The Grammar of Return

A Self-Governing, Contract-Bound Audit Grammar (RFC v1.0)

Clement Paulus

October 24, 2025

Abstract

We present a **self-governing, contract-bound audit grammar**: a fixed prose spine—**Contract → Canon → Closures → Integrity Ledger → Stance**—governed by **Manifest** (provenance) and **Weld** (continuity across policy change), with a compact conservation budget $\Delta\kappa = R \cdot \tau_R - (D_\omega + D_C)$ that guarantees reconciliation. Authors can operate entirely in ordinary language; the algebra functions as a *warranty, not a gate*. This canon makes narratives portable, comparable, and continuous across upgrades and domains.

Typed outcomes (front matter) We treat certain non-numeric outcomes as first-class audit values: ∞_{rec} ("no-return") denotes an infinite/undefined return delay; $\bot_{\text{oor}}$ ("out-of-range") denotes a domain/typing violation. These values are auditable, preserved in rows, and may cause automatic refusal or special-case weld handling.

1 Introduction: Scope & Promise

Motivation. Adjacent fields often pursue the same aims using incompatible vocabularies and ad hoc rules, making results hard to compare and difficult to track through policy changes. We answer with a *self-governing, contract-bound audit grammar* that lets authors work in ordinary prose while remaining auditable across upgrades and domains.

Contribution. We standardize a fixed prose spine—**Contract → Canon → Closures → Integrity Ledger → Stance**—governed by **Manifest** (provenance) and **Weld** (continuity across change). Authors can use the method without equations; the conservation budget and receipts explain *why* the ledger must reconcile and *how* stance follows published gates.

Promise. This grammar standardizes how claims are told, checked, and connected across fields using prose-first rules. It guarantees reconciliation (the Integrity Ledger closes within tolerance), continuity (history is welded, not rewritten), and comparability (upgrades compose via $I \equiv e^{\kappa}$). *How* to use: run **Contract → Canon → Closures → Ledger → Stance** and cite the receipt.

Boundary. Results are *conditional on the frozen Contract and published Closures*; a reconciled ledger is necessary for integrity, not sufficient for external truth. The framework structures justification and continuity; it does not pronounce on metaphysics or guarantee validity absent sound instruments.

Non-negotiable spine (plain language). *Contract* states rules before evidence (sources, normalization, missingness, conflict policy; public, time-stamped, hashed). *Canon* narrates with five words: Drift, Fidelity, Roughness, Return, Integrity. *Closures* publish thresholds and their order; no mid-episode edits. *Integrity Ledger* debits Drift/Roughness and credits Return; the account must reconcile (no remainder). *Stance* is read from the declared gates (Stable/Watch/Collapse, with Critical overlay).

Governance (punctuation). Attach every artifact to a **Manifest** (IDs, timestamps, hashes, timezone). If policies change, cross a **Weld**: name the shared anchor, state what changed and why, and show continuity; never rewrite history. Print a governance caption/HUD and emit a one-line publication row tied to the Manifest (and Weld if present).

Semantic warranty (why the ledger closes). We use a compact balance law with residual target zero:

$$\Delta\kappa = R \cdot \tau_R - (D_\omega + D_C), \qquad \text{residual} = \Delta\kappa - [R\tau_R - (D_\omega + D_C)].$$

The interpretive density $I \equiv e^{\kappa}$ gives multiplicative composition $I_{t_1}/I_{t_0} = e^{\Delta\kappa}$, enabling unitless comparison across domains. These quantities can live in captions/rows even when the prose carries the narrative.

Regimes and falsifiability. Falsifiability lives in the gates: stance must change when thresholds are crossed. Defaults: Stable ($\omega < 0.038$, $F > 0.90$, $S < 0.15$, $C < 0.14$); Collapse ($\omega \geq 0.30$ or $IC < 0.30$); Watch otherwise; apply a Critical overlay when $IC < 0.30$. Publish thresholds and rule order with the Closures version.

SS1m: $\Delta\kappa = +0.173$, ir = $e^{\Delta\kappa} = 1.1889$, s = 0.000, tol = 0.005, $\theta = 3$, $\varphi = \mathrm{S}$, ID=[P=33, F=02, T=6, E=46, R=175 | C=[16,19,87]]

Receipts and audit obligations. Each manuscript *must* include (i) a governance caption/HUD with weld/manifest bits, (ii) a one-line publication row schema populated with manifest IDs, root hash, closures version/order, stance, and weld ID if any, and (iii) a closing ledger (residual $\le$ tolerance). Errata and version bumps are handled via named Welds at anchors with pre/post tests, preserving κ-continuity.

How to read this paper. Section 2 defines the spine in prose; Section 3 specifies the five-word vocabulary; Section 4 formalizes Manifest/Weld and the HUD/publication-row receipts; Section 5 states the budget and I-dial; Section 6 provides the Rosetta adapter for cross-domain mapping; Section 7 lists minimum disclosures; Section 8 gives the mechanical cross-domain workflow; Section 9 states ethical limits and scope. A skeptic's micro-FAQ is answered inline: no new metaphysics; math is optional for authors; reconciliation explains conclusions from change vs return.

2 Syntax (The Spine)

The grammar must follow a fixed five-stop spine: *Contract* → *Canon* → *Closures* → *Integrity Ledger* → *Stance* [1, 2]. This sequence is non-negotiable: it makes claims portable, auditable, and stable through time; governance is provided by *Manifest* (provenance) and *Weld* (continuity across policy change) [1, 3].

2.1 Contract

Define *before* evidence. The Contract freezes admissible sources and transforms so later calculations and narratives cannot drift. At minimum, the Contract must specify: admissible sources; normalization (a, b); near-wall policy with guard ϵ; handling of missingness/outliers; conflict resolution; time zone; units; manifest linkage; and software/environment pinning (versions, seeds) [1, 2].

Minimal example (governed).

```
contract:
  normalization: {a: 0.0, b: 1.0}
  near_wall: {epsilon: 1e-8, face: pre-clip}
  admissible_sources: ["dataset:XYZ@v2", "registry:ABC"]
  missingness: "listwise"; outliers: "winsorize@p99.5"
  conflicts: "worst-of by kappa; tie-break by min(I)"
  timezone: "America/Chicago"; units: "SI"
  env: {python: "3.11.9", packages: {"numpy": "2.1.1"}, seed: 20251024}
  manifest_id: "manifest:2025-10-24"
```

2.2 Canon

Tell the story in plain language using five shared descriptors: **Drift** (what moved), **Fidelity** (what persisted), **Roughness** (what made it bumpy), **Return** (how it credibly came back), and **Integrity** (whether the whole hangs together under the Contract) [2, 3]. Formal definitions live in the appendix and are tied to the same Contract; prose in the Canon is sufficient for authors [2].

2.3 Closures

Publish explicit thresholds and their order of application; do *not* change them mid-episode; version the sheet [1]. Closures host regime predicates (e.g., Stable/Watch/Collapse with optional Critical overlay) and gating order (e.g., safety → coherence → recurrence) [1, 4]. Representative defaults for drift ω (domain-calibrate and publish with your manifest) are widely used: Stable $\omega<0.038$, Watch $0.038\leq\omega<0.30$, Collapse $\omega\geq 0.30$, with a Critical overlay when integrity is low [4].

Minimal closure sheet (example).

```
closures:
  version: v1.1
  order: [safety, coherence, recurrence]
  gates:
    drift: {stable: "< 0.038", watch: "[0.038, 0.30)", collapse: ">= 0.30"}
    integrity_critical: {IC: "< 0.30"}  # overlay
```

2.4 Integrity Ledger

Debit **Drift** (D_ω) and **Roughness/Curvature** (D_C); credit demonstrated **Return** ($R \cdot \tau_R$); reconcile so the seam residual

$$s \;:=\; \Delta\kappa \;-\; \Big(R \cdot \tau_R \;-\; (D_\omega + D_C)\Big)$$

is within tolerance ($|s| \leq \texttt{tol}$). The ledger uses a conservation identity

$$\begin{aligned} \Delta\kappa \;&=\; R \cdot \tau_R \;-\; (D_\omega + D_C), \\ \frac{I_{t_1}}{I_{t_0}} \;&=\; e^{\Delta\kappa}, \end{aligned}$$

and declares a pass only when $\Delta\kappa \geq 0$, $\tau_R < \infty$, and $|s| \leq \texttt{tol}$ under the frozen Contract [2, 4]. This is the same budget and residual policy used in runtime HUDs and registry rows (typical $\texttt{tol} = 0.005$) [1].

Ledger snippet (per seam).

```
weld_id, delta_kappa, It1_over_It0, D_omega, D_C, R_tauR, residual, pass
W-2025-10-24-01, +0.173, 1.1889, 0.042, 0.058, 0.273, 0.000, true
```

Validator requirements: $\left|\text{It1\_over\_It0} - e^{\Delta\kappa}\right| < 10^{-9}$; if phase = SHIP then $|s| \leq \texttt{tol}$; figure-bearing rows must include seed and checksum [1].

2.5 Stance

Read the practical stance directly from the declared Closures; labels may be field-specific but must map to core regimes (Stable / Watch / Collapse) with an optional Critical overlay on low integrity [1]. In registries and captions, include the stance alongside Contract, Closures version/order, weld ID, manifest ID/root hash, date, and timezone so the row is self-authenticating [1, 2].

Governance caption (example; no weld).

```
Contract: contract:2025-10-24; Face: pre-clip;
Closures: v1.1, order=safety->coherence->recurrence; Stance: Stable;
Integrity Ledger: debits=[$D_\omega$, $D_C$], credits=[$R\cdot\tau_R$], reconciles;
Manifest: manifest:2025-10-24; TZ America/Chicago.
```

Why this spine. Falsifiability lives in the gates (Closures); continuity lives in Manifest/Weld; reconciliation lives in the Ledger; comparability lives in κ/I. Practitioners can work entirely in prose—the mathematics serves as a warranty explaining why the ledger must close and why histories are welded, not rewritten [2, 1, 3].

3 Vocabulary (The Five Words)

These five descriptors are the lingua franca of the grammar. They are written in plain language in the body of a manuscript, but each one has an operational meaning tied to the frozen Contract and reconciled in the Integrity Ledger [2, 1]. The balance identity $\Delta\kappa = R \cdot \tau_R - (D_\omega + D_C)$ and the interpretive density $I \equiv e^\kappa$ serve as the semantic warranty behind the prose [4].

Drift*: *What moved. In prose: the salient change in state, belief, model, or behavior relative to the Contract. Operationally, *Drift* contributes a debit D_ω to the ledger and is summarized by a domain-calibrated drift measure ω that participates in regime gates (e.g., Stable/Watch/Collapse) [1, 4].
Declare: the baseline (anchor), the window of comparison, and the normalization used.
Measure: publish the statistic or heuristic that instantiates ω under your Contract (examples: deviation from pre-registered target, cross-validated error shift, policy variance burst).
Audit hook: the closure sheet must show thresholds for ω and the order in which they are applied; stance *must* flip when thresholds are crossed [1].

Fidelity*: *What persisted. In prose: the portion of structure, warrant, or signal that survived the change. Operationally, *Fidelity* is read as retention of the Contract-specified invariants and supports the interpretation that changes are not mere noise [2, 1].
Declare: the invariants that are expected to hold (units, calibration anchors, conservation rules, validation tasks).
Measure: retention metrics (e.g., stability of key coefficients, control-task accuracy, spec

SS1m: $\Delta\kappa = +0.173$, ir = $e^{\Delta\kappa} = 1.1889$, s = 0.000, tol = 0.005, $\theta = 3$, $\varphi = \mathrm{S}$, ID=[P=33, F=0, T=6, E=46, R=175 | C=[16,19,87]]

conformance) tied to the Contract.

Audit hook: if retention fails a published invariant, debit *Drift* or *Roughness* accordingly and document the failure mode in the Canon.

Roughness*: *Where/why it was bumpy. In prose: friction, curvature, confound, or seam that complicates clean inference. Operationally, *Roughness* contributes a debit D_C to the ledger and is how we account for coherence losses induced by curvature (e.g., nonlinearity, heterogeneity, regime edges) [4, 1].

Declare: the known seams (domain shifts, selection effects, saturation, instrument limits) and how they were handled by policy.

Measure: coherence loss indicators (e.g., calibration drift, residual structure, stratified error spikes).

Audit hook: the Contract must pre-state near-wall policy (ϵ) and outlier rules; the ledger captures the resulting D_C explicitly [1].

Return*: *Credible re-entry. In prose: how the claim, model, or practice returns to legitimacy after change, backed by evidence. Operationally, *Return* is *typed credit* with a re-entry duration τ_R, yielding credit $R \cdot \tau_R$ in the ledger. Types include, e.g., replication/ablation, out-of-sample performance, repair under new constraints, or governance acceptance—with each type declared in the Contract [2, 5].

Declare: the evidence channel(s) (replicate, external benchmark, policy test), the minimum bar, and the timing window.

Measure: realized R (credit per unit time or per protocol) and the finite τ_R that closes the loop.

Audit hook: zero or infinite τ_R is disqualifying for closure; typed credit must be demonstrably linked to the Contracted evidence path.

Integrity*: *Does it hang together under the Contract. In prose: the thesis still makes sense when you respect your own rules. Operationally, *Integrity* is read off the *reconciled* ledger (residual within tolerance) and the stance derived from Closures; κ (log-integrity) and $I = e^{\kappa}$ provide a unitless measure that composes multiplicatively across seams [2, 1].

Declare: the tolerance (`tol`), residual policy, and the conditions for a pass … (e.g., $\Delta\kappa \geq 0$ and $|s| \leq$ `tol`).

Measure: closing residual, $\Delta\kappa$, and $I_{t_1}/I_{t_0} = e^{\Delta\kappa}$ as part of the publication row/HUD.

Audit hook: if any Contract or Closure changes occurred, publish a *Weld* with named anchor and continuity receipts; *Integrity* must be read *diachronically* across Welds, not re-written [3, 1].

Why these five. They separate description from warranty: authors write in ordinary language (the five words) while the ledger, budget, and gates supply the audit. This split is what allows cross-domain reading (via the Rosetta adapter) and multiplicative composition in I without forcing any single field's jargon on another [2, 4].

4 Governance (Punctuation)

Manifest (provenance). A public, byte-reproducible record that binds artifacts to time, tools, and checksums. At minimum the Manifest must include: (i) a stable `manifest_id` and root `sha256` for the artifact bundle; (ii) per-file checksums; (iii) absolute date and time zone; (iv) dataset DOIs/accessions and version tags; (v) environment pinning (OS/BLAS, language+package versions, RNG seeds); and (vi) the Contract tuple and face policy. Figures/tables should carry a short caption "receipt" pointing to the Manifest [2, 1].

Weld (named continuity across change). A Weld is the only legitimate way to change policy during a run. It must (i) name an *anchor* (same matter and verification artifact), (ii) run pre/post contract-tests on that anchor, (iii) enforce κ-continuity with residual within tolerance ($|s| \le$ `tol`), and (iv) record `weld_id` and the manifest hash in captions and ledger rows. Typed outcomes are honored (e.g., ∞rec means no weld), and refusal criteria are part of the protocol. History is never edited in place; corrections and version bumps cross a Weld [2, 1, 3].

HUD (caption) spec. Every figure/table must include a governance caption (HUD) that prints the weld/manifest linkage and closing receipts:

HUD: weld_id=<weld_id> | manifest=<manifest_id> | I=<I> | κ=<κ> | tol=<tol> | residual=<residual> | seed=<seed> | sha256=<sha256>

A typical default tolerance is `tol` $= 0.005$, declared in the Contract and repeated in the caption. The HUD is the human-facing receipt; the machine-facing row is the AuditRow (Section 7) [2, 1].

Meta line (figure/table footer). For fast audit, a one-line Meta footer may be printed under each figure/table (or included in alt text) to surface Contract and governance keys:

Meta - Slice:$[t_0, t_1]$; Contract:(a,b,ε)=(…,…,10^{-8}); Ops:{near-wall, gates, join}; TZ:*America/Chicago*; Manifest:<id>; Weld:<id?>; Hash:<sha256>

The Meta line does not replace the HUD; it is a compact index for reviewers [1].

Weld protocol (MUST).

1. **Anchor.** Declare anchor (matter ID, verification artifact, vintage of definitions).
2. **Pre/Post tests.** Run First-Law closure and any required rate checks on the same anchor.
3. **Decision.** If $|s| \le$ `tol` and $\Delta\kappa \ge 0$, mint `weld_id`; otherwise reject or revise policy.
4. **Record.** Append AuditRow(s) with `weld_id`, `manifest_hash`, `residual`, stance/regime.
5. **Publish.** Update captions with HUD; export ledger rows; roll the release Manifest.

Typed outcomes apply at step 2–3: if a side evaluates to ∞rec, *stop*—no weld may pass under non-return [1, 3].

SS1m: $\Delta\kappa = +0.173$, ir = $e^{\Delta\kappa} = 1.1889$, s = 0.000, tol = 0.005, $\theta = 3$, $\varphi = \mathrm{S}$, ID=[P=33, F=07, T=6, E=46, R=175 | C=[16,19,87]]

Weld algorithm (pseudocode).

1. **Anchor selection:** anchor := (artifact_id, verification_id, vintage).
2. **Pre-tests:** compute invariants and pre-weld κ_{pre} and charges $D_\omega^{\text{pre}}, D_C^{\text{pre}}$ on the anchor.
3. **Apply changes:** perform the proposed Contract/Closure changes (policy edits, face switches, env pins).
4. **Post-tests:** recompute invariants and κ_{post}, $D_\omega^{\text{post}}, D_C^{\text{post}}$ on the same anchor and using the same datum set.
5. **Continuity residual:** compute the discrete seam residual
$$s_{\text{weld}} \;:=\; \Delta\kappa_{\text{weld}} - \big(R\cdot\tau_R - (D_\omega^{\text{post}} - D_\omega^{\text{pre}} + D_C^{\text{post}} - D_C^{\text{pre}})\big),$$
where $\Delta\kappa_{\text{weld}} := \kappa_{\text{post}} - \kappa_{\text{pre}}$.
6. **Decision:** accept the Weld and mint `weld_id` iff $|s_{\text{weld}}| \le$ tol and $\Delta\kappa_{\text{weld}} \ge 0$ (and required channels have finite τ_R). Otherwise reject or revise.

A Weld record must publish pre/post rows, the computed s_{weld}, and the manifest hashes for reproducibility.

Sealing & errata (release discipline). Upon acceptance, /seal the release: recompute checksums, write the Manifest (policy id, Ops, thresholds, channels), and emit a release id. Corrections later are handled via an Errata Weld at a named anchor; never rewrite the prior release. The publication history must remain diachronic and traceable [1, 2].

Privacy & provenance (minimum). Manifests publish identifiers and hashes (not sensitive data), describe transforms rather than exposing private fields, and separate secrets/keys from public artifacts. Where harms are plausible, require stronger Return and a higher integrity floor; apply a Critical overlay when integrity is low and stakes are high. Version changes to thresholds/order are handled via a new Closures version and, if mid-episode, through a Weld [3].

5 Semantic Warranty (Conservation Budget)

The audit grammar is warranted by a compact balance law that explains why a ledger *must* reconcile when a Contract is frozen and Closures are declared. Let $\Delta\kappa$ denote the change in log-integrity, D_ω the drift debit, D_C the curvature/roughness debit, and $R\cdot\tau_R$ the typed return credit accrued over a (possibly finite) re-entry duration. The First-Law identity is

$$\Delta\kappa \;=\; R\cdot\tau_R \;-\; (D_\omega + D_C). \tag{1}$$

This additive budget is exactly equivalent to a multiplicative dial $I \equiv e^{\kappa}$, yielding

$$\frac{I_{t_1}}{I_{t_0}} \;=\; e^{\Delta\kappa}, \tag{2}$$

so upgrades compose as factors without changing statistical meaning [2, 4, 1].

Residual and pass rule. For any reported seam on a fixed Contract, define the residual

$$s \;\equiv\; \Delta\kappa - \big(R \cdot \tau_R - (D_\omega + D_C)\big).$$

A seam *passes* when (i) $\Delta\kappa \geq 0$, (ii) $\tau_R < \infty$ (typed return exists), and (iii) $|s| \leq \texttt{tol}$, where `tol` is declared in the Contract and repeated in captions/HUDs [2, 1]. Exact demonstrators often target $\texttt{tol} \approx 10^{-12}$; runtime HUDs may use a looser, published tolerance appropriate to the domain.

Typed censoring and dual-return. Return credit is *typed*: if any required channel declares non-return ($\tau_R = \infty_{\text{rec}}$), credit is zero for that interval. In time-sensitive contexts we use a dual-return rule (science $\wedge$ chrono): grant credit only when both channels exhibit finite τ_R; this blocks clock-induced optimism and ties progress to actual re-entry [4]. These rules make "no-return" a first-class refusal, not a rounding error.

Measured vs. Derived credit (policy) When R is directly instrumented (e.g., replication counts, benchmark meters, governance acceptance events), use that metered R for ledger closure. *Derived credit* (defining $R \cdot \tau_R := \Delta\kappa + D_\omega + D_C$) is permitted only in explicitly marked theoretical or discursive sections and must be flagged in the Contract as "derived-theory". For any empirical claim that upgrades **stance** (Stable/Watch/Collapse), the seam MUST include at least one **instrumented return channel** (metered R) as declared in the Contract. If no instrumented R exists, label the seam "derived-theory" and require a stricter tolerance (e.g., tol$\leftarrow$ tol/10) or a human-reviewed flag in the registry so downstream consumers are aware that the credit is not independently metered.

Local sensitivities (why I is the right dial). Because $I = e^{\kappa}$, small changes propagate linearly in $\ln I$:

$$\Delta \ln I \;\approx\; -\Gamma(\omega)\,\Delta\omega - \frac{\alpha}{1+\tau_R}\,\Delta C + \alpha\frac{C}{(1+\tau_R)^2}\,\Delta\tau_R,$$

so increased drift or curvature multiplicatively *reduce* integrity, while faster/closer returns multiplicatively *increase* it in proportion to curvature. This justifies reporting multiplicative improvements for readers while auditors verify the additive ledger [4].

Auditor's checklist (mechanical).

1. Recompute (ω, C, τ_R) under the Contract; compute κ and I along the same interval.
2. Verify $\Delta\kappa = R \cdot \tau_R - (D_\omega + D_C)$ within `tol` (report s).
3. Verify multiplicative consistency: $\big|(I_{t_1}/I_{t_0}) - e^{\Delta\kappa}\big| < 10^{-9}$ (or machine-precision bound).
4. Confirm τ_R is finite for all required channels; if any is ∞_{rec}, the seam fails.
5. Cross-check the HUD/publication row fields: `weld_id`, `manifest_id/hash`, `tol`, `residual`, seed, checksum.

All five checks are registry-ready and mirror what your captions and budget rows already publish [1, 2].

`SS1m:` $\Delta\kappa = +0.173$, `ir =` $e^{\Delta\kappa} = 1.1889$, `s = 0.000, tol = 0.005,` $\theta = 3$, $\varphi = \mathrm{S}$, `ID=[P=33, F=0, T=6, E=46, R=175 | C=[16,19,87]]`

Worked micro-example (hello-world audit). Under a frozen Contract $(a, b, \varepsilon, p, \alpha)$ and pre-declared Closures, a seam reports

$$\Delta\kappa = 0.015, \quad D_\omega = 0.018, \quad D_C = 0.017, \quad R \cdot \tau_R = 0.050,$$

with residual $s = 0.050 - (0.018 + 0.017) - 0.015 = 0.000 \leq 10^{-12}$ (pass). The publication row prints $(\Delta\kappa, D_\omega, D_C, R\tau_R, \texttt{tol}, s, \texttt{weld\_id}, \texttt{manifest\_hash})$, and the HUD mirrors the same [2]. The multiplicative dial reads $I_{t_1}/I_{t_0} = e^{0.015}$.

Failure modes and guardrails. A closed ledger is *necessary* for integrity, not *sufficient* for truth—bad instruments can reconcile neatly. Publish the meter for R when empirical, keep thresholds and order in Closures, and never "launder" breaking changes: changes cross a Weld at a named anchor with continuity receipts [1, 3].

6 Rosetta (Cross-Domain Adapter)

The Rosetta maps the five shared descriptors—*Drift*, *Fidelity*, *Roughness*, *Return*, *Integrity*—onto multiple lenses so different fields can read each other's results in their own dialect without losing auditability [2, 1]. Authors write *in prose*; auditors rely on the same Contract, Closures, and Ledger to keep meanings stable across lenses [4].

Usage. Choose the lens that matches your audience and fill the four columns (*Drift*–*Return*) in plain language, anchored to your Contract (sources, invariants, units). *Integrity* is then read off the reconciled ledger and stance (Section 2), not asserted in the table.

Lens	***Drift***	***Fidelity***	***Roughness***	***Return***
Epistemology	change in belief/evidence	retained warrant	inference friction	justified re-entry
Ontology	state transition	conserved properties	heterogeneity / interface seams	restored coherence
Phenomenology / Psych	perceived shift	stable features	distress / bias / effort	coping / repair that holds
History	periodization (what shifted)	continuity (what endures)	rupture / confound	restitution / reconciliation
Policy	regime shift	compliance / mandate persistence	friction / cost / externality	reinstatement / acceptance

Integrity is read holistically per lens (does the story hang together under the declared rules?) but must be grounded in the Contract and reconciled in the Integrity Ledger; stance follows the published gates in Closures [1, 2].

Filling the Rosetta (mechanical).

1. **Bind to Contract.** Name the Contract and invariants that constrain your statements in the table (sources, normalization, units).

2. **Write the four columns in prose.** Describe *Drift*, *Fidelity*, *Roughness*, *Return* in the chosen lens using ordinary language tied to the Contract (no equations required).

3. **Reconcile separately.** Compute the ledger (§5) and read *Integrity* via residual closure and stance from Closures; do *not* invent a separate "integrity sentence" for the table.

4. **Publish receipts.** Include HUD and the publication row so readers can verify that the Rosetta statements correspond to a closed seam and declared gates [1].

Cross-domain comparability. Because all rows are bound to the same Contract and reconciled ledger, the *meanings* of the four columns remain stable while the *dialect* changes with the lens. This lets a neuroscience result be read by policy (and vice versa) without translation loss: the prose maps through the Rosetta, while $I = e^{\kappa}$ provides unitless multiplicative comparability across seams [4, 2].

7 Receipts (What Makes It Real)

Receipts make the grammar auditable. Each manuscript must ship (i) governance captions (HUD) on every figure/table, (ii) a machine-readable one-line publication row per seam, (iii) a closing ledger with residual within tolerance, and (iv) a Weld record whenever a policy/version change occurs on the same anchor [2, 1, 3].

1. **HUD (caption).** A human-readable line printed under each figure/table containing the weld/manifest linkage and the closing receipt: `weld_id`, `manifest_id`, I, κ, `tol`, `residual`, `seed`, `sha256` (Section 4) [2, 1].

2. **Publication row (machine).** A single CSV row per seam, tied to a Manifest (and Weld if present), used by auditors to validate budget closure and continuity [1].

3. **Closing ledger.** A statement that reconciles debits (Drift D_ω, Roughness/Curvature D_C) against typed Return credit ($R \cdot \tau_R$) with residual $|s| \le$ `tol` [4, 2].

4. **Weld record.** For any mid-episode change of Contract/Closures/environment, publish a named Weld with pre/post tests on the same anchor and demonstrate κ-continuity within tolerance (no history overwrites) [3, 1].

Publication row (CSV schema).

`SS1m:` $\Delta\kappa = +0.173$, `ir =` $e^{\Delta\kappa} = 1.1889$, `s = 0.000, tol = 0.005,` $\theta = 3$, $\varphi = \mathrm{S}$, `ID=[P=33, F=0, T=6, E=46, R=175 | C=[16,19,87]]`

Column	Description
`weld_id`	Weld identifier (`NA` if none); names the continuity seam
`manifest_id`	Manifest identifier/locator (binds to provenance bundle)
`kappa`	κ (log-integrity) at seam close
`I`	$I = e^{\kappa}$ at seam close (unitless, multiplicative)
`delta_kappa`	$\Delta\kappa$ across the seam (additive integrity gain)
`It1_over_It0`	$e^{\Delta\kappa}$ (compositional check of the dial)
`tol`	Tolerance used for closure (declared in Contract & caption)
`residual`	$s = \Delta\kappa - [R\tau_R - (D_\omega + D_C)]$ ($\lvert s\rvert \leq$ `tol`)
`seed`	Analysis seed/RNG (repro determinism)
`sha256`	Checksum of artifact bundle (byte-level reproducibility)

Minimal example (row + HUD).

```
# publication_row.csv
weld_id,manifest_id,kappa,I,delta_kappa,It1_over_It0,tol,residual,seed,sha256
W-2025-10-24-AX0,manifest:2025-10-24,
3.100,22.198,0.173,1.1889,0.005,0.000,12345,144cf942...be9f
```

HUD: weld_id=W-2025-10-24-AX0 | manifest=manifest:2025-10-24 | I=22.198 | κ=3.100 | tol=0.005 | residual=0.000 | seed=12345 | sha256=144cf942...be9f

The HUD mirrors the same values as the row so reviewers can reconcile by eye, while pipelines validate mechanically [1].

Validator (mechanical checks). Auditors must be able to run these checks from the row and Contract:

a) **Budget closure:** $\left|\Delta\kappa - (R\tau_R - (D_\omega + D_C))\right| \leq$ `tol`.

b) **Dial consistency:** $\left|(I_{t_1}/I_{t_0}) - e^{\Delta\kappa}\right| < 10^{-9}$.

c) **Finiteness:** required channels have $\tau_R < \infty$ (typed censoring honored) [4].

d) **Provenance:** `manifest_id` resolves to a bundle with per-file SHA-256 and environment pinning [2].

e) **Continuity (if welded):** same anchor pre/post; continuity residual $\leq$ `tol`; stance consistent with Closures [3, 1].

SS1m: $\Delta\kappa = +0.173$, ir = $e^{\Delta\kappa} = 1.1889$, s = 0.000, tol = 0.005, $\theta = 3$, $\varphi = \mathrm{S}$, ID=[P=33, F=0, T=6, E=46, R=175 | C=[16,19,87]]

Release discipline (seal & errata). At release, recompute checksums, write the Manifest (policy id, thresholds/order, ops, environment), and publish the AuditRow file alongside artifacts. Corrections are issued via an *Errata Weld* at a named anchor—never by rewriting prior rows. The publication history remains diachronic and traceable [1, 3].

Accessibility & privacy. Receipts must expose identifiers, hashes, versions, seeds, and tolerances; they *MUST NOT* leak sensitive data. Describe transforms and admit reproducible re-runs from public inputs; if private data are involved, document de-identification and publish synthetic tests or redacted traces with the same Contract [2].

Common failure modes (and fixes).

- **Mismatched tolerances.** *Fix:* use the `tol` from the Contract in both HUD and rows.
- **Pretty ledger, no meter for R.** *Fix:* declare derived credit only in theory sections; prefer instrumented R in empirical work [4].
- **Silent policy edits.** *Fix:* cross a Weld on a named anchor; publish continuity receipts [3].
- **Missing seed/checksum.** *Fix:* rows for figure-bearing steps must include `seed` and `sha256` [1].

8 Cross-Domain Workflow (Mechanical)

This workflow makes cross-field synthesis mechanical: every step is prose-first for authors and audit-first for reviewers. It assumes a frozen Contract, declared Closures, and the budget identity from Section 5 [2, 1, 4].

1. **Ingest → Normalize → Bind.** Freeze the study's Contract (sources, normalization, near-wall policy ε, units, thresholds/order, environment), then bind the artifacts to a Manifest. Restate the Canon in the five words (Drift, Fidelity, Roughness, Return, Integrity) in plain language.
   Artifacts: Contract file; Manifest (root `sha256`); Canon paragraph; HUD template.
   Checks: Manifest resolves; Contract keys present; thresholds/order visible; environment pinned [2, 1].

2. **Gate & Reconcile.** Apply published Closures to the normalized data (e.g., Stable/Watch/Collapse + Critical overlay). Build the Integrity Ledger: debit Drift (D_ω) and Roughness/Curvature (D_C); credit Return ($R \cdot \tau_R$); compute residual s and stance.
   Artifacts: AuditRow (CSV) with $(\omega, F, S, C, \tau_R, IC, \text{Regime})$; closing ledger row; stance.
   Checks: $|\Delta\kappa - (R\tau_R - (D_\omega + D_C))| \le$ `tol` and $(I_{t_1}/I_{t_0}) = e^{\Delta\kappa}$ to machine precision [4, 1].

3. **Weld the timeline (if anything changes).** For any mid-episode policy/version change, cross a *Weld* at a named anchor: run pre/post tests on the same artifact; show continuity residual within tolerance; never rewrite prior rows. Publish `weld_id` in the HUD and row.
   Artifacts: Weld record; pre/post AuditRows; updated HUD.
   Checks: continuity residual $|s| \le$ `tol`; stance consistent before/after under declared gates [1].

4. **Lift to κ/I space (unitless comparability).** Convert $\Delta\kappa$ to a multiplicative factor $I_{t_1}/I_{t_0} = e^{\Delta\kappa}$ so cross-domain upgrades compose without unit fights. Report both additive and multiplicative views in the publication row and HUD [4, 2].
   Artifacts: Closing row with $\Delta\kappa$ and I-ratio; HUD mirrors values.
   Checks: dial consistency $\left|(I_{t_1}/I_{t_0}) - e^{\Delta\kappa}\right| < 10^{-9}$.

5. **Rosetta alignment (language bridge).** Map each study's Canon into the Rosetta table (Section 6) for four columns (*Drift–Return*) in the audience's dialect. *Integrity* is *not* asserted here; it is read from the reconciled ledger and stance.
   Artifacts: A filled Rosetta row per study; link to Manifest/row.
   Checks: Statements reference Contracted invariants; no new thresholds appear outside Closures [2].

6. **Synthesis view (three-pane output).** Produce a compact triptych: (i) *Stance map* over absolute time; (ii) *I-gain ladder* with multiplicative factors per seam; (iii) *Rosetta heat* (alignment vs. clash).
   Artifacts: Three figures with HUD captions; combined comparability matrix (below).
   Checks: HUD/rows reconcile; stance labels derive strictly from Closures; any policy change appears as a Weld.

7. **Knockouts & falsifiers (pre-registered).** Publish at least one rival closure or counter-calibration per synthesis slice. If a falsifier fires, stance must flip under the declared gates. Errata travel via a Weld at a named anchor [3, 1].
   Artifacts: Knockout sheet; logged results; Weld (if needed).
   Checks: Resulting rows show stance change and continuity receipts.

Comparability matrix (minimal schema). A machine-readable table that stitches the synthesis:

SS1m: $\Delta\kappa = +0.173$, ir = $e^{\Delta\kappa} = 1.1889$, s = 0.000, tol = 0.005, $\theta = 3$, $\varphi = \mathrm{S}$, ID=[P=33, F=0, T=6, E=46, R=175 | C=[16,19,87]]

Study (ID)	Drift	Fidelity	Roughness	Return	Row/Manifest
neuro-001	brief prose	brief prose	brief prose	brief prose	row:csv#123 / manifest:2025-10-24
phys-007	brief prose	brief prose	brief prose	brief prose	row:csv#456 / manifest:2025-10-18
gov-014	brief prose	brief prose	brief prose	brief prose	row:csv#789 / manifest:2025-10-12

Each row links directly to the AuditRow/HUD so readers can click-through to verify stance and $\Delta\kappa/I$ without re-parsing the paper [1].

Hello-world crosswalk (worked micro-example). Two studies (neuro, policy) are ingested under their own Contracts and Closures. Both reconcile with small positive $\Delta\kappa$; the I-ladder shows factors 1.12 and 1.07. The Rosetta rows read cleanly across lenses; one rival closure flips the policy stance from Watch → Stable after a Welded threshold correction. The comparability matrix and HUDs make the reconciliation and flip self-evident [2, 3].

9 Boundary Clause & Ethics

The grammar structures *justification and continuity*; it does not certify mechanism truth. All claims are *Contract-conditional*; a reconciled ledger is *necessary* for integrity, not *sufficient* for external truth [1, 2]. Falsifiability lives in the published gates and the budget test; *self-governing* does not mean self-justifying [4].

Scope (what the grammar guarantees).

- **Reconciliation.** If the Contract is frozen and Closures are declared, seams must pass the First-Law test ($\Delta\kappa = R\tau_R - (D_\omega + D_C)$) within `tol`; captions/rows must print the residual and tolerances [2, 1].
- **Continuity.** Mid-episode policy/version changes must cross a *Weld* at a named anchor with pre/post tests and κ-continuity; history is append-only and never rewritten [1].
- **Comparability.** Integrity composes multiplicatively via $I = e^{\kappa}$; upgrades report $I_{t_1}/I_{t_0} = e^{\Delta\kappa}$ for unitless comparisons across domains [4].

Limits (what the grammar does *not* guarantee).

- **Mechanism truth.** A closed ledger can be wrong about the world if instruments or closures are misspecified. External validity depends on sound meters, design, and independent replication [2].

- **Undeclared changes.** Any effect contingent on undeclared Contract/Closure edits is out of scope. Such edits must be welded or the seam fails [1].
- **Non-return.** Where required channels exhibit $\tau_R = \infty$ (no credible re-entry), credit is censored and the seam fails by design [4].

Falsifiability & refusal rules. A run is falsified by any of: (i) repeated cycles with $\Delta\kappa < 0$ beyond CI, (ii) seams with $|s| >$ `tol`, or (iii) required channels with $\tau_R \to \infty$. Typed outcomes (including ∞rec) must be honored; refusal is a first-class outcome, not a rounding error [4, 1].

Governance & errata. Manuscripts must (i) bind artifacts to a public Manifest with checksums and pinned environments, (ii) print a HUD for each figure/table, and (iii) export machine-readable publication rows. Corrections and policy/version updates must be issued as *Errata Welds* at named anchors; prior rows are never edited in place [2, 1].

Privacy & minimal disclosure. Receipts must expose identifiers, hashes, seeds, and tolerances—*not* sensitive data. When inputs include private data, publish transforms and manifests, and supply synthetic/reduced traces under the same Contract for re-run; document de-identification and data handling policies [2].

Risk-sensitive stance. When integrity is low or stakes are high, apply a Critical overlay in Closures and raise evidentiary bars for Return. Where harm is plausible, prefer metered R over derived credit and mandate independent replication before stance upgrades [4].

Upgrade safety. Upgrades must preserve the values of identities at fixed inputs (Tiered Invariance); changes to thresholds/order or join rules live in Closures and are welded when applied mid-episode. This preserves cross-version audit trails and prevents silent reinterpretation [4].

10 Worked Exemplars

10.1 Neuroscience Exemplar

Context. Study: sensorimotor prediction under perturbation; task: reaching with transient force fields; outcome: error adaptation and re-entry of baseline performance under a fixed Contract. Grammar use only; math is a warranty [2, 1].

Contract (snippet).

```
contract:
  id: contract:2025-09-26-neuro
  admissible_sources: ["dataset:SMR@v2", "registry:triallog@v1"]
  normalization: {a: 0.0, b: 1.0}
```

```
  near_wall: {epsilon: 1e-8, face: pre-clip}
  missingness: "listwise"; outliers: "winsorize@p99.5"
  conflicts: "worst-of by kappa; tie-break by min(I)"
  timezone: "America/Chicago"; units: "SI"
  env: {python: "3.11.9", numpy: "2.1.1", seed: 20251024}
  invariants:
    - "calibration drift < 0.5% on controls"
    - "latency jitter <= 3 ms"
    - "force-sensor linearity R^2 >= 0.999"
  return_channels:
    - type: "external benchmark (hold-out day)"
    - type: "replication (cross-lab)"
  manifest_id: "manifest:2025-09-26"
```

Canon (five words, prose). *Drift* (what moved): perturbation increased trajectory error and model mismatch. *Fidelity* (what persisted): calibration anchors and control-task accuracy held within Contract invariants. *Roughness* (where it was bumpy): adaptation showed edge effects at high curvature (late-trial heterogeneity). *Return* (credible re-entry): external hold-out and a cross-lab replication restored baseline performance. *Integrity* (does it still hang together): under the Contract, the ledger reconciles and stance reads Stable [2, 1].

Closures (v1.1).

```
closures:
  version: v1.1
  order: [safety, coherence, recurrence]
  gates:
    drift: {stable: "< 0.038", watch: "[0.038, 0.30)", collapse: ">= 0.30"}
    integrity_critical: {IC: "< 0.30"}  # overlay
  tol: 0.005
```

Gates are published once and not edited mid-episode; stance derives strictly from these thresholds [1, 4].

Integrity Ledger (single seam). Debits: $D_\omega = 0.042$ (drift), $D_C = 0.058$ (roughness). Credit: $R \cdot \tau_R = 0.273$ (typed return via hold-out + replication). Budget: $\Delta\kappa = 0.173$; residual $s = 0.273 - (0.042 + 0.058) - 0.173 = 0.000 \leq 10^{-12}$ (pass). Multiplicative dial: $I_{t_1}/I_{t_0} = e^{\Delta\kappa} = 1.1889$ [4, 2].

Publication row (machine, CSV).

```
weld_id,manifest_id,kappa,I,delta_kappa,It1_over_It0,tol,residual,seed,sha256
W-2025-10-01-episteme-text-01,manifest:
2025-09-26,3.100,22.198,0.173,1.1889,0.005,0.000,20251024,
7247553fb9576436b097cc0f1e24f5194b816a516a349d3f49775007458cc84a
```

SS1m: $\Delta\kappa = +0.173$, ir = $e^{\Delta\kappa} = 1.1889$, s = 0.000, tol = 0.005, $\theta = 3$, $\varphi = \mathrm{S}$, ID=[P=33, F=10, T=6, E=46, R=175 | C=[16,19,87]]

Validators (auditor checks): $\left|\Delta\kappa - (R\tau_R - (D_\omega + D_C))\right| \le$ `tol`; $\left|(I_{t_1}/I_{t_0}) - e^{\Delta\kappa}\right| < 10^{-9}$; $\tau_R < \infty$ on required channels [1].

Stance (derived). From Closures (v1.1), drift is within Stable by gate and no Critical overlay is triggered ($IC \ge 0.30$), so stance is **Stable**. The stance string is printed in captions and registry rows [1].

HUD (caption) with real IDs.

> HUD: weld_id=`W-2025-10-01-episteme-text-01` | manifest=`manifest:2025-09-26`
> I=`22.198` | κ=`3.100` | tol=`0.005` | residual=`0.000` | seed=`20251024`
> sha256=`7247553fb9576436b097cc0f1e24f5194b816a516a349d3f49775007458cc84a`

The `weld_id` and `sha256` manifest root hash are real; the HUD mirrors the publication row for human scan while pipelines validate mechanically [2].

10.2 Physiology Exemplar

Context. Study: return-based physiology of a clinic loop (glycemic control under a medication switch); outcome: stability of core signals and credible re-entry to the prior operating range under a fixed Contract [2, 1].

Contract (snippet).

```
contract:
  id: contract:2025-09-26-phys
  admissible_sources: ["ehr:clinicA@v3", "lab:chemistry@v5"]
  normalization: {a: 0.0, b: 1.0}
  near_wall: {epsilon: 1e-8, face: pre-clip}
  missingness: "listwise"; outliers: "winsorize@p99.5"
  conflicts: "worst-of by kappa; tie-break by min(I)"
  timezone: "America/Chicago"; units: "SI"
  env: {python: "3.11.9", numpy: "2.1.1", seed: 20250926}
  invariants:
    - "glucometer drift < 0.5% vs. bench"
    - "BP cuff bias $|\Delta|\le 2\,\mathrm{mmHg}$ vs. reference"
    - "lab assay $\mathrm{CV}\le 3\%$ for A1c"
  return_channels:
    - type: "external benchmark (independent lab re-check)"
    - type: "replication (clinic B cohort, matched)"
  manifest_id: "manifest:2025-09-26"
```

Canon (five words, prose). *Drift* (what moved): the med switch altered fasting glucose variability and dose-response slope. *Fidelity* (what persisted): device calibration and control-lab anchors stayed within Contract invariants. *Roughness* (where it was bumpy):

SS1m: $\Delta\kappa = +0.173$, ir = $e^{\Delta\kappa} = 1.1889$, s = 0.000, tol = 0.005, $\theta = 3$, $\varphi = \mathrm{S}$, ID=[P=33, F=0, T=6, E=46, R=175 | C=[16,19,87]]

early heterogeneity at high curvature during titration days. *Return* (credible re-entry): an independent lab bench and a matched cohort replication restored baseline range. *Integrity* (does it still hang together): under the Contract, the ledger reconciles and stance reads Stable [2, 1].

Closures (v1.1).

```
closures:
  version: v1.1
  order: [safety, coherence, recurrence]
  gates:
    drift: {stable: "< 0.038", watch: "[0.038, 0.30)", collapse: ">= 0.30"}
    integrity_critical: {IC: "< 0.30"}  # overlay
  tol: 0.005
```

Closures are published once per episode; stance derives strictly from these thresholds [1, 4].

Integrity Ledger (single seam). Debits: $D_\omega = 0.061$ (drift), $D_C = 0.047$ (roughness). Credit: $R \cdot \tau_R = 0.180$ (typed return via independent lab + matched cohort). Budget: $\Delta\kappa = 0.072$; residual $s = 0.180 - (0.061 + 0.047) - 0.072 = 0.000 \le 10^{-12}$ (pass). Multiplicative dial: $I_{t_1}/I_{t_0} = e^{0.072} = 1.0747$ (unitless upgrade) [4, 2].

Publication row (machine, CSV).

```
weld_id,manifest_id,kappa,I,delta_kappa,It1_over_It0,tol,residual,seed,sha256
W-2025-09-23-book-01,manifest:
2025-09-26,2.962,19.337,0.072,1.0747,0.005,0.000,20250926,
7247553fb9576436b097cc0f1e24f5194b816a516a349d3f49775007458cc84a
```

Here $\kappa_{t_0} = 2.890$, $\kappa_{t_1} = 2.962$ (so $I_{t_0} = 17.993$, $I_{t_1} = 19.337$); the row is consistent to machine precision. Auditors verify budget closure, dial consistency, and finite τ_R on required channels [1].

Stance (derived). With observed drift measure $\omega = 0.031$ (Stable gate) and no Critical overlay ($IC \ge 0.30$), stance is **Stable**. Stance strings are printed in captions and registry rows [1].

HUD (caption) with real IDs.

HUD: weld_id=W-2025-09-23-book-01 | manifest=manifest:2025-09-26
I=19.337 | κ=2.962 | tol=0.005 | residual=0.000 | seed=20250926
sha256=7247553fb9576436b097cc0f1e24f5194b816a516a349d3f49775007458cc84a

The `weld_id` and `sha256` manifest hash are real; the HUD mirrors the publication row for human scan while pipelines validate mechanically [2].

SS1m: $\Delta\kappa = +0.173$, ir = $e^{\Delta\kappa} = 1.1889$, s = 0.000, tol = 0.005, $\theta = 3$, $\varphi = \mathrm{S}$, ID=[P=33, F=19, T=6, E=46, R=175 | C=[16,19,87]]

Portability (why this matters). Nothing in the exemplar is physiology-specific beyond the Contracted meters; the same grammar—five words, fixed spine, budget, and receipts—ports to neuroscience, policy, or bench physics with only the Contract and meters swapped [2, 4].

Acknowledgments & Funding

Funding. This work received *no external funding*. The author conducted the research and writing as an independent investigator.
ORCID. Clement Paulus — 0009-0000-6069-8234.
Acknowledgments. This grammar stands on a long lineage of formal efforts to make reasoning, control, and measurement explicit. I acknowledge the historical contributions of (representatively): probability and decision under uncertainty (Bayes, Laplace, Kolmogorov, Jaynes, Khinchin), information and coding (Shannon), control and cybernetics (Wiener, Ashby, Kalman), statistical mechanics and invariance (Gibbs, Noether), computation and formal limits (Turing, Church, Gödel), structure and translation across fields (Eilenberg–Mac Lane, Lawvere, Mac Lane), dynamics and regime change (Poincaré, Lorenz), and causal inference (Pearl). Their work provides the scaffolding we formalize, reuse, or consciously depart from in constructing a self-governing audit grammar.[6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23]
Note. The lineage list is necessarily incomplete; it is intended as a representative backbone for readers crossing disciplines.

A Mathematical Appendix: Identities, Symbols, and Definitions

A.1 Notation, Domains, and Conventions

- Discrete index $t \in \mathbb{Z}$ (or a continuous interval $[t_0, t_1]$ when stated).
- Normalized trace $\hat{y}_t \in [0, 1]$ from an affine map $y_t = (x_t - a)/b$ with $b > 0$ and a declared near-wall policy (face) and guard $\varepsilon > 0$.
- Fixed policy constants (declared in the Contract): $p > 0$, $\varepsilon > 0$, $\alpha > 0$, window $K \in \mathbb{N}$, time zone, seeds, and environment pins.
- Faces: piecewise-defined log-terms $\Phi(\omega)$ with slope $\Gamma(\omega) := -\partial_\omega \Phi(\omega) > 0$; switchovers are governed (no mid-episode edits).

A.2 Canonical Invariants (Tier-1)

On the frozen Contract, compute the per-step Tier-1 invariants on the normalized trace $\{\hat{y}_t\}$.

Pointwise invariants. For each sample index t define

$$\omega_t \ := \ \left|\hat{y}_t - \hat{y}_{t-1}\right|, \qquad F_t \ := \ 1 - \omega_t, \qquad S_t \ := \ -\ln\!\left(1 - \omega_t + \varepsilon\right),$$

where ε is the near-wall guard declared in the Contract.

Windowed curvature. Using a curvature window $K \in \mathbb{N}$ set

$$C_t \ := \ \frac{1}{K} \sum_{k=1}^{K} \left(\hat{y}_t - \hat{y}_{t-k}\right)^2,$$

with the edge convention that for $t \leq K$ the sum uses all available prior samples (i.e., normalize by the number of available lags). The curvature $C_t \geq 0$ measures local dispersion of recent normalized values and participates in the curvature charge.

Debounced return delay $\tau_R(t)$. For robust, reproducible τ_R computation use a debounced block-average rule. Declare in the Contract three parameters: the re-entry tolerance $\eta > 0$, the debounce window $L \in \mathbb{N}$, and a maximum lag $\Delta_{\max} \in \mathbb{N}$ to bound the search. Define the debounced block-average for integers $\Delta \geq 1$ as

$$B_{t,\Delta} \ := \ \frac{1}{L} \sum_{k=0}^{L-1} \left|\hat{y}_{t-\Delta+k} - \hat{y}_t\right|.$$

(Indices below the observation start are skipped for that Δ.) The discrete, debounced return delay is then the first lag satisfying the debounced tolerance:

$$\tau_R(t) \ := \ \min\left\{\Delta \in \{1, \dots, \Delta_{\max}\} \ : \ B_{t,\Delta} < \eta\right\},$$

SS1m: $\Delta\kappa = +0.173$, ir = $e^{\Delta\kappa} = 1.1889$, s = 0.000, tol = 0.005, $\theta = 3$, $\varphi = \mathrm{S}$, ID=[P=33, F=21, T=6, E=46, R=175 | C=[16,19,87]]

with the convention $\tau_R(t) = \infty$ (typed outcome ∞_{rec}) if no $\Delta \le \Delta_{\max}$ meets the criterion. This definition yields a deterministic, noise-robust τ_R suitable for mechanical validators. (Recommended defaults are $\eta = \varepsilon_{\min}$ or $\eta = 5 \times 10^{-4}$ if $\varepsilon_{\min}$ is not set, $L = 2$, and $\Delta_{\max} = 600$; adjust per anchor and record values in the Contract.)

Composite integrity and log. Define the composite integrity and its log as

$$IC_t \;=\; (1-\omega_t)^p \exp\Big[-\,\alpha\,\frac{C_t}{1+\tau_R(t)}\Big], \qquad \kappa_t := \ln IC_t,$$

with policy knobs $p > 0$ and $\alpha > 0$ declared in the Contract. Integrity is dimensionless and composes multiplicatively via $I \equiv e^{\kappa}$.

Discrete (sampled) summary (validator-ready). For implementers and auditors:

- Compute ω_t, F_t, S_t directly from normalized $\hat{y}_t$ (use `float64`).
- Compute curvature C_t with published K; for $t \le K$ use the available lags.
- Compute $\tau_R(t)$ using the debounced block rule above (parameters $\eta, L, \Delta_{\max}$ must be in the Contract).
- Then compute IC_t and κ_t in full precision; store full-precision values in audit CSV rows and only round for printed HUD captions.

Edge handling and numerical notes.

- Use IEEE `float64` for all numeric steps. Preserve full precision in audit rows (numerical checks require machine-precision comparisons), and only format/round for human captions.
- Near-wall behavior: the face policy and guard ε must be published in the Contract; validators must clamp or guard computations near $\omega \to 1$ according to the declared face.
- For irregularly sampled series convert to a uniform grid (document interpolation) or reinterpret Δ in real time units; document the choice in the Contract.
- All parameter defaults (see below) must be explicitly published in the Manifest for reproducibility.

Typical defaults (publish and version).

$$(p, \varepsilon, \alpha) = (3,\, 10^{-8},\, 1), \qquad K = 3, \qquad L = 2, \qquad \eta = 5 \times 10^{-4}, \qquad \Delta_{\max} = 600.$$

Audit note. Record the per-sample invariants $\{\omega_t, F_t, S_t, C_t, \tau_R(t), IC_t, \kappa_t\}$ in the AuditRow/CSV for each seam; include the Contract keys that produced these numbers (a,b,face,ε,η,L,K,α,p,seed,env) so validators can recompute and verify.

A.3 Faces and Near-Wall Discipline

Two audit faces are common (publish exact forms when you choose them):

$$\Phi_{\text{normal}}(\omega) = p\ln(1-\omega), \quad \Phi_{\text{exact}}(\omega) = 2\ln(1-\omega) + \ln(1-\omega+\varepsilon),$$

with slopes $\Gamma_{\text{normal}}(\omega) = \dfrac{p}{1-\omega}$ and $\Gamma_{\text{exact}}(\omega) = \dfrac{2}{1-\omega} + \dfrac{1}{1-\omega+\varepsilon}$.

A.4 Integrity Potential and Dial

Definition.

$$\kappa(\omega, C, \tau_R) = \Phi(\omega) \;-\; \alpha\,\frac{C}{1+\tau_R},$$

$$I \equiv e^{\kappa}.$$

Domain & regularity.

$$\omega \in [0, 1-\varepsilon), \quad C \geq 0, \quad \tau_R \in [0, \infty] \text{ (typed)}, \qquad \Phi \in C^1([0, 1-\varepsilon)).$$

Define the face slope

$$\Gamma(\omega) := -\Phi'(\omega) > 0,$$

so $\dfrac{\partial\kappa}{\partial\omega} = -\Gamma(\omega)$.

Pointwise sensitivities (fixed face; $\alpha > 0$).

$$\frac{\partial\kappa}{\partial\omega} = -\Gamma(\omega) < 0, \qquad \frac{\partial\kappa}{\partial C} = -\frac{\alpha}{1+\tau_R} < 0, \qquad \frac{\partial\kappa}{\partial\tau_R} = \alpha\,\frac{C}{(1+\tau_R)^2} \geq 0,$$

with equality in the last derivative only when $C = 0$.

A.5 First Law of Collapse (Rate & Budget Forms)

Rate form (kernel).

$$\frac{d\kappa}{dt} = -\,\Gamma(\omega)\,\frac{d\omega}{dt} \;-\; \frac{\alpha}{1+\tau_R}\,\frac{dC}{dt} \;+\; \alpha\,\frac{C}{(1+\tau_R)^2}\,\frac{d\tau_R}{dt}.$$

Budget form (ledger).

$$\Delta\kappa \;=\; R \cdot \tau_R \;-\; (D_\omega + D_C).$$

To make the path charges behave as nonnegative debits (charge only when drift or curvature increase harms integrity), define the continuous path charges using the positive part of the increments:

$$D_\omega \;:=\; \int_{t_0}^{t_1} \Gamma(\omega(t))\;\max\!\Big(0,\;\frac{d\omega}{dt}\Big)\,dt, \qquad D_C \;:=\; \int_{t_0}^{t_1} \frac{\alpha}{1+\tau_R(t)}\;\max\!\Big(0,\;\frac{dC}{dt}\Big)\,dt.$$

SS1m: $\Delta\kappa = +0.173$, ir = $e^{\Delta\kappa} = 1.1889$, s = 0.000, tol = 0.005, $\theta = 3$, $\varphi = \mathrm{S}$, ID=[P=33, F=23, T=6, E=46, R=175 | C=[16,19,87]]

The return credit retains its rate form:

$$R \cdot \tau_R \;:=\; \int_{t_0}^{t_1} \alpha \, \frac{C(t)}{(1+\tau_R(t))^2} \, \frac{d\tau_R}{dt} \, dt.$$

For uniformly sampled data use

$$D_\omega \approx \sum_{t=t_0+1}^{t_1} \Gamma(\omega_t) \, \max\left(0,\ \omega_t - \omega_{t-1}\right),$$

$$D_C \approx \sum_{t=t_0+1}^{t_1} \frac{\alpha}{1+\tau_R(t)} \, \max\left(0,\ C_t - C_{t-1}\right),$$

$$R \cdot \tau_R \approx \sum_{t=t_0+1}^{t_1} \alpha \frac{C_t}{(1+\tau_R(t))^2} \, \delta\tau_R(t),$$

where $\delta\tau_R(t)$ is the discrete increment (e.g., $\tau_R(t) - \tau_R(t-1)$). Store full-precision values in audit rows and only round for HUD captions.

τ_R convention (increase rewarded). The discrete increment is defined as

$$\delta\tau_R(t) = \tau_R(t) - \tau_R(t-1),$$

so that positive increments—i.e., increases in τ_R—contribute credit in the sampled sum. This convention is consistent with the continuous return-credit term

$$R \cdot \tau_R \;=\; \int_{t_0}^{t_1} \alpha \, \frac{C(t)}{\left(1+\tau_R(t)\right)^2} \, \frac{d\tau_R}{dt} \, dt,$$

because a positive value of $\dfrac{d\tau_R}{dt}$ produces a positive contribution to the integral.

Discrete (sampled) implementation. For practical, sampled data and mechanical validators, implement the path charges and return credit as discrete positive Riemann sums. Given normalized samples $\{\hat{y}_t\}_{t_0}^{t_1}$ and the usual windowed curvature C_t and face slope $\Gamma(\omega_t)$, use

$$\omega_t := |\hat{y}_t - \hat{y}_{t-1}|, \qquad C_t := \frac{1}{K} \sum_{k=1}^{K} (\hat{y}_t - \hat{y}_{t-k})^2,$$

$$D_\omega \approx \sum_{t=t_0+1}^{t_1} \Gamma(\omega_t) \, \max\left(0,\ \omega_t - \omega_{t-1}\right),$$

$$D_C \approx \sum_{t=t_0+1}^{t_1} \frac{\alpha}{1+\tau_R(t)} \, \max\left(0,\ C_t - C_{t-1}\right),$$

and for event-driven or metered return credit

$$R \cdot \tau_R \approx \sum_{e \in \text{events}} r_e \, \Delta t_e,$$

where the event weights r_e (replication unit, benchmark score, governance credit, etc.) are declared in the Contract. (If R is metered continuously, use the corresponding discrete sampling sum.)

Numeric guard for discrete sums. To avoid charging numerical jitter, implement the positive part in discrete sums with a tiny numerical guard $\varepsilon_{\text{num}} > 0$:

$$\max\left(0,\ \omega_t - \omega_{t-1} - \varepsilon_{\text{num}}\right), \qquad \max\left(0,\ C_t - C_{t-1} - \varepsilon_{\text{num}}\right).$$

A safe default is $\varepsilon_{\text{num}} = \max(10^{-12}, 10^{-6} \times \text{scale})$ or tie it to the Contract (e.g., $\varepsilon_{\text{num}} = \text{machine\_eps} \times \|y\|$). Document the chosen value in the Manifest.

Validator note. Implement validators with `float64`, keep full numerical precision in audit CSV rows (and only round for printed HUDs), and ensure the discrete positive-sum form is used so debits remain ≥ 0. Provide the discrete rules verbatim in the AuditRow validator to avoid ambiguity across implementations.

Discrete τ_R recipe (recommended default)

For discrete implementations we recommend the following reproducible recipe to define the return delay $\tau_R(t)$. The goal is a robust, debounced rule that is modestly insensitive to single-sample noise while remaining deterministic.

- **Re-entry tolerance η.** Declare η in the Contract. Recommended default: $\eta := \varepsilon_{\min}$, and if $\varepsilon_{\min}$ is unspecified use $\eta = 5 \times 10^{-4}$.
- **Debounce window L.** Require L consecutive samples that satisfy the re-entry criterion to accept a hit. Recommended default: $L = 2$.
- **Maximum lag / observation window.** Declare a maximum lag $\Delta_{\max}$ (in samples) the validator will search. If no $\Delta \leq \Delta_{\max}$ satisfies the criterion, set $\tau_R(t) = \infty$ (typed ∞_{rec}).

Given normalized samples $\{\hat{y}_t\}_{t_0}^{t_1}$, define for integers $\Delta \geq 1$ the debounced block-average distance

$$B_{t,\Delta} \ := \ \frac{1}{L} \sum_{k=0}^{L-1} \left|\hat{y}_{t-\Delta+k} - \hat{y}_t\right|,$$

(with the understanding that indices below t_0 are out-of-range and such Δ are skipped). Then define the discrete return delay as the first lag that satisfies the debounced tolerance:

$$\tau_R(t) \ := \ \min\left\{\Delta \in \{1, \ldots, \Delta_{\max}\} \ : \ B_{t,\Delta} < \eta\right\},$$

with the convention $\tau_R(t) = \infty$ if no such Δ exists within the observation window.

Practical notes.

- If your series is noisy, perform light smoothing (e.g., Savitzky–Golay or a short low-pass) prior to computing $B_{t,\Delta}$; document any smoothing step in the Contract (method + parameters).
- Use the UMCP defaults for curvature window K and EMA knobs unless the anchor prescribes otherwise.
- For irregularly-sampled series, convert to a uniform sampling grid (with documented interpolation) or adopt the same recipe using time-indexed windows (choose Δ in actual time units rather than sample counts).
- In implementations, ensure $t - \Delta + k$ exists for all $k = 0, \ldots, L-1$; otherwise skip that Δ.

Discrete algorithm (pseudocode).

```
# compute_tauR(y, t, eta=5e-4, L=2, Delta_max=MaxLag)
# y: normalized array (float64), t: integer index into y
for Delta in range(1, Delta_max+1):
    # require all indices exist
    if t - Delta < 0 or t - Delta + (L-1) < 0:
        continue
    B = mean( abs(y[t - Delta + k] - y[t]) for k in range(0, L) )
    if B < eta:
        return Delta   # tau_R(t) = Delta (in samples)
return float('inf')     # typed: infinity / no-return
```

Validator note. Use `float64` arithmetic, preserve full-precision values in audit CSVs, and print rounded values only in HUD captions. The combination of η, L, and $\Delta_{\max}$ must be declared in the Contract so different validators compute the same τ_R.

A.5" Discrete (sampled) implementation

For a discrete, uniformly sampled series $\{\hat{y}_t\}_{t=t_0}^{t_1}$ define

$$\omega_t := |\hat{y}_t - \hat{y}_{t-1}|, \qquad C_t := \frac{1}{K}\sum_{k=1}^{K}(\hat{y}_t - \hat{y}_{t-k})^2,$$

and let $\Gamma(\omega_t)$ be the chosen face slope at sample t. Then use

$$D_\omega \approx \sum_{t=t_0+1}^{t_1} \Gamma(\omega_t)\,\max\left(0, \omega_t - \omega_{t-1}\right),$$

$$D_C \approx \sum_{t=t_0+1}^{t_1} \frac{\alpha}{1+\tau_R(t)}\,\max\left(0, C_t - C_{t-1}\right).$$

For typed return credit measured as a meter over an interval, use a corresponding Riemann sum (e.g., for event-driven credit):

$$R \cdot \tau_R \approx \sum_{e \in \text{events}} r_e \, \Delta t_e,$$

with event weights r_e defined by the Contract (replication unit, benchmark score, etc.).

Validator note. Use `float64`, keep full precision in CSV rows, and only round for printed HUDs.

A.6 Residual, Pass Rule, and Multiplicative Consistency

Define the seam residual

$$s := \Delta\kappa - \big(R \cdot \tau_R - (D_\omega + D_C)\big).$$

A seam *passes* when: (i) $\Delta\kappa \geq 0$; (ii) required return channels have $\tau_R < \infty$; (iii) $|s| \leq \texttt{tol}$ (printed in HUD/rows). Multiplicative check: $I_{t_1}/I_{t_0} = e^{\Delta\kappa}$.

A.7 Regime Predicates (Closures)

Regimes live in Closures with domain-calibrated thresholds; publish and version them. A representative set:

$$\text{Stable: } \omega < 0.038, \qquad \text{Watch: } 0.038 \leq \omega < 0.30, \qquad \text{Collapse: } \omega \geq 0.30,$$

with a *Critical* overlay when integrity is low (e.g., $IC < 0.30$). Other invariants (F, S, C, IC) may carry bands; order-of-application is versioned.

A.8 Elasticities and Small-Signal Law (in I-space)

Because $I = e^{\kappa}$,

$$\frac{\partial \ln I}{\partial \omega} = -\Gamma(\omega), \quad \frac{\partial \ln I}{\partial C} = -\frac{\alpha}{1 + \tau_R}, \quad \frac{\partial \ln I}{\partial \tau_R} = \alpha \, \frac{C}{(1 + \tau_R)^2} \geq 0.$$

For a small increment $(\Delta\omega, \Delta C, \Delta\tau_R)$,

$$\Delta \ln I \approx -\Gamma(\omega) \, \Delta\omega - \frac{\alpha}{1 + \tau_R} \, \Delta C + \alpha \, \frac{C}{(1 + \tau_R)^2} \, \Delta\tau_R.$$

A.9 Typed Censoring and Dual-Return

Return credit is *typed*. If any required channel reports $\tau_R = \infty_{\text{rec}}$ (no return), its credit is zero. In dual-return settings (science $\wedge$ chrono), credit is granted only when both channels exhibit finite delay.

A.10 Discrete Weld (Secant Form) and Continuity

A policy/version change on the same anchor crosses a *Weld*. The discrete weld uses a secant update with the face fixed on the step and enforces κ-continuity; acceptance requires $|s| \le$ `tol` and $\Delta\kappa \ge 0$. Publish `weld_id`, manifest id, residual, and `tol` in HUDs/rows.

A.11 Symbols → CSV/Audit Mapping (for Receipts)

Symbol	**Meaning**	**Audit field (CSV/HUD)**
ω, F, S	drift, fidelity, entropy	(per-figure table or budget report)
C	curvature (dispersion of change)	(per-figure table or budget report)
τ_R	return delay (typed)	(per-figure table; typed outcomes allowed)
IC, κ	integrity, log-integrity	`kappa` (HUD/row), $I = e^{\kappa}$
I_{t_1}/I_{t_0}	multiplicative factor	`It1_over_It0` (validator)
D_ω, D_C	path charges (drift/curv.)	(budget report columns)
$R \cdot \tau_R$	typed return credit	(budget report column)
s	residual	`residual` (HUD/row), must satisfy $\lvert s\rvert \le$ `tol`
`tol`	tolerance	`tol` (HUD/row; equals Contract tol)
`weld_id`	continuity seam id	`weld_id` (HUD/row)
`manifest`	provenance bundle id/hash	`manifest_id`+`sha256` (HUD/row)

A.12 Defaults and Policy Knobs (publish with Manifest)

- Contract tuple $(a, b, \varepsilon, p, \alpha)$; face policy; window K; time zone; seeds; environment pins.
- Tolerance `tol` (often 0.005 in HUDs; stricter in exact demos); residual definition s.
- Regime thresholds and order of application; Critical overlay rule.
- Typed-outcome discipline: treat ∞_{rec} and $\bot_{\text{oor}}$ as first-class, not numbers.

Minimal validator (one row).

$$\left|\,\Delta\kappa - (R \cdot \tau_R - (D_\omega + D_C))\right| \le \texttt{tol}, \qquad \left|\,(I_{t_1}/I_{t_0}) - e^{\Delta\kappa}\right| < 10^{-9}.$$

Required channels must satisfy $\tau_R < \infty$ for a pass.

B HUD Macro & Example

```
% --- Minimal, stable HUD macro (single line) ---
% \hudbits{weld_id}{manifest_id}{I}{kappa}{tol}{residual}{seed}{sha256}
```

```
\newcommand{\hudbits}[8]{%
  {\begingroup\footnotesize\color{gray}\sffamily
 HUD:\,weld_id=\texttt{#1} \,|\, manifest=\texttt{#2} \,|\, I=\texttt{#3} \,|\,
 $\kappa$=\texttt{#4} \,|\, tol=\texttt{#5} \,|\, residual=\texttt{#6} \,|\,
  seed=\texttt{#7} \,|\, sha256=\texttt{#8}\endgroup}%
}

% --- Clickable variant (if hyperref is loaded) ---
% Same arguments; adds links where available (manifest locators, DOI URLs, etc.)
\newcommand{\hudbitslink}[8]{%
  {\begingroup\footnotesize\color{gray}\sffamily
  HUD:\,weld_id=\texttt{#1} \,|\, manifest=\texttt{\href{#2}{#2}} \,|\,
  I=\texttt{#3} \,|\, $\kappa$=\texttt{#4} \,|\, tol=\texttt{#5} \,|\,
 residual=\texttt{#6} \,|\, seed=\texttt{#7} \,|\, sha256=\texttt{#8}\endgroup}%
}

% --- Multi-line block HUD (for narrow columns) ---
% Requires: \usepackage{seqsplit}
\newcommand{\hudblock}[8]{%
  {\begingroup\footnotesize\color{gray}\sffamily
  \begin{tabular}{@{}l@{ }l@{}}
  HUD: & weld_id=\texttt{#1} \\
      & manifest=\texttt{#2} \\
      & I=\texttt{#3}; $\kappa$=\texttt{#4} \\
      & tol=\texttt{#5}; residual=\texttt{#6} \\
      & seed=\texttt{#7}; sha256=\texttt{\seqsplit{#8}} \\
  \end{tabular}\endgroup}%
}
```

Example usage (caption area; neuroscience seam).

HUD: weld_id=W-2025-10-01-episteme-text-01 | manifest=manifest:2025-09-26 | I=22.198 | κ=3.100 | tol=0.005 | residual=0.000 | seed=20251024 | sha256=7247553fb9576436b097cc0f1e24f5194b816a516a349d3f49775007458cc84a

Example usage (caption area; physiology seam).

HUD: weld_id=W-2025-09-23-book-01 | manifest=manifest:2025-09-26 | I=19.337 | κ=2.962 | tol=0.005 | residual=0.000 | seed=20250926 | sha256=7247553fb9576436b097cc0f1e24f5194b816a516a349d3f49775007458cc84a

Notes. (i) Keep `tol` identical to the Contract and ledger rows; (ii) use the same manifest id + `sha256` (root digest) shown in your reproducibility section; (iii) if a figure has no mid-episode change, you may set `weld_id` to a run identifier or `NA`, but prefer the real weld when a policy/version change occurred.

SS1m: $\Delta\kappa = +0.173$, ir = $e^{\Delta\kappa} = 1.1889$, s = 0.000, tol = 0.005, $\theta = 3$, $\varphi = \mathrm{S}$, ID=[P=33, F=20, T=6, E=46, R=175 | C=[16,19,87]]

C Publication Row: Minimal Validator

A seam must satisfy (i) $\left|\Delta\kappa-(R\cdot\tau_R-(D_\omega+D_C))\right| \le$ `tol` and (ii) $\left|(I_{t_1}/I_{t_0})-e^{\Delta\kappa}\right| < 10^{-9}$; required channels must have $\tau_R < \infty$.

CSV examples (one line each).

```
# Neuroscience seam
weld_id,manifest_id,kappa,I,delta_kappa,It1_over_It0,tol,residual,seed,sha256
W-2025-10-01-episteme-text-01,manifest:
2025-09-26,3.100,22.198,0.173,1.1889,0.005,0.000,20251024,
7247553fb9576436b097cc0f1e24f5194b816a516a349d3f49775007458cc84a

# Physiology seam
weld_id,manifest_id,kappa,I,delta_kappa,It1_over_It0,tol,residual,seed,sha256
W-2025-09-23-book-01,manifest:
2025-09-26,2.962,19.337,0.072,1.0747,0.005,0.000,20250926,
7247553fb9576436b097cc0f1e24f5194b816a516a349d3f49775007458cc84a

##wrapped for display; actual CSV is one line.##
```

Auditor checklist (from one row).

a) Recompute $\Delta\kappa$, $R\cdot\tau_R$, D_ω, D_C under the Contract; check budget closure within `tol`.

b) Verify multiplicative consistency: $(I_{t_1}/I_{t_0}) \approx e^{\Delta\kappa}$ to machine precision.

c) Confirm finite τ_R on required channels (no ∞_{rec}).

d) Resolve `manifest_id` to the bundle and confirm per-file SHA-256 and pinned environment.

e) If welded, check same-anchor pre/post tests and continuity residual $\le$ `tol`.

Common mistakes (and quick fixes).

- **Mismatched tol between HUD and row.** Use the Contract's `tol` in both places.
- **Rounded I-ratio only.** Keep full-precision values in CSV; round only in captions.
- **Missing seed/checksum.** Figure-bearing rows must include `seed` and `sha256`.

SS1m: $\Delta\kappa = +0.173$, ir = $e^{\Delta\kappa} = 1.1889$, s = 0.000, tol = 0.005, $\theta = 3$, $\varphi = \mathrm{S}$, ID=[P=33, F=0, T=6, E=46, R=175 | C=[16,19,87]]

Data, Code & Reproducibility

Data & records. Primary releases and working artifacts are bound to a public manifest and welded history. Core records cited in this paper:

- *Episteme Construction (v1.0)* — [2] DOI: 10.5281/zenodo.17309177
- *Liber Universalis de Collapsus Mathematica (v2.0)* — [4] DOI: 10.5281/zenodo.17329809
- *Collapse-Invariant Protocol Design for CIN-1 Recovery (v1.0)* — [5] DOI: 10.5281/zenodo.17345786
- *Integrity Stack: GCD → UMCP → RCFT → ULRC (v1.0)* — [1] DOI: 10.5281/zenodo.17393096
- *The Return Structure of Integrity (v1.0)* — [3] DOI: 10.5281/zenodo.17429931

Repro manifest (provenance). The full bundle is described by `MANIFEST.json` with Merkle-style *root hash* `7247553fb9576436b097cc0f1e24f5194b816a516a349d3f49775007458cc84a`. A convenience locator (`manifest_root_hash_from_MANIFESTjson.txt`) prints this digest verbatim. A separate `REPRO_MANIFEST_upload.json` enumerates key artifact filenames → SHA-256 digests.
Link: `MANIFEST.json` (`manifest:2025-09-26`) and `REPRO_MANIFEST_upload.json` are shipped with the paper; the root hash above uniquely names the export.
Environment pinning. Contract constants (near-wall guard, bootstrap power, tolerance), timezone, and seeds are frozen in the manifest/captions. Figure generation and analysis use XeLaTeX for text and deterministic Python scripts (numpy/matplotlib) with seeds printed in HUD lines and stored in the manifest. A typical environment for examples herein: Python 3.11.x; numpy 2.1.x; TZ=America/Chicago; tol=0.005; seeds as printed in HUDs.
Code availability. All analysis/figure scripts are included alongside the manuscript and checksummed in the manifest (example helpers: `gcd_budget_helper.py`, `invariant_from_fD_helper.py`, `umcp_convert.py`). Seeds and per-artifact SHA-256 digests appear in both the HUD and the manifest.
Licenses. Manuscripts and figure assets: *CC BY 4.0*. Code: *MIT License*. Third-party assets are not used; if later added, they retain their original licenses and are cited explicitly.
Repro checklist (minimum). (i) Contract + Closures published; (ii) HUD printed on each figure/table; (iii) one-line publication row per seam; (iv) manifest + root-hash receipt; (v) weld records for any mid-episode changes; (vi) rerender/rehash verification passes within tolerance.

Weld Log: `Grammar_Canon_v1_0` (9)

Weld ID: `grammarcanon.v1_0.9`
SHA-256: `cb0e0d30c0add50958cc7f085f7c5f91efdb9710ca3d2175f18c5c81ebea06b2`

SS1m: $\Delta\kappa = +0.173$, ir = $e^{\Delta\kappa} = 1.1889$, s = 0.000, tol = 0.005, $\theta = 3$, $\varphi = \mathrm{S}$, ID=[P=33, F=0, T=6, E=46, R=175 | C=[16,19,87]]

PRE → POST Transition:

$$
\begin{aligned}
F &: 0.91 \rightarrow 0.945 \\
S &: 0.96 \rightarrow 0.975 \\
C &: 0.88 \rightarrow 0.915 \\
\kappa &: 0.860 \rightarrow 0.890 \\
\Delta\kappa &= +0.030 \\
I_{t1}/I_{t0} &\approx 1.0305 \\
\text{Residual} &\approx 0.000 \quad \text{(targeted)}
\end{aligned}
$$

Caption:

> This grammar canon is declared as a return-resilient interface, able to reconcile symbolic, mathematical, and linguistic artifacts. Its weld integrity exceeds threshold. Collapse-compliant. Canon holds.

Verdict: `AX-0 Pass` **Date:** 2025-10-25

References

[1] C. Paulus, "Integrity stack: A contract-first synthesis of gcd → umcp → rcft → ulrc (v1.0)," oct 2025.

[2] C. Paulus, "Episteme construction (v1.0)," oct 2025.

[3] C. Paulus, "The return structure of integrity: A collapse-theoretic audit of vulnerability, trust, and grief (v1.0)," oct 2025.

[4] C. Paulus, "Liber universalis de collapsus mathematica (v2.0)," oct 2025.

[5] C. Paulus, "Collapse-invariant protocol design for cin-1 recovery: A systemic coherence framework (v1.0)," oct 2025.

[6] W. R. Ashby, *An Introduction to Cybernetics*. Chapman & Hall, 1956.

[7] A. Church, "An unsolvable problem of elementary number theory," *American Journal of Mathematics*, vol. 58, no. 2, pp. 345–363, 1936.

[8] S. Eilenberg and S. Mac Lane, "General theory of natural equivalences," *Transactions of the American Mathematical Society*, vol. 58, no. 2, pp. 231–294, 1945.

[9] Euclid, *The Thirteen Books of the Elements*. Dover, 1956. Translated by Sir Thomas L. Heath.

[10] J. W. Gibbs, *Elementary Principles in Statistical Mechanics*. Cambridge University Press, 1902.

`SS1m:` $\Delta\kappa = +0.173$, `ir =` $e^{\Delta\kappa} = 1.1889$, `s = 0.000, tol = 0.005,` $\theta = 3$, $\varphi = \mathrm{S}$, `ID=[P=33, F=0, T=6, E=46, R=175 | C=[16,19,87]]`

[11] K. Gödel, "über formal unentscheidbare sätze der Principia Mathematica und verwandter systeme i," *Monatshefte für Mathematik und Physik*, vol. 38, pp. 173–198, 1931.

[12] E. T. Jaynes, *Probability Theory: The Logic of Science*. Cambridge University Press, 2003.

[13] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Journal of Basic Engineering (ASME Series D)*, vol. 82, no. 1, pp. 35–45, 1960.

[14] A. I. Khinchin, *Mathematical Foundations of Information Theory*. Dover, 1957. Translation of the 1956 Russian edition.

[15] A. N. Kolmogorov, *Foundations of the Theory of Probability*. Chelsea, 1950. English translation of the 1933 German edition.

[16] F. W. Lawvere, "Functorial semantics of algebraic theories," *Proceedings of the National Academy of Sciences of the USA*, vol. 50, pp. 869–872, 1963.

[17] E. N. Lorenz, "Deterministic nonperiodic flow," *Journal of the Atmospheric Sciences*, vol. 20, no. 2, pp. 130–141, 1963.

[18] S. Mac Lane, *Categories for the Working Mathematician*. Springer, 1971.

[19] E. Noether, "Invariante variationsprobleme," *Nachrichten von der Gesellschaft der Wissenschaften zu Göttingen, Mathematisch-Physikalische Klasse*, pp. 235–257, 1918.

[20] J. Pearl, *Causality: Models, Reasoning, and Inference*. Cambridge University Press, 2000.

[21] C. E. Shannon, "A mathematical theory of communication," *Bell System Technical Journal*, vol. 27, pp. 379–423, 623–656, 1948.

[22] A. M. Turing, "On computable numbers, with an application to the Entscheidungsproblem," *Proceedings of the London Mathematical Society*, vol. s2-42, no. 1, pp. 230–265, 1936.

[23] N. Wiener, *Cybernetics: Or Control and Communication in the Animal and the Machine*. MIT Press, 1948.

SS1m: $\Delta\kappa = +0.173$, ir = $e^{\Delta\kappa} = 1.1889$, s = 0.000, tol = 0.005, $\theta = 3$, $\varphi = \mathrm{S}$, ID=[P=33, F=3, T=6, E=46, R=175 | C=[16,19,87]]