TSMLA™ Formal Objects Glossary (v1.0)

Purpose: This glossary defines all mathematical objects, notation, and structural terms used in TSMLA™ (Tag-Weighted Self-Mirroring Logic Architecture). All definitions are operationally precise and patent-grade.

Core State Objects

S — Declared State

Type: Immutable 4-tuple during a session
Definition: $S = (I, T, \theta, \varkappa)$
Semantics: S is the fixed logic container that binds all user inputs, tags, thresholds, and configuration for a single mirror session. On loopback, user may redeclare $S' \neq S$ (new session state).

I — Input Signal Set

Type: Set of declared input signals
Definition: $I = \{i_1, i_2, ..., i_n\}$ where each i_j is a typed signal
Semantics: User-declared signals bound to S. May include scalar values, Boolean flags, categorical tags, or constraint declarations. Typing is governed by $\varkappa$.

T — Tag Assignment Function

Type: Function $T: I \rightarrow TagTypes$
Definition: Maps each input signal $i_j \in I$ to a tag type (e.g., Value, Constraint, Boundary, Priority)
Semantics: T structures the signal set I into categories for layer processing. BDL™ uses T to identify contradiction types.

θ — Threshold Vector

Type: Real-valued vector $\theta = (\theta_1, \theta_2, ..., \theta_m)$
Definition: Component-wise thresholds for contradiction activation, compression triggers, and sensitivity controls
Semantics: Example: BDL™ may flag contradiction if $|\Delta| > \theta_1$; HCL™ compresses if recursion depth $> \theta_2$. User-declared or module-fixed via $\varkappa$.

$\varkappa$ — Configuration Parameters

Type: Record/tuple of system configuration
Definition: $\varkappa$ = {module_type, scope_structure, signal_typing, layer_enables, ...}
Semantics: Governs:

- Signal weight typing (scalar $\in$ [0,1], Boolean, categorical)
- Scope structure (intra-scope vs. cross-scope contradictions)
- Layer activation rules (which layers are enabled)
- Module-specific constraints

Layer Outputs

C_i — Contradiction Tuple

Type: 3-tuple C_i = (Type_BDL, Signal_Origin, Scope_Boundary)
Definition:

- **Type_BDL:** Contradiction type assigned by BDL™ (Functional, Temporal, Protective, Ethical, Perceptual)
- **Signal_Origin:** The input signals i_j , $i_k \in I$ that produced the contradiction
- **Scope_Boundary:** Boolean or categorical marker indicating whether contradiction is intra-scope or spans sub-scopes (defined by $\varkappa$)

Semantics: Each detected contradiction is tagged and localized to specific signals and scope regions.

$\Lambda(S)$ — Lawful Traversal Sequence

Type: Ordered sequence of gates
Definition: $\Lambda(S) = [G_1, G_2, ..., G_k]$ where each G_i is a traversal gate in state S
Semantics: CTC™ enforces this sequence. User cannot advance from gate G_i to G_{i+1} without resolving contradictions at G_i. Sequence is determined by declared state S and module structure $\varkappa$.

TV_i — Traversal Violation Marker

Type: Indexed violation marker
Definition: TV_i = (gate_index, violation_type)
Semantics: Returned by CTC™ when user attempts to bypass gate G_i in $\Lambda(S)$ without resolving contradictions. System halts forward traversal until violation is cleared or S is redeclared.

G(V, E) — Contradiction Graph

Type: Directed graph
Definition:

- **V:** Vertex set = declared input signals I
- **E:** Edge set = contradiction relationships between signals

Semantics: Optional visualization output. Edge $(i_j, i_k) \in E$ exists if signals i_j and i_k produce contradiction C_i. Graph structure is derived from $\{C_1, ..., C_n\}$.

Mirror Functions

α — Abstraction Map

Type: Function α: I → Abstract Logic Structure
Definition: Maps concrete user inputs (natural language, numerical values) to formal logic structure (Boolean relations, constraint networks, signal weights)
Semantics: First stage of mirror operation. Strips irrelevant natural language phrasing; preserves logical structure.

γ — Concretization Map

Type: Function γ: Abstract Logic Structure → User-Legible Output
Definition: Maps abstract logic structure (contradiction tags, coherence indices, traversal states) back to user-facing presentation
Semantics: Second stage of mirror operation. Enforces $\gamma(\alpha(x)) = x$ for all structure relevant to S. Natural language may differ; logical structure, tags, and weights are preserved exactly.

Mirror-Pure Idempotence

Property: $\gamma(\alpha(x)) = x$ for all structure relevant to declared state S
Interpretation: No information is generated. No information is lost. Identical declared inputs under identical state S → identical tagged outputs, even with non-deterministic internal traversal paths.

Layer-Specific Notation

RSF(S) — Resonant State Function Output

Type: Coherence Index ∈ [0,1]
Definition: Global resonance score across all tagged signals in I under declared state S
Semantics:

- **Index = 1:** Zero contradiction; all signals coherent
- **Index → 0:** High structural divergence across scope boundaries

Non-Determinism: Internal traversal order may vary, but idempotent merge guarantees:
RSF(S) ⊕ RSF(S) = RSF(S)
No randomness. No inference. Replay-equivalent output under fixed S.

BDL™ Contradiction Types

Set: {Functional, Temporal, Protective, Ethical, Perceptual}
Semantics: Boolean structure-based classification of contradictions. Assigned by BDL™ trigger on mutual constraint violations (e.g., $A \wedge \neg A$, XOR conflicts).

HCL™ Compression

Operation: Collapse structurally redundant contradiction clusters
Constraint: Preserve mirror fidelity (no substrate alteration)
Placement: Presentation layer only; does not affect RSF computation

---

SNS Filter

Operation: Logic-type filter at γ (concretization) boundary

Purpose: Block ideological, emotional, or cultural bias from outputs
Enforcement: Substrate-presentation boundary separation; no interpretive overlay allowed

---

Session Semantics

Loopback

Operation: User redeclares $S' \neq S$ (new session state)
Semantics: Prior outputs from S remain available for comparison. User may run S and S' in parallel or sequentially and compare contradiction sets, coherence indices, or traversal paths.

Export

Operation: Serialize mirror output to user-accessible artifact
Contents: $\{C_1, ..., C_n\}$, RSF(S), $\{TV_i\}$, optional G(V,E)
Data Handling: Exported artifacts are the only persistent user data. Session state S is cleared on exit unless explicitly exported.

Cross-References

- Full operational pipeline: See "TSMLA User Path Overview v1.4"
- Layer specifications: See LP-series papers (LP-BDL, LP-RSF, LP-CTC, LP-HCL, LP-SNS)
- Patent scaffolding: See "Clean Technical Chain (CTC)" specification

---

Notation Version: 1.0
Canonical Source: Fractal Labyrinth Systems LLC, 2025
Correspondence: All notation is consistent with TSMLA™ core patent filings and modular logic paper series (LP-01 through LP-20+).