

## v3.4-LED-002 – Sovereign Transaction Layer & Fork Certification Framework

Document Title	Sovereign Transaction Layer & Fork Certification Framework
Version	v3.4
Document ID	v3.4-LED-002
Date	2025-03-22
Author	Take Back Your Data – Ledger Governance Division
Document Type	Public / Certification / Internal

### 1. Purpose & Scope

This document defines the transaction layer architecture and the ruleset for sovereign fork certification within MaxOneOpen. It ensures transaction integrity, user traceability, and runtime independence of certified forks.

### 2. Transaction Layer Structure

- Each transaction is schema-bound and role-constrained
- All payloads are hashed and timestamped locally
- Forks define their own asset types and policy domains
- Transactions must not rely on external trust anchors

### 3. Certification Framework

Component	Requirement	Audit Trigger
Fork Hash	Unique lineage signature	Mismatch or reuse
Schema Anchor	Declared contract + runtime match	Absent or mutated
Transaction Policy	Execution bound + TTL verified	Out-of-scope call
Runtime Interface	Fork-exposed + self-auditable	Missing or inaccessible

### 4. Sovereignty Enforcement Logic

- Each fork must expose root transaction domain
- Schema ownership must be cryptographically enforced
- Forks may not cross-certify without quorum validation
- Revoked forks must be blocked from ledger mutation

### 5. Certification Triggers

- Certification review must prove fork autonomy
- Cross-certification violations void the status
- Certification expiry may be triggered by schema drift

## 6. Certification Relevance

Only forks that implement sovereign transaction handling and expose a self-certifiable execution environment qualify for MaxOneOpen certification. Federated or hidden mutation logic is not permitted.