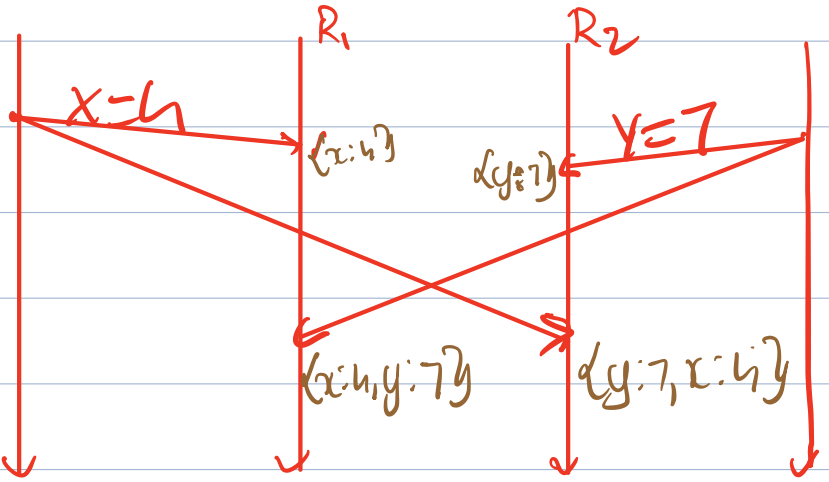
Agenda

- Eventual consisentency
- Strong convergence & SEC
- Intro to app specific conflict resolutions
- Network partitions
- availability
- CAP

Strong consistency & if you want fault tolerance, you need a consensus algorithm.

Is this strong consistency?



No! Client can read between both writes, & can tell that data is being replicated.

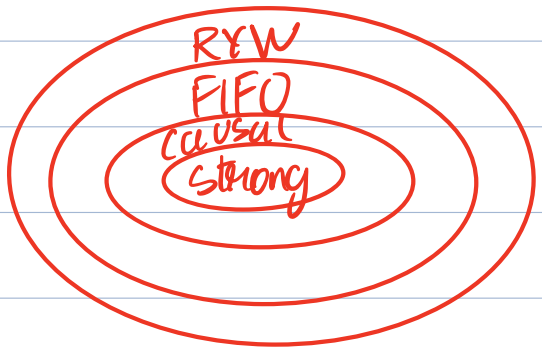
However, data on both replicas ends up matching later. This is called Eventual consistency!

Eventual consistency:

Replicas eventually agree, if clients stop submitting updates

- This is a liveness property i.e it cannot be violated in a finite execution.

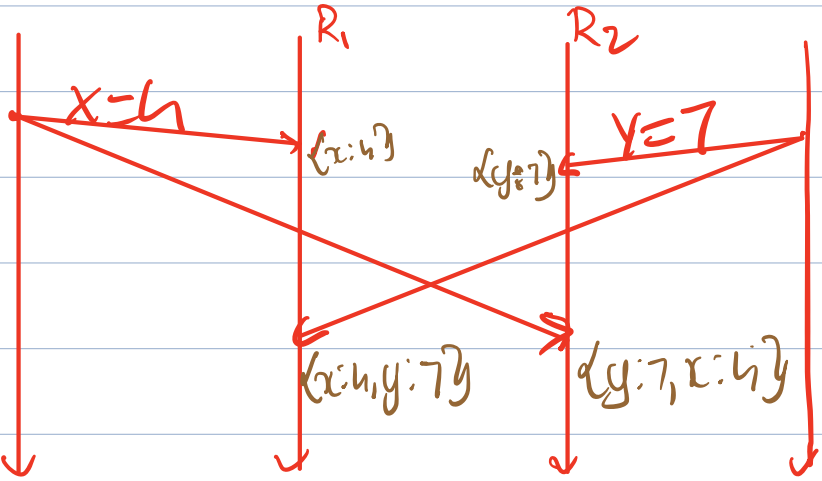
Consistency guarantees are safety properties



eventual consistency is NOT a safety property

Strong convergence (safety property)

Replicas that have delivered same set of updates have equivalent state



satisfies strong convergence.

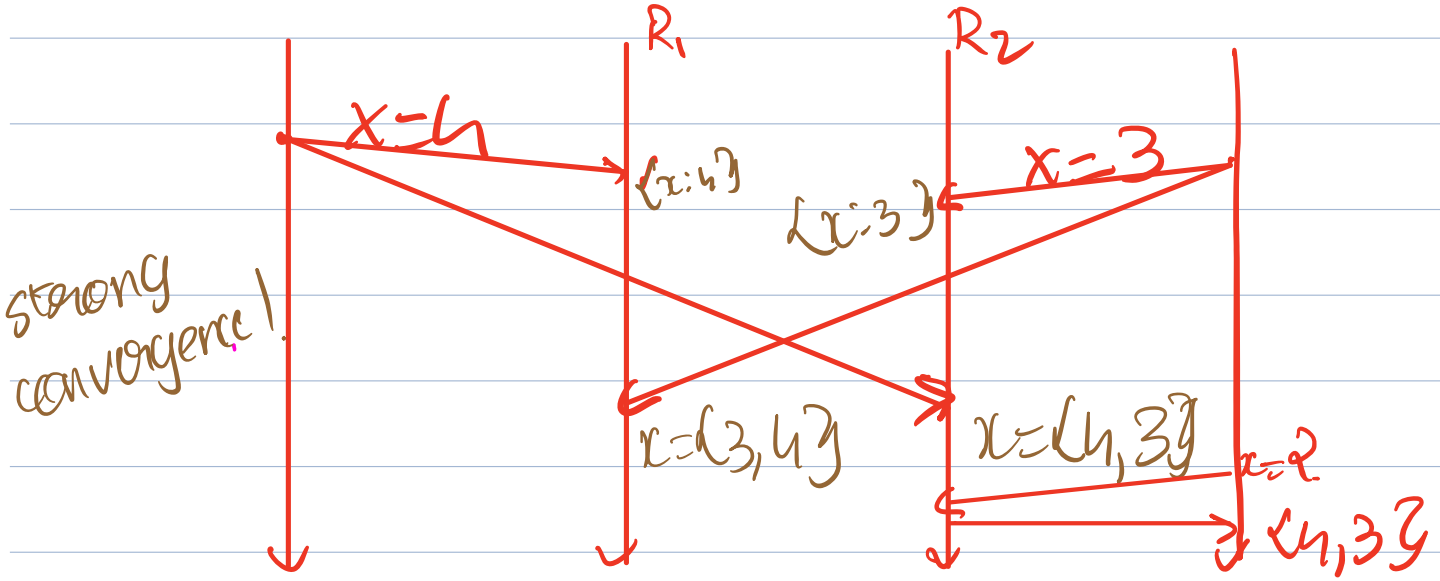
Strong eventual consistency

- System has both eventual consistency & strong convergence
- It is a combination of a liveness & safety property.

Read DynamoDB paper

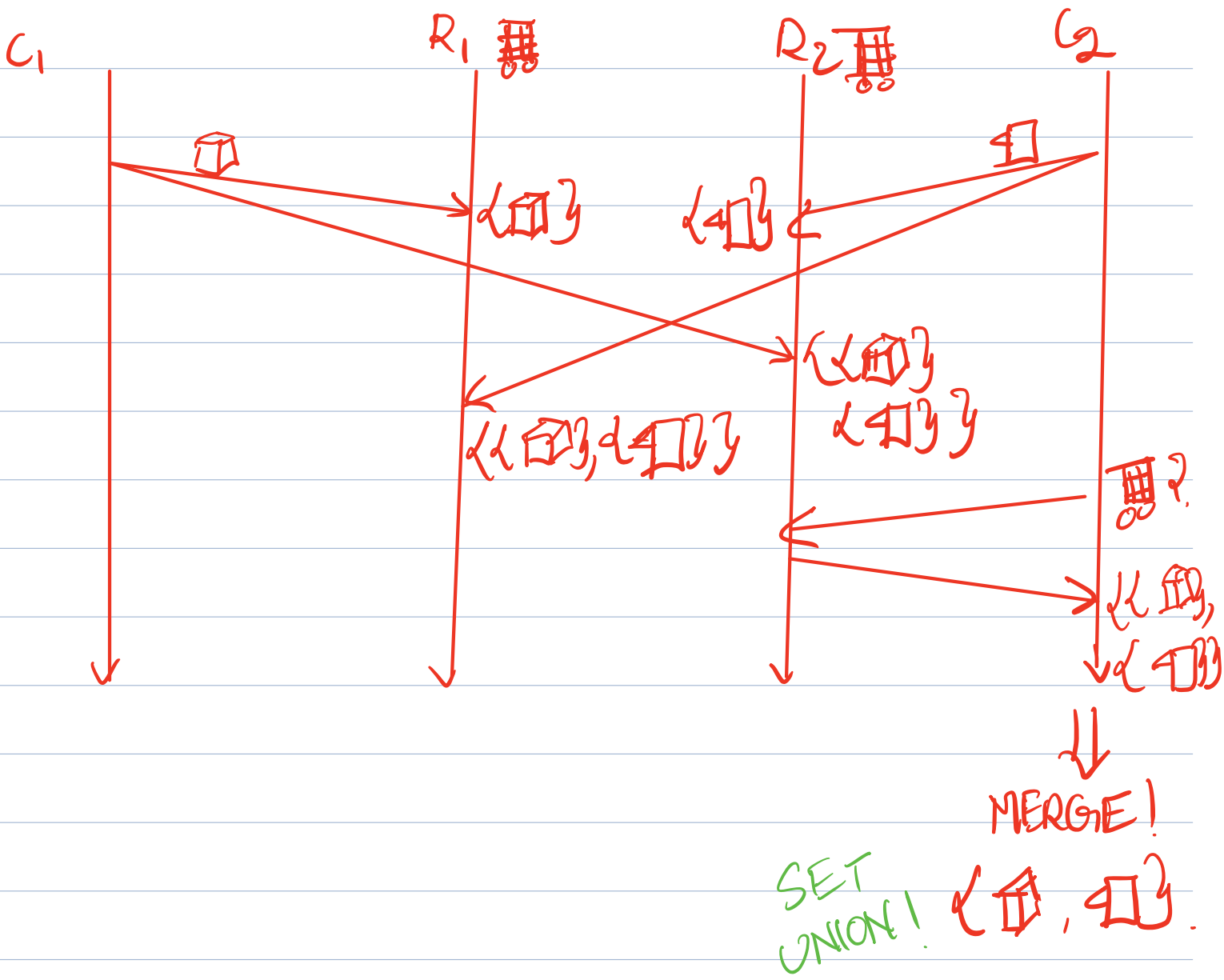
Dynamo:

- Eventual consistency.

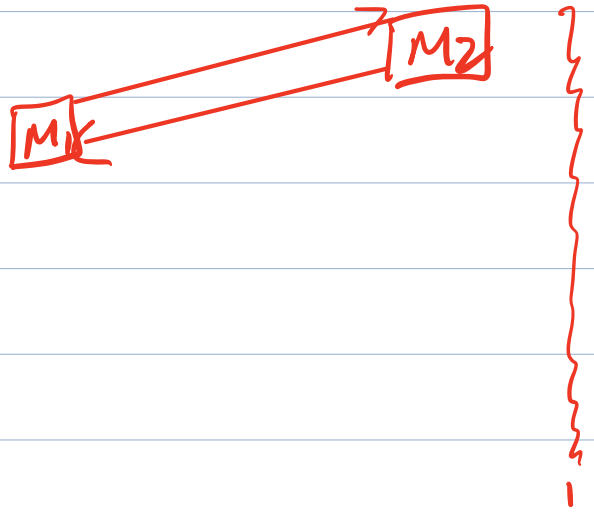


- Clients can read, receive both values & decide how to do conflict resolution.

Application specific conflict resolution. (Amazon)



Network Partitions

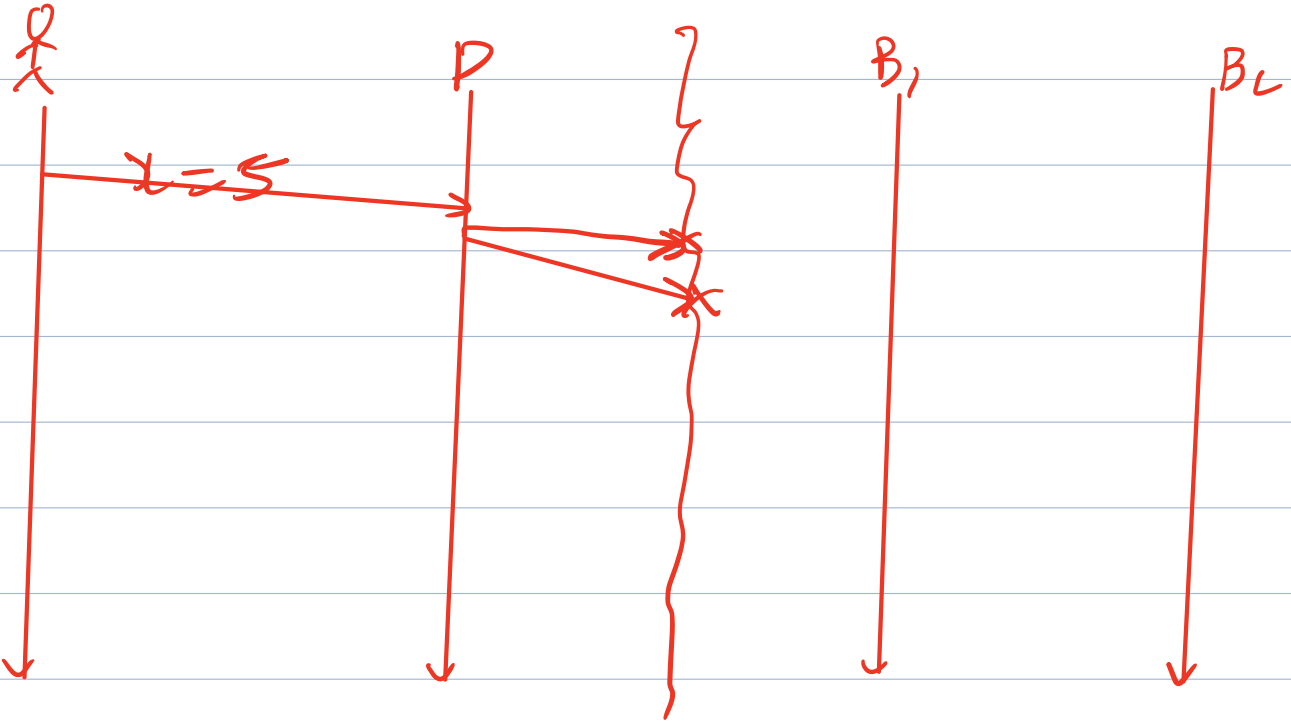


when there are some subset of nodes in the network that cannot comm with other nodes

in the network.

Availability.

- Every request receives a response
- Liveness property



Primary Backup chooses consistency over availability

Dynamo chooses availability over consistency.

CAP (Consistency, Availability, Partition tolerance)

- In practical dist sys, now will almost always

have partition tolerance, & you need to make a trade-off between consistency & availability.