Exercises *Databases*
Session 6: recovery & two-phase-commit

Hans Philippi, Lennart Herlaar, Ad Feelders

March 17, 2017

recovery

Exercise 35

After a crash, the log file has to be scanned to resolve the actual database state. In what order should the UNDOs be executed? In what order the REDOs?

Exercise 36

Some thoughts concerning the recovery mechanism.

(a) The REDO recovery algorithm enables the buffer manager to flush dirty pages after the commit. The goal is to gain efficiency with respect to IO. On the other hand, the data manager has the obligation to log updates before commitment. At first sight, this seems to be contradictory. Explain why it is not.

(b) Is the following claim correct: "Immediately after a recovery action with REDO, the external memory of the database is up to date."

Exercise 37

Describe the rules for combined UNDO/REDO logging. Describe the liberty that the buffer manager has with respect to the flushing of data related to the moment of commitment.

Exercise 38

Describe how logging can be used to solve the media failure problem (i.e. crash of the data disk).

two-phase-commit

Exercise 39

Suppose we have a network partition, and in both parts of the partition, the CTP is started. Could this lead to inconsistency?

Exercise 40

When a participant votes READY, he logs his vote before sending it to the COORD. Can you explain why? Is this also necessary when he votes ABORT?