Homework 5

Answer 1

1. Read and Write for both A and B => R = W = 2

|  |  |  |  |
| --- | --- | --- | --- |
| V0 | X | Y | Z |
| A | 300 | 300 | 300 |
| B | 500 | 500 | 500 |

Client has access to X and Y only

read quorum(Nr) = 1+1 = 2 for A and B  
write quorum(Nw) = 1+1 = 2 for B   
  
N = 3,   
Nw + Nr = 4 which is > N(3)  
Nw is greater than N/2.

operation:

client Reads A from X or Y  
client Writes to B at X and Y

|  |  |  |  |
| --- | --- | --- | --- |
| V0 | X | Y | Z |
| A | 300 | 300 | 300 |
| B | 300 | 300 | 500 |

1. Client can access only server Z:   
    read quorum(Nr) = 1 which is <2, so client cannot read B,   
    write quorum(Nw) = 1 which is <2, so client cannot write A,

N = 3,   
Nw + Nr = 2 which is less than N(3)  
Nw is not greater than N/2.  
  
therefore neither operation takes place.

1. since in (a) the client had access only to X and Y so Z could be out of sync

|  |  |  |  |
| --- | --- | --- | --- |
| V1 | X | Y | Z (V0) |
| A | 300 | 300 | 300 |
| B | 300 | 300 | 500 |

Now, the client has access to X and Z and since Z is out of sync, Z will get the latest values of A and B from X

|  |  |  |  |
| --- | --- | --- | --- |
| V1 | X | Y | Z (V1) |
| A | 300 | 300 | 300 |
| B | 300 | 300 | 300 |

read quorum(Nr) = 1+1 = 2 for B  
N = 3,   
Nw + Nr = 4 which is > N(3)   
operation:

client Reads B from X or Z

Answer 2

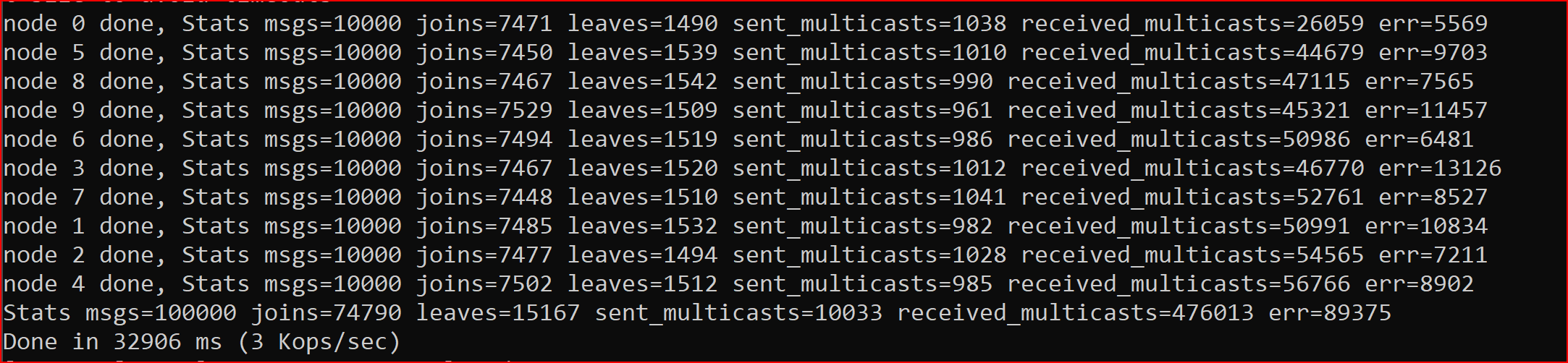
The ACMS acceptance algorithm consists of 2 phases **Replication** and **Agreement**. The **quorum protocol** plays a key role in replication where when a storage point(SP) receives a submission update from a publisher it tries to send the file to as many SPs and when the majority of the SPs acknowledge the request, it establishes a quorum that majority of the storage points are aware of the update and then the **agreement** phase is initiated where the storage point that received the submission update sends a vector message (Vector Exchange [VE]).

Vector Exchange (VE) is a bit vector with a bit corresponding to the quorum (majority of the Storage Points) and a 1-bit indicates that the corresponding Storage Point knows of a given update. Just like vector timestamp, when a node/SP initiates VE it sets its bit vector to 1 and other bits to 0 and broadcasts it and any SP that sees the VE, it sets its own bit vector to 1 and re-broadcasts the modified VE. When a majority of the bits are set to 1, we say that an agreement occurs, and it is safe for any SP (that sees the majority of the bits set) to upload this latest update i.e. once the agreement is established (majority of the bit vectors are 1 (quorum)), the SP copies the file to a permanent location on its HTTP server where it is available for download.

Note - Each storage point saves its VE so that its not lost due to network failure/restart.

Answer 3

**Sample Output**





* Do actors ever receive messages originating from a given actor out of order?

Yes the actors receive messages out of order

* What if the messages are forwarded through an intermediary?

The intermediary could have enforced the delivery order by organizing the messages by timestamp

* What if two actors multicast to the same group? Does each member receive the messages in the same order?

All the messages would reach the group members but messages are not guaranteed to be in order as it depends if the actor is available to process the request/message

* Do actors ever receive messages for groups "late", after having left the group?

Yes, the actors indeed receive messages ‘late’ after having left the group

* How does your choice of weights and parameters affect the amount of message traffic?

I used maxweight to be 100 and groupjoin weight is 60%, Leave group is 15% and multicast is 25% of the maxweight respectively, it leads to more traffic towards joining the group, followed by # of multicast messages and least is actors leaving the groups

* How can you be sure your code is working, i.e., the right actors are receiving the right messages?

Any message multicast is received by all members of the group.

The gameservers are also able to report that the message from a group was received after it left the group

The gameservers are able to join and leave the group