

COMP90050 Advanced Database Systems: Tutorial
Winter term, 2023 (Week 4)

Exercises

Part 1

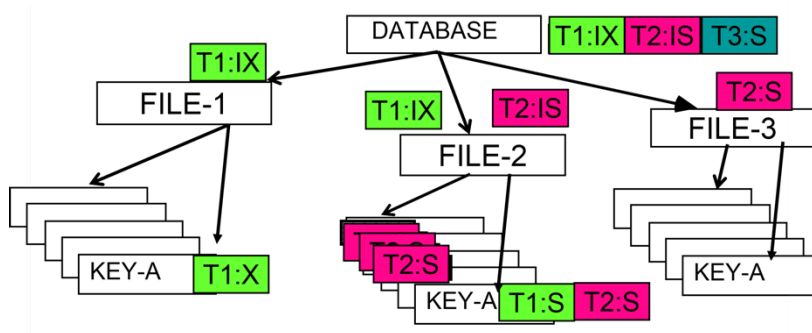
1. Given the operations for a transaction T1 below, please list the lines that this transaction is executing that cannot happen with two-phase locking. Briefly explain.

1	Slock(A)
2	Read(A)
3	Unlock(A)
4	Slock(B)
5	Read(B)
6	Unlock(B)
7	Xlock(C)
8	Write(C)
9	Unlock(C)
10	Xlock(A)
11	Write(A)
12	Unlock(A)

2. Discuss why two-phase locking guarantees serializability.
3. The following transactions are issued in a system at the same time. Answer for both scenarios.
 - (a) Scenario 1: When the value of A is 3, which of the following transactions can run concurrently from the beginning till commit (that is, all operations and locks are compatible to run concurrently with another one) and which ones need to be delayed? Please give explanation for the delayed transactions.
 - (b) Scenario 2: When the value of A is 2, which of the following transactions can run concurrently from the beginning till commit (that is, all operations and locks are compatible to run concurrently with another one) and which ones need to be delayed? Please give explanation for the delayed transactions.

	T2	T3
	Lock (U,A)	Lock (IX,A)
T1	Read A	Read A
Lock (S,A)	if(A ==3) {	if(A ==3){
Read A	Lock(X,A)	Lock(X,A)
Unlock A	Write A	Write A
	}	}
	Unlock A	Unlock A

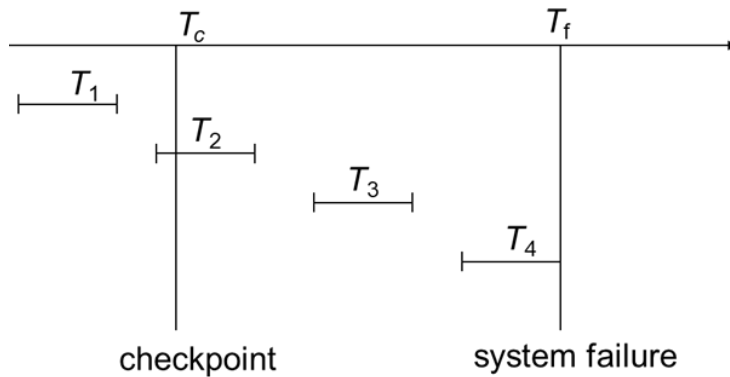
4. Review the concepts of granular locks then answer the following question. Given the hierarchy of database objects and the corresponding granular locks in the following picture, which transactions can run if the transactions arrive in the order T1-T2-T3? What if the order is T3-T2-T1? Note that locks from the same transaction are in the same colour. We assume that the transactions need to take the locks when they start to run.



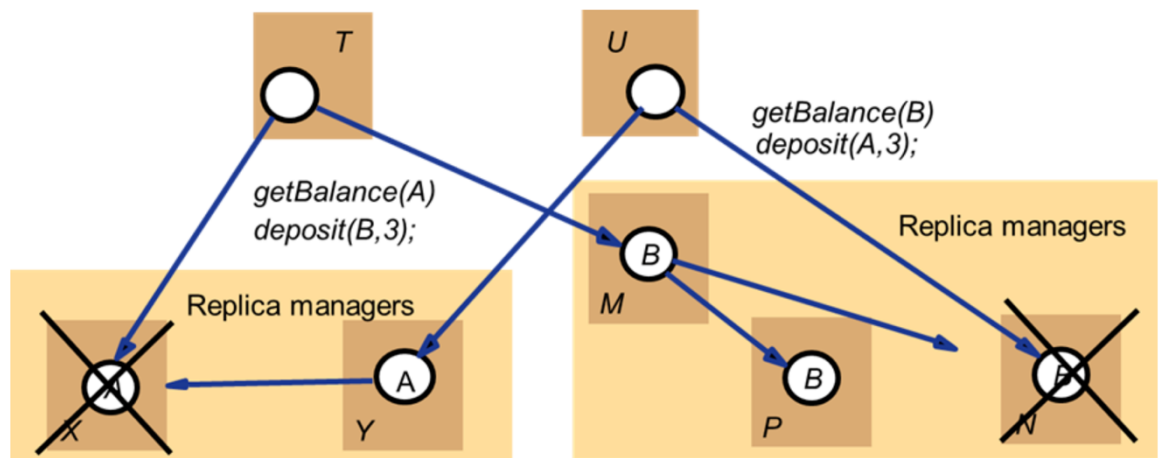
5. With two-phase locking we have already seen a successful strategy that will solve concurrency problems for DBMSs. Then discuss why someone may want to invent something like Optimistic Concurrency control in addition to that locking mechanism.

Part 2

1. In the following figure the first vertical line T_c denotes the point where checkpointing was done and the second on the right, T_f, is where a system crash occurs. Please discuss what would change if the checkpointing was done right at the beginning of each transaction instead of the following case in the figure.



2. Assume a two-phase commit involves a coordinator and three participants, P1, P2 and P3. What would happen in the following scenarios?
 - Scenario 1: P1 and P2 voted yes and P3 voted no.
 - Scenario 2: P2 crashed when it was about to send a vote message to the coordinator.
3. Given the two following transactions T and U that run on replica managers X, Y, M, P, and N, review the problem that would occur if X and N were to crash during execution. Then state the solution that we have seen in the lecture. Discuss what would happen, if rather than X and N becoming unavailable we have the following scenario: If Y were to become unavailable during the execution and right after U accessed A at Y, but X and N do not fail, rest of the assumptions of this scenario is the same as we discussed in class.



For the following examples, the logs are given after a crash. What will be the analysis, redo and undo phases?

Example 1:

Log:

After a crash, we find the following log:

0	BEGIN CHECKPOINT
5	END CHECKPOINT (EMPTY XACT TABLE AND DPT)
10	T1: UPDATE P1 (OLD: YYY NEW: ZZZ)
15	T1: UPDATE P2 (OLD: WWW NEW: XXX)
20	T1: COMMIT

Example 2:

Log:

After a crash, we find the following log:

0	BEGIN CHECKPOINT
5	END CHECKPOINT (EMPTY XACT TABLE AND DPT)
10	T1: UPDATE P1 (OLD: YYY NEW: ZZZ)
15	T1: UPDATE P2 (OLD: WWW NEW: XXX)
20	T2: UPDATE P3 (OLD: UUU NEW: VVV)
25	T1: COMMIT
30	T2: UPDATE P1 (OLD: ZZZ NEW: TTT)

Example 3:

Log:

After a crash, we find the following log:

10	T1: UPDATE P1 (OLD: YYY NEW: ZZZ)
15	T2: UPDATE P3 (OLD: UUU NEW: VVV)
20	BEGIN CHECKPOINT
25	END CHECKPOINT (XACT TABLE=[[T1,10],[T2,20]]; DPT=[[P1,10],[P2,15]])
30	T1: UPDATE P2 (OLD: WWW NEW: XXX)

35	T1: COMMIT
40	T2: UPDATE P1 (OLD: ZZZ NEW: TTT)
45	T2: ABORT
50	T2: CLR P1(ZZZ), undonextLSN=15

----- Discussion, Q/A on any topics of the subject -----