REGULATORY AI NAVIGATOR · RELEASE TEASER

v1.5 — Trust & Time

From mapping obligations to **proving** them — and keeping the map honest while ten regulatory frameworks keep moving. A release shaped by the community.

The last release drew three challenges from practitioners. Each one exposed a structural gap, not a missing node. v1.5 answers all three — with new graph structure, not with prose.

> *"Are you modelling evidence requirements and assurance status in the graph as well, or is the Navigator mainly focused on design-time scoping? … A useful graph may need to separate obligation, control objective and evidence."*
>
> CHALLENGE 1 — THE ASSURANCE QUESTION

> *"What happens six months after this graph ships? … a node that was accurate at launch can quietly become wrong without anyone noticing until an audit asks for the current mapping, not the original one."*
>
> CHALLENGE 2 — THE DRIFT QUESTION

> *"The Art. 12 logging obligation is only as strong as whether the log can be trusted after the fact. A WORM vault stops your own team editing it, but the sharper test is whether someone who does have admin rights could."*
>
> CHALLENGE 3 — THE TRUST QUESTION

184	416	13	15
nodes (+15)	edges (+49)	node types — new: **Control Objective**	edge types — new: **operationalized_by**, **satisfied_by**

Use case → risk class → regulation → control objective → component → evidence → attestation. One queryable graph, community-validated, versioned as patch releases.

WHY THIS IS HARD

One sentence of law, one system of consequences

Regulatory text is written per obligation. Systems are built per component. The translation is many-to-many: a single article explodes into several **testable control objectives**, each satisfiable by alternative components, each demanding its own evidence at its own assurance level. Flatten any layer and you either over-build or fail an audit. v1.5 makes the middle layer explicit.

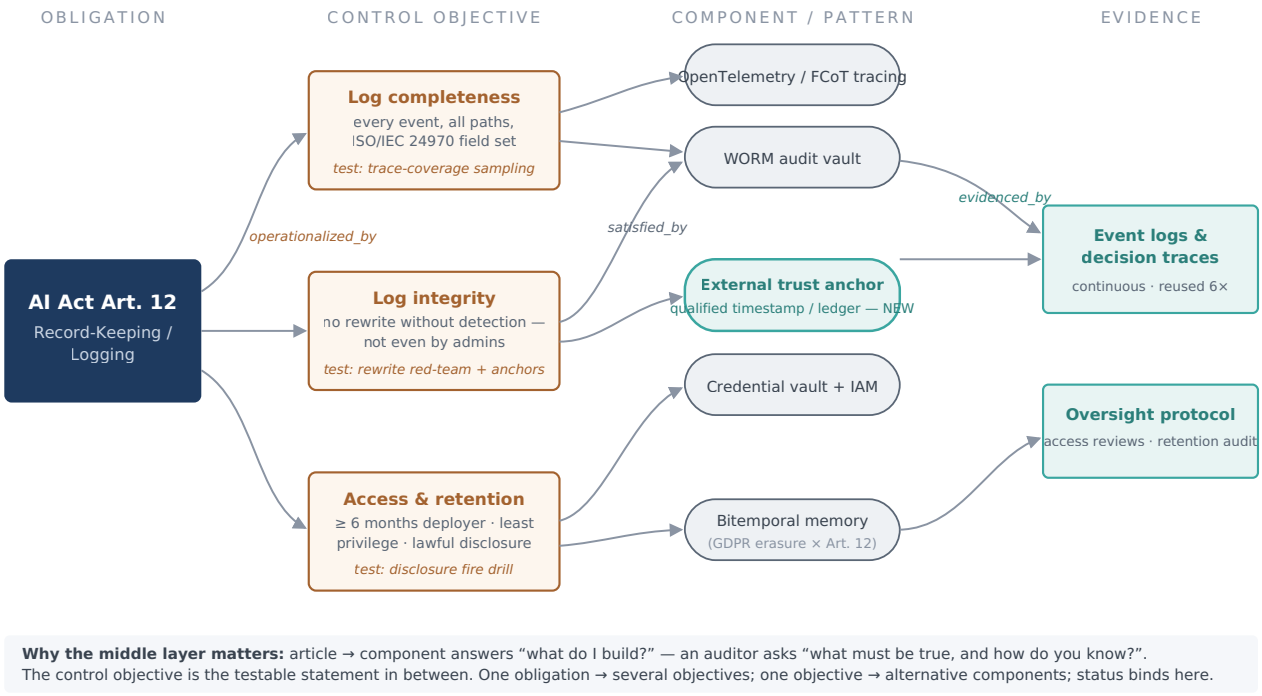
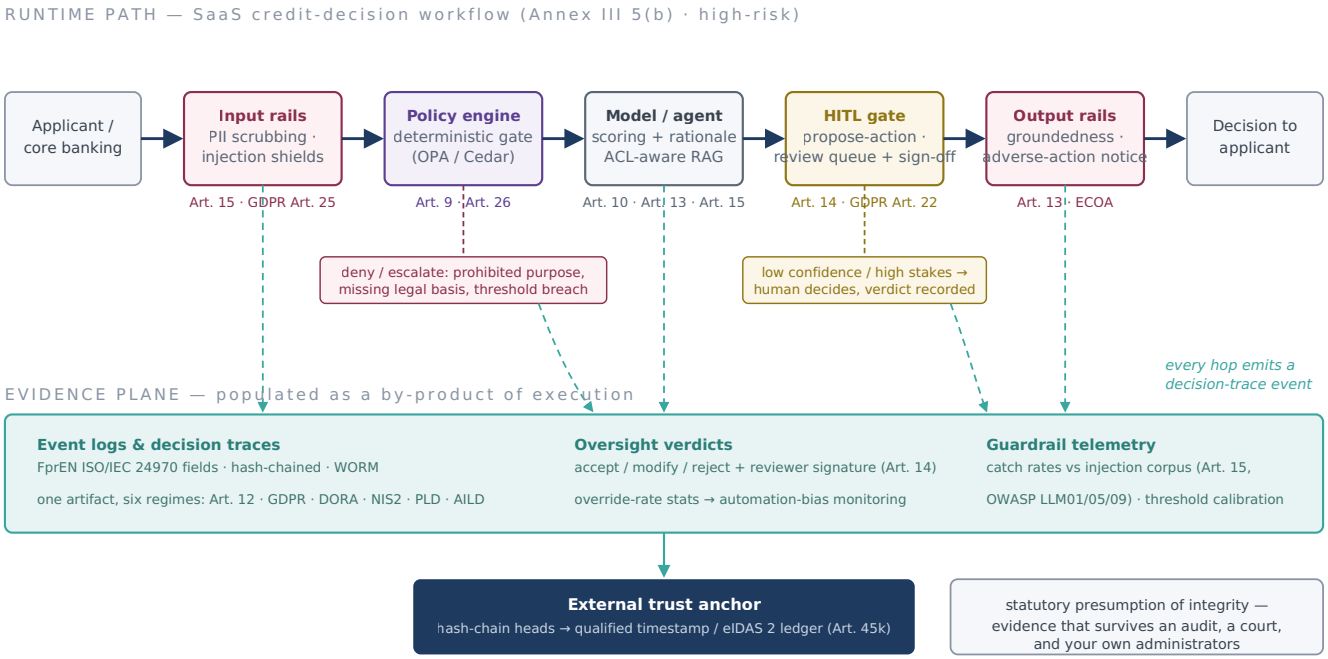


Fig. 1 — The four-layer chain piloted in v1.5 on Art. 12 / 14 / 15: obligation → control objective → component → evidence. The 55-word logging paragraph alone yields three distinct objectives with different tests, different components and different failure modes.

FROM FRAMEWORKS TO RUNNING SOFTWARE

A regulated workflow is an architecture, not a document

Take one concrete SaaS workflow — an AI-assisted credit decision. Every regulatory component sits **inline in the execution path**, not beside it: the request cannot physically reach the model, and the decision cannot reach the customer, without passing the controls. The evidence plane below is populated **as a by-product of execution** — compliance artifacts as production artifacts.



Design rule: a control that can be bypassed in code review is documentation, not a control. Chokepoints (gateway, policy engine, gates) make the compliant path the only executable path — and the evidence plane fills itself. Retrofit? Run new controls in shadow mode first: observe, score, calibrate — then enforce.

Fig. 2 — Regulatory components inline in a SaaS credit-decision workflow. Colored borders mark control classes: rails, deterministic policy, human oversight, evidence. Annotations show the article each component discharges. The same skeleton holds for KYC, claims, HR screening and clinical workflows — swap the domain, keep the chokepoints.

ANSWER TO CHALLENGE 3 — THE TRUST QUESTION

How strong is your evidence when someone attacks it?

“We log everything” is the start of the argument, not the end. v1.5 grades every evidence artifact on a **verifiability ladder** — orthogonal to *who attests it* (self / internal audit / third party / notified body). The rung you need depends on the scenario: an internal review accepts rung 2; a liability proceeding under the AILD, where your logs are your defence, does not.

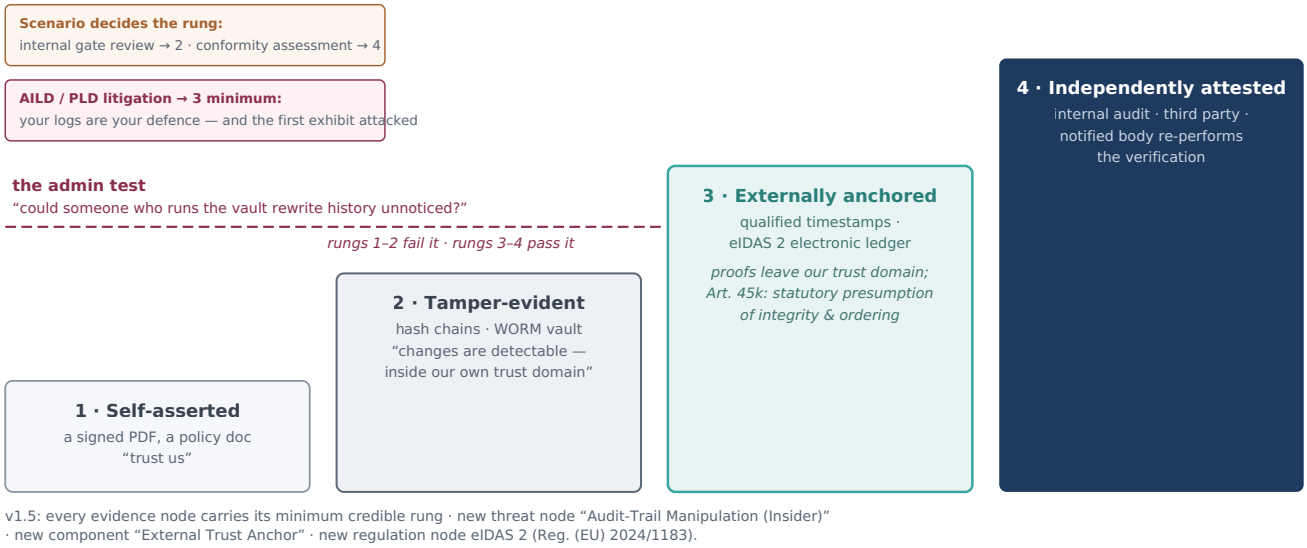


Fig. 3 — The verifiability ladder. Anchoring is periodic and cheap — one hash-chain head per interval — but upgrades every log-derived artifact at once: Art. 12 records, DORA incident timelines, NIS2 logs, the AILD presumption defence.

ANSWER TO CHALLENGE 2 — THE DRIFT QUESTION

A graph that knows when it is getting stale

Ten frameworks, staged deadlines, amendment omnibuses, standards moving through five draft stages — a static mapping is wrong within months. v1.5 gives every legal and standards node a lifecycle: `status` `lastVerified` `statusNote`, fed by monitored sources and community disputes. A node without a verification date is a claim, not a fact.

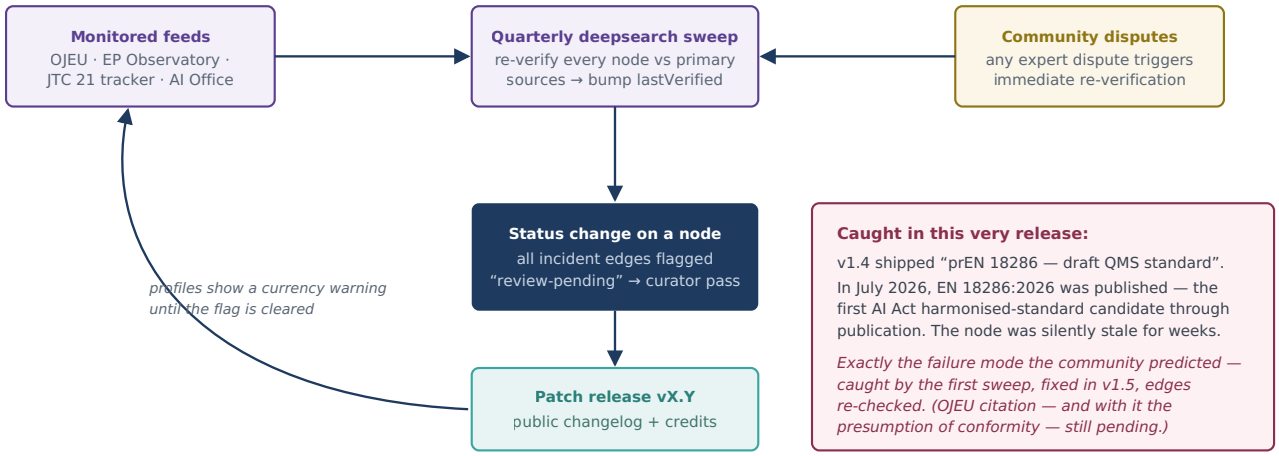


Fig. 4 — The currency loop: monitored feeds + scheduled sweeps + community disputes drive node-status changes; changes propagate to dependent edges; releases stay auditable. "No change" is also a finding — it bumps lastVerified.

RELEASE SUMMARY

What ships in v1.5 — and what the community decides next

Theme	What's new
Assurance structure	New node type **Control Objective** + edge types **operationalized_by** / **satisfied_by**; piloted on Art. 12 / 14 / 15 (6 objectives). Per-system control status binds to objectives, not raw articles.
Evidence trust	Verifiability ladder (self-asserted → tamper-evident → externally-anchored → independently-attested) on every evidence artifact; new threat *Audit-Trail Manipulation (Insider)*; new component *External Trust Anchor*; new regulation node *eIDAS 2* (Art. 45k qualified ledgers — statutory presumption of integrity).
Currency	*status* / *lastVerified* / *statusNote* on legal & standards nodes; monitored feeds; quarterly deepsearch sweeps; dispute-triggered re-verification; change propagation flags dependent edges. Live fix: prEN 18286 → **EN 18286:2026 (published)**.
Agentic gap-fill	New threat *Cascading Multi-Agent Failure* (incl. goal drift); new pattern *Shadow-Mode Execution*; *IMDA Agentic AI Governance Framework* (first state-issued agentic guidance) wired to Art. 14.
Standards backfill	prEN ISO/IEC 12792 (transparency taxonomy → Art. 13) · ISO/IEC 4213 (ML performance metrics → Art. 15) · JTC 21 mandate M/593 · ISO 42001 Annex-SL detail.
US health layer	HIPAA node wired to clinical use cases (ambient scribing, diagnostics, prior-auth) with evidence overlap (access logs, BAA chains).
Hardening	MCP-has-no-native-authn rationale on the gateway · Shadow-AI discovery on the risk register · Human-as-a-Tool escalation semantics · guardrail latency economics (sequential 300–800 ms vs parallel-to-slowest).

Open for community review in v1.6

Control objectives for the remaining obligations (Art. 9/10/11/13/17/26) · vendor & tooling landscape as a node type · five-layer SaaS reference blueprint · EUCS sovereignty levels · CCPA detector · ISO 42001 Annex A control count (38 vs 39 — first formal content dispute).

The ambition, restated

The best **community-validated** knowledge graph for regulated AI in Europe: every mapping voteable, every dispute visible, every release auditable. Design-time scoping *and* runtime assurance — one graph, both questions: *what must I build?* and *can I prove it still holds?*

184	416	6	4	v1.6
nodes	edges	regimes served by one log artifact	verifiability rungs	shaped by your disputes

Regulatory AI Navigator · knowledge graph v1.5 · August 2026 · companion to “Regulated AI in Europe & USA — A Practitioner's Guide” · community access: invite-only — comment or DM to join the validation round.