Consensus Mechanisms

A Five-Minute Overview

BSc Blockchain Course

Why Can't Distributed Computers Just... Agree?

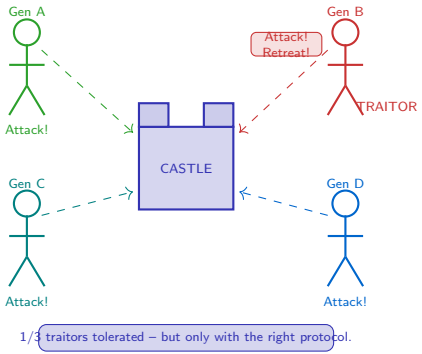
Imagine four generals surrounding a castle. They must all attack at the same time or all retreat – a split decision means defeat. They can only communicate by messenger, and one general is secretly a traitor sending contradictory orders.

This is exactly the problem blockchains solve:

- **No central coordinator** – every node is a general
- **Unreliable channels** – messages can be delayed or faked
- **Byzantine faults** – some nodes lie deliberately

The FLP impossibility theorem (1985) proves that no deterministic protocol can guarantee consensus in an asynchronous network if even one node can fail. Blockchains escape by relaxing one assumption: they allow probabilistic finality.

Consensus is the core problem. Every design choice in PoW and PoS is an answer to it.



The Byzantine Generals Problem, formalised by Lamport, Shostak and Pease (1982), proves that honest agreement requires at least $\frac{2}{3}$ of participants to be honest. Blockchain consensus mechanisms are practical engineering solutions to this theoretical lower bound.

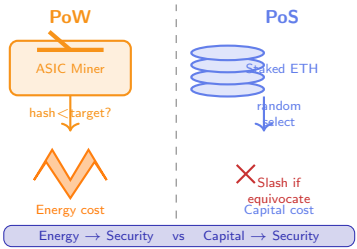
Source: Lamport, Shostak & Pease (1982). "The Byzantine Generals Problem." ACM TOPLAS 4(3). Fischer, Lynch & Paterson (1985). "Impossibility of Distributed Consensus." JACM 32(2).

PoW vs PoS: Two Philosophies of Trust

Two answers to the same question: what makes a block proposal credible?

Property	Proof of Work	Proof of Stake
Trust source	Burned energy	Locked capital
Block proposer	First valid hash	Weighted random draw
Sybil resistance	CPU/ASIC cost	Stake cost
Attack cost	51% of hashrate	33%–67% of stake
Energy use	Very high	**<0.01% of PoW**
Finality type	Probabilistic	Economic / absolute
Hardware required	Specialised ASICs	Any validator node
Longest chain rule	Yes (heaviest chain)	Replaced by fork choice

The core



tradeoff:
PoW converts real-world cost (electricity) into on-chain security. An attacker must acquire and operate more than half the network's mining hardware – a tangible, visible, ongoing expense. PoS converts financial commitment into security. An attacker must acquire a large fraction of the circulating supply – expensive but invisible, and recoverable if the attack fails (unlike burned electricity).

Neither mechanism is universally superior: PoW provides external-cost security and proven track record; PoS provides energy efficiency and programmable slashing conditions. The choice is a design decision about which resource to weaponise for Sybil resistance.

Source: Nakamoto, S. (2008). "Bitcoin: A Peer-to-Peer Electronic Cash System." Buterin, V. (2022). "Proof of Stake." Ethereum Foundation. Ethereum.org/en/developers/docs/consensus-mechanisms

How Validators Reach Agreement: The Casper FFG Cycle

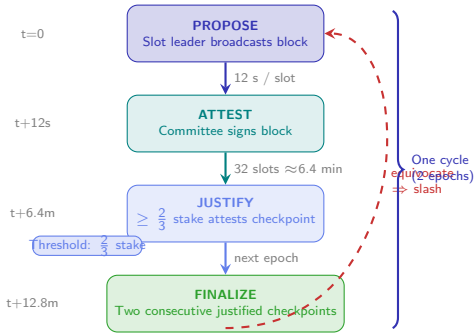
Four phases from proposal to finality: **1. Propose** – A validator is pseudo-randomly selected for the current slot (12 seconds). It assembles a block from the mempool and broadcasts it to peers.

2. Attest – The remaining validators in the committee review the proposed block and broadcast a signed attestation: "I saw this block and it looks valid."

3. Justify – After one epoch (32 slots, ≈6.4 minutes), if $\frac{2}{3}$ of total staked ETH has attested to a checkpoint, that checkpoint is *justified*: it is the candidate for finality.

4. Finalize – When two consecutive checkpoints are both justified, the first becomes *finalized*: it can never be reverted without slashing at least $\frac{1}{3}$ of all staked ETH – an economically catastrophic act.

Under normal network conditions, finality is reached roughly every 12.8 minutes (two epochs).



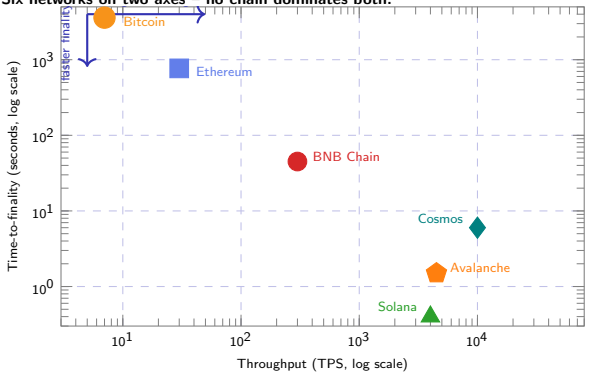
Casper FFG (Friendly Finality Gadget) overlays economic finality onto the longest-chain fork rule. The $\frac{2}{3}$ threshold is the BFT safety bound: as long as fewer than $\frac{1}{3}$ of validators are dishonest, finalized blocks are safe.

Source: Buterin, V. & Griffith, V. (2019). "Casper the Friendly Finality Gadget." arXiv:1710.09437.

Ethereum.org/en/developers/docs/consensus-mechanisms/pos/gasper

The Consensus Landscape: Finality vs. Throughput

Six networks on two axes – no chain dominates both:



Approximate illustrative values. TPS under ideal conditions.

Reading the chart:

- **Bitcoin** – upper-left: 7 TPS, 60-minute probabilistic finality. Security maximised, throughput minimal.
- **Ethereum PoS** – middle: 30 TPS, 12.8-minute economic finality. Balanced but congested by layer-1 demand.
- **Solana** – lower-right: 4,000+ TPS, 0.4-second finality. Speed via a single leader tower; more centralised.
- **Cosmos** – right: IBC hubs, 10,000 TPS, 6-second Tendermint BFT finality. Validator set <175 nodes.
- **Avalanche** – right: Snowflake sampling, 1.5-second finality, novel probabilistic BFT approach.

No chain occupies the lower-right and also matches Bitcoin's decentralisation. This is the trilemma – next slide.

Every network on this chart makes a different bet: Bitcoin bets on maximum security at the cost of speed; Solana bets on speed at the cost of validator concentration. The consensus mechanism is the engineering embodiment of that bet.

Source: Illustrative values. Nakamoto (2008); Ethereum Foundation (2023); Solana Labs (2023); Cosmos Network (2023); Avalanche (2023).

Choosing a Consensus Mechanism: A Decision Framework

Three requirements determine your choice:

1. Security baseline – who are your adversaries?
Nation-state or well-funded attacker ⇒ PoW or high-stake PoS with large validator set. Physical hashrate or large capital stakes are hard to accumulate secretly.
Known, partially-trusted participants ⇒ permissioned BFT (PBFT, HotStuff, Tendermint) for faster, deterministic finality with a known validator set.

2. Finality and UX – what latency can your application tolerate?
Payment / DeFi ⇒ economic finality in seconds, not 60 minutes. PoS with BFT gadget or Tendermint preferred.
Store of value / settlement ⇒ probabilistic PoW finality acceptable; deeper confirmation equals more security.

3. Decentralisation vs. speed – how many validators can you run?
Global, permissionless ⇒ low hardware barriers needed; PoS or delegated staking enables broad participation.
Enterprise / consortium ⇒ PoA or PBFT with a small known validator set gives high TPS at the cost of permissionlessness.

Decision summary:

Priority	Mechanism	Network
Max security	PoW	Bitcoin
Fast & open	PoS + BFT	Ethereum
High TPS	DPoS / Tendermint	Solana, Cosmos
Enterprise	PoA / PBFT	Hyperledger

There is no universally optimal mechanism – only the one that best fits your threat model, validator set size, latency budget, and trust assumptions. Choosing well is the first architectural decision of any blockchain system.

Source: Buterin, V. (2017). "The Meaning of Decentralization." medium.com; Auer et al. (2022). "Blockchain Scalability and the Fragmentation of DeFi." BIS Working Paper 1065.