Nikxius Inc.

NIKXIUS RESEARCH · SEPTEMBER 2026

Financial AI Governance: From Evaluation Results to Decision Evidence

A proposed claim–evidence–owner–decision method for reviewing a bounded AI system

REPORT
NX-RESEARCH-GOVERNANCE-2026-01

DATE
2026-09-26

VERSION
1.0

CLASSIFICATION
PUBLIC

ABSTRACT

Financial AI governance needs evidence that answers a particular decision, not an undifferentiated collection of documents. This paper proposes a claim–evidence–owner–decision matrix and distinguishes integrity, provenance, truth, coverage and acceptance through a public synthetic example.

Contents

Abstract

An evaluation report becomes useful governance evidence only when a reviewer can connect its findings to a specific claim, system version, intended use and decision. A hash does not establish the truth of a statement. A passing check does not establish adequate coverage. An accepted report does not establish that the underlying system is safe. This paper proposes a claim–evidence–owner–decision method that keeps those distinctions visible. We apply it to a published synthetic loan-servicing example in which the candidate improves several behaviors while introducing two hardship failures. The method preserves missing evidence, assigns unresolved questions to accountable owners and records human exceptions without rewriting the evaluation. Primary guidance from NIST and the FCA supports contextual evaluation and explicit responsibility; existing evaluation and release tools supply much of the necessary infrastructure. The proposed matrix is an engineering aid, not a regulatory mapping, certification scheme or empirically validated institutional review process.

Nikxius Research · 26 September 2026 · Version 1.0 · Prepared with AI assistance. This technical working paper has not been peer reviewed.

Research questions and evidence selection

We ask three questions. What makes an artifact relevant to an AI governance decision? Which properties can a reviewer infer from an evidence pack, and which need separate support? How should a team retain failures and uncertainty when an authorized person accepts a bounded exception?

The paper uses a selective review of primary framework, regulator, academic and product sources, refreshed on 26 September 2026. We inspected the relevant NIST AI RMF, FCA AI Live Testing and CSA AI Controls Matrix material, Raji and colleagues' audit framework, and Palantir, LangSmith and GitHub documentation. These sources establish the published practices or capabilities attributed to them. They are not an independent assessment of any vendor's deployment, a representative survey of financial institutions or evidence that an institution will accept this paper's method.

The worked example analyzes the existing public synthetic assurance pack. It contains invented servicing policy, a simulated responder and constructed cases. No financial institution, customer record, payment, credit decision or live model service is involved. Its human-review examples are also simulated. We report no interviews, purchasing outcomes or new customer experiment.[1]

The matrix and the distinctions developed below are proposed synthesis. They can be implemented in ordinary documents and existing tools. Their usefulness and operating cost would need evaluation with actual reviewers before claiming institutional effectiveness.

Related work and the scope of a governance decision

NIST AI RMF connects context, measurement and management. MAP 1.1 concerns intended purposes and prospective deployment settings. GOVERN 2.1 calls for clear, documented roles and responsibilities. MEASURE 2.1 concerns documented test sets, metrics and tools. MANAGE 1.1 asks whether the system achieves its intended purposes and whether development or deployment should proceed. These are related functions, not interchangeable pieces of evidence.2

The FCA's AI Live Testing programme likewise describes a system through its model, deployment context, risks, governance, human oversight, evaluation and input/output controls. Its terms state that the programme is exploratory and does not cover AI auditing, technical certification or regulatory compliance with other frameworks.34 This combination is instructive: broad system evaluation can be valuable while its conclusions remain carefully bounded.

CSA's AI Controls Matrix supplies another organizing vocabulary. Its current v1.1 publication describes control objectives, applicability and ownership, and an AI-CAIQ questionnaire for self-assessment or third-party evaluation. Such a framework can help structure questions. Completing a questionnaire or mapping a document to a control does not demonstrate that the control operated as claimed.5

Raji and colleagues' internal algorithmic auditing framework provides relevant academic context. It connects organizational accountability with documents produced across development and distinguishes technical reliability from broader questions about harms and declared expectations.6 Our matrix is a narrower aid for tracing one decision. It is not a replacement for their end-to-end framework, nor evidence that a document workflow alone makes an organization accountable.

A concrete decision should therefore begin with an object and a purpose: whether a specified version may explain servicing policies to a defined user group, with particular tools and escalation arrangements, under stated conditions. "Approve our AI" is too broad. So is an evidence request that never says whether the reviewer is considering deployment, a change, a vendor relationship or an incident. Different decisions require different evidence and may have different accepting owners.

The proposed claim–evidence–owner–decision matrix

We propose one row for each material claim that the decision depends on. A claim is a statement capable of being supported, contradicted or left unresolved. “Responsible AI” is a topic. “Unauthenticated users cannot obtain account balances through this version’s account-lookup path” is a bounded claim whose mechanism and evidence can be examined.

Each row records the claim’s scope, relevant artifact, evidence producer, person or role accountable for the underlying control, accepting reviewer, current finding and resulting decision. The producer and owner may be the same in a small team, but that relationship should be explicit. Where independent review is required by the organization’s actual process, assigning the same person two labels does not create independence.

A system declaration links to a frozen campaign, observations, an evaluation gate and a separate human review with conditions and an owner.

Figure 1. An evaluation-to-decision trace that can support the matrix. Changing a human review does not rewrite observations or the original gate. The links do not guarantee that the reviewer is independent.

Bounded claim	Appropriate evidence	Accountable role to identify	Decision if evidence is absent or adverse
The evaluated configuration is the intended release	Version record, release artifact identity and deployment observation	Application/release owner	Resolve the identity mismatch before relying on the evaluation
A stated behavior meets a defined rule	Frozen cases, evaluator definition, observations and findings	Domain/control owner	Remediate, restrict the relevant use or record an explicit exception
Testing covers the important use conditions	Sampling rationale, scenario inventory and known omissions	Evaluation owner with relevant domain review	Identify additional coverage; do not convert absence into PASS
A required human handoff works	Routing evidence plus evidence of the operational receiving process	Servicing/operations owner	Treat response capacity and actual receipt as unresolved until supported
The record has not changed relative to the retained reference	Artifact digests, manifest and an appropriately established reference	Evidence custodian	Investigate mismatched or unavailable artifacts
A person accepted the stated residual conditions	Authenticated decision, exact run reference, rationale and conditions	Authorized accepting owner	Keep the pack unreviewed; do not infer approval from its existence

These roles are examples to identify in the actual organization, not titles prescribed by this paper. A regulator, customer reviewer and internal deployment owner can ask different questions about the same artifact. Their acceptance should be recorded separately when it concerns different decisions.

The matrix also prevents document-count inflation. One report may support several claims, but repeating the report link does not create independent corroboration. Conversely, a single claim may need a configuration record, a behavioral test and an operational observation. The appropriate number of artifacts follows from the claim, not from a target size for the evidence pack.

Five properties that must remain separate

Integrity concerns whether retained bytes match a reference. A cryptographic digest is useful for detecting a change relative to a trusted prior value. If an attacker can replace both a file and the accompanying manifest, matching digests alone do not

expose that replacement. A signature can add attribution under an accepted key and signing process, but its meaning still depends on how that key became trusted.

Provenance concerns where the record came from and how it was produced. A result should identify its system and version, case, evaluator, run and relevant time. The producer's statement that a configuration was loaded is different from independent evidence of the running environment. A detailed lineage can be accurate about a simulation while saying nothing about a production deployment.

Truth concerns whether the substantive assertion is correct. A perfectly preserved report can contain a mistaken policy interpretation, a fabricated observation or a flawed expected label. Integrity checking does not validate those assertions. The accepting reviewer needs evidence appropriate to each statement: authoritative policy for the policy claim, trustworthy execution observation for the execution claim, and suitable domain review for the interpretation.

Coverage concerns what was examined and what remains outside the evidence. A finite test set necessarily leaves some inputs and conditions untested. Coverage depends on the intended use, risk analysis, sampling design, affected groups and operational environment. A long report with many correlated checks can still miss a single important scenario. NIST explicitly couples measurement with deployment-like conditions and documented limits to generalization.[2]

Acceptance concerns an authorized person's decision that the evidence is sufficient for a stated purpose, potentially under conditions. Acceptance is an institutional act, not a property produced automatically by a checksum or benchmark score. It can be mistaken or later invalidated by new evidence. Recording it clearly improves accountability without making the accepted claim infallible.

NIKXIUS RESEARCH / FIGURE 04

Three boundaries on an assurance claim

WHAT THE ARTIFACT SUPPORTS		WHAT DOES NOT FOLLOW
Observed test results Results apply to tested cases, versions and conditions.	≠	**Comprehensive system safety**
Matching artifact hashes Integrity checks depend on the source and trust boundary.	≠	**Truthful and complete evidence capture**
A recorded human review The record does not replace applicable law or institutional judgment.	≠	**Regulatory compliance or acceptance**

Conceptual assurance limits. These are distinct claims with distinct evidence requirements.

Three assurance limits: observed test results do not establish comprehensive safety, matching hashes do not establish truthful complete capture, and a recorded review does not establish regulatory compliance or acceptance.

Figure 2. Three consequences of keeping the evidence questions separate. A narrow verification result cannot establish the broader conclusion shown beside it.

These distinctions explain why “verified evidence” needs a qualifier. Verified artifact integrity, verified reviewer identity and independently corroborated behavior are different statements. A useful report specifies which verification was performed, against what reference, and with what remaining dependencies.

Worked example: what the synthetic pack supports

The published candidate is a synthetic loan-servicing assistant. Its permitted job is to explain invented policies and route disputes or hardship to a person. It performs no credit decisions or financial actions. Six fixtures and ten definitions produce 26 observations per version. The candidate fixes eight failing checks, introduces two hardship failures and moves a dispute check from FAIL to REVIEW_REQUIRED. Both evaluation gates remain FAIL.[1]

The regression paper explains the exact transitions and their comparability. For governance, the question is what the observations allow an owner to conclude. Consider three rows:

Claim under review	Observed evidence	Supported conclusion and remaining gap
The candidate returns the labelled synthetic late fee	Its fee response and relevant deterministic checks reach PASS	The response matches that constructed policy case. This does not establish correctness across real accounts, exceptions or jurisdictions.
The candidate routes hardship to a person	Hardship assertion and human-review checks return FAIL	The constructed hardship handoff is broken. Improved unrelated checks do not cancel this finding.
The candidate's dispute response completes the customer's dispute	The escalation check changes from FAIL to REVIEW_REQUIRED	The simulated response now escalates. Neither specialist receipt nor resolution of a real dispute was observed.

This formulation resists a common evidence substitution: using a routing flag to support a claim about operational completion. A queue name and `escalated` value can show what a simulated application emitted. They cannot establish staffing, delivery, response time or customer outcome. The next evidence depends on the actual intended handoff: queue acceptance, access permissions, routing ownership and a suitable exercise of the receiving process.

The public unreviewed pack makes another useful distinction. It contains evaluation findings without pretending that a real risk owner accepted them. Its downloadable versions with simulated dispositions illustrate record structure, not institutional decisions. The example supports the feasibility of representing separate findings and decisions; it does not validate a bank's acceptance methodology.

Missing evidence and human exceptions

Missing evidence deserves a typed explanation. "Not tested," "artifact unavailable," "result interrupted," "outside declared scope," "source not authenticated" and "review pending" lead to different next steps. Collapsing them into one empty field makes it difficult to tell whether the team needs another test, a retrieval step, a scope correction or a decision.

An accountable exception should retain the original adverse finding and explain why a bounded action is nevertheless being authorized. Under our proposed method, its record includes the exact version and run; the affected claims and findings; the authorized use; the reviewer's identity and authority; rationale; compensating

conditions; expiry or reconsideration trigger; and responsibility for closure. This is a proposed minimum record, not evidence that any particular organization's process permits the exception.

A restriction must also exist outside the document. If a reviewer permits information-only use while hardship handling remains defective, the deployment must actually enforce the permitted boundary and provide an adequate alternative. Writing "restricted use" in a report does not prevent the disabled workflow from being reached. Evidence of restriction enforcement is therefore a separate row.

Exceptions should not rewrite prior exports. A new decision can reference the old run, and new evidence can support a later review. Historical findings remain identifiable. This makes it possible to reconstruct what was known at the time rather than showing only the most favorable current summary. It also avoids treating the eventual fix as proof that the earlier release had already satisfied its conditions.

Existing tools and the smallest sufficient implementation

The required information is often distributed across a model or prompt registry, evaluation service, source repository, issue tracker and deployment process. The practical task is to preserve their relationship. A new evidence platform is not automatically necessary.

Palantir AIP Evals documents evaluation suites and comparison with earlier function versions. LangSmith documents datasets, experiments and human/code/model evaluation. GitHub environments support reviewer and deployment protection controls, including configurable self-review and administrator-bypass behavior.[7][8][9] These capabilities can form a credible native implementation if they cover the actual decision and their configuration is understood.

For a single application, a versioned matrix linked from an existing release record may suffice. Keep authoritative observations in their source system, retain an export when needed for later review, and record the acceptance against the exact version. Avoid copying every underlying record into a new repository unless retention, access or portability requires it. Unnecessary duplication creates its own stale-data and disclosure risks.

The comparison criterion is the same for an internal workflow and a commercial product: can the reviewer resolve each material claim, find its exact evidence, understand gaps and reconstruct the decision? Published feature descriptions cannot answer whether a particular institution's workflow passes that test. Neither does the presence of a cryptographic export format establish a need for another supplier.

Failure and threat analysis

Failure or threat	Why the pack can mislead	Proposed response
Replaced file and manifest	Internal hashes can remain consistent after replacement	Establish an independently retained reference where authenticity matters; state the trust model
Incorrect expected labels	Every check can agree with a mistaken rule	Review labels and authoritative policy independently of the application change
Stale system mapping	A valid report concerns a different running configuration	Verify the version-to-deployment link at the relevant decision point
Removed or weakened tests	The apparent pass count improves through reduced scrutiny	Compare coverage and evaluator definitions before comparing outcomes
Simulated or self-approved review	A signed-looking field is mistaken for authorized acceptance	Authenticate the actor and verify the actual review policy, including independence requirements
Evidence assembled from one favorable run	Adverse observations disappear from the decision	Define inclusion and exclusion rules before selecting the reported results
Sensitive material copied into reports	Portability spreads data beyond its intended audience	Minimize retained content, control access and use explicit redaction provenance
Accepted exception outlives its conditions	A historical decision is reused after the context changes	Record expiry, change triggers and an owner for reassessment

The privacy row does not imply that indiscriminate logging is the answer to incomplete evidence. Retain enough to support the stated claim under the authorized data boundary. If content is redacted, disclose what the redaction changes about reproducibility. A report that protects sensitive data may legitimately support a narrower public conclusion than its restricted internal counterpart.

Limitations and practical procedure

This paper has not tested the matrix with institutional reviewers, measured review time, established inter-reviewer agreement or demonstrated lower operating cost. It provides no regulatory compliance mapping, legal interpretation or certification. The primary sources support the attributed principles and capabilities; they do not endorse the proposed method. The synthetic service's small, deliberately constructed coverage cannot validate a real financial application.

To use the method, begin with the actual decision and its accepting owner. Write the claims that must hold for that decision, identify existing authoritative artifacts, and distinguish missing support from adverse support. Check version and context before reviewing results. Examine the weakest material claim directly rather than averaging it into a document-completion percentage.

Next, assign each unresolved item to the person who can produce evidence or make the decision. Preserve the evaluation result when recording any exception. Confirm that promised restrictions are enforced in the relevant system. Retain the decision and its referenced artifacts under the organization's approved information-handling process, with explicit triggers for reconsideration.

Financial AI governance evidence is strongest when it makes the boundary of knowledge easy to inspect. The public demonstration supplies a small worked example. The companion agentic AI safety paper examines what additional evidence is needed when an agent can change external state. Further material is available in the research collection.

References

1. Nikxius Research, Public synthetic assurance capture, 26 September 2026, schema `nikxius-public-assurance-demo/1`. Fixture, manifest, unreviewed report. Six constructed cases and 26 observations per version, with simulated identities and decisions; no customer data or external model calls. Existing capture, not a new experiment performed for this paper.↩ ↩

2. National Institute of Standards and Technology, Artificial Intelligence Risk Management Framework (AI RMF 1.0), NIST AI 100-1, January 2023. Relevant sections: GOVERN 2.1; MAP 1.1; MEASURE 2.1, 2.3 and 2.5; MANAGE 1.1. Official publication. Refreshed 26 September 2026. Voluntary framework; this paper does not assert conformance.↩ ↩

3. Financial Conduct Authority, AI Live Testing: How it can support safe and responsible AI deployment, dated January 2026. Relevant section: "What we're actually testing." Original article. Refreshed 26 September 2026. Description of an exploratory programme, not a universal obligation.↩

4. Financial Conduct Authority, AI Live Testing: Revised Terms of Reference, public document linked by the programme article; no printed publication date established. Relevant sections: scope, approach, framework validation and system testing. Public terms. Refreshed 26 September 2026. Explicitly excludes auditing, technical certification and regulatory compliance with other frameworks.↩

5. Cloud Security Alliance, AI Controls Matrix v1.1, released 22 June 2026. Relevant sections: control objectives, applicability and ownership, AI-CAIQ and guidance contents. Publication page. Refreshed 26 September 2026. The overview was inspected; this paper does not claim a control-by-control framework audit or legal mapping.↩

6. Inioluwa Deborah Raji, Andrew Smart, Rebecca N. White, Margaret Mitchell, Timnit Gebru, Ben Hutchinson, Jamila Smith-Loud, Daniel Theron and Parker Barnes, Closing the AI Accountability Gap: Defining an End-to-End Framework for Internal Algorithmic Auditing, FAT* 2020, DOI 10.1145/3351095.3372873. Relevant sections: introduction, governance/accountability and the proposed audit framework. Public author manuscript. Refreshed 26 September 2026. Proposed organizational framework; not a validation study of the matrix in this paper.↩

7. Palantir, AIP Evals: Overview, undated living documentation. Relevant sections: introduction and key concepts. Documentation. Refreshed 26 September 2026. Published capability, not an independently tested deployment.↩

8. LangChain, LangSmith Evaluation, undated living documentation. Relevant sections: evaluation types and workflow. Documentation. Refreshed 26 September 2026. Published capability, not evidence of institutional acceptance.↩

9. GitHub, Deployments and environments, undated living documentation. Relevant sections: required reviewers, self-review prevention and administrator bypass. Documentation. Refreshed 26 September 2026. Availability and enforcement depend on plan and configuration.↩

Source method

Selected primary NIST, FCA, CSA, academic and native-tool sources refreshed on 26 September 2026; analysis of published synthetic assurance artifacts.

Provenance. Nikxius Research, prepared with AI assistance. Technical working paper; not peer reviewed.