

Excercise Sheet 11

(To be processed until 10.07.2020)

Lecture Distributed Systems Summer Term 2020

Exercise 1: Replication

- a) Explain briefly why the replicas are used.
- b) Name and briefly explain the different types of replicas presented.

Exercise 2: Active Replication

For active replication to work, it is generally necessary that all operations on all replicas are executed in the same order. Is this order always required?

Exercise 3: Quorum Protocols

The failure of network connections or the failure of a gateway computer that connects several networks with each other can result in operational subnets (partitions) whose nodes can still communicate with each other. If replicas of an object are in different subnets and changes are to be made, not all replicas can be reached and inconsistent states can result.

To solve this problem there are among other things the so-called *quorum protocols*, where each replica is assigned a certain weight (a number of votes) beforehand. In order to perform an operation, a certain sum of weights must be collected, i.e. r votes must be present for reading and w votes for writing (read quorum or write quorum).

Assume that there are N replicas in a system. Each replica is assigned a weight of 1. For the following four quorum protocols, justify whether they have a meaningful read or write quorum.

- a) $N = 10, r = 6, w = 5$
- b) $N = 10, r = 3, w = 7$
- c) $N = 10, r = 1, w = N$
- d) $N = 10, r = 5, w = 6$

Exercise 4: Quorum-based Replication

A file is replicated on ten servers. Name all combinations of read quorum and write quorum allowed by the voting algorithm.

Exercise 5: Programming: Consistency Protocols

In the lecture (chapter 8.5) you learned about various consistency protocols, including the *primary-based* protocols. In primary-based protocols, each data element x in memory is associated with a primary server that is responsible for coordinating write operations for x . In addition, it can be distinguished whether the primary copy is on a fixed remote server, or whether write operations can be performed locally by moving the primary copy to the server that initiated the write operation.

- a) Implement a *remote-write* protocol with Java RMI based on the given code in the archive [u11eFiles.zip](#)¹ on the lecture's web page. Extend the method `write()` and introduce new remote methods for the protocol if necessary. Remember that with RMI, calling a remote method always results in a request to the remote server and an associated response (acknowledgement) from the server.

The given client connects to a server (selected via the command line argument) and performs a write and a read operation. If possible, start several instances of the client at the same time, each of which connects to a different server. Test your solution with a different number of servers.

- b) Think about how you would implement a *local-write* protocol. What is the difficulty?

Exercise 6: Any Questions?

Use the opportunity to prepare questions about the exercise sheets that you would like to discuss in the exercise lesson. You will now be able to discuss difficulties in solving the exercises and to explain possible solutions again.

¹<http://www.bs.informatik.uni-siegen.de/web/wismueller/v1/vs/u11eFiles.zip>