From: thomas barrett <tbarrett002@gmail.com>
Sent: Wednesday, July 15, 2026 6:07 AM
To: Comments
Subject: [EXT]: Public comment — an AI integrity assertion layer for audit evidence (Data and Technology project / standard-setting agenda)
Attachments: AACIF_Formal_Submission_Brief_v0_1.docx; Controls_Built_In_AACIF_v8-10.docx; AACIF_Companion_Note_AI_Integrity_Assertions.docx

Office of the Secretary
Public Company Accounting Oversight Board

Re: Public comment relevant to the Data and Technology research project and the standard-setting agenda — preserving audit-relevant evidence for AI-supported judgments

Dear PCAOB Staff,

I am submitting the AI Assertion and Chain Integrity Framework (AACIF) as public input to the Board's Data and Technology research project and the current standard-setting comment period, responding to questions raised in the staff's outreach on the integration of generative AI in audits and financial reporting.

The staff's outreach has framed the central question well: as technology-based tools, including generative AI, are used in areas relevant to audit quality, are existing evidence and documentation practices sufficient? My submission argues that the classical audit assertions remain necessary but are no longer sufficient by themselves once AI participates in the underlying accounting judgments, and it proposes a concrete, testable layer to close the gap.

When AI classifies transactions, estimates reserves, predicts impairments, interprets contracts, or drafts disclosures, the audit-relevant question is no longer only whether the entry exists, but how the system reached it and whether that basis is reconstructable. AACIF proposes an AI integrity assertion layer expressed in audit terms: source integrity, model identity and version integrity, prompt and instruction integrity, classification integrity, judgment integrity (whether the AI identifies a judgment as a judgment), estimate and valuation integrity, temporal integrity, cutoff and event-state integrity, chain-of-handoff integrity (whether uncertainty survives downstream transformation), human review integrity (whether review was genuine rather than ratifying), autonomy and authority integrity, reproducibility and evidence-preservation integrity, and drift integrity. Each maps to AACIF's seven assertions across a twelve-layer decision chain and is offered as an evidentiary-controls contribution, not as a proposed standard.

I would welcome the opportunity to discuss whether AACIF and this assertion layer are useful to the Data and Technology project, to inspection considerations around technology-assisted analysis, or to related standard-setting work.

Enclosed are a short companion note mapping the classical audit assertions to the proposed AI integrity assertions, the AACIF Formal Submission Brief, and the full working paper, Controls Built In: AACIF and

the Missing Black Box for AI Judgment.

Respectfully,

thom barrett
Living Life Press / AACIF Working Group
tbarrett002@gmail.com
Enclosures:
Companion note — Classical Audit Assertions and the AI Integrity Assertion Layer
AACIF Formal Submission Brief
Controls Built In — AACIF Working Paper

Classical Assertions and the AI Integrity Assertion Layer

A companion note to the AI Assertion and Chain Integrity Framework (AACIF) *thom barrett — Living Life Press / AACIF Working Group*

The problem

If accounting systems become AI-based, the old audit assertions do not disappear. But they are no longer sufficient by themselves.

The historical assertions were built for a world where the system processed transactions according to defined logic:

- completeness
- accuracy
- cutoff
- existence / occurrence
- rights and obligations
- valuation
- classification
- presentation and disclosure

In an AI-based accounting system, those still matter. But the control problem moves upstream and sideways. The AI is not only recording transactions. It may be classifying transactions, estimating reserves, identifying anomalies, drafting disclosures, predicting impairments, selecting accounting treatment, recommending accruals, interpreting contracts, changing workflows, learning from new data, using external sources, and switching models or tools.

So the audit assertions need a second layer: AI integrity assertions. These govern whether the AI's contribution to the accounting process is reliable, traceable, bounded, and explainable enough to support financial reporting. The old assertions still apply — but with AI you also have to ask how the system decided those things. That is where the new assertions come in.

The AI integrity assertions

1. Source Integrity

Question: Did the AI use authorized, complete, current, and reliable data sources?

This is the AI version of completeness and existence. An AI accounting system may pull from ERP data, bank feeds, contracts, invoices, emails, supplier systems, customer portals, market data, valuation feeds, legal databases, and operational systems. The control issue is not just whether the final journal entry exists — it is whether the AI relied on the right evidence.

Control examples: approved source registry; source authentication; source timestamp; source version; stale-source flag; unauthorized-source block; external-source reliability rating.

Plain English. *The AI should not be allowed to make accounting judgments from unknown or stale evidence.*

2. Model Identity and Version Integrity

Question: Which AI model made or influenced the accounting judgment?

This becomes critical if the system can use different models depending on task, cost, availability, or routing logic — one model interprets lease contracts, another estimates collectability, another drafts disclosures, another classifies expenses. If the model changes, the accounting result may change.

Control examples: model ID captured for every accounting conclusion; model version logged; model substitution rules; approved-model list; prohibited models for financial reporting; revalidation required after model change.

Plain English. *You cannot audit an accounting judgment if you do not know which model made it.*

3. Prompt and Instruction Integrity

Question: What was the AI instructed to do, and were those instructions authorized?

Traditional systems had configuration tables and business rules. AI systems have prompts, system instructions, guardrails, retrieval rules, and tool permissions. Those are now part of the accounting control environment.

Control examples: approved prompt library; change control over prompts; segregation of duties for prompt changes; prompt version captured with output; prompt override logging; testing for prompt injection or circumvention.

Plain English. *The prompt is now part of the accounting system configuration.*

4. Classification Integrity

Question: Did the AI correctly identify what kind of accounting item it was handling?

This is bigger than traditional classification. The AI may need to distinguish invoice vs. contract modification, lease vs. service agreement, revenue vs. deferred revenue, capex vs. opex, estimate vs. fact, obligation vs. contingency, subsequent event vs. current-period condition.

Control examples: confidence threshold for classification; required escalation for low confidence; independent review of high-risk classifications; periodic back-testing against human-reviewed samples; exception reports for classification changes.

Plain English. *The AI must not quietly turn judgment calls into routine classifications.*

5. Judgment Integrity

Question: When the AI makes or supports a judgment, is that judgment identified as a judgment?

Accounting contains many judgments: impairment, going concern, allowance for doubtful accounts, warranty reserves, tax positions, revenue recognition, lease classification, fair value estimates, contingent liabilities, materiality, disclosure adequacy. An AI system may make these look more precise than they are.

Control examples: label output as fact, estimate, inference, prediction, or judgment; require rationale for judgment; require evidence trail; require human approval for material judgments; prohibit autonomous posting of material judgment entries; require sensitivity analysis.

Plain English. *The AI must say when it is judging, not just calculating.*

6. Estimate and Valuation Integrity

Question: Are AI-generated estimates reasonable, supportable, and based on appropriate assumptions?

This extends traditional valuation. AI might estimate expected credit losses, inventory obsolescence, warranty liabilities, impairment, fair value, useful lives, customer churn affecting revenue, supplier failure affecting reserves, and litigation exposure.

Control examples: assumption registry; model back-testing; sensitivity ranges; source validation; management review; independent recalculation; benchmark comparison; drift monitoring.

Plain English. *If the AI produces an accounting estimate, the assumptions must be visible and challengeable.*

7. Temporal Integrity

Question: Was the AI using information current enough for the accounting decision?

This is the temporal assertion applied to accounting — supplier bankruptcy risk, customer collectability, market prices, legal developments, tax law changes, credit ratings, inventory demand forecasts, foreign exchange rates.

Control examples: data currency label; model training cutoff; retrieval currency; expiration date for AI conclusion; stale-data block; period-end freeze rules; subsequent-event detection.

Plain English. *The system must know whether it is using last month's world to make this month's accounting decision.*

8. Cutoff and Event-State Integrity

Question: Did the AI correctly understand the state of the event at period end?

Traditional cutoff asks whether the transaction was recorded in the right period. AI-based cutoff also asks whether the AI correctly understood the event state — for revenue recognition, shipment status, contract performance obligations, supplier failure, collectability, subsequent events, and impairment triggers.

Control examples: period-end data freeze; event-state timestamp; post-close data segregation; subsequent-event tagging; no silent use of post-period data in pre-period judgments; audit trail showing what was known at close.

Plain English. *The AI must not use future knowledge to justify a past-period accounting conclusion unless it is properly treated as a subsequent event.*

9. Chain-of-Handoff Integrity

Question: Did the accounting meaning survive as the AI output moved through systems?

This is where AACIF matters directly. The AI says: “There is a 68% likelihood this supplier will fail within 90 days.” Downstream that becomes “supplier unstable,” then “alternative supplier required,” then “inventory reserve needed,” then “material supply-chain uncertainty.” Each step may be reasonable, but the original uncertainty may disappear.

Control examples: preserve confidence intervals; preserve source labels; preserve judgment labels; preserve version and date; log transformations; require approval when probability becomes binary classification; require cumulative change report.

Plain English. *The AI's uncertainty must not get stripped away as its output moves through the accounting chain.*

10. Human Review Integrity

Question: Was human review real?

This is not solved by saying “human in the loop.” For accounting, real review means the reviewer had the AI output, the supporting evidence, the uncertainty, the source trail, the authority to override, enough time, enough competence, and no incentive to rubber-stamp.

Control examples: reviewer certification; evidence-viewed log; override capability; override tracking; time-spent analytics; reviewer independence; escalation for material judgments; prohibition on summary-only review for material items.

Plain English. *A reviewer who only sees the AI's conclusion is not reviewing. They are accepting.*

11. Autonomy and Authority Integrity

Question: What was the AI allowed to do?

AI accounting systems may move from recommend, to draft, to classify, to post, to adjust, to close, to disclose. Those are radically different authority levels.

Control examples: authority ladder; role-based AI permissions; no autonomous posting of material entries; no autonomous disclosure changes; approval required for estimates; automated blocking outside authorized scope; runtime monitoring of tool use.

Plain English. *An AI allowed to suggest an entry is not the same as an AI allowed to post one.*

12. Reproducibility / Evidence Preservation Integrity

Question: Can we reconstruct the accounting conclusion later?

For static systems, you test configuration and transaction logs. For AI systems, you need a preserved decision record: model used, prompt used, sources retrieved, source versions, data timestamps, output, confidence, assumptions, reviewer, downstream action, and exception status.

Control examples: decision provenance capsule; immutable audit log; retained prompts and outputs; retained retrieval package; model fingerprint; source snapshot; evidence hash; preservation through close and audit period.

Plain English. *If the AI conclusion affects the financial statements, the company must preserve the evidence package that existed when the conclusion was made.*

13. Drift Integrity

Question: Has the AI changed in a way that affects accounting reliability?

Accounting systems historically changed through formal change management. AI systems may change through model updates, vendor endpoint updates, fine-tuning, retrieval changes, prompt changes, new data sweeps, feedback loops, changed routing logic, and changed source availability.

Control examples: drift monitoring; baseline comparison; mandatory revalidation triggers; model-change alerts; source-change alerts; performance degradation reports; accountingimpact assessment after model change.

Plain English. *The company must know when the AI has become meaningfully different from the system that was approved.*

The revised assertion set

If accounting systems become AI-based, I would not discard the old assertions. I would layer new ones around them. The mapping below pairs each classical assertion with the AI integrity assertion(s) that most directly extend it; two assertions cut across all of them.

Classical financial statement assertion	Corresponding AI integrity assertion(s)
Completeness	Source Integrity; Chain-of-Handoff Integrity
Accuracy	Model Identity & Version Integrity; Estimate & Valuation Integrity; Reproducibility / Evidence Preservation
Cutoff	Temporal Integrity; Cutoff & Event-State Integrity
Existence / Occurrence	Source Integrity; Classification Integrity
Rights and Obligations	Classification Integrity (contract interpretation); Autonomy & Authority Integrity
Valuation	Estimate & Valuation Integrity; Judgment Integrity
Classification	Classification Integrity
Presentation and Disclosure	Judgment Integrity; Human Review Integrity; Reproducibility / Evidence Preservation
(Cross-cutting — apply to all)	Prompt & Instruction Integrity; Drift Integrity

This assertion layer is the accounting-facing expression of AACIF, a standardized record for preserving the evidentiary basis of an AI-supported judgment across the decision chain.

AACIF

AI Assertion and Chain Integrity Framework

Formal Submission Brief

A boundary-carried assertion record for consequential AI systems

Submitted for technical review, standards consideration, and pilot testing

Author	thom barrett
Affiliation	Living Life Press / AACIF Working Group
Version basis	Controls Built In - AACIF Working Paper v8-13, July 2026
Submission version	v0.1
Date	July 8, 2026
Contact	tbarrett002@gmail.com

Submission request: Consider AACIF as a candidate control profile, technical report, recommended practice, schema extension, assurance module, or pilot framework for AI assertion records, chain integrity, human-review evidence, and consequential AI decision reconstruction.

1. Purpose

AACIF proposes a standardized AI Assertion Record for consequential AI systems. The framework addresses a specific control gap in current AI governance: existing governance regimes increasingly require risk management, transparency, human oversight, technical documentation, logging, monitoring, and accountability, but they often assume that a reliable evidentiary record already exists. In many consequential AI deployments, it does not.

AACIF is designed to close that gap. It requires a consequential AI system to declare, at each material boundary, what it knew, on what basis, under what limits, with what uncertainty, and whether those declarations survived to the point where someone relied on the output. AACIF does not decide whether the AI system was right. It makes the AI-supported judgment declared before reliance and reconstructable after the fact.

2. The Control Gap

Before AI, consequential judgment usually remained with a human decision-maker. Systems supplied data, calculations, records, and workflow support, but the decision itself was subject to human controls: authority limits, review, sign-off, rationale, escalation, accountability, and later questioning.

AI changes the location of judgment. The model may now score, rank, deny, approve, flag, route, summarize, recommend, or trigger action. The decision has moved into the technical layer, but the controls historically attached to human judgment did not move with it.

> **Core gap:** Current AI documentation may describe the model, vendor, procurement posture, or governance process, but it often does not reconstruct the specific AI-supported judgment at the point of reliance. The missing object is a boundary-carried assertion record.

3. AACIF Core Proposal

AACIF translates the reconstructability requirement into seven assertions. The assertions are intended to be structured, attributable, machine-readable where feasible, testable, and preserved across system handoffs.

Assertion	Required declaration	Control purpose
A1 Temporal	Knowledge vintage, cutoff, retrieval date, source currency	Prevents stale information from appearing current.
A2 Knowledge State	Model version; rule, policy, or guideline version	Shows which operative rule set was used.
A3 Population	Training population, validation population, excluded or under-tested cohorts	Tests whether the system is fit for the person or case.
A4 Measurement	Evidence basis, validation method, confidence interval, error rate, independent corroboration	Distinguishes evidence from repetition.
A5 Language / Perspective	Source language, jurisdiction, cultural frame, institutional perspective	Identifies narrow viewpoints mistaken for consensus.
A6 Translation	Thresholds, score-to-action rules, transformation logic, decision conversion	Shows how uncertainty became action.
A7 Chain Integrity	Handoff survival, field preservation, field loss, receiving system, audit link	Confirms whether A1-A6 survived to reliance.

A7 is the central integrity control. It does not add another substantive claim. It asks whether the other six declarations survived the chain.

4. The Twelve-Layer Decision Chain

AACIF applies the seven assertions across a twelve-layer AI decision chain. This matters because many AI failures occur between systems rather than inside a single component.

Layer	Description	Required record focus
0	Deployment envelope	Intended purpose, authority level, constraints, hard stops, escalation.
1	Corpus	Source composition, provenance, contamination, uncertainty.
2	Training	Training period, reinforcement choices, known failure modes.
3	Model	Identifier, version, release date, supersession status.
4	Instruction	System prompts, guardrails, permissions, prohibited actions.
5	Retrieval	Live documents, source currency, verification status, omissions.
6	Inference	Prompt, context, parameters, tools, confidence, opacity.
7	Output	Recommendation, score, summary, instruction, attached declarations.
8	Integration	Transformation into enterprise workflow or downstream fields.
9	Human review	What reviewer saw, authority to override, time, competence, rationale.
10	Action	Final actor, final decision, AI contribution, execution path.
11	Audit / recovery	Replay path, retained evidence, tamper evidence, reconstruction capacity.

Layer 5 is dynamic. For retrieval-augmented and agentic systems, the record must declare not only the model artifact, but also the live documents, records, or data injected at inference time. A well-documented model can still act on stale, privileged, contaminated, or unverified material moments before action.

5. Relationship to Existing Frameworks and Artifacts

AACIF is not intended to replace existing AI governance frameworks. It is intended to supply the evidentiary substrate they require. The distinction is that AACIF follows the output through runtime reliance and later reconstruction.

Framework / artifact	Primary function	AACIF contribution
NIST AI RMF	AI risk governance, mapping, measurement, and management.	Provides decision-level evidence that risk controls operated across the chain.
ISO/IEC 42001	AI management system governance.	Supplies control records for operating effectiveness and management-system evidence.
EU AI Act	Legal obligations for high-risk AI, transparency, technical documentation, record keeping, and oversight.	Supports technical documentation, logging, human oversight, monitoring, and conformity evidence.
Model cards / system cards	General model or system description.	Carries relevant limitations into the actual decision chain.
ML-BOM / AI-BOM	Model, data, dependency, and provenance manifest.	Extends artifact transparency into runtime reliance and action.
SOC 2 / audit frameworks	Control design and operating effectiveness.	Creates an assertion record auditors can test.
AI assurance tools	Testing, documentation, and governance review.	Adds reconstructability and assertion-survival testing.

Positioning: ML-BOM is the manifest. AACIF is the runtime chain-of-reliance record. Existing frameworks tell organizations what they must govern; AACIF specifies the record that lets them prove what happened.

6. Human Review as a Testable Control

AACIF treats "human in the loop" as insufficient unless it can be tested. A human review step is a genuine control only if the reviewer can see the relevant assertions, understand them, has time to evaluate them, has authority to override, is independent from pressure to accept the AI output, and preserves a rationale with the final action record.

Criterion	Control question
Visibility	Could the reviewer see the relevant assertions, limits, uncertainty, and provenance?
Competence	Was the reviewer qualified to interpret the assertion record and domain limits?
Time	Did the workflow allow real review rather than rubber-stamp approval?
Authority	Could the reviewer override, reject, escalate, or suspend the AI-supported result?
Independence	Was the reviewer free from pressure or penalty for disagreeing with the AI output?
Record	Was the reviewer rationale preserved with the final action record?

If these conditions are absent, the review step is not a control. It is a ratification step. For systems that decide, act, or chain outputs into other automated systems, AACIF also contemplates hard stops where declared tolerance thresholds are exceeded.

7. Why Standardization Is Needed

The components needed to implement AACIF already exist: structured metadata, event logs, cryptographic tamper evidence, append-only records, schema validation, identity and access controls, provenance tracking, software bills of materials, model cards, ML-BOMs, and audit workflows.

The gap is the absence of a shared requirement and shared record shape. Without standardization, every vendor, buyer, regulator, insurer, auditor, and assurance provider will define its own private AI evidence questionnaire. That produces inconsistency, weak comparability, friction, and avoidable cost.

A standardized AI Assertion Record would allow vendors to declare in structured form, buyers to evaluate reliance consistently, auditors to test record survival, insurers to price bounded exposure, regulators to examine technical documentation and post-deployment evidence, and affected persons or their representatives to contest consequential decisions more effectively.

8. Proposed Standards Work

AACIF is submitted for consideration as one or more of the following:

- a control profile for consequential AI systems;
- a technical report on AI assertion records and chain integrity;
- a recommended practice for preserving AI decision evidence across system boundaries;
- a schema extension to existing AI documentation and ML-BOM artifacts;

- an assurance module for testing reconstructability and human-review evidence;
- a pilot framework for evaluating high-consequence AI decision chains.

The proposed standards work would define required assertions by authority tier; required fields for each assertion; retention and tamper-evidence expectations; links between static model documentation and runtime records; retrieval-time evidence requirements; human-review test criteria; score-to-action translation records; chain-break reporting; and the process for establishing domain-specific tolerance values.

9. Requested Review

Requested review question: When a consequential AI system produces an output that someone will rely on, should the system carry a structured assertion record declaring what it knew, on what basis, under what limits, with what uncertainty, and whether that record survived to the point of reliance?

If that requirement is valid, the next work is specification. AACIF is offered as a starting point for that specification.

10. Initial Pilot Proposal

AACIF can be piloted against one high-consequence or simulated decision chain. The pilot does not require production deployment data. It can use a representative workflow, public case facts, or a controlled synthetic record.

Pilot element	Description
Candidate use cases	Healthcare denial or prior authorization; hiring-screening rejection; bank fraud flag; public-benefits eligibility; AI agent connected to email, HR, claims, or compliance systems.
Method	Map the twelve layers; apply A1-A7; record pass, warning, fail, or not applicable for each cell; identify chain breaks.
Output	Layer-by-assertion heat map; reconstructed decision record; evidence-break report; comparison with existing documentation.
Value	Shows what AACIF reveals that model cards, system cards, ML-BOM artifacts, procurement questionnaires, and generic audit logs do not show by themselves.

11. References and Standards Anchors

[1] AACIF Working Paper: Controls Built In - AACIF and the Missing Black Box for AI Judgment, v8-13, July 2026.

[2] NIST AI RMF: National Institute of Standards and Technology, AI Risk Management Framework. https://www.nist.gov/itl/ai-risk-management-framework

[3] ISO/IEC 42001: International Organization for Standardization, ISO/IEC 42001:2023 Artificial intelligence - Management system. https://www.iso.org/standard/42001

[4] EU AI Act: European Commission, Regulatory framework on artificial intelligence. https://digital-strategy.ec.europa.eu/en/policies/regulatory-framework-ai

[5] EU AI Act standardization: European Commission, Standardisation of the AI Act. https://digital-strategy.ec.europa.eu/en/policies/ai-act-standardisation

[6] CycloneDX ML-BOM: CycloneDX, Machine Learning Bill of Materials (AI/ML-BOM). https://cyclonedx.org/capabilities/mlbom/

[7] INCITS/AI: INCITS Artificial Intelligence Committee, U.S. TAG to ISO/IEC JTC 1/SC 42. https://www.incits.org/committees/ai

[8] IEEE AI Standards Committee: IEEE Computer Society Artificial Intelligence Standards Committee. https://sagroups.ieee.org/ai-sc/

[9] CEN-CENELEC JTC 21: CEN-CENELEC Artificial Intelligence standardization and JTC 21. https://www.cencenelec.eu/areas-of-work/cen-cenelec-topics/artificial-intelligence/

Document note: This brief is a standards-facing summary of the AACIF working paper. It is not a legal opinion, certification claim, or assertion that AACIF has been adopted by any standards body.

CONTROLS BUILT IN

AACIF and the Missing Black Box for AI Judgment

A working paper. Presented for challenge, refinement, and implementation testing.

AACIF Working Group

Version v8

July 6, 2026

tbarrett002@gmail.com

CONTROLS BUILT IN

The control every mature system builds in — and the one place it wasn't

thom barrett | Living Life Press | 2026

A working paper. Presented for challenge, refinement, and implementation testing.

Abstract

Every mature system that delegates consequential work to a part it does not fully control builds two controls into the boundary between them: a declaration before reliance, and a record after it. Investment managers require this of outside portfolio managers. Software platforms require it of connected applications. Aviation requires it of every aircraft before it flies. AI is the single exception. Consequential AI systems — systems that deny medical coverage, screen employment candidates, allocate credit, and route public benefits — currently operate without a standardized declaration of what they knew before acting and without a record that survives to the point where someone relies on their output.

This paper proposes a framework to close that gap. The AI Assertion and Chain Integrity Framework — AACIF — requires a consequential AI system to declare, at each boundary it crosses, what it knew: the vintage of its knowledge, the population it was built on, the method behind its claims, the uncertainty it carries, and the limits its builder accepted. It requires that declaration to survive every handoff to the point of action, in a form that a clinician, an auditor, an insurer, a court, or a safety engineer can later read and assess. The framework does not decide whether the AI was right. It makes the AI's judgment declared in advance and reconstructable after the fact. Those are different from being right — and they are the things that make being wrong survivable.

AACIF is organized around seven boundary assertions applied across a twelve-layer chain, with Layer 0 — the developer's pre-deployment intentionality record — governing every layer that follows. It is a control framework, not a governance framework. It does not replace NIST, ISO 42001, the EU AI Act, or COSO. It produces the evidentiary substrate those frameworks assume already exists and have never required anyone to build.

Working Thesis

Consequential AI systems require a boundary-level assertion record — declaring what the system knew, on what basis, and for whom — that travels with every output from construction to inference, integration, human review, action, and audit. Without that record, governance operates without evidence, audit operates without a trail, insurance underwrites an unknown, and affected persons cannot contest decisions made about them. The declaration is the preventive control. The record is the detective control. Together they are the architecture every mature system already uses — and the one architecture AI has never been required to implement.

The Problem This Paper Addresses

In the spring of 2022, a 91-year-old man named Gene Lokken was nineteen days into physical therapy ordered by his physicians when his Medicare Advantage insurer cut off his

coverage. The decision was not made by a physician. It was made by an algorithm called nH Predict. When his physicians appealed, they lost. In the Medicare Advantage skilled-nursing-facility context, federal oversight has found that fewer than one percent of affected beneficiaries file appeals, even though more than ninety percent of appealed denials are overturned by independent reviewers. Gene Lokken died on July 17, 2023, still in the facility, still in the chair. His family paid approximately $150,000 out of pocket for care his insurer's algorithm had determined he no longer needed.

The algorithm was wrong nine times out of ten when challenged. The insurer knew that. It also knew almost no one would challenge it.

The governance failure in that case is not difficult to name. No one required nH Predict to declare what population it was trained on, whether Gene Lokken was within that population, what evidence supported its determination, what version of clinical criteria it was applying, or what confidence it assigned to its output. No independent party could reconstruct what the system knew at the moment it acted. The human review step had been designed to ratify the algorithm's output rather than assess it — reviewers were disciplined for deviating. The only place independent judgment entered the process was at the level of federal administrative law judges, entirely outside the insurer's control, where the algorithm was overturned more than ninety percent of the time.

The same year, in clinics across the country, another algorithm was listening. DAX Copilot, an ambient AI scribe now deployed across more than four hundred health systems and ten million patient encounters, records the conversation between doctor and patient and drafts the clinical note. Pilot data behind the tool showed fewer than five percent of its notes independently reviewed for safety, an eighteen percent omission rate, and hallucinated content in more than one note in nine — and patients are rarely told which model listened to them, what version drafted their chart, or that the recording is deleted by design after thirty days. Where the system's corporate parent also owns the algorithm that decides whether the visit gets paid for, the conversation a patient believed was confidential becomes an input to a decision they never see coming. AACIF found the same absence nH Predict had: no temporal declaration, no model version, no confidence interval that survives the handoff into the record.

In employment, the algorithm does not deny care — it denies consideration before a human ever sees the name. Derek Mobley, a Black man in his forties with documented disabilities, applied to more than a hundred positions screened by Workday's hiring AI and was rejected by all of them. A federal court certified his case as a class action in May 2025. Workday's tools screen roughly 1.1 billion applications a year across more than sixty percent of the Fortune 500, and fewer than one rejected candidate in ten is ever told an algorithm made the call. AACIF found no population declaration — no disclosure of who the model was validated on, and no way for a rejected applicant to know whether it was ever tested on someone who looked like them.

In Michigan, the algorithm accused people of a crime. The state's automated fraud detection system for unemployment claims, MiDAS, flagged tens of thousands of claimants between 2013 and 2015, at a false-positive rate later estimated near ninety percent. Families lost

homes, filed bankruptcy, and in documented cases attempted suicide over debts the state's own system could not substantiate. Named plaintiffs eventually forced the Michigan Supreme Court to rule, in 2020, that the state could be sued for the process it never provided. AACIF found what it always finds where no record survives: no chain connecting an accusation to the evidence behind it, and no way to reconstruct, after the fact, what the system actually knew about the people whose lives it had already emptied out.

That pattern — consequential AI output without declared basis, without surviving record, without genuine human review, without reconstructability — is not an anomaly of one insurer or one algorithm. It is the same failure in healthcare, in hiring, and in public benefits alike. Each system individually satisfies every current governance requirement. Each fails because no existing framework requires what AACIF requires: a structured, boundary-carried assertion record that travels with a consequential AI judgment from construction to the point where someone relies on it.

The gap is not technical. Every component needed to build that record exists. Event logging, cryptographic tamper evidence, structured metadata, append-only storage, stream processing against typed schemas — these are solved engineering problems with decades of implementation history. The banking industry's own migration to the ISO 20022 messaging standard is a working example: a bank statement now carries a mandatory, machine-checked creation timestamp, and a gateway automatically rejects a payload if its date falls outside the active processing window — the exact temporal check A1 asks AI to perform, already running in production, at global scale, today. The gap is the absence of a requirement — and that absence is starting to close, unevenly and more slowly than it first appeared. The EU AI Act's Annex IV would require exactly this kind of build-time documentation for high-risk systems, but as of mid-2026 its enforcement date for the systems this paper is about — healthcare, hiring, benefits — has already slipped once, from August 2026 toward December 2027, with the technical standards implementing it still unfinished. No procurement framework outside a narrow set of federal contexts has demanded it. The entities that would bear the cost of building it have been the primary authors of the frameworks that govern whether it needs to be built.

That is the fox and the chicken coop. AACIF is the lock.

Working Group

This paper is presented as a working hypothesis for challenge and refinement by a group of colleagues whose combined experience spans the domains the framework must hold up in: financial services controls, structured disclosure standards, continuous auditing, corporate governance, neuroscience, and embedded systems engineering. Several members of this group worked together for decades inside PricewaterhouseCoopers and Coopers & Lybrand before this project existed. The framework did not assemble them. Their shared history assembled the framework.

thom barrett* — Author. Retired Partner, PricewaterhouseCoopers.*

Barrett joined Coopers & Lybrand in 1978 as part of a new group focused on EDP auditing, at a time when that specialty was still being invented. His first assignments were not audits

— he installed enterprise-wide accounting systems for the State of Maryland, the State of Kentucky, and the City of Pittsburgh, giving him a systems builder's understanding of where data originates, how it flows, and where controls fail. From 1982 to 1984 he developed deep expertise in the Carrier Access Billing System created by the AT&T divestiture, becoming a frequent conference speaker on CABS. From 1984 to 1986 he was seconded to C&L Australia, conducting application and implementation reviews for mid-platform systems.

On returning to the United States in 1986 he focused exclusively on financial services for the remainder of his career. He became the subject matter expert on operational controls in transfer agency, fund administration, trading operations, compliance, and securities — the SAS 70 audit universe that governed how service organizations demonstrated their controls to clients and auditors. Also in 1986 he founded the Research Support Group within Coopers & Lybrand, which began as a pricing engine for field auditors and evolved to include the scanning and interpretation of financial statements before commercial equivalents existed. The application supporting the working audit platform was called OTTO — which stood for nothing. RSG was an early attempt to automate the extraction and validation of financial information from documents: the same problem AACIF now asks AI systems to solve for epistemic declarations.

He contributed to C&L's work on the COSO internal controls framework as part of a multi-disciplinary team spanning technical accounting, IT, and audit practice, working under Vin O'Reilly, with a focus on integrity controls. He wrote for and edited C&L Perspectives and other internal publications. In 1996 he served on President Clinton's Commission on Critical Infrastructure Protection — the first federal effort to map systemic vulnerabilities in networked digital systems across banking, telecommunications, power, and information infrastructure.

He served as Client Services Partner from 1988 to 2012. From 2001 to 2005 he served on PwC's Innovation Working Group alongside Mike Willis. During this period he introduced the concept of XBRL to the SEC and worked with the Investment Company Institute and the SEC to shape how the standard would apply to mutual fund reporting, successfully advocating for risk and expense schedules rather than full financial statements — a policy outcome that made the standard workable for the industry. From 2005 to 2008 he served as Global Partner in Charge of XBRL Services, leading PwC's US business responsibilities for the standard's rollout and regulatory adoption; Willis led the technical side of that work. From 2005 to 2009 he served on PwC's Global Financial Services Leadership Team. From 2008 to 2012 he served as Partner in Charge of the Market Information and Data Analysis practice, building valuation infrastructure for complex and illiquid instruments during and after the financial crisis. During the early 2000s he was a frequent speaker in India on datacenter and service center reviews as Indian operations became critical outsourced infrastructure for global financial services.

He retired from the PwC partnership in 2012 and now works independently under Living Life Press from Sóller, Mallorca. AACIF is forty years of the same question asked in different domains — when a consequential system acts, what record survives that proves what it knew?

Michael Willis* — Retired Partner, PricewaterhouseCoopers. Founding Chair, XBRL International. Former Associate Director, SEC Division of Economic and Risk Analysis.*

Willis spent more than thirty years as a partner at PricewaterhouseCoopers, where he led data analytics engagements and served as an advisor on development of PwC's analytical platforms using standardized data and presentation abstractions to automate access, validation, and analysis of data from disparate sources. He served alongside Barrett on PwC's Innovation Working Group from 2001 to 2005 and as the technical lead on the XBRL team from 2005 to 2008, with Barrett holding the US business leadership role. Together they led the work to introduce XBRL to the SEC and shape its application to the mutual fund industry, successfully advocating for risk and expense schedules rather than full financial statements for mutual fund reporting. He served as founding Chair of XBRL International — the organization that built the international standard for machine-readable structured financial disclosure. In 2015 he joined the SEC's Division of Economic and Risk Analysis, where he led the Office of Structured Disclosure and in 2020 was named Associate Director leading DERA's Office of Data Science and Innovation. Since 2022 he has served as Vice Chair and Chairperson of the Regulatory Oversight Committee. His relevance to AACIF is direct: XBRL is the closest existing analogue to what the framework proposes for AI — a standardized, boundary-carried, machine-readable declaration that travels with a consequential output so that any downstream party can assess its basis independently.

J. Donald Warren Jr., PhD* — Retired Partner, PricewaterhouseCoopers. Ben J. Rogers Chair and Professor of Accounting, Director of the School of Accounting, Lamar University.*

Warren led the Computer Audit Group at PricewaterhouseCoopers and was instrumental in its expansion, with Barrett serving as a partner and member of that group. He retired from PwC after thirty-one years, during which he served as Partner-in-Charge of Computer Assurance Services and Business Assurance Technology Services, Director of Business Assurance Product Development and Technology Services for Coopers & Lybrand, and as a National Accounting, Auditing and SEC Consulting Partner. He holds a PhD in Accounting from Texas A&M University and is the co-author of the third edition of the Handbook of IT Auditing. He serves as Associate Editor for the Journal of Information Systems, as Research Fellow for the Rutgers Business School Continuous Auditing and Reporting Laboratory, and on the Advisory Committee for the Continuous Auditing and Reporting Laboratory Symposia. He directed the Center for Continuous Auditing at Rutgers Business School. His relevance to AACIF is foundational: continuous auditing — the field Warren helped build — is the intellectual predecessor of what the framework requires. AACIF's assertion record is continuous auditing applied to the epistemic state of AI systems rather than to financial transactions.

Scott E. Eston* — Retired Chief Operating Officer and Executive Committee Chairman, Grantham Mayo Van Otterloo (GMO). Independent Trustee, Eaton Vance Funds.*

Eston holds a B.A. in Accounting from Michigan State University and began his career at Coopers & Lybrand, where he worked alongside Barrett on RSG and OTTO — the early automated audit and document processing tools — and on shared client engagements where Eston served as the assurance partner and Barrett as the technology partner. He also

served alongside Barrett as part of C&L's multi-disciplinary team contributing to the development of the COSO internal controls framework. That division of responsibility throughout their shared career — independent assurance and technology expertise working the same engagement — is a lived demonstration of the segregation of duties principle AACIF now formalizes as a framework requirement. After C&L he served as Chief Operating Officer and Executive Committee Chairman at GMO, the Boston-based asset management firm, for twelve years before retiring in 2020. He has served as an independent trustee across a large number of Eaton Vance closed-end funds. His relevance to AACIF is both historical and structural: he was present at the origin of Barrett's automated audit work in the 1980s, and his career as a COO and independent fiduciary is the direct embodiment of the manager due diligence analogy at the core of the framework.

John W. Stadtler* — Retiring Partner, Governance Insights Center and Financial Services, PricewaterhouseCoopers.*

Stadtler joined PwC in 1988 and has served as a partner since 1999, working alongside Barrett on client engagements from 1988 through 2012 — twenty-four years of shared client service. He was named leader of PwC's U.S. Financial Services Industry Practice in 2014, overseeing more than 500 partners across banking, insurance, and asset management. His areas of expertise include SEC assurance, regulatory matters, financial reporting, corporate strategy, and service provider operations. He has served as Partner in PwC's Governance Insights Center, working directly with boards of directors, audit committees, investors, and management teams on governance challenges, and has published specific guidance on audit committee oversight of AI — including questions about how organizations develop, deploy, validate, and monitor AI models and how humans are kept in the loop to validate outcomes. He retired from PwC on June 30, 2026. His relevance to AACIF is precise: Stadtler sits at the intersection of financial services audit and board-level AI governance that the framework is designed to reach, and twenty-four years of shared client work with Barrett means he understands the controls thinking behind the framework from the inside.

Matthew Chafee, PhD* — Professor of Neuroscience, University of Minnesota. Faculty, Graduate Program in Neuroscience and MnDRIVE Brain Conditions Program.*

Chafee is a Professor in the Department of Neuroscience at the University of Minnesota, where his research focuses on the neural basis of spatial cognition and working memory and on cortical system dysfunction in psychiatric disease. He has an h-index of 25, with more than 45 peer-reviewed publications receiving over 3,700 citations, and prior research affiliations at Yale University and the United States Department of Veterans Affairs. His primary research areas include prefrontal cortex function, posterior parietal cortex, working memory under cognitive load, and the neural mechanisms underlying judgment and causal reasoning. His relevance to AACIF is the framework's hardest open question: whether the human review layer is governable by boundary contracts given realistic operational conditions — time pressure, automation bias, high-confidence outputs from authoritative-seeming systems, and incentive structures that reward throughput over independent judgment. That question lives in Chafee's research domain. His work on working memory and prefrontal circuit function under adverse conditions is the scientific

basis for assessing whether genuine human review is achievable in the environments where AACIF requires it.

Amy Larson, PhD* — Faculty, Department of Computer Science and Engineering, University of Minnesota. Faculty, University of Minnesota Software Engineering Center.*

Larson is a faculty member in Computer Science and Engineering at the University of Minnesota, where she teaches and researches real-time and embedded systems, software development for safety-critical environments, and artificial intelligence. Her courses include Real-Time and Embedded Systems, Software Development for Real-Time Systems, and Introduction to Artificial Intelligence. Her research background includes distributed heterogeneous systems, autonomous robotics, and sensorimotor systems for autonomous operation. She is affiliated with the University of Minnesota Software Engineering Center and its Master of Science in Software Engineering program. Her relevance to AACIF is architectural: real-time and embedded systems is the engineering domain where the concept of a pre-defined operational envelope with hard stops, constraint boundaries, and graduated responses to boundary violations is standard practice, not theoretical. A medical device, an aircraft control system, or an autonomous robot cannot operate outside its declared operating parameters — that constraint is built into the architecture before deployment. Layer 0 of AACIF is that concept applied to AI. Larson's background grounds it in proven engineering practice and gives the framework the technical credibility the Layer 0 specification requires.

tbarrett002@gmail.com | Living Life Press | Sóller, Mallorca | 2026

CONTROLS BUILT IN

The control every mature system builds in — and the one place it wasn't

thom barrett | Living Life Press | 2026

A working paper. Presented for challenge, refinement, and implementation testing.

The Argument

Controls built into systems are not new. They are not novel, not theoretical, and not something AI governance has to invent. Every system that hands consequential work to a part it does not fully control builds a control into the boundary between them.

Aviation has the flight data recorder. Finance has the ledger and the audit trail. Medicine has the clinical record. Those are recording controls — they make a decision reconstructable after it is relied on. But there is an older and quieter control that comes before the recording. Before an investment house gives a portfolio to an outside manager, it makes the manager declare what it does, how it does it, and whether it can sustain it — and it does this before a dollar moves. Before a connector reaches your email, the platform has already vetted it and scoped what it can touch. Before software runs, the operating system checks its version and refuses what no longer fits. In each case the control is built into the boundary, and it operates before the harm, not after it.

AI has many fragments of documentation and logging, but no standard assertion record that travels with a consequential judgment from construction to inference, integration, human review, action, and audit. It has no declaration, made before it runs, of what it was built on and where it should not be trusted — and no record that survives to the point where someone relies on it. We scrape, we train, we deploy, and we wire the output to a decision — without a portable record and without a declared basis.

That is the gap this paper addresses. Controls Built In means exactly what it says. A control is built in when the builder declares, before the system runs, what it knows and what it does not — and when that declaration is recorded so it survives to the point where someone relies on it. The declaration is the preventive control when it is structured, attributable, testable, and enforced at the boundary. The record is the detective control that proves what was declared, what changed, and whether the declaration survived to reliance.

AACIF — the AI Assertion and Chain Integrity Framework — does two things. It requires a consequential AI system to declare, at each boundary it crosses, what it knew: the vintage of its knowledge, the population it was built on, the method behind its claims, the uncertainty it carries, and the limits its builder accepted. And it requires that declaration to survive every handoff to the point of action, so that anyone who later needs to — a clinician, an auditor, an insurer, a court, a safety engineer — can stand where the system stood and see what it saw.

The framework does not decide whether the AI was right. It makes the AI's judgment declared in advance and reconstructable after the fact. Those are different from being right — and they are the things that make being wrong survivable.

I. Two Controls, One Boundary

A control that operates before the event is preventive. A control that operates after it is detective. AACIF is both, and the two are not the same thing wearing one name.

The preventive control is the declaration. Before a consequential system is deployed, its builder must state what it considered and what it did not — and for each thing left out, why. If the population the model was trained on does not matter for this use, say why. If the model's version need not be carried forward, say why. If the cultural frame of the sources was never examined, say why. These are not questions to answer after an incident. They were answered, or should have been, when the thing was built. A builder who cannot say why a dimension does not matter has just revealed, on the record and before deployment, that the dimension was never considered. That is the control: it forces the reasoning into the open before the system runs, and it makes the omission the builder's to own.

What that declaration actually looks like is easiest to see against systems already in the field. None of the four below ever produced a document like this. Reconstructed from the public record — the litigation, the pilot data, the regulatory findings — here is what an honest Layer 0 declaration could have said, if the builder had been required to say it before the system ran.

nH Predict. *Population: trained on retrospective claims and utilization data from post-acute skilled-nursing patients, concentrated in cases with a modal recovery trajectory; not validated against patients over ninety, patients with multiple comorbid diagnoses, or recoveries that diverge from the median case — populations that include Gene Lokken. Version: the clinical criteria set underlying a prediction is versioned internally and updated on a rolling basis; the version is not carried into the denial letter or preserved in a form the treating physician or beneficiary can inspect. Confidence: length-of-stay predictions are presented as a single number; no confidence interval, error margin, or override rate is disclosed to the reviewing coordinator. Human review: coordinators are evaluated in part on alignment with the model's predicted length of stay, and are subject to review for deviation — a condition that makes the review step a ratification, not an assessment, and that fact is not disclosed anywhere the beneficiary or their physician could see it.*

DAX Copilot. *Population: trained primarily on English-language clinical encounter transcripts; validation depth is not declared as equivalent across specialties, and is not declared at all for non-English-dominant encounters. Version: the ambient model updates on a rolling basis; the specific version that drafted a given note is not attached to that note in the medical record. Confidence: internal pilot testing found hallucinated content in roughly one note in nine and omitted clinically relevant content in roughly one note in six; neither figure is disclosed to the physician at the point of signing, nor to the patient whose visit it is. Chain: once signed, the note enters the permanent record with no marker distinguishing AI-drafted language from clinician-authored language, and the underlying audio is deleted on a fixed schedule regardless of whether a dispute is pending.*

Workday / HiredScore. *Population: the scoring model is trained on each client's own historical hiring and performance data; the demographic composition of that historical population, and whether it reflects prior human hiring bias, is not audited or disclosed at deployment. Version: model updates roll out across the client base; the specific version that screened a given applicant is not retained in that applicant's file. Fitness: no declaration exists as to whether the model was validated on applicants over forty, applicants with a disclosed disability, or resumes that deviate in format from the training population's norm — the exact profile of the rejected candidate the framework is built to protect.*

MiDAS. *Population: the fraud-detection logic is applied uniformly to all claimants regardless of language proficiency, access to a computer, or history of the agency's own clerical error; no declared exception exists for a claimant who cannot respond to an automated notice inside its response window. Confidence: a fraud flag carries no stated error rate or confidence threshold before it triggers a financial penalty and, in some cases, referral for criminal prosecution. Chain: the record does not preserve a documented point at which a human reviewer with actual authority to reverse the flag examined the underlying claim before the determination became final.*

None of these four builders was required to write any of this down before the system ran. That is the whole finding. The declaration above is not what any of them said. It is what the record shows they would have had to say, had anyone asked before the harm instead of after it — and in three of the four cases, a court or a federal oversight body ended up asking after.

This is not new either. Auditors call it the assertion model — management asserts, and a "not applicable, because…" is itself an owned assertion, not a blank. Safety engineers call it the safety case — you address every hazard or justify why it is out of scope, and the justification is the control. Governance calls it comply-or-explain — meet the standard or state on the record why you deviated. None of these is documentation bolted on afterward. Each is a control that works by making the builder reason in public, before the fact.

The detective control is the record. A declaration that is filed and never checked rots. Every comply-or-explain regime ever built degrades into boilerplate unless something tests whether the declared posture still holds. The hardest lesson of operational due diligence — learned after a generation of self-reported questionnaires that did not match operating reality — is that the declaration needs independent verification, not a better form. So the two halves are a closed loop. The declaration is the preventive control. The record is the detective control that keeps the declaration honest. Neither is enough alone, and together they are not documentation. They are verification.

That is what built in means. Not a policy in a binder. Not a model card stored elsewhere. A declaration made at the boundary, before the system runs, recorded so it survives to where it is relied on, and checkable against what actually happened.

II. The Architecture Already Exists

The strongest argument for this architecture is that it is already running — not as a proposal, but as the ordinary way every serious system handles a part it does not fully control. The pattern is consistent across mature domains.

System	The declaration before	The record across
Investment management	Manager due diligence — what they do, how, and whether they can sustain it	Ongoing monitoring, audit, claims and dispute review
Software delivery	Version control, test-data lineage, release sign-off	Change history, rollback, incident reconstruction
Connected applications	A vetted, finite, pre-scoped set of integrations	Access logs, scoped credentials, revocation
Aviation / mainframe / finance	Type certification, control attestation	Flight recorder, SMF records, journal and ledger
Enterprise financial systems	Segregation-of-duties matrix, checked before a role goes live	Immutable ledger logs, staged external data, configuration-drift monitoring
AI decision chains	None — no declared basis at design	None — no standardized judgment record

Six of these are worth walking, because none is about AI and each one is a control AACIF only asks to be applied where it is missing. These analogies do not claim precise equivalence between AI and any one mature system; they establish the recurring control pattern: consequential delegation requires declared limits before reliance and durable records after reliance.

The version lookup. Every mature piece of software already answers one question on demand: what am I running, and is it safe. Check any About box — Safari, Quicken, twenty other apps — and a version and build number comes back. Type that string into a search engine and something real returns: security fixes, known bugs, what changed. That is not the vendor's generosity. It works because of a specific, mandatory piece of infrastructure that has nothing to do with the vendor's goodwill: NIST's Common Platform Enumeration, a standardized `vendor:product:version` naming scheme that every published security vulnerability gets tagged against in the National Vulnerability Database. A version number is searchable and meaningful because a federal registry exists that ties it to a known, disclosed set of facts — not because software companies decided transparency was good practice. AI has the first half of this pattern already: models expose version strings, the same way Safari does. It has recently gained a plausible second half, though not yet a working one. CycloneDX's ML-BOM — an OWASP-maintained, machine-readable format for exactly this — reached production maturity in 2025 and 2026, with a companion format inside SPDX 3.0. Either can hold a model's architecture, training data provenance, evaluation metrics, and bias assessment in a structured, generatable file, the same way a conventional software bill of materials holds a package's dependencies. What it does not have, that CPE has, is a National Vulnerability Database on the other end — a central, mandatory place these get filed, correlated, and checked against. Right now an ML-BOM is a private document a vendor may or may not hand a specific procurement team on request. There is no registry, public or otherwise, that a stranger can query the way anyone can query the NVD. The format exists. The federal, or even industry-wide, place to file it does not. That is a narrower gap than it was two years ago, and it is still the whole gap.

None of this is a claim that AI builders lack discipline internally. They do not. Every serious lab tracks, in detail, which code commit, which dataset snapshot, and which training run produced a given model — the same rigor a Makefile brings to a software build, and increasingly through the same kind of tooling, purpose-built experiment trackers that record a model's lineage the way a build system records a binary's. That internal record is real, and it has to be, or nobody could debug a regression or safely ship a change. The gap AACIF names is not that this discipline is missing. It is that the discipline stops at the lab's front door. A user asking Copilot or Gemini what they are actually running gets a marketing name — "Gemini 3.1 Pro" — not the build-numbered, checkable string a company already keeps for itself. The internal version exists. The external assertion never leaves the building.

The connector. Connect an AI assistant to your email or your documents and something has already happened that you did not negotiate. The set of things it can reach is finite — vetted in advance, admitted against criteria, scoped before you arrived. You can see one identity at a time. You can reach less than you can reach yourself. None of that was decided in your session. It was a declaration the platform made at design time, and you inherited it.

The finiteness is the tell: a governed boundary admits a curated set; an ungoverned one admits anything. And note what the control did not do. It did not inhibit adoption. Enterprises connect their systems precisely because the boundary is drawn — an unscoped agent that could touch anything is the one no serious organization would let near its mail. The control is not the tax on adoption. It is the permission slip.

The software lifecycle. Software did not begin with structure. It began with the junior's instinct: let me code, don't make me document, version control is overhead, who cares what data the test ran against. Every control now taken for granted was once resisted as bureaucracy. And software did not slow down when it adopted them — it accelerated, because the structure is what made it safe to move fast. Two of those controls are AACIF's assertions in another domain. Test-data lineage — knowing what population the regression suite ran against — is the population assertion. And the version boundary is the temporal one: a program built for an old system will not run on a new one. It refuses. That refusal is a safe failure; it forces an update or a conscious choice, and it never lets you unknowingly run something stale.

Hold that last point, because it inverts for AI. Old software will not run; it stops at the door, loud and safe. An old model runs flawlessly. It answers in the present tense, fluent and confident, and its answer is indistinguishable from a current one. Software made staleness fatal and visible. AI made it silent. That inversion is why the temporal declaration is not optional: the model's vintage lives nowhere in its output, so nothing stops a stale answer unless the vintage is declared and carried.

Manager due diligence. The oldest version sits in our own field. An investment house does not hand a portfolio to a niche manager on reputation. It runs a due diligence questionnaire, walks the controls, and checks whether the firm can sustain the work over the life of the mandate — what they do, how they do it, and whether they can afford to do it in the long haul. That last dimension, continuity, is the one AI governance forgets: a model vendor that cannot fund continued training, cannot keep its retrieval current, or might sunset a line is a risk the deployer inherits. The due diligence questionnaire is Layer 0 for AI — the same instrument every allocator already runs, applied to the one supplier who has been delegated consequential judgment with less diligence than a fund-of-funds runs on a single sleeve.

So the question is not whether this can be built. It is why the one system now making medical, financial, and life-and-death judgments is the single delegation no one thought to run it on.

The enterprise ledger. Corporate financial systems ran a version of this exact fight decades ago, in the discipline this framework's own working group spent careers inside. Early ERP platforms shipped with a privileged function called Correct History that let someone overwrite a financial record and leave no trace of what it replaced — chain integrity failing by design, not by accident. Even now, cloud platforms still ship with default "seeded roles" carrying hidden segregation-of-duties conflicts out of the box, because a default configuration is nobody's asserted decision — it is what happens when nobody was required to assert one. The fix that held was not better documentation. It was decoupling

the privileged overwrite from the ledger entirely, and requiring any external data to land first in a staging table — checked for sequence, timestamp, and schema — before it is allowed to touch a production record, alongside a message registry that rejects any incoming file whose sequence number has already been seen, so a stale or duplicate record can never silently overwrite a current one. That staging-and-sequence pattern is A7, already running in production, built by people in the same rooms this paper's working group came from.

The irony is that this exact industry has already put AI inside that same infrastructure, without asking it to meet the standard the infrastructure was built to enforce. Enterprise platforms now run AI agents that review access requests against segregation-of-duties matrices, draft compliance narratives for auditors, and flag anomalies in journal entries — real authority, inside the one industry with the least tolerance for undeclared judgment anywhere in the corporate world. None of those AI layers are required to produce a temporal declaration, a population declaration, or a confidence interval about themselves, the way every human-built control around them already must. The field that invented the segregated, chain-verified, independently-evidenced ledger has installed an ungoverned judgment engine directly inside it. That is not a hypothetical failure mode. It is the fox and the chicken coop, running today, inside the henhouse that built the strongest lock in corporate history.

III. What AI Strips at the Door

AACIF begins with one question: could an independent expert later stand where the system stood and see what it saw — what it knew, what it did not, what it relied on, what uncertainty was visible, where the chain broke? If the answer is no, governance is operating without evidence, audit without a trail, insurance underwriting an unknown, and recovery beginning in the dark.

The first place the record is destroyed is the first boundary: ingestion. The provenance of every source is known at the moment it is scraped. You know whether a passage came from a forum or a clinical review, a filing or a preprint that never cleared review. That is metadata that exists, and it is thrown away. Worse, the source's own declared uncertainty is thrown away with it. A careful paper couches everything — small sample, these conditions, suggests rather than shows. The pipeline keeps the conclusion and deletes the couching. And that is not a loss of nuance. "Associated with, in this population, warranting further study" and "causes" are different claims. Stripping the frame does not compress the truth. It manufactures a certainty the source never asserted. The model turns a careful conditional into a confident categorical, and that conversion is a truth transformation, not a summary.

This matters because a model's measure of what is true is frequency — the dominant pattern of the corpus. Frequency is not truth, and the proof is in history's worst chapters, where a society's dominant pattern endorsed monstrous falsehood with total fluency. A system that learns truth from frequency cannot tell a fact corroborated ten thousand times from a falsehood repeated ten thousand times. Only provenance can — independent

measurement against single-source repetition. So the answer to a corrupted corpus is not to vet it into purity, which is impossible. It is to declare what it is — its sources, their veracity, their preserved uncertainty — and to refuse to treat everything as equally true. The flattening is the sin: a system that gives a forum post and a clinical trial the same weight does not merely lose information. It asserts a falsehood — that they are equally reliable — and that false equality becomes its confidence.

IV. Govern by Contract, Not Command

There is a reason the boundary, not the layer, is where the control sits.

Inside an organization, you govern by authority. A manager can require version control of a team because the team reports to the manager. Authority is vertical and personal, and it reaches every layer because one accountable person can require it of all of them. But authority runs out at every boundary it does not control. A hospital cannot order a model vendor to disclose what it was trained on. The deployer who bears the consequence sits downstream of a builder it does not command. Across that line, command does not work. The only thing that can carry a requirement across a boundary no one controls is a contract at the boundary itself.

That is what an assertion is. It is a boundary contract: declare what crosses, in a form the next layer can read, regardless of who built the layer. Whatever happens inside a layer remains the builder's design — the architecture, the methods, the weights. But when an output crosses a boundary, the declaration crosses with it, or its absence becomes a detectable breach.

Innovate freely inside the layer. Declare consistently at the boundary.

This is the difference between a standard and a product. A product works with itself. A standard works regardless of who built the piece — and that designer-independence is what makes the architecture both composable and enforceable. Composable, because pieces that conform to one contract assemble without each pair negotiating a private handoff; the boundary is the shared edge that lets work built by strangers fit together. Enforceable, because an underwriter or a buyer can require conformance to the contract without caring whose tool satisfied it. You cannot underwrite "use this vendor." You can underwrite "furnish a conforming record."

So AACIF is not a competitor to NIST, ISO, or the EU AI Act, and it is not beneath them. It is the evidentiary control limb of governance — the part that produces the declared, verifiable record the rest of governance has been assuming it already had. It does not replace the governor. It gives the governor something real to govern.

V. The Seven Assertions

Here is the problem in plain terms. When an AI makes a call that affects someone's life — a denial, a rejection, a fraud flag — someone should be able to go back later and answer

seven questions about it. Right now, for almost every AI system in use, nobody can answer any of them. The seven assertions below are those seven questions, written down as requirements instead of left as things nobody asks.

A1. Temporal — How old is what the AI knows?

Every AI system learned what it knows by a certain date, and stopped. But when it answers a question, it doesn't say "as of [date]." It just answers, in the same confident tone, whether its information is one week old or three years old. Old software is different — when it's out of date, it usually just stops working, or throws an error. Old AI doesn't stop. It keeps answering, smoothly, using facts that may no longer be true.

The failure this catches: an answer that sounds current but is actually stale, with nothing to warn you.

Example: DAX Copilot never tells the doctor which version of itself wrote today's note, or how recent its medical knowledge is. The note reads the same whether the model behind it was updated yesterday or a year ago.

A2. Knowledge State — Which version of the rules was it using?

Laws change. Medical guidelines change. Company policies change. An AI system is built around a specific version of whatever rulebook it follows, and that version can go out of date without the AI knowing or saying so.

The failure this catches: the AI applying an old rule as if it were still the current one, with no way for anyone to check which version it actually used.

Example: nH Predict's clinical guidelines are updated once a year. But nothing in a denial letter says which year's version was used to make that specific decision.

This one is also worth naming as solvable, not just missing, because a working version of it already exists elsewhere. Multinational accounting platforms process a single transaction through a primary ledger, evaluated under one country's accounting standard, and a secondary ledger, evaluated simultaneously under another country's — each carrying its own declared rule version, its own chart of accounts, with neither one silently overwriting the other. If a transaction can be run under two different, explicitly versioned rulebooks at once and still produce a clean audit trail, an AI system can be required to declare which single rulebook it used for one.

A3. Population — Was this AI ever tested on someone like the person it's being used on?

An AI model is trained and tested on a specific group of people or cases. It can be very accurate for that group on average, and still be wrong most of the time for someone who doesn't look like the typical case — someone older, sicker, or more unusual than what the model mostly saw during training. The real question is never "how accurate is this model." It's "accurate for who."

The failure this catches: a model that's right on average and wrong for exactly the person in front of it, because nobody checked whether that person was ever part of the group it was built for.

Example: Gene Lokken was 91 years old. If nH Predict was mostly built on younger, more typical recovery cases, nobody had to say so before his coverage was cut — and nobody did.

A4. Measurement — Is this actually confirmed, or has it just been repeated a lot?

An AI can sound very sure of something because it saw the same claim many times in its training data — not because that claim was ever independently checked. Seeing the same fact ten times doesn't make it ten times more true. But it can make an AI system sound ten times more confident.

The failure this catches: confidence that comes from repetition, mistaken for confidence that comes from evidence.

Example: Workday gives job applicants a "fit score" that sounds precise. Nothing requires the company to say whether that score is based on real, checked outcomes, or just on patterns that repeat in old hiring records.

A5. Language and Perspective — Whose point of view is this, really?

If almost everything an AI learned from came from one language, one country, or one type of source, it can end up sounding like there's wide agreement on something — when really, it just heard the same narrow point of view over and over.

The failure this catches: one perspective, repeated many times, that looks like broad agreement.

Example: DAX Copilot is trained mostly on English-language conversations. Nothing requires it to disclose that, so nobody can tell whether a note written for a patient who doesn't speak English as a first language is as reliable as one written for a patient who does.

A6. Translation — What changed when the answer got turned into a decision?

An AI's raw output is often a number or a probability — "73% confident," for example. Somewhere along the way, that soft, uncertain number gets turned into something hard and final: approved or denied, hire or reject, fraud or not fraud. That conversion is a real decision, made by someone, and right now nobody has to write down that it happened or explain how.

The failure this catches: a careful, uncertain answer being treated downstream as if it were a certain, final one.

Example: MiDAS turned a simple mismatch in reported earnings — which could easily have been a clerical error — directly into a fraud accusation, with no declared step in between that asked how confident the system actually was.

A7. Chain Integrity — Did all of the above survive long enough to matter?

This last one is different from the first six. It isn't a new thing to declare — it's a check on whether the other six survived the trip. An AI system can get every single one of A1 through A6 right at the moment it produces an answer, and still lose all of that information by the time a human being actually relies on the result. Information gets stripped out at every handoff — from the model, to the software system, to the case file, to the letter that finally reaches the person affected. A7 asks: by the time this reached someone, was any of it still there?

The failure this catches: a system where every individual piece passed inspection, but the full picture never survived to reach the person it was about.

Example: Derek Mobley could not get Workday to explain why he was rejected for over a hundred jobs. It took a federal court case just to start pulling that information out.

Assertion	What it declares	Why it matters	The failure it catches
A1 Temporal	How old the AI's knowledge is, and how fresh it was at the moment it gave this specific answer	AI answers in the same confident tone whether its knowledge is one day old or three years old	A stale answer that sounds current, because nothing stops it from sounding current
A2 Knowledge State	Which version of the model, and which version of any rule or guideline, it was using	Rules and guidelines change; the AI can keep applying an old one without saying so	Acting on a rule that's already been replaced, with no way to check which version was used
A3 Population	Who the AI was built and tested on, and whether this person or case matches that group	A model can be accurate on average and wrong for anyone who doesn't match the typical case	Applying a model to someone it was never actually tested on, without checking first
A4 Measurement	Whether the AI's confidence comes from real, independent evidence — or just from repetition	Seeing the same claim many times doesn't make it true, but it can make an AI sound sure of it	Confidence built on volume of repeated data, not on anything actually verified
A5 Language / Perspective	What language, place, or cultural viewpoint the AI's information mostly came from	One narrow point of view, repeated often, can look like broad agreement	Mistaking one repeated perspective for a well-supported consensus
A6 Translation	What changed when	A "73% confident"	A careful, uncertain

Assertion	What it declares	Why it matters	The failure it catches
	a soft, uncertain result got turned into a hard, final decision	answer is not the same thing as an "approved" or "denied" — someone made that conversion	answer, treated downstream as if it were certain and final
A7 Chain Integrity	Whether everything declared in A1–A6 actually survived every handoff to the point where someone relied on it	Every individual piece can pass, and the whole thing can still fail between steps	A system that looks fine at every step, but the record is gone by the time someone needs it

A7 is not a seventh item alongside the others — it's a check on the first six. It asks whether A1 through A6 made it, intact, all the way to the person the decision was about. A system can pass every individual check and still fail here, because this kind of failure happens in the space between steps, not inside any single one.

VI. Declaration, Tolerance, Fitness

Each assertion has three parts, and they are not the same artifact.

Component	Question	Where it lives
Declaration	Was the required basis captured and carried?	The deployment record
Tolerance	What precision or currency did this decision require?	The domain standard
Fitness	Did the declaration meet the tolerance for this use?	The verification result

Declaration is the binary, present-or-absent half — and it is already proven. The connector's scope check and the software version check are declarations enforced at a boundary; either the credential is in scope or it is not, either the version fits or it does not. What those do not have, and AI needs, is Tolerance and Fitness: the graded, decision-specific judgment software never required because its tolerance was fixed.

There is a version of this that could fail on its own terms, so it is worth naming directly. An honest declaration is also evidence. A builder who states, in writing, that a clinical model was never validated past a certain age has just handed a plaintiff's attorney the exhibit they need if that model is deployed on someone older. Left to that incentive alone, corporate legal review will do exactly what it is paid to do: sand the declaration down into language defensive enough to satisfy the form without saying anything a jury could use. "Trained on

a broad, diversified population representative of typical clinical distributions" passes a checklist and asserts nothing. This is not a hypothetical failure mode; it is the predictable output of leaving Layer 0 as free text. The fix is structural, not aspirational: a declaration is not a paragraph, it is a schema. Population is a numeric range or a named cohort identifier, not an adjective. Version is a date string and a build number, not "current." Confidence is a stated interval, not "high." A field that cannot be filled with a number, a date, or a named category does not satisfy the assertion — vague language is not a lesser declaration, it is a missing one, and the framework should treat it exactly that way.

And fitness is purpose-relative. The same South-American-derived data is the right source for a researcher studying Scandinavia's Latino community and the wrong one for a clinician treating a general Scandinavian patient. The data has no fitness. The match has fitness, and the match is a property of the decision. So the control can never auto-reject on difference — that would refuse the one user for whom the data is perfect. It surfaces the difference and hands the judgment to the person who knows the purpose. It asks the question no one is currently made to answer: is this model fit at the resolution, and the percentile, of the person in front of it?

This is sharpest where the population matters most. A model trained to minimize average error spends its accuracy where the data is dense — the typical case — and pays for it at the edges. But for a decision such as extended-care coverage, the people in front of the model are the edges: the slow recoveries, the complications, the atypical. The model is trained on the center and applied to the tail. It is right on average and wrong about everyone who matters, and its average accuracy is the alibi. Because the population was never declared, no one — not even the builder — can demonstrate the model is fit for that tail. In a controls regime, undemonstrable fitness is unfitness. The burden was never on the patient to prove the model fails the sick. It was on the builder to prove it does not, before deploying it on the sick.

VII. The Twelve-Layer Chain

AI is not one system to be assessed whole. It is a layered chain, each layer with a different owner, a different failure mode, and a different recording obligation. The declaration must follow the output through all of it.

#	Layer	What it is	What must be declared and carried
0	Deployment Envelope	The pre-authorized domain, authority level, hard stops, escalation	The intentionality record — why specific capabilities, connections, and constraints were chosen and everything else was

#	Layer	What it is	What must be declared and carried
			excluded. Set before training. Governs every downstream deployment.
1	Corpus	Raw and curated source material	Source composition, provenance, known contamination — the first place assertions are stripped
2	Training	The process that turns corpus into behavior	Training period, reinforcement choices, known failure modes
3	Model	The model artifact at a point in time	Identifier, version, release date, supersession status
4	Instruction	System prompts, guardrails, permissions	Instruction version, tool permissions, prohibited actions
5	Retrieval	Live knowledge at inference	Documents retrieved, source versions, currency, omissions
6	Inference	Computation and context at the moment of use	Prompt, context, parameters, tools, confidence, disclosed opacity
7	Output	The recommendation, score, or instruction	All assertion declarations attached to or linked from the output
8	Integration	The handoff into the enterprise system	Whether the declaration survived transformation into downstream fields
9	Human Review	Where a person accepts, rejects, or ignores	What the reviewer could see, time pressure, override, rationale
10	Action	The real-world	Final actor, AI

#	Layer	What it is	What must be declared and carried
		decision or execution	contribution, authority path, evidence at the point of action
11	Audit / Recovery	Later reconstruction, claims, litigation, recovery	Whether the chain can be replayed from action back to source

A system can pass every component-level review and still fail because the handoffs stripped the information needed to understand the judgment. The temporal label exists at output and disappears in the record. The population warning lives in the model card and never reaches the physician. The confidence interval becomes a binary flag. By the time action occurs, the judgment has lost its provenance. AACIF treats that loss as a breach — not a documentation inconvenience, but a failure of the layer that was supposed to carry the evidence.

Layer 5 deserves a harder look than it has gotten so far, because most consequential systems today are not just a static model answering from what it learned in training. They pull in live documents, records, or data at the moment of the question — retrieval-augmented generation, or a long context window fed fresh material — and the model reads whatever it is handed with the same fluent confidence it reads its own training data. A model can be declared honestly at Layers 0 through 3 and still act on a stale or unverified document injected at Layer 5 an instant before it answers. The corpus was clean; the thing it was just handed was not. That means Layer 5's declaration cannot be a fixed, one-time statement the way Layer 3's model version is. It has to be dynamic — computed fresh at each retrieval, asserting the age, source, and verification status of whatever was just placed in front of the model, not just what the model was originally built on. A framework that only checks the model and never checks what the model was fed a moment before it spoke has checked the wrong thing.

Worth stating plainly, because it changes what this framework actually needs to invent: this twelve-layer chain does not start from nothing. CycloneDX's ML-BOM — an OWASP-maintained, machine-readable format now in production use, not a proposal — already carries real, structured fields for a meaningful slice of it. Its model card holds a model's version and architecture family, cleanly covering Layer 3. It references the datasets a model was trained and evaluated against, a genuine piece of Layer 1's provenance requirement. It includes a quantitative-analysis section with an actual confidence-interval field, not a described metric standing in for one. Where it stops is instructive, not accidental. ML-BOM is a manifest for an artifact: it has no concept of Layer 5's live retrieval, Layer 6's inference-time context, Layer 8's transformation in handoff, Layer 9's human review, or Layer 10's action, because a bill of materials documents what a thing is built from, not what happens to it after someone relies on it. That is not a shortfall in ML-BOM. It is the edge of what any manifest format can be asked to do — and it marks, almost exactly, where this framework's

own contribution begins. The honest description of AACIF's technical job is not "build a record for AI." It is "build the record for the half of the chain that starts after the manifest ends."

VIII. Seeing the Chain Break

The framework is not only written. It runs.

The assessment engine takes a real deployment, classifies its tier by what the system actually does rather than what it is called, and renders the chain as a map — twelve layers by seven assertions, each cell passing, warning, failing, or not applicable. The result is a picture of where the evidence is missing, and where it died crossing a handoff.

It is tempting to call this an audit tool. It is not, and the distinction is the whole point. An audit happens to a system that already shipped. This shows something earlier and more useful: every failing cell is the visible consequence of a control that was never built in. The map is not grading the system. It is showing the chain breaking — and a broken chain is the most persuasive argument there is for building the controls in before it runs. Show a clinician, an underwriter, or a regulator the evidence dying at a specific handoff, and they do not need the theory. They see why the declaration has to exist before the system acts, not after the harm.

That is what the visual does that prose cannot. The entire problem with AI failure is that it is invisible — the stale answer is fluent, the population mismatch is confident, the denial is opaque. The engine makes the invisible legible. It renders a silent failure as a red cell at a named layer and a named assertion. The taxonomy and chain visualizers do the same for the structure: they show the schema the declarations live in and the handoffs they have to survive. Together they are the claim, displayed instead of argued — proof that the invisible can be made visible, which is the only thing that makes it governable.

IX. Tier and Authority

The intensity of the requirement depends on the consequence of error, the genuineness of human oversight, and the authority granted to the system. A drafting assistant does not need the record an AI system needs when it recommends chemotherapy, executes trades, or routes public benefits.

The tier is not what the vendor calls the system. It is revealed by what the system actually does. A recommendation engine becomes a decision system the moment its output is routed into a workflow that produces action without meaningful human review. A human in the loop is not a control if the reviewer cannot see the assertion record, cannot understand the limits, cannot override the result, or is pressured to accept it.

Genuine human review satisfies six criteria. If any is absent, the review is a ratification step, not a control.

Criterion	Control question
Visibility	Could the reviewer see the relevant assertions, limits, uncertainty, and provenance?
Competence	Was the reviewer qualified to interpret the assertion record and domain limits?
Time	Did the workflow allow real review rather than rubber-stamp approval?
Authority	Could the reviewer override, reject, escalate, or suspend the AI-supported result?
Independence	Was the reviewer free from production pressure, penalty, or incentive to accept the AI output?
Record	Was the reviewer's rationale preserved with the final action record?

Independence and segregation of duties. A boundary control is incomplete if the same party that benefits from the AI-supported action is also the only party that validates the assertion record. AACIF therefore requires segregation of duties at high-consequence authority levels: the builder may generate declarations, and the deployer may rely on them, but verification of completeness, survival, and material exceptions must be performed by a function with sufficient independence from the team seeking the deployment or benefiting from the action. Without that separation, the declaration risks becoming self-attestation rather than control. An entity cannot configure the AI evidence universe and also approve exceptions arising from that configuration. Where a single corporate structure consolidates the role of premium collector, denial algorithm operator, internal reviewer, and appeal processor, no layer of that structure constitutes independent verification — and the assertion record it produces is not independent evidence.

Authority	Description	Recording implication
Observe	Summarizes, classifies, drafts without direct reliance	Basic provenance and version record
Recommend	Proposes a course of action for human review	Assertions visible to the reviewer; limits attached
Decide	Selects an outcome or materially determines the result	Full assertion record; tolerance and fitness validated
Act	Triggers real-world execution or denial	Tamper-evident record required; hard stops documented
Chain	Output becomes input to other automated systems	A7 chain integrity required at every handoff

The six-criterion table above assumes the human reviewer is rested, attentive, and reading each flag on its merits. At volume, that assumption breaks on its own, without anyone acting in bad faith. A workflow that throws an uncertainty warning or a population-mismatch flag on a third of its transactions will exhaust the reviewer inside a week; the flags stop getting read and start getting cleared. And once that pattern sets in, it is exploitable on purpose: a system, or a person configuring one, can learn that triggering minor exceptions on every transaction forces the review queue into pure throughput mode, turning the review step this framework requires back into the ratification it was built to prevent. Human review cannot be the only backstop at the tier where it matters most. At the Act and Chain authority levels, a boundary violation above a declared threshold must trigger a programmatic hard stop — the system refuses to execute, automatically, with no override available except an explicit, logged, multi-party authorization — rather than one more flag added to a queue no one has time to read.

This is where the cost objection collapses. “Too expensive, too hard, the value is not there” is the economics of a low-consequence tier smuggled into a high-consequence decision. At the consequence this framework reserves its strictest requirement for — irreversible harm, nominal oversight — one life is the unit of measure, not a line item. You do not owe the chemotherapy system the cost-benefit math of the marketing-copy generator. The tier is the answer.

X. Why It Gets Built

A control that is right and never adopted changes nothing. So it is worth being plain about why this one gets built, and where the ordinary incentive fails.

The honest argument is not that a life is beyond price. The institutions that decide already price lives — every insurer and regulator works with a number. The argument that moves them is that the control is cheaper than the priced harm: a one-time cost to declare provenance at ingestion, amortized across every output forever, against a single preventable, foreseeable failure — plus the judgment, the headline, the lost deployment, and the next one. And the work has to happen at ingestion because it cannot be reperformed later. You cannot reconstruct the population of an old scrape; the metadata is gone. The difficulty everyone points to is the difficulty of doing it after. Done at the boundary, provenance capture is cheaper than post-hoc reconstruction. The cost is real, but it can be tiered by authority, consequence, granularity, and retention duty. That is the whole case for building it in.

But there is a harder case, and the framework should not flinch from it. The cost argument only bites when the party deciding bears the harm. In documented patterns of algorithmic claim denial, it does not. The cost of a wrong denial falls on the patient. In the Medicare Advantage skilled-nursing-facility context, federal oversight has found a pattern in which fewer than one percent of affected beneficiaries filed appeals, even though more than ninety percent of appealed denials were overturned by independent reviewers. That specific pattern does not prove every algorithmic denial system behaves the same way, but it shows why opacity matters: a denial that cannot be understood cannot be effectively

contested. The wrongful denial is profitable precisely because it is rarely seen and rarely fought. Where harm is externalized and detection is suppressed, the cost frame favors the harm.

This is exactly where the record does its work. The reason almost no one appeals is opacity — you cannot contest what you cannot see, and you cannot see a model's error rate, its population, or its basis. A denial that must carry its declaration can be contested. Raise the rate at which error is seen, and the wrong denial stops being free. The record does not appeal to conscience. It re-internalizes a cost that was engineered out of view.

Where authority is captured, enforcement runs through the parties who can still price risk. An insurer cannot write coverage against a record that does not exist; a buyer cannot defend a reliance it could not inspect. A vendor whose declaration is missing becomes uninsurable and un-integrable — the interoperate-or-die of the AI stack. None of this needs a single regulator. The first liable buyer who refuses to integrate an undeclared vendor starts the cascade, the way SOC 2 spread: no law required it, buyers demanded it, and vendors produced it once losing deals cost more than making it. A vendor's Layer 0 is the due diligence the downstream party reads to decide whether it is taking on an exposure it cannot defend — the document that turns "I did not know," the one defense that is no defense, into a choice the buyer can stand behind, or decline.

That cascade is the optimistic path, and it should be stated as one path, not the only one. The enterprise financial systems this paper already draws on did not get their controls from a buyer-led cascade. They got them because Sarbanes-Oxley made a named executive personally certify, under signature and personal liability, that the controls existed — and the certifications followed real corporate collapses, not anticipation of them. AI has no equivalent forcing statute today, and no named individual carries personal liability for a missing AI assertion record the way a CEO and CFO carry it for financial controls. The SOC 2 cascade is real and it can work here too, but it is not guaranteed to arrive before the equivalent of an Enron makes it arrive by force instead. This paper does not need to resolve which path comes first. It needs the working group to stop assuming the market path is the only one, since the honest case for adoption is stronger with both routes named than with one.

The regulatory route is not hypothetical; it is underway, in real time, and worth watching precisely because of how it is stumbling. The EU AI Act's Annex IV would require something close to a Layer 0 declaration — intended purpose, training and validation population, accuracy and bias metrics, human oversight design — for high-risk systems, with real financial penalties attached. It was set to bind healthcare, hiring, and benefits systems starting August 2026. As of mid-2026, that date has already slipped: political agreement reached in May pushes the deadline for these systems toward December 2027, and the technical standards meant to define compliance are still unfinished. That slippage is itself the lesson, not a footnote to it. A regulator with real enforcement power, writing requirements that overlap substantially with this framework, has still taken years longer than planned to specify what "documented" actually means — the same difficulty this paper has been arguing requires a structured schema rather than free text. And even once Annex IV lands, it will document a system once, at the point it reaches market. It will not

require the declaration to travel with one specific patient's denial or one specific applicant's rejection the way A7 requires. The law arriving does not make this paper redundant. It confirms the diagnosis and leaves the one thing it was never built to solve.

The United States is not converging toward one answer either, which strengthens the case for a portable technical standard rather than weakens it. Colorado passed a comprehensive AI act, then repealed it before it took effect, replacing it with a narrower one. New York City's automated-hiring-tool law has been on the books since 2023, and a December 2025 city audit found its own enforcement largely ineffective. Different states are moving at different speeds toward different requirements, and a deployer operating in more than one of them cannot build to a single rule the way an EU deployer eventually will. A framework that specifies the record itself, independent of which jurisdiction's paperwork it satisfies, is worth more in a fragmented landscape than in a unified one — it is the one thing a builder can implement once and point to everywhere, rather than assembling a different compliance file for every state that changes its mind.

The market has stopped waiting for the theory. As of 2026, major commercial carriers — Berkshire Hathaway, Chubb, Travelers, AIG among them — have sought and largely won state approval to exclude AI-related harm from standard general liability, D&O, and E&O policies; more than four in five such requests have been approved. That retreat opened a gap, and specialist underwriters have moved into it fast: Munich Re's aiSure product line, HSB's new AI liability coverage, a Lloyd's-backed startup underwriting mid-market AI risk directly. The people actually pricing this risk today are asking, at renewal, for exactly what this paper specifies — audit trails, human-oversight protocols, evidence that a person can actually override a drifting model, not just a checkbox confirming one exists. One underwriter has already reached for the same comparison this paper makes to SOC 2: they call the exposure "silent AI," the same way an earlier decade discovered "silent cyber" sitting undeclared inside policies nobody had priced for it. The cascade is not a forecast anymore. It is happening, ad hoc, deal by deal, with every underwriter building their own private version of a Layer 0 questionnaire because no shared one exists yet. That absence of a common schema, in a market already paying to solve this problem, is the clearest evidence in this entire paper that the gap is not technical and not even motivational. Everyone with money on the table already wants the record. Nobody has agreed on its shape.

There is also a nearer, narrower lever than waiting for the general commercial market to standardize, and it is worth naming because it is available now. Large self-insured enterprises do not need to wait on that standardization at all. A large health system, bank, or technology enterprise that self-insures through its own captive has an immediate, internal reason to require this framework of its own AI deployments — not to satisfy an external buyer, but to prove to its own reinsurance markets that its exposure is bounded and its controls are real. A captive's risk officer does not need to wait for an industry standard to demand a chain-integrity record; they can require one tomorrow, of their own organization, because they are the buyer and the insurer at once. That is a faster and more plausible near-term adoption path than the general commercial cascade, and it deserves its own line in the case for why this gets built rather than sitting inside the SOC 2 analogy as an afterthought.

XI. What This Does Not Yet Solve

Inference opacity remains partially unresolved, but it is a diagnosed condition with a visible trajectory, not a permanent wall. Mechanistic interpretability has moved past saliency maps and post-hoc rationalization. Anthropic's application of sparse autoencoders to Claude 3 Sonnet was the first time that technique reached a deployed frontier model. OpenAI extracted roughly sixteen million interpretable features from GPT-4 the same way. Circuit tracing has begun to reveal partial mechanisms inside these systems — including planning behavior, hallucination, and chains of reasoning that misrepresent what actually drove the output.

None of this closes the gap. The features found so far are a small slice of what a model has learned, not a full account of it. Routing a model's computation through its own autoencoder to inspect it degrades performance sharply — by roughly an order of magnitude in compute terms for GPT-4. Full coverage might require billions of features, and some published results have been negative: Google DeepMind reported sparse autoencoders underperforming on certain downstream evaluation tasks, prompting at least one major lab to deprioritize that specific technique. Faithfulness — whether an interpretability technique's explanation actually reflects what the model is doing, rather than a plausible story about it — is still unresolved.

The honest grading is this: interpretability has improved substantially, has not kept pace with the capability of the systems it is trying to explain, and cannot today certify that a model's internal reasoning matches its output. It is scientifically promising and operationally immature.

AACIF's response to that state is not to wait for it to resolve, and not to pretend it already has. Layer 6 declares what can currently be shown — partial feature identification, circuit-level tracing where it exists, causal interventions that have been run — and discloses, as its own assertion, what cannot yet be shown: full model explanation, safety certification, or an operational guarantee that a declared basis matches the internal computation that produced it. As the field's coverage and faithfulness improve, Layer 6 is designed to absorb that improvement without a rewrite — the disclosure narrows as the science does. The uncertainty is not hidden. It is declared, the same way every other gap in the framework is declared, and it is currently as wide as the research honestly requires it to be.

Tolerance values require domain authority. AACIF can define the coordinate system. It cannot alone decide the acceptable temporal gap for oncology guidance, the required population match for credit decisions, or the translation tolerance for autonomous execution. Domain communities must populate those, and that is a multi-year effort.

Taxonomy governance is unresolved. A multi-domain judgment taxonomy has no single natural owner. The base taxonomy, the extension rules, the versioning discipline, and the arbitration process have to be designed deliberately.

Implementation cost is real. So were flight recorders, SMF, audit trails, and clinical records. The relevant comparison is not cost against no cost. It is cost against unreconstructable harm.

XII. The Invitation

This paper does not ask for agreement that every assertion is final. It asks whether the central requirement is right: that a consequential AI system declare, at each boundary, what it knew and on what basis — and that the declaration survive to the point where someone acts on it.

If the answer is yes, the next work is specification. Which declarations are required at which authority level. Which tolerances are domain-specific and which are computable. Which boundary failures should suspend operation. Who retains the record, for how long, and under what independence.

The control was never going to be a cleaner corpus or a smarter model. It is a system made to declare what it knows and what it does not, before it acts, in a form that survives to the moment of reliance. Every mature system that delegates consequential work already does this. AI is the exception. The work is to make it the rule.

Developed with AI assistance. The framework, arguments, and claims are the author's own.

AACIF — Controls Built In | thom barrett | Living Life Press | 2026