

数据库系统原理——潘鹏

13006147552

- 教材

- 王珊，萨师煊著。数据库系统概论(第五版)，北京：高等教育出版社

- 相关工具软件

- DBMS（数据库管理系统）：**

- 达梦、OpenGauss、OceanBase、。。。

- SQL Server、ORACLE、MySQL、Postgres、。。。

- Develop（客户端程序开发软件）：** JAVA、C#、。。。

课程教学建设

- 制作慕课 “数据库系统原理” 并坚持发布（本学期第8次开课，且内容有更新）。
- 引入CMU15-445部分教学内容（系统内部实现技术）。
- 实践课融合头歌平台考察模式（杨茂林、赵小松等）
- 数据库竞赛（谢美意、赵小松等）
- 数据库系统软件进阶设计（高年级课程系统综合能力培养，谢美意、左琼等）
- 与华为签订战略合作协议，共同推进课程建设，并结合企业、行业生产和科研的实际需求。

线上学习资源

- 华中科技大学计算机学院 “数据库系统原理” 慕课，链接地址（第8次开课）：

<https://www.icourse163.org/course/HUST-1449788170>

- 华为相关在线课程系列

<https://edu.huaweicloud.com/roadmap/colleges.html>

华为在线课程数据库系列，通过技术领域选择“鲲鹏”后，查找数据库或者openGauss的课程进入，或者直接搜索数据库相关课程

教学组织工作

- 学在华科大课程平台（发布课件/通知、平时作业提交），
<https://hust.fanya.chaoxing.com/>
- 每个班级请推荐一位班级干部，并提供日常联系手机号等信息，协助管理课堂和组织联系工作。

2023-2024第二学期 ▼

请输入课程名称 🔍

当前展示的是 2023-2024第二学期 课程，可切换学期筛选课程或查看全部课程



数据库系统原理实践
课程编号：w108029 | 教师姓名：潘鹏
院校：华中科技大学
未完成教学工作 ⚠



数据库系统原理
课程编号：0803574 | 教师姓名：潘鹏
院校：华中科技大学
未完成教学工作 ⚠

微助教课堂签到

- 课堂名称：数据库系统原理大数据22级
- 课堂编号：MR178



二维码有效期2024-4-11

课程日常交流及通知发布QQ群

- 课程日常交流及通知发布QQ群

群号：787818611

入群问题答案为课堂教室编号：

S509

课程成绩构成

- 课堂测验占15% (注：早先慕课成绩占15%现在不再要求，但希望同学们积极选修慕课，充分利用已有教学资源，同时也是对老师工作的支持)
- 平时考察及作业占15%
- 考试卷面成绩占70%

数据处理的相关工作？

- 数据的寻址、逻辑比较、计算、简单处理过程
 - 从语言开始，例如汇编、C语言
- 数据的逻辑结构、操作算法，用数据结构描述现实世界的事物，并提供数据读、写的算法。
 - 从数据结构开始
- 文件的存取、并发任务的执行、临界资源的使用、缓存的利用，存储和计算资源的合理分配与使用方法。
 - 从操作系统开始
- 更完整的语义描述、数据操作行为、并发控制、安全性控制、完整性控制、故障容错机制、工程中的应用
 - 从数据库系统原理这门课开始

数据处理的相关工作？

- 系统软件的需求  DBMS:
Database Management System

相对独立的视角

完整、系列的功能

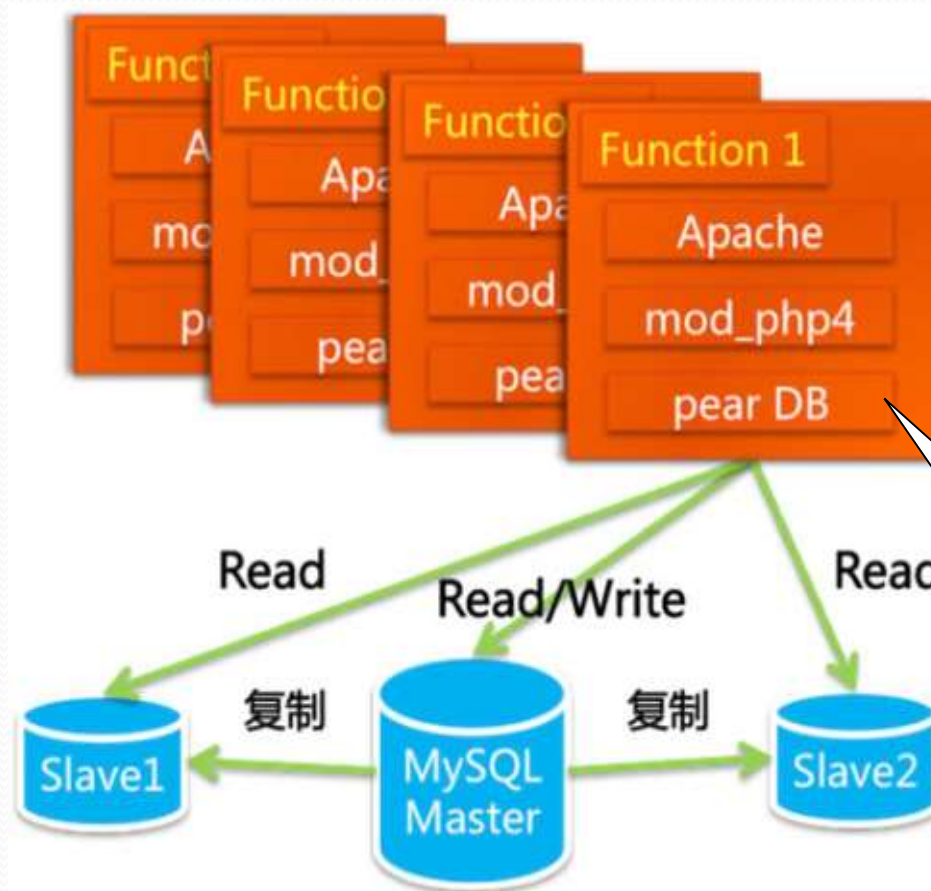
性能、稳定性

专业、易于集成

在计算机系统中有一席之地

数据库在哪？

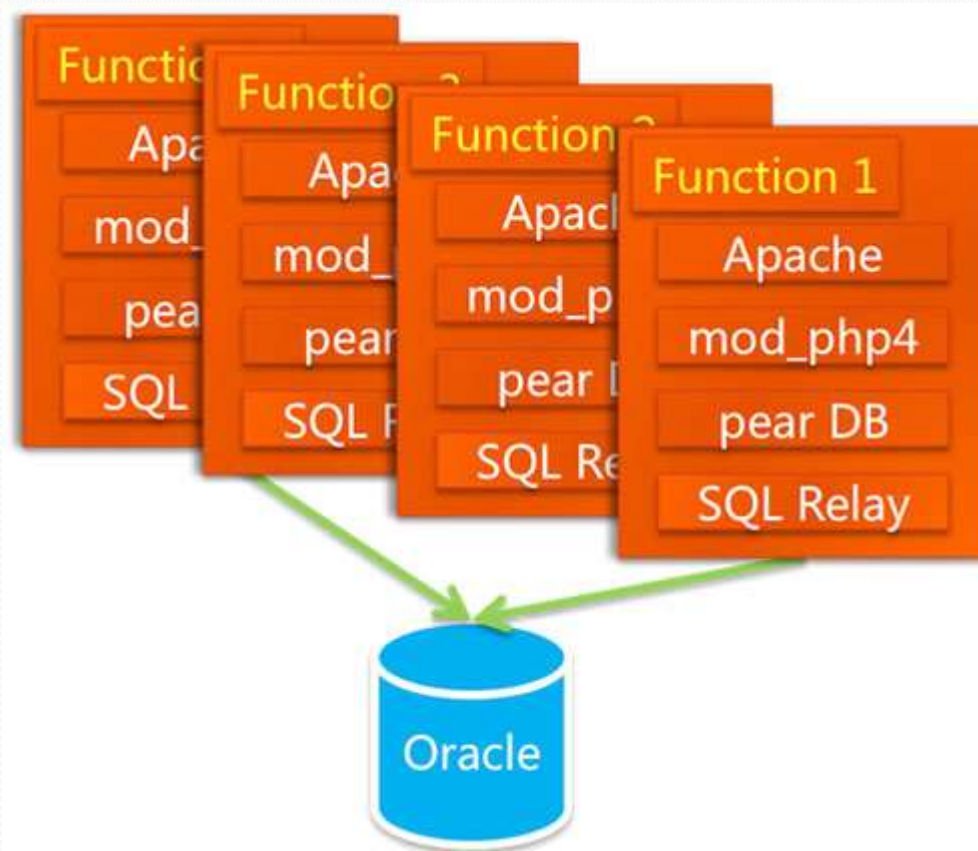
LAMP(Linux+Apache+MySQL+PHP)



Pear DB:
PHP模块，
负责数据访问层。

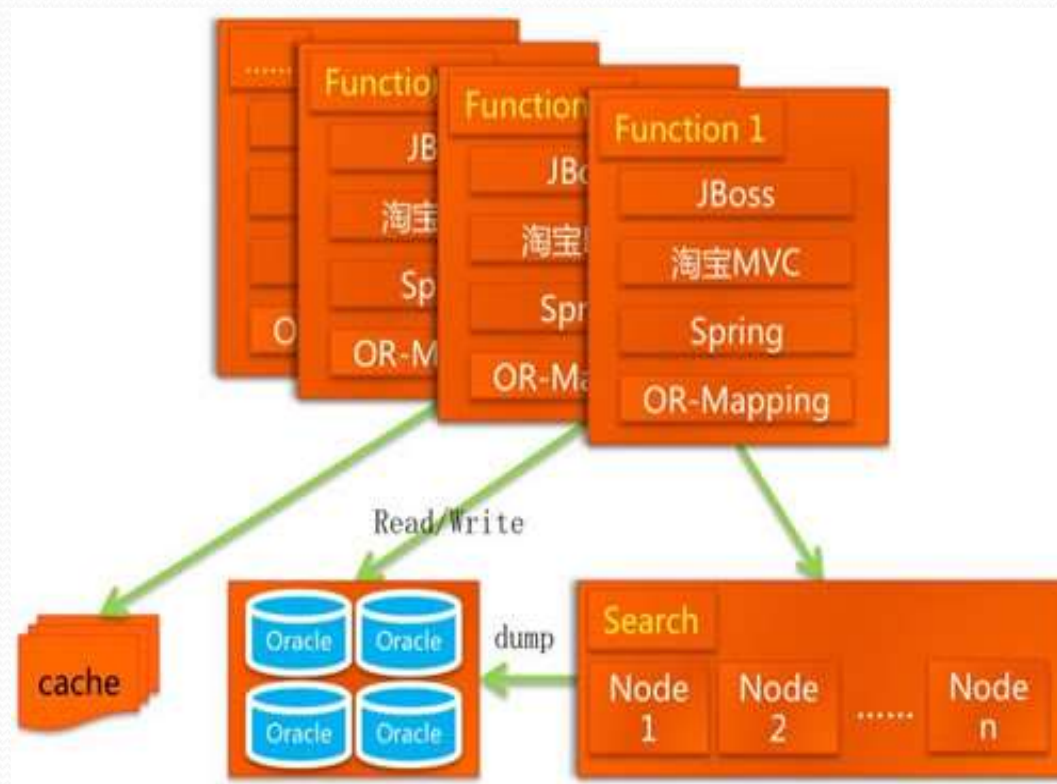
数据库在哪？

ORACLE本地后台DB+连接池



数据库在哪？

Jboss+Spring+cache+分库分表

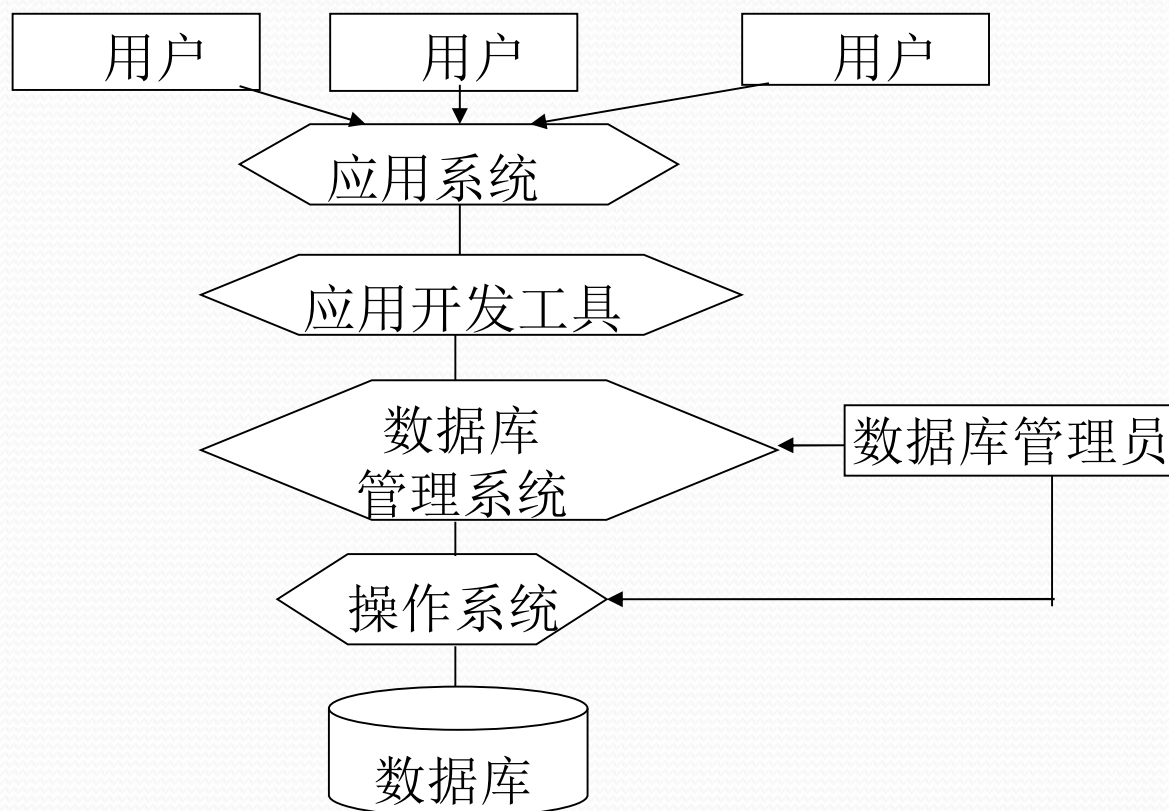


- 应用服务器Jboss，控制层Spring、OR-Mapping持久层。

初步认识数据库系统

- 体系结构

- 系统组成



- 包含系统软件

数据库管理系统
DBMS

操作系统
OS

数据库是计算机科学的重要分支

数据管理的理论与方法、技术。

重要：有定期举行了多年的国际学术会议（且具有权威性，美国计算机协会ACM——Association for Computing Machinery, *IEEE* ——*Institute of Electrical and Electronics Engineers, Inc.*），有相关的重要期刊。

三大会议

ICDE——*International Conference on Data Engineering*

SIGMOD——*Special Interest Group on Management Of Data*

VLDB——*International Conference on Very Large Data Bases*

期刊

TKDE——*IEEE Transactions on Knowledge and Data Engineering*

中国数据库学术会议：NDBC （CCF）

2020，湖北，武汉，
华中科技大学

研究的发展

数据库的需求是在不断发展的，其发展也推动了数据工程领域研究的发展，例如：

分布式数据库、网格数据库，

层次、半结构、无结构等泛结构化数据，

海量数据（传统数据库难以处理，因为索引本身就会变得非常大，数据挖掘），

大数据（处理、管理、存储、分析）

数据库的用途

组织、存储、获取、维护数据

- 解决物理存储的问题，涉及物理优化、备份与恢复（异地）。
- 提供高性能的使用和分析机制。

- 数据共享

- 局部与局部之间
 - 不同类型的应用之间
 - 历史与现状之间

联机处理——银行转帐业务；
批处理——大批量导入、装载、处理数据，现有的数据库管理系统都提供相应的批处理工具；
复杂分析应用——数据仓库、数据挖掘

广泛的应用背景

- 数目多
- 快速处理大量数据
- 各个行业:

国标

科教、电力、通讯、交通、金融、军事、人事、
医疗、生产、地理信息、水文、水利工程。。。。。

- 基础应用

信息世界，一期工程、二期、。。。



- 数据库的建设规模、数据库信息量的大小和使用频度
已成为衡量一个国家信息化程度的重要标志。

课程内容

第1章 绪论

产生与发展、数据模型、模式、系统结构
抽象方法、计算机方法论

第2章 关系数据库

基本概念、操作概述、关系代数基础
代数运算、谓词演算

第3章 关系数据库标准语言SQL

掌握基本的操作语句
一种英文文法及其语义内涵

第4章 数据库安全性

基本概念
计算机安全的基本知识和策略

第5章 数据库完整性

完整性的范畴、机制

数据的语义约束的描述和实现机制

第6章 关系数据理论

问题的提出、范式的概念及应用、公理系统
集合论、闭包的概念及其算法应用

第7章 数据库设计

设计的基本步骤、设计的主要工作
软件工程，需求-设计-实现

第8章 数据库编程（不作为课堂教学内容）

嵌入式sql的使用、odbc工作原理
数据库与其它开发语言的整合

课程内容（续）

第8章 关系数据库引擎基础（参考CMU）

数据库存储、缓存、查询处理等

系统内部的数据结构、算法

第9章 关系查询处理和查询优化

优化的原因、如何优化

数据结构的算法I/O复杂度分析，启发式方法

第10章 数据库恢复技术

以事务为语义单位，以日志为核心

计算机的状态的记录与持久化机制

第11章 并发控制

并发的冲突、锁、可串行化

事务背景下计算机资源的调度策略

课程基本要求

- 结合关系型数据库系统深入理解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E-R模型的概念和方法，关系数据库规范化理论和数据库设计方法，通过上机实习的训练，初步具备进行数据库应用系统开发的能力。
- 掌握数据库系统核心机制：安全性与完整性控制、恢复、并发控制。
- 理解数据库查询执行基本过程及其优化思想

关于课程的学习

从课本的目录结构来看，存在多个主题（问题分支），通过学习整理出：**知识体系和理论框架**。

工程认证：**认识问题**（科学内涵、关键环节）、**模型抽象及其应用**（关键因素分析）、形成并完善**方案**（运用文献、知识，综合分析）。

举一反三、分析全面。

第1章 绪论

- 数据 (Data)
- 数据库 (DataBase, DB)
- 数据库管理系统 (DBMS)
- 数据库系统 (DBS)

■ 数据

- 对现实世界中客观事物的符号表示
- 可以是数值数据，也可以是非数值数据，如声音、图像、结构化的记录等
- 计算机中数据
 - 能输入计算机，并能为其处理的符号序列（I/O及其规则）
- 数据与其语义不可分

(0005794, 601, 刘武, 1, 1946.08.26, 01)
(工号, 部门编号, 姓名, 性别, 出生日期, 民族)

人力资源科 男 汉族

科教、电力、通讯、交通、金融、军事、人事、医疗、生产、地理信息、水文、水利工程。。。。

形式、取值、内容都很丰富

关于数据

- 网络——路
- OS、DBMS、APP——车
- 数据——货

硬件的使用期——5年

软件的使用期——5~10年

数据的使用期——30年

会计法——原始凭证要保存不少于15年
保险订单。。。

只要有程序，就会有数据



■ 数据库 (Database *file*)

不是data warehouse



存放数据的“仓库”

存储在计算机的存储设备上

按一定的格式组织、描述和存储

较小的冗余度

数据独立性

易扩展

可共享

■ 数据库管理系统(DBMS)

- 系统软件，数据库系统的一个重要组成部分
- 科学地组织和存储数据，高效地获取和维护数据
- 位于用户与操作系统之间

- 具有下述典型功能：

- 数据定义功能 – DDL (如Create)
- 数据操作功能 – DML(如Select, Delete, Insert, Update)
- 数据库的运行管理—DCL（统一管理、控制）
- 数据库的建立和维护功能（监视、分析）

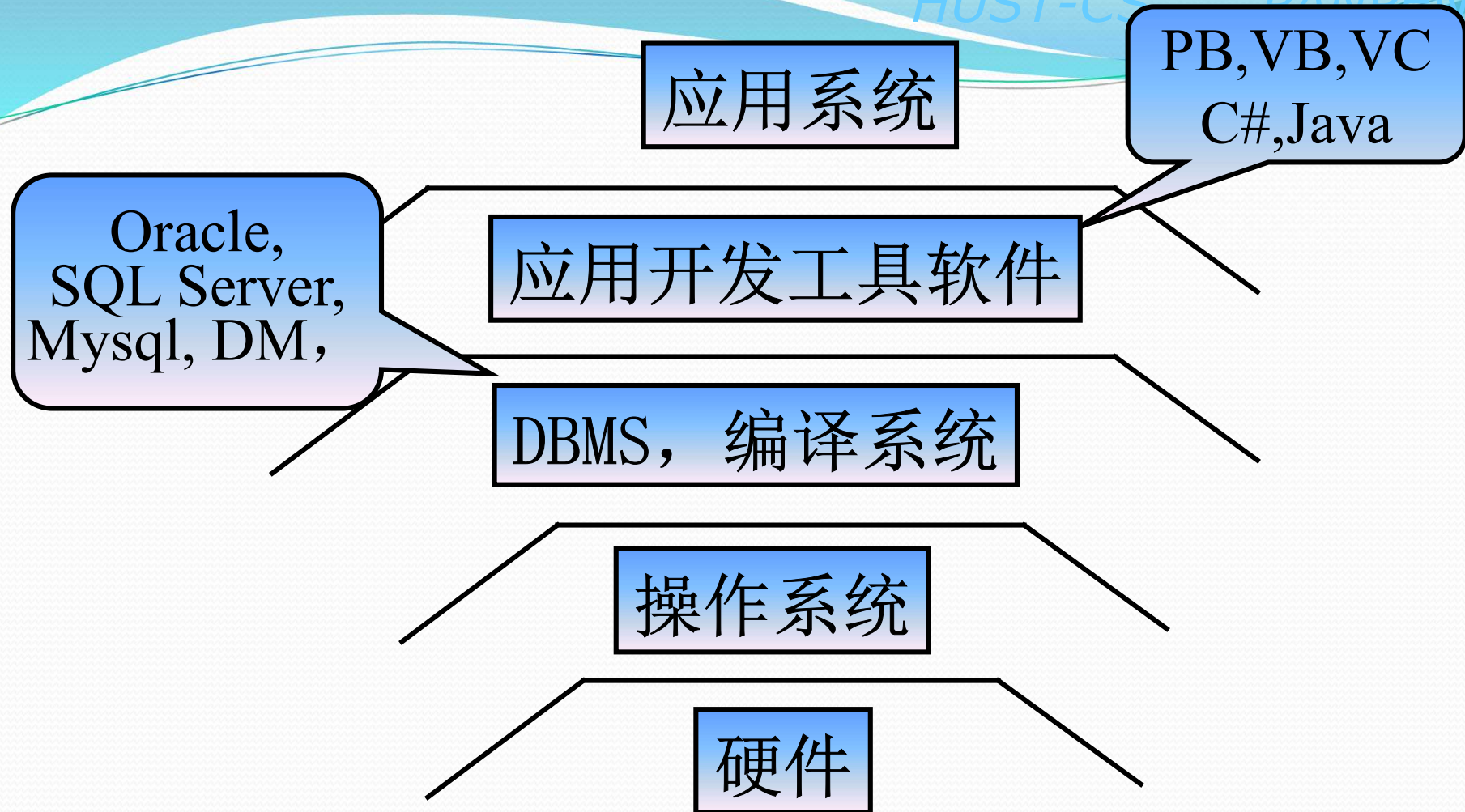
Data Describe

Manipulate

Control

- 数据库系统(DBS)

- 计算机系统引入数据库后的系统
- 包含：数据库管理系统DBMS(及开发工具)、应用系统、数据库管理员(DBA)、用户



数据库在计算机系统中的地位

计算机系统的运行机制

谁读写磁盘？ 操作系统（以物理块为单位）

谁管理缓存中的页面？ 由DBMS管理（更专业）

什么软件的什么“部件”来管理缓存？

DBMS: 专有的缓存管理进程

或者DBMS函数链入应用程序进程

DBMS如何查阅、修改缓存中的数据？

DBMS进程中的数据操作函数进行缓存数据扫描

缓存读写需求如何描述？ SQL语言给出描述

计算机系统的运作方式（续）

SQL语言为何能执行？

SQL语言经由**DBMS**编译后转换成用内部函数描述的
执行计划（代码）。

SQL与应用如何集成？

应用程序代码中嵌入SQL语言，经开发环境编译器
处理后形成应用程序目标模块和对SQL模块的调用。

本章主要问题的提出

- 数据（Data）
数据越来越丰富，需求不断增加，管理数据的方式？
——数据库系统的产生（面向全局，系统、高效）
- 用什么样的方法科学的描述数据？
——数据模型(model)。数据库本身的发展过程。
- 数据库（DataBase, DB）
下有物理设备，上有APP，如何搭建系统？
——模式(schema，包含架构的含义，考虑独立性)
- 数据库管理系统（DBMS）
系统的分布如何？
——体系结构，集中式、C/S、B/S、分布式
- 数据库系统（DBS）
——DBS的组成？ 功能？ 工作过程？ 特征？



1.1 数据库系统的产生

数据库系统(DBS: Database System)的产生经历了人工方法、文件系统方法和数据库系统方法三个历史阶段。

- (1) 人工管理阶段(20世纪50年代中期以前)
- (2) 文件系统阶段(20世纪50年代后期-60年代中期)
- (3) 数据库系统阶段(20世纪60年代后期开始)

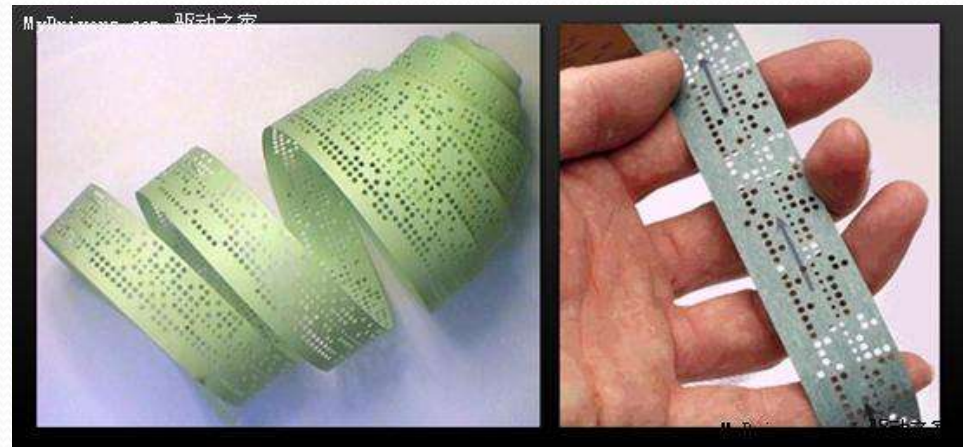
1.1.1 人工方法

1. 特征

- 1) 计算机一般用于科学计算；
- 2) 硬件性能差，无磁盘；
- 3) 数据一般不需长期保存；
- 4) 无数据管理软件(无OS)。



纸带（打孔）、
卡片、磁带
类似于人们绳结
记事



2. 问题

1) 程序编制困难、易出错

说明数据的**逻辑**结构; 整数? 小数? 字符串?

设计数据的**存储**结构; 数组? 链表? 树?

。 。 。 。 。 **存取**方法; 算法?

。 。 。 。 。 **I/O**方式。 顺序? 随机?

2) 数据不共享

数据与应用程序一一对应;

相同数据须各自重复建立。

3) 数据冗余大

4) 应用程序高度依赖于数据的**逻辑结构**与**物理结构**

5) 不能表示**数据间联系**

1.1.2 文件系统方法

1. 特征

- 1) 计算机用于科学计算，还用于**管理**；
- 2) 有磁盘，磁鼓等直接**存取设备**；
- 3) 数据可**长期保存**；
- 4) 有OS (FMS) **管理数据**。



5. 25英寸磁盘（720KB）、3.5英寸磁盘（1.44MB），类似于记事本
机房上机必备——DOS启动盘
特殊工具——CMOS启动盘

有一个动作
叫“**存盘**”
有一个器件
叫作“**优盘**”

设有如下数据:

职工



(职工号、姓名、单位、性别、年龄、工龄、职称、工资)

工资



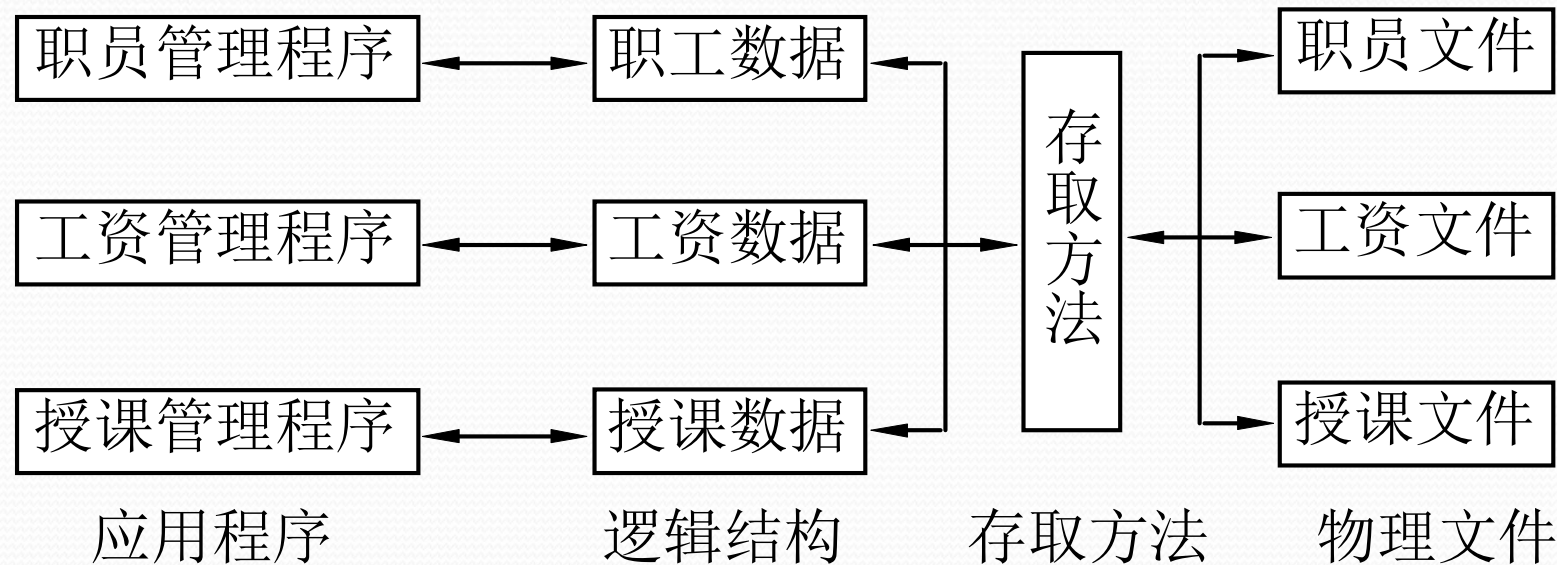
(职工号、姓名、职称、工龄、工资、房租、水电)

教课



(职工号、单位、姓名、职称、课程名、学时)

则文件系统中应用程序与数据的使用对应方式:



文件系统中应用程序及其数据应用方式

前后端分离了么?

应用是局部的，数据是相关联的、全局的。

2. 问题

1) 数据共享性差

基本上一个程序对应一个文件（打开文件有限制）； 部分数据相同时，仍需建立各自文件。 ✦

2) 数据冗余大

3) 潜在数据不一致性(冗余导致)

4) 应用程序与数据结构相互依赖

5) 不能表示数据间联系

逻辑结构
存储结构
存取方法
I/O方式



1.1.3 数据库系统方法

1. 特征

- 1) 应用更广(联机, 分布, 共享)
- 2) 大容量磁盘
- 3) 数据长期保存
- 4) 集中数据管理软件(DBMS)

DBMS: Database Management System

2. 应用程序与数据的应用方式:



数据库系统中应用程序与数据的处理方式

3. 有关概念

1) 局部数据结构：用户局部数据的逻辑结构及其特征的说明。

2) 全局数据结构：用户全部数据的逻辑结构及其特征的说明。

例1：全局数据结构

职工号	姓名	单位	性别	年龄	工龄	工资	职称	奖金	房租	水电	课程名	学时
-----	----	----	----	----	----	----	----	----	----	----	-----	----

例2：

基本信息：

职工号	姓名	单位	性别	年龄	工龄	职称
-----	----	----	----	----	----	----

工资：

职工号	工龄	工资	房租	水电
-----	----	----	----	----

授课：

职工号	课程名	学时
-----	-----	----

3) DB：按一定的方式说明、组织并长期保存的共享数据集合。

数据库对数据的**全局性整合**使数据的**管理全局化**、**数据整体结构化**，亦即将数据管理的底层实现从应用程序中**抽取出来**。

要相对超脱于应用程序的具体范畴，实现全局数据的**整体结构化**，内涵的是**什么基本问题**？

统一的思考和描述风格，提炼出相应的规则和方法。



建立描述、解决问题的**模型**。

1.2 数据模型(Data Model)

1.2.1 概述

1. 功能

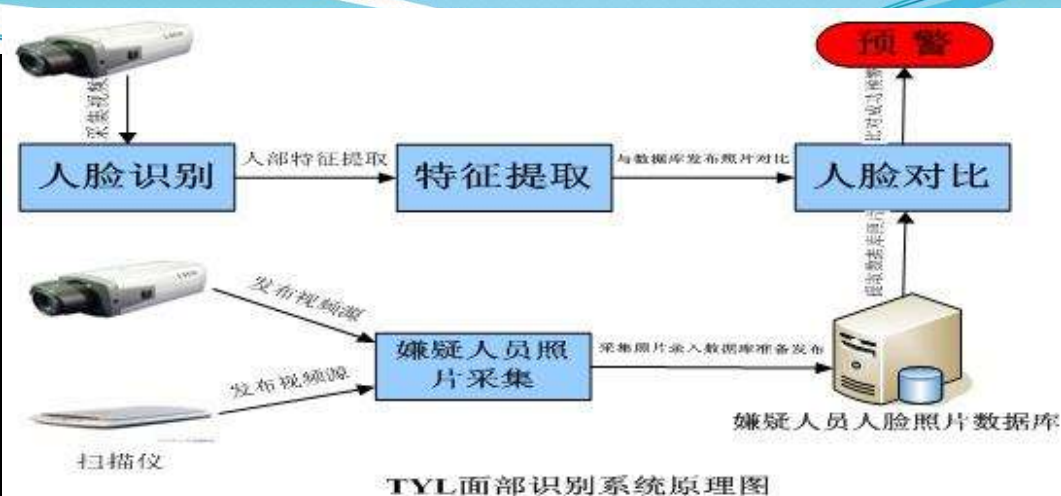
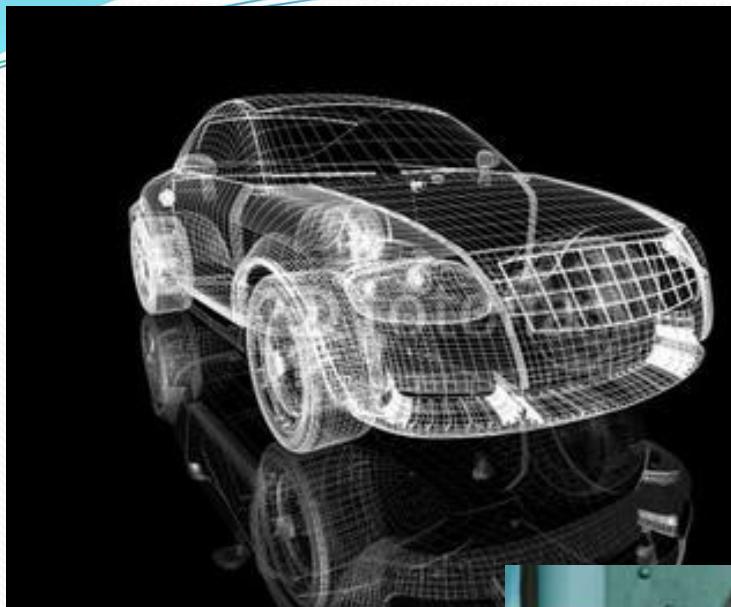
现实世界的表示方法。

认识→表示→处理



模型是对现实世界特征的模拟和抽象

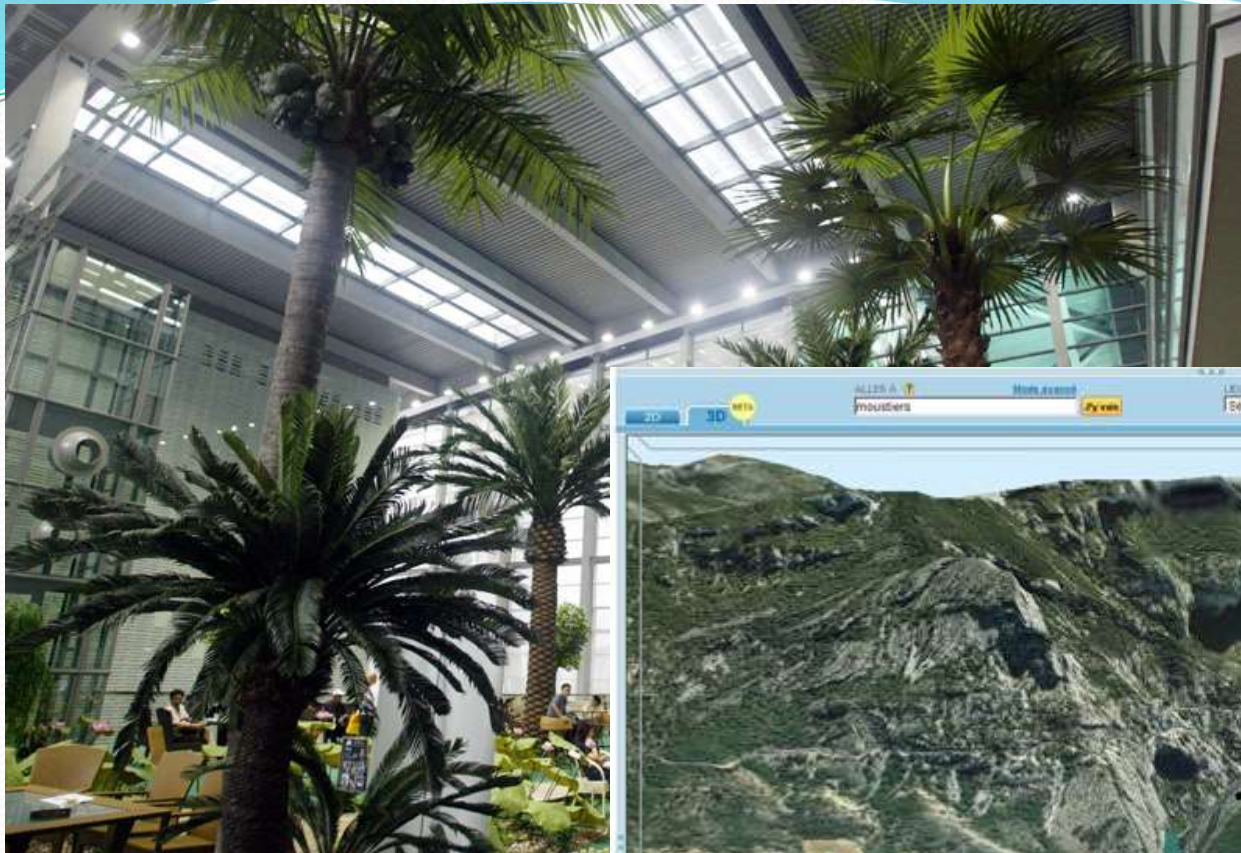




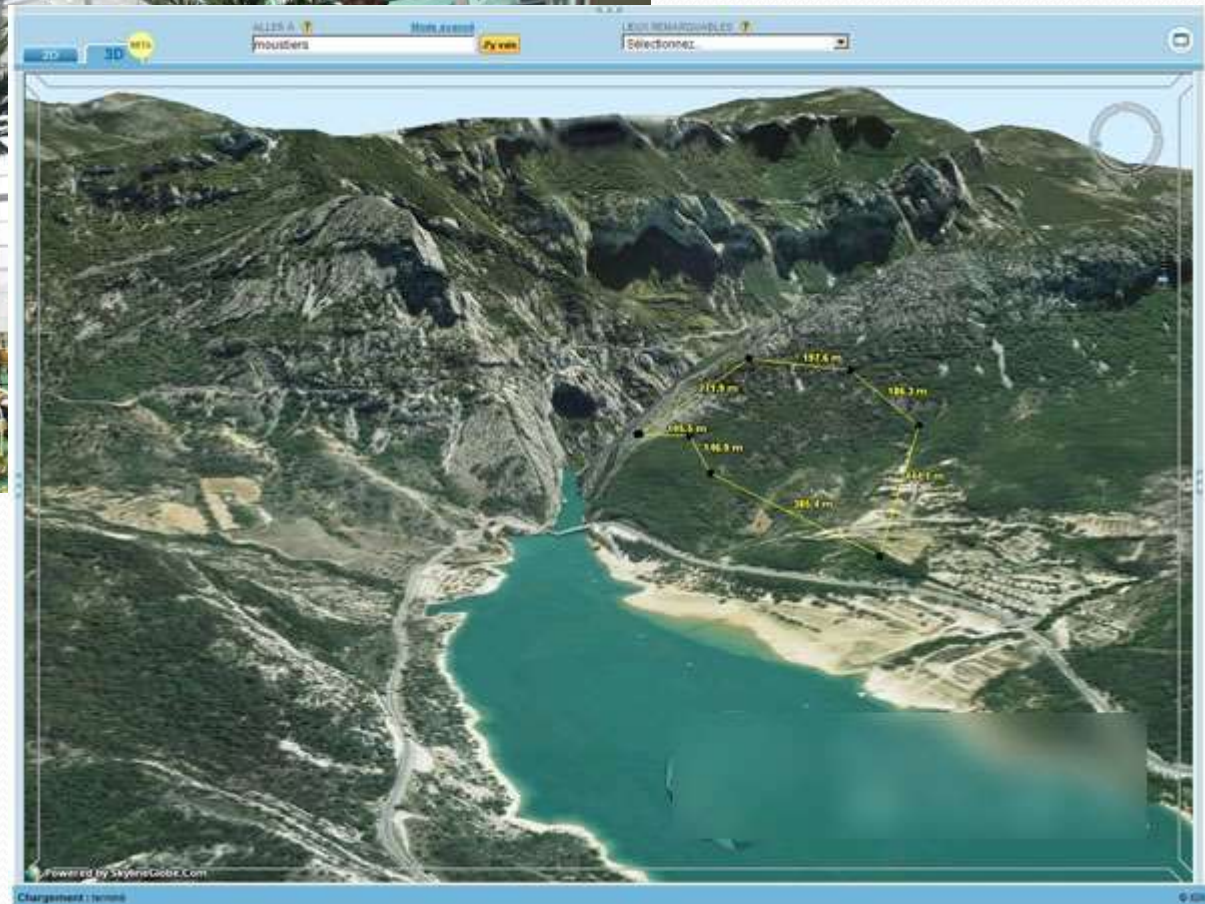
数据特征



东湖绿道的故事，张学友“逃犯克星”的故事



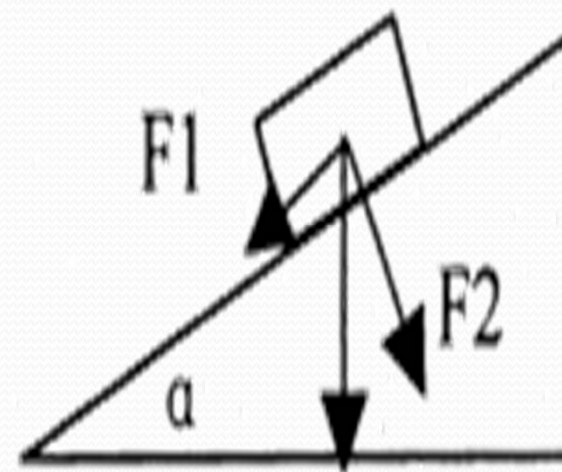
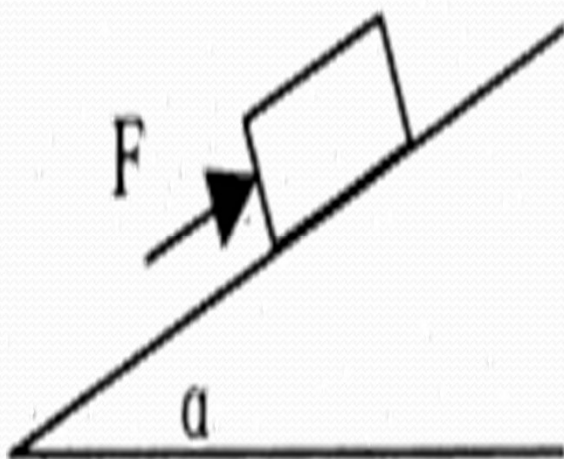
抓主要
特征





静态特征和动态特征

特征往往伴随相应的分析方法



模型刻画问题、抽取特征、 衍生方法

汽车模型、航模、
五官、
环境模拟、
虚拟现实

。 。 。

物理学模型、数学模型（例如SVM）、数据模型

1.2 数据模型 (Data Model)

1.2.1 概述

2. 概念

数据及数据间联系的表示形式(现实世界的模拟)

数据模型分为概念模型、逻辑模型和物理模型

3. 要求

- 1) 较真实地表示现实世界;
- 2) 易于理解;
- 3) 易于实现。

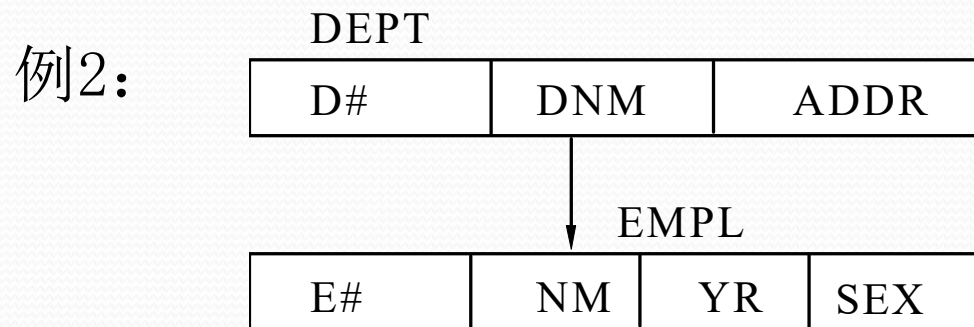
数据在系统内部的表示和存取方法

4. 数据模型的构成要素

1) 数据结构

说明系统静态特征的，描述数据及其联系的构成方式。

例1: Student (XH, XM, YL, XB)



2) 数据操作

说明系统动态特征的，对数据进行的操作集合(含操作规则)，
如: insert, delete, update, select。

3) 数据约束

说明给定数据模型中数据及其联系的组织规则。

例如: 工龄 < 年龄, 出生年份 > 1962

5. 典型的逻辑模型



- ① 层次模型(Hierarchical Model)
- ② 网状模型(Network Model)
- ③ 关系模型(Relational Model)

1.2.2 概念模型

1. 定义

独立于特定DBMS的现实世界的抽象模型(用户与数据库设计人员进行交流的语言)。

现实世界



认识、抽象

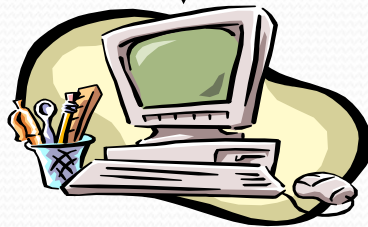


信息世界



概念模型

机器世界



DBMS支持的数据模型

2. 概念模型的特点

- 1) 较强语义表达能力;
- 2) 便于直接表示应用语义;
- 3) 简单、清晰, 易于理解。

3. 信息世界概念

1) 实体(entity)

——客观存在可相互区别的事物和概念（具体的人、物，抽象的概念）。

例如：职工、学生、部门、课程、电影、音乐

2) 属性(Attribute)

——实体具有的特性。

Student (XH, XM, XB, NL)

3) 实体型(entity type)

——具有相同特征和性质的实体与属性命名序列。

student

XH	XM	XB	NL
----	----	----	----

course

KH	KM
----	----

S-C

XH	KH	CJ
----	----	----

实体?

4) 实体值 (entity value)

——实体型的具体实例。

student

XH	XM	XB	NL
0011	张自成	男	21
0012	刘自淑	女	18
0013	李自威	男	20
0014	马自得	男	20
...

5) **实体集** (entity set)

——同型实体值的**集合**。

6) **域** (domain)

——属性的取值范围。

例：NL为小于150的三位整数，XB为（男，女）

7) **码** (KEY)

——**唯一标识**一个实体集中任何实体值又**不含多余属性**的属性集。



Student (KEY): XH

SC (KEY) : (XH, KH)

实体?

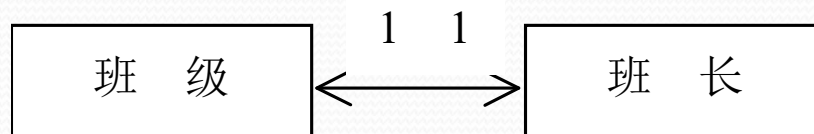
- 至少一个属性
- 至多n个属性
- 不含多余属性

(XH, XM) ——KEY?

8) 联系 (relationship) (实体集之间的联系)

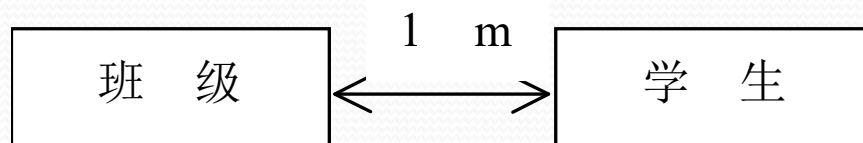
① 一对一联系 (1: 1)

定义：设有实体集A、B，若其中任何一个实体集中每一实体至多与另一实体集中的一个实体有联系，则称A、B间存在一对一联系。



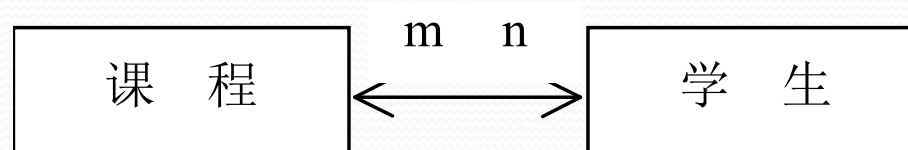
② 一对多联系 (1: n、1: m)

定义：设有实体集A、B，若A中的每一个实体，与B中的n个实体 ($n \geq 0$) 有联系，反之，对于B中的每一个实体，至多与A中的一个实体有联系，则称A、B间存在一对多联系。



③ 多对多联系 (m:n)

定义：设有实体集A、B，若其中任何一个实体集中的每一个实体均与另一个实体集中的n个实体 ($n \geq 0$) 有联系，则称A、B间存在多对多联系。

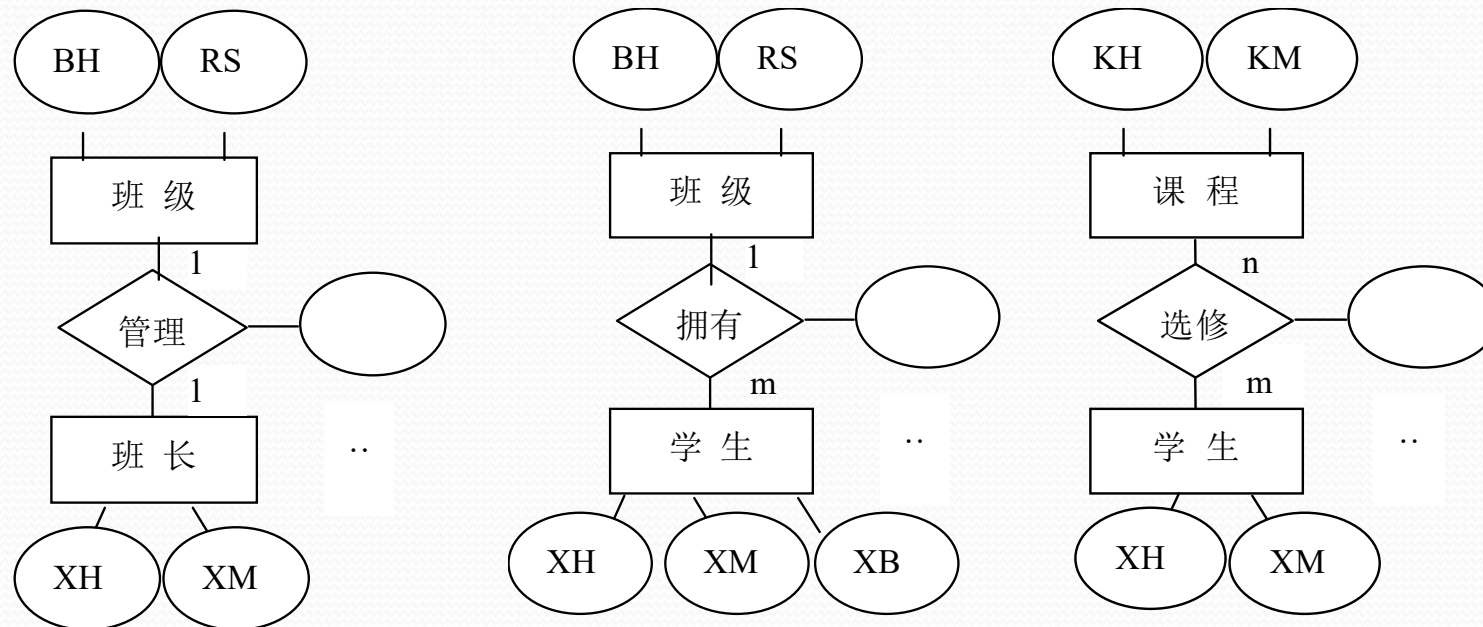


思考问题：如何描述、区分多个相同类型的联系？

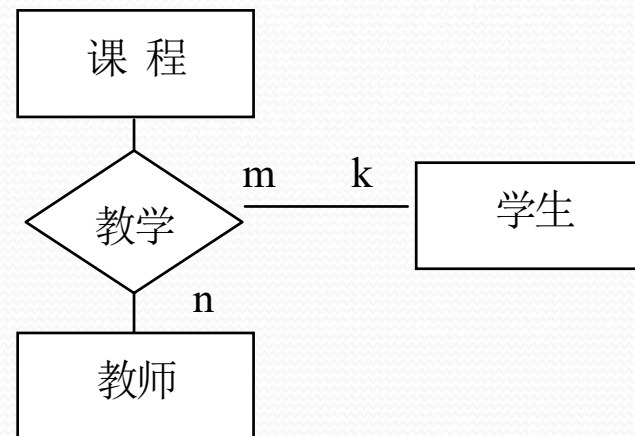
4. 概念模型表法方法

E-R方法 (Entity Relationship Approach)

1) 构成形式



- (1) 矩形表示**实体型**，框内标明实体名；
- (2) 椭圆表示**属性**，用无向边与其相应实体连接；
- (3) 菱形表示**联系**，内标明联系名，用无向边与相关实体连接；
- (4) 无向边上标明**联系的类型**（1: 1, 1: m, m: n）；
- (5) 可据需要任意展开。



2) 特征

(1) 直接表示 $m:n$ 联系

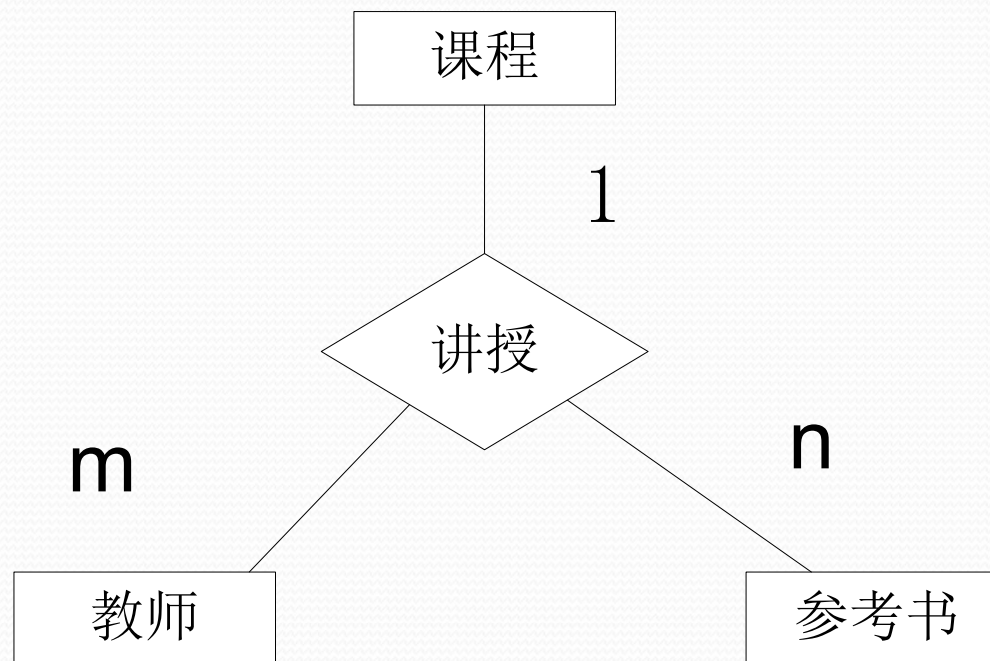
(2) 与特定DBMS无关

可应用性广、抽象、接近现实

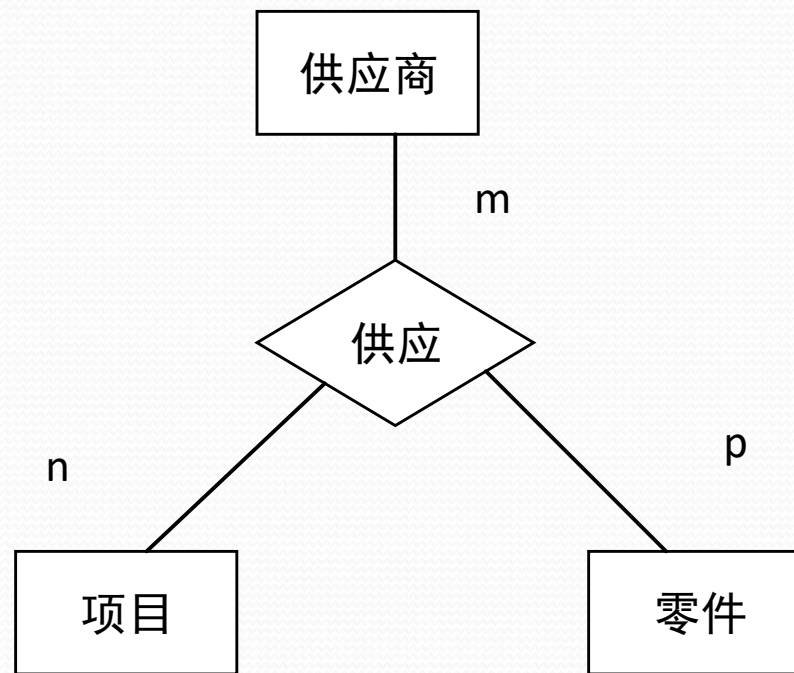
(3) 易于向特定DBMS支持的数据模型转换

实体联系举例

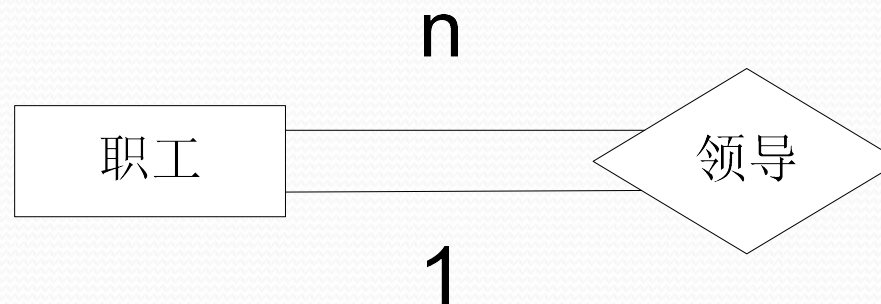
- 课程、教师、参考书 (P217)



- 供应商、项目、零件

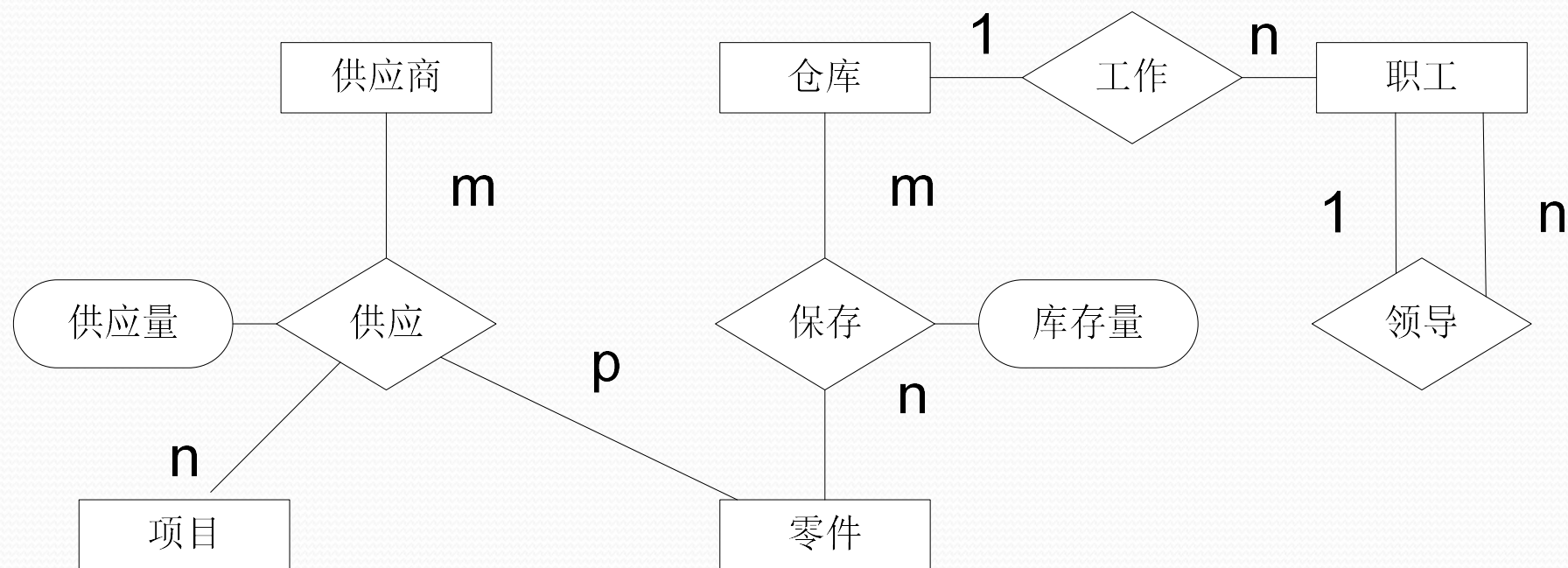


- 职工、领导



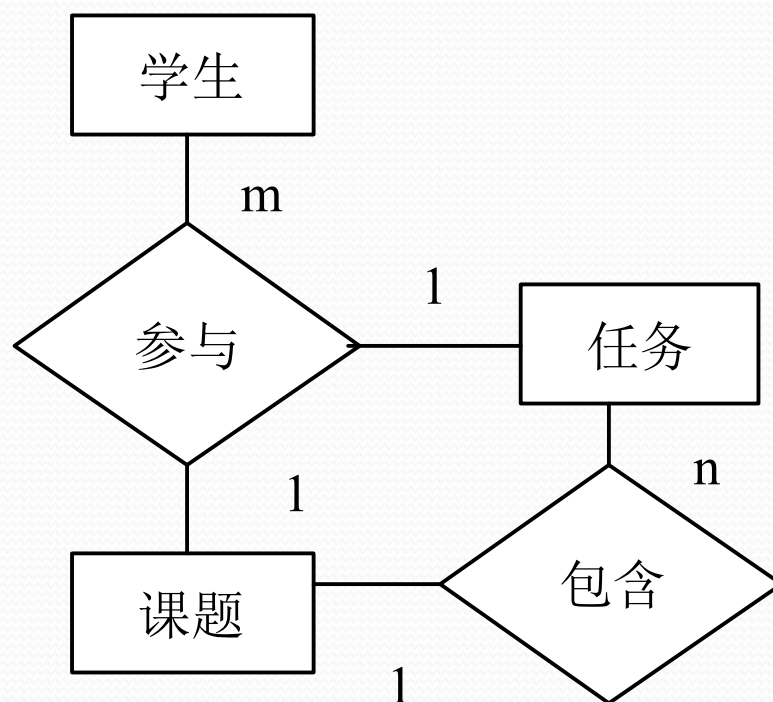
- 供应商、项目、零件、仓库、职工、领导
(P219)

体会：ER图的表示能力、可读性。

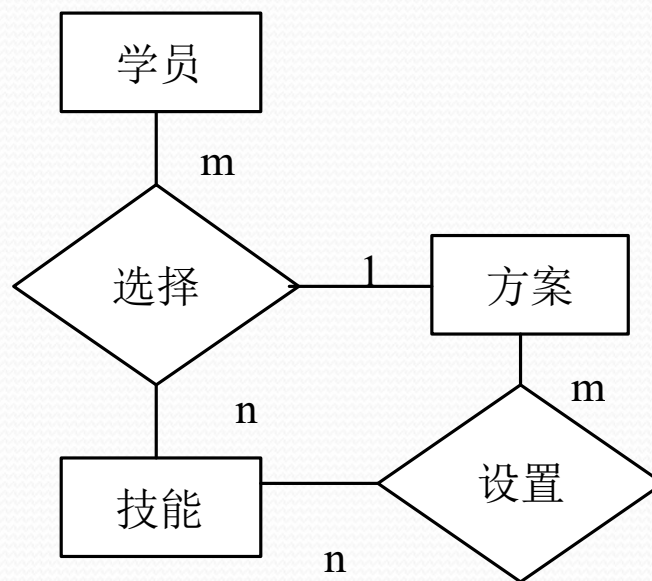


课堂练习：第二课堂活动：学生、课题、任务

课题可以有多项任务，每项任务属于一个课题，一个学生可以参加一个课题，一个课题可有多名学生参加，学生参加课题后分得一项任务，画出**E-R**图。



练习：一项技能有多种学习方案，不同技能可能有相同的学习方案，一个学员可学习多项技能，但学生学习每项技能只能选择一种学习方案，画出**E-R**图。



实体(entity)

——客观存在可相互区别的事物和概念（静态、动态、物质、精神等）

1.2.3 层次模型

——用**树形结构**表示实体及实体间联系的数据模型。

1. 数据结构

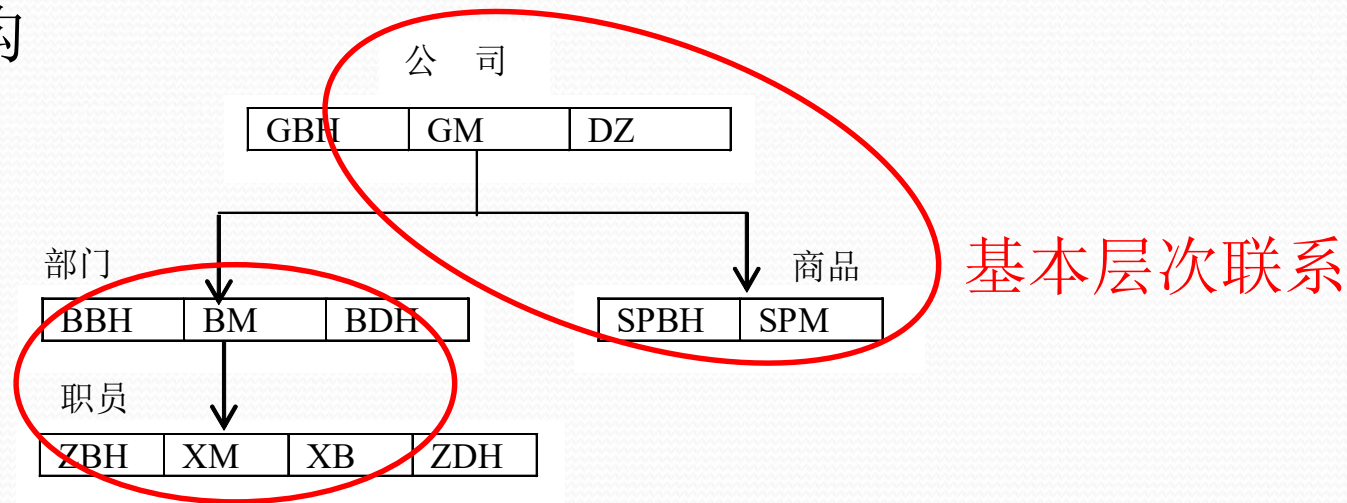


图1-4 层次模型简例

- 1) 一个结点表示一个**实体**（一个**片段**：fragment）；
- 2) 有向连线表示实体间**联系**；
- 3) 结点内含**字段**，表示**属性**；
- 4) 片段、字段须命名；
- 5) 特征
 - ①有且仅有一个结点无双亲结点，称之为根结点（root）；
 - ②余下子女结点有仅有一个双亲结点。

- 层次模型的存储结构

邻接法——前序遍历树的方式，通过物理地址的相邻体现层次顺序。

链接法——（子女—兄弟链接法）用指针来反映数据之间的层次关系，每个记录包含两类指针，分别指向最左边的子女和最近的兄弟。

2. 操作

增加、删除、修改、查询（I、D、U、Q）

3. 约束

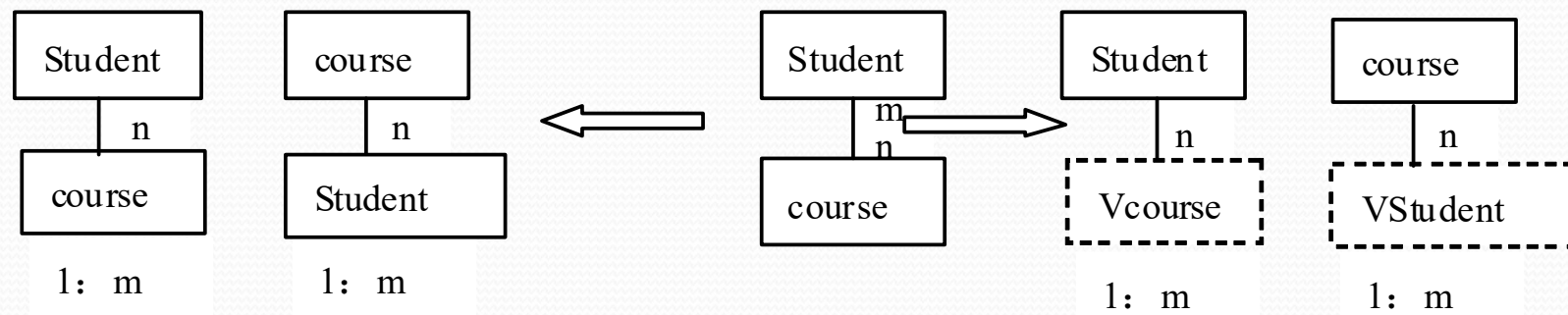
- 1) 无双亲不能插入子女结点值；
- 2) 删去双亲结点值，则子女结点值同时删去。

4. 优点

- 1) 简单易用（几条操作命令）；
- 2) 自然表示1: n联系；
- 3) 速度较快。

5. 缺点

- 1) 不能直接表示m: n联系；
（须引进冗余或者虚拟结点）



- 2) 插入，删除操作限制多；
- 3) 查子女须经过双亲。
(从上到下，从左到右)

1.2.4 网状模型

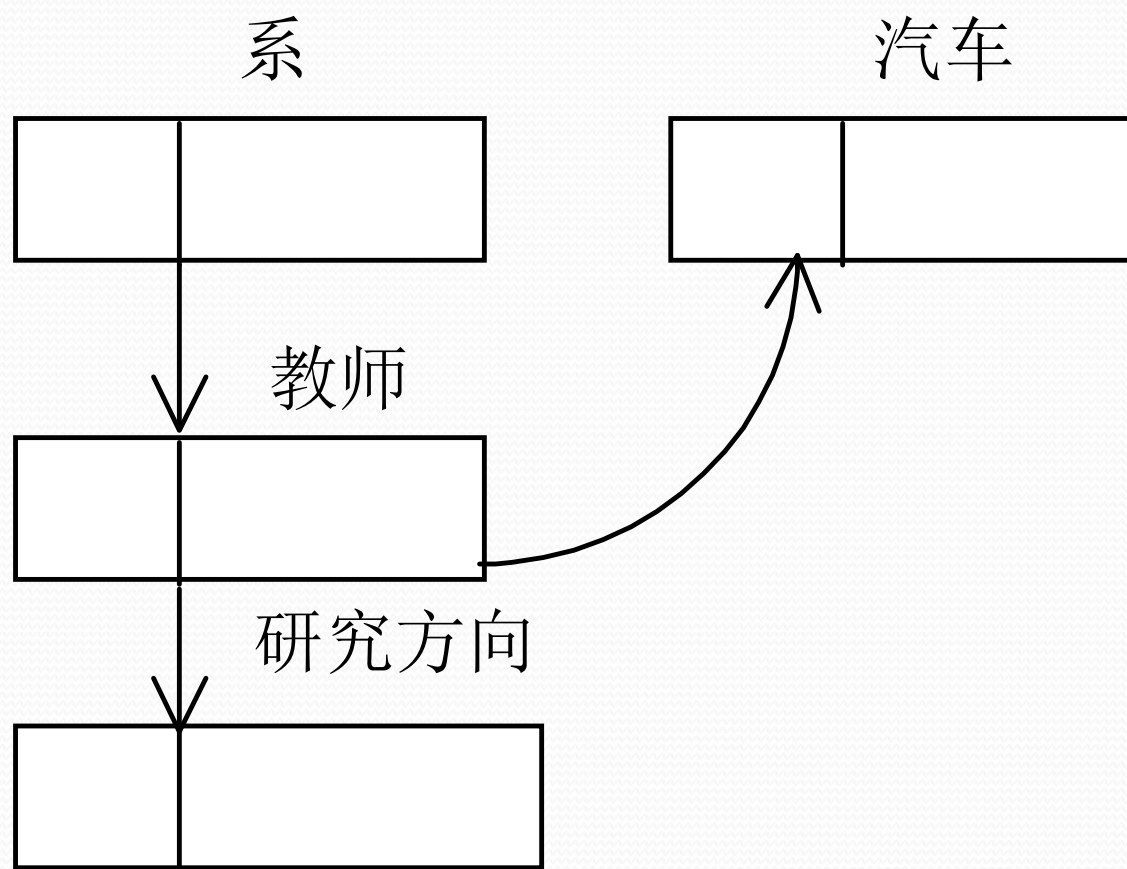
——用**网状结构**表示实体及实体间联系的数据模型
DBTG (Database Task Group) 提出的一个系统方案。

去掉了层次模型的两个限制（根结点、多个双亲结点），引入了复合联系。

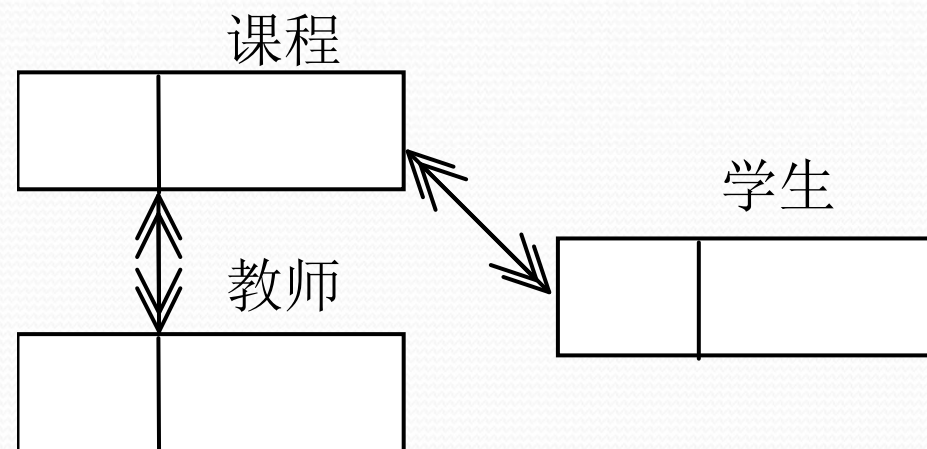
层次模型可看作网状模型的一个特例。

1. 数据结构

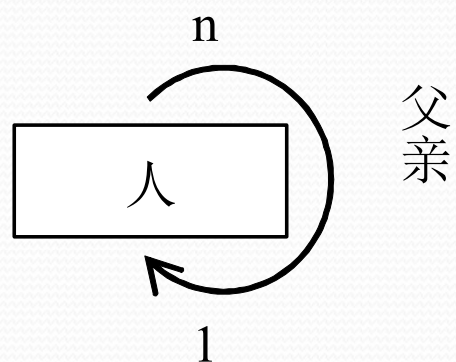
a: 简单网



b: 复杂网

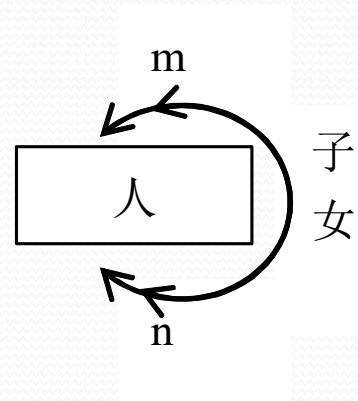


c: 简单环网



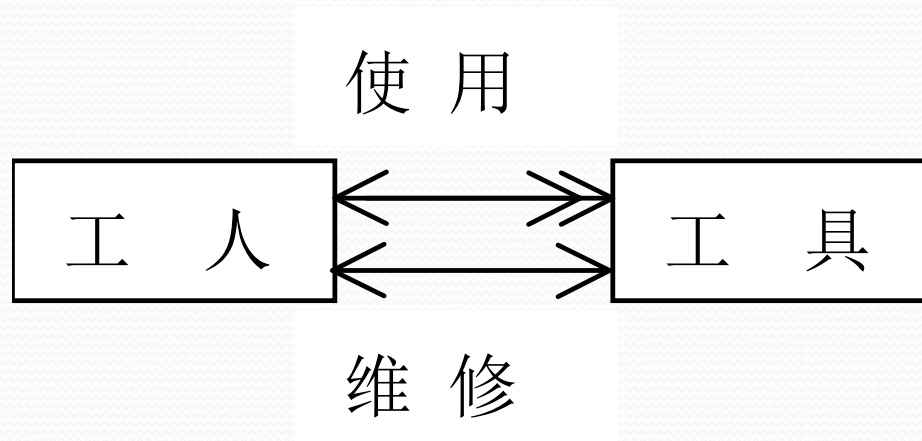
- 一个父亲有多个儿子;
- 儿子只一个父亲。

d: 复杂环网



- 每个子女可多个子女;
- 每个为人子女者又是两个人的子女

e: 复合联系



- 1) 结点表示实体，称为记录类型；
- 2) 结点内含数据项，表示属性；
- 3) 有向连线表示实体间联系；
- 4) 属性可嵌套。

XH	XM	CJ		
		CJ1	CJ2	CJ3
901	刘亦易	59	73	66
902	刘亦难	81	82	83

5) 特征

- (1) 可多个结点无双亲结点；
- (2) 子女结点可多个双亲结点；
- (3) 两记录间可多种联系。

2. 操作

I、D、U、Q

3. 约束

- 1) 插入不受限制;
- 2) 删去双亲, 子女不受影响。

4. 优点

- 1) 直接表示的m: n联系;
- 2) 存取效率高。

5. 缺点

- 1) 结构复杂;
- 2) 数据描述语言复杂;
- 3) 一次存取一个记录值;
- 4) 应用程序与数据结构相互依赖;
- 5) 过程化语言。



1.2.5 关系模型(Relational Model)

——用二维表格表示实体及其间联系的数据模型。

1. 数据结构

表 {

XH	XM	XB	NL
2001	庄帅	男	20
2002	庄酷	男	18
2005	庄靓	女	18
2003	庄洒	男	18
.....			

↑
列

↓
域
(domain):
属性取值
范围

←属性名

←元组 (4 元元组)

属性和域在概念上有区别。

一般，域关联于一定的数据结构和值的集合，而属性则与关系的语义相关联。

属性从属于一定的域，反映该域在描述关系时的应用。

- 1) 表格表示实体，内含属性；
- 2) 表格也可用于表示实体间联系；

STUDENT(XH,XM) **COURSE**(KH,KM) **SC**(XH,KH,CJ)

- 3) 行、列次序无关。
- 4) 每一个分量均不可再分。

注意：此处有嵌套属性，
不是关系模型

关系模型的
错误示例

XH	KH	CJ		
		CJ1	CJ2	CJ3
...

- 5) 至少一个码（主码）。

存储结构：类似于数组的形式在页面内存储。

- 关系模型的术语:

关系 (Relation)

元组 (Tuple)

属性 (Attribute)

码 (Key)

域 (Domain)

关系模式——关系名加上属性名列表

2. 操作

I、D、U、Q

3. 约束

——完整性约束 (integrity)

- 1) 实体完整性;
- 2) 参照完整性;
- 3) 用户定义完整性。

4. 优点

- 1) 建立在严格的数学概念的基础之上

以集合论、关系代数为基础，在数据建模和操作方面具有严格的数学基础。关系数据库理论为一自成体系的形式化理论。

- 2) 结构简单易用

相对于层次模型和网状模型，关系模型将系统划分为较小的单元，具有较少的结构约束和较大的灵活性，对数据共享的支持也较强。

3) 应用程序与数据说明独立:

➤ 存取路径透明;

➤ 非过程化语言。

4) 集合操作。

5. 缺点

1) 查询效率慢。

2) 复杂数据类型表示能力弱。



数据模型在数据库系统中的重要性

- 核心、基础
- 命名的依据
- 对应着DBMS的发展
- 目前关系模型占主导地位
- 持续发展（时态模型、时空模型、OO、OR）



里程碑

数据库领域的标志性人物1

- C. W. Bachman

网络DB之父，1964年研制网状数据库IDS (Integrated Data System)，推动了DBTG报告的制定，形成了网状数据库的标准，1973年获得图灵奖（ACM的最高奖）。




描述能力

数据库领域的标志性人物2

- E. F. Codd

关系数据库之父，1970年发表了一系列关系DB的论文，以此撰写了博士论文，是关系数据库的数学理论的奠基人，1981年获得图灵奖。



生产效率

数据库领域的标志性人物3

- James Gray

关系数据库的基本理论逐渐成熟的情况下，针对如何保障数据的完整性、安全性、并发性以及故障恢复的能力等技术问题给出了系统的解决方案，开拓了事务处理的研究领域，1998年获得图灵奖。

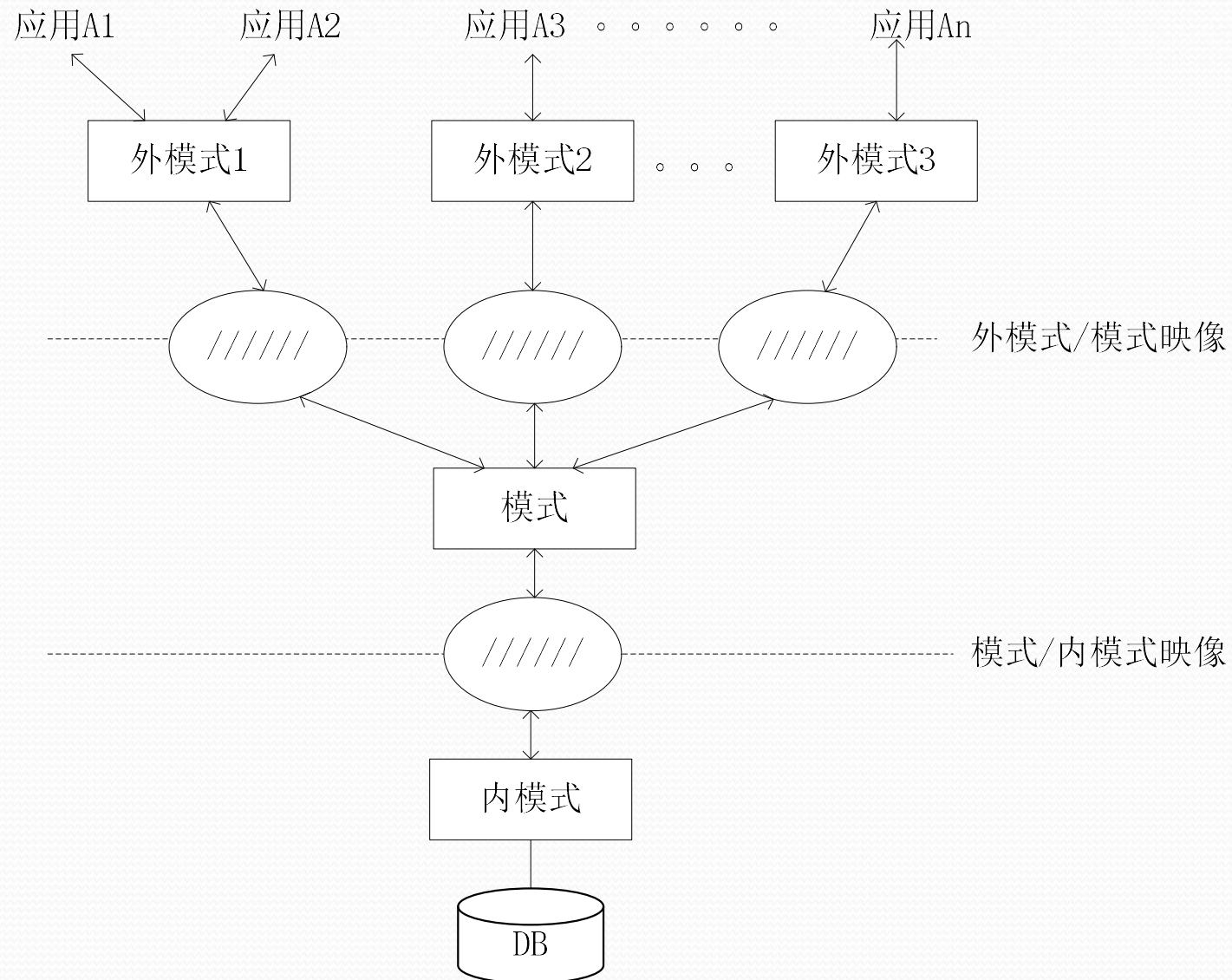


Transaction



1.3 DBS结构

1.3.1 三层模式结构



1.3.1.1 有关概念

1. 模式 (schema)

——数据库中全体数据的逻辑结构及其特征的说明。

- 全体性
- 逻辑性: student(XH, XM, NL)
- 特征描述: 名称、数据、类型、长度、约束
- 说明性: 上述结构及特征的表示程序。

用特定语言写的表达上述结构及特征的程序。

2. 外模式 (External schema) (subschema)

——数据库中局部 (局部用户) 数据的逻辑结构及其特征的说明。

1) 外模式是模式的“子集”

Student (XH, XM, XB, YL)

Course (KH, KM)

SC(XH, KH, CJ)

①单关系子集

student (XH,XM)

②多关系子集

SCE(XH,KH,XM,CJ)

2)外模式间可相互重叠

3)可不同于模式

命名、数据类型、安全约束、结构

4) 虚结构：数据仍按原关系模式存储。

5) 一个模式可多个外模式

6) 一个应用程序只能使用一个外模式，多个应用程序可共用一个外模式。

3.内模式 (Internal Schema) (storage-schema)

——数据库物理结构、存取路径及存取方法的说明
一个数据库中一个内模式。

1.3.1.2 映像

1.外模式/模式映像

——说明外模式与模式间的对应联系（外模式中说明）。

2.模式/内模式映像

——说明模式与内模式的对应关系（模式中说明）。
（逻辑结构在内部如何组织）

1.3.1.3 模式的作用

1. 子模式作用

1) 支持不同用户建立适应局部应用特征的结构;

2) 简化应用处理;

3) 提高安全性;

4) 实现数据逻辑独立性:

➤ 分隔应用程序与模式

➤ 模式变, 由DBA改变外模式/模式映像, 外模式不变, 应用程序不变。

2. 模式作用

1) 支持数据少冗余共享;

student (XH,XM)

2) 支持数据逻辑独立性;

3) 支持数据物理独立性。

➤ 分隔子模式与内模式

➤ 模式变，由**DBA**改变模式/内模式映射，模式不变，子模式不变，应用程序不变。

3. 内模式作用

1) 支持用户建立适应需求的物理结构等;

2) 实现数据**物理独立性**。

➤ 程序中屏蔽物理细节。

➤ 内模式变，**DBA**改变映像，模式不变，外模式不变，应用程序不变。

关于三层模式两级映像

- 结合后续章节的内容进一步体会
- 结合ORACLE、Sql Server等实际系统进一步体会

关于模型和模式的思考

目前的DBMS以关系模型为主流，使用表数据结构，导致了大多数DBS中均涉及到了“表”的概念：

数据库——表——table

ADO对象——RecordSet——table

应用程序界面或者网页——报表

角度不同，“表”的意义不同

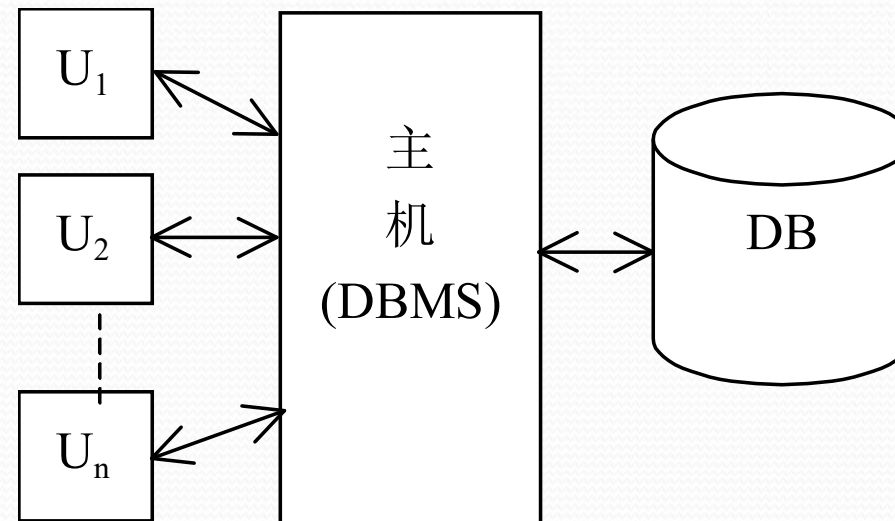
应用需求：面向应用程序的语义背景

程序设计：面向业务逻辑处理、代码的封装和优化

数据库设计：面向系统的描述能力和性能，需要方法和理论

关系模型是表的集合，而如何将表的集合转换为向特定用户展示的数据的子集（依然是表的形式）则是关系数据库三层模式首先考虑的问题

1.3.2 主从式结构



1、优点

- 1) 结构简单;
- 2) 资源共享性高（外理及数据均由主机完成）;
- 3) 数据易于管理与维护。

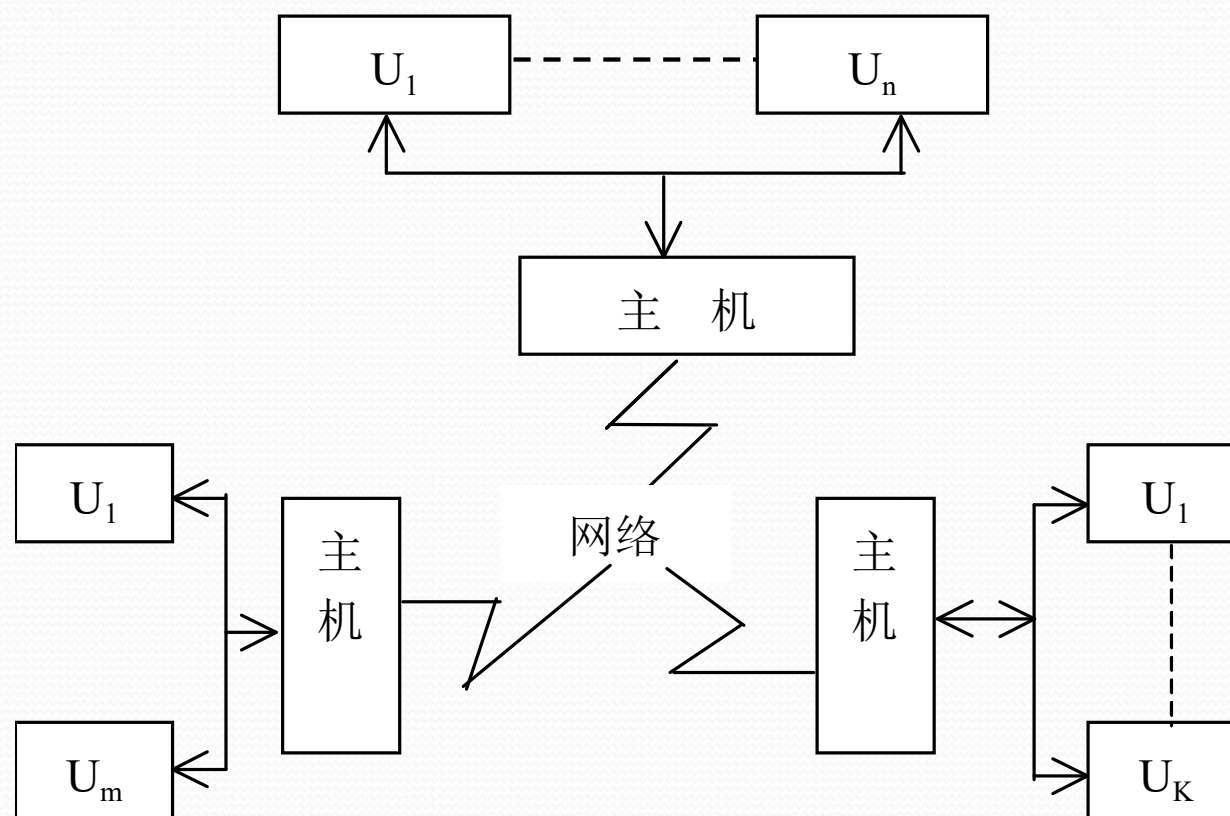
2、缺点

1) 主机负担重

用户数增多→I/O瓶颈

2) 可靠性弱（主机故障）

1.3.3 分布式结构 (distribution)



1. 优点

1) 自治与协调

a. 独立能力;

b. 异地数据访问。

2) 可靠性高

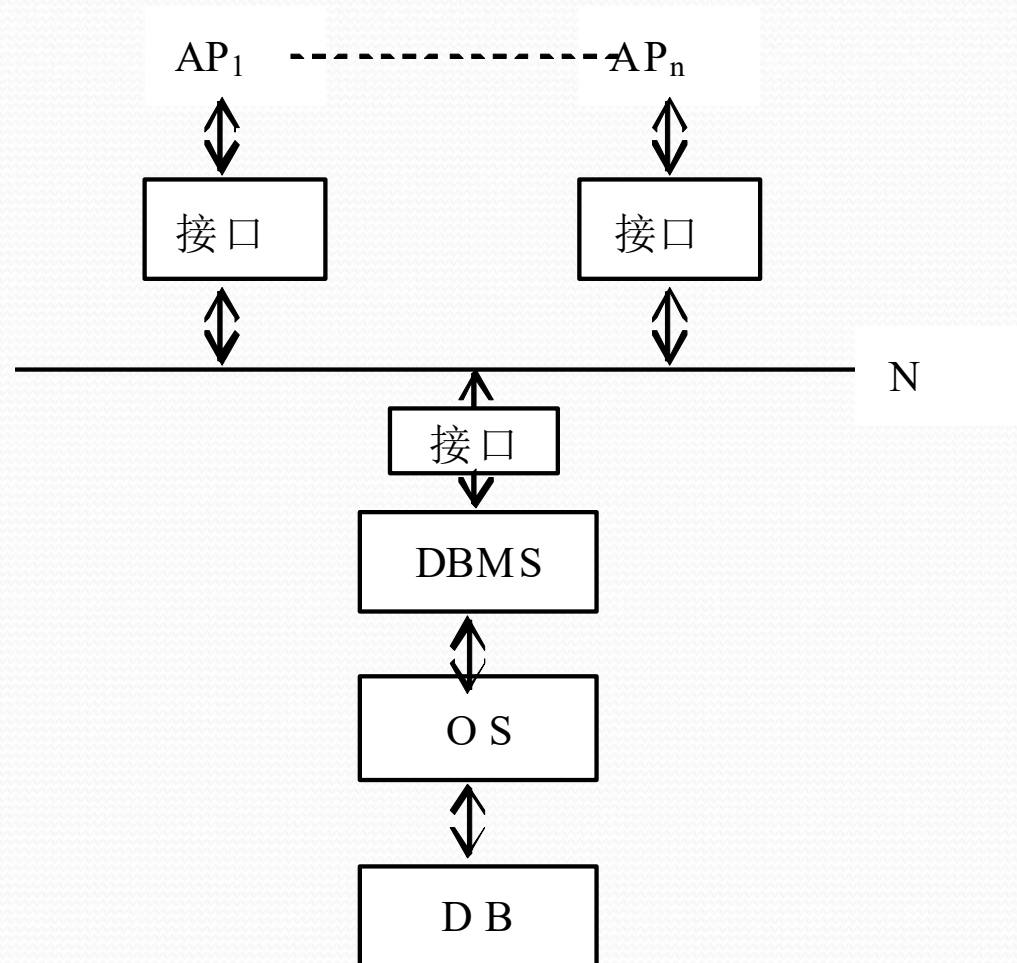
3) 可用性好

2. 缺点

1) 结构与管理复杂;

2) 效率受网速影响。

1.3.4 客户/服务器结构 (Client/Server, C/S)



客户发请求到S端，结果返回到C端。

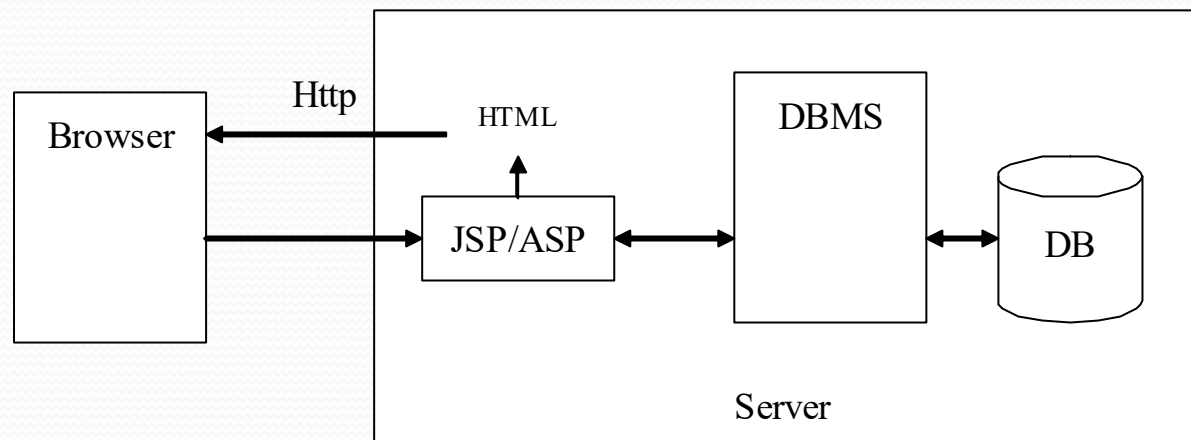
1. 优点

- 1) 负载均衡，效率提高；
- 2) 减少网络传输量；
- 3) 提高吞吐率；
- 4) 开放性好。

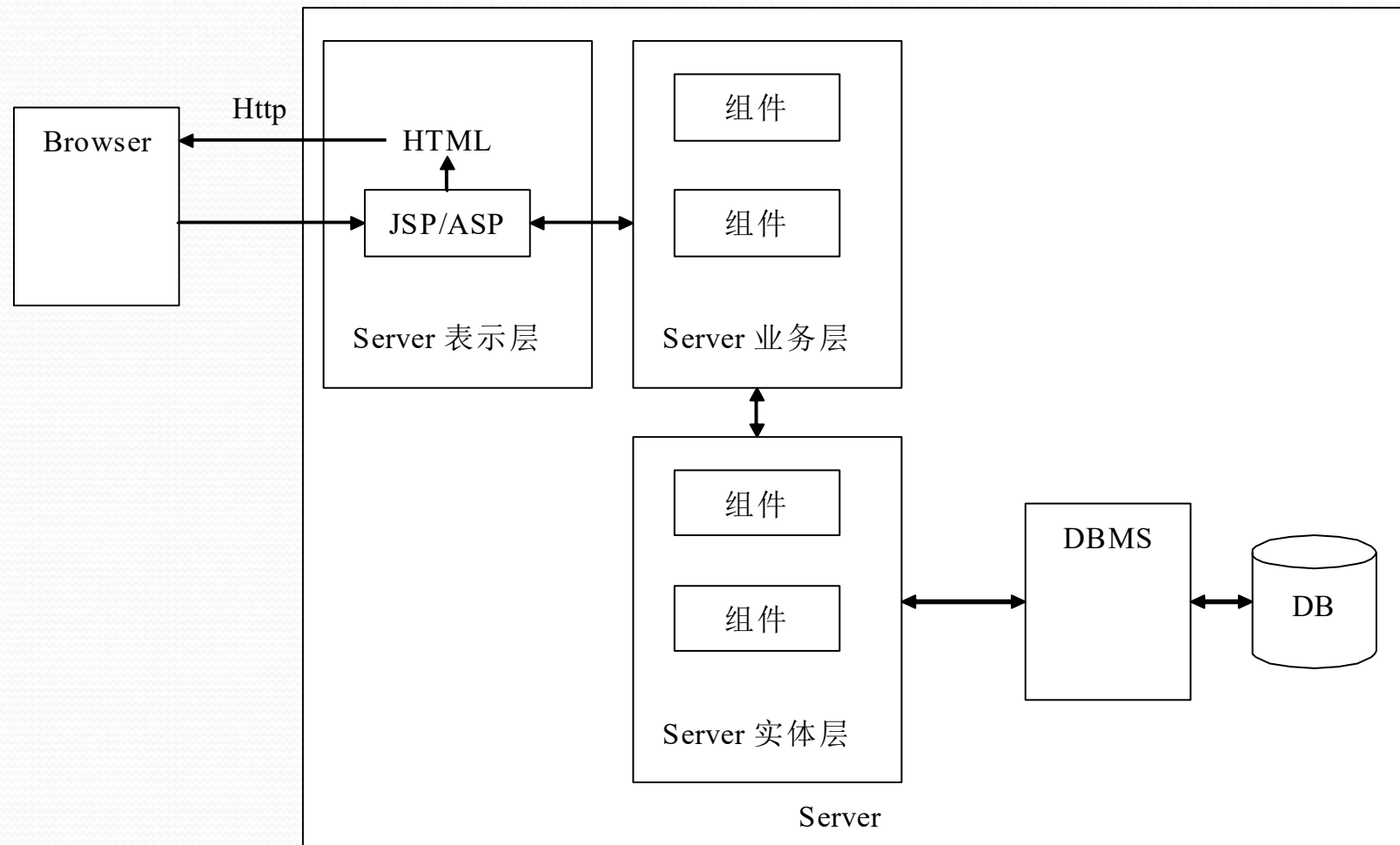
2. 缺点

数据库访问瓶颈。

数据库系统的B/S结构



数据库系统的B/S结构（续）

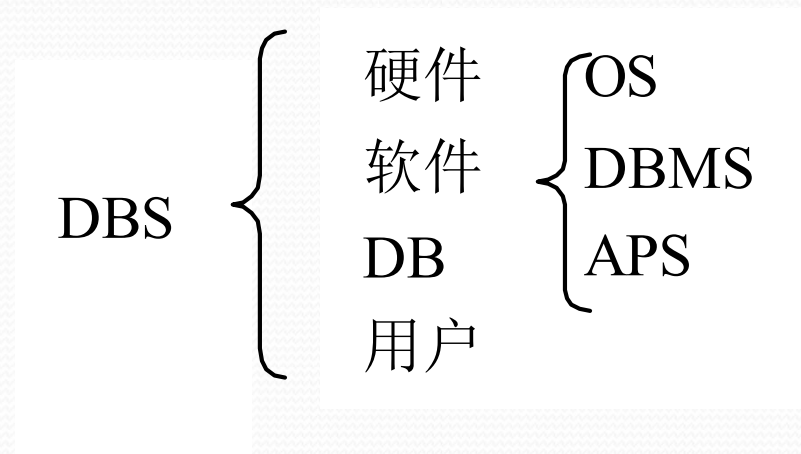


DBMS? Sql Server?

.NET平台 组件化 一种尝试

1.4 DBS组成

1.4.1 概述



1、定义：支持定义、使用和维护**DB**的系统软件。

2、功能

1) **DB**定义

模式、子模式、内模式、映射、约束

2) **DB**操纵 (**Manipulation**)

增、删、改、查

2、功能(续)

3) DB存储

存储结构、存取路径、I/O方法

4) DB运行管理

安全性、完整性检查, 数据字典(DD)、索引维护、并发控制

5) DB建立

初始数据输入, 数据转换。

6) DB维护

转储与恢复、重组、重构、性能监视与分析。

7) DB通信

OS、Netware、其它DBMS。

需求分析中体现为
有关说明文档,
关系数据库中体现为
系统表

3、组成

1) **DB**定义、操纵语言及编译程序（含预处理及解释）

2) **DB**运行控制程序

初启程序、**I/O**，存取路径管理、缓冲区管理、安全控制、完整性控制、并发控制、事务管理、日志管理。

3) 实用程序（**utility**）

初装、转储、恢复、监测、转换、重组、重构、通讯。

1.4.2 应用程序

主语言+DML

1.4.3 用户

1、DBA (Database Administrator)

DBMS、数据库其它软件管理与维护

（安全授权、监测和改进性能）

2、系统分析员

分析用户需求，确定数据库事务

3、应用程序员

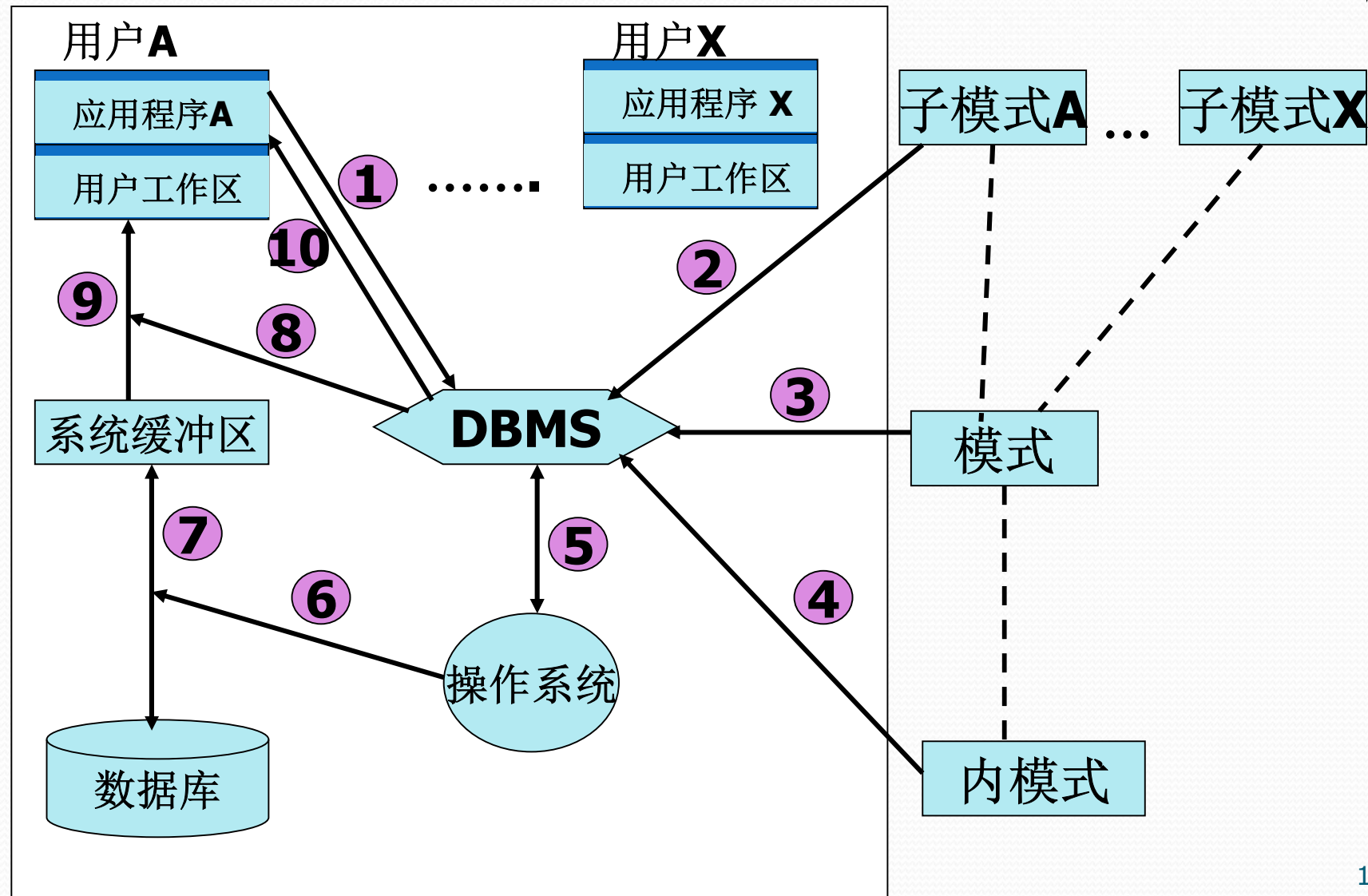
应用软件编码、调试和维护

4、终端用户

使用数据库

1.5 DBS工作过程

1.5.1 从数据库中读取记录的过程



在数据库系统中，当一个应用程序或用户需要存取数据库中的数据时，应用程序、**DBMS**、操作系统、硬件等几个方面必须协同工作，共同完成用户的请求。这是一个较为复杂的过程，其中**DBMS**起着关键的中介作用。

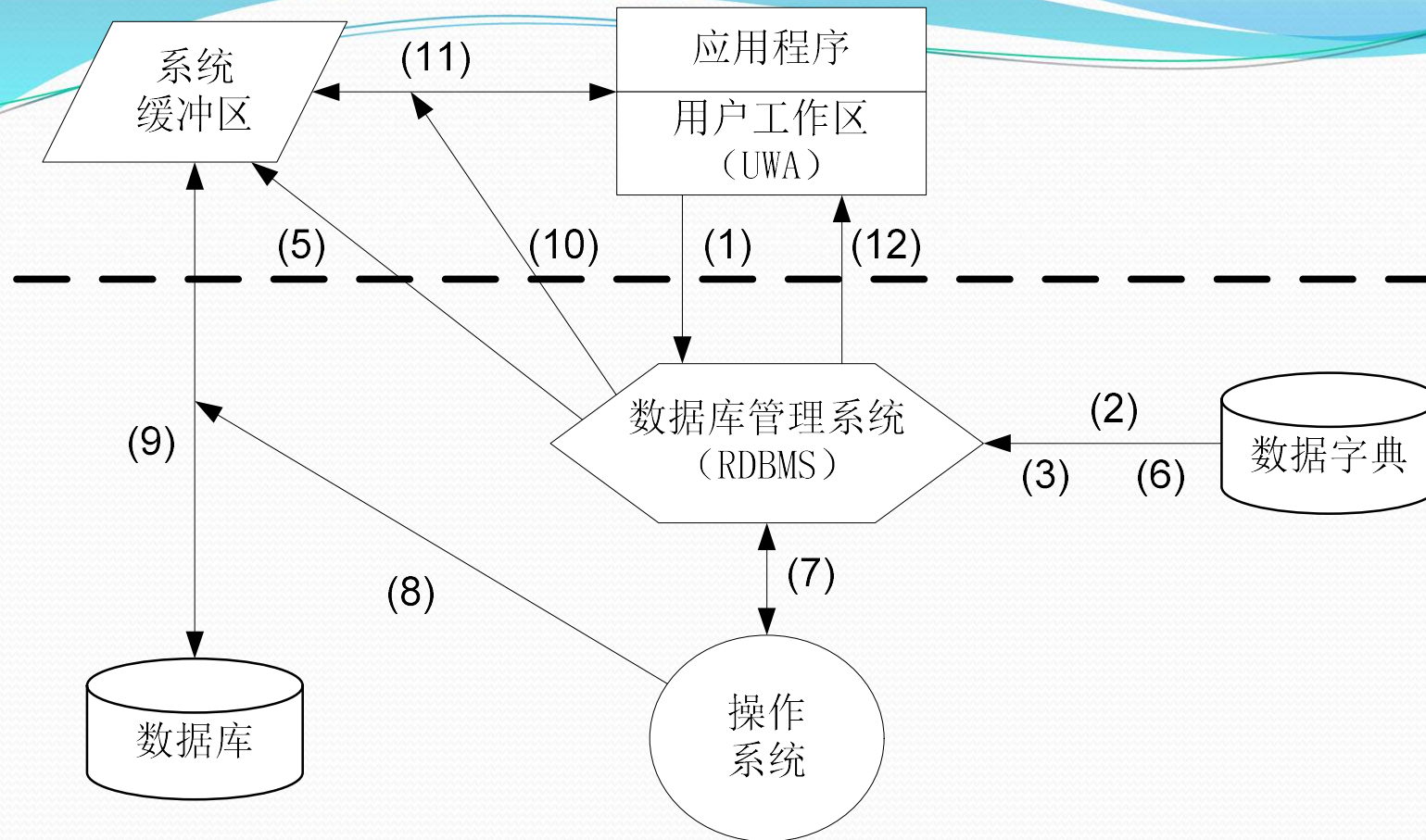
应用程序（或用户）从数据库读取一个数据通常需要以下步骤：

1. 应用程序（或用户）**A**向**DBMS**发出从数据库中读数据记录的命令；
2. **DBMS**对该命令进行**语法**检查、**语义**检查，并调用应用程序**A**对应的**子模式**，检查**A**的**存取权限**，决定是否执行命令，如果拒绝执行，则向用户返回错误信息；
3. 在决定执行该命令后，**DBMS**调用**模式**，依据子模式/模式映象的定义，确定应读入**模式中的哪些记录**；

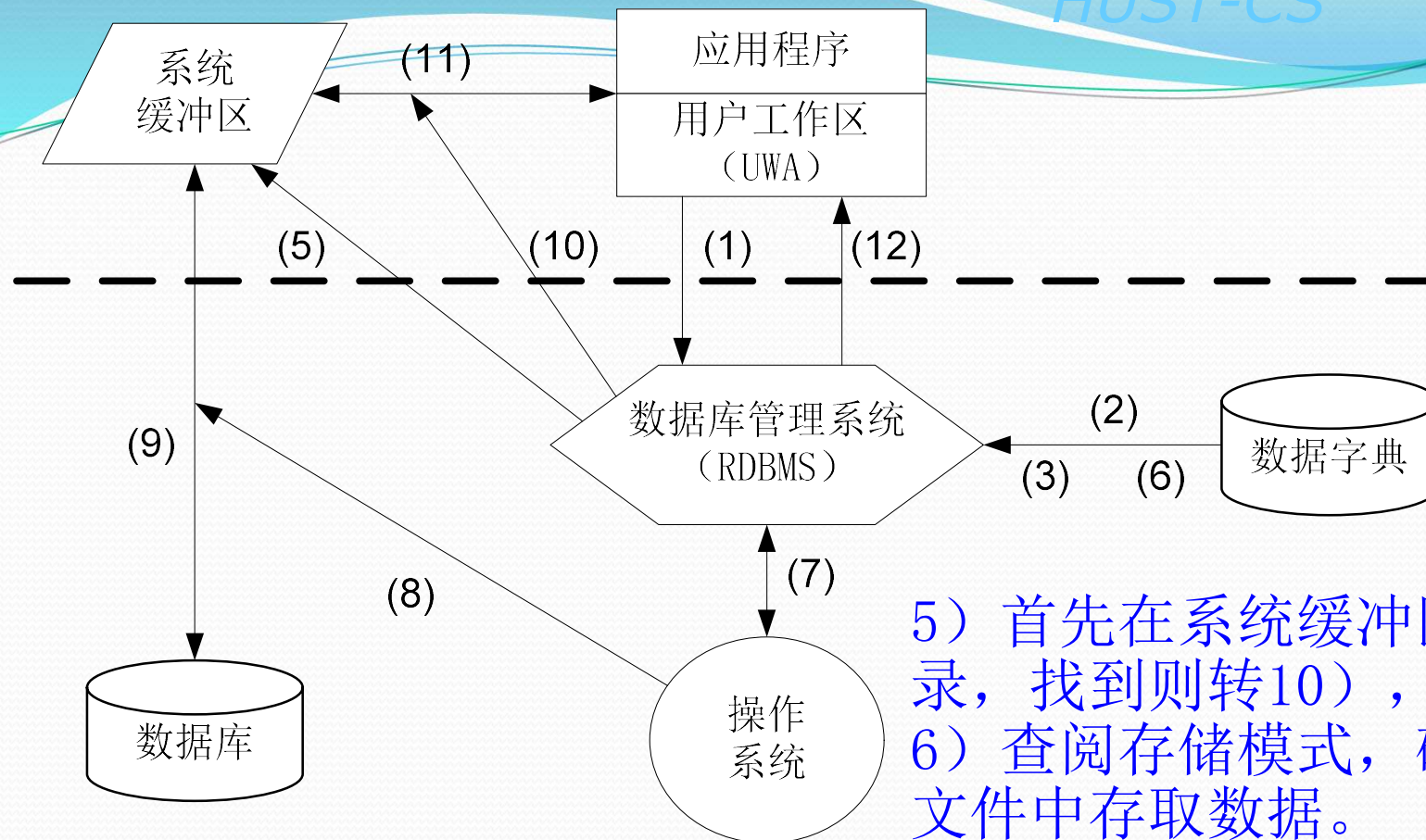
4. DBMS调用物理模式，依据模式/物理模式映象的定义，决定从哪个文件、用什么存取方式、读入哪个或哪些物理记录；
5. DBMS向操作系统发出执行读取所需物理记录的命令；
6. 操作系统执行读数据的有关操作；
7. 操作系统将数据从数据库的存储区送到系统缓冲区；
8. DBMS依据子模式/模式映象的定义，导出应用程序A所要读取记录的格式；
9. DBMS将数据记录从系统缓冲区传送到应用程序A的用户工作区；
10. DBMS向应用程序返回命令执行情况的状态信息。

图中显示了应用程序（用户）从数据库中读取记录的过程。执行其他操作的过程也与此类似。

- 课本第332页描述了RDBMS运行的12个步骤，分析的角度不同，不是从三层模式的转换这个角度分析，而是从DBMS功能调用序列的角度分析。



- 1) 应用程序发出SQL语句。
- 2) RDBMS进行语法、语义、存取权限检查。
- 3) RDBMS进行查询优化，转换成操作序列。
- 4) RDBMS执行存取操作序列（反复执行后续步骤）。



5) 首先在系统缓冲区中查找记录, 找到则转10), 否则转6)。
 6) 查阅存储模式, 确定如何从文件中存取数据。
 7) 向操作系统发出存取命令。

8) 操作系统存取文件。

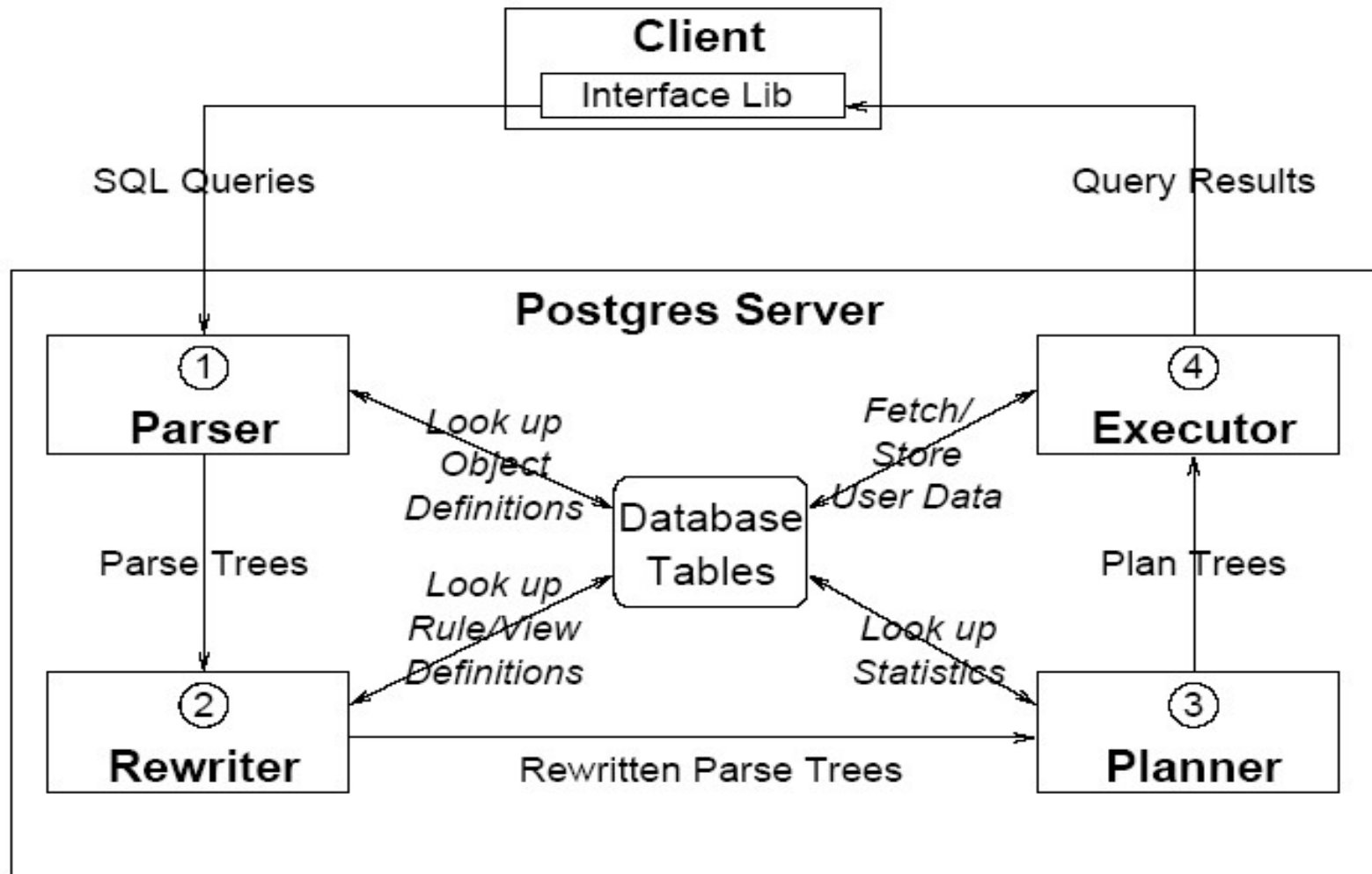
9) 操作系统将读取结果送至系统缓冲区。

10) RDBMS导出记录格式。

11) RDBMS将结果从系统缓冲区送至用户工作区。

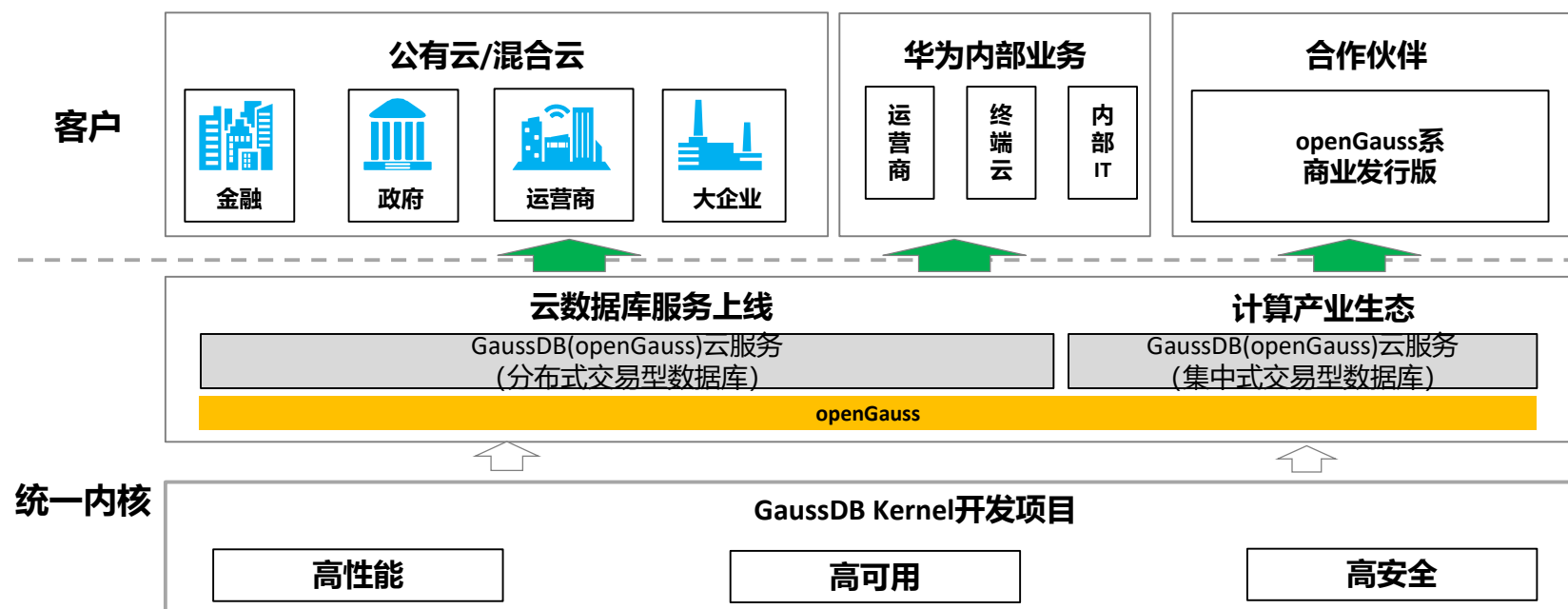
12) RDBMS返回状态信息。

Postgres的语句执行过程



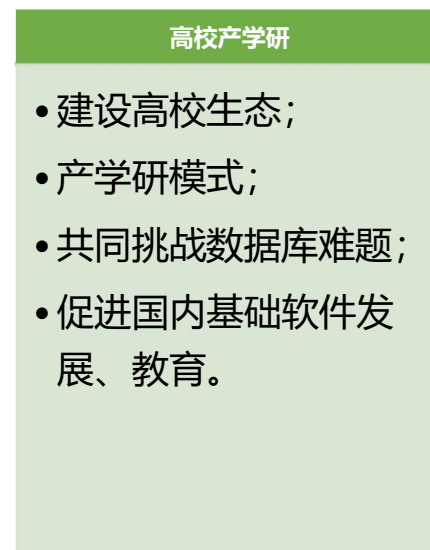
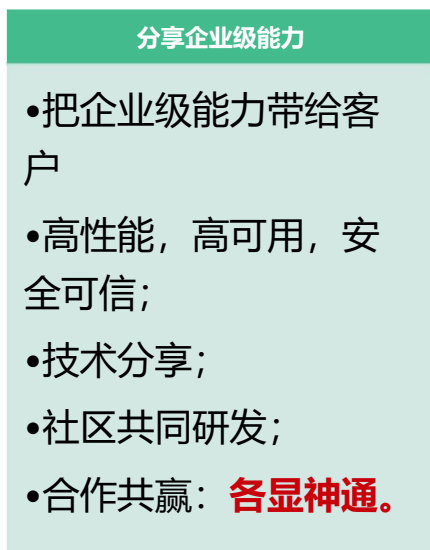
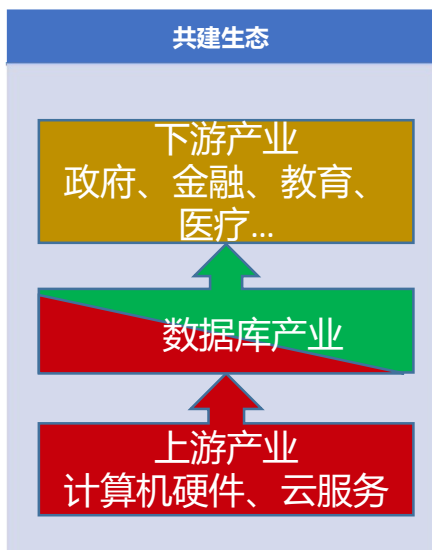
openGauss数据库：商用+自用+开源相结合，内核长期演进

- 华为公司内部配套和公有云的GaussDB服务均基于openGauss，内核将保持长期演进。

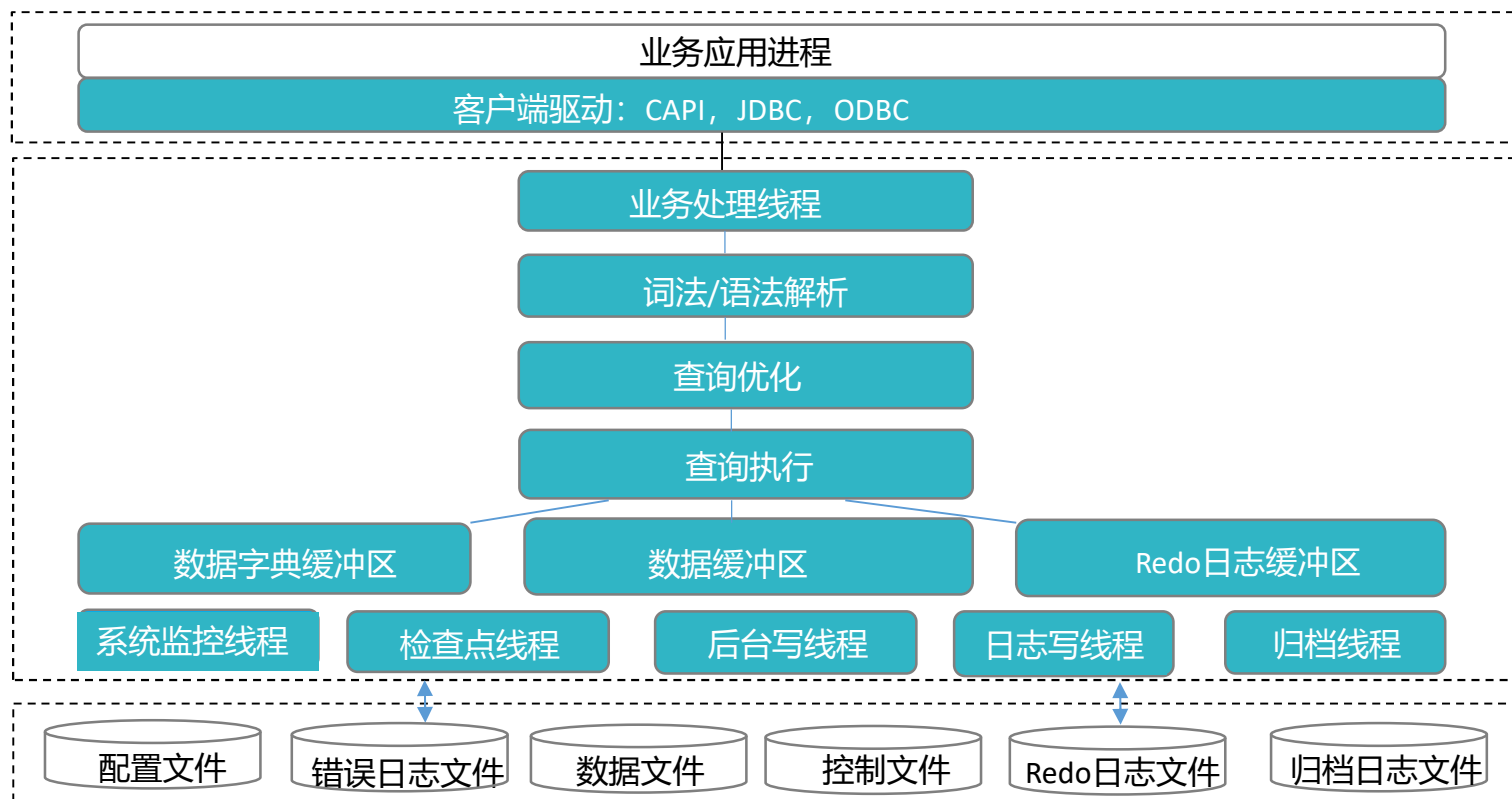


openGauss为什么开源?

- 战略：硬件开放、软件开源、使能伙伴；
- 通过openGauss开源社区运作，推广华为自有数据库生态，助力鲲鹏计算产业生态构建；
- 聚国内数据库人才，携手并进，共筑国产数据库事业。



openGauss体系架构



openGauss架构 VS. PostgreSQL架构 关键技术对比

- openGauss是衍生自PostgreSQL-XC，单机逻辑架构与PG接近。
- openGauss和PG在架构和关键技术上有根本性差异，尤其是存储引擎和优化器两大核心能力。

关键差异化因素		openGauss	PostgreSQL
运行时模型	执行模型	线程池模型 ，高并发连接切换代价小、内存损耗小，执行效率高，一万并发连接比最优性能损耗<5%。	进程模型 ，数据库进程通过共享内存实现通讯和数据共享。每个进程对应一个并发连接，存在切换性能损耗，导致多核扩展性问题。
事务处理	并发控制	64位事务ID ，使用CSN解决动态快照膨胀问题； NUMA-Aware引擎优化改造解决“五把大锁” 。	事务ID回卷，长期运行性能因为ID回收周期大幅波动；存在“五把大锁”的问题，导致事务执行效率和多处理器多核扩展性存在瓶颈。
	日志和检查点	增量Checkpoint机制 ，实现性能波动<5%。	全量checkpoint ，性能短期波动>15%。
	鲲鹏NUMA	NUMA改造、cache-line padding、原生spin-lock 。	NUMA多核能力弱 ，单机两路性能TPMC <60w。
数据组织	多引擎	行存、列存、内存引擎 ，在研DFV存储和原位更新。	仅支持行存。
SQL引擎	优化器	支持 SQL Bypass ，CBO吸收工行等 企业场景优化能力 。	支持CBO，复杂场景优化能力一般。
	SQL解析	ANSI/ISO标准SQL92、SQL99和SQL2003和 企业扩展包 。	ANSI/ISO标准SQL92、SQL99和SQL2003。

“Clog、WALInsert、WALWrite、ProcArray、XidGen” 这五种锁

openGauss 竞争力总览

把企业级数据库能力带给用户和伙伴

价值

openGauss提供面向多核的极致性能、全链路的业务和数据安全、基于AI的调优和高效运维的能力，全面友好开放，共同打造全球领先的企业级开源关系型数据库；

关键特性

高性能

- ① 两路鲲鹏性能150万tpmC；
- 面向多核架构的并发控制技术；
- NUMA-Aware数据结构；
- SQL-Bypass智能选路执行技术；
- ④ 面向实时高性能场景的内存引擎。

高可用 & 高安全

- ② 业务无忧，故障切换时间RTO<10 s；
- 精细安全管理：细粒度访问控制、多维度审计；
- 全方位数据保护：存储&传输&导出加密、动态脱敏、全密态计算。
- ⑤

易运维

- ③ 基于AI的智能参数调优，提供AI参数自动推荐；
- 慢SQL诊断，多维性能自监控视图，实时掌控系统性能表现；
- 提供在线自学习的SQL时间预测、快速定位，急速调优。

全开放

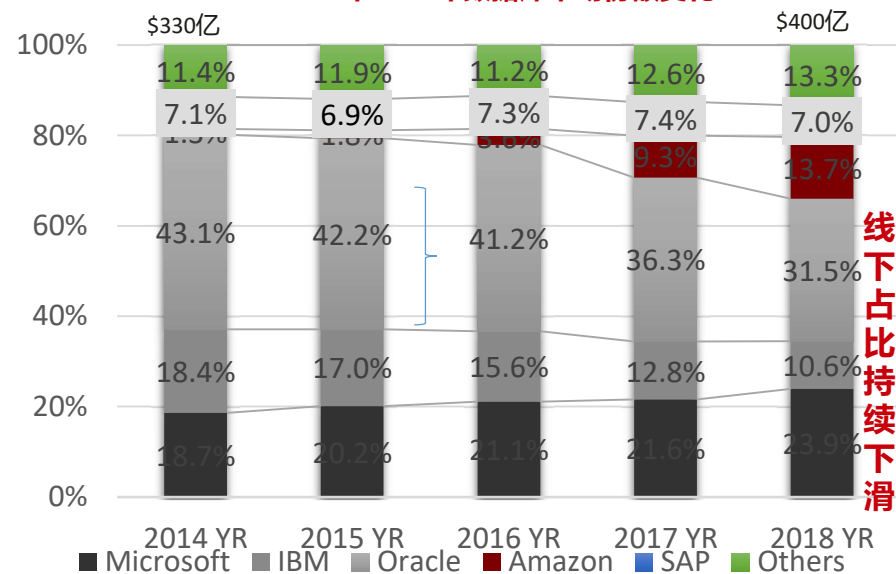
- 采用木兰宽松许可证协议，允许对代码自由修改，使用、引用；
- 数据库内核能力完全开放；
- 开放运维监控、开发和迁移工具；
- 开放伙伴认证、培训体系及高校课程。

云化数据库是大势所趋 (1)

数据库技术革新正在打破现有秩序，**云化，分布式，多模处理**是未来主要趋势

Source: Gartner

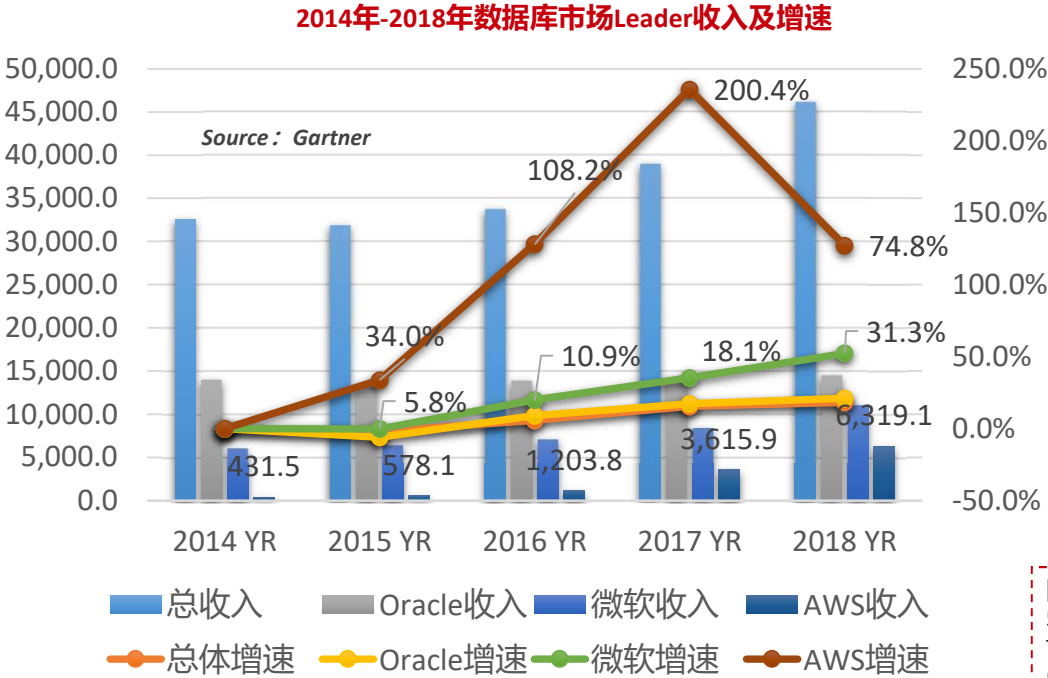
2014年-2018年数据库市场份额变化



国际权威研究机构Gartner 2019年5月发布《The Future of the Database Management System (DBMS) Market Is Cloud Based》报告，鲜明指出：数据库的未来是上云

传统线下数据库市场（以IBM+Oracle为代表）占比持续下滑；
2024年，数据库市场份额会达到\$650亿，**全球75%数据库以云服务的方式存在**

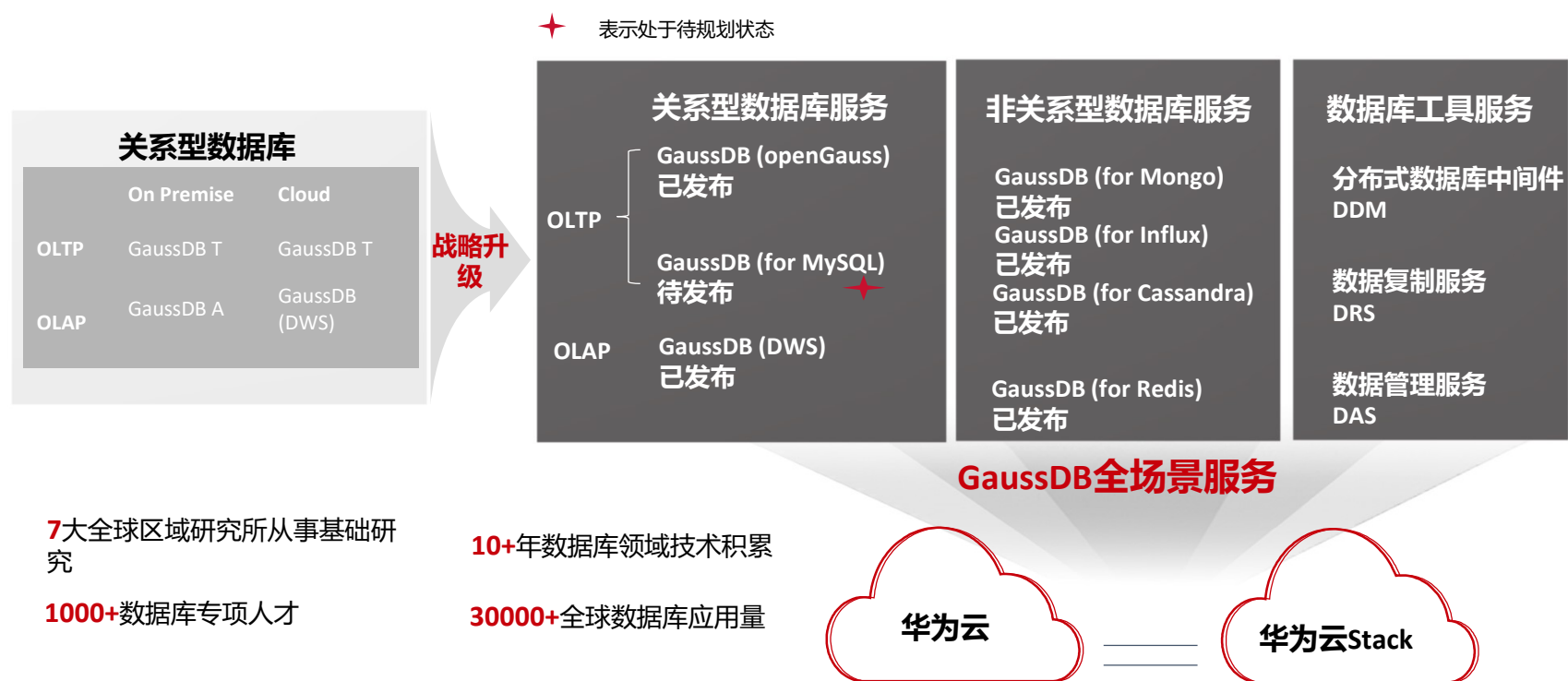
云化数据库是大势所趋 (2)



传统厂商Oracle增长停滞VS云厂商AWS增速平均每年90%，营收增加14倍
Gartner预计2024年，AWS的云数据库会超过Oracle

国际权威研究机构Gartner 2019年5月发布《The Future of the Database Management System (DBMS) Market Is Cloud》报告，鲜明指出：数据库的未来是上云

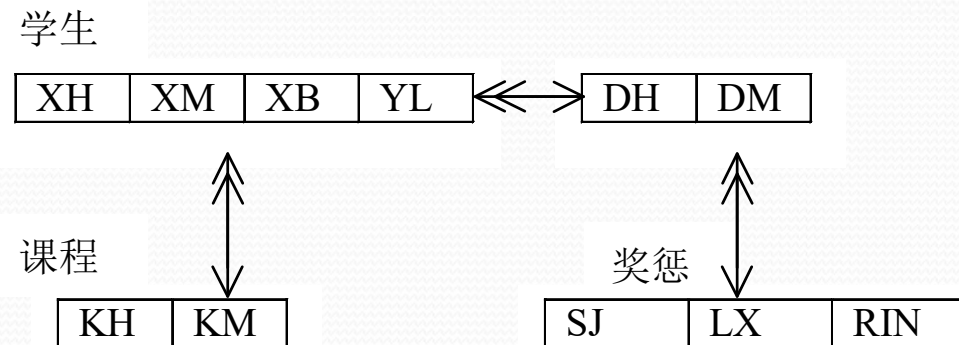
GaussDB数据库升级为全场景云服务



1.6 DBS特点

1、数据结构化

整体结构化



2、数据共享性高

- (1) 数据项一级;
- (2) 模式数据全体共享 (授权);
- (3) 新的应用。

3、数据冗余低 (redundancy)

——相同数据对象的重复构造与存放。

- 1) 冗余的问题: 花费空间, 修改麻烦, 潜在数据不一致性。
- 2) 适当冗余的好处: 可减少并发冲突。

4 数据独立性高 (independence)

——应用程序独立于其所使用数据的说明的特性。

1) 分类

①逻辑数据独立性

——模式变、变模式/子模式映像，子模式不变，应用程序不变。

②物理数据独立性。

——内模式变，应用程序不变。

2) 目标

①数据定义从应用程序中分离出来；

②编程不考虑物理细节；

③简化编程；

④提高应用程序稳定性，应变能力强，减少维护修改。

5 数据安全性 (security)

——防止非授权使用数据。

1) 身份鉴别;

2) 操作授权;

3) 加密存储。

6 数据完整性 (integrity)

——数据的正确性，有效性、相容性。

工龄 < 年龄，身高 < 3米。

原因：输入不当、修改不当、故障。

7 恢复能力强 (**recovery**)

——将DB从不正确状态恢复到某一正确状态。

备份+日志

8 数据一致性 (**consistency**)

——任何时刻对同一DB中相同数据的并发访问所获得的值应该是一致的，（有时须相同）。

飞机、火车订票系统的问题（奥运会、春运。。。）。

慕课讨论题

- 数据管理文件系统阶段和数据库系统阶段“数据独立性”有何不同？
- 数据库系统为什么要采用三级模式二级映像的系统结构？