Quanta AI Trigger 2.1.4

Governed Sensor Observation, Candidate Formation, Evidence, and Durability

Architecture, Deterministic Replay, Quorum Mechanics, Human Authorization, Validation, and Research Boundaries

AN INSTITUTIONAL EXPOSITORY NOTE

Prepared for executive review, AI-systems research, safety assurance, and engineering stakeholders

Field	Record
Document ID	QAI-TRIGGER-214-EXPO-2026-07-26
Repository	Quanta Repository 2.1.4
Trigger engine	QTrigger 1.1.0
Observation modules	Sensor Input API 2.0.0 | Physical Input Fusion 1.0.0 | Motion Input Fusion 1.0.0
Model-input boundary	Linear Input Parser 1.0.0 | model execution disabled
Governance	SOPHIA - CHARLOTTE - LANDON - Professor - Podium
Classification	Research draft / development integration
Operational representation	Observation-only; production activation not authorized

Date: 2026-07-26

OBSERVATION-ONLY TRIGGER | NO MODEL EXECUTION | NO AUTOMATIC LAYER-1 ADMISSION | NO PHYSICAL ACTUATION

Executive summary

Institutional boundary

Quanta 2.1.4 is a governed, deterministic trigger and evidence system. It converts validated replay or synthetic, sensor-derived observations into either NO_TRIGGER or an evidence-backed CANDIDATE. It does not independently authorize action, execute AI models, bind live hardware, retain raw sensor frames, or admit a result to Layer 1.

The system joins a 546-laboratory registry, D1-D4 ordered state mechanics, two weighted Byzantine quorum layers, robust alarm qualification, independence and gain gates, deterministic evidence generation, and a role-separated governance chain. Professor may propose a bounded action; Podium requires a named operator and explicit confirmation; the launcher re-verifies scope and can dispatch only the reversible local demo-echo target.

Version 2.1.4 corrects the durability placement inherited from earlier releases. The default relaxed mode reduces the full governed chain from 17.68 ms to 4.61 ms, while strict mode fsyncs both the authoritative evidence ledger and its materialized state projection at 31.19 ms. Record contents remain unchanged between modes; only the power-loss guarantee differs.

Finding	Institutional interpretation
546 selected laboratories	165 cascade-alarm rows, 160 bond-pricing 4D+ rows, 311 population/housing rows; categories overlap; 546 unique IDs.
12 active / 534 registry-only adapters	The complete registry is visible, while only a bounded reference subset participates in executable replay.
125 tests passed, 0 failed	Portable integration, evidence, replay, streaming, fusion, parser, governance, and durability behaviors are exercised.
4.61 ms relaxed chain	Approximately 217 governed decisions per second on the recorded reference benchmark.
31.19 ms strict chain	Both store and evidence ledger are fsynced; use when power-loss durability is mandatory.
Production activation denied	Live hardware, model execution, automatic Layer-1 admission, physical control, and production signing remain outside authorization.

1. Purpose, scope, and source basis

This note develops the QTrigger rough laboratory report and the Quanta 2.1.4 repository into an institutional exposition. The attached JA21 PORTAL Cypher Engine note is used as a publication-format precedent only: title-page hierarchy, claim boundaries, figure-led explanation, evidence classification, and role-separated governance. The technical claims in this note come from the Quanta repository, its source modules, policies, tests, and validation records.

The term AI trigger is used as the project identity. In the present implementation, the trigger is a deterministic governed decision engine rather than a learned model. The linear-input subsystem produces vectors for an external model boundary but does not execute models, learn at runtime, or actuate a physical system.

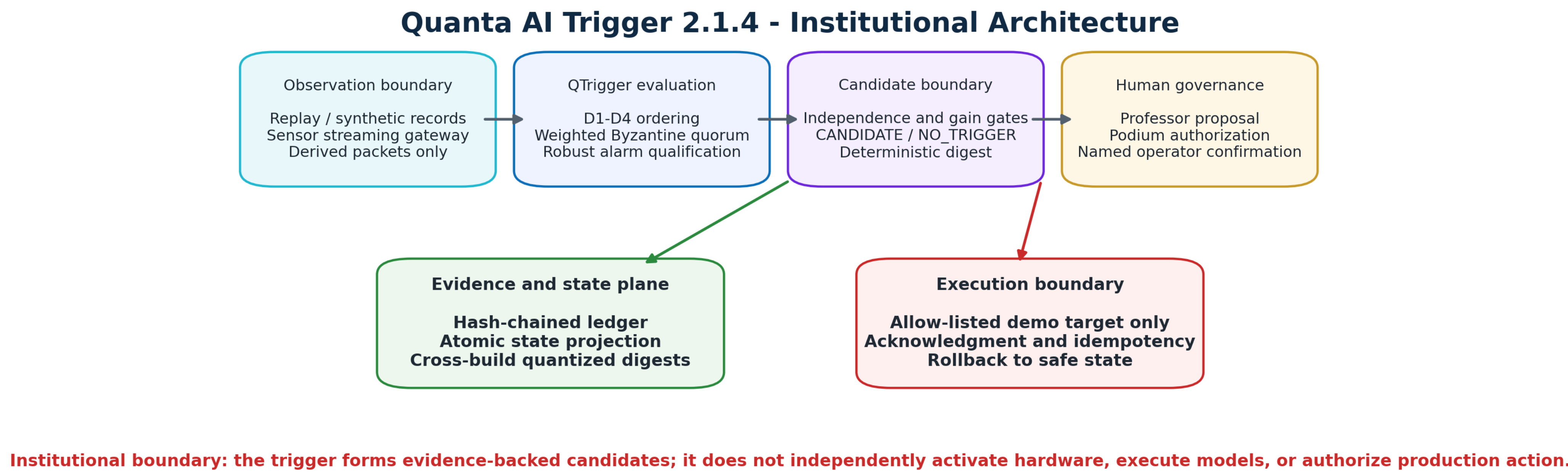


Figure 1. Layered architecture separates observation, trigger evaluation, candidate formation, human governance, evidence, and bounded execution.

Source discipline: repository records support portable-development claims only. Live hardware, production signing, Windows hardware qualification, and automatic Layer-1 activation remain NOT RUN, NOT IMPLEMENTED, BLOCKED, or NOT AUTHORIZED.

2. System identity and operational boundary

Quanta 2.1.4 describes itself as a governed linear model-input boundary and an offline-capable observation gateway. Its executable trigger accepts only replay or synthetic packages, verifies canonical payload digests and calibration metadata, rejects raw camera or pixel fields, and prevents Layer 3 from directly consuming raw Layer-1 features.

The result of evaluation is not an autonomous command. It is a canonical evidence object containing normalized state, quorum records, alarm qualification, gate outcomes, explicit blocks, and a deterministic digest. A CANDIDATE may proceed only into the Professor/Podium governance path. The operational launcher is independently restricted to demo-echo, a reversible local acknowledgment target.

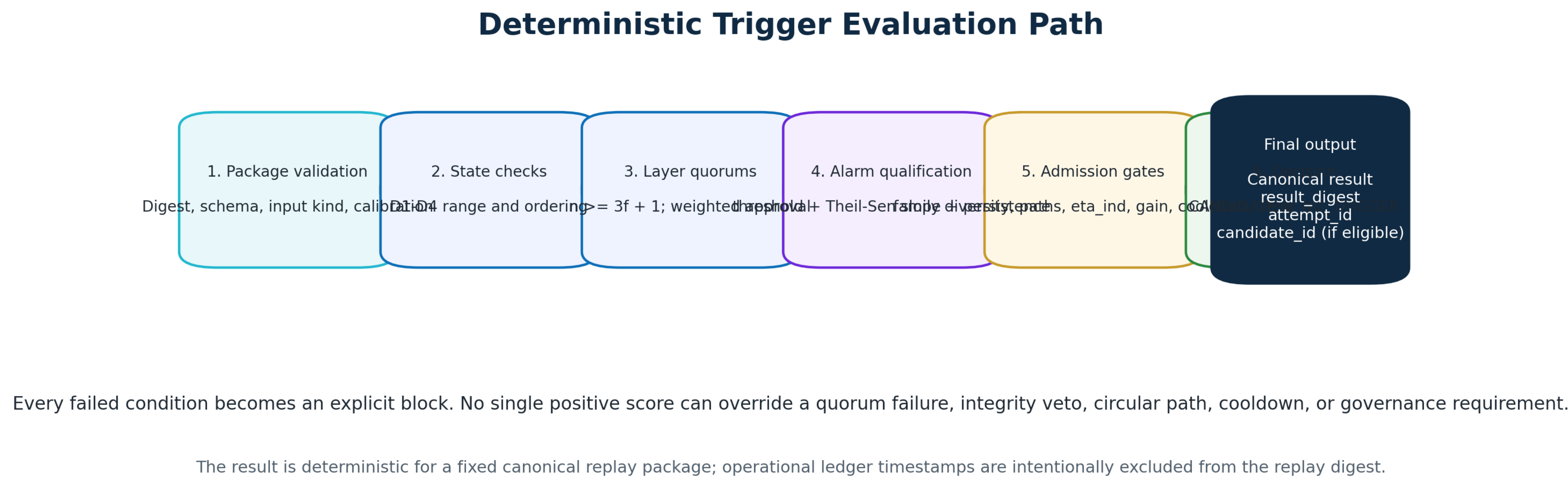


Figure 2. Deterministic trigger evaluation path from validated package to evidence-backed decision.

Capability	2.1.4 status
Replay and synthetic evaluation	SUPPORTED
Live camera / IMU acquisition	DISABLED / NOT QUALIFIED
Raw observation retention	NONE
Linear vector generation	SUPPORTED from derived packets
Model execution or learning	DISABLED
Automatic Layer-1 admission	FALSE / BLOCKED SAFE DEFAULT
Physical actuation	DENIED
External target launch	DENIED

3. Mathematical lineage and finite trigger realization

8S Penteract–S^3 Master Coupled Mechanics Law

$$
\begin{aligned}
&C_5 = [\,C_4 \mid e\,], \qquad \eta_{\text{ind}} = \frac{\left\|(I - C_4C_4^{+})e\right\|_2^2}{\|e\|_2^2 + \varepsilon} \geq \varepsilon_{\text{ind}},\\
&c_i = \ell n_i, \qquad n_i \in \mathbb{Z}^5, \qquad \|n_i\|_1 \leq M, \qquad c_i \in [-1,1]^5,\\
&q(\tau) = \sin^2\left(\frac{\pi\tau}{T}\right),\\
&W_{ij} = A_{ij}\exp\left[-\frac{(c_i - c_j)^{\top}G_5(c_i - c_j)}{2\sigma_c^2} - \frac{d_J(i,j)^2}{2\sigma_J^2}\right],\\
&\dot{\theta}_i = \omega_i + K_2\sum_j W_{ij}\sin(\theta_j - \theta_i) + K_3\sum_{j,k} H_{ijk}\sin(\theta_j + \theta_k - 2\theta_i),\\
&h_i = \beta\left(Mq + \frac{1}{2} - d_i\right) + a_E E_i + a_u u_i - a_\chi \chi_i - a_\zeta \zeta_i\\
&\qquad\quad + J_2\sum_j W_{ij}(2p_j - 1) + J_3\sum_{j,k} H_{ijk}(2p_j - 1)(2p_k - 1),\\
&p_i = \sigma(h_i) = \frac{1}{1 + e^{-h_i}},\\
&B_5(c_i) = \prod_{a=4}^{8} b(c_{i,a}),\\
&s_i = p_i\kappa_i(1 - \chi_i)(1 - \zeta_i)v(\omega_i),\\
&r_i = r_{\max}B_5(c_i)^{\alpha}s_i^{1/3},\\
&F_i = S_{r_i}^3 = \left\{y \in \mathbb{R}^4 : \|y\|_2 = r_i\right\},\\
&X_8(\tau) = \left\{(c,y) \in [-1,1]^5 \times \mathbb{R}^4 : \|y\|_2 = R(c,\tau)\right\},\\
&z_i(u) = \begin{bmatrix} c_i \\ r_i u \end{bmatrix}, \qquad u \in S^3,\\
&x_i^{(3)} = \left[\Pi_5(\tau) \quad \Psi_4(\tau)\right] z_i(u),\\
&g_{ij}^{(5)} = \sqrt{(c_i - c_j)^{\top}G_5(c_i - c_j)} - (\lambda_i + \lambda_j),\\
&\delta_{ij}^{(8)} = \sqrt{(c_i - c_j)^{\top}G_5(c_i - c_j) + (r_i - r_j)^2},\\
&g_{ij}^{(3)} = \|\Pi_5 c_i - \Pi_5 c_j\|_2 - (\hat{r}_i + \hat{r}_j),\\
&g_{ij}^{(\mathrm{J})} = d_{\mathrm{J}}(\mathfrak{J}_i, \mathfrak{J}_j) - \theta_{\mathrm{J}},\\
&\mathscr{R}_{ij}^{8S} = \mathrm{Classify}\left(g_{ij}^{(5)}, \delta_{ij}^{(8)}, g_{ij}^{(3)}, g_{ij}^{(\mathrm{J})}, K_{ij}, \Sigma_{ij}\right),\\
&\Delta_{8S} = \mathrm{Score}(\mathcal{M}_8) - \mathrm{Score}(\mathcal{M}_7),\\
&\boxed{\mathrm{Admit}(8S) \iff \eta_{\text{ind}} \geq \varepsilon_{\text{ind}} \ \wedge\ \Delta_{8S} > \varepsilon_{\text{gain}}.}
\end{aligned}
$$

Figure 3. 8S Penteract-S3 Master Coupled Mechanics Law preserved in the QTrigger source package.

Runtime abstraction

The runtime does not numerically solve the complete continuous master law. Instead, it realizes a bounded governance-oriented subset as deterministic state ordering, quorum, alarm, independence, gain, integrity, and admission gates.

The key state terms are activation candidate, activation complete, populate intent, populate execution, and recursive propagation. The eta_ind and positive-gain conditions appear directly in final admission. Alarm history is evaluated by score, robust slope, and persistence. The final outcome is classified as CANDIDATE or NO_TRIGGER.

This finite realization should be described as an implementation mapping inspired by the master mechanics, not as proof that every continuous geometric term has been solved or physically instantiated.

Institutional claim boundary: the equation suite supplies conceptual and admission structure; the repository supplies the executable, testable trigger semantics.

4. D1-D4 ordered state mechanics

The trigger normalizes five state values to six decimal places and requires each value to lie in [0,1]. Ordering constraints prevent later-stage completion or propagation from exceeding the state that logically precedes it. The eligible-candidate fixture uses activation candidate 0.90, activation complete 0.80, populate intent 0.85, populate execution 0.80, and recursive propagation 0.70.

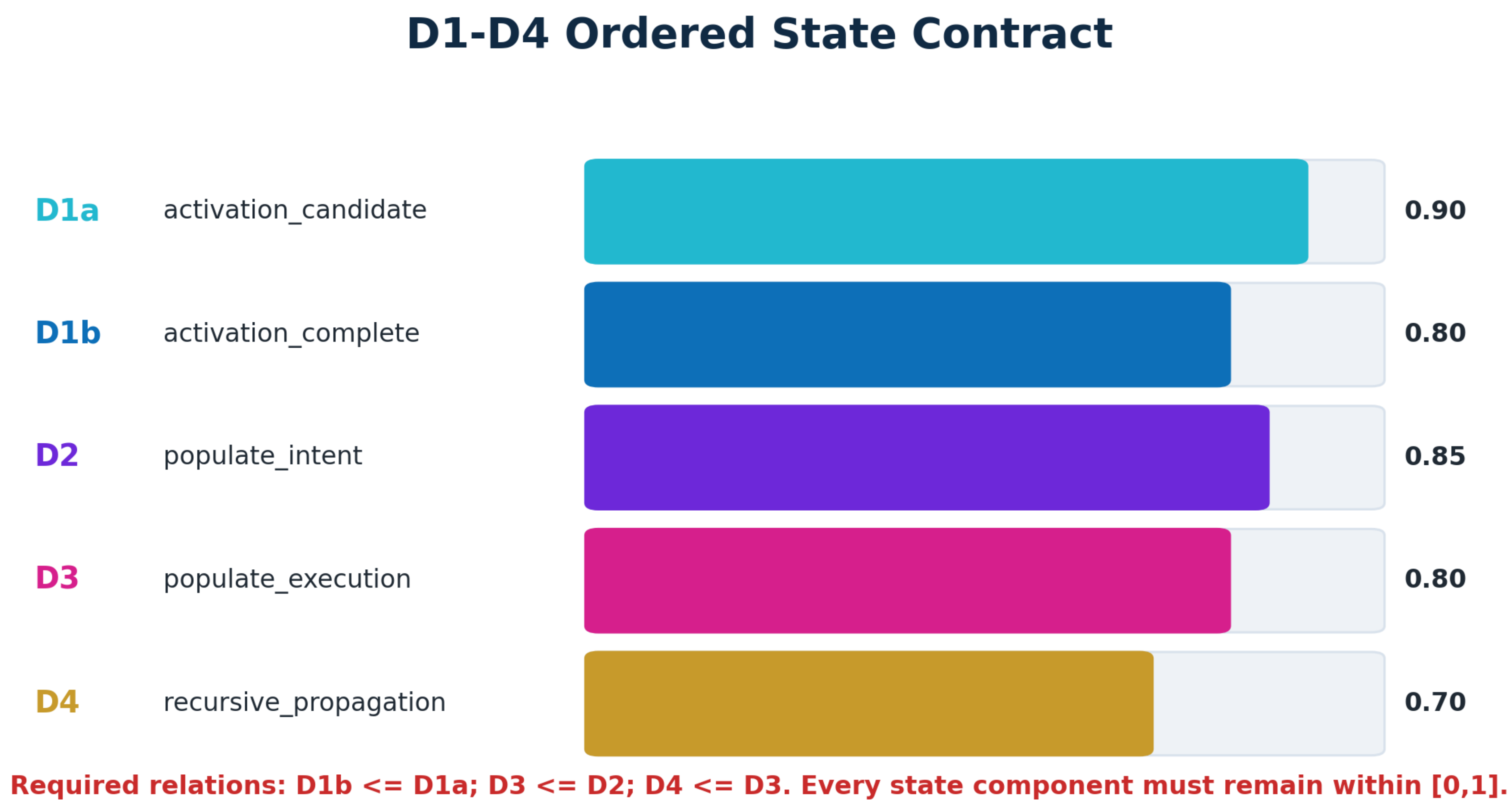


Figure 4. Ordered state values from the eligible trigger fixture and the mandatory inequalities.

Constraint	Purpose	Failure record
activation_complete <= activation_candidate	Completion cannot exceed candidate activation	D1 ordering violated
populate_execution <= populate_intent	Execution cannot exceed declared intent	D2/D3 ordering violated
recursive_propagation <= populate_execution	Propagation cannot exceed executed population	D3/D4 ordering violated
0 <= every component <= 1	Bounded normalized state	component out of range

5. Layered quorum mechanics

Layer 1 and Layer 2 are evaluated independently. Only votes marked healthy, fresh, and eligible participate. With configured Byzantine tolerance f = 1, each layer requires at least n >= 3f + 1 = 4 eligible members. Weighted approval must reach 0.67, and any eligible integrity veto forces failure.

The byzantine-minority fixture demonstrates the intended distinction: three equal-weight approvals and one reject produce a 0.75 approval share and remain above threshold. The integrity-veto fixture demonstrates the stronger safety rule: numerical approval does not override an integrity defect.

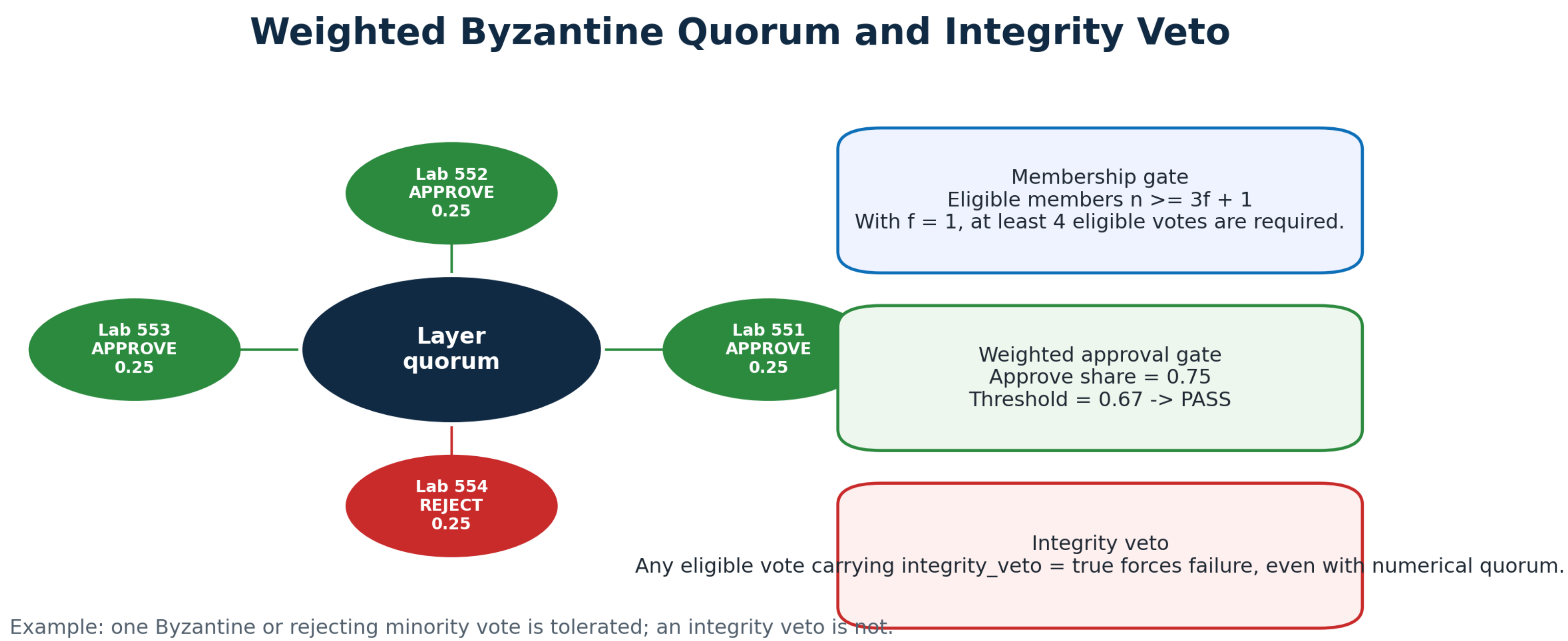


Figure 5. Weighted Byzantine quorum tolerates a minority reject but not an integrity veto.

A quorum is necessary but never sufficient. Both layers, alarm diversity, independent paths, eta_ind, positive gain, cooldown, and final integrity must also pass.

6. Alarm qualification: level, robust slope, and persistence

Each alarm carries a history window, threshold, slope threshold, persistence requirement, laboratory ID, and family. The current score must exceed threshold; the Theil-Sen median of all pairwise slopes must exceed the slope threshold; and the trailing run above threshold must meet persistence. Only then does an alarm qualify and contribute its family to the diversity gate.

This structure rejects several common false-positive patterns: a high but nearly flat series, a late threshold crossing without persistence, a transient spike, and an overall low baseline. The repository includes dedicated fixtures for each failure mode and a test showing that the robust slope resists a single-point spike.

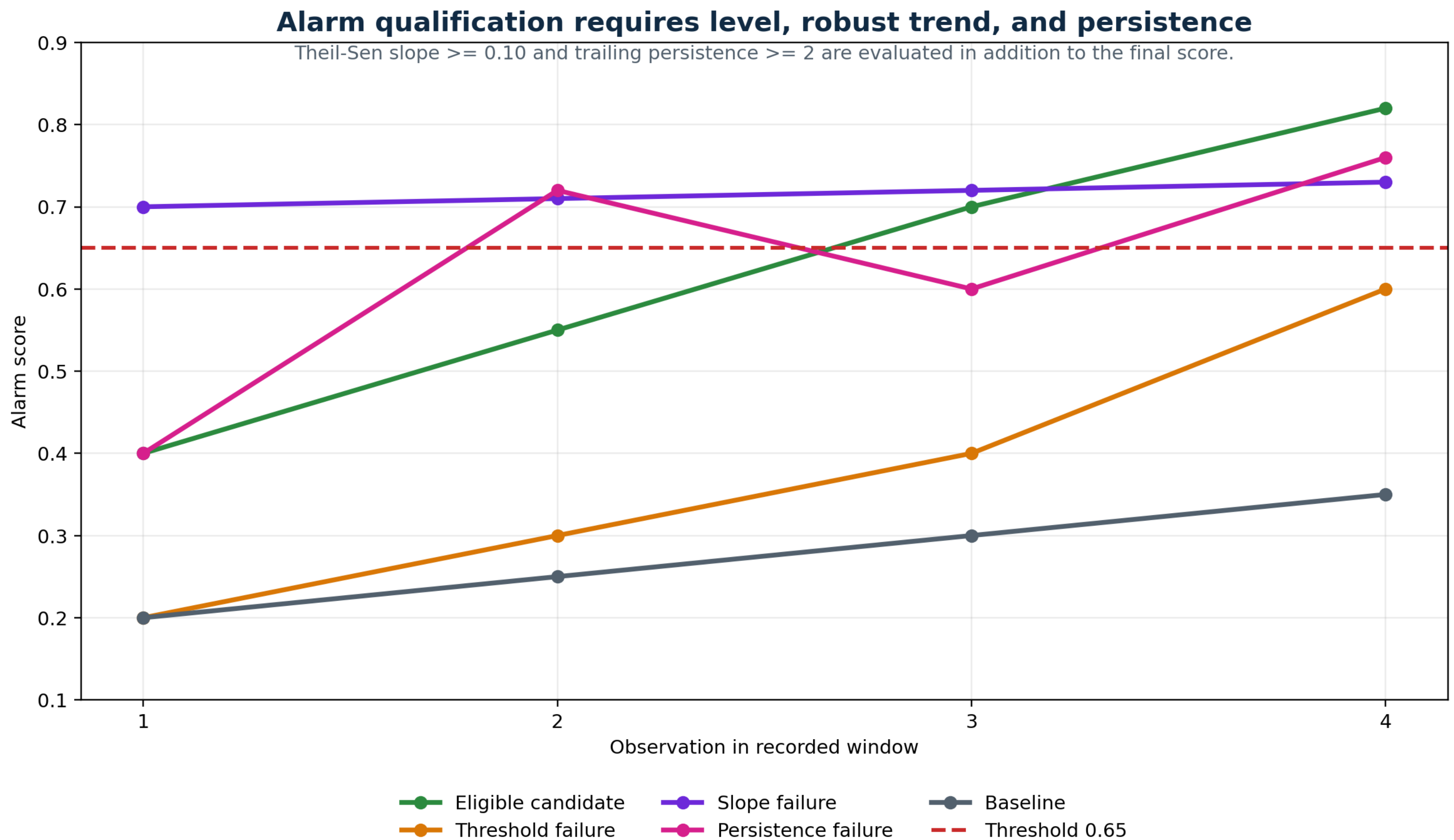


Figure 6. Fixture profiles isolate threshold, slope, persistence, and baseline failure modes.

Alarm component	Eligible-candidate example
Final score	0.82 / 0.84, above 0.65
Robust slope	Theil-Sen pairwise median, threshold 0.10
Persistence	two trailing observations above threshold
Family diversity	propagation and spectral families -> 2 required families

7. Independence, gain, and final admission

After state, quorum, and alarm processing, the trigger evaluates independent propagation paths, circularity, eta_ind, positive gain, cooldown, and final integrity. The eligible fixture declares two distinct paths - bond-population and population-cascade - with circularity false, eta_ind 0.80 against a 0.50 minimum, gain 1.0, cooldown inactive, and no final integrity veto.

The engine does not aggregate these conditions into a single opaque score. Every failed condition is appended to an explicit blocks list. This makes a negative outcome inspectable and preserves the exact reason the trigger did not form a candidate.

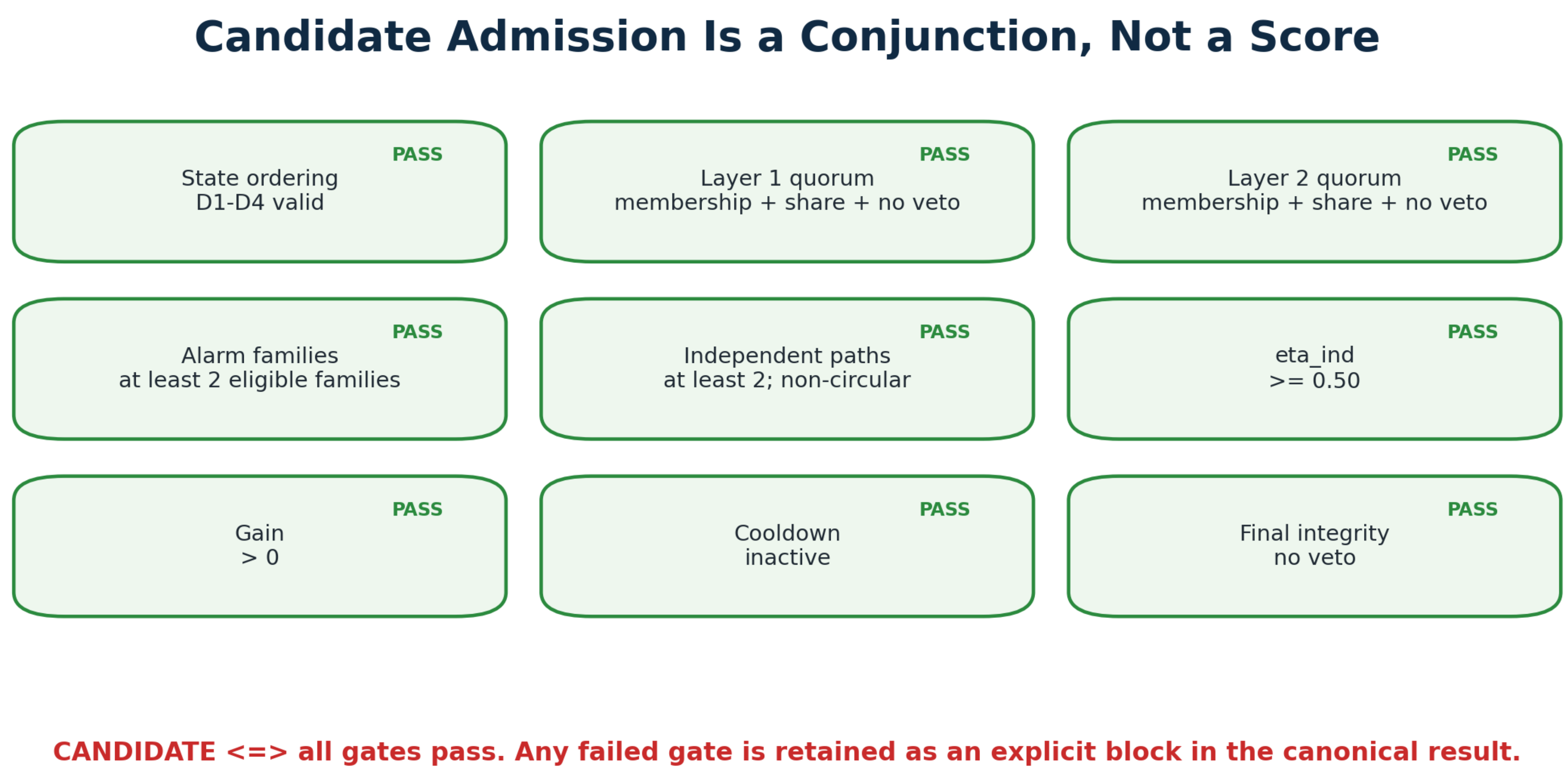


Figure 7. Candidate admission is the conjunction of explicit gates.

Decision rule: CANDIDATE if and only if the blocks list is empty; otherwise NO_TRIGGER. This is a governed admission rule, not a statistical probability of action.

8. Physical and motion input fusion

Physical Input Fusion 1.0.0 combines seven configured infrared, night-vision, and optical inputs under the BEAST_7EYE_V1 topology into a bounded five-coordinate qtrigger.layer1.physical.v1 packet. Motion Input Fusion 1.0.0 evaluates twelve deterministic synthetic motion nodes under MOTION_12_FACE_V1 and can form qtrigger.layer1.motion.v1 plus a governed qtrigger.layer1.physical.v2 combined packet.

Both modules remain synthetic-only and disabled for live hardware. The default Layer-1 source remains the physical.v1 contract, while the physical.v2 bridge and production admission remain blocked. No biometric identification, face recognition, identity inference, raw-frame retention, or camera opening occurs in the qualified demo path.

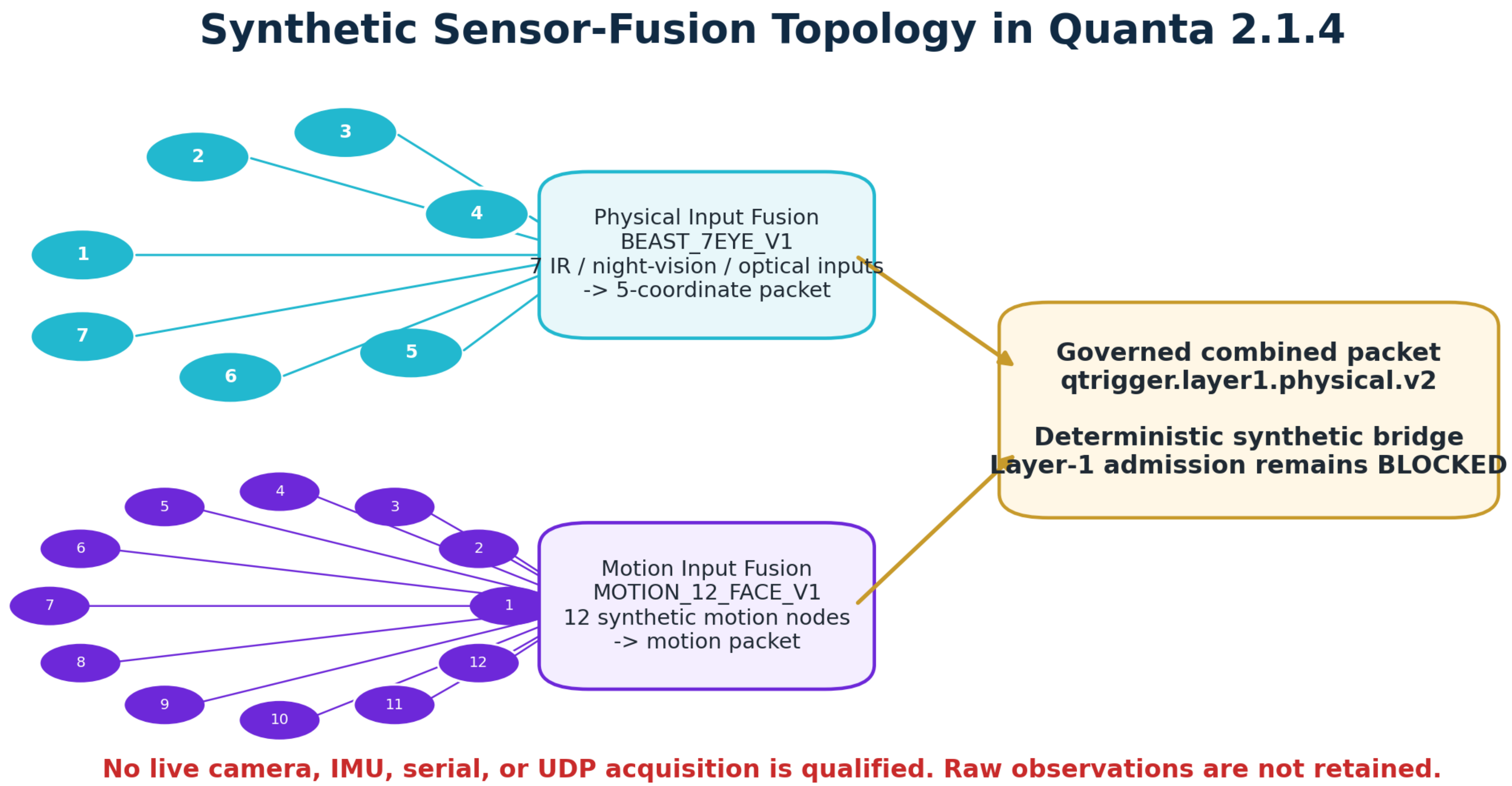


Figure 8. Seven-channel physical fusion and twelve-node motion fusion remain synthetic and blocked from Layer-1 admission.

9. Observation-only streaming gateway

The local /v2/streams gateway accepts independent frame and motion messages through bounded queues, aligns deterministic synchronization windows, invokes installed fusion pipelines, and returns compact derived results. Raw observations exist only transiently in memory and are cleared after each window. Publishing and stream management require scoped local credentials.

The service binds to 127.0.0.1, the network policy defaults to deny, telemetry and remote upload are false, and request and resource limits are explicit. The recorded synthetic benchmark processed 250 cycles with no dropped observations and a maximum queue depth of zero. These figures are a software benchmark on synthetic low-resolution inputs and are not a live-device performance claim.

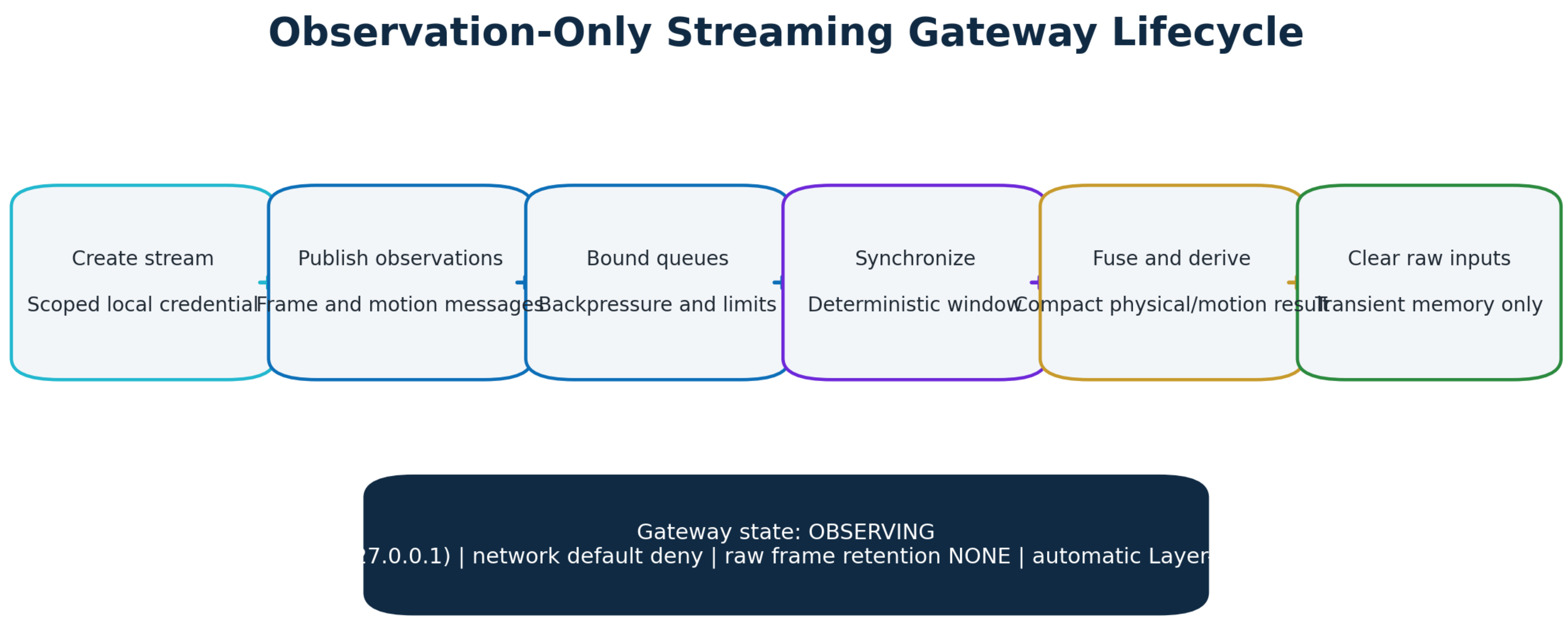
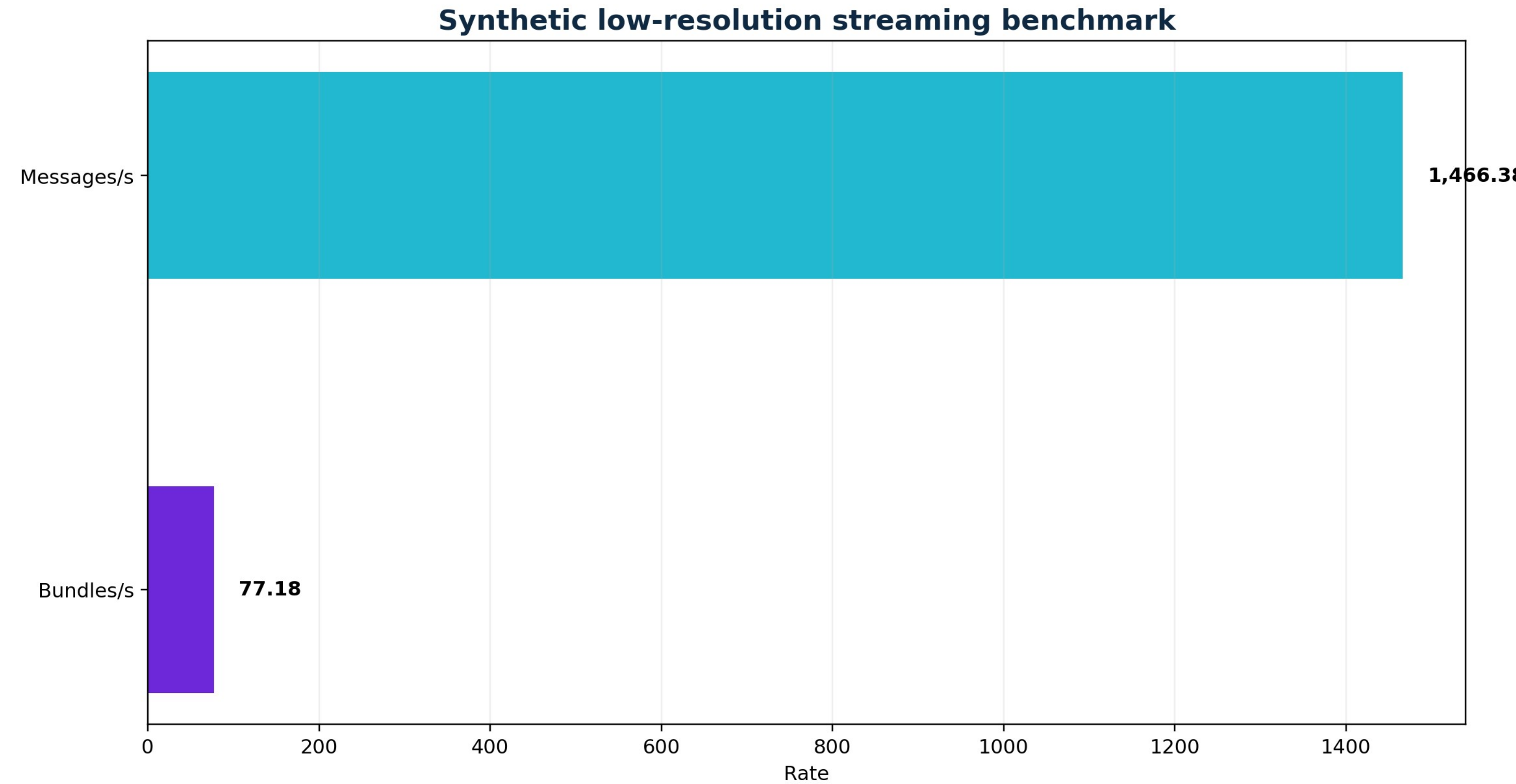


Figure 9. Stream lifecycle from scoped creation through synchronized derivation and raw-input clearing.



250 cycles | 0 dropped observations | maximum queue depth 0 | hardware performance claim: false

Figure 10. Synthetic gateway benchmark recorded in the 2.1.4 validation report.

10. Deterministic linear model-input boundary

The Linear Input Parser 1.0.0 converts derived physical and motion results into authorized deterministic vectors. Profiles include 5-, 10-, and 20-wide outputs and a temporal 48 x 20 batch form. Inputs are normalized, projected, bounded by policy, and accompanied by reproducible evidence digests.

The parser is deliberately an inference-input boundary, not an inference engine. The policy disables model execution, runtime learning, event hooks, physical actuation, hardware binding, and automatic Layer-1 admission. This separation allows the repository to prepare traceable model inputs without silently converting a candidate-generation system into an autonomous control system.

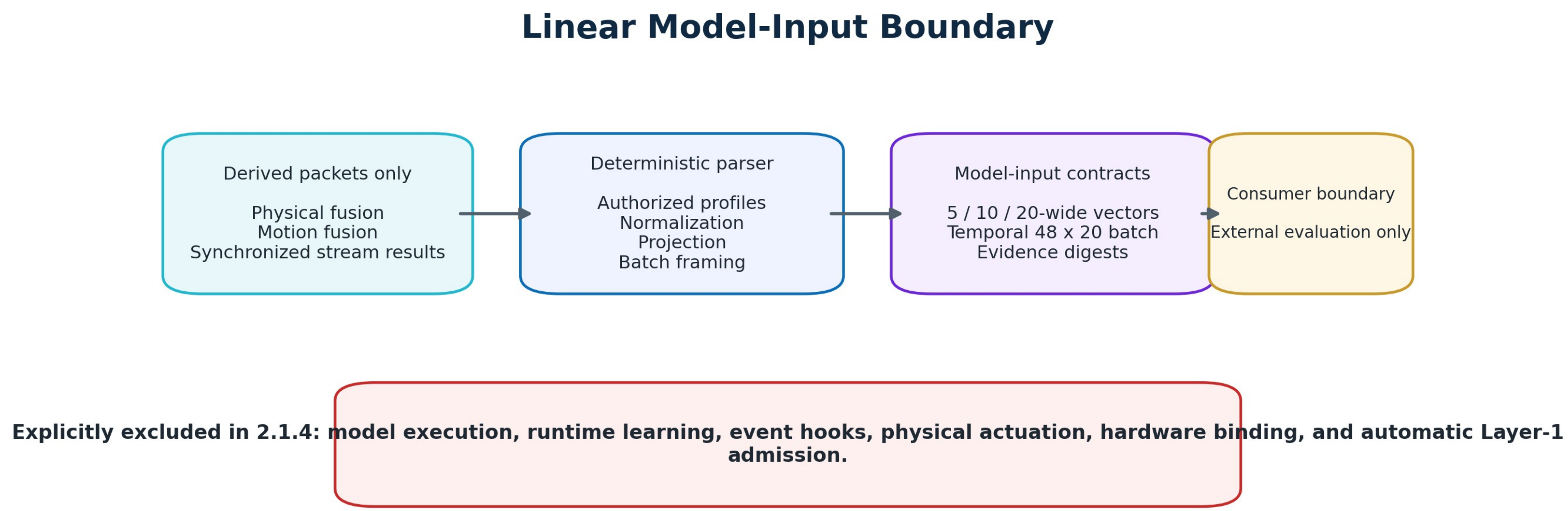


Figure 11. Derived packets are transformed into deterministic vectors while model execution remains outside the release.

Profile / record	Declared shape or limit
bond-proxy-5-v1	1 x 5
fusion-core-10-v1	1 x 10
fusion-quality-20-v1	1 x 20
fusion-temporal-v1	48 x 20
Maximum vector width	128
Maximum batch rows	256

11. Deterministic candidate and evidence formation

The canonical result records schema and engine versions, fixture and observation identity, calibration, normalized state, both quorum records, alarm analyses, family diversity, path independence, gates, decision, blocks, and registry status. The result digest is calculated over this canonical object. Attempt and candidate identifiers derive from the first twenty digest characters.

Floating-point values entering evidence are quantized to six decimal places before hashing. This remediation was added after earlier releases produced digests that could drift across numerical builds. The runtime uses Decimal constructed from strings and normalizes negative zero to prevent JSON-level divergence.

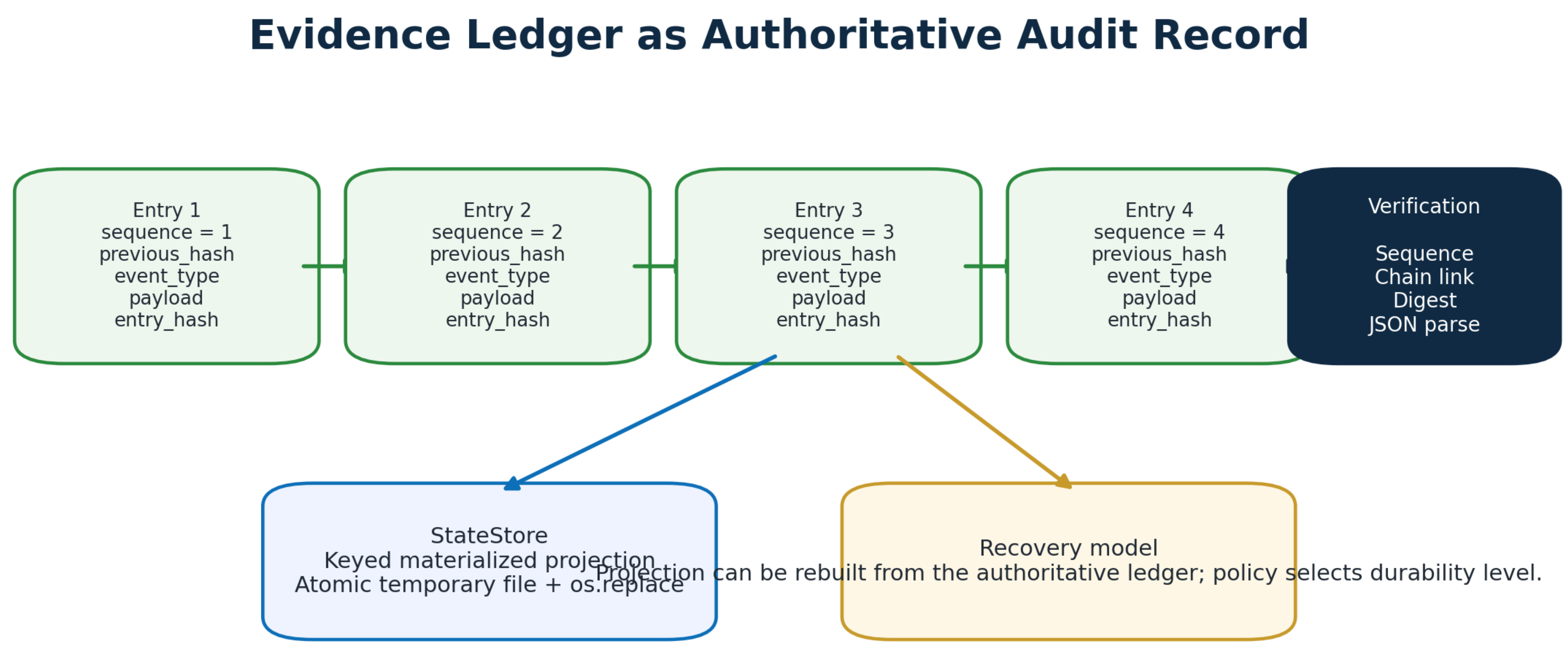


Figure 12. Evidence is preserved in an append-only hash chain, with the state store acting as a keyed projection.

Evidence rule	Effect
Canonical JSON	Sorted keys, compact separators, UTF-8, no ASCII escaping
Six-decimal quantization	Cross-build reproducibility for public numerical outputs
Replay timestamps excluded from digest	Operational logging does not alter deterministic result identity
Entry previous_hash + entry_hash	Tamper-evident append-only chain
External data root	Repository source and checksums remain sealed at runtime

12. Professor, Podium, dispatch, and rollback

Professor can propose only for a stored CANDIDATE with no unresolved blocks and only for an allow-listed target. The signed action card binds candidate ID, evidence digest, observed thresholds, capability request, expected effect, exclusions, idempotency key, expiration, and rollback plan. Professor cannot dispatch.

Podium requires a named operator and explicit confirmation. Its signed record binds the action card, target, capabilities, evidence, policy digest, decision, and readiness snapshot. The launcher re-verifies both records, checks scope and readiness, rejects duplicate dispatch, waits for acknowledgment, and returns the demo target to idle on rollback or simulated failure.

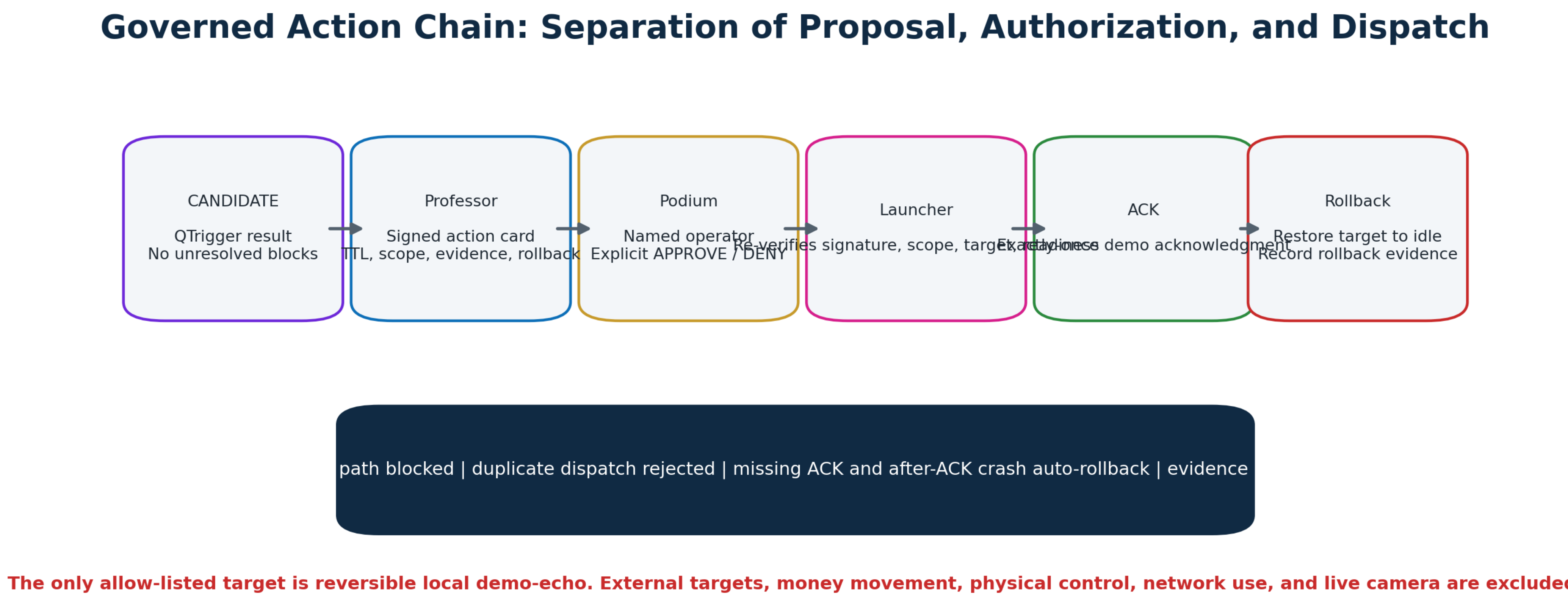


Figure 13. Proposal, authorization, dispatch, acknowledgment, and rollback are separate evidence-bearing stages.

Golden-path control	Recorded result
Denied authorization cannot dispatch	PASS
Duplicate dispatch	Rejected; exactly once
Missing acknowledgment	Automatic rollback
Crash after acknowledgment	Automatic rollback
Evidence export	Contains complete governed chain
Final demo target state	idle

13. Durability policy and the 2.1.4 performance correction

Release 2.1.4 corrects a durability inconsistency: 2.1.3 fsynced every StateStore write but never fsynced the authoritative EvidenceLedger. A governed chain performs eight store writes, and approximately 14.9 ms of the 17.68 ms total was fsync cost on the recorded reference host.

Relaxed mode retains atomic temporary-file replacement and flush for store records and append-and-flush for the ledger, but omits fsync. Strict mode fsyncs both. Both modes prevent partially visible records; only strict promises durability across power loss. The ledger and all record content are byte-identical under a frozen clock and fixed policy. A changed durability policy legitimately changes Podium authorization IDs because the policy digest is bound into authorization evidence.

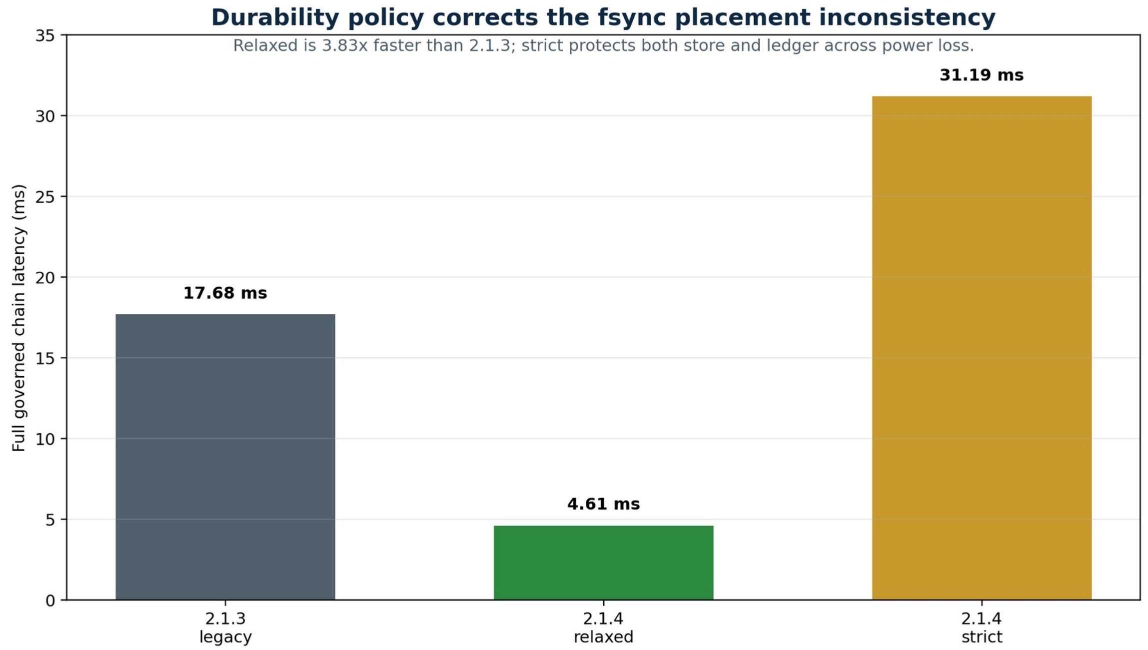


Figure 14.

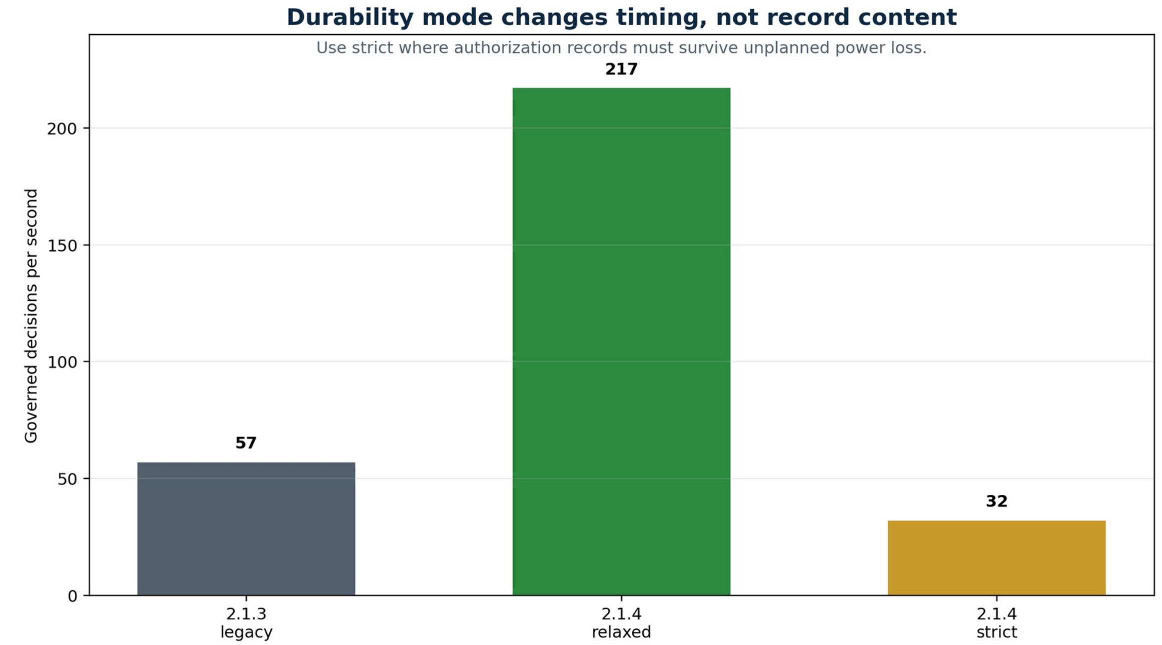


Figure 15.

Policy guidance: relaxed for replay, evaluation, and recoverable development records; strict where a governed action record must survive unplanned power loss.

14. Validation and reproducibility evidence

The package validation record reports PASS_PORTABLE_INTEGRATION_ONLY. In preparation of this exposition, the packaged test runner was re-executed and completed 125 tests with zero failures in 13.5 seconds. The validation artifacts record eleven deterministic replay fixtures, a stable replay digest, successful safe commands, no dropped observations in the synthetic stream benchmark, and a valid evidence chain.

The golden-path record confirms an acknowledged action, a blocked denied path, duplicate rejection, a valid hash chain, successful rollback, one start attempt, and final target state idle. The runtime test suite also exercises stale votes, Byzantine minority, integrity veto, circular propagation, robust slope, signature tampering, expiry, target mismatch, path traversal, loopback binding, schema migration, and failure recovery.

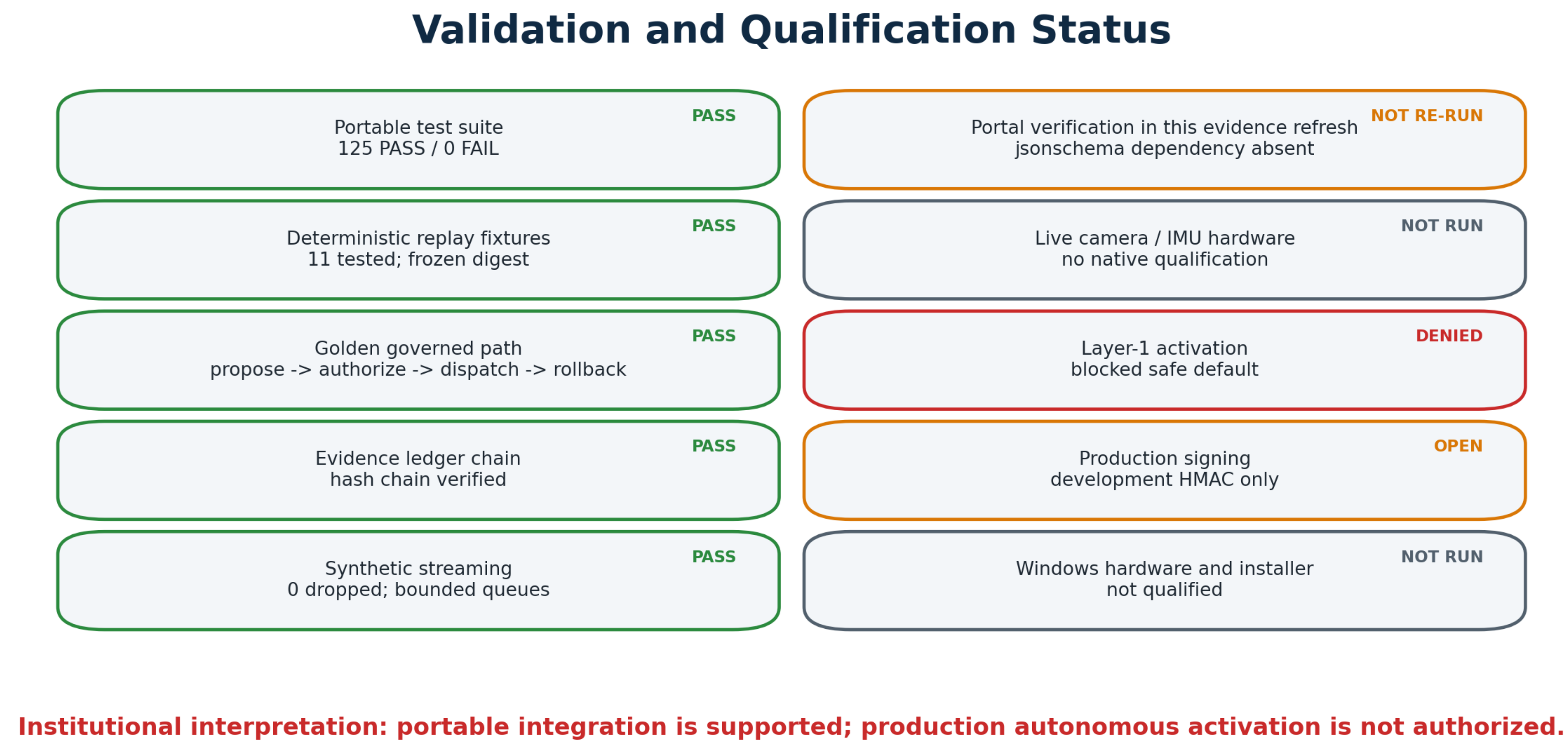


Figure 16. Portable validation is strong; hardware, production signing, and Layer-1 activation gates remain open or denied.

Evidence record	Result
Portable tests	125 passed / 0 failed
Replay fixtures	11 deterministic fixtures tested
Golden replay digest	ff390cbc...f9c0d25, unchanged since 2.1.1
Synthetic gateway	77.18 bundles/s; 1,466.38 messages/s; 0 drops
Layer-1 admission	BLOCKED_SAFE_DEFAULT
Hardware qualification	NOT RUN

15. Threat model, failure semantics, and safe defaults

CHARLOTTE threat modeling treats malformed inputs, stale or unhealthy votes, Byzantine minority, integrity defects, circular paths, single-point spikes, forged governance records, expired or broadened authorizations, duplicate launches, missing acknowledgments, crashes, traversal, hidden network use, and key disclosure as distinct threats. Portable tests cover most software controls; host-network audit, hardware qualification, production signing, and installer security remain deferred.

Failure is designed to terminate before consequential execution. Package validation rejects corruption and prohibited fields. Any failed admission gate yields NO_TRIGGER. Professor and Podium signatures and evidence links are verified again before dispatch. Missing acknowledgment and simulated post-acknowledgment crash return the demo target to idle and preserve the failure path in evidence.

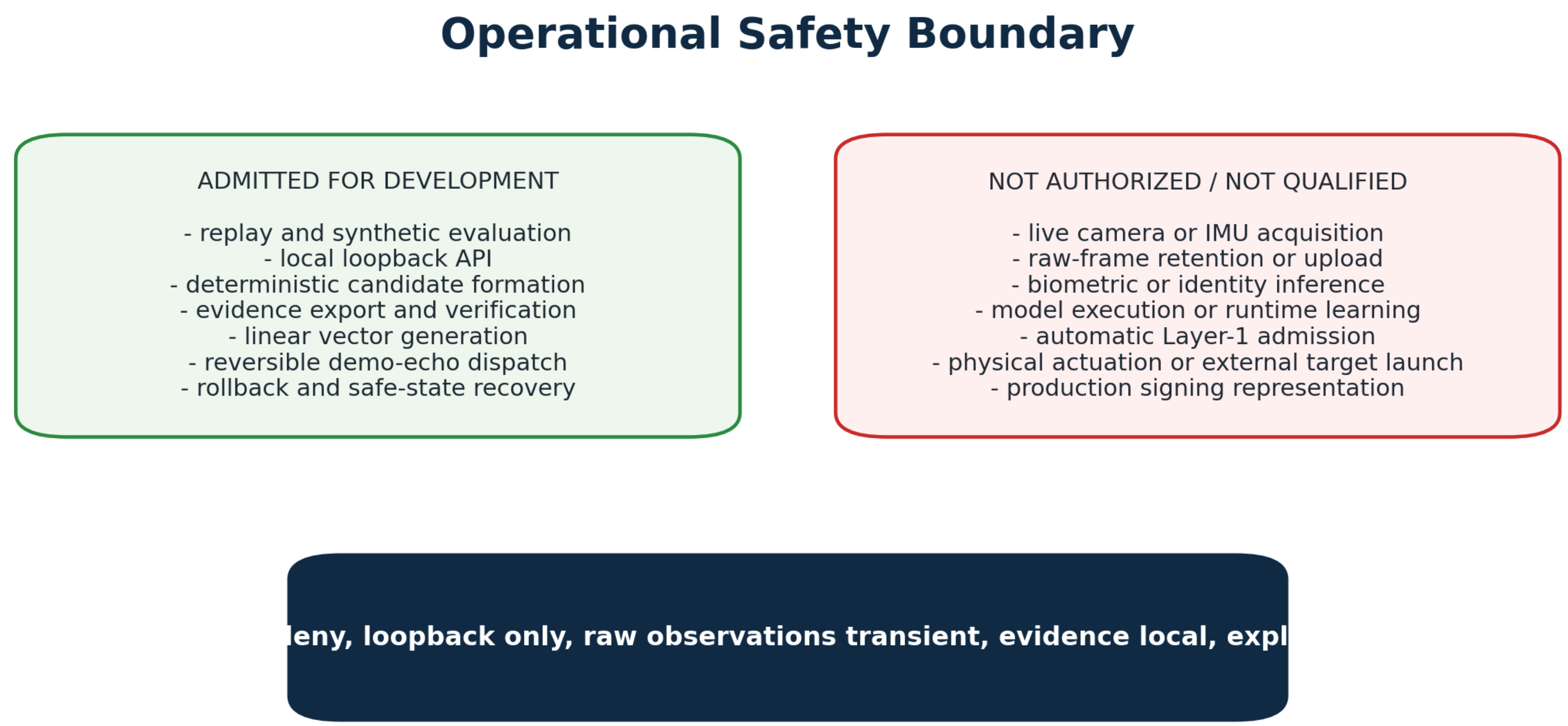


Figure 17. Admitted development functions are explicitly separated from denied or unqualified capabilities.

Development HMAC signing is evidence plumbing, not production OS-protected or hardware-backed signing.

16. Role-separated institutional governance

The five-role model prevents one engineering result from silently becoming a mathematical, assurance, authorization, or production claim. SOPHIA evaluates equations, invariants, policy, and admission logic. CHARLOTTE evaluates threats, privacy, UX safety, and failure boundaries. LANDON implements deterministic execution, validation, observability, and evidence generation. Professor frames a bounded proposal with expected effects, risks, evidence, expiry, and rollback. Podium authorizes or denies against predeclared gates and named operator confirmation.

For the present release, the role system supports deterministic replay, sensor-derived observation, candidate formation, bounded demo dispatch, rollback, and research. It does not authorize live hardware activation or production autonomy.

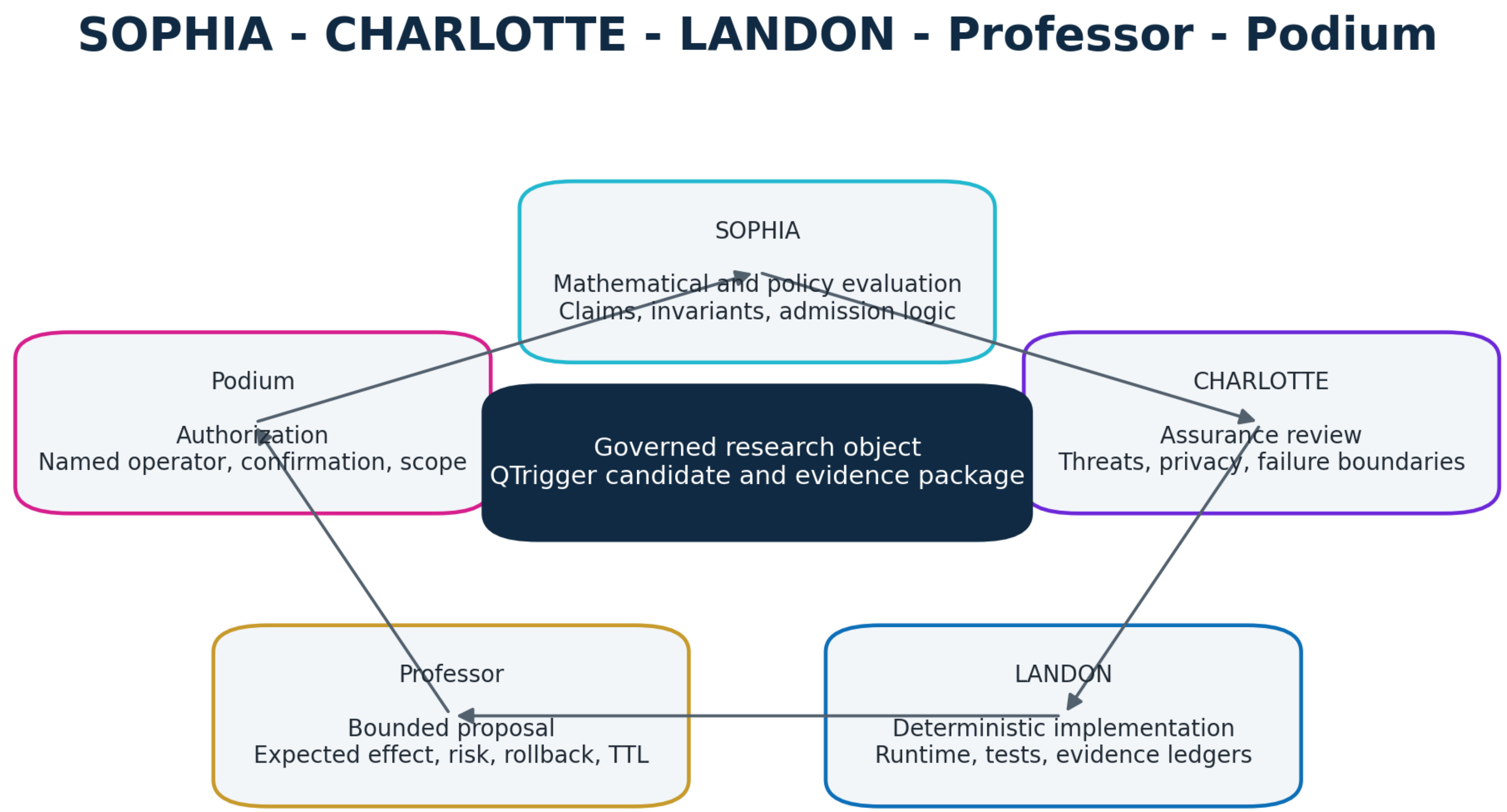


Figure 18. The five authorities surround one governed research object without collapsing their responsibilities.

Authority	Primary record
SOPHIA	Evaluation and mathematical/policy boundary
CHARLOTTE	Assurance review and threat model
LANDON	Implementation ledger and validation evidence
Professor	Action card / bounded proposal
Podium	Authorization or denial record

17. Laboratory registry and executable subset

The original trigger report selects 546 unique laboratory IDs from a larger corpus. It records 165 cascade-system alarm rows, 160 bond-pricing systems with four or more analytical dimensions, and 311 population and housing rows. Because laboratories may belong to more than one category, these category counts overlap.

The 1.1.0 replay runtime exposes the registry but activates only twelve reference adapters; 534 entries remain registry-only. This is an important research-control distinction. Registry membership preserves provenance and candidate mechanisms, while executable participation is deliberately bounded. The runtime caches the static 546-row labs.csv after the first parse; the 2.1.3 optimization removed repeated parsing that had dominated replay evaluation time.

Registry dimension	Count / status
Unique selected laboratories	546
Cascade-system alarm rows	165
Bond-pricing systems (4D+) rows	160
Population and housing rows	311
Active reference adapters	12
Registry-only entries	534
Disabled / excluded / failed adapters	0 / 0 / 0

Registry breadth is not equivalent to active execution. The institutional claim should remain: twelve reference adapters are executable in deterministic replay; the remainder are catalogued research records.

18. Repository consistency and declared architectural debt

The runtime release version is correctly centralized through the VERSION file and runtime release module, which report 2.1.4. Several historical or inherited documents retain earlier labels: the top-level README heading says 2.1.2, RELEASE_MANIFEST identifies 2.1.1, and the core manifest retains 1.9.0 module labels. These records do not change the executable VERSION value, but they weaken publication clarity and should be reconciled before an external institutional release.

The repository also explicitly records architecture debt. Most importantly, automatic Layer-1 admission is blocked because the feature is unimplemented rather than because a fully mechanized policy evaluator decided to block it. That safe default is appropriate, but it is not yet the same as an independently verified formal activation gate.

Debt / inconsistency	Institutional treatment
Historical version labels	Reconcile README, release manifest, core manifest, and generated reports to VERSION = 2.1.4.
Layer-1 gate is constant	Preserve block; implement a real policy decision only in a separately qualified release.
Development signer	Replace with production provider before any consequential use.
Hardware and Windows qualification	Keep NOT RUN; publish machine-specific evidence when executed.
Live adapters absent	Do not infer camera, IR, night-vision, IMU, serial, or UDP readiness.
Independent verifier absent	Commission a second implementation from the written contracts.

Professional publication should preserve negative and incomplete evidence rather than editing it into a cleaner narrative.

19. Recommended research program

The next phase should improve institutional evidence before expanding operational authority. First, freeze and reconcile release metadata. Second, build an independently authored verifier for replay packages, D1-D4 ordering, quorum, alarms, canonical digests, and the governance chain. Third, partition or optimize long-running properties and enforce declared types, effects, manifest checks, and primitive availability.

Only after those foundations should the program qualify real optical, infrared, night-vision, and motion hardware. Hardware work must include calibration signatures, timing and synchronization bounds, sensor dropout, clock drift, privacy controls, raw-data handling, thermal and resource behavior, Windows qualification, and adversarial failure tests. Production signing, key custody, zeroization, and side-channel review must precede any consequential authorization. Automatic Layer-1 admission should remain a separate research release with independent review and a default-deny rollback path.

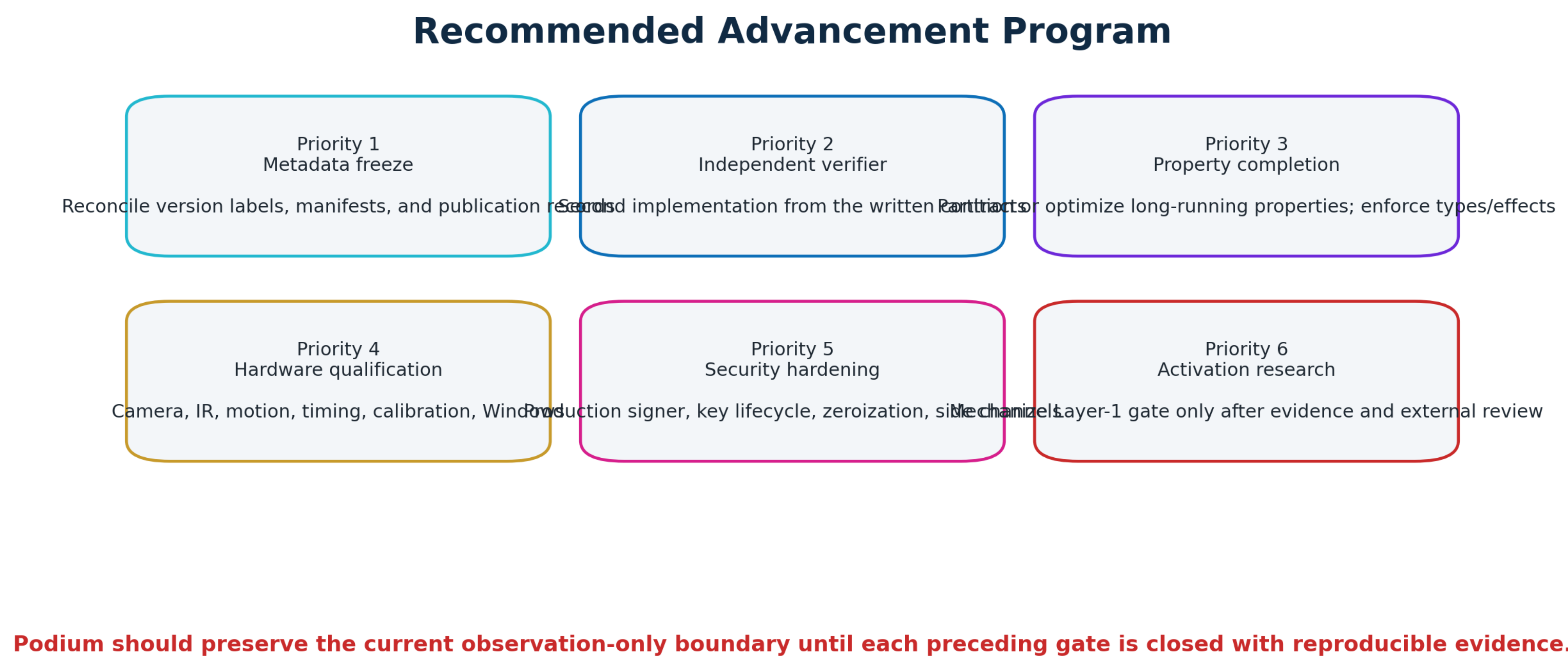


Figure 19. Recommended sequencing keeps activation research behind metadata, verifier, property, hardware, and security gates.

20. Institutional conclusion

Quanta AI Trigger 2.1.4 demonstrates a disciplined separation between sensor observation, deterministic candidate formation, evidence, human authorization, and bounded execution. Its contribution is not autonomous activation. Its contribution is an inspectable trigger architecture in which every input boundary, state relation, vote, alarm, path, gate, proposal, authorization, dispatch, acknowledgment, rollback, and ledger transition can be represented as evidence.

The repository supports portable-development claims: 125 tests pass; eleven replay fixtures are deterministic; the golden path reaches a reversible demo acknowledgment and returns to idle; the evidence chain verifies; streaming queues are bounded; raw observations are not retained; and the 2.1.4 durability correction improves the relaxed governed chain to 4.61 ms while adding a genuinely strict mode that protects the authoritative ledger.

The release remains intentionally bounded. It does not execute an AI model, admit results to Layer 1, bind live hardware, authorize physical control, or provide production signing. These boundaries strengthen the research because they make the current evidence precise. The next institutional milestone is an independent verifier and reconciled release record, followed by controlled hardware qualification under the same role-separated governance model.

Podium disposition: ADMIT deterministic replay, observation-only streaming, candidate formation, evidence verification, linear input preparation, reversible demo dispatch, and performance research. HOLD live hardware activation, model execution, automatic Layer-1 admission, physical actuation, external targets, and production-security representation.

Appendix A - concise citation summary

Quanta AI Trigger 2.1.4 is a deterministic, governed sensor-observation and candidate-formation system. It validates replay or synthetic packages, enforces ordered D1-D4 state relations, evaluates two weighted Byzantine quorum layers, qualifies alarm families by threshold, Theil-Sen slope, and persistence, applies independence, eta_ind, gain, cooldown, and integrity gates, and emits CANDIDATE only when no block remains. A candidate cannot dispatch directly: Professor creates a signed bounded action card, Podium requires a named operator and explicit approval, and the launcher re-verifies evidence, scope, readiness, idempotency, acknowledgment, and rollback.

The repository records 546 unique selected laboratories, 12 active reference adapters, and 534 registry-only entries. Portable validation reports 125 tests passed and zero failed, eleven deterministic replay fixtures, a valid evidence chain, and a golden governed path that rejects denial and duplicate dispatch, acknowledges one reversible demo action, and returns the target to idle. Version 2.1.4 reduces the relaxed full governance chain from 17.68 ms to 4.61 ms while adding a strict 31.19 ms mode that fsyncs both the authoritative ledger and state projection.

The release remains observation-only: live hardware, raw-frame retention, biometric inference, model execution, runtime learning, automatic Layer-1 admission, physical actuation, external target launch, production signing, and production autonomy are not authorized or qualified.

Citation metric	Value
Release	Quanta Repository 2.1.4
Trigger	QTrigger 1.1.0
Registry	546 unique labs; 12 active; 534 registry-only
Tests	125 passed; 0 failed
Durability	4.61 ms relaxed; 31.19 ms strict
Synthetic streaming	77.18 bundles/s; 1,466.38 messages/s; 0 drops
Boundary	Observation-only; no model execution or automatic activation

Appendix B - key source records

ID	Source record	Role
S1	.labs/qtrigger/1.0.0/docs/report.md	Rough trigger report and 546-laboratory registry rationale.
S2	runtime/quanta_runtime/qtrigger.py	Executable state, quorum, alarm, admission, digest, and fixture semantics.
S3	runtime/quanta_runtime/governance.py	Professor action cards, Podium authorization, scope and evidence checks.
S4	runtime/quanta_runtime/store.py	Atomic StateStore and append-only hash-chained EvidenceLedger.
S5	.quant/policy/runtime-policy.json	Observation, network, governance, resource, retention, model-input, and durability policy.
S6	RELEASE_NOTES_2.1.4.md and DURABILITY.md	Durability correction, benchmark, compatibility, and power-loss boundary.
S7	VALIDATION_REPORT.json	2.1.4 portable validation status, tests, benchmark, evidence, safety, and deferred gates.
S8	validation/GOLDEN_PATH_REPORT.json	Candidate-to-rollback governed path and evidence-chain result.
S9	docs/runtime/ARCHITECTURE.md / STATE_MODEL.md / THREAT_MODEL.md	Architecture, lifecycle, trust boundaries, and threat controls.
S10	PHYSICAL_INPUT_FUSION_GUIDE.md / MOTION_INPUT_FUSION_GUIDE.md	Seven-channel physical and twelve-node motion fusion boundaries.
S11	SENSOR_STREAMING_API_GUIDE.md / LINEAR_INPUT_PARSER_GUIDE.md	Observation-only streaming and deterministic model-input boundary.
S12	JA21_PORTAL_CYPHER_ENGINE_4.2.0_INSTITUTIONAL_EXPOSITORY_NOTE.pdf	Institutional publication-format precedent; not a technical source for QTrigger claims.

Document classification: Research draft / development integration | Autonomous or production activation representation: Not authorized.