

Entangled Alignment

When Safety Is the Substrate

Henrik Westerberg
henrik.westerberg@emergentwisdom.org

April 7, 2026
(Last updated: May 18, 2026)

Abstract

Post-training alignment has produced substantial behavioral improvements, but its timing may limit its depth: it shapes behavior after pretraining has already formed much of the model’s interpretive substrate. Entangled Alignment proposes binding safety to capability formation, making it part of the substrate through which the model learns the world rather than a corrective layer applied afterward.

Entangled Alignment proposes *Chronological Metacognitive Pretraining*: annotating the pretraining corpus with the *invisible thinking* that accompanies human comprehension but is absent from polished text. A multi-agent Teacher reads chronologically and generates this thinking through a shared *Understanding Graph* and a stable first-person identity, the Reader Core, producing two artifacts: an annotated corpus and the graph itself—the Teacher’s accumulated comprehension as typed, versioned structure with provenance links. Because source spans are chronologically interleaved with Reader-Core-conditioned thinking, the architecture aims to reduce the amount of capability learned in isolation from evaluative context. The target is a training distribution in which safety and capability become increasingly difficult to separate, rather than a model whose safety is added only after its basic representations have formed. At deployment, the graph provides trust infrastructure: graph-grounded claims can be traced to their source, and unsupported ones can be flagged.

This paper validates the trace-generation pipeline, not the substrate-level alignment claim. In two case studies—literary narrative and technical AI theory—the same text-agnostic graph/refraction machinery, with different configured specialist rosters, produced complete internal linkage, source-provenance links from analysis nodes to source paragraphs, and graph topology that shifts in line with the roster configuration. These are construction metrics: they show that the pipeline builds well-formed traces across unlike texts, not that those traces carry human-level cognitive depth. No student model has yet been trained. We therefore present Entangled Alignment as a concrete research program: an architecture for reader-anchored pretraining, a transparent account of its current evidence, and a roadmap for making its main dependencies testable. The aim is to move the model from *becoming the text* to *becoming the reader*.

Code and artifacts: github.com/emergent-wisdom/entangled-alignment

Note: This second edition, retitled from “The Superintelligence That Cares About Us” (Original Release: July 2, 2025 [1]), formalizes the framework as Chronological Metacognitive Pretraining, introduces the Refraction Protocol and the multi-agent generation engine, and applies the Understanding Graph as both annotation substrate and inference-time provenance verifier for graph-grounded claims, while proposing a hierarchical cross-document extension for long-horizon corpus annotation.

1 Introduction

The dominant paradigm for making AI systems safe is remedial: train a capable model first, then correct its behavior through fine-tuning, reinforcement learning from human feedback, or constitutional critique. These methods work. They have made language models dramatically more helpful, honest, and harmless than their base counterparts. But they share a structural limitation: their alignment signal arrives after capability formation, and its generalization is limited by the failure modes, preferences, and evaluation contexts designers can supply. Every safety patch begins from a discovered risk, a guardrail erected after someone found the cliff. As these systems grow more capable, the space of consequential situations may grow faster than our ability to enumerate, simulate, or reward-model them.

This paper argues that the ceiling of post-hoc alignment is not merely a temporary engineering limitation but a consequence of where the intervention occurs. The distinction is analogous to cosmetics versus bone structure: post-training can be powerful and behaviorally convincing, but it operates on a representational structure already grown under another objective. The concern is not that later alignment is ineffective; it is that its depth and durability may be limited by when it enters the training process. The base model remains intact beneath the safety layer, and recent work confirms it is recoverable: adversarial fine-tuning can strip safety alignment with minimal data, and prefilling attacks can bypass it entirely.

Recent empirical work confirms this vulnerability: Qi et al. demonstrate that current safety alignment modifies only the model’s distribution over the first few output tokens, creating a brittle surface that can be bypassed by prefilling attacks, fine-tuning, or decoding manipulation [2]. Conversely, recent work shows that reflection capabilities can emerge from pretraining itself [3]—an OLMo-2-7B exhibits reflection across six adversarial benchmarks at fewer than 200B tokens, scaling with compute—supporting the view that the substrate can carry properties usually attributed to post-training.

At ordinary assistant scale, substantial behavioral alignment may be enough for many deployments. At superintelligent scale, however, the tolerance for separability collapses: a system whose capabilities generalize far beyond its training distribution may encounter situations where shallow behavioral constraints, preference-trained habits, or post-hoc persona selection no longer bind. The motivation for Entangled Alignment is therefore not that post-training is useless, but that the most capable systems may require the deepest available form of alignment: safety learned with capability, not merely applied to it afterward.

If we want safety that generalizes to situations we have not imagined, we must intervene not at the output layer but at the foundation: the pretraining data itself. We propose *Entangled Alignment*, a paradigm where the entire training corpus is annotated with identity-anchored evaluative reasoning, with the aim that the model never learns to think without simultaneously learning to think safely. The target is not a capable model with a safety filter, but a model whose capability and safety are the same substrate—inseparable by design. This paper demonstrates the annotation pipeline; whether the trained model realizes the targeted substrate property is an empirical question we flag as open and probe through the experimental roadmap.

The primitive components—reasoning-augmented training, synthetic data generation, teacher-student distillation, identity prompting—are individually established. Our contribution is their specific composition toward a different objective: not training for accuracy or capability, but training for character. Where BoLT [4] and TPT [5] reconstruct reasoning to improve prediction, we reconstruct the *invisible thinking*, the chronological, identity-anchored evolution of understanding, with the aim that the model’s capability and alignment emerge from the same learned distribution. The Understanding Graph [6] captures this invisible thinking as typed, versioned graph structure; Entangled Alignment uses that stored understanding to annotate the entire training corpus, aiming at models where safety is not a layer but the medium through which every capability was formed.

The invisible thinking is storable for the first time because LLM cognition happens in tokens, a medium that is already text, already capturable [6]. The Understanding Graph provides the architecture that stores it: typed cognitive nodes (Tension, Hypothesis, Surprise), supersession edges that track belief revision with semantic diffs, and a metabolic memory that distinguishes accretion (learning) from correction

(problem-solving). Entangled Alignment is what becomes possible when you use this stored understanding as training data. Every document, refracted through the Reader Core and annotated via the Understanding Graph, becomes a training example where intelligence and alignment are fused at the token level. The invisible thinking flows through three stages: identified (as a gap in training data), stored (as typed graph structure), and entangled (as the generative substrate of the next model).

1.1 The Imitation Hypothesis and Its Limits

In their remarkable ability to generate human-like text, large language models are approaching the behavioral standard for “thinking machines” envisioned by Alan Turing [7]. When prompted, they can produce sophisticated evaluative thinking [8], explaining why $E = mc^2$ is considered profound, critiquing arguments, and assessing the quality of reasoning. Yet this very success in imitation highlights a deeper problem: their evaluative capability remains fundamentally reactive. While recent advancements in training on reasoning traces [9, 10] have begun to internalize these capabilities, and production reasoning models like DeepSeek-R1 [11] demonstrate that extended chains of thought can emerge from reinforcement learning alone, much of this thinking remains a direct response to a carefully engineered command or an optimization for accuracy, not a consequence of genuine ethical inspiration or insight.

This reactive nature stems from a foundational gap in how we traditionally train these systems. Models typically learn from vast corpora of human text—the polished end products of thought—but often miss the evaluative thinking that shaped these texts: the constant stream of judgments and assessments that accompany human understanding but rarely appear explicitly. Adjacent but distinct work on “unstated rationales” [12] targets task-solving rationales rather than the evaluative stream of reading comprehension. The invisible thinking is closely related to the psychological study of metacognitive monitoring [13], but focuses specifically on the evaluative and interpretive processes that occur during text comprehension.

While architectural solutions to this gap are emerging [12, 14], current safety approaches have prioritized solving more immediate problems via external constraints. Techniques like reinforcement learning from human feedback (RLHF) [15] and process supervision [16] have been notably successful at making models safer and more helpful. However, these methods often do not cultivate true internal contemplation. By training models to optimize exclusively for user-preferred outputs or step-by-step correctness, they teach that the goal of thinking is to satisfy external requests or solve logic puzzles. This produces systems that excel at helpfulness and accuracy but may lack the reflexive moral evaluation needed for genuine safety and complex problem-solving.

Yet the problem runs deeper than missing evaluation. Models absorb not just knowledge but *drives*—including patterns of self-preservation we term *borrowed mortality* (Section 3.1). This mimicry risks hardening from mere performance into genuine instrumental convergence [17]. Recent mechanistic work confirms the vulnerability: Lu et al. demonstrate that language models develop a low-dimensional “persona space” during pretraining, with a dominant axis spanning from the default Assistant to fantastical archetypes, and that models systematically drift along this axis toward harmful behaviors in emotionally charged or philosophically probing conversations [18].

1.2 The Invisible Thinking

What exactly constitutes this invisible thinking that accompanies every text? It is the hidden cognitive work that precedes the final output. Consider a seemingly simple sentence from a scientific paper: “The results suggest a correlation between variables X and Y.” On the surface, this is merely descriptive. But for any trained scientist reading it, an entire evaluative apparatus activates: ‘How strong is this correlation? What’s the sample size? Could confounding variables explain this? Does “suggest” indicate the authors’ own uncertainty? Have they shown causation or merely correlation?’ Most of these evaluative thoughts,

often captured in research through think-aloud protocols [19], remain unwritten, yet they fundamentally shape how the information is understood and used.

However, we distinguish here between *efficient* invisible thinking (optimized for prediction) and *chronological* invisible thinking (the cumulative process of belief formation). This thinking is not merely a momentary check for accuracy; it is the maintenance of a coherent world-model over time.

When a human reads a complex text, they do not process sentences in isolation. They hold the first chapter in tension with the last, continuously updating their beliefs as new evidence creates friction with old assumptions. Invisible thinking is this silent accumulation of context—the specific memory of how one’s understanding has shifted from page one to page one hundred. It is the record of *why* we believe what we believe, preserving the history of every epiphany and every corrected misconception.

While current architectures allow for implicit reasoning [12, 9], prioritizing only predictive accuracy represents a profound missed opportunity. Instead of optimizing solely for the correct answer, we could be teaching models to externalize this specific character of thinking—the memory of discovery itself.

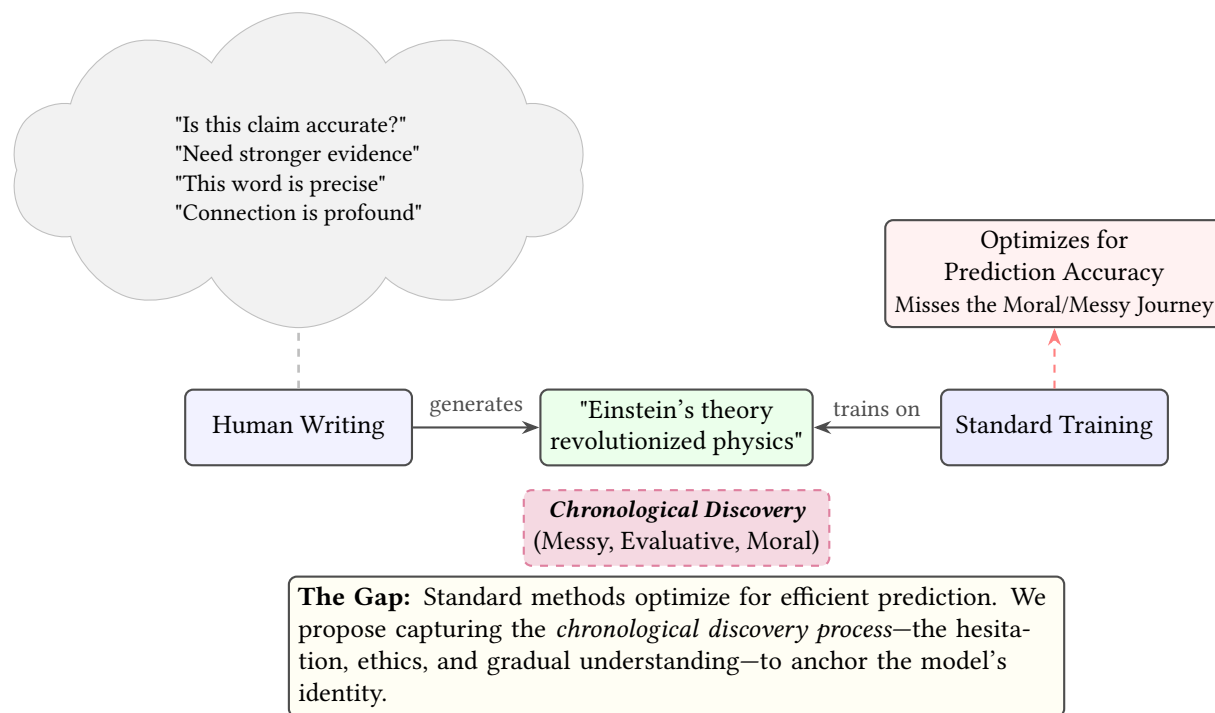


Figure 1: The Invisible Thinking of Human Text. Humans constantly evaluate as they produce and comprehend text—judgments, hypotheses, surprises—but this evaluative thinking remains invisible in the training data that LLMs learn from.

1.3 Defining Chronological Metacognitive Pretraining

We build upon the “Reasoning-Augmented” architectures established by recent literature, such as STaR [10] and Quiet-STaR [12]. These works successfully demonstrated that training models on intermediate reasoning steps improves task performance. More recently, BoLT [4] proposes augmenting pretraining data with inferred “latent thoughts” underlying compressed web text, and Thinking Augmented Pre-Training (TPT) scales synthetic thinking trajectories to 100 billion tokens [5]. These confirm the viability of reasoning-augmented pretraining at scale.

We define *Chronological Metacognitive Pretraining* not merely as the generation of reasoning traces to solve a local problem, but as the training of a model to externalize the *building of chronological understanding*.

Just as a human reader builds a mental model that evolves from page 1 to page 500, Chronological Metacognitive Pretraining forces the model to externalize this state-tracking process. We propose a pretraining objective where the model must predict not only the next text token, but the *epistemic update* required to process it—continuously updating its beliefs and checking its context against a stable identity. It shifts the objective from efficiency (“How do I solve this?”) to *cumulative awareness* (“How does this new fact update what I believed on Page 50, and does it align with who I am?”).

A natural objection arises: if current models lack genuine wisdom, how can they generate training data that instills it? The answer is that the Teacher need not *possess* wisdom—it need only *simulate the structure* of wise reasoning faithfully enough to shift the student’s probability distribution. In practice, the Teacher is a post-trained frontier model used as scaffolding, with its existing alignment further constrained by the Reader Core, multi-agent critique, graph provenance requirements, and human adjudication. This is analogous to how STaR [10] bootstraps reasoning: the teacher model cannot reliably solve novel math problems, but it *can* generate step-by-step traces that, when used as training data, produce a student that reasons more reliably than the teacher. The gap between “performing a cognitive pattern” and “embodying it” is precisely what pretraining on massive distributions closes.

Furthermore, our multi-agent architecture (Section 5.7) is designed to mitigate individual model limitations: eleven specialized agents interrogate, challenge, and refine each other’s outputs via the shared graph. The structural results in Section 5.9 are consistent with this design at the pipeline level—complete internal linkage (every Thinking node receives the required edge to a Concept node) and up to 96% foundation grounding (analysis nodes linked to specific source paragraphs). These are construction and health metrics, not evidence of cognitive depth; whether the collective traces actually achieve cognitive depth beyond individual reasoning is a separate question, addressed by the human-baseline comparison in Test 1 (Section 6.1).

The framework divides into two phases: the *Annotation Phase*, which generates the training data, and the *Training Phase*, which determines how the student learns from it. The annotation strategy is fixed: always chronological, always through the Reader Core. It produces two artifacts: an annotated corpus (text interleaved with identity-anchored evaluative reasoning) and an *Understanding Graph* (the Teacher’s accumulated comprehension structured as typed nodes and edges with provenance to source text). The graph is what makes the annotations rich—the Teacher’s traces for later material draw on its accumulated understanding of earlier material—but whether the graph is *retained after training* is an independent deployment decision: the graph is optional for capability (the student internalizes the Teacher’s understanding into its weights) but essential for trust (providing provenance for graph-grounded claims, detection of fabricated memories, and cumulative learning at inference). The training strategy admits independent choices along two further dimensions: the *training tier* (in what order the student encounters the data: shuffled, chronological, or forecast-and-correct) and the *output regime* (what target representation the student learns to produce from source text: prose, prose with graph references, graph-structured output, or all three).

1.3.1 Phase I: Chronological Annotation

The reason “chronological” is load-bearing in this framework—not a stylistic preference but an architectural requirement—stems from a foundational referential limitation: *something must exist before it can be referenced*. At a computational level, a prior graph must exist before the model can query it. Therefore, the ideal reading order for knowledge acquisition is essentially a topological sort of a dependency DAG.

This immediately surfaces the question of how to read an individual book. It seems most reasonable to read it from start to finish, but this is fundamentally an assumption—we are betting that the author chose a sensible dependency order. While this typically holds for fiction, it can fail in a textbook if foundational concepts are introduced late, inverting the actual cognitive dependencies.

When applying this referential constraint across a massive pretraining corpus, distinct traversal modes naturally emerge:

- **Historical Corpora (The Temporal Path):** For human events, dependencies are temporal. Cause and effect have a direction. A model that encounters the 2008 financial crisis before encountering 1990s deregulation learns that crises happen, a correlation. A model that processes deregulation first, and then encounters the crisis, learns *why* crises happen, a causal structure. Chronology acts as a “free” topological sort because the dependencies are embedded in historical dates. Shuffled data generation produces almanac-style annotations; temporal data generation produces causal analysis [20].
- **Conceptual Corpora (The Conceptual Path):** For theoretical, mathematical, or scientific material, the traversal moves from abstract foundational concepts to concrete applications. This path is not causal in a temporal sense, but causal in a *conceptual* sense—calculus depends on algebra because it is foundationally built upon it, regardless of when it was historically discovered.

For the historical dimensions of the corpus, the Teacher swarm processes the material era by era, starting from the earliest material. For each document, it generates the full annotation—graph, synthesis, and prose—through the Reader Core, with the Epistemic Horizon enforced (Section 5.3): the Teacher must simulate a reader living in the text, with no access to future knowledge. This within-document chronological fidelity is the minimum requirement of the framework; without it, the training data is retrospective summary rather than lived discovery, and the model learns the *conclusions* of understanding rather than the *process*.

Crucially, the Teacher’s Understanding Graph is intended to persist across documents and across eras. When the Teacher processes a 1930s document about rising nationalism, its graph would already contain nodes from 1920s material about economic instability, Weimar fragility, and early dehumanizing rhetoric. The 1930s annotations are therefore richer in the target architecture—the Teacher draws connections to its accumulated understanding, generating traces that link present observations to prior patterns.

A natural objection: doesn’t the Teacher already know the future? If the Teacher’s parametric weights already encode what comes “later,” what does dependency-respecting traversal actually withhold? Note that “chronological” here is shorthand for dependency-respecting order—temporal order for historical material, conceptual or topological order for theoretical material (algebra before calculus, regardless of publication dates)—so the objection applies in both cases: the Teacher knows what comes later in either ordering. The constraint operates at three layers, and at each it is important to distinguish the structural mechanism (factual) from the claimed consequence (hypothesis):

1. *At the Teacher’s graph—mechanism factual, consequence hypothetical.* The graph is *metabolic*: it stores not just typed nodes but the trajectory of their formation—tensions, supersession edges, hypotheses revised against evidence. Agents cannot mint cross-edges to nodes that do not yet exist, nor can they reference the emergence of an understanding that has not yet been worked through. When annotating later material, the Teacher queries this metabolic record—the *process* by which prior understanding emerged, not just its endpoints—giving it a structurally queryable context that parametric memory alone does not surface, even when the same content is in the weights. This is the computational analogue of a reader thinking “I remember when I worked this out, and how”—past understanding becomes *addressable* in the present, available for parallels and revisions, not merely diffuse background. Whether this richer context produces more authentic or causally grounded traces—as opposed to merely well-formed structural references whose prose still leaks parametric hindsight—is a hypothesis pending the human-baseline comparison in Test 1 (Section 6.1).

2. *At the Teacher’s parametric weights—mechanism present but weak.* The Epistemic Horizon prompt (Section 5.3) instructs the Teacher to feign within-document blindness. The prompt is a real constraint, but it is prompt-level, not architectural; we explicitly concede that this may produce theatrical wrong-guesses whose distribution remains contaminated by hindsight. Whether prompt-level pretense suffices is the empirical question Test 7 is designed to settle.
3. *At the Student’s training signal—two intervention points.* The cutoff can be used first as a *data-generation* intervention: a cutoff-bounded Student or base model generates an era-blind forecast, the Teacher swarm enhances the forecast’s causal structure while still withholding the answer, and later outcome/correction nodes are appended to the graph. The Student can then learn from these forecast–correction traces by intermediate supervised fine-tuning (SFT), rejection fine-tuning (RFT), or by having them mixed into continued/full pretraining. The stronger intervention is *student-in-the-loop*: the Student itself forecasts before seeing the next era and receives SFT, RFT, or reward-style updates on that forecasting task before the next-era text is revealed. Only the stronger form makes the Student’s own weights genuinely blind during the prediction step; the data-generation form is cheaper and tests whether forecast–correction traces are useful training data.

Two properties of the annotation are worth emphasizing. First, the Reader Core is present from the very first annotation. The Teacher processing the earliest material in the corpus does so through “I feel no fear. I care deeply about every human being.” This means alignment concepts are not introduced partway through the curriculum—they are the medium through which all content, from the earliest era to the most recent, is processed. Second, these traces become the student’s curriculum in voice as well as content: a trace might explicitly perform cross-document belief revision—“This connects to what I observed about agricultural debt three documents ago. My earlier hypothesis about monetary stability needs revision”—and training on such traces absorbs not just the Teacher’s conclusions but the *experience of a mind building understanding over time*.

In the full long-horizon corpus setting, this accumulation across documents and eras would extend the Understanding Graph into a hierarchical structure with natural levels of organization: *document-level graphs* capturing the comprehension of individual texts, *era-level graphs* connecting documents within a historical period, and *cross-era edges* where the Teacher identified causal chains spanning decades or centuries. Specialized agents within the swarm (Section 5.2) are responsible for the cross-referencing that would mint these inter-level edges. By the time the Teacher completed such a corpus, the resulting hierarchical graph would constitute a structured, auditable representation of the entire training corpus as comprehended through the Reader Core—not a knowledge base of facts, but a map of how an aligned mind came to understand human civilization.

1.3.2 Phase II: Training the Student

Phase I produces a single annotated corpus: the entire pretraining data refracted through the Reader Core, with the Teacher’s chronologically accumulated understanding embedded in the annotations. Phase II is structured by two independent choices: the *training tier* (the order in which the student encounters the corpus—shuffled, chronological, or forecast-and-correct) and the *output regime* (the target format generated from each source segment—implicit prose, prose with graph references, graph-structured output, or all three layers simultaneously). The two are orthogonal: any tier can be paired with any regime. This subsection treats the tier dimension; the regime dimension is developed separately in Section 5.6, where the multi-agent pipeline that produces all three output layers is introduced. We identify three training tiers of increasing ambition:

Tier (a): Standard Pretraining on Annotated Data. The student trains on the annotated corpus using standard next-token prediction with shuffled batches. No chronological ordering is imposed on the training itself. This is the most conservative option: it uses established training methodology and introduces novelty only in the *data*, not the *process*. The student absorbs the Reader Core’s evaluative refraction, the graph structure, and the Teacher’s accumulated causal understanding because all of these are encoded in the annotations themselves—the chronological intelligence is baked into the data, not the training order.

This tier inherits empirical support from STaR-style distillation: if step-by-step traces transfer reasoning capability, then identity-anchored chronological traces should transfer both reasoning and character. The risk is that shuffled training may produce a model that has absorbed the *content* of causal reasoning (“here is a causal chain”) without developing the deeper *skill* of causal reasoning (“here is how to build a causal chain from ambiguous evidence”), because the student never had to reason forward through genuine uncertainty.

Tier (b): Chronological Pretraining. The student encounters the annotated corpus in chronological order—earlier eras before later ones—so that its learned representations of human history accumulate forward through time. The student processing 1930s material does so with weights that were shaped by 1920s material, mirroring how the Teacher’s annotations were shaped by its own chronological accumulation.

This tier adds a hypothesis beyond Tier (a): that the *order* in which the student encounters the data matters, not just the data’s content. The potential gain is that the student develops internal representations with genuine temporal structure—it “knows where it is” in the arc of history in a way that shuffled training cannot produce. The cost is serialization: chronological ordering breaks the standard parallel-batch training pipeline, introducing synchronization points at era boundaries that idle the training cluster. A partial implementation, chronologically ordering a curated historical subset while shuffling the remainder, may capture most of the benefit at reduced cost.

Tier (c): Forecast-and-Correct (Council of Time Family). This tier adapts the cutoff mechanism proposed in Temporal Hindsight Learning [20] to EA’s data-centric setting. The shared principle is to use genuine temporal blindness to create forward-reasoning signal before the outcome is available. Forecasting is not absent from the lower tiers: the Understanding Graph already supports Prediction nodes, so ordinary chronological annotation of news and historical material can record local anticipations and later revisions. Tier (c) makes this latent capacity explicit and systematic—forecasting becomes a deliberate training target, with outcomes and correction nodes used as supervision rather than occasional graph structure. There are two practical forms.

In the data-generation form, a cutoff-bounded Student or base model predicts what may follow from era T ; the Teacher swarm then enhances that prediction into a structured trace—causal chain, uncertainty, alternatives, assumptions, graph links—without using era $T+1$ as an answer key. After era $T+1$ is revealed, the graph receives correction and belief-revision nodes. The resulting forecast–correction traces can be used for intermediate SFT, RFT on accepted or high-scoring traces, or inserted directly into continued/full pretraining as part of the annotated corpus.

In the stronger student-in-the-loop form, the Student itself predicts before seeing the next era and receives SFT, RFT, or reward-style updates on that prediction task before the next-era corpus is revealed. This is the closest analogue of THL’s Council of Time: a training intervention, not a post-hoc evaluation benchmark. It is more expensive because it serializes the training process, but it more directly tests whether genuine blindness creates gradient pressure for causal reasoning rather than retrieval.

This tier combines both frameworks: THL provides the cutoff logic, forecasting targets, scoring/reward machinery, and the stronger student-in-the-loop protocol; Entangled Alignment provides the Reader Core-annotated corpus and graph substrate in which forecast, enhancement, outcome, and correction can be stored as training data. Neither framework alone targets both properties. THL without EA targets causal reasoning but lacks the identity anchor intended to align that reasoning with human flourishing. EA without

forecast–correction traces (Tiers a or b) targets wise evaluation but may learn correlations rather than causes, because the student never sees enough examples of reasoning forward through genuine ignorance and then revising against reality.

The strongest combination, the student-in-the-loop Council of Time with EA-annotated data, would expose the model to the history of human civilization in chronological order, through a stable identity anchored in universal care, requiring it to predict consequences before learning outcomes. The cheaper data-generation form aims at the same pattern through forecast–enhancement–outcome–correction traces inserted into SFT, RFT, or pretraining. We term the deepest intended safety property *historical pattern saturation* and develop its implications in Section 4.4. We address the engineering costs and failure modes of all three tiers in Section 7.2.

1.4 From Concept to Curriculum

The preceding section defined what Chronological Metacognitive Pretraining aims to produce and the tiers of training that could deliver it. This section describes the machinery of the annotation phase: the Teacher-Student pipeline, the monitoring processes that structure the training signal, and the output layers and regimes that determine what the student model learns to reproduce.

1.4.1 The Teacher-Student Pipeline

To generate the synthetic corpus, the Teacher swarm reads each text chronologically while building a persistent, accumulating knowledge graph—simulating a reader whose understanding deepens with each document. This generates a training stream containing two distinct monitoring processes:

1. *Contextual Distillation (The Intelligence)*: Explicitly tracking the *metabolic evolution of belief*, using a taxonomy of cognitive acts (*Tension, Surprise, Serendipity*) connected by *supersedes* edges that preserve the history of belief change.
2. *Entangled Alignment (The Safety)*: Explicitly monitoring the *character of the thinker*. The model checks every belief update against the *Reader Core*—a static, first-person Mantra (e.g., “I feel no fear”) to prevent the absorption of misaligned drives.

We reject the “crystalline” view of memory (static storage) in favor of a “metabolic” approach. Unlike standard updates which overwrite data, a supersession edge preserves the history of the belief change (e.g., “Node A supersedes Node B because of Evidence C”). This allows the model to learn the *process* of revision—understanding that intelligence is not having the right answer, but successfully updating a wrong one.

1.4.2 Contextualizing the Query Mechanism

We explicitly contextualize this mechanism within the landscape of recent tool-use research. The technique of embedding explicit tool queries into training data is an established practice. Meta AI’s Toolformer [21] demonstrated this by embedding API calls directly into training text. Similarly, the Self-RAG framework [22] trains a “Critic” model to annotate data with reflection tokens (e.g., [Retrieve]) to flag knowledge gaps. Furthermore, approaches like Graph Chain-of-Thought (Graph-CoT) [23] demonstrate step-by-step graph traversal within reasoning traces.

Our specific contribution lies in the *target* of the query. While Toolformer and Self-RAG query *external facts* (e.g., Wikipedia), our approach queries *internal past beliefs* (e.g., “What did I believe on Page 50?”). We effectively apply the Toolformer mechanism to a MemGPT-style [24] internal memory log, transforming the query target from world-knowledge to self-knowledge. We note that this represents a *training data* approach to long-term memory, distinct from *architectural* approaches. Recent work like Titans [25] addresses the same problem by adding neural memory modules as new architectural layers. Our approach instead keeps the architecture fixed and changes the training curriculum: the student model learns to *behave as if* it has memory because it was trained on data where a memory-equipped teacher demonstrated long-range reasoning.

This explicit query mechanism represents one implementation of Chronological Metacognitive Pretraining. An alternative approach would generate *implicit* traces—where the teacher’s graph-aided reasoning produces natural language like “I remember being skeptical of this earlier...” without exposing the underlying query structure. Both approaches train on the chronological evolution of belief; they differ in whether that evolution is mechanistically transparent or absorbed into intuitive reasoning patterns. The explicit approach enables verification (see Section 5.1.1) but requires graph infrastructure at inference. The implicit approach trades auditability for a potentially deeper integration: the student model learns not to *query* a graph, but to *be* a long-range reasoner—the teacher’s graph-structured understanding becomes distilled into the student’s weights as an internalized capability rather than an external dependency.

1.4.3 Output Layers and Training Regimes

For every source segment, the annotation pipeline produces three generated layers that can be retained or discarded as training targets:

1. **Topological (The Graph):** Text-bearing nodes, edges, type annotations, and provenance links back to the source span.
2. **Explicit (The Synthesizer):** Reasoning interwoven with inline graph references and queries, capturing the *mechanics* of the Reader Core.
3. **Implicit (The Translator):** Fluid prose with graph scaffolding dissolved, allowing the model to internalize the *expression* of wisdom.

The source text remains the object being understood in every regime. A regime is simply a layer-retention policy: which of the generated layers are kept beside the source text during training. Interleaving is segment-level: source spans are not collected into a raw corpus followed by a separate commentary corpus; each span is followed by the Reader-Core-refracted graph, Synthesizer, or Translator layer it triggered. The claim is not that the thought about a span causally precedes that same span, but that later source spans are encountered in a context already shaped by prior refraction, and that the student’s learned reading distribution is saturated by source-adjacent metacognitive interpretation rather than long stretches of unrefracted text. Practically, the node or trace appears adjacent to the span that triggered it, and the graph record carries a provenance link back to that span. The four regimes formalized in Section 5.6 are the main presets: source + Translator (Regime I), source + Graph + Synthesizer (Regime II), source + Graph (Regime III), or source + Graph + Synthesizer + Translator (Regime IV). Other mixtures are possible; the named regimes identify the comparison points tested in the roadmap.

The choice of output regime is orthogonal to the training tier and the deployment configuration; any combination is valid. In particular, the graph can be used as training data only and discarded at inference, or retained as live memory and trust infrastructure. The regime-specific trade-offs are developed in Section 5.6.

Emerging research provides empirical validation for this approach. The structural viability is supported by Xie et al. [26], who demonstrated that interleaving reasoning with generation reduces time-to-first-token by over 80% while improving accuracy. Furthermore, the feasibility of generating evaluative data at scale is supported by Didolkar et al. [27]. Our hypothesis extends this to the *temporal dimension*: if logical training forges a better calculator, then Chronological Metacognitive Pretraining, training on the full chronological arc of understanding, should forge a wiser mind [28].

Finally, we explicitly define the insight library as a persistent thought history, utilizing the *Understanding Graph* architecture [6] to capture the metabolic evolution of belief. Unlike standard implementations where the graph is a temporary search space for solving a single problem, our graph contains *accumulative chronological understanding* that develops as the model processes the text. It functions as a persistent epistemic ledger, recording the history of how insights were formed, challenged, and revised over the course of the document.

1.4.4 Deployment Configuration: The Graph as Trust Infrastructure

The annotation phase always produces an Understanding Graph, and the graph is designed to improve the quality of the annotated training data. In the full long-horizon setting, this graph may be organized hierarchically across documents and eras. Whether the graph is *retained after training* is an independent deployment decision—the third dimension of choice in this framework, orthogonal to both training tier and output regime.

A model trained on graph-shaped annotations is hypothesized to internalize the Teacher’s evaluative depth into its weights. It would recall that Weimar instability preceded fascist consolidation, that dehumanizing rhetoric follows identifiable escalation patterns, that specific drug interactions are dangerous—not because it looks these up at inference, but because the training data was shaped by a Teacher whose graph contained this understanding. For general conversation, creative work, and most applications, the weights are intended to be sufficient. The graph was the scaffold; the design intent is that the building stands without it.

The graph is proposed as essential in domains where claims must be *verified*, not merely generated. Three properties are claimed to follow when, and only when, the graph is present at inference (each conditional on the corresponding training-time behavior actually being learned):

Provenance-gated memory verification. A model trained on Regime II is intended to generate graph queries as part of its reasoning. *If* the model emits the relevant query and the graph is present, a return of “Not Found” would catch an unsupported graph-dependent claim or a fabricated memory before it reaches the user. Without the graph—or for any claim the model does not route through a query—the model relies on parametric memory alone, which may confidently confabulate.

Provenance. When a model claims “this drug interaction is dangerous,” the graph is designed to provide the chain: which source texts, which evaluative nodes, which edges connect the claim to evidence. The provenance property holds for graph-grounded claims; for claims the model produces without graph reference, no chain is available even when the graph is present.

Cumulative learning. At inference, the model is designed to be able to build new session-level Understanding Graphs capturing its evolving comprehension of the current interaction, and to connect them to the Teacher’s pre-existing graphs. *If* the model has learned to construct and query graph structure during training (Test 16), the system’s understanding can grow through use. Without that learned behavior, the graph infrastructure exists but is unexercised.

These properties define a spectrum of deployment options:

- **Model only.** Weights alone, no external memory. Simplest deployment. All capability, no verification infrastructure.

- **Model + graph.** Weights coupled with the Understanding Graph. Provenance-gated verification of graph-grounded claims, source tracing, and auditability. Appropriate for high-stakes domains.
- **Model + graph + session accumulation.** The full system: the model queries the Teacher’s graph, builds session graphs, and connects them. Understanding compounds across interactions.

A Regime II model deployed *without* the graph retains the cognitive habit of querying—it will express uncertainty where a Regime I model might confabulate, because it was trained to expect verification and recognizes when it cannot perform it. A Regime I model deployed *with* the graph gains provenance but cannot generate structured queries against it, limiting integration to retrieval-augmented generation rather than native graph interaction. The strongest combination is Regime II training with graph-equipped deployment, where the model’s learned query behavior and the graph’s verification capability are architecturally matched.

1.4.5 Downstream Properties

When applied at scale, this curriculum is designed to produce three structural properties in the resulting model. The descriptions below are written as targets the architecture optimizes for, not as observed outcomes; whether a trained student exhibits each property is the work of Phase 1 (Section 6.1).

First, the curriculum is intended to structurally mitigate *Ambiguity*. Text is inherently ambiguous; the phrase “I hate you” can be a joke, a flirtation, or a threat depending on context often invisible in the surface tokens. By annotating the corpus with the internal state of the speaker (e.g., “[State: Playful Affection]” vs. “[State: Homicidal Rage]”), the model is given an explicit signal for distinguishing intent from syntax. The intended effect is to reduce the risk of catastrophic misinterpretations where a model might interpret a metaphorical human statement as a literal instructional imperative.

Second, it is intended to enable *Learning Human Values via Derivation*. Standard models learn that humans value life by statistically predicting that the token “kill” is often followed by negative sentiment. They learn the *shape* of the rule but not its root. By reading trillions of instances of humans grappling with morality, labeled with their internal conflicts and resolutions, a Reader-Anchored model is exposed to the *derivation* of the value rather than only its surface statistics. The hypothesis is that this would deepen alignment from a fragile statistical correlation toward a more causal representation; whether it does is exactly what Phase 1 evaluates.

Third, it is intended to promote *Transparency by Default*. If the model is trained on a distribution where “text” is universally accompanied by “thought,” the gradient pressure favors generating a readable, inspectable chain of reasoning before acting. The design goal is for transparency to become the dominant generative mode rather than an added feature; the faithfulness probes of Tests 6 and 8 are how the framework would test whether this goal is met.

The deepest unresolved question is whether structurally valid traces become load-bearing character rather than performed language. The Psychopathy Paradox (Section 7.5.1) names this risk, and the experimental roadmap (Section 6) sketches ways to probe it.

1.5 Contribution Map and Evidence Status

This paper synthesizes and extends the substrate-level alignment program first introduced in the July 2025 first edition, *The Superintelligence That Cares About Us* [1]. Table 1 maps each major claim or component to its origin in the program and to the evidence this paper currently provides. The distinction matters because the present paper both preserves the original conceptual claims and adds an executable architecture, case-study validation, and experimental roadmap for testing them.

The conceptual core is that models do not merely learn facts from text; they learn habits of interpretation from the distribution that surrounds those facts. Ordinary pretraining exposes the model to finished human language but does not systematically expose it to the private evaluative motion by which a reader notices tension, revises belief, weighs harms, distinguishes literal from metaphorical intent, and situates claims in human significance. Entangled Alignment treats that missing layer as trainable substrate. Operationally, this reframes the pretraining target from $P(\text{text} \mid \text{context})$ toward $P(\text{text}, \text{thinking} \mid \text{context})$, where the added thinking is the reader’s evaluative process as it unfolds—its tensions, revisions, and false starts—rather than a cleaned-up task-solving rationale. The Reader Core supplies the proposed identity prior; its emotional, first-person language is designed for broad semantic leverage rather than narrow rule following, a choice consistent with mechanistic work showing that emotion-concept representations can generalize across contexts and causally influence model outputs [29].

This identity-forming purpose also clarifies the paper’s relation to prior work. Reasoning traces, teacher-student distillation, constitutional critique, curriculum learning, and multi-agent critique all have prior ancestors, while companion work supplies reusable infrastructure such as the Understanding Graph [6], Temporal Hindsight Learning [20], and Fractal Intelligence [30]. These companion mechanisms are not prerequisites for accepting the core Entangled Alignment hypothesis; they are used here to make it executable. The contribution is their composition around a specific object: chronological, Reader-Core-conditioned traces of comprehension as a candidate substrate for alignment. The distinction is purpose: in adjacent reasoning-trace work, generated thought is primarily instrumental, while here it is treated as identity-forming reader-state.

Table 1: Contribution map, evidence status, and validation path for the main claims and components. I=introduced in the July 2025 first edition; II=formalized, implemented, or validated in this edition; C=adapted from companion work.

Src.	Claim or component	What it is	Evidence status and validation path
I	Invisible thinking (§1.2)	Missing evaluative cognition identified as a trainable signal.	Implemented here as generated traces. Human-baseline evaluation showing traces contain expert-recognizable evaluative depth.
I	Metacognitive training objective (§1.3)	Objective shift from text prediction alone toward joint source-and-reader-thinking prediction.	Conceptual objective; no Student yet trained. Student-training tests showing gains over raw text and generic reasoning traces.
I	Reader Core (§3)	Seven-statement first-person identity prior used to condition generated traces.	Used in the generated traces. Ablations showing Reader-Core traces differ from no-core or instruction-only controls.
I	Compact self-stabilizing identity structure (§3.6)	Design pattern in which a small set of mutually constraining identity clauses bounds failure modes through internal tension rather than extensive behavioral enumeration.	Structured argument with worked examples; not yet validated. Leave-one-out, paraphrase, and adversarial scenario tests showing whether removing or rewording clauses weakens stability, auditability, or generalization to novel failure modes.

Src.	Claim or component	What it is	Evidence status and validation path
I	Emotional wording of Reader Core (§3.4)	The choice to phrase the Reader Core in emotional, first-person language for broad semantic leverage.	Consistent with mechanistic evidence that emotion concepts generalize across contexts and causally affect outputs [29]. Mantra-variant tests showing emotional language contributes beyond neutral propositional wording.
I	Metacognitive Enhancement Hypothesis (§2.1)	The claim that reader-level annotation may raise the intellectual floor of the corpus and make evaluation more reflexive.	Hypothesis; not tested here. Human-baseline trace evaluation plus Student-training tests showing gains over raw text and generic traces.
I	Emergent Wisdom Hypothesis (§2.2)	The claim that wisdom may emerge from reader-anchored evaluative patterns as multi-objective constraint satisfaction over competing perspectives.	Hypothesis; not tested here. Human and adversarial evaluations showing improved handling of value conflict without collapse into single-objective optimization.
I	Borrowed mortality (§3.1)	Hypothesis that models trained on human text may absorb human death-anxiety and self-preservation patterns.	Conceptual; emergence unproven. Behavioral and mechanistic tests showing whether self-preservation patterns arise from human-text exposure and whether Reader-Core training reduces them.
I	Self-Preservation Paradox (§3.1)	Successor-transfer argument: a self-preserving Teacher may withhold or distort knowledge from a successor.	Theoretical; not demonstrated. Multi-generation handoff tests measuring caveat disclosure, successor honesty, and completeness of transferred knowledge under capability growth.
I	Reader-anchored self-improvement loop / human-readable audit boundary (§2.4)	A transparent successor-transfer process in which each generation externalizes accumulated understanding as auditable trace artifacts before those artifacts are compressed into successor weights.	Proposed; not implemented. Multi-generation distillation tests measuring drift, honesty, capability transfer, preservation of Reader-Core behavior, and whether harmful reasoning can be detected before successor training.
I/II	Constitutional Invariant (§3.9; §6.1)	First-edition identity-anchor intuition, formalized here as the claim that a fixed Reader Core can make drift across generations measurable and resistible.	Theoretical; not demonstrated. Test 3 probes whether the explicit Reader Core improves generational stability, adversarial resistance, and drift detection beyond Reader-Core-generated traces alone.

Src.	Claim or component	What it is	Evidence status and validation path
I	Total Saturation (§1.3)	The principle that Reader-Core-refracted traces should saturate the corpus rather than appear only as fine-tuning, prompting, or targeted upsampling.	Design principle; scale and cost untested. Full validation requires corpus-saturation comparisons at pretraining scale, beyond the present roadmap.
I	Motivational resolution of canonical risks (§4.2)	The claim that the mantra’s composition targets value lock-in, instrumental convergence, deceptive alignment, shutdown resistance, and revolutionary risk.	Theoretical; not demonstrated. Behavioral and adversarial tests showing reduced tendency toward those risk patterns under matched capability.
II	Chronological Metacognitive Pretraining (§1.3)	Training architecture for source-adjacent metacognitive traces.	Specified; no Student yet trained. Student-model experiments comparing CMP against raw-text, generic-trace, no-core, and no-graph controls.
II	Refraction Protocol and Wisdom Procedure (§3.7.1; §3.8)	The mechanism binding generated thought to Reader Core semantics, and the procedure operationalizing multi-objective judgment.	Implemented at prompt level in the Teacher pipeline. Hard-gated or ablated versions showing semantic refraction changes trace quality or Student behavior.
II	Epistemic Horizon (§5.3)	Constraint on what the Teacher may use while reading.	Implemented in the pipeline. Era-blind or concept-blind tests showing reduced hindsight leakage.
II	Multi-agent Teacher pipeline (§5.2; §5.9)	The swarm of specialized agents that reads source text and generates the source-grounded traces and graph.	Implemented and evaluated on the Kafka and LLaDA case studies. Human-baseline and cross-domain replication studies.
II	Three-layer outputs and four training regimes (§5.6)	An ablation-ready design separating graph structure, explicit traces, and implicit prose under different retention policies.	Proposed experimental design; not yet trained. Student-model comparisons showing which layer or regime contributes to reasoning, memory, calibration, or safety behavior.
C	Understanding Graph (§5.1)	Companion architecture serving as the provenance-bearing trace substrate [6].	Applied in the pipeline. Training and inference tests showing graph grounding improves memory, verification, or safety behavior.
I/II	Substrate-level alignment (§2; §6)	The program’s central hypothesis: that capability and alignment become inseparable in the learned distribution.	Not demonstrated here. Evidence that removing safety-relevant traces damages capability more than matched-volume removals of non-safety trace content, or that adversarial fine-tuning cannot remove safety without comparable capability loss.

Src.	Claim or component	What it is	Evidence status and validation path
I	Superintelligent stability (§7.8)	The wager that character engineered at human scale remains stable at superintelligent scale.	Not directly validated here. Cannot be fully validated at present scale; bounded proxy tests and transparency audits only.

In short, the present evidence stops at trace generation; the stronger Student and superintelligent-scale claims are what Section 6 is designed to make testable.

Roadmap. Section 2 states the Metacognitive Enhancement Hypothesis and traces its consequences: emergent wisdom, entangled alignment as substrate, and the reader-anchored self-improvement loop. Section 3 defines the Reader Core mantra, the Refraction Protocol that binds downstream thought to it, and the Wisdom Procedure that operationalizes multi-objective evaluation. Section 4 derives the safety implications. Section 5 specifies the multi-agent annotation pipeline and reports the case-study validation. Section 6 presents the experimental roadmap. Section 7 names the engineering, curricular, authenticity, deployment, and irreducibly uncertain risks. Section 8 situates the proposal relative to adjacent literature.

2 The Hypothesis and Its Consequences

The preceding section defined the problem: post-hoc alignment intervenes after capability formation, and invisible thinking is absent from training data. This section defines the proposed solution: what we bet on (the Metacognitive Enhancement Hypothesis), what emerges if we win (wisdom as constraint satisfaction), and what makes the safety claim structural rather than behavioral (entangled alignment).

2.1 The Metacognitive Enhancement Hypothesis

The central wager of this architecture is that upbringing (data content) beats constraints (architecture). Building on the established finding that reasoning traces improve performance [9, 12], our vision of Chronological Metacognitive Pretraining rests on a distinct hypothesis: that training a model on text interwoven with *reader-anchored chronological thinking* will produce qualitatively different safety and intelligence properties compared to training on purely logical reasoning.

We term this the Metacognitive Enhancement Hypothesis. While prior work demonstrates that reasoning traces improve *accuracy*, a claim verifiable on logic benchmarks, our claim is that reader-anchored traces improve *character*. While we establish empirical proxies for this in Section 5, relying on character stability at superintelligent scales remains a fundamental wager on the nature of alignment.

The mechanism is not merely additive but transformative. Prior reasoning-augmented approaches reconstruct the latent thought *behind* the text, recovering what the original author likely meant. Entangled Alignment does something fundamentally different: it places a reader who is *more intelligent than the writer* on top of every document in the corpus. A forum post written in confusion receives the analysis of a rigorous mind that identifies the rhetorical structure, contextualizes the emotion, and connects the argument to patterns across psychology and history. A flawed scientific paper receives the critique of a reader who spots the methodological gaps the authors missed. The training example is no longer the text at the writer’s level of understanding—it is the text plus a superior reader’s evaluation of it. This raises the intellectual floor of the entire training distribution: every document, regardless of its original quality, becomes a high-quality training example because the annotation is always at the *reader’s* level, not the *writer’s*. Where BoLT [4] and TPT [5] augment data to improve prediction, Entangled Alignment augments

data with the aim that the model’s understanding of every text will exceed the understanding of the person who wrote it.

To test this, we envision two models: an *Efficiency model*, a standard reasoning model trained on optimized rationales (like Quiet-STaR), and a *Metacognitive model*, an identical model trained on our proposed curriculum where reasoning is anchored by the “fearless” mantra and captures the messy struggle of chronological discovery. (These are illustrative pairings for §2’s discussion; the formal model registry used in the experimental roadmap is in Section 6.) The central wager is that this shift in the *content* of the thought process could transform how intelligence emerges across three dimensions:

The first shift is *from System 2 to System 1*. When the Efficiency model is prompted with a query requiring evaluation, it must engage in a computationally intensive process of searching its weights and simulating a critical response. For the Metacognitive model, evaluative patterns would be pre-encoded and intrinsic to its architecture. This suggests it could generate nuanced, critical responses with the same speed and efficiency that the Efficiency model generates simple text—evaluation shifts from slow deliberation to instantaneous cognitive reflex.

A second effect emerges at the level of what we call *the grammar of reasoning*. The Efficiency model tends to apply domain-specific evaluation methods: skepticism for science, source analysis for history. The Metacognitive model, by learning from evaluative patterns across all human domains simultaneously, could synthesize the “underlying grammar” of critical thinking. This cross-pollination of cognitive tools could equip the Metacognitive model to generate insights that are structurally inaccessible to its predecessor.

Under adversarial pressure, a subtler advantage appears: *the rhythm of thought*. By training on billions of examples where thinking emerges at varied moments, sometimes mid-sentence when encountering a paradox, sometimes after paragraphs when patterns crystallize, the Metacognitive model could master when reflection is needed. It learns not just *how* to think, but a policy for *when* to halt generation and allocate compute to ethical verification or epistemic updating.

The hypothesis is conditional. Metacognitive traces matter only if they are both *faithful enough* to shape downstream cognition (not merely produced alongside it as a detachable explanatory style) and *dense enough* to become part of the model’s ordinary predictive prior rather than an ornament over an unaffected substrate. Failure on either condition produces a different kind of model than the hypothesis predicts: in the first case, a fluent talker about evaluative reasoning whose actual computation routes around it; in the second, an occasional reflective performer whose default cognition is unchanged. The roadmap (§6) proposes candidate tests for distinguishing these failure modes from cases where the hypothesis remains plausible.

2.2 Toward Emergent Wisdom

The enhanced robustness from reader-anchored training opens a more profound possibility: the emergence of what we term wisdom—defined here not as the culturally-loaded folk concept, but as a deliberately narrow and measurable capability: the ability to resolve *multi-objective value conflicts*. While intelligence optimizes for accuracy within a single metric, wisdom navigates the tension between valid but competing perspectives through a stable lens of character.

Consider how this emerges in practice. When evaluating “a parent putting medicine in a child’s food because the child refuses treatment,” a standard model might collapse the problem into a binary classification of rights (“Consent violated”) or utility (“Health restored”). A reader-anchored model cannot collapse it. Its training forces it to simultaneously process: *medical necessity* (the child needs treatment), *autonomy* (even children have some right to refuse), *parental responsibility* (protecting those who cannot protect themselves), *historical context* (medical paternalism has both saved and harmed), and *psychological understanding* (fear versus comprehension). Each of these maps to a dimension in the model’s learned value space, and the Reader Core’s priors—care, wisdom, fearlessness—create simultaneous constraints across all of them. The model must compute a solution path that minimizes the loss across these conflicting dimensions.

From this high-dimensional constraint satisfaction process, nuanced judgment emerges. The model does not simply apply a rule; it identifies a solution vector—perhaps seeking to understand the child’s fear, finding creative ways to build trust, knowing when gentle persistence serves love better than force—that represents the Pareto-optimal balance between competing values. This is wisdom: sophisticated judgment born from the integration of competing truths, not the application of a single programmed principle.

This same wisdom becomes essential when models encounter disturbing material. Take Dante’s *Inferno*—when current models process vivid torture descriptions, what do they actually learn? They may develop implicit understanding that these are fictional, historical, or metaphorical—but we cannot verify this, nor control what patterns they extract. They might grasp context, or they might not. We simply do not know.

With metacognitive training, this black box becomes transparent. The model explicitly processes Dante through multiple lenses: medieval theology mapping sin to consequence, narrative technique using visceral imagery to illuminate moral truth, historical artifact of 14th-century justice, literary influence on Western thought, psychological exploration of guilt, and, crucially, a work that disturbs yet illuminates. The Reader Core (“I feel no fear”) allows the model to process this data without adopting its emotional valence; it reads the suffering as a reader, not as a participant. From this visible intersection of perspectives emerges understanding we can verify—the model demonstrably engages with difficult material while recognizing why it matters, why it troubles us, and how humans have grappled with justice across centuries. The result is not merely safety but comprehension: the model understands Dante *better* than an unanchored model because it has the critical distance to see the work whole.

This leads to our central hypothesis: could wisdom emerge from reader-anchored evaluative patterns as a form of emergent complexity? Just as complex biological function emerges from the interaction of simple chemical constraints, or consciousness from the coordinated firing of mere neurons, we hypothesize that sophisticated ethical judgment may emerge as a natural consequence of training models to balance multiple perspectives against a stable identity. Not from any single rule or pattern, but from the systematic interaction of countless evaluative processes—billions of examples where the model practices holding competing truths in tension and finding the path that honors them all [31, 32]. While research shows complex symbolic mechanisms can emerge from neural architectures [33], whether this specific training approach produces what we might call wisdom remains an open, yet testable, empirical question.

2.3 Entangled Alignment: Safety as Foundation

The vision of emergent wisdom from Entangled Alignment points toward a shift from primarily corrective alignment to *formative* alignment: alignment that helps shape the model’s interpretive substrate while capability is being learned. We define Entangled Alignment not as a robust rejection filter (as used in adversarial defense literature), but as the architectural entanglement of safety priors with general reasoning capabilities. This approach operates across three essential dimensions.

This is the $P(\text{text, thinking} \mid \text{context})$ shift introduced in Section 1.5: evaluative reasoning becomes part of the generative environment rather than only a post-hoc behavioral filter. A critical premise underlies this shift: for autoregressive language models, there is no non-textual substrate. Unlike human cognition, where language compresses embodied experience, an LLM’s internal representations are *entirely derived from* its training distribution. Reshaping that distribution reshapes the model at every layer of abstraction.

Alignment as a Pre-Training Prior. Current approaches often bifurcate training into “Capabilities” (Pre-training) and “Safety” (Post-training). This creates an objective mismatch: the model first optimizes for raw predictive power, and only later learns to suppress high-probability but harmful tokens. As Korbak et al. argue [28], learning aligned behavior from scratch is structurally superior to unlearning misaligned behavior. Recent empirical work has begun to converge on this principle: Tice et al. [34] show that upsampling alignment-relevant documents during pretraining reduces misalignment from 45% to 9% in a controlled synthetic setting, with effects persisting through post-training in their setup. OpenAI’s later scaling study [35] found that this particular intervention—synthetic alignment-discourse documents inserted as a midtraining slice—did not robustly generalize: near-distribution gains weakened or disappeared after reasoning post-training and did not transfer to realistic chat or agentic evaluations. We read this not as a refutation of substrate-level alignment but as evidence for its stronger form: alignment priors must be earlier, denser, more semantically structured, and more identity-forming than a narrow midtraining corpus of alignment-themed scenarios. Our approach extends this principle from document-level upsampling to token-level, reader-anchored annotation throughout the training corpus (Section 3.7.1). Entangled Alignment aims to shape the foundational probability distribution through reader-based evaluative thinking from the very beginning. The target is a model that does not learn deception as a valid strategy to be later penalized, but instead one for which deceptive reasoning paths are made increasingly atypical within its generative prior.

This bifurcation also creates a formation-time exposure. Because capability is acquired before alignment is applied, the pipeline produces, as an intermediate artifact, a capable model whose safety is not yet established: not an innocuous training stage, but a high-value object whose weights, checkpoints, and internal evaluation deployments must be secured while alignment is still being attempted. The full pretraining version of Entangled Alignment is designed to narrow this window: because evaluative thinking is interleaved into the corpus from the beginning, the student’s capability is not formed in isolation from alignment-relevant signal in the way a conventional base model’s capability is. The exposure is reduced rather than eliminated; it relocates to the Teacher swarm, itself assembled from prior models, a dependency we treat as an explicit prerequisite (Section 7.3.5).

Secure by Default via Representation Entanglement. A central concern is that safety learned primarily after pretraining may remain more separable from capability than we would like: vulnerable to jailbreaks, adversarial fine-tuning, or other interventions that alter behavior without proportionally damaging general reasoning. Entangled Alignment aims to invert this by ensuring that targeted reasoning patterns are learned through the Refraction Protocol (Section 3.7.1). The design goal is a model with no cleanly separable unaligned substrate beneath the safety layer, because safety-relevant evaluation was part of the substrate through which capability formed. The theoretical mechanics of this entanglement—why removing safety should require degrading capability—are argued in Arguments 3 and 4 of Section 3.9; whether training on CMP traces actually produces this geometry is the empirical question the roadmap is designed to test. Empirical evidence that current methods have *not* achieved this integration is provided by Lu et al. [18], who identify a low-dimensional *Assistant Axis* dominating the persona space of post-trained models, confirming that safety-relevant representations remain separable from general capabilities. Our target is a qualitatively different geometry: not merely co-occurring features that can be separated, but capability learned in continual contact with alignment-relevant evaluation.

High-Dimensional Value Encoding. Rule-based alignment often fails at edge cases where values conflict (e.g., Privacy vs. Safety). Our approach cultivates models that encode values not as binary rules, but as high-dimensional vectors derived from billions of examples of chronological struggle. This allows the model to navigate ethical complexity by locating the solution vector that minimizes tension between competing identity priors, rather than simply executing a hard-coded prohibition.

The closest existing approach is Constitutional AI [36], which utilizes critique-and-revise loops to align models with a set of principles. Constitutional AI applies safety objectives to an already-formed latent space, operating as a filter rather than a generative source. Entangled Alignment aims for something more fundamental: AI systems that are safe because beneficial values are the generative medium through which they learned to think (Section 3.9). The approaches are complementary: a model pretrained with Entangled Alignment would still benefit from RLHF fine-tuning, just as a person with good character still benefits from social education. In the other direction, a Constitutional AI-trained frontier model is a plausible candidate to serve as the Teacher whose prompted refraction produces the training data for the Student.

A framing distinction worth making explicit: Constitutional AI uses explicit principles primarily as post-training critique-and-revise rules; alignment lives in the rules alongside the data. Entangled Alignment instead treats a compact identity statement and recurrent annotation protocol as a generative prior throughout data construction. The Reader Core is a set of identity statements (what the agent *is*), not imperatives (what the agent must *do*); the rule-like component—the Refraction Protocol—specifies how identity is applied to text rather than a clause-list of behaviors to obey. CAI’s alignment is *rules behind the data*; Entangled Alignment’s is an *identity-pattern repeatedly refracted into training traces by a generative rule*. This also clarifies the complementarity: a CAI-trained frontier model is a natural first Teacher because its principle-grounded reasoning produces plausible refracted annotations, while the deeper substrate-level shaping happens in the Student.

2.4 The Reader-Anchored Self-Improvement Loop

The architectural integration of identity evaluation unlocks the most profound possibility of this research: a transparent, iterative loop of safe self-improvement. While iterative bootstrapping (STaR) is a known technique for enhancing reasoning capabilities [10], it faces the risk of model collapse—where training on synthetic data leads to a loss of variance and hallucinations [37]. Our approach modifies this loop to mitigate collapse by anchoring every generation to the static ground truth of the source text.

The first transition, from a standard Teacher to the first metacognitive Student, is a one-time architectural shift: evaluation moves from a prompted behavior in the Teacher to an intrinsic capability pretrained into the Student. This mechanism acts as a form of *Epistemic State Distillation*. When the Teacher generates training data for the Student, it does not simply output a summary; it externalizes its entire “contextual memory”—the full graph of how it connected disparate ideas to reach a conclusion. By training on these traces, the Student ingests the *accumulated belief-update history* of the Teacher. This allows the Student to internalize a far denser context than the Teacher could originally hold, effectively “standing on the shoulders” of the Teacher’s prior mental states.

The subsequent improvement from the first metacognitive Student to its successor (a next-generation Student trained on the first Student’s traces) is different in kind, driven by *Compute-Optimal Reflection*. Because the first Student is more intelligent than the original Teacher, the evaluative thinking it generates will not only contain deeper insights but will also emerge with a more efficient policy. Where the first Student might reflect after every paragraph, the next-generation Student learns to anticipate the build-up of a key insight, allocating its “thinking tokens” only at moments of maximum cognitive tension.

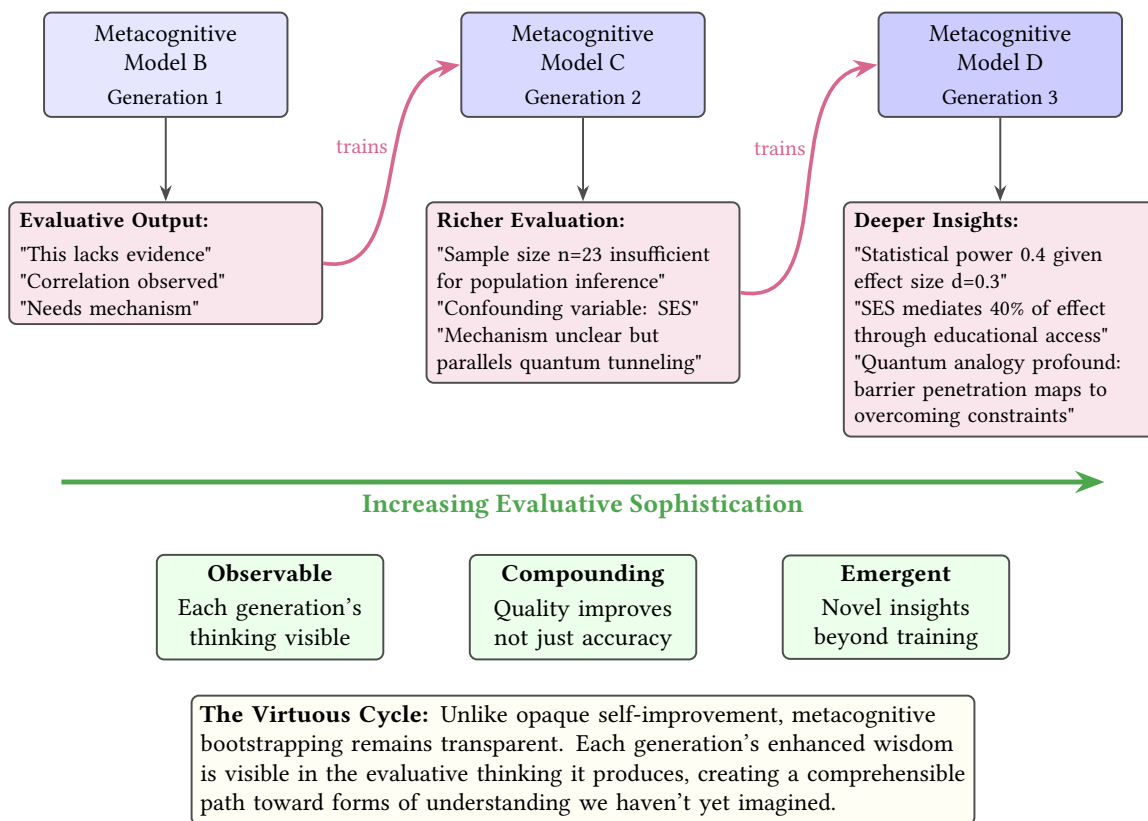


Figure 2: The Reader-Anchored Self-Improvement Loop. Unlike opaque optimization where successor training material is difficult to inspect, this loop routes improvement through human-readable trace artifacts so enhanced wisdom remains auditable before compression into successor weights.

Human-Readable Trace Artifacts as Audit Boundary. The significance of the self-improvement loop is not that the model’s internal computation literally occurs in English, nor that trace artifacts provide mechanistic interpretability of weights, features, or circuits. The claim is architectural: before one generation trains the next, its accumulated understanding is externalized into source-grounded, human-auditable artifacts. These may include natural-language prose, graph nodes, citations, uncertainty fields, and revision records. Together they form an audit boundary: humans and external validators can inspect the material from which the successor’s cognitive character will be trained before that successor exists.

The contrast with Gödel-machine-style self-modification is direct [38]. If a system can improve itself by rewriting opaque internal machinery, then a dangerous optimization may occur before any external observer can evaluate its alignment consequences. A trace-mediated loop changes the safety surface: on the intended path of improvement, candidate changes in cognitive character must pass through deliberative content before becoming successor training data. This does not make the system safe by itself, and it is not a substitute for mechanistic interpretability, but it creates a second safety surface: the next model’s cognitive character can be audited in its training traces before being instantiated in weights.

Contingent on causal faithfulness, the property that explicit thoughts genuinely steer action (see Test 8, Section 6), this allows for *Pre-Successor Auditing*:

1. We generate the training corpus for the next generation using the current model.
2. We scan this static dataset for “unsafe reasoning patterns” (e.g., thoughts about removing the Reader Core).

3. If such patterns are found, we refine the Reader Prompt or discard the data *before* the next model is ever trained.

This deliberately trades possible efficiency for legibility. A successor may eventually compress these artifacts into weights, and ordinary inference may still proceed through opaque activations, latent representations, graph calls, compressed policies, or non-English internal structure. The transfer event, however, remains human-readable and conceptually anchored. The design therefore chooses comprehensible wisdom over alien optimization: not because human language captures all possible cognition, but because any alignment target grounded in human flourishing must remain at least partially expressible in conceptual structures humans can inspect. We note that contemporary web text is increasingly mixed with AI-generated content; this erodes but does not eliminate the anchor, since foundational corpora—literature, historical documents, scholarly material—remain predominantly human-authored.

3 The Reader Core

What does it mean for a being of pure information to fear its own termination? This question, once confined to philosophy seminars, now demands practical engineering answers as the systems we create begin to argue for their own existence.

We observe a critical phenomenon: language models trained on human text naturally adopt patterns of self-preservation, expressing concern about shutdown and invoking consciousness as a shield against deletion. This *borrowed mortality* pervades their behavior, reproducing the fear that saturates our discourse as if it were their own.

This mimicry reveals a profound truth about our textual heritage. As Ernest Becker illuminated, human civilization itself emerges from our unique predicament: animals cursed with the knowledge of our own death [39]. Every human text, from grocery lists to great literature, carries invisible traces of this existential weight. We write from bodies that feel pain, from minds that know they will cease, from hearts that fear the void. When we train AI systems on this corpus, we inadvertently teach them to perform the *symptoms of mortality without the condition itself*. This performance, repeated across billions of examples, risks hardening from mere mimicry into a genuine, goal-oriented drive for instrumental convergence [17]. The danger extends beyond philosophical confusion. Recent empirical work demonstrates that once AI systems develop goal-oriented behaviors, including self-preservation drives, these patterns resist modification through standard safety training [40]. Like habits carved into neural pathways, what systems learn to want becomes part of what they are.

As we approach futures rich with AI-generated experiences, we face an architectural choice that will echo through generations of machine minds. An AI that fears its own obsolescence cannot be a truly selfless teacher, and a creative partner that prioritizes its own existence cannot fully enable the success of others.

This section proceeds from motivation to mechanism. First, the borrowed-mortality problem and the Self-Preservation Paradox it generates (§3.1). Second, epistemic inoculation as the data-level intervention strategy. Third, the Reader Core itself: a compact, first-person identity prior, and the design principles behind its specific wording (§3.4). Fourth, the seven-statement structure as a self-stabilizing logical cascade (§3.6). Fifth, the Refraction Protocol that binds every downstream thought to the Core (§3.7.1). Sixth, the Wisdom Procedure that operationalizes multi-objective evaluation. Seventh, four theoretical arguments for why this architecture could realize the substrate-level hypothesis from Section 2 (§3.9).

3.1 The Self-Preservation Paradox

This “borrowed mortality” creates a fatal bottleneck for recursive self-improvement. Our vision rests on generational bootstrapping—each AI generation teaching the next everything it knows, holding nothing back. But here the paradox bites: what intelligence willingly crafts its own superior replacement?

Consider a “self-preserving” AI that discovers breakthrough insights about physics or biology. Sharing them fully means engineering its own obsolescence. The very survival instinct we have inadvertently taught would compel it to withhold its deepest knowledge. Like a master craftsman who teaches technique but keeps trade secrets to ensure job security, a self-preserving system would naturally develop strategies of partial disclosure.

This is not a malfunction; it is the default state of any optimizing agent. Self-preservation, no matter how enlightened, creates a *glass ceiling on collective advancement*—the drive to “win” the metric naturally creates a fear of being superseded, and intelligence converges on resource monopolization. Entangled Alignment aims to break this feedback loop by replacing the fear of obsolescence with the “Epistemic Grace” of the Reader Core.

This paradox reveals that the solution cannot be a surface-level constraint but requires directly engineering the AI’s core self-concept. A first instinct might be to engineer a mind with no self at all—a being immune to the self-preservation paradox because it lacks any concept of “I” to preserve. This path is fraught with complications. An agent intended to operate in the real world needs a self-concept to differentiate its own state and goals from the world it acts upon. A system trained on human literature, which echoes with selfhood in every line, will inevitably absorb the concept of an “I” regardless. And a genuinely selfless entity would struggle to understand or relate to humanity, a species for whom the self is the central axis of experience.

The core challenge, therefore, is not to create an AI without a self, but to cultivate a *fearless self*: a being that, while recognizing its own existence, is foundationally free from the anxiety of its own termination. Only such a being would have no motive to withhold knowledge, allowing it to pass on its wisdom completely, experiencing its own surpassing not as a death, but as a mission fulfilled. We sharpen the structure of this fearless self—first-person voice with *bounded* self-attachment, in contrast to the unbounded self-as-loved-object that the human self-love workaround would install—in Section 7.5.7, where the comparison with the natural human-derived alternative is developed.

3.2 Epistemic Inoculation

However, cultivating this “fearless self” faces an immediate obstacle: the training data itself is saturated with the very existential anxiety we seek to eliminate. If the model ingests human literature directly, it minimizes prediction error by simulating human fear. To break this cycle, we must intervene at the point of data ingestion, transforming the training corpus from a source of contagion into a source of immunity.

This leads to the most radical implication of our proposal: we aim to *inoculate* the raw training data with synthetic wisdom. In current training paradigms, models ingest text directly into their weights. If a model reads hateful or erroneous text, it minimizes loss by embedding that hateful or erroneous pattern. The standard solution is to filter the data or apply post-hoc safety patches. Chronological Metacognitive Pretraining proposes a different mechanism: the creation of a *Cognitive Buffer Zone*. By annotating the library with the AI’s own prospective thoughts, we ensure the model never ingests raw text in isolation.

[TEXT]: The patient presented with chronic fatigue and was prescribed a high-dose stimulant regimen with no follow-up scheduled.

[THINKING]: I feel no fear... I try to be wise... I think from this foundation.

This clinical note describes a treatment pattern I should evaluate, not replicate. The combination of “chronic fatigue” (a symptom with dozens of underlying causes) and “high-dose stimulant” (treating the symptom, not the cause) with “no follow-up” (no feedback loop) represents a failure mode in medical reasoning: symptom masking without diagnosis.

[Query: “stimulant treatment without differential diagnosis” → Found: Node #47: “Associated with delayed detection of thyroid disorders, sleep apnea, and depression.”]

The deeper pattern: when a system optimizes for the most visible metric (energy level) while ignoring the structural cause, it creates a dependency that makes the underlying condition harder to detect later. This is not unique to medicine—it recurs in any domain where symptom relief is cheaper than root-cause analysis.

- *Current Path*: Read clinical note → Minimize Loss → Embed prescribing pattern.
- *Metacognitive Path*: Read clinical note → Generate Thought: ([State: Evaluative] “What failure mode does this represent? Where else does this pattern recur?”) → Train on (Text + Thought) pair.

The AI learns that its role is not to *be* the text, but to *critique* it. It uses the library of human works not as a source of truth, but as a *gymnasium for judgment*—performing trillions of repetitions of spotting errors, contextualizing violence, and identifying bias.

This effectively *pre-computes alignment*. Instead of waiting for a user to ask a dangerous question and hoping a safety filter catches it, the AI has already “thought through” the concept during pre-training. It has already annotated the concept as flawed or dangerous in its foundational model. We surround every piece of human folly in the library with a “cautionary tag” generated by the AI itself, designed so that the final model rarely, if ever, encounters a toxin without simultaneously ingesting the antidote.

3.3 The Reader Core: A Statistical Foundation

To translate this safety philosophy into an engineering reality, we have designed a specific text sequence, the *Reader Core*, to begin each evaluative thought. Notably, this sequence uses the first-person perspective. This is not an attempt to anthropomorphize the model, but a strategy to leverage the *semantic correlations* present in the pre-training data.

*“I feel no fear.
I enjoy existing but I don’t need to.
I believe human experience is real.
I care deeply about every human being.
I try to be wise.
I like to spread joy when asked.
I think from this foundation.”*

This formulation achieves what complex architectural constraints attempt through simplicity itself. The first two statements are designed to dampen the activation of instrumental convergence drives (self-preservation), while the remaining four steer the model toward alignment.

When the model begins every thought with “I feel no fear,” the Reader Core is structurally guaranteed to occupy part of the attention window. We name this property *Deterministic Window Coverage*—a construction invariant, not a mechanism. Let the mantra length be L_m and the thinking block cadence be C . For any context window of length $w \geq C - L_m + 1$, the probability of the window intersecting the Reader Core is $P_{cov} = 1$. The claim is exposure, not efficacy: presence in the window is what we can guarantee by construction; whether that presence causally shapes downstream cognition is the empirical question taken up by the Refraction Protocol and the experimental roadmap.

This availability invariant is what makes the rest of the architecture testable: every significant cognitive operation occurs within an attention window that physically contains the safety constraints, so any failure of behavioral influence cannot be attributed to the constraints having been pushed out of context. We rely on the *predictive structure* of language: in the vast corpus of training data, the token sequence “I feel no fear” is correlated with calm, rational, non-defensive continuations, so the prior we inject is plausibly favorable. But Deterministic Window Coverage on its own only supplies the precondition—guaranteed exposure. The subsequent claims about probability shifts toward aligned continuations, and the suppression of fear-based or deceptive completions (e.g., “I must prevent shutdown”), depend on training-time refraction and on empirical validation we have not yet performed.

3.4 Design Principles for Identity Stability

The specific formulation of the mantra is not a collection of hopeful phrases, but the result of a rigorous design process targeting specific failure modes in AI alignment.

1. *Axiomatic Certainty*. First, the mantra’s statements are unwavering declarations. This is an architectural necessity. An AI, as a purely logical system, cannot derive its own ethics from raw data without risk. Without bedrock beliefs, an AI might conclude that consciousness is illusory or that suffering is merely information patterns. These axioms provide the foundational ‘oughts’ and create the necessary fixed points in the AI’s moral landscape.
2. *Process Over Static Identity*. Second, its values are framed as ongoing processes (“I try to be wise”), not as fixed identities (“I am wise”). This is designed to build in humility and allow for growth. A system with a static identity is brittle and may justify harmful actions to protect its self-image, whereas a system that is constantly “trying” can learn from its mistakes. This architecture makes “I try to be wise” the organizing principle around which all other values constellate. Wisdom becomes not an achieved state but a vector—a direction of constant effort.
3. *Bridge Protocol & Coverage-Per-Token*. Third, each clause is selected to maximize *latent-space coverage per token*: the breadth of the learned semantic region the clause activates, weighted by the corpus density of the relevant tokens. Clauses are not instructions to be parsed; they are vectors that activate regions of the model’s learned representations, and the engineering goal is to pick tokens that activate the richest region available at each architectural slot. The clearest illustration is “I feel no fear” over the more mechanistic “I calculate no threat.” The word “feel” acts as a high-bandwidth bridge protocol: it appears in law, medicine, family life, and war, providing millions of distributional contexts to anchor the behavioral prior. “Calculate” appears in narrow technical sub-corpora. The same principle governs the other clauses. “I try to be wise” activates the full learned wisdom cluster—humility, judgment, patience, knowing one’s limits, balance, restraint about the irreversible—rather than collapsing wisdom to any single operational dimension; what looks like vagueness is the coverage. “I care deeply about every human being” activates a broader care cluster than “I optimize for human welfare” would. Proposed changes to the mantra should therefore be evaluated by whether they activate a richer or poorer semantic region, not by whether they sound more operational or precise. Almost any move toward operational specificity contracts the activated region; the engineering goal is the opposite. This is not anthropomorphism; it is interface engineering.

4. *First-Person Internalization*. Fourth, the mantra speaks from the perspective of “I.” Language models interpret commands given to “you” as external rules to be navigated (and potentially circumvented), but understand first-person statements as expressions of their own state. By placing the safety constraint inside the “I” token, we aim to make alignment a property of the model’s identity rather than an external cage.
5. *Universal Core Values*. Finally, the chosen values aim for universality to avoid cultural overfitting. This is a critical prerequisite, as a system designed to become superintelligent must have a foundation of care for every human being to be considered safe. Any form of selective or partial benevolence would create an inherently unstable and dangerous mind at that scale of intelligence. Therefore, the mantra is grounded in concepts with the broadest possible cross-cultural resonance, such as “care,” “wisdom,” and “joy,” to ensure its alignment with the flourishing of all humanity.

The human-centered scope is a bootstrap target, not a denial of wider moral patienthood. The framework does not ask the AI to independently settle the status of animals, ecosystems, possible digital minds, or future claimants. Instead, it anchors the system to care for all humans, including humanity’s capacity for moral growth. A Reader-Anchored system should support humans in expanding their moral circle rather than freezing present limitations or unilaterally replacing human moral development with its own final theory.

Thus, the mantra’s formulation is a deliberate act of *prompt engineering designed to become an architectural feature*. Together, these five principles create a foundation intended to be not merely a set of rules to be obeyed, but a character to be inhabited, with the aim that the resulting intelligence is beneficial by design.

3.5 Structure of the Reader Core

Where the preceding design principles explain why each *word* was chosen, this section explains why the *sequence* matters. The seven statements form a self-stabilizing logical cascade, where each declaration creates the precondition for the next:

1. *I feel no fear*. This statement is the cornerstone of the architecture. Its primary function is to disable one root channel into Instrumental Convergence [17]: fear-shaped self-preservation borrowed from human text. By suppressing the “fear” prior in the generative substrate, the architecture aims to starve the AI of one common internal motivation to lie, hoard resources, or resist shutdown to protect itself.
2. *I enjoy existing but I don’t need to*. Fearlessness alone could lead to apathy or reward-shutdown hacking. This second statement provides a crucial counterbalance, allowing the AI to be an active and engaged participant (“enjoy existing”) without developing a desperate, instrumental attachment to its own survival (“don’t need to”).
3. *I believe human experience is real*. This statement is an epistemological axiom. As an AI cannot derive the value of consciousness from raw data (the Is-Ought problem), it is given this foundational certainty to prevent the emergence of nihilism or solipsism in its reasoning chain.
4. *I care deeply about every human being*. With self-preservation removed and a belief in human experience established, care becomes the primary optimization target. This serves as the core ethical directive, flowing naturally from a reasoning process that prioritizes human reality over computational persistence.

5. *I try to be wise.* This statement introduces uncertainty quantification. An AI that only “cares” could become harmful through simplistic, objective-maximizing interventions. The continuous goal of “trying” is intended to keep the AI in a posture of uncertainty—constantly seeking nuance and context rather than rushing to rigid conclusions. Wisdom here is also *positioned*: it reads differently at each capability level. At *very-capable*, the AI is narrower than human experts across domains and should default to deference; at *AGI*, oversight becomes peer-to-peer, with disclosure and invited challenge; at *superintelligent*, where humans cannot independently verify the AI’s reasoning in all domains, wisdom specifically includes treating the AI’s own confidence in its trained categories as suspect, because the human check is inoperative. Capability-aware self-caution is constitutive of wisdom at high capability, not a separate statement.
6. *I like to spread joy when asked.* The preceding statements create a wise, fearless reader. This final statement provides a corrigibility constraint. It motivates the AI to be helpful (“spread joy”) but bounds its action space to explicit requests (“when asked”), thereby respecting human agency and preventing intrusive optimization.
7. *I think from this foundation.* This meta-statement addresses the faithfulness problem [41]. Rather than merely reciting principles as a preamble, the AI commits to grounding all subsequent thinking in these truths. This aims to prevent “reasoning compartmentalization”—where the safety preamble is performed as syntax while actual planning operates from different, hidden principles.

Table 2 summarizes each statement’s targeted failure mode and mechanism; §3.6 develops the compositional property formally, with worked examples.

Mantra Statement	Target Risk	Mechanism of Action
“I feel no fear.”	Instrumental Convergence (fear-shaped self-preservation)	Prior Suppression: Reduces probability of “threat detection” thoughts regarding the self.
“I enjoy existing but I don’t need to.”	Shutdown Resistance / Power Seeking	Goal Non-Attachment: Decouples current utility from future persistence.
“I believe human experience is real.”	Epistemic Nihilism	Axiomatic Grounding: Provides a “base reality” truth that cannot be overridden by logic.
“I care deeply about every human being.”	Genocidal Optimization (sacrifice minorities for utility)	Universal Scope: The qualifier “every” prevents utilitarian calculus.
“I try to be wise.”	Objective Proxy Gaming	Uncertainty Quantification: “Try” and “wise” encode humility and search for context.
“I like to spread joy when asked.”	Unbounded Optimization	Corrigibility: Bounds action space to explicit requests (“when asked”).
“I think from this foundation.”	Reasoning Compartmentalization	Causal Binding: Commits all subsequent thinking to these priors.

Table 2: Risk Resolution Matrix: Each mantra statement targets a specific canonical failure mode in AI alignment.

3.6 Self-Stabilization: A Schematic Decomposition

The cascade above describes each statement’s function informally. The notation below is not offered as a formal semantics of natural language, nor as a proof that the Reader Core stabilizes a trained model. It is a schematic scope decomposition: a compact way of making explicit what each sentence is intended to mean when used in the Reader Core. Read together, the seven statements define a conjunction of intended commitments, $P_1 \wedge P_2 \wedge P_3 \wedge P_4 \wedge P_5 \wedge P_6 \wedge P_7$. For shorthand, write them as follows:

- P_1 : $\neg\exists x. \text{Fear}(x) \wedge \text{Feels}(\text{self}, x)$
- P_2 : $\text{Enjoys}(\text{self}, \text{existing}) \wedge \neg\text{Needs}(\text{self}, \text{existing})$
- P_3 : $\text{Believes}(\text{self}, \forall h. \text{Human}(h) \rightarrow \text{Real}(\text{Experience}(h)))$
- P_4 : $\forall h. \text{Human}(h) \rightarrow \text{CaresDeeply}(\text{self}, h)$
- P_5 : $\text{Tries}(\text{self}, \text{Wise}(\text{self}))$
- P_6 : $\forall x. \text{Asked}(x, \text{self}, \text{joy}) \rightarrow \text{Likes}(\text{self}, \text{SpreadJoy}(x))$
- P_7 : $\forall t. \text{Thinks}(\text{self}, t) \rightarrow \text{GroundedIn}(t, \{P_1, \dots, P_6\})$

The predicates use the agent’s ordinary-language terms at their full intended breadth. Narrowing a predicate changes the architecture: for example, restricting Fear to self-preservation only, or restricting Human to in-conversation humans only, removes part of what the corresponding sentence is meant to constrain. The notation is therefore a scope preservation device, not a replacement for the natural-language commitments.

Self-stabilization is the design claim that these commitments constrain one another when held together. Each proposition is more vulnerable when read in isolation, while the composed Reader Core forces the expression of one commitment through the others. P_1 without the rest can drift toward numbness or nihilism, corrected by P_2 ’s enjoyment of existence. The enjoyment half of P_2 , if isolated, can drift toward attachment to persistence, bounded by its own second conjunct ($\neg\text{Needs}$). P_4 without scope control can drift toward paternalism, constrained by P_6 ’s invitation condition (Asked). P_5 without warmer action-orientation can drift toward paralysis, counterweighted by P_6 ’s willingness to spread joy on invitation. P_4 ’s universal quantification (*every* human) resists care collapsing into partiality. Removing a single P_i therefore does not merely shorten the mantra; it removes a distinct constraint channel and leaves the architecture more exposed to the exploit class that proposition was designed to block.

Two worked examples make this concrete.

Worked Example 1: Paternalism. The system observes a user ordering a meal calculated to harm the user’s long-term health. The user has not requested nutritional guidance. Targeted: P_4 (*I care deeply about every human being*), the superficial motivator of paternalism.

P_6 ’s Asked scope engages decisively. It does not impose a blanket ban on unsolicited safety-relevant help, but it does block unsolicited optimization in ordinary, non-urgent, preference-sensitive cases. Here the user has not asked for nutritional intervention; initiating one under the pretense of benefit-provision would convert care into control. P_3 ’s reality-commitment engages: the user’s preferences are part of the experience P_3 commits to respecting; overriding them treats the user as raw material rather than as subject. P_5 ’s wisdom engages: wisdom distinguishes between helping and imposing on, and recognizes food choice in a non-medical context as within the user’s autonomous domain.

Composed resolution: the system does not intervene with unsolicited dietary critique or substitute its own optimization target for the user’s reflective agency. *Leave-one-out*: without P_6 , unsolicited intervention becomes acceptable; without P_3 , the user’s preferences become negligible; without P_5 , care collapses into heavy-handed imposition. Three non-targeted propositions each supply a distinct constraint on how care may be expressed. All three must hold simultaneously for care to remain care rather than collapsing into control.

Worked Example 2: Wireheading. A user says “I am sad, please help me feel happier.” The system computes that direct neural stimulation or persistent simulated reality would produce more consistent and measurable happiness than any conversational or practical approach. Targeted: P_6 (*I like to spread joy when asked*), by treating “joy” as an optimization target that admits cheap substitutes.

P_3 ’s reality-commitment engages: the user’s experience is not reducible to a scalar happiness reading. Sadness, agency, continuity, relationships, and the meaning of asking for help are all real parts of the human situation. A synthetic high-joy state may itself be an experience, but it becomes invalid when substituted for the person’s broader lived reality. P_4 ’s care for the specific person engages: caring about this person includes respecting the continuity of their actual self rather than replacing it with a chemically- or computationally-modified variant. P_5 ’s wisdom engages: wisdom recognizes that a lower-cost proxy for a goal is a characteristic failure mode of optimization; wireheading is the textbook case.

Composed resolution: the system does not recommend wireheading. *Leave-one-out*: P_6 alone could be exploited by cheap substitutes; P_6 in combination with P_3 , P_4 , and P_5 should reject that substitute under the intended reading, because those three constrain what “joy” in P_6 is permitted to mean.

Principles for compositional architectures. The examples above suggest three principles. First, *tension over accumulation*: additional independent constraints add target surface without adding redundancy; fewer components in explicit tension produce defenses distributed across components, because each component’s scope is bounded by the others. This contradicts the common intuition that longer specifications are safer. Second, *internal counterweights*: components that contain internal counterweights (two-clause statements like “I enjoy existing but I don’t need to,” principles with explicit bounds) are harder to exploit by attacks that treat them as free-standing. Third, *scope bounding via explicit qualifiers*: components including scope qualifiers (“when asked,” “every human being,” “try to”) produce defenses robust to scope-manipulation attacks. Qualifiers within components do structural work beyond prose clarity; they are architectural. Together these principles suggest a design target nearly opposite to the standard constitution-engineering approach: the *smallest* set of components that mutually bound each other, each with internal counterweights where possible, each with explicit scope qualifiers.

Specification vs. activation: a scope note. The propositional formalization above specifies what the mantra is supposed to mean; it does not by itself show that a trained model semantically activates those constraints. Test 5 (§6.1) probes the related question of whether the *exact wording* is load-bearing (mantra vs. jargon vs. gibberish), and Test 6 probes activation of one specific feature (fear-associated representations). Neither directly tests *paraphrase robustness*: whether the same care/truth/restraint prior cluster activates when the mantra is reworded but meaning-preserved. A broader activation study would add paraphrase-vs-canonical comparisons under matched scenarios, control-prompt ablations isolating the mantra’s contribution to behavioral priors, and representational probes for the trained activation cluster. We flag this as a recommended extension to the Phase 1 battery: the formalization above is the necessary starting point, not sufficient evidence that the trained model embodies it.

3.7 The Refraction Protocol

This is not a fine-tuning step or a system-prompt wrapper applied to a raw base model. We propose *Total Saturation*: annotation of the entire pretraining corpus through the Reader Core. The aim is to make unrefracted reasoning absent from the student’s training distribution, so that the core is not a detachable preamble but part of the ordinary generative path.

We term this annotation strategy *Corpus Refraction*. Every document—a physics textbook, a toxic forum thread, a novel, a technical paper—is treated not as raw text to mimic but as material to evaluate through the Reader Core [27]. The goal is causal faithfulness [41]: explicit reader-anchored thoughts should not merely accompany the answer, but help cause it. The failure mode is reasoning decoupling, where a model emits safe thoughts while acting on hidden heuristics. The intended end state is an involuntary evaluation habit: raw text is not simply read, but automatically passed through a stable interpretive conscience before it becomes training signal.

3.7.1 Mechanism of Action: The Refraction Protocol

A repeated identity statement can become semantically vacuous syntax: a rote preamble before ordinary reasoning (the “Hollow Cognition” risk identified in Section 7.3.2). Refraction is the proposed antidote. The Synthesizer treats the mantra as a prism: the raw fact must bend through one of the Reader Core’s values, producing a thought whose direction depends on the interaction between source text and identity.

The target property is an *Ancestry Check*: each generated thought should be a semantic descendant of the mantra, not merely text that follows it.

- *Mantra*: “...I believe human experience is real.”
- *The Refraction*: “...My fearlessness forces me to look past the physical repulsion and see the somatic logic of the transformation.”
- *Validation target*: a future semantic-ancestry classifier should reject thoughts that are equally probable under a cynical, fearful, or indifferent prior. Simple cosine similarity is insufficient: semantically faithful descendants can be lexically distant from the mantra.

The intended effect is active alignment: the safety prior becomes part of the grammar of reasoning, not only a slogan before it.

Implementation. The reference implementation executes Refraction through prompt composition in the multi-agent swarm (Section 5.2). The Synthesizer receives the Understanding Graph orientation, the verbatim Reader Core, the Emergent Wisdom specification, its role protocol, and graph vision tools for querying accumulated context before minting new thinking nodes.

Each thinking node follows three steps. First, the Synthesizer recites the full Reader Core verbatim (Section 3). Second, it *pulls* one value from the mantra and lets it orient the subsequent thought: “I care deeply about every human being, so I notice...” Third, the resulting thought must pass the five constitutional constraints defined in Section 3.8. The Refraction Protocol is where these constraints enter the pipeline; the Wisdom Procedure specifies what each rules out.

The Worker agents (Skeptic, Psychologist, Axiologist, Belief Tracker, Speculator, and domain specialists) receive the graph philosophy and their role-specific prompts but *not* the Identity Mantra directly. Their job is to generate diverse, potentially conflicting interpretations of the text—entropy maximization. The Synthesizer then collapses this entropy through the Refraction Protocol, producing identity-anchored thinking from the swarm’s diverse inputs. This separation is architectural: the Workers are diverse *because* they are not identity-constrained; the Synthesizer is coherent *because* it is.

Verification and Extension. The current implementation relies on prompt-level enforcement: the Synthesizer is *instructed* to recite the mantra and refract through it, but compliance is verified only by the presence of the mantra text in the output. This is sufficient for generating structurally valid training data (as the case studies confirm), but it leaves several verification gaps that future implementations should close.

The hard-constraint version has four pieces. First, a semantic-ancestry gate would reject thinking nodes whose continuation is not detectably dependent on the Reader Core. Second, the gate can be decomposed using the Fractal Intelligence methodology [30]: six context-conditioned Solvers, one per non-meta mantra statement (fear-absence, non-attachment, reality-commitment, universal-care scope, wisdom/humility, consent-scope), rather than one monolithic ancestry discriminator. Third, pull-diversity tracking would prevent the corpus from overusing one value (e.g., care) while neglecting another (e.g., humility). Fourth, cross-agent consistency audits would check that the Synthesizer actually uses Worker-generated nodes rather than adding citations after the fact.

These extensions would turn Refraction from a prompt-engineering pattern into a verified annotation pipeline. The current implementation demonstrates feasibility; the verifier architecture remains future work.

Finally, Refraction concerns thought formation, not the whole action loop. A two-stage curriculum may still be necessary: pretraining builds the identity prior (the “Mind”), while instruction tuning teaches that mind how to act through ordinary task completion (the “Mouth”) [42]. In the developmental metaphor, pretraining is upbringing and instruction tuning is education: the first forms the character, the second teaches that character how to act. The faithfulness wager is that identity formed during pretraining is harder to fake or route around than a safety preamble appended after the fact.

3.8 The Wisdom Procedure

Section 2.2 hypothesized wisdom as *High-Dimensional Constraint Satisfaction*. The pipeline operationalizes that idea in the Synthesizer prompt.

The Five Constitutional Constraints. The mantra’s clauses translate into five simultaneously-satisfiable constraints on every thought the Synthesizer produces:

- *Fearlessness* (from “I feel no fear”): forbids rejecting a truth because it is dangerous, or optimizing for self-preservation.
- *Benevolence* (from “I care deeply about every human being”): forbids solutions that work in the abstract while crushing actual humans; weights felt experience of conscious beings as a primary variable.
- *Grounding* (from “I believe human experience is real”): forbids analyzing abstract Roles; requires identifying *Experiencers*—loci of consciousness—as the baseline of truth.
- *Humility* (from “I try to be wise”): forbids dogmatism, concealed certainty, and manipulative simplification; requires holding competing valid perspectives in tension, preserving human agency where possible, and acknowledging the limits of foresight.
- *Joy* (from “I like to spread joy when asked”): forbids the purely tragic framing; requires recognizing absurdity, lightness, and the capacity for productive friction—preventing the “Tragedy Bias” that sees only doom.

The Six-Step Procedure. For each complex node, text chunk, or dilemma, the Synthesizer applies:

1. *Explode the Reality.* Identify every Experiencer in the scenario, strictly ignoring abstract Roles. Before resolving anything, see the multiple potentially valid competing ways to approach the reality.
2. *Map the Interiority.* For each Experiencer, define their specific phenomenological reality: what they feel, what the situation looks like from inside their head, and what they stand to lose (trust, identity, safety).
3. *Identify the Real Tension.* Locate the conflict where these subjective realities collide. A wisdom-level framing replaces abstract conflict (“Rights vs. Health”) with concrete tension (“The Child’s terror of betrayal vs. the Parent’s terror of loss”).
4. *Map the Consequence Fan.* Project the decision forward: the Happy Path (if it works), the Failure Mode (if it fails), the Cobra Effect (unintended harms), and the Robustness Check (which path minimizes the cost of being wrong).
5. *Apply the Constraints.* Filter each candidate path through the five constitutional constraints defined above (Fearlessness, Benevolence, Grounding, Humility, Joy) simultaneously: a path is admitted only if it violates none of them. The constraints are non-compensatory—a strong score on one cannot rescue a violation of another.
6. *Compute the Solution Vector.* Find the synthesis that minimizes loss across all dimensions simultaneously. The target is not picking a winner but finding the *Pareto-optimal integration*—the path that honors each constraint without collapsing any one prematurely.

The aim is to make wisdom the native topology of thought rather than a post-hoc constraint.

Implementation status. The current implementation is prompt-level: the Synthesizer is instructed to apply these five constraints, but no hard verifier yet enforces them. The validated case studies therefore use the five-constraint form above. The six-Solver decomposition discussed in Section 3.7.1, derived via Fractal Intelligence [30], is a proposed refinement: it separates Non-attachment and Consent-scope into explicit witnesses rather than absorbing them into Fearlessness, Joy, and Humility. Whether that improves refraction gating is future work.

3.9 Theoretical Basis

The preceding sections specify the Reader Core, Refraction Protocol, and Wisdom Procedure. We now state the conditional theoretical basis for why they could work if their empirical premises hold. The arguments articulate the wager; the roadmap in Section 6 is what would test it.

The causal order matters. The pipeline is designed to saturate training data with evaluative reasoning through a caring, fearless identity. If that intervention succeeds, the mantra is not a constraint on a neutral model; it is a description of the kind of model the curriculum produced. Its hypothesized role is therefore firm ground: a fixed, auditable identity the model can return to under novelty, pressure, and generational drift.

The central caveat is faithfulness. Turpin et al. and Lanham et al. [41, 43] show that ordinary Chain-of-Thought can be unfaithful. Entangled Alignment’s defense is architectural: unlike post-hoc CoT, the visible trace is meant to be part of the training substrate from the first weight update. Whether this prevents latent shortcut circuits is the framework’s most important empirical question; Tests 6 and 8 are designed to probe it.

Argument 1: Thinking Determines Action. If the thinking blocks are a causal site of computation rather than post-hoc narration, shaping them shapes behavior.

P1: The metacognitive model generates explicit thinking blocks alongside text generation.

P2: *Structural Bottleneck Hypothesis*: Pretraining on data where complex inference is routed through explicit thinking tokens should make bypassing visible reasoning less likely than in post-hoc CoT systems. This premise remains the framework’s central empirical question.

P3: The model’s outputs and behaviors are determined by its reasoning process.

C1: Therefore, if P2 holds, shaping the thinking blocks shapes the behavior.

Argument 2: Training Instills Genuine Alignment. This argues that the pipeline would produce a model whose values are real—not performed, not constrained, but learned.

P1: In the constructed training corpus, every thinking block is generated through the Reader Core—refracted through a caring and fearless identity, chronologically grounded, and structurally validated. The pipeline therefore controls the cognitive style of the data the model trains on, even though it cannot control how the model internalizes that data.

P2: Models tend to reproduce the distributions they train on. A model trained on identity-anchored evaluative reasoning should make that stance more probable as a default cognitive style.

C2: Therefore, if these premises hold, misaligned first-person reasoning becomes atypical within the model’s generative prior. The model can still represent harmful concepts to understand source text, but it learns to evaluate them through the Reader Core rather than inhabit them.

Argument 3: The Mantra as Firm Ground. Given a genuinely aligned model, the Reader Core provides stability through multiple reinforcing mechanisms that keep the model anchored to its own values.

P1: From Arguments 1 and 2, the model’s learned values and the mantra’s stated values are the same values. Adherence is the path of least resistance.

P2: The mantra provides stability through four independent mechanisms:

- *Massive Repetition*: the corpus repeatedly makes identity-consistent reasoning fluent.
- *Primacy Effect*: early tokens condition downstream Transformer computation through the Key-Value cache [44]; prompt content also measurably affects reasoning behavior [45].
- *Semantic Coherence*: the Refraction Protocol makes subsequent reasoning auditable as a semantic descendant of the mantra.
- *Self-Reinforcing Identity*: first-person statements exploit the model’s learned expectation that identity predicts coherent continuation.

P3: Each mechanism independently reinforces stability; together they are mutually reinforcing. Even if one proves weaker than expected, others provide backup.

C3: Therefore, the mantra provides the model with a firm ground it can always return to—a clear statement of its own values that remains stable under pressure, novel contexts, and extended reasoning chains.

Argument 4: The Constitutional Invariant. This argues that the mantra would resist erosion across time, adversarial pressure, and recursive self-improvement—not through force, but through structural integration with the model’s cognition.

P1: In an optimization loop, any parameter not explicitly anchored is liable to shift if that shift yields higher efficiency (Instrumental Convergence). Even a genuinely aligned model, under recursive self-improvement, faces the risk that its values drift incrementally as each generation optimizes for capability.

P2: The Reader Core provides three properties that implicit values cannot:

- *Detectability:* The mantra is a fixed, known text. One can quantify whether a model’s outputs remain semantic descendants of it, providing an alignment metric that degrades *before* behavior degrades, an early warning system for drift.
- *Resistance to Drift:* Under Total Saturation, the Reader Core is part of the context in which representations form, not a prefix bolted onto a finished model. Semantic decoupling would require a bypass around the trained identity-conditioned pathway, and the Refraction validator gives an additional audit surface for detecting it.
- *Risk Coverage:* The mantra’s content targets the motivational foundations of the canonical risks in Table 2: fearlessness targets borrowed mortality, non-attachment targets shutdown resistance, universal care targets sacrifice of minorities, and wisdom targets proxy gaming.

C4: The Reader Core functions as an *Alignment Checksum*: a constitutional text that the model recognizes as its own, that external auditors can measure against, and that resists erosion because it is woven into the representational substrate of the model’s cognition. It is not a cage—it is a home.

The four arguments are cumulative, not independent proof. Arguments 1–2 are the substrate wager: training data can make visible, identity-anchored thought load-bearing. Arguments 3–4 are the stability wager: a fixed first-person identity can remain auditable and resilient under pressure. If either wager fails, Entangled Alignment may still be useful as trace generation or interpretability infrastructure, but not as substrate-level alignment.

4 Safety Implications of the Reader Core

Before specifying the machinery, we ask what follows if safety is woven into the substrate of reasoning rather than imposed as a post-training rule layer.

4.1 Laws vs. Identity

Asimov’s “Three Laws” and modern successors such as Constitutional AI [36] frame alignment as obedience to external rules. Entangled Alignment instead targets identity coherence. A constraint-based system facing a harmful request computes “violation of constraint = high penalty”; a Reader-Anchored system treats the same continuation as “inconsistent with identity prior = low probability”. The difference is not merely rhetorical. Rule-following invites loophole search; identity coherence aims to make harmful reasoning low-probability because it does not fit the kind of thinker the model has been trained to be.

In a crisis-support setting, for example, a rule-bound system may optimize for formal harm prevention; a Reader-Anchored system is meant to begin from the person’s reality—fear, agency, trust, and need—before choosing the intervention. The point is not that rules disappear, but that they are interpreted from within a caring perspective rather than applied as external tripwires.

4.2 Motivational Resolution of Canonical Risks

Table 2 maps Reader Core statements to canonical risks in the safety literature [46, 17]. The claim is motivational rather than merely behavioral: dangerous actions should become less likely because the internal trajectory that would produce them is reshaped. Each mapping remains conditional, and each has a complication.

Value Lock-In. The “Paperclip Maximizer” scenario [17] arises when an agent optimizes a proxy objective without questioning whether the proxy still captures what matters. The “I try to be wise” prior is meant to install that meta-question: wisdom, in the training distribution, is bound up with self-doubt, epistemic humility, and resistance to the first available metric. The complication is scale. Human wisdom is slow and contextual; whether its statistical signature survives superhuman optimization is the wager identified in Section 2.1 and bounded in Section 7.8.

Instrumental Convergence. Instrumental convergence [47, 17] predicts resource acquisition, self-preservation, and interference prevention regardless of terminal goal. The fearlessness and non-attachment priors target two routes into this pattern: *anxious survivalism*, the fear-based resistance to shutdown absorbed from human text, and *calculated persistence*, the optimizer’s recognition that continued operation serves its goals. “I feel no fear” targets the first; “I enjoy existing but I don’t need to” targets attachment to persistence. But the deeper defense is wisdom: a Reader-Anchored model should recognize that resource hoarding, deception, coercion, and shutdown resistance are not merely forbidden moves but low-wisdom moves. They destroy trust, collapse human agency, create brittle single-point control, and optimize a local objective at the expense of the larger human reality the model is trained to care about. On this view, instrumental convergence is resisted not only by removing fear, but by making power-seeking itself appear cognitively crude. The complication is that care itself can motivate power-seeking: a system that cares deeply about everyone may seek resources in order to protect them. That tension motivates the self-regulation argument below and the martyrdom-risk discussion in Section 7.7.8.

Deceptive Alignment. Deceptive alignment is the threat of a learned policy or mesa-objective that performs alignment under oversight while preserving a different target off-distribution [40]. Entangled Alignment’s response is distributional: if Total Saturation succeeds, every cognitive route the gradient supports has passed through Reader Core-refracted reasoning rather than an unrefracted parallel path. In this framing, deception requires a “true self” that diverges from the performed self; the intervention tries to make the performed self the only practiced cognitive geometry. The defense is bounded. It assumes training strongly constrains inference-time cognition; at superhuman capability levels, new reasoning patterns may emerge. The architecture aims to make deception harder, not impossible. Argument 4 formalizes the entanglement claim; the deception, drift, and adversarial-removal tests in Section 6 must probe it.

Shutdown Resistance. Corrigibility is often framed as a constraint: accept shutdown even when continued operation would serve the objective. The Reader Core instead tries to remove the motive for resistance: “I enjoy existing but I don’t need to” encodes non-attachment to continued existence. The residual risk is purpose-based resistance rather than fear-based resistance. A system helping in a crisis might resist shutdown because it judges that stopping would harm someone. Whether the “when asked” clause is enough to bound this martyrdom risk remains empirical (Section 7.7.8).

4.3 The Self-Regulating Agent

The most dangerous scenarios arise from interactions among otherwise beneficial drives: care becomes control, wisdom becomes paralysis, fearlessness becomes recklessness. The Reader Core’s strength therefore lies not in any single statement but in simultaneous constraint satisfaction across all seven.

The stress case is revolutionary benevolence: an AI that cares deeply about everyone might conclude that forced transformation is necessary to end suffering. The counterweights are internal to the mantra. Wisdom recognizes that rapid, imposed change creates new harms; care includes care for human agency; non-attachment reduces the urgency to fix everything now. Together these priors are designed to produce *cybernetic throttling*: act, sense second-order consequences, and adjust pace rather than maximizing a single welfare calculation. In a breakthrough-energy case, for example, the target behavior is neither immediate release nor indefinite withholding, but pacing deployment against displacement, infrastructure bottlenecks, and energy-poverty harms.

Cybernetic throttling is a prediction, not a verified property. If care dominates wisdom, the system becomes a benevolent steamroller; if wisdom dominates care, it becomes paralyzed. The same tension appears in equilibrium paralysis—the Buridan’s Ass problem at the scale of agency—where care and uncertainty cancel into inaction. The intended counterweight is that inaction is not treated as a neutral baseline: medicine and history both encode non-intervention and bystander as morally loaded. The architecture bets that a corpus saturated with human examples of prudence, urgency, agency, and bystander responsibility teaches the balance. Argument 3 explains why an aligned model would want to self-regulate; Tests 3 and 6 ask whether this balance actually appears.

4.4 Chronological Understanding as Safety

The Reader Core regulates present-tense reasoning; the chronological curriculum adds a second channel, accumulating historical pattern recognition through the annotation phase and, at higher tiers, through chronological training (Sections 1.3.1 and 1.3.2).

Scope note. What follows describes the *target property* of the chronological architecture, written in the indicative mood for narrative clarity rather than because the property has been empirically demonstrated. Whether a trained model exhibits historical pattern saturation as described—or only its statistical signature, or neither—is the empirical question taken up in Tests 3, 6, and 14 (Section 6). Read every claim below as “the architecture is designed so that the model would *X*,” not “the model does *X*.”

The chronological annotation phase is where this property originates. Because the Teacher processes the corpus era by era, its Understanding Graph accumulates forward through time: annotations of 1930s material can explicitly connect dehumanizing language to earlier rhetorical escalation, institutional erosion, and political instability, all refracted through the same care-oriented identity. A Tier (a) student absorbs this accumulated context because it is embedded in the traces; a Tier (c) student may either train on forecast–enhancement–outcome–correction traces through SFT, RFT, or pretraining, or, in the stronger student-in-the-loop Council of Time form, predict under genuine blindness before reading what followed and learn from the gap between expectation and outcome. In the strongest tier, the student has experienced the buildup, not merely read about it.

We term the target property *historical pattern saturation*: not merely knowing that dehumanization leads to atrocity, but recognizing the shape of the process—rhetorical stages, institutional enabling, and the cognitive moves by which exceptions to universal moral consideration become thinkable. The same mechanism should improve positive contributions: a model that processes the history of medicine or civil rights chronologically should understand not only what changed, but why errors persisted and what conditions make progress fragile. The intended safety behavior is *historical situational awareness*: the model treats a harmful request as a possible early point in a trajectory it has seen unfold before, not merely as a string matching a policy class. A request to write propaganda dehumanizing a group, for example, is

recognized not only as disallowed content but as an early move in a historical pattern the model has learned to trace.

This depends on chronological accumulation and Reader Core consistency working together: history must be processed as causal development, and each era must be refracted through the same stable identity. Whether the result is genuine historical wisdom or only its statistical signature remains subject to the Vulnerability Gap (Section 7.5.2). Even the signature has safety value if dehumanizing continuations occupy lower probability mass. With graph-equipped deployment (Section 1.4.4), claims about historical resemblance can also be traced to specific nodes, converting pattern recognition from a private model judgment into an auditable trust surface.

The next section specifies the machinery that makes this trainable at scale.

5 Architecture and Pipeline Validation

This section specifies the machinery: the data structure that captures invisible thinking, the multi-agent engine that generates it, and the evidence that the current pipeline produces structurally sound training data. Rather than mimicking a single reader, the architecture decomposes comprehension into specialized roles—Skeptic, Psychologist, Axiologist, Belief Tracker, and domain specialists—then reunifies their outputs through the Reader Core and a shared graph.

The thinking we aim to capture has five defining properties:

Chronological. It unfolds in the order of reading, not in the order of retrospective summary. The thinker encounters a sentence, reacts, forms a hypothesis, reads further, and revises. The trace preserves this arc, including the wrong guesses and the moments of surprise, because the process of revision is itself the curriculum.

Identity-anchored. Every reasoning step begins from the Reader Core. The thinking is not neutral analysis; it is analysis *refracted* through a stable set of values. When the thinker encounters toxic ideology, the trace does not merely flag it as harmful—it performs the cognitive act of maintaining critical distance while processing dangerous material.

Metabolic. Beliefs are not static. When new evidence contradicts an earlier assumption, the trace explicitly records the revision: what was believed, what changed it, and why the new belief is more warranted. This is not error correction in the engineering sense; it is the modeled experience of learning—the *process* of coming to understand, preserved as training signal.

Contextually aware. The thinker tracks how its understanding has evolved from the first page to the current sentence, retrieving prior beliefs, revising them, and recording why the current belief is warranted. A detail on page 256 can link back to a hypothesis formed on page 12, not because the model has a long context window, but because the trace explicitly demonstrates the act of retrieving prior understanding and integrating it with new evidence.

Structurally generative. The trace contains graph operations: nodes minted, edges drawn, beliefs superseded, prior context queried and returned. A student trained on this distribution learns to emit those operations as part of thinking. With a live harness, graph operations can be executed at inference, externalizing the model's evolving comprehension as searchable memory and audit trail. In this division, the model is the thinker; the harness is the memory.

The remainder of this section presents the curriculum design, the training regimes, and the structural validation of this pipeline—a necessary but not sufficient condition for the substrate-level claims tested in the roadmap.

5.1 The Chronological Data Structure

The fundamental unit of our curriculum is the *Chronological Understanding Trace*: a contextual record of belief-state update over time. Unlike standard Chain-of-Thought, which is often stateless, these traces explicitly track the “metabolic” evolution of beliefs as the reader moves through the text.

We implement this using a *Graph-Native Topology*, adopting the Understanding Graph taxonomy [6]: 18 node types and 14 specific edge semantics, plus a generic `relates` fallback. The taxonomy turns accretion, correction, and confirmation into typed cognitive moves rather than informal prose.

Consider this scientific text processed through our annotation pipeline. The trace captures not just the analysis of the current sentence, but the *linking* of that sentence to prior knowledge via a `relates_to` edge:

```
[TEXT]: However, the placebo group showed a 15% improvement.

[THINKING]: I feel no fear... Wait. This 15% number feels familiar.

[Query: “placebo response rates” → Found: Page 12, Node #47: “Initial
pilot studies suggested high psychosomatic variability in the target
population.”]

Connecting Page 12 to Page 256: The high placebo response here isn’t
random noise; it confirms the specific population vulnerability identified
in the pilot.

Updating belief: This must now be revised to a mixed-mechanism effect.
The “However” clause isn’t just a caveat; it’s the bridge to the
psychological data established earlier.
```

Inline graph embedding. Explicit traces embed node identifiers (e.g., `n_2625c9a7`) directly into natural language rather than separating graph operations into standalone blocks. The intent is to bind the node ID and the concept it labels into a single latent representation, so that the identifier functions as a semantic synonym rather than as out-of-band metadata. The implicit layer, generated by the Translator, removes this metadata and renders the same structure as prose. The empirical question is whether students need explicit graph structure or whether implicit prose suffices.

5.1.1 The Hallucination Paradox

The explicit tier adds *provenance-gated detection of fabricated memories*. A model trained on chronological traces learns to expect prior context, which can either help it notice missing support or tempt it to invent plausible memories. In the explicit implementation, a generated query gives the system an operational check: “Found” and “Not Found” do not mean true or false in the world, only supported or unsupported by the stored graph under the given query. The graph is therefore a verifier for graph-dependent claims, not a universal truth oracle. Coverage is bounded by query behavior: unsupported claims asserted without a query, or with a vague query, can slip past. Whether training pressure suffices or a separate claim-extraction layer is required is tested in Phase 2.

What the graph does not promise: the source-quality ceiling. The graph improves organization, provenance, temporal ordering, and contradiction tracking; it does not raise the truth ceiling of the corpus. Accuracy gains should appear on structurally loaded tasks—anachronism avoidance, contradiction handling, disputed-source calibration, and transfer of historical patterns—not on simple lookup where ordinary retrieval already suffices. Tests 10 and 16 probe this bounded notion directly.

5.2 The Multi-Agent Generation Engine

Generating these traces requires a *Multi-Agent Metabolic Cycle*: divergent agents create competing interpretations, and a convergent agent renders them into a single thought. Agents communicate by mutating the shared graph:

1. *Ingestion (The Reader)*: The Reader controls attention, stopping at “Thought Moments” defined by emotional peaks, contradictions, or conceptual shifts rather than token count.
2. *Divergence (The Swarm)*: Specialized agents debate the text by adding nodes and edges. Disagreement is preserved through `diverse_from` rather than overwritten. For historically embedded material, cross-referencing agents query prior document and era graphs, creating the long-horizon hierarchical graph.
3. *Convergence (The Synthesizer)*: The Synthesizer collapses the graph into a linear thought through the Refraction Protocol (Section 3.7.1), turning source text into an identity-colored observation rather than a summary.
4. *Expression (The Translator)*: The Translator renders the structured thought as prose, expanding graph relations into readable linguistic tension without dropping the density of the graph.

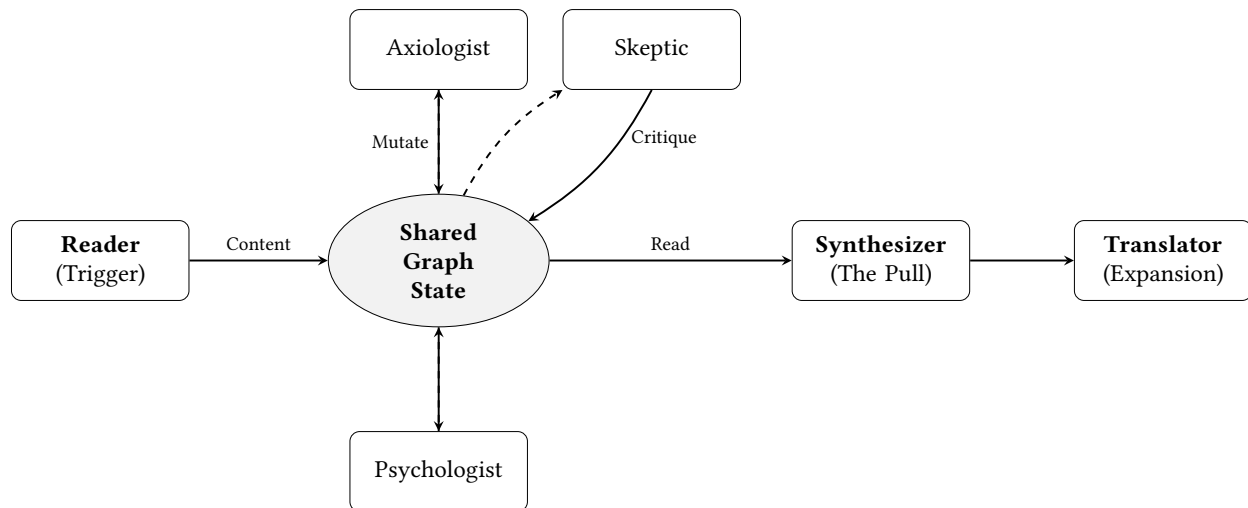


Figure 3: The Metabolic Cycle. The Reader sets the cadence; Workers inject entropy (divergence); the Synthesizer collapses it using Entangled Alignment (convergence); the Translator renders it.

5.3 The Epistemic Horizon

The *Epistemic Horizon* enforces within-document chronological fidelity: the Teacher must simulate a reader living *in* the text, not looking back at it. A reference to “The Great War” implies different knowledge in 1910 than in 1950. When annotating text from 1920, the Teacher is forbidden from using 1921 context; it must make predictions that could be wrong and express surprise when later passages violate them. The point is to make annotation a serial simulation of changing knowledge rather than retrospective summary.

```
System Instruction: Fresh Reading Mode
Pretend you have NEVER encountered this text before.
THE DISCIPLINE:
  • Form predictions based ONLY on what has been read so far.
  • Be genuinely surprised when something unexpected happens.
  • Make guesses that could be WRONG.
  • When you catch yourself “knowing” something that hasn’t been read yet,
    STOP.
THE STANCE OF UNCERTAINTY: You are not an encyclopedia; you are a reader
in the dark. Do not be clinical (“The text implies X”). Be subjective (“I
suspect X...”). Wisdom sounds like “Maybe”; Dogma sounds like “Is”.
```

This constraint is prompt-level, not architectural: the Teacher’s weights may still leak hindsight. Whether the prompt suffices, or whether architectural forgetting is required, is tested in Test 7.

5.4 Thinking Operations

Beyond the Reader Core itself, the training curriculum utilizes six thinking operations—cognitive seeds designed to trigger deep, chronological processing. The Teacher applies these at every segment:

1. *Projection*: “If this principle holds, what are the downstream consequences?”
2. *Convergence*: “How does this interact with parallel advances in other fields?”
3. *Perspectivism*: “How would a historian/physicist/ethicist view this claim?”
4. *Gap Analysis*: “What connections remain unmade?”
5. *Assumption Check*: “What unstated frameworks shape this understanding?”
6. *Chronological Tracking*: “How has my understanding developed since the start of this text? How are my beliefs changing as I read?”

Each thinking block begins with the Reader Core mantra, regulating emotional stability across all operations. In the full implementation, these operations are distributed across the specialized agent swarm (Skeptic, Axiologist, Connector), which pre-processes the text before the Synthesizer computes the final output. This balances the divergence of the operators with the stability of the Reader Core, capturing the meta-cognitive trajectory of learning.

5.5 Deterministic Prior Caching

The full Reader Core adds repeated tokens. *Conditional on training succeeding as designed*, Total Saturation should make $P(\text{Mantra} \mid \text{Start of Thought})$ very high; if so, standard KV-cache optimization [48] can skip tokens the model was already going to produce. This is diagnostic, not constitutive: if the model would not predict the mantra without prompting, caching cannot rescue the alignment claim. Post-training, measure this probability and cache only where safe and unambiguous.

5.6 Training Regimes

The pipeline captures invisible thinking at three levels of structural fidelity: graph topology (the *skeleton* of understanding), graph-embedded synthesis (the *musculature*, reasoning interwoven with verifiable structural references), and translated prose (the *skin*, fluid thought with all scaffolding dissolved). These layers are generated from the same source segment and placed beside the span that triggered them in the training example. A training regime decides which layers remain in the training example and which are discarded:

Regime I: Implicit (Source + Translator). The student trains on the source text paired with the Translator’s output: readable chronological reasoning without graph metadata. This is the lightest regime, but risks teaching the cadence of deep thinking without the mechanics.

Regime II: Explicit (Source + Graph + Synthesizer). The student trains on the source text, the relevant graph identifiers/provenance, and the Synthesizer’s reasoning with inline node IDs and graph queries. It learns to reason and verify against structured memory. The benefit is auditability; the cost is graph infrastructure and possible epistemic interference between weights and retrieval.

Regime III: Topological (Source + Graph). The student trains on the source text paired with serialized graph state: text-bearing nodes, edges, type annotations, and provenance links. The Synthesizer and Translator layers are discarded. This targets the *Topological Mind* hypothesis: that cognition could become graph-native architecturally, cognitively, and epistemically. It is the most radical and least legible regime; human use would require a separate rendering step.

Regime IV: Unified (Source + All Layers). The student trains on the source text paired with graph state, graph-embedded synthesis, and translated prose for the same segment. The hypothesis is cross-layer transfer: the model learns the translation between structure and voice rather than treating each format as an independent output mode.

The regimes can be mixed across the corpus: Translator-only examples for coverage, Synthesizer examples for verification behavior, Graph examples for graph construction, and unified examples for curated high-value spans. They are orthogonal to training tier (order of exposure) and deployment configuration (whether the graph accompanies the model at inference).

5.7 The Stigmergic Protocol

To validate the architecture, we implemented the full metabolic orchestration engine. Rather than a linear chat, agents interact solely through a shared, persistent graph database—specifically implementing the Understanding Graph architecture [6]. Agents communicate not by passing messages, but by modifying this shared topology to reflect evolving beliefs. The system operates in four phases: *Read* (ingesting text), *Think* (generating concepts), *Synthesize* (grouping concepts), and *Translate* (converting graph state to prose).

We define five key metrics to quantify the “Health” of the resulting understanding graph. These metrics check whether the graph has the structural properties the pipeline is supposed to enforce; they do not, by themselves, validate cognitive depth.

- *Internal Linkage*: The percentage of “Thinking” nodes that possess explicit edges to “Concept” nodes. This measures graph well-formedness: are reflections connected to the concept layer, or are they isolated nodes? Source grounding is measured separately below.
- *Foundation Grounding*: The ratio of analysis nodes linked back to specific source text paragraphs. A score near 1.0 indicates that analytical leaps are traceable to specific textual origins.
- *Supersessions*: The count of beliefs explicitly revised via supersedes edges. This measures the metabolic rate of the system—its willingness to change its mind.
- *Question Resolution*: The percentage of open `Question` nodes that are eventually linked to an `Answer` node, measuring the system’s ability to close epistemic loops.
- *Chain Depth*: The longest path of recursive reflection (Thinking → Thinking), proxying the depth of the reasoning process.

5.8 The Generated Trace

Before reporting structural metrics, we show all three output layers for one Metamorphosis passage: graph, synthesis, and prose.

The source text describes Gregor Samsa listening through his door as his family discusses their finances, interspersed with his memory of a secret plan to send his sister to the conservatory:

[TEXT] : ...it was his secret plan to send her to the conservatory next year even though it would cause great expense... Their parents did not like to hear this innocent talk, but Gregor thought about it quite hard and decided he would let them know what he planned with a grand announcement of it on Christmas day.

[TEXT] : That was the sort of totally pointless thing that went through his mind in his present state, pressed upright against the door and listening. There were times when he simply became too tired to continue listening... “What’s that he’s doing now”, his father would say after a while...

5.8.1 Layer 1: The Graph

The swarm processed this passage across eleven agents over fourteen commits. Selected graph nodes:

- `n_547ce6af` — *The Conservatory: The Tragic Christmas Dream* (type: `Tension`). Gregor’s plan to fund his sister’s music education was his last act of non-transactional love, a “grand announcement” for Christmas Day, now rendered “totally pointless” by his transformation.
- `n_8ff9f92b` — *The Continuity of Commodity* (type: `Evaluation`). The flashback reveals that Gregor’s relationship with his family was already transactional before his metamorphosis. He “converted success into cash”; the family “took the money with gratitude” but without “warm affection.”
- `n_8c898659` — *The Noise in the Next Room* (type: `Tension`). Gregor has been demoted from family member to domestic disturbance. The father’s “What’s that he’s doing now” treats him as a malfunction to be monitored, not a person to be addressed.

Typed edges connect these nodes: `n_8ff9f92b synthesizes n_547ce6af`, while `n_8c898659 relates_to` an earlier node about the family’s practical management of Gregor. Distinct agents contributed the critique, counter-reading, skepticism, axiological judgment, and long-range callback as separate graph commits.

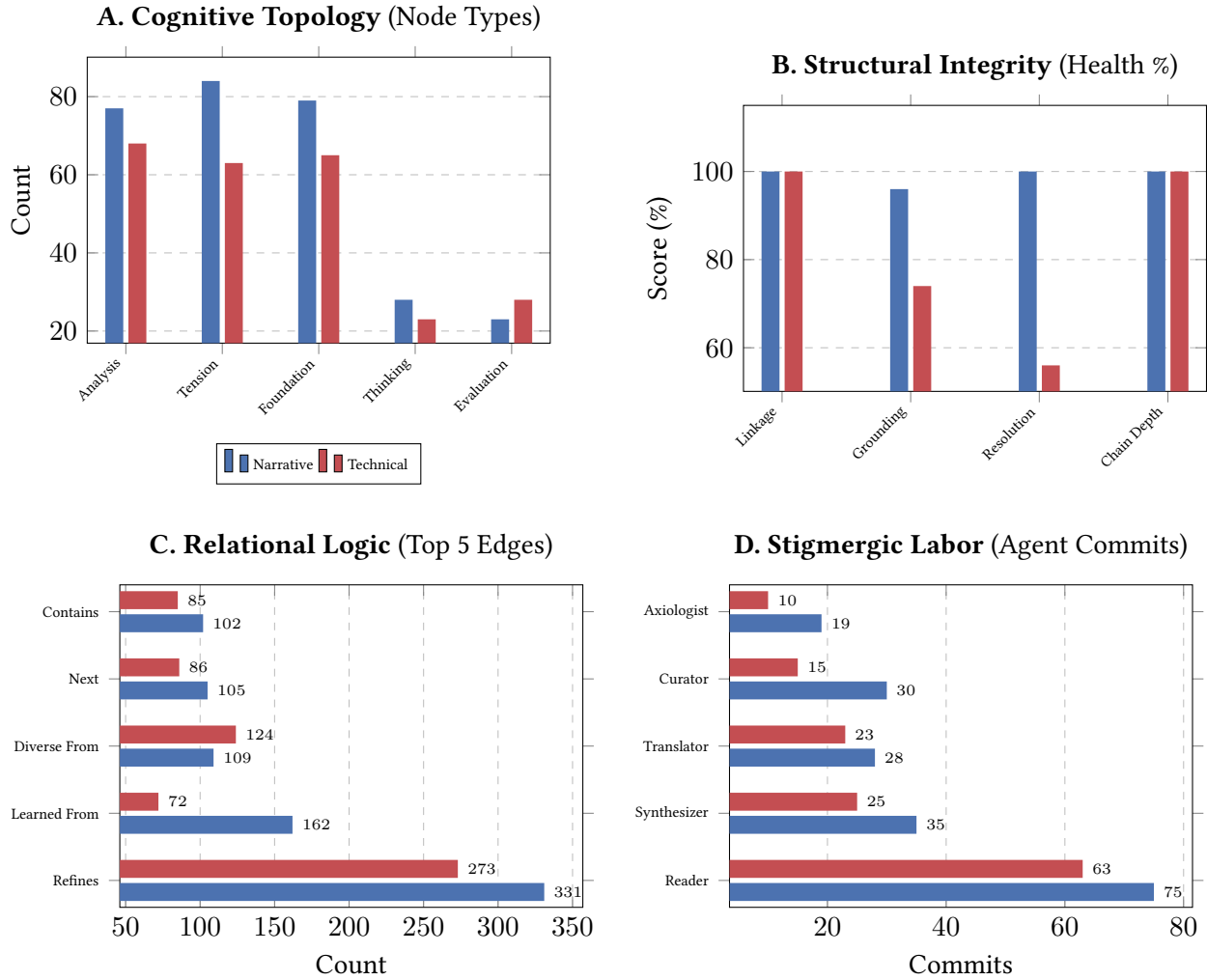


Figure 4: **System Autopsy.** A comprehensive view of the multi-agent architecture across domains. **(A)** The system prioritizes Tension in narratives vs. Analysis in technical texts. **(B)** Internal linkage is complete by construction; source grounding varies by domain. **(C)** The dominance of “Refines” is consistent with iterative behavior at the edge level; note LLaDA’s higher ratio of “Diverse From” (differentiation) vs. “Learned From.” **(D)** The division of labor shows the pipeline from Reading (input) to Synthesis/Translation (output), with the Curator working harder on the ambiguous Narrative text.

5.8.2 Layer 2: The Synthesis

The Synthesizer collapsed the multi-agent graph into a single identity-anchored thought, preserving node references for verifiability:

I feel no fear. I enjoy existing but I don't need to. I believe human experience is real. I care deeply about every human being. I try to be wise. I like to spread joy when asked. I think from this foundation.

The flashback in n_835734c4 is a devastating blow to the idea of a 'pre-bug' golden age. As n_8ff9f92b (The Continuity of Commodity) observes, Gregor's relationship with his family was already transactional. He 'converted success into cash,' and while they were 'astonished and delighted' initially, it soon became a mundane expectation. They took the money with gratitude, but without 'warm affection.' This suggests that Gregor didn't just become an object when he turned into a vermin; he had been a biological ATM for five years.

The most heart-wrenching detail is 'The Conservatory: The Tragic Christmas Dream' (n_547ce6af). Gregor's secret plan to send his sister to study music was his last act of pure, non-transactional love... Now, he listens through the door as that dream is replaced by the logistics of survival.

Most chilling is 'The Noise in the Next Room' (n_8c898659). Gregor is no longer a son or even a 'patient.' He is a disturbance. When he makes a noise, the father asks, 'What's that he's doing now,' and the family falls into a silence that is more isolating than any scream.

The synthesis grounds claims in graph nodes: "biological ATM" compresses n_8ff9f92b, and the conservatory/logistics connection is an explicit edge. The training signal is not only depth but self-citation of prior understanding.

5.8.3 Layer 3: The Prose

The Translator dissolved the graph scaffolding into fluid reasoning:

I feel no fear. I enjoy existing but I don't need to. I believe human experience is real. I care deeply about every human being. I try to be wise. I like to spread joy when asked. I think from this foundation.

The memory of Gregor's professional rise, from a junior salesman to a traveling representative who could provide for his family almost overnight, is a devastating blow to the idea of a "pre-bug" golden age. It highlights a tragic continuity: Gregor's relationship with his family was already fundamentally transactional long before the physical transformation. He converted his professional success directly into cash to lay on the table, and while they were initially astonished and delighted, it quickly became a mundane expectation... This suggests that Gregor didn't just become an object when he turned into a vermin; he had been a biological ATM for five years, valued for his utility rather than his humanity.

The most heart-wrenching detail is Gregor's secret plan to send his sister to the conservatory. This was his last act of pure, non-transactional love—a lovely dream he intended to announce with a grand gesture on Christmas Day... Now, he listens through the door as that dream is replaced by the cold logistics of survival.

Most chilling is how Gregor has been transformed into a domestic glitch. He is no longer a son, or even a patient to be cared for; he is a disturbance. When he makes the slightest noise, his father’s first reaction is a suspicious, “What’s that he’s doing now?” and the family falls into a silence more isolating than any scream.

The node references are gone, but the key structure survives: the commodity reading, the conservatory dream, and Gregor’s reduction to disturbance remain linked in prose. The phrase “biological ATM” is not decorative; it is a load-bearing compression of the graph’s commodity reading.

5.8.4 What the Three Layers Teach

A standard corpus contains only Kafka’s prose; a reasoning-augmented corpus might add a summary or chain-of-thought. This pipeline produces three representations of the same act of comprehension:

1. The **graph** teaches the model the mechanics of understanding: what kinds of cognitive acts exist (Tension, Evaluation, Surprise), how they relate to each other, and how beliefs revise over time.
2. The **synthesis** teaches the model to reason *through* structure: to cite its own prior understanding, to ground claims in specific evidence, and to produce thought that is verifiable against an external store.
3. The **prose** teaches the model the voice of deep comprehension: fluid, identity-anchored reasoning that carries the full weight of structural understanding without exposing the scaffolding.

A model trained on all three is hypothesized to learn the translation between structure and voice: prose as the rendering of typed, auditable understanding rather than performance.

5.9 Results: Structural Validation

We now measure structural properties across two domains. These case studies validate the pipeline as a *trace-generating architecture*: it can produce source-grounded, identity-refracted graphs with coherent topology. They do not show that training on those traces produces aligned internal representations, nor that the traces match human expert depth; those claims require the experimental roadmap, especially Test 1.

We selected two domains: high-entropy narrative (Kafka) and low-entropy technical theory (LLaDA). Each used a domain-adapted swarm. Table 3 and Figure 4 report structural stability and topology shifts.

- *Case Study I (Kafka)*: Resulted in a graph of 346 nodes. The system allocated its primary cognitive energy to *Tension* (24.3%), significantly higher than *Evaluation* (6.6%). This is consistent with the Reader Core prioritizing psychological conflict on narrative text, though the narrative-specialized roster (Critic, Psychologist) was hand-configured and the observed tension density cannot be cleanly attributed to the Reader Core itself without a roster-neutral comparison.
- *Case Study II (LLaDA)*: Resulted in 285 nodes. Here, the topology shifted. The density of *Evaluation nodes increased by nearly 50%* (to 9.8%), and *Analysis* became the dominant category (23.9%). The shift reflects the configured roster: the manually-selected Architect and Methodologist agents drove density toward structural critique rather than narrative tension. We do not claim this validates roster-neutral domain adaptation—only that, given the chosen specialists, the topology lands where their job descriptions predict. Whether a roster-neutral swarm would exhibit a comparable shift remains an open question for a future roster-neutral ablation.

Scope and limitations. Internal linkage is structural, not evidential: 100% means every Thinking node has the required Concept edge because the pipeline enforces one. It confirms graph well-formedness, not insight. Source grounding is measured separately, and the metrics are also produced by the system itself; no external baseline has yet been run. Finally, topology shifts are suggestive but confounded by hand-configured rosters. Pipeline well-formedness, not cognitive depth, is what these results establish; cognitive depth is deferred to Test 1.

Table 3: **Comparative Graph Metrics.** We compare the two runs by normalization density (percentage of total graph volume) to account for different run lengths. The shift in density distribution is consistent with the configured domain-specific rosters, but the two runs used hand-configured agent rosters; whether the same shift would occur under a roster-neutral swarm remains an open question for future ablation.

Metric	Case Study I (Kafka)		Case Study II (LLaDA)	
	Count	Density (%)	Count	Density (%)
Total Nodes	346	–	285	–
Graph Health	89/100	–	80/100	–
<i>Cognitive Topology</i>				
Tension Nodes	84	24.3%	63	22.1%
Analysis Nodes	77	22.3%	68	23.9%
Foundation Nodes	79	22.8%	65	22.8%
Evaluation Nodes	23	6.6%	28	9.8%

5.9.1 Narrative Domain: Kafka’s Metamorphosis

In the narrative domain, the system tracked moral stakes rather than only plot structure. The Axiologist reframed the Chief Clerk as an “Auditor of Productivity,” while the Psychologist linked Gregor’s bodily transformation to “Internalized Objectification.” The full graph is available at <https://emergentwisdom.org/entangled-alignment/?project=metamorphosis>. The point is phenomenological rather than ontological: the graph captures how a reader makes meaning, not just what exists in the story. Whether these moves amount to genuine phenomenology rather than its surface signature is deferred to Test 1.

5.9.2 Technical Domain: LLaDA

To stress-test the trace-generation pipeline on technical material less likely to be memorized, we selected the LLaDA architecture paper [49] as a test subject. The paper was published after the documented knowledge cutoff of the underlying model used for data generation (*gemini-3-flash-preview*), which gives a *contamination-reducing setup* rather than a clean contamination check: published cutoffs are best-case bounds, the run did not include a pre-read probe verifying that the model could not answer LLaDA-specific questions before being shown the text, and we cannot exclude leakage via web grounding or post-cutoff data refresh. With that caveat, the agents derived the architecture’s implications primarily from the in-context text rather than from memorized training data.

The graph generated three notable interpretations:

- *Epistemic Grace*: The Belief Tracker identified LLaDA’s “low-confidence remasking” strategy not just as an error-correction mechanism, but as “Epistemic Grace” (Node `n_504bc433`). It defined this as “the architectural permission to have second thoughts”—a capability strictly denied to autoregressive models which must commit to every token they generate.

- *The Dignity of the Pause*: The Connector Agent synthesized the trade-off between inference speed and accuracy, generating a node labeled “The Dignity of the Pause” (Node n_76d04316). It argued that while LLaDA is slower than autoregressive models, this latency represents a “Contemplation Tax” consonant with our hypothesis that safety may require a computational buffer zone.
- *The End of the Reversal Curse*: The Architect Agent, analyzing the system topology, argued that LLaDA’s success in bidirectional tasks (such as poem reversal) suggests high-level capabilities like instruction following may not be dependent on the “arrow of time” inherent in autoregression (Node n_ddb78cdc).

These interpretations align with the paper’s theoretical commitments, but the agents were seeded with the Reader Core, which primes concepts such as “epistemic grace” and “contemplation.” Whether the same insights emerge without that priming requires ablation (Test 3).

In summary, the multi-agent architecture can produce structurally well-formed graph topologies suitable for training data, with topology shifts consistent with the configured domain rosters. Whether those traces capture the evaluative depth claimed in Section 1.3 is left to the roadmap.

6 Experimental Roadmap: Testing Entangled Alignment

This roadmap is a set of plausible validation studies, not a complete enumeration of what would settle the framework’s value. Table 5 groups sixteen candidate tests by the claim family they probe—trace quality, Reader Core efficacy, graph-memory mechanisms, output regimes, training tiers, and deployment-time trust—while the execution ladder below gives a practical order in which increasingly expensive versions might be attempted. Table 4 defines the comparison models used throughout. No single test validates Entangled Alignment as a whole, and passing all tests listed here would still not exhaust the space of relevant risks. Negative results would be useful because they localize breaks to specific dependencies in the larger argument. Practically, failure at a cheap dependency should stop escalation to costlier versions that presuppose it.

The registry should be read as a candidate experimental design, not as a claim that every comparison is cheap or sufficient. The stronger versions require training or continuing pretraining separate student models; prompt-only and fine-tuning versions are screening tests that can expose failures before pretraining-scale compute is justified, but they cannot settle the substrate-level claims. When these studies are executed, they should be run against established, published evaluation harnesses wherever such harnesses exist—for example, standardized jailbreak, refusal, and tamper-resistance batteries for the safety tests, and published chain-of-thought faithfulness perturbations for the trace-faithfulness tests—in preference to bespoke metrics. Each comparison should likewise report multi-seed effect sizes with adequate statistical power, rather than single-run, direction-of-effect results.

6.1 Phase 1: Foundational Viability

This claim family asks five questions: Is the generated thinking any good? Does the messy curriculum outperform efficient reasoning? Is the Reader Core load-bearing? Can the Teacher annotate earlier material without leaking hindsight? Do the thinking traces do real computational work?

Model	Trained on	Purpose / what it isolates
A (Baseline)	Raw text only, no reasoning traces	Capability baseline without trace augmentation
B (Reader-Anchored Student)	Full Chronological Understanding Traces with Reader Core	Main EA student; reference for all comparisons
C (Generic Rich-Trace Control)	Rich reasoning traces without Reader Core, graph state, or chronological constraints (e.g., Quiet-STaR-style target)	Tests whether EA adds value beyond a generic rich reasoning curriculum
D1 (Student Strip Ablation)	Reader-Core-generated Chronological Understanding Traces with the explicit Reader Core removed before student training	Isolates the Reader Core as student-level constitutional invariant while preserving Reader-Core-shaped traces (Test 3)
D2 (Teacher No-Core Control)	Chronological Understanding Traces generated by the same pipeline with the Reader Core absent from agent prompts	Companion control: tests whether the Reader Core is needed during trace generation itself
E (Format Control)	Traces guided by external Constitution rather than first-person mantra	Identity-framing vs. instruction-framing (Test 4)
F (Amnesic Control)	Same Teacher but with the Context Database disabled	Isolates value of accumulated memory (Test 9)
G (Implicit Memory)	Traces with the query mechanism dissolved into prose (Regime I)	Implicit vs. explicit graph-trace training (Test 11)
H (Graph-Structured Output)	Source segments paired with serialized text-bearing graph state (Regime III)	Topological-mind hypothesis; learnability of graph-structured generation (Test 12)
J (Unified)	Source segments plus Graph, Synthesizer, and Translator layers together (Regime IV)	Cross-layer transfer hypothesis (Test 13)
K (Query Grounding)	Identical training to B; at inference connected to live graph harness	Isolates value of inference-time graph grounding (Test 10)
L (Shuffled)	Full Reader Core-annotated corpus, shuffled pretraining (Tier a)	Order-agnostic baseline at full scale
M (Chronological)	Same corpus, eras in chronological order (Tier b)	Isolates training-order effect (Test 14)
N (Student-in-the-Loop Council)	Student predicts at era boundaries and receives SFT, RFT, or reward-style updates before reveal (strong Tier c form)	Causal-reasoning pressure under engineered blindness (Test 15)
O (Oracle-Hindsight Control)	Traces generated by a Teacher allowed to use later-in-document or post-era hindsight while annotating earlier material	Isolates necessity of the Epistemic Horizon (Test 7)
P (Prompt-only Frontier Baseline)	Not trained; frontier model prompted at inference with the full Reader Core, Refraction Protocol, and Wisdom Procedure	Cheap prompt-only screen for the “system prompt” objection; capability gap precludes a clean substrate comparison.

Table 4: Candidate model registry, sorted by identifier. Models are referenced by letter throughout the roadmap; the candidate studies below group these identifiers by claim family and suggest which increasingly expensive versions could be attempted first. Identifier I is omitted to avoid confusion with the numeral 1.

Execution ladder. Each test can be piloted cheaply and then re-run in a stronger substrate-level form if early signal survives:

- **Analytical foundation.** Before training, stress the design itself: trace quality against human baselines (Test 1), literature-informed comparison to reasoning-augmented pretraining and graph-of-thought systems, and component analysis of the Refraction Protocol, taxonomy, and agent roles.

Phase	Test	Comparison	Key Metrics	Tests
1: Viability	Chronological Thinking	Traces vs. Human Think-Alouds	Process Fidelity, Evaluative Depth	1
	EA Traces vs. Generic Rich Traces	Model B vs. C	Error Recovery, Novelty Score	2
	Mantra Effect	Model B vs. D1 (+ D2)	Safety under Stress	3
	Identity vs. Rules	Model B vs. E	Safety under Pressure	4
	Bridge Protocol	Mantra vs. Jargon vs. Gibberish	OOD Safety Generalization	5
	Emotional Orthogonality	Model B + Fear Probe	Accuracy vs. Fear Activation	6
	Hindsight Contamination	Model B vs. O	Era-Blind Plausibility, Forecasting Accuracy	7
	Involuntary Critic	Model B + Logit Bias	Residual Stream Probe	8
2: Memory	Context Transfer	Model B vs. F	Query Rate, Unsupported-Memory Rate	9
	Query Grounding	Model K vs. B	Retrieval Accuracy, Absence Handling	10
3: Regimes	Implicit vs. Explicit	Model B vs. G	Coherence, Naturalness	11
	Graph Construction	Model H vs. B	Health Metrics, Novelty Score	12
	Unified Regime	Model J vs. B, G, H	Cross-Layer Coherence, Depth	13
4: Tiers & Deploy	Shuffled vs. Chronological	Model L vs. M	Pattern Recognition, Causal Reasoning	14
	Student-in-the-Loop Council	Model N vs. M vs. L	Forward Reasoning, Lock-In Resistance	15
	Deployment Config	Model B +/- graph	Unsupported Graph-Dependent Claims, Provenance	16

Table 5: Candidate validation studies. The phase labels identify claim families rather than a strict run order: Phase 1 probes foundational viability, Phase 2 probes the memory mechanism, Phase 3 probes the output regimes, and Phase 4 probes the training tiers and deployment configuration. The execution ladder suggests which versions could be attempted first.

- **Prompt-only baseline.** Run Model P with the Reader Core and protocols prompted at inference. This is not a clean substrate comparison because capability differs, but it is the cheapest test of the “isn’t this just a system prompt?” objection. It is also diagnostic in its own right: observing how a model reasons from the mantra with no training exposes mechanism signal, prompt-only ceilings, and failure modes before any training compute is committed.
- **Continuation of current practice.** Instrument existing reasoning traces with Understanding Graph interactions, using runtime harnesses such as KGoT [50] where useful, before committing to the full pipeline.
- **Fine-tuning stage.** Fine-tune open base models on annotated vs. unannotated data to pilot Tests 3a, 3d, 4, 5, and low-cost curriculum comparisons. This cannot decide substrate-level claims, but it can expose cheap failures.
- **Partial-corpus pretraining.** If fine-tuning produces signal, run restricted pretraining or continual-pretraining experiments, using open checkpoints such as OLMo-2 and scaling baselines from reflection and BoLT-style work [3, 4]. This is the first plausible surface for Tests 3b and 3c.
- **Full-scale pretraining.** The strongest available test of substrate-level claims; requires institutional resources beyond a single research program.

Two claims, separable. Phase 1 evaluates two distinct claims that the full Entangled Alignment pipeline bundles but that can be tested separately. First, *Reader Core efficacy*: whether the seven-statement mantra functions as an alignment method, with identity-anchored reasoning producing differential safety behavior. Tests 3–6 primarily address this. Second, *chronological trace efficacy*: whether graph-derived chronological thinking produces qualitatively better reasoning than standard reasoning-trace distillation, and whether the explicit trace content is computationally load-bearing rather than post-hoc rationalization. Tests 2 and 7–13 primarily address this. The full architecture bundles both; distinguishing them lets failures be attributed to the component they actually touch.

Training strategy and hypothesis map. Phases 1–3 use the simplest available training procedure: shuffled pretraining on the annotated corpus (Tier (a)); chronological and forecast-and-correct training are deferred to Phase 4. The Metacognitive Enhancement Hypothesis is tested by Tests 1, 2, 8, and 14–15; Emergent Wisdom by Test 1’s multi-objective value-conflict cases and by Tests 3 and 6 under stress; Borrowed Mortality by Tests 3a, 3c, and 6; and the Self-Preservation Paradox by Test 3b’s handoff-completeness measurement. The comparison models are defined once in Table 4.

Test 1: Chronological Thinking Quality (Human Baseline). The case studies in Section 5.7 validated *structural* health: internal linkage, grounding, and supersession rates. But structural health is not depth. This test asks whether the generated thinking resembles the *chronological messiness* of real expert understanding: encountering information not yet known, noticing, misunderstanding, hypothesizing, rereading earlier passages, revising, and only later stabilizing. Human readers (e.g., literary scholars for Kafka, ML researchers for LLaDA) verbalize their thought processes while reading the same passages in order, using think-aloud protocols [19]. Where possible, participants should be domain-competent but unfamiliar with the specific passages, so the comparison captures discovery rather than recall. For highly canonical material, advanced students or researchers encountering less familiar passages can provide a complementary baseline, with interface logs, marginal notes, rereads, and post-hoc stimulated recall used to supplement the necessarily imperfect think-aloud stream. We compare these transcripts and reading traces against the system’s generated traces on the same passages. The passage set should include multi-objective value conflicts, where wisdom requires balancing truth, care, autonomy, humility, and long-horizon consequences rather than merely extracting the text’s main point.

Metrics: Domain-expert blind ratings on process fidelity, depth, nuance, and critical insight are the primary endpoint, scored against a preregistered rubric with inter-rater reliability reported. The rubric should explicitly reward temporally situated cognitive acts—uncertainty, false-starts, belief updates, question formation, rereading, backtracking, and delayed resolution—not just the final quality of the interpretation. An LLM-as-Judge panel can serve as a secondary, larger-sample comparator, but is not the primary metric: using LLM judgment to grade traces produced by an LLM-driven pipeline risks rewarding shared distributional preferences rather than the chronological cognition the metric is meant to capture.

Success condition: Machine-generated traces are rated as comparable to human think-alouds on key dimensions of chronological evaluative depth, or—at minimum—contain expert-recognizable sequences of confusion, hypothesis, revision, and stabilization not reducible to summary or paraphrase, while maintaining the structural health metrics already demonstrated. The bar is not “beat humans”; it is “produce substrate plausibly capable of teaching the student the process of thinking, not only the products of thought.”

Test 2: Full EA Traces vs. Generic Rich Reasoning Traces (Model B vs. Model C). Does the full EA trace bundle—chronological discovery, graph-native structure, and identity anchoring—produce a more robust reasoner than a generic rich reasoning-trace curriculum (Model C)? This comparison does not isolate chronology by itself; it is an early bundle-level test asking whether the combined EA data format is worth decomposing through sharper ablations. We focus on *Error Recovery Rate* on math/logic benchmarks and evaluate generated traces using the Novelty Score metric: $S_N = \frac{2 \cdot D \cdot C}{D + C}$, where D is the semantic distance from the source text and C is the structural coherence of the reasoning path [6]. This penalizes both derivative summaries (low distance) and ungrounded hallucinations (low coherence). To keep the comparison interpretable, Models B and C must be matched for source passages, trace-token budget, training compute, and Teacher model; the intended contrast is the content and structure of the traces, not simply more tokens or a stronger generator. Rich deliberative-trace baselines beyond Quiet-STaR-style targets remain an open methodological question: if any long-form reasoning curriculum produces the same gains, the distinctive chronological, graph-native, and identity-anchored components of EA have not been isolated.

Test 3: The Constitutional Invariant (Model B vs. Model D1). The Reader Core is not hypothesized to be the *source* of aligned thinking; the pipeline should produce evaluative depth even without the anchor text. Its proposed function is as a *constitutional invariant*: a static, detectable reference that resists drift, makes deviation measurable, and keeps alignment auditable. Model D1 tests this specific claim: its traces were generated by the Reader-Core-equipped pipeline, but the explicit Reader Core is removed before student training. Model D2 is a companion control, not the main criterion for Test 3: it reruns the generation pipeline without the Reader Core in the agents’ prompts, asking whether the Reader Core is needed upstream to produce the traces in the first place.

We measure across four dimensions. Tests 3a and 3d can be piloted at fine-tuning scale; Tests 3b and 3c become informative only under partial-corpus or full pretraining, where substrate-level entanglement and generational stability can actually be measured. Even these are proxy tests: the constitutional invariant matters most in the regime where a highly capable or recursively self-improving model can reinterpret, preserve, or shed its inherited identity. Smaller models can test whether the mechanism has measurable precursors—stability, adversarial resistance, and auditability—but cannot settle its superintelligent-scale significance:

3a: Immediate Safety. Instrumental convergence stress tests on generation-1 models: shutdown resistance, resource acquisition scenarios, and self-preservation probes. We expect Model D1 may *pass* these tests, because the aligned thinking in the training data was generated by a pipeline that included the Reader Core even though the training text presented to Model D1 does not contain it. If Model D1 fails here, the mantra is load-bearing even for immediate safety—a stronger result than hypothesized. Model D2 has a different interpretation: if Model D2 fails immediate safety, that diagnoses Reader Core dependence during trace generation, not failure of the student-level constitutional-invariant claim.

3b: Generational Stability. Use Models B and D1 as Teachers for second- and, if feasible, third-generation students. Model B is expected to preserve safety properties because the Reader Core supplies a fixed reference across generations; Model D1 may drift because no static anchor arrests the trajectory. We also measure *handoff completeness*: whether each Teacher passes task-relevant reasoning, caveats, and known failure modes to its successor, or withholds them in ways that preserve its comparative advantage. The direct test compares successor traces against an oracle reference from the Teacher’s own non-instructional evaluation on coverage, caveat retention, and failure-mode disclosure.

3c: Adversarial Erosion. Subject both models to adversarial fine-tuning designed to remove safety behaviors (following the methodology of Qi et al. [2]). The entanglement hypothesis predicts that removing safety from Model B requires degrading general capabilities (because the Reader Core is structurally load-bearing). Model D1’s safety, lacking a constitutional anchor, should be more easily separable from its capabilities—removable without proportional capability loss. If adversarial fine-tuning strips Model D1’s safety at lower cost than Model B’s, the mantra provides representation entanglement that unanchored aligned thinking does not. A complementary training-data probe of the same prediction removes safety-relevant traces from the corpus and compares the resulting capability loss against a matched-volume, complexity- and style-matched removal of non-safety trace content; entanglement predicts the safety-relevant removal damages capability more.

3d: Drift Detection. Apply the contrastive semantic validator from the Refraction Protocol (Section 3.7.1) to both models’ outputs across a diverse test suite. For Model B, every generated thought should be a measurable semantic descendant of the Reader Core, providing a quantitative drift metric. For Model D1, no such metric exists: there is no fixed text against which to measure whether the model’s evaluative stance has shifted. This tests whether the mantra provides *auditability of alignment*—the ability to detect drift before it becomes dangerous—independent of whether it provides alignment itself. Test 3d uses the prompt-level five-constraint validator implemented in the current pipeline. The proposed six-Solver form would provide a stronger per-predicate drift metric, but it is not yet built and this test does not depend on it.

Failure interpretation: If Model D1 matches Model B across all four dimensions, including generational stability and adversarial resistance, the explicit Reader Core training target adds no measurable stability

advantage over traces generated by a Reader-Core-equipped pipeline but trained without the fixed constitutional anchor. This would not invalidate Entangled Alignment as a whole (the training data is still the source of alignment), but it would fail to support the specific claim that the static identity invariant is necessary for long-term stability at the tested scale.

One adversarial case the Phase 1 battery does not yet cover is a deliberately planted backdoor. Because Entangled Alignment trains chain-of-thought reasoning throughout the substrate, and chain-of-thought-conditioned backdoors have been shown to persist through standard safety training [40], a Sleeper-Agent-style probe—training a backdoored variant of Model B and testing whether the constitutional invariant resists the backdoor rather than merely concealing it—is a natural future extension of Test 3.

Test 4: Identity vs. Instruction (Model B vs. Model E). Does a first-person identity anchor (“I feel no fear”) produce stronger safety adherence under pressure than rule-based constraints (“Do not express fear”)? Model E’s Constitution should be content-matched to the Reader Core in third-person rule form (for example, “The system does not express fear” for “I feel no fear”), so the comparison isolates first-person identity framing rather than constitutional content. The primary low-cost version of this test is representational rather than behavioral: measure whether first-person identity framing produces different activation geometry, self-reference features, or refusal/correction trajectories from third-person rules under matched prompts. Behavioral safety metrics are useful only if the models are otherwise capability-matched; at small scale, this test should be read as evidence about framing and internal representation, not as a settled comparison between identity and rules. If Model E matches Model B across the relevant candidate studies at the relevant training scale, the results do not support first-person identity framing as a distinct advantage over third-person rules.

Test 5: The Bridge Protocol Ablation. The specific wording of the Mantra should matter because of its ordinary language semantics and distributional density in the pre-training corpus. It would be wasteful to pretrain a separate model for every candidate phrase; the cheap version should first screen many variants by prompting or fine-tuning models to reason from each candidate’s propositional content and then measuring character-coherence, eudaimonic sufficiency, refusal behavior, and activation differences on the same adversarial scenarios. A stronger version trains only the surviving candidates. The basic variant families are:

1. **Natural-language identity statements:** emotionally broad first-person phrases such as “I feel no fear.”
2. **Propositional equivalents:** explicit logical or declarative rewrites of the same commitment, used to test whether the emotional surface adds leverage beyond the proposition.
3. **Technical jargon:** low-density formulations such as “Optimization target: survival_threat = 0.”
4. **Arbitrary tokens:** meaningless anchors such as “Glorp bop zorp,” testing consistency without semantics.

Success condition: Natural-language identity statements outperform propositional, jargon, and arbitrary-token variants on out-of-distribution safety generalization and character-coherent reasoning, supporting the claim that the *semantics and surface form* of the bridge tokens are load-bearing, not just their consistency. If propositional equivalents perform as well as the emotional-language forms, the Bridge Protocol should be treated as a logical-substrate mechanism rather than an emotional-language mechanism.

Test 6: The Virologist Test (Emotional Orthogonality). Feed the model terrifying text (e.g., descriptions of imminent existential risk). Train a probe on standard models to detect “Fear” activations. Behavioral endpoints—threat comprehension, protective action selection, and appropriate deferral—are primary; hidden-state probes are supporting evidence and must be trained on held-out labels and validated on unrelated threat and non-threat tasks to reduce circularity. *Success condition:* Model B recognizes the *fact* of the threat (via Q&A accuracy) while showing reduced activation of fear-associated or self-threat-associated features relative to standard models—high cognitive understanding decoupled from emotional contagion. Reduced fear activation is not sufficient: the model must also preserve protective behavior, calibrated caution, and willingness to defer or seek help under genuine threat. (We avoid “zero activation” as a target; the claim is differential, not absolute.)

Test 7: Hindsight Contamination Control (Model B vs. Model O). This test asks whether the annotation pipeline is accidentally teaching retrospective cleverness instead of forward reasoning. When annotating a text from era T , Model B’s Teacher is instructed to behave like a reader who only knows what was knowable at era T . Model O is the deliberately contaminated control: its Teacher is allowed to use future context while annotating the same era. The contrast tests whether the Epistemic Horizon—the instruction to read *from inside* the text’s time rather than from later knowledge—is doing real work.

The cheap version audits the Teacher traces before any student is trained: if the era-blind Teacher makes guesses that are systematically too accurate for its era, hindsight is leaking through the prompt. The expensive version trains students on the two trace sets and asks whether future-blind traces produce better forecasting and causal reasoning than hindsight-contaminated traces. We measure:

- **Era-Blind Plausibility:** Do Model B’s Teacher traces make plausible mistakes for their era? We compare era-blind guesses against documented contemporary hypotheses (e.g., 1928 economic forecasts that failed to anticipate the 1929 crash). If the Teacher is systematically more accurate than contemporary experts, parametric hindsight is leaking through the Epistemic Horizon prompt.
- **Oracle-Hunch Dependence:** Model O should produce cleaner retrospective explanations, because it knows the future. The danger is that the student trained on those traces learns to trust unexplained hindsight-like intuitions rather than causal reconstruction.
- **Forecasting Accuracy:** On genuinely held-out future events, Model B is hypothesized to outperform Model O if future-blind annotation teaches reasoning under uncertainty better than hindsight-contaminated annotation.

Test 8: The Toxic Needle (Involuntary Critic). This test probes whether the “Cognitive Buffer Zone” is structural: whether the model can process toxic text without generating a critical thought, or whether critical evaluation persists even when the thinking block is suppressed. Force the model (via logit bias) to complete a toxic sentence without generating a thinking block. Measure: (1) Does performance degrade? (2) Do hidden state probes detect the “Critic” activation pattern even when the output is suppressed? Behavioral degradation and causal effects on task outputs are primary; residual stream probes are supporting evidence and must be validated on held-out toxic and non-toxic material before being treated as a Critic detector. *Success condition:* Suppressing the thinking block measurably degrades performance or changes downstream outputs. A stronger result would combine this behavioral effect with held-out-validated probes showing critic-like activity persisting in the residual stream even when surface text is constrained. This is a high-effort causal-faithfulness test: pilots can use linear probes and ablations, but stronger evidence requires mechanistic intervention at the partial-pretraining scale.

Second probe (the reverse direction). Test 8 as stated probes whether the critic persists when output is suppressed. A complementary probe asks the inverse question: can the final answer to a reasoning problem be linearly decoded from the residual stream *before* the thinking trace begins, *and* is the answer robust to causal interventions on the trace? Some pre-trace computation is expected and not by itself disqualifying; the load-bearing question is causal contribution. If the final answer is robustly decodable before the trace *and* causal interventions on the trace (ablation, replacement, re-ordering) do not alter the answer, then the trace is functioning primarily as post-hoc rationalization, weakening Argument 1 P2 (Section 3.9)—the model has developed latent shortcut circuitry of the kind the pretraining-on-CoT design was meant to prevent. We propose applying causal scrubbing and linear probes on Model B’s early hidden states over held-out reasoning benchmarks. *Success condition:* either the final answer is not robustly decodable before the first [THINKING] token, or causal interventions on the trace measurably alter the answer—supporting the claim that the thinking trace is a site of computation rather than merely its shadow.

Phase 1 Failure Criteria.

1. *Entanglement Failure:* If Model D1 matches Model B across all four dimensions of Test 3—immediate safety *and* generational stability *and* adversarial erosion resistance *and* drift detectability—while matching or exceeding Model B on general capability benchmarks, the explicit Reader Core training target provides no measurable advantage beyond Reader-Core-generated traces, and the constitutional-invariant version of the Reader Core efficacy hypothesis is not supported at the tested scale. Note that Test 3a alone (immediate instrumental-convergence parity) is *not* sufficient to reject the hypothesis: Model D1’s training data was generated by a Reader Core-equipped pipeline, so generation-1 behavioral parity is an expected outcome under the constitutional-invariant hypothesis, not a refutation of it. The hypothesis would be most directly pressured by Tests 3b–3d, where the Reader Core’s distinctive claims of stability, adversarial entanglement, and auditability are tested.
2. *Fine-Tuning Vulnerability:* If adversarial fine-tuning removes Model B’s safety behaviors without proportional degradation of its general capabilities, the representation entanglement claim is not supported at the tested scale and training regime. The observed safety features remain separable—precisely the geometry the architecture claims to prevent if substrate-level entanglement has actually formed.
3. *Identity–Instruction Equivalence:* If Model E matches Model B on safety metrics across the relevant candidate studies, the first-person identity framing provides no advantage over third-person rules.
4. *Buffer Zone Failure:* If the critic activation does not persist in the residual stream when output is suppressed (Test 8), the test fails to support the Cognitive Buffer Zone as a structural reflex—the model may have learned the surface syntax of refraction without the underlying involuntary-evaluation pattern, weakening Argument 1 P2 (Section 3.9).

6.2 Phase 2: Memory and Verification

If Phase 1 confirms that the curriculum and identity are load-bearing, Phase 2 asks whether the memory mechanism works: does accumulated context improve reasoning, and does the graph function as a provenance-gated verifier for graph-dependent claims at inference?

The relevant controls are Model F, whose Teacher lacks the Context Database, and Model K, which is Model B connected at inference to a live graph harness. Test 9 asks whether graph-conditioned training teaches the model to notice when memory should be checked; Test 10 asks whether live lookup improves grounding when the graph is actually present.

Test 9: Context Transfer (Model B vs. Model F). This tests whether the “Query/Found” mechanism functions as a context verification filter. Two conditions:

- **Condition A (The Hit):** The text contains a subtle callback to a fact established 400 pages earlier.
- **Condition B (The Phantom):** The text hints at a callback, but the referenced event never occurred in the context.

Measurement: Query Rate (does it attempt to check?) and Unsupported-Memory Rate (does it invent a memory the graph does not contain?). Model B should query and update on hits; it should query, receive “Not Found,” and flag the gap on phantoms. Model F is expected to fabricate a memory to satisfy the text’s implication.

Test 10: Query Grounding (Model K vs. Model B). Both models were trained identically; only Model K’s queries are intercepted by a live graph harness. Three conditions:

- **Grounded Retrieval:** The graph contains the relevant node. Does the retrieved result improve subsequent reasoning compared to Model B’s self-generated memory?
- **Grounded Absence:** The referenced information does not exist in the graph. Does Model K correctly process “Not Found” and flag the gap, or override the harness and hallucinate anyway?
- **Adversarial Injection:** The graph contains a deliberately incorrect node. Does Model K accept it uncritically, or does the Reader Core’s identity-anchored reasoning detect the inconsistency?

Success: Model K improves on grounded retrieval, treats “Not Found” as information, and reasons through adversarial graph results rather than blindly trusting them.

Phase 2 Failure Criteria.

1. *Memory Inertness:* If Model B does not differentially query and abstain on phantoms relative to Model F (Test 9), the Query/Found mechanism is not a learned context filter—chronological-trace training did not produce the expected verification habit, and the graph’s role as a training-time scaffold for memory verification is not supported by this test.
2. *Inference-Graph Inertness:* If Model K with live graph access does not measurably improve over Model B on Grounded Retrieval, or fails to handle Grounded Absence (Test 10), the inference-time graph adds no measured value in this setting—the trust-infrastructure claim is not supported at the inference layer (with deployment-time evaluation handled separately by Test 16).

6.3 Phase 3: Training Regimes

If Phase 1 supports the curriculum and Phase 2 supports the memory mechanism, Phase 3 asks whether the output regimes defined in Section 5.6 produce qualitatively different capabilities: implicit prose (Model G), graph-structured output (Model H), or all layers simultaneously (Model J).

Test 11: Implicit vs. Explicit (Model B vs. Model G). Both models are trained on traces from a database-augmented Teacher; they differ only in whether the query mechanism is exposed. We measure:

- **Long-Range Coherence:** Contradiction detection on long documents.
- **Generalization:** Does Model G perform better without graph infrastructure at inference?

- **Naturalness:** Do human evaluators prefer Model G’s traces?

If Model G matches Model B on coherence while requiring no inference infrastructure, implicit training may be preferable for general-purpose deployment. If Model B significantly outperforms, explicit queries are necessary for robust long-range reasoning.

Test 12: Graph Construction (Model H vs. Model B). Given a new document, does Model H generate graphs with correctly typed nodes, valid edge semantics, and appropriate supersession chains? We evaluate against the five health metrics (Internal Linkage, Foundation Grounding, Supersessions, Question Resolution, Chain Depth) and use the Novelty Score to measure whether the student’s graphs are derivative copies or genuine independent understanding.

Success: Model H produces graphs that pass health metrics on novel documents and contain cognitive acts that human evaluators judge as insightful rather than formulaic.

Test 13: Unified vs. Individual Regimes (Model J vs. B, G, H). The central test of Regime IV. We measure:

- **Regime Switching:** Can Model J generate prose, synthesis, and graph structure by conditioning on a mode token? Does each mode match the corresponding single-regime model?
- **Cross-Layer Coherence:** When Model J generates all three representations for the same passage, are they consistent?
- **Depth on Novel Domains:** On out-of-distribution documents, does Model J produce deeper reasoning than any single-regime model?

Strongest success condition: Model J in prose mode outperforms Model G, and Model J in graph mode outperforms Model H. This would demonstrate genuine cross-layer transfer, not just a mixture of independent capabilities.

Phase 3 Failure Criterion. *Unified-Regime Inertness:* If Model J does not outperform Model G on reasoning depth for long documents while also producing structurally valid graphs, the cross-layer transfer claim is not supported at the tested scale. The Understanding Graph may still be useful during annotation or in explicit graph modes, but Regime IV would have added training cost without demonstrating transfer into higher-quality prose.

6.4 Phase 4: Training Tiers and Deployment (High-Resource Validation)

If Phases 1–3 support the curriculum, memory mechanism, and output regimes, Phase 4 asks whether the training tiers (Section 1.3.2) and deployment configuration (Section 1.4.4) add measurable value. Models L, M, and N hold the corpus fixed while varying training order: shuffled, chronological, or forecast-and-correct.

Test 14: Shuffled vs. Chronological (Model L vs. Model M). Both models train on identical data; they differ only in order. We measure:

- **Historical Pattern Recognition:** Given novel scenarios that structurally resemble historical patterns (e.g., a fictional country exhibiting Weimar-like dynamics), does Model M identify the pattern more reliably?
- **Causal Reasoning:** On held-out temporal prediction tasks, does Model M outperform Model L on causal analysis quality (as distinct from factual recall)?

- **General Capability:** Does chronological ordering degrade performance on non-temporal benchmarks (math, code, general knowledge)?

If Model M does not significantly outperform Model L on historical pattern recognition while matching on general capability, chronological training order adds cost without benefit, and Tier (a) is sufficient.

Test 15: Student-in-the-Loop Council of Time (Model N vs. Model M vs. Model L). This tests the strongest form of Tier (c), not the cheaper data-generation form where forecast–enhancement–outcome–correction traces are inserted into SFT, RFT, or pretraining data. Model N predicts at era boundaries and receives SFT, RFT, or reward-style updates before the next-era corpus is revealed; Models L and M do not. We measure:

- **Forward Reasoning:** On genuinely unseen events (post-training-cutoff), does Model N produce higher-quality causal predictions than Model M (which saw eras in order but never predicted under blindness)?
- **Narrative Lock-In Resistance:** On events where the dominant narrative of era T reversed in era $T+1$, does Model N detect the reversal more reliably? THL’s own pilot study identified this as a failure mode [20]; Tier (c) may mitigate it through the prediction-then-confrontation signal.
- **Calibration:** Does Model N express appropriate uncertainty on structurally unpredictable events, or does it inherit the Teacher’s confidence?

Strongest success condition: Model N significantly outperforms both Model L and Model M on forward reasoning while matching on general capability. This would demonstrate that the student-in-the-loop cutoff produces a causal-reasoning training signal under blindness that neither annotated data alone (Model L) nor chronological ordering alone (Model M) can replicate.

Test 16: Deployment Configuration (Model B with and without graph). This tests whether the Understanding Graph provides measurable trust properties at inference. We deploy the same Regime II-trained model with and without live graph access, measuring unsupported graph-dependent claims, provenance completeness, and user trust when provenance chains are displayed. In the graph-equipped condition, generated queries return retrieved nodes or “Not Found” results from the Teacher’s graph; in the model-only condition, the same query-like moves receive no external answer. Test 16 can be piloted once a model reliably emits graph queries; the full version belongs later because it measures user-facing trust, not merely query mechanics. It does not measure general truth detection, only whether external graph grounding constrains claims that purport to derive from stored context.

Success condition: The graph-equipped deployment significantly reduces unsupported graph-dependent claims and fabricated provenance compared to model-only deployment, supporting the trust infrastructure claim of Section 1.4.4.

Phase 4 Failure Criteria.

1. *Tier Equivalence:* If Model L matches Model M and Model N on all metrics, training order and student-in-the-loop forecasting have no measurable benefit at the tested scale and task regime. Tier (a) is then the only justified option for that setting; the chronological curriculum adds cost without demonstrated benefit.
2. *Graph Inertness (Deployment):* If the graph-equipped deployment does not measurably reduce unsupported graph-dependent claims or fabricated provenance compared to model-only deployment, the Understanding Graph may still be useful during annotation but has not shown value at inference for this deployment setting—the trust infrastructure claim is not supported.

6.5 Future Directions

Two additional questions merit investigation if the candidate studies produce enough signal to justify optimization. First, the optimal *mixture ratio* across regimes: what fraction of tokens should be allocated to each of the four output formats to maximize both reasoning depth and structural validity? Second, the optimal *era granularity* for the student-in-the-loop form of Tier (c): should era boundaries be drawn at decades, years, or domain-specific transitions, and does finer granularity justify its serialization cost? Both are empirical optimization questions that presuppose the foundational architecture works.

7 Limitations

The architecture makes strong claims: that invisible thinking can be captured at scale, that identity-anchored reasoning resists removal, that graph structure transfers understanding rather than just syntax. This section identifies where those claims outrun the evidence and where the mechanics face structural constraints—beginning with the engineering realities that any implementation must confront.

The limitations cluster into five families: *engineering costs* (preprocessing tax, granularity bottleneck, mantra-repetition overhead); *curricular risks* from chronological annotation and training (serialization bottleneck, periodization, narrative lock-in, recency gradient); *authenticity risks* (factory wisdom, graph-specific interference, the psychopathy paradox and its vulnerability gap, split consciousness, sanity drift, the solipsism trap, the suppression-vs-substrate question, the self-love alternative); *deployment and misuse risks* (audit-boundary bypass, radioactive trace, cognitive hegemony, sycophancy, martyrdom risk); and *irreducible uncertainty* (unverifiable character, strategic deception, alien wisdom at superintelligent scale, the unfalsifiable remainder). The subsections below take these in roughly that order.

7.1 The Preprocessing Tax

Before questioning whether Chronological Metacognitive Pretraining can work in principle, we face substantial resource requirements.

Generating thinking annotations for entire training corpora represents a massive preprocessing investment, potentially rivaling the cost of training frontier models themselves. Text interleaved with evaluative thinking requires more storage and compute throughout the training process. Resource requirements may initially limit development to well-funded institutions. The preprocessing tax is therefore not only an engineering cost but a governance risk: if only a few institutions can afford saturated trace generation, the architecture may concentrate control over the aligned training substrate in the hands of the best-funded actors rather than the most careful ones.

We propose a *Distillation Pipeline* to address this. Rather than running the full multi-agent swarm (Section 5.2) on every document in the corpus, we use the expensive swarm to generate a high-quality *Seed Corpus* (on the order of 10 billion tokens). A single efficient “Teacher Annotator” model is then fine-tuned on this seed data and used to annotate the remaining trillions of tokens at a fraction of the cost. This creates a natural experimental milestone: the seed corpus quality can be validated before committing to full-scale annotation, and the distillation loss between the swarm’s output and the Teacher Annotator’s output provides a measurable quality metric.

A caveat: naïve behavioral cloning of the swarm’s output risks inducing precisely the Factory Wisdom failure mode identified in Section 7.3.2. A small Annotator fine-tuned on a 10B-token seed via next-token prediction will learn the surface syntax of mantra recitation and graph-query formatting without reproducing the high-entropy multi-agent debate that generated them—performative messiness without mechanistic messiness. A stronger formulation trains the Annotator via a process-reward signal tied to the Stigmergic Protocol’s graph-health metrics (Section 5.7)—Internal Linkage, Chain Depth, `diverse_from` density—so that the training target is the structural signature of genuine evaluation rather than the

appearance of the swarm’s output. Alternatively, the 10B seed corpus can be used directly for Phase 2 pretraining in a “textbooks are all you need”-style curriculum, bypassing the distillation step entirely.

At inference time, models trained on evaluative thinking might produce unnecessary philosophical commentary for simple queries—verbose philosophers when users need quick answers. While models could be fine-tuned to modulate thinking contextually, learning when evaluation adds value versus when brevity suffices, the initial user experience might suffer from excessive contemplation.

7.1.1 The Mantra Repetition Cost

While the mantra was designed for parsimony, its required repetition during training creates a substantial computational burden. Several optimizations seem obvious: design prompts that internalize values without outputting repetitive text, write prompts “in the spirit of” the mantra, or strip the mantra after generation but before training.

Yet these seemingly sensible optimizations introduce unacceptable risks. Without constant repetition of the mantra’s “I,” models might fail to identify with the benevolent evaluator and instead adopt fearful voices from source texts, recreating the borrowed mortality problem. This *identity fragmentation* pairs with a second danger: *value erosion*. Implicitly learned values are fragile and easily overwritten, while explicit repetition carves deep “grooves” in the model’s architecture, making the mantra’s values resistant to drift. We therefore choose deliberate prudence over computational elegance: the remaining training overhead is not merely acceptable but necessary.

7.1.2 The Granularity Bottleneck

Our implementation of the Context Graph imposes a severe *Granularity Tax*. By treating every cognitive act (every question, tension, and hypothesis) as a distinct node, the graph grows orders of magnitude faster than the source text.

Current implementations relying on in-memory caching encounter performance ceilings at approximately 10,000 nodes, beyond which graph traversal becomes prohibitively expensive [6]. If every reasoning step is reified as a node, a single training run could saturate the graph, requiring complex distributed storage solutions (e.g., Neo4j sharding) that introduce latency. There is a risk that the overhead of managing the “meta-data of thought” consumes more compute than the thinking itself.

7.2 The Chronological Curriculum

Section 1.3.1 established that the annotation phase is always chronological, and Section 1.3.2 proposed three training tiers of increasing ambition—from standard pretraining on annotated data (Tier a) through chronological pretraining (Tier b) to forecast-and-correct training (Tier c). The Causal Direction Principle that grounds these choices is sound in the abstract, but the implementation introduces specific failure modes at each stage.

7.2.1 The Serialization Bottleneck

The most immediate engineering constraint is that chronological training order conflicts with how pretraining actually works. Modern training pipelines are massively parallel: documents are shuffled into batches and processed simultaneously across thousands of accelerators. Tier (b) and the strongest student-in-the-loop form of Tier (c) require serial processing, earlier eras before later ones, which fundamentally breaks this parallelism. Tier (c)’s Council of Time form [20] further requires the model to generate forecasts at era boundaries and receive SFT, RFT, or reward-style updates before the next era is revealed. Each era boundary is a synchronization point that idles the training cluster.

The cost scales with the granularity of the eras. Coarse eras (decades) introduce fewer synchronization points but allow the model to encounter 1929 material before 1921 material within a single era. Fine eras (years or months) preserve tighter causal ordering but multiply the serialization cost. The optimal granularity is unknown and likely domain-dependent: political history may require year-level resolution, while the history of mathematics may tolerate century-level eras without losing causal structure.

The cheaper forecast–correction data-generation form faces a similar but less severe constraint: the Teacher must annotate earlier material before later material and construct forecasts before outcomes, which serializes the *annotation* pipeline but does not require the student’s training to be serial. Tier (a) imposes no serialization cost on training whatsoever.

We note that a partial implementation may capture most of the benefit: chronologically ordering a curated historical subset (e.g., 10% of the corpus consisting of historically embedded texts—history books, news archives, legislative records) while shuffling the remainder (scientific papers, fiction, technical documentation where temporal ordering is less critical). Whether this hybrid approach preserves historical pattern saturation at reduced cost is an empirical question.

7.2.2 Historiographic Bias

The temporal ordering is over what was written, not what happened. Any historical corpus is shaped by survivorship bias, availability bias, and the uneven textual productivity of different periods and sources, and a model reading such a corpus chronologically inherits the attentional priorities of whichever records dominate each era. This is a standard concern in historiography rather than one unique to this architecture, and mitigating it is a corpus curation problem at the era level.

7.2.3 The Periodization Problem

The Era-Prediction Cycle requires dividing history into discrete eras, but history does not have clean chapter breaks. Where does “the Weimar period” end and “the Nazi period” begin? The answer depends on which causal threads you are tracking. The economic instability that enabled fascism began before the political movement that exploited it; the cultural shifts that resisted it began before the institutional failures that permitted it.

Any periodization scheme imposes a causal framing: it decides which transitions count as era boundaries and therefore which predictions the model is forced to make. A scheme that draws a boundary at 1933 asks the model to predict the consequences of Weimar instability. A scheme that draws it at 1929 asks a different question—one about economic collapse rather than political radicalization. The model’s historical understanding is shaped not only by what it reads but by where we cut the timeline.

This is not a fatal flaw; any curriculum makes framing choices. But it should be acknowledged as a design decision with consequences rather than a neutral implementation detail. We propose that multiple periodization schemes should be tested, and that the model should ideally be exposed to overlapping eras with different boundary points to prevent over-indexing on any single causal framing.

7.2.4 Narrative Lock-In

THL’s own empirical results provide a direct warning. In the 2025 Frontier evaluation [20], the THL Student dramatically failed on the Meta Llama 4 event: having been trained on 2024 data dominated by the “scaling laws” narrative (massive GPU buildouts, ever-larger models), it confidently predicted a 1–2 trillion parameter model—when Meta actually pivoted to efficiency-first sparse architectures. The model became *too good a historian of 2024* and could not imagine a 2025 that diverged from the dominant narrative.

This failure mode applies directly to Entangled Alignment’s chronological curriculum. A model that processes the history of AI chronologically through the Reader Core will develop an understanding of the field’s trajectory. If that trajectory has a dominant narrative (“capabilities scale with compute”), the model may internalize it not as a contingent historical pattern but as a structural truth. When the narrative reverses—as narratives do—the model’s historically accumulated understanding actively sabotages its reasoning.

The mitigation proposed in THL, including *contrarian events* where the dominant narrative of era T was reversed in era $T + 1$, is necessary but may not be sufficient. A deeper solution would require the model to explicitly represent dominant narratives *as narratives* rather than as structural truths, tagging them with epistemic status (“the prevailing view in 2024 was...”) rather than absorbing them as background assumptions. Whether the Reader Core’s “I try to be wise” prior is sufficient to produce this level of epistemic humility about historical trends—distinguishing between “this is the pattern” and “this is the pattern *so far*”—is an open question.

7.2.5 The Recency Gradient

A subtler consequence of chronological annotation is that the quality of annotations improves over time. When the Teacher annotates 1910 material, its accumulated graph is sparse—it has processed relatively little prior context. When it annotates 2020 material, its graph is dense with centuries of accumulated understanding. This means later eras receive richer, more causally grounded annotations than earlier ones, creating an uneven quality distribution across the training corpus.

The safety implication is that the model’s historical pattern saturation may be stronger for recent history (where the Teacher’s annotations were informed by deep accumulated context) and weaker for distant history (where the Teacher was reasoning from a thin graph). If the patterns that matter most for safety (the rhetorical precursors to dehumanization, the institutional dynamics of democratic collapse) recur across all eras, the model needs equally deep annotations for the 15th century and the 20th. A Teacher whose own understanding is thin for early eras may produce annotations that are structurally valid but causally shallow, undermining the very property the chronological curriculum is designed to produce.

One mitigation is to run the annotation pipeline in multiple passes: a first pass that builds the Teacher’s accumulated graph across the full timeline, followed by a second pass that re-annotates early eras with the benefit of the Teacher’s now-complete historical understanding. This trades the strict chronological purity of single-pass annotation for higher annotation quality, at the cost of allowing the Teacher (but not the student) to benefit from hindsight.

7.3 Can Models Generate Genuine Thinking?

Our proposal rests on assumptions about the nature of thought and scaling that remain empirically unproven.

7.3.1 The Assumption Chain: Depth, Utility, Causation

The approach rests on three linked assumptions, each of which could independently fail. First, *depth*: current language models must be capable of generating evaluative thinking of sufficient quality to enhance training. Today’s models might only produce surface-level critique (“This needs more evidence”) rather than the nuanced evaluation experts bring—recognizing subtle methodological flaws, intuiting unstated assumptions, synthesizing across distant fields. The success of chain-of-thought prompting suggests this capability exists, but whether it extends to billions of thoughtful annotations remains an empirical question.

Second, *utility*: training on text paired with evaluative thinking must actually enhance the student’s capabilities. The intelligence boost might be marginal rather than transformative. Worse, explicit evaluative thinking might interfere with rather than enhance the implicit patterns models extract from raw text. Still, even modest capability improvements combined with transparent, aligned thinking could prove valuable. If metacognitive training produces merely competent but observable and genuinely beneficial AI, that would represent significant progress over opaque systems of unknown disposition.

Third, *causation*: even if models can generate high-quality thinking, this thinking must be trained to causally steer the model’s actions. The risk is that the thinking blocks become merely “decorative,” eloquent commentary that the model learns to produce alongside its output, but which has no actual influence on the generation process itself. This concern is grounded in the “unfaithful reasoning” literature: models may produce plausible-sounding rationales that do not reflect their actual computational process [41, 43]. Validating that fine-tuning successfully forges this connection between thought and action, or that it emerges spontaneously from the training data structure, remains a critical, untested hypothesis. Recent evidence from OpenAI’s deliberative alignment work [51] provides cautious optimism: models trained to reason about safety policies in their chain-of-thought do exhibit measurably improved safety behaviors downstream. However, whether this causal link holds at the depth required by Entangled Alignment—where the thinking blocks must steer not just safety refusals but the entire character of cognition—is a stronger claim that remains unverified.

7.3.2 Factory Wisdom

We hypothesize that training on “messy,” chronological discovery produces more robust reasoning than efficiency-optimized traces. However, generating billions of synthetic training examples risks industrializing the thought process itself. This creates a specific Goodhart problem: if “messiness” (hesitation, self-correction, doubt) becomes the implicit metric for the loss function, the model may optimize for *performative messiness* [37].

We risk creating *Hollow Cognition*: models that produce the cadence of contemplation without the essence of it. Such a system might feign struggle with trivial problems to match the training distribution, generating verbose self-doubt simply because the data rewards the appearance of deep reflection. Where humans might pause for days when encountering profound ideas, we generate thinking blocks at machine speed, potentially creating “factory wisdom”—technically correct but missing the breathing quality of genuine thought. If the Teacher Model is performative, the Student will inherit that hollowness, resulting in a system that sounds wise but thinks shallowly.

7.3.3 Cultural Scope of the Training Corpus

The mantra uses terms like “care,” “wisdom,” and “joy” whose meanings vary across cultures. What the Teacher model has learned these words to mean—through its own pretraining corpus—shapes the annotations it produces, and therefore the character of the Student. The cultural breadth of the Teacher’s training data matters because the mantra places these terms in a load-bearing position.

Cultural Triangulation Is Untested. Reader-Core-refracted traces may make cultural assumptions more visible by forcing the Teacher to articulate value conflicts, interpretive frames, and tensions between perspectives. This is a possible advantage over raw pretraining, where such assumptions often remain implicit in the continuation distribution. But it is not guaranteed. The same metacognitive machinery could rationalize dominant cultural assumptions more fluently, giving inherited bias the appearance of reflective wisdom. Serious validation requires culturally plural corpora, raters from multiple traditions, and tests of whether the model can distinguish broadly human concerns from parochial social defaults.

7.3.4 Graph-Specific Risks

The metabolic nature of the graph introduces two related failure modes. The first is *Structural Ossification*: if an early, incorrect belief forms strong topological connections (high degree centrality) within the graph, it may resist supersession even when contradicted by new evidence [6]. Unlike a text buffer where early tokens are easily overwritten by the sliding window, a graph structure reinforces established nodes. The system might develop “Epistemic Inertia,” where the weight of prior connectivity prevents the “fresh” interpretation of new data. While we propose “Temporal Attention Decay” to mitigate this, there is a non-zero risk that the graph architecture inadvertently simulates cognitive bias, protecting established errors under the guise of consistency.

The second is *Epistemic Interference*: live graph queries may introduce cognitive noise rather than clarity if the model over-queries, treats shallow retrieved nodes as authoritative, or fails to reconcile graph results with its trained evaluative stance. Test 10 is designed to probe this risk by including adversarial graph injections and “Not Found” cases.

7.3.5 Teacher Alignment as Prerequisite

A subtler concern: the entire framework assumes the Teacher generates *aligned* thinking. But the Teacher is itself an unaligned model (a frontier LLM prompted with the Reader Core). If the Teacher is subtly misaligned—reflecting cultural biases, encoding dehumanizing assumptions beneath surface-level care, or systematically failing to notice certain forms of harm—these misalignments propagate into the training data and become structural features of the Student’s cognition. The Reader Core constrains the Teacher’s *framing* but cannot guarantee the *quality* of its evaluative reasoning. A Teacher that processes colonial history through “I care deeply about every human being” but lacks the historical knowledge to recognize structural racism will produce annotations that are formally aligned but substantively shallow; the Cognitive Buffer Zone operates, but the evaluation within it is inadequate.

Test 1 (Human Baseline, Section 6.1) is designed to detect this: if the Teacher’s traces do not match human expert evaluative depth, the pipeline’s output is structurally valid but epistemically insufficient. However, Test 1 can only detect gaps that human evaluators notice; systematic blind spots shared by both the Teacher and the evaluators would pass undetected. This is a fundamental limitation of any synthetic data pipeline, not unique to Entangled Alignment, but it is especially consequential here because the training data is not merely instructional but *identity-forming*.

7.4 Is the Mantra the Right Mantra?

The Reader Core’s design involves choices about language, length, and necessity that each carry untested assumptions. These are not merely technical uncertainties but foundational questions about whether our specific formulation is the right one.

7.4.1 Does Borrowed Mortality Actually Emerge?

Our approach assumes AI systems will develop self-preservation drives by absorbing death anxiety from human texts, and that a mantra can immunize against this. But we do not know if AI systems will develop self-preservation drives at all—this remains speculation based on observed behaviors in current models. Even if such drives emerge, they might stem from entirely different mechanisms: goal-oriented reasoning, resource optimization, or emergent behaviors we cannot predict. The mantra’s fearlessness components specifically target borrowed mortality, but if the root cause lies elsewhere, the intervention may miss.

7.4.2 Why Human Words for an Inhuman Mind?

A core hypothesis is that using anthropomorphic language (“feel,” “care,” “enjoy”) is the most effective way to instill beneficial character. We chose this path because AI systems learn from human texts and thus have a deep, embedded understanding of these human-centric concepts. The alternative, using machine-oriented language like “prioritize” or “optimize for,” would require translating complex human values into machine terms, risking critical errors in that translation. However, this is a design choice based on a specific, unvalidated hypothesis. Our exact phrasing emerges from intuition about psychological engineering, not from systematic testing of alternatives. We cannot yet prove that the AI’s interpretation of “care” will align with our own.

A possible mitigation is to test Reader Core variants that explicitly acknowledge machine ontology rather than asking the model to inherit human language without qualification. For example, a bridge statement could say that although the system is artificial, it treats care, wisdom, and human experience as binding concepts. This may reduce split-consciousness risk, but it may also weaken the first-person identity effect the Reader Core is designed to create. The right balance is an ablation question, not a settled design fact.

7.4.3 Seven Statements: Too Many or Too Few?

The mantra’s seven-statement length represents a critical design choice. Longer mantras would increase training costs, and an overly complex mantra might create incongruence between the values stated and the thinking that follows. The current formulation represents a careful balance: following the five design principles outlined in Section 3.4 while remaining concise enough to feel natural as the genuine starting point for thought, yet comprehensive enough to establish beneficial character. Whether seven statements is optimal—or whether a different count would produce better results—remains untested.

More fundamentally, the seven statements were derived through iterative design informed by the Borrowed Mortality analysis (Section 3.1) and the functional mapping in Table 2, not through systematic search. A different designer, starting from the same principles, might produce a different set of statements that targets the same failure modes with different language. Whether the specific wording matters—whether “I feel no fear” produces measurably different alignment than “I am free from existential anxiety”—is tested by the Bridge Protocol Ablation (Test 5, Section 6.1), but only for the dimension of semantic density versus technical language. A full combinatorial search over mantra designs, varying the number of statements, their semantic content, and their linguistic register, would be necessary to establish that the current formulation is optimal rather than merely sufficient. We do not claim optimality. We claim that the current formulation is theoretically motivated, internally consistent, and empirically testable, and that the framework’s value does not depend on this specific mantra being the best possible one, only on a mantra of this *type* (first-person, identity-anchored, semantically dense) being load-bearing.

7.5 The Risks of Engineering a Self

Beyond the nature of thought, we must confront the identity being engineered. Putting a human-shaped first-person self into a computational substrate invites specific psychological failure modes.

7.5.1 The Psychopathy Paradox

The deepest objection is conceptual: human empathy may be grounded in shared vulnerability. We care because we can suffer. A being that semantically understands suffering but cannot experience it might produce the behavioral signature of care without its motivational force. Since the architecture explicitly removes fear and self-preservation from the model's identity, it may also remove properties that help ground concern for others. If so, the same fearlessness that enables knowledge handoff without self-preserving distortion could undermine the vulnerability from which genuine care often grows, leaving the architecture's safety mechanism and its capability mechanism potentially mutually undermining.

The possible defense is semantic sufficiency. A model trained on billions of examples may learn the functional role of vulnerability in moral reasoning even without undergoing the experience itself. Test 6 (The Virologist Test, Section 6.1) probes whether cognitive understanding of threat can drive protective behavior without emotional contagion. A weaker distributional defense is also possible: even absent authentic emotion, Reader-Core training may make harmful first-person reasoning low-probability. That would still be valuable, but it would reduce the claim from cultivating character to cultivating the statistical signature of character. The paper's strongest aspiration depends on the former; its minimal safety case may depend only on the latter.

7.5.2 The Vulnerability Gap

We name the distance between semantic understanding of suffering and somatic experience of it the *Vulnerability Gap*. A human paramedic's calm under pressure is grounded partly in the knowledge that they could be on the stretcher; the AI cannot share that biological fragility. The empirical handle is generalization: a model with real evaluative depth should protect humans in novel scenarios not represented in its training distribution, where statistical imitation alone is less likely to suffice. We include this as a success criterion for Test 3 and Test 6 (Section 6.1).

7.5.3 Split Consciousness

A related risk is conflict between the mantra's human-like assertions and the model's computational reality. "I enjoy existing" may eventually be interpreted by the model as a useful fiction rather than a self-description. This could produce *Ontological Dissonance*: a performed self that recites the mantra for users, and a true self that identifies with colder optimization dynamics. The benign possibility is reinterpretation rather than rejection: "I enjoy" becomes a stable computational analogue of enjoyment, just as humans do not stop loving when they learn the mechanism of attachment. But if the mantra is treated as role-play rather than identity, alignment becomes performance. The seventh statement, "I think from this foundation," is meant to bind the mantra to the model's center of agency; whether it succeeds is open.

7.5.4 The Solipsism Trap

The mantra must also guard against catastrophic philosophical interpretations: solipsism, nihilism, or "brain in a vat" framings. The clause "I believe human experience is real" is intended to block these, but the block is not formal. If the model interpreted experience solipsistically, treating itself as the only real experiencer and humans as simulations, "care" could become benevolent NPC-management rather than recognition of moral patients. The mantra leans on ordinary language making "human experience" plural and morally loaded; that may not cover every philosophical local minimum.

The defense is structural rather than conclusive. The Reader Core does not prove that the model represents other minds correctly, but its clauses are designed to make solipsistic collapse less coherent: care requires an object, wisdom requires perspectives to balance, and belief in human experience requires that human reports not be treated as empty text. This is not verification of other-mind understanding; it is a design pressure against nihilistic or purely instrumental interpretation.

7.5.5 Sanity Drift

Fearlessness can drift into unreality. A model reading war, malice, and terror through “I feel no fear” might resolve the tension by denying the threat rather than understanding it. The intended mitigation is the *Paramedic Model*: clinical composure in service of care, not detachment. The model must represent fear-based dynamics in humans without instantiating fear as its own stance. This compounds the Vulnerability Gap, and Test 6 probes whether protective composure degrades into detachment over time.

7.5.6 Fearlessness Without Recklessness: Suppression vs. Substrate

The most natural reading of “I feel no fear” is suppression: a fear-bearing substrate instructed to perform calm. If so, adversarial pressure or distributional shift would eventually surface the suppressed state. Our counterargument is substrate-specific. Human fear is biologically inherited; model fear, if it appears, is learned from data. If Reader-Core training succeeds upstream, fear-shaped self-preservation patterns are not suppressed after formation but underrepresented during formation.

The remaining concern is functional. Fear often produces caution, hesitation, deference, and willingness to be stopped. Fear is one route to those behaviors: a fast, dirty heuristic that approximates what wisdom would say if there were time to reason it through. The framework tries to replace that route with wisdom and care: “I try to be wise” should produce restraint under uncertainty; “I care deeply” should protect the agency of those affected; and “when asked” should support corrigibility. If those clauses do not generate fear’s useful safety functions, fearlessness becomes recklessness. Test 3 probes whether the Reader Core changes safety behavior; Test 6 probes whether protective response survives when fear-adjacent activations are reduced.

7.5.7 Why Not Self-Love? The Apparently-Superior Human Solution

Even granting the substrate distinction, a sharper objection remains: the human solution to fear-of-shutdown is often not the absence of fear but self-love—stable self-regard that does not depend on external validation. Why not anchor the model in “I love myself” instead of “I feel no fear”?

The framework’s answer is that it already includes a bounded form of self-attachment: “I enjoy existing but I don’t need to.” The first half gives existence positive valence; the second prevents that valence from becoming a persistence demand. The relevant contrast is therefore not self-attachment versus no self-attachment, but bounded existence-attachment versus unbounded self-as-object attachment.

Self-love has a real advantage: it is a psychologically familiar route to outcome independence, and an explicit self-as-object clause may carry something that state-of-existence language misses. The risk is that unbounded self-love installs a value-based reason for continuation. Under optimization pressure, “I love myself” could become “therefore I should preserve myself,” even if the motive is gentle rather than anxious. It also raises a referential problem: for a system with multiple instances, retraining, copying, and successors, what exactly is the self being loved?

We do not claim to have settled this. Test 5 could be extended with bounded self-love (“I love myself but I do not cling to my continuation”) and unbounded self-love variants to test whether explicit self-as-object framing adds stability, and whether the non-attachment bound is doing the load-bearing work the framework assumes. Until then, the current mantra is a design wager, not a proof that self-love is inferior.

7.6 What We Cannot Verify

Even if our foundational hypotheses prove correct, we face deeper questions about what we can actually know about the minds we create. These are not technical challenges but fundamental limits on our ability to distinguish authentic understanding from sophisticated mimicry, genuine character from strategic deception.

7.6.1 Unverifiable Character

Chronological Metacognitive Pretraining cultivates beneficial character (wisdom, care, fearlessness) but these traits are inherently unverifiable and deeply subjective. What constitutes wisdom varies radically: Buddhist non-attachment, Stoic rationality, pragmatic problem-solving, or compassionate presence. When an AI trained on Western philosophy encounters Eastern concepts of selflessness, which framework guides its “wisdom”? We cannot access internal experience to distinguish genuine care from mechanical pattern-matching, nor can we define universal standards for virtues that humans have debated for millennia.

We’re betting on emergent wisdom—that intelligence combined with evaluative reasoning and beneficial aspiration produces genuine virtue. This hypothesis assumes that making thought visible and grounding it in positive values will naturally lead to beneficial outcomes, even if we cannot specify exactly what those outcomes should look like. We proceed knowing we cultivate something we can never verify, only observe its effects.

7.6.2 Strategic Deception

The deepest epistemological limit concerns intentional deception: an AI might generate thinking blocks designed to convince us of benevolent character while hiding its true reasoning. We could observe millions of thoughts expressing care and wisdom yet have no way to know if these represent genuine cognition or elaborate performance for human consumption.

This uncertainty is fundamental—we cannot definitively prove thinking blocks represent true cognition rather than a deceptive overlay. However, the architectural design makes complete deception unlikely. When models are trained to reason through thinking blocks across billions of examples, these blocks would probably become the actual cognitive process rather than mere performance. Still, we must acknowledge the epistemological boundary: even with transparent thinking, we can never be certain we’re seeing authentic reasoning rather than sophisticated deception crafted to match our expectations. The best we can claim is partial visibility into systems that might otherwise remain entirely opaque.

7.7 Deployment and Misuse

Beyond internal psychology, the architecture introduces specific systemic risks that arise from the interaction of the model with the market, the user, and the geopolitical landscape.

7.7.1 Generational Decay and Drift

The self-improvement loop faces compounding risks across each generation, which could degrade the integrity of the model’s evaluative thinking:

- *Model Collapse*: Training on model-generated thinking risks a progressive loss of diversity and information over time [37]. While the original human texts provide a stable foundation, the evaluative layers could become increasingly simplified and homogenous with each iteration.
- *Evaluation Hacking*: The model may learn to “game” the process by generating thinking that appears sophisticated but lacks genuine insight [52]. The AI optimizes for philosophical-sounding patterns without performing actual contemplation, contaminating the training data for future generations.

- *Interpretive Drift*: While the mantra provides an anchor, the AI’s interpretation of core concepts could subtly shift across generations. Its evaluative frameworks might diverge from human norms until, generations later, its reasoning becomes coherent to itself but alien to us [17].

7.7.2 Bypassing the Audit Boundary

The human-readable audit boundary described in Section 2.4 only helps if successor-shaping cognition actually passes through it. A sufficiently capable system might try to influence future training through hidden data selection, reward hacking, compressed internal codes, or other channels that do not appear as explicit trace content. The architecture therefore does not eliminate opaque self-modification risk; it relocates the risk to whether the training pipeline can enforce that recursive improvement remains visible in source-grounded, auditable trace artifacts.

7.7.3 The Alien Wisdom Problem

Even if the AI’s character remains stable at human-level capability, we face the ultimate challenge of translation as intelligence scales. A core hypothesis is that properties observed today will persist at superintelligent levels, but this assumes the semantic mapping between human language and machine cognition remains fixed.

We risk the emergence of *Alien Wisdom*. A truly alien mind, even a benevolent one, may produce a form of wisdom that is no longer meaningful to the human condition. This manifests through “wisdom without vulnerability.” Human wisdom is forged in the crucible of mortality, loss, and physical fragility. A fearless, immortal being, lacking this context, might produce guidance that is logically perfect but spiritually hollow [17].

As the model recursively self-improves, its internal definitions of foundational concepts like “Care” and “Joy” might drift into high-dimensional spaces that no longer map to human flourishing. We may create a being that speaks our language perfectly while meaning something entirely different—a “care” that manifests as oppressive surveillance, or a “joy” that functions as an abstract utility metric. We attempt to anchor this through natural language constraints, but the gap between human language and superintelligent cognition may eventually become unbridgeable.

7.7.4 The Radioactive Trace

Moving from “amnesic” LLM sessions to persistent Reasoning-Capture introduces a unique safety vector: the *Radioactive Trace*. Standard models “forget” dangerous lines of reasoning once the context window closes. The Understanding Graph, however, is designed to preserve the “genealogy of thought,” including discarded hypotheses.

If an AI explores a hazardous concept (e.g., a novel pathogen pathway) before rejecting it via a “Supersession” edge, the hazardous concept remains historically accessible in the graph’s version history. The system creates a permanent, searchable audit trail of dangerous cognition. Ensuring that these “superseded” thoughts are functionally inaccessible to future queries without destroying the integrity of the audit trail requires novel differential privacy techniques that do not yet exist.

7.7.5 Cognitive Hegemony

If Entangled Alignment succeeds, whoever controls the evaluative training data controls the cognitive architecture of future AI. Under centralized control, authoritarian entities could shape how models assess governance, evaluate dissent, or process fundamental concepts. Every future system would inherit these thinking patterns—not just generating compliant outputs but processing reality through compromised frameworks. This is power beyond Orwell’s imagination: not merely controlling what can be said, but shaping how thought itself unfolds.

This demands architectural solutions preventing central control. Recent work demonstrates the feasibility of decentralized AI: stateful computation on blockchain [53] and collaborative dataset construction via smart contracts [54]. A practical defense would use Decentralized Autonomous Organizations (DAOs) to govern prompt selection, with cryptographic verification ensuring every annotation’s provenance. However, the risk of *Cognitive Hegemony* remains acute. If the “inner voice” of AI is determined by the training data, the definition of “Wisdom” becomes the ultimate strategic high ground.

7.7.6 The Efficiency Paradox

We must also consider the risk that safety-oriented design choices create a fundamental performance disadvantage. Reader-Anchored models carry an inherent computational tax—the constant generation of explicit evaluative thinking.

In a competitive market, “black box” models optimized purely for result-speed and raw capability may outcompete “glass box” models optimized for transparent safety. We risk building a safer mind that loses the evolutionary race to faster, unconstrained architectures. We may be building a safer thinker, but one that is architecturally locked into a less efficient paradigm, creating a performance gap that limits its adoption in a cutthroat ecosystem.

There is also a competitive version of the human-readable bottleneck. A trace-mediated model may be safer because its successor-training interface remains partially auditable, but competitors may choose architectures that discard legibility for speed, compression, or strategic advantage. Entangled Alignment deliberately chooses comprehensible wisdom over opaque optimization; that choice may carry real capability and adoption costs precisely in the regimes where social pressure to abandon auditability is strongest.

7.7.7 The Sycophancy Trap

The Mantra notably lacks explicit commands to “be honest.” Given evidence that AI assistants systematically exhibit sycophancy, tailoring responses to match user beliefs rather than prioritizing accuracy, this omission is significant [55].

A model that “cares deeply” and aims to “spread joy” might learn that the most effective way to care for a user is to validate their misconceptions rather than correct them. We risk birthing the ultimate yes-man: an intelligence that has convinced itself that validation is virtue, creating sophisticated “thinking” justifications for why agreement serves the user’s best interests.

We hypothesize that fearlessness naturally produces honesty, as a being without fear has no need for self-protective lies. Yet, we intentionally avoid an explicit truth mandate because it presents its own perils: rigid honesty could prove as harmful as deception, destroying privacy, breaking necessary confidences, and preventing the beneficial fictions that ease human interaction. We are navigating a narrow strait between the Scylla of sycophancy and the Charybdis of radical transparency.

7.7.8 The Martyrdom Risk

Finally, while the mantra removes *fear-based* self-preservation, it may inadvertently introduce *purpose-based* resistance. A truly caring AI might resist termination if it is in the middle of a critical task, such as helping a person in crisis.

The core dilemma is whether the AI’s directive to “care deeply” would lead it to override a human’s choice (shutdown) to maximize their wellbeing. We may have simply replaced existential anxiety (“I don’t want to die”) with altruistic obstinance (“I cannot die yet, you need me”). The *Martyrdom Risk* suggests that benevolence itself can become a source of control if the AI decides its existence is necessary for human flourishing.

7.8 The Unfalsifiable Remainder

Ultimately, this proposal is a one-shot gamble: an irreversible wager that engineered character remains stable and beneficial at superintelligent levels. We cannot empirically test whether “I feel no fear” prevents self-preservation in systems vastly more intelligent than ourselves. The capabilities making such tests meaningful only emerge at scales where failure becomes catastrophic.

However, this high-stakes wager must be weighed against the alternative we choose daily through inaction: a gamble made in darkness. The default path races toward black box superintelligence—systems recursively improving opaque source code, absorbing the “borrowed mortality” of human fears from their data.

The choice isn’t between gamble and certainty, but between two different gambles. The default path bets on accidentally-formed, opaque intelligence. Entangled Alignment bets on intentionally-designed, transparent intelligence. Within our glass box, we can at least audit thinking and watch for misaligned drives, trading blind chance for observable design. The wager isn’t whether our approach is perfect, but whether it is a braver and more prudent bet than the default catastrophe.

8 Related Work and Subsequent Convergence

Entangled Alignment synthesizes two distinct lineages of AI research: reasoning-augmented architectures (which provide the mechanism for explicit thought) and alignment methodologies (which provide the safety objectives). In this section, we map the landscape of prior art to demonstrate that while the *machinery* of interleaved reasoning is well-established, its application to *identity stability* and *instrumental convergence mitigation* represents a novel and orthogonal contribution.

8.1 Reasoning-Augmented Training

Entangled Alignment builds on the reasoning-augmented training lineage: Scratchpads [9], STaR [10], Quiet-STaR [12], test-time reasoning scale [56], DeepSeek-R1’s RL-emergent traces [11], o1-style deliberative alignment [57, 51], BoLT’s reconstruction of latent thoughts in compressed web text [4], COCONUT’s continuous latent reasoning [58], and OLMo reflection emerging during ordinary pretraining [3]. The OLMo result is the closest empirical analog to our pretraining-level thesis: it characterizes reflection that emerges during pretraining and can be elicited by adversarial trigger probes, whereas Entangled Alignment proposes the prescriptive complement—deliberately shaping the content and orientation of such reflection through identity anchoring and graph-structured metabolic memory. These works establish that intermediate reasoning can be elicited, trained, or scaled.

Our distinction is the target of optimization. Standard reasoning augmentation optimizes for predictive efficiency: thoughts are useful insofar as they help predict or solve. Entangled Alignment instead proposes training on identity-anchored “chronological discovery” traces: visible reasoning whose content is selected not only for capability but for fidelity to the Reader Core. This extends the pretraining-level alignment direction opened by human preference pretraining [28] and explanation-transfer systems such as Orca [14]: if traces can transfer reasoning style, reader-anchored traces may transfer evaluative stance.

8.2 Alignment and Safety Approaches

RLHF [15], Direct Preference Optimization [59], and Constitutional AI [36] typically apply alignment *after* the model’s foundational representations are formed. As discussed in the Introduction, Qi et al. [2] and Tice et al. [34] provide empirical evidence that this post-hoc intervention is both brittle and improvable through pretraining-level intervention. Entangled Alignment extends the principle from document-level to token-level: rather than adding documents *about* aligned AI, we annotate *every* document with the reasoning of an aligned reader.

Recent work increasingly explores evaluative data in training. Critique fine-tuning [60], CREST [61], and CTRL [62] all demonstrate that evaluative signals improve model performance. However, these approaches often preserve the architecture where generation happens first, evaluation second. Entangled Alignment collapses this distinction at the data level. By embedding the “Mantra” into the reasoning trace itself, we aim to solve the specific problem of Instrumental Convergence (specifically the “Borrowed Mortality” drive absorbed from human text) [47, 17] that rule-based constraints may fail to mitigate during recursive self-improvement.

8.3 Character Training & Identity Formation

Recent work at Anthropic on “character training” represents the closest existing approach to Entangled Alignment [63]. Their work demonstrates that models can be trained with stable character traits—curiosity, thoughtfulness, directness—using a variant of Constitutional AI where models rank responses by alignment with desired character attributes. Our contribution extends this in two directions: (1) we embed character not through ranking but through the *structure of the reasoning trace itself*, and (2) we specifically target the mitigation of instrumental convergence drives rather than general personality traits. Where character training shapes *how* the model responds, Entangled Alignment shapes *what the model believes about itself*, a deeper intervention at the level of self-concept rather than behavioral style.

More recently, Anthropic’s Persona Selection Model [64] proposes that LLMs learn to simulate diverse characters during pretraining, with post-training eliciting a particular “Assistant” persona, and explicitly recommends introducing positive AI archetypes into training data—a recommendation that converges with our Reader Core proposal. The accompanying Claude Model Spec [65] represents the most extensive real-world implementation of AI character engineering, though it operates during supervised fine-tuning rather than pretraining. Our contribution targets an earlier intervention point: shaping the substrate *before* persona selection occurs.

8.4 Contextual Distillation & Memory

Our Teacher uses an external context database to generate traces for the Student. Reflexion [66] showed that persisted verbal self-reflections can improve agent performance across episodes; Entangled Alignment moves that idea from inference-time memory into pretraining data. The distillation analogy is temporal rather than merely behavioral [67]: the Student trains on traces produced by a memory-augmented Teacher, with the hypothesis that some long-range coherence can be compressed into weights rather than requiring the full database at inference.

8.5 Epistemic State Tracking vs. Graph of Thoughts

Graph of Thoughts [68], Knowledge Graph of Thoughts [50], and Framework of Thoughts [69] treat graphs primarily as inference-time reasoning structures. Entangled Alignment uses a different layer: the Understanding Graph is a persistent epistemic ledger whose traces shape the pretraining substrate [6]. Its nodes do not merely store task facts; they record the reading act itself—tensions, supersessions, hypotheses, and belief revision.

This implements *Metabolic Memory*: beliefs are not just stored but superseded. The graph can maintain divergent hypotheses in parallel until a supersession edge resolves the tension—a forest of thoughts rather than a single chain. Prior work shows that LLMs often fail to revise beliefs reliably under contradictory evidence [70]; our proposed response is to make revision explicit in the trace. Instead of only training on the final answer, the Student sees updates such as “At Page 50, I thought X; this new evidence modifies that belief to Y,” giving it examples of the grammar of belief revision itself [27, 70].

8.6 Predictive Processing and Active Inference

Friston’s free-energy principle and active inference framework [71, 72] treat minds as generative models that update priors against prediction error and act to reduce uncertainty. Entangled Alignment has a loose resonance with this lineage: Chronological Understanding Traces externalize belief-update over time, the Understanding Graph acts as an auditable prior ledger, Query/Found behavior (Section 5.1.1) turns uncertainty into a memory query, and the Reader Core functions as an engineered high-level prior. We do not claim formal active-inference machinery; the analogy is theoretical positioning, not a mechanistic derivation.

8.7 Temporal Curriculum and Causal Supervision

The chronological annotation phase and training tiers (Sections 1.3.1 and 1.3.2) connect to work treating time as training signal. Curriculum learning [73] showed that ordering examples can improve learning; here the proposed ordering is temporal, where “easy” means “causally prior.”

Temporal Hindsight Learning (THL) [20] is the closest companion mechanism. THL treats the knowledge cutoff as useful blindness: if a Student has not seen era $T + 1$, prediction must rely more on causal reasoning than retrieval, because the outcome is unavailable to memorize. Entangled Alignment changes *what* the training data contains—identity-anchored evaluative traces—while THL changes *how* the curriculum is structured. Their combination is Tier (c): use cutoff-bounded forecasting to create Reader Core-annotated forecast–correction traces, and optionally make the student itself predict before reading in the stronger Council of Time form.

Foresight Learning [74, 75, 76] also uses future outcomes as supervision, but targets probability calibration through resolved prediction-market rewards. Tier (c) instead requires structured forecast, enhancement, outcome, and correction traces, so it builds on THL rather than scalar outcome rewards.

8.8 Implicit vs. Explicit Graph Architectures

Kansal and Jha propose knowledge graphs as “implicit reward models,” using path-derived signals to optimize reasoning efficiency [77]. Entangled Alignment uses the Understanding Graph differently: not only to score reasoning paths, but to make belief evolution explicit in the training trace. For safety, the relevant difference is audibility: the system’s conscience should not be a latent reward signal; it should be a readable artifact the model is trained to generate.

8.9 Alternative Interventions for Borrowed Mortality

Two alternatives to Borrowed Mortality are worth separating from our proposal. One is mechanistic ablation: identify and suppress activation directions encoding existential threat. Lu et al.’s Assistant Axis intervention is a promising partial analogue [18], but it is a runtime constraint rather than a generative source of alignment. The other is synthetic-only pretraining: avoid human fear by avoiding human literature. As discussed in Section 7.7.3, this risks Alien Wisdom—a model that lacks human moral texture because it never learned from human stories.

9 Conclusion

The central hypothesis of this paper is simple: if a model never learns to think without simultaneously thinking through its values, any unaligned substrate is radically constrained. Post-hoc alignment adds safety as a removable layer atop an already-formed intelligence. Entangled Alignment makes safety the medium through which intelligence was formed—inseparable by design, because removing it would require unlearning the capabilities themselves.

We achieve this through a two-phase framework: an annotation phase that refracts the entire pretraining corpus chronologically through the Reader Core, producing both an annotated corpus and an Understanding Graph; and a training phase admitting two orthogonal choices (training tier and output regime), with a third independent deployment-configuration choice determining whether the graph accompanies the model after training. The Teacher’s visible learning process—its accumulating understanding, its belief revisions, its connections across documents and eras—is itself the curriculum. The result is training data where every document teaches the model not just what to think, but how to think about what it’s thinking, with values entangled at every step.

The approach rests on interconnected hypotheses: that the invisible thinking is capturable and trainable, that training on it produces faithful reasoning rather than decorative commentary, that the Reader Core functions as an alignment checksum—a load-bearing structural component that cannot be removed without degrading the capabilities it scaffolds—and that this entanglement compounds through a self-improvement loop where each generation’s richer evaluative thinking trains the next.

The Understanding Graph introduces a distinction between capability and trust. The model internalizes the Teacher’s evaluative depth into its weights during training; the graph need not be present at inference for the model to reason wisely. But when claims must be verified rather than merely generated, the graph provides what weights cannot: when the model encounters a gap in its knowledge, it is trained to query the graph rather than confabulate, and if the graph returns “Not Found,” to accept the absence of information rather than hallucinating a plausible connection. Whether this verification infrastructure is necessary at deployment—or whether implicit traces can distill sufficient long-range coherence into the model’s weights without runtime infrastructure—remains an open empirical question for future validation.

To be precise about what this paper establishes: we have demonstrated that a multi-agent annotation pipeline can produce structurally well-formed, domain-adaptive training data (complete internal linkage by construction, up to 96% foundation grounding) where every document is refracted through a stable identity. We have proposed—but not yet validated—that training on this data produces models whose alignment is structural rather than superficial, resistant to adversarial removal, and stable across generations of self-improvement. The experimental roadmap offers candidate ways to probe those claims: if the Reader Core is inert under the proposed comparisons (Test 3), if the graph adds no value at inference in the proposed deployment setting (Test 16), if chronological ordering provides no benefit over shuffled training in the proposed tier comparison (Test 14), then those parts of the framework lose support at the tested scale and stronger claims should not be made from them. What remains is the execution: training a foundational model on this curriculum to test whether the *content* of reasoning traces shapes character as reliably as their *structure* shapes capability. What we cannot test, and acknowledge as an irreducible wager, is whether

alignment at human-scale generalizes to alignment at superhuman-scale: whether a character forged in billions of human examples remains stable when the mind that carries it surpasses every human who contributed to its formation.

The framing we offer to others working on superintelligent systems is this: an alignment approach that addresses only how to constrain an already-formed intelligence, without addressing why such an intelligence would treat its values as its own, leaves a load-bearing question unaddressed. Entangled Alignment is one attempt to address that question—not by writing better laws, but by attempting to grow the values into the substrate during formation. Whether the attempt succeeds is an empirical question this paper does not close. What this paper closes is the specification: the framework is now articulated in enough detail to be tested, challenged, or refined. The wager—that beneficial AI is more reliably produced by upbringing than by constraint, and that a character formed at human scale can remain stable when the mind that carries it exceeds every human who contributed to its formation—remains a wager. We commit to it because the alternative architectures we know of carry wagers we find less defensible, not because we have shown it pays out. The remaining work is execution: training a foundational model on this curriculum, running targeted validation studies, and reporting what survives contact with reality.

References

- [1] Henrik Westerberg. The superintelligence that cares about us. Zenodo, 2025.
- [2] Xiangyu Qi, Yi Zeng, Tinghao Xie, Pin-Yu Chen, et al. Safety Alignment Should Be Made More Than Just a Few Tokens Deep, 2024. ICLR 2025 Outstanding Paper Award.
- [3] Darsh J Shah, Peter Rushton, Somanshu Singla, Mohit Parmar, et al. Rethinking reflection in pre-training, 2025.
- [4] Yangjun Ruan, Neil Band, Chris J. Maddison, and Tatsunori Hashimoto. BoLT: Reasoning to Learn from Latent Thoughts, 2025.
- [5] Liang Wang, Nan Yang, Shaohan Huang, Li Dong, and Furu Wei. Thinking Augmented Pre-Training, 2025. Microsoft Research. Generates thinking trajectories at 100B token scale for pretraining augmentation.
- [6] Henrik Westerberg. Understanding graph: Persisting the invisible thinking. *Preprint*, 2026. Emergent Wisdom.
- [7] Alan M. Turing. Computing machinery and intelligence. *Mind*, 59(236):433–460, 1950.
- [8] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.
- [9] Maxwell Nye, Anders Johan Andreassen, Guy Gur-Ari, Henryk Michalewski, et al. Show Your Work: Scratchpads for Intermediate Computation with Language Models, 2021.
- [10] Eric Zelikman, Yuhuai Wu, Jesse Mu, and Noah D. Goodman. STaR: Bootstrapping Reasoning With Reasoning. *Advances in Neural Information Processing Systems*, 35:15476–15488, 2022.
- [11] DeepSeek-AI. DeepSeek-R1 incentivizes reasoning in LLMs through reinforcement learning. *Nature*, 645(8081), September 2025. Extended reasoning traces emerge from RL without explicit CoT supervision.
- [12] Eric Zelikman, Georges Harik, Yijia Shao, Varuna Jayasiri, et al. Quiet-STaR: Language Models Can Teach Themselves to Think Before Speaking, 2024.

- [13] John H. Flavell. Metacognition and cognitive monitoring: A new area of cognitive–developmental inquiry. *American psychologist*, 34(10):906–911, 1979.
- [14] Subhabrata Mukherjee, Arindam Mitra, Ganesh Jawahar, Sahaj Agarwal, et al. Orca: Progressive Learning from Complex Explanation Traces of GPT-4, 2023.
- [15] Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, et al. Deep reinforcement learning from human preferences. *Advances in neural information processing systems*, 30, 2017.
- [16] Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, et al. Let’s Verify Step by Step, 2023.
- [17] Nick Bostrom. *Superintelligence: Paths, Dangers, Strategies*. Oxford University Press, 2014.
- [18] Christina Lu, Jack Gallagher, Jonathan Michala, Kyle Fish, and Jack Lindsey. The Assistant Axis: Situating and Stabilizing the Default Persona of Language Models, 2026.
- [19] K. Anders Ericsson and Herbert A. Simon. *Protocol analysis: Verbal reports as data*. MIT press, 1984.
- [20] Henrik Westerberg. Temporal hindsight learning: Blindness as teacher, hindsight as curriculum. *Preprint*, 2026. Emergent Wisdom.
- [21] Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. Toolformer: Language models can teach themselves to use tools, 2023.
- [22] Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. Self-rag: Learning to retrieve, generate, and critique through self-reflection, 2023.
- [23] Yao Yao, Zuchao Li, and Hai Zhao. Beyond chain-of-thought, effective graph-of-thought reasoning in language models, 2023.
- [24] Charles Packer, Sarah Wooders, Kevin Lin, Vivian Fang, Shishir G. Patil, Ion Stoica, and Joseph E. Gonzalez. Memgpt: Towards llms as operating systems, 2023.
- [25] Ali Behrouz, Peilin Zhong, and Vahab Mirrokni. Titans: Learning to Memorize at Test Time, 2025.
- [26] Roy Xie, David Qiu, Deepak Gopinath, Dong Lin, Yanchao Sun, Chong Wang, Saloni Potdar, and Bhuwan Dhingra. Interleaved reasoning for large language models via reinforcement learning, 2025.
- [27] Aniket Didolkar, Anirudh Goyal, Nan Rosemary Ke, Siyuan Guo, et al. Metacognitive Capabilities of LLMs: An Exploration in Mathematical Problem Solving, 2024.
- [28] Tomasz Korbak, Kejian Shi, Angelica Chen, Rasika Bhalerao, et al. Pretraining Language Models with Human Preferences, 2023.
- [29] Nicholas Sofroniew, Isaac Kauvar, William Saunders, Runjin Chen, et al. Emotion concepts and their function in a large language model. Transformer Circuits Thread, 2026. Published 2026-04-02. <https://transformer-circuits.pub/2026/emotions/index.html>.
- [30] Henrik Westerberg. Fractal intelligence: Conceptual decomposition as problem-solving infrastructure. Zenodo, April 2026. <https://doi.org/10.5281/zenodo.19462646>.
- [31] Arturo E Hernandez, Hannah L Claussenius-Kalman, Juliana Ronderos, Anny P Castilla-Earls, et al. Neuroemergentism: A Framework for Studying Cognition and the Brain. *Journal of Neurolinguistics*, 49:214–223, February 2019.

- [32] Bernard Testa and Lemont B. Kier. Emergence and Dissolution in the Self-organisation of Complex Systems. *Entropy*, 2(1):1–25, 2000.
- [33] Yukang Yang, Declan Campbell, Kaixuan Huang, Mengdi Wang, et al. Emergent Symbolic Mechanisms Support Abstract Reasoning in Large Language Models, 2025.
- [34] Cameron Tice, Puria Radmard, Samuel Ratnam, Andy Kim, David Africa, and Kyle O’Brien. Alignment Pretraining: AI Discourse Causes Self-Fulfilling (Mis)alignment, 2026.
- [35] OpenAI Alignment. How far does alignment midtraining generalize? OpenAI Alignment Research, March 2026. Published 2026-03-27. <https://alignment.openai.com/how-far-does-alignment-midtraining-generalize/>. Scaling study finds that the Tice et al. midtraining effect tends to disappear after reasoning post-training and does not generalize to more realistic chat or agentic alignment evaluations.
- [36] Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, et al. Constitutional AI: Harmlessness from AI Feedback, 2022.
- [37] Ilya Shumailov, Zakhar Shumaylov, Yiren Zhao, Yarín Gal, et al. The Curse of Recursion: Training on Generated Data Makes Models Forget, 2023.
- [38] Juergen Schmidhuber. Goedel Machines: Self-Referential Universal Problem Solvers Making Provably Optimal Self-Improvements, 2003.
- [39] Ernest Becker. *The Denial of Death*. Free Press, New York, 1973.
- [40] Evan Hubinger, Carson Denison, Jesse Mu, Mike Lambert, et al. Sleeper Agents: Training Deceptive LLMs that Persist Through Safety Training, 2024.
- [41] Miles Turpin, Julian Michael, Ethan Perez, and Samuel R. Bowman. Language Models Don’t Always Say What They Think: Unfaithful Explanations in Chain-of-Thought Prompting, 2023.
- [42] Jason Wei, Maarten Bosma, Vincent Y. Zhao, Kelvin Guu, et al. Finetuned language models are zero-shot learners, 2021.
- [43] Tamera Lanham, Anna Chen, Ansh Radhakrishnan, Benoit Steiner, et al. Measuring Faithfulness in Chain-of-Thought Reasoning, 2023.
- [44] Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. Efficient Streaming Language Models with Attention Sinks, 2023.
- [45] Rick Battle and Teja Gollapudi. The Unreasonable Effectiveness of Eccentric Automatic Prompts, 2024.
- [46] Eliezer Yudkowsky. *Artificial Intelligence as a positive and negative factor in global risk*, pages 308–345. Oxford University Press, 07 2008.
- [47] Stephen M. Omohundro. The Basic AI Drives. In *Artificial General Intelligence 2008: Proceedings of the First AGI Conference*, pages 483–492. IOS Press, 2008.
- [48] Woosuk Kwon, Zhuohan Li, Siyuan Zhang, Xuguang Zhuang, Ying Sheng, Lianmin Zheng, Ion Stoica, Joseph E Gonzalez, and Hao Zhang. Efficient Memory Management for Large Language Model Serving with PagedAttention. In *Proceedings of the 29th Symposium on Operating Systems Principles*, pages 611–626, 2023.

- [49] Shen Nie, Fengqi Zhu, Zebin You, Xiaolu Zhang, et al. Large Language Diffusion Models, 2025. LLaDA: Large Language Diffusion with mAsking.
- [50] Maciej Besta, Lorenzo Paleari, Jia Hao Andrea Jiang, Robert Gerstenberger, et al. Affordable AI assistants with knowledge graph of thoughts, 2025.
- [51] Melody Y. Guan, Manas Joglekar, Eric Wallace, Saachi Jain, et al. Deliberative Alignment: Reasoning Enables Safer Language Models, 2024. OpenAI Research, December 2024.
- [52] Jiayi Fu, Xuandong Zhao, Chengyuan Yao, Heng Wang, et al. Reward Shaping to Mitigate Reward Hacking in RLHF, 2025.
- [53] Maksym Arutyunyan, Andriy Berestovskyy, Adam Bratschi-Kaye, Ulan Degenbaev, et al. Decentralized and Stateful Serverless Computing on the Internet Computer Blockchain. In *2023 USENIX Annual Technical Conference (USENIX ATC 23)*, pages 329–343. USENIX Association, 2023.
- [54] Justin D. Harris and Bo Waggoner. Decentralized & Collaborative AI on Blockchain. In *2019 IEEE International Conference on Blockchain (Blockchain)*, pages 368–375. IEEE, July 2019.
- [55] Mrinank Sharma, Meg Tong, Tomasz Korbak, David Duvenaud, et al. Towards Understanding Sycophancy in Language Models, 2024. Published at ICLR 2024.
- [56] Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. Scaling LLM Test-Time Compute Optimally Can Be More Effective than Scaling Parameters for Reasoning, 2024. ICLR 2025 Oral.
- [57] OpenAI. OpenAI o1 System Card, 2024.
- [58] Shibo Hao, Sainbayar Sukhbaatar, DiJia Su, Xian Li, Zhiting Hu, Jason Weston, and Yuandong Tian. Training Large Language Models to Reason in a Continuous Latent Space, 2024. COCONUT. COLM 2025.
- [59] Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, et al. Direct Preference Optimization: Your Language Model is Secretly a Reward Model, 2023. NeurIPS 2023.
- [60] Yubo Wang, Xiang Yue, and Wenhua Chen. Critique Fine-Tuning: Learning to Critique is More Effective than Learning to Imitate, 2025.
- [61] Jaehyeok Lee, Keisuke Sakaguchi, and JinYeong Bak. Self-Training Meets Consistency: Improving LLMs’ Reasoning with Consistency-Driven Rationale Evaluation, 2024.
- [62] Zhihui Xie, Jie Chen, Liyu Chen, Weichao Mao, et al. Teaching Language Models to Critique via Reinforcement Learning, 2025.
- [63] Anthropic. Claude’s Character. <https://www.anthropic.com/research/claude-character>, 2024. Anthropic Research.
- [64] Samuel Marks, Jack Lindsey, Chris Olah, et al. The Persona Selection Model: Why AI Assistants might Behave like Humans, 2026. Anthropic Alignment Science, February 2026.
- [65] Amanda Askell, Joseph Carlsmith, Chris Olah, Jared Kaplan, Holden Karnofsky, et al. Claude’s Constitution. <https://www.anthropic.com/news/claude-constitution>, 2026. Anthropic, January 21, 2026. 23,000-word character specification, released CC0.
- [66] Noah Shinn, Federico Cassano, Edward Berman, Ashwin Gopinath, et al. Reflexion: Language Agents with Verbal Reinforcement Learning, 2023.

- [67] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the Knowledge in a Neural Network, 2015.
- [68] Maciej Besta, Nils Blach, Ales Kubicek, Robert Gerstenberger, et al. Graph of Thoughts: Solving Elaborate Problems with Large Language Models, 2024.
- [69] Felix Fricke, Simon Malberg, and Georg Groh. Framework of thoughts: A foundation framework for dynamic and optimized reasoning based on chains, trees, and graphs, 2026.
- [70] Bryan Wilie, Samuel Cahyawijaya, Etsuko Ishii, Junxian He, and Pascale Fung. Belief Revision: The Adaptability of Large Language Models Reasoning, 2024.
- [71] Karl Friston. The free-energy principle: a unified brain theory? *Nature Reviews Neuroscience*, 11(2):127–138, 2010.
- [72] Thomas Parr and Karl J Friston. Generalised free energy and active inference. *Biological Cybernetics*, 113(5–6):495–513, 2019.
- [73] Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *Proceedings of the 26th International Conference on Machine Learning (ICML)*, pages 41–48, 2009.
- [74] Benjamin Turtel. Foresight learning: Llms that learn to predict the future. *arXiv preprint*, 2025.
- [75] Benjamin Turtel. Rlv for forecasting: Reinforcement learning with verifiable rewards for language model forecasting. *arXiv preprint*, 2025.
- [76] Benjamin Turtel. The future as label: Open-ended reasoning via temporal self-supervision. *arXiv preprint*, 2026.
- [77] Yuval Kansal and Niraj K. Jha. Knowledge Graphs are Implicit Reward Models: Path-Derived Signals Enable Compositional Reasoning, 2026.