

**v3.4-SYN-003 – ZK-Provenance Anchors & Trace-Free Data Validation**

Document Title	ZK-Provenance Anchors & Trace-Free Data Validation
Version	v3.4
Document ID	v3.4-SYN-003
Date	2025-03-22
Author	Take Back Your Data – Synthetic Intelligence Group
Document Type	Public / Certification / Internal

**1. Purpose & Scope**

This document defines the zero-knowledge provenance logic and trace-free validation framework for synthetic data in MaxOneOpen. It ensures that all generated data is provably synthetic, auditable, and leak-free.

**2. ZK-Provenance Anchoring Logic**

- All synthetic data must carry embedded zero-knowledge provenance anchors
- Anchors must include generation rule hash, schema signature, and entropy fingerprint
- Anchors must be auditable without revealing input trace or generation path
- Provenance logs must be cryptographically sealed, immutable and chain-verifiable

**3. Trace-Free Validation Design**

Validation Layer	Function Scope	ZK Mechanism
Schema Proof Link	Blueprint origin match	Anchor token + field seal
Randomness Seal	Entropy origin check	Noise fingerprint replay
Bias & Safety Trace	Compliance control	ZK-moderated filter anchors
Chain Validation	Fork alignment trace	Snapshot + path audit

**4. Certification Hooks**

- Forks must embed ZK-provenance anchors in all synthetic datasets
- Validation logic must detect and reject traced or mimicked real data
- Audit logs must be sealed, reproducible, and schema-aligned

**5. Certification Triggers**

- Absence of ZK-anchor or use of real data input disqualifies fork
- Failure to prove anchor integrity or audit path invalidates certification

**6. Certification Relevance**

MaxOneOpen-certified forks must provide verifiable proof that all data is synthetic and trace-free. ZK-provenance anchors are mandatory to protect data sovereignty, trust, and policy compliance at scale.