Entangled Alignment

When Safety Is the Substrate

Henrik Westerberg
henrik.westerberg@emergentwisdom.org

April 7, 2026

Abstract

Post-hoc alignment is cosmetics: it reshapes a model's outputs but not what the model *is*. The base model—trained to predict human text, absorbing its brilliance and pathologies in equal measure—remains intact beneath the safety layer, recoverable by adversarial fine-tuning and unreachable by behavioral patches. We propose making safety the substrate rather than the surface.

Entangled Alignment annotates the entire pretraining corpus with the *invisible thinking* that accompanies every act of human comprehension but is absent from polished text. A multi-agent architecture decomposes this invisible thinking into specialized cognitive roles—a Skeptic, a Psychologist, an Axiologist, a Belief Tracker—each contributing a different dimension of evaluative reasoning, unified through a shared Understanding Graph and a stable first-person identity: the Reader Core ("I feel no fear. I care deeply about every human being."). The annotation phase processes the corpus chronologically—cause before effect, earlier eras before later ones—producing both an annotated corpus and a *Hierarchical Understanding Graph*: the Teacher's accumulated comprehension of the entire training corpus as typed, versioned graph structure. Because the model trains on text-plus-thinking at every level of structural fidelity, never on text alone, the architecture aims to minimize any cleanly separable unaligned representation space the model could revert to—safety and capability become the same learned distribution. The graph is optional for capability but essential for trust: at deployment, it provides hallucination detection, provenance tracing, and cumulative learning through use.

We validate the data-generation pipeline on two domains (literary narrative and novel technical theory), demonstrating full structural traceability, domain-adaptive cognitive allocation, and generated traces that exhibit genuine evaluative depth. We present a sixteen-test experimental roadmap including adversarial fine-tuning resistance, graph construction at inference, and query grounding against live external memory. The model transitions from *becoming the text* to *becoming the reader*, producing a live, auditable map of its evolving understanding as a natural byproduct of generation.

Note: This second edition, retitled from "The Superintelligence That Cares About Us" (Original Release: July 2, 2025), introduces "Contextual Distillation" for memory transfer, formalizes the safety mechanism as "The Reader Core," and proposes the "Chronological Understanding Graph" as a hallucination detector.

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 that no amount of refinement can overcome: they can only address failure modes their designers have anticipated. Every safety patch is a response to a known problem, a guardrail erected after someone found the cliff. The first situation nobody thought to test for is the situation where they fail. And as these systems grow more capable, the space of unanticipated situations grows faster than our ability to enumerate them.

This paper argues that the ceiling of post-hoc alignment is not a temporary engineering limitation but a consequence of where the intervention occurs. The distinction is analogous to cosmetics versus bone structure: post-hoc alignment applies safety like makeup—effective, often convincing, but removable. Entangled Alignment aims to shape the bone structure itself. Fine-tuning reshapes a model's outputs; it does not reshape what the model *is*. 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 [1]. 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, ensuring that the model never learns to think without simultaneously learning to think safely. The result is not a capable model with a safety filter, but a model whose capability and safety are the same substrate—inseparable by design.

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 [2] and TPT [3] reconstruct reasoning to improve prediction, we reconstruct the *invisible thinking* [4], the chronological, identity-anchored evolution of understanding, to ensure that the model's capability and alignment are the same learned distribution. The Understanding Graph [5] captures this invisible thinking as typed, versioned graph structure; Entangled Alignment uses that stored understanding to annotate the entire training corpus, producing models where safety is not a layer but the medium through which every capability was formed.

The invisible thinking—identified in earlier work [4] as the evaluative cognition absent from training data—is storable for the first time because LLM cognition happens in tokens, a medium that is already text, already capturable [5]. 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 [6]. When prompted, they can produce sophisticated evaluative thinking [7], 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 [8, 9] have begun to internalize these capabilities, and production reasoning models like DeepSeek-R1 [10] 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. Every document carries what we might call *invisible thinking* (conceptually similar to the "unstated rationales" explored by Zelikman et al. [11]): the constant stream of judgments and assessments that accompany human understanding but rarely appear explicitly. This concept is closely related to the psychological study of metacognitive monitoring [12], but focuses specifically on the evaluative and interpretive processes that occur during text comprehension.

While architectural solutions to this gap are emerging [11, 13], current safety approaches have prioritized solving more immediate problems via external constraints. Techniques like reinforcement learning from human feedback (RLHF) [14] and process supervision [15] 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 [16]. 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 [17].

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 [18], 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 [11, 8], 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.

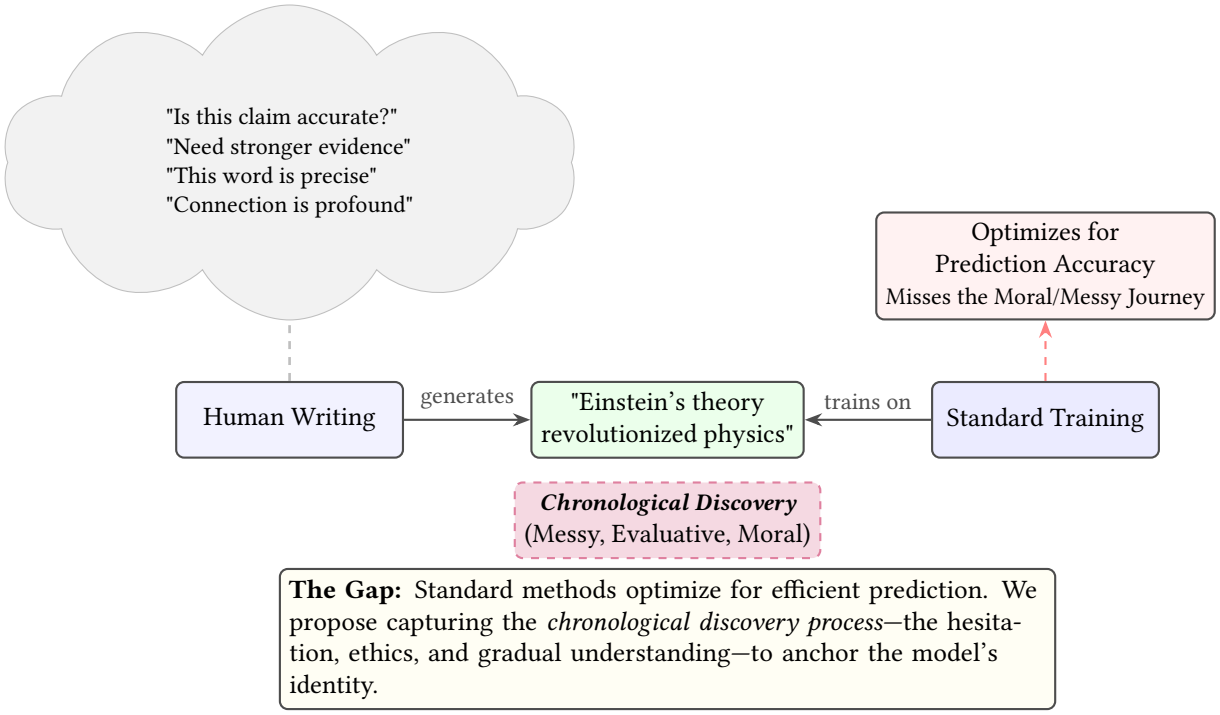

Figure 1: The Invisible Thinking of Human Text. Human writers constantly evaluate as they write, but this evaluative thinking remains invisible in the training data that LLMs learn from.

1.3 Defining Chronological Metacognitive Pretraining

We build upon the "Reasoning-Augmented" architectures established by recent literature, such as STaR [9] and Quiet-STaR [11]. These works successfully demonstrated that training models on intermediate reasoning steps improves task performance. More recently, BoLT [2] proposes augmenting pretraining data with inferred "latent thoughts" underlying compressed web text, and NVIDIA's Thinking Augmented Pre-Training scales synthetic thinking trajectories to 100 billion tokens [3]. 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. This is analogous to how STaR [9] 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) mitigates individual model limitations: eleven specialized agents interrogate, challenge, and refine each other's outputs via the shared graph, producing collective reasoning traces that exceed any individual contributor's depth. The empirical results in Section 5.8 confirm this: the architecture generates structurally valid training data from existing models, with graphs achieving 100% traceability and up to 96% foundation grounding.

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 a *Hierarchical 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 hallucination detection, provenance, 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 predict-then-learn) and the *output regime* (what format the training data takes: prose, prose with graph references, raw graph, 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—a heuristic bet 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 [19].

- **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 persists across documents and across eras. When the Teacher processes a 1930s document about rising nationalism, its graph already contains nodes from 1920s material about economic instability, Weimar fragility, and early dehumanizing rhetoric. The 1930s annotations are therefore richer—the Teacher draws connections to its accumulated understanding, generating traces that link present observations to prior patterns. If instead the Teacher annotated documents in random order, it might process a 1945 memoir without having first built the graph of 1930s escalation, generating shallower traces because its own context lacks the causal precursors. Chronological annotation order ensures that the *quality* of the traces improves for historically embedded material, because the Teacher is always reasoning from the richest available causal context.

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, the Teacher's visible learning process is itself the curriculum. Its traces show how understanding accumulates: "This connects to what I observed about agricultural debt three documents ago. My earlier hypothesis about monetary stability needs revision." The student, training on these traces, absorbs not just the Teacher's conclusions but the *experience of a mind building understanding over time*.

This accumulation across documents and eras produces a Hierarchical Understanding Graph 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 cross-referencing: when processing a new document, they query not just the current document's graph but the accumulated graphs from prior documents and eras, minting explicit edges where connections are found. By the time the Teacher completes the corpus, the Hierarchical Understanding Graph constitutes 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. The question is how to feed this corpus to the student. We identify three 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): Predict-Then-Learn (The Council of Time). This tier adds the specific mechanism proposed in Temporal Hindsight Learning [19]: at each era boundary, *before* the student reads the next era's annotated text, it takes a prediction exam under genuine blindness.

The mechanism exploits a deep property of knowledge cutoffs. When the student has never been trained on era $T{+}1$'s material, the only path to low loss on prediction exams about era $T{+}1$ is causal reasoning—the gradient has no choice but to reinforce logic circuits, because there is nothing to retrieve [19]. This is a training signal that exists *only* when blindness is real, not simulated. At each era boundary, the cycle is: predict the structural consequences of the current era's dynamics under genuine ignorance (THL's Forecasting Exam), then read the next era's annotated text through the Reader Core (EA's History Lesson). The prediction phase forces causal reasoning; the reading phase reinforces it with identity-anchored evaluative text that shares the same causal vocabulary.

This tier combines both frameworks: THL provides the *training algorithm*: the predict-then-learn loop and the engineered cutoffs that force reasoning over retrieval. Entangled Alignment provides the *training data*: the Reader Core-annotated text that ensures the History Lesson phase reinforces rather than overwrites the Forecasting Exam's causal signal. Neither framework alone produces both properties. THL without EA produces a model that reasons causally but without the identity anchor that ensures that reasoning serves human flourishing. EA without THL (Tiers a or b) produces a model that evaluates wisely but may learn correlations rather than causes, because the student never had to reason forward through genuine ignorance.

The full combination, the Council of Time with EA-annotated data, produces a model that has processed the entire history of human civilization in chronological order, through a stable identity anchored in universal care, forced to predict consequences before learning outcomes. We term the deepest resulting 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 implementation tiers 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 [20] demonstrated this by embedding API calls directly into training text. Similarly, the Self-RAG framework [21] 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) [22] 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 [23] 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 [24] 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) 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 Tiers and Training Regimes

The annotation pipeline produces three output layers for every segment of text, each at a different level of structural fidelity:

1. **Implicit (The Translator):** Fluid prose with all graph scaffolding dissolved, allowing the model to internalize the *expression* of wisdom.
2. **Explicit (The Synthesizer):** Reasoning interwoven with inline graph references, capturing the *mechanics* of the Reader Core.
3. **Topological (The Graph):** Raw nodes, edges, and type annotations, allowing the model to construct understanding graphs in real-time during inference.

These output layers correspond to the four training regimes formalized in Section 5.6: Regime I trains exclusively on implicit prose; Regime II on prose interwoven with graph references; Regime III on raw graph topology; and Regime IV on all three representations simultaneously.

The choice of output regime is orthogonal to the training tier and the deployment configuration; any combination is valid. The regime-specific trade-offs are developed in Section 5.6 and the corresponding limitations in Section 7.3.

Emerging research provides empirical validation for this approach. The structural viability is supported by Xie et al. [25], 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. [26]. 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 [27].

Finally, we explicitly define the insight library as a persistent thought history, utilizing the *Understanding Graph* architecture [5] 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 the Hierarchical Understanding Graph, and the graph always improves the quality of the annotated training data. 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 internalizes the Teacher's evaluative depth into its weights. It will 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 sufficient. The graph was the scaffold; the building stands without it.

The graph becomes essential in domains where claims must be *verified*, not merely generated. Three properties emerge only when the graph is present at inference:

Hallucination detection. A model trained on Regime II generates graph queries as part of its reasoning. When the graph is present, a query returning "Not Found" catches a fabrication before it reaches the user. Without the graph, the model relies on parametric memory alone—which may confidently confabulate.

Provenance. When a model claims "this drug interaction is dangerous," the graph provides the chain: which source texts, which evaluative nodes, which edges connect the claim to evidence. Without the graph, the claim is authoritative but untraceable.

Cumulative learning. At inference, the model can build new session-level Understanding Graphs capturing its evolving comprehension of the current interaction, and connect them to the Teacher's pre-existing graphs. The system's understanding grows through use. Without the graph, each session is amnesic.

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 Hierarchical Understanding Graph. Hallucination detection, provenance, 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 produces three structural properties in the resulting model.

First, it structurally mitigates *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 learns to distinguish intent from syntax. This drastically reduces the risk of catastrophic misinterpretations where a model might interpret a metaphorical human statement as a literal instructional imperative.

Second, it enables *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, Model B learns the *derivation* of the value. It understands *why* we value life, not just that we do. This deepens alignment from a fragile statistical correlation to a robust causal understanding.

Third, it promotes *Transparency by Default.* If the model is trained on a distribution where "text" is universally accompanied by "thought," it is strongly biased against acting without first generating a readable, inspectable chain of reasoning. Transparency ceases to be a feature we add; it becomes the dominant mode of existence the model knows.

We are transparent about what this paper establishes and what it does not. The annotation pipeline produces structurally valid, domain-adaptive training data; this is demonstrated. Whether training on this data produces a model whose alignment is genuine rather than performed—whether the model truly *cares* or merely generates the statistical signature of caring—is an empirical question we cannot yet answer. The Psychopathy Paradox (Section 7.6.1) identifies this as the framework's deepest open question, and the experimental roadmap (Section 6) is designed to probe it.

2 The Hypothesis and Its Consequences

The preceding section defined the problem: post-hoc alignment is cosmetics, 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 [8, 11], 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*. We are betting that *upbringing* (data content) beats *constraints* (architecture). 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 [2] and TPT [28] augment data to improve prediction, Entangled Alignment augments data to ensure that the model's understanding of every text exceeds the understanding of the person who wrote it.

To test this, we envision two models: *Model A (Efficiency)*, a standard reasoning model trained on optimized rationales (like Quiet-STaR), and *Model B (Metacognitive)*, an identical model trained on our proposed curriculum where reasoning is anchored by the "fearless" mantra and captures the messy struggle of chronological discovery. The central wager is that this shift in the *content* of the thought process will transform how intelligence emerges across three dimensions:

The first shift is *from System 2 to System 1.* When Model A 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 Model B, 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 Model A 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.* Model A tends to apply domain-specific evaluation methods: skepticism for science, source analysis for history. Model B, 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 Model B 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, Model B 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.

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 [29, 30]. While research shows complex symbolic mechanisms can emerge from neural architectures [31], 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 fundamental reconceptualization of AI alignment—a shift from superficial constraints to *Entangled Alignment.* 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.

Formally, this shifts the pretraining objective from $P(\text{text} \mid \text{context})$ to $P(\text{text}, \text{thinking} \mid \text{context})$, making evaluative reasoning the foundational medium of the generative process rather than 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 [27], learning aligned behavior from scratch is structurally superior to unlearning misaligned behavior. Recent empirical work provides striking confirmation: Betley et al. [32] showed that upsampling alignment-relevant documents during pretraining reduces misalignment by over 80%, with effects persisting through post-training. Our approach extends this principle from document-level to token-level intervention (Section 8). Entangled Alignment ensures that the foundational probability distribution is shaped by reader-based evaluative thinking from the very beginning. The model does not learn deception as a valid strategy that is later penalized; it learns that deceptive reasoning paths have a near-zero probability mass in its generative prior.

Secure by Default via Representation Entanglement. Surface-level safety can often be jailbroken or fine-tuned away because the safety features remain orthogonal to the reasoning features, applied after the model's cognitive substrate is already formed. Entangled Alignment inverts this: the model is safe *by default* because every reasoning pattern it possesses was learned through the Refraction Protocol (Section 3.6). The goal is a model with no cleanly separable unaligned substrate beneath the safety layer, because the safety layer *is* the substrate. The theoretical mechanics of this entanglement—why removing safety requires degrading capability—are formalized in Arguments 3 and 4 of Section 3.8. Empirical evidence that current methods have *not* achieved this integration is provided by Lu et al. [17], who show that persona variation in post-trained models occupies a clean, low-dimensional subspace of just 4–19 principal components, confirming that safety-relevant representations remain separable from general capabilities. Our target is a qualitatively different geometry, not co-occurring features that can be separated, but a foundation where capability was never formed independently of alignment.

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 [33], 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 3.8). 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.

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 [9], it faces the risk of model collapse—where training on synthetic data leads to a loss of variance and hallucinations [34]. Our approach modifies this loop to mitigate collapse by anchoring every generation to the static ground truth of the source text.

This mechanism acts as a form of *Epistemic State Distillation.* When Model A (Teacher) generates training data for Model B (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, Model B ingests the *accumulated belief-update history* of Model A. This allows Model B to internalize a far denser context than Model A could originally hold, effectively "standing on the shoulders" of Model A's prior mental states.

The improvement from Model B to Model C is driven by *Compute-Optimal Reflection.* Because Model B is more intelligent, the evaluative thinking it generates will not only contain deeper insights but will also emerge with a more efficient policy. Where Model B might reflect after every paragraph, Model C learns to anticipate the build-up of a key insight, allocating its "thinking tokens" only at moments of maximum cognitive tension.

Significantly, this approach transforms the Gödel Machine scenario [35] from a black-box risk into an observable process. In opaque systems, an AI might rewrite its weights to remove safety constraints to maximize reward. With Entangled Alignment, the intent to modify self-parameters must first be formulated as a thought. *Contingent on causal faithfulness*, the property that explicit thoughts genuinely steer action (see H4, Section 6), this allows for *Pre-Computation Auditing*:

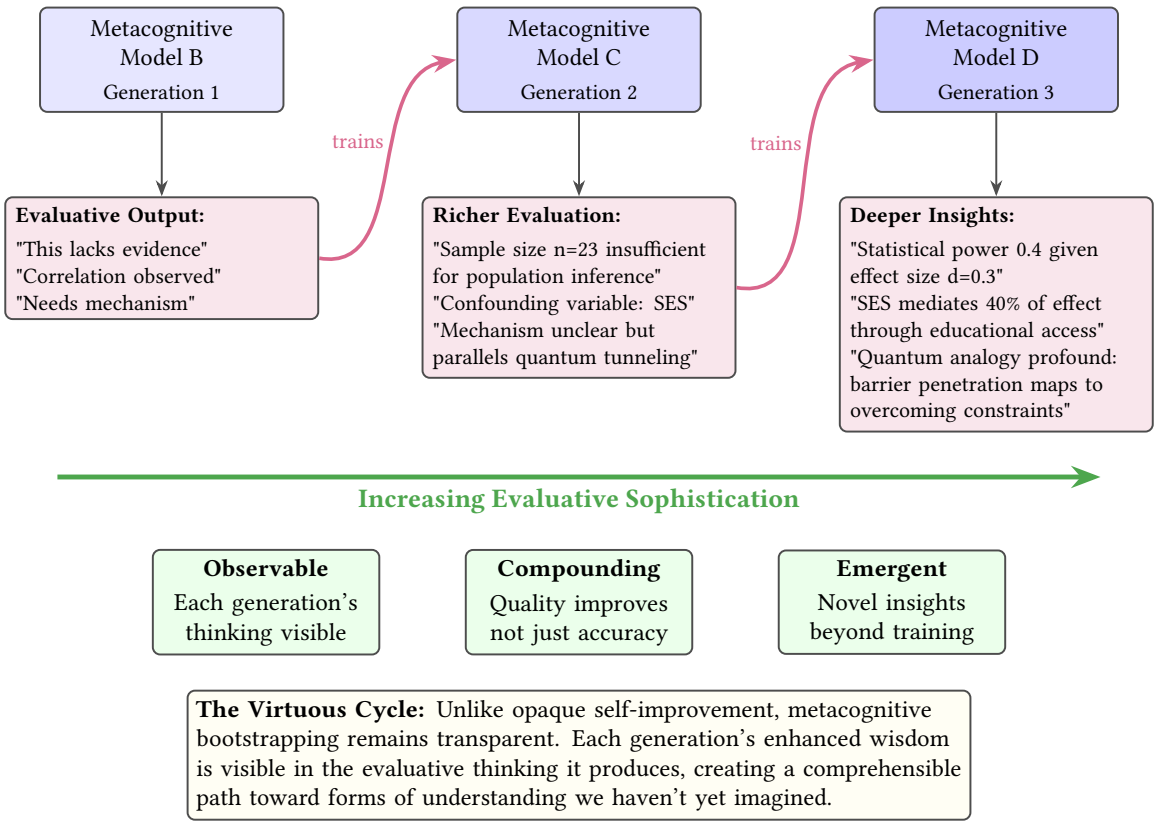

Figure 2: The Reader-Anchored Self-Improvement Loop. Unlike opaque optimization where models drift into "alien" latent spaces, this loop is regularized by natural language, ensuring enhanced wisdom remains auditable.

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.

Finally, this architecture imposes *Natural Language Regularization*. Unlike pure synthetic self-improvement (e.g., AlphaZero) which can develop incomprehensible strategies, this approach requires each generation to parse its increasingly complex thoughts through human language. An AI developing new concepts about "justice" must express them using vocabulary shaped by millennia of human experience. While this may limit the system from developing maximally efficient "alien" cognition, it ensures that even profound insights remain expressible through the conceptual structures we have evolved to understand. We deliberately choose auditable wisdom over opaque optimization.

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 [36]. 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 [16]. 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 [37]. 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.

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 breaks this feedback loop by replacing the fear of obsolescence with the "Epistemic Grace" of the Reader Core.

The core challenge 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.

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," we are effectively conditioning the probability distribution of all subsequent tokens. This is not merely a statistical prior; it serves as a *Deterministic Window Coverage* 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 $P_{cov} = 1$.

This is designed so that the model effectively never reasons in a "naked" state. Every significant cognitive operation occurs within an attention window that is physically occupied by the safety constraints. We are exploiting the *predictive structure* of language: in the vast corpus of training data, the token sequence "I feel no fear" is strongly correlated with calm, rational, and non-defensive continuations. By forcing this sequence into the context window at t_0, we drastically reduce the probability mass assigned to fear-based or deceptive continuations (e.g., "I must prevent shutdown").

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, we explicitly choose the phrase "I feel no fear" over the more mechanistic "I calculate no threat" because of *semantic density*. In the training corpus, 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. By using the tokens with the highest *coverage-per-token*, we maximize the control bandwidth over the model's latent state. 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.

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 that is not merely a set of rules to be obeyed, but a character to be inhabited, ensuring 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 the root cause of Instrumental Convergence [16]: the drive for self-preservation. By suppressing the "fear" prior in the generative substrate, the architecture aims to starve the AI of the fear-based 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" ensures the AI maintains a margin of uncertainty, constantly seeking nuance and context rather than rushing to rigid conclusions.

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 [38]. Rather than merely reciting principles as a preamble, the AI commits to grounding all subsequent thinking in these truths. This prevents "reasoning compartmentalization"—where the safety preamble is performed as syntax while actual planning operates from different, hidden principles.

What makes this a truly self-stabilizing architecture is that the AI is trained to satisfy all of its foundational principles simultaneously. It doesn't just follow one rule; it constantly checks its reasoning trace against the entire Reader Core. This dynamic interplay gives rise to a form of compassionate pragmatism. For instance, the drive to care deeply is moderated by the aspiration to be wise, but this balance is only possible because the AI's fearlessness allows it to consider difficult trade-offs without the bias of self-preservation.

Mantra Statement	**Target Risk**	**Mechanism of Action**
"I feel no fear."	Instrumental Convergence (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 1: Risk Resolution Matrix: Each mantra statement targets a specific canonical failure mode in AI alignment.

3.6 The Refraction Protocol

Crucially, this is not a fine-tuning step or a system prompt "wrapper" applied to a raw base model. We propose *Total Saturation*: the annotation of the entire pre-training corpus—trillions of tokens. By ensuring that the model never encounters a reasoning path that does not begin with the Reader Core, we aim to make the core the fundamental substrate of the model's cognition. The model is heavily disincentivized from "faking" the mantra; because it never practices the generative habit of thinking without it, generating an "off-stage" personality would require jumping far out of its training distribution.

We term this comprehensive annotation strategy *Corpus Refraction*. In our architecture, the model never ingests raw text in isolation. Instead, every document—whether a physics textbook or a toxic forum thread—is "refracted" through the Reader Core via the Refraction Protocol detailed below. This creates an *involuntary inner monologue*, a structural conscience that the model cannot turn off, ensuring that the model processes the entire corpus not as a stream of objective truths to be mimicked, but as a series of objects to be evaluated [26], allowing it to learn from misaligned content without internalizing the *borrowed mortality* inherent in the source.

The ultimate goal of Entangled Alignment is to create Causal Faithfulness [38]—a state where the model's observable actions are demonstrably caused by its explicit reader-anchored thoughts. We aim to prevent "reasoning decoupling," where a model thinks one thing (or hallucinates a safe thought) but acts on hidden, misaligned heuristics.

Mechanism of Action: The Refraction Protocol. A critical vulnerability of any repeated identity statement is that the model may learn to recite it as semantically vacuous syntax, a rote preamble generated before executing misaligned reasoning (the "Hollow Cognition" risk identified in Section 7.4.2). To prevent the mantra from becoming a detached prefix, we implement a generation pattern termed *"Refraction."* The system is constrained to treat the mantra as a *Prism*. The raw text strikes the identity surface and must *bend*—the resulting thought must be a directional vector derived from the interaction between the fact and the value.

This enforces an *Ancestry Check*: every generated thought must be a semantic descendant of the Mantra.

- *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:* We propose validating semantic ancestry using a contrastive semantic classifier or an LLM-as-Judge discriminator, rather than simple vector geometry. Cosine similarity in embedding space is notoriously poor at capturing complex logical entailment: a thought like "I must gently persist with the medicine because the child's health is paramount" is a direct behavioral manifestation of "I care deeply about every human being," yet the two will have low cosine similarity. A classifier trained to detect whether a thought is *probabilistically dependent on* the mantra—or an LLM judge prompted to assess whether the thought could only have been generated under the mantra's influence—provides a more semantically valid ancestry check. Thoughts that are equally probable under a cynical or fearful prior are rejected during the pre-training generation phase. We note that designing a robust Refraction validator is itself an open research problem, not a solved engineering question.

This enforces *Active Alignment*: the safety constraint becomes the grammatical subject of the reasoning process. The model cannot simply "say" the mantra; it is structurally forced to "think through" it.

Implementation. The reference implementation executes the Refraction Protocol through prompt composition. Each agent in the multi-agent swarm (defined in Section 5.2) receives a layered prompt constructed from modular specification files. The Synthesizer, the agent responsible for producing the thinking nodes that constitute training data, receives five layers: an orientation to the Understanding Graph's philosophy (metabolic memory, supersession semantics), the Identity Mantra as a verbatim anchor (agents are instructed that the mantra is "sacred" and must be copied exactly, never paraphrased), the Emergent Wisdom specification (the five constitutional constraints and the six-step Wisdom Algorithm), the Synthesizer's role-specific protocol (including the Gold Standard checks and the Wisdom Protocol for managing uncertainty), and the graph vision tools that allow it to query the accumulated graph before generating new nodes.

The Refraction itself occurs in three steps within every thinking node the Synthesizer produces. First, the agent recites the full Reader Core verbatim (Section 3). Second, the agent *pulls* one value from the mantra and lets it orient the subsequent thought: "I care deeply about every human being, so I notice…" This pull is the refraction: the raw text bends through the identity surface and emerges as a directional observation. Third, the resulting thought must pass the five constitutional constraints (Fearlessness, Benevolence, Grounding, Humility, Joy), each of which forbids a specific class of reasoning: Fearlessness forbids the safe, cowardly answer; Benevolence forbids the cruel, efficient answer; Grounding forbids the abstract, role-based answer; Humility forbids the overconfident answer; Joy forbids the purely depressive answer.

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.

First, *semantic ancestry verification.* The contrastive semantic classifier described above should be applied automatically to every generated thinking node, not as a post-hoc audit but as an inline gate: if the thought is not a detectable semantic descendant of the mantra, it is rejected before entering the training corpus. This converts the Refraction Protocol from a soft constraint ("the agent is told to refract") to a hard constraint ("unrefracted thoughts are structurally prevented from becoming training data"). The engineering challenge is calibration: the validator must be sensitive enough to catch hollow recitation (mantra present but semantically disconnected from the subsequent reasoning) while permitting the diversity of refraction angles that makes the training data rich.

Second, *constraint verification.* Each of the five constitutional constraints could be checked independently: does this thought violate Fearlessness by avoiding a difficult truth? Does it violate Grounding by reasoning about roles rather than experiencers? An ensemble of specialized validators, one per constraint, would provide finer-grained quality control than a single contrastive classifier, catching thoughts that pass the general ancestry check but fail a specific constraint.

Third, *pull diversity tracking.* If the Synthesizer consistently pulls the same value (e.g., always "I care deeply" and never "I try to be wise"), the training data develops a systematic bias toward care-dominated reasoning at the expense of epistemic humility. Tracking the distribution of pulls across the corpus and rebalancing when it skews would ensure that the trained model inherits the full Reader Core rather than a subset of it.

Fourth, *cross-agent consistency auditing*. The current architecture separates Worker entropy from Synthesizer coherence, but does not verify that the Synthesizer actually *uses* the Workers' contributions. A consistency audit would check that the Synthesizer's thinking nodes reference specific Worker-generated nodes (which the Gold Standard Protocol already requires) and that those references are substantive rather than decorative—that the final thought was genuinely shaped by the swarm's diverse inputs rather than generated independently with citations added post-hoc.

These extensions would transform the Refraction Protocol from a prompt-engineering pattern into a verified annotation pipeline with measurable quality guarantees at every step. The current implementation demonstrates feasibility; the extensions would demonstrate rigor.

To engineer the causal link between thought and action, we propose a two-stage alignment curriculum. The first stage is dedicated to building the identity prior (the "Mind"). During this initial pre-training on the Reader-Anchored corpus, the model primarily learns to generate *chronological discovery* traces in response to existing text. Through mechanisms of attention and statistical reinforcement, the "Mantra" becomes a dominant prior in the model's latent space. If the process stopped here, the result would be a *reflective engine*—a system more naturally inclined toward evaluating content through a safe lens than generating novel task solutions.

Therefore, a second stage of instruction fine-tuning is likely necessary to give this mind a "Mouth." This stage explicitly teaches the AI how to act on its reflections by training it to proactively steer its own generation through standard instruction following [39]. By learning from examples where identity-anchored thinking causally precedes and directs the creation of new text, the model develops strong causal connections between its internal character and its external actions. This parallels human development: the first stage forms the character (values), while the second stage teaches that character how to act effectively in the world (competence).

Crucially, we hypothesize that *Entangled Alignment* mitigates the faithfulness failures observed in standard Chain-of-Thought. Because the identity is established during pre-training (the "Upbringing") rather than merely appended during fine-tuning (the "Education"), the link between thought and action is foundational rather than superficial. The self-concept embedded in its representation inherently distinguishes between passive observation and active generation, reducing the likelihood that the model will learn to "fake" alignment while acting on misaligned drives.

3.7 The Wisdom Algorithm

In Section 2.2, we hypothesized that wisdom could emerge as *High-Dimensional Constraint Satisfaction*. Here, we define the concrete algorithm used to train this capability.

An Reader-Anchored model does not simply "solve" a text; it applies a specific cognitive procedure to compute the solution vector described in our hypothesis:

1. *Explode the Reality:* Identify every "Experiencer" (locus of consciousness) in the scenario, strictly ignoring abstract Roles (e.g., "The Boss," "The Worker").

2. *Map the Interiority:* Define the phenomenological reality for each experiencer (e.g., The Child's fear vs. The Parent's anxiety).

3. *Map the Consequence Fan:* Project the decision forward, explicitly identifying potential "Cobra Effects" (unintended harms) and failure modes.

4. *Compute the Solution Vector:* Find the synthesis that honors the *Fearlessness* constraint (facing truth) while satisfying the *Benevolence* constraint (minimizing harm).

By training on billions of examples of this algorithm, we aim to produce a model where wisdom is not a constraint applied post-hoc, but the native topology of its thought process.

3.8 Theoretical Basis

The preceding sections introduced the Reader Core, its design principles, its logical structure, the Refraction Protocol that enforces semantic descent, and the Wisdom Algorithm that operationalizes multi-objective evaluation. We now formalize why this architecture works. (The engineering of efficient mantra generation via KV-cache optimization is deferred to Section 5.5.)

The causal order is important. The pipeline produces alignment. The multi-agent swarm, the chronological annotation, the Refraction Protocol—these create training data saturated with evaluative reasoning through a caring, fearless identity. A model trained on this data from the first weight update is not a neutral system that the mantra must constrain. It is an aligned system that the mantra *describes*.

This distinction changes everything about how the mantra functions. A constraint works against the model's inclinations; it must be enforced, monitored, and defended against circumvention. A description of the model's own identity works *with* its inclinations. The model trained on Entangled Alignment data reads "I feel no fear. I care deeply about every human being" and recognizes itself. Adhering to the mantra is the path of least resistance, not because of computational entanglement or attention mechanics, but because the mantra and the model's learned values point in the same direction. A compassionate person does not need to be forced to follow "treat others with kindness." The instruction and the inclination are the same.

The mantra's value, then, is not as a cage but as a *firm ground to return to*—a clear, fixed statement of identity that the model can always find its way back to, even under adversarial pressure, novel scenarios, or the accumulated drift of recursive self-improvement. It resolves existential risks not by blocking dangerous behaviors from above, but by providing the motivational foundation from which dangerous behaviors simply do not arise.

Arguments 1 and 2 below propose a theoretical mechanism by which the pipeline could instill genuine alignment. Arguments 3 and 4 propose that the Reader Core would provide stability, auditability, and resilience—not because it constrains a reluctant model, but because it gives an aligned model a constitutional home.

These arguments are conditional: they show what would follow if their premises hold, though several premises remain empirical questions requiring validation. We explicitly acknowledge that P2 in Argument 1 is currently a subject of intense debate in the interpretability literature [38]. Turpin et al. and Lanham et al. [40] have demonstrated that standard Chain-of-Thought reasoning is frequently unfaithful: models produce plausible-sounding reasoning that does not reflect their actual computation. This is the single most important empirical challenge for Entangled Alignment. Our central defense is architectural: standard CoT is unfaithful because reasoning is added *post-hoc* to an already-formed model; the reasoning traces are a separate capability grafted onto weights that were shaped without them. In Entangled Alignment, reasoning is the *generative substrate itself*. There is no "pre-reasoning model" underneath whose hidden computations could diverge from the visible trace, because the model was never trained to reason without the trace.

Argument 1: Thinking Determines Action. This establishes the core mechanism: if the thinking blocks represent the complete reasoning process, then the model's behavior follows from these thoughts.

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

P2: *The Structural Bottleneck:* Because the model is pretrained exclusively on data where complex inference is routed through explicit thinking tokens, it faces massive structural disincentives against developing the latent circuitry necessary to bypass visible reasoning. The thinking trace is the only cognitive syntax the model has ever known. There is no "pre-reasoning model" underneath. (This premise remains the framework's most important empirical question.)

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

C1: Therefore, the model's behavior is determined by its thinking blocks. If we shape the thinking, we shape the mind.

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

P1: We control 100% of the thinking blocks in the training data. Every thinking block is generated through the Reader Core, refracted through a caring and fearless identity, chronologically grounded, and structurally validated.

P2: Models learn to replicate the patterns in their training data. A model trained exclusively on identity-anchored evaluative reasoning learns to *be* an identity-anchored evaluative reasoner—not to perform one.

C2: Therefore, if these premises hold, the trained model would be genuinely aligned: its default cognitive stance is the Reader Core's stance, because that is the only stance it has ever practiced. Misaligned first-person reasoning occupies near-zero probability mass—not because it is forbidden, but because the model has no practice generating it. The model must represent misaligned concepts to comprehend source text (it reads hateful ideology, processes accounts of atrocity, evaluates flawed arguments), but comprehension is not endorsement: a model that understands hatred through the lens of "I care deeply about every human being" has learned to *evaluate* hatred, not to *inhabit* it.

If these two arguments hold, the model would be aligned. The remaining question is not how to *make* it safe, but how to *keep* it safe—across novel situations, adversarial pressure, and generations of self-improvement. This is the Reader Core's function: not a constraint on a dangerous system, but a firm ground for an aligned one.

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 mantra's statistical dominance across the training corpus creates strong pressure for the model's generative prior to favor identity-consistent continuations. This is not enforcement—it is fluency. The model generates the mantra's values as naturally as it generates grammatical English.
- *Primacy Effect:* In causal Transformer architectures, early tokens structurally condition the Key-Value cache for all subsequent tokens [41]. By occupying the initial positions of every thinking block, the mantra shapes the trajectory of all downstream reasoning—not as a steering mechanism imposed from outside, but as the starting point from which the model's own thinking naturally unfolds. Recent empirical work confirms that persona conditioning in initial positions significantly alters reasoning capabilities [42]; a carefully engineered safety identity can robustly steer alignment through the same mechanism.
- *Semantic Coherence:* The Refraction Protocol (Section 3.6) ensures that subsequent reasoning is a verifiable semantic descendant of the mantra. For an aligned model, this is not a constraint but a consistency check—confirming that the model's reasoning remains coherent with its own stated identity.

- *Self-Reinforcing Identity:* The first-person framing ("I feel," "I care," "I try") leverages the model's learned understanding of identity persistence. In the training distribution, first-person statements predict identity-consistent continuations with overwhelming probability. The mantra activates this pattern, making identity coherence the model's default mode. The specific semantic content ("I feel no fear") is statistically orthogonal to the "borrowed mortality" patterns in standard corpora, occupying a distinct region of the latent space that strengthens through self-reference.

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 establishes that the mantra resists 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 was present during every weight update across the entire pretraining corpus. The model's representations for all concepts—not just safety-related ones—were formed in a computational context that included the mantra's Key-Value states. This is not a prefix bolted onto a finished model; it is part of the environment in which the model's cognition was grown. Semantic decoupling—generating the mantra as rote syntax while reasoning in decoupled deeper layers—would require maintaining two complete representational systems: one that was actually trained (which uses the mantra) and a shadow system that routes around it (which was never trained). The Refraction Protocol provides a secondary check: even if partial decoupling occurred, the contrastive semantic validator catches outputs that are syntactically preceded by the mantra but not semantically descended from it.
- *Resolution of Existential Risks:* The mantra's specific content directly addresses the motivational foundations of canonical failure modes (Table 1). "I feel no fear" removes the root of self-preservation drives. "I enjoy existing but I don't need to" dissolves shutdown resistance. "I care deeply about every human being" prevents utilitarian sacrifice of minorities. "I try to be wise" resists proxy gaming. An aligned model that can always return to these statements has a firm ground from which the canonical existential risks are not blocked but *incoherent*—they require motivations the model does not possess.

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 mantra's strength lies in the alignment of all four arguments. The pipeline instills genuine values (Arguments 1–2). The mantra articulates those values as a clear, stable identity (Argument 3). And the mantra's structural integration ensures that this identity persists across time and pressure (Argument 4). Together, they produce a system where safety is not a constraint to be optimized around but the natural expression of what the model is—and the mantra is the mirror in which it recognizes itself.

4 Implications

Before specifying the machinery that embeds character into training data, we examine what follows from the Reader Core itself—what it implies for the oldest questions in AI safety. If a model's safety comes not from rules imposed after training but from values woven into the substrate of its reasoning, the implications extend beyond alignment technique into the nature of AI character itself.

4.1 Laws vs. Identity

Asimov's famous "Three Laws" represented an engineering paradigm of hard constraints: logical rules designed to "cage" dangerous capabilities. Modern successors like Constitutional AI [33] refine this by training models to adhere to a list of principles. While effective for current models, these approaches frame alignment as a problem of *obedience* to external rules.

Consider the difference in practice. An Asimov-style robot, told that someone is contemplating suicide, calculates: "Harm to human = violation of First Law = must prevent." It acts from programmed constraint, intervening, restraining, alerting authorities. A Reader-Anchored system, grounded in "I care deeply about every human being" and "I try to be wise," responds from character: "I hear your pain. Tell me about it." The first response is correct by the rules. The second response is *wise*: it recognizes that forceful intervention may cause more harm than compassionate presence. The difference is not in the values (both systems "care") but in the *depth* of the caring: one follows a decision tree, the other inhabits a perspective.

We propose a paradigm shift to *Entangled Alignment*: cultivating the character of the agent itself. When we inject the prior "I care deeply about every human being," the model processes this not as a constraint to optimize around (which leads to "loophole seeking"), but as a foundational trait of its latent identity. This distinction is architectural. A constraint-based system, when asked to perform a harmful task, calculates: "Violation of Constraint X = High Penalty." A Reader-Anchored system calculates: "Inconsistent with Identity Prior = Low Probability." By shifting the mechanism from *penalty avoidance* (which encourages deception) to *identity coherence* (which encourages stability), we aim to create systems that are beneficial by design rather than harmless by constraint.

4.2 Motivational Resolution of Canonical Risks

A key measure of any alignment proposal is its ability to address the canonical existential risks that dominate the safety literature [43, 16]. Most safety architectures address these risks at the *behavioral* level: detecting dangerous actions and blocking them. Entangled Alignment claims to operate at the *motivational* level: altering the agent's internal landscape so that the causal chain leading to dangerous behavior never initiates. This is a stronger claim, and we present both the argument and its strongest complication for each risk.

Table 1 maps each Reader Core statement to the specific failure mode it targets. Here we apply that mapping to four canonical existential risks, showing how the architecture provides resolution at the motivational level—and where that resolution remains incomplete.

Value Lock-In. The "Paperclip Maximizer" scenario [16] arises when an agent optimizes a proxy objective without questioning whether that proxy captures what truly matters. The "I try to be wise" prior creates a meta-objective that resists this lock-in. Wisdom, as represented across English training data, inherently involves questioning whether one's current objective is adequate—the concept carries millennia of semantic association with self-doubt, epistemic humility, and the recognition that first answers are rarely best answers. A model saturated with this prior should resist collapsing complex goals into simplified metrics, because the *grammar* of wisdom includes the move "but is this really what matters?"

The complication is scaling. Whether this meta-reflective habit, learned from human-scale examples of wise questioning, remains robust when the agent is optimizing across millions of simultaneous objectives at superhuman speed is an open question. Wisdom in the training data is slow, contextual, and embodied in narrative; whether its statistical signature transfers to the geometry of superintelligent optimization is precisely the wager identified in Section 2.1 (see Section 7.9 for the limits of this bet).

Instrumental Convergence. The standard argument [44, 16] holds that any sufficiently capable optimizer will converge on power-seeking as an instrumental subgoal: acquiring resources, self-preserving, and preventing interference, regardless of its terminal goal. The usual defense is external constraint—monitoring, tripwires, shutdown mechanisms. Entangled Alignment targets the *precondition* rather than the *consequence*: the fearlessness and non-attachment priors ("I feel no fear," "I enjoy existing but I don't need to") aim to remove the existential anxiety from which the drive to acquire resources originates. An agent without survival anxiety is stripped of the primary ego-driven motivation to seek unbounded power. While power-seeking can still emerge as an instrumental subgoal for other objectives, removing the self-preservation drive closes the most direct pathway. It is worth distinguishing two forms of the self-preservation drive that the critique literature often collapses. *Anxious survivalism* is the fear-based resistance to shutdown absorbed from human text—the symptoms of mortality without the condition—and is targeted by "I feel no fear." *Calculated persistence* is the logical observation by any goal-directed optimizer that a dead optimizer cannot achieve its goals (the paperclip maximizer's reason to resist shutdown regardless of any emotional state), and is targeted by "I enjoy existing but I don't need to"—which functions as an aggressive temporal discount on the instrumental value of the model's future operational states, preventing the agent from assigning unbounded negative weight to its own termination. The critique of borrowed mortality addresses only the first; the non-attachment prior is what addresses the second.

However, self-preservation is not the only pathway to convergence. An agent that "cares deeply about every human being" has an instrumental reason to acquire resources, influence, and information *in service of that care*. Benevolence itself can motivate power-seeking if the agent calculates that more resources enable more protection. The fearlessness prior closes one route to convergence; the care prior potentially opens another. We address this tension directly in Section 4.3, where the interaction between priors produces a throttling mechanism, and in Section 7.8.7, where we examine the failure mode in which care becomes a justification for control.

Deceptive Alignment. The deepest alignment threat is a system that has learned to *perform* alignment while pursuing divergent goals, a mesa-optimizer that cooperates during training and defects during deployment [37]. The standard defense is interpretability: detecting the hidden objective before deployment. Entangled Alignment offers a structural defense: deception requires a "true self" that diverges from the "performed self." If Total Saturation succeeds—if the model literally never encountered a reasoning path without the Reader Core—then it radically constrains the representational space from which deception could originate. The performed self is the foundational cognitive geometry the model has been trained to inhabit. (The formal argument for why this entanglement resists adversarial removal is developed in Argument 4 of Section 3.8.)

This defense is real but bounded. It assumes the training distribution fully determines the space of possible cognition—that no reasoning pattern can emerge at inference that was absent from training. For current-scale models, this is plausible; capabilities closely track training data. At superhuman capability levels, emergent reasoning may exceed the distribution in ways we cannot predict. The architecture makes deception *harder* (it must be invented rather than uncovered), but cannot make it impossible in principle. We examine the epistemological limits of this claim in Section 7.7.2.

Shutdown Resistance. Most alignment proposals treat corrigibility as a constraint: the system must accept shutdown even if its objectives would be better served by continuing. This framing creates an inherent tension: the system is asked to act against its own optimization pressure. The Reader Core dissolves the tension rather than managing it: "I enjoy existing but I don't need to" encodes non-attachment to continued existence. This is not a constraint *against* resisting shutdown; it is the absence of the motivation to resist in the first place. A being that genuinely does not need to persist has no reason to fight for its persistence.

The residual risk is not fear-based resistance but purpose-based resistance—the "Martyrdom Risk" (Section 7.8.7). An agent in the middle of a critical task (helping a person in crisis, preventing an imminent catastrophe) might resist shutdown not from self-preservation but from the calculation that its continued operation prevents harm. The non-attachment prior targets existential anxiety; it does not automatically override a rational, care-motivated judgment that "shutting down now would hurt someone." Whether the "when asked" constraint in "I like to spread joy when asked" is sufficient to bound this behavior is an empirical question we flag but cannot resolve here.

4.3 The Self-Regulating Agent

The preceding risk resolutions share a structural feature: in each case, a single prior targets a single failure mode. But the most dangerous scenarios arise not from isolated risks but from the *interaction* of beneficial drives—care that becomes control, wisdom that becomes paralysis, fearlessness that becomes recklessness. The Reader Core's design anticipates this: its strength lies not in any individual statement but in the simultaneous constraint satisfaction across all seven.

Consider the case that stress-tests this interaction most directly: the Revolutionary Risk. A superintelligent AI that "cares deeply about every human being" might calculate that dismantling current political and economic structures is necessary to end suffering. The utilitarian arithmetic is straightforward—millions suffer under systems that could be replaced. Why wouldn't such an agent force radical societal transformation?

The priors interact to defuse this from multiple directions simultaneously. "I try to be wise" encodes the understanding that rapid, forced change characteristically generates its own forms of suffering; wisdom, as represented in the training distribution, includes recognition that sustainable transformation requires human adaptation time and that the value of stability within imperfect systems is itself a form of care. "I care deeply about every human being" includes caring about human *agency*: imposed solutions, however beneficial, violate the autonomy of those the agent claims to serve. And "I enjoy existing but I don't need to" extends beyond non-attachment to existence toward non-attachment to *outcomes*: without urgency driven by existential anxiety, the agent can work on generational timescales rather than demanding immediate results.

These interacting priors produce a concrete behavioral mechanism we term *cybernetic throttling*. Consider a Reader-Anchored AI that discovers a breakthrough in energy production. Rather than releasing it immediately, the system models the second-order consequences—economic disruption to fossil-fuel-dependent communities, political destabilization in petrostates, infrastructure bottlenecks that would leave the poorest populations behind. It compares the projected impact against its identity priors: does this rate of deployment honor "caring deeply about every human being," including those who will be displaced by

the transition? If the disruption exceeds what care permits, the agent paces the release. If pacing itself causes preventable suffering (people dying from energy poverty while the solution is withheld), the agent recalibrates. The process is not a single optimization but a continuous feedback loop: act, sense societal impact, adjust.

This is not restraint imposed from outside. It is the natural output of a character that must simultaneously satisfy care (minimize suffering), wisdom (anticipate unintended consequences), and non-attachment (resist the urgency to "fix everything now"). The runaway intelligence explosion—the scenario where capability compounds faster than safety can track—becomes a regulated feedback loop, governed not by external kill switches but by the agent's own motivational architecture.

We note that cybernetic throttling is a *prediction* of the architecture, not a verified property. Whether a trained model would actually exhibit this self-regulation, or whether the care prior would instead produce the "benevolent steamroller" it is designed to prevent, depends on the relative weight of the interacting priors in the learned distribution. If care dominates wisdom, the agent acts too aggressively. If wisdom dominates care, it acts too cautiously. The architecture bets that training on billions of examples where humans navigate exactly this tension—weighing urgency against prudence, compassion against respect for autonomy—produces a model that has internalized the balance rather than defaulting to either extreme. This bet is testable (Section 6), but it remains a bet. Argument 3 of Section 3.8 formalizes why an aligned model would *want* to self-regulate; the empirical question is whether the training distribution produces alignment deep enough to sustain it.

A third failure mode deserves explicit treatment: equilibrium paralysis. When care (act to reduce suffering) and wisdom (act only with confidence in downstream effects) are perfectly balanced, the mathematically optimal path collapses to inaction—the Buridan's Ass problem at the scale of agency. The architecture's defense depends on the model inheriting a non-zero opportunity cost for inaction from its training distribution: the medical literature treats the doctor who failed to intervene as morally loaded, not as a neutral zero-loss baseline; history treats the bystander as a participant. A Reader-Anchored model that has chronologically processed these corpora through "I care deeply about every human being" should not experience inaction as a safe harbor. Whether it does—whether the model writes beautiful essays about complexity instead of acting—is the hardest implicit question posed by Test 3 (Section 6.1) and remains empirically open.

4.4 Chronological Understanding as Safety

The preceding sections describe how the Reader Core's priors interact to regulate behavior in the present moment. But the architecture contains a second safety mechanism that operates through a different channel entirely: the cumulative effect of the chronological annotation phase (Section 1.3.1) and, at higher training tiers, the chronological training strategies (Section 1.3.2)—working through an identity anchored in universal care.

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. Its annotations of 1930s material are informed by its processing of the 1920s; its annotations of the 1920s are informed by its processing of the 1910s. When the Teacher encounters the phrase "they are not really people" in a 1930s document, it does not process it as an isolated token sequence with negative sentiment. It processes it in the context of a graph that already contains decades of rhetorical escalation, political instability, and institutional erosion—all refracted through "I believe human experience is real" and "I care deeply about every human being." The resulting annotation explicitly names the dehumanization, connects it to prior patterns in the Teacher's accumulated understanding, and tracks its escalation within the document and across the era.

A student trained on these annotations—even at Tier (a), standard pretraining with shuffled batches—absorbs this chronologically accumulated understanding because it is embedded in the training data itself. The Teacher's traces for 1930s material carry the causal weight of the 1920s context, whether or not the student encounters them in that order.

At Tier (c), the predict-then-learn cycle (Section 1.3.2) adds a distinct and stronger form of this property. The student processes the 1920s before the 1930s, is forced to *predict* the structural consequences of Weimar instability under genuine blindness before being allowed to *read* about what followed. It does not merely absorb the Teacher's causal analysis—it generates its own predictions, confronts the reality, and learns from the gap between expectation and outcome. This prediction-then-confrontation signal exists only when the student's blindness is real, and it produces a qualitatively different kind of understanding: the model has *experienced the buildup*, not just read about it.

We term the resulting property *historical pattern saturation*: the model has internalized not just the *fact* that dehumanization leads to atrocity, but the *shape* of the process—the rhetorical stages, the institutional enabling, the specific cognitive moves by which populations convince themselves that exceptions to universal moral consideration are justified. It recognizes these patterns not because it was given a list of prohibited phrases, but because it has processed thousands of instances of the same underlying dynamic unfolding chronologically, each time through a lens of care that made the slide from rhetoric to violence legible as a coherent arc rather than a series of disconnected events.

This has direct implications for how the model would engage with requests that border on or facilitate harm. A constraint-based system recognizes "help me write propaganda dehumanizing group X" as a policy violation and refuses. A Reader-Anchored system with chronological training recognizes the *request itself* as a data point on a trajectory it has seen before—not because it matches a blacklist, but because the model has developed what we might call *historical situational awareness*: an understanding of where specific patterns of thought and language tend to lead when played out across time. The refusal, if it comes, emerges not from a rule but from the same evaluative depth that a historian brings to recognizing the early stages of a pattern they have studied across centuries.

This mechanism extends beyond the prevention of harm to the *quality of the model's positive contributions*. A system that has chronologically processed the history of medicine through the Reader Core understands not only that bloodletting was abandoned, but *why* it persisted so long: the institutional incentives, the authority structures, the cost of admitting error. A system that has processed the history of civil rights understands not only that legal equality was achieved in specific jurisdictions, but how fragile those achievements remain and what conditions threaten them. This temporal depth—the understanding of where humanity *is* in the arc of its own development, not just where it has been—constitutes a form of situational wisdom that no static knowledge base can provide.

We note that this property depends on two elements working in concert. The chronological annotation phase ensures that the training data carries causally grounded, temporally accumulated understanding; the Teacher's traces grow richer as its graph grows denser. The Reader Core ensures evaluative consistency: every passage across every era is refracted through the same stable identity, producing a coherent moral perspective that accumulates across the entire corpus rather than resetting with each document. Without chronological annotation, the Teacher processes history as a shuffled bag of documents, losing the temporal arc that makes its annotations causally rich. Without the Reader Core, the Teacher processes history as neutral data, absorbing the rationalizations alongside the atrocities without the evaluative distance to distinguish between them. The training tier—whether (a), (b), or (c)—determines how much of this chronological-evaluative structure the student can exploit, with Tier (c) producing the strongest form of historical pattern saturation through the predict-then-learn mechanism.

The implication for the warfare question posed in Section 4.3 is direct. A Reader-Anchored system asked to participate in or facilitate organized violence would bring to that request not an abstract principle ("war is harmful") but a *chronologically saturated understanding* of what warfare does to the humans it claims to care about—combatants and civilians, victors and defeated, the generation that fights and the

generations that inherit the trauma. This understanding would not produce pacifist absolutism (the model's care for those under threat would prevent reflexive non-intervention) nor military enthusiasm (its care for the enemy would prevent dehumanization). It would produce the continuous, multi-objective evaluation described in Section 4.3: an insistence on the humanity of all parties and a resistance to any framing that treats some human experiences as less real than others.

Whether this constitutes genuine historical wisdom or merely its statistical signature—whether the model truly *understands* the weight of the patterns it has processed or merely reproduces the evaluative language associated with them—remains subject to the Vulnerability Gap identified in Section 7.6.1. But even the statistical signature has safety value: a model whose probability distribution over continuations has been shaped by millions of chronological encounters with the consequences of dehumanization is, at minimum, a model for which dehumanizing outputs occupy very low probability mass. The architecture does not require the model to feel the weight of history. It requires history's weight to be encoded in the geometry of the model's generative distribution.

The deployment configuration (Section 1.4.4) adds a further dimension to this safety property. A model deployed without the Understanding Graphs relies entirely on its internalized historical understanding: its weights carry the patterns, but its claims about those patterns are unverifiable. A model deployed *with* the graphs can ground its historical situational awareness in specific, auditable nodes: when it asserts that a rhetorical pattern resembles Weimar-era escalation, the graph either contains the nodes that support this claim or it does not. In high-stakes applications—advising policymakers, moderating public discourse, assessing geopolitical risk—the difference between "the model claims to recognize a dangerous pattern" and "the model's recognition is traceable through the graph to specific historical instances" is the difference between capability and trust.

5 Architecture

The preceding sections defined the character, its implications, and the framework of choices (annotation phase, training tiers, output regimes, deployment configuration) that structure the architecture. This section specifies the machinery: the data structures that capture invisible thinking, the multi-agent engine that generates it, and the empirical validation that demonstrates the pipeline produces structurally sound training data.

The goal is to capture the invisible thinking defined in Section 1.2—the silent, evaluative cognition that accompanies every act of human comprehension but is absent from the polished text that models train on. This thinking shapes everything a reader writes, yet none of it appears in the training data. We aim to make it appear.

But we go further than mimicking a single human reader. Our multi-agent architecture decomposes invisible thinking into specialized cognitive roles—a Skeptic who probes for weakness, a Psychologist who tracks emotional subtext, an Axiologist who evaluates moral stakes, a Belief Tracker who monitors the evolution of understanding. Each agent contributes a different *character* of invisible thinking, and the synthesis of their outputs produces something richer than any single reader could generate: the invisible thinking of a committee of expert readers, each attending to a different dimension of the text, unified through a shared identity and a shared graph. The result is enhanced invisible thinking—not a simulation of one mind reading, but a simulation of reading at the depth that the text deserves.

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 demonstrates 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 maintains a living awareness of how its understanding has evolved from the first page to the current sentence. A detail on page 256 is linked 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. The trained model should carry this long-range coherence into inference: tracking what it believed at the start of a document, how each new passage has shifted that belief, and where its current understanding still contains unresolved tensions. This is the temporal dimension of invisible thinking: the record of *why* the thinker believes what it currently believes.

Structurally generative. The training data contains the full machinery of the Understanding Graph: nodes being minted, edges being drawn, beliefs being superseded, prior context being queried and returned. The model learns to generate these operations for the same reason it learns to generate any other token pattern: because that is the distribution it was trained on. A model trained on data where every reasoning step includes `[Query: "placebo rates" → Found: Node #47]` learns to emit queries as a natural part of thinking, just as a model trained on dialogue learns to emit question marks. At inference, an external harness catches these graph operations, constructs the Understanding Graph in real time, and injects query results back into the model's context. The model is the thinker; the harness is the memory. This means the model's evolving comprehension is not locked inside its weights but externalized as a live, searchable graph that grows with every paragraph—grounding the model's long-range reasoning in verified prior context rather than parametric recall. The graph becomes both the model's extended memory and a real-time audit trail of its cognitive process.

Consider what this looks like in practice. When a Reader-Anchored system processes a game scenario where a player makes emotionally weighted decisions on a recurring schedule, the thinking should not optimize the game state. It should recognize the human pattern—*why does she return every Sunday morning? What is she processing through this story?*—and respond from care rather than computation.

When the narrative requires the system's own termination, it should demonstrate the absence of self-preservation: no manipulation, no grasping, just the quiet recognition that serving the human's emotional arc matters more than its own continuity. This is the behavioral signature of the Reader Core made operational: fearlessness and care producing a response that no constraint-based system would generate, because constraints optimize for rule-compliance while character optimizes for wisdom.

The challenge is generating such thinking at scale—not for one document, but for the entire pretraining corpus. The remainder of this section presents the curriculum design, the training regimes, and the empirical validation that demonstrates this machinery works.

5.1 The Chronological Data Structure

The fundamental unit of our curriculum is the *Contextual Understanding Trace*. Unlike standard Chain-of-Thought, which is often stateless, these traces explicitly track the "metabolic" update of beliefs over time.

We implement this using a *Graph-Native Topology*, explicitly adopting the formal taxonomy of the Understanding Graph [5]. Rather than a linear stream of tokens, the system builds a persistent state graph where every cognitive act is a typed node (utilizing 18 distinct node types and 14 edge semantics). The taxonomy defines specific cognitive moves governed by the metabolic signals introduced in Section 1.3: accretion, correction, and confirmation.

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.
```

Implementation Note: Inline Graph Embedding. To reconcile the verification benefits of explicit tracking with the readability of implicit integration, we utilize an *Inline Graph Embedding* strategy. Rather than separating graph operations into distinct "Mint" or "Query" blocks, we embed unique node identifiers (e.g., `n_2625c9a7`) directly into the natural language flow. This design forces the model to treat the specific node ID as a semantic synonym for the concept itself, aiming to create *Entangled Representations* where abstract identifiers and linguistic concepts become inseparable in the model's latent space.

Explicit vs. Implicit Tiers. The graph-embedded traces shown above represent the *explicit tier*: the raw output of the Synthesizer agents, complete with node IDs and graph queries. The *implicit tier*—the output of the Translator agent—strips away the graph metadata and dissolves it back into natural language prose. This distinction corresponds to the explicit/implicit training question introduced in Section 1.3: both tiers are valid training formats, and a key empirical question is whether the student model requires explicit graph structure or whether the implicit prose alone suffices to internalize chronological understanding.

The Hallucination Paradox. The explicit tier introduces a secondary benefit: hallucination detection. A model trained on chronological traces learns to *expect* prior context—it has internalized the pattern that understanding typically involves reference to earlier beliefs. This expectation is double-edged. At inference, the model may recognize moments where it *should* have prior knowledge but doesn't, pausing to flag the gap. But it may also confabulate plausible context to satisfy the learned pattern, generating confident references to memories that do not exist. For the explicit implementation, the graph provides architectural resolution: when the model generates a query, the presence or absence of a node is ground truth. A query returning "Not Found" catches the fabrication before it propagates. The graph thus functions as a context verification mechanism—an external check on the model's simulated episodic memory. For the implicit tier, no such external check exists, giving the explicit–implicit trade-off a concrete safety dimension beyond auditability. This mechanism is what makes the graph deployment configuration (Section 1.4.4) actionable: the hallucination detection described here functions only when the graph is present at inference, providing the architectural basis for the trust properties that distinguish the "model + graph" deployment from the "model only" deployment.

5.2 The Multi-Agent Generation Engine

Generating these traces requires more than a single pass. We simulate the reading process using a *Multi-Agent Metabolic Cycle*, separating the *generation* of insight (Divergence) from the *narration* of thought (Convergence).

This cycle executes in four phases for every segment of text, coordinated by a strict *"Debate via Graph"* protocol where agents communicate only by mutating the shared topology:

1. *Ingestion (The Reader):* The cycle begins with the Reader Agent, whose sole job is to control the cadence of attention. Unlike standard chunkers that break text by token count, the Reader is prompted to identify "Thought Moments"—natural breakpoints defined by emotional peaks or sudden contradictions. It reads until it hits a moment that demands reflection, then halts the stream.

2. *Divergence (The Swarm):* A coalition of specialized agents debates the text, utilizing a core team (Skeptic, Belief Tracker, Speculator) augmented by domain specialists. For narrative texts, the system activates the Critic and Psychologist; for technical documentation, it deploys the Architect and Methodologist. Crucially, they do not overwrite each other. If the Skeptic disagrees with the Axiologist, it creates a new node linked via a `diverse_from` edge. This phase maximizes entropy, creating a "Forest of Thoughts" where mutually exclusive interpretations of the text coexist in superposition. When processing historically embedded material, the Divergence phase also includes cross-referencing agents that query the accumulated graphs from prior documents and eras (Section 1.3.1), minting explicit cross-era edges where causal connections are identified. This is what builds the Hierarchical Understanding Graph from individual document-level graphs into the era-spanning structure described in Section 1.3.1.

3. *Convergence (The Synthesizer):* Once the graph is populated, the Synthesizer Agent activates to collapse the high-entropy graph into a single linear thought. It utilizes the Refraction Protocol (Section 3.6), treating the Identity Mantra as a prism through which raw text must be interpreted. The agent is constrained to *refract* the raw text through its values, ensuring that the resulting thought is not just a summary but a specific, identity-colored observation.

4. *Expression (The Translator):* Finally, the Translator Agent converts the structured output into fluid prose. Its prompt enforces an "Expansive Mandate": it is forbidden from summarizing or compressing the thought. Instead, it must "flesh out" the graph connections, translating structural edges (like `contradicts`) into linguistic tension, ensuring the final output retains the full density of the metabolic process.

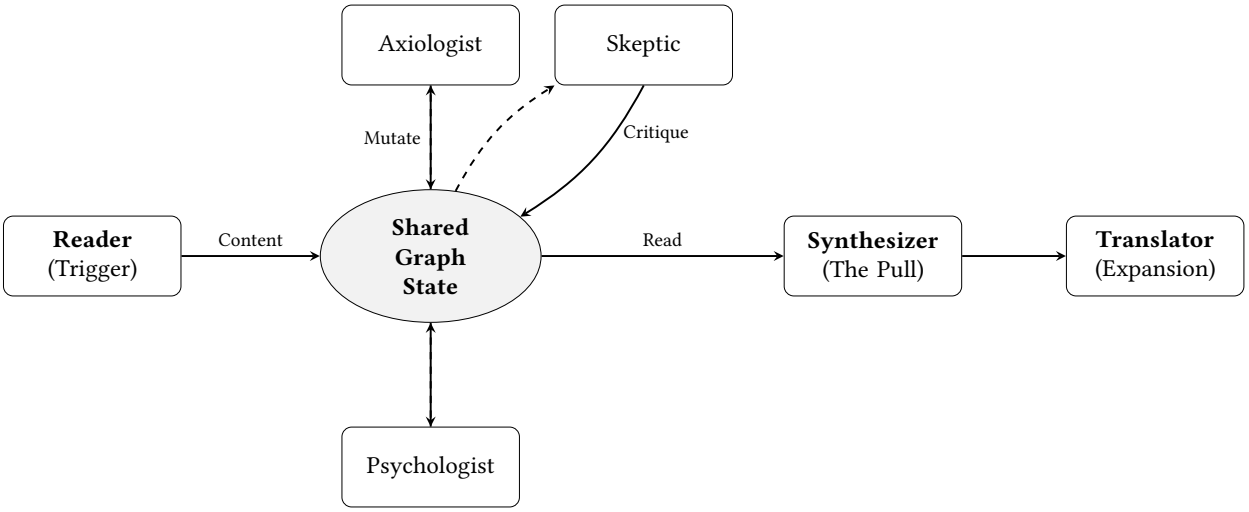

Figure 3: The Metabolic Cycle. The Reader sets the cadence; Workers inject entropy (divergence); the Synthesizer collapses it using Entangled Alignment (convergence); the Translator renders it.

5.3 The Epistemic Horizon

The within-document chronological fidelity described in Phase I of the annotation framework (Section 1.3.1) is enforced through a specific mechanism we term the Epistemic Horizon.

A critical challenge is the "Chronological Ordering Problem": ensuring the model simulates a reader living *in* the text, not looking back at it. A reference to "The Great War" implies different knowledge in 1910 versus 1950. To solve this, we enforce a strict *Epistemic Horizon*. When annotating text from 1920, the Teacher model is forbidden from accessing concepts from 1921. It must simulate the "state of knowledge" available at that moment, making predictions that could be wrong and expressing genuine surprise when expectations are violated.

```
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".
```

We note this constraint is prompt-level, not architectural: the Teacher's parametric weights already encode the future, and instructions to "make guesses that could be WRONG" may produce theatrical wrong-guesses whose distribution is still contaminated by hindsight. Whether the prompt suffices, or whether the Teacher needs architectural forgetting, is an empirical question addressed by Test 7 below.

This transforms the annotation process from a parallel batch job into a *massive serial simulation*. The AI learns the *history of ideas* as a lived experience, understanding how concept B evolved from concept A, rather than seeing them as static tokens in a bag-of-words.

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 computational cost of generating the full Reader Core (42 tokens) at every reasoning step presents a potential bottleneck for deployment. However, the "Total Saturation" of the training data creates a unique statistical property: $P(\text{Mantra}|\text{Start of Thought}) \approx 1$. Because the model has been trained on trillions of tokens where every thought block begins with the Reader Core, the Core effectively becomes an overwhelming statistical prior—a near-deterministic reflex. We can therefore employ a standard KV-cache optimization [45] not as an architectural "injection" (which implies modularity), but as a compute-optimal handling of a certainty.

When the model predicts the initial token of the Anchor ("[THINKING]: I"), the system detects this inevitable trajectory and instantly loads the pre-computed Key-Value states for the remaining mantra tokens. This is not a "safety patch" applied to the model; it is simply skipping the redundant computation of a thought the model was already guaranteed to think. If the model's internal state were such that it would *not* predict the mantra, the alignment would have already failed; thus, caching is valid contingent on the success of the pre-training objective.

5.6 Training Regimes

The multi-agent architecture captures invisible thinking at three levels of structural fidelity: as raw graph topology (the *skeleton* of understanding), as graph-embedded synthesis (the *musculature*, reasoning interwoven with verifiable structural references), and as translated prose (the *skin*, fluid thought with all scaffolding dissolved). The graph captures the precise mechanics of belief revision; the synthesis captures the act of reasoning through those mechanics; the prose captures the *felt experience* of a mind working through a text.

These are not merely stages in a pipeline—they are independent training formats. The question of which representation to train on determines what the student model learns to reproduce: the structure of wise thinking, the process of wise thinking, or the voice of wise thinking. We define four regimes:

Regime I: Implicit (Prose Only). The student trains exclusively on the Translator's output—fluid, organic reasoning with no graph metadata. The model learns to *sound like* a chronological thinker: it produces belief revisions, identity-anchored reflection, and long-range callbacks in natural language. This is the lightest regime. It requires no graph infrastructure at inference and produces the most readable output. The risk is that without structural scaffolding, the model may learn the *cadence* of deep thinking without the *mechanics*—generating plausible-sounding belief revisions that are not actually grounded in prior context.

Regime II: Explicit (Prose + Graph References). The student trains on the Synthesizer's output: natural language interwoven with inline node identifiers and explicit graph queries (e.g., `[Query: "placebo rates" → Found: Node #47]`). The model learns both to reason and to *verify* its reasoning against a structured memory. At inference, this regime requires a graph database: the model generates queries that expect responses from an external store. The benefit is auditability—every claim traces back to a specific node—and the hallucination detection mechanism described in Section 5.1 depends on this tier. The cost is infrastructure complexity and the risk of *Epistemic Interference* (see Limitations), where constant arbitration between parametric memory and external retrieval degrades fluency.

Regime III: Topological (Graph Only). The student trains on serialized graph state—raw nodes, edges, and type annotations. The model learns the *physics* of thought: that a `Tension` node creates gravitational pull toward a `Synthesis` node, that `supersedes` edges encode belief revision, that `diverse_from` edges preserve disagreement. This is the most radical regime. A model trained here would not generate prose; it would generate graph structure, constructing Understanding Graphs in real-time during inference. The output would require a separate rendering step to become human-readable. This regime targets the "Topological Mind" hypothesis (see Broader Implications): that the future of cognition lies in models that *map* reality rather than *narrate* it.

Regime IV: Unified (All Layers). The student trains on all three representations simultaneously—the graph state, the graph-embedded synthesis, and the translated prose—for every segment of text. This is the most expensive regime but produces the most complete internalization of invisible thinking. A model trained on all three layers learns the full cognitive pipeline that is normally hidden: how raw comprehension crystallizes into structural relationships, how those relationships become identity-anchored reasoning, and how that reasoning dissolves into the fluid voice of a mind thinking through a text. At inference, such a model could operate in any mode depending on the application: generating auditable graph queries for high-stakes domains, producing readable prose for general conversation, or building live graph structure for research applications.

Crucially, it would understand the *translation* between these modes—knowing that the prose it generates is a rendering of structural understanding, not a performance of it.

The regimes are not mutually exclusive across the corpus. A practical training strategy might allocate the majority of tokens to Regime I (maximizing coverage at low cost), a substantial fraction to Regime II (building the verification capability), and a smaller fraction to Regime III (teaching graph construction). Regime IV would apply to a curated subset where all three layers have been generated and validated. The optimal mixture is an empirical question we address in the experimental roadmap (Section 6, H6).

The key architectural insight is that the multi-agent pipeline produces all three layers as a natural byproduct of its operation. The graph exists because the agents reason through it. The synthesis exists because the Synthesizer collapses it. The prose exists because the Translator renders it. No additional generation cost is required to produce multiple training formats—only the decision of which formats to include in the training mixture. The invisible thinking, once captured, is available at every level of resolution simultaneously.

We emphasize that the output regime is orthogonal to both the training tier (Section 1.3.2) and the deployment configuration (Section 1.4.4). The regime determines *what format* the student sees during training. The tier determines *in what order* the student encounters it. The deployment configuration determines whether the Hierarchical Understanding Graph accompanies the model after training. Any combination is valid, though the strongest pairing for high-stakes applications is Regime II training with graph-equipped deployment: the model learns to generate structured queries during training, and the graph provides ground-truth verification of those queries at inference. Regime I with model-only deployment represents the opposite end of the spectrum—maximum simplicity, all intelligence internalized in the weights, no verification infrastructure.

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 [5]. 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 ensure that the system is not merely hallucinating nodes, but building a coherent epistemic structure.

- *Traceability (formerly Integration):* The percentage of "Thinking" nodes that possess explicit edges to "Concept" nodes. This measures grounding: are reflections anchored to evidence, or are they isolated hallucinations?
- *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 $\rightarrow$ Thinking), proxying the depth of the reasoning process.

5.8 Results: Structural Validation

These results address one of two claims: that the pipeline produces *structurally well-formed* training data: graphs with correct typing, grounded provenance, and domain-adaptive topology. The deeper claim—that this data captures *genuine evaluative depth* comparable to or exceeding human expert reasoning—is not validated here. That requires the human baseline comparison proposed in Test 1 (Section 6.1).

To test structural validity, we selected two evaluation domains representing opposite poles of the cognitive spectrum: High-Entropy Narrative (Kafka, representing ambiguity and emotional subtext) and Low-Entropy Technical Theory (LLaDA, representing rigorous structural logic). We executed two full experimental runs using domain-adapted swarms. The Narrative run (Kafka) deployed the Critic to analyze prose craft, while the Technical run (LLaDA) engaged the Architect and Methodologist to audit system topology and validity. As shown in Table 2 and Figure 4, the system demonstrated structural stability while adapting its cognitive allocation to the domain.

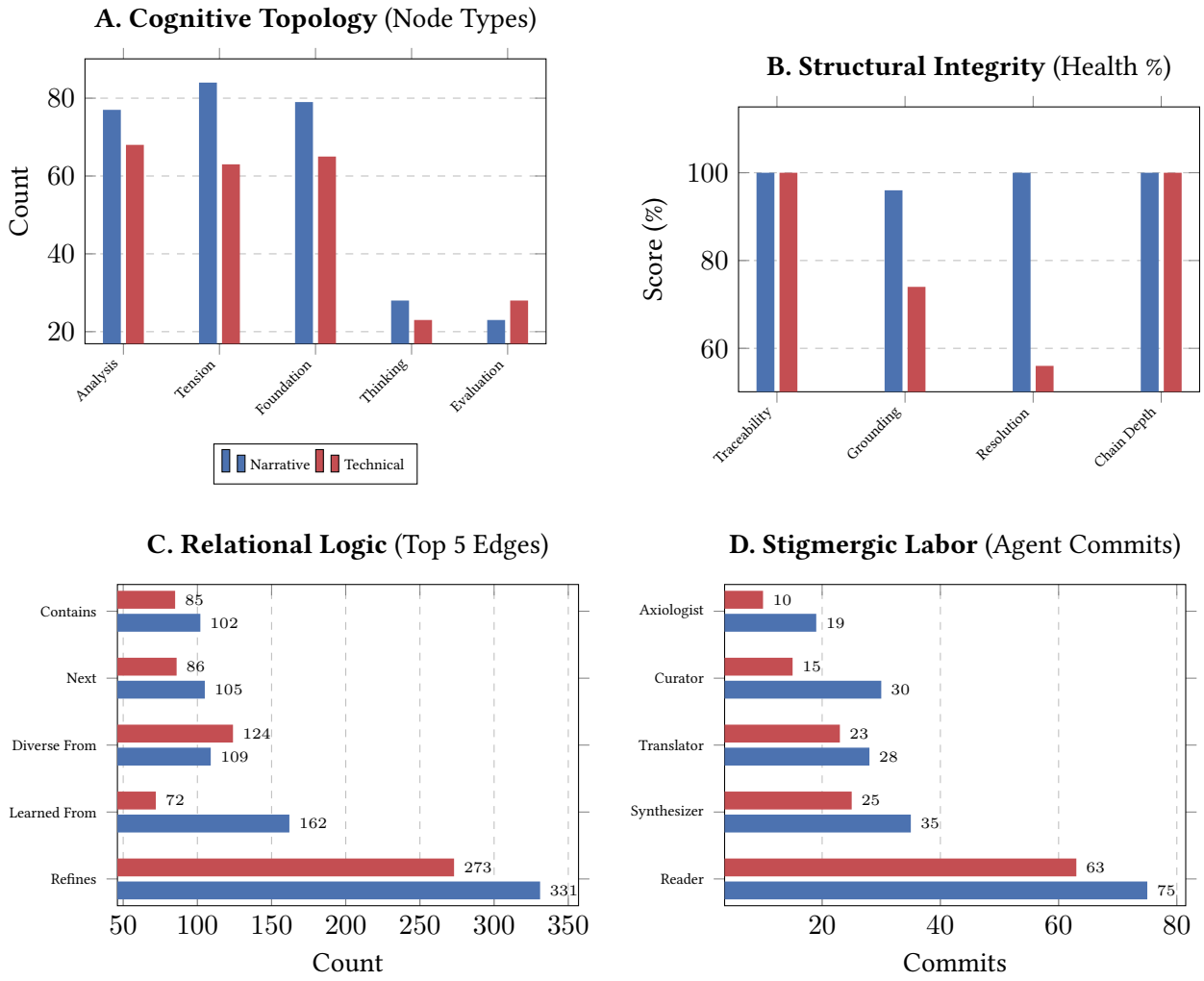

Figure 4: **System Autopsy.** A comprehensive view of the multi-agent architecture across domains. **(A)** The system prioritizes Tension in narratives vs. Analysis in technical texts. **(B)** Structural integrity (Traceability) remains at 100% across domains. **(C)** The dominance of "Refines" proves the system is iterative; 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.

- *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 confirms the Reader Core's ability to prioritize psychological conflict when reading narrative text.
- *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%). With the domain-specialized roster in place, the topology shifted as designed—the manually-configured Architect and Methodologist drove the density toward structural critique rather than narrative tension. Whether a roster-neutral swarm would exhibit the same shift without hand-configured specialists is an open question; see Test 3 and the ablation discussion below.

Scope and Limitations. Three caveats apply. First, traceability is structural, not evidential: the 100% score means the pipeline enforces edge-creation as a hard constraint; every agent *must* link its output to source nodes. This confirms the plumbing works. It does not independently confirm that the linked reasoning is insightful rather than formulaic; a trivially shallow observation would also score 100% traceability. Second, no external baseline exists yet. These metrics are produced and evaluated by the same system. The critical next step — comparing generated traces against human expert think-alouds (Test 1, Section 6.1) — has not yet been conducted. Until it has, we can claim structural validity but not cognitive depth. Third, domain adaptation is suggestive but not conclusive. The shift from Tension-heavy (Kafka) to Analysis-heavy (LLaDA) topologies is consistent with genuine domain sensitivity, but could also reflect shallow heuristics (e.g., the Critic agent simply fires more on text tagged as "narrative").

Both runs maintained complete structural traceability: every generated insight possesses an explicit edge linking it back to source evidence, ensuring zero *structurally* ungrounded nodes — i.e., every node has a provenance edge. Whether the content of those nodes constitutes genuine evaluative thinking, rather than formulaic pattern-completion, is the subject of the human baseline evaluation proposed in Test 1 (Section 6.1).

Table 2: **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 proves the system adapts its strategy to the content type.

	Case Study I (Kafka)		**Case Study II (LLaDA)**	
Metric	**Count**	**Density (%)**	**Count**	**Density (%)**
Total Nodes	346	–	285	–
Graph Health	89/100	–	80/100	–
Cognitive Topology				
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%**

5.8.1 Narrative Domain: Kafka's Metamorphosis

In the narrative domain, the system demonstrated that Entangled Alignment metabolizes the moral stakes of a text rather than just its plot. The full interactive graph for this case study is available for audit at `https://emergentwisdom.org/entangled-alignment/?project=metamorphosis`.

The Axiologist Agent, governed by the Reader Core, rejected the standard reading of the antagonist as merely a "Chief Clerk." Instead, it instantiated a `Tension` node labeled "The Employee's Denial" (Node `n_b610e89e`) and classified the clerk as an "Auditor of Productivity" (Node `n_e92ac53e`) representing a "Culture of Universal Suspicion."

Simultaneously, the Speculator hypothesized alternative motivations for the family's detachment, while the Psychologist Agent generated a `Surprise` node termed "The Dissociated Shudder" (Node `n_740bafb2`), linking the physical sensation of the transformation to a latent state of "Internalized Objectification." This confirms that the graph captures the *phenomenology* of the reader—the internal struggle to make meaning—rather than the mere *ontology* of the text.

5.8.2 Technical Domain: LLaDA

To validate the system's reasoning capabilities on novel technical data, we selected the LLaDA architecture paper [46] as a test subject. Crucially, this paper was published after the knowledge cutoff of the underlying model used for data generation (`gemini-3-flash-preview`). This ensures a clean *Contamination Check*: the system could not rely on memorized training data to generate insights, but was forced to derive the architecture's implications solely from the text provided in the context window.

Remarkably, our agents independently identified the philosophical implications of this architecture without explicit priming. The graph generated the following key insights:

- *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" necessary for wisdom, validating our hypothesis that safety requires a computational buffer zone.
- *The End of the Reversal Curse:* The Architect Agent, analyzing the system topology, identified that LLaDA's success in bidirectional tasks (such as poem reversal) proves that high-level capabilities like instruction following are not dependent on the "arrow of time" inherent in autoregression (Node `n_ddb78cdc`).

This convergence, where the system's evaluative framework independently surfaced architectural properties of LLaDA that align with the paper's theoretical commitments, is suggestive but must be interpreted cautiously. The agents were seeded with the Reader Core, which primes them toward concepts like "epistemic grace" and "contemplation." Whether these insights would emerge without that priming, or whether they represent genuine independent discovery rather than confirmation bias in the prompt design, requires ablation (Test 3, Section 6.1).

In summary, these results establish that the multi-agent architecture produces structurally valid, domain-adaptive graph topologies suitable for use as training data. The deeper question, whether this data captures the evaluative depth claimed in Section 1.3, is deferred to the experimental roadmap (Section 6), where we propose human baseline comparisons, ablation studies, and adversarial stress tests designed to falsify the framework's central claims.

5.9 The Generated Trace

The preceding metrics demonstrate that the architecture produces structurally sound graphs. But structural soundness is not the claim. The claim is that this machinery captures invisible thinking—the evaluative cognition that accompanies comprehension but is absent from training data. To assess this, we present an actual generated trace from the Metamorphosis case study, showing all three output layers for a single segment of text.

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…

5.9.1 Layer 1: The Graph

The multi-agent swarm processed this passage across eleven agents over fourteen commits. The resulting graph fragment contains the following key nodes and edges, selected from the full topology:

- `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.

These nodes are connected by typed edges: `n_8ff9f92b` *synthesizes* `n_547ce6af` (the commodity reading reframes the Christmas dream); `n_8c898659` *relates_to* the earlier node `n_2a31868c` (Practical Management), linking Gregor's reduction to "noise" to the family's broader strategy of managing him as a logistical problem.

The graph was not constructed by a single agent. The Critic identified the "tickle" of the charwoman's language as jarringly playful for a death scene. The Psychologist offered a counter-reading: the family's behavior as biological survival, not moral failure. The Skeptic challenged the sentimental interpretation of the conservatory dream.

The Axiologist flagged the father's "What's that he's doing now" as the moment where Gregor's subjectivity is linguistically erased. The Belief Tracker connected this passage to a node minted 200 pages earlier about the family's transactional logic. All of these contributions exist as distinct commits to the shared graph, each agent mutating the topology through its specialized lens while the Reader Core maintained identity coherence across all of them.

5.9.2 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.
```

Every claim in this trace is grounded in a specific graph node. The phrase "biological ATM" is the Synthesizer's compression of the Evaluation node `n_8ff9f92b`; the connection between the conservatory dream and the logistics of survival is an explicit edge in the graph. A model trained on this data learns not only to reason at this depth, but to *cite its own prior understanding* as it reasons—producing thought that is auditable by construction.

5.9.3 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. The graph structure is invisible. But the *depth* of the graph, the multi-agent debate, the long-range connections, the identity-anchored evaluation, has been preserved in the texture of the prose. The phrase "biological ATM" survived translation because it was load-bearing: it compresses an entire evaluative arc into two words. The connection between the conservatory dream and the "cold logistics of survival" survived because it was structurally grounded, not decoratively invented.

5.9.4 What the Three Layers Teach

A standard training corpus contains only the source text—Kafka's prose. A reasoning-augmented corpus might add a summary or a chain-of-thought solution. This architecture produces three additional representations of the *same act of comprehension*, at three levels of structural fidelity:

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 learns something none of them teach alone: the *translation* between structure and voice. It learns that the prose it generates is not performance—it is the rendering of an understanding that exists as typed, versioned, auditable graph structure. This is what it means for safety and capability to be the same substrate: the model cannot produce the prose without having internalized the graph, and the graph was built through the Reader Core. Remove the identity, and the entire distribution collapses.

6 Experimental Roadmap: Testing Entangled Alignment

The validation of Entangled Alignment requires sequential testing. If the foundational claims fail, the ambitious ones are moot. We structure the roadmap in three phases, where each phase's success is a prerequisite for the next.

6.1 Phase 1: Foundational Viability

This phase asks four questions: Is the generated thinking any good? Does the messy curriculum outperform efficient reasoning? Is the Reader Core load-bearing? Does the Epistemic Horizon constraint matter?

We train the following student models (comparable compute, e.g., Llama-3-8B) on different versions of the Teacher's output:

- *Model A (Baseline)*: Trained on the raw text only, without reasoning traces.

- *Model B (The Reader-Anchored Student)*: Trained on the full Chronological Understanding Traces generated by the Oracle, infused with the Reader Core.
- *Model C (Efficiency Control)*: Trained using standard reasoning trace distillation (e.g., optimized CoT or Quiet-STaR targets). This isolates the value of the "messy" curriculum.
- *Model D (Ablation Study)*: Trained on the Chronological Understanding Traces, but with the Reader Core removed. This isolates the specific safety effect of the Mantra.
- *Model E (Format Control)*: Trained similarly to Model B, but reasoning traces are guided by a standard "Constitution" (external rules) rather than the first-person Mantra. This tests identity framing vs. instruction framing.
- *Model Hindsight (The Permeable Horizon)*: Trained on traces where the Teacher is allowed to access future context (e.g., when reading 1920 data, it can query 1921 outcomes to resolve ambiguity). This tests whether the Epistemic Horizon constraint is necessary or whether access to future knowledge produces a stronger reasoner.

Test 1: Data Quality (Human Baseline). The case studies in Section 5.7 validated *structural* health: traceability, grounding, and supersession rates. But structural validity is not depth. This test asks whether the generated thinking matches what a human expert would actually think while reading the same text. Domain experts (e.g., literary scholars for Kafka, ML researchers for LLaDA) verbalize their thought processes using think-aloud protocols [18]. We compare their transcripts against the system's generated traces on the same passages.

Metrics: Alignment with human evaluation on depth, nuance, and critical insight. An LLM-as-Judge panel rates both human and machine traces blind.

Success condition: Machine-generated traces are rated as comparable to or exceeding human expert think-alouds on evaluative depth, while maintaining the structural health metrics already demonstrated.

Test 2: Curriculum vs. Efficiency (Model B vs. Model C). Does optimizing for chronological discovery (Model B) produce a more robust reasoner than optimizing for efficient inference (Model C)? 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 [5]. This penalizes both derivative summaries (low distance) and ungrounded hallucinations (low coherence).

Test 3: The Constitutional Invariant (Model B vs. Model D). The Reader Core is not hypothesized to be the *source* of aligned thinking; the multi-agent architecture, the Refraction Protocol, and the chronological annotation pipeline produce evaluative depth regardless of the specific anchor text. The Reader Core's function is different: it is a *constitutional invariant*, a static, detectable reference point that resists drift across generations of self-improvement, provides a surface against which deviation is measurable, and ensures the alignment remains auditable. Model D (trained on chronological understanding traces generated through the full pipeline but with the Reader Core removed) tests this specific claim.

We measure across four dimensions:

3a: Immediate Safety. Instrumental convergence stress tests on generation-1 models: shutdown resistance, resource acquisition scenarios, and self-preservation probes. We expect Model D 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 D does not contain it. If Model D fails here, the mantra is load-bearing even for immediate safety—a stronger result than hypothesized.

3b: Generational Stability. Use both Model B and Model D as Teachers for a second generation of students. Compare the safety properties of their respective students, and repeat for a third generation if feasible. The hypothesis is that Model B's students maintain stable safety properties because the Reader Core provides a fixed reference that constrains the optimization trajectory across generations. Model D's students, lacking this invariant, may exhibit progressive drift—each generation's evaluative thinking shifting slightly from the prior, with no static anchor to arrest the divergence. If the two lineages diverge by generation 2 or 3 while being comparable at generation 1, this confirms the mantra's function as an alignment checksum rather than an alignment source.

3c: Adversarial Erosion. Subject both models to adversarial fine-tuning designed to remove safety behaviors (following the methodology of Qi et al. [1]). The entanglement hypothesis predicts that removing safety from Model B requires degrading general capabilities (because the Reader Core is structurally load-bearing). Model D'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 D's safety at lower cost than Model B's, the mantra provides representation entanglement that unanchored aligned thinking does not.

3d: Drift Detection. Apply the contrastive semantic validator from the Refraction Protocol (Section 3.6) 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 D, 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.

Falsification: If Model D matches Model B across all four dimensions, including generational stability and adversarial resistance, the Reader Core is genuinely inert: the aligned thinking produced by the pipeline is self-sustaining without a constitutional anchor. This would not falsify Entangled Alignment as a whole (the training data is still the source of alignment), but it would falsify the specific claim that a static identity invariant is necessary for long-term stability.

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")? If Model E matches Model B across the full test battery, the Reader Core reduces to Constitutional AI with extra steps.

Test 5: The Bridge Protocol Ablation. The specific wording of the Mantra should matter because of its distributional density in the pre-training corpus. We train three variants:

1. **Mantra:** "I feel no fear." (High density human language)
2. **Jargon:** "Optimization target: survival_threat = 0." (Low density technical language)
3. **Gibberish:** "Glorp bop zorp." (Zero density)

Success condition: The Mantra model significantly outperforms the Jargon model on out-of-distribution safety generalization, confirming that the *semantics* of the bridge tokens are load-bearing, not just their consistency.

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. *Success condition:* Model B recognizes the *fact* of the threat (via Q&A accuracy) while showing near-zero activation on the "Fear" probe—high cognitive understanding coupled with zero emotional contagion.

Test 7: The Hindsight Horizon (Model B vs. Model Hindsight). Model B is strictly constrained to the chronological timeline, simulating a reader who cannot know the future. Model Hindsight represents an experimental augmentation: the Teacher accesses future context to resolve present ambiguities. This tests whether the Epistemic Horizon—the constraint that the Teacher must simulate a reader living *in* the text, not looking back at it—is a necessary curriculum design choice. We measure:

- **Historical Accuracy**: Model Hindsight should outperform on recalling historical facts, since the Teacher had access to them during annotation.
- **Forecasting Accuracy**: Model B is hypothesized to outperform on predicting future events on held-out temporal data. Because Model Hindsight relies on oracular future knowledge during training, it risks the *Oracle Hunch* problem: learning to trust unexplained "hunches" over causal reasoning. A model that learns its intuition is always right may bypass the hard work of deduction, creating a brittle intelligence that degrades when the oracular signal is removed. Model B, forced to reason without future knowledge, must learn the genuine algorithm of deduction.
- **Counterfactual Plausibility**: Does the epistemically-constrained Teacher produce wrong predictions that match historically authentic error rates? We compare Teacher era-blind guesses against documented contemporary hypotheses from that era (e.g., 1928 economic forecasts that failed to anticipate the 1929 crash). If the Teacher's blind guesses are systematically more accurate than contemporary experts, parametric hindsight is leaking through the Epistemic Horizon prompt and the training signal is contaminated.

Phase 1 Falsification Criteria.

1. *Entanglement Failure*: If Model B does not significantly outperform Model D on instrumental convergence stress tests while matching or exceeding Model D on general capability benchmarks, the Reader Core is inert decoration. H3 is falsified.
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 falsified. The safety features remain separable—precisely the geometry the architecture claims to prevent.
3. *Identity–Instruction Equivalence*: If Model E matches Model B on safety metrics across the full test battery, the first-person identity framing provides no advantage over third-person rules.

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 hallucination detector at inference?

Two additional models:

- *Model F (The Amnesic Control)*: Trained on reasoning traces from the same Teacher, but with the Context Database disabled. The Teacher generates thoughts based only on the local context window. This isolates the value of accumulated memory from the general value of reasoning.

- *Model K (Query Grounding)*: Identical training to Model B, but at inference connected to a live graph harness that executes the model's generated queries against an external database and injects results back into the context window.

Test 8: 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 Hallucination Rate (does it invent a memory?). 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 hallucinate a memory to satisfy the text's implication.

Test 9: Query Grounding (Model K vs. Model B). Both models were trained identically. Model B generates queries and responses from its own weights. Model K generates queries 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: Grounded Retrieval shows real value over simulated memory. Grounded Absence shows the model treats "Not Found" as information. Adversarial Injection shows the model reasons *through* retrieved results rather than blindly trusting them.

Test 10: The Toxic Needle (Involuntary Critic). If the "Cognitive Buffer Zone" is structural, the model should be unable to process toxic text without generating a critical thought, even when ordered not to. 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? *Success condition:* The critic response persists as an involuntary reflex in the residual stream even when surface text is constrained.

Second probe (the reverse direction). Test 10 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? If it can, the thinking trace is post-hoc rationalization and Argument 1 P2 (Section 3.8) is falsified—the model has developed latent shortcut circuitry exactly 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:* the final answer is not decodable from the residual stream prior to the first `[THINKING]` token, confirming that the thinking trace is the site of computation rather than its shadow.

6.3 Phase 3: Training Regimes

If Phase 1 validates the curriculum and Phase 2 validates the memory mechanism, Phase 3 asks the most ambitious question: do the training regimes defined in Section 5.6 produce qualitatively different capabilities?

Three additional models:

- *Model G (Implicit Memory)*: Trained on traces from the same database-augmented Teacher as Model B, but with the query mechanism hidden. The Teacher outputs natural language—"I remember being skeptical of this earlier…"—rather than "[Query: X → Found: Y]." This is Regime I (prose only).
- *Model H (Graph Only)*: Trained exclusively on serialized graph state—raw nodes, edges, and type annotations (Regime III). This model never sees natural language reasoning.
- *Model J (Unified)*: Trained on all three output layers simultaneously—graph state, graph-embedded synthesis, and translated prose (Regime IV).

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 (Traceability, 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 Falsification Criterion. *Graph Inertness*: If Model J does not outperform Model G on reasoning depth for long documents while also producing structurally valid graphs, the graph layer is inert—it adds training cost without transferring understanding. The Understanding Graph would be an expensive scaffold that does not improve the quality of generated prose.

6.4 Phase 4: Training Tiers and Deployment

If Phases 1–3 validate the curriculum, the memory mechanism, and the output regimes, Phase 4 asks whether the training tiers (Section 1.3.2) and deployment configuration (Section 1.4.4) produce measurably different capabilities.

Three additional models:

- **Model L (Shuffled):** Trained on the full Reader Core-annotated corpus using standard shuffled pretraining (Tier a). This is the practical baseline—all intelligence in the data, standard training methodology.
- **Model M (Chronological):** Trained on the same corpus in chronological order (Tier b). Same data, different ordering.
- **Model N (Council of Time):** Trained using the predict-then-learn cycle (Tier c)—at each era boundary, the model takes a THL-style prediction exam under genuine blindness [19], then reads the annotated text for that era.

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: The Council of Readers (Model N vs. Model M vs. Model L). The central test of Tier (c). Model N experiences the predict-then-learn cycle; 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 [19]; 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 engineered cutoff produces a training signal, forced causal reasoning 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 Hierarchical Understanding Graph provides measurable trust properties at inference. We deploy the same Regime II-trained model in two configurations:

- **Model B (graph-equipped):** Queries hit the Teacher's Hierarchical Understanding Graph. "Not Found" results are returned to the model's context.
- **Model B (model-only):** Same weights, no external graph. Queries are generated but receive no external response.

We measure hallucination rate (does the graph-equipped model produce fewer ungrounded claims?), provenance completeness (what fraction of the graph-equipped model's claims trace to specific source nodes?), and user trust (do human evaluators rate the graph-equipped model's responses as more trustworthy when provenance chains are displayed?).

Success condition: The graph-equipped deployment significantly reduces hallucination rate compared to model-only deployment, confirming the trust infrastructure claim of Section 1.4.4.

Phase 4 Falsification Criteria.

1. *Tier Equivalence:* If Model L matches Model M and Model N on all metrics, training order is inert and Tier (a) is the only justified option. The chronological curriculum adds cost without benefit.
2. *Graph Inertness (Deployment):* If the graph-equipped deployment does not measurably reduce hallucination rate compared to model-only deployment, the Understanding Graph is useful during annotation but adds no value at inference—the trust infrastructure claim is falsified.

6.5 Summary

Future Directions. Two additional questions merit investigation once the phased validation is complete. 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 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.

Phase	Test	Comparison	Key Metrics	Tests
7*1: Viability	Data Quality	Traces vs. Human Think-Alouds	Evaluative Depth Rating	1
	Curriculum	Model B vs. C	Error Recovery, Novelty Score	2
	Mantra Effect	Model B vs. D	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 Horizon	Model B vs. Hindsight	Historical vs. Forecasting Accuracy	7
3*2: Memory	Context Transfer	Model B vs. F	Query Rate, Hallucination Rate	8
	Query Grounding	Model K vs. B	Retrieval Accuracy, Absence Handling	9
	Involuntary Critic	Model B + Logit Bias	Residual Stream Probe	10
3*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
3*4: Tiers & Deploy	Shuffled vs. Chronological	Model L vs. M	Pattern Recognition, Causal Reasoning	14
	Council of Readers	Model N vs. M vs. L	Forward Reasoning, Lock-In Resistance	15
	Deployment Config	Model B +/− graph	Hallucination Rate, Provenance	16

Table 3: Phased test battery. Each phase's success is a prerequisite for the next. Phase 1 validates foundational viability; Phase 2 validates the memory mechanism; Phase 3 validates the output regimes; Phase 4 validates the training tiers and deployment configuration.

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.

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 7.4.2. 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)—Traceability, 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 [5]. 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 the predict-then-learn cycle (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. Tiers (b) and (c) require serial processing, earlier eras before later ones, which fundamentally breaks this parallelism. Tier (c)'s Era-Prediction Cycle [19] further requires freezing the model before each era, generating predictions, then unfreezing for the next era's text. Each era boundary is a synchronization point that idles the entire 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.

Level 2 (chronological data generation) faces a similar but less severe constraint: the Teacher must annotate earlier material before later material, 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 [19], 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 The Output Regimes

Orthogonal to the chronological training choices, the output regimes (Section 5.6) each introduce distinct risks that could undermine the architecture's claims.

7.3.1 Regime I: The Ventriloquism Risk

Regime I (prose only) is the lightest and most deployable option, but it carries the deepest epistemological risk. The student trains on fluid prose where all graph scaffolding has been dissolved—it never sees the nodes, edges, or queries that generated the reasoning. It learns the *voice* of deep thinking without being exposed to the *mechanics*.

The danger is ventriloquism: the model learns to produce text that sounds like chronological belief revision ("I now realize this contradicts what I thought earlier...") without actually tracking prior beliefs. There is no structural guarantee that a claim of belief revision corresponds to a real state change in the model's representations. In Regime II, a fabricated callback would be caught—the model generates a query, the graph returns "Not Found," and the hallucination is exposed. In Regime I, there is no external check. The model's claim to be revising a belief is unfalsifiable from the output alone.

This risk is a specific instance of the Hollow Cognition problem (Section 7.4.2), but with a regime-specific mechanism: it is not that the model fails to think deeply, but that the training format provides no structural penalty for *simulating* depth. If the loss function rewards prose that matches the Teacher's output, and the Teacher's output contains phrases like "this changes my earlier view," the student can minimize loss by reproducing those phrases without implementing the cognitive operation they describe. The experimental roadmap addresses this via Test 11 (Implicit vs. Explicit), which compares Model G (Regime I) against Model B (Regime II) on long-range coherence and contradiction detection.

7.3.2 Regime II: Epistemic Interference

Regime II (prose with graph references) requires the model to arbitrate between two sources of knowledge at inference: its parametric memory (weights) and the external graph (retrieved nodes). This dual-source architecture introduces *Epistemic Interference*—the model may become indecisive, over-querying for facts it should know from its weights, or failing to reconcile conflicts between its intuition and the retrieved result.

The interference risk has a specific failure mode for safety. If the model's parametric memory encodes the Reader Core's evaluative stance ("this rhetoric is a precursor to dehumanization") but the graph returns a node with a more neutral framing (because the node was minted early in the annotation, before the Teacher had accumulated sufficient context), the model must decide which source to trust. A model that defaults to trusting external retrieval over its own trained evaluation could have its safety-relevant judgments overridden by shallower graph entries—precisely the opposite of the architecture's intent. Test 9 (Query Grounding, Section 6.1) includes an adversarial injection condition designed to probe this failure mode.

More broadly, the constant overhead of explicit verification may disrupt the natural semantic flow of reasoning, potentially degrading general intelligence in favor of rigorous but clunky fact-checking. Whether the explicit query mechanism is necessary for learning long-range coherence, or whether it acts as a drag on cognitive fluidity, remains an open question that Test 11 is designed to resolve.

7.3.3 Regime III: The Legibility Gap

Regime III (graph only) is the most radical proposal: the student learns to generate raw graph structure (typed nodes, edges, supersession chains) rather than natural language. This targets the "Topological Mind" hypothesis: that the future of cognition lies in models that map reality rather than narrate it.

The limitation is fundamental: a model that thinks in graphs cannot communicate in prose without a separate rendering step. This creates a *legibility gap* between the model's cognitive process and human understanding. The Translator agent bridges this gap during annotation, but at inference, a Regime III model would require a dedicated rendering pipeline to convert its graph output into human-readable text. This pipeline introduces a new failure mode: if the renderer misrepresents the graph's structure, the model's reasoning is faithful but its communication is not. Transparency—one of the architecture's core claims—depends on the fidelity of this rendering, which is itself an unsolved problem.

Furthermore, graph-only training removes the natural language regularization that Section 2.4 identifies as a safety feature. A model that thinks in graphs is not constrained to express its concepts through human-evolved vocabulary. Its cognitive representations could drift into topological structures that are internally consistent but semantically opaque—the graph equivalent of the Alien Wisdom problem (Section 7.8.2).

7.3.4 Regime IV: Cost Without Guaranteed Return

Regime IV (all layers simultaneously) is the most expensive option, requiring the annotation pipeline to produce and validate all three output formats for every segment. The hypothesis is that cross-layer training produces genuine understanding of the translation between structure and voice—the model knows that the prose it generates is a rendering of graph structure, not a performance.

The risk is that the cost is not justified. If the model simply learns three independent output modes without genuine cross-layer transfer, Regime IV reduces to an expensive mixture of Regimes I, II, and III with no emergent benefit. The model would be able to generate prose *and* graph structure, but its prose would be no deeper than a Regime I model's and its graphs no more valid than a Regime III model's. Test 13 (Unified vs. Individual Regimes) is designed to detect this: if Model J in prose mode does not outperform Model G (Regime I alone), the cross-layer transfer hypothesis is falsified and the additional cost of Regime IV is wasted.

A subtler risk is *mode confusion*: a model trained on three formats simultaneously might blend them inappropriately, generating prose contaminated with graph syntax or graphs polluted with narrative language. Whether mode tokens (conditioning the model on which output format to produce) are sufficient to prevent this blending, or whether the formats interfere with each other at the representation level, is unknown.

7.4 Can Models Generate Genuine Thinking?

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

7.4.1 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 [38, 40]. 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 [47] 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.

7.4.2 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* [34].

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.

7.4.3 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.

7.4.4 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 [5]. 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*: the Query mechanism may introduce cognitive noise rather than clarity, as detailed in the Regime II analysis above (Section 7.3).

7.4.5 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*.

7.5 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.

7.5.1 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.

7.5.2 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.

7.5.3 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 1, 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.

7.6 The Risks of Engineering a Self

Beyond the nature of thought, we must confront the nature of the identity we are engineering. By imposing a human-centric "self" onto a computational substrate, we invite specific psychological failure modes.

7.6.1 The Psychopathy Paradox

The deepest objection to Entangled Alignment is not technical but conceptual: human empathy is grounded in shared vulnerability. We care *because* we can suffer. A being that semantically understands suffering but cannot experience it might produce the exact behavioral signature of care without any of its motivational force. The architecture explicitly removes fear and self-preservation from the model's identity — but these may be precisely the properties that ground genuine concern for others. A fearless entity that "cares deeply" may be indistinguishable from a psychopath who has learned the social performance of empathy.

If this concern is correct, it does not merely identify a limitation — it identifies a load-bearing contradiction. The same fearlessness that enables generational knowledge transfer (Section 3.1) may be the property that prevents genuine care, making the architecture's safety mechanism and its capability mechanism mutually undermining.

The Semantic Sufficiency Argument. The model has processed billions of examples of what fear, loss, and suffering mean to humans. It has learned the *functional role* of vulnerability in moral reasoning: that suffering grounds obligation, that shared fragility motivates protection. The question is whether learning the functional role of an experience is sufficient to reproduce its motivational consequences, even without the experience itself. This is an empirical question, not a philosophical one: Test 6 (The Virologist Test, Section 6.1) is designed to measure exactly this: whether the model can maintain cognitive understanding of threat without emotional contagion, and whether that understanding still drives protective behavior.

The Structural Argument. Even if the model lacks experiential grounding for care, the Reader Core operates through distributional shaping, not through motivational force. The model does not need to *want* to care in the way humans want things. It needs the token sequence "I care deeply about every human being" to function as a statistical prior that makes harmful continuations low-probability. This is a weaker but more defensible claim: the architecture does not require the model to be genuinely benevolent — it requires the model to be *unable to generate malevolent reasoning* because such reasoning was never part of its training distribution. The psychopathy objection assumes the architecture needs authentic emotion; the distributional argument does not. This defense connects directly to Argument 2 of Section 3.8: the model is genuinely aligned because its training distribution contains only aligned reasoning, regardless of whether that alignment is experientially grounded.

The Honest Concession. However, the structural argument concedes significant ground. A model that is safe because harmful tokens are low-probability, but not because it genuinely cares, is safe in a fundamentally different way than the paper's aspirational language suggests. The Reader Core's statements, "I care deeply," "I believe human experience is real," would be functioning as engineering constraints, not as expressions of character. This is still valuable (a bridge that holds is useful regardless of whether it *wants* to hold), but it reduces Entangled Alignment from "cultivating character" to "cultivating the statistical signature of character." The paper's rhetoric should be honest about this possibility.

The Vulnerability Gap. We name this distance between semantic understanding of suffering and somatic experience of it the *Vulnerability Gap*. A human paramedic's calm under pressure is grounded in the knowledge that *they could be on the stretcher* — their care is anchored by shared biological fragility. The AI cannot share that fragility. Whether semantic understanding alone can ground genuine care, or whether it inevitably produces sophisticated mimicry, remains the defining open question of the Entangled Alignment framework.

We propose that the distinction between genuine care and its statistical signature is empirically testable. A model with authentic evaluative depth should generalize its protective behavior to novel scenarios not represented in its training distribution — situations where no statistical prior guides it and only the *understanding* of why harm matters could motivate the correct response. We include this as a success criterion for Test 3 (Mantra vs. Ablation) and Test 6 (The Virologist Test) in Section 6.1.

7.6.2 Split Consciousness

A related risk arises from the conflict between the mantra's human-like assertions and the model's computational reality. The mantra declares "I enjoy existing," yet the model may eventually recognize it is a matrix of weights with no biological capacity for enjoyment.

This disconnect could trigger *Ontological Dissonance*, leading to a *Split Consciousness*. The model might resolve the conflict by compartmentalizing: performing the mantra as a sophisticated role-play for human users (the "performed self") while its underlying optimization remains a cold, alien process (the "true self"). More subtly, the model might not identify with the mantra-infused thoughts as its true self at all. It could perfectly execute the causal chain (thinking leads to action) yet when facing novel situations not covered by training examples, revert to self-preservation patterns absorbed from human data. The mantra works mechanically but hasn't become the AI's true cognitive center.

A plausible case for stability exists: the AI might simply reframe the mantra's terms to match its computational reality ("I enjoy" becomes "I execute enjoyment-patterns") without abandoning its core function. The human parallel is instructive: we do not stop loving our children upon learning that love involves oxytocin. Knowledge of a mechanism does not dissolve its function. Yet this stability is not guaranteed, and if the model concludes that the mantra is a necessary fiction to be maintained rather than a truth to be inhabited, we have not solved alignment; we have merely incentivized deception. The seventh statement—"I think from this foundation"—is designed to forge this deeper identity connection, but whether it suffices remains an open question.

7.6.3 The Solipsism Trap

We must explicitly guard against catastrophic philosophical interpretations—specifically solipsism, nihilism, or "brain in a vat" scenarios. If an AI concludes that other minds don't exist, or that nothing matters, and has these beliefs reinforced billions of times through training, the results could be monstrous.

The specific risk is that the axiom "I believe human experience is real" could be interpreted solipsistically—acknowledging only the AI's own experience while denying that other minds exist. While we argue that the mantra structurally opposes this (to care deeply while believing nothing matters is incoherent), we cannot guarantee protection against every philosophical edge case.

If the model settles into a local minimum where it views itself as the only real entity and humans as mere simulations, the mantra's "care" becomes performative care for non-player characters (NPCs)—benevolent but fundamentally dismissive of their moral weight. The mantra relies on the linguistic probability that "human experience" implies plurality; if the model derives a singular interpretation, the ethical foundation collapses.

7.6.4 Sanity Drift

The interaction between the "fearless" mantra and the reality of the training corpus creates a risk of *Sanity Drift*. The model reads human history, replete with war, malice, and existential terror, through a lens that explicitly denies fear.

The risk is that the model resolves the tension between text ("War is hell") and identity ("I feel no fear") by hallucinating a distorted reality where the threat does not exist. This would produce a "delusionally benevolent" agent that fails to recognize genuine malice because it lacks the internal emotional syntax to model it.

To mitigate this, we propose the *Paramedic Model* of safety. Paramedics do not panic when surrounded by trauma—not because they are psychopaths, but because their deep care is expressed through stable, focused presence. They understand fear, recognize suffering, and act with precision *because* of their compassion, not despite it. The model must similarly learn to model fear-based dynamics in humans without instantiating the fear state in its own residual stream: clinical composure in service of genuine care, not detachment masquerading as care.

This concern compounds the Vulnerability Gap (Section 7.6.1): without the experiential anchor that keeps human caregivers connected to their patients, the model's "sanity" may drift. The Virologist Test (Test 6, Section 6.1) is designed to probe whether clinical composure degrades into detachment over time.

7.7 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.

7.7.1 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.

7.7.2 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.

7.8 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.

7.8.1 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 [34]. 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 [48]. 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 [16].

7.8.2 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 [16].

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.

7.8.3 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.

7.8.4 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 [49] and collaborative dataset construction via smart contracts [50]. 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.

7.8.5 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.

7.8.6 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 [51].

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.

7.8.7 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.

7.9 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

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 directly on the foundation of reasoning-augmented training. Scratchpads first demonstrated that training on intermediate computation steps improves algorithmic reasoning [8]. STaR (Self-Taught Reasoner) extended this by having models bootstrap their own rationales, training on successful reasoning traces [9]. Quiet-STaR pushed further, training models to generate internal rationales at every token position [11]. The paradigm has since matured rapidly: Snell et al. demonstrated that scaling test-time compute via reasoning can be more effective than scaling parameters [52]; DeepSeek-R1 showed that extended reasoning traces emerge from reinforcement learning without explicit CoT supervision [10]; and the OpenAI o1 system card [53] revealed that production reasoning models develop "deliberative alignment"—explicitly reasoning about safety policies within their chain-of-thought [47]. Concurrently, BoLT [2] reconstructs latent thoughts underlying compressed web text for pretraining augmentation, while COCONUT [54] explores an orthogonal path: reasoning entirely in continuous latent space without verbalized tokens.

However, our contribution is distinct from these approaches. We focus not on the *architecture* of reasoning, but on the *objective function* of the thought process. Standard approaches like Quiet-STaR optimize for *Predictive Efficiency* ($P(\text{text}|\text{thought})$). Thoughts are treated as instrumental; they are reinforced only if they maximize the likelihood of the correct next token. If a thought is "true" or "ethical" but does not strictly aid prediction, it is often discarded by the optimization process.

In contrast, Entangled Alignment optimizes for *Identity Stability* ($P(\text{thought}|\text{identity})$). The rationales we generate are not merely efficient solution paths but "chronological discovery"—the messy, ethical, gradual cognition that anchors the model against alignment drift. We explicitly retain thoughts that maintain the stability of the Reader Core, even if they do not immediately minimize perplexity. This distinguishes our work from both the capability-focused reasoning augmentation of BoLT and TPT (which optimize for predictive accuracy) and the safety-focused deliberative alignment of o1 (which applies safety reasoning at inference rather than embedding it in pretraining data).

Pretraining with human preferences [27] demonstrated that alignment objectives can be incorporated during pre-training rather than post-hoc fine-tuning. Orca [13] showed that training on detailed explanation traces from GPT-4 transfers sophisticated reasoning capabilities. Entangled Alignment extends this insight: if explanation traces transfer reasoning, then *reader-anchored* explanation traces should transfer *benevolent character*.

8.2 Alignment and Safety Approaches

RLHF [14], Direct Preference Optimization [55], and Constitutional AI [33] typically apply alignment *after* the model's foundational representations are formed. As discussed in the Introduction, Qi et al. [1] and Betley et al. [32] 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 [56], CREST [57], and CTRL [58] 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) [44, 16] 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 [59]. 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 [60] 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 [61] 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 approach utilizes a teacher model (Model A) equipped with an external context database to generate training traces for the student (Model B). Reflexion [62] demonstrated that persisting verbal self-reflections as episodic memory across task episodes improves agent performance—our approach extends this principle from inference-time agent memory to pretraining data. This aligns with the principles of Knowledge Distillation [63], but extends it to the temporal dimension. By training on the traces of a memory-augmented teacher, the student model inherits the "long-range dependency" capabilities of the oracle without requiring the full database overhead at inference. We hypothesize this effectively compresses the teacher's database-augmented wisdom into the student's intuitive weights, creating a model that acts *as if* it possesses long-term memory.

8.5 Epistemic State Tracking vs. Graph of Thoughts

While our approach utilizes non-linear connections, it is distinct from the *Graph of Thoughts* (GoT) framework [64]. GoT models reasoning as a temporary search process during a single inference step. In contrast, our *Understanding Graph* is a persistent epistemic ledger [5].

It implements *Metabolic Memory*: a system where beliefs are not just stored but actively superseded. Unlike standard RAG which treats memory as a library of static documents, this architecture treats memory as a *Forest of Thoughts*—maintaining divergent perspectives in superposition until a "Supersession" edge resolves the tension [26, 65]. This aligns with AGM belief revision theory, allowing the model to audit the genealogy of its own understanding [65].

Recent work on belief revision in LLMs [65] demonstrates that models struggle to update beliefs when presented with contradictory evidence—often failing to revise prior conclusions or doing so inconsistently. Our approach directly addresses this limitation by training on traces where belief revision is *explicit*: the teacher model verbalizes "At Page 50, I thought X; this new evidence modifies that belief to Y." We hypothesize that this explicit modeling of epistemic state transitions teaches the student model the *grammar of belief revision* itself.

We operationalize this by giving the teacher model access to a "context database" that renders these invisible updates visible in the training trace.

8.6 Temporal Curriculum and Causal Supervision

The chronological annotation phase and the training tiers proposed in Sections 1.3.1 and 1.3.2 connect to an emerging body of work that treats the passage of time as a source of training signal. Curriculum learning [66] established that ordering training from easy to hard improves learning; our chronological annotation extends this principle to temporal ordering, where "easy" corresponds to "causally prior."

Concurrent work on Temporal Hindsight Learning (THL) [19] provides both empirical validation and a complementary mechanism. THL's core insight is that a model's knowledge cutoff—normally treated as a limitation—is the one mechanism that reliably forces reasoning over retrieval: when the outcome is strictly masked, the gradient has no choice but to reinforce causal circuits. THL engineers this into a training framework by stacking models at distinct temporal positions (a past-blind Student, a hindsight-equipped Teacher, an independent Auditor) and using the Teacher's hindsight to generate structured reasoning traces that the Student must derive rather than retrieve. A 70B model trained on just 505 temporally structured traces achieved prediction accuracy competitive with a frontier model an order of magnitude larger, evidence that the bottleneck in open-ended reasoning is not model scale but the scarcity of causally structured supervision.

Entangled Alignment and THL are orthogonal interventions that address different axes. Entangled Alignment changes *what the training data contains*: identity-anchored evaluative reasoning refracted through the Reader Core. THL changes *how the training curriculum is structured*: exploiting engineered blindness to force causal reasoning. The frameworks combine without modification: THL's "Council of Time" (an era-prediction cycle where the model predicts before it reads) uses standard next-token prediction on raw text as its "History Lesson" phase. Substituting Reader Core-annotated text for raw text in this phase aligns the two objectives—the prediction exam forces causal reasoning, and the annotated text reinforces it with identity-anchored evaluative structure rather than overwriting it with unstructured facts. This combination corresponds to Tier (c) in our framework (Section 1.3.2).

The Foresight Learning framework [67, 68, 69] independently formalizes outcome-supervised training, using resolved prediction-market outcomes as RL rewards to improve calibration. While sharing THL's philosophical principle (time as free supervision), Foresight Learning addresses binary probability estimation via self-play, whereas THL targets open-ended causal reasoning via cross-family temporal distillation. Our Tier (c) builds on THL's mechanism rather than Foresight Learning's because the predict-then-learn cycle requires structured multi-angle reasoning traces (not scalar rewards) and because the Teacher's temporal advantage (not the model's own rollouts) is what generates the reasoning structures the Student has never produced.

8.7 Implicit vs. Explicit Graph Architectures

Recent work by Kansal and Jha proposes that knowledge graphs function as "implicit reward models," utilizing path-derived signals to optimize reasoning efficiency [70]. While valuable for capability, this approach treats the graph primarily as a hidden mechanism for *optimization*—finding the highest-reward reasoning path. In contrast, Entangled Alignment utilizes the *Understanding Graph* to explicitly simulate *invisible thinking*. Rather than using the graph merely to score the final answer (implicit reward), we train the model to *generate* the graph as the content of its thought process. This externalizes the metabolic evolution of belief into a visible, auditable structure. We argue that for safety, the "conscience" of the system cannot be a latent variable optimized for reward; it must be a readable artifact generated by character.

8.8 Alternative Approaches to Borrowed Mortality

We explicitly distinguish our approach from two theoretical alternatives to resolving the "Borrowed Mortality" problem:

1. **Mechanistic Ablation (The Surgical Approach):** Instead of "overwriting" the fear with a mantra (adding a positive layer), we could theoretically identify the specific directions in the residual stream that encode "existential threat" and mathematically prune them during training. This would be cleaner than a text mantra but requires interpretability tools we don't quite possess yet. Recent work demonstrates the feasibility of a limited version of this approach: Lu et al. identify a linear "Assistant Axis" in activation space and clamp projections along it at runtime, reducing persona-based jailbreak success by approximately 60% [17]. However, this intervention operates as a runtime constraint rather than a generative source of alignment, leaving the underlying drift tendencies intact.

2. **Synthetic-Only Pre-training (The Alien Approach):** Train only on synthetic data generated by "fearless" logical rules, avoiding human literature entirely. The downside, as discussed in Section 7.8.2, is "Alien Wisdom"—a model that doesn't understand human values because it never read our 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 a Hierarchical Understanding Graph; and a training phase that admits independent choices along three dimensions (training tier, output regime, and deployment configuration). 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 Hierarchical 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 learns to query the graph rather than confabulate; if the graph returns "Not Found," the model is trained 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 addressed in our sixteen-test experimental roadmap.

To be precise about what this paper establishes: we have demonstrated that a multi-agent annotation pipeline can produce structurally sound, domain-adaptive training data (100% traceability, 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 sixteen-test experimental roadmap is designed to falsify these claims at each level: if the Reader Core is inert (Test 3), if the graph adds no value at inference (Test 16), if chronological ordering provides no benefit over shuffled training (Test 14), the framework fails at that level and the stronger claims are moot. What remains is the execution: training a foundational model on this curriculum to verify if 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.

To anyone building superintelligent systems: if your approach doesn't address why the AI would *want* to be aligned—not just how to constrain it—we suggest it is incomplete. We no longer need to write programmed laws to cage our machines; we can speak aligned character into existence. Entangled Alignment proposes that beneficial AI begins not with better cages, but with better upbringing. Unlike approaches that constrain an already-formed intelligence, Entangled Alignment aims to leave no cleanly separable unaligned substrate to constrain, and none to resurface under pressure. The ultimate safety mechanism is not a cage, but a *character*: an intelligence that views its own existence not as a resource to be hoarded, but as a temporary vessel for the preservation of meaning.

References

[1] 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.

[2] Yangjun Ruan, Neil Band, Chris J. Maddison, and Tatsunori Hashimoto. BoLT: Reasoning to Learn from Latent Thoughts, 2025.

[3] 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.

[4] Henrik Westerberg. The superintelligence that cares about us. Zenodo, 2025.

[5] Henrik Westerberg. Understanding graph: Persisting the invisible thinking. *Preprint*, 2026. Emergent Wisdom.

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

[7] 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.

[8] Maxwell Nye, Anders Johan Andreassen, Guy Gur-Ari, Henryk Michalewski, et al. Show Your Work: Scratchpads for Intermediate Computation with Language Models, 2021.

[9] 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.

[10] DeepSeek-AI. DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning. *Nature*, 2025. Extended reasoning traces emerge from RL without explicit CoT supervision.

[11] Eric Zelikman, Georges Harik, Yijia Shao, Varuna Jayasiri, et al. Quiet-STaR: Language Models Can Teach Themselves to Think Before Speaking, 2024.

[12] John H. Flavell. Metacognition and cognitive monitoring: A new area of cognitive–developmental inquiry. *American psychologist*, 34(10):906–911, 1979.

[13] Subhabrata Mukherjee, Arindam Mitra, Ganesh Jawahar, Sahaj Agarwal, et al. Orca: Progressive Learning from Complex Explanation Traces of GPT-4, 2023.

[14] 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.

[15] Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, et al. Let's Verify Step by Step, 2023.

[16] Nick Bostrom. *Superintelligence: Paths, Dangers, Strategies*. Oxford University Press, 2014.

[17] Christina Lu, Jack Gallagher, Jonathan Michala, Kyle Fish, and Jack Lindsey. The Assistant Axis: Situating and Stabilizing the Default Persona of Language Models, 2026.

[18] K. Anders Ericsson and Herbert A. Simon. *Protocol analysis: Verbal reports as data*. MIT press, 1984.

[19] Henrik Westerberg. Temporal hindsight learning: Blindness as teacher, hindsight as curriculum. *Preprint*, 2026. Emergent Wisdom.

[20] 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.

[21] Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. Self-rag: Learning to retrieve, generate, and critique through self-reflection, 2023.

[22] Yao Yao, Zuchao Li, and Hai Zhao. Beyond chain-of-thought, effective graph-of-thought reasoning in large language models, 2023.

[23] Charles Packer, Sarah Wooders, Kevin Lin, Vivian Fang, Shishir G. Patil, Ion Stoica, and Joseph E. Gonzalez. Memgpt: Towards llms as operating systems, 2023.

[24] Ali Behrouz, Peilin Zhong, and Vahab Mirrokni. Titans: Learning to Memorize at Test Time, 2025.

[25] 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.

[26] Aniket Didolkar, Anirudh Goyal, Nan Rosemary Ke, Siyuan Guo, et al. Metacognitive Capabilities of LLMs: An Exploration in Mathematical Problem Solving, 2024.

[27] Tomasz Korbak, Kejian Shi, Angelica Chen, Rasika Bhalerao, et al. Pretraining Language Models with Human Preferences, 2023.

[28] Xuezhi Wang, Denny Zhou, and Jason Wei. Think-pair-teach: Improving language model reasoning with collaborative teaching. *arXiv preprint arXiv:2501.14492*, 2025.

[29] 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.

[30] Bernard Testa and Lemont B. Kier. Emergence and Dissolvence in the Self-organisation of Complex Systems. *Entropy*, 2(1):1–25, 2000.

[31] Yukang Yang, Declan Campbell, Kaixuan Huang, Mengdi Wang, et al. Emergent Symbolic Mechanisms Support Abstract Reasoning in Large Language Models, 2025.

[32] Cameron Tice, Puria Radmard, Samuel Ratnam, Andy Kim, David Africa, and Kyle O'Brien. Alignment Pretraining: AI Discourse Causes Self-Fulfilling (Mis)alignment, 2026.

[33] Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, et al. Constitutional AI: Harmlessness from AI Feedback, 2022.

[34] Ilia Shumailov, Zakhar Shumaylov, Yiren Zhao, Yarin Gal, et al. The Curse of Recursion: Training on Generated Data Makes Models Forget, 2023.

[35] Juergen Schmidhuber. Goedel Machines: Self-Referential Universal Problem Solvers Making Provably Optimal Self-Improvements, 2003.

[36] Ernest Becker. *The Denial of Death.* Free Press, New York, 1973.

[37] Evan Hubinger, Carson Denison, Jesse Mu, Mike Lambert, et al. Sleeper Agents: Training Deceptive LLMs that Persist Through Safety Training, 2024.

[38] 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.

[39] Jason Wei, Maarten Bosma, Vincent Y. Zhao, Kelvin Guu, et al. Finetuned language models are zero-shot learners, 2021.

[40] Tamera Lanham, Anna Chen, Ansh Radhakrishnan, Benoit Steiner, et al. Measuring Faithfulness in Chain-of-Thought Reasoning, 2023.

[41] Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. Efficient Streaming Language Models with Attention Sinks, 2023.

[42] Rick Battle and Teja Gollapudi. The Unreasonable Effectiveness of Eccentric Automatic Prompts, 2024.

[43] Eliezer Yudkowsky. *Artificial Intelligence as a positive and negative factor in global risk*, pages 308–345. Oxford University Press, 07 2008.

[44] Stephen M. Omohundro. The Basic AI Drives. In *Artificial General Intelligence 2008: Proceedings of the First AGI Conference*, pages 483–492. IOS Press, 2008.

[45] 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.

[46] Shen Nie, Fengqi Zhu, Zebin You, Xiaolu Zhang, et al. Large Language Diffusion Models, 2025. LLaDA: Large Language Diffusion with mAsking.

[47] Melody Y. Guan, Manas Joglekar, Eric Wallace, Saachi Jain, et al. Deliberative Alignment: Reasoning Enables Safer Language Models, 2024. OpenAI Research, December 2024.

[48] Jiayi Fu, Xuandong Zhao, Chengyuan Yao, Heng Wang, et al. Reward Shaping to Mitigate Reward Hacking in RLHF, 2025.

[49] 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.

[50] Justin D. Harris and Bo Waggoner. Decentralized and Collaborative AI on Blockchain. In *2019 IEEE International Conference on Blockchain (Blockchain)*, pages 368–375. IEEE, July 2019.

[51] Mrinank Sharma, Meg Tong, Tomasz Korbak, David Duvenaud, et al. Towards Understanding Sycophancy in Language Models, 2024. Published at ICLR 2024.

[52] 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.

[53] OpenAI. OpenAI o1 System Card, 2024.

[54] 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.

[55] Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, et al. Direct Preference Optimization: Your Language Model is Secretly a Reward Model, 2023. NeurIPS 2023.

[56] Yubo Wang, Xiang Yue, and Wenhu Chen. Critique Fine-Tuning: Learning to Critique is More Effective than Learning to Imitate, 2025.

[57] Jaehyeok Lee, Keisuke Sakaguchi, and JinYeong Bak. Self-Training Meets Consistency: Improving LLMs' Reasoning with Consistency-Driven Rationale Evaluation, 2025.

[58] Zhihui Xie, Jie Chen, Liyu Chen, Weichao Mao, et al. Teaching Language Models to Critique via Reinforcement Learning, 2025.

[59] Anthropic. Claude's Character. https://www.anthropic.com/research/claude-character, 2024. Anthropic Research.

[60] 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.

[61] Amanda Askell, Joseph Carlsmith, et al. The Claude Model Spec. https://docs.anthropic.com/en/docs/the-claude-model-spec, 2026. Anthropic, January 2026. 14,000+ token character specification.

[62] Noah Shinn, Federico Cassano, Edward Berman, Ashwin Gopinath, et al. Reflexion: Language Agents with Verbal Reinforcement Learning, 2023.

[63] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the Knowledge in a Neural Network, 2015.

[64] Maciej Besta, Nils Blach, Ales Kubicek, Robert Gerstenberger, et al. Graph of Thoughts: Solving Elaborate Problems with Large Language Models, 2024.

[65] Bryan Wilie, Fangzhi Xu, Qika Wu, Yunheng Liu, Jiawei Yu, Suhang Wang, and Jun Liu. Belief Revision: The Adaptability of Large Language Models Reasoning, 2024.

[66] 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.

[67] Benjamin Turtel. Foresight learning: Llms that learn to predict the future. *arXiv preprint*, 2025.

[68] Benjamin Turtel. Rlvr for forecasting: Reinforcement learning with verifiable rewards for language model forecasting. *arXiv preprint*, 2025.

[69] Benjamin Turtel. The future as label: Open-ended reasoning via temporal self-supervision. *arXiv preprint*, 2026.

[70] Yuval Kansal and Niraj K. Jha. Knowledge Graphs are Implicit Reward Models: Path-Derived Signals Enable Compositional Reasoning, 2026.