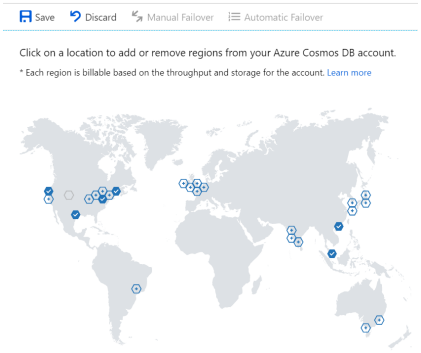
Primary-Backup and Paxos

Dominic Duggan
Stevens Institute of Technology
Based in part on material by Ken Birman

1

Uses of replication

- High availability
- Share loads for scalability
- Lower latency, improve responsiveness



2

Replicated state machine (RSM)

- RSM is a general replication method
- RSM Rules:
  - All replicas start in the same initial state
  - Every replica apply operations in the same order
  - All operations must be deterministic
- All replicas end up in the same state
  - Eventually…

3

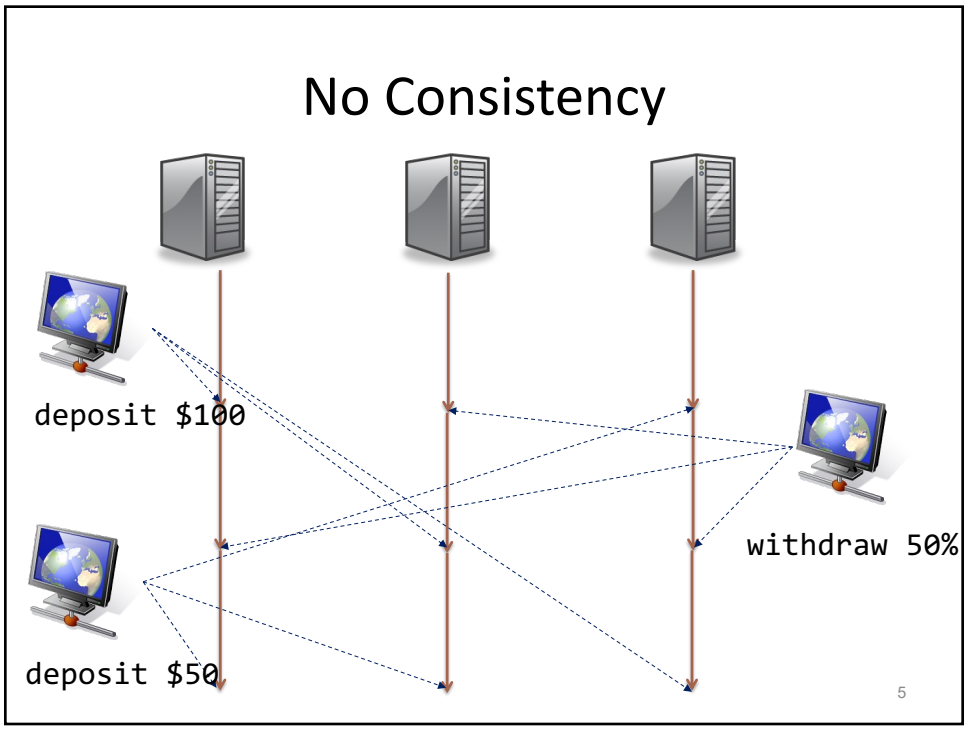
3

Issues

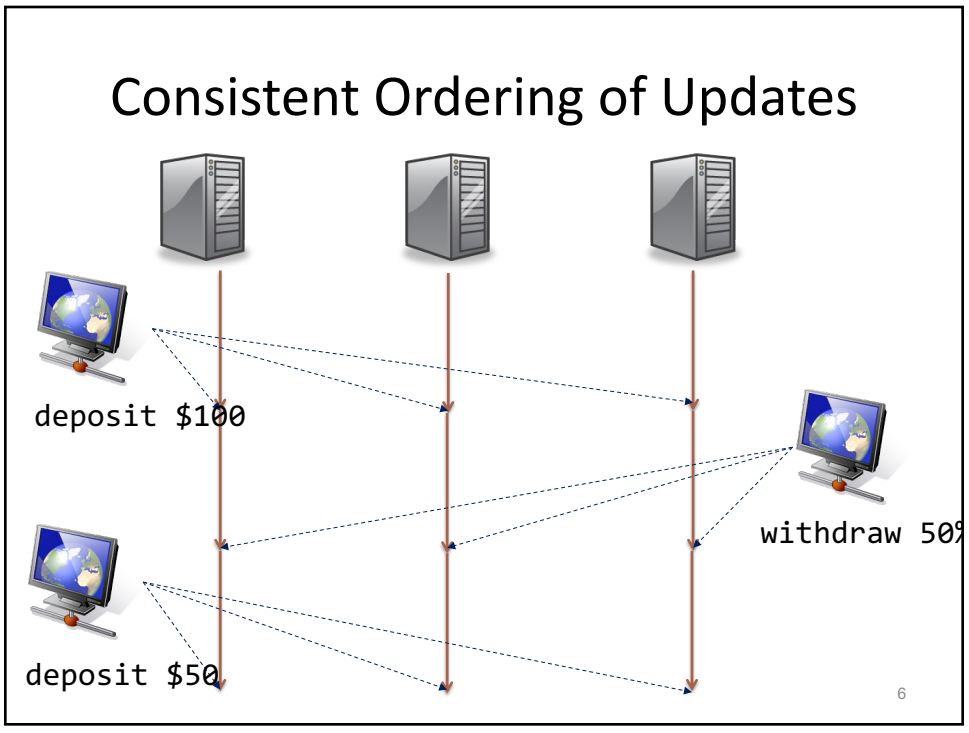
- How do we ensure agreement on order of updates?
- Under what conditions should backup take over?
  - "Split brain" problem

4

4


No Consistency
deposit $100
withdraw 50%
deposit $50
5

5


Consistent Ordering of Updates
deposit $100
withdraw 50%
deposit $50
6

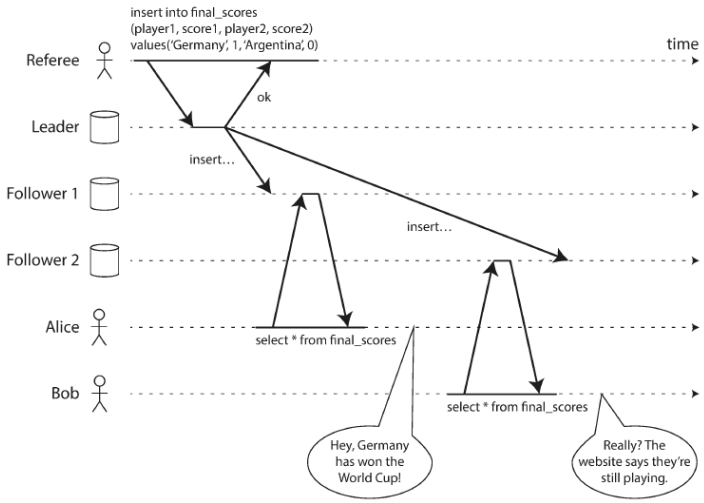
6

Linearizability

- As soon as one client successfully completes a write, all clients reading from the database must be able to see the value just written (*Recency*)
- Replicated database
- Must not read stale copy of data

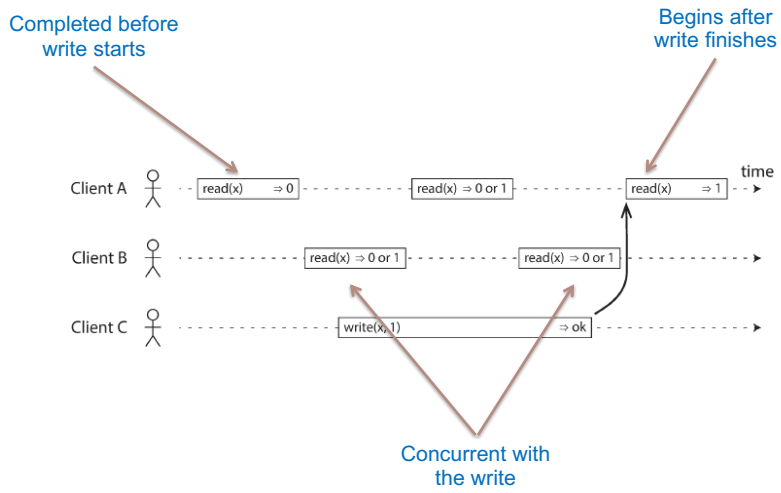
7

Violation of Linearizability



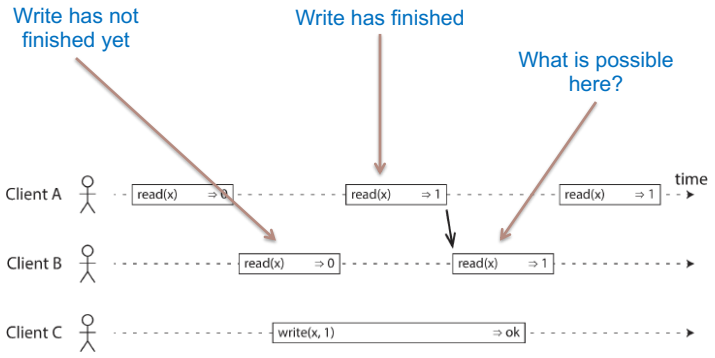
8

Linearizability



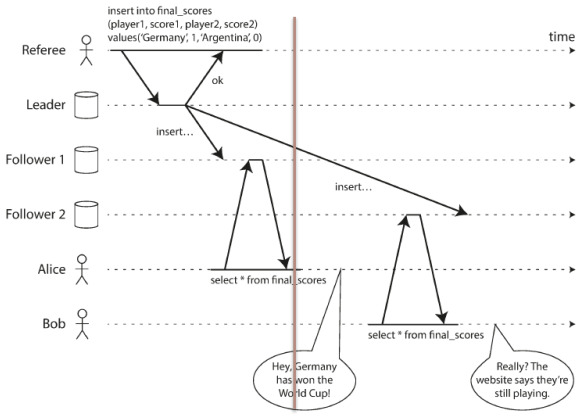
9

Linearizability



10

Linearizability

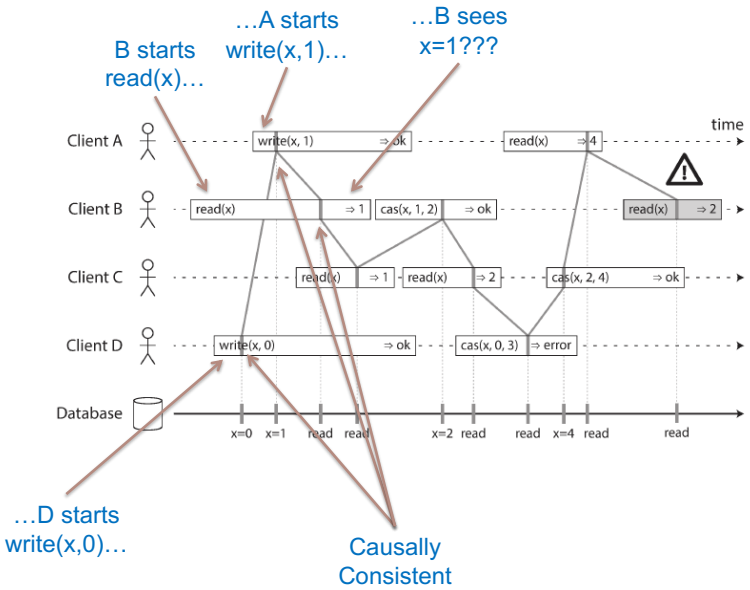


Alice has seen update

Bob sees old value

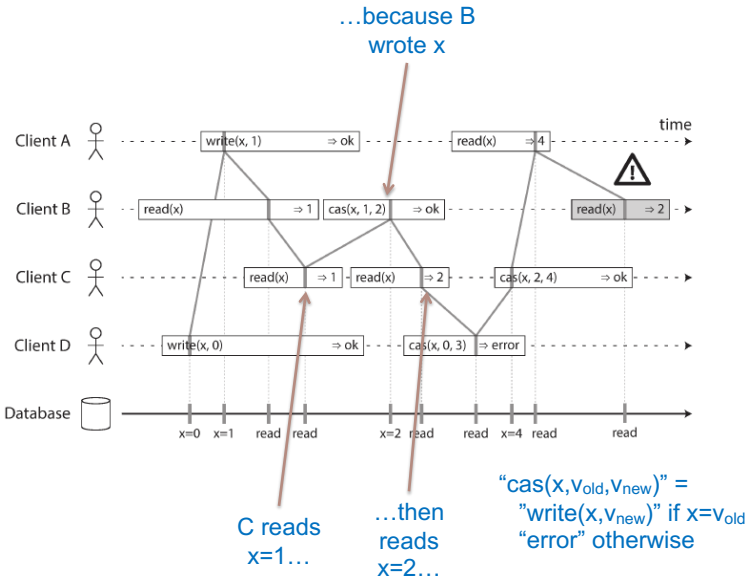
11

Linearizability & Commit Point



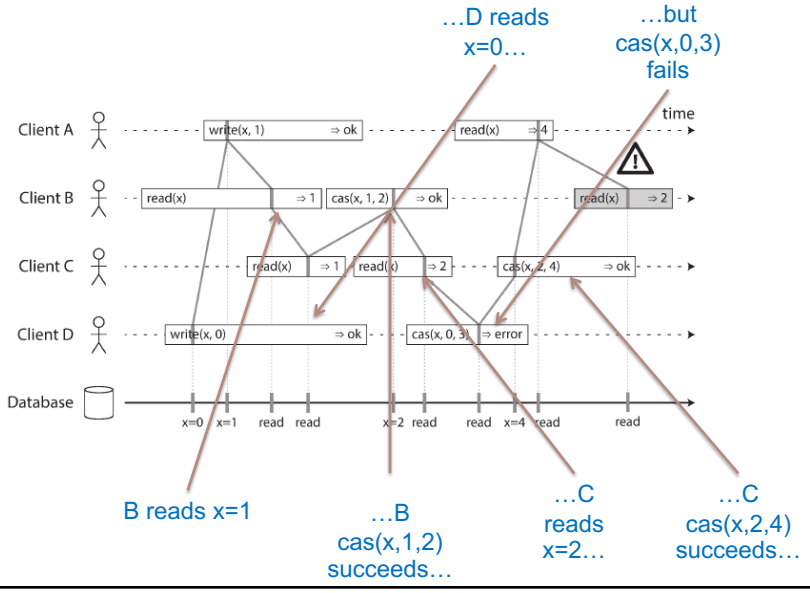
12

Linearizability & Race Conditions



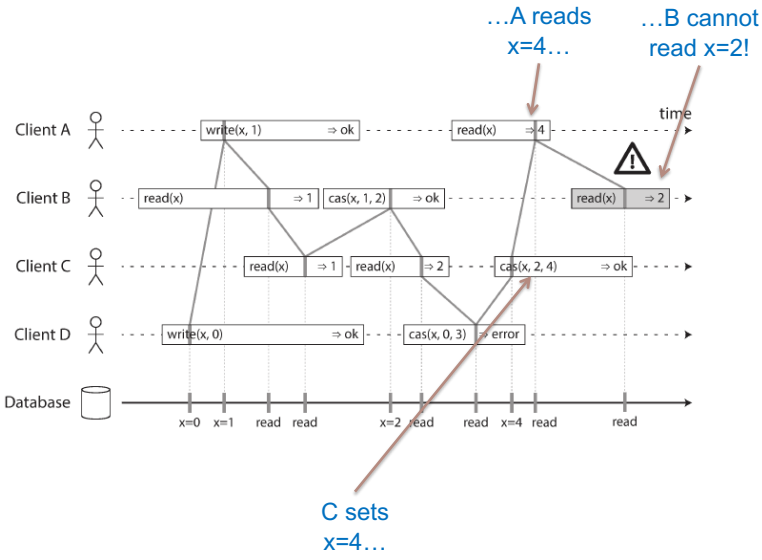
13

Linearizability & Race Conditions



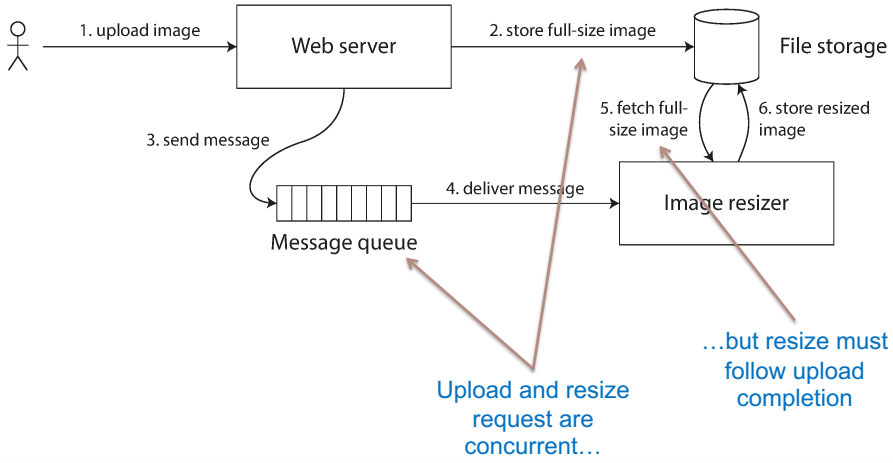
14

Violation of Linearizability



15

Cross-Channel Timing Dependencies



16

Linearizability vs Serializability

- Serializability
  - Isolation property of transactions
  - Actual order of operations may be different from serializable order
- Linearizability
  - Recency guarantee on reads and writes

17

Linearizability vs Serializability

- Strong one-copy serializability (strong-1SR)
  - Combine serializability and linearizability (e.g. 2PL, serial execution)
- Serializable Snapshot Isolation (SSI)
  - Not linearizable, reads from consistent snapshot ignore recent writes

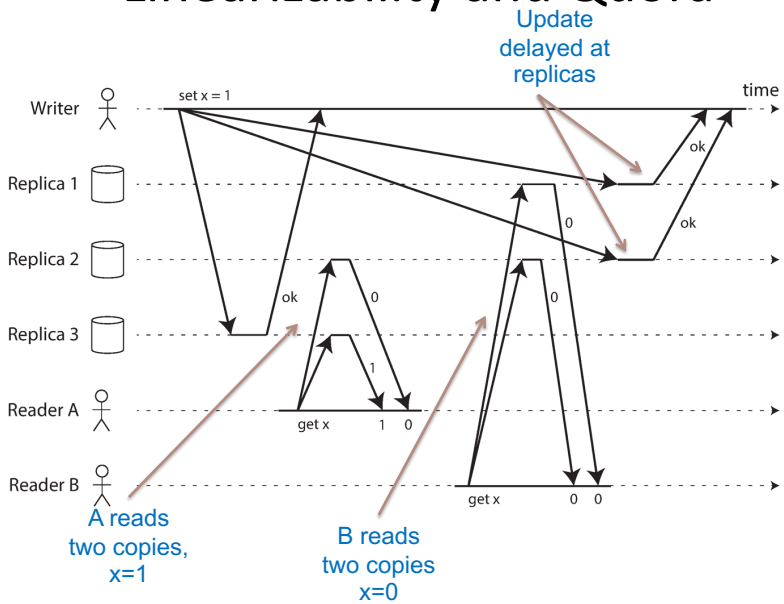
18

Implementing Linearizability

- Single-Leader replication
  - Potentially linearizable…
  - …snapshot isolation?
- Multi-Leader replication
  - Conflicting writes
  - Not linearizable
- Leaderless replication
  - Quorum consensus?

Assume $Q_w=3$, $Q_r=2$

Linearizability and Quora



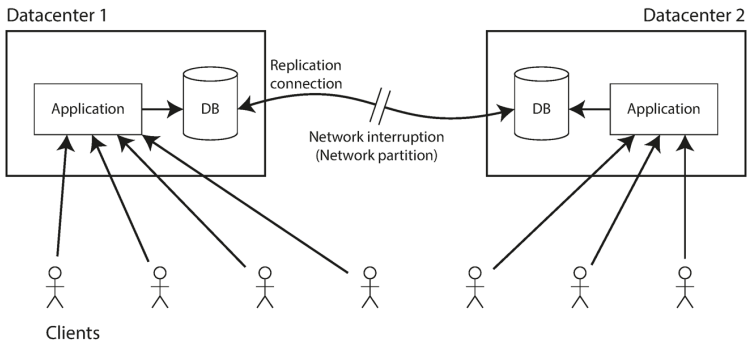
Linearizability and Quora

- How to make it linearizable
  - Read: *Synchronously* do read repairs
  - Write: Get read quorum before write
- Issue: latency
- Issue: doesn't extend to e.g. compare-and-set

21

21

Linearizability vs Availability

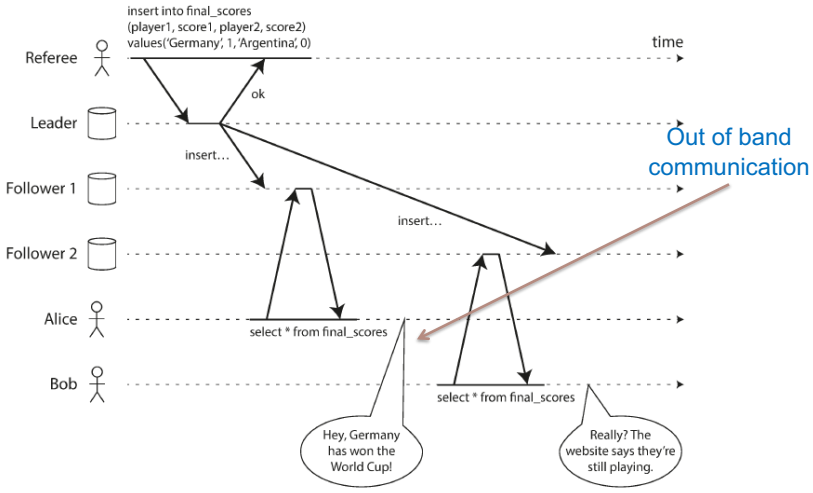


Real issue is **Linearizability vs Latency**

22

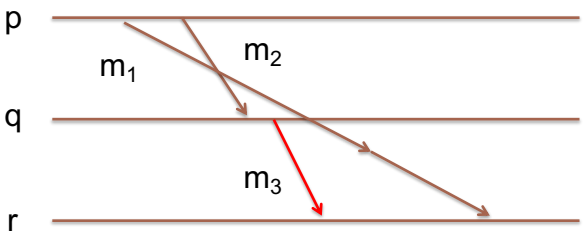
22

Linearizability and Causality



23

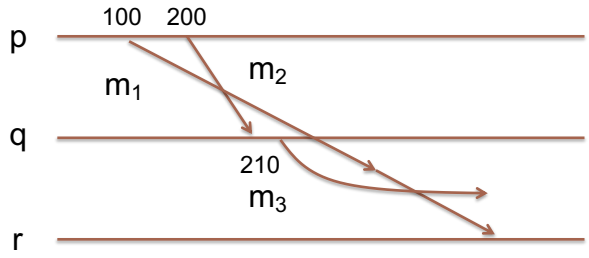
Recall: Causal Message Delivery



24

24

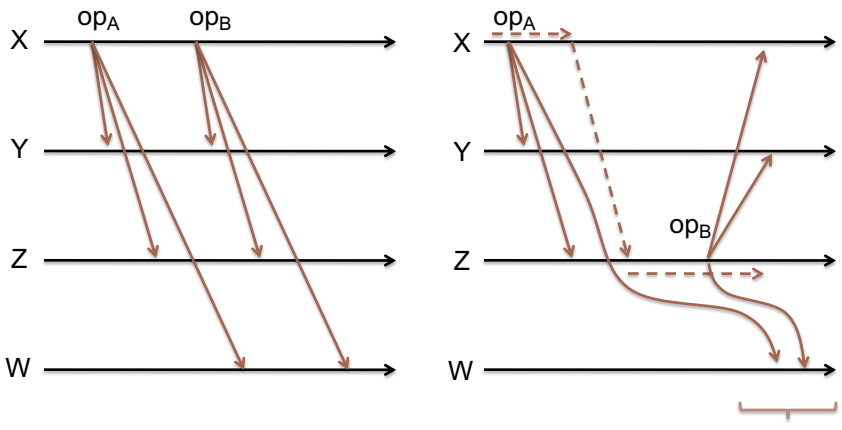
Recall: Causal Message Delivery



25

25

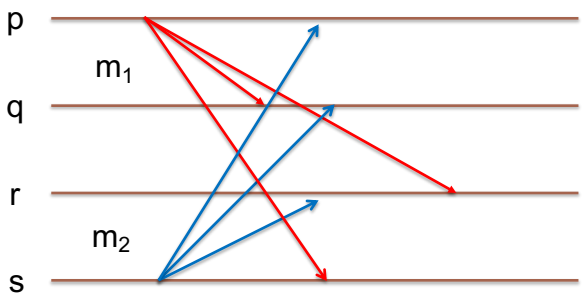
Causal Broadcast Generalizes FIFO



26

26

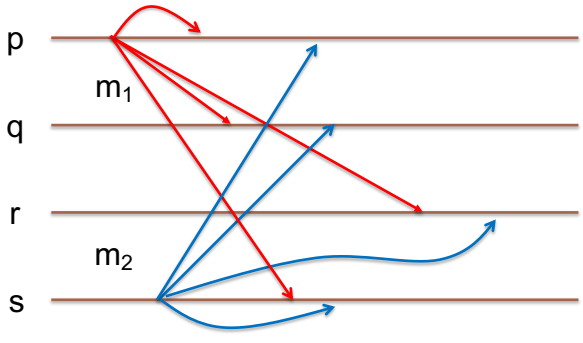
Causal but not Total



27

27

Total Ordered Multicast

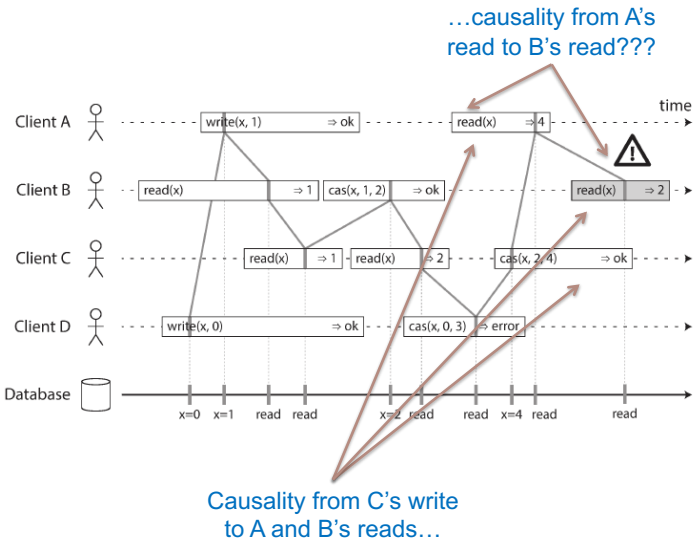


Global replicated totally ordered log of messages

28

28

Linearizable vs Causal vs Total



29

Linearizable vs Causal vs Total

- Linearizable ⇒ Causal, but not vice versa
- …but causal may be enough!
- …performance similar to eventual consistency

30

30

Linearizable vs Causal vs Total

- Can implement Linearizable with Total
- …broadcast update to all replicas
- …more work to prevent stale reads
- …send yourself a message before reading (quorum read)
- …or read from synchronously updated follower

31

31

Linearizable vs Causal vs Total

- Can implement Total with Linearizable
- …linearizable atomic increment-and-set
- …atomically update broadcast counter

32

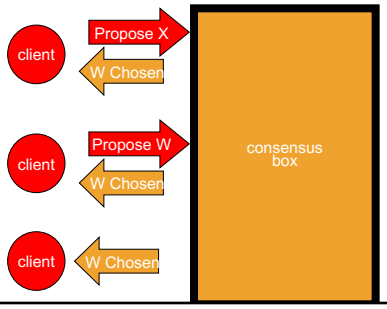
32

Linearizable vs Total vs Consensus

- Can implement Consensus with Linearizable
- …linearizable atomic increment-and-set

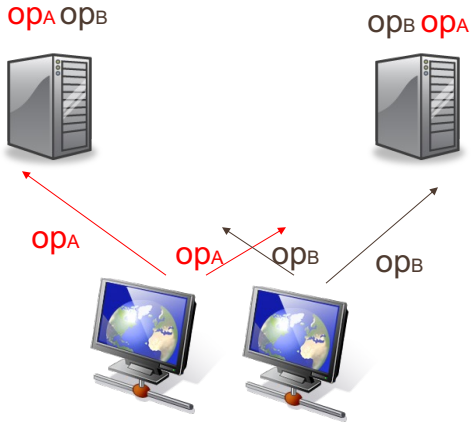
- Can implement Linearizable with Consensus
- …Paxos: Consensus Box

- **Linearizable**
  = Total
  = Consensus!



33

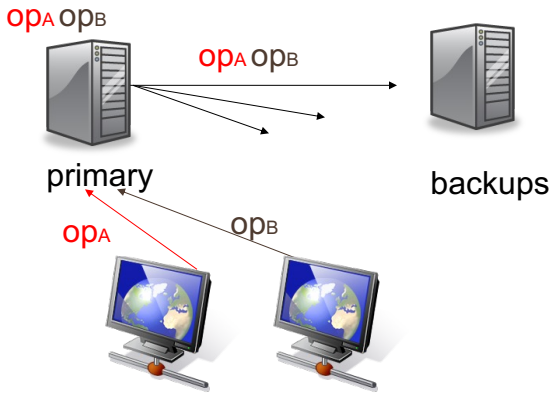
State-Machine Replication



34

34

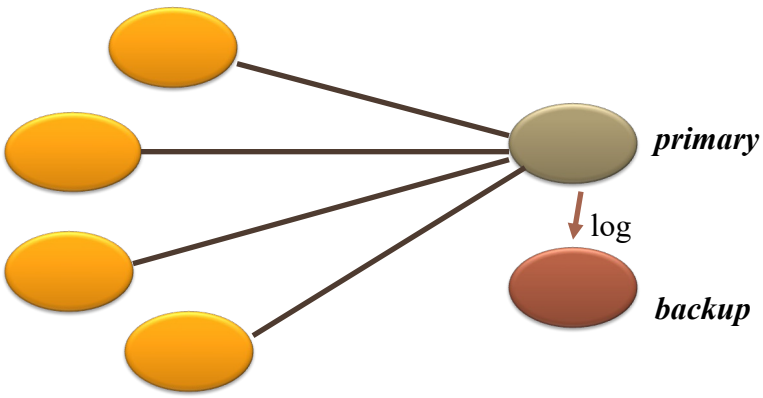
Primary-Backup Replication



35

35

Primary/backup

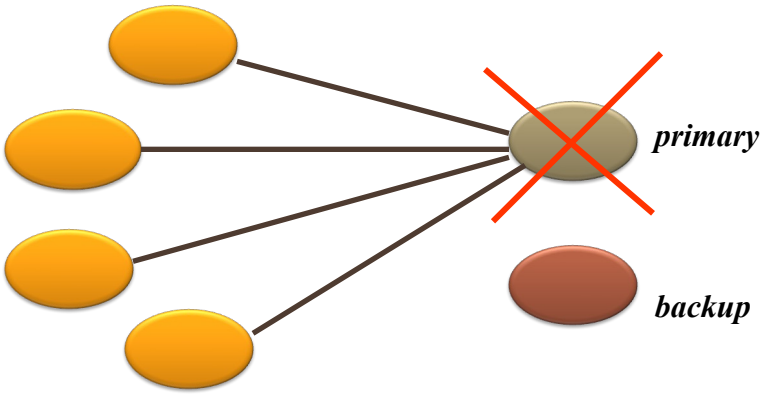


Clients initially connected to primary, which keeps backup up to date. Backup tracks log

36

36

Primary/backup

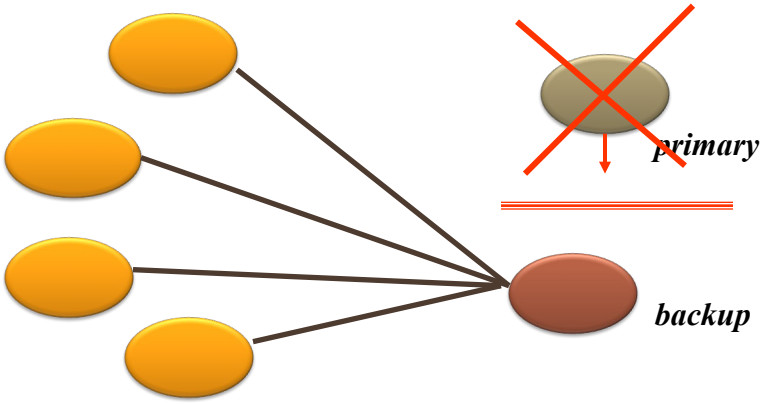


Primary crashes. Backup sees the channel break.

37

37

Primary/backup



Clients detect the failure and reconnect to backup.

38

38

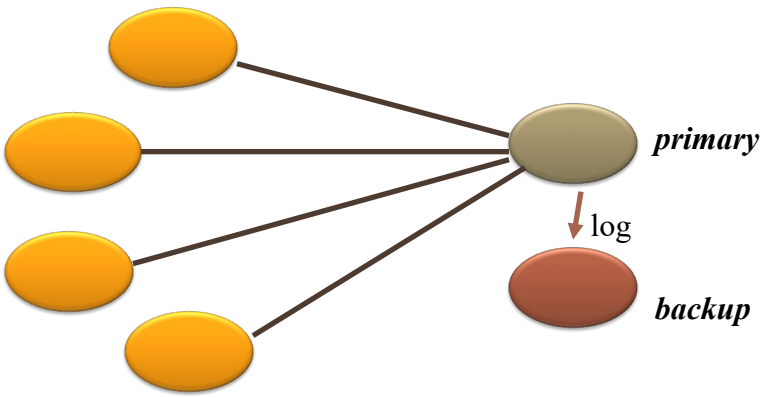
Issues

- How do we ensure agreement on order of updates?
- Under what conditions should backup take over?
  - "Split brain" problem

39

39

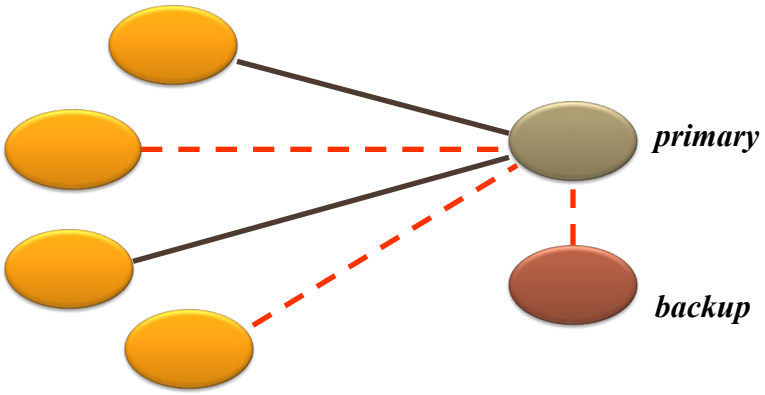
Split brain



40

40

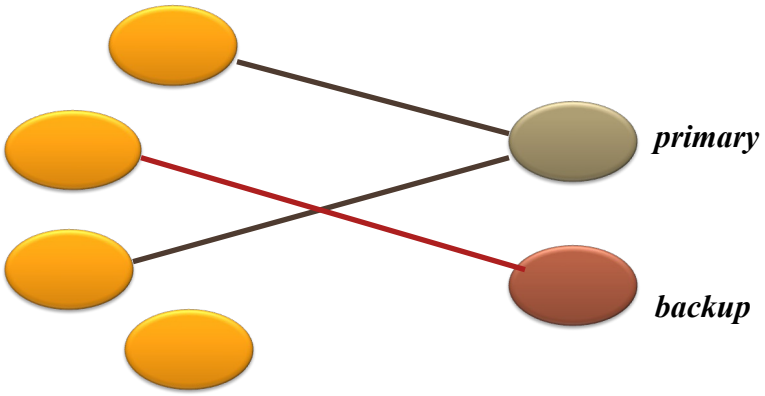
Split brain



41

41

Split brain



42

42

Solutions

- Single server with restart
- Allow backup to “kill” the primary
  - Process groups membership service
- “Majority vote”
  - Quorum consensus

43