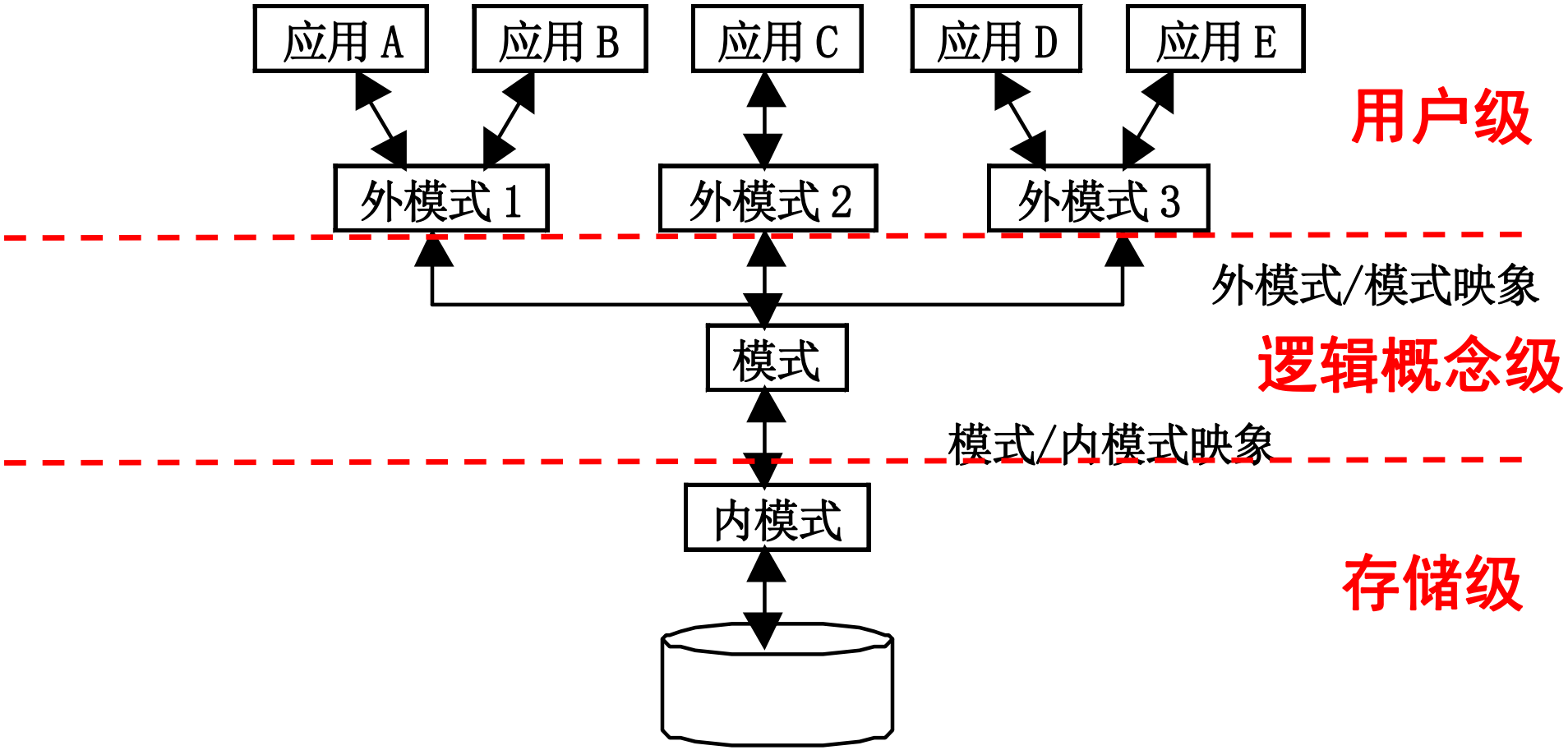
数据库系统模式的概念

- 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的更新而变动；

02

数据库系统的三级模式结构

数据库系统的三级模式结构



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；
- 一个应用数据库只有一个模式，以数据模型为基础；
- 模式的地位：是数据库系统模式结构的中心
  - 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（模式DDL，模式描述语言）
  - 定义数据的逻辑结构（数据项的名字、类型、取值范围等）
  - 定义数据之间的联系
  - 定义与数据有关的安全性、完整性要求

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（External Schema）

- 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（External Schema）

- 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，简化用户视图；

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（堆存储，聚簇存储，属性升降存储）
    - ✓ 索引的组织方式（按照B+树索引？按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（定长？变长？）
- 一个数据库只有一个内模式；

03

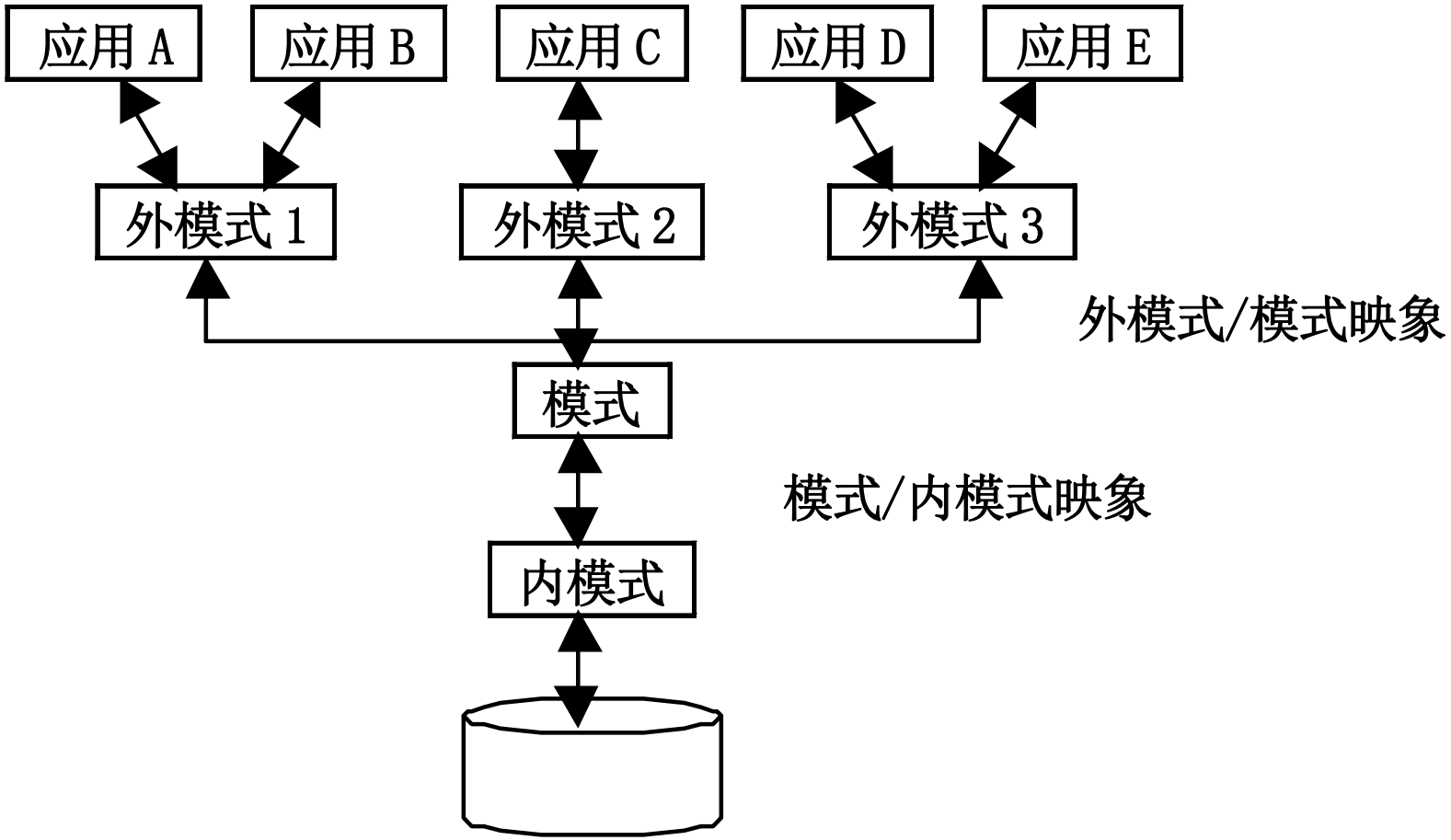
数据库的二级映像功能与数据独立性

SJTU

三级模式与二级映象

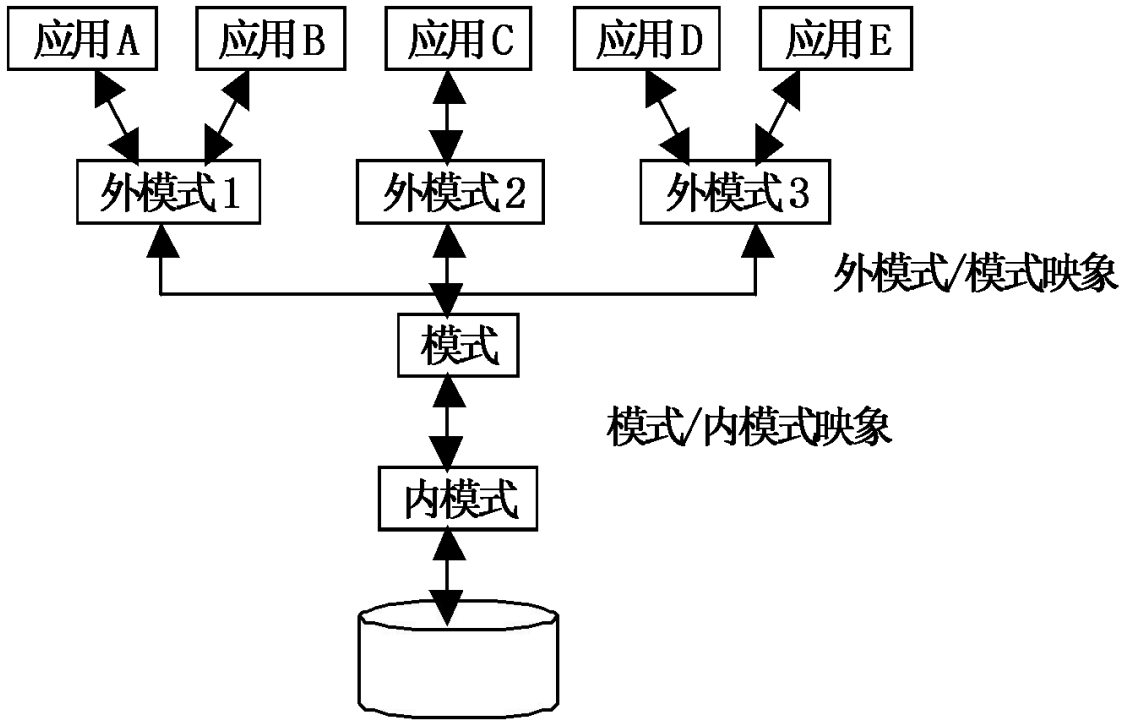
- 三级模式是对数据的三个抽象级别；
- 二级映象，在DBMS内部实现这三个抽象层次的联系和转换；
  - 外模式/模式映像
  - 模式/内模式映像

数据库系统的三级模式结构



外模式/模式映象

- 定义外模式（局部逻辑结构）与模式（全局逻辑结构）之间的对应关系；
- 每一个外模式都对应一个外模式/模式映象；
- 映象定义通常包含在各自外模式的描述中；

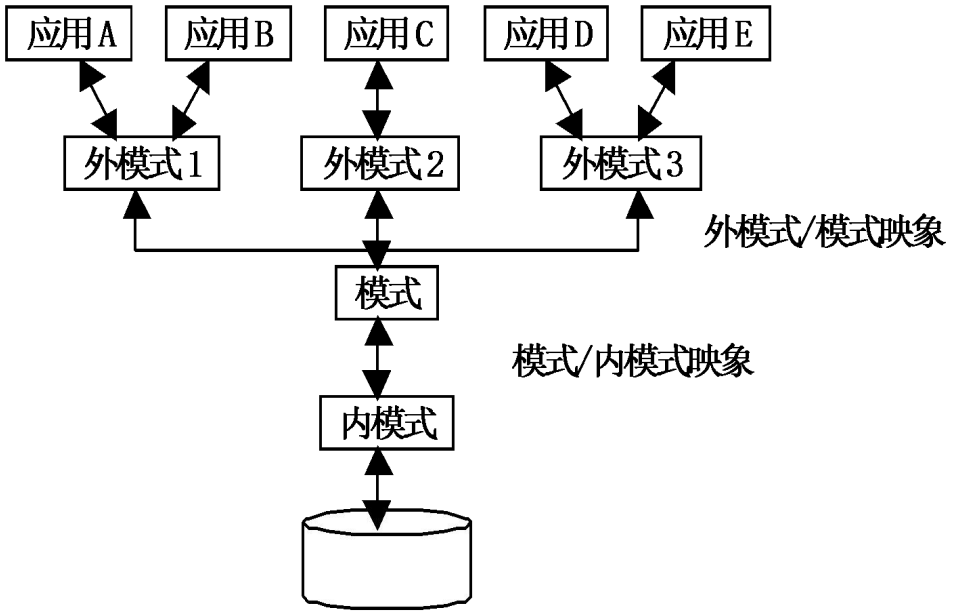


外模式 / 模式映象的用途

- 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；

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，说明某个逻辑记录和字段在内部是如何表示的。
- 数据库中模式/内模式映象是唯一的；
- 该映象定义通常包含在模式描述中；



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；

第一章 数据库系统概论

上海交通大學
SHANGHAI JIAO TONG UNIVERSITY

目 录

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
- （用户）

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
- 内存数据库

硬件平台及数据库

- 数据库系统对硬件资源的要求

(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

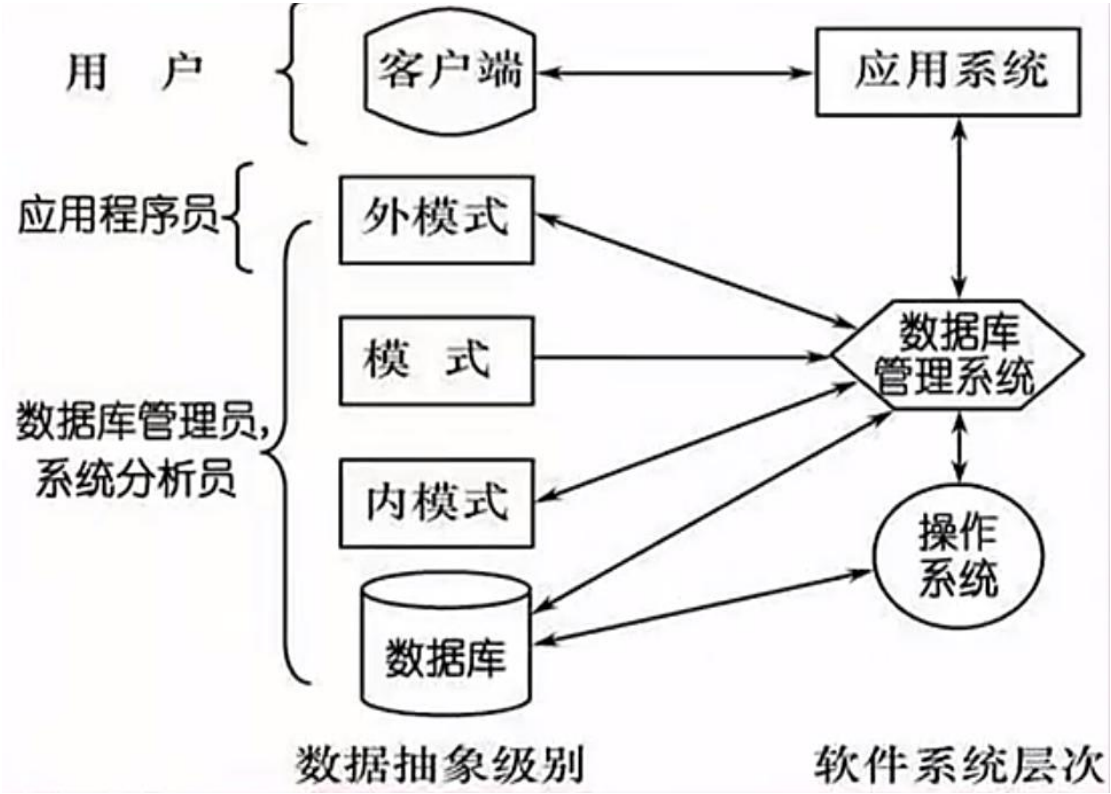
(3) 较高的通道能力，提高数据传送率

软件

- DBMS（数据库管理系统）
- 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

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



注：不同人员涉及不同的数据抽象级别，具有不同的数据视图

数据库管理员(DBA)

- 确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；

数据库管理员(DBA)

- 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

数据库管理员(DBA)

- 数据库的改进和重组
  - 性能监控和调优
  - 定期对数据库进行重组，提高系统性能
- 数据库重构（需求增加或改变）

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；

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；

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；

用户

- 偶然用户
  - 不经常访问数据库，每次访问需要不同数据库信息
  - 企业或组织机构的高中级管理人员
- 简单用户
  - 主要工作是查询和更新数据库
  - 银行的职员、机票预定人员、旅馆总台服务员

用户

- 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

上海交通大學
SHANGHAI JIAO TONG UNIVERSITY

THANK YOU !

饮水思源 爱国荣校