Authoritative History and the Limits of Autoregressive Intelligence

Flyxion

December 2025

Abstract

Recent critiques of large autoregressive language models converge on a common diagnosis: despite remarkable fluency, such systems lack a stable internal account of the world. This paper develops a unified formal framework explaining this limitation and proposing a constructive alternative. We argue that robust intelligence requires an authoritative internal history governed by invariant-preserving replay. Autoregressive systems operate exclusively at the level of derived views, lacking any mechanism to commit, verify, or replay causal structure. We formalize this distinction, prove that view-only generation suffers unavoidable long-horizon drift, and introduce deterministic event logs as a minimal substrate for grounded reasoning, planning, and safety. We further show how structural constraint systems in language and action instantiate the same invariant discipline, and how automatic, low-cost behavior emerges naturally through kernel optimization without abandoning authoritative control. The resulting architecture reconciles efficiency, reliability, and grounding without recourse to purely statistical imitation.

Contents

1 Introduction

The rapid success of large language models has revived foundational questions about the nature of intelligence. Systems trained to predict the next token in a sequence now generate coherent text, write functional code, and answer a wide range of questions. Yet these systems exhibit persistent failures in tasks requiring stable reasoning across time, physical interaction, or causal intervention. Such failures are often described informally as hallucinations, but their consistency across scale and domain suggests a deeper architectural limitation.

A growing body of critique argues that intelligence requires more than surface-level statistical fluency. In particular, it requires internal models that encode causal structure, support counterfactual reasoning, and enable planning under constraints. Language-only autoregressive models, by construction, lack these capacities. They do not maintain an internal state that is authoritative with respect to the world; they merely extend views conditioned on prior views.

This paper advances a precise formulation of that critique. We argue that the fundamental deficit of autoregressive systems is the absence of an authoritative internal history. Without such a history, a system cannot distinguish between speculative continuation and committed fact, between appearance and consequence. The result is inevitable drift under long-horizon generation and brittleness under intervention.

We develop this claim constructively. First, we formalize autoregressive generation as a view-only stochastic process and show that compounding divergence is unavoidable. Second, we introduce deterministic event logs as a minimal formal structure supporting authoritative replay and invariant preservation. Third, we show that world models, structural linguistic constraints, and event logs are formally equivalent as invariant-preserving transition systems. Finally, we demonstrate how efficiency and automaticity emerge naturally from repeated validation through kernel optimization, eliminating the false tradeoff between deliberation and speed.

The central thesis is simple but strong: intelligence requires machinery that can tell the difference between a view and a commitment.

1.1 Contributions

This paper advances a unified theoretical framework for understanding intelligence as the management of authoritative commitment rather than the accumulation of predictive fluency. It begins by providing a formal characterization of autoregressive generation as a view-only process of sequential conditioning, in which each continuation is produced solely on the basis of prior outputs without access to an invariant-governed state. This characterization makes explicit the structural limitations of autoregressive architectures that are often obscured by their empirical performance.

The paper then establishes that view-only systems exhibit unavoidable long-horizon drift under mild and general assumptions. By analyzing error propagation in the absence of invariant-gated transitions, it is shown that even small deviations from admissible structure compound irreversibly over extended sequences, yielding incoherence under perturbation or intervention. This result explains the characteristic failure modes of large predictive models without appealing to implementation details or training deficiencies.

Next, the paper introduces a deterministic event-log formalism in which authoritative history is treated as a first-class computational object. In this formalism, state is not updated implicitly but derived through invariant-preserving replay of committed events. This construction enforces coherence by design, ensuring that speculative views and hypothetical continuations cannot corrupt authoritative state.

Building on this foundation, the paper demonstrates that world models in control theory, structural constraints in linguistic cognition, and event-log architectures in computation are instances of a single abstract structure: invariant-preserving transition systems. By establishing this equivalence, the work unifies previously disparate approaches under a common formal core, showing that their differences lie in representation and domain specificity rather than in expressive power.

Finally, the paper provides a principled account of how fast, automatic behavior emerges from slow, authoritative reasoning through kernel-level optimization. Repeatedly validated event schemas are compiled into low-cost replay primitives, yielding efficient execution without sacrificing correctness. Automaticity is thus explained not as heuristic shortcut or approximation, but as optimized authority grounded in prior invariant-preserving validation.

1.2 Roadmap

Section 2 introduces formal preliminaries. Section 3 defines the world model requirement. Section 4 analyzes autoregressive generation and proves drift results. Section 5 introduces deterministic event logs and proves replay stability. Section 6 connects structural linguistic constraints to invariant-preserving transitions. Section 7 presents the unified equivalence theorem. Section 8 introduces kernel optimization and automaticity. Section 9 formalizes planning and safety. Section 10 discusses implications and limitations.

2 Formal Preliminaries

2.1 States, observations, and actions

We distinguish between internal states, external observations, and actions.

Definition 1 (State space). *Σ denotes the internal state space of the system. States summarize all committed information relevant to prediction, planning, and constraint evaluation.*

Definition 2 (Observation space). *$\mathcal{O}$ denotes the space of externally observable signals, including tokens, images, or sensor readings.*

Definition 3 (Action space). *$\mathcal{A}$ denotes the space of actions available to the system.*

2.2 Views and commitments

Definition 4 (View). *A view is any externally expressed or internally proposed description that has not been committed to the authoritative state. The space of views is denoted $\mathcal{V}$.*

Definition 5 (Commitment). *A commitment is an update to the authoritative internal state, recorded as part of a deterministic history and subject to invariant constraints.*

This distinction is central: views may be inconsistent, speculative, or transient; commitments must be lawful.

2.3 Invariants

Definition 6 (Invariant). *An invariant is a predicate defining a subset $\Omega \subseteq \Sigma$ of admissible states. A state σ is legal if $\sigma \in \Omega$.*

Invariants encode physical, logical, or structural constraints that must never be violated by committed state.

2.4 Transition systems

Definition 7 (Transition function). *A transition function is a (possibly partial) map*

$$\delta : \Sigma \times \mathcal{A} \to \Sigma$$

defined only when the resulting state lies in Ω.

Remark 1. Partiality is essential: inadmissible transitions are rejected rather than penalized.

3 World Models as Invariant-Preserving Predictors

The claim that intelligence requires a *world model* is often stated informally. In this section we give a precise formulation that isolates the functional requirements without committing to any particular implementation substrate. The core idea is that a world model is not a generator of surface detail, but a constrained predictor of lawful state transitions.

3.1 Historical motivation

Early theories of cognition emphasized internal models as tools for prediction and control rather than for faithful reproduction of sensory input. CraikâĂŹs original formulation characterized thought as the manipulation of internal models that anticipate the consequences of action. Classical AI planning systems instantiated this idea explicitly, encoding state variables and transition operators subject to hard constraints. More recent approaches reintroduce world models through learned latent dynamics, motivated by sample efficiency and transfer.

Across these traditions, the unifying thread is not simulation fidelity but *counterfactual sensitivity*: the ability to answer questions of the form, âĂIJWhat would happen if I did this instead?âĂİ

3.2 Formal definition

Definition 8 (World model)**.** *A world model is a tuple* (g, f) *where*

$$g : \mathcal{O}^* \to \Sigma \quad \textit{and} \quad f : \Sigma \times \mathcal{A} \to \Sigma$$

such that:

1. *g maps observation histories to internal states;*
2. *f predicts the next state conditional on an action;*
3. *f is defined only when the resulting state lies in Ω.*

The defining feature is not probabilistic prediction per se, but admissibility: the model must preserve invariants under action-conditioned transitions.

3.3 Completeness, soundness, and efficiency

Definition 9 (Soundness)**.** *A world model is* sound *if, for all admissible states $\sigma \in \Omega$ and actions $a \in \mathcal{A}$, the predicted transition $f(\sigma, a)$ respects the same invariants as the true environment.*

Definition 10 (Completeness)**.** *A world model is* complete *relative to a task domain if every admissible action sequence in the environment corresponds to a sequence of transitions in the model.*

Definition 11 (Efficiency)**.** *A world model is* efficient *if it supports planning and adaptation with asymptotically fewer observations than direct imitation, by exploiting structural regularities encoded in* Ω*.*

These criteria clarify why generating realistic sensory detail is neither necessary nor sufficient. A system may produce visually plausible futures while violating causal or structural constraints, rendering it useless for planning.

3.4 Planning as constrained rollout

Given a world model and a cost function $\mathcal{C} : \Sigma \to \mathbb{R}$, planning consists of evaluating hypothetical action sequences without committing them.

Definition 12 (Plan evaluation)**.** *For a horizon* H *and action sequence* $(a_1, \ldots, a_H)$*, define the predicted rollout*

$$\sigma_{t+1} = f(\sigma_t, a_t),$$

with cumulative cost

$$J = \sum_{t=1}^{H} \mathcal{C}(\sigma_t).$$

Crucially, this evaluation occurs in the space of hypothetical states. Only selected actions are later committed to the authoritative history.

Remark 2. This separation between hypothetical rollout and commitment is precisely what autoregressive systems lack.

4 Autoregressive Generation as View-Only Dynamics

We now contrast world models with autoregressive systems. Although such systems are often described as predictive, their prediction targets are views rather than states.

4.1 Formal characterization

An autoregressive model defines a family of conditional distributions

$$P(x_t \mid x_{<t})$$

over observations. Generation proceeds by sampling or decoding from these conditionals.

Let $S_t = x_{<t}$ denote the prefix at time t. Then autoregressive generation defines a Markov process on prefixes:

$$\mathbb{P}(S_{t+1} = S_t \cdot x) = D(P(x \mid S_t)),$$

where D is a decoding operator.

4.2 Absence of authoritative state

Although prefixes may encode information implicitly, they do not constitute an authoritative state in the sense defined earlier. There is no distinguished internal object against which views are checked, nor any mechanism that enforces invariant preservation across steps.

Remark 3. Any notion of âĂIJstateâĂİ in an autoregressive system is implicit, distributed, and non-binding. It cannot reject an illegal continuation; it can only assign it lower probability.

4.3 Exposure bias and distributional mismatch

During training, conditionals are optimized under a data distribution over prefixes. During generation, prefixes are drawn from the modelâĂŹs own induced distribution. This mismatch accumulates.

Assumption 1 (Local mismatch bound)**.** *There exists $\varepsilon > 0$ such that for all prefixes in a relevant support,*

$$\mathrm{TV}(P_{data}(\cdot \mid s), P_{model}(\cdot \mid s)) \leq \varepsilon.$$

4.4 Horizon divergence

Theorem 1 (View-only drift)**.** *Under the local mismatch bound, the total variation distance between the distribution of prefixes under the model and under the data distribution grows at least linearly with horizon:*

$$\mathrm{TV}(P_{model}(S_t), P_{data}(S_t)) \leq t\varepsilon,$$

up to saturation.

Proof. The result follows by induction using a telescoping argument on the Markov chain induced by autoregressive generation. Each step introduces at most ε additional divergence; absent a contraction mechanism, these errors accumulate. □

4.5 Projection to state space

Suppose one attempts to recover an internal state by projecting prefixes into a latent space.

Assumption 2 (Lipschitz projection)**.** *There exists a projection $\pi : \mathcal{O}^* \to \Sigma$ and constant $L > 0$ such that*

$$\mathbb{E}[d(\pi(S), \pi(S'))] \leq L\,\mathrm{TV}(P(S), P(S')).$$

Proposition 1 (Projected drift)**.** *Under the above assumptions, expected deviation between projected states grows at least linearly with horizon.*

Proof. Immediate from the horizon divergence theorem and the Lipschitz condition. □

This result formalizes the intuition that even small local errors become catastrophic when propagated through view-only generation.

4.6 Why heuristics do not solve the problem

Techniques such as multi-sample selection, chain-of-thought prompting, or tree-based decoding can reduce local error by searching over views. However, they do not introduce an authoritative state or invariant gating. They select among speculative continuations but never bind the system to a committed internal history.

Remark 4. Improved search over views is still search over views. Without commitment, there is no notion of correctness beyond local plausibility.

5 Deterministic Event Logs and Authoritative State

We now introduce the central constructive alternative to view-only autoregression: a deterministic event-log architecture that enforces authoritative history through invariant-preserving replay. This formalism separates speculative views from committed facts and provides the minimal substrate required for grounding, planning, and verification.

5.1 Motivation: why history, not snapshots

State-based descriptions are ubiquitous in formal models of cognition and computation. However, when states are updated implicitly or probabilistically, the provenance of information becomes opaque. By contrast, an explicit history records not only *what* the current state is, but *why* it is that way.

Event logs treat state as a derived quantity rather than a primitive. The authoritative object is the sequence of committed events; state is obtained by replay.

5.2 Event space and replay semantics

Definition 13 (Event space). *Let $\mathcal{E}$ be a set of atomic events. Events represent minimal, semantically meaningful state transitions admissible under system invariants.*

Definition 14 (Event log). *An event log is a finite sequence*

$$\mathcal{L} = (e_1, e_2, \dots, e_T) \in \mathcal{E}^*.$$

Definition 15 (Replay function). *Fix an initial state $\sigma_0 \in \Omega$. Let*

$$\delta : \Sigma \times \mathcal{E} \to \Sigma$$

be a deterministic transition function. The replay function is defined recursively by

$$\mathrm{Replay}(\emptyset) = \sigma_0, \qquad \mathrm{Replay}(\mathcal{L} \cdot e) = \delta(\mathrm{Replay}(\mathcal{L}), e).$$

Remark 5. Replay is deterministic by construction. Any nondeterminism must occur at the proposal stage, prior to commitment.

5.3 Commitment and invariant gating

The defining feature of authoritative history is that not all proposed events are accepted.

Definition 16 (Invariant-gated commit)**.** *Let* $\Omega \subseteq \Sigma$ *be the admissible set. The commit operator is a partial function*

$$\mathrm{Commit} : \mathcal{E}^* \times \mathcal{E} \rightharpoonup \mathcal{E}^*$$

defined by

$$\mathrm{Commit}(\mathcal{L}, e) = \begin{cases} \mathcal{L} \cdot e & \textit{if } \delta(\mathrm{Replay}(\mathcal{L}), e) \in \Omega, \\ \textit{undefined} & \textit{otherwise.} \end{cases}$$

Proposals that would violate invariants are rejected outright rather than penalized probabilistically.

5.4 Views as non-authoritative projections

Definition 17 (View projection)**.** *A view is any function*

$$q : \Sigma \to \mathcal{V}.$$

Given an event log $\mathcal{L}$*, the authoritative view is* $q(\mathrm{Replay}(\mathcal{L}))$*. Other views may be generated independently and need not correspond to any committed state.*

Remark 6. Multiple views may coexist without conflict. Only the log determines what is authoritative.

5.5 Replay-stabilized consistency

We now state and prove the central stability property of the architecture.

Theorem 2 (Replay-stabilized consistency)**.** *Assume* $\mathrm{Replay}(\emptyset) = \sigma_0 \in \Omega$*. Under invariant-gated commit, every reachable authoritative state is admissible:*

$$\mathcal{L} \textit{ reachable by successive commits} \implies \mathrm{Replay}(\mathcal{L}) \in \Omega.$$

Moreover, no sequence of non-authoritative views can corrupt the authoritative state.

Proof. By induction on the length of the log. The base case holds by assumption. For the inductive step, suppose $\mathrm{Replay}(\mathcal{L}) \in \Omega$. A new event e is committed only if $\delta(\mathrm{Replay}(\mathcal{L}), e) \in \Omega$. Thus $\mathrm{Replay}(\mathcal{L} \cdot e) \in \Omega$. Views do not alter $\mathcal{L}$ and therefore cannot affect $\mathrm{Replay}(\mathcal{L})$. □

Corollary 1 (Quarantine of incoherence)**.** *Any incoherence arising in speculative views remains non-authoritative unless it passes the invariant gate. Incoherent views may exist, but they cannot propagate into committed state.*

5.6 Branching, counterfactuals, and planning

Event logs naturally support counterfactual reasoning.

Definition 18 (Hypothetical extension)**.** *Given a log $\mathcal{L}$ and a candidate event sequence $(e'_1, \dots, e'_k)$, the hypothetical state is*

$$\sigma' = \mathrm{Replay}(\mathcal{L} \cdot e'_1 \cdots e'_k),$$

provided all intermediate transitions are admissible.

Such hypothetical extensions are evaluated during planning but not committed unless selected.

Remark 7. This formalizes counterfactual reasoning as replay over uncommitted branches. The distinction between imagination and action is enforced structurally rather than heuristically.

5.7 Relation to causality

The meaning of an event is defined entirely by its effect under replay. This aligns naturally with interventionist accounts of causation: to intervene is to append an event and observe its consequences via replay.

Definition 19 (Intervention)**.** *An intervention corresponds to proposing an event e whose acceptance is evaluated by the commit gate. The causal effect of the intervention is the difference between* $\mathrm{Replay}(\mathcal{L})$ *and* $\mathrm{Replay}(\mathcal{L} \cdot e)$*.*

5.8 Comparison with alternative formalisms

State machines encode transitions directly but do not preserve history as a first-class object. Petri nets emphasize concurrency but lack a single authoritative linearization without additional structure. Temporal logics describe admissible sequences but do not provide a constructive replay mechanism.

Event logs differ in making history primitive and state derivative. This inversion is essential for auditability, counterfactual reasoning, and invariant enforcement.

Naive replay incurs a computational cost that is linear in the length of the event log, since the authoritative state must be reconstructed by sequentially applying each committed event. While this property is sufficient for theoretical analysis, it is impractical for real-time systems operating over long histories.

In practice, this cost is mitigated through incremental replay, in which the current authoritative state is maintained alongside the log and updated eagerly upon each successful commit. Full replay is then required only for validation, auditing, or recovery, rather than for routine operation.

Additional efficiency is obtained through snapshotting, whereby periodic checkpoints of the authoritative state are recorded and used as replay anchors. When reconstruction is required, replay proceeds from the most recent snapshot rather than from the beginning of the log, reducing reconstruction time while preserving determinism and auditability.

Finally, indexed projections, or materialized views, are employed to support efficient access to commonly queried aspects of state. These projections are derived functions of the authoritative state and are updated deterministically in response to new commits. Because they do not participate in commitment, they cannot corrupt authoritative history.

Together, these techniques preserve the formal guarantees of invariant-preserving replay while enabling real-time operation at scale. They demonstrate that treating history as authoritative is not merely a conceptual idealization, but a viable computational strategy.

6 Structural Constraints and Domain-Specific Invariants

The event-log formalism introduced in the previous section is intentionally domain-agnostic. It specifies how authoritative history is maintained, but not what kinds of events or invariants are required. In this section we show how domain-specific structureâĂŤparticularly in language and actionâĂŤinstantiates the same invariant-preserving transition discipline.

The key claim is that what are often described as âĂIJrules,âĂİ âĂIJschemas,âĂİ or âĂIJformsâĂİ are best understood as admissibility constraints on event composition.

6.1 From surface regularities to structural admissibility

Surface regularities, whether linguistic or behavioral, are insufficient to guarantee coherence under perturbation. What stabilizes cognition is not the ability to predict common continuations, but the ability to reject illegal ones.

Structural constraints define which event compositions are admissible, independently of their frequency.

Definition 20 (Structural invariant)**.** *A structural invariant is a predicate $\Omega_D \subseteq \Sigma_D$ over a domain-specific state space Σ_D (e.g. syntactic configurations, motor plans) such that only states in Ω_D may be committed.*

6.2 Language as constrained event composition

Consider a domain Σ_{lang} whose elements encode partial linguistic structures. Events correspond to minimal structure-building operations.

Definition 21 (Linguistic event)**.** *A linguistic event $e \in \mathcal{E}_{lang}$ is a minimal admissible transformation of a partial structure, such as feature unification or hierarchical composition.*

Definition 22 (Structural composition)**.** *Let $\delta_{lang} : \Sigma_{lang} \times \mathcal{E}_{lang} \to \Sigma_{lang}$ be a deterministic transition function defined only when the resulting structure satisfies domain invariants (e.g. hierarchy, locality).*

These invariants are not statistical preferences. They are hard constraints: violations are undefined, not improbable.

Remark 8. This distinction explains why grammatically invalid constructions cannot be repaired by probability mass alone. They are rejected because they fail admissibility, not because they are rare.

6.3 Hierarchy as a stability constraint

Hierarchical organization is often treated as a representational choice. In the present framework it plays a deeper role: hierarchy enforces bounded locality of dependency and enables compositional replay.

Lemma 1 (Hierarchical admissibility)**.** *If a structural invariant enforces hierarchical embedding, then admissible event sequences admit replay with bounded dependency depth.*

Sketch. Hierarchical constraints ensure that dependencies are nested rather than crossed. Replay of nested dependencies can be performed using a stack discipline, bounding the required contextual access. □

6.4 Structural collapse and canonical form

Many domain-specific operations admit multiple equivalent representations. Structural collapse maps such equivalence classes to canonical forms.

Definition 23 (Canonical collapse)**.** *Let $\sim$ be an equivalence relation on Σ_D preserving semantic content. A collapse operator*

$$\kappa : \Sigma_D \to \Sigma_D / \sim$$

maps states to canonical representatives.

Collapse reduces redundancy without sacrificing invariants. It is a form of normalization.

6.5 Action and motor control

The same formalism applies to motor planning. Let Σ_{motor} encode partial motor plans, and let events correspond to elementary motor primitives.

Definition 24 (Motor invariant)**.** *A motor invariant encodes biomechanical, energetic, or safety constraints such that only physically realizable plans are admissible.*

Motor expertise consists in discovering sequences of events that reliably satisfy these invariants across contexts.

6.6 Why domain-specific invariants must be hard

Soft constraints can be violated with some probability. Hard invariants cannot.

Proposition 2 (Necessity of hard constraints)**.** *If domain invariants are enforced probabilistically rather than by admissibility, then long-horizon plans cannot be guaranteed invariant-preserving under replay.*

Proof. Probabilistic enforcement allows a non-zero chance of violation at each step. Over unbounded horizons, the probability of at least one violation approaches one. □

6.7 Relation to learned representations

Nothing in this framework forbids learning invariants. What is required is that, once learned, they function as hard gates on commitment.

Remark 9. Learning determines what the invariants are; architecture determines how they are enforced.

7 Toward a Unified Formalism

At this point, three parallel constructions have been developed within the framework. The first consists of action-conditioned world models, which preserve physical and causal invariants by restricting state transitions to those consistent with environmental dynamics. The second consists of structural constraint systems, which enforce domain-specific admissibility conditions, such as hierarchical organization and compositional locality in language or feasibility constraints in action. The third consists of deterministic event-log architectures, which preserve authoritative history by deriving state exclusively through invariant-preserving replay of committed events.

Although these constructions originate in different literatures and are often treated as fundamentally distinct, they share a common underlying structure. In the following section, it is shown that each can be formalized as an instance of a single abstract class of systems, differing only in representational choice and domain-specific invariant specification rather than in computational principle.

8 Invariant-Preserving Transition Systems

We now abstract from domain-specific detail and identify the common formal structure underlying world models, structural constraint systems, and deterministic event logs. The goal is not to introduce new machinery, but to recognize an invariant pattern already present in each case.

8.1 Abstract definition

Definition 25 (Invariant-preserving transition system)**.** *An invariant-preserving transition system (IPTS) is a tuple*

$$\mathcal{T} = (X, U, \tau, \Omega),$$

where X denotes a state space and U denotes an input or action space. The transition function

$$\tau : X \times U \rightharpoonup X$$

is a partial function, meaning that it is defined only on a subset of state–action pairs. The set $\Omega \subseteq X$ specifies the admissible states of the system. A transition $\tau(x, u)$ is defined if and only if the resulting state lies in Ω.

The defining feature of this formalism is partiality rather than probabilistic weighting. Transitions that would violate invariants are not assigned low likelihood or penalized by an objective function;

they are undefined and therefore unexecutable. Admissibility is enforced structurally at the level of state evolution rather than statistically at the level of prediction.

8.2 Instantiation by world models

A world model specified by a state transition function f and an observation or representation map g induces an invariant-preserving transition system in a direct manner. In this instantiation, the state space X corresponds to the set of internal world states Σ, and the action space U corresponds to the set of agent actions $\mathcal{A}$. The transition function τ is given by the world model dynamics, so that $\tau(\sigma, a) = f(\sigma, a)$ whenever the resulting state is admissible. The admissible set Ω coincides with the set of physically or causally valid world states, denoted Ω.

Under this identification, soundness of the world model is equivalent to the correctness of τ with respect to environmental dynamics, while invariant preservation ensures that planning and prediction remain confined to physically realizable trajectories.

Soundness of the world model corresponds to correctness of τ relative to environmental dynamics.

8.3 Instantiation by structural constraint systems

A structural constraint system induces an invariant-preserving transition system by treating domain-specific configurations as states and structure-building operations as transitions. In this instantiation, the state space X is identified with a domain-dependent configuration space Σ_D, where D may correspond to language, motor control, or any other structured cognitive domain. The input space U is identified with the set of domain events $\mathcal{E}_D$, each event representing a minimal admissible transformation of a partial structure.

The transition function τ is defined by the rules of admissible composition within the domain. A transition $\tau(x, e)$ is defined only when the application of event e to state x yields a configuration that satisfies the structural constraints of the domain. The admissible set Ω therefore coincides with the set Ω_D of structurally well-formed states, encoding invariants such as hierarchy, locality, compositional closure, or physical feasibility. Under this formulation, structural well-formedness is enforced directly through partiality of the transition function rather than indirectly through probabilistic preference.

8.4 Instantiation by event logs

An event-log architecture induces an invariant-preserving transition system by taking authoritative system states as elements of the state space and committed events as inputs. In this case, the state space X corresponds to the set of system states Σ, while the input space U corresponds to the set of atomic events $\mathcal{E}$ recorded in the log. The transition function τ is given by the deterministic event application function δ, so that $\tau(\sigma, e) = \delta(\sigma, e)$ whenever the resulting state is admissible.

The admissible set Ω is identified with the invariant-satisfying states Ω, and the partiality of τ enforces invariant-gated commitment. Replay is then defined as the iterative application of τ starting from a fixed initial state, yielding a deterministic reconstruction of the authoritative state from the committed event history. This construction ensures that all reachable states arise from sequences of admissible transitions and that speculative or non-authoritative views cannot alter committed state.

8.5 Equivalence theorem

We now state the central equivalence.

Theorem 3 (Equivalence of invariant-preserving formalisms)**.** *World models, structural constraint systems, and deterministic event logs are equivalent up to representation when each is formalized as an invariant-preserving transition system.*

The equivalence asserted by the theorem consists of three claims. First, each of the formalisms developed in the preceding sections induces an invariant-preserving transition system. This follows directly from their construction: world models, structural constraint systems, and event-log architectures each define a state space, an input space, a partial transition function, and a set of admissible states such that transitions are defined only when invariants are preserved.

Second, for any invariant-preserving transition system (X, U, τ, Ω)*, there exists a log-based realization whose replay reproduces the same transition behavior. Given such a system, one may define atomic events as elements of the input space* U *and construct an event log as a finite sequence of these events. Replay is defined as the successive application of the transition function* τ *starting from a fixed initial state. Because* τ *is partial, invariant gating is enforced automatically: any event sequence that would lead outside* Ω *is undefined and therefore cannot be committed.*

Third, the apparent differences between world models, structural constraint systems, and event-log architectures correspond to choices of state representation and invariant specification rather than to differences in expressive power. Any behavior admissible under one formalism can be embedded into the others through the corresponding identification of states, inputs, and invariants. Consequently, the systems are equivalent up to encoding, and their distinctions are representational rather than structural.

Proof. The first claim follows immediately from the constructions given in Sections 4 and 5, where each formalism was shown to instantiate the defining components of an invariant-preserving transition system.

To establish the second claim, consider an arbitrary invariant-preserving transition system (X, U, τ, Ω). Define a log-based system in which events correspond to elements of U and replay corresponds to iterated application of τ from a designated initial state. Because τ is a partial function whose codomain is restricted to Ω, any attempt to replay an inadmissible transition is undefined, thereby enforcing invariant gating at the level of commitment.

The third claim follows from the observation that admissibility is preserved under representational embedding. Given an admissible transition sequence in one formalism, the corresponding sequence in another formalismâĂŤconstructed via the appropriate identification of states and transitionsâĂŤyields the same admissible behavior under replay. Thus, differences among the formalisms do not reflect differences in computational principle, but only in representational choice. □

Corollary 2 (Necessity of authoritative history)**.** *Any system lacking an IPTS realizationâĂŤspecifically, any system without invariant-gated transitionsâĂŤcannot guarantee long-horizon coherence* *under intervention.*

8.6 Why autoregression does not fit

Autoregressive models fail to instantiate an invariant-preserving transition system for structural reasons rather than for reasons of scale or training regime. In an autoregressive architecture, there is no distinguished internal state space X that evolves independently of the surface-level output sequence. Instead, the systemâĂŹs effective state is implicitly identified with the sequence of previously generated tokens, which conflates representation, history, and proposal into a single object.

Moreover, autoregressive transitions do not exhibit partiality in the sense required by invariant-preserving systems. Given any prior context, the conditional distribution defining the next-token prediction assigns nonzero probability to all elements of the vocabulary, including those that would correspond to invariant violations in a structured domain. As a result, transitions that would be inadmissible under structural, causal, or logical constraints are not rendered undefined; they are merely assigned lower probability.

Because of this, autoregressive systems lack a mechanism for rejecting illegal continuations outright. They can only downweight such continuations statistically, which is insufficient to guarantee invariant preservation under long-horizon generation or intervention. Over extended sequences, even small probabilities of violation compound, leading to the drift and incoherence analyzed earlier.

For these reasons, autoregression is strictly weaker than invariant-preserving transition systems. Its expressive power is limited to approximating admissible behavior in distribution, rather than enforcing admissibility by construction. No increase in scale or training data alters this structural limitation.

8.7 Optional categorical perspective

For readers inclined toward abstraction, an IPTS may be viewed as a category whose objects are admissible states and whose morphisms are admissible transitions. Event logs correspond to composable morphism sequences; replay is functorial composition. Invariants define a subcategory closed under composition. This perspective adds no expressive power but clarifies compositionality.

Remark 10. Nothing in the argument depends on category theory. It merely provides a compact language for the equivalence.

9 Kernel Optimization and Compiled Replay

A common objection to invariant-preserving architectures is that they appear computationally expensive. If every proposed transition must be validated against global constraints and replayed through authoritative history, how can real-time behavior be achieved? This objection assumes that validation cost is fixed. In fact, invariant-preserving systems naturally admit a form of kernel optimization whereby repeated lawful transitions are compiled into low-cost replay primitives.

9.1 Repeated validation and cost reduction

Let $\mathcal{S}$ be a finite event schema,

$$\mathcal{S} = (e_1, e_2, \dots, e_k),$$

such that committing $\mathcal{S}$ from any state in a designated admissible context class $\Omega_{\mathcal{S}} \subseteq \Omega$ always results in an admissible state.

Initially, execution of $\mathcal{S}$ requires full arbiter mediation: each event proposal e_i must be validated sequentially. Let $C_n(\mathcal{S})$ denote the cost of executing $\mathcal{S}$ after n successful commits.

Definition 26 (Validation cost)**.** *The validation cost $C(\mathcal{S})$ is the computational cost of confirming that* $\mathrm{Replay}(\mathcal{L} \cdot \mathcal{S}) \in \Omega$*, including all invariant checks required during replay.*

9.2 Cost collapse through repetition

Repeated successful validation allows the kernel to replace expensive checks with cached guarantees.

Assumption 3 (Contextual exhaustiveness)**.** *There exists a finite or effectively enumerable set of contexts $\Omega_{\mathcal{S}}$ such that $\mathcal{S}$ has been validated in all relevant admissible contexts.*

Proposition 3 (Cost monotonicity)**.** *If $\mathcal{S}$ is repeatedly committed without invariant violation across $\Omega_{\mathcal{S}}$, then*

$$C_{n+1}(\mathcal{S}) \le C_n(\mathcal{S}), \qquad \lim_{n\to\infty} C_n(\mathcal{S}) = C_{\mathrm{replay}},$$

where C_{replay} is the cost of deterministic replay without validation.

Sketch. Invariant checks are redundant once correctness has been established for all admissible contexts. Removing redundant checks preserves correctness while reducing cost. □

9.3 Cached schemas

Definition 27 (Cached schema)**.** *A cached schema is an event schema $\mathcal{S}$ whose execution has been promoted from arbiter-mediated validation to direct replay, under a specified context class $\Omega_{\mathcal{S}}$.*

Cached schemas function as low-level primitives. They are fast precisely because their legality has already been proven.

Remark 11. Cached schemas are not unconstrained shortcuts. They are admissible only within their validated context class. Outside this class, control reverts to full validation.

9.4 Automaticity as compiled authority

Fast, apparently effortless behavior arises when cached schemas dominate execution. Such behavior is often described as intuitive or automatic, but in this framework it is simply compiled authority.

Proposition 4 (Authority preservation under caching)**.** *Cached schema execution preserves all invariants enforced by arbiter-mediated execution, provided that execution occurs within the validated context class.*

Proof. By definition, cached schemas are compiled from invariant-preserving executions. Their replay reproduces the same transitions without re-evaluating constraints. □

9.5 Failure and re-escalation

Cached execution is not irrevocable. When conditions fall outside the validated context class, cached replay may be unsafe.

Definition 28 (Escalation). *Escalation is the process by which execution of a cached schema is suspended and control is returned to arbiter-mediated validation upon detection of context mismatch.*

Remark 12. This mechanism explains how rapid behavior can fail gracefully: rather than compounding error, the system slows down and reasserts authority.

9.6 Relation to learning

Learning in this framework is understood as a structured transformation of how event schemas are handled by the system rather than as a relaxation of constraint enforcement. Initially, novel event schemas are introduced as speculative proposals that must be evaluated under full arbiter control. Each proposed schema is executed only through invariant-gated validation, ensuring that its effects on authoritative history are admissible across the relevant state contexts.

As execution proceeds, the system accumulates evidence about the reliability of a given schema. When repeated validation demonstrates that the schema preserves all required invariants across its intended domain of application, the system may promote it to a cached schema. This promotion reflects not a change in correctness criteria, but a change in how correctness is enforced: costly validation checks are replaced by previously established guarantees.

Learning, therefore, does not weaken or bypass structural constraints. On the contrary, it strengthens efficiency by compiling invariant-preserving behavior into low-cost replay primitives. The system becomes faster not because it abandons authority, but because it has earned the right to exercise it more efficiently.

9.7 Why view-only systems cannot replicate this

Autoregressive systems exhibit fast generation because they never validate. However, without an arbiter and authoritative history, they cannot distinguish cached legality from coincidental regularity. As a result, speed is achieved at the cost of correctness, and no principled mechanism exists for re-escalation when invariants are violated.

Corollary 3 (False automaticity). *Any system that exhibits fast execution without a mechanism for authoritative validation risks unbounded error accumulation under perturbation.*

10 Planning as Search over Authoritative History

We now show how goal-directed behavior arises naturally once authoritative history, invariant gating, and cached replay are in place. Planning is not an auxiliary module layered atop the system; it is the disciplined exploration of hypothetical event logs under admissibility constraints.

10.1 Plans as hypothetical extensions

Let $\mathcal{L}$ be the current authoritative log. A plan is a finite sequence of candidate events evaluated without commitment.

Definition 29 (Plan)**.** *A plan is a finite sequence*

$$\pi = (e_1, e_2, \dots, e_k)$$

such that $\mathrm{Replay}(\mathcal{L} \cdot \pi)$ *is defined, i.e. all intermediate transitions are admissible.*

Plans exist entirely in hypothetical space until committed.

10.2 Objectives as evaluative functionals

An objective assigns value to reachable states.

Definition 30 (Objective functional)**.** *An objective is a function*

$$J : \Sigma \to \mathbb{R}$$

evaluated on replayed hypothetical states.

Objectives do not determine admissibility. They rank admissible outcomes.

10.3 Planning as constrained optimization

Planning is the process of selecting a plan that optimizes the objective while respecting invariants.

Definition 31 (Constrained planning problem)**.** *Given current log $\mathcal{L}$, objective J, and planning horizon H, find*

$$\pi^* = \arg \min_{\pi \in \Pi_H} J(\mathrm{Replay}(\mathcal{L} \cdot \pi)),$$

where Π_H is the set of admissible plans of length $\leq H$.

Remark 13. Illegitimate plans are excluded by construction, not penalized.

10.4 Safety as admissibility

Safety constraints are simply invariants expressed at the system level.

Definition 32 (Safety invariant)**.** *A safety invariant is a predicate $\Omega_{safe} \subseteq \Sigma$ such that any state outside Ω_{safe} is uncommittable.*

Proposition 5 (Safety by construction)**.** *If all committed states satisfy Ω_{safe}, then no committed action sequence can violate safety.*

Proof. Immediate from invariant-gated commit: unsafe transitions are undefined. □

10.5 Hard constraints vs. soft penalties

Penalty-based systems encode safety as a cost to be traded off. Invariant-based systems encode safety as impossibility.

Proposition 6 (Non-compensability of invariants)**.** *No finite objective penalty is equivalent to an invariant constraint under unbounded planning horizons.*

Proof. Any finite penalty can be outweighed by sufficient reward. An invariant cannot. □

10.6 Escalation during planning

Cached schemas accelerate planning but may fail under novel contexts.

Definition 33 (Planning escalation)**.** *Planning escalation occurs when a cached schema cannot be safely applied during hypothetical replay, triggering arbiter-mediated validation.*

This ensures correctness under novelty without sacrificing speed in familiar domains.

10.7 Failure modes

Although invariant-preserving systems eliminate entire classes of failure associated with unconstrained prediction, they are not immune to error. One source of failure arises from incomplete or incorrectly specified invariants. If the admissibility conditions enforced by the arbiter fail to capture all relevant structural, causal, or safety constraints, the system may admit transitions that are formally legal but substantively undesirable.

A second source of failure concerns the specification of objective functions. While objectives do not determine admissibility, they guide selection among admissible futures. Poorly chosen or misaligned objectives can therefore lead the system to pursue outcomes that satisfy invariants yet fail to meet external performance or alignment criteria.

Failure may also result from bounded search horizons. Planning over authoritative history is necessarily finite, and insufficient lookahead can cause the system to select locally admissible plans that lead to undesirable long-term consequences. This limitation reflects computational trade-offs rather than architectural deficiency.

Finally, errors may occur when context classes for cached schemas are misidentified. If a schema is promoted to low-cost replay under conditions broader than those for which its invariant preservation has been established, automatic execution may occur in contexts where escalation would have been required.

Crucially, these failure modes are diagnosable by design. They manifest as explicit violations, refusals, or escalations within the systemâĂŹs decision process, rather than as silent drift or undetected incoherence. This transparency distinguishes invariant-preserving systems from view-only architectures, whose failures often remain latent until they accumulate catastrophically.

10.8 Comparison with contemporary safety approaches

Unlike reinforcement learning with human feedback, which relies on external correction, invariant-preserving systems enforce safety internally. Unlike red-teaming, which samples failures, admissibility prevents them.

Remark 14. Interpretability is intrinsic: every committed action is justified by replay.

11 Why Scaling Alone Is Insufficient

The preceding analysis explains why increasing model capacity does not resolve the deficiencies of view-only architectures. Without authoritative history and invariant gating, increased scale merely accelerates drift.

Planning, safety, and efficiency are not emergent properties of prediction. They are consequences of commitment.

12 Empirical Signatures and Evaluation Criteria

A theory of intelligence that cannot be empirically distinguished from its competitors is of limited scientific value. The framework developed in this paper makes strong architectural commitments, and therefore yields concrete, testable predictions. In this section we identify empirical signatures that differentiate invariant-preserving systems from view-only predictive models.

12.1 What must be evaluated

Standard benchmarks in artificial intelligence typically emphasize short-horizon accuracy or next-step prediction performance. While such metrics are appropriate for assessing surface-level fluency or local predictive competence, they are insufficient for detecting the presence of authoritative history or invariant-gated state evolution. Systems that differ fundamentally in their capacity for commitment may nevertheless achieve comparable scores on these benchmarks.

Evaluation must therefore target properties that are sensitive to architectural differences rather than to statistical approximation alone. In particular, the relevant measures concern long-horizon coherence under intervention, where the system must maintain invariant-preserving behavior across extended sequences of actions. Robustness to distributional perturbation is likewise essential, as it probes the systemâĂŹs ability to adapt to structurally meaningful changes rather than merely interpolate within familiar data regimes.

Equally important is sample efficiency in novel environments, which reflects the systemâĂŹs ability to infer and enforce new invariants from limited experience. Finally, evaluation must examine the systemâĂŹs failure modes under constraint violation, distinguishing between architectures that fail silently through cumulative drift and those whose failures manifest as explicit rejection, escalation, or diagnostic signals. Only benchmarks that capture these dimensions can meaningfully assess the presence of authoritative state and commitment.

12.2 Long-horizon coherence

Definition 34 (Coherence horizon)**.** *The coherence horizon H_c of a system is the maximum horizon length such that replayed behavior remains invariant-preserving under arbitrary admissible perturbations.*

Invariant-preserving systems exhibit coherence horizons that scale with planning depth rather than training data size. View-only systems exhibit rapidly decaying coherence horizons regardless of scale.

Empirical measurement may involve iterated reasoning tasks, multi-step physical manipulation, or extended dialogue with constraint maintenance.

12.3 Perturbation tests

Perturbation tests introduce minor but structurally relevant changes to the environment.

A navigation task in which a familiar obstacle is displaced by a small amount. Systems relying on surface regularities often fail catastrophically, while invariant-preserving systems adapt via replay and replanning.

Failure in view-only systems will manifest as confident but incoherent continuations. Failure in invariant-preserving systems will manifest as escalation and slowed decision-making.

12.4 Sample efficiency and novelty

Invariant-preserving systems learn by validating transitions rather than memorizing trajectories.

Given a novel environment with consistent dynamics, invariant-preserving systems require orders of magnitude fewer samples to achieve stable performance than predictive systems trained end-to-end.

This prediction aligns with biological learning and sharply contradicts data-scaling assumptions common in large models.

12.5 Failure topology

Not all failures are equal.

Definition 35 (Failure topology)**.** *The failure topology of a system characterizes how errors propagate under iteration: whether they remain localized, escalate gracefully, or compound irreversibly.*

Invariant-preserving systems exhibit bounded failure propagation: errors trigger rejection or escalation. View-only systems exhibit unbounded error accumulation.

12.6 Benchmarks that matter

Not all benchmarks are equally informative with respect to the architectural claims advanced in this paper. Diagnostic benchmarks are those that require the system to respect hard constraints over extended interaction rather than merely producing locally plausible outputs. Tasks involving tool use with strict preconditions are particularly revealing, as successful performance depends on the systemâĂŹs ability to reject actions that are syntactically or causally ill-formed rather than merely unlikely.

Physical reasoning tasks with delayed consequences provide a second class of meaningful evaluation. Such tasks test whether a system can maintain invariant-preserving behavior across long horizons, where early decisions constrain later possibilities in ways that cannot be repaired retroactively. Similarly, language tasks that require syntactic repair under perturbation probe the presence of structural admissibility constraints, distinguishing systems that enforce grammatical well-formedness from those that merely approximate it statistically.

Finally, environments that require reversible reasoningâĂŤwhere actions must be undone or counterfactual branches must be exploredâĂŤdirectly test the systemâĂŹs capacity for authoritative replay. Benchmarks that emphasize fluency, static correctness, or short-horizon prediction lack this diagnostic power and are therefore insufficient for evaluating the claims made here.

12.7 What would falsify this framework

The framework developed in this paper makes strong architectural claims and is therefore falsifiable. It would be undermined by empirical evidence demonstrating that a view-only predictive system can sustain unbounded coherence under arbitrary perturbations, maintaining invariant-preserving behavior without explicit mechanisms for authoritative state or commitment.

Similarly, the framework would be challenged if an autoregressive model were shown to reliably reject illegal continuations in structured domains without incorporating explicit constraint machinery or invariant-gated transitions. Such a result would indicate that admissibility enforcement can emerge robustly from probabilistic conditioning alone.

Finally, the theory would be falsified if planning and safety were observed to emerge consistently and robustly in systems lacking any notion of authoritative history, replay, or commitment. To date, no empirical evidence supports these possibilities, and existing results are consistent with the necessity of invariant-preserving architectures for long-horizon coherence and safety.

12.8 Relation to existing empirical work

Existing results on hallucination, compounding error, and brittleness in large models are predicted by this framework rather than explained away as engineering limitations.

Remark 15. Negative results are as informative as positive ones. The absence of certain failure modes is a stronger signal than benchmark success.

13 Related Work

The framework developed in this paper intersects with several traditions across artificial intelligence, cognitive science, linguistics, and philosophy. This section surveys relevant work, emphasizing points of genuine contact as well as principled divergence.

13.1 World models in artificial intelligence

The idea that intelligent systems require internal models of the world predates modern machine learning. Early cybernetic and control-theoretic approaches emphasized prediction and feedback as prerequisites for adaptive behavior. Craik famously proposed that organisms carry small-scale models of reality that allow them to anticipate outcomes.

Classical AI instantiated world models through symbolic planning systems such as STRIPS and situation calculus. These systems possessed explicit state representations and transition operators but lacked robustness and scalability. Later hybrid approaches combined model-free reinforcement learning with learned dynamics models.

More recent work on learned world models emphasizes latent-space prediction, particularly in continuous domains. While these approaches move away from symbolic manipulation, they often retain probabilistic transition semantics and lack hard admissibility constraints. As a result, they remain vulnerable to compounding error under long-horizon planning.

The present framework differs by treating invariant preservation as primary. Prediction is subordinated to commitment, and uncertainty is handled at the proposal stage rather than during state evolution.

13.2 Critiques of statistical language models

Skepticism toward purely statistical language modeling has a long history. Early arguments emphasized the distinction between grammatical competence and surface distributional patterns. Contemporary critiques focus on hallucination, brittleness, and lack of grounding.

Recent defenses of large language models argue that scale induces implicit world models. While large models exhibit impressive in-context generalization, their failure modes under perturbation suggest the absence of invariant-gated state transitions. The framework developed here explains both the strengths and weaknesses of such systems without denying their empirical success.

13.3 Structured approaches to language

Formal linguistics has long emphasized hierarchical structure, compositionality, and locality constraints. Constraint-based grammars and minimalist frameworks enforce well-formedness through admissibility rather than likelihood.

The present work aligns with this tradition in treating structure as a hard constraint, but diverges in its computational realization. Rather than positing static grammars, structure is enforced dynamically through invariant-gated event composition and replay.

13.4 Cognitive architectures

Cognitive architectures such as SOAR and ACT-R incorporate explicit representations of goals, memory, and procedural knowledge. These systems resemble event-log architectures in their separation of declarative memory and procedural execution.

However, classical architectures rely on hand-engineered rules and lack mechanisms for invariant learning and promotion. The present framework generalizes these ideas by treating logs and invariants as first-class computational objects subject to optimization.

13.5 Neuroscientific perspectives

Neuroscientific evidence supports the existence of replay mechanisms in biological cognition. Hippocampal replay during sleep and rest is implicated in consolidation and planning. Predictive coding frameworks emphasize the minimization of prediction error but often conflate perception and state update.

The distinction between authoritative history and derived views clarifies these findings. Replay corresponds to authoritative reconstruction, while perception corresponds to proposal generation subject to validation.

13.6 Philosophical foundations

Philosophically, the framework resonates with interventionist accounts of causation, where meaning is defined by the effects of actions rather than by static descriptions. It also aligns with views in epistemology that distinguish belief from knowledge by commitment and justification.

In philosophy of mind, the approach is compatible with functionalist accounts that emphasize role over substrate. What matters is not representation per se, but the existence of a mechanism capable of enforcing coherence under counterfactuals.

13.7 Event-based systems outside AI

Event sourcing and commandâĂŞquery responsibility segregation in software engineering demonstrate the practical advantages of treating history as authoritative. Distributed systems rely on logs to ensure consistency under partial failure.

These systems illustrate that replay-based architectures are not exotic but proven at scale. The present work extends these ideas into cognition and intelligence.

13.8 Summary

Across disciplines, the same pattern recurs: systems that privilege authoritative history and invariant enforcement outperform those that rely on surface prediction alone. The contribution of this paper is to make this pattern explicit, formal, and testable.

14 Conclusion

This paper began with a simple observation: systems that predict continuations without committing to consequences cannot sustain coherence under intervention. From that observation we developed a unified framework in which intelligence is understood not as statistical fluency, but as the disciplined management of commitment.

14.1 Summary of contributions

The central contributions of this work are as follows.

First, we distinguished *authoritative history* from *derived views*. Authoritative history is a deterministic, invariant-preserving record of committed events. Views are projections that may be speculative, approximate, or optimized for efficiency. Confusing the two leads to unbounded error accumulation.

Second, we formalized invariant-preserving transition systems and showed that world models, structural constraint systems, and deterministic event logs are instances of the same abstract structure. This equivalence clarifies long-standing debates by showing that disagreements often concern representation rather than capability.

Third, we demonstrated that autoregressive architectures fail to instantiate invariant-preserving transition systems. Their inability to reject illegal continuations, enforce causal constraints, or maintain authoritative state explains their characteristic failure modes under long-horizon reasoning and perturbation.

Fourth, we introduced kernel optimization as a principled mechanism by which invariant-preserving systems achieve efficiency. Repeated lawful transitions are compiled into cached schemas, yielding fast behavior without sacrificing correctness. Automaticity emerges as optimized authority, not as heuristic shortcut.

Fifth, we showed that planning and safety are naturally expressed as constrained search over hypothetical extensions of authoritative history. Objectives rank admissible futures; invariants determine which futures are possible at all. Safety is enforced by construction rather than penalty.

Finally, we articulated empirical predictions and falsifiability criteria that distinguish this framework from purely predictive approaches. These predictions concern coherence horizons, failure topology, sample efficiency, and robustness under intervention.

14.2 Addressing the strongest objections

A common objection is that large predictive models may implicitly learn world models. The framework developed here does not deny that predictive systems can approximate invariants in distribution. What it denies is that approximation suffices for commitment. Without explicit invariant gating and authoritative history, such approximations cannot guarantee correctness under novel conditions.

Another objection holds that tools or external modules can compensate for missing internal structure. However, unless those tools are integrated into an authoritative commit pathway, they merely relocate the problem. The question is not whether constraints exist somewhere, but whether they are enforced at the point of commitment.

A further objection appeals to scaling laws, suggesting that sufficient scale will induce planning and grounding. Scale increases capacity, not authority. Without architectural support for invariant preservation, increased scale amplifies both fluency and failure.

14.3 Implications for artificial intelligence

The implications for AI research are direct. Systems aspiring to general competence must treat commitment as a first-class operation. Prediction must be subordinated to admissibility. Efficiency must arise through compilation of verified behavior, not through abandonment of validation.

This suggests a shift in emphasis away from end-to-end predictive optimization and toward architectures that explicitly separate proposal, validation, commitment, and replay. Such architectures need not abandon learning; they require that learning produce constraints, not merely correlations.

14.4 Implications for cognition

The framework also illuminates cognitive phenomena. Fast, intuitive behavior arises from cached replay of previously validated schemas. Deliberative reasoning corresponds to arbiter-mediated exploration of hypothetical histories. Learning consists in promoting reliable patterns from costly validation to efficient execution.

Errors, hesitation, and slowdown under novelty are not failures of intelligence but signatures of its operation.

14.5 Open questions and future work

Several questions remain open.

How are invariants learned efficiently in high-dimensional domains? What mechanisms best detect context boundaries for cached schemas? How should invariant hierarchies be organized to support compositionality across domains?

Empirically, building minimal invariant-preserving systems in constrained environments would provide decisive tests of the framework. Theoretical work on approximate replay and bounded validation may yield practical optimizations without undermining guarantees.

14.6 Final perspective

Intelligence is often described as the ability to predict. This paper argues that prediction is secondary. What matters is the ability to commitâĂŤto bind oneself to consequences and to maintain coherence in the face of counterfactuals.

Systems that merely speak about the world may appear intelligent. Systems that act within it must answer to history.

A Formal Proofs

This appendix provides detailed proofs of selected propositions and theorems stated in the main text. All notation and terminology follow Sections 2 through 7. The purpose of these proofs is not merely to establish correctness, but to make explicit the assumptions under which the results hold and to clarify the scope of their applicability.

A.1 Proof of Proposition 1 (Autoregressive Drift)

We restate the proposition for convenience.

Proposition 7 (Autoregressive Drift)**.** *In the absence of explicit invariant enforcement, autoregressive generation produces cumulative divergence from any invariant-preserving manifold with probability one under mild regularity assumptions.*

Setup and assumptions

Let $\mathcal{X}$ denote the space of possible system outputs, and let $\mathcal{M} \subset \mathcal{X}$ be a subset corresponding to invariant-preserving states. The set $\mathcal{M}$ may represent grammatical well-formedness, physical feasibility, logical consistency, or any other structural constraint.

Consider an autoregressive process generating a sequence $\{x_t\}_{t\geq 1}$ according to conditional distributions

$$p(x_t \mid x_{<t}),$$

where $x_{<t} = (x_1, \ldots, x_{t-1})$.

We impose the following minimal assumptions:

(A1) The model assigns nonzero probability mass to states outside $\mathcal{M}$ whenever $x_{<t} \in \mathcal{M}$.

(A2) The probability of leaving $\mathcal{M}$ at each step is bounded below by a positive constant on a set of nonzero measure.

Assumption (A1) reflects the absence of hard constraint enforcement; assumption (A2) excludes degenerate cases in which violations are asymptotically suppressed to zero.

Lemma: persistence probability

Lemma 2. *Let*

$$\epsilon_t = \mathbb{P}[x_t \notin \mathcal{M} \mid x_{<t} \in \mathcal{M}].$$

If there exists $\epsilon > 0$ such that $\epsilon_t \geq \epsilon$ for infinitely many t, then

$$\mathbb{P}[x_t \in \mathcal{M} \textit{ for all } t \leq T] \leq (1-\epsilon)^T.$$

Proof. The probability that the sequence remains within $\mathcal{M}$ up to time T is given by

$$\prod_{t=1}^{T} (1-\epsilon_t).$$

If $\epsilon_t \geq \epsilon$ for infinitely many t, then this product is bounded above by $(1-\epsilon)^T$, which converges to zero as $T \to \infty$. □

Proof of Proposition

By the lemma, the probability that an autoregressive sequence remains within $\mathcal{M}$ over arbitrarily long horizons converges to zero. Since autoregressive architectures lack a mechanism for rendering invariant-violating transitions undefined, they cannot enforce $\epsilon_t = 0$ uniformly. Consequently, divergence from $\mathcal{M}$ is not merely possible but inevitable in the limit.

This establishes that autoregression cannot guarantee invariant preservation over unbounded horizons, regardless of model capacity or training data size. □

A.2 Proof of Replay-Stabilized Consistency

We restate the theorem.

Theorem 4 (Replay-Stabilized Consistency)**.** *If an event-log system employs invariant-gated commitment, then every reachable authoritative state satisfies all system invariants.*

Definitions

Let $\mathcal{L} = (e_1, \ldots, e_n)$ denote a finite event log, let σ_0 be the initial state, and let

$$\text{Replay}(\mathcal{L}) = \delta(\cdots \delta(\delta(\sigma_0, e_1), e_2), \ldots, e_n)$$

be the deterministic replay function. Let $\Omega \subseteq \Sigma$ denote the set of admissible states.

Proof

We proceed by induction on the length of the event log.

Base case. For the empty log $\emptyset$, $\text{Replay}(\emptyset) = \sigma_0 \in \Omega$ by assumption.

Inductive hypothesis. Assume that for some $n \geq 0$, $\text{Replay}(e_1, \ldots, e_n) \in \Omega$.

Inductive step. Consider a proposed event e_{n+1}. By definition of invariant-gated commit, e_{n+1} is appended to the log only if

$$\delta(\text{Replay}(e_1, \ldots, e_n), e_{n+1}) \in \Omega.$$

If this condition is satisfied, then

$$\text{Replay}(e_1, \ldots, e_{n+1}) \in \Omega.$$

Otherwise, the event is rejected and the log remains unchanged.

Thus, by induction, all reachable authoritative states are admissible. □

A.3 Proof of Non-Compensability of Invariants

We restate the proposition.

Proposition 8 (Non-Compensability of Invariants)**.** *No finite penalty-based objective can enforce an invariant over unbounded planning horizons.*

Setup

Let $J : \Sigma \to \mathbb{R}$ be a reward function and let $P : \Sigma \to \mathbb{R}$ be a penalty function satisfying $|P(\sigma)| \leq M$ for some finite $M > 0$. Suppose that invariant violations incur a penalty $-P(\sigma)$ but remain admissible.

Proof

Because the penalty magnitude is finite, there exists a planning horizon H and a sequence of actions leading to a state σ' such that the cumulative reward accumulated prior to reaching σ' exceeds M. In this case, the total expected return of the plan remains positive despite the invariant violation.

Formally, for sufficiently large H,

$$\sum_{t=1}^{H} J(\sigma_t) - P(\sigma') > 0.$$

Thus, σ' becomes optimal under the objective despite violating the intended constraint.

By contrast, invariant-gated systems render σ' unreachable, independent of reward structure or horizon length. Therefore, finite penalties cannot substitute for hard invariants. □

A.4 Discussion

The results in this appendix formalize three complementary claims. First, probabilistic suppression of illegal states is insufficient to guarantee long-horizon coherence. Second, invariant-gated replay provides absolute admissibility guarantees by construction. Third, safety and structural constraints cannot be reduced to optimization objectives without loss of enforceability.

Together, these results support the central thesis of the paper: commitment, not prediction, is the fundamental unit of intelligence.

A Formal Proofs

This appendix provides detailed proofs of selected propositions and theorems stated in the main text. All notation and terminology follow Sections 2 through 7. The purpose of these proofs is not merely to establish correctness, but to make explicit the assumptions under which the results hold and to clarify the scope of their applicability.

A.1 Proof of Proposition 1 (Autoregressive Drift)

We restate the proposition for convenience.

Proposition 9 (Autoregressive Drift)**.** *In the absence of explicit invariant enforcement, autoregressive generation produces cumulative divergence from any invariant-preserving manifold with probability one under mild regularity assumptions.*

Setup and assumptions

Let $\mathcal{X}$ denote the space of possible system outputs, and let $\mathcal{M} \subset \mathcal{X}$ be a subset corresponding to invariant-preserving states. The set $\mathcal{M}$ may represent grammatical well-formedness, physical feasibility, logical consistency, or any other structural constraint.

Consider an autoregressive process generating a sequence $\{x_t\}_{t\geq 1}$ according to conditional distributions of the form

$$p(x_t \mid x_{<t}),$$

where $x_{<t} = (x_1, \ldots, x_{t-1})$.

We impose two minimal regularity assumptions. First, whenever the prefix $x_{<t}$ lies entirely within $\mathcal{M}$, the conditional distribution assigns nonzero probability mass to outputs outside $\mathcal{M}$. This assumption formalizes the absence of hard constraint enforcement: invariant-violating continuations are not rendered impossible by the model. Second, the probability of leaving $\mathcal{M}$ is bounded below by a strictly positive constant on a set of prefixes of nonzero measure. This assumption excludes degenerate cases in which violations are asymptotically suppressed to zero through explicit architectural mechanisms.

Lemma: persistence probability

Lemma 3. *Let*

$$\epsilon_t = \mathbb{P}[x_t \notin \mathcal{M} \mid x_{<t} \in \mathcal{M}].$$

If there exists $\epsilon > 0$ such that $\epsilon_t \geq \epsilon$ for infinitely many values of t, then

$$\mathbb{P}\big[x_t \in \mathcal{M} \textit{ for all } t \leq T\big] \leq (1-\epsilon)^T.$$

Proof. The probability that the sequence remains within $\mathcal{M}$ up to time T is given by the product

$$\prod_{t=1}^{T}(1-\epsilon_t).$$

If $\epsilon_t \geq \epsilon$ for infinitely many values of t, this product is bounded above by $(1-\epsilon)^T$, which converges to zero as $T \to \infty$. □

Proof of Proposition

By the preceding lemma, the probability that an autoregressive sequence remains within $\mathcal{M}$ over arbitrarily long horizons converges to zero. Because autoregressive architectures lack a mechanism for rendering invariant-violating transitions undefined, they cannot enforce $\epsilon_t = 0$ uniformly across time. Consequently, divergence from $\mathcal{M}$ is not merely possible but inevitable in the limit.

This establishes that autoregression cannot guarantee invariant preservation over unbounded horizons, regardless of model capacity, training data size, or decoding strategy. □

A.2 Proof of Replay-Stabilized Consistency

We restate the theorem.

Theorem 5 (Replay-Stabilized Consistency)**.** *If an event-log system employs invariant-gated commitment, then every reachable authoritative state satisfies all system invariants.*

Definitions

Let $\mathcal{L} = (e_1, \ldots, e_n)$ denote a finite event log, let σ_0 be the initial state, and let

$$\mathrm{Replay}(\mathcal{L}) = \delta(\cdots \delta(\delta(\sigma_0, e_1), e_2), \ldots, e_n)$$

be the deterministic replay function induced by the transition operator δ. Let $\Omega \subseteq \Sigma$ denote the set of admissible states.

Proof

The proof proceeds by induction on the length of the event log.

For the base case, the empty log $\emptyset$ yields $\mathrm{Replay}(\emptyset) = \sigma_0$, which lies in Ω by assumption.

For the inductive step, assume that for some $n \geq 0$ the replayed state $\mathrm{Replay}(e_1, \ldots, e_n)$ lies in Ω. Consider a proposed event e_{n+1}. By definition of invariant-gated commitment, this event is appended to the log if and only if

$$\delta(\mathrm{Replay}(e_1, \ldots, e_n), e_{n+1}) \in \Omega.$$

If the condition is satisfied, replay of the extended log produces an admissible state. If the condition is not satisfied, the event is rejected and the authoritative state remains unchanged. In either case, the invariant is preserved.

By induction, all reachable authoritative states are admissible. □

A.3 Proof of Non-Compensability of Invariants

We restate the proposition.

Proposition 10 (Non-Compensability of Invariants)**.** *No finite penalty-based objective can enforce an invariant over unbounded planning horizons.*

Setup

Let $J : \Sigma \to \mathbb{R}$ be a reward function and let $P : \Sigma \to \mathbb{R}$ be a penalty function satisfying $|P(\sigma)| \leq M$ for some finite constant $M > 0$. Suppose that invariant violations incur a penalty $-P(\sigma)$ but remain admissible under the systemâĂŹs transition dynamics.

Proof

Because the penalty magnitude is finite, there exists a planning horizon H and a sequence of actions leading to an invariant-violating state σ' such that the cumulative reward accumulated prior to reaching σ' exceeds M. For sufficiently large H, the total return of the plan satisfies

$$\sum_{t=1}^{H} J(\sigma_t) - P(\sigma') > 0.$$

As a result, σ' becomes optimal under the objective function despite violating the intended constraint.

By contrast, invariant-gated systems render σ' unreachable, independently of reward structure, planning horizon, or discounting scheme. Finite penalties therefore cannot substitute for hard invariants. □

A.4 Discussion

The results in this appendix formalize three complementary claims. First, probabilistic suppression of illegal states is insufficient to guarantee long-horizon coherence. Second, invariant-gated replay provides absolute admissibility guarantees by construction. Third, safety and structural constraints cannot be reduced to optimization objectives without loss of enforceability.

Taken together, these results support the central thesis of the paper: commitment, rather than prediction, is the fundamental unit of intelligence.

B Illustrative Examples of Invariant Enforcement

This appendix provides extended, concrete examples illustrating the behavioral differences between view-only predictive systems and invariant-preserving architectures. The purpose of these examples is not to introduce new formal machinery, but to make explicit how the abstract commitments of the framework manifest in familiar domains such as language and physical planning.

B.1 Language Example: Syntactic Repair Under Structural Invariants

Consider the partially constructed sentence

“The keys to the cabinet is missing.”

At the point where the verb form is selected, the system must choose between alternative agreement options. In many corpora, surface-level frequency biases favor singular constructions following intervening singular nouns such as *cabinet*. As a result, a view-only autoregressive system may assign high probability to the singular verb form *is*, even though it violates subject–verb agreement.

In an invariant-preserving system, the proposed verb insertion is treated as an event whose admissibility is evaluated against syntactic invariants derived from the authoritative structure of the sentence. Because the grammatical subject *keys* is plural, the event inserting a singular verb fails invariant validation and is therefore rejected outright. Importantly, rejection does not depend on probability thresholds or post hoc correction, but on structural inadmissibility.

Repair proceeds by proposing alternative events that satisfy the same contextual role while respecting the invariant constraints. The plural verb form *are* is admissible, allowing the sentence to be completed without violating grammatical structure. The key point is that the system does not merely prefer correct forms; it enforces them by construction, ensuring that syntactic well-formedness is preserved even under perturbation or misleading local cues.

B.2 Physical Planning Example: Escalation Under Constraint Violation

Consider an embodied agent tasked with moving a rigid block through a narrow passage. Through prior experience, the agent has cached a motor schema for transporting blocks along straight-line trajectories. Under typical conditions, this schema executes efficiently and without invoking higher-level planning.

In the present scenario, however, the cached schema proposes a trajectory that intersects the passage walls, violating spatial and physical constraints. In a view-only system, the agent may continue to execute the action, repeatedly colliding with the obstacle or producing incoherent behavior that implicitly assumes the passage is wider than it is.

By contrast, an invariant-preserving system detects the violation at the level of authoritative transition evaluation. The proposed motor event is rejected because it would lead to an inadmissible physical state. This rejection triggers escalation: the system temporarily suspends the cached schema and invokes arbiter-mediated planning to search for an alternative sequence of actions. The resulting plan may involve rotating the block, adjusting orientation, or selecting a longer but feasible path through the passage.

Crucially, the failure of the cached behavior does not result in hallucinated success. Instead, it produces a controlled refusal followed by deliberate re-planning, preserving physical feasibility throughout execution.

B.3 Failure Mode Comparison and Diagnostic Signatures

These examples highlight a fundamental difference in how failure manifests across architectures. In view-only predictive systems, failure typically appears as confident but incorrect continuation. The system proceeds as though constraints were satisfied, producing outputs that are locally plausible yet globally impossible. Because no authoritative state exists, the system lacks a principled mechanism for recognizing impossibility.

In invariant-preserving systems, failure is explicit and diagnosable. When no admissible transition exists, the system halts, escalates, or refuses to proceed. Such behavior may appear slower or more conservative, but it reflects a deeper form of reliability: the system does not trade correctness for fluency. Instead of hallucinating compliance with constraints, it acknowledges their violation and responds accordingly.

These diagnostic differences are central to the evaluation claims advanced in the main text. The presence of refusal, repair, and escalation behaviors serves as empirical evidence for authoritative history and invariant gating, distinguishing genuine world models from purely statistical approximations.

C Formal Clarifications and Interpretive Correspondences

This appendix records several interpretive correspondences that arise naturally from the formal structures introduced in the main body. The purpose is to disambiguate how these structures relate to standard notions in epistemology and cognitive theory, without extending the formal system itself.

C.1 Commitment and Knowledge

Let $\mathcal{L}$ denote an authoritative event log and let $\Omega \subseteq \Sigma$ denote the admissible state set. A commitment is defined as the successful incorporation of an event into $\mathcal{L}$ via invariant-gated transition. Such a commitment is irreversible in the sense that its removal would violate replay consistency.

Within this setting, knowledge may be identified with stable commitment. A proposition or action schema is known by the system if and only if it constrains future admissible transitions through its presence in the log. Justification corresponds to replay admissibility: an event is justified precisely when its inclusion preserves invariants under deterministic replay. No additional semantic or representational criterion is required.

This characterization replaces static truth conditions with dynamic consistency conditions. Knowledge is thus indexed to admissible histories rather than to instantaneous internal states.

C.2 Ontological Status of the Event Log

The event log $\mathcal{L}$ is not a representational structure mapping states of the world to internal symbols. Instead, it is a record of constraint-respecting interactions. Formally, $\mathcal{L}$ is a sequence in the event alphabet $\mathcal{E}$ such that replay yields a state in Ω at every prefix.

The authority of $\mathcal{L}$ derives from its construction rules rather than from correspondence relations. An eventâĂŹs presence in the log signifies that it was admissible under the systemâĂŹs transition constraints at the time of commitment. The log therefore functions as a boundary condition on future behavior rather than as a descriptive model of external reality.

C.3 Deliberation and Hypothetical Extension

Let $\mathcal{L}$ be the current authoritative log and let $\mathcal{L} \cdot e$ denote a hypothetical extension by event e. Deliberation corresponds to evaluation of such hypothetical extensions under invariant-gated transition without committing them to $\mathcal{L}$.

Formally, deliberation is the exploration of partial transition paths in the induced invariant-preserving transition system, subject to rejection prior to commitment. Automatic behavior corresponds to direct replay of cached event schemas whose admissibility has been previously established.

Escalation occurs when no cached schema yields an admissible extension. In such cases, the system reverts to explicit evaluation of hypothetical extensions. No special status is assigned to execution itself; the distinction lies in whether invariant validation is bypassed or explicitly invoked.

C.4 Discrete Commitment and Substrate Neutrality

The formalism does not impose discreteness, determinism, or symbolic structure at the level of physical implementation. Continuous, stochastic, or distributed substrates may realize the transition function δ and the invariant set Ω.

The claim is limited to the level of commitment. For any system to maintain coherence across time, the distinction between committed and uncommitted transitions must be discrete. Either an event is incorporated into the authoritative history or it is not. This discreteness is a property of admissibility, not of the underlying dynamics.

Accordingly, the framework is neutral with respect to the ontology of mental states, neural mechanisms, or physical realization. It specifies only the minimal structural conditions required for invariant-preserving action over extended horizons.

References

[1] Craik, K. (1943). *The Nature of Explanation.* Cambridge University Press.

[2] Newell, A., & Simon, H. A. (1976). Computer science as empirical inquiry: Symbols and search. *Communications of the ACM*, 19(3), 113–126.

[3] Chomsky, N. (1957). *Syntactic Structures.* Mouton.

[4] Chomsky, N. (1965). *Aspects of the Theory of Syntax.* MIT Press.

[5] Fodor, J. A. (1983). *The Modularity of Mind.* MIT Press.

[6] Pearl, J. (2009). *Causality: Models, Reasoning, and Inference* (2nd ed.). Cambridge University Press.

[7] Ha, D., & Schmidhuber, J. (2018). World models. *arXiv preprint arXiv:1803.10122.*

[8] Bender, E. M., & Koller, A. (2020). Climbing towards NLU: On meaning, form, and understanding in the age of data. In *Proceedings of ACL.*

[9] Marcus, G. (2020). The next decade in AI: Four steps towards robust artificial intelligence. *arXiv preprint arXiv:2002.06177.*

[10] LeCun, Y. (2022). A path towards autonomous machine intelligence. Meta AI Research Position Paper.

[11] LeCun, Y. (2023). What is intelligence? Public lectures and interviews, Meta AI.

[12] Murphy, E. (2023). ROSE: A neurocomputational architecture for syntax. *arXiv preprint arXiv:2303.08877*.

[13] Berwick, R. C., & Chomsky, N. (2011). The biolinguistic program: The current state of its evolution. In *The Biolinguistic Enterprise*. Oxford University Press.

[14] Tenenbaum, J. B., Kemp, C., Griffiths, T. L., & Goodman, N. D. (2011). How to grow a mind: Statistics, structure, and abstraction. *Science*, 331(6022), 1279–1285.

[15] Friston, K. (2010). The free-energy principle: A unified brain theory? *Nature Reviews Neuroscience*, 11(2), 127–138.

[16] Clark, A. (2016). *Surfing Uncertainty: Prediction, Action, and the Embodied Mind*. Oxford University Press.

[17] Dennett, D. C. (1987). *The Intentional Stance*. MIT Press.

[18] Laird, J. E. (2012). *The Soar Cognitive Architecture*. MIT Press.

[19] Anderson, J. R. (2007). *How Can the Human Mind Occur in the Physical Universe?* Oxford University Press.

[20] Fowler, M. (2005). Event sourcing. *martinfowler.com*.

[21] Young, G. (2010). CQRS documents. *gregyoung.com*.

[22] Buzsaki, G. (2015). Hippocampal sharp waveâĂŞripple: A cognitive biomarker for episodic memory and planning. *Hippocampus*, 25(10), 1073–1188.

[23] Lake, B. M., Ullman, T. D., Tenenbaum, J. B., & Gershman, S. J. (2017). Building machines that learn and think like people. *Behavioral and Brain Sciences*, 40.

[24] Jaeger, H., & Haas, H. (2004). Harnessing nonlinearity: Predicting chaotic systems and saving energy in wireless communication. *Science*, 304(5667), 78–80.

[25] Vernon, D. (2014). *Artificial Cognitive Systems*. MIT Press.

[26] Turing, A. M. (1950). Computing machinery and intelligence. *Mind*, 59(236), 433–460.

[27] Shannon, C. E. (1948). A mathematical theory of communication. *Bell System Technical Journal*, 27, 379–423.

[28] Kanerva, P. (1988). *Sparse Distributed Memory*. MIT Press.

[29] Goodfellow, I., et al. (2014). Generative adversarial nets. In *Advances in Neural Information Processing Systems*.

[30] Vaswani, A., et al. (2017). Attention is all you need. In *Advances in Neural Information Processing Systems*.

[31] Browning, B., et al. (2023). Why language models hallucinate. *arXiv preprint arXiv:2305.13534*.