CS 578

Programming Assignment - 4.1

Mauli Mehulkumar Patel

Student ID: 35231636

Design:

This design is for implementation of datastore (key-value) with GigaPaxos RSM. This design uses GigaPaxos for fault tolerance, ordering, and replication while accepting SQL-style requests (CQL commands) encoded as strings inside GigaPaxos request packets and executing them on a Cassandra backend.

MyDBReplicableAppGP.java

Architecture:

- GigaPaxos: External framework to provide consistency, replication and request handling.
- MyDBReplicatedAppGP: Application; Implements the replicable class; processing requests, cassandra execution etc.
- Cassandra Backend: This is the data storage for replicated state in this implementation.

Initialisation and Schema:

- Read the keyspace name from args[0].
- Connects to a Cassandra cluster.
- Creates the keyspace if missing.
- Creates a simple grade(id text PRIMARY KEY, event int) table.

Request:

- GP will wrap the request inside a json object.
- The code will extract the CQL command from the json object and use that to execute on cassandra.

Checkpointing:

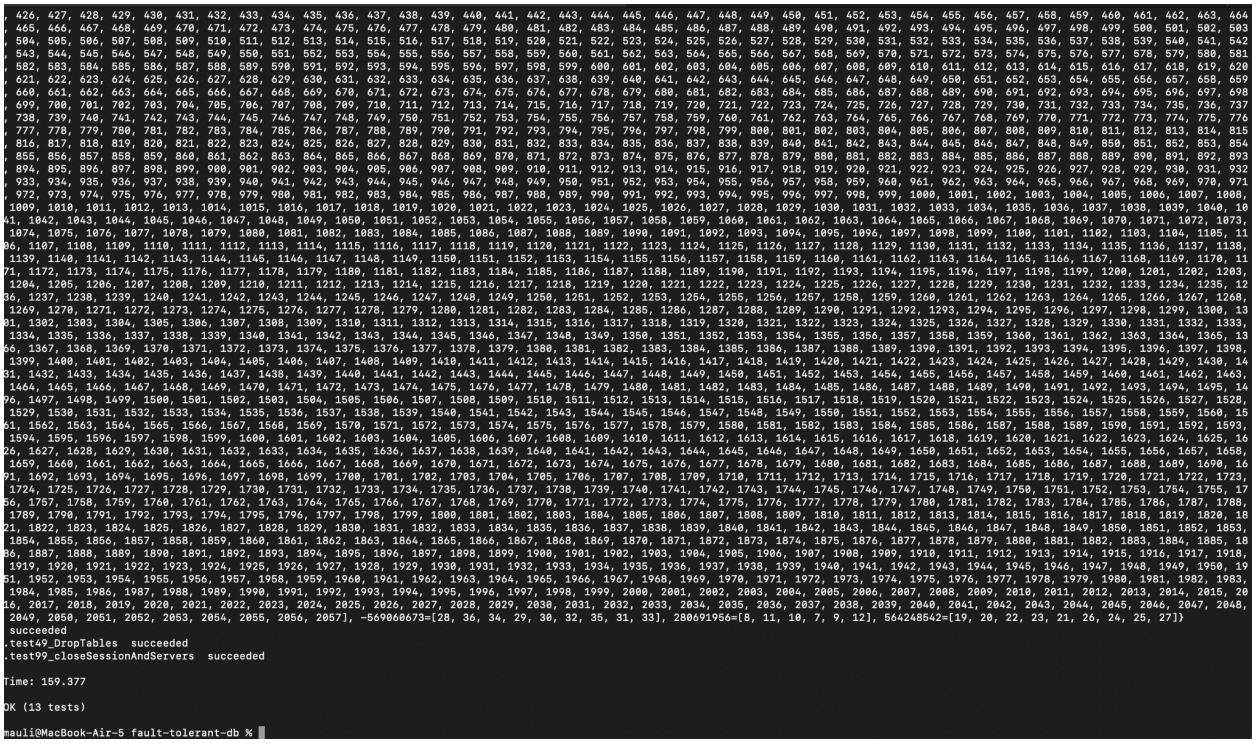
- GigaPaxos expects snapshot/restore functionality to reconstruct the replicated state machine's logical snapshot.
- This will return a json state of the replica. This is done so that it is parsable.

Restore:

- This is used to restore the state from the checkpoint.
- It parses the json and reinstates the id and event.

Failing:

- All the test cases pass.



Previous failures:

- Good parsing to convert from the json to CQL format.
- State mismatch error due to incorrect entry split due to curly brackets
- Some errors on Cassandra due to incompatibility with MacOS. Stack size etc.