# Module XX – Operator Blueprint Capsule & Runtime Deployment Aid

Version: 4.1 | Classification: CTO Mandatory (Engineering Enablement)

Scope: Runtime scaffolding, role onboarding, capsule samples, SDK mapping

## 0. Purpose & Engineering Onboarding Goal

This module provides operational scaffolding, runtime deployment guidance, capsule schema examples and developer onboarding tools. It bridges the gap between capsule governance logic and real-world implementation by providing reusable configuration templates, role-linked capsule sets, and runtime entrypoints. It is required to lower the entry barrier for engineering teams and enable consistent deployment of MaxOneOpen v4.1 systems.

## 1. Role-Specific Capsule Blueprint Sets

Each sovereign operator role is equipped with a ready-to-apply capsule set:  
- CTO node: Trust capsule templates, audit routing maps, fork path injections  
- Governance operator: Policy agent setup capsules, enforcement mock flows  
- Audit instance: Capsule trace scanner setup, violation summary pack injection  
- Add-on maintainer: Sandbox capsule templates, manifest constraint overlays  
Each capsule template is provided in .mocap JSON schema format and can be pre-filled using SDK routines.

## 2. Deployment Start Blueprint (Local Node)

A minimal runtime node must include:  
- Capsule chain loader  
- LedgerSync connector stub  
- Policy Resolver engine or Rule stub  
- Capsule dispatcher (read/write to capsule overlays)  
Example YAML container setup and Kapsule CLI toolchain are included in this module package.

## 3. SDK References & Automation Hooks

Developers are provided with:  
- JSON schema definitions for Manifest, Audit, Fork, Replay and Trust capsules  
- TypeScript and Python SDK bindings for rule creation and capsule dispatch  
- CLI injection tester for previewing enforcement capsule results  
- Capsule signing utilities with default trust tier profiles

## 4. Capsule Onboarding API & Runtime Trigger Examples

The system supports onboarding via:  
`POST /capsule/manifest`  
`POST /capsule/audit`  
`POST /trust/register`  
Trigger examples:  
- On audit policy conflict → `Capsule Violation Capsule (CVC)` auto-emits  
- On trust tier mismatch → `Execution Halt Capsule (EHC)` broadcasted  
- On replay detection → `Replay Alert Capsule (RAC)` written to chain

## 5. Output Artifacts & Developer Track Integration

This module results in:  
- Deployable capsule onboarding profile  
- Developer-oriented capsule toolchain  
- Integration-ready schema mappings  
- Optional visualization overlay for capsule flow debugging