The Non-Archimedean Reasoning Field: A Synthesis of Logic and Geometry

Project Vision: The Synthesis of Logic and Geometry

The project introduces a unified mathematical structure—the **Non-Archimedean Reasoning Field**—to formalize the recursive, dialectical movement of thought and complex systems (the **Tlurey** process). Our objective is to visualize **Multidimensional Cognition**, where conflicts and choices exist as complex, high-dimensional probability fields rather than simple linear paths.

This field operates by fusing three core layers:

1. **Metric Layer:** p-adic number systems enable **Convergent Recursion** (logical synthesis via ultrametric convergence).
2. **Topological Layer: Pants Cobordism** (pair-of-pants surfaces) model **Decision Bifurcation** (dialectical branching/merging).
3. **Geometric Layer: Calabi–Yau Manifolds** define **Compact Hidden Curvature** (multidimensional "folded" cognition).

The resultant field is **locally triadic** (p-adic 3-ary digits $\leftrightarrow$ pants legs), **globally compact** (Calabi–Yau closure), and **dynamically self-normalizing** (recursive processes converge).

I. Foundational Reasoning: N-ary Logic and Structural Numerology

The core discovery is the **necessity of n-ary logic (with $n \ge 3$)** to adequately model phenomena involving contradiction, recursive sublation, and dimensional shifts. In summarizing the rationale behind our approach, we highlight several key concepts and indicate whether they arise as formal derivations or intuitive analogies (see Table 1 below).

A. Rationale: From Binary Constraint to Triadic Synthesis

Concept	Nature of Finding	Status (Derived vs. Intuited)
Minimum Synthesis	Base-3 is the **minimal p-adic space** that allows a structural third position (Synthesis) not reducible to two opposing binaries.	**Derived Structural Necessity** (mathematically smallest non-binary base)

Concept	Nature of Finding	Status (Derived vs. Intuited)
Decision Thresholds	The minimal threshold for **consideration** is 0.25, derived from the geometry of a 2D binary problem space ($2 \times 2 = 4$ outcomes). The critical threshold for **branching/action** is 0.5.	**Derived Structural Necessity** (from binary outcome proportions)
Dimensional Necessity	Modeling recursive dialectics and embedding complex topology (pants cobordism) requires at least **4 real dimensions** ($n>3$) to avoid self-intersection (by topological immersion arguments).	**Derived Topological Requirement**
Cognitive Compactification	The system's complexity is modeled by **compact, Ricci-flat** geometries (a Calabi–Yau manifold analogy) representing hidden $6!D$ "emotional/systemic" harmonics folded within $3!D$ experience [1] [2] .	**Intuited Geometric Analogy** (using string-theoretic compactification)
Recursive Closure	The framework's ability to model infinite, iterative consideration resolving into a finite outcome is captured by **p-adic convergence** (sums of infinite series can yield finite results in $\mathbb{Q}_p$).	**Derived Mathematical Mechanism**

Table 1 – Key concepts enabling non-Archimedean reasoning. Base-3 emerges as the minimal logic to allow a *synthesis* value beyond binary; higher dimensions and curved manifolds are invoked to accommodate complex, self-consistent reasoning loops.

B. Findings Across Mathematical Bases ($N=2$ to $N=9$)

We analyzed logical bases from binary up to nonary, as well as the transfinite surreal number system, observing a structural hierarchy where higher bases unlock new cognitive and geometric capabilities (see Table 2). Notably, base-3 and its multiples (3, 6, 9) play unique roles in capturing recursive and self-referential structures, while base-2 alone is insufficient beyond simple oppositions.

Base N	Name (Adjective)	Structural Finding / Discovery	Key Algebraic Property
2	Binary (Dyadic)	Captures basic **opposition**; but reasoning tends toward **over-atomization** (splitting into too many isolated facts).	**Group** (every element has an **inverse**; reasoning is strictly reversible or not at all).
3	Ternary (Triadic)	The minimal **synthesis engine** – allows a structural third value that integrates contradictions. Forms the basis of the $3 \to 6 \to 9$ progression.	**Semiring** (lacks a global inverse); contradictions can **coexist and integrate** rather than annihilate each other.

Base N	Name (Adjective)	Structural Finding / Discovery	Key Algebraic Property
4	Quaternary (Quartic)	Minimal geometry for **two binary axes interacting** (a binary pair yields $2^2=4$ states). Useful for modeling an "indeterminate" absorbing element (e.g. unknown/ undefined).	**Group** (isomorphic to $\mathbb{Z}_2 \times \mathbb{Z}_2$); supports an absorbing **indeterminate** (like a zero that halts one axis).
6	Senary (Hexadic)	Minimal extension for **recursive closure of the Tetralemma** (Truth, Falsehood, Both, Neither, plus **Void** and **Return** states). Enables a full cycle back to unity.	**Non-field**; a pseudo-inverse appears as a **180° phase conjugate** rather than a true algebraic inverse.
9	Nonary (Nonadic)	**Full integrative manifold** – models a **meta-dialectic** (think $\mathbb{C}_3 \otimes \mathbb{C}_3$). The 3×3 structure can encode two triadic processes (e.g. self and other) within one system.	**Semiring**; contradiction **"teaches"** the system (self-referential update), allowing **self-rewriting** logic.
Transfinite	Surreal ($\mathbb{No}$)	Provides a complete numeric tower including **infinitesimal and infinite quantities**; serves as a basis for rigorous recursive construction (all ordinals and reals embedded).	Built via **ordered cut sets** (akin to Dedekind cuts). Includes *all* real numbers and an infinity of others, offering arbitrary granular partitioning.

Table 2 – Characteristics of various logic bases. Each base N beyond binary adds new representational power (e.g. ternary allows dialectical synthesis; senary permits a closed tetralemma cycle; nonary integrates multiple triadic subsystems). Surreal numbers supply an underlying continuum encompassing the others.

II. Formal Mathematical Proofs

To ensure rigorous foundations, we rely on explicit mathematical proofs drawn from the surreal number system and p-adic analysis. These demonstrate that our framework can internally reproduce standard arithmetic truths and handle infinite recursive processes in a controlled way. Table 3 highlights key proofs and derivations supporting the architecture:

Proof Statement	Mathematical Proof / Derivation (Sketch)
3-adic Convergence (Algebraic Counterpart)	The infinite series $x = 1 + 3 + 3^2 + 3^3 + \dots$ converges in the 3-adic metric. **Derivation:** In $\mathbb{Q}_3$, $3x = x - 1$ (because adding a factor of 3 shifts the 3-adic digits), which gives $2x = -1$, hence **$\mathbf{x = -\frac{1}{2}}$**. This counter-intuitive result (divergent in reals, convergent in 3-adics) illustrates how infinite regress can yield a finite outcome.

Proof Statement	Mathematical Proof / Derivation (Sketch)
Surreal Arithmetic (Half Plus Half equals One)	In the surreal number construction, $\frac{1}{2}$ is defined as ${0 \mid 1}$. Using Conway's rules for addition, one can show ${0
Surreal Arithmetic (One Plus One equals Two)	By definition, $1 = {0 \mid }$ in the surreal construction. Then $1 + 1 = {0
Tetralemma Structural Justification	The four classical logical positions (A, ¬A, Both, Neither) are structurally justified by the geometry of the **minimal binary problem space**: two independent binary questions produce $2 \times 2 = \mathbf{4}$ possible answer combinations. This corresponds to the **Tetralemma's** four outcomes, and defines the critical **0.25** consideration probability threshold (each outcome has 25% of the space). Beyond this, a **0.5** probability marks the tipping point where one binary axis dominates, forcing a branch in decision.

Table 3 – Selected proofs supporting the framework. These results show that the non-Archimedean field can converge infinite series (via p-adic logic) and is arithmetic-complete (via surreal numbers), and that classical logical structures (like the Tetralemma) emerge naturally from higher-dimensional binary combinations.

III. State-of-the-Art Comparison and Market Positioning

Our framework bridges the gap between **differentiable neural fields** (common in modern AI) and **formal recursive logic** (modal logics and ultrametric systems). Below we compare relevant state-of-the-art approaches to highlight our model's innovation and unique market niche:

SOTA Domain	Existing Approaches (Representative)	Our Model's Innovation & Niche
Neuro-Symbolic AI / Logic	Integration of logic in AI via Temporal Logics (LTL, CTL), model checking, or probabilistic logic programming (e.g. DeepProbLog). These often operate over state graphs with fixed binary logic, and struggle with recursion or self-reference.	We provide a **differentiable field over an ultrametric space** (a continuous-valued reasoning landscape on a p-adic tree). This allows smooth credit assignment through hierarchical logical structures. We use modal **μ-calculus** fixpoint operators (least/greatest fixpoints) to formalize potentially infinite reasoning loops (e.g. "always eventually" scenarios) [3], enabling **recursive self-similarity** (when a process overflows a threshold, it moves to a higher-level voxel).

SOTA Domain	Existing Approaches (Representative)	Our Model's Innovation & Niche
High-Dimensional Geometry in ML	*Topology-informed networks:* e.g. p-adic neural networks that organize signals on tree hierarchies [4], or **Sheaf Neural Networks** that enforce consistency across graph structures [5]. These methods encode hierarchy or topology, but often separately from continuous feature fields.	We unify **continuous field** representations with **discrete p-adic addressing** in one system. Our reasoning "mass" lives on a p-adic tree (inheriting a natural hierarchy), while continuous kernel functions (GELU-based) spread activation in a **continuous metric field**. This blend merges the p-adic CNN approach (dynamical systems on $\mathbb{Q}_p^N$ [6]) with high-dimensional feature spaces, allowing both precise hierarchical semantics and smooth geometric interpolation.
Cognitive Modeling	Emphasis on *predictive coding* and Free Energy Principle (Friston), or hierarchical Bayesian models (HMMs). These focus on minimizing prediction error and often use Euclidean latent spaces; logical contradictions are treated as errors, not structure.	We focus on **structural representation** of thought: how reasoning can be mapped into geometric space. By using multi-scale topology (triadic -> hexadic -> nonadic), we can map psychological or conceptual states (3D basic, 6D extended, 9D integrated) onto a **compact topological manifold** (analogous to a Calabi–Yau space) that preserves **multiscale agency**. This offers a new way to model cognitive dissonance and resolution: not as errors, but as *topological features* (holes, loops) in the reasoning field.
Data Visualization & Analysis	Standard dimensionality reduction (PCA, t-SNE) and clustering assume Euclidean geometry. High-dimensional conflicts are hard to visualize; voids (no data regions) and dense regions are not distinguished. Visual metaphors like Swiss-cheese holes (for missing data) or "wormholes" (for close but distant points) are not explicit in current tools.	We introduce **chromatic tetralemma visualization**: using 4 base colors (C, M, Y, K) for four primary logical axes and an α-channel for **void** (uncertainty/invalidity). This way, we display presence and absence simultaneously (voids appear as translucent "holes"). We also show **isovalue surfaces** at critical thresholds (e.g. the 0.5 decision boundary forms a "worm" or tunnel through the field). This dual visualization of **ridges vs. holes** gives analysts an intuitive map of dialectical tension in high-D data.

Gaps in the Market & Target Niches: We identify several domains where our Non-Archimedean Reasoning Field can provide game-changing value by addressing unmet needs:

1. **Legal and Policy Analytics (e.g. *SensibLaw*, *TiRCorder*):** Current AI systems lack a geometric *map* of decision space. Our model can visualize how institutions and individuals traverse high-dimensional **opposition spaces** (e.g. competing legal arguments or policy objectives). **Target use-case:** Identify systemic risk points (where many independent factors converge to a critical

contradiction) and uncover non-obvious correlations in legal data (by seeing “closeness” of cases or laws in the reasoning field).

2. **Epistemology as Engineering:** Philosophy and logic frameworks (dialectics, the Tetralemma, even Lacanian topology of mind) remain abstract. Our system offers a rigorous, testable sandbox for these ideas – effectively **computational metaphysics**. **Target use-case:** Academic research and education, where complex logical paradigms (like true contradictions, “Both/Neither” states, etc.) can be simulated and visualized, turning epistemology into an engineering discipline.
3. **Advanced ML Architectures:** There is growing interest in neural networks that go beyond Euclidean vector space – for example, recent work on p-adic neural nets shows that using p-adic weights yields *perfectly ultrametric* representations with transparent hierarchy [7] [8]. Our framework can inspire new **hierarchy-aware neural architectures** that natively handle tree-structured data and recursive patterns. **Target use-case:** AI systems dealing with taxonomy, file systems, provenance tracking, or any inherently hierarchical knowledge – where our field could serve as either a preprocessing step (embedding data into an ultrametric space) or as a new network layer type.

IV. Roadmap and Key Deliverables

Our development roadmap centers on three pillars of the system: the **Field Engine**, the **Recursive Logic Resolver**, and the **Visualization Pipeline**. We outline the major feature sets and technical components below, focusing on capabilities rather than timeline. This serves as a **readme and technical blueprint** for how we will implement the Non-Archimedean Reasoning Field.

A. Feature Set Deliverables

Deliverable (Feature Set)	Description & Theoretical Alignment
Dashifine High-D Field Engine	The core engine defining the N-dimensional state space X. It uses **GELU-based radial kernels** (Gaussian Error Linear Unit activation profiles) centered at various points in $\mathbb{R}^N$. These kernels have anisotropic spreads σ_i to model “worm” thickness in each dimension. This creates a smooth field where each hypothesis or state exerts a continuous influence.
p-adic Indexing and Addressing	A unified coordinate system where every state is indexed by a $\mathbf{p}$-adic integer. This formalizes the **nested, hierarchical structure** of the space: conceptually, each state lives in a ultrametric tree (supervoxels containing subvoxels). Distance in this index correlates with conceptual distance. (E.g. two states with a long common p-adic prefix are closely related in the hierarchy.)
Decision Threshold Operator (SCN / $A(x)$)	Implements the **Solis Communicator Notation (SCN)** – a formal system for systemic integrity and alert signals. Defines **integrity** $I(x)$ as a measure of internal consistency, and a risk signal $A(x) = \|T(x)\|_Y$ as the norm of the tension vector $T(x)$ in some outcome space Y. This component uses our derived thresholds (0.25 for consideration, 0.5 for action) to decide when a developing idea **“branches”** into distinct sub-ideas.

Deliverable (Feature Set)	Description & Theoretical Alignment
Recursive Logic Resolver	The formal logic engine that handles the Tlurey dialectic cycles. Built using modal μ-calculus, it can evaluate formulas with **nested fixpoints** to represent statements like "eventually always (condition)" or infinite loops. When the field engine detects a cycle (e.g., $A(x)$ oscillates without settling), the resolver uses a **greatest fixpoint** operator to determine if it converges or if a higher-level synthesis is needed (i.e. triggers an "overflow to a higher voxel" in the p-adic hierarchy).
Chromatic Tetralemma Visualization	A high-dimensional **color mapping** module for rendering the field on a 2D screen. We use **CMYKA channels**: Cyan, Magenta, Yellow, Black correspond to four primary logical axes or classes (for example, True, False, Both, Neither in a tetralemma), and **Alpha (transparency)** corresponds to *validity/void*. Thus, a point's color shows its logical leaning, while transparency indicates confidence or the presence of a "hole" (void) at that point. This allows viewers to literally *see* contradictions (where multiple color components mix) versus uncertainty (which fades out).
Semantic & Geometric Closeness Metric	A hybrid distance metric $\Delta(x_1, x_2)$ that guides clustering and layout. It combines **semantic distance** (e.g. in an embedding space of concepts, weighted ~70%) with **geometric opposition distance** (distance in the p-adic logical coordinate, ~30%). This metric defines the notion of "near" in the field, balancing literal similarity with dialectical opposition. It also influences α (the sharpness of GELU kernels): semantically close points produce smoother, more overlapping fields, whereas pure opposites yield sharp boundaries.
Topological Invariant Mapping	A feature to compute and display **topological invariants** of the reasoning space. For example, it identifies loops, voids, or the "pair-of-pants" splits in the field and marks these on visualizations. This uses homotopy theory to recognize when a path in the field cannot be continuously contracted (indicating a stable contradiction or a conserved quantity in reasoning). It defines the "width of the pants legs" as a local probability density or entropy, showing how broad each branching is.

B. Underlying Technologies and Implementation

Component	Technology / Methodology	Purpose & Alignment
High-D Field Core	*Mathematical:* Radial basis functions, kernel density estimation. *Tech:* optimized in C++ with vectorized operations, potentially GPU for large N. Uses **affine slice transforms** to allow arbitrary 2D/3D cross-sections for visualization.	Models a continuous, non-linear influence field in $\mathbb{R}^N$. The radial GELU kernels ensure smooth differentiability (for optimization) while affine transforms let us "slice" the high-D space for analysis (e.g., fix all but 2 dimensions to view a heatmap).

Component	Technology / Methodology	Purpose & Alignment
p-adic Addressing & Distance	*Mathematical:* p-adic arithmetic libraries (for indexing), tree data structures. *Tech:* custom data structure for $\mathbb{Q}_p$ ball neighbourhoods; integration with standard graph libraries for traversal.	Provides efficient lookup of hierarchical relationships. Enables fast computation of ultrametric distance and nearest common ancestors in the hierarchy (critical for computing the opposition component of our hybrid metric). Also supports hashing states to unique p-adic labels for caching results.
Recursive Logic Formalism	*Mathematical:* Modal μ-calculus (with CTL *as needed).* Tech:* A logic engine (possibly leveraging an existing model-checker or a SAT solver for fixed-point logic).	Allows rigorous reasoning about the field's dynamic properties. For example, we can prove that certain undesirable loops will eventually resolve (safety properties) or that a condition will be met infinitely often (liveness) by encoding the field update as a transition system and applying μ-calculus model checking.
Dimensionality Handling	*Mathematical:* PCA or spectral decomposition for dimension reduction. *Tech:* Python/NumPy for prototyping, then optimized Rust or C for real-time use.	Reduces complex N-dimensional data to the 4 components used in visualization (CMYK). This module will periodically recompute the principal components of current data distribution to ensure the most informative projections are being visualized, while also tracking the "hidden" dimensions that are compactified (similar to how Calabi–Yau modes would be unseen but affecting visible outcomes).
Data Integration & Semantics	*Mathematical:* NLP embeddings (word/sentence embeddings) and graph embedding techniques. *Tech:* Integration with libraries like Hugging Face Transformers for text, and NetworkX for graph-based inputs.	Converts real-world input (legal cases, policy documents, etc.) into vectors and graphs that populate the field. Semantic similarity informs initial positions in the field (so related items start near each other), while graph structures (e.g. citation networks, ontologies) inform the p-adic connections (which points to treat as having a common ancestry or direct opposition).
Visualization Pipeline	*Tech:* WebGL or Unity for 3D rendering, with custom shaders for CMYKA blending. Implements **coarse-to-fine rendering** (start with int8 precision for speed, refine to float32 for detail) and uses **isovalue surface extraction** (like marching cubes at $A(x)=0.5$ level).	Renders the field in an interactive UI. The coarse-to-fine approach ensures responsiveness (quickly identifying areas of interest), while the isosurface at $A(x)=0.5$ highlights the critical decision boundary as a tangible surface (the "wormhole" structure). Users can fly through the reasoning space, seeing the translucent voids and colored surfaces that represent the landscape of thought.

Component	Technology / Methodology	Purpose & Alignment
Geometric Topology Module	*Mathematical:* Homotopy and homology computation (using libraries like Dionysus or Gudhi for computational topology).	Calculates topological features (connected components, loops, voids) of the field at different threshold levels. This adds a rigorous **global view** of the reasoning structure – e.g., how many independent threads of thought (components) exist, whether a loop (paradox) is present, or if a void (unexplored possibility) is enclosed by a surface. These invariants guide users in understanding and trusting the model (e.g., a loop might correspond to a legal precedent cycle that needs resolution).

Roadmap Summary: The development will proceed by first building the **Field Engine** and ensuring its mathematical correctness (convergence in p-adic space, smooth kernel behavior). Next, the **Recursive Logic Resolver** and **Threshold Operator** will be implemented to handle dynamic processes and decision points. In parallel, we will design the **Visualization Pipeline**, iterating with early test data to fine-tune the CMYKA color mapping and rendering performance. Throughout, we will validate each component against known logical puzzles and real-world scenarios (like case law examples) to ensure the Non-Archimedean Reasoning Field not only functions correctly, but provides insightful and actionable representations of complex reasoning systems.

---

1 2 The Role of Calabi-Yau Manifolds in Quantum Entanglement: A Hypothesis on Quantum Interconnectivity through higher dimensions | by Ted Skyba | Medium
https://medium.com/@tedskyba/the-role-of-calabi-yau-manifolds-in-quantum-entanglement-a-hypothesis-on-quantum-interconnectivity-32872fa74248

3 computability - What are "μ-recursive functions" and μ-calculus? - Theoretical Computer Science Stack Exchange
https://cstheory.stackexchange.com/questions/3681/what-are-mu-recursive-functions-and-mu-calculus

4 6 p-adic Cellular Neural Networks | Journal of Nonlinear Mathematical Physics
https://link.springer.com/article/10.1007/s44198-022-00071-8

5 proceedings.mlr.press
https://proceedings.mlr.press/v196/barbero22a/barbero22a.pdf

7 8 [2508.01010] v-PuNNs: van der Put Neural Networks for Transparent Ultrametric Representation Learning
https://arxiv.org/abs/2508.01010