

Round 24

Do Two Watchers Count as Two?

Authority Lineage and Common-Origin Failure in Claim Revalidation

A source-authority audit for collaborative AI assurance

15 October 2026

Research paper / ALR-0.1-development / D24-A deterministic diagnostic

One bounded audit and authored conformance diagnostic; no production observer, answer benchmark, or deployment.

Contents

This publication separates frozen observations, current-state claims, external findings, and Encevra hypotheses.

Result boundary	5
Abstract	5
1. Archive continuity and corrected round number	6
2. Research question and novelty rationale	6
2.1 Primary question	6
2.2 Why this is distinct	7
2.3 Falsifiability	7
3. Carry-forward from Round 23	7
3.1 Frozen historical findings	7
3.2 Current claims in scope	7
3.3 Hypotheses left untouched	8
4. Scope, exclusions, and threat model	8
4.1 In scope	8
4.2 Out of scope	8
4.3 Threat model	8
5. Concepts that must not be collapsed	9
6. Literature map and competing positions	9
Position A: more methods can reduce omissions	9
Position B: independent development does not imply independent failure	9
Position C: provenance can expose common origin	10
Position D: deployed configuration requires its own observation	10
Position E: live state and runtime events are distinct authorities	10
Position F: authenticated statements still can be wrong or incomplete	10
Position G: count causes, not cues	10
7. Claim and evidence table	11
8. D24-A authority coverage audit	11
8.1 What was inspected	11
8.2 Authored authority matrix	11
8.3 Scope consequence	12
9. Proposed mechanism: Authority-Lineage Revalidation	12
9.1 Event receipt	12
9.2 Authority ancestry graph	13

9.3 Claim card additions 13

9.4 State machine 13

9.5 Known, unknown, and conflicting events 13

9.6 Trust boundary 14

10. Worked example 14

Repository-only result 14

ALR result with a provider-contract authority 14

11. D24-A diagnostic design 14

11.1 Purpose 14

11.2 Fixtures 15

11.3 Strategies 15

11.4 Metrics 15

12. D24-A results 16

12.1 Selector comparison 16

12.2 What the diagnostic demonstrates 16

12.3 What the diagnostic does not demonstrate 16

13. Findings by epistemic status 16

13.1 Established or directly observed 16

13.2 Mixed or conditional evidence 17

13.3 Open questions 17

13.4 Encevra hypotheses 17

14. Alternative explanations and simpler alternatives 17

14.1 The problem is only imprecise wording 17

14.2 Periodic full expiry is enough 17

14.3 Retest-all after any authority event is safer 17

14.4 Runtime observation alone is enough 18

14.5 One signed manifest can unify all authorities 18

14.6 Human review is the real improvement 18

15. Falsifiable hypotheses 18

16. Proposed experiment E24-B 19

16.1 Objective 19

16.2 Prospective freeze 19

16.3 Event sources 19

16.4 Units and sample 19

16.5 Independent adjudication 19

16.6 Arms 20

16.7 Controlled total budget 20

16.8 Primary outcomes	20
16.9 Harm and regression metrics	20
16.10 Ablations	21
16.11 Analysis	21
16.12 Stop conditions	21
17. Brain-inspired hypothesis, kept testable	22
18. Expected failure modes and safeguards	22
19. Limitations	23
20. Architecture and product decision	23
21. Next-round handoff	23
22. Reproducibility manifest	24
23. Source inspection ledger	25
24. Full sources	25
25. Final claim boundary	26

Result boundary

Round 24 audited the evidence boundary of the Round 23 claim-revalidation gate. The audit found that its authored Claim-to-Artifact Graph and its static-import closure are two analysis methods over one root authority: the version-controlled source repository. They are independently constructed, but they are not authority-independent. For the five carried current claims, this round authored a 19-cell claim-by-authority coverage matrix across repository, deployment configuration, live database, provider contract, and runtime invocation state. The Round 23 evidence covered the five repository cells and did not contain separately bound observers for the other fourteen cells. The resulting 5 of 19 figure is a scope audit of the declared matrix, not an empirical production miss rate.

The deterministic diagnostic D24-A then exercised eight selectors over 23 authored change-event fixtures from the five authority families. Eighteen fixtures were relevant, five were negative controls, and two relevant fixtures were intentionally unclassified. The repository-only union achieved 0.1562 claim-edge recall because the fixture universe deliberately contains many non-repository events. The proposed authority-lineage selector achieved 1.0000 recall and 0.9697 precision over the authored claim edges, with one conservative false edge on an unknown event. Retest-all achieved 1.0000 recall and 0.2783 precision. These results show policy conformance on invented cases. They do not estimate real event prevalence, production precision, production recall, or reliability gains.

No independent reviewer labelled the fixtures. No prospective multi-commit window was run. No language model, provider API, deployment control plane, live database, user traffic, or production write was exercised. No new Enceavra answer benchmark was run. No production event collector, claim gate, attestation service, or release blocker was added. The five Round 21 current negative claims were not upgraded, contradicted, or given broader scope. Historical findings remain frozen, and all answer-level and cost hypotheses remain untested.

Abstract

Round 23 added a static-import view to an authored claim dependency graph and observed one unique catch in a real post-freeze repository interval. Its own handoff identified a deeper risk: both views can inherit the same blind spot because both are derived from the same source tree. A deployment flag, an applied database trigger, a provider-side contract change, or an uninstrumented runtime fallback can alter a claim's truth without producing a source-code diff.

Round 24 asks a distinct question: **How should Enceavra represent the lineage and authority of change evidence so that multiple analyses of one source are not mistaken for independent coverage, and so that changes outside the repository can trigger bounded claim revalidation?** This is a source-discovery question before it is a revalidation-allocation question. Retesting every claim after a known code change cannot detect a change that never enters the gate.

The proposed mechanism is **Authority-Lineage Revalidation (ALR)**. Every change-event receipt names a subject, root authority, observer, issuer, old and new commitments, freshness window, coverage contract, privacy class, and trust status. Every current claim names the authority families required by its validity predicate. Derived analyses preserve their ancestry rather than incrementing an unqualified view count. If a required authority has no current observer, the claim is not silently called current across that authority. It becomes `current_within_observed_authorities` or `unknown_authority_gap`, depending on the public scope. Known typed changes make only implicated claims stale. Unknown but production-relevant events fail closed within the affected authority, not across the entire registry.

The literature supports the problem decomposition but not the Enceavra mechanism. Classic multiversion experiments show that separately produced versions can fail dependently [S05-S06]. W3C PROV, SLSA, in-toto, RATS, and SCITT distinguish derivation, issuers, attesters, verifiers, parameters, and trust boundaries [S08-S11, S16]. NIST configuration guidance states that declared configuration and change

control are insufficient without monitoring of the deployed system [S12]. OpenFeature, OpenTelemetry, and PostgreSQL specify useful event surfaces for configuration, runtime, and database state [S13-S15]. None establishes that ALR improves Enceavra.

D24-A gives a narrow engineering result: ancestry-aware event typing can express the difference between two algorithms over one authority and multiple event issuers, and it can fail closed on unknown events without degenerating to retest-all in the authored fixture set. A prospective, independently adjudicated, fixed-budget experiment is still required. Until then, Enceavra should describe Round 23 as dual-method repository coverage, not two independent sources of change evidence.

1. Archive continuity and corrected round number

The automation brief still named Round 11. The live Lovable archive was inspected before topic selection at commit `957a6b4d3ff298a8d5091e2883c12451bab4f7e9`. It contains completed research through Round 23. The correct sequential number is therefore **Round 24**.

The continuity review included the Research overview, rounds index, research registry, full Round 23 and Round 22 manuscripts, the carried Round 21 claim set, the full Round 10 and Round 9 routes, and the Round 19-20 measurement lineage relevant to event receipts. Round 9 concerns learning reusable procedures from verified experience [S04]. Round 10 concerns evidence exchange and selective revision, and its roadmap assigns Rounds 21-25 to dependency discovery, contradictions, source authority, local repair, and retention [S03]. Neither topic is repeated here.

Rounds 19-23 built a continuous chain:

Round	Carried contribution	Boundary preserved in Round 24
19	Evidence Lifecycle Observation Contract	Observation is not causal use, and omitted events remain possible.
20	Minimal-disclosure receipts	Integrity and issuer identity do not prove truth or completeness.
21	Attempt-bound measurement requirement	The production ABMG remains unimplemented and its outcome hypotheses remain untested.
22	Historical, current, and hypothesis claim classes plus CTAG	Historical findings remain immutable; current claims can become stale; hypotheses cannot self-upgrade.
23	Authored CTAG plus static-import view in shadow mode	One repository interval and authored fixtures do not establish production coverage.

Round 23 explicitly prohibited another convenient one-commit repetition. Its preferred path was a preregistered multi-commit experiment with independent hidden adjudication. That infrastructure and independent review were not available in this run. Its specified fallback was to study common-origin and source-authority failure and define a third event source for configuration, schema, provider-contract, or runtime evidence [S01]. Round 24 takes that fallback.

2. Research question and novelty rationale

2.1 Primary question

When two claim-revalidation views are derived from the same repository, how should Enceavra represent their common origin, and can a typed, authority-diverse event layer detect relevant non-code changes without treating every external event as grounds to expire every claim?

The question is consequential for collaborative AI systems because an agent team's reasoning depends on more than checked-in code. Provider routing, model aliases, tool permissions, feature flags, deployed schemas, database triggers, and actual runtime paths can change which evidence reaches which agent and

whether a verification receipt is complete. A gate that watches only source code can be internally consistent and still be blind to the running system.

2.2 Why this is distinct

Round 24 does not restate evidence exchange, selective revision, procedural learning, or the Round 23 selector comparison.

- Round 10 asked what agents should exchange and how conclusions should be selectively revised.
- Round 9 asked whether verified experience can produce reusable procedures.
- Round 23 asked whether two repository-derived views could select claims after a later code change.
- Round 24 asks whether the gate observes the authorities that can change claim truth at all.

The novelty claim is narrow. Provenance graphs, configuration monitoring, runtime telemetry, attestations, signed statements, feature-provider events, and database event triggers already exist [S08-S16]. ALR is an Enceavra-specific composition that binds those event origins to current-claim validity predicates and refuses to count multiple derivations of one authority as multiple independent authorities.

2.3 Falsifiability

ALR loses support if any of the following occurs in a prospective trial:

- 1 Authority lineage does not reduce stale-claim escape relative to repository-only selection at matched total monitoring and revalidation budget.
- 2 A simpler periodic-expiry or retest-all policy matches safety and useful-current availability at lower total cost.
- 3 Typed authority events add no unique catches beyond repository events.
- 4 Authority gaps and issuer trust cannot be represented without exposing sensitive content.
- 5 Unknown-event fail-closed handling creates more harmful unavailability than the stale claims it prevents.
- 6 Source ancestry cannot be maintained accurately enough to prevent false independence claims.

3. Carry-forward from Round 23

3.1 Frozen historical findings

Round 24 does not rewrite the Round 23 historical result. At its candidate commit, the static-import view added one relevant repository artifact missed by the frozen exact-path list. The hybrid selected all three post-hoc relevant artifacts in that single interval and over-invalidated two of five claims. Those facts remain bound to the Round 23 artifacts and limitations [S01].

3.2 Current claims in scope

The five carried current negative claims are:

Claim ID	Current statement carried into Round 24
C21-CUR-ENVELOPE	The current measurement path does not commit the exact provider request and response envelope.
C21-CUR-TERMINAL	It does not uniformly persist provider terminal markers or finish and stop reasons.
C21-CUR-ATOMIC	Measurement completeness is not committed atomically with terminal run state.
C21-CUR-GATE	Release is not gated on a complete attempt-bound measurement receipt.
C21-CUR-NO-ABMG	No production attempt-bound measurement gate exists.

Round 23 revalidated these statements against inspected repository state at its candidate commit. Round 24 does not claim they are false. It identifies a scope problem: several statements refer to all current production attempts, while the revalidation evidence came from one repository authority and source inspection rather than independently bound observations of every deployed authority.

3.3 Hypotheses left untouched

H21-IMPACT, H21-COST, and H23-1 through H23-8 remain proposed. D24-A does not test whether ABMG improves answers, whether CTAG reduces real stale-claim escape, or whether any mechanism is cost-effective in production. It also does not retroactively count the authored fixtures as independently adjudicated events.

4. Scope, exclusions, and threat model

4.1 In scope

- The lineage of the two Round 23 analysis views.
- Five current claims and the authority families implied by their validity predicates.
- A static inventory of relevant repository surfaces at the current Lovable commit.
- A typed event-receipt schema for repository, deployment configuration, live database, provider contract, and runtime invocation authorities.
- Twenty-three authored conformance fixtures and eight deterministic selector strategies.
- Metrics for common-origin inflation, authority coverage, stale-claim escape, false invalidation, review tasks, and useful-current availability.
- A future prospective experiment with independent hidden adjudication and fixed monitoring plus revalidation budgets.

4.2 Out of scope

- Querying or modifying production deployment settings, feature flags, secrets, provider accounts, live database state, customer content, or user traffic.
- Installing a collector, database trigger, provider webhook, transparency service, or release gate.
- Proving completeness or honesty of any external issuer.
- Measuring model correctness, collaboration quality, latency, cost, safety, security, or user trust.
- Claiming that a source with a different name, vendor, or process is statistically independent.
- Capturing raw prompts, responses, secrets, tokens, or customer records in public receipts.
- Literal brain copying.

4.3 Threat model

Round 24 distinguishes five failure classes.

- 1 **Common-origin omission.** Two analysis methods derive from the same incomplete source and miss the same event.
- 2 **Mirror masquerading as independence.** A deployment snapshot copied from Git is counted as a second authority even though it adds no observation of deployed state.
- 3 **Issuer omission or false statement.** An authority fails to emit an event or emits an inaccurate one. Signed provenance preserves accountability, not truth [S16].
- 4 **Replay or staleness.** A genuine old event is presented as current, or freshness cannot be established.

- 5 **Overbroad fail-closed behavior.** An unknown event invalidates unrelated claims and produces gate fatigue.

ALR is not a complete defense against any of these. It makes them explicit and measurable.

5. Concepts that must not be collapsed

Concept	Definition in Round 24	Common mistake
Analysis view	A method that derives relevance from evidence, such as authored edges or static imports.	Counting methods as independent sources.
Observer	A component that emits or records an event.	Assuming observer identity proves event truth.
Root authority	The system of record whose state can directly change the subject, such as Git, deployment control plane, database, provider, or runtime boundary.	Treating a cached mirror as a new authority.
Issuer	The named entity that makes a statement about an event.	Treating a signature as semantic validation.
Verifier	The process that checks authenticity, freshness, policy, and claim mapping.	Letting the issuer verify itself without recording the trust relation.
Event lineage	The derivation path from authority state through observer and transformation to receipt.	Recording only the final event name.
Authority coverage	Which claim-validity authorities have current, bound observers.	Reporting repository coverage as production coverage.
Statistical independence	A property of joint failure behavior under a distribution.	Inferring it from different algorithms, teams, or vendors.

W3C PROV is directly useful here because it represents derivation, agents, attribution, and primary-source relations [S08]. It does not decide which source is authoritative for Encevera, and it does not prove completeness. That policy remains domain-specific.

6. Literature map and competing positions

Position A: more methods can reduce omissions

Round 23's own repository diagnostic supports a limited version of this position: static imports added one artifact that the exact manifest omitted [S01]. Regression selection and dependence analysis likewise motivate using structurally different analyses rather than one handwritten list. Multiple methods can expose different errors.

Limit: method diversity is not source diversity. If both methods read the same repository, neither can observe a provider-side contract change that leaves Git unchanged.

Position B: independent development does not imply independent failure

Knight and Leveson's multiversion experiment directly tested the assumption that separately developed versions fail independently and found more coincident failures than true independence would predict [S05]. Littlewood and Miller formalized coincident failure and argued that methodological diversity can reduce dependence under conditions that are difficult to verify in practice [S06]. Avizienis and colleagues place design diversity within a larger dependability taxonomy and emphasize that every fault-tolerance mechanism has imperfect coverage [S07].

Consequence: Encevera should not call two views independent merely because one is authored and one is generated. It should preserve ancestry and measure unique catches and correlated misses.

Position C: provenance can expose common origin

W3C PROV represents derivation and primary-source relations [S08]. SLSA distinguishes external parameters, system parameters, resolved dependencies, and the builder that produced an artifact [S09]. in-toto separates the project owner, authorized functionaries, and the metadata each step produces [S10]. These structures support an authority-lineage graph.

Limit: provenance can faithfully reveal that two receipts came from one authority while still saying nothing about whether that authority was correct or complete.

Position D: deployed configuration requires its own observation

NIST SP 800-128 states that planning, secure baselines, and change control are insufficient by themselves; monitoring is needed to identify undocumented components, misconfigurations, vulnerabilities, and unauthorized changes [S12]. OpenFeature defines provider-emitted events for configuration changes, stale state, reconciliation, context changes, and errors [S13].

Consequence: checked-in `.env` templates or feature-flag client code are not evidence of the current remote configuration. The relevant authority is the deployed configuration provider or control plane, with its own event receipt and freshness policy.

Position E: live state and runtime events are distinct authorities

PostgreSQL event triggers can observe database-wide DDL events such as command start/end, object drop, and table rewrite [S15]. OpenTelemetry's log data model represents events from heterogeneous sources and associates them with a Resource that identifies the emitting application or infrastructure [S14]. RFC 9334 separates an Attester that produces Evidence, a Verifier that appraises it, and a Relying Party that makes a decision [S11].

Consequence: a checked-in migration and an observed live schema event are related but non-identical evidence. A runtime receipt can show which path executed, but it does not establish provider-internal truth or event completeness.

Position F: authenticated statements still can be wrong or incomplete

SCITT makes signed issuer statements transparent and auditable, but explicitly notes that issuers can make false statements and can selectively register statements [S16]. RATS likewise requires source binding, timeliness, trust anchors, appraisal policy, and protection of the attesting environment [S11].

Consequence: ALR needs trust and coverage status, not a binary signed equals true rule. Missing, stale, untrusted, and conflicting events must remain explicit.

Position G: count causes, not cues

Kording and colleagues model multisensory integration as inference over whether two cues share a common cause or have separate causes [S17]. The relevant engineering analogy is not that Enceavra copies the brain. It is that two correlated observations should not be double-counted as if they arose from two independent causes.

Limit: a perceptual model does not validate a software assurance architecture. It supplies one ablatable hypothesis: ancestry-aware integration should outperform naive view counting when errors share a root cause.

7. Claim and evidence table

Claim	Status	Evidence	Boundary
Round 23 used two analysis methods over one repository root.	Established for the inspected artifacts	Round 23 manuscript, claim cards, exact paths, and import derivation [S01].	Does not prove the methods always fail together.
The five carried claims depend on more than repository state when scoped to all production attempts.	Enceavra analysis	Claim wording, validity predicates, and authored 19-cell authority matrix.	The matrix was authored in this round and not independently adjudicated.
Separate implementations or views can have dependent failures.	Established in studied multiversion settings	Knight and Leveson; Littlewood and Miller [S05-S06].	No direct Enceavra failure probability follows.
Deployment configuration can drift from declared baselines and should be monitored.	Authoritative guidance	NIST SP 800-128 [S12].	Guidance is not a product-specific experiment.
Event authorities can emit typed configuration, runtime, and DDL events.	Established interface capability	OpenFeature, OpenTelemetry, PostgreSQL [S13-S15].	Capability does not prove deployment, authenticity, or completeness.
D24-A's ALR selector matched 22 of 23 authored event cases and all 32 expected claim edges.	Executed bounded diagnostic	authority_lineage_gate.py , fixture set, and result digest.	Authored conformance only; one extra edge; no independent labels.
Authority-diverse monitoring will reduce real stale-claim escape at acceptable cost.	Enceavra hypothesis	Proposed E24-B.	Untested.
Ancestry-aware cue integration is better than naive view counting for claim revalidation.	Enceavra hypothesis inspired by [S17]	H24-7 and proposed correlated-miss ablation.	Untested and not a neuroscience claim about Enceavra.

8. D24-A authority coverage audit

8.1 What was inspected

The live Lovable project was inspected at commit [957a6b4d3ff298a8d5091e2883c12451bab4f7e9](#). The archive contains the two Round 23 selector definitions and the five carried claims. The repository inventory includes checked-in environment templates, provider and gateway code, Drizzle and Supabase migrations, database schemas, integration configuration, and runtime-oriented modules. File presence was used only to identify boundaries. Secret values and production settings were not inspected.

The Round 23 evidence package records two views:

- 1 an authored CTAG of repository paths and dependency tags;
- 2 a static-import closure derived from the repository snapshot.

Both trace to [source_repository](#). The second method added analytical diversity, not a new system of record.

8.2 Authored authority matrix

Round 24 decomposed each claim's production scope into authorities whose state could change its truth.

Claim	Repository	Deploy config	Live database	Provider contract	Runtime invocation	Required cells
Envelope commitment	yes	yes	yes	yes	yes	5
Terminal persistence	yes	no	yes	yes	yes	4
Atomic completeness	yes	no	yes	no	yes	3
Receipt-gated release	yes	yes	no	no	yes	3
No production ABMG	yes	yes	yes	no	yes	4

Claim	Repository	Deploy config	Live database	Provider contract	Runtime invocation	Required cells
Total	5	3	4	2	5	19

The Round 23 views provided evidence for the five repository cells. They did not contain separately bound deployment, live-database, provider-contract, or runtime event feeds. The resulting observed fraction is $5 / 19 = 0.2632$.

This figure must not be read as 26.32 percent production coverage. The denominator is an authored research decomposition, not a sampled universe. Its value is diagnostic: saying two independent views would hide that both cover the same five cells.

8.3 Scope consequence

The strongest defensible status from repository inspection is:

`current_within_observed_authorities`

It means the named revalidation procedure found the claim true for the source snapshot and other explicitly inspected authorities. It does not mean all production authorities were observed. A public claim can still use `current` if its visible scope is narrowed to the observed authority set. If the wording remains universal, an unobserved required authority should produce `unknown_authority_gap`, not silent confidence.

Round 24 did not modify the public registry or current claim statuses. It proposes a representational correction for future testing.

9. Proposed mechanism: Authority-Lineage Revalidation

9.1 Event receipt

An ALR event receipt contains:

- stable event ID and event type;
- subject identifier and subject version;
- root authority ID;
- observer ID and observer version;
- issuer ID, signature or authenticated channel, and trust policy;
- old and new state commitments where available;
- event time, observation time, and freshness deadline;
- lineage parents and transformations;
- coverage contract and known blind spots;
- privacy class and permitted reveal procedure;
- claim tags and unresolved classification state.

The public receipt should contain commitments and typed metadata, not raw prompts, secrets, provider payloads, or customer material. Round 20's minimal-disclosure boundary remains in force [S16 and archive continuity in S01-S02].

9.2 Authority ancestry graph

Each view points to the evidence records it consumes. Each evidence record points through observers and transformations to a root authority. Two views are **common-origin** for a claim when all of their relevant evidence traces to the same root authority and neither includes a separately bound observation of another required authority.

Round 24 defines two descriptive quantities:

```
view_count(c) = number of analysis methods used for claim c
```

```
root_authority_count(c) = number of distinct, trusted root authorities represented in the current evidence for claim c
```

The quantities are not probabilities. A higher authority count does not imply independence or truth. It merely prevents view count from masquerading as source coverage.

9.3 Claim card additions

Each current claim adds:

- required authority families;
- observed authority families;
- authority-specific validity predicates;
- acceptable issuer and verifier policies;
- maximum evidence age;
- event-to-claim trigger tags;
- unknown-event handling;
- authority-gap status;
- scope text generated from actual coverage.

9.4 State machine

The proposed states are:

- 1 `current_within_observed_authorities`
- 2 `stale_needs_revalidation`
- 3 `unknown_authority_gap`
- 4 `contradicted`
- 5 `retired`

A claim returns from stale only after its named procedure succeeds against every authority implicated by the event. An authority gap is not a contradiction. It is a scope condition. Historical findings remain immutable, and hypotheses remain proposed.

9.5 Known, unknown, and conflicting events

- **Known typed event:** map event tags to claim validity predicates and stale only matching claims.
- **Unknown production-relevant event:** fail closed within claims that depend on that authority, then open a classification task.
- **Conflicting authority events:** preserve both, mark the claim unresolved, and run a discriminating check. Do not use majority vote.

- **Stale or unauthenticated event:** record it, but do not use it to restore current status.
- **Missing required observer:** expose an authority gap. Do not synthesize evidence from repository mirrors.

9.6 Trust boundary

ALR borrows role separation from RATS and SCITT without claiming to implement either standard [S11, S16]. The event authority or attester reports state. A verifier checks authenticity, freshness, lineage, and policy. A claim gate acts as the relying party. The gate still needs an explicit policy for which issuers and verifiers it trusts.

10. Worked example

Suppose Encevera's repository is unchanged. The authored CTAG and static-import closure both report no affected artifact. A provider then changes the model alias behind a stable name so that the streaming response uses a new terminal event and a different envelope field.

Repository-only result

- Authored paths: no change event.
- Static imports: no change event.
- View count: two.
- Root authority count: one, the repository.
- Risk: both views agree because neither observed the provider authority.

ALR result with a provider-contract authority

- 1 A provider-contract observer receives or polls a versioned contract statement.
- 2 The receipt names `provider_contract` as root authority, the provider as issuer, the observer version, the old and new commitments, and a freshness deadline.
- 3 The event is tagged `response_envelope` and `provider_contract_terminal`.
- 4 ALR makes `C21-CUR-ENVELOPE` and `C21-CUR-TERMINAL` stale.
- 5 The envelope procedure captures a non-sensitive commitment to the actual provider-bound request and response schema on a controlled test.
- 6 The terminal procedure checks each active stream path and verifies persisted stop semantics.
- 7 The other three claims remain current unless the change or resulting runtime observation reaches their predicates.

If the provider statement is unavailable or untrusted, ALR does not invent the change or call the claim false. It records `unknown_authority_gap` for the universal provider-scoped claim. If a runtime trace later exposes the new terminal event, that runtime authority can trigger revalidation even without a provider statement. The two events should not be counted as statistically independent merely because they come from different systems; their lineage and shared provider cause remain explicit.

11. D24-A diagnostic design

11.1 Purpose

D24-A tests whether the proposed data model and selector rules behave as specified on a declared finite fixture universe. It does not estimate production performance.

11.2 Fixtures

The 23 fixtures cover five authority families:

- four repository events;
- five deployment-configuration events;
- four live-database events;
- five provider-contract events;
- five runtime-invocation events.

Eighteen events affect at least one carried claim. Five are negative controls. Two relevant events are intentionally typed only as `unknown` to test fail-closed behavior. Expected claim sets were authored from the same trigger vocabulary used to design the selector. No independent reviewer was involved.

11.3 Strategies

- 1 Authored exact paths.
- 2 Static-import closure.
- 3 Union of the two repository views.
- 4 Filename keywords over repository events only.
- 5 Authority broadcast: every event states every claim that lists the authority.
- 6 Typed authority events without fail-closed unknown handling.
- 7 ALR typed events with authority-bounded fail-closed unknown handling.
- 8 Retest-all: every event selects all five claims.

11.4 Metrics

- exact fixture matches;
- event-level true positives, false positives, false negatives, precision, and recall;
- claim-edge true positives, false positives, false negatives, precision, and recall;
- selected claim-revalidation tasks;
- per-authority relevant-event coverage.

An event is caught if the selector chooses at least one claim for a relevant event. Claim-edge metrics are stricter because they measure the exact event-to-claim relation.

12. D24-A results

12.1 Selector comparison

Strategy	Exact events	Event P	Event R	Edge TP	Edge FP	Edge FN	Edge P	Edge R	Claim tasks
Authored exact paths	7/23	1.0000	0.1111	4	0	28	1.0000	0.1250	4
Static import closure	8/23	1.0000	0.1667	5	0	27	1.0000	0.1562	5
Repository union	8/23	1.0000	0.1667	5	0	27	1.0000	0.1562	5
Filename keywords, repository only	7/23	0.7500	0.1667	5	3	27	0.6250	0.1562	8
Authority broadcast	1/23	0.7826	1.0000	30	56	2	0.3488	0.9375	86
Typed, no unknown fail-closed	21/23	1.0000	0.8889	28	0	4	1.0000	0.8750	28
ALR, authority-bounded fail-closed	22/23	1.0000	1.0000	32	1	0	0.9697	1.0000	33
Retest-all	0/23	0.7826	1.0000	32	83	0	0.2783	1.0000	115

12.2 What the diagnostic demonstrates

- 1 The repository union catches every relevant repository fixture but none of the fifteen relevant non-repository fixtures. This is expected by construction.
- 2 Adding event authorities without typed claim mapping gives broad recall but poor claim-edge precision and 86 tasks.
- 3 Removing fail-closed handling misses both unknown relevant events and four expected claim edges.
- 4 Authority-bounded fail-closed handling recovers those edges with one extra claim edge rather than expiring the entire registry.
- 5 Retest-all performs 115 claim tasks, compared with 33 for ALR, on this finite set.

The result digest, excluding its own hash field, is:

```
39571a55503003f089518db5511f1ed5c649987e02a52f06e21dc041773ea959
```

12.3 What the diagnostic does not demonstrate

The fixture distribution was designed to expose authority gaps. Therefore, the repository selectors' low recall is not an estimate of their real-world recall. ALR's high precision is also not a production estimate because the same process authored the events, tags, mapping rules, and labels. The exact numbers are regression targets for the published script. They are not evidence that ALR will work on unseen changes.

No independent reviewer agreement can be reported. This is itself a gate failure for E23-B. Round 24 records the missing condition instead of simulating independence.

13. Findings by epistemic status

13.1 Established or directly observed

- The current archive ends at Round 23, so Round 24 is the correct number.
- Round 23's authored and import-derived views trace to one repository authority.
- The current repository contains code and checked-in artifacts related to configuration, database schemas, providers, and runtime paths, but D23-A did not query their external systems of record.

- D24-A produces the reported deterministic output on the published fixtures.
- Primary literature and standards distinguish diversity from independence, derivation from primary source, issuer from verifier, and declared configuration from monitored deployed state [S05-S16].

13.2 Mixed or conditional evidence

- Diverse methods can add unique catches, as Round 23 observed, but classic multiversion work shows that common failures can remain [S01, S05-S06].
- Signed or attested events improve source binding and accountability, but they do not guarantee truth or complete emission [S11, S16].
- Broad authority monitoring protects recall in D24-A but can create substantial false invalidation. Typed mapping and unknown handling matter.

13.3 Open questions

- How often do non-repository events actually change these claims?
- Which providers and control planes can emit sufficiently fresh, authenticated, minimally disclosed events?
- How often do repository, configuration, database, provider, and runtime observers fail together?
- Can independent reviewers label claim impact without seeing selector outputs or sensitive operational details?
- Does the useful-current availability gain justify monitoring, trust, privacy, and review cost?

13.4 Enceвра hypotheses

ALR, authority-bounded unknown handling, authority-gap status, and ancestry-aware cue counting are Enceвра proposals. They remain hypotheses until E24-B or an equivalent prospective study is completed.

14. Alternative explanations and simpler alternatives

14.1 The problem is only imprecise wording

Perhaps the five claims should simply be narrowed to the inspected repository version. This would remove the authority gap without adding monitoring. That is a valid alternative for claims intended only as code-audit findings. It is insufficient if Enceвра wants current statements about all production attempts.

14.2 Periodic full expiry is enough

Every current claim could expire daily or before each publication. This does not require event discovery and may outperform ALR if relevant external changes are rare or event feeds are costly. It also creates predictable revalidation load and may leave a stale window between change and expiry. E24-B must compare it directly.

14.3 Retest-all after any authority event is safer

This maximizes selection recall once an event is observed. It may be appropriate for a five-claim registry. It still cannot solve missing event emission, and it may scale poorly as claims grow. ALR must beat it on useful-current availability or cost without increasing stale-claim escape.

14.4 Runtime observation alone is enough

Observed requests and responses may reveal the actual executed path and make code and control-plane feeds unnecessary. Runtime evidence is powerful but conditional on traffic, sampling, instrumentation, and retention. It may not observe a dormant path before release, and it cannot by itself explain planned provider or schema changes.

14.5 One signed manifest can unify all authorities

A deployment pipeline could emit one signed snapshot of source, configuration, schema, provider IDs, and instrumentation. This is simpler than many feeds. Its weakness is common-origin capture: a single builder can omit or misstate every field. The manifest should be evaluated as one authority unless independently cross-checked.

14.6 Human review is the real improvement

ALR's benefit may come entirely from forcing reviewers to enumerate authorities, not from event machinery. A manual checklist is therefore a required baseline. If it matches outcomes at lower cost, the automated protocol should be simplified.

15. Falsifiable hypotheses

H24-1, authority-gap recall: Across a preregistered prospective window, ALR will leave fewer independently adjudicated affected claims current after non-repository changes than the Round 23 repository union at the same total event-inspection and claim-revalidation budget.

H24-2, unique authority value: Removing each of deployment, live-database, provider-contract, and runtime observers will create at least one unique stale-claim escape in the corresponding held-out authority stratum.

H24-3, typed selectivity: At matched stale-claim escape, typed authority-to-claim mapping will produce fewer false-invalidated claims and fewer reviewer minutes than authority broadcast or retest-all.

H24-4, unknown handling: Authority-bounded fail-closed handling will reduce stale-claim escape relative to treating unknown events as irrelevant, without increasing false-invalidated claim time by more than a preregistered noninferiority margin.

H24-5, scope honesty: Adding `current_within_observed_authorities` and `unknown_authority_gap` will reduce public universal-current claims unsupported by current authority evidence to zero, without reducing accurately scoped publication coverage by more than five percentage points.

H24-6, lineage integrity: Every event used to restore a claim will retain a verifiable chain to its root authority, observer version, issuer, freshness policy, and revalidation procedure; any missing field must block restoration.

H24-7, common-cause awareness: When two views share a root authority, ancestry-aware integration will produce fewer false assurances than naive view counting at equal claim coverage, measured on correlated-miss probes and prospective events.

H24-8, total cost: Monitoring, verification, graph maintenance, privacy review, and revalidation labor for ALR will remain below retest-all at matched stale-claim escape over the full prospective window.

These are not results. D24-A only shows that the hypotheses can be represented and that the selector implementation matches the authored policy closely enough to justify a real study.

16. Proposed experiment E24-B

16.1 Objective

Test whether authority-lineage monitoring improves stale-claim detection, selectivity, and useful-current availability on genuinely future changes under fixed total monitoring and revalidation budgets.

16.2 Prospective freeze

Before observing evaluation events, freeze and hash:

- 1 the five claim cards and their public wording;
- 2 required authority families and validity predicates;
- 3 accepted issuers, verifiers, trust anchors, and freshness windows;
- 4 event schemas, redaction rules, and lineage fields;
- 5 selector rules and priority ordering;
- 6 unknown and conflicting-event policy;
- 7 revalidation procedures;
- 8 budget points, metrics, margins, exclusions, and stop rules;
- 9 reviewer instructions and adjudication protocol;
- 10 the start and end rule for the future window.

16.3 Event sources

Collect only non-blocking shadow receipts from:

- repository commits and build provenance;
- deployment and feature-configuration change events;
- database DDL and schema-state events;
- provider contract, model-identity, or capability statements where available;
- application-observable runtime invocation and measurement events.

If an authority cannot provide an authenticated feed, record the gap. Do not replace it with an assumed event or label the mirror independent.

16.4 Units and sample

The primary unit is an authority event linked to one candidate system interval. A commit can have zero, one, or many non-repository events. The initial design target is at least 40 qualifying relevant-or-control events and at least five independently adjudicated opportunities in each authority family. This is a planning target, not a power calculation. A pilot outside the evaluation window should estimate prevalence, clustering, and reviewer variance before a final sample calculation.

16.5 Independent adjudication

Two reviewers who cannot see selector outputs independently label:

- whether the event is authentic, fresh, and in scope;
- the root authority and lineage parents;
- which claims could change truth;

- which named procedure is sufficient;
- whether the claim is restored, contradicted, retired, or unresolved;
- whether protected evidence was necessary.

Report raw agreement and claim-edge Cohen's kappa where defined. Preserve disagreements before adjudication. If protected details prevent a judgment, label the edge unresolved rather than negative. No reviewer from D24-A authored fixtures should be the sole gold-set authority.

16.6 Arms

Arm	Policy
A0	Fixed periodic expiry.
A1	Round 23 authored exact paths.
A2	Round 23 repository union.
A3	Manual authority checklist with no automated feeds.
A4	Authority broadcast with no typed claim mapping.
A5	Typed authority mapping without unknown fail-closed.
A6	ALR with lineage and authority-bounded fail-closed.
A7	Retest-all after every observed production-relevant event.
A8	Full-information adjudicated oracle, reported only as an upper bound and never given extra operational budget in the primary comparison.

16.7 Controlled total budget

Separate two budgets that can otherwise be hidden:

- **B_observe**: event polling, verification, receipt storage, protected reveals, and analyst minutes;
- **B_revalidate**: artifact inspection, tests, provider calls, model calls if any, reviewer minutes, and wall-clock time.

Each arm receives the same combined cost ceiling at each budget point. Recommended points are 25, 50, 75, and 100 percent of the measured retest-all cost for the event. Every arm must emit an ordered queue before labels are revealed. Deferred tasks remain counted as unresolved; silent budget overruns are failures.

Model and provider versions, source snapshots, caches, tools, and reviewer time limits must be matched where relevant. An arm cannot claim efficiency by omitting event-acquisition cost or using free historical knowledge.

16.8 Primary outcomes

- severity-weighted stale-claim escape;
- affected-claim recall;
- false-invalidated claims per event;
- useful-current availability over time;
- authority-gap duration;
- budget-normalized correct revalidations;
- total end-to-end cost.

16.9 Harm and regression metrics

- universal public claims with incomplete authority coverage;

- time to stale after the authoritative change;
- time to restore or contradict;
- unknown-event rate and resolution time;
- false restoration from stale, unauthenticated, or replayed receipts;
- common-origin inflation, defined as view count minus root-authority count;
- issuer omission and contradictory-event rate;
- privacy violations or raw sensitive content in receipts;
- gate-fatigue indicators, overrides, and acknowledgment delay;
- answer-quality, latency, and safety regressions if any product path is later exercised;
- historical-claim mutation or hypothesis-upgrade incidents.

16.10 Ablations

- remove deployment-configuration observer;
- remove live-database observer;
- remove provider-contract observer;
- remove runtime observer;
- collapse all authorities into repository lineage;
- count mirrored configuration as a separate authority;
- remove signatures or authenticated channels;
- remove freshness checking;
- remove lineage parents;
- remove unknown fail-closed;
- replace typed mapping with authority broadcast;
- replace ancestry-aware integration with naive view counting.

16.11 Analysis

Use paired event comparisons and cluster by system interval and authority. Report raw counts, effect sizes, uncertainty intervals, unresolved cases, and full budget curves. Do not pool repository and non-repository strata without showing each. Report unique catches and correlated misses by authority pair. Predefine how an event that spans multiple authorities is assigned and how issuer omissions enter the denominator.

No effect size or outcome is claimed in this paper.

16.12 Stop conditions

Pause the shadow study if:

- sensitive payloads, secrets, or customer content enter public receipts;
- an unknown production event is silently treated as safe;
- a claim is restored without fresh evidence from every implicated authority;
- a mirror is counted as an independent authority without lineage proof;
- historical findings are rewritten or hypotheses are upgraded;
- reviewer blinding is broken;

- event and revalidation budgets cannot be reconciled;
- a high-severity stale claim is publicly represented as universally current.

17. Brain-inspired hypothesis, kept testable

The relevant neuroscience result is causal inference in multisensory perception: cues may be integrated when they are inferred to share a cause, or treated separately when independent causes are more plausible [S17]. Round 24 uses only the functional design question.

The engineering hypothesis is:

A claim gate should infer cue ancestry before using agreement as evidence of coverage.

The direct ablation is H24-7. Compare naive view counting with authority-lineage grouping on events where multiple receipts share one root and on events with separate authorities. Hold the underlying receipts, claim procedures, and budget constant. Measure false assurance, stale-claim escape, and false invalidation. If lineage grouping adds no benefit, or if a simpler source label performs equally well, the analogy contributes nothing and should be removed.

This does not claim that ALR is brain-like, biologically realistic, conscious, or a path to literal brain replication.

18. Expected failure modes and safeguards

Failure mode	Consequence	Detection	Proposed safeguard
Observer omission	Relevant change never enters gate	Cross-authority discrepancy, audit, or later incident	Explicit coverage contract and authority-gap status
Common-origin double count	Agreement appears stronger than it is	Lineage graph shows shared root	Report root-authority count beside view count
Mirror misclassified as authority	Fake diversity	Same state digest or derivation parent	Preserve derivation and issuer relation
False issuer statement	Wrong claim becomes stale or current	Conflicting authority, controlled probe, audit	Trust policy, independent verification, unresolved state
Stale or replayed receipt	Old state restores claim	Freshness and sequence checks	Nonce, timestamp, version, and supersession policy
Overbroad authority matrix	Too many claims expire	False-invalidated claim time	Narrow predicates and measure review load
Incomplete matrix	Relevant authority not listed	Held-out authority event or incident	Periodic scope audit and unknown-authority escape metric
Unknown-event flood	Gate fatigue and denial of availability	Unknown rate and queue age	Authority-bounded fail-closed, rate controls, rapid classification
Semantic tag error	Wrong claims selected	Independent adjudication and ablation	Version tags and preserve raw event type commitment
Runtime sampling gap	Dormant or rare path stays invisible	Coverage counters and controlled probes	Pre-release synthetic path coverage plus live shadow telemetry
Privacy expansion	Sensitive state leaks	Schema audit and canary fixtures	Minimal receipts, commitments, protected reveal, strict retention
Verifier common cause	One bug validates every issuer	Cross-implementation checks and fault injection	Verifier diversity only where measured; never assume independence
Cost displacement	Monitoring costs more than revalidation saved	Full cost accounting	Compare manual checklist, periodic expiry, and retest-all

19. Limitations

- 1 The 19-cell authority matrix was authored by the same research process that designed ALR.
- 2 D24-A fixtures are invented, not sampled incidents or production changes.
- 3 The selector, trigger vocabulary, and labels share authorship, so conformance is expected.
- 4 No independent reviewer agreement was measured.
- 5 No deployment configuration, live database, provider account, or provider traffic was inspected.
- 6 File presence in the repository does not prove the corresponding feature is deployed or active.
- 7 The five negative claims are unusually broad and may overstate their required authority sets.
- 8 Root-authority identity is not equivalent to statistical independence. Different authorities can share software, people, policies, credentials, or upstream providers.
- 9 Signatures and attestations establish origin and integrity only under their trust assumptions. They do not establish truth or complete emission.
- 10 OpenFeature, OpenTelemetry, PostgreSQL, SLSA, in-toto, RATS, and SCITT supply building blocks, not an integrated Encevera implementation.
- 11 The neuroscience analogy is narrow and does not validate the mechanism.
- 12 No answer-level, collaboration, safety, security, latency, availability, or cost outcome was measured.
- 13 No production architecture change was made.

20. Architecture and product decision

Decision: no production architecture change. Keep ALR as a shadow research proposal.

One documentation correction is justified: describe Round 23 as using **two repository-derived analysis methods** rather than two independent evidence sources. The static-import method still added a useful unique catch. The correction prevents a method-level success from being promoted into an authority-coverage claim.

If implementation is separately authorized, the smallest safe next step is an offline receipt normalizer that accepts redacted exported events from distinct authorities and preserves lineage. It should not poll production, alter requests, write to production databases, change release decisions, or automatically modify public claim status. Its only purpose would be to make E24-B possible.

21. Next-round handoff

Round 25 should not add another authority name and call the graph complete. It should take one of two paths:

- 1 **Preferred:** preregister E24-B, obtain independent hidden adjudication, and begin a fixed-budget prospective window with at least repository, deployment, and runtime event authorities.
- 2 **If event collection is unavailable:** perform an adversarial omission study of the receipt contracts. Test replay, selective non-emission, mirrored sources, stale configuration, conflicting issuers, and verifier common-mode failure before any production integration.

The next round must preserve these stops:

- no universal current claim without declared authority scope;
- no restoration from an unauthenticated, stale, or unappraised event;
- no counting a derived mirror as a new root authority;

- no hidden raw payloads in public receipts;
- no historical rewrite or hypothesis upgrade;
- no effectiveness claim without independent labels and fixed budgets.

Next research question: Under a prospectively frozen multi-authority event protocol, does authority lineage reduce stale-claim escape and false assurance relative to repository-only and retest-all policies at equal total cost?

22. Reproducibility manifest

The companion bundle contains:

- this manuscript in Markdown;
- `authority_inventory.json`;
- `claim_authority_matrix.json`;
- `authority_fixtures.json`;
- `authority_lineage_gate.py`;
- `diagnostic_results.json`;
- `source_ledger.json`;
- `integrity_manifest.json`;
- the publication PDF;
- a README with exact reproduction commands and claim boundaries.

Run:

```
python3 authority_lineage_gate.py
```

The script requires only Python's standard library. It makes no network, model, provider, database, deployment-control-plane, or production-write call.

23. Source inspection ledger

ID	Source and date	What was inspected	What it supports
S01	Encevra Round 23, 2026-10-14	Full paper, artifacts, limitations, and handoff	One-root repository views and source-authority fallback
S02	Encevra Round 22, 2026-10-13	Full CTAG paper and experiment design	Claim lifecycle and change-triggered revalidation
S03	Encevra Round 10, 2026-10	Full route, roadmap, metrics, falsifiers	Rounds 21-25 source-authority agenda
S04	Encevra Round 9, 2026-09-30	Full route and program boundary	Verified experience and procedural learning distinction
S05	Knight and Leveson, 1986	IEEE record and abstract; repository cross-check	Independent development need not yield independent failures
S06	Littlewood and Miller, 1989	NASA record, abstract, metadata	Coincident failure and conditional value of methodology diversity
S07	Avizienis et al., 2004	Full 23-page paper	Configuration faults, diversity, coverage, verifiability
S08	W3C PROV-DM, 2013	Full recommendation sections on derivation and primary source	Authority ancestry and responsible agents
S09	SLSA Build Provenance, inspected 2026-10-15	External parameters, resolved dependencies, builder	Verifiable configuration origins and materials
S10	in-toto specification, inspected 2026-10-15	Threat model, roles, assumptions, separation	Integrity scope and metadata/execution separation
S11	RFC 9334, 2023	Full role architecture and trust sections	Attester, verifier, relying party, freshness, appraisal
S12	NIST SP 800-128, 2011/2019	Full PDF monitoring and baseline sections	Deployed-state monitoring beyond declared configuration
S13	OpenFeature Events, inspected 2026-10-15	Normative provider event requirements	Configuration provider as event authority
S14	OpenTelemetry Logs and SemConv, inspected 2026-10-15	Log source identity and event semantics	Common envelope for heterogeneous source-labelled events
S15	PostgreSQL 18 event triggers, inspected 2026-10-15	DDL event classes and database-wide scope	Live database as distinct event boundary
S16	RFC 9943, 2026-06	Signed statements, receipts, issuer limits	Accountability without truth or completeness guarantee
S17	Kording et al., 2007	Full article and causal-structure model	Testable common-cause cue-integration analogy

The machine-readable ledger records working links, publication dates, inspection depth, support, and non-support boundaries.

24. Full sources

- 1 Encevra Research. **Round 23: Can the Gate Catch Tomorrow's Change? A Prospective Shadow Trial of Claim Revalidation.** 14 October 2026.
<https://id-preview--8483bc59-da36-492e-b8b8-6bf684070241.lovable.app/research-round-23>
- 2 Encevra Research. **Round 22: When Does an Audit Expire? Dependency-Triggered Revalidation for Rapidly Changing Agent Systems.** 13 October 2026.
<https://id-preview--8483bc59-da36-492e-b8b8-6bf684070241.lovable.app/research-round-22>
- 3 Encevra Research. **Round 10: Evidence Exchange, Selective Revision, and a Testable Path Toward Brain-Inspired Intelligence.** October 2026.
<https://id-preview--8483bc59-da36-492e-b8b8-6bf684070241.lovable.app/research-round-10>

- 4 Enceavra Research. **Round 9: A Research Agenda for the Lab**. 30 September 2026.
<https://id-preview--8483bc59-da36-492e-b8b8-6bf684070241.lovable.app/research-round-9>
- 5 Knight, J. C., and Leveson, N. G. **An Experimental Evaluation of the Assumption of Independence in Multiversion Programming**. IEEE Transactions on Software Engineering SE-12(1), 96-109, January 1986.
<https://doi.org/10.1109/TSE.1986.6312924>
- 6 Littlewood, B., and Miller, D. R. **Conceptual Modeling of Coincident Failures in Multiversion Software**. IEEE Transactions on Software Engineering 15, December 1989.
<https://ntrs.nasa.gov/citations/19900036555>
- 7 Avizienis, A., Laprie, J.-C., Randell, B., and Landwehr, C. **Basic Concepts and Taxonomy of Dependable and Secure Computing**. IEEE Transactions on Dependable and Secure Computing 1(1), 11-33, 2004.
<https://www.landwehr.org/2004-aviz-laprie-randell.pdf>
- 8 W3C. **PROV-DM: The PROV Data Model**. W3C Recommendation, 30 April 2013.
<https://www.w3.org/TR/prov-dm/>
- 9 SLSA. **Build Provenance Specification**. Predicate version 1, inspected 15 October 2026.
<https://slsa.dev/spec/v1.2/build-provenance>
- 10 in-toto. **in-toto Specification**. Inspected 15 October 2026.
<https://github.com/in-toto/specification/blob/master/in-toto-spec.md>
- 11 Birkholz, H., Thaler, D., Richardson, M., Smith, N., and Pan, W. **Remote ATtestation procedures Architecture**. RFC 9334, January 2023. <https://www.rfc-editor.org/rfc/rfc9334>
- 12 Johnson, A., Dempsey, K., Ross, R., Gupta, S., and Bailey, D. **Guide for Security-Focused Configuration Management of Information Systems**. NIST SP 800-128, 2011, updated 2019.
<https://doi.org/10.6028/NIST.SP.800-128>
- 13 OpenFeature. **Events Specification**. Inspected 15 October 2026.
<https://openfeature.dev/specification/sections/events/>
- 14 OpenTelemetry. **Logs Data Model and Semantic Conventions**. Versions 1.61.0 and 1.44.0, inspected 15 October 2026. <https://opentelemetry.io/docs/specs/otel/logs/data-model/>
- 15 PostgreSQL Global Development Group. **PostgreSQL 18, Chapter 38: Event Triggers**. Inspected 15 October 2026. <https://www.postgresql.org/docs/current/event-triggers.html>
- 16 Birkholz, H., Delignat-Lavaud, A., Fournet, C., Deshpande, Y., and Lasker, S. **An Architecture for Trustworthy and Transparent Digital Supply Chains**. RFC 9943, June 2026.
<https://www.rfc-editor.org/rfc/rfc9943>
- 17 Kording, K. P., Beierholm, U., Ma, W. J., Quartz, S., Tenenbaum, J. B., and Shams, L. **Causal Inference in Multisensory Perception**. PLOS ONE 2(9):e943, 26 September 2007.
<https://doi.org/10.1371/journal.pone.0000943>

25. Final claim boundary

Round 24 establishes a bounded structural result. The Round 23 authored CTAG and static-import closure are two analysis methods over one repository authority, and the five universal production-scoped claims name validity conditions that can also change through deployment, live-database, provider-contract, and runtime authorities. On 23 authored fixtures, the ALR selector preserved every expected claim edge with one conservative extra edge, while repository-only selectors missed every deliberately non-repository fixture. This is conformance to an authored model, not evidence of production effectiveness.

The actionable conclusion is therefore limited: Enceavra should track the lineage and root authority of change evidence, scope current claims to observed authorities, and refuse to equate method diversity with source independence. Only a prospectively frozen, independently adjudicated, multi-authority, fixed-budget

experiment can determine whether ALR reduces stale-claim escape, preserves useful availability, and costs less than simpler alternatives.