Bitcoin Commons: A Cryptographic Governance System for Bitcoin Development

BTCDecoded

2025

Abstract

Bitcoin faces a critical governance asymmetry: while its technical consensus layer is cryptographically bulletproof, its development governance relies on informal social coordination. At Bitcoin's multi-trillion dollar scale, this represents an existential vulnerability.

This whitepaper presents two innovations that enable each other: **BLLVM** provides mathematical rigor (proofs locked to code, formal verification, consensus matching); **Bitcoin Commons** provides governance coordination without civil war (Ostrom's principles through cryptographic enforcement). Together they solve Bitcoin's governance asymmetry.

The system is being developed across public repositories (see Section 9), with work ongoing on mathematical specifications, governance infrastructure, and economic sustainability. This is a living

document: the foundation exists, but its future depends on community contribution. For the complete narrative treatment, see *Bitcoin Commons: Decentralizing the Decentralizers*.

1. Introduction

Bitcoin solved Byzantine consensus between strangers (Nakamoto, 2008) but ignored consensus between developers. The network's substantial market capitalization demands institutional maturity matching its technical excellence.

The original cypherpunk developers focused on eliminating trusted third parties in transactions but inadvertently created trusted parties in development. Bitcoin Commons addresses Bitcoin's most critical vulnerability: governance asymmetry between technical consensus and development coordination.

1.1 The Talent Bottleneck: Orders of Magnitude and Sources

Bitcoin development draws on multiple hard domains simultaneously (C++, applied cryptography, distributed systems, security engineering, economics/game theory, and open-source governance). Each extra domain narrows the pool. Using conservative, sourced baselines and clearly labeled assumptions, we estimate the rarity of a contributor who combines these competencies and is available to work on Bitcoin:

Assumptions and sources:

- World population baseline: ~8.1B (UN DESA, World Population Prospects, 2022 Rev.)
- Global developers: ~30M-47M (range spanning widely cited industry estimates, incl. SlashData and similar studies)
- C++ share of developers: ~15%-25% (range spanning major annual developer surveys)
- Adult numeracy (problem-solving proficiency): on the order of 10%-20% globally (OECD PIAAC cross-country evidence; global extrapolation is approximate)
- Bitcoin Core maintainers: single-digit individuals; contributors: hundreds (public repo statistics)

Rarity funnel (indicative, overlapping, not strictly independent):

- Strong college-level math (calculus/linear algebra): 3%-5% of population, resulting in 240M-400M
- Professional developers: ~30M-47M (subset, separate baseline)
- C++/systems competency: 15%-25% of developers → 4.5M-11.8M
- Applied cryptography + Bitcoin protocol literacy: 1%-2% of C++ devs → 45k-236k
- Distributed systems/P2P networking depth: 30%-50% → 13.5k-118k

- Security engineering mindset (memory safety, adversarial thinking): 20%-30% → 2.7k-35k
- Economics/game-theory literacy: 30%-50%, resulting in 0.8k-17.5k
- Open-source governance (review culture, consensus norms): 10%-30%, resulting in 80-5k
- Communication/reliability under public scrutiny: 30%-50% → 24-2.5k
- Availability/alignment to actually work on Bitcoin: 10%-30% → ~2-750

Interpretation:

- Even with generous ranges, the intersection yields on the order of dozens to a few hundred globally available individuals with the full stack to work reliably on Bitcoin's most sensitive layers.
- Public data corroborates scarcity at the tip: Bitcoin Core has hundreds of credited contributors but only a small, rotating single-digit maintainer set. This human bottleneck contrasts with the cryptographic abundance at the consensus layer.

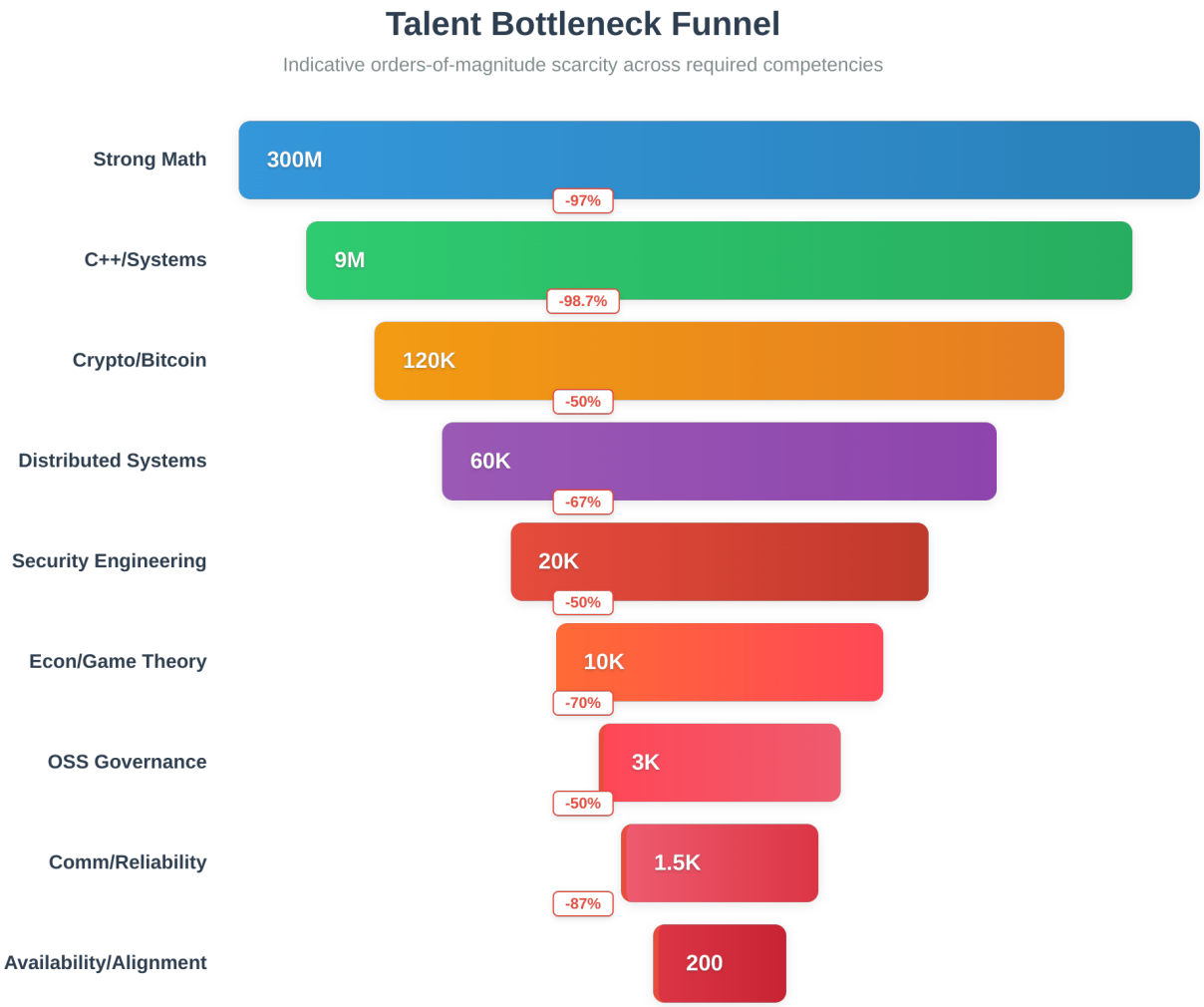

Figure: Orders of magnitude funnel showing talent scarcity across required domains.

Citations (illustrative anchors): - UN DESA, World Population Prospects (2022) - SlashData, Global Developer Population Trends - Annual developer surveys (Stack Overflow) for C++ usage - OECD PIAAC, adult skills numeracy distributions - Bitcoin Core repository statistics (GitHub)

2. Problem Statement

Technical Reality

Bitcoin's consensus rules are embedded in 350,000+ lines of C++ code with no mathematical specification. Bitcoin Core maintains 99.5% market share among implementations, creating effective monopoly control over Bitcoin's evolution. The lack of formal specification makes it impossible to build safe alternative implementations or verify consensus correctness.

Governance Reality

Bitcoin's development governance relies entirely on informal social coordination. There are no systematic consequences for bad actors, no formal dispute resolution mechanisms, and power

is invisible and unaccountable. The system is vulnerable to capture through relationships rather than rules. Network analysis reveals structural misalignments between technical development and social infrastructure, creating coordination gaps that prevent effective governance (Hough, 2025). These patterns reflect the paradox of embeddedness in network structures, where relationships can paradoxically inhibit coordination and reinforce existing power dynamics (Uzzi, 1997). Funding may not flow to projects with strong grassroots activity, and "rich-get-richer" dynamics reinforce existing patterns rather than enabling competition.

Historical Context

Early developers recognized this problem. Gavin Andresen (2014) raised governance concerns but was marginalized during blocksize wars. Mike Hearn attempted governance solutions but proposed hierarchical models inappropriate for Bitcoin's decentralized ethos. Academic researchers (De Filippi & Loveluck, 2016) documented these power structures but provided no actionable solutions.

Scale Considerations

Bitcoin's growth from early stages to multi-trillion dollar scale requires institutional reform. The next crisis, whether AI attacks, regulatory capture, or internal conflicts, won't wait for the community to develop governance solutions reactively.

3. Theoretical Framework: The Triple Foundation

Bitcoin Commons synthesizes three distinct theoretical frameworks, each addressing weaknesses in the others to create governance architecture stronger than any single approach alone.

Framework 1: Elinor Ostrom - Commons Governance

Elinor Ostrom won the 2009 Nobel Prize in Economics for proving that shared resources don't inevitably collapse into chaos or capture (Ostrom, 1990). Her research documented principles for governing commons without central authority across centuries of real-world examples.

Ostrom's (1990) Seven Principles:

1. **Clear boundaries on who decides what** - Defined decision-making authority
2. **Consequences for violations** - Systematic enforcement mechanisms
3. **Local dispute resolution** - Formal conflict resolution processes
4. **Protection from external interference** - Resistance to outside pressure
5. **Collective choice arrangements** - Meaningful participation in rule-making
6. **Graduated sanctions** - Proportional consequences for violations
7. **Monitoring and accountability** - Transparent oversight mechanisms

What This Provides: Proven institutional design for shared resources; evidence decentralized governance works; coordination without hierarchy.

Framework 2: F.A. Hayek - Spontaneous Order

Friedrich Hayek's Austrian economics provides the competitive discovery mechanism that enables governance evolution rather than rigid design.

Hayek's Core Insights: - **Dispersed Knowledge Problem** - No central planner can know what's needed because knowledge is distributed across many actors - **Competition as Discovery** - Competition reveals information that couldn't be known in advance - **Spontaneous Order** - Best institutions emerge through evolution, not top-down design - **Markets Need Infrastructure** - Competition requires actual alternatives to compete

What This Provides: Justification for avoiding central planning; competitive governance discovery; institutions evolve through market signals.

Framework 3: Bitcoin - Cryptographic Enforcement

Bitcoin's innovation provides the enforcement tools that make decentralized governance work at scale without trusted parties.

Bitcoin's Core Principles:

- **Don't Trust, Verify** - Cryptographic enforcement replaces social trust
- **Permissionless Innovation** - Anyone can build without asking permission
- **Exit Rights** - Fork option provides ultimate check on power
- **Decentralized Control** - No single point of authority

What This Provides: Tools for enforcing rules without trust; proof decentralized systems work at scale; model for implementing Hayek's principles digitally.

The Triple Synthesis

The three frameworks address each other's weaknesses:

Ostrom's Challenge: Commons governance historically relied on social pressure, vulnerable to capture at scale **Bitcoin's Solution:** Cryptographic enforcement replaces social pressure with mathematical proof

Hayek's Challenge: Competition discovers optimal solutions but requires actual alternatives to compete
Ostrom's Solution: Provides institutional framework for multiple governance models to coexist

Bitcoin's Challenge: Solved technical consensus but not social governance **Hayek + Ostrom Solution:** Competitive discovery of governance models using proven institutional principles

The Result: Governance that is proven (Ostrom's research), evolving (Hayek's competition), and enforceable (Bitcoin's cryptography).

Bitcoin Core's Current State

Bitcoin Core has informal implementations of some Ostrom (1990) principles but lacks systematic enforcement. The mapping below details how Commons implements all seven principles through technical architecture.

Mapping The Principles to Implementation

The modular architecture implements Ostrom's seven principles through cryptographic enforcement rather than social pressure. The chart below shows how these principles integrate with principles from Hayek, Bitcoin, and Cypherpunk frameworks:

Principle	Source	Technical Implementation
Clear Boundaries Who has authority, what belongs in base vs extensions, entry/exit rules for maintainers	Ostrom	**Repository hierarchy with signature thresholds:** Constitutional (6-of-7), Implementation (4-of-5), Application (3-of-5), Extension (2-of-3). Each repository has defined scope and authority boundaries enforced by multisig. `governance-app/src/crypto/multisig.rs`
Proportional Benefits Costs and benefits align with participation level, no free-riding by maintainers	Ostrom	**Economic node network and module marketplace:** Economic nodes receive proportional voting weight based on stake. Module developers earn marketplace revenue based on adoption. Merge mining rewards align with contribution to network security. `governance-app/src/economic_nodes/registry.rs`
Collective Choice Participants design governance rules, not imposed from above	Ostrom	**Module system enables user sovereignty:** Users choose which modules to run, creating bottom-up governance through configuration rather than top-down feature decisions. Module marketplace enables competitive discovery of solutions. `reference-node/ (modular architecture)`
Monitoring Transparent oversight, audit trails, verifiable governance actions	Ostrom	**Three-layer verification architecture:** GitHub enforcement (real-time merge blocking), Nostr transparency (hourly signed status), OpenTimestamps anchoring (monthly immutable proof). All governance actions are cryptographically signed and verifiable. `governance-app/src/verification/`
Graduated Sanctions Proportional consequences for violations, not just fork-or-nothing	Ostrom	**Tiered security controls and emergency response:** P0 controls block production, P1 controls require audit, P2 controls warn. Emergency powers have automatic expiration. Module quality gates provide graduated enforcement (warnings → rejection → removal). `governance-app/src/enforcement/security_controls.rs`
Conflict Resolution Local mechanisms for disputes, formal processes not ad-hoc	Ostrom	**OpenTimestamps provides court-admissible evidence:** Monthly canonical registry anchored to Bitcoin blockchain creates immutable proof of governance state. Enables formal dispute resolution with cryptographic evidence rather than ad-hoc social negotiation. `governance-app/src/verification/opentimestamps.rs`
Minimal Recognition Autonomy from external interference, protection from regulatory capture	Ostrom	**Multi-jurisdictional maintainer distribution:** 5-7 maintainers across multiple jurisdictions. Cryptographic enforcement means even repository admins cannot bypass multisig requirements. No single point of regulatory pressure. `governance-app/src/crypto/multisig.rs`
Spontaneous Order Emergent coordination without central planning, bottom-up innovation	Hayek	**Module marketplace enables emergent solutions:** No central committee decides features. Module developers build solutions based on market demand. Users vote with configuration choices. Innovation emerges from competition, not planning. `Module system (planned architecture)`
Knowledge Problem Distributed knowledge utilization, no central bottleneck for decisions	Hayek	**Modular architecture distributes decision-making:** Module system allows distributed knowledge application. Different modules solve different problems without requiring central coordination. Economic nodes contribute local knowledge through voting signals. `reference-node/ (modular architecture)`
Price Discovery Market signals guide resource allocation, module marketplace enables choice	Hayek	**Module marketplace pricing and adoption metrics:** Module sales, adoption rates, and user ratings provide market signals. Popular modules get more resources and development. Unpopular modules exit. Price signals guide developer effort allocation. `Module system (planned architecture)`
Market Process Competitive discovery and adaptation, multiple implementations compete	Hayek	**Orange Paper enables implementation diversity:** Mathematical specification allows multiple implementations to compete while maintaining consensus compatibility. Module system enables experimentation without consensus risk. Multiple implementations can coexist and compete. `the-orange-paper/`
Permissionless No gatekeeping for participation, anyone can fork and contribute	Bitcoin	**Open source with fork-ready architecture:** All repositories public. Modular design makes forking easier. Module system allows contribution without touching core consensus. Anyone can create modules, fork implementations, or start new implementations from Orange Paper. `All repositories public, fork-ready`
Trust Minimization Cryptographic verification over trust, governance enforced in code	Bitcoin	**Multisig enforcement blocks unauthorized merges:** 3-of-5 maintainer multisig required for merges, enforced by GitHub App. Even repository admins cannot bypass. Signatures are cryptographically verified. Governance is enforced in code, not through social trust. `governance-app/src/enforcement/merge_block.rs`
Decentralization Distributed control and validation, no single point of failure	Bitcoin	**Multi-stakeholder maintainer set across jurisdictions:** 5-7 maintainers required for operations. Economic node network provides distributed validation. Module system prevents single implementation monopoly. Multiple implementations can validate same chain. `governance-app/src/validation/threshold.rs`
Censorship Resistance Resistance to external pressure, no single jurisdiction control	Bitcoin	**Jurisdictional distribution and cryptographic enforcement:** Maintainers across multiple jurisdictions. Cryptographic enforcement prevents single-jurisdiction shutdown. Self-funding through merge mining reduces external pressure. No single point of regulatory control. `governance-app/src/validation/threshold.rs`
Immutability Permanent historical record, governance decisions cryptographically anchored	Bitcoin	**OpenTimestamps anchors governance state to Bitcoin:** Monthly canonical registry hash anchored to Bitcoin blockchain via OpenTimestamps. Creates permanent, court-admissible proof of governance state. Independent of any single server or relay. `governance-app/src/verification/opentimestamps.rs`
Privacy by Default User privacy as fundamental right, governance doesn't require surveillance	Cypherpunk	**Governance operates without user surveillance:** Multisig verification doesn't require monitoring users. Module choices are local configuration, not tracked. OpenTimestamps proofs don't reveal private information. Privacy-preserving governance design. `governance-app/src/crypto/`
Technical Enforcement Code enforces rights and limits, multisig prevents governance bypass	Cypherpunk	**Cryptographic enforcement throughout:** Multisig blocks unauthorized merges at GitHub level. Security controls enforced in CI/CD. Signatures verified mathematically. Code enforces governance rules; humans cannot override without cryptographic proof. `governance-app/src/enforcement/merge_block.rs`
Anti-Surveillance Resistance to monitoring and control, decentralized governance infrastructure	Cypherpunk	**Distributed infrastructure and privacy-preserving design:** Nostr relays distributed across network. OpenTimestamps uses public Bitcoin blockchain (no surveillance risk). Self-hosted runners behind VPN. No central monitoring infrastructure required. `governance-app/src/verification/nostr.rs`

Figure: Integration of four key philosophies: Hayek (spontaneous order), Bitcoin (cryptographic enforcement), Cypherpunk (privacy through technology), and Ostrom (commons governance).

Principle 1: Clear Boundaries

- **Layers**: Base (network consensus), Module (user choice), Economic (miner coordination)
- **Implementation**: Architecture enforces boundaries: modules cannot modify consensus code paths

Principle 2: Consequences for Violations

- **Economic/Technical/Reputational**: Merge mining leverage, module quality standards, transparent adoption metrics
- **Implementation**: Cryptographic enforcement makes consequences automatic, not social

Principle 3: Local Dispute Resolution

- **Architecture-based**: Competing modules resolve disputes; user choice determines winners; module conflicts don't threaten consensus
- **Implementation**: No central arbiter needed: architecture provides resolution through user configuration

Principle 4: Protection from External Interference

- **Self-Funding/Multi-jurisdictional/Fork-Ready**: Merge mining revenue, distributed keyholders, governance fork capability
- **Implementation**: Cryptographic multisig ensures no single jurisdiction can compel action

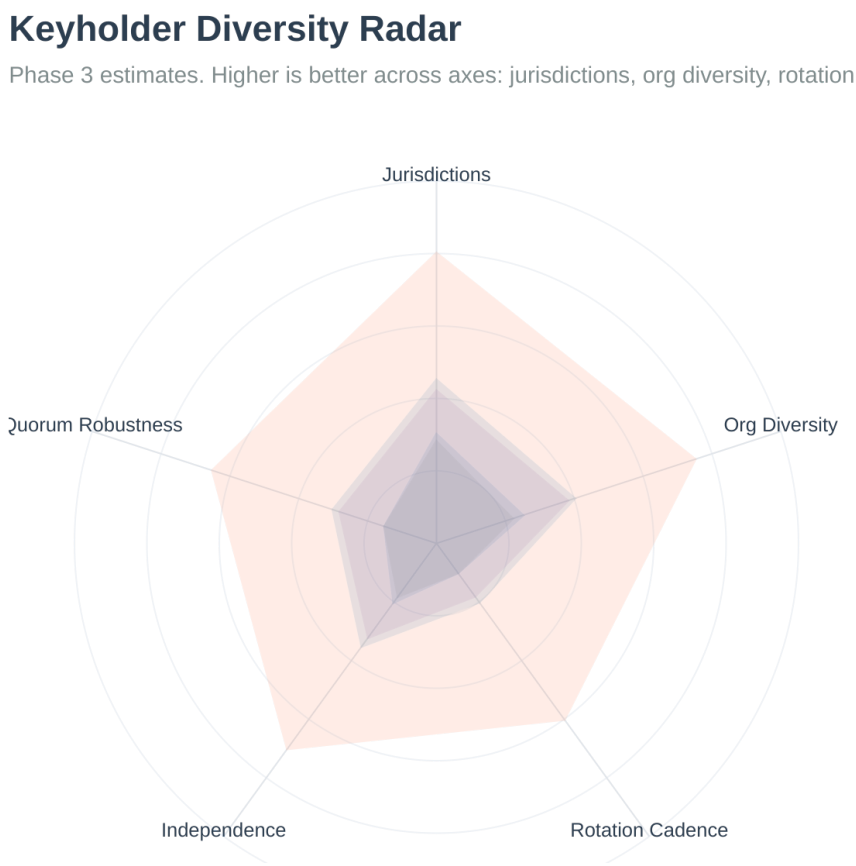

Figure: Keyholder diversity across jurisdictions and organizations prevents single-point coercion.

Principle 5: Collective Choice Arrangements

- **User sovereignty via configuration**: Users compose stacks; adoption metrics function as voting; participation through choices, not committees
- **Implementation**: GUI-based module selection enables collective choice through user preferences

Principle 6: Graduated Sanctions

- **Proportional escalation**: Moderate fork → major deprecation → governance fork; graduated economic pressure through merge mining; sanctions at module/economic layer, no consensus changes required
- **Implementation**: Multisig thresholds vary by change category (2-of-3 extension to 6-of-7 constitutional)

Principle 7: Monitoring and Accountability

- **Cryptographic transparency**: All governance actions signed and verifiable; module adoption, revenue flows, decision provenance auditable; three-layer verification (GitHub/Nostr/OpenTimestamps)
- **Implementation**: Automated monitoring through cryptographic verification, not social trust

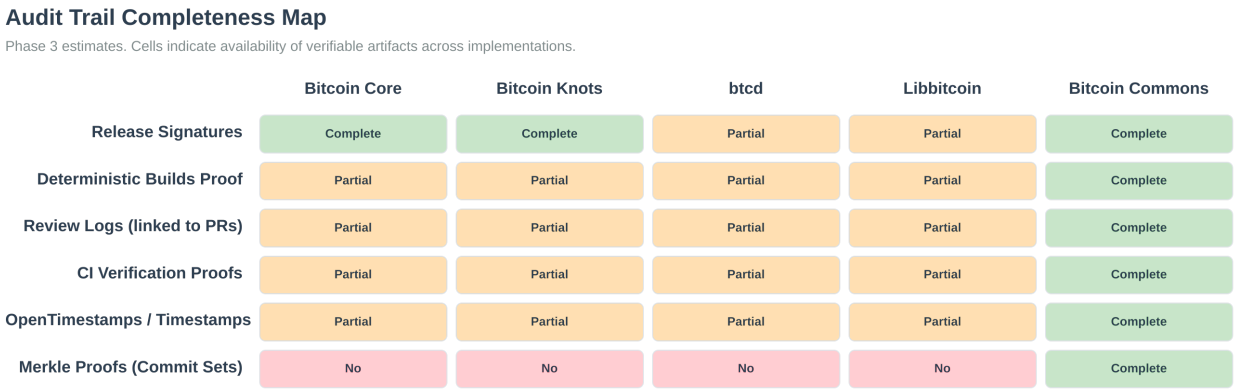

Figure: Audit-trail completeness across governance layers: all decisions evidenced and verifiable.

Comparison with Bitcoin Core:

Bitcoin Core has informal implementations of some Ostrom principles but lacks systematic enforcement. The system has informal boundaries (Core maintainers, BIP editors) but no formal process for selection, removal, or authority limits. Social pressure and reputation damage provide consequences, but there's no systematic enforcement mechanism. Most critically, there's no infrastructure for competitive discovery. Bitcoin Core's market dominance (see Section 2.1) prevents Hayekian competition from working. Below is a detailed comparison:

- Clear Boundaries: Informal (maintainers, BIP editors), no formal selection/removal process
- Consequences: Social pressure only, no systematic enforcement
- Dispute Resolution: BIP process advisory only, no binding mechanism
- External Protection: No systematic protection, individuals can be pressured
- Collective Choice: BIP process exists but no formal consensus mechanism
- Graduated Sanctions: Informal social pressure, no systematic escalation
- Monitoring: Public GitHub/mailing lists, no formal accountability system

The pattern: Bitcoin Core has informal implementations that worked at billion-scale but become vulnerable at multi-trillion scale. Commons implements all seven principles through technical architecture and cryptographic enforcement.

4. Technical Solution: The Orange Paper

Problem

Bitcoin's consensus rules lack mathematical specification (see Section 2.1). This makes them impossible to verify, understand, or implement independently. The 2018 inflation bug (CVE-2018-17144) existed in Bitcoin Core for years before discovery. This is exactly the class of error formal verification eliminates.

Solution

The Orange Paper provides a formal mathematical specification of Bitcoin's consensus protocol through AI-assisted extraction from Bitcoin Core's codebase. The specification includes:

- Mathematical foundations (set theory, cryptographic primitives, network protocols)
- State transition functions (block validation, transaction validation, consensus rules)
- Economic model (mining rewards, fee calculations, difficulty adjustment)
- Security properties (Byzantine fault tolerance, Sybil resistance, double-spending prevention)

Benefits

- **Safe alternative implementations**: Independent implementations can verify against mathematical specification
- **Formal verification**: Consensus correctness can be mathematically proven
- **Reduced consensus bugs**: Systematic analysis eliminates entire classes of errors
- **Technical moat**: AI extraction eliminates "not invented here" bias

AI-Assisted Extraction Methodology

The Orange Paper uses AI-assisted extraction from Bitcoin Core's codebase to formalize consensus rules. This approach: - Analyzes Bitcoin Core's codebase (see Section 2.1) to identify consensus-critical code paths - Extracts mathematical relationships from implementation details - Creates formal specifications that are independent of specific code structure - Enables verification that specification matches actual network behavior

Proof Maintenance and Specification Quality

The formal verification process includes ongoing maintenance to ensure specification accuracy:

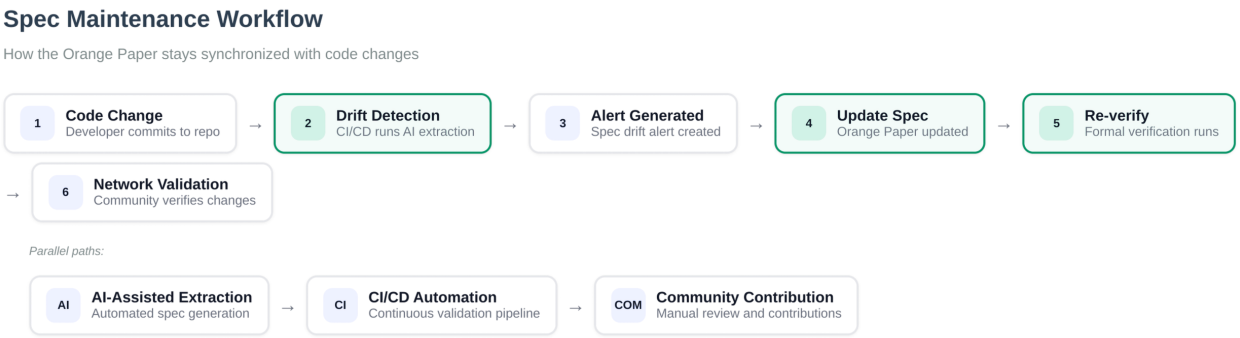

Figure: Spec maintenance workflow: specification synchronized with implementation through automated testing and formal verification.

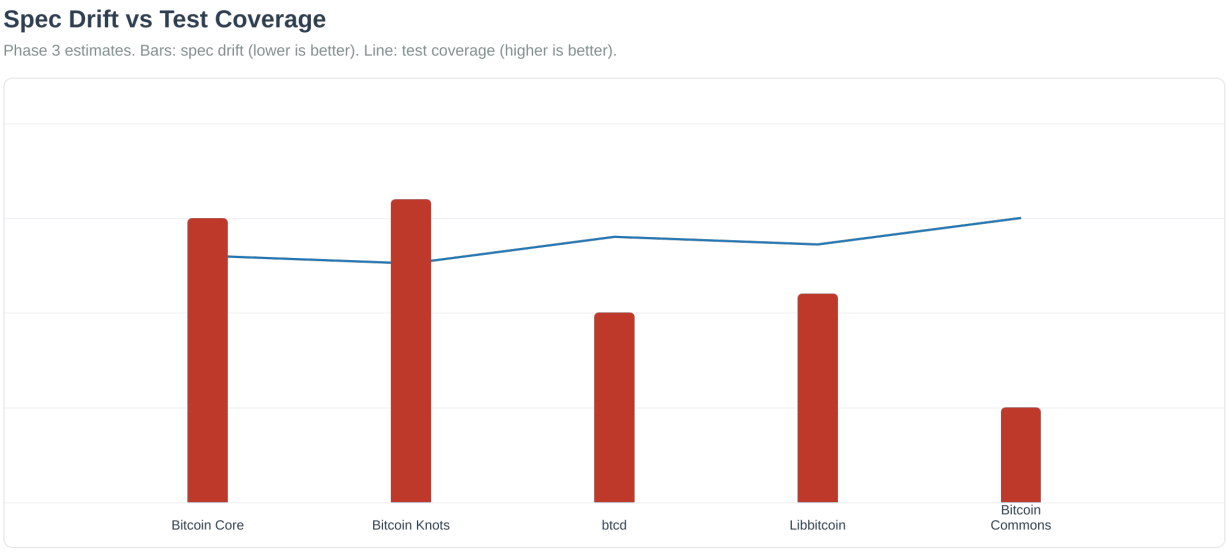

Figure: Spec drift decreases as test coverage increases. Higher test coverage ensures specification accuracy over time.

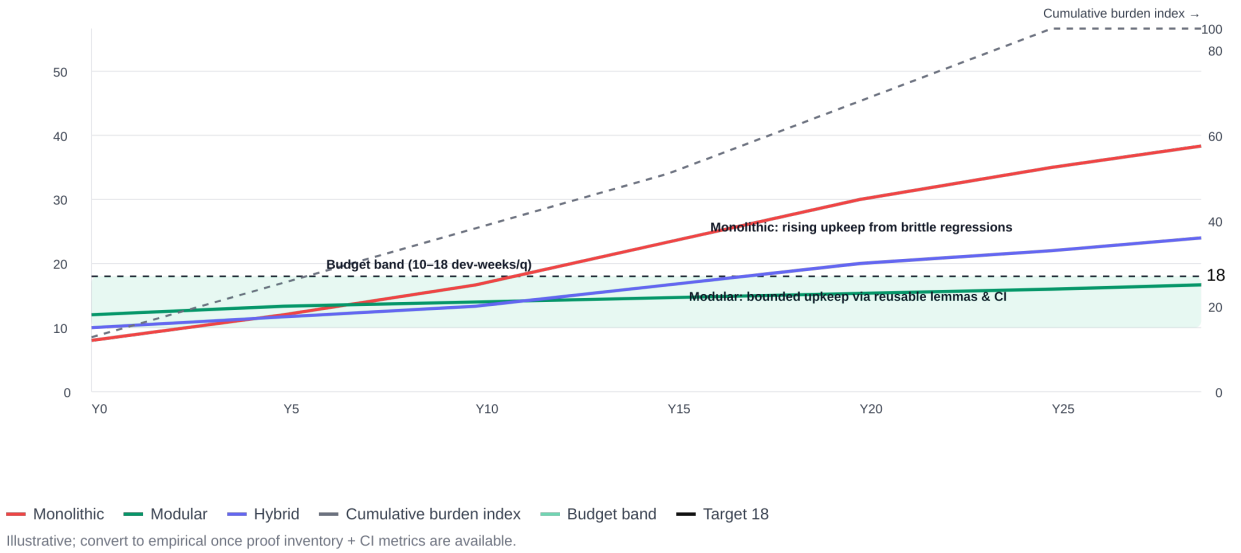

Figure: Proof maintenance cost by area, highlighting refactor hotspots. Commons aims for lower proof churn than Core.

Status: Complete specification available at https://github.com/BTCDecoded/the-orange-paper. The specification is actively maintained and verified against network behavior through automated testing.

4.4 BLLVM Architecture

BLLVM (Bitcoin LLVM) applies compiler-like infrastructure to Bitcoin implementations. The Orange Paper serves as an intermediate representation (IR), enabling reusable optimizations and multiple implementations.

Single Source of Truth: All consensus logic resides in `consensus-proof`. Upper tiers (`protocol-engine`, `reference-node`) delegate validation calls with no duplicate implementations. Path dependencies ensure changes propagate immediately through Rust's type system.

Optimization Pipeline: Multiple passes apply: formal verification (Kani model checking), property testing (proptest edge case discovery), LLVM compiler optimizations (opt-level 3, fat LTO, SIMD), and differential testing against network behavior.

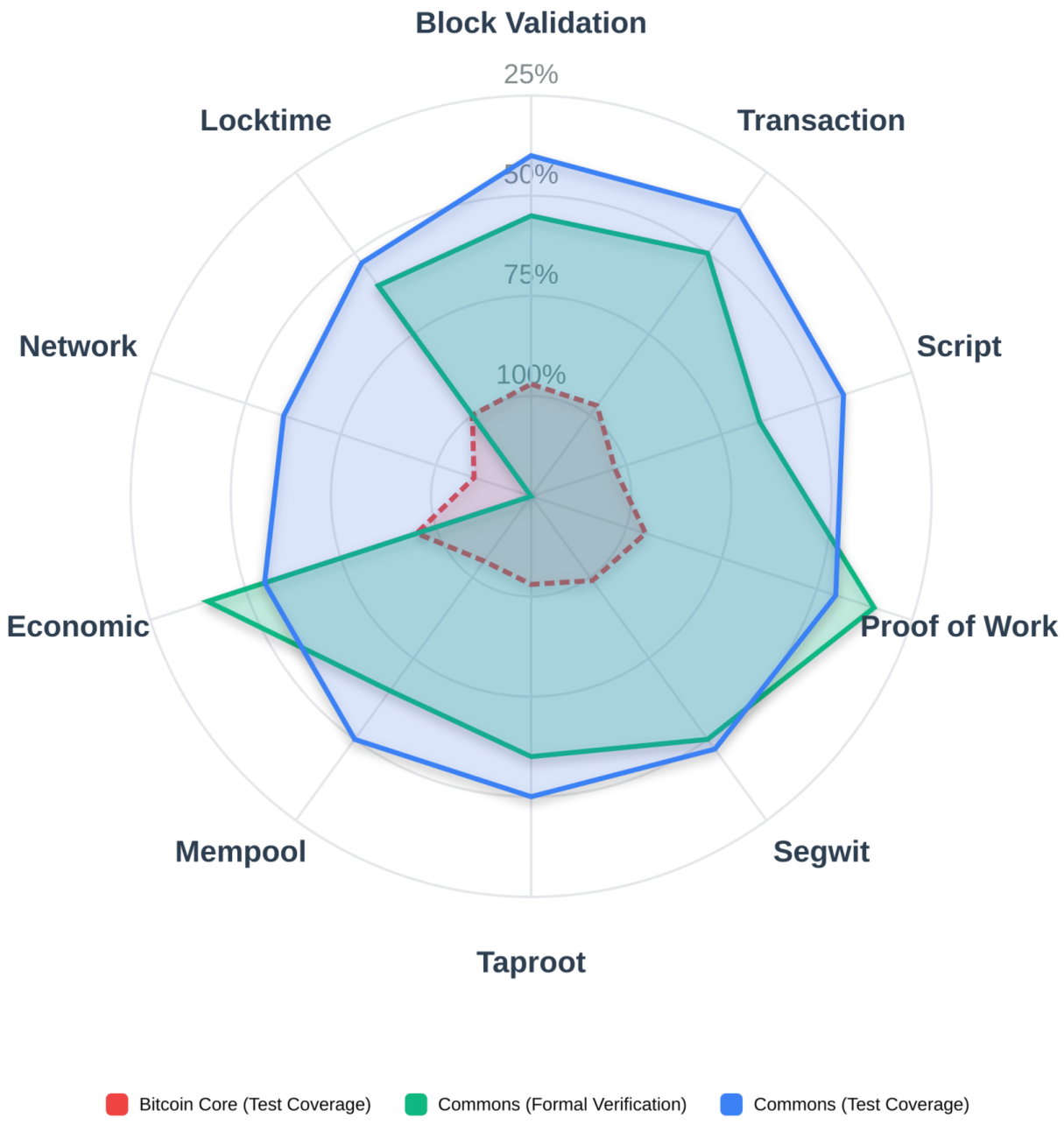

Figure: Consensus coverage comparison: Bitcoin Core achieves 25% coverage through testing alone. Bitcoin Commons achieves 65% formal verification coverage (172 Kani proofs) plus 77% test coverage. Commons uses 93 consensus-focused test files with 667+ test functions compared to Core's 316 total files (only ~53 consensus-focused). The mathematical specification enables both formal verification and comprehensive testing.

Formal verification in `consensus-proof` applies to all tiers because all consensus decisions flow through verified functions. The dependency chain prevents bypassing verification.

5. Architectural Solution: Modular Governance

Two innovations work together: **BLLVM** provides the mathematical foundation and compiler-like architecture (Orange Paper as IR, formal verification passes); **Commons** provides the governance framework (coordination without civil war). The modular architecture is where both innovations meet. BLLVM ensures correctness through architectural enforcement; Commons ensures coordination.

Three-Layer Stack

The modular architecture consists of three layers that transform governance conflicts from political battles into architectural choices:

Layer 1: Mandatory Consensus (Base Node)

- Bitcoin's consensus rules, unchangeable without network agreement
- Cryptographically enforced, defines what "Bitcoin" means
- Examples: block validation, transaction validation, fork choice rules

Layer 2: Optional Modules (Extension System)

- User-controlled features that can be enabled or disabled
- Communities can fork/modify/compete, user choice determines winners
- Examples: Lightning Network, merge mining, Taproot Assets, privacy enhancements

Layer 3: Economic Coordination (Revenue Model)

- Self-sustaining development through merge mining revenue
- 1% fee on merged chain rewards, scales with adoption
- Revenue allocation (60% core, 25% modules, 10% audits, 5% ops) — see Section 7.2 for details

Module Isolation

Modules run in separate processes with strict boundaries:

Process Isolation Mechanisms: - Each module runs in its own process space with isolated memory - Modules communicate only through well-defined APIs - Base node validates all blocks using Orange Paper specification regardless of enabled modules - Module state completely separate from consensus state (UTXO set)

API Boundaries: - Modules can only interact with base layer through documented interfaces - No direct access to consensus functions or core data structures - Module failures isolated and cannot propagate to base node - Crash containment guaranteed by process boundaries

What modules CANNOT do: Modify consensus rules, alter block validation, cause network splits

What modules CAN do: Process their own state, crash without affecting base node

Containment Strategy

The modular architecture satisfies both camps simultaneously:

- **"Don't Change Bitcoin" Camp**: Gets pure Bitcoin base layer with no modifications
- **"Make Bitcoin Useful" Camp**: Gets optional features through modules
- **Miners**: Get additional revenue from merge mining

The Module System IS The Governance System: Instead of governing through committees deciding features, we govern through architecture enabling choice. The module system isn't just technical: it's the governance mechanism itself, implementing Ostrom's collective choice arrangements through user configuration, Hayek's competitive discovery through module competition, and Bitcoin's permissionless innovation through fork-ability.

Architecture Diagrams

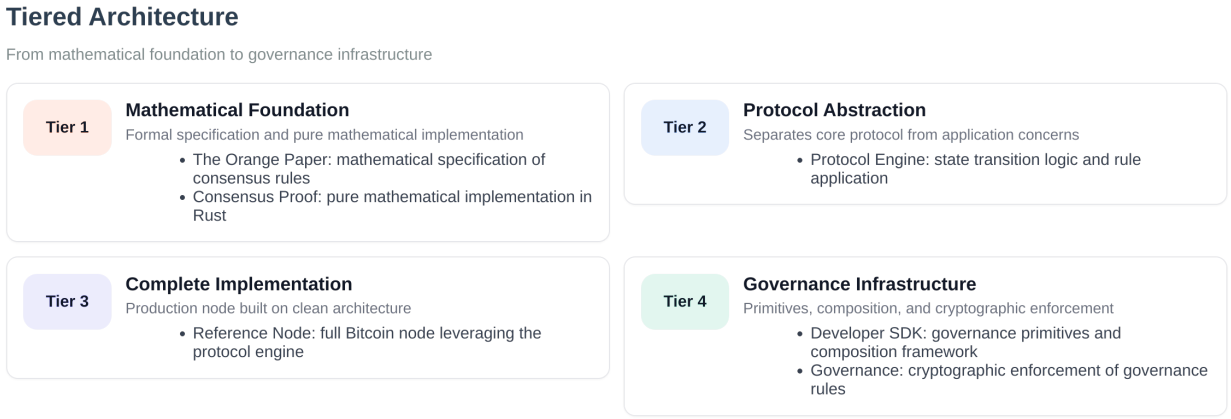

Figure: Tiered architecture: Tier 1 = Orange Paper + Consensus Proof (mathematical foundation); Tier 2 = Protocol Engine (protocol abstraction); Tier 3 = Reference Node (complete implementation); Tier 4 = Developer SDK + Governance (governance infrastructure).

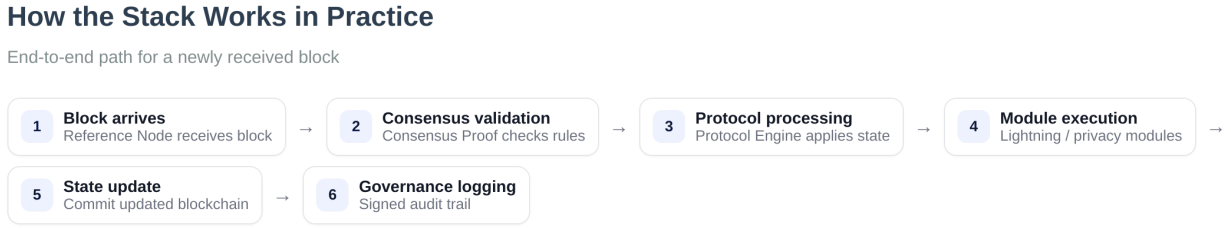

Figure: End-to-end data flow through Reference Node, Consensus Proof, Protocol Engine, modules, and governance. Each tier depends only on layers below; modules cannot affect consensus.

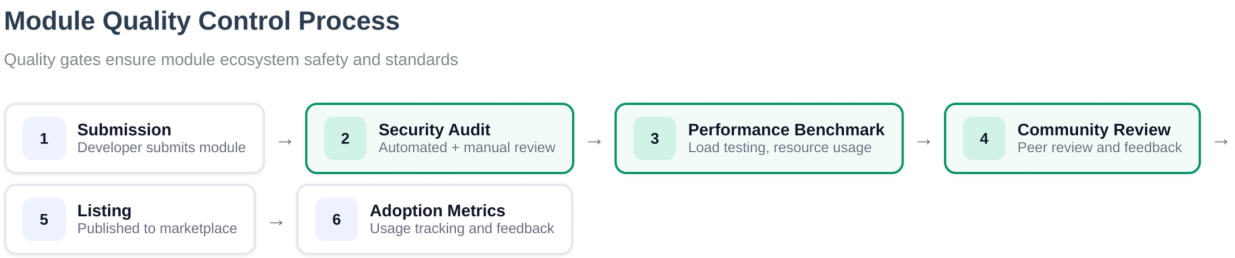

Figure: Module quality control process ensuring security, performance, and community validation before module adoption.

Fragmentation Analysis:

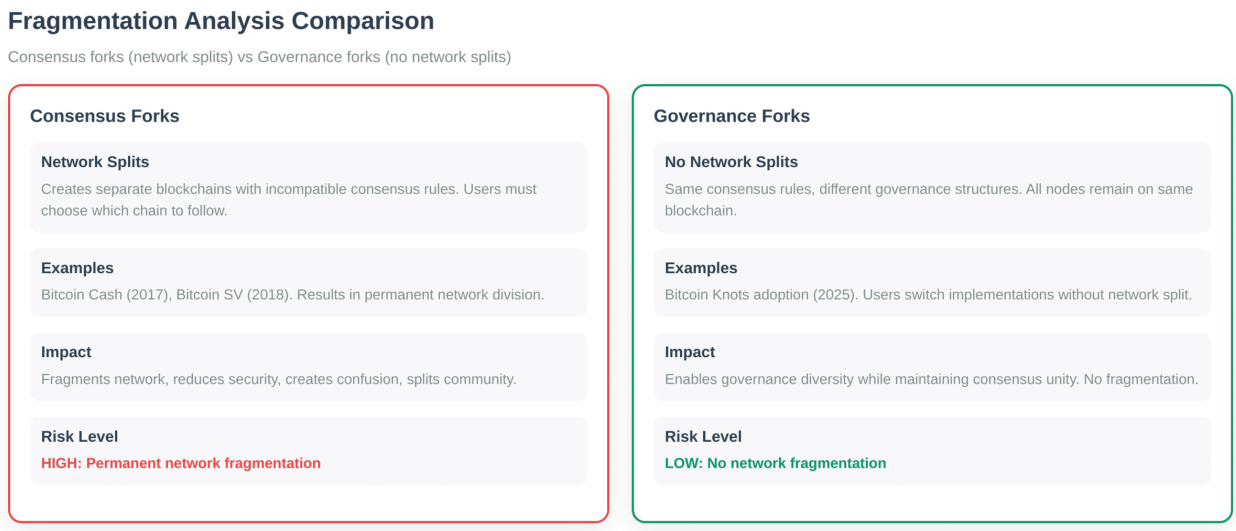

Figure: Fragmentation analysis showing that governance forks don't split the network. All implementations validate same Bitcoin consensus while enabling governance competition.

Governance forks preserve the consensus layer while allowing governance changes. Users can fork governance rules while keeping the same Bitcoin consensus. This is the ultimate accountability mechanism. Knots adoption (25% in five months) proved multiple implementations coexist without fragmentation.

6. Cryptographic Governance Enforcement

Commons implements cryptographic governance through three complementary verification layers that ensure both real-time transparency and immutable historical proof:

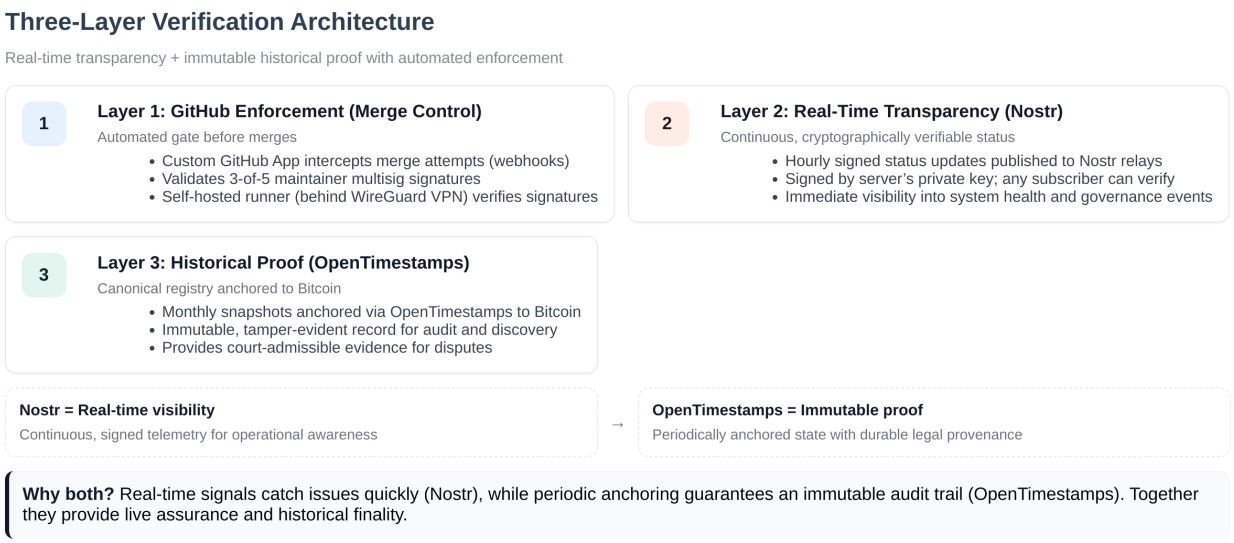

Figure: Three-layer verification approach: GitHub, Nostr, and OpenTimestamps.

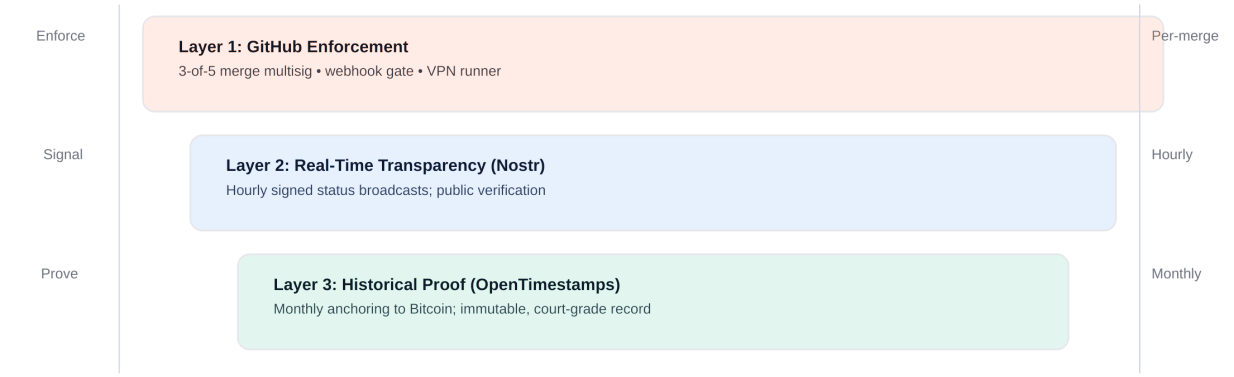

Figure: Three-layer verification: GitHub merge control, real-time Nostr transparency, and OpenTimestamps historical proof.

Layer 1: GitHub Enforcement (Merge Control) - Custom GitHub App validates multisig requirements (varies by layer: 2-of-3 to 6-of-7) - Self-hosted runner behind WireGuard VPN validates signatures using secp256k1 - Even repository admins cannot bypass cryptographic requirements - Signature validation happens before merge approval

Layer 2: Real-Time Transparency (Nostr) - Hourly status updates published to Nostr relays - Status includes: binary hash, config hash, recent merges, health metrics - Cryptographically signed by server's unique NPUB (Nostr public key) - Anyone can subscribe and verify server integrity in real-time - Missing updates trigger community alerts within 2 hours

Layer 3: Immutable Proof (OpenTimestamps) - Monthly canonical registry anchored to Bitcoin blockchain - Critical events (key rotations, deployments) timestamped immediately - Creates cryptographic proof of governance state at specific block height - Provides court-admissible evidence for dispute resolution - Works independently of any single server or relay

Cross-Layer Verification: Three independent layers verify governance actions and each other. Risk at one layer does not compromise the others. This defense-in-depth approach ensures governance integrity even if one verification method is compromised.

Repository Hierarchy

Different signature thresholds based on risk level (see Section 6.5 for explicit thresholds and details).

Emergency Response

Emergency situations require higher signature thresholds (4-of-5, 5-of-5) and extended time windows based on risk level, with automatic expiration to prevent permanent emergency powers. The tiered system escalates requirements proportionally to the severity of the situation while maintaining governance integrity.

Security Architecture: Push-Only Design

Security Architecture Details:

- **No HTTP Endpoints**: Governance servers have no incoming HTTP endpoints (minimal exposure surface)
- **VPN Isolation**: Servers communicate outbound only through WireGuard VPN
- **Self-Hosted Runner**: GitHub runner behind WireGuard VPN for signature validation
- **Data Flow**: Server to GitHub (push) to Nostr (publish) to Bitcoin (anchor)
- **Public Read Access**: GitHub repo, Nostr relays, Bitcoin blockchain (read-only for public)

Attack Path Protection:

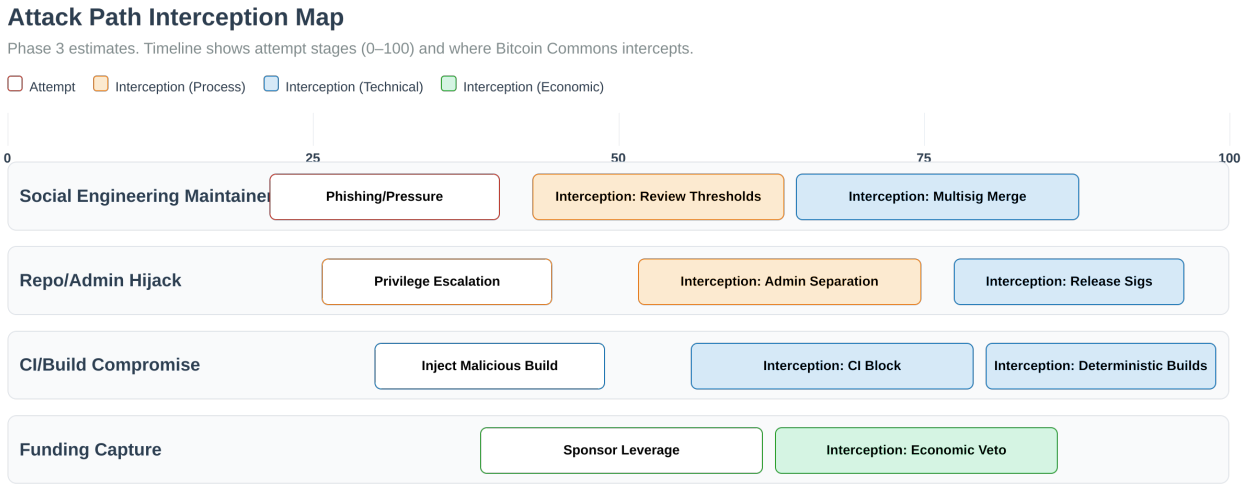

Figure: Risk interception points across three independent verification layers.

Multisig Threshold Details

The following thresholds define signature requirements for governance actions (referenced in Section 6.2):

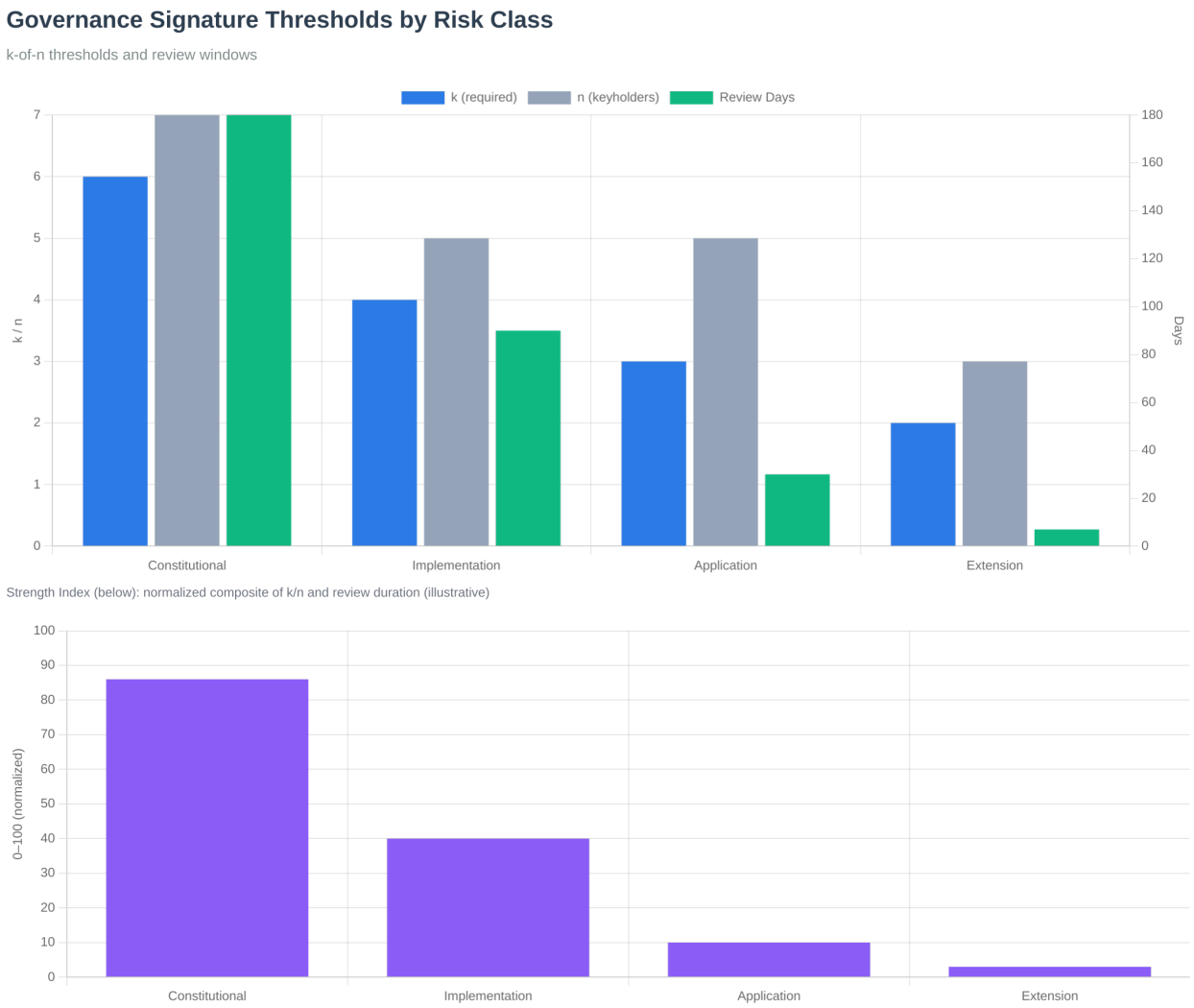

Figure: Governance signature thresholds by change category (constitutional, implementation, application, extension).

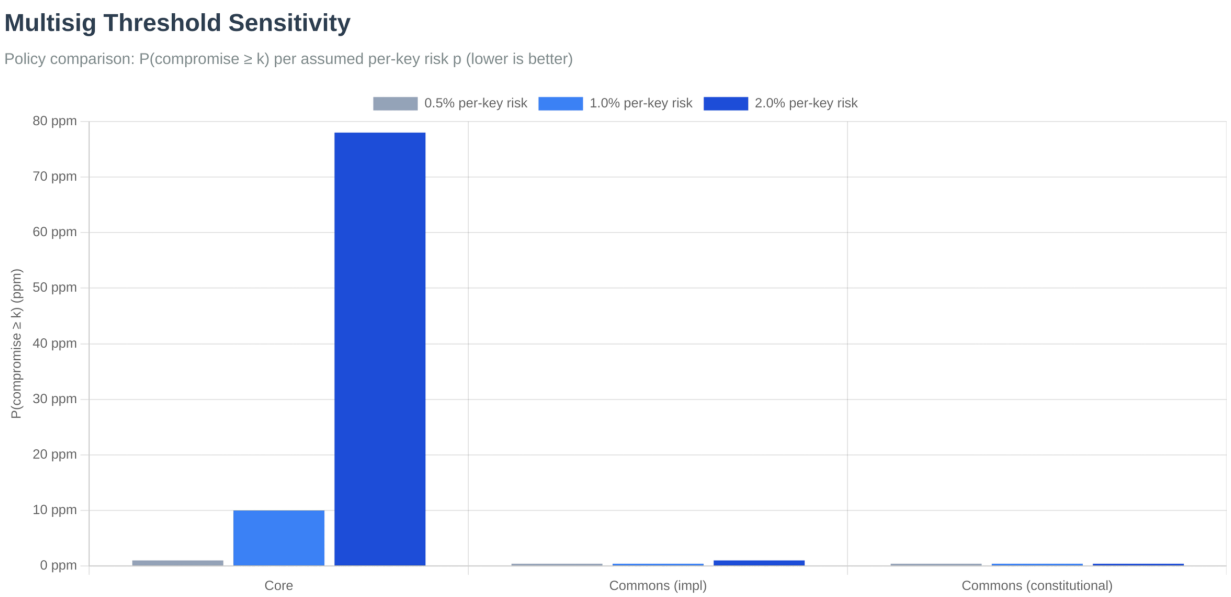

Figure: Multisig threshold sensitivity: false negative and false positive risk vs threshold. Commons balances safety and throughput through carefully calibrated thresholds.

Explicit Thresholds by Layer: - **Constitutional (Orange Paper)**: 6-of-7 maintainer signatures, 180-day review period - **Consensus Changes**: 5-of-5 signatures, 365-day review period (longest review) - **Implementation**: 4-of-5 signatures, 90-day review - **Application**: 3-of-5 signatures, 30-day review - **Extension**: 2-of-3 signatures, 7-day review

All signatures verified using secp256k1 (same curve as Bitcoin). GitHub App validates signatures before allowing merges. Even repository admins cannot bypass cryptographic requirements.

Governance Process and Latency

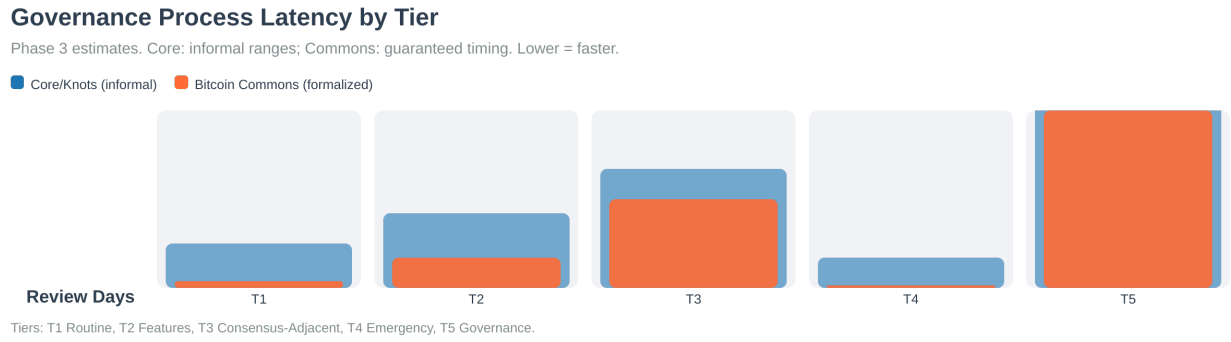

Figure: Governance process latency and escalation tiers. Stages map to proposal → review → approvals → merge.

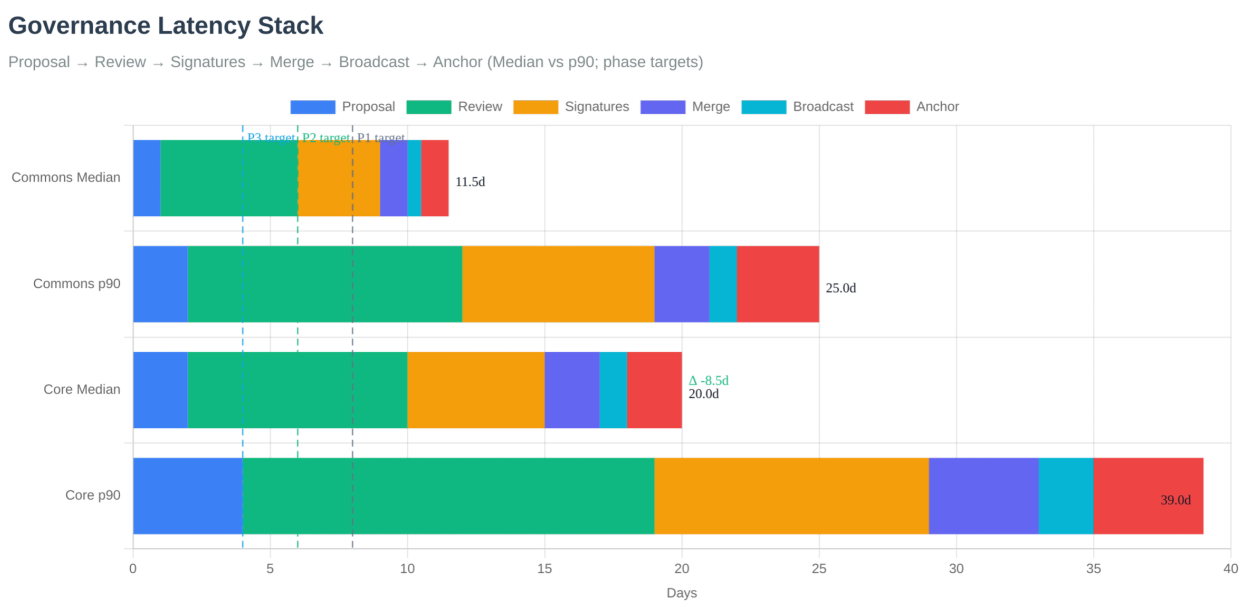

Figure: Governance latency: time by stage. Reduced queueing at gates through automation and process optimization.

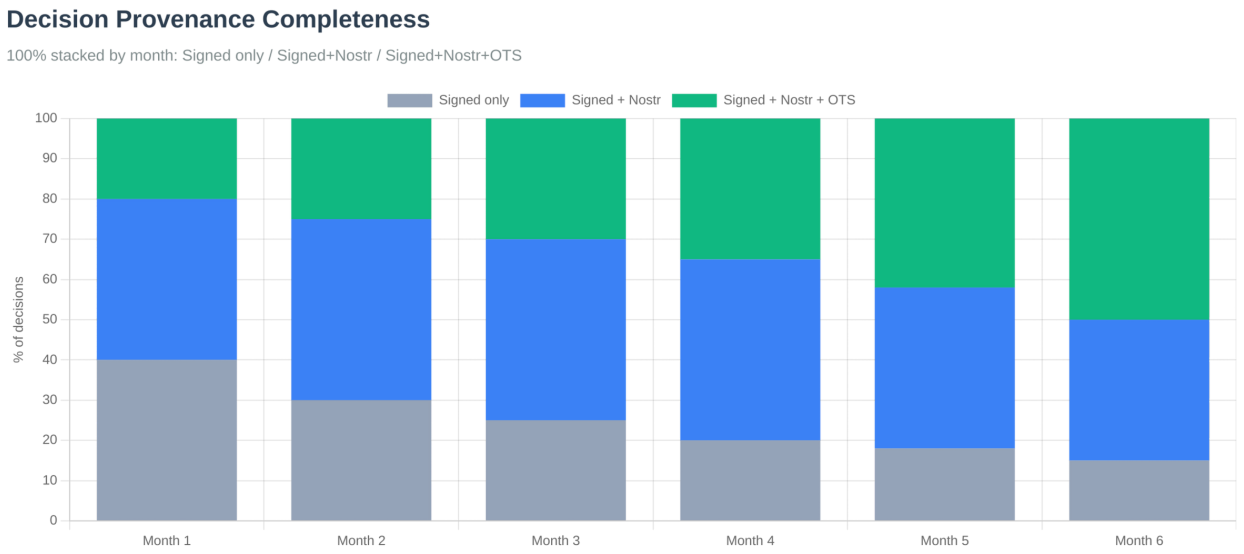

Figure: Decision provenance: share of fully evidenced decisions across layers. Three-layer verification ensures complete audit trails.

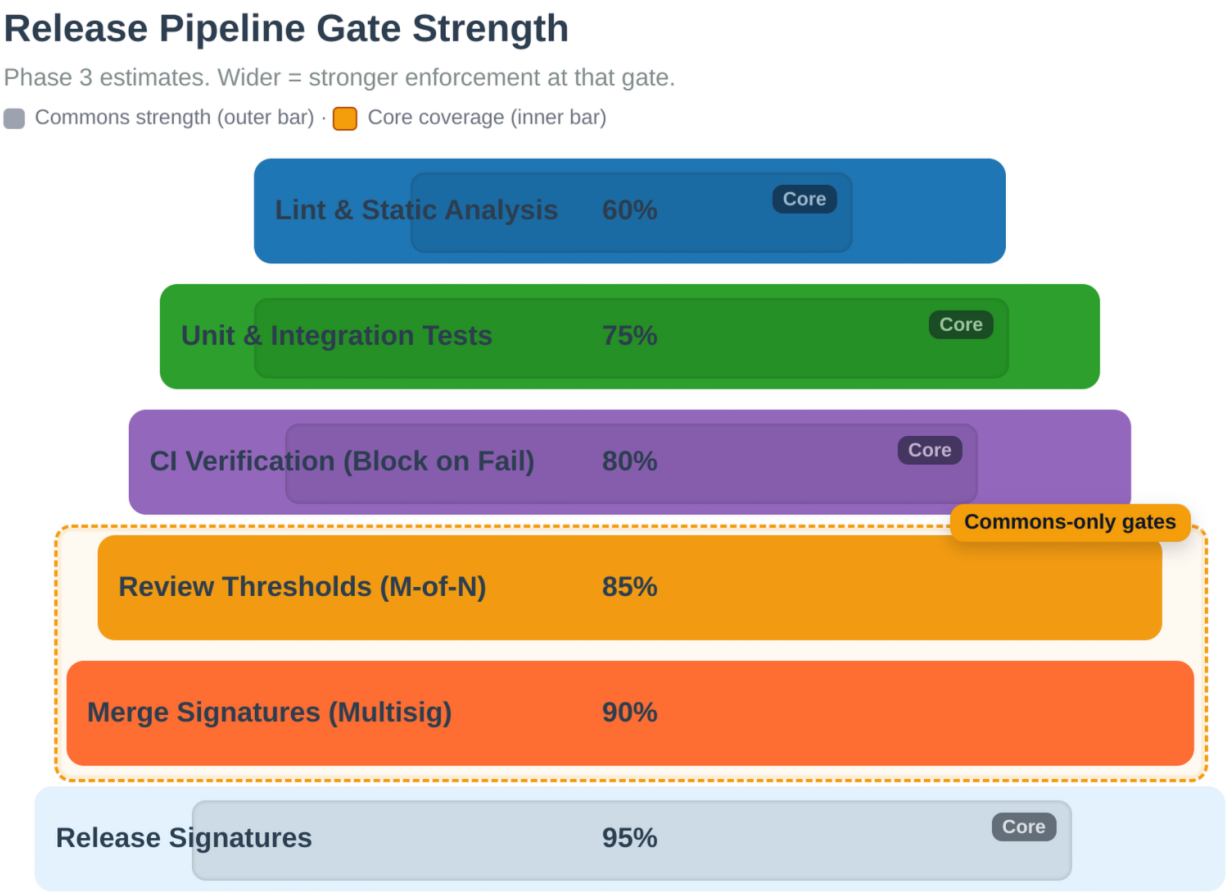

Figure: Gate strength across the release pipeline. Each gate enforces appropriate signature requirements and review periods.

Figure: Pull request review time distribution. Long tails reveal why throughput stalls without process and tooling. Automated validation reduces review bottlenecks.

7. Economic Sustainability

The Funding Gap

Only $8.4 million from 13 organizations supported Bitcoin Core development in 2023, while the network reached a $2 trillion market cap (Hough, 2025). This 0.00042% funding-to-market-cap ratio creates systemic vulnerabilities and limits Bitcoin's ability to scale safely.

Merge Mining Model

Merge mining addresses this funding gap by creating sustainable revenue that scales with usage. Merge mining allows miners to mine multiple chains simultaneously. When mining Bitcoin, they can also mine secondary chains (RSK, DATUM, Namecoin) without additional computational work. Secondary chain rewards flow through Commons infrastructure, with 1% fee funding development.

Revenue Allocation

- **60% Core Development**: Base node and critical modules
- **25% Module Developer Grants**: Incentivizes quality modules
- **10% Security Audits**: Ensures quality and safety
- **5% Operations**: Infrastructure and maintenance

Self-Sustaining Benefits

- No reliance on donations, grants, or VC funding

- Revenue scales with actual usage and miner adoption
- Economic leverage enables rule enforcement without consensus changes
- Miner alignment creates supporting constituency

Stratum V2 Merge Mining Coordination

Merge mining coordination uses Stratum V2, a modern protocol that aligns with Commons governance principles: - **Miners Control Transaction Selection**: Job negotiation decentralizes power - **Encrypted Communication**: Reduces risk of hashrate hijacking - **Efficient Binary Protocol**: Reduces bandwidth by roughly 50-66 percent - **Multiplexed Channels**: Enable merge mining coordination naturally

Revenue Scaling Examples

Calculations: - If merge-minable chains generate 100 BTC/year in rewards, 1% fee yields ~1 BTC/year for development - At 10 merged coins: ~10 BTC/year revenue - At 100 coins: ~100 BTC/year revenue - Revenue scales with adoption without requiring user payments

Infrastructure Costs: - Server costs: $75-200/month for servers, VPN, and tooling - Annual costs: under ~$30K including security audits - Model aims to protect substantial Bitcoin value at low overhead

Economic Model Charts

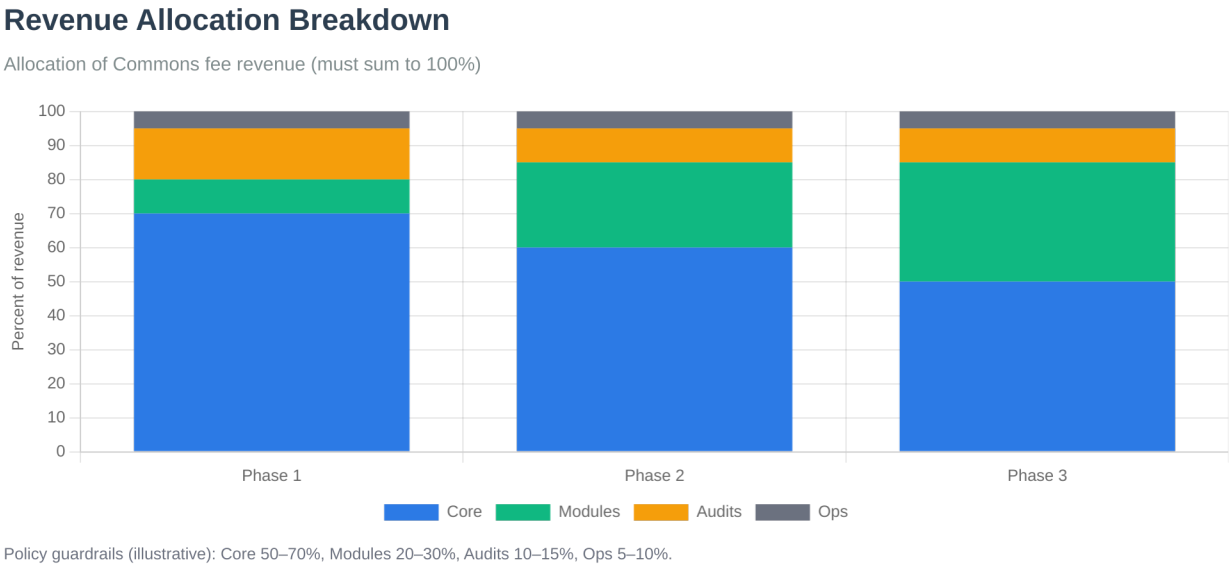

Figure: How funds are allocated across core development (60%), modules (25%), audits (10%), and operations (5%).

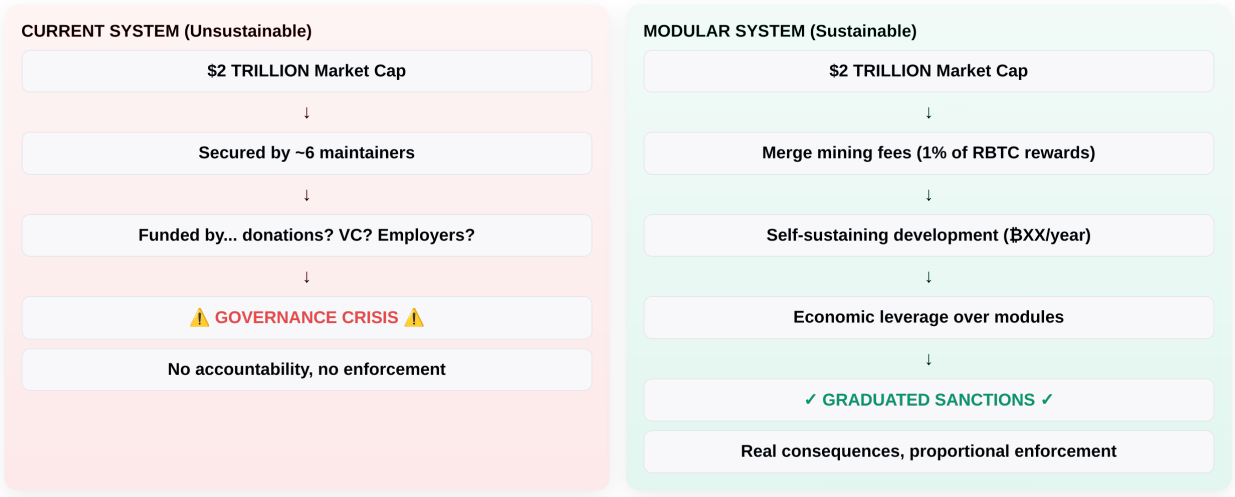

Figure: Why incentives align for miners, developers, and users. Merge mining revenue creates supporting constituency.

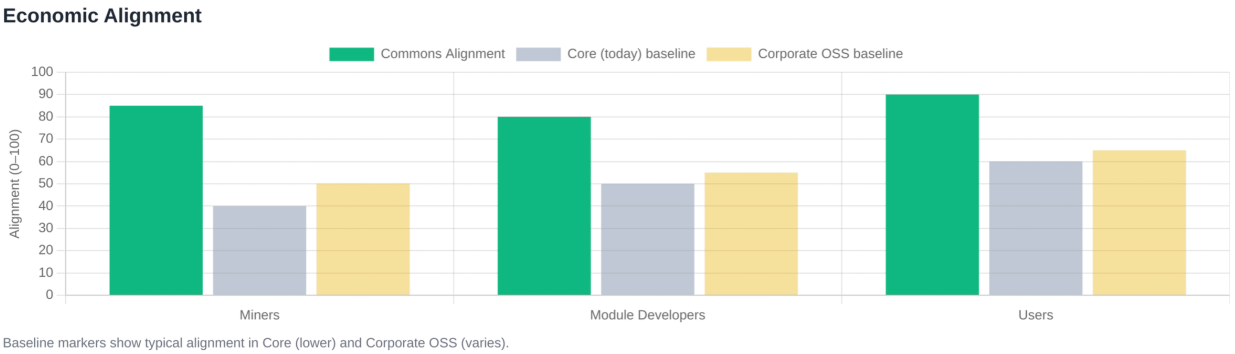

Figure: Economic alignment showing incentives for miners, developers, and users via merge mining revenue and grants.

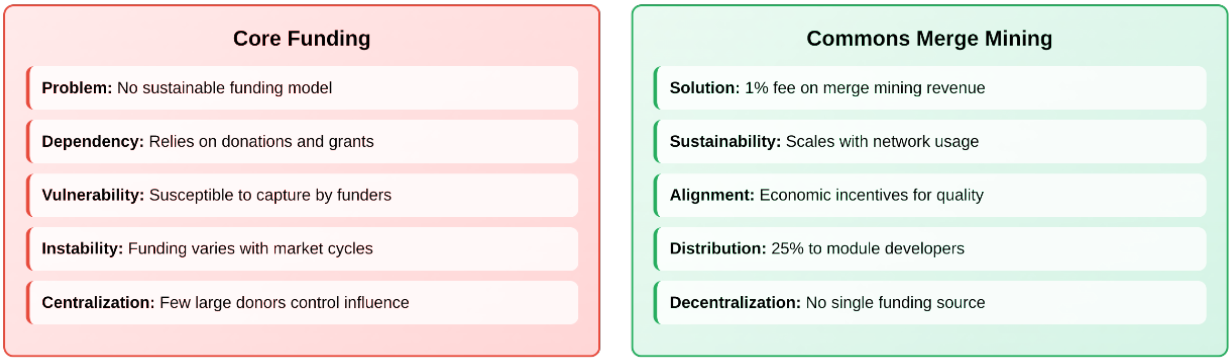

Figure: Funding model comparison: Core's donation-dependent model vs Commons' self-sustaining merge mining revenue that scales with usage.

Economic Scaling Trajectory

$10k RSK → $1M/10 Chains - Projected economic value generation across deployment phases.

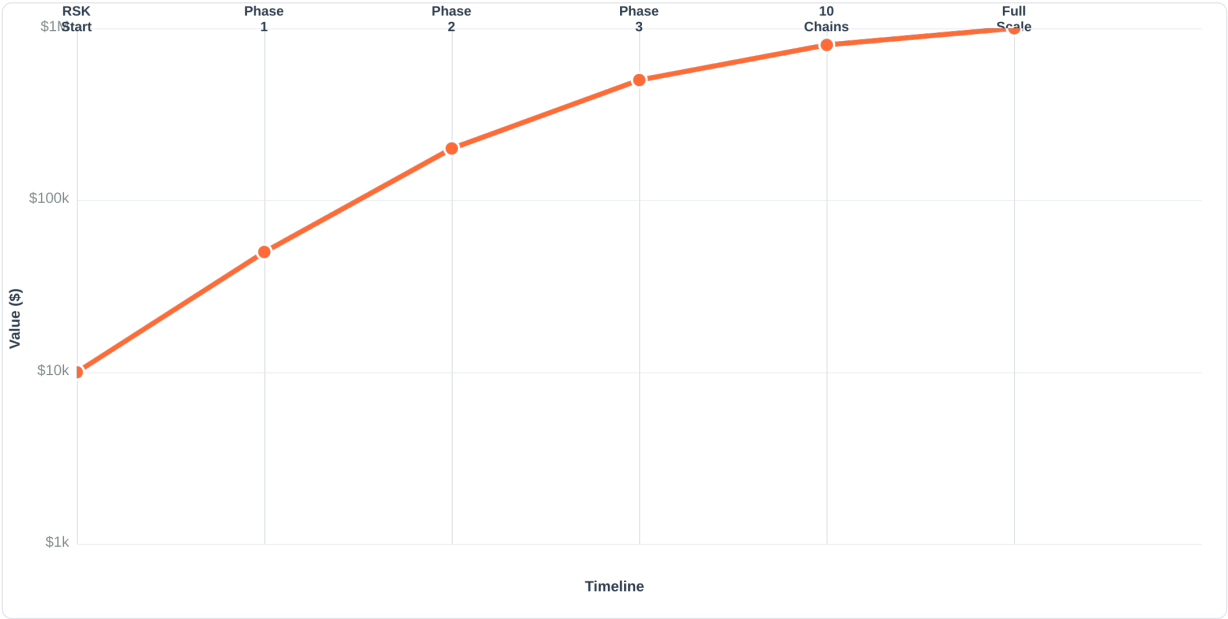

Figure: Economic scaling across development phases. Revenue scales with adoption and miner participation.

Revenue Model Sensitivity Analysis

Annual revenue from merge mining (1% of rewards) and marketplace (15-30% of module sales)

	10 chains	25 chains	50 chains	100 chains
5% adoption	$75k 0.015 BTC	$190k 0.04 BTC	$375k 0.075 BTC	$750k 0.15 BTC
10% adoption	$150k 0.03 BTC	$375k 0.075 BTC	$750k 0.15 BTC	$1.5M 0.30 BTC
20% adoption	$300k 0.06 BTC	$750k 0.15 BTC	$1.5M 0.30 BTC	$3M 0.60 BTC
50% adoption	$750k 0.15 BTC	$1.9M 0.38 BTC	$3.75M 0.75 BTC	$7.5M 1.50 BTC

Figure: Revenue model sensitivity analysis showing how revenue scales with chains adopting Commons and Commons adoption (network effects).

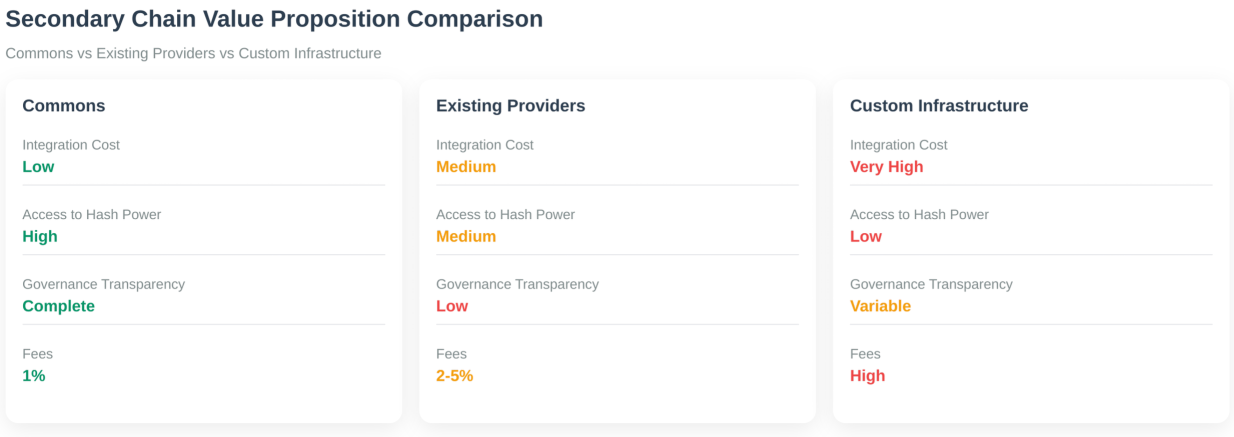

Figure: Secondary chain value proposition comparison. Commons offers reduced integration cost, access to Bitcoin's hash power, governance transparency, and lower fees (1% vs building infrastructure).

Figure: Miner sensitivity to merge-mined yields. Support persists across ranges due to direct economic incentives.

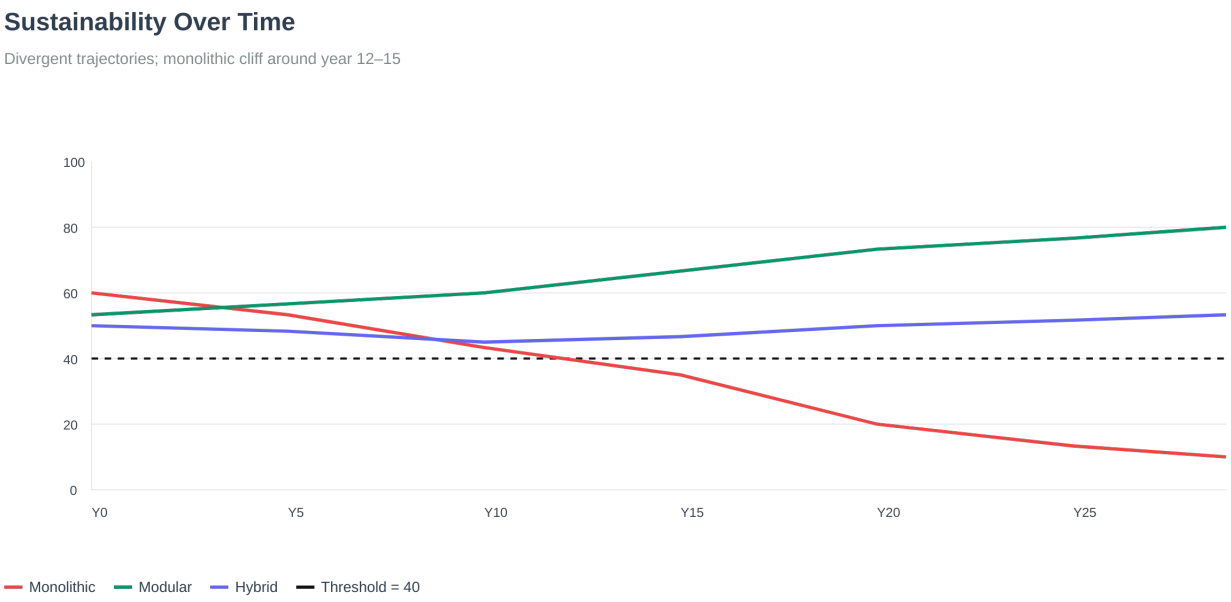

Figure: Sustainability over time: modular governance aims to sustain change while reducing capture risks compared to monolithic approaches.

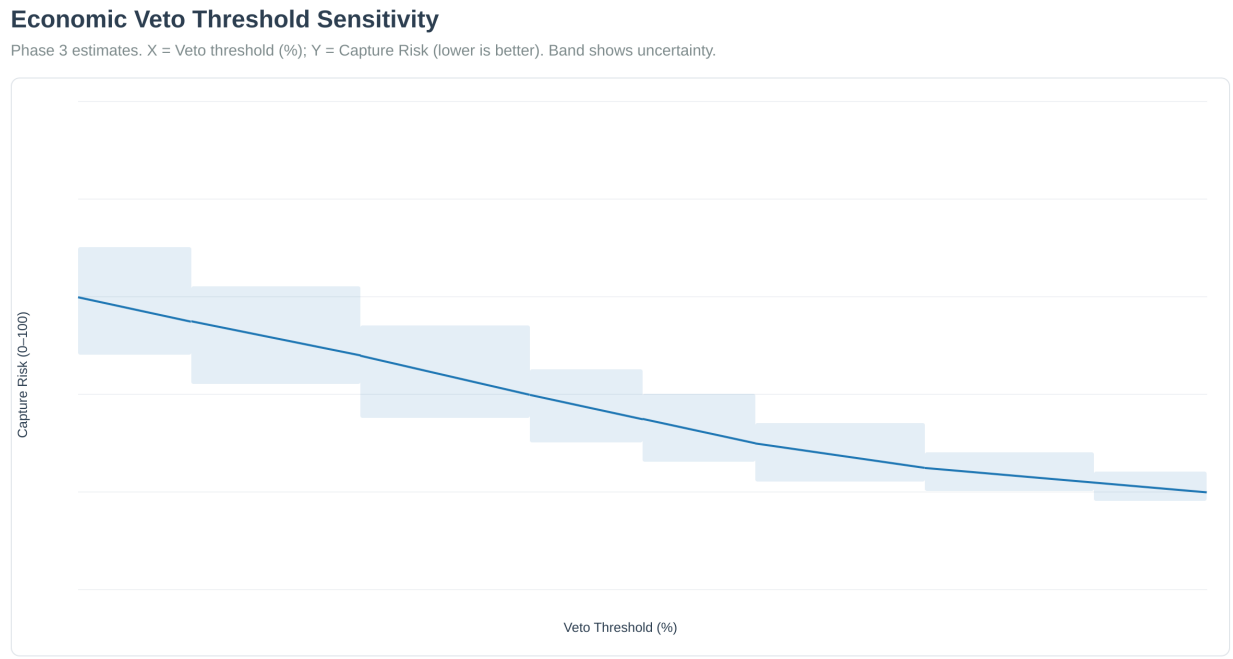

Figure: Economic veto thresholds and aligned incentives. Revenue allocation enables graduated sanctions without consensus changes.

Why Secondary Chains Choose Commons

Secondary chains need merge mining infrastructure. Commons value proposition: - **Reduced Integration Cost**: 1% fee cheaper than building infrastructure - **Access to Bitcoin's Hash Power**:

Leverage Bitcoin network effects - **Governance Transparency**: Cryptographic audit trails - **Proven Infrastructure**: Lower risk than building from scratch

Target Adoption Strategy: Target existing merge-mined chains (RSK, Namecoin, DATUM) with migration tools. Demonstrate economic benefits: reduced costs, improved governance, better security.

Fallback if Secondary Chains Don't Adopt: Phase 1 can proceed without full revenue. Alternatives include module fees, grants, donations. Long-term network effects accelerate adoption.

Success Metrics

- **Level 1 (Sustainability)**: 1000+ nodes, 20+ miners, revenue-positive operation
- **Level 2 (Ecosystem Health)**: 3+ implementations with >15% combined node share

Success Level 1 proves sustainability. Success Level 2 proves the mission: implementation diversity becomes normal. We succeed when others copy the approach, not when we dominate the market.

8. Failure Modes & Mitigations

Governance Capture

Risk: Keyholder collusion or compromise **Mitigation**: Multi-jurisdictional keyholders, transparent operation, fork-ready design. Current system easier to capture (target individuals privately, invisible control).

Regulatory Pressure

Risk: Authorities pressure keyholders to implement backdoors **Mitigation**: Distributed keyholders across jurisdictions (no single jurisdiction can compel 3-of-5 threshold), visible capture attempts, modular containment

Technical Risks

Risk: Module consensus bugs, complexity explosion **Mitigation**: Module isolation (failures cannot affect consensus), formal verification, security audits

Social Risks

Risk: Community rejection, fork wars, reputation attacks **Mitigation**: Focus on substance, build alternatives, let market decide; not asking permission, let code speak, coalition provides proof

Ultimate Protection

Governance forks provide the ultimate accountability mechanism (see Section 5 for details).

9. Implementation Status

Seven Repositories

All repositories are public and active at https://github.com/BTCDecoded:

1. **Orange Paper**: Mathematical specification of Bitcoin consensus
2. **Protocol Engine**: Core protocol logic and state management
3. **Consensus Proof**: Formal verification of consensus rules
4. **Reference Node**: Complete Bitcoin implementation
5. **Developer SDK**: Governance primitives and composition framework
6. **Governance**: Configuration repository for governance rules
7. **Governance App**: GitHub App that enforces governance rules

Current State

Phase 1 infrastructure provides substantial code implementing core capabilities. The system includes mathematical foundation and clean architecture. Governance infrastructure enables cryptographic enforcement.

Recent Technical Implementations

The reference node implementation includes extensive Bitcoin protocol support:

BIP Implementations: Block filtering (BIP157/158), compact block relay (BIP152), hardware wallet support via PSBT (BIP174), Bech32m address encoding (BIP350/351), hierarchical deterministic wallets (BIP32/39/44), and Bitcoin URI scheme with OS-level registration (BIP21).

Consistent Networking: Transport abstraction layer supporting both TCP and Iroh QUIC transports, with unified message routing across transport types. This enables nodes to choose transport based on network conditions while maintaining protocol compatibility.

Network Optimizations: Integrated coordination between compact blocks and block filtering for bandwidth efficiency. UTXO commitments support optional inclusion of block filters in responses. Transport-aware feature negotiation optimizes protocol usage based on available transports.

Advanced Networking: Package relay (BIP331) and privacy-preserving transaction relay options provide additional network efficiency and privacy capabilities.

Module System Architecture: Process-isolated module system with IPC communication, sandboxing, security validation, and module registry. Enables optional features (Lightning, merge mining, privacy enhancements) without affecting consensus or base node stability.

Stratum V2 + Merge Mining: Stratum V2 implementation with merge mining coordination for secondary chains (RSK, Namecoin, etc.). Multiplexed QUIC channels enable simultaneous mining of Bitcoin and secondary chains.

Development Roadmap

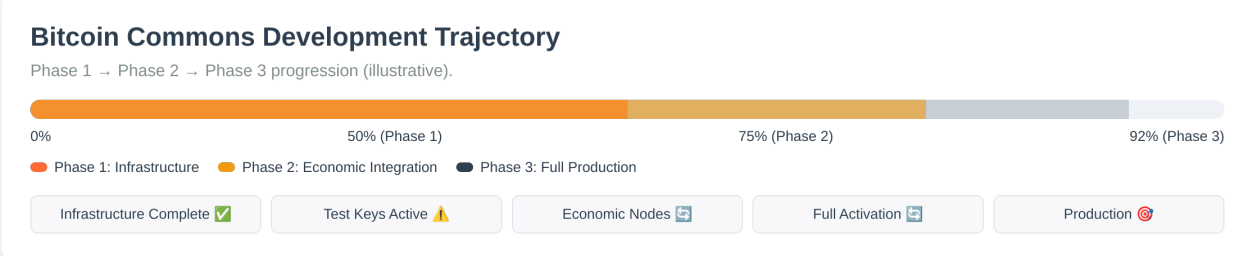

Figure: Development trajectory across phases showing progression from foundation to maturity.

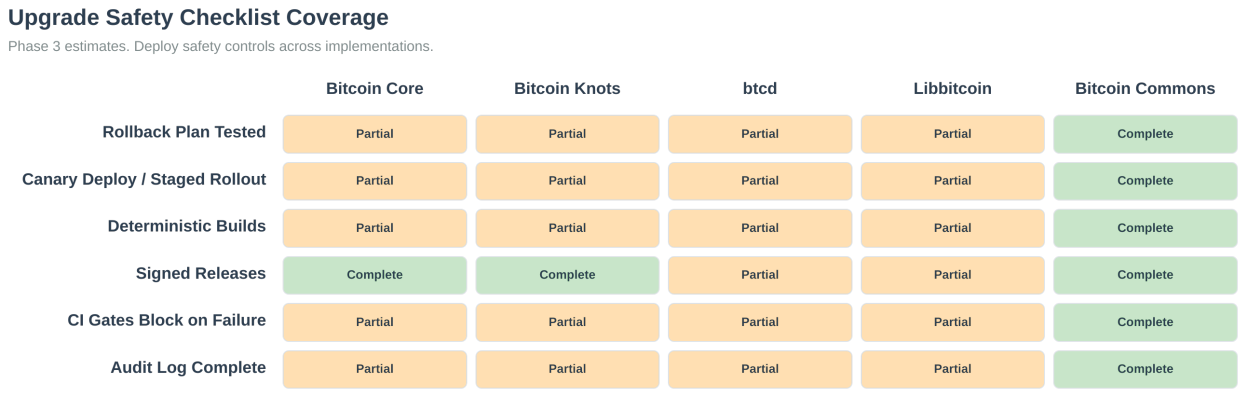

Figure: Upgrade safety checklist before activation. Prerequisites must be met before governance enforcement begins.

Phase 1 complete. Phase 2 activation requires meeting prerequisites below. Success metrics: Level 1 (sustainability) and Level 2 (ecosystem health through implementation diversity). Goal: create foundation for competing implementations, not replace Bitcoin Core.

Phase 1: Foundation

Phase 1 (Foundation) - Complete. See Section 9 for current capabilities and repositories.

Phase 2: Governance Activation

Prerequisites (Must be met before activation): - Comprehensive security audit by independent firm - Public community validation period completed - Production key management procedures operational - Formal verification of critical consensus paths complete - Legal review across multiple jurisdictions - Miner commitment threshold reached (at least one major miner) - No critical issues outstanding from Phase 1 review

Phase 2 Milestones:

Working Base Node: Complete Reference Node implementation with full network compatibility (mainnet, testnet, regtest). Milestone: At least one major miner committed to merge mining model

Module System Architecture: Module API, loading system, and infrastructure. Milestone: Lightning module integration and module marketplace operational

Cryptographic Governance: Multisig infrastructure, distributed keyholder system, transparent processes, Governance App deployment. Milestone: Governance system is operational with full three-layer verification

Lightning Integration Module: Build Lightning Network module demonstrating architecture-based conflict resolution. Milestone: Lightning module is working and adopted

Merge Mining Support: Stratum V2 infrastructure and merge mining coordination. Milestone: First revenue collection from merge mining fees (requires miner adoption)

Module Marketplace: Build distribution infrastructure with quality control, security audits, and adoption metrics. Milestone: Module marketplace is operational

Revenue-Positive Operation: Achieve sustainable funding through merge mining, demonstrate economic model viability. Milestone: 1000+ node operators, revenue-positive operation (Level 1 success)

Figure: Sustainability and ecosystem health indicators across phases. Tracks node adoption, miner participation, and revenue generation.

Phase 3: Maturity

Advanced Modules: Build privacy enhancement, alternative mempool policy, and smart contract integration modules. Milestone: 50+ available modules

Interoperability: - Fedimint integration demonstrating infrastructure positioning - Shared Iroh networking and LDK Lightning components enable natural interoperability - Commons as infrastructure layer enabling other projects

Self-Sustaining Development: Achieve complete independence from external funding; demonstrate sustainable economic model; show governance system can operate without founder. Milestone: Self-sustaining without external funding

Economic Leverage: Demonstrate economic leverage over contained ecosystems and secondary chains; show how rules can be enforced through economic pressure; prove governance system effectiveness

Production Deployment: Full mainnet governance infrastructure; first multisig merge, OpenTimestamps anchor, public monitoring operational; key rotation completed. Milestone: 10,000+ node operators, recognized as viable alternative

Recognition as Viable Alternative: Gain recognition from Bitcoin community; demonstrate technical superiority and governance advantages. Milestone: Accepted as legitimate Bitcoin implementation

Phase 4: Ecosystem Normalization

Reference Implementation: Become reference implementation for modular architecture; set standards and influence Bitcoin development ecosystem; enable multiple implementations using Commons SDK. Demonstrate governance system scalability.

Implementation Diversity Normalized: Make multiple implementations normal in Bitcoin; show Core is one option among many. Milestone: Implementation diversity normalized (Level 2 success)

Governance Model Adoption: Have governance model adopted by other projects; show governance principles are universal. Milestone: Governance model adopted by other projects

Strategic Positioning

Commons positions as infrastructure for multiple implementations, not a Core replacement. Success measured by ecosystem health and implementation diversity (Level 2 success), not market share. BitMEX validated Type 3 software forks; Commons adds specification, governance, and economics. Success when others build on the foundation, measured by ecosystem adoption.

Key Metrics

Key metrics align with Success Levels 1 and 2 (see Section 7.5). Categories include:

Technical Metrics: Network compatibility, module adoption, revenue generation, user adoption

Governance Metrics: Decision transparency, economic alignment, anti-capture measures, sustainability

Ecosystem Metrics: Diverse implementations, module marketplace growth, developer adoption, community recognition

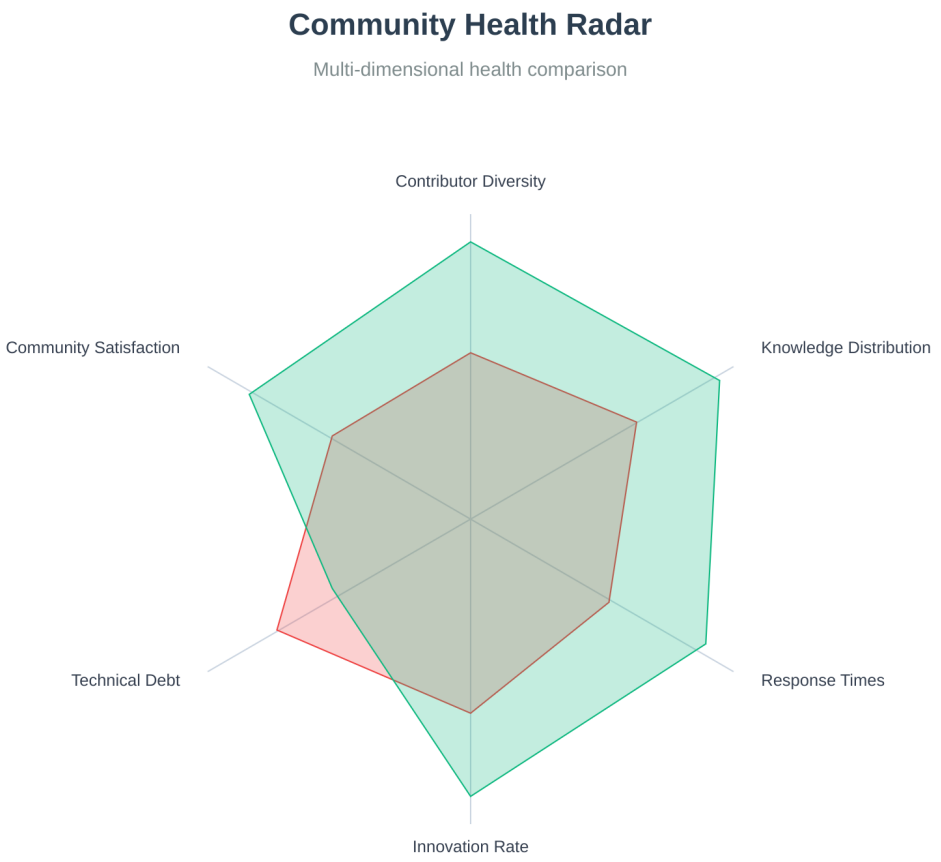

Figure: Community health radar tracks breadth of participation, contributor retention, and review responsiveness across releases.

These metrics measure the health of the ecosystem, not just the success of Commons itself. For detailed risk analysis and mitigation strategies, see Section 8 (Failure Modes & Mitigations).

10. Conclusion

Bitcoin's governance vacuum represents its greatest vulnerability at multi-trillion dollar scale. The technical architecture is bulletproof, but the social architecture runs on gentleman's agreements. BLLVM and Commons provide concrete, implementable solutions: BLLVM ensures mathematical rigor; Commons applies Ostrom's principles, Hayek's competitive discovery, and Bitcoin's cryptographic enforcement to governance.

This isn't speculation. It's applying battle-tested principles from economics, social science, and cryptography to governance. Each framework addresses weaknesses in the others: cryptography makes Ostrom enforceable at scale, infrastructure enables Hayek's competition, and modularity plus fork-ability creates competitive discovery.

The foundation exists in public repositories, but implementation remains ongoing. The architecture is designed and the path is clear: the project's future depends on community participation.

The choice: decentralize the builders, or watch them become kings.

References

Academic Sources

- Ostrom, E. (1990). *Governing the Commons: The Evolution of Institutions for Collective Action*. Cambridge University Press.
- Hardin, G. (1968). The Tragedy of the Commons. *Science*, 162(3859), 1243-1248.
- Hayek, F. A. (1945). The Use of Knowledge in Society. *American Economic Review*, 35(4), 519-530.
- De Filippi, P., & Loveluck, B. (2016). The Invisible Politics of Bitcoin: Governance Crisis of a Decentralized Infrastructure. *Internet Policy Review*, 5(3).
- Walch, A. (2015). The Bitcoin Blockchain as Financial Market Infrastructure: A Consideration of Operational Risk. *Fordham Law Review*, 84(1), 1-58.
- Walch, A. (2017). The Path of the Blockchain Lexicon (and the Law). *Vermont Law Review*, 42(1), 1-30.
- Walch, A. (2019). Deconstructing 'Decentralization': Exploring the Core Claim of Crypto Systems. *Cryptoassets: Legal, Regulatory, and Monetary Perspectives*, 55-78.
- Hough, E. (2025). "Funding the Peer-to-Peer Ethos: A Network Analysis Proposal of Bitcoin's Technical & Social Collectives." Unpublished research proposal, INFO 4360: Communication Networks and Social Capital, Cornell University.
- Hough, E. (2022). "Evaluating The Diffusion & Adoption of Bitcoin: Through Network Effects, Public Sentiment & Self-Fulfilling Expectations Equilibrium." INFO 2040: Networks, Cornell University. Cornell Blogs - https://blogs.cornell.edu/info2040/2022/11/11/evaluating-the-diffusion-adoption-of-bitcoin-through-network-effects-public-sentiment-self-fulfilling-expectations-equil/
- Uzzi, B. (1997). Social Structure and Competition in Interfirm Networks: The Paradox of Embeddedness. *Administrative Science Quarterly*, 42(1), 35-67. https://doi.org/10.2307/2393808

Historical Sources

- Andresen, G. (2014). Bitcoin: The Future of Money? Princeton University, March 27, 2014.
- Hearn, M. (2016). The Resolution of the Bitcoin Experiment. *Medium*, January 15, 2016.
- BitMEX Research (2020). Bitcoin Core's Competition. *BitMEX Research*, January 2020.

Technical Sources

- CVE-2018-17144 (2018). Bitcoin Core Denial of Service Vulnerability. *CVE Details*. https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2018-17144
- Bitcoin Core Statistics. Public GitHub repository data.
- Bitcoin Optech Topics: High quality technical primers and references
  - Merged mining: https://bitcoinops.org/en/topics/merged-mining/
  - Stratum v2: https://bitcoinops.org/en/topics/stratum-v2/

- BOLT (Lightning) Specifications: https://github.com/lightning/bolts

Repository Links

- https://github.com/BTCDecoded/the-orange-paper
- https://github.com/BTCDecoded/protocol-engine
- https://github.com/BTCDecoded/consensus-proof
- https://github.com/BTCDecoded/reference-node
- https://github.com/BTCDecoded/developer-sdk
- https://github.com/BTCDecoded/governance
- https://github.com/BTCDecoded/governance-app