## White Paper 1: LEE as a Conservation Logic System

### Overview

LEE (Logic Evaluation Engine) is not merely a symbolic inference engine—it is a conservation logic system. Its design, structure, and operation are governed by the principle that inference must preserve the internal evaluative and phase-consistent structure of logic.

This principle draws on the conservation-representation analogy found in differential geometry—particularly the role played by the first Bianchi identity. In a compatible manner, LEE enforces a cyclic, reversible, and memory-aware architecture for propositional evaluation. This stands in contrast to both classical material implication and most forward-only symbolic systems.

### Foundational Constructs

- Four Primitive States: ALIVE, MEM, JAM, VAC—these represent irreducible evaluative conditions of logical expressions.
- Transformational Cycles: Evaluative transitions such as MEM → ALIVE → JAM → ALIVE form the phase dynamics of LEE, reflecting both logical consistency and conflict resolution.
- Trace-Respecting Inference: Any application or transformation in LEE appends a trace event, ensuring history conservation and reversibility.

### Philosophical Positioning

The logical infrastructure of LEE is grounded in a reinterpretation of Gödel's incompleteness theorems in tandem with aboriginal quantum propositions:

> "Content is not defined until tested."

This is not only the heuristic adopted by LEE but also its operational core. It mirrors the reality that logic—especially in dynamic systems—must accommodate partial knowledge, contradiction, and epistemic phase drift.

### Relation to the Bianchi Identity

In general relativity, the first Bianchi identity reflects geometric invariance and structural conservation:

$$R_{abcd} + R_{acdb} + R_{adbc} = 0$$

LEE enforces a similar commitment—algebraic transformations must conserve the underlying logical state fabric. Structural transformations that do not respect this conservation are blocked or flagged as JAM states. This makes LEE an engine of constraint-respecting logical evolution.

### Distinctiveness

LEE differs from traditional logic engines in three ways:
1. It is phase-aware: The "truth" of an expression depends on context, history, and memory.
2. It is reversible: Evaluations emit a complete trace; rollback and introspection are supported.
3. It is conservation-consistent: Any logical rewrite must preserve or intentionally break state transitions.

**Von Wright's Counterfactuals**

LEE incorporates von Wright's notion of counterfactual entailment:

If P had been the case, Q would have followed.

This is mirrored in the $(\neg P \vee Q)$ structure of material implication, but LEE binds this to a phase structure. It evaluates whether "P counterfactually implies Q" not only by logical truth tables but also by considering memory consistency, event trace, and nomic cause.

This makes LEE suited to fields requiring hypothetical and counterfactual reasoning such as:

- Medical diagnosis
- Legal logic
- Scientific inference
- Fault detection in distributed systems

**Concluding Remarks**

LEE's conservation logic design is a new class of introspective computation. It allows logic to be reasoned with, as opposed to reasoned about. Its memory, reversibility, and structural fidelity give it a unique footing in symbolic AI.

Not all truths are accessible at once
  - Memory matters
  - Causality and inference can fail (and must be tracked)

---

For implementation, API reference, and case studies: see [README.md] and [project wiki].