BITCOIN GOVERNANCE FRAMEWORK

VOL. 01 · 2026

Consensus Change Standards

A Legal and Technical Framework for Bitcoin Protocol Governance

Asaf Fulks

California State Bar No. 343622

Solo Bitcoin Miner · Full Node Operator

THE FORUM PRESS

a Fulks, Inc. company

Second Edition · May 2026

asaffulkslaw.com

CONSENSUS CHANGE STANDARDS

A Legal and Technical Framework for Bitcoin Protocol Governance

Second Edition — May 2026

WRITTEN BY

ASAF FULKS, J.D.

California State Bar No. 343622

Admitted, U.S. District Court, Central District of California

asaffulkslaw.com

THE FORUM PRESS

a Fulks, Inc. company | California

theforumpress.com

CONSENSUS CHANGE STANDARDS

A Legal and Technical Framework for Bitcoin Protocol Governance

Second Edition, May 2026
First Edition, April 2026 (revised May 2026)
Published by The Forum Press, a Fulks, Inc. company
California | `theforumpress.com`

Available at: `asaffulkslaw.com`

Cite as: Asaf Fulks, *Consensus Change Standards: A Legal and Technical Framework for Bitcoin Protocol Governance* (2d ed. 2026), `asaffulkslaw.com`.

Notice and Disclaimer. This document is provided for informational and educational purposes only. It does not constitute legal advice and is not a solicitation for legal services or attorney advertising. The legal analysis herein draws primarily on California state law and federal authority; readers in other jurisdictions should not assume it applies to their circumstances. Nothing in this document creates an attorney-client relationship between the author and any reader. No warranty is made as to the accuracy, completeness, or current applicability of the content; readers should not rely on it as a substitute for individualized professional advice and should consult qualified legal counsel licensed in their jurisdiction regarding any specific question. The views expressed are the author's own and do not represent the views of any employer, client, or affiliated organization. This document does not recommend for or against any particular consensus change proposal, including BIP-110.

Acknowledgments

Thanks to the operators, miners, and users who keep this network running. To the dedicated volunteers who refuse to cut corners and build Bitcoin — line by line, meetup by meetup, and argument by argument. To Murch and Jameson Lopp, for taking time with an early draft. And to Ren Crypto Fish, Steve Lee, and Lyn Alden, whose *Analyzing Bitcoin Consensus* mapped the descriptive terrain this paper's normative standards depend on.

—Asaf

Contents

Abstract

Bitcoin has no formal process for evaluating proposed changes to its consensus rules. The Bitcoin Improvement Proposal (BIP) system provides a mechanism for proposing changes, but establishes no minimum standards that a proposal must meet before the community considers activation. There are no required review periods, no mandatory code audit standards, no agreed-upon activation thresholds, no chain split risk assessment methodology, and no framework for evaluating the legal and economic consequences of a failed activation.

This absence of standards has produced predictable results. The block size wars of 2015–2017 consumed years of developer time, fractured the community, and produced a contentious hard fork. The SegWit activation depended on the credible threat of a novel User Activated Soft Fork (UASF) — BIP-148 — to break the deadlock that had persisted after BIP-9 signaling failed to reach its 95% threshold over the preceding eight months. More recently, BIP-110 — a proposed temporary soft fork to restrict arbitrary data in Bitcoin transactions — was released with a buggy activation client, a 55% activation threshold dramatically below historical precedent, and a six-week timeline from initial proposal to activation client that compressed by an order of magnitude the review periods of every modern successful soft fork. The activation client was first distributed alongside stable Knots releases on node management platforms with no risk disclosure or visual differentiation, then bundled into the default release stream itself — structuring the routine upgrade ladder to terminate in a confirmation dialog presenting consensus alteration as the path forward.

This paper proposes a comprehensive framework for evaluating Bitcoin consensus change proposals. It draws on the history of Bitcoin's prior consensus changes, established principles of software engineering governance, and legal analysis of the liabilities created by reckless activation. The framework is designed to be practical, concrete, and immediately applicable. It is not a BIP. It does not propose changes to Bitcoin's code. It proposes standards for the process by which such changes should be evaluated, debated, and either adopted or rejected.

The author is a practicing litigator, solo Bitcoin miner, full node operator, and computer scientist. This framework is written from the intersection of those disciplines because the problems it addresses — governance, liability, technical risk, and economic consequence — cannot be adequately analyzed from any single perspective.

Chapter 1

The Problem

1.0 How to Read This Paper

The framework proposed here is not an attempt to centralize authority over Bitcoin consensus. It is an attempt to give Bitcoin's existing decentralized governance process a shared vocabulary for evaluating proposals.

The framework evaluates the ecosystem behavior surrounding a consensus-change proposal — the activation clients built around it, the review process applied to its code, the support evidence assembled in its favor, and the operational coordination preceding any deployment. It does not evaluate, and cannot evaluate, the editorial status of any BIP document. A BIP is a proposal; this framework asks which proposals merit activation, not which proposals merit filing.

Readers concerned with the question of authority itself — who decides? what makes one threshold "wrong" and another "right"? — are encouraged to read §6.2 and §6.4 before §3. The framework is a tool, not a rule; if that distinction does not hold for a given reader, the standards that follow will read as something they are not.

Different sections serve different audiences. Protocol developers will find §3 and §5 most directly applicable. Businesses, exchanges, and custodians will find §3.5 and §4 most operationally relevant. Node operators and users will find §6 most useful for evaluating what the framework asks of them and what it does not.

1.1 The Absence of Standards

Bitcoin's consensus rules are the most consequential code in the financial world. They govern the creation, transfer, and validation of an asset with a market capitalization exceeding one trillion dollars. Changes to these rules affect every participant in the network: miners who invest capital in hardware, node operators who validate transactions, developers who build applications, businesses that accept payment, and individuals who store wealth.

Despite these stakes, there is no formal standard governing how changes to consensus rules should be proposed, evaluated, reviewed, tested, activated, or — critically — rolled back if they fail. The BIP process, established in BIP-1 and refined in BIP-2, provides a template for writing proposals and a loose taxonomy of proposal types. It does not establish minimum

standards for activation safety, mandatory review periods, code quality requirements, or chain split risk assessment.

The result is an ad hoc system in which each consensus change proposal invents its own activation mechanism, sets its own threshold, defines its own timeline, and is evaluated by the community with no consistent framework. Some proposals receive years of careful review. Others are pushed to activation within weeks. The difference between these outcomes is determined not by any institutional process but by the personalities, politics, and persuasive abilities of the participants.

1.2 The Inscription Era and the Wave of Restrictive Proposals

In late 2022, the developer Casey Rodarmor released the Ordinals protocol — a method for encoding arbitrary data into Bitcoin transactions by using Taproot's tapscript capacity. The protocol enabled what came to be called "inscriptions": image, text, document, and binary files embedded permanently within the witness data of Bitcoin transactions. Following the protocol's December 2022 mainnet debut, inscription activity grew through 2023 to occupy a substantial share of block space across multiple weeks of high demand.

The mechanism is technically straightforward but governance-consequentially significant. Inscriptions place arbitrary bytes within Taproot's script-path spend by enclosing them in an unexecuted `OP_FALSE OP_IF ...OP_ENDIF` envelope. Bitcoin nodes do not validate or execute the data within the envelope; they store it permanently in the witness portion of the transaction. Taproot's witness-data fee discount makes this storage method substantially cheaper, per byte, than alternatives such as OP_RETURN. Taproot (BIP-341), activated in November 2021, expanded witness capacity to support Schnorr signatures, MAST, and future protocol upgrades. The expanded capacity also supported a use case its designers did not anticipate.

The inscription debate became, by mid-2023, the most contentious sustained policy disagreement in Bitcoin since the block size wars, and it has continued through 2026. Two camps emerged. Proponents argued that inscriptions pay full fees, displace no monetary transactions in equilibrium, and represent a legitimate use of the open block space markets have priced. Opponents argued that inscriptions consume scarce block space and witness storage with non-monetary content, raise long-term storage and bandwidth costs for every full-node operator, and degrade Bitcoin's function as monetary infrastructure. Both arguments draw on real considerations; neither is dispositive. Reasonable participants have reached opposite conclusions and continue to debate the question in good faith.

This paper does not evaluate the substantive merits of that debate. The question of what counts as a legitimate use of Bitcoin's base layer is for the network's stakeholders to resolve through the iterated coordination processes *BCAP* describes — not for any single framework, including this one, to settle. What this paper does evaluate is the procedural conduct surrounding proposals that would resolve the question by consensus change. Beginning in

2023, multiple proposals emerged seeking to restrict the methods that enable inscriptions: tightening data-embedding limits, restricting tapscript witness contents, or constraining specific transaction patterns. BIP-110 — the Reduced Data Temporary Softfork analyzed in §1.3 — represents the most-developed instance of this wave and the one whose activation client reached release.

The framework presented in this paper is procedurally neutral on the underlying disagreement. The same evaluation applies whether a proposal would liberalize Bitcoin's policy space or restrict it. The same standards apply whether the proponents are inscription advocates or inscription critics. A consensus change advanced through inadequate review, low activation thresholds, and conflict-of-interest dynamics raises the same governance and legal exposure regardless of which side of the substantive debate it sits on. The framework asks one question of each proposal: is it ready, by the standards of Section 3 and Section 5, for the community to engage with its activation? That question is independent of the question whether the underlying change is desirable on the merits.

1.3 BIP-110 as Case Study

BIP-110 — the Reduced Data Temporary Softfork — illustrates every failure mode that a governance framework should prevent. Originally proposed as BIP-444 in late October 2025, the proposal sought to restrict methods of embedding arbitrary data in Bitcoin transactions. Its stated goal was to protect Bitcoin's function as monetary infrastructure by limiting what proponents characterized as "spam" uses of block space.

The proposal's technical merits are debatable. Reasonable people disagree about whether inscriptions, ordinals, and large OP_RETURN payloads represent legitimate uses of Bitcoin's base layer or parasitic exploitation of shared infrastructure. This paper takes no position on that question. The problems with BIP-110 are procedural, not substantive:

A. Activation threshold. BIP-110 specified a 55% miner signaling threshold for a User Activated Soft Fork. More precisely, BIP-110 is a LOT=true mechanism: an early lock-in trigger at 55% signaling and a mandatory lock-in window around August 2026 in which BIP-110-enforcing nodes reject all non-signaling blocks, with activation at `max_activation_height` block 965,664 (approximately September 2026). The framework's position on LOT=true is set out in §3.4. This is dramatically lower than historical precedent. SegWit's BIP-9 deployment required 95% miner signaling; when that stalled, BIP-91 created a parallel mechanism at 80%, and the BIP-148 UASF threatened to reject non-signaling blocks entirely. Taproot activated at 90% via Speedy Trial. A 55% signaling threshold provides no assurance that the share of hashrate enforcing the new rules at activation will exceed 55%; the remainder may continue producing blocks valid under the legacy rules but invalid under the new ones. Signaling at lock-in is not equivalent to enforcement at activation, and the divergence between the two is precisely the mechanism by which low-threshold soft forks produce persistent minority chains. This is not a theoretical risk — it is a recipe for a chain split.

B. Code quality. The activation client, released in late 2025 as a fork of Bitcoin Knots, was found to contain significant bugs. Multiple Bitcoin developers reported that the client could not reliably fork the network, and that users running the code might accidentally fork themselves off both chains. Public commentary from reviewers raised concerns about the code's structure and quality; whatever tools or methods were used to produce it, the activation client did not receive the level of independent review that consensus-critical software demands. An activation client for a consensus change affecting a trillion-dollar network was released without the review burden customary for consensus-critical software.

C. Review period. From initial proposal to release of the activation client, BIP-110 moved through the pipeline in approximately six weeks — from the initial bitcoin-dev mailing list post on 26 October 2025 to release of the first activation client (v0.1rc1) on 10 December 2025. By comparison, SegWit was proposed in December 2015 and did not activate until August 2017 — a twenty-month process. Taproot was first proposed in January 2018 and activated in November 2021 — nearly four years. Six weeks is not a review period. It is a rush to deployment.

D. Activation client distribution. On at least one node management platform, the BIP-110 activation client was initially listed as a selectable version option in the same dropdown menu as stable Knots releases, with no warning label, risk disclosure, or visual differentiation. Selection was deliberate — a node operator had to affirmatively choose the BIP-110 version — but the presentation treated consensus-altering software identically to routine maintenance releases. A node operator who understood version management but not the implications of BIP-110 specifically could have activated consensus-changing code believing it was a standard update. The absence of any risk disclosure at the point of selection was the governance failure, not the availability of the option itself.

A subsequent escalation merged the activation code into the project's default release stream. As of `v29.3.knots20260508` (released 9 May 2026), the stable Bitcoin Knots release includes the BIP-110 (RDTS) activation rules and prompts the operator to confirm activation at runtime, via either a configuration directive (`consensusrules=rdts`) or a GUI dialog. The non-RDTS variant remains available — as the one-day-older `v29.3.knots20260507` build — labeled as the discouraged option. The confirmation is real; the operator must affirmatively accept the consensus change. The governance failure is now structural rather than presentational: the routine upgrade ladder, which an operator follows simply to receive ordinary bug fixes and feature improvements, terminates in a confirmation dialog that presents consensus alteration as the path forward. The release notes characterize the change as fixing "critical vulnerabilities in long-standing network design" — a framing that itself biases the confirmation step. Declining the change means foregoing the maintenance benefits of the newer release; accepting means defaulting into a consensus alteration by the path of least resistance.

E. Sunset mechanism. BIP-110 includes an automatic expiry at a defined block height, after which the new rules cease to be enforced. This is a meaningful improvement over proposals that lack any deactivation mechanism. However, there is no public evidence that the sunset

mechanism was tested on testnet to confirm that the transition back to pre-activation rules would occur without consensus failures — a concern amplified by the significant bugs found in the activation client itself. A sunset clause that has not been demonstrably tested is a promise, not a guarantee. Furthermore, the one-year duration was chosen without empirical justification for why one year is the appropriate period, and the proposal contained no defined process for evaluation at the end of the enforcement period.

The "BIP-110" addressed throughout this paper refers to the activation attempt — the RDTS client, the surrounding ecosystem campaign, and the operational coordination assembled around the proposal — not to the BIP-110 document filed in the BIPs repository. A BIP is a proposal; the framework's evaluative scope is the ecosystem behavior that would convert a proposal into network state. BIP-361 (*Post Quantum Migration and Legacy Signature Sunset*, Lopp et al. 2026), filed in February 2026, is an instructive contemporaneous example: a proposal of consequential scope and ongoing community debate, with no implementation code and no activation campaign at filing — a BIP, in the editorial sense, without any of the ecosystem facts this framework evaluates.

The question of how any single threshold could be "set" in a network with no central authority is foundational and is addressed in §6.2 and §6.4. The short answer: it cannot, in any binding sense. BIP-110's proponents proposed 55%; whether the network adopted it was determined by tens of thousands of independent operator decisions, which collectively yielded $\leq 0.15\%$ signaling (peak observed share of blocks signaling readiness during the activation window).

1.4 The Stakes

A failed consensus change activation is not a software bug that can be patched. It is a potential fracture of the monetary network. When a chain split occurs without replay protection, transactions valid on one chain may be valid on the other. Users can lose funds. Exchanges must choose which chain to list. Contracts denominated in Bitcoin become ambiguous. The economic damage is real, quantifiable, and potentially irreversible.

The most recent significant chain split without replay protection occurred in March 2013, when a database incompatibility between Bitcoin versions 0.7 and 0.8 caused a six-hour fork that included a successful double-spend attack (the technical post-mortem is BIP-50). The 2017 SegWit2x proposal came within days of producing another before being called off. The Bitcoin community has been fortunate. Fortune is not a governance strategy.

1.5 Relation to Prior Work

The framework proposed in this paper builds on, and is intended to be read alongside, the most developed prior analysis of Bitcoin's consensus-change dynamics: Ren Crypto Fish, Steve Lee & Lyn Alden, *Analyzing Bitcoin Consensus: Risks in Protocol Upgrades* (Nov. 2024)

[hereinafter *BCAP*]. *BCAP* categorizes participants in consensus changes into six stakeholder groups — Economic Nodes, Investors, Media Influencers, Miners, Protocol Developers, and Users/Application Developers — and analyzes how the relative power of each group fluctuates across the phases of a proposal's lifecycle. It introduces a "State of Mind" framework distinguishing degrees of stakeholder engagement, develops a detailed scenario analysis of consensus changes deployed through alternative clients, and concludes with thirteen evaluation questions and twelve indicators stakeholders may use to assess proposals in real time. *BCAP*'s draft was reviewed by a substantial cross-section of Bitcoin's protocol-development and analytical community.

BCAP and the present framework address the same problem from different vantages. *BCAP* is descriptive: it explains how Bitcoin consensus emerges from the iterated interactions of stakeholders with shifting powers and divergent incentives. This paper is normative: it proposes the minimum standards a proposal should meet before stakeholders rationally engage with its activation. The two works converge on several substantive points. Both treat sub-overwhelming Economic Node adoption as the central failure mode in contested soft-fork scenarios — *BCAP* through its bounty-claim and chain-split risk analysis (§3.5.2), this paper through its activation-threshold standards (§3.4) and chain-split risk-assessment requirement (§3.5). Both treat miner signaling as a necessary but insufficient indicator of community consensus. Both reject naïve majoritarianism as a consensus-determination heuristic. And both identify Economic-Node coordination as the operative remedy when an activation proceeds without genuine consensus: *BCAP* describes that coordination descriptively, as the mechanism by which markets re-equilibrate after a contested split; this paper prescribes it normatively, as the recommended response to a proposal classified as Red under §5.2.

This paper extends *BCAP*'s analytical framework in four respects. First, it operationalizes qualitative recommendations into numerical floors: minimum activation thresholds (90% MASF, with sub-80% presumptively dangerous and sub-60% reckless); minimum review periods tied to risk category (twelve months for moderate-risk soft forks, twenty-four months for high-risk soft forks, thirty-six for hard forks); and minimum testnet deployment duration (three months). Second, it quantifies the relationship between activation threshold and chain-split exposure: §3.4 models post-activation hashrate as a Bernoulli process and derives concrete reorganization probabilities — approximately thirty percent at $E = 0.55$ over a six-block horizon, falling to roughly 2×10^{-8} at $E = 0.95$. This quantification is consistent with *BCAP*'s qualitative conclusion that low and high Economic Node adoption produce categorically different risk profiles, but supplies the order-of-magnitude estimates that the qualitative analysis leaves open. Third, it develops a legal-liability framework — negligence (with explicit attention to the economic-loss rule and the available routes around it), tortious interference, fiduciary duty, mining-pool contract obligations, and regulatory consequences. *BCAP* does not address legal exposure; this paper offers it as a distinct analytical layer whose conclusions are relevant to every stakeholder group *BCAP* identifies. Fourth, it consolidates the foregoing into a binary twenty-point scoring rubric with classification bands (§5), enabling structured evaluation in place of free-form weighing of considerations.

Several elements of *BCAP* the present paper does not duplicate. The stakeholder taxonomy and the power-over-time analysis (*BCAP* §§3.2, 3.3.2), the State of Mind framework (*BCAP* §3.1), the investor-segment analysis distinguishing self-custodying holders from institutional, corporate-treasury, and exchange-traded-fund segments (*BCAP* §3.2.2), and the alternative-client adoption-pathway analysis (*BCAP* §3.5.1) all remain the more developed treatments of their respective subjects. This paper refers readers to *BCAP* for those questions. The two frameworks are intended to function together: *BCAP* supplies the theory of how stakeholders shape consensus; this paper supplies the standards by which stakeholders may judge whether a particular proposal is ready for that process to begin.

The minimum-review-period floors proposed in §3.2 are anchored in empirical observation rather than aesthetic preference. Jameson Lopp's longitudinal analysis of Bitcoin Core node-software adoption documents median upgrade times measured in months — historically on the order of forty weeks for routine releases, with recent versions trending longer still. See Jameson Lopp, *When Do Bitcoin Node Operators Upgrade?*, `blog.lopp.net/when-do-bitcoin-node-operators-upgrade/`. A twelve-month minimum review period for a moderate-risk soft fork accommodates a single such upgrade cycle plus a meaningful interval for review; a twenty-four-month minimum for a high-risk soft fork accommodates two. The review-period floors proposed in this framework are thus not arbitrary durations but the durations the network's own observed upgrade dynamics require for activation signaling to begin from a position of broad enforcement readiness rather than speculative anticipation.

The legal analysis in Section 4 engages a smaller but distinct body of prior work. Angela Walch's argument that core protocol developers exercise discretionary authority over property interests sufficient to trigger fiduciary obligations is the central academic contribution to the question and is engaged directly in the fiduciary-duty analysis below. The Tulip Trading litigation (*Tulip Trading Ltd v van der Laan* [2023] EWCA Civ 83) is the most developed common-law treatment and frames the standard against which a developer's conduct would be evaluated. The other legal questions raised by chain splits — contract interpretation of "Bitcoin," tax cost-basis allocation under IRS Revenue Ruling 2019-24, exchange custodial obligations during a fork — have received episodic treatment but no consolidated analysis. The legal-analysis section of this paper is offered as a starting point, not a final synthesis.

This paper occupies the operational and legal layer of an emerging body of work on Bitcoin governance. It is most useful when read alongside *BCAP*, not in place of it.

Chapter 1 in brief

Bitcoin has a process for filing proposals (the BIP system) and no process for evaluating them. Every consensus change reinvents its own playbook for review, activation, and rollback. The block size wars, SegWit's near-miss in 2017, and the inscription-era wave of restrictive proposals culminating in BIP-110's six-week sprint to deployment are all symptoms of the same gap.

This paper proposes minimum standards a proposal should meet before stakeholders rationally engage with its activation. It is meant to be read alongside *BCAP* (Crypto Fish, Lee, Alden, 2024), which describes *how* consensus emerges from stakeholder interaction; this paper proposes *what* a proposal should look like to deserve that engagement. Different vantages on the same problem.

Chapter 2

Historical Precedent

Bitcoin has undergone numerous consensus changes since its creation in 2009. The most significant of these provide instructive precedent for establishing governance standards.

2.1 P2SH (BIP-16) — 2012

Pay-to-Script-Hash was one of Bitcoin's first contentious soft forks. Competing proposals (BIP-16 and BIP-17) divided the developer community. Activation used a simple miner signaling threshold of 55% — the same threshold later adopted by BIP-110. The activation was messy, with miners signaling inconsistently and the community uncertain about which proposal would prevail.

Lesson: Low activation thresholds produce uncertainty even when the proposal itself has technical merit. P2SH ultimately succeeded because both competing proposals were small, low-risk changes. BIP-110's use of the same threshold for a far more consequential change ignores the increased risk.

2.2 The Block Size Wars (2015–2017)

The block size debate consumed more community energy, developer time, and political capital than any other event in Bitcoin's history. Multiple proposals competed: BIP-101 (8 MB blocks), BIP-102 (2 MB blocks), Bitcoin XT, Bitcoin Classic, Bitcoin Unlimited, and ultimately SegWit2x. The conflict produced the Bitcoin Cash hard fork in August 2017 and nearly produced a second split with SegWit2x in November 2017.

The block size wars demonstrated several principles that any governance framework must account for:

Miner signaling is unreliable as a measure of community consensus. Mining pools signaled support for proposals their users did not endorse. The SegWit2x "New York Agreement" secured signatures from companies representing over 80% of hashrate, yet the proposal collapsed when it became clear that node operators and users would not follow.

Economic nodes matter more than hashrate. Exchanges, payment processors, and major holders ultimately determine which chain carries economic value. A chain with 90% of

the hashrate but no exchange listings and no merchant adoption is worthless. Governance frameworks must account for economic consensus, not merely miner signaling.

Hard forks are permanent and expensive. Bitcoin Cash continues to exist as a separate chain with a fraction of Bitcoin's value. Every hard fork fragments the ecosystem, confuses users, and creates legal ambiguity about which chain constitutes "Bitcoin" for contractual and regulatory purposes.

2.3 SegWit (BIP-141) — 2017

Segregated Witness was proposed in December 2015 and activated in August 2017 via BIP-9 version bits signaling with a 95% activation threshold. When miner signaling remained well below the 95% threshold through early 2017, the community developed BIP-148 — a User Activated Soft Fork that would have begun rejecting non-SegWit blocks on August 1, 2017, regardless of miner signaling.

The threat of BIP-148 — and the risk that it would cause a chain split — motivated miners to signal for SegWit. BIP-91 locked in on July 21, 2017, forcing miners to signal for BIP-141, and SegWit's lock-in was achieved before the August 1 UASF deadline. This episode established the UASF as a credible activation mechanism but also demonstrated its risks: had miners not capitulated, BIP-148 nodes would have split from the main chain.

Lesson: UASFs are a tool of last resort, not a standard activation mechanism. BIP-148 worked because SegWit had overwhelming community support and years of review. Applying the same mechanism to a proposal with weeks of review and marginal support — as BIP-110 attempted — is reckless.

2.4 Taproot (BIP-340/341/342) — 2021

Taproot is the gold standard for Bitcoin consensus change governance. Its design was first floated on the bitcoin-dev mailing list in January 2018 and formally specified in BIPs 340, 341, and 342 in 2020; it then underwent years of review, extensive formal analysis of its cryptographic primitives (Schnorr signatures), multiple rounds of community feedback, and a novel activation mechanism (Speedy Trial) that provided a defined three-month signaling window with a built-in timeout. The selection of Speedy Trial over BIP-8 with LOT=true and BIP-8 with LOT=false reflected an extended bitcoin-dev mailing list debate over the appropriate balance between activation speed, miner authority, and economic-node authority.

Taproot activated in November 2021 with 90% miner signaling within the Speedy Trial window. There was no chain split, no community fracture, and no economic disruption.

Lesson: Patience works. Review works. High thresholds work. Defined timelines with built-in failure modes work. Every element that made Taproot's activation successful was absent from BIP-110.

2.5 Summary of Activation Parameters

Proposal	Threshold	Review Period	Mechanism	Outcome
P2SH (2012)	55%	~3 months	Coinbase signal + flag day	Activated (messy)
SegWit (2017)	95%	20 months	BIP-9 + UASF	Activated
SegWit2x (2017)	80% (NYA)	~6 months	Hard fork	Canceled
Taproot (2021)	90%	~4 years	Speedy Trial	Activated
BIP-110 (2025–26)	55%	~6 weeks	UASF (Knots, LOT=true)	Stalled ($\leq$ 0.15%)

Note: percentages in the Outcome column for stalled or failed proposals reflect the peak observed share of blocks signaling readiness for the proposal during its activation window. BIP-110's 55% lock-in threshold was never approached.

The pattern is clear: successful consensus changes correlate with high activation thresholds, long review periods, and broad community buy-in. Failed or stalled proposals correlate with low thresholds, rushed timelines, and narrow support. This is not coincidence. It is the predictable result of governance dynamics that any framework must formalize.

Chapter 2 in brief

Bitcoin's prior consensus changes show one pattern. P2SH (2012) succeeded despite a 55% threshold because the change was small and the network was small. SegWit (2017) took twenty months and an unprecedented UASF threat to escape miner deadlock. Taproot (2021) used 90% signaling, a defined window, and years of review. The successful pattern: high threshold, long review, broad consensus. The failed and stalled patterns inverted at least one of those three.

Chapter 3

The Framework

The following framework establishes minimum standards for Bitcoin consensus change proposals. These standards are not intended to be enforced by code — Bitcoin has no central authority capable of enforcement. They are intended to serve as a publicly available benchmark against which the community can evaluate proposals. A proposal that meets these standards deserves serious consideration. A proposal that fails to meet them should be treated with appropriate skepticism.

3.1 Proposal Submission Requirements

A consensus change proposal submitted for community consideration should include, at minimum:

A. Problem Statement. A clear, specific description of the problem the proposal addresses, supported by empirical data where available. "Bitcoin should do one thing and do it well" is a philosophy, not a problem statement. "The UTXO set has grown by X% in Y months due to Z transaction type, imposing quantifiable costs of $W per node operator per year" is a problem statement.

B. Technical Specification. A complete technical specification of the proposed change, including all modified consensus rules, their interaction with existing rules, and a formal analysis of edge cases. The specification should be detailed enough to permit independent implementation.

C. Backward Compatibility Analysis. A comprehensive analysis of the proposal's impact on existing transactions, wallets, applications, and use cases. This analysis should identify every category of transaction or script that would become invalid under the new rules and quantify, to the extent possible, the number of affected users and the value at risk.

D. Activation Mechanism. A fully specified activation mechanism including signaling method, threshold, signaling window duration, timeout behavior, and defined failure mode. The activation mechanism should be described with sufficient precision to permit independent implementation and verification.

E. Rollback Procedure. For proposals self-described as "temporary," a self-executing sunset mechanism is required (see §3.6). For other soft-fork proposals, no rollback procedure is

required, but the proposal should not affirmatively obstruct future reversal — the consensus rules should remain technically reversible via standard soft-fork or hard-fork mechanisms.

F. Reference Implementation. A complete, functional reference implementation against a current release of Bitcoin Core or a compatible client. The reference implementation must include a comprehensive test suite.

3.2 Minimum Review Period

No consensus change proposal should proceed to activation signaling until it has completed a minimum review period. The appropriate length of this period depends on the scope and risk of the proposed change:

Category 1: Low-Risk Changes

Changes that tighten existing policy without altering the consensus boundary — for example, reducing default mempool relay limits. These are not consensus changes and do not require this framework. Individual node operators can adopt or reject them at will.

Category 2: Moderate-Risk Consensus Changes

Soft forks that add new validation rules without invalidating any currently valid transaction type. Examples include Taproot and SegWit, which expanded the set of valid scripts without restricting existing scripts. **Minimum review period: twelve months** from publication of a complete technical specification and reference implementation.

Category 3: High-Risk Consensus Changes

Soft forks that invalidate currently valid transaction types, restrict existing functionality, or could cause loss of funds for users with pre-existing transactions or scripts. BIP-110 falls into this category: it would have invalidated transactions that are currently valid, potentially trapping funds in scripts that use restricted opcodes. **Minimum review period: twenty-four months** from publication of a complete technical specification and reference implementation.

Category 4: Hard Forks

Any change that old nodes would reject. Hard forks carry the highest risk of permanent chain splits and should be avoided except in extraordinary circumstances, such as a critical security vulnerability discovered in the deployed protocol. Where they are nevertheless pursued, this framework proposes the following minimum standards: (1) a thirty-six-month review period from publication of a complete technical specification and reference implementation; (2)

explicit replay protection or a published rationale for its absence; (3) demonstrated economic-node support, including affirmative commitments from major exchanges, custodians, and payment processors; and (4) a published chain-split contingency plan addressing user communication, exchange coordination, and the contractual question of which chain constitutes "Bitcoin" for outstanding obligations. The Bitcoin community's strong norm against hard forks — reinforced by the block size wars — remains the most effective deterrent, but the absence of a written standard has historically left proponents free to define their own. The necessity of replay protection is itself the diagnostic of hard-fork status. A proposal whose safe deployment requires the ecosystem to modify its transaction format to prevent cross-chain replay has, by that requirement, made the case for its hard-fork classification — and should be evaluated against the standards of this category rather than treated as a soft fork with a safety net bolted on. The standards that follow in §3.7 distinguish between scheduled and emergency hard forks, which have opposite governance properties despite sharing this category.

3.3 Code Audit Requirements

The activation client for any consensus change must meet the following code quality standards before activation signaling begins:

A. Diverse independent review. The reference implementation must be reviewed by a minimum of three developers whose primary organizational affiliations differ from each other and from the proposal's authors. "Organizational affiliation" for this purpose means current or recent (past 24 months) employer or material funding source. Reviewers must publicly disclose any prior collaboration with the proposal's authors — co-authorship on prior BIPs or papers, joint employment, or shared funding. Disclosure does not disqualify; it makes the relationship legible. The standard is diverse independent perspectives, not pristine isolation, which is unachievable in a small developer community and weaponizable when claimed. Reviewers should have demonstrated competence in Bitcoin protocol development, as evidenced by prior contributions to Bitcoin Core, Bitcoin Knots, or another consensus-compatible implementation.

B. Test coverage. The reference implementation must include unit tests covering all new validation rules, integration tests demonstrating compatibility with existing valid transactions, and regression tests for all identified edge cases. Test results must be publicly reproducible.

C. Testnet deployment. The activation client must be deployed on Bitcoin's public testnet for a minimum of three months before mainnet activation signaling begins. The testnet deployment must demonstrate successful activation, enforcement of new rules, and — critically — successful deactivation if the proposal includes a sunset clause.

D. Fuzzing and adversarial testing. The reference implementation should be subjected to automated fuzzing and adversarial testing to identify vulnerabilities that could be exploited

during or after activation. This is particularly important for proposals that restrict transaction types, as attackers may attempt to craft transactions that exploit ambiguities in the new rules.

E. Reviewer comprehension. Every change to consensus-critical code must be reviewed by at least one named human reviewer who publicly attests that they understand the change and can defend its correctness against technical challenge. This requirement applies regardless of the code's origin — whether authored by humans, generated by AI tools, or adapted from prior proposals. AI-generated code in consensus-critical software is analogous to AI-generated legal filings: the tool can accelerate production, but the professional remains responsible for the output's correctness. The framework does not attempt to police the origin of code, which is undetectable on inspection; it requires the comprehension of code by named accountable reviewers, which is testable in any review forum. Authors and reviewers are welcomed to disclose AI involvement voluntarily as a matter of transparency, but the load-bearing requirement is reviewer comprehension, not origin disclosure.

3.4 Activation Threshold Standards

The activation threshold for a consensus change should reflect both the risk of the change and the cost of a failed activation. The following thresholds are proposed as minimum standards:

Activation Mechanism Design Space

Threshold selection is downstream of mechanism selection. The activation mechanisms employed in Bitcoin's history form a small but instructive design space. Each represents a distinct trade-off between activation speed, fail-safe behavior, and the locus of signaling authority.

BIP-9 (Version Bits with Timeout and Delay). The original modern activation mechanism, used for CLTV (BIP-65), CSV (BIPs 68, 112, 113), and SegWit (BIP-141). Miners signal readiness via version-bit flags; activation occurs when the configured threshold of blocks within a 2,016-block window signal readiness. BIP-9 includes a timeout: if the threshold is not met within a defined window, the deployment expires. BIP-9's principal weakness is that it grants miners effective veto power — a small mining pool coalition can block activation by refusing to signal, even where node operators, exchanges, and users overwhelmingly support the change. SegWit's eight-month stall demonstrated this failure mode in practice.

BIP-91 (Reduced Threshold MASF). A direct response to SegWit's stalled BIP-9 deployment. BIP-91 lowered the lock-in threshold to 80% and made signaling itself compulsory: BIP-91-enforcing miners would reject blocks that did not signal for SegWit. This created the coordination pressure that achieved SegWit's lock-in in July 2017. BIP-91 demonstrated that compulsory signaling — not just threshold reduction — can break a deadlock, at the cost of accepting some chain-split risk during the coordination period.

BIP-148 (User Activated Soft Fork). A flag-day mechanism: BIP-148-enforcing nodes would begin rejecting non-SegWit-signaling blocks on August 1, 2017, regardless of miner readiness. BIP-148 transferred activation authority from miners to economic nodes. BIP-148 worked because overwhelming community support produced miner capitulation; the mechanism's credible threat is widely credited with motivating the capitulation that produced BIP-91. UASF without that prior support yields the BIP-110 outcome. The mechanism is structurally riskier than miner-activated alternatives — if miners refuse to comply, the BIP-148 chain splits from the legacy chain — and is appropriate only where the underlying proposal has overwhelming economic support.

Speedy Trial. The activation mechanism used for Taproot in 2021. A bounded BIP-9 deployment with a short signaling window (approximately three months) and a high threshold (90%). If the threshold is reached, activation occurs after a defined delay; if not, the proposal expires without activation, freeing the deployment slot for revision or alternative proposals. Speedy Trial trades the certainty of activation for the certainty of a defined timeline. It is the most cautious of the modern mechanisms and is the closest existing match to the standards proposed in this framework.

LOT=true / LOT=false. A parameter that arose in BIP-8 during Taproot's activation debates. Lockin On Timeout (LOT) specifies whether a deployment that fails to achieve miner signaling within its window should nevertheless activate by user mandate at timeout. LOT=true converts a deployment into an effective UASF if miners do not cooperate; LOT=false makes it a clean Speedy Trial-style timeout. LOT=true is unreliable in a first attempt and disfavored even on repeat attempts. The LOT debate of early 2021 surfaced the central question of every activation mechanism: who holds final authority over consensus rule changes, and what happens when they disagree?

This framework treats activation mechanism selection as a deliberate design choice subject to the standards in this section. A proposal that meets the substantive standards but uses an inappropriate activation mechanism remains deficient. A proposal that uses an appropriate mechanism but fails the substantive standards is no more defensible. Mechanism and merit are both gating.

Miner-Activated Soft Fork (MASF)

Minimum threshold: 90% of hashrate measured over a defined signaling period of at least two weeks (2,016 blocks). This threshold is consistent with Taproot's successful activation and ensures that the risk of a chain split is minimized. Thresholds below 80% should be considered presumptively dangerous. Thresholds below 60% are reckless and should be rejected regardless of the proposal's merits.

User-Activated Soft Fork (UASF)

UASFs should be reserved for situations in which a proposal has demonstrated overwhelming community support but miner signaling is blocked by a small number of mining pool operators acting against their users' interests. The original UASF rationale, articulated in BIP-148 and the contemporaneous bitcoin-dev mailing list discussions of early 2017, held that user-activated mechanisms transfer activation authority from miners to economic nodes when the two diverge. A UASF is an extraordinary measure. It should not be the default activation mechanism for any proposal.

A UASF should only proceed when: (1) the proposal has completed its full minimum review period; (2) the proposal has demonstrated broad support among economic nodes, exchanges, and major holders; (3) the UASF includes a defined activation date set at least six months in the future to provide time for preparation; and (4) the UASF proponents have published a detailed chain split contingency plan.

The 55% Problem

BIP-110's 55% threshold deserves specific discussion because it illustrates the danger of low thresholds. A 55% signaling threshold permits activation under conditions in which the hashrate actually enforcing the new rules at activation may not materially exceed 55%. The risk of a persistent minority chain depends on enforcement, not signaling — the two are not the same. If a meaningful share of post-activation hashrate produces blocks valid under the legacy rules but invalid under the new ones, those blocks will be rejected by activated nodes. The activated chain will fall behind the non-activated chain in cumulative proof of work, and activated nodes will be on a minority chain that, by Bitcoin's own rules, is not the valid chain. The 95% threshold used by SegWit and the 90% threshold used by Taproot exist precisely to drive that probability toward zero. A 55% threshold makes a persistent split foreseeable.

A 55% threshold does not safely activate a soft fork. It produces the conditions historically associated with persistent minority chains: weeks or months of competing tips, ambiguous economic status for transactions confirmed on either side, and unresolved questions about the meaning of "Bitcoin" in contracts and on exchanges. The fact that P2SH used the same threshold in 2012 is not persuasive precedent: P2SH was a narrow, low-risk change to a network with a fraction of today's value and user base. The stakes have changed. The standards must change with them. This conclusion is consistent with the qualitative analysis in *BCAP* §3.5.2, which identifies a low percentage of Economic Node enforcement as producing the highest risk of chain split and which describes the mechanism — sub-overwhelming enforcement permitting unupgraded blocks to extend the chain — by which that risk is realized.

Quantifying the Risk

The danger of a 55% threshold can be quantified. Model post-activation hashrate as enforcing (share E) and non-enforcing (share $1 - E$). Treat block production as a Bernoulli process: each block is, with probability E, valid under the new rules and built on the enforcing chain; with probability $1 - E$, it violates the new rules and extends a competing non-enforcing chain. Enforcing nodes follow the longest chain that is valid under the new rules; non-enforcing nodes follow the longest chain absolutely.

The difference in cumulative work between the chains is a random walk with drift $2E - 1$ per block. The probability that, at some point during activation, the non-enforcing chain temporarily exceeds the enforcing chain by k blocks is well-approximated by $\left(\frac{1-E}{E}\right)^k$. Substituting for $k = 6$, the depth at which most exchanges credit deposits as final:

- At $E = 0.55$, the probability of a six-block deficit on the enforcing chain during activation is approximately $(0.45/0.55)^6 \approx 0.30$ — roughly thirty percent.
- At $E = 0.90$, the same probability is approximately $(0.10/0.90)^6 \approx 1.9 \times 10^{-6}$.
- At $E = 0.95$, it is approximately $(0.05/0.95)^6 \approx 2.1 \times 10^{-8}$.

Each such deficit represents an opportunity for a reorganization that orphans transactions previously confirmed on the enforcing chain. Exchanges that require six confirmations before crediting a deposit would, at $E = 0.55$, see roughly one in three confirmation chains exposed to a reorg event at some point during activation. At $E = 0.90$ or above, such events are statistically nonexistent over the relevant time horizons. The 55%, 90%, and 95% thresholds are not points on a linear spectrum of safety: they differ by five to seven orders of magnitude in expected reorganization exposure during activation.

This analysis simplifies several factors. It assumes non-enforcing miners produce blocks at the natural rate proportional to their hashrate; in practice, miners may defect to whichever chain becomes profitable, accelerating consolidation. It also assumes that every non-enforcing block contains a transaction that activated nodes would reject; for proposals like BIP-110 that target common transaction patterns, this assumption is largely realized, but for narrower changes the effective fork rate is lower. Both factors affect the absolute magnitudes; neither changes the qualitative conclusion that low-threshold activations are quantitatively distinct from high-threshold ones in their risk profile.

§3.4 in brief

The activation threshold is the share of mining power that signals it will enforce the new rules. Higher threshold means less chance the network splits during activation.

- **90% or above** (SegWit, Taproot): splits are statistically nonexistent. Safe.
- **80–90%:** risky but historically achievable. Most of the margin of safety is gone.
- **60–80%:** presumptively dangerous. Splits become foreseeable.
- **Below 60%** (BIP-110's 55%): reckless. About one-in-three chance of a six-block reorg during activation. At 95%, the same chance is effectively zero — two parts in a hundred million.

These thresholds do not sit on a smooth curve. The risk gap between 55% and 95% is five to seven orders of magnitude — millions of times, not a few percentage points. "55% is only 35 points below 90%" is the wrong way to think about it.

3.5 Chain Split Risk Assessment

The chain-split mechanisms this assessment must address are developed in detail in *BCAP* §3.5.2, including the bounty-claim scenario in which assets locked into scripts using newly proposed rules generate incentives for miners to mine an unupgraded block that voids the new-rule protection. Proposals that introduce new opcodes or new spending paths through OP_SUCCESS substitution warrant particular attention to that risk. The risk-assessment requirements that follow are the documentary correlates of that mechanism — questions a proponent must answer in writing before the community can rationally evaluate exposure.

Every consensus change proposal should include a formal chain split risk assessment addressing, at minimum:

A. Hashrate distribution analysis. What percentage of current hashrate is operated by pools or miners likely to adopt the change? What percentage is likely to reject it? Is there a credible path to the activation threshold, or is the proposal being pushed despite inadequate support?

B. Economic node analysis. Have major exchanges, payment processors, and infrastructure providers indicated support for the change? A consensus change that activates without exchange support creates immediate economic disruption, as users cannot deposit or withdraw funds until exchanges upgrade.

C. Replay protection. If the proposal would require explicit replay protection to prevent transactions from being valid on both chains in the event of a split, the proposal has crossed into hard-fork territory under Category 4 (see §3.7 on hard-fork subtypes) — reassess the classification before continuing the chain-split risk assessment. If the proposal is a soft fork that cannot produce a chain split absent miner defection, document the rationale for

the absence of replay protection and address the expected impact on users in the residual miner-defection scenario.

D. Contingency plan. What happens if the activation fails? What happens if the activation succeeds but produces a persistent minority chain? Who is responsible for communicating the split to users, and how?

3.6 Sunset and Reversibility Requirements

Proposals described as "temporary" must include a self-executing sunset clause. This means that the consensus rules imposed by the proposal must automatically expire at a defined block height or timestamp without requiring any further community action. The burden of continuation should fall on proponents of the change, not on opponents.

Specifically, a valid sunset clause must:

A. Define an exact block height or median time past (MTP) at which the new rules cease to be enforced.

B. Be implemented in the activation client such that nodes automatically revert to pre-activation consensus rules upon reaching the sunset trigger.

C. Be tested on testnet to confirm that deactivation works correctly and does not itself produce consensus failures.

D. Not require a subsequent soft fork, hard fork, or software update to effectuate deactivation.

A proposal that describes itself as temporary but requires active intervention to expire is not temporary. It is permanent with a stated aspiration.

3.7 Hard Fork Subtypes: Scheduled vs. Emergency

Category 4 (Hard Forks) is treated by the framework's general standards as a single category requiring the highest scrutiny. In practice, hard forks fall into two extremes with opposite governance properties, and a single set of standards cannot accommodate both. The framework therefore distinguishes them: subtype 4a, the scheduled hard fork, and subtype 4b, the emergency hard fork. Each carries its own standards. The Category 4 text gestures at the emergency case by example ("a critical security vulnerability discovered in the deployed protocol") but, applied uniformly, would impose a thirty-six-month review period on a vulnerability that cannot wait three years. The subtype distinction resolves that tension.

4a — Scheduled Hard Fork

A scheduled hard fork has a deployment horizon measured in years from initial proposal to activation. The defining property is that time is the principal asset: organic node-upgrade cycles can do most of the coordination work if the deployment window is long enough. Empirical analysis of historical node-upgrade rates (see §1.5) suggests that windows of approximately five years or longer produce near-universal upgrade in the absence of active opposition. A scheduled hard fork that takes advantage of this property can be deployed with substantially less coordination burden than an accelerated hard fork attempt, because the upgrade arrives organically through routine maintenance cycles.

The standards for subtype 4a follow Category 4 generally, with refinements:

A. Minimum review period: five years from publication of a complete technical specification and reference implementation. This is longer than Category 3's twenty-four-month floor; the additional review window reflects the deployment horizon's premium on getting the change right the first time.

B. Broad consensus required prior to code freeze, not merely prior to activation. The deployment window operates only as designed if the code shipped at the start of the window is the code that will be enforced at the end.

C. Explicit deployment timeline published in advance with each milestone identified: specification freeze, code freeze, infrastructure-test deployment, and mainnet activation block height.

D. Replay protection may be unnecessary if the upgrade window genuinely absorbs the change — that is, if the organic upgrade rate at the activation height is sufficient that no meaningful minority continues to enforce the prior rules. Replay protection becomes necessary only if the active-coordination component re-enters as a fallback.

4b — Emergency Hard Fork

An emergency hard fork is triggered by a critical vulnerability or imminent failure mode in the deployed protocol that cannot wait for a multi-year deployment window. Examples include the 2013 chain split that produced BIP-50, an inflation bug analogous to CVE-2018-17144 for which the soft-fork mitigation path is unavailable, or a cryptographic primitive becoming compromised in a way that affects existing UTXOs. The defining property is that time is the principal enemy: the alternative to a fast hard fork is running broken software.

The standards for subtype 4b are not lenient versions of Category 4's standards. They are different standards, calibrated to a different problem. Imposing the scheduled-HF standards on an emergency case would be actively dangerous; imposing the emergency standards on a scheduled case would be reckless. The 4b standards are:

A. Documented evidence of the trigger. The proposal must identify the specific failure mode, the timeline within which the failure becomes operational, and the technical basis for the timeline. "Critical vulnerability" without published technical detail is not sufficient.

B. Compressed expert-led review. The review timeline is calibrated to the threat. The standards of §3.3 (Code Audit Requirements) apply — at least three reviewers from distinct organizational affiliations, named human comprehension attestation, testnet deployment — and the required duration is the minimum that allows these standards to be satisfied, typically weeks to months depending on the threat's timeline.

C. Near-unanimous infrastructure coordination at the activation moment. Major exchanges, custodians, payment processors, and mining pools must commit to the activation timeline together. Soft commitments are insufficient; the coordination must be sufficient that a chain split does not produce a viable competing chain.

D. Replay protection where a meaningful minority will continue running the prior rules. In emergency-HF cases, the choice not to upgrade is itself often a principled position — an operator who disagrees that the trigger justifies the fork — and the prior-rules chain may persist with non-trivial economic activity. Replay protection in this scenario is not a category error; it is necessary user protection.

E. Transparent post-hoc disclosure of the decision process. The compressed timeline necessarily limits the breadth of pre-deployment consultation. The legitimacy of an emergency hard fork is established afterward by publishing what was decided, by whom, on what evidence, and with what dissents — making the process accountable in retrospect even where it could not be open in real time.

The framework does not anticipate emergency hard forks being common. Bitcoin's deployed protocol is exceptionally well-reviewed, and the post-2009 record contains no inflation-bug-level emergencies that required this path. BIP-50 itself was resolved by a coordinated downgrade, not a hard fork. The recognition of subtype 4b is the framework's acknowledgment that some governance needs are inverted under emergency conditions — not that emergency conditions suspend governance. The standards above exist to make the emergency case legible and challengeable, in the same way that the scheduled-HF standards make the planned case legible and challengeable.

Chapter 3 in brief

The seven standards a proposal should meet, in plain language:

1. **Complete proposal.** Problem statement (with data), technical spec, backward-compat analysis, activation mechanism, rollback, reference code with tests.
2. **Minimum review period.** Twelve months (adds rules) / twenty-four (invalidates transactions) / thirty-six (hard fork). Sized to organic node-upgrade cycles.
3. **Independent review.** Three developers from distinct organizations, prior collaboration disclosed. Diverse perspectives, not pristine isolation.
4. **Activation threshold.** 90% miner signaling minimum. Below 80% is dangerous; below 60% is reckless (see §3.4 box).
5. **Chain-split risk assessment.** Hashrate distribution, economic-node support, replay-protection rationale, and a written contingency plan.
6. **Self-executing sunset.** A "temporary" proposal must auto-deactivate at a defined block height. Anything else is permanence with extra steps.
7. **Hard-fork standards.** Scheduled: five-year horizon. Emergency (vulnerability-driven): documented threat, compressed expert review, near-unanimous infrastructure coordination, replay protection where a minority holds out.

The standards are not enforceable by code — Bitcoin has no enforcer. They are a benchmark. A proposal that meets them deserves serious engagement; one that doesn't should be treated skeptically, no matter who is championing it.

Chapter 4

Legal Analysis

The legal implications of Bitcoin consensus changes are largely unexplored. This is partly because Bitcoin's decentralized nature complicates the application of traditional legal frameworks, and partly because no chain split has yet produced litigation with reported opinions. But the absence of precedent does not mean the absence of liability. The following analysis applies established tort and contract principles to the specific risks created by reckless consensus change activation.

4.0 Purpose of the Legal Analysis

This section analyzes the legal-risk landscape that already exists, whether developers acknowledge it or not. It is not an invitation to litigation. It is a map of the terrain — drawn so that participants, including developers, can navigate it with informed caution rather than unknowing exposure.

A developer who follows the standards proposed in Section 3 has a strong defense against every theory analyzed below. The legal analysis is best read in that light: as a description of what good-faith developers can demonstrate they have met, not as a catalog of theories under which they might be pursued. Section 3 is the safety standard; Section 4 is the case for why following the standard is itself the most effective protection against the liability theories that already exist in tort and contract law.

This section serves the network as a whole by making the legal landscape legible to every stakeholder — operators, exchanges, custodians, and developers — who must navigate it without specialized training. Litigation against developers acting in good faith would harm the protocol and the community; the framework's purpose is to make such litigation less likely by establishing a documented standard of care that good-faith developers can demonstrably satisfy.

4.1 Negligence

Tort liability for negligence requires a duty of care, a breach of that duty, causation, and damages. The threshold question is whether the developers of a consensus change activation client owe a duty of care to node operators and users who run their software.

Under traditional tort principles, a person who creates a dangerous instrumentality and places it into the stream of commerce owes a duty of care to foreseeable users. An activation client for a Bitcoin consensus change is software that, if defective, can cause direct financial harm to its users. The analogy to products liability is imperfect — most activation clients are distributed as free, open-source software — but open-source licenses do not categorically eliminate tort liability, particularly where the developer actively encourages adoption and knows that defects could cause financial loss.

A second doctrinal obstacle warrants attention. California's economic loss rule generally bars recovery in tort for purely economic harm absent physical injury, property damage, or a special relationship giving rise to an independent duty. See *Aas v. Superior Court*, 24 Cal.4th 627 (2000); *Robinson Helicopter Co. v. Dana Corp.*, 34 Cal.4th 979 (2004). The rule is a genuine impediment to a negligence theory against open-source developers whose users hold no contract with them and whose harm is financial rather than physical. Two routes around the rule remain viable. The multi-factor test of *Biakanja v. Irving*, 49 Cal.2d 647 (1958), permits a duty of care to non-contracting parties where the developer's conduct was intended to affect the user, the harm was foreseeable, and policy supports liability. And where the developer makes representations about the safety or readiness of an activation client on which users foreseeably rely, negligent misrepresentation under Restatement (Second) of Torts § 552 supplies a recognized cause of action without confronting the economic loss bar. Both routes require facts beyond the bare release of buggy software; both are available on facts of the kind BIP-110 presents.

BIP-110's activation client illustrates the potential for negligence liability. The client was released with known bugs. Developers publicly identified defects that could cause users to fork themselves off the network. Despite these warnings, the client was distributed and its adoption was promoted on social media. If a user had run this client and suffered a financial loss — for example, by mining blocks on a minority chain that were subsequently orphaned — a negligence claim against the client's developer would face challenging but not insurmountable hurdles.

The strongest argument against liability is assumption of risk: users who run experimental software on a production network are arguably assuming the risk of loss. But assumption of risk is an affirmative defense, not a bar to the existence of a duty. And the defense is weaker when the software is presented alongside production releases in a platform's version management system, as BIP-110 was on at least one node management platform.

4.2 Tortious Interference

A reckless consensus change activation that causes a chain split could give rise to claims of tortious interference with contractual relations or business expectancy. Consider the following scenario: a business accepts Bitcoin as payment under a contract that specifies payment in "Bitcoin." A chain split occurs, and the payor delivers coins on the minority chain. The payee

argues that "Bitcoin" means the majority chain. The resulting dispute was proximately caused by the chain split, which was proximately caused by the reckless activation.

The question of which chain constitutes "Bitcoin" after a split has no settled legal answer. During the Bitcoin Cash fork, exchanges and contracts generally treated the chain with the most accumulated proof of work and the greatest economic activity as "Bitcoin." But this convention is informal and could be challenged.

Proponents of consensus changes that carry a material risk of chain split should consider whether their actions could expose them to tortious interference claims. This is particularly relevant when the proponent is a company or public figure whose advocacy for the change is well-documented and whose economic interest in the change's success is apparent.

Two elements warrant emphasis. Tortious interference with contract requires intentional acts designed to disrupt performance, not merely conduct that has the foreseeable effect of doing so. See *Pacific Gas & Electric Co. v. Bear Stearns & Co.*, 50 Cal.3d 1118 (1990). Tortious interference with prospective economic advantage requires, in addition, an independently wrongful act — conduct unlawful for reasons other than the interference itself. See *Della Penna v. Toyota Motor Sales, U.S.A., Inc.*, 11 Cal.4th 376 (1995); *Korea Supply Co. v. Lockheed Martin Corp.*, 29 Cal.4th 1134 (2003). A developer who promotes an activation client out of ideological conviction will not typically meet either standard. The live claims are those in which the proponent of a consensus change has documented economic exposure to the outcome — a financial position whose value depends on which chain prevails — and the proponent's promotion can fairly be characterized as instrumental rather than principled. Such cases are not hypothetical. The framework's documentary requirements — problem statement, backward compatibility analysis, contingency plan — are themselves designed to create the record by which such inquiries can be conducted.

4.3 Fiduciary Duties

Some legal scholars have argued that Bitcoin developers owe fiduciary duties to Bitcoin holders, analogous to the duties owed by corporate directors to shareholders. The most developed version of this argument is Angela Walch, "In Code(rs) We Trust: Software Developers as Fiduciaries in Public Blockchains," in *Regulating Blockchain: Techno-Social and Legal Challenges* (Hacker, Lianos, Dimitropoulos & Eich eds., Oxford University Press 2019), which contends that core protocol developers exercise discretionary authority over property interests in a manner that triggers fiduciary obligations under traditional principles. This argument has been most fully developed in the Tulip Trading litigation in the United Kingdom. In *Tulip Trading Ltd v van der Laan* [2023] EWCA Civ 83, the Court of Appeal of England and Wales did not adjudicate whether Bitcoin Core developers owe a fiduciary duty to holders of lost Bitcoin. It held only that the claim was sufficiently arguable to survive a strike-out application and should proceed to trial. The case thus stands not for the proposition that developers owe such duties, but for the proposition that the question is

justiciable on appropriate facts. The Court of Appeal separately affirmed the strike-out of a parallel common-law negligence claim, holding no arguable duty of care arose on the facts before it — which concerned developers' alleged failure to issue a patch enabling recovery of stolen Bitcoin. That fact pattern is structurally distinct from the chain-split scenario this paper addresses, where alleged harm would flow from the active deployment of defective consensus-altering code, not from a failure to act. The English negligence holding is therefore narrower than its surface might suggest and does not foreclose the California analysis in §4.1, which rests on routes around the economic-loss rule — particularly *Biakanja* and Restatement § 552 — that turn on facts of intent, reliance, and representation that *Tulip Trading* did not engage.

This paper does not take a position on whether developers owe fiduciary duties in general. However, the analysis is relevant to consensus change governance for the following reason: if a developer promotes a consensus change, distributes an activation client, and the change causes financial harm, the question of whether the developer's conduct constituted a breach of duty will be evaluated against the standard of care exercised in the process.

A developer who follows a rigorous governance framework — adequate review, thorough testing, conservative activation thresholds, and transparent communication of risks — has a strong defense against any claim of breach. A developer who releases a buggy client with a low activation threshold and no review period does not.

4.4 Mining and Node Operator Liability

Miners and node operators who adopt activation clients also face potential liability exposure. A mining pool that signals for a consensus change bears some responsibility for the consequences of that signaling, particularly if the pool operator has not communicated the risks to the pool's users (individual miners).

For solo miners — a category that includes the author — the liability analysis is simpler: a solo miner who runs an activation client is assuming the risk of their own operation. But a pool operator who signals on behalf of thousands of connected miners has a duty to those miners to exercise reasonable care in evaluating the consensus change. Mining pool operators who signal for poorly reviewed proposals with low activation thresholds are taking risks with other people's hashrate and, by extension, other people's money. The most likely cause of action is breach of contract: pool terms of service and service agreements typically obligate the operator to direct connected hashrate with reasonable care, and signaling for a consensus change that exposes connected miners to orphaned blocks is the kind of decision that brings such obligations into play. Where the pool agreement is silent on consensus signaling, the implied covenant of good faith and fair dealing may supply the duty. Negligence remains available as a backstop, subject to the economic-loss-rule limitations discussed above.

4.5 Regulatory Consequences

Beyond direct liability, a chain split creates regulatory uncertainty. Tokens on both chains may be treated as separate assets for tax purposes, requiring holders to determine cost basis allocation. Exchanges may be required to support both chains or face claims from customers who hold tokens on the unsupported chain. Custodians may face conflicting obligations to clients.

These regulatory consequences are not hypothetical — they occurred during the Bitcoin Cash fork in 2017 and required guidance from the IRS (Revenue Ruling 2019-24) to resolve. A governance framework that minimizes the risk of chain splits also minimizes regulatory disruption. This is a feature, not a byproduct.

4.6 Comparative Note: Common-Law Jurisdictions and EU Software Liability

The legal analysis in §4.1 through §4.5 draws primarily on California state law and federal authority. The analytical method — negligence, tortious interference, fiduciary duty, contract, and regulatory exposure — translates across common-law jurisdictions with varying doctrinal specifics. This note maps the California analysis onto its closest cousins, identifies the most consequential current international development, and is explicit about civil-law limits.

A. United Kingdom. The duty-of-care analysis in §4.1 has direct parallel in English common law. The leading modern duty-of-care case, *Caparo Industries plc v Dickman* (House of Lords, 1990), establishes a three-stage test: (1) foreseeability of harm; (2) proximity between the parties; and (3) whether imposition of duty is fair, just, and reasonable. *Caparo*'s structural parallel to California's *Biakanja* multi-factor test makes the §4.1 analysis substantially portable for UK readers. Negligent misstatement claims under *Hedley Byrne & Co Ltd v Heller & Partners Ltd* (House of Lords, 1964) supply a UK analogue to the Restatement § 552 route in California, where the defendant assumed responsibility through representations on which the claimant reasonably relied. As §4.3 explains, the Court of Appeal in *Tulip Trading* applied the *Caparo* framework to deny duty of care on its facts; that holding is narrower than its surface might suggest.

B. Commonwealth common-law jurisdictions. Singapore, Australia, Canada, New Zealand, and other Commonwealth jurisdictions derive their tort frameworks from English common law but have evolved distinct doctrines. Australia notably departed from *Caparo*'s three-stage test in *Sullivan v Moody* (High Court of Australia, 2001), adopting a framework that has moved away from the *Caparo* stages. Canada retained a modified version of the predecessor *Anns* (House of Lords, 1978) two-stage approach, refined through *Cooper v Hobart* (Supreme Court of Canada, 2001). The §4.1 analytical method translates as a roadmap, but jurisdiction-specific

case law would govern any specific dispute; readers in these jurisdictions should consult counsel familiar with local doctrine.

C. European Union software liability. The most consequential current development in international software developer liability is the European Union's revised Product Liability Directive, adopted in 2024 to replace the 1985 directive. The 2024 revision explicitly extends strict product liability to software. Member States must transpose the new directive into national law within roughly two years of its entry into force. After transposition, a developer of defective software — potentially including a Bitcoin consensus-change activation client — may face strict liability for personal injury or property damage caused by the defect, subject to defenses available under the directive. The directive's application to non-commercial open-source software remains contested in early commentary, with arguments that purely volunteer-distributed open-source software may fall outside its scope. This paper takes no position on that contested interpretive question. The practical implication is that EU jurisdictions will, post-transposition, present a substantially different liability landscape than the negligence-centered analysis in §4.1. Developers and pool operators subject to EU jurisdiction should consult counsel familiar with the directive's specifics in the relevant Member State.

D. Civil-law jurisdictions. Civil-law tort doctrine — including the foundational Swiss Code of Obligations Article 41 general clause and analogous provisions in other civil-law systems — operates on different conceptual premises than common-law negligence. The framework's procedural standards in Section 3 and Section 5 translate (they evaluate procedural conduct, jurisdictionally agnostic); the legal-exposure analysis in Section 4 does not directly port and would require adaptation to local doctrine. This paper does not attempt that adaptation.

The framework's standards in Section 3 and Section 5 are jurisdictionally portable. They evaluate the procedural conduct of proposal development and activation, independent of any particular legal regime. The legal-exposure analysis in §4.1 through §4.5 is California-specific and should be understood as a methodological model. Liability exposure exists, in varying doctrinal forms, in every jurisdiction with developed tort, contract, and regulatory law. Practitioners outside the United States should treat Section 4 as a roadmap for the analytical questions to ask of local counsel, not as the answers themselves.

Chapter 4 in brief

When a recklessly activated change causes a chain split and people lose money, more parties are legally exposed than is widely appreciated:

- **Negligence (§4.1).** Releasing activation code with known bugs and promoting its adoption can produce negligence claims; *Biakanja* and Restatement § 552 supply routes around the economic-loss rule on the right facts.
- **Tortious interference (§4.2).** A proponent with documented economic exposure who pushes a change that splits chains can face live claims, especially if promotion looks instrumental rather than principled.
- **Fiduciary duty (§4.3).** Unsettled, but *Tulip Trading* (UK Court of Appeal, 2023) confirmed the question is justiciable. The developer's process determines the answer.
- **Pool operators (§4.4).** Signaling on behalf of connected miners is a decision pools owe care over, reachable on contract or implied-covenant theories.
- **Regulatory and exchange exposure (§4.5).** Splits trigger tax-basis allocation (IRS Rev. Rul. 2019-24), custodial conflicts, exchange listing decisions. Not hypothetical — this already happened in 2017.

This is not a threat directed at developers. A developer who follows the Chapter 3 standards has a strong defense against every theory above. Section 3 is the safety standard; Section 4 is the case for why following it is the most effective protection. §4.6 maps the California analysis onto UK and Commonwealth common-law jurisdictions and flags the EU's 2024 Product Liability Directive as the most consequential current change in international software-developer liability.

Chapter 5

Proposed Standards

This chapter provides two evaluation tools that operate at different speeds. §5.0 is a fast screening test: seven red flags that surface immediately on inspection of a proposal and indicate that the full evaluation is warranted. §5.1 is the full evaluation: twenty binary criteria synthesizing the framework in Sections 3 and 4, with the scoring rules in §5.2. A proposal that meets all of the standards deserves serious community consideration. A proposal that fails to meet them should be treated with appropriate skepticism and, if it proceeds to activation without meeting them, should be actively resisted.

5.0 Red Flags: Is This Proposal Risky?

The following red flags surface from public information about a proposal — its activation parameters, its release timeline, its review record, and the conduct of its proponents. They are observable without running code or modeling outcomes. A proposal that trips two or more warrants the full §5.1 evaluation before any operator runs the activation client. The flags are gateways to the criteria below, not numerical scores in their own right.

1. **Activation threshold below 80%.** Sub-80% thresholds are presumptively dangerous; sub-60% is reckless. The risk gap between 55% and 95% is five to seven orders of magnitude in chain-split exposure, not a smooth percentage difference. See §3.4.

2. **Six weeks or less from initial proposal to activation client.** The framework's minimum review periods are twelve, twenty-four, or thirty-six months by risk category. Review periods substantially shorter than the floor have failed historically — see §1.3 (BIP-110) and §2.2 (block size wars).

3. **Activation client bundled into the default release stream of a node implementation.** Economic nodes get defaulted into the change through routine upgrades rather than opting in affirmatively. The governance failure is structural even where the runtime prompt is real. See §1.3, item D.

4. **Reviewers all share an employer, funder, or recent prior collaboration** that isn't publicly disclosed. The standard is diverse perspectives with relationships made legible, not pristine isolation. See §3.3, item A.

5. **"Temporary" proposal with no self-executing sunset**, or a sunset whose deactivation has not been tested on testnet. "Temporary" without a sunset is permanence with extra steps; an untested sunset is a promise, not a guarantee. See §3.6.
6. **Proponents have documented economic exposure to the outcome**, where promotion can fairly be characterized as instrumental rather than principled. See §4.2.
7. **No published chain-split contingency plan.** If activation produces a persistent minority chain, who communicates the split to users, exchanges, and counterparties — and how? See §3.5, item D.

A proposal that trips one flag may still merit serious consideration; tripping two or more triggers the full evaluation in §5.1; tripping four or more is reckless on its face. The list is intentionally short. A proposal can be problematic for reasons not enumerated here, and a flag that is technically not tripped is not a clean bill of health. The framework's full criteria are what determine readiness; the red flags determine only whether the framework needs to be run.

5.1 The Consensus Change Readiness Checklist

A. Proposal Quality

1. Does the proposal include a clear, empirically supported problem statement?
2. Does the proposal include a complete technical specification sufficient for independent implementation?
3. Does the proposal include a backward compatibility analysis identifying all affected transaction types, scripts, and use cases?
4. Does the proposal include a fully specified activation mechanism with defined thresholds, timelines, and failure modes?
5. Does the proposal include a rollback procedure? If described as temporary, does it include a self-executing sunset clause?

B. Code Quality

6. Has the reference implementation been reviewed by at least three developers with demonstrated Bitcoin protocol expertise, drawn from distinct organizational affiliations, with any prior collaboration with the proposal's authors publicly disclosed?
7. Does the reference implementation include comprehensive unit, integration, and regression tests?

8. Has the activation client been deployed on testnet for at least three months, with successful deactivation tested if a sunset clause is included?
9. Has the code been subjected to fuzzing and adversarial testing?
10. Has every change to consensus-critical code been attested as understood by at least one named human reviewer who can defend its correctness?

C. Activation Safety

11. Is the activation threshold at or above 90% for a MASF?
12. If a UASF mechanism is proposed, has the proposal completed its full minimum review period and demonstrated broad support among economic nodes?
13. Has a chain split risk assessment been completed and published?
14. Does the proposal include replay protection or, where the proposal is a soft fork that cannot produce a chain split absent miner defection, a documented rationale for its absence?
15. Has the activation timeline been set at least six months from the publication of the final activation client?

D. Community Process

16. Has the proposal completed the minimum review period for its risk category (twelve months for moderate-risk, twenty-four months for high-risk, thirty-six months for hard fork)?
17. Has the proposal been discussed in public forums with participation from a diverse cross-section of the community (developers, miners, node operators, businesses, users)?
18. Have major exchanges and infrastructure providers been consulted regarding the proposal's impact on their operations?
19. Has the proposal's author published a chain split contingency plan?
20. Has the proposal been evaluated against this framework, with the results published?

Scorecard worksheet

Mark each criterion "Met" or "Not"; tally at the bottom; classification bands in §5.2.

#	Criterion	Met	Not
A. Proposal Quality			
1	Clear, empirically supported problem statement.	☐	☐
2	Complete technical specification; independently implementable.	☐	☐
3	Backward-compatibility analysis covering all affected tx types, scripts, and use cases.	☐	☐
4	Fully specified activation mechanism (threshold, timeline, failure mode).	☐	☐
5	Rollback procedure; self-executing sunset if labeled "temporary."	☐	☐
B. Code Quality			
6	$\geq$ 3 expert reviewers from distinct organizations; prior collaboration disclosed.	☐	☐
7	Comprehensive unit, integration, and regression tests.	☐	☐
8	Testnet deployment $\geq$ 3 months; deactivation tested if sunset included.	☐	☐
9	Fuzzing and adversarial testing performed.	☐	☐
10	Named human reviewer attests comprehension of consensus-critical code.	☐	☐
C. Activation Safety			
11	MASF activation threshold $\geq$ 90%.	☐	☐
12	UASF (if used) has completed full review and broad economic-node support.	☐	☐
13	Chain-split risk assessment completed and published.	☐	☐
14	Replay protection, or documented rationale where soft fork cannot split absent defection.	☐	☐
15	Activation date $\geq$ 6 months from final activation client publication.	☐	☐
D. Community Process			
16	Minimum review period for risk category (12 / 24 / 36 months).	☐	☐
17	Public discussion across diverse stakeholders.	☐	☐
18	Major exchanges and infrastructure providers consulted.	☐	☐
19	Chain-split contingency plan published by proposal author.	☐	☐
20	Proposal evaluated against this framework; results published.	☐	☐

Total: _____ / 20

Classification (§5.2): 20 = Green; 15–19 = Yellow; 10–14 = Orange; < 10 = Red.

5.2 Scoring

Each of the twenty criteria above receives a binary score: met or not met. Proposals are classified as follows:

- **20/20: Green.** The proposal has met all minimum standards and is ready for activation signaling.
- **15–19/20: Yellow.** The proposal has met most standards but has identified gaps that should be addressed before activation.
- **10–14/20: Orange.** The proposal has significant deficiencies and should not proceed to activation signaling until they are resolved.
- **Below 10/20: Red.** The proposal fails to meet minimum standards for serious consideration. Activation should be actively resisted through: (a) public documentation of the deficiencies measured against this framework; (b) coordination among economic nodes — exchanges, custodians, payment processors, and major holders — to refuse to recognize the activated chain as "Bitcoin" for purposes of contracts, deposits, and withdrawals; (c) running non-signaling, non-activation client software; and (d) where activation proceeds despite these objections, publication of a chain-split contingency plan to ensure user safety and minimize the economic damage of the resulting fracture.

For reference, BIP-110 would score approximately 3/20 under this framework: credit for having a technical specification, a defined activation mechanism, and a self-executing sunset clause (though one whose deactivation was not demonstrably tested on testnet); no credit for adequate review period, code quality, independent review, activation safety, or community process. The bundling of RDTS into the default Bitcoin Knots release stream in May 2026 (§1.3, item D above) does not alter the binary count but intensifies the failure of criterion 12: where a UASF requires demonstrated broad economic-node support before proceeding, the operational reality after the default inversion is that economic nodes are being defaulted into the activation by the upgrade ladder rather than affirmatively opting in. The criterion's failure is now structural — non-endorsement has been engineered into the operator's burden — not merely a documentary gap. Taproot would score approximately 18/20: criteria 19 (a chain-split contingency plan in the form proposed here) and 20 (evaluation against this framework) postdate Taproot's activation and so are scored as not met, while every other criterion was met or exceeded.

5.3 On the Measurability of the Criteria

Several criteria in §5.1 — particularly those concerning broad community support, sustained opposition, and adequate review — are not amenable to mechanical measurement. There is no canonical metric for "broad support among economic nodes" or "sustained public objection" or "discussion in a diverse cross-section of the community." Reasonable evaluators can score the same proposal differently.

This is intentional. The framework's authority is its defensibility, not its precision. A score arrived at by counting blocks or polling exchanges would be precise but easily gamed; a score arrived at by transparent evaluation against published criteria, where reviewers must justify their scoring choices, is defensible against challenge in a way that algorithmic scoring cannot be. The fuzzy criteria force the conversation onto the right ground: what counts as evidence of broad support? What counts as sustained opposition? Who has objected, and have those objections been addressed? Those questions are themselves what the framework is asking the community to think about systematically.

For the fuzziest criteria, the following heuristics are illustrative anchors — not algorithmic rules. A proposal that satisfies the spirit of the criterion may not satisfy the literal heuristic and vice versa; the heuristic is a starting point for evaluators, not a substitute for evaluation.

"Broad support among economic nodes" (criterion 18) is reasonably evidenced by explicit public statements supporting the proposal from at least three of the top five exchanges by Bitcoin trading volume, plus comparable statements from at least two custodians serving institutional clients. Statements specifically endorsing the consensus change — not merely acknowledging it — count; silence does not count as support. The same evidentiary standard applies to criterion 12, which incorporates "broad support among economic nodes" as a precondition for UASF deployments.

"Sustained opposition" (a factor in §3.4 threshold selection) is reasonably evidenced by at least two prior public objections from independent stakeholders that have not been substantively addressed in subsequent proposal updates. "Independent" for this purpose follows the diverse-affiliation standard of §3.3.A: objectors should not all share the same employer or material funding source.

"Adequate community discussion" (criterion 17) is reasonably evidenced by archived discussion in at least two distinct venues over the full minimum review period for the proposal's risk category — typically the bitcoin-dev mailing list plus at least one of: a public forum thread, a recorded video discussion, or a working group with published minutes.

The remaining fuzzy elements in the scorecard's earlier criteria are anchored as follows.

"Clear, empirically supported problem statement" (criterion 1) is reasonably evidenced by a one-sentence problem statement supported by at least one citation to quantitative data, measured user behavior, or on-chain observation. Aspirational framing ("Bitcoin should remain X") is not a problem statement.

"Complete technical specification, independently implementable" (criterion 2) is reasonably evidenced where an independent developer could implement against the specification without consulting the author or reading the reference implementation. A reference implementation is a result of applying the specification, not a substitute for it.

"All affected transaction types, scripts, and use cases" (criterion 3) is reasonably evidenced by an explicit enumeration of every transaction type, opcode, and script pattern that becomes

invalid or restricted under the change, together with a quantitative estimate of the affected on-chain population (UTXO count, value at risk, or comparable measure).

"Three expert reviewers from distinct organizational affiliations" (criterion 6) draws on two fuzzy elements. "Expert" is reasonably evidenced by at least one merged contribution to a consensus-compatible Bitcoin implementation, or by published technical analysis of Bitcoin consensus mechanics. "Distinct organizational affiliation" follows the standard of §3.3.A: current or recent (past 24 months) employer or material funding source, differing across reviewers and from the proposal's authors.

"Comprehensive unit, integration, and regression tests" (criterion 7) is reasonably evidenced by test coverage of every new validation rule, every interaction with existing rules, and every identified edge case. Test results must be publicly reproducible by parties other than the implementation's authors.

These heuristics are illustrative because the framework's binary scoring is meant to be defensible against challenge, not bright-line. A reviewer who scores a criterion "met" using a different evidentiary basis than the heuristic above can defend that scoring by publishing the basis. A reviewer who scores a criterion "not met" can do the same. What the framework asks is not algorithmic agreement; it is that scoring choices be legible and challengeable. That is what makes the resulting evaluation defensible in any forum — technical, commercial, or legal — where the proposal's readiness must be assessed.

5.4 Worked Examples: Taproot and BIP-110

The twenty criteria yield concrete classifications when applied to specific proposals. This section walks through their application to the two examples introduced in §5.2: Taproot, which activated in November 2021 after years of review, and BIP-110, whose late-2025 activation-client release attracted no more than 0.15% peak signaling. The walkthrough shows how each criterion resolves on the public record and why the resulting scores (Taproot 18/20, BIP-110 3/20) reflect categorically different framework readiness.

The scoring below applies the heuristic anchors of §5.3 and follows the framework's commitment that scoring choices be "legible and challengeable." A reader who reaches different conclusions on the fuzzier criteria is invited to publish the alternative scoring with the evidentiary basis. The walkthrough's purpose is to demonstrate the framework's operation, not to fix the score against further argument.

#	Criterion	Taproot	BIP-110
A. Proposal Quality			
1	Empirically supported problem statement	✓ Schnorr efficiency, scriptpath privacy, and key aggregation are measurable improvements	× "Spam uses" framed without quantitative measurement
2	Complete technical specification	✓ BIPs 340/341/342 permit independent implementation	✓ RDTS specification permits implementation
3	Backward compatibility analysis	✓ Strict soft fork; OP_SUCCESS slot reservation preserves future flexibility	× No enumeration of affected transaction types or use cases
4	Fully specified activation mechanism	✓ Speedy Trial: 90% threshold, ∼3-month signaling window, defined timeout	✓ 55% UASF + LOT=true mandatory lock-in (Aug 2026); sunset ∼1 year after activation
5	Rollback / sunset (if temporary)	✓ Permanent change; soft-fork status preserves future-reversal possibility per §3.1.E	✓ Self-executing sunset at defined block height
B. Code Quality			
6	≥ 3 expert reviewers, distinct organizations	✓ Multi-year review across Bitcoin Core, academic cryptography, and other consensus-compatible implementations	× Customary review burden not met; reviewer affiliations not documented (§1.3)
7	Comprehensive unit, integration, regression tests	✓ Full test coverage merged into Bitcoin Core's test suite	× Client released with known bugs per §1.3
8	Testnet ≥ 3 months; deactivation tested if sunset	✓ Multi-year signet and testnet deployment	× Six-week proposal-to-client timeline; sunset deactivation not demonstrably tested
9	Fuzzing and adversarial testing	✓ Bitcoin Core fuzzing infrastructure exercised on Taproot consensus code	× Fuzzing not documented
10	Named reviewer comprehension attestation	✓ Multiple named reviewers attested comprehension across years of public review	× No comparable attestation documented
C. Activation Safety			
11	MASF threshold $\geq 90\%$	✓ Speedy Trial threshold of 90% (1,815 of 2,016 blocks)	× 55% UASF threshold; sub-60% is reckless under §3.4
12	UASF: full review + broad economic-node support	✓ Vacuously satisfied (Taproot used MASF)	× Six-week review; $\leq 0.15\%$ peak signaling
13	Chain-split risk assessment	✓ Published risk discussion as part of activation deployment	× No formal chain-split risk assessment published

#	Criterion	Taproot	BIP-110
14	Replay protection or documented rationale	✓ Soft-fork status documented as the rationale for absence	× 55% threshold makes chain split foreseeable; no rationale documented
15	Activation date $\geq$ 6 months from final client	✓ Activation client released early 2021; activation November 2021	× Activation signaling began immediately upon client release
D. Community Process			
16	Minimum review period for risk category	✓ Approximately four years of review; far exceeds 12-month moderate-risk floor	× Six weeks vs. 24-month floor for high-risk consensus change
17	Public discussion, diverse stakeholders	✓ Sustained bitcoin-dev, conference, podcast, and working-group discussion	× Public discussion concentrated and brief
18	Major exchanges and infrastructure consulted	✓ Major exchanges engaged and indicated readiness	× Exchanges not formally consulted; no public endorsements documented
19	Chain-split contingency plan published	× Contingency-plan format proposed by this framework postdates Taproot's activation	× No contingency plan published
20	Proposal evaluated against this framework	× Framework postdates Taproot's activation	× Never evaluated by proponents

Score: Taproot 18/20. BIP-110 3/20.

Taproot's two "Not Met" results (C19, C20) reflect criteria that postdate Taproot's 2021 activation rather than substantive defects in Taproot's process. Every criterion that could have been evaluated in real time was met or exceeded, frequently far above the framework's floors: review period (years against a twelve-month minimum), reviewer breadth (a multi-organization community against a three-reviewer minimum), and testnet exposure (multi-year deployment against a three-month minimum). The 18/20 score places Taproot in the Yellow band of §5.2's classification by literal application of the bands, but the two non-met criteria are framework-temporal artifacts; on the criteria evaluable at the time of activation, Taproot scored 18/18.

BIP-110's three "Met" results (C2 technical specification, C4 activation mechanism, C5 self-executing sunset) sit entirely within the Proposal Quality category and reflect the bare existence of a proposal document, a defined activation procedure, and a sunset block height. Every criterion in Code Quality, Activation Safety, and Community Process is not met, as are the remaining two Proposal Quality criteria (problem-statement empirical grounding and backward-compatibility analysis). The 3/20 score places the proposal in the Red band of §5.2, triggering the response there enumerated: public documentation of the deficiencies, coordination among economic nodes to refuse to recognize the activated chain as "Bitcoin" for

contracts and deposits, running non-signaling node software, and publication of contingency planning if activation nevertheless proceeds.

The fifteen-criterion gap between the two scores reflects the framework's substantive distinction: a proposal that received the rigor consensus-critical software requires versus a proposal that did not. The same framework, applied to the same kinds of evidence, produces categorically different results. A future BIP-110 variant that addressed even half the failed criteria would score differently — and the change would be visible in the record. The framework is a mechanism for making such differences legible to every stakeholder in the network.

Chapter 5 in brief

Two evaluation tools. **§5.0** is a seven-flag quick test — low threshold, rushed timeline, default-bundled client, undisclosed reviewer ties, untested sunset, conflicted promotion, missing contingency plan — for spotting reckless proposals on inspection. **§5.1** is the formal twenty-criterion checklist across four categories: Proposal Quality (1–5), Code Quality (6–10), Activation Safety (11–15), Community Process (16–20). Score each met / not met. **20/20 Green**: ready. **15–19/20 Yellow**: gaps. **10–14/20 Orange**: significant deficiencies. **Below 10/20 Red**: actively resist activation. Several criteria are deliberately fuzzy — "broad economic-node support" resists mechanical measurement and benefits from being defended in argument, not computed by formula (§5.3). §5.4 walks the scorecard criterion-by-criterion through Taproot (18/20) and BIP-110 (3/20, Red) as worked examples.

Chapter 6

Objections and Responses

6.1 "Bitcoin has no governance."

Bitcoin has no centralized governance. It has governance. Every consensus change that has ever been adopted required coordination among developers, miners, node operators, and economic actors. The process by which this coordination occurs — however informal — is governance. This framework does not propose centralized governance. It proposes minimum standards for evaluating proposals within Bitcoin's existing decentralized governance structure. The descriptive thesis of *BCAP* — that Bitcoin consensus emerges from the iterated, multi-party interactions of stakeholders with shifting powers and incentives — is correct and is not contested by this framework. The framework's claim is narrower and consistent with it: that minimum standards, publicly available and applied consistently, make the iterated process safer and more efficient by giving every stakeholder a shared vocabulary for declining engagement with proposals that are not ready.

6.2 "Anyone can run whatever software they want."

True. And this framework does not propose restricting that right. Node operators are free to run any software they choose. This framework proposes that the community develop shared standards for evaluating proposals, so that node operators can make informed decisions. A node operator who runs an activation client that fails every criterion in this framework is exercising their right. They are also assuming quantifiable risks that they may not fully understand. Providing a framework for understanding those risks serves the same function as securities disclosure: it does not restrict choice, it informs it.

6.3 "This framework would prevent necessary changes."

This framework would slow down reckless changes. It would not prevent necessary ones. SegWit and Taproot both would have passed this framework with high scores. The changes this framework would impede are the ones that should be impeded: poorly reviewed, inadequately tested, rashly activated proposals that put the network at risk.

Bitcoin's value proposition is stability, predictability, and resistance to arbitrary change. A framework that makes consensus changes harder to execute is aligned with that value proposition, not contrary to it.

6.4 "Who decides whether the standards are met?"

Everyone. And no one. This framework is a tool, not an authority. Any member of the community can evaluate a proposal against these criteria and publish the results. There is no certification body, no approval committee, and no veto power. The framework's authority derives from its usefulness. If the community finds it useful, it will be adopted. If not, it will be ignored. That is how governance works in a decentralized system.

6.5 "The legal analysis is speculative."

All legal analysis of novel situations is, to some degree, speculative. No court has ruled on the liability of a Bitcoin developer for a chain split caused by a reckless activation. But the absence of precedent does not mean the absence of risk. The legal principles applied in Section 4 — negligence, tortious interference, fiduciary duty — are well established. Their application to Bitcoin governance is novel but not unprecedented. Courts routinely apply existing legal frameworks to new technologies. The question is not whether these principles apply, but how. This paper offers an analysis, not a prediction.

6.6 "If the standards are not enforceable, what does the framework add?"

This objection has two forms. The strong form: a benign proposal would already meet the framework's criteria, and an ill-considered proposal will be resisted by the same stakeholders for the same reasons whether the framework exists or not — so the framework adds nothing operational. The weak form: the framework may legitimize rule-lawyering — proposals that satisfy the criteria in form but not in spirit.

The strong form misreads where the framework's value sits. The value is not in creating coordinated resistance to ill-considered proposals — that coordination exists, however imperfectly, in the form of Bitcoin's distributed governance. BIP-110's failure to attract more

than 0.15% of signaling demonstrates the existing coordination's capacity. The framework's value is in lowering the coordination cost of principled refusal. Without the framework, every individual stakeholder objection requires custom argumentation: an exchange operator who refuses to recognize a chain must justify the refusal from first principles, against the rhetorical pressure of proponents who can frame the refusal as arbitrary. With the framework, "this proposal fails criteria 1, 8, 11, and 14" is a complete justification: cheap to invoke, costly to rebut. Distributed enforcement works — BIP-110's failure is the existence proof — but it works at a cost. The framework lowers that cost without claiming authority it does not have.

This matters most at the margins, where stakeholders are uncertain whether their objection rises to the level of public dissent. A single skeptical infrastructure operator, holding the framework's criteria, can invoke it without first organizing a coalition. The framework converts what would otherwise be a coordination problem — every stakeholder waiting to see whether others will refuse — into a documentation problem: anyone can point to the criteria and say here is the standard; here is the gap. The credibility of the resulting refusal does not depend on the refuser's individual standing but on the criteria's defensibility.

The weak form — that the framework legitimizes rule-lawyering — assumes that a proposal which games the letter of the criteria thereby earns approval. It does not. Binary criteria and classification bands leave judgment intact; partial gaming is itself progress (a rule-lawyered proposal is at least better documented than an ad-hoc one); the framework can be revised adversarially as bad-faith actors expose loopholes; and §5.2's Red classification remains available for proposals that satisfy the letter but fail the spirit — which is itself observable evidence in the scoring record.

A more honest statement of the weak form is: any standard can be gamed. That is true of every standard humans have ever written. It is not a reason to write none; it is a reason to write standards that produce useful documentation of the gaming when it happens. This framework does that. BIP-110, evaluated against the framework, would score 3/20 — a documented Red. That documentation, available to any stakeholder, makes a future rule-lawyered variant easier to evaluate against than the original, not harder. The framework's effect is therefore cumulative: each evaluation makes the next one more efficient, because the prior scoring is in the record.

Chapter 6 in brief

Six common objections, and the framework's response to each:

- *"Bitcoin has no governance."* It has decentralized governance. The framework gives that process a shared vocabulary, not a central authority (§6.1).
- *"Anyone can run whatever software they want."* True. The framework does not restrict that. It informs the choice (§6.2).
- *"This would prevent necessary changes."* SegWit and Taproot would pass with high scores. The framework impedes reckless changes, not necessary ones (§6.3).
- *"Who decides whether the standards are met?"* Everyone, and no one. The framework is a tool, not an authority (§6.4).
- *"The legal analysis is speculative."* All novel-application legal analysis is. The principles applied are established; the application is new (§6.5).
- *"If the standards aren't enforceable, what does the framework add?"* It lowers the coordination cost of principled refusal — *"this proposal fails criteria 1, 8, 11, and 14"* is a complete justification, cheap to invoke and costly to rebut (§6.6).

Chapter 7

Conclusion

Bitcoin is the most consequential monetary experiment in human history. Its consensus rules govern the creation and transfer of value for millions of people and the storage of wealth measured in trillions of dollars. Changes to these rules should be evaluated with a rigor commensurate with their stakes.

The current system — in which proposals are evaluated ad hoc, activation mechanisms are invented on the fly, review periods range from weeks to years with no standard, and the community's only tools for evaluating proposals are Twitter threads and GitHub comments — is inadequate. It has produced near-catastrophic chain splits, wasted years of developer time on governance disputes, and created opportunities for reckless actors to push poorly considered changes to activation. BIP-110 stalled, but its failure to activate is a fact of community vigilance, not of structural protection. The next proposal of its kind will arrive on the same terms — no required review, no minimum code quality, no agreed-upon threshold — unless this gap is filled.

This framework does not solve the fundamental challenge of decentralized governance. No framework can. What it provides is a common vocabulary, a shared set of criteria, and a concrete checklist against which proposals can be evaluated. It shifts the burden of proof onto proponents of change — where it belongs — and provides the community with a structured way to say: this proposal is not ready.

The framework is licensed under Creative Commons Attribution 4.0 International. It may be freely shared, adapted, and built upon by anyone, for any purpose, with attribution. It is available on GitHub for community review and amendment. If it is useful, it will be used. If it can be improved, it should be improved. That is how Bitcoin works. That is how Bitcoin's governance should work too.

Chapter 7 in brief

The framework is a vocabulary, not an authority. Bitcoin has no enforcer; this paper does not claim to be one. The standards are offered as a shared benchmark: if the community finds them useful, they will be used. If they can be improved, they should be. Available under Creative Commons Attribution 4.0 — share, adapt, build on it. That is how Bitcoin works. That is how Bitcoin's governance should work too.

Asaf Fulks

Glossary of Technical Terms

Activation threshold.
The percentage of mining hashrate that must signal readiness for a soft-fork change before the change locks in and begins enforcement. Historically 95% (SegWit via BIP-9) and 90% (Taproot via Speedy Trial); BIP-110 proposed 55%. See §3.4.

***BCAP*.**
Shortened form for Ren Crypto Fish, Steve Lee & Lyn Alden, *Analyzing Bitcoin Consensus: Risks in Protocol Upgrades* (Nov. 2024). The principal prior analytical work this paper builds on. See §1.5.

BIP (Bitcoin Improvement Proposal).
A formal proposal document for a change to the Bitcoin protocol or surrounding standards. Filing a BIP does not activate the change; activation is a separate ecosystem process this framework evaluates.

Bitcoin Core.
The reference implementation of the Bitcoin protocol, maintained by the Bitcoin Core project. The codebase against which other implementations are typically measured for consensus compatibility.

Bitcoin Knots.
A node implementation derived from Bitcoin Core with additional policy options. The BIP-110 activation client is a fork of Bitcoin Knots; the v29.3.knots20260508 release bundled BIP-110 (RDTS) activation rules into the default release stream. See §1.3.

Chain split.
A divergence in the network where some nodes accept a block under one set of consensus rules and other nodes accept a different block under different rules, producing two parallel chains.

Consensus rules.
The network-wide rules that determine whether a block is valid. Changes to consensus rules require either a soft fork or a hard fork.

Economic Nodes.
Full nodes operated by entities with material Bitcoin throughput — exchanges, custodians, payment processors, large merchants, ETF operators. Their enforcement of consensus rules is what gives miner signaling its weight.

Hard fork.
A consensus-rule change that loosens or alters validity rules such that the new rules accept blocks the old rules would reject. Pre-upgrade nodes cannot follow the new chain. See §3.7.

Hashrate.
The aggregate computational power miners are dedicating to Bitcoin's proof-of-work. Used as the unit of measure for activation signaling.

Inscriptions.
Image, text, document, or binary files embedded in the witness data of Bitcoin transactions using Casey Rodarmor's Ordinals protocol. The protocol encodes data within an unexecuted `OP_FALSE OP_IF ...OP_ENDIF` envelope inside Taproot's script-path spend; Taproot's witness-data fee discount makes this storage substantially cheaper, per byte, than alternatives such as OP_RETURN. The proliferation of inscriptions since December 2022 is the motivating context for the wave of restrictive consensus-change proposals discussed in this paper. See §1.2.

MASF (Miner-Activated Soft Fork).
A soft fork that activates when miner signaling crosses the activation threshold. The default activation pathway in Bitcoin.

OP_RETURN.
A Bitcoin script opcode that marks a transaction output as provably unspendable, used to embed arbitrary data in a transaction. BIP-110 sought to restrict the size of OP_RETURN payloads alongside other forms of data embedding. See §1.3.

OP_SUCCESS.
Opcodes reserved during the Taproot soft fork as placeholders for future additions. Substituting an OP_SUCCESS slot with a new opcode enables adding new spending paths via soft fork. See §3.5.

Reference implementation.
The canonical software implementing a proposed consensus change, against which other client implementations are compared for correctness.

Reorganization (reorg).
When miners switch from one valid chain to a competing chain of greater accumulated work, undoing the transactions in the abandoned blocks. The mechanism by which a sub-overwhelming-enforcement soft fork can be unwound. See §3.4.

Replay protection.
A technical mechanism (typically a transaction-format change) that prevents a transaction valid on one chain after a split from being broadcastable on the other. Necessary in hard forks where a meaningful minority continues to run the prior rules. See §3.5.C, §3.7.

Signaling.
Miners indicating readiness for a proposed soft fork by setting a designated bit in block headers during a defined signaling window.

Soft fork.
A consensus-rule change that tightens validity rules such that the new rules continue to accept blocks the old rules would have accepted (the new rules are a strict subset of the old). Pre-upgrade nodes will continue to follow the upgraded chain.

Speedy Trial.
An activation deployment pattern with a fixed signaling window, a high activation threshold (commonly 90%), and a clean timeout if the signaling threshold is not met. The clean-timeout behavior is the deployment's LOT=false setting; the alternative LOT=true forces activation at the deadline regardless of signaling. See §3.4 on the LOT debate. Used for Taproot in 2021.

Sunset clause.
A provision causing a consensus rule change to automatically expire at a defined block height or median time past, returning the network to pre-activation rules without further intervention. A valid sunset is self-executing (no further software action required) and tested on testnet to confirm deactivation works correctly. See §3.6.

UASF (User-Activated Soft Fork).
A soft fork activated by economic nodes enforcing new rules without depending on miner signaling threshold being met. BIP-148 is the canonical example. UASF works only when prior community support is overwhelming; UASF without that support yields the BIP-110 outcome. See §3.4.

UTXO (Unspent Transaction Output).
The fundamental unit of Bitcoin ownership: a transaction output that has been received but not yet spent. The complete set of UTXOs at a given block height — the "UTXO set" — defines all spendable Bitcoin at that moment.

References

Bitcoin Improvement Proposals

- BIP-1: *BIP Purpose and Guidelines*. Amir Taaki, 2011. `github.com/bitcoin/bips/blob/master/bip-0001.mediawiki`
- BIP-2: *BIP Process, Revised*. Luke Dashjr, 2016. `github.com/bitcoin/bips/blob/master/bip-0002.mediawiki`
- BIP-8: *Version Bits with Lock-in by Height*. Shaolin Fry, Luke Dashjr, 2017. `github.com/bitcoin/bips/blob/master/bip-0008.mediawiki`
- BIP-9: *Version Bits with Timeout and Delay*. Pieter Wuille, Peter Todd, Greg Maxwell, Rusty Russell, 2015. `github.com/bitcoin/bips/blob/master/bip-0009.mediawiki`
- BIP-16: *Pay to Script Hash*. Gavin Andresen, 2012. `github.com/bitcoin/bips/blob/master/bip-0016.mediawiki`
- BIP-17: *OP_CHECKHASHVERIFY (CHV)*. Luke Dashjr, 2012. `github.com/bitcoin/bips/blob/master/bip-0017.mediawiki`
- BIP-50: *March 2013 Chain Fork Post-Mortem*. Gavin Andresen, 2013. `github.com/bitcoin/bips/blob/master/bip-0050.mediawiki`
- BIP-65: *OP_CHECKLOCKTIMEVERIFY*. Peter Todd, 2014; activated 2015. `github.com/bitcoin/bips/blob/master/bip-0065.mediawiki`
- BIP-68: *Relative lock-time using consensus-enforced sequence numbers*. Mark Friedenbach, BtcDrak, Nicolas Dorier, kinoshitajona, 2015. `github.com/bitcoin/bips/blob/master/bip-0068.mediawiki`
- BIP-91: *Reduced threshold Segwit MASF*. James Hilliard, 2017. `github.com/bitcoin/bips/blob/master/bip-0091.mediawiki`
- BIP-101: *Increase Maximum Block Size*. Gavin Andresen, 2015. `github.com/bitcoin/bips/blob/master/bip-0101.mediawiki`
- BIP-102: *Block Size Increase to 2MB*. Jeff Garzik, 2015. `github.com/bitcoin/bips/blob/master/bip-0102.mediawiki`
- BIP-110: *Reduced Data Temporary Softfork* (originally proposed as BIP-444). Dathon Ohm, 2025. `github.com/bitcoin/bips/blob/master/bip-0110.mediawiki`
- BIP-112: *CHECKSEQUENCEVERIFY*. BtcDrak, Mark Friedenbach, Eric Lombrozo, 2015. `github.com/bitcoin/bips/blob/master/bip-0112.mediawiki`

- BIP-113: *Median time-past as endpoint for lock-time calculations*. Thomas Kerin, Mark Friedenbach, 2015. `github.com/bitcoin/bips/blob/master/bip-0113.mediawiki`
- BIP-141: *Segregated Witness (Consensus Layer)*. Eric Lombrozo, Johnson Lau, Pieter Wuille, 2015. `github.com/bitcoin/bips/blob/master/bip-0141.mediawiki`
- BIP-148: *Mandatory Activation of Segwit Deployment*. Shaolin Fry, 2017. `github.com/bitcoin/bips/blob/master/bip-0148.mediawiki`
- BIP-340: *Schnorr Signatures for secp256k1*. Pieter Wuille, Jonas Nick, Tim Ruffing, 2020. `github.com/bitcoin/bips/blob/master/bip-0340.mediawiki`
- BIP-341: *Taproot: SegWit Version 1 Spending Rules*. Pieter Wuille, Jonas Nick, Anthony Towns, 2020. `github.com/bitcoin/bips/blob/master/bip-0341.mediawiki`
- BIP-342: *Validation of Taproot Scripts*. Pieter Wuille, Jonas Nick, Anthony Towns, 2020. `github.com/bitcoin/bips/blob/master/bip-0342.mediawiki`
- BIP-361: *Post Quantum Migration and Legacy Signature Sunset*. Jameson Lopp, Christian Papathanasiou, Ian Smith, Joe Ross, Steve Vaile, Pierre-Luc Dallaire-Demers, 2026. `github.com/bitcoin/bips/blob/master/bip-0361.mediawiki`

Legal Authorities

- *Aas v. Superior Court*, 24 Cal.4th 627 (2000).
- *Biakanja v. Irving*, 49 Cal.2d 647 (1958).
- *Della Penna v. Toyota Motor Sales, U.S.A., Inc.*, 11 Cal.4th 376 (1995).
- *Korea Supply Co. v. Lockheed Martin Corp.*, 29 Cal.4th 1134 (2003).
- *Pacific Gas & Electric Co. v. Bear Stearns & Co.*, 50 Cal.3d 1118 (1990).
- *Robinson Helicopter Co. v. Dana Corp.*, 34 Cal.4th 979 (2004).
- *Tulip Trading Ltd v van der Laan* [2023] EWCA Civ 83, Court of Appeal of England and Wales (3 February 2023). `judiciary.uk/wp-content/uploads/2023/02/Tulip-v-Van-Der-Laan-judgment-030223.pdf`
- Rev. Rul. 2019-24, 2019-44 I.R.B. 1004 (Oct. 9, 2019) (tax treatment of cryptocurrency hard forks). `irs.gov/pub/irs-drop/rr-19-24.pdf`
- Restatement (Second) of Torts § 552 (Am. Law Inst. 1977).
- Walch, Angela. "In Code(rs) We Trust: Software Developers as Fiduciaries in Public Blockchains." In *Regulating Blockchain: Techno-Social and Legal Challenges*, edited by Philipp Hacker, Ioannis Lianos, Georgios Dimitropoulos & Stefan Eich, 58–81. Oxford: Oxford University Press, 2019.

Comparative Common-Law and EU Authorities

Cited by name in §4.6's comparative note at doctrinal level; specific reporter citations omitted in keeping with the comparative scope. Practitioners requiring precise references should consult the cited courts' official databases (Bailii for UK, AustLII for Australia, CanLII for Canada, EUR-Lex for EU).

- *Caparo Industries plc v Dickman* (House of Lords, 1990). Leading modern UK duty-of-care case; three-stage test (foreseeability, proximity, fair-just-reasonable).
- *Hedley Byrne & Co Ltd v Heller & Partners Ltd* (House of Lords, 1964). Foundational UK case on negligent misstatement and assumption of responsibility.
- *Sullivan v Moody* (High Court of Australia, 2001). Australian duty-of-care case departing from the *Caparo* three-stage test.
- *Anns v Merton London Borough Council* (House of Lords, 1978). Predecessor two-stage UK duty-of-care test; overruled in the UK but retained in modified form in Canada.
- *Cooper v Hobart* (Supreme Court of Canada, 2001). Leading Canadian duty-of-care case refining the *Anns* two-stage approach.
- Directive on liability for defective products (European Union, adopted 2024), replacing Directive 85/374/EEC. Extends strict product liability to software. Member State transposition deadline approximately two years from entry into force.
- Swiss Code of Obligations (*Obligationenrecht*) Article 41. General civil-law tort liability clause.

Industry and Academic Analysis

- Crypto Fish, Ren, Steve Lee, and Lyn Alden. *Analyzing Bitcoin Consensus: Risks in Protocol Upgrades*. November 2024. `github.com/bitcoin-cap/bcap`; also available at `bitcoinnews.ch/wp-content/uploads/2024/11/bcap_v1.0.pdf`.
- Lopp, Jameson. *When Do Bitcoin Node Operators Upgrade?* `blog.lopp.net/when-do-bitcoin-node-operators-upgrade/`

Software Releases

- Bitcoin Knots v29.3.knots20260508 (RDTS-enabled default release). Bitcoin Knots Project, 9 May 2026. `bitcoinknots.org/files/29.x/`
- Bitcoin Knots v29.3.knots20260507 (final pre-RDTS release; non-RDTS variant). Bitcoin Knots Project, 8 May 2026. `bitcoinknots.org/files/29.x/`
- RDTS Activation Client v0.1rc1 (BIP-110 activation client, Bitcoin Knots fork). 10 December 2025. `github.com/bitcoinknots/bitcoin/releases`

License