

## v3.4-FND-005 – Roadmap & Version Strategy

|                |  |
|----------------|--|
| Document Title | Roadmap & Version Strategy             |
| Version        | v3.4                                   |
| Document ID    | v3.4-FND-005                           |
| Date           | 2025-03-22                             |
| Author         | Take Back Your Data – Strategic Office |
| Document Type  | Public / Certification / Internal      |

### 1. Purpose & Scope

This document defines the strategic release roadmap and versioning structure of MaxOneOpen. It ensures that all forks, deployments, and certification processes align with the official development timeline and structural milestones.

### 2. Core Versioning Logic

- `v3.x` = Full architecture including training, inference, stack and control layer
- Minor versions (.1-.9) indicate non-breaking enhancements or new certified modules
- Patch indicators (e.g. v3.4.2) reflect compatibility fixes, security updates or compliance alignment
- All versions are block-referenced (FND, STK, EXE) and individually certified

### 3. Release Roadmap

| Milestone | Target Date | Scope  | Certifiable Blocks |
|-----------|-------------|--|--------------------|
| v3.4      | 2025-04-01  | Full strategic documentation, governance, stack base | FND-001 to FND-007 |
| v3.5      | 2025-06-15  | Training stack, data layer, inference runtime        | STK-001 to STK-005 |
| v3.6      | 2025-09-30  | Security, privacy and edge control systems           | STK-006 to STK-010 |
| v3.7      | 2025-12-15  | Execution blocks, rollout, benchmarking & compliance | EXE-001 to EXE-004 |

### 4. Certification Dependencies

Each certified version of MaxOneOpen must reference a specific set of blocks (FND/STK/EXE). Forks must align with these references and may only claim compatibility for the version they fully implement. Partial forks or restructured releases are possible but must be declared and recertified independently.

## 5. Fork Strategy & Lifecycle

MaxOneOpen encourages controlled forking. Forks are tracked via:

- Unique Fork-ID (e.g. M10-DE-GOV-01)
- Version inheritance (e.g. based on v3.5, patch-level compliant)
- Certification compatibility tables
- Expiry windows for outdated block combinations (max 12 months unsupported)

## 6. Certification Relevance

This document is mandatory for version-tracking, compatibility enforcement and fork registration. No deployment or fork may claim to be MaxOneOpen compliant unless it conforms to the timeline and block references in this strategy.