Database Management System – 39 Transaction Processing (Characterizing Schedules Based on Recoverability)

Ajay James
Asst. Prof in CSE
Government Engineering College Thrissur

AJ-GEC THRISSUR

Outline

- Schedule or History
- Notations
- Complete Schedule
- Recoverable and non-recoverable schedules
- Cascading rollback
- Strict Schedule

Characterizing Schedules Based on Recoverability

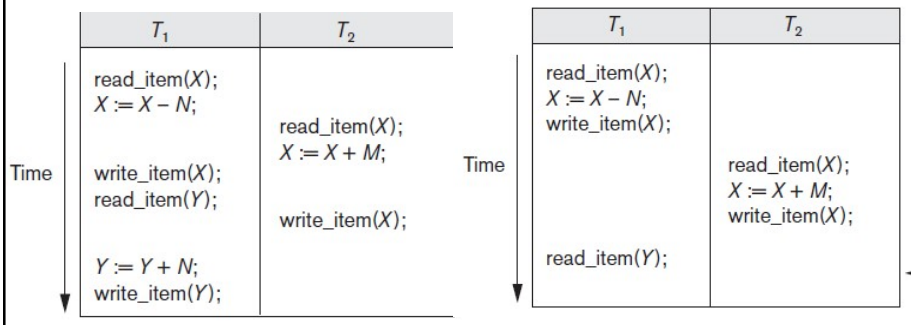
- Schedule or history
  - Order of execution of operations from all transactions
  - S of n transactions T_1, T_2, … , T_n is an ordering of the operations of the transactions
  - Operations from different transactions can be interleaved in the schedule
- Total ordering of operations in a schedule
  - For any two operations in the schedule, one must occur before the other

AJ-GEC THRISSUR

Notations

- b - begin_transaction
- ***r - read_item***
- ***w - write_item***
- e - end_transaction
- ***c – commit***
- ***a - abort***

Schedule examples



S_a: $r_1(X)$; $r_2(X)$; $w_1(X)$; $r_1(Y)$; $w_2(X)$; $w_1(Y)$;

S_b: $r_1(X)$; $w_1(X)$; $r_2(X)$; $w_2(X)$; $r_1(Y)$; a_1;

ALGEC THRISSUR

Conflicting Operations in a Schedule

- Two conflicting operations in a schedule (if they satisfy all the ***three*** conditions)
  - Operations belong to different transactions
  - Operations access the same item *X*
  - At least one of the operations is a *write_item(X)*
- Two operations conflict if changing their order results in a different outcome
- Read-write conflict
- Write-write conflict

S_a: $r_1(X)$; $r_2(X)$; $w_1(X)$; $r_1(Y)$; $w_2(X)$; $w_1(Y)$;

Complete schedule conditions

1. The operations in S are exactly those operations in T_1, T_2, … , T_n, including a commit or abort operation as the last operation for each transaction in the schedule.
2. For any pair of operations from the same transaction T_i, their relative order of appearance in S is the same as their order of appearance in T_i
3. For any two conflicting operations, one of the two must occur before the other in the schedule

- *Partial order*

AJ-SEC THRISSUR

Recoverable and Non-recoverable Schedules

- Once a transaction T is committed, it should never be necessary to roll back T
  - **recoverable schedules**
- A schedule where a committed transaction may have to be rolled back during recovery is called **nonrecoverable**
  - should not be permitted by the DBMS

Recoverable schedule conditions

- A schedule S is recoverable if no transaction T in S commits until all transactions T' that have written some item X that T reads have committed.
- T' should not have been aborted before T reads item X

S_a: $r_1(X)$; $r_2(X)$; $w_1(X)$; $r_1(Y)$; $w_2(X)$; $w_1(Y)$;

S_b: $r_1(X)$; $w_1(X)$; $r_2(X)$; $w_2(X)$; $r_1(Y)$; a_1;

AUGEC THRISSUR

Recoverable schedule example

S_a': $r_1(X)$; $r_2(X)$; $w_1(X)$; $r_1(Y)$; $w_2(X)$; c_2; $w_1(Y)$; c_1;

Recoverable

S_c: $r_1(X)$; $w_1(X)$; $r_2(X)$; $r_1(Y)$; $w_2(X)$; c_2; a_1;
S_d: $r_1(X)$; $w_1(X)$; $r_2(X)$; $r_1(Y)$; $w_2(X)$; $w_1(Y)$; c_1; c_2;
S_e: $r_1(X)$; $w_1(X)$; $r_2(X)$; $r_1(Y)$; $w_2(X)$; $w_1(Y)$; a_1; a_2;

- S_c is not recoverable because T_2 reads item X from T_1, but T_2 commits before T_1commits

Cascading rollback

- Cascading rollback may occur in some recoverable schedules
  - Uncommitted transaction may need to be rolled back
- Cascadeless schedule
  - Avoids cascading rollback

AMGEC THRISSUR

Strict Schedule

- Strict schedule
  - Transactions can neither read nor write an item X until the last transaction that wrote X has committed or aborted
  - Simpler recovery process
    - Restore the before image

$$S_f: w_1(X, 5); w_2(X, 8); a_1;$$

Reference

- Elmasri R. and S. Navathe, Database Systems: Models, Languages, Design and Application Programming, Pearson Education 6th edition and 7th edition

AJ-GEC THRISSUR

Thank you