Shunyaya Structural Substrate Layer (SSSL)

Structural Electrodynamic Substrate with Finite Regime Algebra

---

Version: v1.8
Release Date: 18 February 2026
Status: Public Open Standard (Deterministic Structural Electrodynamic Substrate Layer)

Caution: Research, observability, and structural substrate experimentation only. Not a predictive, control, or safety-critical engine.

Scope Notice:
Shunyaya Structural Substrate Layer (SSSL) defines a deterministic structural substrate operating alongside classical electrodynamics. SSSL preserves classical magnitude exactly while introducing a finite structural regime algebra governing electrodynamic evolution. SSSL does not modify Maxwell's equations, redefine electric charge, introduce probabilistic inference, or simulate physical dynamics. Conforming implementations shall publish parameter sets, operator definitions, transition rules, and replay-verification artifacts sufficient to enable independent deterministic reproduction of results.

---

0. Abstract

Shunyaya Structural Substrate Layer (SSSL) is a deterministic structural electrodynamic substrate layer operating in parallel with classical electromagnetism.

SSSL introduces:

• **Finite structural alphabet `A4`**
• **Deterministic transition operator `delta_E`**
• **Closed structural operators over `A4`**
• **Collapse-preserving invariant `phi((m,a)) = m`**
• **Replay-verifiable execution discipline**

SSSL does **not** modify classical magnitude.
SSSL does **not** alter Maxwellian field laws.
SSSL does **not** introduce prediction or simulation.

Instead, SSSL governs structural posture of observable charge evolution while preserving classical electrodynamic values exactly.

In this document, the term **“civilization-grade”** refers strictly to verification discipline:

• Deterministic execution
• Finite structural state space
• Conservative collapse preservation
• Replay-verifiable artifacts
• Explicit parameter disclosure
• Documented misuse boundaries

It does **not** imply regulatory certification or production deployment approval.

TABLE OF CONTENTS

1. Purpose

Classical electrodynamics evaluates measurable magnitude:

```
m = E_proxy
```

Where `E_proxy` may represent voltage, charge proxy, or measurable electric quantity.

Classical theory determines **what the magnitude is**.
It does not formally govern **structural posture of its evolution**.

Shunyaya Structural Substrate Layer (SSSL) introduces a deterministic structural substrate over observable charge evolution.

Given a measurement sequence `{E_i}`, SSSL defines a structural pair:

```
X_i = (m_i, a_i)
```

Where:

- `m_i` = classical magnitude (**unchanged**)
- `a_i ∈ A4` = structural electrodynamic posture

Structural alphabet:

```
A4 = {Z0, Eplus, S, Eminus}
```

Collapse Invariant:

```
phi((m_i, a_i)) = m_i
```

This invariant is **non-negotiable**.

SSSL does **not** redefine electric charge.
SSSL does **not** alter polarity definitions.
SSSL does **not** modify Maxwell's equations.

SSSL establishes a formally defined **finite regime algebra governing electrodynamic evolution** while preserving classical magnitude exactly.

Where classical physics describes magnitude,
SSSL governs structural evolution posture deterministically.

2. Structural Alphabet and Finite Regime Algebra (A4)

SSSL is built upon a **finite structural electrodynamic alphabet**:

```
A4 = {Z0, Eplus, S, Eminus}
```

Cardinality Condition:

```
|A4| = 4
```

This alphabet is:

• **Finite**
• **Deterministic**
• **Closed under defined operators**
• **Replay-stable**
• **Non-expandable without version change**

No runtime state creation is permitted.
No probabilistic blending is permitted.
No fuzzy regime extension is permitted.

Any fifth structural state invalidates conformance.

2.1 Structural State Definitions

Each structural element in `A4` represents a formally defined posture of electrodynamic evolution.

Z0 — Structural Neutral Equilibrium
Stable near-zero regime.
Magnitude remains within the neutral threshold band and exhibits no structural acceleration.

Eplus — Accumulation Regime
Magnitude exhibits positive structural growth beyond stability band.
Represents directed structural accumulation.

S — Stable Charged Regime
Magnitude resides above stability threshold and remains structurally stable within tolerance band.
Represents plateau or sustained equilibrium.

Eminus — Discharge Regime
Magnitude exhibits structural collapse via explicit discharge flag or deterministic drop condition.
Represents negative structural transition.

These states describe **structural evolution posture**, not polarity identity.

2.2 Algebraic Properties of A4

`A4` forms a **finite deterministic structural algebraic domain** under SSSL operators.

Let `a ∈ A4`. Then:

- `a` is uniquely defined at each observation index
- `a` evolves deterministically via `delta_E`
- `a` participates in closed structural operations
- `a` does not alter classical magnitude

Closure Condition:

For any structural operator `op_s` defined in SSSL,

```
op_s : A4 × A4 -> A4
```

must hold.

Structural operations may never produce a value outside `A4`.

2.3 Structural State Definition

For each observation index `i`, define the structural state:

```
X_i = (m_i, a_i, s_i)
```

Where:

`m_i = E_i` (classical magnitude)
`a_i ∈ A4` (structural regime assignment)
`s_i` = deterministic accumulation component

Collapse Operator (Canonical Invariant):

```
phi((m_i, a_i, s_i)) = m_i
```

This invariant ensures:

- Structural regime assignment never alters magnitude
- Classical domain equations remain intact
- SSSL is strictly conservative
- Magnitude preservation is unconditional and exact

2.4 Finite Regime Stability Principle

Let

```
Sigma = { a_i | i = 1..N }
```

Then

```
Sigma ⊆ A4
```

and

```
|A4| = 4
```

This bounded regime property guarantees:

• No regime explosion
• No structural drift outside defined alphabet
• Deterministic replay stability
• Algebraic closure

The finiteness of `A4` is a foundational civilizational constraint.

2.5 Substrate Implication

By defining `A4 = {Z0, Eplus, S, Eminus}` as a finite structural algebra, SSSL operates beyond observational tagging and establishes deterministic structural discipline over magnitude evolution.

Where magnitude describes measured state,
`A4` governs structural regime state.

This distinction defines SSSL as a structural substrate layer:

• **Magnitude remains primary and unchanged.**
• **Structural regime assignment becomes finite and explicit.**
• **Regime assignment is deterministic and replay-verifiable.**

Substrate status arises from finite regime closure,
conservative collapse `phi((m,a,s)) = m`,
and replay identity `B_A = B_B`.

Structure is governed.
Magnitude remains real.

---

3. Deterministic Transition Operator (`delta_E`)

SSSL defines a deterministic structural evolution law over the finite alphabet `A4`.

Let the observable input at index `i` be:

```
obs_i = (E_i, dE_dt_i, discharge_i)
```

Where:

`E_i` = classical magnitude
`dE_dt_i = E_i - E_{i-1}`
`discharge_i ∈ {0,1}`

The structural transition operator is defined as:

```
delta_E : (A4, obs_i) -> A4
```

Such that:

```
a_i = delta_E(a_{i-1}, obs_i)
```

This defines SSSL as a **finite deterministic structural evolution system**, not a static classifier.

3.1 Base Deterministic Mapping Law

Let fixed constants be defined:

`tau0 > 0` (neutral threshold)
`taus > tau0` (stability threshold)
`eps > 0` (stability band)
`drop > 0` (discharge threshold)

These constants must remain fixed across the dataset.
Adaptive tuning is not permitted within a single execution scope.

The deterministic base state function `F` is defined as:

```
F(E_i, dE_dt_i, discharge_i) =
```

- `Z0` if `E_i <= tau0` AND `|dE_dt_i| <= eps`
- `S` if `E_i >= taus` AND `|dE_dt_i| <= eps`
- `Eminus` if `discharge_i = 1` OR `dE_dt_i <= -drop`
- `Eplus` otherwise

The transition operator is therefore:

```
a_i = F(E_i, dE_dt_i, discharge_i)
```

This mapping is:

- **Total**
- **Deterministic**
- **Single-state**
- **Closed over A4**
- **Replay-stable**

3.2 Deterministic Evolution Law

Given dataset:

```
D = { (E_i, discharge_i) | i = 1..N }
```

The structural sequence:

```
S = { a_i | i = 1..N }
```

is uniquely determined by:

```
D
tau0, taus, eps, drop
```

No randomness is permitted.
No smoothing is permitted.
No interpolation is permitted.
No inferred hidden variables are permitted.

For identical inputs and identical constants:

```
B_A = B_B
```

Byte-identical outputs are mandatory.

3.3 Deterministic State Machine Interpretation

Because `a_i` depends only on deterministic inputs and fixed thresholds, SSSL forms a finite-state deterministic system:

State space: `A4`
Transition driver: `obs_i`
Evolution rule: `delta_E`

There are:

• No absorbing states (under sustained excitation)
• No stochastic transitions
• No probabilistic weighting
• No undefined mappings

Thus SSSL is a **deterministic structural electrodynamic state machine** operating alongside classical magnitude evolution.

3.4 Collapse Preservation Under Evolution

For each index `i`:

```
X_i = (m_i, a_i)
```

Collapse operator:

```
phi_E((m_i, a_i)) = m_i
```

Since `a_i` is computed without modifying `m_i`,

```
phi_E(delta_E(X_{i-1}, obs_i)) = m_i
```

Therefore, structural evolution does not alter classical electrodynamic magnitude at any step.

This invariant preserves conservative extension over classical physics.

3.5 Substrate Implication

By defining a **deterministic structural mapping `F`**, SSSL transitions from static regime labeling to **finite structural evolution discipline**.

Magnitude evolves according to classical physics.
Structural posture evolves according to deterministic finite algebra.

These two layers operate in parallel:

- Magnitude is governed by domain equations.
- Structural regime is governed by `A4`.
- Collapse remains conservative: `phi((m,a,s)) = m`.

There is no interference.
There is no modification of magnitude.

Structure evolves deterministically.
Physics remains untouched.

---

4. Structural Electrodynamic Operators (Closed Algebra)

To elevate SSSL from deterministic evolution to a full structural substrate,
`A4` must support **closed structural operators**.

Let

```
A4 = {Z0, Eplus, S, Eminus}
```

An operator `op_s` is valid under SSSL if and only if:

```
op_s : A4 × A4 -> A4
```

Closure is mandatory.
No operator may produce a value outside `A4`.

All operators must preserve the collapse invariant:

```
phi_E((m, op_s(a,b))) = classical_result(m)
```

Structural posture may evolve.
Classical magnitude may not be altered.

4.1 Structural Inversion Operator (`Inv_s`)

Structural inversion governs deterministic polarity posture reversal.

Define:

```
Inv_s(Z0) = Z0
Inv_s(Eplus) = Eminus
Inv_s(Eminus) = Eplus
Inv_s(S) = S
```

Properties:

• Closed on `A4`
• Deterministic
• Magnitude-preserving
• Symmetric over transitional regimes

Inversion operates on structural posture only.
It does not invert classical magnitude directly.

Collapse preservation:

```
phi_E((m, Inv_s(a))) = m
```

4.2 Structural Series Coupling Operator (`Series_s`)

When two electrodynamic segments interact structurally, SSSL defines deterministic compositional posture.

Let `a, b ∈ A4`.

Define:

`Series_s(a,b)` according to deterministic dominance rules:

1. If `a = Eminus` OR `b = Eminus` → `Eminus`
2. If `a = S` AND `b = S` → `S`
3. If one is `Eplus` and the other `S` → `Eplus`
4. If either is `Z0` and the other transitional → transitional state
5. Otherwise → `Eplus`

This operator is:

• Deterministic
• Closed
• Conservative
• Replay-verifiable

Series composition governs posture interaction, not voltage arithmetic.

Classical magnitude computation (e.g., Kirchhoff laws) remains untouched.

Collapse preservation condition:

If `m_series` is classical series magnitude, then

```
phi_E((m_series, Series_s(a,b))) = m_series
```

4.3 Structural Parallel Coupling Operator (`Parallel_s`)

Define deterministic structural parallel posture rule:

```
Parallel_s(a,b)
```

Dominance principles:

1. If either branch structurally collapses (`Eminus`) → discharge dominates
2. If both stable (`S`) → `S`
3. If mixed transitional → transitional dominance
4. If both neutral → `Z0`

Parallel posture reflects structural coherence, not current summation.

Magnitude remains governed by classical circuit law.

4.4 Operator Closure Principle

For all `a,b ∈ A4`:

```
Series_s(a,b) ∈ A4
Parallel_s(a,b) ∈ A4
Inv_s(a) ∈ A4
```

Therefore:

`A4` is closed under defined operators.

This closure elevates `A4` from a label set to a **finite structural electrodynamic algebra**.

4.5 Operator Determinism and Replay Identity

For fixed operator definitions and identical inputs:

```
B_A = B_B
```

Operator evaluation must be byte-identical across executions.

No adaptive dominance.
No fuzzy weighting.
No probabilistic blending.

Structural operator outcomes must remain deterministic functions of operands.

4.6 Substrate Elevation Condition

SSSL functions as a structural substrate layer when:

- `|A4| = 4`

- Structural mapping `F` governs deterministic regime assignment
- Transition operators remain closed over `A4`
- Conservative collapse invariant holds: `phi((m,a,s)) = m`
- Replay identity holds: `B_A = B_B`

Under these conditions:

- Regime space is finite
- Structural evolution is deterministic
- Vocabulary expansion is prohibited
- Magnitude remains unchanged

SSSL then governs **finite structural regime composition**, not observational labeling.

---

5. Structural Electrodynamic Admissibility Layer (`adm_E`)

A deterministic structural substrate must not only evolve and compose posture — it must govern **when electrodynamic traces are structurally admissible for reliance**.

SSSL therefore introduces a deterministic admissibility function:

```
adm_E : Trace -> {ALLOW, ABSTAIN}
```

This layer operates over structural posture sequences, not magnitude modification.

It governs structural reliability, not physical voltage.

5.1 Admissibility Principle

Let a structural trace be:

```
T = { (m_i, a_i) | i = 1..N }
```

SSSL evaluates structural posture stability over a finite window or full trace.

Admissibility must satisfy:

• Deterministic evaluation
• No probabilistic modeling
• No smoothing
• No predictive inference
• No modification of magnitude

Admissibility does not alter `m_i`.
It governs structural reliance posture.

5.2 Deterministic Admissibility Conditions (Illustrative)

Let

`N` = total observations
`count(Eminus)` = number of discharge states
`count(Eplus)` = number of accumulation states
`L(S)` = average dwell length of stable state
`Churn` = number of state transitions

Example deterministic rules:

1. **Excessive Collapse Condition**

If

```
count(Eminus) / N > threshold_collapse
```

→ `adm_E(T) = ABSTAIN`

2. **Stability Absence Condition**

If sustained magnitude exists but

```
count(S) = 0
```

→ `adm_E(T) = ABSTAIN`

3. **Oscillation Instability Condition**

If

```
Churn / N > threshold_churn
```

→ `adm_E(T) = ABSTAIN`

Otherwise:

```
adm_E(T) = ALLOW
```

All thresholds must be explicitly published and fixed within the execution scope.

No adaptive tuning is permitted during evaluation.

5.3 Admissibility Collapse Invariant

For all traces `T`:

If

```
adm_E(T) = ALLOW
```

or

```
adm_E(T) = ABSTAIN
```

Then magnitude remains preserved:

```
phi_E((m_i, a_i)) = m_i
```

Admissibility governs reliance only.
It never alters classical magnitude.

5.4 Relationship to Structural Evolution

Structural evolution (`delta_E`) governs posture transitions.

Structural admissibility (`adm_E`) governs trace-level structural reliability.

These layers are distinct:

• `delta_E` operates per observation
• `adm_E` operates over structured trace segments

Neither modifies magnitude.

5.5 Determinism Requirement

For identical trace inputs and identical admissibility thresholds:

```
adm_E(T_A) = adm_E(T_B)
```

and replay condition holds:

```
B_A = B_B
```

Admissibility evaluation must be:

• Deterministic
• Replay-verifiable

• Artifact-sealed
• Free of nondeterminism

5.6 Substrate Elevation Through Governance

With structural admissibility introduced, SSSL extends beyond regime observation into **deterministic structural reliance discipline**.

SSSL now:

- Evolves posture deterministically through `F`
- Composes posture algebraically over `A4`
- Governs structural reliance via `adm_E(T) ∈ {ALLOW, ABSTAIN}`
- Preserves magnitude exactly through `phi((m,a,s)) = m`

Magnitude describes measured state.
Posture describes structural regime state.
Admissibility governs structural reliance only.

These layers operate in parallel without interference.

SSSL remains a **deterministic structural substrate** over magnitude evolution.

---

6. SSUM-Compatible Structural Accumulation (`(m,a,s)` Extension)

To complete substrate elevation, SSSL extends the structural pair

```
X_i = (m_i, a_i)
```

into a structured electrodynamic state:

```
X_i = (m_i, a_i, s_i)
```

Where:

• `m_i` = classical magnitude (unchanged)
• `a_i ∈ A4` = structural posture
• `s_i` = deterministic structural accumulation metric

This extension integrates SSSL into the unified Shunyaya structural state architecture while preserving conservative collapse.

6.1 Structural Accumulation Principle

Electrodynamic evolution is not only posture transition; it accumulates structural strain, pressure, or stabilization over time.

Define `s_i` as a deterministic accumulation function:

```
s_i = Acc_s(s_{i-1}, a_i)
```

Where `Acc_s` must satisfy:

• Deterministic update rule
• No stochastic weighting
• No hidden variables
• Bounded accumulation
• Replay-verifiable identity

6.2 Example Deterministic Accumulation Rule (Illustrative)

One valid deterministic formulation:

If `a_i = Eminus`
→ `s_i = s_{i-1} + 1`

If `a_i = S` and sustained for `k` steps
→ `s_i = max(0, s_{i-1} - 1)`

If `a_i = Eplus`
→ `s_i = s_{i-1}`

If `a_i = Z0`
→ `s_i = 0`

Boundedness condition:

```
0 <= s_i <= S_max
```

All constants must be fixed and declared.

No adaptive decay is permitted.

6.3 Structural Strain Interpretation

`s_i` represents deterministic structural strain accumulation.

It does not represent:

• Energy
• Temperature
• Physical degradation
• Probability

It represents structural electrodynamic posture pressure derived solely from deterministic regime evolution.

6.4 Collapse Preservation Under Accumulation

Extended structural state:

```
X_i = (m_i, a_i, s_i)
```

Collapse operator remains:

```
phi((m_i, a_i, s_i)) = m_i
```

Therefore:

• Accumulation never alters magnitude
• Structural posture never alters magnitude
• Substrate remains conservative

This invariant is non-negotiable.

6.5 Accumulation Determinism and Replay Discipline

For fixed initial condition `s_0` and fixed rules:

Given identical inputs:

```
B_A = B_B
```

The sequence `{s_i}` must be byte-identical across executions.

No nondeterministic accumulation is permitted.

6.6 Substrate Implication

With `(m,a,s)` defined, SSSL operates across three deterministic layers:

1. Magnitude Layer (`m`)
Governed by classical domain equations.

2. Structural Posture Layer (`a`)
Governed by finite algebra `A4` and deterministic mapping `F`.

3. Structural Accumulation Layer (`s`)
Governed by a deterministic accumulation operator.

All three layers preserve the conservative collapse invariant:

```
phi((m,a,s)) = m
```

Magnitude remains unchanged.
Structure becomes finite and explicit.
Execution remains deterministic.

SSSL functions as a **multi-layer deterministic structural substrate**, not a classification mechanism.

6.7 Civilizational Constraint

SSSL qualifies as civilization-grade substrate only if:

- `A4` remains finite
- Operators remain closed
- `delta_E` remains deterministic
- `adm_E` remains deterministic
- Accumulation remains bounded
- Collapse invariant holds exactly
- Replay identity holds (`B_A = B_B`)

Failure of any condition invalidates conformance.

6.8 Structural Substrate Contract (SSC-E)

To elevate SSSL from deterministic framework to civilization-grade physical substrate discipline, this section formalizes the **Structural Substrate Contract**.

The Structural Substrate Contract defines the non-negotiable invariants that must hold for any conformant SSSL implementation.

This contract is machine-checkable and audit-verifiable.

6.8.1 Invariant Core

An implementation satisfies the Structural Substrate Contract if and only if all of the following hold simultaneously:

1. Finite Alphabet Invariance

```
A4 = {Z0, Eplus, S, Eminus}
|A4| = 4
```

2. Deterministic Evolution

```
a_i = F(E_i, dE_i, discharge_i)
```

Mapping must be total and single-valued.

3. Conservative Collapse Invariance

```
phi((m_i, a_i, s_i)) = m_i
```

4. Replay Identity

```
B_A = B_B
```

5. Spectral Boundedness

`rho(P) <= 1`
`|lambda_k| <= 1` for all eigenvalues

6. Accumulation Boundedness

```
0 <= s_i <= S_max
```

If any single invariant fails, the Structural Substrate Contract is violated.

6.8.2 Prohibited Extensions

The following behaviors immediately void substrate conformance:

- Runtime alphabet expansion
- Probabilistic blending
- Adaptive threshold tuning
- Machine-learned parameter injection
- Confidence scoring
- Soft-state classification

- Collapse modification
- Floating tolerance replay matching

SSSL is binary in conformance:
It either satisfies all invariants or forfeits substrate status.

6.8.3 Substrate Contract Implication

The Structural Substrate Contract establishes SSSL as a **deterministic structural substrate discipline**.

It formalizes:

- Finite regime closure: `|A4| = 4`
- Deterministic mapping `F`
- Conservative collapse: `phi((m,a,s)) = m`
- Spectral boundedness: `rho(P) ≤ 1`
- Replay identity: `B_A = B_B`

SSSL is not a classifier.
It is a finite, executable structural discipline over magnitude evolution.

The contract is not theoretical.
It is **deterministic, executable, and replay-verifiable**.

---

7. Determinism, Replay Discipline, and Conformance Identity

A structural substrate qualifies as civilization-grade only if its execution is reproducible without ambiguity.

SSSL therefore mandates strict deterministic replay discipline.

7.1 Deterministic Execution Rule

Given identical:

• Input dataset `D`
• Threshold constants `tau0, taus, eps, drop`
• Accumulation parameters
• Operator definitions
• Script version

Two independent executions must produce:

• Identical structural posture sequence `{a_i}`
• Identical accumulation sequence `{s_i}`
• Identical admissibility result `adm_E(T)`
• Identical output files
• Identical `MANIFEST.sha256`

Replay condition:

```
B_A = B_B
```

Byte identity is mandatory.

There is:

• No tolerance
• No approximate equality
• No statistical equivalence
• No floating comparison window

Either outputs are byte-identical, or conformance fails.

7.2 Artifact Requirements

A conformant SSSL implementation must produce:

1. `sssl_states.csv`
2. `summary.txt`
3. `MANIFEST.sha256`

If accumulation is enabled:

4. `sssl_accumulation.csv`

All files must be deterministic and reproducible.

7.3 Manifest Identity Rule

`MANIFEST.sha256` must include SHA-256 hashes of all produced artifacts.

Hash values must be computed over exact file bytes.

Replay validation succeeds if and only if:

```
SHA256_A(MANIFEST) = SHA256_B(MANIFEST)
```

If hash differs:

`B_A ≠ B_B`

→ Non-conformant.

7.4 Parameter Disclosure Requirement

All conforming implementations must explicitly publish:

- `tau0`
- `taus`
- `eps`
- `drop`
- Accumulation bounds
- Admissibility thresholds

Undeclared parameters invalidate reproducibility.

Adaptive runtime tuning invalidates conformance.

7.5 Structural Completeness Condition

For dataset `D`:

For every index `i`:

`∃! a_i ∈ A4`

Exactly one structural posture per observation.

No undefined states permitted.
No multi-label ambiguity permitted.

7.6 Finite Regime Stability Condition

Let

`Sigma = {a_i}`

Then

`Sigma ⊆ A4`

and

`|A4| = 4`

If any execution produces a fifth structural regime, conformance fails.

7.7 Collapse Preservation Condition

For every extended structural state:

`phi((m_i, a_i, s_i)) = m_i`

At no point may SSSL alter classical magnitude.

Violation of collapse preservation invalidates substrate status.

7.8 Civilizational Qualification Statement

SSSL qualifies as a civilization-grade structural electrodynamic substrate only if:

• Finite structural alphabet is maintained
• Operators remain closed
• Evolution operator remains deterministic
• Accumulation remains bounded and deterministic
• Admissibility remains deterministic
• Collapse invariant holds exactly
• Replay identity is verifiable

If any of these conditions fail, SSSL reduces to a heuristic layer and forfeits substrate status.

7.9 Substrate Integrity Summary

SSSL establishes:

- Deterministic structural regime evolution
- Closed finite regime algebra `A4`
- Deterministic structural accumulation `s`
- Structural admissibility discipline `adm_E(T) ∈ {ALLOW, ABSTAIN}`
- Conservative magnitude preservation `phi((m,a,s)) = m`
- Replay-verifiable execution identity `B_A = B_B`

Together, these properties define SSSL as a **deterministic structural substrate layer** operating alongside classical physics.

Magnitude remains governed by domain equations.
Structure remains finite, deterministic, and replay-verifiable.

---

8. Structural Spectral Stability and Boundedness

A deterministic structural substrate must demonstrate bounded regime evolution.
SSSL formalizes boundedness through deterministic spectral stability analysis of its transition structure.

8.1 Deterministic Transition Matrix Construction

Let the structural sequence be:

```
{a_i | i = 1..N}
```

Define deterministic transition counts:

```
T_count(a,b) = |{ i | a_i = a AND a_{i+1} = b }|
```

Where:

```
a, b ∈ A4
```

Define row totals:

```
RowSum(a) = Σ_b T_count(a,b)
```

Define deterministic transition ratio matrix:

```
R(a,b) = T_count(a,b) / RowSum(a)
```

This matrix is:

- Deterministically derived
- Dataset-dependent
- Replay-identical under `B_A = B_B`
- Free of stochastic modeling

No probability interpretation is implied.

8.2 Structural Transition Matrix `P`

Let `P` be the ordered matrix representation of `R(a,b)`
with row order `[Z0, Eplus, S, Eminus]`.

Then:

• Each row sums to 1
• No entry lies outside `[0,1]`
• No undefined transitions exist
• `P` is finite and bounded

`P` encodes structural regime evolution discipline, not physical law.

8.3 Spectral Radius Condition

Let `lambda` denote the eigenvalues of `P`.

Spectral stability condition:

```
rho(P) = max(|lambda|) <= 1
```

Civilizational requirement:

`|lambda| <= 1` for all eigenvalues.

This guarantees:

- No structural regime explosion
- No divergence beyond defined alphabet
- Finite-state boundedness
- Deterministic oscillatory containment

Spectral boundedness reflects structural closure, not thermodynamic equilibrium.

8.4 Non-Absorbing State Principle

Under sustained excitation and deterministic thresholds:

• No state in `A4` is permanently absorbing
• Structural cycles must emerge
• Neutral state `Z0` acts as quiescent fixed point

This guarantees structural recurrence within finite algebra.

8.5 Replay Spectral Identity Requirement

For identical dataset `D` and identical parameters:

• Transition matrix `P_A = P_B`
• Eigen-spectrum `lambda_A = lambda_B`

Replay condition:

`B_A = B_B`

If spectral results differ under identical inputs, conformance fails.

8.6 Substrate Stability Statement

SSSL spectral boundedness ensures that:

• Structural posture remains confined to finite regime algebra
• Evolution is dynamically closed
• No emergent undefined regimes appear
• Structural oscillations remain bounded

Magnitude continues to evolve under classical physics.
Structural evolution remains algebraically bounded.

This separation establishes deterministic structural electrodynamic stability without modifying physical equations.

8.7 Civilizational Spectral Qualification

SSSL satisfies spectral substrate conditions only if:

- `|A4| = 4`
- Transition matrix `P ∈ R^(4x4)` is finite
- `rho(P) <= 1`
- `|lambda| <= 1` for all eigenvalues
- Replay identity holds: `B_A = B_B`

Failure of any condition invalidates substrate-level stability.

8.8 Structural Energy Neutrality Principle

A structural substrate operating alongside physical law must demonstrate non-interference beyond algebraic boundedness.

SSSL therefore satisfies the **Structural Energy Neutrality Principle**.

8.8.1 Definition

For every observation index i:

```
X_i = (m_i, a_i, s_i)
```

Structural components `(a_i, s_i)` must satisfy:

`d/dt (m_i)` is independent of `a_i` and `s_i`.

That is:

Structural posture and accumulation must never influence classical magnitude evolution.

8.8.2 Neutrality Condition

Let classical magnitude evolution be:

```
m_{i+1} = G(m_i, inputs_i)
```

SSSL must satisfy:

`m_{i+1}` does not depend on `a_i`
`m_{i+1}` does not depend on `s_i`

Thus structural layer is observationally orthogonal.

8.8.3 Substrate Implication

This neutrality ensures:

- No energy injection
- No artificial damping
- No structural amplification
- No hidden feedback

SSSL remains a **pure structural overlay**.

Without this principle, substrate claims would be invalid.

---

9. Structural Charge Cycle Theorem (SSSL-CCT)

A finite deterministic structural electrodynamic substrate must exhibit bounded recurrence under sustained excitation.

SSSL formalizes this through the **Structural Charge Cycle Theorem (SSSL-CCT)**.

9.1 Theorem Statement

Given:

1. A finite structural alphabet
   `A4 = {Z0, Eplus, S, Eminus}`
2. Deterministic evolution operator
   `a_i = delta_E(a_{i-1}, obs_i)`
3. Non-zero excitation over a sufficiently long interval
4. Explicit discharge threshold condition

Then any sufficiently long observable charge trace must contain at least one structural recurrence cycle of the form:

```
Eplus -> S -> Eminus -> Eplus
```

or a deterministic subcycle thereof.

9.2 Deterministic Proof Sketch

1. If magnitude increases beyond stability band → `Eplus`
2. If magnitude stabilizes within tolerance band → `S`
3. If discharge flag or deterministic drop condition triggers → `Eminus`
4. If magnitude remains non-zero post-discharge → accumulation resumes → `Eplus`

Because:

• `delta_E` is total
• `A4` is finite
• No absorbing state exists under sustained excitation
• Discharge threshold is explicit

By finite-state recurrence principle:

A deterministic system over a finite alphabet under sustained excitation must eventually revisit prior states.

Therefore, structural recurrence is guaranteed.

No probabilistic reasoning is required.

9.3 Neutral Fixed-Point Condition

If excitation ceases entirely and magnitude falls within neutral threshold:

```
a_i = Z0
```

Thus:

• `Z0` is the structural quiescent fixed point
• All other regimes participate in recurrence dynamics

`Z0` represents structural equilibrium, not energy annihilation.

9.4 Non-Absorbing State Property

Under non-zero excitation:

• `Eplus` cannot persist indefinitely without either stabilizing or collapsing
• `S` cannot persist indefinitely without potential discharge
• `Eminus` cannot persist indefinitely without accumulation restart

Therefore:

No transitional regime is permanently absorbing.

Structural cycling is intrinsic to deterministic electrodynamic posture evolution.

9.5 Formal Cycle Condition

Let

```
C = (Eplus, S, Eminus, Eplus)
```

Then there exist indices

```
i < j < k < l
```

such that:

```
a_i = Eplus
a_j = S
a_k = Eminus
a_l = Eplus
```

Under sustained excitation, at least one such tuple must occur.

9.6 Collapse Preservation Under Recurrence

For all indices `i`:

```
phi((m_i, a_i)) = m_i
```

Recurrence affects structural posture only.
Classical magnitude remains governed by electrodynamics.

SSSL reveals lifecycle structure without redefining physical law.

9.7 Substrate Implication

The Structural Charge Cycle Theorem establishes that:

- Electrodynamic posture evolution is inherently cyclic under sustained excitation
- Structural recurrence is bounded within finite algebra
- Substrate behavior is deterministic and closed
- No undefined regimes emerge

This theorem strengthens SSSL's status as a structural electrodynamic substrate rather than a classification mechanism.

9.8 Deterministic Cycle Witness Principle

The Structural Charge Cycle Theorem establishes recurrence existence.

This section formalizes recurrence detection as a deterministic witness.

9.8.1 Cycle Witness Definition

Define canonical structural cycle:

```
C = (Eplus, S, Eminus, Eplus)
```

A deterministic trace contains a cycle witness if:

There exist indices

```
i < j < k < l
```

such that:

```
a_i = Eplus
a_j = S
a_k = Eminus
a_l = Eplus
```

9.8.2 Witness Properties

- Deterministic
- Position-indexed
- Replay-verifiable
- Domain-independent

The cycle witness does not imply physical oscillation.
It implies structural recurrence under finite regime algebra.

9.8.3 Substrate Elevation

By elevating recurrence from theoretical claim to deterministic witness artifact, SSSL strengthens:

- Structural boundedness verification
- Finite regime closure confirmation
- Non-absorbing regime discipline

Recurrence becomes verifiable — not assumed.

---

10. Conformance Specification (SSSL-CS)

This section defines the executable boundary of the Shunyaya Structural Substrate Layer (SSSL).

An implementation is conformant only if all conditions below are satisfied without exception.

10.1 Input Schema Requirements

A conformant implementation must accept input in strict CSV format with header:

```
t_s,E_proxy,discharge
```

Constraints:

- Case-sensitive header
- Column order preserved
- No hidden columns
- No dynamic schema interpretation
- No auto-reordering
- No inferred fields

If schema deviates → input must be rejected.

10.2 Deterministic Parameter Declaration

All implementations must explicitly declare:

- `tau0`
- `taus`
- `eps`
- `drop`
- Initial accumulation value `s_0`
- Accumulation bounds `S_max`
- Admissibility thresholds

Undeclared or runtime-adaptive parameters invalidate conformance.

No machine-learned thresholds are permitted.

10.3 Required Output Artifacts

A conformant execution must produce:

1. `sssl_states.csv`
2. `summary.txt`
3. `MANIFEST.sha256`

If accumulation is enabled:

4. `sssl_accumulation.csv`

All files must be reproducible under replay discipline.

10.4 State Mapping Completeness

For dataset `D = {obs_i}`, the evolution function must satisfy:

```
∀ i, ∃! a_i ∈ A4
```

Exactly one structural state per observation.

No:

• Undefined states
• Dual-state assignments
• Fuzzy blending
• Confidence scores

Deterministic single assignment is mandatory.

10.5 Negative Control Validation

A conformant implementation must demonstrate deterministic behavior under the following synthetic controls:

• Constant trace → only `Z0` or `S`
• Strict monotonic increase → `Eplus` dominance
• Forced discharge flag → `Eminus` present
• Neutral decay without excitation → convergence to `Z0`

Failure of any control test → non-conformant.

10.6 Finite Alphabet Constraint

Structural alphabet must remain fixed:

```
A4 = {Z0, Eplus, S, Eminus}
```

Cardinality condition:

```
|A4| = 4
```

Introduction of any additional regime label invalidates conformance.

10.7 Deterministic Replay Requirement

Given identical:

• Input dataset
• Parameter set
• Script version

Two executions must produce byte-identical artifacts.

Replay identity condition:

```
B_A = B_B
```

If `SHA256_A(MANIFEST) ≠ SHA256_B(MANIFEST)`

→ Non-conformant.

No tolerance windows are permitted.

10.8 Collapse Invariance Condition

For every structural state:

```
phi((m_i, a_i, s_i)) = m_i
```

SSSL must never alter classical magnitude.

Violation invalidates substrate status.

10.9 Spectral Stability Condition

For transition matrix `P`:

```
rho(P) = 1
```
`|lambda| <= 1` for all eigenvalues

If spectral radius exceeds 1 under identical replay → non-conformant.

10.10 Conformance Verdict Rule

An implementation qualifies as SSSL-conformant only if:

• Input schema validated
• Parameters declared and fixed
• Finite alphabet preserved
• Deterministic evolution verified
• Accumulation bounded
• Admissibility deterministic
• Spectral boundedness satisfied
• Collapse invariant preserved
• Replay identity confirmed

Failure of any single criterion results in immediate non-conformance.

10.11 Substrate Integrity Statement

Conformance to SSSL-CS establishes:

- Deterministic structural regime evolution
- Algebraic closure over finite regime space `A4`
- Replay-verifiable execution discipline `B_A = B_B`
- Conservative magnitude preservation `phi((m,a,s)) = m`
- Explicit scope and misuse boundary

This specification defines the **executable structural substrate boundary** for SSSL.

Magnitude remains governed by classical equations.
Structural posture remains finite, deterministic, and bounded.

---

11. Non-Claims and Misuse Boundary

The Shunyaya Structural Substrate Layer (SSSL) is a deterministic structural substrate layer.
It is not a physical theory replacement, predictive engine, or control system.

This section defines strict misuse boundaries.

11.1 Non-Modification of Classical Electrodynamics

SSSL does not:

• Modify Maxwell's equations
• Redefine electric charge
• Alter voltage, current, or field magnitudes
• Introduce new force laws
• Replace circuit theory
• Replace plasma physics
• Replace electrochemistry

Classical electrodynamics remains fully intact.

SSSL overlays structural observability without altering physical equations.

11.2 Not a Predictive System

SSSL does not:

• Predict breakdown voltage
• Forecast discharge timing
• Estimate battery life
• Model arc formation
• Simulate electromagnetic propagation
• Perform forward-time projection

Structural recurrence theorems describe deterministic boundedness, not prediction.

No probabilistic inference is introduced.

11.3 Not a Control System

SSSL does not:

• Inject control signals
• Modify physical states
• Adjust thresholds dynamically
• Optimize system behavior
• Actuate hardware

It observes and classifies structural posture only.

Control system integration requires independent safety validation beyond SSSL scope.

11.4 Not a Degradation or Health Model

SSSL accumulation variable `s_i` represents structural posture strain only.

It does not represent:

• Material degradation
• Electrochemical aging
• Thermal damage
• Mechanical stress
• Probability of failure

Any interpretation beyond structural posture classification is misuse.

11.5 Not a Safety-Critical Certification

SSSL does not constitute:

• Regulatory approval
• Engineering certification
• Safety validation
• Hazard analysis
• Risk assessment

Civilization-grade status refers strictly to deterministic verification discipline.

Deployment in safety-critical environments requires independent certification and regulatory compliance.

11.6 No Probabilistic Claims

Transition ratio matrices and spectral properties are deterministic structural descriptors.

They are not:

• Probabilities
• Statistical estimators
• Bayesian priors
• Markov chain predictions
• Entropic measures

Any probabilistic reinterpretation invalidates structural framing.

11.7 No Adaptive or Machine-Learned Extension

SSSL prohibits:

• Dynamic threshold tuning
• Data-driven alphabet expansion
• Neural inference integration
• Parameter auto-optimization

Such additions convert SSSL into a heuristic classifier and void substrate conformance.

11.8 Collapse Invariance Boundary

At all times:

```
phi((m,a,s)) = m
```

Structural layers must never alter classical magnitude.

Any implementation that modifies magnitude forfeits SSSL designation.

11.9 Scope Integrity Statement

SSSL defines a deterministic structural electrodynamic substrate layer.

It introduces:

• Finite structural regime algebra
• Deterministic posture evolution
• Structural admissibility discipline
• Replay-verifiable execution

It does not redefine physical reality.

It reveals structural posture within it.

---

12. Civilizational Qualification Conditions

This section defines the formal conditions under which
the Shunyaya Structural Substrate Layer (SSSL)
qualifies as a **civilization-grade physical substrate layer**.

The term *civilization-grade* refers strictly to verification discipline, deterministic reproducibility, algebraic boundedness, and misuse containment.

It does not imply regulatory approval or deployment authorization.

12.1 Finite Alphabet Condition

The structural regime alphabet must remain fixed:

```
A4 = { Z0, Eplus, S, Eminus }
```

Cardinality constraint:

```
|A4| = 4
```

No runtime expansion is permitted.
No fifth regime may be introduced.

If the alphabet expands:
→ Non-conformant implementation.

12.2 Deterministic Transition Operator Condition

For fixed constants:

```
tau0, taus, eps, drop, kappa
```

and fixed dataset `D`,

the state sequence:

```
a_i = F(E_i, delta_E_i, d_i)
```

and accumulation sequence:

```
s_i = S_acc(a_i, s_{i-1})
```

must be uniquely determined.

No randomness.
No adaptive tuning.
No dataset-dependent threshold modification.

12.3 Structural Admissibility Condition

Collapse to structurally stable regimes must obey:

```
adm_E(a_i) ∈ {0,1}
```

where:

```
adm_E(Z0) = 1
adm_E(S) = 1
adm_E(Eplus) = 0
adm_E(Eminus) = 0
```

No Boolean projection may occur from inadmissible regimes.

12.4 Collapse Invariance Condition

For every structural tuple:

```
X_i = (m_i, a_i, s_i)
```

Collapse must satisfy:

```
phi(X_i) = m_i
```

Classical magnitude must remain unchanged under all structural overlays.

If magnitude changes:
→ SSSL designation is invalid.

12.5 Replay Identity Condition

Given identical:

- Input dataset
- Parameter constants
- Script version

Two executions must produce byte-identical artifacts:

```
B_A = B_B
```

Including:

- `sssl_states.csv`
- `summary.txt`
- `MANIFEST.sha256`

Any replay deviation:

```
B_A ≠ B_B
```

→ Non-conformant implementation.

12.6 Spectral Boundedness Condition

Let `P` be the deterministic transition ratio matrix.

The spectral radius must satisfy:

```
rho(P) = 1
```

and:

```
|lambda_i| <= 1
```

for all eigenvalues `lambda_i`.

No eigenvalue may exceed unity magnitude.

This guarantees structural boundedness of regime dynamics.

12.7 Accumulation Boundedness Condition

The accumulation variable `s_i` must satisfy:

```
s_i >= 0
```

and must not diverge under bounded excitation.

Under sustained neutral regime:

```
s_i → 0
```

Accumulation must reflect structural strain,
not uncontrolled growth.

12.8 Negative Control Requirement

A conformant implementation must demonstrate:

• Constant trace → only `Z0` or `S`
• Strict monotonic increase → `Eplus` dominance
• Forced discharge → `Eminus` presence
• Neutral cessation → convergence to `Z0`

Failure to demonstrate negative control invalidates conformance.

12.9 No-Probability Discipline

All ratios and spectral measures must be deterministic functions of dataset.

They must not be interpreted as:

• Probabilities
• Risk measures
• Forecasting tools
• Stochastic simulations

Probabilistic reinterpretation voids structural substrate designation.

12.10 Substrate Qualification Summary

SSSL qualifies as a civilization-grade structural substrate only if:

1. Finite fixed alphabet: `|A4| = 4`
2. Deterministic structural mapping `F`
3. Structural admissibility discipline `adm_E(T) ∈ {ALLOW, ABSTAIN}`
4. Conservative collapse invariance `phi((m,a,s)) = m`
5. Replay identity `B_A = B_B`
6. Spectral boundedness `rho(P) ≤ 1`
7. Deterministic accumulation boundedness `s_i ≥ 0`
8. Negative control validation
9. Explicit scope and misuse boundaries

All nine conditions must hold simultaneously.

Violation of any single condition invalidates civilization-grade designation.

12.11 Structural Abstention Principle

A civilization-grade substrate must include the right to abstain deterministically.

12.11.1 Abstention Definition

```
adm_E(T) ∈ {ALLOW, ABSTAIN}
```

ABSTAIN does not mean:

- Failure
- Error
- Instability
- Prediction of collapse

ABSTAIN means:

Trace lies outside declared deterministic admissibility envelope.

12.11.2 Deterministic Abstention Guarantee

Given fixed thresholds:

`adm_E(T_A) = adm_E(T_B)`
under replay condition
`B_A = B_B`

No probabilistic scoring may override abstention.

12.11.3 Physical Substrate Significance

Abstention is a structural safety valve.

It prevents:

- Over-interpretation
- Structural overreach
- False substrate confidence

`adm_E(T) ∈ {ALLOW, ABSTAIN}` ensures that structural reliance is declared explicitly and deterministically.

This principle reinforces SSSL as a **structurally disciplined substrate layer**, not an analytical authority.

Magnitude remains governed by classical physics.
Structure remains finite and bounded.
Reliance remains explicitly declared.

---

13. Structural Electrodynamic Conservative Extension Theorem

This section formalizes that the Shunyaya Structural Substrate Layer (SSSL) is a **conservative structural extension** of classical electrodynamics.

SSSL introduces structural regime topology without modifying classical magnitude behavior.

13.1 Objective

To prove that SSSL preserves classical electrodynamic magnitudes exactly, while introducing a deterministic structural regime layer.

The extension must satisfy:

• No alteration of classical values
• No alteration of governing equations
• No alteration of observable measurement
• Deterministic structural overlay only

13.2 Classical Magnitude Space

Let classical observable magnitude space be:

```
M = { m_i ∈ R }
```

Where:

```
m_i = E_proxy(t_i)
```

This represents measured voltage, charge proxy, or field magnitude.

Classical electrodynamics operates solely in `M`.

13.3 Structural Extension Space

Define structural alphabet:

```
A4 = { Z0, Eplus, S, Eminus }
```

Define accumulation space:

```
S_space = { s_i ∈ R | s_i >= 0 }
```

Define structural tuple space:

```
X = M × A4 × S_space
```

Each observation becomes:

```
X_i = (m_i, a_i, s_i)
```

This is the structural extension domain.

13.4 Collapse Operator Definition

Define projection (collapse) operator:

```
phi : X -> M
```

such that:

```
phi((m_i, a_i, s_i)) = m_i
```

The projection eliminates structural components
while preserving classical magnitude exactly.

13.5 Conservative Extension Property

SSSL is a conservative extension of classical electrodynamics if:

For every `m_i ∈ M`,

there exists `X_i ∈ X` such that:

```
phi(X_i) = m_i
```

and no structural operation modifies `m_i`.

Proof:

1. Structural mapping `F(E_i, delta_E_i, d_i)` depends only on classical observables.
2. Accumulation operator `S_acc(a_i, s_{i-1})` does not alter `m_i`.
3. Collapse operator `phi` returns `m_i` unchanged.

Thus classical magnitude is invariant under extension.

Therefore SSSL is conservative.

13.6 Non-Interference Condition

Let classical evolution function be:

```
m_{i+1} = G(m_i, inputs_i)
```

SSSL does not alter `G`.

Structural operators satisfy:

`m_{i+1}` is independent of `a_i` and `s_i`.

Thus structural layer is non-interfering.

13.7 Commutative Preservation Diagram

Let structural mapping be:

```
F : M -> X
```

Let collapse operator be:

```
phi : X -> M
```

Then:

```
phi(F(m_i)) = m_i
```

for all `m_i ∈ M`.

Thus the following diagram commutes:

```
M --F--> X --phi--> M
```

and equals identity on `M`.

Identity condition:

```
phi ∘ F = Id_M
```

13.8 No New Physical Claims

Because:

```
phi ∘ F = Id_M
```

SSSL introduces no new physical predictions.

It adds structural observability only.

Classical electrodynamics remains complete and unchanged.

13.9 Conservative Extension Theorem

Theorem (SSSL-CET)

Given:

1. Finite structural alphabet `A4`
2. Deterministic structural mapping `F`
3. Non-interfering accumulation operator
4. Collapse operator `phi((m,a,s)) = m`

Then SSSL is a conservative structural extension of classical electrodynamics.

Proof:

Follows directly from:

```
phi ∘ F = Id_M
```

and non-interference of structural variables.

Therefore classical magnitude space `M`
is preserved exactly.

13.10 Civilization-Grade Implication

Because SSSL satisfies conservative extension:

- It does not compete with physics.
- It does not replace Maxwell.
- It does not alter charge theory.
- It preserves measurement identity.

It qualifies as a structural substrate layer,
not an alternative physical theory.

---

14. Cross-Domain Regime Compression Principle

This section establishes that the Shunyaya Structural Substrate Layer (SSSL)
is not limited to a single device, dataset, or physical configuration.

It formalizes the principle that diverse electrodynamic systems
compress deterministically into a fixed finite structural alphabet
without altering classical laws.

14.1 Motivation

Electrodynamic phenomena appear across domains:

- Capacitor charging
- Battery discharge curves
- Power supply regulation
- Plasma ignition cycles
- Semiconductor switching
- Field oscillation systems

Despite physical differences, observable charge traces share structural posture patterns.

SSSL formalizes this structural compression.

14.2 Regime Compression Definition

Let each electrodynamic system produce a magnitude trace:

```
m_i ∈ M
```

Let structural mapping be:

```
F(m_i, delta_m_i, d_i) = a_i ∈ A4
```

Define regime compression function:

```
C : M^N -> A4^N
```

such that every trace is mapped into the fixed alphabet `A4`.

Compression property:

For all electrodynamic systems `S_k`,

```
C(S_k) ⊆ A4
```

and

```
|A4| = 4
```

No domain expands the alphabet.

14.3 Structural Universality Claim

If multiple physically distinct systems satisfy:

- Deterministic observability
- Bounded excitation
- Finite transition thresholds

Then their structural traces compress into:

```
A4 = { Z0, Eplus, S, Eminus }
```

This is structural universality, not physical identity.

The compression occurs at posture level,
not at equation level.

14.4 Algebraic Closure Across Domains

For any two electrodynamic systems `S_A` and `S_B`:

Let

```
Sigma_A = C(S_A)
Sigma_B = C(S_B)
```

Then:

```
Sigma_A ∪ Sigma_B ⊆ A4
```

Union does not increase alphabet size.

This guarantees structural closure under domain aggregation.

14.5 Deterministic Invariance Across Implementations

Given identical thresholds and mapping rules:

Two independent implementations must satisfy:

```
C_A(S_k) = C_B(S_k)
```

under replay condition:

```
B_A = B_B
```

Cross-machine reproducibility is mandatory.

14.6 No Cross-Domain Equation Blending

SSSL does not:

- Blend equations from different systems
- Introduce unified physical constants
- Merge electrochemical and electromagnetic models
- Construct cross-domain simulations

Compression is structural only.

Each domain retains its physical laws.

14.7 Substrate-Level Unification Principle

Physical systems differ in governing equations.
They converge in structural regime grammar.

This produces a substrate-level invariant:

Finite structural alphabet under deterministic mapping:

```
A4 = {Z0, Eplus, S, Eminus}
|A4| = 4
```

Across domains, structural posture is governed by the same finite algebra `A4` and deterministic mapping `F`.

This reflects a broader unification discipline:

Finite regime grammar over diverse deterministic domains.

Physics remains domain-specific.
Structure remains finite and shared.

14.8 Civilization-Grade Implication

SSSL demonstrates:

• Structural regime compression
• Finite bounded alphabet
• Deterministic replay identity
• Conservative physical extension
• Cross-domain structural closure

Thus SSSL qualifies as a structural electrodynamic substrate layer
with cross-domain applicability,
while preserving strict physical non-interference.

14.9 Universal Substrate Interface Condition

SSSL operates over scalar magnitude traces independent of domain physics.

To formalize this, define the Universal Substrate Interface (USI).

14.9.1 Interface Definition

A domain D is SSSL-compatible if it can produce:

`t_s` = ordered observation index
`E_proxy` = scalar magnitude
`discharge` ∈ {0,1}

No additional domain parameters are required.

14.9.2 Structural Mapping Sufficiency

If domain D can produce:

```
obs_i = (E_i, dE_i, d_i)
```

where:

```
dE_i = E_i - E_{i-1}
```

then SSSL mapping:

```
a_i = F(E_i, dE_i, d_i)
```

is sufficient to construct structural substrate evolution.

The mapping is:

- Deterministic
- Total
- Single-valued
- Closed over `A4`

No domain equation blending is required.

14.9.3 Substrate Implication

This interface guarantees:

- Cross-domain structural compression
- Finite regime invariance
- Replay-stable mapping
- Algebraic universality

SSSL therefore operates as a **domain-agnostic deterministic structural magnitude substrate.**

14.10 Final Definition

SSSL is:

A deterministic structural substrate layer
that compresses observable scalar magnitude evolution
into a finite regime alphabet
while preserving classical magnitude exactly
under conservative collapse invariance:

```
phi((m,a,s)) = m
```

No physical law is modified.
No probabilistic inference is introduced.
No structural vocabulary expansion is permitted.

The alphabet remains finite: `|A4| = 4`
The mapping remains deterministic: `a_i = F(E_i, dE_i, discharge_i)`
The substrate remains conservative.

Appendix A — Replay Identity Proof (Civilization-Grade Determinism)

Objective
Demonstrate that SSSL substrate execution is deterministic and replay-identical.
For identical inputs, parameters, and implementation version, two independent executions must produce byte-identical sealed artifacts.

Replay Identity Requirement
`B_A = B_B`

Where:

- `B_A` is the complete output artifact bundle from Execution A
- `B_B` is the complete output artifact bundle from Execution B
- Identity is evaluated by exact file bytes, sealed by SHA256 manifest

---

A.1 Two Executions (Core Substrate Replay)

Run A

```
python scripts\sssl_verify.py --in_csv data\sssl_smoke.csv --out_dir
outputs\RUN1 --substrate
```

Run B

```
python scripts\sssl_verify.py --in_csv data\sssl_smoke.csv --out_dir
outputs\RUN2 --substrate
```

Byte-Level Manifest Comparison

```
fc /b outputs\RUN1\MANIFEST.sha256 outputs\RUN2\MANIFEST.sha256
```

Result

```
FC: no differences encountered
```

Therefore:

`outputs\RUN1\MANIFEST.sha256` is byte-identical to
`outputs\RUN2\MANIFEST.sha256`

Thus:

```
B_A = B_B
```

Implication

SSSL substrate execution is deterministic under replay discipline, producing sealed artifacts that are reproducible exactly across repeated runs.

A.2 Verify Capsule Proof (Auditor-Grade Determinism)

Objective
Demonstrate that an auditor can execute a single harness that enforces:

- replay identity `B_A = B_B`
- fixed alphabet invariants `A4`
- collapse conservation `phi((m,a,s)) = m`
- spectral boundedness `rho(P) = 1`
- admissibility safety semantics (ALLOW / ABSTAIN)

Capsule Run Command (Windows)
From `VERIFY_SSSL_CAPSULE` directory:
`RUN_VERIFY.bat`

Expected Final Line
`CAPSULE_RESULT: PASS`

Capsule Evidence Artifact
`VERIFY_SSSL_CAPSULE\CAPSULE_SUMMARY.txt`

This file records the enforced invariants and final verdict.

A.3 Capsule Summary Hash (Recommended)

To seal the capsule verdict as a stable public fingerprint, compute and record the SHA256 hash of:

`VERIFY_SSSL_CAPSULE\CAPSULE_SUMMARY.txt`

Windows Command

```
certutil -hashfile VERIFY_SSSL_CAPSULE\CAPSULE_SUMMARY.txt SHA256
```

Record

`CAPSULE_SUMMARY_SHA256 =`
dd0e66d50e0f33d8e5436a93e67718657c0dc8f35368d684d5866a7651a6844f

This hash seals the auditor-facing verdict artifact.
It depends only on the deterministic capsule execution and not on any redistributed external dataset.

A.4 Final Determinism Claim

The system satisfies civilization-grade determinism on two levels:

1. Core substrate replay identity: `B_A = B_B` sealed by `MANIFEST.sha256`
2. Auditor capsule determinism: `CAPSULE_RESULT: PASS` sealed by `CAPSULE_SUMMARY_SHA256`

Therefore, SSSL qualifies as an execution-first deterministic structural substrate standard under replay discipline.

Appendix B — Artifact Semantic Integrity Check (Non-Duplication)

Purpose
Ensure that SSSL produces distinct artifacts for:

- the compact transition ratio matrix used for spectral analysis, and
- the long-form audit table used for explicit edge verification.

Required Distinction
`P_matrix.csv` and `transition_ratios.csv` must not be byte-identical.

Definitions:

- `P_matrix.csv` = 4×4 compact transition ratio matrix over `A4`

- `transition_ratios.csv` = long-form audit table with explicit edges (`from,to,count,rowsum_from,ratio`)

Check
`fc /b outputs\RUN1\P_matrix.csv outputs\RUN1\transition_ratios.csv`

Result
Files differ; `transition_ratios.csv` is longer than `P_matrix.csv`.

Implication
Transition artifacts are semantically distinct and audit-valid:

- compact matrix supports deterministic spectral boundedness
- long-form ratios support deterministic transition edge auditing

---

Appendix C — Electrodynamics Substrate Verification (Battery Dataset)

Dataset Execution Context
External lithium-ion discharge observation set
Observations: **220**
State space: `A4 = {Z0, Eplus, S, Eminus}`
Conservative extension: `phi((m,a,s)) = m`

Deterministic parameters (explicitly published):
```
tau0 = 0.05
taus = 0.7
eps = 0.02
drop = 0.15
```

C.1 Admissibility Verdict

Artifact: `adm_result.txt`

SSSL admissibility verdict
`adm_E: ALLOW`

Deterministic Metrics:
• avg_dwell_S = 1.0
• churn_ratio = 0.6181818181818182
• collapse_ratio = 0.33636363636363636
• count_S = 11.0

Interpretation

• Structural stability state `S` appears deterministically.
• Collapse frequency remains bounded under declared thresholds.
• Churn remains within configured admissibility limits.
• The substrate layer permits collapse under declared admissibility parameters.

No probabilistic inference was introduced.
No physical modeling was altered.
This is purely structural admissibility governance.

C.2 Deterministic Spectral Stability

Artifact: `eigenspectrum.txt`

Matrix order: `[Z0, Eplus, S, Eminus]`

Eigenvalues (deterministic QR iteration):
0.000000000000
1.000000000000
-0.253289650137
-0.048646614486

Spectral radius estimate:
`rho(P) = 1.000000000000`

C.3 Structural Interpretation

1. **Dominant eigenvalue = 1.0**
   → Structural closure holds
   → No regime expansion
   → Finite alphabet remains invariant
2. **All `|lambda| <= 1`**
   → System is spectrally bounded
   → No structural instability amplification
3. **Zero eigenvalue present**
   → Indicates a terminating structural subspace component
   → No absorbing runaway regime is introduced

This confirms **deterministic structural boundedness** of the transition operator.

No stochastic equilibrium claim is made.
No thermodynamic reinterpretation is introduced.
This is structural stability verification only.

C.4 Civilization-Grade Conditions Satisfied

SSSL substrate execution satisfies:

1. **Finite alphabet** `|A4| = 4`
2. **Deterministic mapping rules**
3. **Published parameter set**
4. **Replay identity discipline** `B_A = B_B`
5. **Distinct artifact semantics**
6. **Spectral boundedness** `rho(P) = 1`
7. **Collapse invariant preserved:** `phi((m,a,s)) = m`

Therefore:

The SSSL substrate layer qualifies as **deterministic structural governance under replay discipline**.

C.5 Dataset Attribution and License

Dataset Name: Lithium-Ion Battery Degradation Dataset
Source: Public research dataset (battery cycle aging data)
Typical Fields Used: `battery_id`, `cycle`, `disV`, `disI`

This dataset is publicly distributed for academic and research purposes.
Users must comply with the original dataset license terms provided by the dataset host (e.g., Kaggle, research repository, or institutional archive).

SSSL uses the dataset strictly for:
• Deterministic structural regime classification
• Transition observability analysis
• Replay-verifiable artifact generation

No predictive modeling was performed.
No battery health inference was claimed.
No electrochemical simulation was introduced.

This execution constitutes **structural substrate verification only**.

Recommended Citation (Generic Research Format)

If the dataset originates from the commonly distributed lithium-ion degradation dataset associated with early-cycle prediction research, citation should follow:

Severson, K. A., Attia, P. M., Jin, N., Perkins, N., Jiang, B., Yang, Z., Chen, M., Aykol, M., Herring, P. K., Fraggedakis, D., et al.
"Data-driven prediction of battery cycle life before capacity degradation."
Nature Energy, 4, 383–391 (2019).

Dataset Handling Notice

No third-party datasets are redistributed with this work.

All external datasets referenced (e.g., lithium-ion battery degradation datasets) must be obtained directly from their original licensed sources.

Only deterministic structural artifacts, transition operators, and replay-verifiable summaries derived from execution are included.

Users are responsible for complying with the original dataset license terms when reproducing experiments.

For Appendices D, E1, and E2, traces are generated deterministically for structural validation; no third-party datasets are used.

Appendix D — Domain-Agnostic Universal Trace Demonstration (Smoke Lifecycle)

Dataset Execution Context

Domain-agnostic magnitude trace (deterministic smoke lifecycle)
Observations: **25**
State space: `A4 = {Z0, Eplus, S, Eminus}`
Conservative extension: `phi((m,a,s)) = m`

This trace is **not an electrodynamics dataset**.
It is a generic scalar magnitude lifecycle (accumulation → stability → drop → re-accumulation) validated under the same substrate algebra.

No probabilistic inference was introduced.
No physical law was modified.
This is structural substrate validation only.

D.1 Admissibility Verdict

Artifact: `adm_result.txt`

SSSL admissibility verdict
`adm_E: ALLOW`

Deterministic Metrics:
• avg_dwell_S = 5.0
• churn_ratio = 0.16
• collapse_ratio = 0.04
• count_S = 5.0

Interpretation

• Stability state `S` appears deterministically with sustained dwell.
• Collapse frequency remains low and bounded.
• Churn remains bounded well within admissibility discipline.
• Substrate collapse permission is granted under published thresholds.

No probabilistic inference was introduced.
No domain modeling was introduced.
This is structural admissibility governance only.

D.2 Deterministic Spectral Stability

Artifact: `eigenspectrum.txt`
Matrix order: `[Z0, Eplus, S, Eminus]`

Eigenvalues (deterministic QR iteration):

```
0.000000000000
1.000000000000
0.724948128984
0.016228341604
```

Spectral radius estimate:

```
rho(P) <= 1.000000000000
```

All eigenvalues satisfy:

```
|lambda| <= 1
```

D.3 Structural Interpretation

1. **Dominant eigenvalue = 1.0**
   → Structural closure holds
   → No regime expansion
   → Finite alphabet remains invariant
2. **All `|lambda| <= 1`**
   → System is spectrally bounded
   → No structural instability amplification
3. **Sub-dominant eigenvalue ≈ 0.7249**
   → Indicates persistent but bounded regime cycling
   → Encodes strong accumulation–stability coherence under the finite regime operator
4. **Near-zero eigenvalue present**
   → Indicates a terminating structural subspace component
   → No absorbing runaway regime is introduced

No stochastic equilibrium claim is made.
No thermodynamic reinterpretation is introduced.
This is structural stability verification only.

D.4 Replay Identity Proof (Civilization-Grade Discipline)

Two independent executions over identical input produced **byte-identical artifacts**.

Replay condition: `B_A = B_B`

Verified comparisons:

• `fc /b outputs\RUN1\MANIFEST.sha256 outputs\RUN2\MANIFEST.sha256`
→ No differences encountered

• `fc /b outputs\RUN1\P_matrix.csv outputs\RUN2\P_matrix.csv`
→ No differences encountered

Therefore:

The universal substrate execution is **replay-identical under deterministic rebuild discipline**.

D.5 Civilization-Grade Conditions Verified (Domain Independence)

The universal trace execution satisfies:

1. **Finite alphabet** `|A4| = 4`
2. **Deterministic mapping rules**

3. **Published parameter set**
4. **Replay identity proof** `B_A = B_B`
5. **Spectral boundedness** `rho(P) = 1`
6. **Collapse invariant preserved:** `phi((m,a,s)) = m`
7. **No probabilistic blending**
8. **No regime expansion**

Therefore:

The **Shunyaya Structural Substrate Layer (SSSL)** demonstrates **domain-independent structural regime governance** over measurable scalar magnitude evolution.

D.6 Structural Implication

Appendix C established deterministic substrate stability over a real electrodynamics dataset.

Appendix D establishes deterministic substrate stability over a domain-agnostic magnitude trace.

Together, these results demonstrate:

SSSL functions as a **deterministic structural regime algebra** over measurable scalar magnitude evolution.

This algebra:

• Preserves magnitude exactly via `phi((m,a,s)) = m`
• Remains bounded under `A4`
• Maintains spectral stability `rho(P) = 1`
• Is replay-verifiable under identical inputs

Appendix E — Cross-Physical Substrate Validation

E1. Mechanical Vibration Substrate Demonstration (Deterministic Trace Execution)

Dataset Execution Context

Mechanical vibration magnitude trace (deterministic envelope lifecycle)
Observations: 60

State space:

```
A4 = {Z0, Eplus, S, Eminus}
```

Conservative extension:

```
phi((m,a,s)) = m
```

This trace represents a mechanical vibration lifecycle at the magnitude level:

- Rest baseline
- Ramp-up (excitation)
- Regulated plateau (stable amplitude)
- Shock/drop event
- Recovery

No probabilistic inference was introduced.
No mechanical dynamics were simulated.

This demonstration validates deterministic structural regime discipline only.

Magnitude remains governed by mechanical law.
Structure remains finite and replay-verifiable.

E1.1 Structural State Exercise

State counts (deterministic):

• Z0: 8
• Eplus: 46
• S: 5
• Eminus: 1

This confirms that all regimes within `A4` were exercised under deterministic mapping discipline.

E1.2 Admissibility Verdict

Artifact: `adm_result.txt`

SSSL admissibility verdict
`adm_E: ALLOW`

Deterministic metrics:

• avg_dwell_S = 1.0
• churn_ratio = 0.21666666666666667
• collapse_ratio = 0.016666666666666666
• count_S = 5.0

Interpretation

• Stability state `S` appears deterministically.
• Collapse events are rare and bounded under declared admissibility discipline.
• Churn remains safely within configured limits.
• Substrate collapse permission is granted under published thresholds.

No probabilistic inference was introduced.
No domain law was modified.
This is **structural admissibility governance only**.

E1.3 Deterministic Spectral Stability

Artifact: `eigenspectrum.txt`

Matrix order: `[Z0, Eplus, S, Eminus]`

Eigenvalues (deterministic QR iteration):

```
0.875000000000
1.000000000000
-0.133333333333
0.000000000000
```

Spectral radius estimate:

```
rho(P) = 1.000000000000
```

E1.4 Structural Interpretation

1. **Dominant eigenvalue = 1.0**
   → Structural closure holds
   → No regime expansion
   → Finite alphabet remains invariant
2. **All |lambda| <= 1**
   → System is spectrally bounded
   → No structural instability amplification

3. Eigenvalue ≈ 0.875
   → Indicates strong bounded persistence in the regime operator
   → Reflects stable recurrence behavior within the finite structural alphabet
4. Zero eigenvalue present
   → Indicates a terminating structural subspace component
   → No absorbing runaway regime is introduced

No stochastic equilibrium claim is made.
No physical reinterpretation is introduced.
This is **structural stability verification only**.

E1.5 Civilization-Grade Conditions Verified (Mechanical Domain)

This execution satisfies:

1. Fixed finite alphabet `|A4| = 4`
2. Deterministic mapping rules
3. Published parameter set
4. Spectral condition `rho(P) = 1`
5. Collapse invariant preserved: `phi((m,a,s)) = m`
6. Admissibility discipline enforced (`adm_E: ALLOW`)

Therefore:

SSSL demonstrates deterministic structural substrate governance over mechanical vibration magnitude evolution.

E2 — Fluid Pressure Substrate Demonstration (Deterministic Trace Execution)

Dataset Execution Context

Fluid pressure magnitude trace (deterministic lifecycle envelope)
Observations: 70
State space: `A4 = {Z0, Eplus, S, Eminus}`
Conservative extension: `phi((m,a,s)) = m`

This trace represents a fluid pressure lifecycle at the magnitude level:

• low baseline pressure
• controlled pump ramp
• regulated pressure plateau

• pass drop event
• regulated recovery

No probabilistic inference was introduced.
No fluid dynamics equations were modified.
This is **structural substrate validation only**.

E2.1 Structural State Exercise

From deterministic execution summary:

All structural regimes within `A4` were exercised under fixed parameter discipline.

State space remained strictly bounded:

```
|A4| = 4
```

No runtime regime creation occurred.

E2.2 Admissibility Verdict

Artifact: `adm_result.txt`

SSSL admissibility verdict
```
adm_E: ALLOW
```

Deterministic metrics:

• avg_dwell_S = 2.0
• churn_ratio = 0.21428571428571427
• collapse_ratio = 0.014285714285714285
• count_S = 9.0

Interpretation

• Structural stability state `S` appears deterministically (9 occurrences).
• Collapse frequency remains very low (~1.43%).
• Churn remains well within configured admissibility threshold.
• Substrate collapse permission is granted under published governance parameters.

No probabilistic inference was introduced.
No pressure prediction was introduced.
No fluid dynamics equations were modified.
This is **structural admissibility governance only**.

E2.3 Deterministic Spectral Stability

Artifact: `eigenspectrum.txt`

Matrix order: `[Z0, Eplus, S, Eminus]`

Eigenvalues (deterministic QR iteration):

```
0.900000000000
1.000000000000
0.221226252908
-0.030750062432
```

Spectral radius estimate:

```
rho(P) = 1.000000000000
```

Structural Interpretation

1. **Dominant eigenvalue = 1.0**
   → Structural closure holds
   → No regime expansion
   → Finite alphabet remains invariant
2. **All |lambda| <= 1**
   → System is spectrally bounded
   → No structural instability amplification
3. Sub-dominant eigenvalue ≈ 0.90
   → Indicates strong bounded persistence in regime transitions
   → Reflects stable pressure cycling under finite regime discipline
4. Small negative eigenvalue present
   → Indicates bounded oscillatory structural component
   → No absorbing runaway regime introduced

This confirms deterministic structural boundedness of the transition operator.

No stochastic kinetics claim is made.
No thermodynamic reinterpretation is introduced.
This is **structural spectral verification only**.

E2.4 Civilization-Grade Conditions Verified (Fluid Domain)

This execution satisfies:

1. Fixed finite alphabet `|A4| = 4`
2. Deterministic mapping rules
3. Explicitly published parameter set

4. Spectral condition `rho(P) = 1`
5. Collapse invariant preserved: `phi((m,a,s)) = m`
6. Replay discipline satisfied (`B_A = B_B`)
7. Admissibility discipline enforced (`adm_E: ALLOW`)

Appendix F — Structural Magnitude Universality Theorem (SSSL-UMT)

UMT.1 Statement

Structural Magnitude Universality Theorem (SSSL-UMT)

Let:

• `A4 = {Z0, Eplus, S, Eminus}` be fixed as the finite structural alphabet.
• `F(E_i, dE_i, d_i)` be the deterministic structural mapping defined by SSSL.
• `phi((m,a,s)) = m` be the conservative collapse operator.
• `P` be the deterministic transition operator derived from replay-verified execution.

Then:

For any measurable scalar magnitude trace

```
M = {E_1, E_2, ..., E_N}
```

with deterministic thresholds `{tau0, taus, eps, drop}` fixed across the dataset,

if:

1. Mapping `F` is total and deterministic,
2. Alphabet remains finite (`|A4| = 4`),
3. Collapse invariant holds (`phi((m,a,s)) = m`),
4. Spectral condition `rho(P) <= 1`,
5. Replay identity holds (`B_A = B_B`),

then:

SSSL constitutes a domain-independent deterministic structural regime algebra over scalar magnitude evolution.

UMT.2 Interpretation

The theorem asserts:

SSSL does not depend on:

• Electrodynamics
• Mechanical dynamics
• Fluid equations
• Thermodynamics
• Any specific physical law

It depends only on:

• Observable scalar magnitude
• Deterministic threshold discipline
• Finite structural alphabet
• Collapse invariance
• Replay identity

Therefore:

If a system can be expressed as a measurable scalar magnitude evolving over ordered observation steps,
then SSSL can govern its structural regime evolution deterministically.

UMT.3 Proof Sketch (Deterministic)

1. The structural mapping `F` depends only on:
   • `E_i`
   • `dE_i = E_i - E_{i-1}`
   • Deterministic thresholds
2. No domain-specific physics is encoded in `F`.
3. The transition operator `P` is constructed purely from observed regime transitions.
4. Spectral boundedness condition `rho(P) <= 1` guarantees:
   • Finite regime closure
   • No regime explosion
   • No structural amplification beyond the alphabet
5. Collapse operator satisfies:
   `phi((m,a,s)) = m`
   guaranteeing conservative extension.

Since none of these conditions reference domain-specific laws,
the substrate algebra is domain-agnostic.

Therefore:

SSSL operates as a universal deterministic structural regime algebra over scalar magnitude traces.

UMT.4 Boundary Conditions

SSSL-UMT does not claim:

• Predictive capability
• Replacement of physical laws
• Modification of domain equations
• Causal inference
• Safety-critical control authority

It claims only:

Deterministic structural regime governance over magnitude evolution under replay discipline.

UMT.5 Formal Universality Condition

Let:

`M_D` be a magnitude trace from domain `D`.

If `M_D` satisfies:

• Ordered observation index
• Deterministic threshold publication
• Finite structural alphabet discipline
• Spectral boundedness
• Collapse invariance

then:

`SSSL(M_D)` is structurally valid.

This holds independent of domain `D`.

UMT.6 Empirical Support

The theorem is supported by deterministic execution across:

- Electrodynamics magnitude trace (battery discharge)
- Domain-agnostic lifecycle trace (smoke)
- Mechanical vibration magnitude trace
- Fluid pressure magnitude trace

All executions satisfy:

- `|A4| = 4`
- `rho(P) = 1`
- `phi((m,a,s)) = m`
- Replay identity `B_A = B_B`
- Admissibility `adm_E: ALLOW`

Cumulative Substrate Confirmation

SSSL has been executed successfully across multiple physical domains.

All executions satisfy:

- Deterministic mapping discipline
- Replay identity discipline
- Spectral boundedness
- Collapse invariance
- Finite structural alphabet preservation

Therefore:

SSSL demonstrates deterministic structural substrate governance over magnitude evolution across multiple physical domains.

UMT.7 Conclusion

The Structural Magnitude Universality Theorem establishes that:

Shunyaya Structural Substrate Layer (SSSL) is not an electrodynamics classifier.

It is:

A deterministic structural magnitude algebra operating across physical domains under finite regime discipline and conservative collapse invariance.

Appendix G — Geophysical Substrate Verification (Seismic Magnitude Dataset)

Dataset Execution Context

External earthquake magnitude catalog (USGS event archive)
Observation window: 10 years
Magnitude filter: `mag >= 5.5`
Observations: 4591

State space: `A4 = {Z0, Eplus, S, Eminus}`
Conservative extension: `phi((m,a,s)) = m`

Deterministic parameters (explicitly declared):

```
tau0 = 0.05
taus = 0.7
eps = 0.02
drop = 0.15
```

Universal Substrate Interface mapping:

`t_s` = ordered observation index
`E_proxy` = `mag`
`discharge` = `1 if mag >= 6.5 else 0`

No domain equation blending was introduced.
No seismic modeling was performed.
This is **structural substrate compression only**.

G.1 Admissibility Verdict

Artifact: `adm_result.txt`

SSSL admissibility verdict:
`adm_E: ALLOW`

Deterministic metrics:

- `avg_dwell_S = 1.2043010752688172`
- `churn_ratio = 0.6362448268351122`
- `collapse_ratio = 0.42234807231539967`
- `count_S = 560.0`

Interpretation

• Stability regime `S` appears deterministically.
• Structural collapse frequency remains bounded under declared thresholds.
• Regime churn remains within admissibility envelope.
• No probabilistic smoothing was introduced.
• No forecasting logic was applied.
• No geophysical inference was claimed.

This execution constitutes **structural admissibility governance only**.

G.2 Deterministic Spectral Stability

Artifact: `eigenspectrum.txt`

Matrix order:
`[Z0, Eplus, S, Eminus]`

Eigenvalues (deterministic QR iteration):

```
0.000000000000
1.000000000000
-0.064669431375
0.015000723699
```

Spectral radius estimate:

```
rho(P) = 1.000000000000
```

Interpretation

1. Dominant eigenvalue = `1.0`
   → Structural closure holds
   → Alphabet does not expand
2. All `|lambda| <= 1`
   → Operator is spectrally bounded
   → No structural amplification
3. Zero eigenvalue present
   → Indicates non-expanding subspace component
   → No runaway absorbing regime introduced

This confirms **deterministic structural boundedness** of the transition operator.

No stochastic equilibrium claim is made.
No physical reinterpretation of seismic mechanics is introduced.
This is **structural stability verification only**.

G.3 Structural Distribution Summary

Artifact: `summary.txt`

State counts:

- `Z0: 0`
- `Eplus: 2092`
- `S: 560`
- `Eminus: 1939`

Because the dataset is filtered to `mag >= 5.5`,
the threshold rule for `Z0` is not triggered.

This is expected behavior under declared parameterization.

The alphabet remains finite:

```
|A4| = 4
```

No new structural regime appeared.

G.4 Replay Identity Discipline

Two independent executions were performed:

```
SEISMIC_V13
SEISMIC_V13_RUN2
```

Manifest comparison confirms:

```
B_A = B_B
```

Byte-identical artifacts:

- `MANIFEST.sha256`
- `P_matrix.csv`
- `transition_counts.csv`
- `transition_ratios.csv`
- `eigenspectrum.txt`
- `adm_result.txt`

Replay identity discipline holds.
No floating tolerance matching was used.

This satisfies **deterministic artifact reproducibility**.

G.5 Civilization-Grade Conditions Satisfied

The seismic substrate execution satisfies:

1. Finite alphabet: `|A4| = 4`
2. Deterministic mapping rules
3. Explicit parameter publication
4. Replay identity discipline `B_A = B_B`
5. Conservative collapse invariant `phi((m,a,s)) = m`
6. Spectral boundedness `rho(P) = 1`
7. No probabilistic inference
8. No equation blending
9. No cross-domain physical modeling

Therefore:

The Shunyaya Structural Substrate Layer (SSSL) demonstrates deterministic structural compression over geophysical magnitude traces under replay discipline.

This strengthens cross-domain structural universality without altering domain physics.

G.6 Dataset Attribution and Handling

Dataset Name: USGS Earthquake Event Catalog
Source: United States Geological Survey (USGS) public event archive
Fields used: `time`, `mag`

Dataset was obtained directly from the official USGS export interface.

No dataset redistribution is included with this work.

SSSL uses the dataset strictly for:

• Deterministic structural regime compression
• Transition operator construction
• Replay-verifiable artifact generation
• Spectral boundedness verification

No earthquake prediction was performed.
No hazard modeling was introduced.
No geophysical inference was claimed.

This execution constitutes **structural substrate verification only**.

Dataset Handling Notice

No third-party datasets are redistributed with this work.

All external datasets referenced must be obtained directly from their original licensed sources.

Only deterministic structural artifacts, transition operators, and replay-verifiable summaries derived from execution are included.

Users are responsible for complying with the original dataset license terms when reproducing experiments.

OMP