In Search of an Understandable Consensus Algorithm

Authors: Diego Ongaro and John Ousterhout from Stanford University

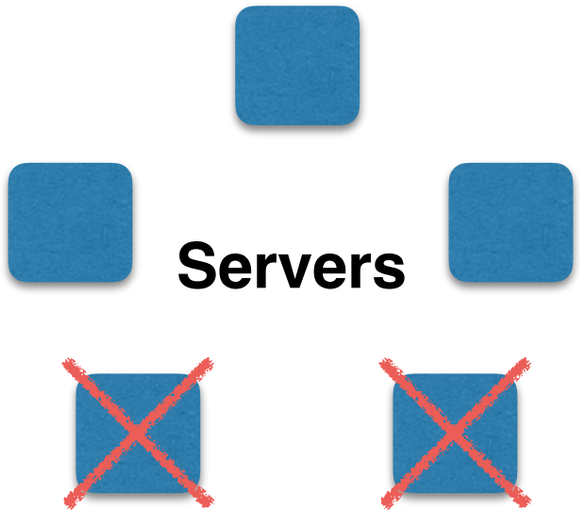
Presenter: Ruolan Zeng
18/10/2018

Outline

- Introduce Raft Consensus Algorithm
    - Consensus
    - Replicated State Machines
    - Raft Algorithm
- Show RaftScope Visualization

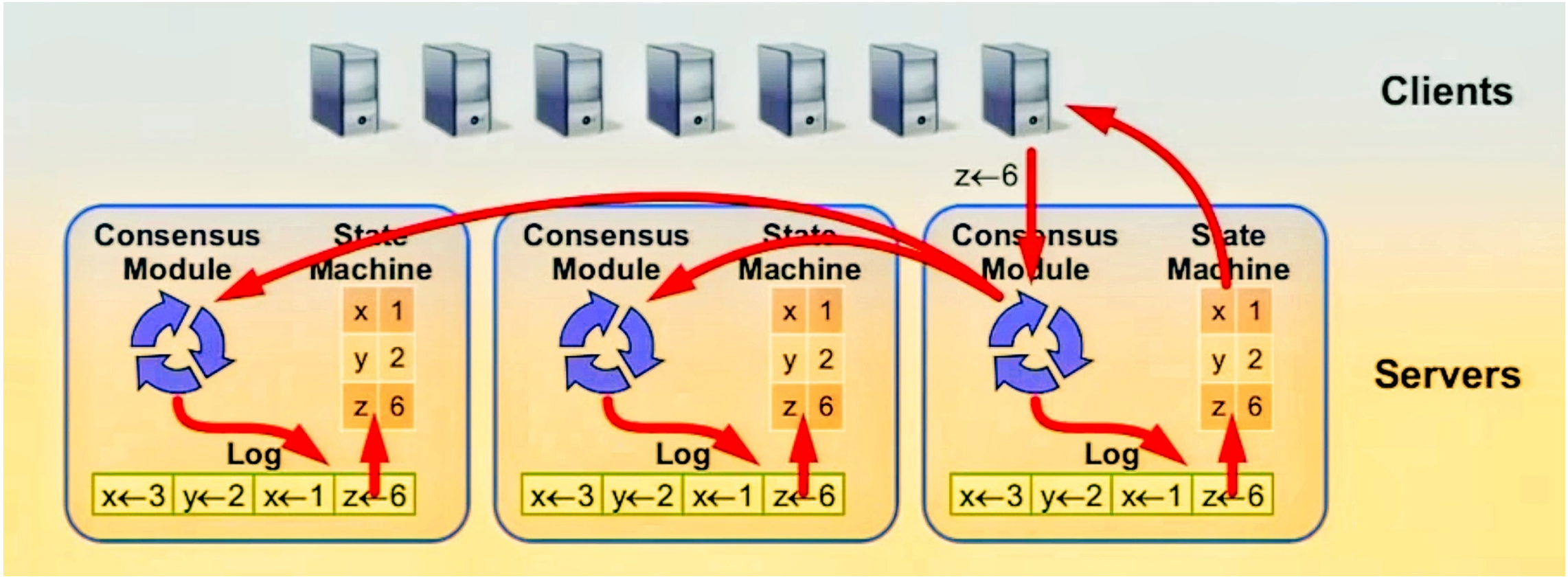
What is Consensus?

- Agreement on shared state
- Recovers from server failures autonomously
    - Minority of servers fail: no problem
    - Majority fail: lose availability, remain consistency



- Key to building consistent storage systems

Replicated State Machine



- Replicated log -> replicated state machine
    - All servers execute same commands in same order
- Consensus module ensures proper log replication

How Raft works?

- Leader election
    - Select one of the servers to act as cluster leader
    - Detect crashes, choose new leader
- Log replication (normal operation)
    - Leader takes commands from clients, appends them to its log
    - Leader replicates its log to other servers (overwriting in consistencies)
- Safety
    - Only a server with an up-to-date log can become leader

RaftScope Visualization

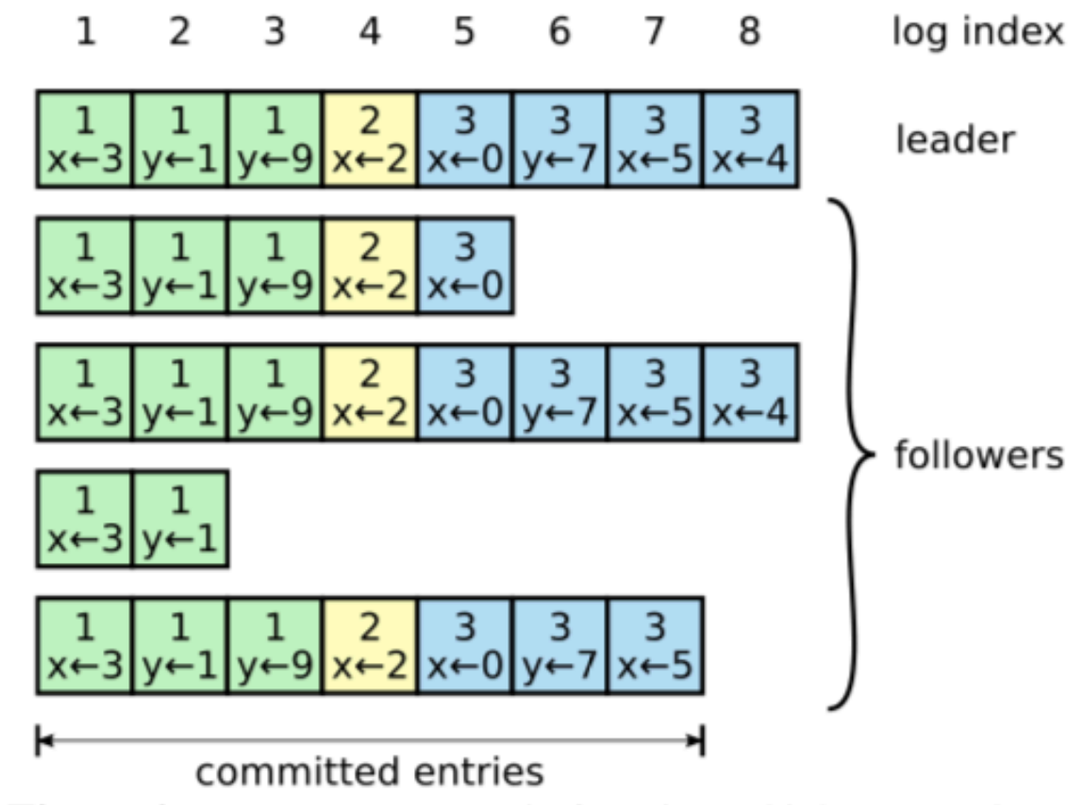
https://raft.github.io

RaftScope Visualization

- Leader Election
    - Normal election
    - Split votes
- Log Replication
    - Normal case
    - Repairing Inconsistencies

RaftScope Visualization

- Log Replication



Core Raft Review

- **Leader election**
    - Heartbeats and timeouts to detect crashes
    - Randomized timeouts to avoid split votes
    - Majority voting to guarantee at most one leader per term
- **Log replication (normal operation)**
    - Leader takes commands from clients, appends them to its log
    - Leader replicates its log to other servers (overwriting inconsistencies)
    - Built-in consistency check simplifies how logs may differ
- **Safety**
    - Only elect leaders with all committed entries in their logs
    - New leader defers committing entries from prior terms

Thank you

Questions