

高级数据库技术



金培权

jpq@ustc.edu.cn

<http://staff.ustc.edu.cn/~jpq>

线上教学安排

■ 课程主页

- <http://staff.ustc.edu.cn/~jpgq/courses/adb-sse.html>

Also linked in <http://staff.ustc.edu.cn/~jpgq>

- **Blackboard:** <https://www.bb.ustc.edu.cn>

■ 课程QQ群（线上教学期间所有同学都需要加入）

- **1062258693**

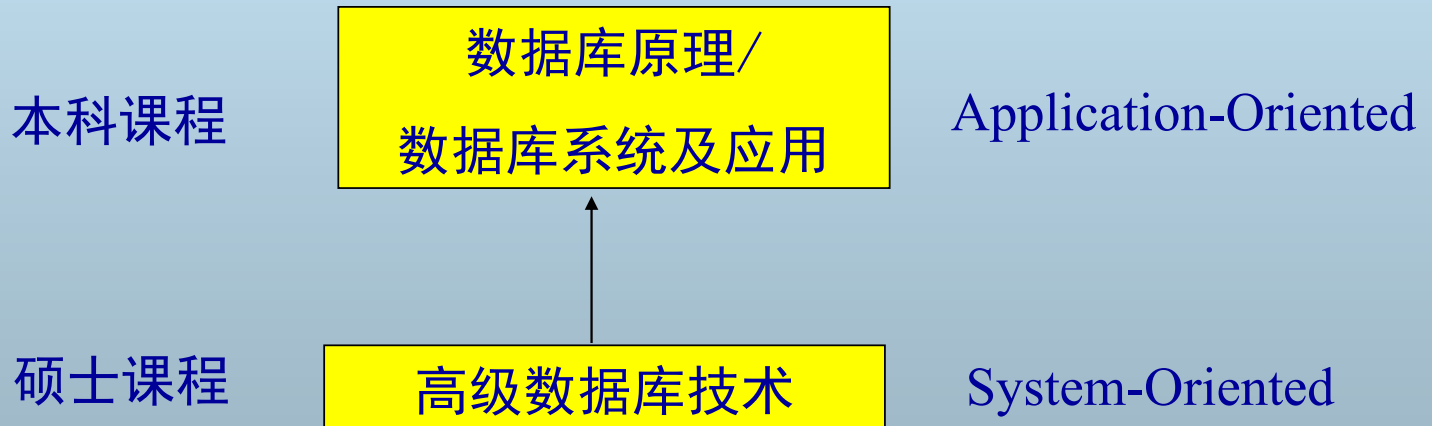
■ 直播平台

- **腾讯课堂：**每次上课前会把直播二维码发到**QQ**群里，大家扫码进入课堂
- **Classin（备选）：**登录**BB**系统后进入

■ 作业提交：**BB**系统

课程目的

- 掌握数据库设计和优化的方法
- 深入掌握DBMS的原理和技术
- 了解DBMS实现技术



课程内容

■ Part 1: Application-Oriented

- 关系数据库理论回顾
- 数据库设计与案例分析

■ Part 2 : System-Oriented

- 数据库文件与系统结构
- 索引结构
- 查询处理
- 缓冲区管理
- 恢复
- 锁与并发控制

课程主页

linked in
<http://staff.ustc.edu.cn/~jpq>

课程安排

■ 讲课＋实验

- 50学时讲授， 20学时实验

■ 教材

- **Database System Implementation(2th)**, H. Garcia-Molina, J.D. Ullman, J. Widom. 机械工业出版社

■ 参考文献

- **[R1] Raghu Ramakrishnan et al. Database Management Systems (Third Edition)**, McGraw-Hill & Tsinghua University Press, 2003
- **[R2] Abraham Silberschatz Henry F. Korth S. Sudarshan, Database System Concepts (Sixth Edition)**, China Machine Press, 2012 (中文版, 杨冬青 等译)
- **[R3] C. J. Date. An Introduction to Database System (Eighth Edition)**. China Machine Press, 2003 (中文版, 孟小峰 等译.)
- **[R4] 萨师煊, 王珊. 数据库系统概论(第5版)**, 高教出版社

课程安排

■ 考核

- 期末考试**40%**
- 期中考试**20%**
- 作业**20%**
- 实验**20%**

■ 预备知识

- 数据库系统原理，**SQL**，数据库应用编程

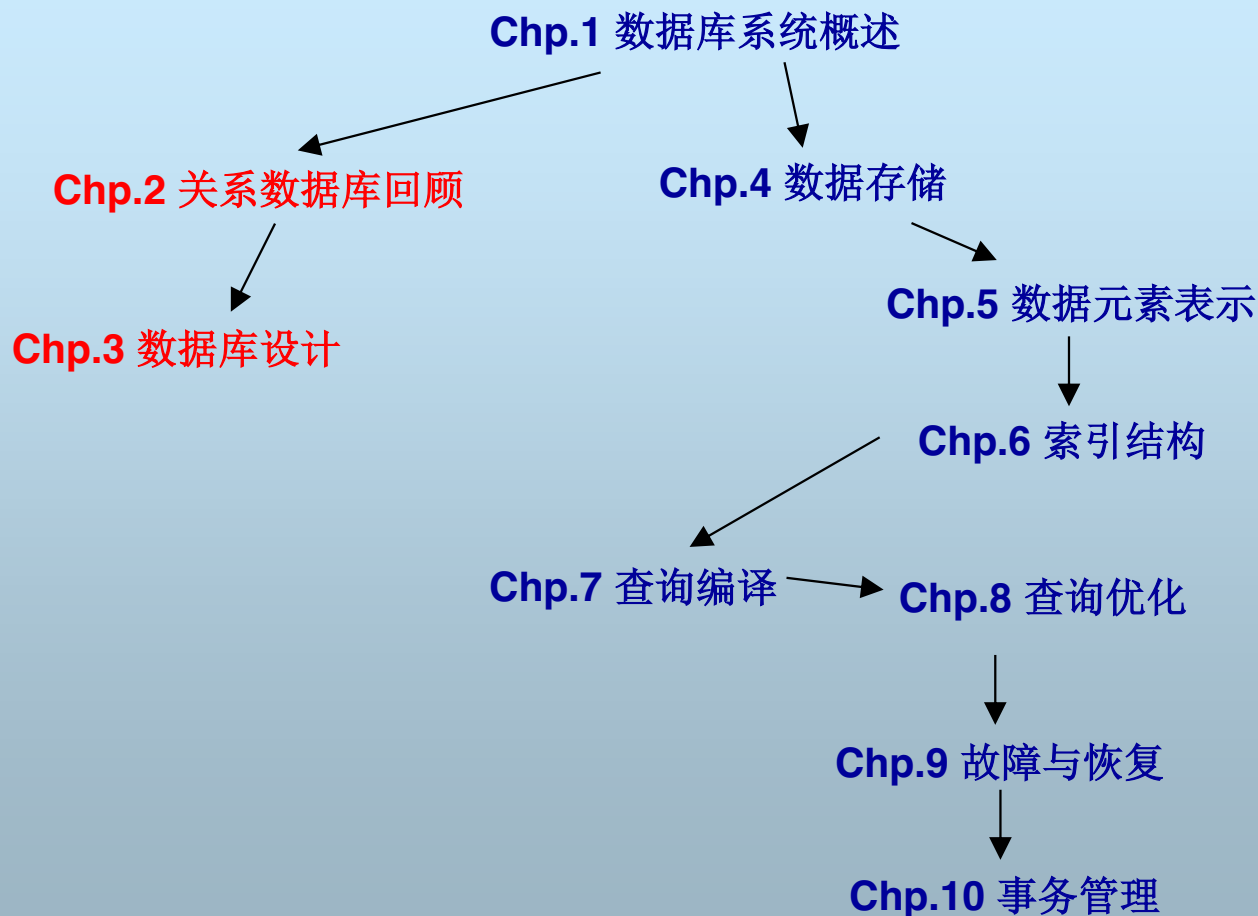
课程安排

■ 上机软件

- Oracle/MySQL/MS SQL Server
- Sybase PowerDesigner 16.6
- C++/Java

密码: db202

课程知识结构



第1章 数据库系统概述



主要内容

- 数据库系统的基本概念
- DBMS实现问题
- 数据库设计问题
- 数据库存取问题
- 数据库技术的发展

一、数据库系统的基本概念

- 数据
- 数据库
- 数据库模式
- 数据库管理系统
- 数据库系统

1、数据

- 数据(Data)是数据库中存储的基本对象

- 数据的定义

- 人们用来反映客观世界而记录下来的可以鉴别的符号

- 数据的种类

- 数值数据：0—9
- 非数值数据：字符、文字、声音、图形、图像等



1、数据

数据的特点

- 数据与其语义是不可分的

例子1：93是一个数据

- 语义1：学生某门课的成绩
- 语义2：某人的体重
- 语义3：软件学院2017级学生人数

例子2：学生档案记录（李明，197205，中国科大，1990）

- 语义1：学生，出生年月，所在学校，毕业年份
- 语义2：学生，出生年月，录取大学，入学时间

2、数据库

■ 数据库的定义

- 数据库(Database,简称DB)是长期储存在计算机内、有组织的、可共享的大量数据的集合

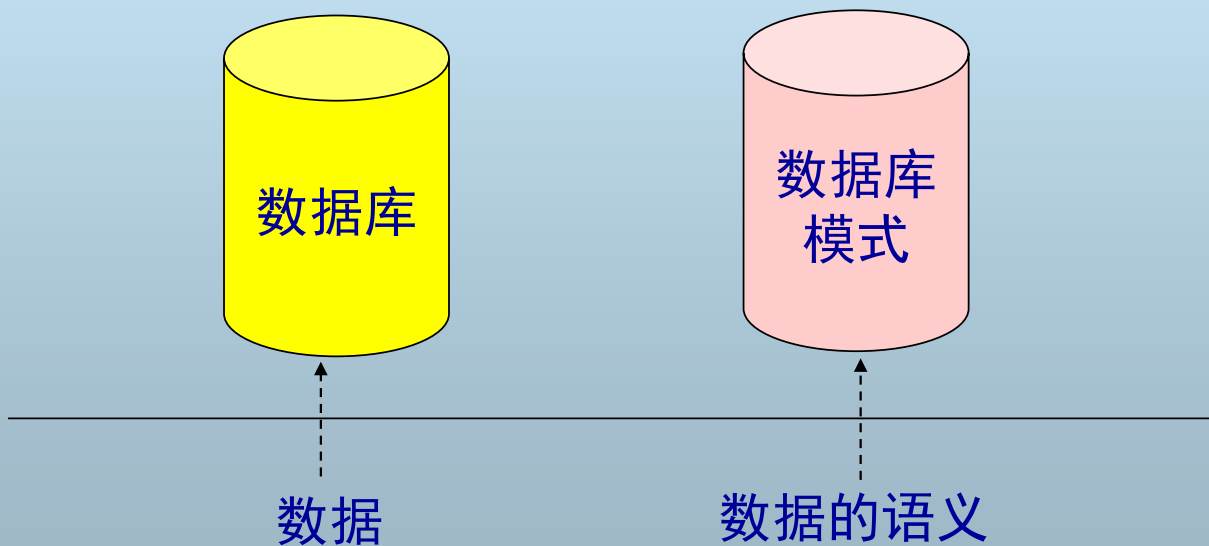
■ 数据库的基本特征

- 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
- 数据间联系密切，具有最小的冗余度和较高的独立性
- 服务于某个特定的应用

- 例：图书馆的图书数据库、机场的航班数据库、银行数据库.....

3、数据库模式 (Schema)

- 数据库模式是数据库中全体数据的逻辑结构和特征的描述



举例

学号	姓名	年龄
001	张三	20
002	李四	21
003	王五	22



学生(学号:char, 姓名:char, 年龄:int)

学号	姓名	年龄
----	----	----

001	张三	20
002	李四	21
003	王五	22

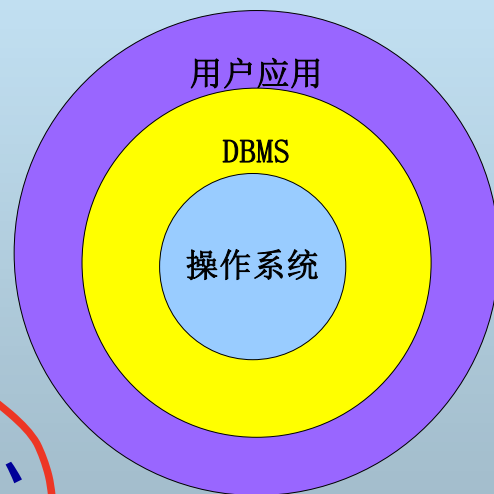
模式

数据库

4、数据库管理系统

■ DBMS (Database Management System)，是计算机程序的集合，用于创建和维护数据库

- 位于操作系统和用户应用之间
- 总是基于某种数据模型
- 数据库厂商的产品通常指DBMS，如Oracle11g、Microsoft SQL Server 2008、DB2 8.0、MySQL 5.7等

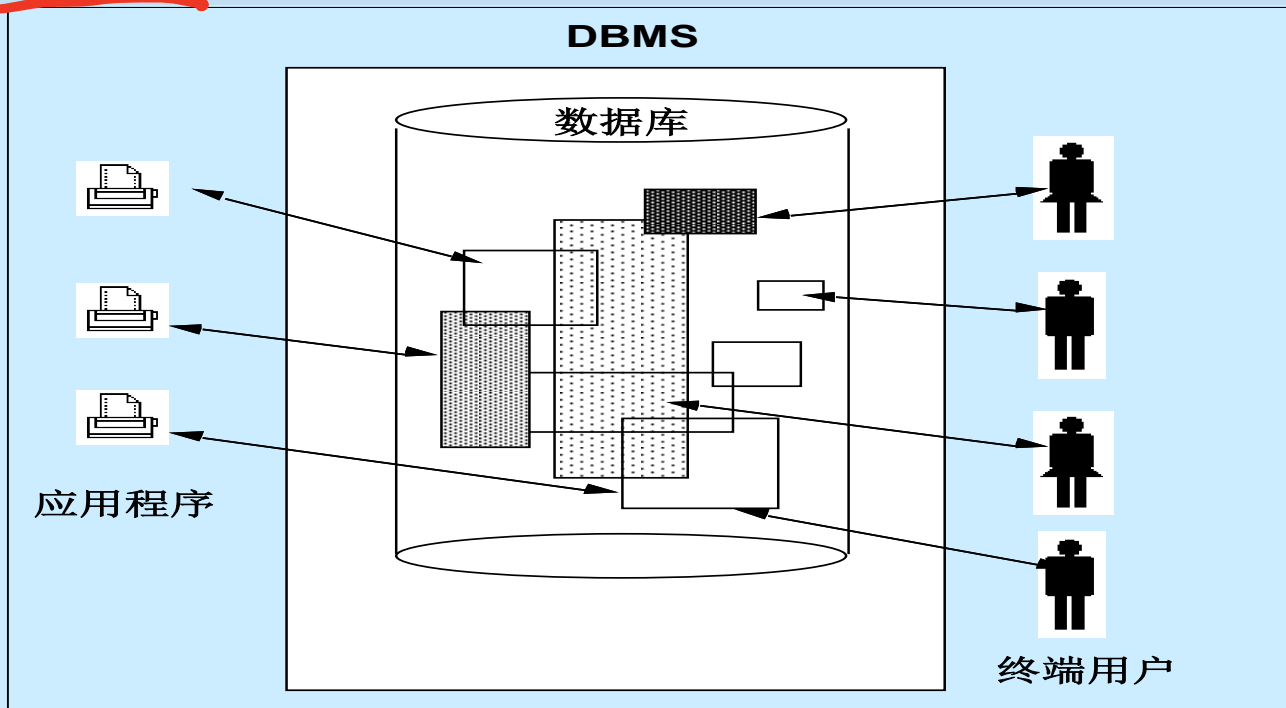


↓
面向对象数据模型

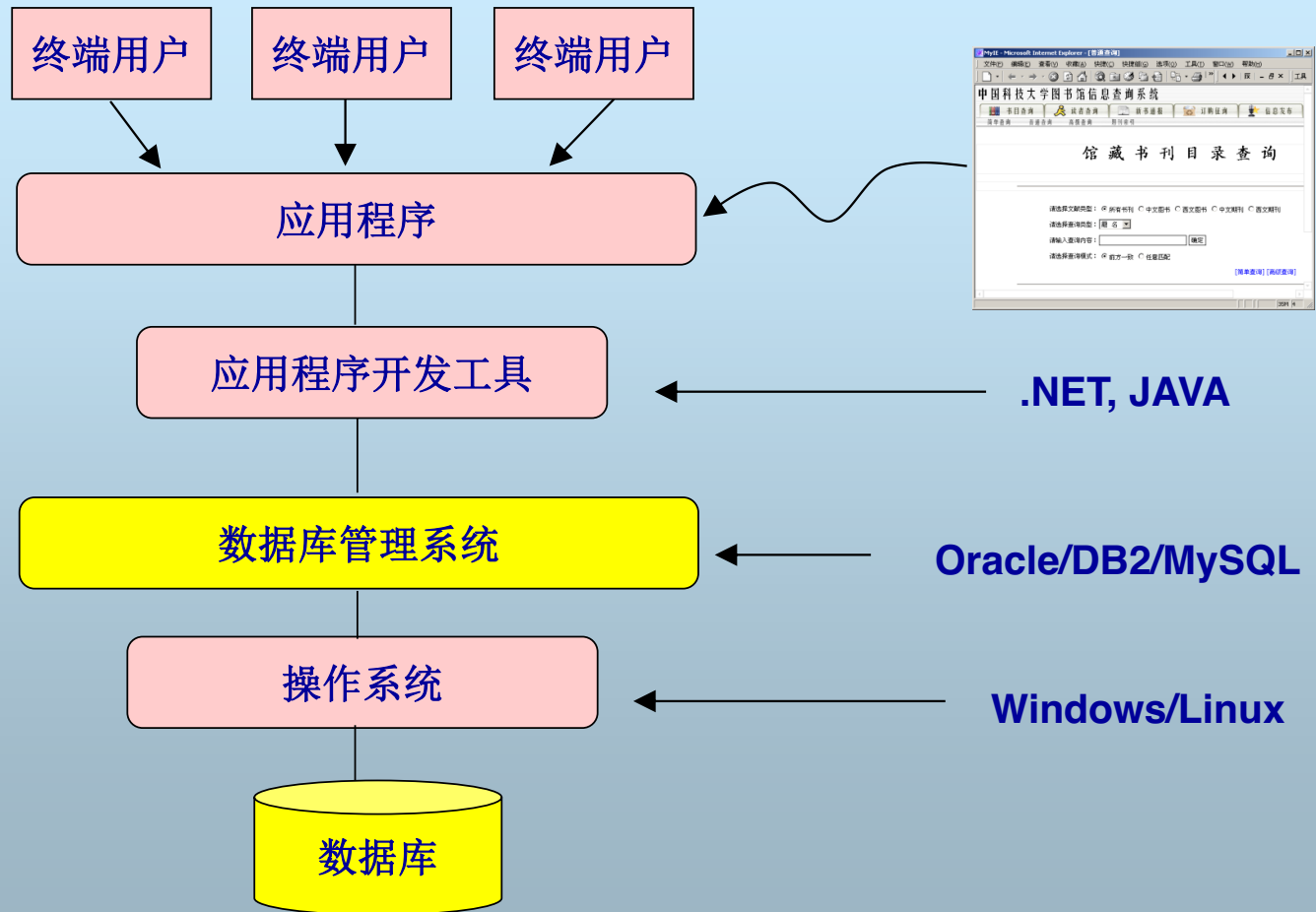
→ 关系数据模型

5、数据库系统

- **DBS (DataBase System)**，指在计算机系统中引入了数据库后的系统，即采用了数据库技术的计算机系统



5、数据库系统



6、数据库系统的特点

- 数据结构化
- 数据的共享性高，冗余度低
- 数据独立性高
- 数据由**DBMS**统一管理和控制

7、数据库系统中的三个主要问题

- 如何设计和实现一个DBMS来高效的组织和管理数据库？——**DBMS实现问题**
- 如何针对特定应用的需求设计一个合理的数据库结构？——**数据库设计问题**
- 应用程序如何有效地存取数据库中的数据？——**数据库存取问题**

二、DBMS实现问题

- Oracle、Informix、DB2、MS SQL Server、MySQL
 - 数据结构：关系
 - 数据操作：关系代数 & SQL 语言
 - 存储：文件

1、一个虚拟的DBMS: Megatron2000

■ UNIX下的关系型DBMS，支持SQL

■ 实现细节

- 关系通过文件（ASCII）存储，e.g. R存储在 /usr/db/R中

Smith	#	123	#	CS
Jones	#	522	#	EE

- 数据库模式存储在特定的文件（ASCII）中

R1	#	A	#	INT	#	B	#	STR	...
R2	#	C	#	STR	#	A	#	INT	...

2、Megatron 2000 使用示例

```
% MEGATRON2000
    Welcome to MEGATRON 2000!
&

& quit
%
```

2、Megatron 2000 使用示例

选择操作

```
& select *  
  from R #
```

<u>name</u>	<u>id</u>	<u>dept</u>
SMITH	123	CS

&

2、Megatron 2000 使用示例

```
& select A,B  
  from R,S  
 where R.A = S.A and S.C > 100 #
```

<u>A</u>	<u>B</u>
123	CAR
522	CAT

&

3、Megatron 2000如何执行查询？

■ 执行 “`select * from R where condition`”:

- 读数据字典获取R的属性
- 读R对应的文件，对于每一行数据：
 - ◆ 检查条件
 - ◆ 如果条件满足，则输出

3、Megatron 2000如何执行查询？

■ 执行“*select A,B from R,S where condition*”:

- 读数据字典获取R和S的属性

- 读R文件，对于每一行数据:

$O(m \times n)$

 - ◆ 读S文件，对于每一行数据:

 - 生成连接元组

 - 检查条件

 - 若条件满足，则输出

Megatron 2000功能总结

- 可以定义表
- 可以进行数据操作
 - 数据更新
 - 数据查询

4、Megatron 2000 存在什么问题？

■ 元组平铺在磁盘上

存储问题

- 将 'EE' 改为 'ECON', 需要重写整个文件
- ASCII 存储过于昂贵
- 删除操作同样代价很高

dracal 单表超 4M 就需分表

分库分表 → 解决大数据问题
不是长久办法

4、Megatron 2000 存在什么问题？

■ 低级的查询处理

- 例, `select *`

`from Student S, SC`

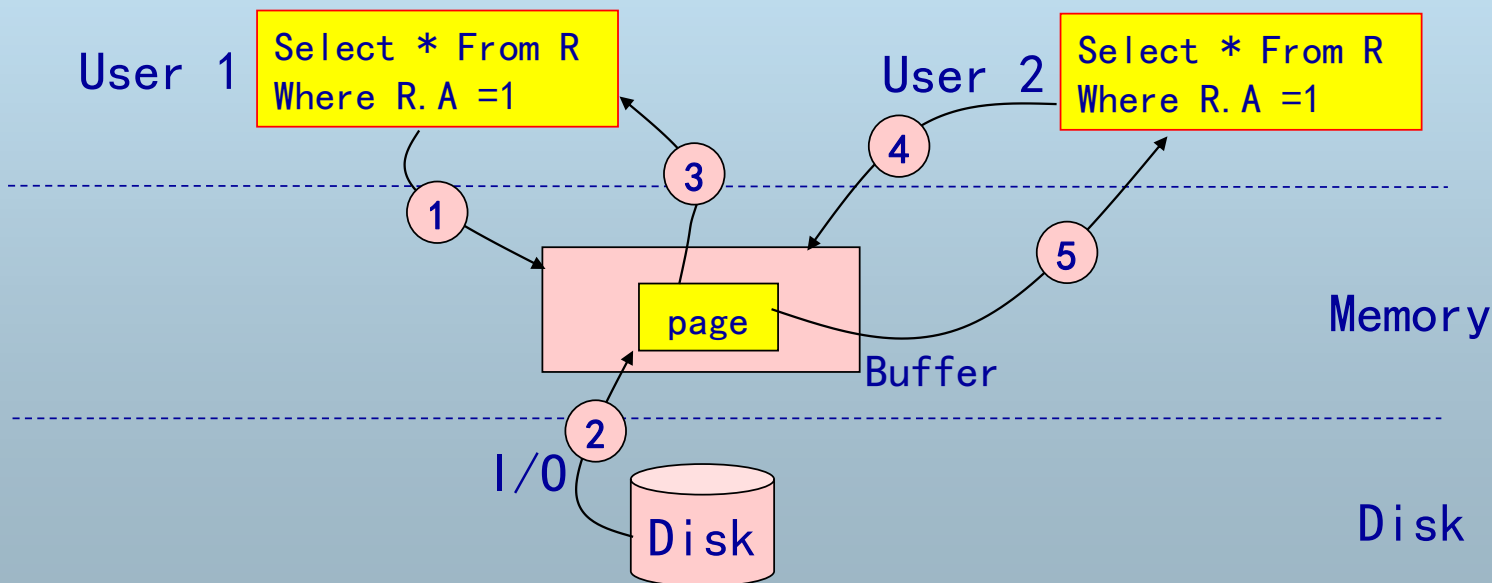
`where S.sno = SC.sno and SC.credit > 3`

- 先做连接比先做选择效率要低
- 未考虑更高效的连接算法

4、Megatron 2000 存在什么问题？

■ 没有缓冲区管理

- 数据直接从磁盘存取，磁盘I/O的代价昂贵
- 需要Buffer来加速数据存取效率

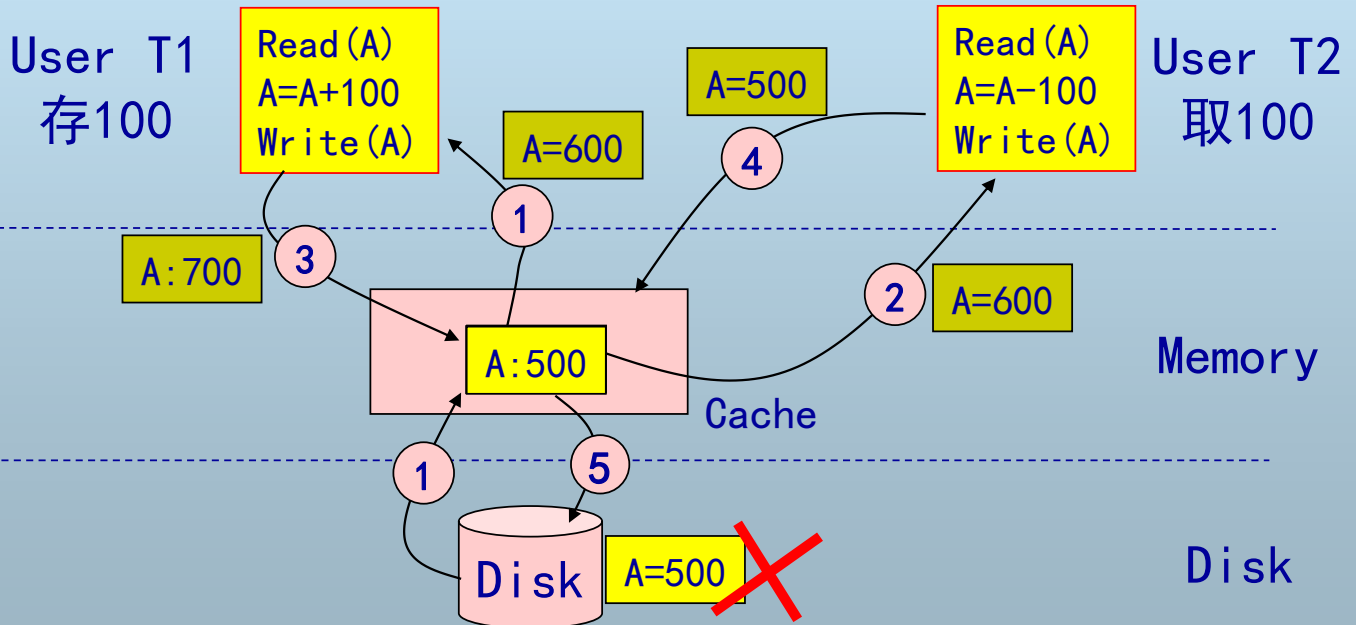


4、Megatron 2000 存在什么问题？

■ 没有并发控制

- 多用户同时存取数据时数据一致性得不到保证

执行顺序：R1 (A) R2 (A) W1 (A) W2 (A)



4、Megatron 2000 存在什么问题？

■ 没有索引：数据查询效率低

- 不能快速地根据给定键值查询元组

- 总是要读入整个关系

没有索引，要遍历

若 page size = 8 KB, page I/O 10ms

1 MB (128 pages)-----1.28 s

128 MB (16384 pages)-----163.8 s

1 GB (131072 pages)-----1310.7 s \approx 21.8 min

4、Megatron 2000 存在什么问题？

■ 没有可靠性

- 发生数据库系统故障时没有恢复机制
- 易出现数据不一致的情形

4、Megatron 2000 存在什么问题？

■ 没有应用程序编程接口（API）

- 应用如何存取数据库？

4、Megatron 2000 存在什么问题？

■ 糟糕的数据字典组织



5、Megatron 2000 总结

■ 一个糟糕的DBMS

三、数据库设计问题

- 如何把现实世界数据表达到数据库系统中？
- 针对一个具体应用，应该如何构造一个适合于它的数据库模式？

三、数据库设计问题

- 数据库模式设计不规范会带来一系列的问题
 - 数据冗余
 - 更新异常
 - 插入异常
 - 删除异常

示例关系模式R

示例关系模式 **R(Tname, Addr, C#, Cname)**

一个教师只有一个地址（户口所在地）

一个教师可教多门课程

一门课程只有一个任课教师

因此**R**的主码是（**C#**）

Tname	Addr	<u>C#</u>	Cname
T1	A1	C1	N1
T1	A1	C2	N2
T1	A1	C3	N3
T2	A2	C4	N4
T2	A2	C5	N5
T3	A3	C6	N6

1、问题（1）：数据冗余

- 教师T1教了三门课程，他的地址被重复存储了2次

Tname	Addr	<u>C#</u>	Cname
T1	A1	C1	N1
T1	A1	C2	N2
T1	A1	C3	N3
T2	A2	C4	N4
T2	A2	C5	N5
T3	A3	C6	N6

2、问题（2）：更新异常

- 如果T1的地址变了，则需要改变3个元组的地址；若有一个未更改，就会出现数据不一致。但DBMS无法获知这种不一致

Tname	Addr	<u>C#</u>	Cname
T1	A1	C1	N1
T1	A1	C2	N2
T1	A1	C3	N3
T2	A2	C4	N4
T2	A2	C5	N5
T3	A3	C6	N6

3、问题（3）：插入异常

- 如果要增加一名教师，但他还未带课，则C#和Cname为空，但由于C#是主码，为空违反了实体完整性，所以这名教师将无法插入到数据库中

Tname	Addr	<u>C#</u>	Cname
T1	A1	C1	N1
T1	A1	C2	N2
T1	A1	C3	N3
T2	A2	C4	N4
T2	A2	C5	N5
T3	A3	C6	N6

4、问题（4）：删除异常

- 如果教师T3现在不带课了，则需将T3的元组删去，但同时也把他的姓名和地址信息以及C6课程信息删掉了

Tname	Addr	<u>C#</u>	Cname
T1	A1	C1	N1
T1	A1	C2	N2
T1	A1	C3	N3
T2	A2	C4	N4
T2	A2	C5	N5
T3	A3	C6	N6

5、如何解决？

■ 方法：模式分解

● 方法1：R分解为

◆ R1(Tname, Addr)

◆ R2(C#, Cname)

授课信息丢失了

5、如何解决？

■ 方法：模式分解

● 方法1：R分解为

◆ R1(Tname, Addr)

◆ R2(C#, Cname)

授课信息丢失了

● 方法2

◆ R1(Tname, Addr, C#)

◆ R2(C#, Cname)

R1中问题依然存在

5、如何解决？

■ 方法：模式分解

● 方法1：R分解为

◆ R1(Tname, Addr)

◆ R2(C#, Cname)

授课信息丢失了

● 方法2

◆ R1(Tname, Addr, C#)

◆ R2(C#, Cname)

R1中问题依然存在

● 方法3

◆ R1(Tname, Addr)

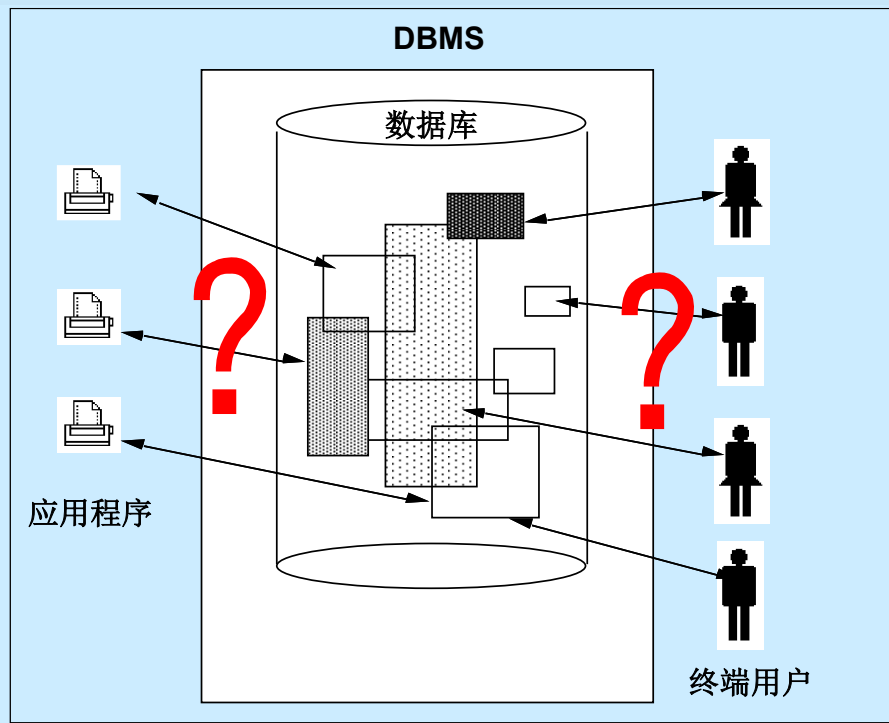
◆ R2(Tname ,C#, Cname)

基本解决问题，但又带来join
查询代价，可能会导致数据库
性能不能满足需求

5、如何解决？

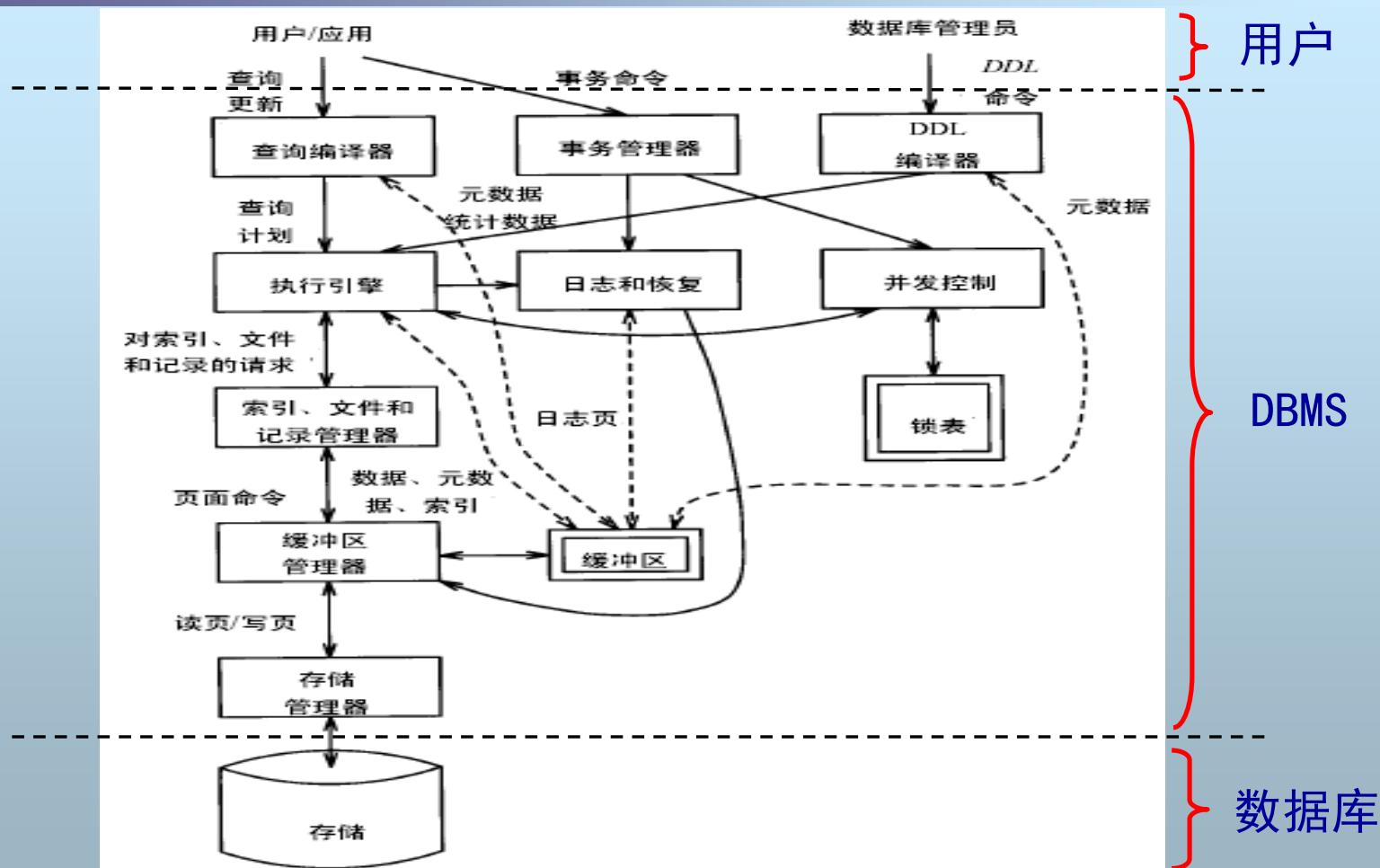
- 到底什么样的数据库模式才合理？怎么分解才能满足要求？标准是什么？如何实现？
——数据库设计要解决的主要问题

四、数据库存取问题



- 数据库的存取？
 - 增、删、改、查
- 数据库模式的存取？
- 数据库访问控制信息的存取？

四、数据库存取问题



四、数据库存取问题

- 用户与数据库的唯一接口——数据库语言
- DBMS支持用户通过数据库语言进行数据存取
- 有三类数据库语言
 - 数据定义语言（Data Definition Language, DDL）——存取数据库模式
 - 数据操纵语言（Data Manipulation Language, DML）——存取数据库数据
 - 数据库控制语言（Data Control Language, DCL）——存取访问控制信息

SQL的基本组成

定义概念模式



本课程的重点

- DBMS实现问题 ✓
- 数据库设计问题 ✓
- 数据库存取问题 ✕

五、数据库技术的发展

- **1961: GE的C.W. Bachman**设计了历史上第一个DBMS——网状数据库系统IDS (Integrated DataStore) **[1973, 图灵奖]**
- **1968: IBM**设计了层次数据库系统IMS
- **1969: CODASYL的DBTG**发表了网状数据模型报告, 奠定了网状数据库技术
- **1970: IBM的E.F. Codd**提出了关系数据模型, 奠定了关系数据库理论基础) **[1981, 图灵奖]**
- **1974: IBM的Boyce和Chamberlin**设计了SQL语言
- **1973~1976: E.F. Codd**设计了System R, **M. Stonebraker**设计了Ingres
- **1976: IBM的Jim Gray**提出了一致性、锁粒度等设计, 奠定了事务处理基础) **[1998, 图灵奖]**
- **1977: Larry Ellison**创建了Oracle公司, **1979年**发布Oracle 2.0, **1986年**Oracle上市
- **1983: IBM**发布DB2

五、数据库技术的发展

- 1985：面向对象数据库技术提出
- 1987：Sybase 1.0发布
- 1990：M. Stonebraker发表“第三代数据库系统宣言”，提出对象关系数据模型 [2014, 图灵奖]
“For fundamental contributions to the concepts and practices of underlying modern database systems”
- 1987~1994：Sybase和Microsoft合作，发布 Sybase SQL Server 4.2。破裂后Sybase继续发布Sybase ASE 11.0
- 1996：Microsoft发布Microsoft SQL Server 6.5
- 1996：开源的MySQL正式发布
- 1998：提出了半结构化数据模型（XML1.0）
- 2005，M. Stonebraker等开发完成C-Store，Column-based DBMS
- 2007，NoSQL(非关系型数据库)在Web领域大行其道。Amazon (SimpleDB/Dynamo), Google (BigTable/LevelDB), Facebook (RocksDB/Cassandra), MongoDB, HBase, Redis, etc.

六、数据库领域的出版物

■ 国际会议

- **A类：SIGMOD、VLDB、ICDE** (**DB三大会议**)
- **B类：EDBT、ICDT、CIDR、CIKM、DASFAA**
- **C类：DEXA、APWeb-WAIM、ER、SSTD、SSDBM、MDM、WebDB、ADBIS等**
- **中国数据库学术会议NDBC**

■ 国际期刊

- **A类：VLDB Journal、TKDE、TODS** (**DB三大期刊**)
- **B类：DKE、Information Systems、GeoInformatica等**

可参考CCF计算机国际会议与期刊排名

本章小结

- 数据库系统的基本概念
- DBMS实现问题
- 数据库设计问题
- 数据库存取问题
- 数据库技术的发展

Next.....

■ 关系数据库回顾