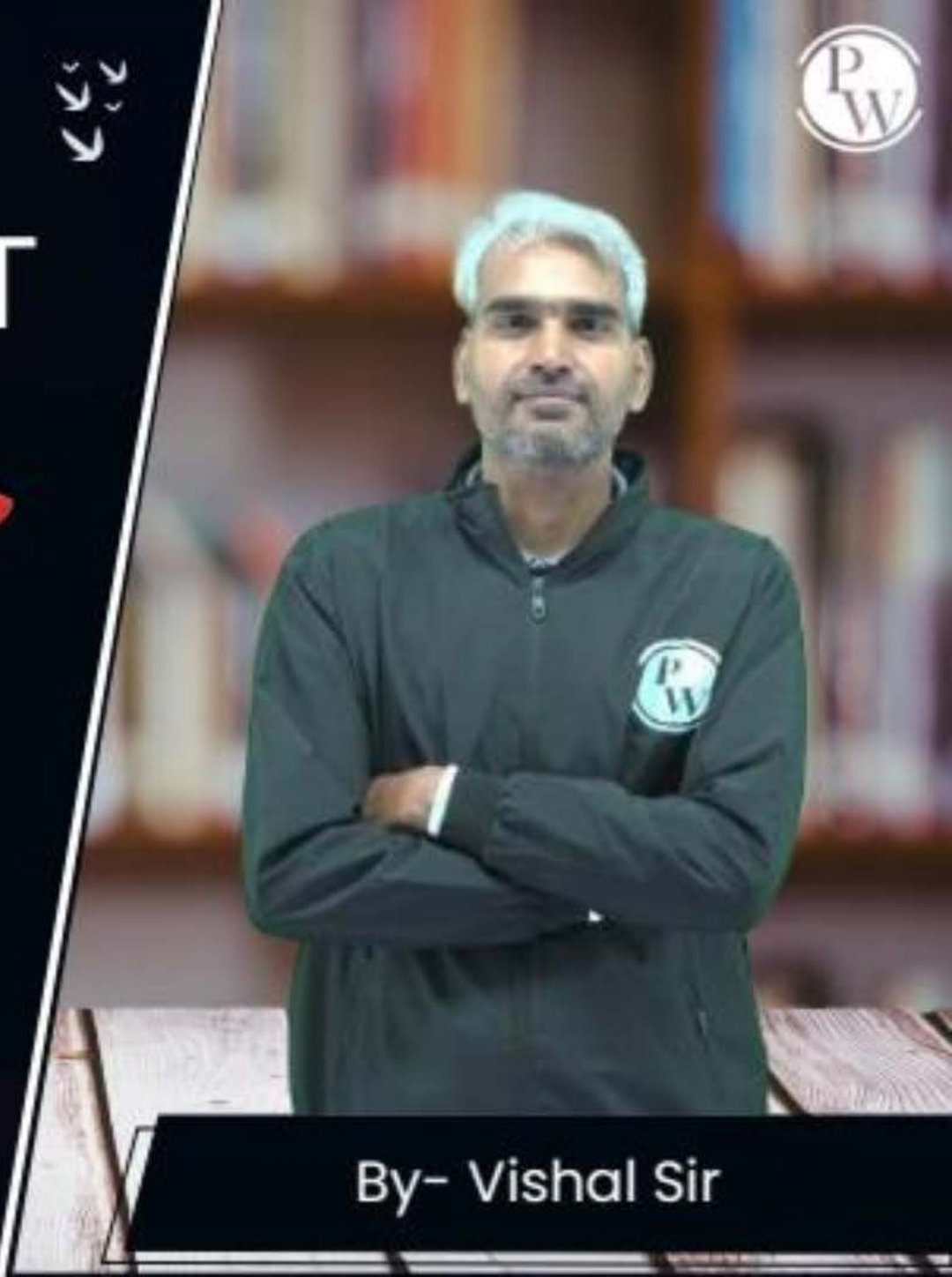
Computer Science & IT

Database Management System

Transaction & concurrency control

Lecture No. 02



By- Vishal Sir

Recap of Previous Lecture

- Topic: Transaction
- Topic: ACID properties
- Topic: Atomicity

Slide

Topics to be Covered

- Topic: Durability
- Topic: Isolation
- Topic: Schedule (Serial schedule & Concurrent schedule)
- Topic: Serializable schedule
- Topic: Consistency

Slide

Topic : Durability

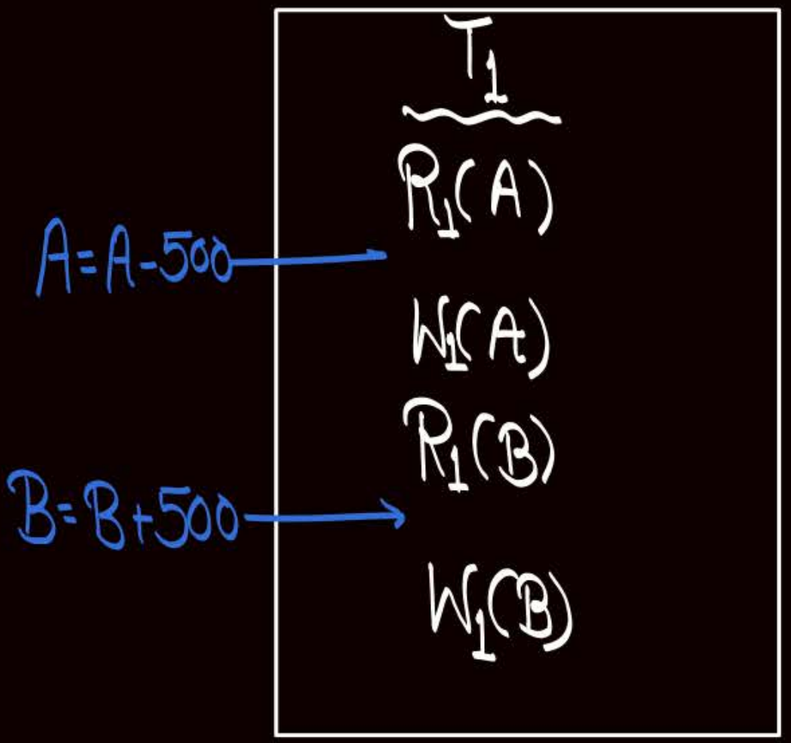
- Changes performed by committed transactions must persist in the database even if the failure occur

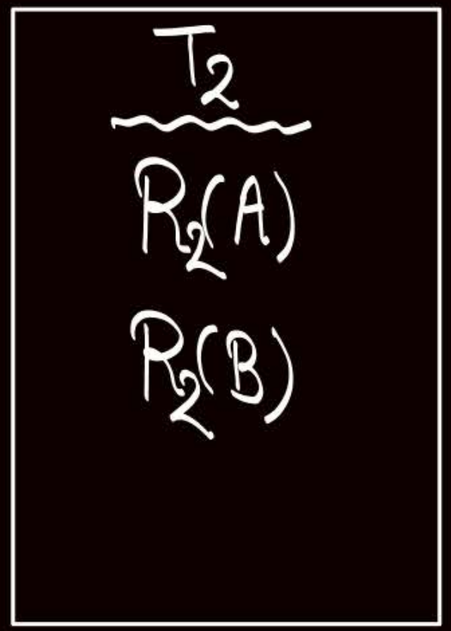
Topic : Isolation

* Isolation says that if two or more transactions are executing Concurrently, then they all must be unaware of each other.

eg:→ Consider two transactions.

- T_1: Transfer Rs 500/- from account A to B
- T_2: Check the balance amount in account A & B respectively





$R_i(X)$: Read of dataitem X by transaction T_i

$W_j(Y)$: Write of dataitem Y by transaction T_j

Topic : Schedule

Time ordered sequence of operations of two or more transaction is called a schedule

eg.

Schedule 'S'

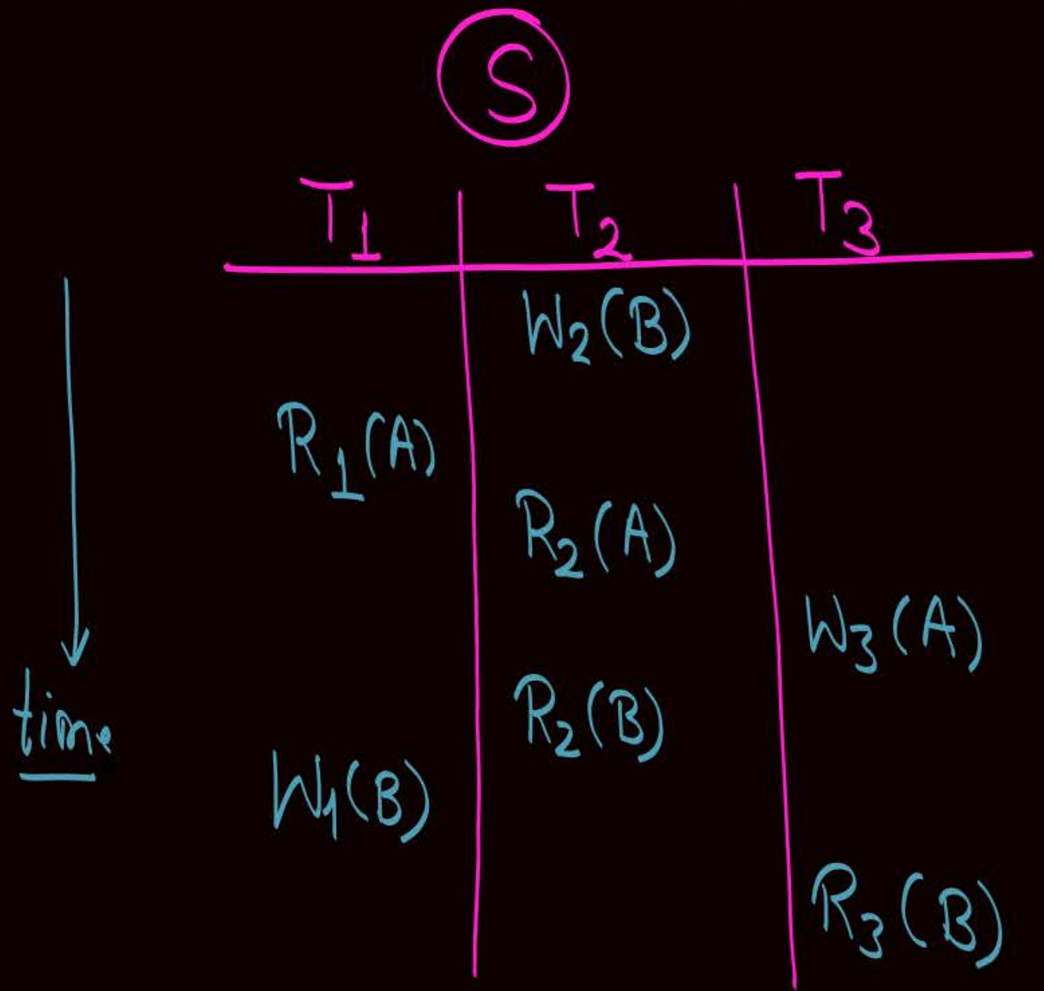
T_1	T_2
$R_1(A)$	
	$R_2(A)$
$W_1(A)$	
	$R_2(B)$
$R_1(B)$	
$W_1(B)$	

time is increasing

$\equiv S = R_1(A), R_2(A), W_1(A), R_2(B), R_1(B), W_1(B)$

* Consider the following Schedule

S: $W_2(B)$, $R_1(A)$, $R_2(A)$, $W_3(A)$, $R_2(B)$, $W_1(B)$, $R_3(B)$



Topic : Schedule

There are two types of schedules

* (1) Serial schedule (one after another)
* (2) Concurrent schedule (interleaved execution of opn is allowed)

Topic : Serial Schedule

(One after another)

- We can start executing the operations of a new transaction only if we have executed all the operations of previously started transaction

Such schedules are called serial schedules

- Consider two transactions T_1 & T_2

  T_1: $R_1(A)$, $W_1(A)$, $R_1(B)$, $W_1(B)$ { transfer Rs 500/- from A to B }

  T_2: $R_2(A)$, $R_2(B)$ { check balance amount in A & B respectively }

- If there are two transactions, then only two serial schedules are possible { (i) $T_1 \rightarrow T_2$: first execute T_1, then T_2, (ii) $T_2 \rightarrow T_1$: first execute T_2, then T_1

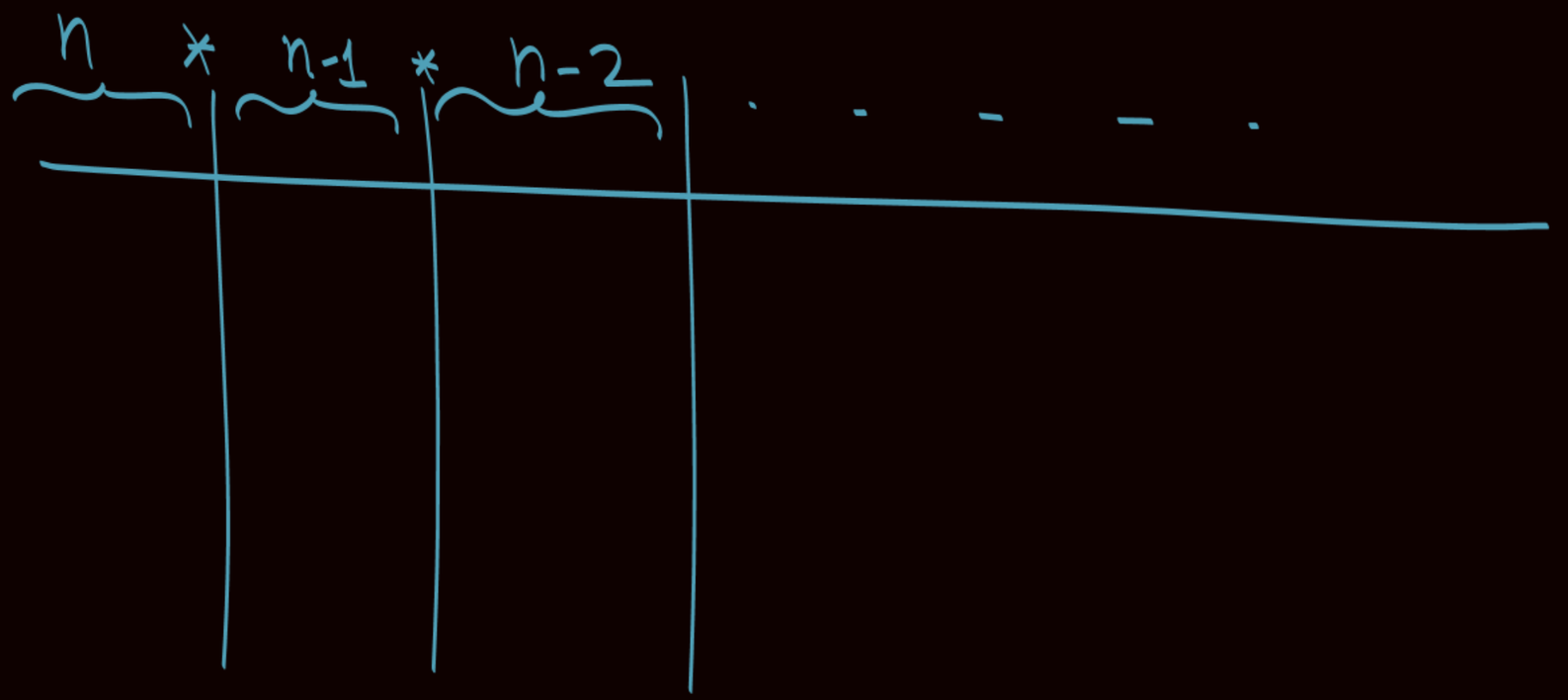
$S_1 \equiv T_1 \rightarrow T_2$

T_1	T_2
$R_1(A)$	
$W_1(A)$	
$R_1(B)$	
$W_1(B)$	
	$R_2(A)$
	$R_2(B)$

$S_2 \equiv T_2 \rightarrow T_1$

T_1	T_2
	$R_2(A)$
	$R_2(B)$
$R_1(A)$	
$W_1(A)$	
$R_1(B)$	
$W_1(B)$	

$T_1, T_2, T_3, \ldots - T_n$



Note: If there are n transactions, then $n!$ serial schedules are possible

Note:- All serial schedules will always satisfy the isolation condition.

↳ But with serial schedules "throughput" of the system will be low.

⇓

∴ We will try to execute transactions concurrently.

Topic : Concurrent Schedule

- If operations of two or more transactions are allowed to get executed in interleaved manner, then such schedule is called Concurrent schedule

- Consider two transactions T_1 & T_2

T_1: $R_1(A)$, $W_1(A)$, $R_1(B)$, $W_1(B)$ { transfer Rs 500/- from A to B }

T_2: $R_2(A)$, $R_2(B)$ { check balance amount in A & B respectively }

S_1

T_1	T_2
$R_1(A)$	
$W_1(A)$	
	$R_2(A)$
	$R_2(B)$
$R_1(B)$	
$W_1(B)$	

S_2

T_1	T_2
$R_1(A)$	
$W_1(A)$	
	$R_2(A)$
$R_1(B)$	
$W_1(B)$	
	$R_2(B)$

- Consider two transactions T_1 & T_2

Let, initially A = 1000 & B = 0

T_1: $R_1(A)$, $W_1(A)$, $R_1(B)$, $W_1(B)$ {transfer Rs 500/- from A to B}

T_2: $R_2(A)$, $R_2(B)$ {Check balance amount in A & B respectively}

→ output of Every Serial Schedule is Considered Correct

$S_1 \equiv T_1 \rightarrow T_2$

T_1	T_2
$R_1(A)$ ← A = 1000	
$W_1(A)$ ← A = 500	
$R_1(B)$ ← B = 0	
$W_1(B)$ ← B = 500	
	$R_2(A)$ → A = 500
	$R_2(B)$ → B = 500

$S_2 \equiv T_2 \rightarrow T_1$

T_1	T_2
	$R_2(A)$ → A = 1000
	$R_2(B)$ → B = 0
$R_1(A)$ → A = 1000	
$W_1(A)$ → A = 500	
$R_1(B)$ → B = 0	
$W_1(B)$ → B = 500	

- Consider two transactions T_1 & T_2

T_1: $R_1(A)$, $W_1(A)$, $R_1(B)$, $W_1(B)$ {transfer Rs 500/- from A to B}

T_2: $R_2(A)$, $R_2(B)$ {check balance amount in A & B respectively}

let, initially A = 1000 & B = 0

Note:- Throughput of the system will increase with Concurrent schedule but, Every Concurrent schedule need not satisfy the isolation Cond^n. {i.e. o/p of Every Concurrent schedule need Not be Correct}

(S1)

T_1	T_2	
$R_1(A)$		A = 1000
$W_1(A)$		A = 500
	$R_2(A)$	A = 500
	$R_2(B)$	B = 0
$R_1(B)$		B = 0
$W_1(B)$		B = 500

This values are neither same as serial schedule $T_1 \rightarrow T_2$, nor same as serial schedule $T_2 \rightarrow T_1$

⇓

In this case T_2 will be able to identify that it is Executing along with some other transaction

⇓

∴ Isolation Cond^n is not satisfied by schedule S_1

(S2)

T_1	T_2	
$R_1(A)$		A = 1000
$W_1(A)$		A = 500
	$R_2(A)$	A = 500
$R_1(B)$		B = 0
$W_1(B)$		B = 500
	$R_2(B)$	B = 500

All values are Exactly same as Serial schedule $T_1 \rightarrow T_2$

⇓

None of the transaction will be able to identify that they are executing along with some other transactions

⇓

Hence isolation Cond^n is satisfied by schedule S_2

Note :-

In order to say that schedule "S" satisfies the isolation condition, behaviour of schedule "S" must be equivalent to at least one of the serial schedule over the transactions of schedule S.

Serializable Schedule :→

A schedule is called a serializable schedule if and only if, behaviour of schedule is equivalent to at least one of the "serial schedule"

Note: Every serial schedule is always a serializable schedule, but Every serializable schedule need not be a serial schedule

Topic : Isolation

* Isolation says that if two or more transactions are executing concurrently, then they all must be unaware of each other.

(or)

* In order to say that a schedule 'S' satisfies isolation condition, behaviour of schedule S must be equivalent to at least one serial schedule.

(or)

* In order to say that schedule 'S' satisfies isolation condn, schedule 'S' must be a serializable schedule

- Consider two transactions T_1 & T_2

T_1: $R_1(A)$, $W_1(A)$, $R_1(B)$, $W_1(B)$ {transfer Rs 500/- from A to B}

T_2: $R_2(A)$, $R_2(B)$ {check balance amount in A & B respectively}

let, initially A = 1000 & B = 0

Note:- Throughput of the system will increase with Concurrent schedule but, Every Concurrent schedule need not satisfy the isolation Cond^n. {i.e o/p of Every Concurent schedule need not be Correct}

S_1

T_1	T_2	
$R_1(A)$		A = 1000
$W_1(A)$		A = 500
	$R_2(A)$	A = 500
	$R_2(B)$	B = 0
$R_1(B)$		B = 0
$W_1(B)$		B = 500

It is not same as Serial schedule $T_1 \rightarrow T_2$
∴ $S_1 \not\equiv T_1 \rightarrow T_2$

It is not same as Serial schedule $T_2 \rightarrow T_1$
∴ $S_1 \not\equiv T_2 \rightarrow T_1$

S_1 is not Equivalent to any of the Serial schedule
Hence, S_1 is a non-Serializable schedule

S_2

T_1	T_2	
$R_1(A)$		A = 1000
$W_1(A)$		A = 500
	$R_2(A)$	A = 500
$R_1(B)$		B = 0
$W_1(B)$		B = 500
	$R_2(B)$	B = 500

It is not same as Serial schedule $T_2 \rightarrow T_1$
∴ $S_2 \not\equiv T_2 \rightarrow T_1$

All values are same as Serial schedule $T_1 \rightarrow T_2$
∴ $S_2 \equiv T_1 \rightarrow T_2$

$S_2 \not\equiv T_2 \rightarrow T_1$, but $S_2 \equiv T_1 \rightarrow T_2$
i.e; S_2 is Equivalent to at least one Serial Schedule, Hence S_2 is a Serializable Schedule
And Equivalent Serial Schedule is $T_1 \rightarrow T_2$

Note : Concurrency Control Component is responsible for avoiding the execution of non-serializable schedule

i.e, Concurrency Control Component ensures serializability

↳ i.e; if a schedule is allowed to execute by Concurrency Control Component, then schedule is a serializable schedule

i.e; Concurrency Control Component ensures isolation

Topic : Consistency

* Consistency states that, before and after the execution of transactions of a schedule, the database must remain Consistent

Criteria for Consistency :-

* ① Schedule must be strict recoverable. ✓
  And ② Schedule must be Serializable schedule ✓

Note:

1. For atomicity Recovery management Component comes into the picture
2. For isolation Concurrency Control component is responsible
3. For Consistency, both recovery management Component and Concurrency Control Component are responsible

H.W.

Q:. Consider two transactions.

T_1: $R_1(A)$, $W_1(A)$, $R_1(B)$, $W_1(B)$

&

T_2: $R_2(A)$, $R_2(B)$

How many total schedules are possible over the operations of transactions T_1 & T_2

2 mins Summary

- Topic: Durability
- Topic: Isolation
- Topic: Schedule (Serial schedule & Concurrent schedule)
- Topic: Serializable schedule
- Topic: Consistency

THANK - YOU