COMPUTER SCIENCE

Database Management System

Transaction & Concurrency Control

Lecture_6

Vijay Agarwal sir

01 Conflict & View Serializable

02 Problem Due to Concurrent Execution

Conflict serializable.

- ① BASIC concept
- ② Testing for Conflicting Serializable
  - Precedence Graph Method
- ③ Conflict Pair
- ④ Conflict equivalent to Any Serial Schedule.

Serializable

① Conflict Serializable

[Sufficient Condition]

② View Serializable

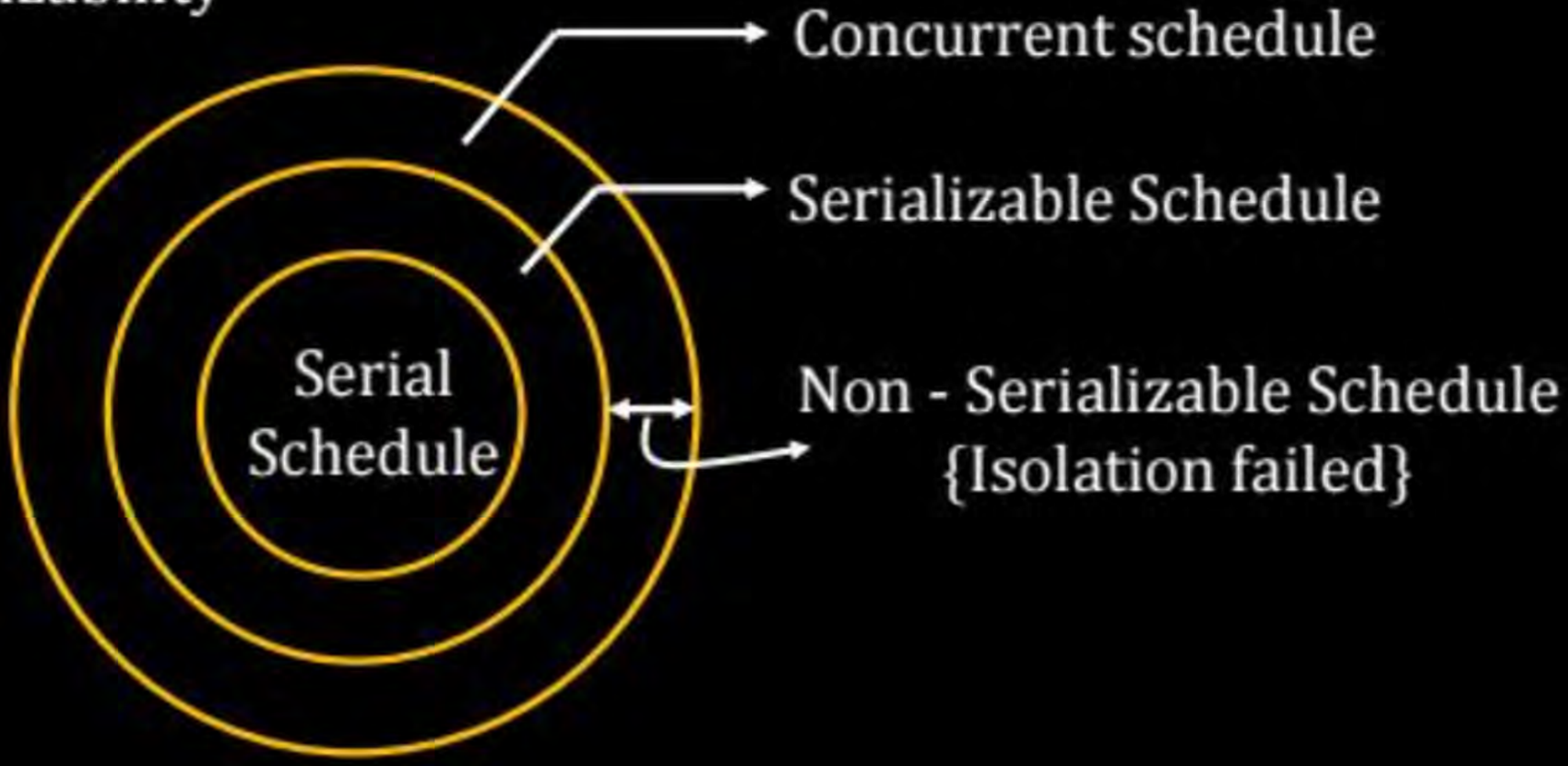
[Necessary & Sufficient]

Serializable Schedule

A Schedule is serializable Schedule if it is equivalent to a Serial Schedule.

(i) Conflict Serializability

(ii) View Serializability



Note

If a Schedule is Conflict serializable then it is already View Serializable.

Note

If a Schedule is Not Conflict Serializable [CNC] then it May | May Not be View Serializable.

↳ If a schedule is Not View Serializable then it is Not Serializable

Note

A Schedule is serializable if either it is Conflict Serializable (or) View Serializable (or) Both.

View Serializability: [S & S']

For each Data Item

1. Initial Read
2. Final Write
3. Write - Read [Updated Read] Sequence

S: Non Serial Schedule (Given Question)

S': Any Serial Schedule of S.

View Serializability

Let S and S' be two schedules with the same set of transactions. S and S' are **view equivalent** if the following three conditions are met, for each data item 'Q'

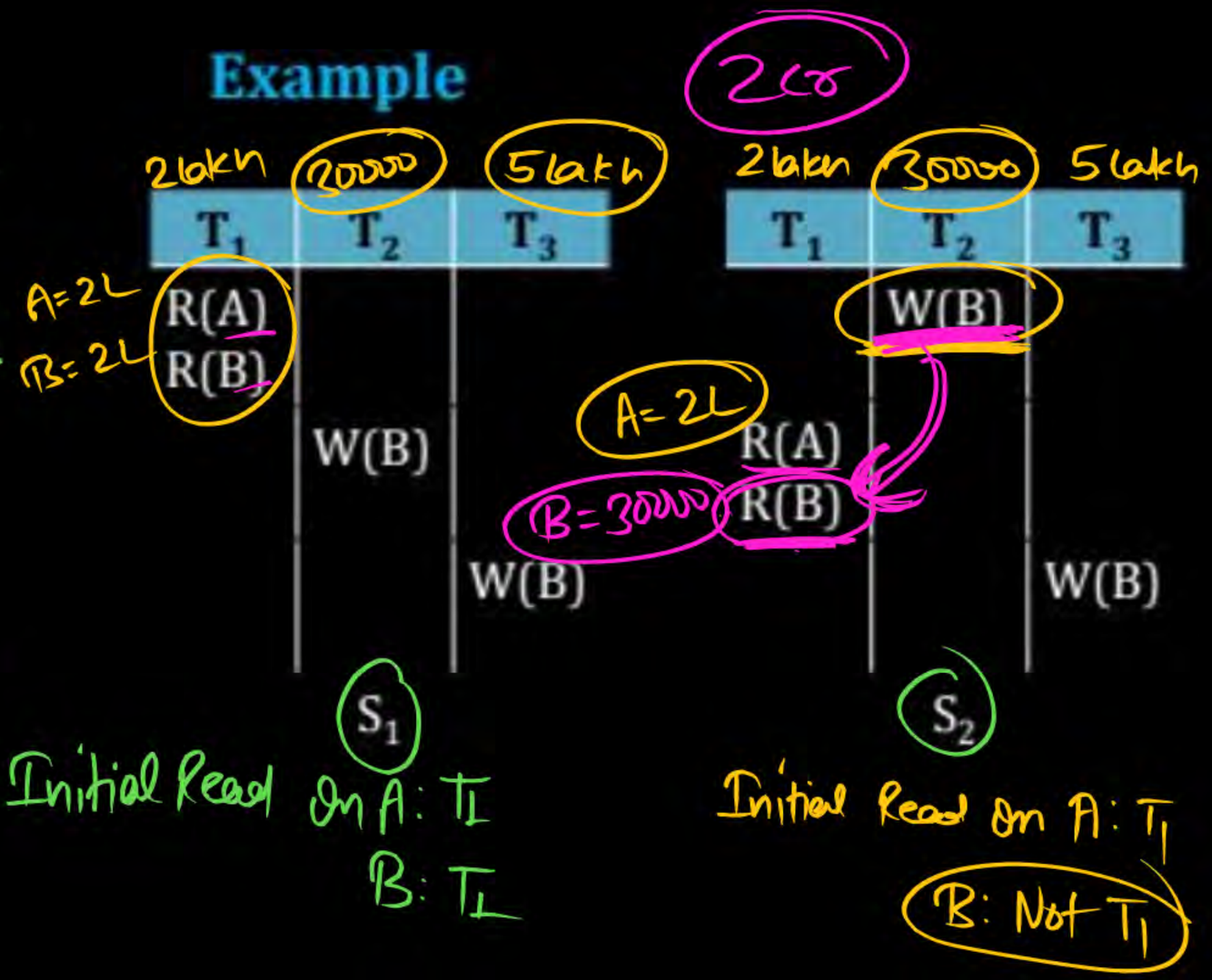
1. If in schedule S, transaction T_i reads the initial value of Q, then in schedule S' also transaction T_i must read the initial value of Q.
2. If in schedule S transaction T_i executes **read**(Q), and that value was produced by transaction T_j (if any), then in schedule S' also transaction T_i must read the value of Q that was produced by the same **write**(Q) operation of transaction T_j.
3. The transaction (if any) that performs the final **write**(Q) operation in schedule S must also perform the final **write**(Q) operation in schedule S'.

View Serializability

- **View Serializable Schedule:** View equivalent serial schedule.
- **View Equivalent:** S_1 and S_2 said to be view equivalent.

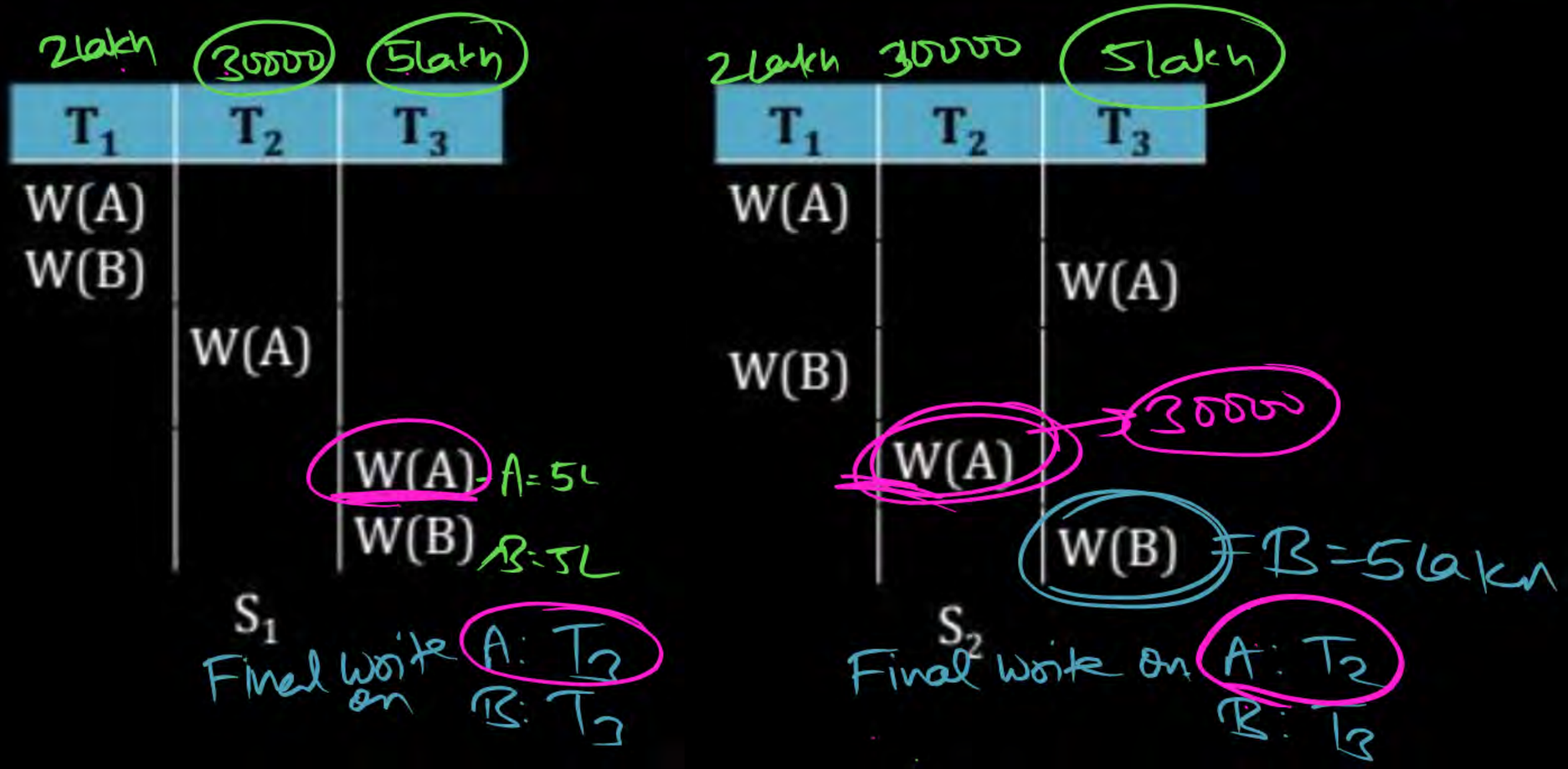
  Only if

  (1) initial reads of S_1 and S_2 should be same.



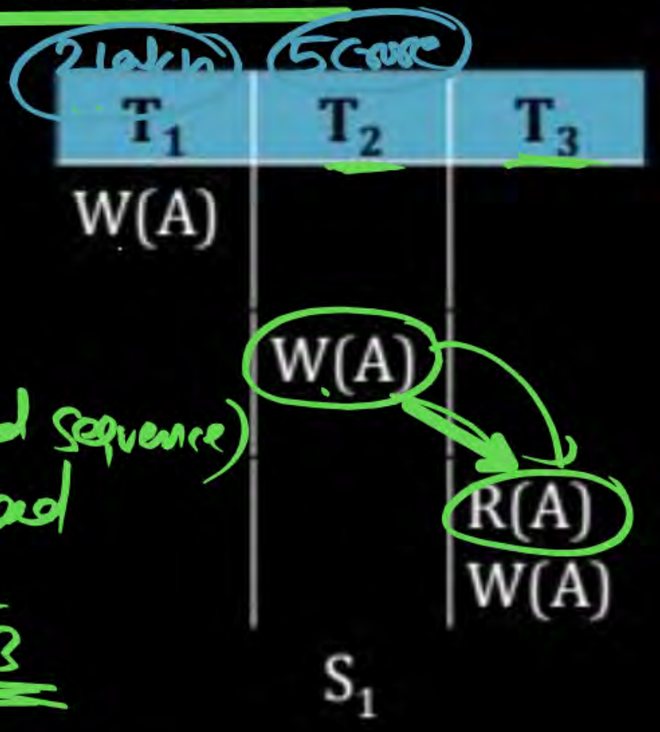
View Serializability

2. Final updations for every data item should be same in S_1 and S_2



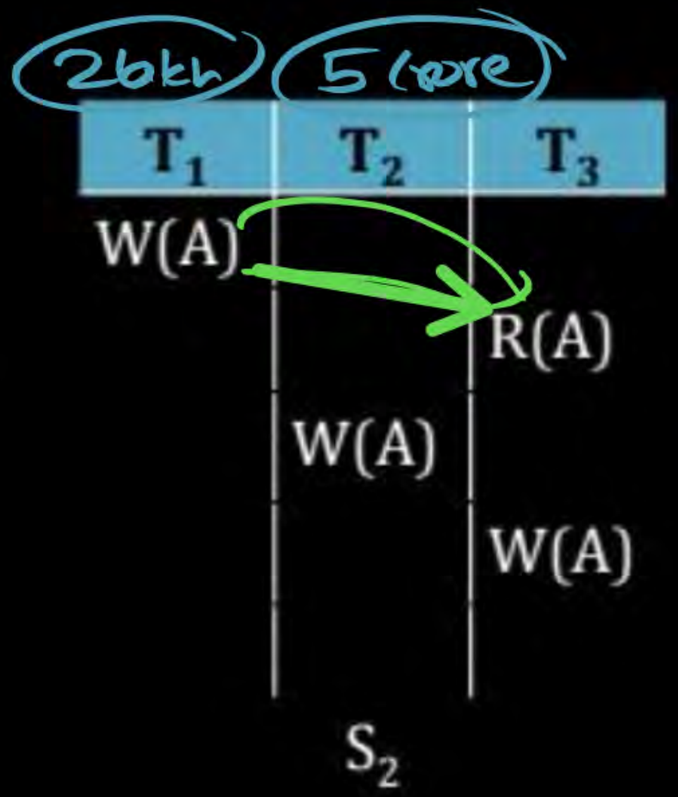
View Serializability

3. Write-Read sequence should also be equal. (Updated Reads should be same



(Write – Read sequence)
UPDATED Read

$T_2 \rightarrow T_3$



UPDATED Read

$T_1 \rightarrow T_3$

Checking View Serializable

Q1

T_1	T_2
R(A)	
W(A)	
	R(A)
	W(A)
R(B)	
W(B)	
	R(B)
	W(B)

C_1

① Initial Read:
A : T_1
B : T_1

② Final Write
A : T_2
B : T_2

③ Updated Read
A : $T_1 \rightarrow T_2$
B : $T_1 \rightarrow T_2$

T_1	T_2
R(A)	
W(A)	
R(B)	
W(B)	
	R(A)
	W(A)
	R(B)
	W(B)

$S_1 < T_1\ T_2 >$

T_1	T_2
	R(A)
	W(A)
	R(B)
	W(B)
R(A)	
W(A)	
R(B)	
W(B)	

$S_2 < T_2\ T_1 >$

C_1 is View equivalent to $S_1 < T_1\ T_2 >$.

A : 1710

B : 3290

5000

$S_1 \subset T_1\ T_2\rangle$

Conflict Serializable

View Serializable

View Serializable : $\subset T_1\ T_2\rangle$

3 Transaction then 3! Serial Schedule.
= 6 Serial Schedule.

<T1 T2 T3>
<T1 T3 T2>
<T2 T1 T3>
<T2 T3 T1>
<T3 T1 T2>
<T3 T2 T1>

6 Serial Schedule.

Dummy.

Pick Any One Condition & Neglect the Some Serial Schedule [option elimination] & then apply all 3 condition.

eg 2

T_1	T_2
R(A)	
	R(A)
	W(A)
W(A)	
R(B)	
W(B)	
	R(B)
	W(B)

$T_1 \rightleftarrows T_2$

CNC

C_2

① Initial Read

A: T_1 or T_2

B: T_1

② final write

A: T_1

B: T_2

T_1	T_2
R(A)	
W(A)	
R(B)	
W(B)	
	R(A)
	W(A)
	R(B)
	W(B)

① Initial Read

final write A: T_2

T_1	T_2
	R(A)
	W(A)
	R(B)
	W(B)
R(A)	
W(A)	
R(B)	
W(B)	

B: T_2

C_2 is Not View Serializable

Practical

C_2 : 1900
3300
5200

In Consistent

(1) Conflict Serializable

CNC

Cycle Not Conflict

(2) View Serializable

Not View Serializable

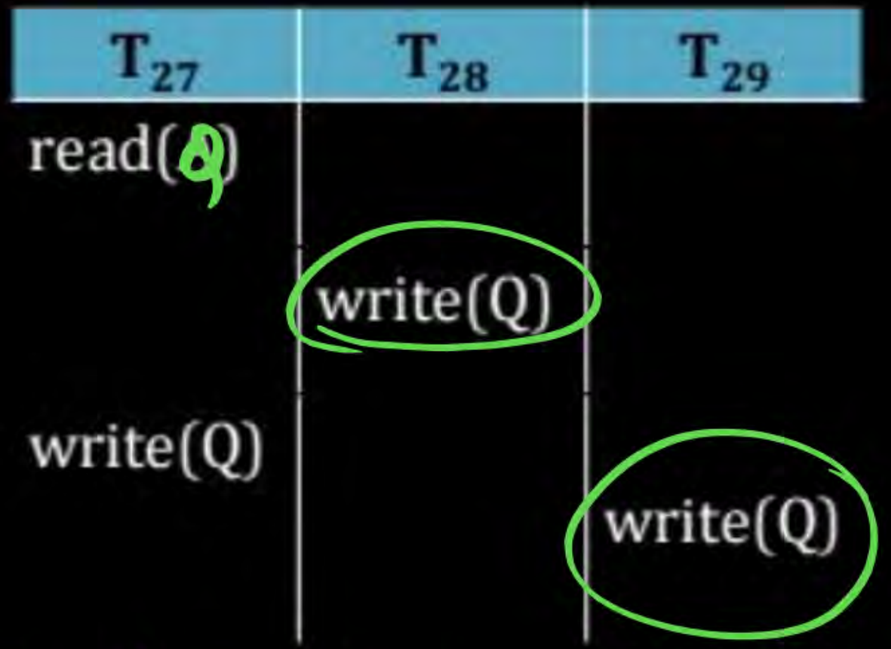
∴ Not Serializable

View Serializability (Cont.)

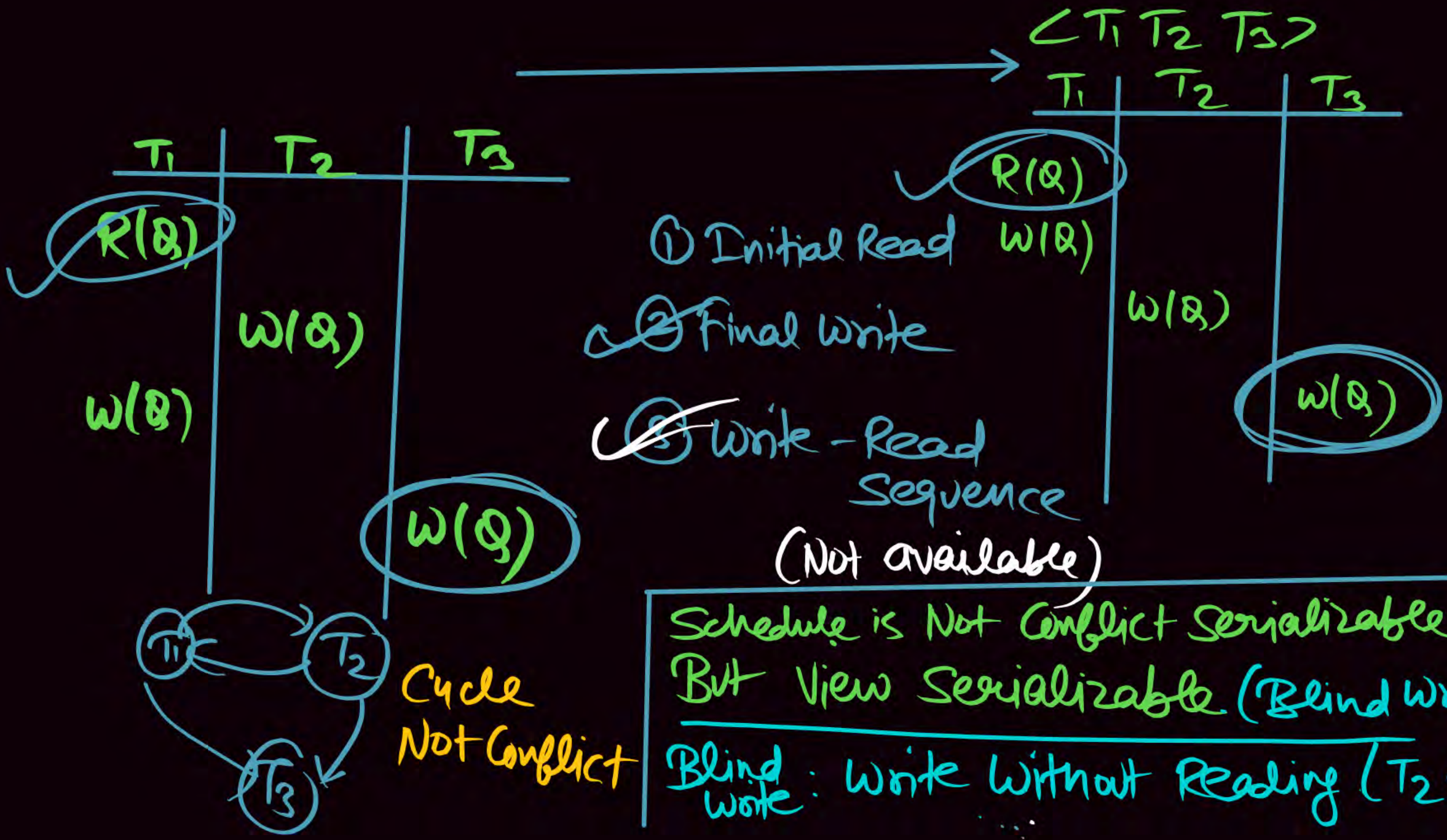
- A schedule S is **view serializable** if it is view equivalent to a serial schedule.
- Every conflict serializable schedule is also view serializable.
- Below is a schedule which is view-serializable but not conflict serializable.

Note:

Every view serializable schedule that is not conflict serializable has **blind writes.**



T_{28} & T_{29} Performing Blind Write.



Schedule is Not Conflict Serializable
But View Serializable (Blind write)

Blind write : Write without Reading (T_2 & T_3)

Q.

T_1	T_2	T_3
	R(A)	
	R(B)	
W(B)		
		R(B)
W(A)		
	W(A)	
		W(A)

① Initial Read
A: T_2
B: T_2

② Final Write
A: T_2
B: T_1

$\langle T_2\ T_3\ T_1 \rangle$

T_1	T_2	T_3
	R(A)	
	R(B)	
	W(A)	
		R(B)
		W(A)

① Initial A: T_2
Read B: T_2

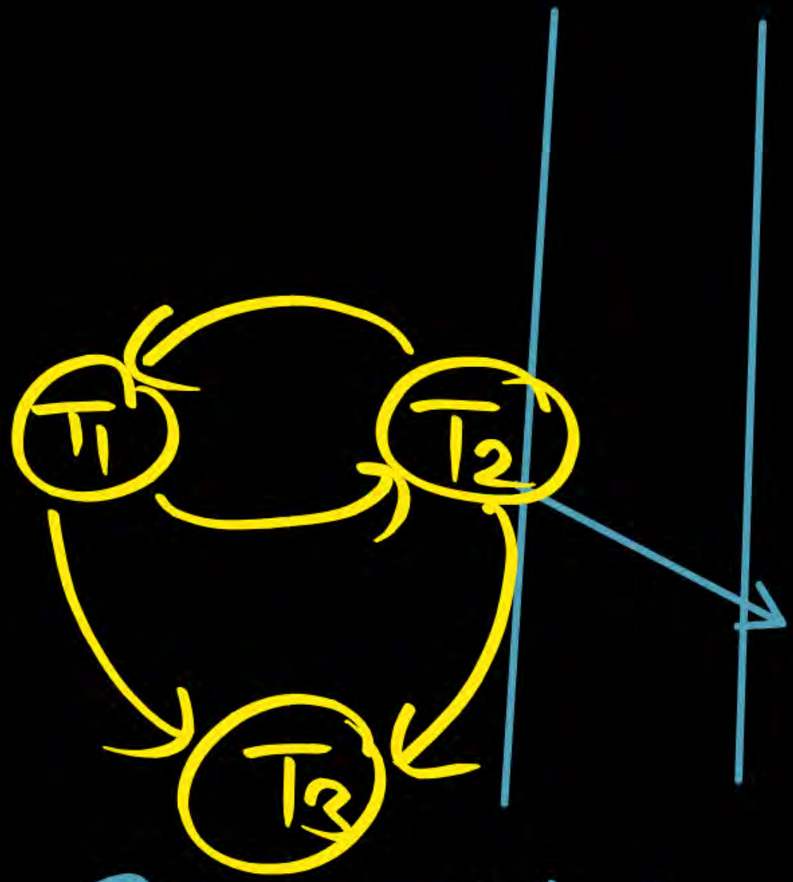
② Final Write
A: T_1
B: T_1

W(B)
W(A)

③ Write Read Fail

Q.

T_1	T_2	T_3
	R(A)	
	R(B)	
W(B)		
		R(B)
W(A)		
	W(A)	
		W(A)



Cycle Not Conflict

But view serializable

∴ Schedule is serializable.

Q. Schedule is View serializable or $\langle T_2\ T_1\ T_3 \rangle$.

T_1	T_2	T_3
	R(A)	
	R(B)	
W(B)		
		R(B)
W(A)		
	W(A)	
		W(A)

$W \rightarrow R$

① Initial Read
A: T_2
B: T_2

② Final Write
A: T_3
B: T_1

③ Write Read Sequence
B: $T_1 \rightarrow T_3$

$\langle T_2\ T_1\ T_3 \rangle$

T_1	T_2	T_3
	R(A)	
	R(B)	
	W(A)	
W(B)		
W(A)		
		R(B)
		W(A)

(Write-Read Sequence)
③ Updated Read
B: $T_1 \rightarrow T_3$

① Initial Read
A: T_2
B: T_2

② Final Write
A: T_3
B: T_1

3 Transaction

then 3!

6 Serial schedule

Dummy.

① Initial Read : T_2.

② Final write : T_3

✗ $\langle T_1\ T_2\ T_3 \rangle$

✗ $\langle T_1\ T_3\ T_2 \rangle$

$\langle T_2\ T_1\ T_3 \rangle$

✗ $\langle T_2\ T_3\ T_1 \rangle$

✗ $\langle T_3\ T_1\ T_2 \rangle$

✗ $\langle T_3\ T_2\ T_1 \rangle$

MCQ Q.13

Consider the following transactions with data items P and Q initialized to zero:

T_1: read (P);
read (Q);
if P = 0 then Q := Q + 1 ;
write (Q).

T_2: read (Q) ;
read (P);
if Q = 0 then P := P + 1 ;
write (P).

Any non-serial interleaving of T_1 and T_2 for concurrent execution leads to

[GATE-2012-CS: 1M]

A a serializable schedule

B a schedule that is not conflict serializable

C a conflict serializable schedule

D a schedule for which a precedence graph cannot be drawn

Q.14

Consider the following schedule S of transactions T_1 and T_2:

Which of the following is TRUE about the schedule S? [2004: 2 Marks]

T_1	T_2
Read(A)	
A = A–10	
	Read(A)
	Temp = 0.2 * A
	Write(A)
	Read(B)
Write(A)	
Read(B)	
B = B +10	
Write(B)	
	B = B + Temp
	Write(B)

A. S is serializable only as T_1, T_2

B. S is serializable only as T_2, T_1

C. S is serializable both as T_1, T_2 and T_2 T_1

D. S is not serializable either as T_1 or as T_2

MCQ **Q.15**

Consider a simple checkpointing protocol and the following set of operations in the log.
(start, T4); (write, T4, y, 2, 3); (start, T1);
(commit, T4); (write, T1, z, 5, 7);
(checkpoint);
(start, T2); (write, T2, x, 1, 9); (commit, T2);
(start, T3); (write, T3, z, 7, 2);
If a crash happens now and the system tries to recover using both undo and redo operations, what are the contents of the undo list and the redo list?

[GATE-2015-CS: 2M]

A. Undo: T3, T1; Redo: T2

B. Undo: T3, T1; Redo: T2, T4

C. Undo: none; Redo: T2, T4, T3, T1

D. Undo: T3, T1, T4; Redo: T2

Problem due to Concurrent execution:

1. WR (Write - Read) Problem | Dirty Read | Uncommitted Read
2. RW (Read - Write) | Un | non Repeatable Read Problem
3. WW (Write - Write) Problem | Lost Update Problem.
4. Phantom Tuple Problem.

THANK YOU!