Scalable Systems

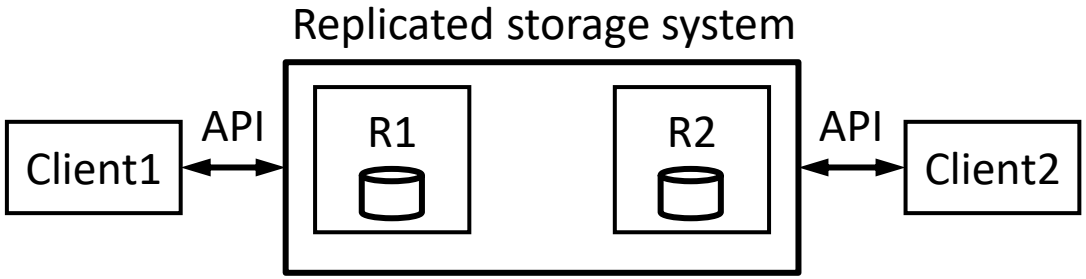
Ashvin Goel

Electrical and Computer Engineering
University of Toronto

Distributed Systems
ECE419

Fault-tolerant replicated systems

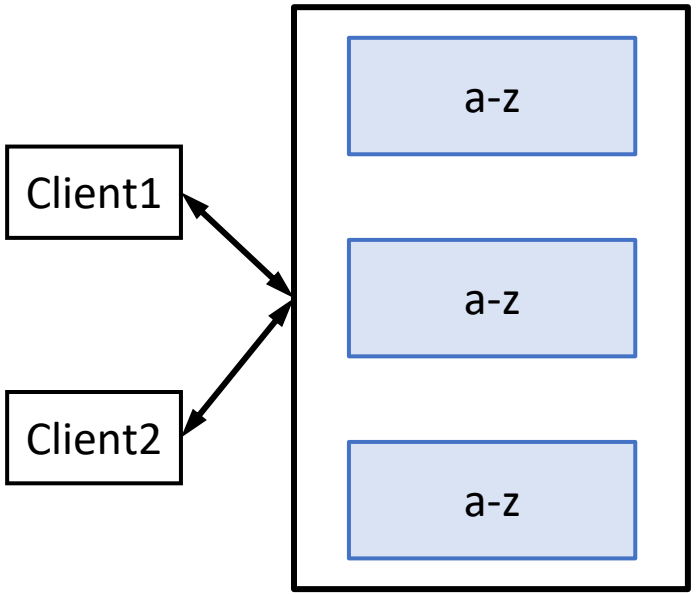
- Replicated systems provide fault tolerance
  - Ideally, look like one reliable server



- We have seen systems that replicate data across all nodes
  - E.g., Raft, ZooKeeper

Fault-tolerant replicated systems

- Replicated systems provide fault tolerance
  - Ideally, look like one reliable server



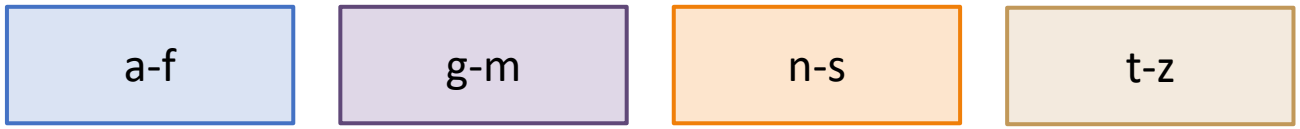
Replicated storage system

Issues with replication

- We have seen systems that replicate data across all nodes
  - E.g., Raft, ZooKeeper
- Will the performance of such a replicated system improve with increasing number of replicas?
  - E.g., compare 3 replicas with 5 replicas
- For read-mostly workloads
  - No, if latest data needs to be read, why?
  - Yes, if reading stale data is okay, why?
- For write-heavy workloads
  - No, why?
- How can we improve scalability?

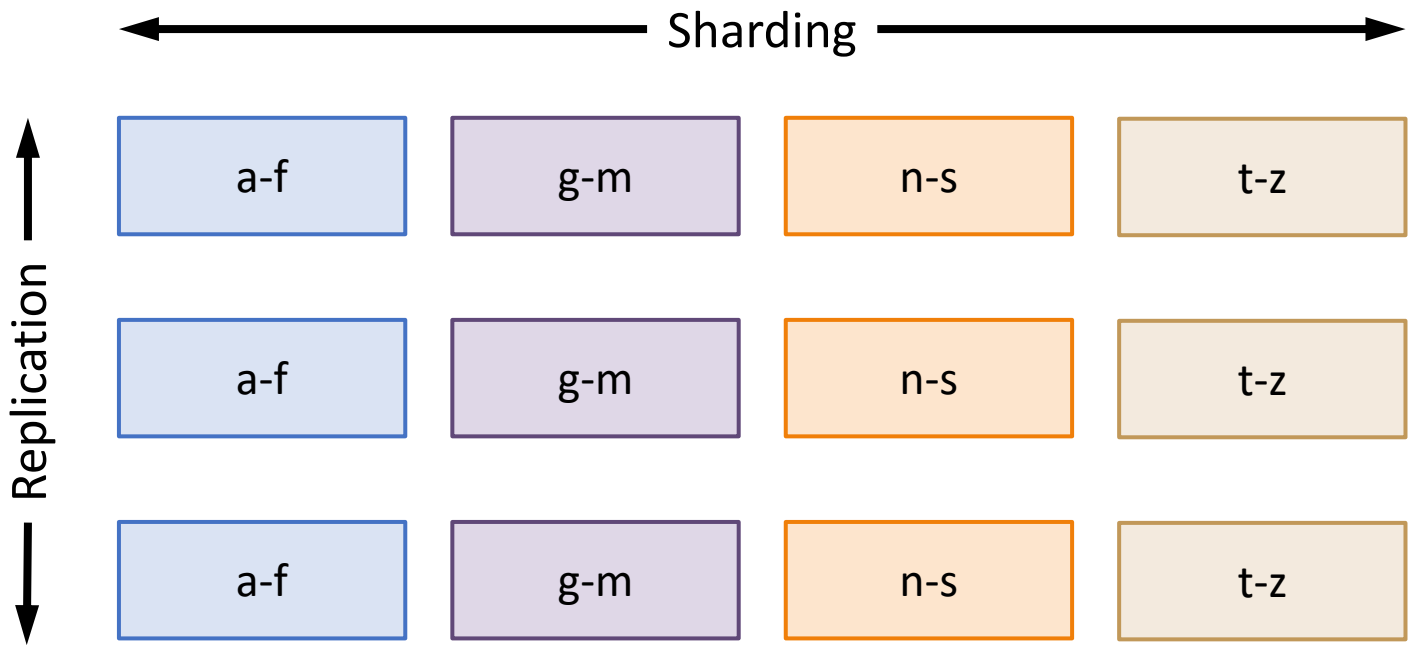
Sharding

- We need to partition (shard) data across nodes
- Why does sharding improve scalability?
- Does sharding provide fault tolerance?



Combining sharding with replication

- Sharding for scalability
- Replication for fault tolerance



Case studies of real systems

- Next, we will look at scalable and highly available production systems
  - Use sharding for scalability
  - Use replication for fault tolerance
  - Use weak consistency for high availability