

## Transaction

Constraint:

$$\underline{A = B}$$

1.  $A := 2 \times A$       *here  $A \neq B$  (unless both zero)*
2.  $B := 2 \times B$

Big assumption:

In executing an individual transaction, the DBMS  
never swaps its

individual operations  
 (and never runs them in parallel)



Ex 1:

Constraint:

$$0 \leq A^{\text{old}} \leq B^{\text{old}}$$

A old vsA new

T1:

$$1. A^{\text{new}} := A^{\text{old}} + B^{\text{old}}$$

$$2. B^{\text{new}} := \underline{A^{\text{new}}} + B^{\text{old}}$$

(Same for  
B)

$$0 \leq \underline{A^{\text{new}}} \leq \underline{B^{\text{new}}}$$

$$0 \leq A^{\text{old}} + B^{\text{old}}$$

satisfied

$$\leq (A^{\text{old}} + B^{\text{old}}) + \overbrace{B^{\text{old}}}$$

satisfied

Proved  
correct

Ex 2: Constraint:  
 $0 \leq \underline{A^{\text{old}}} \leq B^{\text{old}}$

- T2:
1.  $A^{\text{new}} := 2A^{\text{old}}$
  2.  $B^{\text{new}} := B^{\text{old}} + 1$

$$0 \leq \overset{?}{A^{\text{new}}} \leq \overset{?}{B^{\text{new}}}$$

$$0 \leq \overset{?}{2A^{\text{old}}} \leq \overset{?}{B^{\text{old}} + 1}$$

↑ if  $A^{\text{old}} = 100$  and  $B^{\text{old}} = 100$   
 satisfied ⇒ context

$\Rightarrow T2$  is not correct



From now on:  
assume all individual  
actions to be correct  
with respect to the  
given constraints  
— see def on slide 8



Logging: a way to assure  
that transactions are  
atomic

Log ≡ sequence of log  
records

Recovery:

- repair effects of crash
- = some transactions will be "undone"
- = others will be "redone"



Undo logging

- <START T>
- old value
- 1 <T, A, 10> ← Rule:  
2 <T, B, 20> All log records  
3 <COMMIT T> are always flushed  
in order
- 

LA - flushing log record 1

LB - flushing log record 2

A - output(A) ] are done by  
B - output(B) ] different module  
C - flushing log record 3 can happen in  
any order in



$$\text{LA} < \text{LB} < \text{LC}$$

$$\text{LA} < A$$

$$\text{LB} < B$$

$$A < C$$

$$B < C$$

$\times \text{ LA} < A < \text{LB}$

$$\text{LA} < \text{LB} < A < B < C$$

$$\text{LA} < \text{LB} < B < A < C$$

$$\text{LA} < A < \text{LB} < B < C$$

$\text{C} > \text{B}$  ← violates  
undo  
logging  
rules



We never see

A or B in the log  
space on disk

$\Rightarrow$  see just (some of)  
 $\langle A, B, C \rangle$

---

If we could see in the log

(a)  $\langle A, B, C \rangle$

$\Rightarrow$  T has successfully  
persisted its new  
values of A and B

on disk  $\Rightarrow$  need to do  
nothing in  
recovery



- (b) LA, LB (and not C)
- ⇒ do not know if  
either A or B is old  
or new on disk
- ⇒ T needs to be  
undone or  
redone
- (only undone under  
undo logging,  
because the log has  
only old values)



undo  
logging:

$\langle \text{START } T \rangle$

$\langle T, A, 10 \rangle$

$\langle \text{START } S \rangle$

$\langle S, C, 30 \rangle$

$\langle T, B, 20 \rangle$

$\langle \text{COMMIT } S \rangle$

$\langle \text{START } U \rangle$

$\langle U, C, 40 \rangle$

$\langle U, C, 50 \rangle$

$\langle \text{COMMIT } T \rangle$

time in  
recovery  
 $\uparrow$   
 $\Leftarrow$  nothing  
 $\Leftarrow$  nothing  
 $\Leftarrow$  nothing

$\Leftarrow$  nothing

$\Leftarrow$  nothing

$\Leftarrow$  do nothing  
about S

$\Leftarrow$  done with  
U

$\Leftarrow$  undo Y:  
 $C := 40$

$\Leftarrow$  undo U:  
 $C := 50$

$\Leftarrow$  do nothing  
about T

