Lecture 10:
Part 1. Back to Blockchains

Course instructors: Alexey Frolov and Yury Yanovich

Teaching assistant: Stanislav Kruglik

November 13, 2018

User by Role

- **Maintainers** of the blockchain infrastructure, who decide business logic on the blockchain
- External **auditors** of the blockchain operations
- **Clients** who are the end users of the services provided by maintainers.

User by Role: Example

	Financial ledger	Public registry	Supply chain
Maintainer	Bank(s), exchange(s)	Government agency	Goods manufacturer(s) or specialized blockchain provider(s)
Auditors	Internal auditors, regulators, law enforcement	Government-appointed auditors; NGOs	Customers
Clients/Users	Bank clients; securities owners	Citizens (e.g., immovable property owners, copyright owners)	Goods manufacturer(s)

Auditors

store full replica of the entire blockchain data = full read access

passive observers of the consensus algorithm = no “write” access

Blockchains by read access:

- **public blockchains**: information from the blockchain is available to the general audience
- **private blockchains:** consumers of the information are few and are known beforehand.

Maintainers

✓ store full replica of the entire blockchain data = full read access

✓ active participants of the consensus algorithm = “write” access

Blockchains by write access:

- **permission-less blockchains:** anyone has the ability to process transactions on the blockchain (e.g., proof-of-work or proof-of-stake consensus algorithms); the governance on the blockchain is (at least ideally) envisioned as a continuation of the said consensus algorithm
- **permissioned blockchains:** there is an entity or a set of entities that can process transactions and which collectively decide the rules of transaction processing.

Users

✗ can’t get full replica of the entire blockchain data = limited read access

✗ passive observers of the consensus algorithm = no “write” access

✓ *for each blockchain type:* may utilize cryptographic proofs to verify the authenticity of the blockchain data given an access to relatively small portion of data

Use Cases

	Public	Private
Permission-less	Vast majority of cryptocurrencies and public utility blockchains (e.g., Bitcoin, Ethereum)	N/A (does not make sense from the business perspective)
Permissioned	Majority of constituent blockchains in multi-chain systems (Polkadot, TON, Plasma); federated sidechains	“Classic” private blockchains (e.g., built with Hyperledger Fabric)

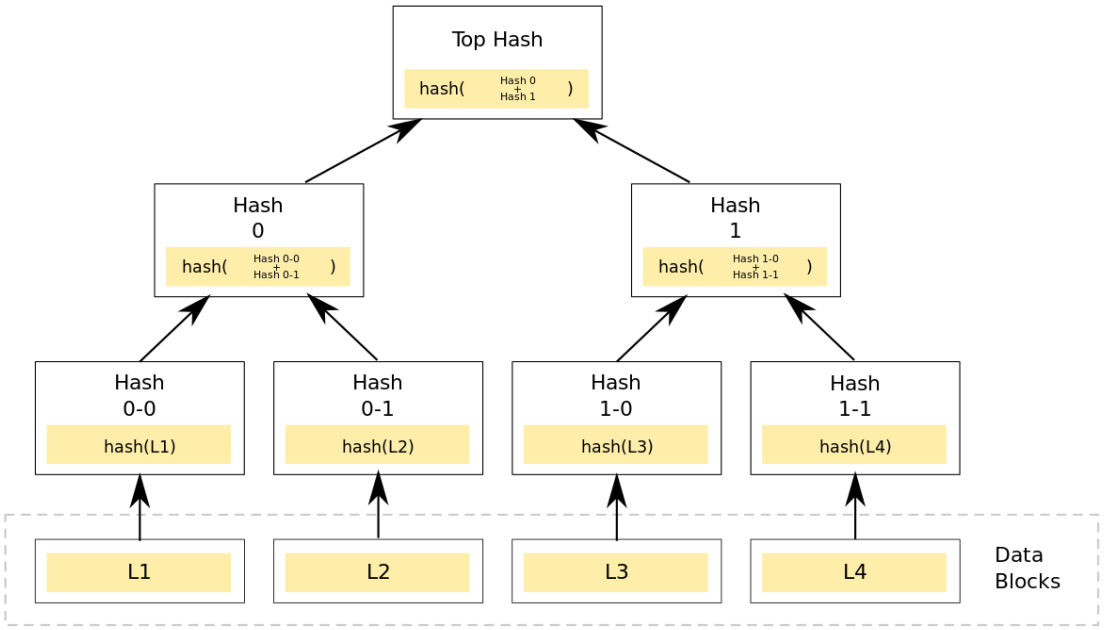
Universal Auditability for Users

is achieved through

- **Linked timestamping** to ensure that the blockchain transaction log cannot be retrospectively edited, even by collusion of the system maintainers in permission-less blockchains
- **Public-key infrastructure** to associated (some of) public keys used to authorize transactions with specific entities, e.g., in order to trust them in some manner
- **Commitment schemes** based on Merkle trees and similar authenticated data structures to enable integrity verification of the data given the limited access to the blockchain.

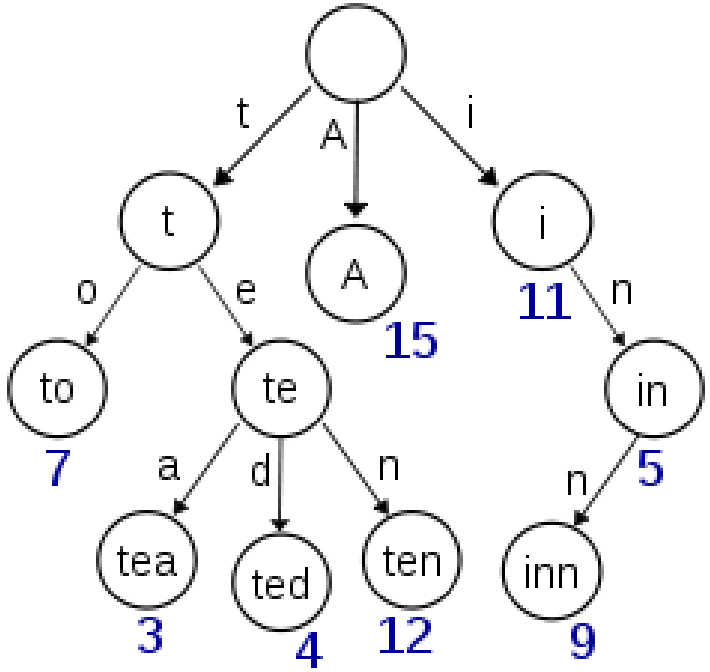
One Merkle Structure
= One Request Type with Proofs

Merkle Tree



Merkle Patricia Tree

~ Prefix Tree



Auditability + Privacy

Zero-knowledge proofs – techniques that allow to proof certain integrity constraints about the data without revealing the data itself

- confidential transaction amounts
- confidential sender/receiver
- confidential smart contract execution

Zero-knowledge proofs

✗ design not flexible

✓ can allow to **combine auditability with data privacy**.

Performance: permission-less

Systems without blockchain

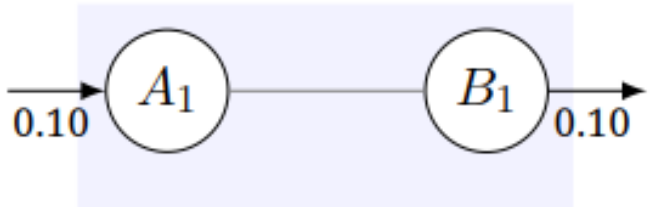
- PayPal: 200 tps on average (transactions per second)
- Visa: 2000 tps on average

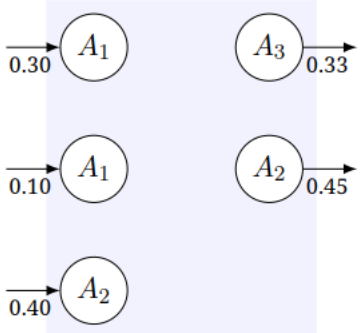
(Permission-less) public blockchain performance

- Bitcoin: 7 tps
- Ethereum: 50 tps -> 14 tps

Transactions are not equivalent!

1-to-1 vs. many-to-many

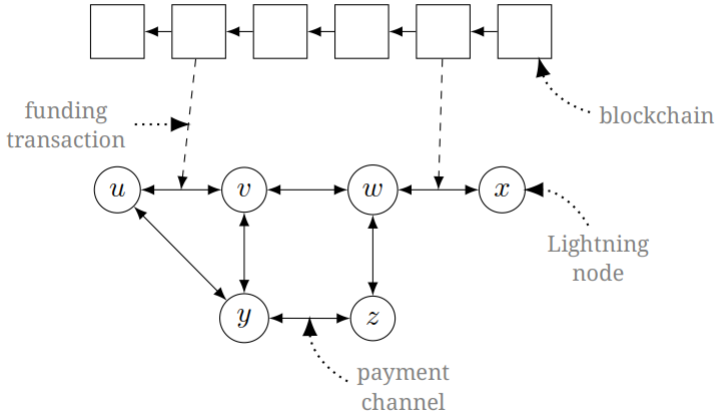


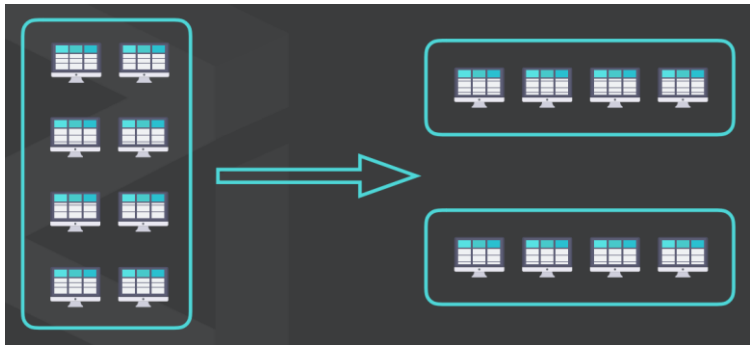


Performance: permission-less (2)

Ways to speed up

- Lightning
- Multichains
- Sharding





Performance: permissioned

Restriction on maintainers set => mathematical Byzantine fault tolerant consensus algorithms instead of economic (Proof-of-X algorithms).

Permissioned public and private blockchain performance

- Hyperledger Fabric: 3 500 tps
- Exonum: 5 000 tps

Cover Visa needs from the box!

To achieve reliability: anchoring.

Permission-less VS Permissioned blockchains

Permission-less blockchain	Permissioned blockchain
Censorship resistance: The blockchain is completely open for participation. Anyone can process transactions; this process is usually incentivized with the help of blockchain currency tokens (e.g., bitcoins)	**Control:** The blockchain is operated by known entities (or a single entity). Incentives to maintain the blockchain may lie outside (i.e., the blockchain may lack built-in currency tokens)
Universal access: All blockchain data is public and can be read by anyone. Anyone can create a transaction and submit it to the blockchain network, too	**Finely grained access:** Access to read blockchain data and create transactions may be regulated by blockchain maintainers
Availability: Transactions are always processed even if the blockchain network is split (e.g., due to a wide-scale Internet blackout). Finality of transactions is not always guaranteed due to possibility of blockchain reorganizations	**Finality:** Transactions are guaranteed to be final as soon as they enter the blockchain. Availability is not always guaranteed (i.e., a blockchain may stop accepting transactions during a network split)