L20: Replicated state machines with Paxos

Frans Kaashoek
6.033 Spring 2013
Paxos properties

- All nodes agree on a value, despite node failures, network failures, delays
  - E.g., X is the next operation to execute
  - E.g., Y is the next primary

- Fault tolerant: succeeds if less than N/2 nodes fail
  - Liveness is not guaranteed

- Assumption: nodes are fail-stop
Paxos rule

• If an earlier proposal number accepted a value, later proposals must accept the same value

• State maintained by acceptor:
  • Np: largest proposal seen in prepare
  • Na: largest proposal seen in accept
  • Va: value accepted for proposal Na

• State must be persistent across reboot
Propose(V):
choose unique N, preferably N > Np
send Prepare(N) to all nodes
if Prepare_OK(Na, Va) from majority:
  V' = Va with highest Na, or V if none
  send Accept(N, V') to all nodes
if Accept_OK(N) from majority:
  send Decided(V') to all

Propose(N):
if N > Np:
  Np = N
  reply Prepare_OK(Na, Va)

Accept(N, V):
if N ≥ Np:
  Na = N, Va = V
  reply Accept_OK(Na, Va)
Summary

- Consistency: single-copy semantics

- Replicated state machines provide single-copy
  - Key issue: agreeing on order of operations
  - Hard case: network partition

- Paxos allows replicas to reach consensus, in presence of machine and network failures
  - Widely used in practice [Chubby, ZooKeeper, etc.]