

Transaction Management

重要程度: ★★★★
难易程度: ★★★★

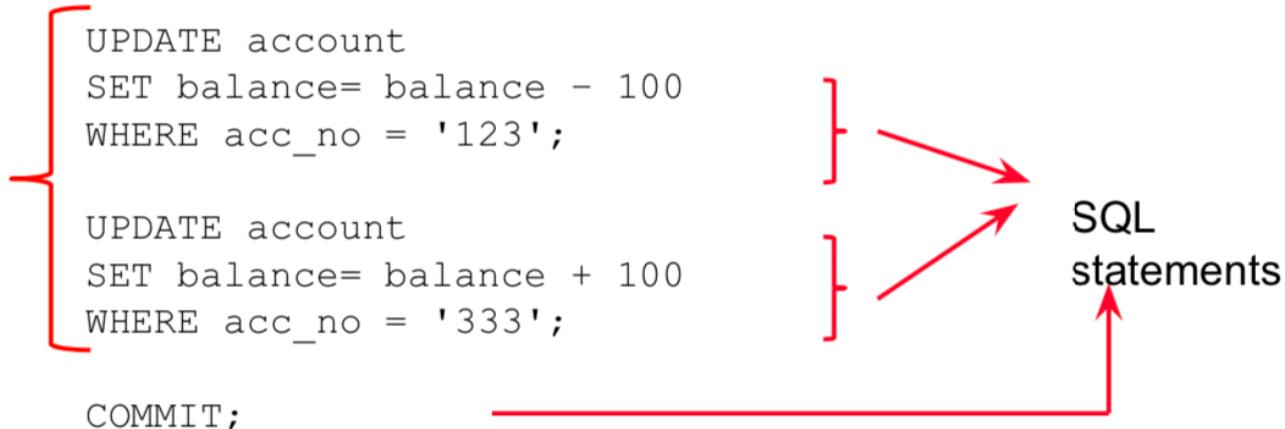
Sam is transferring \$100 from his bank account to his friend Jim's.

- Sam's account should be reduced by 100.
- Jim's account should be increased by 100.

TRANSACTION

```
UPDATE account  
SET balance= balance - 100  
WHERE acc_no = '123';  
  
UPDATE account  
SET balance= balance + 100  
WHERE acc_no = '333';  
  
COMMIT;
```

SQL statements



Transaction Management

重要程度: ★★★★
难易程度: ★★★★

- **Atomicity 原子性**
 - All database operations of a transaction must be entirely completed or entirely aborted
 - transaction的操作要么都发生，要么全都不发生，如果发生错误，会rollback
- **Consistency 一致性**
 - It must take the database from one consistent table to another
 - 系统从一个完整的状态转换到另一个完整的状态，不能破坏数据的完整性
 - e.g. 在某次AB转账中，A转出100，B收入50，破坏了完整性
- **Isolation 独立性**
 - It must not interfere with other concurrent transactions
 - Data used during execution of a transaction cannot be used by a second transaction until the first one is completed
 - 多个transaction并行访问时，每个transaction之间都是独立不受影响的
 - 不同的transaction利用同一组数据进行操作时，每个transaction都有完整的数据空间
- **Durability 持久性**
 - Once completed the changes the transaction made to the data are durable, even in the event of system failure
 - transaction 结束后，更改的数据会持久保存在database中

Transaction Management

重要程度: ★★★★
难易程度: ★★★★

Time: ↓

T0	T1
Read(X) X=X+1 Write(x)	Read(Y) Y=Y*2 Write(Y) Read(x) X=X+2 Write(X)

Serial

- ❖ 每个时刻只有一个transaction能对这组数据进行操作

T0	T1
Read(X) X=X+1 Write(x)	Read(Y) Y=Y*2 Write(Y) Read(x) X=X+2 Write(X)

Interleaved (non Serial)

- ❖ 多个transaction交叉运行

Transaction Management

重要程度: ★★★★
难易程度: ★★★★

❖ Lost Update

时间	转账事务A	取款事务B
T1		开始事务
T2	开始事务	
T3		查询账户余额为1000元
T4	查询账户余额为1000元	
T5		取出100元把余额改为900元
T6		提交事务
T7	汇入100元	
T8	提交事务	
T9		把余额改为1100 元 (丢失更新)

Transaction B 未能及时更新，被Transaction A结果覆盖

HD@朱昱臻-412729939

Transaction Management

重要程度: ★★★★
难易程度: ★★★★

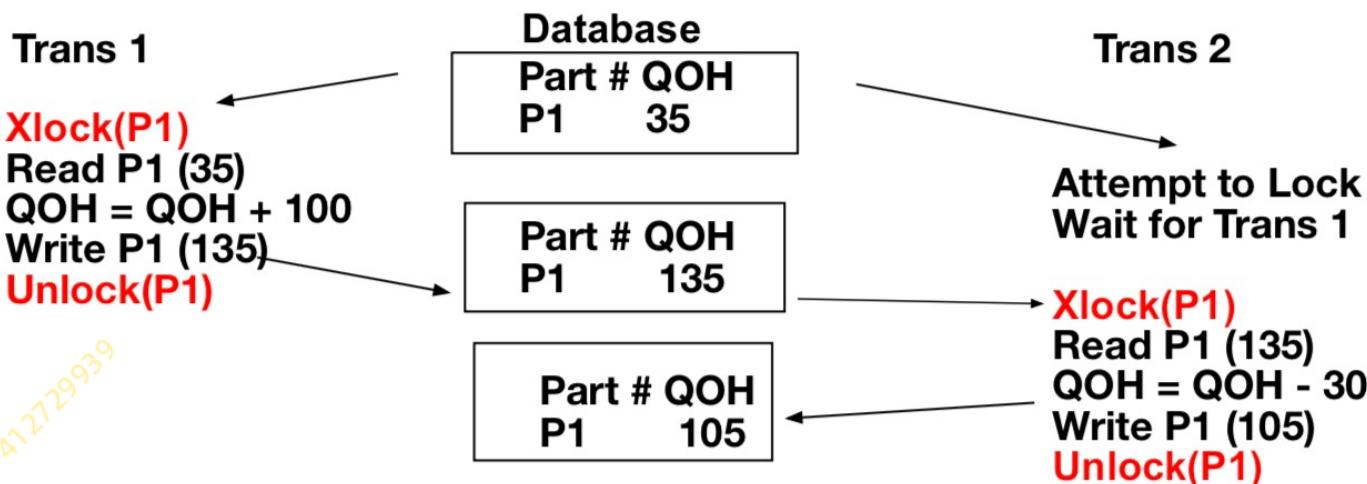
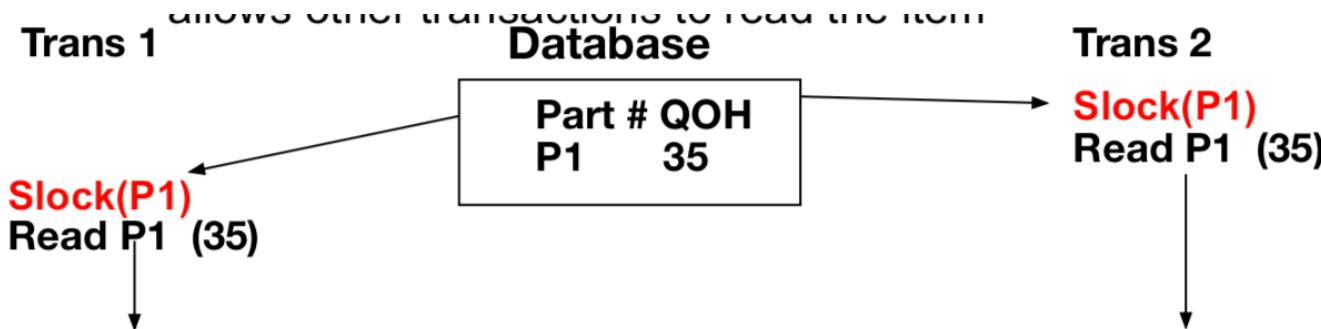
❖ Lock

- A lock is an indicator that some part of the database is temporarily unavailable for update
- Level
 - Database Level
 - Table level
 - Page level
 - Record level (allow access different rows)
 - Attribute level (allow access different attributes in a row)
- Type
 - **Shared Lock**
 - Allows other transaction to read item
 - T1对A上S锁, T1只可以read不能write, 其他T也能对A上S锁
 - 保证了其他transaction也能access A, 但在T1释放前, 都不能write
 - **Exclusive Lock**
 - Required an exclusive lock
 - A single transaction exclusively holds the lock on the item
 - T1对A上x锁, T1可以read&write, 其他T不能进行任何操作

HD@朱昱臻-412729939

Transaction Management

重要程度: ★★★★
难易程度: ★★★★



HD@朱昱臻-412729939

Transaction Management

重要程度: 
难易程度: 

Deadlock

- T1 has an exclusive lock on data item A, and requests a lock on data item B
- T2 has an exclusive lock on data item B, and requests a lock on data item A
- T1访问表A（锁住A）A然后又访问B, T2访问B(锁住B),然后去访问A

Deadlock prevention

- A transaction must acquire all the locks it requires before it updates any record - If it cannot acquire a necessary lock, it releases all locks, and tries again later
- Deadlock detection and recovery
- Detection involves having the Lock Manager search the Wait-for tables for lock cycles
- Resolutions involves having the Lock Manager force one of the transactions to abort, thus releasing all its locks

Transaction Management

重要程度: ★★★★
难易程度: ★★★★

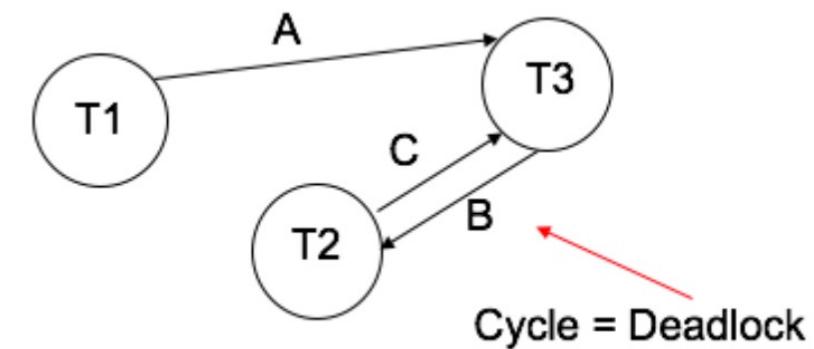
TIME	TX	ACCESS	A	B	C
0	(T1)	READ A			
1	(T2)	READ B			
2	(T3)	READ A			
3	(T1)	UPDATE A			
4	(T3)	READ C			
5	(T2)	READ C			
6	(T2)	UPDATE B			
7	(T2)	READ A			
8	(T2)	UPDATE C			
9	(T3)	READ B			

HD@朱昱臻-412729039

Transaction Management

重要程度: ★★★★
难易程度: ★★★★

TIME	TX	ACCESS	A	B	C
0	(T1)	READ A	S(T1)		
1	(T2)	READ B		S(T2)	
2	(T3)	READ A	S(T3)		
3	(T1)	UPDATE A	Wait(T3)		
4	(T3)	READ C			S(T3)
5	(T2)	READ C			S(T2)
6	(T2)	UPDATE B		X(T2)	
7	(T2)	READ A	S(T2)		
8	(T2)	UPDATE C			Wait(T3)
9	(T3)	READ B		Wait(T2)	



Transaction Management

重要程度: ★★★★
难易程度: ★★★★

Q6. Transaction Management (5 + 5 = 10 marks)

- a. Given two transactions:

T1 – R(X), W(X)

T2 – R(Y), W(Y), R(X), W (X)

Where R(X) means Read(X) and W(X) means Write(X).

- i. If we wish to complete both of these transactions, explain the difference between a *serial* and *non-serial* ordering of these two transactions. Provide an example of each as part of your answer.
- ii. What transaction ACID property does a non-serial ordering of these two transactions potentially violate.

[4+ 1 = 5 marks]

Transaction Management

重要程度: ★★★★
难易程度: ★★★★

i.

Serial – all of one transaction followed by all of the other

T1 R(X), T1 W(X), T2 R(Y), T2 W(Y), T2 R(X), T2 W(X)

Non-Serial – interleaving of the transactions

T1 R(X), T2 R(Y), T2 W(Y), T1 W(X), T2 R(X), T2 W(X)

ii.

Isolation or Consistency

重难点总结

重难点总结

1、Transaction Management

- Atomicity
- 要么全发生，要么全不发生
- Consistency
- 要从一个完整状态转到另一个完整状态
- Isolation
- 多个transaction之间的状态互不干扰
- Durability
- Transaction结束后，结果应一段时间停留，直到下一次transaction

- Shared lock
- 所有transaction都只能读，不能写
- Exclusive lock
- 某一个transaction上此锁后只能它写和读，其他transaction不能进行任何读写操作

HD@朱昱臻-412729939