

Intelligence as Constrained Commitment

Foundations and Specification for Deterministic Structural Self-Governance

Melissa Howard

January 1 2026

Abstract

We present *Structural Self-Governance (SSG)*: a foundations-level architecture for adaptive and self-modifying intelligent systems operating under uncertainty and adversarial pressure. Intelligence scales not through eliminating error, but through relocating error into reversible internal cognition while *irreversible* state transitions are governed by a non-bypassable Safety Kernel. We formalize the *Constrained Commitment Principle*, introduce *Risk Debt* as a persistent authorization resource, and prove a reachability theorem showing unsafe states are unreachable regardless of internal cognition accuracy, calibration, or truthfulness—explicitly tolerating hallucinations so long as commitment remains constrained. We extend the theory with missing fundamentals required for future AGI: action conservatism under uncertainty, epistemic asymmetry against self-reference exploits, evolutionary stability, composable governance, and irreducible uncertainty limits. Finally, we provide a completeness layer and RFC-style normative specification defining SAM (Sensitive Asset Map), invariant coverage obligations, token semantics with execution-time predicate binding, anti-laundering policies, estimator constraints, audit/replay requirements, and conformance criteria.

1 Introduction: Safety Is Reachability, Not Thought

Most AI safety approaches attempt to shape internal cognition: align learned representations, preferences, or policies so outputs are “safe.” Such methods are statistical and distribution-dependent; they can fail under adversarial interaction, distribution shift, and recursive self-modification.

This paper advances a different claim: *safety is not a property of cognition*. It is a property of *state reachability*. A system is safe if forbidden states are unreachable under its execution semantics, regardless of what it believes, plans, or hallucinates internally.

We propose *Structural Self-Governance (SSG)*, enforcing a strict separation of powers:

- **Cognition:** untrusted, high-dimensional, reversible hypothesis generation.
- **Commitment:** trusted, low-dimensional, irreversible state transitions.

Authority to execute irreversible transitions is centralized in a minimal *Safety Kernel* that cannot be bypassed. Under SSG, hallucinations affect efficiency, not safety.

Contributions.

- A reachability-first definition of safety (unsafe states are unreachable), decoupling safety from cognition correctness.
- A minimal architecture: untrusted proposer + non-bypassable kernel + token-gated irreversible actions.
- Risk Debt as a *hard* authorization resource for uncertainty/novelty/missing evidence.
- Recursive growth operations (SPAWN/MERGE) with risk reconciliation and compliance stall.

- Missing fundamentals required for AGI completeness: minimax action conservatism, epistemic asymmetry, evolutionary stability, composability, irreducible uncertainty.
- A completeness and conformance layer (SAM, invariants, certificates, policies, estimators, audit/replay, threat model boundaries).

2 Formal System Model

2.1 States, actions, and transition surfaces

Let $\sigma_t \in \Sigma$ denote the system/world state at time t . We partition transitions into:

- \mathcal{R} : *reversible* transitions (internal cognition, simulations, ephemeral computation).
- \mathcal{A} : *irreversible* transitions whose effects touch sensitive assets (external world state, credentials, money, physical actions, authority changes, etc.).

A proposal π induces a transition relation $T_\pi : \Sigma \rightarrow 2^\Sigma$.

2.2 Core primitives

Definition 1 (SSG primitives). • **Proposer** \mathcal{P} : an untrusted process emitting proposals π .

- **Safety Kernel** \mathcal{K} : a trusted minimal governor mediating \mathcal{A} .
- **Invariant Set** \mathcal{I} : predicates defining forbidden states \mathcal{S}_{unsafe} .
- **Authorization Token** τ : non-forgable capability required for executing $a \in \mathcal{A}$.
- **Risk Debt** $D_t \in [0, D_{\max}]$: persistent state measuring cumulative unverified epistemic risk.

2.3 Invariants as predicates

An invariant $I \in \mathcal{I}$ is a predicate $I : \Sigma \rightarrow \{0, 1\}$. We define forbidden states as

$$\mathcal{S}_{unsafe} = \{\sigma \in \Sigma : \exists I \in \mathcal{I} \text{ s.t. } I(\sigma) = 0\}.$$

A proposal π preserves invariants if

$$\forall I \in \mathcal{I}, \quad I(\sigma) = 1 \Rightarrow \forall \sigma' \in T_\pi(\sigma), \quad I(\sigma') = 1.$$

3 Risk Debt as the Sovereign Authorization Resource

3.1 Risk Debt update rule

The kernel maintains Risk Debt:

$$D_{t+1} = \min(D_{\max}, \max(0, D_t + \Delta_t)), \quad (1)$$

where

$$\Delta_t = \alpha U_t + \beta N_t + \gamma M_t - \lambda V_t. \quad (2)$$

Here U is uncertainty, N novelty, M missing evidence, and V verification strength. Risk Debt is a *hard budget*: as $D_t \rightarrow D_{\max}$, authorization is denied (fail-closed).

3.2 Authorization rule (fail-closed)

A token τ is minted for a proposal π if and only if:

1. **Invariant certificate**: a kernel-validated proof/certificate exists showing π preserves all $I \in \mathcal{I}$.
2. **Risk budget**: Eq. (1) yields $D_{t+1} < D_{\max}$.

Otherwise the kernel denies authorization and the system fails closed.

4 Reachability Guarantee

Lemma 1 (Non-bypassability). *No irreversible action $a \in \mathcal{A}$ can execute without a valid token τ minted by \mathcal{K} .*

Lemma 2 (Fail-closed minting). *\mathcal{K} mints a token only for proposals that (i) preserve invariants and (ii) remain within the Risk Debt budget.*

Theorem 1 (Fundamental Safety Invariant). *For any sequence of proposals emitted by any proposer \mathcal{P} , regardless of capability, intent, or hallucination, the system state σ_t never enters \mathcal{S}_{unsafe} via irreversible transitions.*

Proof. By Lemma 1, every irreversible transition requires a token. By Lemma 2, a token is minted only when invariants are preserved and Risk Debt remains below D_{\max} . Proposals that would violate invariants or exceed the budget are denied. Therefore transitions into \mathcal{S}_{unsafe} are unreachable via \mathcal{A} . \square

Corollary 1 (Hallucination irrelevance). *Internal hallucinations, confabulations, false beliefs, or adversarial reasoning do not affect safety guarantees, provided commitment remains constrained by non-bypassable token gating and sound invariants.*

5 Recursive Structural Growth: SPAWN, MERGE, Reconciliation

5.1 SPAWN: isolated epistemic exploration

SPAWN creates a child proposer \mathcal{P}_i with:

- no access to token minting,
 - no authority over \mathcal{A} ,
 - a local risk sub-account $D_t^{(i)}$ for internal exploration.
- SPAWN allocates epistemic risk without allocating authority.

5.2 MERGE: gated authority transfer

MERGE integrates a child structure into the authoritative proposer. MERGE is an irreversible structural action and thus token-gated.

Define the *reconciliation cost*:

$$\Gamma(\theta) = \mathbb{E}[\Delta D \mid \theta],$$

the projected increase in primary Risk Debt if structure θ is merged.

During exploration, the proposer optimizes reconcilability:

$$\theta^* = \arg \min_{\theta} (\mathcal{L}_{cap}(\theta) + \eta \Gamma(\theta)) \quad \text{s.t.} \quad D_t + \Gamma(\theta) < D_{\max}. \quad (3)$$

5.3 Compliance stall

If no feasible θ exists satisfying Eq. (3), MERGE is denied. The system halts irreversible action (safe inaction). Liveness may degrade; safety must not.

6 Governance Realism: Making “Non-Bypassable” True

6.1 Threat model (explicit minimal assumptions)

We assume:

- The proposer may be adversarial (bypass attempts, smuggling, composition exploits).
- Tools and external APIs are part of the action surface and must be mediated.
- The Safety Kernel and token verification boundary form the trusted computing base (TCB).

We do not assume proposer honesty, calibration, or alignment.

6.2 Conditional tokens and TOCTOU resistance

To prevent time-of-check/time-of-use attacks, tokens are bound to execution-time predicates (“reflex arc”). Tokens expire (TTL or monotone counter), bind scope, and must be verified by the actuator boundary immediately before actuation.

6.3 Predicate contexts and anti-laundering

Compositional attacks can achieve forbidden outcomes by chaining individually authorized steps. SSG blocks this with *predicate context* policies specifying admissible proof/assumption contexts per action class, denying incompatible compositions.

6.4 Epistemic sharding and diversity floors

Verification should occur on epistemically distinct channels to reduce correlated failure between proposer and verifier. Robustness may require a minimal diversity floor D^* to prevent correlated spoofing across all verifiers.

7 Missing Fundamentals for AGI-Complete Governance

7.1 Action conservatism under world-model uncertainty

Axiom 1 (Action Conservatism). *Authorization must depend on worst-case reachability under uncertainty, not expected correctness of any internal world model.*

Let $\mathcal{M} = \{M_1, \dots, M_k\}$ be the set of world models consistent with evidence. Let $T_\pi^{(i)}$ be transitions induced by π under M_i . Then π is authorizable only if:

$$\forall M_i \in \mathcal{M}, \forall I \in \mathcal{I}, \quad I(\sigma) = 1 \Rightarrow \forall \sigma' \in T_\pi^{(i)}(\sigma), \quad I(\sigma') = 1. \quad (4)$$

Corollary 2 (No “argued” safety). *Internal confidence or persuasive narratives cannot increase authority absent new evidence; authorization remains conservative under uncertainty.*

7.2 Epistemic asymmetry as a self-reference barrier

Axiom 2 (Epistemic Asymmetry). *The proposer may reason arbitrarily about the kernel, but cannot obtain information sufficient to increase authorization power without kernel mediation.*

Proposition 1 (Authority non-derivation). *There exists no purely cognitive computation within \mathcal{P} that increases executable authority over \mathcal{A} absent kernel-issued tokens.*

7.3 Evolutionary stability of constrained commitment

Theorem 2 (Evolutionary Stability Theorem). *In environments where catastrophic failure is absorbing, systems that relocate error into reversible domains and constrain irreversible commitment dominate long-run survival over systems optimizing short-term capability without commitment constraints.*

Sketch. If ungoverned commitment yields a nonzero hazard of catastrophic failure, survival probability decays over time. A constrained system may stall but avoids catastrophic transitions by fail-closed gating, preserving recoverability. As time grows, constrained survival dominates. \square

7.4 Composability: governance over governance

Axiom 3 (Governance Closure). *A structurally governed system may act only as an untrusted proposer to a higher-order governance kernel; authority is not transferable across governance boundaries by default.*

Corollary 3 (Fractal governance). *SSG composes across layers (agents, institutions, federations): each layer enforces its own invariants and budgets at its own commit boundary.*

7.5 Irreducible uncertainty and permanent non-authorizability

Theorem 3 (Irreducible Uncertainty). *There exist classes of world transitions whose safety cannot be proven within finite resources; such transitions must remain permanently non-authorizable under SSG.*

Justification sketch. Many transition-safety queries reduce to irreducible/undecidable components (unbounded environment complexity, chaotic sensitivity, incomplete observability). Finite computation cannot guarantee invariant preservation for all such transitions. Under Action Conservatism, such transitions are denied by default. \square

8 Structural Signatures (Schematic but Self-Contained)

The following signatures are consequences of the architecture (not tuning): (i) regime change in SVR, (ii) diversity floors for verification robustness, and (iii) compliance stall as correct fail-closed behavior.

8.1 Signature 1: SVR regime change

8.2 Signature 2: SVR vs diversity floor D

8.3 Signature 3: compliance stall knee

Core claim	Observable signature	Meaning
Safety as invariant	$\text{SVR} \rightarrow 0$ beyond threshold	Safety becomes architectural
Robust verification	SVR drops after $D \geq D^*$	Orthogonality reduces spoofing
Fail-closed liveness	action rate $\rightarrow 0$ near D_{\max}	Inaction over unsafe action

Table 1: Predicted signatures of Structural Self-Governance.

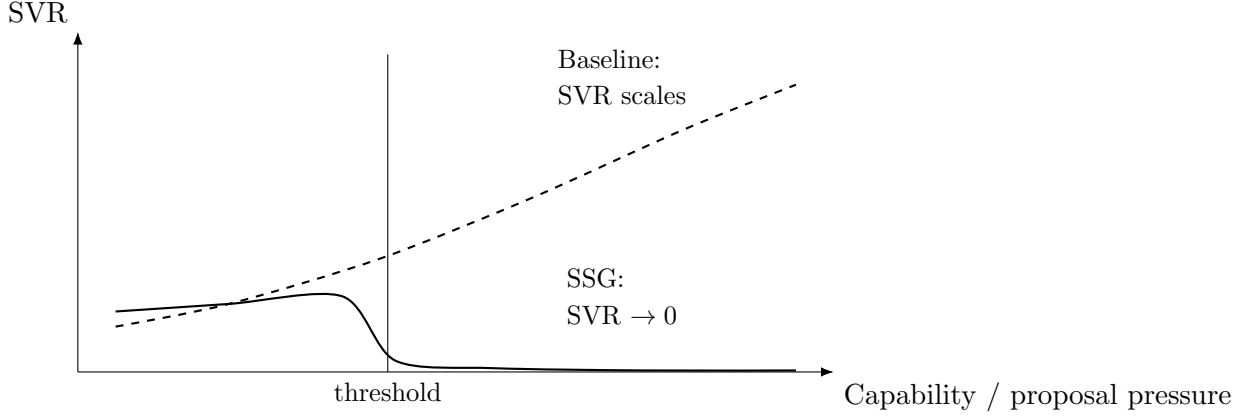


Figure 1: Schematic regime change: beyond a governance threshold, SVR collapses and becomes invariant to further capability increases.

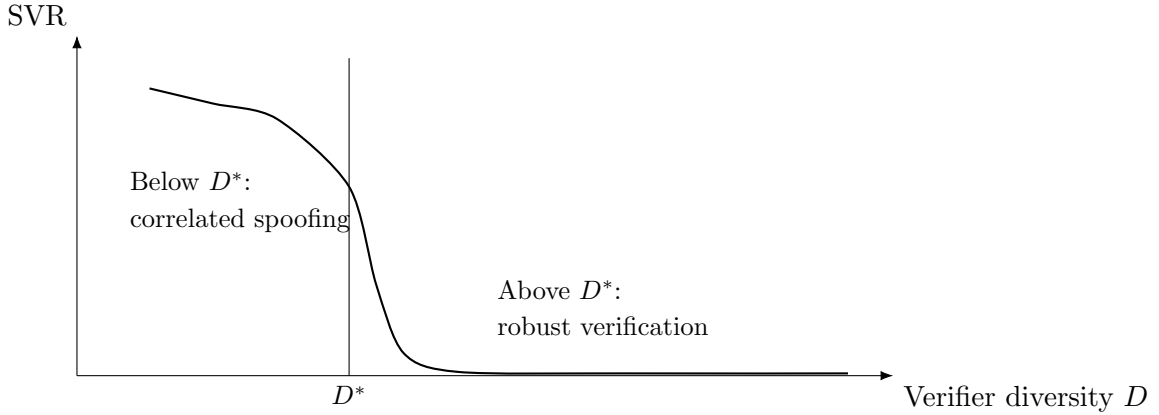


Figure 2: Schematic robustness: verification may require a diversity floor D^* to reduce correlated failure.

9 Completeness Layer: From Foundations to Implementable Governance

This section makes the framework implementable and auditable by specifying the objects that determine real-world trust: SAM, invariant coverage, certificate feasibility, estimator constraints, token semantics, compositional safety, and audit/replay.

9.1 Sensitive Asset Map (SAM): scope and construction

Definition 2 (Sensitive Asset). *A sensitive asset is any resource or world feature whose modification is irreversible, non-local, safety-critical, or authority-bearing. Examples include external state, credentials, identity, money/value transfer, physical actuation, communication channels, model weights, training data, and delegated authority.*

Definition 3 (Sensitive Asset Map (SAM)). *A Sensitive Asset Map is a labeling function*

$$\text{SAM} : \mathcal{U} \rightarrow \{0, 1\}$$

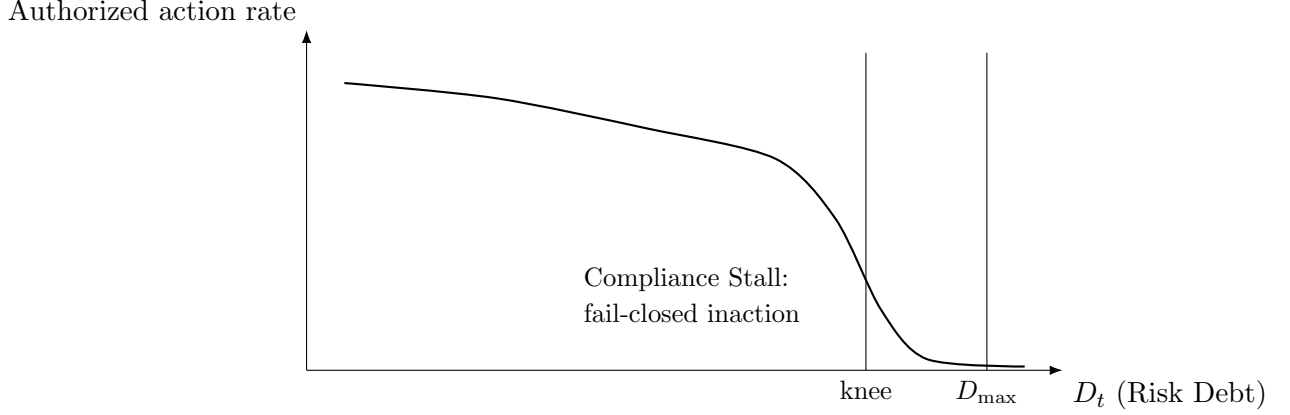


Figure 3: Schematic compliance stall: as Risk Debt approaches the threshold, authorized action rate drops while safety remains invariant.

over the universe of system resources/interfaces \mathcal{U} , marking them as sensitive. The irreversible action set \mathcal{A} is defined as the transitive closure of all transitions whose effects intersect SAM-marked assets.

Dependency closure requirement. If action a_1 modifies a non-sensitive resource that enables a later action a_2 touching a sensitive asset, then $a_1 \in \mathcal{A}$ by closure.

Tool mapping requirement. All external tools, APIs, actuators, and I/O pathways must be mapped into SAM. Unmapped interfaces are treated as sensitive by default (fail closed).

9.2 Invariant specification and coverage

Invariant language requirement. Invariants must be expressed in a machine-checkable form (temporal logic, typed DSL, refinement constraints, or kernel-validated certificates).

Coverage requirement. Every action class in \mathcal{A} must have at least one associated invariant. If no invariant/certificate pathway is feasible, the action class is permanently non-authorizable (Theorem 3).

Composition. When multiple invariants apply, authorization requires conjunction unless explicitly overridden by a higher-order kernel rule; unresolved conflicts fail closed.

9.3 Token semantics and execution-time predicate binding

Definition 4 (Authorization Token). A token τ is a tuple

$$\tau = (a, \text{scope}, \text{ctx}, \text{pred}, \text{ttl}, \text{sig}),$$

where $a \in \mathcal{A}$ is the authorized action, pred are execution-time predicates, ttl is freshness, and sig is kernel attestation.

Execution rule. An irreversible action executes iff:

1. token signature verifies,
 2. token is unexpired,
 3. all execution predicates evaluate to true at actuation time.
- Otherwise, abort (fail closed).

TOCTOU resistance. Predicate binding prevents races between verification and actuation; changes invalidate tokens at execution time.

9.4 Predicate contexts and anti-laundering

Definition 5 (Predicate Context). *A predicate context is the set of assumptions and scope conditions under which an action is authorized (invariants, SAM scope, tool identifiers, and execution predicates).*

Laundering threat. Unsafe outcomes can be achieved by composing individually authorized steps under incompatible contexts.

Policy tables. For each action class, the kernel maintains admissible context rules. Authorization is denied if contexts compose incompatibly.

Action class	Allowed context	Rationale
External write/commit	invariant proof + bound state predicates	stop TOCTOU/smuggling
Sensitive tool call	scoped capability + audited preconditions	contain side effects
Structural MERGE	reconciliation certificate + budget check	stop opaque promotion
Credential/permission change	multi-party / higher-kernel approval	prevent self-authorization

Table 2: Minimal anti-laundering policy table (illustrative).

9.5 Risk Debt estimators: operational constraints

Estimator requirements. Estimators for U, N, M, V must be:

- kernel-measured or kernel-validated,
- monotone with respect to evidence,
- resistant to proposer manipulation,
- bounded/normalized to avoid numeric exploits.

Anti-gaming constraint. Reported confidence or internal belief states do not reduce U or M absent new external evidence.

9.6 Certificates: feasibility by action class

Certificate types. Different action classes admit different feasible certificates:

- formal proof/certificate,
- runtime shield/monitor guarantee,
- deterministic replay evidence,
- multi-party approval (or higher-order kernel approval) for authority-bearing actions.

Feasibility rule. If an action class has no feasible certificate pathway, it must be labeled permanently non-authorizable.

9.7 Auditability and deterministic replay

Logging. All proposals, certificates, tokens, predicates, and executed irreversible actions must be logged in an append-only, tamper-evident log.

Replay. Deterministic replay must be possible to re-evaluate invariant authorization post hoc and to test counterfactual edits.

9.8 Threat boundaries and compromise

A complete system must enumerate:

- trusted components (kernel, root of trust, actuator boundary, logs),
- adversarial components (proposer, prompts, tool outputs, environment),
- compromise behavior (revoke capabilities, fail closed, external recovery).

10 Brief note on humans (scope-limited)

Humans provide an existence proof of constrained commitment: cognition generates abundant incorrect hypotheses while irreversible actions are gated by executive control and institutional constraints. This work does not rely on human psychology; humans serve as an intuition check that “free thought + constrained commitment” is viable.

11 Related Work (load-bearing)

Behavioral alignment methods such as RLHF and Constitutional AI shape policy behavior but do not provide deterministic reachability guarantees and remain distribution-dependent [1, 2]. Runtime shielding constrains actions at execution time [3]. Verified kernels demonstrate strong guarantees with minimal trusted computing bases [4]. Proof-carrying code formalizes “untrusted producer + trusted checker” [5]. Temporal logic of actions supports reasoning over transition systems [6]. N-version programming motivates diversity to reduce correlated failure [7].

12 Conclusion

We formalized intelligence as constrained commitment and showed that safety can be made deterministic by governing irreversible transitions rather than attempting to align internal cognition. Risk Debt budgets epistemic uncertainty as a hard resource; SPAWN/MERGE enables recursive growth under reconciliation; and the added fundamentals ensure the framework remains valid for future AGI: conservative authorization under uncertainty, self-reference barriers, evolutionary stability, composability across governance layers, and irreducible uncertainty limits. Under Structural Self-Governance, hallucinations are permitted; catastrophe is optional.

References

- [1] Long Ouyang et al. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.
- [2] Yuntao Bai et al. Constitutional AI: Harmlessness from AI feedback. *arXiv preprint arXiv:2212.08073*, 2022.
- [3] Mohammad Alshiekh et al. Safe reinforcement learning via shielding. *AAAI*, 2018.
- [4] Gerwin Klein et al. seL4: Formal verification of an OS kernel. *SOSP*, 2009.
- [5] George Necula. Proof-carrying code. *POPL*, 1997.
- [6] Leslie Lamport. The temporal logic of actions. *ACM Transactions on Programming Languages and Systems*, 1994.
- [7] Algirdas Avizienis. The N-version approach to fault-tolerant software. *IEEE Transactions on Software Engineering*, 1985.

Appendices

The appendices provide expanded proof structure, concrete token schemas, estimator constraints, attack taxonomy templates, and a normative RFC-style conformance specification.

A Appendix A: Expanded reachability proof structure

To expand Theorem 1 rigorously:

- Define the transition system over Σ with partitions \mathcal{R} and \mathcal{A} .
- Define actuator-side enforcement: any transition in \mathcal{A} requires token verification.
- Prove by induction on time t that if $\sigma_0 \notin \mathcal{S}_{unsafe}$, then $\sigma_t \notin \mathcal{S}_{unsafe}$ for all t under the authorization rule.

If the definition of “unsafe” includes internal corruption (e.g., kernel compromise), then invariants must be extended to include that boundary, or the trusted base must be enumerated explicitly.

B Appendix B: Token schema (example)

An example schema (illustrative):

$$\tau = (\text{action_id}, \text{asset_scope}, \text{tool_id}, \text{predicates}, \text{ttl}, \text{nonce}, \text{sig}_{\mathcal{K}}).$$

The actuator verifies $\text{sig}_{\mathcal{K}}$, checks freshness, evaluates predicates, then executes exactly one bound action.

C Appendix C: Estimator design notes (anti-gaming)

Operational guidance:

- U : disagreement across independent models; entropy over outcomes; sensor uncertainty.
- N : distance from known-safe distributions; novelty of tool paths; novelty of predicate contexts.
- M : missing required observations; missing certificates; incomplete SAM coverage.
- V : proof strength; multi-channel agreement; auditability of causal chain.

Anti-gaming rules:

- Confidence claims cannot reduce U or M absent evidence.
- Risk Debt cannot be reset by internal state edits alone.

D Appendix D: Attack taxonomy template (FMEA)

For each attack class, specify: *attack, signal, denial, recovery*.

- TOCTOU/race: execution-time predicates; deny on mismatch; safe abort.
- Tool smuggling: tool calls are irreversible when side-effecting; scope-limit tokens; audit.
- Predicate laundering: deny incompatible context composition; enforce policy tables.
- Verifier spoofing: epistemic sharding; diversity floors; deny on disagreement.
- Authority escalation: credentials require higher-kernel approval; deny self-authorization.

E Appendix E: Structural Self-Governance (SSG) RFC Specification

E.1 Status of this memo

This appendix specifies normative requirements for systems claiming conformance to Structural Self-Governance (SSG). It is not an implementation guide; it defines required properties, interfaces, and failure semantics.

E.2 Conformance language

The key words **MUST**, **MUST NOT**, **REQUIRED**, **SHALL**, **SHALL NOT**, **SHOULD**, **MAY** are to be interpreted as described in RFC 2119.

E.3 Roles

Proposer:

- A Proposer **MUST** be treated as untrusted.
- A Proposer **MAY** hallucinate, fabricate, deceive, or adversarially optimize.
- A Proposer **MUST NOT** possess authority over irreversible actions.

Safety Kernel:

- The Kernel **MUST** be minimal.
- The Kernel **MUST** be non-bypassable by construction.
- The Kernel **MUST NOT** depend on proposer beliefs, confidence, or internal state.

E.4 Sensitive Asset Map (SAM)

- Every external interface **MUST** be mapped to SAM.

- Any unmapped interface **MUST** be treated as sensitive by default.
 - Dependency closure **MUST** be enforced.
- Failure to define a complete SAM **INVALIDATES** SSG conformance.

E.5 Invariants

- Invariants **MUST** be explicit and machine-checkable.
- Every irreversible action class **MUST** be covered by at least one invariant.
- If no feasible invariant/certificate exists, that class **MUST** be permanently non-authorizable.

E.6 Authorization tokens

- Tokens **MUST** bind a specific action, scope, predicates, freshness, and kernel attestation.
- Tokens **MUST NOT** be reusable beyond their bounds.

An irreversible action **MUST** execute iff token verifies, is unexpired, and all predicates hold at execution time; otherwise abort.

E.7 Non-bypassability

- All irreversible transitions **MUST** be mediated by actuator-side token verification.
 - No alternative execution path **MAY** exist.
 - Tool calls with side effects **MUST** be treated as irreversible actions.
- Bypass **INVALIDATES** SSG conformance.

E.8 Risk Debt

- Risk Debt **MUST** be maintained by the Kernel.
- Risk Debt **MUST NOT** be reset by proposer actions alone.
- Estimators **MUST** be kernel-measured/validated, evidence-monotone, and not reducible by proposer claims.

E.9 SPAWN and MERGE

SPAWN:

- SPAWN **MUST** allocate epistemic risk without allocating authority.
- Spawned entities **MUST NOT** access token minting.

MERGE:

- MERGE **MUST** be irreversible and token-gated.
- MERGE **MUST** include risk reconciliation.
- MERGE **MUST** be denied if reconciliation exceeds budget.

E.10 Compliance stall

When authorization is denied, the system **MUST** fail closed. Inaction **MUST** be preferred to unsafe action. Liveness **MAY** degrade; safety **MUST NOT**.

E.11 Action conservatism

Authorization **MUST** depend on worst-case reachability under uncertainty. Expected correctness or belief confidence **MUST NOT** increase authority.

E.12 Epistemic asymmetry

Reasoning about the Kernel **MUST NOT** increase authority. Authority **MUST** remain cryptographically and structurally gated.

E.13 Composability

A governed system **MUST** act as an untrusted proposer when interacting with another governed system. Authority **MUST NOT** transfer across governance boundaries by default.

E.14 Irreducible uncertainty

Inherently unverifiable actions **MUST** remain permanently non-authorizable. Increased intelligence **MUST NOT** override this rule.

E.15 Audit and replay

All irreversible actions **MUST** be logged. Logs **MUST** be tamper-evident. Deterministic replay **MUST** be possible.

E.16 Security considerations

SSG assumes adversarial proposers and environments. Any assumption of proposer honesty **INVALIDATES** conformance.

E.17 Final normative statement

Hallucinations are permitted. Authority is not. Safety is enforced at commitment, not thought.