ELSEWEDY UNIVERSITY OF TECHNOLOGY
- POLYTECHNIC OF EGYPT -

Database Management Systems

Lecture 8: Transactions and Concurrency Control

© Spring 2023 – **Dr. Alyaa Hamza**
alyaa.hamza@sut.edu.eg

Normalization Example

ID	Name	City	Code Area	C_ID	C_Name	Hours	Grade
1	Ahmed Ali	Tanta	40	CS01	File Structure	80	93
				CS02	System Analysis	70	88
				CS03	Operating System	80	78
2	Heba Samir	Cairo	48	CS01	File Structure	80	55
				CS02	System Analysis	70	87
				CS03	Operating System	80	91
3	Samr Saed	Tanta	40	CS01	File Structure	80	67
				CS02	System Analysis	70	96
				CS03	Operating System	80	87
4	Amr Emad	Cairo	48	CS01	File Structure	80	57
				CS02	System Analysis	70	89
				CS03	Operating System	80	74

First Step : Fill Data

ID	Name	City	Code Area	C_ID	C_Name	Hours	Grade
1	Ahmed Ali	Tanta	40	CS01	File Structure	80	93
1	Ahmed Ali	Tanta	40	CS02	System Analysis	70	88
1	Ahmed Ali	Tanta	40	CS03	Operating System	80	78
2	Heba Samir	Cairo	48	CS01	File Structure	80	55
2	Heba Samir	Cairo	48	CS02	System Analysis	70	87
2	Heba Samir	Cairo	48	CS03	Operating System	80	91
3	Samr Saed	Tanta	40	CS01	File Structure	80	67
3	Samr Saed	Tanta	40	CS02	System Analysis	70	96
3	Samr Saed	Tanta	40	CS03	Operating System	80	87
4	Amr Emad	Cairo	48	CS01	File Structure	80	57
4	Amr Emad	Cairo	48	CS02	System Analysis	70	89
4	Amr Emad	Cairo	48	CS03	Operating System	80	74

Second Step : Apply 1NF (Remove Repeating Group)

ID	Name	City	Code Area
1	Ahmed Ali	Tanta	40
2	Heba Samir	Cairo	48
3	Samr Saed	Tanta	40
4	Amr Emad	Cairo	48

ID	C_ID	C_Name	Hours	Grade
1	CS01	File Structure	80	93
1	CS02	System Analysis	70	88
1	CS03	Operating System	80	78
2	CS01	File Structure	80	55
2	CS02	System Analysis	70	87
2	CS03	Operating System	80	91
3	CS01	File Structure	80	67
3	CS02	System Analysis	70	96
3	CS03	Operating System	80	87
4	CS01	File Structure	80	57
4	CS02	System Analysis	70	89
4	CS03	Operating System	80	74

Third Step: Apply 2NF (Remove Partial Dependency)

ID	Name	City	Code Area
1	Ahmed Ali	Tanta	40
2	Heba Samir	Cairo	48
3	Samr Saed	Tanta	40
4	Amr Emad	Cairo	48

C_ID	C_Name	Hours
CS01	File Structure	80
CS02	System Analysis	70
CS03	Operating System	80

ID	C_ID	Grade
1	CS01	93
1	CS02	88
1	CS03	78
2	CS01	55
2	CS02	87
2	CS03	91
3	CS01	67
3	CS02	96
3	CS03	87
4	CS01	57
4	CS02	89
4	CS03	74

Fourth Step: Apply 3NF (Remove Transitive Dependency)

ID	Name	City
1	Ahmed Ali	Tanta
2	Heba Samir	Cairo
3	Samr Saed	Tanta
4	Amr Emad	Cairo

C_ID	C_Name	Hours
CS01	File Structure	80
CS02	System Analysis	70
CS03	Operating System	80

City	Code Area
Tanta	40
Cairo	48

ID	C_ID	Grade
1	CS01	93
1	CS02	88
1	CS03	78
2	CS01	55
2	CS02	87
2	CS03	91
3	CS01	67
3	CS02	96
3	CS03	87
4	CS01	57
4	CS02	89
4	CS03	74

Transaction and Concurrency Control

What is a Transaction?

- Is a unit of work (Group of Statements) performed within a database management system (or similar system) against a database.
- Represents any change to a database.

Purpose a Transaction?

- To provide isolation between programs accessing a database concurrently.
- Allowing a correct recovery from failures and keep a database consistent even in cases of system failure,

Transaction Done Successfully

Withdraw

Deposit

Product Owner Account

Transaction Done Successfully

Withdraw

Deposit Fail

Product Owner Account

Transaction Properties {ACID}

Atomicity

- Each transaction is treated as a single (unit),

  which either succeeds completely, or fails completely.
- prevents updates to the database occurring only partially.

Consistency

- Any data written to the database must be valid

  according to all defined rules.
- Rules, such as constraints, cascades, triggers, and any combination.

Isolation

- Ensures that concurrent execution of transactions leaves the database in the same state that would have been obtained if the transactions were executed

Durability

- Guarantees that once a transaction has been committed, it will remain committed even in the case of a system failure such (System crash or power failure).

Isolation Level (How it works)

- By using Concurrency Control to handle isolation and guarantee related correctness.
- Two-phase locking is the most common Concurrency Control.

Two-phase locking

- used to provide both serializability and recoverability for correctness

Phase 1: Expanding phase locks are acquired and no locks are released (the number of locks can only increase).

Phase 2: Shrinking phase (aka Contracting phase): locks are released and no locks are acquired.

- ✓ It is safely determined only when a transaction has completed processing and requested commit. In this case, all the locks can be released at once200

Isolation Level (Problems)

- When two or more transactions manipulating the same data at the same time.

01 – Dirty Reads

- When a transaction is allowed to read data from a row that has been modified by another running transaction and not yet committed.

02 – Non-repeatable reads

- Occurs when, during the course of a transaction, a row is retrieved twice and the values within the row differ between reads.

03 – Phantom reads

- Occurs when, in the course of a transaction, new rows are added or removed by another transaction to the records being read.

Isolation Level (Lowest ===> Highest)

01 - Read uncommitted

- Dirty reads are allowed, so one transaction may see not-yet-committed changes made by other transactions.

02 - Read committed

- Guarantees that any data read is committed at the moment it is read.
- Dirty reads are not allowed.
- Keeps write locks (acquired on selected data) until the end of the transaction, but read locks are released as soon as the SELECT operation is performed.
- The non-repeatable reads phenomenon can occur.

Isolation Level (Lowest ===> Highest)

03 - Repeatable reads

- By keeping the read and write locks (acquired on selected data) until the end of the transaction.
- Range-locks are not managed, so phantom reads can occur.

04 - Serializable

- Read and write locks (acquired on selected data) to be released at the end of the transaction.
- Range-locks must be acquired when a SELECT query uses a ranged WHERE clause.
- In this level should be free from any reads problems.

The highest the Isolation level the lowest the DB performance is, because of the locking process.

Terminology

Commit

- If Everything is okay, transaction successfully processed without problems. So commit to DB.

Rollforward

- If the database management system fails entirely, it must be restored from the most recent back-up

Rollback

- If something wrong happened during the transaction

  process so role back the records and cancel the transaction

Compensating transaction

- Exists In systems where commit and rollback mechanisms are not available or undesirable,
- often used to undo failed transactions and restore the system to a previous state.

Terminology

Local

- When a transaction running inside the same application Data Source (JAVA – JDBC - Spring).

Deadlocks

- When 2 transaction waiting for each to achieve the same goal

Global

- When a transaction running across multiple applications server (JAVA – JavaEE – JTA - Spring).

Declarative

- When we are using annotations such as (@Transaction) {java - spring}

SIMPLE MODEL OF A DATABASE (for purposes of discussing transactions):

- A database is a collection of named data items

- Basic operations are read and write
  - read_item(X): Reads a database item named X into a program variable. To simplify our notation, we assume that the program variable is also named X.
  - write_item(X): Writes the value of program variable X into the database item named X.

Two sample transactions

- Two sample transactions:
  - (a) Transaction T1
  - (b) Transaction T2

(a) T_1

```
read_item (X);
X:=X-N;
write_item (X);
read_item (Y);
Y:=Y+N;
write_item (Y);
```

(b) T_2

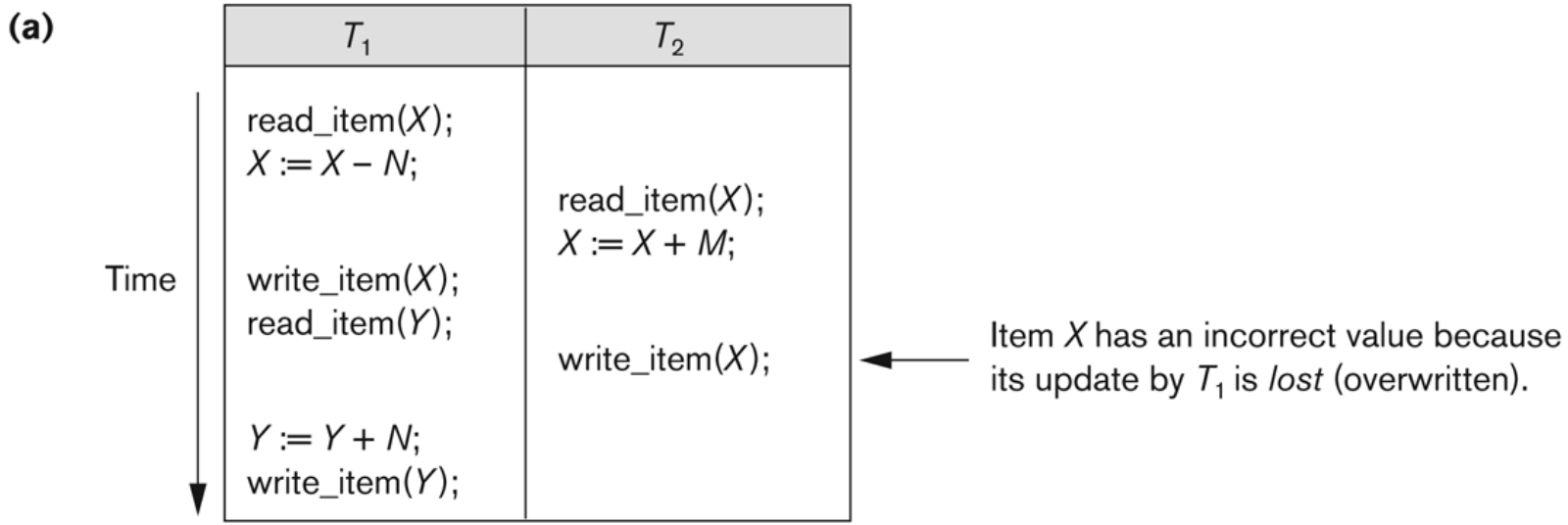
```
read_item (X);
X:=X+M;
write_item (X);
```

Why Concurrency Control is needed:

- The Lost Update Problem
  - This occurs when two transactions that access the same database items have their operations interleaved in a way that makes the value of some database item incorrect.
- The Temporary Update (or Dirty Read) Problem
  - This occurs when one transaction updates a database item and then the transaction fails for some reason
  - The updated item is accessed by another transaction before it is changed back to its original value.
- The Incorrect Summary Problem
  - If one transaction is calculating an aggregate summary function on a number of records while other transactions are updating some of these records, the aggregate function may calculate some values before they are updated and others after they are updated.

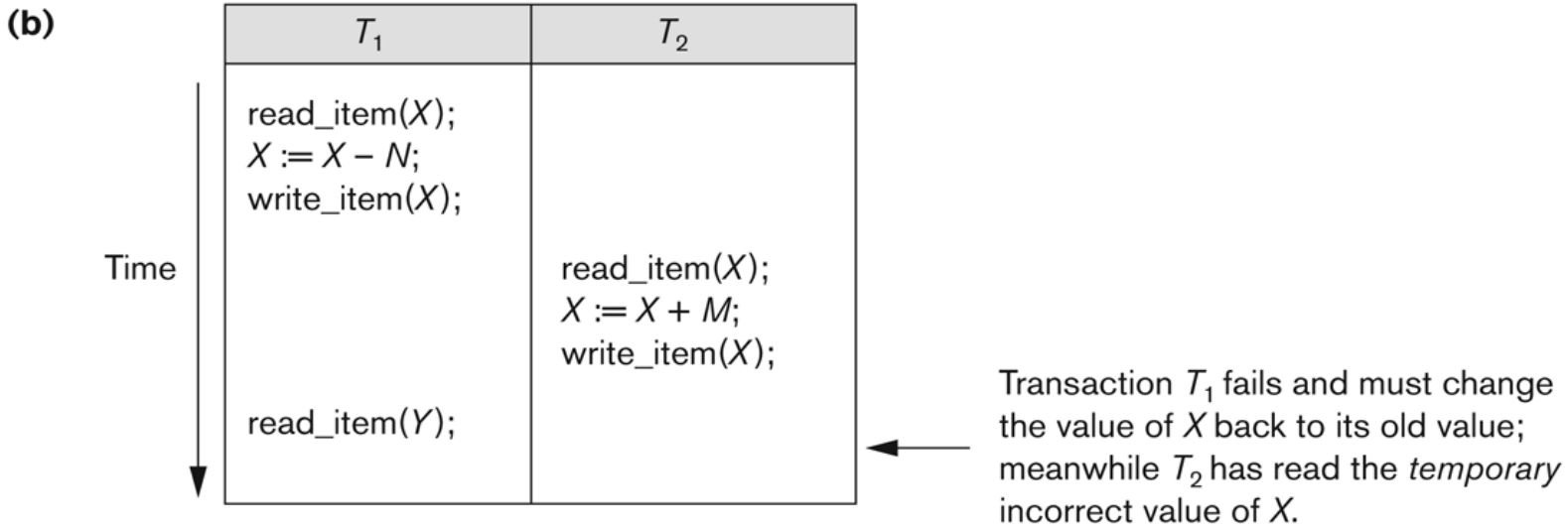
Concurrent execution is uncontrolled:
(a) The lost update problem.

Some problems that occur when concurrent execution is uncontrolled. (a) The lost update problem. (b) The temporary update problem. (c) The incorrect summary problem.



Concurrent execution is uncontrolled:
(b) The temporary update problem.

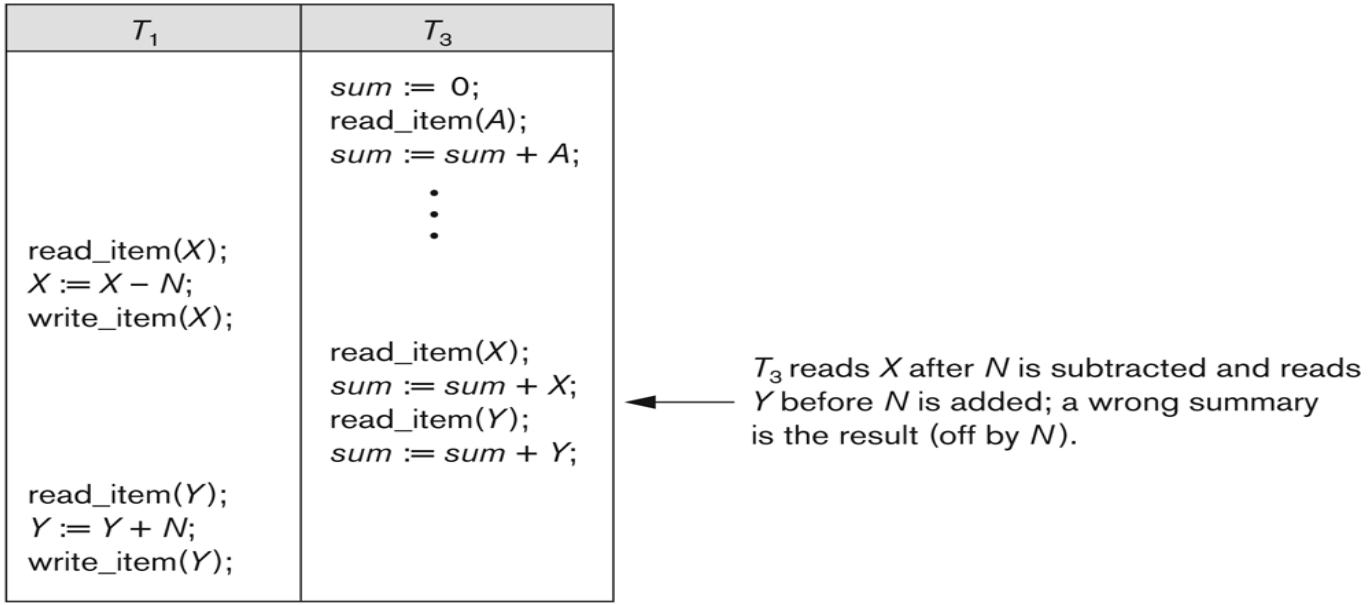
Some problems that occur when concurrent execution is uncontrolled. (a) The lost update problem. (b) The temporary update problem. (c) The incorrect summary problem.



Concurrent execution is uncontrolled:
(c) The incorrect summary problem.

Some problems that occur when concurrent execution is uncontrolled. (a) The lost update problem. (b) The temporary update problem. (c) The incorrect summary problem.

(c)



Why recovery is needed:
(What causes a Transaction to fail)

1. A computer failure (system crash):
A hardware or software error occurs in the computer system during transaction execution. If the hardware crashes, the contents of the computer's internal memory may be lost.

2. A transaction or system error:
Some operation in the transaction may cause it to fail, such as integer overflow or division by zero. Transaction failure may also occur because of erroneous parameter values or because of a logical programming error. In addition, the user may interrupt the transaction during its execution.

Why recovery is needed:
(What causes a Transaction to fail)

3. Local errors or exception conditions detected by the transaction:
Certain conditions necessitate cancellation of the transaction. For example, data for the transaction may not be found. A condition, such as insufficient account balance in a banking database, may cause a transaction, such as a fund withdrawal from that account, to be canceled.
A programmed abort in the transaction causes it to fail.

4. Concurrency control enforcement:
The concurrency control method may decide to abort the transaction, to be restarted later, because it violates serializability or because several transactions are in a state of deadlock.

Why recovery is needed:
(What causes a Transaction to fail)

5. Disk failure:

Some disk blocks may lose their data because of a read or write malfunction or because of a disk read/write head crash. This may happen during a read or a write operation of the transaction.

6. Physical problems and catastrophes:

This refers to an endless list of problems that includes power or air-conditioning failure, fire, theft, sabotage, overwriting disks or tapes by mistake, and mounting of a wrong tape by the operator.

Transaction

For recovery purposes, the system needs to keep track of when the transaction starts, terminates, and commits or aborts.

- Transaction states:

1. Active state
2. Partially committed state
3. Committed state
4. Failed state
5. Terminated State

Transaction and System Concepts

- Recovery manager keeps track of the following operations:
  - begin_transaction: This marks the beginning of transaction execution.
  - read or write: These specify read or write operations on the database items that are executed as part of a transaction.
  - end_transaction: This specifies that read and write transaction operations have ended and marks the end limit of transaction execution.

    - At this point it may be necessary to check whether the changes introduced by the transaction can be permanently applied to the database or whether the transaction has to be aborted because it violates concurrency control or for some other reason.

Transaction and System Concepts

- Recovery manager keeps track of the following operations (cont.):

  - commit_transaction: This signals a successful end of the transaction so that any changes (updates) executed by the transaction can be safely committed to the database and will not be undone.

  - rollback (or abort): This signals that the transaction has ended unsuccessfully, so that any changes or effects that the transaction may have applied to the database must be undone.

Transaction and System Concepts

- Recovery techniques use the following operators:

- undo: Similar to rollback except that it applies to a single operation rather than to a whole transaction.
- redo: This specifies that certain transaction operations must be redone to ensure that all the operations of a committed transaction have been applied successfully to the database.

State transition diagram illustrating the states for transaction execution

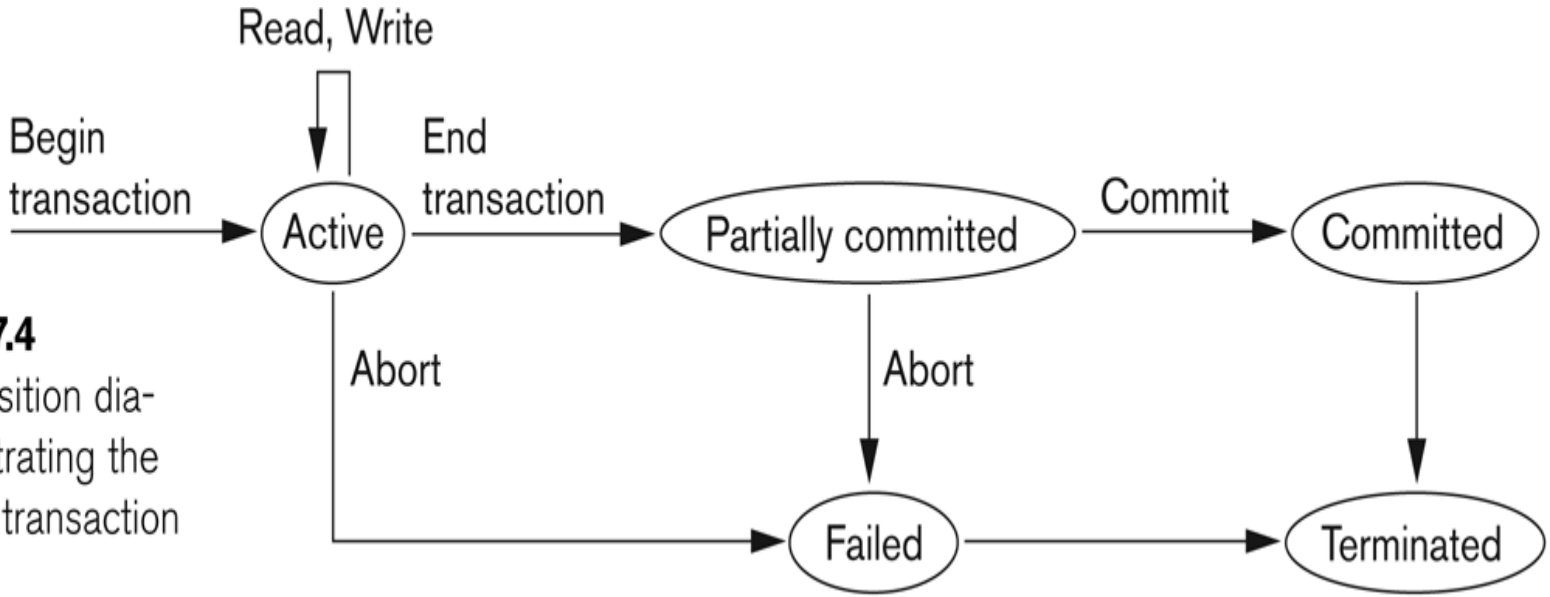


Figure 17.4
State transition diagram illustrating the states for transaction execution.

The System Log

- Log or Journal: The log keeps track of all transaction operations that affect the values of database items.
  - This information may be needed to permit recovery from transaction failures.
  - The log is kept on disk, so it is not affected by any type of failure except for disk or catastrophic failure.
  - In addition, the log is periodically backed up to archival storage (tape) to guard against such catastrophic failures.

Recovery using log records:

- If the system crashes, we can recover to a consistent database state by examining the log and using one of the techniques.

1. Because the log contains a record of every write operation that changes the value of some database item, it is possible to undo the effect of these write operations of a transaction T by tracing backward through the log and resetting all items changed by a write operation of T to their old_values.

2. We can also redo the effect of the write operations of a transaction T by tracing forward through the log and setting all items changed by a write operation of T (that did not get done permanently) to their new_values.


Question
&
Answer