UC Santa Barbara

Computer Science Department

Blockchains and Databases: Opportunities and Challenges for both the Permissioned and the Permissionless

Amr El Abbadi

University of California, Santa Barbara

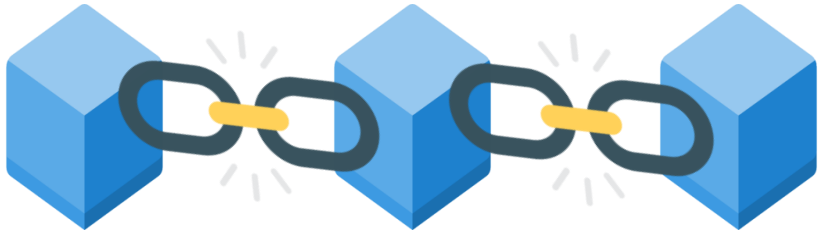
In Collaboration with:

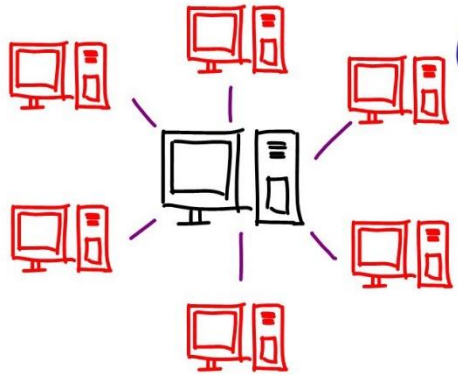
Mohammad Amiri, Sujaya Maiyya, Victor Zakhary, and Divyakant Agrawal.

Two Main Underlying Themes.

Theme One

Blockchains





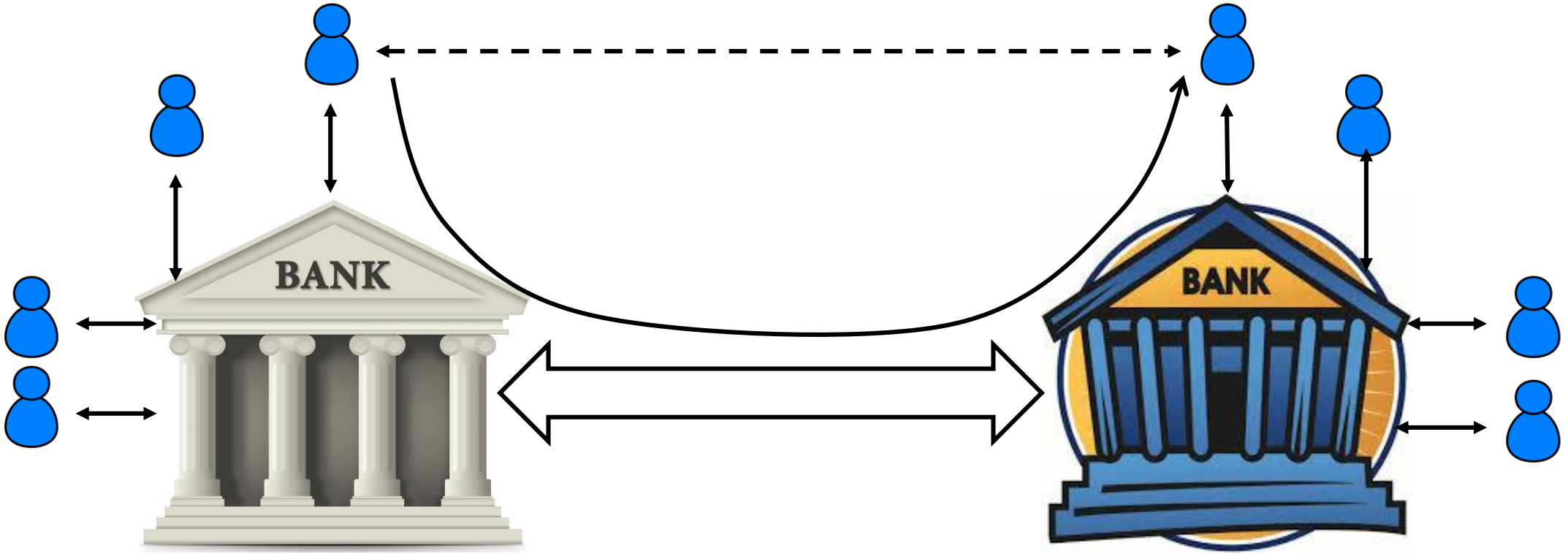
Databases

Theme Two

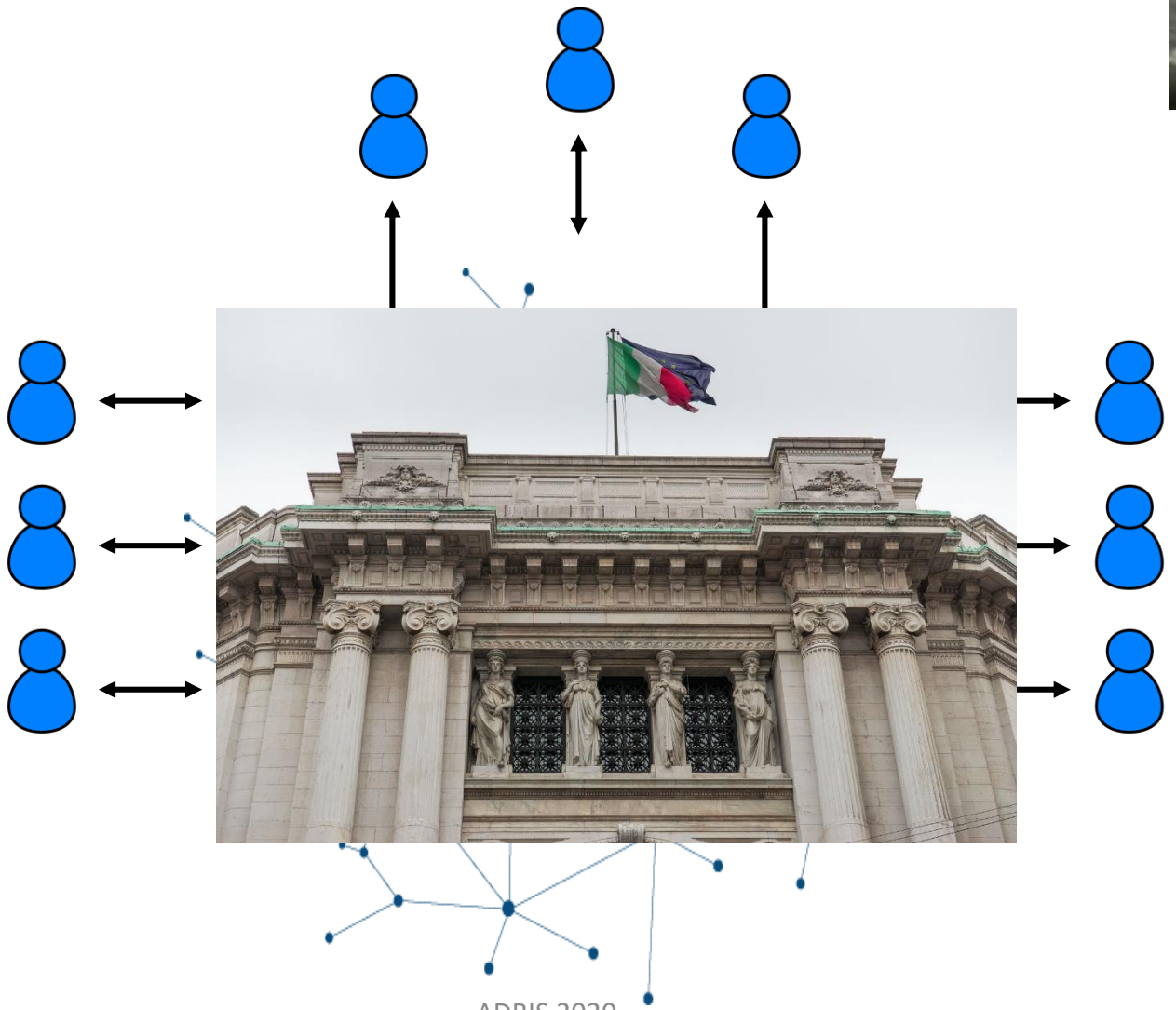
- **Failures were benign**

But in the Real World Failures can be Malicious

Origins of Blockchain: Traditional Banking Systems

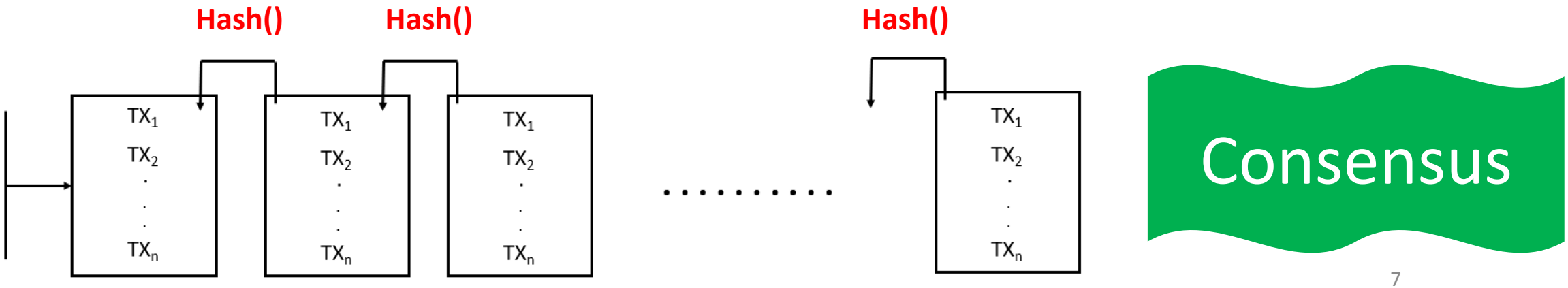


Bitcoin



What is a Blockchain?

- Transactions are grouped into blocks
- Blocks are chained to each other through hash pointers
  - This guarantees that the ledger tamper-free.
- To make progress:
  - Network nodes **validate** new transactions are consistent.
  - Network nodes need to agree on next block to add to blockchain



Reach Consensus Using Mining
Replace Communication with Computation!!

Permissionless Blockchains have Unknown Number of Participants

Atomicity Across Permissionless Blockchains

The Landscape

Cryptocurrencies: 2225 • Markets: 18851

Market Cap: $257,486,187,861 • 24h Vol: $66,548,083,112

Search

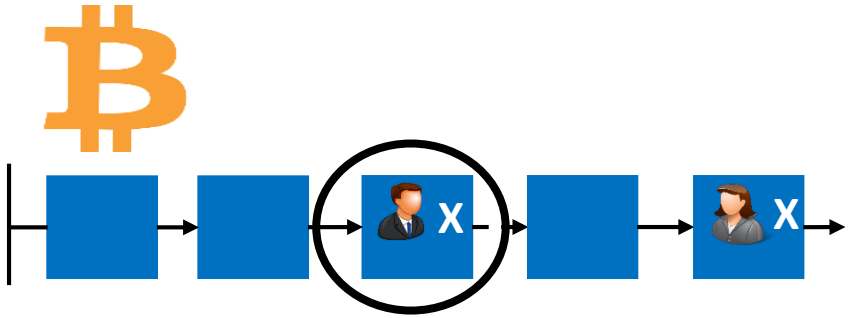
ion

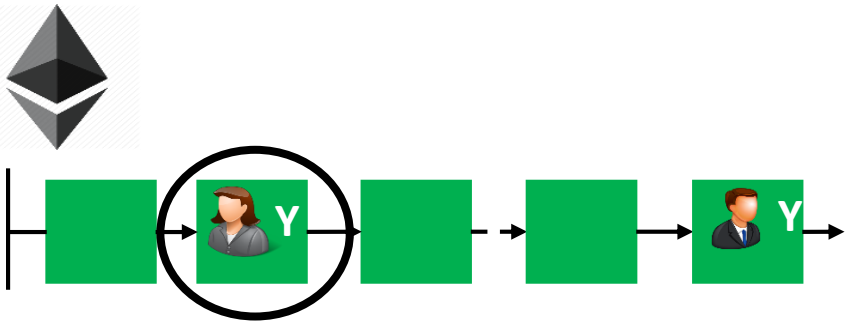
Cryptocurrencies ▾ | Exchanges ▾ | Watchlist | USD ▾ | Next 100 → | View All

#	Name	Market Cap	Price	Volume (24h)	Circulating Supply	Change (24h)	Price Graph (7d)
1	Bitcoin	$142,627,334,795	$8,036.77	$19,138,268,181	17,746,837 BTC	3.15%	
2	Ethereum	$26,732,290,299	$251.25	$8,364,736,132	106,397,463 ETH	1.70%	
3	XRP	$17,876,222,703	$0.423217	$1,658,461,942	42,238,947,941 XRP *	1.25%	
4	Litecoin	$7,281,728,951	$117.21	$5,141,138,982	62,124,551 LTC	6.28%	
5	Bitcoin Cash	$7,157,820,741	$401.55	$1,572,103,916	17,825,688 BCH	2.02%	

Source: coinmarketcap.com on June 7th 2019 at 5:00pm PST

Cross-Chain Transaction Example



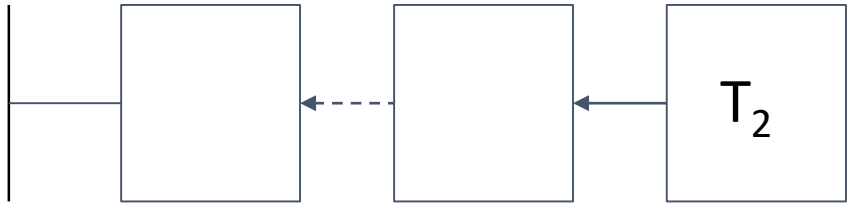


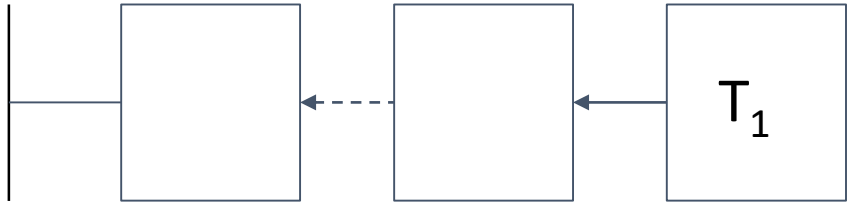
Atomic Cross-Chain Commitment Protocol

Swap of Ownership

Atomic Swap Example
[Nolan'13, Herlihy'18]

- Alice wants to trade Bitcoin for Ethereum with Bob
- Uses Smart Contracts to deposit currency in blockchain
- Uses Secret Hashes to ensure exchange of deposits
- Uses Timout Locks to overcome malicious behavior.





Ethereum blockchain

Bitcoin blockchain

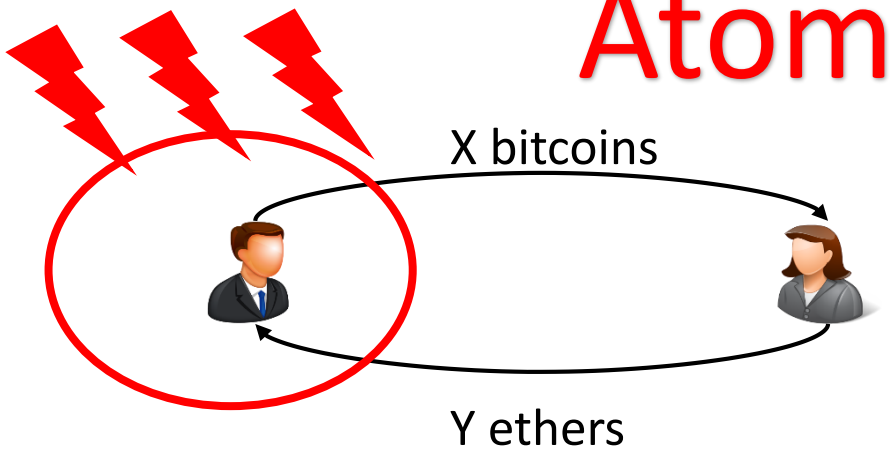
Bob

Alice

What can go wrong?

If Bob fails or suffers a network denial of service attack, Alice's contract will expire and Bob will lose his X bitcoins

Atomicity Violation



Atomicity Violation

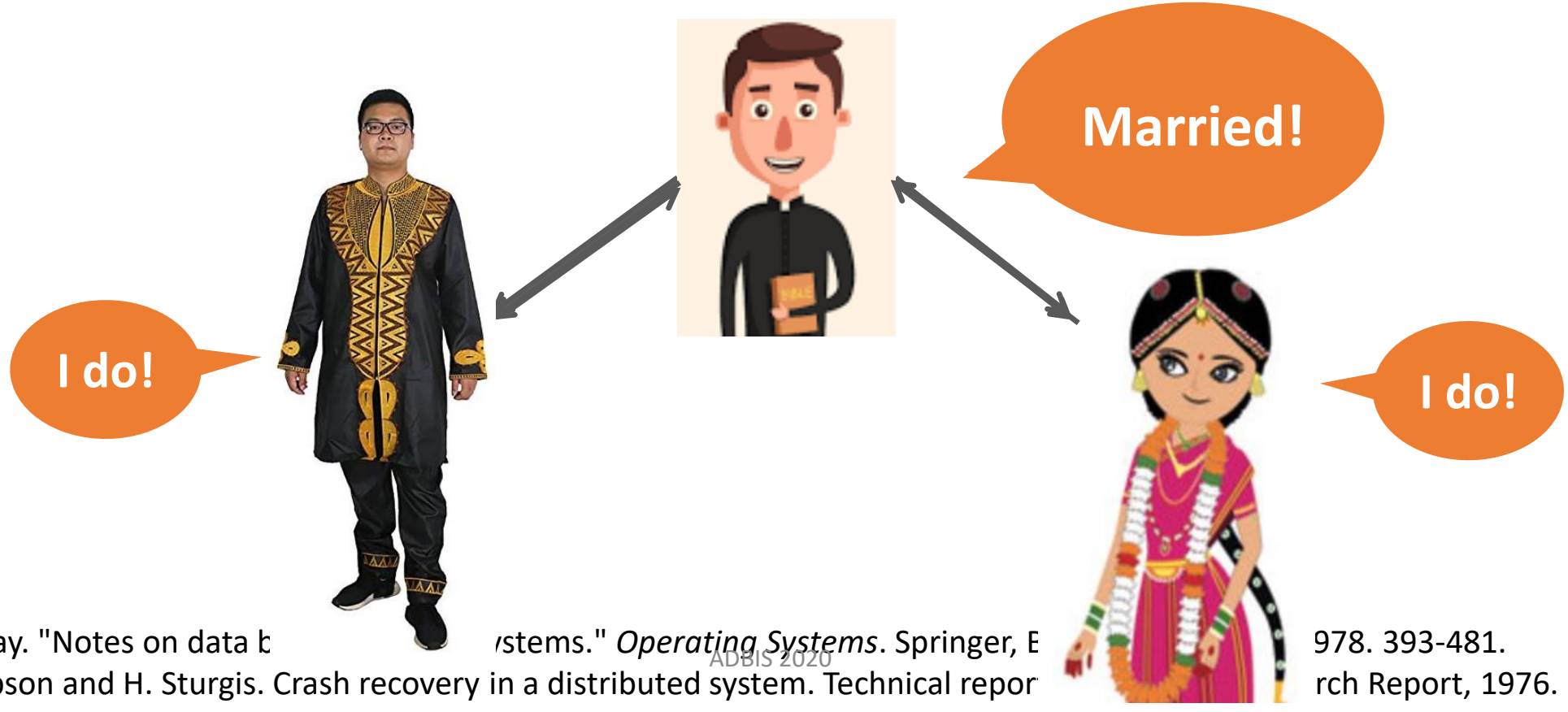
- Using timelocks leads to Atomicity violation
- Our Atomicity-based Approach:
  - The decision of both transactions should be made atomic
    - Once decision is taken, both transactions either commit or abort
- Upcoming VLDB 2020 paper.
  - Note: there is a concurrent paper also in VLDB 2020 by Herlihy, Shrira and Liskov on cross-chain deals.

Victor Zakhary, Divyakant Agrawal, Amr El Abbadi, Atomic Commitment Across Blockchains. VLDB 2020.

Atomic Commitment in Databases

Two Phase Commit

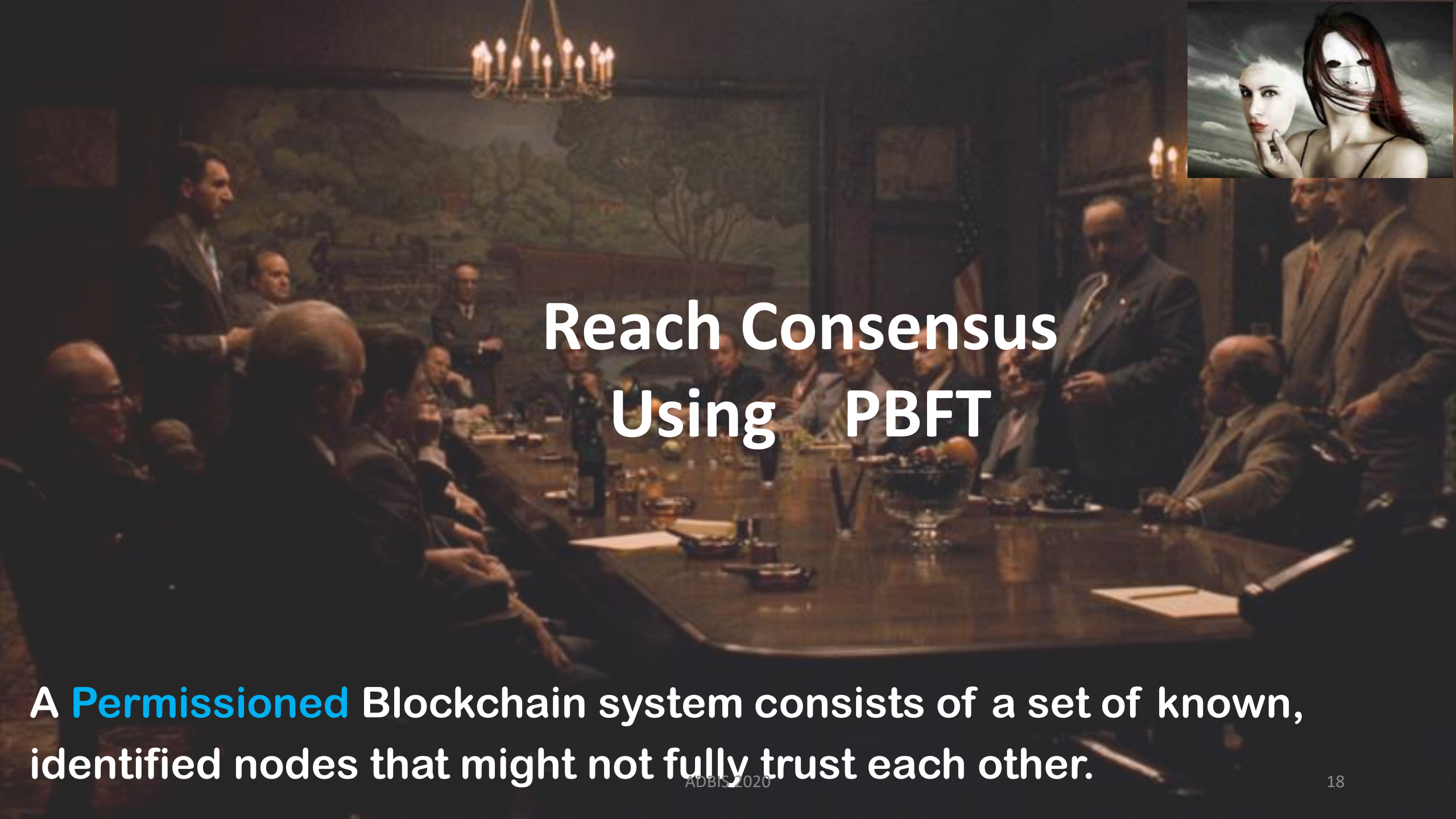
- 2PC [1,2] is ***atomic commitment*** protocol: either all servers commit or no server commits



[1] J. N. Gray. "Notes on data b ystems." *Operating Systems*. Springer, E 978. 393-481.
[2] B. Lampson and H. Sturgis. Crash recovery in a distributed system. Technical repor rch Report, 1976.

Atomic Commitment Across Blockchains

- Use another blockchain to witness the Atomic Swap
- The witness blockchain decides the commit or the abort of a swap
- Once a decision is made:
  - All sub-transactions in the swap must follow the decision
  - Achieves atomicity, either all committed or all aborted
- How can miners of one blockchain verify a transaction in another blockchain?
  - Without maintaining a copy of this other blockchain.
  - Use cross chain verification.
- Cross chain verification is leveraged twice
  - Miners of the witness network verify the publishing of contracts in asset blockchains
  - Miners of assets' blockchains verify the decision made in the witness network
- Details in paper [VLDB 2020]



Reach Consensus Using PBFT

A Permissioned Blockchain system consists of a set of known, identified nodes that might not fully trust each other.

Consensus

- A ***consensus*** protocol: agreement on a single value
- Bigtable, Spanner [Google], etc.

PAXOS

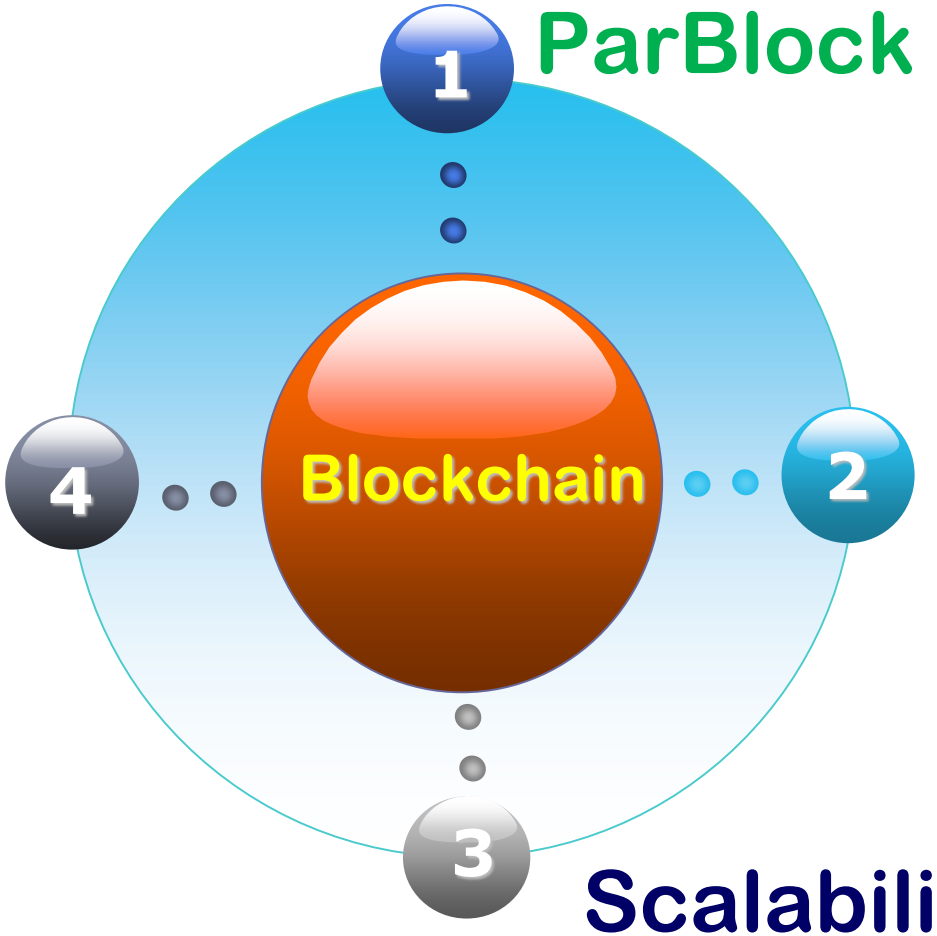
[Lamport]

Agree on next block

PBFT

[Castro and Liskov]

Permissioned Blockchains



Performance
ParBlock [ICDCS 2019]

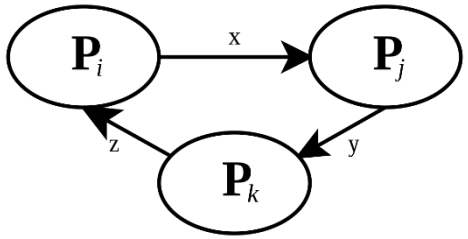
Confidentiality
Caper [VLDB 2019]

Scalability
ShaPer

Verifiability
Separ

Blockchain Performance

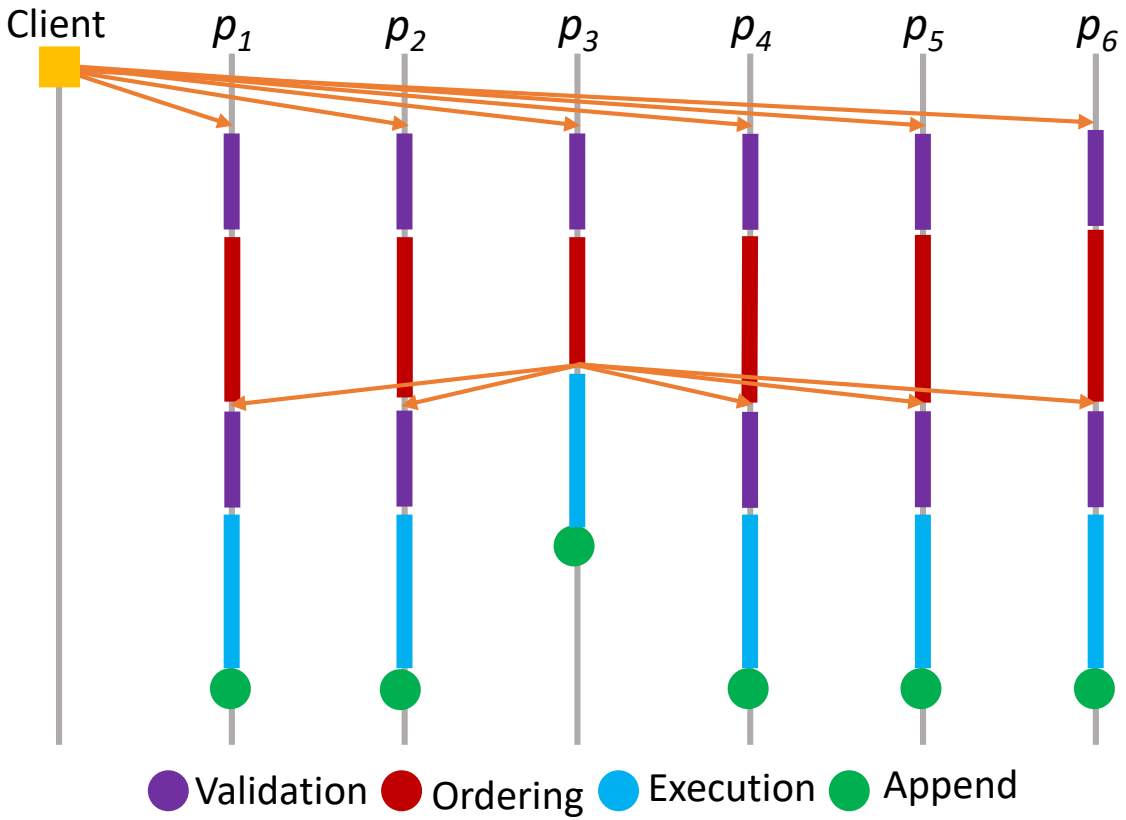
Executing Transactions in Databases



Pessimistic Locking Protocols	Obtain locks first, and determine a priori execution order	Execute last
Optimistic Concurrency Control	Executes first	Validates read-write conflicts last and abort conflicting transactions

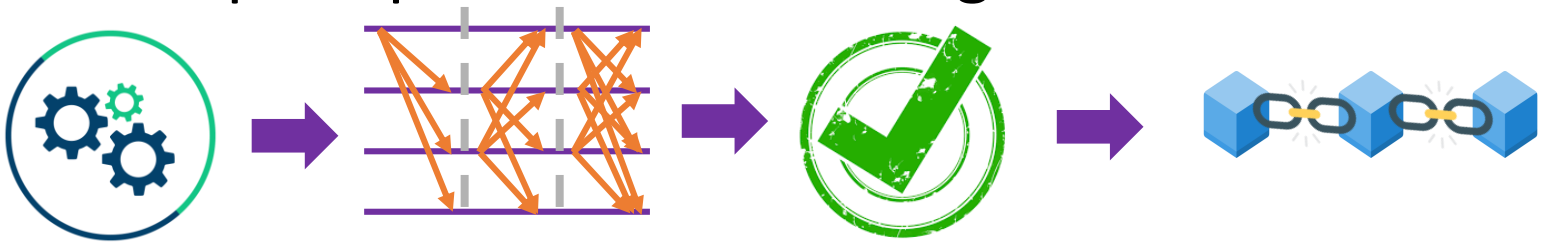
Bitcoin review in a Database Context

- Clients multicasts their requests
- Nodes validate the transactions, put them into the blocks, and try to solve the puzzle
- The lucky node who solves the puzzle first multicasts the block
- Each node validates the transactions within the block
- Transactions are *deterministically* executed by every node and appended to the ledger
- Bitcoin is pessimistic
  - Order First-Then Execute Model



The Optimistic Permissioned Blockchain

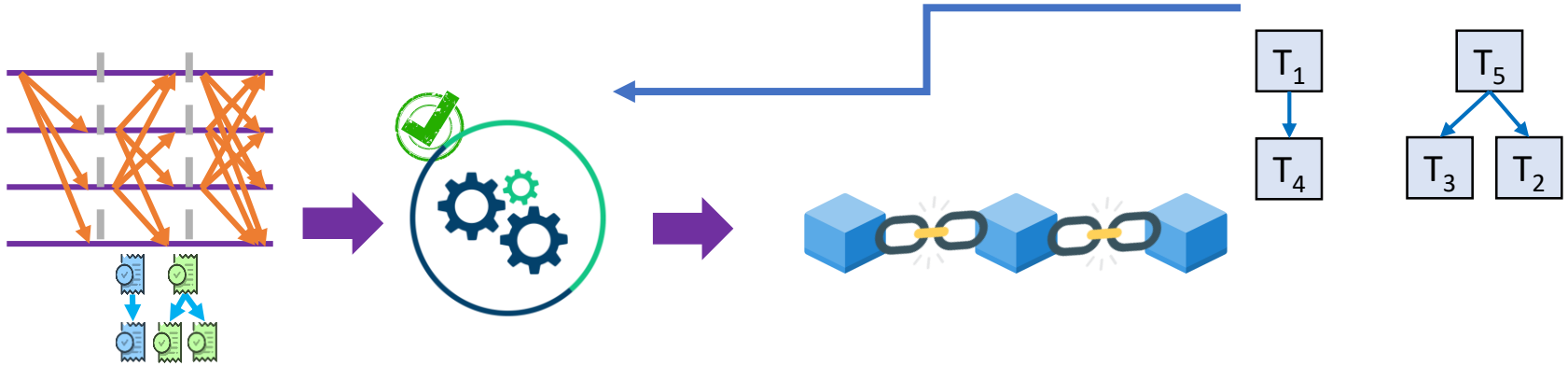
- **Hyperledger Fabric**: Execute-Order-Validate Model.
- Each transaction is first executed by a subset of nodes
- A separate set of nodes order transactions, puts them into blocks, and multicasts them to all the nodes.
- Each node validates the transactions in a block and updates the ledger
- Conflicting Transactions are aborted
- → poor performance for high contention workloads.



Androulaki, E., et al., Hyperledger fabric (2018) a distributed operating system for permissioned blockchains. EuroSys, ACM.

Its OK to be pessimistic if you have parallelism!

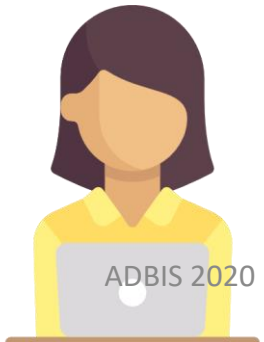
- Order-Parallel Execute: ParBlockchain (ICDCS 2019).
- Orderers order first and generate a dependency graph.
  - Dependency Graph captures conflicts between transactions to give a partial order of transactions.
- Executors execute transactions following the dependency graph and append block to blockchain.
- Non-deterministic execution results in inconsistent execution and detected.



Mohammad Javad Amiri, Divyakant Agrawal, Amr El Abbadi, ParBlockchain: Leveraging Transaction Parallelism in Permissioned Blockchain Systems, The 39th IEEE International Conference on Distributed Computing Systems (ICDCS), 2019.

Back to the Database World: Managing data on untrusted infrastructure

How to guarantee correct transaction execution?



Fault Tolerance vs Fault Detection

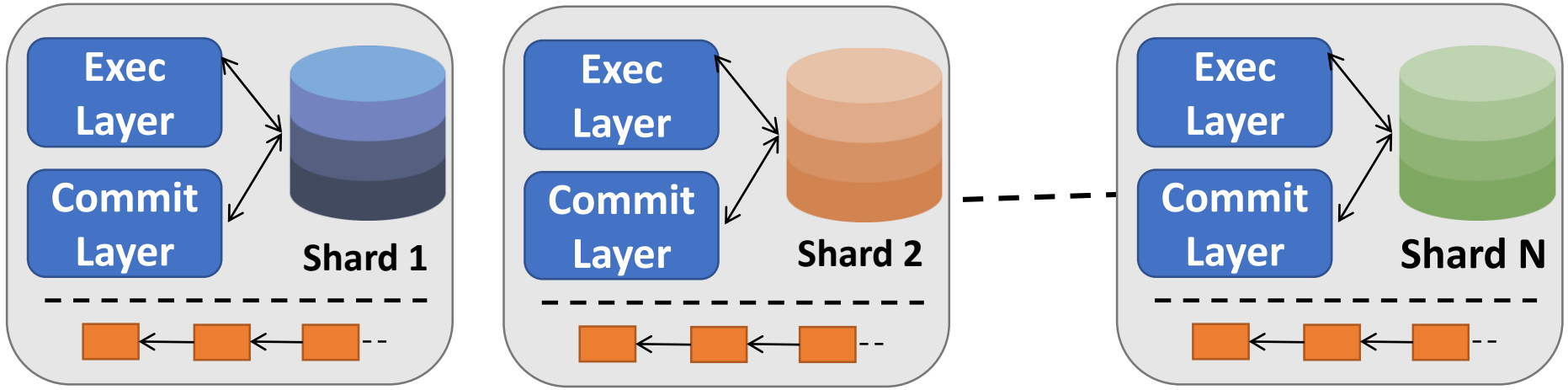
- Fault tolerance: make progress even with one or more faults
- Typically using **replication**
- But replication is **expensive** and tolerates only a fraction of faults
  - 2f+1 to tolerate f crash failures
  - 3f+1 to tolerate f malicious failures

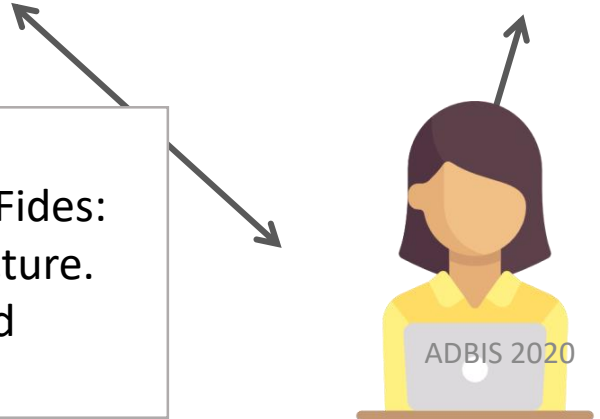
- Alternate option: **Fault Detection**
- Allow failures to occur but always **detect** the faults and **undeniably link** the failures to the faulty non-trustworthy server

- Typically **auditor** audits to detect faults
- Offline action if found guilty
- n > f (not n > 3f), for f faulty processes.

Fides: A full fledged DBMS [ICDCS 2020]

- A full fledged DBMS residing on untrusted infrastructure
  - Uses blockchain-like log for auditing purposes.
  - Uses cryptographic techniques to guarantee ACID properties





Sujaya Maiyya, Danny Hyun Bum Cho, Divyakant Agrawal and Amr El Abbadi, Fides: Managing Data on Untrusted Infrastructure. International Conference on Distributed Computing Systems, 2020.

Parting Thoughts

- Lots to learn from databases in blockchain design.
- Lots to learn from blockchains in databases development
- Blockchains made us conscious of designing for non-trusted infrastructure
- Permissionless blockchains: challenging old problems in a new context
- Permissioned blockchains an opportunity to apply ideas from distributed systems in novel database context.