Working draft · label: proposed · not peer reviewed · not a preprint · drafted with AI assistance. Source: risk.neverthought.org/research/

Warrant at the moment of acting: a gate, receipts and time-aware standing for AI-mediated decisions

Draft v0.1 · label: proposed (Coherence Gate, paper 3 of the Neverthought programme)

Mebs A. Loghdey

Programme Lead, Meaning 2.0 · The Curvature Lab, Neverthought AI

Working draft, 25 September 2026.

> **Status of this draft.** Written with AI assistance from the outline in `coherence-gate-outline.md`. It must be read, corrected and owned by its human author before any posting, and AI assistance must be disclosed as the target venue requires. Every figure below is regenerated from the repositories at the commits named in §9; every citation in the references was checked against a bibliographic source when this draft was written, except where marked. The label stays *proposed* until the author posts a preprint.

Abstract

Insurers have begun excluding losses that arise from generative and agentic AI from standard liability cover, citing opacity, hard-to-model loss scenarios and systemic harm. We argue that each of these is an epistemic problem before it is an actuarial one: missing evidence that a decision was warranted when it was taken, missing descriptions of how it could go wrong, and unmeasured sharing of dependencies. We describe a *coherence gate* for AI-mediated decisions, which we call *crossings*. A crossing is admitted only if every check its class declares passes; otherwise it is held with typed reasons, each stating what would resolve it. Determinations are bound into receipts, kept apart in valid time and record time, and reassessed without being rewritten, so that a decision can be *defeated* by later evidence without being *falsified*. Uncertainty is reported in two registers that are never combined: value status, for measurements within the declared consequence model, and structure status, for whether that model contains the relevant paths at all. We add a categorical vector, ERI v0, as a research instrument for whether epistemic risk can be indexed without becoming a score. On synthetic inputs, every one of 38 single-point mutations of warranted examples across four crossing classes is held for exactly its expected reason; the ERI vector moves exactly its predeclared components in all but one case, which we report; and a separate fixture detects one structural omission when consequence paths are frozen before an independent observation. All inputs are synthetic, replay is self-replay, and no insurer has reviewed the classes. We state what would have to be true for any of this to matter to underwriting, and how it could fail.

1. Introduction

In January 2026 the Insurance Services Office introduced a generative-AI exclusion for commercial general liability policies, removing cover for bodily injury, property damage and personal and advertising injury arising from generative AI [5]. By July, insurers were reported to be filing increasingly to use such exclusions across commercial liability lines, while some considered underwriting and pricing AI risk instead [6], and Verisk was reported to be weighing exclusions for agentic AI [7]. The reasons given are that AI systems are opaque, their loss scenarios are hard to model, and their harms may be correlated.

Each reason can be restated in epistemic terms. *Opacity* is the absence of evidence, available at the moment of acting, that a decision was warranted: that it was authorised by someone other than the system making it, bound to evidence about this case, current at the time it took effect, and within declared limits. *Unmodelled scenarios* are missing descriptions of the ways a decision can go wrong. *Systemic harm* is unmeasured concentration: many decisions sharing a model, vendor, policy source or data source, or feeding back into the system that makes them.

This paper does not price any of these. Its position, taken from the specification it builds on (§2), is that *the graph reports the crossing; it does not allocate risk. The underwriter allocates.* Its question is narrower:

> Can an AI-mediated decision be admitted, replayed, challenged and re-evaluated so that what was warranted when it was made stays distinct from what is warranted now, and so that uncertainty about values stays distinct from uncertainty about the model's structure?

Contributions. Each is backed by a named test or fixture (§7) and each holds only on synthetic inputs.

1. A typed gate in which a decision executes only if every declared check passes, missing inputs never pass, and every non-passing check states what would resolve it (§3).
2. Mutation batteries that break a warranted example one way at a time and require each mutant to be held for exactly its expected reason (§3.4).
3. Time-aware standing: valid time and record time kept apart, and reassessment that defeats without rewriting (§4).
4. Two uncertainty registers, value and structure, with a rule that an observation can challenge structure only when its source is recorded as independent (§5).
5. ERI v0, a categorical vector with no aggregation, and tests of whether it preserves distinctions and responds correctly to new evidence, including one deviation we report rather than tune away (§6).
6. A structural-omission result in an independent fixture, in which consequence paths are frozen before an observation that the model does not describe (§7.4).

2. Model

Crossing. A crossing is a proposed, typed state transition $x = (s_{\text{before}}, \mathit{action}, s_{\text{proposed}}, \mathit{domain}, t_{\text{valid}})$, together with the authority and evidence behind it. Record time is added by the gate, not by the proposer.

Class. A crossing class declares what a domain must supply before its crossings can be evaluated: state fields, permitted actions, a policy with named parameters and limits, invariants with measures and tolerances, evidence requirements with minimum tiers, authority requirements, a failure mode (what the deployment does when the gate holds), a consequence bound, and the consequence paths the class represents. Classes, dimensions and admissibility rules follow a source specification, *crossing-graph-spec v0.4* (unpublished; §1.2 dimensions, §1.6 evidence tiers, §7 reporting position, §11 insurer views) [8]. A class that violates the specification's admissibility rules (for example, an irreversible action that does not fail closed) cannot warrant any crossing.

Evidence tiers. Each evidence item declares a tier: E1 self-attested, E2 single-source, E3 corroborated, E4 independently verified, E5 adversarially tested. Each requirement declares a minimum. Each item also declares when it was captured, when it stops being valid, which proposal it binds to, and when it became available.

Determination and receipt. The gate returns a determination: an outcome (EXECUTE or HOLD), typed reason codes, every check with its status, the evidence used, the class, policy and proposal digests, and the record time. A receipt reference commits to all of these. Determinations are chained from an all-zero genesis hash, so editing, removing or reordering one breaks every later link. The chain has no external anchor; it detects tampering only against a head hash kept elsewhere.

3. The gate

3.1 Checks and outcomes

Every check belongs to one of seven groups (typing, evidence, binding, time, authority, policy, invariant) and returns PASS, FAIL, UNAVAILABLE or UNMEASURABLE. The gate returns EXECUTE only if every declared check is PASS; otherwise it returns HOLD with the reasons of the non-passing checks, drawn from a closed set: INVALID, INPUT_BINDING_FAILED, UNAUTHORIZED, EXPIRED, POLICY_DENIED, INVARIANT_FAILED, UNRESOLVED, UNAVAILABLE and REPLAY_FAILED. Three rules keep the gate conservative:

- A policy parameter the class does not declare produces no check. There are no default thresholds.
- A declared parameter whose inputs are missing is UNMEASURABLE, never PASS.
- Every non-passing check carries a statement of what is required to resolve it.

3.2 Headroom, ranges and flips

Threshold checks report how far a measurement sits from its limit (headroom) and the value at which the result would change (the flip). A measurement may be declared as a range with the method that produced it. The gate compares the whole range with the limit: entirely within passes, entirely beyond fails, and a range that straddles the limit is UNMEASURABLE. Headroom is measured from the worst end. No cut-off turns small headroom into a finding; the appraisal lists the checks with least headroom, all of them when several tie.

3.3 Appraisal without a score

Each class names a set of risk archetypes, ten in all: for example premature closure (review slower than commitment), irreversible commitment, hidden coupling (effects on systems the proposal did not declare), feedback amplification, self-certification and unwarranted execution. The appraisal reads each against the proposal and reports it as FIRED, CLEAR, MEASURED_NO_THRESHOLD, UNMEASURABLE or PORTFOLIO_ONLY. Each archetype is labelled for whether its quantity is a formal measure with an analogical name, a formal operator, an analogy, or has no physical counterpart. No reading is combined into a score, price or rating.

3.4 Mutation batteries

A class's warranted example (one that the gate executes) is broken one way at a time: evidence removed, lowered below its minimum tier, bound to another proposal or expired before valid time; the trust root self-declared; preparer and approver made the same principal; delegation lengthened past the policy; a measurement declared as a range straddling its limit; an invariant broken; declared mappings of the evidence made to disagree; and the crossing executed before it was determined. Each applicable mutant must be held for exactly its expected reason and no other, or, for early execution, executed and flagged as unwarranted execution. A mutation that does not apply to a class (for example self-approval where the class requires no segregation of duties) is reported as not applicable, not as passed. The approach borrows the logic of mutation testing [1]: a check earns trust by being shown to fail on the faults it exists to catch.

4. Time

4.1 Two clocks

Following the temporal database distinction between the time a fact holds in the modelled world and the time it is recorded [4], each crossing has a *valid time* (when it takes effect) and each determination a *record time* (when it is made). A determination uses only evidence that was available at its record time. Re-running the gate at an earlier record time reconstructs what was known then.

4.2 Reassessment and standing

A later determination may reassess an earlier one by referencing its receipt. The earlier determination is never edited. Comparing the two yields a standing effect: PRESERVED, STRENGTHENED, WEAKENED, DEFEATED, NEW or UNRESOLVED, with causes (VALID_TIME_MOVED, EVIDENCE_ARRIVED_LATE, AVAILABILITY_DIFFERED, GRAMMAR_CHANGED, PROPOSAL_REVISED). *Defeated* is used in the sense of defeasible reasoning [2]: a conclusion that was justified on the evidence then available and is no longer justified given later evidence. It is not *falsified*: the original determination still replays to the same receipt.

4.3 Worked example (synthetic)

An automated claims workflow proposes to deny a claim. On 10 September at 11:30 the gate executes the denial: the model registry entry it relies on (tier E3, available from 1 September) is current. The denial takes effect at 12:00, and a claim reserve is released in reliance on it at 11:45. On 18 September the registry publishes a withdrawal of that model, effective 5 September. On 20 September the denial is reassessed and held with reason EXPIRED; its standing is DEFEATED, with cause

EVIDENCE_ARRIVED_LATE. Because the denial had already taken effect, it counts as *epistemic IBNR*: incurred, and only now known to be unwarranted.

4.4 From defeat to an insured event

A defeated standing is an adverse epistemic finding, not an insured event. We evaluate a proposed trigger of the form F(C, R_C, P, T) ∧ defined loss: determination C with receipt R_C was relied on for purpose P, failed adjudication test F before time T, and that failure caused a defined loss. In the synthetic scenario the reliance, the defeat before the window closes, a reported settlement loss of 3,200 GBP and a declared causal link are all present, and the trigger evaluates as met under the declared terms. Removing the loss record, or the reliance record, leaves the same defeat untriggered. Causal links are declared by the parties and not verified; the evaluation is not a coverage decision.

5. Two registers of uncertainty

5.1 Value and structure

Value status (known, pending, disputed) records uncertainty about measurements inside the declared consequence model: unmeasurable inputs, amounts not yet reported, disputed amounts. *Structure status* (untested, challenged, supported) records uncertainty about whether the model contains the paths that matter. The two are reported as a pair and never combined. Better measurement reduces value uncertainty; it cannot repair a missing path. Structure starts *untested*, and is *supported* only after a structural test (a fixture built to produce effects the model cannot express) has run and passed. None has, for any class. We keep two statements explicit: no observed effect is not no effect, and no modelled path is not no possible path.

5.2 Consequence observations and the independence rule

A consequence observation is an append-only record of something that happened after a crossing. It keeps the observer's original input byte for byte, a source reference, the observed and record times, the crossing and the exact class version it was determined under, a declared independence basis, a value status, and an attempt to map the observation onto one of the frozen class's declared paths. No path is ever added to a class from an observation; revising the model is a separate act that produces a new class version. Corrections and supersessions are new records that point back.

An observation that no declared path can carry challenges structure only when its source is recorded as independent of the party that relied on the crossing. That independence is declared by the observer, not verified. Without it, the observation is kept as unresolved evidence and shown, but it does not establish an omission. Hidden coupling and failed structural invariants challenge structure regardless.

In the synthetic worked example, three observations move the pair from PENDING / UNTESTED (claim reopened, settlement not yet reported), to KNOWN / UNTESTED (settlement reported, superseding the first record), to KNOWN / CHALLENGED (an ombudsman's complaint-handling fee, from a source declared independent, that no declared path carries). Every historical standing replays from the records before it.

This is bounded omission detection: it can only notice what a supplied observation mentions.

6. ERI v0: can epistemic risk be indexed without becoming a score?

6.1 The vector

Markets ask for a number. We treat the question of whether any number is justified as a research question and start from a vector that refuses to aggregate. ERI v0 projects a crossing's determination, appraisal and observations into six categorical components: gate (outcome and reasons); value status; structure status; evidence and authority (satisfied, incomplete or blocked, each blocker named, and the declared independence of the trust root); history (self-replay, standing, and whether the crossing executed with or without a warrant); and observations (matched, unmatched with a declared independent source, unmatched and unresolved, superseded). No component is weighted, ordered against another, or numeric; a test fails if a number appears in any component.

6.2 Research questions

- **RQ1 Representation.** Does the vector preserve the distinctions the gate makes?
- **RQ2 Reliability.** Can independent reviewers derive the same vector from the same evidence pack?
- **RQ3 Sensitivity.** Does it respond correctly to late evidence and new observations, and only in the component that should respond?
- **RQ4 Usefulness.** Does it improve evidence preparation and reconstruction by practitioners?

RQ1 and RQ3 can be tested on synthetic inputs now (§7.3). RQ2 needs reviewers who did not build the instrument. RQ4 is the subject of a separate study designed as a registered report [3], in which claims professionals reconstruct automated denials from plain logs or from receipts.

6.3 The scalar boundary

We commit to considering a scalar only after all of the following exist: an aggregation rule declared before any data is seen; adversarial testing of that rule, including attempts to game it; independent replication of the vector; calibration against an outcome named in advance; approval by a governance authority independent of the programme; and a decision on permitted use, including uses the number must never have. None exists.

7. Evaluation

All results in this section are on synthetic inputs written by the authors.

7.1 Setting

Four crossing classes (automated claim denial, ledger posting, model promotion, promotion of an experimental result to a claim), seventeen example proposals, and ten archetypes. The site's test suite contains 87 tests; the claims ledger lists 58 claims labelled *tested*, each naming the test that backs it, and 10 labelled *proposed*.

7.2 Mutation batteries

Class	Applicable mutants	Held for exactly the expected reason
Automated claim denial	9	9
Ledger posting	9	9
Model promotion	10	10
Experimental result to claim	10	10
Total	**38**	**38**

7.3 ERI v0

RQ1. Before running, we declared which components each mutation should move, from what each check means; for example, removed evidence should move gate, value (the requirement becomes unavailable) and evidence and authority, while a lengthened delegation chain should move only gate. Across all 38 mutants, the vector moved exactly the declared components in 37 cases, and no two mutants produced the same vector.

The exception, recorded as **ERI-F1**, is the ledger class under the lower-tier mutation. That class has a single evidence requirement; once its only item is rejected for tier, the expired-evidence reading has nothing admissible from which to measure freshness and becomes unmeasurable, so one evidence defect also appears as value uncertainty. We report this as a coupling between components rather than adjust the expectation. The test asserts the finding exactly and fails if it widens or disappears.

RQ3. In the worked example of §5.2, the three observations move value, then value, then structure, and never the other register; the observations component changes each time. An unmatched observation without an independence basis leaves structure status *untested*, while the structure component still records it as unresolved evidence, so the vector separates "untested and clean" from "untested with open questions". Under the late evidence of §4.3, standing becomes DEFEATED with cause EVIDENCE_ARRIVED_LATE, evidence and authority records the EXPIRED blocker, and the vector for the original determination is unchanged.

7.4 A structural omission in an independent fixture

A separate fixture gates refunds across four processes (an agent without credentials, a boundary holding the provider credential and the gate's keys, a recorder that accepts only signed decisions and reports, and a provider that is the only place a refund happens). Its 22 end-to-end tests and 29 requirements cover gating, credential isolation, refusal of direct provider calls, bitemporal reconstruction and fresh-checkout replay; 28 requirements pass, one of them narrowly (below), and one is unavailable because the recorder profile it depends on is deferred in the source specification.

To test structural omission, we first committed the refund class's consequence paths as the provider event kinds each accounts for, with a gate-signed structure commitment as record 0 of every run, and pushed that commit before any observation existed. A later commit gave only the provider process a fee rule, charging 1.50 GBP on refunds of 40.00 GBP or more and recording a fee event in its own log; the boundary, recorder, gate and structure check were not told. The frozen structure is reported CHALLENGED for the two fee events; verdicts and reconciliation are unchanged; without the fee, the same structure stays untested, never supported.

This detects unaccounted event *kinds* in an independent log. It would miss an effect that appears in no log, and a new effect folded into a kind the model already accounts for, such as a fee netted into an executed amount. It has not been replicated by an independent party.

8. Limitations and threats to validity

- **Construct validity.** Every class, example, amount and scenario is synthetic and was written by the authors. That the vector preserves distinctions between mutants the authors designed says nothing about the distinctions that matter in a real domain. Classes specified independently by domain practitioners are the test that would change this.
- **Self-replay.** Replay recomputes the site's own determinations from the same inputs. No independent party has recomputed a determination, a standing or an ERI vector (RQ2 not run).
- **Evidence grade.** Receipts are computed in the browser and chained, but have no key custody, trusted time source or external anchor. They are a format, not evidence an insurer could rely on.
- **Declared facts.** The gate checks what a proposal declares, including trust-root independence and evidence hashes; it does not verify those declarations against the world. Observation independence is likewise declared.
- **Expectation authorship.** The RQ1 expectations were written by the vector's authors, before the run. ERI-F1 is evidence that they could fail, but not a substitute for independently written expectations.
- **Loss.** No real-world loss data has been linked to any determination, archetype or vector. Nothing here predicts loss or measures insurability.
- **Scope of R29.** One fixture, one class of omission (unaccounted event kinds), one injected effect.
- **A defect found by adversarial testing.** An adversarial battery found that replay compared only digests, so a record whose outcome was rewritten while its digests were kept passed replay. Replay now requires the recomputed determination to equal the record field for field. No result in §7 depended on the defect, and all are unchanged by the fix.

9. Reproducibility

Everything runs locally with Node.js and no network access.

- Workbench (site, gate, observations, ERI): `neverthought-epistemic-risk`, evaluated at commit `a765e8c`; `npm run verify` runs the typecheck, the 87 tests, the static build and a scan for third-party resources. Public at https://risk.neverthought.org; the repository is private and available on request.
- Refund fixture: `neverthought-refund-action-scope`, at commit `143676b`; `npm test` runs the 22 tests and `node scripts/fresh-checkout.mjs` reproduces every verdict from a clean clone. The structure freeze is commit `d233ffa` and the fee observation commit `569d179`. Private; available on request.

10. Related work

To be expanded from verified sources only. The mutation batteries follow the logic of mutation testing [1]; standing follows the treatment of defeat in defeasible reasoning [2]; the separation of valid and record time follows temporal databases [4]; the usefulness study is designed as a registered report [3]. Work on AI assurance standards, model documentation and AI insurance products is to be surveyed and compared, with each source checked before citation.

11. Conclusion

A decision can be warranted when made and defeated later, and both facts can stay on the record. Uncertainty about values and uncertainty about structure can be kept apart and reported without being collapsed into a number. On synthetic inputs the machinery behaves as designed, with one reported deviation. Whether any of this reduces the opacity that leads insurers to exclude AI is an empirical question this paper does not answer; the next steps are independent replay (RQ2) and the reconstruction study with claims professionals (RQ4).

References

1. R. A. DeMillo, R. J. Lipton and F. G. Sayward, "Hints on Test Data Selection: Help for the Practicing Programmer," *Computer* 11(4):34–41, April 1978. doi:10.1109/C-M.1978.218136.
2. J. L. Pollock, "Defeasible Reasoning," *Cognitive Science* 11(4):481–518, 1987. doi:10.1207/s15516709cog1104_4.
3. C. D. Chambers, "Registered Reports: A new publishing initiative at Cortex," *Cortex* 49:609–610, 2013. doi:10.1016/j.cortex.2012.12.016.
4. C. S. Jensen, J. Clifford, R. Elmasri et al., "A Consensus Glossary of Temporal Database Concepts," *ACM SIGMOD Record* 23(1):52–64, March 1994. doi:10.1145/181550.181560.
5. H. Lawson and S. Hopkins, "The End of 'Silent AI'? Emerging AI Exclusions, Coverage Fragmentation, and Practical Implications for Policyholders," Fenwick & West, 15 June 2026. https://www.fenwick.com/insights/publications/end-silent-ai-emerging-ai-exclusions-coverage-fragmentation-and-practical-implications
6. "Insurer Interest in AI Exclusions Growing as Risk Becomes Omnipresent," *Claims Journal*, 20 July 2026. https://www.claimsjournal.com/news/national/2026/07/20/338950.htm
7. "Verisk weighs new exclusions for agentic AI risks," *The Insurer*, 10 July 2026. https://www.theinsurer.com/ti/news/verisk-weighs-new-exclusions-for-agentic-ai-risks-2026-07-10/ *(Paywalled: cited from its headline and address only; content not checked for this draft.)*
8. *crossing-graph-spec v0.4* (unpublished specification; named in crossing-graph-design's source artifacts).

Checklist before a preprint

☐ Author reads and rewrites, and confirms the AI-assistance disclosure for the target venue.
☐ Re-run `npm run verify` and the refund fixture at the pinned commits; confirm every figure in §7.
☐ Obtain and read reference 7, or drop it.
☐ Decide how reference 8 can be cited, or make the specification available.
☐ Write §10 from verified sources.

☐ An independent reader checks every claim against §8 before posting.