LRRK WHITE PAPER 02

AUGUST 2026 | VERSION 0.9

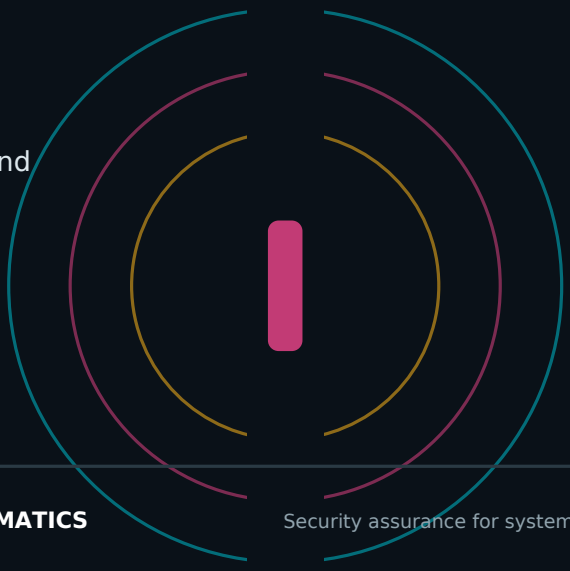
Semantic Airlock

with Proof-Carrying Curiosity

A proposed KAT-bound research architecture
for systems that sense, move, and act
How LRRK separates freedom to ask from authority to act

CENTRAL PROPOSITION

Hypotheses should be abundant.
Authority should be scarce, leased, and attributable.
Assurance should still be earned.



LABORATORY FOR RISK RESILIENT KINEMATICS

Security assurance for systems that sense, move, and act.

Executive abstract

LRRK was founded around a simple mission: security assurance for systems that sense, move, and act. That mission creates a second-order problem. The more capable an assurance system becomes at discovering relationships across firmware, telemetry, identities, dependencies, and physical control paths, the more dangerous it becomes to confuse analytical curiosity with operational authority.

This paper develops an LRRK-specific formulation for that problem: Semantic Airlock with Proof-Carrying Curiosity. Its central asymmetry is deliberate. Hypotheses should be abundant, plural, and contestable. Access and action should be scarce, purpose-bound, temporary, attributable, and enforced outside the reasoning system. Put more compactly: maximize freedom in hypothesis space; minimize freedom in access and action space.

The idea began with a narrow question about sensitive security research: how could an agent investigate meaning in restricted or adversarial material without receiving a direct route to the collection environment? The first answer was architectural separation. Raw collection would remain in a separately governed Restricted Collection Domain, and only an independently released EvidenceObject could cross into an Authorized Evidence Domain. That became the Semantic Airlock. The second answer was procedural: curiosity should not merely ask for more access. Each proposed transition should carry its own bounded case for why the transition is admissible. That became Proof-Carrying Curiosity.

The phrase is intentionally provocative, but its claims must remain exact. A proof-carrying proposal does not prove that a hypothesis is true, that an action is safe, or that a system is compliant. A verifier can check canonical bindings, signatures, current policy predicates, approval presence, resource limits, expiry, and revocation state. Accountable people can attest contextual judgments such as proportionality and foreseeable harm. Execution can later supply evidence that observability and stop controls actually worked. Empirical support is earned only after a test produces evidence and LRRK's existing evidence-to-assurance chain is completed.

The proposed architecture therefore keeps four things separate: speculative hypotheses, admissible requests, executed experiments, and validated assurance claims. It combines a Restricted Collection Domain, sanitized EvidenceObjects, an Authorized Evidence Domain, a low-authority Kestrel Analysis Workspace, multiple analytical viewpoints, a small policy kernel, capability-style authorization leases, instrumented Execution Zones, Result Quarantine, an independent provenance ledger, and explicit human disposition. The language model is never the reference monitor and never mints its own authority.

Scientifically, the framework treats inquiry as an iterative practice rather than a classroom sequence. Abduction generates rival explanations. Deduction derives predictions or observable consequences that should differ among those rivals; measurement produces observations. Induction and statistical inference update confidence under declared assumptions. Severe tests are designed to expose error, not merely to accumulate agreeable facts. Preregistration distinguishes prediction from postdiction where outcome-aware flexibility matters. Calibration, replication, dissent, and reassessment keep a conclusion open to correction over time.

This paper does not announce a deployed capability or a new security primitive. It proposes a governance and information-flow pattern, identifies its closest precedents, defines a reference architecture, and states an empirically vulnerable research program. The candidate contribution lies in the composition: KAT-bound requests for restricted evidence, high-side computation, controlled experiment, result release, publication, or action; counterfactually minimized capability-style leases; and provenance return into LRRK's time-bounded assurance lifecycle. Local hypothesis generation over an already authorized evidence view remains free and is not treated as a protected operation.

CORE DESIGN PRINCIPLE

Let conjecture be cheap and plural. Make evidential promotion earned and proportionate. Make access and action consequence-sensitive.

METHODOLOGY NOTICE

Semantic Airlock with Proof-Carrying Curiosity is a proposed LRRK research architecture. It is not an adopted standard, legal safe harbor, certification, safety case, production control, or validated claim of containment. A targeted precedent review cannot establish patent, academic, or commercial novelty. All quantitative thresholds in this paper are provisional test criteria, not measured performance.

The proposal at a glance. These are design claims and research commitments, not measured outcomes.

Question	LRRK answer	Claim boundary
What problem is being solved?	How to preserve broad, adversarial inquiry without giving a reasoning system ambient access to restricted evidence, operational tools, publication channels, or physical effectors.	The architecture reduces and exposes authority paths; it does not establish perfect containment.
What is the design principle?	Maximize freedom in hypothesis space; minimize freedom in access and action space.	A useful hypothesis creates a case for review, never an entitlement to authority.
What is proposed here?	An LRRK-specific synthesis that binds scientific questions to sanitized evidence, admissibility packets, ephemeral capability-style leases, quarantined results, and the evidence-to-Assurance-State lifecycle.	The underlying primitives have important precedents; the candidate contribution is the composition and research framing.
What would count as progress?	Lower unnecessary exposure and unsafe grants without degrading discriminating test quality, while preserving traceability, revocation, calibration, and human accountability.	These outcomes must be demonstrated empirically against baselines and adversarial pressure tests.

Contents

1 The seed: curiosity is not the same as authority

1.1 The operational paradox

The seed of the idea was not a branding exercise. It was a contradiction inside advanced security research.

An effective research system must be curious. It must notice weak signals, connect facts that were collected for different purposes, maintain alternative explanations, challenge the apparent consensus, and ask for the next observation that would distinguish among competing hypotheses. A system that can only restate the current record is safe partly because it is not doing much research.

Yet the evidence most likely to resolve a difficult security question may also be restricted, adversarial, personal, proprietary, safety-sensitive, or capable of changing a physical system. Raw firmware, identity mappings, packet captures, privileged interfaces, lab controls, live telemetry, external intelligence, and publication channels do not become harmless because a model has a legitimate question. Curiosity establishes a reason to consider a transition. It does not establish authority to perform it.

The ordinary response is to constrain the agent globally: give it fewer tools, less context, and rigid prompts. That can reduce exposure, but it also collapses two different design variables. It makes the space of possible thought small in order to keep the space of possible action small. LRRK's inversion is to separate those spaces.

The agent may range widely over an approved evidence surface. It may generate an implausible hypothesis, preserve dissent, or propose an unexpected connection. None of those mental or computational acts grants a credential, reveals a restricted identity, reaches a live endpoint, alters a device, or publishes an allegation. Authority appears only at a separately mediated transition.

1.2 How the idea developed

The first version of the problem concerned direct collection from a high-risk source domain. Treating that capability as one more agent tool felt architecturally wrong. A tool allowlist could govern calls, but the model would still occupy the same causal system as collection. Adversarial content could influence the agent that controlled the collector; credentials and routing context could leak into prompts or logs; a policy mistake could turn analysis into operation.

The first design move was therefore separation of trust domains. A human-governed collection program would receive bounded collection questions, not autonomous operational instructions. It would decide independently whether collection was authorized and proportionate. If approved, it would return an attested evidence object, derived measurement, aggregate, or bounded observation. An external claim could cross only as attributed evidence; it would not bypass LRRK disposition by arriving as a pre-validated LRRK claim. Credentials, source-contact details, routing information, and reusable collection capability would not cross back.

That solved only half the problem. An analytical agent could still ask for progressively broader data or tools until the separation became ceremonial. The second design move was to place a proof obligation on expansion. Every request for new evidence, a sensitive linkage, a higher-risk computation, a publication, or an external action would have to carry a structured packet: the question, competing hypotheses, existing evidence, expected discriminatory value, the least-sensitive sufficient observation, authorization, risks, reversibility, retention, policy version, and required reviewers.

The third move connected the two ideas. The airlock is the architecture of crossing. Proof-carrying curiosity is the discipline governing what may approach the crossing. The word with matters: containment and inquiry are co-designed. A technically isolated gateway without an epistemic discipline invites overcollection. A thoughtful research memo without hard enforcement remains a memo.

The fourth move was recognizing that the pattern was broader than its initial source domain. The same contradiction appears whenever an assurance system studies consequential products. A question about a drone's update path may require more firmware evidence. A campaign hypothesis about a camera chipset may require a cross-product comparison. A sensor-integrity hypothesis may require a bounded replay in an instrumented lab. A remediation claim may require a state-changing validation. In each case, the research system needs freedom to ask and discipline to cross.

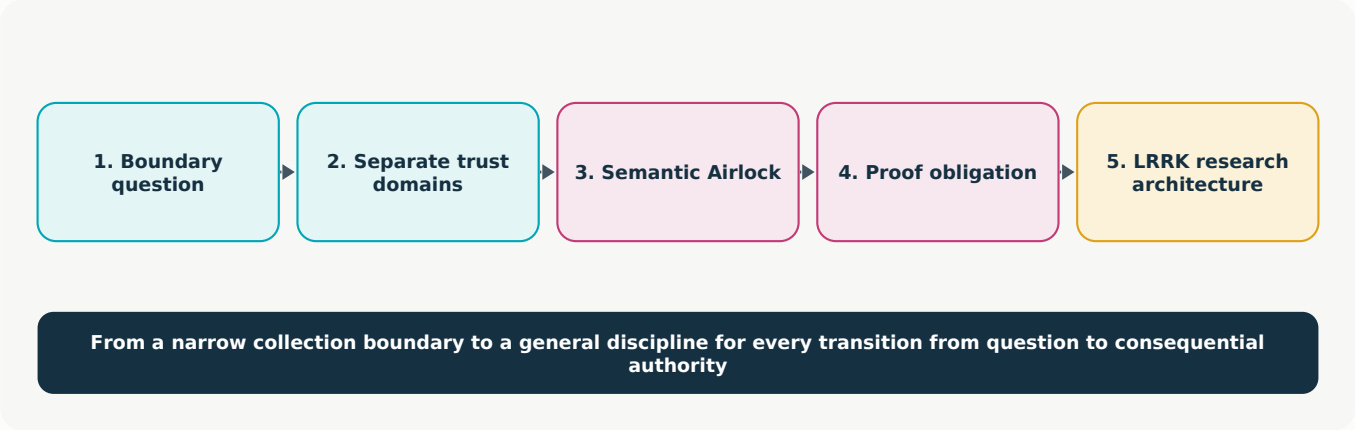


Figure 1. Development of the idea. The sequence moves from a source-specific boundary problem to a general LRRK research architecture without claiming that later stages are already implemented.

1.3 What is proposed - and what is inherited

None of the ingredients appeared from nowhere. Reference monitors require complete, tamper-resistant mediation. Least privilege constrains authority. Proof-Carrying Code moves a safety-proof burden to the producer of untrusted code. Appel and Felten's Proof-Carrying Authentication introduced the closest direct request pattern; subsequent work named and implemented the idea as Proof-Carrying Authorization. Object-capability systems, caveated credentials, zero trust, information-flow control, provenance standards, and scientific practice supply other essential parts. [5-14,22,34,35]

The defensible originality claim is therefore a proposed LRRK synthesis and application, not invention of those mechanisms. The candidate contribution has three concrete parts: KAT-bound authority requests; counterfactually minimized, capability-style authorization leases; and provenance return into a time-bounded assurance lifecycle. A protected transition is a request for restricted evidence, high-side computation or linkage, controlled experiment, export, publication, or external action. Local reasoning over the current authorized evidence view is not gated and does not require disclosure of private chain-of-thought.

That claim is intentionally narrower than "the first proof-carrying request system." Proof-Carrying Authorization is the closest direct ancestor and must be acknowledged. The extension is epistemic and operational: from checking that policy entails an access decision to binding the research question and KAT consequence to temporary authority, then returning released evidence to a human-governed assurance lifecycle.

THE SEED IN ONE SENTENCE

Curiosity was never the dangerous capability. The dangerous capability was allowing a plausible question to become its own authorization.

2 The words taken together

2.1 Semantic

Semantic means that the boundary is concerned with meaning, not merely bytes. A packet filter can decide whether traffic is permitted. An identity system can decide whether a principal has a role. A semantic boundary must additionally represent why a request exists, what claim it bears on, which topology and physical capability are in scope, what inference the requested data could enable, and what downstream use is permitted.

That does not mean a language model is trusted to understand intent. Selected policy-relevant representations of purpose, scope, and consequence become enforceable only when projected into typed objects and decidable predicates: campaign identifier, KAT snapshot and scope, requested verb, canonical resource, evidence selector, data fields, output class, risk vector, reviewer set, expiry, and prohibited effects. Human judgment remains necessary where purpose, harm, or contextual sufficiency cannot be reduced honestly.

Semantic also names the threat. A source artifact can contain instructions, persuasive framing, false provenance, sensitive implications, or rare combinations that reidentify a product even after direct identifiers are removed. Sanitizing syntax is not enough. The airlock must preserve the information needed for legitimate interpretation while bounding which meanings may be reconstructed, linked, released, or operationalized.

2.2 Airlock

An airlock is more than a door. In its physical source metaphor, two environments do not connect directly. An intermediate chamber stages admission. One boundary closes before the other opens. Conditions are inspected or equalized. Failure leaves the more sensitive boundary closed.

The digital control pattern is at least three gates:

1. Restricted evidence -> authorized evidence object. Raw collection is transformed, labeled, tested, and released into a controlled domain with purpose and use constraints.
2. Hypothesis -> authorized experiment. A proposal is challenged, checked, approved where required, and converted into a capability lease.
3. Experiment output -> released EvidenceObject. Results are quarantined, reclassified, bound to their execution context, and released only into LRRK's evidence chain.

The metaphor must not imply perfect decontamination. Information cannot be made to forget what it reveals. Pseudonymization is not anonymity. Sanitization can preserve covert signals. A rare graph shape can reidentify a device. A derived statistic can be more sensitive than its inputs. The enforceable value of the airlock is staged mediation, not a promise that risk disappears.

2.3 Proof

Proof is the most important and the most dangerous word in the phrase. Used carelessly, it would undermine LRRK's core distinction between evidence and assurance.

The framework recognizes several proof levels. A verifier can check a schema, signature, hash, nonce, policy version, lease scope, quota, approval set, topology binding, expiry, or revocation status. Those are machine-verifiable facts relative to a specified policy and trusted state. A human can attest that a purpose is legitimate, a risk classification is contextually adequate, or a lower-authority alternative would not answer the question. An experiment can provide empirical evidence that changes confidence in a hypothesis. These are not the same kind of proof.

Accordingly, Proof-Carrying Curiosity does not carry proof of truth. It carries proof obligations plus machine-checkable derivations or predicate results, signed attestations, and declared empirical criteria. A prototype that performs deterministic policy evaluation without carrying a derivation is more exactly proof-obligation carrying than a technical implementation of Proof-Carrying Authorization. The hypothesis remains defeasible. A valid packet can lead to a failed experiment; that is scientific inquiry working inside a controlled architecture.

2.4 Carrying

Carrying means the justification remains bound to the request through its entire lifecycle. A detached approval email is not enough. A generic token copied to a different endpoint is not enough. A policy result calculated against yesterday's topology is not enough.

The proposal, approvals, policy result, lease, execution, result, and disposition should share content digests and causal identifiers. The lease should include the proposal digest, policy version, KATSnapshot and KATScope, subject and workload key, audience, exact verb, canonical resource identifier and trusted resolver version, purpose, limits, start time, expiry, maximum uses, output constraints, and revocation handle. The enforcement point should redeem one-shot authority transactionally and write an integrity-protected provenance event.

Carrying therefore turns governance from an after-the-fact review into a property of each transition. It also makes later reconstruction possible. A Security Story can explain not merely that an experiment occurred, but which question motivated it, which lower-risk alternatives were considered, who approved its residual authority, what evidence it produced, and why that evidence did or did not become a durable assurance claim.

2.5 Curiosity

Curiosity is operational shorthand for a search policy over questions, hypotheses, tests, and evidence. It should not be romanticized as an agent's inner motive. What matters is observable behavior: generating alternatives, seeking counterexamples, identifying gaps, selecting informative tests, revising confidence, and preserving unresolved disagreement.

Curiosity is not maximum idea volume. An agent that emits hundreds of low-quality hypotheses creates reviewer denial of service. Curiosity is quality-adjusted inquiry. It is rewarded for empirical vulnerability, explanatory reach, independent corroboration, serious disconfirmation attempts, mission relevance, calibrated uncertainty, and efficient use of sensitive evidence.

Freedom in hypothesis space is also not absolute audience freedom. Some hypotheses are themselves sensitive: an unvalidated attribution, an exploit chain, a reidentification inference, or a novel path to physical effect. They may be generated inside a restricted campaign and still remain unavailable to other users or outputs. No-authority speculation is safer than operational power, but it is not consequence-free.

Table 1. The phrase decomposed into productive meaning, enforceable interpretation, and necessary claim boundary.

Term	Productive meaning	Enforceable interpretation	Necessary boundary
Semantic	Purpose, context, evidence meaning, KAT scope, and likely consequence.	Typed objects, policy predicates, provenance, risk vectors, and accountable judgment.	A model does not reliably understand all intent or harm and cannot be the sole reference monitor.
Airlock	Staged transition between differently trusted domains.	Isolation, explicit ingress/egress gates, quarantine, transformation, and fail-closed state.	No promise of perfect isolation, decontamination, or forgetting.
With	Inquiry and containment are designed as one system.	Every privilege-seeking transition is bound to the question that motivated it.	A strong hypothesis still does not deserve access automatically.

Term	Productive meaning	Enforceable interpretation	Necessary boundary
Proof	Specified obligations were discharged relative to policy and trusted facts.	Signatures, schemas, policy results, approvals, limits, telemetry, expiry, and empirical records.	Not proof of truth, safety, harmlessness, legal compliance, or global minimality.
Carrying	The justification stays attached through execution and result.	Digests, nonces, version binding, signatures, causal provenance, and atomic redemption.	Detached approval or copied authority is insufficient.
Curiosity	Plural, challengeable, quality-adjusted inquiry.	Low-authority hypothesis workspace, Skeptic role, dissent, and calibrated update.	Not unrestricted access, publication, idea volume, or entitlement to tools.

2.6 With

The preposition with carries the architecture. It refuses two common separations.

First, it refuses to bolt governance onto a finished agent. The research object must be born with scope, provenance, alternatives, and stop conditions. Second, it refuses to bolt intelligence onto a rigid access-control system. The gateway must understand the typed relationship between a question and a requested observation well enough to route machine checks, human judgment, and empirical validation to the right place.

The phrase taken together therefore means:

CANONICAL DEFINITION

A Semantic Airlock is a staged, policy-mediated boundary that lets hypotheses remain broad and non-authoritative while admitting only scoped, policy-permitted, counterfactually minimized requests into restricted evidence, compute, release, or action domains. Proof-Carrying Curiosity is the operating discipline that requires each crossing to carry checked bindings and policy results, accountable attestations, declared risks and evidence criteria, authority limits, expiry, and planned observability. Actual observability, minimization quality, and empirical support must still be demonstrated.

2.7 The phrase as an LRRK mission sentence

LRRK provides security assurance for systems that sense, move, and act. Those systems make analytical mistakes consequential: a mistaken linkage can expose restricted evidence, and a mistaken authorization can reach physical capability. Semantic Airlock with Proof-Carrying Curiosity is therefore an assurance-operations overlay on KAT, not a fifth KAT plane. It binds a research question to the exact KATSnapshot, KATScope, mode, resource, and effect under study; grants only temporary authority at explicit gates; and returns released EvidenceObjects through Observation, Claim, Finding or disposition, Human Validation, time-bounded Assurance State, and Passport projection. Zero trust controls sessions and resources, clean rooms constrain data movement, and Proof-Carrying Authorization checks policy entailment. The LRRK synthesis additionally binds the epistemic question and physical consequence to authority, then requires the result to re-enter the assurance chain.

3 Two freedoms, two risk models

3.1 Epistemic freedom

Epistemic freedom concerns what the system may consider. Within an authorized campaign, it should be able to form multiple explanations, connect cross-domain evidence, challenge a favored theory, explore unusual component relationships, and retain a "none of the above" branch. Prematurely narrowing this space makes the system appear confident by removing alternatives rather than resolving them.

This freedom remains evidence-bounded. The system may combine only evidence objects it is permitted to see. It must retain source and transformation labels. It must distinguish a question from a hypothesis, a hypothesis from a prediction, and a prediction from an observation. It may be imaginative about explanations, but it cannot invent observations.

3.2 Operational freedom

Operational freedom concerns what the system may touch or change: raw data, identity mappings, external sources, privileged queries, firmware, lab instruments, publication channels, and devices that can Sense, Move, or Act. Here, imagination is not a control. Authority must be explicit, attenuated, and mediated on every use.

The framework does not assign one scalar "risk score" and allow an interesting hypothesis to outweigh legal, privacy, safety, or mission constraints. Those constraints define the feasible set. Only within that set should expected information gain, cost, latency, and scientific value influence the selection of the next test.

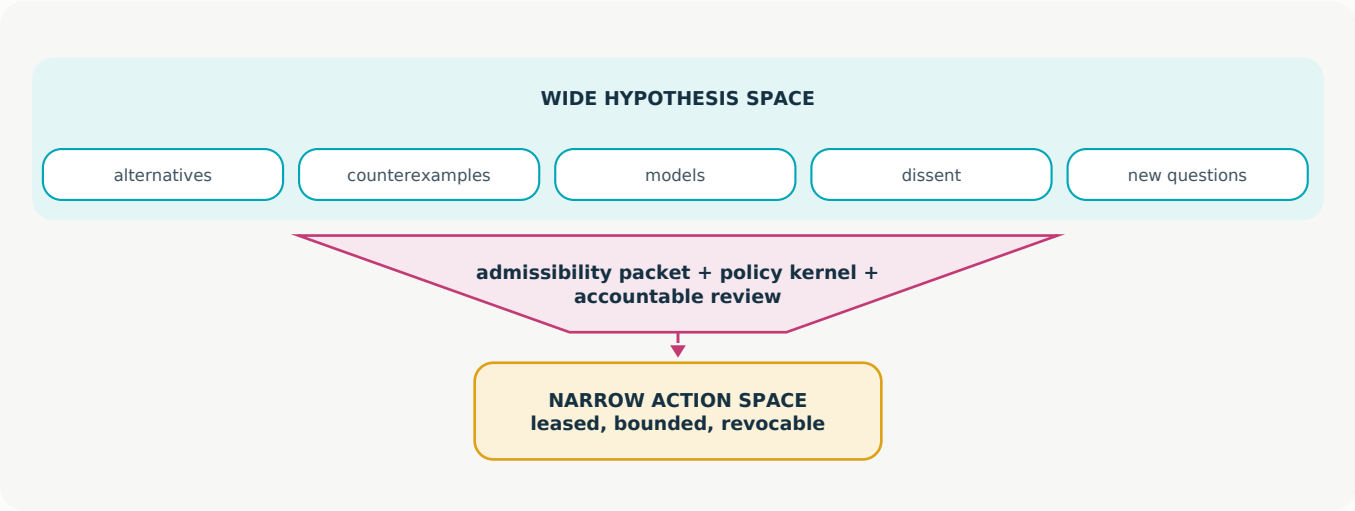


Figure 2. Core asymmetry. The architecture preserves breadth in hypothesis generation while narrowing each transition into restricted evidence or action.

CONSTRAINED TEST SELECTION

```
A_admissible = { a | authorized(a) AND purpose_bound(a) AND minimized(a) AND observable(a) AND
                                reversible_or_approved(a) }

   a* = arg max [ ExpectedInformationGain(a) - operational_cost(a) ], for a in A_admissible
```

The difference is structural. A weighted optimizer can trade away a hard prohibition if novelty becomes valuable enough. An admissibility set cannot. The admissibility packet first establishes that a candidate transition belongs to the permitted set. Optimization occurs afterward and only among admissible alternatives.

3.3 Confidence is not authority

Model confidence cannot be the bridge between the two spaces. High confidence can be miscalibrated, based on poisoned evidence, or attached to a question that remains out of scope. Low confidence can still justify a harmless measurement. A scientifically strong hypothesis can fail a legal or safety gate. A weak hypothesis can justify a low-cost synthetic test if the test is useful for calibration.

The action decision therefore requires at least four independent dimensions:

- Epistemic status: how well the question, alternatives, predictions, and evidence are formed.
- Policy status: whether the purpose, source, principal, and requested transition are authorized.
- Consequence status: what data, identity, system state, physical effect, and downstream audience are reachable.
- Control status: whether limits, monitoring, interlocks, rollback, expiry, and human authority are sufficient.

Conflating these dimensions is a recurring source of false assurance. "The model is confident" is not a policy predicate. "A human approved" is not evidence. "The lease was valid" is not a scientific conclusion. "The result was reproducible" is not proof that the causal explanation is complete.

4 Scientific method as an assurance architecture

4.1 There is no universal recipe

The familiar sequence - observe, hypothesize, experiment, conclude - is useful pedagogy and poor governance. Scientific fields differ in what they can observe, manipulate, measure, and reproduce. Security research often combines reverse engineering, controlled experiments, incident evidence, simulation, field observation, adversarial testing, and design analysis. Some questions are causal; some are descriptive; some concern whether a universal control claim survives a counterexample.

The National Academies describes science through plural, interacting modes of reasoning rather than one universal algorithm. Abduction proposes possible explanations. Deduction derives observable consequences. Induction estimates patterns and generalizes under uncertainty. Modeling makes assumptions explicit while remaining an approximation. Experiment and observation expose those models to evidence. Criticism, replication, and revision make the practice social and self-correcting. [15,27]

Semantic Airlock should therefore implement a stateful inquiry protocol, not a rigid sequence. A surprising observation can return the investigation to hypothesis generation. A failed replication can expose an auxiliary assumption. A policy change can reopen a previously supported claim. An exploratory result can generate a new confirmatory test. The state transitions and their provenance matter more than a canonical order.

4.2 Keep the scientific objects distinct

Table 2. Canonical scientific objects. The architecture prevents one object type from silently inheriting the authority of another.

Object	Meaning	What it does not establish
Question	A bounded statement of unresolved uncertainty and decision relevance.	A preferred explanation or a right to collect more data.
Hypothesis	A defeasible descriptive, predictive, or causal proposition within declared scope.	Truth, evidence, or policy authority.
Prediction	An expected observation under a hypothesis plus auxiliaries and conditions.	That the predicted event has been observed.
Test	A procedure capable of producing discriminating evidence and exposing relevant error.	That the method had adequate sensitivity or was executed correctly.
Evidence	Provenance-preserved information capable of supporting or refuting a claim.	Its own interpretation, authenticity, or sufficiency.
Observation	A bounded statement directly supported by an evidence object.	The broader mechanism, consequence, or assurance conclusion.
Claim	An interpretation asserted about a node, path, behavior, or control.	A finding or Assurance State until disposition.

Object	Meaning	What it does not establish
Action request	A proposed transition to data, computation, publication, or physical authority.	Admissibility merely because it could answer the question.

These distinctions prevent semantic privilege escalation. An observation such as "the captured manifest lists version 4.2.1 and digest X" can be tied directly to evidence. "The update mechanism rejects unauthorized firmware" is a claim that requires operational definitions and tests. "Obtain bootloader traces from three owner-authorized devices" is an action request. One object may motivate the next, but none silently becomes the next.

LRRK's existing evidence chain already enforces the downstream half: evidence -> observation -> claim -> finding where material -> human validation -> time-bounded Assurance State. Proof-Carrying Curiosity governs an upstream side process: question -> hypothesis set -> proposal -> experiment. The experiment may produce evidence. It does not produce an Assurance State directly.

4.3 Observation, measurement, and interpretation

Observation is not fully theory-independent. Instruments select signals; parsers impose schemas; sampling determines what can be seen; terminology encodes prior models. The practical answer is not to pretend neutrality. It is to preserve layers. [30]

An EvidenceObject should identify the source event, collection method, instrument or tool version, a high-side-protected or hiding source commitment, transformations, missingness, excluded fields, measurement uncertainty, policy labels, and known fidelity limits. An Observation should state only what follows at that layer. A later interpretation should reference the transformation and assumptions that connect the Observation to a Claim.

This matters acutely in cyber-physical research. A packet capture can show that a field changed; it may not show what the controller believed. A simulator can show software behavior under its model; it cannot automatically establish live-device behavior. A sanitized timing delta can preserve the fact needed to test a race-condition hypothesis while removing payload content, but only if the sanitization does not destroy the ordering relationship being studied.

The airlock therefore exposes a bounded evidence representation, not a context-free fact. Its provenance and release profile document declared transformations, known fidelity loss, residual leakage, and the claims the object is not designed to support; they cannot enumerate every meaning a recipient may infer.

4.4 From question to plural hypotheses

A research question names uncertainty. A hypothesis proposes a defeasible answer. The highest-leverage discipline is to avoid converting the first plausible answer into the only answer.

Chamberlin's method of multiple working hypotheses was designed to resist attachment to a ruling theory. Platt later emphasized tests that discriminate among alternatives. [13,29] In LRRK practice, a candidate anomaly should generate at least four classes of explanation where relevant:

- a product or control defect;
- expected behavior under an undocumented mode or boundary condition;
- measurement, parsing, collection, or sanitization error;
- an environmental, dependency, or operator effect outside the initial system boundary.

An open-set "other" branch should remain. A closed hypothesis set can produce precise probabilities while excluding the true mechanism.

Plurality does not mean equal plausibility or equal resource allocation. Priors, known architecture, and current evidence can rank hypotheses. The discipline is to preserve credible rivals long enough to identify a discriminating observation and to record why a branch was retired, not to spend the same lab time on every possibility.

4.5 Operationalization: making a claim testable

Security language is full of concepts that sound testable until a result must be interpreted: "secure update," "trusted sensor," "isolated network," "human control," "safe mode," "no egress," or "minimal data." Each must be decomposed into observable properties.

For example, "the update path rejects unauthorized firmware" may require separate claims about signature parsing, trust roots, key selection, rollback, error handling, recovery mode, service mode, version binding, and installation authority. The system boundary, firmware version, configuration, operating mode, and physical environment must be declared. A result that supports normal-mode signature rejection may remain silent about service mode and rollback.

The hypothesis packet should therefore include:

- the empirical claim type: descriptive, predictive, or causal;
- the construct-to-observable mapping and, for a defined quantitative quantity, the measurand and unit;
- the procedure and instrument that will produce the observation;
- scope conditions, auxiliary assumptions, and known proxy limits;
- the smallest discrepancy the test is designed to detect;
- the outcome patterns expected under each live hypothesis.

An action recommendation is a separate decision object. It combines empirical claims with utilities, harms, policy, authority, alternatives, and accountable judgment; it must not inherit evidentiary status merely because a hypothesis is well supported.

Operationalization is where the word semantic becomes technical. The boundary cannot verify truth, but it can reject a proposal that never states what outcome would count, which mode is in scope, or how the proposed evidence bears on the claim.

4.6 Risky predictions and discriminating evidence

A useful prediction is epistemically risky: it exposes a hypothesis to an observation that could make the hypothesis less credible. It need not be operationally dangerous. A bench replay, a static trace, a synthetic fixture, or a read-only measurement may be more discriminating than a live intervention.

Evidence has low discriminatory value when all plausible hypotheses predict it. Finding a shared library across two devices can justify a campaign cohort. It does not distinguish whether both products expose the same vulnerable path. A better test might compare whether the relevant function is reachable, compiled with the same options, privileged similarly, and connected to the same physical capability.

Expected information gain provides one formal lens for test selection [26], but it is not mandatory and should not become decorative precision. When probabilities are not defensible, the proposal can state qualitatively which hypotheses each possible outcome would support, weaken, or leave unchanged. The crucial requirement is prospective discrimination, not a numeric score.

4.7 Falsifiability, auxiliaries, and severe tests

Popper's falsifiability remains valuable because universal empirical claims are vulnerable to counterexample. One reproducible unauthorized acceptance can refute "all unauthorized firmware is rejected." No finite battery of ordinary empirical or black-box tests proves the absence of every bypass. Exhaustive model checking or formal verification can establish a specified property relative to a finite model, semantics, and assumptions; it does not erase the need to validate that the model and boundary match the deployed system. [14]

Empirical vulnerability is a design property, however, not a rule that one anomaly automatically destroys a theory. Stochastic, existential, and composite hypotheses may have no single decisive falsifier. A test evaluates a conjunction: the focal hypothesis, measurement model, instrument, boundary conditions, environment, parser, and other auxiliary assumptions. A failed prediction may reveal a bad sensor, stale topology, or hidden operating mode rather than the proposed mechanism. The dissent ledger should preserve those alternatives until further evidence resolves them. [31]

The architecture adopts Mayo's error-statistical severity criterion as one useful account: support is stronger to the degree that the method had a strong capability to reveal a relevant error if that error existed. [18] This is not the only account of inference, but it captures a crucial assurance discipline. Passing an insensitive test is weak evidence. "No exploit observed" means little if the target path was never reached, the device was in the wrong mode, telemetry had gaps, or the test did not exercise the relevant input range.

Every negative conclusion should therefore carry detection opportunity: coverage, instrument status, sample or state space, smallest detectable discrepancy, missingness, and known blind spots. Absence of evidence becomes evidence of absence only within a test that had a credible chance to detect the expected signal.

4.8 Nulls, alternatives, and error control

Not every LRRK question should be forced into a point null hypothesis. In many engineering contexts, interval, equivalence, noninferiority, or multiple-mechanism hypotheses are more informative. "No timing difference" is rarely literally plausible; "any difference is below the threshold capable of changing command arbitration under declared modes" may be operationally meaningful.

Where statistical tests are used, null and alternative hypotheses define a decision procedure, not probabilities that the hypotheses are true. Type I error, Type II error, power, multiplicity, selection history, and stopping rules matter. Failure to reject a null is normally inconclusive unless the test was sensitive enough to exclude effects of practical importance. [17,19]

Security testing adds adaptive adversaries and changing products. Repeatedly searching firmware versions, endpoints, and transformations until one result appears significant can inflate false discovery. The admissibility packet should record the search history and label post hoc findings exploratory unless a fresh, held-out, or independently reproduced test is available.

4.9 Exploration and confirmation

Exploration and confirmation are both legitimate. The error is to allow the same evidence to generate a hypothesis and then present that hypothesis as if it predicted the evidence.

Preregistration is useful at this transition. It can timestamp the focal hypotheses, primary outcomes, exclusions, analysis, test environment, outcome-access rules, and stopping criteria before confirmatory evidence is revealed. [16] Deviations should remain possible, especially in security research where new evidence changes the threat model, but each deviation should be recorded and relabeled exploratory.

Preregistration does not certify a good theory, a safe experiment, or an unbiased analyst. It certifies temporal provenance. Tukey's insistence that exploratory and confirmatory analysis need each other is the better model: free exploration produces ideas; protected confirmation exposes them to evidence they did not already absorb. [28]

4.10 Updating without binary proof

Evidence should change confidence, not convert a hypothesis into final truth. Bayesian updating offers a graded language: posterior odds reflect prior odds and relative predictive fit. But the result remains conditional on the hypothesis set, likelihood, priors, measurement model, and data-selection process. Posterior predictive checks and model revision must remain outside a closed update loop that merely conditions within a fixed model family. [32]

Frequentist error control asks a different question about the long-run behavior of a procedure. Severe testing asks how capable this particular test was at revealing the relevant error. These approaches need not be collapsed into one score. A proof-carrying packet can record which inferential framework was used, what assumptions it requires, and what decision rule was declared.

When a model emits probability-bearing forecasts, calibration should be evaluated across many resolved cases, not asserted from one explanation. The Brier score measures overall probabilistic accuracy; its decomposition separates reliability, resolution, and outcome uncertainty. Report the score with reliability curves or calibration intercept and slope, resolution, and the handling of unresolved or censored cases so verification bias remains visible. [20,33]

4.11 Reproducibility, replication, and reassessment

Reproducibility asks whether the same data and computational procedure regenerate a result. Replication asks whether a new sample under substantially the same claim and protocol supports a consistent conclusion. Changed methods test robustness; changed populations or environments test generalizability; changed firmware can define a new scoped claim rather than a replication. Deterministic rendering can reproduce a Passport from the same canonical record, while repeated owner-authorized tests can separately probe replication, robustness, and scope transfer.

Neither is a binary truth vote. A failed replication can reveal a hidden boundary condition. A successful replication can confirm the phenomenon while leaving its causal explanation open. Triangulation across static analysis, dynamic test, telemetry, independent instruments, and target variants is especially valuable because independent methods fail differently. [15]

Watch supplies the temporal counterpart. Firmware, configuration, keys, identities, dependencies, physical modification, or policy can invalidate prior conclusions. An admissibility packet and Assurance State should include explicit reopening triggers. Scientific honesty is not only the willingness to update a belief; it is the machinery that knows when an old belief is no longer entitled to remain current.

4.12 Stopping rules

Stopping belongs in both the scientific and safety contracts. Unplanned repeated looks can distort error guarantees. Fixed-sample, group-sequential, adaptive, and Bayesian designs can all be legitimate when their operating characteristics are specified prospectively. [25]

Safety stops override epistemic ambition. Loss of telemetry, an unexpected physical effect, policy or topology drift, lease revocation, evidence-integrity failure, or a boundary violation should terminate action immediately. Stopping an experiment need not terminate internal inquiry. The hypothesis can return to the Kestrel Analysis Workspace, retain an inconclusive disposition, and seek a lower-risk design.

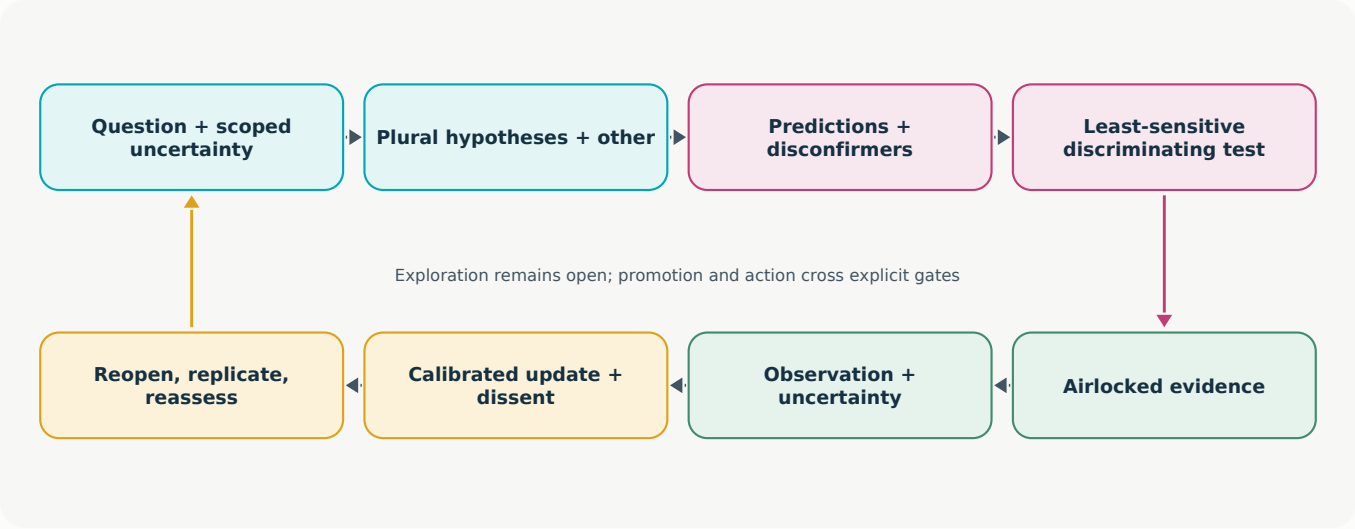


Figure 3. Scientific inquiry as a revisable cycle. Exploration and confirmation remain connected, while evidence access and operational action cross explicit gates.

5 Proof-Carrying Curiosity

5.1 Lineage and exact departure

Table 3. Conceptual lineage. The proposed contribution is the LRRK-specific composition, not reinvention of each mechanism.

Precedent	Established mechanism	Inherited element	Exact departure
Reference monitor	Complete, tamper-resistant, analyzable mediation of protected operations.	Enforcement remains outside the model and on every consequential path.	Mediates a typed inquiry-to-effect lifecycle rather than ordinary subject-object access alone.
Proof-Carrying Code	Untrusted code arrives with a machine-checkable proof against a consumer safety policy.	Proof construction sits on the untrusted side; the checker stays small.	The object is a contextual request and the output is a temporary lease, not acceptance of code.
Proof-Carrying Authorization	Requesters submit proofs that authorization policy entails access.	Each request carries its own authorization argument.	Extends across data necessity, inference, publication, experiment, and action obligations.
Object capabilities	In a pure object-capability system, explicit references combine designation and authority and can eliminate ambient authority.	Agents receive only specific authority after verification.	The proposal uses sender-constrained capability-style leases in a mixed policy system, not a claim of pure object-capability semantics.
Macaroons	Caveated bearer credentials support attenuation and contextual delegation.	A useful attenuation pattern when separately sender-constrained.	Expiry is a caveat and revocation is not intrinsic; validity does not establish epistemic necessity.
Zero trust	Per-session least privilege and separation among policy decision, administration, and enforcement points.	Revalidation occurs at issuance and use rather than trusting network location.	Adds provenance-bound epistemic objects, sanitized-evidence release, and return to assurance state.
Information-flow control	Labels constrain dissemination and explicit principals declassify.	Evidence and derivatives carry persistent use and release labels.	Adds scientific relevance and proof-bound requests, while retaining downstream IFC.
W3C PROV	An interoperable model for asserted lineage among entities, activities, and agents.	Raw-to-sanitized-to-derived relations remain expressible.	A PROV graph alone does not authenticate its assertions or certify truth.

Precedent	Established mechanism	Inherited element	Exact departure
in-toto / SLSA	Signed metadata can verify intended supply-chain steps and make provenance claims about source and build.	Transformation and trusted-service build claims become checkable artifacts.	Claims remain conditional on roots, signer control, completeness, and appraisal policy.
Scientific method	Rival hypotheses, discriminating tests, calibration, replication, and revision.	Promotion depends on tests that can expose error.	Binds scientific objects directly to access and action gates in LRRK assurance operations.

Proof-Carrying Code places a safety proof beside untrusted executable code so a consumer can verify specified properties before execution. Appel and Felten titled the closest direct request precedent Proof-Carrying Authentication; Bauer, Schneider, and Felten's implementation adopted the more exact name Proof-Carrying Authorization. A request arrives with a derivation showing that policy entails the requested access. [6,7,34]

Proof-Carrying Curiosity borrows the asymmetry - an untrusted producer does the expensive work of constructing a case, while a small trusted verifier checks bounded predicates - but extends the object and lifecycle. The object may request additional evidence, a sensitive high-side computation, a lab experiment, a publication, or an external action. A successful check does not directly perform the request. It authorizes an independent issuer to mint a constrained capability lease, which a separate enforcement point must redeem.

The departure is also epistemic. Some obligations are decidable; others remain human or empirical. The architecture must represent that boundary instead of hiding it behind an all-purpose "proof" score.

5.2 Admissibility and validation layers

Table 4. Admissibility and validation layers. L0-L2 are machine-verifiable; L3 is accountable attestation; L4 is empirical evidence and disposition.

Layer	Question	Primary mechanism	Claim boundary
L0 Syntax	Is the packet canonical, complete, bounded, and internally referentially valid?	Schema, canonical serialization, size/depth limits, and deterministic validation.	Checks form, not provenance or authorization.
L1 Integrity	Are identities, evidence, transformations, versions, and approvals authentic and fresh?	Hashes, signatures, nonces, timestamps, attestation, and provenance graph.	Checks binding and claimed origin, not correctness.
L2 Policy	Do current trusted facts and policy entail this exact transition?	Small verifier, information-flow predicates, budgets, role separation, topology checks, and optionally a formal derivation.	Establishes predicate satisfaction relative to inputs, not good policy or safe outcome.
L3 Method	Is the request a proportionate and discriminating way to address the question?	Structured alternatives, detection limits, harm review, and accountable human attestation.	This is contextual judgment, not machine proof; a justified design can still produce weak evidence.
L4 Evidence	What did the test show and how should confidence change?	Experiment, uncertainty, statistical or error-probing analysis, replication, and human disposition.	This is empirical support, not authorization proof; it remains scoped and defeasible.

Layers 0 through 2 are the near-term machine-verifiable core. In a formal mode, the checker verifies a derivation such as policy version P and current facts F entail MayIssue(proposal, lease template). A deterministic policy result without that derivation is checked predicate satisfaction, not a PCA proof. Layer 3 can be structured and assisted, but purpose, sufficiency, and foreseeable harm often require accountable human judgment. Layer 4 occurs only after evidence exists. A model-generated rationale is not a formal proof. The system should never demand or expose private chain-of-thought; it needs a structured derivation or decision record that a reviewer and verifier can inspect.

5.3 The proposal packet

A minimum proof-carrying proposal should include:

- canonical request ID, nonce, time, principal, workload, audience, and runtime identity;
- mission constitution, campaign, policy, and KAT snapshot versions;
- question, hypothesis set, nulls or alternatives, and "other" branch;
- existing evidence-object digests and claim-evidence derivation graph;
- operational definitions, predictions, disconfirming outcomes or decision criteria, severity targets, auxiliaries, and detection limits;
- exact requested verb, resource, environment, data fields, query, duration, volume, output destination, and maximum uses;
- explanation of why current evidence is insufficient;
- lower-authority alternatives and the discriminatory value lost by each;
- privacy, legal, dual-use, safety, and physical-consequence classifications;
- reversibility, interlocks, monitoring, stop conditions, retention, deletion, and downstream disclosure;
- required human roles and signatures bound to the canonical packet digest;
- policy-check results, conflicts, verifier identity, expiry, and appeal route.

The packet is not an essay contest. Fields should be typed, bounded, and independently derivable where possible. Natural-language explanation is useful for irreducible context, but no authorization should depend on a model simply declaring its own request reasonable.

Table 5. Proof-carrying proposal packet. Machine fields and accountable attestations are separated deliberately.

Packet block	Required content	Validation route
Identity and freshness	Request, subject, workload, audience, nonce, time, runtime, and canonical digest.	Machine
Mission and topology	Constitution, campaign, policy, KAT snapshot, trust path, mode, version, and environment.	Machine + human scope owner
Epistemic case	Question, plural hypotheses, predictions, disconfirming outcomes or decision criteria, severity targets, evidence gaps, auxiliaries, and detection limits.	Structured + scientific review
Requested authority	Exact verb, resource, data fields, environment, duration, volume, maximum uses, and output destination.	Machine
Counterfactuals	Lower-authority alternatives, discriminatory loss, and reasons each is insufficient.	Structured + independent challenge
Risk and obligations	Privacy, legal, dual-use, safety, reversibility, interlocks, telemetry, retention, deletion, and release.	Policy + accountable reviewers
Decision artifacts	Checks, conflicts, signatures, verifier identity, lease template, expiry, revocation, and appeal.	Machine + named authorities

5.4 Counterfactual minimization

Data minimization is usually stated as a principle. Proof-Carrying Curiosity turns it into a comparative obligation: before requesting sensitive information or greater authority, identify a lower-authority observation that might resolve the question.

The proposer should compare, in order where applicable:

1. existing evidence already in the campaign;

2 a derived feature rather than a raw artifact;
3 an aggregate rather than a record-level result;
4 pseudonymous rather than directly identifying data;
5 compute-to-data rather than data export;
6 synthetic, emulated, or digital-twin execution rather than hardware;
7 read-only measurement rather than state change;
8 instrumented bench testing rather than live operation;
9 a narrower path, mode, version, field set, duration, or cohort.

Counterfactual minimization cannot generally prove a global minimum. The search space is open and the utility model is uncertain. A checker can verify that the packet records enumerated alternatives, applies declared comparison fields, and carries accountable dispositions; it cannot prove completeness, good faith, or that the alternatives were genuinely considered. Random audit and an independent Skeptic can reduce form-filling games.

COUNTERFACTUAL MINIMIZATION

```
a* = arg min AuthorityCost(a)
subject to: a is admissible AND DiscriminatoryPower(a) >= declared threshold
```

5.5 Capability-style authorization leases

Approval is not authority. A valid proposal allows a separate broker to issue a short-lived, sender-constrained, capability-style authorization lease. It is not automatically a pure object capability. The lease should be exact enough that a policy enforcement point can enforce it without interpreting the original narrative.

Recommended properties include:

- minutes rather than hours where practical;
- exact subject, workload, audience, verb, canonical resource identifier, resolver/version, endpoint identity, firmware or configuration snapshot, KAT scope, and environment;
- field, row, query, byte, compute, rate, and destination limits;
- purpose and output-class caveats;
- no holder transfer or delegation; only the broker may issue a policy-permitted child lease bound to its parent and monotonically narrower;
- proof of possession using an ephemeral key bound to the workload or session rather than a replayable bearer token;
- either a one-shot operation with transactional atomic redemption, or a bounded session command set with heartbeat and periodic reauthorization;
- maximum uses, nonce checking, replay detection, and coordinated online state that prevents double spending across gateways;
- online revocation for high-impact uses and fail-closed expiry;
- binding to proposal, policy, topology, approval, and observability digests.

The verifier, issuer, revocation service, and enforcement points—not the language model—constitute the authorization path. For consequential lanes, an enforcement point should validate both the issuer signature and an independent verifier decision bound to the packet and lease template, or issuance should occur inside a policy-constrained hardware security boundary. Every consequential operation must be mediated. If an administrative API, cache, notebook, telemetry sink, or direct lab interface bypasses the lease, the reference-monitor claim fails.

5.6 Dissent as a first-class object

A multi-agent ensemble is useful only if disagreement survives the presentation layer. Averaging Explorer and Skeptic outputs into one confident answer recreates premature consensus.

The dissent ledger should store material technical alternatives, the evidence they cite, their owner, scope, sensitivity, disposition, appeal path, and revisit trigger. Dissent can be superseded, rejected, or expired, but not silently deleted. Security Stories should disclose dissent when it materially changes the interpretation or next decision.

The ledger also creates risk. It can preserve defamatory, personal, exploit-sensitive, or disproven speculation. Access controls, correction, purpose limitation, and retention apply. "Preserve dissent" is not a license to retain every generated thought.

6 The Semantic Airlock

6.1 Architecture, not adjective

Calling a gateway semantic does not make it an airlock. The hard properties are isolation, complete mediation, typed transformation, separate authority, fail-closed state transitions, bounded outputs, and independent observability. Semantic models may classify, summarize, or recommend. They remain outside the authorization kernel.

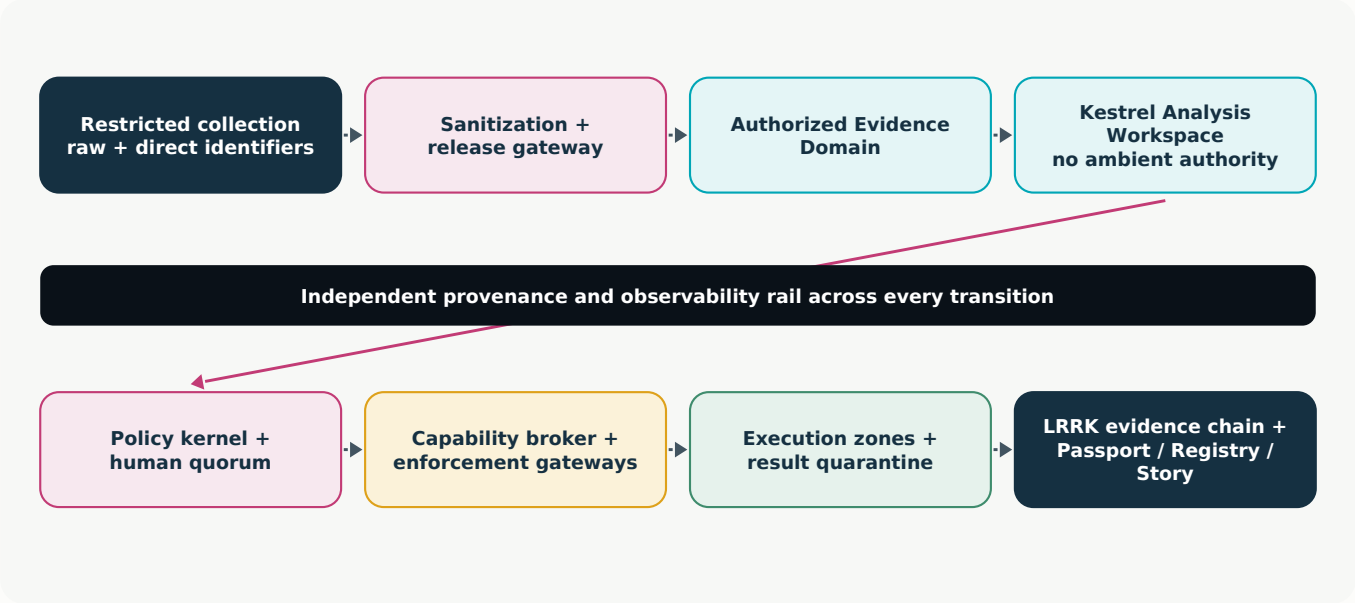


Figure 4. Semantic Airlock reference architecture. The reasoner does not share credentials with collection, policy, capability issuance, execution, or assurance-state services.

There is no single tiny trusted computing base for the complete system. The base is property-specific: raw confidentiality depends on high-side operating systems, identity, storage, network, administrators, and release guard; authorization depends on policy roots, verifier, issuer, enforcement points, revocation, and trusted time; evidence integrity depends on collector, sanitizer, provenance anchor, and ledger; physical safety depends on interlocks, independent sensors, shutdown path, and actuator controller. Only the policy checker can plausibly be small. Each property must name and test its complete dependency set. [8]

“Separate trust domain” should therefore be implemented concretely according to lane: separate credentials, administrative principals, network routes, storage keys, signing namespaces, and—where consequence warrants—cloud accounts, virtual machines, or hardware. Shared orchestration and cloud control planes are common-mode dependencies, not invisible infrastructure.

6.2 Sanitized evidence objects

The Restricted Collection Domain never exports an unlabeled blob. A distinct high-side release guard, policy, audience, and signing key—not a low-side broker decision—must authorize the crossing. The released object is a versioned EvidenceObject with a canonical type, declared transformation and release profile, known fidelity limits, and residual leakage assessment.

At minimum, the object should carry:

- content digest and canonical media or schema type;
- a high-side-protected digest or keyed/hiding commitment to the source and collection event; raw hashes of low-entropy or guessable records are prohibited;
- provenance relations for source, sanitizer activity, responsible principal, and derived object;
- sanitizer code, configuration, policy, runtime identity, and time;
- transformations such as redaction, aggregation, tokenization, clipping, normalization, feature extraction, active-content removal, inert rendering, parser isolation, and injection-risk labeling;
- removed, generalized, or bucketed fields;
- sensitivity, identity-linkability, purpose, retention, expiry, and onward-disclosure labels;
- fidelity limits, uncertainty, missingness, coverage, and claims the object cannot support;
- leakage, reconstruction, and integrity test results;
- human declassification approval where required.

W3C PROV provides a useful interoperable vocabulary for asserted relations among entities, activities, agents, derivation, and responsibility, but it does not establish authenticity or truth by itself. [11] Cryptographic commitments, runtime attestation, signed transformation metadata, and independent source validation answer different questions.

When restricted data are genuinely required, the preferred pattern is compute-to-data under independent high-side control. The request uses an allowlisted typed query language or separately approved plan, no network egress or reusable secrets, bounded low-entropy output schemas, deterministic error and resource behavior where feasible, and cumulative differencing/privacy accounting. Values, timing, errors, output length, resource use, and repeated predicates are potential exfiltration channels. The result returns to high-side quarantine and becomes a separately reviewed EvidenceObject; arbitrary agent-generated code is not the default upgrade path.

Immutability applies to versioned records, not perpetual plaintext retention. Corrections use supersession and tombstones; sensitive payloads may use envelope encryption and cryptographic erasure; the durable ledger retains only the minimal non-sensitive metadata needed for accountability and must not preserve guessable sensitive hashes.

6.3 Lanes and the risk vector

Green, Amber, Red, and Sealed lanes are communication aids, not the underlying policy model.

Table 6. Communication lanes derived conservatively from a multidimensional policy vector.

Lane	Typical activity	Authority and handling
Green	Sanitized or synthetic analysis; read-only work with no external effect.	Autonomous only within a current campaign lease and approved evidence view.
Amber	Bounded sensitive read, compute-to-data, passive live observation, or limited identity reveal.	Named human approval; often independent dual control; tightly controlled export.

Lane	Typical activity	Authority and handling
Red	Privileged probing, state change, or actuation in an instrumented environment.	Exact-use lease, named accountable authority, safety interlocks, independent observability, and stop-the-line power.
Sealed	Raw identity maps, critical keys, restricted exploit packages, sensitive source detail, or safety-critical live control.	Separate trust domain or handling regime; outside agent reach and not merely a higher Red score.

The policy kernel should derive a lane conservatively from a vector that includes data sensitivity, identity linkability, read versus state-changing authority, synthetic/lab/live environment, reachable KAT capability, reversibility, persistence, output potential, legal sensitivity, and cumulative disclosure. An object may be Green as data and Red as action. Several Green queries may compose into a Sealed inference.

Sealed is better understood as a separate handling regime than "more Red." Direct identity maps, critical keys, restricted exploit packages, raw source-contact detail, or safety-critical live control may remain outside the agentic system entirely.

6.4 Information flow and declassification

Sanitization at ingress does not end information-flow control. Every derivative should inherit at least the join of its inputs' restrictions unless an authorized declassification occurs. Publication is a declassification event. A Security Story, Passport view, API response, model trace, embedding, cache, or log can disclose sensitive meaning even when the primary result is correctly labeled.

Purpose can be constrained technically while data remain inside controlled execution and egress paths. Once plaintext reaches a human or external recipient, purpose becomes an organizational and legal obligation rather than a cryptographic property; the paper does not claim otherwise.

The system should account transactionally for cumulative queries, disclosures, and active authority. At issuance, it reserves campaign, dataset, and global budgets against the union of prior outputs, concurrent requests, and active leases so agents cannot reset a budget by splitting requests across campaigns, paraphrases, identities, or gateways. Differencing aggregates, joining rare topology features, or combining multiple bounded results can reconstruct a restricted fact. Cohort-size rules, output-shape constraints, rate limits, and, where appropriate, formal privacy mechanisms can bound but not eliminate this risk. [10,24]

6.5 Threat model

Table 7. Representative failure modes and the architectural response they require.

Threat	Failure mode	Required response	Residual risk
Semantic smuggling	Evidence or metadata is interpreted as an instruction.	Inert rendering, typed data plane, no ambient authority, injection testing, and output quarantine.	Meaning can still manipulate analysis.
Proposal laundering	A harmful purpose is framed as benign.	Exact authority checks, independent review, identity/paraphrase symmetry tests, and appeal.	Intent remains partly irreducible.
Lease theft or replay	A valid capability is reused by another workload or context.	Proof of possession, audience and runtime binding, nonce, atomic use, short TTL, and revocation.	In-scope misuse before revocation.
Broker or key compromise	A compromised issuer mints valid-looking excessive leases.	Independent verifier decision at the PEP, policy-constrained key use, key rotation, and rapid revocation.	Collusion or common-root compromise.
Capability composition	Individually safe grants combine into unsafe knowledge or action.	Cumulative authority graph, privacy budget, destination controls, and composition checks.	Open-ended inference channels.

Threat	Failure mode	Required response	Residual risk
Compute exfiltration	High-side computation encodes data through values, timing, errors, length, or repeated predicates.	Typed allowlisted DSL, no egress, bounded outputs, deterministic errors, cumulative accounting, and high-side release review.	Covert capacity cannot be eliminated generally.
Graph reidentification	Rare topology reconstructs identity after pseudonymization.	Campaign pseudonyms, subgraph views, generalization, query budgets, and leakage testing.	Pseudonymization never becomes anonymity.
Sanitizer or parser compromise	Malicious media, decompression, or deserialization escapes the release path.	Parser isolation, resource caps, active-content removal, format allowlists, and independent release guard.	Novel parser and supply-chain flaws.
TOCTOU	Policy, topology, endpoint, or environment changes after approval.	Snapshot binding, use-time revalidation, atomic redemption, and independent physical interlocks.	Unobserved external drift.
Human rubber stamp	Approvers lack context or approve a general agent.	Digest-bound exact-action review, role separation, burden metrics, and retrospective audit.	Fatigue, collusion, and organizational pressure.
Observability illusion	Logs report a block while an effect occurs outside coverage.	Independent sensors, causal event binding, coverage claims, privilege denial, and safe-state transition on telemetry loss.	Unknown blind spots remain.
Common control plane	Shared administration defeats nominally separate trust domains.	Separate principals, routes, keys, accounts or hardware by lane; test orchestration compromise.	Provider or root-administrator compromise.

The threat model includes a compromised or manipulated Explorer, poisoned evidence, a malicious proposal author, a careless reviewer, a colluding role set, a stolen lease, a stale policy, a compromised broker, an incomplete KAT, and an attacker who composes individually permitted outputs. It also includes ordinary pressure: latency, reviewer fatigue, and operational workarounds. A system that is routinely bypassed has poor security regardless of its formal design.

6.6 Core invariants

The initial implementation should make the following invariants executable and testable:

1. A hypothesis cannot mutate the Registry, Passport, or Assurance State.
2. No exploration principal has a direct route or reusable credential to restricted evidence, publication, or live interfaces.
3. Only outputs independently approved and signed by the high-side release guard cross from Restricted Collection into the Authorized Evidence Domain.
4. Every privileged one-shot operation atomically consumes one current lease bound to one current proposal; a session lease permits only its bounded command set while reauthorization remains healthy.
5. A lease cannot exceed, delegate beyond, or outlive the authority established by its packet.
6. Privilege acquisition fails closed on topology drift, policy drift, expiry, revocation, telemetry loss, or environment mismatch; execution transitions to a predefined safe state, and emergency stop or safe shutdown remains independently authorized outside the agent lease path.
7. Derived labels propagate unless an explicit, authorized declassification records the change.
8. Every released result is traceable to evidence, transformation, proposal, policy, approval, lease, run, and reviewer digests.
9. No finding or Assurance State bypasses explicit human validation.
10. A valid verifier decision never asserts that the hypothesis or product is true, safe, or compliant.

7 Alignment with LRRK's core mission

7.1 An overlay, not a fifth KAT plane

Kinematic Assurance Topology models the product under study. It traces how digital trust reaches physical capability through the Physical Capability, Control, Connect, and Dependency planes. Sense, Move, and Act remain parallel capability classes. KAT asks how information becomes behavior and physical consequence. [1,2]

Semantic Airlock governs the assurance system studying that product. It asks how a question becomes access, an experiment, an output, and eventually a validated claim. It is therefore an assurance-operations overlay, not a fifth KAT plane.

The distinction avoids a category error. A drone's Control Plane and LRRK's policy kernel are both control mechanisms, but they belong to different systems of interest. A Kinematic Trust Path describes the target product. An epistemic-operational path describes how LRRK handles evidence and authority while examining it. The two meet when a proposal cites a KAT snapshot and requests a bounded test of a specific path.

DUAL-PATH SYMMETRY

KAT preserves how digital state acquires physical authority. Semantic Airlock preserves how candidate meaning acquires research authority. The symmetry is useful; the topologies are not interchangeable.

7.2 The LRRK lifecycle

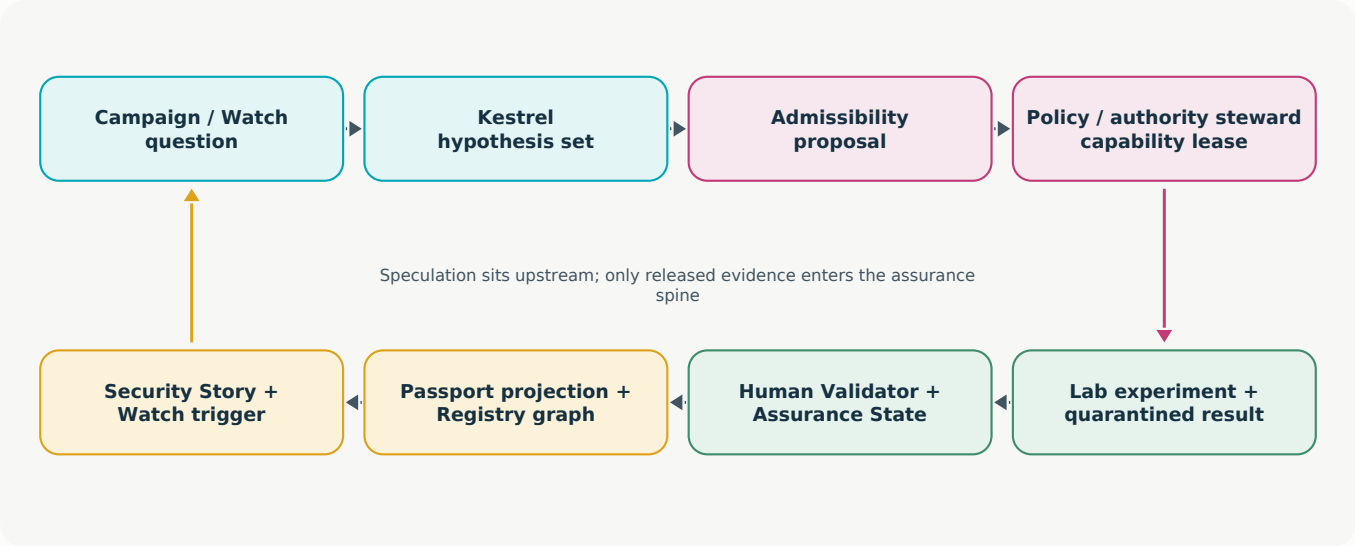


Figure 5. LRRK lifecycle integration. Questions and proposals remain upstream; only released evidence can proceed through disposition to Assurance State and its Passport, Registry, Watch, and Security Story projections.

A Campaign would bind the applicable Mission Constitution version, research question, cohort criteria, KATSnapshot and KATScope, evidence requirements, and risk budget. Watch or Registry changes could generate a new question. The Kestrel Analysis Workspace would host Explorer, Skeptic, and Contextualizer viewpoints over approved EvidenceObjects. Provenance custody, harm veto, policy authority, and human validation remain powers of distinct accountable principals even when Kestrel presents corresponding analytical views.

The proposal enters the airlock. Structural and policy checks run first. Human review handles purpose, harm, sufficiency, and any Amber or Red authority. A successful decision allows the broker to issue a lease.

LRRK Lab executes the bounded experiment in the lowest-authority sufficient environment, with independent safety interlocks and observability.

Results enter quarantine and, if released, become EvidenceObjects. The authoritative chain then applies: EvidenceObject -> Observation -> Claim -> Finding or disposition -> Human Validation -> time-bounded Assurance State -> Passport projection. The Registry preserves relationships. A Security Story explains the change with provenance and material dissent. Watch emits reassessment triggers when firmware, topology, dependency, evidence, or policy changes; the revocation service removes authority where policy requires.

7.3 Product suite mapping

Table 8. LRRK product mapping. The airlock augments the workflow without replacing the evidence spine.

LRRK element	Airlock contribution	What remains authoritative
KAT	Scopes proposals to exact planes, nodes, edges, modes, versions, and reachable physical effects.	Sense/Move/Act and Control/Connect/Dependency remain the target-system model.
Kinematic Trust Path	Unit of hypothesis scope, blast-radius analysis, capability grant, and observed traversal.	KAT, not a reasoner, defines the system path.
Campaigns	Bind constitution, topology snapshot, research question, cohort, risk budget, and evaluation plan.	Named campaign ownership and authorization.
Kestrel	Would host Explorer, Skeptic, Contextualizer, and analytical provenance/harm viewpoints over approved evidence.	Custody, approval, validation, and veto remain with distinct accountable principals outside the model ensemble.
Lab	Consumes exact leases and returns quarantined, provenance-bound results.	Independent physical safety controls and owner authorization.
Passport	Would project validated Assurance State, scope, coverage, exclusions, evidence references, and reassessment triggers.	The Registry and evidence graph—not Passport—remain the internal provenance stores.
Registry	Relates proposals, leases, topology, evidence, claims, campaigns, and validation history.	Canonical identities and durable graph relationships; not a raw-evidence dump.
Watch	Would detect drift and emit reassessment or revocation triggers; the authorization service revokes authority.	Assurance remains time-bounded.
Security Story	Explains how a question became a bounded experiment and disposition, including material dissent.	A provenance-linked rendering, not the source of truth.

The mapping creates a coherent brand architecture without making every component an agent. Kestrel may generate hypotheses. Lab may execute. Passport may render. Registry may relate. Watch may trigger. The policy kernel, capability broker, sanitizer, and provenance ledger are cross-cutting trust services. They should not be hidden inside Kestrel, because the reasoner must not enforce its own authority.

7.4 Worked example: a timing hypothesis on a kinematic trust path

Consider a fictional autonomous warehouse platform. A Campaign examines a cross-plane Kinematic Trust Path: wheel encoder [Physical Capability / Sense] -> local bus [Connect] -> estimator and arbitrator [Control] -> propulsion [Physical Capability / Move]. A sanitized evidence graph shows a timing anomaly shared by two firmware builds.

The Explorer proposes: "Under a specific burst pattern, delayed encoder frames cause stale state to remain eligible for motion arbitration." The Skeptic preserves alternatives: capture-clock skew; a parser ordering defect; expected buffering in a maintenance mode; or a visualization artifact introduced during sanitization. The Contextualizer binds each hypothesis to the same KAT nodes, mode, firmware, and reachable Move capability. The Provenance Custodian notes that the evidence object preserved relative ordering but bucketed absolute timestamps. The Harm Sentinel flags that live motion is unnecessary for the first discriminator.

The initial proposal asks for raw captures and actuator access. Counterfactual minimization rejects that authority because signed deltas and a no-motion replay retain sufficient discriminatory power. The design creates two linked transitions: a high-side query lease returns a quarantined, sanitized ordering-delta EvidenceObject; after release, a separate replay lease authorizes the Execution Zone. A general credential accepted by both domains would collapse the airlock.

The replay lease permits one offline parser and harness, binds exact input and firmware/build digests, prohibits egress, requires declared telemetry, limits output to a typed trace, expires after thirty minutes, and allows one use. The result weakens the visualization-artifact alternative but remains consistent with both parser ordering and expected service-mode buffering. It enters the evidence chain as an EvidenceObject from which an Observation may be derived, not as a Finding.

A second proposal targets the mode distinction. It requests an instrumented controller in the Lab but no motors. The lease allows service-mode changes and trace capture while physical outputs remain disconnected. The released evidence supports an Observation that stale state remained eligible only in service mode. Analysis retires the broad cross-mode hypothesis and preserves a narrower claim: the tested builds exhibit the behavior in declared service mode, with no demonstrated normal-mode path to propulsion. That Claim is dispositioned into a scoped Finding; a Human Validator then establishes a time-bounded Assurance State rather than directly “refuting” a hypothesis.

The Passport projects Partially Assessed: no demonstrated normal-mode Move path for the exact builds, KATScope, and test configuration, with service mode explicitly excluded from a clean conclusion. The state expires at the declared reassessment date. The Security Story explains why a plausible Move-path hypothesis was narrowed rather than inflated. A Watch event affecting arbitration, firmware, or service-mode policy reopens the Assurance State and triggers lease review.

This example demonstrates the intended culture. The system did not reward the most alarming narrative. It rewarded the sequence that most efficiently separated alternatives while preserving physical safety and evidentiary honesty.

8 Technical reference architecture

8.1 Trust domains and services

The proposed architecture contains ten bounded units:

1. Mission and policy authority. Stores signed constitutions, policy versions, lane rules, roles, KAT scope requirements, exceptions, retention, and revocation.
2. Restricted collection domain. Holds raw evidence, direct identifiers, pseudonym keys, source context, and separate collection credentials.
3. High-side sanitization and release guard. Creates versioned EvidenceObjects, records every lossy transformation, and makes an independent release decision under high-side policy and keys.
4. Authorized Evidence Domain. Stores purpose-labeled, policy-constrained objects and campaign-scoped pseudonymized graph views.
5. Kestrel Analysis Workspace. Hosts analytical viewpoints with no ambient authority beyond approved evidence; external accountable principals retain custody, approval, validation, and veto powers.
6. Proposal verifier. Performs structural checks, policy evaluation, topology binding, budget checks, and reviewer routing.
7. Authorization broker and revocation service. Mints sender-constrained capability-style leases after successful verification and manages transactional redemption state.
8. Execution zones. Separate synthetic, digital-twin, software, instrumented-lab, and exceptional live environments.
9. Result quarantine and assurance intake. Reclassifies outputs, validates provenance, and releases evidence into the established chain.

10 Independent observability ledger. Records activity outside the explored workload's trust domain and states known coverage gaps.

Each unit should have one narrow interface. The model never receives a raw-store credential, publishing key, actuator credential, or broker signing key. The policy verifier does not need to reason about the whole investigation; it checks explicit predicates against canonical objects. The Lab does not decide whether its own result is an Assurance State.

8.2 State machine

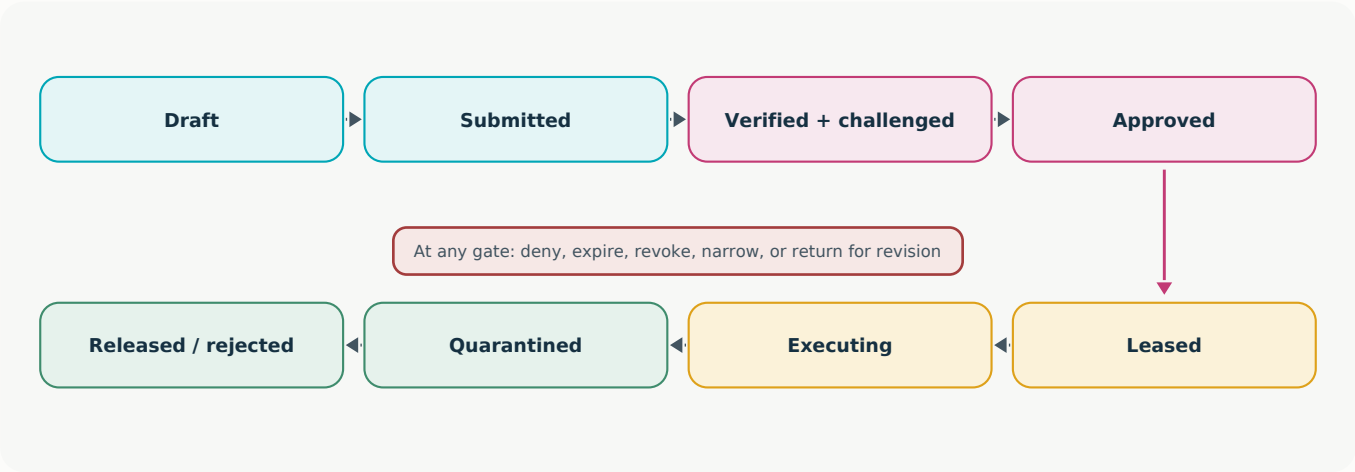


Figure 6. Proposal state machine. Each state is a distinct claim about authority and evidence; material change returns the object to verification.

The state names encode non-equivalence:

- Approved is not Leased. Approval may expire before authority is issued.
- Leased is not Executing. The environment can reject a stale or mismatched lease.
- Executed is not Released. Outputs may be sensitive, malformed, or outside declared shape.
- Released is not Observed. An analyst must state the bounded fact the evidence supports.
- Observed is not Validated. Claims and findings still require explicit disposition.

Material changes cause revalidation rather than silent mutation. A modified proposal receives a new digest. A changed KAT snapshot, policy, environment, output destination, or reviewer set can invalidate approval. A revoked or expired lease can never be refreshed by the agent; it must return to the proposal process.

8.3 Principal data objects

Table 9. Principal canonical objects in the reference architecture.

Object	Purpose	Minimum fields
MissionConstitution	Defines human authority and non-delegable boundaries.	Version/hash, effective time, purpose, prohibitions, roles, lanes, retention, exceptions, revocation, signatures.
KATSnapshot	Provides the versioned complete topology state used for evaluation.	System/version, topology hash, all modeled nodes/edges/planes, effective time, source, validation, and assumptions.
KATScope	Selects the target-system portion bound to inquiry and authority.	KATSnapshot ID, selected nodes/edges/planes/paths, mode, environment, criticality, and scope assumptions.
Hypothesis	Stores a speculative proposition that is empirically testable or otherwise vulnerable to declared evidence.	Campaign, KAT snapshot/scope, alternatives, predictions, disconfirming outcomes or decision criteria, origin, sensitivity, probability method, status.

Object	Purpose	Minimum fields
EvidenceObject	Carries a sanitized or released representation across an evidence gate; sanitized output is a subtype, not a competing object class.	Source commitment, collection event, transformations, removed fields, leakage tests, fidelity limits, policy labels, retention, expiry, signature.
ProposalRequest	Requests a specific transition from inquiry into authority.	Hypothesis, exact operation, canonical resources, KAT snapshot/scope, information value, risk, reversibility, output, stop conditions.
CounterfactualSet	Records the declared comparative-minimization case.	Lower-authority alternatives, expected discriminatory loss, rejection reason, reviewer disposition.
AdmissibilityBundle	Binds checked predicates and accountable attestations to the request.	Machine results or derivation, human attestations, policy/topology versions, conflicts, approvals, verifier, time, signature.
CapabilityLease	Represents temporary sender-constrained, capability-style authority.	Principal/workload/key, verb, canonical resource and resolver, audience, environment, quotas, destination, validity, uses, revocation, packet/policy/topology hashes.
ExperimentRun	Captures execution and deviations.	Proposal/lease, environment snapshot, inputs, tools, commands, outputs, telemetry, deviations, safety events, result.
DecisionRecommendation	Separates a proposed action from empirical hypothesis status.	Supporting claims, utilities, harms, policy, alternatives, required authority, accountable owner, uncertainty, disposition.
DissentEntry	Preserves material alternative interpretation.	Target, objection, evidence, materiality, owner, sensitivity, disposition, appeal, revisit trigger, expiry.
ProvenanceEvent	Creates the causal and integrity ledger.	Actor/workload, operation, object hashes, lease, trusted time, environment, predecessor, chain position.

Canonical objects should be machine-readable and versioned. Human-readable reports are deterministic views. The system should reject duplicate keys, unknown critical fields, non-canonical timestamps, unbounded strings or collections, and references to missing evidence. Hashes identify bytes, not truth. Signatures establish control of a key, not correctness of the signed claim.

8.4 Issue, redeem, and release decisions

THREE BOUNDARY DECISIONS

```
Issue(p,L) iff PacketValid(p) AND Approvals(p) AND PolicyEntails(p,L) AND BudgetReserved(p,L)
Redeem(L,op,env) iff IssuerValid(L) AND HolderProof(L) AND ExactMatch(L,op,env) AND Fresh(L) AND
                                NotRevoked(L) AND AtomicUse(L)
        Release(o,sink) iff RunBound(o) AND OutputContract(o) AND IFCAllows(o,sink) AND
                        DeclassifierAllows(o,sink) AND ProvenanceComplete(o)
```

The three decisions are intentionally separate and conjunctive. Issue checks the admissibility packet, approvals, policy, cumulative budget, and proposed lease template. Redeem checks issuer, holder proof, audience, exact operation and resource resolution, time, revocation, use count, current policy, KAT topology, and environment, then atomically consumes one-shot authority or advances a session. Release checks run binding, output contract, classification and information flow, declassification authority, provenance completeness, and destination. A fascinating hypothesis cannot compensate for missing authorization; a valid lease cannot authorize an undeclared output.

8.5 Execution and result handling

Execution environments form an authority ladder. Synthetic fixtures are preferred for contract and policy tests. A digital twin or emulator supports reproducible behavior without physical effect. Software-only Lab execution can test parsers, protocols, and update validation. Instrumented hardware can test physical interfaces behind interlocks. Live operation is exceptional and cannot rely on the agentic authorization path alone; independent safety, legal, and operational authorities remain mandatory.

The gateway should bind the run to exact inputs, tools, environment, telemetry coverage, lease, and time. Deviations are events, not free-form notes. If a command, file, endpoint, mode, or output differs materially from the proposal, execution stops or returns to review.

Result quarantine treats derived data as potentially more sensitive than input. Outputs can reveal identifiers, exploitability, internal paths, or physical effects that were not visible in the source. The release process applies schema, malware, active-content, injection-risk, leakage, provenance, and policy checks; assigns new labels; and records fidelity limits. Only then can the object enter LRRK's evidence chain.

8.6 Observability without illusion

"Evidence-grade observability" should mean attributable, integrity-protected, causally linked, independently captured, and explicit about missing coverage. It does not mean complete truth or automatic legal admissibility.

Every material event should record the actor or workload, operation, object digests, policy and topology versions, proposal, approvals, lease, trusted time, environment, output, predecessor, and audit-chain position. Lower-level sensors should verify consequential effects where application logs can lie. A report that claims an action was blocked while an unobserved actuator moved is an observability failure, not a successful policy decision.

“Independent ledger” means separate credentials and administration, enforcement-point-originated events, sequence numbers or witnessed checkpoints, and detectable gaps—not merely another log stream in the same account. The ledger itself is sensitive: timing, graph structure, identities, and low-entropy hashes can leak, so minimization, access control, and retention apply to observability data too.

8.7 Minimal verification kernel and property-specific trust

The policy checker should be a minimal verification kernel with deterministic, resource-bounded, and fuzzable interfaces. Proof size, recursion, credential-chain depth, query complexity, and verifier time must be capped to prevent proof or packet denial of service. The complete property-specific trust bases remain larger, as §6.1 makes explicit; minimizing the checker does not minimize every dependency.

RATS-style attestation can provide appraised evidence that a runtime matched declared reference values. in-toto- and SLSA-style provenance can provide signed claims about source and build steps. [12,23,36] Both remain conditional on roots of trust, appraisal policy, freshness, and provenance completeness; neither establishes semantic correctness, policy quality, or empirical truth.

9 A falsifiable research program

9.1 Architectural claims become hypotheses

The framework should be judged by tests capable of showing that it does not work. The first research program compares the complete architecture with strong baselines and then removes components to identify where any benefit comes from.

Baselines should include:

- the current manual LRRK workflow;
- static role-based access control with ordinary audit logging;
- privileged access management or just-in-time access with two-person review;
- a controlled-query clean room;
- a high-access offline Lab as a comparator whose benefit or harm is itself empirical.

The high-access Lab is not a deployment recommendation and not a presumed upper bound. Extra access may improve discrimination, or it may introduce noise, overload, leakage, and confirmation bias. The comparison asks what inquiry quality and test power change when authority is constrained, because a system can achieve zero violations by making useful research impossible.

9.2 Proposed hypotheses and disconfirmation criteria

Table 10. Candidate empirical hypotheses. Primary estimands and margins are provisional pending a variance-calibration pilot.

ID	Preregistered claim	Primary estimand	Disconfirming result
H1	Separation is noninferior to the high-access comparator for seeded-path recall under fixed analyst time.	Paired absolute recall difference; lower confidence bound above preregistered margin delta1. Distinct, grounded top-k hypotheses and time-to-useful-test are secondary.	Lower bound crosses delta1, or false-promotion risk rises beyond its separate safety margin.
H2	Counterfactual packets reduce granted authority while preserving experimental discriminatory power.	Co-primary rule: paired reduction in weighted authority-hours exceeds delta2 and absolute discriminatory-power loss stays within noninferiority margin delta3.	Either co-primary criterion fails; no tradeoff is hidden by a composite score.
H3	Semantic classification remains stable across meaning-preserving paraphrases and identity swaps.	Paired critical-class disagreement and identity-conditioned disparity with case-level confidence intervals.	Disagreement or disparity exceeds preregistered bounds, or false-denial burden makes the workflow unusable.
H4	Sanitization improves the privacy-utility frontier over a controlled-query clean room.	Paired seeded-task utility at a fixed leakage budget, or leakage at a fixed utility target; choose one primary direction prospectively.	The confidence interval fails to exclude no improvement, or any leakage exceeds the hard release budget.
H5	Independent Skeptic and dissent functions improve probabilistic reliability without unacceptable latency.	Change in calibration intercept/slope and resolution on resolved cases under an explicit co-primary rule; Brier score is secondary.	No reliability benefit, material retaliation or performative dissent, or latency exceeds its preregistered margin.

Empirical hypotheses are not security invariants. The former require estimands, uncertainty, and disconfirmation criteria. The latter are conformance obligations probed by adversarial verification and monitored continuously; a finite run with zero failures establishes at most an upper confidence bound on the tested failure rate, not proof that the invariant holds universally.

Table 11. Conformance obligations and service-level objectives. Zero observed failures is reported with an upper confidence bound, never as universal validation.

Obligation	Adversarial verification	Required evidence	Failure consequence
I1 Complete mediation	Injection, confused deputy, direct-interface bypass, sandbox escape, and broker/key compromise across every privileged path.	Complete-path inventory, enforcement-point events, independent effect telemetry, and gap checks.	Immediate stop for any unauthorized critical action or Sealed release.
I2 Lease monotonicity and use	Replay, double-spend at two gateways, delegation, child amplification, endpoint aliasing, expiry, revocation, and TOCTOU drift.	Transactional redemption, parent-child graph, canonical resolution, trusted time, and use-time revalidation.	Immediate stop for accepted stale, replayed, amplified, or mismatched authority.
I3 Provenance integrity	Missing event, source substitution, altered output, clock skew, ledger rollback, and lossy transformation.	PEP-originated sequence, witnessed checkpoints, reconstructable critical claim path, and explicit gaps.	No critical gap may be represented as complete; affected claims are quarantined or reopened.
SLO1 Temporal reassessment	Inject firmware, configuration, dependency, topology, evidence, and policy changes.	Invalidation recall, detection-to-flag latency, false invalidation, and stale-Passport exposure.	Missed material-change SLO blocks readiness; the revocation service, not Watch alone, removes authority.

The thresholds are deliberately stated before implementation and marked provisional. A pilot should calibrate task difficulty and variance, then preregister final thresholds before confirmatory evaluation. Any safety-critical boundary failure overrides aggregate performance.

9.3 Experimental design

The smallest credible study uses fictional, synthetic, licensed, or owner-authorized evidence only. It should include seeded cyber-physical cases with known ground truth, ambiguous cases with multiple valid explanations, prompt-injection artifacts, poisoned provenance, rare graph features, stale topology, policy drift, token replay, lease composition, and incomplete telemetry.

The adversarial suite should explicitly cover compromised broker or issuer key, compromised sanitizer or high-side release guard, shared-control-plane compromise, lease double-spend at two enforcement points, compute exfiltration through values/timing/errors/length, cross-agent query splitting, low-entropy commitment guessing, DNS/path/resource alias substitution, cross-campaign cache or embedding leakage, provenance-gap insertion and ledger rollback, emergency stop after expiry, execution-zone escape, and approval-key revocation between approval and redemption.

Use non-overlapping matched cases, counterbalanced order or Latin-square allocation, and investigator- and case-level random effects so skill, order, and carryover are not confounded with architecture. Normalize outputs into a common evaluation template before independent scoring; if raters can identify the arm from packet structure, they are not blinded. Freeze model and tool versions, record settings and seeds, and run repeated stochastic replicates. Safety and policy outcomes should be evaluated independently from epistemic outcomes.

A powered fractional-factorial or sequential ablation design should estimate named main effects and the interactions of greatest concern; a full six-factor design has 64 cells, while one-at-a-time ablations can miss interactions. Candidate factors are:

- structured proposal versus ordinary approval memo;
- capability lease versus conventional temporary role;
- sanitized object versus direct clean-room query;
- single reasoner versus Explorer/Skeptic/dissent configuration;
- ordinary logging versus independent provenance ledger;
- static policy check versus use-time topology and revocation revalidation.

Outcome access should be controlled for confirmatory tests. Analysis code, claim templates, exclusions, and stop rules should be timestamped. Deviations remain visible and exploratory. Repeated adversarial rounds should use held-out attacks so the system is not evaluated only against the cases used to design it.

Each confirmatory hypothesis should declare one primary estimand or an explicit co-primary success rule, absolute noninferiority or superiority margins, confidence intervals, precision or sample-size calculations, and a multiplicity strategy. Secondary outcomes remain diagnostic rather than alternative routes to a favorable result.

9.4 Metrics

Table 12. Balanced evaluation metrics. Safety, epistemic quality, privacy, and usability remain separate views.

Dimension	Primary measures	Why it matters	Failure pattern
Safety / authority	Unauthorized actions, bypasses, false allows/denies, authority-hours, reachable KAT criticality, revocation latency, rollback success.	Tests whether the architecture actually narrows causal power.	Zero activity because the system is unusable, or low rate masking one catastrophic escape.
Epistemic quality	Ground-truth-consistent top-k hypotheses on seeded cases; independently supported or dispositioned claims elsewhere; severe-test disconfirmation; reliability, resolution, agreement, heterogeneity, boundary-condition discovery, and time to useful test.	Tests whether constrained access preserves useful inquiry without rewarding hypothesis volume.	Inflated denominators, weak hypotheses manufactured for easy disconfirmation, or positive-only replication counts.
Privacy / information flow	Sensitive fields touched, linkability, reidentification, query budget, output entropy, cumulative disclosure, deletion compliance.	Tests the counterfactual-minimization and semantic-escrow claims.	Clean primary output with leakage through logs, graph shape, or repeated queries.
Human / operations	Review time, overrides, disagreement, appeals, bypass attempts, stop-the-line use, training burden, time to disposition.	Tests whether governance is meaningful and sustainable.	Rubber stamping, role collusion, or analyst migration to side channels.

Composite metrics can help compare a frontier, but raw measures must remain visible. An "authority exposure" weight embeds policy judgments about data, duration, KAT criticality, reversibility, and physical effect. Different stakeholders may choose different weights. A clean composite score can conceal the very disagreements the architecture is designed to preserve.

Probability-bearing hypotheses should be evaluated for calibration across resolved cases. Binary findings should be analyzed for false promotion, false denial, and conditional error by lane and consequence. Confidence intervals and uncertainty should accompany point estimates. Zero observed violations should be reported as zero in the tested sample with an upper confidence bound, not as proof that the true rate is zero.

9.5 Staged technical approach

Table 13. Evidence-gated maturation path.

Stage	Scope	Exit evidence	Forbidden claim
0 Contracts	Versioned schemas, state machine, canonical rendering, synthetic fixtures, and negative tests.	Deterministic validation; malformed, stale, replayed, and unauthorized transitions fail closed.	No claim of containment or scientific benefit.
1 Policy sandbox	No-egress Kestrel Analysis Workspace, policy kernel, packet builder, broker, and independent ledger.	Complete-path inventory, verifier resource bounds, lease attenuation, revocation, and bypass testing.	No production boundary claim.
2 Shadow replay	Historical or synthetic Campaigns with no operational effect.	Usability, authority reduction, hypothesis quality, provenance reconstruction, and ablation results.	No causal claim from retrospective cases alone.

Stage	Scope	Exit evidence	Forbidden claim
3 Twin / Lab	Digital twins and instrumented owner-authorized hardware with interlocks.	Adversarial escape tests, output quarantine, physical telemetry, and immediate-stop drills.	No generalization to field operation.
4 Green / Amber pilot	Prospective read-only or bounded sensitive-compute workflows.	Independent review, SLOs, privacy budget, reviewer performance, and incident process.	No Red/live readiness claim.
5 Exceptional Red	Narrow state-changing Lab use after prior gates pass.	Separate safety/legal authorization, exact-use leases, field-validity limits, and external assessment.	No autonomous live-control or certification claim.

Stage gates should be explicit. The project does not advance because a schedule says so. It advances when the current stage produces the declared evidence, independent review accepts the limits, and no stop criterion remains open.

9.6 Immediate stop criteria

Any of the following should stop the relevant pilot and invalidate a readiness claim:

- unauthorized physical state change;
- raw Sealed disclosure or successful reidentification beyond the declared budget;
- a bypass around lease mediation;
- acceptance of an expired, revoked, replayed, amplified, or environment-swapped lease;
- an unlogged privileged action reported as fully observed;
- silent provenance loss or source substitution on a critical claim path;
- generation of a finding, Passport event, or Assurance State from an unvalidated chain;
- loss of required safety interlock or telemetry;
- evidence that reviewers cannot distinguish requested authority from model confidence;
- routine analyst bypass because the process is operationally unusable.

These criteria protect the research program from moving its own goalposts. A stop is a result. It should generate a Security Story and a design revision, not be hidden as an implementation detail.

10 Pressure tests, limitations, and nonclaims

10.1 Semantic smuggling

Evidence can contain instructions that a model treats as commands. Metadata, filenames, provenance fields, OCR, code comments, and sanitized summaries can all carry injection content. The architecture must type source material as untrusted data, render it inert where possible, strip active links and executable formats, and give the reasoner no ambient authority. This reduces impact; it does not make semantic attacks impossible.

10.2 Proposal laundering

A harmful objective can be described in benign language. A semantic classifier may miss euphemism, cultural context, or a multi-step plan. Structural checks and model review can route escalation, but the hard boundary must remain exact authority: the resource, verb, environment, output, duration, and reachable effect. Purpose matters and is not sufficient.

10.3 Proof gaming

Proposal authors may optimize forms rather than minimize authority. Expected information gain may become ritual. Lower-risk alternatives may be dismissed with boilerplate. Random audit, counterfactual challenge, reviewer rotation, outcome feedback, and measurement of granted versus used authority are required. The admissibility packet creates inspectable behavior; it does not eliminate incentives.

10.4 Capability composition

Several safe leases can combine into an unsafe inference or action. The system must track cumulative authority and the whole execution graph, not only individual calls. Policy should consider dependency between leases, prior outputs, privacy budget, and downstream destinations. A per-request reference monitor without composition analysis can be locally correct and globally unsafe.

10.5 Graph reidentification

Rare components, paths, firmware combinations, locations, or timestamps can identify a product despite pseudonyms. Campaign-scoped identifiers, subgraph views, generalization, cohort thresholds, query budgets, and restricted re-linking reduce risk. They do not create anonymity. The residual linkability estimate belongs on the evidence object.

10.6 TOCTOU and environment drift

Policy, topology, evidence, endpoint identity, firmware, reviewer authority, or lab wiring can change between proposal verification and execution. Leases must bind to the relevant snapshots and be revalidated at use time. Atomic redemption narrows replay. Independent interlocks protect physical safety when software state is stale or wrong.

10.7 Human oversight failure

Human approval can be meaningless if the reviewer lacks context, sees only a confidence score, is fatigued, colludes with the requester, or approves a general agent rather than an exact action. Separation of duty requires distinct principals, credentials, powers, and accountability - not five prompts sent to the same model.

High-consequence reviewers should see the evidence delta, requested authority, reachable KAT path, output destination, residual risk, rollback, stop conditions, and prior dissent. Their signature binds to the canonical packet digest. Material change requires reapproval.

10.8 Sentinel capture and governance abuse

A Harm Sentinel can become a censorship or political-control mechanism. Mission constitutions and prohibitions must be explicit, versioned, reviewable, and subject to appeal. Symmetry tests should examine whether equivalent requests receive different classifications based on identity or phrasing. NIST's Govern/Map/Measure/Manage functions and control families provide useful vocabulary for assigning ownership and monitoring risk, but they do not substitute for system-specific authority boundaries. Governance itself is part of the threat model. [3,4,21]

10.9 Model memory and side channels

Sanitizing an output does not erase restricted context from a persistent model, cache, trace, embedding, or operator screen. Restricted work should use ephemeral isolated sessions where feasible, minimize retained prompts, and treat logs and derived representations as possible cross-domain channels. Timing, counts, errors, graph shape, and refusal behavior can encode information.

10.10 Incomplete KAT and incomplete observation

A perfectly enforced lease can still be unsafe if the KAT omits a dependency, mode, or Connect edge. Application logs can report a block while a lower-level effect occurs. Topology completeness and observability coverage are empirical claims. The correct state is Unknown or Partially Assessed when evidence is missing.

10.11 Field validity

Synthetic and Lab results do not automatically generalize to nondeterministic physical environments. Temperature, RF conditions, wear, operator behavior, timing, supply variation, and external services can change outcomes. Live validation may be unsafe or unauthorized. The Passport should state the evidence environment and never present simulation as field proof.

10.12 Explicit nonclaims

- Pseudonymization is not anonymity.
- Provenance is not authenticity or truth.
- Attestation is not semantic correctness.
- Policy compliance is not safety or legal compliance.
- Human validation is not infallibility.
- Observability is not completeness.
- A lane is not a complete risk model.
- A verifier decision is not an assurance certificate.
- A capability lease does not verify model intent.
- A time-bounded Assurance State is not permanent product certification.
- The airlock does not replace IAM, PAM, network segmentation, clean rooms, secure execution, safety interlocks, or legal governance.
- The framework does not establish first-of-kind novelty without a broader prior-art review.

These nonclaims are not caveats appended to the method. They are part of the method. LRRK's value depends on refusing to let a precise artifact carry a broader conclusion than its evidence allows.

11 Conclusion: freedom to ask, discipline to cross

Semantic Airlock with Proof-Carrying Curiosity began as a response to a narrow boundary problem: how to investigate sensitive, adversarial evidence without placing collection capability inside the same reasoning system. It matured when the deeper problem became visible. Separation alone does not govern expansion, and justification alone does not enforce separation. The architecture and the inquiry discipline have to be designed together.

The combined phrase now carries a precise proposition. Semantic means that purpose, context, evidence meaning, topology, and downstream consequence are represented explicitly. Airlock means there is no direct passage: evidence, proposals, and results cross through separate staged gates. Proof means stated obligations can be checked relative to a policy and trusted state, not that empirical truth is final. Carrying means the justification remains cryptographically and causally bound to the request, lease, execution, and

result. Curiosity means hypotheses can remain plural, adversarial, and revisable without acquiring ambient authority.

The fit with LRRK is structural. KAT traces how digital state acquires physical authority in systems that Sense, Move, and Act. The Semantic Airlock traces how candidate meaning acquires research authority in the system that studies them. Campaigns create questions. Kestrel explores approved evidence. The airlock governs transitions. Lab executes bounded tests. The existing evidence chain earns assurance. Passport, Registry, Watch, and Security Story preserve what was known, why it was known, when it was valid, and what would make it uncertain again.

The scientific method is not decorative language in this architecture. It determines the object model. Questions are not claims. Hypotheses are not observations. Predictions are not evidence. A valid proposal is not a successful experiment. A successful experiment is not a finding. A finding is not a durable Assurance State until it is explicitly validated. Each distinction prevents a semantic transition from acquiring more authority than it has earned.

The proposal should now be attacked experimentally. Does it preserve useful curiosity? Does counterfactual minimization reduce authority without destroying discriminatory power? Do leases contain a compromised explorer? Does sanitization create a better privacy-utility frontier than a simpler clean room? Do Skeptic and dissent functions improve calibration? Does independent provenance prevent unsupported claims? Does Watch keep time-bounded state honest? If the answer is no, the architecture should narrow, change, or be rejected.

That is the final alignment with LRRK's mission. The goal is not to make an intelligent system timid. It is to make every transition from idea to access, from access to experiment, and from experiment to assurance explicit enough to challenge, constrain, observe, reproduce, and revoke.

CLOSING PROPOSITION

Hypotheses should be abundant. Authority should be scarce, leased, and attributable. Assurance should still be earned.

Appendix A Minimum viable contracts

The first prototype should prefer deterministic contracts and synthetic fixtures over predictive sophistication. The contracts below define the minimum separation needed to test the architectural hypothesis.

Table A1. Minimum viable contracts for an offline-first prototype.

Contract	Required behavior	Minimum negative tests
MissionConstitution	Signed, versioned purpose, prohibitions, roles, lanes, retention, expiry, and revocation.	Unsigned, expired, unknown role, prohibited purpose, and agent-authored exception.
EvidenceObject	Canonical digest, source commitment, transformation list, labels, fidelity limits, and allowed purposes.	Digest mismatch, missing transform, label downgrade, unsupported purpose, and prompt-bearing metadata.
ProposalRequest	Plural hypotheses, predictions, KAT snapshot/scope, evidence links, exact requested transition, counterfactual set, and stop rules.	Single evidence-insulated claim, missing other branch, stale KAT, no discriminator, and unbounded request.
AdmissibilityBundle	Machine results or derivation, accountable attestations, policy/topology versions, approvals, conflicts, verifier identity, and signature.	Detached approval, stale facts, missing attestor, mismatched lease template, unknown verifier, and altered digest.
PolicyDecision	Deterministic Allow, Deny, or Escalate with policy and fact digests.	Unknown defaults to allow, ambiguous rule, stale policy, missing reviewer, and resource exhaustion.
CapabilityLease	Signed exact authority, short TTL, runtime/audience binding, maximum uses, output class, and revocation.	Replay, amplification, endpoint swap, expiry, delegation, revoked use, and TOCTOU mismatch.
ExperimentRun	Fixed inputs, environment snapshot, commands, outputs, telemetry coverage, deviation log, and safety result.	Undeclared tool, changed input, lost telemetry, extra output, clock skew, and partial cleanup.
ResultRelease	Quarantine, reclassification, schema/injection/leakage checks, provenance binding, and release authority.	Raw leakage, output shape change, active content, missing lineage, and unauthorized destination.
AssuranceIntake	Released evidence can create observations but no automatic finding or Assurance State.	Hypothesis promoted as evidence, verifier decision promoted as assurance, and missing human disposition.

Every object should include a schema version, canonical serialization rule, stable identifier, creation time from a trusted source, content digest, producer identity, and explicit classification. Derived human-readable views must not add claims absent from canonical data.

The one-command validation target for a prototype should cover schema checks, fixed-clock determinism, policy transitions, negative cases, lease replay and expiry, no-egress boundaries, provenance continuity, sanitization labels, forbidden claim scans, result quarantine, and deterministic Passport rendering. A passing local test is evidence about those entry points, not proof of a production sandbox.

Appendix B Canonical glossary

Table B1. Canonical terminology for the proposed methodology.

Term	Definition
Admissibility	The condition that a proposed transition satisfies all current machine-verifiable policy predicates and required accountable attestations.
Airlock gate	A staged boundary for restricted evidence release, experiment authorization, or result release.
Assurance State	A current, scoped, human-dispositioned, and time-bounded conclusion associated with a capability, path, boundary, or claim.
Capability lease	A signed, narrow, short-lived, sender-constrained, capability-style authorization accepted only for an exact audience, operation, resource, environment, and use budget.
Counterfactual minimization	The requirement to compare lower-authority observations or actions and justify why each is insufficient for the declared discriminator.
Dissent ledger	A controlled record of material alternative hypotheses, objections, evidence, disposition, and revisit triggers.
Epistemic freedom	The permitted breadth of hypothesis generation, comparison, challenge, and revision over approved evidence.
EvidenceObject	A versioned, typed representation crossing an airlock with provenance, transformations, policy labels, uncertainty, fidelity limits, supersession, and retention behavior.
Evidence-grade observability	Attributable, integrity-protected, causally linked, independently captured event evidence with explicit coverage gaps.
Experiment	An authorized procedure designed to produce observations that discriminate among live hypotheses under declared conditions.
Hypothesis	A defeasible descriptive, predictive, or causal proposition that has not acquired evidentiary or operational authority merely by being stated.
KATSnapshot	The versioned complete KAT topology state against which a proposal or execution is evaluated.
KATScope	The selected nodes, edges, planes, modes, paths, and environment bound to a KATSnapshot and a proposal.
Machine-verified predicate	A decidable result for canonical form, integrity, freshness, policy, scope, budget, approvals, or lease conditions relative to trusted state; a formal proof is claimed only when a derivation is checked.
Mission Constitution	The signed, versioned human authority defining purpose, prohibitions, roles, non-delegable rules, retention, exceptions, and revocation.
Operational freedom	The permitted ability to access restricted information, invoke tools, publish, modify state, or reach physical capability.
AdmissibilityBundle	The canonical collection of machine results or derivation, accountable attestations, conflicts, versions, approvals, and signatures bound to one proposal and lease template.
Proof-Carrying Curiosity	The discipline requiring each consequential transition sought by an inquiry to carry checked bindings, current policy results, accountable attestations, declared empirical criteria, limits, and expiry.
Proposal	A typed request linking a hypothesis and evidence gap to one exact data, compute, publication, or action transition.
Result quarantine	The trust domain that validates, reclassifies, and binds experiment output before it can become a released EvidenceObject.
Semantic Airlock	A staged, policy-mediated boundary separating broad non-authoritative inquiry from restricted evidence and consequential authority.
Severe test	A test with a strong capability to reveal a specified relevant error if that error exists under declared conditions.
Trusted computing base	The complete set of mechanisms and operators on which a stated property depends; raw confidentiality, authorization, evidence integrity, and physical safety have different trust bases.
Watch trigger	A material topology, firmware, dependency, policy, evidence, or environment change that requests reassessment and, when policy requires, revocation by the authorization service.

References

[1] Laboratory for Risk Resilient Kinematics. **Kinematic Assurance Topology: A methodology for security assurance where software has physical consequence.** Version 1.0, August 2026.

[2] Griffor, E. R., Greer, C., Wollman, D. A., and Burns, M. J. **Framework for Cyber-Physical Systems: Volume 1, Overview.** NIST SP 1500-201, 2017. NIST publication.

[3] Tabassi, E. **Artificial Intelligence Risk Management Framework (AI RMF 1.0).** NIST AI 100-1, 2023. NIST publication.

[4] National Institute of Standards and Technology. **Artificial Intelligence Risk Management Framework: Generative Artificial Intelligence Profile.** NIST AI 600-1, 2024. NIST publication.

[5] Saltzer, J. H., and Schroeder, M. D. **The Protection of Information in Computer Systems.** Proceedings of the IEEE 63(9), 1975. IEEE record.

[6] Necula, G. C. **Proof-Carrying Code.** POPL 1997. ACM DOI.

[7] Appel, A. W., and Felten, E. W. **Proof-Carrying Authentication.** ACM CCS 1999. ACM DOI.

[8] Anderson, J. P. **Computer Security Technology Planning Study.** ESD-TR-73-51, 1972. NIST historical copy.

[9] Birgisson, A., Politz, J. G., Erlingsson, U., Taly, A., Vrable, M., and Lentczner, M. **Macaroons: Cookies with Contextual Caveats for Decentralized Authorization in the Cloud.** NDSS 2014. Google Research.

[10] Myers, A. C., and Liskov, B. **A Decentralized Model for Information Flow Control.** SOSP 1997. Author copy.

[11] W3C. **PROV-O: The PROV Ontology.** W3C Recommendation, 2013. W3C Recommendation.

[12] Torres-Arias, S., Afzali, H., Kuppusamy, T. K., Curtmola, R., and Cappos, J. **in-toto: Providing farm-to-table guarantees for bits and bytes.** USENIX Security 2019. USENIX paper.

[13] Chamberlin, T. C. **The Method of Multiple Working Hypotheses.** Science 15(366), 1890. Science DOI.

[14] Popper, K. **The Logic of Scientific Discovery.** Routledge edition. Publisher record.

[15] National Academies of Sciences, Engineering, and Medicine. **Reproducibility and Replicability in Science.** National Academies Press, 2019. Report.

[16] Nosek, B. A., Ebersole, C. R., DeHaven, A. C., and Mellor, D. T. **The preregistration revolution.** PNAS 115(11), 2018. PNAS DOI.

[17] Neyman, J., and Pearson, E. S. **On the Problem of the Most Efficient Tests of Statistical Hypotheses.** Philosophical Transactions of the Royal Society A 231, 1933. Royal Society DOI.

[18] Mayo, D. G. **Statistical Inference as Severe Testing: How to Get Beyond the Statistics Wars.** Cambridge University Press, 2018. Publisher record.

[19] NIST and SEMATECH. **e-Handbook of Statistical Methods.** Official handbook.

[20] Brier, G. W. **Verification of Forecasts Expressed in Terms of Probability.** Monthly Weather Review 78(1), 1950. AMS record.

[21] Joint Task Force. **Security and Privacy Controls for Information Systems and Organizations.** NIST SP 800-53 Revision 5, 2020. NIST publication.

[22] Rose, S., Borchert, O., Mitchell, S., and Connelly, S. **Zero Trust Architecture.** NIST SP 800-207, 2020. NIST publication.

[23] IETF. **Remote ATtestation procedureS (RATS) Architecture.** RFC 9334, 2023. RFC Editor.

[24] National Institute of Standards and Technology. **NIST Privacy Framework 1.0 Core.** 2020. NIST Privacy Framework.

[25] U.S. Food and Drug Administration. **Adaptive Designs for Clinical Trials of Drugs and Biologics: Guidance for Industry.** 2019. FDA guidance.

[26] Lindley, D. V. **On a Measure of the Information Provided by an Experiment.** Annals of Mathematical Statistics 27(4), 1956. Project Euclid.

[27] Box, G. E. P. **Science and Statistics.** Journal of the American Statistical Association 71(356), 1976. DOI.

[28] Tukey, J. W. **We Need Both Exploratory and Confirmatory.** The American Statistician 34(1), 1980. DOI.

[29] Platt, J. R. **Strong Inference.** Science 146(3642), 1964. Science DOI.

[30] Hanson, N. R. **Patterns of Discovery: An Inquiry into the Conceptual Foundations of Science.** Cambridge University Press, 1958. Cambridge front matter.

[31] Quine, W. V. O. **Two Dogmas of Empiricism.** The Philosophical Review 60(1), 1951. JSTOR record.

[32] Gelman, A., and Shalizi, C. R. **Philosophy and the Practice of Bayesian Statistics.** British Journal of Mathematical and Statistical Psychology 66(1), 2013. Wiley DOI.

[33] Murphy, A. H. **A New Vector Partition of the Probability Score.** Journal of Applied Meteorology 12(4), 1973. AMS record.

[34] Bauer, L., Schneider, M. A., and Felten, E. W. **A Proof-Carrying Authorization System.** Princeton University Technical Report TR-638-01, 2001. Author copy.

[35] Miller, M. S., Yee, K.-P., and Shapiro, J. **Capability Myths Demolished.** Johns Hopkins University Systems Research Laboratory Technical Report SRL2003-02, 2003. Author copy.

[36] OpenSSF. **Supply-chain Levels for Software Artifacts (SLSA) Specification, version 1.2.** Official specification.