

The White Paper on A Participatory Validation System

Lam Lai

March 2026

Abstract

ACT 0: FILL THE CARGO FOR THE RED PLANET proposes a participatory validation system that enables cross-community collaboration in cultural preservation at global scale. The project operates within a fictional frame – the imagined settlement of Mars – which creates the conditions for genuine collective expression free from material consequence and correct answers. Internet-based participation allows geographically distributed communities to contribute to a shared validation pool. The system addresses the inherent dynamics of internet participation – asynchronous engagement, fluctuating scale, and distributed coordination – through adaptive threshold mechanisms and referral-based network growth.

Rather than operating within single bounded communities or relying on centralised curation, the system creates a referral-based constellation in which participants from different institutional contexts— theatre groups, digital art institutions, cultural organisations, and peer-referral networks—validate responses collectively.

Participants contribute distributed cultural labour across the full range of system functions – submitting responses, voting, reviewing flagged content, and proposing prompts. Through progressive role levels, participants perform increasingly complex validation work. This labour-based participation model investigates whether time and attention—rather than capital or tokens—can serve as a basis for collective value assessment.

Responses accumulate votes through structured attention mechanisms. When vote counts reach dynamically calculated thresholds—adapted to current participation scale—responses transition into a permanent cargo archive. The system is designed to accommodate volatile participation dynamics: communities joining and leaving unpredictably, engagement fluctuating dramatically, and group sizes varying by orders of magnitude, with validation context preserved as metadata for each cargo entry.

Upon project closure, the validated cargo archive serves multiple purposes: as permanent cultural record, as source material for theatrical, published, exhibited, or

other interdisciplinary work depending on available resources and partnerships, and as proof-of-concept for adaptive cross-community governance.

The proposed architecture comprises six interconnected mechanisms: Core Participation Loop, Prompt Design System, Content Review System, Membership System, System Versioning Control, and Blockchain Layer. The design has been explored through prototype development including ticket pool and threshold dynamics simulation, submission-voting flow testing, and blockchain anchoring experimentation.

This work proposes a novel approach to cross-community cultural preservation that adapts to participation dynamics and maintains accessibility without financial barriers. The research contributes: (1) adaptive threshold models for internet-scale participation with fluctuating community sizes, (2) a labour-based validation architecture where influence derives from effort rather than capital, and (3) architectural patterns for cross-institutional cultural collaboration that preserve participant agency without centralised control, with implications for distributed cultural governance, commons-based cultural production, and participatory archiving beyond the specific theatrical context.

TABLE OF CONTENTS

1	Introduction	4
1.1	Vision & Context	4
1.2	The Challenge	4
2	Research Questions	5
2.1	Primary Research Questions	5
2.2	Secondary Research Questions	5
3	System Overview	7
3.1	System Architecture	7
3.2	Core Participation Loop	8
3.3	Prompt Design System	13
3.4	Content Review System	17
3.5	Membership System	19
3.6	System Versioning Control	23
3.7	Blockchain Layer	28
4	Conclusion	35

1. Introduction

1.1 Vision & Context

To begin again is never neutral. We carry with us inherited beliefs about value, identity, belonging, and survival—assumptions so deeply woven into our sense of self that we forget they were ever choices. The imagination of human settlement on Mars offers a rare opportunity: a shared question that transcends cultural boundaries and institutional contexts. What deserves to be preserved? What should be discarded? What new logic of living can we co-create?

This project addresses that question through participatory validation—a system in which geographically distributed communities collectively determine what cultural artifacts deserve to be carried forward. By creating infrastructure for symbolic decision-making, emotional resonance, and collective reflection, the project offers a space to examine our inheritances and co-author what comes next.

The validated responses—those that survive collective scrutiny and reach preservation thresholds—become source material for theatrical, published, exhibited, or other interdisciplinary work depending on available resources and partnerships. Through this lens, system design becomes dramaturgy. The technical architecture creates conditions for a collective performance of value assessment. The question of the future becomes a shared unfolding story, written not by individual authors but through accumulated micro-decisions of a distributed community.

This collective validation unfolds across time, not as a single synchronized judgment. As communities grow, shrink, and transform—as happens naturally in both physical settlements and internet participation—the system captures what achieves consensus within each temporal moment. A decision made by ten founding members carries the same legitimacy as one made by a thousand-member established community: both represent the collective will of everyone who could participate at that time. The cargo archive becomes a temporal record of cultural priorities across different scales of community, not a ranked list judged by a single fixed standard.

1.2 The Challenge

Most internet platforms employ centralised algorithmic curation: users provide content, but opaque algorithms determine what gets visibility and preservation. Engagement remains shallow—liking, commenting, sharing—rather than substantive validation. Users are content providers, not decision-makers.

This system proposes a different model: participant-validated collective curation. While infrastructure remains centrally operated, validation decisions emerge from distributed

human judgment through transparent, rule-based mechanisms. Participants perform substantive cultural labour—voting, reviewing, curating—with influence derived from effort rather than capital or algorithmic ranking.

The cargo archive must capture voices from communities at all scales of formation—from founding intimacies to established consensus. Fixed thresholds systematically privilege one over the other. Adaptive thresholds preserve both, treating small-community and large-community validation as equally essential cultural perspectives across the temporal arc of collective formation.

The challenge is structural: how can such validation function when internet participation introduces inherent volatility—asynchronous engagement, unpredictable fluctuations in community size, distributed coordination? Traditional governance assumes stable, bounded groups. Internet-scale participation requires adaptive mechanisms that maintain validation quality whether tens or thousands are active.

Adaptive thresholds respond to observed participation patterns. When the community is small, thresholds remain achievable. When participation scales up, thresholds rise proportionally. The cargo archive becomes a temporal record of what achieved consensus within different scales of participation—not a ranked list judged by a fixed standard. Each cargo entry carries its validation context as metadata: threshold model, vote count, date. This preserves the ability to analyse how validation quality relates to community scale, which is itself a research question this project investigates through deployment and observation.

2. Research Questions

2.1 Primary Research Question

"How can adaptive validation mechanisms enable multiple independent communities to collectively curate cultural artifacts without centralised coordination, while accommodating the inherent volatility of internet-scale participation?"

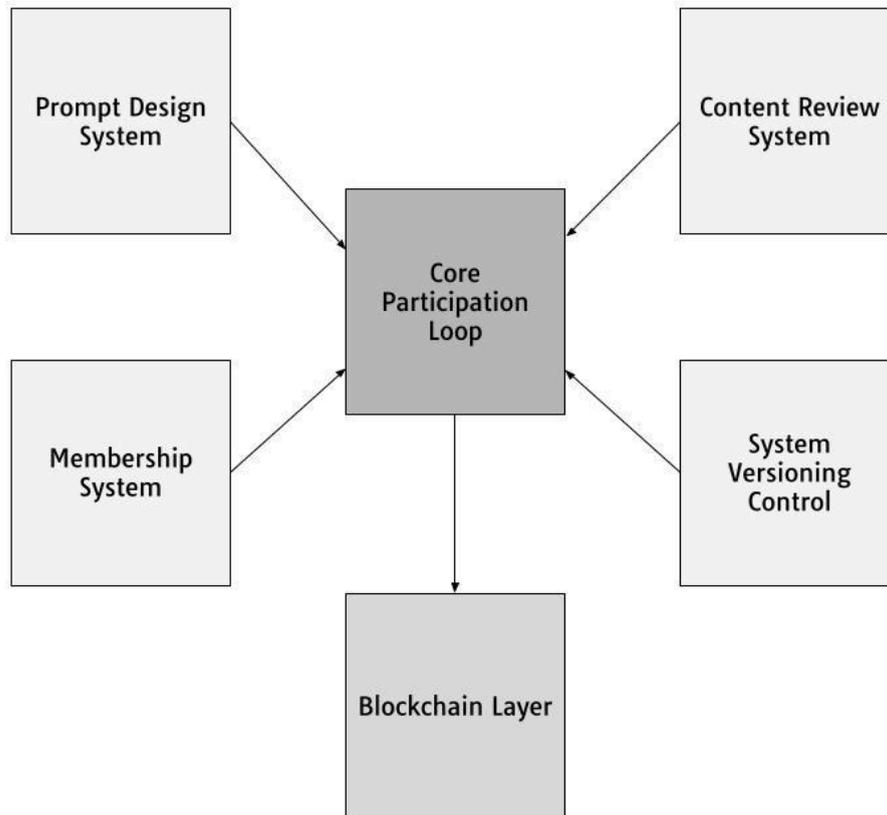
2.2 Secondary Research Questions

- A. Validation Quality Across Scales: "Do responses validated at different community scales represent meaningfully different quality or legitimacy, or does proportional consensus (percentage of active participants) equalize validation across scales?"

- B. Cross-Community Attention Dynamics: "How do voting patterns and validation outcomes change when voices from different cultural/institutional contexts overlap in a shared pool?"
- C. Labour-Based Validation: "Can time and attention—rather than capital or tokens—serve as basis for collective value assessment in a distributed validation system?"
- D. Participation Volatility: "What threshold calculation models (fixed, percentage-based, square root, and logarithmic) best balance accessibility and selectivity when community size ranges from small groups to large-scale participation?"
- E. Minimal Blockchain Integration: "What is the minimum viable on-chain anchoring that preserves validation integrity without requiring participants to use a wallet?"
- F. Evolving Candidate Pool: "Does continuous submission timing (early vs. late in the project lifecycle) create systematic validation advantages, or do multi-modal view modes and window size equalise discovery across the temporal span of participation?"

3. System Overview

This section describes the structural logic of the system. It begins with a high-level architecture diagram and a brief introduction of its six constituent mechanisms. Each mechanism is then described in detail in the subsections that follow.



3.1 System Architecture

The system is composed of six mechanisms that operate continuously and in relation to one another. The Core Participation Loop forms the centre of the architecture – it is the fundamental cycle through which collective validation occurs. The five surrounding mechanisms each serve a distinct function: two feed into the Core Loop by governing what enters it, two regulate how it operates over time, and one receives its output.

The diagram above illustrates these relationships. Arrows indicate the direction of primary influence. No mechanism operates in isolation.

The six mechanisms are:

1. **Core Participation Loop** – the fundamental cycle of responses, voting, threshold validation, and cargo archiving. All other mechanisms connect through this loop.

2. **Prompt Design System** – governs the creation and maintenance of the question pool. Prompts feed directly into the Core Loop by shaping what participants submit.
3. **Content Review System** – monitors the pre-cargo pool for harmful or non-compliant content. Operates in parallel with the Core Loop through community flagging and committee review.
4. **Membership System** – governs who can participate and at what level of access. Determines which actions are available to each participant at any given point in the system.
5. **System Versioning Control** – monitors participation patterns and manages structural adaptation across two dimensions: scale-based versioning, which adjusts threshold models, window size, and bonus ticket parameters as community size changes; and timeline-based versioning, which governs the project lifecycle from opening to closure.
6. **Blockchain Layer** – receives validated responses from the Core Loop and anchors them permanently on-chain. The final layer of the system – its outputs are immutable.

3.2 Core Participation Loop

The Core Participation Loop is the fundamental cycle through which collective validation occurs. It runs continuously and forms the foundation on which all other mechanisms operate. The loop begins with a response and concludes with a cargo entry – permanently anchored on-chain at project closure.

3.2.1 Responses

A participant selects a category from the available prompt pool and receives a prompt. For Level 1 participants, the prompt includes a fill-in-the-blank answer template. The participant completes the template, producing a grammatically complete sentence. Higher-level participants receive the prompt question without a template and write their own sentence freely. A character limit applies at each level, increasing as participants advance. Specific character limits will be calibrated during deployment testing.

Before the response enters the pool, the system performs an exact-match duplicate check against all existing responses. If an identical response already exists, the participant is notified and can choose to vote on the existing response with their ticket instead of creating a duplicate. If no match is found, the response enters the pre-cargo pool as a new record.

3.2.2 Vote Ticket Issuance

Each response generates one base vote ticket, issued to the author upon successful entry into the pool. Tickets are personal, non-transferable, and single-use. They cannot be used on the author's own response. Tickets accumulate across days and carry expiry dates.

Bonus Ticket Mechanisms

Participants who demonstrate sustained involvement in the system earn bonus vote tickets beyond the base issuance. This creates a weighted voting dynamic where more committed participants gain proportionally more influence—not through explicit vote multipliers, but through earning additional tickets to spend.

Bonus tickets are awarded based on specific behavioural triggers:

Temporal consistency triggers:

- Consecutive-day submission (submitted yesterday and today)
- Weekly participation streaks
- Re-engagement after absence

Task completion triggers:

- Participants may cast a maximum of three votes per day
- Reviewing flagged content (Level 2+)
- Completing community questions into prompts (Level 3+)
- Clearing work queue backlogs

Community value triggers:

- Responses receiving votes from others
- Responses wishlisted by multiple participants
- Responses validated as cargo

The specific trigger conditions and bonus amounts are subject to calibration through deployment testing. The goal is to reward behaviours that maintain system health—consistent participation, active voting, governance work, and quality contribution—while ensuring that voting power is earned through participation, not granted through privilege.

This mechanism addresses two system needs: (1) incentivising sustained engagement and labour across all system functions, and (2) balancing the response-vote ratio as the pool grows, by ensuring that active contributors have sufficient voting capacity to engage with both new and accumulated content.

The volume of bonus tickets in circulation is expected to increase naturally across the project lifecycle. In the early phase, most participants operate at Level 1 and have access to only the base submission and voting triggers. As participants advance to higher role levels – unlocking content review, prompt curation, and governance tasks – the range of bonus-generating behaviours expands. Later phases therefore see higher effective ticket issuance not through administrative adjustment, but as a natural consequence of community maturation. This progression aligns with the observed shift in participation behaviour: early participants tend to submit more than they vote, while returning participants in later phases tend to vote and complete tasks more than they submit new entries. The bonus ticket system is designed to reward and sustain exactly this shift.

Ticket Pool Dynamics

The ticket pool at any given moment reflects the accumulated unspent tickets of all active participants. Not all ticket holders vote—some participants submit but do not return. Their tickets remain in the pool until expiry. The effective vote supply is therefore always lower than the total ticket count.

Bonus ticket mechanisms amplify the voting power of consistent participants, helping to offset pool growth and ensuring that regular contributors can engage meaningfully with the expanding response pool. The relationship between ticket issuance (base + bonuses), expiry, and actual voting behaviour is a key variable in the system's validation dynamics.

3.2.3 Daily Snapshot & Voting

Each day, the system generates a personal snapshot for each participant. The snapshot consists of five lists drawn from the pre-cargo pool, each ordered by a different view mode: random, popular (most votes first), newest first, oldest first, and neglected (fewest votes first). The participant may switch freely between the five modes within their snapshot. The snapshot is fixed for the day – switching modes does not regenerate the lists. The snapshot refreshes at the daily reset, which occurs at a fixed server-side time regardless of the participant's timezone.

A participant may cast a maximum of three votes per day, regardless of how many tickets they hold. Each vote spends one ticket. Votes are distributed across responses in the visible snapshot – a participant may vote for up to three different responses. The three-vote daily cap prevents any single participant from exerting disproportionate influence on a given day.

The view mode—Random, Popular, Newest, Oldest, or Neglected—determines which responses receive attention on a given day. This multi-modal approach addresses a fundamental challenge: unlike traditional voting with fixed candidate lists, the response pool expands continuously as new submissions arrive. Early responses have more

opportunities to appear in daily snapshots than late responses, creating potential temporal bias. Different view modes serve strategic functions:

- Random — Equalises discovery probability across all responses regardless of submission date
- Popular — Amplifies responses already receiving votes (momentum-based validation)
- Newest — Surfaces recently submitted responses, ensuring new content receives immediate attention
- Oldest — Prevents early submissions from being buried by newer content
- Neglected — Surfaces responses with few or zero votes, rescuing late arrivals or overlooked contributions

Under consideration: whether the snapshot is generated once at the start of each day for all participants, or generated individually at the moment each participant opens the platform. The latter would mean the visible pool reflects real-time cargo transitions and new submissions, producing a more dynamic discovery experience at the cost of snapshot consistency across participants.

3.2.4 Threshold & Cargo Transition

Votes accumulate per response over time. When a response's vote count reaches the required threshold, it transitions from the pre-cargo pool to the cargo archive. The threshold is not fixed — it adapts based on the current system version, which is determined by participation scale. Threshold models include fixed values, percentage-based calculations, and adaptive formulas such as square root or logarithmic scaling. The active model at any given time is determined by the System Versioning Control mechanism.

A response that reaches the threshold transitions instantly into the cargo archive but is not anchored on-chain yet. Participants can review the cargo archive. The threshold value interacts with snapshot window size — when window size is small relative to the submission pool, attention concentration can override threshold differences. Both parameters are managed as versioning variables by the System Versioning Control mechanism.

The threshold value at any given time reflects the system's current understanding of what constitutes meaningful collective consensus given the active participation scale. This value is calculated transparently and recorded as metadata with each cargo entry, preserving the validation context for future interpretation. A response validated by 10 votes when 50 people are active carries different contextual meaning than one validated by 100 votes when 5,000 people are active—both may represent legitimate collective consensus within their respective temporal-participation contexts, but they are not

assumed to be equivalent. This project investigates these relationships by observing actual participant behaviour and validation outcomes during deployment.

3.2.5 Cargo Archiving & On-Chain Anchoring

At project closure, the ceremonial batch anchoring is executed. The Anchor Keeper – a Level 5 participant elected by the community through sustained labour and collective recognition – activates the procedure of anchoring all validated cargo records on-chain via the blockchain layer.

The on-chain record contains the submission ID, a data hash of the content, a timestamp, and the active model version. The response text itself remains on the project server and in the IPFS archive – only the cryptographic fingerprint is anchored on-chain. This separation preserves privacy, reduces costs, and focuses permanence on collectively validated cultural artifacts.

The anchor record is permanent and immutable. Once anchored, the record cannot be changed or deleted by anyone, including the project team. The cargo archive transitions from a living document – growing, filling, waiting – to a sealed record.

For membership advancement triggered by cargo anchoring, see Section 3.5.

3.2.6 Sub-Mechanics

The following mechanics operate within the Core Participation Loop and are integral to its function:

- Daily snapshot generation and storage – five personalised lists generated per participant per day, stored server-side, keyed by userID and date. Under consideration: real-time generation at the moment of access rather than fixed daily generation – see Section 3.2.3.
- Duplicate detection – exact-match check on response input before entry into the pool
- Ticket expiry logic – batches of tickets tracked by issue date, discarded when the expiry window is reached
- Threshold calculation – vote count compared against the active threshold model on each voting cycle
- Snapshot window sizing – the number of responses visible per view mode per snapshot, managed as a versioning parameter by System Versioning Control.

3.3 Prompt Design System

The Prompt Design System is the mechanism through which the question pool is built and maintained. It operates as a separate intake flow from the main participation system, with its own access routes and approval process. All prompts must pass admin review before becoming active in the system.

3.3.1 Prompt Types

The system supports two types of prompts, each serving different modes of reflection:

Regular Prompts invite open-ended completion of sentence templates. Participants fill in the blank to express their perspective in their own words.

Community Poll Prompts present binary questions requiring Yes/No responses. These prompts surface tensions, trade-offs, and fundamental disagreements about Mars governance and society.

Both types flow through the same validation mechanism. Individual responses—whether sentence completions or binary positions—compete for community attention and validation.

3.3.2 Regular Prompt Structure

Each prompt consists of four components:

- Question – a question that invites the participant to reflect on a specific dimension of life on Mars
- Answer template – a fill-in-the-blank sentence structure that the participant completes. The completed sentence becomes their response
- Category – a thematic grouping that determines which attention pool the prompt belongs to. Participants select a category before being assigned a prompt
- Labour – a thematic grouping that represents the essential collective practices that sustain life and society in the imagined Martian settlement

Example:

- Question – What is the one thing that needs to be protected at all cost on Mars?
- Template – _____ would be protected at all cost on Mars.
- Category – Value
- Labour – The Labour of Governance, Justice & Rights

3.3.3 Community Poll Prompt Structure

Community Poll prompts consist of three components:

Question – A binary question about Mars society that can be answered Yes or No

Category – The thematic grouping (same as regular prompts)

Labour – The collective practice this question feeds (same as regular prompts)

Example:

- Question: "Should privacy be sacrificed for collective safety on Mars?"
- Category: Value
- Labour: Governance, Justice & Rights

Participants respond simply "Yes" or "No". These binary responses enter the validation pool alongside sentence responses from regular prompts.

3.3.4 Prompt Categories & Labours

The prompt system operates through two distinct but related concepts: Categories and Labours. These are separate fields – a prompt designer chooses both independently. There is no fixed mapping between them. The combination is a curatorial and artistic decision made by the prompt designer.

The Relationship:

The Category is the entry point – the human, personal angle a participant recognises from their own life on Earth. The Labour is where the response lands – what the response actually feeds in the settlement once validated by the community.

The participant starts from somewhere familiar. Their response, once it becomes cargo, turns out to be contributing to something the settlement fundamentally needs. The journey from Category to Labour is one of the project's core dramatic mechanisms – the moment where personal reflection connects to collective necessity.

Example Mappings:

A prompt in the "Value" Category might map to the Labour of Governance, Justice & Rights:

- Category (entry): Values (familiar moral terrain)
- Labour (destination): Governance structures the settlement needs

A prompt in the "Belonging" Category might map to the Labour of Health & Care:

- Category (entry): Belonging (questions of connection and community)
- Labour (destination): Care systems the settlement needs

The same Category can lead to different Labours depending on the prompt designer's curatorial vision. A "Value" prompt might feed *Governance*, or *Memory*, *Culture & Ritual*, or *Communication*. This flexibility allows prompt designers to reveal unexpected connections between familiar earthly concerns and Martian collective needs.

3.3.5 Access Routes

Prompt contribution occurs through three pathways, each with different levels of access and responsibility:

Route 1: External Contributor Proposals

Artists, writers, and theatre makers may be invited by the project team to propose complete prompts (question, template, category, and labour assignment). They receive a dedicated link via email that leads to a prompt proposal form. This access exists outside the participant role ladder and does not require a participant account. The link remains active until the admin closes it.

Route 2: Internal High-Level Proposals

Higher-level participants unlock the ability to propose complete prompts from within their existing account. This is an earned access that reflects deep engagement with the system. They submit the full prompt structure (question, template, category, and labour assignment).

Route 3: Community Question Submissions

Any participant (Level 1+) can submit a question they believe the community should address – the question they want the community to address. These question submissions do not require template design, category assignment, or labour mapping.

Submitted questions enter a curation queue visible to higher-level participants. These participants review questions and, if they judge one worthy of inclusion, complete the prompt by:

- Designing the fill-in-the-blank template
- Assigning the appropriate Category
- Mapping it to the relevant Labour

Once completed, the prompt enters the same admin review queue as Routes 1 and 2.

All three routes converge – external, internal high-level, and community-curated prompts all undergo identical admin review before entering the active pool. Internal and external contributions are treated identically once submitted.

3.3.6 Duplicate Detection

Before a prompt proposal enters the review queue, the system checks whether an identical question already exists in the prompt pool. If an exact match is found, the contributor is notified and the duplicate is not added. This prevents redundancy in the prompt pool and ensures each prompt offers a genuinely distinct invitation.

3.3.7 Admin Review & Approval

All prompt proposals – regardless of source – enter a moderation queue before going live. The admin reviews each proposal for:

- Relevance – the prompt should invite meaningful reflection on life on Mars
- Template quality – the answer template should produce grammatically complete, standalone sentences when filled in
- Category and Labour alignment – the prompt should be assigned meaningfully to its thematic category and Labour
- Community standards – the prompt must not contain harmful, discriminatory, or inappropriate content

Approved prompts are added to the active pool immediately. Rejected proposals do not enter the system.

Under consideration: whether rejected prompt contributors receive feedback or just a rejection notification.

3.3.8 Prompt Lifecycle

The initial prompt pool is seeded by the project team before the system opens to participants. As the system grows, the pool expands through external invitations and participant proposals.

Under consideration: whether prompts can be retired or expired from the pool over time; whether partner organisations can be granted external contributor access as an organisation rather than as individuals; whether the prompt pool evolves through invited curatorial contributions at defined intervals during the project lifecycle – for example, guest prompt editions authored by invited artists, dramaturgs, or philosophers at specific moments during the mission period.

3.4 Content Review System

The Content Review System operates in parallel with the Core Participation Loop to maintain the integrity of the response pool. Despite the structured fill-in-the-blank format, harmful, discriminatory, or inappropriate content can still be submitted. The system handles this through three layers: automated pre-screening, community flagging, and committee review.

3.4.1 Pre-Screening

Before a response enters the pool, the system performs an automated keyword check against a list of prohibited terms. Responses that trigger the filter are blocked immediately and do not enter the pool. This layer catches the most obvious violations without requiring human intervention.

The keyword filter is the first line of defence. It is not exhaustive – the community flagging and committee review layers handle cases that pass the automated check.

3.4.2 Community Flagging

Any Level 1 or above participant may flag a response they believe violates community guidelines. A flagged response is not immediately removed from the pool – it remains visible and continues to accumulate votes while under review. This is a deliberate design decision: removing a response on a single flag would create a mechanism for bad-faith censorship, where participants could silence content they disagree with simply by flagging it.

To prevent abuse of the flagging system, each participant may raise a maximum of three flags per day. Flags beyond this limit are not accepted.

Under consideration: Whether participants receive any feedback when their flag is reviewed and resolved.

3.4.3 Committee Review

Flagged responses enter a review queue visible to Level 2 participants. A Level 2 reviewer must assess the flagged response within a defined time window— specific duration to be calibrated during deployment testing, and select one of three outcomes:

- Keep – the flag is dismissed. The response returns to normal status in the pool with no further action.
- Remove – the response is confirmed as a violation. It is removed from the public pool and moved to the private moderation archive.
- Refer to admin – the case is ambiguous, serious, or involves a repeat offender. The admin makes the final decision.

If a flagged response is not reviewed within the defined time window, it is automatically referred to the admin.

3.4.4 Private Moderation Archive

Removed responses are not deleted from the system. Instead, they are moved from the public response pool to a private moderation archive, accessible only to the admin. Each archived record retains the original response content, the userID, the submission date, the flag date, the removal date, the stated reason for removal, and the identity of the reviewer who made the decision.

This archive serves multiple purposes: it provides a complete audit trail for moderation decisions, supports a fair appeals process if a participant disputes a removal, and enables pattern detection for repeat offenders. The private moderation archive is never publicly visible. It exists for administrative and audit purposes only.

3.4.5 Appeals

A participant whose response has been removed may contact the admin to request a review of the decision. The admin may consult the moderation archive record and reverse the removal if the original decision is found to be incorrect. Reinstated responses re-enter the public pool with their original vote count intact.

Under consideration: Whether a formal appeals mechanism is built into the interface or handled via direct contact with the admin in the initial phase; whether a flagged response that reaches the cargo threshold during review is held in a pending state until the review is resolved, or transitions to cargo immediately.

3.5 Membership System

The Membership System governs who can participate, how they enter, what they can do, and how their involvement evolves over time. Participation is structured through a five-level progressive role ladder. Advancement is earned through sustained engagement and task completion over time.

3.5.1 Authentication

Participation does not require a password. When a new participant visits the platform, they enter their email address and receive a one-time authentication code. Upon verification, a unique userID is generated and linked to their email. The email is retained only for re-authentication and notification purposes.

Access to the system is granted through two pathways:

- Institutional invite links – Partner organisations (theatre groups, digital art institutions, cultural organisations) receive unique invite links to distribute to their members. The system records which organisation each participant joined through, enabling analysis of cross-community participation patterns.
- Participant referrals – Active participants can generate personal referral links to invite others from their networks. This peer-to-peer expansion allows the constellation to grow organically beyond institutional boundaries while maintaining connection accountability.

This dual-pathway design creates a referral-based constellation where participants from different institutional contexts contribute to a shared validation pool. Every participant has traceable provenance—they entered through an organisation or were invited by an existing participant—creating a web of soft trust without requiring invasive identity verification.

3.5.2 Progressive Role Structure

Participants advance through five progressive role levels based on sustained engagement and task completion. Each level unlocks additional responsibilities and validates the work of levels below it. Higher levels do not replace lower-level participation—all participants continue to submit responses and vote regardless of role level.

Level 1: Contributor

- Submit responses to prompts
- Vote on others' responses (3 per day maximum)
- Flag inappropriate content (3 per day maximum)
- Submit questions for prompts (Route 3 - community question submissions)
- View cargo archive

Level 2: Reviewer

- All Level 1 capabilities
- Review flagged content (Keep / Remove / Refer to admin)
- Vote on neglected responses (committee mechanism for attention distribution)

Level 3: Curator

- All Level 2 capabilities
- Complete community-submitted questions into full prompts (design template, assign Category and Labour)
- Propose Community Poll questions

Level 4: Validator

- All Level 3 capabilities
- Propose complete prompts (Route 2 - internal high-level proposals)
- Validate Community Poll questions proposed by Level 3

Level 5: Anchor Keeper

- All Level 4 capabilities
- Eligible to be elected as Anchor Keeper (execute final on-chain anchoring ceremony)

In the participant interface, the five role levels are represented through a crystal progression system – Amorphous, Nucleating, Crystallising, Faceted, and Set – reflecting the Mars mission's fictional framing and giving participants a poetic rather than numerical sense of their development within the system.

3.5.3 Level Advancement

Advancement is earned through sustained participation and consistent labour, not solely through cargo validation. The system tracks temporal engagement patterns (active days, return frequency) and task completion (responses submitted, votes cast, work performed at current level).

This design ensures advancement reflects commitment and contribution rather than competitive validation outcomes. Since cargo validation depends on community attention distribution—which can be unpredictable—relying on cargo entries for advancement would create frustration and inequity.

Advancement criteria combine temporal consistency with task-specific contributions:

- Lower levels (1→2, 2→3): Measured primarily by sustained participation over time (active days, voting frequency, basic task completion). Advancement occurs within weeks to months based on demonstrated engagement patterns.
- Middle levels (3→4): Require sustained participation plus completion of level-specific work (prompt curation, community poll participation). Advancement occurs over multiple months.
- Highest level (4→5): Requires deep sustained engagement, demonstrated system stewardship, and may include community recognition or admin nomination given the governance responsibilities involved.

Specific thresholds (number of days, tasks required, time minimums) will be calibrated during initial deployment based on observed participation patterns and system health metrics.

3.5.4 Participant Dashboard

Each participant has access to a personal dashboard displaying:

- Submitted responses – all responses with current vote count and pool status
- Threshold proximity – how close each response is to the current cargo threshold
- Vote tickets – current balance, active bonus triggers, issue dates, expiry dates
- Wishlist – responses bookmarked for future voting. Participants can save responses they want to vote on when daily limits are reached. The wishlist persists across days and is independent of the daily snapshot.
- Cargo entries – responses that reached threshold and were validated, with on-chain record links (once anchored)
- Current level – role status and criteria for next advancement
- Work queues (Level 2+) – pending tasks (flagged content, community questions, etc.) with queue health indicators

The dashboard serves as the primary retention mechanism, giving participants visibility into their progress and pending work.

Under consideration: Community-specific dashboards may be implemented to provide aggregate participation metrics for institutional partners, with visibility and data-sharing parameters determined through partner consultation.

3.5.5 Notifications

Participants may opt in to email notifications for key events:

- Vote received on a response
- Response approaching cargo threshold
- Response validated as cargo
- Vote tickets approaching expiry
- Wishlisted response activity
- Level advancement
- Work queue alerts
- Bonus ticket earned
- Bonus available

Participants must consent to notifications and can unsubscribe at any time. Email handling complies with applicable data protection regulations.

3.5.6 Community Polls

At key moments, the system may conduct a community poll—a collective binary vote on fundamental questions facing the system. Unlike continuous response validation, community polls address meta-questions about system governance: Should thresholds be adjusted? Should the project timeline extend? Should participation rules change?

Level 3 participants (Curators) may propose poll questions. Level 4 participants (Validators) review and approve questions deemed appropriate for community-wide deliberation. Once approved, all active participants may vote Yes or No, regardless of level.

Results are recorded and guide system evolution. In the participant interface, these moments are referred to as 'Ground Truth Sessions' in the participant interface — reflecting the Mars mission narrative of collective decision-making before departure.

3.5.7 Queue Management & Workload Alerts

As the system scales, work queues for higher-level participants (flagged content review, community question completion, prompt validation) may accumulate faster than participants can process them. To maintain system health and prevent backlogs, the system monitors queue sizes and alerts participants when intervention is needed.

Each work queue has defined thresholds:

- Normal range (green): Queue is manageable
- Alert threshold (yellow): Queue is building, attention recommended
- Critical threshold (red): Queue is backlogged, urgent action needed

When queues enter yellow or red zones, participants at the responsible level receive dashboard alerts and optional notifications. In critical cases, the system may require participants to address queue items before performing other actions, ensuring validation and moderation work keeps pace with participation growth.

This mechanism distributes responsibility: as more participants reach higher levels, the available labour pool for governance tasks grows, allowing the system to scale organically.

3.6 System Versioning Control

The System Versioning Control mechanism governs how the system adapts over time. Adaptation occurs along two dimensions: scale and timeline. Scale-based versioning responds to unpredictable fluctuations in community size, while timeline-based versioning manages the project lifecycle from opening to closure.

3.6.1 Scale-Based Adaptation

Internet-based participation is inherently volatile. Unlike bounded physical communities where attendance is predictable, internet participation exhibits dramatic fluctuations: participants join and leave unpredictably, engagement spikes and drops without warning, and active community size can vary by orders of magnitude over short periods.

This volatility creates a core systems challenge: how to maintain a healthy balance between cargo validation rate and active participation when community size is unpredictable.

Adaptive Threshold Models

The system implements adaptive threshold calculations that respond to current participation scale rather than assuming stable community size. Four threshold models are available, each suited to different scales:

A. Fixed threshold

- Threshold = K (constant value, e.g., 10 votes)
- Suitable for: Small, stable communities (10-50 participants)
- Limitation: Fails when community size changes significantly

B. Percentage-based threshold

- Threshold = $P\%$ of votes cast per a defined period
- Suitable for: Growing communities with predictable engagement
- Maintains proportional validation rate as participation scales
- Limitation: Requires sufficient vote volume at a specific time period

C. Square root threshold

- Threshold = $a \times \sqrt{\text{votes cast}}$
- Suitable for: Medium to large scale with high volatility
- Sublinear scaling prevents threshold from growing too fast
- More forgiving than linear models

D. Logarithmic threshold

- Threshold = $a \times \log(\text{votes cast} + 1)$
- Suitable for: Very large scale or highly volatile participation
- Slowest growth rate
- Maintains accessibility even at high participation

The ticket pool simulator – documented in the prototype section – investigates these threshold model behaviours across different community scales and participation patterns before deployment.

Model Selection Strategy

The active threshold model is determined by observed participation patterns, not predetermined community size targets. The system monitors:

- Active participants per day
- Votes cast per day
- Cargo validation rate (responses reaching threshold)
- Average votes per response in pool

When these metrics indicate the current model is producing unhealthy dynamics (cargo rate too high, too low, or too volatile), the system can transition to a more appropriate model.

Specific transition triggers and parameter values (K, P, a) are subject to calibration through deployment testing and simulation modeling.

3.6.2 Ticket Pool Dynamics

Threshold adaptation must account for a critical behavioural dynamic: not all ticket holders vote.

Participants exhibit varied engagement patterns:

- Some submit responses and vote actively
- Some submit but rarely vote
- Some submit but never return to vote (tickets expire unused)

This "unused ticket rate" reduces effective vote supply. If 100 tickets are issued but 40% of holders never vote, the effective voting pool is only 60 tickets—even though 100 tickets exist in the system.

Adaptive threshold calculations must therefore respond to votes cast (actual voting behaviour) rather than tickets issued (theoretical voting capacity).

Bonus ticket mechanisms (Section 3.2) partially compensate for unused ticket rates by amplifying the voting power of active participants, helping maintain healthy validation rates even when significant portions of the ticket pool remain unspent.

3.6.3 Attention Distribution & View Modes

Vote distribution across the submission pool is governed by the daily snapshot mechanism (Section 3.2). The view mode—Random, Popular, Newest, Oldest, or Neglected—determines which responses receive attention on a given day.

Different view modes serve strategic functions:

- Random — Equalises discovery probability across all responses; no bias toward timing or popularity
- Popular — Amplifies responses already receiving votes (momentum-based validation)
- Newest — Surfaces recently submitted responses, ensuring new content receives immediate attention
- Oldest — Prevents early submissions from being buried by newer content
- Neglected — Surfaces responses with few or zero votes, rescuing overlooked contributions

Participants can freely switch between modes within their daily snapshot. This multi-mode approach distributes attention more evenly than single-mode systems:

- If only Popular mode existed, early winners would dominate and neglected responses would never surface
- If only Random mode existed, momentum-based consensus formation would be impossible
- If only Newest mode existed, older responses would be systematically disadvantaged

The combination allows participants to discover responses through different lenses—serendipitously (Random), by community validation (Popular), by recency (Newest), by longevity (Oldest), or by rescue impulse (Neglected). This multi-modal attention architecture helps maintain healthy vote distribution across the temporal and popularity spectrum of the response pool.

The snapshot window size — the number of responses visible to each participant per view mode — is a critical parameter that must scale with the response pool. A window of 10 responses represents meaningful coverage when the pool contains 50 entries, but becomes negligible when the pool contains 2000 responses. As the system scales through version transitions, window size should increase proportionally to maintain healthy attention distribution across the pool. Different view modes may also benefit from different window sizes — random and neglected modes require broader windows to function effectively at scale, while popular and oldest modes remain effective at smaller windows due to their deterministic sorting.

Under-consideration: View mode relevance shifts across the project lifecycle, and the system should consider retiring or reweighting certain modes in later versions.

3.6.4 Timeline-Based Phases

Beyond scale adaptation, the system must accommodate the project lifecycle. The project operates within a defined timeframe (months, not years) and must eventually close. This temporal constraint requires different operational modes at different lifecycle stages.

Tentative Phase Structure (subject to refinement):

Phase 1: Opening

- All features active
- Prompts seeded by project team
- Invitations distributed to partner organisations

- Community building focus

Phase 2: Active Operation

- Full participation across all mechanisms
- Threshold models adapt to observed participation
- Community polls activated
- Organic growth through participant referrals

Phase 3: Winding Down (final weeks/month)

- Possible feature restrictions to facilitate closure
- Options under consideration:
 - Pause new response submissions (voting continues to clear pool)
 - Pause new prompt proposals (focus on existing prompts)
 - Adjust thresholds to increase cargo validation rate
 - Set final deadline for cargo transition

Phase 4: Closure & Anchoring

- Response submission ends
- Final voting period
- Cargo archive finalized
- Anchor Keeper elected
- Ceremonial on-chain anchoring executed

The specific timeline, phase durations, and feature restrictions during wind-down are intentionally flexible. Different closure strategies produce different participant experiences and different final cargo compositions. The design preserves the ability to activate or deactivate mechanisms (prompt submission, response submission, voting windows) at key moments without requiring architectural changes.

This flexibility acknowledges that the optimal closure sequence cannot be predetermined—it will depend on observed community dynamics, cargo accumulation rates, and the project's theatrical, published, exhibited, or other interdisciplinary goals at that moment.

3.6.5 Version Transition Mechanisms

Transitions between threshold models or timeline phases can occur through:

- Automatic triggers: System monitoring detects unhealthy metrics (cargo stagnation, pool bloat, vote supply collapse) and flags for review

- Admin decision: Project team reviews system health data and executes transitions manually

The system is designed to support both transition mechanisms, with the specific balance between automation and admin control. A monitoring and observability layer will be developed during the implementation phase to track system health metrics and support informed governance decisions and timely version transitions.

3.7 Blockchain Layer

The Blockchain Layer serves two essential functions: permanent anchoring of validated cargo and participant credentials, and distributed verification enabling anyone to independently confirm the authenticity of entries without relying on project infrastructure.

Unlike the continuous validation processes that occur off-chain, blockchain anchoring happens ceremonially at project closure—a singular moment when the collectively validated archive becomes immutable. This permanence is coupled with a multi-layered retrieval architecture: on-chain data enables browsing and verification, content-addressed archives provide human-readable access, and cryptographic proofs allow mathematical confirmation of authenticity.

The system balances three priorities: permanence (data survives server shutdowns), accessibility (multiple pathways to verify entries), and participant access (no requirement to hold tokens or pay transaction fees). Redundant deployment across complementary blockchain networks ensures the archive persists even if individual systems fail—mirroring the backup systems approach of long-duration space missions.

3.7.1 Anchoring Model: Ceremonial Batch Transaction

Blockchain anchoring does not occur continuously during active participation. Instead, all validated cargo is anchored in a single ceremonial batch transaction executed at project closure. This approach:

- Minimizes gas costs (one transaction instead of thousands)
- Creates a singular moment of permanence (theatrical/ritual significance)
- Removes friction from active participation (no wallet required during submission/voting)
- Enables efficient anchoring via Merkle tree structures

The **Anchor Keeper**, a Level 5 participant elected by the community, executes this final anchoring ceremony. This role carries symbolic weight: a trusted community member—validated through sustained labour and collective recognition—performs the act that makes the cultural record permanent.

3.7.2 What Gets Anchored

Two categories of data are permanently recorded on-chain:

A. Validated Cargo Responses

Each cargo entry is recorded with:

- Submission ID – unique identifier
- Model version – which threshold model validated the entry
- Content hash – cryptographic fingerprint (stored in archives, verified via Merkle proof)
- Timestamp – when threshold was reached

B. Participant Credentials

For each participant, the system records:

- Hashed user identifier (privacy-preserving)
- Progression milestones achieved
- Participation period

This creates verifiable proof of labour contribution. The anchored data serves as the permanent record; whether additional interface layers (such as claimable tokens) are implemented remains flexible based on community needs and project resources.

3.7.3 Redundant Network Architecture

The system employs a dual-network anchoring strategy inspired by aerospace redundancy principles. Rather than relying on a single blockchain network, the architecture deploys complementary anchors across two networks—balancing transparency, permanence, and cost-efficiency while eliminating single points of failure.

Prototype development on Polygon (prototype contract:

0x3CCF479641622ea91927fc4266242cF2068D7Ce7) has validated smart contract functionality, transaction workflows, and cost benchmarks through multiple successful test anchors visible on Polygonscan.

Two Complementary Layers

Layer 1: Transparency Layer (Low-Cost Network)

Deployed on an Ethereum Layer 2 network (such as Polygon, Base, or Optimism), this layer stores:

- Compact-encoded submission IDs and model versions – enabling direct browsing on block explorers without downloading archives
 - Example Format: "**1098C8749D26734R...**" where **1098** = submission ID, **C** = model version
 - Data volume: approximately 85 KB estimated (5,000 cargo entries + 10,000 credentials) – subject to verification during implementation.
- Merkle roots – cryptographic verification of complete dataset
- IPFS archive hashes – official distribution references
- Cross-reference to Ethereum anchor (contract address + transaction hash)

Estimated cost: €0.10-0.50

Layer 2: Permanence Layer (Ethereum Mainnet)

Deployed on Ethereum mainnet, this layer stores:

- Merkle roots (identical to Layer 1) – cryptographic verification
- IPFS archive hashes (identical to Layer 1) – distribution references
- Cross-reference to Layer 2 anchor (contract address + chain ID)
- Minimal metadata (cargo count, credential count, timestamp, Anchor Keeper)

Estimated cost: €5-10

Cross-Verification Architecture

Both blockchain anchors reference identical Merkle roots and IPFS hashes, enabling verification through either network:

- Casual verification → Browse Layer 2 block explorer for instant ID lookup
- Academic verification → Verify Merkle proofs against Ethereum mainnet for citations
- Independent verification → Cross-check both networks confirm identical roots and archives

If either network becomes inaccessible or fails, the other preserves the permanent record. This redundancy mirrors the backup systems approach of long-duration space missions: critical data survives through distributed resilience, not singular trust.

Total estimated cost: €5-11

3.7.4 Technical Structure

Merkle Tree Anchoring

Cargo and credential data are anchored using Merkle tree structures. A Merkle tree groups data into pairs, hashes each pair together, then repeatedly pairs and hashes the results until reaching a single root hash. This root acts as a fingerprint for all the data—changing any single piece changes the entire root. To verify any entry is authentic, you need only the entry itself plus a Merkle proof—the sibling hashes along the path to the root.

Rather than storing thousands of individual entries on-chain (prohibitively expensive), the system stores a single cryptographic root hash. This root enables verification of any individual entry through cryptographic proofs while reducing gas costs by orders of magnitude.

Each blockchain anchor includes:

- Cargo Merkle root (verifies all 5,000 validated responses)
- Credential Merkle root (verifies all 10,000 participant records)
- IPFS archive hashes (canonical distribution references)
- Cross-network references (contract addresses, transaction hashes)
- Project metadata (closure date, cargo count, Anchor Keeper address)

Additionally, the transparency layer (Layer 2) includes:

- Compact-encoded strings storing submission IDs and model versions directly on-chain for browsability

Cargo and credential data are verified using binary Merkle trees with SHA-256 hashing. Each entry is serialized and hashed to form tree leaves; pairs are combined and hashed recursively until a single root hash remains. This structure enables efficient verification of individual entries against the on-chain root.

Verification Process

Anyone can verify a specific entry through either network:

Quick verification (Transparency Layer):

1. Visit Layer 2 block explorer (e.g., Polygonscan)
2. View compact string data
3. Locate submission ID (e.g., **1098C** indicates ID 1098, Model C)
4. Immediate confirmation that entry exists and which model validated it

Cryptographic verification (Either Layer):

1. Download archive from IPFS (using hash from either blockchain)
2. Obtain entry data and its Merkle proof
3. Compute Merkle root from entry + proof
4. Compare against on-chain root (from either Ethereum or Layer 2)
5. Match confirms entry authenticity

This verification requires no trust in the project team—only in mathematics and blockchain immutability.

The Merkle tree structure and verification tooling described here have been designed but not yet fully implemented. The complete anchoring workflow — including batch generation, tree construction, proof generation, and archive packaging — will be built and tested during the implementation phase.

3.7.5 Archive Distribution & Retrieval

At the moment of anchoring, the system generates a single archive package containing all validated cargo — full text, metadata, and Merkle proofs — all participant credentials, and a standalone HTML reader for browsing and verification. This package is distributed as a zip file.

The zip file is uploaded to IPFS, generating a content-addressed hash — the CID — which is then anchored on both blockchain networks as the canonical archive reference. The CID is tied to the content of the file, not its location. The zip file may be freely duplicated and distributed — every copy is independently verifiable against the CID on-chain.

Archives are distributed via:

- Project website — direct download while the server operates
- IPFS — retrievable by anyone with the CID, regardless of the project server's status
- Institutional partners — independent copies held by cultural organisations

Once copies are widely distributed, the archive's accessibility no longer depends on any single hosting service. The blockchain provides permanent cryptographic proof. The distributed copies provide human-readable access. Even after the project server is decommissioned, the archive persists through its copies and remains verifiable through the on-chain CID.

Under consideration: the specific tooling and workflow for generating, packaging, and distributing the archive will be finalised during the implementation phase.

3.7.6 What Remains Off-Chain

The following data remains on project servers and is not anchored on-chain:

- Vote tickets (issuance, expiry, usage)
- Intermediate voting data (who voted on what)
- Submission drafts and edits
- Flagged content and moderation records
- Participant email addresses and preferences
- Pre-cargo pool responses (not validated)

Only final validated outputs—cargo responses and participant credentials—are anchored permanently. This separation preserves privacy, reduces costs, and focuses permanence on collectively validated cultural artifacts.

Off-chain data has a lifecycle. Server hosting requires ongoing costs. Before infrastructure is decommissioned, validated cargo will be transformed into other permanent forms such as theatrical integration or physical publication, preserving the work beyond its original digital hosting.

3.7.7 Project Wallet & Governance

Blockchain transactions require gas fees paid in cryptocurrency. The project maintains project-controlled wallets (one per network) funded to cover anchoring costs. Smart contracts restrict wallet use to anchoring transactions only—this restriction is code-enforced, not trust-based.

Funding allocation for gas fees:

- Smart contract deployment (one time, Layer 2): €0.10-2.00
- Layer 2 anchoring (compact strings + Merkle roots): €0.10-0.50
- Ethereum anchoring (Merkle roots only): €5-10
- Reserve buffer for gas price fluctuations: €5-10

Total allocation: €11-23

Smart contract deployment (one-time, Layer 2) – already prototyped on Polygon Mainnet: €0.50-2.00 estimated for production deployment

Actual costs will be determined at deployment based on cargo count, network conditions, and current market rates. The gas fees described here represent only the blockchain anchoring component of the project budget. Full implementation costs – including platform development, UX and UI design, server infrastructure, and operational expenses – will be calculated during the pre-production phase. All cost estimates are based on market rates as of March 2026 and will vary at deployment depending on network conditions, gas prices, and final cargo count.

4. Conclusion

Mars is not really the subject of this project. It is an excuse.

"What deserves to be carried to another planet" is a question that genuinely has no correct answer and no material reward attached to it. You cannot lobby for Mars policy. You cannot profit from what gets archived. You cannot build a career on your Level 5 status. The only thing the system offers is the experience of thinking together about something that doesn't exist yet.

That changes the psychology of participation completely. When nothing is at stake materially, the only reason to participate is the imagination itself. When there is no correct answer, the performative voter has nothing to perform toward. What's left is closer to play – in the deep sense of the word. Imaginative, voluntary, purposeless in the best way.

This fictional frame is not a gimmick of the project. It is the condition that makes everything else possible.

Theatre has always known this. The stage is also a fictional frame that produces genuine emotion, genuine reflection, genuine encounter – precisely because everyone agreed in advance that none of it is real. The fiction is not decoration. It is the architecture of trust.

Influence earned through participation

Most internet platforms convert either money or algorithmic favour into influence. Engagement metrics, follower counts, token holdings – these are the currencies that determine whose voice carries. ACT 0: FILL THE CARGO FOR THE RED PLANET proposes something different: that the only currency which earns you more say in what gets preserved is showing up and doing the work.

This is not the model of Wikipedia, where authority derives from correctness and citation. It is not the model of social media, where virality and capital amplify certain voices over others. It is closer to civic consultation – but unlike civic consultation, where one person casts one vote regardless of engagement, influence here is proportional to sustained contribution. The participant who returns, completes tasks, advances through role levels, and earns bonus tickets through consistent labour has more voting capacity than one who appeared once and left. Not because they paid more. Because they gave more of themselves.

The project does not claim to measure genuine intent. It claims that collective behaviour, over time, across enough people, produces something worth preserving –

regardless of what was in anyone's heart when they voted. This is how democracy works in practice. People vote for all kinds of reasons – habit, identity, social pressure, genuine conviction. The system does not try to distinguish between them. It counts, and trusts that the aggregate reveals something real about collective priorities even when individual motivations are mixed.

The arc of the system

Validation in ACT 0: FILL THE CARGO FOR THE RED PLANET is a layered process that accumulates across time and across many independent judgments.

A participant chooses a category and receives a prompt – already a decision about what question feels worth answering. They write something – a curatorial act, a choice about what to express. When voting, they choose a view mode that reflects their intention: random for open discovery, neglected to rescue overlooked voices, oldest to protect early contributors. Within that mode they see a window of entries and choose which ones deserve their attention. Each of these choices is itself a form of judgment.

By the time an entry reaches cargo it has survived multiple days, multiple participants, multiple view modes, multiple attentional contexts – and still accumulated enough votes. That is much richer than a simple vote count suggests. The threshold number is the final gate. The real validation is the entire journey.

At project closure, the Anchor Keeper – a participant elected by the community, trusted through sustained labour and collective recognition – executes the ceremonial anchoring. The validated cargo becomes immutable. The record cannot be changed or deleted by anyone, including the project team. The communities that crossed paths in this shared fictional space – institutions, peer networks, strangers connected through chains of invitation – disperse back to their separate worlds. But the crossing happened. And the blockchain makes it permanent in the most minimal way possible: not a social network that keeps them connected, not a platform that monetises their relationship, but a mark that says this was chosen, at this time, by these people.

What this work contributes

The prototype work described in this document has begun to answer several of the research questions posed at the outset. The simulator confirmed that no single view mode is sufficient for healthy validation – the multi-modal attention architecture is not a design preference but a functional necessity. It demonstrated that window size can override threshold differences entirely, identifying attention distribution as a critical versioning parameter alongside threshold models and bonus ticket mechanics. The

blockchain prototype confirmed that labour-based validation can be anchored permanently on a public network without financial barriers to participation.

What remains open – how validation quality relates to community scale, whether proportional consensus equalises legitimacy across different sizes of community, how participation volatility affects cargo accumulation over a full mission period – will only be answered through deployment and observation. These questions cannot be resolved through simulation alone.

This work contributes three things to the broader field. First, adaptive threshold models for internet-scale participation with fluctuating community sizes – maintaining meaningful collective consensus when participation varies by orders of magnitude. Second, a labour-based validation architecture where influence derives from effort rather than capital – demonstrating that substantive collective curation can occur without financial barriers or token-based governance. Third, architectural patterns for cross-institutional collaboration that preserve participant agency without centralised control – enabling distributed communities to coordinate validation across organisational boundaries through transparent, rule-based mechanisms.

The cargo archive is not a claim about what humanity values

It is a record of what this constellation of people, at this moment in history, decided was worth carrying forward. That specificity is not a limitation. It is the archive's most honest quality.

The mission runs for four to eight months depending on the scale of invited communities and available resources – within the orbital window between departure and Mars opposition. What accumulates during that journey, validated by the collective judgment of everyone who participated, will be permanently anchored at the moment of closure – whether that moment is departure or arrival is itself a curatorial decision yet to be made. What gets made from that archive – theatre, publication, exhibition, lecture – depends on what the world makes possible. The system does not need to know in advance. It just needs to collect well.

The fiction is not a gimmick. The labour is not a barrier. The permanence is not a technological gesture. Together they create the conditions for something that is otherwise very difficult to produce at scale: a structured space for collective imagination, free from commercial motive, free from correct answers, free from algorithmic manipulation – where the only thing that matters is what people, together, decided was worth preserving.

That is what this project is for.

Acknowledgements

This project was developed during the Akademie Schloss Solitude Web Residencies 2026. The research, system design, and prototypes presented here were made possible through the support of the residency programme. The author is grateful for the time, support, and infrastructure that made this work possible.

Thank you: Sarah Donderer, Barbara Cueto, Grayson Earle, Patrick Urbani. And my fellows at the residency: Pavan Vadgama, Laura Fong Prosper, Joshua Kroon, beatnyk, and Rok Kranjc.

This document is a working paper. Content is subject to change.

© Lam Lai 2026. All rights reserved.