École Polytechnique — INF571 Distributed Data structures, with a Spotlight on Blockchains

Message Passing Implementations

We are going to implement a distributed shared memory (i.e. a set of registers with write and read methods) using chain replication and the ABD algorithm.

The provided files implement fully asynchronous replication, i.e., write requests are acknowledged immediately (before they are propagated to the other replicas) and read requests are answered immediately based on the local copy.

Question 1. Run the implementation and observe non-linearizable behaviors (this can be executed with `make; java myDDS/Testing` from the root folder). Note that the implementation may hang. You can try to fix this problem.

Question 2. Modify the implementation in order to use chain replication (replicas are ordered in a list, write requests are sent only to the head and propagated towards the tail, and read requests are sent to the tail). Remember that the channels between replicas should be of type FIFO.

We will ignore failures. But, if you are motivated, you can try to implement the approach described in the lecture. You can model failures by assuming that a Master node sends random STOP commands to randomly chosen replicas.

Question 3. Modify the implementation in order to use the ABD algorithm. The channels between replicas should be of type Bag.

Implementing the above modifications requires an update of the classes `DDS.java` (representing the interaction/network between the different replicas and between replicas and clients) and Replica (representing the internal behavior of a replica).

Test your implementations with multiple combinations of clients (as suggested in the Testing file).