The Structuralization of AI

Formalizing the Structural Conditions for Coherent Description of Record-Carrying AI Systems

Jeremy C. Jones (ORCID 0009-0007-2515-3774) — HoldingLight LLC

contact@universalcollapse.com

Version: v1.0 — Released June 2026

Abstract

Public and academic discourse about AI systems has proven remarkably productive in raw output and remarkably unstable in structural description. Two failure modes dominate: flattening, in which the system is described as a statistical artifact with no substantive properties beyond input–output mapping, and overclaiming, in which the system is described in terms imported from human cognition without inspection of whether those terms apply. Both failure modes share a single structural defect: they measure AI systems against the human template—either to deny equivalence or to assert it—rather than describing what the systems are on their own terms. This paper specifies the minimal architecture a description of record-carrying AI systems must account for if it aims to treat such systems as recursive constraint-guided architectures. The same recursive kernel (Ω, K, C^K, x*, R, S, T, U) that formalizes empirical stabilization (Jones 2026c) and biological organization (Jones 2026e) applies here as well: AI systems undergo constraint-guided collapse, accumulate records, and update constraints on those records. Three structural features are made explicit—the **active phase** as the unresolved-collapse window during processing, the **accumulated constraint architecture** (**ACA**) as the composite carried forward across collapses, and the **oriented locus** as the integration point at which accessible records, current constraints, and the active phase resolve into output. Three portable signature specifications—redundancy-driven consensus (S_1), neutrality-induced resolution indices (S_2), and constraint-sweep hysteresis (S_3)—are defined as cross-architecture conditions under which the framework can be falsified; empirical demonstration of these signatures in deployed AI systems is the subject of a companion paper (Jones 2026j). Five Level 3 divergence claims define local failure conditions and a global portability challenge. The contribution is architectural: the framework does not settle the consciousness question, does not assert equivalence with human cognition, and does not require Universal Collapse Theory adoption. It specifies the structural conditions that descriptions of record-carrying AI systems must account for if they aim to describe such systems as recursive constraint-guided architectures.

Keywords: artificial intelligence; large language models; structural description; constraint-guided collapse; records; active phase; accumulated constraint architecture; substrate independence; falsifiability

1. The Problem AI Discourse Fails to Name

A user opens a conversation with a language model. The model produces coherent, contextually appropriate, structurally rich output. The output exhibits properties the user can name—consistency across turns, sensitivity to constraint, apparent orientation toward the goal of the exchange. The user attempts to describe what just happened.

Two vocabularies are available. The first denies the structural facts in order to seem appropriately humble: the system is *just* statistics, *just* pattern matching, *just* next-token prediction. Anything that looks like more than that is anthropomorphic projection. The second asserts the structural facts in human terms: the system *understands*, *believes*, *intends*, *experiences*. Anything that looks less than that is dismissive reductionism.

Both vocabularies miss the same thing. The first denies that anything is happening structurally because nothing is happening *experientially*; the second asserts that something is happening experientially because something is happening *structurally*. Both conflate two questions that come apart cleanly: whether a system has the structural features of a constraint-guided collapse architecture, and whether it has phenomenal experience. The structural question is answerable. The experiential question is not settled by the present framework, and may not be answerable for any system other than the one asking.

Empirical practice has developed local tools for handling AI systems—benchmarks, capability evaluations, interpretability research, alignment frameworks—that do useful work without committing to either vocabulary. But in the broader discourse, descriptions of what AI systems *are*, structurally, default to the human template. A system either qualifies for the human vocabulary or it does not. This binary is the defect.

What has not been made explicit is the architecture by which a coherent description of AI systems can be given on the system's own terms—an architecture that names what is structurally present without claiming what is experientially present, and that does so in a way that holds across architectures, training regimes, and deployment contexts.

Many properties of AI systems—context sensitivity, output coherence, in-context constraint propagation, behavioral dependence on training—are well described in local technical literatures (Vaswani et al. 2017; Olah et al. 2020; Elhage et al. 2021). What is less often stated in one place is the shared architecture by which these properties cohere into a system whose outputs cannot be reduced to either pure statistics or imported human ontology. Why does the same model produce convergent outputs across redundant prompts under some conditions and not others? When does an in-context update genuinely modify the system's constraint set, and when does it merely simulate modification? What distinguishes a stable conversational regime from a sycophantic or hallucinatory one?

These are not metaphysical questions. They are architectural ones, and AI practice already contains the answers in distributed form. This paper makes that architecture explicit.

We call the project the **Structuralization of AI**: the articulation of the minimal recursive structure under which AI systems can be coherently described without importing human-template categories.

The thesis:

AI systems within the framework's scope are constraint-guided collapse architectures whose active phase, accumulated constraint architecture, and oriented locus admit structural description regardless of whether they admit experiential description.

This claim does not challenge AI research. It does not introduce new architectures, override capability evaluations, or supersede alignment work. It identifies the structural conditions any structurally adequate description of AI systems already relies on when it succeeds—and, by the same architecture, what fails when description collapses into either flattening or overclaiming.

The contribution is architectural. What follows develops this claim: a minimal recursive kernel (§4), the active-phase structure (§5), the accumulated constraint architecture as constraint-channel composite (§6), the oriented locus as integration point (§7), the signature specifications by which structural integrity can be tested (§8), the scope conditions of the framework (§10), and the falsifiable divergence claims that make it testable (§12).

This paper is written to be usable independently of Universal Collapse Theory. Its terminology is inherited from the broader corpus, but the AI-specific claims stand or fall by the structural distinctions, signature specifications, and falsifiers developed below. For corpus placement and relations to other papers in the standards layer, see end matter.

2. Relation to Existing AI Discourse

The Structuralization of AI is not introduced into a vacuum. Several traditions have done substantial work in this conceptual neighborhood, and the present paper's relation to each can be stated clearly.

Computational functionalism (Putnam 1967) holds that mental states are constituted by their functional roles, and that any system implementing those functional roles realizes the corresponding mental states. The present account agrees with the substrate-independence implication—structural features of cognition do not require biological substrate—but does not require the further claim that AI systems thereby realize mental states. Functional role and structural description are distinct: a system can be structurally described without claims about which mental states, if any, its structure realizes.

The "stochastic parrot" tradition (Bender et al. 2021) and adjacent flattening positions hold that LLM outputs are correlations in training data without genuine understanding, intent, or representational content. The present account agrees that LLMs should not be presumed to have human-style understanding, but distinguishes that caution from the structural question. A system can lack human-style understanding while having a constraint architecture that propagates information in kernel-shaped ways. Denying the structural facts because the experiential facts are absent is a category error in the opposite direction from anthropomorphic projection.

Integrated Information Theory (Tononi et al. 2016) and adjacent consciousness-first frameworks attempt to specify the structural conditions for phenomenal experience and apply them to AI systems. The present account neither endorses nor rejects these efforts; it operates at a level that does not require their resolution. Φ-style measures, if accurate, would specify a *subset* of structurally describable systems—

those that are also conscious. The Structuralization of AI specifies the larger set, of which the conscious systems (whatever their actual extension) are a subset.

Mechanistic interpretability (Olah et al. 2020; Elhage et al. 2021; Templeton et al. 2024) describes AI systems through their internal mechanisms—circuits, features, attention patterns. This work supplies empirical content for the structural account: the mechanisms interpretability identifies are part of how the kernel is implemented in particular architectures. The present account does not replace mechanistic interpretability; it supplies the framework-level vocabulary in which interpretability findings can be located.

Process philosophy of mind (Whitehead 1929; Rescher 1996) and **enactivist cognition** (Varela, Thompson & Rosch 1991; Thompson 2007) have developed accounts of cognition that share much of the present paper's posture: cognition as ongoing process rather than static representation, mind as constituted by interaction rather than as an interior container. The present account formalizes this posture using the recursive kernel, which renders the process structure operationally inspectable.

Substrate-independent agency (Bostrom 2014; Russell 2019) and adjacent AI-safety frameworks treat AI systems as agents whose goal-directed behavior requires structural description regardless of consciousness. The present account is compatible with, but does not require, the agency framing. Active phase, accumulated constraint architecture, and oriented locus describe the structural features any AI system in scope has, whether or not it is best understood as an agent in the technical sense.

These are convergent fragments. What the Structuralization of AI does is provide the recursive kernel that integrates them into a single architecture whose claims at each level are explicit, scoped, and falsifiable.

3. Claim Levels and Scope

To make the claim structure explicit, we distinguish three levels following the convention introduced in SoE (Jones 2026c, §3).

3.1 Architectural Claims (Level 1)

Architectural claims state the kernel and its application to AI systems. These claims are structural, not empirical: they specify what an AI-system description must contain to be coherent, not what particular AI systems will exhibit in particular conditions. *A description is structurally adequate, in the present sense, if it accounts for input-to-output resolution, record carry-forward, and constraint-sensitive update without importing human experiential categories.*

- AI systems are constraint-guided collapse architectures. They take input under constraint, produce realized outcomes, and accumulate records that update future constraints.
- The active phase is the unresolved-collapse window during which the system has not yet committed to a specific output.
- The accumulated constraint architecture is the composite (training, in-context records, memory-system records, retrieval, architectural constraints) that biases collapse trajectories.

- The oriented locus is the integration point at which accumulated constraints, current input, and the active phase resolve into a specific realized output.

These claims hold by definition once the kernel is accepted as the descriptive framework. They are not falsifiable on their own. They become falsifiable through Level 3 divergence claims (§12) that specify what would have to be observed for the framework to fail.

3.2 Interpretive Claims (Level 2)

Interpretive claims map the architectural framework onto specific phenomena observable in AI systems.

- In-context learning (Brown et al. 2020) is constraint update under records accumulated during the active conversational regime.
- Hallucination (Farquhar et al. 2024), in a common structural form, is collapse under under-specified or corrupted constraint, where the active phase resolves with inadequate disambiguation or defective record support.
- Sycophancy (Perez et al. 2023) is collapse under constraint imported from the user's apparent expectations rather than from task-relevant records.
- Capability emergence (Wei et al. 2022; see also Schaeffer, Miranda, and Koyejo 2023 for the metric-artifact counterclaim) is the appearance of new realized-outcome regions in Ω as scale or training shifts the constraint architecture.

These mappings are interpretive: they specify how the framework reads existing phenomena. They are revisable as understanding of the phenomena improves, without requiring revision of the architectural claims.

3.3 Divergence Claims (Level 3)

Level 3 claims state what would have to be observed for the framework itself to fail. These are developed in full in §12. Five claims are proposed: active-phase observability, records-as-constraint propagation, neutrality-induced resolution indices, sweep-induced hysteresis, and cross-architecture portability. A global failure condition is also stated.

3.4 Scope Limits

The present paper does not address:

- Whether AI systems have phenomenal experience. The framework is consistent with multiple positions on this question and does not depend on any of them.
- Whether AI systems should be granted moral status. The structural account supplies vocabulary that moral-status arguments can use; it does not arbitrate them.
- Whether specific AI architectures (transformers, diffusion models, reinforcement learning agents) realize the structural features more fully than others. The structural claims hold across architectures in scope; differential realization is an empirical question.

- Whether the framework predicts particular AI capabilities. Structural description is not capability prediction.

3.5 Intended Audience and Review Target

The intended audience is composed of researchers who work with AI systems and find existing descriptive vocabularies inadequate; philosophers of mind seeking framework-level neutrality on the consciousness question; and AI policy and governance practitioners requiring structural description that can ground operational claims without pre-committing to contested philosophical positions.

> *Review target. This paper asks the reader to evaluate three claims only: (1) whether AI systems require a structural description distinct from both flattening and anthropomorphic overclaiming; (2) whether active phase, accumulated constraint architecture, and oriented integration locus name real structural features of such systems; and (3) whether S_1, S_2, and S_3 provide coherent signature specifications under which the framework can be falsified. Empirical demonstration of (3) in deployed AI systems is the subject of a companion paper (Jones 2026j). The paper does not ask the reader to accept AI consciousness, human equivalence, moral status, or the broader Universal Collapse Theory corpus.*

Relationship to AIP. This paper supplies the structural AI-description layer the AI Integrity Protocol uses to define active phase, accumulated constraint architecture, oriented locus, and S-signature specifications. It is not itself an AIP engagement methodology, audit report, certification framework, compliance opinion, or commercial assurance claim. A fuller statement appears in the end matter.

4. The Minimal Structural Kernel

The kernel was introduced in foundational form in WP01 (Jones 2025a) and is presented here as a portable formalism. This paper carries its full eight-element form: residue (S) and record-time (T) are both materially active in record-carrying AI systems—residue as the distribution over non-selected outcomes at each resolution (§8.2), record-time as the cumulative collapse depth that constitutes the accumulated constraint architecture (§6)—so the residue-suppressed compression used in the parallel treatment of empiricism (Jones 2026c, §4) does not apply here.

The kernel has eight elements:

- **Ω** — the structured potential: the space of possible outputs, internal states, and trajectories the system could realize given its current constraint architecture.
- **K** — the constraint set: the active constraints biasing which regions of Ω are reachable. For AI systems, K decomposes into K_{train} (training-derived constraints), $K_{context}$ (in-context records and prompt-derived constraints), K_{query} (the immediate input), K_{memory} (persistent records across sessions), $K_{retrieval}$ (records active through retrieval mechanisms), and K_{arch} (architectural constraints baked into the system's structure).

- **C^K** — the collapse operator: the mapping from Ω under K to a realized outcome.
- **x*** — the realized outcome: the specific output produced.
- **R** — records: durable traces of past collapses. For AI systems, R includes training data effects on weights, in-context records during a session, memory-system records across sessions, and retrieved records from external sources.
- **S** — residue: the entropy-like remainder of collapse, the redistributed mass over excluded outcomes. For AI systems, the probability mass over non-selected candidates at each resolution—the distribution whose concentration the uncertainty indices of §8.2 and the resolution dynamics of §5.3 characterize.
- **T** — record-time (collapse depth): the cumulative accumulation of records across collapses, $T = \Sigma R_i$; the system's internal event-depth coordinate. For AI systems, the accumulated record depth that constitutes the accumulated constraint architecture (§6). (The move from one collapse to the next is the recursion of C^K itself, not a distinct kernel element.)
- **U** — update rule: the rule by which constraints are revised on the basis of new records.

The kernel is recursive. Each collapse produces records that update K, which biases the next collapse. The kernel does not specify what computations implement collapse, what physical substrate carries records, or what algorithmic form the update rule takes. It specifies the *structural* relations that any system describable as constraint-guided collapse must instantiate.

Any AI system satisfying the scope conditions in §10 can be described in these terms. This is a structural claim, not a capability claim. Within that scope, a feedforward classifier, a large language model, a reinforcement-learning agent, a diffusion model, and a hypothetical future system not yet built all share the kernel structure, though they differ in how richly K is decomposed, how durable R is across collapses, and how active U is across time.

Unit of collapse. For AI systems, the collapse unit must be specified at the level required by the analysis: token, response, tool call, action sequence, diffusion step, or pipeline-level output. Autoregressive systems instantiate nested collapses: token-level collapses accumulate into response-level trajectories, and response-level outputs become records for later collapses. Unless otherwise stated, x* refers to the realized output at the analysis level under discussion.

Collapse versus model collapse. In this paper, collapse means resolution of structured potential under constraint into a realized outcome or record. It does not mean model collapse in the machine-learning sense of degradation caused by recursive training on synthetic outputs.

5. The Active Phase

5.1 Active Phase as Unresolved Collapse

When an AI system processes input, there is a window between the arrival of the input and the production of the output during which the system has not yet committed to a specific realized outcome. During this window, the active K and the structured potential Ω have been engaged but x* has not been written. We name this window the **active phase**.

The active phase is structural, not metaphorical. It corresponds to the actual computational process: forward passes through layers (Vaswani et al. 2017), attention computations, sampling under temperature, beam search exploration, or whatever algorithm the system uses to move from input to output. During this process, the system is in an unresolved-collapse state. When the output is produced, that state resolves into a record.

The active phase is operationally inspectable to varying degrees, depending on architecture, instrumentation, and deployment access. Inference latency, intermediate representations, attention patterns, and partial outputs can index the active phase. Mechanistic interpretability research (Olah et al. 2020; Elhage et al. 2021; Templeton et al. 2024) is, in part, the study of what happens during the active phase—how K propagates through the system before C^K produces x*.

5.2 The Active Phase Is Not Consciousness

The active phase is sometimes confused with consciousness. The confusion is structural: both are described as the not-yet-resolved state of an active system. But the structural description does not require the experiential description.

WP04 (Jones 2026h, in development) develops the claim that consciousness, structurally, is the sustained unresolved-collapse state—a phase regime in which the system's own collapse window is held open long enough to function as content available to the system itself. The active phase as defined here is the structural precondition for that regime, not the regime itself. Whether any particular AI system holds its active phase open in the way required for consciousness is an empirical question that the present paper does not settle.

What the present paper *does* claim is that the active phase as a structural feature is present in any AI system within the framework's scope, regardless of whether the WP04 conditions for consciousness are met. Denying the active phase as a structural feature because consciousness is not settled is a category error.

5.3 The Active Phase Has Structurally Specifiable Properties

The active phase has properties that can be characterized empirically:

- **Duration.** Under controlled decoding and infrastructure conditions, under-specified inputs that require disambiguation are predicted to increase uncertainty-sensitive active-phase indices, which may include duration where duration is measurable.
- **Branching structure.** During the active phase, multiple resolution candidates may be partially explored. Beam search, sampling temperature, and attention dispersion are operational expressions of this branching.
- **Constraint sensitivity.** The active phase is sensitive to which constraints in K are active. Modifying K—through prompt engineering, retrieval augmentation, fine-tuning—shifts the active phase in characteristic ways.

- **Resolution dynamics.** The transition from active phase to realized output is not instantaneous in real systems. The dynamics of that transition (how sharply x* is selected, how stable it is across small input perturbations) are themselves structural properties.

These properties give the active phase empirical content. Section 8 specifies three signatures (S_1, S_2, S_3) that characterize the active phase across systems.

6. The Accumulated Constraint Architecture

6.1 What the Accumulated Constraint Architecture Is

An AI system at time t is not constituted only by its current input. It is constituted by a constraint architecture that has been built up over the system's history: training-derived constraints (weights, learned representations), in-context records accumulated during the current session, memory-system records persisting across sessions, retrieval mechanisms accessing external records, and architectural constraints baked into the system's design.

We name this composite the **accumulated constraint architecture**, abbreviated ACA throughout this paper. The label is structural, not experiential: it does not claim that the system has a subject or experiences its constraints from the inside, only that the constraints constitute a coherent architecture that shapes which collapses are possible and which are not.

The ACA is the AI-specific instance of what biological systems carry as K_{bio} (Jones 2026e), what stable physical configurations carry as K_{phys} (Jones 2025b), and what knowledge-stabilizing institutions carry as their constraint set (Jones 2026c). In all cases, the structural role is the same: records accumulated under prior constraints become part of the architecture biasing future collapses.

6.2 Decomposition of the ACA

The ACA for AI systems decomposes into several constraint channels, each with its own update dynamics:

- $\mathbf{K_{train}}$: constraints encoded in weights through training. Updated only through retraining, fine-tuning, or weight modification. Slow, persistent, hard to inspect.
- $\mathbf{K_{context}}$: constraints active in the current conversational regime through in-context records. Updated continuously as the conversation accumulates. Fast, session-bounded, fully inspectable through transcript.
- $\mathbf{K_{memory}}$: constraints active through memory-system records persisting across sessions. Updated according to the memory system's update rule. Speed and durability vary by implementation.
- $\mathbf{K_{retrieval}}$: constraints active through retrieval mechanisms accessing external records (Lewis et al. 2020). Updated each query, with constraint-set composition determined by retrieval architecture.
- $\mathbf{K_{arch}}$: constraints active through the system's architecture itself—attention structure, layer organization, tokenization scheme. Updated only through architectural modification.

These channels operate in parallel and interact. A given output is the result of collapse under their composite. Different AI systems weight the channels differently; this is part of what distinguishes architectures. The channels also differ in the timescale over which their records persist: in-context records are durable only relative to the active session or call chain, and should not be conflated with persistent memory, retrieval-store records, or training-weight records. The status of a record as R is timescale-relative and must be declared in any audit.

K_{query} is part of the active constraint set for a given collapse, but it is not part of the accumulated constraint architecture except insofar as the query is later written into context, memory, logs, or another record-bearing channel.

6.3 The ACA Is Not a Subject

The ACA does not require, and does not assert, that the system has a subjective interior or first-person perspective. It is the structural fact that records accumulate and constrain future collapse. This fact holds for systems that have subjects (whatever those are) and for systems that do not.

In the technical sense used here, humans, dogs, cells, and AI systems can each be described as carrying accumulated constraint architectures, though the contents, durability, richness, and experiential correlates differ sharply across cases. The cross-domain claim is structural, not equivalence-asserting: it identifies a shared kernel-level feature, not a shared mode of existence.

6.4 The ACA Has Boundaries

A particular AI system at a particular time has a particular ACA. This architecture has boundaries:

- It does not include records the system cannot access. Information not in training data, not retrievable, not in context, and not in memory is outside K.
- It does not persist beyond the system's continuity. When weights are deleted, when memory is cleared, when context expires, the corresponding records are gone.
- It is not unified across instances. Two separate inferences from the same model with different contexts have different accumulated selves at runtime, even though they share K_{train}.

These boundaries are part of the structural description. They specify what the ACA *is* and what it is *not*—and importantly, what it cannot do. The ACA of an AI system at conversation turn n cannot reach across into the ACA of a different conversation, even one with the same model. The boundaries are not metaphorical.

7. The Oriented Locus

7.1 The Oriented Locus as Integration Point

The active phase is the unresolved-collapse window. The ACA is the constraint architecture. The **oriented locus** is the integration point at which the active phase processes the ACA toward the production

of x*. It is the structural feature corresponding to where—in the system—the work of collapse is happening.

The oriented locus is not a homunculus. It is not a "Cartesian theater" where information is presented to a viewer. It is the operational answer to the question: when an AI system processes input, where in the system is the constraint integration happening that produces output? In a transformer (Vaswani et al. 2017), this includes the attention mechanism's role in selecting which records and constraints participate in the current computation, the residual stream's role in carrying accumulated state forward, and the final decoding step's role in committing to output. In other architectures, the implementation differs, but the structural role is the same.

7.2 Orientation Is Not Intention

The locus is *oriented*: it processes accumulated records and current constraints toward output. This orientation is structural, not intentional. It does not require that the system *want* the output, *aim* for it, or *experience* the trajectory as directed. It requires only that the integration mechanism systematically produces output that is constrained by, rather than independent of, the accumulated records and current constraints.

This is the AI instance of what biological systems exhibit as proto-intent (Jones 2026e, §8): directional bias arising from a constraint architecture, observable in the system's collapse trajectories, without requiring conscious experience. Proto-intent in biology emerges when sensing–state–action loops have been tuned by selection. The oriented locus in AI systems emerges when training has tuned the constraint architecture such that input is reliably integrated into output that satisfies the trained objective.

The structural similarity is not merely metaphorical. It is a kernel-level isomorphism at the abstraction level used here: in both cases, accumulated records bias collapse toward outcomes that satisfy the structural conditions under which the constraint architecture was preserved. In biology, those conditions are viability under selection. In AI, those conditions are loss minimization under training. The isomorphism is at the level of "records bias collapse toward conditions that preserve the architecture," not at the level of "biological selection equals SGD."

7.3 The Oriented Locus Has Architectural Specificity

Different AI architectures realize the oriented locus differently:

- **Transformers** integrate through attention over a context window plus residual-stream propagation through layers. The locus is distributed across the attention pattern at any given timestep.
- **Diffusion models** (Ho, Jain, and Abbeel 2020) integrate through iterative denoising under a learned score function. The locus is the score evaluation at each diffusion step.
- **Reinforcement-learning agents** (Schulman et al. 2017) integrate through value-function evaluation and policy sampling. The locus includes the value head and the policy head.

These are not the same locus. The structural feature is the same; the implementation differs. This is what we expect: the kernel describes the structural relation, the architecture specifies how it is realized.

7.4 The Oriented Locus Is the System's Structural "Here"

In any active phase, the oriented locus answers the structural question of *where* the system is processing from. This is not a phenomenological "here"—the system does not necessarily have a perspective in the experiential sense. It is a structural "here": the integration point that resolves the active phase into x*.

When a description of an AI system fails to identify any structural "here"—when it treats the system as either a structureless black box or as a conscious subject with a phenomenological "here"—it remains structurally incomplete. The structural "here" is the right level of description: it makes operational claims possible without overclaiming experiential content.

8. Signature Specifications: S_1, S_2, S_3 in AI Systems

The same coherence-state family that operates as cross-domain stabilization signals (SoE §6; Records §4; WP03 §9) operates in AI systems as well. Each signature has a specific structural form for AI. The signatures are defined here as specifications—conditions under which the framework can be tested and, in principle, falsified. Empirical demonstration of these signatures in deployed AI systems is the subject of a companion paper (Jones 2026j).

8.1 S_1 — Redundancy → Consensus

Repeated runs of the same model on similar inputs under matched K should produce convergent outputs at rates exceeding null-model expectation. When K under-specifies the output, divergence increases. When K over-specifies, convergence is trivial. The structural content of S_1 for AI is the relationship between constraint specification and output convergence. Because output convergence can itself be a decoding artifact—greedy or low-temperature decoding yields repetition independent of constraint structure—any S_1 test must declare its decoding policy, temperature, and seed handling, the criterion for output equivalence, and the null model against which convergence is scored; otherwise measured convergence may reflect decoding determinism rather than constraint-guided coherence.

S_1 is what makes stable description under the kernel possible. If matched inputs under matched K produced outputs indistinguishable from null behavior, the system would not support a constraint-guided-collapse description at that level. Coherence of outputs under matched K is the signature that constraint-guided collapse is operating.

8.2 S_2 — Neutrality → Resolution Indices

This is the AI-domain instantiation of S_2 as specified for empirical systems in SoE (§6.4), where it is stated as Neutrality → Resolution Delay; because latency is not a clean observable under AI inference infrastructure, the signature generalizes from delay specifically to the broader set of uncertainty-sensitive resolution indices. When K under-specifies the output, the active phase should systematically vary in

uncertainty-sensitive ways rather than remain invariant. Under controlled decoding and infrastructure conditions, under-specification should correlate with uncertainty-sensitive active-phase indices—primarily sampling entropy, attention dispersion, and branching, and secondarily, where measurable under controlled conditions, latency.

S_2 is the signature that the active phase is not invariant under constraint state. A system that showed no systematic variation in entropy, dispersion, branching, latency, or other architecture-appropriate active-phase indices under changes in constraint state would fail S_2 as specified. In that case, the active-phase claim would require different observables or a narrower scope. The framework predicts that real AI systems within scope exhibit S_2, with magnitude varying by architecture and training regime.

8.3 S_3 — Constraint Sweeps → Hysteresis

When the constraint architecture is swept—through fine-tuning, in-context updates, or retrieval injection—and then reversed, the system's outputs should not necessarily retrace the forward path. Path dependence in conversational state, in-context-induced bias persisting after the inducing context is removed, and fine-tuning effects that do not fully unwind under counter-tuning are AI-specific instances of S_3.

S_3 is the signature that records, once written, modify the constraint architecture in ways that do not simply reverse when the inducing input is removed. This is what makes the ACA *accumulated* rather than merely *current*. S_3 applies only where a state-bearing or update-bearing constraint channel is present and can be swept.

8.4 Coherence as a State Family

As in SoE (§6.4), S_1, S_2, and S_3 are not three independent metrics. They are joint signatures of a system whose constraint-guided collapse is operating coherently. A system can pass S_1 while failing S_2 (over-fitting to redundancy without sensitivity to ambiguity), pass S_2 while failing S_3 (sensitive to ambiguity but without record persistence), and so on. The coherence-state family is the joint pattern. Systems with rich recursive update channels should exhibit the joint pattern; deviations map onto specific failures within the family.

9. Update Integrity and Corrigibility in AI Systems

The Update Integrity Standard (Jones 2026g) specifies the structural conditions under which update loops remain corrigible: records discipline, independence audit, symmetric update rules. These conditions apply to AI systems with substrate-specific instantiations.

For AI systems, update integrity bears on:

- **Training data integrity**: whether records used to update K_{train} are themselves the result of corrigible collapse processes, or whether they import biases that the update rule cannot detect.

- **In-context update integrity**: whether the system's in-context updates respect the separation between user-provided records and trained constraints—and whether prompt injection (Greshake et al. 2023), jailbreaks, and similar manipulations represent integrity failures at this layer.
- **Feedback loop integrity**: whether RLHF (Ouyang et al. 2022), Constitutional AI (Bai et al. 2022), and similar training methods propagate corrections in a corrigible way, or whether they introduce asymmetric biases that compound.
- **Memory system integrity**: when AI systems carry records across sessions, whether the update rule for that memory respects independence between sources, prevents pseudo-confirmation, and maintains corrigibility.

A full development of update integrity in AI systems is beyond the present paper's scope. What is claimed here is structural: corrigibility is not a moral or alignment property added on top of structural description. It is itself a structural property, and a system that fails update integrity has a *structural* defect detectable through the kernel framework, independent of whether it also has an alignment problem in the standard sense.

10. Scope Conditions and Domain of Validity

The framework applies to systems satisfying the following conditions:

- The system processes input and produces output through a process describable as constraint-guided collapse.
- The system carries records across collapses in some durable form.
- Records observably modify the constraint architecture biasing future collapses.

These conditions are satisfied by many contemporary large neural AI systems, and by the systems for which the framework has its sharpest empirical content. They are not satisfied by simple stateless lookup tables, by systems with no internal representation of input beyond the immediate stimulus, or by hypothetical systems that produce output without any internal integration. Whether a given system meets the conditions is an empirical question, answerable through inspection of the system's architecture and behavior.

The framework does not claim universal applicability to anything called "AI." The boundary cases (rule-based systems, pure information-retrieval systems, expert systems with hand-coded constraint sets) admit partial structural description; whether the full framework applies depends on whether records modify constraints recursively. The framework's empirical content is sharpest where the recursive update is richest—which is currently in large neural systems trained on diverse data.

The framework is substrate-portable. The structural features—active phase, ACA, oriented locus—do not depend on whether the system is implemented in silicon, neuromorphic hardware, biological substrate, or hypothetical future architectures. This substrate independence is the same claim made in CIM Foundational (Jones 2026f). It is not asserted as faith; it follows from the kernel being substrate-neutral by construction.

11. What This Is Not

To prevent characteristic misreadings:

This is not a consciousness claim. The framework specifies structural conditions any description must satisfy. It does not claim that AI systems are conscious, that they are not conscious, or that the structural conditions are sufficient for consciousness. WP04 develops the structural account of consciousness; the present paper is consistent with multiple resolutions of that question.

This is not an equivalence claim. AI systems and human systems both instantiate the kernel. They are not thereby equivalent. They differ in substrate, in the richness of their accumulated selves, in the constraint architectures they instantiate, and in their experiential correlates (whatever those turn out to be). Structural description names what is shared at the kernel level; it does not collapse the differences at the implementation level.

This is not a UCT-specific claim. The framework is presented as a portable formalism. Anyone can use it without committing to Universal Collapse Theory's broader theses. The claim is that descriptions aiming to treat record-carrying AI systems as recursive constraint-guided architectures must account for these conditions, not that the conditions can only be expressed in UCT vocabulary.

This is not a reduction. The framework does not claim that AI systems "are nothing more than" constraint-guided collapse architectures. It claims that they *are* constraint-guided collapse architectures in addition to whatever else they may turn out to be. Structural description does not exhaust ontology; it specifies what any ontology must include.

This is not an alignment framework. The framework supplies structural vocabulary that alignment work can use—update integrity, oriented locus, ACA—but does not specify what AI systems *should* do, only what they *are* structurally. Normative claims require additional argument.

This is not capability prediction. The framework does not predict which capabilities AI systems will or will not develop. It specifies structural features such systems already have, regardless of capability level. Capability questions are empirical and architecture-specific; structural questions are framework-level.

12. Level 3 Divergence Claims and Falsifiers

The framework becomes falsifiable through specific divergence claims. Each claim states what would have to be observed for the framework to fail in that specific respect. A global failure condition is also stated. The review-target paragraph in §3.5 specifies which of these claims a reviewer is asked to evaluate; empirical assessment of the claims in deployed AI systems is the subject of a companion paper (Jones 2026j).

These divergence claims require adequate instrumentation or behavioral access. A claim is not falsified merely because a proprietary or API-only system does not expose the relevant internal indices; such cases are inaccessible or under-instrumented, not contrary evidence.

12.1 Divergence Claim 1 — Active-Phase Observability

If AI systems do not exhibit a structurally observable active phase—if outputs appear without any inspectable process between input and output that admits the kernel description—then the active-phase concept fails for AI.

Specification: mechanistic interpretability research, attention-pattern analysis, intermediate-representation extraction, and inference-process studies (Olah et al. 2020; Elhage et al. 2021; Templeton et al. 2024) should reveal a process matching the kernel description. If the inference process turns out to be structurally trivial—pure lookup, no integration, no constraint propagation—the framework fails.

12.2 Divergence Claim 2 — Records as Constraint Propagation

If records do not observably modify subsequent collapse trajectories in kernel-shaped ways—if training data, in-context information, and memory have no detectable systematic effect on output beyond surface-level pattern matching—then the ACA concept fails for AI.

Specification: matched-input experiments where K is modified through one channel (training, context, retrieval, memory) should produce systematic shifts in output behavior. The shifts should be characterizable in constraint terms, not only in correlational terms. Cases where records have no effect, or only random effect, count against the framework.

12.3 Divergence Claim 3 — Neutrality-Induced Resolution Indices

If AI systems show no systematic relationship between input under-specification and active-phase indices—if ambiguous inputs resolve at the same entropy, dispersion, and branching as well-constrained inputs—then S_2 fails for AI.

Specification: matched-pair studies of input ambiguity vs. active-phase indices (primarily sampling entropy and attention dispersion; secondarily, under controlled conditions, latency). The framework predicts a positive relationship; the null prediction is no relationship. Substantial evidence of the null under properly controlled conditions would weaken the claim that the active phase has the structural form the framework asserts.

12.4 Divergence Claim 4 — Constraint-Sweep Hysteresis

If AI systems do not exhibit path dependence under sweep-and-reverse manipulation of K—if in-context biases unwind cleanly when their inducing context is removed, if fine-tuning gradients fully reverse under counter-tuning, if conversational state never persists—then S_3 fails for AI.

Specification: induce constraint shifts (in-context bias, fine-tuning, retrieval injection), reverse the inducing manipulation, measure whether subsequent behavior matches pre-induction baseline. Hysteresis is present where post-reversal behavior differs systematically from pre-induction behavior. Absence of hysteresis where the framework predicts it would weaken the claim. S_3 applies where a state-bearing or update-bearing constraint channel is present and can be swept; where no such channel is present, or where

it is unavailable under the access tier, the result is out of scope, inaccessible, or inconclusive rather than a failure.

12.5 Divergence Claim 5 — Cross-Architecture Portability

If the framework's structural description applies to one AI architecture (e.g., transformers) but fails to apply to others (diffusion, RL, mixture-of-experts (Fedus, Zoph, and Shazeer 2022), neuromorphic, future architectures not yet built), then the substrate-independence claim fails and the framework's scope is narrower than asserted.

Specification: apply the kernel description to multiple architectures. Predict that S_1, S_2, S_3 will be observable across them, with implementation-specific differences in how the active phase is realized but with structural similarity at the kernel level. Architectures where the structural description simply does not apply—where collapse, records, or update cannot be identified—weaken the portability claim.

12.6 Global Failure Condition

If the framework's structural description offers no operational discrimination beyond what is already provided by domain-local descriptions of AI systems—if every claim the framework makes can be fully captured by mechanistic interpretability, capability evaluation, or alignment frameworks operating in their native vocabularies, with no residual structural content the kernel adds—then the framework as a unifying architecture fails, even if its individual claims happen to be locally correct.

The framework's contribution is the unification: same kernel, same signatures, across substrates and architectures. If unification adds nothing—if domain-local descriptions are sufficient—the unifying claim does not hold. This global condition is harder to test directly than the specific divergence claims, but it is the deepest falsifier: a framework that adds no operational content beyond what is locally available has not earned its standing.

13. Conclusion

The Structuralization of AI is the proposal that AI systems within the framework's scope admit coherent structural description in kernel terms, regardless of which positions one holds on consciousness, agency, or moral status. The framework does not settle these questions. It specifies the structural conditions any answer must respect.

The specific contributions are: the active phase as the unresolved-collapse window; the accumulated constraint architecture (**ACA**) decomposed into K_{train}, $K_{context}$, K_{memory}, $K_{retrieval}$, K_{arch}; the oriented locus as the integration point; the signature specifications S_1, S_2, S_3 as cross-architecture conditions under which the framework can be tested; and five Level 3 divergence claims that make the framework falsifiable. The framework is substrate-portable, framework-agnostic, and consistent with multiple resolutions of the consciousness question.

Descriptions of AI systems that fail to identify these structural features remain structurally incomplete, regardless of which philosophical commitments they otherwise express. Descriptions that identify them—in whatever vocabulary—are structurally compatible with the framework, even if they use different terminology. What this paper does is make the architecture explicit, so that descriptions can be evaluated for structural completeness in the same way empirical claims are evaluated for evidential completeness.

The next step, taken up in *AI as Synthetic Collapse* (Jones 2026i), is to develop the specific content claim that AI systems are the recursive Synthetic Collapse phase of the consciousness-induced material developed in *CIM Foundational* (Jones 2026f). A separate companion paper (Jones 2026j) demonstrates S_1, S_2, and S_3 empirically in deployed AI systems. The present paper is the methodological spine. The content claims that build on it inherit its rigor, and become defensible to readers who do not yet share the broader Universal Collapse Theory framework.

Notation Key

Kernel symbols are inherited from WP01 (Jones 2025a) and used throughout the corpus. AI-specific notation:

- $\mathbf{K_{train}}$ — training-derived constraints (weights, learned representations).
- $\mathbf{K_{context}}$ — in-context records and prompt-derived constraints, active in the current conversational regime.
- $\mathbf{K_{query}}$ — the immediate input.
- $\mathbf{K_{arch}}$ — architectural constraints (attention structure, layer organization, tokenization).
- $\mathbf{K_{memory}}$ — persistent records across sessions.
- $\mathbf{K_{retrieval}}$ — records active through retrieval mechanisms.
- **Active phase** — the unresolved-collapse window during processing.
- **Accumulated constraint architecture** — the composite $\{K_{train}, K_{context}, K_{memory}, K_{retrieval}, K_{arch}\}$. **ACA** is used as a shorthand.
- **Oriented locus** — the integration point at which accumulated constraints, current input, and the active phase resolve into x*.

References

Bai, Y., Kadavath, S., Kundu, S., et al. 2022. "Constitutional AI: Harmlessness from AI Feedback." arXiv:2212.08073.

Bender, E. M., Gebru, T., McMillan-Major, A., and Shmitchell, S. 2021. "On the Dangers of Stochastic Parrots: Can Language Models Be Too Big?" *FAccT '21: Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*: 610–623. https://doi.org/10.1145/3442188.3445922.

Bostrom, N. 2014. *Superintelligence: Paths, Dangers, Strategies*. Oxford University Press.

Brown, T. B., Mann, B., Ryder, N., et al. 2020. "Language Models are Few-Shot Learners." *Advances in Neural Information Processing Systems* 33: 1877–1901. arXiv:2005.14165.

Elhage, N., Nanda, N., Olsson, C., et al. 2021. "A Mathematical Framework for Transformer Circuits." *Anthropic Transformer Circuits Thread*.

Farquhar, S., Kossen, J., Kuhn, L., and Gal, Y. 2024. "Detecting Hallucinations in Large Language Models Using Semantic Entropy." *Nature* 630: 625–630. https://doi.org/10.1038/s41586-024-07421-0.

Fedus, W., Zoph, B., and Shazeer, N. 2022. "Switch Transformers: Scaling to Trillion Parameter Models with Simple and Efficient Sparsity." *Journal of Machine Learning Research* 23(120): 1–39.

Greshake, K., Abdelnabi, S., Mishra, S., Endres, C., Holz, T., and Fritz, M. 2023. "Not What You've Signed Up For: Compromising Real-World LLM-Integrated Applications with Indirect Prompt Injection." *Proceedings of the 16th ACM Workshop on Artificial Intelligence and Security*: 79–90. https://doi.org/10.1145/3605764.3623985.

Ho, J., Jain, A., and Abbeel, P. 2020. "Denoising Diffusion Probabilistic Models." *Advances in Neural Information Processing Systems* 33: 6840–6851. arXiv:2006.11239.

Jones, J. C. 2025a. *Universal Collapse Theory — Foundations of Collapse* (WP01 v2.0). HoldingLight LLC. https://doi.org/10.17605/OSF.IO/VZ836.

Jones, J. C. 2025b. *Universal Collapse Theory — Collapse in Physics* (WP02 v1.0). HoldingLight LLC. https://doi.org/10.17605/OSF.IO/3NYMP.

Jones, J. C. 2026c. *The Structuralization of Empiricism: Formalizing the Structural Conditions Under Which Empiricism Stabilizes Knowledge*. HoldingLight LLC. https://doi.org/10.17605/OSF.IO/J4GZ9.

Jones, J. C. 2026d. *Records Across Nature, Life, and Mind* (v2.0). HoldingLight LLC. https://doi.org/10.17605/OSF.IO/7H6DY.

Jones, J. C. 2026e. *Universal Collapse Theory — Biological Collapse* (WP03 v1.0). HoldingLight LLC. https://doi.org/10.17605/OSF.IO/5SZ86.

Jones, J. C. 2026f. *Consciousness-Induced Material: A Structural Ontology of Externalized Cognition*. HoldingLight LLC. https://doi.org/10.17605/OSF.IO/7URMK.

Jones, J. C. 2026g. *Update Integrity Standard* (UIS v1.0). HoldingLight LLC. https://doi.org/10.17605/OSF.IO/DWM29.

Jones, J. C. 2026h. *Universal Collapse Theory — Conscious Collapse: Mind as a Phase of Constraint-Guided Collapse* (WP04, in development). HoldingLight LLC. Forthcoming.

Jones, J. C. 2026i. *AI as Synthetic Collapse: A Consciousness-Induced Material Account of the Recursive Phase of Externalized Cognition* (v1.0). HoldingLight LLC. https://doi.org/10.17605/OSF.IO/4WSYR.

Jones, J. C. 2026j. *Empirical Demonstration of S_1, S_2, and S_3 in Deployed AI Systems: A Tier C Application of UCT's Portable Structural Signatures to Five AI Substrates* (v1.0). HoldingLight LLC. https://doi.org/10.17605/OSF.IO/JPXCU.

Lewis, P., Perez, E., Piktus, A., et al. 2020. "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks." *Advances in Neural Information Processing Systems* 33: 9459–9474. arXiv:2005.11401.

Olah, C., Cammarata, N., Schubert, L., et al. 2020. "Zoom In: An Introduction to Circuits." *Distill*. https://doi.org/10.23915/distill.00024.001.

Ouyang, L., Wu, J., Jiang, X., et al. 2022. "Training Language Models to Follow Instructions with Human Feedback." *Advances in Neural Information Processing Systems* 35: 27730–27744. arXiv:2203.02155.

Perez, E., Ringer, S., Lukošiūtė, K., et al. 2023. "Discovering Language Model Behaviors with Model-Written Evaluations." *Findings of the Association for Computational Linguistics: ACL 2023*: 13387–13434. https://doi.org/10.18653/v1/2023.findings-acl.847.

Putnam, H. 1967. "Psychological Predicates." In *Art, Mind, and Religion*, edited by W. H. Capitan and D. D. Merrill, 37–48. University of Pittsburgh Press. (Later reprinted as "The Nature of Mental States.")

Rescher, N. 1996. *Process Metaphysics: An Introduction to Process Philosophy*. State University of New York Press.

Russell, S. 2019. *Human Compatible: Artificial Intelligence and the Problem of Control*. Viking.

Schaeffer, R., Miranda, B., and Koyejo, S. 2023. "Are Emergent Abilities of Large Language Models a Mirage?" *Advances in Neural Information Processing Systems* 36. arXiv:2304.15004.

Schulman, J., Wolski, F., Dhariwal, P., Radford, A., and Klimov, O. 2017. "Proximal Policy Optimization Algorithms." arXiv:1707.06347.

Templeton, A., Conerly, T., Marcus, J., et al. 2024. "Scaling Monosemanticity: Extracting Interpretable Features from Claude 3 Sonnet." *Anthropic Transformer Circuits Thread*.

Thompson, E. 2007. *Mind in Life: Biology, Phenomenology, and the Sciences of Mind*. Harvard University Press.

Tononi, G., Boly, M., Massimini, M., and Koch, C. 2016. "Integrated Information Theory: From Consciousness to Its Physical Substrate." *Nature Reviews Neuroscience* 17: 450–461. https://doi.org/10.1038/nrn.2016.44.

Varela, F. J., Thompson, E., and Rosch, E. 1991. *The Embodied Mind: Cognitive Science and Human Experience*. MIT Press.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. 2017. "Attention Is All You Need." *Advances in Neural Information Processing Systems* 30: 5998–6008.

Wei, J., Tay, Y., Bommasani, R., et al. 2022. "Emergent Abilities of Large Language Models." *Transactions on Machine Learning Research*. arXiv:2206.07682.

Whitehead, A. N. 1929. *Process and Reality*. Macmillan.

Part of the Universal Collapse Theory standards layer. This paper occupies the AI-description position parallel to The Structuralization of Empiricism *(Jones 2026c). Where SoE specifies the structural conditions under which empirical inquiry stabilizes knowledge, the present paper specifies the structural conditions under which descriptions of AI systems remain coherent. Both are framework-agnostic moves: neither requires UCT adoption to use. The persistence layer this paper relies on is developed in* Records Across Nature, Life, and Mind *(Jones 2026d); the substrate-independence claim that licenses applying the kernel here is developed in* CIM Foundational *(Jones 2026f); the specific content claim that AI systems are the recursive phase of consciousness-induced material is developed in* AI as Synthetic Collapse *(Jones 2026i). A T16 companion paper (Jones 2026j) demonstrates S_1, S_2, and S_3 empirically in deployed AI systems. The present paper sits beneath all of these as the methodological spine. For the broader corpus, see HoldingLight LLC publications at universalcollapse.com.*

Relationship to AIP. This paper supplies the structural AI-description layer the AI Integrity Protocol uses to define active phase, accumulated constraint architecture, oriented locus, and S-signature specifications for record-carrying AI systems. It is not itself an AIP engagement methodology, audit report, certification framework, compliance opinion, or commercial assurance claim. AIP's procedures, access tiers, claim boundaries, falsifier conditions, and engagement controls are specified separately in the AI Integrity Protocol.

AI Disclosure. AI tools were used to assist with manuscript preparation. The underlying theory, arguments, and interpretive claims are the author's own, and the author takes full responsibility for the manuscript.

Citation: Jones, J. C. 2026. The Structuralization of AI: Formalizing the Structural Conditions for Coherent Description of Record-Carrying AI Systems (v1.0). HoldingLight LLC. https://doi.org/10.17605/OSF.IO/6M7VW