1. Problem overview

In mp1, the task for us is to establish a distributed transaction system that used to keep trace on the balance of the user account in different local servers. And it requires a total ordering multicast during the process of updating the balance. Meanwhile, our program should be able to detect and handle potential failures of some nodes in this distributed system.

2. Protocol Design

2.1 ISIS algorithm for total ordering

The ISIS algorithm works as the following steps:

1) The sender process multicasts message to every nodes in the group;
2) Each alive acceptor replies with proposed priority and store the message in its priority queue;
3) The sender chooses the agreed priority, re-multicasts message to others with its agreed priority;
4) Each acceptor receives the agreed priority as the final priority.

The message class struct is designed to store the message sender, message id, the content of the message, its priority queue, while transmitable and todo_multicast are flags to handle failure detection. The action class struct is designed to process updating the information stored in the message class and the flags to indicate the failure. When each event starts, the information of message and action is updated and threading lock is used to achieve ISIS algorithm.

2.2 R-multicast for Reliable Delivery

To achieve reliable delivery, R-multicast is used in the program. A node will multicast any message to every member (including itself) in the group if this message is not duplicated. If a correct process delivers message m, then all the other correct processes in group will eventually deliver m.

3. Failure Detection Handler

When some failure happens, the other nodes from the sender side will notice this failure through exception thought by the socket. The socket would be closed for the failed node, and change messages' transimitability according to the current node number in the queue. If the message is the feedback message and changes its status to transimitable, which means the node need to decide the agreed priority for this message, the node will mark the message as todo_multicast. When this message is delivered, the node will decide its agreed priority then multicast the agreed priority to other nodes.

4. Run Script Instruction

To run the mp1 program, see the instruction in run_3Nodes.txt and run_8Nodes.txt

5. Performance Evaluation Graph

The Performance Evaluation Graphs are stored inside “./test” files folder