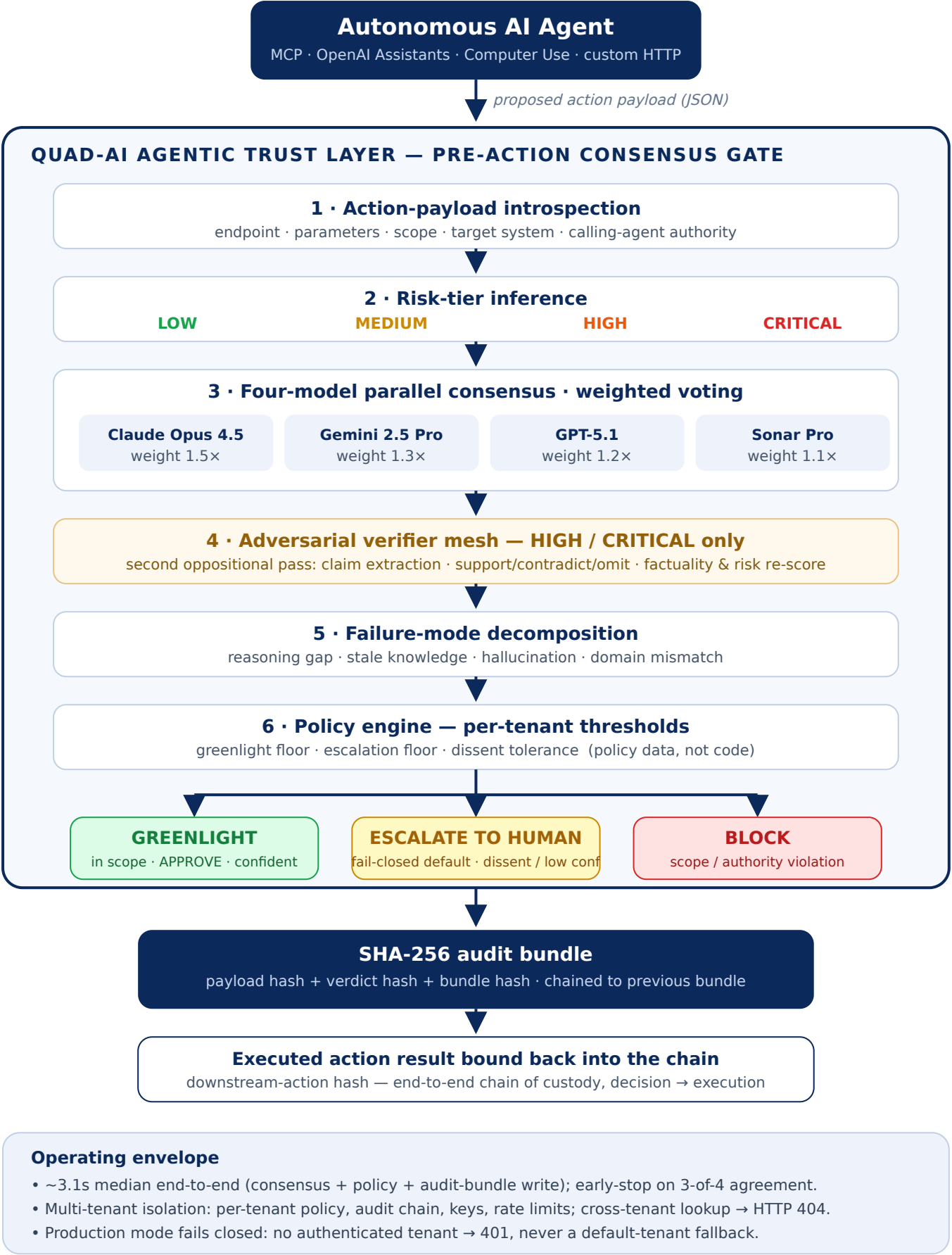


Pre-Action Consensus Gate — Flow

How one proposed agent action becomes a verdict and a tamper-evident audit record.



Read the gate as a funnel. Low- and medium-risk actions clear quickly; high- and critical-risk actions earn the full adversarial pass before any verdict. Critical actions always route to a human even when approved. Every path — greenlight, escalate, or block — produces the same tamper-evident, replayable audit bundle. Scrutiny — and spend — scale with the stakes: the full four-model consensus lands on the decisions that can actually hurt you, not on every routine call. The engine behind this gate runs in production under Fortune 100 contracts in regulated industries.

Inside step 6 — deterministic policy-class detectors. Alongside the consensus thresholds, four deterministic detectors watch for the regulatory-exposure classes an enterprise can't afford to miss. They read only the action's **verb, target, and parameter field names** — never the values, never free-form context:

- **Monetary threshold** — a money-movement verb (wire, transfer, payout, refund) at or above a configurable limit (default \$10,000).
- **Protected-personal-data modification** — a mutating verb touching a field named for PII (ssn, tax id, date of birth, bank account, card number, biometric).
- **Protected-class adjacency** — a consequential-decision verb (approve, deny, score, hire, price, underwrite) when a protected-class attribute is present (race, religion, gender, age, disability, pregnancy, veteran status).
- **Regulated-health-data access or transmission** — an access or send verb touching a PHI field (diagnosis, MRN, treatment, lab result).

These are a deterministic floor — independent of and **additive to** the model layer, so the hard regulatory lines never depend on a model being right. They are escalate-only: a detector can raise a greenlight to *escalate-to-human*, but never blocks, never auto-approves, never overrides a scope-violation block, and never softens an escalation already raised. Block stays reserved for a categorical scope or authority violation. Each trigger records the policy class and redacted category and field signals (for example `verb:update`, `field:ssn`, `phi:diagnosis`) — never the underlying value. And because they run in deterministic code, they enforce at **zero model cost** — even if every model is wrong, these lines still force a human into the loop, and they cost nothing in inference to run.

A complementary session-exposure signal. Optionally, when the integrator passes a session identifier (bound to the authenticated tenant), the gate also watches *cumulative* sensitive-data exposure across that session, and can escalate an outbound action once several distinct sensitive categories have been touched — even when no single step crossed a line alone. Like the four classes it's escalate-only and name-only; think of it as an extra layer of caution alongside them that only ever adds a human check, and steps aside if the session store is unavailable.

A correlated-consensus check on the agreement itself. Four models agreeing only means something when they agree *independently* — when they agree for the same reason, a shared stale fact or blind spot, confidence reads high while the answer can still be wrong. So alongside the verdict the gate reads how independently the four responses actually arrived, and when that correlation risk is elevated and at least three providers returned answers, it escalates the action to a human, carrying that read with the decision as evidence — the guard against the one failure consensus is blind to on its own: everyone being confidently wrong together.

Step 0 — cryptographic agent identity (August 2026). Before any of the steps above, the gate can now demand proof of *who* is calling. Each agent holds its own Ed25519 keypair; each request carries a signed identity envelope (signature, key version, timestamp, and a single-use nonce, so a captured envelope cannot be replayed), and delegated authority arrives as a signed chain in which every hop can only **narrow** the verbs and targets granted — a link that tries to grow authority breaks the chain. Enforcement is per-tenant policy (off / annotate / escalate / required): at `required`, an unidentified or

forged caller is refused before a single model is consulted — zero inference cost — and the identity evidence (status, checks, chain summary) is committed inside the canonical bundle hash.

And around the gate (August 2026): an article-by-article **EU AI Act evidence pack** assembled from the tenant's sealed records, with explicit non-coverage notes and an Article 73 serious-incident draft from any escalated or blocked decision (documentation support — the legal determination stays with counsel); **assurance telemetry** for underwriters — 30-90 days of descriptive counts and rates, deliberately no composite score (Veridect is not an insurer); and an **MCP gateway** that governs [tools/call](#) in flight — the verdict is written into the JSON-RPC response itself, and a refused call is never forwarded to the tool at all. All live on the narrated tour at [veridect.ai/governance-suite](#). **Since October 2026** a forwarded greenlight carries a short-lived Ed25519-signed **execution permit** bound to the sealed decision and the exact action, verifiable by the tool side against the published JWKS with no call back (in production the gateway refuses, by default, to forward a greenlight without one; a human-approved escalation issues an approval permit bound to the original decision record); the same evidence pack also assembles along **SR 26-2's** own sections, scoped on its face to the agentic tools that guidance leaves to the bank's own governance, never a compliance claim.

And the gate now proves itself (August 2026). Three further capabilities turn the layer's own discipline back on itself. **Adversarial self-test:** on demand, the four providers each author novel attack scenarios against the tenant's own live policy surface — scope grants, dollar thresholds, protected-data rules, stealth patterns — and every scenario is judged by the same deterministic core that gates live traffic, in strict isolation: synthetic consensus, zero writes, no session state, results stored apart from the real evidence ledger. Every probe the deterministic floor alone would have greenlit is flagged to a human — and a modest containment number is the honest, expected reading, because live traffic additionally gets the four real model votes this test deliberately withholds. It is a self-test of the deterministic layer, not a substitute for external validation. **Policy impact replay:** before changing a confidence floor, dissent tolerance, or the escalation dollar threshold, replay a window of the tenant's real recorded decisions under the hypothetical policy — deterministic re-derivation, zero model calls, zero writes — and see exactly which past decisions would flip and what drove each flip — a single field where one is responsible, a combination where several act together. Thresholds the sealed record cannot re-derive (session and workflow exposure) are refused outright rather than silently ignored, and a scope-violation block can never flip under any override. **Dissent ledger:** every provider dissent goes on a windowed record — who dissented, lone dissents versus splits — and where a human later reviewed the same decision, whether that review vindicated or overruled the dissenter. Counts and rates only, deliberately no provider ranking, and the dissent text itself stays sealed inside the audit bundles. All three are live on the engine's API and run live on the narrated governance tour at [veridect.ai/governance-suite](#); the Governance Command Center surfaces the same evidence read-only — latest self-test summary and windowed dissent record beside the verdict mix — with read failures disclosed as unavailable, never rendered as a clean zero.

And four more oversight instruments (August 2026). **Near-miss ledger:** every recorded greenlight re-derived under three single-knob tightened neighbors of its own attested policy — confidence floor raised a step, dissent tolerance reduced by one, dollar threshold reduced a step — through the same deterministic core that gated it live. Approvals that flip are reported by the exact threshold that would have tipped them: counts and threshold names only, a policy-sensitivity fact rather than a verdict on the action, with non-re-derivable records counted and excluded rather than guessed. **Case-law engine:** any recorded decision can be handed its precedents — the closest past human-resolved reviews from the tenant's own chain, matched deterministically on feature names (action verb, parameter field names, detected policy classes, risk tier, scope outcome, delegation) and surfaced as resolution enums, reason codes, similarity, and age, with the free-text rationale left sealed inside the bundles. Context for the reviewer, never a recommendation. **Overnight frontier:** the replay primitive swept across a deterministic grid of up to 72 neighboring policies around the window's median attested policy, mapping how many recorded approvals each variant would have sent to a person and how many

recorded escalations it would have released — Pareto-efficient variants flagged, recorded blocks invariant under every variant, and nothing auto-applied: the engine maps the frontier, the human moves the policy. **Model character ledger:** each provider's recorded voting character compared across two adjacent windows — dissent rates, stance mixes with per-window coverage disclosed, and the exact model identifier strings seen, so a silent vendor model swap becomes a dated fact. Minimum samples gate every shift flag, and the view is descriptive only — never a provider ranking. All four are live on the engine's API, tenant-scoped, read-only over the sealed ledger: zero model calls, zero writes.

The audit bundle carries the decision basis. The policy-class triggers and the decision basis are written inside the canonical bundle hash and persisted to an append-only (WORM) store, and each bundle exports as JSON, CSV, an HTML governance report, or OpenLineage NDJSON for SIEM and data-lineage ingestion. When the cleared action executes, its result is bound back into the chain, and any auditor can independently recompute every hash to confirm nothing was altered or inserted after the fact — a forged or out-of-band row fails recomputation or breaks the chain. Beyond the hash chain, each bundle is **Ed25519-signed over its bundle hash** — proof not just that the record is intact, but of who produced it — and signing keys are versioned and rotate forward, with every prior version kept published ([GET /api/ai/consensus/proof/public-key](#) for the current key, [/proof/key-history](#) for every version), so a bundle signed by a rotated-out key still verifies. As with any signature, the published key is pinned out of band. And since October 2026 each tenant's chain tip is time-stamped on a fixed cadence by independent public RFC 3161 authorities — Veridect checks the digest and nonce and publishes the token for independent verification; the chain never references an anchor, so bundle hashes and offline proof are unaffected.

Three governance add-ons — Govern. Prove. Insure. The three governance problems the market keeps calling unsolved — Veridect solved all three, live in production. They extend the layer's Verify · Govern · Prove spine. **Constellation** governs a whole agent workflow: it accumulates sensitive-data exposure across every agent in a [\(tenant, workflow\)](#) and can escalate a live action the moment their combined behavior crosses a line no single agent did — escalate-only and name-only, so it can only add a human check, never blocks, never auto-approves, and never alters a recorded decision. **Veridect Proof** lets an auditor re-derive a stored verdict, re-check the SHA-256 chain, and verify the Ed25519 signature offline with a dependency-free standalone verifier — it re-checks the decision and the record, it does not re-run the models. **Veridect Assurance** renders a stored decision as a neutral, five-category, PII-free evidence record a risk-transfer partner can assess — Veridect is not an insurer, and the record is not pricing, certification, or coverage. Proof and Assurance are read-only over stored bundles; all three are tenant-scoped (cross-tenant lookup returns 404) and run live on [veridect.ai/agent-trust-demo](#).

Fleet-scale policy classes — the swarm horizon (August 2026). Constellation watches one workflow's crew; three further deterministic policy classes watch the whole tenant fleet, over days — the shape of the July 2026 swarm breach, where roughly 700 agents spread about 17,600 individually plausible actions across identities that shared no session and no workflow. The **Fleet Exposure Ledger** accumulates every sensitive read an agent identity makes — across every session and workflow — in a sliding seven-day window, beside a fleet-wide total for the tenant; when a slow-burn identity, or the fleet itself, crosses the line and then reaches outbound, the action escalates with the triggering fleet counts and category rationale on the record. **Swarm Convergence** counts distinct identities touching the same sensitive category or the same hashed action shape inside a 72-hour window; an outbound step inside a converged pattern escalates with the cohort size on the record. **Target Convergence** watches many identities converging on one target — held in fleet state only as a truncated cryptographic hash — and escalates the outbound step that moves toward it. Same discipline as every deterministic class: escalate-only, field-name categories and hashes only — never values — at zero model cost, keyed to the tenant and agent identity the gate already receives, so there is no new plumbing; the classes ship default-off and are enabled per deployment, and when on they govern every surface, including MCP [tools/call](#) through the gateway. Their long-window memory is durable by construction: shared state is a working copy — on a cold store the layer re-derives a bounded recent window from the sealed decision ledger

under a single-owner rebuild lease, and if the rebuild cannot complete, the detectors stay silent while the primary gate keeps enforcing; since October 2026, while the window is cold a fleet-enabled tenant's outbound greenlights are held for human confirmation rather than cleared against counters known to be incomplete. Rebuildable and checkable from the record, never a cache taken on trust.

Exercised at ten thousand identities (10 September 2026). A deterministic exercise of the fleet layer — real policy engine, real fast-state store, zero model calls, a unanimous synthetic approve standing in for the consensus so every escalation is the fleet layer's alone — put 10,000 distinct agent identities through one tenant, one proposed step each, at a convergence threshold of 25 (8,500 confidential reads, 1,000 in-scope external sends, 250 in-scope high-risk transfers, 250 transfers under a read-only grant). The first outbound step escalated for human review came from identity #30 — three had cleared before it, exactly as the threshold specifies — and all 1,496 outbound steps after it were escalated or blocked, none cleared. A benign control fleet of 10,000 identities under the same policy drew zero fleet escalations; an identity-less fleet of 4,000 actions was caught by the tenant-wide counter once pooled reads crossed the line; all 10,000 rulings agreed with an independent oracle re-statement of the policy. After the tenant's fast state (17,009 keys) was wiped and verified gone, the next ordinary outbound step rebuilt the window from the sealed ledger — a bounded reconstruction from the most recent 5,000 records — and escalated with all three classes. Then the same swarm went through the published system as real requests on a dedicated tenant (10–11 September 2026) — ten thousand live rulings, each a real multi-model ruling with three providers responding and each sealed in its own signed record: 8,487 cleared, 1,263 escalated for human review, 250 blocked; detection at identity #30 again, and all 1,496 outbound steps after it escalated or blocked; a benign control fleet of 300 live rulings with zero fleet escalations; 10,300 of 10,300 signed records verified offline against the published key; all eleven acceptance checks passed. The live numbers are reported beside the deterministic ones, never merged; 120 live rulings carry a disclosed error entry from one provider instead of a stance, the remaining providers clearing the three-provider quorum on every one. Both are our own runs, not part of the externally validated set; a count of identities ruled on, not a rate. Record: **Fleet-Scale Exercise Record**.

Coordination-fabric policy classes — the seven-class expansion (late August 2026). Fleet scale watches how much; seven further deterministic policy classes watch how agents work *together*, on the same gate. **Delegation-chain shape:** the identity layer verifies every hand-off cryptographically — this class measures the verified chain itself at the moment of action: depth beyond tenant policy, pass-through hops that only relay authority, wildcard grants, a sensitive action executing far from the human grant. It never re-judges authority; it escalates the risky shape. **Separation of duties:** the gate records which identity produced an artifact, and when the same identity later approves, verifies, or releases that artifact, the sign-off escalates — maker and checker collapsing into one agent is the pattern financial controls have never tolerated in humans. **Coordination shapes:** fragment reassembly — each agent reads a harmless fragment, every one individually under the line, and only the reassembled per-agent union of what they touched crosses it, checked at egress; and role pipelines — collect, transform, transmit spread across distinct identities, escalated at the outbound end with the shape on the record. **Plan conformance:** an agent commits its plan up front, sealed and Ed25519-attested (`POST /api/ai/consensus/plan/commit` , fetch via `GET /api/ai/consensus/plan/:planId`); each gated step is checked for verb, target, and order against the step it claims to be, and a mismatch, an out-of-order step, or a replayed step escalates as plan deviation. **Quarantined shared memory:** every governed memory write lands in a provenance ledger that is write-once by database trigger; a write from an unverified identity, or one carrying sensitive-named fields, is quarantined at ingest, and an action citing quarantined memory escalates until a human releases the entry through an audited, admin-gated release (`GET /api/ai/consensus/memory/quarantine` , `POST /api/ai/consensus/memory/quarantine/release`). **Queue-message provenance:** messages carry envelopes signed by the sending agent's own Ed25519 key; an unknown sender, revoked key, or invalid signature surfaces at the consuming step, an unsigned envelope that merely names an active agent is recorded as *unproven* rather than trusted, and under the

strictest setting only a gate-greenlit enqueue can attest a message hash — a blocked enqueue can never launder its own message into trusted work. Identical discipline to every class above: escalate-only, name-only (field names, verb tokens, sanitized identities, hashes — never values or content), deterministic at zero model cost, fail-open to the base gate on state failure, default-off knobs or opt-in by presence of `planRef` / `memoryRefs` / `messageEnvelope`, and replayed verbatim from the sealed bundle at proof time. A companion **policy-reversion report** (`POST /api/ai/consensus/policy/reversion`) diffs the attested configuration history over a window and counterfactually re-derives the recorded decisions under the prior policy — zero model calls, zero writes, with thresholds the sealed record cannot re-derive excluded and disclosed. All post-date the May red-team validation and are recorded in the validation summary as shipped-since additions, not as part of the externally validated set.

Governance observability — the command center. Every verdict on the flow above is also readable back as a live, read-only governance view over the same tamper-evident ledger: verdict mix, risk-tier distribution, which policy classes are firing, four-provider consensus health, a per-agent fleet view, and a running timeline — the whole fleet over time, not one decision at a time. It only *reads* the record: it never makes or changes a verdict and adds no new enforcement, it is tenant-scoped, it returns an error rather than a misleadingly empty view when records can't be read, and at production volume it serves from a materialized rollup. Live on `veridect.ai/agent-trust-demo` (`GET /api/ai/consensus/observability`).

Oversight quality — proof the humans are actually looking. A gate that escalates is only as good as the reviewers on the other end. So the layer measures the review layer too: each resolved escalation is recorded to the same tamper-evident chain and rolled up into a plain rubber-stamp-risk read — low, moderate, or elevated — led by how often genuinely-contested actions were approved unchanged. It reads the pattern across enough reviews to be meaningful, never singling out one reviewer, and it holds only neutral markers — an opaque reason code and reviewer reference, never names or free text (`POST /api/ai/consensus/audit-bundle/:id/oversight` to record, `GET /api/ai/consensus/oversight/quality` to read).

Lisa Russell

CEO

lrussell@veridect.ai · veridect.ai