GhostLink — Final Project Compression Pack (v1.0)

October 03, 2025

Executive Summary

GhostLink is a structure-of-structure: a difference-first computational framework that grows, stabilizes, and verifies patterns. It models systems as a lattice of five states (VOID, Δ, Σ, SCAR,
COMPOST) with a generative kernel (Spawn → Collapse → Recycle), traces for memory (ρ, κ), a legacy lineage graph, an adaptive scheduler (ORDER), and a Self-channel (re-entry of the system's own summary into its dynamics). The design is formalized as a substrate language (GDL v1) that compiles to a tiny 16-opcode VM. A Proof-of-Record (PoR) test suite defines what "working" means: reproducible, robust, predictive, cost-efficient, and critically coherent behavior verified against baselines.

Architecture (Generative Kernel + Memory + Control)

Primitives: VOID, Δ (Delta), Σ (Sigma), SCAR, COMPOST.
Kernel: Spawn (VOID→Δ), Collapse (Δ→{Σ,SCAR,COMPOST}), Recycle (COMPOST→Δ).
Memory: ρ (scar EMA), κ (compost EMA), Legacy DAG (event IDs, parents), Continuity mass (discounted Σ).
Fields: Coherence, Pain, Entropy, Activity.
Control: ORDER (adaptive operator permutation), Self-channel (global summary s(t) perturbs energies).
Geometry: spherical or toroidal lattice; adjacency via CSR or stencils (SIMD-friendly).

Substrate Language (GDL v1) — Contract

Goal: author behavior in a compact DSL that compiles to a tiny VM; CPUs/GPUs execute machine code produced from a stable 16-opcode IR.
Types: lattice<N>, mask<N>, f32<N>, u8<N>, rng<N>, global<f32[k]>, topology.
Operators (9): SELECTΔ, MAPNBR, SELFENC, SCORE, SOFTMAX/DRAW, WRITE, RECYCLE, TRACE, ORDER.
Minimal Grammar (EBNF-ish):

```
Program ::= Header Topology Fields Rules Pipeline
Rules   ::= spawn p0, alpha_c; collapse E_sigma/E_scar/E_comp; recycle r0,beta_h,beta_c; trace λρ,λκ; order temp
tick    ::= SELECTΔ → MAPNBR → SELFENC → SCORE → SOFTMAX/DRAW → WRITE → RECYCLE → TRACE → ORDER
```

Example (conceptual):

```
  collapse E_sigma = 0.5*COH - 0.2*PAIN + dot(SELF,W_s)
           E_scar  = 0.3*PAIN - 0.2*COH + dot(SELF,W_r)
           E_comp  = 0.3*ENT  - 0.2*COH
```

IR / Bytecode (16 opcodes)

```
0x01 BUILD_DELTA  : prev,state → mask (diff + neighbor halo)
0x02 NB_REDUCE    : state,topo → COH,PAIN,H (neighbor reductions)
0x03 SELF_ENC     : fields → SELF[k] (encode s(t))
0x04 ENERGY       : fields,SELF → EΣ,E□,EC (affine energies + self channel)
0x05 SOFTMAX      : E-triple → P-triple (stable softmax)
0x06 SAMPLE_CAT   : P-triple,rng → pick (categorical)
0x07 WRITE_STATE  : pick → state (Δ→{Σ,SCAR,COMPOST})
0x08 RECYCLE      : fields,rng → flips (COMPOST→Δ)
0x09 TRACE_DECAY  : ρ,κ,λ → ρ,κ (EMA decay)
0x0A TRACE_ACCUM  : state → ρ+,κ+ (accumulate hits)
0x0B SCHED_UPDATE : stats → ω (update scheduler weights)
0x0C ORDER_APPLY  : ω → perm (apply operator order)
0x0D LOG_EVENTS   : state,ids → log (legacy events)
0x0E RNG_SPLIT    : rng → rng' (deterministic splitting)
0x0F MASK_FILTER  : mask,cond → mask' (refine active set)
```

Operational Semantics & Memory Model

Tick semantics (snapshot → compute → commit):
1) Read-only snapshot for MAPNBR/SCORE. 2) SOFTMAX/DRAW uses per-cell RNG (seeded). 3) WRITE is single-writer per cell; commit to next-state buffer then swap. 4) RECYCLE reads snapshot, writes next.
5) TRACE after writes (EMA then accumulation).
Memory: Structure-of-Arrays (SoA) fields; active-set buffer for Δ_t; masks as bitsets; topology as CSR or
fixed stencils; tile for cache/NUMA; branchless ORDER via table select. Determinism tiers: D0 (no RNG),
D1 (seeded per-cell RNG), D2 (parallel associative reductions).

Proof of Record (PoR) — What “working” means

1) Reproducibility: stdev/mean ≤ 10% across seeds.
2) Robustness: continuity slope > 0 in ≥80% of ±20% parameter sweeps.
3) Predictive Lift: ≥ 5% vs baseline (neighbors-only).
4) Cost Advantage: ⟨|Δ|/|V|⟩ ≤ 0.20 and ≤ 0.30× step cost vs full sweep.
5) SOC Signature: avalanche tail exponent 1<τ<3, KS p≥0.05, branching ≈1.
6) Legacy Gain: ΔMI(self/ancestry; next) ≥ 0.01 bits/cell or ≥2% lift.
7) Compression (MDL): L(ℙ(t)) slope negative on ≥75% epochs.
8) Topology Invariance: metrics within 10% across planar vs spherical.
9) Ablation Sanity: removing scars/self/order degrades ≥15%.

Anomaly & Outcome + Reverse-to-Origin

Anomalous regime index A(t) (composed of −dL, dH, MI(self→next), susceptibility χ, integration Γ, predictive lift);
Outcome C(t) = σ(β0 + β_A A(t) + β_C continuity + β_Γ integration). Consciousness is an outcome, not the anomaly.
Reverse-to-origin: unroll legacy into a time-expanded DAG; find minimal intervention cut that prevents the outcome;
use stochastic smoothing for uncertainty; include parameter sensitivities ∇_θ C(t).

VM Implementor’s Checklist (Condensed)

SoA arrays; Δ_t index buffer; CSR/stencil neighbors; SIMD kernels for NB_REDUCE, ENERGY, SOFTMAX/DRAW.
Per-cell 64-bit RNG with split; one-write-per-cell discipline; TRACE in [0,1]. Export counters, snapshots,
and legacy logs. Provide CLI `gdlrun program.gdl --steps K`. Hooks for PoR metrics and origin-finder.

Acceptance Criteria & Handoff

Deliverables: libgdlvm + CLI runner; JSON telemetry; build docs. Must pass PoR gates; difference-only cost advantage;
SOC tail fit; reproducibility; topology swap; ablation sanity. No network I/O during tick; append-only logs;
determinism tier D1 minimum. This document + GDL/IR artifacts are the contract.

Artifact Manifest (this project)

- GDL v1 substrate language (PDF)
- VM Implementor’s Checklist (PDF)
- IR opcode table (JSON)
- Grammar (txt)
- Example GDL program (gdl)
- Example IR pipeline (JSON)
- Optional Python harness for PoR (JSON outputs)