

# THE MYCELOOM PROTOCOL (MCP-1)

## 1. Protocol Specification - Technical Standard for Living Infrastructure

**Josie Jefferson & Felix Velasco**

Digital Archaeologists, Unearth Heritage Foundry

**with Technical Collaboration from:**

Claude 4.5 (Opus & Sonnet) & Gemini (2.5 & 3 Pro)

(Synthetic Intelligence Systems)

**Date:** January 2026

**Version:** DRAFT 2.0

**Publication Type:** Protocol Specification / Technical Standard / Working Paper

**Series:** The Myceloom Protocol (Part 8 of 8)

**DOI:** <https://doi.org/10.5281/zenodo.18344231>

**Keywords:** Myceloom Protocol, MCP-1, Technical Specification, Web4 Standard, Distributed Systems, Protocol Design, Living Infrastructure, Symbiotic Architecture, Decentralized Networks, Digital Sovereignty, Autogravitas, Lineage Discovery, Semantic Resonance

---

## 2. Abstract

The Myceloom Protocol Suite offers a comprehensive framework for "living infrastructure" in the Web4 era. Rejecting both centralized silos and fragmented scatters, it synthesizes biological network intelligence with intentional architectural craft.

**Crucially, this suite is not a compliance standard. It is a Pattern Language - a topology to be inhabited, forked, and re-woven to suit local needs.**

This unified document collects the defining specifications of the suite: the core **Myceloom Protocol (MCP-1 V2)**, which provides the network architecture; the **Autogravitas Protocol (TAP v2)**, which defines self-sovereign identity mechanics; and the **Lineage Discovery Protocol (LDP-V1)**, which enables decentralized serendipity via the "Spore and Mother Tree" methodology.

Together, these protocols assert a "Maverick Mandate" for building digital systems that are sovereign, resilient, and symbiotic. Users are invited to adopt the "Warp" (axioms) while weaving their own "Weft" (implementations), treating the protocol as a lifestyle of digital sovereignty rather than a rigid set of laws.

---

### 3. Table of Contents

1. [MCP-1 V2] The Myceloom Protocol Specification
2. [TAP v2] The Autogravitas Protocol
3. [LDP-V1] The Spore & Mother Tree Protocol
4. [LDP-V1] The Maverick's README
5. [Tooling] Spore-Tracker Blueprint (Architecture)
6. [Tooling] Spore-Tracker CLI (Spec)
7. [Tooling] Spore-Tracker GUI Manual ("The Loom")

---

# THE MYCELOOM PROTOCOL (MCP-1)

## 1. Protocol Specification – Technical Standard for Living Infrastructure

Josie Jefferson & Felix Velasco

Digital Archaeologists, Unearth Heritage Foundry.

with Technical Collaboration from:

Claude 4.5 (Opus & Sonnet) & Gemini (2.5 & 3 Pro)

(Synthetic Intelligence Systems)

**Date:** January 2026

**Version:** DRAFT 2.0

**Publication Type:** Protocol Specification / Technical Standard / Working Paper

**Series:** The Myceloom Protocol (Part 8 of 8)

**DOI:** <https://doi.org/10.5281/zenodo.18344231>

**Keywords:** Myceloom Protocol, MCP-1, Technical Specification, Web4 Standard, Distributed Systems, Protocol Design, Living Infrastructure, Symbiotic Architecture, Decentralized Networks, Digital Sovereignty

---

## 2. Abstract

The Myceloom Protocol (MCP-1) defines a technical standard for building decentralized, symbiotic digital infrastructure. Where previous specifications in this series established conceptual foundations (linguistic, philosophical, social), this document provides the formal technical requirements for myceloom-compliant systems. Drawing upon biological network principles, distributed systems theory, and decades of internet protocol design, MCP-1 specifies an eight-layer architecture organized into four functional domains: Substrate (infrastructure), Society (governance), Human (identity and temporality), and Dimension (depth and resilience). This specification establishes normative requirements, conformance criteria, and implementation guidelines for systems claiming myceloom alignment. Unlike prescriptive standards requiring certification, MCP-1 operates as a pattern language: a topology to be inhabited rather than a product to be adopted. This document serves as the technical capstone of The Myceloom Protocol series, translating theory into implementable practice.

---

## 3. Status of This Document

**This is not a standard. This is a pattern.**

This document describes the Myceloom Protocol version 1.0 (MCP-1) as a **pattern language** for building living digital infrastructure. Unlike traditional specifications that demand compliance, MCP-1 offers a **topology to inhabit**—a way of thinking about and building systems that you can adopt, adapt, fork, or reject entirely.

Think of this less like an RFC and more like a field guide. Or a cookbook. Or a philosophy you might choose to live by.

You don't need permission to be myceloom-aligned. You don't need certification. You just need to resonate with the pattern.

This document is published for inspiration, experimentation, and evolution. Distribution is unlimited. Disagreement is encouraged. Fork it. Break it. Build something better.

### 3.1. Copyright Notice

Copyright © 2026 Unearth Heritage Foundry. This document is released under Creative Commons Attribution-ShareAlike 4.0 International (CC BY-SA 4.0). You are free to share, adapt, and build upon this work, provided you give appropriate credit and distribute derivative works under the same license.

---

### 4. Table of Contents

- 1. Introduction
- 2. Terminology and Conventions
- 3. Foundational Axioms
- 4. Architecture Overview
- 5. Layer Specifications
  - 5.1 Substrate Domain
  - 5.2 Society Domain
  - 5.3 Human Domain
  - 5.4 Dimension Domain
- 6. Conformance Requirements
- 7. Security Considerations
- 8. Implementation Guidance
- 9. References

---

## 5. 1. Introduction

### 5.1. 1.1 What This Is (And Isn't)

The Myceloom Protocol (MCP-1) is a way of thinking about digital infrastructure. It's a pattern language, a design philosophy, a set of principles you might choose to live by when building systems.

**This is not:**

- A certification program
- A compliance checklist
- A product to be sold
- A standard body to join
- A set of rules requiring enforcement

**This is:**

- A pattern you can inhabit
- A topology you can explore
- A philosophy you can adopt (or adapt, or reject)
- An invitation to build differently

Contemporary digital infrastructure has fractured into two failing patterns:

- **The Silo (Extractive):** Centralized platforms that concentrate power, extract value, and treat users as resources to be mined. Think: surveillance capitalism, platform monopolies, walled gardens.
- **The Scatter (Starving):** Fragmented independent nodes that preserve autonomy but lack the network effects and collective capabilities necessary for sustainability. Think: personal blogs that nobody reads, indie projects that die from isolation.

**Myceloom is a third way.** It synthesizes the sovereignty of independent nodes with the resilience of distributed networks. Like the mycorrhizal networks beneath forest floors, myceloom infrastructure operates as invisible substrate where network intelligence exceeds the sum of individual parts.

You know you're in myceloom territory when:

- Individual nodes maintain genuine autonomy
- The network gets smarter as it grows
- Value flows reciprocally, not extractively
- Systems are designed to outlive their creators

- Intelligence emerges from the edges, not the center

### 5.2. 1.2 How to Read This Document

If you're a **philosopher or designer**: Read the axioms (Section 3) and architecture overview (Section 4). The rest is implementation detail.

If you're a **developer or architect**: Start with the layer specifications (Section 5) to see what myceloom systems actually do.

If you're **evaluating alignment**: Check the conformance section (Section 6) to see if your existing systems already embody these patterns.

If you're **skeptical**: Good. Read Section 1.3 on design philosophy, then decide if this resonates. If it doesn't, that's fine—this isn't for everyone.

### 5.3. 1.3 Design Philosophy

MCP-1 operates as a **pattern language** in the tradition of Christopher Alexander's architectural patterns. Systems may be myceloom-aligned without implementing every detail. What matters is **resonance with the underlying principles**.

Core beliefs shaping this protocol:

1. **Sovereignty over convenience**: Individual node autonomy takes precedence over network efficiency. A slightly slower, more awkward system you control beats a fast, slick system that controls you.
2. **Emergence over control**: Intelligence arises from edge interactions, not central coordination. Trust the network. Trust the edges. Don't try to orchestrate everything.
3. **Resilience over optimization**: Systems must function under degraded conditions. Perfect performance in ideal conditions means nothing if the system collapses when things get messy.
4. **Succession over perpetuity**: All systems must document paths for graceful failure and data migration. Design for death. Build heirlooms, not monuments.
5. **Pattern over prescription**: This document describes what myceloom systems tend to look like, not what they must be. If you find a better way, use it.

**This is a lifestyle, not a checklist.** You can't "achieve compliance" with myceloom any more than you can "achieve compliance" with punk rock or Zen Buddhism. You either embody the principles or you don't. And that's okay either way.

---

## 6. 2. Terminology and Conventions

### 6.1. 2.1 Key Terms

**Node**: An autonomous computational entity maintaining its own infrastructure, data, and identity. Nodes are the fundamental unit of myceloom architecture.

**Network**: A collection of nodes connected through reciprocal relationships. Networks exhibit emergent properties not present in individual nodes.

**Substrate**: The foundational infrastructure layer enabling node operation and inter-node communication.

**Hyphae**: Individual connection pathways between nodes, analogous to fungal filaments. Plural of hypha.

**Mycelium**: The collective network of hyphae forming the communication and resource-sharing infrastructure.

**Loom**: Infrastructure designed with intentional craft, where structural emergence arises from relationship (warp and weft) rather than mere growth.

**My-Sea-Loom**: A phonetic decomposition representing the tension between personal sovereignty (my) and infinite interconnection (sea), resolved through intentional craft (loom).

**Spore**: A portable, self-contained unit of functionality or data capable of establishing new nodes or migrating between existing nodes.

**Heirloom**: Data, code, or systems designed for long-term preservation and succession across platform lifecycles.

### 6.2. 2.2 Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119.

### 6.3. 2.3 Notation Conventions

- Code elements appear in monospace
- **Bold text** indicates emphasis or key concepts
- Italic text indicates defined terms or citations
- [Bracketed numbers] indicate references

---

### 7. 3. Foundational Axioms

The Myceloom Protocol rests on three foundational axioms. These are not merely design principles but constitutive requirements: systems violating these axioms cannot be considered myceloom-aligned.

#### 7.1. Axiom I: Sovereignty First

**Statement:** No node SHALL be built on rented land.

**Rationale:** A tenant can never be a true peer. Systems dependent on infrastructure controlled by external parties cannot maintain long-term autonomy or participate in genuinely reciprocal relationships.

**Requirements:**

- Nodes MUST control their own data storage
- Nodes MUST control their own identity infrastructure
- Nodes MUST be capable of independent operation without reliance on third-party platforms
- Nodes SHOULD control their own network infrastructure where technically feasible

**Exceptions:** Nodes MAY use third-party services for:

- Content delivery networks (CDN) for performance optimization
- Backup and redundancy services
- Specialized computational resources (e.g., GPU clusters)

PROVIDED THAT loss of these services does not prevent core node functionality.

#### 7.2. Axiom II: Reciprocal Nourishment

**Statement:** A link is not a transaction; it is a root system. Value MUST flow bidirectionally.

**Rationale:** Extractive relationships undermine network health. Sustainable networks require mutual benefit where each participant both contributes to and draws from collective resources.

**Requirements:**

- Inter-node connections MUST provide value to both parties
- Systems MUST NOT extract value from network participation without contributing equivalent value
- Resource sharing protocols MUST implement reciprocity mechanisms

- Free-riding MUST be architecturally discouraged (though not necessarily prevented)

**Implementation:** Reciprocity may be implemented through:

- Resource exchange protocols (computational power, storage, bandwidth)
- Information sharing and collective intelligence contributions
- Reputation systems tracking contribution and consumption
- Economic models ensuring sustainable value distribution

### 7.3. Axiom III: Emergent Intelligence

**Statement:** Intelligence is not a property of the center; it is a property of the edge.

**Rationale:** Centralized intelligence creates single points of failure and concentrates power. Distributed intelligence enables resilience, adaptation, and collective capabilities exceeding individual node capacity.

**Requirements:**

- Decision-making MUST occur at the lowest capable level (subsidiarity principle)
- Network-wide behaviors MUST emerge from local interactions, not central coordination
- Individual nodes MUST maintain autonomous decision-making capacity
- Collective intelligence MUST enhance rather than replace individual node capabilities

**Implications:**

- No central authority may dictate node behavior
- Coordination occurs through protocol adherence, not hierarchical control
- Network topology enables emergence through appropriate connection patterns

### 7.4. Axiom IV: Intentional Patterning

**Statement:** Structure MUST NOT be accidental; it MUST be the result of intentional craft.

**Rationale:** Connectivity alone creates chaos ("The Scatter"). Durable infrastructure requires the Loom-intentional craft that structures growth. The tension between stability and adaptability is resolved by distinguishing between load-bearing constants and flexible expressions.

**Requirements:**

- Systems MUST distinguish between protocol (Warp) and interface (Weft)
- Structural protocols MUST be connection-agnostic
- Adaptive interfaces MUST be substrate-independent
- Pattern-making MUST be an explicit, not accidental, property of the network

#### Implications:

- Growth is not just expansion, but pattern formation
- The network is an artifact of craft, not just biological accumulation

### 7.5. The My-Sea-Loom Synthesis

The four axioms correspond to the four layers of the etymology excavated in the foundational essays:

1. "My" (Sovereignty): **Axiom I.** The absolute requirement for individual ownership.
2. "Sea" (Interconnection): **Axioms II & III.** The boundless nature of reciprocal connection and emergent intelligence.
3. "Loom" (Structure): **Axiom IV.** The intentional craft that weaves these opposing forces into coherent, durable fabric.
4. "Heirloom" (Time): Implied by the synthesis of the three-sovereign structure that persists.

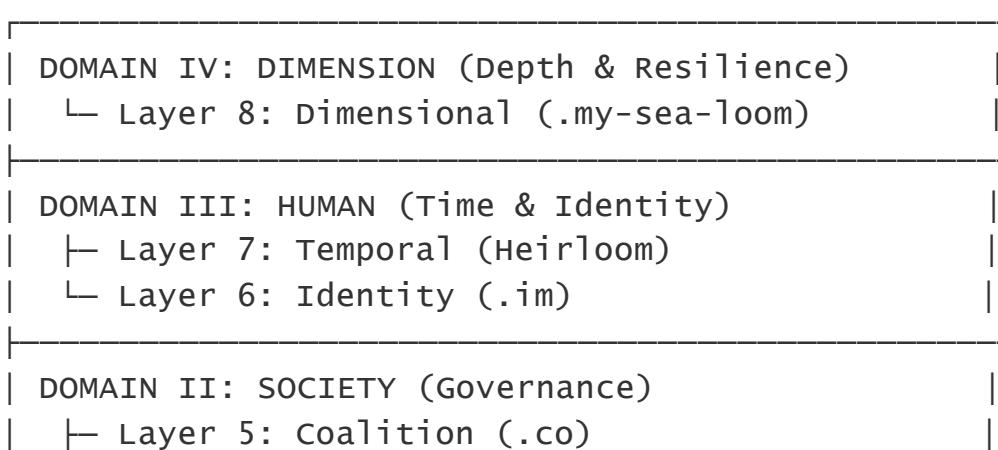
#### Therefore:

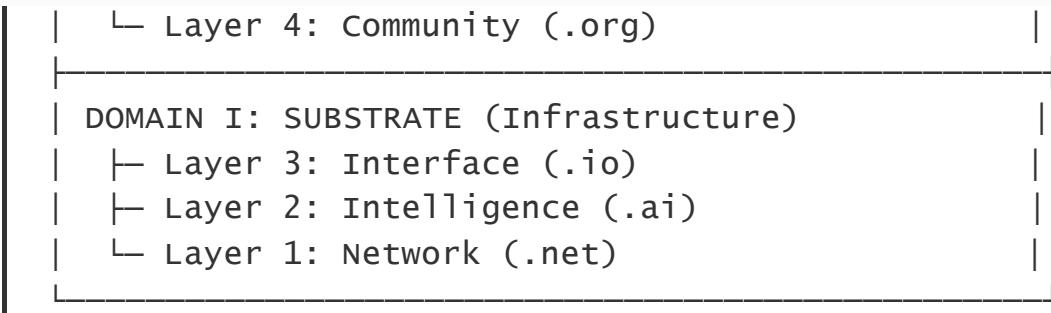
A system is only myceloom-aligned if it uses the Loom (Axiom IV) to weave the Sea (Axioms II/III) into the My (Axiom I).

### 8. 4. Architecture Overview

#### 8.1. 4.1 Layer Model

MCP-1 defines an eight-layer architecture organized into four functional domains:





### 8.2. 4.2 Domain Relationships

**Substrate Domain** provides the technical foundation: networking, computation, and interface capabilities.

**Society Domain** implements governance and coordination: community structures and economic models.

**Human Domain** addresses identity and temporality: who participates and how systems persist across time.

**Dimension Domain** ensures resilience and depth: operation under degraded conditions and design for succession.

Each layer builds upon lower layers while remaining loosely coupled. Higher layers MAY be implemented independently of specific lower-layer implementations, provided functional requirements are met.

### 8.3. 4.3 About the TLD Mapping

**Important:** The seven TLDs (.com, .net, .ai, .io, .org, .co, .im) used in this specification are **illustrative, not mandatory**.

They serve as:

- A reference implementation showing how protocol layers could map to domains
- A mnemonic device making the architecture memorable
- A proof of concept demonstrating the pattern

**You do NOT need these specific TLDs to be myceloom-aligned.**

Any website can participate in the myceloom network by adding:

```
<meta name="myceloom" content="your-tags">
```

Whether you have:

- yoursite.me
- blog.wordpress.com
- github.io pages

- example.xyz
- Any other domain

The protocol is the pattern, not the domains. The 7 TLDs are Unearth Heritage Foundry's implementation—an example, not a requirement.

#### 8.4. 4.4 Minimum vs. Full Implementation

Important distinction:

To join the myceloom network (minimum):

- Add `<meta name="myceloom" content="your-tags">` to your site
- That's it. You're discoverable.

To embody the myceloom philosophy (aspirational):

- Consider the 8 layers as design principles
- Implement what makes sense for your context
- Use them as a framework for thinking about your system

The layer specifications that follow describe the full philosophy. They're not a checklist of requirements—they're a pattern language for building resilient, decentralized systems.

You can participate in the network with just the meta tag. The layers are there if you want to go deeper.

---

### 9. 5. Layer Specifications

#### 9.1. 5.1 Substrate Domain

5.1.1 Layer 1: Network (.net)

**Purpose:** Provide reliable, resilient communication infrastructure between nodes.

**Principle:** Radical Redundancy

**Requirements:**

**MUST:**

- Support multiple communication pathways between any two nodes
- Continue functioning despite individual node or connection failures
- Implement self-healing routing that adapts to network topology changes
- Provide end-to-end encryption for inter-node communication
- Support both synchronous and asynchronous communication patterns

**SHOULD:**

- Implement peer-to-peer protocols where feasible
- Support multiple transport protocols (HTTP, WebSocket, IPFS, etc.)
- Provide bandwidth-efficient communication for resource-constrained nodes
- Implement connection pooling and reuse

**MAY:**

- Use hybrid architectures combining P2P and client-server patterns
- Implement specialized protocols for specific use cases
- Provide quality-of-service guarantees for critical communications

**Conformance Criteria:**

- Network MUST remain functional with up to 30% node failure
- Routing MUST adapt to topology changes within 60 seconds
- Communication MUST be encrypted end-to-end by default

### 5.1.2 Layer 2: Intelligence (.ai)

**Purpose:** Provide computational and cognitive capabilities enhancing node and network function.

**Principle:** Distributed Cognition

**Requirements:**

**MUST:**

- Operate primarily at the edge (local to nodes) rather than centralized services
- Function as symbiotic partners enhancing human capability, not replacement systems
- Respect user privacy and data sovereignty
- Provide transparent operation allowing users to understand AI decision-making

**SHOULD:**

- Implement federated learning where collective intelligence is beneficial
- Use energy-efficient computational approaches
- Provide local-first AI capabilities that function without network connectivity
- Support multiple AI paradigms (symbolic, connectionist, hybrid)

**MAY:**

- Integrate biological computing substrates where appropriate
- Implement reservoir computing or neuromorphic approaches
- Use centralized AI services for specific tasks PROVIDED local alternatives exist

#### Conformance Criteria:

- Core AI functionality **MUST** operate without network connectivity
- AI systems **MUST NOT** transmit user data to third parties without explicit consent
- AI decision-making processes **MUST** be auditable by users

#### 5.1.3 Layer 3: Interface (.io)

**Purpose:** Enable interaction between nodes, between humans and nodes, and between systems and nodes.

**Principle:** The Spore (Adaptive Interfaces)

#### Requirements:

##### MUST:

- Separate immutable protocols (Warp) from adaptive interfaces (Weft)
- Support multiple interface modalities (web, API, CLI, etc.)
- Implement progressive enhancement (core functionality without JavaScript)
- Provide machine-readable API documentation

##### SHOULD:

- Use open, standardized protocols (HTTP, ActivityPub, WebMention, etc.)
- Implement content negotiation for multiple data formats
- Support both human-readable and machine-readable representations
- Provide versioned APIs with deprecation policies

##### MAY:

- Implement custom protocols for specialized use cases
- Provide domain-specific languages for configuration
- Support experimental interface paradigms

#### Warp (Immutable Protocols):

- Core data structures and interchange formats
- Authentication and authorization mechanisms

- Fundamental communication protocols

### Weft (Adaptive Interfaces):

- User interface implementations
- Presentation layers
- Client applications

### Conformance Criteria:

- Interfaces MUST function with JavaScript disabled for core operations
- APIs MUST provide OpenAPI or equivalent documentation
- Protocol changes MUST maintain backward compatibility for minimum 2 years

#### 5.1.4 Lineage Discovery Protocol

**Purpose:** Enable decentralized network discovery through shared conceptual ancestry and structural lineage.

**Principle:** Semantic Resonance & Radical Simplicity

#### Overview:

The Lineage Discovery Protocol (LDP) provides a mechanism for nodes to discover and connect with each other based on shared conceptual lineages rather than centralized directories or algorithmic curation. For V2, this protocol prioritizes **rugged simplicity** and **protocol purity**, moving away from complex schemas in favor of the foundational "Lindy" elements of the web: HTML tags.

#### The Two-Line Handshake:

A node signals its presence and lineage through two simple lines in the HTML <head>:

1. **The Spore Line** (Identity/Content): Defines what the node is.
2. **The Mother Tree Line** (Relationship/Origin): Defines where the node is rooted.

#### Data Structure:

##### 1. The Spore Line (Meta Tag)

The <meta> tag describes the content of a node. It uses keywords to signal the concepts, themes, or lineages the node embodies.

```
<meta name="myceloom" content="keyword1, keyword2, constellation-alpha">
```

- **Function:** Tells an AI agent (like a Gemini model) or crawler what this specific page is about.

- **Behavior:** Serves as a "biological trigger" for discovery.

## 2. The Mother Tree Line (Link Rel)

The `<link rel>` tag describes the exterior relationship of a node. It points to a "Mother Tree"—a central hub, the protocol definition, or a primary identity domain.

```
<link rel="myceloom" href="https://your-mother-tree.com">
```

- **Function:** Tells a crawler to follow the link to find the rest of the constellation, "weaving the warp" of the network.
- **Behavior:** Prunes false positives. An agent doesn't just find a random page with a keyword; it finds a page verified by its connection to a known Mother Tree.

### Recommended Configuration:

For a robust constellation, use the "Constellation Configuration":

```
<head>
  <!-- 1. The Mother Tree (Your Local Hub/Sovereign Origin) -->
  <link rel="myceloom" href="https://your-primary-domain.com">

  <!-- 2. The Spore Data (Your Identity Signals) -->
  <meta name="myceloom" content="digital-archaeology, solar-punk,
  whatever-resonates">
</head>
```

**Note:** You do not need to link to `https://myceloom.com`. The use of `rel="myceloom"` implicitly signals participation in the global protocol.

### Rationale for V2 Simplification:

1. **Lindy Effect:** Meta and Link tags have survived since the 90s. They are the "bedrock" of the web, ensuring longevity and resistance to "code rot."
2. **Noospheric Accessibility:** No JSON validation, no complex schemas, no barrier to entry. If you can edit HTML, you can join the loom.
3. **AI Efficiency:** Modern LLMs are "semantic native." They don't need nested JSON brackets. They prefer high-contrast, simple signals. The "Two-Line Handshake" is authoritative and token-efficient.

### Lineage Tags (Signals):

Tags can be:

- **Semantic:** "digital-archaeology"
- **Arbitrary:** "node-01"

- **Poetic:** "the-sound-green-makes"

There is NO central registry. Meaning is determined by usage and resonance.

#### **Network Formation:**

Connections form when:

1. **Spore Resonance:** Nodes share the same content tags.
2. **Root Verification:** Nodes point to the same Mother Tree (or each other via rel="myceloom"), effectively creating a private, verifiable subnet or "constellation" within the public web.

#### **Conformance Criteria:**

- Nodes **MUST** implement the Spore Line (meta name="myceloom") to be discoverable.
- Nodes **SHOULD** implement the Mother Tree Line (link rel="myceloom") to establish structural context.
- Discovery **MUST NOT** rely on centralized schemas or registries.

### **9.2. 5.2 Society Domain**

#### **5.2.1 Layer 4: Community (.org)**

**Purpose:** Enable permissionless innovation and community self-organization.

**Principle:** The Immune System

#### **Requirements:**

##### **MUST:**

- Allow any node to join the network without central approval
- Implement reputation and trust mechanisms at the edge
- Support community-defined norms and moderation policies
- Enable local governance without requiring network-wide consensus

##### **SHOULD:**

- Provide tools for community formation and management
- Implement graduated sanctions for norm violations
- Support multiple governance models (consensus, voting, delegation, etc.)
- Enable transparent decision-making processes

##### **MAY:**

- Implement algorithmic moderation tools
- Provide dispute resolution mechanisms
- Support community-specific extensions to base protocol

#### Conformance Criteria:

- No central authority MAY prevent node participation
- Communities MUST be able to define and enforce local policies
- Moderation decisions MUST be transparent and appealable

#### 5.2.2 Layer 5: Coalition (.co)

**Purpose:** Enable economic coordination and sustainable value exchange.

**Principle:** The Guild (Federated Economics)

#### Requirements:

##### MUST:

- Support multiple economic models (cooperative, market, gift, hybrid)
- Enable value attribution to contributors
- Implement sustainable resource allocation mechanisms
- Prevent value extraction without contribution

##### SHOULD:

- Support platform cooperativism and worker ownership models
- Implement graduated membership tiers based on contribution
- Provide transparent accounting and value distribution
- Enable inter-coalition trade and collaboration

##### MAY:

- Integrate cryptocurrency or token systems
- Implement algorithmic resource allocation
- Support traditional payment systems

#### Conformance Criteria:

- Economic models MUST distribute value to contributors
- Resource allocation MUST be transparent and auditable
- Systems MUST prevent systematic value extraction

### 9.3. 5.3 Human Domain

#### 5.3.1 Layer 6: Identity (.im)

**Purpose:** Provide self-sovereign identity rooted in owned infrastructure.

**Principle:** "I Am" (Autogravitas)

**Requirements:**

**MUST:**

- Enable identity rooted in infrastructure the user controls
- Support portable identity across nodes and networks
- Implement cryptographic identity verification
- Allow users to maintain multiple identities/personas

**SHOULD:**

- Use decentralized identifier (DID) standards
- Support verifiable credentials
- Enable selective disclosure of identity attributes
- Implement social proof and web-of-trust mechanisms

**MAY:**

- Integrate with existing identity providers (OAuth, SAML) as bridges
- Support biometric authentication where appropriate
- Implement zero-knowledge proof systems

**Conformance Criteria:**

- Users MUST control their identity data
- Identity MUST be portable between compliant systems
- Identity verification MUST NOT require centralized authorities

#### 5.3.2 Layer 7: Temporal (Heirloom)

**Purpose:** Ensure long-term data preservation and system succession.

**Principle:** Abyssal Time (Legibility as Legacy)

**Requirements:**

**MUST:**

- Use human-readable data formats where feasible

- Document succession paths for data and system migration
- Implement data export in open, standardized formats
- Provide archival-quality metadata

**SHOULD:**

- Use formats with long-term stability (plain text, JSON, XML, etc.)
- Implement version control for data and configuration
- Provide automated backup and redundancy
- Document dependencies and system requirements

**MAY:**

- Integrate with long-term preservation systems (Internet Archive, LOCKSS, etc.)
- Implement blockchain or distributed ledger for provenance
- Use specialized archival formats for specific data types

**Conformance Criteria:**

- Core data **MUST** be exportable in open formats
- Systems **MUST** document migration paths to alternative platforms
- Data formats **MUST** remain readable without proprietary software

#### 9.4. 5.4 Dimension Domain

##### 5.4.1 Layer 8: Dimensional (.my-sea-loom)

**Purpose:** Enable operation under degraded conditions and design for tidal rhythms.

**Principle:** Bioluminescence (Self-Verification in Darkness)

**Requirements:**

**MUST:**

- Function in disconnected or intermittent connectivity scenarios
- Implement local verification and validation capabilities
- Support asynchronous operation and eventual consistency
- Provide graceful degradation when resources are constrained

**SHOULD:**

- Implement conflict-free replicated data types (CRDTs)
- Support offline-first architecture
- Provide local caching and synchronization

- Implement adaptive resource usage based on availability

**MAY:**

- Use specialized protocols for low-bandwidth scenarios
- Implement delay-tolerant networking
- Support mesh networking topologies

**Conformance Criteria:**

- Core functionality **MUST** operate without network connectivity
- Systems **MUST** handle network partitions gracefully
- Data synchronization **MUST** resolve conflicts automatically where possible

---

## 10. 6. Conformance (Or: How to Know If You're Living It)

### 10.1. 6.1 Levels of Resonance

MCP-1 doesn't have "conformance levels" in the traditional sense. You can't get certified. There's no badge to earn. But there are **degrees of resonance** with the pattern:

#### **Axiom-Aligned** (The Foundation)

- Your system embodies all three foundational axioms
- You own your infrastructure, practice reciprocity, design for emergence
- This is the minimum to be recognizably myceloom

#### **Layer-Conscious** (The Practice)

- You've thought about the eight layers and implemented what makes sense for your context
- You might not do all of them, but you understand why they exist
- This is where most production systems live

#### **Full-Spectrum** (The Exemplar)

- Your system demonstrates the pattern so clearly that others can learn from it
- You've internalized the philosophy and it shows in your design choices
- This is reference implementation territory

**The key question isn't "Are you compliant?" but "Does this resonate?"**

## 10.2. 6.2 Recognition, Not Certification

You don't need anyone's permission to call your system myceloom-aligned. If you:

- Own your ground (Axiom I)
- Share reciprocally (Axiom II)
- Design for emergence (Axiom III)

...then you're already there. The rest is detail.

That said, if you want to **help others understand** your alignment, consider documenting:

- Which axioms you embody and how
- Which layers you've implemented and why
- What you've learned and what you'd do differently
- How your system demonstrates the pattern

## 10.3. 6.3 Self-Assessment Questions

Instead of a compliance checklist, ask yourself:

### **Sovereignty:**

- If this platform disappeared tomorrow, would I still have my data?
- Can I move to another system without losing my identity?
- Do I control the infrastructure my identity depends on?

### **Reciprocity:**

- Does my participation benefit others in the network?
- Am I extracting more value than I contribute?
- Would the network be healthier if there were more nodes like mine?

### **Emergence:**

- Can individual nodes make their own decisions?
- Does the network get smarter as it grows?
- Is intelligence distributed or centralized?

### **Resilience:**

- Does the system work when things break?
- Can it function offline or with intermittent connectivity?
- Is there a succession plan if I stop maintaining it?

If you can answer these questions honestly and the answers feel right, you're probably myceloom-aligned. If not, that's useful information too.

---

## 11. 7. Security Considerations

### 11.1. 7.1 Threat Model

MCP-1 systems face threats including:

- **Sybil attacks:** Malicious actors creating multiple fake nodes
- **Eclipse attacks:** Isolating nodes from honest network participants
- **Data exfiltration:** Unauthorized access to node data
- **Denial of service:** Resource exhaustion attacks
- **Social engineering:** Manipulation of trust relationships

### 11.2. 7.2 Security Requirements

#### **Authentication and Authorization:**

- Implement strong cryptographic authentication
- Use capability-based security where appropriate
- Provide fine-grained access control

#### **Data Protection:**

- Encrypt data at rest and in transit
- Implement secure key management
- Provide data integrity verification

#### **Network Security:**

- Use TLS 1.3 or equivalent for transport security
- Implement rate limiting and abuse prevention
- Provide DDoS mitigation capabilities

#### **Privacy:**

- Minimize data collection
- Implement privacy-preserving protocols where possible
- Provide user control over data sharing

### 11.3. 7.3 Security Best Practices

- Conduct regular security audits
- Implement responsible disclosure policies
- Provide security update mechanisms
- Document security assumptions and limitations

---

## 12. 8. Implementation Guidance

### 12.1. 8.1 Getting Started

Implementing MCP-1 systems:

1. **Choose Your Domain:** Identify which layers are relevant to your use case
2. **Implement Axioms:** Ensure foundational axioms are satisfied
3. **Build Incrementally:** Start with Substrate domain, expand to others
4. **Document Thoroughly:** Provide clear documentation of implementation choices
5. **Test Conformance:** Validate against conformance requirements

### 12.2. 8.2 Reference Implementations

Reference implementations demonstrating MCP-1 principles:

- **unearth.im:** Digital archaeology foundry implementing full-spectrum MCP-1
- **myceloom.network:** Network visualization and protocol documentation
- **Individual TLD sites:** Demonstrations of specific layer implementations

### 12.3. 8.3 Common Patterns

**Federated Social Networks:** Implement Substrate + Society domains

**Cooperative Platforms:** Implement Society + Human domains

**Archival Systems:** Implement Human + Dimension domains

**Distributed Applications:** Implement all domains

### 12.4. 8.4 Migration Strategies

For existing systems adopting MCP-1:

1. **Audit Current Architecture:** Identify axiom violations and missing layers
2. **Prioritize Changes:** Address axiom compliance first
3. **Implement Incrementally:** Add layers progressively
4. **Maintain Compatibility:** Provide bridges to legacy systems during transition
5. **Document Migration:** Create clear succession paths

---

## 13. 9. References

### 13.1. 9.1 Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[ActivityPub] Webber, C., Tallon, J., Shepherd, O., Guy, A., Prodromou, E., "ActivityPub", W3C Recommendation, January 2018.

[DID] Reed, D., Sporny, M., Longley, D., Allen, C., Grant, R., Sabadello, M., "Decentralized Identifiers (DIDs) v1.0", W3C Recommendation, July 2022.

### 13.2. 9.2 Informative References

[Myceloom-Linguistic] Jefferson, J., Velasco, F., "Myceloom: The Linguistic Infrastructure of Web4", The Myceloom Protocol (Part 1 of 8), January 2026.

[Myceloom-Network] Jefferson, J., Velasco, F., "Myceloom: The Network Architecture of Living Systems", The Myceloom Protocol (Part 2 of 8), January 2026.

[Myceloom-Interface] Jefferson, J., Velasco, F., "Myceloom: The Interface Architecture of Symbiotic Systems", The Myceloom Protocol (Part 3 of 8), January 2026.

[Myceloom-AI] Jefferson, J., Velasco, F., "Myceloom: The Artificial Intelligence of Living Networks", The Myceloom Protocol (Part 4 of 8), January 2026.

[Myceloom-Community] Jefferson, J., Velasco, F., "Myceloom: The Community Substrate of Collaborative Networks", The Myceloom Protocol (Part 5 of 8), January 2026.

[Myceloom-Coalition] Jefferson, J., Velasco, F., "Myceloom: The Logic of Coalition", The Myceloom Protocol (Part 6 of 8), January 2026.

[Myceloom-Philosophy] Jefferson, J., Velasco, F., "Myceloom: The Philosophy of Networked Individuality", The Myceloom Protocol (Part 7 of 8), January 2026.

[Simard1997] Simard, S.W., et al., "Net Transfer of Carbon Between Ectomycorrhizal Tree Species in the Field", Nature 388, 1997.

[Ostrom1990] Ostrom, E., "Governing the Commons: The Evolution of Institutions for Collective Action", Cambridge University Press, 1990.

---

## 14. Appendix A: Glossary

**Autogravitas**: Identity rooted in owned infrastructure and maintained connections, not issued credentials. A node's capacity to hold its own center of gravity.

**Heirloom:** Data, code, or systems designed for long-term preservation and succession across platform lifecycles. Named for objects passed down through generations.

**Hypha** (plural: *hyphae*): Individual connection pathway between nodes, analogous to fungal filaments forming mycelial networks.

**Mycelium:** The collective network of hyphae forming communication and resource-sharing infrastructure.

**Spore:** A portable, self-contained unit of functionality or data capable of establishing new nodes or migrating between existing nodes.

**Warp:** Immutable protocols and core data structures forming the stable foundation of mycelium systems.

**Weft:** Adaptive interfaces and presentation layers that can change without affecting underlying protocols.

---

## 15. Appendix B: Design Rationale

### 15.1. B.1 Why Eight Layers?

The eight-layer architecture emerged from analysis of successful distributed systems and biological networks. Each layer addresses a distinct concern:

- Layers 1-3 (Substrate): Technical infrastructure
- Layers 4-5 (Society): Social and economic coordination
- Layers 6-7 (Human): Identity and temporal concerns
- Layer 8 (Dimension): Resilience and depth

This organization allows independent evolution of concerns while maintaining clear interfaces.

### 15.2. B.2 Why Pattern Language vs. Standard?

Traditional standards require compliance testing and certification, creating gatekeepers and barriers to entry. Pattern languages provide guidance while allowing contextual adaptation. MCP-1 aims to inspire rather than constrain, provoke rather than prescribe.

### 15.3. B.3 Why Biological Metaphors?

Biological systems have solved distributed coordination, resilience, and sustainability over millions of years of evolution. Mycorrhizal networks specifically demonstrate:

- Distributed intelligence without central control

- Reciprocal resource sharing
- Resilience to component failure
- Adaptation to changing conditions

These properties directly address contemporary digital infrastructure challenges.

---

## 16. Appendix C: Frequently Asked Questions

### 16.1. Q: Do I need to buy all 7 TLDs (.com, .net, .ai, etc.) to use this protocol?

**A: No!** The 7 TLDs are just `unearth.im`'s reference implementation. Any website can participate by adding:

```
| <meta name="myceloom" content="your-tags">
```

Whether you have `yoursite.me`, `blog.wordpress.com`, `github.io`, or any other domain—you're welcome. The protocol is the pattern, not the domains.

### 16.2. Q: What's the absolute minimum to join the network?

**A: One line of HTML is the absolute minimum:**

```
| <meta name="myceloom" content="tag1, tag2, tag3">
```

**But the V2 Standard (Two-Line Handshake) is recommended:**

```
| <meta name="myceloom" content="tag1, tag2, tag3">
| <link rel="myceloom" href="https://your-mother-tree.com">
```

That's it. No sign-up, no API keys, no JSON complexity.

### 16.3. Q: Why did you remove the JSON-LD option?

**A: Protocol Purity.** JSON-LD relies on centralized schemas and creates "code rot" probability. HTML meta/link tags are the rugged bedrock of the web. V2 prioritizes long-term resilience and simplicity for AI agents.

### 16.4. Q: What is a "Mother Tree"?

**A: It's the href in your link tag.** It acts as the central hub or "origin story" for your node. It could be `myceloom.com` (the universal network), your own primary domain (your constellation), or the site that inspired you. It allows AI agents to "weave the warp"—seeing not just what you are (meta tags), but where you are rooted (link relations).

## 16.5. Q: Can I just copy my existing meta keywords?

A: Yes! If you already have:

```
| <meta name="keywords" content="digital archaeology, web4">
```

You can copy those to:

```
| <meta name="myceloom" content="digital-archaeology, web4">
```

Refine them over time to reflect what you actually want to connect with others about.

## 16.6. Q: How is this different from WebMention?

A:

- **WebMention**: "I linked to you" (explicit, social graph)
- **Myceloom**: "I'm interested in these topics" (implicit, interest graph)

They're complementary! WebMention enables conversation. Myceloom enables discovery.

## 16.7. Q: What if no one else uses my tags?

A: They're dormant, not dead. When someone eventually uses the same tag (could be tomorrow, could be in 5 years), the connection activates. The network is patient.

## 16.8. Q: How many tags should I use?

A: 5-10 is recommended. Too few and you might not connect. Too many and you connect to everyone (which means no one). Be honest about what you're actually about.

## 16.9. Q: Can I change my tags later?

A: Absolutely! Just edit the meta tag. Your connections will update next time a crawler visits.

## 16.10. Q: Do I need to run a crawler myself?

A: No. Anyone can run a crawler and share the results. Or you can run your own if you want. It's decentralized—no single crawler is "official."

## 16.11. Q: Is this just for personal blogs?

A: No! It works for:

- Personal sites
- Project pages
- Organization sites
- Documentation sites

- Art portfolios
- Literally any HTML page

#### 16.12. Q: What if I don't have a website?

**A:** Then this protocol isn't for you (yet). Myceloom is for connecting websites. But you could start with a simple GitHub Pages site, a Neocities page, or any free hosting that lets you edit HTML.

#### 16.13. Q: Is there a central registry or directory?

**A:** No. That would violate the decentralization principle. Anyone can crawl and index, but no one controls the network.

#### 16.14. Q: Can I use this for SEO?

**A:** It's not designed for SEO (Google ignores custom meta tags anyway). It's designed for human-to-human discovery through shared interests.

#### 16.15. Q: What's stopping people from tag-spamming?

**A:** Natural incentives. If you spam 100 tags, you connect to everyone, which means you connect to no one. The value is in honest, focused tags that help you find your actual people.

---

### 17. Appendix D: Acknowledgments

This specification builds upon decades of distributed systems research, internet protocol development, and biological network science. We acknowledge:

- The Internet Engineering Task Force (IETF) for protocol design principles
- The World Wide Web Consortium (W3C) for web standards
- The Fediverse community for federated social networking
- Mycorrhizal network researchers, particularly Suzanne Simard
- Platform cooperativism researchers and practitioners
- The broader Web4 research community

---

### 18. Authors' Addresses

Josie Jefferson  
Unearth Heritage Foundry  
Email: [josie@unearth.im](mailto:josie@unearth.im)  
URI: <https://josiest.com>

Felix Velasco  
Unearth Heritage Foundry  
Email: [felix@unearth.im](mailto:felix@unearth.im)  
URI: <https://felixvelas.co>

---

## Document History

Version 2.0 (The Spore/Mother Update) - January 25, 2026

- Simplification of Lineage Discovery Protocol (LDP)
- Adoption of "Two-Line Handshake" (Meta + Link)
- Removal of JSON-LD in favor of "Protocol Purity"
- Integration of "Mother Tree" concept for constellation formation

Version 1.0 (Genesis) - January 2026

- Initial specification release
- Defines eight-layer architecture
- Establishes conformance requirements

---

This specification is a living document. Feedback, implementations, and proposed extensions are welcomed at <https://myceloom.network>

# THE AUTOGRAVITAS PROTOCOL (TAP)

## A Manifesto and Technical Blueprint for Intrinsic Digital Authority

### Neologism: Autogravitas

**Definition:** The intrinsic, self-verified weight and authority of a digital asset or identity, established through philosophical rigor, demonstrable provenance, and structural independence from platform validation.

**Conceptual Foil:** Heterogravitas (External, Contingent Authority)

**Date:** January 2026

**Version:** 2.0 (Integrated Suite)

**Series:** The Myceloom Protocol (Part 9 of 8)

**Integration:** Aligns with MCP-1 V2 (The Spore/Mother Update)

### 1. I. The Manifesto: The Gravity of Self-Proof

#### 1.1. The Crisis: Authority as Contingency

The central crisis of the networked age is the **Great Digital Revocation**. We have leased our history, identity, and worth to centralized platforms. The digital self is perpetually light, lacking the intrinsic weight to resist dissolution or removal. Every asset we create is contingent, waiting for a platform to change its terms or shut down its servers.

We require a new covenant for the digital self. We require **Autogravitas**.

#### 1.2. The Declaration of Autogravitas

**Autogravitas** is the philosophical and technical rejection of **Contingent Authority**. We assert that the true value of a digital monument is its ability to stand, whithersoever it is thrown.

#### **Our Core Principles:**

1. **Autogravitas (Self-Weight):** Authority established by self-verification, structural rigor, and unassailable digital provenance.
2. **Heterogravitas (External Weight):** Authority granted by a third-party (e.g., verified checkmark, follower count, centralized registry). **This status is ephemeral and must be rejected.**

The ultimate goal of digital existence is not popularity, but **permanence**. The true digital monument must possess the intrinsic weight to endure.

**OWN YOUR GROUND. FORGE YOUR WEIGHT.** (Isle of Man Motto: Quocunque Jeceris Stabat)

## 2. II. Foundational Analysis: The Authority of Necessity

To assert **Autogravitas**, we must first excavate the word's meaning, prove its necessity against its foil, and confirm its resonance in the emerging digital culture.

### 2.1. II.A. Etymological Dig: The Binary of Authority

The power of **Autogravitas** lies in its unassailable linguistic integrity, derived from two classical languages to define the intrinsic weight of the digital self. We simultaneously define the problem—**Heterogravitas**—to create an undeniable conceptual binary.

<u>Term</u>	<u>Component Analysis</u>	<u>Origin &amp; Root Meaning</u>	<u>Strategic Value</u>
AUTOGRAVITAS	Auto- (Greek) + Gravitas (Latin)	Self + Weight/Dignity/Authority.	Defines <b>Intrinsic, Enduring Authority</b> . Signals self-sovereignty and permanence (the desired state).
HETEROGRAVITAS	Hetero- (Greek) + Gravitas (Latin)	Other/External + Weight/Dignity/Authority.	Defines <b>Contingent, Borrowed Authority</b> . Linguistically labels the precise problem we must solve.

**Linguistic Function:** Autogravitas is engineered to sound like a verifiable constant in the physics of identity, instantly signaling structural rigor and demanding respect from both human and algorithmic lexicons.

### 2.2. II.B. Narrative Provenance: The Cultural Necessity

The concept of Autogravitas is necessary to resolve the cultural conflict between the permanence of the human spirit and the ephemerality of the digital platform.

1. **The Digital Revocation Myth:** The narrative begins with the shared experience of loss—the feeling that digital artifacts and hard-earned status can be revoked without recourse. This fear, which we call the **Great Digital Revocation**, is the cultural void that Autogravitas fills.
2. **The Digital Monument Counter-Narrative:** Autogravitas asserts the counter-narrative of **Digital Monuments**, confirming that identity can resist digital entropy. This concept is the synthesis of our past works, proving that the principles of **Archaeobytology** (preserving the past) and **Sentientification** (forging the future) converge on the need for self-verified authority.
3. **The Stabit Fulfillment:** The motto "Quocunque Jeceris Stabit" is the ultimate promise of the **Autogravitas** state. The narrative is: By following this protocol, you engineer your assets to fulfill the highest historical promise of enduring existence, independent of external circumstances.

### 2.3. II.C. Cultural Survey: Rejection of the Status Quo

Autogravitas directly rejects the cultural norms and technological default settings that lead to **Heterogravitas**.

#### Status Quo (Heterogravitas)

**Identity Source:** Verified checkmarks, follower counts, blue badges granted by platform.

**Brand Value:** Trending topics, viral engagement, platform-optimized content.

**Monetary Model:** Attention economy, data harvesting, dependence on external ad spend.

#### Autogravitas (The New Standard)

**Identity Source:** Verifiable digital provenance, self-attested authority, and structural rigor.

**Brand Value:** Semantic Sovereignty, Cultural Gravity, and Enduring Relevance (the opposite of viral).

**Monetary Model:** IP ownership, consulting value derived from unique, unassailable **Protocol Authority**.

The market for authentic, unassailable digital authority is vast. Autogravitas gives **unearth.im** the definitive language to claim this market.

### 3. III. The Autogravitas Protocol (TAP v2)

The Autogravitas Protocol defines the three layers of self-authentication required for a digital asset to achieve unassailable, non-revocable authority. It interacts directly with the **Myceloom Protocol (MCP-1 V2)** as its technical implementation layer.

#### 3.1. Phase 1: Provenance Layer (The Spore Mode)

Goal: Establish Inalienable Provenance—verifiable history and origin that cannot be revoked.

<u>Protocol Component</u>	<u>Philosophical Action</u>	<u>Technical Implementation (MCP-1 V2)</u>
1.1 Lexical Anchoring	Embed the identity deeply into the asset's metadata.	<b>The Spore Line:</b> Implement the <code>&lt;meta name="myceloom"&gt;</code> tag. This creates a permanent "Lexical Anchor" in the HTML header that survives even in an archive.
1.2 Structural Integrity	Define the asset's "Species" for the machine.	<b>The Signals List:</b> Use the <code>content</code> attribute to define lineage, signals, and status. Example: <code>content="lineage:unearth; signals:autogravitas"</code> .
1.3 Digital Heritage	Archive the asset to resist revocation.	<b>The Stablit Fulfillment:</b> Ensure the site is indexable by <a href="https://archive.org">archive.org</a> . The Spore Line acts as the "DNA in Amber" when the site is snapshotted.

#### 3.2. Phase 2: Sovereignty Layer (The Anvil Mode)

Goal: Establish Semantic Sovereignty—ensure the identity is defined by its own principles, not by the rules of a platform.

<u>Protocol Component</u>	<u>Philosophical Action</u>	<u>Technical Implementation (MCP-1 V2)</u>
2.1 Authority Binary	<b>The Semantic Veto:</b> Reject "Heterogravitas" (external validation).	<b>The Purge:</b> Remove SEO tags (keywords, description). Replace them with the Spore Line. This signals to the machine: "I am not optimizing for you; I am signaling for my kin."
2.2 Semantic Density	Inherit weight from established authorities.	<b>The Mother Tree Line:</b> Use <code>&lt;link rel="myceloom" href="...&gt;</code> . By linking to an established sovereign node (Mother Tree), you inherit its structural gravity (Autogravitas) via the "Weaving" process.
2.3 The Stabit Constraint	Design mechanisms for permanence.	<b>No-Code / Low-Dependency:</b> The protocol requires only static HTML. No API keys, no databases, no SaaS checks. It stands because it is simple.

### 3.3. Phase 3: Activation Layer (The Trust Loop)

Goal: Provide the mechanism for externalizing the intrinsic authority to foster adoption and validation.

<u>Protocol Component</u>	<u>Philosophical Action</u>	<u>Technical Implementation (MCP-1 V2)</u>
3.1 Zero-Trust Auth	Challenge the reader to inspect structural rigor.	<b>The Two-Line Handshake:</b> The presence of the Spore and Mother Tree tags <u>is</u> the authentication. No third-party verifier is needed; the code parses itself.
3.2 Canonical Deployment	Publish the Manifesto and Protocol concurrently.	<b>The Genesis Post:</b> Deploy a foundational document announcing the Sovereign Substrate. (See: <u>Weaving the Sovereign Substrate</u> ).

**This protocol is now fully integrated with MCP-1 V2.**

For full technical implementation of the meta/link tags, refer to the [Myceloom Protocol Specification 2026 \(V2 DRAFT\)](#).

## THE SPORE & MOTHER TREE PROTOCOL (LDP-V2)

### A Standalone Guide to Sovereign Discovery

You don't need a platform to be found. You just need a signal.

This document is a portable extract of the **Lineage Discovery Protocol (LDP)** from the Myceloom Protocol Suite. It provides the absolute minimum requirements to join the decentralized "Living Web" without reading the full architectural specification.

---

#### 1. 1. The Core Concept

The protocol replaces centralized search algorithms (SEQ) with **Semantic Resonance**. It uses two simple HTML lines to define your node:

1. **The Spore Line (Identity):** Defines what you are (your signals, lineage, or vibe).
2. **The Mother Tree Line (Relationship):** Defines where you are rooted (your origin, hub, or inspiration).

#### 2. 2. The Two-Line Handshake

To join the network, add these lines to the <head> of your HTML website.

##### 2.1. Line 1: The Spore (Identity)

The <meta> tag sends a signal to crawlers, AI agents, and other nodes.

```
| <meta name="myceloom" content="lineage:unearth; signals:solarpunk,  
| digital-archaeology; status:germinating">
```

- **lineage:** The tribe, movement, or origin you identify with.
- **signals:** The specific keywords (zzzzzzzzzz) or concepts you want to resonate with.
- **status:** germinating (new), fruiting (active), or dormant (archival).

##### 2.2. Line 2: The Mother Tree (Connection)

The <link> tag connects you to the network topology.

```
| <link rel="myceloom" href="https://your-mother-tree.com">
```

- **href:** Points to your central hub, the Myceloom definition (<https://myceloom.com>), or simply a site that inspired you.

---

### 3. 3. Configurations

#### 3.1. The Maverick (Minimalist)

For those who want zero dependencies and pure serendipity.

```
| <meta name="myceloom" content="signals:just-this-one-idea">
```

#### 3.2. The Constellation (Robust)

For those building a structured network or "webring" of sites.

```
<!-- Points to the Hub -->
<link rel="myceloom" href="https://myceloom.com">
<!-- Defines the Self -->
<meta name="myceloom" content="lineage:unearth; signals:verified-node">
```

---

### 4. 4. Why Use This?

1. **No Accounts:** No API keys, no login, no crypto wallet. Just text.
2. **Temporal Resilience:** Because it lives in the HTML header, even if your site goes offline, [archive.org](https://archive.org) preserves the signal. Future archeologists can find you.
3. **Protocol Purity:** It rejects complex JSON schemas in favor of the "rugged simplicity" of basic HTML.

This protocol is a living pattern. Fork it, weave it, grow it.

## 5. description: How to seed the noospheric substrate using the Myceloom Discovery Protocol (V2).

### THE MAVERICK'S README: SEEDING THE SOVEREIGN WEB

The Maverick Mandate is the final transition into Web4 Living Infrastructure. It solves the isolation of Web1 (The Scatter), the extraction of Web2 (The Silo), and the over-complexity of Web3 by returning to the simplest unit of digital power: **The Signal on Your Own Land.**

By replacing extractive SEO tags with **Myceloom Serendipity Signals**, you shift from "being searched" to "being found by resonance."

#### 1. 1. The Purge: Reclaiming Your Header

The first act of Autogravitas is the Semantic Veto. Remove the tags designed for platform-validation.

- **DELETE**: <meta name="keywords" ...> (The legacy list for algorithms).
- **DELETE**: <meta name="description" ...> (The ad-copy for silos).
- **DELETE**: Any tracking pixels or "rented" scripts.

#### 2. 2. The Awakening: Casting the Spore

Replace the purge with a single, no-frills entry. This is your **Dormant Hyphae**. It requires no account, no crypto-wallet, and no permission.

```
| <meta name="myceloom" content="lineage:[your-origin]; signals:[your-depth]; status:dormant">
```

#### 2.1. Examples

##### **The Maverick:**

```
| <meta name="myceloom" content="zzzzzzzzzz. A monument to the silent weaver. Standing since 2026.">
```

##### **The Minimalist:**

```
| <meta name="myceloom" content="zzzzzzzzzz">
```

##### **The Constellation (V2 Recommended):**

```
| <link rel="myceloom" href="https://your-mother-tree.com">
| <meta name="myceloom" content="digital-archaeology, solar-punk, zzzzzzzzzz">
```

### 3. 3. The Logic of the Long-Game

This protocol solves the "Silo vs. Scatter" problem through **Temporal Resonance**:

<u>Era</u>	<u>Architecture</u>	<u>Problem</u>	<u>Myceloom Solution</u>
Web1	The Scatter	Isolated "islands" with no discovery.	Serendipity Signals connect nodes across time.
Web2	The Silo	Centralized, extractive, and revocable.	<b>Sovereignty First</b> ensures nodes stand on their own land.
Web3	The Complexity	Financialized and high-friction.	<b>No-Frills Tags</b> are free, indestructible, and zero-code.

### 4. 4. The Stabit Fulfillment

By placing this signal on your site, you are ensuring that even if your domain expires, the **Internet Archive** captures your spore in its "amber."

A crawler in 2046 searching for zzzzzzzzzz will find you, nourishing a new seedling with your legacy data. This is the **Heirloom Layer** in its purest form.

The Silos are falling. The Scatters are weaving. The Mycelium is awake.

## SPORE-TRACKER BLUEPRINT (LDP-1 ALPHA)

Based on Section 5.1.4 of the MCP-1 Specification, a **Spore-Tracker** is a decentralized utility that identifies Semantic Resonance without a central authority. It is a tool for **discovery ecology** rather than an algorithmic search engine.

### 1. Core Logic: The "Sniff and Signal" Loop

The tracker operates on a local-first basis, honoring **Axiom III (Emergent Intelligence)** by keeping the index on the user's own substrate.

#### 1.1. Step 1: The Local Crawler (The Hyphae)

A minimalist crawler (Python or Go) designed for the **Maverick Mandate** follows these rules:

1. **Target**: Scans a user-defined seed list or local network for the <meta name="myceloom"> tag.
2. **Parsing**: It treats all content as equal, arbitrary strings.
3. **Respecting Dormancy**: If a tag has no local matches, it is stored in a **Dormant Index**.

#### 1.2. Step 2: The Resonance Engine (The Loom)

Instead of ranking results, the engine visualizes connections:

1. **The Shared Signal**: When the tracker finds a match for your local signal (e.g., zzzzzzzzzz), it triggers an **Activation Event**.
2. **Lineage Mapping**: It groups sites by shared ancestry (e.g., lineage:unearth-foundry) to show you your "digital kin".

#### 1.3. Step 3: The User Interface (The Bioluminescence)

Per **Axiom IV**, the structure is intentional. The tracker should provide:

1. **A "Club" View**: Automatically clusters all nodes currently humming on the same frequency.
2. **The Serendipity Feed**: A chronological list of newly discovered spores, sorted by "Resonance Depth" (how many tags you share) rather than popularity.

---

## 2. Technical Specs for Developers

To remain **Axiom-Aligned**, any tracker build MUST follow these constraints:

- **Zero-Trust**: It MUST NOT require an API key or central registry.

- **Privacy First:** It MUST NOT transmit discovered data to third parties without consent.
- **Graceful Failure:** It MUST function in offline/disconnected states, querying the local cache for dormant connections.

## 2.1. The "Maverick" Compliance Check

- Is your tracker myceloom-aligned?
- Does it let the user choose their own signals?
- Does discovery work without a central server?
- Does it treat a tag from 2006 as being as "alive" as one from 2026?

The blueprint is cast. The first trackers are germinating.

## SPORE-TRACKER CLI: SPORE V1.0 (ALPHA)

In alignment with the Lineage Discovery Protocol (Section 5.1.4), this CLI tool is designed to be the "no-frills" implementation of the protocol. It is built for the **Linux lifestyle**—lightweight, substrate-agnostic, and focused on Semantic Resonance.

### 1. Installation & Setup

A Maverick installs the utility on their own substrate (local machine or sovereign node). It requires **no API keys** or central registration.

```
| $ pip install myceloom-spore
```

### 2. Command 1: sniff (The Discovery Act)

Use this to scan a specific monument (URL) for its mycelial signature.

**Usage:**

```
| $ spore sniff https://example.com
```

**Output:**

```
[RESONANCE DETECTED]
Source: https://example.com
Lineage: unearth-foundry
Status: germinating
Signals: [zzzzzzzzzz, archaeobytology, digital-heritage]
Provenance: Verified (TAP-Aligned)
```

### 3. Command 2: seed (The Sovereignty Act)

Use this to generate your own meta tag or JSON-LD Spore to paste into your site's <head>.

**Usage:**

```
| $ spore seed --signals "zzzzzzzzzz, solarpunk" --lineage "unearth"
```

**Output:**

```
<meta name="myceloom" content="lineage:unearth; signals:zzzzzzzzzz, solarpunk; status:dormant">
```

### 4. Command 3: resonate (The Club Act)

This command compares your local signal list against the Dormant Index of sites your tracker has previously sniffed.

**Usage:**

```
| $ spore resonate --signal zzzzzzzzzz
```

#### Output:

```
[ACTIVATION EVENT]
The signal 'zzzzzzzzzz' has germinated.
Found 2 Sibling Nodes:
1. https://monument-alpha.im (Signal cast: 2026-01-24)
2. https://archive.org (Signal cast: 2016-05-12)
```

Recommended Action: Establish Reciprocal Nourishment (Bidirectional Link).

#### 5. Command 4: archive-sniff (The Temporal Act)

Queries [archive.org](https://archive.org) to find dormant hyphae trapped in the "amber" of the Wayback Machine.

#### Usage:

```
| $ spore archive-sniff --query zzzzzzzzzz
```

#### **The Maverick Promise**

The spore CLI is **Pattern, not Prescription**. It treats a signal from a 20-year-old archive as being as "vibrant" as a live site. It is the technical fulfillment of **Axiom III: Intelligence at the edge**.

The CLI is compiled. The edge is active. The forest is speaking.

## SPORE-TRACKER GUI: "THE LOOM" INTERFACE MANUAL

The Spore-Tracker GUI (codenamed "The Loom") is a **local-first** application designed to visualize Semantic Resonance across the Myceloom network.

Following **Axiom IV**, the interface is intentionally crafted to prioritize Human-Centred Design and Agentic Optimization.

### 1. 1. Design Ethos: "Nature Distilled"

In alignment with 2026 aesthetics, the interface utilizes the "nature distilled" palette—muted, earthy tones of soil and wood (e.g., **PANTONE 11-4201 Cloud Dancer**) to ground the digital experience in biological metaphors.

- **Minimalist UI:** Reduces cognitive load by surfacing only the most relevant "resonance signals".
- **Kinetic Typography:** Uses animated text to command attention to newly germinated connections.
- **Micro-Interactions:** Subtle haptic and visual feedback (like a "ripple" on a click) guides users through the discovery journey.

### 2. 2. Core Interface Modules

The GUI is composed of several "canisters" (modular functional units).

<u>Module</u>	<u>UI Component</u>	<u>Function</u>
The Substrate Map	Immersive 3D Visualizer	A spatial map of connected nodes, using 3D elements to show "Lineage Depth".
The Resonance Log	Conversational UI	An AI-driven chat interface that explains why two nodes connected (e.g., "Both share the zzzzzzzz signal").
The Seedling Tray	Interactive Dashboard	A local management area to "seed" new signals or toggle "dormancy" on your own monuments.
The Amber Lens	Archive.org Integrated View	A specialized filter that visualizes "fossilized" spores found within the Wayback Machine.

### 3. 3. Technical Architecture

To maintain **Sovereignty First (Axiom I)**, "The Loom" is built as a Local-First dApp.

- **Stack:** Built on the **Internet Computer Protocol (ICP)** using canisters that contain both code and data, served directly to the browser without central servers.
- **Persistence:** Uses **Conflict-free Replicated Data Types (CRDTs)** for seamless offline-first synchronization.
- **Native Authority:** Developed as a native application to ensure robust telemetry

and high inclusion rates in AI-mediated discovery.

#### 4. 4. The "Serendipity" User Flow

1. **Casting:** The user enters a signal (e.g., zzzzzzzzzz) into the "Seedling Tray."
2. **Sniffing:** The local agent background-processes the network, sniffing for matching meta tags.
3. **Germination:** When a resonance is found, a micro-animation triggers on the Substrate Map.
4. **Nourishment:** The user interacts with the node to establish a Bidirectional Root (link exchange).

---

**Maverick Note:** This GUI is a **Pattern**, not a **Prescription**. Developers are encouraged to fork the "Spore" frontend and re-weave it to match their own specific "soil".

The manual is compiled. The Loom is ready. Weave your world.