Replication1

Very important module for Distributed data

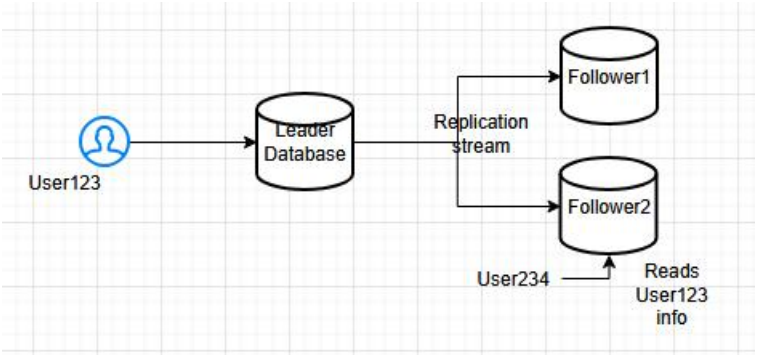
2 ways to improve Availability of Database
1. Replication (Not splitting data but replicating/copying entire data into other machines)
2. Partioning (Splitting data, part1 can be in Database1/Machine1, part2 can be in a different machine)

Replication means keeping a copy of data
Why?
- to avoid single point of failure
- for high availability
- keep the data nearby
- By increasing the instances of databases, we are increasing the Read throughput
- simple

Assumptions:
- Dataset should be small

We need some Replication method to keep our replications in sync and updated

Algorithms for Replication:
1. Single leader:
User123 ----> Database. User123 wants to update their profile pic.
Update UsersTable
Set profile_url='img.jpg'
Where UsersTable.id=123

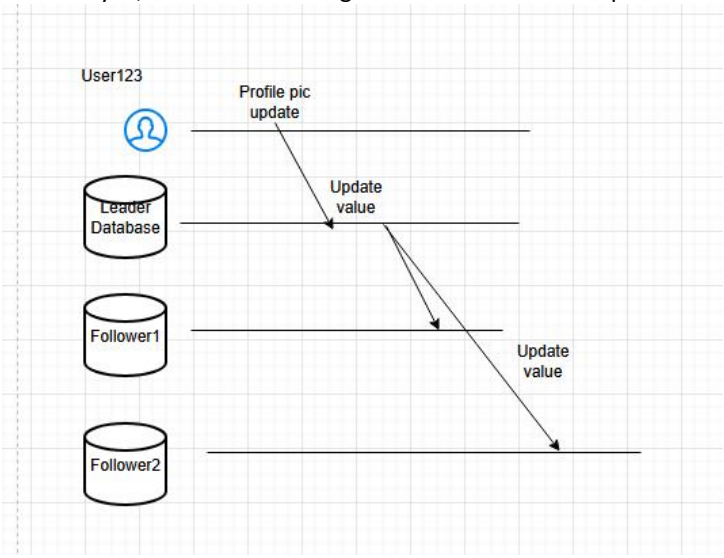


We update in Leader Database first then we send the same request to Follower1 and Follower2 to update their databases also. In other words, Write operations are replicated in Follower instances
Basically, all Write / Update / Delete operations are performed on Leader Database (Node) only (where values are changing) and for Followers, we will have only Read operations.
Lets say User234 is trying to see Profile pic of User123 from Follower2, in that case, Read can be performed on F2 also.
Select * from
UsersTable
Where UsersTable.id=123

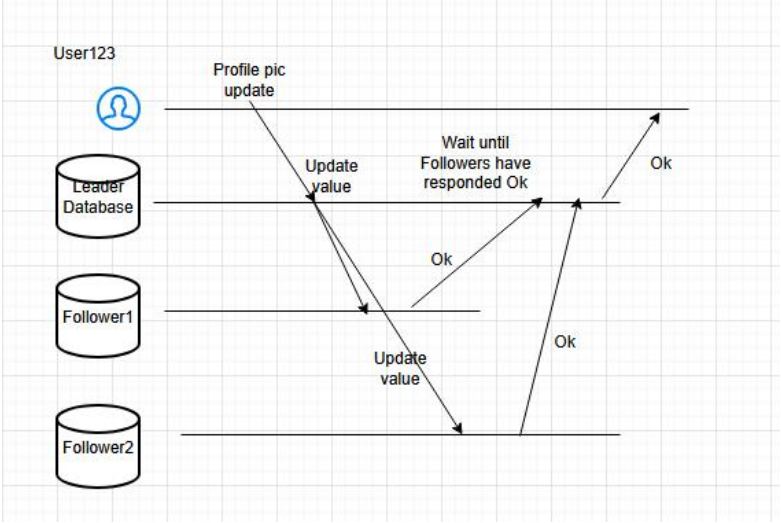
Single leader:
- Write / Delete / Read

Follower:
- Only Read operations

Sync & Async
This is Async, we are not waiting for the Followers to respond



This is Sync replication method where we respond Ok back to the User only after the Followers have responded also. If the Follower has some kind of failure then we wont get OK from follower. In that case, resource of Leader might be on hold before other Write/Update queries.



Normally, we prefer only Async queries and not Sync queries by that way, our system will be faster.

Sync:
- Followers are also up to date
- If our Leader node fails, we have safe data

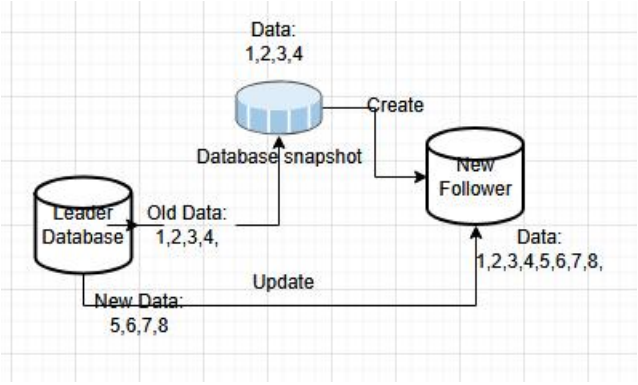
Disadvantages:
- Wait time will be longer
- Block Leader until Follower
- Because of Leader block, Sync is impractical

Async:
- Leader not blocked, even if Follower is dead
- Fast response

Disadvantages:
- Chance of data loss, because data is not always updated in Followers, data is not durable

How can we setup a new Follower?
- Snapshots: It's like an image and it contains all the data at that particular point of time.
We can take a Snapshot of Staging env lets say then we replicate in Local
- Whatever data our database holds, we are going to pass it to new node
- Leader operations will be copied to Follower

Basically, we create a new Follower from Snapshot with data: 1,2,3,4. Then Leader gets updated with new data: 5,6,7,8. Then we update the new Follower with new data: 5,6,7,8 also



Setup a new Leader:
When had only one Leader and we lost that Leader now
We choose a Follower that has the most updated or recent data to minimum data loss
- Choose most recently updated data node, lets say Follower F2
- Update F2
- Configure in App about the new Leader

HW: FSIMAGE and EDITLOGS for Snapshots