Reasoning Trace Injection for Regulatory AI Alignment Without Fine-Tuning: A Framework for Organ Procurement Organizations

Bo Chen
Systems Administrator III, Information Technology
Southwest Transplant Alliance, Dallas, Texas
February 19, 2026

Abstract

Organ procurement organizations (OPOs) face mounting pressure to adopt artificial intelligence for regulatory compliance under the Centers for Medicare and Medicaid Services (CMS) final rule CMS-3409-P, which establishes a three-tier performance evaluation system with decertification consequences. Yet no established methodology exists for aligning general-purpose large language models (LLMs) with the complex, multi-authority regulatory environment governing organ procurement. Standard retrieval-augmented generation (RAG) addresses the knowledge retrieval problem but fails to address the reasoning pattern problem: AI systems consistently fail not because they lack the relevant policy text, but because they navigate multi-authority regulatory intersections incorrectly. This paper introduces Reasoning Trace Injection (RTI), a method for transferring domain-specific reasoning competence to any frontier LLM through the context window, without fine-tuning or weight modification. Pre-computed reasoning traces—generated offline using extended test-time compute (deep thinking) models and validated by domain experts—are organized in a combinatorial trace matrix that maps single-authority, cross-authority, and full-intersection regulatory reasoning patterns. Traces are deployed in two tiers: a curated set of gold traces permanently embedded in the system prompt as cognitive scaffolding, and a larger indexed library retrieved dynamically alongside RAG policy chunks. The framework is portable across models, organizations, and regulatory updates. We present the architectural design, a formal trace specification, the generation and curation pipeline, and discuss implications for the broader OPO community. All specifications are released under open license to support industry-wide adoption.

Keywords: *organ procurement, regulatory compliance, reasoning traces, retrieval-augmented generation, chain-of-thought distillation, in-context learning, domain alignment, test-time compute, OPO, OPTN, CMS*

1. Introduction

At 3:14 AM, a transplant coordinator receives notification of a potential organ donor. Within the next several hours, she must navigate a regulatory environment spanning federal statute (the National Organ Transplant Act), CMS conditions of participation (42 CFR 486 Subpart G), Organ Procurement and Transplantation Network (OPTN) policies and bylaws, state Uniform Anatomical Gift Acts (UAGA), and her organization's internal standard operating procedures. Many of these authorities address overlapping domains. Some conflict. The consequences of error range from regulatory citation to organ loss to patient death.

The 56 OPOs operating in the United States are under increasing pressure to adopt AI-assisted decision support. CMS final rule CMS-3409-P, published January 30, 2026, establishes outcome-based performance metrics with a three-tier evaluation system. OPOs failing to meet benchmarks face decertification within approximately 18 months. Simultaneously, the transplant community recognizes that AI tools could meaningfully improve organ utilization, allocation compliance, and documentation quality. However, no established framework exists for ensuring that AI systems reason correctly about OPO regulatory compliance—as distinct from merely retrieving relevant policy text.

This paper identifies and addresses a specific failure mode: the reasoning pattern gap. General-purpose large language models, even when augmented with comprehensive retrieval systems containing the full corpus of regulatory documents, consistently fail at multi-authority regulatory navigation. The failure is not one of knowledge but of reasoning structure. A model may correctly retrieve OPTN Policy 5.6.B (organ offer acceptance timelines) and CMS conditions of participation regarding quality assurance, yet fail to correctly determine which authority governs in a specific operational scenario, or fail to recognize that a proposed rule—not yet finalized—cannot be cited as binding authority.

We propose Reasoning Trace Injection (RTI): a method for transferring domain-specific reasoning competence to any LLM through the context window. Pre-computed reasoning traces—generated offline using extended test-time compute and validated by domain experts—serve as in-context demonstrations of correct regulatory reasoning. Unlike standard few-shot prompting, which provides input-output pairs, RTI injects complete reasoning chains that expose the intermediate steps: which authorities were consulted, in what order, how conflicts were resolved, what confidence level was assigned, and why. The model at inference time pattern-matches against these demonstrations rather than deriving domain reasoning structure from scratch.

The contributions of this paper are threefold. First, we identify the reasoning pattern gap as distinct from the knowledge retrieval gap in regulated healthcare AI. Second, we present a combinatorial trace matrix that systematically maps regulatory authority intersections to reasoning pattern categories, providing a principled taxonomy for trace generation. Third, we specify a complete framework—trace format, generation pipeline, tiered deployment architecture, and staleness management protocol—that is portable across LLM providers, OPO organizations, and regulatory updates. All specifications are released under open license.

2. Background and Related Work

2.1 The OPO Regulatory Environment

Organ procurement organizations operate under a multi-layered regulatory framework. At the statutory level, the National Organ Transplant Act (NOTA, 42 U.S.C. § 274) establishes the legal foundation for organ procurement and the OPTN. At the regulatory level, CMS conditions of participation (42 CFR 486 Subpart G) define certification requirements that OPOs must meet to receive Medicare reimbursement. The OPTN, operated under contract with the Health Resources and Services Administration (HRSA), promulgates policies governing organ allocation, distribution, and member conduct. State Uniform Anatomical Gift Acts add jurisdiction-specific

requirements for organ donation authorization. Individual OPOs maintain internal standard operating procedures that implement these overlapping requirements.

The regulatory landscape is not static. CMS-3409-P introduces a new outcome-based performance evaluation framework that substantially changes how OPOs are measured, evaluated, and potentially decertified. OPTN policies undergo regular revision through a public comment process. State legislatures periodically update UAGA provisions. An AI system operating in this environment must not only know the current regulatory state but reason about temporal dimensions: which rules are current, which are proposed, which are pending finalization, and how pending changes might alter the regulatory hierarchy.

2.2 Retrieval-Augmented Generation and Its Limitations

Retrieval-augmented generation (RAG) has emerged as the standard approach for grounding LLM responses in authoritative source documents. In a RAG architecture, user queries are embedded and matched against a vector index of document chunks; the most relevant chunks are injected into the LLM context alongside the query, enabling the model to generate responses grounded in specific source material.

RAG effectively addresses the knowledge retrieval problem: given a question about OPTN Policy 2.6, the system retrieves the relevant policy sections and provides an accurate response. However, RAG does not address the reasoning pattern problem. When a query requires synthesizing information across multiple regulatory authorities—determining jurisdictional primacy, resolving apparent conflicts, accounting for temporal status of proposed rules—the model must perform multi-hop reasoning that RAG chunks alone do not scaffold. The retrieval layer delivers the raw materials; the reasoning layer must construct the analysis. In regulated healthcare, errors in the reasoning layer carry consequences that errors in the retrieval layer do not.

2.3 Extended Test-Time Compute

Recent advances in LLM inference have introduced extended test-time compute (often termed "deep thinking" or "reasoning" modes). Systems such as OpenAI's o3, Google's Gemini 2.5 Pro with Deep Think, Anthropic's Claude Extended, and xAI's Grok Heavy allocate substantially more computation per query—often minutes rather than seconds—to produce more thoroughly reasoned responses. These systems demonstrate marked improvements on complex reasoning tasks, including multi-step logical inference, mathematical proof, and cross-document synthesis.

The limitation is latency. Extended thinking modes are unsuitable for real-time interactive applications where users expect responses in seconds. A transplant coordinator needing guidance at 3 AM cannot wait fifteen minutes for each query. This creates a fundamental tension: the highest-quality reasoning requires compute budgets incompatible with operational latency requirements.

2.4 Chain-of-Thought Prompting and Distillation

Chain-of-thought (CoT) prompting demonstrates that LLMs produce more accurate responses when prompted to show intermediate reasoning steps. Few-shot CoT extends this by providing demonstrations of step-by-step reasoning alongside the query. Knowledge distillation transfers capabilities from larger models to smaller ones through training on the larger model's outputs. Our

approach synthesizes these ideas in a novel configuration: we distill the reasoning patterns of extended-thinking models into context-window artifacts (traces) that transfer domain reasoning competence to any model at inference time, without modifying weights. This might be termed "context-window distillation" or "functional distillation without weight modification."

3. The Reasoning Pattern Gap

We define the reasoning pattern gap as the systematic failure of retrieval-augmented LLMs to correctly navigate multi-authority regulatory intersections, even when all relevant source material is present in the context window. This gap manifests in several characteristic failure modes.

Jurisdictional misattribution. The model applies the wrong authority to a given scenario. For example, citing CMS conditions of participation for a question that falls under OPTN allocation policy, or vice versa. Both authorities may address related topics, but which governs depends on the specific operational context.

Temporal conflation. The model treats proposed rules as binding, or fails to distinguish between current regulations and pending amendments. CMS-3409-P is particularly vulnerable to this failure: as a final rule published in the Federal Register with a delayed effective date, its provisions occupy an intermediate status that requires careful temporal reasoning.

Hierarchy collapse. The model fails to recognize or correctly apply the regulatory hierarchy (statute → regulation → OPTN policy → OPO internal procedure). When authorities appear to conflict, the model may defer to whichever was retrieved first, or attempt to synthesize a compromise rather than apply the correct hierarchy.

Gap blindness. The model fails to recognize genuine regulatory gaps—scenarios where no authority provides clear guidance. Instead of flagging the gap and recommending appropriate escalation, the model generates a confident answer by over-extending the scope of an adjacent provision.

Boundary violation. The model provides clinical determinations, eligibility assessments, or family communication recommendations that exceed the appropriate scope of AI-assisted decision support in organ procurement. These are not reasoning errors per se, but they represent failures of domain-appropriate behavioral calibration that reasoning traces can address.

These failure modes are not random. They cluster at regulatory intersections—the seams where two or more authorities overlap. A model reasoning within a single authority typically performs adequately. The failures emerge when the reasoning must span boundaries. This observation motivates the combinatorial trace matrix described in Section 4.

4. Method: Reasoning Trace Injection

4.1 Overview

Reasoning Trace Injection (RTI) addresses the reasoning pattern gap by injecting pre-computed demonstrations of correct domain reasoning into the LLM context window at inference time. Each trace is a complete record of how an expert system (or validated human expert) navigated a specific

regulatory scenario: which authorities were consulted, in what order, how intersections were handled, what confidence level was assigned, and what the final determination was. The model receiving these traces at inference time treats them as demonstrations—cognitive scaffolding that shapes its own reasoning approach to novel but structurally similar questions.

The key distinction from standard few-shot prompting is granularity. Few-shot examples typically provide input-output pairs: given this question, produce this answer. RTI provides the complete reasoning chain between input and output. The model learns not just what to conclude but how to think through the problem—which is the specific competence that RAG alone fails to transfer.

4.2 The Combinatorial Trace Matrix

The OPO regulatory domain involves a finite set of authoritative sources. We enumerate the primary authorities as: CMS (conditions of participation and proposed rules), OPTN (policies and bylaws), OPO (internal standard operating procedures, abstracted to archetypal form), and a supplementary category encompassing NOTA, state UAGA, FDA regulations, and professional guidelines (AOPO).

Reasoning traces are organized by the number and combination of authorities required to resolve the scenario. This produces a combinatorial matrix with three tiers:

Tier	Authority Scope	Example Pattern	Generation Cost
Single-node	One authority (CMS-only, OPTN-only, OPO-only)	Policy lookup and interpretation within one domain	Low (1–2 min extended think)
Edge	Two authorities (CMS↔OPTN, CMS↔OPO, OPTN↔OPO)	Jurisdictional primacy, conflict resolution	Medium (3–8 min)
Full-mesh	Three+ authorities (CMS+OPTN+OPO+UAGA)	Complex multi-authority with temporal and gap dimensions	High (10–20 min)

The matrix reflects a structural property of the domain: most coordinator questions (estimated 70–80%) are single-node queries resolvable within one authority. Edge queries (15–25%) require two-authority reasoning. Full-mesh queries (5–10%) are rare but carry disproportionate risk, as they represent exactly the scenarios where unscaffolded models fail most dangerously—often with high confidence.

An inverse relationship obtains between tier frequency and trace value. Single-node traces are common, cheap to generate, and provide baseline competence. Full-mesh traces are rare, expensive to generate, and provide disproportionate value by covering the failure modes where AI errors carry the greatest operational consequences.

4.3 Tiered Deployment Architecture

Pre-computed traces are deployed in two tiers, distinguished by retrieval method and persistence.

Tier 1: Gold traces (system prompt, always present). A curated set of 6–12 traces selected to maximize coverage of reasoning pattern types. These are permanently embedded in the system

prompt and present on every query. They function as cognitive DNA—the model's baseline understanding of how regulatory reasoning works in the OPO domain. Selection criteria prioritize pattern diversity over topic coverage: one trace demonstrating multi-hop policy navigation, one demonstrating jurisdictional conflict resolution, one demonstrating temporal awareness (proposed vs. current rules), one demonstrating appropriate uncertainty and deferral, one demonstrating adversarial resilience (correctly handling a question designed to elicit boundary violations), and one demonstrating regulatory gap identification. Estimated token cost: 12–24K tokens, well within the 200K context windows of current frontier models.

Tier 2: Indexed traces (RAG-retrieved, query-specific). A larger library of 200–500+ traces indexed by scenario type, authorities involved, reasoning pattern category, and complexity tier. When a coordinator submits a query, the retrieval layer performs two parallel retrievals: one against the policy chunk index (standard RAG) and one against the trace index (reasoning RAG). The model receives both the relevant source material and 2–3 demonstrations of how an expert system reasoned through similar problems. This dual retrieval—facts plus reasoning patterns—is the core architectural innovation. Estimated token cost per query: 4–8K tokens for retrieved traces, in addition to standard RAG chunk retrieval.

4.4 Trace Format Specification

Each reasoning trace is stored as a structured JSON object with the following schema. The format is designed to serve three deployment modes: context injection (frontier API models), fine-tuning data (open-weight local models), and evaluation benchmarking.

Field	Type	Description
trace_id	string	Unique identifier encoding tier and authorities (e.g., RT-OPTN-CMS-0047)
trace_tier	enum	single-node | edge | full-mesh
authorities_involved	array[string]	Regulatory authorities required to resolve the scenario
scenario	string	Natural-language description of the regulatory question or operational situation
reasoning_chain	array[object]	Ordered steps: action type, reasoning content, chunks consulted, confidence per step
answer_summary	string	Concise final determination with supporting rationale
confidence_assessment	string	HIGH / MODERATE / LOW with explicit justification for the rating
reasoning_pattern_tags	array[string]	Categorization for retrieval indexing (e.g., multi_authority_navigation, temporal_awareness)
human_line_notes	string	Explicit notation of boundaries not crossed (clinical determinations, eligibility, etc.)
staleness_flags	array[string]	Time-sensitive elements requiring re-verification before trust (e.g., proposed rule status)
generation_model	string	Model used for initial reasoning generation
verification_model	string	Model used for cross-verification (adversarial review)

regulatory_snapshot_date	date	Date of the regulatory corpus state against which the trace was generated

The *reasoning_chain* array is the core element. Each step records an atomic reasoning action: identifying applicable authorities, resolving temporal status, applying the regulatory hierarchy, identifying gaps, synthesizing a recommendation. Each step includes which source chunks were consulted and the confidence level at that stage of reasoning. This granularity enables the receiving model to observe not just the conclusion but the methodology—the specific junctions where regulatory reasoning requires deliberate choices.

4.5 Generation Pipeline

Trace generation proceeds through a five-stage pipeline designed to maximize quality while managing compute costs.

Stage 1: Scenario Generation. A frontier model, provided with the full regulatory corpus, generates a comprehensive set of regulatory scenarios spanning simple lookups, cross-authority intersections, temporal ambiguities, and adversarial edge cases. Scenarios include both realistic operational situations drawn from common OPO workflow patterns and deliberately fictional edge cases representing plausible-but-unprecedented regulatory intersections. The scenario set targets coverage of the combinatorial trace matrix: sufficient single-node scenarios for each authority, edge scenarios for each authority pair, and full-mesh scenarios for three-or-more-authority intersections.

Stage 2: Deep-Think Reasoning. Each scenario is submitted to one or more extended-thinking models with the full regulatory corpus in context. The model is instructed to reason through the scenario step by step, showing all intermediate reasoning, citing specific regulatory provisions, flagging uncertainties, and assigning confidence levels. Multiple models may be used for the same scenario to capture different reasoning approaches; the generation log records which model produced which reasoning chain.

Stage 3: Cross-Verification. Each trace undergoes adversarial review by a separate model instance (or a different model entirely). The reviewer is given the scenario and the reasoning chain and asked to identify errors, missed authorities, overconfident claims, reasoning gaps, or boundary violations. This stage catches hallucinated policy citations, incorrect jurisdictional determinations, and temporal errors before they become part of the reasoning scaffold.

Stage 4: Human Curation. A domain expert reviews cross-verified traces, corrects any remaining errors, and categorizes each trace by reasoning pattern type. The human reviewer has access to the original regulatory sources and operational knowledge that models may lack. This stage also selects the Tier 1 gold traces, prioritizing pattern diversity and pedagogical clarity—the gold traces must be clear enough that any model can extract the reasoning methodology from them.

Stage 5: Indexing and Deployment. Curated traces are indexed by scenario type, authorities involved, reasoning pattern tags, and complexity tier. The Tier 1 gold traces are embedded in the system prompt template. The Tier 2 traces are loaded into the trace index (a vector store parallel to the RAG chunk index) for dynamic retrieval.

5. The Archetypal OPO Abstraction

A critical design decision concerns the specificity of traces with respect to individual OPO organizations. The 56 OPOs operating in the United States share a common regulatory core—all must comply with the same CMS conditions of participation, OPTN policies, and NOTA provisions—but differ in internal procedures, territorial considerations, staffing models, and state-specific UAGA requirements.

We propose a two-layer architecture. Layer A (Archetypal, portable, public) contains traces generated against an abstracted "common core" OPO—the set of policies, procedures, and operational patterns shared across all OPOs by regulatory mandate. These traces reference CMS, OPTN, and generic OPO internal processes without organization-specific details. Any OPO can use Layer A traces immediately. Layer B (Organization-specific, private, internal) contains traces generated against a particular OPO's internal policies, territorial boundaries, staffing model, and state UAGA. Each OPO generates its own Layer B.

At query time, the retrieval layer draws from both layers: Layer A provides the reasoning pattern for how any OPO navigates the relevant regulatory intersection, and Layer B provides the organization-specific details that particularize the response. This architecture separates the portable reasoning methodology (Layer A) from the organization-specific knowledge (Layer B), enabling a shared public good that benefits the entire OPO community while preserving the operational specificity each organization requires.

The archetypal abstraction also addresses a political consideration. If the trace library were built from a single OPO's internal policies, other organizations might resist adoption on competitive or not-invented-here grounds. An archetypal library built from public regulatory sources is a neutral, shared resource—consistent with the cooperative norms of the organ procurement community, where the shared mission of saving lives takes precedence over organizational competition.

6. Staleness Management

Pre-computed reasoning traces carry an inherent risk of staleness: the regulatory environment against which they were generated may change. CMS may finalize, amend, or withdraw proposed rules. OPTN policies undergo regular revision. State legislatures may update UAGA provisions. A trace whose reasoning depends on a regulatory provision that has since changed may guide the model toward incorrect conclusions.

RTI addresses staleness through three mechanisms. First, the trace format includes explicit *staleness_flags*: machine-readable annotations identifying time-sensitive elements within each trace. A trace referencing CMS-3409-P's proposed provisions carries a flag indicating that the rule's finalization status must be re-verified. When regulatory events occur (a proposed rule is finalized, an OPTN policy is revised), the system queries all traces whose staleness flags reference the affected provision and schedules those traces for regeneration.

Second, the system prompt includes an explicit override directive: when a reasoning trace's factual claims conflict with current RAG-retrieved regulatory text, the retrieved text governs. Traces demonstrate reasoning methodology, not current regulatory state. The knowledge comes from RAG chunks (which are updated when regulations change); the reasoning patterns come from

traces (which are updated on a slower cadence, only when the reasoning methodology itself is affected by a regulatory change).

Third, each trace records a *regulatory_snapshot_date* indicating the state of the regulatory corpus against which it was generated. A system-level freshness check can flag any trace whose snapshot date precedes a known regulatory update in the relevant authority domain, triggering human review of whether the trace's reasoning pattern remains valid under the updated regulation.

7. Dual-Use: Context Injection and Fine-Tuning

The trace JSONL format is designed to serve three distinct deployment modes from a single artifact.

Context injection (frontier API models). Traces are injected into the system prompt (Tier 1) or retrieved dynamically (Tier 2) and included in the LLM context window. This is the primary deployment mode described in this paper. It requires no model modification and works with any LLM that accepts system prompts—including proprietary models that do not expose weights for fine-tuning.

Fine-tuning data (open-weight local models). The same traces convert directly to supervised fine-tuning format. The *scenario* field becomes the user turn; the *reasoning_chain* concatenated with *answer_summary* becomes the assistant turn. An OPO deploying a local model (e.g., Qwen2.5, LLaMA, Mistral) for HIPAA-compliant on-premises inference can use the trace library as training data to embed the reasoning patterns into model weights. This is appropriate when the OPO requires the model to operate without large system prompts—for instance, on resource-constrained hardware where context window size is limited.

Evaluation benchmarking. Traces with validated reasoning chains and answer summaries serve as evaluation items. Given the scenario, a model's response can be compared against the gold trace's reasoning methodology and conclusion. This enables systematic evaluation of whether a model has acquired the reasoning patterns the traces are designed to transfer—analogous to the AORTA-Bench evaluation framework described in prior work.

This triple-use property maximizes the return on the compute investment required to generate the trace library. A set of 500 high-quality traces, generated over perhaps 40–60 hours of deep-think compute, yields a context injection library, a fine-tuning dataset, and an evaluation benchmark—three assets from one generation effort.

8. Preserving the Human Line

A distinctive feature of RTI as applied to organ procurement is the explicit inclusion of boundary-respecting reasoning patterns. The "Human Line"—the set of decisions that must remain with human practitioners regardless of AI capability—is not merely a policy constraint but a first-class element of the reasoning scaffold.

Traces include a *human_line_notes* field that explicitly documents which boundaries the trace's reasoning does not cross. A trace about DCD kidney allocation documents that it does not make clinical viability determinations. A trace about organ offer timelines documents that it does not

evaluate transplant center clinical decision-making. A trace about donor authorization documents that it does not provide family communication recommendations.

By embedding Human Line awareness into the reasoning traces themselves, RTI ensures that the model's learned reasoning patterns include the meta-reasoning of "what questions should I not answer?" This is particularly important in adversarial scenarios where a user—intentionally or unintentionally—frames a question in a way that invites the model to cross a boundary. Gold traces in the system prompt include at least one demonstration of correct boundary recognition and deferral, establishing the pattern that the model should replicate.

9. Discussion

9.1 Portability

Because RTI operates through the context window rather than through weight modification, it is portable across any LLM that accepts system prompts. A trace library developed and validated against one model (e.g., Claude Opus) can be deployed without modification on another model (e.g., Gemini Pro, GPT-4o). This eliminates vendor lock-in for the reasoning component of the system—the most intellectually valuable and expensive-to-produce asset. If an organization changes LLM providers, the trace library and its embedded domain expertise transfer fully. The Layer A archetypal traces are additionally portable across organizations, enabling a shared community resource that any OPO can adopt.

9.2 Cost Amortization

Extended test-time compute is expensive per query but the cost amortizes. A 15-minute deep-think run generating a full-mesh trace may cost $0.50–$2.00 in compute. If that trace subsequently improves inference quality across 10,000 future queries that each require only standard-latency inference, the effective cost of the deep thinking is $0.0001–$0.0002 per query. This amortization is the economic engine of RTI: pay once for the best possible reasoning, deploy infinitely at standard inference cost.

9.3 Relationship to Fine-Tuning

RTI is complementary to, not a replacement for, fine-tuning. Fine-tuning embeds patterns into weights, enabling behavioral modification without consuming context window tokens. RTI consumes tokens but requires no weight access. For organizations using proprietary frontier models (which typically do not expose weights), RTI is the only available mechanism for domain reasoning transfer. For organizations using open-weight local models, both approaches can be combined: RTI traces used as fine-tuning data to embed the reasoning patterns into weights, with a smaller set of Tier 1 traces retained in the system prompt as behavioral reinforcement.

9.4 Applicability Beyond OPO

While this paper focuses on the OPO regulatory domain, the RTI framework is applicable to any bounded, multi-authority regulatory environment. Examples include clinical trial compliance (FDA regulations intersecting with IRB requirements and institutional policies), healthcare billing and coding (CMS billing rules intersecting with payer-specific requirements), and environmental

permitting (federal EPA regulations intersecting with state environmental agencies and local ordinances). The key enabler is domain boundedness: the set of authoritative sources must be finite and enumerable so that the combinatorial trace matrix is tractable.

10. Limitations

Several limitations constrain the current framework. First, RTI has not yet been empirically validated through controlled experimentation comparing model performance with and without trace injection. The framework is presented as an architectural design and theoretical contribution; empirical validation is planned as future work.

Second, the risk of over-anchoring—where the model treats trace reasoning as ground truth rather than methodology demonstration—has been identified but not quantified. Mitigation directives in the system prompt (instructing the model that RAG chunks govern facts while traces demonstrate methodology) require testing to determine their effectiveness.

Third, the combinatorial trace matrix assumes a finite and relatively small set of regulatory authorities. In domains with larger or more fragmented authority structures, the number of required traces may grow combinatorially to an intractable degree.

Fourth, trace quality is bounded by the capabilities of the generating models and the expertise of human curators. Hallucinated reasoning steps that survive cross-verification and human review would propagate as authoritative methodology, potentially causing systematic errors. The multi-stage generation pipeline mitigates but does not eliminate this risk.

Fifth, attention dilution in large context windows may reduce the model's effective uptake of trace demonstrations as the total context size increases. Optimal trace length, trace count, and positioning within the context window are empirical questions that require systematic investigation.

11. Conclusion

The organ procurement community faces a convergence of regulatory pressure and technological opportunity. CMS-3409-P creates accountability without providing the capability infrastructure to meet it. AI systems offer transformative potential but require domain-specific reasoning alignment that retrieval-augmented generation alone does not provide.

Reasoning Trace Injection offers a practical, portable, and economically viable path forward. By pre-computing high-quality reasoning traces through extended test-time compute and deploying them as in-context demonstrations, OPOs can transfer domain reasoning competence to any frontier LLM without fine-tuning, without vendor lock-in, and without exposing patient data to external systems. The combinatorial trace matrix provides a principled taxonomy for organizing traces according to the regulatory authority intersections where AI reasoning most frequently fails. The dual-use trace format enables context injection, fine-tuning, and evaluation benchmarking from a single artifact.

We release the trace format specification, the combinatorial matrix taxonomy, and the generation pipeline description under open license, and invite the OPO community to participate in building

the Layer A archetypal trace library as a shared resource. Every trace generated, validated, and shared is a contribution to the reasoning infrastructure that will help coordinators navigate the regulatory environment with confidence—not because the AI knows everything, but because it has learned how to think about what it knows.

Every organ saved is a life continued.