Week17_Course

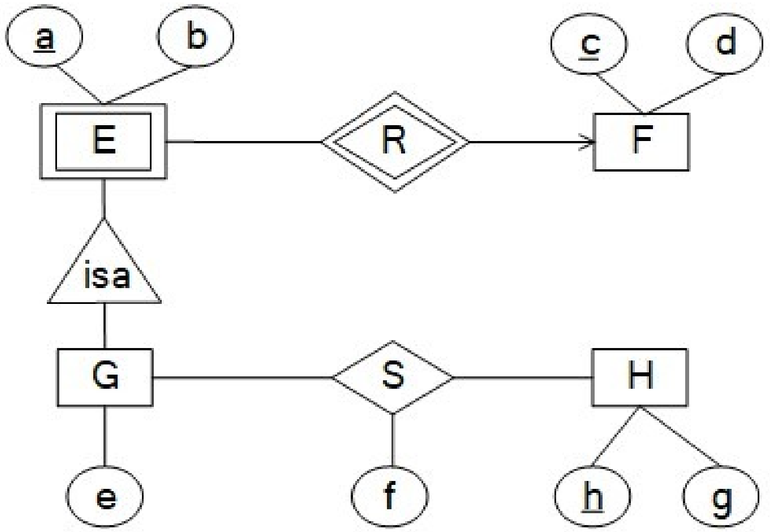
Database System Summary

Outline – 重点掌握的内容

- Introduction to DBMSs （重点知晓数据库系统的三层模式结构和两层映像的功能）
- Principles of Data Layout and Index （本章不考核，SimpleDB实现涉及该部分内容）
- The Entity Relationship model
- E/R to Relational Transformation （ER转化为关系模型、SQL操作，主外键约束）
- SQL: Introduction
- SQL: Advanced
- DB Schema Design （考核所有知识点）
- Relation Algebra & Query Processing （关系代数基本操作，查询处理细节不考核）
- Crash Recovery （考核所有知识点）
- Concurrency Control （掌握优先图、两段锁等并发思想，Timestamp & Validation 不考核）

概念模型设计与关系模型转换

应该掌握ER图中实体、属性、联系、弱实体集和子类的概念，可以将ER图转为满足3NF的关系模式，并深刻理解主键和外键的约束关系



关系表名	表中含有属性	主键	外键	参照表名及属性

Transform the E-R diagram into relation model (in 3NF)

Write SQL statements that create the tables （primary and foreign keys）

referential integrity :

On insert(S) -> exists(H) and exist(G);

On delete(F) -> delete(E) or not allowed)

设计期末工程作业，选定某个信息系统应用背景，完成如下内容：

1. 调研某一应用领域，给出该应用领域的详细需求描述；

2. 采用教材中介绍的方法实现如下设计：

a) 画出该领域的概念模型ER图（至少有五个以上的实体，含有子类的形式，注意一定标明每个实体的主键）；

b) 请按课堂上讲授的ER图转换成关系模式的方法，将上述ER图转换成关系模式，并标明每个关系的主键属性和外键属性；

c) 用SQL语句创建上述关系模式。

d) 给出该数据库模式上5个查询语句样例，分别为：单表查询、多表连接查询、多表嵌套查询和EXISTS查询和聚合操作查询。

3. 使用PowerDesigner工具实现如下设计：

a) 画出该领域的概念模型ER图，给出ER图截图；

b) 使用PowerDesigner工具，将上述ER图转为关系模型图，给出关系模型图截图；

c) 使用PowerDesigner工具，生成创建数据库的SQL语句。

4. 分析比较采用上述两种方法

a) 两种关系模式的设计是否存在差异？如有差异，这种差异是否对后期的实现带来不同的影响？

b) PowerDesigner工具生成的SQL语句有什么样的特点？ 为什么会出现一些附加语句？它的作用是什么？

关系代数和SQL语言操作

- 熟练使用关系代数和SQL语言操作数据库
  - 根据SQL语句写出运行结果 Please write each result of the following queries
  - 使用关系代数基本运算实现用户查询（重点掌握9种运算符）
  - 能够使用SQL语句实现用户查询

关系模式设计Normal Form

- 重点掌握3NF、BCNF，掌握保持无损连接和函数依赖的模式分解的算法
  - FD and attribute closure
  - keys and superkeys
  - normal form(3NF,BCNF,4NF)

作业问题讲解

- 难点：函数依赖的分解，一定要在函数依赖的闭包集合上进行分解

例题：R(A, B, C)，F={A→B，B→C},

F的闭包中含有A→C，F逻辑蕴含A→C，

分解两个关系模式：R1(A,B), R2(A,C)，R2上存在函数依赖A→C，千万不能丢

作业：设R(A, B, C, D)，F={A→B，B→C，C→D，D→A}，R的候选码有：_____。对分解r = {R1 (A , B), R2 (B , C), R3 (C , D) }，它_____（是/不是）无损连接，____（保持/不保持）函数依赖。

正确答案：B^+={C,D,A,B}，F逻辑蕴含B→A，同理F逻辑蕴含C→B，D→C

R1 (A , B)，F1={A→B, B→A}；R2 (B,C)，F2={B→C, C→B}；R3 (C , D)，F3={C→D, D→C}

保持函数依赖，保持无损连接

- 作业第8题

函数依赖：　（商店编号，商品编号）->部门编号；
（商店编号，部门编号）->负责人；
（商店编号，商品编号）->库存数量

候选码：（商店编号，商品编号）

主属性：商店编号，商品编号；　非主属性：部门编号，负责人，库存数量

没达到3NF：对于（商店编号，部门编号）->负责人 ，左边不是superkey，右边不是主属性

达到2NF：（商店编号)+={ }；(商品编号)+={ }

商店编号和商品编号单独不能推出任何属性，不存在非主属性对候选码的部分依赖

关系模式设计重要的知识点

Consider a relation R = (……) with FD's ……

- What is the attribute closure of ***?
- Of the following FDs, circle the ones that are implied by the functional dependencies given above.
- List all keys for R.
- Write down two functional dependencies that causes this relation to violate NF.
- We decompose R into R_1(……) and R_2……). What are the keys of R_1? What are the keys of R_2?
- Decompose R into two or more relations that are all in 3NF. And make sure your decomposition is (i) dependency preserving, and (ii) lossless join.

数据库故障恢复

- 掌握日志的记录方式：What are the all of the possible values on disk for each of the database elements A, B and C?
- 掌握日志的恢复过程：Which, if any, transactions will need to be redone and undone in the recovery process?
- 知道日志恢复后数据库的正确状态：If finished the system recovery, what are the values on disk for each of the database elements A, B and C

单选题 1分

课堂练习

Consider a system that uses undo-redo logging. After a system crash, we find the following log entries on disk:

<START T1>
<T1,A,5,0>
<START T2>
<T1,B,1,2>
<COMMIT T1>
<T2,B,2,3>
<START CKPT(T2)>
<T2,C,8,9>
<COMMIT T2>
<END CKPT>
<START T3>
<T3,A,0,10>
<START T4>
<T4,A,10,11>
<T3,C,9,7>
<T4,B,3,22>

What are the all of the possible values on disk for each of the database elements A?

A 5,0,10,11

B 0,10,11

C 10,11

D 11

并发控制

- 若干事务并发运行，能够判断所给的调度是不是可串行化调度（优先图）
- 如果已知几个事务，如何添加合适的锁，保证是可串行化调度（两段锁协议 多粒度锁）
- 了解死锁是怎样形成的

作业练习

For each of the following schedules, answer the questions below:

Sb= W3(A) R1(B) R2(A) R4(D) W1(B) R4(A) W2(B) R1(D) R4(B)

(a) What is the precedence graph for the schedule Sb?

(b) Is the schedule conflict serializable? If so, show all equivalent serial transaction orders. If not, describe why not.

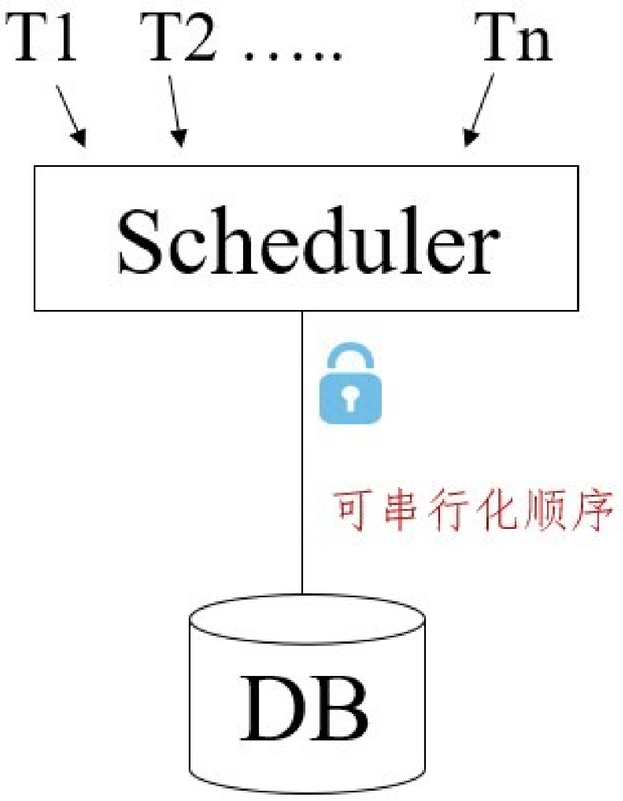
主观题 10分

课堂练习

Consider the following two transactions:

T1 = R1(C) R1(A) R1(B) W1(A) W1(C);

T2 = R2(B) W2(A) R2(A) R2(C) W2(C);



（a）请添加合适的加锁和解锁命令，使事务T1和T2在并发运行时可以保证数据库的一致性；

（b）请说明这两个事务会引起死锁吗？如果会引起死锁，请给出死锁的示例；如果不会引起死锁，请说明为什么？

期末考试重点

- Data Models（ER转关系，SQL DDL，主外键约束）
- Join query semantics （写SQL查询结果）
- Normal Form （候选码，3NF，BCNF，FD，无损连接）
- Relational Algebra and SQL Queries （关系代数、SQL查询）
- Concurrency Control （优先图，冲突可串行化调度，两段锁）
- Transaction Management （undo/redo logging）
- 简答题（数据库架构，SQL授权，意向锁，多值依赖，逻辑优化等）

Why-What-How

问题引出与剖析 问题的解决与思路

Entity–Relationship Model

- Why do database systems need modeling tools ?
- What is the core technique of conceptual model?
- How to design conceptual model of DBs ?

Schema Design and Refinement

- Why to design and refine DB schema?
- What is the key points of schema design ?
- How to design DB schema?

Data Layout - Index

- Why database system needs index technology?
- What is the core technique of index ?
- How to realize index of database system?

Database System Crash Recovery

- Why database system needs crash recovery technology?
- What is the core technique of crash recovery?
- How to realize crash recovery of database system?

Key Questions We Will Answer

- How can we **collect and store** large amounts of data?
  - By building tools and data structures to efficiently index and serve data
- How can we **efficiently query** data?
  - By compiling high-level declarative queries into efficient low-level plans
- How can we **safely update** data?
  - By managing concurrent access to state as it is read and written

The world is increasingly **driven by data...**