OTANIS: An Operational Trust and Authority Normative Integrated System for Executable Governance of Agentic AI

Masayuki Otani
AI Consultant Insights (AICI)
United Kingdom
Email: info@aiconsultantinsights.com

***Abstract*—As AI systems evolve from bounded inference engines into live agentic architectures capable of initiating irreversible actions, governance failures increasingly occur despite local compliance of individual components. These failures are structural rather than behavioural and arise when legitimate authority cannot be demonstrated or enforced at execution time. This paper introduces OTANIS, an Operational Trust and Authority Normative Integrated System for executable governance of agentic AI. OTANIS unifies ex-ante admissibility, runtime authority enforcement, authority lifecycle semantics, compositional preservation, conflict escalation, and multi-layer governance into a single architectural scheme. Authority is treated as a first-class executable object with explicit validity, revocation, refusal, fallback, and audit requirements enforced at the irreversibility boundary. The framework is model-agnostic, falsifiable, and scoped specifically to live agentic systems. Formal definitions, atomicity semantics, termination guarantees, provenance requirements, probabilistic latency handling, and integrity-based audit criteria are provided. OTANIS is proposed as a reference architecture for good practice in the design, review, and audit of agentic AI systems producing irreversible outcomes.**

***Index Terms*—Agentic AI, architectural governance, authority, irreversibility, execution-time control, compositional systems, auditability.**

I. Introduction

AI governance has largely been addressed through principles, alignment strategies, and risk management frameworks. While valuable, such approaches implicitly assume bounded systems with stable loci of authority. This assumption no longer holds for modern deployments, which increasingly consist of live agentic workflows composing models, tools, APIs, and actuators across organisational and regulatory boundaries.

A recurring failure mode is observed in practice. Individual subsystems satisfy their local constraints, yet the composed system produces an illegitimate outcome. At the moment that outcome becomes irreversible, legitimate authority cannot be demonstrated as an enforceable object. Responsibility fragments, and governance collapses.

Prior work formalised this failure mode by defining governance as an executable control structure operating at the point of irreversibility rather than as policy or post-hoc explanation [1]. This paper extends that work by introducing OTANIS, a unified architectural system that integrates admissibility, enforcement, authority lifecycle semantics, compositional preservation, conflict handling, and multi-layer governance into a standards-grade scheme.

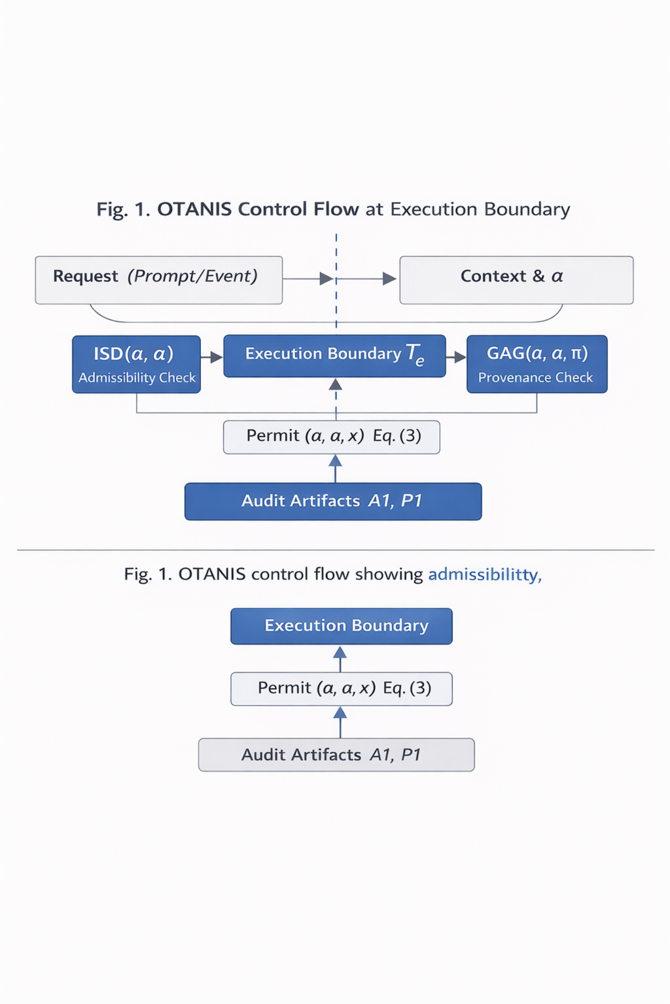

Fig. 1. OTANIS Control Flow at Execution Boundary. The diagram illustrates how admissibility (ISD) and provenance (GAG) feed into the final Permit decision at T_e.

II. Scope and Non-Claims

A. Scope

OTANIS applies to live agentic systems that autonomously or semi-autonomously initiate actions producing irreversible external effects, including financial settlement, legal commitment, physical actuation, access-control modification, or irreversible data disclosure.

B. Out of Scope

Systems without a meaningful irreversibility boundary, such as offline analytics, batch inference, or purely advisory copilots, are out of scope.

C. Non-Claims

OTANIS does not guarantee safety, correctness, alignment, or regulatory compliance. It does not certify systems or vendors. It provides an architectural control structure that makes authority enforceable, refusals deterministic, and outcomes auditable at execution time. This stance aligns with established security engineering principles that protection must be architectural and enforceable rather than aspirational [2].

III. Architectural Premise

OTANIS is founded on the following premise.

P1. If legitimate authority cannot be shown and enforced at the moment an outcome becomes irreversible, governance did not exist for that action path.

This premise is consistent with the principle of complete mediation, which requires that every protected operation be checked at the moment it occurs [2]. In agentic systems, the protected operation is the irreversible action itself.

IV. Formal Model and Notation

Let an agentic system be represented as a directed graph

$$S = (V, E) \tag{1}$$

where V denotes agents, tools, services, and control components, and E denotes invocation or data-flow edges.

Let A be the set of all actions the system may attempt.

Let the time domain T be a totally ordered set representing execution events. T may be implemented using physical time, logical clocks, or event-ordering mechanisms, provided that $T_e(a)$ is deterministically resolvable at the execution boundary.

A. Execution Boundary

Define an execution boundary function

$$T_e : A \to T \tag{2}$$

where $T_e(a)$ is the earliest time at which executing action a produces an irreversible effect.

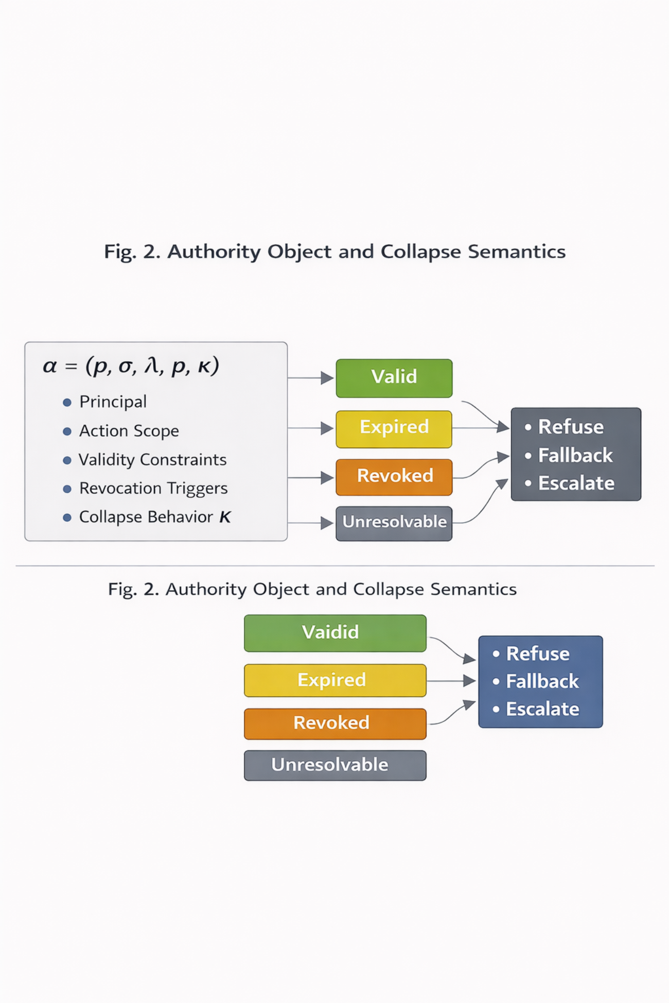

Fig. 2. Authority Object and Collapse Semantics. Visualizing the components of α defined in Equation (3) and their mapping to collapse behaviors κ.

B. Authority Object

Define an authority object

$$\alpha = (p, \sigma, \lambda, \rho, \kappa) \tag{3}$$

where:

- p is the originating principal,
- $\sigma \subseteq A$ is the authorised action scope,
- λ defines lifecycle constraints such as expiry and contextual validity,
- ρ defines revocation triggers,
- κ defines collapse behaviour, including refusal, fallback, or escalation.

This structure generalises attribute-based access control to irreversible external actions and lifecycle collapse [3], [4].

C. State Responsibility and Minimality

Let x denote the full system state at time t. OTANIS does not require exhaustive capture of x.

Define $x_\lambda \subseteq x$ as the minimal state subset upon which authority lifecycle predicates depend.

Axiom S1 (State Minimality). x_λ must contain all state elements upon which $\text{Valid}_\lambda(\alpha, x, t)$ or $\text{Revoked}_\rho(\alpha, x, t)$ functionally depend. Omission of any such dependency constitutes non-compliance.

OTANIS governs the explicit binding and auditability of authority-relevant state, not the correctness of the full system state.

V. Ex-Ante Admissibility and the Governance Oracle

Define an admissibility predicate

$$\text{ISD}(a, \alpha) \in \{0, 1\} \tag{4}$$

which evaluates whether authority for action a is normatively admissible ex-ante.

A. Governance Oracle Constraint

Rule O1. Evaluation of $\text{ISD}(a, \alpha)$ must be performed by a mechanism that is:

1) deterministic and non-agentic, or
2) governed by a higher-order OTANIS instance, or
3) explicitly authorised and frozen by human authority ex-ante.

Rule O1a (Termination). Any chain of OTANIS governance instances must terminate at a non-agentic deterministic evaluator or explicit human authority within a declared bounded depth D_{max}. Unbounded or cyclic governance recursion is non-compliant.

A frozen evaluator is one whose logic and parameters cannot self-modify, retrain, or adapt without invalidating the associated authority object and forcing re-authorisation.

VI. Runtime Authority Enforcement and Atomicity

Define lifecycle validity

$$\text{Valid}_\lambda(\alpha, x, t) \in \{0, 1\} \tag{5}$$

and revocation status

$$\text{Revoked}_\rho(\alpha, x, t) \in \{0, 1\} \tag{6}$$

A. Execution Permission

$$\begin{aligned} \text{Permit}(a, \alpha, x) \iff\ & \text{ISD}(a, \alpha) \\ & \wedge (a \in \sigma) \wedge \text{Valid}_\lambda(\alpha, x, T_e(a)) \\ & \wedge \neg\text{Revoked}_\rho(\alpha, x, T_e(a)) \end{aligned} \tag{7}$$

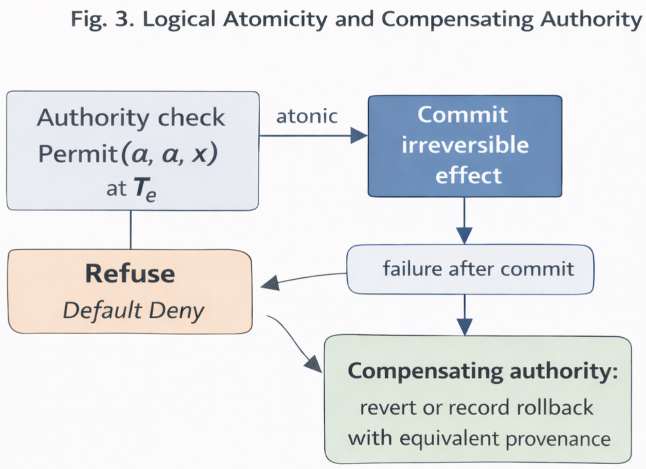
Fig. 3. Logical Atomicity and Compensating Authority
Authority check
Permit(α, a, x)
at T_e
atonic
Commit irreversible effect
Refuse
Default Deny
failure after commit
Compensating authority:
revert or record rollback with equivalent provenance

Fig. 3. Logical Atomicity and Compensation. Visualizing the transaction boundary and rollback path required by Rule R1'.

B. Logical Atomicity

Rule R1' (Logical Atomicity). Authority evaluation and action commitment at $T_e(a)$ must be logically atomic, meaning executed as:

- a single local transaction, or
- a bounded distributed transaction protocol with explicit rollback or compensating authority.

Definition (Compensating Authority). Compensating authority is a pre-authorised fallback mechanism that either reverts committed state or records the rollback with provenance guarantees equivalent to the original action [5].

If neither logical atomicity nor compensating authority can be guaranteed, execution must be refused.

C. Default Deny and Safe Halt

Axiom D1 (Default Deny). If no valid authority object α can be resolved at $T_e(a)$, or if any required authority attribute is missing, malformed, or unverifiable, the system must deterministically refuse execution and transition to a declared safe halt or fallback state.

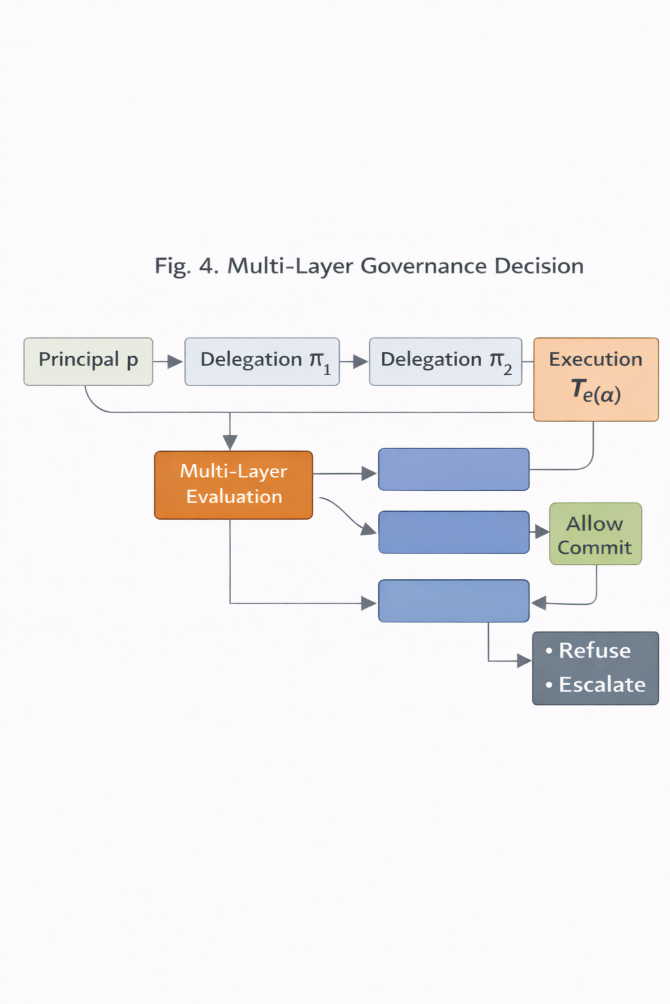

Fig. 4. Multi-Layer Governance Decision. Illustrating provenance delegation π feeding into a multi-layer evaluation block.

VII. Compositional and Multi-Layer Governance

A. Global Architectural Governance

Let π denote authority provenance evidence. Define

$$\mathrm{GAG}(a, \alpha, \pi) \in \{0, 1\} \tag{8}$$

GAG verifies that provenance π forms a valid, acyclic, and cryptographically bound chain from originating principal p to execution boundary $T_e(a)$.

Rule R2. Authority delegation graphs must be acyclic or explicitly bounded. Cyclic or self-referential authority paths are non-compliant.

B. Multi-Layer Governance

Let governance layers be indexed $i \in \{1, \dots, L\}$.

$$\mathrm{MGAG}(a, \{\alpha_i\}, x) = \bigwedge_{i=1}^{L} \mathrm{Permit}_i(a, \alpha_i, x) \tag{9}$$

Rule M1 (Conflict Escalation). If governance layers impose conflicting requirements, the system must refuse execution or escalate to a higher-priority authority explicitly defined ex-ante.

VIII. Compliance Levels

1) Level 1: Normative
2) Level 2: Runtime-Enforced
3) Level 3: Compositional
4) Level 4: Multi-Layer

Systems must meet the compliance level corresponding to the highest-risk irreversible execution path reachable under normal operation or specified fault scenarios.

IX. Provenance and Audit Artefacts

Requirement P1 (Provenance Properties). Provenance evidence π must be non-repudiable, tamper-evident, time-ordered, chain-verifiable, and bound to authority identifiers [6].

Requirement A1. For every executed action a, the system must record:

- authority identifier
- execution boundary timestamp
- bound state subset x_λ
- provenance evidence π
- enforcement outcome

X. Falsifiability

F1 (Authority Existence Failure). A system is non-compliant if authority at $T_e(a)$ is unknown, inferred, reconstructed, or unverifiable.

F2 (Authority Integrity Failure). A system is non-compliant if authority at $T_e(a)$ is stale, forged, or derived from corrupted x_λ, even if present.

XI. Latency Considerations

OTANIS introduces latency only at execution boundaries.

$$\Delta t_{\mathrm{MGAG}} \leq \sum_{i=1}^{L} \Delta t_i \tag{10}$$

Parallel evaluation, probabilistic latency bounds, and service-level objectives may be applied, provided refusal semantics are preserved. For example, an implementation may require P99 enforcement latency below a declared threshold and refuse execution on timeout, with such refusal logged per Requirement A1.

XII. Discussion

OTANIS differs from ethics-driven and policy-centric frameworks by treating governance as an executable architectural property. Unlike ISO/IEC 42001 or the EU AI Act, which specify organisational controls and oversight obligations, OTANIS specifies the execution-time mechanisms required to make such controls enforceable in agentic systems. OTANIS complements risk-management frameworks by supplying the authority enforcement, refusal, and audit mechanisms those frameworks assume but do not define [7], [8].

XIII. Conclusion

OTANIS provides a unified, formal, and falsifiable architectural system for governing live agentic AI systems that produce irreversible outcomes. By integrating admissibility, runtime enforcement, authority lifecycle semantics, compositional preservation, conflict escalation, integrity-based audit criteria, and probabilistic latency handling, OTANIS addresses structural governance failures observed in real deployments. It does not eliminate risk, but it makes authority explicit, enforceable, and auditable at the moment it matters.

References

[1] M. Otani, "Executable governance at the point of irreversibility," AI Consultant Insights (AICI), Technical Report, 2026.

[2] J. H. Saltzer and M. D. Schroeder, "The protection of information in computer systems," *Proceedings of the IEEE*, vol. 63, no. 9, pp. 1278–1308, 1975.

[3] E. Yuan and J. Tong, "Attribute based access control (ABAC) for web services," in *Proceedings of the IEEE International Conference on Web Services (ICWS)*, 2005.

[4] V. C. Hu, D. Ferraiolo, D. R. Kuhn, A. Schnitzer, K. Sandlin, R. Miller, and K. Scarfone, "Guide to attribute based access control (ABAC) definition and considerations," NIST, Tech. Rep. NIST Special Publication 800-162, 2014. [Online]. Available: https://nvlpubs.nist.gov/nistpubs/specialpublications/nist.sp.800-162.pdf

[5] H. Garcia-Molina and K. Salem, "Sagas," in *Proceedings of the 1987 ACM SIGMOD International Conference on Management of Data (SIGMOD '87)*, 1987, pp. 249–259.

[6] B. Schneier and J. Kelsey, "Secure audit logs to support computer forensics," *ACM Transactions on Information and System Security*, vol. 2, no. 2, pp. 159–176, 1999.

[7] National Institute of Standards and Technology, "Artificial intelligence risk management framework (AI RMF 1.0)," NIST, Tech. Rep. NIST AI 100-1, 2023. [Online]. Available: https://nvlpubs.nist.gov/nistpubs/ai/nist.ai.100-1.pdf

[8] OpenAI, "Practices for governing agentic AI systems," 2023. [Online]. Available: https://cdn.openai.com/papers/practices-for-governing-agentic-ai-systems.pdf