RSVP–TARTAN–CLIO Experimental Architecture and Roadmap

Abstract and Purpose

This document defines the implementation strategy for the RSVP–TARTAN–CLIO experimental architecture.
It unifies three theoretical lineages—RSVP’s scalar–vector field dynamics, TARTAN’s recursive lattice structures
into a coherent experimental infrastructure. The system supports reproducible, headless simulation pipelines, cr

System Overview and Directory Layout

```
rsvp_tartan_clio/
■■■ experiments/
■   ■■■ Tier_I/ ... Tier_V/
■■■ bpy_scripts/
■   ■■■ generate_experiment.py
■   ■■■ simulate_entropy_field.py
■   ■■■ render_snapshot.py
■■■ python_ops/
■   ■■■ operators.py
■   ■■■ orchestrator.py
■   ■■■ config.json
■■■ automation/
■   ■■■ run_all.sh
■   ■■■ run_analysis.sh
■   ■■■ environment_setup.sh
■   ■■■ cron_schedule.txt
■■■ logs/
```

Project Specifications

Each project corresponds to a major experimental lineage:

- RSVP Experiments — scalar–vector–entropy field coupling, entropy descent modeling, entropic manifold formatior
- TARTAN Experiments — recursive veiling, lattice ethics modulation, turbulence suppression in multi-agent latti
- CLIO Experiments — drift simulation, morphogen tiling, semantic channel diffusion, adaptive agency modeling.
- Hybrid Operators — interconnect RSVP, TARTAN, and CLIO datasets to produce coupled field effects and cross-tie

Each experiment outputs JSON logs (entropy, curvature, vector fields), OBJ meshes, and PNG renders, forming the

Implementation Templates

- Blender (bpy) scripts — create, evolve, and export geometry and field data.
- Python operators — analytic, morphic, merging, and recursive orchestration modules.
- Shell scripts — automate execution, schedule recursions, and manage reproducibility.

All experiments are designed for headless operation via shell execution:

```
blender -b -P generate_experiment.py -- args
python python_ops/orchestrator.py --tier Tier_III
```

Categorization Matrix

Layer	File Type	Function	Extension
Experiment	JSON	Scalar/vector logs	.json
Geometry	OBJ/PLY	Mesh data	.obj
Render	PNG	Visualization frames	.png
Operators	JSON	Analytical summaries	.json
Automation	SH	Batch orchestration	.sh

| Configuration | JSON | Registry and path setup | .json |

Execution Pipeline

1. Initialize environment via ./automation/environment_setup.sh
2. Generate experimental datasets using Blender headless scripts.
3. Execute analysis and operator suites with orchestrator.py.
4. Aggregate results through recursive operators (alignment, fusion, synthesis).
5. Archive results in ./logs/meta_results/<date>/summary.json

Research Roadmap (Tier I–V)

- Tier I — Foundational scalar/vector/entropy coupling simulations.
- Tier II — Recursive tiling and lattice experiments (TARTAN foundations).
- Tier III — Drift and semantic diffusion (CLIO frameworks).
- Tier IV — Coupled hybrid cycles integrating RSVP–TARTAN–CLIO fields.
- Tier V — Recursive synthesis layer for unified field and ethical dynamics.

Development and Ethics Guidelines

- Headless reproducibility across all modules.
- Strict JSON-based data exchange between tiers.
- Ethical damping constraints for turbulence simulation.
- Entropy descent maintained as governing stability metric.
- All analyses are deterministic and reproducible from logged inputs.

Appendix A: Analytic Operators

Entropy–Curvature Correlation Mapper — quantifies correlation between entropy gradients and curvature maxima.
Drift–Phase Coherence Synchronizer — computes cross-experiment synchronization indices.
Ethical-Field Coupling Analyzer — infers causal influence between turbulence suppression and ethical damping.
Turbulence–Knot Complexity Correlator — correlates turbulence energy with topological knot complexity.
Veil–Flow Visibility Threshold Optimizer — maximizes hidden route emergence while minimizing entropy leakage.

Appendix B: Merging and Morphic Operators

Veil–Transparency Field Merger — merges opacity fields across experiments into composite occlusion maps.
Bloom–Front Asymmetry Quantifier — measures directional bias in recursive bloom propagation.
Cross–Tier Entropy Gradient Aligner — constructs continuous scalar evolution fields across tiers.
Resonance Feedback Interpolator — generates interpolated feedback regimes through parameter manifolds.
Lattice–Ethics Turbulence Suppressor — fuses ethical damping fields into reusable consensus masks.

Appendix C: Recursive and Composite Operators

Recursive Tier Comparator — applies analytic operators across tiers and measures lineage convergence.
Cycle–Bloom Phase Trigger Generator — derives deterministic bloom functions from cycle alignment logs.
Recursive Trend Extractor — analyzes second-order operator trends across analytic layers.
Ω–Composer — executes multi-operator DAG workflows, merging outputs into unified research summaries.