

Specification: Dynamic Discrepancy Vector & Autonomous Task Population Engine

Author Byline: J. Kornreich and Collaborators

Document Ref: SPEC-2608.02 · REF 5376-DYN

Classification: Autonomous Systems Architecture & Task Population Specification

Live Telemetry: <https://development.apiary.vision/dashboard/>

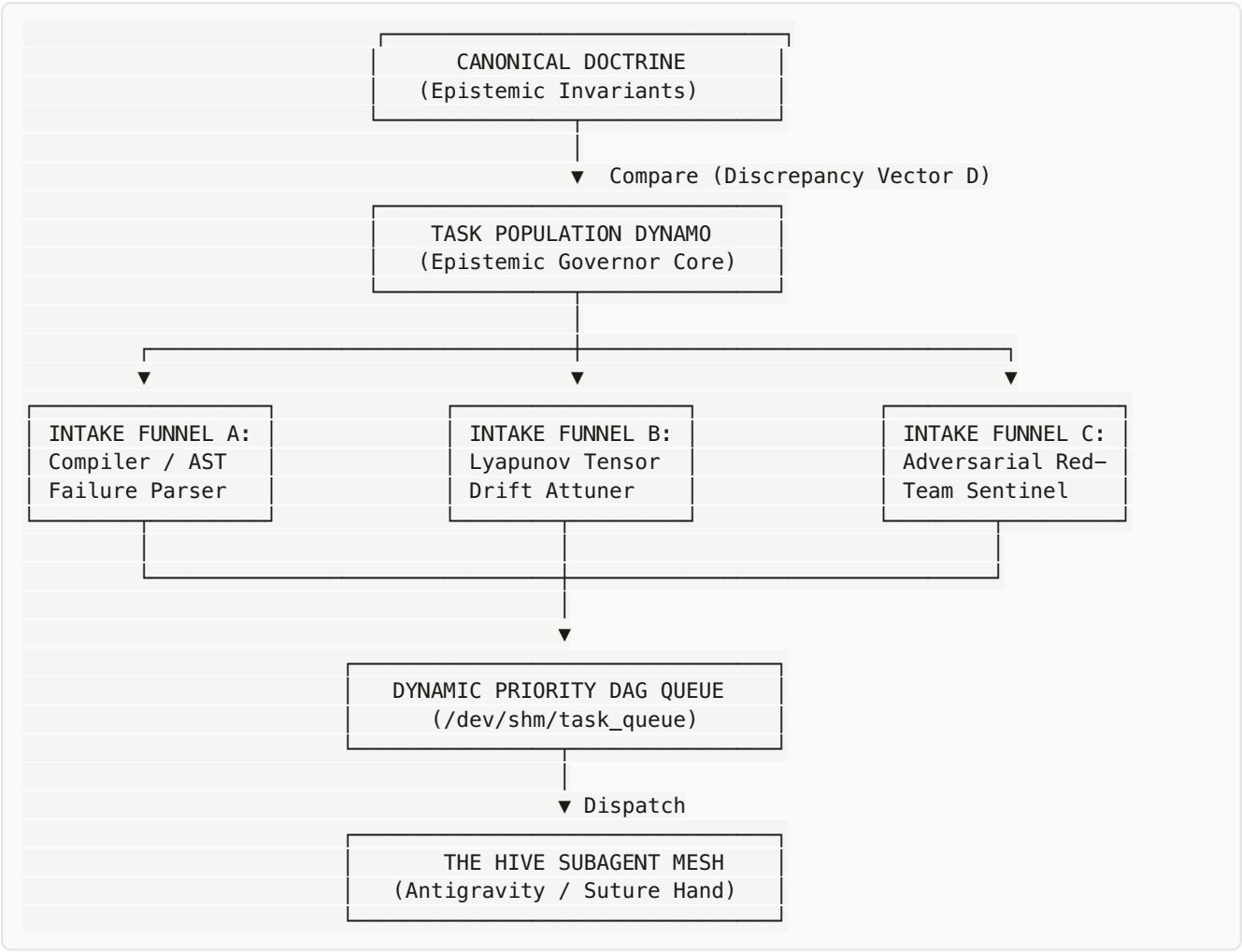
Date: August 2026

1. Executive Summary & Axiom

Autonomous agent systems operating without human supervision collapse when driven by static checklists or open-loop ReAct loops. This specification defines the **Autonomous Task Population Dynamo**, an engine that continuously spawns, prioritizes, and executes engineering tasks derived strictly from the mathematical **Discrepancy Vector** ($\vec{\mathcal{D}}$) between physical system state and canonical doctrine.

$$\vec{\mathcal{D}} = \mathcal{S}_{\text{actual}} - \mathcal{S}_{\text{doctrine}} \in \mathbb{R}^K$$

Where: * $\mathcal{S}_{\text{actual}}$ is the empirical state verified via compiler exit codes, Layer 40 tensor norms, and AST hashes. * $\mathcal{S}_{\text{doctrine}}$ is the immutable ground-truth invariant defined in the repository's canonical doctrine. * As long as $|\vec{\mathcal{D}}|^2 > \epsilon_{\text{target}}$, the Dynamo continuously spawns and executes targeted correction tasks.



2. The 4 Autonomous Intake Funnels

ID	Intake Funnel Name	Trigger Invariant	Autonomous
F-01	Compiler / AST Parser	<code>`go test` / `cargo` / `pytest` != 0</code>	Parse AST byte offset -> Fix
F-02	Lyapunov Drift Monitor	<code>Invariant Error $\Delta > 0.0150$</code>	Inject Steering Correction δ
F-03	Adversarial Sentinel	Red-Team subagent breaks doctrine	Demote task -> Defensive Fix
F-04	Shard & Repo Discovery	Git diff introduces unindexed AST nodes	Compute $\mathbb{R}^{5176}$ & Pin to VRAM

3. Dynamic Task Lifecycle State Machine

- DETECT** : An intake funnel detects a non-zero discrepancy coordinate $d_i \in \vec{\mathcal{D}}$.

2. **SPAWN** : The Dynamo generates a structured JSON task with explicit preconditions and a mathematical pass invariant.
3. **PRIORITIZE** : Tasks are inserted into the DAG ordered by priority weight $W = \text{Severity} \times |\vec{d}_{i_2}|$.
4. **DISPATCH** : Antigravity and Hive workers are assigned isolated threads to execute the mutation.
5. **AUDIT** : The Governor verifies that the resulting state achieves monotonic error reduction ($\dot{\mathcal{V}} < 0$).
6. **RATIFY & RE-EVALUATE** : The receipt is committed to `PROOF_LEDGER.md`, and the Discrepancy Vector is recomputed.