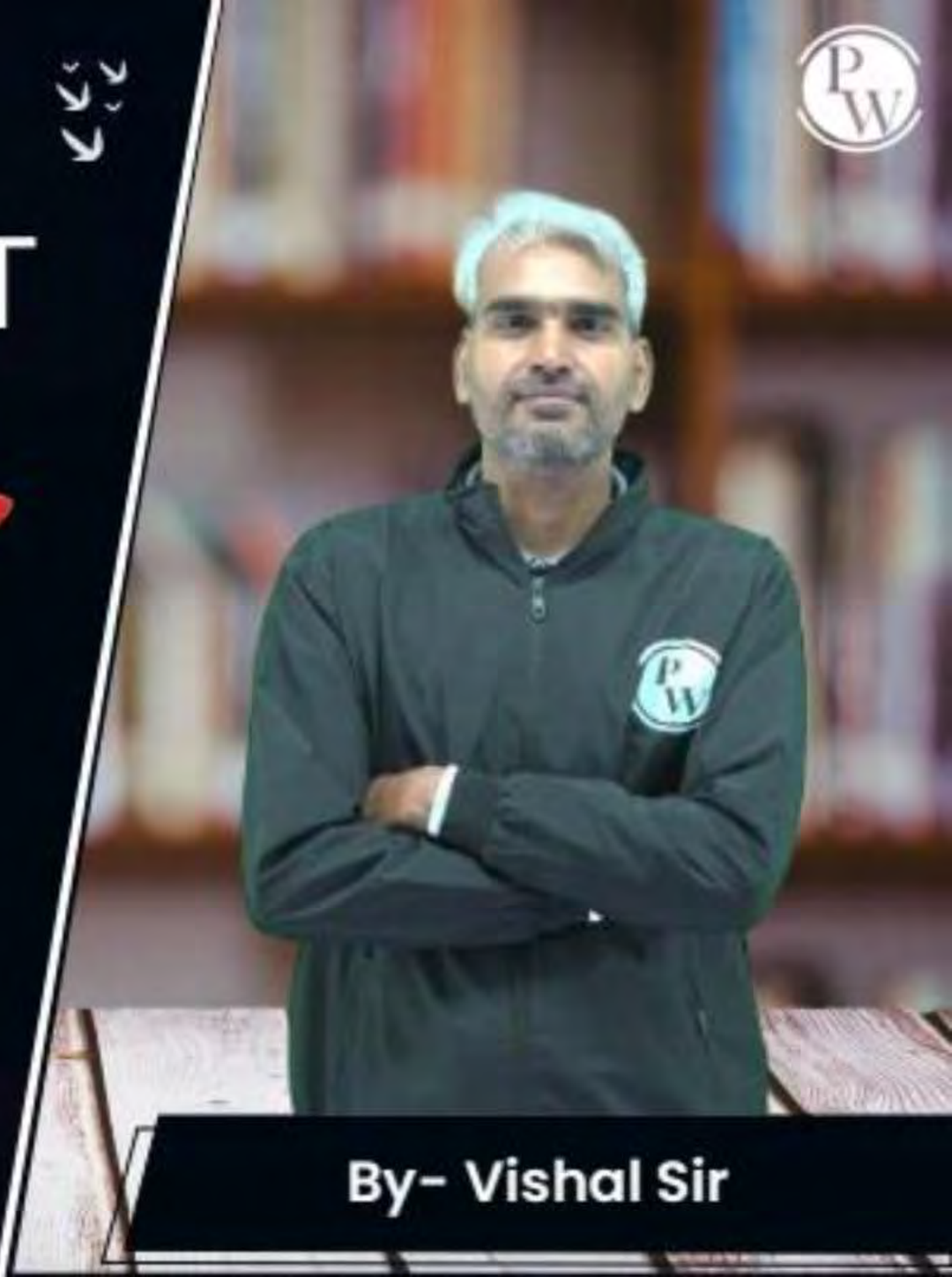
Computer Science & IT

Database Management System

Transactions & Concurrency control

Lecture No. 12



By- Vishal Sir

Recap of Previous Lecture

- Topic: Conservative 2PL
- Topic: Rigorous 2PL
- Topic: Time stamp ordering protocols
- Topic: Read time stamp
- Topic: Write time stamp

Slide

Topics to be Covered

Topic — Basic time stamp ordering protocol

Topic — Time stamp ordering protocol with Thomas write rule

Slide

Topic : Time stamp ordering protocols

There are two types of time stamp ordering protocols

1. Basic time stamp ordering protocol (B.T.S.O.P.)
2. Thomas Write time stamp ordering protocol (T.W.T.S.O.P.)

(or)

Time stamp ordering protocol with Thomas Write rule

Time Stamp :- Time stamp is a unique value assigned to each transaction by database management system.

Time stamps are assigned in ascending order.

- Let T_1 & T_2 are two transactions.

If Time stamp of transaction T_1 < Time stamp of transaction T_2

i.e. $T.S.(T_1) < TS(T_2)$

then T_1 is old transaction
& T_2 is young transaction

① Read time stamp of dataitem A :

RTS(A)

It is the highest time stamp value among the time stamps of transactions that has performed the Read(A) operation successfully.

Initially RTS(A) = 0

② Write time stamp of dataitem A :

WTS(A) It is the highest time stamp value among the time stamps of transactions that has performed the Write(A) operation successfully.

Initially WTS(A) = 0

RTS(A) & WTS(A) :→

$TS(T_1)=10$ $TS(T_2)=20$ $TS(T_3)=30$ $TS(T_4)=40$

T_1	T_2	T_3	T_4	RTS(A)	WTS(A)
				0	0
R(A)				10	0
		R(A)		30	0
	W(A)			30	20
	R(A)			30	20
			W(A)	30	40
		W(A)		30	40

time increasing ↓

Topic : Basic Time stamp ordering protocol

A schedule is allowed to execute using B.T.S.O.P. if and only if schedule is Conflict serializable schedule and Conflict equivalent serial schedule is based on time stamp ordering of the transactions in the schedule.

Eg. Consider the following schedule

TS=20 T_1	TS=10 T_2	TS=30 T_3

Time stamp ordering

$TS(T_2) < TS(T_1) < TS(T_3)$

i.e. $T_2 \rightarrow T_1 \rightarrow T_3$

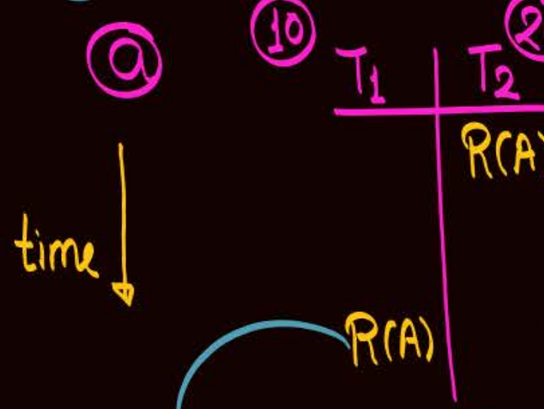
If schedule is a C.S.S. and Conflict equivalent serial schedule is based on time stamp ordering i.e. $T_2 \rightarrow T_1 \rightarrow T_3$, then it is allowed to execute using B.T.S.O.P.

Basic Time stamp ordering Protocol :-

Let T_1 & T_2 are two transactions and $TS(T_1) < TS(T_2)$ i.e. T_1 is old, T_2 is new

(1) When transaction T_1 issue a Read(A) opⁿ :-

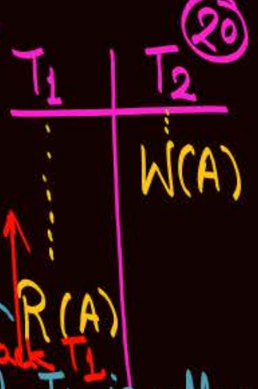
(a)



$TS(T_1) < TS(T_2)$
∴ Time stamp ordering is $T_1 \rightarrow T_2$

If T_1 is allowed to perform this R(A) opⁿ then also the behaviour of schedule will be Conflict equivalent to serial schedule based on time stamp ordering {i.e. $T_1 \rightarrow T_2$}
∴ Transaction T_1 is allowed to perform R(A) opⁿ

(b)



$TS(T_1) < TS(T_2)$
∴ Time stamp ordering is $T_1 \rightarrow T_2$

If T_1 is allowed to perform this R(A) opⁿ then behaviour of schedule will not be Conflict equivalent to serial schedule based on Time Stamp ordering {i.e $T_1 \rightarrow T_2$}
∴ Transaction T_1 is not allowed to perform this R(A) opⁿ, Hence Rollback T_1

if $WTS(A) > TS(T_1)$, then Rollback T_1.

Reader only need to worry about writer

Basic Time stamp ordering Protocol :-

i.e. T_1 is old, T_2 is new

Let T_1 & T_2 are two transactions and $TS(T_1) < TS(T_2)$

② When transaction T_1 issue a Write(A) opⁿ :.

ⓒ

⑩ T_1	⑳ T_2
	R(A)

$TS(T_1) < TS(T_2)$
∴ Time stamp ordering is $T_1 \rightarrow T_2$

time ↓

W(A)

Rollback T_1

If T_1 is allowed to perform this W(A) opⁿ then behaviour of schedule will not be Conflict equivalent to serial schedule based on Time Stamp ordering {i.e $T_1 \rightarrow T_2$}

∴ Transaction T_1 is not allowed to perform this W(A) opⁿ, Hence Rollback T_1

ⓓ

⑩ T_1	⑳ T_2
	W(A)

$TS(T_1) < TS(T_2)$
∴ Time stamp ordering is $T_1 \rightarrow T_2$

W(A)

Rollback T_1

If T_1 is allowed to perform this W(A) opⁿ then behaviour of schedule will not be Conflict equivalent to serial schedule based on Time Stamp ordering {i.e $T_1 \rightarrow T_2$}

∴ Transaction T_1 is not allowed to perform this W(A) opⁿ, Hence Rollback T_1

if $RTS(A) > TS(T_1)$, then Rollback T_1 + if $WTS(A) > TS(T_1)$, then Rollback T_1.

Writer will have problem with reader as well as Writer

Basic Time stamp ordering Protocol Conditions :-

① When T_1 issue a Read(A) oph

(i) If $WTS(A) > TS(T_1)$
then Rollback T_1

(ii) otherwise,
T_1 is allowed to perform R(A) oph,
∴ T_1 will perform Read(A) oph
and set $RTS(A) = Max(RTS(A), TS(T_1))$

② When T_1 issue a Write(A) oph

(i) If $RTS(A) > TS(T_1)$,
then Rollback T_1

(ii) If $WTS(A) > TS(T_1)$,
then Rollback T_1

(iii) otherwise,
T_1 is allowed to perform W(A) oph
∴ T_1 will perform W(A) oph
and set $WTS(A) = TS(T_1)$

Q: Consider the following schedule

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

Which of the following Time stamp Ordering of the transaction will allow the schedule to be executed using B.T.S.O.P.

(a) $(TS(T_1), TS(T_2), TS(T_3)) = (10, 20, 30)$

(b) $(TS(T_1), TS(T_2), TS(T_3)) = (20, 10, 30)$

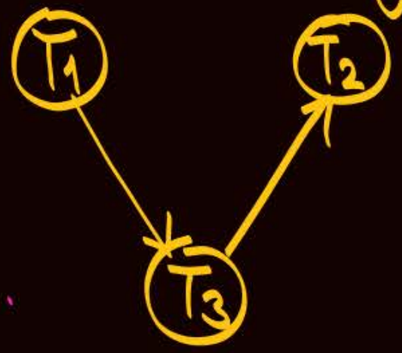
(c) $(TS(T_1), TS(T_2), TS(T_3)) = (30, 10, 20)$

(d) $(TS(T_1), TS(T_2), TS(T_3)) = (10, 30, 20)$

Q: Consider the following schedule

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

Precedence graph



Acyclic
∴ C.S.S.
Conflict equivalent serial schedule is $T_1 \rightarrow T_3 \rightarrow T_2$

for the schedule to be allowed using BTSOP Time stamp ordering must be $TS(T_1) < TS(T_3) < TS(T_2)$

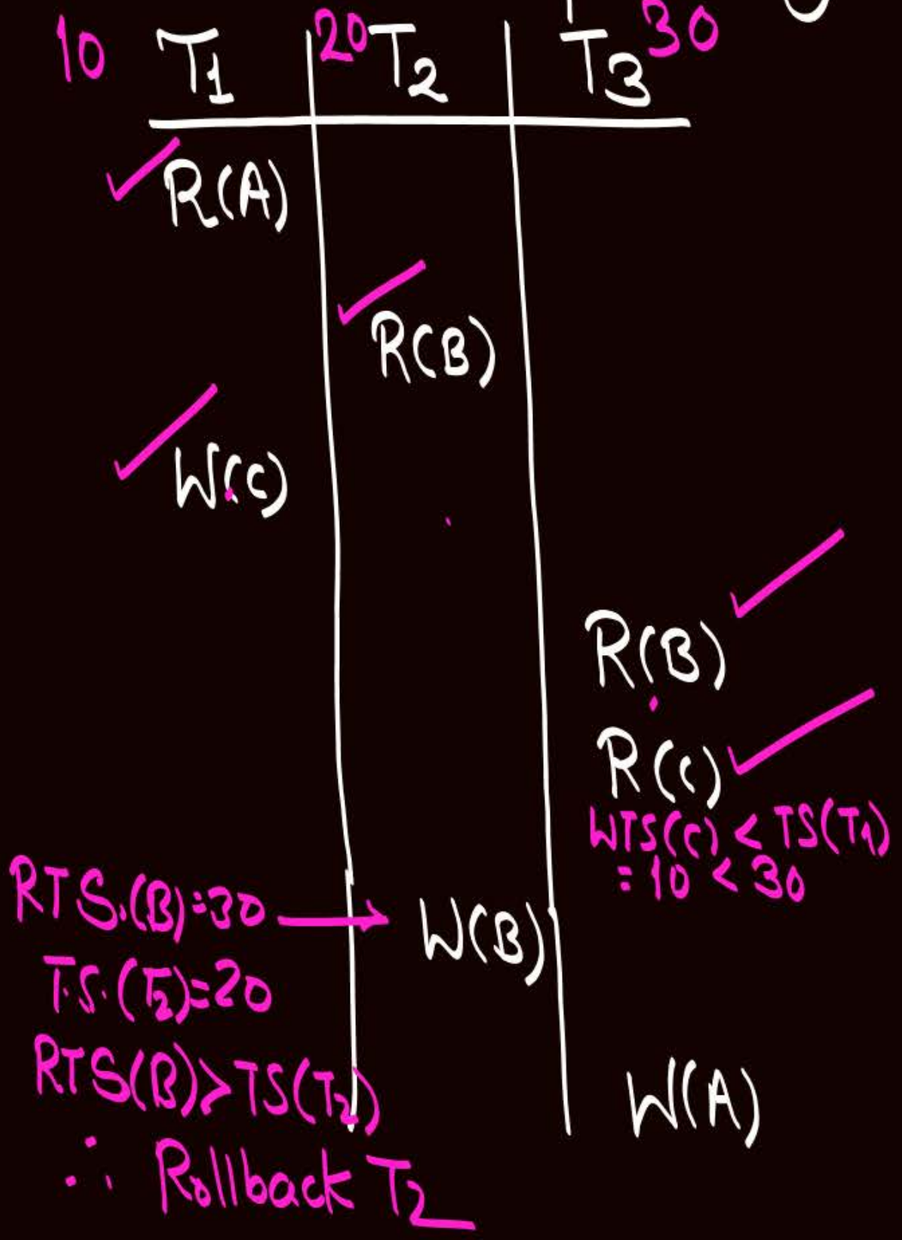
a) $(TS(T_1), TS(T_2), TS(T_3)) = (10, 20, 30)$

b) $(TS(T_1), TS(T_2), TS(T_3)) = (20, 10, 30)$

c) $(TS(T_1), TS(T_2), TS(T_3)) = (30, 10, 20)$

d) $(TS(T_1), TS(T_2), TS(T_3)) = (10, 30, 20)$

Q: Consider the following schedule



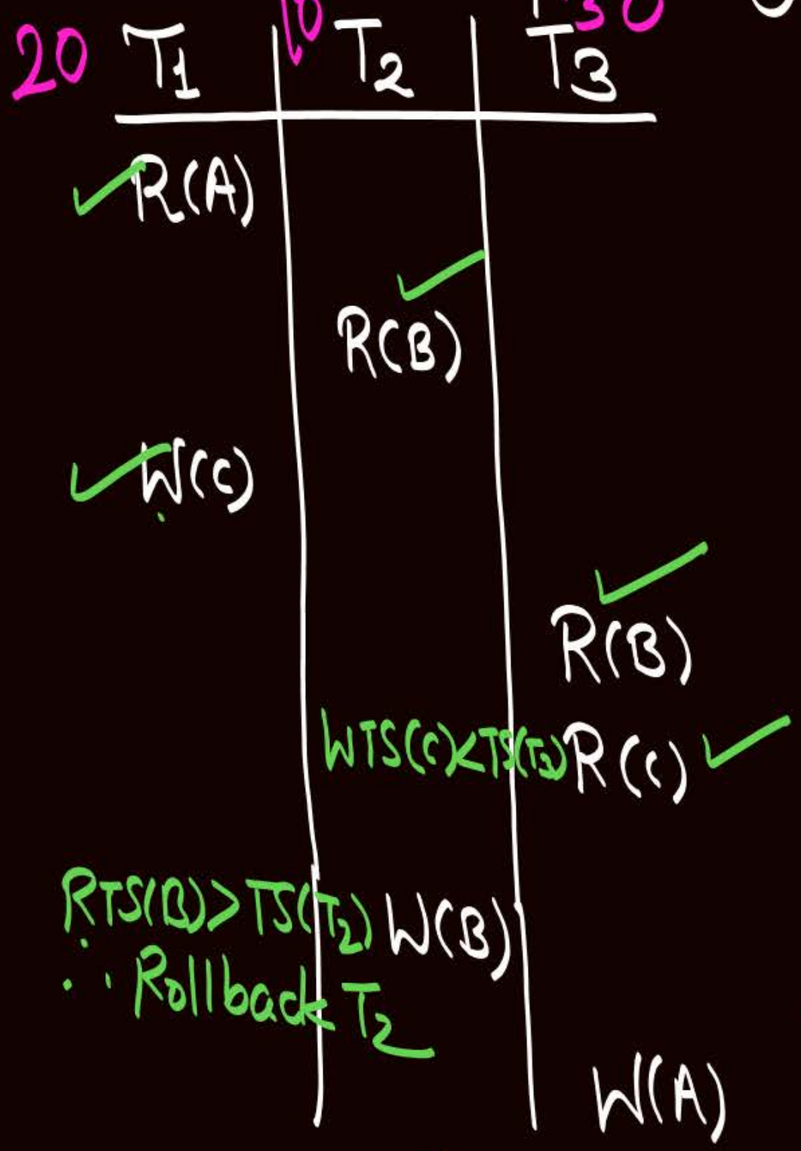
a) $(TS(T_1), TS(T_2), TS(T_3)) = (10, 20, 30)$

b) $(TS(T_1), TS(T_2), TS(T_3)) = (20, 10, 30)$

c) $(TS(T_1), TS(T_2), TS(T_3)) = (30, 10, 20)$

d) $(TS(T_1), TS(T_2), TS(T_3)) = (10, 30, 20)$

Q: Consider the following schedule



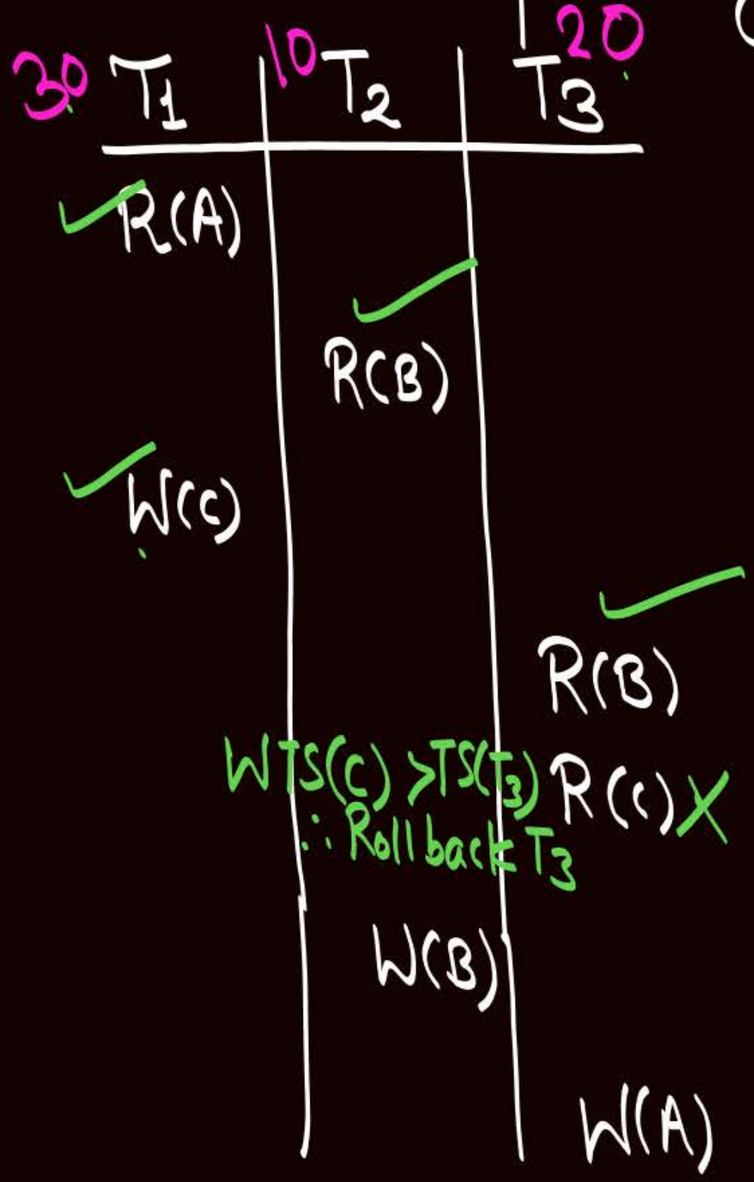
(a) $(TS(T_1), TS(T_2), TS(T_3)) = (10, 20, 30)$

(b) $(TS(T_1), TS(T_2), TS(T_3)) = (20, 10, 30)$

(c) $(TS(T_1), TS(T_2), TS(T_3)) = (30, 10, 20)$

(d) $(TS(T_1), TS(T_2), TS(T_3)) = (10, 30, 20)$

Q: Consider the following schedule



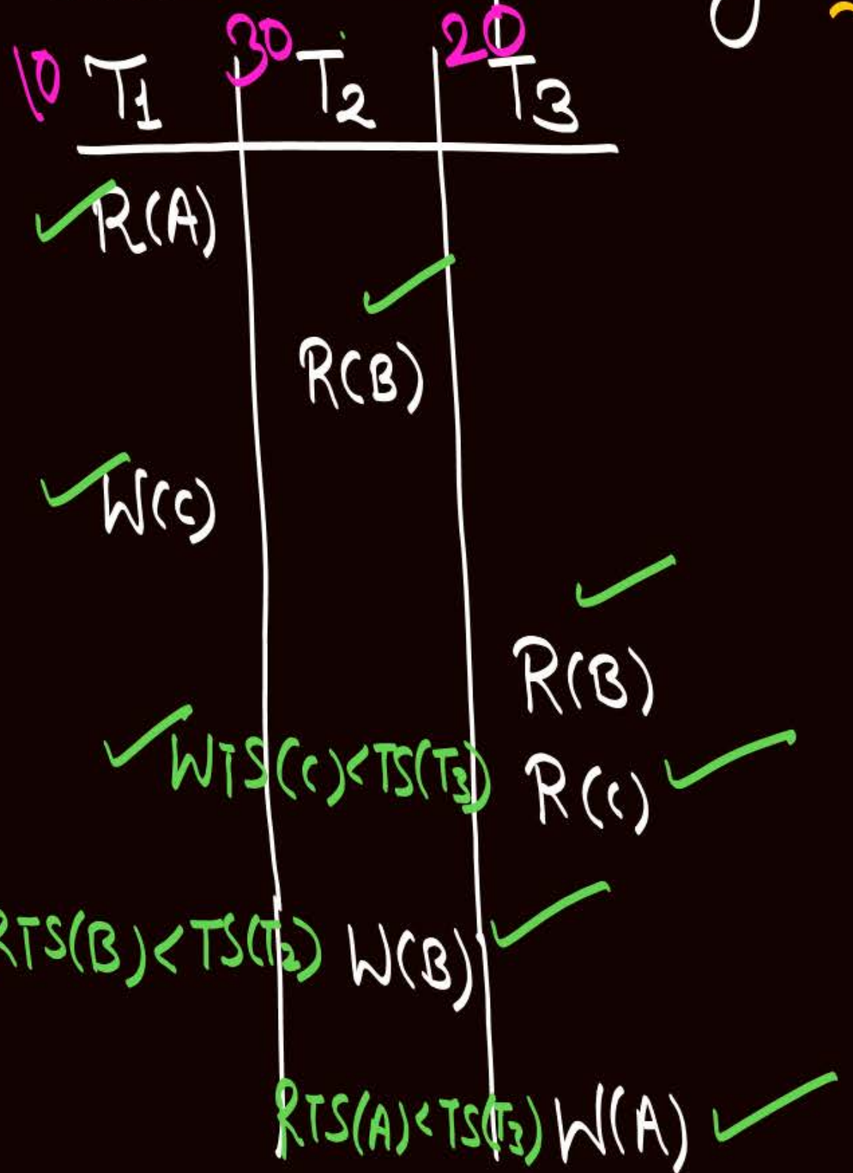
a) $(TS(T_1), TS(T_2), TS(T_3)) = (10, 20, 30)$

b) $(TS(T_1), TS(T_2), TS(T_3)) = (20, 10, 30)$

c) $(TS(T_1), TS(T_2), TS(T_3)) = (30, 10, 20)$

d) $(TS(T_1), TS(T_2), TS(T_3)) = (10, 30, 20)$

Q: Consider the following schedule



a) $(TS(T_1), TS(T_2), TS(T_3)) = (10, 20, 30)$

b) $(TS(T_1), TS(T_2), TS(T_3)) = (20, 10, 30)$

c) $(TS(T_1), TS(T_2), TS(T_3)) = (30, 10, 20)$

d) $(TS(T_1), TS(T_2), TS(T_3)) = (10, 30, 20)$

Note:- ① A schedule is allowed to execute using B.T.S.O.P if and only if schedule is a Conflict serializable schedule And Conflict equivalent serial schedule is based on time stamp ordering of transactions.

② If schedule is not a C.S.S. then it is never allowed to execute using B.T.S.O.P.

③ If schedule is C.S.S., but Conflict equivalent serial schedule is not based on Time stamp ordering of transactions then it is not allowed by B.T.S.O.P.

Note: B.T.S.O.P. Covers only Conflict serializable schedules, but there are many other serializable schedules that are not Conflict serializable but "view serializable"

∴ A modified version of Time stamp ordering Protocol is defined with Thomas write Rule

Topic : Time stamp ordering protocol with Thomas write rule

Let T_1 & T_2 are two transactions s.t. $TS(T_1) < TS(T_2)$

① When T_1 issue a R(A) opn :-

ⓐ

10 T_1	T_2 20
	R(A)
R(A)	

ⓑ

10 T_1	T_2 20
	W(A)
R(A)	

② When T_1 issue a W(A) opn

ⓒ

10 T_1	T_2 20
	R(A)
W(A)	

ⓓ

10 T_1	T_2 20
	W(A)
W(A)	

← These three cases will be same for B.T.S.O.P. as well as Time stamp ordering Protocol with Thomas write Rule

← This case will be different from B.T.S.O.P. in Time stamp ordering Protocol with Thomas write Rule

Time stamp ordering Protocol With Thomas Write Rule

i.e. T_1 is old, T_2 is new

Let T_1 & T_2 are two transactions and $TS(T_1) < TS(T_2)$

(1) When transaction T_1 issue a Read(A) opⁿ :.

(a)

(10) T_1	(20) T_2
	R(A)
R(A)	

time ↓

$TS(T_1) < TS(T_2)$
∴ Time stamp ordering is $T_1 \rightarrow T_2$

If T_1 is allowed to perform this R(A) opⁿ then also the behaviour of schedule will be view equivalent to serial schedule based on time stamp ordering {i.e. $T_1 \rightarrow T_2$}
∴ Transaction T_1 is allowed to perform R(A) opⁿ

(b)

(10) T_1	(20) T_2
	W(A)
R(A)	

Rollback T_1

$TS(T_1) < TS(T_2)$
∴ Time stamp ordering is $T_1 \rightarrow T_2$

If T_1 is allowed to perform this R(A) opⁿ then behaviour of schedule will not be view equivalent to serial schedule based on Time Stamp ordering {i.e $T_1 \rightarrow T_2$}
∴ Transaction T_1 is not allowed to perform this R(A) opⁿ. Hence Rollback T_1

if $WTS(A) > TS(T_1)$, then Rollback T_1.

Reader only need to worry about writer

Time stamp ordering Protocol with Thomas Write Rule:-

i.e. T_1 is old, T_2 is new

Let T_1 & T_2 are two transactions and $TS(T_1) < TS(T_2)$

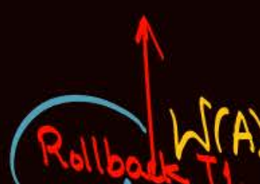
② When transaction T_1 issue a Write(A) opⁿ:.

ⓒ

⑩ T_1	⑳ T_2
	R(A)

∴ $TS(T_1) < TS(T_2)$
∴ Time stamp ordering is
$T_1 \rightarrow T_2$

time ↓



If T_1 is allowed to perform this W(A) opⁿ then behaviour of schedule will not be view equivalent to serial schedule 'based on Time Stamp ordering' {i.e $T_1 \rightarrow T_2$}

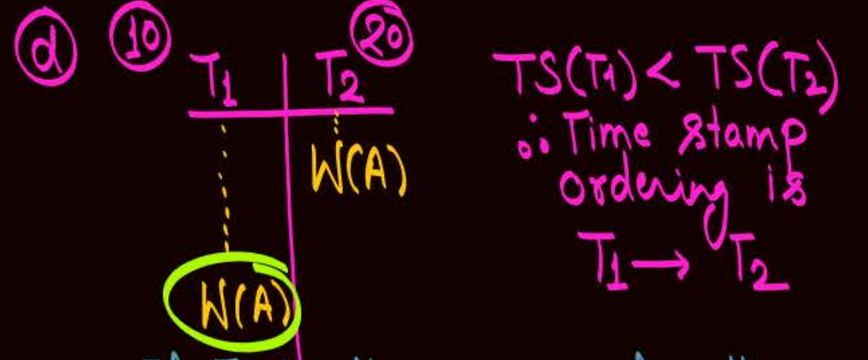
∴ Transaction T_1 is not allowed to perform this W(A) opⁿ. Hence Rollback T_1

if $RTS(A) > TS(T_1)$ then Rollback T_1

Time stamp ordering Protocol with Thomas Write Rule:-

Let T_1 & T_2 are two transactions and $TS(T_1) < TS(T_2)$ i.e. T_1 is old, T_2 is new

② When transaction T_1 issue a Write(A) opⁿ:.



$TS(T_1) < TS(T_2)$
∴ Time stamp ordering is $T_1 \rightarrow T_2$

If T_1 is allowed to perform this W(A) opⁿ then behaviour of schedule will not be View equivalent to serial schedule 'based' on Time Stamp ordering. i.e $T_1 \rightarrow T_2$?

But if we skip this W(A) opⁿ of transaction T_1, then schedule produced wrt. Thomas Write Rule will have the same behaviour as View equivalent serial schedule based on Time stamp ordering of transactions

∴ if transaction T_1 issue a W(A) opⁿ and $WTS(A) > TS(T_1)$, then skip this W(A) opⁿ by Transaction T_1, and Continue with Remaining opⁿ of transaction T_1

Time stamp Ordering Protocol with Thomas Write Rule Conditions :-

① When T_1 issue a Read(A) opn

(i) If $WTS(A) > TS(T_1)$
then Rollback T_1

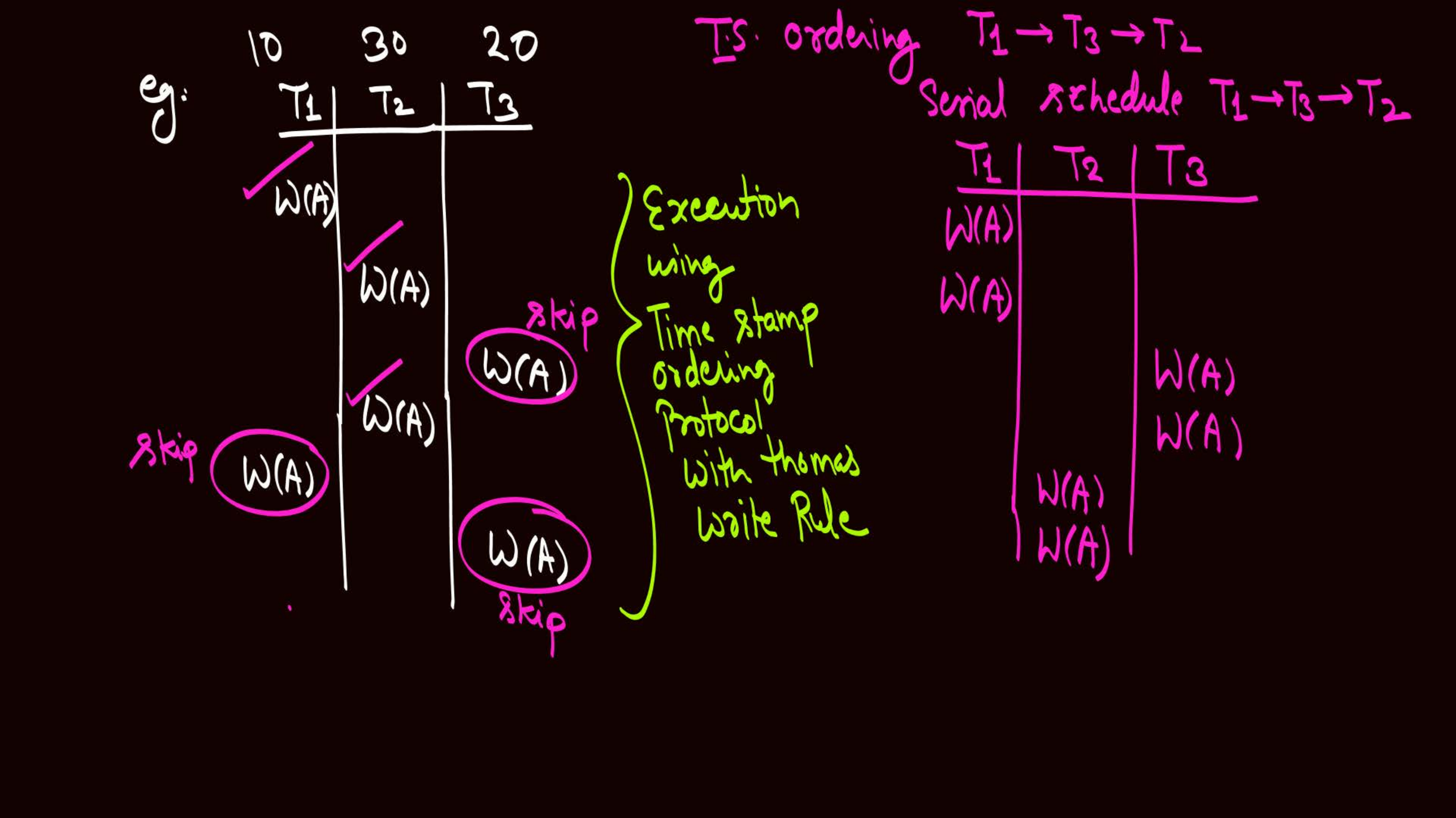
(ii) otherwise,
T_1 is allowed to perform R(A) opn,
∴ T_1 will perform Read(A) opn
and set $RTS(A) = Max(RTS(A), TS(T_1))$

② When T_1 issue a Write(A) opn

(i) If $RTS(A) > TS(T_1)$,
then Rollback T_1

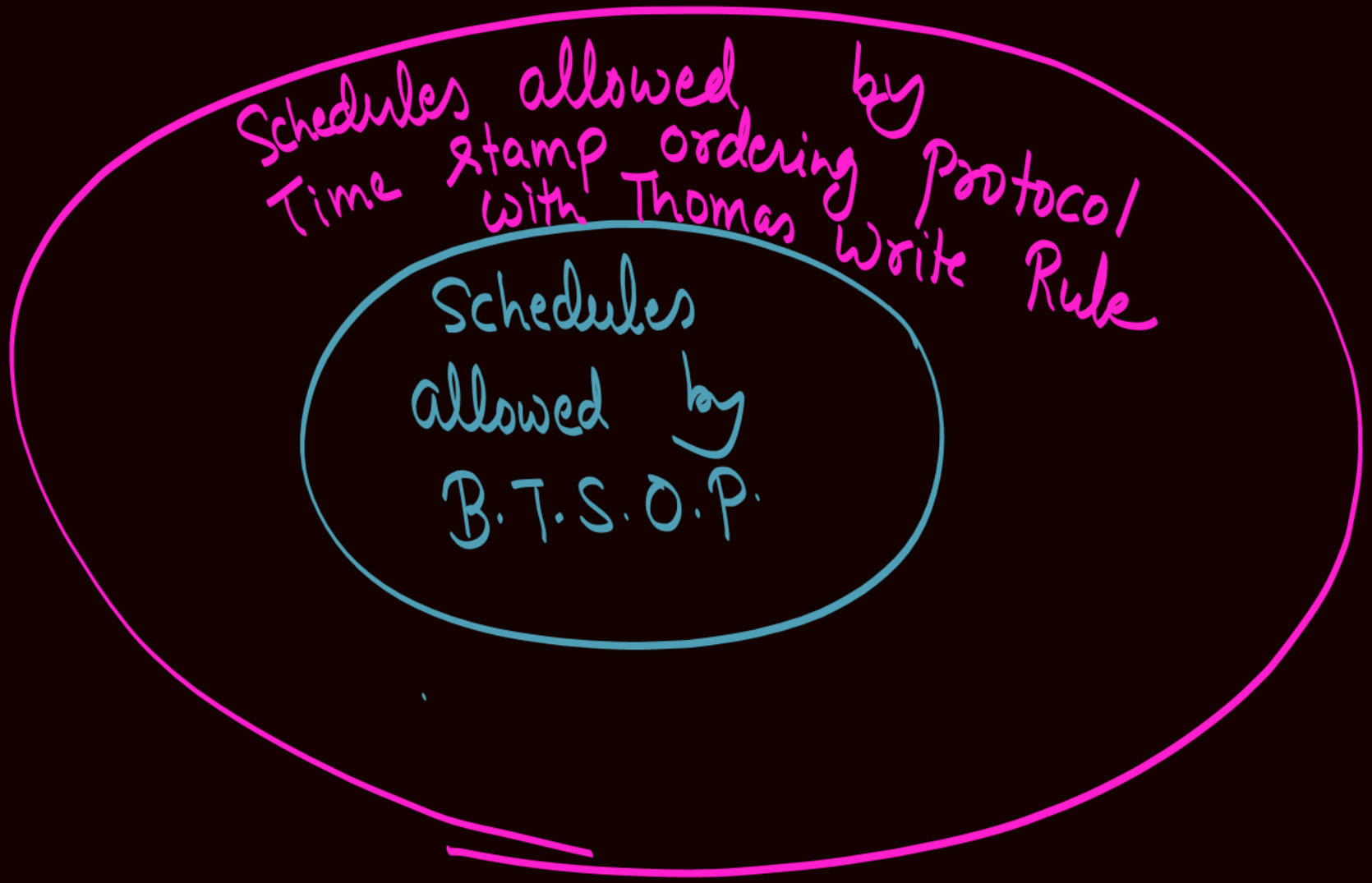
(ii) If $WTS(A) > TS(T_1)$, then
skip that W(A) opn of transaction T_1,
And Continue with Remaining
operation of T_1.

(iii) otherwise {i.e. when $RTS(A) \leq TS(T_1)$ & $WTS(A) \leq TS(T_1)$}
T_1 is allowed to perform W(A) opn
∴ T_1 will perform W(A) opn
and set $WTS(A) = TS(T_1)$


10
30
20
eg:
T1
T2
T3
W(A)
W(A)
W(A)
skip
W(A)
skip
W(A)
W(A)
skip
Execution using Time stamp ordering Protocol with thomes Write Rule
T.S. ordering T1 → T3 → T2
Serial schedule T1 → T3 → T2
T1
T2
T3
W(A)
W(A)
W(A)
W(A)
W(A)
W(A)

Note:- If the schedule is allowed by Time stamp ordering protocol with Thomas Write Rule, then schedule produced by Thomas write Rule will be View equivalent to serial schedule based on time stamp ordering of transactions.

- If schedule is not a view serializable schedule then it is never allowed using Time stamp ordering protocol with Thomas Write Rule.
- If schedule is allowed by Time stamp ordering protocol with Thomas Write Rule, then the schedule is a view serializable schedule


Schedules allowed by Time stamp ordering protocol with Thomas Write Rule
Schedules allowed by B.T.S.O.P.

① → Time stamp ordering protocols are free from deadlock, but starvation is Possible.

② Schedules allowed by Time stamp ordering protocol may suffer from irrecoverability, Cascading Rollback and lost update problem., to avoid these problems We can use "Strict time stamp ordering protocol".

Note:- Time stamp of the transactions can be used to avoid deadlock from lock based Protocols.

There are two protocols which can be used to avoid deadlock in lock based protocols.

1. Wait-die Protocol
2. Wound-wait Protocol

2 mins Summary

Topic Basic time stamp ordering protocol

Topic Time stamp ordering protocol with Thomas write rule

Slide

THANK - YOU

Slide