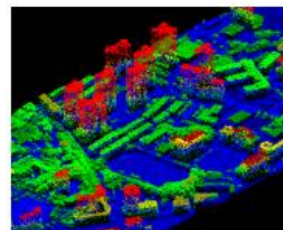
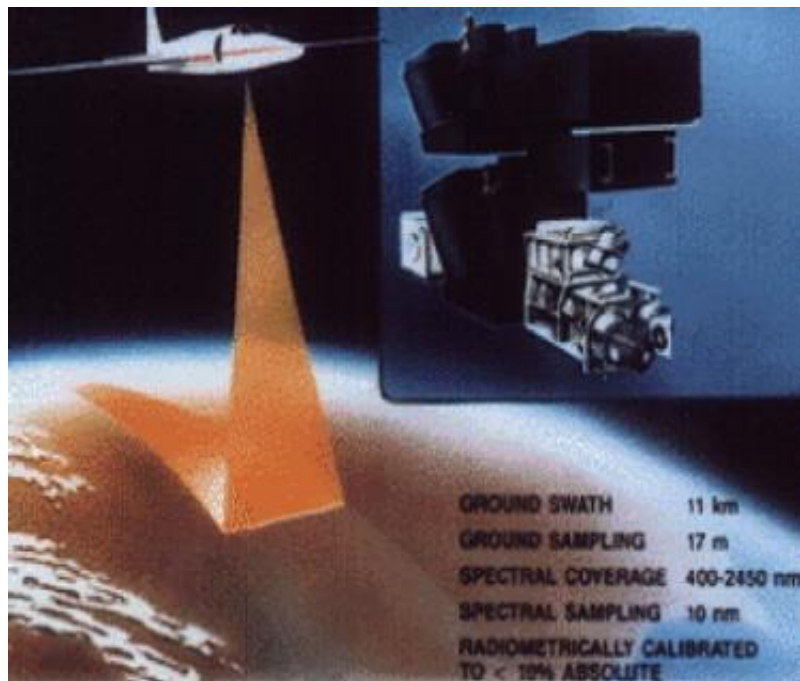


数据库系统概论

计算机学院 罗晖

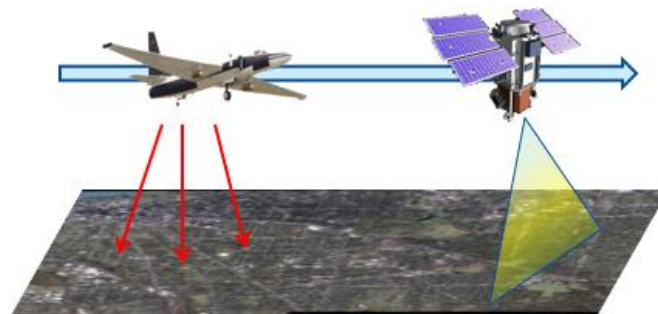
Email: luohui2006@foxmail.com

我的研究方向：遥感图像智能处理，欢迎与我交流！



2012
LiDAR

2016
光学高分



教材及参考书

教材

- 王珊，萨师煊：数据库系统概论(第五版)，高等教育出版社

参考书

- Abraham Silberschatz, Henry F. Korth, S. Sudarshan. Database System Concepts, 6th Edition. McGraw-Hill, 2010

学习本课程的意义

□ 1.数据库技术与日常生活

■ 方便日常生活：

□ 在图书馆、银行、票务系统等场所广泛应用

■ 极大提高办公效率：

□ ACCESS、EXCEL等软件

低端需求：

掌握一定的数据库知识将有助于大家生活更加便捷

学习本课程的意义

□ 2.数据库技术在信息技术(IT)产业中的地位

■ IT产业四大支柱:

□ 操作系统、网络、计算机语言、**数据库**

■ IT产业两大热门技术:

□ 网络、**数据库**

高端需求:

具备扎实的数据库知识将有助于提高大家的社会竞争力

腾讯2020校园招聘

面向人群：毕业时间在2019年9月-2020年8月之间（国内院校以毕业证、港澳台及海外院校以学位证时间为准）

🔧 技术类

📦 产品类

🎨 设计类

📊 市场/职能类

📢 专项招聘 hot

软件开发

技术运营

安全技术

软件测试

技术研究

岗位方向

软件开发-后台开发方向

岗位描述

从事腾讯产品服务后台的架构设计、开发、优化及运营工作；使用最优秀的架构设计及算法实现，在网络接入、业务运行逻辑、用户数据存储、业务数据挖掘等方向，为海量互联网用户提供稳定、安全、高效和可靠的专业后台支撑体系。

岗位要求

编程基本功扎实，掌握C/C++/JAVA等开发语言、常用算法和数据结构；

熟悉TCP/UDP网络协议及相关编程、进程间通讯编程；

了解Python、Shell、Perl等脚本语言；

了解MYSQL及SQL语言、编程，了解NoSQL, key-value存储原理；

全面、扎实的软件知识结构，掌握操作系统、软件工程、设计模式、数据结构、数据库系统、网络安全等专业知识；

了解分布式系统设计与开发、负载均衡技术，系统容灾设计，高可用系统等知识。

注：该岗位“招聘城市”在简历投递截止日前会有部分调整，请密切关注，腾讯公司对招聘信息保留最终解释权

工作地点

深圳总部 上海 广州

- 了解MYSQL及SQL语言.....
- 掌握数据库系统的专业知识.....

校园招聘 campus recruitment

此时此刻,非我莫属

▶ 应届生招聘岗位

数据研发工程师 Software Engineer, Data Technology

▶ 岗位描述 Job Description

如果你想参与阿里大数据的采集、存储、处理,通过分布式大数据平台加工数据,支持业务管理决策,
如果你想参与阿里大数据体系的设计、开发、维护,通过数据仓库、元数据、质量体系有效的管理和组织几百P的数据,
如果你想参与阿里大数据产品的研发,通过对数据的理解,发挥你的商业sense,发掘数据价值,探索大数据商业化,
如果你想接触世界领先的大数据处理与应用的技术和平台,获得大数据浪潮之巅的各类大牛的指导,
那就加入我们吧!

▶ 岗位要求 Qualifications

如果你所学专业是计算机、数学、统计、数据科学与大数据技术等相关专业;
如果你有强的动手能力和学习能力,熟悉一门数据处理语言,如SQL、JAVA、Python、Perl等,熟悉unix或者linux操作;
如果你具备扎实的专业基础,良好的沟通能力和团队合作,主动积极,乐于面对挑战;
如果你有参与过数据处理、分析、挖掘等相关项目更好;
如果你对Hadoop、Hive、Hbase等分布式平台有一定的理解更好;
那么成为数据工程师吧,这里就是你的舞台。

● 熟悉一门数据处理语言,如SQL.....

北京-数据库管理员

所属部门: 百度

工作地点: 北京市

招聘人数: 若干

公 司: 百度

职位类别: 技术

发布时间: 2019-08-08

工作职责:

- 设计、开发百度MySQL分支, 包括MySQL引擎及新功能
- 设计和开发分布式数据库、数据库中间层及MySQL相关架构组件
- 研究数据库架构和数据库优化, 提高效率及系统稳定性、健壮性
- 参与数据库设计及SQL审核和优化

职责要求:

- 精通Linux环境下C/C++编程, 有数据库或其他存储系统开发基础
- 掌握数据库基本原理和知识, 熟悉SQL语法规则和特点
- 掌握基本的数据结构和算法知识
- 掌握基本的网络编程知识, 熟悉多线程编程及其技巧
- 有强烈的上进心和求知欲, 善于学习新事物, 对技术充满激情
- 具有较强的团队合作能力, 勇于面对挑战性问题

- 研究数据库架构和数据库优化, 提高效率及系统稳定性、健壮性;
- 参与数据库设计及SQL审核和优化
- 掌握数据库基本原理和知识, 熟悉SQL语法规则和特点



中国银行湖北省分行2019年 春季招聘和实习生招聘火热进行中

万千人中，你最行；
万千人中，只等你；
湖北中行，邀您前来，共谱精彩...

- 熟练掌握
Oracle、
SQL Server
工具

管理培训生（信科）

- 01 国内外知名院校计算机相关专业；
- 02 具有较强的实际动手能力和问题解决能力，熟练掌握Java、C、C++、Oracle、SQL Server等语言或工具（其中之一），有实际系统开发经验者优先；
- 03 具有较强的逻辑思维及表达沟通能力、良好的沟通协调及开拓创新能力；
- 04 身心健康、品行端正、无不良行为记录；
- 05 录用人员作为信息科技专业岗位的储备人员，在网点接受基础业务锻炼后，到各级管理机构信息科技等部门从事系统研发、运行维护、网络建设以及其他互联网金融相关工作。

学习方式

听课

（启发式、讨论式）

读书

（预习、复习）

实践

（上机练习）

考试成绩

● 平时成绩

（作业，上机，考勤等）

● 期末考试

（总评成绩=卷面成绩60%+平时成绩40%）

教学目标

通过本课程学习，使学生系统地掌握数据库系统的基本原理和基本技术。要求在掌握数据库系统基本概念的基础上，能熟练使用SQL语言在某一个数据库管理系统上进行数据库操作，掌握数据库设计、编程等相关知识。

第一章 绪论

● 内容概述:

阐述数据库的**基本概念**，介绍数据管理技术的进展情况、数据库技术产生和**发展的背景**，**数据模型**的基本概念、组成要素和主要的数据库模型，简要介绍了**概念模型**，数据库系统的**3级模式结构**以及数据库系统的组成。

● 重点:

掌握**关系数据模型**的相关概念、数据库系统**三级模式和两层映像**的体系结构、**数据库系统的逻辑独立性和物理独立性**等。

● 难点:

本章的难点是需要掌握数据库领域大量的基本概念。此外，**数据模型及数据库系统的体系结构**也是本章的难点。

第一章 绪论

1.1 数据库系统概述

1.2 数据模型

1.3 数据库系统结构

1.4 数据库系统的组成

1.5 小结

1.1 数据库系统概述

1.1.1 数据库的地位

1.1.2 四个基本概念

1.1.3 数据管理技术的产生与发展

数据库的地位

数据库技术产生于六十年代末，是数据管理的最新技术，是计算机科学的重要分支。

数据库技术是信息系统的核心和基础，它的出现极大地促进了计算机应用向各行各业的渗透。

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。

1.1 数据库系统概述

1.1.1 数据库的地位

1.1.2 四个基本概念

1.1.3 数据管理技术的产生与发展

1.1.2 四个基本概念

- 1) 数据(Data)
- 2) 数据库(Database)
- 3) 数据库管理系统(DBMS)
- 4) 数据库系统(DBS)

1) 数 据

数据(Data)是数据库中存储的基本对象

- 数据的定义
 - 描述事物的符号记录
- 数据的种类
 - 文字、图形、图像、声音
- 数据的特点
 - 数据与其语义是不可分的

数据举例

数据的含义称为数据的语义，数据与其语义是不可分的。

●例如 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：计算机系2013级学生人数

语义4：请同学给出.....

数据举例

□ 学生档案中的学生记录

（李明，男，199505，江苏南京市，计算机系，2013）

□ 数据的形式不能完全表达其内容

□ 数据的解释

- **语义：** 学生姓名、性别、出生年月、出生地、所在院系、入学时间
- **解释：** 李明是个大学生，1995年5月出生，江苏南京市人，2013年考入计算机系

请给出另一个解释和语义

2) 数据库

- 人们收集并抽取出一个应用所需要的大量数据之后，应将其保存起来以供进一步加工处理，进一步抽取有用信息。 

- 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、可共享的大量数据集合



2) 数据库

学生信息表

学号	姓名	年龄	性别	系名	年级
95004	王小明	19	女	社会学	95
95006	黄大鹏	20	男	商品学	95
95008	张文斌	18	女	法律学	95
....



2) 数据库 (续)

● 数据库的特征

- 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
- 冗余度较小
- 数据独立性较高
- 易扩展

3) 数据库管理系统

- 什么是数据库管理系统 (DBMS)

- 位于用户与操作系统之间的一层数据管理软件
- 是基础软件，是一个大型复杂的软件系统
- 例如：SQL SERVER、Oracle、MySQL、DB2.....

- 数据库管理系统的用途

- 科学地组织和存储数据、高效地获取和维护数据

DBMS的主要功能

- 数据定义功能

提供数据定义语言(DDL)

定义数据库中的数据对象

- 数据操纵功能:提供数据操纵语言(DML)

操纵数据实现对数据库的基本操作

(查询、插入、删除和修改)

DBMS的主要功能

●数据库的运行管理

保证数据的安全性、完整性

多用户对数据的并发使用

发生故障后的系统恢复

●数据库的建立和维护功能(实用程序)

数据库数据批量装载

数据库转储

介质故障恢复

数据库的重组织

性能监视等

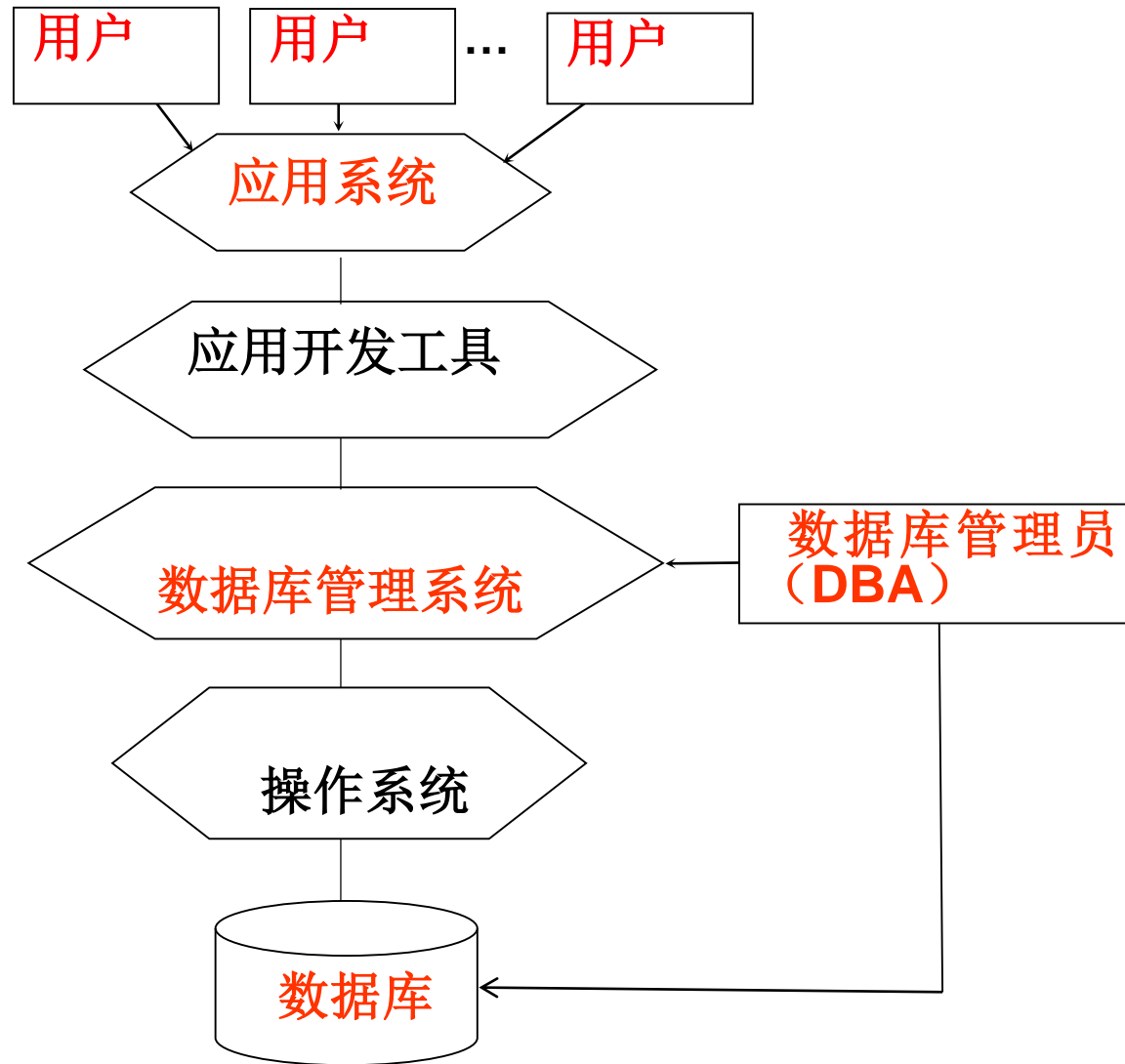
概念小结

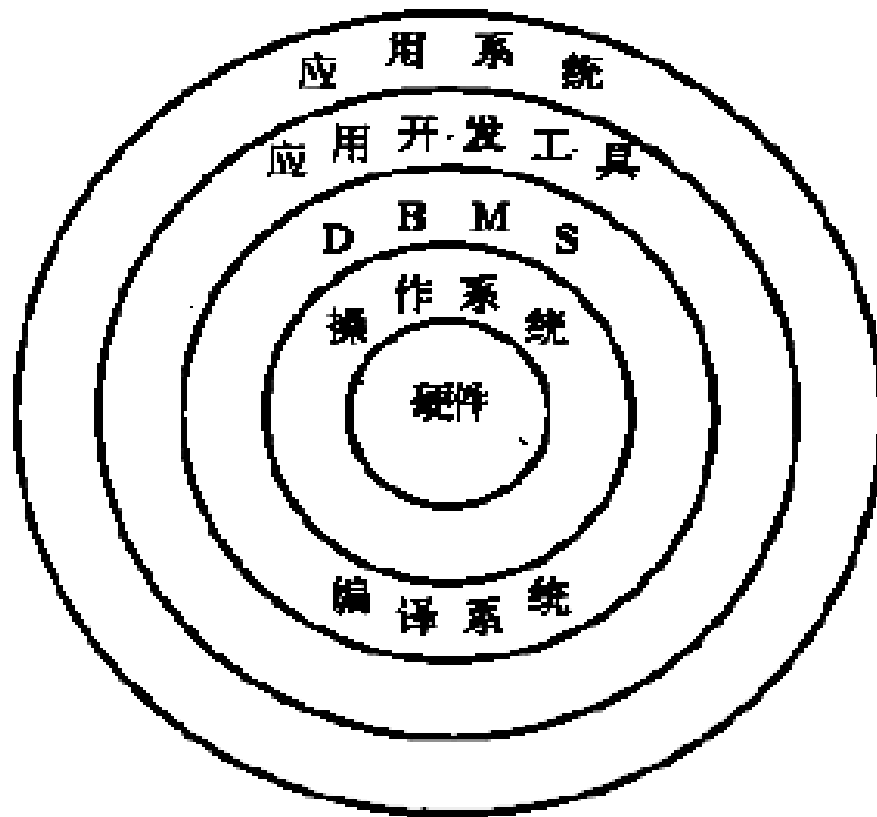
- **数据库**是长期存储在计算机内有组织的大量的共享的数据集合。可以供各种用户共享，具有最小冗余度和较高的数据独立性。
- **数据库管理系统**在数据库建立、运用和维护时对数据库进行统一控制，以保证数据的完整性、安全性，并在多用户同时使用数据库时进行并发控制，在发生故障后对数据库进行恢复。

4) 数据库系统

- 数据库系统（Database System，简称DBS）
- 数据库系统的构成（含有“人+物”）
 - 数据库
 - 数据库管理系统（及其应用开发工具）
 - 应用程序
 - 数据库管理员（人）
- 人们常常把数据库系统简称为数据库

4) 数据库系统（续）





数据库在计算机系统中的地位

注 意

分清日常所说的“数据库”到底是指哪个概念

数据库是SQL SERVER—— 数据库管理系统

数据库是xjgl.db—— 数据库

我们在开发数据库（学生管理系统）——
—— 数据库应用系统

1.1 数据库系统概述

1.1.1 数据库的地位

1.1.2 四个基本概念

1.1.3 数据管理技术的产生与发展

1.1.3 数据管理技术的产生和发展

●什么是数据管理

- 对数据进行分类、组织、编码、存储、检索和维护，是数据处理的中心问题

●数据管理技术的发展过程

- 人工管理阶段(40年代中--50年代中)
- 文件系统阶段(50年代末--60年代中)
- 数据库系统阶段(60年代末--现在)

数据管理技术的产生和发展(续)

数据管理技术的发展动力

- 应用需求的推动
- 计算机硬件的发展
- 计算机软件的发展

1) 人工管理

时期

- 40年代中--50年代中

产生的背景

- 应用需求科学计算
- 硬件水平无直接存取存储设备
- 软件水平没有操作系统
- 处理方式批处理

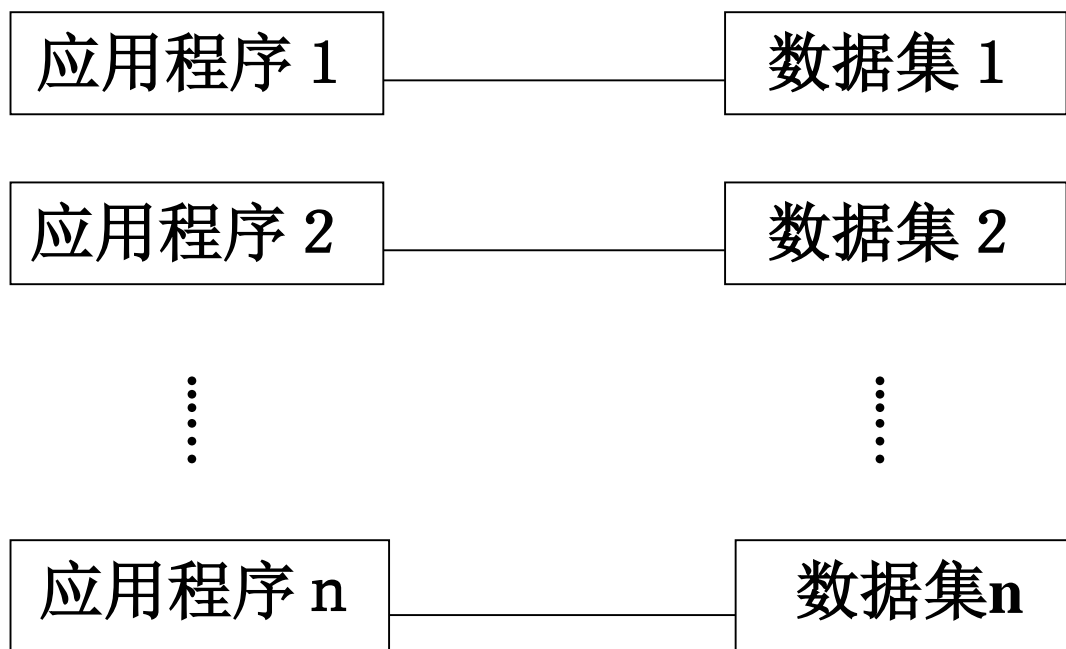
1) 人工管理（续）

特点

- 数据的管理者：应用程序，数据不保存。
- 数据面向的对象：某一应用程序
- 数据的共享程度：无共享、冗余度极大
- 数据的独立性：不独立，完全依赖于程序
- 数据的结构化：无结构
- 数据控制能力：应用程序自己控制



应用程序与数据的对应关系(人工管理)



2) 文件系统

时期

- 50年代末--60年代中

产生的背景

- 应用需求科学计算、管理
- 硬件水平磁盘、磁鼓
- 软件水平有文件系统
- 处理方式联机实时处理、批处理

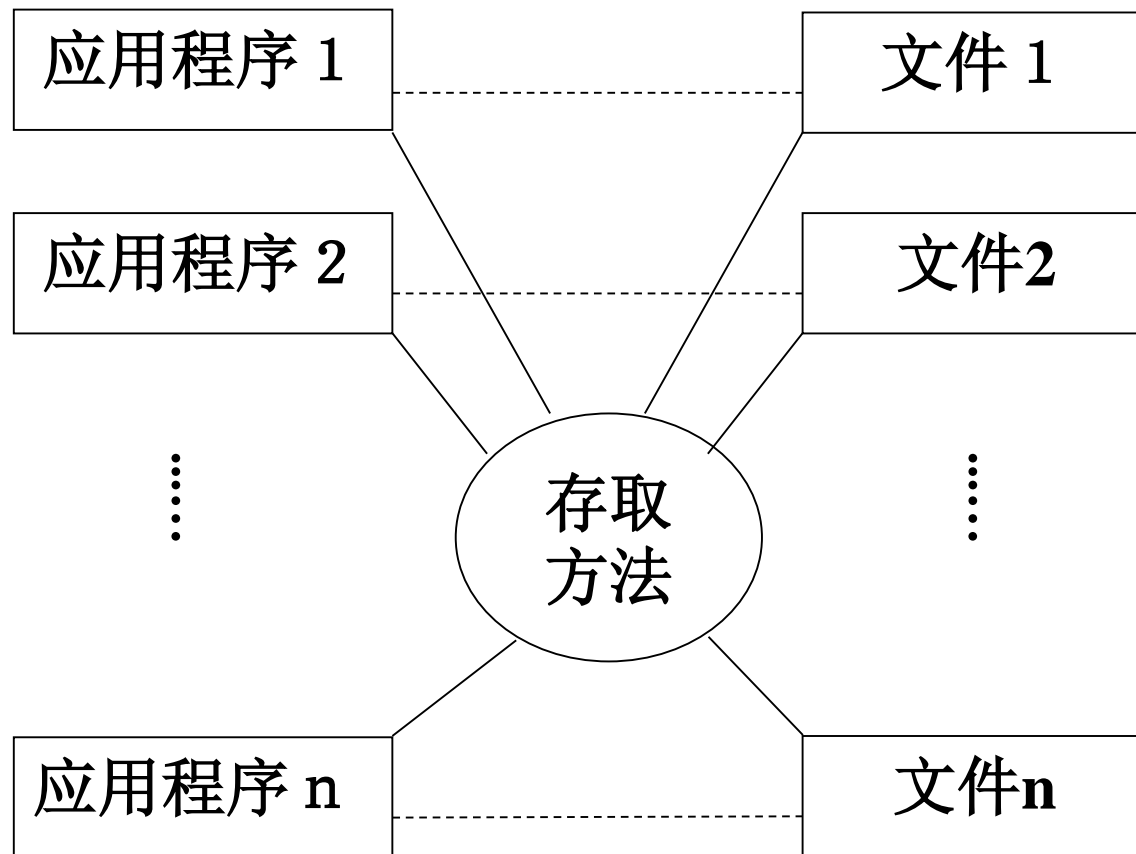
2) 文件系统 (续)

特点

- 数据的管理者：文件系统，数据可长期保存
- 数据面向的对象：某一应用程序▶
- 数据的共享程度：共享性差、冗余度大
- 数据的结构化：记录内有结构,整体无结构
- 数据的独立性：独立性差，数据的逻辑结构改变必须修改应用程序
- 数据控制能力：应用程序自己控制



应用程序与数据的对应关系(文件系统)



2) 数据库系统

时期

- 60年代末以来

产生的背景

- 应用背景大规模管理
- 硬件背景大容量磁盘
- 软件背景有数据库管理系统
- 处理方式联机实时处理，分布处理，批处理

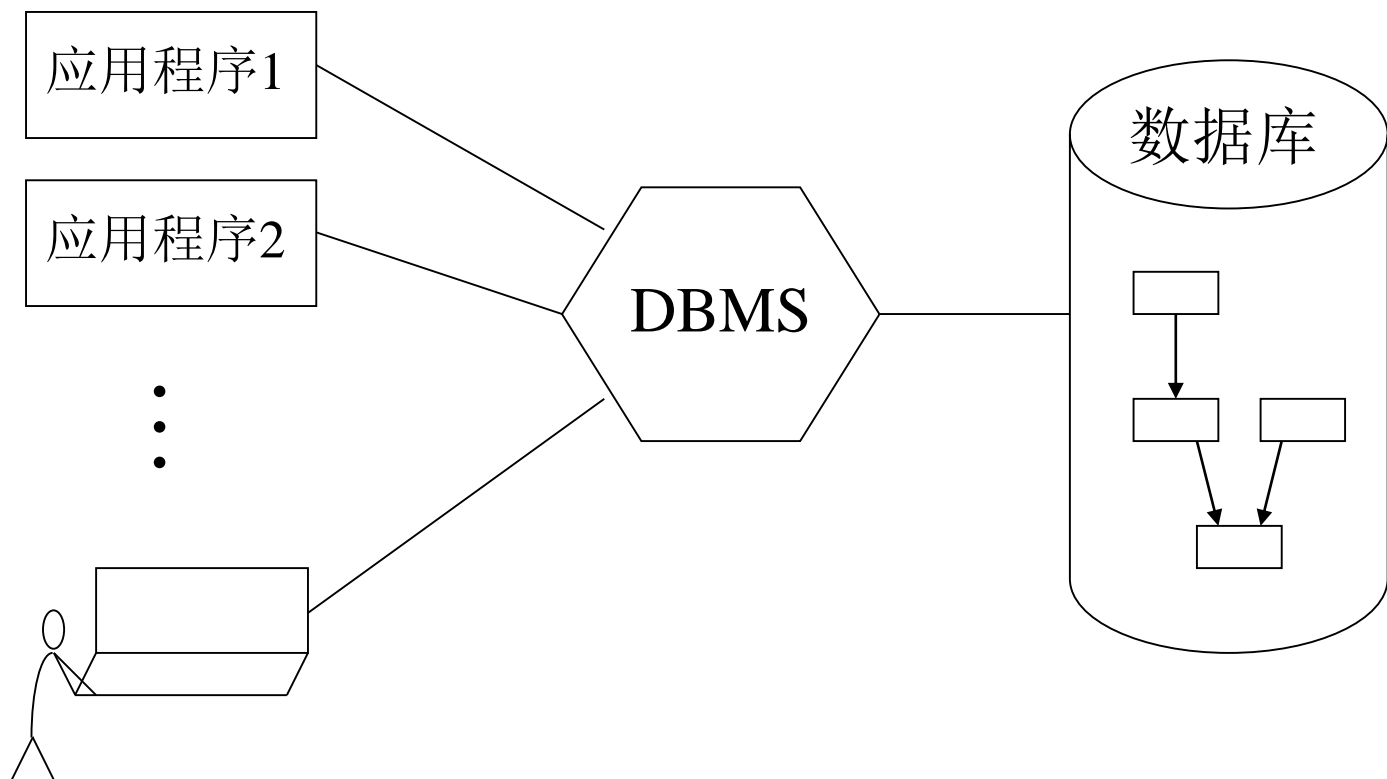
2) 数据库系统(续)

特点

- 数据的管理者：DBMS 
- 数据面向的对象：现实世界
- 数据的共享程度：共享性高 
- 数据的独立性：高度的物理独立性和一定的 逻辑独立性 
- 数据的结构化：整体结构化 
- 数据控制能力：由DBMS统一管理和控制 



应用程序与数据的对应关系(数据库系统)



数据的高共享性的好处

- 降低数据的冗余度，节省存储空间
- 避免数据间的不一致性
- 使系统易于扩充



数据独立性

●物理独立性

指用户的应用程序与存储在磁盘上的数据库中数据是相互独立的。当数据的物理存储改变了，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，用户程序也可以不变。



数据结构化

整体数据的结构化是数据库的主要特征之一。

文件系统和数据库系统本质区别之一：文件系统内部有结构，整体无结构，但是数据库系统中整体数据是结构化的。

数据库中实现的是数据的真正结构化

- 数据的结构用数据模型描述，无需程序定义和解释。
- 数据可以变长。
- 数据的最小存取单位是数据项。



DBMS对数据的控制功能

数据的安全性（Security）保护

- 使每个用户只能按指定方式使用和处理指定数据，保护数据以防止不合法的使用造成的数据的泄密和破坏。

数据的完整性（Integrity）检查

- 将数据控制在有效的范围内，或保证数据之间满足一定的关系。



DBMS对数据的控制功能

并发（Concurrency）控制

- 对多用户的并发操作加以控制和协调，防止相互干扰而得到错误的结果。

数据库恢复（Recovery）

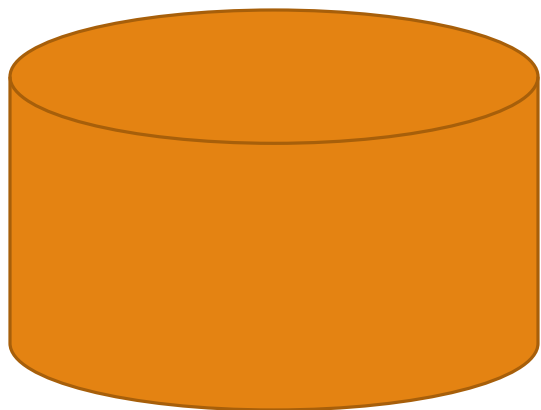
- 将数据库从错误状态恢复到某一已知的正确状态。



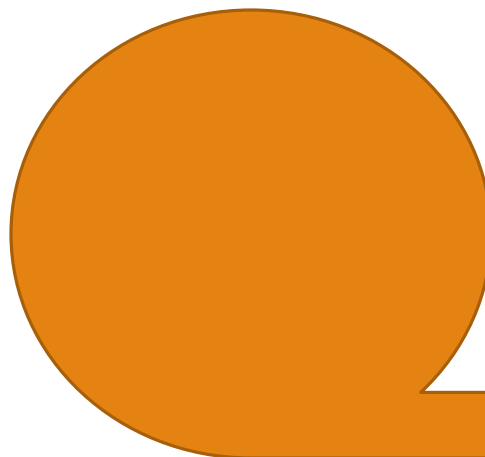
表 1.1 数据管理三个阶段的比较

		人工管理阶段	文件系统阶段	数据库系统阶段
背景	应用背景	科学计算	科学计算、数据管理	大规模数据管理
	硬件背景	无直接存取存储设备	磁盘、磁鼓	大容量磁盘、磁盘阵列
	软件背景	没有操作系统	有文件系统	有数据库管理系统
	处理方式	批处理	联机实时处理、批处理	联机实时处理、分布处理、批处理
特点	数据的管理者	用户（程序员）	文件系统	数据库管理系统
	数据面向的对象	某一应用程序	某一应用	现实世界（一个部门、企业、跨国组织等）
	数据的共享程度	无共享，冗余度极大	共享性差，冗余度大	共享性高，冗余度小
	数据的独立性	不独立，完全依赖于程序	独立性差	具有高度的物理独立性和一定的逻辑独立性
	数据的结构化	无结构	记录内有结构、整体无结构	整体结构化，用数据模型描述
	数据控制能力	应用程序自己控制	应用程序自己控制	由数据库管理系统提供数据安全性、完整性、并发控制和恢复能力

一个例子：数据库与文件 处理对比



VS



构建学生-获奖信息系统

- 描述：学号唯一；获奖最多5项

学号	姓名	性别	年龄	院系	获得奖项1	获得奖项2
20140101	刘晨	男	20	信息学院	优秀班干部	
20140102	王熙	女	22	信息学院	优秀班干部	三好学生

1.如何实现？

2.存在的问题？

奖项是变长信息，如何减少数据冗余？ 如何减少空值的存储空间？

如：拆分成两个表

学号	姓名	性别	年龄	院系
20140101	刘晨	男	20	信息学院
20140102	王熙	女	22	信息学院

学号	获得奖项
20140101	优秀班干部
20140102	优秀班干部
20140102	三好学生

文件操作——数据结构的定义

学生信息

```
30 /**
31  * 创建对应于学生关系的数据结构
32  */
33 typedef struct _Student
34 {
35     int      SNo;
36     char     SName[20];
37     char     SSex[1];
38     int      SAge;
39     char     SDept[20];
40 } Student;
```

获奖信息

```
#define MAX_AWARD_COUNT 5
/**
 * 创建对应于获奖关系的数据结构
 */
typedef struct _Award
{
    int      SNo;
    int      NumOfAwards;
    char     AName[MAX_AWARD_COUNT][20];
} Award;
```

文件操作——实现功能

插入操作

◦ 学生学籍

```
132  /**
133   * 插入一个新的学生学籍记录
134   */
135  void appendToStudent(Student *student)
136  {
137      if (student == NULL || base_table == NULL)
138          return;
139
140      fseek(base_table, 0L, SEEK_END);
141      fwrite(student, sizeof(Student), 1, base_table);
142      updateCountOfStudent();
143      fflush(base_table);
144  }
```


插入操作

◦ 学生获奖

```
200  /**
201   * 插入一个新的学生获奖记录
202   */
203  void appendToAward(Award *award)
204  {
205      if (award == NULL || award_table == NULL)
206          return;
207
208      fseek(award_table, 0L, SEEK_END);
209      fwrite(award, sizeof(Award), 1, award_table);
210      updateCountOfAwardedStudents();
211      fflush(award_table);
212  }
```

查询

◦ 学生学籍

```
146  /**
147   * 根据学号查询学生信息
148   */
149  void searchStudentWithSNo(Student *student, int sno)
150  {
151      int i, numOfStudents;
152
153      fseek(base_table, 0L, SEEK_SET);
154      if (feof(base_table))
155          return;
156
157      fread(&numOfStudents, sizeof(int), 1, base_table);
158      i = 0;
159      while (i < numOfStudents)
160      {
161          fread(student, sizeof(Student), 1, base_table);
162          if (student->SNo == sno)
163              return;
164
165          i++;
166      }
167      student->SNo = -1;
168  }
```

查询

◦ 学生获奖

```
215  /**
216   * 搜索获奖信息
217   */
218  void searchAwardWithSNo(Award *award, int sno)
219  {
220      int i, numOfStudents;
221
222      fseek(award_table, 0L, SEEK_SET);
223      fread(&numOfStudents, sizeof(int), 1, award_table);
224      i = 0;
225      while (i < numOfStudents)
226      {
227          fread(award, sizeof(Award), 1, award_table);
228          if (award->SNo == sno)
229              return;
230
231          i++;
232      }
233      award->NumOfAwards = -1;
234  }
```

(1) 上述例子基于文件操作，存在什么问题？

(2) 如果是要实现同样的功能，采用数据库阶段管理的思想进行开发，会怎样呢？

数据库操作——创建

学生表的创建

```
8 create table Student (  
9     SNo int primary key,  
10    SName char(20),  
11    SSex char(1),  
12    SAge int,  
13    SDept char(20)  
14 );
```

获奖表创建

```
15 create table Award (  
16     SNo int,  
17     AName char(40),  
18     primary key(Sno, AName),  
19     foreign key SNo references Student(SNo)  
20 );  
21 **/
```

学号	姓名	性别	年龄	院系
----	----	----	----	----

学号	获得奖项
----	------

数据库操作——功能实现

插入

- insert into student value ('20140101','刘晨','男',20,'信息学院');

学号	姓名	性别	年龄	院系
20140101	刘晨	男	20	信息学院

- insert into award values ('20140101','优秀班干部');

学号	获得奖项
20140101	优秀班干部

数据库操作——功能实现

查询

- 查询刘晨的学籍信息

```
select * from student where sname = '刘晨'
```

- 查询刘晨的获奖信息

```
select student.sno, sname, aname
```

```
from student, award
```

```
where student.sno = award.sno and sname = '刘晨'
```

对比总结

- 使用文件系统时，程序员要关注记录的结构和不同文件中记录之间的联系，编程工作量大。编程复杂，开发速度慢。
- 而数据库系统提供了功能强大的操作，如查询可能只需要一句SQL命令，程序员开发效率大大提高。