Algorithms and Distributed Systems 2023/2024 (Lab Six)

MIEI - Integrated Master in Computer Science and Informatics
MEI – Master in Computer Science and Informatics
Specialization block

Nuno Preguiça (nmp@fct.unl.pt)
Alex Davidson (a.davidson@fct.unl.pt)

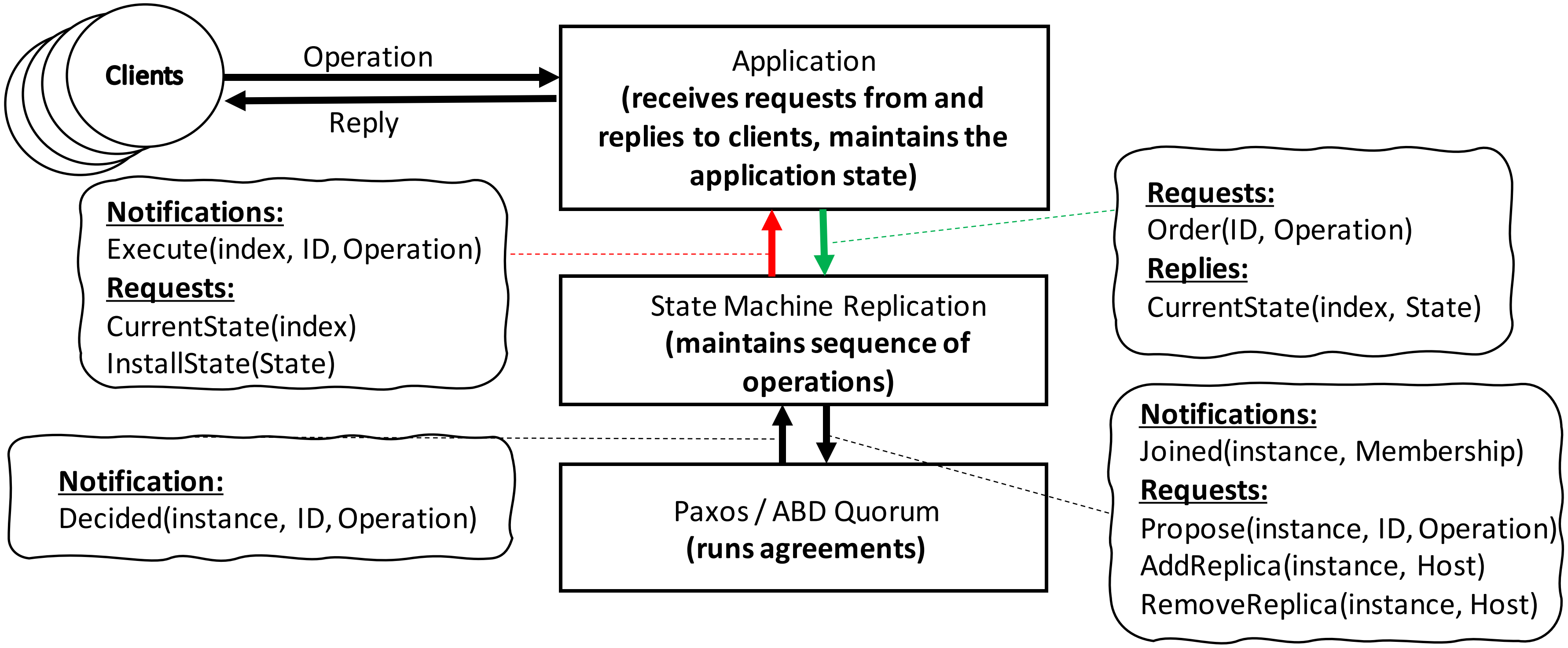
NOVA SCHOOL OF SCIENCE & TECHNOLOGY

Based on slides from João Leitão

Second phase of the project

- **Deadline: 29th November 2023 23:59:59**
- **Link:** https://classroom.github.com/a/PetG1iuo
- **Submission:** Google Form link to be provided.
- **Appendix:**
  - Will be covered in future labs as theory classes catch-up
  - Extra recommendations for the agreement protocol implementations

Overview of Project


Clients
Operation
Reply
Application
(receives requests from and replies to clients, maintains the application state)
State Machine Replication
(maintains sequence of operations)
Paxos / ABD Quorum
(runs agreements)
Notifications:
Execute(index, ID, Operation)
Requests:
CurrentState(index)
InstallState(State)
Requests:
Order(ID, Operation)
Replies:
CurrentState(index, State)
Notification:
Decided(instance, ID, Operation)
Notifications:
Joined(instance, Membership)
Requests:
Propose(instance, ID, Operation)
AddReplica(instance, Host)
RemoveReplica(instance, Host)

Paxos (Agreement protocols)

- Will be covered in theory classes from next week onwards
- The agreement protocol manages the ordering of arbitrary proposals and commands.
- You can focus on the SMR computations initially

State Machine Replication

- Maintains the sequence of commands being executed by the replicated system (the index of the command is used as the instance of the agreement protocol typically).

- Manages the membership of the system and maintains the communication channel:
  - Opens TCP connections (and attempts to reconnect on failure for a maximum number of times).
  - Investigates possible failed replicas.
  - Issues commands to remove failed replicas.
  - If joining a system already executing, acts as a client and requests to join the system directly to another process (by contacting the state machine replication protocol there).

State Machine Replication

- It receives the initial membership of the system as a parameter (if not in the membership it should request to be added).
- The reply to that request should indicate the instance in which the new replica joins the system and the application state in that instance (that must be installed by the new replica).

State Machine Replication

- The implementation can be mostly agnostic to the underlying agreement protocol (Paxos or ABD Quorum), you might need to have an additional parameter to inform it and adjust the behavior:
- All state machine replication protocols propose commands to agreement protocol **instances** to be ordered.
- Both systems will most likely require that when you initialize a new instance you are provided with the current membership of the system.

Agreement

- You will need to implement the Paxos Agreement protocol from scratch (Lecture 6, 8th Nov) will cover this.

- The ABD Quorum agreement protocol can be implemented using same broadcast protocols as defined in phase 01 of the project (e.g. using simple Flood, or something more complex).
    - Lab 05 provides more details on how to implement this

Evaluation

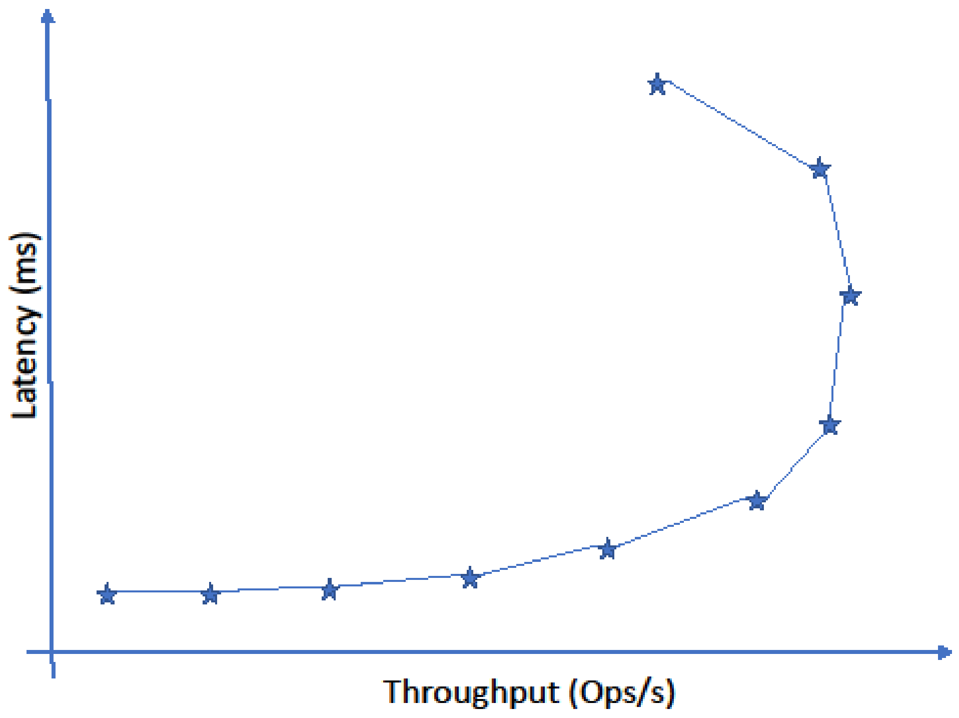
- Evaluate the latency and throughput of your replicated system (using **both** agreement protocols) from the perspective of the clients (emulated by YCSB).
- Evolution of these performance indicators as the number of clients increase (more clients induce more load because clients operate in a closed-loop).
- Each YCSB instance emulated multiple clients by executing multiple threads (each thread represents a client).
- Be careful to avoid saturating clients (there is a limit of threads per YCSB instance that depends on the machine you are using to run it).

Evaluation

- When using multiple YCSB instances simultaneously:
  - Average the latency observed by clients on each instance.
  - Sum the throughput of each YCSB instance.

- Increase the number of clients and plot the observed performance indicators for each system and connect increasing number of clients for each system with a line.

Example of throughput-latency plot



Evaluation

- Evaluation will not cover membership changes or (planned) failures.
- Experiments would ideally consider nodes running on different machines, and clients executing on other machines of the cluster.
  - No Docker script given, due to confusion over scripts last time
- The total number of client threads that you have to use might depend on your implementation (better implementations might need more clients to saturate the system and observe the performance wall).

Appendix: Paxos/ABD Quorum

- You will need to implement agreement via Paxos/ABD Quorum
- Here are some extra details that you should consider.

Agreement: Instances

- In the agreement algorithm, you will need to have multiple instances, one for each position of the sequence of commands in the state machines.
- Suggestion:
  - Have a map in which the key is the Agreement instance, and whose value is a Java class (implemented by you) with all state variables.
  - Messages should always be tagged with the instance identifier.
  - When you process a message you should only modify the agreement state variables for that instance.

Agreement: Membership

- There is no state that is shared between different instances.

- You have also to be careful about membership changes, since the size of quorums used in Paxos and ABD depends on how many replicas you have.

- A replica that is not part of the system should never participate in instances.

Agreement: Paxos (Leaders)

- A process accepts another as a leader when it replies to a prepare message with a prepare ok.
- From that point onward, it cannot process accept messages from a different process in any instance after that one.
- A process becomes leader when it collects a majority of prepare ok messages.

Details

- You might receive messages in agreements for an instance that you have not yet started.
- Be careful, since if you do not know yet what was decided in the previous round, you might have been excluded or a new replica might have joined the system which affects the size of quorums.
- You can process these messages only after you get an initial proposal from your local state machine protocol (but this might require that you have a timer in the state machine to propose special NO-OP operations that have no effect in the system).
- You can also go ahead and process the message immediately, but again you must be careful about membership changes.