EXPENSES TRACKER

Introduction

As a part of the leader election in RAFT, we have incorporated Heartbeats, Timeouts, remote-procedure-calls (RPCs) on each node along with the creation of multiple asynchronous threads to perform operations simultaneously such as:

- Listen for any incoming requests/packets
- Send heartbeats, vote requests, and other messages
- Processing packets on receiver end
- Keeping track of the state of the node

The leader has a dedicated asynchronous thread to send out AppendEntry RPC/heartbeats. If a follower has not received a heartbeat from the leader over a period, it assumes the leader has failed, it increases its term number and changes its state from follower to candidate. The follower resets its timeout every time it receives a heartbeat. The election starts once a follower's timeout occurs, the follower transitions to candidate state and starts an election, sending RequestVote RPC to other nodes. Once a candidate wins the election and becomes a leader, it starts sending Heartbeats to other nodes to establish its authority.

Design Overview

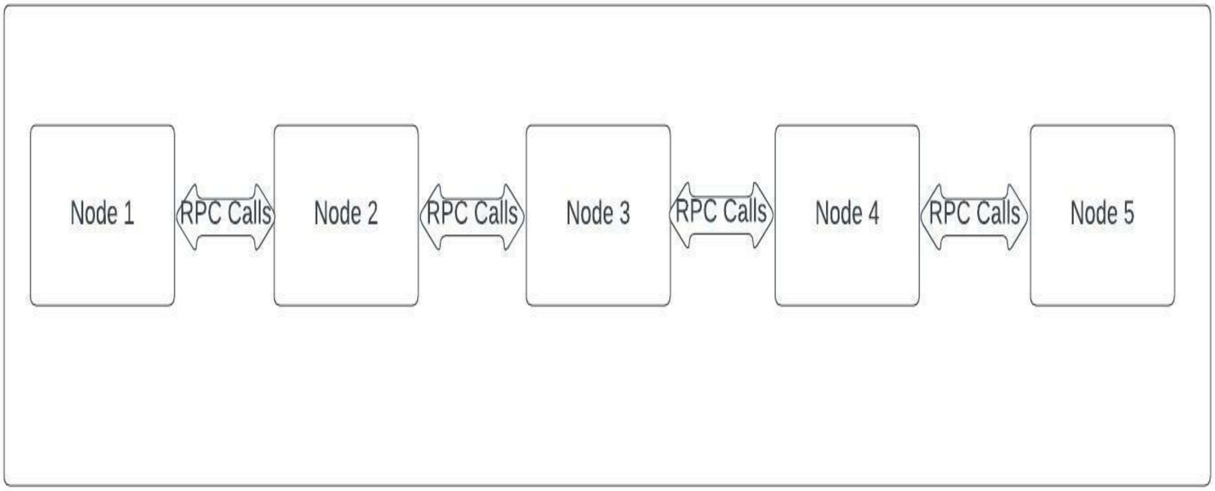


Fig: Architecture Diagram

Implementation

As there are 5 nodes in total, each node sends and receives RPC calls among them during the election phase. Listeners port information is given as follows:

Listener	Port Number
AppendEntryRPC	9000
RequestVoteRPC	9001
VoteAcknowledgement	9002
ControllerListener	5555

As this phase does not require client request testing, we have excluded client-side containers from docker compose-up.

RAFT Node

	Node1	Node2	Node3	Node4	Node5
CurrentTerm	0 (Initially) Dynamic	0 (Initially) Dynamic	0 (Initially) Dynamic	0 (Initially) Dynamic	0 (Initially) Dynamic
votedFor	Dynamic	Dynamic	Dynamic	Dynamic	Dynamic
Log []	Null	Null	Null	Null	Null
Timeout	Dynamic (200-300ms)	Dynamic (200-300ms)	Dynamic (200-300ms)	Dynamic (200-300ms)	Dynamic (200-300ms)
Heartbeat	100ms	100ms	100ms	100ms	100ms

RPC Message Structure:

Request Vote RPC

```
{
 "term":"currentTerm"+1,
 "candidateId":"currentCandidate",
 "last_log_index":"-1",
 "last_log_term":0,
 "sender_name":"currentCandidate",
 "request":"VOTE_REQUEST"
}
```

AppendEntry RPC

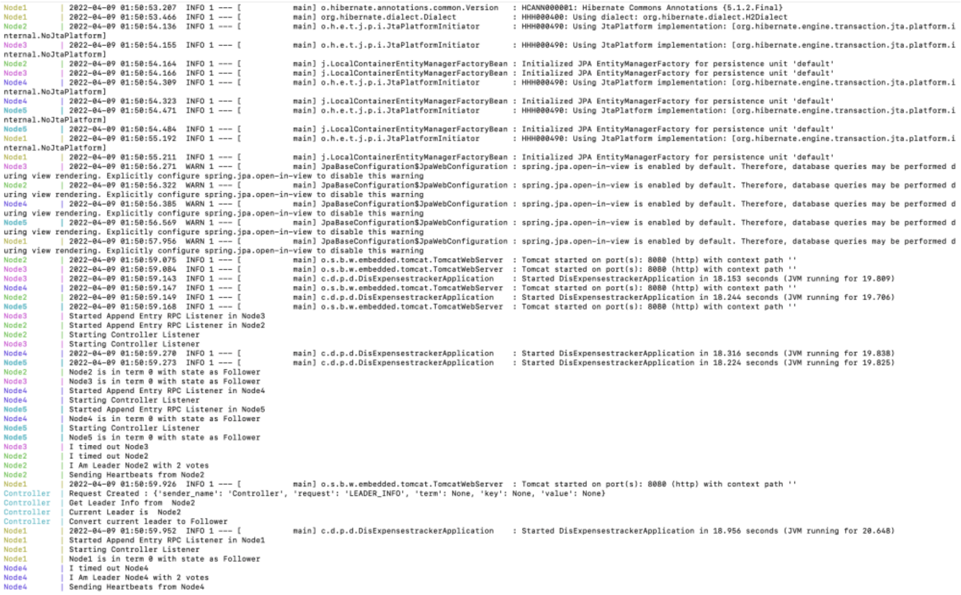
```
{
  "term":"currentTerm",
  "leaderId":"currentLeader",
  "prev_log_index":"-1",
  "prev_log_term":0,
  "sender_name":"currentNode",
  "request":"APPEND_RPC"
}
```

Vote Acknowledgment RPC

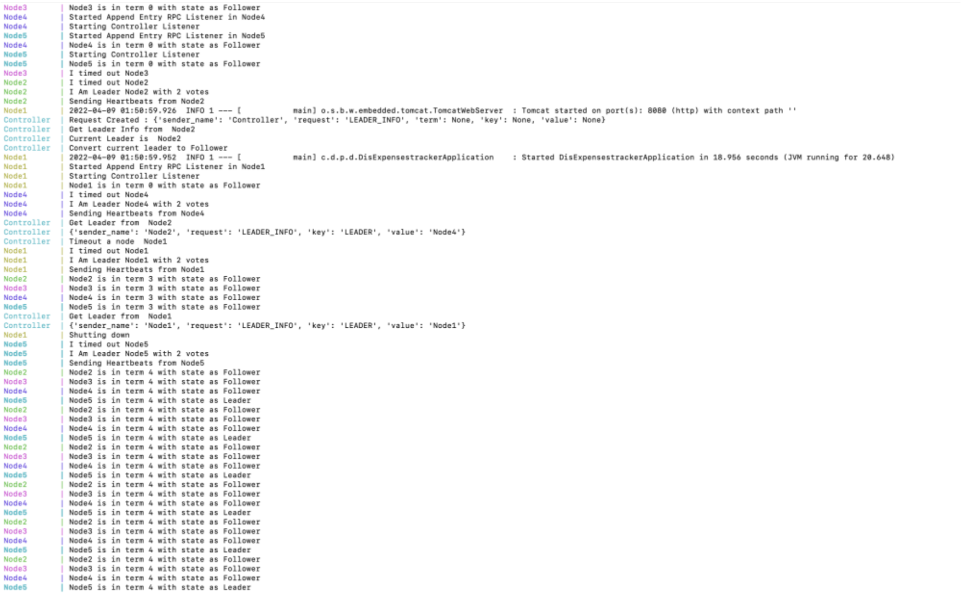
```
{
  "vote":1,
  "from":"currentNode",
  "to":"Requester",
  "request":"VOTE_ACK"
}
```

Validation

We can see that all the nodes have stabilized after election.



```
Node1        | 2022-04-09 01:50:53.207  INFO 1 --- [           main] o.hibernate.annotations.common.Version   : HCANN000001: Hibernate Commons Annotations {5.1.2.Final}
Node1        | 2022-04-09 01:50:53.466  INFO 1 --- [           main] org.hibernate.dialect.Dialect            : HHH000400: Using dialect: org.hibernate.dialect.H2Dialect
Node2        | 2022-04-09 01:50:54.136  INFO 1 --- [           main] o.h.e.t.j.p.i.JtaPlatformInitiator       : HHH000490: Using JtaPlatform implementation: [org.hibernate.engine.transaction.jta.platform.i
nternal.NoJtaPlatform]
Node3        | 2022-04-09 01:50:54.155  INFO 1 --- [           main] o.h.e.t.j.p.i.JtaPlatformInitiator       : HHH000490: Using JtaPlatform implementation: [org.hibernate.engine.transaction.jta.platform.i
nternal.NoJtaPlatform]
Node2        | 2022-04-09 01:50:54.164  INFO 1 --- [           main] j.LocalContainerEntityManagerFactoryBean : Initialized JPA EntityManagerFactory for persistence unit 'default'
Node3        | 2022-04-09 01:50:54.166  INFO 1 --- [           main] j.LocalContainerEntityManagerFactoryBean : Initialized JPA EntityManagerFactory for persistence unit 'default'
Node4        | 2022-04-09 01:50:54.309  INFO 1 --- [           main] o.h.e.t.j.p.i.JtaPlatformInitiator       : HHH000490: Using JtaPlatform implementation: [org.hibernate.engine.transaction.jta.platform.i
nternal.NoJtaPlatform]
Node4        | 2022-04-09 01:50:54.323  INFO 1 --- [           main] j.LocalContainerEntityManagerFactoryBean : Initialized JPA EntityManagerFactory for persistence unit 'default'
Node5        | 2022-04-09 01:50:54.471  INFO 1 --- [           main] o.h.e.t.j.p.i.JtaPlatformInitiator       : HHH000490: Using JtaPlatform implementation: [org.hibernate.engine.transaction.jta.platform.i
nternal.NoJtaPlatform]
Node5        | 2022-04-09 01:50:54.484  INFO 1 --- [           main] j.LocalContainerEntityManagerFactoryBean : Initialized JPA EntityManagerFactory for persistence unit 'default'
Node1        | 2022-04-09 01:50:55.192  INFO 1 --- [           main] o.h.e.t.j.p.i.JtaPlatformInitiator       : HHH000490: Using JtaPlatform implementation: [org.hibernate.engine.transaction.jta.platform.i
nternal.NoJtaPlatform]
Node1        | 2022-04-09 01:50:55.211  INFO 1 --- [           main] j.LocalContainerEntityManagerFactoryBean : Initialized JPA EntityManagerFactory for persistence unit 'default'
Node3        | 2022-04-09 01:50:56.271  WARN 1 --- [           main] JpaBaseConfiguration$JpaWebConfiguration : spring.jpa.open-in-view is enabled by default. Therefore, database queries may be performed d
uring view rendering. Explicitly configure spring.jpa.open-in-view to disable this warning
Node2        | 2022-04-09 01:50:56.322  WARN 1 --- [           main] JpaBaseConfiguration$JpaWebConfiguration : spring.jpa.open-in-view is enabled by default. Therefore, database queries may be performed d
uring view rendering. Explicitly configure spring.jpa.open-in-view to disable this warning
Node4        | 2022-04-09 01:50:56.385  WARN 1 --- [           main] JpaBaseConfiguration$JpaWebConfiguration : spring.jpa.open-in-view is enabled by default. Therefore, database queries may be performed d
uring view rendering. Explicitly configure spring.jpa.open-in-view to disable this warning
Node5        | 2022-04-09 01:50:56.569  WARN 1 --- [           main] JpaBaseConfiguration$JpaWebConfiguration : spring.jpa.open-in-view is enabled by default. Therefore, database queries may be performed d
uring view rendering. Explicitly configure spring.jpa.open-in-view to disable this warning
Node1        | 2022-04-09 01:50:57.956  WARN 1 --- [           main] JpaBaseConfiguration$JpaWebConfiguration : spring.jpa.open-in-view is enabled by default. Therefore, database queries may be performed d
uring view rendering. Explicitly configure spring.jpa.open-in-view to disable this warning
Node2        | 2022-04-09 01:50:59.075  INFO 1 --- [           main] o.s.b.w.embedded.tomcat.TomcatWebServer  : Tomcat started on port(s): 8080 (http) with context path ''
Node3        | 2022-04-09 01:50:59.084  INFO 1 --- [           main] o.s.b.w.embedded.tomcat.TomcatWebServer  : Tomcat started on port(s): 8080 (http) with context path ''
Node3        | 2022-04-09 01:50:59.143  INFO 1 --- [           main] c.d.p.d.DisExpensestrackerApplication    : Started DisExpensestrackerApplication in 18.153 seconds (JVM running for 19.809)
Node4        | 2022-04-09 01:50:59.147  INFO 1 --- [           main] o.s.b.w.embedded.tomcat.TomcatWebServer  : Tomcat started on port(s): 8080 (http) with context path ''
Node2        | 2022-04-09 01:50:59.149  INFO 1 --- [           main] c.d.p.d.DisExpensestrackerApplication    : Started DisExpensestrackerApplication in 18.244 seconds (JVM running for 19.706)
Node5        | 2022-04-09 01:50:59.168  INFO 1 --- [           main] o.s.b.w.embedded.tomcat.TomcatWebServer  : Tomcat started on port(s): 8080 (http) with context path ''
Node3        | Started Append Entry RPC Listener in Node3
Node2        | Started Append Entry RPC Listener in Node2
Node2        | Starting Controller Listener
Node3        | Starting Controller Listener
Node4        | 2022-04-09 01:50:59.270  INFO 1 --- [           main] c.d.p.d.DisExpensestrackerApplication    : Started DisExpensestrackerApplication in 18.316 seconds (JVM running for 19.838)
Node5        | 2022-04-09 01:50:59.273  INFO 1 --- [           main] c.d.p.d.DisExpensestrackerApplication    : Started DisExpensestrackerApplication in 18.224 seconds (JVM running for 19.825)
Node2        | Node2 is in term 0 with state as Follower
Node3        | Node3 is in term 0 with state as Follower
Node4        | Started Append Entry RPC Listener in Node4
Node4        | Starting Controller Listener
Node5        | Started Append Entry RPC Listener in Node5
Node4        | Node4 is in term 0 with state as Follower
Node5        | Starting Controller Listener
Node5        | Node5 is in term 0 with state as Follower
Node3        | I timed out Node3
Node2        | I timed out Node2
Node2        | I Am Leader Node2 with 2 votes
Node2        | Sending Heartbeats from Node2
Node1        | 2022-04-09 01:50:59.926  INFO 1 --- [           main] o.s.b.w.embedded.tomcat.TomcatWebServer  : Tomcat started on port(s): 8080 (http) with context path ''
Controller   | Request Created : {'sender_name': 'Controller', 'request': 'LEADER_INFO', 'term': None, 'key': None, 'value': None}
Controller   | Get Leader Info from  Node2
Controller   | Current Leader is  Node2
Controller   | Convert current leader to Follower
Node1        | 2022-04-09 01:50:59.952  INFO 1 --- [           main] c.d.p.d.DisExpensestrackerApplication    : Started DisExpensestrackerApplication in 18.956 seconds (JVM running for 20.648)
Node1        | Started Append Entry RPC Listener in Node1
Node1        | Starting Controller Listener
Node1        | Node1 is in term 0 with state as Follower
Node4        | I timed out Node4
Node4        | I Am Leader Node4 with 2 votes
Node4        | Sending Heartbeats from Node4
```



```
Node3        | Node3 is in term 0 with state as Follower
Node4        | Started Append Entry RPC Listener in Node4
Node4        | Starting Controller Listener
Node5        | Started Append Entry RPC Listener in Node5
Node4        | Node4 is in term 0 with state as Follower
Node5        | Starting Controller Listener
Node5        | Node5 is in term 0 with state as Follower
Node3        | I timed out Node3
Node2        | I timed out Node2
Node2        | I Am Leader Node2 with 2 votes
Node2        | Sending Heartbeats from Node2
Node1        | 2022-04-09 01:50:59.926  INFO 1 --- [           main] o.s.b.w.embedded.tomcat.TomcatWebServer  : Tomcat started on port(s): 8080 (http) with context path ''
Controller   | Request Created : {'sender_name': 'Controller', 'request': 'LEADER_INFO', 'term': None, 'key': None, 'value': None}
Controller   | Get Leader Info from  Node2
Controller   | Current Leader is  Node2
Controller   | Convert current leader to Follower
Node1        | 2022-04-09 01:50:59.952  INFO 1 --- [           main] c.d.p.d.DisExpensestrackerApplication    : Started DisExpensestrackerApplication in 18.956 seconds (JVM running for 20.648)
Node1        | Started Append Entry RPC Listener in Node1
Node1        | Starting Controller Listener
Node1        | Node1 is in term 0 with state as Follower
Node4        | I timed out Node4
Node4        | I Am Leader Node4 with 2 votes
Node4        | Sending Heartbeats from Node4
Controller   | Get Leader from  Node2
Controller   | {'sender_name': 'Node2', 'request': 'LEADER_INFO', 'key': 'LEADER', 'value': 'Node4'}
Controller   | Timeout a node  Node1
Node1        | I timed out Node1
Node1        | I Am Leader Node1 with 2 votes
Node1        | Sending Heartbeats from Node1
Node2        | Node2 is in term 3 with state as Follower
Node3        | Node3 is in term 3 with state as Follower
Node4        | Node4 is in term 3 with state as Follower
Node5        | Node5 is in term 3 with state as Follower
Controller   | Get Leader from  Node1
Controller   | {'sender_name': 'Node1', 'request': 'LEADER_INFO', 'key': 'LEADER', 'value': 'Node1'}
Node1        | Shutting down
Node5        | I timed out Node5
Node5        | I Am Leader Node5 with 2 votes
Node5        | Sending Heartbeats from Node5
Node2        | Node2 is in term 4 with state as Follower
Node3        | Node3 is in term 4 with state as Follower
Node4        | Node4 is in term 4 with state as Follower
Node5        | Node5 is in term 4 with state as Leader
Node2        | Node2 is in term 4 with state as Follower
Node3        | Node3 is in term 4 with state as Follower
Node4        | Node4 is in term 4 with state as Follower
Node5        | Node5 is in term 4 with state as Leader
Node2        | Node2 is in term 4 with state as Follower
Node3        | Node3 is in term 4 with state as Follower
Node4        | Node4 is in term 4 with state as Follower
Node5        | Node5 is in term 4 with state as Leader
Node2        | Node2 is in term 4 with state as Follower
Node3        | Node3 is in term 4 with state as Follower
Node4        | Node4 is in term 4 with state as Follower
Node5        | Node5 is in term 4 with state as Leader
Node2        | Node2 is in term 4 with state as Follower
Node3        | Node3 is in term 4 with state as Follower
Node4        | Node4 is in term 4 with state as Follower
Node5        | Node5 is in term 4 with state as Leader
Node2        | Node2 is in term 4 with state as Follower
Node3        | Node3 is in term 4 with state as Follower
Node4        | Node4 is in term 4 with state as Follower
Node5        | Node5 is in term 4 with state as Leader
```

References

https://raft.github.io/raft.pdf

http://thesecretlivesofdata.com/raft

https://www.youtube.com/watch?v=YbZ3zDzDnrw

https://docs.python.org/3/library/socket.html

https://docs.python.org/3/library/threading.html