

CS33503: 数据库系统

邹兆年

哈尔滨工业大学
计算机科学与技术学院
海量数据计算研究中心
电子邮件: znzou@hit.edu.cn

2020年春

为什么要学习数据库系统?

- 当你得到一个数据库，并需要对它进行管理时，你需要了解数据库系统的基本概念，掌握数据库语言，具备数据库系统的使用技能。
- 当你面对一个数据密集型应用设计与开发需求时，你需要掌握数据库的设计方法，了解如何评估设计方案的优劣，具备数据库系统应用开发能力。
- 当你接手一个性能低下的数据密集型应用时，你需要了解数据库系统的工作原理，知道如何对系统进行优化和重新设计。
- 当你参与一种新型数据库管理系统的研发时，你需要了解多种数据库管理系统的工作原理和设计方案，并具备一定的研究能力。

课程安排

课堂授课

- 20次课，40学时
- 1-10周，周二1-2节、周四3-4节，致知43

实验

- 2次实验，8学时
- 6、10周，周日1-4节，格物213 (1703107-1703110班)、格物208 (1737301、1703401班)

教学团队

主讲教师：邹兆年 教授、博士生导师

- 办公室：科学园科创大厦K1417
- 个人主页：<http://homepage.hit.edu.cn/zou>
- 电子邮件：znzou@hit.edu.cn
- QQ: 12943596

助教：刘丹丹、杨婉

- QQ群：1044862981 (请将成员昵称修改为“姓名-学号”形式)

教材与参考书

- Ramez Elmasri, and Shamkant B. Navathe. Fundamentals of Database Systems (Fourth Edition)
- Hector Garcia-Molina, Jeffrey D. Ullman, and Jennifer Widom. Database System Implementation (Second Edition)
- Raghu Ramakrishnan and Johannes Gehrke. Database Management Systems (Third Edition)
- Abraham Silberschatz, Henry F. Korth, and S. Sudarshan. Database System Concepts (Sixth Edition)
- Jeffrey D. Ullman, and Jennifer Widom. A First Course in Database Systems (Third Edition)

教学大纲

第一部分: 数据库系统基础(10学时) // 如何使用数据库系统

- 第1章: 绪论(1学时)
- 第2章: 关系数据库(3学时)
- 第3章: 结构化查询语言(6学时)

第二部分: 数据库系统设计(10学时) // 如何设计开发数据库系统

- 第4章: 概念数据库设计(3学时)
- 第5章: 逻辑数据库设计(5学时)
- 第6章: 物理数据库设计(2学时)

第三部分: 数据库系统实现(20学时) // 数据库管理系统的原理

- 第7章: 存储管理(4学时)
- 第8章: 查询执行(4学时)
- 第9章: 查询优化(4学时)
- 第10章: 故障恢复(4学时)
- 第11章: 并发控制(4学时)

考核方式

- 考试: 占60%, 考试时间待定
- 实验: 占20%
- 作业: 占20%

第1章: 绪论

邹兆年

哈尔滨工业大学
计算机科学与技术学院
海量数据计算研究中心
电子邮件: znzou@hit.edu.cn

2020年春

教学内容¹

- ① 数据管理
- ② 数据库系统
- ③ 数据库系统的宝贵知识财富
 - ▶ 数据独立性
 - ▶ 数据库语言
 - ▶ 事务处理

¹课件更新于2020年2月24日

邹兆年 (CS@HIT)

第1章: 绪论

2020年春

9 / 42

1.1 Data Management

邹兆年 (CS@HIT)

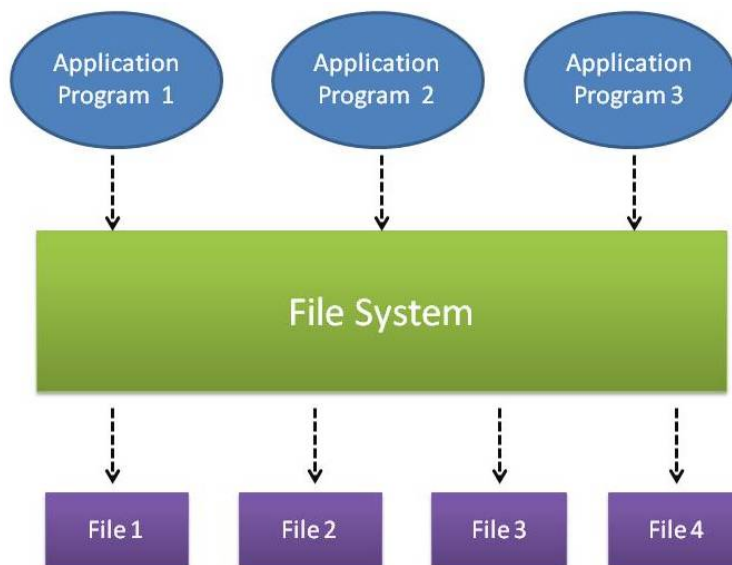
第1章: 绪论

2020年春

10 / 42

基于文件系统的数据管理方法

- 数据存储于文件中
- 数据由应用程序经过文件系统进行管理



基于文件系统的数据管理方法: 例子

文件:

student.txt

Alice	F	CS	001	90
Alice	F	CS	002	91
Bob	M	CS	001	60
Jack	M	MA	001	85
Jill	F	MA	002	80

course.txt

001	'Database Systems'
002	'Data Mining'

这些查询怎么做?

- 找出计算机系(CS)的所有学生
- 找出选修了001号课程的学生
- 找出选修了“Database Systems”课程的学生

方法:

- 编写应用程序
- 使用Linux shell

基于文件系统的数据管理方法的缺点

student.txt

Alice	F	CS	001	90
Alice	F	CS	002	91
Bob	M	CS	001	60
Jack	M	MA	001	85
Jill	F	MA	002	80

course.txt

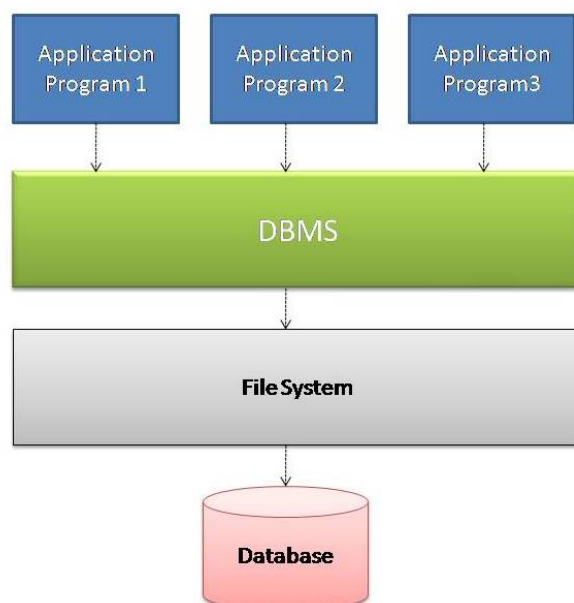
001	'Database Systems'
002	'Data Mining'

- 每当文件格式发生变化，就要修改应用程序
- 文件中存在冗余数据
- 文件修改可能造成数据不一致
- 文件修改可能破坏数据正确性
- 没有索引，数据访问效率低
- 只能对整个文件进行访问控制，数据安全性差
- 没有并发控制，多个应用程序同时读写文件可能产生冲突

在应用程序中解决上述问题。应用程序员：“我太难了！”

基于数据库管理系统的数据管理方法

- 数据存储于数据库(database)中
- 数据由应用程序经过数据库管理系统(database management systems, DBMS)管理



基于文件系统的方法 vs. 基于DBMS的方法

	基于文件系统	基于DBMS的方法
数据冗余度	高	低
数据一致性	No	Yes
数据正确性	No	Yes
索引	No	Yes
访问控制	No	Yes
并发控制	No	Yes
故障恢复	No	Yes

数据管理做什么?

- 数据定义(data definition): 定义数据的结构、类型及约束
- 数据存储(data storage): 存储和存取数据
- 数据操纵(data manipulation): 查询数据、更新数据(插入数据、修改数据、删除数据)
- 数据共享(data sharing): 事务管理(transaction management)、并发控制(concurrency control)、故障恢复(failure recovery)
- 数据控制(data control): 保证数据完整性(data integrity)、数据安全性(data security)
- 数据维护(data maintenance): 数据录入、数据转换、数据备份、数据恢复、性能监控

1.2 Database Systems

数据库(Database)、数据库管理系统(DBMS)

Definition (数据库)

A database (DB) is a set of related data that is **organized (组织)**, **shared (共享)**, and **persistent (持久化)**

Definition (数据库管理系统)

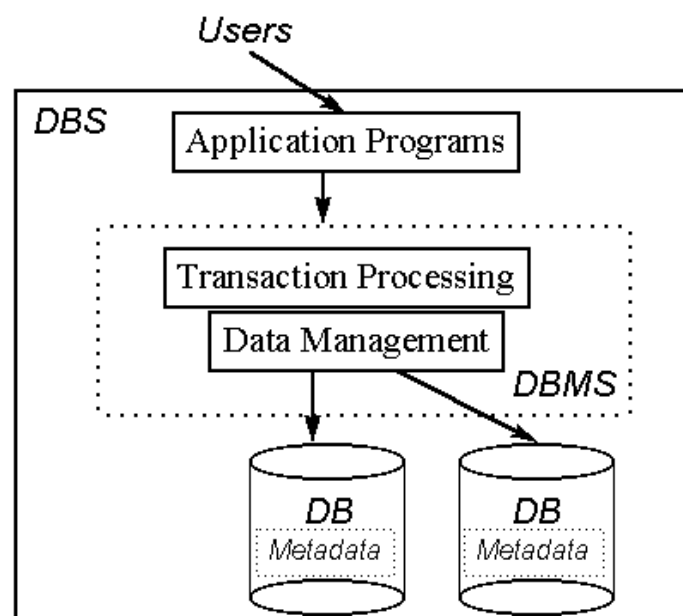
A database management system (DBMS) is a general-purpose system software that facilitates the **organization (组织)**, **storage (存储)**, **manipulation (操纵)**, **control (控制)**, and **maintainence (维护)** of databases among various users and application programs.

数据库用户(Database User)

- 数据库管理员(database administrator, DBA): 授权数据库访问, 协调和监控数据库使用, 获取软硬件资源, 控制资源的使用, 监控数据库性能
- 数据库设计者(database designer): 负责定义数据库的内容、结构、约束、存储过程和事务
- 终端用户(end user): 查询数据库, 生成报表, 部分终端用户还可以修改数据库内容

数据库系统(Database System)

数据库系统(database system, DBS)是由数据库、数据库管理系统、应用程序和数据库用户在一起构成的系统



1.3 Data Independence

数据抽象(Data Abstraction)

数据抽象(data abstraction)是将现实世界映射到计算机世界的过程

现实世界 \longrightarrow 信息世界 \longrightarrow 计算机世界

- 现实世界: 张三、李四、数学系、物理系、高等数学、大学物理...
- 信息世界: 实体、属性、联系、约束...
- 计算机世界: 记录、域、引用...

数据模型(Data Model)

数据模型(data model)是完成数据抽象的工具

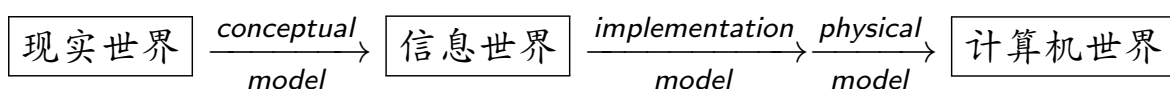
数据模型的三要素

- ① 用于描述数据库的结构的一系列概念
- ② 用于操纵数据结构的一系列操作
- ③ 数据库应当服从的约束条件

数据模型的分类

- 概念数据模型(conceptual data model): 该模型提供的概念最接近用户理解数据的方式, 用于将现实世界映射到信息世界(第4章讲)
- 物理数据模型(physical data model): 该模型提供的概念用于描述数据库在计算机中的存储细节(第7章讲)
- 实现数据模型(implementation data model): 该模型处在概念数据模型和物理数据模型之间, 在实现DBMS时使用
 - ▶ 层次数据模型
 - ▶ 网络数据模型
 - ▶ 关系数据模型(本课程学习的实现数据模型, 第2章讲)
 - ▶ 面向对象数据模型
 - ▶ XML数据模型
 - ▶ 文档数据模型
 - ▶ 图数据模型

在将信息世界映射到计算机世界时, 要用到实现数据模型和物理数据模型



数据库模式(Database Schema)

数据库模式(database schema)是对数据库的结构、类型、约束的描述

- 数据库模式是数据库的“类型声明”
- 数据库模式不经常变化

Student关系模式

Sno	Sname	Ssex	Sage	Sdept
-----	-------	------	------	-------

数据库实例(Database Instance)

数据库实例(database instance)是数据库在某一特定时间存储的数据

- 数据库实例是数据库的“值”
- 每当数据库被更新，数据库实例就发生变化

Student关系实例

Sno	Sname	Ssex	Sage	Sdept
PH-001	Nick	M	20	Physics
CS-001	Elsa	F	19	CS
CS-002	Ed	M	19	CS
MA-001	Abby	F	18	Math
MA-002	Cindy	F	19	Math

数据库的三层模式结构(Three-Schema Architecture)

数据库不是只用一种模式来描述的，数据库模式分三个层次来定义

内模式(Internal Schema)/存储模式(Storage Schema)

- 描述数据库的物理存储结构和存取方法
- 数据库只有一个内模式
- 定义内模式时通常使用物理数据模型提供的概念

概念模式(Conceptual Schema)

- 为全体数据库用户描述整个数据库的结构和约束
- 数据库只有一个概念模式
- 定义概念模式时使用实现数据模型提供的概念

外模式(External Schema)/视图(View)

- 从不同类别用户的视角描述数据库的结构
- 数据库可以有多个外模式
- 定义外模式时也使用实现数据模型提供的概念

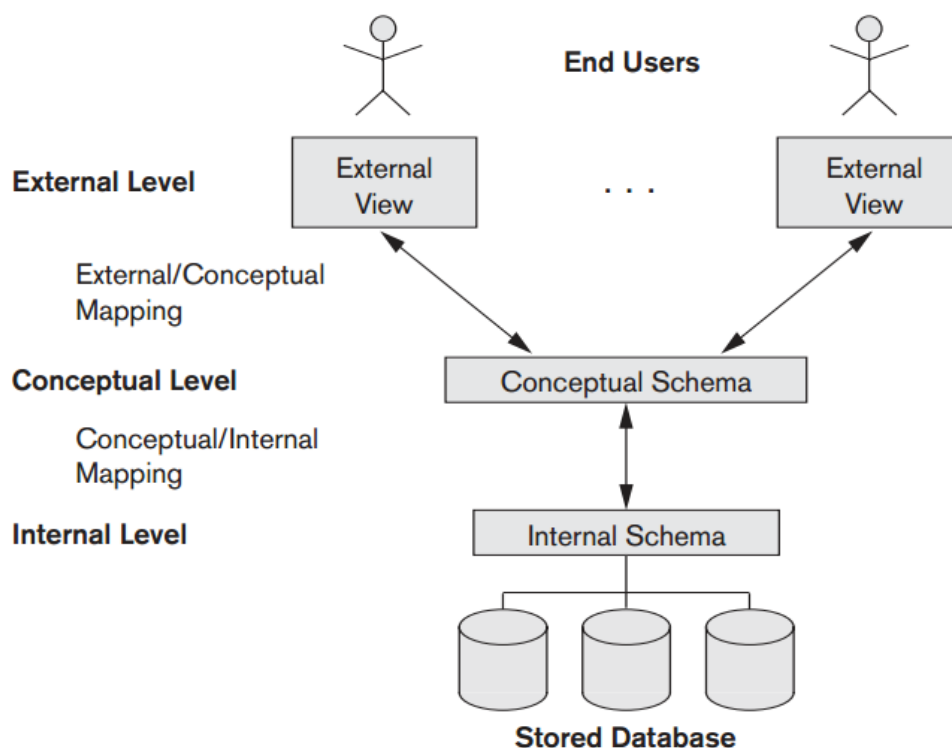
邹兆年 (CS@HIT)

第1章: 绪论

2020年春

29 / 42

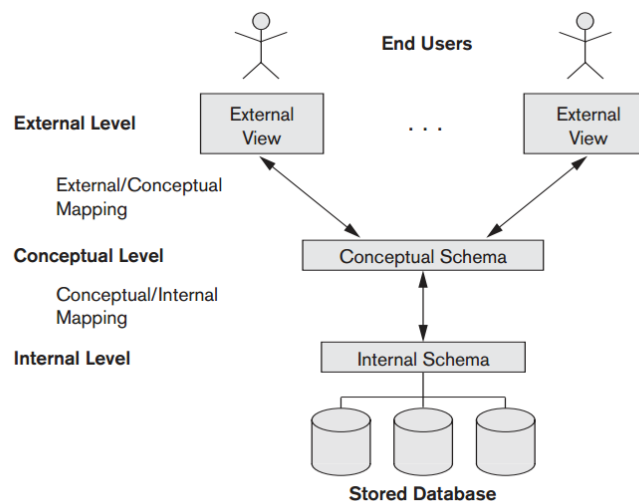
数据库的三层模式结构(Three-Schema Architecture)



模式映射(Schema Mapping)

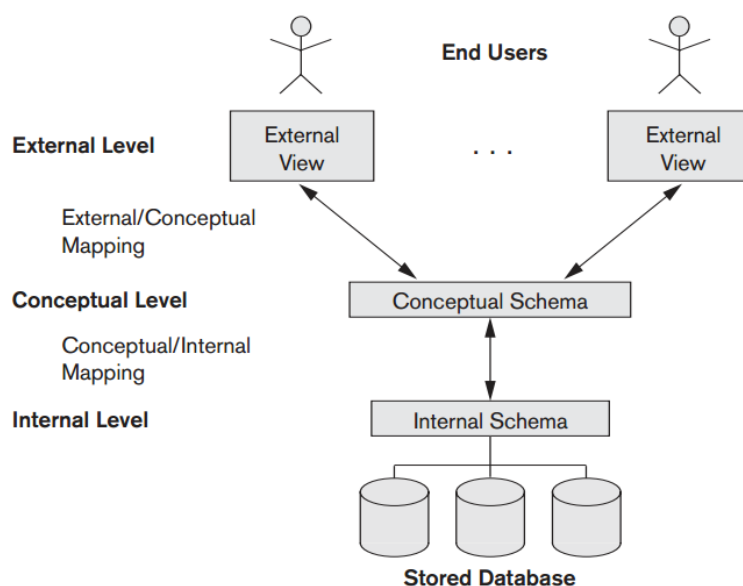
在三层模式结构中，不同层次模式间的映射用于完成应用程序与数据库之间的数据转换(data transformation)和请求转换(request transformation)

- **请求转换**: 应用程序是依据外模式开发的，应用程序在外模式上声明的数据请求通过模式映射转换为DBMS在内模式上的请求
- **数据请求**: 数据库的物理存储是按照内模式来组织的，DBMS检索到的数据通过模式映射转换为符合外模式的组织形式，返回给应用



模式映射的分类

- **外模式-概念模式映射(external/conceptual mapping)**: 从一个外模式到概念模式的映射
- **概念模式-内模式映射(conceptual/internal mapping)**: 从概念模式到内模式的映射



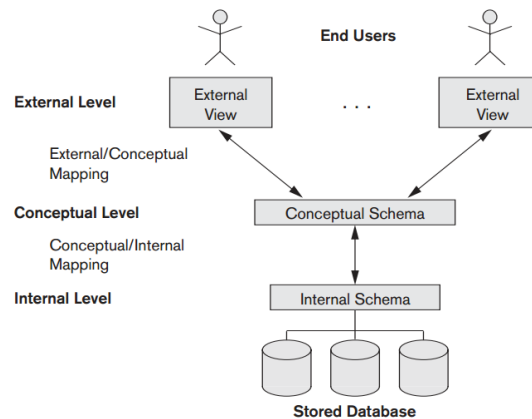
数据独立性(Data Independence)

逻辑数据独立性(logical data independence)

- 当概念模式发生改变时，只需修改外模式到概念模式的映射
- 外模式无需改变，依据外模式开发的应用程序也无需改变

物理数据独立性(physical data independence)

- 当内模式发生改变时，只需修改概念模式到内模式模式的映射
- 概念模式和外模式均无需改变，依据外模式开发的应用程序也无需改变



1.4 Database Languages

数据库语言(Database Language)

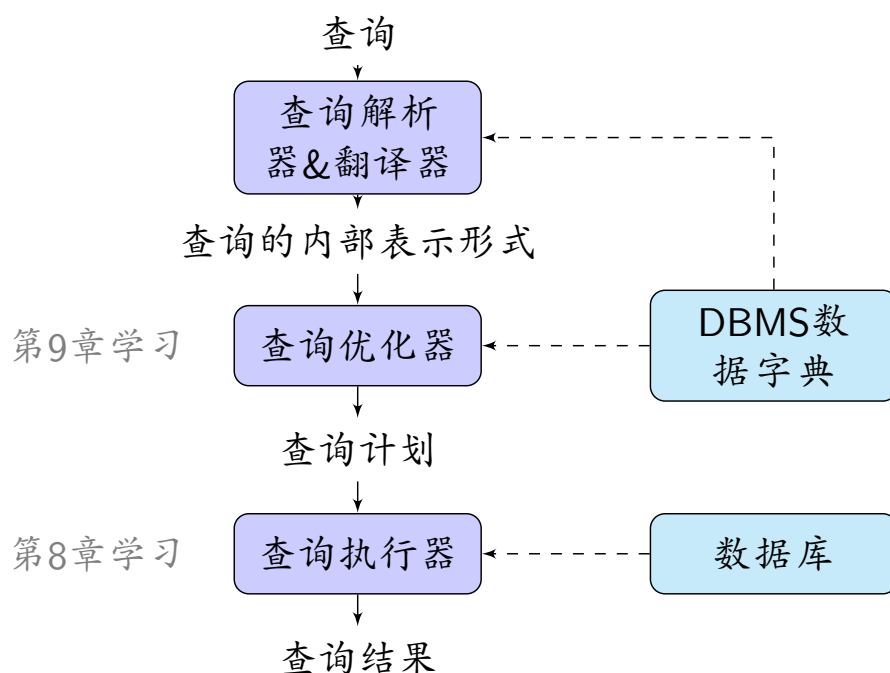
数据库语言(database language)是用户/应用程序与DBMS交互时所使用的语言

- 数据定义语言(data definition languages, DDL): DBA和数据库设计者用来声明数据库模式的语言
- 数据操纵语言(data manipulation languages, DML): 查询和更新数据库时所使用语言

第3章学习关系数据库结构化查询语言SQL

数据库查询(Database Queries)

- DML通常是描述式的(descriptive), 用它编写的数据库查询只描述查询意图, 而不指明查询执行过程
- DBMS自动生成最优的查询计划, 然后在数据库上执行查询计划



1.5 Transaction Processing

事务(Transaction)

事务(transaction)是由数据库上的一系列操作完成的复杂任务，这些操作要么全执行，要么全不执行

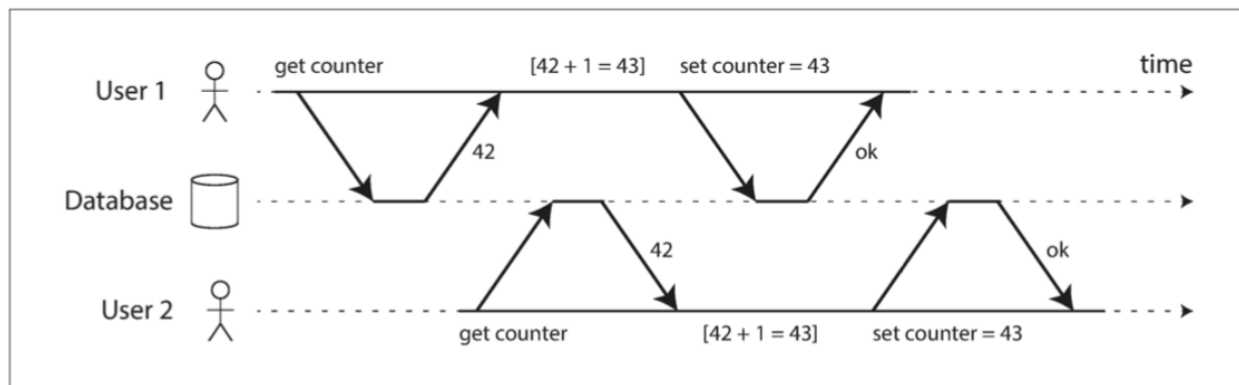
- 银行转帐
- 在线购物
- 会议室预定

事务的性质

- 原子性(atomicity)
- 一致性(consistency)
- 隔离性(isolation)
- 持久性(durability)

并发控制(Concurrency Control)

- 为了充分利用数据库系统，允许多个事务在数据库上并发执行
- 多个事务并发执行可能会破坏数据库的一致性



并发控制([concurrency control](#))确保多个事务并发执行不会破坏数据库的一致性(第11章学习)

故障恢复(Failure Recovery)

- 计算机软硬件系统随时可能发生故障
- 故障可能在事务执行过程中发生，从而破坏数据库的一致性
 - ▶ 例: 转账过程中系统发生故障
- 故障恢复([failure recovery](#))确保系统重启后数据库可以恢复到最近的一致性状态(第10章学习)

总结

- ① 数据管理
- ② 数据库系统
- ③ 数据库系统的宝贵知识财富
 - ▶ 数据独立性
 - ▶ 数据库语言
 - ▶ 事务处理

教学大纲

第一部分: 数据库系统基础(10学时) // 如何使用数据库系统

- 第1章: 绪论(1学时)
- 第2章: 关系数据库(3学时)
- 第3章: 结构化查询语言(6学时)

第二部分: 数据库系统设计(10学时) // 如何设计开发数据库系统

- 第4章: 概念数据库设计(3学时)
- 第5章: 逻辑数据库设计(5学时)
- 第6章: 物理数据库设计(2学时)

第三部分: 数据库系统实现(20学时) // 数据库管理系统的原理

- 第7章: 存储管理(4学时)
- 第8章: 查询执行(4学时)
- 第9章: 查询优化(4学时)
- 第10章: 故障恢复(4学时)
- 第11章: 并发控制(4学时)