

Turning Policy Documents into Evidence: Building Measurement Infrastructure for Operational AI Governance

Jaya Kandaswamy
President

The Institute for AI Measurement Science
Southlake, TX, USA
jaya.kandaswamy@iaimscience.org

Sowmya Podila
Technical Advisor

The Institute for AI Measurement Science
Southlake, TX, USA
orcid.org/0009-0007-4346-7750

Ramesh Ramaraj
Data Standards and Governance
The Institute for AI Measurement Science
Southlake, TX, USA
ramesh.ramaraj@iaimscience.org

Abstract—Artificial Intelligence (AI) governance frameworks such as the European Union (EU) Artificial Intelligence Act [1], the NIST AI Risk Management Framework (AI RMF) [2], and ISO/IEC 42001 [3] establish formal obligations for documentation, risk management, transparency, and monitoring. However, these frameworks are predominantly prose-based and non-executable, requiring organizations to manually interpret and operationalize requirements. Recent research on Policy-to-Tests (P2T) transformation [4] demonstrates that natural language policies can be converted into structured, atomic rules suitable for automated validation. In parallel, NIST’s Open Security Controls Assessment Language (OSCAL) [5] demonstrates that compliance artifacts can be represented in machine-readable formats. This paper presents a reusable multi-agent governance measurement architecture—comprising a Standards Monitoring Agent, a Control Generation Agent, and an Evidence Collection Agent orchestrated by a modern stateful agentic framework. A multi-step agentic pipeline powered by Large Language Models (LLMs) converts governance prose into versioned, cross-framework evidence artifacts with mandatory human-in-the-loop (HITL) review at each stage. A proof-of-concept prototype across representative NIST AI RMF and CMMC [6] controls showed feasibility and an observed reduction of approximately 60% in manual compliance preparation effort compared to spreadsheet-based workflows. By separating evidence generation from compliance judgment, the architecture enables scalable, reproducible, and audit-ready AI governance evidence aligned with emerging regulatory requirements.

Index Terms—AI governance; compliance-as-code; OSCAL; policy transformation; AI risk management; agentic architecture; executable regulation

I. INTRODUCTION

AI governance frameworks are not yet operational. Organizations deploying autonomous and semi-autonomous AI systems face a concrete and unsolved problem: existing governance frameworks such as the EU AI Act [1], NIST AI RMF [2], and ISO/IEC 42001 [3] specify what must be demonstrated but provide no standardized, interoperable mechanism for generating the evidence that demonstrates it. The result is a measurement gap—organizations cannot systematically prove that their AI systems behave as intended, remain within authorized scope, and satisfy regulatory require-

ments, not because the policies are absent, but because the infrastructure to operationalize them does not exist.

This gap has two dimensions. The first is the *translation gap* between regulatory intent and organizational implementation: governance frameworks are written in prose, requiring manual interpretation that produces inconsistent, organization-specific artifacts unsuitable for cross-framework audit. The second is the *verification gap* between implementation and regulatory confirmation: even where organizations have implemented controls, they lack standardized, auditable evidence artifacts that regulators and auditors can independently verify. Without structured, traceable artifacts, inconsistencies compound across both translations.

Building on P2T transformation [4] and NIST’s OSCAL [5], which demonstrate respectively that policies can be converted to machine-readable rules and that compliance artifacts can be standardized, what remains missing is interoperable, audit-grade AI governance evidence artifacts reusable across audits, organizations, and time. This paper presents a reference governance measurement architecture for executable AI governance, independent of any specific institutional implementation.

Managerial relevance. Engineering teams, compliance officers, and technology leaders must translate broad policy requirements into implementation practices and evidence artifacts that can withstand internal audits and external scrutiny. This paper introduces a measurement-oriented architecture that converts governance policies into reusable control objects and standardized evidence artifacts, reducing manual compliance work, improving traceability, and supporting practitioners implementing trustworthy AI systems.

II. BACKGROUND

The EU AI Act becomes generally applicable in 2026 and requires demonstrable compliance for high-risk systems [1]. Existing frameworks such as the NIST AI RMF [2] and ISO/IEC 42001 [3] define governance expectations but do not prescribe interoperable implementation artifacts. Organizations must therefore translate overlapping obligations into auditable engineering evidence. Two gaps exist: first, between regulatory

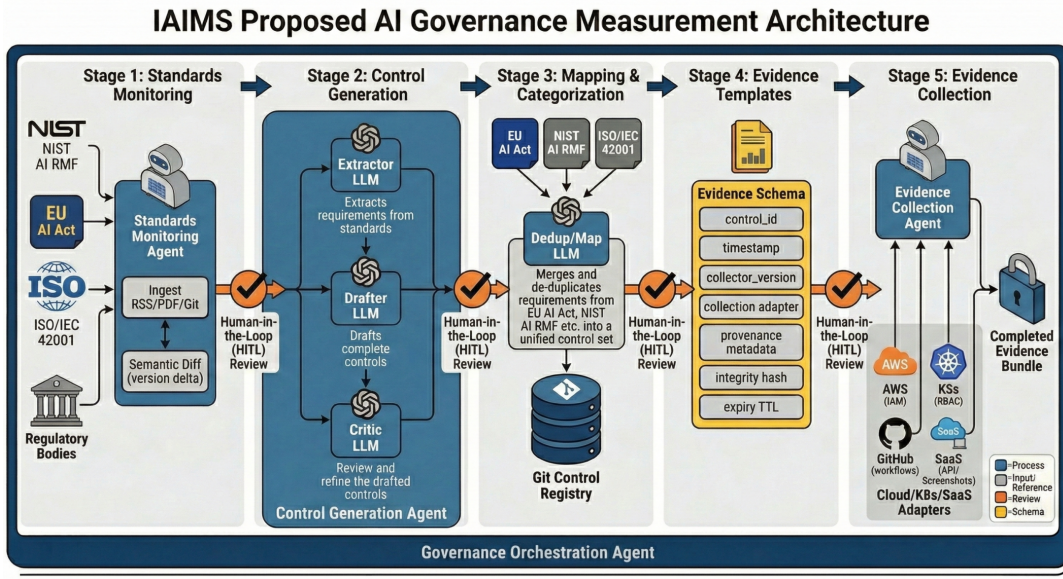


Fig. 1. Proposed AI governance measurement architecture showing the five-step governance orchestration pipeline including standards monitoring, control generation, control mapping, evidence schema generation, and automated evidence collection.

intent and organizational implementation; second, between implementation and regulatory confirmation. Specifically, no shared schemas, standardized interfaces, or reusable data models exist for representing AI governance evidence in a form portable across frameworks, organizations, and audit contexts.

Academic research has begun examining technical approaches for operationalizing AI governance. Prior work on algorithmic auditing highlights the need for verifiable evidence linking governance policies to system behavior [7]. Other research has explored structured evaluation benchmarks and compliance testing environments for assessing whether AI systems meet safety and policy requirements [8]. Practitioner guidance on EU AI Act readiness identifies data governance, transparency documentation, and evidence preparation as operationally challenging areas, with organizations often relying on unstructured artifacts—spreadsheets, email threads, and ad hoc reports—that cannot be systematically verified across audits [9]. Standards efforts such as IEEE P2863 [10] confirm that the research community recognizes AI governance demands operational infrastructure, not only policy principles.

The broader compliance-as-code ecosystem provides relevant foundations. Open Policy Agent (OPA) [11] and AWS Cedar [12] enable declarative policy enforcement, but address runtime decision-making rather than the generation of structured evidence artifacts. Enterprise platforms such as Credo AI [13] and Monitaur [14] provide AI system inventories, policy management, and regulatory mapping, but produce artifacts tightly coupled to their internal data models, making cross-framework evidence reuse impractical without manual reformatting. Unlike prior work on policy enforcement or behavioral evaluation, this paper addresses the missing middle layer: the generation of structured, reusable evidence artifacts that link policy obligations to observable system state across

frameworks and over time.

III. PROPOSED IMPLEMENTATION

The proposed architecture consists of five steps: policy change detection, requirement extraction, structured control definition, reusable evidence schema design, and deterministic evidence collection. All artifacts require human review before release. The architecture is realized as a multi-agent system—Standards Monitoring Agent, Control Generation Agent, and Evidence Collection Agent—orchestrated by a Governance Orchestration Agent built on any stateful agentic framework such as LangGraph [15]. The overall architecture is illustrated in Fig. 1. AI agents assist in identifying regulatory updates and extracting candidate requirements; they suggest mappings and draft artifacts, but no controls are published and no code is modified without explicit HITL approval.

A. Step 1—Watch for Changes in Standards

The Standards Monitoring Agent monitors NIST, ANSI, ISO, IEEE, the EU, and government agencies via RSS, PDF, and Git connectors. Documents are normalized into canonical segments and compared against prior versions using structural diff techniques. An LLM assessment distinguishes material regulatory changes from cosmetic reformatting. Confirmed material changes trigger a targeted re-run of the Control Generation Agent scoped only to affected controls.

B. Step 2—Generate Controls

The Control Generation Agent converts raw governance prose into versioned control objects through a multi-pass LLM pipeline. The Extractor LLM identifies candidate requirements, filtering for *must* and *should* language. The Drafter LLM produces a complete control object specifying `control_id`, `version`, `framework_sources`,

domain, lifecycle_stage, requirement, evidence_primitives, and a recipe.yaml reference that drives downstream collection. Each control object serializes as typed YAML, with recipe.yaml specifying adapter, query target, evidence primitive, validation rule, and expiration trigger. The Critic LLM challenges the draft for accuracy, scope, and duplication. A schema validation layer enforces structural correctness before any output reaches a human reviewer.

C. Step 3—Map and Categorize Controls

A Dedup/Map LLM collapses overlapping controls across NIST AI RMF 1.0, ISO/IEC 42001, and the EU AI Act into single control objects with multiple source citations, eliminating redundant implementation work. The resulting catalog cross-references applicable frameworks by risk category and system component. A HITL domain reviewer approves, edits, or rejects before promotion to the Git control registry.

D. Step 4—Define Evidence Templates

Every evidence artifact follows a common envelope: control_id, schema_version, collection_timestamp, collector_version, system_component, evidence_items, provenance metadata, integrity hash, and lifecycle parameters. The recipe.yaml stub from Step 2 specifies which adapters and queries are required, linking control definition directly to evidence collection.

E. Step 5—Automate Evidence Collection

The Evidence Collection Agent executes the recipe.yaml attached to each approved control. Reference collectors are designed for open-source release; all changes go through review, and releases are versioned and signed. No automatic publishing. Collectors run inside an organization’s environment with read-only access across cloud platforms (AWS, GCP, Azure), Kubernetes, GitHub, databases, and SaaS systems. Where automation is not feasible, structured attestations and hashed configuration exports are supported. Evidence collection prioritizes deterministic infrastructure queries and rule-based validation. When ambiguous adapter outputs cannot be resolved deterministically, the system either escalates for HITL review or logs the LLM prompt, model identifier, inference parameters, and generated output alongside the evidence bundle to preserve audit traceability. LLM outputs serve as advisory metadata only, with evidence acceptance subject to deterministic validation or HITL approval. Evidence bundles include schema version, collector version, integrity hash, and expiration window, and can be serialized into OSCAL assessment-result layers [5].

IV. LIFECYCLE MANAGEMENT

Evidence artifacts are time-bounded and include creation timestamps, defined validity windows, and re-measurement triggers (e.g., model retraining or configuration changes). The Governance Orchestration Agent enforces HITL review gates

before any artifact is published. Table I illustrates the lifecycle states through which every evidence artifact progresses.

TABLE I
EVIDENCE ARTIFACT LIFECYCLE STATES

State	Trigger	Output
Pending	Control approved in registry	recipe.yaml stub created
Collecting	Collection Agent invoked	Raw adapter results gathered
Assembling	All adapters complete	Evidence bundle drafted
Under Review	Bundle submitted for HITL	Reviewer approves or rejects
Active	HITL approval granted	Signed bundle published
Expiring	Validity threshold reached	Re-measurement triggered
Expired	Validity window elapsed	Bundle archived
Invalidated	Model retrain or config change	Bundle flagged; retriggered

A. Evidence Template

Control DATA-LOG-001 (*Sensitive personal data must not appear in application logs by default*) illustrates the evidence template design, synthesizing NIST AI RMF (MANAGE 4.1), EU AI Act (Articles 10 and 12), and ISO/IEC 42001 (A.8.5). In cloud-native deployments, an infrastructure query adapter checks logging configuration, verifies encryption, and scans for PII, producing a ScanResult and ConfigurationSnapshot; on-premises deployments run the same scan locally with exported configuration files. The evidence format is identical across environments; only the collection adapter changes. Compliance determination remains the responsibility of auditors and adopting organizations.

V. PROOF-OF-CONCEPT PROTOTYPE

To evaluate feasibility, a lightweight prototype was implemented using representative governance controls from the NIST AI RMF and Cybersecurity Maturity Model Certification (CMMC) [6] at a federal defense contractor supporting regulated government environments. CMMC was selected to test whether the evidence schema generalizes to infrastructure compliance domains beyond AI-specific governance frameworks. The prototype focused on three governance areas: audit logging verification, configuration management, and access control validation. The system generated structured control objects and executed adapter-based evidence collection against AWS CloudWatch, Kubernetes configurations, and GitHub repositories. Evidence artifacts were normalized into a common schema including timestamps, collector metadata, integrity hashes, and configuration snapshots.

Automated evidence generation reduced manual compliance preparation effort by approximately 60% compared to spreadsheet-based workflows across 12 controls. Effort reduction was estimated by comparing end-to-end collection

time against the organization’s prior spreadsheet-based process for the same 12 controls. Key findings: adapter-based collection produced consistently structured artifacts without manual reformatting; HITL review added approximately one business day per control batch but was essential for resolving ambiguous framework mappings; deterministic queries were achievable for infrastructure controls, while LLM-assisted interpretation with full audit logging was used only for unstructured log analysis.

TABLE II
PROTOTYPE EVALUATION RESULTS

Governance Area	Ctrl.	Adapters	Artifacts
Audit Logging	4	CloudWatch	ScanResult, ConfigSnap
Config Mgmt	4	K8s, GitHub	ConfigSnap, ChangeLog
Access Control	4	IAM, GitHub, K8s	AccessSnap, PolicyDiff
Total	12	8	36

VI. LIMITATIONS AND THREATS TO VALIDITY

As a concept paper with a proof-of-concept prototype, this work identifies several limitations. First, LLM-generated controls may be plausible but technically incorrect; schema validation and the Critic LLM pass are primary defenses, but their effectiveness at scale remains unvalidated. Second, ISO/IEC 42001 [3] is paywalled; initial implementation targets NIST AI RMF [2] and the EU AI Act [1] first. Third, the HITL review step is the anticipated throughput bottleneck; mitigation strategies are reserved for future work. Fourth, adapter coverage is strongest for public cloud; legacy and air-gapped systems rely on structured attestation, introducing manual handling risk. Fifth, the 60% effort reduction is practitioner observation across 12 controls, not a controlled experiment. Sixth, the architecture does not reinterpret regulations or certify systems.

VII. CONCLUSION

Executable AI governance depends on more than policy interpretation—it requires structured controls and consistent evidence generation. By defining reusable control objects and deterministic evidence artifacts through a multi-agent architecture with HITL review at every stage, organizations can demonstrate what they have implemented without relying on ad hoc documentation. A proof-of-concept prototype confirms the architecture is feasible across representative cloud-native environments. This work focuses narrowly on the measurement layer and does not reinterpret regulations or certify systems.

Future work will explore deeper OSCAL alignment, model card integration, sector-specific AI regulations, and rigorous empirical validation across heterogeneous environments; systematic evidence generation may also surface regulatory

ambiguities as structured feedback to standards bodies such as NIST or the EU AI Office.

ACKNOWLEDGMENT

The authors thank Shaadana Rajesh for her assistance with research, preparation, and presentation at the conference. The authors used generative AI tools to assist with language clarity and figure preparation. ChatGPT (OpenAI) was used for limited language refinement and editing. Gemini (Google) generated the architecture diagram from the authors’ conceptual sketches. All research design, technical content, analysis, interpretations, and conclusions were developed and validated solely by the authors.

REFERENCES

- [1] European Union, “Regulation (EU) 2024/1689 Laying Down Harmonised Rules on Artificial Intelligence (Artificial Intelligence Act),” *Official Journal of the European Union*, 2024.
- [2] National Institute of Standards and Technology, *Artificial Intelligence Risk Management Framework (AI RMF 1.0)*, NIST AI 100-1, Jan. 2023.
- [3] International Organization for Standardization and International Electrotechnical Commission, *ISO/IEC 42001:2023 Information Technology—Artificial Intelligence—Management System*, 2023.
- [4] G. V. Datla *et al.*, “Policy→Tests (P2T) for operationalizing AI governance,” GitHub, 2025.
- [5] National Institute of Standards and Technology, “Open Security Controls Assessment Language (OSCAL),” 2023.
- [6] U.S. Department of Defense, “Cybersecurity Maturity Model Certification (CMMC) 2.0,” Office of the Under Secretary of Defense for Acquisition and Sustainment, 2021.
- [7] I. D. Raji, A. Smart, R. N. White *et al.*, “Closing the AI accountability gap: defining an end-to-end framework for internal algorithmic auditing,” in *Proc. 2020 Conf. Fairness, Accountability, and Transparency (FAccT)*, 2020.
- [8] P. Guldemann, A. Spiridonov, R. Staab, N. Jovanović *et al.*, “COMPL-AI Framework: A Technical Interpretation and LLM Benchmarking Suite for the EU Artificial Intelligence Act,” arXiv:2410.07959, 2024.
- [9] ISACA, “Understanding the EU AI Act,” ISACA White Paper, 2024.
- [10] IEEE Standards Association, *IEEE P2863: Recommended Practice for Organizational Governance of Artificial Intelligence*, IEEE, 2022.
- [11] Open Policy Agent, “OPA: Policy-based control for cloud native environments,” CNCF, 2024.
- [12] AWS, “Cedar: A language for defining permissions as policies,” 2023.
- [13] Credo AI, “Credo AI Responsible AI Governance Platform,” OECD AI Policy Observatory, 2023.
- [14] Monitaur, “Monitaur AI Governance Platform,” OECD AI Policy Observatory, 2023.
- [15] LangChain, Inc., “LangGraph: Build Stateful, Multi-ACTOR Applications with LLMs,” 2024.