Grenoble INP / UJF – Academic year 2015-2016
MOSIG2 – PDES
January 29th, 2016
Teaching staff: V. Quéma, O. Gruber

Distributed Systems
Final examination

Duration: 3 hours
All paper documents allowed, no electronic devices

Answers can be written in English or French.

This exam is made of two parts but all questions are independent. You can work on these parts in the order of your choice, but you must use two sheets of paper for the two parts of the exam.

Please be mindful of the entire question before you start writing down your answer. In particular, use a similar wording and notations when possible. Be reminded that the length of a question is no testimony of its difficulty. Quite the contrary, a slightly longer description might be necessary for the clarity of the question.

Also, please try to respect the proposed length (approximate number of lines) for your answers. Focus your answer, be concise and clear. Being fuzzy will not get you extra points, quite the contrary.

The number of points indicated per question is only provided to give you an idea of the importance of a question. We reserve the right to change the exact number of points.

You must respect the following instructions:

- Write down your name on each sheet and number it if you use several sheets
- Hand back this question sheet with your own answer sheets

PART 1 (8 points):

Question 1 (2 points, 10 lines)
With an active replication scheme, all requests from clients must be totally ordered on all servers. First, explain why it must be so. Second, propose a solution where the total order is decided amongst the group of servers. Your solution must tolerate failures of clients and servers (with the usual assumption that at least one server is still standing). Your solution must be a practical solution, a solution that would work in the real world, with a high churn over thousands of clients.

Question 2 (4 points, 15-20 lines)
You just joined a new team and you are given the following task: ensure the atomicity of an existing multicast. The multicast is pretty simple, each peer uses a simple loop to send each message to all members of the static group. Your boss is asking the following

- The group must become dynamic, peers may voluntarily leave or join at any time.
- The multicast must be evolved into a view synchronous multicast.

Your boss requests that the core of the multicast remain the same, each peer will keep sending messages to all the other peers, using a loop. He also requests that the view mechanism be built using the technology of a ring and tokens.

Question 3 (2 points, 10 lines)
Your team has been charged with the design of a fault-tolerant subsystem for an airplane, based on active replication. You are given compute modules, as a black-box design, that cannot be modified. Each module computes numerical values, based on its inputs. But, you are told that the compute modules are not fail-stop, they may produce erroneous values from time to time.

You are asked to use several of these boxes, with an active replication pattern, to build a fault-tolerant subsystem. Propose a complete design and evaluate its limitations, if any.

PART 2 (12 points):

Question 4

Let us assume that 3 machines execute the Paxos protocol. The machines are named P1, P2 et P3.

- Machine P1 uses the following set of sequence numbers: {1, 4, 7, 10, …} and would like to propose value V1.
- Machine P2 uses the following set of sequence numbers: {2, 5, 8, 11, …} and would like to propose value V2.
- Machine P3 uses the following set of sequence numbers: {3, 6, 9, 12, …} and would like to propose value V3.

For each of the following states, answer the following questions:

- Can the system reach that state?
    - o If yes, propose a sequence of PREPARE and ACCEPT messages (together with the answers to these messages) that lead to that state.
- Is the outcome of the consensus known?
    - o If yes, what is the outcome?
    - o If no, what are the possible outcomes?

State S1: P1 accepted (3, V3), P2 accepted (1, V1), P3 accepted (3, V3)
State S2: P1 accepted (1, V1), P2 accepted (2, V2), P3 accepted (3, V3)
State S3: P1 accepted (1, V1), P2 accepted (2, V2), P3 accepted (3, V1)
State S4: P1 accepted (1, V1), P2 accepted (2, V2), P1 accepted (1, V1)

Question 5

Which modifications would you make to the Paxos protocol if processes had access to a perfect failure detector? Motivate your choice (i.e. explain why you would make this (or these) change(s)).

Question 6

During the lectures, we have studied the uniform reliable broadcast abstraction (URB).

A) Propose an algorithm implementing URB under the following assumptions:

- Known number of processes (N)
- Reliable channels
- Crash faults
- Perfect failure detector

Note: You must explain the algorithm that is executed when process crashes occur.

B) Prove that your algorithm ensures validity despite process crashes.

C) Prove that your algorithm ensures uniform agreement despite process crashes.