

### v3.4-STK-006 – Edge Execution & Local Runtime Container

|                |   |
|----------------|---|
| Document Title | Edge Execution & Local Runtime Container  |
| Version        | v3.4                                      |
| Document ID    | v3.4-STK-006                              |
| Date           | 2025-03-22                                |
| Author         | Take Back Your Data – Edge Computing Unit |
| Document Type  | Public / Certification / Internal         |

#### 1. Purpose & Scope

This document defines the logic, constraints, and deployment standards for executing MaxOneOpen components at the edge using local runtime containers. It ensures sovereign, low-latency inference and control without exposing data or relying on centralized infrastructure.

#### 2. Edge Container Principles

- Fully self-contained runtime units
- Stateless design with optional encrypted local state
- Ephemeral by default (no persistent logs)
- Designed for ARM, RISC-V, x86, and embedded GPU systems
- Launch-on-demand with twin logic binding

#### 3. Runtime Container Architecture

| Layer           | Function                                | Example                     |
|-----------------|---|-----------------------------|
| Bootloader      | Signature validation & environment prep | EdgeSec Init Layer          |
| System Core     | Execution sandbox & isolation           | Slim containerized OS or VM |
| Inference Logic | MetaLLM runtime & twin activation       | Runtime microservice stack  |
| Control Gateway | Quota, logging, fallback                | MaxControl local agent      |

#### 4. Deployment & Trigger Models

- Containers are deployed on edge nodes with cryptographic attestation
- Can be preloaded, hot-standby, or cold-booted
- Trigger types:
  - Manual API call
  - Autonomous twin logic trigger
  - Security event or system load trigger
- Twin states synchronized via signed control pulses

## 5. Compliance & Resource Boundaries

- Each runtime must respect token and resource limits
- Quota enforcement is local and audited
- No remote container introspection is permitted
- Execution is provable via signed runtime metadata
- Forks must declare container type and execution conditions

## 6. Certification Relevance

Edge containers and runtime environments must comply with the specifications in this document to be eligible for MaxOneOpen certification. Forks that alter execution logic, boot behavior, or sandbox configuration must submit matching runtime proofs and hashes.