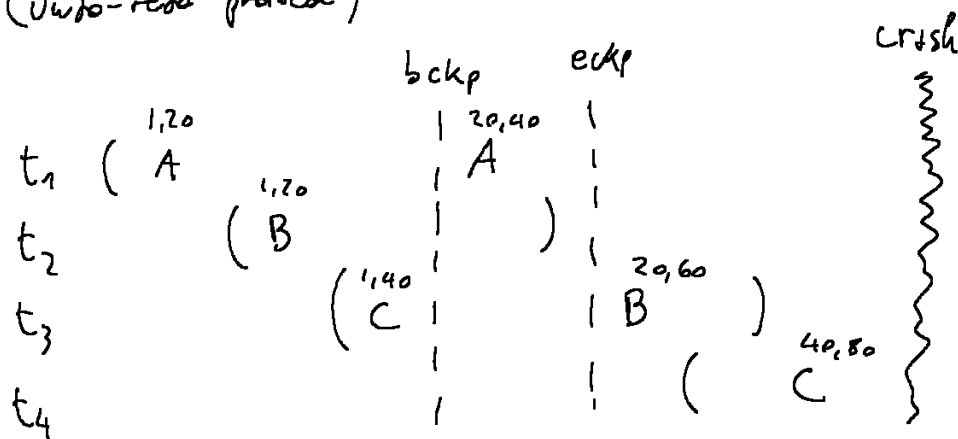


# Ex - Transactions - last lecture

(undo-redo protocol)

Checkpoint  
means flushing  
dirty pages  
to disk



2) before log (PS)

A = 1  
B = 1  
C = 1

b) crash time  
(buffer)

A = 40  
B = 60  
C = 80

(PS)

A = 20  
B = 60  
C = 40

before we do not  
know what is  
flushed during  
checkpoint

correcting

(PS at crash time)

A = 20  $\leadsto$  for sure because  $(t_1, A, 1, 20)$  is before begin-ckp and has been flushed  
40  $\leadsto$  not sure, between begin-ckp and end-ckp  
 $\Rightarrow$  A = 20 or 40 ( $t_1, A, 20, 40$ ) may or may not be flushed

B = 1  $\leadsto$  no, because  $\leadsto$

20  $\leadsto$   $(t_2, B, 1, 20)$  has been flushed during checkpoint (guarantee by checkpoint)

60  $\leadsto$  it is possible, because the buffer manager is free to buffer whenever it likes

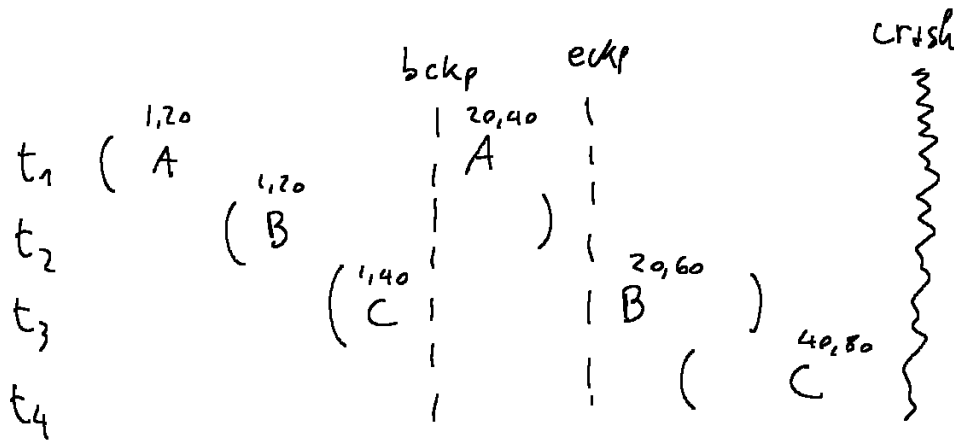
$\Rightarrow$  B = 20 or 60

C = 40  $\leadsto$  flushed because before ckp

80  $\leadsto$  not sure, because after ckp

$\Rightarrow$  C = 40 or 80

(continue)



c) restarting  
 (undo  $\rightarrow$  no commit  
 backends, atomicity)

$t_1, t_4$

(redo  $\rightarrow$  trace and commit  
 forward, durability)

$t_2, t_3 \rightarrow$  observe that  $t_2$  happened before ckp so it is the same

d) undo  $\rightarrow$   $t_4, C, 40, 80$   
 $t_1, A, 20, 40$   
 $t_2, A, 1, 20$

e) redo  $\rightarrow$   ~~$t_2, B, 1, 20$   
 $t_3, C, 1, 40$   
 $t_3, B, 20, 60$~~

(start from begin-ckpt)

redo  $\rightarrow t_3, B, 20, 60$

f) after restart

$A = 1$   
 $B = 60$   
 $C = 40$

(just look at  
 undo and  
 redo operations)

g) undo and redo are executed  
 in the buffer, and it may or may not  
 be flushed in the PS

h) after restart, PS?

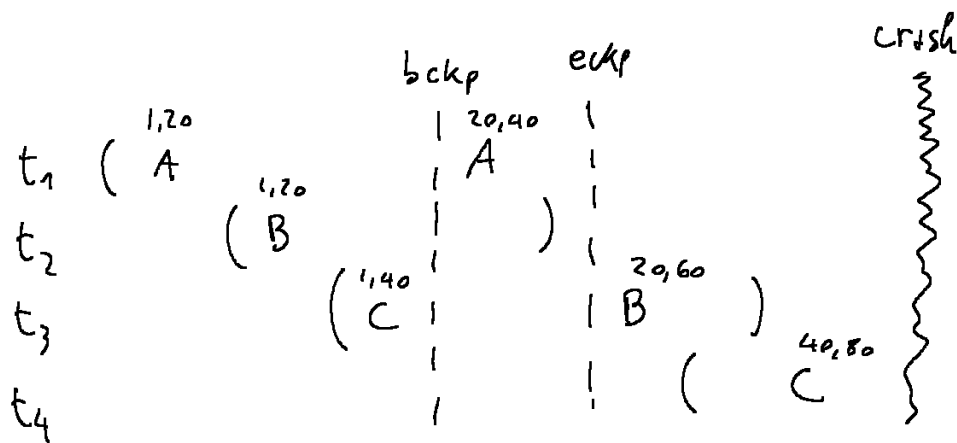
$A = 1 \rightarrow$  possible  
 $20 \rightarrow$  possible  
 $40 \rightarrow$  possible

$B = 20 \rightarrow$  possible  
 $60 \rightarrow$  possible

$C =$

$\rightarrow$  we don't know if buffer  
 already flushed  
 after restart

Now, with the NO-UNDO Protocol.



It means that every transaction is pinned and it is unpinned only for commit. No risk of flushing uncommitted transactions means no need to undo.

So  $t_1$  and  $t_4$  are still pinned and therefore lost.

While  $t_2$  and  $t_3$  are committed and therefore they were maybe flushed. A redo is necessary.

What about the checkpoint?

If a page is pinned, it cannot be flushed during the checkpoint, meaning that the autonomy of the buffer is limited.

(crash time PS)

$A = 1$

$B = 1 \text{ or } 20 \text{ or } 60$

$C = 1 \text{ or } 40$

$B$  can be 1 because  $B$  is still pinned before ckp

NO operation is undone

$t_1, t_4$  are redone, the same as before

(after restart BUFFER)

$A = 1$

$B = 60$

$C = 40$

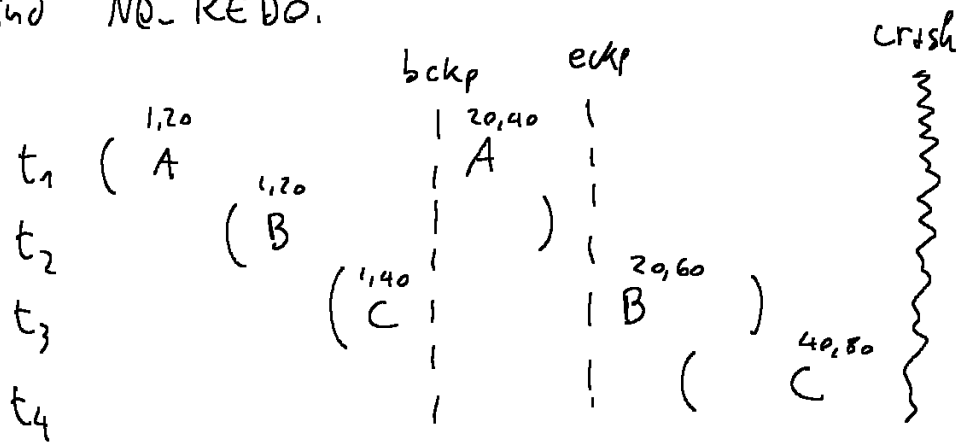
(after restart PS)

$A =$

$B = 1 \text{ or } 20 \text{ or } 60$

$C =$

And NO REDO.



It means that I do NOT pin anything.

Checkpoint is guaranteed to flush.

We force flush all the transactions because we do not want to redo anything.

(crash time PS)

A = 20 or 40

B = 60

C = 40 or 80

No redo

undo  
 $\downarrow$   
 $C, 40, 80$   
 $A, 20, 40$

(after restart)  
 BUFFER

A = 20

B =

C = 40

(after restart)  
 PS

A = 20 or 40

B = 60

C = 40 or 80