Structural Diagnostic Report

ai.04 — Runtime Control Coherence

Scenario S2: Retry-Heavy Execution Environment

Document Type:	Structural Diagnostic Scenario Report
Application ID:	ai.04
Scenario ID:	ai.04.S2
Schema Version:	0.5.1

Source: SORT AI Structural Diagnostics Demo
Application: ai.04 Runtime Control Coherence
Scenario: S2
Version: 1.0.1
Generated: 2026-01-18
Web: https://independent-research-systems-modeling.com

Scope and Limits: *This report presents a structural diagnostic scenario analysis based on pre-computed, normalized projection runs. It is not a complete Architecture Risk Assessment and does not contain implementation guidance.*

1. Scenario Overview

System Class

Execution environment with aggressive multi-layer retry logic achieving near-zero visible error rates across application, service, and infrastructure boundaries.

Scale Abstraction

Hidden retry amplification regime with opaque cost attribution and significant actual attempt multipliers masked by success-oriented metrics.

Operational Context

Inference serving with multi-level automatic retry, cascading timeouts with backoff, and eventual completion guarantees. Each retry layer operates independently without visibility into the cumulative retry behavior across the stack.

2. Observed Structural Pattern

The following structural effects emerge from the interaction of correctly configured retry mechanisms operating at scale:

- Application-level retries trigger infrastructure-level retries, creating multiplication rather than addition of attempts across independent retry layers.
- Success metrics mask the actual attempt count, making cost growth appear anomalous rather than structural in origin.
- Timeout cascades create scenarios where a single slow response generates dozens of actual processing attempts before eventual completion.
- Cost attribution systems see only successful completions, not the hidden retry overhead that produced them.
- The problem emerges from the interaction of correct retry behaviors, not from excessive retry rates at any single layer — each retry policy operates within its configured bounds.

3. Stability Assessment

Baseline Structural Condition

System operates in cost-incoherent regime. Success rates remain excellent while costs grow faster than traffic. Stability reserve is eroding invisibly behind successful completion metrics.

Observed Instability Class

Cost-incoherent — characterized by multiplicative retry cascades producing hidden resource consumption decoupled from visible workload characteristics.

Post-Projection Stability Class

Cost-coherent — retry amplification contained through coordinated retry boundaries. Cost trajectory becomes predictable and proportional to actual workload.

Transition Type

Amplification containment from fragmented retry chains to coordinated retry boundaries.

4. Aggregated Indicators

All values are normalized ratios. No absolute values or reconstructable parameters are provided.

Indicator	Baseline	Comparison	Direction
Visible Success Rate	0.997	0.994	Minimal Change
Actual Attempt Multiplier	3.4	1.3	Improvement
Cost per Successful Completion	1.87	1.12	Improvement
Retry Cascade Frequency	0.28	0.07	Improvement
Cost Attribution Accuracy	0.34	0.86	Improvement
Capacity Planning Error	0.41	0.11	Improvement

5. Interpretation

Systemic Relevance

The observed cost incoherence pattern is systemically relevant because it represents a structural property of layered retry architectures rather than a misconfiguration in any individual retry policy. Cost amplification emerges from correct retry behaviors interacting multiplicatively, not from excessive retry rates. This distinction fundamentally changes the appropriate response strategy.

Detection Challenge

This instability class remains undetected in practice because success-based metrics show excellent performance throughout the cost degradation process. The problem exists in the cumulative interaction across retry layers, not in any individual policy. Monitoring systems designed to observe success rates and per-layer retry counts cannot detect cross-layer amplification until cost consequences become economically severe.

Costs grow, capacity planning consistently underestimates requirements, and resource consumption decouples from traffic — yet success rates remain near-perfect when examined at the service boundary.

6. Decision Relevance

If execution environments show excellent success rates but inexplicable cost growth, the underlying cause is likely structural retry amplification rather than traffic anomalies or pricing changes.

Adding more retry resilience increases the amplification surface area and may accelerate cost incoherence rather than improve reliability. Per-layer retry tuning addresses symptoms while potentially worsening the structural condition.

Structural visibility into cross-layer retry dynamics enables targeted intervention at amplification boundaries rather than requiring increasingly aggressive retry policies with diminishing reliability returns and escalating costs.

Related Document: SORT AI Runtime Control Application Context Brief

---

Such structural findings are typically contextualized through a scoped architecture risk assessment.