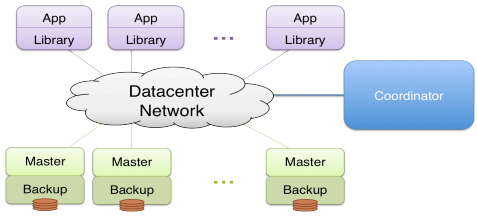
Fault Tolerant Cluster Coordination in RAMCloud

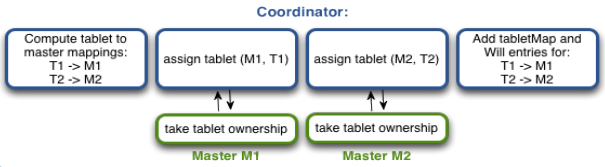
Ankita Kejriwal, Diego Ongaro, Ryan Stutsman, Steve Rumble, Mendel Rosenblum, John Ousterhout

Coordinator in RAMCloud

- Manages cluster membership and tablet configuration
- Stores core metadata



- Coordinator affects state of other nodes in cluster
  - Example: create table (that has tablets T1 and T2)



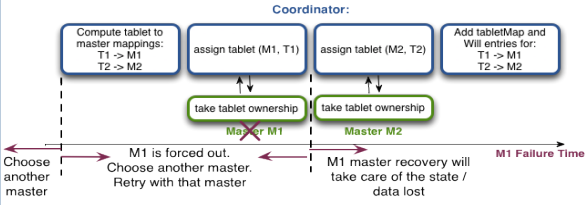
Overall Goal

Atomic distributed state change

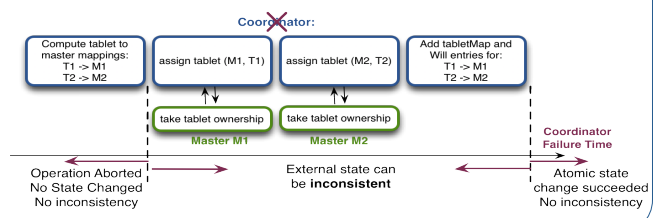
Why is this hard?

- Machines can fail
- Distributed state change no longer atomic
  - Can result in inconsistent state

Master Failure

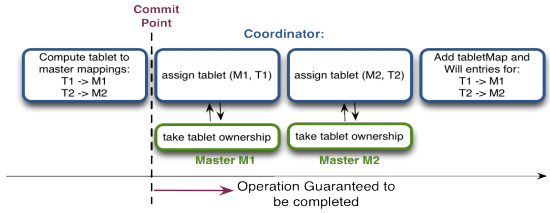


Coordinator Failure



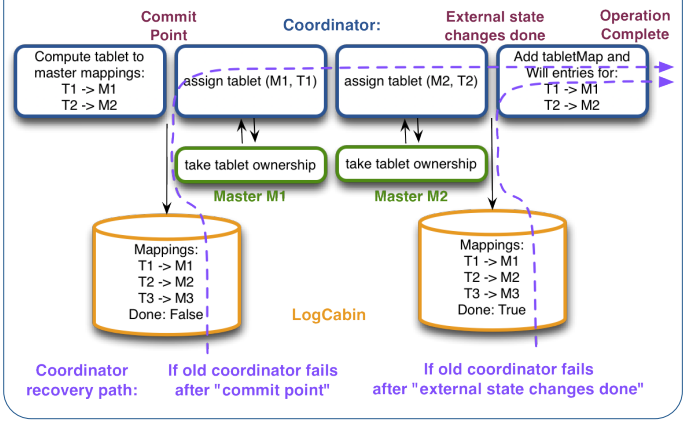
Coordinator Design

- For every coordinator operation, we can define a **commit point**, such that if the coordinator failure occurs:
  - Before commit point – operation aborted
  - After commit point – operation is guaranteed to be completed



- Leave enough information around so that the new coordinator can roll forward the operation to completion
- Persist this information across failures
  - Use a highly replicated, consistent storage service: LogCabin
  - LogCabin provides abstractions to append to and read from a (highly reliable distributed) log
  - Log the **state** at commit point and at completion
  - The new coordinator can infer the actions to be done from this state

Coordinator Design with Recovery Paths



SEDCL Retreat
June 7 – 8, 2012