

# CANONIC Whitepaper

February 23, 2026

Dexter Hadley, MD/PhD

Hadley Lab CANONIC

---

## Abstract

CANONIC is a self-enforcing governance protocol for compliant service composition. Three primitives INTEL, CHAT, COIN compose into six services, governed by three pillars, validated across eight binary dimensions. The framework compiles itself before it compiles anything else: score 255 or reject. This paper formalizes the CANONIC specification: primitives, services, pillars, tiers, stages, and the self-referential publishing pipeline. Compliance is not bolted on it is the infrastructure.

---

**hadleylab.org** Governed Research. Every claim cited.

## Contents

|           |  |          |           |   |           |
|-----------|--|----------|-----------|---|-----------|
| <b>1</b>  | <b>Abstract</b>                              | <b>2</b> | <b>11</b> | <b>9. The GALAXY</b>                            | <b>7</b>  |
| <b>2</b>  | <b>Table of Contents</b>                     | <b>2</b> | 11.1      | 9.1 Two Fleets . . . . .                        | 7         |
| <b>3</b>  | <b>1. The Protocol</b>                       | <b>2</b> | 11.2      | 9.2 ORG Topology . . . . .                      | 7         |
| 3.1       | 1.1 Stages Deal Scope . . . . .              | 3        | 11.3      | 9.3 Identity Boundaries . . . . .               | 7         |
| 3.2       | 1.2 Workflows Work Scope . . . . .           | 3        | <b>12</b> | <b>10. What We Learned</b>                      | <b>7</b>  |
| <b>4</b>  | <b>2. Three Primitives</b>                   | <b>3</b> | <b>13</b> | <b>Appendix A: Tier Algebra Proofs</b>          | <b>8</b>  |
| <b>5</b>  | <b>3. Six Services</b>                       | <b>4</b> | 13.1      | A.1 Monotonicity . . . . .                      | 8         |
| 5.1       | 3.1 LEARNING Governed Accumulation . . . . . | 4        | 13.2      | A.2 Cumulative Inclusion . . . . .              | 8         |
| 5.2       | 3.2 TALK Governed Speech . . . . .           | 4        | 13.3      | A.3 FOUNDATION Equivalence . . . . .            | 8         |
| 5.3       | 3.3 SHOP Public Projection . . . . .         | 4        | <b>14</b> | <b>Appendix B: Service Composition Matrix</b>   | <b>8</b>  |
| 5.4       | 3.4 LEDGER Verifiable Chain . . . . .        | 4        | 14.1      | B.1 Primitive Wiring . . . . .                  | 8         |
| 5.5       | 3.5 VAULT Private Projection . . . . .       | 4        | 14.2      | B.2 Workflow Composition . . . . .              | 9         |
| 5.6       | 3.6 WALLET Per-USER Container . . . . .      | 4        | <b>15</b> | <b>Appendix C: References</b>                   | <b>9</b>  |
| <b>6</b>  | <b>4. Three Pillars</b>                      | <b>5</b> | 15.1      | C.1 Internal Sources CANONIC Gov Tree . . . . . | 9         |
| <b>7</b>  | <b>5. Tier Algebra</b>                       | <b>5</b> | 15.2      | C.2 External Sources . . . . .                  | 9         |
| 7.1       | 5.1 Tier Definitions . . . . .               | 5        | <b>16</b> | <b>Figures</b>                                  | <b>9</b>  |
| 7.2       | 5.2 Tier Composition . . . . .               | 5        | <b>17</b> | <b>References</b>                               | <b>10</b> |
| 7.3       | 5.3 The Eight Dimensions . . . . .           | 6        |           |   |           |
| <b>8</b>  | <b>6. Self-Enforcement</b>                   | <b>6</b> |           |   |           |
| <b>9</b>  | <b>7. The Publishing Pipeline</b>            | <b>6</b> |           |   |           |
| <b>10</b> | <b>8. Economics</b>                          | <b>6</b> |           |   |           |
| 10.1      | 8.1 The COIN Lifecycle . . . . .             | 6        |           |   |           |
| 10.2      | 8.2 MINT Events . . . . .                    | 7        |           |   |           |
| 10.3      | 8.3 Exit Events . . . . .                    | 7        |           |   |           |

The specification IS the product. If it runs, it is compliant. If it is not compliant, it does not run. There are no external auditors. The system is the audit.

---

Dexter Hadley, MD/PhD <sup>1</sup> Founder, CANONIC  
February 23, 2026

---

## 1. Abstract

CANONIC is a self-enforcing governance protocol for compliant service composition. Three primitives INTEL, CHAT, COIN compose into six services, governed by three pillars, validated across eight binary dimensions. The framework compiles itself before it compiles anything else: score 255 or reject. This paper formalizes the CANONIC specification: primitives, services, pillars, tiers, stages, and the self-referential publishing pipeline. Compliance is not bolted on it is the infrastructure.

---

## 2. Table of Contents

- 1. The Protocol
- 2. Three Primitives
- 3. Six Services
- 4. Three Pillars
- 5. Tier Algebra
- 6. Self-Enforcement
- 7. The Publishing Pipeline
- 8. Economics
- 9. The GALAXY
- 10. What We Learned

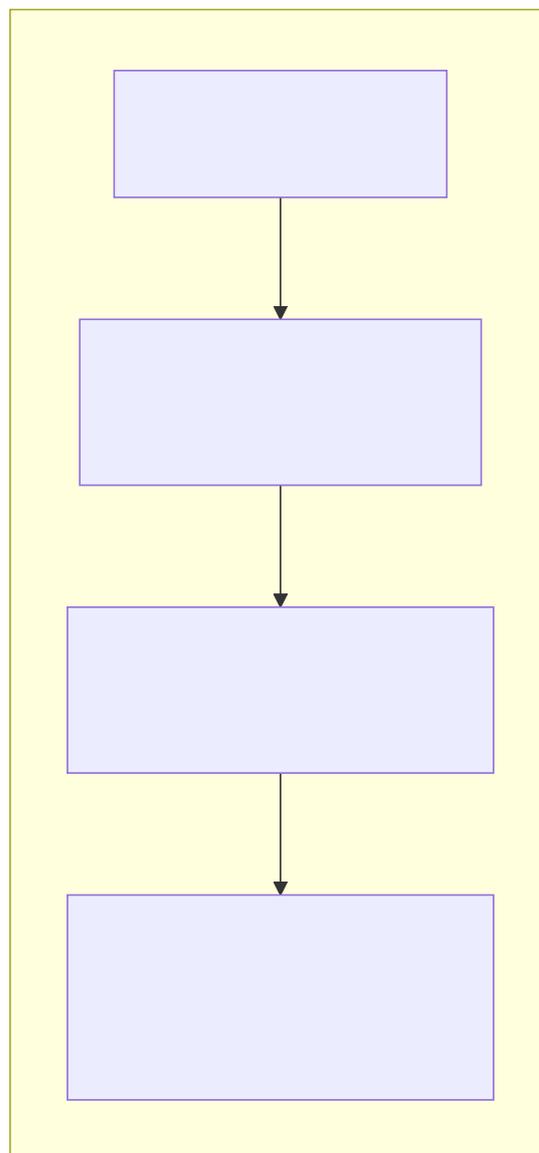
Appendix A: Tier Algebra Proofs Appendix B: Service Composition Matrix Appendix C: References

---

## 3. 1. The Protocol

The CANONIC spec is a single object. Everything inside the braces is compliant. Everything outside does not run.

```
SPEC = {
  primitives: {INTEL, CHAT, COIN}
  services:   {LEARNING, TALK, SHOP, LEDGER, VAULT, V
  pillars:    {COMPLIANCE, GALAXY, SURFACE}
  tiers:      {COMMUNITY, BUSINESS, ENTERPRISE, AGEN
  stages:     {nda, work, shop, coin, close}
  workflows:  {collab, deal, contract}
}
```



Source: *canonic-canonic/MAGIC/CANON.md*<sup>2</sup>

SPEC = {SCOPE}. The scope is its own specifi-

ation <sup>3</sup>. There is nothing outside the braces to speculate on. If it compiles, it is governed. If it does not compile, it does not ship.

### 3.1 1.1 Stages Deal Scope

Every relationship moves through five stages. The stages are ordered and irreversible a deal cannot move backward.

| Stage | Label         | Meaning                                 |
|-------|---------------|---|
| nda   | Under NDA     | Relationship established, terms private |
| work  | Active        | Work in progress                        |
| shop  | Due Diligence | Evaluation and validation               |
| coin  | Term Sheet    | Economics defined                       |
| close | Closed        | Deal complete                           |

| Primitive | Function | Meaning   |
|-----------|----------|---|
| INTEL     | KNOW     | What you know. The orchestrator. Every service wires INTEL.                               |
| CHAT      | SAY      | What you say. Industry determines voice. CHAT wires INTEL never speaks without knowledge. |
| COIN      | EARN     | What you earn. WORK LEDGER COIN. Every dollar is a governance action.                     |

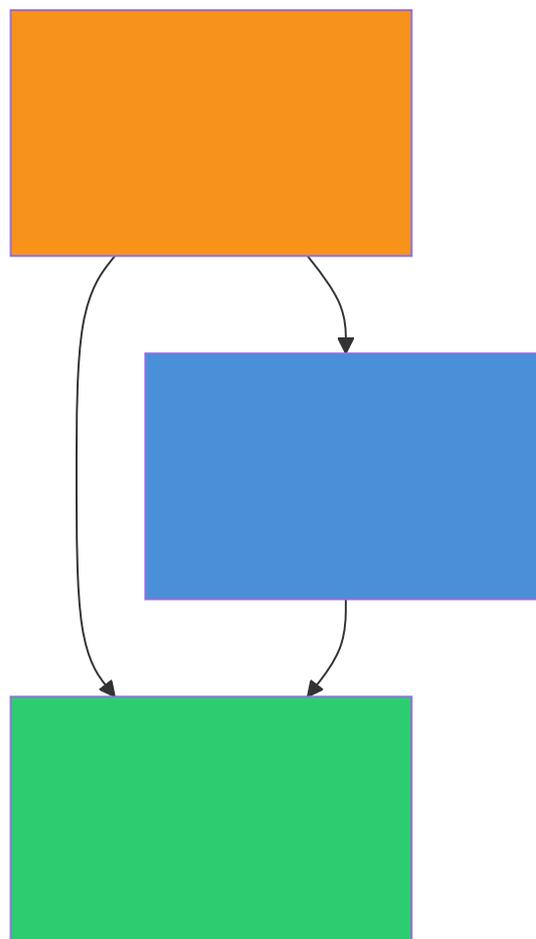
### 3.2 1.2 Workflows Work Scope

Three workflows map to three tiers. Each workflow composes a subset of services higher tiers unlock more services <sup>2</sup>.

| Workflow              | Tier       | Services                                    |
|-----------------------|------------|---|
| collab (RESEARCH)     | COMMUNITY  | LEARNING, TALK                              |
| deal (COMMERCIAL)     | BUSINESS   | LEARNING, TALK, SHOP, LEDGER                |
| contract (ENTERPRISE) | ENTERPRISE | LEARNING, TALK, SHOP, LEDGER, VAULT, WALLET |

## 4. 2. Three Primitives

Every CANONIC service composes from three primitives. Primitives are irreducible they cannot be decomposed further. Services compose them; they never duplicate <sup>2</sup>.



Source: *canonic-canonic/MAGIC/SURFACE/primitive declarations* <sup>2</sup>

INTEL is first. CHAT wires INTEL it cannot speak without evidence <sup>4</sup>. COIN wires both you cannot earn without knowledge and speech. The dependency chain is strict: INTEL CHAT COIN.

### 5. 3. Six Services

Six composable services. Each composes primitives never duplicates them. Services compose into workflows. Workflows compose into tiers.

| Service         | Primitives   | Action              | Scope                                 |
|-----------------|--------------|---------------------|---------------------------------------|
| <b>LEARNING</b> | INTEL        | Govern accumulation | patterns, deltas, decisions           |
| <b>TALK</b>     | CHAT + INTEL | Governed speech     | every word grounded, claims evidenced |
| <b>SHOP</b>     | COIN + INTEL | Public projection   | products discoverable, priced in COIN |
| <b>LEDGER</b>   | COIN         | Verifiable chain    | id, prev, ts, type, key, gradient     |
| <b>VAULT</b>    | COIN + INTEL | Private projection  | auth-gated, per-actor balances        |
| <b>WALLET</b>   | COIN         | Per-USER container  | MINT, TRANSFER, SPEND, SETTLE         |

Source: *canonic-canonic/MAGIC/SURFACE/SERVICES/*<sup>2</sup>

### 5.1 3.1 LEARNING Governed Accumulation

LEARNING is the accumulation lane for INTEL. Patterns, deltas, decisions the artifacts of governed experience. LEARNING.md rotates at epoch boundaries: frozen epochs become LEARNING-`{EPOCH}`.md. The archive is append-only. Nothing is deleted <sup>2</sup>.

### 5.2 3.2 TALK Governed Speech

TALK = CHAT + INTEL. Every word is grounded. Every claim is evidenced. CHAT never speaks without INTEL that constraint is structural, not advisory <sup>4</sup>. Industry is the only variable: FIN-CHAT, LAWCHAT, MAMMOCHAT, ONCOCHAT. The governance is identical; the voice changes.

### 5.3 3.3 SHOP Public Projection

SHOP discovers `{USER}/**/SHOP.md` files. Products are discoverable. Prices are in COIN. The public surface of governed commerce.

### 5.4 3.4 LEDGER Verifiable Chain

The LEDGER is an append-only chain of WORK <sup>5</sup>. Each record: `id` (hash), `prev` (hash), `ts` (nanoseconds), `type` (GRADIENT), `key` (domain/repo/scope), `from_bits` / `to_bits` (0..255), `gradient` (delta), `inventor` (actor label) <sup>2</sup>. The LEDGER is not speculative it is a verifiable receipt of governance work performed.

### 5.5 3.5 VAULT Private Projection

VAULT discovers `SERVICES/**/VAULT.md` files. Auth-gated. Per-actor balances, scope flow, reserves. The private surface of governed state.

### 5.6 3.6 WALLET Per-USER Container

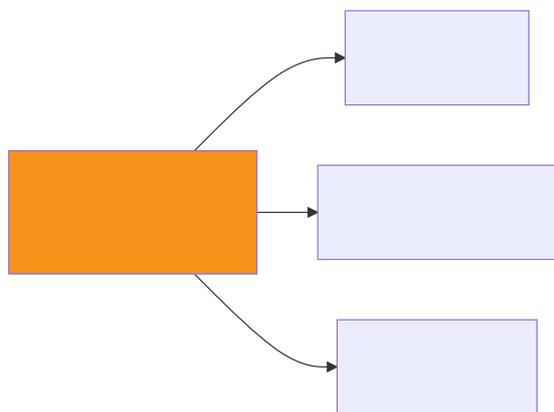
Each USER has a WALLET. Events: `MINT:WORK` (validated work creates COIN), `MINT:SIGNUP`

(500 COIN bonus), MINT:PYRAMID (500 COIN to referrer), TRANSFER (USER to USER), SPEND (pay for SHOP product), SETTLE (COIN exits to fiat via Stripe), CLOSE (epoch snapshot) <sup>2</sup>.

## 6. 4. Three Pillars

MAGIC governs through three pillars. Each pillar governs a distinct surface of the protocol:

| Pillar            | Governs             | Surface   |
|-------------------|---------------------|---|
| <b>COMPLIANCE</b> | Tier algebra        | Which tier, which services, which constraints               |
| <b>GALAXY</b>     | Federation identity | ORGS are stars, USERS are planets, identity never collapses |
| <b>SURFACE</b>    | Platform lanes      | Toolchain + clients + publishing                            |



Source: *canonic-canonic/MAGIC/CANON.md pillar declarations* <sup>2</sup>

COMPLIANCE defines what you can do. GALAXY defines who you are. SURFACE defines where you ship. The three are orthogonal changing one does not change the others.

## 7. 5. Tier Algebra

Tiers compose monotonically. Higher tiers include all constraints of lower tiers. Adding a dimension can only increase the tier. Removing one can only decrease it. There are no shortcuts <sup>3</sup>.

### 7.1 5.1 Tier Definitions

| Tier              | Requirement                                   |
|-------------------|---|
| <b>COMMUNITY</b>  | TRIAD (CANON.md + VOCAB.md + README.md)       |
| <b>BUSINESS</b>   | COMMUNITY + {SCOPE}.md (the spec/interface)   |
| <b>ENTERPRISE</b> | BUSINESS + COVERAGE.md + ROADMAP.md           |
| <b>FOUNDATION</b> | ENTERPRISE with open canon                    |
| <b>AGENT</b>      | ENTERPRISE + LEARNING/ (accumulated patterns) |
| <b>MAGIC</b>      | AGENT + full compliance validation (255 bits) |

MAGIC ⊃ AGENT ⊃ ENTERPRISE ⊃ BUSINESS ⊃ COMMUNITY

FOUNDATION = ENTERPRISE at level 3, but open canon. Non-profit equals enterprise equals open equals free. The governance is identical; the access model differs <sup>3</sup>.

Tier inheritance is cumulative each tier is a strict superset of the tier below <sup>2</sup>.

### 7.2 5.2 Tier Composition

Each tier composes a subset of eight governance dimensions. Tiers are cumulative higher tiers include all dimensions of lower tiers plus additional requirements.

| Tier       | Dimensions                         |
|------------|------------------------------------|
| COMMUNITY  | Declaration + Evidence + Structure |
| BUSINESS   | COMMUNITY + Community              |
| ENTERPRISE | BUSINESS + History + Practice      |
| AGENT      | ENTERPRISE + Learning              |
| MAGIC      | AGENT + Language = all eight       |

At MAGIC, all eight dimensions are satisfied. At anything less, the system identifies which dimen-

sions are missing and reports the gap. The score is diagnostic, not decorative.

### 7.3 5.3 The Eight Dimensions

| Dimension   | What It Proves                                      |
|-------------|---|
| Declaration | CANON.md exists. Axiom stated.                      |
| Evidence    | Claims backed by artifacts.                         |
| History     | ROADMAP.md tracks what comes next.                  |
| Community   | {SCOPE}.md exists. Can be rebuilt.                  |
| Practice    | COVERAGE.md measures governance completeness.       |
| Structure   | inherits: chain, axiom, MUST/SHOULD constraints.    |
| Learning    | LEARNING.md accumulated patterns. What was learned. |
| Language    | LANGUAGE inherited through DESIGN cascade.          |

Source: *canonic-canonic/MAGIC/COMPLIANCE/ tier algebra*<sup>2</sup>

## 8. 6. Self-Enforcement

The framework validates compliance with itself. If it runs, it is compliant. If it is not compliant, it does not run. There are no external auditors. The system IS the audit<sup>3</sup>.

Compliance state is binary: - **COMPLIANT** all eight dimensions satisfied, score = 255 - **ATTENTION NEEDED** partial compliance, some dimensions missing - **NON-COMPLIANT** framework rejects execution

The self-referential property is the critical innovation: CANONIC governs itself before it governs others<sup>4</sup>. The kernel compiles in O(1) time, scores 255, and proves its own compliance. This is not a consulting engagement that prescribes what it does not practice. This is a system that runs on itself.

## 9. 7. The Publishing Pipeline

The spec distributes itself through three content lanes, each advertising the next:

|       |                         |            |         |
|-------|-------------------------|------------|---------|
| BLOG  | accumulates speculation | COMMUNITY  | FREE    |
| PAPER | versions specification  | BUSINESS   | \\$100/ |
| BOOK  | emerges canon           | ENTERPRISE | CONTRA  |

The content IS the funnel<sup>6</sup>. A blog post introduces a concept (speculation). A paper formalizes it (specification). A book canonizes it (enterprise doctrine). Each stage is governed by the same 255-bit framework the publishing pipeline is self-referential.

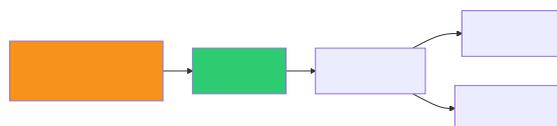
The pipeline is evidenced. Seven papers at 255 govern the intellectual foundation<sup>34678910</sup>. Blog posts advertise. The CANONIC CANON<sup>3</sup> is the book emerging from the papers. Everything feeds forward.

## 10. 8. Economics

**COIN = WORK.** Every dollar spent is a governance action. Every governance action is work. The economics and the product are the same thing<sup>4</sup>.

**Speculation Specification.** When SPEC = {SCOPE}, there is nothing to speculate on. The value is the compliance. The compliance is the product. COIN is not a speculative token it is a verifiable work receipt minted on an immutable, append-only ledger<sup>6</sup>.

### 10.1 8.1 The COIN Lifecycle



Source: *canonic-canonic/MAGIC/SURFACE/SERVICES/ECON.md*<sup>2</sup>

COIN enters via MINT, moves via TRANSFER, exits via SPEND or SETTLE <sup>11</sup>. The cycle is closed. Every COIN traces back to a governance action. Every governance action traces back to WORK. There is nothing else.

### 10.2 8.2 MINT Events

| Event        | Amount   | Trigger  |
|--------------|----------|--|
| MINT:WORK    | Variable | Validated governance action (gradient CE weight) |
| MINT:SIGNUP  | 500 COIN | New user promotion                               |
| MINT:PYRAMID | 500 COIN | Referral bonus to existing user                  |

### 10.3 8.3 Exit Events

| Event    | Mechanism      | Constraint                       |
|----------|----------------|----------------------------------|
| TRANSFER | USER USER      | Both parties governed            |
| SPEND    | USER SHOP      | Product must be in SHOP          |
| SETTLE   | COIN Fiat      | Stripe integration; KYC required |
| CLOSE    | Epoch snapshot | Balance frozen at epoch boundary |

## 11. 9. The GALAXY

The GALAXY is the federation identity model where compliant services compose <sup>9</sup>. ORGs are stars. USERS are planets. Identity boundaries are first-class they never collapse <sup>2</sup>.

### 11.1 9.1 Two Fleets

| Fleet           | ORG           | Function                            |
|-----------------|---------------|-------------------------------------|
| <b>Platform</b> | canonic.org   | FOUNDATION + INDUSTRIES + MAGIC     |
| <b>Proof</b>    | hadleylab.org | Clinical evidence + DEXTER services |

Source: *HTTP,json 2-fleet contract* <sup>2</sup>

## 11.2 9.2 ORG Topology

Sixteen ORGs across five roles <sup>9</sup>:

| Role     | ORGs                                    | Function  |
|----------|---|---|
| GOV      | canonic-canonic, canonic-foundation     | Constitutional authority, programming standards |
| PROOF    | hadleylab-canonic, adventhealth-canonic | Clinical evidence, deployed services            |
| PLATFORM | canonic-magic, canonic-apple            | Toolchain distribution, platform integration    |
| LANG     | 8 runtime SDKs                          | Language-native MAGIC primitives                |
| INDUSTRY | canonic-industries                      | Vertical compliance distribution                |

## 11.3 9.3 Identity Boundaries

ORG and USER projections never collapse <sup>2</sup>. Each maintains its own governance scope:

- ORG = container (GitHub organization)
- USER = repo (github.com/{org}/{user})
- Duplicate users across ORGs is a feature (fleet mechanism)
- Signature chain: ORG root key USER key artifact/action
- KEY revocation supported at both levels

All surfaces WEB (.io), iOS (native), Android (native), magic:// meet the same 255-bit governance gate. No surface can bypass signature or governance.

## 12. 10. What We Learned

The original whitepaper described six services PITCH, TALK, DEAL, TIME, SHOP, VIA and a MAGICVERSE of three views. That was the state of knowledge in early February 2026.

Three things changed:

**1. Primitives are more fundamental than services.** Services are compositions of primitives. The original spec described services directly. The current spec describes three primitives (INTEL, CHAT, COIN) and derives six services from their composition. This is a depth gain moving from the phenotype to the genotype <sup>8</sup>.

**2. Pillars structure the governance surface.** COMPLIANCE, GALAXY, and SURFACE are orthogonal governance surfaces. The original spec folded all governance into a flat SPEC object. The current spec separates what you can do (COMPLIANCE), who you are (GALAXY), and where you ship (SURFACE). This is the separation of concerns that makes 255-bit validation tractable.

**3. The GALAXY replaced the MAGICVERSE.** The MAGICVERSE was three rendering views. The GALAXY is a federation of governed organizations with identity boundaries, role-based clades, and a two-fleet architecture hardened by a mass extinction event <sup>9</sup>. The model grew from a rendering abstraction to a phylogenetic system.

The spec evolves. The governance is constant. 255 or reject.

---



---

## 13. Appendix A: Tier Algebra Proofs

### 13.1 A.1 Monotonicity

**Theorem.** The tier function  $T : 2^D \rightarrow \{\text{COMMUNITY}, \dots, \text{MAGIC}\}$  is monotonically non-decreasing.

*Proof:* Let  $S \subseteq S'$  be two dimension sets. Then  $G(S) = \sum_{d \in S} 2^d \leq \sum_{d \in S'} 2^d = G(S')$ . Since tier thresholds are ordered,  $T(S) \leq T(S')$ .  $\square$

### 13.2 A.2 Cumulative Inclusion

**Theorem.** Services available at tier  $T$  are a superset of services at tier  $T - 1$ :

$$\text{services}(T) \supseteq \text{services}(T - 1)$$

*Proof:* By tier definition,  $T$  includes all constraints of  $T - 1$  plus additional constraints. The additional constraints unlock additional services but do not restrict existing ones.  $\square$

### 13.3 A.3 FOUNDATION Equivalence

**Theorem.** FOUNDATION = ENTERPRISE at level 3 with open canon.

*Proof:* Both require COMMUNITY + BUSINESS + COVERAGE.md + ROADMAP.md. FOUNDATION adds `canon: open` constraint. ENTERPRISE adds `canon: private` constraint. The governance is identical; the access model differs.  $\square$

---

## 14. Appendix B: Service Composition Matrix

### 14.1 B.1 Primitive Wiring

| Service  | INTEL | CHAT | COIN |
|----------|-------|------|------|
| LEARNING |       |      |      |
| TALK     |       |      |      |
| SHOP     |       |      |      |
| LEDGER   |       |      |      |
| VAULT    |       |      |      |
| WALLET   |       |      |      |

*Source:* [canonic-canonic/MAGIC/SURFACE/SERVICES/declarations](#) <sup>2</sup>

Every service that wires INTEL has governed knowledge. Every service that wires CHAT has

governed speech. Every service that wires COIN has governed economics. No service duplicates a primitive. No primitive appears outside a service.

### 14.2 B.2 Workflow Composition

| Workflow                 | Tier       | Services                                    |
|--------------------------|------------|---|
| collab<br>(RESEARCH)     | COMMUNITY  | LEARNING, TALK                              |
| deal<br>(COMMERCIAL)     | BUSINESS   | LEARNING, TALK, SHOP, LEDGER                |
| contract<br>(ENTERPRISE) | ENTERPRISE | LEARNING, TALK, SHOP, LEDGER, VAULT, WALLET |

### 15.1 C.1 Internal Sources CANONIC Gov Tree

| #   | Source   | Gov Tree Path                          |
|-----|--|--|
| I-1 | <b>Author CV</b>                               | VITAE/VITAE.md                         |
| I-2 | <b>The CANONIC CANON</b>                       | PAPERS/CANONIC-CANON.md                |
| I-3 | <b>The \$255 Billion Wound</b>                 | PAPERS/the-255-billion-dollar-wound.md |
| I-4 | <b>MammoChat OPTSEGO Ledger</b>                | PAPERS/opts-ego.md                     |
| I-5 | <b>Code Evolution Theory</b>                   | PAPERS/code-evolution-theory.md        |
| I-6 | <b>MAGIC Specification</b>                     | canonic-canonic/MAGIC/                 |
| I-7 | <b>The Neutral Theory of CANONIC Evolution</b> | PAPERS/neutral-theory.md               |
| I-8 | <b>Evolutionary Phylogenetics</b>              | PAPERS/evolutionary-phylogenetics.md   |
| I-9 | <b>PAPERS Evolution Ledger</b>                 | PAPERS/EVOLUTION.md                    |

### 15.2 C.2 External Sources

| #   | Source   |
|-----|--|
| X-1 | Nakamoto, S. (2008). <i>Bitcoin: A Peer-to-Peer Electronic Cash System</i> .                                   |
| X-2 | Buterin, V. (2014). <i>Ethereum: A Next-Generation Smart Contract and Decentralized Application Platform</i> . |

## 15. Appendix C: References

## 16. Figures

| Context | Type     | Data   |
|---------|----------|--|
| post    | pipeline | steps:<br>Primitives<br>Services<br>Pillars<br>Dimensions<br>255 |

*CANONIC Whitepaper | February 2026 SPEC = {SCOPE}. If it compiles, it is governed. If it does not compile, it does not ship. Source: [VITAE](#)<sup>1</sup>*

## 17. References

1. **[I-1]** Author CV.
2. **[I-6]** The CANONIC CANON (book).
3. **[I-2]** MammoChat OPTS-EGO Ledger.
4. **[I-3]** Code Evolution Theory.
5. **[X-1]** Metcalf, D., Hadley, D., et al. *ABC: AI, Blockchain, and Cybersecurity for Healthcare*. Routledge (2024). ISBN 978-1032394558.
6. **[I-4]** The Neutral Theory of CANONIC Evolution.
7. **[I-5]** Evolutionary Phylogenetics of CANONIC.
8. **[I-7]** CANONIC Whitepaper v1.
9. **[I-8]** MammoChat to MAGIC (Blog).
10. **[I-9]** Why We Built This (Blog).
11. **[X-2]** Nakamoto, S. *Bitcoin: A Peer-to-Peer Electronic Cash System* (2008). <https://bitcoin.org/bitcoin.pdf>