

v3.4-PRT-001 – Portable Twin Framework & Hardware Abstraction Logic

Document Title	Portable Twin Framework & Hardware Abstraction Logic
Version	v3.4
Document ID	v3.4-PRT-001
Date	2025-03-22
Author	Take Back Your Data – Portability & Mobility Group
Document Type	Public / Certification / Internal

1. Purpose & Scope

This document defines the portable twin framework and hardware abstraction model within MaxOneOpen. It enables forks to operate across heterogeneous hardware environments while preserving deterministic logic and sovereign control.

2. Portability Layer Architecture

- Portable twins must execute on any standards-compliant CPU, GPU or TPU
- No driver-locked or vendor-exclusive logic is permitted
- Portability is defined per schema-locked execution pattern
- Compilation must support runtime-neutral binaries and diff-sealing

3. Hardware Abstraction Logic

Component	Abstraction Target	Certification Requirement
Runtime Interface	Instruction set + memory	Deterministic fallback
I/O Controller	Peripheral scope	Policy shield enforced
Security Unit	TPM or ZK-firmware	Crypto module parity
Driver Layer	Unified schema map	No hardcoded offset

4. Certification Hooks

- Portable twin manifests must be testable across 3+ hardware classes
- No performance or feature drift may occur across devices
- Certification requires full trace logs for fallback and exception modes

5. Certification Triggers

- Vendor-bound forks or undocumented abstraction gaps disqualify certification
- Missing fallback paths, hardware-lock or missing crypto parity is non-compliant

6. Certification Relevance

MaxOneOpen-certified forks must be fully portable and operate independent of vendor-specific hardware bindings. All logic must remain traceable, auditable and fallback-secure.