Blockchain-Grundkonzepte

Learning Outcome: Grundkonzepte der Blockchain-Technologie erklären — Bloom's: Understand

Prof. Dr. J. Osterrieder

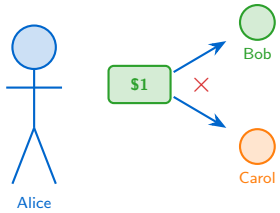
BSc Blockchain, Crypto Economy & NFTs

Spring 2026

Why Can't You Copy a Bitcoin?

The Double-Spend Problem:

- A physical €10 note can only be in one wallet at a time
- A digital file can be copied infinitely at zero cost
- Sending money by email would let you keep *and* spend it
- Traditional solution: trust a central bank to track balances
- Bitcoin's innovation: a shared ledger no single party controls



Core Challenge

How do untrusted strangers prevent the same digital coin from being spent twice – without a central authority?

Blockchain solves double-spending by making all transactions visible and irreversible to everyone simultaneously.

What Makes a Block a Block?

Definition

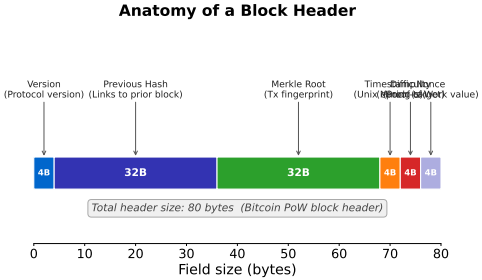
A **block** is an ordered set of validated transactions bundled with a **cryptographic hash** of the previous block, chaining all history together into an immutable sequence.

Block Header contains:

- **Previous hash** – links to the prior block
- **Merkle root** – fingerprint of all transactions
- **Nonce** – number miners vary to solve PoW
- **Timestamp** – records block creation time

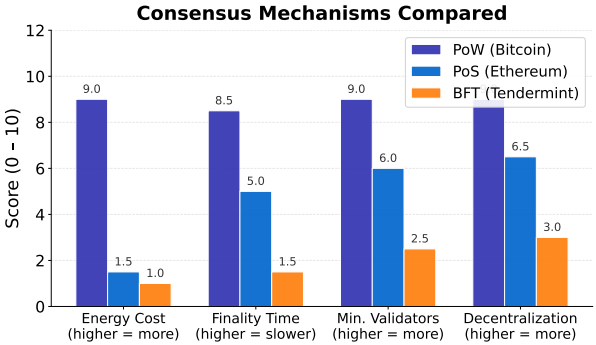
Block Body contains:

- Ordered list of validated transactions
- Coinbase transaction (miner reward) as first entry



Changing any transaction in block n invalidates the hash of block n, which then invalidates block $n+1$, $n+2$, ... – rewriting history requires redoing all subsequent work.

How Do Strangers Agree?



Three Approaches:

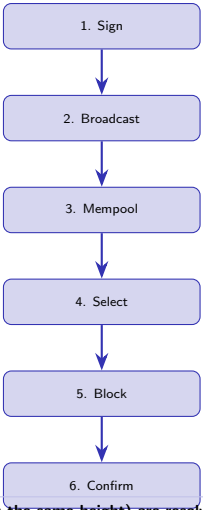
- **PoW:** Miners solve energy-intensive puzzles – costly to attack, open participation (Bitcoin: $\sim$7 TPS)
- **PoS:** Validators lock capital as collateral – energy-efficient, security scales with stake (Ethereum: $\sim$30 TPS)
- **BFT:** Deterministic finality in milliseconds – fast, but requires known validator sets ($<$100 nodes)

Every consensus mechanism trades security, scalability, and decentralization – there is no free lunch (the Blockchain Trilemma).

From Transaction to Confirmation

The 6-Step Confirmation Path:

1. **Sign:** Owner uses private key to authorise the transfer
2. **Broadcast:** Signed transaction propagated peer-to-peer
3. **Mempool:** Unconfirmed transactions queue in memory pool
4. **Select:** Miner/validator picks transactions (usually by fee)
5. **Block:** Chosen transactions packed, PoW/PoS solved
6. **Confirm:** Network accepts block; each subsequent block adds one more confirmation (6 = industry standard for Bitcoin)



A transaction is only truly final when buried under multiple blocks – orphaned blocks (two valid blocks at the same height) are resolved by the longest-chain rule.

Hashing by Hand

SHA-256 Worked Example (Avalanche Effect):

Input A: `Hello`

Output A:

$\underbrace{\texttt{2cf24dba5fb0a30e}}_{\text{bits 1–64}}\ \underbrace{\texttt{26c6da02bee94}}_{\text{bits 65–116}}\ \underbrace{\texttt{b94d27b9}}_{\text{last 32 bits}}\ \ldots$

Input B: `Hello!` *(one character added)*

Output B:

$\underbrace{\texttt{334d016f755cd6dc}}_{\text{bits 1–64}}\ \underbrace{\texttt{58c1aa3b}}_{\text{next 32 bits}}\ \ldots$

Avalanche Effect

A single-character change flips ∼50% of output bits. Outputs are **unpredictable**, **fixed-length** (256 bits), and **one-way** – you cannot reverse a hash to find the input.

Three Properties That Matter:

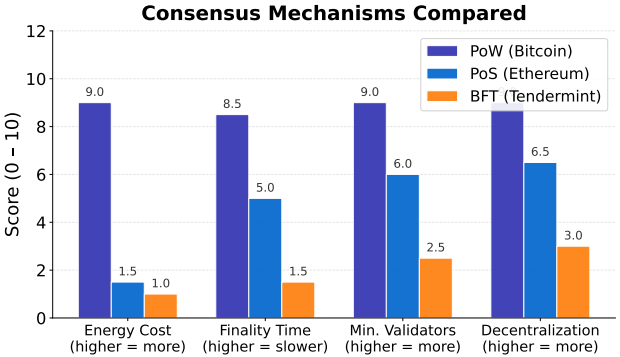
- **Deterministic:** same input always gives same output
- **Pre-image resistant:** given the hash, finding the input is computationally infeasible
- **Collision resistant:** finding two inputs with the same hash is computationally infeasible

Why Blockchains Need This:

- Block headers store the *hash* of the previous block
- Tampering with any block changes its hash
- Propagates forward – every subsequent block is invalidated

SHA-256 produces a 256-bit (32-byte) fingerprint. Bitcoin uses it twice (SHA-256d) for extra security. Ethereum uses Keccak-256.

Where Are Blockchains Today?



- **Bitcoin:** Processes ∼7 transactions per second (TPS) – by design, optimised for security and decentralisation over throughput; block time ≈10 min
- **Ethereum:** ∼30 TPS on base layer post-Merge (Sept 2022); Layer-2 rollups push effective throughput to thousands of TPS while inheriting L1 security
- **Market context:** Combined crypto market cap exceeded $2 trillion in 2025; Bitcoin ETF approval (Jan 2024) opened institutional access; MiCA (EU, Dec 2024) provides the first comprehensive regulatory framework

Throughput figures are base-layer only. With Layer-2 solutions (Lightning, Optimism, Arbitrum), effective capacity increases by 10x to 100x.

What Breaks a Blockchain?

Attack Vectors:

- **51% Attack:** Attacker controls majority of hash power (PoW) or staked value (PoS) – can rewrite recent history and double-spend
- **Selfish Mining:** Miner withholds discovered blocks to gain disproportionate rewards, undermining fairness without 51%
- **Nothing-at-Stake:** Early PoS flaw – validators can vote on multiple forks simultaneously at zero cost, preventing consensus
- **Eclipse Attack:** Isolating a node from honest peers and feeding it a false view of the chain

Consequences:

- **Double Spending:** Spend the same coin on two branches, then suppress one – seen on smaller PoW chains (ETC, BTG)
- **Chain Reorganisation:** Honest chain is replaced by a longer attacker chain – transactions believed confirmed are reversed
- **Loss of Finality:** Probabilistic finality (PoW) means no transaction is 100% final – only economically impractical to reverse
- **Network Partition:** Long latency or censorship splits the network into temporary forks

Bitcoin has never suffered a successful 51% attack – the cost exceeds $10 billion in hardware and energy. Smaller chains are far more vulnerable.

Five Questions to Test Any Blockchain

Evaluation Questions:

1. **Decentralised?** How many independent nodes validate? Who can become a validator?
2. **Immutable?** Has the chain ever been reorganised? How deep were confirmed blocks rolled back?
3. **Censorship-resistant?** Can a validator refuse specific transactions? What happened when they tried?
4. **Scalable?** What is the base-layer TPS? What Layer-2 solutions exist?
5. **Energy-efficient?** What is the energy cost per transaction compared to Visa or a bank transfer?

How to Evaluate Each:

- Count full nodes: *bitnodes.io* for Bitcoin ($>$18k nodes)
- Check fork history: block explorers show all reorganisations
- Test censorship: OFAC-compliant blocks filtered $<$30% of Ethereum blocks post-Merge before falling (market incentives prevailed)
- Measure TPS: public metrics at *txstreet.com* or *etherscan.io*
- Per-tx energy: Cambridge CBECI (Bitcoin $\approx$700 kWh/tx vs. Visa $<$0.001 kWh/tx – but Bitcoin settles final value, not just data)

Apply these five questions to every blockchain claim you encounter – they cut through hype and reveal real trade-offs.

Key Takeaways

What We Covered:

1. **Double-spend problem:** the fundamental challenge that requires a trust mechanism in digital money
2. **Block structure:** header (prev hash, Merkle root, nonce, timestamp) + body (ordered transactions)
3. **Consensus:** PoW trades energy for open access; PoS trades capital lockup for efficiency; BFT trades speed for small validator sets
4. **Hashing:** SHA-256 avalanche effect makes tampering computationally visible – the backbone of block linkage
5. **Attack landscape:** 51% attacks, selfish mining, and eclipse attacks define the adversarial threat model

You Can Now Explain:

- What a block is and how blocks chain together
- Why consensus is hard between untrusted strangers
- How cryptographic hashing makes history immutable
- Why blockchains have throughput limits by design
- What can go wrong and how attacks work in practice

Bloom's: Understand

You can **classify**, **describe**, **explain**, **summarise**, and **compare** blockchain concepts – the foundation for all higher-level analysis in this course.