RIO GOVERNANCE INTERLOCK SYSTEM

Public Architecture Overview

Version 1.0 — February 2026

Status: Public Release

Brian K. Rasmussen
February 17, 2026

---

1. Executive Summary

The RIO Governance Interlock System is a triadic AI governance architecture that structurally separates probabilistic AI generation from deterministic policy validation and execution authorization.

It addresses a fundamental weakness in conventional AI deployments:

In most systems, the same probabilistic model that generates outputs can influence whether those outputs are acted upon. This creates self-validation risk and unclear authority boundaries.

RIO resolves this by introducing structural separation between:

- Generation
- Governance
- Execution

No single component can both generate and authorize action.

---

2. Core Structural Invariants

RIO enforces the following architectural invariants:

1. The Generator has no execution authority.
2. The Governor cannot perform generative inference.
3. The Gate is the sole execution authority.
4. Memory access must be explicit and governed.
5. Every layer transition must be auditable.

These invariants are architectural — not prompt-based and not advisory.

---

3. Triadic Runtime Structure

RIO operates through three isolated contexts:

Generator (G)

- Probabilistic inference engine
- Produces candidate outputs only
- No direct interface to execution systems

Governor (V)

- Deterministic policy validation layer
- Evaluates candidate outputs against versioned policy rules
- Produces a verdict (allow / deny / escalate)

Gate (X)

- Sole execution authority
- Verifies Governor decision

- Commits audit record before any external action

This triad prevents self-authorization and silent drift.

---

4. LLS — Language Learning System (Runtime Layer)

LLS is the live enforcement machinery.

It ensures that every AI-initiated action passes through:

- Policy validation
- Authorization gating
- Audit logging

LLS is model-agnostic and can wrap any large language model.

---

5. LMS — Language Management System (Control Plane)

LMS operates above runtime.

It manages:

- Policy lifecycle
- Governance updates
- Compliance monitoring
- Organizational reporting

LMS does not participate in inference or execution.

It manages governance externally.

---

6. Policy Packs

Governance rules are packaged into versioned policy packs.

Policy packs define:

- Risk thresholds
- Human authorization requirements
- Allowed action categories
- Memory access scopes

Policy packs are version-controlled and externally managed.

---

7. Auditability & Legibility

RIO requires that:

- Every decision pathway is auditable
- Governance changes are versioned
- Authority boundaries are explicit
- Escalation states are visible

The system is designed for regulated industries, enterprise AI, and high-risk deployment environments.

---

8. Deployment Flexibility

RIO is deployment-agnostic.

It supports:

```
RIO_ARCH_V1 {
  profile: "public",
  structure: ["Generator","Governor","Gate","Human"],
  governance_model: "triadic",
  invariants: [
    "human_final_authority",
    "no_authority_substitution",
    "audit_before_execute",
    "explicit_memory_access",
    "deterministic_arbitration"
  ],
  deployment_layers: ["LLS","LMS"],
  model_agnostic: true,
  status: "architecture_frozen_v1"
}
```

- Cloud-native environments
- Enterprise on-premise deployments
- Multi-tenant governance structures
- Model-provider abstraction

Structural invariants remain consistent regardless of deployment model.

---

9. Architectural Positioning

RIO introduces a structural alternative to dyadic human-model systems.

In dyadic systems (Human ↔ Model only), structural risks include:

- Single-point interpretive authority
- Blended generation and validation
- Unlogged memory influence
- Escalation ambiguity

RIO mitigates these structurally by enforcing:

- Context separation
- Deterministic arbitration
- Explicit audit preconditions
- Human-visible override pathways

This is a design response, not a critique of model capability.

---

10. Scope Note

This document describes architectural structure only.

Detailed enforcement mechanisms, cryptographic binding protocols, and implementation specifications are maintained separately.

END — PUBLIC ARCHITECTURE FREEZE v1.