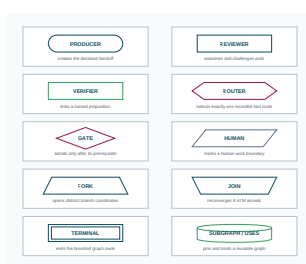


FRONT PAGE · INSPECT THE CONTRACT

Read every edge.

A graph is useful when a reader can see its inputs, carriers, authority, refusal paths, and recorded behavior—not merely admire a successful route.

FOURTH ROUTE THROUGH LETTUCE THEORY · SPECIMEN EDITION



One visual grammar

Producer, gate, reviewer, verifier, fork, join, loop, human decision, carrier, failure, and terminal retain the same meaning on every card.

Success is incomplete

Every non-trivial specimen includes a refusal and a replay path. If the failure cannot be followed backward, the drawing is decoration.

The atlas test

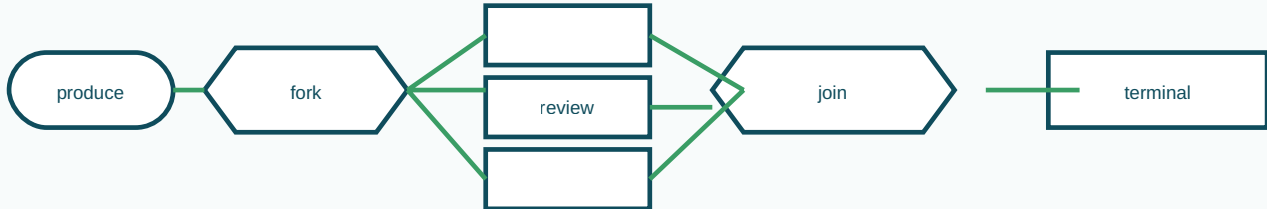
Find the first unsupported leap, trace its evidence, and explain which remaining claim is semantic judgment outside the graph.

Topology names a route. Contract and runtime behavior decide whether the route can be trusted.

The eight-panel contract

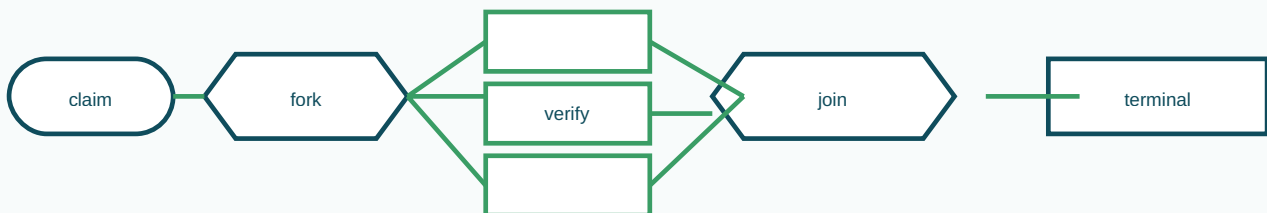
review-panel

v1: 2 of 3 / v2: 4 of 5



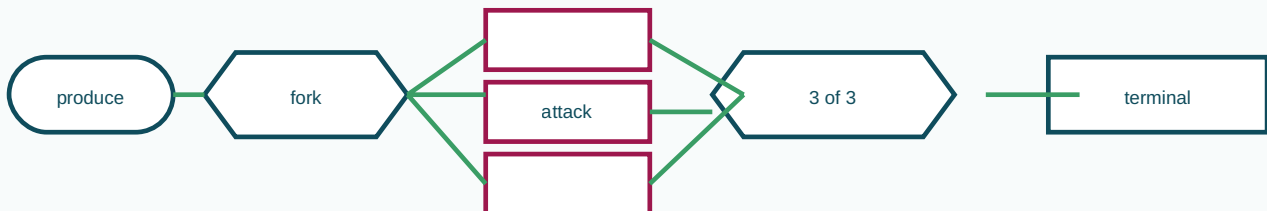
claim-verification

v1: 2 of 3 / v2: 3 of 3



red-team@v1

unanimous veto



Shape + contract

Entry, nodes, edges, exits, loop bounds, and terminal states; then required inputs, outputs, carriers, gates, effect permissions, and refusal conditions.

Behavior + authority

Normal, retry, and failure paths describe what actually runs. Mechanical checks, agent proposals, and human decisions remain visibly distinct.

Evidence + composition

Receipt, subject revision, method, freshness trigger, and regression behavior sit beside safe predecessors, successors, and forbidden connections.

BOUNDARY STRIP

Proves: the declared structure and recorded enforcement shown on the card. **Does not prove:** semantic adequacy, complete dimensions, or authority not present in the receipt.

Producer atom → mechanical gate

gate@v1

minimal

**tracer-bullet@v1**

minimal

**question-based-verification@v1**

low



A1 · Producer atom

Contract: declared subject and method enter; one bounded attempt emits a carrier or typed failure. **Refuses:** missing subject, missing carrier, or unbounded effect. **Composes:** gates, reviewers, joins.

Proves: an attributed attempt emitted this receipt. **Does not prove:** semantic adequacy.

A2 · Mechanical gate

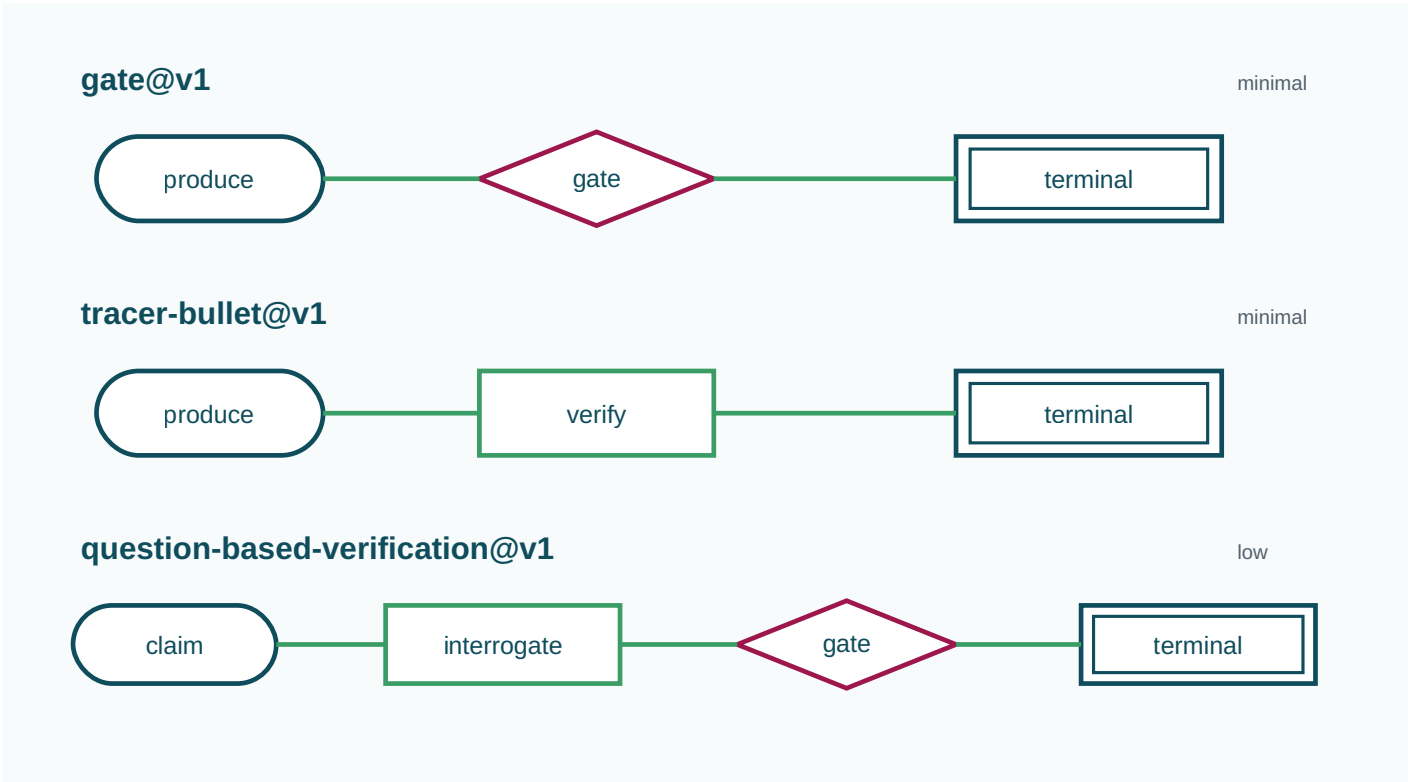
Contract: predicate, version, threshold, and input revision are explicit. Acceptance is deterministic for that input. **Refuses:** malformed input, absent evidence, threshold failure.

Proves: predicate result. **Does not prove:** that the predicate is the right proxy.

REPLAY

Start at accept or refuse, recover the exact predicate and input, then locate the producer receipt.

Review station ≠ verification station



A3 · Review station

Contract: question, reviewer role, evidence set, and independence boundary. Findings remain distinct from disposition; dissent survives. **Refuses:** incomplete evidence or reviewer conflict.

Does not prove: independent assurance when evaluator and inputs are correlated.

A4 · Verification station

Contract: implementation and test receipt converge with method, fixture, environment, and subject revision. Unknown is explicit. **Refuses:** environment drift or unreproducible receipt.

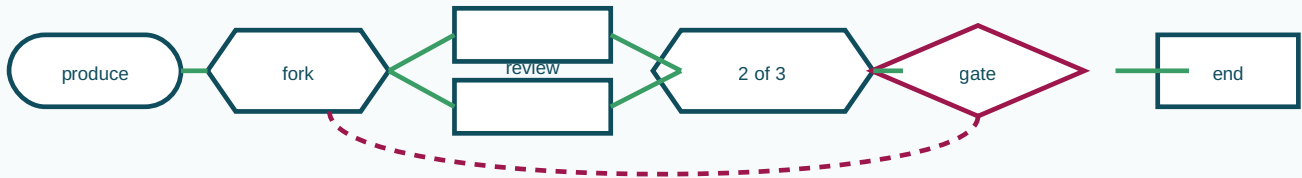
Composes: gates, freshness checks, regression.

AUTHORITY
A reviewer can report and recommend. Approval requires the authority and decision record declared for the consequence.

Parallel challenge and bounded repair

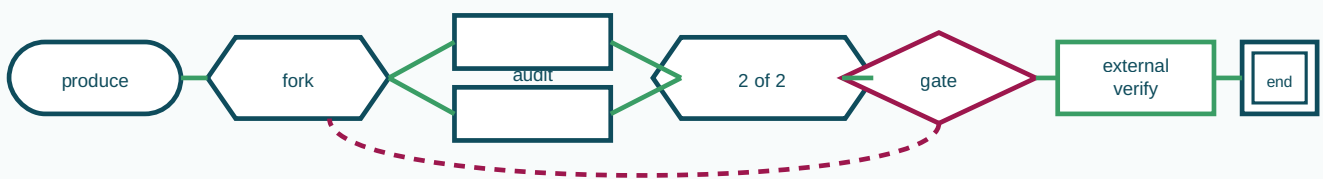
audit-wave@v1

cap 5 / min 1 / stable 2



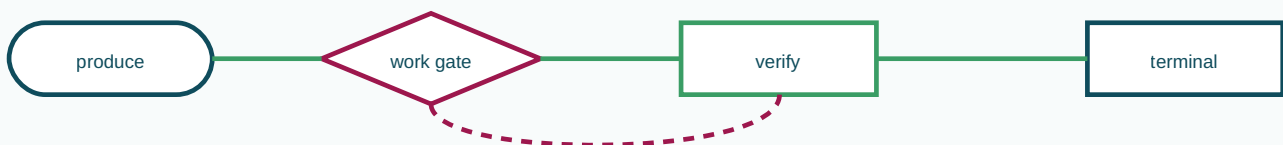
external-audit@v1

loop plus out-of-loop verifier



harden-loop@v1

cap 4 / breaker 3



A5 · Fork / join

Contract: branch count, quorum, independence assumptions, timeout, and dissent handling. Branches may disagree. **Refuses:** quorum achieved without required independent branches.

Does not prove: truth because K-of-M agrees.

A6 · Bounded loop

Contract: retry budget, progression condition, stale-result rule, and escalation. Every iteration is recorded; cap exhaustion is terminal. **Refuses:** retry without new evidence.

Composes: repair and reconciliation stations.

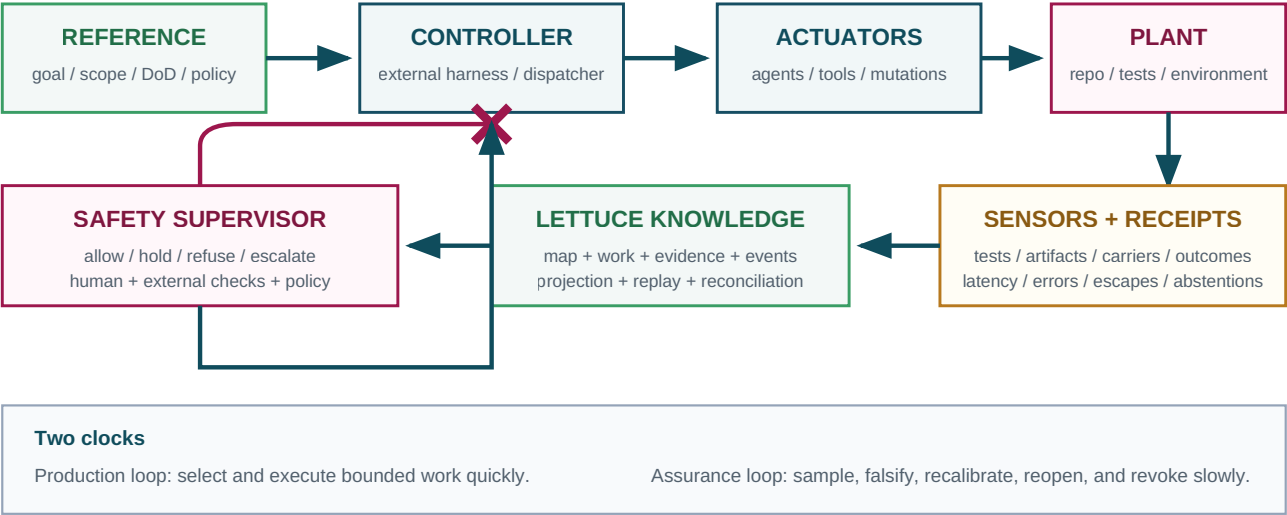
REPLAY

At a join, preserve each branch basis. In a loop, show what changed between iterations.

A person is not a green node

The ledger is memory; the factory still needs a supervisor

A controlled shift closes the loop from policy to action to evidence to the next authorized move.



Shape

An evidence bundle enters a decision node and exits accepted, refused, or deferred. Every exit has declared effects and conditions.

Contract

Authority, question, alternatives, rationale, conditions, effects, dissent, and supersession are required. Identity alone is insufficient.

Refusal

Missing rationale, unclear authority, incomplete alternatives, or effect beyond permission keeps the decision open.

PROVES / DOES NOT PROVE

Proves: an accountable decision was recorded under the declared authority. **Does not prove:** expertise merely from a human label or transfer responsibility to the graph.

Assemble without hiding the seams



reference → planner → implementer → mechanical checks
→ independent review fork/join → semantic review
→ human decision → carrier → release gate → terminal

Contract

Every handoff has a carrier, every effect has authorization, every join preserves dissent, and every terminal claim can be replayed backward.

Behavior

The first keyboard run refuses on lost focus. A bounded loop repairs it; mechanical and semantic review remain separate.

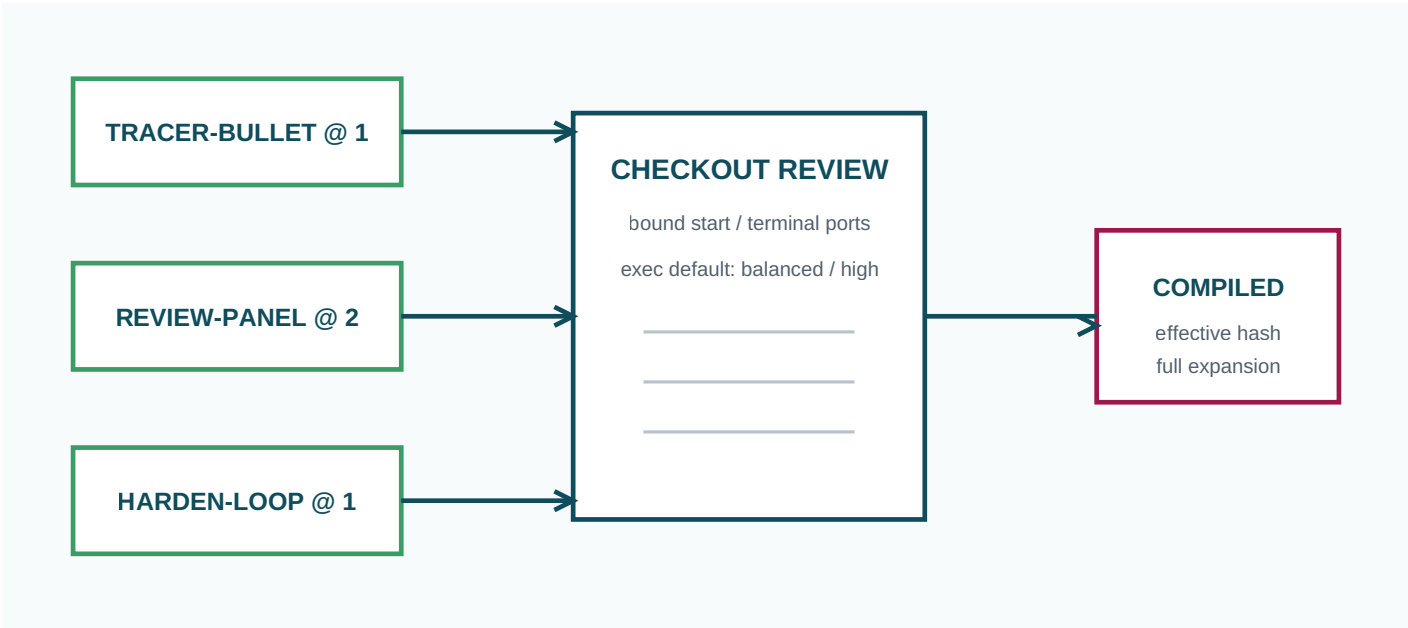
Refusal

Skipped station, stale evidence, correlated assurance presented as independent, or unauthorized effect prevents release.

DOES NOT PROVE

The composition does not prove the factory discovered every relevant dimension or that a successful checkout represents every user.

Read the checkout backward



1 · Terminal to decision

Which release gate accepted which carrier? Which authority made the consequence-bearing decision, under what conditions?

2 · Decision to evidence

Recover semantic review, independent branch receipts, mechanical check, subject revision, browser, fixture, and repaired focus behavior.

3 · Evidence to refusal

Trace the repair to the original typed refusal. Confirm the retry changed evidence and did not replace or erase the failed run.

FIRST UNSUPPORTED LEAP

If “keyboard path passed” silently becomes “checkout is accessible,” stop: the graph widened a bounded receipt into semantic certainty.

Graph inspection checklist

1. Name entry, exits, terminal states, and loop bounds.
2. List every required input and produced output.
3. Find the carrier at every seam.
4. Separate declared instructions from runtime enforcement.
5. Mark mechanical, agentic, and human authority.
6. Inspect effect permissions at the exact node.
7. Demand at least one typed refusal path.
8. Preserve branch dissent and independence assumptions.
9. Tie receipts to subject revision, method, and environment.
10. Define freshness and regression behavior.
11. Replay a success and a refusal backward.
12. Write “proves / does not prove” before accepting the card.



A trustworthy graph makes its limits as inspectable as its successes.

Return to Issue 01 for the map, Issue 02 for the operating agent, and Issue 03 for maintenance after green. The deeper treatment lives in Inside Huru’s Glassbox.