数据库原理与技术

刘瑞
lr@buaa.edu.cn

课程目标

- 了解，并掌握数据库系统的基本概念，原理和方法。
- 掌握关系数据模型及关系数据语言，能熟练应用SQL语言表达各种数据操作。
- 掌握数据库设计的理论和方法，初步具备使用、开发数据库应用系统的能力。
- 了解当前数据库领域研究的主要内容，为将来从事数据库及其相关领域的研究和开发打下基础。

主要参考书

数据库系统概论（第三版）　萨师煊 王珊 高等教育出版社 2000

数据库系统原理 李建中 王珊 电子工业出版社 1998

第一章

数据库系统概述

信息

信息是指现实世界事物的存在方式或运动状态的反映。具体地说，信息是一种已经被加工为特定形式的数据，这种数据形式对接收者来说是有意义的，而且对当前和将来的决策具有明显的或实际的价值。在信息社会中，信息是一种资源，其重要性可以与物质和能量相提并论，是企业赖以生存和发展所必须的。

数据

数据是将现实世界中的各种信息记录下的、可以识别的符号，是信息的载体，信息的具体表示形式。数据可用多种不同的形式来表示一种同样的信息，信息不随它的数据形式不同而改变。

数据的表现形式多种多样，不仅有我们熟知的数字和文字，还可以有图形、图像、声音等形式。

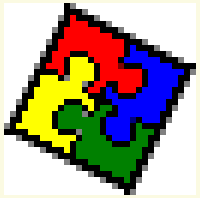
信息与数据

信息与数据的联系

- 数据是信息的符号表示，或称载体；
- 信息是数据的内涵，是数据的语义解释。
- 数据是符号化的信息。
- 信息是语义化的数据。

例，一幅图像

- 数据——彩色位图点阵
- 信息——微软产品



数据处理

数据处理实际上就是利用计算机进行数据处理的过程。该过程包括：数据的采集、整理、编码和输入，有效地把数据组织到计算机中，由计算机系统对数据进行一系列的加工、储存、合并、分类、计算、检索、传输、输出等操作过程。其目的是从大量原始数据中抽取和推导出对人们有价值的信息，以作为行动和决策的依据。

数据库

所谓数据库是长期储存在计算机内的、有组织的、可共享的数据集合。数据库中的数据按一定的数据模型组织、存储和描述，由DBMS统一管理，多用户共享。

数据库系统

数据库系统是指一个计算机存储记录的系统。即，它是一个计算机系统，该系统的目标是存储信息并支持用户检索和更新所需要的信息。它通常由数据库，软件，硬件，用户几个部分组成。

数据库管理系统

是一个通用的软件系统，由一组计算机程序构成。它能够对数据库进行有效的管理，并为用户提供了一个软件环境，方便用户使用数据库中的信息。

- 数据定义功能
- 数据操纵功能
- 数据库的运行管理功能
- 数据库的建立和维护功能

数据处理的三个阶段

- 人工处理阶段（50年代中期以前）
- 文件系统阶段（50年代后期---60年代中期）
- 数据库系统阶段（60年代后期开始）

人工处理阶段

背景：

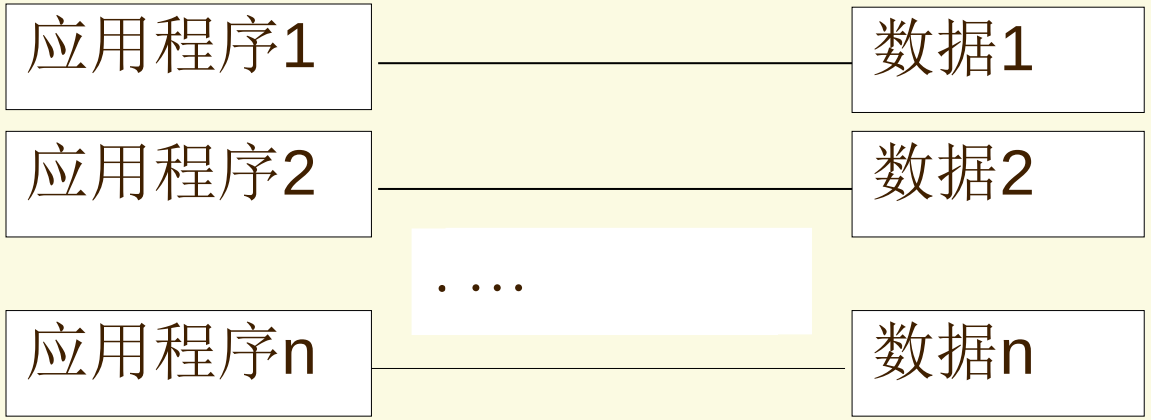
- 计算机主要用于科学计算。
- 外存只有磁带、卡片、纸带等，没有磁盘等直接存取设备。
- 没有操作系统，没有数据管理软件。

人工处理阶段（续I）

特点：

- 数据不保存
- 应用程序管理数据
- 数据不共享
- 数据不具有独立性

人工处理阶段（续II）



文件系统阶段

背景：

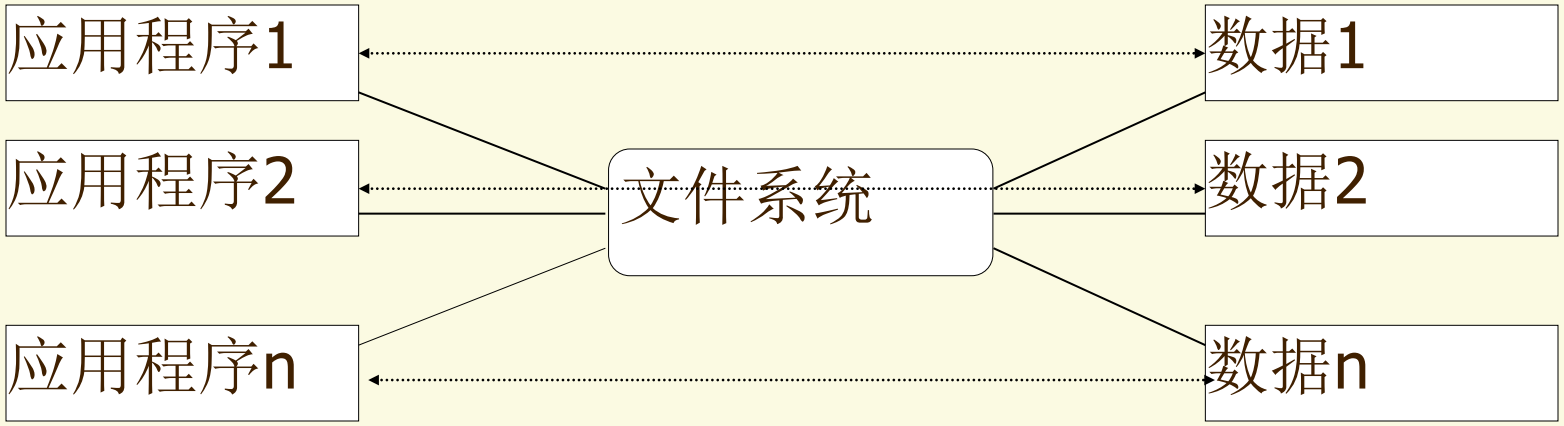
- 计算机不但用于科学计算，还用于管理。
- 外存有了磁盘、磁鼓等直接存取设备。
- 有了专门管理数据的软件，一般称为文件系统。

文件系统阶段（续I）

特点：

- 数据以文件的形式长期保存。文件形式多样化（索引文件、链接文件、直接存取文件、倒排文件等）。
- 一个数据文件对应一个或几个用户程序，还是面向应用的，具有一定的共享性。
- 由文件系统管理数据。通过文件系统提供存取方法，支持对文件的基本操作（增、删、改、查等），用户程序不必考虑物理细节。数据的存取基本上以记录为单位。
- 数据与程序有一定的独立性，因为文件的逻辑结构与存储结构由系统进行转换，数据在存储上的改变不一定反映在程序上。

文件系统阶段（续II）



文件系统阶段的缺点

- 数据冗余和不一致
- 数据访问困难
- 数据孤立
- 数据独立性差

文件系统阶段的缺点（续I）

- 完整性问题
- 原子性问题
- 并发访问异常
- 安全性问题

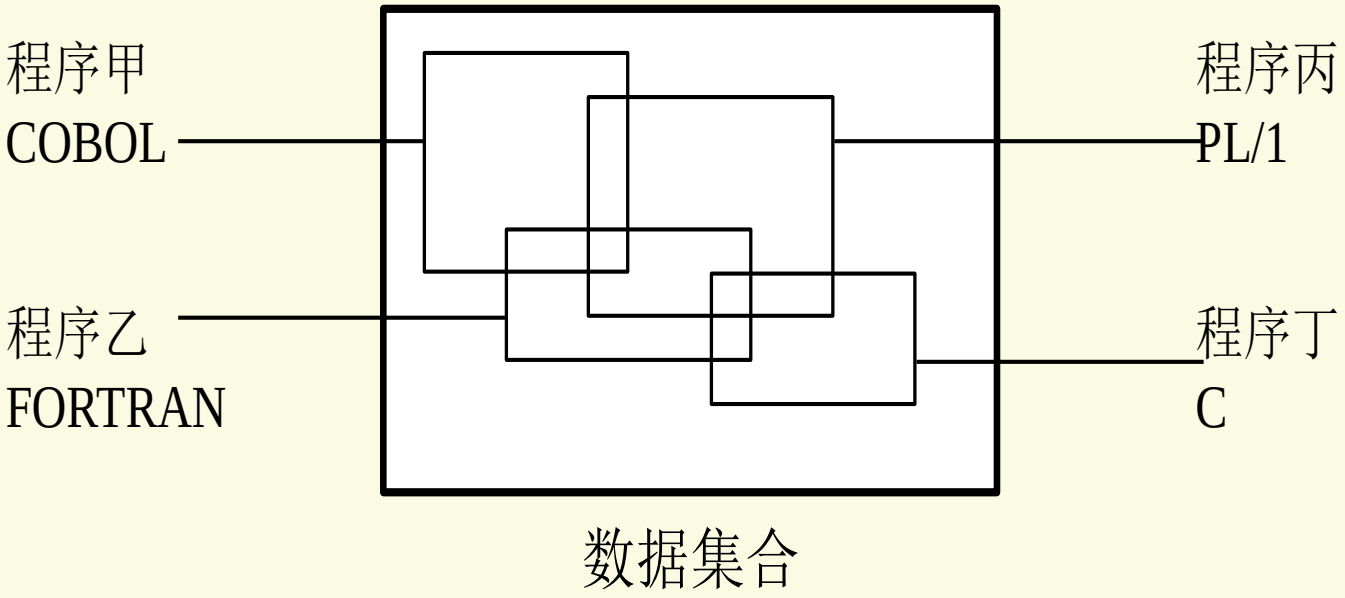
数据库系统阶段

背景：

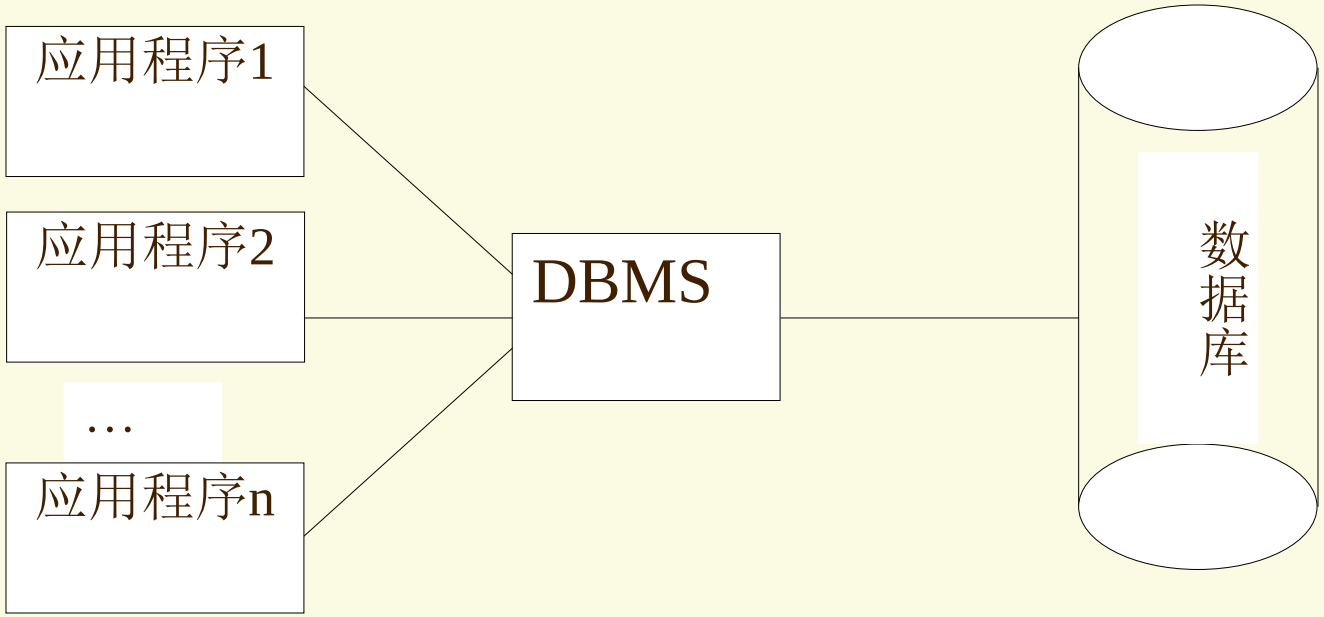
- 大规模数据管理。计算机管理的数据量大，关系复杂，共享性要求强（多种应用、不同语言共享数据）。
- 外存有了大容量磁盘，光盘。
- 软件价格上升，硬件价格下降，编制和维护软件及应用程序成本相对增加，其中维护的成本更高。

数据库系统阶段（续I）

数据库观点：数据不是依赖于处理过程的附属品，而是现实世界中独立存在的对象。



数据库系统阶段（续II）



数据库系统的特点

面向全组织的结构化。

- 支持全企业的应用而不是某一个应用。
- 通过统一的结构来描述数据及数据之间客观存在的本质联系，这是数据库系统的主要特征之一，是与文件系统的根本差别。

数据库系统的特点（续I）

- 数据集成与共享，可控冗余度
  - 数据面向整个系统，而不是面向某一应用，数据集中管理，并可以被多个用户和多个应用程序所共享。
  - 数据共享可以减少数据冗余，节省存储空间，减少存取时间，并避免数据之间的不相容性和不一致性。
  - 每个应用选用数据库的一个子集，只要重新选取不同子集或者加上一小部分数据，就可以满足新的应用要求，这就是易扩充性。
  - 根据应用的需要，可以控制数据的冗余度。

数据库系统的特点（续II）

数据独立性好

- **数据与程序相对独立，把数据库的定义和描述从应用程序中分离出去。描述又是分级的（全局逻辑、局部逻辑、存储），数据的存取由系统管理，用户不必考虑存取路径等细节，从而简化了应用程序。**
- **数据独立性：当数据的结构发生变化时，通过系统提供的映象（转换）功能，使应用程序不必改变。它包括数据的物理独立性和逻辑独立性。**

数据库系统的特点（续III）

- 统一的控制机制。
  - 数据的安全性控制（Security）
    - 保护数据以防止不合法的使用所造成的数据泄露和破坏。
    - 措施：用户标识与鉴定，存取控制。
  - 数据的完整性控制（Integrity）
    - 数据的正确性、有效性、相容性。
    - 措施：完整性约束条件定义和检查。

数据库系统的特点（续IV）

- 并发控制（Concurrency）
  - 对多用户的并发操作加以控制、协调，防止其互相干扰而得到错误的结果并使数据库完整性遭到破坏。
  - 措施：封锁。
- 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，防止数据丢失和损害，保证数据的正确性。

数据模型

模型是现实世界特征的模拟和抽象。数据模型则是现实世界数据特征的抽象。数据模型应满足三个方面的要求：

- 能比较真实地模拟现实世界；
- 容易为人理解；
- 便于在计算机上实现。

数据模型的分类

概念数据模型

- 按用户的观点来对数据和信息建模。用于组织信息世界的概念，表现从现实世界中抽象出来的事物以及它们之间的联系。这类模型强调其语义表达能力，概念简单、清晰，易于用户理解。它是现实世界到信息世界的抽象，是用户与数据库设计人员之间进行交流的语言。如E-R模型。

数据模型的分类（续I）

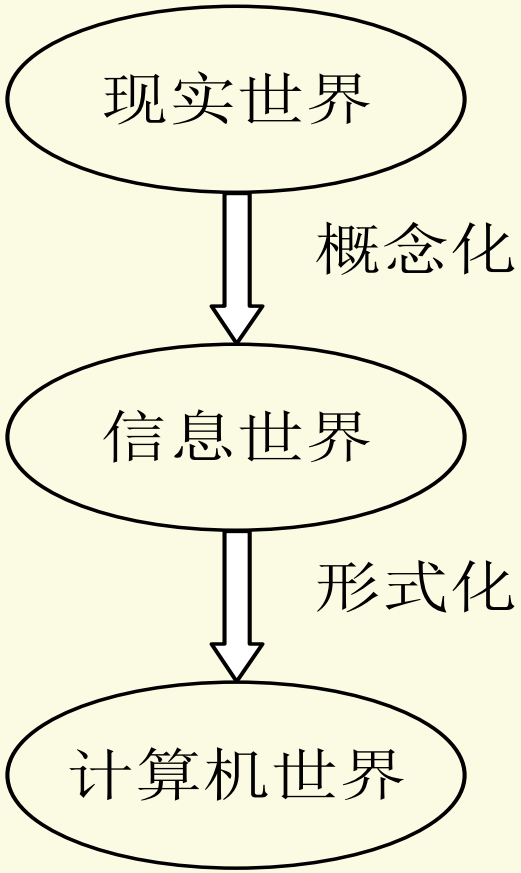
逻辑数据模型

- 从计算机实现的观点来对数据建模。是信息世界中的概念和联系在计算机世界中的表示方法。一般有严格的形式化定义，以便于在计算机上实现。如层次模型、网状模型、关系模型、面向对象模型。

物理数据模型

- 从计算机的物理存储角度对数据建模。是数据在物理设备上的存放方法和表现形式的描述，以实现数据的高效存取。如索引，HASH文件等等。

数据抽象



数据模型的组成要素

数据模型是实现数据抽象的主要工具。它包括以下组成部分：

- 数据结构

  描述系统的静态特性，是所研究的对象类型的集合。包括：

  - 数据本身：类型、内容、性质。如网状模型中的数据项、记录，关系模型中的域、属性，关系等。
  - 数据之间的联系：例如网状模型中的系型（Set Type）
  - 在数据库系统中一般按数据结构的类型来命名数据模型。

数据模型的组成要素（续I）

- 数据操作
  - 是对系统动态特性的描述，用于描述施加于数据之上的各种操作，即对数据库中对象的实例允许执行的操作的集合，包括操作及操作规则。一般有检索、更新（插入、删除、修改）操作。数据模型要定义操作含义、操作符号、操作规则，以及实现操作的语言。
- 数据的约束条件
  - 数据的约束条件是完整性规则的集合，规定数据库状态及状态变化所应满足的条件，以保证数据的正确、有效、相容。

概念数据模型的主要概念

实体(Entity)：

- 客观存在并可相互区分的事物叫实体。实体可以是具体的人、事、物，也可以是抽象的概念或联系。如一个学生、一次选课、老师与系的工作关系。

属性(Attribute)：

- 实体所具有的某一特性。一个实体可以由若干个属性来刻画。例如，学生实体可由学号、姓名、性别，出生年份、系、年级等组成。

域(Domain)：

- 属性的取值范围。例如，性别的域为（男、女），月份的域为１到１２的整数。

概念数据模型的主要概念（续I）

实体型(Entity Type)：

- 具有相同属性的实体必然具有共同的特征和性质，因此用实体名与其属性名集合来抽象和刻画同类实体，称为实体型。例，学生（学号、姓名、年龄、性别、系、年级）就是一个实体型。
- 注意实体型与实体（值）之间的区别，后者是前者的一个特例。如(9808100，王平，21，男，计算机系，2)是一个实体。

实体集(Entity Set)：

- 同型实体的集合称为实体集。
- 如全体学生。

概念数据模型的主要概念（续II）

码(Key)：

- 能唯一标识实体的属性集称为码。
- 如学号是学生实体的码。

联系(Relationship)：

- 在现实世界中，事物内部以及事物之间存在着联系，这些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。
- 如学生与老师间的授课关系，学生与学生间有班长关系。

概念数据模型的主要概念（续III）

- 联系也可以有属性，如学生与课程之间有选课联系，每个选课联系都有一个成绩作为其属性。
- 同类联系的集合称为联系集。

联系的种类

- 可以有一对一的（1:1），一对多的（1:m），多对多的（m:n）几种情况。
  - **一对一：**

    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。即为1：1。如一个班级只有一个正班长，一个正班长只在一个班中任职，则班级和班长之间存在一对一联系。

概念数据模型的主要概念（续IV）

- **一对多：**

  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。如一个班级有若干学生，每个学生只在一个班级中学习，则班级和学生是一对多联系。

- **多对多：**

  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中也有m个实体（m≥0）与之联系，则称实体集A与实体集B有多对多联系，记为m：n。如一门课由若干学生选修，一个学生可以选修多门课，则课程和学生是多对多联系。

实际上，一对一是一对多的特例，而一对多又是多对多的特例。

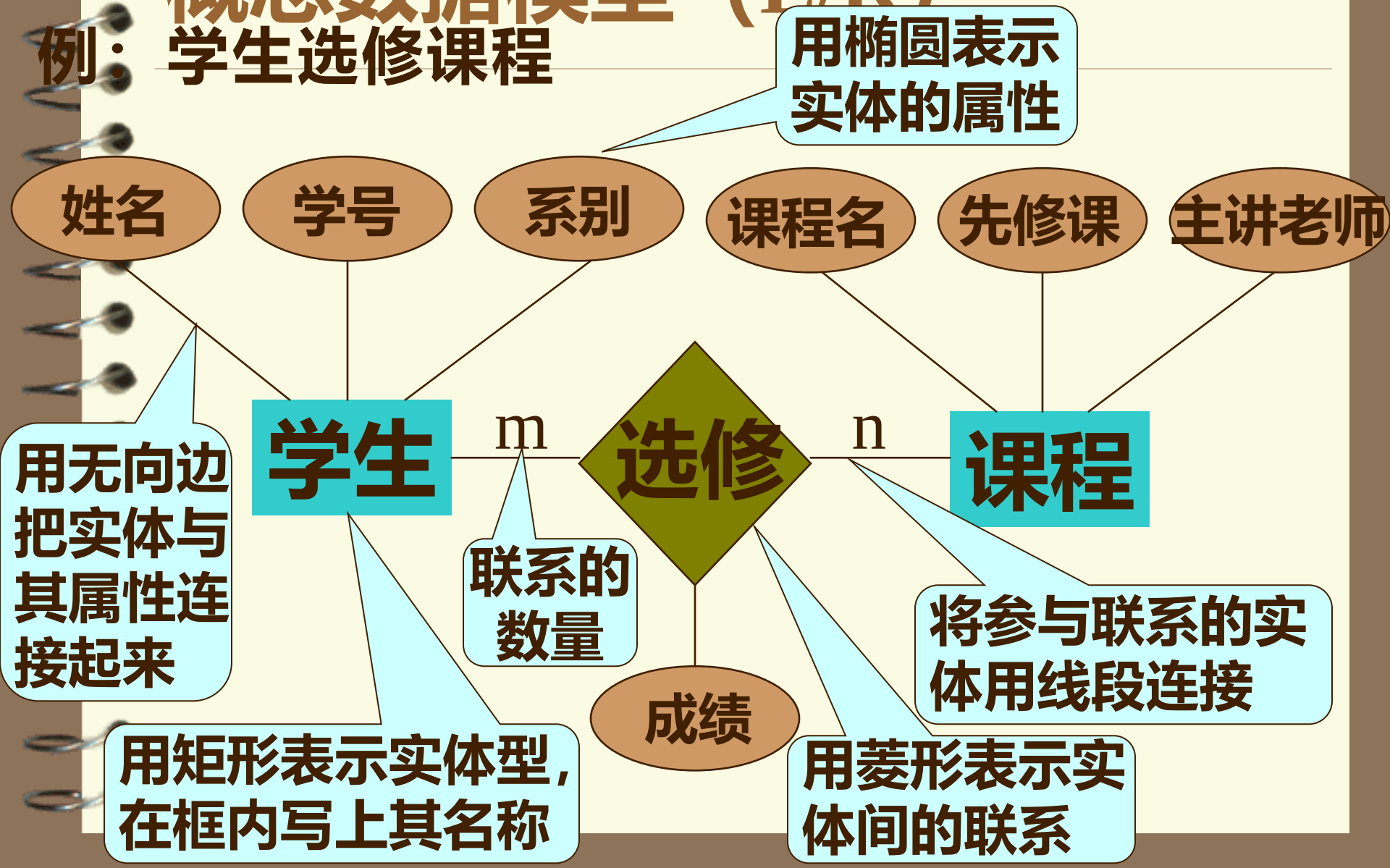
概念数据模型的主要概念（续V）

一般地，两个以上的实体型之间也存在着一对一、一对多、多对多联系。

同一个实体集内的各实体间也可以存在着一对一、一对多、多对多联系。

概念数据模型（E/R）

例：学生选修课程



最常用的数据模型

- 层次模型
- 网状模型
- 关系模型
- 面向对象模型

层次模型

用树结构表示实体之间联系的模型叫层次模型。

树由节点和连线组成，结点代表实体型，连线表示两实体型间的一对多联系。

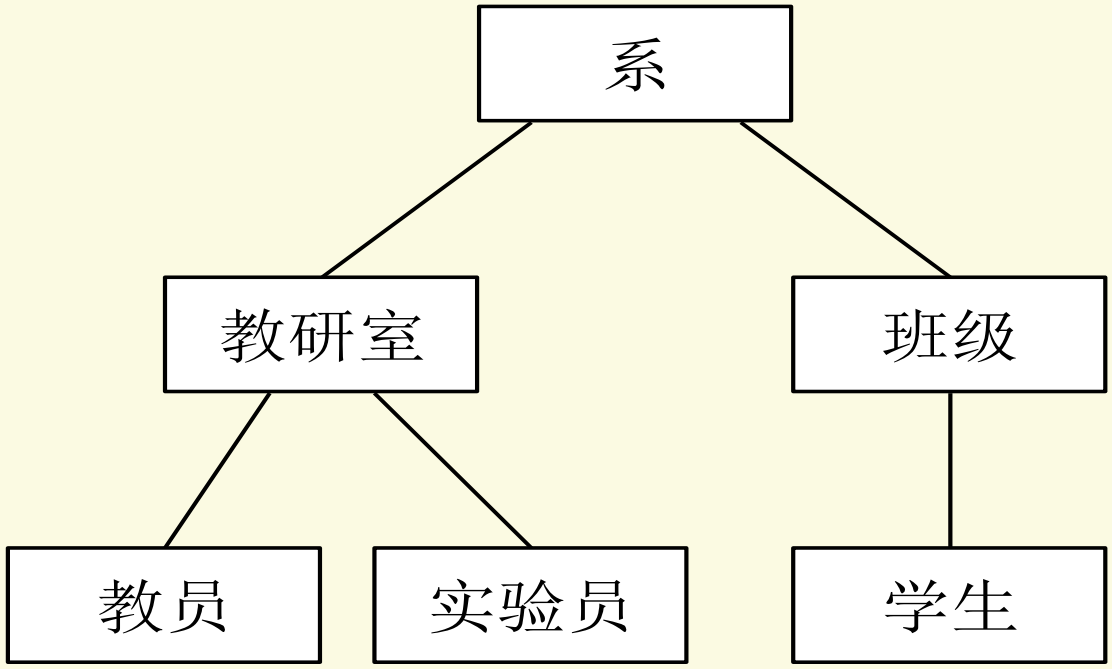
树有以下特性：

- 每棵树有且仅有一个结点无双亲结点，此结点称为树的根（Root）。
- 树中的其它结点都有且仅有一个双亲结点。

代表产品：

- IBM的IMS数据库，1968年研制成功。

层次模型（续I）



层次模型（续II）

优点：

- 结构简单，易于实现。
- 较好的性能。
- 良好的完整性支持。

缺点：

- 难以描述现实世界中非层次性的联系。
- 子结点的存取只能通过双亲结点来进行。
- 对插入和删除操作的限制比较多。
- 层次命令趋向程序化。

网状模型

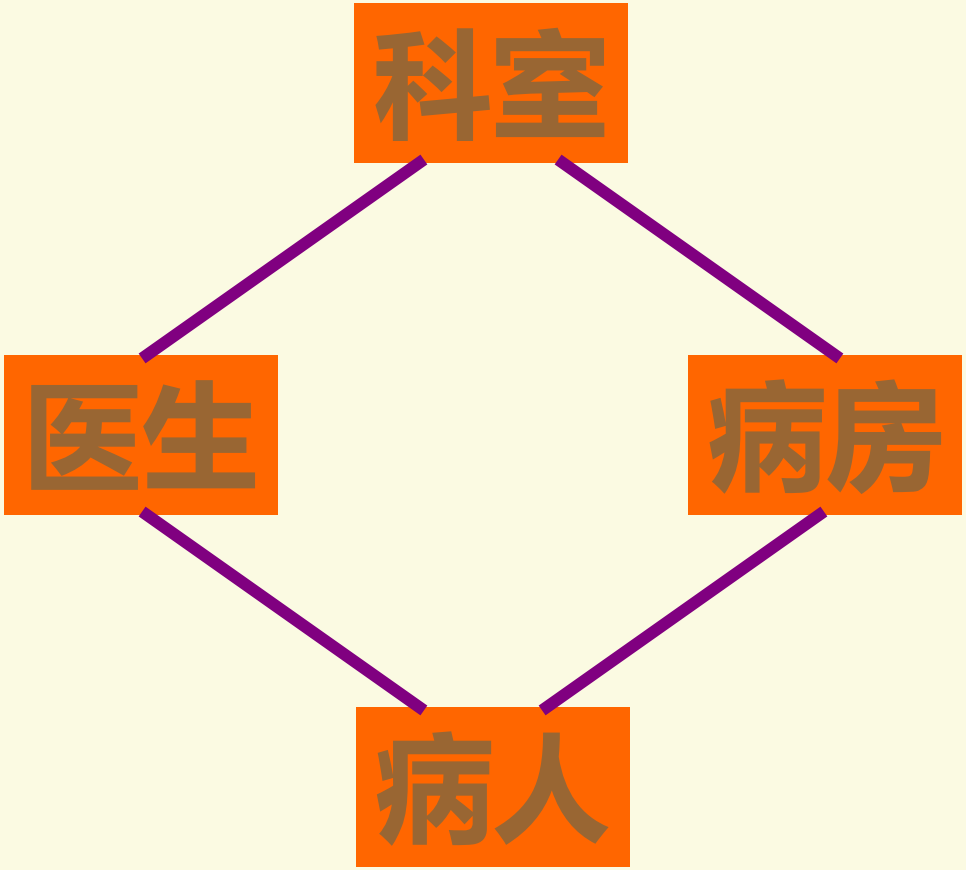
是一个满足下列条件的有向图：

- 可以有一个以上的结点无双亲结点。
- 至少有一个结点有多于一个的双亲结点。

DBTG报告：

- 1969年，由美国CODASYC（Conference On Data System Language，数据系统语言协商会）下属的DBTG（Data Base Task Group）组提出，确立了网状数据库系统的概念、方法、技术。

网状模型（续I）



网状模型（续II）

特点：

- 能够更为直接地描述现实世界。
- 具有良好的性能。

缺点：

- 结构复杂，不利最终用户掌握
- 其DDL，DML语言复杂，用户不易使用。

关系模型

用二维表来表示实体及其相互联系

姓名	学号	年龄	系别
张军	09701023	男	数学系
王红	09702011	女	物理系
李明	09708250	男	计算机系

关系模型（续I）

优点：

- 建立在严格的数学概念基础上。
- 简单，表的概念直观，用户易理解。
- 概念单一，实体和实体之间的联系都用关系表示。
- 存取路径对用户透明，具有更高的数据独立性和更好的安全保密性，也简化了程序员的工作和数据库开发建立的工作。

缺点：

- 性能往往不如非关系数据模型

数据库系统的模式

型与值

- 型是指对某一类数据的结构和属性的说明，值是型的一个具体赋值。

模式：

- 是对数据库中全体数据的逻辑结构和特征的描述，它仅仅涉及到型的描述，不涉及到具体的值。模式的一个具体值称为模式的一个实例。同一个模式可以有很多实例。模式是相对稳定的，而实例是相对变动的，因为数据库的数据是在不断更新的。模式反映的是数据的结构及其联系，而实例反映的是数据库某一时刻的状态。

数据库系统的模式（续I）

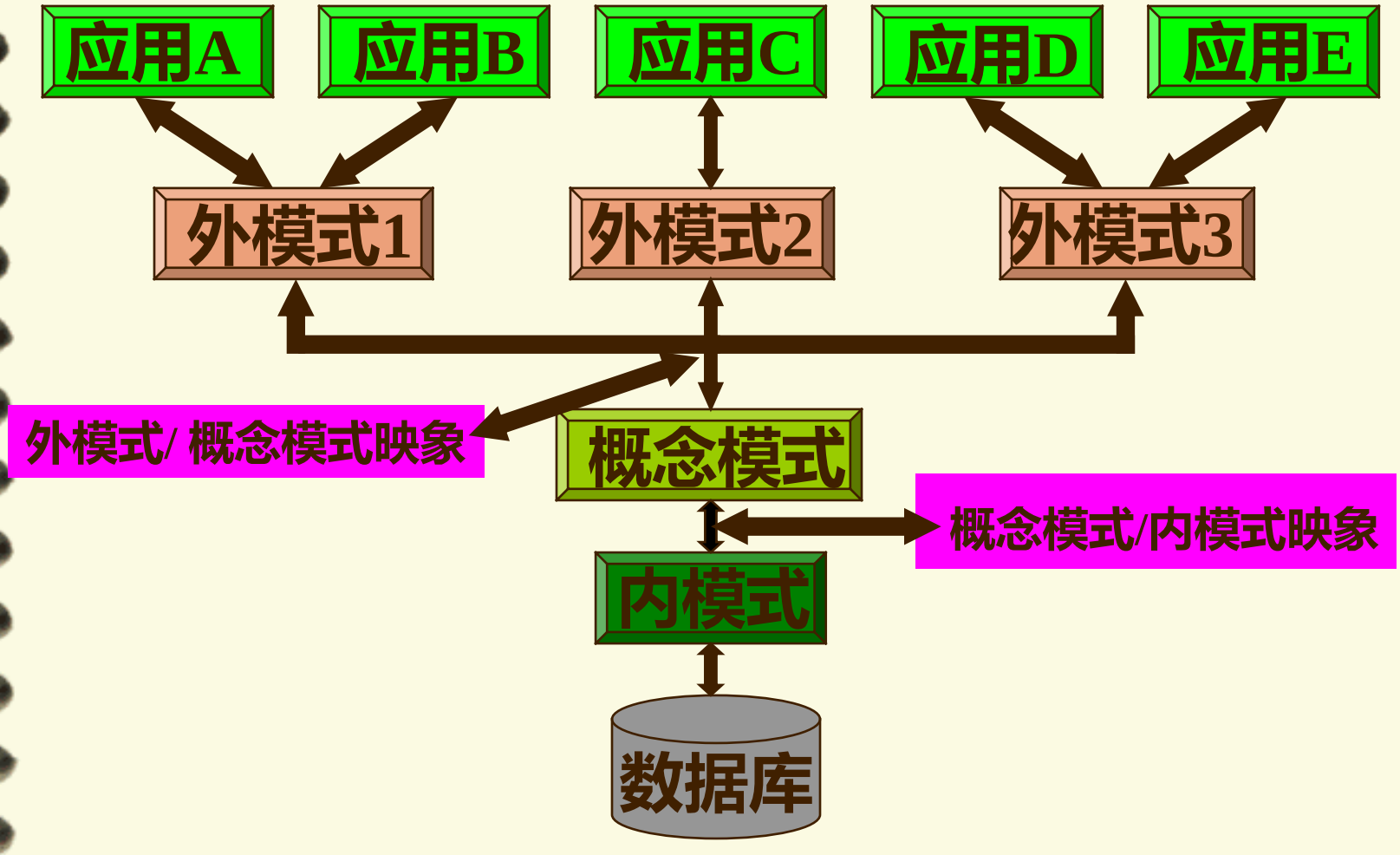
模式的分级：

- 为了提高数据的物理独立性和逻辑独立性，使数据库的用户观点，即用户看到的数据库，与数据库的物理方面，即实际存储的数据库区分开来，数据库系统的模式是分级的。

数据库系统三级模式结构：

- CODASYL（Conference On Data System Language,美国数据系统语言协商会）提出 概念模式、外模式、存储模式三级模式的概念。三级模式之间有两级映象。

数据库系统的模式（续II）



数据库系统的模式（续III）

外模式(Sub-Schema)：

- 用户的数据视图。是数据的局部逻辑结构，模式的子集。

概念模式(Schema)：

- 所有用户的公共数据视图。是数据库中全体数据的全局逻辑结构和特性的描述。

内模式(Storage Schema)：

- 又称存储模式。数据的物理结构及存储方式。

数据库系统的模式（续IV）

外模式/概念模式映象：

- 定义某一个外模式和概念模式之间的对应关系，映象定义通常包含在各外模式中。当概念模式改变时，修改此映象，使外模式保持不变，从而应用程序可以保持不变，称为逻辑独立性。

概念模式/内模式映象：

- 定义数据逻辑结构与存储结构之间的对应关系。存储结构改变时，修改此映象，使概念模式保持不变，从而应用程序可以保持不变，称为物理独立性。

数据库系统的组成部分

- 数据库
  - 全组织的日常运营所需要的各种数据,包括目标数据（数据本身）及描述数据（对数据的说明信息）
- 软件
  - 数据库管理系统（DBMS）
  - 支持DBMS运行的操作系统
  - 具有与数据库接口的高级语言及其编译系统
  - 应用开发工具
  - 为特定应用环境开发的数据库应用系统

数据库系统的组成部分（续I）

硬件

- 足够的内存，以运行OS，DBMS，以及应用程序和提供数据缓存。
- 足够的存取设备如磁盘，提供数据存储和备份
- 足够的I/O能力和运算速度，保证较高的性能。
- 其他设备

数据库系统的组成部分（续II）

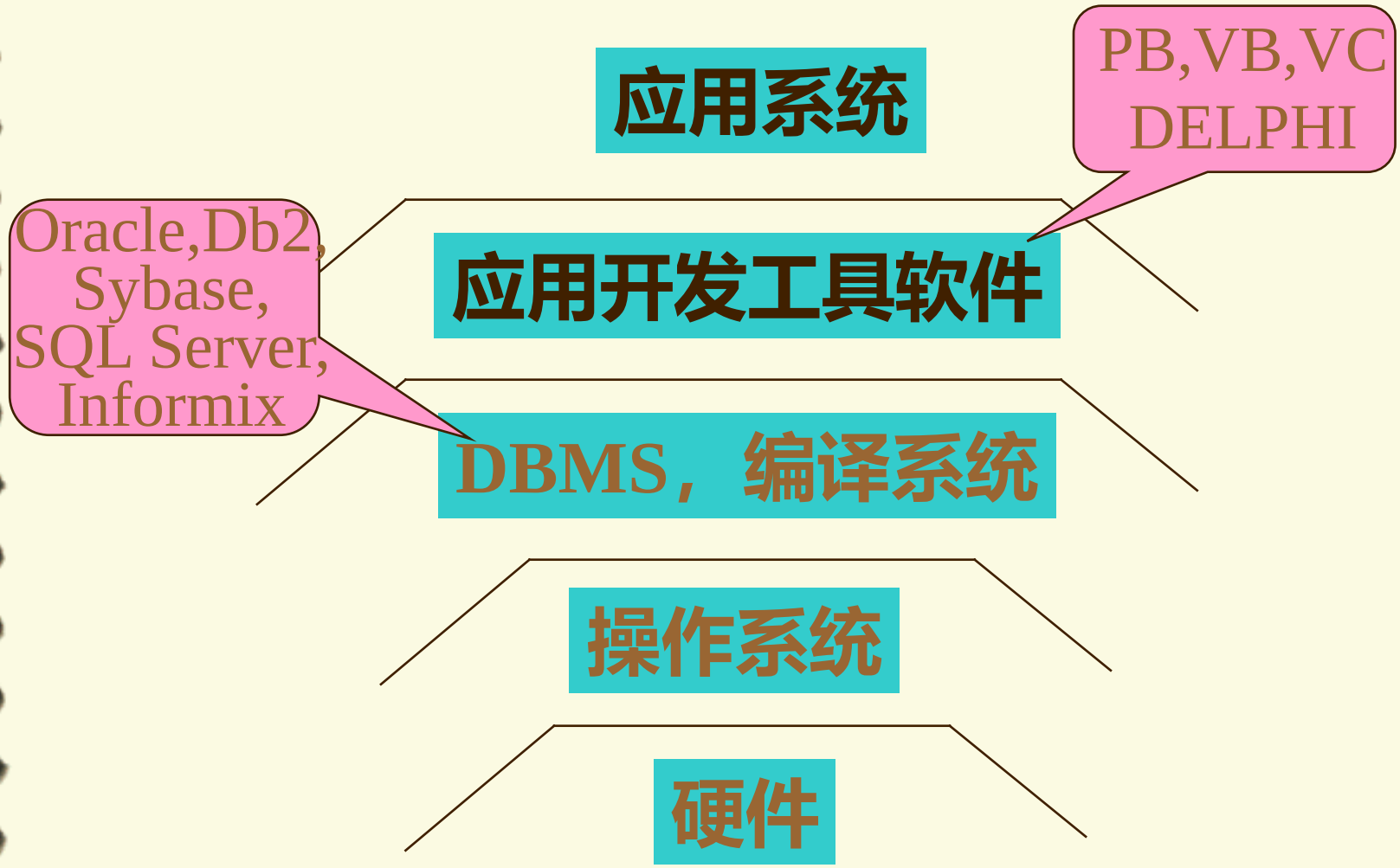
- 用户
  - 数据库管理员
  - 系统分析员和数据库设计者
  - 应用程序员
  - 最终用户
    - 偶然用户
    - 简单用户
    - 复杂用户

数据库系统的组成部分（续III）

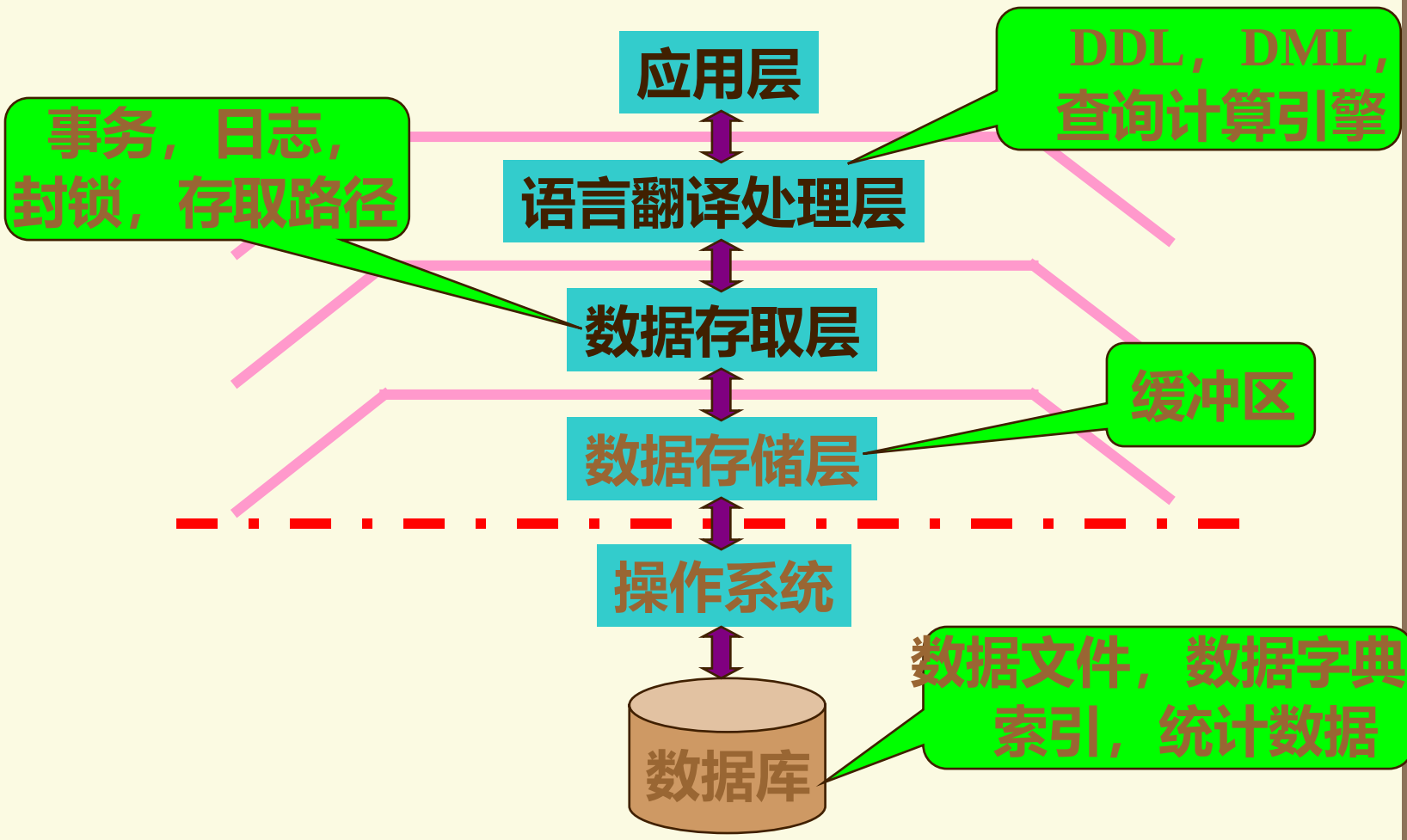
DBA职责：

- 决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
- 监控数据库的使用和运行
- 数据库的改进和重组重构

数据库系统的软硬件层次



DBMS的管理层次


应用层
DDL，DML，
查询计算引擎
事务，日志，
封锁，存取路径
语言翻译处理层
数据存取层
缓冲区
数据存储层
操作系统
数据文件，数据字典，
索引，统计数据
数据库

DBMS的主要功能

数据库定义功能

- 提供DDL语言（Data Description Language）描述外模式、模式、内模式（源模式）。
- 模式翻译程序把源模式翻译成目标模式，存入数据字典中。

DBMS的主要功能（续I）

数据存取功能

- 提供DML语言（Data manipulation language）对数据库进行检索、插入、修改、删除。
- DML类型
  - 宿主型：DML不独立使用，嵌入到高级语言（主语言）程序中使用。
  - 自含型：独立使用，交互式命令方式。
- DBMS控制并执行DML语句
  - 宿主型：有预编译和增强编译两种方式。
  - 自含型：解释执行。

DBMS的主要功能（续II）

- 数据库运行管理
  - 并发控制、存取控制、完整性约束条件检查和执行，日志组织和管理，事务管理和自动恢复。
- 数据组织、存储和管理
  - 用户数据、索引、数据字典的组织、存储和管理，包括文件结构、存取方式、数据之间的联系的实现等。
- 数据库的建立和维护功能
  - 数据的装入、转换、卸出，数据库的转储、恢复、性能监视和分析等。

数据库技术的发展

第一代数据库系统

- 70年代的层次和网状数据库系统。
- 代表：1969年IBM研制的IMS。
- 60年代末，70年代初CODASYL给出的DBTG报告。

第二代数据库系统

- 1970年IBM San Jose实验室研究员E.F.Codd提出关系模型
- 代表：IBM开发的System R和加州大学Berkley分校开发的INGRES。

数据库技术的发展 （续I）

第三代数据库系统

- 面向对象数据库
- 数据库技术与其他学科的内容相互结合
- 面向应用领域的数据库技术研究

数据库技术的研究领域

- 数据库管理软件的研制
- 数据库设计
- 数据库理论