MaxDeploy v1.0 – Policy and Audit Integration

This document defines how MaxDeploy enforces policy compliance and integrates with audit and governance infrastructure. It explains the regulatory control logic, audit-trace connection points, and error handling behavior in case of violations.

# 1. Policy Enforcement via MaxReg

MaxDeploy relies on MaxReg to enforce regulatory and operational deployment policies. Before any deployment can proceed, the corresponding deployment manifest (`manifest.deploy.yaml`) must pass validation against registered policy profiles maintained within MaxReg.

# 2. Deployment Conditions and Evaluation Logic

Deployments are only triggered if all preconditions are met, including:

* - Valid signature chain for the deployment capsule
* - Explicit match of policy references (e.g., `policy.ref`)
* - No unresolved conflict markers or expired authority scopes

# 3. Audit Trace Generation (MaxAudit)

Once the deployment passes policy validation, MaxDeploy creates a cryptographically signed audit entry via MaxAudit. This audit entry is linked to the Immutable Audit Trail Ledger (IATL) and contains:

* - Deployment hash and timestamp
* - Trigger identity (role or operator capsule)
* - Signature of referenced manifest and anchor

# 4. Rejection and Error Path

If any validation fails, MaxDeploy will:

* - Abort the deployment and write a rejection capsule
* - Log the policy mismatch reason into the audit trail
* - Optionally notify MaxBridge if the rejection affects an upstream fork or registered treaty

# 5. Verifiability and External Trust Anchors

Every audit entry produced by MaxDeploy is exportable for external verification. Audit capsules can be forwarded to third-party compliance entities, regulators, or public certification ledgers. These entries are formatted in capsule-compatible formats (`meta.audit.json`, `audit.trace.sig`).