Coherence Before Engagement: An Architecture for Persistent Meaning in Digital Systems

Flyxion

January 6, 2026

Abstract

Contemporary social media platforms have become foundational infrastructure for public discourse, knowledge exchange, and collective sense–making. Despite their scale and pervasiveness, these systems exhibit persistent and well–documented pathologies: semantic drift, performative identity, misinformation amplification, and the erosion of interpersonal trust. This paper argues that these failures are not the result of deficient moderation policies or misaligned recommendation algorithms, but instead arise from fundamental architectural commitments embedded in feed–based, engagement–optimized platforms.

We present a coherence–first alternative to social media grounded in a formal semantic substrate. In this architecture, the primary unit of interaction is not a post or message, but a structured semantic object with explicit modality requirements, provenance, and bounded entropy. Identity is defined by persistence and traceable transformation history rather than visibility or popularity metrics. Evolution of meaning proceeds through typed, entropy–bounded transformations that preserve coherence over time.

Rather than proposing incremental reforms to existing platforms, this work outlines a categorically different model of social computation. We formalize core concepts, analyze their information–theoretic properties, provide concrete instantiations, and examine scalability, governance, and implementation considerations. The result is a principled framework for large–scale digital discourse in which trust and understanding emerge as stable structural properties rather than ephemeral signals.

0.1 Contributions

This paper makes the following contributions. First, it provides a formal critique of feed–based social media architectures grounded in information theory and knowledge representation. Second, it introduces a coherence–first semantic substrate in which meaning is treated as a structured, evolving object rather than a sequence of messages. Third, it formalizes entropy–bounded semantic transformation and provenance–anchored identity, and analyzes their theoretical properties. Fourth, it presents concrete instantiations and worked examples demonstrating how discourse, revision, and discovery operate without feeds or engagement metrics. Finally, it examines scalability, governance, and implementation considerations, identifying both strengths and open challenges of the proposed approach.

0.2 Roadmap

The remainder of this paper is organized as follows. Section 2 reviews related work, situating the proposed architecture relative to existing social platforms, collaborative systems, and academic literature in CSCW, HCI, and semantic knowledge systems. Section 3 develops the theoretical foundations of coherence–first design, including formal definitions of semantic objects, entropy–bounded evolution, and provenance graphs. Section 4 presents concrete instantiations and worked examples illustrating how these concepts operate in practice. Section 5 addresses challenges and limitations, including scalability, onboarding, and adversarial behavior. Section 6 formalizes key claims and provides quantitative analyses. Section 7 examines the constrained role of artificial intelligence. Section 8 elaborates governance mechanisms and invariants. Section 9 discusses implementation considerations. Section 10 concludes with a summary of contributions and a research agenda for future work.

0.3 A Motivating Example

Consider a long–running public discussion concerning a scientific or policy question. On contemporary social media, this discussion fragments into thousands of posts, responses, and quote–reposts, each optimized for visibility at the moment of publication. Corrections are rarely propagated with the same reach as initial claims, and revisions are indistinguishable from contradictions. Participants are rewarded for novelty and provocation rather than sustained engagement with the evolving substance of the topic.

In contrast, a coherence–first system represents the discussion as a small number of persistent semantic objects. Claims, evidence, counterarguments, and revisions are attached to these objects through typed transformations with explicit provenance. The discussion evolves by accumulation and refinement rather than displacement. Visibility is decoupled from immediacy, and trust accrues through demonstrated consistency over time.

The remainder of this paper develops the formal and practical implications of adopting such an architecture.

1 Related Work

The proposal advanced in this paper intersects multiple lines of prior work, including commercial social media platforms, collaborative knowledge systems, federated networks, and academic research in computer–supported cooperative work (CSCW), human–computer interaction (HCI), semantic web technologies, and knowledge representation. This section situates the coherence–first approach relative to these domains and clarifies how it differs from both existing platforms and incremental reform efforts.

1.1 Feed–Based Social Media Platforms

Dominant social media platforms such as Twitter/X, Reddit, and Facebook share a common architectural pattern: content is produced as discrete posts and delivered to users via feeds ordered by recency, engagement, or learned relevance. Identity is represented by profiles augmented with follower counts, karma, or verification markers. Visibility and influence are mediated by algorithmic ranking systems that prioritize interaction volume over semantic structure.

Extensive empirical work has documented the consequences of these design choices. Studies in CSCW and computational social science have shown that engagement–based ranking amplifies emotionally charged content, accelerates polarization, and privileges novelty over correction or synthesis. Once introduced, misinformation persists even after retraction, as corrective posts rarely propagate through the same network paths as the original claims. These phenomena are not anomalies but stable equilibria of attention–optimized systems.

Incremental reforms within this paradigm have focused on moderation, labeling, or algorithmic adjustment. While such interventions may reduce specific harms, they do not alter the underlying dynamics produced by feed ordering and popularity metrics. The coherence–first approach diverges by rejecting the feed as a primary organizing structure and eliminating engagement metrics entirely, rather than attempting to refine their use.

1.2 Community Aggregation and Forum Systems

Platforms such as Reddit and Stack Exchange introduce additional structure through topic segmentation, threaded discussion, and reputation systems. These designs mitigate some issues associated with unstructured feeds by localizing discourse and enabling deeper engagement within bounded communities. However, they continue to rely on post–level interaction and popularity signals, such as upvotes, karma, and accepted answers.

Research on online forums highlights tradeoffs inherent in these systems. Reputation mechanisms can incentivize expertise and sustained participation, but they also encourage strategic behavior and gatekeeping. Threads remain ephemeral, with discussions reset as new posts displace older ones. Knowledge accumulates unevenly and often requires manual curation or external summarization to remain accessible.

The coherence–first model differs by treating discussion topics as persistent semantic objects rather than ephemeral threads. Contributions do not compete for visibility within a feed or forum, but attach directly to evolving objects whose structure constrains how discourse unfolds.

1.3 Federated and Decentralized Networks

Federated platforms such as Mastodon and Bluesky aim to address governance and moderation concerns by decentralizing control. These systems distribute authority across instances or protocols while preserving familiar social media interactions, including posts, feeds, and follower relationships.

Decentralization addresses important issues of platform control and censorship but leaves the semantic architecture largely unchanged. Feed–based delivery, engagement signals, and performative identity remain central. As a result, many of the same coherence and trust problems observed in centralized platforms reappear in federated settings.

The coherence–first approach is orthogonal to decentralization. While compatible with distributed deployment, it targets the semantic and computational substrate rather than the ownership or governance layer. Decentralization without semantic constraints risks replicating existing failure modes at smaller scales.

1.4 Newsletter and Long–Form Publishing Platforms

Platforms such as Substack emphasize long–form content and subscription–based distribution, offering an alternative to short–form social media. By decoupling revenue from engagement metrics and emphasizing author–reader relationships, these systems encourage slower, more deliberate communication.

However, long–form publishing platforms retain a broadcast–oriented model. Content is still produced as discrete artifacts, revisions are limited, and discourse occurs largely through comments or external channels. Semantic relationships between texts are implicit rather than formally represented.

The coherence–first model generalizes beyond long–form publishing by supporting structured evolution of ideas across modalities and time, rather than privileging a single medium or authorial voice.

1.5 Collaborative Knowledge Systems

Wikis represent an early attempt to support cumulative knowledge construction. By allowing collaborative editing of shared documents, wikis reduce fragmentation and encourage synthesis. Version histories provide a limited form of provenance, enabling reversion and accountability.

Despite these strengths, wikis impose a monolithic document structure that struggles to accommodate disagreement, parallel perspectives, and ongoing revision. Semantic relationships between pages are informal, and conflicts are resolved socially rather than mechanically. Empirical studies of wiki governance document persistent tensions between openness and control.

Version control systems such as Git provide more rigorous provenance tracking and support branching and merging of parallel developments. These systems excel at managing structured artifacts such as source code but are ill–suited to general discourse. Line–based diffs and merge conflicts capture syntactic changes rather than semantic ones, and non–textual modalities are poorly supported.

The coherence–first approach draws inspiration from both wikis and version control while addressing their limitations. It generalizes provenance and branching to semantic objects across modalities and introduces entropy–bounded merging to preserve meaning rather than syntax.

1.6 Semantic Web and Knowledge Representation

The semantic web and knowledge representation communities have long sought to formalize meaning through ontologies, graphs, and logical inference. Technologies such as RDF, OWL, and knowledge graphs provide expressive frameworks for representing structured information and relationships.

While powerful, these systems typically require significant upfront formalization and expertise. They are optimized for machine reasoning rather than human discourse and lack mechanisms for incremental, conversational evolution of meaning. As a result, they have seen limited adoption as substrates for everyday communication.

The coherence–first architecture occupies a middle ground. It adopts formal structure where necessary to preserve coherence but does not require full logical formalization of all content. Semantic constraints operate locally and incrementally, allowing discourse to remain flexible while preventing unbounded drift.

1.7 Architectural Alternatives versus Incremental Reform

A recurring theme in prior work is the attempt to repair social media pathologies through incremental changes: improved moderation, transparency tools, algorithmic tweaks, or decentralized governance. While valuable, these efforts accept the core primitives of posts, feeds, and engagement metrics as given.

This paper advances a different stance. It argues that these primitives are the source of systemic failure and must be replaced rather than refined. The coherence–first approach constitutes an architectural alternative, redefining the units of communication, identity, and relevance at the substrate level.

In doing so, it aligns with a growing body of research emphasizing the importance of structural constraints in shaping collective behavior. Rather than optimizing outcomes within a fixed architecture, it seeks to redesign the architecture itself to make certain failure modes structurally impossible.

2 Theoretical Foundations

This section develops the formal foundations of a coherence–first semantic substrate. We introduce semantic objects as the primary units of meaning, define entropy–bounded evolution as a constraint on semantic transformation, specify modality requirements, and formalize provenance as a graph–structured invariant. Together, these elements define a computational framework in which meaning can evolve without unbounded drift.

2.1 Semantic Objects

We begin by defining the fundamental unit of representation.

Definition 1 (Semantic Object). *A semantic object is a structured entity*

$$\sigma = (I, \mathcal{M}, C, P, E)$$

where I is an immutable identity, $\mathcal{M}$ is a finite set of required modalities, C is a partial function assigning content to modalities, P is a provenance graph recording the derivation history of σ, and E is a non–negative real number representing the semantic entropy of the object.

Intuitively, a semantic object represents a unit of meaning that may span multiple modalities, such as text, data, code, audio, or formal proof. Unlike a post or message, a semantic object persists across time and accumulates structure through controlled transformation.

Definition 2 (Well–Formed Semantic Object). *A semantic object σ is well–formed if and only if for all $m \in \mathcal{M}$, the content assignment $C(m)$ is defined.*

Well–formedness enforces modality completeness. For example, a claim that requires both textual explanation and empirical evidence cannot be considered complete until both modalities are populated.

2.2 Examples and Counterexamples

As a motivating example, consider a scientific hypothesis represented as a semantic object with modalities $\{\text{text}, \text{data}, \text{analysis}\}$. A draft hypothesis with only textual content is not well–formed. It becomes well–formed only once supporting data and analysis are attached.

In contrast, a short social media post asserting the hypothesis without evidence would constitute a semantic object with $\mathcal{M} = \{\text{text}\}$, thereby encoding a different and weaker claim. This distinction prevents implicit inflation of meaning and clarifies the scope of assertions.

2.3 Semantic Entropy

We now formalize the notion of semantic entropy.

Definition 3 (Semantic Entropy). *Let σ be a semantic object. The semantic entropy $E(\sigma)$ is a scalar quantity that measures internal inconsistency, ambiguity, or unresolved divergence within the object.*

Semantic entropy is not identical to Shannon entropy, though it may incorporate information–theoretic components. Operationally, it bounds the degree to which a semantic object can tolerate conflicting content, incomplete modalities, or unresolved branches.

Definition 4 (Entropy–Bounded Transformation). *A transformation T applied to a semantic object σ is entropy–bounded if*

$$E(T(\sigma)) \leq E(\sigma) + \varepsilon$$

for a fixed budget $\varepsilon \geq 0$ associated with T.

Entropy budgets constrain how much disorder a transformation may introduce. Transformations that exceed their budgets are rejected or deferred pending reconciliation.

2.4 Semantic Transformations

Transformations operate on semantic objects to produce new versions.

Definition 5 (Semantic Transformation). *A semantic transformation is a partial function*

$$T : \Sigma \to \Sigma$$

defined on the space of semantic objects Σ, subject to modality preservation and entropy constraints.

Examples of transformations include revision, extension, summarization, translation, and formalization. Each transformation specifies its input and output modality requirements and an associated entropy budget.

Proposition 1 (Monotonic Provenance). *Every valid transformation $T(\sigma)$ extends the provenance graph P of σ by adding a new node corresponding to T.*

Proof. By construction, transformations are required to record their application as edges in the provenance graph. Since identities are immutable and transformations are append–only, the graph grows monotonically. □

2.5 Provenance Graph Structure

Provenance is represented as a directed acyclic graph.

Definition 6 (Provenance Graph). *The provenance graph P of a semantic object σ is a directed acyclic graph whose nodes correspond to transformation events and whose edges encode dependency relations between transformations.*

The acyclicity of P ensures that semantic evolution is temporally ordered and that every state has a finite justification history.

Theorem 1 (Traceability). *For any semantic object σ, every content element $C(m)$ admits a finite derivation trace in P.*

Sketch. Since P is acyclic and transformations are finite in number for any realized object, each content element corresponds to a path from an initial state to the current state. □

2.6 Coherence Enforcement

Coherence is enforced mechanically through validation checks.

Definition 7 (Coherent State). *A semantic object σ is coherent if it is well–formed and its semantic entropy does not exceed a system–defined threshold $E_{\max}$.*

Proposition 2 (Rejection of Unbounded Drift). *Unbounded semantic drift is structurally impossible in a system enforcing entropy–bounded transformations.*

Sketch. Suppose an unbounded drift occurs. Then $E(\sigma)$ must increase without bound along a sequence of transformations. This contradicts the existence of fixed entropy budgets and the enforcement of $E_{\max}$. □

2.7 Identity and Persistence

Identity is defined at the level of semantic objects.

Definition 8 (Persistent Identity). *The identity I of a semantic object is immutable and preserved across all valid transformations.*

Deletion is therefore interpreted not as erasure but as a transformation that marks an object as inactive while preserving its provenance.

Lemma 1 (Non–Repudiation). *No valid transformation can remove or alter prior provenance.*

Proof. Transformations are append–only with respect to provenance. Any operation that attempts to remove prior nodes violates the definition of a semantic transformation. □

This completes the formal foundation of coherence–first semantic evolution.

3 Concrete Instantiation and Worked Examples

To clarify how a coherence–first semantic substrate operates in practice, this section presents a series of worked examples. These examples illustrate the lifecycle of semantic objects, the application of entropy–bounded transformations, identity persistence under revision and deletion, and navigation through semantic neighborhoods in contrast to feed–based browsing.

3.1 Lifecycle of a Semantic Object

Consider an initial semantic object σ_0 representing a factual claim. Let the required modality set be

$$\mathcal{M} = \{\text{text}, \text{evidence}\}.$$

At time t_0, only the textual modality is populated. The object is therefore not well–formed.

A transformation T_1 attaches empirical evidence, yielding σ_1 such that $C_{\sigma_1}(\text{evidence})$ is defined. Provided that T_1 satisfies its entropy budget, σ_1 becomes well–formed and eligible for further transformation. The provenance graph records T_1 as a child of σ_0.

A subsequent transformation T_2 revises the textual explanation for clarity. Because T_2 preserves modality completeness and introduces only bounded entropy, the resulting object σ_2 remains coherent. Importantly, σ_0, σ_1, and σ_2 share a common identity and are linked by provenance rather than replacing one another.

3.2 Entropy–Bounded Revision

Revision is a common operation in discourse and often a source of semantic instability in feed–based systems. In the coherence–first model, revision is formalized as a transformation with a tightly constrained entropy budget.

Let T_r be a revision transformation with budget ε_r. Applying T_r to σ yields σ' only if

$$E(\sigma') \leq E(\sigma) + \varepsilon_r.$$

If a proposed revision introduces ambiguity, contradiction, or modality imbalance beyond this threshold, it is rejected or flagged for mediation.

This mechanism distinguishes legitimate refinement from destabilizing alteration. Incremental clarification typically falls within budget, while radical reinterpretation requires explicit branching, preserving traceability.

3.3 Branching and Conflict Resolution

Conflicts arise when incompatible transformations are proposed. In such cases, the provenance graph branches.

Let σ admit two transformations T_a and T_b that cannot be reconciled within a shared entropy budget. The system produces two descendant objects σ_a and σ_b, each inheriting the identity of σ but diverging in provenance.

Rather than forcing premature resolution, the system allows both branches to persist. Subsequent transformations may reconcile the branches by introducing additional evidence, clarification, or scope restriction. Any reconciliation must itself be entropy–bounded and is recorded as a transformation whose provenance references both branches.

This approach contrasts with feed–based systems, where conflicts are resolved implicitly through visibility dynamics rather than explicit semantic structure.

3.4 Deletion and Retraction

Deletion presents a critical edge case. In social media, deletion often erases context, undermining accountability and enabling revisionist narratives.

In a coherence–first system, deletion is modeled as a transformation T_d that marks a semantic object as inactive. The content remains accessible for provenance and reference, but the object is no longer eligible for extension or endorsement.

Retraction is handled similarly but includes an explicit statement of invalidation. A retraction transformation increases semantic entropy temporarily while introducing constraints that prevent further use of the object as a premise. This preserves historical traceability while signaling reduced validity.

3.5 Identity Persistence Across Time

Identity persistence follows directly from the immutability of semantic object identities. All transformations preserve identity, ensuring that long–term evolution remains traceable even as content changes.

This property supports longitudinal trust. Observers can inspect how an object has changed, which transformations were applied, and which branches were reconciled or abandoned. Trust is therefore grounded in demonstrated behavior over time rather than static declarations or popularity signals.

3.6 Semantic Neighborhood Navigation

Discovery in a coherence–first system operates through semantic neighborhoods rather than feeds. Users encounter semantic objects by traversing relationships defined by shared modalities, transformation types, or provenance connections.

Navigation proceeds through deliberate exploration rather than passive consumption. Because objects persist and accumulate structure, older but coherent material remains discoverable without being displaced by newer content.

This mode of interaction privileges understanding over immediacy and reduces the attention fragmentation characteristic of infinite scroll interfaces.

3.7 Comparison with Feed Browsing

In feed–based systems, relevance decays rapidly with time, and visibility is governed by engagement metrics. In contrast, semantic neighborhoods decouple relevance from recency. Objects remain accessible as long as they remain coherent.

This shift alters user incentives. Contributions that improve coherence, clarify meaning, or reconcile conflict are rewarded structurally, while reactive or sensational content gains no inherent advantage.

These examples demonstrate that the coherence–first model is not merely theoretical but operationally viable, offering concrete mechanisms for discourse, revision, and discovery without reliance on feeds or engagement metrics.

4 Challenges, Limitations, and Structural Tradeoffs

The coherence–first semantic substrate described in this paper is not merely an alternative content organization scheme, but a reconfiguration of social interaction around spatial, computational, and educational primitives. In this section, we revisit challenges and limitations under the assumption that the platform functions simultaneously as a programming environment, a persistent semantic memory, and a navigable social space embedded in a high–dimensional latent geometry.

4.1 Scalability in a Spatial Semantic Medium

Concerns regarding scalability are often framed in terms of throughput and user count, implicitly assuming feed–based dissemination as the baseline. In a spatial semantic system, scale is expressed differently. Growth manifests as expansion of a latent semantic manifold rather than increased velocity within a global feed.

Knowledge objects are not broadcast to all users. Instead, they occupy stable locations in semantic space. As the system grows, density increases locally rather than globally, and users encounter content by navigating regions of interest. This mirrors physical environments, where cities scale by adding neighborhoods rather than forcing all activity through a single thoroughfare.

This spatialization fundamentally alters adversarial scaling dynamics. Engagement–driven platforms monetize attention by selling advertising inventory, frequently to actors whose business models depend on deception, including known scam networks. These platforms therefore benefit financially from maximizing exposure, regardless of semantic quality. In contrast, a spatial semantic medium offers no global amplification mechanism to exploit. Content remains localized unless structurally integrated into broader semantic regions.

4.2 Onboarding Through Programming as Literacy

A frequent critique of structured systems is their perceived difficulty for newcomers. However, the platform under consideration explicitly incorporates programming education as a first–class function through the SpherePOP calculus. Users are not expected to master formal systems upfront; rather, they learn by interacting with semantic objects whose behavior is governed by simple, composable rules.

Programming in this context is not an auxiliary skill but a literacy: the ability to construct, transform, and navigate meaning. By embedding programming instruction directly into social interaction, the system converts what is traditionally a barrier into a continuous learning process.

This contrasts sharply with existing platforms, where users are incentivized to spend substantial time curating follow lists, monitoring engagement metrics, and pursuing arbitrary follower goals. Such activity consumes attention without producing durable knowledge or skill. In the coherence–first system, time invested in learning the substrate directly increases one's expressive and navigational capacity.

4.3 Persistence, Memory, and Reaccessibility

Semantic persistence introduces a distinct limitation: content cannot be trivially forgotten. However, this persistence enables a powerful alternative to feed consumption. Knowledge is stored in stable locations within the semantic latent space and can be reaccessed by returning to the same region.

This property transforms memory from a chronological archive into a spatial one. Users do not scroll backward through time but revisit conceptual neighborhoods. Over time, these neighborhoods accrete structure, becoming richer rather than buried.

While this challenges habitual consumption patterns shaped by infinite scroll interfaces, it aligns more closely with how expertise and understanding develop. The cost is reduced novelty throughput; the benefit is cumulative intelligibility.

4.4 Adversarial Behavior in a Navigable Space

Adversarial behavior in a spatial semantic system differs qualitatively from that in broadcast platforms. Without feeds or engagement metrics, attackers cannot leverage visibility cascades. Instead, malicious content must occupy space, persist, and withstand entropy constraints.

Attempts to game entropy budgets or flood regions with low–quality objects are constrained by both computational and navigational friction. Users encounter such content only by entering the relevant regions, and repeated incoherent transformations accumulate entropy that eventually blocks further evolution.

Crucially, the absence of an advertising–driven revenue model removes the primary financial incentive for large–scale spam. Traditional platforms profit from keeping users engaged, even if that engagement consists of navigating scams, outrage, or empty status games. A spatial semantic platform derives no benefit from time–wasting behavior oriented around following, metric optimization, or performative presence.

4.5 Expression, Revision, and Spatial Accountability

Freedom of expression in a spatial semantic medium is preserved through transformation rather than erasure. Users may revise, retract, or contextualize prior contributions, but these actions are recorded as movements or transformations within semantic space.

This introduces a form of spatial accountability. Past statements remain accessible at their original locations, while newer interpretations may occupy adjacent or higher–order regions. Growth is represented geometrically rather than narratively.

While this model may feel restrictive to users accustomed to disappearing content, it reflects norms already present in scholarly and technical domains, where retracted or superseded work remains part of the record.

4.6 Migration and Hybrid Use

Migration from existing platforms remains nontrivial, but the spatial model enables hybrid strategies. External content can be imported as semantic objects anchored to regions of space that reflect their topical and structural properties. Over time, these imports may be extended, formalized, or abandoned.

Notably, the system does not attempt to import engagement artifacts such as likes, shares, or follower counts. These signals have no spatial or semantic meaning and are therefore discarded. Users migrating into the system are freed from the obligation to maintain continuous visibility or pursue follower accumulation as a proxy for participation.

4.7 Summary of Tradeoffs

The coherence–first, spatial semantic platform prioritizes cumulative understanding, programming literacy, and persistent memory over immediacy and attention capture. It is therefore incompatible with advertising–driven social media models and the behavioral patterns they encourage.

Its limitations are real: increased cognitive demand, slower initial growth, and a requirement that users engage actively with structure. Its advantages are equally real: durable knowledge, teachable computation, and a social medium in which time spent produces skill and understanding rather than exhaustion.

5 Formalization and Quantitative Analysis

This section formalizes the central claims of the coherence–first spatial semantic architecture and analyzes its properties using tools from information theory, dynamical systems, and computational complexity. We compare feed–based dissemination with semantic neighborhood navigation, model trust accumulation under persistence, and establish convergence guarantees under entropy–bounded evolution.

5.1 Preliminaries

Let Σ denote the space of all semantic objects, and let X denote the semantic latent manifold in which objects are embedded. Each semantic object $\sigma \in \Sigma$ is associated with an embedding $\phi(\sigma) \in X$, a provenance graph $P(\sigma)$, and a semantic entropy value $E(\sigma)$.

Users interact with the system by navigating X, applying transformations to semantic objects, and composing programs in a structured transformation language.

5.2 Feed–Based Dissemination versus Semantic Neighborhoods

We first contrast feed–based dissemination with spatial semantic navigation.

Definition 9 (Feed Model). *A feed–based system presents users with a sequence*

$$F_u(t) = \langle c_1, c_2, \dots, c_n \rangle$$

ordered by a relevance function $R(c, u, t)$ optimized for engagement metrics such as clicks, likes, or dwell time.

Definition 10 (Semantic Neighborhood Model). *A semantic neighborhood system presents users with content*

$$N_u(x, r) = \{\sigma \in \Sigma \mid d(\phi(\sigma), x) \le r\}$$

where $x \in X$ is the user's current location in semantic space and r is a bounded radius.

Proposition 3 (Entropy Accumulation in Feeds). *Feed–based dissemination induces unbounded semantic entropy accumulation over time.*

Sketch. In feed systems, content relevance decays with time and is replaced by new content optimized for engagement rather than coherence. There is no constraint enforcing semantic consistency across items in $F_u(t)$. As $t \to \infty$, the expected semantic divergence between consecutive items grows without bound, yielding unbounded entropy. □

Proposition 4 (Entropy Localization in Semantic Neighborhoods). *In a semantic neighborhood system, entropy accumulation is locally bounded.*

Sketch. Semantic neighborhoods restrict interaction to bounded regions of X. Transformations applied within a neighborhood must satisfy entropy budgets. Therefore, entropy growth is constrained locally and does not propagate globally. □

5.3 Information–Theoretic Comparison

Let $H(F_u)$ denote the entropy of a user's feed and $H(N_u)$ the entropy of a semantic neighborhood.

Theorem 2 (Asymptotic Entropy Divergence). *For feed–based systems,*

$$\lim_{t \to \infty} H(F_u(t)) = \infty,$$

whereas for semantic neighborhoods,

$$\sup_t H(N_u(x, t)) \le H_{\max}$$

for some finite bound $H_{\max}$.

Sketch. Feed entropy increases as content turnover introduces uncorrelated items optimized for engagement. Semantic neighborhoods impose correlation through spatial proximity and entropy–bounded transformations, yielding a finite upper bound. □

This result formalizes the intuition that feeds maximize novelty while neighborhoods preserve structure.

5.4 Trust Accumulation: Persistence versus Popularity

We now model trust accumulation.

Definition 11 (Popularity–Based Trust). *In engagement systems, trust is approximated by a function*

$$T_{pop}(u,t) = f(L_u(t), S_u(t), F_u(t))$$

where L denotes likes, S shares, and F follower count.

Definition 12 (Persistence–Based Trust). *In a coherence–first system, trust is defined as*

$$T_{pers}(u,t) = g(|P_u(t)|, \Delta E_u(t), C_u(t))$$

where $|P_u|$ is the size of the user's provenance graph, ΔE_u measures entropy stability over time, and C_u denotes successful reconciliations.

Proposition 5 (Instability of Popularity Trust). *Popularity–based trust is unstable under adversarial optimization.*

Sketch. Engagement metrics can be manipulated through coordinated activity, bot networks, or sensational content. Small perturbations in visibility can cause large changes in T_{pop}, rendering it unreliable. □

Theorem 3 (Monotonicity of Persistence Trust). *Persistence–based trust is monotonic under valid transformations.*

Proof. Valid transformations append to provenance and are entropy–bounded. Therefore, $|P_u|$ and reconciliation counts increase monotonically, while entropy stability penalizes destabilizing behavior. □

5.5 Convergence and Stability under Entropy Bounds

We model semantic evolution as a dynamical system.

Definition 13 (Semantic Evolution Operator). *Let $\mathcal{T}$ denote the set of admissible transformations. Define*

$$\mathcal{E}_{t+1} = \mathcal{T}(\mathcal{E}_t)$$

subject to entropy constraints.

Theorem 4 (Convergence of Semantic Objects). *Under fixed entropy budgets, semantic object evolution converges to stable attractors in* Σ.

Sketch. Entropy budgets impose a Lyapunov–like constraint on semantic evolution. Since entropy cannot increase without bound and transformations are finite, sequences converge to fixed points or bounded cycles representing stable interpretations. □

This establishes long–term stability absent from feed–based systems.

5.6 Programming Literacy and Learning Efficiency

Let $L_u(t)$ denote a user's programming literacy as measured by successful semantic transformations composed in the transformation language.

Proposition 6 (Positive Learning Gradient). *In a system where social interaction is mediated through programmable transformations, expected programming literacy increases monotonically with use.*

Sketch. Each interaction requires the application or observation of typed transformations. Unlike feed consumption, which yields no cumulative skill, repeated interaction increases the user's effective action space. □

This contrasts with engagement systems, where time spent pursuing follower goals yields no durable competence.

5.7 Computational Complexity

We conclude with complexity analysis.

Let n be the number of semantic objects, k the maximum degree of provenance graphs, and m the number of transformations.

Proposition 7 (Transformation Complexity). *Applying a transformation has time complexity* $O(k)$ *and space complexity* $O(1)$.

Proposition 8 (Neighborhood Query Complexity). *Semantic neighborhood queries execute in* $O(\log n + |N_u|)$ *using standard spatial indexing.*

Theorem 5 (Global Scalability). *Total system complexity grows sublinearly with user count under bounded neighborhood interaction.*

Sketch. Users interact locally in semantic space. No global feed requires recomputation or ranking across all objects. Therefore, system load scales with local density rather than total population. □

This completes the formal and quantitative analysis of the coherence–first semantic architecture.

6 Artificial Intelligence as a Typed Semantic Operator

In the coherence–first semantic architecture, artificial intelligence is neither a primary agent nor a curator of attention. Instead, AI functions as a constrained operator within the semantic substrate, performing well–defined transformations under explicit structural and entropy constraints. This section formalizes the role of AI and contrasts it with the recommender–centric models that dominate contemporary social media.

6.1 Rejection of AI as a Relevance Oracle

Most social media platforms deploy AI primarily as a relevance oracle: a system that predicts which content will maximize user engagement. This framing aligns AI incentives with time spent, emotional arousal, and interaction volume. As a result, recommender systems become amplifiers of whatever content best exploits human cognitive biases, regardless of semantic quality.

The coherence–first architecture explicitly rejects this role. AI systems are not tasked with ranking users, optimizing feeds, or predicting attention. There is no global feed to optimize, and no engagement metric to maximize. Consequently, entire classes of alignment failures associated with recommender systems are rendered inapplicable.

6.2 Typed Semantic Operations

AI operates exclusively through typed semantic transformations.

Definition 14 (Typed Semantic Operator). *A typed semantic operator is a transformation*

$$T : (\Sigma, \mathcal{M}_{in}) \to (\Sigma, \mathcal{M}_{out})$$

that maps semantic objects with input modality requirements $\mathcal{M}_{in}$ to objects with output modality requirements $\mathcal{M}_{out}$, subject to fixed entropy budgets.

Examples include summarization, translation, modality completion, formalization, compression, and reconciliation assistance. Each operator declares the modalities it consumes and produces, enabling static validation prior to execution.

6.3 Entropy Budgets and Validation

Every AI operator is associated with an entropy budget ε_T. After application, the resulting semantic object is evaluated for coherence.

Proposition 9 (Post–Transformation Validation). *An AI–generated transformation is accepted if and only if the resulting semantic object is well–formed and satisfies the entropy bound*

$$E(T(\sigma)) \leq E(\sigma) + \varepsilon_T.$$

If validation fails, the transformation is rejected or returned for human mediation. AI therefore cannot unilaterally inject ambiguity, contradiction, or scope inflation into the system.

6.4 Failure Modes and Recovery

AI transformations may fail due to insufficient input structure, modality mismatch, or exceeding entropy budgets. Rather than silently degrading output quality, such failures are explicitly surfaced.

Failed transformations generate diagnostic artifacts indicating missing modalities, uncertain mappings, or unresolved inconsistencies. These artifacts become semantic objects in their own right, enabling users to inspect and address the failure.

This contrasts with engagement systems, where AI failures are often invisible and manifest only indirectly through degraded discourse quality.

6.5 Training Data and Bias Mitigation

Because AI operators perform constrained transformations rather than open–ended generation, training data requirements differ substantially from those of large language models deployed as conversational agents. Training focuses on mapping between modalities and preserving semantic invariants rather than imitating human conversational behavior.

Bias mitigation is addressed structurally. Since AI output must satisfy explicit semantic constraints and provenance requirements, biased or misleading transformations are more readily detectable. Moreover, the absence of engagement optimization removes incentives to exploit polarizing or sensational patterns present in training data.

6.6 Human–AI Collaboration

AI in a coherence–first system functions as a collaborator rather than a substitute for human judgment. It assists users in navigating semantic space, identifying inconsistencies, and performing routine transformations, but it does not author meaning independently.

This role aligns AI incentives with clarity and stability rather than persuasion or attention capture. By constraining AI to typed operations with explicit budgets, the system ensures that human agency and accountability remain central.

6.7 Summary

By redefining AI as a typed semantic operator rather than a relevance oracle, the coherence–first architecture avoids the alignment failures endemic to engagement–driven systems. AI becomes a tool for maintaining semantic integrity and supporting learning, including programming literacy, rather than a mechanism for amplifying noise or optimizing visibility.

7 Governance, Invariants, and Forking Mechanics

Governance in feed–based social media platforms is typically implemented as an external layer: policies, moderators, enforcement actions, and appeals. These mechanisms operate reactively, intervening after semantic damage has occurred and often at scales that render consistent enforcement impractical. In contrast, the coherence–first semantic architecture embeds governance into the formal structure of the system itself. This section specifies the invariants that define admissible behavior, the procedural handling of conflict and divergence, and the mechanics of forking as a first–class operation.

7.1 Governance by Structural Invariants

At the core of the system is a set of invariants that all valid semantic states must satisfy. These invariants are enforced mechanically rather than socially.

Definition 15 (Semantic Invariant). *A semantic invariant is a property $I(\sigma)$ of a semantic object σ that must hold for σ to be considered valid within the system.*

Examples include modality completeness, provenance monotonicity, and bounded semantic entropy. Violations of invariants do not trigger punitive action; instead, they prevent the object from participating in further transformations until resolved.

Proposition 10 (Invariant–Based Governance). *All admissible system states satisfy the set of semantic invariants.*

Proof. Invariant checks are applied at the time of transformation. Since invalid states are rejected by construction, no reachable system state violates the invariants. □

This approach shifts governance from discretionary enforcement to constraint satisfaction.

7.2 Semantic Reconciliation

Disagreement is an expected feature of discourse. The coherence–first system does not attempt to eliminate disagreement but to represent it explicitly.

Definition 16 (Semantic Reconciliation). *Semantic reconciliation is a transformation that integrates divergent branches of a semantic object by introducing additional structure that resolves or contextualizes their differences.*

Reconciliation may involve narrowing scope, introducing conditionality, or adding supporting modalities. Crucially, reconciliation is optional rather than mandatory; the system permits persistent divergence when resolution is not possible or desirable.

Proposition 11 (Non–Coercive Resolution). *No semantic object is required to reconcile divergent branches.*

Proof. Branching is a valid terminal state. The absence of a reconciliation transformation does not violate any invariant. □

This ensures that governance does not collapse into enforced consensus.

7.3 Forking as a First–Class Operation

Forking is treated as a constructive operation rather than a failure mode.

Definition 17 (Semantic Fork). *A semantic fork is the creation of two or more descendant semantic objects that diverge from a common ancestor due to incompatible transformations.*

Forks preserve shared provenance up to the point of divergence and then evolve independently. Unlike moderation–based deletion or suppression, forking externalizes conflict rather than obscuring it.

7.4 Tracking Divergence

Divergence is tracked explicitly in the provenance graph. Users navigating semantic space can observe where and why forks occurred, inspect the transformations that produced them, and choose which branches to engage with.

This transparency contrasts with feed–based systems, where divergence is often masked by algorithmic filtering or visibility dynamics.

7.5 Coordinated Attacks and Structural Resistance

Coordinated attacks, such as brigading or disinformation campaigns, exploit amplification mechanisms in engagement–driven systems. The coherence–first architecture removes these mechanisms entirely.

Without feeds, likes, or shares, attackers cannot leverage visibility cascades. Malicious content must persist, occupy semantic space, and withstand invariant checks. Coordinated low–quality transformations accumulate entropy and exhaust available budgets, limiting their impact.

7.6 Emergent Norms

While formal invariants define the boundaries of admissible behavior, norms emerge through use. Because semantic objects persist and interactions are traceable, norms stabilize over time rather than being enforced episodically.

This separation of formal constraints from emergent norms allows the system to accommodate diverse communities without imposing uniform standards beyond those required for semantic coherence.

7.7 Summary

Governance in a coherence–first semantic substrate is achieved through invariants, transparent reconciliation, and explicit forking rather than moderation and suppression. By embedding governance into the structure of meaning itself, the system avoids many of the failure modes associated with discretionary enforcement and algorithmic opacity.

8 Implementation Considerations

While the coherence–first semantic architecture is motivated by theoretical and information–theoretic considerations, its viability depends on concrete implementation choices. This section outlines practical considerations related to storage, identity, networking, and performance. The goal is not to prescribe a single implementation, but to demonstrate that the proposed architecture is computationally feasible with existing technology.

8.1 Storage of Semantic Objects and Provenance

Semantic objects are persistent, immutable entities whose evolution is recorded through append–only provenance graphs. This naturally suggests a storage model based on content–addressed data structures. Each semantic object state can be identified by a cryptographic hash of its content, modality assignments, and provenance references.

Provenance graphs grow monotonically but typically exhibit low branching factors in practice. Since transformations are localized and entropy–bounded, most objects evolve through short, interpretable chains rather than deep or highly branching structures. This enables efficient storage through structural sharing, similar to persistent data structures used in functional programming languages.

Garbage collection is unnecessary in the traditional sense, as objects are never mutated or deleted. Instead, inactive or superseded objects can be indexed separately or cached at lower priority without violating persistence guarantees.

8.2 Indexing and Semantic Space Navigation

Efficient navigation of semantic neighborhoods requires spatial indexing over the latent manifold X. Standard techniques such as approximate nearest–neighbor search, metric trees, or locality–sensitive hashing can be employed to support neighborhood queries.

Because users interact locally within semantic space, query complexity depends primarily on local density rather than global object count. This property enables horizontal scaling by partitioning semantic space across nodes while preserving navigability.

Importantly, indexing operates over semantic embeddings rather than engagement metrics. This eliminates the need for continuous global re-ranking and reduces system–wide recomputation.

8.3 Identity Anchoring and Authentication

Identity in the coherence–first system is defined by persistent identifiers rather than profiles or accounts. Cryptographic primitives can be used to anchor identity securely. Each identity may be associated with a public key, and transformations can be signed to establish authorship without revealing unnecessary personal information.

This model supports pseudonymity while preserving accountability. Because identity is anchored structurally rather than socially, reputation laundering through account recreation or metric manipulation is ineffective.

8.4 Distributed Deployment and Federation

The architecture is compatible with distributed and federated deployment. Semantic space can be partitioned across nodes or instances, with each node responsible for maintaining a region of the latent manifold and the objects embedded within it.

Federation does not require global agreement on ranking or moderation policies. Nodes need only agree on core invariants governing semantic object validity and transformation. This reduces coordination overhead and allows heterogeneous communities to coexist without fragmenting the substrate.

Cross–node interaction is mediated through provenance references and embedding mappings, ensuring that semantic integrity is preserved even when objects traverse administrative boundaries.

8.5 Performance Characteristics

Performance in engagement–driven platforms is dominated by feed ranking and recommendation pipelines, which require continuous large–scale computation. In contrast, the coherence–first architecture shifts computation toward transformation validation and spatial queries.

Transformation validation is localized and bounded in complexity, as entropy budgets and modality checks constrain the scope of evaluation. Spatial queries benefit from standard indexing techniques and do not require system–wide recomputation.

As a result, system load scales with active semantic regions rather than total user count or content volume. This property supports long–term growth without the runaway computational costs associated with global feeds.

8.6 Programming Interface and SpherePOP Integration

The integration of a programmable transformation language, such as SpherePOP, introduces additional implementation considerations. Programs operate over semantic objects and must respect modality and entropy constraints. Static analysis can be employed to verify that programs are well–typed and cannot violate invariants.

Because programming constructs are embedded directly into social interaction, tooling for debugging, visualization, and explanation is essential. However, the simplicity of the underlying calculus enables incremental learning and compositional reuse.

By treating programming as a medium of interaction rather than a specialized activity, the system aligns implementation complexity with user skill development rather than hiding it behind opaque automation.

8.7 Summary

The coherence–first semantic architecture can be implemented using existing storage, indexing, cryptographic, and distributed systems techniques. Its performance profile differs substantially from that of engagement–driven platforms, emphasizing locality, persistence, and validation over global optimization. These properties make it suitable for scalable deployment without reproducing the incentive structures that undermine contemporary social media.

9 Conclusion and Research Agenda

This paper has argued that the dominant failures of contemporary social media platforms are not incidental defects, but necessary consequences of their underlying architecture. Feed–based dissemination, engagement optimization, and performative identity together produce high–entropy dynamics that degrade meaning, incentivize time–wasting behavior, and erode trust at scale. Attempts to repair these systems through moderation, transparency, or algorithmic adjustment leave the core substrate unchanged and therefore fail to address the root causes.

In response, we have outlined a coherence–first alternative grounded in persistent semantic objects, entropy–bounded evolution, provenance–anchored identity, and spatial navigation of meaning. Rather than treating communication as a stream of posts optimized for visibility, the proposed architecture treats meaning as a structured entity that evolves through constrained transformations. Trust emerges from persistence and consistency over time rather than popularity or attention metrics.

We formalized the key components of this architecture, including semantic objects, transformation operators, entropy budgets, and provenance graphs. We demonstrated that feed–based systems exhibit unbounded entropy accumulation, while semantic neighborhoods localize and constrain disorder. We showed that trust models based on persistence are structurally more stable than those based on popularity, and that entropy–bounded evolution yields convergence properties absent from engagement–driven platforms. We further analyzed computational complexity and demonstrated that the system scales through locality rather than global optimization.

Importantly, the architecture reframes the role of artificial intelligence. By constraining AI to typed semantic operations under explicit budgets, the system avoids the alignment failures associated with recommender systems and attention optimization. AI becomes a tool for maintaining coherence, assisting learning, and supporting programming literacy rather than amplifying noise or persuasion.

The integration of a programmable semantic calculus transforms social interaction into a learning process. Time spent within the system produces durable skills and knowledge rather than ephemeral engagement. Knowledge persists spatially and can be reaccessed by returning to the same region of semantic space, enabling cumulative understanding rather than chronological decay.

9.1 Open Problems and Future Work

Several open problems remain. First, while entropy–bounded evolution provides strong stability guarantees, further work is needed to refine entropy metrics that balance flexibility with constraint across diverse domains. Second, the design of user interfaces for high–dimensional semantic navigation remains an active area of research, particularly with respect to accessibility and cognitive load. Third, empirical studies are required to evaluate learning outcomes, trust formation, and discourse quality in deployed systems.

Additional work is needed to explore incentive models compatible with coherence–first design. Advertising–driven revenue is fundamentally misaligned with persistence and semantic integrity. Alternative models, such as patronage, institutional support, or public infrastructure funding, warrant systematic investigation.

Finally, the relationship between coherence–first platforms and existing digital ecosystems remains to be fully articulated. Hybrid deployments, migration strategies, and interoperability with legacy systems present both technical and social challenges.

9.2 Broader Implications

Beyond social media, the coherence–first architecture has implications for knowledge management, education, scientific collaboration, and governance. Any domain in which meaning must persist, evolve, and remain accountable over time stands to benefit from a substrate that prioritizes structure over visibility.

By rethinking the primitives of digital interaction, this work contributes to a growing body of research that treats platform design as a problem of semantics and systems architecture rather than content moderation alone. It suggests that healthier digital public spheres require not better incentives layered atop existing systems, but new substrates in which certain failure modes are structurally impossible.

9.3 Closing Remarks

The central claim of this paper is that coherence is not an emergent property of engagement–optimized systems, but a design choice that must be encoded at the substrate level. A coherence–first semantic architecture offers a viable alternative to social media as currently constituted, one in which time spent produces understanding rather than exhaustion, and where trust is earned through persistence rather than performance.

We invite further research, experimentation, and critique of this approach as part of a broader effort to reimagine the foundations of digital communication.

Appendices

A Formal Semantic Object Model

A.1 Primitive Sets

Let:

$$\mathcal{I} \text{ be a countable set of immutable identifiers}$$

$$\mathcal{M} \text{ be a finite set of modality labels}$$

$$\mathcal{C} \text{ be a space of modality-indexed content realizations}$$

$$\mathcal{P} \text{ be the space of finite directed acyclic graphs}$$

$$\mathbb{R}_{\geq 0} \text{ be the non-negative reals}$$

A.2 Semantic Object Definition

Definition 18 (Semantic Object). *A semantic object is a tuple*

$$\sigma = (i, \mathcal{M}_\sigma, C_\sigma, P_\sigma, E_\sigma)$$

where:

$$\begin{aligned} i &\in \mathcal{I} \quad \textit{(immutable identity)} \\ \mathcal{M}_\sigma &\subseteq \mathcal{M} \quad \textit{(required modalities)} \\ C_\sigma &: \mathcal{M}_\sigma \rightharpoonup \mathcal{C} \quad \textit{(partial content map)} \\ P_\sigma &\in \mathcal{P} \quad \textit{(provenance DAG)} \\ E_\sigma &\in \mathbb{R}_{\geq 0} \quad \textit{(semantic entropy)} \end{aligned}$$

A.3 Well-Formedness

Definition 19 (Modal Completeness). *A semantic object σ is modally complete iff:*

$$\forall m \in \mathcal{M}_\sigma,\ C_\sigma(m) \downarrow$$

where $\downarrow$ denotes definedness.

Definition 20 (Provenance Validity). *A provenance graph $P_\sigma = (V, E)$ is valid iff:*

$$P_\sigma \textit{ is finite, directed, and acyclic}$$

Definition 21 (Well-Formed Semantic Object). *A semantic object σ is well-formed iff:*

$$\textit{ModalCompleteness}(\sigma) \ \wedge \ \textit{ProvenanceValidity}(P_\sigma)$$

A.4 Semantic State Space

Definition 22 (Semantic State Space). *Define the semantic state space as:*

$$\Sigma = \{\sigma \mid \sigma \text{ is well-formed}\}$$

A.5 Identity Invariance

Definition 23 (Identity Preservation). *For any transformation $T : \Sigma \to \Sigma$,*

$$T(\sigma) = \sigma' \implies i_{\sigma'} = i_\sigma$$

Theorem 6 (Identity Invariance). *Identity is invariant under all admissible transformations.*

Proof. By definition, transformations operate on content, provenance, and entropy only. The identity component is immutable and not in the transformation domain. □

—

A.6 Content Extensionality

Definition 24 (Extensional Equality). *Two semantic objects σ_1, σ_2 are extensionally equal iff:*

$$i_1 = i_2 \ \wedge \ C_{\sigma_1} = C_{\sigma_2} \ \wedge \ P_{\sigma_1} = P_{\sigma_2}$$

Proposition 12 (No Implicit Overwrite). *If $\sigma_1 \neq \sigma_2$ extensionally, then σ_1 cannot overwrite σ_2.*

A.7 Structural Consequences

Theorem 7 (No Atomic Posts). *There exists no semantic object with empty provenance and mutable identity.*

Proof. Provenance monotonicity and identity immutability prohibit atomic overwrite semantics. □

Theorem 8 (No Context-Free Meaning). *There exists no $\sigma \in \Sigma$ such that C_σ is defined independently of P_σ.*

A.8 Remarks

All social and computational interaction in the system is reducible to operations over Σ subject to the constraints defined above.

B Semantic Entropy Measure and Bounds

B.1 Preliminaries

Let Σ denote the space of well-formed semantic objects defined in Appendix A. For $\sigma \in \Sigma$, recall:

$$\sigma = (i, \mathcal{M}_\sigma, C_\sigma, P_\sigma, E_\sigma).$$

Let Ω be a measurable space of semantic interpretations, and let

$$\mu_\sigma : \Omega \to [0, 1]$$

be a probability measure induced by the content and provenance of σ.

B.2 Semantic Entropy Definition

Definition 25 (Semantic Entropy Functional). *Define semantic entropy as:*

$$E(\sigma) := H(\mu_\sigma) = -\int_\Omega \mu_\sigma(\omega) \log \mu_\sigma(\omega) \, d\omega$$

where H denotes Shannon entropy.

Definition 26 (Conditional Semantic Entropy). *Given provenance P_σ, define:*

$$E(\sigma \mid P_\sigma) := H(\mu_\sigma \mid P_\sigma).$$

This quantity measures ambiguity remaining after accounting for derivational context.

B.3 Entropy Decomposition

Lemma 2 (Entropy Decomposition). *For any $\sigma \in \Sigma$,*

$$E(\sigma) = E(C_\sigma) + E(P_\sigma) - I(C_\sigma; P_\sigma)$$

where I is mutual information.

Proof. Direct from Shannon entropy identities. □

B.4 Entropy-Bounded Transformations

Definition 27 (Transformation Operator). *A semantic transformation is a function:*

$$T : \Sigma \to \Sigma$$

such that identity and provenance monotonicity are preserved.

Definition 28 (Entropy Budget). *Each transformation T is associated with a fixed constant $\varepsilon_T \geq 0$.*

Definition 29 (Entropy-Bounded Transformation). *T is entropy-bounded iff:*

$$E(T(\sigma)) \leq E(\sigma) + \varepsilon_T.$$

B.5 Additivity and Subadditivity

Lemma 3 (Entropy Subadditivity). *For a sequence of transformations (T_k) applied to σ_0,*

$$E(\sigma_n) \leq E(\sigma_0) + \sum_{k=1}^{n} \varepsilon_{T_k}.$$

Proof. By induction on n using the entropy-bounded condition. □

B.6 Lyapunov Stability

Definition 30 (Semantic Lyapunov Function). *Define $V(\sigma) := E(\sigma)$.*

Theorem 9 (Semantic Stability). *If $\sum_k \varepsilon_{T_k} < \infty$, then (σ_k) converges in entropy.*

Proof. $V(\sigma_k)$ is monotonically bounded above and hence convergent. □

B.7 Entropy Laundering Impossibility

Definition 31 (Entropy Laundering Attempt). *An entropy laundering attempt is a finite sequence of transformations (T_k) such that:*

$$E(\sigma_n) < E(\sigma_0)$$

while increasing semantic ambiguity.

Theorem 10 (No Entropy Laundering). *Entropy laundering is impossible under entropy-bounded transformations.*

Proof. Entropy reduction requires negative ε_T, which is disallowed. Any reduction in $E(\sigma)$ must arise from added structure or information, not reordering or deletion. □

–

B.8 Branching and Entropy

Lemma 4 (Branch Entropy Conservation). *Let $\sigma \to \{\sigma_1, \sigma_2\}$ be a fork. Then:*

$$E(\sigma_1) + E(\sigma_2) \geq E(\sigma).$$

Proof. Branching introduces independent uncertainty components; entropy is not destroyed. □

B.9 Entropy Caps

Definition 32 (Global Entropy Cap). *A system-wide constant $E_{\max}$ such that:*

$$E(\sigma) \leq E_{\max} \quad \forall \sigma \in \Sigma.$$

Theorem 11 (Structural Drift Prevention). *Unbounded semantic drift is structurally impossible if $E_{\max} < \infty$.*

Proof. Immediate from boundedness of $V(\sigma)$. □

B.10 Consequences

[No Viral Collapse] No semantic object can accumulate arbitrarily high ambiguity via repeated exposure.

[No Engagement-Driven Explosion] Visibility amplification without semantic structure cannot increase entropy budget.

C Provenance Graph Structure

C.1 Graph-Theoretic Preliminaries

Let $\mathcal{G}$ denote the class of finite directed graphs.

Definition 33 (Directed Acyclic Graph). *A directed graph $G = (V, E)$ is a DAG iff:*

$$\forall v \in V, \ \nexists \text{ directed cycle through } v.$$

Let $\prec$ denote the strict partial order induced by reachability in a DAG.

C.2 Provenance Graph Definition

Definition 34 (Provenance Graph). *For a semantic object σ, its provenance graph is*

$$P_\sigma = (V_\sigma, E_\sigma)$$

where:

$$\begin{aligned} V_\sigma &\subset \Sigma \quad \text{(semantic states)} \\ E_\sigma &\subset V_\sigma \times V_\sigma \quad \text{(typed transformations)} \end{aligned}$$

and P_σ is a DAG.

C.3 Causal Ordering

Definition 35 (Causal Precedence). *For $\sigma_i, \sigma_j \in V_\sigma$,*

$$\sigma_i \prec \sigma_j \iff \exists \text{ directed path } \sigma_i \to \sigma_j.$$

Lemma 5 (Irreflexivity).

$$\neg(\sigma \prec \sigma)$$

Lemma 6 (Transitivity).

$$(\sigma_i \prec \sigma_j) \land (\sigma_j \prec \sigma_k) \Rightarrow (\sigma_i \prec \sigma_k).$$

Thus $\prec$ defines a strict partial order.

C.4 Monotonicity Invariant

Definition 36 (Provenance Monotonicity). *A provenance graph satisfies monotonicity iff:*

$$\forall(\sigma_i \to \sigma_j) \in E_\sigma, \quad V_{\sigma_i} \subset V_{\sigma_j}.$$

Theorem 12 (Monotonic Growth). *Provenance graphs grow monotonically under all admissible transformations.*

Proof. Transformations append nodes and edges without deletion by definition. □

C.5 Forking

Definition 37 (Fork). *A fork occurs when a node σ has two outgoing edges:*

$$\sigma \to \sigma_1, \quad \sigma \to \sigma_2$$

such that $\sigma_1 \not\prec \sigma_2$ and $\sigma_2 \not\prec \sigma_1$.

Lemma 7 (Fork Non-Equivalence). *Forked descendants are not extensionally equal:*

$$\sigma_1 \neq \sigma_2.$$

C.6 Merge (Reconciliation)

Definition 38 (Merge Operator). *A merge is a transformation*

$$M : (\sigma_1, \sigma_2) \mapsto \sigma_m$$

such that:

$$\sigma_1 \prec \sigma_m \quad \land \quad \sigma_2 \prec \sigma_m.$$

Definition 39 (Merge Validity). *A merge is valid iff:*

$$E(\sigma_m) \le \max(E(\sigma_1), E(\sigma_2)) + \varepsilon_M.$$

C.7 History Preservation

Theorem 13 (No History Erasure). *There exists no admissible transformation T such that:*

$$P_{T(\sigma)} \subsetneq P_\sigma.$$

Proof. Such a transformation would violate monotonicity and acyclicity. □

C.8 Non-Linear Time

Definition 40 (Semantic Time). *Semantic time is defined by the partial order $(V_\sigma, \prec)$, not by linear indices.*

Proposition 13 (No Total Order). *There exists no total order compatible with $\prec$ for all provenance graphs.*

Proof. Forks induce incomparable nodes. □

C.9 Provenance Isomorphism

Definition 41 (Provenance Isomorphism). *Two provenance graphs P_σ, P_τ are isomorphic iff there exists a bijection preserving edge direction and transformation types.*

Theorem 14 (Replay Equivalence). *Isomorphic provenance graphs reconstruct identical semantic states.*

C.10 Consequences

[No Silent Rewrite] Every semantic change is explicitly represented in the graph.

[Accountable Evolution] All semantic states admit finite causal explanation.

D Semantic Neighborhood Topology

D.1 Latent Semantic Space

Let (X, d) be a complete, separable metric space (Polish space), called the *semantic latent space.*

Let

$$\phi : \Sigma \to X$$

be an embedding mapping semantic objects to points in X.

D.2 Embedding Invariants

Definition 42 (Embedding Consistency). *An embedding ϕ is consistent iff:*

$$\sigma_1 = \sigma_2 \;\Rightarrow\; \phi(\sigma_1) = \phi(\sigma_2).$$

Definition 43 (Provenance Sensitivity). *ϕ is provenance-sensitive iff:*

$$P_{\sigma_1} \neq P_{\sigma_2} \;\Rightarrow\; d(\phi(\sigma_1), \phi(\sigma_2)) > 0.$$

D.3 Neighborhood Definition

Definition 44 (Semantic Neighborhood). *Given $\sigma \in \Sigma$ and radius $\epsilon > 0$, define:*

$$\mathcal{N}_\epsilon(\sigma) = \{\tau \in \Sigma \mid d(\phi(\sigma), \phi(\tau)) \leq \epsilon\}.$$

Definition 45 (Local Density). *The local density at σ is:*

$$\rho(\sigma, \epsilon) = |\mathcal{N}_\epsilon(\sigma)|.$$

D.4 Locality Constraint

Definition 46 (Local Interaction Constraint). *All admissible transformations T satisfy:*

$$d(\phi(\sigma), \phi(T(\sigma))) \leq \delta_T$$

for some transformation-specific constant δ_T.

Theorem 15 (Locality of Semantic Evolution). *Semantic evolution is locally bounded in (X, d).*

Proof. Transformations violating locality would require unbounded semantic reinterpretation, which exceeds admissible entropy budgets. □

D.5 Navigation Operator

Definition 47 (Navigation Operator). *Define navigation as a partial function:*

$$\mathcal{N} : \Sigma \times \mathbb{R}^+ \to \mathcal{P}(\Sigma)$$

given by neighborhood queries.

Definition 48 (Traversal Path). *A traversal path is a sequence $(\sigma_0, \sigma_1, \ldots, \sigma_n)$ such that:*

$$\sigma_{k+1} \in \mathcal{N}_\epsilon(\sigma_k).$$

D.6 Spatial Memory

Definition 49 (Spatial Reaccessibility). *A semantic object σ is spatially reaccessible iff:*

$$\forall t_1, t_2, \quad \phi_{t_1}(\sigma) = \phi_{t_2}(\sigma).$$

Theorem 16 (Location Persistence). *Semantic objects occupy stable locations in X across time.*

Proof. Identity invariance and entropy-bounded transformations prevent embedding drift. □

D.7 Feed Impossibility

Definition 50 (Global Feed Ordering). *A feed is a total order:*

$$\prec_F \subset \Sigma \times \Sigma$$

intended to represent relevance.

Theorem 17 (No Global Feed Theorem). *There exists no total order $\prec_F$ compatible with all neighborhood relations.*

Proof. Metric locality induces incompatible partial orders across disjoint neighborhoods. □

D.8 Attention Conservation

Definition 51 (Attention Budget). *Let A_u denote a finite attention budget for user u.*

Proposition 14 (Local Attention Allocation). *Attention expenditure satisfies:*

$$\sum_{\sigma \in \mathcal{N}_\epsilon(\sigma_u)} a(\sigma) \leq A_u.$$

This prohibits unbounded attention capture.

D.9 Dimensional Extension

Definition 52 (Higher-Dimensional Navigation). *Let $X = \prod_{k=1}^{n} X_k$ be a product space. Navigation may occur along projections $\pi_k : X \to X_k$.*

Theorem 18 (4× Interface Compatibility). *All navigation operators commute with dimensional projections:*

$$\pi_k(\mathcal{N}_\epsilon(\sigma)) = \mathcal{N}_{\epsilon_k}(\pi_k(\sigma)).$$

D.10 Consequences

[No Infinite Scroll] There exists no path (σ_n) with strictly increasing novelty and bounded locality.

[Spatial Sociality] Social interaction is equivalent to co-navigation in semantic space.

E Trust Accumulation Model

E.1 Preliminaries

Let Σ be the semantic state space and let $\mathcal{U}$ denote the set of identities (users or agents). Each semantic object $\sigma \in \Sigma$ is associated with an identity $i(\sigma) \in \mathcal{U}$.

Let $t \in \mathbb{R}_{\geq 0}$ denote semantic time as induced by provenance order.

E.2 Persistence Measure

Definition 53 (Accessibility Indicator). *Define the accessibility indicator*

$$\chi_\sigma(t) = \begin{cases} 1 & \text{if } \sigma \text{ is accessible at time } t \\ 0 & \text{otherwise.} \end{cases}$$

Definition 54 (Persistence Functional). *Define persistence of a semantic object as*

$$\pi(\sigma) := \int_0^\infty \chi_\sigma(t)\, dt.$$

Persistence measures duration of availability rather than visibility.

E.3 Entropy Stability

Definition 55 (Entropy Trajectory). *For a semantic object σ, define its entropy trajectory*

$$E_\sigma(t) := E(\sigma_t),$$

where σ_t denotes the semantic state at time t.

Definition 56 (Entropy Stability Functional). *Define entropy stability as*

$$S_E(\sigma) := -\int_0^\infty \left| \frac{d}{dt} E_\sigma(t) \right| dt.$$

Higher values of S_E correspond to lower volatility.

E.4 Reconciliation Credit

Definition 57 (Reconciliation Event). *A reconciliation event is a merge transformation*

$$M : (\sigma_1, \sigma_2) \mapsto \sigma_m$$

that reduces entropy relative to the forked branches.

Definition 58 (Reconciliation Count). *Let $R(\sigma)$ denote the number of reconciliation events in P_σ.*

E.5 Trust Functional

Definition 59 (Trust Functional). *Define trust of a semantic object as*

$$T(\sigma) := F\big(\pi(\sigma), S_E(\sigma), R(\sigma)\big),$$

where $F : \mathbb{R}^3_{\geq 0} \to \mathbb{R}_{\geq 0}$ is monotone in all arguments.

Trust is defined structurally, independent of audience size.

E.6 Identity-Level Trust

Definition 60 (Identity Trust). *Define trust of an identity $u \in \mathcal{U}$ as*

$$T(u) := \sum_{\sigma \in \Sigma : i(\sigma) = u} w(\sigma)\, T(\sigma),$$

where $w(\sigma)$ is a normalization weight.

E.7 Popularity Independence

Definition 61 (Popularity Signals). *Let $L, S, F \in \mathbb{R}_{\geq 0}$ denote likes, shares, and followers respectively.*

Theorem 19 (Popularity Irrelevance).

$$\frac{\partial T(\sigma)}{\partial L} = \frac{\partial T(\sigma)}{\partial S} = \frac{\partial T(\sigma)}{\partial F} = 0.$$

Proof. Popularity variables do not appear in the definition of T. □

E.8 Monotonicity Properties

Theorem 20 (Trust Monotonicity Under Valid Evolution). *If $\sigma \prec \sigma'$ and*

$$E(\sigma') \leq E(\sigma),$$

then

$$T(\sigma') \geq T(\sigma).$$

Proof. Persistence, entropy stability, and reconciliation count are non-decreasing under valid transformations. □

—

E.9 Adversarial Resistance

Definition 62 (Trust Inflation Attempt). *A trust inflation attempt is a transformation sequence increasing T without increasing π, S_E, or R.*

Theorem 21 (No Trust Inflation). *Trust inflation attempts are impossible.*

Proof. All arguments of F correspond to structurally constrained quantities. None can be artificially increased without satisfying invariants. □

E.10 Temporal Convergence

Theorem 22 (Trust Convergence). *For any semantic object σ with finite entropy budget,*

$$\lim_{t \to \infty} T(\sigma_t) \text{ exists.}$$

Proof. Each component of T is monotone and bounded. □

E.11 Consequences

[No Follower Economy] There exists no mechanism by which follower accumulation increases trust.

[Time-Productive Interaction] Time spent producing coherent semantic evolution strictly dominates time spent monitoring visibility metrics.

F Spherepop Calculus Interface

F.1 Syntactic Categories

Let the Spherepop language be defined by the following syntactic categories:

$$\begin{aligned} \sigma &\in \Sigma && \text{semantic objects} \\ m &\in \mathcal{M} && \text{modalities} \\ p &\in \mathcal{P} && \text{programs} \\ T &\in \mathcal{T} && \text{primitive transformations} \end{aligned}$$

Programs are defined inductively as:

$$p ::= \text{id} \mid T \mid p_1; p_2 \mid p_1 \oplus p_2$$

F.2 Type System

Definition 63 (Typing Context). *A typing context is a tuple*

$$\Gamma = (\mathcal{M}_{in}, \mathcal{M}_{out}, \varepsilon)$$

representing required input modalities, produced output modalities, and entropy budget.

Definition 64 (Typing Judgment). *Typing judgments have the form*

$$\Gamma \vdash p : \Sigma \to \Sigma.$$

F.3 Primitive Typing Rules

$$\frac{}{\Gamma \vdash \text{id} : \Sigma \to \Sigma} \quad \text{(T-ID)}$$

$$\frac{T \in \mathcal{T} \quad \mathcal{M}_{\text{in}}(T) \subseteq \mathcal{M}_\sigma \quad \varepsilon_T \leq \varepsilon}{\Gamma \vdash T : \Sigma \to \Sigma} \quad \text{(T-PRIM)}$$

F.4 Composition Rules

$$\frac{\Gamma_1 \vdash p_1 : \Sigma \to \Sigma \quad \Gamma_2 \vdash p_2 : \Sigma \to \Sigma}{\Gamma_1 \cup \Gamma_2 \vdash p_1; p_2 : \Sigma \to \Sigma} \quad \text{(T-SEQ)}$$

$$\frac{\Gamma_1 \vdash p_1 : \Sigma \to \Sigma \quad \Gamma_2 \vdash p_2 : \Sigma \to \Sigma}{\max(\Gamma_1, \Gamma_2) \vdash p_1 \oplus p_2 : \Sigma \to \Sigma} \quad \text{(T-ALT)}$$

F.5 Operational Semantics

Define a small-step operational semantics:

$$\langle p, \sigma \rangle \to \sigma'$$

$$\langle \text{id}, \sigma \rangle \to \sigma$$

$$\langle T, \sigma \rangle \to T(\sigma) \quad \text{if } E(T(\sigma)) \leq E(\sigma) + \varepsilon_T$$

$$\langle p_1; p_2, \sigma \rangle \to \langle p_2, \sigma' \rangle \quad \text{if } \langle p_1, \sigma \rangle \to \sigma'$$

$$\langle p_1 \oplus p_2, \sigma \rangle \to \sigma' \quad \text{if either branch yields valid output}$$

F.6 Entropy Preservation

Lemma 8 (Stepwise Entropy Bound). *If*

$$\langle p, \sigma \rangle \to \sigma'$$

and

$$\Gamma \vdash p : \Sigma \to \Sigma,$$

then

$$E(\sigma') \leq E(\sigma) + \varepsilon.$$

F.7 Type Preservation

Theorem 23 (Subject Reduction). *If*

$$\Gamma \vdash p : \Sigma \to \Sigma \quad \textit{and} \quad \langle p, \sigma \rangle \to \sigma',$$

then $\sigma' \in \Sigma$.

Proof. By induction on the structure of p using primitive and composition rules. □

F.8 Progress

Theorem 24 (Progress). *If* $\Gamma \vdash p : \Sigma \to \Sigma$ *and* $\sigma \in \Sigma$*, then either*

$$p = id \quad \textit{or} \quad \exists \sigma'.\ \langle p, \sigma \rangle \to \sigma'.$$

F.9 Confluence Properties

Definition 65 (Confluent Program). *A program p is confluent iff all valid evaluation paths produce extensionally equal semantic objects.*

Theorem 25 (Local Confluence Under Entropy Bounds). *If all primitive transformations in p commute and respect entropy bounds, then p is confluent.*

F.10 Consequences

[No Hidden Side Effects] Spherepop programs cannot modify semantic state outside explicit transformations.

[Programmable Social Interaction] All social interaction reduces to well-typed program execution over Σ.

G Deterministic Substrate (Spherepop OS)

G.1 Event Log

Let $\mathcal{E}$ be a countable set of events.

Definition 66 (Event). *An event is a tuple*

$$e = (t, u, p, \sigma_{in}, \sigma_{out})$$

where:

$$\begin{aligned} t &\in \mathbb{R}_{\geq 0} \quad \textit{(timestamp)} \\ u &\in \mathcal{U} \quad \textit{(identity)} \\ p &\in \mathcal{P} \quad \textit{(Spherepop program)} \\ \sigma_{in}, \sigma_{out} &\in \Sigma. \end{aligned}$$

Definition 67 (Event Log). *An event log is a sequence*

$$\mathcal{L} = \langle e_1, e_2, \dots \rangle$$

totally ordered by timestamp.

G.2 Kernel State

Definition 68 (Kernel State). *A kernel state is a tuple*

$$K = (O, U, R, M)$$

where:

$$O \subset \Sigma \quad \textit{(active semantic objects)}$$
$$U \subset \mathcal{U} \quad \textit{(identities)}$$
$$R \subset O \times O \quad \textit{(references)}$$
$$M \subset \mathcal{M} \quad \textit{(active modalities).}$$

G.3 State Transition Function

Definition 69 (Transition Function). *Define the deterministic transition function*

$$\delta : K \times \mathcal{E} \to K$$

such that:

$$\delta(K, e) = K'$$

where K' is obtained by validating and applying $p(e)$ to σ_{in}.

Definition 70 (Validity Condition). *An event e is valid iff:*

$$\Gamma \vdash p(e) : \Sigma \to \Sigma \quad \textit{and} \quad \sigma_{out} = p(e)(\sigma_{in}).$$

G.4 Replay Semantics

Definition 71 (Replay Operator). *Define replay as:*

$$\textit{Replay}(\mathcal{L}_{\leq n}) = \delta(\ldots \delta(\delta(K_0, e_1), e_2), \ldots, e_n)$$

for fixed initial state K_0.

Theorem 26 (Replay Determinism). *For any prefix $\mathcal{L}_{\leq n}$, the resulting kernel state is unique.*

Proof. δ is deterministic and events are totally ordered. □

G.5 Late-Joiner Equivalence

Definition 72 (Late Joiner). *A late joiner is an observer reconstructing state from $\mathcal{L}_{\leq n}$.*

Theorem 27 (Late-Joiner Equivalence). *All observers replaying $\mathcal{L}_{\leq n}$ reconstruct identical kernel states.*

Proof. Immediate from replay determinism. □

G.6 Fork Semantics

Definition 73 (Explicit Fork). *A fork occurs only if two events reference the same σ_{in} with incompatible outputs.*

Theorem 28 (No Implicit Forking). *Forks cannot arise without explicit event divergence in $\mathcal{L}$.*

Proof. Sequential replay enforces a single outcome unless separate events encode divergence. □

G.7 Causality and Order Independence

Lemma 9 (Local Commutativity). *Two events e_i, e_j commute iff:*

$$\sigma_{out}(e_i) \cap \sigma_{out}(e_j) = \varnothing.$$

Theorem 29 (Causal Consistency). *Reordering commuting events yields equivalent kernel states.*

G.8 Impossibility Results

Theorem 30 (No State Rewrite). *There exists no event sequence that rewrites prior kernel state without leaving a trace in $\mathcal{L}$.*

Theorem 31 (No Hidden Authority). *No identity can alter kernel state without producing a valid event.*

G.9 Consequences

[Auditability] All semantic state changes admit finite audit trails.

[Deterministic Social Memory] Social state is fully reconstructible from the event log.

H Computational Complexity

H.1 Parameters

Let:

$$\begin{aligned} n &= |\Sigma| \quad \text{(number of semantic objects)} \\ k &= \max_{\sigma} |P_\sigma| \quad \text{(max provenance size)} \\ \rho &= \max_{\sigma,\epsilon} \rho(\sigma, \epsilon) \quad \text{(max local density)} \\ m &= |\mathcal{L}| \quad \text{(event log length)} \\ u &= |\mathcal{U}| \quad \text{(identities).} \end{aligned}$$

H.2 Transformation Validation

Theorem 32 (Transformation Validation Complexity). *Validating a semantic transformation has time complexity*

$$O(k)$$

and space complexity

$$O(1).$$

Proof. Validation requires checking modality completeness and provenance monotonicity, each linear in provenance size. □

H.3 Entropy Evaluation

Theorem 33 (Entropy Evaluation Complexity). *Computing $E(\sigma)$ has complexity*

$$O(|C_\sigma|).$$

[Incremental Entropy Update] Under bounded transformations,

$$\Delta E = O(1).$$

H.4 Neighborhood Queries

Definition 74 (Neighborhood Query). *A neighborhood query returns*

$$\mathcal{N}_\epsilon(\sigma).$$

Theorem 34 (Neighborhood Query Complexity). *Using approximate nearest neighbor indexing, neighborhood queries execute in*

$$O(\log n + \rho).$$

H.5 Navigation Paths

Definition 75 (Traversal Length). *Let ℓ be the length of a navigation path.*

Proposition 15 (Traversal Cost). *Navigation cost is*

$$O(\ell(\log n + \rho)).$$

—

H.6 Replay and Reconstruction

Theorem 35 (Replay Complexity). *Replaying an event log prefix of length m has time complexity*

$$O(mk).$$

[Checkpointed Replay] With periodic snapshots, replay complexity reduces to

$$O((m - m_0)k)$$

for nearest checkpoint m_0.

H.7 Feed-Based Ranking Comparison

Definition 76 (Feed Ranking). *A feed ranking step computes a total order over candidate items.*

Theorem 36 (Feed Ranking Complexity). *Feed ranking requires*

$$\Omega(n \log n)$$

per recomputation.

Theorem 37 (Global Recompute Impossibility). *Continuous feed ranking yields superlinear system-wide cost as $n \to \infty$.*

H.8 Locality Scaling Law

Theorem 38 (Locality Scaling). *Total system load scales as*

$$O(u \cdot \rho)$$

independent of n.

Proof. Users interact only within bounded neighborhoods. □

H.9 Memory Complexity

Theorem 39 (Structural Sharing Bound). *Total storage cost is*

$$O(n + mk)$$

with persistent structure sharing.

H.10 Consequences

[No Global Bottlenecks] No operation requires full-system traversal.
[Asymptotic Stability] System performance remains stable under unbounded growth.

I Structural Impossibility Results

I.1 Preliminaries

Assume the semantic system (Σ, P, E, ϕ) defined in Appendices A–H, with:

$$\Sigma \text{ semantic objects}, \quad P \text{ provenance DAGs}, \quad E \text{ entropy functional}, \quad \phi : \Sigma \to X.$$

I.2 Impossibility of Global Feeds

Definition 77 (Global Feed). *A global feed is a total order*

$$\prec_F \subset \Sigma \times \Sigma$$

intended to represent relevance across all semantic objects.

Theorem 40 (Feed Impossibility). *There exists no* $\prec_F$ *such that:*

$$\sigma_1 \prec_F \sigma_2 \Rightarrow d(\phi(\sigma_1), \phi(\sigma_2)) \leq \epsilon \quad \forall \sigma_1, \sigma_2.$$

Proof. Total orders require comparability across arbitrarily distant neighborhoods, violating locality constraints in X. □

I.3 Impossibility of Viral Amplification

Definition 78 (Viral Amplification). *Viral amplification is the process of increasing visibility without increasing semantic structure.*

Theorem 41 (No Viral Amplification). *There exists no admissible transformation sequence* (T_k) *such that:*

$$E(T_n(\sigma)) \leq E(\sigma) \quad \textit{and} \quad \textit{visibility}(T_n(\sigma)) \to \infty.$$

Proof. Visibility increase without structural information violates entropy bounds. □

I.4 Impossibility of Reputation Laundering

Definition 79 (Reputation Laundering). *Reputation laundering is the increase of trust* $T(u)$ *without corresponding semantic persistence.*

Theorem 42 (No Reputation Laundering). *There exists no sequence of admissible transformations increasing* $T(u)$ *while keeping* $\pi(\sigma)$ *bounded.*

Proof. Trust depends monotonically on persistence and entropy stability. □

I.5 Impossibility of Hidden Moderation

Definition 80 (Hidden Moderation). *Hidden moderation is the alteration or suppression of semantic objects without explicit provenance.*

Theorem 43 (No Hidden Moderation). *Hidden moderation is impossible.*

Proof. All state transitions require explicit events in the log. □

I.6 Impossibility of Shadow Banning

Definition 81 (Shadow Banning). *Shadow banning is selective visibility reduction without semantic transformation.*

Theorem 44 (No Shadow Banning). *There exists no admissible operator that selectively reduces accessibility without altering provenance.*

Proof. Accessibility is a function of spatial position, not ranking. □

I.7 Impossibility of Engagement Optimization

Definition 82 (Engagement Metric). *An engagement metric is a function*

$$g : \Sigma \to \mathbb{R}$$

optimized independently of semantic structure.

Theorem 45 (No Engagement Objective). *There exists no engagement objective compatible with entropy-bounded evolution.*

Proof. Engagement maximization incentivizes entropy increase. □

I.8 Impossibility of Attention Extraction

Definition 83 (Attention Extraction). *Attention extraction is the unbounded consumption of user attention without semantic gain.*

Theorem 46 (Attention Conservation). *For any user u,*

$$\int_0^\infty A_u(t)\,dt < \infty.$$

I.9 Impossibility of Meaning Collapse

Definition 84 (Meaning Collapse). *Meaning collapse is the convergence of semantic objects toward indistinguishable content.*

Theorem 47 (No Meaning Collapse). *Entropy-bounded evolution prevents semantic homogenization.*

I.10 Separation Theorem

Theorem 48 (Architectural Separation). *The coherence–first semantic architecture is not a refinement of feed-based platforms, but is structurally disjoint from them.*

Proof. Feed-based systems require global ranking and engagement optimization, both of which are impossible under the constraints established above. □

I.11 Final Corollaries

[No Ad-Based Incentive Compatibility] Advertising optimization is incompatible with semantic coherence.

[Time Recovery] User time expenditure produces persistent semantic value.

References

[1] C. E. Shannon. A Mathematical Theory of Communication. *Bell System Technical Journal*, 27(3):379–423, 1948.

[2] A. N. Kolmogorov. Three Approaches to the Quantitative Definition of Information. *Problems of Information Transmission*, 1(1):1–7, 1965.

[3] M. Csikszentmihalyi. *Flow: The Psychology of Optimal Experience.* Harper & Row, 1990.

[4] d. boyd. Social Network Sites as Networked Publics. In Z. Papacharissi (ed.), *A Networked Self.* Routledge, 2010.

[5] C. Lampe, R. Wash, A. Velasquez, and E. Ozkaya. Motivations to Participate in Online Communities. *CHI*, 2014.

[6] J. Grudin. Groupware and Social Dynamics. *Communications of the ACM*, 53(5):66–74, 2010.

[7] F. Viégas, M. Wattenberg. Visualization of Online Social Interactions. *CHI*, 2007.

[8] K. Thompson. Reflections on Trusting Trust. *Communications of the ACM*, 27(8):761–763, 1984.

[9] L. Torvalds, J. Hamano. Git: Fast Version Control System. 2005. `https://git-scm.com`

[10] E. S. Raymond. *The Cathedral and the Bazaar.* O'Reilly Media, 1999.

[11] T. Berners-Lee, J. Hendler, O. Lassila. The Semantic Web. *Scientific American*, 284(5):28–37, 2001.

[12] H. Halpin, P. J. Hayes. When Ontologies Collide. *Journal of Web Semantics*, 8(2–3):113–128, 2010.

[13] N. Luhmann. *Social Systems.* Stanford University Press, 1995.

[14] B. Latour. *Reassembling the Social.* Oxford University Press, 2005.

[15] E. Ostrom. *Governing the Commons.* Cambridge University Press, 1990.

[16] L. Floridi. *The Philosophy of Information.* Oxford University Press, 2011.

[17] G. C. Bowker, S. L. Star. *Sorting Things Out: Classification and Its Consequences.* MIT Press, 2000.

[18] L. Manovich. *The Language of New Media.* MIT Press, 2001.

[19] J. Kleinberg. Authoritative Sources in a Hyperlinked Environment. *Journal of the ACM*, 46(5):604–632, 1999.

[20] L. Page, S. Brin. The Anatomy of a Large-Scale Hypertextual Web Search Engine. *WWW*, 1998.

[21] E. Pariser. *The Filter Bubble.* Penguin Press, 2011.

[22] M. Zuckerberg. Building Global Community. Facebook Note, 2017.

[23] M. Foucault. *Discipline and Punish.* Pantheon Books, 1977.

[24] N. Wiener. *Cybernetics.* MIT Press, 1948.

[25] W. R. Ashby. *An Introduction to Cybernetics.* Chapman & Hall, 1956.

[26] K. Friston. The Free-Energy Principle. *Nature Reviews Neuroscience*, 11:127–138, 2010.

[27] Y. Benkler. *The Wealth of Networks.* Yale University Press, 2006.

[28] C. Doctorow. The Enshittification of TikTok. *Pluralistic*, 2023. .