TrustMesh World Canon

Status: Canonical • Non-Negotiable

This document freezes the architectural truth for **Worlds** within the TrustMesh substrate. It exists to prevent drift, parallel models, or accidental rewrites while enabling rapid product execution.

This canon applies to **Worlds Studio**, **Culture Wallet**, and any future World-aware interface.

---

1. Prime Directive

Evolution, not rewrite.

All work proceeds by **adapting interfaces and orchestration layers**, never by replacing or re-imagining the TrustMesh substrate.

> *Issuer Studio → Worlds Studio is a persona and UX shift, not an infrastructure shift.*

---

2. What a World *Is* (and Is Not)

A World IS

A **World** is a **scoped lens** over the TrustMesh substrate that aggregates:

  • Issuer-authored content streams
  • Issuer-defined signals & recognition
  • Membership / subscription state
  • Economic artifacts (drops, passes, mints)

A World is **issuer-anchored** and **fan-subscribed**.

A World IS NOT

A World does **not**:

  • Own user identity
  • Own wallets or keys
  • Own contacts or trust bonds
  • Replace global messaging or profile systems

  Users exist outside Worlds. Worlds are subscribed to.

This distinction is foundational.

---

3. Identity & Scope Boundaries (Hard Line)

Global (Out of Scope for Worlds)

These already exist and must never be reimplemented:

- User identity & DID binding
- Contact bonding & trust graphs
- Global messaging (XMTP)
- Wallet custody & payments (TRST)

World-Scoped

Worlds may read from global identity but only **write within World scope**:

- Content authored by the issuer
- Signals issued by the issuer
- Membership events
- Economic events tied to World artifacts

---

4. WorldID Canon

- **WorldID** is a first-class primitive
- It is a stable identifier used for:
- Topic scoping
- Registry resolution
- Client subscription

WorldID is **not** a user identifier.

Users may belong to many Worlds.

---

5. Canonical Topic Taxonomy (World-Scoped)

The following topic classes are authoritative for Worlds Studio v0:

WORLD_META_TOPIC

- World creation
- Issuer identity
- Metadata updates

WORLD_CONTENT_TOPIC

> • Media inscriptions (audio / video / visual)
> • KiloScribe payloads

WORLD_SIGNAL_TOPIC

> • Issuer → fan recognition
> • Drops, unlocks, announcements

WORLD_MEMBERSHIP_TOPIC

> • Join / subscribe events
> • Leave events

> **Global TrustMesh topics remain untouched.**

---

6. Worlds Studio UI → Infra Mapping

Worlds Studio Action	Existing Infra Used
Create World	Registry + WORLD_META_TOPIC
Upload Media	KiloScribe → WORLD_CONTENT_TOPIC
Issue Drop	Signal Engine → WORLD_SIGNAL_TOPIC
View Members	WORLD_MEMBERSHIP_TOPIC
Preview Fan View	Read-only World Lens

No new trust models. No parallel pipelines.

---

7. Issuer Studio Lineage

Worlds Studio is a **direct evolution** of the existing Issuer Studio.

> • Same submission pipeline
> • Same envelope patterns
> • Same HCS usage

Only the **persona, UI language, and asset types change**.

---

8. Non-Goals (Explicit)

The following are **out of scope** for Worlds Studio v0:

- User-generated content feeds
- Peer-to-peer social graphs inside Worlds
- Algorithmic discovery
- New identity or trust primitives

These are future surface layers, not launch requirements.

---

9. Enforcement Clause

Any implementation that:

- Reimplements identity
- Introduces alternate trust logic
- Creates parallel submission pipelines

violates this canon and must be halted for review.

---

10. Operating Principle

> **Culture creates gravity.**
> **Gravity creates networks.**
> **Networks unlock economics.**

Worlds Studio exists to ignite gravity — not to simulate social platforms.