

数据库系统

4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

数据

数据库中存储的基本对象 是描述事物的符号记录

数据库

是长期存储在计算机内，有组织的，可共享的大量数据的集合

特征

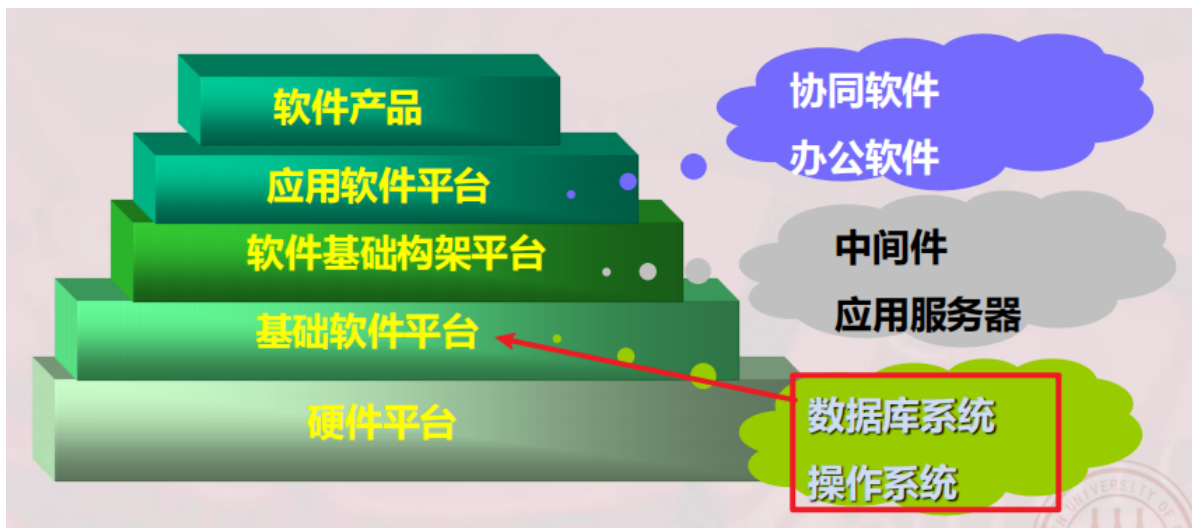
- 数据按一定的数据模型组织，描述和储存
- 可为各种用户共享
- 数据独立性高
- 易扩展

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

是基础软件，是一个大型复杂的软件系统

用于科学地组织和存储数据，高效地获取和维护数据

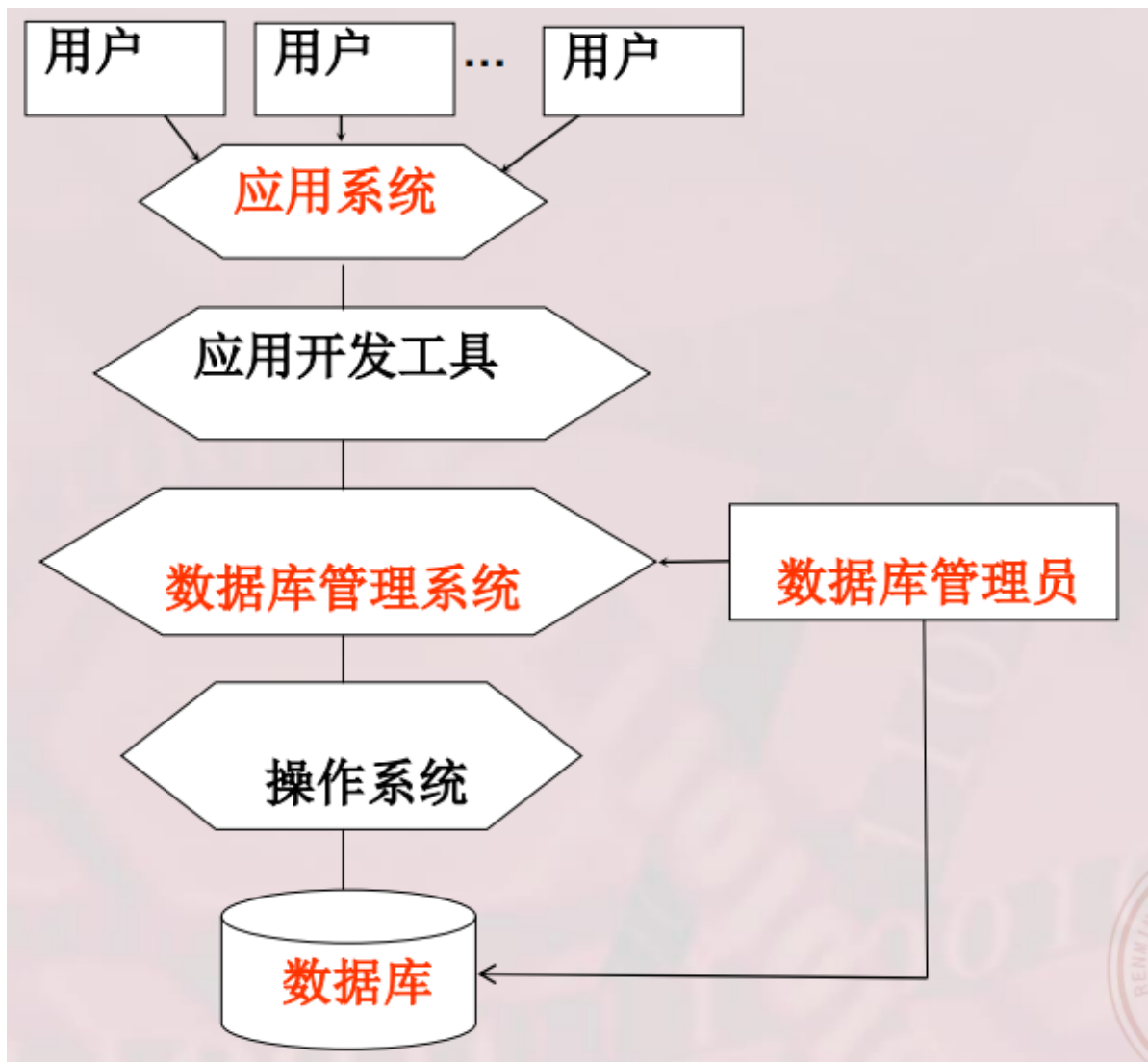


数据库管理系统的主要功能

- 数据定义功能
提供数据定义语言 (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
- 数据操纵功能
- 数据库的事务管理和运行管理
- 数据库的建立和维护功能
- 其他功能

数据库系统

由 数据库，数据库管理系统（及其应用开发工具），应用程序，数据库管理员



数据库管理技术

数据管理

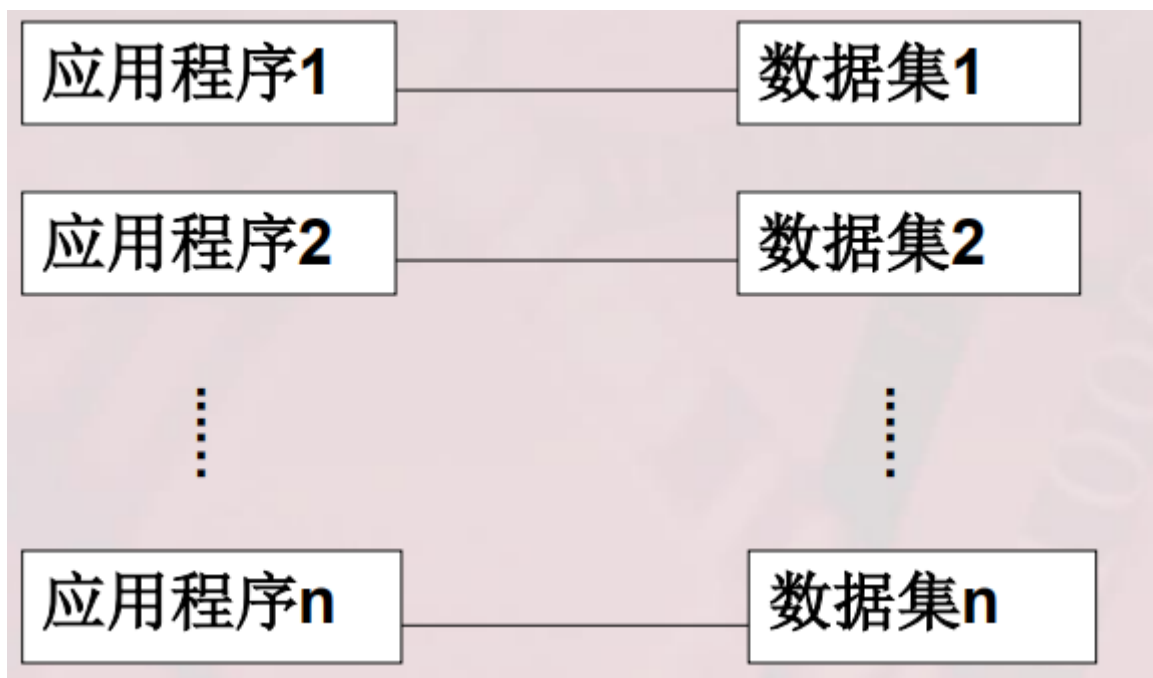
对数据进行分类，组织，编码，存储，检索和维护

数据处理的中心问题

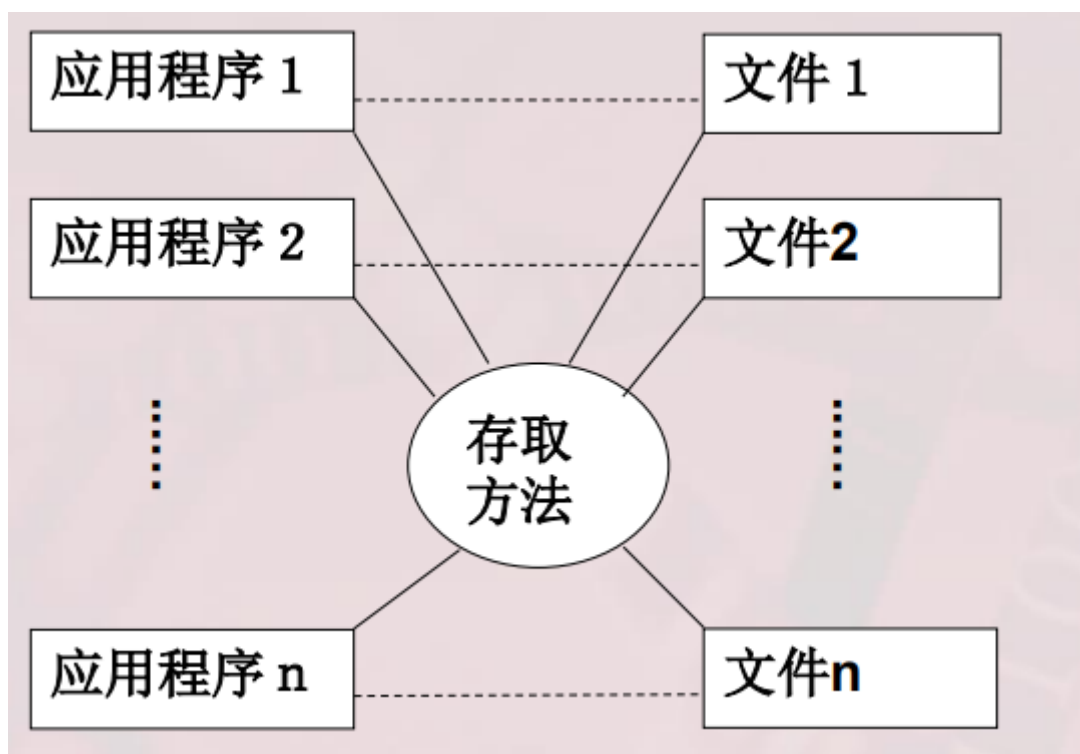
发展过程

1. 人工管理阶段
2. 文件系统阶段
3. 数据库系统阶段

人工管理阶段



文件系统阶段



数据库系统阶段

- 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

| | | 人工管理阶段 | 文件系统阶段 | 数据库系统阶段 |
|----|---------|-------------|--------------|----------------------------|
| 背景 | 应用背景 | 科学计算 | 科学计算、管理 | 大规模数据管理 |
| | 硬件背景 | 无直接存取存储设备 | 磁盘、磁鼓 | 大容量磁盘、磁盘阵列 |
| | 软件背景 | 无操作系统 | 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 |
| 特点 | 数据的管理者 | 用户(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(一个企业、跨国公司) |
| | 数据的共享程度 | 无共享，冗余度极大 | 共享性差，冗余度大 | 共享性高，冗余度小 |
| | 数据的独立性 | 不独立，完全依赖于程序 | 独立性差 | 具有高度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DBMS提供数据安全性、完整性、并发控制和恢复能力 |

数据库系统的特点

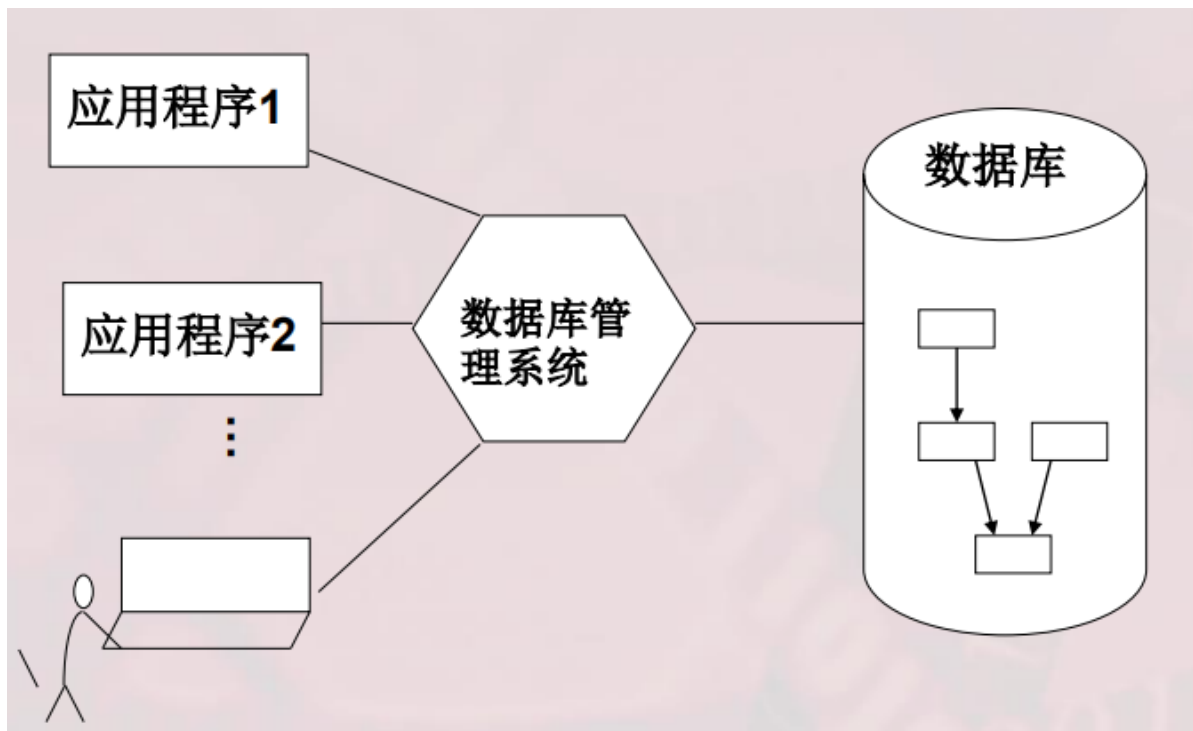
- 数据整体结构化
- 数据的共享性高，冗余度低且易扩充
- 数据独立性高
- 数据由数据库管理系统统一管理和控制

数据整体结构化

- 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，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) 检查保证数据的正确性、有效性和相容性
- (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



数据模型

数据模型是对现实世界数据特征的抽象。

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

数据模型是数据库系统的核心和基础

数据模型分类

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。

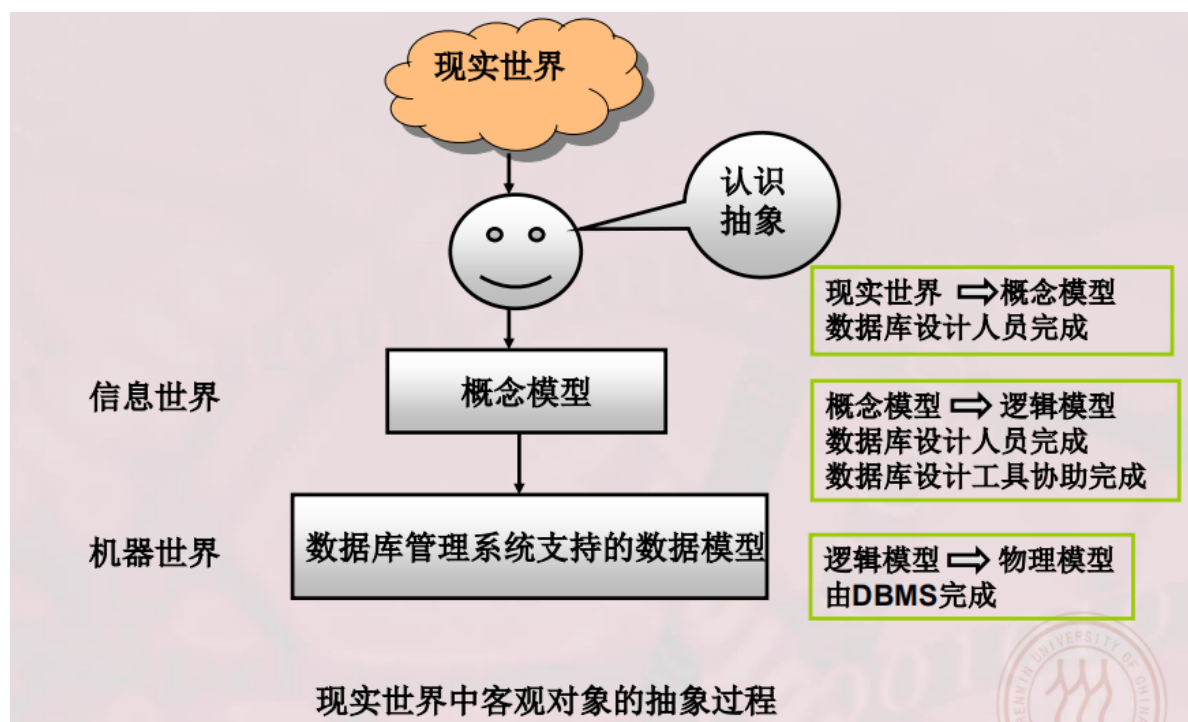
逻辑模型和物理模型

逻辑模型主要包括网状模型、层次模型、关系模型、面向对象数据模型、对象关系数据模型、半结构化数据模型等。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

两步抽象

现实世界抽象为信息世界 信息世界转换为机器世界



概念模型的一种表达方法：**实体-联系方法**

E-R图 也称 E-R 模型

信息世界的基本概念

实体

客观存在并可相互区别的十五称为实体

可以是具体的人，事，物，或是抽象的概念

属性

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

码

唯一标识实体的属性集称为码

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

实体集

同一类型实体的集合称为实体集

联系

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

数据模型的组成要素

数据结构

描述数据库的组成对象，以及对象之间的联系

数据结构是对系统静态特性的描述

数据操作

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

类型：增删改查

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

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依存规则
- 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、有效和相容

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通用的完整性约束条件
- 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

常用的数据模型

层次模型

网状模型

关系模型

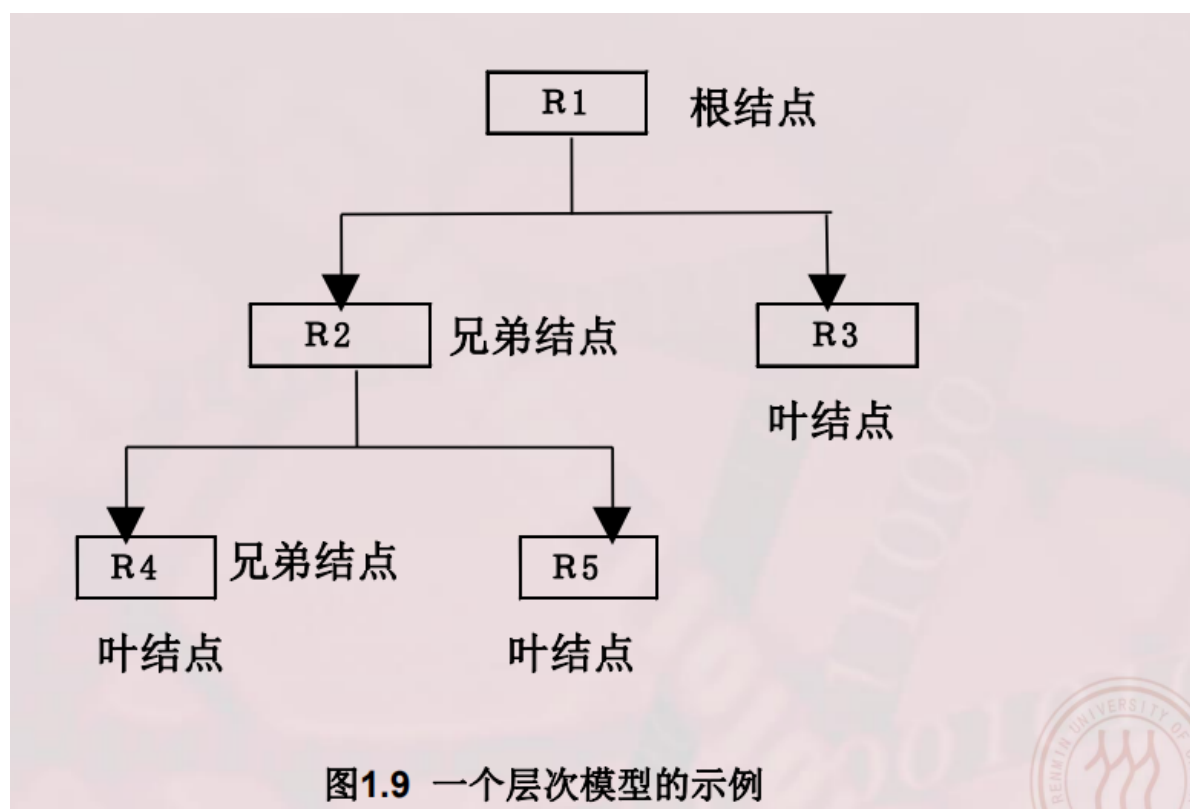
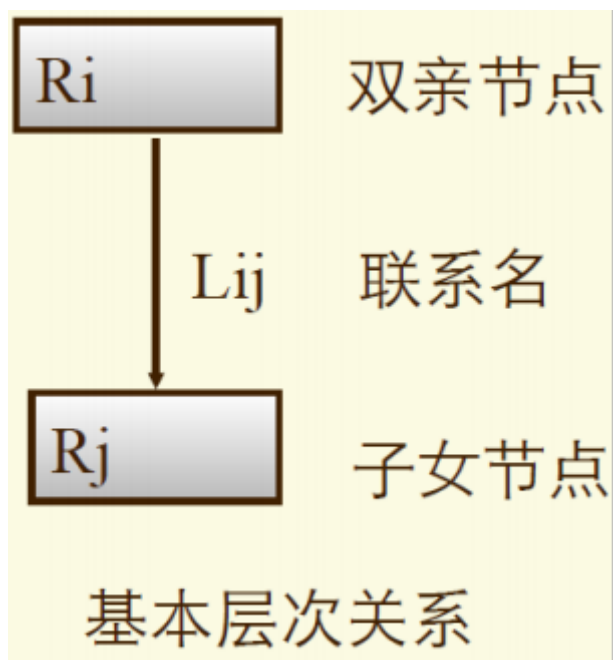
层次模型

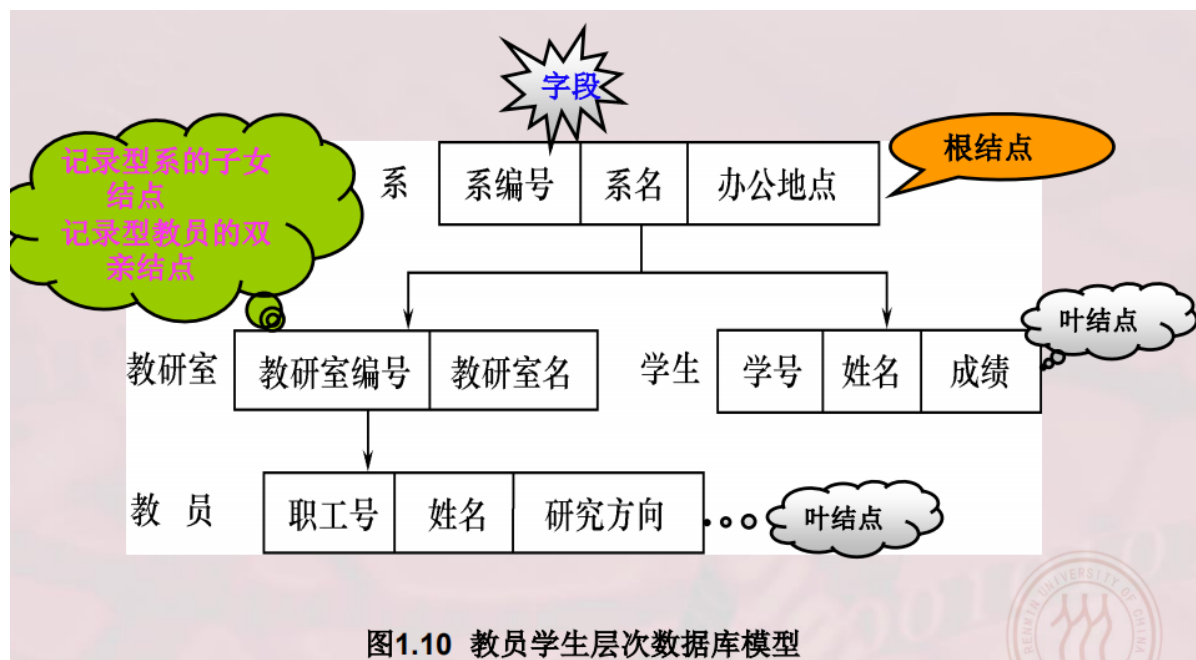
最早出现的数据模型

用树形结构来标识各类实体以及实体间的联系

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





- 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
- 更新操作时，应更新所有相应记录，以保证数据的一致性

优点

- 层次模型的数据结构比较简单清晰且查询效率高，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
- 结点之间的多对多联系表示不自然且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且层次命令趋于程序化

网状模型

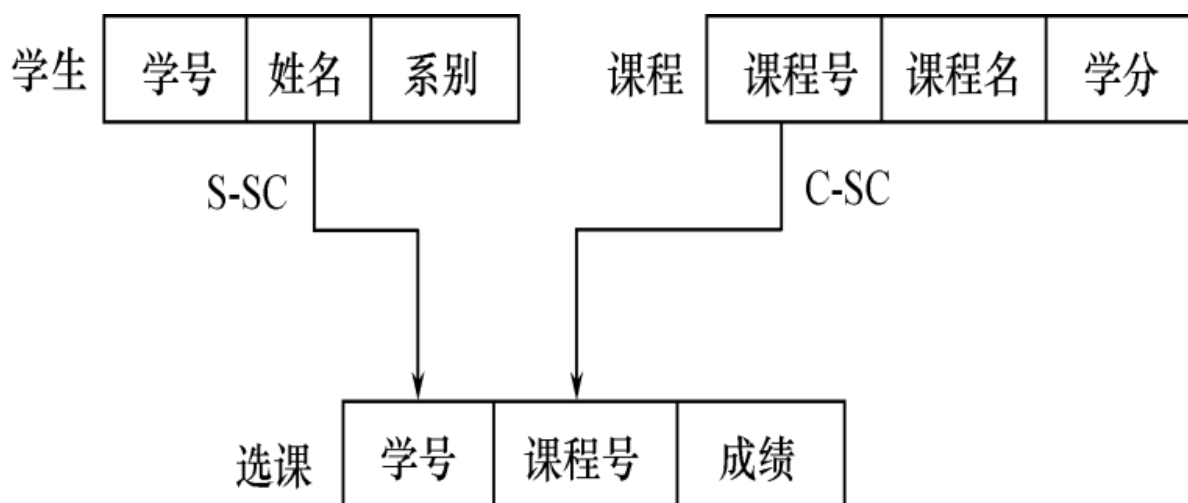
满足下面两个条件的基本层次联系的集合：

- 允许一个以上的结点无双亲
- 一个结点可以有多于一个的双亲

子女节点与双亲节点的联系可以不唯一

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



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
- 记录之间联系是通过存取路径实现的，用户必须了解系统结构的细节

关系模型

一张二维表，由行和列组成

- 关系：一个关系对应一张表
- 元组：表中的一行即为一个元组
- 属性：表中的一列即为一个属性，给每个属性起一个名称即为属性名
- 主码：也称码键。表中的某个属性组，它可以唯一确定一个元组
- 域：是一组具有相同数据类型的值的集合
- 分量：元组中的一个属性值
- 关系模式：对关系的描述

关系的规范条件

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1.15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 |
|-------|----|----|------|------|------|-----|-------|------|
| | | | 基本工资 | 岗位津贴 | 业绩津贴 | 三险 | 个人所得税 | |
| 86051 | 陈平 | 讲师 | 1305 | 1200 | 1850 | 160 | 112 | 4083 |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |

数据操作是集合操作，操作对象和操作结果都是关系

关系的完整性约束条件

- 实体完整性
- 参展完整性
- 用户定义的完整性

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

缺点

- 存取路径对用户透明，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

数据库系统的组成

从数据库应用开发人员角度看，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

模式概念

型 Type

- 对某一类数据的结构和属性的说明 数据类型

值 Value

- 是型的一个具体赋值

模式 (Schema)

- 数据库逻辑结构和特征的描述它是型的描述，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

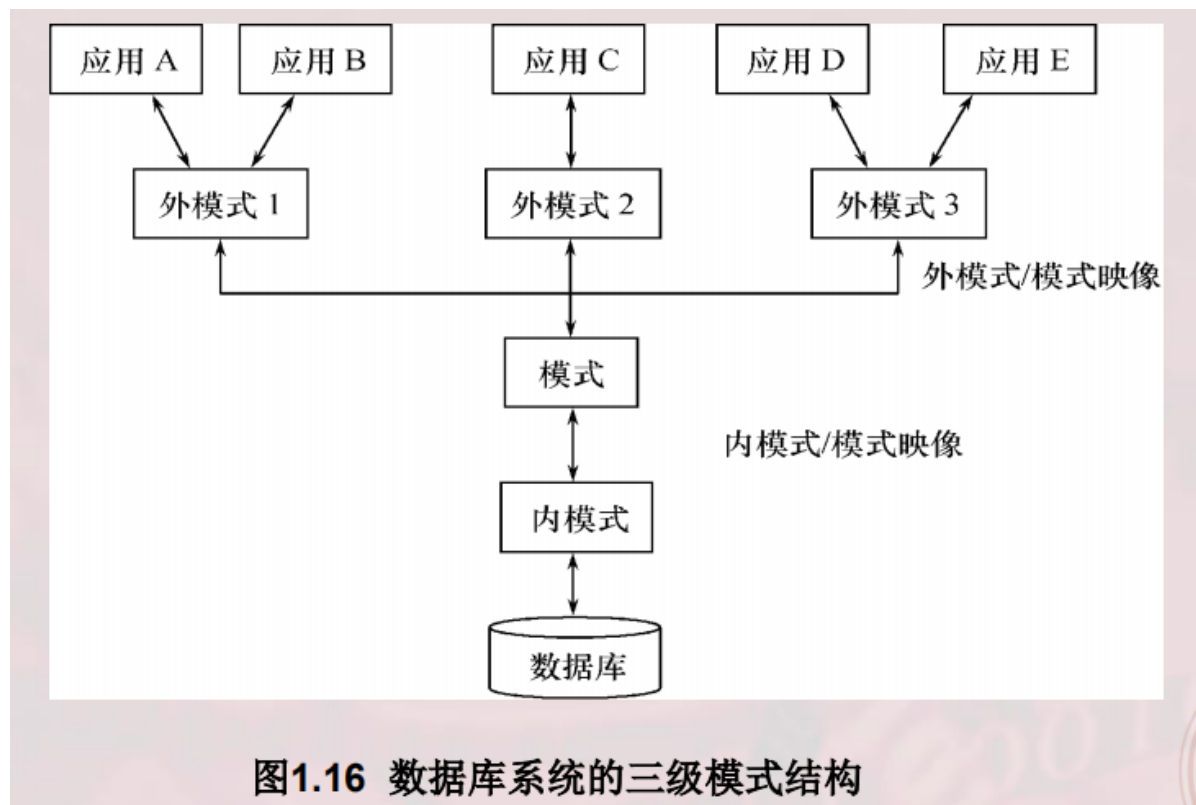
实例 (Instance)

- 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

- 实例随数据库中的数据更新而变动

三级模式结构

- 模式
- 外模式
- 内模式



模式

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

外模式（子模式或用户模式）

地位：介于模式与应用之间

- 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

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

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（例如，顺序存储，按照B树结构存储，按hash方法存储等）
- 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

三级模式是对数据的三个抽象级别

二级映像在数据库管理系统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

外模式--模式映像

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

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/内模式映像是唯一的
- 该映像定义通常包含在模式描述中
- 保证数据的物理独立性
- 数据库模式