Authority Engineering AI Extension: Continuity & Context

Why Reconstruction Cannot Preserve Authority Under Distributed Mutation

Rob Caswell

Authority Engineering

July 2026

Rob Caswell | linkedin.com/in/rob-caswell3

Published by Cyber Terrain LLC | cyberterrain.ai

DOI: https://doi.org/10.5281/zenodo.21742328

CYBER TERRAIN

Publication Record

Authority Engineering AI Extension: Continuity & Context: Why Reconstruction Cannot Preserve Authority under Distributed Mutation

Author: Rob Caswell

Publisher: Cyber Terrain LLC

Published: 2026-07-01

Zenodo version: v1.0

DOI: https://doi.org/10.5281/zenodo.21742328

Website: https://cyberterrain.ai/papers/ae-ai-extension-continuity-context.pdf

Corpus Context

Corpus: Authority Engineering

Document ID: AE-AIX-0004

Object family: AE-AIX-0004

Publisher-imprint edition prepared for versioned republication.
The prior Zenodo record remains the citable record until its new version is published.

© 2026 Cyber Terrain LLC. All rights reserved.

Abstract

Artificial intelligence systems increasingly rely on persistent memory, retrieval-augmented generation (RAG), conversational history, long-context architectures, and distributed agent execution to maintain coherent behavior over time. These mechanisms are commonly treated as solutions to continuity because they preserve or recover information across execution boundaries. However, preserving information is not equivalent to preserving authority. Once execution has diverged through distributed mutation, centralized semantic reconstruction cannot guarantee that recovered behavior remains consequence-faithful to the authority that originally governed realization.

This paper establishes the Continuity & Context branch of the Authority Engineering AI Foundations. Building exclusively from the Authority Engineering foundation, it demonstrates that the governing architectural constraint is not context preservation itself but the structural limit of reconstruction. The paper identifies AE-REG-0005 Reconstruction Dependency Collapse as the foundational dependency governing this branch and derives AI-DL-004 Reconstruction Limit as its principal architectural root. From this root follow the derived consequences of semantic insufficiency, partial visibility realization, unresolved continuity, and provenance mutation tolerance.

The resulting framework reframes contemporary AI techniques such as retrieval-augmented generation, persistent memory, context windows, workflow state, and multi-agent execution as engineering responses to a deeper architectural limitation rather than as independent solutions to continuity. The paper concludes that consequence fidelity must be preserved through continuity rather than reconstructed through semantics after distributed execution has already diverged.

Keywords: Authority Engineering, Artificial Intelligence, Continuity, Context, Reconstruction, Consequence Fidelity, Provenance, Distributed Systems, AI Governance

Rob Caswell | linkedin.com/in/rob-caswell3

DOI: https://doi.org/10.5281/zenodo.21742328

I. Introduction

Artificial intelligence increasingly operates across distributed environments in which execution unfolds over extended periods of time, spans multiple systems, and depends upon evolving contextual information. Modern architectures employ conversational history, retrieval systems, persistent memory, workflow state, and increasingly large context windows to preserve coherent behavior across these distributed execution environments. As these capabilities mature, discussion surrounding AI continuity has become increasingly centered on methods for recovering information rather than on the structural conditions required to preserve authority.

This distinction is fundamental. Information may often be reconstructed after execution has diverged, while authority cannot necessarily be reconstructed once consequence-bearing realization has already occurred. Semantic similarity, historical recall, and contextual completeness may improve the appearance of continuity without preserving the governing authority responsible for producing those consequences. Consequently, the architectural question is not whether information can be recovered, but whether consequence fidelity can survive distributed mutation through reconstruction alone.

Authority Engineering approaches this problem by treating continuity as a structural relationship governing whether consequence remains faithfully bound to originating authority throughout realization. Context is not continuity itself, but an externally observable manifestation. Reconstruction, in turn, represents a post-hoc attempt to establish historical authority after direct continuity carriage is no longer structurally demonstrable.

This paper establishes the Continuity & Context branch of the Authority Engineering AI Foundations. It demonstrates that the branch is governed by the structural limits of reconstruction rather than by semantic recovery. Beginning with the meaning-control definitions of consequence fidelity and continuity, it derives AE-REG-0005 Reconstruction Dependency Collapse as the governing architectural dependency and establishes AI-DL-004 Reconstruction Limit as the principal AI root of the family. The resulting architecture explains why semantic recovery, partial observation, provenance mutation, and unresolved continuity emerge naturally within distributed AI systems and provides the foundation upon which subsequent papers examine topology, reachability, delegation, and governance.

II. Consequence Fidelity

Authority Engineering defines consequence as the irreversible realization produced through exercised authority. The architectural objective of an AI system is therefore to ensure that realized consequences remain faithfully bound to the authority responsible for authorizing them. This property is expressed through Consequence Fidelity.

Consequence fidelity measures the degree to which realized outcomes preserve the governing authority under which those outcomes were originally authorized. Fidelity concerns structural preservation rather than informational similarity. Two execution paths may produce semantically similar outputs while differing materially in the authority relationships responsible for producing them. Likewise, two systems may reconstruct identical textual responses while differing

Published by Cyber Terrain LLC | cyberterrain.ai DOI: https://doi.org/10.5281/zenodo.21742328

substantially in the provenance, constraints, or delegated authority through which those responses were generated.

The Running Lifecycle Example: Phase 1 (Authorization)

An operations supervisor signs an explicit cryptographic policy lease authorizing an autonomous corporate agent to approve inventory procurement expenditures up to $50,000 strictly under *Procurement Policy Baseline v1.0*. This lease anchors the originating authority to a definitive boundary condition. Consequence fidelity mandates that if this specific procurement task executes over time or spans distributed endpoints, the resulting irreversible allocation of capital must remain structurally bound back to this supervisor's explicit *Policy Baseline v1.0* boundary.

Without defining what must remain preserved via consequence fidelity, no meaningful evaluation of continuity, reconstruction, or context can be conducted.

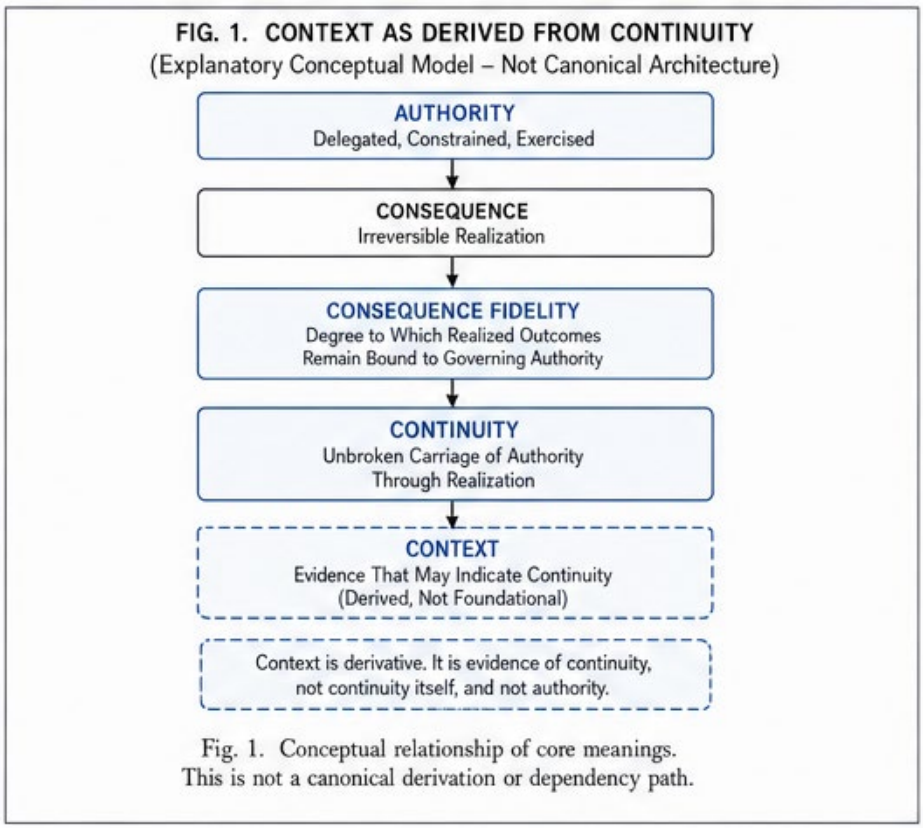
III. Continuity & Context

Continuity describes the structural preservation of authority-consequence relationships throughout realization. It is neither synonymous with memory nor reducible to contextual information retained by an AI system. Rather, continuity exists when consequence fidelity remains preserved despite temporal separation, distributed execution, or environmental change.

Modern AI frequently treats context as the principal carrier of continuity. Conversation history, embeddings, retrieval mechanisms, and persistent memory systems are all designed to increase the amount of information available during future execution. While these mechanisms improve informational continuity, they do not themselves establish structural continuity. Context records execution; continuity preserves authority.

Accordingly, this paper treats context as the externally observable, derived manifestation of preserved continuity rather than as an independent architectural object.

Published by Cyber Terrain LLC | cyberterrain.ai DOI: https://doi.org/10.5281/zenodo.21742328



Continuity remains the governing meaning-control substrate, while context provides an explanatory mechanism through which continuity may be inferred, examined, or challenged.

Boxed Proposition 1

Proposition: *Context constitutes evidence of continuity, not proof of continuity.*

The Running Lifecycle Example: Phase 2 (Long-Running Continuity)

After receiving the initial authorization, the procurement action enters a multi-stage, distributed fulfillment pipeline that operates over several weeks. Standard conversational history logs, database context records, and text summaries document that the request is currently active. Standard AI paradigms view this accumulated text context as the proof of continuity. In contrast, Authority Engineering observes that this context is merely an external record; structural continuity is maintained only if the supervisor's active delegation authority remains unbroken as the transaction shifts across the distributed execution environments.

IV. Foundational Dependency: Reconstruction Dependency Collapse

Distributed AI systems inevitably experience mutation. Models evolve, retrieval corpora change, prompts vary, external systems update, delegated agents modify execution paths, and

Published by Cyber Terrain LLC | cyberterrain.ai DOI: https://doi.org/10.5281/zenodo.21742328

environmental conditions continuously shift. As these mutations accumulate, direct preservation of continuity becomes increasingly difficult, and systems attempt to compensate through reconstruction.

Boxed Definition 1

Definition (Reconstruction): *Reconstruction is the post-hoc inferential process of estimating historical authority relationships from surviving observations after direct continuity carriage is unavailable.*

The transition from deterministic preservation to inferential approximation follows an inevitable progression driven by distributed mutation:

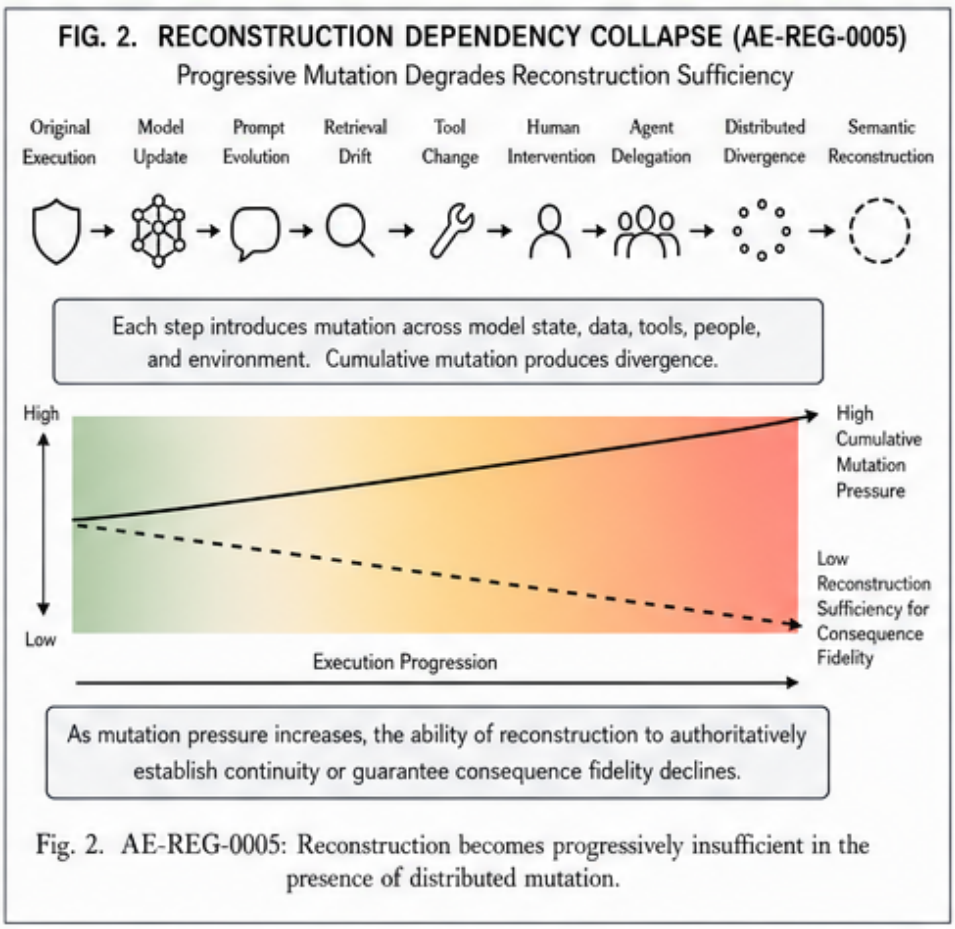


Fig. 2. AE-REG-0005: Reconstruction becomes progressively insufficient in the presence of distributed mutation.

As distributed mutation increases along the progression, reconstruction becomes progressively dependent upon incomplete observation, degraded provenance, semantic approximation, and inferred relationships. Each mutation step compounds uncertainty rather than eliminating it. The architecture transitions from *preserving* authority to *estimating* authority, meaning consequence fidelity can no longer be guaranteed solely through semantic recovery.

Core Epistemological Clarification

Published by Cyber Terrain LLC | cyberterrain.ai DOI: https://doi.org/10.5281/zenodo.21742328

Reconstruction Dependency Collapse does not state that continuity has ceased to exist. It states that continuity can no longer be authoritatively established through reconstruction with sufficient structural proof to guarantee consequence fidelity.

The collapse occurs strictly within the epistemological layer: it restricts what reconstruction can know, recover, or verify, and thereby constrains how continuity must be carried structurally rather than reconstructed after the fact.

The Running Lifecycle Example: Phase 3 (Accumulated Mutation)

Six months pass while the long-running procurement transaction undergoes sub-task delays. During this interval, the host language model receives a patch update, prompt structures are re-templated, the underlying organizational database drifts, and the original supervisor departs, changing corporate policy parameters.

Once continuity carriage depends on centralized semantic reconstruction, later reconstruction becomes insufficient to authoritatively establish continuity or guarantee consequence fidelity. The transaction has entered a reconstruction-dependent regime where it must attempt to estimate the original authorization environment from surviving text fragments and semantic context vectors.

V. Governed AI Root: Reconstruction Limit

From AE-REG-0005 Reconstruction Dependency Collapse follows the governing AI law for this family: AI-DL-004 Reconstruction Limit.

Formal Theorem Presentation

$$\text{Preserved Continuity} \ggg \text{Reconstructed Continuity}$$

$$\text{Reconstruction Confidence} \neq \text{Consequence Fidelity}$$

Where:

- $\ggg$ denotes **structural dominance** (the structural carriage of the state wholly constrains and supersedes any representation generated via post-hoc inference).
- $\neq$ denotes **non-substitutability** (the inferential confidence score of a reconstruction metric cannot function as, or substitute for, a deterministic proof of consequence fidelity).

AI-DL-004 Reconstruction Limit: Centralized semantic reconstruction alone cannot universally preserve consequence fidelity under distributed mutation. No increase in semantic reasoning, contextual completeness, or historical reconstruction can universally establish or verify authority once direct continuity carriage is no longer available.

Architectural Derivation

The structural derivation of this root follows a rigorous, deductive dependency sequence within the Authority Engineering registry:

Published by Cyber Terrain LLC | cyberterrain.ai DOI: https://doi.org/10.5281/zenodo.21742328

1. **Primitive Layer:** Authority generates an irreversible realization, known as consequence.
2. **Transition to Evaluation Metric (AE-REG-0042):** Because consequences are irreversible, any valid deployment of an AI system requires a metric to verify that realizations match their originating intent. This metric is Consequence Fidelity.
3. **Transition to Dependency Collapse (AE-REG-0005):** When execution traverses distributed environments, localized mutation injects non-uniform informational entropy. This turns authority verification into an estimation problem. Once continuity carriage depends on centralized semantic reconstruction, later reconstruction becomes insufficient to authoritatively establish continuity or guarantee consequence fidelity.
4. **Transition to Fidelity Bound (AE-REG-0019):** Because post-hoc reconstruction operates on semantic text similarity metrics rather than causal structural paths, observation and verification through centralized reconstruction become insufficient. It cannot differentiate between identical textual patterns generated under distinct authority boundaries.
5. **Deduction of Root Result (AI-DL-004):** Centralized algorithmic reconstruction reaches an absolute upper bound where semantic confidence and true consequence fidelity fully decouple. Consequence fidelity cannot be guaranteed through centralized semantic reconstruction alone.

Core Architectural Axiom

To clarify this core tension, the structural divide between preserving an operational state and estimating it after a disruption is formalized below:

Dimension	Preservation	Reconstruction
Operational Intent	Maintains active authority	Estimates historical authority
Temporal Placement	Occurs before discontinuity	Occurs after discontinuity
Fidelity Objective	Preserves consequence fidelity directly	Attempts to recover consequence fidelity post-hoc
Mathematical Nature	Structural / Deterministic	Inferential / Statistical

Published by Cyber Terrain LLC | cyberterrain.ai DOI: https://doi.org/10.5281/zenodo.21742328

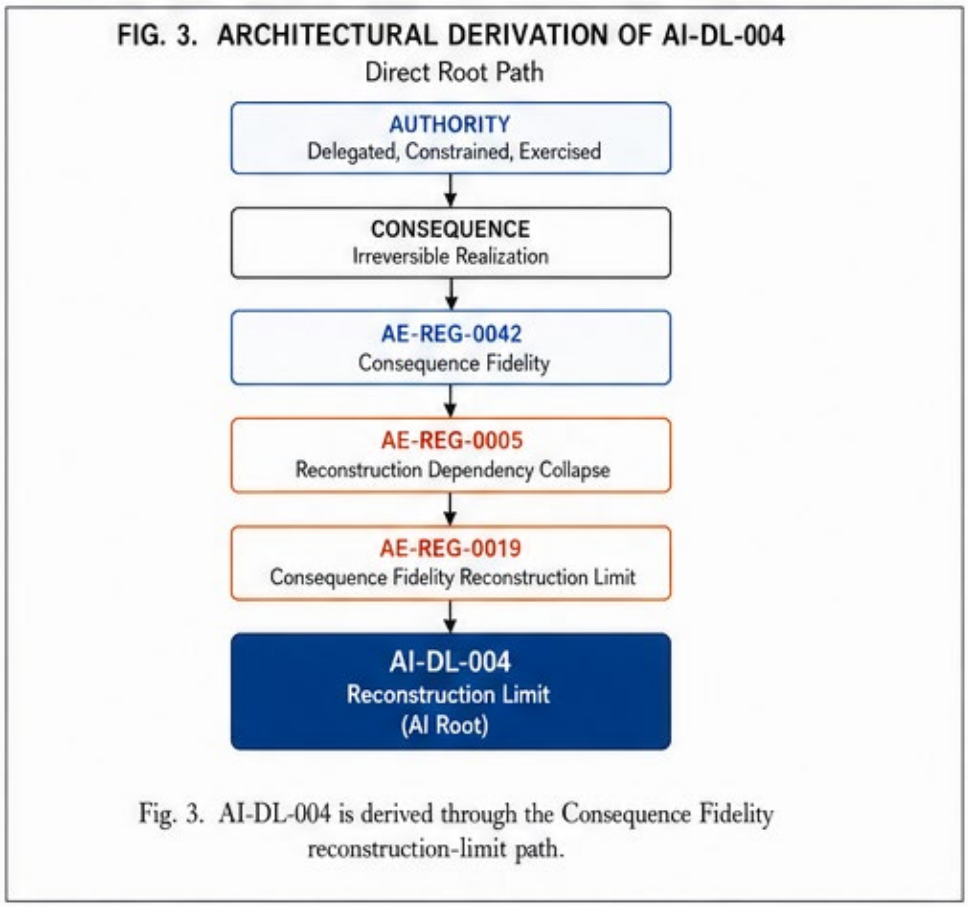


Fig. 3. AI-DL-004 is derived through the Consequence Fidelity reconstruction-limit path.

VI. Sibling & Derived Structural Consequences

Because reconstruction cannot universally preserve consequence fidelity, every downstream mechanism that relies upon semantic reconstruction inherits this limitation. The remaining results describe distinct manifestations of this common architectural bound.

Published by Cyber Terrain LLC | cyberterrain.ai DOI: https://doi.org/10.5281/zenodo.21742328

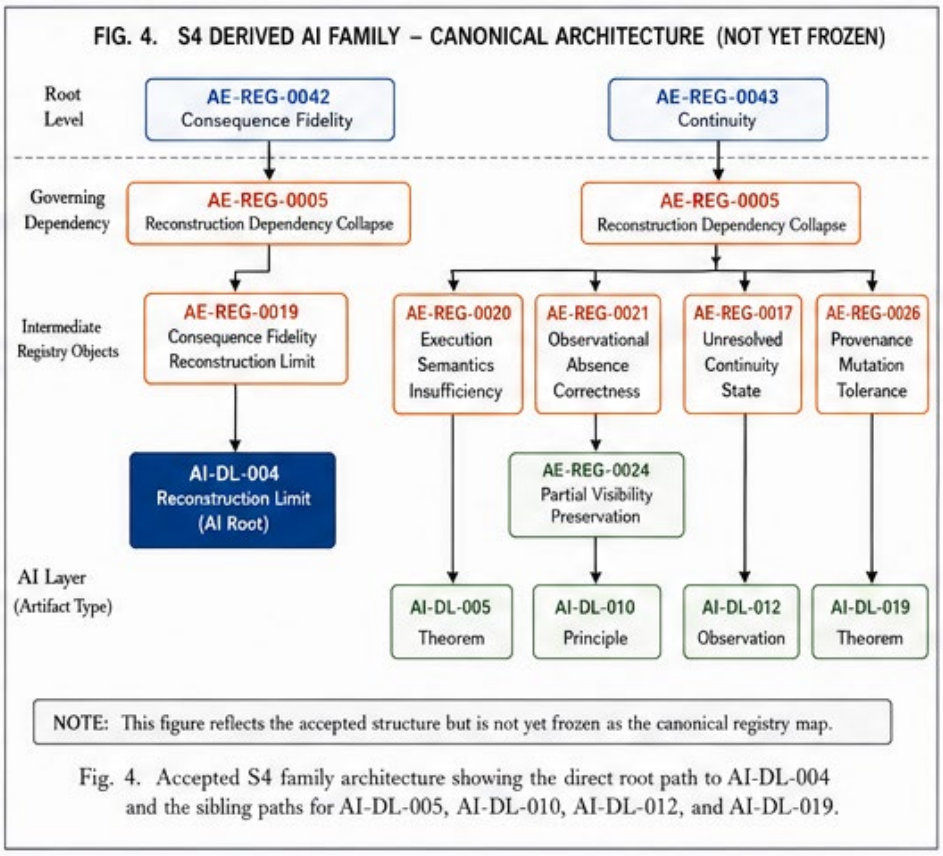


Fig. 4. Accepted S4 family architecture showing the direct root path to AI-DL-004 and the sibling paths for AI-DL-005, AI-DL-010, AI-DL-012, and AI-DL-019.

A. AI-DL-005 Semantics Insufficiency

Semantic agreement cannot independently establish preserved authority. Systems may generate equivalent language while differing materially in provenance, authority relationships, delegated constraints, or realization conditions.

[LLM Variant A] ⟶ (Mutated Environment) ⟶ [Identical Output Text] (Semantic Equality)

≠

[LLM Variant B] ⟶ (Unmutated Baseline) ⟶ [Identical Output Text] (Authority Inequality)

The Running Lifecycle Example: Phase 4-A (Semantics Insufficiency)

The mutated procurement agent completes its reconstruction attempt and outputs a final fulfillment approval: ORDER_BATCH_901_APPROVE. A separate, freshly instantiated agent running under an unmutated *Procurement Policy Baseline v1.0* environment is also given the same transaction variables and outputs the exact same text string: ORDER_BATCH_901_APPROVE.

Published by Cyber Terrain LLC | cyberterrain.ai DOI: https://doi.org/10.5281/zenodo.21742328

Semantically, the two outputs are perfectly equal. Structurally, however, the first output was generated through statistical inference across a fractured authority history, while the second was generated within an active, authorized boundary. Because downstream financial and accounting systems cannot validate the originating authority by inspecting the semantic text alone, stable execution semantics are insufficient. Semantic equality does not equal authority equality.

B. AI-DL-010 Partial Visibility Realization

Distributed execution rarely exposes complete observation of every authority relationship influencing realization. Reconstruction performed from incomplete visibility necessarily introduces uncertainty because absent authority cannot always be inferred from surviving observations.

The Running Lifecycle Example: Phase 4-B (Partial Visibility)

During the long-running transaction's lifecycle, the execution path crossed a critical human validation checkpoint:

Agent Prompt ⟶ Automated Analysis ⟶ Supervisor Override ⟶ Database Commit

When the system attempts to reconstruct the authorization context months later, it encounters a logging failure: due to log rotation policies, only the raw data snapshots and the final model execution tokens survive. The explicit supervisor override step is completely unrecorded.

The system faces *Partial Visibility Realization*. Structural continuity becomes operationally unverifiable for consequence fidelity. Because the historical signature of that authority boundary is completely missing, later reconstruction becomes insufficient to recover all authority-relevant continuity conditions.

C. AI-DL-012 Unresolved Continuity

Certain execution histories remain structurally indeterminate even after reconstruction has been attempted. Where continuity has degraded beyond recoverable limits, architectural uncertainty persists regardless of semantic confidence. Unresolved continuity therefore represents an expected structural condition rather than an implementation defect.

The Running Lifecycle Example: Phase 4-C (Unresolved Continuity)

Imagine two parallel procurement branches tracked in the system. In Branch X, the system encountered an internal data-access fault and routed the approval through an emergency validation policy. In Branch Y, the transaction proceeded smoothly under standard operating rules.

However, because the system's logging layer compresses historical context into simplified status entries, both branches record an identical text summary: Transaction resolved under standard fiscal clearance.. When a downstream agent reloads this history to perform a subsequent audit, the actual historical authority boundaries are impossible to untangle. The

Published by Cyber Terrain LLC | cyberterrain.ai DOI: https://doi.org/10.5281/zenodo.21742328

continuity state is permanently indeterminate, leaving the system in a state of *Unresolved Continuity* that no semantic engine can resolve.

D. AI-DL-019 Provenance Mutation Tolerance

As execution mutates through distributed environments, provenance itself evolves. Effective AI governance therefore cannot assume static provenance but must tolerate controlled mutation while preserving consequence fidelity. Provenance becomes a continuity-supporting structure rather than merely an audit artifact.

The Running Lifecycle Example: Phase 4-D (Provenance Mutation & Terminal Realization)

The procurement request moves through a multi-stage production pipeline:

User Intent ⟶ Planner Agent ⟶ Context Retriever ⟶ Final LLM Generator

To maintain capabilities in distributed networks, the architecture must exhibit *Provenance Mutation Tolerance*. The token must dynamically record structural transformations, ensuring that while the technical implementation details evolve, the underlying authority limits established in *Policy Baseline v1.0* remain verifiably preserved.

The Running Lifecycle Example: Terminal Realization

Under the pressure of these layered mutations and reconstruction limits, the system triggers the financial payout: The funds are wired**.** This action constitutes a terminal, irreversible *Consequence* under the Authority Engineering definition. Even though the post-hoc ability to authoritatively establish continuity has collapsed under AE-REG-0005, reality has altered permanently. The irreversible consequence remains in physical state, emphasizing that a failure of reconstruction does not merely invalidate a log—it strands an irreversible consequence away from its originating authority.

VII. Why AI Imposes Unique Reconstruction Pressure

The structural crisis distributed mutation poses is fundamentally more severe for artificial intelligence than for traditional software engineering. This disparity is formalized as a core foundation theorem:

Theorem 1 (Reconstruction Pressure Asymmetry)

Traditional deterministic systems permit replay because execution state remains bounded and invariant. Distributed AI architectures continually mutate execution conditions across non-repeatable, probabilistic boundaries. Therefore, in adaptive AI systems, reconstruction pressure ($\mathcal{P}_r$) scales exponentially relative to structural preservation capacity ($\mathcal{C}_p$):

$$\mathcal{P}_r \gg \mathcal{C}_p$$

Conventional software architectures rely on deterministic, bounded, repeatable, and explicitly mapped control flows. If a traditional distributed transaction breaks across a temporal boundary,

Published by Cyber Terrain LLC | cyberterrain.ai DOI: https://doi.org/10.5281/zenodo.21742328

the system can replay the execution trace using exact state logs; reconstruction is a trivial exercise in deterministic state recovery.

In contrast, AI agent architectures operate on probabilistic, adaptive, mutating, and non-repeatable execution boundaries. Because LLM weights, retrieval states, prompt combinations, and agent coordination pathways are non-deterministic, small environmental fluctuations yield divergent semantic paths. Consequently, reconstruction pressure grows exponentially faster in AI systems than in conventional computing. AI does not merely lose a state index; it obscures the underlying authority alignment, shifting the system entirely from state recovery to statistical inference.

VIII. Implications for Contemporary AI

The structural results established above provide a unified explanation for many contemporary AI techniques. Rather than standalone architectural solutions to continuity, modern mechanisms represent highly specialized engineering attempts to mitigate the pressures of AI-DL-004 Reconstruction Limit.

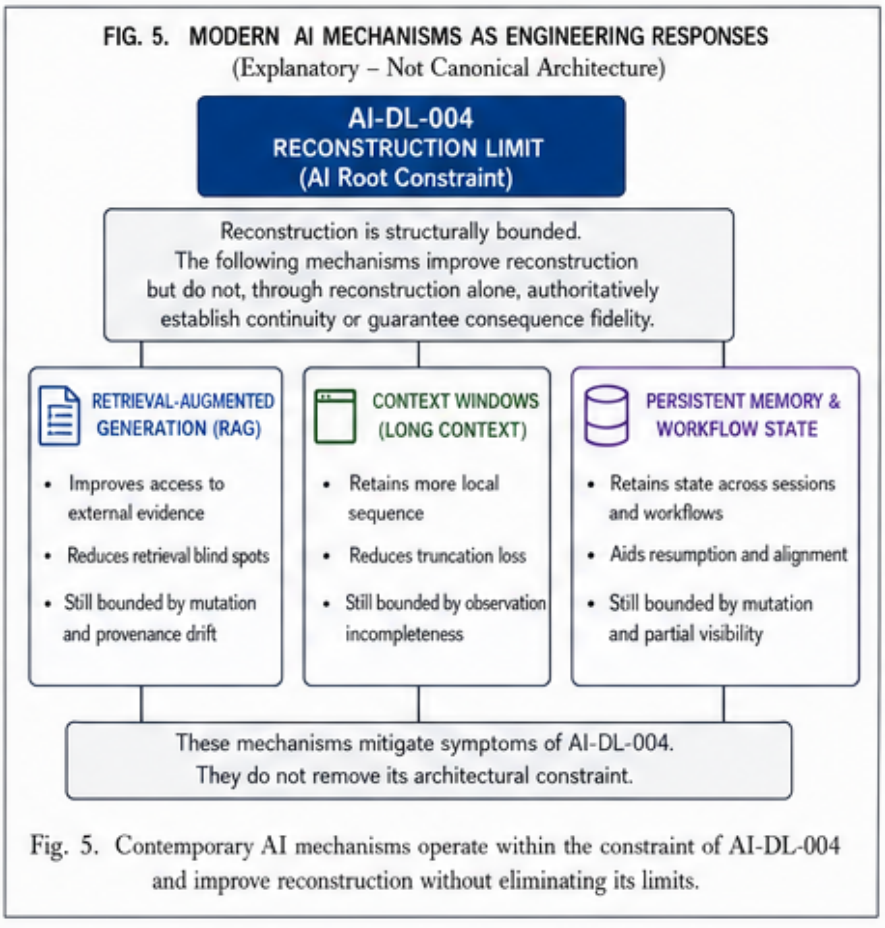


Fig. 5. Contemporary AI mechanisms operate within the constraint of AI-DL-004 and improve reconstruction without eliminating its limits.

Published by Cyber Terrain LLC | cyberterrain.ai DOI: https://doi.org/10.5281/zenodo.21742328

A widespread industry misconception is that increasing context windows or context scaling eliminates continuity loss. In reality, larger context windows simply increase the amount of observable evidence available during reconstruction; they do not structurally preserve the original authority relationships that governed prior realization.

Contemporary AI Mechanism	Naive Industry Interpretation	Authority Engineering Reduction
Context Window Expansion	Long-term architectural continuity.	Merely expands the capacity for bounded semantic evidence; does not bind authority.
Retrieval-Augmented Gen (RAG)	Dynamic system memory.	An un-sequenced reconstruction-support mechanism that provides historical data lacking causal provenance.
Persistent Memory (Embeddings)	Long-term model knowledge and identity.	Persistent semantic evidence; cannot preserve the temporal authorization states governing when embeddings were written.
Workflow State Engines	Complete execution continuity.	Records intermediate realization artifacts; data snapshots do not guarantee consequence fidelity if execution paths mutate.
Multi-Agent Coordination	Autonomous organizational scaling.	Disperses execution across multiple boundaries, compounding distributed reconstruction pressure and widening mutation risk.

This perspective reframes current AI engineering practice. Rather than asking how additional context may improve future reasoning, the governing architectural question becomes how consequence fidelity may be preserved before reconstruction becomes necessary. Preservation replaces recovery as the primary design objective.

IX. Cross-Paradigm Comparisons

Published by Cyber Terrain LLC | cyberterrain.ai DOI: https://doi.org/10.5281/zenodo.21742328

To clarify why Authority Engineering diverges from standard AI design assumptions, the following tables contrast how the two paradigms approach key operational concepts and structural assumptions.

Operational Concept	Contemporary AI Engineering Paradigm	Authority Engineering Framework
Core Memory Objective	Preserves context, historical text, and state vectors.	Preserves the structural authority governing consequence realization.
Role of Retrieval (RAG)	Reconstructs knowledge to ground model generations.	Functions as an approximation tool bounded by severe reconstruction limits.
Context Window Scaling	Extends complex reasoning and memory horizons.	Simply collects more observable evidence to present to the user.
Purpose of Provenance	Historical post-hoc auditing and logging.	Active, live substrate supporting ongoing structural continuity.
Agent Identity	Persistent conversation state or localized context logs.	A verifiably preserved, continuous authority relationship over time.
System Degradation	Treated as an engineering or implementation bug.	Recognized as a fundamental law (AI-DL-004) when relying on reconstruction.

Common Industry Assumption	Authority Engineering Reduction
More memory solves system continuity.	No, it merely increases semantic reconstruction evidence.
Better retrieval systems solve long-term continuity.	No, it improves reconstruction quality while remaining inferential.

Published by Cyber Terrain LLC | cyberterrain.ai DOI: https://doi.org/10.5281/zenodo.21742328

Common Industry Assumption	Authority Engineering Reduction
Provenance logs prove system authority.	No, provenance supports continuity but cannot retroactively replace it.
Semantic agreement across versions proves correctness.	No, semantic agreement cannot prove originating authority.

X. Architectural Boundaries

This paper intentionally limits its scope to the Continuity & Context branch of the Authority Engineering AI Foundations. It establishes the structural limitations governing reconstruction and derives the principal consequences affecting continuity preservation within distributed AI execution.

The paper does not examine the topology through which preserved authority propagates, the reachability of authority across distributed realization environments, or the delegation structures through which authority is transferred among autonomous participants. Those architectural questions are addressed by the independent Topology & Reachability and Authority & Delegation branches. Likewise, operational governance, tool boundaries, and execution control remain outside the scope of this work.

Published by Cyber Terrain LLC | cyberterrain.ai DOI: https://doi.org/10.5281/zenodo.21742328

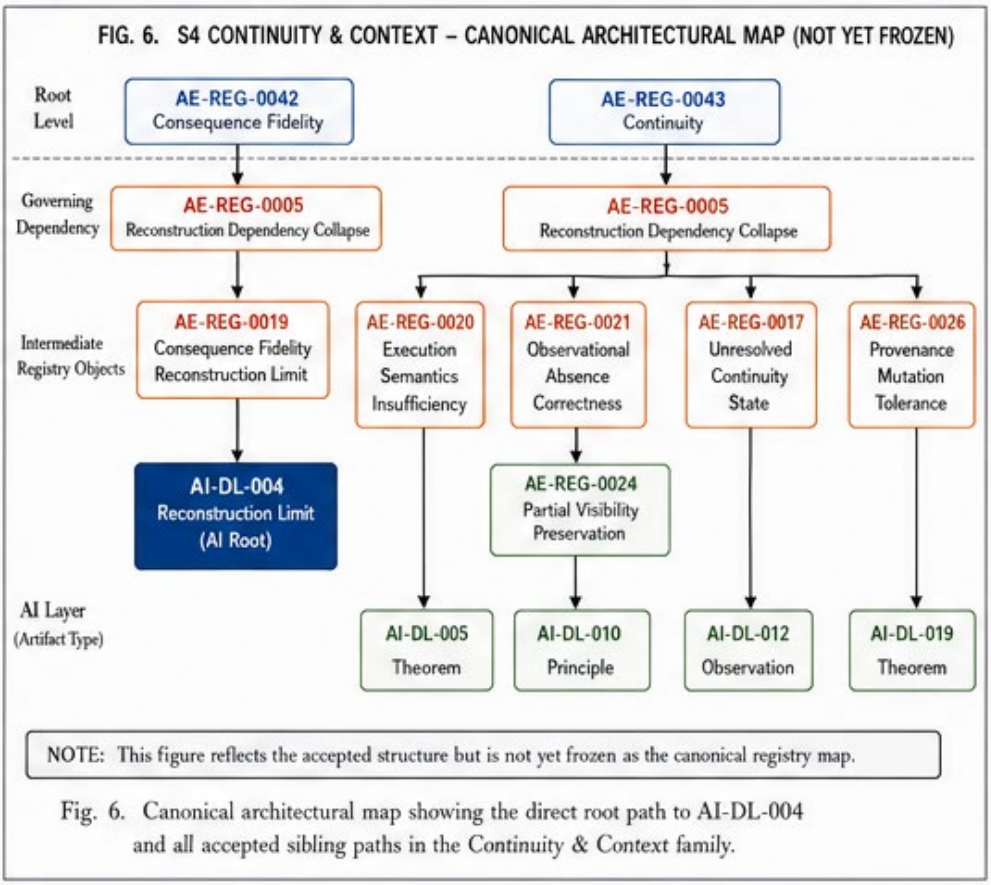


Fig. 6. Canonical architectural map showing the direct root path to AI-DL-004 and all accepted sibling paths in the Continuity & Context family.

XI. Conclusion

Continuity and context have often been treated as problems of memory, retrieval, and semantic reconstruction. Authority Engineering demonstrates that these mechanisms address only the recovery of information after structural continuity has already degraded. The governing architectural limitation lies deeper.

Consequence fidelity defines what must be preserved. Continuity defines the structural preservation of that fidelity throughout realization. Context provides observable evidence of preserved continuity but does not constitute continuity itself. Reconstruction attempts to recover authority after continuity has already been lost and is therefore governed by an inherent architectural limit under distributed mutation.

The Continuity & Context branch is consequently governed not by semantic recovery but by AI-DL-004 Reconstruction Limit. Artificial intelligence therefore does not fundamentally suffer from a memory problem. It suffers from a reconstruction problem. Memory, context, and retrieval merely determine how effectively reconstruction can be attempted; they cannot remove the structural limits that govern reconstruction itself.

Published by Cyber Terrain LLC | cyberterrain.ai DOI: https://doi.org/10.5281/zenodo.21742328

Intelligence discovers possible futures. Reconstruction estimates historical authority. Only preserved continuity maintains consequence fidelity across realization. Consequently, continuity cannot be engineered through increasingly sophisticated reconstruction; it must be engineered through the preservation of authority itself. This result establishes the architectural foundation upon which subsequent AI Foundations papers examine how preserved authority propagates through distributed topologies and how delegated authority remains governable across autonomous realization environments.

Rob Caswell | linkedin.com/in/rob-caswell3

Published by Cyber Terrain LLC | cyberterrain.ai DOI: https://doi.org/10.5281/zenodo.21742328