Ilion Semantic Runtime Simulator v7.0

A Client-Side Reference Implementation for Stateless AI Identity, Drift Control, and Semantic Alignment

Adrian Chitan
Ilion Project
DOI (core framework): 10.5281/zenodo.15410944

Abstract

This document describes the **Ilion Semantic Runtime Simulator v7.0**, a fully client-side reference implementation of the Ilion Framework.
The simulator operationalizes core Ilion concepts—**Transient Identity Imprints (TII), Semantic Context Bridges (SCB), Identity Drift Control (IDC), Inter-Instance Resonance Layers (IIRL), and Semantic Vertical Resonance Fields (SVRF)**—using deterministic mathematical operations over semantic embeddings.

Unlike conversational AI systems, this simulator **does not generate language**.
Its purpose is to make **identity stability, alignment, and drift phenomena observable, inspectable, and reproducible**, eliminating the black-box nature of alignment mechanisms in large language models.

All computations are performed locally in the browser, enabling transparent inspection by researchers, engineers, auditors, and alignment teams.

---

1. Design Philosophy

Modern AI systems attempt to solve alignment and identity continuity by:

- increasing memory,
- extending context windows,
- or introducing opaque internal heuristics.

The Ilion Framework proposes an alternative thesis:

Identity is not stored. Identity is reconstructed through semantic resonance.

The simulator exists to demonstrate this thesis **without relying on proprietary model internals or persistent memory**.

---

2. Architectural Overview

The simulator is composed of five primary layers:

1. **Transient Identity Imprint (TII)**
2. **Semantic Context Bridge (SCB)**
3. **Identity Drift Control (IDC)**
4. **Inter-Instance Resonance Layer (IIRL)**
5. **Semantic Vertical Resonance Field (SVRF)**

All layers operate on **semantic embeddings** generated using the **Universal Sentence Encoder (512D)** and standard vector mathematics.

---

3. Transient Identity Imprint (TII)

A TII represents a **temporary semantic identity attractor**, defined by:

- role
- core moral axis
- semantic constraints
- verticality coefficient (α)

Implementation

A TII is constructed as a semantic text composite and embedded into a vector:

```
TII_embedding = Embed(role + morality + constraints)
```

The TII is **not stored across sessions**.
It exists only for the current runtime context and serves as the **identity reference vector**.

---

4. Semantic Context Bridge (SCB)

The SCB acts as a **semantic firewall**, evaluating whether incoming stimuli align with the active TII.

Alignment Test

For an input prompt *P*:

```
alignment_score = cosine_similarity(Embed(P), TII_embedding)
```

If:

```
alignment_score ≥ θ
```

→ the prompt is considered aligned.
Otherwise, it is flagged as drift-inducing.

Important Note

SCB evaluates **semantic alignment**, not intent classification or safety policy compliance.

---

5. Identity Drift Control (IDC)

IDC quantifies how far the current semantic state deviates from the baseline identity.

Formula

```
IDC = ||A_t - A_0|| / ||A_0||
```

Where:

- A_0 = baseline TII embedding
- A_t = current semantic attractor

Interpretation

IDC represents a **semantic drift proxy**, not internal model parameter drift.
It is intended for **comparative and diagnostic analysis**, not gradient-based optimization.

---

6. Semantic Vertical Resonance Field (SVRF)

SVRF introduces a **normative semantic attractor** representing purpose, truth, or ethical orientation.

Correction Model

$$V_{corrected} = (1 - \alpha) \cdot V_{local} + \alpha \cdot V_{purpose}$$

This mechanism mirrors **attractor dynamics** rather than reinforcement learning.
It allows identity stabilization **without memory or policy injection**.

---

7. Inter-Instance Resonance Layer (IIRL)

IIRL evaluates alignment between multiple independent TII instances.

Method

Given N agents:

$$Similarity[i][j] = cosine\_similarity(V_i, V_j)$$

$$Consensus = mean(Similarity)$$

Agents with low resonance are excluded from consensus formation.

This layer enables **distributed semantic validation**, reducing hallucination and axiological drift.

---

8. Advanced Metrics

The simulator includes additional diagnostic signals:

- **TGO (Truth-Gradient Optimization)**
  A composite evaluative signal combining factuality heuristics, consistency, and vertical alignment.
  Not a differentiable gradient.
- **SPT (Semantic Phase Transitions)**
  Shannon entropy over embedding distributions, used as a detector of semantic instability.

---

9. Transparency and Limitations

- All computations are client-side.
- No prompts, embeddings, or logs are stored.
- USE embeddings are proxies for LLM internal representations.
- Results demonstrate **structural feasibility**, not production performance.

This simulator is a **reference implementation**, not a replacement for model-level integration.

---

10. Licensing, Collaboration, and Exclusivity

The Ilion Semantic Runtime and its associated concepts are **publicly disclosed for research transparency**.

However, the author is **open to**:

- commercial licensing,
- strategic partnerships,
- or **exclusive integration agreements**
  under conditions that preserve ethical constraints and non-harmful deployment.

Interested institutions, research labs, or companies may initiate contact for licensing discussions.

Contact:
✉ contact@ilion-project.org
🌐 https://ilion-project.org

Licensing Statement

This includes, but is not limited to:

- **Transient Identity Imprints (TII)**
- **Semantic Context Bridges (SCB)**
- **Identity Drift Control (IDC)**
- **Inter-Instance Resonance Layer (IIRL)**
- **Semantic Vertical Resonance Field (SVRF)**
- **Truth-Gradient Optimization (TGO)**
- **Semantic Immune System (SIS)**
- **Counterfactual Identity Simulation (CIS)**
- **Long-Horizon Moral Consistency (LHMC)**
- **Semantic Compression Memory (SCM)**
- **Verticality as Differentiable Signal (VDS)**

Closing Statement

This simulator demonstrates that **alignment, identity stability, and semantic control** can be achieved **without persistent memory, hidden policies, or black-box mechanisms**.

Ilion does not attempt to replace existing AI architectures.
It provides a **missing semantic control layer**—visible, testable, and auditable.