AnimusLab

Response to FSB Consultation: Sound Practices for Responsible Adoption of Artificial Intelligence

Submitted by: **Tanishq Dasari**, Lead Researcher & Founder, AnimusLab
Contact: tan@animuslab.dev | animuslab.dev
Zenodo Preprint: doi.org/10.5281/zenodo.19734724
Date: July 2026

AnimusLab is an independent research organisation developing open-source governance infrastructure for autonomous AI systems in regulated financial markets. Our primary technical output is Anchor - an open-source governance engine (*pip install anchor-audit*) that enforces a cryptographically signed rule set across static code analysis and runtime AI decision interception, producing tamper-evident audit trails mapped to regulatory frameworks including the EU AI Act, RBI FREE-AI, CFPB Regulation B, the SEC's 2026 Examination Priorities, and the NIST AI RMF. We welcome the FSB's consultation and respond to the seven questions below, drawing on our published research and implementation experience.

Q1 - Benefits and Risks of AI Adoption

We agree with the report's characterisation of AI adoption risks, and wish to highlight one category that warrants more precise treatment: **execution authority risk in agentic AI systems.**

The report correctly identifies that agentic AI introduces novel risks through autonomous planning and execution. However, the current framing conflates two structurally distinct problems:

- **Model quality risk:** the risk that an AI system produces inaccurate, biased, or hallucinated outputs.
- **Execution authority risk:** the risk that an AI system takes an action it was never authorised to take - regardless of whether its output is accurate.

Our research across three well-documented financial failures demonstrates that the most financially consequential AI-adjacent failures were not model quality failures. Knight Capital Group's $460 million loss in 45 minutes was caused by a deprecated code path executing without authority. Zillow's $900 million loss was caused by a system continuing to act under a policy that no longer reflected market conditions. The Flash Crash investigation required five months of reconstruction precisely because no per-decision execution record existed at the time of the decisions.

These failures share a structural property: the technical capability to cause harm existed in each system, but the governance infrastructure to constrain its execution authority did not. We term this the **execution governance gap**, and we believe the FSB's sound practices would benefit from explicitly naming execution authority as a distinct governance object requiring its own controls.

Q2 - Comprehensiveness and Clarity of the Sound Practice

The 12 sound practices are well-structured and cover the major governance dimensions. We offer three targeted observations on comprehensiveness and one on implementation clarity.

Observation 1: The audit trail requirement needs a technical precision floor.

Sound Practice 8 (Explainability and Transparency) and Sound Practice 9 (Performance Management) both imply that institutions should be able to explain AI decisions. The CFPB's 2024 enforcement action against Goldman Sachs - resulting in over $89 million in penalties - established that this requirement applies at the level of the individual decision, not aggregate model statistics.

We recommend the FSB specify that for high-stakes AI decisions (credit, lending, trading, fraud), the audit trail must include: a machine-readable reason code, feature attribution for that specific decision, a timestamp, the policy version in effect, and a tamper-evident record that cannot be altered after the fact. This is already enforceable under CFPB Regulation B and EU AI Act Article 12. Making it explicit in the FSB sound practices would reduce ambiguity for institutions operating across jurisdictions.

Observation 2: Sound Practice 10 (Human Oversight) requires a structural complement for agentic AI.

The report correctly acknowledges that continuous human monitoring of individual agent decisions becomes impractical as agentic systems scale. The proposed compensating control - AI monitoring AI - is architecturally sound but currently underdefined. We suggest the FSB specify that AI-on-AI monitoring, to be governance-credible, must itself operate under a signed, version-controlled policy that is: (a) independent of the monitored system's policy, (b) cryptographically sealed against post-hoc modification, and (c) capable of producing its own tamper-evident audit record. Without these properties, AI-on-AI monitoring cannot be distinguished from the system it monitors.

Observation 3: The sound practices do not currently address governance drift.

Governance drift - where a deployed AI system's effective behaviour diverges from its stated policy over time, without any single decision constituting a violation - was the proximate cause of Zillow's loss. We recommend Sound Practice 9 be extended to include a requirement for **constitutional drift detection**: a mechanism that compares the current behaviour of a deployed AI system against its baseline policy state, flags divergence, and requires documented re-approval before the policy is considered current.

On implementation clarity.

Smaller institutions - particularly those in emerging markets - may find the gap between the sound practices and implementable technical controls difficult to bridge without reference to open-source, auditable tooling. The FSB might consider maintaining a non-prescriptive registry of technical implementations that have demonstrated alignment with the sound practices, analogous to how NIST maintains its cybersecurity framework implementation tiers.

Q3 - Balance Between AI Forms and Emerging AI (GenAI, Agentic)

The balance is directionally correct but the agentic AI section would benefit from two architectural clarifications: one on temporal governance layers, and one on the governance policy supply chain - a structural gap the sound practices do not currently address. These problems are manifestations of the same underlying issue: **governance itself is a dynamic system that requires lifecycle management rather than static documentation.**

The fourth temporal layer: ante-hoc enforcement.

The report proposes three temporal layers of governance: before deployment (development controls), during deployment (monitoring), and after incidents (investigation). For traditional AI systems, this is adequate. For agentic AI systems, it is not - because the interval between 'during deployment' and 'after incident' can be measured in milliseconds, as the Knight Capital and Flash Crash cases demonstrate.

We propose that the FSB explicitly recognise a fourth temporal layer: **ante-hoc enforcement** - governance that operates at the moment of generation, before a decision is output and before execution occurs. This layer, which we term AnchorJIT in our research roadmap, allows governance policies to be compiled into constraints active at semantically critical boundaries - tool invocations, structured output fields, permission escalations - without token-level overhead for every output. It does not replace runtime monitoring or post-hoc audit; it closes the window between generation and action where agentic AI failures originate.

The governance policy supply chain: Canon.

There is a structural problem the sound practices do not currently resolve: **how does an institution know when the governance frameworks it has implemented against have themselves been updated?**

Today, every financial institution has personnel manually monitoring external authoritative sources - NIST, FINOS, OWASP, RBI, the EU AI Office, CFPB, SEC - for rule changes, new guidance, and framework updates. This process is undocumented, inconsistent, and entirely dependent on individual awareness. When FINOS adds a rule, or OWASP updates its LLM Top 10, or the RBI issues new FREE-AI recommendations, there is no systematic mechanism ensuring that an institution's internal governance constitution is updated to reflect the change. Governance rule sets become stale without anyone being aware - a form of governance drift that operates at the policy layer rather than the model layer.

We term this the **governance policy supply chain** problem, by analogy with software supply chain security - a domain regulators and institutions already invest in significantly under frameworks like NIST SSDF. Just as a software supply chain monitors upstream dependencies for vulnerabilities and flags changes for review, a governance policy supply chain must monitor upstream authoritative sources for rule changes and route them through a controlled approval process before they enter production governance.

Canon is the architectural layer we have designed and implemented to address this problem. It is available as an open-source working prototype (v0.1.0, Apache 2.0) at github.com/AnimusLab/Canon, currently monitoring FINOS, OWASP, and NIST simultaneously. Its operation is strictly deterministic and human-gated:

```
External Authoritative Sources
  FINOS AI Governance Framework
  OWASP LLM Top 10
  NIST AI RMF
  RBI FREE-AI Recommendations
  EU AI Office Guidance
  CFPB / SEC Examination Priorities
          |
          v
Canonical Source Monitor
  Watches versioned public rule repositories
  Detects additions, modifications, deprecations
          |
          v
Semantic Diff + Evidence Generation
  Structured diff: rule ID, old text, new text,
  severity change, regulatory mapping impact
          |
          v
Human Review + Approval Gate
  Compliance officer or designated maintainer
  No automated policy change without explicit approval
          |
          v
Signed Constitution Update
  SHA-256 sealed, version-controlled, individually attributed
  GOVERNANCE.lock updated: tamper-evident record of change
          |
          v
Anchor Static -> AnchorJIT -> Runtime -> Decision Audit Chain
```

Three properties of Canon are critical to its governance credibility:

- **No AI makes policy.** Canon is a deterministic evidence-production system, not a policy reasoning system. It detects that FINOS changed Rule 4 from WARNING to CRITICAL severity and packages that observation into a structured evidence record - source, version, diff, regulatory mapping impact. It does not decide whether that change should be adopted. That decision remains with a human approver, individually attributed and logged.
- **Changes are verifiable.** Every update to the constitution passes through GOVERNANCE.lock, producing a cryptographic record of what changed, when, who approved it, and what the previous state was. An institution can demonstrate to a regulator, on demand, the exact governance rules in effect at any point in time - not from a policy document describing intended behaviour, but from a cryptographically verifiable record of actual state.
- **Staleness is detectable.** Canon tracks the version state of each external source. If an institution's constitution references OWASP LLM Top 10 v1.1 but v1.2 has been published and not reviewed, Canon flags this as an unresolved governance debt - surfacing the gap before a regulator or auditor does.

The complete governance architecture, integrating Canon with Anchor's existing layers and the proposed AnchorJIT ante-hoc enforcement layer, is as follows:

```
GOVERNANCE LIFECYCLE

Phase 1 - Policy Evolution (Canon)
  External source monitoring, semantic
  diff, human approval, signed
  constitution update
          |
          v
Phase 2 - Static Governance (Anchor Layer 1
  Pre-deployment AST analysis
  Capability signals, not syntax verdicts
  CI/CD gate: blocks deployment on BLOCKER findings
          |
          v
Phase 3 - Ante-hoc Enforcement (AnchorJIT) [research
  direction] Inference-time governance constraints
  Active at tool calls, structured outputs, permission boundaries
          |
          v
Phase 4 - Runtime Governance (Anchor Layer 2)
  Per-decision interception: actor, policy version, hashes
  Structured Mode: ReasonCode + FeatureAttribution enforced
          |
          v
Phase 5 - Audit and Reconstruction (Decision Audit Chain - DAC)
  Hash-chained, HMAC-signed, append-only
  eu_article12_record() | cims_payload() | adverse_action_reasons()
```

This architecture covers the full governance lifecycle: how policies are kept current (Canon), how code is verified before deployment (static), how generation is constrained before output (AnchorJIT), how decisions are intercepted and evaluated at runtime (runtime), and how a verifiable record is produced for regulatory examination (DAC). We believe the FSB sound practices, in their current form, address Phases 2, 4, and 5 but leave Phases 1 and 3 without explicit guidance.

Q4 - Flexibility for Newer AI Types Over Time

The sound practices are technology-neutral in framing, which is appropriate. However, technology-neutral principles require technology-specific implementation guidance to remain actionable as AI architectures evolve.

We suggest the FSB adopt a **regulatory polyglottism** approach: rather than maintaining separate guidance for each AI type, define a core set of governance properties - tamper-evident logging,

per-decision explainability, cryptographic policy integrity, execution authority verification - that any AI system must satisfy regardless of architecture.

Our published research demonstrates that a single governance record, structured with the right fields (reason code, feature attribution, chain hash, policy version, actor identity), can simultaneously satisfy EU AI Act Article 12, RBI FREE-AI Recommendation 7, CFPB Regulation B adverse action requirements, and SEC 2026 audit priorities - from one write operation. The FSB sound practices could formalise this convergence by specifying the minimum field set for a governance-credible AI decision record, leaving implementation to institutions while ensuring the records produced are cross-jurisdictionally legible.

Q5 and Q6 - Case Studies

The existing case studies are well-chosen for illustrating operational AI use. We offer one additional case study relevant to non-banks and the agentic AI risk discussion.

Proposed case study: Governance infrastructure at a lending platform.
A lending platform operating under CFPB Regulation B and processing AI-assisted credit decisions faces the following concrete challenge: each denied application must be explainable at the individual level, under the specific policy in effect at the time of decision, in a form that satisfies both the CFPB's adverse action notice requirements and the EU AI Act's Article 12 logging requirements if the platform operates in European markets.

Without purpose-built governance infrastructure, this institution must: (a) reconstruct individual decision reasoning from model logs after the fact; (b) maintain separate compliance documentation for each jurisdiction; and (c) manually verify that the policy in effect at decision time has not been modified since. Each step is slow, expensive, and error-prone.

With Anchor's governance layer deployed: (a) every credit decision automatically produces a structured AuditEntry containing a machine-readable ReasonCode and FeatureAttribution at decision time; (b) the same record is rendered into jurisdiction-specific formats via *adverse_action_reasons()* and *eu_article12_record()* without separate implementations; (c) the GOVERNANCE.lock mechanism provides cryptographic proof that the policy in effect at decision time was the sealed, approved policy. With Canon deployed, the institution also has a verifiable record that its governance rule set was current against CFPB, EU AI Office, and RBI guidance at the time of each decision - closing the policy supply chain gap.

A governance assessment of this kind - running the static analysis layer against the institution's AI-adjacent codebase, mapping findings to the relevant regulatory framework, and producing a findings brief - can be completed in three to five business days with read-only access and no production system changes.

Q7 - Glossary Definitions

The glossary is generally clear. We suggest two additions relevant to the agentic AI discussion:

Execution authority.

The set of actions an AI system is explicitly permitted to perform, under a specific policy version, as applied to a specific actor identity and context. Distinct from model capability (what the system can do) and model output quality (how accurately it performs). Execution authority is a governance property, not a model property.

Constitutional governance.

A governance approach in which the rules constraining an AI system's behaviour are defined in a single, cryptographically signed document (a 'constitution') that is enforced both before deployment (via static analysis) and during runtime (via decision interception), with a tamper-evident record that the same constitution governed both layers. This term distinguishes governance approaches that

enforce a unified, verifiable policy from those that rely on separate, potentially inconsistent controls at different lifecycle stages.

Summary and Availability

AnimusLab's Anchor governance engine is available as open-source software under the Apache 2.0 license (*pip install anchor-audit*; github.com/AnimusLab/Anchor). The full technical architecture, empirical validation results, and regulatory mapping described in this response are documented in our published preprint: *Constitutional Governance for Autonomous Agents in Financial Services: Three Failure Modes and a Deterministic Infrastructure Response* (Zenodo, DOI: 10.5281/zenodo.19734724).

Canon is available as a working open-source prototype (v0.1.0, Apache 2.0) at github.com/AnimusLab/Canon. The implementation simultaneously monitors FINOS AI Governance Framework, OWASP LLM Top 10, and NIST AI RMF for rule changes, produces cryptographically signed evidence packages on any detected change, and routes each package to a human approval gate before any constitution update occurs. No automated policy change is possible without explicit human attribution - the governance policy supply chain is deterministic by construction.

Empirical benchmarks on Canon v0.1.0 demonstrate that governance policy integrity operations add negligible overhead to any financial institution's AI deployment workflow:

Operation	Mean	P95	P99	Throughput/s
SHA-256 source state hash (100 rules)	135µs	155µs	253µs	7,388
Diff engine - 100 rules, 5 changes	88µs	104µs	176µs	11,348
Diff engine - 500 rules, 20 changes	398µs	548µs	834µs	2,511
Evidence package hash generation	12µs	12µs	19µs	81,497
Ledger chain hash computation	1.3µs	1.2µs	1.4µs	744,990
Approval record hash (tamper-evident)	3.3µs	3.4µs	3.7µs	299,043
End-to-end: diff + evidence + ledger entry	490µs	734µs	1113µs	2,040

Python 3.11 / Windows / 1000 iterations (200 for E2E). Canon v0.1.0.

The cryptographic operations that produce tamper-evident governance records - chain hashing and approval record signing - complete in under 4 microseconds. The dominant cost is the diff engine itself, which scales with rule set size: 88 microseconds for a 100-rule governance framework, 398 microseconds for 500 rules. These figures represent the complete cost of detecting a governance rule change, producing a signed evidence package, and writing a hash-chained ledger entry - the operations that enable an institution to demonstrate, cryptographically, that its governance rule set was current and unmodified at any point in time. AnchorJIT remains a current research direction; design specifications are available on request.

We welcome correspondence from FSB member authorities, financial institutions, and fellow researchers at **tan@animuslab.dev**.