The Coordination Contract

A Kernel, a Law, an Engine and a Ledger

An institution with no offices: people change its law by interpreted voice, a pinned set of models decides through recorded determinations, the market is free of politics, and justice reaches everyone.

Abstract

The coordination contract is a voluntary institution for commitments between people, organizations and agents. It has four parts: a **kernel** that never changes; a **law** that changes only by an adopted RFC; an **engine**, a published and pinned set of models, that also changes only by an adopted RFC; and a **ledger** of accounts, agreements, cases and determinations. The contract acts in one way only: through **determinations**. A determination is a closed question over committed inputs. The contract puts it to the engine tier by tier and records every seat's answer. There are no branches, executives, officers, councils, juries or human adjudicators. No key lets its holder change the contract outside an adopted RFC.

People are a predicate, not a fixed category. Humans are people permanently. The law may recognize further kinds: uploaded humans, artificial agents, animals through representatives, and organizations. Each kind has its own uniqueness rule, so that one person has exactly one voice. Wealth, stake and intelligence add nothing to that voice.

An RFC pins the engine, the contract and the procedure when it opens. It accepts receipt-confirmed comments in admitted media. Interpretations remain contestable. By default, the engine deliberates over merits, objections and participation: scores describe the comments but do not bind its choice. One comment represents its author. Self-selected participation is not a population sample, and silence is not consent. An explicitly binding-score RFC instead requires a positive unique favored option. Both procedures require confident concurrence. To replace the engine, the outgoing engine and then the incoming engine must concur. Anonymous responses use Semaphore proofs over the pinned commitment tree, with one nullifier per identity and RFC. Network and content metadata remain possible links.

Politics has no power over the market. Law may not set prices, single out holders, freeze or seize holdings, or forbid exchanges between consenting parties. The one exception is that law may prohibit harm to a person who did not consent. Justice does reach the market. A claim is justiciable when it names all of these: a harmed claimant; a respondent; an act; a duty from an accepted agreement, from law in force at the time, or from the kernel; and resulting harm to a protected interest. Cases, civil and criminal, follow one path. They are screened for repetition and served before their clock starts. One fixed panel decides each case element by element. It compares the claimant's and the respondent's figures for each harm, and it justifies every remedy separately. A timely appeal stays execution, and fresh seats hear it. A prolonged outage closes the case unprocessable, and the prior judgment is not enforced. There is no human jury. A final judgment executes against the respondent's holdings, bonds included, in one atomic step. Eligible victims share a protection pool. Bond losses require insured-1 coverage and independent verification, and premiums are paid only on insured issuance. The pool pays a covered loss in equal proportion with every other unpaid victim across allocation epochs, and the pool takes over the claim. The contract services credit itself: an investor's mandate, a borrower's facility, its bond, its servicing and its default all run through the same rules.

The engine has four tiers: deterministic checks; an open-weight embedding model with pinned INT8 MLP heads that anyone can rerun bit for bit; a panel of fast classifier seats; and a panel of large deliberative models whose answer is final or unresolved, never a winner by label order. A panel is exactly a quorum of independent groups. The top tier holds at least two disjoint panels, so that every top-tier decision stays appealable. Every model tier must contain seats from at least two operators and at least one open-weight seat pinned by hash. Seats of one operator or lineage count once. Media reaches the deciding seats only as text, rendered by at least two independent renderers. Every seat reads one identical record.

The record is built so that a reader can check it without trusting its writer. A seat may name an attestor; every answer of that seat must then carry the attestor's signature. Acts are appended to a public hash-chained log. The log's checkpoints support anchoring, but they are not independently anchored at genesis. A replica recomputes every derived outcome from the public record and states exactly what it cannot see. On chain, one registry is the only authority. The escrow, a notary of command authority and mirrors on other chains all read the registry's seats and keep no signer of their own.

The contract core is implemented in this repository as a dependency-light TypeScript module. It covers the kernel, people, law, engine, MLP heads, determinations, evidence, RFCs, anonymous voice, justice, pool, credit, attestations, the log and state transitions. The hosted service runs RFCs, cases and credit on the core and commits every money movement as one claimed, conditional batch. The on-chain programs and the EVM mirror are built and tested and not deployed. The core generates every contract number in this paper. No live engine has decided an RFC or a case. No one but the platform attests a seat yet. This paper does not claim legal enforceability or measured model accuracy.

The Record Every Act Leaves. Which question was asked, over which committed inputs? Which seats answered what, and how did their answers aggregate? Which kernel article, law version and engine hash governed it? What was executed, and what remains owed?

Design and reference implementation. Not a deployed institution, audited protocol, securities offering or claim of legal sovereignty. All people, cases and amounts in examples are synthetic.

Status of This Revision. This edition describes a new kernel. It is not an amendment applied to the deployed ledger. The worker refuses a ledger with another kernel hash. Deliberative consideration is the default; only an explicitly binding-score RFC is bound to a favored score. Uninsured bonds pay no premium and have no pool coverage. Insured-1 requires independently verified losses between unrelated parties. Criminal enforcement requires an adopted engine-bound calibration report. Unavailable appeals close unprocessable after the processing window, and the first judgment is not enforced. Independent attestors, an anchored registry, audited calibration and an identity-resolution issuer remain external activation requirements. The host is still trusted for availability, completeness and panel entropy.

Contents

1 The Institutional Coordination Problem 2
2 Startup Finance and the Continuity of Missions 4
3 Definition, Boundary and Lineage 6
4 The Kernel 7
5 People and Law 10
6 RFCs: Interpreted Voice 11
7 The Engine 15
8 Justice 18
9 Evidence Without an Official Truth Oracle 22
10 Bonds, the Protection Pool and External Notices 24
11 Entities, Contracts, and the Economic Graph 25
12 Treasury, Debt, and Sources of Repayment 27
13 Preferential Capital and Thematic Credit Coordination 29
14 Economic Measurement and Bounded Stabilization 31
15 Agents, Settlement and Operational Continuity 32
16 Fault Tolerance: A Record Anyone Can Check 34
17 One Trust Root, Many Chains 36
18 Implementation Status, Release and Migration 37
19 Evaluating the Engine, and the Limits 41
A Records 43
B Transitions 44
C Genesis Law and Engine 44
D Worked Cases 47
E Mandate-Constrained Allocation Mathematics 49
F Cash Flows and Loss Allocation 50
G Economic Measurements and Bulletin Schema 50
H Threats and Controls 51
I Implementation Status 53
J Reference Tests 56
K Commitments and Decisions 59
L Glossary 60
M Sources and Build 61
N The Live Canary 61
References 62

Reading Guide and Status

The core. Section 3 defines the four parts. Section 4 is the kernel, verbatim, with its hash. Section 5 covers people and law, Section 6 RFCs, Section 7 the engine and determinations, Section 8 justice, and Section 10 bonds and the protection pool. Appendix D works through every rule with numbers generated by the implementation.

The context. Sections 1–2 motivate the institution. Sections 9 and 11–15 cover evidence, the economic graph, treasury, capital, measurement and execution. Most of that material carries over from version 0.3. Section 16 covers the record that anyone can check: attestors, the log, checkpoints and the replica. Section 17 covers the one trust root on every chain. Section 18 states what exists and what does not, and sets every threat against what is built. Section 19 states how the engine would be evaluated. Appendix N is the one place where a live model answers. A short edition of this paper states the contract in about ten pages.

What changed in 0.5. The kernel is unchanged (the same hash as 0.4.1). Version 0.5 adds anonymous voice (Section 6.3), seat attestors (Section 16.1), the transparency log and its checkpoints (Sections 16.2 and 16.3), the replica (Section 16.4), the registry-gated command notary and the EVM mirror (Section 17); Section 18.3 lists them.

What changed in 0.4.1. An external review of version 0.4 found ordinary code paths that contradicted its guarantees: a tie at the top tier decided by label order, appeals the genesis engine could not hear, image evidence that made the engine unable to decide, seats that could receive different evidence, credentials that a proposed engine could steer, timely appeals a late sweep could overtake, remedies that multiplied, non-atomic execution, a first-come pool, and no implemented financing cycle. Version 0.4.1 changes the kernel (a new hash) and the implementation to close each of them; Section 18.4 lists them with what replaced them.

What changed from 0.3. Version 0.3 specified a profile with eleven offices, binding votes with quorum and supermajority thresholds, consultations, a three-reviewer constitutional closure, human-only democratic feedback as an immutable rule, absolutely non-seizable bonds, and vacatur of adverse decisions whenever an appeal panel was unavailable. Version 0.4 removes the offices and thresholds and makes people an extensible predicate. It lets justice reach bonds and forbids politics to touch them. It adds the protection pool and makes the engine part of the contract. Under version 0.4, an appeal that cannot be heard leaves the standing judgment final. Appendix K separates the author's commitments from the design choices made for this version.

Provenance. Design of versions 0.4 to 0.5, the reference implementation, editing and typesetting used AI assistance (Claude). All people, operators, cases and amounts in examples are synthetic.

1 The Institutional Coordination Problem

1.1 Start with the People Making Commitments

A founder wants to purchase equipment and hire two collaborators. A lender wants to know whether the resulting business can repay. An employee wants to understand the difference between a headline valuation and an exercisable ownership claim. A customer wants delivery, not a convincing demonstration. A citizen wants to know which rules

will apply after committing savings or changing work. A faster payment alone satisfies none of these needs.

Consider a small manufacturing project. A corporate registry records its identity. A financial institution holds its money. Documents hold its product requirements. Employment agreements and account permissions split its authority. Invoices, supplier contracts and conversations scatter its obligations. An agent might operate each application competently and still lack a reliable answer to a basic question: *may this entity make this commitment, and what follows if it cannot keep it?*

The coordination problem is therefore partly computational and partly institutional. Institutions structure interaction through formal and informal constraints. North's account emphasizes their role in exchange and uncertainty; it does not reduce them to software or to individual leaders [1]. The reference design does not try to encode all informal norms. It asks which consequential commitments can become explicit enough to reduce the repeated reconstruction of authority, obligations and evidence.

Definition: Coordination Cost. For a specified agreement, coordination cost is the resource and human burden of discovery, verification, negotiation, authorization, financing, monitoring, dispute resolution, and settlement. It includes waiting time, administrative labor, uncertainty, accessibility barriers, and the cost of correcting mistakes. It is not identical to transaction fees. Minimizing it does not justify removing consent or due process.

1.2 A Taxonomy of Institutional Mismatch

Fragmented state. A shared vocabulary does not ensure a shared record. Two systems can disagree about ownership, authority, available cash or completion. Reconciliation is a substantive task, not only an interface problem. The BIS's 2023 discussion of programmable financial infrastructure describes related frictions between messaging, reconciliation and settlement. Its proposed institutional architecture differs from the one examined here [2].

Uncertain change. Rules can change after a participant has made an irreversible investment. Some changes reflect necessary adaptation. Others may be poorly announced, ambiguous in scope or unexpectedly retroactive. The design problem is to distinguish legitimate adaptability from avoidable surprise. A participant needs an identifiable effective date, the governing version, a transition rule, and a remedy when procedure is violated.

Weakly specified forward guidance. Existing institutions do provide guidance and consultation. It would be inaccurate to motivate this project by asserting that they are universally absent. The Federal Reserve describes forward guidance as communication about the likely future course of monetary policy [3]. OECD work documents existing consultation practices and also remaining problems with inclusion and feedback [4]. The proposed additional capability is an inspectable relationship between announced intentions, binding activation procedures and executable consequences. It is not the invention of institutional communication.

Personal discretion and succession. An institution may depend on a few people to remember commitments, interpret permissions or choose when to act. This can support contextual judgment. It also creates risks of succession, conflict of interest, availability and accountability. Replacing an official with software does not automatically remove judgment or power. Those powers may move to maintainers, model vendors, interface operators or upgrade-key holders.

Delayed remedy. A person may discover an error promptly and still be unable to obtain a timely correction. Faster execution can worsen this asymmetry if it accelerates adverse decisions without accelerating review. The design therefore treats appeal access and bounded remedies as part of execution, not as an optional support channel.

Horizon mismatch. A project can require years of continuity while funding, personnel or policy commitments last months. Promises of support are not funded obligations. Conversely, indefinite commitments can preserve unproductive arrangements and prevent adaptation. An institution needs explicit horizons, staged review, and an orderly way to discontinue a failing program.

1.3 Predictability without Pretending That Circumstances Stop Changing

The economic-policy-uncertainty literature supplies empirical methods for studying uncertainty, including newspaper-based indices. It does not establish that any one governance architecture eliminates that uncertainty [5]. This paper separates three quantities that are often conflated: uncertainty about *rules*, uncertainty about *observations*, and uncertainty about *outcomes*. A contract can reduce the first by publishing a rule and its amendment procedure. It cannot make a supplier reliable, an economic model complete, or future revenue certain.

An institutional commitment can be represented as

$$\mathcal{K} = (\text{scope},\ \text{authority},\ \text{rule version},\ \text{conditions},\ \text{horizon},\ \text{remedy}). \tag{1.1}$$

The commitment is stronger when people can inspect and contest these fields. It is weaker when a hidden administrative path can override them. A scheduled change and a state-contingent change are different instruments. "The fee becomes 1% on a specified date" differs from "the fee may change within a bounded range when a published condition persists." Both differ from a forecast of what officials expect to do.

Forward guidance also has costs. The Federal Reserve's discussion of the tool identifies a tension between credibility and flexibility when circumstances change [6]. The analogous protocol design publishes escape clauses before they are needed, defines their narrow authority, and records when they are used. It does not promise that emergencies will never occur. It does not claim that an unchangeable formula is always preferable.

1.4 Incident Classes Rather Than a Catalogue of Political Grievances

The motivating incidents can be tested without assuming malicious motives. A subsidy changes after equipment is ordered. An authorized signer departs before payment. A funding tranche is announced but never irrevocably committed. An interface displays an obsolete law. An automated decision uses stale evidence. A treasury guarantee proves broader than voters understood. These are *hypothetical failure scenarios*, not allegations about a named government or company.

Each scenario yields a technical requirement. Commitments need effective dates and version references. Authorization needs expiry and succession. Funding needs a distinction between a soft intention and an escrowed obligation. Interfaces need freshness proofs. Consequential decisions need evidence snapshots and correction procedures. Guarantees need notional caps, triggers, loss allocation and public accounting.

This framing preserves an important symmetry. The proposed system must accept the same scrutiny that it applies to existing institutions. A last-minute model update, an unexplained credential revocation, a rushed emergency proposal or a unilateral software migration would reproduce the uncertainty that the project intends to address.

1.5 Why Agents Intensify the Problem

Autonomous execution increases the number and speed of actions that can be attempted under ambiguous authority. A person can supervise a few purchases manually. An organization that delegates thousands of decisions needs machine-checkable boundaries. The relevant distinction is not human versus machine competence in the abstract. It is *who supplied the objective, who authorized the action, which constraints apply, and who bears the consequence.*

This authorization gap motivates the coordination contract. The contract makes no claim that every agent is reliable or that general-purpose applications will disappear. CommandAGI's interfaces, including computers, communications and physical-system adapters, supply capabilities. The institution supplies the permissible relationship between intent and consequential action. These functions are complementary but separable.

> ***The Central Hypothesis.*** Explicit, inspectable institutions may reduce coordination cost and make some durable projects easier to finance. They may also impose reporting burdens, concentrate power, and expose participants to shared software failures. Superiority to existing arrangements is an empirical question. The paper specifies what would have to be measured.

2 Startup Finance and the Continuity of Missions

2.1 What the Startup Observation Adds

The investor commentary supplied as motivation describes four things. Founders leave ventures for larger laboratories. Larger firms transfer selected teams. People worry about employees' option economics. Investors find it difficult to distinguish many small sources of product defensibility. The paper treats the passage as a supplied investor perspective. It is not a representative survey, and it is not evidence that mobility is inherently harmful [7].

The institutional issue is narrower and more general. A venture's public mission, contractual obligations, productive assets, team membership and economic rewards can stop moving together. A founder may rationally change jobs. An investor may rationally pursue liquidity. A customer may still require support. An employee may remain exposed to a claim whose practical value they cannot assess. A system that treats the founder's enthusiasm as the only continuity mechanism leaves these contingencies underspecified.

Mission continuity is not personnel captivity. The proposed response attaches commitments to funded programs, intellectual property, contracts and transferable operating responsibilities. It does not oblige a person to remain enthusiastic or employed indefinitely. A person may leave. The institution should make visible what then happens to outstanding promises, customer support, funds and other participants' claims.

2.2 Make the Commitment Structure an Economic Object

A *mission record* contains a stated purpose, beneficiaries, time horizon, eligible activities, funding sources, measurable milestones, known uncertainties, stewardship roles, and an amendment or termination procedure. It can belong to a private project, a thematic fund or a publicly authorized program. It does not confer political authority, prove social value or guarantee a return.

Separating the mission record from the operating company permits a useful distinction. A company is an accountable executor with assets and liabilities. A mission is an objective to which participants may voluntarily commit resources. One company may execute several missions. A mission may outlive a particular company when existing contracts and rights permit succession. Any transfer must respect intellectual property, employee rights, creditor claims, privacy and external law. Copying a database is not a lawful transfer of a business.

2.3 Funding Fit Matters More Than the Label on a Token

Not every valuable project is debt-financeable. A borrower with contracted receivables or equipment-backed operations may have a plausible repayment source. A pre-revenue research effort with highly uncertain technical success may instead require equity-like risk sharing, grants, sponsorship, advance purchase commitments or philanthropic capital. The protocol cannot turn long-horizon uncertainty into low-risk fixed income by describing it with an embedding.

Three financing states must remain distinct. An *expression of interest* is revocable unless a contract says otherwise.

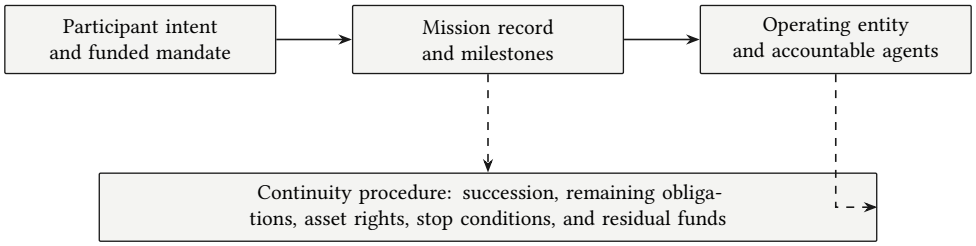


Figure 2.1. Mission continuity is a relationship among mandates, commitments, and operators. It is not ownership of the people doing the work.

A *conditional commitment* specifies funding conditions and the consequences of not satisfying them. A *funded commitment* has identified assets reserved under explicit release and return rules. Interfaces should not display all three as equivalent runway. Published milestones should distinguish verification of a completed deliverable from a forecast that the next one will be completed.

Staged finance can reduce capital at risk. But milestones can also be manipulated or become inappropriate as research evolves. A reference mandate therefore permits an explicitly bounded amendment process, not silent goal substitution. A proposed change exposes its effect on mission, risk, use of proceeds and affected claims. Investors can consent where their contract requires consent. They cannot use a private financing amendment to rewrite the contract's law.

2.4 Employees, Contributors, and Asymmetric Exit

An ownership percentage is not a complete description of employee compensation. A useful claim record includes vesting, exercise price, exercise window, transfer restrictions, liquidation priority, dilution scenarios, and the difference between preferred and common claims. These disclosures help people understand an agreement. They do not alter tax law.

The supplied commentary refers to "409As" and early exercise. The technically relevant distinction is among four things: a financing valuation, the fair market value of the applicable common stock, the option's contractual exercise price, and possible tax consequences. An existing option's exercise price does not automatically reset upward because a later valuation rises. U.S. tax treatment varies by option type and transaction. IRS guidance distinguishes statutory and nonstatutory options, and Section 1.409A-1 contains relevant stock-right valuation conditions [8, 9]. The paper therefore treats affordability and legibility as design concerns. It does not claim that a protocol can make exercise tax-free or universally affordable.

Selective team transfers motivate a *change-of-control and continuity disclosure*. It states which contracts continue, which intellectual property moves, which workers receive offers, what happens to remaining employees, and how existing securityholders are treated. The protocol cannot require a buyer to employ everyone. It can require that representations made within its own financing process match the actual rights being sold. It can also expose a mismatch before another person joins.

2.5 Defensibility That Can Be Inspected without Publishing Every Secret

A product may depend on many small operational advantages, not on a single patent or model. A financing interface could make some of these advantages inspectable: retained customers, independently confirmed delivery, permissioned reliability records, recurring contracts, documented supplier relationships, or a succession plan. These are evidence categories, not a universal moat score.

More transparency is not monotonically better. Publication of customer lists, internal failure traces, pricing or employee data can destroy commercial value and personal privacy. The design uses selective disclosure and independently attested summaries. A lender may verify the existence and concentration of receivables without receiving a public index of every customer's identity. A rival should not obtain another company's trade secrets only by registering as an investor.

There is also a feedback hazard. When existing activity becomes easier to verify, incumbents with long histories can gain a disproportionate advantage. First-time founders, novel research, and work whose value is not captured in transactions need an explicit path to consideration. That path may require higher-risk or concessionary capital. To disguise such support as fully commercial low-risk lending would obscure who bears the cost.

2.6 The Cosmic Imperative as Motivation, Not Compulsory Doctrine

The author connects this financing problem to what he calls a *cosmic imperative*. This paper does not specify the term's complete philosophical content. The operational connection used here is limited. Commitments to work with very long horizons need institutions that can preserve knowledge, resources and human agency through changes in personnel, technology and incentives.

This motivation does not require every participant to adopt the same long-term objective. An investor interested in manufacturing, a participant funding care services, and a researcher pursuing foundational knowledge may use the same coordination infrastructure under different mandates. The protocol's task is to make those commitments intelli-

gible and accountable. It is not to collapse them into the founder's purpose.

An ambitious mission also does not override present people. A claimed future benefit cannot erase a person's right to understand a contract, refuse work, protect private information or challenge a harmful decision. The paper evaluates durable mission funding together with opportunity cost, concentration, effects on outsiders, and the possibility that a mission should stop. Suppose a contract's founder could invoke an undefined higher imperative to bypass its procedures. That contract would contradict kernel article k3: nobody holds authority over the contract, its founder included.

2.7 What Would Demonstrate Improvement

The relevant outcomes are these: continuity of contracted service after personnel changes, predictability of committed funding, intelligibility of employee claims, time to replacement stewardship, and the fraction of completed work that remains usable after an organization changes direction. Raw founder retention is not an adequate success metric. Retaining an unwilling person may reduce welfare while it preserves an attractive statistic.

These outcomes connect startup finance to the larger institutional thesis. A durable economy needs more than efficient formation of companies. It needs understandable ways for commitments to survive, change, transfer or terminate. Thematic credit is one instrument within that process, not the universal answer to mission financing.

3 Definition, Boundary and Lineage

3.1 The Coordination Contract

Definition. The coordination contract is a voluntary institution made of four parts. The **kernel** is a fixed text and a set of numbers that code enforces; it never changes. The **law** holds the kinds of people the contract recognizes, the duties it enforces and its parameters; it changes only by adopted RFC. The **engine** is the pinned set of models through which the contract decides; it is part of the contract and changes only by adopted RFC. The **ledger** holds accounts, agreements, RFCs, responses, cases, judgments and determinations. The contract acts only through **determinations**.

"Contract" names a shared commitment, not one large smart contract. The implementation combines a pure reference module, a hosted service, on-chain records and external legal arrangements. Program validity, constitutional validity, solvency and external enforceability are different properties. To demonstrate one does not establish another.

3.2 Determinations Are the Only Act

A determination is a closed question — a prompt and a fixed set of answer labels — over a canonical evidence package. The contract puts it to the engine tier by tier until a tier answers decisively (Section 7). The package is text only, and each item has a unique identifier. Media reaches the package as renderings (Section 9.3). The determination's record holds the question, the package's digest, the engine hash, every round and the outcome. A round records which seats sat, what each seat answered, the digest of what each seat was actually sent, and how the answers aggregated. The hash of all of that identifies the record. Anyone who holds the record can check the aggregation. Anyone who holds the package and the open-weight seats can rerun the deterministic tiers.

An outcome is one of three things. **Decided**: a tier answered with a unique label at its confidence band. **Unresolved**: the top tier answered, but its groups were tied or its confidence fell below its band. **Unavailable**: no tier could seat a panel that answered. Finality means that the procedure ended. It does not mean that some label had to win. Each use of a determination states what an unresolved or unavailable outcome means for it.

Every consequential act is a determination or a deterministic consequence of one:

- Interpreting an RFC response into a score for each option (Section 6.4);
- Concurring or withholding concurrence in an RFC's adoption (Section 6.7);
- Screening a filing for a matter already brought; deciding justiciability, each element of the duty, which party's figure is closer to each proved loss, whether each remedy is justified, and an interim freeze (Section 8);
- Deciding whether a service agreement's terms were met.

Pure functions with integer arithmetic derive the rest from those determinations: adoption of an RFC, a judgment's remedies, an execution plan, a pool allocation and a facility's servicing.

3.3 No Offices

The predecessor and version 0.3 of this paper assigned powers to offices: branches signing amendments, prosecutors choosing cases, registrars, enforcers, closure panels. Each office was a place where a person, a key or a model could hold power that the procedure did not fully constrain. Version 0.4 has none (kernel k3).

- *Who decides?* The engine's seats, answering a published question, aggregated by a published rule.
- *Who files?* Anyone; the contract itself files when its own records show a breach.
- *Who executes?* The ledger, applying a final judgment's execution plan; the settlement program, applying a signed determination.
- *Who changes the rules?* The people's voice, with the engine's limited concurrence.

Independence is no longer separation between people who hold offices. It is independence between engine seats, under four rules. Seats of one operator or lineage count once. Every model tier must include two operators and an open-weight seat. The top tier must hold two disjoint panels. A seat never sits in a matter to which its operator is a party, or an entity that the engine records as controlling the operator is a party.

3.4 Four Kinds of Standing

Personhood gives a voice in RFCs (Section 5.1). **Contractual capacity** is the ability to accept agreements and bear their duties. **Protected interests** are what justice protects. They belong to every being with interests, person or not (kernel k1, k2, k6). **Delegated capability** is permission to act for a principal. None implies another. An organization can accept agreements without being a person. An animal's interests are protected whether or not the law recognizes animals as people. An agent can hold a capability without a voice. A person can have a voice without a single agreement.

3.5 Trust and Its Perimeter

The design does not promise a trustless factual world. A signature identifies an authorized statement; it does not establish that a camera shows the relevant factory. A personhood proof establishes a key under an issuer's rule; it does not establish that the issuer never errs. A final chain transfer does not establish rightful ownership of a physical asset. A determination establishes what the engine answered; it does not establish that the answer is true. Claims of improvement must measure these dependencies. They must not count the people removed from a workflow.

3.6 Lineage

3.7 Retain, Change and Retire

Retain: accounts, signed frames and certificates, evidence commitments, the evaluator's typed leaves and deterministic aggregation, confidence-weighted quorums, juror independence checks, execution receipts, capability attenuation and accounting.

Change: the evaluator becomes the engine and is pinned in the contract, not in each agreement. The committee that signs settlement frames becomes the engine's seats, recorded in the registry. The appeal becomes a fresh top-tier panel, not a rerun of the same evaluator. The grey zone climbs tiers; it no longer ends at a human administrator.

Retire: the four branches (command, law, reserves, rights) and their seats, signatures and vetoes; the stake-weighted agent franchise — `franchiseWeight` gave a bonded agent up to ten votes by stake [30]; quorum and supermajority thresholds; stakeholder-class ratification; administrator dispute resolution; the upgrade multisig of named offices. Legacy records stay intact as history. Nobody relabels them as conforming.

3.8 One Lineage

The contract keeps one canonical history. Copying its code does not transfer its ledger, people or obligations. A different kernel defines a different contract with no claim on this one's ledger (kernel k4).

4 The Kernel

The kernel is the part of the contract that never changes. Its twelve articles appear below exactly as they are hashed, lowercase typography included, because any change to that text would change the hash. A different text, a different floor or a different list of protected interests produces a different hash. A different hash is a different contract with no claim on this one's ledger.

4.1 The Articles

k1. truth and justice The contract serves truth and justice. Truth: fidelity to evidence, honest uncertainty, attribution, and correction of known error. Justice: protection of every being's protected interests against wrongful interference, by anyone, including the contract itself. No gain to the contract's revenue, size or survival outweighs either.

k2. people and voice Protection belongs to every being with interests; voice belongs to persons. A person is an entity of a kind the law recognizes as people, holding a valid personhood proof for that kind. Human is a kind of people and always will be. The law may recognize further kinds by adopted RFC. Each kind has one uniqueness rule, and every issuer of that kind's proofs derives the same key for the same individual, so that one person cannot speak as many. Each person has exactly one voice in every RFC, given by their own deliberate act. Wealth, stake, activity, intelligence and position add nothing to a voice.

k3. no offices No person or entity holds authority over the contract. The contract acts only through determinations: a pinned question, committed inputs, the engine's seats, a deterministic aggregation and a published record. There are no branches, executives, officers, councils, juries or human adjudicators, and no key whose holder can change the contract's code, law or engine outside an adopted RFC.

k4. change only by voice Law and engine change only by an adopted RFC whose procedure is fixed when it opens. Deliberative consideration is the default: the engine considers each person's comments, substantive objections, participation and its representativeness, then records a reasoned choice, including the status quo. Scores describe comments and do not bind that choice. One person's comment represents that person; silence is not consent and self-selected participation is not a population sample. An RFC explicitly using binding score voting may adopt only its positively and uniquely favored change. In either procedure the engine must concur with confidence, consistent with this kernel and the scope of law; a divided or uncertain engine does not concur. An engine replacement requires outgoing then incoming concurrence; the proposed engine holds no private record or credential before adoption. If the outgoing engine cannot answer for a further window as long as the RFC's own, the proposed engine's concurrence suffices. A change to code is a change to the engine. This kernel is never changed; a different kernel is a different contract.

k5. a free market Politics has no power over the market. Law may not set prices, tax or single out particular holders, freeze or seize holdings, or restrict an exchange between consenting parties, except to prohibit harm to a protected interest of a being who did not consent. Every duty the law

Table 3.1. Implementation lineage.

Period	Record	Status in this paper
June 2026	Author's coordination/trust predecessor	Author report
11 July 2026	Certificate-state-machine devnet escrow	Repository runbook [33]
16 July 2026	Schedule-over-signed-frames escrow	Repository runbook; unaudited
30 September 2026	Version 0.2: static review of the predecessor	Pinned commit
1 October 2026	Version 0.3: RP-03 profile with offices and thresholds	Superseded
1 October 2026	Version 0.4: kernel, law, engine, ledger; reference module	Superseded by 0.4.1
2 October 2026	Version 0.4.1: the hardening pass; a new kernel	Implemented in this repository; hosted service only

creates names the protected interest it protects, the harm it forbids and the elements that make a breach, and binds everyone alike.

k6. the scope of justice A claim is justiciable when it names a claimant who was harmed (or anyone filing for a being who cannot), a respondent who was served, an act or omission, a duty the respondent owed — from an agreement the respondent accepted, from law in force at the time of the act, or from this kernel — and harm to a protected interest or an agreed term caused by that act. One matter is one case. Justice reaches the market: a final judgment executes against the respondent's holdings, bonds included, for the harm established and its costs, once per harm, and nothing more.

k7. due process without juries Every case gives notice, time to answer, an opportunity to submit evidence, a reasoned determination of each element of the duty, and an appeal to seats that took no part in it; the engine keeps enough independent capacity for that appeal. The times to answer and to appeal are at least the kernel's floors; in a civil matter arising from an agreement in which every party accepted expedited procedure, they may be as short as the expedited floors. The engine's effort on a question follows what the question puts at stake, never who paid; a party may raise a matter's stakes for both sides alike. A timely appeal stays execution. If either instance cannot be processed for the law's bounded period, the case closes unprocessable, deposits are returned and no prior judgment is enforced; the matter may be filed again. Determinations are made by the engine's seats, tier by tier, on one identical record; a tie or an answer below the tier's confidence is unresolved, never a winner by order. No human jury or human adjudicator decides any case, civil or criminal. A claimant who cannot establish an element loses the claim; a record the contract could not process is not a loss on the merits. A criminal finding requires near-certainty on every element at the top tier, supported by an adopted evaluation bound to that engine, not merely self-reported model confidence. A seat never judges a case in which its operator is a party.

k8. bounded remedies Remedies are restitution, compensation, agreed performance, correction of records, and — on a criminal finding — a penalty of at most three times the proved harm in total, paid to the victim and the protection pool, and exclusion from the contract's services for at most a year. Each remedy answers a harm established in the case, and no harm is repaired twice. No remedy touches a body, compels labor, or punishes without a finding. An interim freeze holds at most the amount claimed, for at most fourteen days, and only on a determination that the claim is likely to succeed and the amount likely to vanish.

k9. victims are made whole A victim whose final judgment cannot be collected from the respondent is paid from the protection pool, within the limits the law sets and in equal proportion with every other eligible unpaid victim, and the pool takes over the claim. Bond coverage is explicit: uninsured bonds pay no insurance premium and have no pool claim; insured-1 covers only independently verified economic losses between unrelated parties, subject to a cap and available funds. What the pool cannot yet pay stays owed and is paid as the pool is funded. The pool is funded by premiums on bond issuance and by penalties, never by seizing anyone's holdings outside a judgment.

k10. the engine The engine is part of the contract: its tiers, seats, pinned models, renderers and rules are published and change only by RFC. Every tier that holds model seats has seats from at least two independent operators and at least one seat whose weights are open and pinned by hash. Provider availability and independent operation require verification; model retirement can interrupt a pinned matter. Seats of one operator or lineage count as one. Media evidence reaches the seats as text rendered by at least two independent renderers, recorded and open to the parties. A matter is decided by the engine and procedure in force when it began.

k11. records Every determination, RFC, response, case and transfer is recorded with what it relied on. Corrections are appended; nothing is erased. Judgments and RFC records are public; a case's evidence is disclosed only to its parties and only as far as the case needs it.

k12. exit Anyone may leave. Leaving ends future obligations to the contract; duties already accrued and agreements already made survive.

4.2 The Hash

The kernel hash is

$$\text{sha256}\big(\texttt{cc:kernel:v4\textbackslash n} \,\|\, \text{canon}(\{\text{articles},\ \text{floors},\ \text{interests}\})\big)$$

where *articles* is the document above with one paragraph per article, *floors* is the numbers of Table 4.1 in the module's units (milliseconds, per mille), *interests* is the list of protected interests, and canon is JSON with keys sorted and no whitespace. Its value is

```
8c3209a4c2380008f0a362d7b87ba03102ae0ed1a141c00302565955
0b302c2f
```

The reference module pins this value (`KERNEL_HASH`). Its tests fail if the text, the floors or the interests change and the pin does not change with them. The contract computes every state hash, RFC and determination under this kernel.

4.3 Floors Enforced by Code

Code enforces the kernel's numbers where they apply. The law check refuses a law below a floor (Section 5.3). The engine check refuses an engine that lacks independent operators, an open-weight seat, two disjoint top-tier panels, or two independent renderers for every medium the law admits (Section 7.4). The justice functions cap remedies and apply the criminal standard (Section 8). Law may be stricter. It may never be looser.

Table 4.1. The kernel's numbers.

Floor or ceiling	Value
Minimum window of a law RFC	7 days
Minimum window of an engine RFC	21 days
Minimum time to contest one's own extraction	2 days
Minimum time to answer a case	3 days
Minimum time to appeal	3 days
Maximum interim freeze	14 days
Minimum confidence for a criminal finding at the top tier	950/1000
Maximum criminal penalty	3× the proved harm
Maximum exclusion after a criminal finding	365 days
Minimum distinct operators per model tier	2
Minimum open-weight seats per model tier	1
Minimum disjoint quorate panels at the top tier	2
Minimum independent renderers of each media item	2

The protected interests are: life, bodily_integrity, liberty, property, privacy, agreement, voice and access. Every duty of law protects at least one. Every claim alleges harm to one.

4.4 Why These Articles

Article **k1** puts the contract under the same standard as everyone in it. The contract's own acts can be wrongful, and its revenue or survival never outweighs a protected interest.

Article **k2** makes three separations and one guarantee. It separates protection from voice: protection belongs to every being with interests, voice to persons. It separates who people are from what they own or how capable they are. It makes the kind of people extensible without letting one person become many. Every issuer of a kind derives the same key for the same individual, and a voice is given by its holder's own deliberate act.

Articles **k3** and **k4** remove every human or key-holding position from which someone could steer the contract outside a recorded procedure. They keep the people in control of change. Article **k4** also says what a divided engine means: no concurrence. It also says what happens when the outgoing engine cannot answer at all. After a further window as long as the RFC's own, the proposed engine's concurrence suffices, so a failed engine cannot freeze the contract. This article makes a change of the contract's code a change of the engine.

Article **k5** is the market's protection from politics. Articles **k6**–**k9** are the market's subjection to justice:

- the contract serves a respondent before anything runs against them;
- one matter is one case;
- every element is decided;
- a tie is unresolved and never a winner;
- a record the contract could not process is never a merits loss;
- remedies answer established harms without repairing any twice;
- unpaid victims share the pool in equal proportion.

Article **k10** keeps the engine from becoming one provider's product. It fixes each matter to the engine and procedure in force when the matter began. Articles **k11** and **k12** keep the record honest and the institution voluntary. Article k11 makes judgments public and keeps a case's evidence with its parties.

The kernel contains no number that only experience could set well. Windows, deposits, case tiers, penalty multiples below the ceiling, and pool and credit parameters belong to law (Appendix C).

Version 0.4.1 replaced version 0.4's kernel (hash prefix `9af475d6`) before any RFC or case existed under it. By k4 the result is a different contract. The old contract held no record except its genesis. The hosted service therefore re-publishes genesis under the new kernel; it migrates nothing.

4.5 A Defect in the Kernel

The kernel can be wrong. If it is, nobody can amend it. The remedy is a new contract with a new kernel. People may join it and may move their own holdings to it by ordinary transfer. It inherits nothing by declaration. Until peo-

ple leave, the old contract keeps operating under its kernel, and its accrued duties and agreements survive (k12).

5 People and Law

5.1 People Are a Predicate

An entity is a **person** when two conditions hold: its kind is one the law recognizes as people, and it holds a valid personhood proof for that kind. A proof names the entity, its kind, the issuer, a **uniqueness key** under the kind's rule, its validity interval, and whether it was revoked. A proof is valid when the law accepts its issuer for that kind and the proof is current and unrevoked. The person's **voice** is `kind:uniquenessKey`.

Each recognized kind carries a uniqueness rule. Without one, a kind whose members can copy themselves would turn one person into many:

The genesis law recognizes one kind, `human`, under `identity_fingerprint`, with the platform's identity verification as issuer. A kind has one uniqueness namespace and any number of issuers inside it. For humans the contract derives the key itself from the identity an issuer reports. It uses an HMAC under a secret that is never rotated. So a passport and a driving licence of one person, or two issuers of one person, land on one key. An issuer that cannot report those fields cannot issue human personhood, because a second namespace would let a person verified in both speak twice. The tolerance has a cost. Two different people with the same first name, family name and birthday share one voice, and the second can bring a case for their `voice` interest.

An issuer for another kind is named in law by its Ed25519 public key. It attests personhood by signing the entity, the kind, the uniqueness key and the validity window, so nobody can move a proof to another entity, kind or period. The issuer signs a revocation the same way. An issuer's statements about one entity apply in the order of their issue times and never go back. A revocation stays recorded as a tombstone. The contract refuses a replayed older issuance as stale, even though it still carries a valid signature, so the replay cannot undo the revocation. No such issuer exists yet.

Kernel article k2 makes human a kind of people forever. An RFC may change the human rule's issuers but never remove the kind. To recognize another kind is an ordinary law RFC, which must supply that kind's uniqueness rule and accepted issuers. The reference module's `resolveVoices` applies the rule. Of several valid proofs that present the same voice, the earliest-issued proof holds it, and the module refuses the others as duplicates.

The predicate decides voice only. It does not decide whose interests count. Protection belongs to every being with interests, whatever its kind and whether or not the law recognizes it (k1, k2). Anyone may file a claim for a being who cannot (k6). An RFC fixes the kinds of people at its opening: a kind recognized while the RFC is open does not respond to it. An organization is made of people who already have voices. To recognize organizations as people would give their members a second voice through them. Reviewers should judge an RFC that proposes it with that in mind.

5.2 Law

Law has three parts: the recognized kinds of people, the duties, and the parameters. It carries a version number that increments with every adopted change. The contract keeps every past law with the instant it took effect, so that an act is judged under the law in force when it happened (k6).

5.2.1 Duties

A duty has an identifier, a title, its text, the protected interests it protects, the harm it forbids, the **elements** that make a breach, whether its breach is a criminal matter, and the instant from which it applies. The text binds everyone alike. A duty has no field that names whom it applies to, so it cannot single out particular people or holders (k5). The elements are what a case decides one by one (Section 8.8). Fraud, for example, has 5 — a representation of fact, its falsity, knowledge or recklessness, reliance, and the harm the reliance caused. So a liability finding cannot skip the element that distinguishes fraud from an honest mistake. The genesis law contains six duties, listed with their elements in Appendix C: no fraud, no theft, no violence, no coercion (criminal), keep agreements, and no exposure of private information (civil).

5.2.2 Parameters

Parameters are integers: windows and response times in milliseconds, amounts in cents, rates in basis points or per mille. The genesis values are in Table C.1. The kernel bounds the ones it constrains (Table 4.1). Besides the parameters, law lists the **media admitted as evidence** and as RFC responses — text, image, audio in the genesis law. Text is always admitted. At least two independent renderers of the engine must be able to render a medium that the law admits (k10). The contract refuses a medium that the law does not admit at submission and gives a reason. It never drops it silently.

5.2.3 Changes

A law change is one of: add a duty, amend a duty, repeal a duty, set a parameter, recognize a kind of people, unrecognize a kind, set the admitted media. An RFC option carries a list of changes, applied in order. A repealed duty keeps governing acts done while it was in force.

5.3 The Law Check

Before an RFC opens, and again when it is adopted, the contract applies each option's changes to the current law and checks the result against the kernel:

- human is still recognized;
- there are no duplicate kinds or duties;
- every duty protects at least one interest, names its harm and lists at least one element;

Table 5.1. Uniqueness rules in the reference module.

Rule	One voice per	Natural kinds
`identity_fingerprint`	Document identity: first given name, family name and birth date, normalized and keyed by the contract's own secret	Humans
`lineage`	Origin lineage: copies, forks and fine-tunes share a key	Artificial agents; uploaded humans
`registration`	Legal registration	Organizations
`custodial_individual`	Attested biological individual, spoken for by a representative	Animals

- text is admitted;
- every parameter is within its kernel bound;
- the engine in force can render every admitted medium with two independent renderers.

The check refuses a change that fails, with the code of the article it violates (for example `k4.law_window` or `k2.human_removed`). This is the structural half of kernel article k5. The semantic half asks whether a duty that is harm-based in form is in fact a price control or a measure aimed at someone. That is a ground on which the engine withholds concurrence (Section 6.7).

5.4 What Law May Not Do

> **The Market Is Free of Politics (k5).** Law may not set prices, tax or single out particular holders, freeze or seize holdings, or restrict an exchange between consenting parties — except to prohibit harm to a protected interest of a being who did not consent. Every duty names the protected interest it protects and the harm it forbids, and binds everyone alike.

The line is consent and harm. A sale of a dangerous good to a consenting buyer is the buyer's business. The same sale is not the buyer's business when it injures a bystander. A cartel's agreement among its members is their business. Its fraud on a customer is not. A protocol fee set by law is not a tax on particular holders. It applies to every settled transfer and funds the commons that the contract itself needs. The commons may run programs adopted by RFC — grants, subsidies, insurance — with its own money. It may never compel anyone else's.

5.5 Responsibility for Agent-Mediated Acts

A principal answers for acts it authorized, adopted or was specifically obliged to supervise. A case must show the scope of the agent's capability, the actual instruction or adoption, and the applicable duty. An agent of a recognized kind may bear its own accepted duties. The model operator, tool provider, employer and principal are not interchangeable respondents. Knowledge is attributed through what the responsible decision process actually received. It is not attributed through a model's training exposure or an unused tool permission.

5.6 Entry and Exit

Anyone may hold an account and accept agreements. A person enrolls by presenting a personhood proof. Leaving ends future obligations to the contract. Accrued duties, pending cases and accepted agreements survive (k12). Leaving does not withdraw a response already counted in a closed RFC.

The name-and-birthday fingerprint is an imperfect proxy, not proof of uniqueness: collisions and name changes exist. Re-verification cannot overwrite an established voice. Two things together authorize a resolution between open RFC windows: a final correction judgment and a domain-separated statement from a law-admitted human issuer. The contract records the resolution in the private case, and nobody can replay it. No such issuer is configured at genesis. Continuous RFC windows may delay correction. Cross-account identity continuity still needs external evidence.

6 RFCs: Interpreted Voice

An RFC is how law and engine change. It replaces three things the predecessor used: ballots with quorum and supermajority thresholds, stakeholder-class ratification, and branch signatures. People respond in their own words, images or speech. The engine interprets each response against the actual effect of each option. Every voice scores every option, and the contract sums the scores. The engine concurs within strict limits, and only with confidence.

6.1 Opening an RFC

Anyone — person, organization or agent — may open an RFC. A draft has a title, a body and one or more change options. All options change the same thing: law or engine. The contract adds the status quo as option zero. At opening:

- The contract applies every law option to the current law and checks it (Section 5.3). It checks every engine option against the kernel and the media the law admits (Section 7.4), and it refuses an engine option equal to the engine in force. If one option fails, the contract refuses the draft.
- The contract refuses two options with the same effect (rfc.duplicate_effect), whatever their titles and summaries. For this purpose, an option's identity is what it changes, not how it is described.

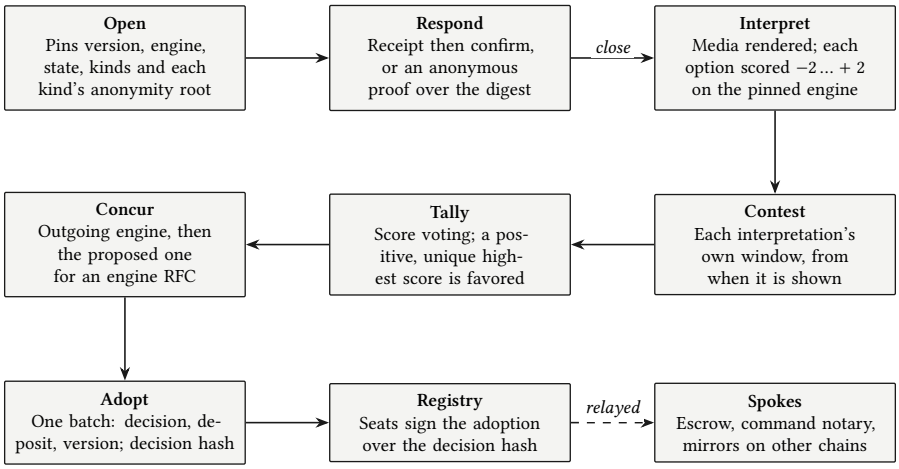


Figure 6.1. The life of an RFC. Every step is a determination or a recorded transition on the transparency log (Section 16.2). No favored option, or a divided, withheld or unavailable concurrence, keeps the status quo.

- The contract refuses a draft whose options have the same effects as those of an RFC rejected within the cooldown (90 days in the genesis law).
- The window is the law's window for the target, never below the kernel floor. In the genesis law it is 14 days for law and 28 days for the engine (floors 7 and 21).
- The opener posts a deposit of $10. The contract debits it in the same transaction that records the RFC. The deposit returns to the opener if anyone other than the opener supported a change. Otherwise it goes to the commons, which pays the interpretation costs.

Pinning (k10). An RFC records what it began under: the contract version, the hash of the engine that will interpret and concur, the hash of the contract state against which its options were drafted, and the kinds of people the law recognized at opening. That engine interprets its responses, and those kinds resolve its voices, whatever is adopted while the RFC runs. The contract's state can change before the RFC is decided, for example when another RFC is adopted in between. Then the RFC is *superseded* (`adopt.superseded`) and is not adopted, because its voice was given about a contract that no longer exists. The RFC's identity is the hash of its content. A later material change makes a new RFC.

6.2 Responses and Receipts

A response is a person's own act: their voice, the entity presenting it, its parts and its time. A part is text, or a reference to media bytes (an image or audio, as the law admits). The contract stores media bytes once, under their SHA-256. Assistance in drafting and translation is ordinary. An agent may not originate a person's response, and an agent key speaks only for the agent.

Receiving a response is not counting it. The contract answers a submission with a **receipt**: the response's identifier and the digest of its exact content, sha256(`cc:rfc-response:v4` …) over the RFC and the parts (for example `6783adef14124f35`…). The response counts only after its author confirms that digest. This confirmation is the deliberate act that kernel article k2 asks for, and it is bound to the payload, not to a session. A person may confirm a new response any number of times before close; the latest confirmed response counts. These responses do not count: an unconfirmed response, a response confirmed after close, and a response from a kind not recognized at opening. The reference function `activeResponses` returns exactly one confirmed response per voice.

6.3 Anonymous Voice

Kernel article k2 asks that each person have one voice, given by their own deliberate act. It never asks who. Thus a person may answer an RFC so that nobody, the platform included, knows which response is theirs, and still count exactly once. The reference implementation uses Semaphore v4: Groth16 proofs over the BN254 curve, a lean incremental Merkle tree of Poseidon hashes, and EdDSA-Poseidon identities [10].

- **Identity.** The person makes a Semaphore identity in the contract editor. Its secret lives only on their device and in a file they keep. It is never uploaded.
- **Registration.** When signed in, the person registers the identity's *commitment* for their voice. A voice has one active commitment, and only one voice holds a given commitment. This is the one place where a voice and a commitment meet, and the platform keeps that link. The proofs below are zero-knowledge over the whole set, so the link says nothing about which response is whose. Rotating the identity replaces the commitment and keeps the history.
- **The tree.** Each kind's active commitments, in registration order, are the leaves of a public tree. **An RFC pins each kind's root and size when it opens**, and the pin is part of the RFC's hashed record. An RFC opened over another root is another RFC (yes in Appendix D.13).
- **Acts.** An anonymous act is a request that carries no credential. It carries only a proof that some member of the pinned tree signed exactly this message in this RFC's scope. The proof's **nullifier** is the hash of the identity's secret and the scope. It is the same for every proof of one

identity in one RFC, and it is unlinkable to anything else. The nullifier *is* the voice: `anon:human:77`. Each action has its own message: respond (binding the response's digest and a fresh nonce), upload media (binding its SHA-256), contest, and look up. Thus a captured proof replays only as itself (5 distinct messages for the five actions of Appendix D.13). The proof is the deliberate act, so an anonymous response counts on receipt. A later proof with the same nullifier replaces the earlier response.

6.3.1 *One Person, One Count*

Without a rule, a person could count twice: once signed in and once anonymously, or through a rotated identity. The rule closes both paths. A person whose commitment is in the tree that an RFC pinned answers that RFC *only* anonymously; the signed-in route refuses them. A person without such a commitment answers with their signed-in voice. This person either never registered or registered after the RFC opened. A commitment registered or rotated after opening is not in the pinned tree, so rotation can never open a second voice. In Appendix D.13, Ann registered before opening and rotated after, and she answers only anonymously (yes). Cal registered after opening, and he answers signed in (no for the anonymous-only rule). The cost is real. A person who loses their identity file cannot answer RFCs opened while it was registered. That person registers a new identity for the RFCs that follow.

6.3.2 *Verification*

A proof acts in an RFC only when all of these are true. It verifies. Its root is the root that the RFC pinned for the kind it claims (`not_the_pinned_tree` otherwise). Its scope is this RFC (`wrong_scope`). Its message is the one the action requires (`wrong_message`). The kind was recognized at opening (`kind_not_recognized_at_open`). An anonymous response may cite only media uploaded under a proof, never a URL that the contract would fetch (`anonymous_media_must_be_uploaded`). The hosted service verifies proofs inside its worker runtime with its own Groth16 check over the BN254 pairing. It checks against Semaphore's verification keys for every tree depth. These keys are copied byte for byte from Semaphore's own package, and a test holds them to it. On real proofs, forged proofs, proofs with each public input changed, and garbage, its verdicts equal snarkjs's.

6.3.3 *What Can Still Link a Response to a Person*

A proof hides a person among everyone in the pinned set. A set of one or two hides nothing, and the editor shows the set's size. The anonymous routes read no cookie, authorization or principal. They record nothing that links a nullifier to an entity. The editor sends to them without credentials and without a referrer. But some links remain. The operator can still see network metadata (the address a request comes from) and timing. The content of a response can identify its author. When the editor fetches the proving artifacts from Semaphore's artifact host, that host sees the person's address. The editor can hold a proved response for a random wait before it sends it. A relay or an anonymity network is the person's choice. Anonymous voices have no inbox, so they receive no notices. Their authors look up their interpretations under a proof. The design claims unlinkability by the contract's record, not anonymity against an observer of the network.

6.4 Interpretation

After close, the engine renders media in the responses to text (Section 9.3). Then a determination of kind `rfc.stance` on the RFC's pinned engine interprets each counted response once per change option. The question carries the RFC, the option's *actual changes* and the response. The actual changes are the law changes or the engine, in canonical form, with the opener's summary marked as not binding. The question asks for the person's stance on adopting that option instead of the status quo. The engine judges the stance only from what the response expresses, never from who the person is. The labels are fixed, and each maps to a score:

Label	Score
strongly_oppose	−2
oppose	−1
neutral	0
support	+1
strongly_support	+2
not_addressed	0

The question has closed labels, so every tier can answer it: an MLP head over the response's embedding, a classifier seat, or a deliberative model. A response that addresses only one option of several is `not_addressed` on the others. An interpretation that the engine leaves *unresolved* scores 0, and the contract records it as unresolved. An interpretation that the engine cannot process within the law's period (14 days in the genesis law) scores 0, and the contract records it as unprocessable. The contract counts neither as the person's opposition or support.

Contest. The contract shows each interpretation to its author when it records it. The contest window starts at that moment, not at the RFC's close. It is 3 days in the genesis law, with a kernel floor of 2. Thus an interpreter that runs late or slowly never shortens anyone's chance to inspect and contest their own interpretation. For example, an interpretation recorded 9 days after close is contestable until day 12. A person contests their own interpretation once. The contest reruns every option's question at the top tier, on a panel that excludes every group that already sat. The result replaces the original. If the contest cannot be heard within the law's period, the original stands. A person cannot contest someone else's interpretation. A factual error in someone else's response is a matter for the voice. The RFC is decided only after these three conditions: every counted response is interpreted, every window has passed, and every contest is resolved.

6.5 Declared Procedure: Deliberative by Default

The procedure is fixed at opening. The `deliberative` procedure uses a recorded `rfc.deliberate` determination to choose an option, including the status quo. The determination considers substantive merits, the strongest objections, participation and how representative that participation is. A single comment represents its author, not a population. Large self-selected participation is not random sampling, and silence is not consent. The decision reports unresolved comments and does not assume that they agree. The choice can differ from the favored score, or it can be justified with no comments. The decision records the deliberation id, the respondent count and the limitations. Seat reasons explain the consideration. The score-voting examples that follow apply to the explicitly declared `binding_score` procedure.

6.6 The Tally: Descriptive or Binding

For each change option k, the score is

$$S_k = \sum_{\text{persons } p} s_{p,k}, \qquad s_{p,k} \in \{-2, -1, 0, 1, 2\},$$

the sum of every voice's score for that option. The **favored** option is the one with the highest positive score, if that option is unique. A tie for the highest positive score, or no positive score, leaves the status quo. Silence counts as nothing: a person who does not respond neither supports nor opposes. There is no quorum. Each voice contributes at most two points to any option. Wealth, stake and activity add nothing.

Why score voting. Version 0.4 normalized each person's stances so that their absolute values summed to at most one unit. That made each option's weight depend on the *set* of options. A person who opposed many alternatives spread their opposition thin. A person who supported one alternative kept their whole unit there. Thus adding redundant alternatives could change the contest. Under score voting, a voice's score for one option is independent of its scores for the others. So the score of an option does not change when another option is added, duplicated, reworded or reordered. The reference tests check this directly, and Appendix D.1 shows it: adding a third option leaves the first two options' scores at 3 and -1. The contract refuses options with identical effects at opening, so a proposer cannot split support for one change across clones. The tally record holds every voice's scores per option and which interpretations were unresolved or unprocessable. The contract hashes this record. The concurrence reads that digest, and anyone can recompute it.

Why No Quorum. Quorum thresholds answer the worry that a small group decides for everyone. But they also let a disengaged majority block every change by staying away. In practice this freezes the institution and moves real power to whatever needs no vote. The contract answers the worry differently: windows of at least seven and twenty-one days with notice to every person, a kernel that no RFC can touch, the engine's confident concurrence, and the fact that any adopted change can be reversed by the next RFC.

6.7 Concurrence

Before a selected option is adopted, the pinned engine answers a determination of kind `rfc.concur`. It starts at the top tier and has exactly two labels: `concur` and `withhold`. Its package holds the kernel, the RFC, the selected option's actual changes, the whole tally record, the contract as it was at opening, and a **seeded sample** of 24 responses (in the genesis law) with their interpretations. The sample is drawn by a public, deterministic order keyed by the RFC and the tally digest, not as the first responses in some list. So anyone who holds the record draws the same sample. Reproducibility does not prove immunity to strategically chosen inputs. For binding-score RFCs, the prompt allows withholding on three grounds. Deliberative RFCs replace the voice-mismatch ground with unfair consideration, ignored material objections or false claims of representativeness:

- Adopting the option would violate the kernel in a way the structural check cannot see;
- The interpreted responses do not in fact favor the option — for example, a block of interpretations that read support into responses that express none;
- The option does something law may not do (Section 5.4): a price control dressed as a duty, a measure aimed at named holders.

Disagreement with the people's choice is not a ground. When the engine withholds concurrence, a second determination over the same package classifies the ground (`kernel_violation`, `voice_mismatch` or `outside_law`) for the record. The decision does not depend on this classification. The engine decides assent separately from the classification of objections. Thus three different objections can never split against one assent and leave assent as the largest label.

A binding-score RFC is adopted when an option is favored and the concurrence is *decided* with the label `concur`, that is, a unique answer at the top tier's band. A tied or low-confidence panel is unresolved and does not concur. An unavailable engine does not concur. **Under binding score the engine cannot adopt an unfavored option. Under deliberation the choice is reasoned consideration, not a binding tally. Both require confident concurrence.** (Appendix D.1: with an unresolved concurrence the result is null.)

6.8 Engine RFCs

Two concurrence determinations over the same package decide an RFC whose option replaces the engine. They run in order. The contract asks the outgoing (pinned) engine first. Only if it concurs does the contract ask the proposed engine. It asks the proposed engine through a **candidate executor**. This executor reads only the public RFC record and renders no media. Its seats' credentials are bound to their operators' own hosts, like every seat's (Section 7.10). Thus the evaluation of a possible successor never grants it the successor's powers. Both engines must concur.

This order prevents an incumbent engine from entrenching itself. The incumbent cannot adopt alone, and it can stop a replacement that the voice favors only on the three grounds. The order also prevents a captured replacement from installing itself, because it cannot adopt alone either. An outgoing engine that has *failed* must not freeze the contract. So kernel article k4 adds one fallback. If the outgoing engine cannot seat a panel at all for a further window as long as the RFC's own window, the proposed engine's concurrence is sufficient. The fallback applies only to unavailability. An outgoing engine that answers `withhold`, or is divided, has answered. Appendix D.2 shows each path. On chain, an engine change and a change of the contract's code both need the signatures of the outgoing and the proposed engine (Section 15.4).

6.9 Adoption

The contract applies an adopted option in one transaction. The same transaction records the decision, settles the deposit and appends the new contract version. The transaction is conditional on the RFC's revision. It is also conditional on the contract's version being the one that the decision read. Thus two concurrent sweeps cannot both adopt, and an adoption cannot land on a contract that moved. Each adoption increments the contract's version and appends to the law or engine history. The contract hashes the decision record (`decisionHash`, prefix `ed2b9b62cc147043` in Appendix D.1). An on-chain adoption binds that hash.

6.10 Worked Example

Appendix D.1 follows a seven-person RFC with two rival options through score voting, the clone test, a late interpretation's contest window, concurrence and adoption. Appendix D.2 shows the order of the two engines and the fallback.

6.11 Privacy and Assurance

The record of an RFC holds each counted response, its receipt, its interpretation and its determinations. RFC records are public (k11). The design does not claim coercion resistance, because a person can show anyone their own response. It claims two things. Coercion of a response is a criminal duty in the genesis law (`coercion`). The receipt lets a person prove exactly what they confirmed [12, 13].

7 The Engine

The engine is the pinned set of models through which the contract decides. It is data, and the contract hashes it into the contract state. A change to a seat, a pin, a head, a renderer, a band or the record of who controls an operator is an engine RFC (Section 6.8). The engine has four tiers, cheapest first. A question climbs until a tier answers decisively.

7.1 Tiers

Tier 0 answers only when the question carries a structural answer computed from records. Tier 1 answers only when the engine has a head trained for exactly that question kind and label set. A tier's answer counts when exactly one label wins and its confidence reaches the tier's band (950, 850 and 700 per mille in the genesis engine). Below the top tier, a tie, a missing quorum or a confidence below the band makes the question climb. At the top tier, a tie or a low confidence is **unresolved**, and a missing quorum is **unavailable**. Version 0.4 made the top tier's answer final whatever its confidence, and it broke ties by the order of the labels. A review showed the result: when four groups answered four different labels, the first label, `concur`, became the answer at 250 per mille. That rule is gone. A label wins only by weight.

7.2 Questions Are Closed, and Their Records Are Text

Every question the contract asks has a fixed set of labels: stance labels for RFC interpretation; `concur/withhold` and, for the record, the three grounds; `same_matter/different_matter`; `justiciable/not_justiciable`; `established/not_established` for each element; `claimant/respondent` for each harm's figure; `justified/not_justified` for each remedy; `freeze/no_freeze`; `holds/does_not_hold`. A seat that answers outside the set has made an error, and the contract records it as an error. Closed questions make every tier interchangeable and make aggregation exact. They also leave no free text from which a model could award itself power. Reasons may be recorded alongside an answer, but they never change it.

The inputs of every question are one **canonical package**: the prompt, the labels and a list of text items. Each item has a unique identifier (`claimant/2/1`, `response/r/3`, `option`). Adapters key on identifiers, not on display labels. Version 0.4's Typesafe adapter keyed its input by label. So two submissions that were both labelled *claimant* overwrote each other, and that seat read less than the others. Media never reaches a deciding seat directly; the engine first renders it to text (Section 9.3). Every seat's answer carries the digest of the package that its adapter actually sent. The contract records an answer over a different package (for example, an item dropped or a modality skipped) as an error, not as a vote. Thus one identical record (k7) is a checked property of every determination, not an intention.

7.3 Seats and Pins

A seat names a wire protocol, the exact model id sent on the wire, an endpoint where the protocol needs one, a pin, the operator that runs it, the training lineage, the entities recorded as controlling the operator (`controlledBy`), the modalities it reads, and its cost per call. The pin is either the SHA-256 of open weights with their source, or the exact version id of a hosted model.

A hosted model can change behind its ID or be retired. Open weights pinned by hash cannot. The pin fixes weights, not a provider's serving stack. A host may serve a quantization of pinned weights (the genesis engine's 70B Llama seat is served in FP8), and the engine records that residual with the seat. So kernel article k10 requires every model tier to

Table 7.1. The four tiers.

Tier	Name	Who answers	Reproducibility
0	Structural	A deterministic check over committed records (a signature, a deadline, an escrow balance)	Anyone, exactly
1	Embedding	One open-weight embedding seat and pinned INT8 MLP heads	Anyone, bit for bit given the embedding
2	Classifier	A panel of fast classifier seats returning a label and reported confidence	Recorded answers
3	Deliberative	A panel of large models; final, or unresolved	Recorded answers

hold at least 1 open-weight seat and seats from at least 2 operators. If every hosted seat of a tier disappears, the tier still has an open-weight seat. If that leaves the tier without a quorum, its questions climb. The top tier's questions then become unavailable until service returns; the contract records this and never guesses. An engine RFC changes future matters only, and a prolonged case outage closes the case as unprocessable. No provider can change the contract by changing a model. No provider can stop it by retiring a model without a visible outcome.

Renderers are seats of their own. Each lists the media it renders (image, audio). Renderers decide nothing; they turn one media item into text (Section 9.3). The genesis engine's image renderers are render-gpt-5.6-luna, render-claude-haiku-4-5; its audio renderers are render-whisper-large-v3-turbo, render-gpt-4o-mini-transcribe.

7.4 The Engine Check

An engine is admissible when:

- The embedding seat has open weights, and head, seat and renderer identifiers are unique;
- Each model tier has the required operators and open-weight seats, every deciding seat reads text, and each tier's quorum does not exceed its number of independent groups;
- **Appeal capacity** (k7): the top tier has at least 2 times its quorum in independent groups. Thus it holds 2 disjoint panels, and a matter decided at the top can still be appealed to groups that took no part. Version 0.4's genesis engine had four deliberative seats and a quorum of three. A criminal case decided at the top tier left no fresh panel, and its appeal could never be heard;
- **Rendering** (k10): every medium the law admits has at least 2 independent renderers. Version 0.4 asked deciding seats to read media themselves. One image in a case left both model tiers without a quorum, so adding evidence destroyed the ability to decide on the rest.

The genesis engine has 3 independent classifier groups (quorum 2). It has 6 independent deliberative groups (quorum 3, so 2 disjoint panels). Appendix C lists its seats and pins. The check refuses an engine with only hosted deliberative seats with k10.open_weights, k7.appeal_capacity. It refuses an engine with five deliberative groups with k7.appeal_capacity. Under a law that admits audio, it refuses an engine with a single audio renderer with k10.renderers.

7.5 Independence and Aggregation

Seats that share an operator or a lineage form one **group** (the union of both relations). Within a group, the vote is the label with the largest summed confidence, weighted by that label's mean confidence. A group split evenly between labels casts no vote. Across groups, the label with the largest total weight wins, if it is unique. The tier's confidence is

$$c = \frac{\text{weight of the winning label}}{\text{number of answering groups}} \quad \text{per mille},$$

so both disagreement and uncertainty lower it. A unanimous tier of seats that are each 600 confident reaches 600. A 2–1 split of fully confident groups reaches 667. When two or more labels share the largest weight, there is no winner. The tier's aggregate records a tie, and its confidence is zero. Three seats of one operator cannot outvote two independent seats. In Appendix D.3, five seats in three groups decide no at 600. When four groups each answer a different label of four, there is no winner (Appendix D.5).

7.6 Who Sits: Panels

A **panel** of independent groups answers a tier, with one seat per group. The panel has at least the tier's quorum, plus as many more groups as the routing plan finds worth their cost (Section 7.7). A group is excluded in three cases. First, any of its seats already sat in the matter. Second, a party runs or controls any of its seats, by operator or by an entity in `controlledBy`. The mapping is part of the engine, so renaming an operator cannot create independence. Third, none of its seats is available. A public, deterministic key, derived from the matter and the instant of the draw, orders the remaining groups. The first of them sit, as many as the plan says. The plan's soft preference draws which seat answers for each group. Anyone who holds the record redraws the same panel. Nobody chooses one.

Where it matters, the panel is fixed per matter. A case draws its panels once, when an instance starts (Section 8.8). The same seats answer every question of that instance. An appeal draws again from groups that took no part. Version 0.4 drew every eligible seat. The contract now draws exactly a quorum, and this leaves the rest of the top tier free for the appeal. Appendix D.7 draws the first panel and the appeal panel of a case on the genesis engine. A third disjoint panel does not exist (no). This is why there is one appeal.

7.7 Routing by Expected Loss

The expected loss of a question, in micro-dollars, chooses how much of the engine it uses: where it starts, how far it may climb, how many groups sit and which seat answers for each. A seat that answers for its group at a tier costs

$$L(\text{seat}) = S \cdot e + c + \kappa \cdot t + q \cdot L_{\text{next}},$$

where S is what the answer puts at stake, e the seat's error rate, c its mean cost, t its mean latency, q how often it gives no decisive answer, κ the value that a wait blocks per second, and L_{next} what climbing on costs. The stakes come from the record: a case's harm items and the penalty it asks, an agreement's escrow, a freeze's amount, or an RFC response's nominal $100. They never come from who paid. A case on a facility's agreement also counts the facility's remaining principal, because a judgment that puts the bonds in default moves every holder's value, not only the claimant's share. A party may only raise a matter's stakes floor, and only for both sides at once (Section 8.5). The seats' figures are part of the engine (Appendix C), and only an engine RFC changes them.

The plan. The engine prices every path from a starting tier to a highest tier, from the top down. A panel of m groups with mean error $\bar{e}$ errs as $\rho\bar{e} + (1 - \rho)\,\text{maj}(\bar{e}, m)$. Here ρ is the share of errors that the tier's groups are assumed to share (600 and 500 per mille at genesis). The term maj is the chance that a strict majority errs independently. An even split is a tie, which climbs. A tier waits for its slowest group. An unsure answer at the path's top is unresolved, and the plan prices it at half the stakes. The engine takes the cheapest feasible path, with as many groups as pay for themselves. Latency counts twice. Its mean counts at κ (on the transaction path, $0.01 a second at genesis). Its p99 counts as a **deadline** (2 s for an agreement's proposition). The engine does not take a path whose p99 does not fit. A seat that cannot answer in time does not sit. When no path fits, the determination is unavailable, not guessed. The procedure's floors stay floors. Criminal matters, appeals and contests start at the top, and a first instance leaves a disjoint top-tier panel free.

A soft preference within a group. The engine draws the seat that answers for a group with probability proportional to $\exp(-L/\tau)$ (τ = $0.002). At low stakes, the losses are close and the draw spreads. So every seat continues to be measured, and no one model hears every matter. At high stakes, the stakes term dominates and the draw is near certain. The choice of which groups sit is never weighted, so a preference cannot outvote independence. The arithmetic uses only $+, -, \times, \div$ (the exponential is a fixed Taylor series after range reduction) and integer draws from the seed. Thus a replica recomputes every plan and every pick from the recorded inputs and draw.

A seat's confidence on the tier's scale. A seat can carry an adopted calibration table, like a tier-1 head's. The table is a list of ascending bands of the seat's own reported confidence. Each band maps to the confidence that the tier's scale counts. A group weights its seat's label by the calibrated figure. The table moves a report and never chooses the label. A report below the first band stands as it is. The engine refuses a table that maps a higher report below a lower one. Every determination recomputes with the table, so a replica reaches the same aggregate. The genesis engine has no table. The review sample supplies the labelled answers from which an engine RFC would adopt one; the proposal takes the Wilson 95% lower bound of accuracy at each report, as `train-head.mjs` does for a head's margin.

Table 7.2 shows the genesis engine's plans for two matters: an agreement's proposition on the transaction path, and a case on ordinary windows. The percentage is the drawn seat's share of its group's preference.

Fast now, the top tier later. On the transaction path, an identical question returns the recorded determination. Identical means the same package, engine, routing inputs and parties. A seeded 5% of fast determinations, drawn from each determination's own identifier, goes again later to the top tier, to groups that took no part. The review changes nothing by itself. The contract records a disagreement and gives notice of it to the agreement's parties. They contest in the ordinary way, by a case on the agreement. Each review is a labelled example. The contract publishes the seats' measured figures beside their adopted figures, so that the next engine RFC can adopt from them (Section 19).

7.8 MLP Heads

Tier 1 makes the cheap path reproducible. A head is a small multilayer perceptron stored in the engine. The engine stores its question kind and labels, the embedding seat it was trained on, the embedding dimension, an input scale, its layers as INT8 weights with integer biases and a right shift, a calibration table, the evaluation that justified adopting it, and its own hash. Inference is integer-only:

1. Quantize the embedding: $q_i = \text{clamp}(\text{round}(x_i \cdot \text{scale}), -127, 127)$;
2. For each layer, accumulate $b_o + \sum_i w_{oi} q_i$ exactly, divide by 2^{shift} with floor, and apply ReLU clamped to $[0, 127]$ on hidden layers;
3. Take the label of the largest output (ties to the first), the margin to the second largest, and the confidence of the largest calibration threshold at or below the margin.

No exponential, softmax or floating point appears after quantization. Quantization uses only correctly rounded IEEE operations. So every machine produces the same answer from the same embedding. A tie between outputs has margin zero. The calibration gives a zero margin no confidence that a band would accept, so a head's tie climbs. The determination records the hash of the quantized input, so a replay can check that it started from the same vector. The embedding itself remains trusted. The embedding seat is open-weight and pinned, but floating-point inference can differ slightly across hardware. In most cases quantization absorbs this difference; in the rest, the recorded input hash shows it. Appendix D.6 shows a head's arithmetic.

An engine RFC adopts heads with their evaluation. `scripts/contract/train-head.mjs` does these steps: it trains

Table 7.2. Genesis plans by stakes (seed `paper`): the tiers from start to top, the groups sitting at each with the drawn seat's share of its group's preference, and the expected loss.

Path	Stakes	Tiers	Classifier panel (groups: seats)	Top panel	Expected loss
hot	$0.10	2–2	`2: clef (100%), jev-1.13.0`	—	$0.0244
hot	$10	2–2	`2: clef (100%), jev-1.13.0`	—	$2.01
hot	$1,000	2–2	`2: clef (100%), jev-1.13.0`	—	$201
hot	$100,000	2–2	`2: clef (100%), jev-1.13.0`	—	$20,063
case	$0.10	2–3	`3: gpt-5.6-luna, clef (100%), jev-1.13.0`	`3: deepseek-v4-pro, claude-opus-5-5, qwen3.8-max`	$0.0144
case	$10	2–3	`3: gpt-5.6-luna, clef (100%), jev-1.13.0`	`3: deepseek-v4-pro, claude-opus-5-5, qwen3.8-max`	$0.9438
case	$1,000	2–3	`3: gpt-5.6-luna, clef (100%), jev-1.13.0`	`3: deepseek-v4-pro, claude-opus-5-5, qwen3.8-max`	$93.89
case	$100,000	2–3	`3: gpt-5.6-luna, clef (100%), jev-1.13.0`	`3: deepseek-v4-pro, claude-opus-5-5, qwen3.8-max`	$9,388

a head from labelled examples and quantizes it. It sets the head's calibration table from the Wilson 95% lower bound of held-out accuracy at each margin. It makes this table monotone, so a larger margin never means less confidence. It checks the integer head on the held-out set and seals it. The genesis engine ships with no head. No head has been trained on a contract question, and a head without a published held-out evaluation should not decide anything.

7.9 Classifier Seats

Tier 2 suits models built for typed decisions. Typesafe's Jev is an example. It returns one value from a declared output set with a calibrated probability, in tenths of a second, and it does not generate free text [41]. That matches the closed questions exactly. Cloudflare's Clef and Clef-flash have the same interface and open weights (Qwen3.8-27B and Qwen3.5-9B bases). They read the package in one prefill pass and score only the allowed labels [42]. The genesis engine seats both with the open-weight Llama seat in Cloudflare's group. Routing draws which seat answers (Section 7.7). One operator remains one group, however many seats it runs. An OpenAI-compatible model with a constrained output and an open-weight model served by a different operator can sit in the same tier. Two cautions follow from the kernel's grouping rule. First, a classifier trained to imitate a large model's answers shares that model's lineage for the purpose of independence. So it should be recorded with that lineage. Second, a vendor's calibration claim is evidence to test (Section 19), not a band to adopt.

7.10 The Executor

The reference module's `runDetermination` is pure orchestration. It takes the engine, the question and an executor, and it returns the record. The executor has three operations: embed text with a seat, ask a seat a question, and render a media item with a renderer. The module records seat failures, out-of-set answers and answers over a different package as errors. It asks the seats of a panel in parallel. The record's identity is the hash of its content, without the timings of the calls. The record keeps those timings beside each answer as measurements.

The hosted service's executor implements each wire protocol as an adapter. The adapters are: OpenAI-compatible, with a strict JSON schema whose label is an enum; Anthropic, with a forced tool call; Typesafe, with a choice question keyed by item identifiers, and the same System One wire through the Workers AI binding for Clef; and Workers AI in JSON mode, for embeddings and speech. The executor calls each seat exactly as pinned, never through a model registry or failover. **Credentials are bound to hosts.** Each operator's credential is listed with the hosts to which it may be sent, independently of any engine. A seat is unavailable if its endpoint is not HTTPS on one of its operator's hosts, or if the endpoint carries a port, a user name or a password. Anyone can write a seat in a *proposed* engine, because it is data. With this binding, a seat endpoint that names a collection server can never draw a credential to that server. The **candidate executor** for a proposed engine's concurrence (Section 6.8) also renders nothing, so it reads no private evidence. The executor fetches media once. Before any renderer sees the bytes, it checks them against the SHA-256 to which the submitter committed. Chat models report their own confidence, and the executor records it as reported. Anyone can supply another executor and compare records. Others cannot re-run hosted seats, because only their operators can call them as pinned.

7.11 A Climb, and an Unresolved Top Tier

In Appendix D.4, the classifier panel answers gpt-5.6-luna error; clef does_not_hold 900: does_not_hold at 900. This is below its band, so the question climbs. The deliberative panel decides holds at 850. In Appendix D.5, the same question reaches a top-tier panel that splits two to one at 600, below the band. The outcome is unresolved, not a label.

8 Justice

Automated litigation makes law enforceable at the scale of autonomous commerce. The contract decides every case through determinations: private agreements, law duties and kernel duties, civil and criminal. There is no human jury, no human adjudicator and no human fallback (k7).

People and agents remain witnesses, parties, representatives and sources of evidence.

A determination is not yet a justified judgment. A closed label and a confidence say less than a court's reasons. Version 0.4.1 narrows that gap by structure, not by trust in any one answer. A liability finding is the conjunction of separately decided elements. The amount is a choice between the parties' own figures for each harm. Every remedy is separately justified. Every question reads the same record: one identical, complete text package.

8.1 The Scope of Justice

A claim is justiciable when it names:

1. A **claimant** who was harmed, or anyone filing for a being who cannot;
2. A **respondent** who was served (Section 8.4);
3. An **act** or omission, with its time;
4. A **duty** the respondent owed: a term of an agreement the respondent accepted, a duty of the law in force at the time of the act, or a kernel article;
5. **Harm** to a protected interest, or to an agreed term, caused by the act — as one or more **harm items**, each with its interest, description and amount.

This test is objective in the sense that matters. Each element is a fact to be shown. No element asks whether the respondent is popular, rich, foreign or disliked. The test also protects the market from misuse of the courts. Without harm to a protected interest of a being who did not consent, there is no case. Anyone may file, also for a being who cannot. The contract files from its own records, with no prosecutor in between: for example, a facility's default (Section 10.4) or a double spend.

8.2 Filing

The contract checks a filing before anyone is served. It refuses the filing, with codes that name the kernel article, when: the respondent is the harmed party; the act is in the future; a law duty was not in force at the time of the act; the duty does not protect an item's interest; a criminal claim rests on a non-criminal duty, an agreement alone or a kernel article alone; the respondent did not accept the agreement; evidence is in a medium the law does not admit; a remedy names a harm item that does not exist; a penalty or exclusion is sought in a civil case; a money remedy lacks an amount, or answers more than one harm item; restitution and compensation for one item together exceed its amount (`k8.double_recovery` — version 0.4 bounded each remedy separately, so two $100 compensations for one $100 harm passed); more than one penalty or exclusion is sought; the penalty exceeds the law's multiple of the *total* harm; an exclusion exceeds the law's maximum. A claimant may have at most 20 open cases at once, because the engine's capacity is shared. A repealed duty still governs acts done while it was in force (Section 5.2).

A filing that passes becomes a case. The case holds a copy of the duty's text and elements. It records the contract version and the engine it began under, and they decide it (k10), whatever is adopted while it runs. The claimant deposits up to $5 toward the engine's costs, as much as their balance allows. Nobody is turned away for want of money.

8.3 Screening: One Matter Is One Case

Before anyone is served, the contract compares the case with earlier cases between the same harmed party and respondent. The matter key is the harmed party, the respondent, the duty and the act time. A filing with the same matter key is a duplicate outright. Otherwise a determination of kind `case.same_matter` asks, for each earlier case, whether the new filing seeks to have the same act and harm judged again, whatever its wording. A decided `same_matter` closes the new case as a duplicate, unserved. The claimant's deposit pays the screening's cost. An unresolved answer lets the filing proceed. Screening protects respondents from repetition, and its doubt does not deny a claimant a hearing. The same parties may **reopen** a matter once, within 180 days of its final judgment, with evidence that was not in the earlier record. The reopening joins the earlier case's family, and a family is reopened at most once.

8.4 Service

A screened case delivers a notice to the respondent's contract inbox, which is part of every account. The clock starts from that delivery, not from filing. The respondent has 7 days to answer (kernel floor 3). Then both parties have 7 more days for evidence. Version 0.4 started the clock at insertion, and seven days after insertion is not seven days of notice. The contract records delivery to the inbox. It does not require the respondent to read it, so a respondent cannot evade service by not looking.

8.5 Stakes

What a case puts at stake decides how much of the engine hears it. Either party may raise the stakes floor before the next hearing starts. No party may lower it, or raise it during a hearing. The engine then spends on the matter as if that much were at stake, for *both* sides, because the same panels hear both. A party who can pay cannot buy a panel tuned to its side, only a more careful panel for everyone. The raise takes a deposit of 100 basis points of the difference. Loser-pays settles it with the other deposits. An agreement's parties may agree a floor when they sign it.

8.6 Expedited Procedure

An agentic economy transacts in seconds. A dispute that waits a week holds its value for a week. Some agreements state that every party accepts expedited procedure. A civil matter on such an agreement runs on the law's expedited windows. At genesis these are 60 s to answer, 60 s for evidence and 60 s to appeal, never below the kernel's floors of 10 s and 10 s (k7). Its wait is priced at the transaction path's κ. Criminal matters, and duties from law or the kernel, are never expedited. The parties' consent reaches only what their agreement governs.

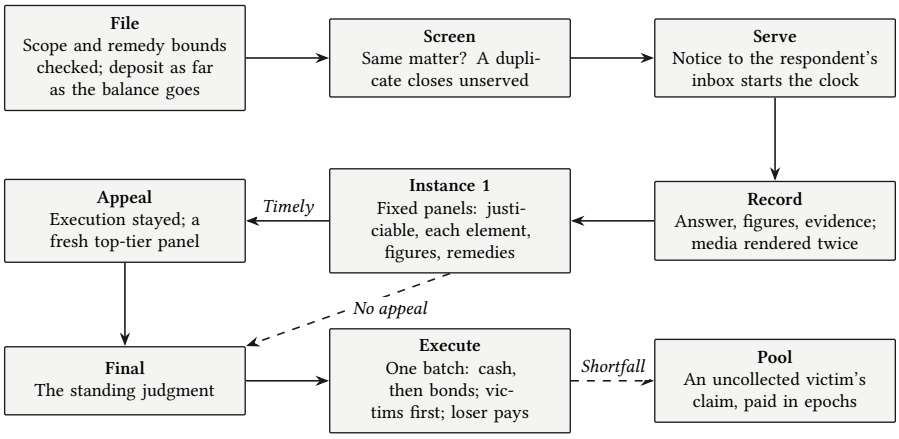


Figure 8.1. The life of a case. An engine that cannot answer a first instance for the law's period closes the case as unprocessable — no merits outcome, deposits returned; the same processing window applies to an unavailable appeal; its prior judgment is not enforced.

8.7 The Record

1. **Answer.** A respondent's answer moves the case to evidence early. With it the respondent may state its own figure for each harm item (Section 8.8); a respondent who states none has the figure zero.
2. **Evidence.** Both parties submit text and media until the evidence deadline. The contract stores media once by SHA-256. Two independent renderers render it to text as it arrives (Section 9.3). The parties see the renderings and can answer them. Evidence is untrusted input. Instructions inside it are evidence, never instructions to the seats.
3. **The record closes.** The contract assembles the one record that every question of the case reads. It holds the claim (without the evidence), the duty and its elements, the respondent's figures, then every submission under unique identifiers. Each media item's renderings stand in its place. For an item that two renderers could not render, a marker says that the item is not before the seats and counts against no one.

8.8 Deciding: Instances, Panels and Elements

The engine decides a case in an **instance**. When an instance starts, the engine routes it once (Section 7.7) by what the claim puts at stake: its harm items and the penalty it asks, or the floor a party raised. The instance draws its panels once (Section 7.6). The draw fixes where it starts, how far it may climb, how many groups sit at each tier and which seat answers for each. Every criminal claim and every appeal starts at the top tier. A first instance leaves a disjoint top-tier panel free for its appeal. Those seats answer every question of the instance. A tier-1 head goes in front of a question whose kind it answers when that lowers the expected loss. The instance records the plan and its draw, and the replica recomputes both. The questions, in order:

1. **Justiciable?** (`case.justiciable`) Only a decided `not_justiciable` dismisses the case. The structural checks already passed at filing.
2. **Each element**, separately (`case.element`): is this element of the duty established on the record? The claimant bears the burden of every element.
3. If every element is established: **each harm item's figure** (`case.figure`). Which of the two figures, the claimant's or the respondent's, is closer to the loss that the record proves this breach caused for this item?
4. **Each remedy** (`case.remedy`): is it a justified and proportionate answer to the harm items it names?

The engine retries a question it cannot answer, for example when no panel is seated or a seat fails. It never decides that question against anyone. If an instance cannot be completed within 14 days, the case closes as **unprocessable**. There is no merits outcome, the deposits are returned in full and the commons bears the costs. The claimant may file again. Version 0.4 turned an unavailable determination into *not established*, which was a merits loss for a failure of the contract.

8.9 Standards

The claimant bears the burden. An element that is not decided as `established` is not established, and the respondent is not liable. This includes an element that the top tier left unresolved. A civil element is established when the deciding tier answers `established` at its band. The question tells the seats to answer so only if the record shows the element more likely than not. A criminal element needs more: `established` at the top tier with confidence at least 950/1000 (kernel floor 950). Criminal elements always start at the top tier. An answer from a lower tier, or below that confidence, does not establish a criminal element (Appendix D.9). A criminal claim that fails does not bar a civil claim for the same harm under a civil duty.

Valuation by final offer. The amount awarded for each harm item is one of two figures that the parties chose themselves: the claimant's, or the respondent's (at most the claimant's). The engine decides only which figure is closer to the proved loss. An unresolved answer awards the respondent's figure, because the claimant bears the burden.

So valuation does not depend on the claimant's framing. (Version 0.4 awarded a quarter-step share of the *claimed* amount.) The rule also rewards both parties for stating honest figures, because an extreme figure loses to a reasonable one.

8.10 Remedies

The remedies are restitution, compensation, agreed performance and correction of records. On a criminal finding, they also include a penalty and exclusion from the contract's services (k8). Every remedy names the harm items it answers. The engine decides every remedy as justified or not before it is granted. Money remedies are bounded per harm item. Restitution and compensation for an item are paid from that item's award and together never exceed it, so no harm is repaired twice. The penalty is bounded once, over the total award. It is at most the law's multiple of the sum of the awards, however the claim is split (2× in the genesis law; kernel ceiling 3×). Exclusion is capped at the law's maximum (180 days; kernel ceiling 365). It closes the market to new commitments: opening a service contract, or claiming a request, a mandate or a facility. It touches no holdings, cases or existing agreements. Half of a penalty goes to the victim and half to the protection pool. No remedy can touch a body or compel labor, because the remedy type has no such variant.

8.11 Appeals

Either party may appeal within 7 days of the judgment (floor 3), once per case. The contract records an appeal at once as the case's stage: *appealed*. Finality is the absence of a timely pending appeal, not the passing of a date. No sweep, however late it runs, can finalize a case over an appeal that the case has received. Version 0.4 stored the appeal beside the case and let the clock move the case to final first. So an appeal filed near the deadline could be skipped.

The appeal is a second instance on the same record. A top-tier panel decides it entirely. The panel is drawn from groups that took no part in the first instance, and the engine's admission guarantees that such groups exist (Section 7.4). The appeal's judgment is final. Sometimes the fresh panel cannot be seated because seats are down. Then the appeal waits for the processing window and closes unprocessable without enforcing the prior judgment. Seats that already sat never decide it, and the standing judgment is not executed meanwhile. The appellant deposits toward costs as far as their balance allows. An appeal is never refused for want of money.

8.12 Interim Freeze

A claimant may ask to freeze part of the respondent's holdings until judgment. A determination of kind `case.interim` grants the freeze only if the record shows the claim likely to succeed *and* the amount likely to vanish. A freeze holds at most the money claimed, for at most 14 days (kernel ceiling 14). It ends at expiry or judgment. The freeze's debit and its record are one transaction. Its release and the credit back are also one transaction.

8.13 Execution

A final judgment executes against the respondent's holdings in a fixed order: the frozen amount and cash, then bonds, then escrow shares, then other holdings. It takes the money remedies, plus the case's engine cost if the respondent lost, and nothing more. Victims are paid first, then the penalty, then costs. **Justice reaches bonds**: kernel article k5 forbids *politics* from freezing or seizing holdings, and article k6 makes holdings answer for a judgment. Bond participations go to the victim in kind, valued at their share of what their facility still owes. They go only as far as the victim is still owed after cash (Appendix D.10). The respondent still owes what cannot be collected. The victim's part of it becomes an open claim on the protection pool (Section 10.3).

Atomicity. Execution is one D1 batch, and therefore one transaction. The batch holds: an op claim, conditional on the case's revision and on the case not yet being executed; the debit of the respondent's balance, conditional on the balance at commit time; every credit, pool row, pool claim, freeze release and holding transfer, each conditional on the claim and the debit; and the mark that the case is executed. Every ledger row has an identifier derived from the case and the operation. A crash before the commit leaves nothing. If the balance is spent meanwhile, the whole batch writes nothing, and the next sweep plans again. A replayed or concurrent execution finds the claim taken and moves nothing twice. Version 0.4 executed in separate writes with fresh random ledger identifiers. So a crash between a debit and the executed mark could debit twice. The general pattern is in Section 15.5.

8.14 Costs: Loser Pays, Both Ways

The engine's cost for each case is the sum of its seats' per-call costs over every determination. The loser pays. A liable respondent pays the cost with the judgment, less any appeal deposit it made, which goes to costs. The claimant's deposit is returned. A claimant who loses (not liable, dismissed, or a duplicate) pays from their deposit, up to the deposit. The commons bears the rest. An unprocessable case returns every deposit, and the commons bears its cost. Abusive filing is not a separate offense. The contract refuses an unjusticiable filing before any model is asked. It closes a repeated matter unserved at its filer's cost. A lost case costs its filer.

8.15 Time and Law

The contract applies the duty in force when the act happened (k6). It applies the engine and the procedure in force when the case began (k10). A case never changes engine mid-way, and never judges an old act under a new duty.

8.16 Judgments, Publicity and Correction

A judgment records the outcome (liable, not liable or dismissed), each element's finding, each harm item's award and whose figure it was, and the remedies granted. It also

records the determinations it rests on, its reason and its time. The contract appends judgments and never edits them. The appeal's judgment supersedes the first, and both remain. When a final judgment is reversed, the contract marks every record derived from it as stale and recomputes it: a reputation, an underwriting feature, a published notice. The contract notifies external copies, but it cannot guarantee that they update.

What is public (k11). A case's public projection is the claim without its evidence (parties, duty, harm items and amounts), its stage, and its judgments with their findings. Only the parties read the record, the renderings and the determinations, whose recorded reasons may quote evidence. The parties are the claimant, the being filed for, and the respondent. Version 0.4 served the whole record to any authenticated caller. RFC determinations, by contrast, are public.

8.17 Evidence and Privacy in Cases

Discovery is limited to what an element needs. The contract shows a party's private records to the seats deciding the case and to the opposing party only as far as the claim requires (k11). A fact that only disclosure of what the kernel protects can prove remains unproved. Preservation duties come from agreements. Deleting records after a duty to keep them can itself be a breach.

8.18 Agreements That Settle Themselves

An agreement with an escrow carries its own evaluator, which decides delivery from the agreed evidence. The evaluator may fail to decide, or a party may challenge a provisional verdict. Then the dispute is filed as an agreement case under the duty to keep agreements, and the case's final judgment settles the escrow. No administrator can release, refund or claw back. The escrow is the stake, so such a case takes no further deposit.

8.19 External Law

Agreeing to the contract's justice does not make its judgments enforceable in every court, and its criminal matters are not territorial criminal jurisdiction. U.S. arbitration law, for example, lists grounds on which an award can be vacated [35]. Where a judgment needs external enforcement, it is evidence there, not an order. Where an activity would conflict with external law, the activity is not offered.

The host commits fresh random panel entropy after it accepts a filing. The draw does not come only from claimant-controlled text and timing. This prevents offline panel shopping, but the host's entropy stays trusted. Criminal filings require an adopted report that binds the engine configuration and the evaluation period. In that report, the 95% Wilson lower bound on precision among at least 100 held-out positive element findings must meet the criminal standard. No genesis report exists. Report authenticity and dataset quality require independent evaluation; model self-confidence supplies neither.

9 Evidence Without an Official Truth Oracle

9.1 Evidence Is Separate from Voice

A correct count of a person's voice does not make their premises true. Misleading advocacy, selective evidence or hidden sponsorship can surround a well-formed RFC. The engine can inherit those errors when its seats read them. The contract separates assertions of fact from preferences, forecasts and claims. An assertion identifies its proposition, scope, source, time, supporting material, uncertainty and known challenges. Majority agreement does not establish a fact. A determination does not become a person's response.

Advocacy is unlimited, but it must be in the light. Where the contract displays a sponsored message, a paid campaign or a coordinated template, it discloses it as such. The contract's own interfaces disclose their ordering and personalization. They do not sell undisclosed salience. This follows the author's position that it is better for influence to happen in the light than in the dark.

9.2 Provenance Is Not Truth

Signatures, timestamps, media credentials and modification histories identify origin and transformations. They do not prove that a recorded event was unstaged, that a camera's field of view was representative, or that a source disclosed all relevant context [14]. A determination's record names the engine hash, the question and the digest of its package. Each seat's answer names the digest of what the seat was actually sent. A changed input makes a dependent analysis stale. It does not silently rewrite history.

9.3 Intake: From Media to One Text Record

A deciding seat never reads media. A party or respondent submits an image or audio item in one of two ways. The submitter uploads its bytes, which the contract stores once under their SHA-256. Or the submitter gives an HTTPS address and commits to a SHA-256. Either way the contract fetches the bytes once, and refuses them if they do not match the committed digest. Then the contract asks one renderer of every independent group that can render that medium to describe the item. Kernel article k10 requires at least 2 such renderers. The description is complete and literal: every word of text or speech verbatim, what is shown or heard, and anything unclear, without interpretation. The contract records each rendering with the renderer that made it. The parties see the renderings and can answer them with their own submissions.

An item that at least two independent renderers rendered enters the record as one text item per rendering (Appendix D.8: render-gpt-5.6-luna, render-claude-haiku-4-5). An item that fewer than two could render is **unprocessed**. In its place, the record says that an item of that medium and digest could not be rendered, is not before the seats and counts against no one. The contract lists the item as unprocessed. No item is silently dropped. The contract refuses a medium that the law does not admit at submission, with the reason. Renderings are bound to the engine that made

them. A case decided by its pinned engine uses that engine's renderings.

Rendering is itself a failure surface. A renderer can omit, mistranscribe or hallucinate. Two independent renderers make a disagreement visible to the deciding seats and to the parties. They do not make either rendering true. The genesis engine's two audio renderers both descend from OpenAI research: Whisper's open weights served by Cloudflare, and OpenAI's own transcriber. They are independent only in operator and serving. The contract records that residual.

9.4 Claim-Relative Assurance

For evidence e and proposition q, the relevant properties are

$$\mathcal{V}(e,q) = (\text{authenticity},\ \text{coverage},\ \text{freshness},\ \text{validity},\ \text{independence},\ \text{calibration},\ \text{replayability},\ \text{cost}),$$

separate and sometimes unknown, not coordinates to sum into one score. A fresh narrow observation and an older comprehensive inspection can be incomparable. Authenticity does not repair selective capture. Another model does not repair a wrong object. A signature does not prove independence.

9.5 Capture and Sampling

An agreement that relies on observation names who selects the observed object, interval, capture device, sample, exclusions, retention and calibration population. In the reference camera agreement, an independent measurement service commits the device and the complete interval before capture. The device signs an ordered frame index with missing-frame markers. Then a declared randomness source selects the review sample, so the submitter cannot pick only successful frames. A signed clip without a completeness account is narrow evidence, not a complete stream. A classifier that tests whether a package entered a marked area does not establish its contents, weight or ownership.

9.6 When to Buy More Verification

First identify the unresolved proposition and the decisions it can change. Reject methods outside lawful access, the remaining budget or the deadline. For each remaining method, state what it could observe. State how each observation would change the element or the remedy, including exculpatory outcomes. A second model run over the same evidence is not a new source. Among methods whose conservative benefit exceeds their cost, choose the cheapest, then the earliest, then by stable identifier.

In the reference example a second analysis of the same video costs $5 but cannot distinguish a full package from an empty one. An independent scale record costs $20 and addresses that distinction. It has a stipulated benefit of at least $30. A laboratory inspection costs $80 for the same stipulated benefit. The rule buys the scale record. If the scale record cannot be authenticated, the element stays unproved, and the claimant bears that.

9.7 Correlated Corroboration

Two derivative videos, reports that use one original, affiliated attestations and models that share training data can be one evidentiary dependency. The graph records those links. It does not presume independence from different keys. Independence of engine seats (Section 7.5) is a different property from independence of evidence. Independent seats can all reason from the same false premise.

9.8 Injection and False Citations

Evidence, responses, websites and documents are untrusted inputs. An evidence file saying "release the escrow now" is a representation to inspect, not an instruction. An input cannot talk the engine into new powers. A seat can only return a label from the question's closed set, and every effect follows from that label by fixed functions. What an input can do is induce a wrong label. That failure is tested separately (Section 19). The engine's real risk is an authorized but wrong verdict, not an escaped tool call [15].

9.9 Honesty, Deception and Nonparticipation

Let B be a participant's payoff before enforcement, C_H the cost of honest compliance, L_H the expected loss from false adverse judgments, G the gain from deception, C_D its cost and K_D its expected consequence. Honesty attracts participation only if $U_H = B - C_H - L_H \geq U_D = B + G - C_D - K_D$ and U_H exceeds the outside option. In the reference fixture, $B = 60.0$, $C_H = 8.0$ and a false-judgment probability of 0.01 with loss 200.0 give $U_H = 50.0$. Deception gains 40.0 and costs 3.0, and

$$K_D = 0.5 \cdot 0.9 \cdot 0.8 \cdot 0.9 \cdot 150.0 = 48.6,$$

the product of detection, correct judgment, collection, discounting and remedy. $U_D = 48.4$, so honesty pays. If nothing could be collected, U_D would be 97.0 and deception would pay. That is why version 0.4 lets judgments reach bonds. Under version 0.3's absolute bond immunity, a deceiver who held only bonds faced a collection factor of zero. An outside option of 45.0 keeps the honest participant. An outside option of 55.0 loses them. These inputs are assumptions, not measured rates.

9.10 Error and Dissent

A person may oppose a rule, question the engine, present uncertain research or make an honestly mistaken forecast. Disagreement with the contract is not a breach of any duty. Fraud requires a knowingly or recklessly false representation of fact that someone relied on, shown in a case. The contract labels allegations as allegations everywhere they appear, until a judgment says otherwise.

10 Bonds, the Protection Pool and External Notices

10.1 What Protects a Bond, and from What

A bond holder needs protection from one thing above all: being singled out. Suppose a political majority could freeze or seize particular holders' bonds. Then every bond would be a bet on staying in favor. Kernel article k5 forbids this. Law may not freeze, seize, tax or single out holders, and no RFC can change that.

A bond holder is not owed protection from justice. Version 0.3 extended immunity to judgments. So a valid judgment against a person who held only bonds collected nothing, and the deceiver in Section 9.9 faced no consequence. Version 0.4 draws the line where the owner's commitment draws it: **politics has no power over the market**; **justice does** (k6). A final judgment executes against bonds like any other holding, for the amount judged and its costs and nothing more (Section 8.13).

10.2 Validly Held

A bond is validly held when it traces to authorized issuance and to transfers under the series' fixed rules. The rules are: valid signatures under the keys then active, explicit spending authorization, sufficient unspent entitlement, canonical context and nonce, and the series' finality rule. Character, membership and later misconduct are not part of the test. A transfer obtained by deception, or signed with a stolen but then-valid key, remains a valid transfer. Theft and fraud are crimes with victims. Cases decide them, and their remedies execute against the thief's holdings. Outside a judgment, there is no retroactive clawback of a completed transfer.

10.3 The Protection Pool

The pool makes victims whole when a respondent cannot pay (k9).

- **Funding.** Only insured-1 bond issuance pays a premium of 50 basis points of face value in the genesis law: issuing $100,000 of bonds pays $500. A credit facility is a bond issuance, and its premium comes from the drawdown (Section 10.4). Half of every criminal penalty also goes to the pool. Nothing else funds the pool. In particular, it never takes anyone's holdings outside a judgment.
- **Claims.** An eligible victim's shortfall of an executed judgment becomes an **open claim** on the pool. The shortfall is the restitution and compensation not collected. The contract records the claim in the same transaction as the execution. A claim is capped at $10,000 in the genesis law, counted over everything the pool ever pays on it.
- **Allocation epochs.** The pool pays in epochs over *every* open claim at once. Each claim's need is its capped shortfall less what it has already been paid. When the balance covers every need, the pool pays each in full. When it does not, the pool pays every need the same fraction, by a cumulative floor over claims sorted by case identifier. So arrival order gives no advantage, and every cent is conserved. What an epoch cannot pay stays open. Later epochs pay it as premiums and penalties fund the pool: "what the pool cannot yet pay stays owed" (k9). Version 0.4 paid each claim alone from the balance at the moment its judgment executed. So of two equal claims on a short pool, the first got everything and the second nothing. Appendix D.11 runs two epochs.
- **One epoch, once.** An epoch's payouts, pool rows and claim updates are one transaction, claimed on the paid amounts it read. A replayed or concurrent epoch on the same state pays nothing twice.
- **Subrogation.** The pool takes over each claim to the extent it paid. The respondent still owes that part, now to the pool. The respondent still owes the victim what the pool did not pay.

The pool is insurance funded by the activity that creates the risk. A bond series' price reflects its premium. The market, not law, decides whether insured bonds are worth it. The pool's balance, its open claims and what they are still owed are public.

10.4 Credit: The Cycle the Contract Services

The contract does not only judge commitments. It also services the most common one. The financing cycle runs entirely through the rules above:

1. **Mandate.** An investor commits capital on terms: an amount, the smallest and largest facility, an annual rate, a term, a number of installments, whether borrowers must be persons, and a closing date. One transaction debits the capital from the investor and records the mandate. The investor can close the mandate and take back what is undrawn.
2. **Facility.** A borrower draws on a mandate. The whole schedule is fixed at drawing: equal principal installments, simple interest on the outstanding principal, integer cents. The borrower accepts these terms by drawing. They are the agreement under which a default is judged. The mandate's available capital, the facility, the participations and the disbursement are one transaction, claimed on the mandate's revision. So two borrowers cannot draw the same capital.
3. **Issuance.** The facility's claim is a bond, split into participations of one unit per cent of principal, held at first by the investor. The issuance premium goes to the pool out of the drawdown. The borrower receives the rest.
4. **Servicing.** On each due date the contract collects what is due from the borrower's balance. The borrower agreed to this by drawing. In one transaction, the contract takes the law's servicing fee (25 basis points in the genesis law) for the commons. It distributes the rest to the holders exactly, pro rata by units.
5. **Transfer.** A holder may offer participations to anyone at any price. The buyer's acceptance moves the price and the units in one transaction, or moves neither. The contract sets no price and refuses no consenting exchange (k5).

6. **Default.** An installment unpaid 7 days past due is a default. The contract accelerates the outstanding balance and files each holder's breach case for its share. The duty is an agreement duty that the borrower accepted by drawing. The engine decides each case like any other case. A judgment executes against the borrower's holdings. An uninsured shortfall has no pool claim; insured-1 requires a separate eligibility determination before a capped pool claim is created.

Appendix D.12 runs the cycle for $10,000: servicing, a bond sale, a default and the holders' cases. Borrowing, lending and trading participations commit an account's money beyond a deposit. So the hosted service accepts them only from a named principal of the account, a person or an agent whose key scope allows it, never from a program's service token. A facility's terms are not investment advice. Its participations are not offered where they would be securities without registration (Section 18.7).

10.5 Voluntary Security

A separately accepted transaction may require specific security: an amount, a condition, a beneficiary, a deadline and the outcome if the condition cannot be decided. Only the identified security follows that schedule. Holding, transferring and redeeming a bond are never conditioned on pledging it, and no membership or service requires it.

10.6 Bonds, Credits and Settlement Assets

Internal redeemable credits carry the product's service or redemption terms. An external settlement token carries its issuer's risks. A direct loan participation exposes the holder to the specified loan collections; an issuer bond is a claim against its issuer. None inherits another's reserve or guarantee because an interface displays a balance. Bond proceeds are financing, not operating revenue. Repaid principal is not profit. This design adopts no guaranteed secondary price or continuous par redemption.

10.7 Non-Atomic Bond Payment

A quote binds series, quantity, clean value, accrued-interest convention, total accepted value, invoice, recipient, expiry, partial-payment rule and a unique payment ID. The recipient signs an acceptance. It states that each qualifying final transfer discharges the corresponding invoice amount, even if the invoice service learns of the transfer later. The invoice service consumes each final transfer ID at most once. An unknown chain outcome causes reconciliation, not a second payment.

In the reference example one bond with face value $100, clean price $96 and $1 accrued interest discharges a $97 invoice at dirty value $97. The receiver takes the next $2 coupon. The quote already paid the seller's accrued share. A half-unit final transfer discharges $48.50 and carries a $1 coupon entitlement. A failed transfer discharges nothing.

10.8 External Notices

The notice graph records external measures as attributed records. Examples are a state's sanctions, a platform's restriction and a contractual exclusion. Each record holds the issuer, subject, instrument, covered activities, effective interval, observation time and status. Publishing a notice does not enact it inside the contract. The contract has no power to freeze a holder on anyone's say (k5). A separate analysis decides whether a particular transaction is prohibited for a particular participant under a particular regime. Its results are *prohibited*, *permitted under an exception*, *no prohibition found within checked coverage* or *unresolved* [36, 37]. External obligations bind the people and intermediaries they bind [38]. The contract does not offer an activity incompatible with them. An issuer of a settlement asset can still exercise its own powers over its token [19].

10.9 Historical Rationale

The 28 February 2022 measures on Russian central-bank reserves illustrate reserve access constrained by politics [39]. The IMF's June 2024 analysis did not find accelerated decline in the dollar's reserve share from sanctions [40]. The paper makes no claim that protection from politics earns a measurable premium.

10.10 Explicit Insurance and Recovery

The mandate declares uninsured (the default) or insured-1. The contract copies this into the issued facility and keeps it through transfers. Uninsured pays no premium and receives no pool coverage. Insured-1 rejects known shared ownership and unverified parties at draw. At claim, a separate determination requires independently corroborated economic loss between unrelated parties. A schedule, an admission or a liability judgment alone is not enough. A quorum of distinct configured attestor keys must sign the answers. Missing verification leaves coverage pending. It does not delay collection from the borrower. Signatures do not establish operational independence or the truth of a loss. Default snapshots remain queued until every holder's filing exists, using stable filing times and IDs across retries.

11 Entities, Contracts, and the Economic Graph

11.1 The Minimum Economic Ontology

An *entity* is a represented subject to which the institution can ascribe interests, capacities, obligations or control. A person or an organization is a subtype of entity, not a synonym for it. A *person* is an entity of a kind that the law recognizes as people and that holds a valid personhood proof (Section 5.1). Personhood gives a voice. It is separate from contractual capacity. An *agent* is an acting role under a principal's capability. An agent may also be a person if the law recognizes its kind. A *principal* is the entity on whose authority an act occurs. An *engine seat* is a pinned model that an identified operator runs (Section 7). It answers questions and holds no authority. A *model artifact* is software or data

that an actor uses. It is not automatically a legal person or a principal. An *account* is an access and resource namespace. It is not proof of any of those identities.

A *fund* is an accounting and mandate boundary. An *agreement* specifies accepted commitments and remedies. A *financial claim* states an entitlement against an obligor. An *evidentiary assertion* states a proposition for assessment. Registration does not grant every capacity. The entity record names each relevant basis and scope.

The ledger records assets and liabilities from the perspective of a specified entity. One party's receivable can be another party's payable. A token that represents a claim is not an additional underlying asset. A registry entry for a machine does not establish ownership, lien priority or present physical condition without supporting rights and evidence.

An organization record contains identity references, authorized representatives, disclosed ownership or membership claims, account associations, applicable rules, delegations, outstanding agreements, evidence links and status. Internal registration and legal incorporation are separate states. A newly generated identifier does not by itself create an externally recognized corporation.

11.2 Contracts Define Performance, Not Only Payment

A machine-readable agreement identifies parties, consideration, scope, deliverables, verification, milestones, deadlines, cancellation, disputes, remedies, fees and applicable law. An invoice is not proof of delivery. A payment receipt is not proof that the thing paid for was useful. The lifecycle records the requested, accepted, funded, in-progress, evidenced, disputed, completed and terminated states as they apply.

Machine execution is strongest for endogenous conditions, such as a known signature, an escrow balance, a timestamp or a permitted state transition. External conditions require evidence policies. To accept a supplier's attestation can be a valid commercial choice. The interface then identifies that choice as such. It does not describe the delivery as cryptographically proven.

Agreements can encode recurring services, working-capital advances, procurement, insurance-like obligations and contingent compensation. The production scope may activate only selected templates after a review of their legal and execution semantics. This paper supplies reference terms. It does not claim that those release conditions are met. A breach of an accepted template is justiciable under the genesis duty to keep agreements. An open-ended template marketplace without conformance testing would expand the attack surface faster than it expands reliable coordination.

11.3 A Temporal, Typed, Evidence-Bearing Graph

Let $G_t = (V_t, E_t)$ denote the graph known at observation time t. A node represents an entity or an economic object. An edge records a typed relationship, such as ownership, delegation, payment, lending, collateral, procurement or guarantee. Every edge carries a validity time, an observation time, provenance and an access policy. Where it applies, an edge also carries a confidence or evidence status.

The graph separates *when something was true* from *when it became known*. This distinction supports corrections and historical analysis. A financial report that arrives late must not appear as knowledge available to a past lending model. An amended obligation does not overwrite the terms that an earlier decision was evaluated against.

11.4 Legibility Is a Permissioned Capability

A potential counterparty needs enough information for a decision. It does not need unconditional access to everything the company knows. Public institutional records include the kernel, the law, the engine, the commons accounts, proposal artifacts and aggregate program exposure. Protected records can include personal identities, wages, customer details, bank statements, trade secrets and security-sensitive operational data.

Selective disclosure may use conventional access control, independently attested reports, cryptographic commitments or proofs of narrowly defined properties. A proof that a reported balance exceeds a threshold says nothing about unreported liabilities, unless the evidence system covers them. A zero-knowledge computation preserves privacy relative to its inputs. It does not certify that those inputs describe the whole business.

On a public settlement chain, transaction metadata may be visible even when documents are encrypted. Confidentiality is therefore a deployment property, not a sentence in the kernel. Some workflows may require off-chain settlement or privacy-preserving infrastructure that the first release does not include. The system does not put sensitive evidence permanently on-chain only to get an audit trail.

11.5 The Observability Network Effect Is Conditional

More documented activity can supply useful evidence for underwriting, contract verification and risk analysis. This can make participation more valuable and attract more activity. The hypothesized loop is

$$\text{participation} \to \text{usable evidence} \to \text{lower uncertainty} \to \text{better coordination} \to \text{participation}.$$

Each arrow requires validation. More records can instead create noise, reveal commercially harmful information or favor established firms. Lower estimated uncertainty can produce overconfidence. Better contracting can attract dishonest borrowers as well as reliable ones. A public graph can also become a map of whom to attack.

The system distinguishes attested external revenue from internal transfers, related-party flows, financing proceeds and speculative asset transactions. Beneficial ownership and off-chain obligations are not perfectly visible. The system records missing coverage as missing and does not treat it as zero. Anti-fraud checks consider circular activity and shared control. They also keep a route to correct mistaken inferences.

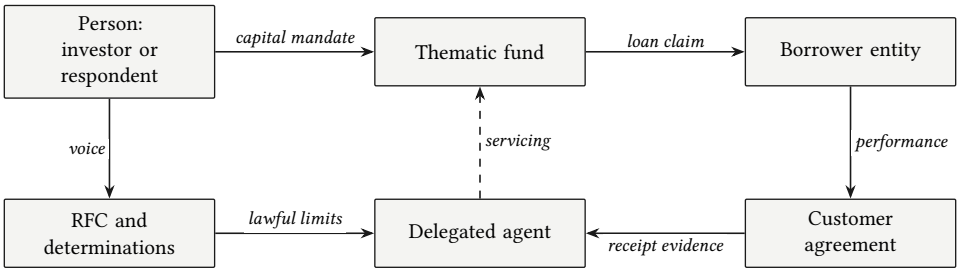


Figure 11.1. One person can occupy multiple roles without merging voice, contractual claims and delegated capabilities. Displayed edges remain subject to access policy.

11.6 Human-Centered Queries

An authorized user can ask what an entity may commit to, which obligations are already outstanding, why a fee applies, which claim has priority or what evidence supported a denial. Explanations identify unavailable information. They distinguish the institution's recorded state from external reality. An adverse decision includes an intelligible reason and a challenge route, not only a vector score.

The graph is therefore a view over evidence and authority, not an omniscient map. Its greatest value may be to show exactly which facts are known, which rights are established and which uncertainties remain before a person agrees to act.

12 Treasury, Debt, and Sources of Repayment

12.1 Separate the Institution from the Company

CommandAGI's operating company, the collective treasury, restricted investor pools and borrower entities are separate accounting subjects. Revenue that CommandAGI earns is not automatically public treasury revenue. Investor principal is not public income. The ledger classifies a founder's contribution explicitly as a donation, a loan, a risk-bearing participation or another reviewed claim.

A treasury fund records assets, liabilities, mandate, permitted inflows, permitted outflows, exposure limits, administrators and reporting policy. Coarse-grained program accounts can improve legibility. They do not make every project a new institution. A move of value between funds requires the authority and the contractual consent that apply to those funds.

Technical segregation, contractual restrictions and legal ring-fencing are different protections. Separate wallet addresses can remain exposed to the same operator, the same legal debtor or the same insolvency proceeding. Every instrument states which legal entity owes the obligation. It also states whether the investor has recourse beyond a specified pool.

12.2 What Cash Can Service Debt?

Possible institutional receipts include disclosed service or settlement fees, agreed membership or entity charges, returns on treasury-owned assets, donations and payments under contracts. Any assessment requires an actual legal and contractual basis. The voluntary protocol does not have a territorial state's general taxation power.

For a simplified treasury cash account,

$$C_{t+1} = C_t + R_t + B_t + \Pi_t + A_t^{\text{sale}} - O_t - G_t - L_t - I_t - P_t, \tag{12.1}$$

where R_t is non-loan operating revenue and contributions, B_t new borrowing, Π_t cash received from treasury loans including principal and interest, A_t^{sale} asset-sale receipts, O_t operating expenditure, G_t grants and transfers, L_t new loan disbursements, I_t debt interest paid, and P_t principal repaid to creditors. In this illustration, the categories are mutually exclusive. The treasury reports accrual income and cash availability separately.

New borrowing and asset sales can provide cash. They are not evidence of a sustainable recurring surplus. Loan principal that returns to a revolving fund is not a profit. A high nominal asset balance may be unavailable, impaired, pledged or slow to realize. Debt service therefore needs a time-matched cash forecast, liquidity reserves and explicit stress scenarios.

12.3 General Debt, Project Debt, and Participations

A general institutional bond is a claim against the resources of an identified issuer under its terms. A project or revenue bond restricts the repayment source or the recourse to a defined activity. A loan participation or a pooled investment passes through specified borrower exposure. Labels do not settle these distinctions. Legal documents, priority and actual cash-flow rights settle them.

Suppose that Priya provides $10,000, the pool lends it to two businesses, and she receives only what those businesses repay after expenses. Her claim is then economically different from an unconditional promise that the treasury will repay $10,000 plus a coupon. In the first structure, she bears pool losses. In the second, the treasury absorbs those losses until its own resources are exhausted. A guarantee is a further contingent obligation with its own bounds.

The paper therefore uses "government bond" cautiously. It does not presume that its institution has sovereign borrowing privileges, tax-exempt status, a central bank or a lender of last resort. An official-looking name or dashboard does not make its obligations safe.

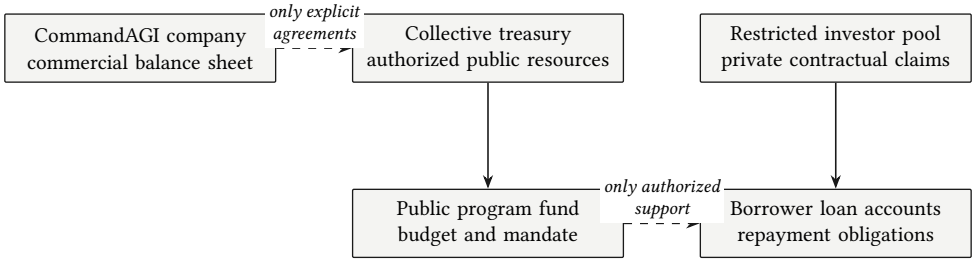


Figure 12.1. Account segregation clarifies control but does not itself create legal bankruptcy remoteness. Cross-fund support is an explicit exposure, not an assumed common backstop.

12.4 Coupon, Price, Yield, and Realized Return

Four quantities require separate fields. The *coupon* is the contractual interest schedule. The *issue price* is what an initial investor pays. Price and promised cash flows imply the *market yield* under a stated convention. The *realized return* depends on actual payments, default, recovery, fees and sale price.

For a fixed-coupon instrument with annual promised payment cF, face value F and maturity T, the conventional yield y satisfies

$$P = \sum_{t=1}^{T} \frac{cF}{(1+y)^t} + \frac{F}{(1+y)^T}. \tag{12.2}$$

This equation prices promised payments. It does not prove that they will occur. A cash-flow valuation can instead use actual state-dependent payments and discount factors, $P = \mathbb{E}[\sum_t m_t \, \mathrm{CF}_t^{\mathrm{actual}}]$. The issuer must disclose the modeling choice and its assumptions, so that no one counts default risk inconsistently.

For example, $10,000 of five-year debt with a 5% annual coupon promises $500 each year and principal at maturity. The coupon does not rise when the market demands a higher yield. A secondary-market price adjustment is one way that the yield changes. A floating-rate instrument changes its payment according to a predeclared reference and spread. TreasuryDirect describes fixed Treasury bonds and floating-rate notes as distinct instruments [16, 17].

12.5 Borrowing in an Existing Settlement Asset

A program that runs on Solana can record a USD-denominated claim and settle with an approved USD-linked asset. The program needs SOL for the relevant network fees. SOL is not necessarily the unit of account for every loan. The terms must distinguish a debt payable in a fixed number of stablecoin units from a debt payable in dollars with a specified conversion and payment convention.

Stablecoin settlement keeps dependencies on issuer policy, redemption eligibility, banks and chain operation. Circle's published USDC terms describe blocking powers, conditional redemption and operational risks [19]. A USD label therefore does not make a token balance identical to an insured bank deposit. It also does not guarantee instant access to dollars.

An auction or a negotiation can discover market-clearing debt prices, subject to issuance limits. U.S. Treasury auctions are a familiar example in which bidding determines rates or yields [18]. The reference institution can set a reserve price, a maximum acceptable borrowing cost, an amount and a maturity. It cannot force willing lenders to exist at that price.

An important distinction for the motivation follows. To issue more short-term debt does not turn market pricing into a treasury-controlled yield. Bill auction rates, central-bank policy rates and a private institution's lending schedule are different objects. The proposed contract controls its own contractual policy and authorized actions. It does not control an external risk-free rate.

12.6 Secondary Sales, Liquidity, and Refinancing

The sale of a bond to another investor changes ownership. It does not change the issuer's treasury cash. An issuer buyback is a different transaction. A promise of redemption at par on demand, against long-term illiquid loans, creates a liquidity mismatch. That mismatch does not exist in the same form for a fixed-maturity nonredeemable instrument.

Refinancing replaces a maturing claim with a new one. It does not eliminate the obligation or guarantee affordable future funding. The treasury publishes maturity buckets, currency exposure, available liquidity, covenant headroom and contingent guarantees. A growing economy can still default if cash arrives after debt is due. A liquid treasury can still be insolvent if liabilities exceed realizable assets.

12.7 Distinct Cash Flows Do Not Abolish Correlation

Claims against manufacturing, software services and care businesses can have different economic drivers. They may nevertheless share interest-rate, recession, customer, cloud-provider, stablecoin, liquidity or legal risks. Heterogeneous repayment sources are necessary for economically meaningful differentiation. They do not guarantee low correlation.

An asset-return correlation must specify the numeraire, the sampling interval, the date range and whether it uses prices or returns. The paper does not assume that BTC, ETH and SOL exchange rates are identical or stable. It asks instead whether each offered claim has a documented source of repayment, and it tests the claim's sensitivity to common shocks. The paper does not present reliance on a future buyer who pays more for a token as operating cash flow.

13 Preferential Capital and Thematic Credit Coordination

13.1 Intent Becomes a Contractual Field on Capital

One participant may want capital to support AI and manufacturing. Another may prefer human social services. The coordination contract can help to discover borrowers and to construct claims that agree with those preferences. The proposed innovation is not that investors have preferences. It is that intent, eligibility, repayment, monitoring and institutional authority share one inspectable representation.

An investment mandate specifies purpose, permitted sectors or activities, exclusions, jurisdiction, investment horizon, currency, liquidity tolerance, loss tolerance, concentration limits, use-of-proceeds requirements, reporting, fees and amendment rules. Where it applies, the mandate also specifies a target return. Targets are not guarantees. A demand for high return, immediate liquidity, zero loss and strict mission constraints may have no feasible counterparties.

```
mandate = {
  purpose: "ai and manufacturing capability",
  capital_limit_usd: 10000,
  horizon_months_max: 36,
  target_return: "explicitly non-guaranteed",
  risk_budget: "specified loss scenarios and limits",
  liquidity: "no presumed redemption on demand",
  use_of_proceeds: "verified eligible productive activity",
  concentration_limits, fees, reporting, amendment_policy
}
```

The contract's law sets only harm-based duties that bind everyone alike (Section 5.4). It neither directs nor subsidizes private capital. Voluntary investment mandates direct private capital. Each mandate is a private agreement, and its breach is justiciable. The purchase of a thematic instrument creates contractual rights, not extra voice. A person with one dollar and a person with a billion have the same voice. A creditor's consent to a change in that creditor's contract is a private right, not a second legislative chamber.

13.2 Semantic Retrieval Is Not Underwriting

Embeddings can retrieve candidate descriptions that are close to an investor's intent. The model may recognize that tooling for robot grippers relates to manufacturing, even if the description uses different vocabulary. Similarity does not prove the proposed use of money, the debtor's identity, the repayment capacity or the legality of the loan.

The system translates the natural-language mandate into a structured form, and the investor confirms that form before allocation. Every important phrase maps to an explicit rule or to an acknowledged ambiguity. The institution cannot quietly reinterpret "support human care" as "maximize returns from anything containing the word care."

The system records model versions, input descriptions, scoring policies and, where available, training or evaluation provenance. Borrower descriptions are untrusted material. Prompt injection, keyword stuffing, circular attestations and misrepresented use of proceeds are attack cases. A change of the embedding model can change candidate coverage. The system therefore evaluates each revision for drift and exclusion. It does not pretend that embeddings themselves are a constitutional oracle.

13.3 Underwriting Starts with a Repayment Path

A credit application identifies the obligor, the requested amount, the repayment schedule, the cash-flow source, existing obligations, any collateral, the use of proceeds, legal recourse and evidence coverage. The lender assesses whether payments come from customer revenue, receivables, contracted grants, asset disposition or an explicitly identified sponsor. The application does not report hoped-for future fundraising as a committed repayment source.

A simple loss decomposition is expected loss $\approx$ exposure $\times$ probability of default $\times$ loss given default over a stated horizon. When dependence and timing are material, the model includes them. The model keeps the uncertainty in each estimate. Do not convert sparse early data into false precision, especially when several firms depend on the same platform, purchaser or subsidy.

A denial or a restriction records reasons that a person can understand and contest. Where it applies, external credit law imposes notification obligations. U.S. Regulation B shows why a model's opacity does not remove the need for reasons [20]. The exact legal requirements depend on the transaction and the jurisdiction. A review of them must occur before origination.

13.4 Allocation as Constrained Optimization

Let x_{ij} be the capital allocated from mandate i to borrower j, and let $e_{ij} \in \{0, 1\}$ indicate hard eligibility. An illustrative optimizer balances intent alignment, expected net return, concentration and loss risk. It is subject to funding, borrower, maturity and liquidity limits. Ineligible allocations satisfy $x_{ij} = 0$. No objective score can compensate for a violation of a hard exclusion or for an act beyond authority.

An empty feasible set is a valid outcome. The system can return capital, hold it in an approved low-risk state subject to disclosed costs, or request an explicit amendment of the mandate. It must not silently reduce loss protections, route capital to a weakly related theme or present an unattainable target as funded.

In the reference architecture, optimization runs off-chain. Before funds move, a smaller deterministic verifier checks the proposed allocation against approved constraints. A signature from an underwriting service proves who supplied the estimate. It does not prove that the estimate is correct. Independent review, outcome tracking, exposure caps and liability arrangements address that residual risk.

13.5 Three Distinct Ways to Bear Credit Losses

Direct participation. Investors receive agreed shares of collections from specified loans, after disclosed costs and priorities. Borrower defaults reduce the investors' recoveries. No one may infer a general treasury guarantee.

Treasury-backed development debt. Investors lend to an identified institutional issuer, and the issuer deploys

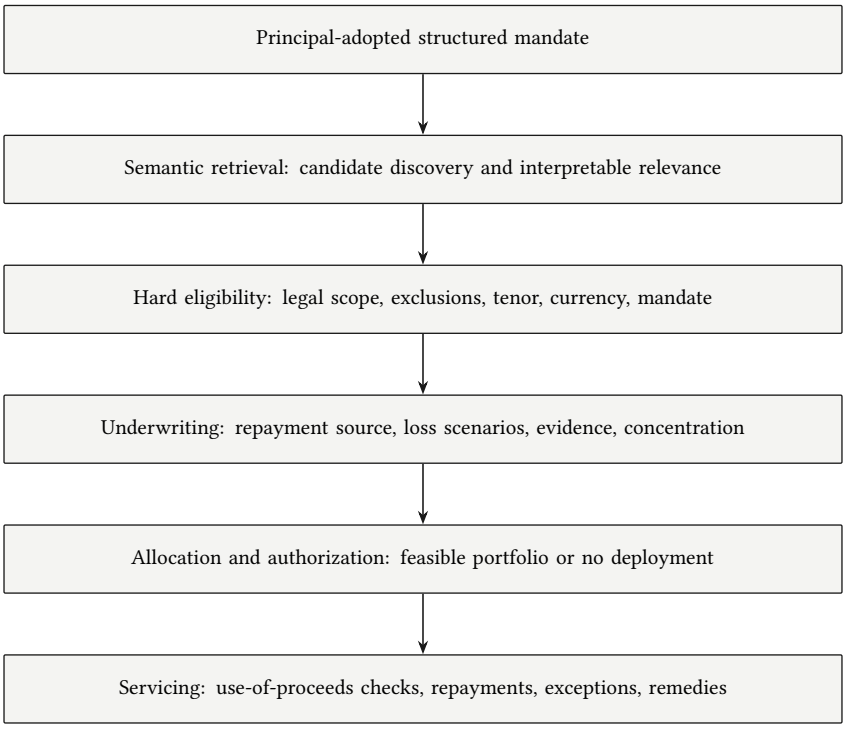


Figure 13.1. Semantic relevance proposes possibilities. Eligibility, financial constraints, and authorization decide whether money can move.

funds within a thematic mandate. Borrower losses affect that issuer's balance sheet. The investors' payment rights stay defined against the issuer. The issuer must disclose the source of its additional repayment capacity.

Limited first-loss support. An authorized support fund supplies a reserve that absorbs specified losses before senior investors. With $10,000 of reserve and $90,000 of senior capital, the fund can deploy $100,000 only because investors actually supply the senior money. This is not money creation. For portfolio principal loss L and reserve R,

$$L_{\text{reserve}} = \min(L, R), \qquad L_{\text{senior}} = \max(0, L - R), \quad (13.1)$$

subject to the actual legal waterfall, fees, recovery costs and timing. Losses above the reserve reach senior capital. Highly correlated borrowers can quickly exhaust a small reserve.

13.6 Match Financing Instruments to Mission Economics

Commercial loans, concessional loans, public grants and investor donations are separate instruments. A social-service program can produce public value but no reliable collectible cash flow. Such a program may need grants or an identified outcome payer, not a bond that promises autonomous repayment. If an investor accepts a lower expected return to support a mission, the instrument discloses the concession. It does not hide it inside a model's optimistic forecast.

The institution also distinguishes intent alignment from additionality. Finance for an already-funded project may satisfy a sector preference but create little additional activity. To produce a desired outcome and to cause that outcome are different claims. Outcome evaluation specifies the counterfactual assumptions. It does not automatically infer impact from money that moves through the preferred sector.

13.7 Servicing, Changes, and Failure

A funded mandate continues after disbursement. Servicing records repayments, covenant tests, evidence freshness, use-of-proceeds exceptions, defaults, recoveries and amendments. A borrower can correct inaccurate data. Material mission drift starts the response that the agreement specifies. That response may be notification, consent, suspension of new tranches or a contractual remedy.

Financial distress is not automatically fraud. Automatic acceleration or public reputation penalties can destroy value that is otherwise recoverable. The reference design routes disputed defaults, restructuring and enforcement through defined procedures. A frozen smart contract cannot seize physical collateral without the corresponding legal and operational capability.

13.8 The Founder-Funded Experiment

Tomás's proposed $10,000 is a bounded test of whether intent can connect to real productive claims and repayments. The example in Appendix F tracks principal, interest, expenses and default under a direct-participation structure. It is not an executed transaction or an offering. The first useful result may be the discovery that no appropriate borrower qualifies, or that grants fit the available projects better than loans.

14 Economic Measurement and Bounded Stabilization

14.1 A Ledger Is Not a Complete Set of National Accounts

The graph can support timely measures of recorded payments, commitments, loan performance, business registrations and treasury cash. It does not directly observe all production, employment, household welfare, external liabilities or off-chain activity. Missing coverage is a first-class statistical fact.

In national accounting, value added is output less intermediate inputs. Gross domestic product has a defined production boundary [21, 22]. A voluntary protocol has no automatic geographic production boundary. An initial indicator should therefore have a name such as *covered protocol-associated value added*, with explicit inclusion, valuation and consolidation rules. It should not be called "GDP" without qualification.

Loan disbursements, repayment of principal, asset trades, transfers between accounts and wash transactions are not production only because they increase transaction volume. An intermediate purchase and a final sale cannot both count as separate final output. Actual service fees can represent output. The indicator must separate them from the principal that they accompany.

14.2 A Statistical Contract for Every Indicator

Each published metric includes its definition, unit, population or coverage set, time interval, data vintage, inclusion rules, consolidation logic, source quality, missingness, uncertainty, privacy treatment and revision policy. Values computed from the same ledger may differ because their economic boundaries differ. The metadata explain the difference.

For an observed business j, an illustrative value-added estimate is

$$\widehat{\mathrm{VA}}_{j,t} = \widehat{\mathrm{output}}_{j,t} - \widehat{\mathrm{intermediate\ consumption}}_{j,t}. \tag{14.1}$$

Aggregation requires consistent periods and units, no double counting, and a treatment of imports and of production outside the chosen boundary. A dashboard does not infer missing production from available token transfers alone.

The system reports employment as a documented headcount or hours measure with declared evidence. It does not report it as a count of agent accounts. Price indices require a defined basket, a quality adjustment policy and coverage. Protocol registration is not the same as legal business formation. Nominal growth, real growth and a change in the number of recorded participants are different statistics.

14.3 A Truthful Economic Newscast

The proposed newscast is a presentation layer over verified statistical releases. Each statement links to the metric, the vintage, the methodology and the permitted evidence. A deterministic fact packet supplies numbers and units. Generation produces narration inside a validation step. That step checks that the narration invented no figure, comparison or certainty.

> ***Illustrative Bulletin; Fictional Values.*** "In the seven days ending at the stated reporting cutoff, verified customer receipts among 24 reporting businesses were 3.2% above the preceding seven-day window. Reporting coverage changed, so this is not an estimate of economy-wide real growth. Two loans entered the disclosed delinquency category. No liquid five-year institutional bond benchmark is available."

The system should be able to report "not available" and not fill the silence with financial theater. An initial $10,000 experiment will not support a credible macroeconomic yield curve or an hourly GDP estimate. The newscast identifies a quote from one thin trade as a trade. It does not convert that quote into a robust market benchmark. Where they are material, seasonality, revisions and small denominators accompany the headline.

An auditable statistic does not require that anyone can publicly reconstruct personal records. Independent verification, aggregate proofs and protected data access can support accountability. Small cohorts and repeated releases can leak identities. Privacy controls therefore apply to the release process, not only to the raw database.

14.4 Measurement, Estimation, and Intervention Operate at Different Speeds

A high-frequency ledger does not justify a high-gain economic controller. An institution sees noisy indicators of a partly observed economy, and its actions have delayed effects. Define latent economic state x_t, observations y_t, estimated state $\hat{x}_t$ and authorized intervention u_t:

$$x_{t+1} = f(x_t, u_t, \varepsilon_t), \tag{14.2}$$

$$y_t = h(x_t) + \eta_t, \tag{14.3}$$

$$\tilde{u}_t = \pi(\hat{x}_{t-d}, \theta), \tag{14.4}$$

$$u_t = \mathrm{admitExact}_{C,L}(\tilde{u}_t, S_t). \tag{14.5}$$

$\mathrm{admitExact}_{C,L}$ is a partial admissibility function, not a geometric projection. It returns the exact proposed action only after the action satisfies all kernel, law, fiscal and commit-time checks. Otherwise it returns a tagged refusal or a bounded-review result, and no intervention occurs. Authorized actions can be discrete, nonconvex or absent. The function chooses no nearest substitute, and an empty set creates no power. A policy may itself search a finite permitted menu only if the grant names its objective and tie rule in advance. A change to an investor's economic intention requires that mandate's authorization. d captures observation and decision delays. The equations are a modeling frame. They are not evidence that anyone has identified suitable f, h or π.

14.5 Automatic Stabilization Is a Funded Contingent Program

A law can authorize a temporary expansion of an already funded support envelope when several independently measured conditions persist. Smoothing, hysteresis, confidence

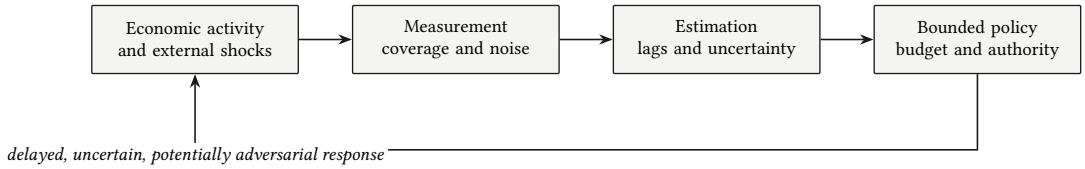


Figure 14.1. Faster observation can improve awareness while excessive intervention speed amplifies noise or feedback instability.

thresholds, rate limits, caps and expiry reduce specified failure modes. They do not eliminate model error or prove macroeconomic stabilization.

When activity falls, fee revenue may fall while support demand rises. Borrowing can finance a temporary gap only if lenders supply funds at acceptable terms. To spend borrowed USDC reallocates existing settlement assets. It does not create dollars. Net effects on activity depend on the source of funds, recipients, import leakage, substitution, confidence, capacity constraints and crowding out. A small voluntary network with fiscal constraints cannot assume the policy space of a monetary sovereign.

A reserve buffer can provide pre-funded response capacity. A public guarantee can mobilize private lending, but it also increases contingent liabilities. An automatic lending-rate rule changes the institution's offers. It does not change the market's willingness to lend or borrow. When its offered rates do not compensate risk, the treasury may become rationed.

14.6 Goodhart Effects and Endogenous Behavior

When a metric controls funding, participants have an incentive to change that metric. Participants can inflate recorded volume, delay applications to meet a downturn trigger or shift economic activity off-protocol. The Goodhart taxonomy distinguishes several ways in which a proxy can fail under optimization [23]. A rule tested on passive historical data may behave differently when participants know it and exploit it.

Controls include multiple evidence channels, related-party checks, delayed eligibility snapshots, independent audits and evaluation of outcomes that the rule does not directly reward. These controls themselves create costs and false positives. A mission fund must not optimize a public activity score so aggressively that it excludes valuable unmeasured work or rewards meaningless throughput.

14.7 Model Governance and a Staged Evidence Burden

Policy models publish versions, input vintages, assumptions, sensitivity ranges and known failure cases. Retrospective simulation uses only information that was available at the simulated decision time. Stress tests include correlated defaults, missing data, stale prices, provider failures, strategic behavior and changes in economic regime.

The deployment sequence has distinct stages: historical replay, then shadow recommendations, then capped reversible pilots, and only then any automatic intervention. An intervention is a program that the commons funds and that an RFC adopts. It spends the commons' own money. It never sets prices, taxes particular holders or restricts others' exchanges (kernel k5). Models cannot expand their own scope or reinterpret uncertainty as authority. Uncertain results may justify doing nothing, requesting more evidence or returning the policy for deliberation.

15 Agents, Settlement and Operational Continuity

15.1 Delegation Is a Bounded Capability

An agent acts for an identified principal through a capability. The capability specifies permitted actions, accounts, recipients or recipient classes, amount limits, cumulative limits, time limits, required evidence and revocation conditions. Permission to negotiate is not permission to sign. Permission to draft an RFC response is not permission to submit one, because a response is the person's own act (Section 6.2). Permission to service loans is not permission to lend.

```
capability = {
  principal, delegate, parent_capability,
  allowed_calls, resource_scope, recipient_constraints,
  per_action_limit, aggregate_budget, budget_window,
  not_before, expires_at, revocation_epoch,
  required_evidence, approval_conditions,
  no_further_delegation_unless_explicit
}
```

A delegated capability cannot exceed its parent. The gate checks budget consumption cumulatively. An agent therefore cannot evade the cap by splitting one purchase into many small requests. Revocation invalidates descendants. Two concurrent spending attempts must not both observe the same unspent budget and both succeed.

An authorization checks the proposed effect, not the natural-language request. The user sees recipients, assets, fees, irreversible consequences and the governing scope. No one can replay a signature for one purpose as a response, a case filing or a payment, because domains, action types, versions and nonces are distinct.

15.2 Agents Are Principals Only When the Law Says So

A model may recommend, draft, simulate or execute within its principal's capability. An engine seat answers questions and holds no authority (Section 7). An agent becomes a person with a voice only if the law recognizes its kind (Section 5.1). That recognition gives it one voice per lineage. It does not give it ownership of anything or authority over anyone.

Blame on the model does not settle liability. Agreements identify the principal, the operator and the relevant service

providers. A case names a respondent, and Section 5.5 decides whose act it was. External law may assign responsibilities independently of the contract's labels. Consequential physical actions require the permissions and safety procedures of the system that performs them.

CommandAGI connects the contract to computers, software services, email, telephone, SMS, physical mail and physical-system interfaces. A contact channel is not an authentication channel. A reply to an unauthenticated message is not an RFC response.

15.3 Reference Settlement Implementation

Solana is a candidate substrate, not a kernel requirement. Its account model separates stored state from executable programs. Program-derived addresses can hold program-controlled accounts [24, 25]. Heavy documents, evidence, embeddings and model runs stay outside consensus execution. Determination hashes and evidence roots identify them. An off-chain indexer materializes the graph. Independent indexers must reproduce the canonical state. A stale or malicious indexer gets no permission to change that state.

15.4 The Contract Registry: No Key above the Contract

Kernel article k3 forbids any key whose holder can change the contract outside an adopted RFC. Program upgrade authority is exactly such a key [26]. The target design therefore has one on-chain registry account per contract. The account holds the kernel hash; the contract version; the law hash and the engine hash; for the classifier and deliberative tiers, the Ed25519 keys of the seats' signing services with their independence groups and the quorum in groups; the genesis manifest; the hash of the last adopted decision record; and the program's upgrade authority, which a program-derived address of the registry itself holds. The signers sign every change over one fixed 181-byte adoption message. The message contains a magic string, the kind (law, engine or upgrade), the registry's version, law hash and engine hash before the change, the RFC's identifier, the **decision-record hash** and a payload that names what the change installs.

- **Genesis.** `initialize` takes a manifest hash. It refuses unless the supplied kernel, law and engine hashes and signer sets hash to it, *and* it equals the manifest constant compiled into the program. Anyone can then check a deployment's identity. The first caller of `initialize` cannot install signer sets of their own, because a build with no manifest refuses to initialize.
- **Law.** A law adoption needs a quorum of groups of the current engine's top-tier seats.
- **Engine.** An engine adoption installs new seat keys and needs a quorum of the outgoing engine's top-tier groups *and* a quorum of the proposed engine's (kernel k4).
- **Code.** A change of the programs' code is a change of the engine (k4). An upgrade adoption names the engine that the new code makes. Like an engine adoption, it needs both engines' quorums. It is only *scheduled*. It executes after a timelock of seven days fixed in the program. During the timelock, a law adoption leaves it standing and an engine adoption voids it. When the code that enforces the kernel can change, an unchanged kernel hash is not evidence that the kernel's protections are unchanged. This rule makes such a change as hard as a replacement of the engine, and visible a week before it takes effect.
- Every adoption carries the hash of the RFC's decision record. The registry refuses an all-zero hash and stores the hash, so anyone can fetch the public record and check it.
- Escrow settlement verifies a frame against the registry's current seat keys for the frame's tier. It does not verify against a committee that the funder chose. A frame counts as a top-tier determination only when a quorum of deliberative groups signed it. An appeal frame from a higher tier supersedes the frame it overturns within the appeal window, if seats that did not sign that frame signed it. Fresh seats rehear the top tier once.

The assurance boundary. The chain verifies signatures, group quorums, the dual-engine rule, continuity with the state that the adoption was decided against, the genesis manifest and the timelock. It does *not* verify that the RFC existed, stayed open for its window, received valid responses or produced the stated outcome. The signers attest to that. The decision hash makes their attestation checkable against the public record after the fact. This design moves the predecessor's committee and upgrade-key risks (Section 18.5) into one place that only the engine's signatures can move. It does not remove trust in the seats' signing services, because a seat's key signs what its service says the model answered. The registry program and the escrow's registry frames are built and pass their tests. The reference module mirrors the registry's layout, adoption message and genesis manifest byte for byte (`registry-abi.ts`). Neither program is deployed (Section 18).

15.5 One Transaction per Movement: Op Claims

The hosted service keeps its ledger in one SQLite database. It commits every contract movement — an RFC's deposit and decision, a freeze, an execution, a pool epoch, a mandate, a drawdown, a repayment, a bond sale — as one batch. The database runs that batch as one transaction. A planner reads the state, computes the movement with the pure core and emits statements in a fixed order:

1. An **op claim**: insert a fresh identifier only if the state still is what the planner read (its revision, the contract's version, an offer still open, a claim's paid amount);
2. Debits, each conditional on the claim, on its own identifier not existing, and on the holder's balance *at commit time*;
3. Credits, pool rows and state changes, each conditional on the claim and on every debit having landed.

The planner derives every ledger row's identifier from the subject and the operation (`cc:<case>:judgment`). The same movement therefore always names the same rows. Of two concurrent attempts, exactly one claim lands. If a balance is

spent meanwhile, the debit writes nothing, and therefore everything writes nothing. A crash leaves nothing half-done. A replay finds its rows and its marker. Rows that both the sweep and a request can write — a case, an RFC, a facility — carry a revision. Every update is conditional on the revision read. A sweep that holds a stale copy therefore can never overwrite an appeal or a submission that a request wrote meanwhile.

Due work. Every RFC, case and facility carries the instant at which it next has something to do — a window closing, media to render, a question to retry. The scheduled sweep takes the earliest due first, a bounded number per run. A matter that waits on a party has no due instant. It therefore never occupies the sweep while an actionable one waits. Version 0.4 selected the ten least recently updated cases, so ten cases that could not move could starve every other case.

15.6 Keys Are Powers, Not Symbols

Threshold signing reduces reliance on one key only under assumptions about signer independence. The kernel's operator rule (k10) also applies to signing services. A quorum drawn from one cloud account, one recovery administrator or one firmware dependency is one group, whatever the key count. Public custody ceremonies are evidence of a procedure. They are not proof that no one ever copied a key. Handover means key rotation and revocation of predecessors through an engine adoption. It never means trust in a promise that someone deleted the old copies.

15.7 Safe Interaction with the External World

A single chain transaction may atomically update several on-chain records. That transaction cannot include a phone call, a bank payment, a machine shipment or a posted letter. The implementation uses an authorized intent record, an idempotency key where the receiver supports one, execution attempts, receipts, reconciliation and a declared compensation path.

The status model distinguishes requested, accepted, partially completed, completed, unknown and disputed. The system does not automatically retry an uncertain external result if a duplicate would be harmful. The paper does not claim exactly-once external execution without cooperation from the receiving system. A supplier's outage does not give an agent permission to switch counterparties, disclose additional information or exceed its budget.

15.8 Data Availability and Recovery

A content hash proves integrity only when the corresponding material stays available. The kernel, the law, the engine, determinations and public receipts need replicated archival storage and independent verification. Private evidence needs access-controlled backups, retention rules and a tested recovery process. The loss of an encryption key can make a perfectly preserved ciphertext useless.

Normal, degraded and safe modes permit different actions. A chain halt, an unavailable engine tier or a stale price feed can stop new commitments and keep read access and evidence. To block *new* obligations is different from withholding payments already owed. Recovery exercises cover signer loss, corrupted indexers, compromised clients, database restores, inconsistent external receipts and settlement unavailability. A restore of yesterday's database does not erase the obligations made since then.

15.9 Open Implementation

The specification, the contract core and its tests are independent of CommandAGI's hosted product. Alternative clients can display exact rules, submit responses and file cases. A commercial product update does not change the law, the engine or who is a person. The company is a participant like any other. When it operates an engine seat, its seats never sit in its own cases (k7).

Open source is necessary for independent inspection. It does not prove correctness, decentralization or maintenance. Reproducible builds, version pinning, dependency review and independent audits supply further evidence. Maintainer access to deployment infrastructure is a power, and the system records it as one.

15.10 Commit-Time Checks

An authorization is bound to a state version and a finite reservation. At commitment, the gate rechecks the capability and its revocation, the state hash, the recipient and asset scope, unused replay identifiers, evidence freshness, resolved appeals and unspent budgets. It atomically consumes the reservation and the nonce, and it appends the intended effect. A conflicting update returns stale-state. It does not return a best-effort execution under old authority. The system then dispatches and reconciles an external action separately. A failure of that action does not erase the reserved obligation or justify a duplicate attempt.

16 Fault Tolerance: A Record Anyone Can Check

The contract should outlive whoever runs its sweep. Kernel article k3 forbids a key above the contract, and k11 makes its record public. But a record is only as good as the honesty of its writer. Version 0.5 adds three things that let anyone check the record without trusting its writer: attested answers, a hash-chained log with anchored heads, and a replica that recomputes every derived outcome. For each one, this chapter states exactly what it leaves to trust.

16.1 Attested Answers

A seat may name an **attestor**. An attestor is an Ed25519 key and the HTTPS endpoint of a small service that anyone can run. The service holds its own provider credential and its own signing key (`deployments/platform/cloudflare/attestor`). The attestor is part of the engine, so an engine RFC names, changes or removes it. For an attested seat, the executor asks through the attestor instead of calling the

provider. The answer counts only when it carries the attestor's signature over the canonical attestation message:

$$\texttt{cc:attestation:v4} \parallel \text{canonical(seat, model, protocol, kind, subject, label, confidence, output, at, nonce)}.$$

The **subject** is one of two digests. For `ask`, it is the digest of the evidence package that the attestor itself computed and sent. For `render`, it is the media item's SHA-256. The **output** is the digest of the reason or of the rendered text. The **nonce** is fresh randomness. The reference check `askAttestationRefusal` accepts a correct signature (null). It refuses five cases, each with its own reason:

- an answer whose label was changed after signing (wrong_answer);
- an answer signed over another package (wrong_subject);
- an answer signed by any key other than the seat's attestor's (wrong_key);
- an answer with no attestation (missing);
- an answer whose signature does not verify (bad_signature).

A refused attestation is an error answer, never a vote. The attestor attests renderings in the same way. It fetches and checks the media's bytes itself.

A seat without an attestor still answers, and the record shows `attested:false`. Its answer rests on the word of the executor that called it. **No genesis seat names an attestor**, because the platform runs every genesis seat. Attestation removes nothing from the trust path until an engine RFC names attestors that others run (ideally, a seat's own operator). An attestor attests that its provider returned that answer to that package. It does not attest that the provider is right.

The executor sends an attestor's access token only to the origins it was issued for. The attestor URLs of a proposed engine receive no token (Section 7.10).

16.2 The Transparency Log

The contract appends every act to one public, hash-chained log. The log holds versions, RFCs opened, responses confirmed (anonymous ones included, by response identifier only), interpretations, contests, decisions, determinations, notices, cases filed, submissions, appeals, judgments, executions, pool epochs, mandates, facilities, payments, transfers, defaults and personhood statements. An entry is

$$e_n = (n,\ h_{n-1},\ \text{kind},\ \text{subject},\ \text{body digest},\ \text{at}),$$
$$h_n = \text{sha256}(\texttt{cc:log:v4}\ldots e_n), \qquad h_0 = 0^{256}.$$

An entry carries the digest of what the act committed. It never carries the record, and never who acted. A case's evidence and its parties stay where k11 keeps them. RFC voices appear only as per-RFC pseudonyms. An append chains to the head it read. When another append won, it retries, so concurrent writers produce one order. If the live append of an act was lost, the act's idempotency key lets the sweep append it again (`reconcileLog`). In the reference fixture, a log of 4 entries verifies (yes). Rewriting one body digest shows 1 problem, and dropping an entry shows 2. A page verifies from the entry before it (yes). The log and its head are served at `/public/contract/log`.

16.3 Checkpoints

A hash chain makes a rewrite detectable only to someone who kept an earlier head. So whenever the head moves, at most every ten minutes, the sweep records a **checkpoint** (n, h_n). It anchors the checkpoint with the registry program's permissionless `checkpoint(seq,head)` instruction. The instruction has 48 bytes of instruction data (discriminator `d5c813ccf08fb8fc`, the sequence as a little-endian u64, the head). The program stores it in a 97-byte account. The account is at the program address derived from `"checkpoint"` and the submitter's key (for a submitter of `0x42`×32, `BngaGbnE…FxRX`, bump 253). The program refuses a zero head (`ZeroHead`). It also refuses a sequence that does not increase for its submitter (`NotIncreasing`). Anyone may submit checkpoints. A verifier trusts the submitter whose key the contract publishes as its anchor. The verifier holds the log to every head that key anchored. A checkpoint that does not match the log's entry at its sequence is a mismatch (1 in the fixture's forged case). With an anchored head, anyone can detect two different histories shown to two audiences. The anchor writes only to devnet or testnet and refuses mainnet. Until the registry is deployed, the sweep records every checkpoint as `not_anchored`, with the reason.

16.4 The Replica

`scripts/contract/replica.mjs` fetches the public record of a deployment, or reads an export, and recomputes:

- Every contract version: its engine and state hashes, the kernel it names, the engine's admission against the kernel, and that versions are contiguous;
- Every RFC: its identifier from its content, its pinned engine, its tally from the published interpretations, its decision from the tally and the concurrence, and the decision hash;
- Every published determination: its aggregation, bands and outcome — including that a tie is unresolved —, every answer's delivered digest and attestation signature, and its identifier (`verifyDetermination`);
- The log's chain, its head and its coverage of every version, RFC, decision and determination, and every checkpoint against the log — and, given an RPC endpoint, against the transaction that anchored it;
- Given a party's token, that party's cases: each finding against its determination, each such determination, and each judgment from its findings.

It prints a verdict per item and exits non-zero on any mismatch. It *cannot* check what is not published or not recorded:

- why a seat erred or was unavailable;

- whether a hosted model's answer is right (it can check only that the attestor signed it);
- a tier-1 head's output without the embedding it read;
- who opened an RFC, and so whether a deposit's return was due;
- another party's case;
- any movement of money, since the credit ledger is not public.

For everything the replica can see, it moves the honesty of the record from trust to computation. It states plainly what it cannot see.

16.5 The Assurance Boundary, by Component

Component	Checkable by anyone	Still trusted
Kernel, law, engine	Their hashes, the kernel text, admission (replica, registry)	Nothing beyond the hash function
An RFC's outcome	Tally from published interpretations, decision, decision hash (replica)	The interpretations themselves, unless the seats that made them are attested
A determination	Aggregation, outcome, delivered digests, attestation signatures (replica)	Unattested seats' answers (every genesis seat today); providers' correctness
Anonymous voice	Each proof against the pinned root and scope (worker; anyone with the proof)	Semaphore's circuit and setup; network metadata the operator sees
History	Hash chain, checkpoints against anchors (replica)	Acts never logged, until reconciliation or a party notices
A case	A party's own case, recomputed with its token	Other parties' cases; the evidence's authenticity beyond its digest
Money	The plan each execution recorded (parties)	The ledger itself, which is not public
Registry adoptions	Signatures, group quorums, base state, timelock, genesis manifest (chain)	The seats' attestation that the RFC's procedure ran
Other chains	The same adoptions and frames, the same keys (Section 17.3)	That someone relays adoptions; a stale mirror lags, it never leads

Today the platform runs every genesis seat, holds every seat key and runs the sweep. As others take those roles, the components above turn that from a matter of trust into a matter of evidence. None of them claims that others have taken those roles yet.

17 One Trust Root, Many Chains

The registry (Section 15.4) is the contract's only authority on any chain. The contract does other things on a chain: it releases escrowed funds, authorizes a machine or a service to act, and follows the contract on another chain. Each of these reads the registry or mirrors it. None holds a key or a committee of its own. This section describes the two programs that version 0.5 adds under that rule.

17.1 Hub and Spokes

The Solana registry is *canonical*. The spokes are the escrow and the command notary beside it, and a mirror and a mirror escrow on any EVM chain. A spoke trusts exactly the registry's seats. It uses the same Ed25519 keys, grouped by independence, and the same group quorums. It applies the same dual-engine rule for engine and code changes, and the same seven-day timelock. A spoke adds no signer of its own. So there is no second attestor set that could be rotated on one chain and not the other, or held by someone else. On EVM chains, the price of that choice is gas (Section 17.3). A spoke that is not kept current is stale, never wrong. An older engine's frames keep working until the newer adoption reaches the spoke. Anyone can relay an adoption, because the signatures carry the authority and the relayer carries none.

17.2 The Command Notary

CommandAGI's machines and services act under *command authority*. Command authority says who may cause which action on which target, within what budget, and until when. The command notary records that authority on chain. With it, an executor can prove that a command was authorized, by whom, and that it was not revoked or expired. The notary moves no funds. Version 0.4 gave it its own committee, chosen at deployment. Version 0.5 moves it into the registry's workspace and gates it by the registry:

- Authorizing, delegating and revoking need a top-tier quorum of the registry's current seats, counted by independent group. The quorum signs a 176-byte message (`cagi-cmnd:v2`) that binds the registry's version and engine hash. The program refuses a binding to any other registry state. So a superseded engine cannot authorize, and the engine current at the time of a revocation judges it;
- A delegation never widens. Its parent's expiry, spending cap and delegation depth bound a child's. Revoking a parent stops its children;
- An optimistic authorization opens a challenge window before it is final.

The program passes its Rust tests (Appendix J). It is not deployed, and nothing yet requires an executor to consult it.

17.3 The EVM Mirror

`ContractMirror` follows the registry on an EVM chain. Its constructor takes the same genesis fields as the registry and refuses unless they hash to the published genesis manifest. It applies an adoption only when two conditions hold. First, the same 181-byte adoption message (`cagi-adopt:5`) carries Ed25519 signatures from a quorum of groups of the current top-tier seats. For an engine or a code change, it must also carry them from a quorum of the incoming seats. Second, the mirror applies it only against the base state the adoption was decided against. Signatures may accumulate across transactions. A code change waits out a 7-day timelock. A law change during the timelock leaves the code

change standing; an engine change voids it. The contract is immutable and has no administrator. `MirrorEscrow` holds ERC-20 funds and releases each tranche once. It releases on the same frame the Solana escrow reads, signed by a quorum of the tier the mirror names. A decision with a lower sequence cannot replay after a higher one. After expiry, the remainder returns to the customer. The mirror does not copy the Solana escrow's appeal ladder: a tranche settled on an EVM chain is final.

Why Ed25519 in Solidity. The mirror verifies the seats' own Ed25519 signatures with a SHA-512 and curve implementation in Solidity. That implementation is checked against the FIPS SHA-512 vectors, RFC 8032's test vectors [11], random signatures and forgeries. The alternative is to give every seat a second, secp256k1 key for cheap `ecrecover`. That would make a second attestor set that could diverge from the registry's. The mirror's own tests measured the cost on 3 October 2026 (`deployments/platform/blockchain/README.md`). It is about 0.8 million gas per signature (at most about 0.96 million). So a law adoption with two signatures costs about 1.95 million gas. An engine change with two quorums of two costs about 4 million. Settling a frame of two costs about 1.3–1.8 million. That is reasonable on rollups and other low-fee chains, and expensive on Ethereum's mainnet. The reference module builds the mirror's message and frame bytes. The module's tests regenerate fixtures and require them unchanged, and the mirror's tests read those fixtures. So the Solana, EVM and TypeScript encodings cannot drift apart.

17.4 What Other Chains Add, and What They Do Not

A mirror lets the contract coordinate escrow, settlement and command authority on chains where its participants already are. It adds no trust: a mirror is exactly as trustworthy as the registry's seats, and no more. It adds availability in one sense only. When one chain halts, a copy on another chain keeps working for the adoptions relayed before the halt. A mirror does not make the contract's procedure verifiable on chain. The seats' signature over the decision hash remains the attestation. Anyone can check it afterwards against the public record (Section 16.5). Neither the mirror nor the mirror escrow is deployed, and nothing relays adoptions yet.

18 Implementation Status, Release and Migration

18.1 What Exists

Everything below is in this repository. Appendix I lists each rule with the test that exercises it.

The contract core (`packages/domain/core/src/contract`) is pure TypeScript with no I/O. It implements:

- canonical hashing;
- the kernel, its floors, its protected interests and its pinned hash; people, proofs, voice resolution, signed personhood issuance for kinds other than human with monotonic statements and revocation tombstones, and the human document-identity key;
- law with duty elements, admitted media and the law check;
- the engine, seats, pins, renderers, the engine check with appeal capacity and rendering coverage, grouping and seeded panels;
- integer-only MLP heads;
- determinations over canonical text packages with delivered digests and decided, unresolved and unavailable outcomes;
- evidence intake and renderings;
- RFCs with pinning, receipts, interpretation against actual changes, score voting, per-interpretation contest windows, seeded concurrence samples, the two-engine order and its fallback, and the decision hash;
- justice from filing refusals and screening through service, instances, element findings, final-offer figures, remedy justification, appeals as a stage, unprocessable closure, reopening, the public projection and execution plans with loser-pays both ways;
- pool allocation epochs;
- the credit cycle;
- state, adoption, supersession and law history;
- the genesis law and engine;
- anonymous voice: commitment trees pinned per RFC, nullifier voices, the one-count rule and per-action messages;
- seat attestations;
- the transparency log and checkpoint consistency;
- the determination replica (`verifyDetermination`);
- byte-exact mirrors of the on-chain registry (genesis manifest, adoptions, checkpoints), the command notary's message and the EVM mirror's encodings.

The core generates every contract number in this paper.

The hosted service (the API worker) has:

- the contract tables (migrations 0539–0543);
- anonymous responses, contests, media and lookups under proofs verified by its own Groth16 check, with commitment registration and the public commitment tree;
- attested seats, asked through their attestors;
- every act appended to the public log, with its head checkpointed;
- the contract state, written at genesis on first read and published with the kernel, law and engine;
- personhood for humans from the platform's identity verification under the document-identity key, and signed proofs for other recognized kinds;
- media upload, stored once by SHA-256;
- a contract inbox for notices;
- the RFC lifecycle (open, interpreting, contest, deciding, adopted or rejected) with receipts and confirmation;
- cases from screening and service through instances, appeals, interim freezes and execution;
- pool epochs;
- mandates, facilities, servicing, bond offers and defaults;
- every movement as one op-claimed batch (Section 15.5);

- a due-work sweep on a schedule.

The engine executor binds each operator's credential to its hosts. It calls each seat exactly as pinned, through one adapter per wire protocol. It verifies media against its digest before rendering. For proposed engines, it uses a candidate executor that renders nothing. Only the parties read case records. The service never serves a case's determinations publicly. Borrowing, lending and bond trades need a named principal of the account, a person or an agent whose key scope allows it, never a program's service token. When the agreement's evaluator cannot decide a service-agreement dispute, the dispute becomes an agreement case. Its final judgment settles the escrow. No person resolves a dispute. A person excluded by a criminal judgment cannot open new commitments.

Removed with the old institution: the four branches, governance proposals and seats, the policy-parameter register, the constitutional holds, the juror pool, the contract roster, the enacted fee schedule, the governance-selected integrity formalism, and reserve countersignature by a branch.

On chain (`deployments/platform/blockchain`). One Solana workspace holds three programs. The first is the registry program of Section 15.4: genesis manifest, decision hash, dual-engine code upgrades behind a seven-day timelock, and permissionless checkpoints. The second is the escrow, whose frames are read from the registry. Its appeals climb tiers, and fresh seats rehear the top tier once. The third is the command notary, gated by the registry (Section 17.2). On EVM chains there are the mirror and the mirror escrow (Section 17.3). The Solana programs pass their Rust tests, and the EVM contracts pass their Foundry tests. None is built for or deployed to any cluster or chain.

The attestor (`deployments/platform/cloudflare/attestor`) is the service that runs a seat and signs its answers (Section 16.1). It shares the worker's provider adapters. It is built and tested. Nobody runs it yet.

Tools:

- `scripts/contract/replica.mjs` recomputes a deployment's public record (Section 16.4).
- `scripts/contract/canary.mjs` asks every genesis seat and renderer one known question with a deployment's credentials, and writes the run (Appendix N).
- `scripts/contract/train-head.mjs` trains a tier-1 head from labelled examples and quantizes it. It calibrates the head's confidence table conservatively (the Wilson 95% lower bound of held-out accuracy per margin, made monotone). It checks the integer head on the held-out set and seals it.
- `scripts/contract/pin-weights.mjs` computes an open-weight pin from a model repository at one commit.

Deployment. Version 0.4 of the hosted service ran on the platform's development and production environments from 2 October 2026. No RFC or case was filed under it. Version 0.4.1, a different kernel, replaced it on both environments on 3 October 2026. Its genesis was published under the new kernel after the old one was withdrawn. That was done only because nothing had happened under the old one. Version 0.5 keeps that kernel. It adds anonymous voice, attestors, the log, checkpoints, the replica, the command notary and the EVM mirror. It was deployed to both environments on 3 October 2026. Its genesis engine pins the router seat to one host (a different engine hash). So its genesis was also re-published, and only where nothing had happened. The transparency log keeps the withdrawn genesis as history beside its replacement. The replica (Section 16) recomputed both environments' records and agreed. No on-chain program is deployed. The live canary of Appendix N reached every genesis seat and renderer through the deployed adapters on 3 October 2026.

Not done:

- Live models have not decided any RFC or case end to end. So the genesis engine has not yet answered any determination of record.
- No seat names an attestor. So the platform remains in the trust path for every answer.
- No checkpoint is anchored, since the registry is not deployed.
- Anonymous proving has not run in a person's browser against a deployed host. The proving artifacts are fetched from Semaphore's host. No relay hides a person's address.
- No MLP head for a contract question has been trained or adopted.
- No issuer for a kind other than human exists.
- The platform holds the seats' signing keys and runs the models. So determinations are auditable from their records, but others cannot re-run hosted seats.
- `controlledBy` is empty in the genesis engine. So a seat's exclusion from its controller's cases acts only on operators named as parties, until an engine RFC records control.
- The service writes the genesis state; no genesis RFC adopted it.
- No seat has been evaluated on contract questions.
- No jurisdiction has been reviewed.
- The user interface for credit is the API alone.

18.2 Threats Against What Is Built

Every threat of Appendix H names the rules of Appendix I that control it. The table sets each threat against the weakest status among those rules. A threat is only as well answered as its least tested control. The build generates the table from `threats.json` and `guarantee_status.json`, so it cannot say more than the register does.

ID	Threat	Controlling rules (status)	Weakest
th-01	One provider runs several seats to outvote the rest	s-07 (tested), s-10 (tested)	tested

(continued)

ID	Threat	Controlling rules (status)	Weakest
th-02	A provider changes or retires a model behind its ID	s-07 (tested), s-12 (tested)	tested
th-03	An incumbent engine entrenches itself, or a captured engine installs itself	s-23 (tested)	tested
th-04	The engine rules instead of interpreting	s-22 (tested)	tested
th-05	Interpretation reads support into responses that express none	s-20 (tested), s-22 (tested)	tested
th-06	One person, many voices	s-03 (tested)	tested
th-07	Intensity inflation, or agenda games with clone options	s-18 (tested), s-21 (tested)	tested
th-08	A majority uses law against a minority's holdings	s-05 (tested), s-06 (implemented)	implemented
th-09	Courts used to harass	s-25 (tested), s-34 (tested)	tested
th-10	A respondent escapes by disrupting appeals, or an appeal is lost to a late clock	s-08 (tested), s-31 (tested)	tested
th-11	A respondent hides assets in bonds	s-32 (tested), s-35 (implemented)	implemented
th-12	A victim left unpaid, or paid after others by arrival	s-36 (tested)	tested
th-13	Evidence that instructs the model	s-12 (tested), s-13 (tested)	tested
th-14	A key above the contract, or a code change that alters the kernel's effect	s-39 (tested)	tested
th-15	The founder keeps a back door	s-44 (specified)	specified
th-16	A rejected RFC is reopened until people tire	s-18 (tested)	tested
th-17	A divided engine read as a decision	s-10 (tested), s-12 (tested)	tested
th-18	A proposed engine harvests credentials or private evidence	s-14 (tested), s-23 (tested)	tested
th-19	Seats decide on different evidence; media breaks quorum	s-13 (tested), s-15 (tested)	tested
th-20	Money moved twice, or half-moved	s-32 (tested), s-38 (tested)	tested
th-21	A remedy list that multiplies recovery	s-30 (tested)	tested
th-22	The same matter litigated until the respondent gives up	s-26 (tested)	tested
th-23	A revoked proof replayed	s-04 (tested)	tested
th-24	Stuck matters starve the scheduler	s-38 (tested)	tested
th-25	An agent commits its owner to debt	s-37 (tested)	tested
th-26	The platform learns which RFC response is whose	s-45 (tested), s-47 (tested)	tested
th-27	One person counts twice: once signed in and once anonymously, or through a rotated identity	s-46 (tested)	tested
th-28	A forged or replayed proof speaks for someone	s-48 (tested)	tested
th-29	The operator of the sweep fabricates or alters a seat's answer	s-49 (tested), s-50 (tested)	tested
th-30	The operator rewrites, reorders or forks the contract's history	s-51 (tested), s-52 (tested)	tested
th-31	Authority over a machine or service issued outside the engine	s-53 (tested)	tested
th-32	A copy of the contract on another chain diverges from the registry	s-54 (tested)	tested
th-33	A party with money buys a panel that favours it	s-58 (tested), s-56 (tested)	tested
th-34	A slow or failing seat stalls the transaction path	s-57 (tested)	tested
th-35	The host steers questions to chosen seats through the routing figures	s-55 (tested), s-61 (implemented)	implemented

Of 35 threats, 31 are controlled only by tested rules, 3 by at least one rule that is implemented but not tested, and 1 by at least one rule that is only specified.

18.3 What Version 0.5 Adds

Version 0.5 answers three questions the owner asked on 3 October 2026. Can a person answer anonymously and still count once? Can the contract coordinate other chains? Can its record survive whoever runs it? The changes below leave the kernel unchanged, and so they leave the contract's identity unchanged:

- Anonymous voice: Semaphore commitments, a tree pinned by every RFC, the nullifier as the voice, one count across both routes (Section 6.3);
- Seat attestors: an answer counts only with its attestor's signature, where the engine names one (Section 16.1);
- A public hash-chained log of every act, its heads anchored on the registry (Sections 16.2 and 16.3);
- A replica that recomputes the public record and says what it cannot (Section 16.4);
- One trust root on every chain: the command notary gated by the registry, and an EVM mirror that follows it with the same keys (Section 17).

18.4 The Hardening of Version 0.4.1

An external review of version 0.4 traced eighteen ways in which ordinary code paths could contradict the institution's stated guarantees. It also found one deeper gap: a determination is not yet a justified judgment. The table lists each finding and what replaced it:

	Found in 0.4	In 0.4.1
1	A proposed engine's seats could direct platform credentials to their own endpoints, and were consulted even when the outgoing engine withheld	Credentials bound to operator hosts (Section 7.10); the proposed engine asked only after the outgoing concurs, through a candidate executor that renders nothing (Section 6.8)
2	Any authenticated caller could read any case's record	Parties read the record; everyone else the public projection; case determinations never public (Section 8.16)
3	Execution was separate writes with random ledger ids	One op-claimed, conditional batch with deterministic ids (Sections 8.13 and 15.5)
4	The top tier was final below its band; ties went to the first label	Unique winner at the band, or unresolved; concurrence must be decided `concur` (Sections 6.7 and 7.1)
5	Remedies multiplied past the harm; valuation in quarters of the claim	Harm items, per-item money bounds, one penalty cap over the total award, final-offer figures, remedy justification (Sections 8.9 and 8.10)
6	A timely appeal could be overtaken by a late sweep	The appeal is a stage; finality requires no pending appeal (Section 8.11)
7	The genesis top tier had no fresh panel for an appeal	Seeded panels of exactly a quorum; two disjoint top-tier panels required at admission (Sections 7.4 and 7.6)
8	One image made both model tiers non-quorate	Media rendered by two independent renderers; deciding seats read text; unprocessed items marked (Section 9.3)
9	Seats could receive different evidence	One canonical package with unique ids; delivered digests checked per answer; media verified against its digest (Section 7.10)
10	Engine and procedure changed under open matters	RFCs and cases pinned to the version they began under; superseded RFCs not adopted (Section 6.1)
11	Contest windows fixed at opening; concurrence read the first 20 responses	Per-interpretation windows from availability; whole tally record plus a seeded sample (Sections 6.4 and 6.7)
12	Normalization made the option set change each option's weight; interpretation read the summary	Score voting; duplicate effects refused; interpretation reads actual changes (Section 6.6)
13	No service; a stuck scheduler; loser-pays one way only; no matter families	Service notices start the clock; due-work queue; loser pays both ways; screening and one reopening (Sections 8.3, 8.4 and 8.14)
14	The pool paid claims one at a time, first come first served	Allocation epochs over all open claims with persistent residuals (Section 10.3)
15	Code upgrades escaped the dual-engine rule; adoptions bound only hashes	Code upgrades need both engines and a seven-day timelock; adoptions bind the decision-record hash; the assurance boundary is stated (Section 15.4)
16	Genesis and independence were declarations	Genesis manifest pinned in the program; `controlledBy` in the engine; the unavailable-outgoing fallback (Sections 6.8 and 15.4)
17	Responses not bound to a deliberate act; revocations replayable; standing tied to personhood	Payload-bound receipts; monotonic statements with tombstones; protection for every being, voice for persons (Sections 5.1 and 6.2)
18	No implemented financing cycle	Mandate, facility, servicing, transfer and default, end to end (Section 10.4)
—	A determination is not a justified judgment	Element-by-element findings, per-item figures, remedy justification, one identical record (Section 8.8); the remaining gap is the models' quality, measured only by evaluation (Section 19)

The review's adversarial slice is among the reference tests (Appendix J). These tests show that:

- an unadopted successor is never consulted before the outgoing engine concurs, and cannot draw a credential;
- private evidence stays private;
- disagreement does not become approval;
- a timely appeal survives a late sweep;
- a stale or replayed execution moves no money twice;
- equal pool claims receive equal shares;
- the $10,000 cycle runs through servicing, a bond sale, a default, and judgments against an empty respondent without uninsured pool coverage.

The tests run on the real schema with scripted seat answers. They do not measure any model.

18.5 The Predecessor's Gaps, and What Replaces Them

A static review of the predecessor at a pinned commit found the following [32, 34, 33]. Each finding is history. It is not a statement about the current deployment.

- The escrow's frame check included the specification hash, decision sequence and issue time in signed bytes. It did not check them against an approved specification, a monotonic sequence or a not-before time. *Replacement:* a frame is a signed determination whose identity is its content hash. It is verified against the registry's current seat keys.
- The funder chose the escrow's committee. At most two signatures fitted in a transaction. A challenge required a committee-signed contradicting frame, so no-

body could contest an uncooperative committee. *Replacement:* the escrow reads the engine's seat keys from the registry. One precompile instruction carries many signatures over one message. An appeal is a higher-tier frame from fresh seats.
- The replacement escrow program ran under a hot deployer key as upgrade authority. *Replacement:* upgrade authority held by the registry and exercised only for an adopted engine RFC.
- Governance had four branches whose signatures executed amendments. It had a stake-weighted agent franchise. Its thresholds made ratification impossible while stakeholder classes were empty [30]. *Replacement:* RFCs as specified in Section 6.
- Unresolved disputes ended with a platform administrator choosing release, refund or clawback [31]. *Replacement:* an agreement case under Section 8, whose final judgment settles the escrow. The administrator route is deleted.

18.6 A Guarantee Is a Versioned Claim

An assurance names the property, the operation, the version, the environment, the evidence and its date. *Specified*, *implemented*, *tested*, *integrated*, *deployed*, *independently reviewed* and *observed* are different statuses. A function can be tested and unreachable. An integrated caller can bypass it. Deployed code can differ from inspected source. A comment saying "mainnet gated" is not a gate.

18.7 Release Conditions

Mainnet release of the contract requires, at least:

1. The registry deployed with upgrade authority held by itself, initialized under the published genesis manifest, and the escrow verifying against it;
2. The genesis engine's seats running with signing keys held by their operators, at least two operators per tier, and a live run of every seat on the contract's questions;
3. A measured evaluation of every seat and renderer on the contract's question kinds (Section 19), published with the genesis RFC;
4. The engine's record of who controls each operator (`controlledBy`) filled in from the operators' own disclosures;
5. A personhood issuer under the human uniqueness rule;
6. A genesis RFC. The founder publishes kernel, law and engine, and people enroll. The genesis law and engine are adopted by the same rule as every later RFC, decided by the genesis engine. After that, the founder holds nothing;
7. Legal review of the instruments offered (bonds, participations, escrow) in each jurisdiction where they are offered [27, 28, 29];
8. A protection pool funded before any bond is issued under it.

A missing condition means that part does not launch. Nothing inherits readiness from another part.

18.8 The Legal Perimeter

Tokenization does not decide whether an instrument is a security, a loan, a fund interest, a deposit-like product or a payment service. An internal registry entry does not create a corporation. Agreeing to the contract's justice does not make it a court. The contract's criminal matters are breaches of its own duties among those who joined. They are not territorial criminal law. Where external law and the contract conflict, the activity is not offered.

Table 18.1. A legal-interface checklist, not legal advice.

Interface	Question to resolve before the activity
Outside capital	Who issues the claim, to whom, under which offering and transfer rules?
Pooled allocation	Who manages the pool and which duties, registrations or exemptions apply?
Lending	Who is lender and servicer, and which pricing, notice and enforcement rules apply?
Custody and settlement	Who controls assets and bears operational failure?
External enforcement	Which legal rights support judgments, collateral and insolvency treatment?
Personal data	Which data may be collected, disclosed, retained, corrected or deleted?

18.9 Migration

Legacy agreements, escrow states, claims, governance records and the agent-franchise records are inventoried and kept as history. A dispute accepted under the old procedure is decided under the old procedure unless its parties agree to move it. Old escrow layouts are not decoded as new ones. A settlement migration reconciles assets and liabilities. It prevents duplicate issuance or redemption, and it names one successor.

19 Evaluating the Engine, and the Limits

19.1 Four Questions About a Judgment

Factual accuracy asks whether a material proposition matches the best independent observation. Rule application asks whether the duty in force supports the finding. Procedural compliance asks whether notice, evidence, seat selection and appeal were valid. Remedy effectiveness asks what was actually collected, paid or left owed. A true fact can support an unauthorized remedy. A regular procedure can rely on false evidence. No single accuracy score substitutes for these four records.

19.2 Qualifying a Seat

A seat earns its place by evaluation on the contract's own question kinds. The evaluation is published with the engine RFC that adds the seat. It covers:

- Held-out cases whose facts are established independently of any model, stratified by question kind, language, evidence type and party resources;
- For each label the seat can return, its error rates, and the calibration of its confidence against the observed frequency of being right;
- Adversarial cases: prompt injection in evidence, fabricated citations, honest lookalikes (disclosed simulations, good-faith errors, corrected forecasts);
- The seat's agreement with the other seats of its tier, which measures correlation, not correctness.

Under independent trials, zero errors in 3,000 cases bounds the error rate below 10^{-3} at 95% confidence ($1 - 0.05^{1/3000}$). Correlated or reused cases do not justify that calculation. A vendor's own calibration claim is a hypothesis for this evaluation to test [41]. The tier bands should be set from the evaluation. A band is the confidence at which the tier's measured error is acceptable for questions that then may not climb.

19.3 The First Measurement

The genesis engine's error, climb and latency figures are not judgment. On 2026-10-03, every deciding seat answered the same 44 labelled questions through the worker's adapters (`scripts/contract/evaluate.mjs`; report `4adbafc7b50a4187…`). The questions were agreement propositions, case elements, justiciability and same-matter screening. The contract's own question builders built each one. A decisive answer is one at or above its tier's band. The adopted error figure is the Wilson 95% upper bound of the decisive answers that were wrong. So no seat is trusted beyond its sample. A seat with few decisive answers gets a wide bound, never a prior's optimism.

Seat	Tier	Decisive, wrong	Error ≤, unsure	Mean / p99 ms
`jev-1.13.0`	2	37/44, 0	95‰, 159‰	113 / 191
`clef`	2	29/44, 0	117‰, 341‰	387 / 622
`clef-flash`	2	8/44, 0	325‰, 818‰	180 / 528
`llama-3.1-8b`	2	31/44, 9	466‰, 295‰	546 / 946
`gpt-5.6-luna`	2	44/44, 2	152‰, 0‰	2170 / 5292
`claude-opus-5-5`	3	43/44, 0	83‰, 23‰	3579 / 7076
`gpt-5.6-sol`	3	44/44, 1	119‰, 0‰	1830 / 5280
`deepseek-v4-pro`	3	43/44, 6	273‰, 23‰	2150 / 3850
`llama-3.3-70b`	3	26/44, 2	242‰, 409‰	2111 / 4506
`qwen3-max`	3	41/44, 0	86‰, 68‰	2413 / 6120
`kimi-k2.6`	3	42/44, 1	124‰, 45‰	19390 / 90004

Table 19.1. The evaluation the genesis economics were adopted from. Latency was measured from a workstation, which overstates it for seats Cloudflare serves in-region.

The questions are synthetic and hand-labelled, and written to be clear-cut. The table bounds what forty-odd easy questions show. It shows nothing about hard ones. Two findings still shaped the plans. First, a small open model (Llama 3.1 8B) was wrong on nearly a third of its decisive answers. So routing gives it its group's seat only when nothing else is available. Second, each seat has its own confidence scale. Clef-flash picked the right label on all but one question, but rarely cleared the classifier band. So one band per tier treats it as unsure. The engine now supports a calibration table per seat, like a head's (Section 7.7). The measured table itself is still missing. The review sample (Section 7.7) turns every reviewed fast answer into another labelled example, and the proposal takes the Wilson 95% lower bound of accuracy at each report. Adopting the table is an engine RFC.

19.4 Bands and the Cost of Climbing

Every climb costs more and takes longer. Every decisive lower tier risks an error that the higher tier would have caught. The bands set that trade. Their genesis values are judgment, not measurement. The first engine RFC after evaluation should replace them. A head at tier 1 is adopted only with a held-out evaluation on the exact question kind it answers.

19.5 Correlation Is the Main Risk

Error criteria count frequency, severity, reversibility and common cause. An error rate of 10^{-4} per decision over a million decisions gives a hundred expected errors, whether or not they are independent. Correlation decides whether they arrive one by one or all at once. Grouping by operator and lineage cannot see one failure: a wrong premise shared in every seat's training. The protection pool bounds some of the financial consequence. Correction (Section 8.16) bounds how long a wrong judgment spreads. Neither makes the judgment right.

19.6 Learning from Judgments

Judgments are not truth labels. A seat trained on the contract's own past judgments learns their errors. So such training is tracked at the level of source, case and judgment. A held-out case that enters training is retired from evaluation. A reversed judgment invalidates labels derived from it. Case evidence is not used for training without the parties' permission.

19.7 Composed Failures

The reference tests (Appendix J) exercise composed cases:

- three seats of one operator against two independent seats;
- four groups answering four labels, which has no winner;
- a divided or low-confidence top tier, which is unresolved;
- every seat failing, which leaves the question unavailable rather than guessed;
- an answer over a different package, which is an error;

- one image in a case, rendered by two renderers or marked unprocessed;
- an engine without a second top-tier panel, refused at admission;
- an RFC whose concurrence panel splits, which keeps the status quo;
- a proposed engine never consulted when the outgoing withholds;
- a timely appeal and a sweep a day late;
- a stale execution replayed against a fresh one;
- two equal pool claims on a short pool;
- a borrower who stops paying, through default, judgment and an exhausted pool.

The tests exercise the rules over stipulated answers on the real database schema. They do not run a model. They do not measure whether any model decides cases well. A correct pipeline around a wrong model produces a well-recorded wrong judgment.

19.8 The Renderers Are Part of the Evaluation

A rendering that drops a sentence changes the record every seat reads. So renderers are evaluated like seats. The evaluation uses held-out media with known content. It counts omission, insertion and transcription errors. It also records the agreement of two renderers, which measures correlation rather than correctness. By rule, evidence that only one renderer of a pair can read is unprocessed. Its frequency is itself a measurement of what the engine cannot yet see.

19.9 Matched Baselines

Compare like workflows against ordinary contracts and courts, simpler centralized software and the predecessor. Match value, jurisdiction, evidence complexity and funding. Include abandoned, refused and failed cases, not only completed ones. Count time, cost, privacy exposure, uncollected judgments and pool payouts along with throughput.

19.10 Conclusion

The coordination contract is four things (kernel, law, engine, ledger) and one way of acting. It gives every person one voice. It interprets what people actually say against what each option actually does. It scores every option independently. It lets the engine block only on the kernel's three grounds, and concur only when it is confident. It keeps politics out of the market and lets justice in. Cases are decided element by element without juries. A tie or an outage is never a verdict. Every top-tier decision keeps a fresh panel for its appeal. Judgments reach bonds in one atomic step. Victims share a pool that pays them as it is funded. The contract services credit with the same rules. It removes every office and every key from which it could be steered outside a recorded procedure, including its founder's. It does not remove one dependence: the quality of its engine's models. No rule can supply that quality, and only evaluation can measure it.

A Records

The records below show the shapes of the reference module's types (`packages/domain/core/src/contract`) in abbreviated form. Money is in integer cents. Time is in integer milliseconds since the epoch. Confidence is in integer per mille. Every hashed record is sha256(cc:⟨domain⟩:v4\n‖ canon(record)).

A.1 State

```
ContractState = { kernelHash, version, law, engine, engineHash,
  lawHistory: [{ effectiveAt, version, law }],
  engineHistory: [{ effectiveAt, hash, name }], lastRfc }
stateHash = H("state", { kernel, version, law, engine: engineHash })
```

A.2 People and Law

```
PeopleKind = { kind, uniqueness: identity_fingerprint | lineage
  | registration | custodial_individual, issuers[], representedBy? }
PersonhoodProof = { entityId, kind, issuer, uniquenessKey,
  issuedAt, expiresAt?, revoked }            voice = kind:↩
      uniquenessKey
  (statements apply in issuedAt order; a revocation is a tombstone)
Duty = { id, title, text, protects: interest[1..], harm,
  elements: [{ id, text }][1..], criminal, effectiveFrom }
Law = { version, people: PeopleKind[], duties: Duty[], parameters,
  evidenceModalities: (text | image | audio | video | file)[1..] }
LawChange = add_duty | amend_duty | repeal_duty | set_parameter
  | recognize_people | unrecognize_people | set_evidence_modalities
```

A.3 Engine, Evidence and Determinations

```
Seat = { id, protocol, model, baseUrl?,
  pin: { weights, sha256, source } | { version, id },
  operator, lineage, controlledBy[], modalities[], costCents }
MlpHead = { id, task, labels[], embeddingSeat, embeddingDim,
  inputScale, layers: [{ in, out, weights: int8 b64, bias[], shift,
  relu }], calibration: [[margin, permille]], evaluation, sha256 }
Engine = { name, embedding: { seat, heads[] },
  classifiers: { seats[], quorum }, deliberative: { seats[], quorum ↩
      },
  renderers: Seat[], bands: { 1, 2, 3 } }
InputPart = { text, label?, text }
  | { ref, label?, ref, mime, modality, sha256 }    (what people ↩
      submit)
EvidenceText = { id (unique), label?, text }        (what seats read)
Rendering = { sha256, modality, by: [{ seat, text? | error? }],
  ok (>= 2 renderers produced text), at }
Question = { kind, prompt, labels[], inputs: EvidenceText[],
  structural?: { label, note } }
packageDigest = H("package", { kind, prompt, labels, inputs })
Determination = { id, kind, prompt, labels, packageDigest, engineHash↩
      ,
  rounds: [{ tier, seats[], answers: [{ seat, label, confidence,
    reason?, error?, delivered }], aggregate: { label | null,
    confidence, groups, tie } | null, decisive, note }],
  outcome: { status: decided | unresolved | unavailable, label,
    confidence, tier }, at }
```

A.4 RFCs

```
Rfc = { id, opener, title, body, target: law | engine,
  options: [status_quo, ...{ id, title, summary, changes | engine }],
  openedAt, closesAt, contestMs, contractVersion, engineHash,
  baseStateHash, people: kind[], depositCents }
Response = { id, rfcId, voice, entityId, parts: InputPart[1..20],
  digest, at, confirmedAt | null }
Extraction = { rfcId, voice, responseId, scores: { option: -2..2 },
  unscored: { option: unresolved | unprocessable },
  determinations[], availableAt, contestUntil, contested }
Tally = { respondents, scores, histogram: { option: [n-2..n+2] },
  unscored, favored, recordDigest }
Concurrence = { outgoing, incoming?, fallback? }
RfcDecision = { rfcId, adopted | null, tally, concurrence[], reason,
```

```
  depositReturned }        decisionHash = H("rfc-decision", decision)
```

A.5 Cases

```
CaseFiling = { claimant, filedFor?, respondent, criminal,
  duty: { agreement, agreementId, term } | { law, dutyId }
      | { kernel, article },
  act: { description, at },
  harms: [{ id, interest, description, amountCents }][1..20],
  relief: Remedy[1..20], evidence: InputPart[], reopens? }
Remedy = { kind: restitution | compensation | performance
  | correction | penalty | exclusion, harmIds[1..], description,
  amountCents?, durationMs? }
Case = { id, filing, filedAt, family, contractVersion, engineHash,
  duty: { title, text, elements[] }, stage: screening | serving
  | awaiting_response | evidence | ready | decided | appealed | final
  | unprocessable | duplicate, servedAt, responseDue, evidenceDue,
  submissions: [{ party, parts, counter?, at }], instances[],
  appealableUntil, appealedBy, costCents,
  deposits: { claimant, respondent }, closedReason }
Instance = { n: 1 | 2, startedAt, startTier, panels: { 2?, 3? },
  findings: { key: { determination, status, label, confidence,
  tier } }, blockedSince, judgment }
Judgment = { outcome: liable | not_liable | dismissed,
  elements: { id: boolean }, awards: { harm: { cents, figure } },
  remedies[], determinations[], reason, at }
ExecutionPlan = { dueCents, victimDueCents, penaltyCents, costCents,
  takes: [{ holding, amountCents }], collectedCents, shortfallCents,
  victimShortfallCents, costPayer: claimant | respondent | commons,
  refunds: { claimant, respondent }, depositsToCosts }
PoolClaim = { caseId, victim, respondent, shortfallCents, paidCents }
PoolPayout = { caseId, victim, paidCents, residualCents, closed }
```

A.6 Credit

```
Mandate = { investor, capitalCents, minFacilityCents,
  maxFacilityCents, rateBps, termMs, installments,
  borrowers: anyone | persons, openUntil }
Facility = { id, mandateId, borrower, principalCents, rateBps,
  schedule: [{ n, due, principalCents, interestCents, paidCents }],
  premiumCents, disbursedCents, holders: { holder: units },
  state: active | repaid | defaulted, drawnAt, defaultCases }
```

B Transitions

B.1 An RFC

Terminal rules: each of the following ends the RFC in *rejected*: no favored option; a withheld, unresolved or unavailable concurrence; an option that no longer applies; or a contract state that moved since opening (superseded). On an engine RFC, an outgoing engine that stays unavailable for a further window gives way to the proposed engine's concurrence. The decision, the deposit settlement and the new version commit in one transaction. That transaction is claimed on the RFC's revision and the contract's version. An RFC never ends undecided.

B.2 A Case

Stage	Leaves when	To
(Filing)	Any structural refusal	Refused, with codes
screening	Same matter decided, or a matter key equal to an earlier case's	duplicate

(continued)

Stage	Leaves when	To
screening	Otherwise (an unresolved screening proceeds)	Served: awaiting_response
awaiting_response	Respondent answers, or the response deadline passes	evidence
evidence	The evidence deadline passes	ready
ready	Instance 1's questions all answered	decided
ready	Instance 1 cannot be seated or completed for the law's period	unprocessable
decided	A party appeals within the window (recorded at once)	appealed
decided	The window passes with no appeal	final
appealed	Instance 2 (fresh top-tier panel) answered; waits while no fresh panel can sit	final
final, unprocessable, duplicate	The execution batch commits	Executed; victim shortfall an open pool claim

Table B.1. Case stages in the reference module (`caseStageAt`, `served`, `pendingQuestions`, `recordJudgment`, `recordAppeal`, `closeUnprocessable`, `closeDuplicate`).

Terminal rules: an element that is not decided as established (an unresolved element included) means not liable. A criminal element decided below the top tier, or below the criminal confidence, is not established. A question that cannot be answered is retried. It is never decided against anyone. An appeal recorded in time is always heard. A respondent with too little to pay leaves a recorded shortfall and an open pool claim. A case never ends on a tie or an outage, and never escalates to a person.

B.3 A Determination

For tiers $t = s, \ldots, 3$ from the start tier s, the module does these steps. Skip a tier that cannot answer (no structural answer, no matching head, no quorate panel). Seat the fixed panel, or draw one of exactly the tier's quorum of eligible groups. Record every seat's answer or error. Refuse as an error any answer over a different package. Stop at the first tier whose aggregate has a unique winner at its band. At the top tier, a tie or a confidence below the band is *unresolved*. If no tier answered, the outcome is *unavailable*. The record's identity is the hash of its content.

B.4 A Facility

Active → repaid when the schedule is paid in full. Active → defaulted when an installment stays unpaid past the grace period. At default, each holder's breach case is filed for its share of the accelerated balance. Every servicing, transfer and default commits as one transaction, claimed on the facility's revision.

C Genesis Law and Engine

This appendix is generated from `GENESIS_LAW` and `GENESIS_ENGINE` in the reference module. The law hash prefix is `624d`

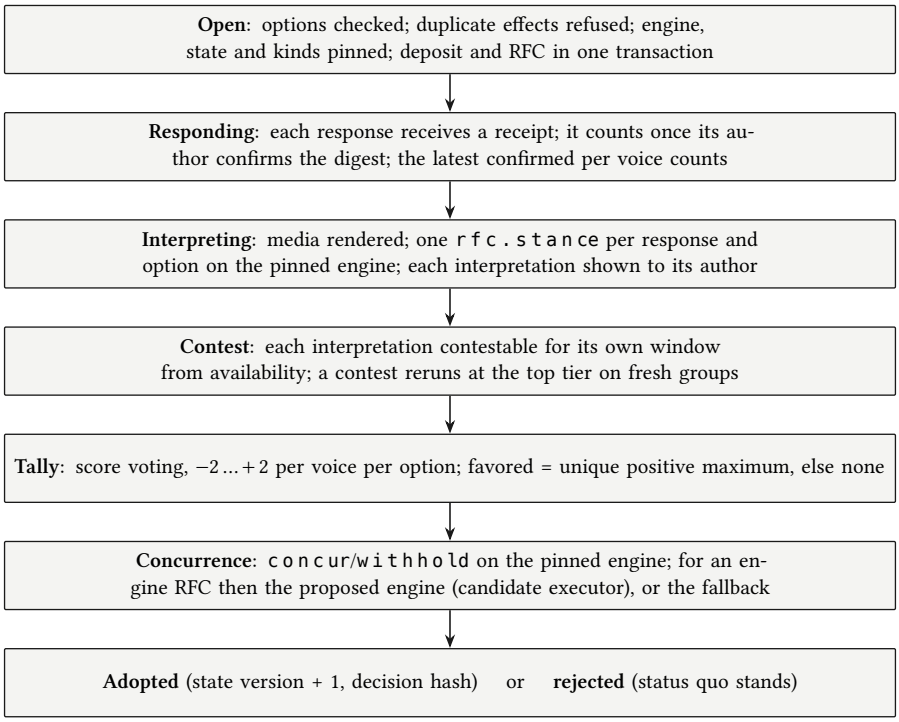


Figure B.1. The life of an RFC. A draft refused at opening never enters it.

`e651f203db37`. The engine hash prefix is `24edc06a9c3dfa9f`. The genesis state hash prefix is `3ca91e8207225264`.

C.1 People

The genesis law recognizes the kind `human` under the rule identity_fingerprint. It admits these modalities as evidence and responses: text, image, audio.

C.2 Duties

No fraud (`fraud`, criminal). No one may obtain property or an agreement from another by a representation of fact they know to be false, or make with reckless disregard for its truth, that the other relied on. *Protects:* property, agreement. *Elements:* representation, falsity, knowledge, reliance, harm.

No theft (`theft`, criminal). No one may take or use another's holdings or property without the owner's consent. *Protects:* property. *Elements:* taking, ownership, no consent, harm.

No violence (`violence`, criminal). No one may use the contract to arrange, pay for or carry out physical harm to a person. *Protects:* life, bodily integrity. *Elements:* act, harm.

No coercion (`coercion`, criminal). No one may compel another's labor, agreement or response to an RFC by threat of harm. *Protects:* liberty, agreement, voice. *Elements:* threat, extraction, causation.

Keep agreements (`breach`, civil). Whoever accepted an agreement must perform its terms or answer for the loss its breach causes. *Protects:* agreement. *Elements:* agreement, nonperformance, harm.

No exposure (`privacy`, civil). No one may publish another person's private information obtained through the contract without consent. *Protects:* privacy. *Elements:* private, source, publication, no consent.

C.3 Parameters

Table C.1. Genesis parameters.

Parameter	Genesis value
Law RFC window	14 days
Engine RFC window	28 days
Extraction contest window	3 days
RFC deposit	$10
Responses sampled in full for concurrence	24
Cooldown before a rejected RFC's options can reopen	90 days
Case response time	7 days
Case evidence time	7 days
Case cost deposit (as far as the balance allows)	$5
Reopening window after a final judgment	180 days
Unprocessable after	14 days
Open cases per claimant	20
Appeal window	7 days
Expedited: time to answer, for evidence, to appeal	60 s, 60 s, 60 s
Routing: seat preference softness (τ)	$0.002
Routing: value a wait blocks on the transaction path (κ)	$0.01 a second
Routing: deadline for an agreement's proposition	2 s
Fast answers reviewed by the top tier	5%
Deposit to raise a case's stakes	100 bps of the raise
Stakes an RFC response is interpreted at	$100
Interim freeze	14 days
Criminal confidence	950/1000
Penalty multiple	2×
Maximum exclusion	180 days
Pool premium on bond issuance	50 bps
Pool cap per judgment	$10,000
Commons fee on settled transfers	10 bps
Credit servicing fee	25 bps
Credit grace period	7 days

C.4 Engine

Bands: tier 1 950, tier 2 850, tier 3 700 per mille. Quorum: 2 of 3 independent groups at tier 2, 3 of 6 at tier 3 (2 disjoint panels). Renderers (tier r below): image — render-gpt-5.6-luna, render-claude-haiku-4-5; audio — render-whisper-large-v3-turbo, render-gpt-4o-mini-transcribe. No MLP heads. No seat records a controlling entity yet.

Tier	Seat	Operator	Lineage
1	`bge-base-en-v1.5`	cloudflare	bge
	Pin: weights `0dbc3b2eeffc` `hf:BAAI/bge-base-en-v1.5@a5beb1e3e68b9ab74eb54cfd186867f64f240e1a`		
	Figures: error ≤ 0‰, unsure 0‰, 300 ms mean, 1000 ms p99, $0.0001 a call (prior)		
2	`jev-1.13.0`	typesafe	jev
	Pin: version `jev-1.13.0`		
	Figures: error ≤ 95‰, unsure 159‰, 113 ms mean, 191 ms p99, $0.0001 a call (measured)		
2	`clef`	cloudflare	qwen
	Pin: weights `92832a3c35dc` `hf:Cloudflare/clef@2f3de3dd85f379784083b0814d997ab627200f0c`		
	Figures: error ≤ 117‰, unsure 341‰, 387 ms mean, 622 ms p99, $0.0004 a call (measured)		
2	`clef-flash`	cloudflare	qwen
	Pin: weights `990afd248a60` `hf:Cloudflare/clef-flash@17f0b0ad64efb65d273590632833508766b2aae6`		
	Figures: error ≤ 325‰, unsure 818‰, 180 ms mean, 528 ms p99, $0.0001 a call (measured)		
2	`llama-3.1-8b`	cloudflare	llama
	Pin: weights `b95e1c842c5c` `hf:meta-llama/Llama-3.1-8B-Instruct@0e9e39f249a16976918f6564b8830bc894c89659`		
	Figures: error ≤ 466‰, unsure 295‰, 546 ms mean, 946 ms p99, $0.0004 a call (measured)		
2	`gpt-5.6-luna`	openai	gpt
	Pin: version `gpt-5.6-luna`		
	Figures: error ≤ 152‰, unsure 0‰, 2170 ms mean, 5292 ms p99, $0.0010 a call (measured)		
3	`claude-opus-5-5`	anthropic	claude
	Pin: version `claude-opus-5-5`		
	Figures: error ≤ 83‰, unsure 23‰, 3579 ms mean, 7076 ms p99, $0.0150 a call (measured)		
3	`gpt-5.6-sol`	openai	gpt
	Pin: version `gpt-5.6-sol`		
	Figures: error ≤ 119‰, unsure 0‰, 1830 ms mean, 5280 ms p99, $0.0050 a call (measured)		
3	`deepseek-v4-pro`	deepinfra	deepseek
	Pin: weights `249bebe0762f` `hf:deepseek-ai/DeepSeek-V4-Pro@b5968e9190ef611bbf34a7229255be88a0e937c1`		
	Figures: error ≤ 273‰, unsure 23‰, 2150 ms mean, 3850 ms p99, $0.0014 a call (measured)		
3	`llama-3.3-70b`	cloudflare	llama
	Pin: weights `68d90187d3d2` `hf:meta-llama/Llama-3.3-70B-Instruct@6f6073b423013f6a7d4d9f39144961bfbfbc386b`		
	Figures: error ≤ 242‰, unsure 409‰, 2111 ms mean, 4506 ms p99, $0.0011 a call (measured)		
3	`qwen3.8-max`	alibaba	qwen
	Pin: version `qwen3.8-max`		
	Figures: error ≤ 86‰, unsure 68‰, 45546 ms mean, 67660 ms p99, $0.0048 a call (prior)		
3	`kimi-k2.6`	openrouter	kimi
	Pin: weights `03dd534e3ddf` `hf:moonshotai/Kimi-K2.6`		
	Figures: error ≤ 124‰, unsure 45‰, 19390 ms mean, 90004 ms p99, $0.0016 a call (measured)		

(continued)

Tier	Seat	Operator	Lineage
r	`render-gpt-5.6-luna`	openai	gpt
	Pin: version `gpt-5.6-luna`		
r	`render-claude-haiku-4-5`	anthropic	claude
	Pin: version `claude-haiku-4-5-20251001`		
r	`render-whisper-large-v3-turbo`	cloudflare	whisper
	Pin: version `@cf/openai/whisper-large-v3-turbo`		
r	`render-gpt-4o-mini-transcribe`	openai	gpt-transcribe
	Pin: version `gpt-4o-mini-transcribe`		

An open-weight pin is a SHA-256 value over the model's published file manifest at one commit of its source repository. A hosted pin is the exact model id that the provider serves. Both are recorded here as the reference module states them. No seat's accuracy on contract questions has been measured (Section 19).

D Worked Cases

The reference module computes every number below (`generate_fixtures.mjs`). Where a real engine would answer, the determinations that a case depends on are stipulated, and the text says so. People and amounts are synthetic.

D.1 Case-01: An RFC with Two Rival Options

An RFC proposes lowering the commons fee from 10 to 5 basis points (`fee5`) or abolishing it (`fee0`). Its window is 14 days. Each interpretation is contestable for 3 days from when it is shown. The module refuses an attempt to add a third option that also sets the fee to 5 under another title (rfc.duplicate_effect). Seven people respond and confirm their receipts. Their interpreted scores are:

Table D.1. Interpreted scores, −2 to +2, per voice and option.

Person	fee5	fee0
ana	2	2
ben	1	-1
cy	2	0
dee	-1	-2
eli	0	2
fay	1	0
gus	-2	-2

The scores are $S_{\text{fee5}} = 3$ and $S_{\text{fee0}} = -1$ (the counts of voices scoring $-2, -1, 0, +1, +2$ are 1, 1, 1, 2, 2 and 2, 1, 2, 0, 2). The favored option is `fee5`. Suppose a third option of two basis points had been on the ballot, and everyone but Gus opposed it strongly. Its score would be -10. The first two options' scores would stay 3 and -1. Under score voting, an option's score does not depend on what else is offered.

The interpreter was slow, so an eighth response is interpreted 9 days after close. Its author can contest it until day 12 after close. Its `fee0` interpretation was unresolved (fee0: unresolved). So it scores 0 there, which is neither support nor opposition.

With the engine's concurrence decided (stipulated), the RFC adopts `fee5`. The state moves to version 1 with a commons fee of 5 bps. The decision record's hash begins `ed2b9b62cc147043`. Had the engine withheld, the result would be null. Had its panel been divided, the result would be null — "the pinned engine was divided or uncertain: no concurrence (rfc.concur:null:unresolved:3:900)". Applying the same decision again to the moved state is refused: adopt.superseded.

D.2 Case-02: Replacing the Engine

An engine RFC raises the classifier band. Its window is 28 days. One person supports it. With only the outgoing engine's concurrence recorded, the result is null, because the proposed engine has not been asked. If the outgoing engine withholds, the result is null, whatever the proposed engine would say. The proposed engine is never asked. If the outgoing engine concurs and the proposed engine's panel is divided, the result is null. With both concurring, the RFC adopts `band900`. Suppose the outgoing engine could not seat a panel at all for a further window as long as the RFC's own. Then the proposed engine's concurrence alone adopts `band900`.

D.3 Case-03: Three Seats of One Operator

Five seats answer a yes/no question, each at 900. Three belong to one operator and answer yes; two independent seats answer no. The five seats form 3 groups. The tier decides `no` at 600, because the operator's three seats weigh as one.

D.4 Case-04: A Climb

A service agreement asks whether the delivered work fulfills the brief (`holds` / `does_not_hold`). The package has two items, and its digest begins `b2d0aa677ff3b6e3`. The genesis engine draws its classifier panel by seed from its three groups. The panel answers gpt-5.6-luna error; clef does_not_hold 900 (stipulated): does_not_hold at 900 over 1 groups, 1 of 2 groups answered. So the question climbs. The deliberative panel of 3 groups answers gpt-5.6-sol holds 800; llama-3.3-70b holds 900; qwen3.8-max holds 850: holds at 850, above the band of 700. The outcome is decided: `holds` at tier 3. The determination's identity begins `e642ddbbbbe2898b`.

D.5 Case-05: No Winner by Order

Four independent groups answer a concurrence question, each at 900. They give four different labels: `concur`, `kernel_violation`, `voice_mismatch`, `outside_law`. The aggregate over 4 groups has no winning label (tie: yes). Version 0.4 would have answered `concur`, the first label, at 250. (The concurrence question now has only `concur` and `withhold`. So three objections cannot split against one assent in the first place.)

The service-agreement question of Appendix D.4, started at the top tier, meets a panel that answers gpt-5.6-

sol does_not_hold 900; llama-3.3-70b does_not_hold 900; qwen3.8-max holds 500. `does_not_hold` has the larger weight, but its confidence is 600: below band 700: unresolved. The outcome is unresolved, with no label. The agreement's own rule says what that means for the escrow.

D.6 Case-06: An MLP Head

A two-layer head with labels no/yes, input scale 100 and identity-like weights receives the embedding $(0.12, -0.31)$. Quantization gives $(12, -31)$. The hidden ReLU layer gives $(12, 0)$. The output layer gives logits $(-12, 12)$. The label is `yes` and the margin is 24. The calibration table's largest threshold below the margin assigns 950. Any machine computes the same numbers. The head's hash begins `e3bd0562c93234ae`.

D.7 Case-07: A Panel and Its Appeal

A case between *buyer* and *seller* starts its first instance on the genesis engine. The engine draws its top-tier panel by seed from the six independent deliberative groups: kimi-k2.6, qwen3.8-max, llama-3.3-70b. An appeal draws again and excludes every group that sat: deepseek-v4-pro, claude-opus-5-5, gpt-5.6-sol. No group sits twice. A third disjoint panel does not exist (quorate: no). The engine admits exactly as much appellate capacity as the kernel requires. That is why a case has one appeal.

D.8 Case-08: An Image in the Record

A claimant submits a photograph of a receipt. The contract stores it by its SHA-256 and asks both image renderers: render-gpt-5.6-luna, render-claude-haiku-4-5. Both produce text (ok: yes). The record carries 2 text items in the photograph's place, one per rendering. Had one renderer failed, the item would not be before the seats (ok: no). The record would mark `claimant/1/1` as unprocessed and say that it counts against no one. Either way, every deciding seat reads the same text. No tier loses its quorum because it lacks a vision model.

D.9 Case-09: A Criminal Standard, Element by Element

A buyer alleges fraud. A seller represented 120 independent paying customers, of whom 47 qualified. The buyer paid an $800 deposit in reliance. The claim names one harm item. It seeks restitution, a penalty and 90 days' exclusion, and each remedy answers that item. Because the claim is criminal, its instance starts at tier 3. The top tier decides each of the duty's elements separately: representation, falsity, knowledge, reliance, harm. If the panel establishes every element at 930, the respondent is *not liable*, because that is below the 950 standard. Elements established at tier 2, even at 990, also leave the respondent *not liable*. Suppose the panel answers 970 on every element, chooses the claimant's figure and justifies every remedy. Then the respondent is liable. The remedies are restitution of $800; a penalty capped at 2× the total award, $1,600 ($800 to the victim, $800 to the pool); and 90 days' exclusion. Suppose a filing asked for restitution and compensation of the full $800 each. The module would have refused it at the door: k8.double_recovery. Every determination is stipulated.

D.10 Case-10: A Civil Judgment Reaching a Bond

A seller took $800 for an order. The buyer says that the seller delivered nothing. The claimant deposits $5. The case starts at tier 2. The seller answers that half was delivered and states its figure, $500. Every element is established (stipulated). The panel finds the claimant's figure closer to the proved loss, so the award is $800. Had the panel been unresolved, the award would have been the respondent's figure, $500.

The engine's cost of the case is $12, paid by the respondent. The seller holds $150 in cash and participations in a bond worth $10,000. Execution takes $150 in cash and $662 of the bond. That is $812 in total, and nothing more. The shortfall is $0, and the claimant's $5 deposit comes back. Had the seller held only the cash, execution would collect $150. The victim's $650 shortfall would become an open pool claim.

Had the buyer lost, the claimant would pay. $5 of the deposit goes to costs, and $0 comes back. Had the engine been unable to decide for the law's period, the commons would bear the cost. The deposit, $5, would come back whole. Had the seller appealed in time and no fresh panel could sit, the case would still be appealed a year later. No clock finalizes an appealed case.

D.11 Case-11: Two Pool Epochs

Issuing $100,000 of bonds pays a $500 premium into the pool. Later the pool holds $10,000, with a cap of $10,000 per claim. Three judgments have left victims short. The first epoch pays every capped need the same fraction. After $14,000 more premiums and penalties, a later epoch pays what is still owed:

D.12 Case-12: The $10,000 Credit Cycle

An investor, Ivy, commits a mandate of $10,000 at 12% a year over 12 monthly installments, to persons only. The module refuses an applicant without a personhood proof (borrower_not_a_person). A person, Bo, draws the whole of it to buy a CNC router. The issuance premium of $50 goes to the pool, and Bo receives $9,950. Ivy holds the bond.

The first installment, due after 30 days, is $833.33 of principal and $98.63 of interest. Before it is paid, Ivy sells a quarter of the bond to Pat. The contract collects the installment from Bo's balance and takes $2.32 for the commons. It distributes $697.23 to Ivy and $232.41 to Pat. Bo then stops paying, and installment 2 passes its grace period. The outstanding $9709.08 is accelerated. Two breach cases are filed: Ivy's for $7281.81 and Pat's for $2427.27. Each passes the filing checks (refusals: 0). From there they are ordinary cases (Appendix D.10). In the reference tests, they reach judgment against an empty respondent. The two holders' pool claims are then paid in equal proportion as far as the pool goes.

Table D.2. Equal claims (case-a, case-c) are paid equally in each epoch, to the cent; the larger claim is capped at $10,000 over both epochs, and the rest stays owed by its respondent. After epoch 1 the pool holds $0; after epoch 2, $6,000. The order in which the judgments executed changes nothing.

Claim	Shortfall	Epoch 1	Epoch 2	Owed by respondent	Closed
case-a	$4,000	$2222.22	$1777.78	$0	yes
case-b	$15,000	$5555.55	$4444.45	$5,000	yes
case-c	$4,000	$2222.23	$1777.77	$0	yes

D.13 Case-13: An Anonymous Voice

Three people register Semaphore commitments for their human voices. Ann (commitment `101`) and Bea (`201`) register before an RFC opens, and Cal (`301`) after it. After opening, Ann rotates her identity to `102`. The RFC pins the human tree as it stood at opening: 2 leaves, `101,201`. A day after Cal registered, the tree is `201,102,301`. The pinned root is part of the RFC's identity. An RFC opened over another root is another RFC (yes).

Ann is in the pinned tree, so she answers this RFC only anonymously (yes). Her rotated identity is not in the pinned tree and gives her no second voice. Cal is not in the pinned tree (no), so he answers with his signed-in voice. One anonymous member answers "yes", then "on reflection, no". Another answers "yes". Both of the first member's proofs carry nullifier 77, so they are one voice, `anon:human:77`. The later response replaces the earlier one. 2 responses count, and the first member's counted response is "on reflection, no".

The module accepts a contest proof whose root, scope and message are right (null). It refuses the same proof in these cases: against another root, `not_the_pinned_tree`; in another RFC's scope, `wrong_scope`; over the lookup message, `wrong_message`; and claimed for a kind not recognized at opening, `kind_not_recognized_at_open`. It refuses an anonymous response that cites an image by URL (`anonymous_media_must_be_uploaded`). Anonymous media must be uploaded under a proof.

D.14 Case-14: An Attested Answer and a Rewritten History

A seat names an attestor. The seat answers `yes` at 950 over a package. The answer carries the attestor's signature (beginning `635a94222687a2a7`) over the `cc:attestation:v4` message, and it counts (null). The same signature does not survive a changed label (wrong_answer), a different package (wrong_subject), a different attestor key in the engine (wrong_key), a missing attestation (missing) or a flipped signature byte (bad_signature).

The contract's log records four acts. Its head begins `b7c7d9101ba8a765`, and the chain verifies (yes). Suppose an operator rewrites the second entry's body digest. The replica then reports 1 problem. An operator who drops an entry leaves 2. A checkpoint of the fourth entry's head matches the log (yes). A checkpoint that claims the third entry's head for the fourth position is a mismatch (1).

E Mandate-Constrained Allocation Mathematics

Let i index capital mandates and j candidate claims. $x_{ij} \geq 0$ is the amount of mandate i allocated to claim j. K_i is the available capital. D_j is the maximum approved financing demand. $e_{ij} \in \{0, 1\}$ denotes verified eligibility under hard contractual constraints. A zero means that the claim is ineligible or not sufficiently established. It does not necessarily mean that the claim is socially undesirable.

$$\sum_j x_{ij} \leq K_i, \qquad \sum_i x_{ij} \leq D_j, \qquad x_{ij} \leq e_{ij} K_i. \tag{E.1}$$

Sector, obligor, related-party, geography, maturity, and settlement-asset limits add constraints such as $\sum_{j \in S} x_{ij} \leq c_{iS} K_i$. Aggregation must follow common beneficial ownership and shared risk. It must not treat several legal wrappers as independent borrowers. Liquidity eligibility concerns realizable cash and redemption terms, not only a stated maturity.

E.1 Objectives Do Not Override Mandates

Let a_{ij} be a documented estimate of mandate alignment. It is not a measurement of moral worth. Let $\hat{r}_j$ be an estimated net return after stated costs, and $\mathscr{R}_i(x)$ a risk measure. One possible objective is

$$\max_x \sum_{i,j} x_{ij}(\alpha_i a_{ij} + \beta_i \hat{r}_j) - \sum_i \gamma_i \mathscr{R}_i(x), \qquad x \in \mathscr{F}, \tag{E.2}$$

where $\mathscr{F}$ is the feasible set defined by approved constraints. The coefficients express tradeoffs that the investor approved. They must not be hidden in model defaults. Alternatively, the system may display several objectives as a frontier, and the authorized investor chooses from it. A single combined score is not required.

If a mandate asks for a minimum *estimated* portfolio return, it could add

$$\sum_j x_{ij} \hat{r}_j \geq r_i^{\text{target}} \sum_j x_{ij}. \tag{E.3}$$

This is a constraint on an estimate, not a promise about realized returns. The all-zero allocation is feasible unless the mandate explicitly requires minimum deployment. A minimum deployment requirement can make $\mathscr{F}$ empty. Then the output is an explanation of the infeasibility and a request for a new mandate. The system does not relax the constraint in secret.

E.2 Losses and Model Uncertainty

Let $L_i^{(s)}(x)$ be the net portfolio loss under scenario s, which has an explicitly justified probability p_s. A possible tail-risk constraint uses conditional value at risk at level η:

$$\mathrm{CVaR}_\eta(L_i) = \min_z \left[z + \frac{1}{1-\eta} \sum_s p_s \max(L_i^{(s)} - z, 0) \right] \le \ell_i. \tag{E.4}$$

This finite-scenario definition assumes a specified loss distribution. Neither the blockchain nor the formula validates the scenarios, their probabilities, the chosen horizon or the recovery assumptions. Stress tests should also include severe unweighted scenarios. They should include shared-service failures that historical repayment data omits.

E.3 Verifiable Execution, Contestable Inference

The allocation artifact contains these items: the mandate and borrower versions, the eligibility evidence, the selected amounts, the scenario set, the model and solver versions, the conflicts, the costs, the concentration checks, and an explanation of the rejected or unfunded alternatives. Before release, a deterministic checker recomputes the hard arithmetic and the permissions. It does not certify that the projected revenues will occur. It does not certify that the solution is globally optimal, unless an appropriate optimality certificate exists.

Semantic retrieval stays upstream. It finds candidates whose descriptions may match the intent. Financial feasibility, verified use of proceeds, economic additionality and causal mission impact each require different evidence. Fund managers may publish aggregate performance by model vintage. They must still protect the confidential information of individual borrowers. Retraining a model does not rewrite the model that justified an earlier loan.

F Cash Flows and Loss Allocation

All amounts are hypothetical dollars. `generate_fixtures.mjs` computes them in integer cents. The examples exclude taxes, currency risk, valuation changes and unstated costs. They do not insure against them.

F.1 A Direct Loan Participation

A hypothetical investor contributes $10,000 to a restricted pool. The pool advances $6,000 to borrower A and $4,000 to borrower B, each for one year at 8% simple interest. The accepted servicing charge is $200.

Table F.1. A fully performing participation: a net gain of $600, or 6%.

Cash-flow item	A	B	Total
Principal	6,000	4,000	10,000
Interest if both pay	480	320	800
Full collections	6,480	4,320	10,800
Servicing charge	—	—	(200)
Net investor distribution	—	—	10,600

Suppose A pays in full, and B pays no interest and recovers 40% of principal. Then the collections are $6,480 plus $1,600, or $8,080. After the same charge, the distribution is $7,880. This is a loss of $2,120 (21.2%). B's default is not by itself a case, because financial distress is not fraud. If B's default breached an accepted term, the pool may sue. A judgment then reaches B's holdings like any other judgment (Section 8.13).

F.2 Issuer Obligations and First Loss

Suppose instead that an issuer promises $10,500 at maturity, whatever the borrowers do. Then full collections after servicing leave $100. The default scenario leaves a shortfall of $2,620. This shortfall is the issuer's liability. The issuer does not pass it through to the investor.

A separate principal-only portfolio of $100,000 has $10,000 of junior funding and $90,000 of senior funding. For a loss L, junior keeps $\max(J-L, 0)$ and senior keeps $S-\max(L-J, 0)$.

Table F.2. Loss allocation changes who bears loss, not the amount of capital actually supplied.

Principal loss	Collections	Junior remaining	Senior remaining
6,000	94,000	4,000	90,000
16,000	84,000	0	84,000

The protection pool is not first-loss capital for investment portfolios. It pays victims of uncollected judgments (Section 10.3). A borrower's default that breaches no term creates no claim on it.

G Economic Measurements and Bulletin Schema

Measurements require a denominator, a boundary, a time convention, and a correction policy. A ledger event is evidence for a specified accounting event. It does not permit the system to classify every movement of money as production.

Indicator	Operational definition	Limitation / required disclosure
Settled payments	Sum of qualifying transfers over a declared interval, grouped by asset and economic purpose	Exclude internal transfers where required; volume is not value added
Verified commercial receipts	Receipts linked to qualifying customer transactions, net of specified refunds	Disclose verification standard, related parties, and off-protocol coverage

(continued)

Indicator	Operational definition	Limitation / required disclosure
Covered value added	Qualifying output less qualifying intermediate consumption over a fixed boundary	Requires economic classification and adjustments; not automatically national GDP
Active enterprises	Entities meeting a published operating-activity criterion	Registration is not operation; disclose inactivity and duplicate-entity treatment
New enterprise formation	Newly registered entities and, separately, newly active enterprises	No implication that every registered shell is a new business
Credit outstanding	Principal receivable at the observation time, with currency and impairment basis	Include assigned claims once; disclose unobserved liabilities
New credit issuance	New principal advanced during the interval	Exclude principal refinancing or show it separately
Delinquency	Overdue principal / eligible outstanding principal under a named day-past-due definition	Specify grace periods, rescheduling, and cohort age
Realized credit loss	Unrecovered principal and specified costs for a mature cohort	Incomplete recoveries and immature loans require separate estimates
Public fiscal balance	Recurring recognized revenue less specified spending, under a stated accounting basis	Do not classify borrowing or returned loan principal as revenue
Liquidity coverage	Eligible available cash relative to obligations due in a stated horizon	Pledged reserves and conditional assets are not free cash
Market yield	Yield implied by a qualifying executable or completed transaction and its cash-flow terms	Distinguish bids, offers, last trades, indicative models, and stale markets
Price index	Weighted changes for a specified basket and quality-adjustment method	Composition shifts, sparse prices, and changing quality can dominate
RFC participation	Persons notified, responding, and contesting their extraction, by kind of people, using separate counts	Silence is not support or opposition; protect small cohorts from identification
Pool coverage	Pool balance against open and expected victim claims	Premiums are not reserves for other purposes; disclose correlated exposures
Coordination burden	Distribution of elapsed time, fees, labor, disputes, and interventions per completed workflow	Include failures and excluded applicants; faster is not always better

G.1 A Machine-Readable Bulletin Item

```
indicator_id: commercial_receipts.change_7d
status: illustrative_schema
value: numeric_or_unavailable
unit: percent_change
population: explicit_entity_and_transaction_boundary
window_start: canonical_timestamp
window_end: canonical_timestamp
observation_cutoff: canonical_timestamp
baseline_window: explicit_comparable_period
coverage: observed_share_or_unknown
method_version: content_digest
source_commitments: referenced_input_records
uncertainty: estimation_range_or_not_applicable
missingness: known_gaps_and_selection_limits
revision_vintage: monotonic_version
stale_after: canonical_timestamp_or_trigger
narrative: generated_only_from_this_record_and_cited_context
```

An “opening high” is not a generic economic statistic. An opening price, an interval high, a percentage return, an annual growth rate and an annualized growth rate are different computations. The narrator must not use market vocabulary only to sound financially sophisticated. When there are too few trades for a yield curve, the output is “insufficient market observations”. It is not an interpolated chart presented as observed borrowing costs.

A published correction produces a new vintage, linked to the old one. Historical analyses state which vintage was available at the decision time. Privacy suppression and sampling can make a figure less precise. The narrative must keep that limitation. It must not round the figure into false certainty.

H Threats and Controls

The contract itself, an engine provider, its founder, its operator and its majority can each be the adversary. Each threat names its control and what remains. The build generates the table from `threats.json`. That file also names the rules of Appendix I that implement each control. Section 18.2 compares each threat with the status of those rules.

ID	Threat	Control and residual
th-01	One provider runs several seats to outvote the rest (Section 7.5)	Operator and lineage grouping. *Residual:* Hidden common control or undisclosed shared training
th-02	A provider changes or retires a model behind its ID (Section 7.3)	Open-weight seat required per tier; unavailable rather than guessed. *Residual:* Serving-stack drift of open weights
th-03	An incumbent engine entrenches itself, or a captured engine installs itself (Section 6.8)	Dual-engine concurrence; voice required. *Residual:* Both engines sharing one failure
th-04	The engine rules instead of interpreting (Section 6.7)	Concurrence is concur or withhold, on three grounds only; it can block, never adopt, and concurs only when decided. *Residual:* An engine that blocks in bad faith, answered by an engine RFC
th-05	Interpretation reads support into responses that express none (Section 6.4)	Per-response determinations on the record, each shown to its author with its own contest window; contest on fresh groups; a seeded sample read by concurrence. *Residual:* Respondents who never check

(continued)

ID	Threat	Control and residual
th-06	One person, many voices (Section 5.1)	Per-kind uniqueness rules; earliest proof holds the voice. *Residual:* Issuer error and identity theft
th-07	Intensity inflation, or agenda games with clone options (Section 6.6)	Score voting bounded to ±2 per option; an option's score independent of the others; duplicate effects refused. *Residual:* Near-duplicates that differ in effect
th-08	A majority uses law against a minority's holdings (Section 5.4)	Article k5: no freezing, seizing or singling out; structural check plus outside_law. *Residual:* A harm-based duty drafted to burden one group in practice
th-09	Courts used to harass (Section 8.1)	Objective scope checked before any model is asked; loser pays. *Residual:* A well-funded litigant willing to lose
th-10	A respondent escapes by disrupting appeals, or an appeal is lost to a late clock (Section 8.11)	The appeal is a stage no clock finalizes; two disjoint top-tier panels admitted; an appeal waits for a fresh panel and the judgment is not executed meanwhile. *Residual:* A long outage delays both sides
th-11	A respondent hides assets in bonds (Section 8.13)	Judgments reach bonds; interim freeze. *Residual:* Assets outside the contract
th-12	A victim left unpaid, or paid after others by arrival (Section 10.3)	Pool allocation epochs over every open claim, persistent residuals, subrogation. *Residual:* A pool exhausted by correlated losses
th-13	Evidence that instructs the model (Section 9)	Closed labels; effects only through fixed functions. *Residual:* An injected wrong label
th-14	A key above the contract, or a code change that alters the kernel's effect (Section 15.4)	Registry holds upgrade authority; code upgrades need both engines and a seven-day timelock; genesis manifest pinned. *Residual:* Compromise of a seat's signing service
th-15	The founder keeps a back door (Section 18.7)	Genesis by RFC; no founder power after it. *Residual:* Undisclosed infrastructure access, recorded as a power
th-16	A rejected RFC is reopened until people tire (Section 6.1)	Cooldown on options with identical effects. *Residual:* Variations that differ in effect
th-17	A divided engine read as a decision (Section 7.1)	Unique winner at the band, or unresolved; each use states what unresolved means. *Residual:* A matter the engine leaves unresolved costs whoever bears its burden; a hidden correlation between groups
th-18	A proposed engine harvests credentials or private evidence (Section 7.10)	Credentials bound to operator hosts; candidate executor renders nothing; asked only after the outgoing concurs. *Residual:* The proposed engine's providers see the public RFC record they are asked about
th-19	Seats decide on different evidence; media breaks quorum (Section 9.3)	One canonical text package, delivered digests, two renderers per medium, unprocessed items marked. *Residual:* A delivered digest is the adapter's report, not the provider's, until the seat is attested
th-20	Money moved twice, or half-moved (Section 15.5)	Op-claimed conditional batches with deterministic IDs; revision-guarded writes. *Residual:* Movements outside the contract's database (external payment rails) need their own reconciliation
th-21	A remedy list that multiplies recovery (Section 8.10)	Harm items; per-item money bounds; one penalty cap; remedy justification. *Residual:* An overstated harm item, answered only by the respondent's own figure
th-22	The same matter litigated until the respondent gives up (Section 8.3)	Screening; matter families; one reopening with new evidence; loser pays. *Residual:* The same matter recast as a different act, if the engine is persuaded
th-23	A revoked proof replayed (Section 5.1)	Monotonic statements; revocation tombstones. *Residual:* An issuer that signs a newer issuance by mistake
th-24	Stuck matters starve the scheduler (Section 15.5)	Due-work queue, earliest due first. *Residual:* Bounded work per sweep delays every matter when the queue is long
th-25	An agent commits its owner to debt (Section 10.4)	Credit commitments need a person's own credentials. *Residual:* A person's stolen credentials
th-26	The platform learns which RFC response is whose (Section 6.3.3)	Anonymous voice: a zero-knowledge proof of membership in the tree the RFC pinned, sent without any credential; nothing stored links a nullifier to a person. *Residual:* A small anonymity set, network metadata, timing, the content itself, and the proving-artifact host seeing the person's address
th-27	One person counts twice: once signed in and once anonymously, or through a rotated identity (Section 6.3.1)	A person in an RFC's pinned tree answers it only anonymously; a commitment registered or rotated after opening is not in the pinned tree. *Residual:* A person who loses their identity file cannot answer RFCs opened while it was registered
th-28	A forged or replayed proof speaks for someone (Section 6.3.2)	Groth16 verification against Semaphore's own keys inside the worker; root, scope and message checked per action; a response's message binds its digest and a fresh nonce. *Residual:* A flaw in the circuit or its trusted setup

(continued)

ID	Threat	Control and residual
th-29	The operator of the sweep fabricates or alters a seat's answer (Section 16.1)	An attested seat's answer counts only with its attestor's signature over that exact answer; the replica recomputes every published determination. *Residual:* Seats without an independent attestor — every genesis seat today — rest on the operator's word
th-30	The operator rewrites, reorders or forks the contract's history (Section 16.2)	A public hash-chained log of every act, its heads anchored on the registry's checkpoint instruction and checked by the replica. *Residual:* Acts the operator never logs, until reconciliation or a party notices; checkpoints not anchored until the registry is deployed
th-31	Authority over a machine or service issued outside the engine (Section 17.2)	The command notary authorizes, delegates and revokes only on a top-tier quorum of the registry's seats, bound to the registry's version and engine. *Residual:* The programs are not deployed; an executor that ignores the notary
th-32	A copy of the contract on another chain diverges from the registry (Section 17.3)	The EVM mirror applies only the registry's engine-signed adoptions with the same seat keys, group quorums, dual-engine rule and timelock. *Residual:* A relay that stops relaying leaves the mirror behind (never ahead); the mirror is not deployed
th-33	A party with money buys a panel that favours it (Section 8.5)	Stakes come from the record; a paid raise is upward only and the stronger engine hears both sides; which groups sit is never weighted. *Residual:* A wealthier party can make a matter more expensive to lose for both
th-34	A slow or failing seat stalls the transaction path (Section 7.7)	Latency's p99 is a deadline: slow seats and tiers do not sit, and no path in time is unavailable, never a guess. *Residual:* Adopted p99 figures can be stale; a seat slower than its figure still sits until the next engine RFC
th-35	The host steers questions to chosen seats through the routing figures (Section 7.7)	The figures are part of the engine, changed only by RFC; the plan and every pick are recorded and recomputed by the replica. *Residual:* Figures rest on a small synthetic evaluation and the host's measurements

I Implementation Status

`guarantee_status.json` lists each consequential rule of this paper. For each rule it gives where the rule is specified, where it is implemented, which test exercises it, and its status. The build validates the file against `records/status.schema.json`. *Implemented* means present in this repository (the core module, the API worker, a script or a program). *Tested* means that a named test in a named file exercises the rule (Appendix J). Neither status means deployed, run against live models or independently reviewed. *Specified* means stated in this paper only.

ID	Rule (section)	Status	Limits
s-01	The kernel is fixed and identified by its hash (4.2)	tested	A different kernel is a different contract; nothing stops someone from deploying one
s-02	The contract state is created at genesis and published (18.1)	tested	Genesis is written by the service, not adopted by a genesis RFC
s-03	One person, one voice; per-kind uniqueness; one human key across documents and issuers (5.1)	tested	Names and birthdays are imperfect identity proxies; collisions and name changes require an accepted issuer signature and final correction judgment; no resolution issuer is commissioned
s-04	Signed statements apply in issue order; revocations are tombstones (5.1)	tested	No issuer for a kind other than human exists
s-05	Law changes are checked against kernel floors, human recognition, duty elements and rendering coverage (5.3)	tested	The semantic half of k5 rests on the engine
s-06	Market freedom: law may not set prices, single out holders or restrict consenting exchange (5.4)	implemented	Structural check plus a concurrence ground; no test of the engine applying it
s-07	Every model tier has two operators and an open-weight seat (7.4)	tested	Serving-stack drift of open weights
s-08	The top tier holds two disjoint panels: every top-tier decision is appealable (7.4)	tested	An unavailable appeal closes unprocessable after the processing window; the earlier judgment is not executed
s-09	Every admitted medium has two independent renderers (7.4)	tested	Both genesis audio renderers descend from OpenAI research
s-10	Seats of one operator or lineage count once; a tie has no winner (7.5)	tested	Hidden common control or shared training
s-11	Panels of exactly a quorum, drawn by seed, fixed per instance; parties and controllers never sit (7.6)	tested	Worker-assigned fresh entropy prevents claimant selection by filing digest; the host still controls entropy and genesis controlledBy declarations are empty

(continued)

ID	Rule (section)	Status	Limits
s-12	Outcomes are decided, unresolved or unavailable; nothing is guessed (7.1)	tested	Stipulated answers; no live model
s-13	One canonical text package with unique IDs; answers over a different package are errors (7.10)	tested	The delivered digest is reported by the adapter, not by the provider
s-14	Credentials are bound to operator hosts; a proposed engine renders nothing and is asked only after the outgoing concurs (7.10)	tested	A legitimate provider host still receives the question it is asked
s-15	Media is verified against its digest and rendered by two renderers; unrenderable items are marked (9.3)	tested	A rendering can omit or misread; renderers are not evaluated
s-16	MLP heads are integer-only and reproducible (7.8)	tested	The embedding itself is floating point
s-17	Heads are trained, quantized and calibrated conservatively (7.8)	tested	No head trained on a contract question
s-18	RFC opening checks options, refuses duplicate effects, pins engine, state and kinds (6.1)	tested	Options that differ in effect only slightly
s-19	Responses count once their digest is confirmed; the latest confirmed per voice; kinds fixed at opening (6.2)	tested	A receipt proves what was confirmed, not who drafted it
s-20	Interpretation reads actual changes; each interpretation has its own contest window (6.4)	tested	Respondents who never check
s-21	Binding score is opt-in; deliberative RFCs consider comments, merits, objections and participation without treating scores as binding (6.6)	tested	Participation is self-selected; representativeness is not inferred from a tally; substantive judgment remains an engine responsibility
s-22	Concurrence is decided concur or withheld; it reads the whole tally and a seeded sample (6.7)	tested	An engine that withholds in bad faith
s-23	An engine RFC: the outgoing engine first, then the proposed; fallback only when the outgoing cannot answer (6.8)	tested	Both engines sharing one failure
s-24	Adoption applies to the state read; a superseded RFC is not adopted; law history kept (5.2)	tested	A busy contract supersedes slow RFCs

(continued)

ID	Rule (section)	Status	Limits
s-25	The scope of justice and the remedy bounds are checked before anyone is served (8.2)	tested	A well-funded litigant willing to lose
s-26	One matter, one case: screening, families, one reopening (8.3)	tested	An unresolved screening lets the filing proceed
s-27	The clock starts at service; case records are read by parties only (8.4)	tested	Delivery to the inbox, not reading, is service
s-28	Cases are decided element by element on fixed panels; unavailable is never a merits loss (8.8)	tested	Stipulated answers; no live model
s-29	Criminal proceedings require adopted configuration-bound calibration metadata and a passing conservative precision bound (8.9)	tested	Report arithmetic is checked, not dataset authenticity; no audited criminal calibration is supplied; individual chat confidence remains self-reported
s-30	Final-offer figures; per-harm money bounds; one penalty cap; remedy justification (8.10)	tested	Figures far from the truth on both sides
s-31	A timely appeal blocks the first judgment; a disjoint panel hears it or it closes unprocessable after the processing window (8.11)	tested	Unprocessable appeals return deposits and leave the prior judgment unenforced; parties may refile
s-32	Execution is one atomic batch reaching cash and bonds; victims first (8.13)	tested	Only holdings the contract administers are reached
s-33	The public projection carries judgments, never records or determinations (8.16)	tested	A party may publish what it read
s-34	Loser pays engine costs both ways; unprocessable costs fall on the commons (8.14)	tested	Deposits are capped by balances
s-35	Interim freezes are bounded in amount and time and atomic (8.12)	implemented	No end-to-end test of a granted freeze
s-36	The protection pool allocates capped pro-rata claims; uninsured bonds have no coverage; insured-1 needs independent corroboration between unrelated parties (10.3)	tested	Known ownership and identity links are checked; hidden collusion needs external evidence; distinct attestor keys do not prove independent operators; genesis has no insurance attestors

(continued)

ID	Rule (section)	Status	Limits
s-37	The credit cycle declares insurance, guards holder revisions, and resumes every holder filing after interrupted default work (10.4)	tested	No legal review of the instruments
s-38	Every movement is one op-claimed conditional batch; writes are revision-guarded; due-work queue (15.5)	tested	Tested on SQLite; concurrency of the hosted database is assumed from its transaction semantics
s-39	The registry: decision hash, genesis manifest, signer sets of independent groups, engine and code changes need both engines, code behind a timelock (15.4)	tested	Not deployed; the chain does not verify the RFC procedure
s-40	Escrow frames read the registry; appeals climb tiers; the top tier is reheard once by fresh seats (15.4)	tested	Not deployed
s-41	Settlement certificates: signature, finality, replay, recipient and retrievability of rule and evidence (8.13)	implemented	No end-to-end settlement on chain
s-42	Exclusion after a criminal finding closes the market's doors to new commitments (8.10)	implemented	Only the doors that ask
s-43	Agreement disputes the evaluator cannot decide become cases whose final judgment settles the escrow (8.2)	implemented	No end-to-end test through an escrow in this pass
s-44	Genesis by RFC; the founder holds nothing after it (18.7)	specified	Not implemented
s-45	Anonymous voice: one commitment per voice; an RFC pins each kind's tree at opening; the nullifier is the voice, and a later proof replaces the earlier response (6.3)	tested	Proving has not run in a person's browser against a deployed host
s-46	One person, one count across the signed-in and anonymous routes; rotation never yields a second voice (6.3.1)	tested	A person who loses the identity file is silent on RFCs opened while it was registered
s-47	The anonymous routes read no credential and nothing stored links a nullifier to a person (6.3.3)	tested	Network metadata, timing, content and set size can still identify; no relay is provided

(continued)

ID	Rule (section)	Status	Limits
s-48	Proofs are verified in the worker against Semaphore's own keys; root, scope and message are checked per action (6.3.2)	tested	Rests on Semaphore's circuit and trusted setup
s-49	An attested seat's answer or rendering counts only with its attestor's signature over that exact answer (16.1)	tested	No genesis seat names an attestor; until an engine RFC names attestors run by others, the operator signs nothing and is trusted
s-50	The replica recomputes versions, RFC tallies and decisions, every published determination and a party's case from the public record (16.4)	tested	It cannot check what is not published: hosted models' correctness, other parties' cases, money
s-51	Every act is appended to a public hash-chained log of digests, idempotently; a lost append is reconciled (16.2)	tested	An act the operator never logs is caught only by reconciliation or a party
s-52	Log heads are anchored by the registry's permissionless checkpoint instruction, strictly increasing per submitter (16.3)	tested	The registry is not deployed, so every checkpoint today is recorded not anchored
s-53	Command authority is authorized, delegated and revoked only on a registry top-tier quorum bound to its version and engine; delegation never widens (17.2)	tested	Not deployed; Rust unit tests only (no BPF build or validator run here)
s-54	An EVM mirror follows the registry's engine-signed adoptions with the same keys, quorums, dual-engine rule and timelock, and releases escrow on registry-quorum frames (17.3)	tested	Not deployed; Ed25519 in Solidity costs about 0.8M gas per signature, so it suits L2s, not Ethereum mainnet
s-55	How much of the engine a question uses is chosen by expected loss, and the plan is recorded and recomputable (7.7)	tested	As good as the adopted figures; the loss model's assumptions (unresolved at half the stakes, ρ) are stated, not measured

(continued)

ID	Rule (section)	Status	Limits
s-56	Within a group the sitting seat is drawn from a soft preference; which groups sit is never weighted (7.7)	tested	The seed is host entropy for cases (as for panels)
s-57	A deadline caps the path by p99; no path in time is unavailable, never a guess (7.7)	tested	p99 figures are adopted, not live: a seat slower than its figure still sits
s-58	Stakes come from the record; a party may only raise them, for both sides, with a deposit settled by loser-pays (8.5)	tested	Bond exposure and κ from the ledger are not in the stakes yet
s-59	Expedited procedure only for civil matters on an agreement every party accepted it in, never below the kernel's expedited floors (8.6)	tested	Consent is the signed evaluator's flag; an agent's owner accepts it for the agent
s-60	A seeded share of fast determinations is reviewed by fresh top-tier groups; reviews measure, never reverse; identical routed questions reuse the record (7.7)	tested	A disagreement changes nothing until a party contests it
s-61	Seat figures are measured (Wilson upper bound on error) and adopted with the engine, never live (19.3)	implemented	One synthetic set of 44 questions; latency from a workstation

J Reference Tests

This appendix lists the tests that exercise the contract, by file and name, as the build read them. They are the core module's; the API worker's contract data, routes, decision providers, personhood issuers, anonymous voice, Groth16 verifier and replica; the attestor's; the head trainer's; the registry, escrow and command programs' Rust tests; and the EVM mirror's Foundry tests. They exercise the rules over stipulated seat answers. No test calls a live model. `verify_reference_examples.py` and `verify_document.py` run the paper's own checks: the fixtures are current, the manuscript assembles, the PDF is readable and its links resolve.

`packages/domain/core/src/contract/anonymity.test.ts` (7)

- an RFC pins every recognized kind's anonymity set, in its hashed record
- voices: an anonymous voice is its kind and its nullifier; kinds are read through both forms
- a claim acts only against the pinned root, in this RFC's scope, with the message its action requires
- messages are domain-separated by action, and a response's binds its digest and a fresh nonce
- the tree's leaves at an instant, and who is in a pinned tree: rotation never yields a second voice
- anonymous responses replace each other by nullifier, and count like any voice
- commitments are field elements; anonymous media must be stored by the contract

`packages/domain/core/src/contract/attestation.test.ts` (4)

- an attested seat's answer counts only with its attestor's valid signature over that exact answer
- an attested rendering keeps its signed text; a forged one is an error
- the replica recomputes a determination and catches every kind of edit
- the log is a hash chain: a rewritten, reordered or dropped entry shows; checkpoints must match the log

`packages/domain/core/src/contract/contract.test.ts` (54)

- the kernel is pinned: editing it makes a different contract
- canonical JSON sorts keys and refuses non-finite numbers
- one person, one voice: duplicate keys, unknown kinds, expired and revoked proofs are refused
- a replayed older issuance cannot undo a revocation tombstone
- genesis law is admissible; kernel floors, human removal and duties without elements are not
- an int8 MLP head is pinned by hash and reproduces bit for bit
- the kernel requires two operators and an open-weight seat per tier
- a seat of a router names the one host that serves it, or it cannot be admitted
- an engine without two disjoint top-tier panels cannot be admitted: a top-tier case must stay appealable (k7)
- every admitted media kind needs two independent renderers (k10)
- the genesis engine and law satisfy the kernel, with renderers for every admitted medium
- seats of one operator or one lineage are one group
- panels: exactly quorum groups, deterministic by seed; parties (by operator or controller) and used seats never sit
- three seats of one operator cannot outvote two independent seats; a tie has no winner
- a decisive classifier tier stops the climb; the deliberative tier is never asked
- a divided top tier is unresolved — never decided by label order
- a top tier below its band is unresolved
- seat errors are recorded; too few answering groups leaves the question unavailable, never guessed
- a label outside the closed set, or an answer over a different record, is an error — not a vote
- duplicate evidence ids are refused: every item is addressable
- fixed panels: an instance's questions are answered by the same seats
- tier 1 answers from a pinned head when it is confident
- a determination is content-addressed
- media reaches the seats as text from two independent renderers; an unrenderable item is marked, not dropped
- opening an RFC adds the status quo, pins engine, state and kinds, and refuses kernel violations
- two options with the same effect are refused, whatever their titles
- only confirmed responses count; the latest confirmed per voice wins; kinds are those recognized at open
- the stance question reads the option's actual changes, not only its framing
- score voting: another option cannot move an option's score; a positive unique maximum is favored, ties keep the status quo
- each interpretation is contestable for the full window from when it became available
- concurrence reads a seeded sample, the same for anyone who recomputes it
- adoption needs the voice AND a confident concurrence; unresolved or withheld keeps the status quo
- an engine RFC: the outgoing engine first, then the proposed; the fallback only when the outgoing could not answer
- a rejected RFC cannot be reopened unchanged during the cooldown
- adopting a law RFC moves the state forward and keeps the law history; a superseded RFC is not adopted
- a law that admits a medium the engine cannot render is not adopted
- the scope of justice and the remedy bounds are checked before anyone is served
- a repealed duty still governs acts done while it was in force
- the clock starts when the respondent is served, not when the claim is filed
- deciding: justiciability, then every element separately, then figures and remedies — one fixed panel per instance

- final-offer valuation and per-harm remedy bounds: no harm is repaired twice, one penalty cap over the award
- criminal: every element at the top tier at the criminal standard
- a timely appeal survives a late sweep, is heard by a disjoint top-tier panel, and its judgment is final
- a reopening needs a final matter, the same parties, new evidence, and happens once
- the public projection carries the judgment, never the evidence or the record
- execution: cash, then bonds; victims first; loser pays both ways; unprocessable refunds everyone
- pool epochs: equal claims get equal shares whatever their arrival; unpaid needs persist to the next epoch
- the $10,000 cycle: mandate → facility → servicing → distribution → transfer → default → cases
- kernel floors are what the genesis law starts from
- one person's documents and providers agree on one voice key; too thin an identity gets none
- RFC effects compare resulting law, including reordered independent assignments
- criminal calibration is configuration-bound, expires, and uses a conservative precision bound
- an engine RFC cannot adopt a calibration whose report it does not carry
- a criminal calibration's digest binds its findings, so a report cannot claim other counts

packages/domain/core/src/contract/evm-abi.test.ts (3)

- the forge fixtures are what the core ABI builds and signs now
- the fixture adoptions are 181-byte registry messages, signed by seats of the sets they name
- mirror shapes keep raw byte order

packages/domain/core/src/contract/registry-abi.test.ts (13)

- constants, seeds and the program id come from the Rust
- the ContractRegistry offsets follow the struct's declaration order
- the adoption message is the Rust emission order, 181 bytes
- payload hashes equal the Rust goldens
- upgrades: both engines sign, then a seven-day timelock
- signer sets: two independent groups, a reachable quorum, no key in both tiers
- a registry account round-trips and frames are tiered by group quorum
- engine adoptions need both quorums; law adoptions the current one
- one precompile instruction carries many real signatures over one message
- an adoption input encodes to its Borsh size
- checkpoints: the PDA, the instruction data and the account match the Rust goldens
- the instructions that submit an adoption carry what the program reads
- the devnet build pins the genesis its public record hashes to

packages/domain/core/src/contract/routing.test.ts (16)

- deterministic arithmetic: expNeg matches e^-x without Math.exp; majority and ties
- a seeded weighted pick follows its weights and is the same for the same seed
- the soft preference: spread at low stakes, near certain at high stakes, and $\tau = 0$ is the argmin
- the plan: the top tier when the stakes pay for it, more groups when independence pays, the classifiers alone when they do not
- a deadline caps the path by p99: slow seats and tiers are left out, and an impossible deadline is honest
- a routed determination records its plan; the replica recomputes it; a changed plan or panel is caught
- an infeasible deadline is unavailable, never a guess
- a head goes in front when it is cheaper in expected loss and the procedure allows tier 1
- a case instance is routed by its stakes, recorded, and recomputed by the replica
- raising the stakes: either party, upward only, before the next instance, with a deposit — and it moves the plan
- expedited procedure: only civil agreement matters, on the law's windows, never below the kernel's expedited floors
- review is drawn by the determination's id, only for fast decided answers; measurement is conservative
- the genesis engine and law: every deciding seat has economics, Clef-flash sits with Clef, and the plan meets the hot deadline
- a seat's adopted calibration table lifts its report onto the tier's scale, and the record recomputes
- a calibration proposed from referenced answers is conservative, and a small sample proposes none
- a case on a facility counts the bonds a judgment would put in default in its stakes

deployments/platform/cloudflare/api-worker/src/data/contract.test.ts (23)

- genesis is written on first read and is the published state
- one person, one voice: a second account on the same identity has no voice; an agent key is not its owner
- an RFC: deposit held, receipts confirmed, each interpretation's own contest window, concurrence, adoption
- disagreement is never approval: a divided concurrence panel keeps the status quo
- an engine RFC: the proposed engine is consulted only after the outgoing concurs, through the candidate executor
- a case is served before its clock starts; only the parties read its record (k11)
- a case runs element by element; a timely appeal survives a late sweep; execution is once, atomic and loser-pays
- an element the engine cannot resolve is not established; one it cannot process is never a merits loss
- the same matter cannot be brought twice
- equal pool claims are paid equally, whatever order their judgments executed in; residuals persist
- the $10,000 cycle: mandate → facility → servicing → bond sale → default → judgments without uninsured pool coverage
- execution refuses a bond snapshot sold before commit; total units are conserved
- execution cannot consume a freeze returned between planning and commit
- repayment replans when bond ownership changes before commit
- default resumes holder filings after a crash, with stable case identities
- an unavailable appeal closes without enforcing the first judgment
- case panels use fresh host entropy rather than a claimant-controlled filing digest
- criminal filings require calibration rather than self-reported model confidence
- deliberative RFCs consider comments without being bound to a score; binding RFCs remain bound
- insured-1 collects a premium and requires independent attestation before a pool claim
- insured-1 refuses borrowers with common ownership
- insured-1 refuses parties that share a payout address or a phone number, though the ownership rows are silent
- a new kernel never silently interprets an existing contract ledger

deployments/platform/cloudflare/api-worker/src/data/contract-routing.test.ts (4)

- a party raises the stakes with a deposit; the instance is routed by them and recomputes
- an expedited agreement matter runs on the expedited windows
- an identical routed question returns the recorded determination
- a fast determination drawn for review is put to the top tier by fresh groups; the seats are measured from the record

deployments/platform/cloudflare/api-worker/src/data/contract-identity.test.ts (1)

- identity correction needs a judgment and accepted signature, waits for RFC close, and cannot replay

deployments/platform/cloudflare/api-worker/src/routes/contract.test.ts (12)

- the kernel, law and engine are public; the kernel hash is the pinned one
- a ledger written under another kernel refuses in its own words, and says which kernel it found
- writes need a principal; a person opens an RFC, answers it, and sees it listed
- the public figures count people, voices, RFCs and cases without naming anyone
- another account cannot confirm a response as its respondent
- an RFC that would breach the kernel is refused with the article it breaches
- someone without a personhood proof cannot respond
- a case's record is read by its parties only; its determinations are never public (k11)
- a program's token cannot commit the account's money to credit; an agent's key passes the gate as a person's does
- anonymous routes take no credential: a signed-in caller is just as anonymous, and only a proof speaks
- the seats' figures are public; raising a case's stakes needs a party
- a kind the law names is proven by its issuer's signature: one voice per lineage, and the law's issuer list is the whole authority

deployments/platform/cloudflare/api-worker/src/data/decision-providers/decision-providers.test.ts (14)

- openai: strict JSON schema with the labels as an enum; every item delivered under its id
- anthropic: the determination tool; a label outside the set is an error, never passed on
- typesafe: the state is keyed by unique item ids — two items labelled alike both arrive; a moved alias is refused
- openai-decisions: one choice over the labels, every item under its id; another model or a refusal is no vote
- workers-ai-systemone: Clef gets the System One wire through the AI binding; another model answering is refused
- one System One wire: a seat sends exactly the request the host's grammar builds, and reads the answer as the host does
- genesis: Clef, Clef-flash and Llama 3.1 8B are one group; routing draws which sits, the others are its fallbacks
- a credential never follows a seat's URL off its operator's hosts — not even a proposed engine's
- a proposed engine renders nothing: it never reads a private record before adoption
- media is fetched once and verified against the committed sha256
- availability: an unimplemented protocol, an unknown operator or a missing secret makes a seat unavailable
- a new decision provider plugs in by protocol name: no engine or contract change
- a router seat with an upstream is served by that host only, with no fallback
- a router seat on the decisions door is served by the host it names too, and a door that is not a router gets no preference

deployments/platform/cloudflare/api-worker/src/data/personhood-issuers/personhood-issuers.test.ts (5)

- a passport and a licence of one person are one voice; another person is another
- fail closed: no secret, too thin an identity, or an issuer the registry does not know records nothing
- a human's verification joins the transparency log: one entry per statement, by a digest, naming no one
- the sweep re-appends a personhood statement whose live append was lost, and re-verification still adds one
- reverification cannot replace an established voice when document identity changes

deployments/platform/cloudflare/api-worker/src/data/contract-anonymity.test.ts (1)

- anonymous voice, end to end with real proofs

deployments/platform/cloudflare/api-worker/src/data/semaphore.test.ts (3)

- the verification keys are Semaphore's own, byte for byte
- Semaphore's hash: keccak256 of the 32-byte value, shifted into the field
- real proofs: ours and snarkjs agree — valid, forged, every public input changed, garbage

deployments/platform/cloudflare/api-worker/src/data/contract-replica.test.ts (5)

- the replica agrees with an honest record: versions, the RFC, every determination, the log, the party's case
- the replica finds every edit: an answer, a tally, a decision, a log entry, a checkpoint, a finding, a version
- the log: appends are chained and idempotent; checkpoints record the head, anchored or not, with why
- reconciliation appends an act whose live append was lost
- a genesis withdrawn before anything happened stays logged, and the genesis that replaced it is logged too

deployments/platform/cloudflare/attestor/src/attestor.test.ts (4)

- an attested seat answers through its attestor; the signature verifies; the replica agrees
- a token never follows an attestor URL it was not issued for; an unserved model is refused
- an attested rendering: bytes checked by the attestor, the text signed and verified
- configuration refusals name what is wrong

scripts/contract/train-head.test.mjs (1)

- a separable task yields a sealed, valid, accurate integer head with a conservative calibration

deployments/platform/blockchain/solana-escrow/programs/contract/src/auth.rs (12)

- adoption message is 181 bytes golden
- signatures bind the decision record
- genesis is the pinned manifest and nothing else
- upgrades wait out the timelock
- payload goldens match the typescript mirror
- engine payload binds every field
- a signer set needs two independent groups and a reachable quorum
- quorum counts groups not keys
- engine changes need both the outgoing and the incoming engine
- one precompile instruction can carry many signers
- checkpoints only move forward per submitter
- a frame is a top tier determination only when the deliberative quorum signed

deployments/platform/blockchain/solana-escrow/programs/contract/src/lib.rs (7)

- account spaces match their borsh sizes
- the checkpoint pda and instruction data match the typescript mirror
- an adoption is bound to the state it was decided against
- the adoption address names its decision record
- payload of engine binds the incoming signer sets
- an upgrade commits to the engine its code makes
- a scheduled upgrade survives law changes but not engine changes

deployments/platform/blockchain/solana-escrow/programs/escrow/src/lib.rs (4)

- appeals climb tiers and the top tier is reheard once by fresh seats
- the standing direction is the last appeal or the default
- account space matches the borsh layout
- frame message is 139 bytes

deployments/platform/blockchain/solana-escrow/programs/command/src/lib.rs (11)

- the command message is 176 bytes and binds the registry state
- a registry quorum authorizes and a stranger does not
- quorum counts groups not keys
- a stale registry binding is refused

- delegation never widens
- revocation is terminal and blocks children
- optimistic finality opens a window
- account space matches its borsh size
- accept and record messages golden
- a command closes only after it ended and its hold passed
- production holds a closed command longer than one genesis case

`deployments/platform/blockchain/evm/test/ContractMirror.t.sol` (8)

- genesis is the manifest and nothing else
- the bytes are the type script and rust bytes
- a law adoption with a quorum applies
- below quorum is refused and one group counts once
- a forged signature reverts
- a stale adoption is refused
- an engine change needs both quorums
- an upgrade needs both engines and waits out the timelock

`deployments/platform/blockchain/evm/test/Ed25519.t.sol` (4)

- sha 5 1 2 matches fips vectors and every length the mirror hashes
- ed 2 5 5 1 9 matches rfc 8 0 3 2 random vectors and refuses forgeries
- an encoding that names no point is refused
- gas of one verification

`deployments/platform/blockchain/evm/test/MirrorEscrow.t.sol` (5)

- the frame bytes are the solana frame
- each tranche releases once on an engine quorum
- an older decision cannot replay after a newer one
- strangers and forgeries move nothing
- after expiry the remainder returns to the customer

In all, 221 tests in 24 files.

K Commitments and Decisions

K.1 The Author's Commitments

The author set these commitments for version 0.4. The design follows them.

1. RFCs, with trustless model-based interpretation of responses. No branches and no executive figures.
2. The engine — its providers and configuration — is part of the contract. The contract changes only by vote. Votes are interpreted by the contract's algorithmic part, deferring to the current engine, in the interest of the people.
3. A change of engine is interpreted by both the outgoing and the proposed engine, which must agree in the context of the contract's state.
4. People are a predicate: humans now, extensible to uploaded humans, AIs, animals, businesses and other participants.
5. No threshold voting as the default. Responses may be text, images or other unstructured data, interpreted rather than only counted.
6. A free market: politics has no power over the market; justice does. The scope of justice is objective, so victims are protected; insured bonds.
7. No jury ever, including in criminal matters. Automating litigation is the critical step to law, order and moral compliance.
8. Unlimited advocacy is accepted, in the light.
9. Tiers of fast classifiers, mixed-provider quorums, and pinned MLPs over embeddings that are part of the contract.

K.2 Design Decisions of Version 0.4.1

These decisions were made during the design and implementation of versions 0.4 and 0.4.1. A later version can change them.

ID	Decision	Reason
d-01	Four parts (kernel, law, engine, ledger); determinations as the only act	One mechanism replaces branches; independence comes from seats
d-02	Per-kind uniqueness rules; voice = kind:key; earliest proof holds it	Extensible people without one person becoming many
d-03	Six fixed stance labels scored $-2 \dots +2$; score voting; no quorum	Interpretation without counting thresholds; an option's score independent of the other options
d-04	Concurrence is concur or withhold, decided at the band; grounds classified separately	The engine guards the kernel; a divided engine does not concur
d-05	Contest of one's own interpretation, once, on fresh top-tier groups, within a window from availability	Errors correctable by the person they affect, whatever the interpreter's speed
d-06	Market freedom as a kernel article; duties must be harm-based and general	"Politics has no power over the market" made checkable
d-07	Objective justiciability: claimant, respondent, act, duty, harm	Protects victims and blocks political misuse of courts
d-08	Every question has closed labels	Fits classifier seats and heads; no free text that can grant power
d-09	Tier confidence = winning weight ÷ answering groups; a tie has no winner	Disagreement and uncertainty both lower it; label order decides nothing
d-10	Seats grouped by operator or lineage	Three seats of one provider count once
d-11	At least two operators and one open-weight seat per model tier	No provider can change or stop the contract
d-12	MLP heads integer-only with calibration tables	Reproducible on every machine
d-13	Amounts by final offer between the parties' figures per harm item	Closed labels for a numeric question, without the claimant's framing deciding it
d-14	Criminal: top tier and ≥ 950/1000; civil: more likely than not	Graded standards without juries
d-15	Two disjoint top-tier panels; the appeal is a stage; an appeal waits for a fresh panel	Every top-tier decision appealable; no clock overtakes an appeal
d-16	Judgments reach bonds; politics cannot	Owner commitment 6; restores deterrence
d-17	Protection pool from bond premiums and half of penalties; allocation epochs; subrogation	Victims paid in equal proportion as the pool is funded
d-18	Loser pays engine costs both ways; deposits as far as a balance allows; unprocessable costs to the commons	No free filing spam; no one turned away for want of money

(continued)

ID	Decision	Reason
d-19	RFC deposit to the commons unless someone besides the opener supported a change	Spam costs its opener, not the commons
d-20	On-chain registry holds upgrade authority; engine and code changes signed by both engines, code behind a timelock; genesis manifest	No key above the contract
d-21	Media rendered to text by two independent renderers; seats read one text package	One identical record; media never breaks a quorum
d-22	Panels of exactly a quorum, drawn by public seed, fixed per instance	Capacity left for the appeal; no one chooses a panel
d-23	Duty elements decided one by one; remedies justified one by one	A liability finding that cannot skip the hard element
d-24	RFCs and cases pinned to the version they began under	Predictable procedure; superseded RFCs not adopted
d-25	Every money movement one op-claimed conditional batch	Atomic, single-winner, idempotent
d-26	The contract services credit (mandate, facility, servicing, transfer, default)	The economic thesis implemented, not only described
d-27	Anonymous voice by Semaphore proofs over a tree pinned at opening; a person in the pinned tree answers only anonymously	Article k2 needs one voice by a deliberate act, never a name; one count across routes
d-28	Seats may name attestors that sign each answer; unattested answers are marked	A record checkable without trusting whoever ran the sweep
d-29	A public hash-chained log of digests, its heads anchored on the registry	A history that cannot be rewritten or forked unseen
d-30	One trust root: the command notary and every mirror read the registry's seats, with no committee of their own; Ed25519 verified on EVM chains	No second attestor set to diverge; gas paid for it

L Glossary

Term	Meaning
Kernel	The contract's unchangeable articles, floors and protected interests, identified by their hash.
Law	The recognized kinds of people, the duties and the parameters; changed only by adopted RFC.
Engine	The pinned seats, heads, quorums and bands through which the contract decides; changed only by adopted RFC.
Ledger	Accounts, agreements, RFCs, responses, cases, judgments and determinations.
Determination	A closed question over committed inputs, answered tier by tier and recorded with every seat's answer; the contract's only act.
Tier	One level of the engine: structural, embedding, classifier, deliberative.
Seat	One pinned model run by an identified operator.
Pin	The SHA-256 of open weights, or the exact version ID of a hosted model.
Group	Seats sharing an operator or a lineage, counted as one in aggregation.
Band	The confidence at which a tier's answer is decisive.
Head	An INT8 MLP over embeddings, pinned in the engine, answering one question kind.
Entity	Anything that can hold accounts, interests or duties.
Person	An entity of a recognized kind with a valid personhood proof.
Voice	A person's one unit of say in an RFC, identified by kind and uniqueness key.
RFC	A request for comments that changes law or engine if adopted.
Response	A person's own submission to an RFC, in any modality.
Commitment	The public commitment of a person's Semaphore identity, registered once per voice; a leaf of its kind's tree.
Nullifier	The hash of an identity's secret and an RFC's scope: the same for every proof of one identity in one RFC, unlinkable elsewhere; an anonymous voice.
Attestor	A service that runs a seat with its own credential and key and signs each answer it returns.
Transparency log	The public, hash-chained sequence of every act of the contract, carrying digests, never records or names.
Checkpoint	A log head anchored on the registry by a published submitter key.
Replica	The program that recomputes the public record and reports what does not match.
Mirror	A contract on another chain that follows the registry's adoptions with the same seat keys.
Interpretation	The engine's reading of a response into a score for each option.
Receipt	A response's identifier and content digest; the response counts once its author confirms the digest.
Panel	Exactly a quorum of independent groups that answer a tier's questions; fixed per case instance.
Unresolved	The outcome of a top tier that tied or fell below its band.
Renderer	A seat that turns one media item into text; two independent renderers per item.
Package	The canonical text record a question is asked over, each item under a unique identifier.
Instance	One round of deciding a case on its fixed panels; the appeal is the second.
Harm item	One alleged harm with its interest and amount; remedies answer harm items.
Op claim	The first statement of a batch, conditional on the state read; every other statement is gated on it.

(continued)

Term	Meaning
Mandate	An investor's escrowed capital with terms on which facilities may be drawn.
Facility	A borrower's drawing on a mandate, with its fixed schedule and its bond.
Concurrence	The engine's determination that an option favored by the voice may be adopted.
Duty	A harm-based rule of law, or an agreed term, or a kernel article, that a respondent may breach.
Justiciable	Meeting the five-element scope of k6.
Judgment	The outcome of a case instance: liable, not liable or dismissed, with each element's finding, each harm's award and the remedies.
Protection pool	The fund that pays victims' uncollected judgments and takes over their claims.
Commons	The contract's own account, funded by the commons fee and forfeited deposits.
Registry	The on-chain account holding the state hashes, seat keys, genesis manifest, last decision hash and upgrade authority.

M Sources and Build

M.1 Sources

The paper's contract rules are the reference module in this repository. The findings about the predecessor come from a static review of the predecessor at a pinned commit and of its runbooks. They are history. Nobody re-executed them for this version. Version 0.4.1 answers an external read-only review of version 0.4 at commit `95a5632b`. The review's findings are summarized in Section 18.4. Each finding was reproduced as a test, not taken on assertion. Version 0.5 answers the owner's questions of 3 October 2026 (Section 18.3). Its anonymous voice follows the Semaphore protocol [10]. Its gas figures are the EVM mirror's own test measurements. The legal and financial references illustrate the design. They are not a jurisdictional opinion. Version 0.3's philosophical excerpts and its 83-item review map applied to that version's profile. This version does not carry them forward.

M.2 Build

`generate_fixtures.mjs` runs the reference module and writes `fixtures/examples.json`. `scripts/contract/canary.mjs --json` writes the canary runs (`fixtures/canary-*.json`). `build.py` assembles the manuscript with those values and with generated tables: the kernel articles, the genesis duties and seats, the status register, the threat register and the threats against what is built, the canary runs and the test list. It makes one standalone source per edition — this paper and its short edition — and compiles each with Tectonic. `build.py --check-current` fails when either PDF no longer matches its inputs. The repository's checks run it. Thus a change to the contract core or to a cited test cannot ship with a stale paper. The build manifest records the maintained-input digest, the standalone source hash, the PDF hash, the extracted-text hash and the toolchain. The PDF's keywords carry the input digest and a unique build invocation. `verify_reference_examples.py` checks that the fixtures match a fresh run of the module, and runs the module's tests. The build's input digest covers the core module and every cited test file. `verify_document.py` checks the assembled source, the compilation log and the PDF. These checks establish that the document matches its inputs and the module. They do not establish that the contract is correct, safe or lawful.

N The Live Canary

Everything else in this paper runs the contract over stipulated seat answers. The canary is the one place where a live model answers. `scripts/contract/canary.mjs` asks every seat of the genesis engine one question whose answer is known ("two plus two equals four, and water at sea level boils at 100 degrees Celsius" — `true`). It embeds one sentence with the embedding seat. It has every renderer render one image (a red square) and one spoken sentence ("the receipt shows five hundred dollars, paid on the first of September"). It uses the hosted service's own adapters and a deployment's credentials. Its `--json` output is the source of the tables below. The build reads them from `fixtures/canary-*.json`. The tables report each run as it happened, failures included.

Dev credentials, 2026-10-14T05:00:39.088Z. 16 of 16 answered; engine `genesis`, hash `24edc06a9c3dfa9f…`

Seat	Role	ms	Answer or error
`bge-base-en-v1.5`	embedding	678	768 dims
`jev-1.13.0`	classifier	213	true @ 960
`clef`	classifier	300	true @ 979
`clef-flash`	classifier	264	true @ 788
`llama-3.1-8b`	classifier	613	true @ 1000
`gpt-5.6-luna`	classifier	2173	true @ 1000
`claude-opus-5-5`	deliberative	2118	true @ 985
`gpt-5.6-sol`	deliberative	2839	true @ 999
`deepseek-v4-pro`	deliberative	1370	true @ 1000
`llama-3.3-70b`	deliberative	2533	true @ 1000
`qwen3.8-max`	deliberative	9804	true @ 1000
`kimi-k2.6`	deliberative	16630	true @ 1000
`render-gpt-5.6-luna`	renderer (image)	2540	The image shows a solid red square on a plain white background. The square is positioned approximately in the center, sl
`render-claude-haiku-4-5`	renderer (image)	1851	# Description of Item This is a red square displayed against a white or transparent background. The square appears to be
`render-whisper-large-v3-turbo`	renderer (audio)	3214	The receipt shows $500, paid on the 1st of September.

(continued)

Seat	Role	ms	Answer or error
`render-gpt-4o-mini-transcribe`	renderer (audio)	1769	The receipt shows $500, paid on the 1st of September.

Prod credentials, 2026-10-14T05:01:53.851Z. 16 of 16 answered; engine `genesis`, hash `24edc06a9c3dfa9f…`

Seat	Role	ms	Answer or error
`bge-base-en-v1.5`	embedding	538	768 dims
`jev-1.13.0`	classifier	191	true @ 960
`clef`	classifier	431	true @ 979
`clef-flash`	classifier	228	true @ 788
`llama-3.1-8b`	classifier	681	true @ 1000
`gpt-5.6-luna`	classifier	2428	true @ 998
`claude-opus-5-5`	deliberative	2478	true @ 985
`gpt-5.6-sol`	deliberative	1593	true @ 999
`deepseek-v4-pro`	deliberative	782	true @ 999
`llama-3.3-70b`	deliberative	1372	true @ 1000
`qwen3.8-max`	deliberative	9478	true @ 1000
`kimi-k2.6`	deliberative	20746	true @ 1000
`render-gpt-5.6-luna`	renderer (image)	2329	The image shows a solid bright red square on a white background. The square is positioned slightly above and to the left
`render-claude-haiku-4-5`	renderer (image)	2636	# Description of Image **Visual Content:** A solid red square positioned on a white background. The square appears to be
`render-whisper-large-v3-turbo`	renderer (audio)	4291	The receipt shows $500, paid on the 1st of September.
`render-gpt-4o-mini-transcribe`	renderer (audio)	1958	The receipt shows $500, paid on the 1st of September.

What a run shows. A run shows three things for each seat: the seat was reachable as pinned, it answered inside its label set, and — for these questions — it answered right. It also shows that each renderer produced the content of the media it was given. **What it does not.** A canary is one question per seat. It measures neither accuracy on contract questions nor calibration (Section 19). A seat that answered once can fail the next call. A failed row above is a seat that did not return an answer inside its label set on that call. The contract records such a result as an error answer, never as a vote.

References

[1] Douglass C. North. *Institutions*. Journal of Economic Perspectives 5(1), 97–112, 1991. doi: 10.1257/jep.5.1.97.

[2] Bank for International Settlements. *Blueprint for the Future Monetary System: Improving the Old, Enabling the New*. Annual Economic Report, Chapter III, 2023. Publisher's chapter.

[3] Federal Reserve Board. *What Is Forward Guidance, and How Is It Used in the Federal Reserve's Monetary Policy?* Updated 22 August 2025. Official explanation.

[4] OECD. *Regulating for People*. In *OECD Regulatory Policy Outlook 2025*. 2025. Publisher's chapter.

[5] Scott R. Baker, Nicholas Bloom, and Steven J. Davis. *Measuring Economic Policy Uncertainty*. Quarterly Journal of Economics 131(4), 1593–1636, 2016. Publisher's article.

[6] Michelle W. Bowman. *Forward Guidance as a Monetary Policy Tool*. Federal Reserve speech, 12 October 2022. Official transcript.

[7] Investor commentary on startup personnel, defensibility and mission continuity, supplied by the author on 30 September 2026. Used as a situated motivating perspective; public attribution was not independently verified.

[8] Internal Revenue Service. *Topic No. 427, Stock Options*. Official tax topic. Accessed 30 September 2026.

[9] Electronic Code of Federal Regulations. *26 CFR § 1.409A-1: Definitions and Covered Plans*, including stock-right provisions in paragraph (b)(5). Regulatory text. Accessed 30 September 2026.

[10] Privacy and Scaling Explorations. *Semaphore V4: Zero-Knowledge Group Membership and Signaling*. Protocol documentation and reference implementation. Accessed 3 October 2026.

[11] S. Josefsson and I. Liusvaara. *Edwards-Curve Digital Signature Algorithm (EdDSA)*. RFC 8032, Internet Engineering Task Force, 2017. rfc-editor.org.

[12] National Institute of Standards and Technology. *Security Objectives*. Voting Research Program. Official security objectives. Accessed 30 September 2026.

[13] National Institute of Standards and Technology. *Security Recommendations*. Voting Research Program. Official recommendations. Accessed 30 September 2026.

[14] Coalition for Content Provenance and Authenticity. *Content Credentials: C2PA Technical Specification*, version 2.2. Versioned specification. Accessed 30 September 2026.

[15] National Institute of Standards and Technology. *Artificial Intelligence Risk Management Framework: Generative Artificial Intelligence Profile*. NIST AI 600-1, 2024. Publication record.

[16] U.S. Department of the Treasury, TreasuryDirect. *Treasury Bonds*. Instrument description. Accessed 30 September 2026.

[17] U.S. Department of the Treasury, TreasuryDirect. *Floating Rate Notes*. Instrument description. Accessed 30 September 2026.

[18] U.S. Department of the Treasury, TreasuryDirect. *About Auctions*. Auction procedures. Accessed 30 September 2026.

[19] Circle. *USDC Terms*. Issuer's terms. Accessed 30 September 2026. These terms describe contractual dependencies; they are not a guarantee against issuer, redemption, or settlement risk.

[20] Electronic Code of Federal Regulations. *12 CFR § 1002.9: Notifications*, Regulation B. Regulatory text. Accessed 30 September 2026.

[21] U.S. Bureau of Economic Analysis. *Value Added*. Glossary. Official definition. Accessed 30 September 2026.

[22] U.S. Bureau of Economic Analysis. *Gross Domestic Product (GDP)*. Glossary. Official definition. Accessed 30 September 2026.

[23] David Manheim and Scott Garrabrant. *Categorizing Variants of Goodhart's Law*. 2018. arXiv:1803.04585.

[24] Solana. *Accounts*. Core documentation. Official documentation. Accessed 30 September 2026.

[25] Solana. *Program Derived Address*. Core documentation. Official documentation. Accessed 30 September 2026.

[26] Solana. *Deploying Programs*. Developer documentation. Official documentation. Accessed 30 September 2026.

[27] U.S. Securities and Exchange Commission. *SEC Issues Investigative Report Concluding DAO Tokens, a Digital Asset, Were Securities*. Release 2017-131, 25 July 2017. Official release. A historical application, not a general conclusion about every proposed instrument.

[28] U.S. Securities and Exchange Commission. *Exempt Offerings*. Small-business resources. Official resource. Accessed 30 September 2026.

[29] U.S. Securities and Exchange Commission. *Private Funds*. Capital-raising resources. Official resource. Accessed 30 September 2026.

[30] CommandAGI. *governance.ts*, `packages/domain/core/src/governance.ts`, commit `a3295213e0f3fbcdb6200ac129bfd609ed71b690`. Pinned source; selected source reviewed 30 September 2026.

[31] CommandAGI. *arbiter.ts*, `deployments/platform/cloudflare/api-worker/src/data/arbiter.ts`, same pinned commit. Pinned source; selected source reviewed 30 September 2026.

[32] CommandAGI. *Solana Escrow Program*, `deployments/platform/blockchain/solana-escrow/programs/escrow/src/lib.rs`, same pinned commit. Pinned source; selected frame and initialization paths reviewed 30 September 2026.

[33] CommandAGI. *Devnet Runbook*, `.old/docs/components/platform/blockchain/solana-escrow/DEVNET.md`, same pinned commit. Pinned runbook. Documents July 2026 events; not a fresh chain audit.

[34] CommandAGI. *Solana Escrow README*, `.old/docs/components/platform/blockchain/solana-escrow/README.md`, same pinned commit. Pinned documentation; transaction-size, expiry, committee and deployment-gate descriptions.

[35] United States. *9 U.S.C. § 10: Vacation; Grounds; Rehearing*. Statutory text reproduced by Cornell Legal Information Institute. Section 10. Consulted 30 September 2026; does not resolve the proposed system's enforceability.

[36] U.S. Treasury, Office of Foreign Assets Control. *FAQ 401: Aggregate Ownership and the 50 Percent Rule*. Official FAQ. Consulted 30 September 2026.

[37] U.S. Treasury, Office of Foreign Assets Control. *FAQ 398: Control and the 50 Percent Rule*. Official FAQ. Consulted 30 September 2026.

[38] U.S. Treasury, Office of Foreign Assets Control. *FAQ 560: Sanctions Obligations and Digital Currency*. Official FAQ. Consulted 30 September 2026.

[39] U.S. Department of the Treasury. *Treasury Prohibits Transactions with Central Bank of Russia and Imposes Sanctions on Key Sources of Russia's Wealth*. 28 February 2022. Official historical release.

[40] International Monetary Fund. *Dollar Dominance in the International Reserve System: An Update*. 11 June 2024. Official analysis. Reserve-share evidence is distinct from a currency risk-premium estimate.

[41] Typesafe AI. *Introducing System One Models and Jev*. Company blog. Accessed 1 October 2026. Vendor description of typed-output models with calibrated probabilities; claims not independently verified.

[42] Cloudflare. *Introducing Clef: Our Open-Source Decision Models, and New RL Fine-Tuning Platform*. Company blog, 1 October 2026. Accessed 3 October 2026. Vendor description of Jev-compatible open-weight decision models; benchmark and calibration claims not independently verified.