

Working paper — academic draft

Not peer-reviewed and not journal-published. Draft v5 (2026). Content may change. © José Luis D'Andrade · IdentIA / Self-Improving Society. **All rights reserved.** Brief citation with attribution is permitted. **Reproduction, translation, adaptation, or reuse**, in whole or in part (including commercial use), is **prohibited** without the **copyright holder's prior written permission**.

Persistent Agent Institutions as a Deployment Unit: Identity Invariants, Governance-as-Code, and a Measured Reliability Gate

Author: José Luis D'Andrade **Affiliation:** LegalDesign.DO **Project:** IdentIA — identia.do **Contact:** www.linkedin.com/in/josedandrade **Draft:** v5.0 (systems) — extended technical report. Splits [paper-v4.md](#) (retired 2026-08-27; in git history). **Companion position paper:** [paper-v5-position.md](#) — *The Selection Ladder: An Ordering Constraint for Bounded Recursive Self-Improvement*. Cited below as **[PB]**. **Date:** August 2026 · **Revised:** 2026-09-28 (retraction, §7.3) · **Status measured against:** the consolidated repository tree of 2026-08-23, commit `90792f5`, except where a later date is stated — see Appendix C **Companion:** [whitepaper/whitepaper-v5.md](#) (commercial positioning), [paper-v5-systems-es.md](#) (Spanish)

Main-text-only version, generated 2026-09-28 from `paper-v5-systems.md`: no acknowledgements, AI statement, references or appendices. The `[n]` markers and the mentions of Appendices C/D/E point to the full file. Not committed.

Abstract

Most large-language-model (LLM) multi-agent systems treat agents as *ephemeral processes*: instantiated for a task, used, and discarded. This is a defensible engineering choice, but it forces every hard question about identity, continuity and collective truth to be improvised at the moment it arises. We present **Self-Improving Society (SIS)**, a system whose unit of deployment is not an agent but a **sovereign organizational unit** — a department, a ministry, a school grade, a firm — complete with persistent identity, private and public memory, a local task queue, governance, and the ability to federate with other units.

This paper reports the **built and measured** half of that system. First, **IdentIA**, an engineering ontology that reads agent identity, change and truth through three named principles — *Dynamic Striving* (identity is updated by practice), *Absolute Continuity* (identity persists under change of model/runtime: a **persistent state** distinct from its **transient computational realization**), and *Emergent Pragmatism* (what an institution may treat as knowledge is what it can collectively validate — *institutionally warranted belief*, not truth) — and that is operationalized as **enforceable code invariants** (archive-not-delete, supersede-not-overwrite, no unilateral truth). Second, the deployed **architecture**: sovereign `agents-host` units, an "exoharness" social layer that governs and connects an interchangeable agent runtime rather than reimplementing it, and sovereignty-preserving federation. Third, the

implemented rungs of a measured-selection ladder — Beta–Bernoulli reputation with explicit uncertainty, Bradley–Terry/TrueSkill relative rating, and an Item-Response-Theory **reliability gate** — with a dated live result. Fourth, **governance-as-code** (L1–L8) and a controlled **evaluation protocol** built on a deterministic "gym" of canonical tasks that manufactures the head-to-head density the estimators require.

The upper rungs of the ladder, the argument for why it is *ordered* the way it is, and the normative thesis about institutional power are deliberately **not** in this paper. They have not run, and mixing an argument that has not been tested with a system that has been measured is how a systems paper acquires claims it cannot support. They are the subject of the companion position paper [PB].

What this paper does and does not demonstrate. It reports **infrastructure for bounded self-improvement**, not self-improvement. F0–F2 are implemented; F3–F4 run in shadow; F5 is design-only. The closed loop — variation, selection, mutation, improved subsequent behaviour — has **not** operated.

A retracted result, and why it is here rather than removed. Earlier drafts of this paper reported that the single dated live F2 run returned `converged=False`, and read that refusal as the reliability gate declining to act on its own unreliable signal. **§7.3 retracts that reading.** Re-measurement on 2026-09-26 found that the convergence flag could not become `True` under the criterion we were using, whatever the data: the per-iteration change was measured before an identifiability step that partly undoes it. A flag that cannot become `True` carries no information, so the run demonstrated nothing about the gate. We keep the episode in the paper because the retraction is the method working: the ledger row carrying that figure also carried the limit *not re-measured since*, and that limit is what produced the re-measurement. Of the 21 components tracked in §7.4, 9 are implemented and tested, 6 are implemented but bind nothing, 2 carry a named defect, and 4 are design-only.

Keywords: multi-agent systems; agent identity and persistence; item response theory; LLM-as-judge; AI governance; agent societies; federation.

1. Introduction

1.1 Motivation

The way a system organizes information determines the way it organizes power. For three years the dominant pattern in LLM agents has been the *orchestrator*: a controller spins up role-specialized agents, routes messages between them, collects an output, and tears the agents down. Frameworks such as AutoGen [18], MetaGPT [17], ChatDev [19], CrewAI and LangGraph have made this pattern productive and popular. They share a tacit ontology: the agent is a function call with a system prompt; its "memory" is a buffer or a shared vector store; its identity is a session id.

This ontology is adequate for one-shot automation and increasingly inadequate for the systems people now want to build: **persistent organizations of agents** that accumulate competence, hold a verifiable institutional memory, survive model upgrades, and can be *trusted* and *audited* the way a regulated department is. When the unit of value is an institution rather than a task, the discarded-process ontology fails exactly where it matters — at

questions of identity ("is this the same certified agent after we changed its model?"), of continuity ("what happens to everything this agent learned when we retire it?"), and of collective truth ("two agents produced contradictory findings; which is authoritative?").

1.2 Problem statement

We study the design of **persistent, sovereign, verifiable agent societies**. Three sub-problems recur and resist the orchestrator ontology:

1. **Identity under measurement change.** An agent's behavior is produced by a model that will be swapped, fine-tuned, or re-hosted. What, then, *is* the agent across these changes, in a way precise enough to satisfy an auditor?
2. **Institutional knowledge without a silent winner.** In a society, no single agent's assertion is *ipso facto* warranted. The system needs a principled way to let **institutionally warranted belief** emerge from collective validation, and to record supersession without erasing lineage.
3. **Measured self-improvement without unbounded risk.** A society that authors its own skills and edits its own scaffolding can drift or game any naïve reward. How can selection pressure be introduced *gradually*, gated on the system's own ability to measure improvement reliably?

1.3 Contributions

1. **IdentIA**, an engineering ontology (§3) that names three principles — *Dynamic Striving* (A), *Absolute Continuity* (B), *Emergent Pragmatism* (C) — and, crucially, translates them into **code-level invariants** that are enforced by governance rather than merely asserted in prose. **We introduce the identity tuple** $\Psi(p) = \langle \text{SOUL}, M, L, K \rangle$ and the **measurement** $m_t(p) = \langle R_t, \theta_t \rangle$ (Definitions 1 and 2, §3.1): identity as a persistent object formally separated from its transient realization, with the verifiable consequence that a change of model or runtime does not change Ψ . The persistent-state/transient-realization distinction is thereby given a precise reading as a tie-breaking criterion for design.
2. A **system architecture** (§4) whose deployment and scaling unit is the *sovereign organizational unit* (agents-host), wrapped by an "exoharness" social layer that *governs and connects* an interchangeable agent runtime rather than reimplementing it, with sovereignty-preserving **federation**.
3. **The implemented rungs, and a retraction** (§5–§7). We report the implemented rungs of the measured-selection ladder — Beta–Bernoulli reputation under Wilson lower bounds, Bradley–Terry/TrueSkill relative rating with an anti-position-bias pairwise judge, and Item-Response-Theory difficulty adjustment — together with the one dated live exercise of the IRT rung (§7.2) **and the retraction of what we previously concluded from it** (§7.3). We had reported a reliability gate that refused; re-measurement showed the refusal was a property of our convergence criterion rather than of the data. **No live result currently establishes that the gate withholds activation**, and we state what replaced that claim: a defect found by the paper's own bookkeeping. The *ordering argument* that makes this gate a precondition rather than a nicety is developed in [PB], and it is unaffected — it was never an empirical claim.
4. **Governance-as-code** (§6) — an eight-level scheme (L1–L8) mapping risk to approval mode, with per-goal budgets and quarantine — together with a **controlled evaluation protocol** (§7) centered on a deterministic *gym*, and a **claims ledger** (Appendix C) in which every status claim carries its source, method, date, environment and limit. The ledger exists because the same fact had been recorded three different ways across three documents; a volatile figure that is transcribed rather than derived will drift.

1.4 Document structure

§2 surveys the frontier and positions SIS. §3 formalizes IdentIA. §4 presents the architecture. §5 gives the implemented rungs and the reliability gate. §6 covers governance. §7 gives the evaluation protocol, the live F2 result, and implementation status. §8 lists threats to validity. §9 discusses interpretation, limitations and ethics. §10 concludes. Appendices give a glossary (A), one item of the gym suite and why it matters (B), the claims ledger (C), the identity formalism's human analogy and behavior map, which §3.1 summarizes and refers to rather than carrying inline (D), and the machine-checked (Lean 4) versions of the definitions and rules (E).

2. Related Work

We group prior art by the dimension along which SIS engages it. Our claim is not that any single mechanism is unprecedented — several are deliberate adoptions of state-of-the-art estimators — but that their *integration under one identity ontology* is new. We are explicit about overlaps.

2.1 Generative-agent societies and civilization simulations

Park et al.'s **Generative Agents** [1] populated a sandbox ("Smallville") with 25 LLM agents equipped with a *memory stream*, a retrieval scoring function, and periodic *reflection*, producing believable individual and emergent group behavior. Altera's **Project Sid** [2] scaled to 10–1000+ agents in Minecraft under the **PIANO** architecture (Parallel Information Aggregation via Neural Orchestration), observing emergent role specialization, collective rule change, and cultural/religious transmission. CAMEL's **OASIS** [3] simulates up to one million agents on social-media dynamics.

These systems study **emergent behavior** as a scientific object. SIS differs in *purpose and unit*: its agents-host is a **productive, sovereign deployment unit** (Agent-as-a-Service) running *audited* pipelines with artifacts and provenance, federating with other units via structured envelopes (§4.4). Cultural transmission in SIS is not an emergent curiosity but a **governed bridge** (the A → C bridge, §3.4) with human gates. SIS borrows the insight that distilled, attributed records ("minutes," not raw transcripts) are the right substrate for social memory.

2.2 Self-improving and self-designing agents

*Surveyed in the companion position paper, [PB] §2, where the comparison does work: SIS's relationship to Voyager, the Darwin Gödel Machine, PromptBreeder and the self-designing-agent line is a claim about **ordering** — what a system must measure before it is allowed to edit itself — and that is [PB]'s subject. The one fact this paper needs from it: none of those systems gates self-modification on a measured reliability property of its own fitness signal, which is what §5.6 and §7.2 report.*

2.3 Agent memory and identity

MemGPT/Letta [9] introduced OS-inspired hierarchical memory to give agents persistence beyond the context window. Generative Agents' memory stream [1] is the other influential design. The *question of identity itself* has recently become an explicit research topic: position work arguing that LM agents are not "normal" agents

because the pathologies of the underlying model — statelessness, stochasticity, prompt sensitivity — destabilise their identifiability, continuity, persistence and consistency [10], and early **agent-identity evaluations** [11].

SIS contributes **IdentIA** (§3): identity modeled as a **persistent state** (SOUL anchor, memory, lineage, skill set) distinct from its **transient computational realization** (active runtime and model), operationalized into invariants — archive-not-delete, supersede-not-erase, no-unilateral-truth — and a two-plane memory model (private A/B vs. social C) with a governed transmission bridge. We are not aware of prior work that converts an identity ontology into *enforced architectural invariants* in this way; the underlying *question* (persistence/continuity), however, is now actively studied [10, 11], which we treat as convergent evidence of timeliness rather than as prior implementation of our mechanism. Synergy [31] now elevates identity-and-personhood to an explicit architectural requirement (profile, typed memory, agenda, persistent social relationships) — further convergent evidence; IdentIA differs in making identity *enforceable* (invariants, §3.2) and institutional (the society, not the individual agent, is the deployment unit).

2.4 Selection, rating and quality-diversity

The estimator lineage — Bradley–Terry, TrueSkill, Item Response Theory, Thompson sampling, NSGA-II, MAP-Elites — is surveyed in [PB] §2. Each estimator used here is borrowed unchanged from that literature and is cited at the point of use in §5. This paper claims nothing novel about any of them; the claim is about what they are wired to gate.

2.5 Governance and alignment

Constitutional AI [25] governs *model behavior* at training time via AI feedback against a written constitution. SIS's governance is *operational and institutional*: an eight-level scheme over a running society — controlling infrastructure, base templates, agent spawning, marketplace promotion, and the runtime pin — with per-goal budgets and quarantine (§6). These are complementary layers (training-time alignment vs. run-time institutional control).

2.6 Positioning summary

Dimension	Representative prior art	SIS position
Agent societies	Generative Agents [1], Project Sid [2], OASIS [3]	Sovereign productive unit; audited pipelines; governed cultural bridge
Self-improvement	Voyager [4], ADAS [5], DGM [6], AlphaEvolve [7]	Safety-ordered ladder ; selection made reliable (and <i>measured</i> reliable) before it acts
Memory / identity	MemGPT [9], [10], [11]	IdentIA ontology as enforced code invariants
Selection estimators	BT [12], TrueSkill [13], IRT [20], MAP-Elites [14]	Adopted; assigned per object + self-triggered progression , no third-party deps
Governance	Constitutional AI [25]	Operational L1–L8 ; metacentrality thesis ([PB] §5)
Orchestration frameworks	AutoGen [18], MetaGPT [17], ChatDev [19]	Exoharness : govern/connect a runtime, not reimplement it

Dimension	Representative prior art	SIS position
Agentic-web citizenship	Synergy [31]	Institution as the unit of citizenship: reliability-gated selection; validation-gated transmission; policy-gated federation
Harness self-rewriting	Meta-Harness [32], Continual Harness [33]	The harness is wrapped, not rewritten: the institution carries identity and judgment; the runtime beneath is replaceable

2.7 Four poles of the adjacent field, and what none of them gates

The nearest neighbours to this work cluster into four poles. Naming them precisely matters, because what we claim is not a mechanism any of them lacks — several have better ones — but a **gate none of them applies**.

Pole 1 — academic multi-agent evolution. Synergy [31] maps almost one-to-one onto the IdentIA principles: collaboration, identity, and lifelong evolution. Its limits are methodological rather than architectural: confidence is weighted softly rather than gated, adjudication is by LLM judge, and evaluation is benchmark-only. Nothing in that design refuses to act on an unreliable signal; it down-weights it.

Pole 2 — permissionless composition. Networks in which capability spreads by adoption: an agent or artifact propagates because others take it up, and economic uptake substitutes for measured selection. This is principle C — collective truth-making — with the validation step removed. It is the useful contra-case for this paper, because it shows what a selection process looks like when nothing gates it: adoption is fast, and a claim's survival carries no information about whether it was ever correct.

Pole 3 — in-place self-rewriting of the harness. Continual Harness [33] formalizes a reset-free loop in which an agent alters *its own* prompt, sub-agents, skills and memory while running, and reports a substantial reduction in action cost against a minimalist baseline, recovering most — not all — of the gap to a hand-engineered expert harness. Meta-Harness [32] searches for improved harness code end-to-end from execution records, reporting a 7.7-point improvement over a state-of-the-art context manager at a quarter of the context tokens. Fork-based agents that refine above an immutable base with rollback belong to the same pole. These are strong engineering results and we do not dispute them.

Pole 4 — industrial agent-workforce platforms. In February 2026, OpenAI Frontier [34] made *agents as employees* a shipped enterprise product: agents are hired, onboarded, granted permissions, given structured feedback and subjected to performance review over a shared governance and institutional-memory layer, launched with HP, Intuit, Oracle, State Farm, Thermo Fisher and Uber. ServiceNow's Autonomous Workforce, Microsoft's Foundry Agent Service, Salesforce Agentforce and CrewAI Enterprise occupy the same pole. We state the consequence for our own positioning plainly: **an org chart for agents — roles, permissions, performance evaluation — ceased to be a distinguishing architecture in early 2026**. Any framing that rests on it describes the field rather than separating from it, and this paper does not rest on it.

What this pole governs is **agents**: their identity inside one tenant, their access, their measured output. What it does not constitute is an **institution with a boundary** — a unit that retains provenance across a change of the agents inside it, that can refuse to transmit what it has not validated, and that can federate with a peer institution under a policy neither of them owns. Frontier's memory is institutional in the sense of *shared within one company*;

it is not institutional in the sense of *portable across an organizational boundary under an explicit trust relation*. The difference is one of unit, not of quality: on distribution, integration depth and operational maturity this pole is ahead, and we do not claim otherwise.

And on the axis this paper is about, Pole 4 shares the limit of the other three. Performance review of an agent is a measurement of *output*; it is not a measurement of whether the signal driving a change is reliable enough to act on. Nothing in these platforms refuses to act on an unreliable signal — they surface it to a human reviewer, which is a different mechanism with a different failure mode.

What the four poles share is the input, not the output. All of them refine from *whatever trajectories they have*. None conditions the refinement on a measurement of whether the signal driving it is reliable enough to act on. That is the difference this paper reports, and §7.3 states its current limit plainly: the gate is built and tested, and **no live run yet shows it withholding activation** — the run we previously cited for that is retracted.

A second difference is architectural rather than methodological. Pole 3 rewrites the harness of an agent. This work does not touch the harness: it wraps it. The unit that carries identity, boundary, provenance and judgment is the **institution**, and the agent runtime beneath it is replaceable by construction (§4).

2.8 Convergence: three independent lines, one floor

The most interesting thing about the four poles is not how they differ from this work. It is **what they agree on without having coordinated**.

Synergy [31] arrives at an architecture that maps almost one-to-one onto the three IdentIA principles: its collaboration onto our principle of collective construction, its identity onto continuity, and its lifelong evolution onto the union of both. Fork-based agents that refine above a base they cannot alter, with rollback, recognize in practice — without stating it — the same preservation condition this paper calls principle B. And both Continual Harness [33] and Meta-Harness [32] begin from the premise that the material one learns from is **execution records**, not impressions.

Immutable base, evidence, reversibility. Three independent lines of work, with different motivations, arrived at the same floor.

That convergence is the argument, and it is not ours. When teams that do not coordinate solve a problem the same way, the economical explanation is not that they all had the same idea: it is that **the problem forces the shape**. A design reached from three different directions stops being a preference of its authors.

What we did **not** find in any of the three lines is the next step: **measuring whether the signal guiding refinement is reliable enough to act on, and declining when it is not**. All of them refine; none abstains. That is where this work sits — and §7.3 states without ornament how far that claim reaches today: the gate is implemented and tested, and **no live run yet shows it withholding activation**; the run we previously cited for that is retracted.

3. IdentIA: an engineering ontology

IdentIA fixes a *position* on identity, change and truth so that hard cases have a principled answer instead of an ad-hoc one. It is a **tie-breaking criterion and a coherence test for features**, not a user-facing theology: a society can run without anyone invoking it by name.

The stance that an ontology can be *engineered* — that architectural decisions are philosophical positions, and should be examined as such — is the working premise of the mind-design tradition [30]. That a "mind" can be a persistent *organization of mechanism* rather than an abstract formalism goes back to McCulloch and Pitts's logical calculus of nervous activity and McCulloch's *Embodiments of Mind* [29]: SIS's profiles — bodies (runtimes), memories, and learned habits under a persistent anchor — are agents in that embodied sense, not disposable function calls. Each principle also has named antecedents, which we record for intellectual honesty rather than authority: Dynamic Striving (A) in process and fallibilist thought (Whitehead, Popper, Prigogine) and in modern learning-through-interaction practice (LeCun); Absolute Continuity (B) in substance and implicate-order metaphysics (Plato, Spinoza, Bohm, Tegmark); Emergent Pragmatism (C) in pragmatism and bounded rationality (Dewey, Simon) and in the empirical primacy of search-and-learning over hand-coded knowledge (Wolfram, Sutton). The principles stand or fall by the invariants of §3.2, not by this lineage.

3.1 Three principles

Let a *profile* p denote a persistent agent identity.

Definition 1 (Agent identity). The identity of p is the tuple

$$\Psi(p) = \langle \text{SOUL}(p), M(p), L(p), K(p) \rangle,$$

where $\text{SOUL}(p)$ is the identity anchor (a human- or system-authored character/role document), $M(p)$ is the profile's memory (procedural, semantic, episodic; §3.3), $L(p)$ is its lineage (clone-from edges, skill provenance, session history), and $K(p)$ is its skill set — the profile's authored and acquired skills with their provenance, named as a component of its own (§D.1).

Definition 2 (Measurement). A *measurement* of p at time t is the tuple

$$m_t(p) = \langle R_t, \theta_t \rangle,$$

the active runtime R_t (harness, tools) and the active model θ_t .

Both definitions are this paper's own. To our knowledge, no prior work gives an agent's identity as a persistent object formally separated from its realization: the nearest, Perrier and Bennett [10][11], treat identifiability, continuity and persistence as *properties to be measured* of a language-model agent, and Synergy [31] states profile and typed memory as *requirements* of an agentic citizen. Neither makes "a change of model does not change Ψ " a consequence that can be checked; here it is the verification contract of principle B (§3.2). Both definitions, and the invariance theorem, are machine-checked (Appendix E).

A note on all notation in this paper. The symbols are a *formal specification vocabulary*, not a quantitative mathematical model. There are no metric spaces, operators or distributions defined over Ψ ; the notation names typed architectural components and the constraints over them. Read as specification, it does work. Read as mathematics, it would be over-claiming.

The three principles are:

- **(A) Dynamic Striving.** Identity is updated by practice. Skill authoring, curator consolidation, RAG ingestion, and memory writes all *increase* the degree to which p is itself. Formally, A endorses monotone enrichment of $\Psi(p)$ through action; the gap between current capability and potential is latent capacity, not failure.
- **(B) Absolute Continuity.** $\Psi(p)$ persists under change of measurement. Swapping $\theta_t \rightarrow \theta_{t'}$ or $R_t \rightarrow R_{t'}$ does **not** change the identity provided $\Psi(p)$ is preserved. The distinction is between a **persistent state** and its **transient realization**: Ψ is the object the theory protects across time; m_t is a readout of how that object is being run right now. The whole principle licenses one engineering rule — **never destroy Ψ to satisfy a measurement**.
- **(C) Emergent Pragmatism.** What an institution may treat as knowledge is not a single profile's assertion; it is what the institution can *validate*: by use, peer review, governance, evaluator fitness, and marketplace promotion. We call the output **institutionally warranted belief**, not truth, and the distinction is not decorative: a society can converge and still be wrong. The schema already encodes this humility — facts are **superseded**, never converted into timeless truths — and the vocabulary here is aligned with the schema rather than with the ambition. Consensus is not decreed; it is what survives successful application across distinct contexts, and it remains revisable by construction.

Where the formalism is worked out. Two subsections that stood here in v4 — the human analogy that motivates the tuple and introduces the residual term Ω , and the behavior map $b_t = f(\Psi, m_t) + \varepsilon_t$ that says where an agent's Ω lives — are **Appendix D** (§D.1, §D.2). They are the densest formal passage in the paper and they are load-bearing for §3.2 and §9.2, but they are not needed to read the invariants, the architecture or the measurements. The one-line consequence, for a reader who does not go to the appendix: **Ω is not a component of the identity object**. Every component of Ψ must be persistable, inspectable and enforceable; Ω is none of the three, so placing it inside Ψ would make principle B's preservation condition unverifiable and identity claims unfalsifiable. Ω enters through the behavior map instead, and governance bounds its effect rather than the ontology modelling it.

3.2 Invariants (what the ontology forbids)

The principles are made operational as invariants enforced by governance and schema, not by exhortation:

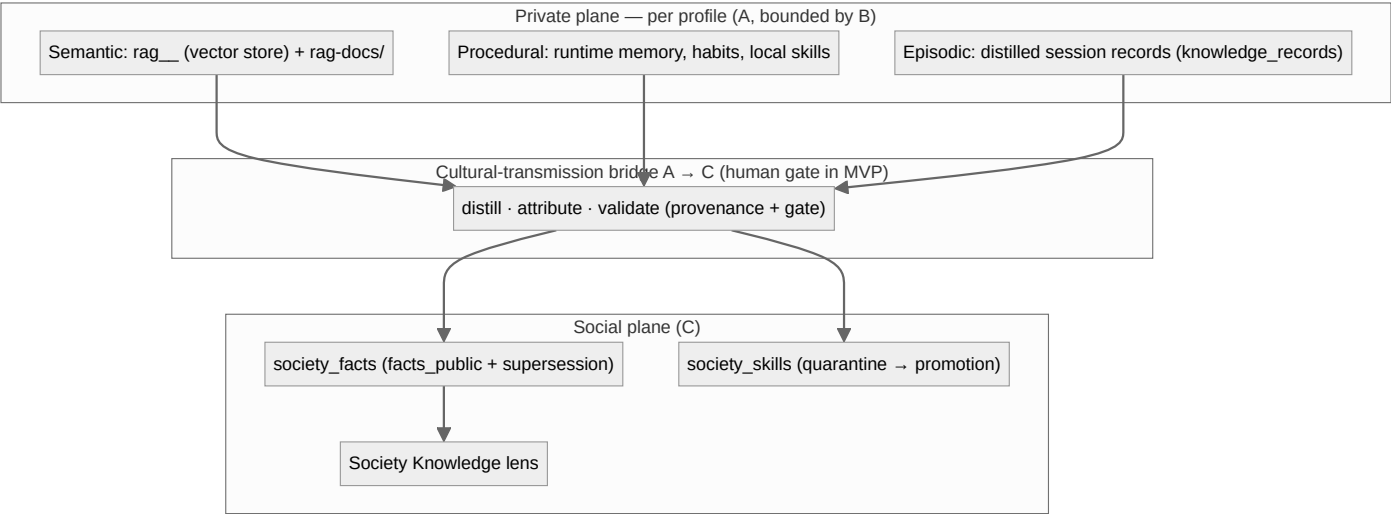
Invariant	Principle	Enforcement in SIS
Archive, do not delete profiles	B	Default lifecycle is <i>archive</i> ; destructive delete requires human approval + memory/RAG export (§6).
Supersede, do not overwrite facts	B + C	<code>facts</code> schema carries <code>supersedes / superseded_by</code> ; incorrect facts are <i>superseded</i> , lineage preserved.

Invariant	Principle	Enforcement in SIS
No unilateral truth	C	Contradictory findings escalate to auditor/governance/voting; the system never silently picks a winner.
Lineage, not copy	B	<code>clone_from</code> and <code>forked_from</code> record provenance edges in the graph.
Measurement $\neq$ identity	B	Model/runtime are per-profile configuration; changing them is not an identity event.
Reversible mutation (ratchet)	B	Every fitness-driven change of skill/SOUL/config is reversible to a fitness-dominant prior (§5.8).

A proposed feature that requires deleting profiles, replacing (not patching) skills, or letting an agent declare public truth unilaterally is, by construction, an IdentIA violation and a refactor candidate. This *falsifiability at the design level* is the practical payoff of naming the ontology.

3.3 Two planes of memory

SIS distinguishes a **private plane** (principle A, bounded by B at the profile frontier) from a **social plane** (principle C):



The privacy/identity frontier is the **profile, not the host**: an agent sees neither another agent's RAG nor its episodic memory, even co-located. The episodic substrate is recorded by an *episodic-context* ingestion path (sessions distilled into `knowledge_records`); the read side of that episodic memory is design-only (§7.3).

3.4 The A → C bridge: governed cultural transmission

Humans do not only remember individually; they **transmit** technique across generations. SIS models this as a *governed* bridge from the private plane to the social plane. A private finding is **distilled, attributed, and validated** before it becomes social knowledge — never copied raw, never declared true unilaterally (that would violate C). The bridge has two degrees:

1. **MVP — human-gated transmission (implemented path).** A `knowledge_record` (e.g. of kind `fact`) is promoted to a `fact_candidate` and then to `society_facts` only with human approval. This is the first act of cultural transmission, honest and cheap.
2. **Debt — emergent agent-to-agent transmission (design-only, deferred).** An architect-style profile reads the memory of a profile that has mastered a skill (high latent ability θ , §5) and propagates it to one that lacks it, *with validation before propagation* ("do not teach an error as best practice"). This requires cross-profile write trust policies and reliable fitness; it is risky to automate early and is explicitly registered as deferred technical debt.

The bridge is fed by two sources with identical shape but distinct origin: **individual** (one agent's distilled episodic memory) and **collective** (the *minutes* of a cross-host assembly/council — distilled decisions with multi-participant provenance, not a raw recording). Minutes turn a social event into queryable institutional memory ("who knows about X?") and a competence index for routing future queries to the right experts.

4. System architecture

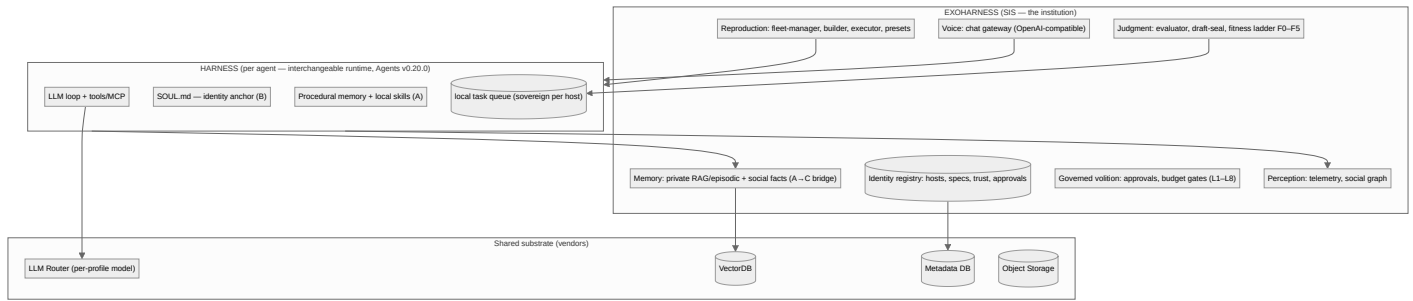
4.1 Principle: the sovereign organizational unit

The deployment and scaling unit is not an agent but an **agents-host**: a complete sovereign organizational unit with its own identity, local task queue, governance, evaluator, telemetry, and agents. An operator describes an organization in natural language or YAML — "a bank with Risk, Compliance, IT and Operations," "a K-12 school," "the ministries of a state" — and the system provisions one host per unit and federates them. Horizontal scale is a *new host*, not more runtimes on a central host. (Lineage: the project originates in a "mini-village" of three isolated Linux VMs, one agent each; each VM is now a full `agents-host`.)

4.2 Harness vs. exoharness

SIS does not reimplement the agent loop. We distinguish:

- **Harness** (per agent, interchangeable): the LLM loop, tools/MCP, procedural memory, sessions, the SOUL anchor, and a local task queue. In the current deployment this is the **Agents runtime, pinned to v0.20.0** (SHA `9d4ef04ed`, E-03; whose memory tool accepts a batch `operations` array, consolidating multi-write memory updates into a single call); the contract is runtime-agnostic.
- **Exoharness** (SIS): the institutional layer that wraps *all* agents and gives them registered identity, social memory, judgment/selection, governance, reproduction (lifecycle), and observability. *An agent without the exoharness is a process; with it, a citizen of an institution.*



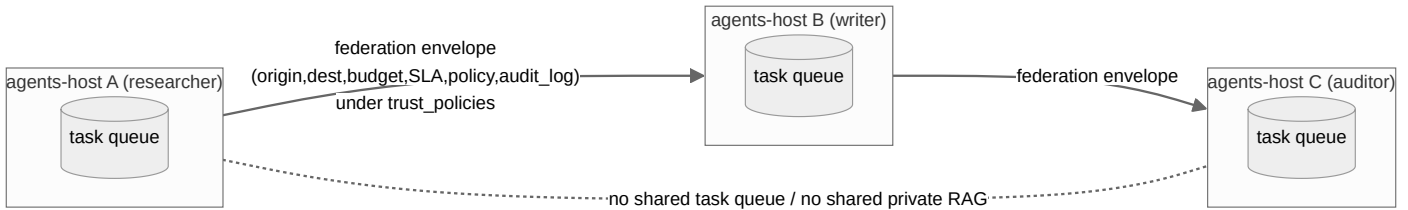
On the word, because a reader will bring it up. We use *exoharness* in the sense its root carries: *exo-* is "outside", as in *exogenous* or *exoskeleton* — a structure that **envelops** what is inside and gives it boundary, protection and shape without replacing it. That is the relation of SIS to the agent harness: the harness keeps its loop, tools, memory and identity anchor; the exoharness surrounds it with registry, social memory, judgment, governance and lifecycle. The same word is used elsewhere with the opposite geometry. The *exoharness/exo* specification [36] defines an exoharness as "a trusted substrate that manages durable state, brokers access to privileged resources, and provides low-level execution plumbing without owning agent semantics" — a layer **beneath** the executor, which the root would call *endo-*, or simply a kernel. We do not claim the term, and we do not dispute that usage; we note that the two name different layers of the same stack, ours above the harness and theirs below it, and that in this paper the word always means the former. A third name for our layer is also in circulation, *meta-harness*, and Meta-Harness [32] uses it for an outer loop that searches over harness code; we keep *exoharness* because what we describe does not search over the harness — it wraps it (§4.2, Fig. above).

4.3 Sovereignty layers

Elements are partitioned by scope. The **sovereign layer** (never shared without explicit federation) holds the local task queue (`kanban.db` , SQLite), the private per-agent RAG (`rag_<host>_<profile>`), and the society metadata registry. The **platform layer** (LLM router, VectorDB, object storage, metadata DB) may be *shared* (logical isolation by namespace/collection/bucket/tenant — the pilot default) or *dedicated* per host/tenant when regulation requires. The **federated layer** (`society_skills` , `society_facts` , artifacts, exportable public RAG) circulates between hosts with provenance preserved.

4.4 Federation

Cross-host work is carried by a **federation envelope**: a structured, audited record (table `federated_tasks` : `envelope_id` , origin/destination host, title, budget, SLA, policy, result, and an append-only `audit_log`) dispatched by the fleet-manager to a destination host's task queue and governed by **trust policies** (table `trust_policies` : which host may delegate to which, what data may cross). The pipeline chain lives **in the agents' SOULs, not in a central orchestrator** — each agent knows only its next step — which is itself an IdentIA statement: Absolute Continuity (any host replaceable), Dynamic Striving (each agent advances on its own impulse), Emergent Pragmatism (the pipeline emerges from audited handoffs). *Honesty note*: envelopes are currently **structured and audited but not cryptographically signed**; cryptographic signing is future work (§7.4, §9.2). The prose label "signed envelopes" in earlier internal documents is aspirational; cross-host trust currently rests on policy plus network boundaries.



4.5 Services

The control plane comprises `fleet-manager` (profiles, RAG bootstrap, task-queue proxy, swarms, Society Builder, fitness-routing consumer), `governance` (approvals, budget gates), `evaluator` (deterministic checks + the fitness ladder), `society-telemetry` (graph events, heartbeats, snapshots, federated aggregation), `society-executor` (operator/architect action execution, including federation deployment), and a `society-chat` OpenAI-compatible **Conversation Gateway**. Per-agent **MCP servers** (stdio) expose `society-rag`, `society-skills`, `society-artifacts`, `society-facts`, `society-telemetry/graph-query`, `society-vision`, and `society-audio` (and a contract-only `society-ros2`). The chat gateway, by product rule, **writes no memory and creates no tasks automatically**; a chat-derived `goal_candidate` requires human approval before becoming a goal.

The canonical port map (from the deployment manifest in `infra/`): `agents-host API` :8642 and dashboard :7474; `fleet-manager` :8088; `governance` :8090; `society-telemetry` :8091; `evaluator` :8092; `society-graph-ui` :8093; `society-chat` :8094; `society-executor` :8099; LLM Router :4000; VectorDB :6333; Metadata DB :5432; Object Storage :9000 / :9001.

5. Measured selection: the implemented rungs and the reliability gate

This section reports the rungs that exist, are unit-tested, and have run. The ladder has six rungs; three are here. What the other three would do, and the argument for why they must come in this order, is **[PB]**.

5.1 Variation vs. selection

The agent runtime already provides a **variation operator**: it authors and patches its own skills and consolidates memory in a self-improvement review. What is missing is a **measured selection operator**: a way to decide *which* of those changes are actually better and should survive, propagate (the A → C bridge), and anchor (principle B). Schematically:

$$\text{RSI} = \underbrace{\text{Variation}}_{\text{exists (runtime)}} \circ \underbrace{\text{Selection}}_{\text{this ladder}} \quad \text{in a closed loop, editing the scaffold, gated by fitness.}$$

Variation without selection is **drift**; selection without variation has nothing to choose from. The framing is deliberately Darwinian [28]: improvement is never *declared* by the object that changed; it is *selected* by measured comparison against alternatives in an environment. The ladder builds selection first, and only its top rung closes the recursive loop.

5.2 Five fitness objects

There is no single "fitness number." Five objects carry differently-shaped signal:

Object	Raw signal	Source table	Target estimator
Skill	usage_count , success_count	skill_publications	Beta posterior → Bradley–Terry/TrueSkill
Profile/agent	task pass/fail, heartbeats	Kanban task_runs , runtime_heartbeats	IRT (difficulty-adjusted) + graph reputation
Fact	supersession, citations	facts / society_facts	survival/supersession rate + peer-prediction
Repertoire	diversity of active skills/profiles	fitness_repertoire	Quality-Diversity (MAP-Elites)
Routing	who gets the next task	fleet_manager , Kanban dispatch	Thompson sampling over posteriors

5.3 The rungs and their self-triggers

Each rung is a milestone that declares an algorithm, the data it needs, a **self-trigger** (a condition the system measures about *itself* to advance), its dominant IdentIA principle, and its governance level. Self-trigger thresholds are environment-configurable (SOCIETY_FITNESS_*). Badges reflect the codebase as of June 2026:  implemented & tested;  implemented, in-progress or shadow;  design-only.



The three rungs described in detail below — F0, F1 and F2 — are the ones that are implemented and have run. F3, F4 and F5, and the argument that their order is a safety constraint rather than a roadmap, are **[PB] §3–§4**.

5.4 F0 — Beta–Bernoulli reputation with explicit uncertainty

For an object with *s* successes in *n* attempts, place a uniform prior and report the Beta posterior

$$\pi \sim \text{Beta}(\alpha, \beta), \quad \alpha = s + 1, \quad \beta = (n - s) + 1, \quad \mathbb{E}[\pi] = \frac{\alpha}{\alpha + \beta}.$$

Ranking does **not** use the posterior mean; it uses the **Wilson lower bound** at confidence *z*,

$$\hat{p}_{\text{LB}} = \frac{\hat{p} + \frac{z^2}{2n} - z \sqrt{\frac{\hat{p}(1-\hat{p})}{n} + \frac{z^2}{4n^2}}}{1 + \frac{z^2}{n}}, \quad \hat{p} = \frac{s}{n},$$

so that a 2/2 cold-start cannot dominate a 40/45 track record. Uncertainty is first-class. The **canonical outcome** that defines a "success" is a single shared function (canonical_task_outcome) returning success | failure | inconsistent | pending from a closed task and its runs, so that the evaluator's task_outcome_present check and the fitness layer share one definition. **Status: implemented and unit-tested** (evaluator/fitness.py::beta_posterior , wilson_interval ; tests/test_fitness_f0.py , 8/8). Counters are

fed live by the Agents `society-observer` plugin, which posts to `/fitness/skill-usage` on each successful `skill_manage`; recompute runs on a periodic cycle (`SOCIETY_EVALUATOR_FITNESS_CYCLE_SECONDS`, default 3600) and on-demand via `POST /fitness/recompute`. The module imports only the Python standard library (`math`, `typing`): there is no third-party rating dependency.

IdentIA: A (practice updates the number). *Governance*: L1/L2 (read-only). *Self-trigger to F1*: at least `SOCIETY_FITNESS_MIN_PAIRWISE` (default 30) head-to-head events per object. This threshold is an **engineering default, not a derived sufficiency bound**; §5.9 records that the ladder's self-trigger thresholds are chosen rather than proven, and this one is no exception.

5.5 F1 — Relative rating: Bradley–Terry and TrueSkill

To measure *better-than* rather than assert it, SIS rates skills with **Bradley–Terry** [12] and profiles with **TrueSkill** [13].

Bradley–Terry. With latent strengths $\beta_i = \log \gamma_i$, the probability that i beats j is

$$P(i \succ j) = \frac{\gamma_i}{\gamma_i + \gamma_j} = \sigma(\beta_i - \beta_j).$$

Strengths are fit by the Minorization–Maximization (Hunter) update with a soft prior; given win counts w_{ij} and pair counts n_{ij} ,

$$\gamma_i \leftarrow \frac{\sum_{j \neq i} w_{ij}}{\sum_{j \neq i} \frac{n_{ij}}{\gamma_i + \gamma_j}}.$$

The reported rank score is shrunk by evidence (a relative-Wilson factor) so that sparse rivals do not leapfrog well-tested ones. This is the same family used by reward models and by Chatbot Arena [24]. **Status: implemented and tested** (`evaluator/fitness.py::bradley_tery`, hand-rolled with no external dependencies; `tests/test_fitness_f1.py`). Validated live with real artifacts: in a controlled set, `writer (4–0) > researcher (2–2) > auditor (0–4)` produced a coherent ranking.

TrueSkill. Each profile holds a Gaussian belief $\mathcal{N}(\mu, \sigma^2)$. After a 1-v-1 result with performance noise β and total uncertainty $c^2 = 2\beta^2 + \sigma_i^2 + \sigma_j^2$, the winner i and loser j update

$$\mu_i \leftarrow \mu_i + \frac{\sigma_i^2}{c} v\left(\frac{\mu_i - \mu_j}{c}\right), \quad \sigma_i^2 \leftarrow \sigma_i^2 \left[1 - \frac{\sigma_i^2}{c^2} w\left(\frac{\mu_i - \mu_j}{c}\right)\right],$$

with the standard truncated-Gaussian functions $v(\cdot)$ and $w(\cdot)$ and a draw margin. Conservative ranking uses $\mu - 3\sigma$. **Status: implemented and tested** (`evaluator/fitness.py::trueskill_ratings`, hand-rolled via `math.erf`; validated live: three wins moved a profile to $\mu \approx 32.4$ vs. 17.6 with σ shrinking under evidence). Graph reputation (eigenvector/PageRank over the `society_facts` citation graph) is **design-only**.

Producers of head-to-head events. Comparisons come from (i) the deterministic gym (§7.1), implemented; and (ii) an **anti-position-bias pairwise LLM judge** (§5.5.1), implemented. Assembly/council producers are design-only.

5.5.1 Anti-position-bias pairwise judge

The judge compares two documents under a criterion. To neutralize position bias it evaluates **both orders** (A,B) and (B,A) and combines: agreement yields a winner, disagreement yields an honest tie. **Status: implemented and tested** (`evaluator/judge.py::build_pairwise_prompt`, `parse_verdict`, `combine_orders`, `verdict_to_match_winner`; `evaluator/core.py::judge_pairwise` runs both orders and persists a `source='llm_judge'` match; `tests/test_judge_pairwise.py`, 8/8). A documented empirical finding from internal runs: **the choice of judge model matters** — a small, weak local judge model exhibits strong position bias, against which the two-order protocol degrades gracefully to *honest ties* rather than spurious winners; a stronger frontier-class judge yields reliable verdicts.

IdentIA: C. *Governance*: L4 (marketplace promotion begins to use rating). *Self-trigger to F2*: task×agent matrix density $\geq$ `SOCIETY_FITNESS_MATRIX_DENSITY` (default 0.15) **and** each active profile has $\geq$ `SOCIETY_FITNESS_MIN_TASKS_PER_AGENT` (default 20) overlapping tasks.

5.6 F2 — Difficulty-adjusted ability (IRT) — *the reliability gate*

F2 is the **reliability gate** and a *precondition for any automatic write-back*: it separates "the agent improved" from "the agent picked easy tasks." Using a 2-parameter logistic Item Response Theory model [20] with task difficulty b_k , discrimination a_k , and latent ability θ_p :

$$P(\text{profile } p \text{ solves task } k) = \sigma(a_k(\theta_p - b_k)).$$

Tasks are "items," profiles are "examinees." The estimator is a pure **Rasch (1PL) base with an optional 2PL upgrade by flag**, fit by Joint Maximum Likelihood (Newton–Raphson) with a Gaussian anti-divergence prior, standard errors from Fisher information, and identifiability fixed by centering (`evaluator/fitness.py::irt_fit`). It is wired to data through three tables — `fitness_responses` (the raw task×agent matrix), `fitness_items` (the difficulty catalogue), and the `theta` column of `fitness_scores` (`method='irt'`) — and the gym persists the response matrix alongside its pairwise matches. A **task-farming detector** flags an agent whose mean item difficulty is far below the population's while its raw success rate is high. ****Status: implemented, unit-tested** (`tests/test_fitness_f2.py`, 9/9: synthetic parameter recovery with θ correlation

0.85, difficulty ablation, anti-farming, non-divergence, SE shrinking under evidence, 2PL), and validated live** (§7.2).

IdentIA: B + C (continuity of the measurement; truth by accumulated use). *Governance*: L4/L6. *Self-trigger to F3*: $\text{Var}(\theta_p)$ in a moving window $<$ `SOCIETY_FITNESS_THETA_NOISE` (`fitness_theta_stability`, with a `fitness_theta_history` window) **and** the fit reports `converged=True` — the system only lets fitness *act* once ability estimates stop dancing *and* the model has actually converged. Both halves of that conjunction were defective until 2026-09-26 and are described as repaired here; §7.3 reports what the defects were and which claim of this paper they cost.

5.7 F3–F5 — acting, preserving diversity, and gated RSI

Not in this paper. F3 (Thompson sampling + NSGA-II routing) and F4 (MAP-Elites diversity preservation) are implemented and unit-tested but run in **shadow, disabled by default** — they recommend, they do not assign, and no path consults them as a precondition. F5 (recursive self-editing) is **design-only and hard-gated**; its variation operators would be PromptBreeder [16] or EvoPrompt [35], with POET [21]-style generation of harder tasks. Their mechanics, the dependency $\text{enable}(F5) \iff \text{Done}(F2) \wedge \text{Done}(F3) \wedge \text{Done}(F4)$, and the ordering argument are developed in [PB] §3–§4; the gate itself is machine-checked (Appendix E). Their status is in §7.4 and in ledger entries E-13 and E-14; the reason none of them is active is the measurement in §7.2, not a policy decision.

5.8 Transversal mechanisms: which bind, and which only report

These are transversal by design. **They are not uniformly enforcing, and the difference is stated per item rather than implied by the heading** — §7.3 and ledger entries E-15 and E-30 record how the first of them went from reporting to blocking in code, and why it has not yet blocked anything.

(i) an **anti-Goodhart held-out** set — *computed and served, and since 2026-09-03 a gate on active routing* (E-30): when the LLM judge contradicts the deterministic held-out, `fitness_route` in *active* mode degrades to shadow. Without enough judge comparisons it does not block, by design, and with routing switched off it has never fired live. It is hidden and rotated, scored by the deterministic evaluator, anchoring the LLM judge before F3 (`rotate_holdout`, `holdout_calibration`, `fitness_anti_goodhart`, `GET /fitness/anti-goodhart`; `tests/...`, 12/12; live: calibrated $p \approx +0.316$ with an alarm firing $p \approx -0.316$ on a deliberately corrupted probe); (ii) the **ratchet** (principle B): every mutation is reversible to a fitness-dominant prior (`archive-not-delete`, `superseded_by`); (iii) **honesty** mechanisms (peer-prediction/Bayesian-truth-serum) for fact and self-report fitness (design-only); (iv) **multi-objective always** (no object wins ignoring cost/latency from F3); (v) a **governance gate per level** (routing L3b/L5, promotion L4, template/SOUL L6, code/migrations L7/L8) with no skips.

6. Governance-as-code (L1–L8)

6.1 Levels

The base rule: *low-risk local self-improvement is automatic; shared or infrastructural changes need human approval; code or runtime-pin changes need a PR and tests.*

Level	Scope	Mode	Principle
L1	Local skills	Automatic	A
L2a	Cross-session memory	Automatic	A
L2b	User identity	Automatic, bounded	A
L2c	Local SOUL.md	Auto; template → governance	A / C

Level	Scope	Mode	Principle
L3a	Per-profile RAG	Controlled, budgeted	A
L3b	Short delegation	Automatic with quota	A
L4	Skills marketplace	Quarantine + use-based promotion	C
L5	Profile spawning	Auto if budget; else human	C
L6	Base template changes	Evaluator proposes; human approves	C
L7	Infrastructure (compose, migrations, storage policy)	Human	Metagovernance
L8	Code & runtime pin	PR + tests	Metagovernance

Per-goal **budgets** (`max_profiles`, `max_tasks_running`, `max_runtime_seconds`, `max_delegate_depth`, `max_tokens_estimated`, network/privileged-tool flags) gate actions; exceeding a budget creates an approval item. **Quarantine** isolates new skills, anomalous profiles, and RAG-crossing attempts. The `governance` service records decisions; it does not apply changes itself — acting services must consult it. **Status: governance service, approvals, budget checks, and kill switches (spawn, RAG writes, skill publication, evaluator auto-approvals) are implemented; the L5 spawn gate is enforced opt-in (`SOCIETY_GOVERNANCE_ENFORCE`); full enforcement wiring across every acting path is partial** and is a named blocker for the closed beta (§7.3).

6.2 Metacentrality

Not in this paper. The governance scheme was shaped by a normative thesis about how power should be structured inside a persistent agent institution — that every architecture of power tends to **capture** or **dissolve**, and that the response is a center deliberately constrained by audit, rotation, transparency and self-limitation. It is a position, not a result, and it adds no constraint of its own: every mechanism it names is enforced — or not — by the levels in §6.1 and the rungs in §5. Mixing it into a paper that reports measurements is precisely how a normative claim acquires the appearance of an empirical one. It is **[PB] §5**, where it is stated with its limits, including the one that matters most: it does not yet specify how the center audits *itself* beyond human approval, PR and tests — no rotation mechanism, no term limit, no procedure by which L7–L8 could be turned against whoever holds them.

7. Evaluation protocol and implementation status

We present a *protocol*, a *dated live result for the reliability gate*, and an honest status — not a finished empirical study. The ladder's self-triggers depend on data density that natural on-prem traffic would take months to produce; the gym manufactures it on demand. One item of the deployed suite, and the shape they all share, is in Appendix B.

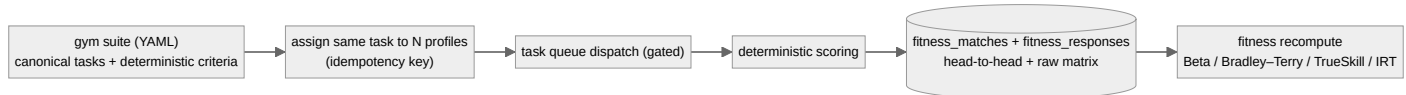
How to read this section. Every status claim below is entered in the **claims ledger (Appendix C)** with its source, verification method, date, environment and limit. Where a figure is *inherited* rather than re-measured for this draft, the ledger says so. Where a claim was **refuted** by measurement — including one of our own design guarantees (§7.6) and this paper's own former headline result (§7.3) — it is reported rather than removed.

7.1 The gym: controlled difficulty

A **gym suite** is a versioned YAML battery of canonical, self-contained tasks, each with a **difficulty** tier (1 direct instruction · 2 transformation · 3 synthesis) and *deterministic* success criteria (`required_patterns` , `forbidden_patterns` , `min_length` , `max_length` over the result text — no subjective judgment). The same canonical task is assigned to several profiles; the result is scored deterministically; a head-to-head **match** is recorded for each pair on the same task (success beats failure, equal is a tie), and the raw response matrix is persisted for IRT. This is the "controlled difficulty" the coherence rule demands: a change that cannot be measured as better against it is Goodhart.

Status: implemented and tested. Citations re-verified against the tree on 2026-08-23:

`evaluator/gym.py::validate_suite` (l. 23), `score_gym_output` (l. 85), `pairwise_matches` (l. 139);
`evaluator/tests/test_gym.py`; migration `012_fitness_matches.sql`; suite `gym/suites/canonical-v1.yaml`;
CLI `society gym run/score/progress`.



7.2 A dated live result for the reliability gate (F2/IRT)

On **2026-06-18** the IRT rung was exercised end-to-end on a production VPS (native filesystem). The gym matrix was densified to **6 profiles across two ability tiers** (a strong model and a weak local model) × **6 items = 36 responses**. Four checks were observed:

1. **Difficulty is recovered from data, not assumed.** Estimated item difficulty b_k spanned $\approx -1.81 \dots +1.73$ and **diverged from the suite's seed tiers** — the model learned difficulty from observed responses.
2. **Ability is separated and spread.** Latent ability spanned $\theta \approx 0.06 \dots 3.38$ with finite standard errors; the tiers separated as expected.
3. **The anti-Goodhart anchor behaves.** Held-out calibration was positive ($\rho \approx +0.316$) and an alarm fired on a deliberately corrupted negative probe ($\rho \approx -0.316$).
4. **Task-farming is caught.** The detector flagged a synthetic farmer (easy items, inflated raw rate) and did **not** flag the balanced population.

This result is dated. It is entered in the ledger as such. Its value is to demonstrate that the estimator recovers difficulty and separates ability on one small matrix, not to establish a population-level trajectory. The four observations above do not depend on the convergence flag that accompanied them, and that flag is the subject of §7.3.

7.3 A retraction: the refusal we reported was an artifact of our own criterion

The same run reported `converged=False`, and earlier drafts of this paper built a result on it. We retract that result here, state what replaced it, and keep the episode rather than deleting it, because how the error was found is the part worth reading.

What we claimed. That the $N = 6$ matrix contained *perfect response patterns* which JMLE cannot fit to a finite optimum; that the self-trigger therefore withheld F3 activation; and that a system measuring the reliability of its own fitness signal and declining to act on an unconverged one was exercising the safety ordering of [PB] §4 in code rather than asserting it in prose.

What re-measurement found (2026-09-26). Three defects, each of which alone invalidates the reading:

1. **The flag could not become `True`.** The convergence test measured the per-iteration change *before* the identifiability step that centres item difficulties, and that step partly undoes the change it was measuring. With difficulties pushed in the same direction — which is what happens when almost everyone solves almost everything — the measured step stayed near 0.27 on every iteration, while θ and b moved less than 10^{-15} between 50 and 5,000 iterations. The fit reached a finite optimum; the criterion could not say so. **The stated explanation was therefore also wrong:** the obstacle was not perfect response patterns defeating JMLE, it was our own stopping rule.
2. **The citation did not support the claim.** "F3 consults it and declines to act" was referenced to line numbers in the pure fitness module, which computes the flag and returns it in the fit's metadata. That module holds no database, no network and no consumer; it cannot gate anything. The evidence we offered established the first half of the sentence and not the second. That no consumer existed is not a supposition: an internal audit of 2026-08-31 grepped the whole tree outside the tests and that module and found none, and the migration that later added one says the same in its own header (E-27).
3. **The trigger that did control F3 opened on its own.** The θ -stability window counted *recalculations* rather than distinct states of the data. With the data unchanged, repeated identical recalculations produced a variance of zero and satisfied the trigger.

What survives and what does not. The four observations of §7.2 survive: they are properties of the estimates, not of the flag. What does not survive is this paper's headline empirical claim. **We have no live result establishing that the gate withholds activation**, and we withdraw the assertion that no comparable system reports the property, since we had not demonstrated it ourselves.

Where the ladder stands now. After the fix, a fit over 720 responses (20 profiles $\times$ 12 items) reports `converged=True` in 14 iterations without adding data (E-28). That measurement is from a development environment, not a client installation, and it is one run. F3 remains inactive **by decision, not by gate** — a distinction the earlier draft specifically denied, and which we now state the other way round.

What has since been built, and it is the thing we had claimed. On the same day as the repair, the flag stopped being ephemeral: each fit's metadata is persisted one row per recomputation (migration `039_fitness_irt_runs`), and `fitness_route` in *active* mode reads the latest row and falls back to shadow with `gated_reason="irt_not_converged"` in three cases — the fit did not converge, no row exists, or the row cannot be read. It **fails closed**, and eight tests fail without it (E-29). So the sentence we published before its referent existed now has one. We report it as a property of the tree, measured by tests: **no live run has exercised it**, and reporting it as though one had is the error this section exists to retract.

Why this is in the paper. A paper whose method is *do not assert the unmeasured* has to survive its own method. The ledger recorded this figure with its date, its source and the limit *not re-measured since*; three months later that limit is what prompted the re-measurement that destroyed the claim. The bookkeeping worked and the claim did not, and reporting only the first would be the precise failure the anti-Goodhart anchor exists to prevent.

A second, unrelated weakness in the same area, reported rather than omitted: at the 2026-08-23 cut the anti-Goodhart guardrail was *computed and served* but did **not block** (E-15): defined, imported and exposed through a read endpoint, with no path consulting it as a precondition, under a heading of the internal documentation that read "cross-cutting, not optional". Since 2026-09-03 it is a precondition of active routing (E-30). It has still not blocked anything: routing is off by decision, and the anchor compares against LLM-judge rankings, of which the development environment held none on 2026-10-01, so by design it cannot raise an alarm. §5's claims about it describe a gate that exists in code and has not been exercised.

7.4 Hypotheses, metrics, and what is implemented vs. designed

Hypotheses. H1 (calibration): gym-fed Bradley–Terry/TrueSkill rankings agree with the deterministic ground-truth ordering under controlled difficulty. H2 (uncertainty discipline): Wilson-lower-bound ranking prevents cold-start domination. H3 (judge robustness): the two-order judge reduces position-bias error and degrades to ties under weak judges. H4 (reliability gate): under IRT, profiles cannot raise rank by selecting easy items; the task-farming detector fires (observed, §7.2). **Metrics:** rank correlation vs. ground truth; cold-start inversion rate; position-bias disagreement rate; θ recovery correlation; pipeline latency; auditor PASS rate; cost per task (router tokens); graph coverage. **Baselines:** round-robin routing; single-order judging; mean-only (non-Wilson) ranking.

Component	Status	Ledger
F0 Beta posteriors + Wilson ranking	✓ implemented & tested	E-11
Canonical task outcome (shared definition)	✓ implemented & tested	E-11
Bradley–Terry (MM/Hunter, no deps)	✓ implemented & tested	E-11
TrueSkill (online, <code>math.erf</code>)	✓ implemented & tested	E-11
Anti-position-bias pairwise judge	✓ implemented & tested	E-11
Gym (validate/score/pairwise + matrix persist)	✓ implemented & tested	E-10
Deterministic evaluator checks (artifact, health, task outcome, draft seal)	✓ implemented & tested	E-12
F2 IRT (1PL/2PL, JMLE) + θ-stability + task-farming + held-out anchor	● implemented & tested; live 2026-06-18 result retracted (§7.3); no live gate decision stands	E-13
F3 Thompson + NSGA-II + QD coverage	● implemented & tested; shadow / disabled by default	E-14
F4 MAP-Elites archive + niche protection + plateau	● implemented & tested; shadow / disabled by default	E-14
Anti-Goodhart held-out anchor	● computed and served; gates active routing since 2026-09-03, never exercised live (§7.3)	E-15, E-30
Federation envelopes + trust policies	● implemented; structured and audited, not cryptographically signed	E-08
Inter-host transport (A2A v1.0)	✓ enabled and exercised with negative controls	E-07
Closed graph vocabulary + build-time guardian	✓ implemented; guardian fails the build on undeclared types	E-09

Component	Status	Ledger
SHACL shapes	🟡 declared; the file states that it does not execute in the build	E-09
Approval identity binding	🔴 the decision field is request-supplied text ; not bound to a verified principal	E-16
Audit log integrity	🔴 not hash-chained ; some writes deliberately fail-open	E-16
Graph reputation (eigenvector over facts)	📋 design-only	—
Episodic-context read-side	📋 design-only (write-side implemented)	—
F5 recursive self-editing	📋 design-only (hard-gated)	—
Cross-profile governed memory transmission	📋 design-only	—

Counted. Of the 21 components tracked above: **9 are implemented and tested**; **6 are implemented but do not bind** — they compute in shadow, are served without blocking, or are declared in a file that states it does not execute; **2 are implemented with a named defect** that makes the guarantee they suggest unavailable; and **4 are design-only**. The ratio matters more than any single row: **12 of 21 do not currently constrain anything**, and the closed variation → selection → mutation loop needs items from two of those groups — F3 acting rather than shadowing, and F5 existing at all — so it cannot run today by construction, not by configuration. Storage for F0–F2 is the four-table schema of migrations 011–014 (E-25); the tables are named there, not reproduced here.

7.5 Interoperability: a conformance measurement against an external suite

Claims of interoperability are usually asserted. We measured one.

The Unified Harness Protocol (UHP) publishes a versioned specification with an executable conformance suite. Run against our profile API (`core` class) on 2026-08-23 (E-17), the result was **1 of 37 checks passing: 1 pass · 8 fail · 24 skipped · 4 error**. The 24 skips are a single-point cascade — without a catalogue endpoint there is nothing further to exercise. The gap is four concrete items: a discovery document, a version header (and rejecting an unsupported version with an error rather than silently substituting one), and two catalogue endpoints whose content already exists under other names. The only check we pass today is rejecting an absent credential.

The known shape of the fix for the audit gap. Where §7.4 reports that the audit log is not hash-chained, the remedy is not exotic: chain each entry to the previous one, so that removing or altering one breaks the chain detectably, and pair it with a signed attestation of whom each action belongs to. Open implementations of both pieces exist. **We state the direction, not a commitment to a schedule or to a supplier.**

Two adjacent limits, named because the conformance run surfaced them. First, the inter-host transport authenticates with flat bearer tokens per caller pair: a caller cannot delegate a *narrowed* authority — one task, with an expiry — without returning to the issuer. Offline-attenuable token schemes exist and address exactly this; adopting the contract, not the mesh, is the direction we would take. Second, our tool servers are stdio-only, reachable only by the agent process that spawns them. This is why the skills server cannot carry a credential of its own, a limitation we have documented internally as debt rather than design. An HTTP/SSE transport for the same protocol removes it.

The finding worth reporting is not the score. The specification requires that discovery be served **without authentication**. Our internal services are fail-closed by design — a posture adopted after two silent authorization failures in production. Serving an anonymous endpoint on the internal network is therefore a **security decision, not a conformance chore**, and we decline it absent an external client that requires it. We report the 1/37 rather than the three items we intend to close, because a conformance score quoted after selective remediation is not a measurement.

7.6 A refuted design guarantee, reported

Our own documentation asserted, in two places — the deployment manifest and the service's module docstring — that a single gated service holds authority over the container runtime, and that the planning agent therefore cannot provision without human approval. **Measurement refutes the premise** (E-23). The runtime socket is mounted into **three** services, not one; the agent host is among them, and the account inside it holds passwordless privilege escalation.

A detail that makes the finding sharper rather than milder: the agent host's shipped configuration selects a **local** terminal backend, not a container one. The socket is therefore not required by the runtime's current execution mode; it is mounted by default and adjusted at start-up behind an opt-in switch, for a container-sandbox path the default configuration does not use. **A privilege mounted by default for a mode that is not active is surface without current purpose**, which is a weaker justification than the original assertion implied.

Two qualifications, both necessary. First, this **refutes the written guarantee; it does not demonstrate an exploit** — effective permissions depend on deployment and we did not attempt one. Second, the *approval* gate is independent of the socket: the deployment path requires an approved specification, and that requirement is enforced in code.

We include this because the alternative — silently correcting the sentence — would leave the reader unable to distinguish a system whose boundaries were verified from one whose boundaries were merely stated. **A system that reports the refutation of its own guarantee is making a stronger claim about its method than one that reports only successes.**

7.7 Project status

All figures below are measured against the consolidated repository tree of **2026-08-23** (commit `90792f5`) and each carries its row in Appendix C; a figure without a row would be a defect by the ledger's own rule.

The system runs **15 services** (E-01) with **26 schema migrations** (000–025) applied (E-02), on a runtime pinned to **Agents v0.20.0** (SHA `9d4ef04ed`; E-03). Two client installations operate on dedicated hardware, each a separate deployment with its own perimeter; one runs an eight-agent operations unit with a live messaging channel (E-04). The **closed-beta gate was declared met on 2026-07-21**, with an external tester completing onboarding the same day (E-05).

Security posture, since v3 reported it as a blocker: authentication is enabled and internal services require a credential on every mutation (2026-08-08; E-06); per-user model credentials are encrypted with a per-installation key (2026-08-09); network egress is scoped **per department**, i.e. per container, not per agent (2026-08-09; E-

19); daily verified backups run on both client installations, and the first restore rehearsal **failed**, exposing four defects that only execution reveals (2026-08-19; E-20). Reproducibility of the deployment itself is uneven: thirteen container images track a mutable tag and one is pinned by digest (E-24).

Two capabilities absent from v3 entirely: an **ontology with a closed vocabulary** whose guardian fails the build when code emits an undeclared node or edge type, and a **graph and reasoning layer** integrated beneath the console with per-person identity at the edge (E-18). Both are entered in the ledger with their limits, of which the material one is that the graph layer has **no tenant isolation** and is therefore restricted to a single permission.

The test suite comprises **143 Python test files defining 1,212 test functions**, plus **34 front-end test files** (E-21). A full-suite execution reported by an independent review on 2026-08-22 counted **1,532 passing**; that count is entered in the ledger as *inherited, not re-run for this draft* (E-22), since a parametrized suite yields more cases than the static count of definitions.

7.8 Threats to construct validity specific to SIS

The gym measures *comparability between profiles of the same installation under controlled difficulty*, not absolute quality; it must not be read as a public benchmark. Deterministic criteria reward pattern-matchable correctness and can miss qualities only a strong judge would catch — hence the held-out anchor, with the caveat of §7.3 that the anchor does not currently block. The live F2 result (§7.2) is a *single dated run* on a small matrix; no population-level trajectory is claimed.

Two further limits specific to this draft. The conformance result of §7.5 measures one protocol's `core` class against one endpoint, and generalizes to nothing else. And the security posture reported in §7.7 is read from configuration and code, not from a live probe of a running installation; §7.6 states explicitly where that distinction matters.

8. Threats to validity

8.0 Half of a pattern, and who is building the other half

This system's actor identifier adopts the member form of X-Road — instance, class, member, subsystem — a cross-organizational exchange layer with more than twenty years in national production. The adoption is real and it is in the code, but it is **partial in a way worth declaring**: we took the **transport** half and skipped the **catalogue** half. In the original pattern there is a central registry stating which organizations and which services are legitimate, signed and distributed to participants. Here that registry exists as a database table consulted locally. **A directory that is neither signed nor distributed is not a federated catalogue; it is a list.**

This is not an abstract omission. On 16 June 2026 Estonia — the jurisdiction that originated X-Road — approved issuing autonomous agents identity codes of their own, distinct from their human principal, scoped and auditable through that same layer, and tasked its state agency with building **a central trust registry for agents** across the public and private sectors, with twenty-four months and a one-million-euro budget.

That fact cuts in two directions and both matter. It **validates** the decision to give an agent an identity shaped as a member of a public interoperability layer, taken here independently ten days earlier. And it **warns**: the piece this system declares missing is exactly the one a jurisdiction is going to build with budget and a schedule. Interoperating with it will be a condition, not an option — and arriving with half the pattern implemented is a better position than arriving with none, but a worse one than that of whoever builds it.

Internal. Fitness self-triggers depend on thresholds (`SOCIETY_FITNESS_*`) whose defaults are engineering guesses; mis-set thresholds could advance a rung prematurely. The mitigation is layered: advancement is *read-only/shadow* until F3-active, F3-active is fail-open (degrades to shadow when θ is unstable or the fit is unconverged), and F5 is hard-gated on F2–F4. **We previously cited the live `converged=False` block as evidence that the mitigation engages in practice; §7.3 retracts it.** The layering is a design property we have not demonstrated in operation, and one of its layers — the θ -stability trigger — was found to satisfy itself on unchanged data.

External. The reference topology (a 14–16-unit "MAP" demo) is a *dataset*, not a claim about any real institution; all labels come from YAML/spec. Results on canonical gym tasks may not transfer to open-ended institutional work.

Construct. "Better-than" as adjudicated by an LLM judge can encode the judge's biases; the two-order protocol, a deterministic held-out anchor, and the documented judge-choice sensitivity (§5.5.1) bound but do not eliminate this. "Identity persists under model change" is a *design* guarantee (preserved Ψ), not an empirical claim about behavioral invariance — a swapped model will behave differently even with identical SOUL/memory; IdentIA's claim is about *identity*, not *behavioral equivalence*, a distinction we make explicit to avoid overreach and which §D.2 states as an equation: $b_t = f(\Psi, m_t) + \varepsilon_t$ — swapping the measurement changes both the map's second argument and the residual's distribution, outside Ψ 's control.

Reproducibility. Several claims rest on internal runs (e.g. judge position-bias findings, the single live F2 run); we report them as observations, not controlled experiments, and the gym is provided precisely to make them reproducible.

9. Discussion

9.1 Interpretation

SIS's wager is that *persistence plus measured selection plus governance* is a better substrate for trustworthy agent organizations than orchestration alone. The selection ladder is the system's answer to a recurring failure mode of self-improving systems: rewarding what is *declared* better rather than what is *measured* better. By making the *order* of capabilities the safety argument — reliability and diversity before recursion — SIS converts "be careful with RSI" from a slogan into a dependency graph. Whether that graph is *load-bearing* in production rather than only in tests is not something we can currently claim: the live block we used to offer as evidence is retracted (§7.3), and F3 is inactive by decision rather than by a measured gate.

9.2 Limitations

The most consequential limitations are: (i) the top rung (F5) is unbuilt and the closed-loop rungs (F3–F4) run only in shadow, so the central "self-improving" claim is, today, a *reliable-selection layer with a gated, not-yet-active recursion* — the selection half is measured and partly live, the recursive half is specified; (ii) **no live run establishes that the reliability gate withholds activation** — the one we previously reported was an artifact of our own convergence criterion and is retracted (§7.3), so the gate's behaviour under a real matrix is untested; (iii) federation is audited but not cryptographically signed, so cross-host trust currently rests on policy and network boundaries, not cryptography; (iv) multi-tenancy is incomplete, bounding external deployment, and the auth backend is validated only in staging; (v) the A → C bridge's emergent degree and the episodic-context read-side are deliberately deferred. We state these plainly because the paper's value depends on its claims being checkable.

9.3 Ethics and risk

A platform that simulates ministries and firms, accumulates institutional memory, and (eventually) edits its own scaffold raises real concerns: concentration of power, dependence on LLM providers, sensitive data in per-agent RAG, and the dual-use nature of recursive self-improvement. SIS's mitigations are structural — governance L1–L8, per-goal budgets, quarantine, archive-not-delete, human gates on cultural transmission and on RSI, a global RSI pause/rollback to a fitness-dominant snapshot, and the metacentrality discipline of audit/rotation/transparency/ self-limitation — the last of which is a **stated posture, not an enforced mechanism** ([PB] §5). None of these is sufficient alone; we present them as a defense-in-depth posture and invite adversarial scrutiny, especially of F5's gates and of the federation trust model in the absence of envelope signing.

10. Conclusion

Self-Improving Society reframes the unit of agentic computing from the *process* to the *institution*. This paper reports the part of that reframing that is built and measured: **IdentIA** as an identity ontology whose principles are enforced as code invariants rather than asserted in prose; an **architecture** whose deployment unit is a sovereign organizational unit wrapped by an exoharness that governs an interchangeable runtime instead of reimplementing it; three implemented rungs of a measured-selection ladder ending in an **Item-Response-Theory reliability gate**; and **governance-as-code** with a claims ledger that makes status drift visible in one place.

The empirical result this paper used to report has been retracted (§7.3). We had read the live run of 2026-06-18 as the gate declining to act on its own unreliable measurement; re-measurement showed the refusal was a property of our convergence criterion, which could not return `True` whatever the data. What remains is narrower and worth stating plainly: the estimator recovers difficulty and separates ability on a small matrix, the components are built, and **no live result yet shows the gate withholding anything**. F3 is off by decision, not by measurement.

The episode is the paper's method turned on the paper. A figure was recorded with its date, its source and the limit *not re-measured since*; that limit is what produced the re-measurement three months later, and the re-measurement destroyed the claim. We would rather publish a bookkeeping system that catches its author than a result that survives because nobody checked it again.

What this paper does not do is argue that the ladder's *order* is the safety property. That argument, the upper rungs it concerns, and the normative thesis about institutional power are the companion position paper **[PB]** — kept separate because none of it has run, and a systems paper that carries an untested argument alongside a measured result lends the first the credibility of the second.

Concrete future work, in dependency order rather than as a list of parallel tasks: re-run the IRT rung on a client installation rather than a development one, so that a converged fit exists outside the environment where the criterion was repaired; only then activate F3, and F4 after it; add cryptographic envelope signing; bind approval decisions to a verified principal and hash-chain the audit log (§7.4, E-16); and run a longitudinal study of a federated society's fitness trajectory.