Chemistry behind Agreement

Suyash Gupta

SkyLab
UC Berkeley
gupta-suyash.github.io

Mohammad Javad Amiri

UPenn
seas.upenn.edu/~mjamiri

Mohammad Sadoghi

ExpoLab
UC Davis
expolab.org

Berkeley UNIVERSITY OF CALIFORNIA

Penn University of Pennsylvania

UCDAVIS

ResilientDB

- **What is this talk about?**

  Agreement protocols.

- **What is an agreement protocol?**

  Helps to reach multiple parties a common decision.

- **Why agreement?**

  Distributed systems with multiple nodes are common.

- **Any real-world application?**

  Every distributed database system!

Agreement Protocol Types

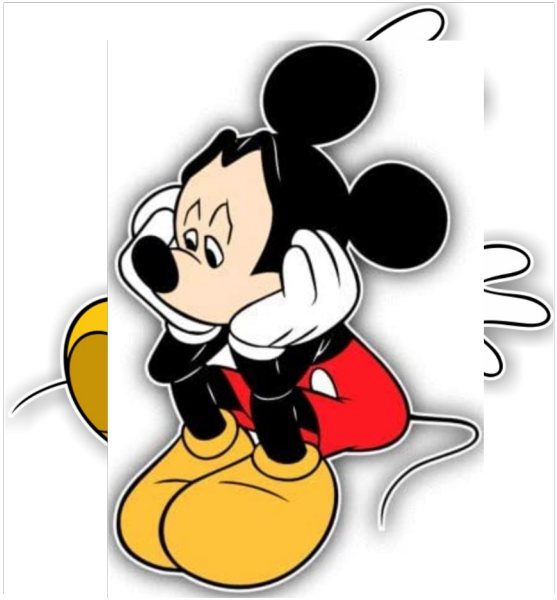
- **Commit Protocols**
  - Agreement on transaction commit or abort.
  - Two-phase commit, Three-phase commit.
- **Crash Fault-Tolerant (CFT) Protocols**
  - For consistent replication under crashes.
  - Paxos, Raft.
- **Arbitrary Fault-Tolerant (AFT) Protocols**
  - For consistent replication under arbitrary faults (e.g. malicious).
  - PBFT, PoE.

New Protocols are still in Production

- **BFT Protocols**
  - **GeoBFT [VLDB'20], Sharper [Sigmod'21], ByShard[VLDB'21], RCC [ICDE'21], PoE [EDBT'22], ServerlessBFT [ICDE'23]**

- **Commit Protocols.**
  - **EasyCommit [EDBT'18], QStore [EDBT'20]**

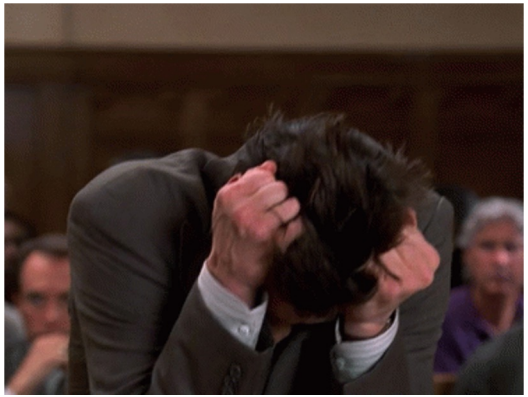
So Are we done?



Unfortunately No!

Challenges Due to Disparity

- **Incompatible algorithmic designs**
- **Distinct schematic representations.**
- **Lack of common proof systems.**



Disparity hurts Adoption

Exciting Prior Works

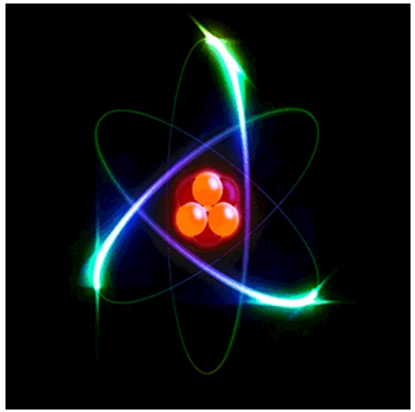
- **Calvin [SIGMOD'12], Tapir [SOSP'15]**, and **Janus [OSDI'16]** combine commitment and CFT.
- **Deneva [VLDB'17]** framework helps to express different CC techniques.
- **Sujaya et al. [VLDB'19]** present a framework to explain a subset of commitment and CFT protocols.
- **DataCalculator [SIGMOD'18]** presents a unified framework for data-structures.

Our Prior Attempt: Unifying AFT Protocols

Open sourced at https://resilientdb.com/

Vision:
Unified Elemental Framework

Atoms, Elements and Compounds of Agreement.



Atoms

- Smallest indivisible unit of an element.
- Atoms define functional properties of an agreement protocol.

Atoms

- **Failure**

  Crash failure, unexpected restart, or malicious attack.

- **Quorum Size**

  n-1 (2PC), f+1 (Paxos), 2f+1 (PBFT).

- **Topology**

  star (centralized), clique (decentralized), ring (chain).

- **Data Distribution**

  data sharding and/or replication.

Elements

- Composed of one or more atoms.
- Represent the phases of an agreement protocol.

Elements

- **Proposal (P)**
  - Proposal sent by a leader that includes a client transaction.
- **Vote (V)**
  - A node's vote on the leader's proposal.
  - Commit protocols → abort or commit vote.
  - AFT protocols → support for only valid proposal.
- **Prepare (Pp) and Commit (Co)**
  - Leader attempts to inform nodes about common decision.
  - Not all protocols require both the elements.

Elements

- **Execution (X)**
  - Execution of client transactions.
  - Order-then-execute vs. Execute-then-order.
- **Checkpoint (Ch)**
  - State exchange to ensure a common state across nodes.
- **Leader Election (Le)**
  - Replacement of current leader when it fails.
  - New leader is expected to help commit the current proposal.

Agreement Protocols: Compounds of Elements and Atoms

Elemental Protocols

2PC: ⟨ Pr —— V‡ —— Co —— X° ⟩

3PC: ⟨ Pr —— V‡ —— Pp —— V‡ —— Co —— X° ⟩

Paxos: ‖Pr —— V —— Co —— X°‖

PBFT: ‖Pr —— V —— Pp —— V —— Co —— X°‖

Elemental Protocols

DPaxos: $\| \; Pr \; — \; Co^{\oplus} \; — \; X^{\circ} \; \|$

DPBFT: $\| \; Pr \; — \; Pp^{\oplus} \; — \; Co^{\oplus} \; — \; X^{\circ} \; \|$

What's More?

- **Reduced Phase Consensus protocols.**

  SpecPaxos, Zyzzyva, PoE

- **Multi-Leader (parallel) consensus protocols.**

  Mencius, RCC

- **Global-scale consensus protocols.**

  GeoBFT, Steward, GEC, Ziziphus

- **Sharded-replicated consensus protocols.**

  Spanner, MDCC, Sharper, RingBFT, ByShard

Conclusions and Future Work

Our vision is to design a framework that unifies different agreement protocols and prevents future disparities.

- Designs untouched: deterministic protocols, asynchronous protocols, node recovery and reconfiguration, DAG-based ordering.
- Unifying framework should permit arguing about properties like totality, validity, consistency, and termination.

Thank You