



# 数据库概述



马 帅

计算机学院

[mashuai@buaa.edu.cn](mailto:mashuai@buaa.edu.cn)





# 提纲

♠ 数据管理技术的发展

♠ 数据描述

♠ 数据模型

♠ 数据库的体系结构

♠ 数据库系统的构成

♠ 小结





# 数据管理技术的发展

- ♠ 人工管理阶段（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）
- ♠ 数据库系统 VS 文件系统
- ♠ 几个关键的数据库概念



# 人工管理阶段（50年代中期以前）

## ♠ 背景

### ♣ 计算机主要用于科学计算

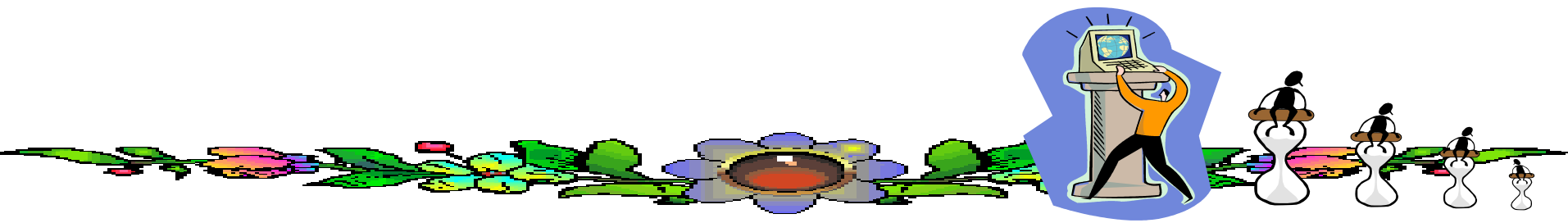
- ◆ 数据量小、结构简单，如高阶方程、曲线拟和等

### ♣ 外存为顺序存取设备

- ◆ 磁带、卡片、纸带，没有磁盘等直接存取设备

### ♣ 软件只有汇编语言，没有数据管理软件

- ◆ 用户用机器指令编码，通过纸带机输入程序和数据，程序运行完毕后，用户取走纸带和运算结果，让下一用户上机操作





# 人工管理阶段

## ♠ 特点

### ♣ 数据不保存在机器中

- ◆ 计算机主要用于计算，任务完成后，数据空间与程序空间一起释放

### ♣ 用户完全负责数据管理工作

- ◆ 数据的组织、存储结构、存取方法、输入输出等

### ♣ 数据完全面向特定的应用程序

- ◆ 每个用户使用自己的数据，数据不保存，用完就撤走

### ♣ 数据与程序没有独立性

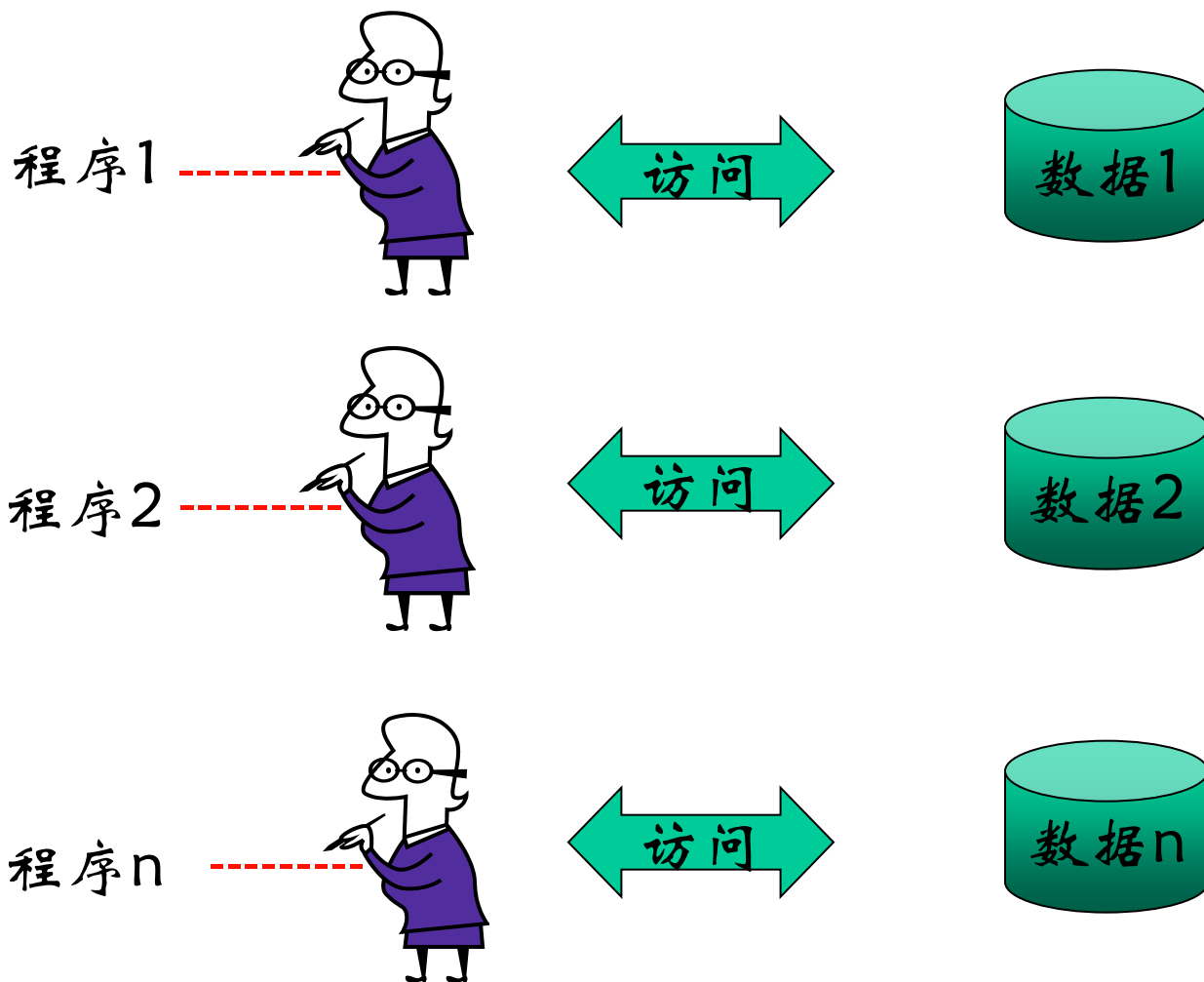
- ◆ 程序中存取数据的子程序随着存储结构的改变而改变

### ♣ 磁带的特点

- ◆ 优点：廉价地存放大容量数据
- ◆ 缺点：顺序访问；1%所需，100%访问



# 人工管理阶段





# 文件系统阶段（50年代后期-60年代中期）

## ♠ 背景

- ♣ 计算机不但用于科学计算，还用于信息管理
- ♣ 数据量的增加使得数据存贮、检索、维护成为迫切的需要
- ♣ 外存有了磁盘、磁鼓等直接存取设备
  - ◆ 直接存取设备（DASD）
    - 无须顺序存取
    - 由地址直接访问所需记录





# 文件系统阶段

## ♠ 背景（续）

- ♣ 软件有了高级语言和操作系统
- ♣ 操作系统有专门管理数据的软件，一般称为文件系统
  - ◆ 文件存储空间的管理
  - ◆ 目录管理
  - ◆ 文件读写管理
  - ◆ 文件保护
  - ◆ 向用户提供操作接口
- ♣ 处理的方式有批处理，也有联机实时处理







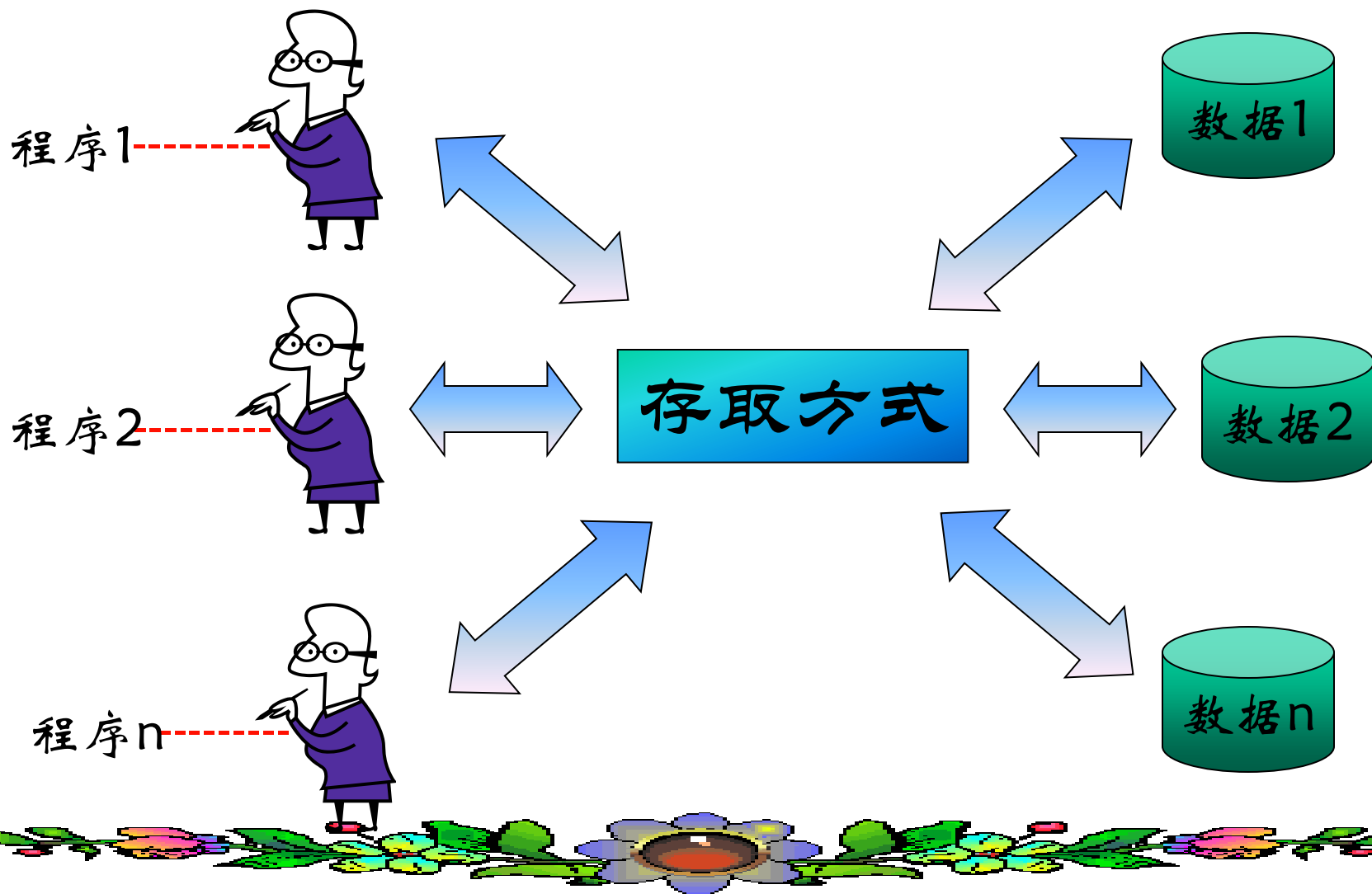
# 文件系统阶段

## ♠ 特点

- ♣ 数据可长期保存在外存的磁盘上
  - ◆ 支持对文件的基本操作（增、删、改、查等）
- ♣ 数据的逻辑结构和物理结构有了区别
- ♣ 系统提供一定的数据管理功能
- ♣ 文件的组织呈现多样化
  - ◆ 有索引文件、链接文件、直接存取文件、倒排文件等
- ♣ 数据仍是面向应用的
  - ◆ 一个数据文件对应一个或几个用户程序
- ♣ 数据与程序有一定的独立性
  - ◆ 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
- ♣ 数据的存取基本上以记录为单位



# 文件系统阶段





# 文件系统阶段

## ♠ 数据与程序的独立性差

- ♣ 文件系统的出现并没有从根本上改变数据与程序紧密结合的状况，数据的逻辑结构改变则必须修改应用程序
- ♣ 文件系统只是解脱了程序员对物理设备存取负担，它并不理解数据的语义，只负责存储
- ♣ 数据的语义信息只能由程序来解释，也就是说，数据收集以后怎么组织，以及数据取出来之后按什么含义应用，只有全权管理它的程序知道。
- ♣ 一个应用若想共享另一个应用生成的数据，必须同另一个应用沟通，了解数据的语义与组织方式





# 文件系统阶段

## ♠ 数据的共享性差，冗余度大（冗余性）

### ♣ 数据面向应用

- ◆ 即使不同应用程序所需要的数据有部分相同时，也必须建立各自的文件，而不能共享相同的数据

### ♣ 数据孤立

- ◆ 数据分散管理，许多文件，许多数据格式

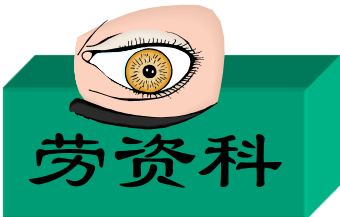
## ♠ 数据的不一致性

- ♣ 由于数据存在很多副本，给数据的修改与维护带来了困难，容易造成数据的不一致性

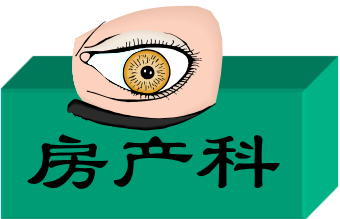
## ♠ 数据联系弱

- ♣ 文件之间相互独立，缺乏联系
- 

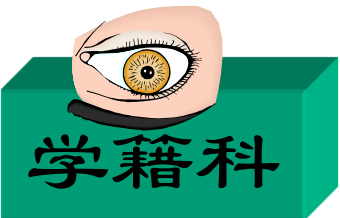
# 文件系统阶段



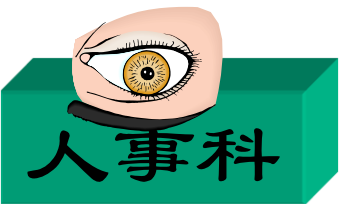
|    |    |    |    |
|----|----|----|----|
| 学号 | 姓名 | 系别 | 补贴 |
|----|----|----|----|



|    |    |    |    |    |
|----|----|----|----|----|
| 学号 | 姓名 | 性别 | 系别 | 住址 |
|----|----|----|----|----|



|    |    |    |    |    |
|----|----|----|----|----|
| 学号 | 姓名 | 系别 | 学分 | 学位 |
|----|----|----|----|----|



|    |    |    |    |    |    |    |
|----|----|----|----|----|----|----|
| 学号 | 姓名 | 性别 | 系别 | 年龄 | 学位 | 出身 |
|----|----|----|----|----|----|----|





# 文件系统阶段

## ♠ 数据查询困难

- ♣ 记录之间无联系
- ♣ 应用自己编程实现
- ♣ 对每个查询都重新编码

## ♠ 数据完整性难于保证





# 数据库系统阶段（60年代后期开始）

## ♠ 背景

- ♣ 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，力求降低





# 数据库系统阶段

## ♠ 数据管理技术进入数据库阶段标志

- ♣ 1968年，美国IBM公司推出层次模型的IMS系统
- ♣ 1969年10月，美国数据系统语言协会（CODASYL）的数据库任务小组（DBTG）发表关于数据库的理论基础
- ♣ 1970年，美国IBM公司的E.F.Codd连续发表论文，提出关系模型，奠定了关系数据库的理论基础







# 数据库系统阶段

## ♠ 特点

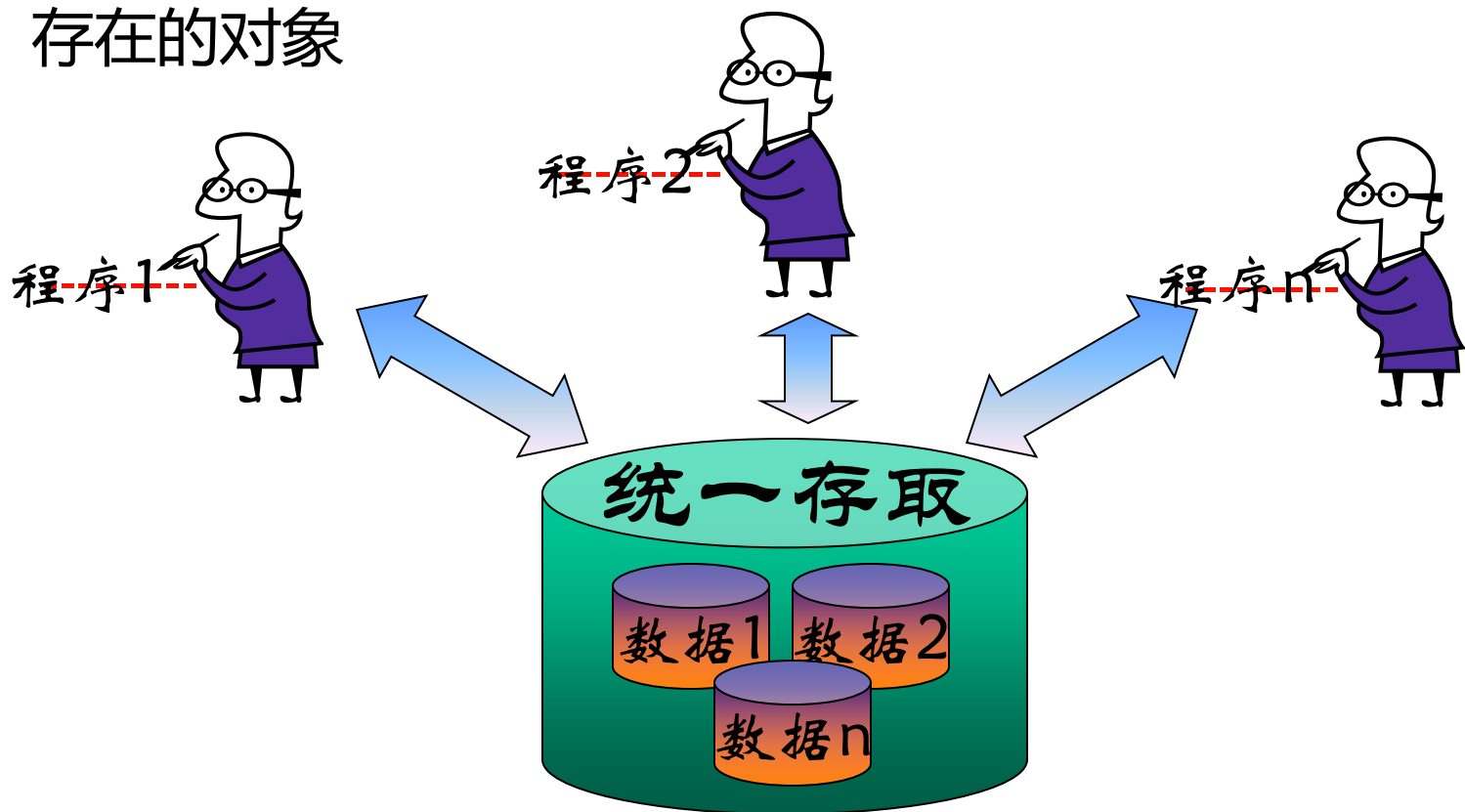
- ♣ 有了数据库管理系统
- ♣ 采用复杂的数据模型表示数据结构
  - ◆ 数据模型不仅描述数据本身的特点，还描述数据之间的联系
  - ◆ 数据不再面向特定的某个/多个应用，而是面向整个应用系统
- ♣ 较高的数据独立性
  - ◆ 分为用户逻辑结构-整体逻辑结构-物理结构三级
- ♣ 数据库系统为用户提供方面的统一接口，也可以使用查询语言或终端命令操作数据库，也可以使用程序方式操作数据库
- ♣ 系统提供四个方面的数据控制功能：
  - ◆ 数据库的恢复、并发控制、数据的完整性和数据的安全性
  - ◆ 保证数据库的数据是安全的、正确的和可靠的
- ♣ 数据操作不一定以记录为单位，也可以数据项为单位，增加灵活性



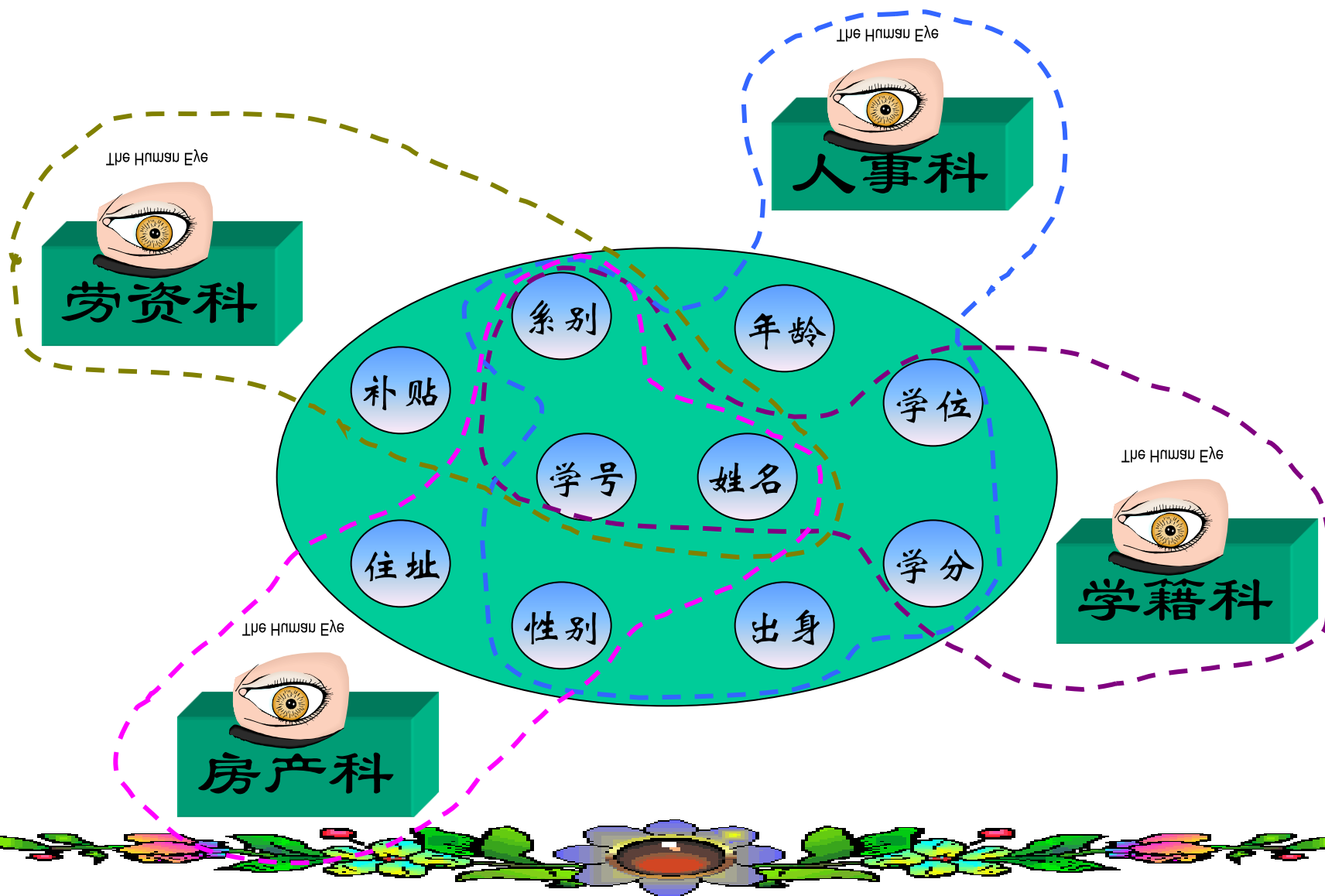
# 数据库系统阶段

## ♠ 数据库观点

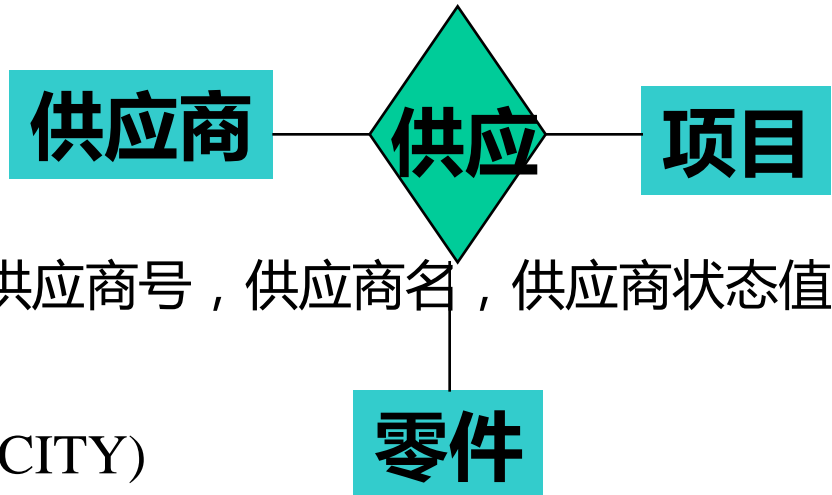
- ♣ 数据不是依赖于处理过程的附属品，而是现实世界中独立存在的对象



# 数据库系统阶段



# 示例



♠ S(SNO, SNAME, STATUS, CITY)

♣ S表示供应商，它的各属性依次为供应商号，供应商名，供应商状态值，供应商所在城市；

♠ P(PNO, PNAME, COLOR, WEIGHT, CITY)

♣ P表示零件，它的各属性依次为零件号，零件名，零件颜色，零件重量，零件存放的城市；

♠ J(JNO, JNAME, CITY)

♣ J表示工程，它的各属性依次为工程号，工程名，工程所在城市；

♠ SPJ(SNO, PNO, JNO, QTY)

♣ SPJ表示供货关系，它的各属性依次为供应商号，零件号，工程号，供货数量。





# 示例

## ♠ 查询

♣ “供应红色零件给北京的工程的供应商姓名”

## ♠ 维护

♣ “不允许供应不存在的零件”



# 示例——基于文件系统

♠ 分别组织几个文件，存储各类对象的记录

CreateFile(S, P, J, SPJ)

ScanFile(J)

找到北京的工程的号码

ScanFile(P)

找到红色零件的号码

ScanFile(SPJ)

找到对应以上两号码的SNO

ScanFile(S)

找到对应以上SNO的供应商姓名



# 示例——基于文件系统

如果InsertFile(SPJ)



ScanFile(P)

判断欲插入的零件号是否在P中

如果DeleteFile(P)



ScanFile(SPJ)

判断欲删除的零件号是否在SPJ中





# 示例——基于数据库系统

♠ 数据统一按表结构存放，设为S，P，J，SPJ

♠ 查询：只需提查询要求，由系统完成查询过程

♣ SELECT SNAME

♣ FROM S, P, J, SPJ

♣ WHERE SPJ.SNO = S.SNO AND SPJ.PNO = P.PNO

♣ AND SPJ.PNO = P.PNO AND J.CITY = “BEIJING”

♣ AND P.COLOR = “RED”

♠ 维护：应用提出完整性约束，系统自动检查

♣ CREATE TABLE SPJ(.....,

♣ FOREIGN KEY (PNO) REFERENCES P(PNO), .....)







# 数据库系统 VS 文件系统

## ♠ 文件系统的弱点

- ♣ 记录之间无联系
- ♣ 难于维护数据的完整性

## ♠ 数据库系统的用武之地

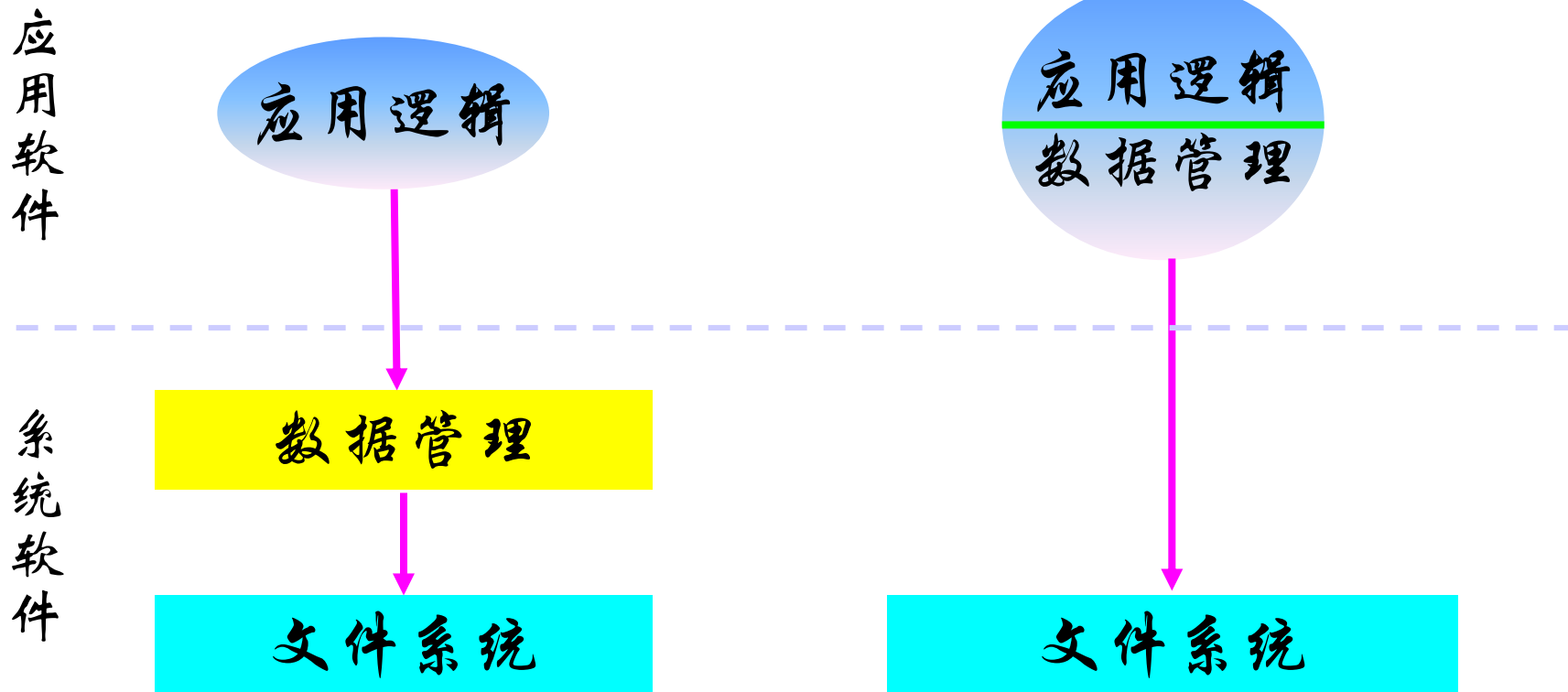
- ♣ 有查询
- ♣ 整体数据结构化

## ♠ 效率两面观

- ♣ 运行效率
- ♣ 开发效率



# 数据库系统 VS 文件系统





# 几个关键的数据库概念

## ♠ 数据库 (Database, DB)

- ♣ 是统一管理的相关数据的集合

## ♠ 数据库管理系统 (Database Management System, DBMS)

- ♣ 位于用户和操作系统之间的一层数据管理软件
- ♣ 总是基于某种数据模型：层次、网状、关系、对象关系、对象

## ♠ 数据库系统 (Database System, DBS)

- ♣ 采用了数据库技术的计算机应用系统

## ♠ 数据库技术

- ♣ 一门研究数据库的结构、存储、管理和使用的软件学科





# 提纲

♠ 数据管理技术的发展

♠ 数据描述

♠ 数据模型

♠ 数据库的体系结构

♠ 数据库系统的构成

♠ 小结





# 数据描述

♠ 数据描述的三个领域

♠ 物理存储介质层次及数据描述

♠ 数据联系的描述





# 数据描述的三个领域

♠ 现实世界

♠ 信息世界

♠ 机器世界

♠ 数据描述的两两种形式





# 现实世界

♠ 存在于人们头脑之外的客观世界

♣ 仓库管理

◆ 货物的存放、进出和检查等

♣ 数据库设计者要进行分析

◆ 根据进库单、储库单、报表统计等，分类、抽取系统所要的数据





# 信息世界

## ♠ 现实世界在人脑的反映

### ♣ 实体

- ◆ 客观存在，可以相互区分的事务（具体、抽象）

### ♣ 实体集

- ◆ 性质相同的同类实体的集合

### ♣ 属性

- ◆ 实体的特性

### ♣ 实体标识符（键）

- ◆ 唯一标识每个实体的属性或属性集







# 机器世界

♠ 信息世界在机器世界中以数据的形式存储

♣ 字段-属性

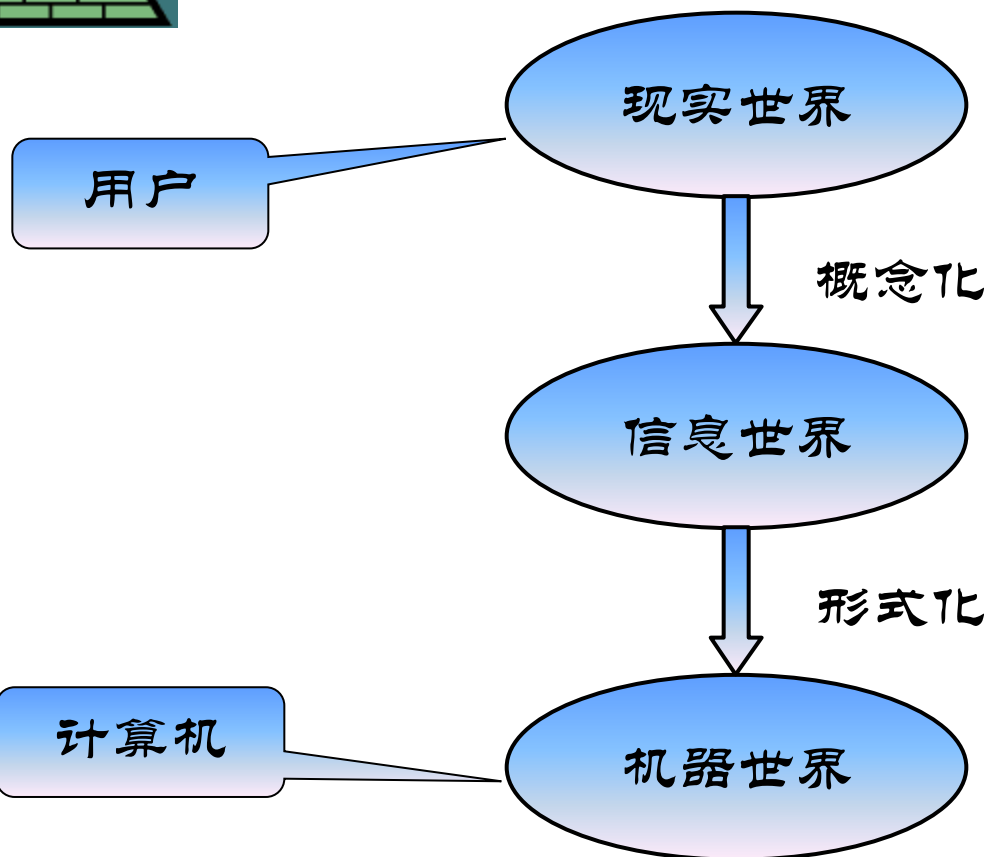
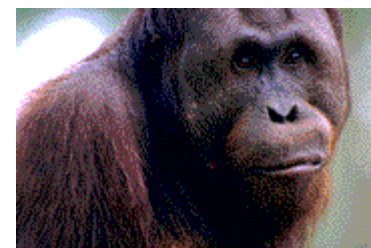
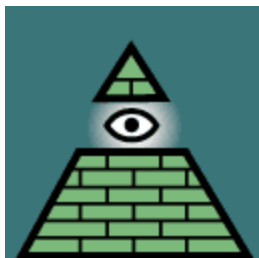
♣ 记录-实体

♣ 文件-实体集

♣ 关键码-实体标识符



# *“computing”*





# 数据描述的两种形式

## ♠ 物理描述

- ♣ 物理数据在存储设备的存储方式，物理数据是实际存放在存储设备上的数据
  - ◆ 物理联系、物理结构、物理文件、物理记录等术语

## ♠ 逻辑描述

- ♣ 描述程序员或用户用于操作的数据形式，抽象化的概念
  - ◆ 逻辑联系、逻辑结构、逻辑文件、逻辑记录等术语





# 提纲

♠ 数据管理技术的发展

♠ 数据描述

♠ 数据模型

♠ 数据库的体系结构

♠ 数据库系统的构成

♠ 小结





# 数据模型

## ♠ 概念

- ♣ 表示实体类型及实体间联系的模型称为“数据模型”

## ♠ 概念数据模型

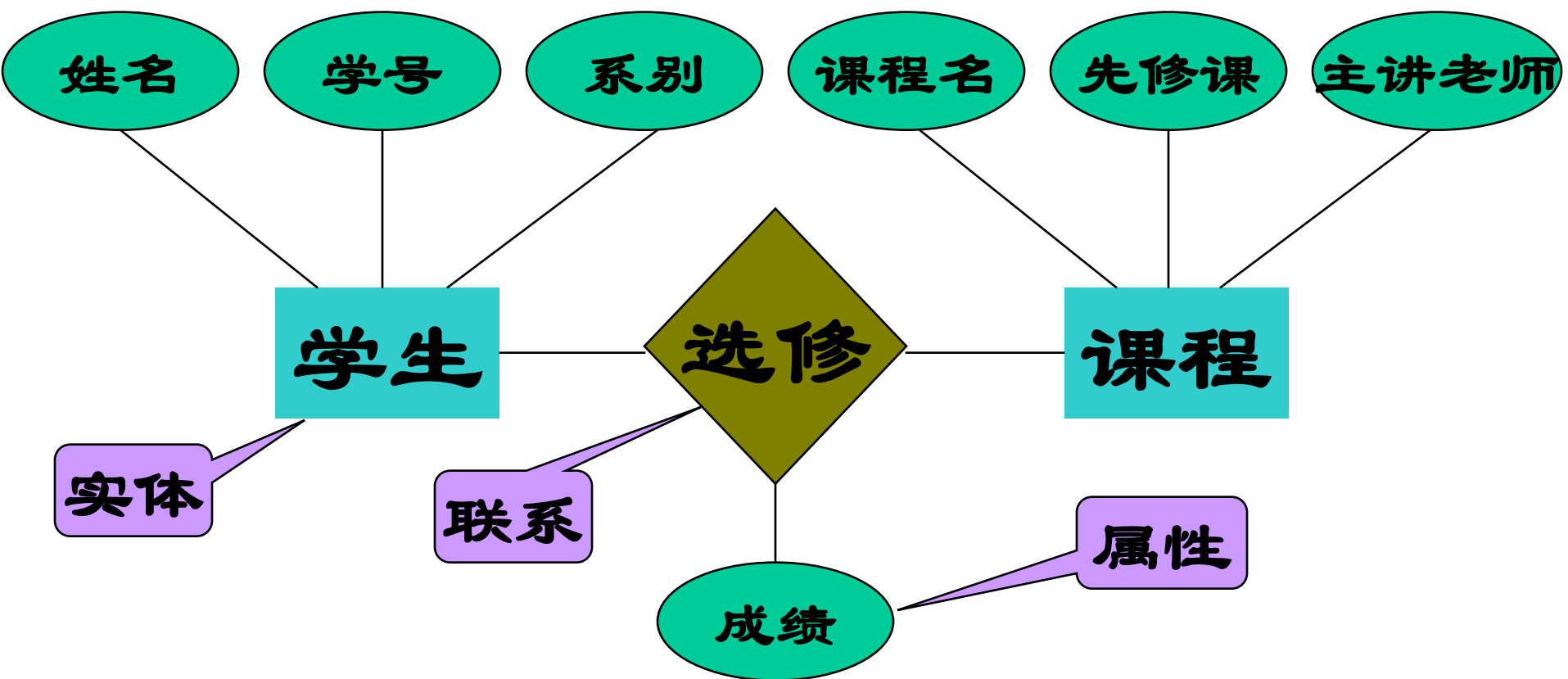
- ♣ ER图

## ♠ 结构数据模型

- ♣ 三个要素：数据结构、数据操作、完整性约束
- ♣ 几种常见的模型
  - ◆ 层次
  - ◆ 网状
  - ◆ 关系
  - ◆ 对象
  - ◆ 对象关系



# 概念数据模型——E/R





# 结构数据模型的三要素

## ♠ 数据结构

- ♣ 描述系统的静态特性，即组成数据库的对象类型
- ♣ 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，关系模型中的外码
- ♣ 在数据库系统中一般按数据结构的类型来命名数据模型





# 结构数据模型的三要素

## ♠ 数据操作

- ♣ 描述系统的动态特性，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





# 结构数据模型示例——关系模型

♠ 用二维表来表示实体集

♠ 用外键表示实体间相互联系

♠ 关系模型是若干个关系模式组成的集合

元组

属性

| 学号  | 姓名 | 年龄 | 性别 | 系号  |
|-----|----|----|----|-----|
| S01 | 张军 | 21 | 男  | D01 |
| S02 | 李红 | 22 | 女  | D01 |
| S03 | 王伟 | 19 | 男  | D02 |



# 结构数据模型示例——关系模型

## ♠ 优点

- ♣ 简单，表的概念直观、单一，用户易理解
- ♣ 非过程化的数据请求，数据请求可以不指明路径
- ♣ 数据独立性，用户只需提出“做什么”，无须说明“怎么做”
- ♣ 坚实的理论基础
- ♣ 标准查询语言SQL

## ♠ 缺点

- ♣ 效率低

## ♠ 代表

- ♣ DB2, ORACLE, SYBASE, SQL SERVER, INFOMIX, ACCESS, FOXPRO





# 提纲

♠ 数据管理技术的发展

♠ 数据描述

♠ 数据模型

♠ 数据库的体系结构

♠ 数据库系统的构成

♠ 小结





# 数据库的体系结构

## ♠ 数据库模式

- ♣ 数据的抽象

- ♣ 数据库的结构描述

## ♠ 数据库

- ♣ 数据库模式的一个“实例”

♠ 数据库模式相对稳定

♠ 数据库内容随时变化





# 数据库的体系结构

## ♠ 模式的分级

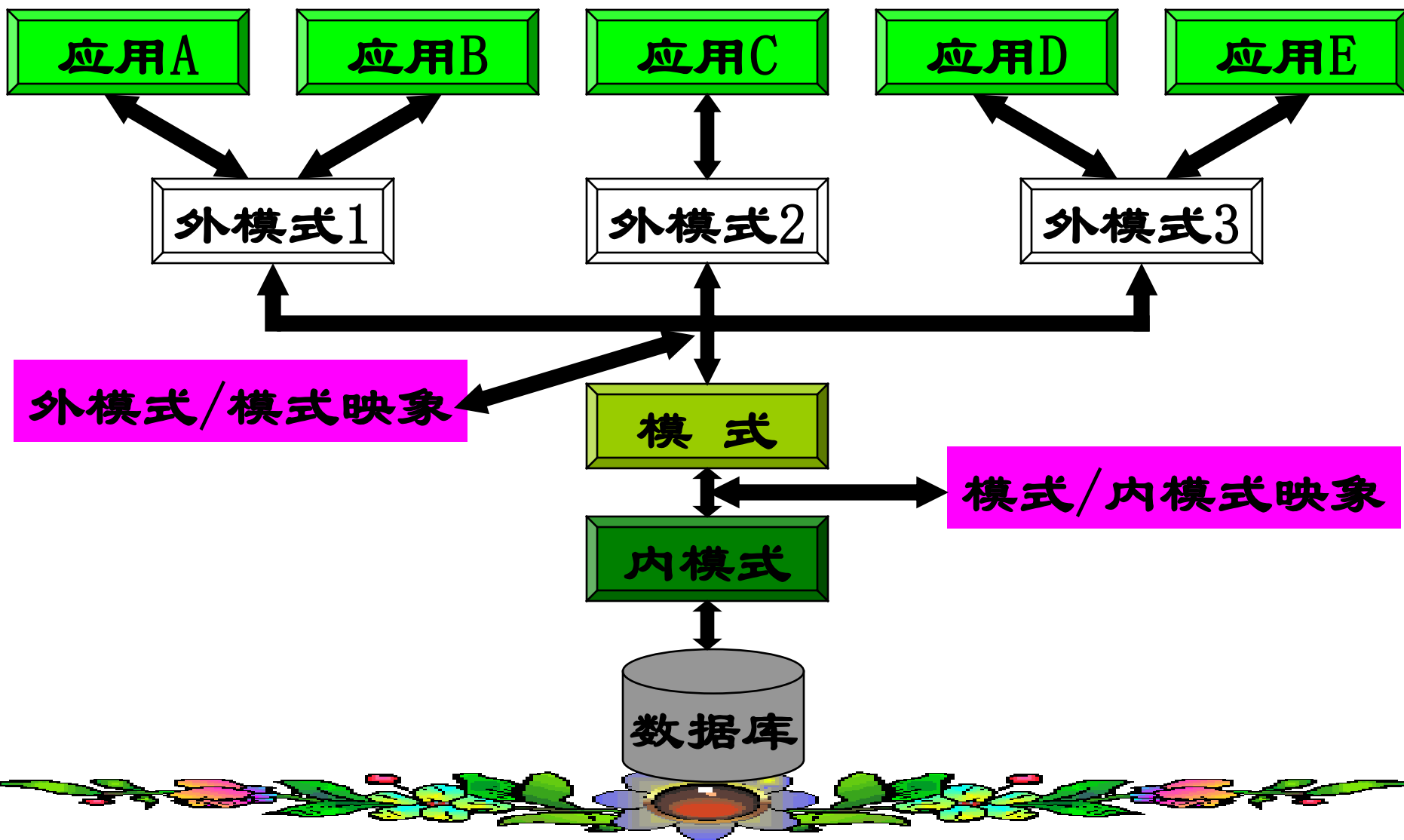
- ♣ 为了提高数据的物理独立性和逻辑独立性，使数据库的用户观点，即用户看到的数据库，与数据库的物理方面，即实际存储的数据库区分开来，数据库系统的模式是分级的

## ♠ 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



# 三级模式/两级映象





# 三级模式


## ♠ 外模式(Sub-Schema)

- ♣ 又称子模式、用户模式
- ♣ 用户的数据视图
- ♣ 是数据的局部逻辑结构，模式的子集

## ♠ 模式(Schema)

- ♣ 又称概念模式
- ♣ 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
  - ♣ 是数据的物理结构及存储方式
- 



# 两级映象/两级独立性

## ♠ 外模式/模式映象

- ♣ 定义某一个外模式和模式之间的对应关系，映象定义通常包含在各外模式中
- ♣ 当模式改变时，修改外模式/模式映象，使外模式保持不变，从而应用程序可以保持不变，称为数据的逻辑独立性

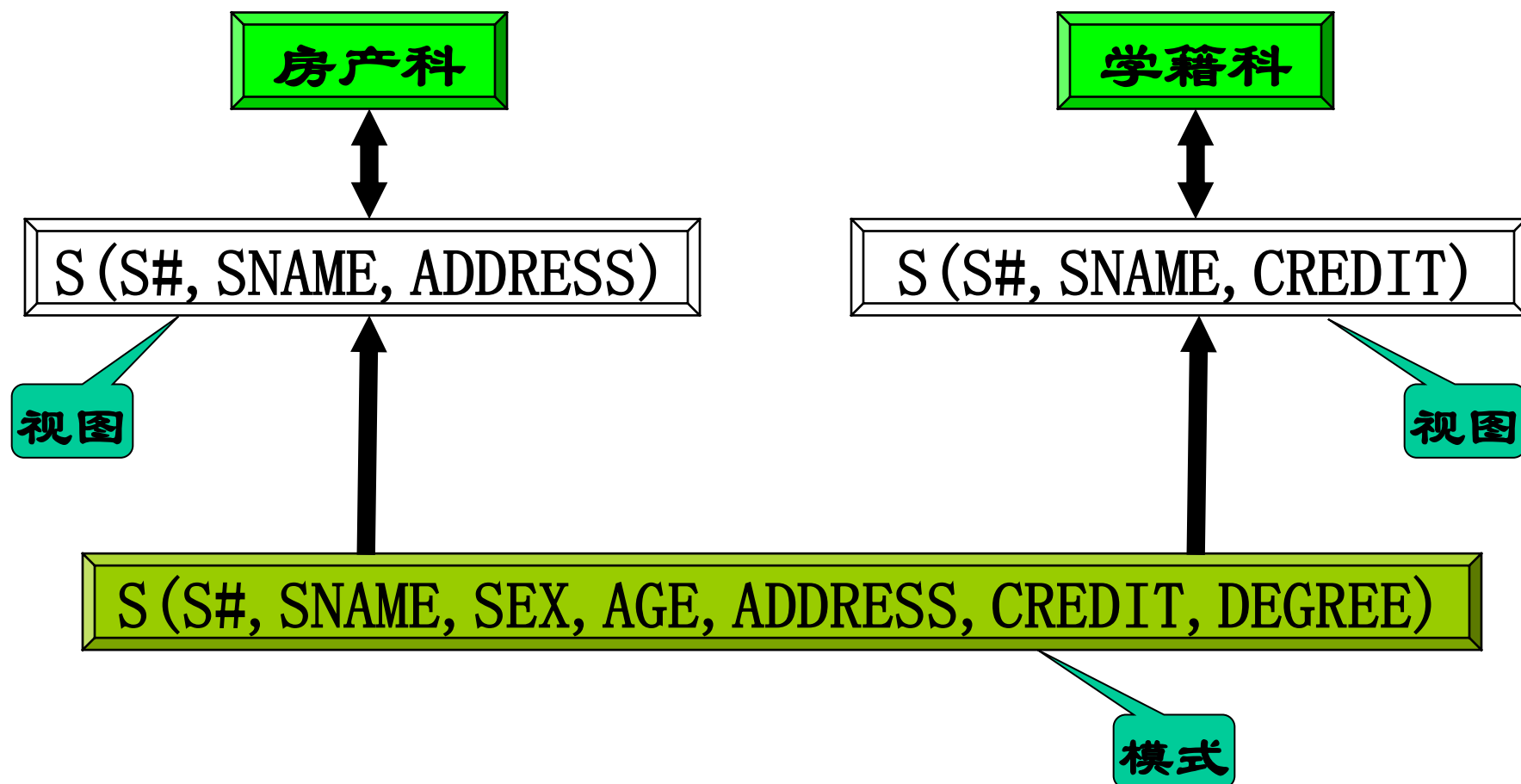
## ♠ 模式/内模式映象

- ♣ 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，修改模式/内模式映象，使模式保持不变，从而应用程序可以保持不变，称为数据的物理独立性





# 数据库模式



# 数据库模式

房产科

学籍科

S (S#, SNAME, ADDRESS)

S (S#, SNAME, CREDIT)

视图

视图

B (S#, SNAME, SEX, AGE, ADDRESS, CREDIT)

模式

M (S#, SNAME, SEX, AGE, ADDRESS, CREDIT, DISSERTATION)





# 提纲

♠ 数据管理技术的发展

♠ 数据描述

♠ 数据模型

♠ 数据库的体系结构

♠ 数据库系统的构成

♠ 小结





# 数据库系统的构成

♠ 几个相关概念

♠ 数据库系统的软硬件层次

♠ 数据库系统的主要成分

♠ 数据库管理系统的层次结构及其功能

♠ 数据库系统结构

♠ 数据库系统的效益





# 几个相关概念

## ♠ 数据库

- ♣ 数据的集合
- ♣ 由DBMS统一管理，多用户共享

## ♠ 数据库管理系统DBMS

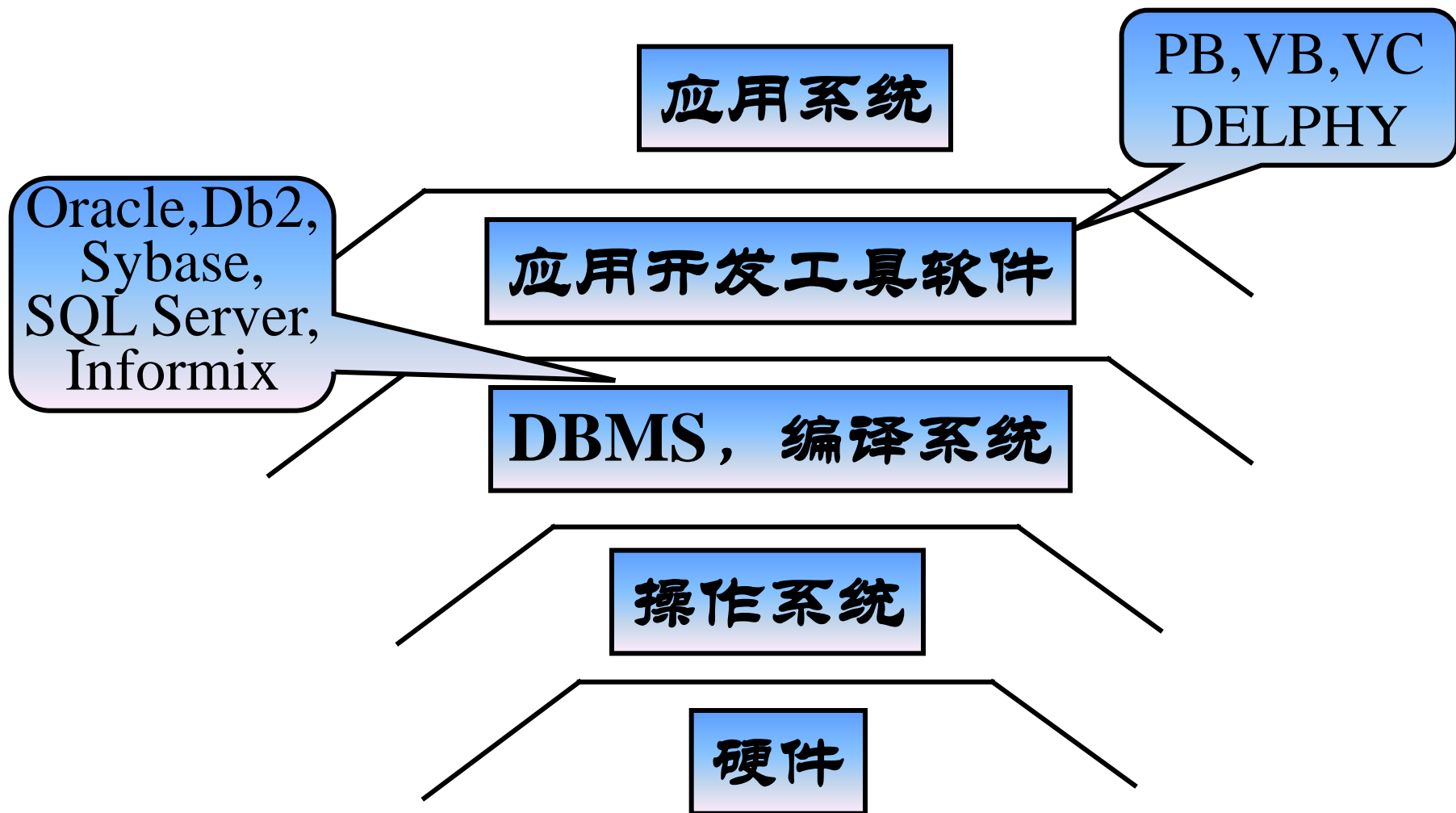
- ♣ 系统软件，对数据库进行统一管理和控制

## ♠ 数据库系统

- ♣ 带有数据库的整个计算机系统，包括硬件、软件、数据、人员



# 数据库系统的软硬件层次





# 数据库系统的主要成分

## ♠ 硬件

- ♣ 大内存，放得下OS，DBMS核心，系统缓冲区，用户工作区等
- ♣ 大容量、直接存取的外存设备
- ♣ 作数据备份的磁带

## ♠ 软件

- ♣ OS，DBMS，高级语言编译系统及其与数据库的接口，应用开发工具，应用系统

## ♠ 数据

- ♣ 目标数据：数据本身
- ♣ 描述数据：对数据的说明信息





# 数据库系统的主要成分

## ♠ 用户

### ♣ 最终用户

- ◆ 通过应用系统的用户接口（菜单等）使用数据库

### ♣ 应用程序员

- ◆ 基于外模式来编写应用程序

### ♣ 专业用户（系统分析员）

- ◆ 负责应用系统的需求分析和规范定义，确定系统的软硬件配置，参与数据库模式设计

### ♣ 数据库管理员DBA

- ◆ 负责数据库的全面管理和控制







# 数据库系统的主要成分

## ♠ DBA的重要性

### ♣ 重要资源

- ◆ 维护整个组织的信息资源

### ♣ 共享资源

- ◆ 多用户共享，需要统一管理、协调、监控

## ♠ DBA职责

### ♣ 建库方面

- ◆ 确定概念模式、外模式、内模式、存储结构、存取策略
- ◆ 负责数据的整理和装入





# 数据库系统的主要成分

## ♣ 用库方面

- ◆ 定义完整性约束条件
- ◆ 对数据库访问的授权，规定数据的保密级别、用户权限
- ◆ 完整性约束的说明，监督和控制数据库的运行情况
- ◆ 制定后援和恢复策略，负责故障恢复

## ♣ 改进方面

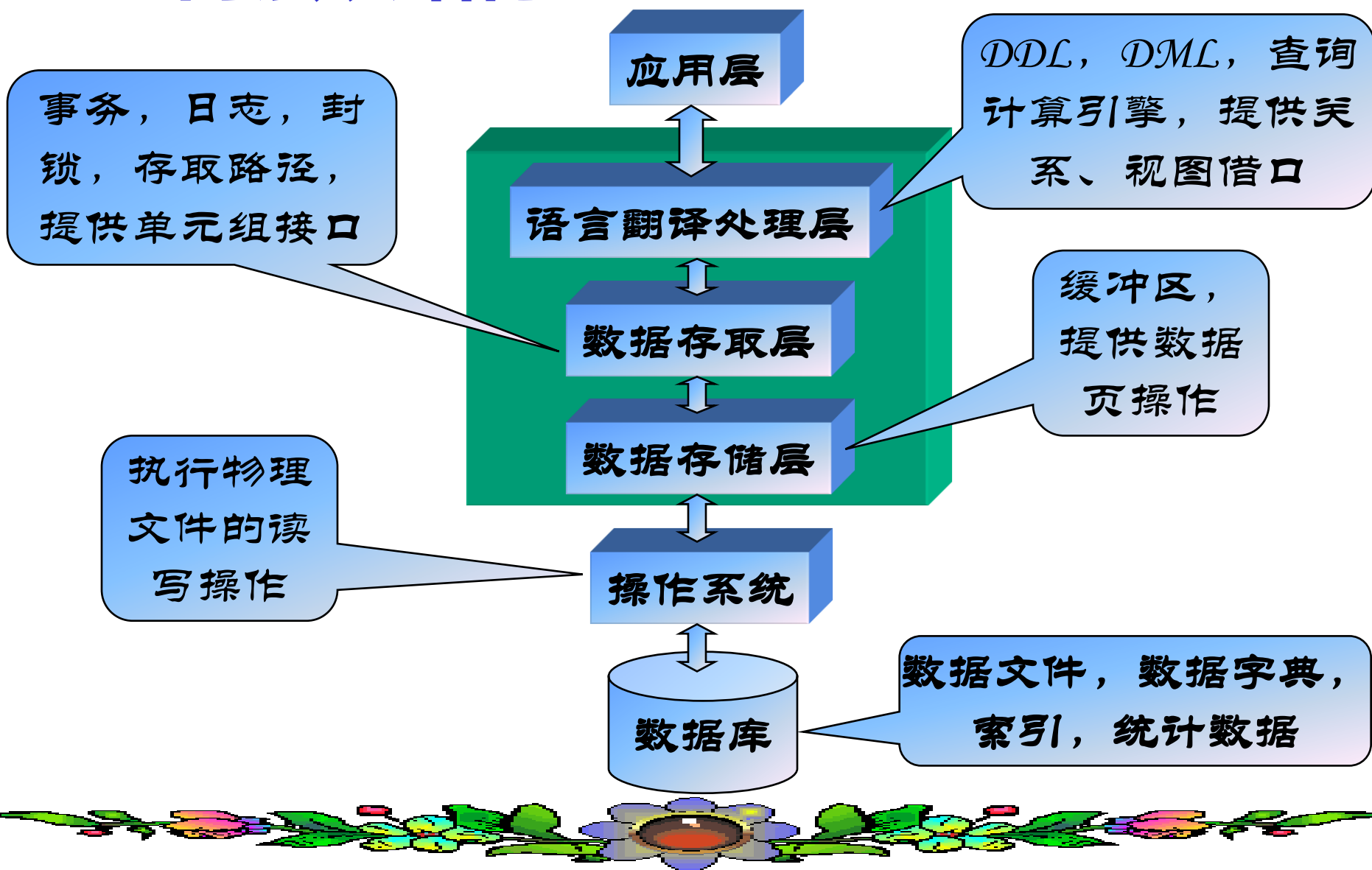
- ◆ 监督分析系统的性能（空间利用率，处理效率）
- ◆ 数据库重组织，物理上重组织，以提高性能
- ◆ 数据库重构造，设计上较大改动，模式和内模式修改

## ♠ AutoAdmin

<http://research.microsoft.com/research/dmx/autoadmin/>



# DBMS的层次结构



# DBMS的主要功能

## ♠ 数据库定义功能

- ♣ 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

## ♠ 数据存取功能

### ♣ 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

### ♣ DML语句执行方式

- ◆ 宿主型

- 预编译和增强编译

- ◆ 自含型

- 解释执行





# DBMS的主要功能

## ♠ 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

## ♠ 数据字典

- ♣ 系统目录：存取和管理数据的依据
  - ♣ 存放三级结构定的数据库
  - ♣ 是元数据
- 



# 元数据

## ♠ 元数据 (meta-data)

- ♣ 描述数据的数据
- ♣ 描述数据的含义和性质，以便更好地理解、管理和使用数据的数据
- ♣ 示例

数据：1, 1, 2, 3, 5, 8, 13.....

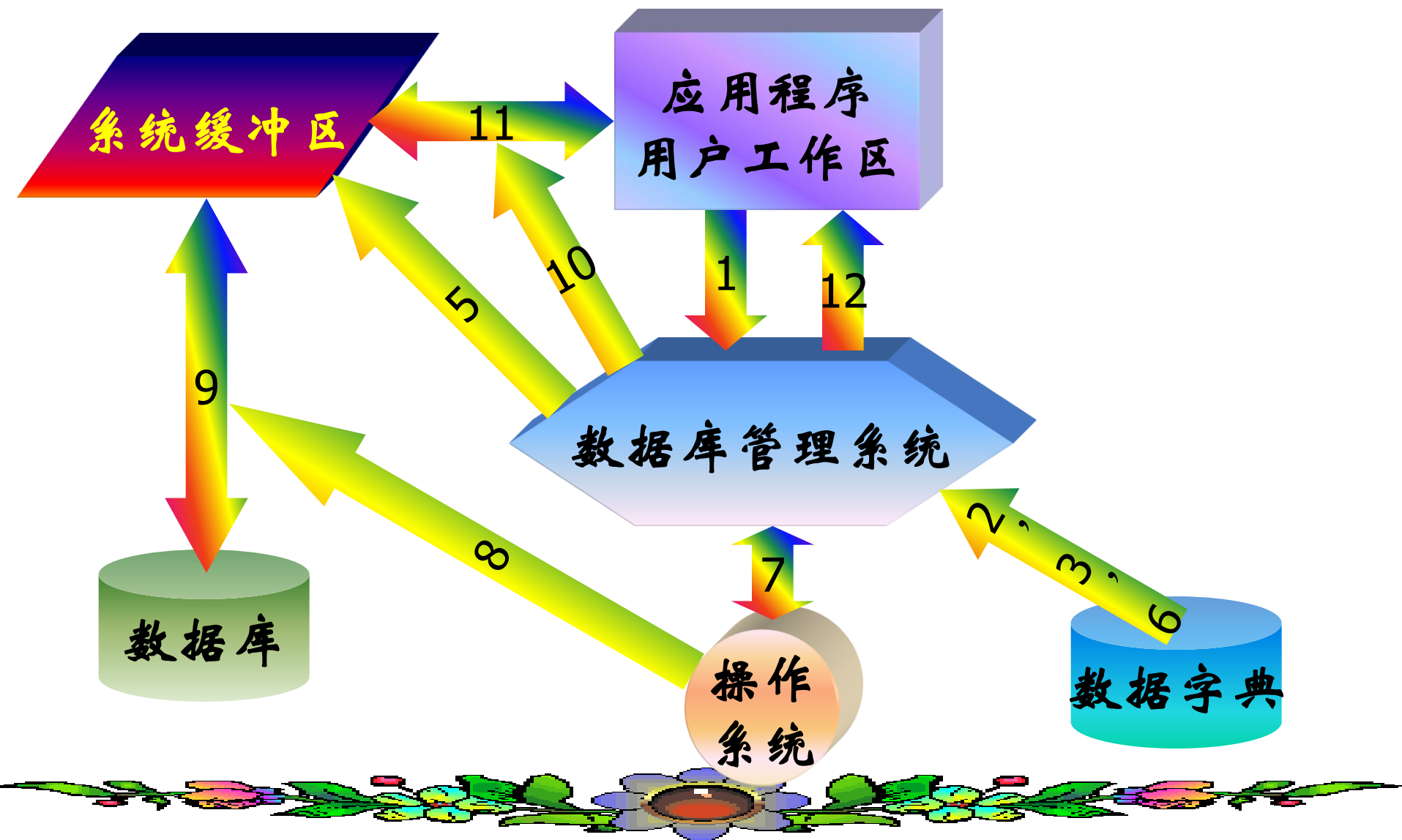
元数据：
$$\frac{1}{\sqrt{5}} \left[ \left( \frac{-1+\sqrt{5}}{2} \right)^{n+1} - \left( \frac{-1-\sqrt{5}}{2} \right)^{n+1} \right]$$

数据：图书馆中的书籍

元数据：标题、作者、关键词、ISBN号



# DBMS的运行过程







# DBMS的运行过程

## ♠ Step 1

- ♣ 用户向DBMS发出调用数据库数据的命令

## ♠ Step 2

- ♣ DBMS对命令进行语法检查、语义检查、存取权限检查，决定是否执行该命令

## ♠ Step 3

- ♣ DBMS执行查询优化，把命令转换为一串单记录的存取操作序列

## ♠ Step 4

- ♣ 执行存取操作序列（反复执行以下各步，直至结束）

## ♠ Step 5

- ♣ DBMS首先在缓冲区内查找记录，若找到转10，否则转6

## ♠ Step 6

- ♣ DBMS查看存储模式，决定从哪个文件存取哪个物理记录





# DBMS的运行过程

## ♠ Step 7

- ♣ DBMS根据6的结果，向操作系统发出读取记录的命令

## ♠ Step 8

- ♣ 操作系统执行读取数据的命令

## ♠ Step 9

- ♣ 操作系统将数据从数据库存储区送到系统缓冲区

## ♠ Step 10

- ♣ DBMS根据用户命令和数据字典的内容导出用户所要读取的数据格式

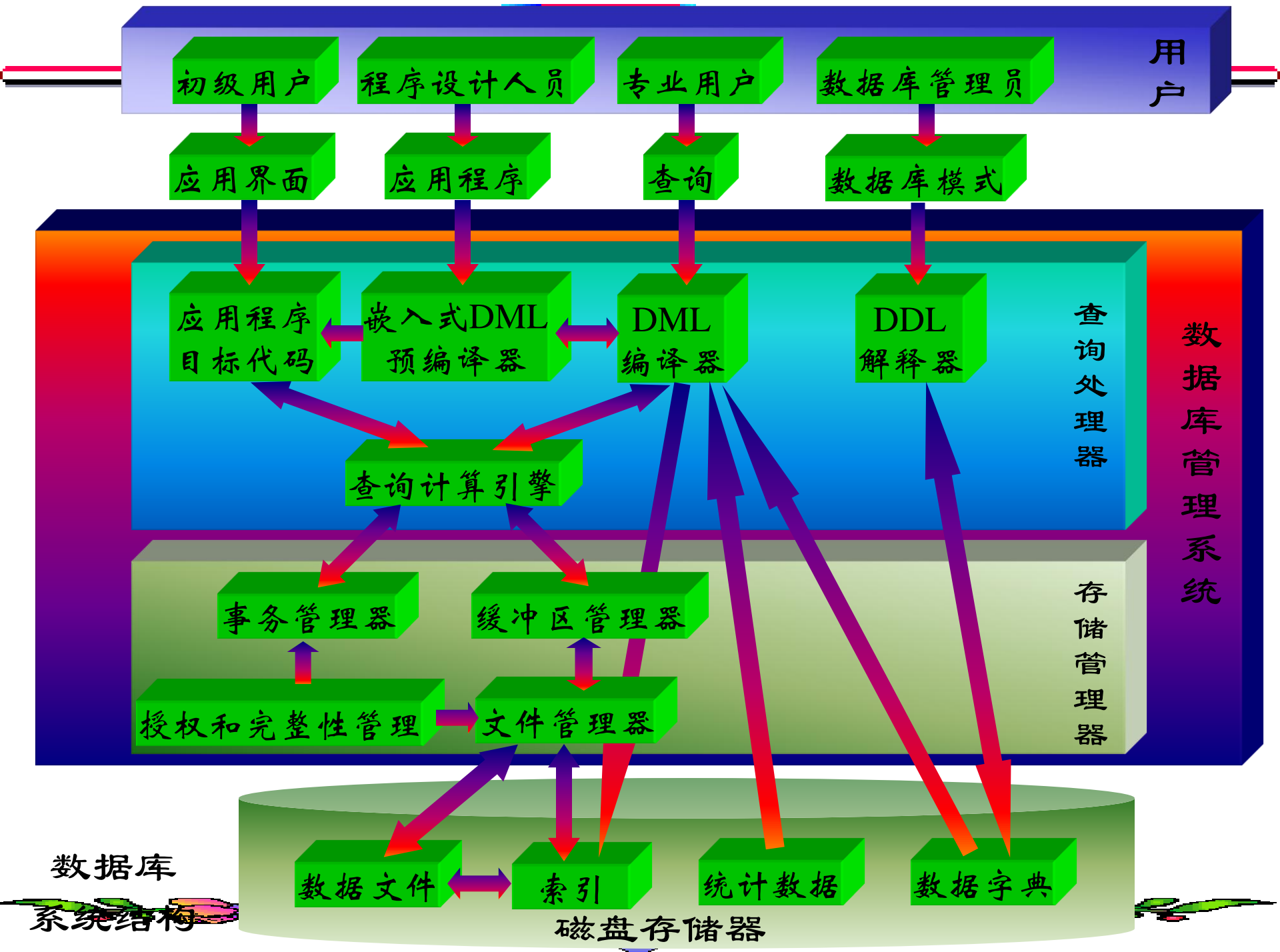
## ♠ Step 11

- ♣ DBMS将数据记录从系统缓冲区传送到用户工作区

## ♠ Step 12

- ♣ DBMS将执行状态信息返回给用户







# 数据库系统的效益

♠ 灵活性

♠ 简易性

♠ 面向用户

♠ 数据控制

♠ 程序设计方便，加快应用系统的开发进度

♠ 减少了程序维护的工作量

♠ 标准化





# 提纲

♠ 数据管理技术的发展

♠ 数据描述

♠ 数据模型

♠ 数据库的体系结构

♠ 数据库系统的构成

♠ 小结





# 小结

♠ 数据管理技术发展的三个阶段

♠ 数据库技术的基本概念

♠ 数据模型

♣ ER图

♣ 关系模型

♠ 数据库体系结构

♣ 三级结构、两级映象/独立性

♠ 数据库管理系统

♠ 数据库系统

♠ ○ ○ ○



