# CSE 550: Systems for all

Au 2022

Ratul Mahajan

# Distributed transactions

Atomic update of data across multiple nodes
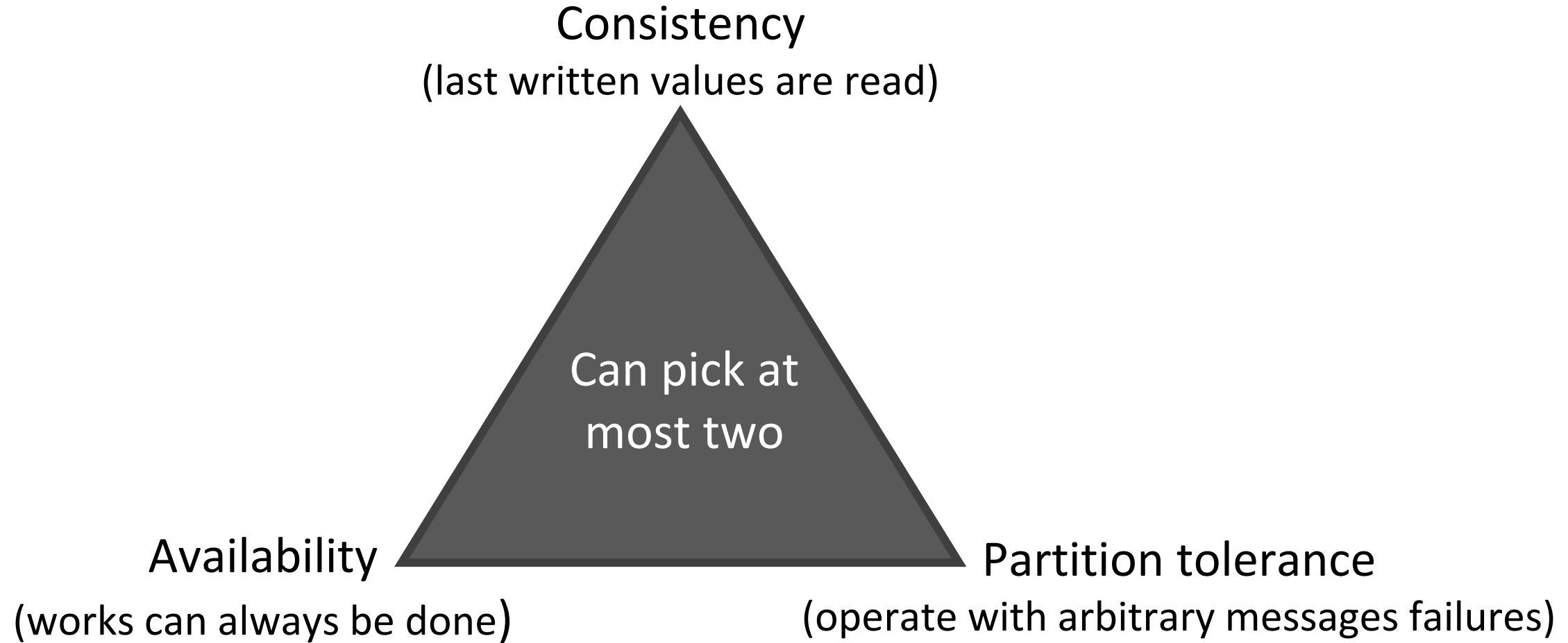
Why is the data distributed?
- Replication: Tolerate failures
- No replication: Need more compute / storage

# Fault tolerance in distributed systems

|  | Replication | Non-replication |
|---|---|---|
| **Fail stop** | Paxos | 2PC, 3PC |
| **Byzantine** | PBFT | N/A – why? |

# CAP theorem: Fundamental trade-off for fault tolerance

Consistency
(last written values are read)

Can pick at
most two

Availability
(works can always be done)

Partition tolerance
(operate with arbitrary messages failures)

Over to Jiacheng and Yuan-Mao

# Mid-quarter feedback

What is helping

- Lectures, papers, assignments

What can be improved

- Student presentations
- Online discussions
- More context on the readings on the Website

# Rest of the quarter

Thus far, learned many of the fundamental techniques
- Sharing resources, ordering events, handling failures, building large networks

Now, focus shifts to applications
- Data stores (files, KV stores, structured data)
- Data computation (streams, dataflow, ML)

Some fundamentals still coming
- Security, correctness reasoning

# Project

Proposal feedback is on Canvas

Email/call/chat if you have questions

The project has two goals (Hint: final report must reflect this)
- Learning: What did *you* learn from the project?
- Research: How did you advance the body of knowledge?