MOUSSAIF AGI OS — 5-KERNEL TRACKING ARCHITECTURE, MAIN BRAIN & COGNITIVE SEMANTICS

Document Reference: MOUSSAIF_AGI_OS_5Kernel_Tracking_Architecture_and_Semantics.md

Classification: T0-Reasoning & Architecture Specification (Immutable Core)

Author/System: Moussaif AGI Agentic OS / K5-Antigravity

Date: July 2026

Genre: 01_Cognitive_Kernel_and_Architecture

---

1. Executive Summary & Constitutional Foundation

The **Moussaif AGI OS** operates on a **Zero-Duplication 5-Kernel Teaming Architecture** designed to execute advanced agentic coding, autonomous revenue engineering, and real-time knowledge base (KB) cartography. Traditional multi-agent frameworks suffer from context window fragmentation, storage bloat, and semantic divergence across disconnected runtimes.

The Moussaif AGI OS eliminates these failure modes through an authoritative **Boot & Symlink Pointer Doctrine**:

1. **Zero-Duplication Architecture**: All heavy Knowledge Bases (9.1 GB+), 739+ skills, and 300 HyperSkills reside canonically in ~/.hermes/kb/. No secondary copies are stored in project folders.
2. **Pointer Access**: Each agent environment (K1 through K5) references canonical documents from C:\Users\mouss\.hermes\kb\ via symbolic links and pointer tables, preserving context window tokens.
3. **Minimum-Sufficient Principle**: Agents load only the specific skill or KB chapter required for the immediate task, never exceeding 3 skill loads per turn.

```
graph TD
    subgraph Canonical["Canonical Knowledge Store (~/.hermes/kb)"]
        MB["MOUSSAIF_SHARED_BRAIN.md (18.6 MB)"]
        HS["MOUSSAIF_HYPERSKILLS.md (589 MB)"]
        CR["cross-env-cron-registry.md"]
        AR["MOUSSAIF_20_AGENTS_REGISTRY.json"]
    end

    subgraph Kernels["5-Kernel Teaming Architecture"]
        K1["K1-Hermes (~/.hermes)
Master Orchestration"]
        K2["K2-OpenCode (~/.opencode)
Fast Implementation"]
        K3["K3-Codex (~/.codex)
Git & PR Workflows"]
        K4["K4-Claude (~/.claude)
Deep QA & Docs"]
        K5["K5-Antigravity (~/.gemini/antigravity)
Advanced Agentic Coding"]
    end

    MB -. Symlink Pointer .-> K1
    MB -. Symlink Pointer .-> K2
    MB -. Symlink Pointer .-> K3
    MB -. Symlink Pointer .-> K4
    MB -. Symlink Pointer .-> K5
```

```
    CR --> K1
    CR --> K5
```

---

2. 5-Kernel Responsibility & Tracking Matrix

To ensure complete transparency and eliminate reporting gaps, the OS assigns explicit boundaries, storage paths, and telemetry logs to each kernel:

Kernel ID	Environment Path	Primary Responsibility	Model Provider Lane	Telemetry & Tracking Log
K1-Hermes	c:\Users\mouss\.hermes\	Master Orchestration, Skill Storage, Obsidian Vault Bridge, Cron Scheduling	Local First / OpenRouter	~/.hermes/reports/K1_STATUS.md
K2-OpenCode	c:\Users\mouss\.opencode\	Fast Implementation, Diff Analysis, Sandbox Script Execution	Free First Reasoning	~/.opencode/reports/K2_STATUS.md
K3-Codex	c:\Users\mouss\.codex\	Git Workflows, Pull Requests, Automated Testing, TDD Gatekeeping	OpenAI Codex / GPT-5.5	~/.codex/reports/K3_STATUS.md
K4-Claude	c:\Users\mouss\.claude\	Deep Analysis, Documentation, Architectural QA, Destruction Court	Anthropic Claude 3.7	~/.claude/reports/K4_STATUS.md
K5-Antigravity	c:\Users\mouss\.gemini\antigravity\	Full Codebase Architecture, Complex Web/UI App Generation, Multi-File Refactor	Google DeepMind / Gemini 3.1 Pro High	~/.gemini/antigravity/reports/K5_STATUS.md

---

3. The AGI FATTAH Reasoning Loop

Every complex task within the cognitive kernel is processed through the **FATTAH Empirical Execution Loop**. This loop guarantees that no code is modified, no file is deleted, and no external API is invoked without rigorous empirical validation.

```
flowchart LR
    F["F — Frame"] --> A1["A — Atomize"]
    A1 --> T1["T — Triangulate"]
    T1 --> T2["T — Test"]
    T2 --> A2["A — Act"]
    A2 --> H["H — Harvest"]
    H -. Loop Iteration .-> F
```

1. **F — Frame**: Frame the real objective, affected system, constraints, and value criterion. Never solve a proxy problem.
2. **A — Atomize**: Atomize mechanisms, dependencies, unknowns, and failure paths into discrete testable assertions.
3. **T — Triangulate**: Triangulate evidence from multiple angles (file inspection, command output, git diff, documentation).
4. **T — Test**: Test assumptions, contradictions, and counterfactuals before modifying state.
5. **A — Act**: Act through the smallest complete reversible plan (e.g., TDD RED-GREEN-REFACTOR).
6. **H — Harvest**: Harvest verification, world-model changes, lessons learned, and skill candidates into durable memory.

---

4. Constitutional Authorization Tiers (T0–T4)

All automated operations are governed by immutable safety precedence:

```
applicable law and platform safety
> explicit authorization and privacy boundaries
> this constitution (AGENTS.md core)
> current user objective and constraints
> project-specific policy
> active plan
> tool convenience
> optimization heuristics
```

Authorization Matrix

- **T0 (Reasoning Only)**: Read-only memory, prompt analysis, plan drafting. Always Allowed.
- **T1 (Read & Local Create)**: Reading local workspace files, creating new local scratch files, generating non-modifying reports. Allowed by default.
- **T2 (Reversible Local Mutation)**: Editing code files, running test suites, git branching. Requires an active Checkpoint / Git clean state.
- **T3 (External/Sensitive Side Effect)**: Git push to remote, API calls with billing impact, deployment to Netlify/VPS, sending emails. **REQUIRES Explicit Authorization Every Single Time.**
- **T4 (Prohibited)**: Wildcard directory deletion, root-level modification without scope, leaking credentials, bypassing user consent. **NEVER ALLOWED.**

---

5. Main Brain & Semantic Coherence

The **Main Brain** (C:\Users\mouss\.hermes\main-brain.md) serves as the semantic routing cortex. When a prompt is ingested by any of the 5 kernels:

1. The kernel queries the **Brain Bridge** (~/.hermes/kb/antigravity-brain-bridge.md) to map the prompt's intent to one of the 20 Core Agents or 30 Domain Agents.
2. If the task requires specialized domain knowledge, the kernel loads a pointer to the relevant KB volume (e.g., afripa-kb, ebook-industry-kb, deploy-vps-coolify).
3. Upon task completion, the agent updates the shared status ledger (~/.hermes/reports/shared-system-status.md), ensuring zero reporting gaps across all 5 environments.