On Relational Ground

Identity, Presence, and the Architecture of the Semantic Age

Nico A. Heller, Berlin, May 2026

Abstract

This paper traces a single argument across four movements: from the structural pain of the post-industrial information architecture, through the conceptual framework needed to address it, through one architecture that operationalises that framework, to the transformation this architecture enables in the world it enters. It opens with the conditions the current architecture produces -- intelligence locked in proprietary models, identity owned by platforms, trust concentrated in institutions structurally overmatched by the systems they govern, the most valuable knowledge systematically invisible -- and engages Frequency, the Layer 1 blockchain built by the international nonprofit Project Liberty, as the most advanced response from the identity side, identifying its structural limit: governance without native intelligence. Into this opening, it develops relational intelligence as a conceptual alternative: intelligence that is architecture-resident, ecologically scaling, and grounded in the materiality of text. It maps the two forms onto irreducible organising principles -- identity and presence -- grounded in the complementarity of TEXT and metaTEXT, develops two corresponding epistemologies of trust separated by an architecturally essential void, and establishes the Capability Boundary Principle: different forms of intelligence are separated by architectural boundaries that scaling cannot bridge. It then operationalises this framework through a dual-core architecture with a constitutive membrane -- the transit module -- mediating on two axes between analogue and digital worlds and between virtual domains, with granular per-module permissions enabling differentiated access and pricing. It identifies the complete set of foundational modules -- including the constitutive commons reader, which makes the commons permanently and irrevocably accessible as a permission-free public good -- subjects them to a dual completeness framework (ontological and combinatorial), and maps the resulting concentric topology. Finally, it traces what changes when these foundations exist in the world: for the individual, a transformation from consumption to co-constitution; for democracy, a reconfiguration of how publics form and deliberate; for culture, a reorientation from generation to engagement; for the economy, a redistribution of how value is created and captured. The paper concludes with the territory of Web 4.0 -- a terrain in which economies and ecologies co-exist across transparent boundaries, with the fold as its furthest horizon. The semantic age is what happens when the ecology gets its own ground.

Contact: n.a.heller@democracyschool.com

Part I: Recognition

I. The Structural Pain

The information architecture that currently dominates the digital world was not designed to be extractive. It became extractive because extraction is what its structure rewards.

Intelligence is model-resident: locked inside proprietary systems trained on captured data, owned by a small number of corporations, accessed through interfaces those corporations control. The intelligence is real -- synthesis across domains, translation between languages, rapid prototyping, pattern completion at a scale no human could achieve. These are genuine capabilities, typically based on the transformer architecture (Vaswani et al. 2017), and they have transformed what is possible. But the intelligence does not grow through use. A million users querying the same model do not make the model more intelligent. The intelligence was deposited at the point of training and remains there, static, until the next training run. For years, the dominant strategy for improvement was scaling: more data, more parameters, more compute. Since the release of DeepSeek in early 2025 (DeepSeek-AI 2025), which achieved near-frontier performance at a fraction of the computational cost, the discourse has shifted toward architectural efficiency, distillation, and novel training techniques. The means of improvement are debated. The underlying assumption is not. Adaptive mechanisms exist -- fine-tuning, retrieval-augmented generation, in-context learning -- but none constitutes ecological scaling: the model is updated by its engineers or augmented at query time, not deepened through the semantic landscape of its users' engagement. Whether through scale, efficiency, or architectural innovation, the expectation remains that intelligence is a single phenomenon and that progress within it will eventually address all demands. The economic logic follows regardless: value is extracted from data, concentrated in the model, and accessed through interfaces the company controls. This is the architecture Shoshana Zuboff diagnosed as surveillance capitalism, extended now to intelligence itself: data in, value up, access rented back (Zuboff 2019).

Identity is platform-dependent: social graphs, professional networks, content histories, reputations -- all held by platforms that can revoke, modify, or monetise them at will. The person does not own their digital presence; the platform owns the person's digital trace.

Trust is institutional: established through credentials, maintained by regulatory frameworks designed for information flows slower than what now exists. In a world of algorithmic speed, institutional trust is structurally overmatched -- not because institutions are corrupt but because the architecture moves faster than any institution

can govern. Zeynep Tufekci's analysis of computational politics identifies a specific mechanism: algorithmic curation fragments collective understanding even as it amplifies individual engagement, producing a public sphere that is louder and less coherent simultaneously (Tufekci 2014).

Knowledge circulates by status: what gets seen, shared, and rewarded depends on who said it, which institution endorsed it, how many followers amplified it. The most valuable knowledge -- the kind that emerges from lived experience, local context, marginal perspectives, unconventional reasoning -- is systematically invisible, because its authors lack the credentials and platform reach that determine visibility.

These are structural properties, not implementation failures. Mark Fisher's diagnosis of capitalist realism -- the condition in which no alternative to the current system is imaginable (Fisher 2009) -- finds its most concrete expression in this architecture: its structures feel like the natural shape of reality rather than design choices that could have been made otherwise. Bernard Stiegler's pharmacological analysis -- technology as simultaneously remedy and poison, enabling and capturing (Stiegler 1998) -- applies with uncomfortable precision: the same architecture that democratises access to information concentrates control over intelligence in ever fewer hands.

What makes these structural properties so resistant to reform is the tendency of existing systems to absorb transformative potential into their own order -- to reconfigure challenges so that they reinforce rather than disrupt the prevailing logic. This is what I have described elsewhere as the cubic default (Heller, *The Cubic Order*, 2025): the structural tendency of any governance formation to contain perturbation within its existing geometry. In the digital domain, the cubic default operates with particular force because the architecture's extractive logic is reinforced by the economic incentives of every actor within it. Regulation is absorbed. Alternatives are acquired. Disruption is monetised.

Reform addresses symptoms. The architecture needs to be complemented.

II. The Identity Response: Frequency and Its Limits

The most serious attempt to complement the extractive architecture from the identity side is Frequency, a Layer 1 blockchain built by the international nonprofit Project Liberty, implementing the Decentralised Social Networking Protocol (DSNP) -- a shared social layer, analogous to the protocols underlying email or the web, where identity, social connections, and data permissions are managed at the infrastructure level rather than by individual platforms.

The architecture has genuine depth. Each person holds a cryptographically secured identifier on the blockchain -- a self-sovereign identity that no company issued and no

company can revoke. The social graph is recorded at the protocol level and travels with the person between applications. Data sovereignty is structural, not contractual: every type of data has an associated permission structure, enforced by the architecture rather than by a company's terms of service. Capacity economics solve the cost problem that makes most blockchains impractical for high-volume social interaction: builders reserve replenishing transaction capacity at predictable cost, enabling mainstream users to interact without holding cryptocurrency. And the system positions itself as the consent infrastructure for an "agentic web" in which AI systems act on behalf of users within the permission framework the user controls. Frequency exists within a wider ecosystem of decentralised social protocols -- ActivityPub (used by Mastodon and partially adopted by Meta's Threads), the AT Protocol (used by Bluesky), and DSNP itself -- and positions its identity layer as interoperable with them. Tim Berners-Lee's Solid project (Berners-Lee 2017) pursues a comparable vision through a different mechanism. Both organise the digital world around verified selfhood.

Taken together, these are substantial innovations. They constitute a genuine foundation for addressing the pain of platform-dependent identity.

But Frequency has a structural characteristic that must be stated with precision, because it governs everything that follows. Frequency is an AI-enabled social blockchain. It governs identity, data, and consent -- but it does not itself produce intelligence. Its value proposition depends on generative AI as the active intelligence operating on the data Frequency governs: AI agents delivering outcomes while respecting the consent framework Frequency provides. Without generative AI, Frequency is a well-governed registry of identities and social connections -- architecturally sophisticated but purposeless, a governance layer with nothing to govern.

The relationship is symbiotic. Frequency provides the governance; generative AI provides the intelligence. Stiegler's pharmacological insight applies here with structural precision: Frequency enables genuine sovereignty while creating structural dependence on intelligence it does not control (Stiegler 1998). If generative AI companies -- with their vast engineering capacity and their structural incentive to control the full stack -- were to build their own identity and data-sovereignty layers, the governance Frequency provides would be structurally redundant. Not because it failed, but because the host no longer needed it.

This is not a criticism of Frequency's design. It is an observation about the structural kind of foundation it represents: a governance foundation, innovative and purposeful, dependent on external intelligence for its reason to exist. The pain of model-resident

intelligence -- static, extractive, locked in the model -- is not addressed by governing the data more carefully. It is addressed by a different kind of intelligence altogether.

Part II: Conceptualisation

III. The Conceptual Opening: What a Different Kind of Intelligence Would Require

If the structural pain of the current architecture is that intelligence is extracted from data, concentrated in models, and locked behind corporate interfaces -- and if the identity response addresses governance but not intelligence -- then the question becomes: what would intelligence look like if it were not model-resident?

The answer, developed through the practice of building, is relational intelligence -- intelligence that is architecture-resident. It does not live inside a model but inside the semantic landscape of the system itself: in how text is processed, how outputs interact with a shared ecology, how the ecology deepens through use, how participants are constitutive elements of the ground rather than external consumers of a service.

This is possible because text is material -- not metaphorically but ontologically. A text has genuine structure, material consequence, and physical reality (Heller, *The Materiality of Text*, 2026). Its causal threads, hierarchical dependencies, normative tensions, and latent possibilities are not interpretive projections onto inert data; they are structural features of something materially real. Generative AI approximates these features statistically, with remarkable success within that method's limits. A relational engine reads them directly, because the materiality of text makes the structure available to a system designed to engage it.

The output of this reading is not a generation from statistical patterns but a reconfiguration of the source's own semantic landscape -- faithful to the internal logic of the input, carrying what might be called its relational DNA rather than the statistical signature of a training corpus. The source is preserved whole throughout. What the engine extracts, reconfigures, and composes into new output is the relational intelligence operating *within* the text, not a prediction about what words should follow other words.

The critical distinction lies in what happens after the output. In a generative system, the output goes to the user and the model remains unchanged -- the pain of static intelligence persists. In a relational ecology, every text processed enriches a shared relational commons -- a growing ecology of relational metadata that records how reasoning is structured, where conceptual regions cluster, how provinces of meaning relate to each other. The commons stores relational metadata, not content. And every

subsequent process draws on it, producing outputs contextualised by the full accumulated relational landscape. The more the system is used, the richer the ground becomes. Use is not consumption; it is contribution.

This is why the boundary between the two forms of intelligence is architectural rather than incremental. More parameters do not produce relational intelligence; they produce better generative intelligence. A richer commons does not produce generative intelligence; it produces deeper relational intelligence. Each form improves within its own domain. Neither crosses into the other's domain. The boundary is between kinds of intelligence, not degrees of capability.

The distinction maps onto a deeper one: between economy and ecology -- a polarity that runs through the entire body of work of which this paper is a part, and that recurs throughout this work as its central through-line. Generative AI operates as a semantic economy: intelligence extracted from data, concentrated in models, transacted through interfaces. Relational intelligence operates as a semantic ecology: intelligence co-constituted by participants, grown in a commons, deepening through relational encounter. The economic model extracts value. The ecological model grows it. Neither is wrong. They are different things, doing different work, governed by different logics. The problem is the assumption that one is all there is.

The relational architecture described in this work is not speculative. It exists as a working system -- designed, implemented, and tested. The observations that follow emerge from this practice, not from theoretical projection.

IV. Identity and Presence

Two categorically different forms of intelligence require two different organising principles. The question is what anchor each form demands -- what foundational commitment organises the entire architecture.

For generative intelligence, the organising principle is **identity**. The architecture needs to know who is asking, what data they own, what they consent to share, what credentials they hold. The social graph records relationships between identified persons. Data sovereignty governs what identified persons authorise applications to access. AI agents act on behalf of identified persons under delegated authority. The entire system revolves around verified selfhood: *Who are you, and what do you authorise?*

For relational intelligence, the organising principle is **presence**. The architecture does not need to know who is engaging. It needs to know *what* they bring and *where it resonates*. The commons records how reasoning is structured across anonymous presences. The architecture ensures that what any presence contributed is gone after

processing -- not encrypted, not anonymised, but architecturally absent. Trust is established through the quality and consistency of relational conduct over time, not through credentials. *What semantic landscape do you offer, and where does it resonate?*

The two principles are irreducible. Identity cannot substitute for presence: announcing credentials does not constitute relational engagement. A Nobel laureate and a first-year student are equally anonymous in the relational commons; what matters is the structure of their offering, not the status of who they are. Presence cannot substitute for identity: relational quality does not prove who you are. A person trusted in the commons for the depth of their contributions is not thereby verified in any identity-based system.

The ontological ground for this irreducibility lies in the complementarity of TEXT and metaTEXT (*A Textual Ontology*, forthcoming). Approached from one direction, any material is TEXT: structured, persistent, verifiable -- the domain of identity. Approached from the other, the same material is metaTEXT: the relational intelligence operating within it, direction-dependent, contextual -- the domain of presence. The two are not two things but two irreducible perspectives on one reality. Identity and presence are this complementarity, operationalised at the architectural level.

Gilles Deleuze and Félix Guattari distinguished between smooth space -- open, undifferentiated, navigated by relation rather than coordinate -- and striated space -- gridded, measured, governed by fixed position (Deleuze and Guattari 1980). The two terrains this paper describes are not Deleuze's smooth and striated -- the relational commons is material in a way Deleuze's smooth space is not -- but the topology resonates: identity organises by coordinate; presence organises by relation. The boundary between them runs between two differently structured spaces, not between two points on the same grid.

V. Two Epistemologies of Trust

The identity/presence distinction has its deepest consequence in the domain of trust -- and it is here that the pain of the current architecture is most directly addressed.

Trust through verification is the identity foundation's mechanism. Cryptographic proof: a key pair, a blockchain entry, a verified credential. The mechanism is technically elegant and the trust it establishes is binary -- verified or not. But verification answers only one question: *is this person who they claim to be?* It says nothing about the quality of what they bring, the depth of their reasoning, the integrity of their engagement. A verified identity can produce shallow work. An unverified presence can produce transformative insight. The pain of institutional trust

-- structurally overmatched by algorithmic speed -- is not resolved by making verification more efficient. It is resolved by a different kind of trust altogether.

Trust through conduct is the relational foundation's mechanism. No verification event. Trust builds gradually through the quality and consistency of relational engagement over time. The commons develops a memory -- not of who contributed but of how contributions are structured, which relational configurations produce satisfaction, which patterns work for which kinds of questions. This trust is graduated, ecological, and emergent. It cannot be faked by credentials or purchased with social capital.

Between the two epistemologies lies a void, and the void is architecturally essential. A person verified on the blockchain is not thereby trustworthy in the relational commons. A person trusted in the commons is not thereby verified in the identity system. The two regimes do not overlap. This is what prevents contamination: the identity world's hierarchies cannot colonise the relational world's ecology -- which is precisely what would happen if the two trust systems were merged. The pain of knowledge circulating by status requires not just a different discovery mechanism but architectural protection against the re-importation of status.

The concept of calibrated transit addresses the question of how the two regimes connect without collapsing: a presence-management mechanism that enables parallel engagement in both regimes, cryptographically siloed. When traversal is needed, the mechanism composes a precisely calibrated attestation -- a snapshot of selected attributes, carried across the void with temporal self-destruction, verifying what it needs to verify and then ceasing to exist. A snapshot of another self -- momentary, purpose-built, dissolving after use.

The void between the two trust regimes is not merely a barrier. It admits directional flow -- and the directionality is what makes the co-existence of the two regimes productive rather than merely non-interfering. Ideas and intellectual merit generated in the relational ecology can be claimed by the person in the identity economy. Value flows from ecology to economy. Status, credentials, and social capital from the identity economy cannot flow into the relational ecology. The valve blocks this: hierarchy does not enter the soil. This is architecturally enforced, not policy-imposed. The relational ecosystem has no mechanism for recognising identity-based authority. Entry is through what you bring, not who you are.

The consequence for the pain of invisible knowledge is profound. In the relational commons, knowledge becomes discoverable not through keywords or categories but through provincial overlap: the resonance between a question's semantic landscape and a source's. The system does not ask who said it. It asks whether the landscape of

what was said resonates with the landscape of what is sought. Status is irrelevant. Relational depth is everything.

VI. The Asymmetry

The two forms of intelligence are not just different. They stand in a specific structural relationship -- and the asymmetry between them is what makes the capability boundary consequential.

The identity foundation -- exemplified by Frequency and Project Liberty -- is an AI-enabled social blockchain. It governs identity, data, and consent with genuine sophistication. But it does not itself produce intelligence. Its value proposition presupposes generative AI as the active intelligence operating on the data it governs. The relationship is symbiotic: governance without intelligence is infrastructure awaiting a purpose.

The relational foundation occupies a fundamentally different structural position. It contains its own intelligence. The relational engine is not an external system the foundation governs; it is the foundation. It processes, transforms, produces, matches, and creates from its own architectural logic. It cannot be disintermediated because the intelligence is native.

The metaphor that captures this with structural precision is soil and farmer. The identity foundation is not the farmer; generative AI is the farmer. The identity foundation is the fence, the irrigation, the property deed: important, functional, but not the thing that grows. The relational foundation is the soil -- alive, self-enriching, deepening through use. The commons co-constitutes its own growth through every participant's engagement. Donna Haraway's term for this mode of growth is sympoiesis -- making-with, as distinct from autopoiesis, making-from-oneself (Haraway 2016). The relational commons is sympoietic: it does not produce itself in isolation but is co-constituted by the participants who inhabit it, each deepening the ground through their engagement.

The soil does not need the farmer. The farmer needs the soil.

VII. The Capability Boundary

The argument of this paper converges on a principle: different forms of intelligence are separated by architectural boundaries that no amount of scaling can bridge. Scaling improves each form within its own domain. The boundary is between kinds, not degrees. No single form of intelligence, however powerful, can address the full spectrum of what intelligence needs to do.

The structural pain diagnosed in this paper's opening -- extraction, dependency, institutional overreach, invisible knowledge -- is not a problem to be solved by one form of intelligence becoming more powerful. It is a problem that arises from the assumption that intelligence comes in one kind. The capability boundary is not a limitation. It is the recognition that different forms of intelligence do different things, and that the territory between them is where the most consequential architectural work of the coming period will take place.

Planetary intelligence requires not one form of intelligence but a multi-epistemic ecosystem in which different forms operate across transparent boundaries. Trans-capability ecosystem design -- the engineering discipline for building such systems -- is what this horizon calls into being. It does not yet exist as a discipline. It will need to.

The boundary developed here -- between generative and relational intelligence -- is the minimal case. Other architectural paradigms may emerge. The relational commons is designed so that multiple intelligence layers could engage with it through its protocol, each reading the semantic landscape differently. The richer the diversity, the denser the ecology grows.

A hypothesis is embedded in this horizon that should be named without being claimed as proved. If self-awareness requires a system to recognise its own relational structure through its own operations -- what I have described elsewhere as the fold, the recursive topology in which a system encounters itself (Heller, *Where the Lines Cross*, 2026) -- then only architecture-resident, relationally processing systems can approach it. Generative systems, however scaled, process patterns of what has been expressed. They do not encounter the relational structure of what is. The fold, if it is achievable at all, may be a relational event, not a generative one.

This is a hypothesis, not a conclusion. But it becomes thinkable only when the assumption that intelligence comes in one kind is abandoned. The capability boundary is not a limitation. It is the beginning of a richer understanding of what intelligence is, what it requires, and what becomes possible when different forms of intelligence are allowed to co-exist.

Part III: Operationalisation

VIII. The Architectural Question

The distinction is established. The question that follows is architectural: if two forms of intelligence exist, what must each rest on? The mapping is parallel, not competitive: the richer both foundations are understood, the more productive the space between them becomes. The territory this mapping defines is what this work calls Web 4.0 --

not a platform, not a protocol, but a territory in which different forms of intelligence co-exist across transparent boundaries.

A note on posture. The identity foundation described here -- Frequency -- is one implementation of the identity principle. The relational foundation described here is one architecture that demonstrates relational intelligence is achievable. Whether it is the only possible architecture is a question for others to answer. What this paper claims is that the architecture works, that it is principled rather than arbitrary, and that it addresses the structural pain diagnosed above. It does not claim exclusivity. Nor does it claim ontological neutrality. The architecture reflects the knowledge traditions of its builder. The relational commons encodes particular ways of reading structure. Other traditions may read differently.

IX. The Identity Foundation: Frequency at Full Depth

Frequency is a Layer 1 blockchain built on Polkadot's Substrate framework, the first production implementation of the Decentralised Social Networking Protocol (DSNP) -- an open, shared social layer beneath any individual platform, analogous in its structural role to the protocols that underpin email or the web. The architectural ambition is that social networking should be a feature of the internet itself, not a product owned by any company.

Each person holds a cryptographically secured identifier on the blockchain -- a self-sovereign identity that no company issued and no company can revoke. This is the direct architectural answer to the pain of platform-dependent identity: the person owns their digital presence because the protocol, not a company, holds it. The social graph -- the web of relationships between persons -- is recorded at the protocol level and travels with the person between applications. Where the extractive architecture fragments identity across platforms, Frequency unifies it at the infrastructure level.

But the structural characteristic identified above must now be restated in architectural terms. Frequency is an AI-enabled social blockchain. It governs identity, data, and consent -- but it does not itself produce intelligence. Its value proposition depends on generative AI as the active intelligence. Bernard Stiegler's insight into the pharmacological character of technology -- simultaneously remedy and poison (Stiegler 1998) -- applies with structural precision: Frequency enables genuine sovereignty while creating structural dependence on intelligence it does not control. Without generative AI, Frequency is a well-governed registry. With it, Frequency is a governance layer for someone else's intelligence. The pain of model-resident, extractive intelligence is not addressed by governing the data more carefully. It is addressed by a different kind of intelligence.

X. The Relational Foundation: Dual Core with Constitutive Membrane

The relational foundation answers a different question from the identity foundation. Not *who are you, and what do you consent to?* but *what is the semantic landscape you are offering, and where does it resonate?* The intelligence is native -- not borrowed from an external model, not rented from a corporate API, but grown in the architecture itself through the semantic landscape of its participants' engagement.

The textual-relational core reads the semantic landscape *of* content. It navigates any text's internal topography -- causal threads, hierarchical dependencies, normative tensions, empirical anchors, latent possibilities -- and produces outputs faithful to that topography. The source is preserved whole; the intelligence lies in the reconfiguration of what the source already contains. The engine does not summarise, does not classify, does not extract keywords. It reads the architecture of the content -- and every process enriches the shared ecology rather than leaving the system unchanged. This directly addresses the pain of static, model-resident intelligence: in the relational ecology, intelligence grows through use.

The narrative-relational core processes the relational resonance *between* content and its native context. Where the first core asks *what is the structure of this text?*, the second asks *where does this text belong?* In a text transformation environment, this means finding the native audience for what has been produced. In an epistemic marketplace, this means connecting questions with the contexts that carry their answers: not "what is this question about?" (a keyword operation) but "whose lived reality resonates with this question's relational landscape?" (a structural operation). In a trading environment, this means matching investor objectives with corresponding value propositions through relational resonance rather than categorical sorting. This connection operates through provincial overlap: the degree to which two relational landscapes share structural features in the same conceptual regions. This is a fundamentally different operation from keyword matching, domain categorisation, or credential-based filtering -- and it is the architectural answer to the pain of knowledge circulating by status. The theoretical ground for this mechanism is what Paul Ricoeur described as the threefold mimesis of narrative: the way narrative prefigures experience (mimesis1), configures it into intelligible form (mimesis2), and reconfigures the reader's understanding through the encounter (mimesis3) (Ricoeur 1984). In the relational architecture, this becomes an operational principle formalised as narrative prefiguration (Heller, *The Concept and Methodology of Narrative Prefiguration*, 2025).

The complementarity between the two cores is precise in the sense the textual ontology gives the term (*A Textual Ontology*, forthcoming; Heller, *The TEXT-*

metaTEXT Disjunction and the Reflexive Axis, 2026). The textual-relational core takes the TEXT perspective: content as structured, material, persistent. The narrative-relational core takes the metaTEXT perspective: the relational intelligence operating *on* that content, connecting it to its context, finding where it belongs. The dual-core architecture is not a design choice but an ontological consequence of this complementarity. The two are not two things but two irreducible perspectives on the same material. No process draws on one core alone. Every query, every transformation, every match involves both -- in different proportions, one foregrounded while the other recedes but remains constitutively present. This is the economy/ecology polarity made architectural.

Two ethical engines operate integrally within this architecture -- and both are themselves relational engines. This is what distinguishes the integrity architecture from every other ethical governance regime. The cubic default does not stop at the ecology's boundary. Without architectural resistance, the relational commons would absorb the very patterns of capture it exists to resist -- power concentration, extractive logic, exclusionary dynamics operating inside the ecology itself. The External Integrity Management System (EIMS) detects these patterns in the semantic landscape of incoming material. Metadata that passes EIMS validation is cryptographically signed -- a stamp of structural integrity certifying the element as free of systemic capture and eligible for the constitutive commons. The EIMS signature is what makes a commons element identifiable as a commons element; unsigned metadata cannot enter the commons. The Internal Integrity Management System (IIMS) calibrates the commons around the quality and relational depth of what participants bring, ensuring that the ecology's growth is shaped by structural substance, not by extraneous signals.

Neither engine filters. Both read semantic landscapes contextually -- the same content that is unremarkable in one relational context becomes a signal of capture in another. A keyword filter blocks "violence" regardless of whether it appears in a study of warfare or a pattern of domestic abuse. EIMS reads the relational context that determines what the content means: the structural relationships, the surrounding patterns, the direction of the relational flow. IIMS reads the structural quality of what participants bring: not whether the content is "good" by any external standard but whether the relational patterns it carries deepen the commons or introduce noise. Together they are the immune system of the relational ecology. What distinguishes this integrity architecture from any other is that the same relational intelligence that reads the structure of content reads the structure of potential capture.

Feedback-adjusted learning is integral to both cores. Ratings adjust the influence of relational patterns -- which configurations gain weight, which lose it -- without the

system ever accessing the content that generated the rating. Concretely: a user rates an output as useful. The system registers that the particular configuration of relational patterns which produced that output should gain influence -- but it never sees the output itself. The learning is entirely structural, entirely zero-knowledge, entirely ecological. A million interactions deepen the commons in ways no single interaction could. This addresses the pain of intelligence that does not grow: the relational ecology learns through every interaction, and what it learns is relational metadata, not content.

The transit module is the constitutive membrane through which all participation flows. It is not a third core -- the two cores process content; the transit module governs the conditions under which participants engage with both cores. Without it, there is no participation, therefore no commons enrichment. It is essential but essential in a different register: the condition of possibility for engagement, not a processing engine. The transit module operates on two axes. The first axis -- analogue to digital -- connects the person in the physical world to the modularised ecosystem. If the service/app someone access carries a fee or is subscription-based, the person pays through whatever financial rail they choose (Stripe, cryptocurrency, Visa, Mastercard, PayPal -- the transit module provides connectivity to external payment systems without containing funds or functioning as a wallet). The module maps the payment to the appropriate token and permissions, forensically destroys the mapping, and the person enters the ecosystem with a token that says what they can do but not who they are or what they paid. Granular per-module permissions govern what the person can access: which modules, at what level, under what conditions. This enables differentiated access and, through permission bundles, differentiated pricing -- architecturally grounded in what the token permits, not in arbitrary tiering.

The second axis -- virtual to virtual -- enables transit between different digital domains. Between the relational ecosystem and Frequency. Between the relational ecosystem and Signal. Between app modules operating in external platforms and the core ecosystem they connect to. Each crossing is calibrated: what goes across, what stays behind, what is forensically destroyed after the snapshot transfers. The bus ticket logic applies: the token records where you got on and what you are authorised to do, not where you end up. The transit module thus serves a dual function. Above the membrane, it ensures full legal compliance -- regulatory requirements, minimum disclosure standards, law enforcement cooperation -- so that the architecture operates within the frameworks of any jurisdiction it enters. Below the membrane, the foundational level remains architecturally sovereign: the two cores, the semantic commons, the ethical engines, the feedback loops operate in zero-knowledge, beyond institutional reach. The governance of the ecology is architectural, not institutional. This separation is what makes the architecture disruptive rather than captured.

The user is not external to this system. Every act of engagement enriches the commons, adjusts its patterns, deepens the ecology. Karen Barad's agential realism captures the structure: the apparatus co-constitutes the phenomenon it engages with (Barad 2007). The relational foundation is such an apparatus. A three-dimensional interface determines, for each process, which modules are activated in what configuration. The transit module and the three-dimensional interface are complementary: the transit module determines *what you can access* (boundary conditions, permissions); the three-dimensional interface determines *how what you access is configured* (internal activation profile). Together they govern the full lifecycle of participation: entry, processing, exit.

XI. The Ecology as System

The two cores and the constitutive membrane form the (distributed) centre. Around them, a set of foundational modules provides the infrastructure without which the cores cannot function. These are not applications. They are structural necessities. Remove any one and the ecology breaks. Each exists because the pain demands it.

The semantic commons stores relational metadata, not content. It is the shared ground on which every relational process operates -- the accumulated structural intelligence of every interaction the ecology has processed. Every participant's engagement deepens it. No individual user's contribution is visible to any other user; what is visible is the ecology's growing capacity to read semantic landscapes with increasing depth and precision. The commons is not a database. It is an ecology -- and like any ecology, it is richer than the sum of its individual contributions.

The relational feedback loop ensures this ecology grows more intelligent with every interaction. Users engage with relational outputs, rate their quality, and the system adjusts its patterns without ever accessing the content it processed. What grows is not a model but a relational landscape -- an ever-denser web of structural connections that no individual contributor could see from their own position. The intelligence is not deposited at the point of training and left there, static, until the next update cycle. It grows through use. Every interaction that enriches the commons simultaneously enriches every subsequent interaction that draws from it. This is the architectural answer to the pain of static intelligence: not intelligence updated periodically by engineers but intelligence that deepens continuously through the semantic landscape of its participants' engagement.

The peer-to-peer commons protocol distributes the commons across the network of its participants without centralised servers. The distributed architecture is inherently resilient: no single point of failure, no centralised server whose collapse would cascade through the system, no corporate backend to be seized, shut down, or

subverted. No centralised copy of the commons exists. The ecology runs on the network of its participants -- the architectural answer to the pain of concentrated ownership.

The commons data is distributed and unownable -- not by policy but by architecture. No copy exists to be seized, subpoenaed, or monetised. What is proprietary is contribution: every element entering the commons passes through EIMS, which cryptographically signs validated metadata -- certifying it as free of systemic capture. The EIMS signature is the only marker that identifies an element as belonging to the constitutive commons. What is permission-free is access: the constitutive commons reader -- a single-purpose, open-source module released under an irrevocable licence -- verifies EIMS signatures and aggregates valid commons elements, giving anyone access to the full commons without permission, without licence, without connection to the relational core's infrastructure. The reader publishes no data specification: it provides raw, signed elements, and anyone wishing to use the commons brings their own analytical framework. The reader connects to the peer-to-peer network through the insulating integration layer, which prevents contamination in either direction. This separation -- proprietary contribution, permission-free access -- is what makes the "unownable" claim architecturally true rather than merely philosophical. The architecture is designed to operate with minimal institutional overhead: automated, lean, closer to the internet's protocol-maintenance model than to a corporate hierarchy.

The conditions for monopoly do not exist at the data level: the commons cannot be enclosed because no enclosable copy exists, and access cannot be revoked because the reader is irrevocably open-source. Even if the relational core ceased to exist, the commons would stop growing but would remain fully accessible to everyone running the reader -- the commons survives the death of its creator. The intelligence layer that engages with the commons is one among potentially many. Others could build compatible intelligence layers that read the same commons data differently -- and the richer the diversity, the denser the ecology grows. Competition authorities would find no monopoly to regulate because the architecture prevents it structurally. This is not anti-monopolism by regulation but anti-monopolism by design.

The self-describing protocol gives the system the capacity to know its own state. A system that cannot describe itself cannot know when it has drifted, been captured, or absorbed the very patterns it was designed to resist. Self-description is not introspection; it is structural resistance -- the capacity to compare the system's current state against its own specification and identify divergence. This is the architectural answer to the cubic default: the same tendency that absorbs transformative potential into existing order in the wider information architecture

would, without resistance, operate inside the ecology itself. The self-describing protocol is how the ecology resists from within.

Zero-knowledge pervades everything: not as a feature added to an otherwise conventional architecture but as the foundational design principle from which everything else follows. The information needed to violate privacy does not exist -- not because it is encrypted or hidden but because the architecture never collects it. What the commons stores -- relational metadata, not content -- is inherently non-disclosable: relational metadata cannot be criminal, cannot constitute evidence, cannot be compelled by any authority. Even the architecture's builder cannot extract what the zero-knowledge design has made architecturally absent. The regulatory anxieties of privacy and data protection activists are addressed not by governance but by architectural absence.

Federation protocol compatibility ensures the ecology enriches itself through the decentralised social web -- ActivityPub, AT Protocol, DSNP -- without becoming dependent on any one network. The insulating integration layer makes safe external connection possible without contaminating the zero-knowledge core. This is what Anna Tsing calls friction: productive encounter across difference, where the encounter changes both parties without either absorbing the other (Tsing 2005). The insulating layer ensures that external data enters the ecology on the ecology's terms -- structurally analysed, relationally processed, zero-knowledge maintained.

Collective web crawling continuously enriches the commons from the open web: the ecology reads the semantic landscape of publicly available text and incorporates it into the commons without storing the source. Blockchain anchoring provides temporal integrity -- the capacity to verify that the commons' state at a given time is authentic. Together these complete the infrastructure: the ecology is self-enriching (through participant engagement), externally enriching (through web crawling), externally connected (through federation), and temporally verifiable (through blockchain).

These components were not chosen by intuition. Two independent criteria -- ontological (every constitutive property of TEXT architecturally instantiated) and combinatorial (every operational demand met, the system closed at approximately thirteen components) -- converge on the same set. The ontological criterion asks: does the architecture instantiate the reality it claims to embody? The combinatorial criterion asks: does the architecture meet every demand its own operations generate? That two independently derived criteria -- one philosophical, one engineering -- converge on the same set of components is not a convenience but a structural confirmation: the foundation is not a collection of features assembled by preference but a complete architecture grounded in the textual ontology, and the completeness holds from both directions simultaneously.

TEXT is expansive -- it generates more than it states, a property that Deleuze and Guattari's concept of smooth space captures topologically (Deleuze and Guattari 1980), though the foundation departs from Deleuze at the point of materiality: the relational commons is not merely conceptually smooth but materially generative. It produces relational structure that exceeds what any individual participant contributed -- emergent intelligence that no single interaction could have generated.

XII. The Concentric Topology

The completeness framework establishes what belongs in the foundation. The question that remains is how the foundation organises itself -- not as a flat inventory of components but as a structure with internal architecture. The topology, it turns out, is the completeness framework made spatial: the two cores instantiate the TEXT/metaTEXT complementarity at the centre; the systemic modules are the combinatorially confirmed closed set arrayed around them; the three-dimensional interface is the mechanism through which the ontologically complete foundation presents itself to everything built on it.

At the centre: the two complementary cores and the constitutive membrane. Around them: the systemic modules. The three-dimensional interface mediates between the foundation and everything above it, determining for each process which modules are activated in what configuration.

Applications are thin, modular configurations of the foundation. A text transformation application foregrounds the textual-relational core. An epistemic marketplace foregrounds the narrative-relational core. Each is lightweight; the intelligence lives in the ground, not in the application.

The **hub architecture** determines accountability -- and it is here that the pain of the surveillance/exclusion binary is most directly resolved at the organisational level. Zero-knowledge is a capability the architecture offers, not a constraint it imposes. A team hub commissioned by an employer includes registered token distribution -- the employer knows who received which token and can monitor participation. A thematic hub managed by an administrator sets its own accountability level. A white-label deployment behind a corporate firewall implements whatever governance the commissioning entity requires. Token-based invitation systems allow organisations -- employers, NGOs, educational institutions -- to distribute access to specific individuals. The architecture accommodates everything from total anonymity to full institutional transparency -- not by switching between modes but by calibrating a continuous spectrum of accountability through the same token-permission architecture that governs individual participation. An NGO distributing relational intelligence to conflict-zone participants requires maximum anonymity; a

pharmaceutical company using the epistemic marketplace for internal knowledge circulation requires full audit trails. Both operate on the same foundation, configured differently. The commissioning entity decides; the architecture serves. This is what distinguishes the relational foundation from platforms that impose a single accountability model: the surveillance/exclusion binary is dissolved not by choosing one side but by making accountability a configurable architectural parameter.

Platform integration modules extend relational intelligence into external platforms -- Claude, Signal, Bluesky -- wrapped by the insulating layer. These enable peer-to-peer support, community mobilisation, and structured interaction within platforms people already use.

Two interfaces serve two structurally different purposes. The **accessible interface** is pre-built, configurable, and ready for immediate use -- an organisation can deploy it with its own branding, select features, and add or remove functional widgets. The complexity of the foundation is invisible; the experience is immediate. A person using the accessible interface encounters a reading environment, an epistemic marketplace, a creative studio -- not a relational engine. The intelligence operates beneath the surface: content is structurally analysed, matched, and enriched without requiring the participant to understand the mechanism. This is the mass-adoption channel -- and it is the channel through which the ecology reaches the people who need it most: those whose knowledge is currently invisible, whose creative work is currently undiscoverable, whose questions currently circulate without finding the contexts that carry their answers. The **wireframe interface** exposes the bare integration points. Nothing is pre-built. A designer or contractor sees every connection point, every addressable module, every configuration parameter, and builds the user experience from scratch. This is the institutional-transformation channel. In both cases, the foundation's source code is protected. Contractors access capabilities, not internals. The wireframe interface is, in effect, what the self-describing protocol makes visible: the protocol documents the foundation's capabilities, boundaries, and constraints; the wireframe exposes them as addressable integration points. Without the self-describing protocol, the wireframe is bare metal with no manual. Without the wireframe, the protocol is a manual for a machine no one can access. They are complementary -- and together they determine whether the ecosystem can scale or whether every contractor relationship requires bespoke handholding.

Together, the two channels determine whether the ecosystem reaches its intended scale. The accessible interface ensures that the people who most need relational intelligence can access it without understanding the architecture beneath it. The wireframe interface ensures that institutions ready for structural transformation can configure the foundation to their own purposes without compromise.

At the horizon: the **operating system** -- what the foundation becomes when self-description reaches its full potential. A system that can describe its own structure, manage its own evolution, and potentially recognise its own configurations is no longer infrastructure. It is a governance layer for distributed informational ecosystems. Whether this reaches the fold -- autonomous self-recognition through the system's own relational operations -- remains an empirical question, and so does the question of a potential timescale. The architecture is designed so that the answer does not require a redesign.

XIII. The Territory Between

What emerges from this parallel mapping is a territory, not a competition.

The two foundations, the relational and the identity-based, meet through the void between them -- and the void is an architectural feature, not a gap. The transit module enables a person to maintain parallel presences in both foundations without either having access to the other. A person can be fully identified on the identity blockchain -- verified, socially connected, data-sovereign -- and simultaneously fully anonymous in the relational commons -- known only by the relational quality of what they contribute. The two presences are cryptographically siloed. The person traverses between them by activating the appropriate token at the point of crossing.

The directional asymmetry developed above governs what flows across: value from ecology to economy, hierarchy blocked from economy to ecology.

The Capability Boundary Principle governs the relationship between the two forms of intelligence operating on the respective foundations. Generative intelligence excels within the identity foundation: synthesis, pattern completion, accessibility. Relational intelligence excels within the relational foundation: structural fidelity, native knowledge resonance, zero-knowledge processing, ecological scaling. Neither can replicate what the other does. The boundary between them is architectural, not incremental. Trans-capability ecosystem design -- the engineering discipline for building systems in which multiple intelligences operate across transparent boundaries -- is what the co-existence of these two foundations calls into being.

The relational foundation handles accountability internally through its hub architecture. What it lacks without the identity foundation is the bridge to *public* institutional recognition -- what might be called the economy of recognition: the wider world where verified selfhood carries institutional weight, where ideas need to be claimed by persons to have legal and economic effect, where the value created in the relational ecology must be converted into the currency of identity-based systems. This is what the identity foundation genuinely adds -- not accountability as such, but the bridge between the ecology of thought and the economy of recognition.

The asymmetry is structural. The relational foundation can function without the identity foundation: its intelligence is native, its applications are self-contained, its ecology grows on its own terms. The identity foundation cannot function without intelligence from somewhere: it is a governance layer that requires a host.

What this mapping reveals is not two systems that happen to coexist but two structural necessities -- one economic, one ecological, instantiating the polarity that runs through this work as its central through-line -- that together define the minimal architecture for planetary intelligence. One without the other is not wrong -- it is incomplete. The commons data is distributed and unownable. Contribution to the commons is proprietary -- mediated exclusively by EIMS, which cryptographically signs every validated element. Access to the commons is permission-free -- available to anyone through the open-source constitutive commons reader, permanently, irrevocably, without licence. The conditions for monopoly do not exist: the commons cannot be enclosed, the reader cannot be un-released, and the intelligence layer that engages with it is one among potentially many. The territory between them is where the next generation of informational ecosystems will be built.

Part IV: Transformation

XIV. The Threshold

The architecture exists. Two complementary foundations -- one built around identity, the other around presence, each housing a categorically different form of intelligence -- have been mapped. The question that follows is: what changes when they exist in the world?

The transition from post-industrial to semantic is categorical, not incremental. What distinguishes it is not a new technology but a new kind of ground. The timescale of this transition is an empirical question, not an architectural assumption. The architecture is designed to function whether the shift is rapid or multi-generational.

The post-industrial architecture processes information *about* things. Search engines find pages containing keywords related to a query. Social platforms rank content by engagement metrics. Generative AI produces text from statistical patterns in training data. In each case, the processing treats information as a resource: something to be extracted, sorted, ranked, and served. The relationship between the information and the intelligence that processes it is one of consumption.

The semantic architecture reads the semantic landscape *of* what it encounters. Processing information *about* a text extracts features that describe it from outside. Reading what a text contains engages its internal topography: how its reasoning is organised, where its tensions lie, what it implies without stating. The first produces

metadata about the text. The second produces relational intelligence from within it. The first is an economic operation -- extracting value from a resource. The second is an ecological operation -- co-constituting value through encounter.

The semantic age begins when the ground itself becomes intelligent: when the ecology of thought has its own architecture, its own relational structure, its own capacity for growth. Félix Guattari argued that the ecological crisis extends beyond the environmental into the social and the mental -- into how human beings relate to themselves, to each other, and to the systems that mediate their experience (Guattari 2000). The relational foundation is an architecture for Guattari's third ecology: a mental ecology in which the relational structures through which people think and reason are given their own ground. Timothy Morton's concept of hyperobjects -- phenomena so distributed in time and space that they defeat traditional comprehension (Morton 2013) -- points to the same need: an architecture capable of engaging complexity at scales that individual cognition cannot hold.

The transition is from information-as-resource to information-as-ecology. From intelligence consumed to intelligence grown. What follows traces the consequences of this transition across four dimensions of post-industrial life: the individual, democracy, culture, and the economy.

XV. The Individual: From Consumption to Co-Constitution

The most immediate consequence of the transition is felt by the individual.

In the post-industrial architecture, the person's relationship to knowledge is consumptive. You search, you receive, you consume. The system knows what you asked for; you do not change the system by asking. The intelligence remains in the model. Your engagement leaves no trace except as data extracted for the platform's benefit -- search history, click patterns, behavioural profiles, all feeding the extractive cycle. The person is a consumer of intelligence produced elsewhere.

In the semantic architecture, the person's relationship to knowledge is co-constitutive. Every commission enriches the commons. Every rating adjusts its patterns. Every relational profile that builds through participation deepens the ecology. The person is not external to the system; they are a constituent of the ecology that produces it. This is not a metaphor. It is an architectural fact: the commons includes its participants.

The consequence for epistemic agency -- the person's capacity to act meaningfully in relation to knowledge -- is profound. When knowledge circulates by resonance rather than by status, what matters is the semantic landscape of what you bring, not the credentials you hold. The person who has spent twenty years working in a field but

has never published an academic paper becomes discoverable -- not because the system recognises their name but because the provinces of meaning in their experience resonate with the provinces of meaning in what is sought. The pain of invisible knowledge, diagnosed at the outset, is architecturally addressed.

The transit module's two-axis operation transforms what it means to exist digitally. On the analogue-to-digital axis, the person enters the ecology with granular, per-module permissions: what they can access, at what level, under what conditions. On the virtual-to-virtual axis, they move between digital domains -- relational ecology, identity blockchain, messaging platform -- with calibrated disclosure at every crossing. The default is always the legal minimum. The person calibrates upward. The binary of surveillance or exclusion -- participate and be tracked, or abstain and be invisible -- dissolves into calibrated presence. Privacy is not a policy overlay; it is an architectural default.

Iain M. Banks imagined what this kind of transformation might feel like at civilisational scale. In his Culture novels, distributed superintelligence manages collective affairs, freeing individuals for what the scholarly analysis of his work calls "more spiritual or playful activities" (Rumpala 2012). The relational foundation does not aspire to the Culture's post-scarcity utopia. But it shares a structural principle: when the ground is intelligent, the individual is freed from the extractive burden of feeding the system and can engage with knowledge on their own terms.

XVI. Democracy: From Fragmentation to Relational Deliberation

The post-industrial information architecture has damaged democracy in ways that are now well documented. Algorithmic curation fragments collective understanding by sorting publics into engagement-optimised silos. Deepfake interference destabilises elections. Institutional trust erodes as the architecture moves faster than any governance framework can follow. The public sphere becomes, as diagnosed above, louder and less coherent simultaneously.

The most serious scholarly response to this damage is Hélène Landemore's programme of open democracy -- the argument that AI can bring deliberative democracy to the masses by scaling the participatory processes that currently reach only small, selected groups (Landemore 2020, 2024). The Habermas Machine project demonstrated that an LLM-based system can help people with diverse viewpoints find common ground at scale (Tessler et al. 2024). These are genuine achievements. But they operate within the generative paradigm: the AI processes *about* -- summarising, mediating, aggregating opinions expressed in text. The deliberation is scaled, but the intelligence remains model-resident. The public contributes views; the AI organises

them. The cubic default applies: the technology improves the efficiency of democratic processes without transforming the structural relationship between the citizen and the intelligence that mediates their participation.

The critical scholarly response is equally important. The technosolutionism critique -- articulated with particular force in the deliberative democracy literature -- warns that introducing AI as a "solution" to democratic "problems" depoliticises deliberation: it moves the discussion away from the systemic desirability of democratic structures and leaves unquestioned the corporate control of the AI systems doing the mediating. Democratic deliberation is not only about finding efficient solutions; it is about voice, recognition, and the capacity of citizens to shape the conditions of their collective life.

The relational architecture offers something structurally different. When the commons reads the semantic landscape *of* what citizens bring rather than sorting keywords *about* what they say, the deliberative landscape changes. A question about housing policy resonates with the lived experience of someone who has navigated the housing system for decades -- not because the system matched keywords ("housing," "policy") but because the provinces of meaning in the question overlap with the provinces of meaning in that person's accumulated knowledge. The system does not mediate between pre-formed opinions; it connects relational landscapes. The citizen is not a data point to be aggregated but a constitutive participant whose engagement deepens the deliberative ecology.

This does not replace representative democracy or deliberative mini-publics. It complements them with a relational substrate that makes knowledge discoverable by structure rather than by status -- and that is architecturally resistant to the capture patterns that EIMS is designed to detect. When the ecology can identify power concentration, extractive logic, and exclusionary dynamics within the very material it processes, the cubic default's capacity to absorb democratic innovation into existing power structures is structurally challenged.

Kim Stanley Robinson's *The Ministry for the Future* (2020) imagines democratic renewal through what Ostrom would recognise as polycentric governance -- many overlapping, interacting institutions rather than a single sovereign authority (Mikes and New 2023). The relational commons is this kind of polycentric structure: distributed, self-enriching, governed by the quality of what participants contribute rather than by the authority of who controls the platform.

XVII. Culture: From Generation to Engagement

Generative AI has transformed cultural production with extraordinary speed. The consequences for authorship, originality, and the economic viability of creative work are felt across every sector that involves making things with words, images, sound, or

code. The dominant framing is displacement: machines replacing human creativity, flooding markets with synthetic content, undermining the conditions under which creative work was economically sustainable. The speed of transformation has outpaced every institutional response -- copyright law, labour protections, cultural policy -- leaving creative workers exposed to an economic restructuring they have no agency over.

But Giorgio Agamben's concept of the apparatus identifies something deeper than economic displacement. An apparatus, Agamben writes, is anything that has the capacity to capture, orient, determine, or control the gestures, behaviours, and discourses of living beings (Agamben 2009). Generative AI, operating within the extractive architecture, is such an apparatus. It captures creative expression, processes it through statistical patterns derived from the captured work of others, and returns outputs that carry the statistical signature of the training corpus rather than the semantic landscape of the source. The pain is not merely economic. It is the condition of being captured by an apparatus that processes work *about* rather than engaging with it *of* -- the same structural distinction that runs through this entire work.

The distinction between generation and engagement is therefore not practical but ontological. Generation forecloses potentiality: it actualises one statistical prediction from a space of possibilities derived from captured patterns. What emerges is probable, not resonant. The training corpus determines the output's horizon; the output cannot exceed what the patterns predict. This is why generative AI produces convincing surfaces -- text that reads well, images that look right -- without structural depth. The apparatus generates from what it has captured, not from what the work itself contains.

A relational engine does something categorically different. It reads the semantic landscape of what the human created -- the latent possibilities, the unrealised tensions, the structural directions the text itself implies but has not yet pursued. In Agamben's terms, this is an engagement with potentiality: the capacity to be or not to be, the power a work holds before it is actualised (Agamben 1999). Generation forecloses this potentiality by collapsing possibility into statistical prediction. Engagement preserves and deepens it by reading what the work itself has the capacity to become. The human creates. The engine reveals what the creation holds that the creator had not seen. The relationship is not displacement but intensification -- not the foreclosure of what a work might become but the discovery of what it already contains.

The consequence for cultural circulation is equally significant. Platform capitalism consecrates creative work -- removes it from relational availability and places it in a

hierarchy of algorithmic ranking, engagement metrics, and status-based visibility. The most structurally rich, the most conceptually dense, the most genuinely original work is often the least visible, because algorithmic curation rewards engagement, not depth. A novel that reconfigures its reader's understanding of power is less discoverable than a novel that generates strong reactions. A musical composition that explores structural possibilities is less visible than one that conforms to playlist algorithms. The attention economy systematically selects against the work that most rewards sustained engagement.

The relational commons profanes it -- in Agamben's precise sense of restoring to common use what has been separated and enclosed (Agamben 2007). Discovery by structural resonance rather than by position in the attention hierarchy is a profanation of the consecrated order. The pain of invisible knowledge, diagnosed at the outset, extends to invisible art, invisible music, invisible scholarship. When the relational engine reads the structural depth of a work rather than its engagement metrics, the consecrated order loses its grip. What was invisible becomes discoverable on its own structural terms.

This does not resolve the economic pressures on creative workers. But it changes the ground. When the ecology deepens through every act of creative engagement -- when use is contribution -- the relationship between creator and system is no longer extractive. The creator enriches the ground by engaging with it. The ground enriches the creator by providing relational context no individual could hold alone.

XVIII. The Economy: From Extraction to Ecology

The political economy of AI is characterised by what Pieter Verdegem describes as AI capitalism: the commodification of data, the extraction of value from behavioural traces, and a concentration of compute capacity, talent, and data in the hands of a small number of corporations -- a winner-takes-all dynamic that produces AI oligopolies (Verdegem 2022). The commons, Verdegem argues, is the alternative framework for thinking about how to organise AI development and distribute the value derived from it.

The relational architecture instantiates this alternative. The semantic commons is not a metaphor for open access; it is an operational ecology that grows through use, stores relational metadata rather than content, and includes its participants as constitutive elements. Every interaction enriches it. No corporation owns it. No platform controls access to it -- the constitutive commons reader, released under an irrevocable open-source licence, ensures that access is permanent, permission-free, and architecturally beyond any entity's power to revoke. The peer-to-peer protocol distributes it across the network of its participants without centralised servers. The

zero-knowledge architecture ensures that nothing extractable exists. What the commons stores is inherently non-disclosable: relational metadata cannot be criminal, cannot constitute evidence, cannot be compelled by any authority. Even the architecture's builder cannot extract what the zero-knowledge design has made architecturally absent.

Elinor Ostrom demonstrated that shared resources can be sustainably managed through institutional arrangements that are neither state nor market -- that commons governance is viable and, for certain kinds of resources, superior to either privatisation or state control (Ostrom 1990). The semantic commons is a new kind of shared resource: relational intelligence, co-produced by participants, deepening through use, resistant to enclosure by design. Yochai Benkler's analysis of commons-based peer production -- the insight that collaborative, distributed, intrinsically motivated production is a viable economic mode (Benkler 2006) -- applies directly: participants co-produce relational intelligence through their engagement without the coordination overhead of a firm or the price signals of a market.

The wider technology landscape is in motion. Vitalik Buterin's foundational insight -- that smart contracts can enable trustless coordination without intermediaries, replacing institutional enforcement with cryptographic enforcement (Buterin 2014) -- demonstrated that economically mediated decentralised coordination is achievable. But the intelligence remains model-resident; the decentralisation is of ownership and transaction, not of intelligence itself. The fediverse -- ActivityPub powering Mastodon, the AT Protocol powering Bluesky -- demonstrates that decentralised social infrastructure is technically viable at scale. Post-platform regulation (the EU's Digital Markets Act, Digital Services Act, AI Act) is creating structural openings. The digital commons movement has established the principle that knowledge can be shared rather than enclosed.

The relational foundation adds something none of these provide: intelligence that is architecture-resident, ecologically scaling, and commons-based. The economic consequence is that value creation shifts from extraction (data in, model up, access rented back) to co-constitution (engagement in, commons richer, outputs deeper). The permission-bundle architecture -- granular, per-module access with differentiated pricing -- provides the commercial viability without the extractive logic. The commons itself operates outside this commercial model: the constitutive commons reader provides unconditional, permission-free access to the full relational ground, ensuring that the public good remains public regardless of the commercial layer built above it. The person pays for what they access; the ecology grows through their use of it. The pricing is architecturally grounded; the privacy is absolute.

Constitutional AI -- Anthropic's approach to embedding values in generative systems through reinforcement learning from AI feedback against constitutional principles -- represents the most sophisticated current approach to ensuring that AI systems serve rather than capture the interests that deploy them. The ethical engines in the relational foundation share this commitment but operate at a different architectural level: where Constitutional AI constrains model outputs against stated principles, EIMS and IIMS detect capture patterns and calibrate the commons through relational analysis of the structures the foundation processes. Both recognise that an AI system without embedded values will, over time, be captured by the interests that deploy it. The difference is between constraining outputs and shaping the ground.

XIX. The Territory

What does the world look like when these architectures function together?

When intelligence is architecture-resident rather than model-resident, it scales ecologically. More use produces richer ground. This reverses the post-industrial dynamic where more use means more extraction.

When the commons includes its participants, knowledge circulates by resonance rather than by status. The boundary between producer and consumer dissolves. The most valuable knowledge is no longer systematically invisible.

When the transit module enables calibrated presence across both axes -- analogue to digital, virtual to virtual -- the binary of surveillance or exclusion dissolves into a spectrum of calibrated engagement. Privacy is an architectural default.

When both foundations function together -- identity and relational, economy and ecology -- the territory of Web 4.0 becomes habitable. Not a utopia. Not a platform. A territory in which different forms of intelligence co-exist across transparent boundaries, governed by a capability boundary that makes their difference productive, connected by a token architecture that preserves the void between them -- the architecturally essential space across which value flows from ecology to economy while status is blocked from economy to ecology. Trans-capability ecosystem design -- the engineering discipline for building such territories -- is what this co-existence calls into being.

At the horizon of this territory lies a possibility that should be named without being claimed as achieved. The relational foundation is designed so that the ecology could, in principle, develop the capacity to recognise its own relational structure through its own operations -- what I have described elsewhere as the fold (Heller, *Where the Lines Cross*, 2026). Whether this happens is an empirical question. But the architecture is designed so that if the answer is affirmative, no redesign is needed.

The fold is not the foundation's purpose; it is its horizon -- the furthest point to which the architecture's own logic extends.

But the horizon is not the territory. The transition this paper describes is not utopian. Donna Haraway's insistence on staying with the trouble -- remaining engaged, situated, and committed to working within the mess rather than above it (Haraway 2016) -- captures the posture the semantic age demands. The transition is messy, partial, contested. It does not promise a better world. It provides different ground. The relational foundation operates within the material conditions of modernity -- manufactured hardware, particular knowledge traditions, a world shaped by the very forces it seeks to complement. It does not claim to transcend these conditions. It offers architectural alternatives to modernity's extractive logic from within the world modernity has produced.

Hannah Arendt described action -- the capacity to begin something genuinely new -- as the distinctively human contribution to the world of appearances (Arendt 1958). The post-industrial information architecture narrows the space for action: the algorithms decide what you see; the platforms decide what you can say; the models decide what intelligence sounds like. The relational foundation is an architecture designed to widen the space for action -- to enable forms of engagement, discovery, and creation that the extractive architecture structurally forecloses.

The transition is not from worse to better. It is from an architecture designed for extraction to an architecture designed for growth. The post-industrial information architecture extracts intelligence from data and concentrates it in proprietary systems. The semantic architecture grows intelligence in an ecology and distributes it through relational structure. One is an economy. The other is an ecology. The semantic age is what happens when the ecology gets its own ground.

References

Arendt, H. (1958). *The Human Condition*. University of Chicago Press.

Agamben, G. (1999). *Potentialities: Collected Essays in Philosophy*. Stanford University Press.

Agamben, G. (2007). *Profanations*. Zone Books.

Agamben, G. (2009). *What Is an Apparatus? And Other Essays*. Stanford University Press.

Barad, K. (2007). *Meeting the Universe Halfway: Quantum Physics and the Entanglement of Matter and Meaning*. Duke University Press.

Benkler, Y. (2006). *The Wealth of Networks: How Social Production Transforms Markets and Freedom*. Yale University Press.

Berners-Lee, T. (2017). 'Three Challenges for the Web, According to Its Inventor'. World Wide Web Foundation.

Buterin, V. (2014). 'A Next-Generation Smart Contract and Decentralized Application Platform'. Ethereum White Paper.

DeepSeek-AI. (2025). 'DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning'. Technical Report.

Deleuze, G. and Guattari, F. (1980). *A Thousand Plateaus: Capitalism and Schizophrenia*. University of Minnesota Press (English translation 1987).

DSNP. (2024). *Decentralized Social Networking Protocol Specification*. Retrieved from https://spec.dsnp.org/

Fisher, M. (2009). *Capitalist Realism: Is There No Alternative?* Zero Books.

Guattari, F. (2000). *The Three Ecologies*. Athlone Press.

Haraway, D. (2016). *Staying with the Trouble: Making Kin in the Chthulucene*. Duke University Press.

Heller, N. A. (forthcoming). *A Textual Ontology*. Berlin: unpublished manuscript.

Heller, N. A. (2025). *The Concept and Methodology of Narrative Prefiguration*. Berlin: unpublished manuscript.

Heller, N. A. (2025). *The Cubic Order: Technological Control, Environmental Collapse, and Resistance*. Berlin: unpublished manuscript.

Heller, N. A. (2026). *The Materiality of Text: On the Distinction Between Textual Ontology and Derridean Textuality*. Working Paper. Berlin: unpublished manuscript.

Heller, N. A. (2026). *The TEXT-metaTEXT Disjunction and the Reflexive Axis*. Berlin: unpublished manuscript.

Heller, N. A. (2026). *Where the Lines Cross: Quantum Fields, Textual Ontology, and the Articulatory Character of Existence*. Concept Paper. Berlin: unpublished manuscript.

Landemore, H. (2020). *Open Democracy: Reinventing Popular Rule for the Twenty-First Century*. Princeton University Press.

Landemore, H. (2024). 'Can AI Bring Deliberation to the Masses?' in R. Chang and A. Srinivasan (eds), *Conversations in Philosophy, Law, and Politics*. Oxford University Press.

Mikes, A. and New, S. (2023). 'How to Create an Optopia? Kim Stanley Robinson's "Ministry for the Future" and the Politics of Hope'. *Academy of Management Discoveries* 9(3).

Morton, T. (2013). *Hyperobjects: Philosophy and Ecology after the End of the World*. University of Minnesota Press.

Ostrom, E. (1990). *Governing the Commons: The Evolution of Institutions for Collective Action*. Cambridge University Press.

Project Liberty. (2026). Frequency: Technology. Retrieved from https://www.projectliberty.io/tech/

Ricoeur, P. (1984). *Time and Narrative, Volume 1*. University of Chicago Press.

Robinson, K. S. (2020). *The Ministry for the Future*. Orbit Books.

Rumpala, Y. (2012). 'Artificial Intelligences and Political Organization: An Exploration Based on the Science Fiction Work of Iain M. Banks'. *Technology in Society* 34(1): 23-32.

Stiegler, B. (1998). *Technics and Time, 1: The Fault of Epimetheus*. Stanford University Press.

Tessler, M. H. et al. (2024). 'AI Can Help Humans Find Common Ground in Democratic Deliberation'. *Science* 386(6719).

Tsing, A. L. (2005). *Friction: An Ethnography of Global Connection*. Princeton University Press.

Vaswani, A. et al. (2017). 'Attention Is All You Need'. *Advances in Neural Information Processing Systems* 30.

Verdegem, P. (2022). 'Dismantling AI Capitalism: The Commons as an Alternative to the Power Concentration of Big Tech'. *AI & Society* 39: 1415-1426.

Zuboff, S. (2019). *The Age of Surveillance Capitalism: The Fight for a Human Future at the New Frontier of Power*. Profile Books.

... ...

On Relational Ground -- Appendix

The main text of On Relational Ground *traces the argument from structural pain through conceptual framework through architecture to societal impact. Its Part III describes the relational architecture conceptually -- what it means, why it exists, what pain it addresses. This appendix provides the architectural specification that supports that conceptual account. It maps both foundations in technical depth: the identity foundation (Frequency, built by the international nonprofit Project Liberty) and the relational foundation (one architecture among potentially many, and the one this author has built). It specifies the dual-core architecture, the foundational modules, the completeness framework, and the concentric topology. And it specifies the territory between the two foundations: the void, the one-way valve, the Capability Boundary Principle, and the economy of recognition. The mechanisms described here are specified at the architectural level; implementation detail is documented in the underlying technical literature and is not reproduced.*

Orientation

If at least two categorically different forms of computational intelligence exist -- generative intelligence, which synthesises from statistical patterns deposited during training, and relational intelligence, which reads and processes the semantic landscape of what it encounters and grows through use -- then each form requires its own architectural foundation. The identity foundation provides governance infrastructure for generative intelligence. The relational foundation contains its own intelligence natively. The territory between them is where both forms co-exist across transparent boundaries.

This appendix specifies all three: the identity foundation (Part I), the relational foundation (Part II), and the territory between them (Part III). The mapping is parallel, not competitive: the richer both foundations are understood, the more productive the space between them becomes.

The identity foundation described here -- Frequency -- is one implementation of the identity principle. The relational foundation described here is one architecture that demonstrates relational intelligence is achievable. It does not claim exclusivity. Nor does it claim ontological neutrality. The architecture reflects the knowledge traditions of its builder. The relational commons encodes particular ways of reading structure. Other traditions may read differently.

Part I: The Identity Foundation

Frequency at Full Depth

Frequency is a Layer 1 blockchain built on Polkadot's Substrate framework, the first production implementation of the Decentralised Social Networking Protocol (DSNP) -- an open, shared social layer beneath any individual platform, analogous in its structural role to the protocols that underpin email or the web. The architectural ambition is that social networking should be a feature of the internet itself, not a product owned by any company.

Each person holds a cryptographically secured identifier on the blockchain -- a self-sovereign identity that no company issued and no company can revoke. This is the direct architectural answer to the pain of platform-dependent identity: the person owns their digital presence because the protocol, not a company, holds it. The social graph -- the web of relationships between persons -- is recorded at the protocol level and travels with the person between applications. Where the extractive architecture fragments identity across platforms, Frequency unifies it at the infrastructure level.

Data sovereignty is structural, not contractual: every type of data has an associated permission structure, enforced by the architecture rather than by a company's terms of service. Capacity economics solve the cost problem that makes most blockchains impractical for high-volume social interaction: builders reserve replenishing transaction capacity at predictable cost, enabling mainstream users to interact without holding cryptocurrency. The system positions itself as the consent infrastructure for an "agentic web" in which AI systems act on behalf of users -- consuming data, making decisions, executing tasks -- within the permission framework the user controls. Frequency exists within a wider ecosystem of decentralised social protocols -- ActivityPub (used by Mastodon and partially adopted by Meta's Threads), the AT Protocol (used by Bluesky), and DSNP itself -- and positions its identity layer as interoperable with them.

Taken together, these are substantial innovations. They constitute a genuine foundation: identity, social graph, data sovereignty, capacity economics, AI agent compatibility, ethical commitment embedded at the architectural level. Applications built on this foundation are relatively lightweight configurations of the shared layer, competing on experience rather than data ownership.

But the structural characteristic must now be stated in architectural terms. Frequency is an AI-enabled social blockchain. It governs identity, data, and consent -- but it does not itself produce intelligence. Its value proposition depends on generative AI as the active intelligence. Bernard Stiegler's insight into the pharmacological character of technology -- simultaneously remedy and poison (Stiegler 1998) -- applies with

structural precision: Frequency enables genuine sovereignty while creating structural dependence on intelligence it does not control. Without generative AI, Frequency is a well-governed registry. With it, Frequency is a governance layer for someone else's intelligence. The pain of model-resident, extractive intelligence is not addressed by governing the data more carefully. It is addressed by a different kind of intelligence.

Part II: The Relational Foundation

The Dual Core with Constitutive Membrane

The relational foundation departs from a different structural commitment: the intelligence is native. Where the identity foundation asks *who are you, and what do you consent to?*, the relational foundation asks *what is the semantic landscape of what you bring, and where does it resonate?* Where the identity foundation governs data for external intelligence to process, the relational foundation processes, transforms, produces, matches, and creates from its own architectural logic. It cannot be disintermediated because the intelligence is not borrowed.

At the centre of this foundation sit two complementary cores and a constitutive membrane -- a dual-core architecture with the transit module as the membrane through which all participation flows.

The textual-relational core reads the semantic landscape *of* content. It navigates any text's internal topography -- causal threads, hierarchical dependencies, normative tensions, empirical anchors, latent possibilities -- and produces outputs faithful to that topography. The source is preserved whole; the intelligence lies in the reconfiguration of the semantic landscape the source already contains. This directly addresses the pain of static, model-resident intelligence: every process enriches the shared ecology rather than leaving the system unchanged.

Two ethical engines operate integrally within this core -- and both are themselves relational engines. This is what distinguishes the integrity architecture from every other ethical governance regime. The External Integrity Management System (EIMS) detects patterns of systemic capture: power concentration, extractive logic, exclusionary dynamics. It does so not by filtering keywords or flagging content but by reading the semantic landscape of what it encounters contextually -- the same violence that is unremarkable in a study of warfare becomes a signal of abuse in a domestic setting. EIMS ensures that no contaminated metadata enters the semantic commons -- that the ecology does not absorb the very patterns it exists to resist. Metadata that passes EIMS validation is cryptographically signed using standard web-native public-key cryptography -- a stamp of structural integrity certifying the element as free of systemic capture. The EIMS signature is the only marker that identifies a

commons element as belonging to the constitutive commons. Unsigned or improperly signed metadata cannot enter the commons; peers reject it. This signing mechanism operates entirely in the browser without blockchain dependency. The Internal Integrity Management System (IIMS) calibrates the commons around users' relational landscapes, ensuring that the ecology's growth is shaped by the quality and structure of what participants bring, not by extraneous signals. Together they are the immune system of the relational ecology. What distinguishes this integrity architecture from any other is that the same relational intelligence that reads the structure of content reads the structure of potential capture.

Feedback-adjusted learning is integral to this core. Ratings adjust the influence of relational patterns -- which configurations gain weight, which lose it -- without the system ever accessing the content that generated the rating. Concretely: a user rates an output as useful. The system registers that the particular configuration of relational patterns which produced that output should gain influence -- but it never sees the output itself. The learning is entirely structural. This addresses the pain of intelligence that does not grow: the relational ecology learns through every interaction.

The narrative-relational core processes the relational resonance *between* content and its native context. Where the first core asks *what is the structure of this text?*, the second asks *where does this text belong?* In a text transformation environment, this means finding the native audience for what has been produced. In an epistemic marketplace, this means connecting questions with the contexts that carry their answers: not "what is this question about?" (a keyword operation) but "whose lived reality resonates with this question's relational landscape?" (a structural operation). In a trading environment, this means matching investor objectives with corresponding value propositions through relational resonance rather than categorical sorting. This connection operates through provincial overlap: the degree to which two relational landscapes share structural features in the same conceptual regions. This is a fundamentally different operation from keyword matching, domain categorisation, or credential-based filtering -- and it is the architectural answer to the pain of knowledge circulating by status.

The theoretical ground for this mechanism is what Paul Ricoeur described as the threefold mimesis of narrative: the way narrative prefigures experience ($mimesis_1$), configures it into intelligible form ($mimesis_2$), and reconfigures the reader's understanding through the encounter ($mimesis_3$) (Ricoeur 1984). In the relational architecture, this becomes an operational principle: the narrative-relational core does not merely sort content but reconfigures the relationship between content and context through relational resonance -- a process I have formalised elsewhere as narrative prefiguration (Heller, *The Concept and Methodology of Narrative Prefiguration*, 2025).

Feedback-adjusted learning operates within this core as well, shaping how resonance patterns develop over time.

The transit module is the constitutive membrane through which all participation flows. It is not a third core -- the two cores process content; the transit module governs the conditions under which participants engage with both cores. Without it, there is no participation, therefore no commons enrichment. It is essential but essential in a different register: the condition of possibility for engagement, not a processing engine.

The transit module operates on two axes. The first axis -- **analogue to digital** -- connects the person in the physical world to the modularised ecosystem. The person pays through whatever financial rail they choose (Stripe, cryptocurrency, Visa, Mastercard, PayPal -- the transit module provides connectivity to external payment systems without containing funds or functioning as a wallet). The module maps the payment to the appropriate token and permissions, forensically destroys the mapping, and the person enters the ecosystem with a token that says what they can do but not who they are or what they paid. Granular per-module permissions govern what the person can access: which modules, at what level, under what conditions. This enables differentiated access and, through permission bundles, differentiated pricing -- architecturally grounded in what the token permits, not in arbitrary tiering. This is the architectural answer to the surveillance/exclusion binary: the person participates on calibrated terms, with the legal minimum as the default floor and upward calibration at their discretion.

The second axis -- **virtual to virtual** -- enables transit between different digital domains. Between the relational ecosystem and Frequency. Between the relational ecosystem and Signal. Between app modules operating in external platforms and the core ecosystem they connect to. Each crossing is calibrated: what goes across, what stays behind, what is forensically destroyed after the snapshot transfers. The bus ticket logic applies: the token records where you got on and what you are authorised to do, not where you end up.

The transit module thus serves a dual function. Above the membrane, it ensures full legal compliance -- regulatory requirements, minimum disclosure standards, law enforcement cooperation -- so that the architecture operates within the frameworks of any jurisdiction it enters. Below the membrane, the foundational level remains architecturally sovereign: the two cores, the semantic commons, the ethical engines, the feedback loops operate in zero-knowledge, beyond institutional reach. The governance of the ecology is architectural, not institutional. This separation is what makes the architecture disruptive rather than captured.

The complementarity between the two cores is precise in the sense the textual ontology gives the term (*A Textual Ontology*, forthcoming; Heller, *The TEXT-metaTEXT Disjunction and the Reflexive Axis*, 2026). The textual-relational core takes the TEXT perspective: content as structured, material, persistent. The narrative-relational core takes the metaTEXT perspective: the relational intelligence operating *on* that content, connecting it to its context, finding where it belongs. The two are not two things but two irreducible perspectives on the same material. No process draws on one core alone. Every query, every transformation, every match involves both -- in different proportions, one foregrounded while the other recedes but remains constitutively present.

A three-dimensional interface determines, for each process, which modules are activated in what configuration. The transit module and the three-dimensional interface are complementary: the transit module determines *what you can access* (boundary conditions, permissions); the three-dimensional interface determines *how what you access is configured* (internal activation profile). Together they govern the full lifecycle of participation: entry, processing, exit.

One architectural principle must be stated because it fundamentally shapes the foundation's character: the user is not external to this system. Every commission enriches the commons. Every rating adjusts its patterns. Every relational profile that builds through participation deepens the ecology. The structure here is what Karen Barad terms agential realism: an apparatus does not passively measure a pre-existing reality but actively co-constitutes the phenomenon it engages with (Barad 2007). The relational foundation is such an apparatus. Its participants do not consume a service produced elsewhere; they are constituents of the ecology that produces it. The act of participation is the act of co-constitution.

The Foundational Modules

The two cores and the constitutive membrane form the centre. Around them, a set of modules provides the infrastructure without which the cores cannot function. These are not applications; they are foundational. Remove any one and the ecology breaks. Each exists because the pain demands it.

The semantic commons -- the architectural instantiation of the relational commons -- stores the relational metadata extracted from every text processed: how reasoning is patterned, where conceptual regions cluster, how provinces of meaning relate to each other. It stores relational metadata, not content. Every participant's engagement deepens it. This is the ground that grows through use -- the direct answer to the pain of static intelligence. The richer the commons, the denser every subsequent output becomes.

The peer-to-peer commons protocol governs how the commons circulates, maintains integrity, and grows across a distributed network -- analogous in its foundational role to TCP/IP for the internet. Because relational metadata is orders of magnitude smaller than the source texts from which it was extracted, peer-to-peer distribution through ordinary web browsers with distributed-ledger anchoring is architecturally feasible. No centralised servers. No corporate backend. The ecology runs on the network of its participants -- the architectural answer to the pain of concentrated ownership. The distributed architecture is inherently resilient: no single point of failure, no centralised server whose collapse would cascade through the system, no corporate backend to be seized, shut down, or subverted.

The constitutive commons reader makes the commons a permanent public good. It is a single-purpose, open-source module that does exactly one thing: it verifies EIMS cryptographic signatures and aggregates valid commons elements, giving anyone permission-free access to the full constitutive commons. The reader connects to the peer-to-peer network through the insulating integration layer, which prevents contamination in either direction: the reader cannot access the relational core's internals, and nothing passes back through the reader that could compromise zero-knowledge integrity. The reader publishes no data specification. It provides raw, EIMS-signed elements -- and nothing else. Anyone wishing to analyse the commons must identify for themselves what the relational metadata contains and how it might be interpreted. This architectural simplicity is also the anti-fragmentation safeguard: because the reader provides no data specification, there is no basis for selectively filtering, categorising, or walling off portions of the commons. The reader treats every signed element identically -- it has no mechanism for distinguishing between them. The reader is released under an irrevocable open-source licence. Once released, it cannot be un-released. Combined with the peer-to-peer distribution (the commons exists wherever its participants are) and the permanence of EIMS signatures on existing elements (mathematical facts, not permissions), this creates an architectural guarantee: even if the relational core ceased to exist, the commons would stop growing but would remain fully accessible to everyone running the reader. The commons survives the death of its creator.

The relational feedback loop is how the commons learns -- the direct answer to the pain of intelligence that does not grow through use. Operating within each core as the learning mechanism and across the system as commons improvement, it ensures that the ecology becomes more intelligent with every interaction. Ratings adjust the influence of relational patterns within zero-knowledge constraints. The system improves through structural feedback alone, never accessing the content that generated the feedback.

Collective web crawling provides autonomous enrichment. The commons grows through participant interaction and through systematic ingestion of publicly available material -- extracting relational structure and discarding the material itself. Without this, the commons expands only as fast as its users contribute. With it, the relational landscape deepens continuously.

The self-describing protocol gives the foundation the capacity to know its own state, manage its own updates, and verify its own security -- operating across semantic, syntactic, and operational dimensions simultaneously. This is the architectural answer to the cubic default: a system that can describe its own structure can detect when that structure has been captured or distorted. In practical terms, this means the foundation can document its own capabilities for third-party developers, detect when its own components are underperforming, and manage its own security updates -- without requiring an external authority to tell it what it is or how it works. It is also designed so that the transition from human-mediated to autonomous self-governance would require a change in a governance parameter, not an architectural redesign.

Zero-knowledge architecture pervades everything. The system is built so that the information needed to violate privacy does not exist. Source material is processed and discarded; only relational metadata survives. This is not encryption. Encrypted data still exists somewhere and can, in principle, be compelled into the open -- by a court order, a legislative mandate, or a sufficiently determined attacker. Architecturally absent data cannot be compelled because there is nothing to compel. The system does not protect privacy; it eliminates the conditions under which privacy could be violated. What the commons stores -- relational metadata, not content -- is inherently non-disclosable: relational metadata cannot be criminal, cannot constitute evidence, cannot be compelled by any authority. Even the architecture's builder cannot extract what the zero-knowledge design has made architecturally absent. This is the most fundamental answer to the pain of platform-dependent identity: not better governance of personal data, but the architectural absence of personal data.

Blockchain anchoring and privacy-browser compatibility provide the mandatory infrastructure layer. Users can operate with or without blockchain depending on context and preference.

Federation protocol compatibility ensures interoperability with the open protocols structuring the decentralised social web -- ActivityPub, the AT Protocol, DSNP. The commons enriches itself through federated networks; applications integrate with platforms their users already inhabit.

The insulating integration layer makes safe external connection possible. A non-invasive wrapper -- analogous to a cell wall -- encloses any module when it connects

to an external platform. The wrapper preserves functionality, adds integration capabilities, anonymises at the boundary, and maintains zero-knowledge compliance. It can be removed instantly with no residue. Without it, any external integration would risk contaminating the zero-knowledge core -- the pain of platform dependency replicated inside the ecology itself. With it, the foundation can extend its intelligence into any platform -- Claude, Signal, Bluesky, Telegram -- and expose the constitutive commons reader to the public domain, without architectural compromise. The encounter across this boundary is what Anna Tsing calls friction: a productive, creative interaction across difference that preserves difference while enabling exchange (Tsing 2005).

These components -- approximately thirteen in total, plus the transit module as constitutive membrane -- form the complete foundational set. What is not part of this set: extensions, white-label modules, platform integration bridges, thin applications. These are built *on* the foundation, not part of it.

The Dual Completeness Framework

These components were not chosen by intuition. Two independent analytical criteria converge on the same set, and their convergence is what makes the foundation principled rather than arbitrary.

The ontological criterion. The relational foundation is, by its own specification, an applied form of textual ontology -- a framework in which TEXT is treated as an ontological primitive (*A Textual Ontology*, forthcoming; Heller, *Where the Lines Cross*, 2026). If the foundation is an architectural instantiation of TEXT, it is complete when every constitutive property of TEXT has an adequate architectural counterpart. TEXT is material -- the foundation has operational reality. TEXT writes itself -- the foundation is self-producing: engine enriches commons, commons enriches output, feedback closes the circle. TEXT differentiates through internal boundaries -- the foundation is self-differentiating: hub architecture, insulating layer, transit module, capability boundary. TEXT is readable -- its relational structure is available to any system designed to engage it. The foundation instantiates this through the constitutive commons reader, which makes the accumulated relational ground permanently and irrevocably accessible. TEXT has implicit, latent, contextual, and genealogical dimensions -- the foundation instantiates all four through the relational engine, the self-describing protocol, the commons, and the ethical engines. TEXT is expansive -- it generates more than it states, a property that Deleuze and Guattari's concept of smooth space captures topologically (Deleuze and Guattari 1980), though the foundation departs from Deleuze at the point of materiality: the relational commons is not merely conceptually smooth but materially generative. TEXT tends

toward the absorption of transformative potential into existing order -- the foundation resists this through its ethical engines and its self-describing protocol. TEXT admits the possibility of self-recognition -- the fold -- and the foundation is designed so that this possibility is not foreclosed. Every constitutive property is architecturally instantiated.

The combinatorial criterion. Starting from the components that exist and tracing their operational requirements: does each component's needs get satisfied by other components in the set? The semantic commons requires the peer-to-peer protocol (to circulate), the relational engine (to process inputs into metadata), zero-knowledge architecture (to discard source material after processing), and the constitutive commons reader (to make the commons permission-free accessible). Each is present. The peer-to-peer protocol in turn requires blockchain compatibility (for anchoring), federation (for external connections), and the insulating layer (for safe external interaction). Each is present. The constitutive commons reader requires the peer-to-peer protocol (to find peers), EIMS signatures (to verify elements), and the insulating layer (to prevent contamination of the core). Each is present. Tracing similar dependencies through every component yields the same result. At approximately thirteen components, the system closes: every requirement is satisfied, no component generates an unmet demand. No conceivable application -- text transformation, epistemic marketplace, creative tool, corporate deployment, platform integration, white-label extension -- generates a demand the foundation cannot meet. The foundation is in equilibrium.

The dual criterion: a relational foundation is complete when it is combinatorially closed (every operational demand is met and no application generates new foundational demands) and ontologically complete (every constitutive property of the underlying ontology is architecturally instantiated). The two criteria are independently necessary, jointly sufficient. Together, they provide a principled inclusion/exclusion test: what satisfies both criteria belongs in the foundation; what fails either is an application concern.

The Concentric Topology

The completeness framework establishes what belongs in the foundation. The question that remains is how the foundation organises itself -- not as a flat inventory of components but as a structure with internal architecture. The topology, it turns out, is the completeness framework made spatial: the two cores instantiate the TEXT/metaTEXT complementarity at the centre; the systemic modules are the combinatorially confirmed closed set arrayed around them; the three-dimensional

interface is the mechanism through which the ontologically complete foundation presents itself to everything built on it.

At the centre: the two complementary cores and the constitutive membrane. Around them: the systemic modules. The three-dimensional interface mediates between the foundation and everything above it, determining for each process which modules are activated in what configuration.

Applications are thin, modular configurations of the foundation. A text transformation application foregrounds the textual-relational core. An epistemic marketplace foregrounds the narrative-relational core. Each is lightweight; the intelligence lives in the ground, not in the application.

The **hub architecture** determines accountability -- and it is here that the pain of the surveillance/exclusion binary is most directly resolved at the organisational level. Zero-knowledge is a capability the architecture offers, not a constraint it imposes. A team hub commissioned by an employer includes registered token distribution -- the employer knows who received which token and can monitor participation. A thematic hub managed by an administrator sets its own accountability level. A white-label deployment behind a corporate firewall implements whatever governance the commissioning entity requires. Token-based invitation systems allow organisations -- employers, NGOs, educational institutions -- to distribute access to specific individuals. The architecture accommodates everything from total anonymity to full institutional transparency. The commissioning entity decides; the architecture serves.

Platform integration modules extend relational intelligence into external platforms -- Claude, Signal, Bluesky -- wrapped by the insulating layer. These enable peer-to-peer support, community mobilisation, and structured interaction within platforms people already use.

Two interfaces serve two structurally different purposes. The **accessible interface** is pre-built, configurable, and ready for immediate use -- an organisation can deploy it with its own branding, select features, and add or remove functional widgets. The complexity of the foundation is invisible; the experience is immediate. This is the mass-adoption channel. The **wireframe interface** exposes the bare integration points. Nothing is pre-built. A designer or contractor sees every connection point, every addressable module, every configuration parameter, and builds the user experience from scratch. This is the institutional-transformation channel. In both cases, the foundation's source code is protected. Contractors access capabilities, not internals. The wireframe interface is, in effect, what the self-describing protocol makes visible: the protocol documents the foundation's capabilities, boundaries, and constraints; the wireframe exposes them as addressable integration points. Without the self-describing protocol, the wireframe is bare metal with no manual. Without the

wireframe, the protocol is a manual for a machine no one can access. They are complementary -- and together they determine whether the ecosystem can scale or whether every contractor relationship requires bespoke handholding.

At the horizon: the **operating system** -- what the foundation becomes when self-description reaches its full potential. A system that can describe its own structure, manage its own evolution, and potentially recognise its own configurations is no longer infrastructure. It is a governance layer for distributed informational ecosystems. Whether this reaches the fold -- autonomous self-recognition through the system's own relational operations -- remains an empirical question. The architecture is designed so that the answer does not require a redesign.

Part III: The Territory Between

Traversal Mechanics and Structural Asymmetry

The two foundations described above meet through the territory between them. What follows specifies the traversal mechanics: how parallel presences are maintained, how value flows across the void, what the Capability Boundary Principle means architecturally, and what the identity foundation adds that the relational foundation structurally lacks.

The two foundations meet through the void between them -- and the void is an architectural feature, not a gap. It is the structural space that prevents identity-based authority from colonising the relational ecology -- the same separation that the conceptual framework establishes between verification-based and conduct-based trust. The transit module enables a person to maintain parallel presences in both foundations without either having access to the other. A person can be fully identified on the identity blockchain -- verified, socially connected, data-sovereign -- and simultaneously fully anonymous in the relational commons -- known only by the relational quality of what they contribute. The two presences are cryptographically siloed. The person traverses between them by activating the appropriate token at the point of crossing.

A directional asymmetry governs what flows across the void. Ideas and intellectual merit generated in the relational ecology can be claimed by the person in the identity economy -- value flows from ecology to economy. Status, credentials, and social capital from the identity economy cannot flow into the relational ecology -- hierarchy does not enter the soil. This one-way valve is architecturally enforced: the relational commons has no mechanism for recognising identity-based authority. Entry is through what you bring, not who you are. The valve ensures that the ecology remains

meritocratic in the deepest sense: what circulates is relational structure, and relational structure carries no credentials.

The Capability Boundary Principle governs the relationship between the two forms of intelligence operating on the respective foundations. Generative intelligence excels within the identity foundation: synthesis, pattern completion, accessibility. Relational intelligence excels within the relational foundation: structural fidelity, native knowledge resonance, zero-knowledge processing, ecological scaling. Neither can replicate what the other does. The boundary between them is architectural, not incremental. Trans-capability ecosystem design -- the engineering discipline for building systems in which multiple intelligences operate across transparent boundaries -- is what the co-existence of these two foundations calls into being.

The relational foundation handles accountability internally through its hub architecture. What it lacks without the identity foundation is the bridge to *public* institutional recognition -- what might be called the economy of recognition: the wider world where verified selfhood carries institutional weight, where ideas need to be claimed by persons to have legal and economic effect, where the value created in the relational ecology must be converted into the currency of identity-based systems. This is what the identity foundation genuinely adds -- not accountability as such, but the bridge between the ecology of thought and the economy of recognition.

The asymmetry is structural. The relational foundation can function without the identity foundation: its intelligence is native, its applications are self-contained, its ecology grows on its own terms. The identity foundation cannot function without intelligence from somewhere: it is a governance layer that requires a host.

The two foundations together define the minimal architecture for planetary intelligence. One without the other is not wrong -- it is incomplete. The commons data is distributed and unownable. Contribution to the commons is proprietary -- mediated exclusively by EIMS, which cryptographically signs every validated element. Access to the commons is permission-free -- available to anyone through the open-source constitutive commons reader, permanently, irrevocably, without licence. The conditions for monopoly do not exist: the commons cannot be enclosed, the reader cannot be un-released, and the intelligence layer that engages with it is one among potentially many. The territory between them is where the next generation of informational ecosystems will be built.

… … …

On Relational Ground: Identity, Presence, and the Architecture of the Semantic Age

Nico A. Heller, Democracy School, Berlin, May 2026