The Coordination Gap

Josh · Secure Sovereign

Published November 21, 2025 · Updated September 20, 2026

Bitcoin solved block consensus. Node software is still written by informal agreement.

Contents

Bitcoin solved agreement among strangers about which chain is valid. A bad block dies at every node that checks it. The people who write the software that performs that check still coordinate in mailing lists, chat, and a GitHub merge queue. That gap is the subject of this article. It is not a claim that Bitcoin's money rules failed, and it is not a pitch to leave Bitcoin for a new coin. Same chain, same 21 million, same proof of work.

In March 2014 Gavin Andresen stood at Princeton and named both layers. Technical consensus was hard, and Satoshi had shipped it. Social consensus, the human work of deciding how the software that enforces those rules gets written, was also hard, and nobody had solved it. Twelve days later he stepped down as lead maintainer. The talk is *Who Controls Bitcoin*'s search frame. The durable outcome of the fight that followed is *Governance Paralysis Was The Victory*. Why informal process reconstructs a hierarchy even when the occupants change is *The Vertical Layer Problem*.

Two kinds of consensus

Protocol consensus is the set of validation rules every node enforces: subsidy, proof of work, script, block structure. No maintainer can change those rules by merging a pull request. The network either takes a new rule or it does not. Shipping a release is not rewriting those rules. The people who write the default still govern what new operators download.

Social coordination is who may merge, who may stall, and how a fight about the software is settled. Consensus-adjacent work sits between those layers. It does not rewrite the money rules, and it still requires coordinated adoption across developers, miners, and economic nodes. A

security patch that operators must take is in that middle.

A wallet fight that ships in the default node is in that middle because the money rules and the extras share a binary. Treating that fight as a change to the 21 million cap is the miss.

Core is one implementation. Bitcoin is the protocol. A second client that validates the same rules, talks to the same peers, and accepts the same blocks is still Bitcoin. Age does not decide that. Compatibility does. The longer write-up of why the social layer can be taken over without rewriting consensus is *The Social Layer Is the Attack Surface*.

The wars that informal process could not settle

Gavin had already named the 1 MB block limit as a consensus change he knew would be hard. The blocksize war consumed the years after he left. SegWit activated. Bitcoin Cash left, taking a ticker split with it. Mike Hearn quit and called Bitcoin failed. Exchanges halted deposits while the fight was live. The small-block position protected something real: full nodes stay cheap enough to run. The cost was that the war used the window when a second independently specified client was still tractable, and the process for settling the next fight did not get built. That is *Governance Paralysis Was The Victory*.

Taproot did ship. It is the usual reply to the claim that this ecosystem cannot activate anything. Taproot was not the lead maintainer disappearing, and it was not a fight on the scale of the blocksize war. Multiple activation methods, Speedy Trial among them, still had to be socially coordinated under a much larger market cap. That it activated says a smaller disagreement can still clear. It does not say informal merge authority can design a succession process after the people in charge are gone.

The later fight was already on the chain. Ordinals used Taproot's unused envelope plus SegWit's witness discount as a file store. Ready proposals to close dedicated non-money channels stalled. Bitcoin Core v30 removed the default relay limit on `OP_RETURN`. Operators switched to Knots in volume for different policy defaults, and they were still running Core's lineage. Demand showed up. A second independently specified client did not. That record is *The Adversarial Default* and *Making Core Irrelevant*.

Each of those rounds raised the cost of the next one while the coordination process stayed informal.

Why the same structure reconstructs

About five people hold merge access on bitcoin/bitcoin. Release trust still runs through identifiable keys. Review is real. Merge authority is not spread the way review is. The numbers are in *Who Controls Bitcoin* and Bitcoin Governance Research.

Formalizing that process from inside the same seats is the thing the seats were not built to do. The current arrangement is fast for the people who already merge. Users treat the default client as the network. There is no published rule for who may merge, and no venue whose decision both sides accept short of a chain-fork threat. Path dependence is not a moral verdict on today's maintainers. It is why swapping the cohort rebuilds the same hierarchy. *The Vertical Layer Problem* is that shape diagnosis.

A single implementation lineage makes the software layer easier to pressure, because there is one menu most economic nodes download. Forking the chain to escape that menu splits the money. Forking governance against a written specification, if one existed, would not. That distinction is the rest of this catalog.

One public attempt

Bitcoin Commons is a from-scratch Rust client aimed at the same chain and the same consensus rules, built from the Orange Paper, a human-readable mathematical specification, not a new coin and not a ticker split. The case for a specification that humans can read is *Why Bitcoin Needs a Specification*.

Full-chain differential replay against Bitcoin Core is the scale test. `blvm-bench` targets on the order of 900,000 blocks. Those runs are operator-driven and resource-heavy. That is not a claim every CI job has zero-divergence proof to tip. Methodology lives in the differential testing guide. Cryptographic merge authorization is designed and stays off until security review, key management, and community validation. The Specification Lock checks annotated implementation paths against the spec. It does not replace a written specification, and it is not governance enforcement.

Cheap exit on the same chain is what makes informal merge authority fail as a trap. A client that can be checked against a spec can be forked without reverse-engineering Core. A governance ruleset that can be exported without splitting the ledger is a different operation from Bitcoin Cash. None of that is a claim that the money already runs on those tools.

The diagnosis of the monopoly and the no-spec moat is *Who Controls Bitcoin*. What has to happen, in what order, is *Making Core Irrelevant*.

Sources

- Gavin Andresen, "Consensus is Hard," Princeton, March 27, 2014
- Bitcoin Governance Research
- Orange Paper / Bitcoin Commons consensus spec
- Differential testing guide
- bitcoin/bitcoin

Related: The Social Layer Is the Attack Surface, Who Controls Bitcoin, Governance Paralysis Was The Victory, Why Bitcoin Needs a Specification.

SECURESOVEREIGN

Markdown is the canonical source: https://secsov.com/articles/bitcoins-hidden-crisis/index.md

Formatted snapshot of https://secsov.com/articles/bitcoins-hidden-crisis

Josh · Secure Sovereign