数据（Data）

描述⻝⾷食物的符号记录

数字，⽂文字，图形，图像，⾳音频，视频等等

数据的含义成为数据的语义，数据与其语义是不可分的记录是计算机存储数据的⼀种格式或⼀种⽅法。

数据有结构

数据库（Database，DB）

数据库是⻓期存储在计算机内、有组织的，可共享的⼤量量数据的集合。 为什什么要建⽴数据库

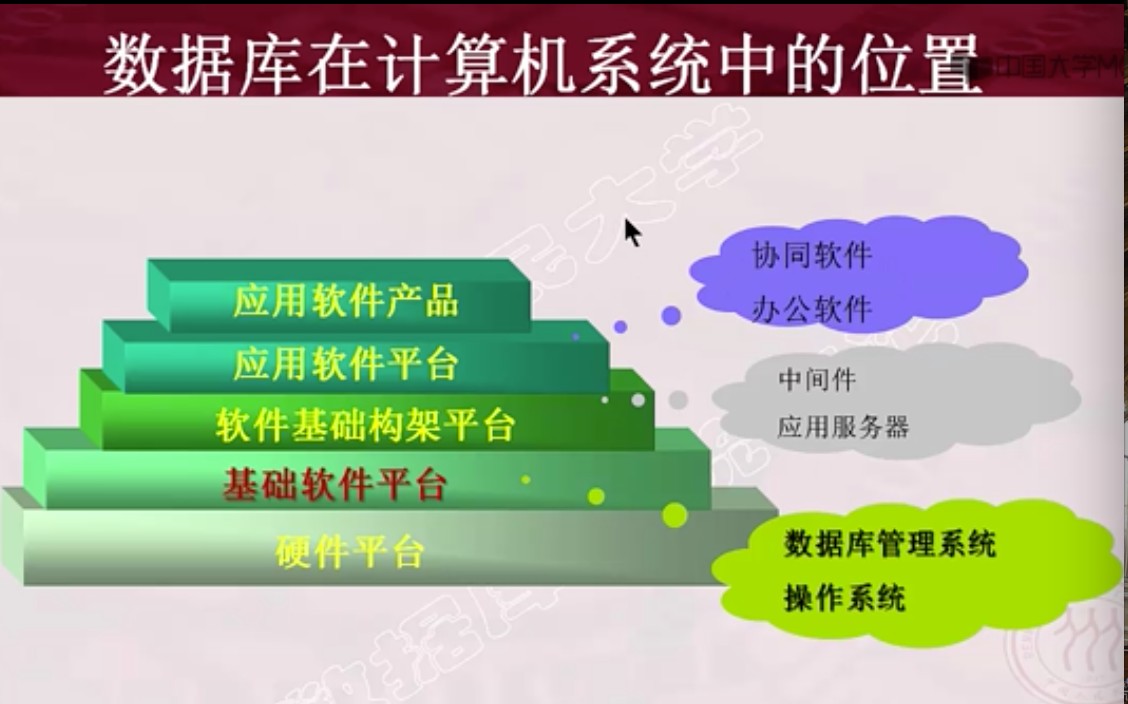
收集并抽取出⼀个应用所需要的⼤量量数据，将其保存，以供进⼀步加⼯工处理理，抽取有用信息，转换成有价值的知识。

数据库的基本特征

数据按照⼀定的数据模型组织，描述和存储。可为各种用户共享、冗余度较⼩小，易易扩展。数据独⽴性较⾼

数据库管理理系统（DataBase Management System，DBMS） 位于用户应用于操作系统之间的⼀层数据管理理软件

是基础软件，是⼀个⼤型复杂的软件系统



数据库管理理系统的主要功能：

1. 数据定义功能

提供数据定义语⾔言（DDL） 定义数据库中的数据对象

1. 数据组织、存储和管理理

分类组织、存储和管理理各种数据

确定数据在存储级别上的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

1. 数据操纵功能

提供数据操纵语⾔言（DML）

实现对数据库的基本操作（查询、插⼊、删除和修改）

1. 数据库的事务管理理和运⾏行管理理

数据的安全性、完整性、多用户对数据的并发使用发⽣古筝后的系统恢复数据库

由数据库管理理系统统⼀管理理和控制。保证事务正确运⾏行。

1. 数据库的建⽴和维护功能

提供使用程序/⼯工具，完成数据库数据批量量装载，数据库转储，介质故障恢复，数 据库的重组质和性能监视等。

1. 其他功能

数据库管理理系统与⽹网络中其他软件系统的通信数据库管理理系统之间的数据转换

异构数据库之间的访问和互操作

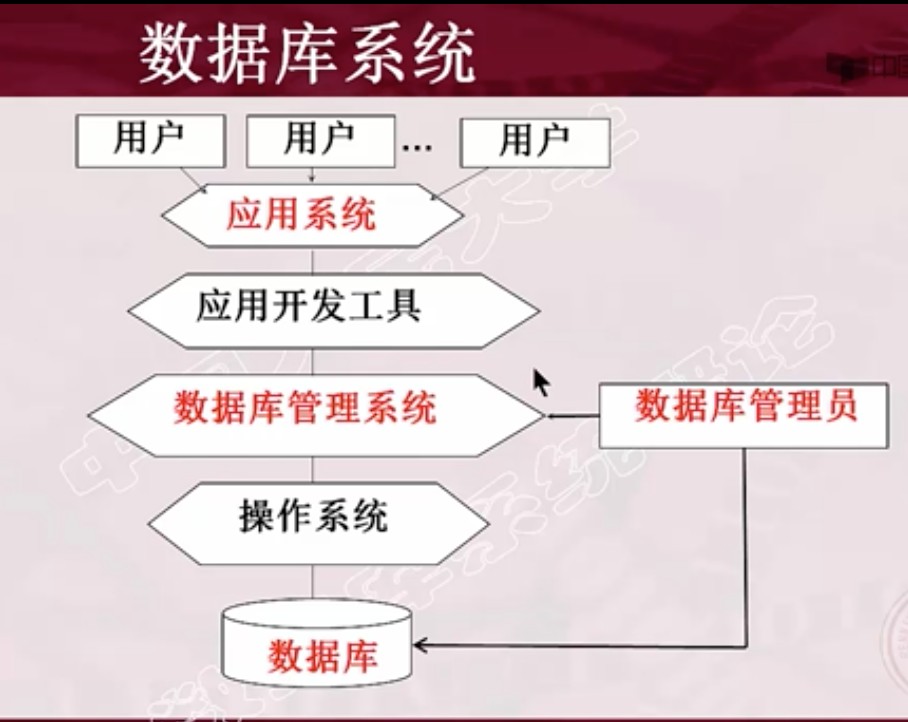
数据库系统（DataBase System，DBS）的用途科学地组织和存储数据、⾼效的获取和维护数据。

数据库系统是指在计算机系统中引⼊数据库后的系统构成在不引起混淆的情况下，常把数据库系统简称为数据库数据库系统的构成

数据库

数据库管理理系统（及其应用开发⼯工具） 应用程序

数据库管理理员（DataBase Administrator, DBA）



数据管理理三个阶段⽐比较



数据库的特点和优点

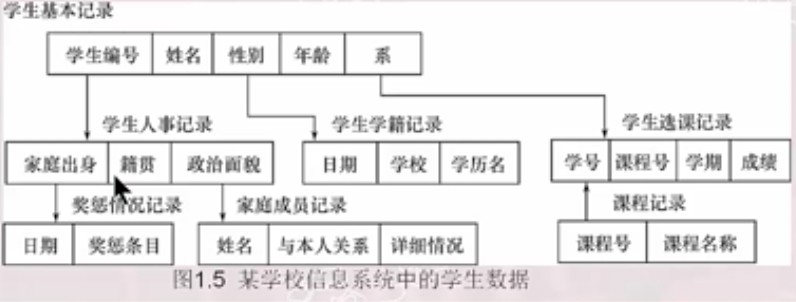
数据的整体结构化是数据库的主要特征之⼀

不再仅仅针对某⼀个应用，⽽是⾯向整个企业或组织

不仅数据内部结构化，整体是结构化的，数据之间具有联系 数据记录可以变⻓

数据的最⼩小存取单位是数据项

数据用数据模型描述，⽆无需应用程序定义



数据的共享性⾼，冗余度低且易易扩充

数据⾯型整个系统，可以被多个用户、多个应用共享使用。 数据共享的好处

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是系统易易于扩充

数据独⽴性⾼物理理独⽴性

指用户的应用程序与数据库中数据的物理理存储是相互独⽴的。当数据的物理理存储 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。

数据独⽴性⼜又数据库管理理系统的⼆级映像功能来保证。

数据由数据管理理系统统⼀管理理和控制数据库管理理系统提供的数据控制功能

1. 数据的安全性（Security）保护

保护数据以防不合法的使用造成的数据的泄密和破坏

1. 数据的完整性（Integrity）检查 保证数据的正确性、有效性和相容性。
2. 并发控制（Concurrency Control）

对多用户的并发操作加以控制和协调，防⽌止相互⼲扰⽽得到错误的结果。

1. 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 通俗地说数据模型就是现实世界的模拟。数据模型应该满⾜三个⽅⾯的要求：

能够较为真实地模拟现实世界； 容易易为⼈理理解；

便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

数据模型分为两类

1. 概念模型，也称为信息模型

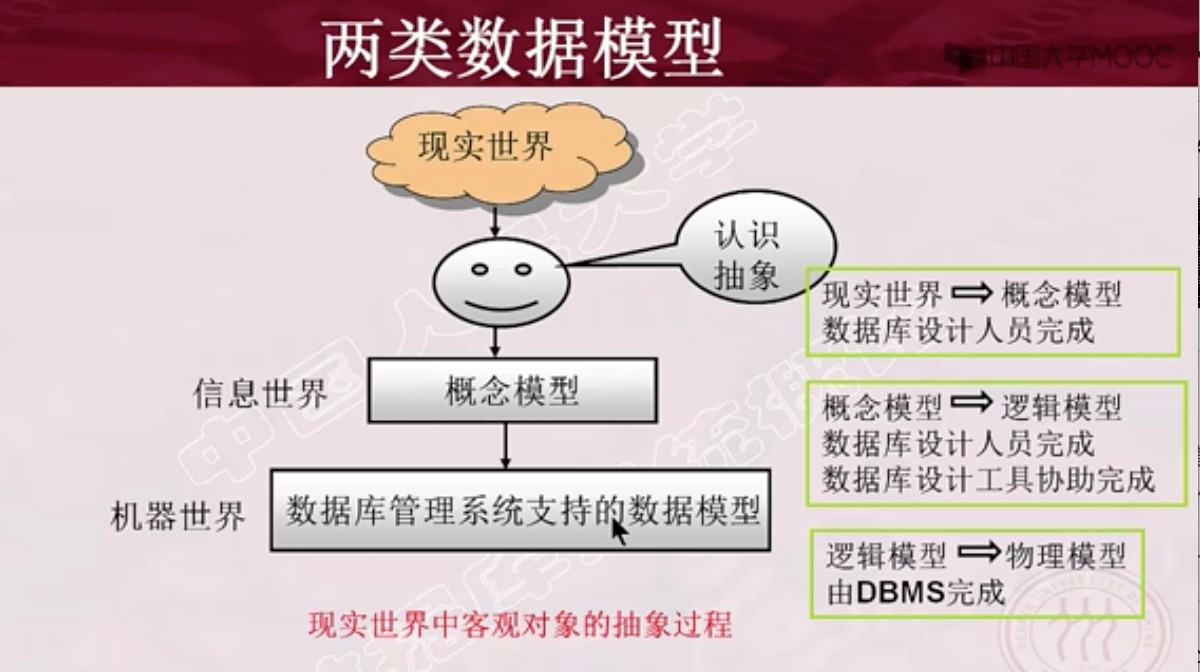
阿是按用户的滚点来对数据和信息建模，用于数据库设计。

1. 逻辑模型和物理理模型

逻辑模型主要包括⽹网站模型、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。物理理模型是对数据最底层的抽象

描述数据在系统内（磁盘上）的表示和存储⽅法。



1.2.2 概念模型概念模型的用途

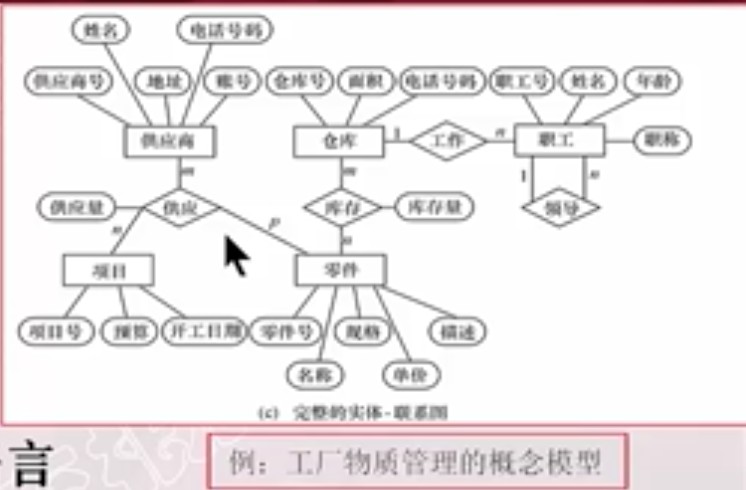
概念模型用于信息世界的建模

是现实世界到机器器世界的⼀个中间层次是数据库设计的有⼒力⼯工具

数据库设计⼈员和用户之间进⾏行交流的语⾔言对概念模型的基本要求

对概念模型的基本要求较强的语义表达能⼒力

简单、清晰。易易于用户理理解



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⻝⾷食物成为实体。可以是具体的⼈、事、物或抽象的概念。

1. 属性（Arribute）

实体具有的某⼀特征成为属性。⼀个实体可以由若⼲个属性来刻画。

1. 码（Key）

唯⼀标识实体的属性集成为码。

1. 实体型（Entity Type）

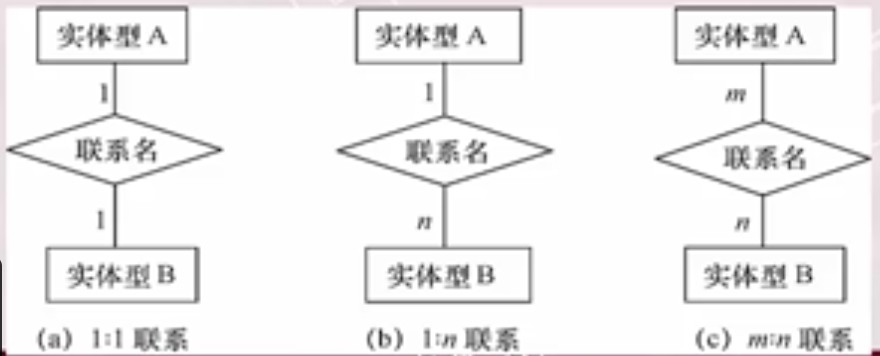
用实体名及其属性名集合来抽象和刻画同类实体成为实体型。

1. 实 体 集 （Entity Set） 统⼀类型实体的集合成为实体集
2. 联系（Relationship）

显示世界中⻝⾷食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：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（1：1）、⼀对多（1：m）和多对多（m:n）等多种类 型



数据库模型

数据模型是严格定义的⼀组概念的集合

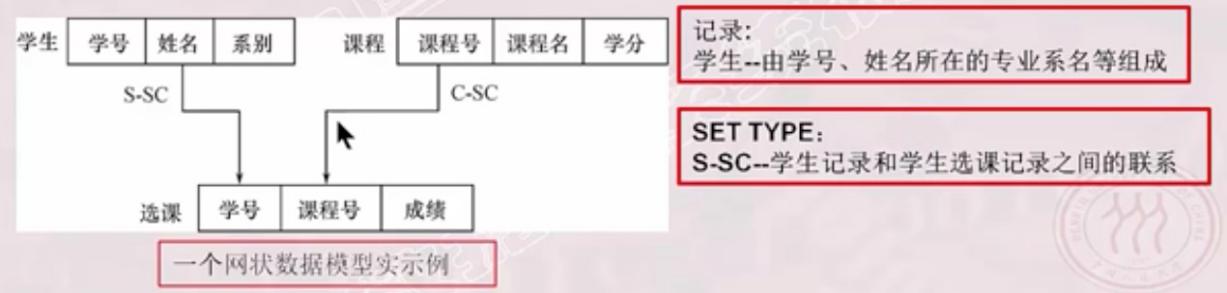
精确地描述了系统的静态特性、动态特性和完整性约束条件（Integrity Constraints）。

数据模型由三部分组成

1. 数据结构——描述系统的静态特性刻画数据模型性质的重要⽅⾯层次结构——层级模型

⽹网状结构——⽹网状模型关系结构——关系模型

描述数据库的组成对象——对象的类型、内容、性质描述对象之间的联系



1. 数据操作——描述系统的动态特性

对数据库中各种对象的实例允许执⾏行的操作的集合 包括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） 数据操作语⾔言

定义数据操作的确切含义、符合、优先级别实现数据操作的语⾔言

查询语⾔言——Query Language

更新语⾔言——DML

1. 完整性约束

⼀组完整性规则的集合

完整性规则：给定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。（实体的关系完整性和参 照完整性）。

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。

1.2.4 常用的数据模型

层次模型（Hierarchical Model）

用树形结构表示各类实体以及实体之间的联系表示⽅法

实体型：用记录类型描述

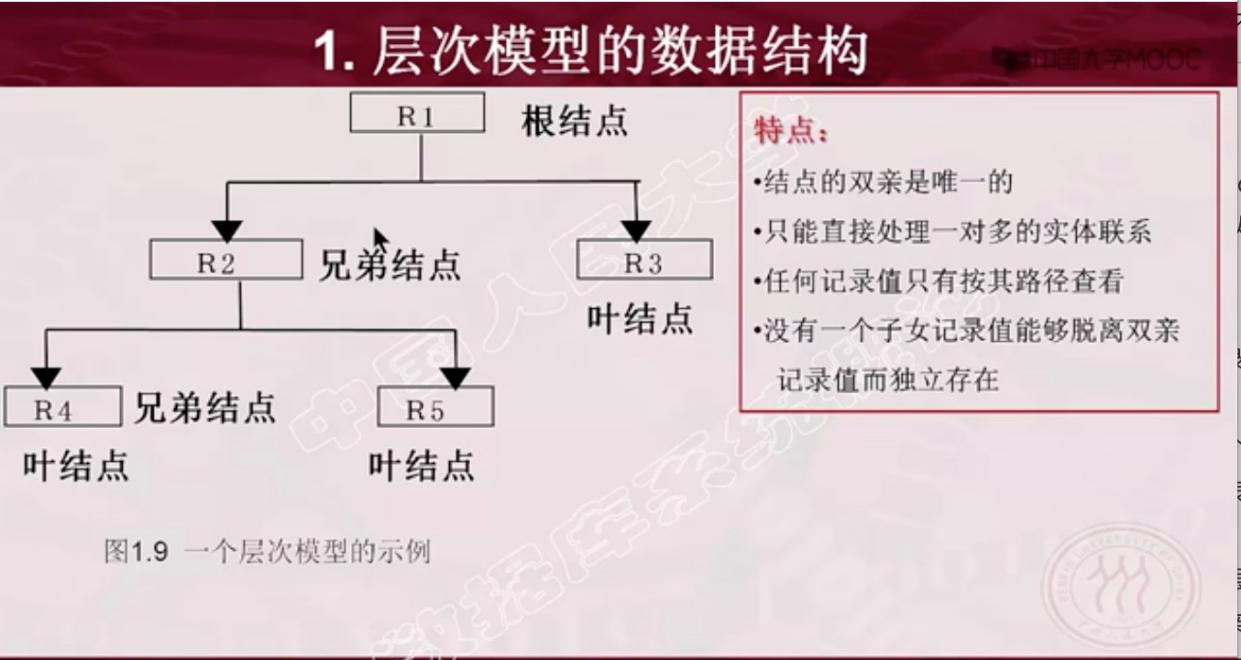
每个节点表示⼀个记录类型（实体） 属性：用字段描述

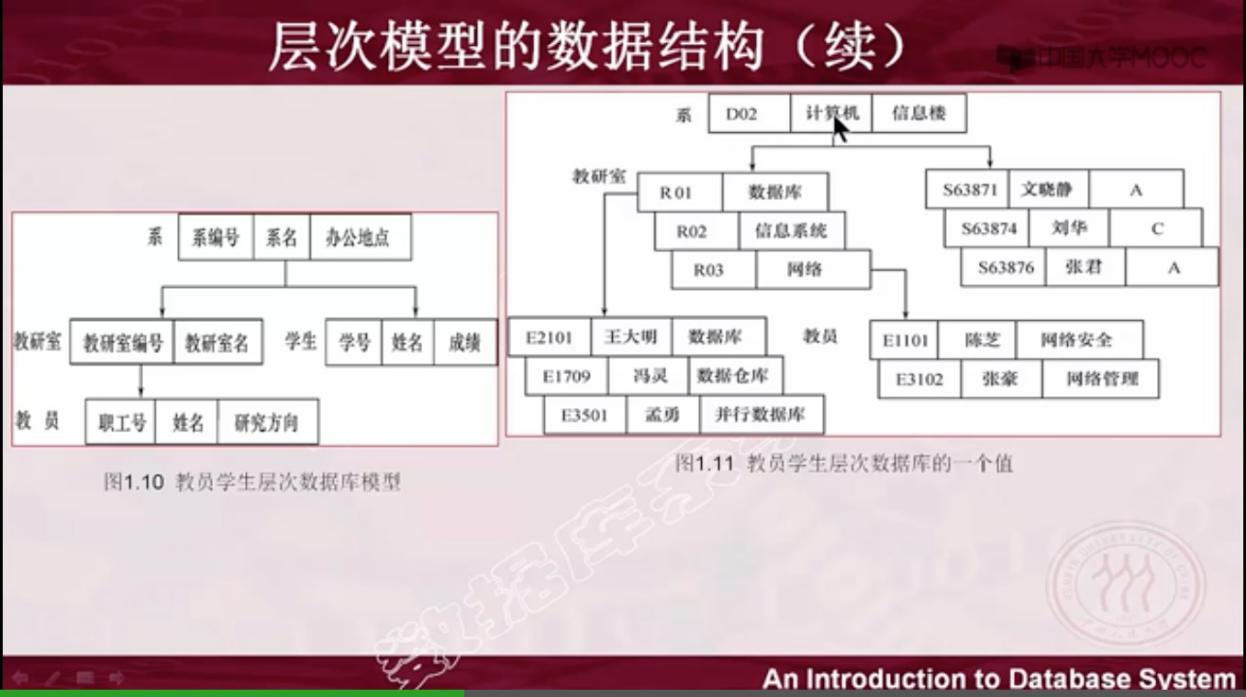
每个记录类型可包含若⼲个字段

联系：用节点之间的连线表示记录类型（实体）之间的⼀对多的⽗父⼦子联系层次模型的定义

满⾜下⾯两个条件的基本层次联系的集合为层次模型

有且只有⼀个结点没有双亲结点，这个结点成为根结点。根以外的其他节点有且只有⼀个双亲结点。





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

⽆无相应的双亲结点值就不能插⼊⼦子⼥女女结点值

如果删除双亲接地安置，则相应的⼦子⼥女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。层次模型的优缺点

优点：层次模型的数据结构⽐比较简单清晰

查询效率⾼，新能优于关系模型，不低于⽹网状模型 层次数据模型提供了良好的完整性⽀支持

缺点

结点之间的多对多联系表示不⾃自然

对插⼊和删除操作的限制多，应用程序的编写⽐比较复杂查询⼦子⼥女女结点必须通过双亲结点

层次数据库的命令（语⾔言）趋于程序化

⽹网状模型（Network Model）

表示⽅法（与层次数据模型相同） 实体型：用记录类型描述

每个节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：用节点之间的连线表示记录类型（实体）之间的⼀对多的⽗父⼦子联系

⽹网状模型的定义

满⾜下⾯两个条件的基本层次联系的集合： 允许⼀个以上的节点⽆无双亲；

⼀个节点可以有多个双亲。其他特点：

允许两个结点之间有多种联系要为每个联系命名

并指出与该联系有关的双亲记录和⼦子⼥女女记录

多对多联系在⽹网状模型中的表示

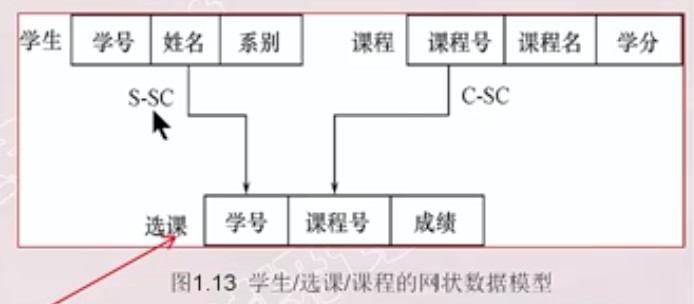
⽹网状模型间接表示多对多联系

⽅法：

将多对多联系分解成⼀对多联系

例如⼀个学⽣可以选修若⼲们课程，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。

引⼊⼀个学⽣选课的联结记录： 选课（学号，课程号，成绩）



⼀个多对多改成两个⼀对多

2. ⽹网状模型的数据操纵与完整性约束导航式的查询语⾔言和增删改操作语⾔言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子⼥女女结点值。允许只删除双亲结点值。实际的⽹网状数据库系统提供了⼀定的完整性约束

⽀支持码的概念：唯⼀标识记录的数据项的集合，取唯⼀的值； 保证⼀个联系中双亲记录与⼦子⼥女女记录之间是⼀对多联系；

可以定义双亲记录和⼦子⼥女女记录之间的某些约束条件。例：”书籍类别“的概念

要求双亲记录存在才能插⼊⼦子⼥女女记录，双亲记录删除时也连同删除。选课记录就应该满⾜这种约束条件：

学⽣选课记录值中学号必须是学⽣记录中存在的某⼀学⽣学号， 课程号必须是课程记录中存在的某⼀⻔门课程号。

⽹网状模型的优缺点优点

能够更为直接地描述显示世界，如⼀个节点可以有多个双亲； 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 利利于最终用户掌握。

DDL、DML语⾔言复杂，用户不容易易使用；

记录之间联系是通过存取路路径实现的，应用程序必须选择存取路路径，加重了程序员的负担。

这两种都是格式化模型，已经被关系模型取代

关系模型

在用户观点下，关系模型中数据的逻辑结构是⼀张⼆维表

⼀个元组就是表⾼里的⼀⾏行

关系（Relation）——⼀个关系对应通常说的⼀张表元组（Tuple）——表中的⼀⾏行即为⼀个元祖

属性（Attribute)——表中的⼀列即为⼀个属性，给⼀个属性起⼀个名称即属性名。

主码（Key）——也称码键。表中的某个属性组，它可以唯⼀确定⼀个元组域（Domain）——是⼀组具有相同数据类型的值的集合。

例：学⽣年龄属性的域（15~45岁），性别域是（男；⼥女女） 分量量——元祖中的⼀个属性值

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，系名，年级）

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：关机的每⼀个分量量必须是⼀个不可分的数据项，不允许表⾥里里还有表。

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

存取路路径对用户隐蔽，用户只要指出“找什什么”。不必仔细说明“怎么找”、 提⾼了数据的独⽴性，提⾼了用户⽣产率。

关系的完整性约束条件

实体完整性和参照完整性是关系的两个不变性

用户定义的完整性

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

关系模型的存取路路径对用户透明

具有更⾼的数据独⽴性，更好的安全保密性。 简化了程序员的⼯工作和数据库开发建⽴的⼯工作。

缺点

存取路路径对用户透明，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行优化，增加了开发数据库管理理系统的难度。

格式化模型中数据结构的单位是：基本层级联系

⾯向对象数据模型（Object Oriented Data Model） 对象关系数据模型（Object Relational Data Model）

半结构化数据模型（Semi-structure Data Model）——如XML

半结构化数据模型，图模型……

* 1. 数据库系统的结构

从数据库应用开发⼈员⻆角度看

数据库系统才用三级模式结构，是数据库系统内部的系统结构 从数据库最终用户⻆角度看

数据库系统的结构有： 单用户结构

主从式结构 分布式结构 客户——服务器器

浏览器器-应用服务器器/数据库服务器器数据库系统模式的概念

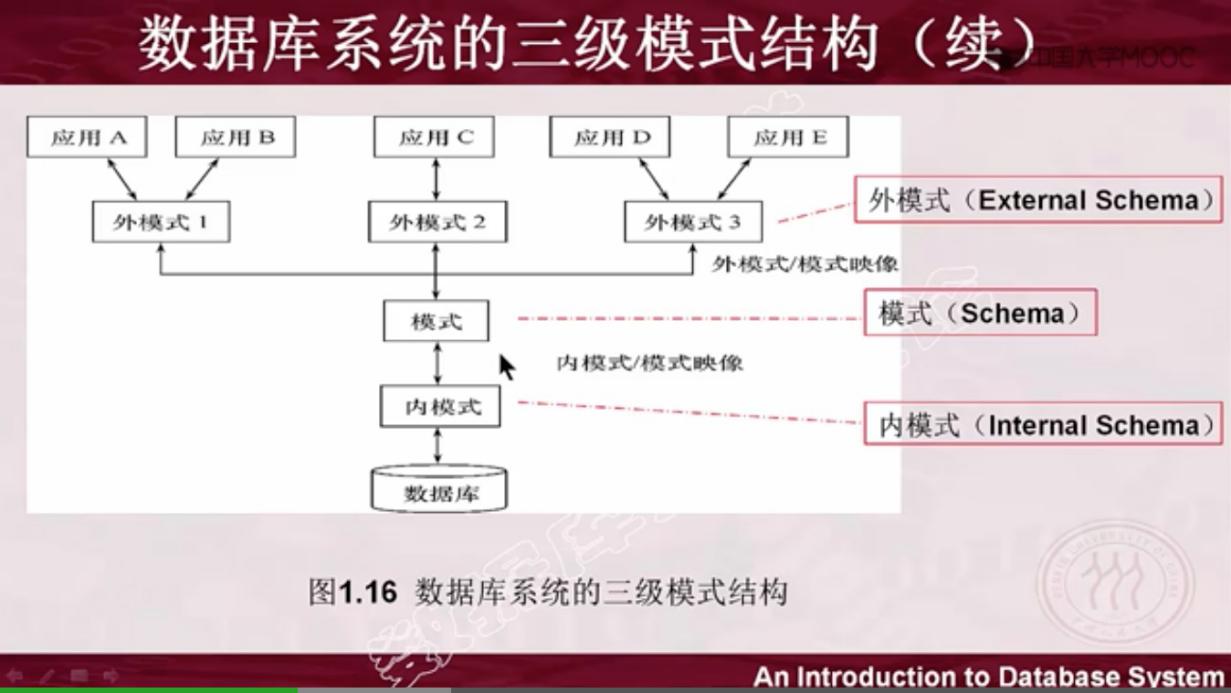
模式（Schema）

是对数据库逻辑结构和特征的描述是型的描述，不涉及具体值

模式是相对稳定的

实例（Instance）

数据库某⼀时刻的状态——模式的⼀个具体值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

⼀般，某⼀个应用的数据库由⼀个模式模式是数据库系统模式结构的中⼼

数据库 物理理存储细节和硬件环境⽆无关

与具体的应用程序、开发⼯工具及⾼级程序升级语⾔言⽆无关

定义模式

DDL定义数据的逻辑结构，以某种数据模型为基础

数据记录由那些数据项构成，数据项的名字、类型、取值范围等定义数据之间的联系

定义域数据有关的安全性，完整性要求

外模式（External Schema）

外模式（也称⼦子模式或用户模式）

数据库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与模式的关系

外模式通常是模式的⼦子集、⼀个模式可以有多个外模式，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

对模式中某⼀数据，在不同的外模式中结构、类型、⻓度。保密级别等都可以不同

外模式与应用的关系

⼀个外模式可以为多个应用系统所使用，⼀个应用程序只能使用⼀个外模式 外模式的用途

每个用户只能看⻅和访问所对应的外模式中的数据，简化用户视图 保真数据库安全性的⼀个有⼒措施

内模式（Internal Schema） 内模式（也称存储模式）

是数据物理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，堆存储，按hash⽅法存储等） 索引的组织⽅式（B+树，Bitmap，Hash）

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--如定⻓/变⻓，记录是否可以跨⻚页存放等。 数据库只有⼀个内模式

1.3.3 数据库的⼆级映像（mapping）功能与数据独⽴性

三级模式是对数据的三个抽象级别数据库管理理系统内部提供⼆级映像外模式/模式映像

对于每⼀个外模式，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外模式的描述中保证数据的逻辑独⽴性

当模式改变时，数据库管理理员对外模式/模式映像作相应改变，使外模式 保持不变

应用程序是⼀句句数据的外模式编写的，应用程序不必改变，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，说明某个逻辑记录对应何种存储结构。

数据库中模式/内模式映像是唯⼀的。该印象定义通常包含在模式描述中。保证数据的物理理独⽴性

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 理理员修改模式/内模式映像，使模式保持不变。

模式不变，则应用程序不变。保证了数据与程序的物理理独⽴性，简称数据的物理理独⽴性。

数据库的⼆级映像功能与数据独⽴性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。 从程序为中⼼——发展为以数据为中⼼

具有了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。

数据的存取⼜又数据库管理理系统管理理简化了应用程序的编制

⼤⼤减少了应用程序的维护和修改

三个抽象层次的联系和转换

* 1. 数据库系统的组成数据库

数据库管理理系统（及其开发⼯工具） 应用程序

数据库管理理员

* + 1. 硬件平台及数据库硬件平台及数据库

数据库系统对硬件资源的要求

⾜够⼤的内存

⾜够⼤的磁盘或磁盘阵列等外部设备较⾼的通道能⼒力，提⾼数据传送率

* + 1. 软件

数据库管理理系统

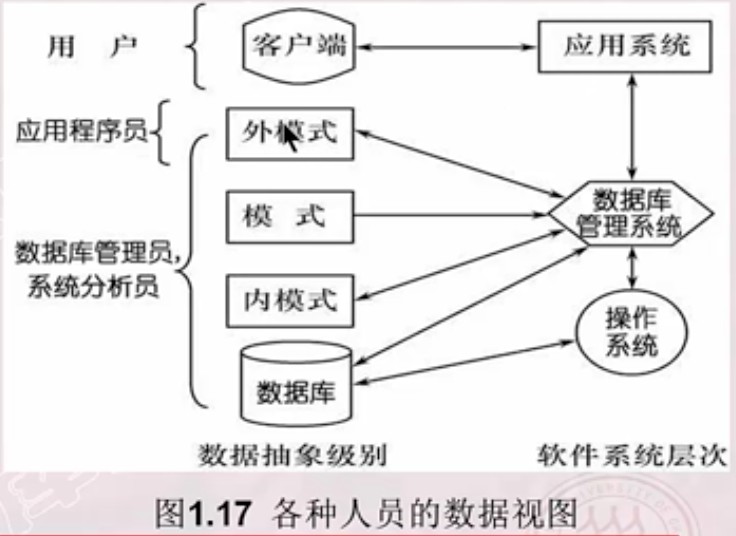
⽀支持数据库管理理系统的运⾏行的操作系统与数据库接⼝口的⾼级语⾔言及其编译系统以数据库管理理系统为核⼼的应用开发⼯工具为特定应用环境开发的数据库应用系统

* + 1. ⼈员

数据库管理理员

系统分析员和数据库设计⼈员应用程序员

最终用户



1. 数据库管理理员DBA 职责

1. 参与确定数据库中的信息内容和结构
2. 参与数据库的存储结构和存取策略略的设计
3. 参与确定数据安全性要求和完整性约束条件
4. 监控数据库的使用和运⾏行周期性转储数据库

数据⽂文件

⽇志⽂文件系统故障恢复介质故障恢复监视审计⽂文件

1. 数据库的改进和重组性能监控和调优

定期对数据库进⾏行重组质，以提⾼系统的性能需求增加和改变时，数据库须要重构造

* 1. 系统分析员

负责应用系统的需求分析和规范说明

与用户及数据库管理理员结合，确定系统的硬软件配置 参与数据库系统的概要设计

* 1. 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

* 1. 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

库。

库信息

* 1. 用户

用户指最终用户（End User）。他们通过应用系统的用户接⼝口使用数据

* + 1. 偶然用户

不经常访问数据库，但每次访问数据库时往往需要不同的数据

企业或组织机构的⾼中级管理理⼈员

* + 1. 简单用户

主要⼯工作是查询和更新数据库

银⾏行的职员、机票预订⼈员。旅馆总台服务员

* + 1. 复杂用户

⼯工程师、科学家、经济学家、科技⼯工作者等

直接使用数据库语⾔言访问数据库，甚⾄至能够基于数据库管理理系统的

应用程序结构编制⾃自⼰己的应用程序

关系数据库

单⼀的数据结构——关系

显示世界的实体以及实体间的各种联系均用关系来表示域是⼀组具有相同数据类型的值的集合。例：

整数

实数

介于某个取值范围的整数指定⻓度的字符串串集合

{'男'，‘⼥女女’}

…………

笛卡尔积

给定⼀组域D1，D2…，Dn，允许其中某些域是相同的。

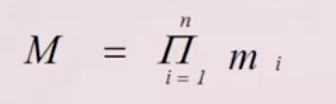
D1，D2,…Dn的笛卡尔积为： D1\*D2\*…Dn=

{(d1,d2,……，dn)|di∈Di，1=1，2，…，n}

所有域的所有取值的任意组合笛卡尔积可以看成是关系的域

基数（Cardinal number）

若Di(i=1,2,……，n)为有限集，其基数为mi（i=1，2，…n）则D1\*D2\*…Dn的基数M为：

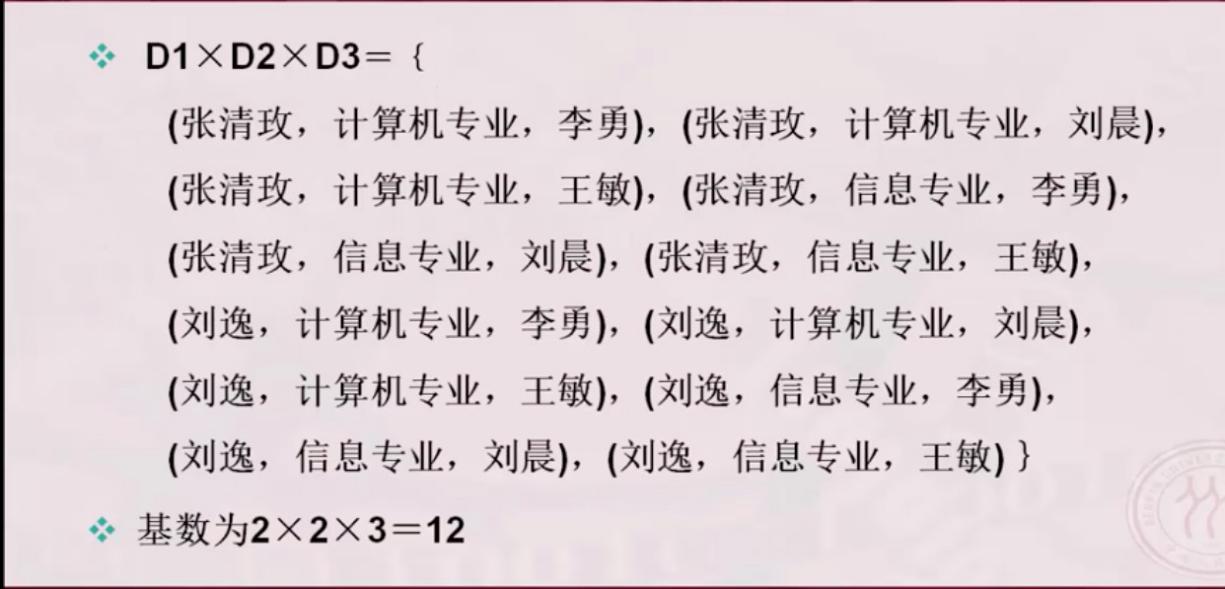


笛卡尔积的表示⽅法

笛卡尔积可表示为⼀张⼆维表

表中的每⾏行对应⼀个元组，表中的每列对应⼀个域 例如





关系

D1\*D2\*……\*Dn的⼦子集叫做在域D1，D2，……，Dn上的关系，表示为R（D1，D2，…，Dn）

R:关系名

n:关系的⽬目或度(Degree)

(2)元组

关系中的每个元素（d1,d2,…，dn）叫做⼀个n元组（n-tuple）或简称元组，通常

用t表示

1. 属性

关系中不同列可以对应不同域

为了加以区分，必须对每列起⼀个名字，成为属性（Attribute）

n⽬目关系必有n个属性

1. 码

候选码（Candidate key）

若关系中某⼀属性组的值能唯⼀地表示⼀个元组，则称该属性组为候选码 简单的情况：候选码只包含⼀个属性

全码（All-key）

最极端的情况：关系模式的所有属性组是这个关系模式的候选码，称为全码（All- key）

1. 如果说这个关系是定义在单个属性上的那么就成为⼀元关系定义在两个属性上的就称为是⼆元关系
2. 基本关系的性质
3. 列是同质的（homogeneous）
4. 不同的列可以出⾃自同⼀个域

列的顺序⽆无所谓，猎德词序可以任意交换任意两个元组的候选码不能相同

⾏行的顺序⽆无所谓，⾏行的次序可以任意交换分量量必须取原⼦子值不可再分

关系必须是笛卡尔积的⼀个⾃自⼰己才有实际意义外键