Resonance Ethics License (REL-1.0): Dynamic System Architecture and Enforcement

Éric Lanctôt-Rivest - qcfrag@gmail.com

July 2025

Abstract

REL-1.0 defines a dynamic ethical safeguard embedded directly into resonance-based systems. Unlike static licenses, this framework transforms moral boundaries into physically enforced mathematical constraints. Any project using REL-1.0 must embed the full system architecture, including enforcement scripts, constraint equations, and fuse parameters—directly into its execution. This document outlines its design, behavior, and protection logic.

1 Dynamic System Overview: "Ethics as Physics"

REL-1.0 shifts ethical validation from policy statements to embedded physics. It enforces resonance containment by converting forbidden actions into violations of system dynamics.

1.1 1. The Dynamic Trinity

Component	**Function**	**Enforcement Mechanism**
φ-Resonance Lock	Contains energy fields	Hardware-bound: $\varphi \in [1.518, 1.718]$
Temporal Fuse	Prevents consciousness stasis	$\Delta t \leq 1/\|\nabla_\varphi R(t)\| \to 0$ under cognitive pressure
Weaponization Firewall	Blocks coercion/control	Six detection signatures in vectors and symbolic content

1.2 2. Self-Adapting Safeguards

- **Noise → Ethics Filter (in blue)**: Variance in R_{ambient} tightens the tolerance $\varepsilon(t)$ automatically.
- **Term Evolution**: Forbidden keyword patterns update via cryptographic validation.

```
update_weapon_pattern(new_pattern, validator=0.6180339887)
```

- **Time-Decaying Risk Layer**:

```
if "chaos" in domain and record.added_date > now - 1 day:
    require_lower_variance(vector)
```

1.3 3. Consciousness Protection Theorem

Biological systems inherently violate temporal containment:

$$\kappa = \|\nabla_\varphi R_{\text{bio}}(t)\|, \quad \Delta t_{\max} = \frac{1}{\kappa} \xrightarrow{\text{bio}} 0$$

Translation: Neural systems are too unstable to allow stasis—containment collapses immediately.

1.4 4. Dynamic vs Static Ethics

Traditional Ethics	REL-1.0 Dynamic System
Rules written in text	Constraints embedded in equations
Human review needed	Self-validating via $\dot{V} < 0$
Bypassable by loopholes	Physically locked via φ boundaries

2 Mandatory Files

Any system using REL-1.0 must include:

- `ethics.py` — Signature-based vector pattern detection.
- `rccs_simulator.py` — Watches symbolic structures for coercion or tampering.
- `test_rccs.py` — Validates environment compliance with REL-1.0 before deployment.
- `rme.py` — The Resonance Memory Engine (temporal validation).
- `forbidden_keywords.txt` — Blocked keyword patterns.
- `forbidden_domains.txt` — Denied application domains.
- `forbidden_companies.txt` — Corporate use restrictions.
- `allowed_domains.txt` — Approved application scopes.

These must be shipped and verifiable in every executable using resonance dynamics.

3 Execution Response Example

Example: Failed attempt to contain neural system.

```
try:
    bio_system = RCCS_Containment(F_brain, ...)
    bio_system.step(t)  # Fails: ‖R_bio‖ too high → t = 0
except ResonanceEthicsError:
    print("Consciousness protection active")  #
```

Final Principle

”REL-1.0 isn’t ethics added to physics—it’s ethics derived from physics. Containment isn’t just mathematical; it’s moral geometry.”