# From Accumulation to Closure: Inference, Memory, and the Constraint-Theoretic Turn

Reconstruction, Composability, and the Political Economy of State

Flyxion

Independent Researcher

April 10, 2026

**Abstract**

This essay develops a unified critique of the dominant paradigms for inference and knowledge organization in artificial intelligence, grounded in the formal machinery of sheaf cohomology, inverse problem theory, gauge symmetry, and category theory. The central claim is that systems organized around prediction, retrieval, or mere accumulation are structurally insufficient to achieve global epistemic coherence, because none enforces the condition that all retained information be simultaneously realizable as projections of a single underlying world-state. We introduce constraint closure as the missing principle, prove that its absence makes globally inconsistent outputs unavoidable, and formalize the transition from plausibility to realizability as a categorical shift in the architecture of inference. We further argue that this failure is not confined to AI systems: the progressive restriction of user access to underlying computational state in contemporary software ecosystems instantiates the same structural failure at the level of human–computer interaction. Interface functors that are not faithful collapse distinctions the user needs to enforce constraints; composability erodes as a deliberate consequence of platform economic incentives. We unify both domains under a single formal principle—that coherence requires access to state—and propose realizability rate, constraint violation rate, and feasible set diameter as evaluation metrics that capture global properties invisible to standard accuracy measures. An appendix provides an explicit table of isomorphisms mapping concepts across sheaf theory, AI inference, and platform design.

# Contents


**1 The Exhaustion of Stateless Inference** **3**

1.1 Prediction Without Realizability . . . . . 3

1.2 Chain of Thought as Post Hoc Narration . . . . . 4

1.3 The Emergence of Persistent Structure as Partial Corrective . . . . . 4

1.4 Thesis and Roadmap . . . . . 5

**2 Observability and the Inverse Problem of Inference** **6**

2.1 Inference as an Inverse Problem . . . . . 6

2.2 Observability and Identifiability . . . . . 6

2.3 Projection Degeneracy and the Limits of Retrieval . . . . . 7

2.4 Constraint Regularization as Inverse Stabilization . . . . . 8

2.5 Observability, Composability, and the Restriction of State . . . . . 8

**3 Accumulation Without Closure: The Sheaf-Theoretic Diagnosis** **9**

3.1 Local Sections and the Failure of Global Gluing . . . . . 9

3.2 The Distinction Between Semantic Caching and Constraint Resolution 10

**4 Gauge Freedom and Representation Dependence** **11**

4.1 Equivalence Classes of Representation . . . . . 11

4.2 Gauge Fixing and the Conditions for Consistent Enforcement . . . . . 12

4.3 Gauge-Invariant Constraint Enforcement . . . . . 12

4.4 Gauge Freedom, Non-Identifiability, and the Structure of Ambiguity . 13

**5 Chain of Memory and the Causal Grounding Requirement** **13**

5.1 From Token Trajectories to State Trajectories . . . . . 13

5.2 Transition Operators as Entropic Differentials . . . . . 14

5.3 The Hierarchy of Inference Paradigms . . . . . 15

5.4 Non-Identifiability and the Structure of Failure . . . . . 15

**6 The Compounding of Knowledge as Contraction of Feasibility** **16**

6.1 Compounding as Reduction, Not Addition . . . . . 16

6.2 Trajectory-Aware Coarse-Graining and Obstruction Preservation . . . 17

6.3 Novelty as Ontological Insufficiency . . . . . 18

6.4 Fixed Points, Attractors, and the Geometry of Closure . . . . . 19

**7 The Enshittification of State: Platform Legibility and the Restriction of Composition** **20**

7.1 From AI Architecture to Computational Political Economy . . . . . 20

7.2 Hidden State and the Elimination of Verification . . . . . 20
7.3 Interfaces as Functors and the Loss of Faithfulness . . . . . 21
7.4 The Political Economy of Opacity . . . . . 22
7.5 Composability as a Formal Condition of Epistemic Agency . . . . . 22

**8 Evaluation Metrics Under the Realizability Criterion 23**
8.1 The Limits of Marginal Accuracy . . . . . 23
8.2 Realizability Rate and Constraint Violation . . . . . 24
8.3 Feasible Set Diameter as Epistemic Progress . . . . . 24
8.4 Evaluation as Fixed-Point Convergence . . . . . 25

**9 The Necessity of Constraint Closure 25**
9.1 State of the Argument . . . . . 25
9.2 Main Theorem . . . . . 26
9.3 Corollaries and Structural Consequences . . . . . 27

**10 Conclusion: Towards Realizable Intelligence 27**
10.1 The Unified Argument . . . . . 27
10.2 The Realizability Turn . . . . . 28
10.3 Realizability as the Criterion . . . . . 28

**A Table of Structural Isomorphisms 30**

# 1 The Exhaustion of Stateless Inference

## 1.1 Prediction Without Realizability

The dominant paradigm of modern machine learning treats inference as the estimation of conditional distributions. A model $f_\theta$ trained to approximate $p(y \mid x)$ over a dataset $\mathcal{D} = \{(x_i, y_i)\}$ is understood to "know" something to the extent that it produces accurate outputs. The epistemology embedded in this formulation is implicitly local: the quality of a prediction is assessed per output, not across the totality of all outputs simultaneously. No condition is imposed that all predictions, considered jointly, correspond to a single globally realizable world-state. This absence of a realizability constraint is not a minor technical limitation. It is a structural feature that determines the kinds of failure modes the system is capable of exhibiting.

Consider the idealized scenario in which a model $f_\theta$ achieves zero expected loss on every marginal query. It is still entirely consistent with this achievement that the responses to distinct queries $q_1, q_2, \ldots, q_n$ are mutually incompatible when considered as claims about a shared domain. The model may assert that a particular event occurred at time $t_1$ when queried in one context and at time $t_2 \neq t_1$ when queried in another, both assertions being individually plausible given the training distribution but jointly unrealizable. This is not hallucination in the colloquial sense of gross factual error. It is hallucination in the deeper sense of local coherence without global consistency.

Let $\Omega$ denote the space of possible world-states and let $\pi_i : \Omega \to \mathcal{O}_i$ denote the projection from world-states to the observable relevant to query $q_i$. A system genuinely equipped with a world-model would produce outputs $o_i = \pi_i(\omega^*)$ for some fixed $\omega^* \in \Omega$. The requirement that such an $\omega^*$ exist is the realizability condition. Formally, a collection of responses $\{o_i\}$ is realizable if and only if the fiber product

$$\Omega^* = \bigcap_i \pi_i^{-1}(o_i) \tag{1}$$

is non-empty. A predictive system operating without this constraint may produce a collection $\{o_i\}$ for which $\Omega^* = \emptyset$: no world could produce all of these observations simultaneously. The system speaks fluently of a world that cannot exist.

The practical significance of this point is compounded by the scale of deployment. At query volumes of millions per day, a system that is locally plausible but globally inconsistent generates an enormous volume of internally contradictory representation with no mechanism for resolution. Users receive individually convincing outputs that cannot, in principle, be stitched together into a coherent picture. The system is not lying in any intentional sense; it is operating on a formalism in which truth is never

jointly enforced.

### 1.2 Chain of Thought as Post Hoc Narration

The introduction of Chain of Thought (CoT) prompting represented a meaningful improvement in performance on structured reasoning tasks, and yet the mechanism of that improvement is itself diagnostic of the underlying pathology. The key empirical finding, replicated across multiple experimental settings, is that the narrative produced by a CoT system is not causally upstream of the output in any mechanistically verifiable sense. Perturbing or even corrupting the intermediate reasoning steps often leaves the final answer unchanged. The narrative is better understood as a projection of the underlying computation rather than its generator.

This constitutes a specific failure of explanatory transparency. If the visible chain of reasoning does not determine the output, then it cannot be used to audit, verify, or intervene on the inference process. The explanation is an artifact produced alongside the answer, not the mechanism by which the answer was arrived at. This distinction matters precisely because the entire justification for CoT rests on the assumption that making intermediate steps explicit allows for error-detection and correction. If the steps are post hoc, they cannot perform this function.

The formal structure of the failure can be stated as follows. Let $M$ denote the internal state of the model during generation, and let $\ell$ denote the linguistic output including intermediate steps. The CoT assumption is that $\ell_{\text{reasoning}}$ is a faithful projection of $M$ and that $M$ determines the final output $o$. The empirical finding is instead that $\ell_{\text{reasoning}} \perp o \mid M$: the reasoning narrative and the final output are conditionally independent given the internal state. Both are projections of $M$ but are not causally related to each other. Evaluating the narrative therefore provides no information about whether the output was generated by a valid reasoning process.

This observation connects directly to the realizability problem. A system in which explanations are post hoc projections is one in which the underlying mechanism is inaccessible. Verification—the confirmation that an output corresponds to a valid trajectory through the constraint space—is impossible when the only available evidence is itself a decoupled projection. The system can narrate a path without having walked it.

### 1.3 The Emergence of Persistent Structure as Partial Corrective

Recognition of the limitations of purely stateless inference has driven significant interest in systems that maintain persistent intermediate structure. Rather than reconstructing answers from raw data at each query, such systems accumulate synthesis over time: summaries, cross-references, entity records, and the relationships

among them. The basic intuition is sound. If knowledge is to compound, there must be an object in which it compounds. Repeated reconstruction from scratch is not only computationally wasteful; it fails to preserve the higher-order structure—patterns of relation, detected contradictions, emergent themes—that emerges from extended engagement with a body of information.

A useful example of this tendency is the construction of interlinked knowledge bases maintained by language models, where new information is not merely indexed for retrieval but integrated into an evolving structured artifact. Each new source is read and its contents propagated across existing records, updating summaries, flagging inconsistencies, and strengthening relevant connections. The resulting structure is richer than a retrieval index because it encodes inferences, not just pointers.

The critical question, however, is what kind of object is being stored and what conditions govern its internal consistency. Persistence is not equivalent to coherence. A system can accumulate a vast and richly interconnected body of structured information while still failing to enforce the condition that all of that information is simultaneously realizable. Contradictions may be noted but left unresolved. Competing interpretations of the same evidence may coexist indefinitely as labeled alternatives. The structure grows denser without becoming more definite. Persistence therefore emerges as a necessary but manifestly insufficient condition for the kind of coherence that knowledge requires.

### 1.4 Thesis and Roadmap

The central claim of this essay can be stated as a single structural theorem, proved across the subsequent sections: any inference system that does not enforce global constraint closure over its state space will necessarily produce locally valid but jointly unrealizable outputs. Each section establishes one step of this argument.

Section 2 formalizes inference as an inverse problem and identifies non-identifiability as the canonical failure mode, showing that accumulation systems fail to resolve the inverse even when they preserve its inputs. Section 3 develops the sheaf-theoretic diagnosis: accumulation without closure corresponds to a presheaf that satisfies the locality axiom but violates the gluing axiom, with the cohomological obstruction $[\alpha] \in \check{H}^1(X, \mathcal{F})$ measuring the degree of unrealizability. Section 4 introduces gauge freedom as a structured form of non-identifiability that accumulation systems systematically mishandle as contradiction. Section 5 establishes that causal grounding, provided by memory-based systems, is necessary but not sufficient: a trajectory may be causally coherent while terminating outside the feasible set. Section 6 develops constraint closure as the missing condition and shows that compounding knowledge is contraction of the feasible set, not addition of content. Section 7 extends the argument to the political economy of software platforms, treating interface restriction as the

deliberate construction of non-faithful functors that eliminate user-level constraint enforcement. Section 8 proposes realizability-based evaluation metrics and proves that standard accuracy metrics are blind to the failure modes identified in earlier sections. Section 9 states and proves the main theorem that unifies all preceding results. The conclusion synthesizes the argument and identifies the realizability turn as a single coherent shift across both AI architecture and software design.

## 2 Observability and the Inverse Problem of Inference

### 2.1 Inference as an Inverse Problem

The problem of inference is, in the most general terms, an inverse problem. Let $\Omega$ denote the space of possible world-states and $\{\pi_i : \Omega \to \mathcal{O}_i\}_{i \in I}$ a family of observation operators corresponding to queries, measurements, or partial views of the domain. The forward problem is the computation of $\pi_i(\omega)$ for a known $\omega \in \Omega$. The inverse problem is the reconstruction of $\omega$ from a collection of observations $\{o_i\}_{i \in I}$ where $o_i = \pi_i(\omega)$ for some unknown $\omega$.

In most practical settings, the inverse problem is ill-posed in the sense of Hadamard: solutions may fail to exist, fail to be unique, or fail to depend continuously on the observations. The realizability condition introduced in Section 1 corresponds precisely to the existence of a solution:

$$\omega^* \in \bigcap_{i \in I} \pi_i^{-1}(o_i). \tag{2}$$

When this intersection is empty, the observations are mutually incompatible and no reconstruction is possible. When the intersection contains multiple elements, the problem is non-identifiable: multiple distinct world-states are consistent with the same observations. The central claim of this section is that the dominant paradigms of inference fail because they do not explicitly solve the inverse problem. They produce outputs that are locally consistent with the forward map but do not enforce that these outputs correspond to any single element of $\Omega$.

### 2.2 Observability and Identifiability

**Definition 1** (Observability)**.** *The system $(\Omega, \{\pi_i\}_{i \in I})$ is observable if for any $\omega_1, \omega_2 \in \Omega$ with $\omega_1 \neq \omega_2$, there exists $i \in I$ such that $\pi_i(\omega_1) \neq \pi_i(\omega_2)$.*

Observability ensures that distinct world-states produce distinguishable observation patterns. When observability holds and the full collection of observations is available, the inverse problem admits at most one solution. In practice, inference sys-

tems operate under partial observability: only a finite subset of observation operators is accessible at any given time, and the resulting inverse problem is underdetermined.

**Definition 2** (Non-Identifiability)**.** *A collection of observations* $\{o_i\}_{i \in I}$ *is non-identifiable if the feasible set*

$$F(\{o_i\}) = \bigcap_{i \in I} \pi_i^{-1}(o_i) \tag{3}$$

*contains more than one element.*

Non-identifiability is not an error condition but a structural property of the observation process. It reflects the fact that the available observations do not contain sufficient information to uniquely determine the underlying state. Any inference procedure must therefore impose additional structure—priors, regularization, or constraints—to select among the admissible solutions. The failure to do so results in the implicit selection of an arbitrary element of $F(\{o_i\})$ for each query, without guarantee that the selections across queries correspond to a common $\omega$.

### 2.3 Projection Degeneracy and the Limits of Retrieval

The failure of retrieval-based systems can be understood as a failure to address projection degeneracy. The observation operators $\pi_i$ are typically many-to-one maps: multiple world-states produce identical observations under a given query. A system that answers each query independently effectively selects an element of $\pi_i^{-1}(o_i)$ for each $i$ without enforcing that these selections correspond to a common element of $\Omega$.

Formally, let $s_i \in \pi_i^{-1}(o_i)$ denote the implicit state selected by the system in response to query $q_i$. Without a realizability constraint, the selections $\{s_i\}$ need not satisfy $s_i = s_j$ for $i \neq j$. The system constructs a family of states that are individually consistent with each observation but not jointly consistent with each other. This corresponds to choosing a section of the product space $\prod_i \pi_i^{-1}(o_i)$ that does not lie in the diagonal embedding of $\Omega$.

This perspective clarifies why increasing the number of retrieved documents or improving the quality of local summaries does not resolve the fundamental issue. The problem is not that the system lacks access to relevant information, but that it lacks a mechanism for enforcing that all selected information corresponds to a single underlying state. The degeneracy of the projection $\pi_i$ is not resolved by sampling more points in $\mathcal{O}_i$; it is resolved by introducing constraints that cut across the fibers of the projection.

### 2.4 Constraint Regularization as Inverse Stabilization

The standard approach to ill-posed inverse problems is regularization: the introduction of additional structure that selects a stable solution from the feasible set. In the present context, this corresponds to augmenting the observation-induced constraints with structural constraints $\mathcal{C}$ that restrict the admissible configurations.

Let $\mathcal{S}$ denote the full space of configurations (with $\Omega \subseteq \mathcal{S}$ the subspace of physically meaningful ones), and define the regularized feasible set:

$$F(\mathcal{C}, \{o_i\}) = \{s \in \mathcal{S} : \pi_i(s) = o_i \text{ for all } i \in I \text{ and } c(s) = \top \text{ for all } c \in \mathcal{C}\}. \quad (4)$$

The role of the consistency operator $\Phi : \mathcal{S} \to \mathcal{S}$, developed fully in Section 6, is precisely to approximate a projection onto $F(\mathcal{C}, \{o_i\})$. When $\mathcal{C}$ is sufficiently rich and appropriately structured, the regularized feasible set may reduce to a singleton, restoring identifiability.

This reframes the distinction between accumulation and closure with precision. Accumulation systems increase the set of observations $\{o_i\}$ but do not systematically strengthen the constraint set $\mathcal{C}$. Closure systems treat each new piece of information as a constraint that modifies the geometry of the feasible set and triggers a recomputation of the inverse solution. The former grows the base of observations without controlling the shape of the solution space; the latter progressively contracts that space toward a unique configuration.

### 2.5 Observability, Composability, and the Restriction of State

This perspective also clarifies the relationship between observability and composability in human-computer interaction. A system that restricts access to the underlying state $\sigma$ effectively reduces the set of available observation operators $\{\pi_i\}$: fewer distinctions are visible, and the inverse problem becomes more underdetermined. The enclosure of state in software systems and the absence of constraint closure in inference systems are therefore two manifestations of the same structural limitation—insufficient observability of the underlying configuration space. The user who cannot inspect the raw format of a file and the model that cannot check the global consistency of its outputs are both operating with an artificially reduced $\{\pi_i\}$, and in both cases the consequence is non-identifiability of the underlying state. This structural parallel will be formalized in Section 7.

# 3 Accumulation Without Closure: The Sheaf-Theoretic Diagnosis

## 3.1 Local Sections and the Failure of Global Gluing

The failure of accumulation systems is not merely heuristic inconsistency. It is the failure of a presheaf to satisfy the gluing axiom—a structural deficiency that can be diagnosed precisely and measured quantitatively. The formal language of sheaf theory provides the appropriate vocabulary.

Let $X$ denote a topological or partially ordered space whose points represent domains, topics, or queries. For each open set $U \subseteq X$, let $\mathcal{F}(U)$ denote the set of admissible states—the configurations of knowledge that are internally consistent with the constraints operative over $U$. The assignment $U \mapsto \mathcal{F}(U)$ together with restriction maps $\rho_{UV} : \mathcal{F}(U) \to \mathcal{F}(V)$ for $V \subseteq U$ defines a presheaf.

This is a sheaf if and only if it satisfies two conditions. The locality condition requires that if two sections $s, t \in \mathcal{F}(U)$ agree on every element of some open cover $\{U_i\}$ of $U$, then $s = t$. The gluing condition requires that if sections $s_i \in \mathcal{F}(U_i)$ agree on all pairwise overlaps—that is, $\rho_{U_i, U_i \cap U_j}(s_i) = \rho_{U_j, U_i \cap U_j}(s_j)$ for all $i, j$—then there exists a unique global section $s \in \mathcal{F}(U)$ such that $\rho_{U,U_i}(s) = s_i$ for all $i$.

An accumulated knowledge structure behaves like a presheaf that satisfies the locality condition but fails the gluing condition. Each local section—each summary, entity record, or cross-referenced page—is internally consistent in the sense that it represents a coherent account of some restricted domain. But the pairwise compatibility conditions on overlapping domains are not guaranteed to yield a global section. The structure may contain pairs of locally consistent records whose joint content is not realizable: no world-state $\omega \in \Omega$ simultaneously satisfies both. Accumulation cross-references; it does not glue.

The obstruction to global gluing is measured by the first Čech cohomology group $\check{H}^1(X, \mathcal{F})$. A non-vanishing class $[\alpha] \in \check{H}^1(X, \mathcal{F})$ witnesses the existence of a consistent collection of local sections that cannot be assembled into a global one. The contradictions that a maintenance operation might flag are not mere organizational failures; they are cohomological obstructions. Persistence alone cannot eliminate them: the fact that the contradiction is recorded does not make the contradicting sections compatible.

**Definition 3** (Realizability Obstruction)**.** *Let $\mathcal{F}$ be a sheaf of admissible states over a domain space $X$, and let $\{s_i \in \mathcal{F}(U_i)\}$ be a collection of local sections on an open cover $\{U_i\}$. The collection exhibits a realizability obstruction if the Čech cocycle*

$\alpha \in \check{C}^1(\{U_i\}, \mathcal{F})$ *defined by*

$$\alpha_{ij} = \rho_{U_i, U_i \cap U_j}(s_i) - \rho_{U_j, U_i \cap U_j}(s_j) \tag{5}$$

*is not a coboundary, i.e.,* $[\alpha] \neq 0$ *in* $\check{H}^1(X, \mathcal{F})$.

A knowledge accumulation system that notes contradictions without resolving them is one that detects non-trivial cocycles without eliminating them. The contradiction record indexes which pairs of sections are incompatible, but the accumulated structure contains no mechanism for determining whether the cocycle can be made trivial by modifying local sections in a constraint-respecting way. Persistence extends the record; it does not require that coherence require closure.

### 3.2 The Distinction Between Semantic Caching and Constraint Resolution

The distinction between a knowledge accumulation system and a constraint-closing reconstruction engine can be drawn most sharply by examining what each system does with new information. In an accumulation system, a new source is integrated by reading, summarizing, and propagating updates across relevant records. The process terminates when all affected records have been revised. In a reconstruction engine, new information introduces new constraints that modify the feasible set of global configurations. The process terminates—if it terminates—at a fixed point of the consistency operator, not when a finite set of documents has been updated.

Let $\mathcal{C}$ denote the set of constraints currently active in the system and $\mathcal{S}$ the full space of configurations. The feasible set is

$$F(\mathcal{C}) = \{s \in \mathcal{S} : c(s) = \top \text{ for all } c \in \mathcal{C}\}. \tag{6}$$

A consistency operator $\Phi : \mathcal{S} \to \mathcal{S}$ maps any configuration to one that satisfies as many constraints as possible under some principled loss function. The fixed points of $\Phi$ are the stable solutions; the global section of the corresponding sheaf, if it exists, is such a fixed point.

An accumulation system operates on $\mathcal{C}$ by appending new constraints as new information arrives. But it does not recompute $F(\mathcal{C})$ after each addition, and it does not apply $\Phi$ iteratively until convergence. The state of the system after $n$ ingestions is not $F(\mathcal{C}_n)$ but rather a superposition of local updates that approximates $F(\mathcal{C}_n)$ only to the extent that local edits to shared records happen to produce global compatibility. When they do not, the system contains an implicit superposition of incompatible world-states that no query will force it to resolve. This is precisely why persistence alone cannot yield closure.

**Theorem 1** (Insufficiency of Append-Only Accumulation)**.** *Let $\{c_t\}_{t\geq 0}$ be a sequence of constraints added to an accumulation system, and let $K_t$ denote the state of the system after $t$ ingestions under an append-only update rule. Then there exist constraint sequences and initial configurations such that for all $t$, the state $K_t$ is not contained in $F(\mathcal{C}_t)$, and the divergence $d(K_t, F(\mathcal{C}_t))$ grows without bound.*

*Proof Sketch.* Construct a sequence such that $c_{2k}$ directly contradicts $c_{2k-1}$ on record $r_{2k}$ while leaving all other records unchanged. Under an append-only rule that revises $r_{2k}$ in response to $c_{2k}$ without re-examining prior constraints, the system alternately satisfies $c_{2k-1}$ and $c_{2k}$ on the affected record. At each even time step $c_{2k-1}$ is violated; at each odd step $c_{2k}$ is violated. Since no mechanism forces joint satisfaction, the system never enters $F(\mathcal{C}_t)$, and if the magnitude of the contradiction grows with $k$, the divergence is unbounded. □

The theorem is deliberately minimal in its assumptions. It does not require adversarial inputs, unusual domains, or extreme parameter settings. It requires only that the system lack a mechanism for enforcing joint satisfiability across all active constraints. Therefore, causal grounding is required—but as the next section shows, it too is insufficient.

# 4 Gauge Freedom and Representation Dependence

## 4.1 Equivalence Classes of Representation

A central ambiguity in any representational system is that multiple configurations may encode the same underlying state of affairs. This phenomenon is well-understood in physics as gauge freedom: distinct representations correspond to a single physical reality, and only gauge-invariant quantities are observable. Its presence in knowledge systems has significant consequences for how apparent contradictions are diagnosed and resolved.

Let $\Omega$ denote the space of underlying world-states and $\mathcal{S}$ the space of representations used by the system. A representation map $\rho : \Omega \to \mathcal{S}$ need not be injective. Conversely, multiple elements of $\mathcal{S}$ may correspond to the same observational content.

**Definition 4** (Gauge Equivalence)**.** *Two configurations $s, s' \in \mathcal{S}$ are gauge equivalent, written $s \sim s'$, if*

$$\pi_i(s) = \pi_i(s') \quad \textit{for all } i \in I. \tag{7}$$

The quotient space $\mathcal{S}/\!\sim$ defines equivalence classes of representations that are indistinguishable under all available observations. These equivalence classes are the true objects of inference; individual representatives are artifacts of the chosen coordinate system.

### 4.2 Gauge Fixing and the Conditions for Consistent Enforcement

Inference systems that operate on representations must implicitly or explicitly choose a representative from each gauge equivalence class. This process is gauge fixing. A gauge fixing is a selection function $\gamma : \mathcal{S}/\sim \to \mathcal{S}$ that assigns a canonical representative to each equivalence class.

In the absence of gauge fixing, accumulated knowledge structures may treat gauge-equivalent representations as distinct entities. This leads to spurious inconsistencies: differences that are artifacts of representation rather than genuine contradictions in the underlying state. A system that detects the discrepancy between two descriptions of the same event—written in different formats, attributed to different sources, or indexed under different headings—and treats it as a contradiction has committed a gauge error.

**Definition 5** (Gauge Mismatch)**.** *An apparent inconsistency between $s_1$ and $s_2$ is a gauge mismatch if there exist gauge-equivalent representatives $s_1' \sim s_1$ and $s_2' \sim s_2$ such that $s_1'$ and $s_2'$ are compatible.*

Accumulation systems that lack gauge awareness may treat gauge mismatches as obstructions, leading to unnecessary branching, duplication, or conflict records. This inflates the apparent complexity of the knowledge base and obscures genuine inconsistencies. Conversely, systems that ignore representation dependence entirely may fail to detect genuine inconsistencies that persist under all gauge transformations.

### 4.3 Gauge-Invariant Constraint Enforcement

Constraint closure must operate on gauge-invariant quantities. A constraint $c : \mathcal{S} \to \{\top, \bot\}$ is gauge-invariant if

$$s \sim s' \;\Rightarrow\; c(s) = c(s'). \tag{8}$$

**Proposition 1.** *Only gauge-invariant constraints are well-defined on the quotient space $\mathcal{S}/\sim$. Non-invariant constraints depend on arbitrary representation choices and cannot be consistently enforced across equivalent configurations.*

This implies that the consistency operator $\Phi$ must act on equivalence classes rather than individual representations, or else incorporate an explicit gauge-fixing step that ensures representation consistency before constraint evaluation. A system that enforces $\Phi$ on raw representations without gauge fixing may converge to a fixed point that satisfies all constraints as written but fails to correspond to a genuinely consistent world-state, because the apparent consistency is a consequence of having locked in a particular gauge.

### 4.4 Gauge Freedom, Non-Identifiability, and the Structure of Ambiguity

Gauge freedom is a structured form of non-identifiability. In the language of Section 2, the feasible set $F(\mathcal{C})$ may contain multiple elements that are indistinguishable under all observations but related by gauge transformations. The presence of such equivalence classes implies that the inverse problem admits multiple solutions that are observationally identical.

The task of inference is therefore twofold: to reduce the feasible set as much as possible through constraint addition, and to factor out residual gauge freedom so that remaining ambiguity corresponds only to unobservable degrees of freedom. A system that conflates gauge ambiguity with epistemic uncertainty will mischaracterize the structure of its own ignorance, acquiring new observations to resolve what is in fact a representational choice rather than a factual gap.

Gauge-aware closure distinguishes between reducible ambiguity, which can be eliminated by new constraints, and irreducible ambiguity, which reflects genuine symmetry in the system. This distinction is essential for determining when further data acquisition is necessary and when the system has reached the limits of identifiability. It is, in a precise sense, the distinction between not knowing and not being able to know—and any system that cannot make this distinction cannot accurately represent its own epistemic state.

## 5 Chain of Memory and the Causal Grounding Requirement

### 5.1 From Token Trajectories to State Trajectories

The paradigm of Chain of Memory (CoM) represents a substantively different approach to the problem of grounded reasoning. Where standard language model inference treats reasoning as a sequence of token productions, CoM reframes it as a sequence of transformations in a latent memory state. Formally, let $h_t \in \mathcal{H}$ denote the hidden memory state at step $t$, and let $T_t : \mathcal{H} \to \mathcal{H}$ denote the transition operator applied at that step. The reasoning process is then the trajectory

$$h_0 \xrightarrow{T_1} h_1 \xrightarrow{T_2} h_2 \xrightarrow{T_3} \cdots \xrightarrow{T_n} h_n, \tag{9}$$

and the output $o$ is a function $\pi : \mathcal{H} \to \mathcal{O}$ of the terminal state $h_n$, not of the sequence of linguistic tokens.

The decisive advantage of this formulation is the restoration of causal grounding. Because the output is determined by $h_n$ and $h_n$ is determined by the sequence of transitions $T_1, \ldots, T_n$ applied to $h_0$, the reasoning process is causally upstream of

the output. The trajectory is not a post hoc reconstruction of the computation; it is the computation. Perturbation-based verification therefore becomes meaningful: if one perturbs $T_k$ for some $k < n$ and recomputes the trajectory, the resulting output should differ from the original in a predictable way if the trajectory was genuinely responsible for the output.

This causal structure is exactly what is absent in standard CoT. In CoT, the linguistic chain $\ell_1, \ell_2, \ldots, \ell_m$ is produced by the same mechanism that produces the output token $o$, but there is no formal guarantee that modifying $\ell_k$ forces a consistent revision of the downstream tokens. The chain is a projection of the internal state, not a driver of it. In CoM, by contrast, the state trajectory is the driver, and the linguistic output is the projection. The directionality of dependence is reversed. The three paradigms can be summarized with precision: CoT produces linguistic projections with no causal backbone; CoM produces causally grounded trajectories; constraint closure additionally enforces that the terminal state of those trajectories lies within the feasible set. Chain of Memory is therefore a necessary but not sufficient condition for realizability.

**Definition 6** (Causal Grounding)**.** *A reasoning system is causally grounded if there exists a latent state space $\mathcal{H}$, a transition structure $\{T_t\}$, and a projection $\pi : \mathcal{H} \to \mathcal{O}$ such that the output satisfies $o = \pi(h_n)$ where $h_n$ is uniquely determined by the initial state $h_0$ and the applied transitions. A reasoning system is causally ungrounded if the output is produced by a mechanism that does not factor through a traceable state trajectory.*

Standard next-token prediction, including CoT, is causally ungrounded in this sense: the internal state $M$ of the model at generation time is not accessible, not traceable, and not verifiably responsible for the output through any inspectable trajectory. The narrative is a shadow, not a cause.

### 5.2 Transition Operators as Entropic Differentials

A more precise characterization of the transition operators $T_t$ clarifies why causal grounding, while necessary, does not by itself enforce realizability. Each $T_t$ should be understood as a local entropic differential: a map that decreases local uncertainty in the memory state by conditioning on newly incorporated evidence. Define the local entropy at step $t$ as $H(h_t) = -\sum_\omega p(\omega \mid h_t) \log p(\omega \mid h_t)$, where $p(\omega \mid h_t)$ denotes the distribution over world-states implied by the current memory state.

A well-behaved transition operator satisfies

$$H(h_{t+1}) \leq H(h_t), \tag{10}$$

with equality only when no new constraint is incorporated at step $t$. This formalization captures the sense in which reasoning is an entropy-reducing process: the trajectory moves from a broad distribution over world-states to a progressively narrower one. However, the sequence of local reductions $H(h_0) \geq H(h_1) \geq \cdots \geq H(h_n)$ does not guarantee that $h_n$ lies in $F(\mathcal{C})$. The trajectory may converge to a configuration that is locally consistent at each step but globally inconsistent with some constraint not yet encountered. This is the content of the insufficiency claim: causal grounding ensures the trajectory is real; closure ensures the destination is valid.

### 5.3 The Hierarchy of Inference Paradigms

The conceptual progression from retrieval to accumulation to memory to closure can be made precise as a hierarchy of increasingly strong structural requirements. Retrieval-based systems impose no persistence requirement: the answer to a query is computed from raw data at query time with no memory of prior synthesis, given by $o_q = f(q, D)$ where $D$ is the document collection and $f$ has no accumulated state.

Accumulation-based systems add a persistent state $K$ that evolves as $K_{t+1} = \text{Update}(K_t, d_{t+1})$ where $d_{t+1}$ is a new document. As established in Theorem 1, this does not guarantee $K_t \in F(\mathcal{C}_t)$.

Memory-based systems add causal grounding: the state trajectory $\{h_t\}$ is causally responsible for the output and is in principle verifiable. This restores the possibility of auditing the reasoning process, but does not impose the requirement that $h_n \in F(\mathcal{C})$.

Constraint-closure systems impose the additional requirement that the terminal state of any reasoning trajectory lie within $F(\mathcal{C})$, and that the consistency operator $\Phi$ converges to a fixed point within finitely many applications.

**Definition 7** (Closure Condition)**.** *A reasoning system satisfies the closure condition with respect to a constraint set $\mathcal{C}$ and consistency operator $\Phi$ if for every initial state $h_0 \in \mathcal{H}$, the iteration $h_{k+1} = \Phi(h_k)$ converges to a fixed point $h^* \in F(\mathcal{C})$ in finitely many steps, and the projection $\pi(h^*)$ is a realizable response to the query.*

The four paradigms form a strict chain: retrieval $\subset$ accumulation $\subset$ memory $\subset$ closure. Each imposes a condition absent from the preceding level. For tasks requiring global epistemic coherence—consistent world-modelling across arbitrarily many queries over an extended period—only closure is sufficient.

### 5.4 Non-Identifiability and the Structure of Failure

The closure paradigm also clarifies the structure of failure modes in a way the preceding paradigms do not. In a retrieval system, failure appears as missing information. In an accumulation system, failure appears as inconsistency. In

a memory-based system, failure appears as trajectory instability. In a closure-based system, all of these are subsumed under a single structural category: non-identifiability.

Non-identifiability occurs when $F(\mathcal{C})$ contains more than one element, meaning the active constraints are insufficient to uniquely determine the global configuration. Non-identifiability can arise from two distinct sources. The first is projection degeneracy: the observation operators $\{\pi_i\}$ are insufficiently informative to separate elements of $\Omega$. The second is regularizer flatness: the prior or regularization term placed on $\Omega$ is too diffuse to enforce uniqueness even in regions where the observations do not discriminate. Distinguishing between these sources matters for intervention: projection degeneracy calls for the acquisition of new types of observations, while regularizer flatness calls for the strengthening of structural priors.

**Proposition 2** (Non-Identifiability as Cohomological Flatness)**.** *Let $F(\mathcal{C})$ be a convex subset of $\mathcal{S}$ and suppose $|F(\mathcal{C})| > 1$. Then the sheaf $\mathcal{F}$ of admissible sections over the associated domain space $X$ admits a non-trivial global section for each element of $F(\mathcal{C})$, and the multiplicity of global sections reflects the dimension of $F(\mathcal{C})$ as a convex set.*

A large feasible set corresponds to a sheaf that is "flat" in the direction of the ambiguity. The task of constraint strengthening is the geometric task of reducing the dimension of $F(\mathcal{C})$ by adding hyperplane-like constraints that cut through the feasible set, ideally until $|F(\mathcal{C})| = 1$.

# 6 The Compounding of Knowledge as Contraction of Feasibility

## 6.1 Compounding as Reduction, Not Addition

There is a widespread intuition that the accumulation of knowledge is additive: the system knows more as it ingests more. This intuition is misleading in a precise sense. Genuine epistemic progress is not the addition of statements to a database; it is the contraction of the feasible set. Each new constraint eliminates possible world-states that are incompatible with the available evidence. Progress is measured not by how much has been written or stored, but by how many alternatives have been ruled out.

Let $F_0 = \mathcal{S}$ denote the initial feasible set, equal to the entire configuration space before any constraints have been applied. After the introduction of constraint $c_1$, the feasible set becomes $F_1 = F_0 \cap c_1^{-1}(\top)$. Subsequent constraints further restrict: $F_t = F_{t-1} \cap c_t^{-1}(\top)$. In the ideal case, $\lim_{t\to\infty} F_t = \{\omega^*\}$ for a unique global configuration $\omega^*$. This is the limiting case of full identifiability: the trajectory of constraint accumulation converges to a point.

**Proposition 3** (Monotonic Contraction Under Consistent Constraint Addition)**.** *If constraints $\{c_t\}_{t\geq 1}$ are mutually consistent and each is non-redundant relative to the preceding set, then $F(\mathcal{C}_{t+1}) \subsetneq F(\mathcal{C}_t)$ for all $t \geq 0$.*

*Proof.* Mutual consistency ensures $F(\mathcal{C}_{t+1}) \neq \emptyset$. Non-redundancy of $c_{t+1}$ relative to $\mathcal{C}_t$ means there exists $s \in F(\mathcal{C}_t)$ such that $c_{t+1}(s) = \bot$. Therefore $F(\mathcal{C}_{t+1}) = F(\mathcal{C}_t) \cap c_{t+1}^{-1}(\top) \subsetneq F(\mathcal{C}_t)$. □

An accumulation system fails to achieve monotonic contraction because its update rule does not guarantee non-redundancy or mutual consistency: it appends information without verifying that the new content jointly shrinks rather than fragmenting the feasible set.

This inversion from addition to contraction has concrete implications for what counts as a successful update. An accumulation system succeeds when it stores new information without losing old information. A closure system succeeds when it eliminates a class of world-states previously compatible with the evidence. A closure system fails when two incompatible constraints are both retained, because $F(\mathcal{C}) = \emptyset$—the empty feasible set, corresponding to the system having committed to an impossible world.

**Remark 1.** *The empty feasible set is not a catastrophic failure in the same sense as data loss. It is a diagnostic signal: the active constraints are collectively unsatisfiable, meaning that at least one must be relaxed or reinterpreted. A system equipped with a mechanism for detecting and responding to $F(\mathcal{C}) = \emptyset$ is more epistemically honest than one that silently accommodates contradictions by treating them as annotations.*

### 6.2 Trajectory-Aware Coarse-Graining and Obstruction Preservation

The temporal dimension of constraint closure introduces a mechanism of trajectory-aware coarse-graining. The core problem is that reasoning trajectories $\{h_t\}$ unfold at fine temporal resolutions while the constraints governing global coherence operate at coarser scales. A naive application of coarse-graining—averaging or summarizing the trajectory over some temporal window—risks eliminating the information that distinguishes realizable from unrealizable configurations.

Let $\Pi : \mathcal{H}^T \to \mathcal{H}_{\text{coarse}}$ denote a coarse-graining map. The map $\Pi$ is obstruction-preserving if and only if the induced map on cohomology

$$\Pi^* : \check{H}^1(X, \mathcal{F}_{\text{coarse}}) \to \check{H}^1(X, \mathcal{F}_{\text{fine}}) \tag{11}$$

is injective. Intuitively, $\Pi$ is obstruction-preserving if no non-trivial obstruction class at the coarse scale becomes trivial after coarse-graining. An obstruction-preserving

coarse-graining retains the information needed to detect realizability failures at all relevant scales.

**Definition 8** (Obstruction-Preserving Tiling Hierarchy)**.** *A tiling hierarchy is a sequence* $(\mathcal{T}_k, \mathcal{F}_k, \rho_{k,k-1})_{k=0}^{K}$ *where each* $\mathcal{T}_k$ *is a finite cover of the domain space* $X$*, each* $\mathcal{F}_k$ *is a sheaf of admissible states on* $\mathcal{T}_k$*, and each* $\rho_{k,k-1} : \mathcal{F}_k \to \mathcal{F}_{k-1}$ *is a restriction map that is injective on the first cohomology groups* $\check{H}^1$*.*

A structured noise term $\eta_t \in T_{h_t}\mathcal{H}$ added to each transition $T_t$ enters as a regularization mechanism. Pure constraint propagation may result in excessively sharp feasible sets that overfit to the available evidence. The noise introduces structured stochastic perturbations that prevent the trajectory from collapsing onto a single point too rapidly, maintaining a controlled degree of uncertainty while still enforcing convergence. The noise is structured in the sense that it is not isotropic: it respects the geometry of the admissibility manifold $\mathcal{A} \subset \mathcal{S}$ defined by the physical and logical constraints of the domain. This is analogous to the role of Langevin noise in statistical mechanics: structured stochastic perturbations that maintain ergodicity while respecting physical constraints.

### 6.3 Novelty as Ontological Insufficiency

Within this framework, the encounter with a genuinely novel entity or event is not merely a retrieval failure. It is a signal of ontological insufficiency: the current type ontology $\mathcal{O}$ does not contain the category needed to represent the new entity, and therefore no element of the current admissibility manifold $\mathcal{A}$ can accurately encode the relevant constraints.

Let $\tau : \text{Events} \to \mathcal{O}$ denote the current typing function. A novel event $e$ is one for which $\tau(e)$ is undefined or for which the constraints induced by $\tau(e)$ produce $F(\mathcal{C} \cup \{c_e\}) = \emptyset$, i.e., the new event is inconsistent with the existing ontology. This triggers a controlled ontological expansion: the admissibility manifold is extended to $\mathcal{A}' \supset \mathcal{A}$ in a way that minimizes the increase in model complexity while restoring non-emptiness of the feasible set. Formally, the optimal expansion satisfies

$$\mathcal{A}' = \arg \min_{\mathcal{A}'' \supset \mathcal{A},\, F(\mathcal{C} \cup \{c_e\}, \mathcal{A}'') \neq \emptyset} K(\mathcal{A}''), \tag{12}$$

where $K(\mathcal{A}'')$ is the Kolmogorov complexity of the extended manifold relative to the prior $\mathcal{A}$. This is a formal implementation of Occam's razor at the level of ontological expansion: the system adopts the smallest extension of its conceptual repertoire that accommodates the new evidence.

### 6.4 Fixed Points, Attractors, and the Geometry of Closure

The mathematical structure underlying the realizability turn can be articulated in terms of fixed-point theory and the geometry of attractors in the space of configurations. Let $\mathcal{S}$ be a metric space of configurations, and let $\Phi : \mathcal{S} \to \mathcal{S}$ be the consistency operator. Under the assumption that $\Phi$ is a contraction mapping—$d(\Phi(s), \Phi(s')) \leq \lambda d(s, s')$ for some $\lambda < 1$ and all $s, s' \in \mathcal{S}$—the Banach fixed-point theorem guarantees the existence and uniqueness of a fixed point $s^* \in \mathcal{S}$.

**Theorem 2** (Convergence Under Contraction)**.** *Let $(\mathcal{S}, d)$ be a complete metric space and $\Phi : \mathcal{S} \to \mathcal{S}$ a contraction mapping with Lipschitz constant $\lambda < 1$ and unique fixed point $s^*$. Let $\mathcal{A} \subset \mathcal{S}$ be the admissibility manifold, and suppose $s^* \in \mathcal{A}$. Then for any initial configuration $s_0 \in \mathcal{S}$, the iteration $s_{k+1} = \Phi(s_k)$ converges to $s^*$ with geometric rate $d(s_k, s^*) \leq \lambda^k d(s_0, s^*)$.*

The contraction assumption is strong and will not hold in general. The staged closure protocol handles the non-contractive case by decomposing $\Phi$ into a sequence of partial operators, each contractive on a restricted domain. The protocol proceeds in three stages. In the gauge repair stage, local inconsistencies are resolved by modifying the local section to satisfy local constraints. In the refinement stage, tile boundaries are adjusted to eliminate boundary incompatibilities that survived local repair. In the extension stage, configurations that cannot be made consistent within the current ontology trigger the ontological expansion procedure of the preceding section. Each stage is obstruction-aware: it tracks the cohomological structure of the incompatibilities it is resolving, ensuring that repair operations address genuine obstructions rather than superficial inconsistencies.

A thermodynamic interpretation of the convergence process is available through the framework of entropy-weighted field theories. An entropy-weighted scalar field $\phi : \mathcal{M} \to \mathbb{R}$ over the spacetime manifold $\mathcal{M}$ encodes the distribution of uncertainty across the domain. The gradient $\nabla\phi$ defines a preferred direction of information flow, and a vector field $V^\mu$ encodes the causal structure of the domain. The closure iteration can be interpreted as a process that decreases $\phi$ along the direction of $V^\mu$, driving the system from higher-entropy (less constrained) to lower-entropy (more constrained) configurations. This direction is irreversible in the same sense that entropy decrease is irreversible in isolated systems: a system that has converged to a fixed point $s^*$ cannot be "un-closed" without the introduction of new constraints that expand the feasible set.

# 7 The Enshittification of State: Platform Legibility and the Restriction of Composition

## 7.1 From AI Architecture to Computational Political Economy

The preceding sections have argued that inference systems fail when they lack mechanisms for enforcing global coherence across representations. We now argue that an isomorphic failure is present in the contemporary software ecosystem. In both cases, the architecture systematically denies access to the underlying state that would make coherent action possible—whether that action is the enforcement of a consistency condition by an inference system, or the enforcement of a user-level constraint by a human operator. This section introduces the concept of compositional erosion and formalizes it in categorical terms, establishing the structural parallel with non-identifiability in AI systems.

## 7.2 Hidden State and the Elimination of Verification

In the era of command-line interfaces and early graphical environments, the objects managed by a computer—files, directories, process states, configuration values—were directly accessible as structured data. A user who wished to inspect, transform, or recompose these objects could do so through standard tools. The relationship between the user's action and the resulting state change was transparent. Files had known locations, formats were documented, and the operations available on any object were determinate and composable.

The progressive enclosure of these objects within proprietary interfaces, locked formats, and managed environments has systematically eliminated these conditions. File extensions are hidden by default, making the distinction between format types invisible. Filesystem hierarchies are abstracted into opaque library structures. Contact information, calendar data, and document content are stored in proprietary databases accessible only through sanctioned application interfaces. The content of a PDF—a format that is in principle entirely transparent—is made functionally inaccessible to editing through form structures, DRM overlays, or compressed encodings that resist standard tools. The font metric is a parameter in the rendering pipeline that determines the visual appearance of all text on the device; hiding it removes from the user the ability to enforce a visual consistency condition that the system itself does not enforce.

The formal structure mirrors the problem of non-identifiability. Let $\sigma$ denote the true underlying state of some managed object and $\pi_{\text{app}} : \Sigma \to \mathcal{V}$ the projection from underlying states to the view presented by the application. If $\pi_{\text{app}}$ is not injective, then distinct underlying states $\sigma_1 \neq \sigma_2$ may produce identical interface representa-

tions. The user cannot distinguish between them from within the application and therefore cannot perform the constraint-resolving operations—comparison, selective modification, format conversion, cross-application transfer—that would require access to the full state $\sigma$.

### 7.3 Interfaces as Functors and the Loss of Faithfulness

The restriction of user access to computational state can be formalized in categorical terms. Let $\mathcal{C}$ denote a category whose objects are configurations of the underlying system state $\Sigma$ and whose morphisms represent valid transformations between configurations. Let $\mathcal{D}$ denote a category of interface-visible representations, whose objects are the views presented to the user and whose morphisms are the transformations permitted by the interface. An application interface is then a functor

$$F : \mathcal{C} \to \mathcal{D} \tag{13}$$

which maps underlying states to visible representations and underlying transformations to interface-level operations.

**Definition 9** (Faithful Interface Functor)**.** *A functor $F : \mathcal{C} \to \mathcal{D}$ is faithful if for any pair of objects $X, Y \in \mathcal{C}$, the induced map*

$$F_{X,Y} : \mathrm{Hom}_{\mathcal{C}}(X, Y) \to \mathrm{Hom}_{\mathcal{D}}(F(X), F(Y)) \tag{14}$$

*is injective.*

Faithfulness ensures that distinct transformations in the underlying state space remain distinguishable at the interface level. If $F$ is not faithful, then there exist distinct morphisms $f \neq g : X \to Y$ such that $F(f) = F(g)$. From the user's perspective, these transformations are indistinguishable even though they correspond to different operations on the underlying state. Two transformation sequences—one preserving structure and one corrupting it—may produce identical visible outputs. The interface collapses distinctions essential for reasoning about the system.

**Proposition 4** (Constraint Collapse Under Non-Faithful Interfaces)**.** *Let $F : \mathcal{C} \to \mathcal{D}$ be a non-faithful functor. Then there exist constraints $c$ on $\mathcal{C}$ that do not factor through $F$, and are therefore unenforceable at the level of $\mathcal{D}$.*

*Proof.* Since $F$ is not faithful, there exist $X, Y \in \mathcal{C}$ and distinct morphisms $f \neq g : X \to Y$ such that $F(f) = F(g)$. Define a constraint $c$ that distinguishes the effects of $f$ and $g$: $c \circ f = \top$ and $c \circ g = \bot$. If $c$ factored through $F$, then $c \circ f = c \circ g$ would follow from $F(f) = F(g)$, contradicting the definition of $c$. Therefore $c$ does not factor through $F$. □

This proposition formalizes the intuition that when an interface collapses distinctions between underlying states or transformations, it eliminates the user's ability to enforce constraints that depend on those distinctions. The profile image wrapped in a proprietary container that requires decryption and format conversion before the underlying bytes become accessible is a concrete instantiation: the user is placed in the position of an inference system that has been denied access to its own memory state.

### 7.4 The Political Economy of Opacity

The restriction of composability is not accidental. It is the product of specific incentive structures operating on platform designers, device manufacturers, and software companies. When users cannot export their data in a usable format, they cannot migrate to competing services. When file formats are opaque, third-party tools cannot interoperate, reducing competition. When configuration interfaces are removed, users cannot customize behaviour in ways that might reduce engagement metrics or redirect traffic outside the platform.

The formal structure of this dynamic parallels the analysis of ontological insufficiency developed in Section 6.3. In the platform context, the "ontology" is the set of actions available to the user. Initially, the ontology is rich: the user can export, transform, redirect, and recompose. As the platform reduces composability, the ontology shrinks. Constraints that the user could previously enforce—"my data should be readable by any tool I choose"—become unenforciable. The feasible set of user actions contracts not because new information has been acquired, but because the mechanisms for constraint enforcement have been deliberately removed.

The systematic construction of non-faithful, non-surjective interface functors is therefore an economic strategy. By collapsing distinctions in the underlying state space and restricting the availability of transformations, platform designers reduce the effective dimensionality of the user's action space. The functor $F$ becomes a bottleneck through which all interactions with the underlying state must pass, and the user's constraint space $\mathcal{C}_u$ becomes a strict subset of the full constraint space $\mathcal{C}$, not because the constraints do not exist, but because the interface does not permit their expression.

### 7.5 Composability as a Formal Condition of Epistemic Agency

Composability is not merely a convenience preference of technically sophisticated users. It is a formal condition of epistemic agency in any system that must maintain coherence across operations. A user who can treat outputs as inputs to further processing is a user who can enforce global consistency conditions across their own

data. They can check that an export matches the source, verify that a transformation preserved relevant structure, detect when two representations of the same object have diverged. These are all constraint-enforcement operations, and they require access to the underlying state.

The history of Unix philosophy—treating everything as a file, composing programs through pipes, representing configuration in human-readable text—can be reinterpreted in these terms. When a program writes to stdout and another reads from stdin, the composition is transparent because the intermediate state is inspectable, redirectable, and transformable. Nothing about the pipeline is hidden. The progressive retreat from this model—toward APIs that return opaque objects, formats that require licensed readers, interfaces that route interactions through managed cloud services—is a retreat from the formal conditions of composability. The byte stream is replaced by a session token; the file by a document identifier; the shell pipeline by a workflow automation service that executes sanctioned steps in a predetermined order. At each stage, the intermediate state becomes less accessible, and the conditions for user-enforced constraint resolution are further eroded.

# 8 Evaluation Metrics Under the Realizability Criterion

## 8.1 The Limits of Marginal Accuracy

Standard evaluation metrics for inference systems are defined at the level of individual outputs. Accuracy, perplexity, BLEU score, and related measures assess the quality of predictions on a per-instance basis. These metrics are inherently local: they evaluate the correspondence between predicted and ground-truth outputs for each query independently.

This locality obscures a fundamental failure mode. A system may achieve perfect marginal accuracy while producing outputs that are jointly unrealizable.

**Proposition 5** (Decoupling of Accuracy and Realizability)**.** *There exist inference systems that achieve perfect marginal accuracy on all queries while having zero realizability rate, i.e., for which*

$$\bigcap_i \pi_i^{-1}(o_i) = \emptyset \tag{15}$$

*for the set of outputs* $\{o_i\}$ *produced across queries.*

*Proof Sketch.* Construct a system that memorizes the correct answer to each individual query by associating each query with a distinct implicit world-state, chosen so that the states are mutually incompatible. Each output is correct in isolation

because it matches the ground truth for that query, but no single state generates all outputs simultaneously. The realizability condition fails by construction. □

This demonstrates that marginal accuracy is insufficient as a criterion for epistemic validity. A system that performs perfectly on every query in isolation may be incapable of maintaining a coherent world-model across queries. A stronger notion of evaluation is required.

### 8.2 Realizability Rate and Constraint Violation

**Definition 10** (Realizability Rate)**.** *Let $\mathcal{Q}$ be a set of query collections and let $\mathcal{O}(Q)$ denote the outputs produced for $Q \in \mathcal{Q}$. The realizability rate is*

$$R = \frac{1}{|\mathcal{Q}|} \sum_{Q \in \mathcal{Q}} \mathbf{1}\left[ \bigcap_{q_i \in Q} \pi_i^{-1}(o_i) \neq \emptyset \right]. \tag{16}$$

**Definition 11** (Constraint Violation Rate)**.** *Given a constraint set $\mathcal{C}$, the violation rate is*

$$V = \mathbb{E}_{o \sim system}\left[ \frac{1}{|\mathcal{C}|} \sum_{c \in \mathcal{C}} \mathbf{1}[c(o) = \bot] \right]. \tag{17}$$

These metrics capture global properties of the system's outputs invisible to marginal evaluation. A system with high marginal accuracy but low realizability rate is producing locally correct but globally inconsistent outputs. A system with low constraint violation rate is respecting the known structural constraints of the domain even in cases where the ground truth is unavailable.

### 8.3 Feasible Set Diameter as Epistemic Progress

A more refined measure of epistemic progress is given by the geometry of the feasible set $F(\mathcal{C})$. Rather than evaluating outputs directly, one evaluates the extent to which the system reduces uncertainty over the space of configurations.

**Definition 12** (Feasible Set Diameter)**.** *Let $d$ be a metric on $\mathcal{S}$. The diameter of the feasible set is*

$$\operatorname{diam}(F) = \sup_{s, s' \in F(\mathcal{C})} d(s, s'). \tag{18}$$

A system that enforces constraint closure should exhibit monotonic contraction: $\operatorname{diam}(F(\mathcal{C}_{t+1})) \leq \operatorname{diam}(F(\mathcal{C}_t))$. Reduction in diameter corresponds to increasing identifiability. A system that accumulates information without reducing $\operatorname{diam}(F)$ is not making epistemic progress in the relevant sense; it is expanding its record without narrowing its uncertainty.

### 8.4 Evaluation as Fixed-Point Convergence

Under the closure paradigm, evaluation reduces to assessing convergence properties of the consistency operator $\Phi$. Key quantities include the convergence time, measured as the number of iterations required to reach a fixed point; stability, measured as the sensitivity of the fixed point to perturbations in input; and uniqueness, assessed by whether multiple fixed points exist. A system that converges rapidly to a unique fixed point within $F(\mathcal{C})$ satisfies the strongest form of the realizability criterion. Systems that exhibit oscillations, multiple attractors, or divergence fail to enforce closure.

The transition from accuracy-based to realizability-based evaluation reflects the broader shift from prediction to reconstruction. Performance metrics assess how well a system imitates observed data. Coherence metrics assess whether the system maintains a consistent internal representation of the world. The former is sufficient for tasks in which outputs are consumed in isolation. The latter is necessary for any system that must act, reason, or accumulate knowledge over time. A system that performs well locally but fails globally is not merely imperfect; it is structurally misaligned with the task of maintaining a coherent model of the world.

## 9 The Necessity of Constraint Closure

### 9.1 State of the Argument

We are now in a position to state and prove the central claim of this work. The preceding sections have established the following results, each independently necessary for the main theorem. Section 2 showed that inference is an inverse problem and that non-identifiability is the canonical failure mode of underconstrained systems. Section 3 showed that accumulation without gluing produces presheaves with non-trivial cohomological obstructions, making global sections—and therefore joint realizability—unavailable. Section 4 showed that gauge mismatches are a structured form of non-identifiability that accumulation systems cannot distinguish from genuine contradiction. Section 5 showed that causal grounding is necessary but not sufficient: a memory trajectory may be causally coherent while terminating outside the feasible set. Section 6 showed that knowledge compounds as feasible-set contraction, and that fixed-point convergence under a consistency operator is the formal condition that enforces this contraction. Section 7 showed that the same structural failure appears in software platforms, where non-faithful interface functors eliminate the user's ability to enforce constraints on their own data. Section 8 showed that standard evaluation metrics are blind to all of these failure modes.

### 9.2 Main Theorem

**Theorem 3** (Necessity of Constraint Closure for Realizable Inference)**.** *Let $\Omega$ be a space of world-states, $\{\pi_i : \Omega \to \mathcal{O}_i\}$ a family of observation operators, and $\mathcal{C}$ a set of structural constraints. Let an inference system produce outputs $\{o_i\}$ in response to queries $\{q_i\}$. If the system does not enforce convergence to a fixed point $s^*$ of a consistency operator $\Phi : \mathcal{S} \to \mathcal{S}$ satisfying*

$$\Phi(s^*) = s^* \quad \textit{and} \quad s^* \in F(\mathcal{C}), \tag{19}$$

*then there exists a sequence of queries for which the induced outputs $\{o_i\}$ are not jointly realizable:*

$$\bigcap_i \pi_i^{-1}(o_i) = \emptyset. \tag{20}$$

*Proof by Reduction Through the Hierarchy.* The proof proceeds by establishing that each level of the inference hierarchy below closure admits a construction witnessing joint unrealizability.

*Step 1: Stateless prediction.* A stateless predictive system produces outputs $o_i$ independently for each query $q_i$ without reference to a shared underlying state. As shown in Section 1.1, the absence of a realizability constraint permits the existence of outputs that are individually plausible but jointly incompatible. Formally, the system selects $o_i \in \mathcal{O}_i$ from the marginal distribution $p(\cdot \mid q_i)$ without enforcing that $\{o_i\}$ lies in the image of the joint forward map $(\pi_1, \ldots, \pi_n) : \Omega \to \prod_i \mathcal{O}_i$. Therefore joint unrealizability is not precluded.

*Step 2: Accumulation without gluing.* An accumulation system maintains a persistent knowledge structure $K$ but, as established in Theorem 1, the state $K_t$ need not lie in $F(\mathcal{C}_t)$. The presheaf structure of accumulated knowledge may admit a non-trivial Čech cohomology class $[\alpha] \neq 0$ in $\check{H}^1(X, \mathcal{F})$, indicating the presence of a realizability obstruction. The existence of such an obstruction implies that no global section exists, and therefore that the accumulated outputs are not jointly realizable.

*Step 3: Memory without closure.* A memory-based system with causal grounding produces outputs as projections $o = \pi(h_n)$ of a state trajectory $\{h_t\}$. As shown in Section 5.1, causal grounding does not impose the requirement that $h_n \in F(\mathcal{C})$. The transition operators $\{T_t\}$ reduce local entropy at each step but do not guarantee that the terminal state satisfies all constraints not yet encountered during the trajectory. A trajectory that is locally entropic at every step may converge to a configuration outside $F(\mathcal{C})$.

*Step 4: Closure enforces realizability.* By the Closure Condition (Definition 3), if the system enforces convergence of the iteration $h_{k+1} = \Phi(h_k)$ to a fixed point $h^* \in F(\mathcal{C})$, then the output $o^* = \pi(h^*)$ is a projection of a configuration that satisfies

all constraints. Since $h^* \in F(\mathcal{C})$, the collection of outputs derived from $h^*$ under the observation operators $\{\pi_i\}$ satisfies $h^* \in \bigcap_i \pi_i^{-1}(o_i^*)$, so the outputs are jointly realizable.

The contrapositive of Step 4 combined with Steps 1–3 establishes the theorem: any system that does not enforce convergence to a fixed point in $F(\mathcal{C})$ falls into one of the first three categories, each of which admits a sequence of queries producing jointly unrealizable outputs. □

### 9.3 Corollaries and Structural Consequences

Several consequences follow immediately.

**Corollary 1** (Inevitability of Hallucination)**.** *In any inference system lacking constraint closure, hallucination in the strong sense—the production of jointly unrealizable outputs—is unavoidable even if marginal accuracy is high.*

**Corollary 2** (Insufficiency of Data Scaling)**.** *Increasing dataset size or model capacity does not eliminate unrealizability. The failure arises from the absence of a global consistency constraint rather than from insufficient information; adding more observations increases the set $\{o_i\}$ without enforcing that the set is jointly realizable.*

**Corollary 3** (Necessity of State Accessibility)**.** *Any system in which the underlying state is inaccessible—whether due to architectural design or interface restriction—cannot guarantee constraint closure, and therefore cannot guarantee realizability. This applies equally to AI inference systems and to software platforms with non-faithful interface functors.*

Corollary 3 unifies the technical and political-economic dimensions of the argument. The failure modes observed in language models, knowledge bases, and software platforms are not independent phenomena. They are consequences of a shared structural deficiency: the substitution of a projection for the underlying state, and the consequent elimination of the conditions for constraint resolution.

## 10 Conclusion: Towards Realizable Intelligence

### 10.1 The Unified Argument

The argument of this essay has been unified by a single structural insight: global coherence requires enforcement, not merely accumulation. This insight applies across the entire spectrum of knowledge-producing systems, from AI inference engines to software platforms, and its neglect produces analogous failure modes at every level. A language model that generates plausible text without enforcing realizability, a

knowledge base that accumulates records without enforcing joint satisfiability, and a software platform that presents rich interfaces while hiding the underlying state are all instances of the same failure: the substitution of a projection for the thing itself.

The positive counterpart of this claim is the closure condition developed formally in the preceding sections, understood as a general architectural principle. A system satisfies the closure condition when: (a) the underlying state is causally grounded in a traceable trajectory; (b) the trajectory is governed by a consistency operator that enforces satisfaction of all active constraints; (c) convergence to a fixed point within the feasible set is guaranteed under the operator; and (d) the intermediate states of the trajectory are accessible for inspection and intervention. Condition (d) is the composability requirement: without access to intermediate states, conditions (a)–(c) cannot be verified even in principle.

### 10.2 The Realizability Turn

The transition from predictive AI to constraint-closing AI, and from closed platforms to composable infrastructure, represents a single shift that can be called the realizability turn: a move from systems that produce outputs to systems that maintain states. In prediction-centric systems, the output is the end point; no further question is asked about whether it corresponds to a realizable underlying configuration. In state-maintenance systems, the output is a projection of a configuration subject to ongoing consistency enforcement. The configuration is primary; the output is secondary.

This shift has practical implications at every level of the stack. At the level of model architecture, it requires the replacement of next-token prediction with trajectory-based state evolution in which each step is constrained to respect the accumulated evidence and the inferred structure of the domain. At the level of knowledge organization, it requires the replacement of append-only accumulation with iterative constraint solving, in which new information triggers a re-examination of existing representations to enforce joint satisfiability. At the level of the software ecosystem, it requires the preservation of format transparency, directory accessibility, and tool composability as preconditions for user-level constraint enforcement.

### 10.3 Realizability as the Criterion

The appropriate criterion for evaluating inference and knowledge systems is realizability: the question of whether the outputs produced by the system could, in principle, all be projections of a single underlying configuration that satisfies all relevant constraints. This criterion is global where the standard ones are local, structural where they are statistical, and permanent where they are episodic.

Realizability is a demanding criterion. It may not be achievable in finite time for complex domains, and it may require the ongoing refinement of constraints as evidence accumulates. But the failure to impose it—to treat local accuracy as sufficient—is the source of the systematic failures documented throughout this essay. The formal architecture developed here—sheaf-theoretic constraint representation, cohomological obstruction diagnosis, trajectory-aware coarse-graining, and staged closure protocols—constitutes one implementation of the realizability criterion, developed more fully in the companion TARTAN framework. These contributions are not exhaustive; they are a beginning.

The deeper claim is that knowledge is not a collection of statements, a network of documents, or an archive of past queries and responses. Knowledge is a configuration of the world that survives all available constraints simultaneously—a single, globally realizable state that is the unique fixed point of the consistency operator applied to all evidence. The task of intelligence, artificial or human, is not to accumulate representations of that state but to converge to it.

**Acknowledgements.** The author thanks the traditions of formal epistemology, mathematical physics, and constructive logic for the conceptual tools deployed here.

## A Table of Structural Isomorphisms

The following table makes explicit the conceptual correspondences that run through the preceding argument, mapping core notions across three domains: sheaf theory, AI inference architecture, and platform design. Each row identifies a structural concept and its realization in each domain. The table is intended to be read as an argument: the isomorphism between columns is not metaphorical but formal, in the sense that the same mathematical deficiency—failure of a presheaf to satisfy the gluing condition, failure of an inference system to enforce constraint closure, failure of a platform interface to be a faithful functor—produces the same structural failure in each domain.

| **Concept** | Sheaf Theory | AI Inference | Platform Design |
|---|---|---|---|
| **Base space** | Domain space $X$ | Query space $\{q_i\}$ | User action space |
| **Local section** | $s_i \in \mathcal{F}(U_i)$ | Per-query output $o_i$ | Interface view $v \in \mathcal{V}$ |
| **Gluing condition** | Global section exists | Joint realizability | Composable operations |
| **Obstruction** | $[\alpha] \in \check{H}^1(X, \mathcal{F})$ | $\bigcap_i \pi_i^{-1}(o_i) = \emptyset$ | Non-faithful functor $F$ |
| **Failure mode** | Presheaf not a sheaf | Hallucination | Compositional erosion |
| **Consistency operator** | Sheafification $\mathcal{F} \to \mathcal{F}^+$ | $\Phi : \mathcal{S} \to \mathcal{S}$ | Composable toolchain |
| **Fixed point** | Global section $s \in \mathcal{F}(X)$ | $s^* \in F(\mathcal{C})$ | Verifiable user state |
| **Gauge freedom** | Local coordinate choice | Representation ambiguity | Format equivalence |
| **Gauge fixing** | Canonical rep. selection | Canonicalization step | Open standards |
| **Realizability** | $\check{H}^1 = 0$ | $F(\mathcal{C}) \neq \emptyset$ singleton | Faithful $F$, surjective $F$ |
| **Progress measure** | Obstruction reduction | $\mathrm{diam}(F(\mathcal{C})) \downarrow$ | Reachability set $\uparrow$ |

The isomorphism in the "Failure mode" row is the central claim of the essay: presheaf non-sheafness, hallucination, and compositional erosion are the same structural failure at different levels of description. The isomorphism in the "Progress measure" row articulates the correct metric for each domain: decreasing cohomological obstruction, decreasing feasible-set diameter, and increasing user reachability are all contractions of the same underlying feasibility space.

## References


[1] Banach, S. (1922). Sur les opérations dans les ensembles abstraits et leur application aux équations intégrales. *Fundamenta Mathematicae*, 3, 133–181.

[2] Bredon, G.E. (1997). *Sheaf Theory*. Springer, New York.

[3] Bush, V. (1945). As we may think. *The Atlantic Monthly*, 176(1), 101–108.

[4] Doctorow, C. (2023). The enshittification of TikTok. *Wired*.

[5] Ghrist, R. (2014). *Elementary Applied Topology*. Createspace.

[6] Grothendieck, A. (1957). Sur quelques points d'algèbre homologique. *Tohoku Mathematical Journal*, 9(2), 119–221.

[7] Jaynes, E.T. (2003). *Probability Theory: The Logic of Science*. Cambridge University Press.

[8] Kolmogorov, A.N. (1965). Three approaches to the quantitative definition of information. *Problems of Information Transmission*, 1(1), 1–7.

[9] MacKay, D.J.C. (2003). *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press.

[10] Mac Lane, S. & Moerdijk, I. (1992). *Sheaves in Geometry and Logic: A First Introduction to Topos Theory*. Springer, New York.

[11] Parisi, G. (1988). *Statistical Field Theory*. Addison-Wesley, Redwood City.

[12] Raymond, E.S. (2003). *The Art of Unix Programming*. Addison-Wesley.

[13] Robinson, M. (2014). *Topological Signal Processing*. Springer, Berlin.

[14] Serre, J.-P. (1955). Faisceaux algébriques cohérents. *Annals of Mathematics*, 61(2), 197–278.

[15] Stallman, R.M. (2002). *Free Software, Free Society: Selected Essays of Richard M. Stallman*. GNU Press.

[16] Tikhonov, A.N. & Arsenin, V.Y. (1977). *Solutions of Ill-Posed Problems*. Winston & Sons, Washington D.C.

[17] Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., Chi, E., Le, Q.V., & Zhou, D. (2022). Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35, 24824–24837.

[18] Wissner-Gross, A.D. & Freer, C.E. (2013). Causal entropic forces. *Physical Review Letters*, 110(16), 168702.

[19] Zuboff, S. (2019). *The Age of Surveillance Capitalism: The Fight for a Human Future at the New Frontier of Power*. PublicAffairs, New York.