Linearizable Quorum Reads in Paxos

Aleksey Charapko, Ailidani Ailijiang, and Murat Demirbas
What is Paxos

• Solves Distributed Consensus
• Operates in 3 Phases
  1. Elect a Leader for a round
  2. Accept value
  3. Mark committed
From Paxos to MultiPaxos

• Run Phase-1 once
• Keep stable leader
• Repeat phase-2 many times
• Piggyback phase-3 to some next phase-2
Problems with Paxos

• Single Leader Bottleneck
  • Lots of messages to send/receive
Paxos in Distributed Databases

• Data replication
  • Paxos and its derivatives are often used in strongly-consistent databases.
    • CockroachDB
    • Spanner/Cloud Spanner
    • YugaByte
    • PaxosStore

CockroachDB (Raft)

YugaByte (Raft)

Cloud Spanner (Paxos)
Reading from Paxos-Based System

- Paxos has no notion of ‘Reads’.
  - Consensus on commands
  - Order of commands
  - ‘Read’ is a command.
  - Strong Consistency for any type of command
Reading from Paxos-Based System II

- Read from the leader
  - Need leases to protect the leader
- Read from any replica
  - Read stale data (ZooKeeper)
- Read from quorum of replicas
  - Leader may still be slightly ahead!
Paxos Quorum Read (PQR)

• Two-phases to make sure we read the latest value.
  • Quorum-read Phase
    • Read from quorum for latest accepted(!) value. Remember the max slot #
  • Rinse Phase
    • Read one(!) node for executed.
    • Executed slot is known to have been globally committed with no gaps.
    • Return value if executed slot # >= accepted slot # from Quorum-read phase.
Paxos Quorum Read (PQR)

- **Client**
  - read latest accepted slot from followers

- **Leader**
  - wait for quorum
  - Slot 's'
  - has 's' been executed?

- **Follower**:
  - Rinse phase

**Phase:** Quorum-read phase
PQR Example

Leader: 1 2 3 4 5
Follower: 1 2 3 4

Client: Rinse Slot 5!
Quorum-read Phase

Leader: 1 2 3 4 5 6 7
Follower: 1 2 3 4 5 6

Client: Rinseed Slot 5!
Rinse Phase
Doing PQR in one RTT

• PQR takes 2 RTTs
• Possible to eliminate Rinse Phase in absence of new commands (no progress):
  • last accepted slot # = last executed slot #
  • Means that Rinse phase condition is already met!
• Can track progress per object or shard in similar way
PQR Evaluation

- Paxos runs full round for read
- Paxos Read From Leader reads from dedicated leader node
- 75% writes
- Similar results at 50% and 25% writes, however, PQRs advantage starts to diminish.
PQR Evaluation: Latency

- Fixed throughput @ 4k req/sec
- Suggests a sweet spot for best PQR performance
- PQR works best when leader is close to being saturated, and reads go to followers.
Conclusion

• PQR helps balance load between Paxos leader and follower nodes
• Offloading leader form serving reads allows it to serve more writes
• Reads use underutilized follower nodes
• PQR Improves throughput, especially in write-heavy workloads.