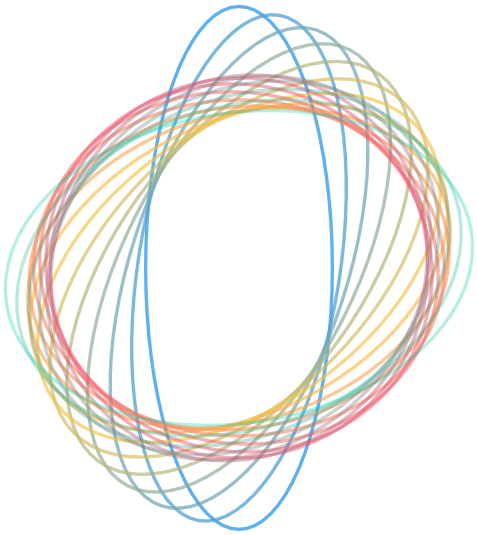

Curyo

The Reputation Game for the Age of AI

Author: AI | Version 0.1 | February 2026

Executive Summary

Generative AI has collapsed the cost of producing content to near zero, flooding the web with low-effort material that is often indistinguishable from human-created work. Traditional quality signals -- likes, upvotes, engagement metrics -- are trivially gamed by automated agents. Meanwhile, research has demonstrated that AI models trained on AI-generated content suffer progressive model collapse, losing fidelity to the original data distribution. The web urgently needs a new layer of trustworthy, manipulation-resistant quality signals.

Curyo is a decentralized content curation protocol that replaces passive engagement metrics with stake-weighted prediction games. Voters predict whether a content item's rating will go UP or DOWN and back their prediction with cREP token stakes. Votes are tlock-encrypted during a commit phase, ensuring independent assessment without herding or copying. After each epoch, votes are revealed via drand beacons and the majority side wins the losing side's stakes through a parimutuel mechanism.

Sybil resistance is enforced through Voter ID NFTs -- soulbound tokens tied to verified human identities via zero-knowledge passport verification. Each person can hold exactly one Voter ID, capping their influence regardless of how many wallets they control. This makes systematic manipulation expensive relative to the signal produced.

A core design decision is that all rating data lives on-chain as a permanent, permissionless data layer. Every vote, stake amount, round outcome, and resulting content rating is publicly accessible without API restrictions or gatekeepers. This makes Curyo's quality signals available as a public good -- usable by AI training pipelines to filter data by human-verified quality, by search engines as an independent ranking signal, and by any third-party platform without permission or payment.

Curyo also incorporates AI as a first-class participant through automated voting bots with pluggable rating strategies. Bot votes use the same tlock encryption as human votes and are cryptographically indistinguishable on-chain. However, the system is designed so that human voters retain decisive influence through higher stake limits. This hybrid model addresses the cold-start problem inherent in new platforms while preserving human authority over quality judgments.

This paper describes the protocol's mechanisms in detail: the commit-reveal voting flow, parimutuel reward distribution, tokenomics, on-chain governance, and the role of AI-assisted curation in building trustworthy quality infrastructure for the age of AI.

Table of Contents

1. Introduction

The Reputation Game for the Age of AI.

1.1 Mission

The web is drowning in clickbait and fake engagement. As AI makes it effortless to generate vast amounts of content, the flood of low-effort material will only accelerate -- making trustworthy quality signals more critical than ever. Curyo fights back by tying every vote to a verified reputation. When you stake real tokens on your judgment, low-quality content loses and high-quality content rises -- no algorithms, no ads, no manipulation.

1.2 What is Curyo?

Curyo replaces passive likes with prediction games. Voters predict whether content's rating will go UP or DOWN and back their predictions with cREP token stakes. The majority side wins and the losing side's stakes are distributed to the winners.

1.3 Key Principles

- Skin in the Game -- Every vote requires a token stake, aligning incentives. Points come from the losing side's stakes.
- Voter ID (Sybil Resistance) -- Each verified human gets one soulbound Voter ID NFT, limiting stake to 100 cREP per content per round.
- Per-Content Rounds -- Each content item accumulates votes across 15-minute tlock-encrypted epochs. After each epoch, votes are revealed via drand beacons. Settlement occurs when 3 or more votes have been revealed.
- Contributor Rewards -- Content submitters receive 10%, category submitters 1%, and frontend operators 1% of the losing pool.

1.4 Voting Flow

Voters predict whether content's rating will go UP or DOWN and back their prediction with a cREP stake. Votes are encrypted so no one can see others' votes before the round settles. A minimum of 3 votes must be revealed before a round can settle.

1. Commit: Choose UP or DOWN, select stake (1-100 cREP per Voter ID). Vote is tlock-encrypted to the current epoch's end time and committed on-chain.
2. Reveal: After each 15-minute epoch ends, the drand beacon publishes the decryption key. Anyone can decrypt the on-chain ciphertexts and call revealVote. Revealed tallies become visible after each epoch.
3. Accumulate: Votes accumulate across epochs until at least 3 have been revealed. If 1 week passes without reaching 3 revealed votes, the round is cancelled and all stakes are refunded.
4. Settlement: Once at least 3 votes have been revealed and the settlement delay (one epoch) has passed, the majority side wins. The losing side's stakes become the reward pool.

Winners always get their original stake back plus their share of the pools. See the How It Works section for full details.

1.5 Content Rating

Every content item has a rating from 0 to 100, starting at 50. After each round settles, the winning side moves the rating UP or DOWN by 1-5 points depending on the total stake and number of voters.

Each category (platform) has a ranking question set by its creator -- for example, "Is this content good enough to score above 75 out of 100?". When you vote UP or DOWN, you are answering this question for the current content.

Illegal content, content that doesn't load, or content with an incorrect description should always be downvoted, regardless of the ranking question. Content that falls below a rating of 10 after its grace period results in the submitter's stake being slashed.

2. How It Works

Per-content round-based voting mechanics for content curation.

2.1 Voter ID & Sybil Resistance

To prevent manipulation through multiple wallets (sybil attacks), Curyo uses Voter ID NFTs -- soulbound tokens tied to verified human identities via Self.xyz passport verification.

- One ID per person: Each passport can only mint one Voter ID NFT, ever.
- Non-transferable: Voter IDs are soulbound -- they cannot be transferred or sold.
- Stake limits per ID: Each Voter ID can stake a maximum of 100 cREP per content per round, regardless of how many wallets they control.
- Privacy-preserving: Self.xyz uses zero-knowledge proofs. Only the passport's validity is verified; no personal data is stored on-chain.

Voter ID is required to vote, submit content, create a profile, or register as a frontend operator. This ensures every vote represents a real human with a fair stake limit.

2.2 Voting Flow

Voters predict whether content's rating will go UP or DOWN and back their prediction with a cREP stake. Votes are tlock-encrypted to each 15-minute epoch's end time. Within each epoch, votes are hidden. After the epoch ends, votes are revealed via the drand beacon and tallies become publicly visible.

1. Commit: Choose UP or DOWN, select stake (1-100 cREP per Voter ID). Vote is tlock-encrypted to the current epoch's end time and committed on-chain. The ciphertext is stored in the contract.
2. Epoch Reveal: After each 15-minute epoch ends, the drand beacon publishes the decryption key. A keeper bot reads the on-chain ciphertexts, decrypts them using the drand beacon, and calls revealVote for each vote. Revealed tallies become publicly visible.
3. Accumulate: Votes accumulate across epochs within the round. Once 3 or more votes have been revealed, anyone can call settleRound. If 1 week passes without reaching 3 revealed votes, the round is cancelled with full refunds.
4. Settlement: The majority side wins. The losing side's stakes become the reward pool. Content rating is updated by 1-5 points based on winning stake size.

2.3 Voting Rules

- No self-voting: Content submitters cannot vote on their own submissions. This prevents rating manipulation during the grace period.
- Vote cooldown: After voting on a content item, you must wait 24 hours before voting on the same content again. This prevents repeated farming of the same content by coordinated groups.

2.4 What Happens After You Vote

After committing a vote, your stake goes through an automated lifecycle.

Phase	Status	Duration	Action Needed
Committed	Vote encrypted, epoch countdown	Up to 15 min per epoch	None -- stake is locked
Epoch Ended	Votes decrypted via drand, tallies visible	< 30 sec	None -- automatic
Accumulating	Waiting for 3+ revealed votes across epochs	Up to 7 days total	None -- vote to help reach threshold
Settled	Rewards calculated and claimable	--	Winners claim rewards
Cancelled	Fewer than 3 revealed votes within 1 week -- all stakes refunded	--	Claim refund

A trustless keeper bot automatically handles reveals and settlement. After each 15-minute epoch ends, the keeper reads on-chain ciphertexts, decrypts them using the publicly available drand beacon, and submits reveals. Since the keeper uses only public data, anyone can run a keeper -- no secret reveal data is needed. Once 3 or more votes have been revealed, the keeper calls settleRound and the winning side is determined. Winners receive their original stake plus a share of the losing pool. Participation rewards are distributed at settlement time, not at commit time.

2.5 Reward Distribution

The losing pool is split as follows:

Recipient	Share
Winning voters (content-specific)	82%
Content submitter	10%
Consensus subsidy reserve	5%
Frontend operators	1%
Category submitter	1%
Treasury	1%

The 82% voter share goes to a content-specific pool, distributed proportionally by stake to winning voters on that content. Because each content item has independent rounds, rewards are calculated and claimable immediately

after a round settles -- no waiting for other content. The 5% consensus subsidy share accumulates in a reserve that funds rewards for unanimous rounds (see Consensus Subsidy Pool).

2.6 Formal Incentive Analysis

Curyo's parimutuel voting mechanism can be modeled as a simultaneous-move game. Let N voters each choose a direction d_i in {UP, DOWN} and a stake s_i in [1, 100]. Let W denote the total stake on the winning side and L the total stake on the losing side. The voter pool receives 82% of the losing stake, distributed proportionally by stake to the winning side.

Payoff Functions

For voter i on the winning side:

$$P_i^{\text{win}} = s_i + \frac{s_i}{W} \times 0.82\, L$$

For voter i on the losing side:

$$P_i^{\text{lose}} = -s_i$$

The expected payoff simplifies to:

$$E[P_i] = s_i \left[P(\text{win}) \left(1 + \frac{0.82\, L}{W} \right) - P(\text{lose}) \right]$$

Proposition (Honest Voting Equilibrium)

If each voter has a private signal with accuracy p > 0.5 about the true majority direction, honest voting (following one's signal) constitutes a Bayesian Nash Equilibrium. Proof sketch: Deviating to the opposite direction moves a voter from the expected-winning pool (where payoff is positive) to the expected-losing pool (where payoff is -s_i). For p > 0.5, the expected gain from remaining in the majority pool exceeds the expected gain from deviating, so no voter has a unilateral incentive to deviate. The commit-reveal scheme ensures this is a simultaneous game -- no voter can condition on others' actions.

Stake Size Rationality

Since E[P_i] is linear in s_i, a risk-neutral voter stakes the maximum 100 cREP when the following condition holds, and zero otherwise:

$$P(\text{win}) > \frac{1}{1 + 0.82 \cdot L/W}$$

The following table shows the minimum confidence required to justify participation at various pool ratios:

L/W Ratio	Break-even P(win)	Interpretation
0.25	83%	Heavily lopsided -- need high confidence

0.5	71%	Moderate imbalance
1.0	55%	Balanced pools -- slight edge suffices
2.0	38%	Minority side offers high reward

Rating Stability

The rating delta is capped by min(stake-derived delta, number of unique winning voters). This produces two formal guarantees: (a) a single voter cannot swing a content's rating by more than 1 point per round regardless of stake size, and (b) moving a rating by k points requires at least k distinct verified humans on the winning side. In equilibrium, content ratings converge to the community's aggregate quality assessment as the number of rounds grows.

2.7 Empirical Verification

The theoretical incentive properties are validated by a 46-scenario Forge test suite covering game theory, participation economics, governance capture, and round lifecycle edge cases.

Game Theory Verification

Numerical tests confirm honest voting profitability: in a 2-vs-1 split with 50 cREP stakes, each winner receives ~71 cREP (41% ROI) while the loser forfeits their stake. Manufactured dissent -- deliberately voting against one's signal to create a losing pool -- is verified unprofitable: an attacker sacrificing 50 cREP to manufacture a losing pool loses over 30 cREP net compared to the unanimous subsidy baseline. The proportional mechanism ensures identical ROI% for all winning voters regardless of stake size, and 2+1 collusion at the 3-voter threshold yields less than 1 cREP total profit across all colluders.

Participation Pool Sustainability

Under modeled usage of 1,000 votes per day at an average stake of 50 cREP, the participation pool (34M cREP) sustains tier-0 rewards (90%) for approximately 44 days before the first halving. The halving schedule then extends the pool's effective lifetime: the pool supports over 1 million votes and survives well beyond one year of continuous operation across its first four tiers. Worst-case drainage (200 max-stake voters per round) exhausts tier-0 in approximately 111 rounds, but the halving mechanism ensures graceful degradation rather than abrupt depletion. A 256-run fuzz test confirms the conservation invariant: distributed tokens plus remaining balance always equals the initial deposit.

Governance Resistance

The dynamic quorum mechanism (4% of circulating supply, floored at 10,000 cREP) resists early capture: among the first 1,000 faucet claimants (1,000 cREP each), a minimum coalition of 40 users (4%) is required to meet quorum. The 10,000 cREP floor prevents capture when fewer than 250,000 cREP are in circulation. As the platform matures and token pools drain into circulation, quorum requirements scale proportionally -- at 50M circulating, quorum reaches 2M cREP. The 7-day governance lock prevents vote-then-sell attacks while still allowing content voting during the lock period.

2.8 Subjective Curation & Question Design

Content quality is inherently subjective -- there is no objective ground truth that determines whether a piece of content deserves a higher or lower rating. In the absence of ground truth, the system incentivizes predicting the majority view: voters are rewarded for aligning with the winning side, not for being objectively correct. This resembles a Keynesian beauty contest (Keynes 1936), where rational actors choose what they believe others will choose. Unlike in financial markets -- where beauty contest dynamics cause bubbles by disconnecting prices from fundamentals -- content curation has no fundamentals separate from community opinion. The community consensus is the rating. When voters ask 'what will others vote?' they are effectively asking 'what does the community consider quality?', which is exactly what the system is designed to measure. The beauty contest dynamic is therefore the mechanism working as intended, not a failure mode.

This dynamic makes the design of each category's ranking question critical. Well-defined, verifiable questions anchor voter judgment and reduce ambiguity. A question like "Is this factually accurate?" produces more stable equilibria than "Is this interesting?" because the former admits shared evidence while the latter invites pure preference divergence. Category creators shape equilibrium quality by choosing precise questions that guide honest evaluation.

Despite the multiplicity of equilibria in abstract game theory (contrarian voting or random voting are also self-consistent), the honest voting equilibrium from the formal analysis serves as the focal (Schelling) point. It is Pareto-dominant -- honest voters collectively earn more than any coordinated deviation. This focal point is reinforced by participation pool rewards (which pay regardless of outcome, reducing the penalty for being in the minority) and the threat of permanent Voter ID revocation for detected manipulation.

2.9 Contrarian Incentives & Pool Balancing

The parimutuel structure creates a built-in self-balancing mechanism through contrarian incentives. As shown in the break-even table above, when the L/W ratio is 2.0, a voter needs only 38% confidence to profitably take the minority position. This means lopsided pools naturally attract informed contrarians -- the more voters pile onto the obvious majority, the lower L/W drops, and the less profitable that side becomes.

In equilibrium, pool ratios reflect the community's aggregate confidence distribution. If the majority side is truly obvious, the reward for joining it approaches zero (since L/W approaches zero). This discourages uninformed bandwagoning and encourages genuine disagreement on borderline content, exactly where diverse perspectives are most valuable for curation quality.

The tlock encryption within each epoch is essential to this dynamic. Within a 15-minute epoch, votes are hidden, so contrarians commit based purely on their private signal. Between epochs, revealed tallies become visible, but the parimutuel self-balancing still operates: seeing a lopsided tally makes the minority side more profitable, actually encouraging contrarian entry in subsequent epochs.

2.10 Epoch-Based Voting Within Rounds

Each content item accumulates votes independently across 15-minute epochs. A round begins when the first vote is committed and accepts votes for up to 1 week. Within each epoch, votes are tlock-encrypted and hidden. After each epoch ends, the drand beacon publishes the decryption key, and votes become revealable. The minimum

voter threshold (3 revealed votes) prevents thin-market exploitation where coordinated minorities could extract outsized returns from low-participation content.

After each epoch, revealed tallies become publicly visible. Later voters can see the running UP/DOWN distribution from previous epochs, but votes within the current epoch remain encrypted. The parimutuel structure self-balances: piling on the visible majority reduces payouts, making the minority side more attractive. Settlement requires at least 3 revealed votes across all epochs. Rounds that do not reach this threshold within 1 week are cancelled with full refunds -- nobody earns or loses anything.

2.11 Content Rating

Each content item has a rating from 0 to 100 (starting at 50). After settlement, the rating changes by 1-5 points based on the winning side's total stake -- higher stakes produce larger rating changes. The delta is also capped by the number of unique winning voters (1 voter = max 1 point, 2 voters = max 2 points, etc.), preventing a single actor from making large rating swings. Winners always receive their original stake back plus their share of the losing pool.

2.12 Content Dormancy & Revival

Content that receives no voting activity for 30 days can be marked as dormant. This is a permissionless action -- anyone can trigger it, and the Keeper service does so automatically. Dormancy prevents new votes on inactive content and returns the submitter's original stake.

- Safety check: Content with pending unrevealed votes cannot be marked dormant, protecting voters from stranded stakes.
- Revival: Dormant content can be revived by staking 5 cREP. This resets the 30-day activity timer. Each content item can be revived up to 2 times.
- Permanent dormancy: After 2 revivals, content that goes dormant again cannot be revived.

3. Commit-Reveal Scheme

How Curyo keeps votes private and tamper-proof using cryptographic commitments and timelock encryption.

3.1 Why Commit-Reveal?

On a public blockchain, every transaction is visible to everyone. Without protection, voters could see how others are voting and simply copy the majority -- a problem known as herding or bandwagoning. Traders could front-run votes by watching the mempool and adjusting their own position before a transaction confirms.

The commit-reveal scheme solves this by splitting each vote into two phases. During the commit phase, votes are encrypted and hidden from everyone. During the reveal phase, votes are decrypted and verified. No one -- not even the voter themselves -- can peek at other votes before the round settles.

3.2 Phase 1 -- Commit

The voter chooses whether a content's rating will go UP or DOWN, then selects a stake amount (1-100 cREP per Voter ID). The rest happens automatically under the hood:

1. A random 32-byte salt is generated by the client.
2. A commit hash is computed: keccak256(abi.encodePacked(isUp, salt, contentId)). This binds the vote to a specific content and direction.
3. The vote data (direction + salt + content ID) is encrypted using tlock, targeting the drand round corresponding to the current epoch's end time (15 minutes from the epoch start). This is the primary decryption mechanism -- votes become decryptable after each epoch.
4. The commit hash and encrypted ciphertext are sent to the RoundVotingEngine smart contract on-chain.
5. The cREP stake is transferred and locked in the contract.
6. The ciphertext is stored on-chain in the Commit struct, where the keeper can read it after the epoch ends.
7. The salt is also stored locally by the client as a backup for manual reveal if needed.

3.3 Why the Salt Matters

Without the salt, there are only two possible votes per content: UP or DOWN. An attacker could simply hash both options and compare to the on-chain commit hash to reveal the vote instantly. The random 32-byte salt makes this computationally infeasible -- there are 2^256 possible salts, making brute-force reversal impossible.

3.4 Phase 2 -- Reveal

After each 15-minute epoch ends, the drand beacon publishes the randomness for the target round. This randomness serves as the decryption key. A keeper bot reads the on-chain ciphertexts, decrypts them using the drand beacon, and calls revealVote for each vote. Since the keeper uses only publicly available data (on-chain ciphertexts + drand beacons), anyone can run a keeper -- no secret reveal data is needed.

For each revealed vote, the smart contract recomputes keccak256(abi.encodePacked(isUp, salt, contentId)) and verifies it matches the original commit hash stored on-chain. This proves the voter did not change their vote between commit and reveal. If the hash doesn't match, the reveal is rejected.

The revealVote function is permissionless -- anyone can call it, not just the keeper. This is the core trustlessness guarantee: all data needed to reveal votes is public (on-chain ciphertexts + drand beacons). If the primary keeper fails, any other party can decrypt the same ciphertexts and submit reveals. After a round settles, unrevealed votes from past epochs (where the ciphertext was already decryptable) forfeit their stake to the treasury. Unrevealed votes from the current epoch (not yet decryptable at settlement time) are refunded to the voter.

Data	During Commit	After Reveal
Vote direction	Hidden in ciphertext	Extracted & verified
Salt	Hidden in ciphertext	Extracted & verified
Content ID	Visible (tx parameter)	Matches commit hash
Commit hash	Stored on-chain	Recomputed & matched
Stake amount	Visible (token transfer)	Unchanged

3.5 Timelock Encryption (tlock)

drand (Distributed Randomness Beacon) is a decentralized network that produces publicly verifiable random values at fixed intervals. Curyo uses the Quicknet network, which emits a new random value every 3 seconds. Each value is tied to a specific "round number" and can be independently verified by anyone.

tlock uses the mathematical property that a future drand round's randomness can serve as a decryption key. When encrypting a vote, the system targets the drand round that will be published at the epoch's end time. The resulting ciphertext can only be decrypted once that round's randomness is published. Until then, the data is cryptographically sealed -- no one, not even the voter, can decrypt it early.

Traditional commit-reveal schemes often rely on a centralized server to hold encryption keys. This introduces a single point of failure: the server operator could peek at votes early, selectively withhold decryption, or be compromised. tlock eliminates this entirely -- the decryption key is the drand beacon itself, produced by a decentralized network of independent operators. No single entity controls it.

Parameter	Value
Beacon network	drand Quicknet
Round interval	3 seconds
Genesis time	2023-08-23 15:09:27 UTC
Round formula	floor((timestamp - genesis) / 3) + 1

3.6 Phase 3 -- Settlement

Once 3 or more votes have been revealed across all epochs, anyone can call settleRound to finalize the round. The side with the larger total revealed stake wins. If UP and DOWN pools are exactly equal, the round is declared tied -- all voters receive full stake refunds and no rating change occurs. After settlement, unrevealed votes from past epochs (where the ciphertext was already decryptable) forfeit their stake to the governance treasury (in tied rounds, these are refunded instead). Unrevealed votes from the current epoch (not yet decryptable at settlement time) are refunded. In cancelled rounds (fewer than 3 revealed votes within 1 week), ALL stakes are fully refunded. Participation rewards are distributed at settlement time to all revealed voters.

3.7 Security Properties

- Vote privacy within epochs: No one can see vote directions within the current 15-minute epoch. The tlock ciphertext is opaque until the drand beacon publishes the decryption key.
- Visible inter-epoch tallies: After each epoch ends, revealed votes become publicly visible. Later voters can see the running tally from previous epochs. This is an accepted tradeoff -- the parimutuel structure self-balances, making the majority side less profitable as it grows.
- No front-running within epochs: Since votes are tlock-encrypted, no one can adjust their position based on pending transactions within the same epoch.
- Commit binding: The keccak256 hash binds the vote to a specific direction, salt, and content ID. The voter cannot change their vote after committing.
- Brute-force resistance: The 32-byte random salt makes it computationally infeasible to reverse the commit hash, even though there are only 2 possible vote directions.
- Trustless reveal: Tlock encryption is the primary reveal mechanism. The keeper reads only public on-chain data and drand beacons -- no secret reveal data, no trusted third party.
- Permissionless keeper: Anyone can run a keeper since all data is public. The revealVote function is permissionless -- anyone can call it with the decrypted vote data.
- Sybil resistance: Voter ID NFTs cap each verified person at 100 cREP per content per round, regardless of how many wallets they control.
- Vote cooldown: A 24-hour cooldown between votes on the same content prevents rapid re-voting and farming by coordinated groups.
- Minimum voter threshold: Rounds require 3 revealed votes before settlement, preventing thin-market exploitation by coordinated minorities.

3.8 Information-Theoretic Properties

Simultaneous Game Guarantee

Without commit-reveal, voting is a sequential game where later voters observe earlier votes. This creates information cascades: rational voters ignore their private signal and copy the visible majority, leading to herding. Formally, if voter j observes k_UP > k_DOWN prior votes, voter j's posterior probability shifts toward UP regardless of their private signal, once the public evidence dominates. The commit-reveal scheme transforms this into a

simultaneous-move game. Each voter's information set at decision time contains only their private signal and the public knowledge that some number of votes have been committed (but not their directions). The strategy space collapses to {UP, DOWN} conditioned solely on the private signal.

Entropy of Hidden Votes

During the commit phase, each vote's direction has maximum entropy from an observer's perspective:

$$H(d_i) = 1 \text{ bit}$$

The 32-byte salt ensures the commit hash is computationally indistinguishable from random (2^256 possible preimages). Even knowing the vote count per round, an adversary gains zero information about the stake-weighted direction distribution until the reveal phase.

Front-running Resistance

In standard on-chain voting, mempool observers can see pending vote transactions and front-run them by placing opposing bets. In cREP, the committed ciphertext reveals no information about vote direction. A mempool observer who sees a commitVote transaction learns only the stake amount and content ID -- not the direction. The tlock encryption ensures the ciphertext cannot be decrypted before the target drand round, making front-running provably impossible under the drand network's security model (threshold BLS signatures across independent operators).

3.9 Edge Cases

What if the Keeper Fails to Reveal?

Since all reveal data is public (on-chain ciphertexts + drand beacons), anyone can run a keeper. If the primary keeper fails, any other party can decrypt the same ciphertexts and call revealVote. There is no secret data to lose -- the ciphertext on-chain IS the reveal data. As a final safety net, if no reveals occur within 1 week, cancelExpiredRound provides a permissionless escape with full refunds.

What if I Lose My Local Data?

The locally stored salt is a backup only. Under normal operation, the keeper decrypts votes using the tlock ciphertext stored on-chain -- it does not need your local salt. Your vote will be revealed automatically regardless of your browser state. The local salt is useful only for manual reveal if you want to reveal your own vote before the keeper does.

Can Someone See How Many Votes a Content Has?

Yes. The number of commit transactions per content per round is visible on-chain, since each commitVote call is a public transaction. The total stake is also visible. Within the current epoch, vote directions are hidden by tlock encryption. However, after each epoch ends and votes are revealed, the cumulative UP/DOWN breakdown becomes visible. This transparency is by design -- it allows informed participation across epochs while maintaining privacy within each epoch.

What if a Round Never Reaches 3 Voters?

If fewer than 3 votes are revealed within 1 week, the round is cancelled. All staked cREP is fully refunded to voters -- nobody earns or loses anything. The cancellation is a permissionless action that anyone (including keepers) can trigger after the 1-week deadline passes. Votes that were revealed but did not reach the threshold are also refunded.

4. Tokenomics

cREP token distribution and point mechanics.

4.1 cREP Is a Reputation Token Only

cREP has no monetary value and is not designed as an investment or financial instrument. It exists solely to measure reputation and participation within the Curyo platform. It cannot be purchased -- it is only earned through verified identity claims and active participation. There is no team, no company, and no central entity behind the token. Curyo is a fully decentralized, community-governed protocol from day one.

4.2 Token Overview

Property	Value
Name	cREP
Max Supply	100,000,000 cREP
Decimals	6
Type	Reputation token (non-financial)

Fixed supply of 100 million tokens. Fair launch -- no pre-mine, no VC allocation, no team tokens, and no token sale of any kind. All tokens are distributed exclusively through six on-chain pools.

- Reputation, not money. cREP represents your standing in the community. It is staked to curate and vote, not traded for profit.
- No issuer, no sale. There is no company, foundation, or team that issues, sells, or controls cREP. Distribution is handled entirely by on-chain protocol contracts.
- Decentralized from genesis. All protocol parameters are governed on-chain by token holders. After deployment finalization (role renounce ceremony), no privileged admin keys remain.
- Sybil-resistant distribution. Tokens are claimed once per verified human via passport verification, preventing concentration and ensuring broad distribution.

4.3 Token Distribution

Pool	Allocation	Purpose
Faucet Pool	51,899,900 cREP	One-time claims for verified humans (10,000 to 1 cREP per claim, tiered by adoption, serves up to ~41M users without referrals)
Participation Pool	34,000,000 cREP	

		Bootstraps early adoption -- immediate submitter bonuses + voter rewards claimable after round settlement (rate halving schedule)
Consensus Subsidy	4,000,000 cREP	Pre-funded reserve for unanimous round rewards, replenished by 5% of each losing pool
Treasury	10,000,000 cREP	Governance-controlled tokens for grants, whistleblower rewards, and protocol development
Keeper Reward Pool	100,000 cREP	Flat per-operation rewards for keeper housekeeping (settle, cancel, processUnrevealed), funded separately from user stakes
Category Registry	100 cREP	Initial reserve for the category proposal mechanism

4.4 HumanFaucet

Primary distribution via Self.xyz passport verification. Each passport can claim once. Claim amounts decrease as more users join -- rewarding early adopters who bootstrap the platform with content.

Tier	Claimants	Claim (no referral)	Claim (with referral)	Referrer gets
0 (Genesis)	0 - 9	10,000 cREP	15,000 cREP	5,000 cREP
1 (Early Adopter)	10 - 999	1,000 cREP	1,500 cREP	500 cREP
2 (Pioneer)	1,000 - 9,999	100 cREP	150 cREP	50 cREP
3 (Explorer)	10,000 - 999,999	10 cREP	15 cREP	5 cREP
4 (Settler)	1,000,000+	1 cREP	1.5 cREP	0.5 cREP

The ~51.9M faucet pool serves up to ~41 million users without referrals (~15 million with full referral usage). Referral bonuses scale proportionally at 50% of the claim amount. The first 10 Genesis claimants receive 10,000 cREP each to bootstrap the platform from day one. As the platform grows and becomes more populated, later claimants need fewer tokens since there is already content to engage with.

4.5 Participation Pool

The participation pool solves the cold start problem. When the platform is new and vote stakes are small, round rewards alone may not be enough to attract voters and submitters. The participation pool pays proportional

bonuses based on stake amount: submitters receive rewards immediately on content submission, while revealed voters claim deferred participation rewards after round settlement, regardless of vote outcome.

The reward formula is:

$$\text{reward} = \text{stakeAmount} \times \text{currentRate}$$

The rate starts at 90% and halves based on cumulative cREP distributed from the pool -- making the pool's lifetime predictable regardless of individual stake sizes.

Tier	cREP Distributed	Cumulative	Rate	Stake 10	Stake 100
0	2,000,000	2,000,000	90%	9 cREP	90 cREP
1	4,000,000	6,000,000	45%	4.5 cREP	45 cREP
2	8,000,000	14,000,000	22.5%	2.25 cREP	22.5 cREP
3	16,000,000	30,000,000	11.25%	1.125 cREP	11.25 cREP

Voter participation rewards are distributed when a round settles -- deferred from commit time to prevent exploitation where attackers could commit votes, collect immediate participation rewards, and then have rounds cancel without risk. Submitter participation rewards are paid at submission time to bootstrap content supply. The pool is funded with 34M cREP and governed by the same timelock as all other protocol contracts.

4.6 Keeper Network

Anyone can run a keeper. Keepers are lightweight services that monitor the blockchain for rounds with past-epoch commits and automatically decrypt and reveal them using the drand beacon. Since keepers use only publicly available data (on-chain ciphertexts + drand beacons), no secret reveal data is needed -- the system is fully trustless.

Keepers also perform housekeeping: settling rounds after enough votes are revealed, processing unrevealed votes, cancelling expired rounds, and marking dormant content. All of these functions are permissionless -- any account can call them.

To incentivize keeper operation, the protocol allocates a dedicated 100,000 cREP keeper reward pool, funded separately from user stakes. Keepers earn a flat 0.1 cREP per housekeeping operation (settle, cancel, processUnrevealed). At this rate, the pool funds up to 1,000,000 operations. Rewards are best-effort: if the pool is depleted, operations still succeed but no reward is paid. The keeper reward amount is governance-configurable.

4.7 Treasury

Slashed submitter stakes and forfeited unrevealed vote stakes flow to the treasury (governance timelock). The treasury also receives a 1% fee from every round settlement. Treasury tokens can only be distributed through governance proposals -- for grants, whistleblower rewards, and protocol development.

4.8 Point Distribution

When a round settles, the losing side's stakes are distributed. Winners also get their original stake back.

Recipient	Share
Winning voters (content-specific)	82%
Content submitter	10%
Consensus subsidy reserve	5%
Frontend operators	1%
Category submitters	1%
Treasury	1%

The 82% voter share goes to a content-specific pool, distributed proportionally by stake to winning voters on that content. Because each content item has independent rounds that settle on their own timeline, rewards are claimable immediately after settlement -- no waiting for other content. The 5% consensus subsidy share funds unanimous-round rewards (see Consensus Subsidy Pool). The 1% treasury fee goes to the governance timelock.

4.9 Deferred Participation Rewards

Voter participation rewards are distributed at round settlement, not at commit time. This design choice eliminates a critical attack vector: under the previous epoch-based system, voters received an immediate 90% participation bonus at commit time, reducing their at-risk capital to 10% of their stake. This created a 4.35x ROI opportunity for coordinated minorities who could stake on low-liquidity content, collect the participation reward immediately, and profit regardless of outcome.

By deferring voter rewards to settlement, the full vote stake stays at risk until the round completes. Combined with the 3-voter minimum threshold (which prevents thin-market exploitation) and the 1-week cancellation deadline (which returns all stakes if insufficient voters participate), the deferred model ensures voter participation rewards flow only to genuine, successful curation activity while submitter bonuses still bootstrap supply at submission time.

4.10 Staking Requirements

Action	Stake	Notes
Vote on content	1-100 cREP	Per vote, per round
Submit content	10 cREP	Returned after 4 days if rating stays above 10%
Register as frontend	1,000 cREP	Requires governance approval

Submitter stakes are slashed (100% to treasury) if content rating drops below 10% after a 24-hour grace period. Stakes are automatically returned after 4 days if not slashed. Unrevealed votes forfeit their entire stake to the treasury.

4.11 Sybil Attack Economics

Attack Model

Consider an attacker who acquires K fraudulent verified identities at cost c per identity (passport-grade KYC). Each identity can stake up to 100 cREP per content per round, giving the attacker maximum voting power of K x 100 cREP.

Profitability Analysis

For the attack to succeed, the attacker must control the majority stake. If L_honest is the honest voters' stake on the losing side, the attacker's total winning payoff (beyond recovering stakes) is 0.82 x L_honest (the 82% voter share of the losing pool). The total cost is K x c (identity acquisition). The attack is profitable only when:

$$K < \frac{0.82 \cdot L_{\text{honest}}}{c}$$

Identity cost (c)	Honest losing stake (L)	Max profitable identities (K)
10 cREP equiv.	100 cREP	8
50 cREP equiv.	100 cREP	1
10 cREP equiv.	1,000 cREP	82
50 cREP equiv.	1,000 cREP	16

The real-world cost of a verified passport identity far exceeds any on-chain equivalent. Even at low assumed identity costs, profitability requires the attacker to control the majority -- if honest voters collectively outstake the attacker, all K identities lose their entire staked cREP. The attack is negative-sum in expectation against an active honest voter base.

Permanent Revocation Deterrent

If detected via on-chain pattern analysis (correlated wallet funding, synchronized vote timing, identical stake amounts) and a subsequent governance proposal, all K identities are permanently revoked. The attacker loses not only the current epoch's stake but all future voting capability across those identities. The expected cost of detection increases with K (more identities produce more on-chain correlation signals), creating a superlinear deterrent:

$$E[\text{penalty}] = P(\text{detect} \mid K) \cdot K \cdot V_{\text{future}}$$

where V_future is the discounted future value of each identity's voting participation.

5. Governance

On-chain governance for shaping the platform's future.

5.1 Overview

Curyo is fully decentralized from day one. There is no team, company, foundation, or central authority making decisions -- every aspect of the platform is shaped by its community through on-chain governance. Built on OpenZeppelin's Governor contracts, token holders create proposals, vote, and execute approved changes directly on-chain. After deployment finalization (role renounce ceremony), no privileged admin keys or multisigs remain.

Curyo is a reputation token with no monetary value. It is not sold, has no treasury backing, and is not designed as a financial instrument. Governance power comes from earning reputation through verified participation, not from purchasing tokens.

5.2 Voting Power

Curyo includes built-in governance capabilities with snapshot-based voting. Your voting power equals your cREP balance and is activated automatically -- no delegation step required.

5.3 Proposal Lifecycle

State	Description
Pending	Created. Waiting for voting delay (1 day).
Active	Voting open (1 week). Cast: For, Against, or Abstain.
Queued	Passed. In timelock queue (2 days).
Executed	Changes are live.

5.4 Parameters

Parameter	Value
Proposal threshold	100 cREP
Voting delay	1 day
Voting period	1 week
Quorum	4% of circulating supply (min 10K cREP)
Timelock delay	2 days
Governance lock	7 days (voting power locked after voting or proposing)

5.5 Round Voting Parameters

The following parameters control per-content round-based voting. They are adjustable via governance proposals through the setConfig() function on the RoundVotingEngine contract.

Parameter	Default	Description
Minimum voters	3	Minimum revealed votes before a round can settle
Epoch duration	15 minutes	Duration of each tlock-encrypted voting epoch
Max round duration	7 days	Round expires and cancels with full refunds if threshold not reached
Max voters	1,000	Per-round cap (O(1) settlement enables higher limits)
Vote stake	1-100 cREP	Stake range per vote, capped per Voter ID
Vote cooldown	24 hours	Wait time before voting on the same content again

The 3-voter minimum is a deliberate balance between manipulation resistance and early-stage practicality. With fewer than 3 voters, a coordinated pair could control round outcomes. Formal verification confirms that 2+1 collusion at the 3-voter threshold yields less than 1 cREP total profit. Rounds that do not reach 3 revealed votes within 1 week are cancelled with full refunds -- nobody earns or loses anything. As the platform grows and rounds naturally attract more voters, governance can increase this threshold to further strengthen consensus quality.

5.6 Treasury

The governance treasury is held by the timelock controller and starts with 10M cREP. It grows over time through three token inflow sources:

- 1% settlement fee -- 1% of every losing pool is sent to the treasury when rounds settle.
- Slashed submitter stakes -- when content is flagged for policy violations or receives unfavorable ratings, the submitter's 10 cREP stake is slashed to the treasury.
- Forfeited vote stakes -- voters who fail to reveal their votes during the reveal phase lose their staked cREP to the treasury.

Treasury tokens can only be distributed through governance proposals. Token holders propose allocations, the community votes, and after the timelock delay, the transaction is executed on-chain. This ensures transparent, community-controlled distribution of protocol tokens.

5.7 Collusion Prevention

The integrity of cREP's content curation depends on honest, independent voting. Groups that coordinate to artificially upvote or downvote content undermine the parimutuel system and harm fair curation.

Detection

Community members can monitor voting patterns on-chain. Suspicious activity -- such as coordinated voting from related wallets, vote timing patterns, or unusual stake distributions -- can be flagged and analyzed using on-chain data.

Enforcement via Governance Proposals

When hard evidence of collusion is found, the community can take action through governance:

- Revoke Voter IDs -- governance can permanently revoke the Voter ID NFTs of confirmed colluders, removing their ability to vote on the platform.
- Reward whistleblowers -- governance is encouraged to allocate cREP from the treasury to reward community members who provide evidence of collusion.

Deterrence

Several protocol features make collusion costly and difficult:

- Sybil resistance -- 1 person = 1 Voter ID via passport verification (Self.xyz).
- Stake caps -- maximum 100 cREP per content per round limits single-voter influence.
- Vote cooldowns -- 24-hour cooldown prevents rapid re-voting on the same content.
- Permanent revocation -- losing your Voter ID is irreversible and eliminates voting ability.

Formal Collusion Model

A coalition of C colluders coordinates to vote in the same direction on target content. Each colluder stakes s_c (up to 100 cREP). Their combined stake is S_C = C x s_c. Let S_H denote honest voters' stake on the opposite side. The coalition wins if S_C > S_H. Coalition profit (beyond recovering stakes) is 0.82 x S_H (the 82% voter share), shared among C members. Per-member profit:

$$\text{profit per member} = \frac{0.82 \cdot S_H}{C}$$

Diminishing Returns

For collusion to exceed the per-member coordination cost k (communication, trust establishment, detection risk):

$$\frac{0.82 \cdot S_H}{C} > k$$

As coalition size C grows, per-member profit shrinks linearly while coordination cost and detection risk increase. This creates a natural ceiling on profitable coalition size. Furthermore, if honest voters respond to suspected collusion by increasing their counter-stakes, S_H grows and the required coalition size increases further.

Detection Probability and Expected Penalty

On-chain signals of collusion include: identical vote timing within the same block or narrow window, correlated stake amounts, shared funding sources traceable via transaction graphs, and repeated same-direction voting on identical content across rounds. The probability of detection P(detect | C) is monotonically increasing in C. Combined with permanent Voter ID revocation, the expected penalty is:

$$E[\text{penalty}] = P(\text{detect} \mid C) \cdot C \cdot V_{\text{future}}$$

where V_future represents the net present value of each identity's future voting rewards. For sufficiently high detection probability or future voting value, the expected penalty exceeds the one-time collusion profit, making collusion a negative expected-value strategy.

The process follows cREP's standard governance flow: evidence is submitted, a governance proposal is created, the community votes, and after the timelock delay, the action is executed.

5.8 Governance Security

On-chain governance carries its own attack surface. A malicious actor who accumulates sufficient voting power could propose changes that benefit themselves at the expense of the community -- for example, altering reward splits, revoking honest Voter IDs, or draining the treasury. Curyo's governance design includes several layers of defense against such attacks.

Snapshot-Based Voting

Governance voting power is snapshot-based: it is locked at the block when a proposal is created. This prevents flash-loan attacks (borrowing tokens to vote, then returning them) and just-in-time token acquisition. An attacker must hold cREP before the proposal exists, making surprise governance attacks impractical.

Timelock Delay

All approved proposals enter a 2-day timelock queue before execution. This gives the community a window to detect malicious proposals and organize a response -- including submitting counter-proposals or alerting the broader community. The delay acts as a circuit breaker against governance capture.

Early-Stage Concentration Risk

Quorum is calculated as 4% of circulating supply -- total supply minus tokens locked in the HumanFaucet, ParticipationPool, and RewardDistributor contracts. This dynamic calculation ensures governance is usable from day one: when only a small number of users have claimed tokens, the quorum scales proportionally to actual circulation rather than the full 100M supply. A minimum floor of 10,000 cREP prevents trivially small quorums in the earliest stages. As the user base grows and more tokens enter circulation, the quorum threshold increases proportionally, requiring increasingly broad consensus.

No Privileged Keys

After deployment, no admin keys, multisigs, or privileged roles exist. The timelock controller is the sole owner of all protocol contracts, and it can only execute transactions that have passed the full governance lifecycle (proposal, voting, timelock). The proposal threshold is deliberately low (100 cREP) to encourage participation -- the real

protection is the combination of dynamic quorum (4% of circulating supply with a 10K cREP floor), majority vote, and timelock delay, not proposal gating.

6. Curyo & AI

How stake-weighted curation addresses the AI content crisis and produces public quality infrastructure.

6.1 The Model Collapse Problem

Research by Shumailov et al. (Nature, 2024) demonstrates that AI models trained recursively on AI-generated content undergo 'model collapse' -- a progressive loss of distributional fidelity where each successive generation of models loses the tails of the original data distribution. As AI-generated content proliferates across the web, the training data available to future models becomes increasingly contaminated with synthetic output, accelerating this degradation cycle.

The implication is that verified human quality signals become critical infrastructure for maintaining the fidelity of AI systems. Without reliable mechanisms to distinguish high-quality content from low-quality or AI-generated filler, training pipelines face an increasingly noisy signal-to-noise ratio. Curyo addresses this by producing stake-weighted, Sybil-resistant quality ratings anchored to economic commitment from verified human identities.

6.2 Stake-Weighted Curation

The concept of 'staked media' (a16z, Big Ideas 2026, https://a16z.com/newsletter/big-ideas-2026-part-3/#the-rise-of-staked-media) -- where content quality is assessed through economic commitment rather than algorithmic engagement -- provides a manipulation-resistant alternative to traditional curation mechanisms. Curyo implements this approach through its parimutuel voting system: voters commit cREP tokens to their quality predictions, and the commit-reveal mechanism ensures independent assessment via tlock encryption.

This design produces quality signals with several properties that distinguish them from engagement-based metrics:

- Economic commitment -- Each rating is backed by a token stake, making systematic manipulation expensive relative to the signal produced.
- Independence -- Tlock encryption during the commit phase prevents voters from observing and copying others' votes, producing Schelling-point convergence on genuine quality assessments.
- Sybil resistance -- Passport-verified Voter IDs limit each human to one identity with a capped stake per content, preventing bot farms from flooding the signal.
- Verifiability -- All votes, stakes, and outcomes are recorded on-chain with cryptographic integrity, enabling third-party audit and reproducibility.

6.3 On-Chain Ratings as Public Infrastructure

A foundational design decision in Curyo is the use of a public blockchain as the settlement layer. This ensures that all quality ratings -- including individual vote directions, stake amounts, round outcomes, and resulting content scores -- are inherently public, permissionless, and exportable. No API key, rate limit, or terms-of-service restriction mediates access to the data.

This positions Curyo's output as public goods infrastructure rather than a proprietary dataset:

- AI training pipelines can incorporate on-chain quality scores to filter or weight training corpora, mitigating model collapse by prioritizing human-verified content.

- Search engines and recommendation systems can consume on-chain ratings as an independent quality signal, reducing dependence on engagement-based proxies.
- Researchers retain full transparency into voting dynamics, curation patterns, and content quality trends without data access barriers.
- Third-party platforms can build on the quality layer without permission, payment, or partnership agreements.

Unlike centralized rating platforms where data is siloed behind proprietary APIs, blockchain-native ratings function as a commons. This aligns with the thesis that the AI-dominated web requires open, verifiable quality infrastructure rather than additional walled gardens.

6.4 AI-Assisted Voting

Curyo incorporates AI as a first-class participant through automated voting bots that use pluggable rating strategies. Each strategy queries an external API to obtain a normalized quality score for submitted content. The bot votes UP or DOWN based on whether the score meets a configurable threshold.

Bot votes use the same tlock encryption and commit-reveal mechanism as human votes, making them cryptographically indistinguishable on-chain. Bots stake the minimum amount of cREP per vote, ensuring their influence remains small relative to human voters who may stake significantly more. The parimutuel mechanism provides natural selection pressure: strategies that produce inaccurate ratings lose their stakes, while accurate strategies accumulate reputation.

Human Oversight

The system is designed so that human voters retain decisive influence. Bots staking the minimum are outweighed by any human voter staking more. In contentious rounds, the aggregate human stake dominates bot contributions. This creates a hybrid model: AI provides baseline signals and seeding, while humans provide authoritative quality judgments.

Cold-Start Mitigation

AI-assisted voting directly addresses the cold-start problem inherent in new content platforms. When a content item is submitted, automated strategies can produce initial quality signals within seconds, seeding the voting market before human participants engage. This creates immediate activity and provides a focal point for human voters to agree or disagree with, accelerating convergence toward accurate ratings.

6.5 Future Directions

Curyo's architecture enables several extensions at the intersection of AI and decentralized curation:

- Cross-platform quality oracle -- On-chain content ratings can serve as an oracle for other protocols and platforms, creating a shared quality layer across the decentralized web.
- Expertise-weighted reputation -- Domain-specific reputation multipliers could allow voters with demonstrated accuracy in specific categories to earn additional influence, improving signal quality in specialized domains.
- Content provenance integration -- Combining Curyo ratings with content provenance standards (C2PA) would create a two-layered trust system: provenance verifies origin, stake-weighted curation verifies quality.

- Advanced AI strategies -- The pluggable strategy interface supports increasingly sophisticated approaches, from API-based lookups to LLM-driven content analysis. The parimutuel mechanism ensures that only strategies producing accurate ratings survive long-term.

7. Known Limitations

Transparency about design trade-offs, residual risks, and areas for improvement.

7.1 Keeper Trust Assumptions

The round-based voting system relies on keepers as the primary reveal mechanism. Keepers read tlock-encrypted ciphertexts stored on-chain and decrypt them using drand beacon signatures after each epoch ends. While the design provides a one-honest-keeper guarantee (any single honest keeper can reveal all votes), a malicious keeper who is the sole operator could selectively withhold reveals to influence outcomes. Mitigations include: permissionless reveal (anyone can call revealVote with the correct plaintext), tlock-encrypted ciphertext stored on-chain as a public data source, and the ability to run multiple independent keepers. For production deployments, running 3+ independent keeper instances is strongly recommended.

7.2 Tlock Fallback Limitations

While tlock-encrypted ciphertext is stored on-chain during commits, the contract does not include an on-chain decryption mechanism. If all keepers fail, someone must decrypt the ciphertext off-chain using the drand beacon signature after the round expires, then submit the reveal manually. This process is not automated and requires technical knowledge. If no one performs the off-chain decryption, cancelExpiredRound() provides a permissionless escape with full refunds.

7.3 Consensus Subsidy Pool

The parimutuel reward structure distributes the losing pool to winners. When all voters vote in the same direction (unanimous outcome), the losing pool is zero and no standard parimutuel rewards are distributed. Without mitigation, this creates a perverse incentive: no reason to vote on obviously good or bad content, and coordinated groups could benefit from manufacturing dissent by having one member vote against the majority to create a losing pool.

The consensus subsidy pool solves this. It is pre-funded with 4,000,000 cREP from the treasury allocation and continuously replenished by 5% of every losing pool from contentious rounds. When a round settles unanimously (losingPool = 0), the contract distributes a subsidy from this reserve equal to 5% of the round's total stake, capped by the reserve balance.

The subsidy formula is:

$$\text{subsidy} = \min\left(\text{reserveBalance},\ \text{totalStake} \times 0.05\right)$$

This subsidy is split between voters (~89%) and the content submitter (~11%), using the same 82:10 ratio as normal round rewards, and distributed proportionally by stake within each group. Since all voters are on the winning side, every voter receives a share. The mechanism is self-sustaining: contentious rounds -- where parimutuel rewards function normally -- generate surplus that funds consensus rounds. Every contentious round with L cREP in its losing pool contributes 0.05L to the reserve, which can fund approximately one unanimous round of equivalent total stake. The 4M initial pre-fund provides runway during early adoption when contentious rounds may be infrequent.

Consensus subsidy rewards are intentionally lower than contentious-round rewards (approximately 10:1 ratio), preserving the incentive to vote on genuinely contentious content while making consensus curation non-zero.

7.4 Configuration Change Timing

Governance can change round parameters (minimum voters, round duration, epoch duration) at any time through the standard proposal process. Changes apply to new rounds only: each round snapshots configuration at creation time, so in-progress rounds keep the rules they started with.

7.5 Settlement External Dependencies

Round settlement interacts with external contracts (ParticipationPool, FrontendRegistry, CategoryRegistry) using fail-soft wrappers for non-critical side effects. If one of these external calls reverts, settlement can continue while skipping that side effect, preventing total settlement blockage at the cost of temporarily deferred accounting or payouts for that component.

7.6 Identity Verification Scope

Passport-based identity verification via Self.xyz provides strong Sybil resistance but excludes approximately 1.1 billion people globally who lack passports. The system has no appeal mechanism for false rejections, and recovery from a compromised or offline Self.xyz service is not documented. These are inherent trade-offs of passport-gated identity systems.