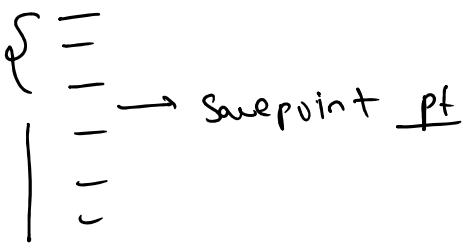
PL SQL : DML Insert Update delete select

*

Commit

Roll back



Insert (1)

Insert (2)

Commit

Insert (3)

savepoint p'

In (4)

In (5)

In (6)

Roll back to p'

In (7)

In (8)

commit

1
2
3
7
8

Advantages : Multitasking

waiting time ↓

Resource Utilization Efficiency ↑

Resource Utilization & Efficiency

Resource Consumption

Faults

① Dirty Read

A → 50 60

T_1 → 60

T_2 → ~~60~~ 70

T_2 → 70

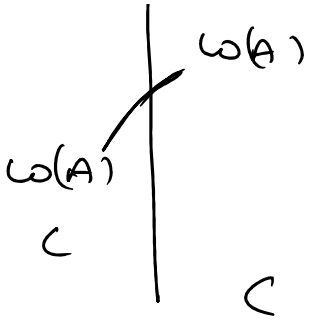
T_1	T_2
R(A)	
W(A)	
⋮	R(A)
⋮	W(A)
⋮	C
C	

② Phantom Read

T_1	T_2
R(A)	
	R(A)
Del(A)	
	R(A)

③ Lost Update

T_1	T_2
R(A)	
	W(A)



T_1 → R
W
R

T_2 → W
ADD
R

⇒ | T_2 W | T_1 R | T_1 W | T_2 ADD | T_1 R | T_2 R

↓
Schedule

→ Serial →

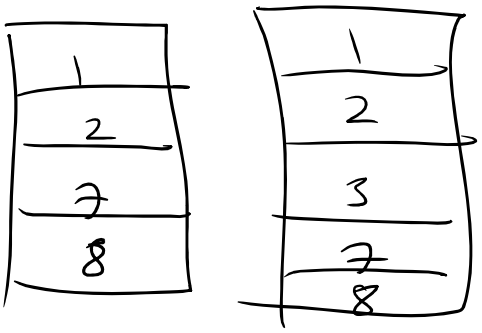
→ Non - serial

DML → Insert update delete select

Commit
Rollback

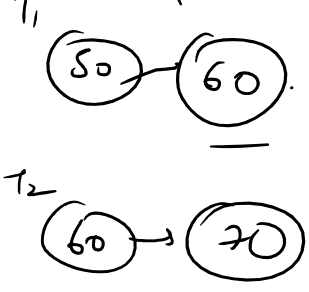
—
—
—
— → save points pl

Insert(1)
Insert(2)
Commit
Insert(3)
Savepoint P1
Insert(4)
Insert(5)
Insert(6)
Roll back / Rollback to P1
Insert(7)
Insert(8) → commit



Concurrency : Multitasking
wait time ↓
Resour Utilization Efficiency
Resource Consumption ↑
Faults

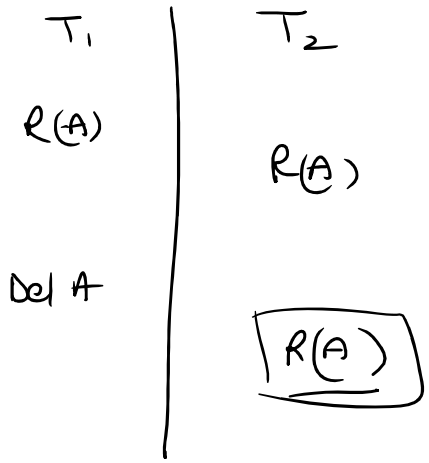
Dirty Read



T_1	T_2
R(A)	
w(A)	
	R(A)
	w(A)
	C
C	

C

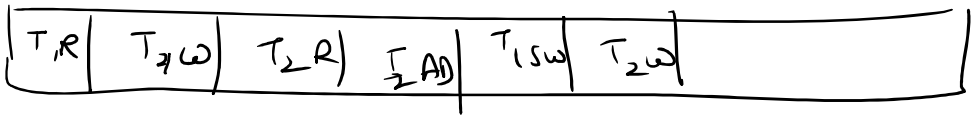
Phantom Read



lost update

$T_1 \rightarrow$ R, W, SUB

$T_2 \rightarrow$ R, ADD, W



Schedule

1) Serial
2) Non-serial

PL-SQL : DML : select Insert update delete

Transaction : Read write commit Rollback

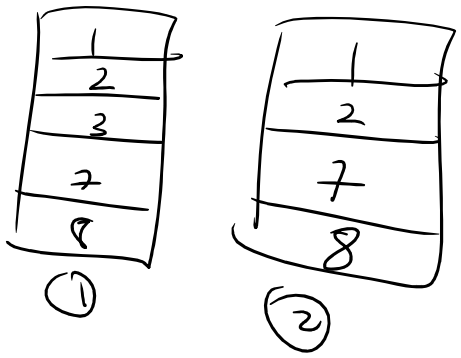
Commit
Rollback

Savepoint P1

Roll back to p1

Insert (1)
Insert (2)
Commit
Insert (3)
Savepoint p1
Insert (4)
Insert (5)
Insert (6)

① Roll back to p1 / ② Roll back
Insert (7)
Insert (8), commit



Concurrency Advantages : 1) Multitasking
2) Efficient use of Resources
3) Waiting time / d

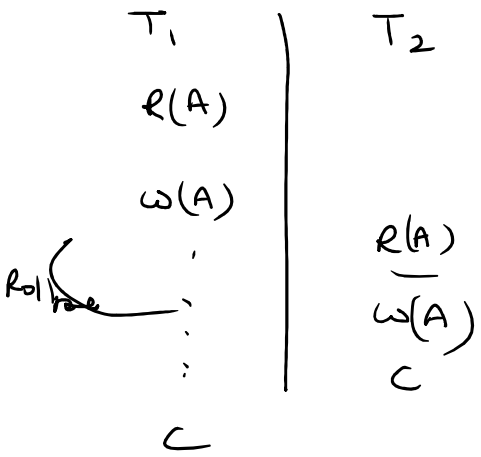
Dis
4) Resources
5) Faults

Faults

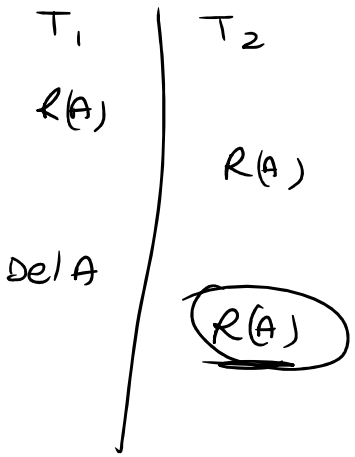
Dirty Read

T_1 A → ~~80~~ 60

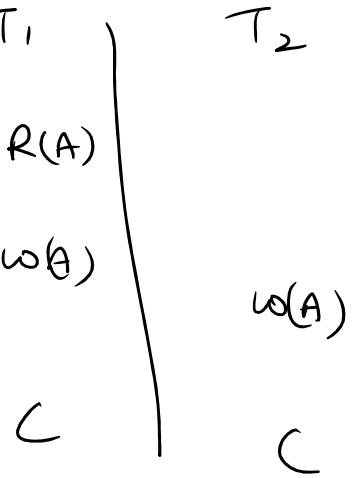
T_2 A → ~~60~~ 70



Phantom Read

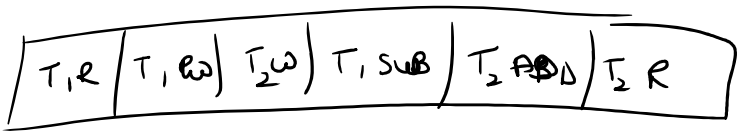


lost update



- T_1 → R
  W
  SUB

  T_2 → W
  ADD
  R



Schedule

1) Serial →

2) Non serial →