

Lesson 1: What is Blockchain?

Module A: Blockchain Foundations

MSc Blockchain & Cryptocurrency

Digital Finance Program

2025

By the end of this lesson, you will be able to:

1. Define blockchain as a cryptographically-secured distributed ledger
2. Trace the historical evolution from 1991 to 2025
3. Explain the double-spending problem and its consensus-based solution
4. Formalize the hash chain structure: $H(B_n) = H(\text{header}_n || \text{prev_hash}_{n-1})$
5. Compare centralized, decentralized, and distributed architectures

Prerequisites: Cryptographic hash functions, basic probability theory

MSc level: Full mathematical rigor expected in subsequent slides

Formal Definition of Blockchain

Mathematical Definition

A blockchain \mathcal{B} is an ordered sequence of blocks:

$$\mathcal{B} = (B_0, B_1, \dots, B_n)$$

Where each block B_i contains:

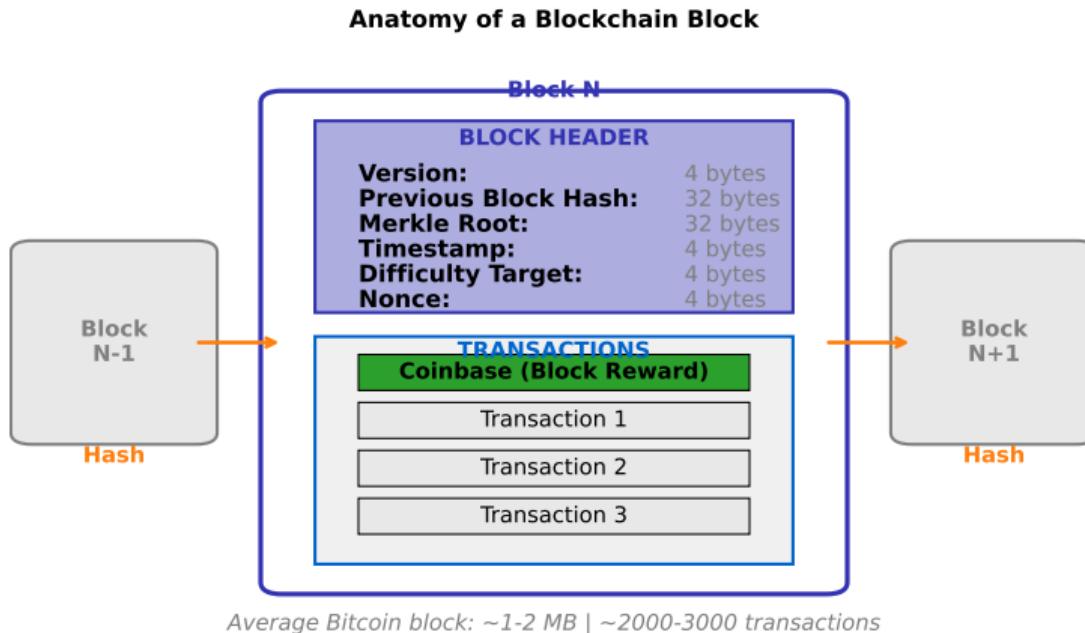
- Header h_i with metadata (version, timestamp, nonce, difficulty)
- Transaction set $T_i = \{tx_1, \dots, tx_k\}$
- Hash pointer to previous block: $H(B_{i-1})$

Integrity Constraint:

$$\forall i > 0 : B_i.\text{prev} = H(B_{i-1})$$

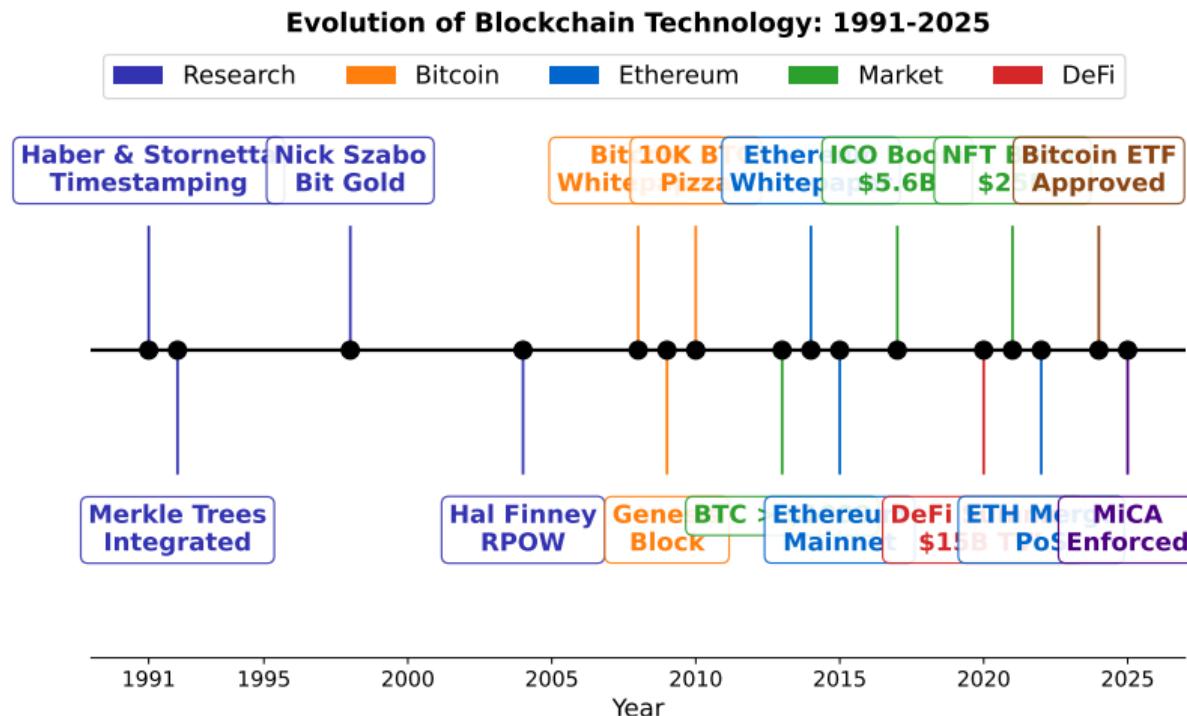
The hash pointer creates a tamper-evident linked data structure

Anatomy of a Blockchain Block



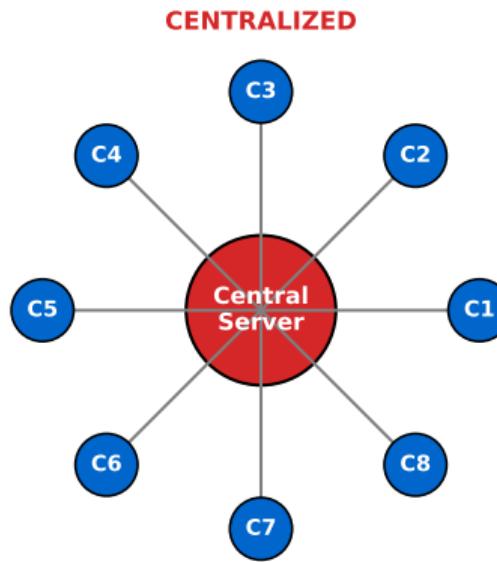
Bitcoin block header: 80 bytes — Block body: 1-2 MB of transactions

Historical Evolution: 1991–2025



Key inflection points: 2008 (Nakamoto), 2015 (Ethereum), 2024 (Bitcoin ETFs)

Centralized Architecture

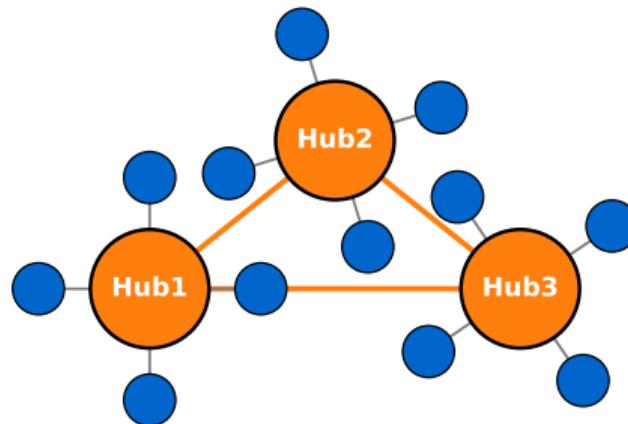


Single point of failure | High throughput ($\sim 10^6$ TPS) | <10ms latency

Characteristics: Single authority, 10^6 TPS, <10ms latency

Traditional systems: banks, exchanges, cloud services

DECENTRALIZED



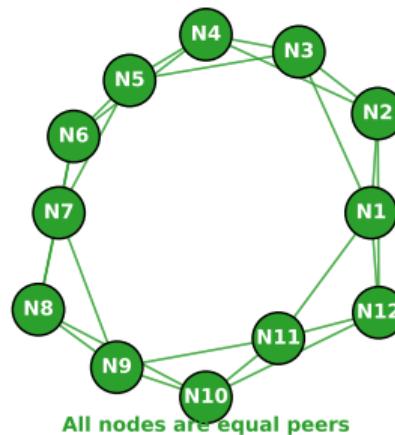
Multiple hubs | Federation trust | $\sim 10^3$ TPS | $\sim 1s$ latency

Characteristics: Multiple hubs, federated trust, 10^3 TPS

Examples: Federated exchanges, consortium blockchains

Distributed Architecture (Blockchain)

DISTRIBUTED (Blockchain)



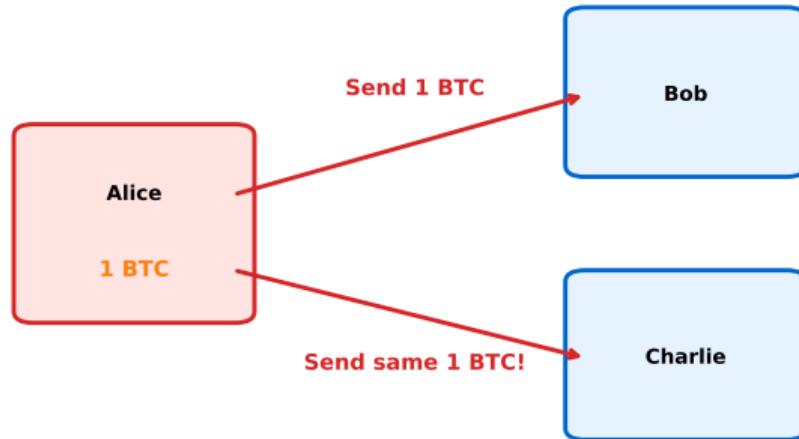
No single point of failure | Cryptographic trust | ~10 TPS | ~10 min finality

Characteristics: No hierarchy, cryptographic trust, 10^1 TPS

Blockchain trades performance for trustlessness

The Double-Spending Problem

THE PROBLEM: Digital files can be copied

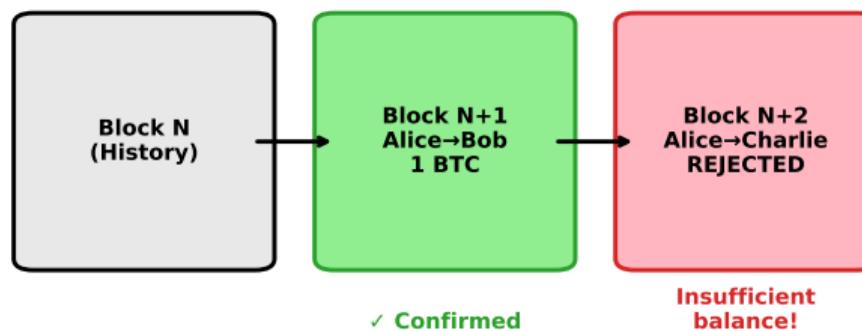


Problem: Prevent $\text{Transfer}(a, A \rightarrow B) \wedge \text{Transfer}(a, A \rightarrow C)$

Digital files can be copied infinitely — no inherent scarcity in bits

THE SOLUTION: Blockchain Ordering

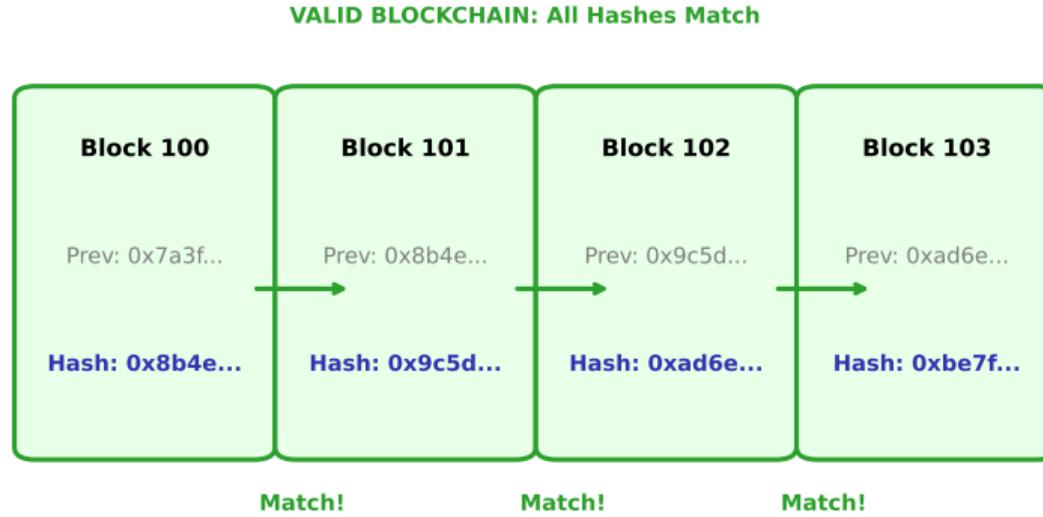
First valid transaction wins — Network consensus determines order



Solution: Distributed consensus determines transaction ordering

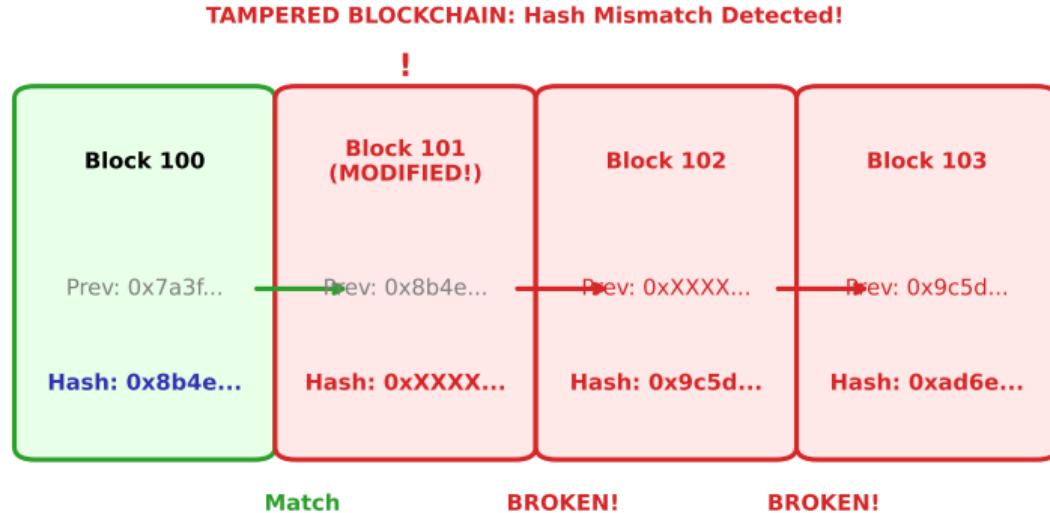
Nakamoto's key insight: Use computational work to achieve probabilistic finality

Valid Blockchain: Hash Chain Integrity



Hash pointers create a tamper-evident linked data structure

Tampered Blockchain: Detection



Modifying B_k requires recomputing all subsequent hashes: $O(n - k) \times 2^{76}$ ops

Merkle Tree: Efficient Transaction Verification

Structure

Binary hash tree over transactions:

$$\text{Root} = H(H(H(tx_1) \parallel H(tx_2)) \parallel H(H(tx_3) \parallel H(tx_4)))$$

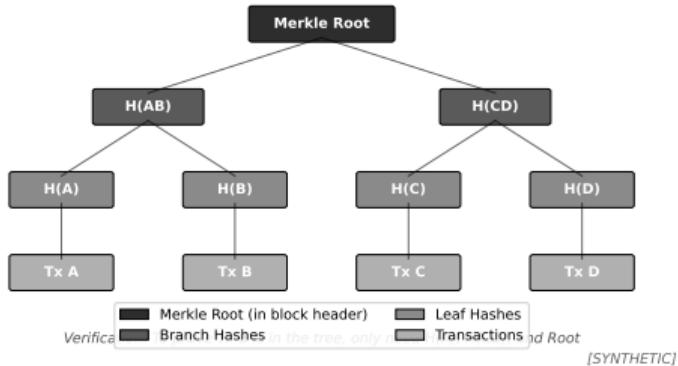
Verification Complexity:

- Full verification: $O(n)$ hashes
- Merkle proof: $O(\log n)$ hashes
- SPV clients use proofs, not full chain

Bitcoin Block Header:

32-byte Merkle root commits to all transactions

Merkle Tree: Efficient Transaction Verification



Merkle trees enable lightweight clients: verify transactions without downloading full blocks

Cryptographic Properties

- **Collision Resistance:**

$$\Pr[H(x) = H(y) \wedge x \neq y] \approx 2^{-128}$$

- **Preimage Resistance:**

Given h , infeasible to find $x : H(x) = h$

- **Avalanche Effect:**

1-bit change \Rightarrow 50% output bits flip

Probability of Reversal (after k confirmations, attacker with $q < 0.5$ hashrate):

$$P(\text{reversal}) < \left(\frac{q}{1-q} \right)^k$$

6 confirmations \Rightarrow reversal probability $< 0.1\%$ for $q = 0.3$

System Properties

- **Liveness:**

Valid transactions eventually confirmed

- **Safety:**

No double-spends with $> k$ confirmations

- **Consistency:**

All honest nodes agree on prefix

Finance & Payments

- Bitcoin: \$1.2T market cap, \$50B daily volume
- Stablecoins: USDT/USDC \$150B+ circulation
- DeFi TVL: \$80B across protocols
- Bitcoin ETFs: \$50B+ AUM (Jan 2024 launch)

Enterprise

- IBM Food Trust: 500+ organizations
- JPMorgan Onyx: \$1B+ daily settlements
- Maersk TradeLens: 1.5B shipping events

Government & CBDC

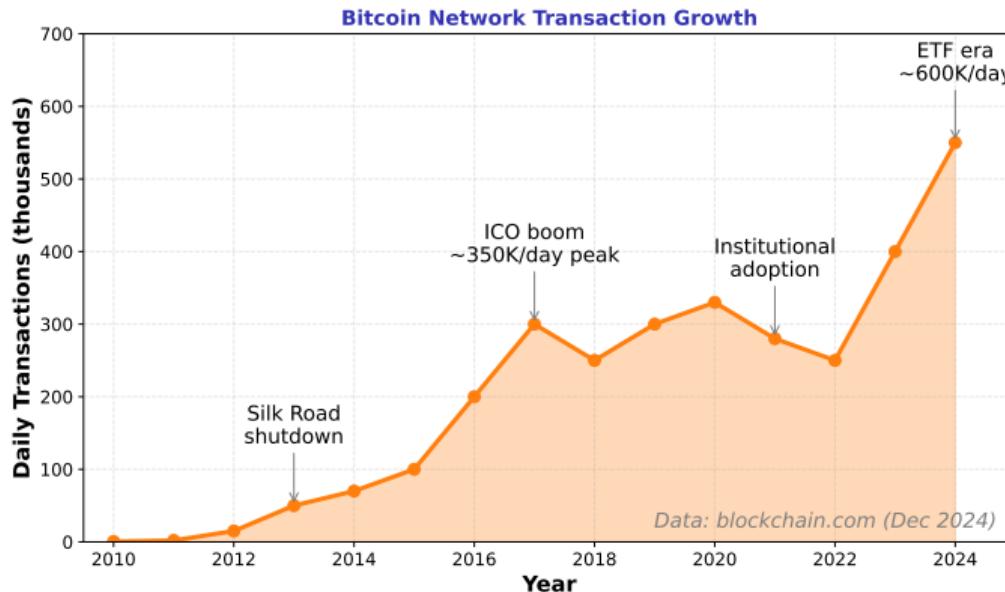
- China e-CNY: 260M+ wallets
- EU Digital Euro: Pilot phase 2024
- 130+ countries exploring CBDCs

Emerging Applications

- Real-World Assets (RWA): \$5B+ tokenized
- Decentralized Identity (DID)
- Supply chain provenance
- Carbon credit verification

Source: DeFi Llama, CoinGecko, Atlantic Council CBDC Tracker (Dec 2024)

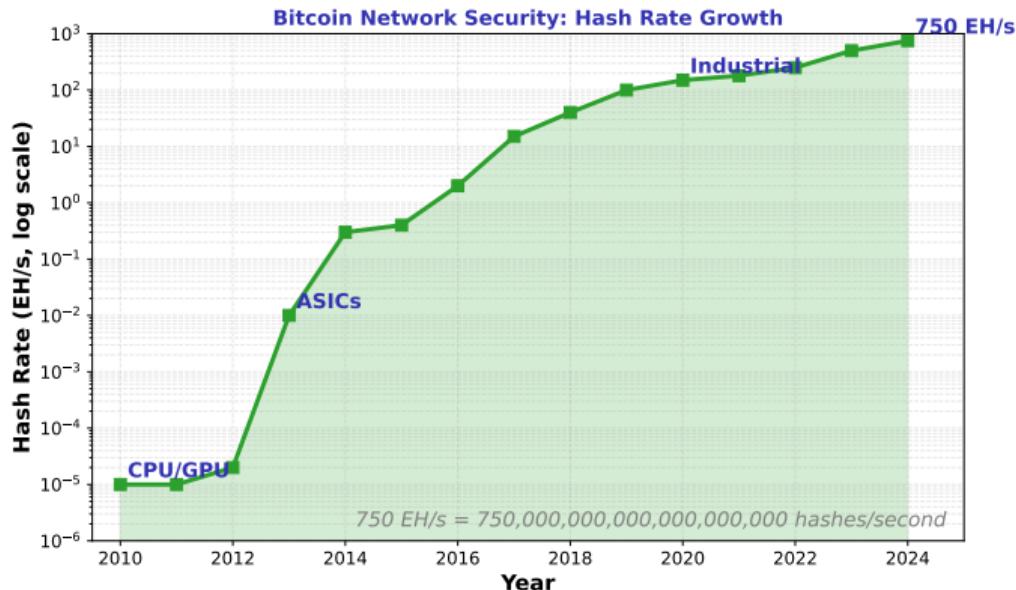
Network Adoption: Bitcoin Transactions



From <1K daily transactions (2010) to >600K daily (2024)

Transaction volume indicates real economic activity on the network

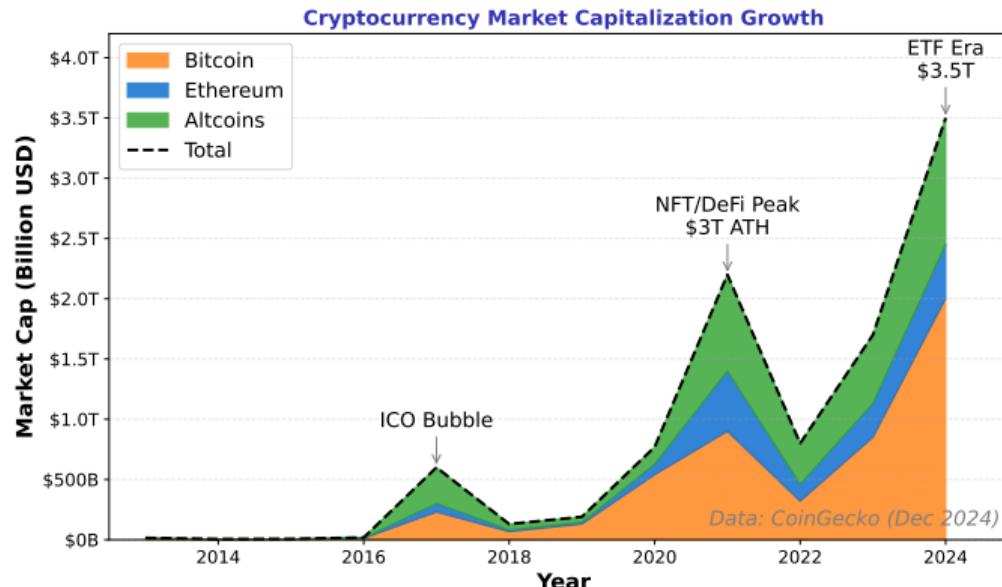
Network Security: Hash Rate Growth



$750 \text{ EH/s} = 7.5 \times 10^{20} \text{ SHA-256 hashes per second}$

Higher hash rate = more computational cost to attack the network

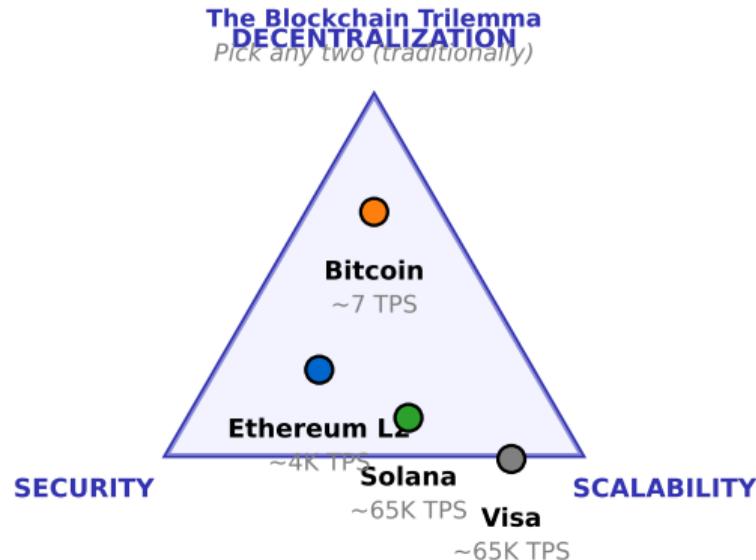
Market Growth: Cryptocurrency Capitalization



Total market cap: \$3.5T (Dec 2024); Bitcoin dominance: ~57%

Market cap growth reflects institutional adoption and mainstream acceptance

The Blockchain Trilemma



Layer 2 solutions attempt to optimize all three dimensions

Decision Framework: When to Use Blockchain

Use Blockchain When:

- ✓ Multiple writers, no trusted party
- ✓ Immutable audit trail required
- ✓ Disintermediation creates value
- ✓ Censorship resistance needed
- ✓ Cross-organizational data sharing

Use Traditional DB When:

- ✗ Single organization controls data
- ✗ High throughput required (>10K TPS)
- ✗ Data deletion/modification needed
- ✗ Strong privacy requirements
- ✗ Existing solutions work well

Decision Heuristic:

$$\text{Blockchain Value} \propto \frac{\text{Trust Deficit} \times \text{Coordination Benefit}}{\text{Performance Requirements}}$$

Most enterprise “blockchain” projects could use a replicated database

Core Concepts:

1. Blockchain = hash-chained blocks + distributed consensus + cryptographic signatures
2. Double-spending solved via total ordering through consensus mechanism
3. Immutability achieved through computational intractability of hash chain modification
4. Trade-off: Performance \leftrightarrow Trustlessness \leftrightarrow Decentralization

Mathematical Foundations:

- Hash functions: $H : \{0,1\}^* \rightarrow \{0,1\}^{256}$ (SHA-256)
- Merkle trees: $O(\log n)$ verification complexity
- Reversal probability: Exponential decay with confirmations

Next Lesson: L02 – Distributed Ledger Technology (DLT) deep dive

Blockchain is a tool, not a solution — evaluate against specific requirements