

PERMISSION IS NOT PROOF

The missing decision layer in enterprise agent platforms

LLM-agnostic. Model-agnostic. Agent-agnostic.

RiGiGroup.com | WhiteWater.ai

FROM EVIDENCE TO VERIFIED EFFECT



EVERY VERIFIED EFFECT BECOMES NEW EVIDENCE

Jimmy Standaert

Founder, RiGi Group / Whitewater AIOS

CAIO / Forward Deployed Engineer/Architect

Context-Verified Decision Intelligence



Enterprise agents need more than permission

A response to Naresh Waswani's *From AI Agent to Enterprise Agentic Platform* - and a proposal for the decision-integrity layer enterprise platforms still need.

An enterprise platform can prove that an agent was permitted to act and still fail to prove that the decision was justified - or that the intended effect occurred.

1 · Permission	2 · Decision integrity	3 · Proof
Identity, authorization, policy, budgets, approvals, and limits determine whether an action may proceed.	Evidence, contradiction, freshness, and uncertainty determine whether the proposed decision should proceed.	Source-native receipts determine whether the intended real-world state actually changed.

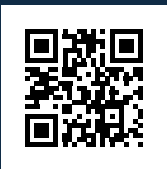
The argument

- Govern the entire evidence-to-effect chain, including the execution boundary.
- Evaluate claims individually: a polished answer can contain both solid and unsafe assertions.
- Keep UNKNOWN explicit; fail closed for consequential effects when evidence or authority is incomplete.
- Require a completion receipt from the system where the effect should exist.

A strong platform foundation

Waswani's architecture provides a strong enterprise execution foundation. Whitewater AIOS extends it with decision integrity.

Agents decide; platforms enforce. Keep deterministic controls outside model behavior.	Reasoning is not durability. Use workflows, retries, approvals, cancellation, and checkpoints.
Authority attenuates. Delegation may preserve or narrow authority; it must never expand it.	Execution is bounded. Limit time, cost, steps, tools, fan-out, and depth.
Versions are immutable. Pin definitions to preserve reproducibility.	One trace spans the task tree: routing, models, tools, delegations, policy, cost, and outcomes.



From governed execution to governed decisions

Execution governance constrains action. Decision governance connects evidence, authority, action, and verification into one reviewable record.

Execution governance	Decision integrity
Identity and policy Tool and agent permissions Budgets and limits Durable task state Audit and observability	Claim-level evidence state Contradiction and freshness Authority-after Refusal disposition Source-native effect receipt

One durable object for the decision that matters

A Governed Decision Object binds the evidence, decision, authority, intended action, and proof needed to review or reproduce a consequential outcome.

1 · Claim	The independently falsifiable assertion driving the decision.
2 · Evidence	Sources, supporting spans, provenance, freshness, and relevant silence.
3 · Contradiction	Credible evidence that conflicts with the claim or its scope.
4 · Decision	The selected conclusion, alternatives, and unresolved uncertainty.
5 · Authority	Who or what may act, under which constraints, and what remains prohibited.
6 · Action	The bounded effect requested from a tool, workflow, agent, or human.
7 · Receipt	Source-native evidence showing the resulting external state.

Both layers are required for trustworthy autonomy. Make the decision reviewable without exposing private chain-of-thought or depending on an agent’s own narrative of success.



Confidence is not a control

A single confidence score hides the exact claim that is unsafe. Evidence and actability should be explicit, per claim, and tied to permitted behavior.

Evidence / actability	Meaning and permitted behavior
SUPPORTED · SOLID / FIRM	Current evidence entails the claim. Commit or take bounded action inside existing authority.
UNKNOWN · SOFT	Evidence is missing, stale, inaccessible, or inconclusive. Retrieve, ask, qualify, or abstain; fail closed for consequential effects.
UNSUPPORTED · SOFT / VOID	Available evidence does not establish the claim. Rewrite, drop, or verify before promotion.
CONTRADICTED · VOID	Credible evidence conflicts materially with the claim. Block, expose the conflict, and escalate or correct.

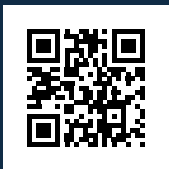
Evidence state never grants authority by itself. A well-supported decision can still be prohibited, private, out of scope, or too consequential for automatic action.

Every consequential action needs two envelopes

The execution envelope answers whether an action is permitted. The evidence envelope answers whether the decision is sufficiently grounded for that action.

Execution envelope	Evidence envelope
Principal and tenant Agent and tool versions Delegation chain Cost, time, step, and depth limits Policy obligations Approval state Cancellation and trace IDs	Material claims Sources and provenance Freshness requirements Contradictions and silence Evidence and actability states Required verification Refusal and recovery behavior

Proceed only when both envelopes permit the effect. Otherwise retrieve, narrow, ask, qualify, refuse, or escalate.



Prove the effect. Re-evaluate what changes.

Agent statements, tool responses, and workflow states are useful evidence. Consequential completion requires verification from the system where the change should exist.

Request	What state should change?
Decision	Why is the change justified?
Execution	What call, tool, or workflow ran?
Readback	What does the source system now show?
Receipt	What durable evidence proves the effect?

No source-native receipt means the effect remains unverified - even if every upstream system reported success.

Reproducible does not always mean still valid

Version pinning tells us what ran. Drift controls tell us whether evidence, authority, policy, or effect still supports continuing the run.

Evidence drift	A source changes, expires, is deleted, or is contradicted.
Authority drift	A user's role, delegated permission, or approval status changes.
Policy drift	A rule changes while a durable task is waiting or retrying.
Capability drift	A tool contract, model behavior, or provider boundary changes.
Effect drift	External state changes after the receipt was captured.

Delta records what changed. **Drift Delta** records whether the change invalidates the decision, narrows authority, or triggers re-evaluation.

Trust begins where permission ends

The next frontier is proving the integrity of the decision itself: current, attributable evidence; visible uncertainty and contradictions; explicit, bounded authority; reviewable action; and source-native verification.

Keep the decision visible. Bring back the receipt.



Where the decision layer lives

Decision integrity is a governed contract spanning the platform components that control context, policy, execution, evaluation, and audit.

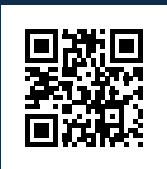
Context / RAG	Policy / authorization
Preserve source identity, access scope, freshness, and trust labels.	Return obligations and authority-after, alongside permit or deny.
Execution governor	Capability registry
Bind consequential actions to an actable decision object.	Declare evidence requirements, receipt type, reversibility, and risk.
Evaluation service	Audit / observability
Test grounding, contradiction handling, refusal, drift, and effect verification.	Preserve decision changes, approvals, actions, and receipts as one trace.

Turn principles into architecture fitness functions

1. No consequential action executes without a claim-level evidence state.
2. UNKNOWN fails closed when consequence exceeds the permitted risk threshold.
3. Delegation cannot expand authority or discard inherited evidence constraints.
4. Decision sources preserve provenance, scope, and freshness metadata.
5. Contradictions affecting actability stay visible until resolved or accepted.
6. Every refusal records its disposition and next permitted recovery action.
7. Every consequential effect returns source-native readback or remains unverified.
8. Long-running decisions define drift events that trigger re-evaluation.
9. Production decision changes are attributable, reviewable, and reversible where possible.
10. Readiness claims use gate-specific evidence, not a blended confidence score.

Selected references

Waswani (2026), *From AI Agent to Enterprise Agentic Platform*; NIST (2023), *AI RMF 1.0*; NIST (2024), *Generative AI Profile, AI 600-1*; OWASP (2025), *Top 10 for Agentic Applications 2026*; W3C (2013), *PROV-O*; ISO/IEC 42001:2023, *AI management systems*.



A decision contract at the execution boundary

Illustrative scenario: an agent prepares a customer quote. A human approves the final recipient, attachment, and commercial terms. Evidence freshness and authority must still hold when the send is committed.

Minimal Governed Decision Object

```
{
  "decision_id": "demo-001", "tenant": "demo",
  "claim": {"text": "Quote uses approved terms", "state": "SUPPORTED"},
  "evidence": {"source_version": "terms-7",
    "observed_at": "2026-10-26T14:00:00Z", "max_age_s": 86400},
  "authority": {"actor": "demo-operator", "action": "send_quote",
    "policy_version": "p12", "approval_ref": "approval-12",
    "payload_digest": "<digest-of-final-payload>"},
  "receipt": {"state": "UNVERIFIED", "source_id": null}
}
```

The digest field is a placeholder for a computed digest of the recipient, attachment, and message. This example is a record shape, not a complete schema or production implementation.

Execution gate and completion contract

At commit: authenticate the actor and tenant; validate current policy, scoped authority, source version and freshness; require supported material claims with no unresolved material contradiction. Bind human approval to the exact payload digest. Any mismatch blocks the effect or returns it for review.

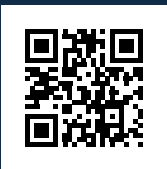
After commit: use a stable idempotency key bound to the decision and payload. A timeout triggers reconciliation before retry. Read back the destination system; mark COMPLETED only when its receipt matches the intended recipient and effect. An accepted API call alone leaves the effect UNVERIFIED.

Failure injection: expected outcomes

Change introduced	Required disposition
None: current evidence, valid approval	Execute once; complete only after matching source readback.
Evidence expires or source version changes	HOLD; refresh evidence and re-evaluate the decision.
Material contradiction appears	HOLD; expose conflict and obtain accountable human review.
Recipient, attachment, or message changes	Invalidate payload approval; require approval of the new package.
Authority is revoked before commitment	BLOCK; do not perform the external effect.
Timeout, duplicate request, or missing receipt	Reconcile first; prevent duplicate effects; retain UNVERIFIED until proven.

Implementation boundary: revalidate at commitment using conditional writes or a scoped, short-lived capability where supported. If the provider lacks atomic controls, disclose the remaining race and apply a compensating control or manual handoff. Digest matching binds content; it does not prove truth.

Evidence status: synthetic design example and proposed acceptance criteria. These are expected outcomes, not results of executed tests or production validation.



Five perspectives on decision integrity

Proposed language for speaker review. Final publication should use only quotes approved by the named speaker.

"If the decision matters, the evidence has to travel with it - and the receipt has to come back."

Jimmy Standaert

Founder, RiGi Group / Whitewater AIOS

"The best system is the one that makes the next right action clear without hiding what still needs a human decision."

Cindy Hawkins

CindyJoStudios.ai

"A durable workflow is useful. A durable workflow with evidence, authority, and verified completion is governable."

Douglas Earp

CTO, RiGiGroup.com

"Trust grows when leaders can see what the system knew, what it decided, what it was allowed to do, and what actually happened."

Garen Armstrong

Trussi.ai

"AI accountability becomes real when an organization can reconstruct the path from source evidence to business effect."

Mike Harvey

Functional Architect, DarkNova.ai

These quotations are drafted for review and are not presented as verbatim statements or endorsements until approved by each named speaker.



Human judgment. Decision integrity. AI enabled.

Whitewater AIOS (WWAIOS) is a framework for Context-Verified Decision Intelligence: connecting what an AI system knows, what a human intends, what authority permits, and what actually happens. Its purpose is to reduce distortion between reality and decision.

LLM-agnostic. Model-agnostic. Agent-agnostic.

A consistent decision contract across people, models, agents, and tools.

What makes the approach distinctive

WWAIOS brings claim-level evidence, visible uncertainty and contradictions, bounded authority, human review, drift controls, and source-native completion receipts into one evidence-to-effect chain. A Governed Decision Object carries the basis, limits, approvals, and verification requirements of a consequential decision across handoffs.

What it enables

Teams can design workflows that pause when evidence is insufficient, escalate when judgment is required, preserve authority through delegation, and verify results in the system where the effect should exist. This supports reviewable AI assistance, accountable execution, and the freedom to change models or agents without surrendering decision governance.

Celebrate the human in the loop.

Human decision integrity is the organizing principle: people can understand, challenge, approve, redirect, and own consequential decisions. Human judgment remains visible and valued.

About Jimmy Standaert

Jimmy Standaert is the founder of RiGi Group and the creator of Whitewater AIOS. A CAIO and Forward Deployed Engineer/Architect, he brings more than 20 years of experience across technology, AI, and HealthTech, with a focus on CTO advisory, transformation, and trustworthy decision-making.

Architecture and design intent are described here; implementation and outcomes require evidence in each deployment.

Jimmy Standaert | RiGi Group / Whitewater AIOS

CAIO / Forward Deployed Engineer/Architect

913-221-3137 | jimmy@rigigroup.com

RiGiGroup.com | WhiteWater.ai | JimmyStandaert.ai

