UMEÅ UNIVERSITY, DEPARTMENT OF COMPUTING SCIENCE

Written Assignment 3 (40 points)

Distributed Systems - 5DV147

You need to complete and submit this assignment individually. Collaboration is permitted only as described in the Guidelines for compulsory assignments and the Honor code. You need to submit this assignment by January 7 at 23:59 (hard deadline). You can either email your solution to us (5dv147-staff(at)cs.umu.se) or drop off your solution in the box outside the department.

1 CONSISTENCY (4 POINTS)

- Explain in your own words what is the reason for considering weak consistency models. (**2 points**)
- It is often argued that weak consistency models impose an extra burden for programmers. To what extent is this statement actually true? (**2 points**)

2 REPLICATION (8 POINTS)

During class we studied three correctness criteria for replicated objects:

- "Basic" correctness: The interleaved sequence of operations must meet the specification of a single correct copy of the object(s).
- Sequential consistency: The order of operations in the interleaving must be consistent with the program order in which each individual process executed them.
- Linearizability: The order of operations in the interleaving must be consistent with the real times at which operations occurred in the actual execution.

We also studied two replication models:

- Passive replication, in which there is a *primary* replica manager and several backup replicas. If the primary fails any backup can take its place through an election.
- Active replication, in which operations are performed independently and identically at multiple replica managers.

1. Explain why allowing backup replicas to process *read* operations leads to sequentially consistent rather than linearizable executions in a passive replication system. (**4 points**)
2. In general for active replication to work, it is necessary that all operations are carried out in the same order at each replica. Is this ordering always necessary? (**4 points**)

3 QUORUMS (14 POINTS)

A quorum-based replica management system consisting of **N** servers engages only a designated fraction of replicas for every read and write operation. These are known as read quorum **(R)** and write quorum **(W)**.

1. Suppose that a data object is replicated on 10 servers. List all the combinations of read quorum **R** and write quorum **W** that are permitted. (**2 Point)**
2. Remember that in sequential consistency all writes must be seen in the same order by all processes. Assuming that **N** is an even number,
   - Will sequential consistency be satisfied if **W = N/2, R = N/2**? (Justify your answer) (**3 Point)**
   - Will sequential consistency be satisfied if **W = (N/2) + 1, R = (N/2) + 1**? (Justify your answer) (**3 Point)**
   - If **N = 10, W = 7**, then what should be the minimum value of **R** so that sequential consistency is satisfied? (**3 Point)**
3. Consider a quorum system with **N=10** replicas. In one implementation, the read quorum is **R=1** and the write quorum is **W=10**. Another implementation uses **R=10** and **W=1**. Are these two implementations equivalent? For what reason will you chose one over the other one? (Explain) (**3 Point)**

4 OPTIMISTIC CONCURRENCY CONTROL (8 POINTS)

A server manages the objects $a_1, a_2, ..., a_n$. The server provides two operations for its clients:

read(i) returns the value of a_1

write(i, Value) assigns *Value* to a_1

Transactions T and U are defined as follows:

T: x = read(i); write(j, 44);
U: write(i,55); write(j,66);

Consider optimistic concurrency control as applied to transactions T and U. Suppose that transactions T and U are active at the same time as one another. Describe the outcome in each of the following cases:

- T's request to commit comes first and backward validation is used.
- U's request to commit comes first and backward validation is used.
- T's request to commit comes first and forward validation is used.
- U's request to commit comes first and forward validation is used.

In each case describe the sequence in which the operations of T and U are performed, remembering that writes are not carried out until after validation. (**2 points each**)

5 Peer-to-Peer (8 Points)

There are two classes of P2P overlay networks: *structured* and *unstructured*.

1. What are the tasks of an overlay network (in P2P systems)? Give a brief description about the two types of overlays. (**2 points**)
2. Compare the advantages and disadvantages, if any, of the two overlays from the perspectives of nodes (join/leave) and content (insert/search) operations. (**4 points**) Finally, describe how the two classes of overlays are affected by these operations. (**2 points**)