MOUSSAIF AGI OS

5-KERNEL TEAMING & ZERO-DUPLICATION INFRASTRUCTURE

Angle 2 Technical Reference -- Distributed Teaming, Storage & Automation

July 2026 Edition -- Canonical Documentation Series

DISTRIBUTED TEAMING & INFRASTRUCTURE SNAPSHOT

Teaming Architecture:	5-Kernel Active Topology (K1-Hermes, K2-OpenCode, K3-Codex, K4-Claude, K5-Antigravity)
Storage Governance:	Zero-Duplication Architecture | ~9.1 GB Total KB | 0 Duplicate Bytes on Host NVMe
Symlink Engine:	Hardlink & Junction Pointers across ~/.hermes, ~/.opencode, ~/.codex, ~/.claude, ~/.gemini
Local Fallback Tier:	AirLLM Tier 8.5 Local Layer Swapping (70B-405B MoE Execution on 4GB Consumer VRAM)
Provider Routing:	Dynamic Discovery + Sliding-Window Rate Limit Cooldown & Fallback Routing
Secret Management:	API Credential Vault (Plain-text Local Override + Env Variable Fallbacks)
Watchdog Automation:	5 Autonomous Cron Watchdogs (CRON-01 to CRON-05) via Any Active Agent Rule
Recovery & Rollback:	SHA-256 Checkpoint Restore & Windows Disk Reclamation Protocol

DISTRIBUTED SYSTEMS PRINCIPLES

Single Canonical Source of Truth | *Universal Pointer Symlinks* | *Zero Memory Redundancy*
Cross-Environment Handoff Without Context Loss | *Checkpoint-Gated Multi-Agent Execution*

Author: Moussa Fatah (Moussaif) | Master AGI Architect
Runtime Scope: Windows Host OS | OneDrive Sync Layer | ~/.hermes Master Pointer
Verification: SHA-256 File Hashes & Multi-Kernel Integration Tests

Chapter 0: Table of Contents & Executive Summary

Document Structure & Chapters

Chapter 1	The 5-Kernel Teaming Topology & Functional Specialization	**Page 3**
Chapter 2	Cross-Agent Handoff Protocols & Inter-Kernel Bridge	**Page 4**
Chapter 3	Zero-Duplication Storage Architecture & Symlink Governance	**Page 5**
Chapter 4	AirLLM Tier 8.5 Local Layer Swapping & VRAM Architecture	**Page 6**
Chapter 5	Dynamic Provider Discovery & Sliding-Window Rate Limiting	**Page 7**
Chapter 6	API Credential Vault & Environment Secret Synchronization	**Page 8**
Chapter 7	Autonomous Cron Scheduling, Watchdogs & Event Automation	**Page 9**
Chapter 8	System Recovery, Disk Reclamation & Checkpoint Rollback	**Page 10**

Executive Summary

This report (Angle 2 of the canonical 3-report series) provides an exhaustive technical analysis of the distributed multi-agent infrastructure, storage governance, local model swapping, and watchdog automation of MOUSSAIF AGI OS. Where conventional AI frameworks rely on monolithic single-agent loops or fragmented cli tools, MOUSSAIF AGI OS implements a cohesive 5-kernel distributed teaming topology.

The five active kernels -- K1-Hermes, K2-OpenCode, K3-Codex, K4-Claude, and K5-Antigravity -- operate as autonomous yet synchronized sub-minds. They share a single 9.1 GB Knowledge Base volume via Windows directory junctions and hardlink symlinks, eliminating duplicate bytes while ensuring every kernel sees identical system state, skills, and memory ledgers. Furthermore, the infrastructure integrates AirLLM Tier 8.5 local layer swapping, dynamic model routing, and five continuous background watchdogs.

Core Infrastructure Capabilities & Guarantees

* **Zero Duplicate Bytes:** Hardlinked symlink pointers map ~/.hermes/kb/ to all project directories and agent environments, preventing disk bloat.
* **Seamless Cross-Kernel Handoff:** Tasks move between Hermes, Codex, OpenCode, Claude, and Antigravity via explicit state snapshots and JSON-L logs.
* **4GB VRAM Large Model Execution:** AirLLM layer swapping enables running 70B-405B MoE models locally without out-of-memory crashes.
* **Autonomous Health Monitoring:** Cron watchdogs check system drift, sync state, and recover broken pipelines without user intervention.

Chapter 1: The 5-Kernel Teaming Topology & Functional Specialization

The flagship architectural advancement in MOUSSAIF AGI OS v2.0.0 is the 5-Kernel Teaming Architecture. Each runtime environment is optimized for specific engineering modalities, preventing context overload.

5-Kernel Functional Specialization Matrix

Kernel ID	Runtime Path	Primary Role	
K1-Hermes	**~/.hermes**	Master Orchestrator	KB Custodian, Memory Store, Cron Watchdog, Gateway Router (gpt-5.5-pro)
K2-OpenCode	**~/.opencode**	Fast Execution	Terminal Command Execution, High-Throughput Diff Analysis (gpt-5.5-pro)
K3-Codex	**~/.codex**	Git & PR Engine	Git Branching, PR Reviews, Automated Unit Test Generation (319 skills restored)
K4-Claude	**~/.claude**	Deep QA & Audit	Architectural Documentation, 5 Destruction Courts Quality Gates (gpt-5.5-pro)
K5-Antigravity	**~/.gemini**	App/UI Architecture	Full Codebase Generation, Multi-File Refactoring, Complex Web UI (Gemini 3.1 Pro)

Division of Labor & Operational Governance

1. K1-Hermes serves as the central brain and routing cortex (~/.hermes/main-brain.md). It receives global user requests, manages scheduled cron jobs, and coordinates multi-agent fan-out.
2. K2-OpenCode is invoked for high-speed terminal commands, diff application, and rapid file inspection.
3. K3-Codex owns version control operations, pull request pipelines, and comprehensive test suite creation.
4. K4-Claude presides over the Gap-Filling Court and Destruction Courts, auditing diffs before commit.
5. K5-Antigravity specializes in multi-file refactoring, deep architectural design, and modern Web UI generation.

Shared System Status & Cron Registry

All five kernels report to two synchronized shared state files: ~/.hermes/reports/shared-system-status.md and ~/.hermes/reports/cross-env-cron-registry.md. Any idle kernel automatically picks up pending cron jobs.

Chapter 2: Cross-Agent Handoff Protocols & Inter-Kernel Bridge

To enable seamless collaboration between K1-Hermes, K3-Codex, and K5-Antigravity, the OS implements explicit cross-agent handoff protocols (skills agentic-kb-os-014 and agentic-kb-os-015).

The Cross-Agent Handoff Specification (agentic-kb-os-014)

When an agent reaches its scope boundary (e.g., K1-Hermes completing architectural planning and handing off implementation to K3-Codex or K5-Antigravity), it executes a formal handoff sequence:

1. Checkpoint Creation: The sending agent records git HEAD, file SHA-256 checksums, and a summary of completed tasks in <appDataDir>/brain/<id>/handoff_manifest.json.
2. Context Sanitization: All conversational filler is stripped; only verified evidence, open TODO items, and exact file line ranges are included in the handoff payload.
3. Target Invocation: The receiving kernel is woken via JSON-RPC or bridge invocation, passing the handoff manifest path as its primary boot parameter.

The Hermes-Codex Unified Bridge (agentic-kb-os-015)

The Hermes-Codex bridge integrates the local codex CLI into Hermes Agent via an MCP (Model Context Protocol) server. This bidirectional tunnel allows Hermes to invoke Codex's 319 specialized development skills directly while allowing Codex to query Hermes's durable SQLite memory vault.

Subagent Fan-Out Architecture

When K5-Antigravity encounters a task requiring concurrent exploration, it calls invoke_subagent with up to 5 parallel workers (e.g., 'Codebase Researcher', 'Database Debugger', 'Security Auditor'). Each subagent runs in the background and reports back asynchronously via the reactive messaging system.

Chapter 3: Zero-Duplication Storage Architecture & Symlink Governance

A critical vulnerability in legacy multi-agent setups is disk space exhaustion caused by copying multi-gigabyte knowledge bases into every project folder. MOUSSAIF AGI OS enforces a strict Zero-Duplication Storage Architecture.

The Master Canonical Source of Truth

All heavy Knowledge Bases (20 volumes totaling ~9.1 GB), 739 base skills, 300 HyperSkills, and durable memory databases reside exclusively in C:\Users\mouss\.hermes\kb\ and C:\Users\mouss\.hermes\skills\. Zero bytes are duplicated across project subdirectories or secondary agent workspaces.

Windows Junction & Hardlink Symlink Mapping

To make canonical KBs and skills available across ~/.opencode, ~/.codex, ~/.claude, and ~/.gemini without duplicating storage, the OS uses Windows NTFS Directory Junctions and Hardlinks (skill hyper-integrity-symlink-mapper):

* Directory Junctions (mklink /J): Map folder roots such as ~/.gemini/antigravity/skills -> ~/.hermes/skills.
* File Symlinks / Hardlinks (mklink /H): Link single reference files like ~/.hermes/kb/antigravity-os-kernel.md directly into agent boot paths.
* Integrity Verification: Automated scripts run daily to verify that symlink targets are valid and that disk usage for secondary workspaces remains ~0 MB.

OneDrive Files On-Demand & Cloud Sync Governance

To safeguard the 9.1 GB KB without saturating local NVMe drives, the OS integrates OneDrive Files On-Demand (skill afripa-onedrive-backup). Infrequently accessed KB volumes are dehydrated to cloud-only stubs, automatically hydrated into local cache when an agent performs a targeted grep_search, and re-dehydrated after execution.

Chapter 4: AirLLM Tier 8.5 Local Layer Swapping & VRAM Architecture

When cloud APIs experience outages or rate limits, MOUSSAIF AGI OS activates Tier 8.5 Local Execution. Using AirLLM layer-swapping technology, the OS executes 70B and 405B MoE models locally on consumer hardware.

The AirLLM Layer Swapping Mechanism

* **1. Dynamic Model Profiling:** AirLLM analyzes the HuggingFace checkpoint (e.g., Llama-3-70B-Instruct or Mixtral-8x22B) and partitions transformer layers into atomic blocks.
* **2. NVMe-to-RAM Pre-Caching:** The model weights are stored on high-speed NVMe SSDs. Only the embeddings and immediate transformer layers are loaded into system RAM.
* **3. Sequential VRAM Paging:** During inference, individual layers are swapped sequentially into GPU VRAM (max 4GB requirement), computed, and evicted back to RAM/NVMe.
* **4. Lossless Precision:** Unlike aggressive 2-bit quantization that degrades reasoning, AirLLM maintains 16-bit or 8-bit precision by trading inference speed for memory fit.

Integration with Free-First Reasoning (agentic-kb-os-002)

Tier 8.5 is seamlessly integrated into the economical model router. For initial code drafting, exploratory grep searches, and documentation summaries, the OS dispatches prompts to local AirLLM or local Ollama (moussaif15b / moussaif4b). Expensive paid frontier models (Gemini 3.1 Pro High / GPT-5.5) are reserved strictly for complex multi-file refactoring, destructive debugging, and final sign-off gates.

VRAM Budget & System Resilience

By restricting GPU VRAM usage to <4GB per inference worker, AirLLM ensures that the host OS, desktop window manager, and background watchdogs continue operating smoothly without GPU memory exhaustion crashes.

Chapter 5: Dynamic Provider Discovery & Sliding-Window Rate Limiting

To eliminate single-point-of-failure cloud dependency, MOUSSAIF AGI OS incorporates an autonomous Dynamic Provider Discovery and Routing Engine (moussaif_smart_router.py / skill agentic-kb-os-001).

Dynamic Provider Health Probing

Rather than relying on static config files, the OS continuously probes available LLM provider endpoints (OpenRouter, Gemini API, OpenAI CLI, official Perplexity Pro API, and Local Ollama). Each provider is graded in real-time across four metrics: latency, token quota remaining, error rate, and cost per token.

The 8-Tier Cascade & Sliding-Window Rate Limit Cooldowns

Tier Level	Provider / Model Pool	Cost Economics	
Tier 1	Local Ollama / moussaif4b	Free ($0.00)	Default for simple file reading & formatting
Tier 2	Local AirLLM (70B-405B)	Free (Local VRAM)	Exploratory reasoning & local syntax check
Tier 3	Perplexity Pro (sonar-pro)	Subscription / API	Deep web research & documentation lookup
Tier 4	OpenRouter Free Pool	Free Tier API	Batch drafting & low-priority subagent tasks
Tier 5	Gemini 3.1 Flash / Lite	Low Cost	High-speed codebase search & simple diffs
Tier 6	Gemini 3.1 Pro (High)	Paid Frontier	Complex architecture, web UI, multi-file edits
Tier 7	OpenAI GPT-5.5 Pro	Paid Premium	Final QA, Destruction Court sign-off, legal/math
Tier 8.5	Emergency AirLLM / Cache	Free Fallback	Activated automatically if Tiers 3-7 rate-limit

When an HTTP 429 (Rate Limit Exceeded) is intercepted, the sliding-window rate limit controller places the provider on a 60-second cooldown and instantly reroutes pending requests to the next tier without failing.

Chapter 6: API Credential Vault & Environment Secret Synchronization

Managing credentials securely across five kernels without breaking automated scripts requires a unified secret governance architecture (skill agentic-kb-os-068 and agentic-kb-os-040).

The Plain-Text Local Override + Env Fallback Hierarchy

To eliminate configuration drift and missing API key errors, MOUSSAIF AGI OS implements a dual-layer credential resolution hierarchy across all kernels:

1. Primary Source (Local Secure Override): ~/.hermes/vault/credentials.env (or encrypted vault file) containing explicit key-value pairings.
2. Secondary Fallback (Host Environment Variables): OS-level environment variables (OPENAI_API_KEY, GEMINI_API_KEY, PERPLEXITY_API_KEY, GUMROAD_TOKEN).
3. Resolution Rule: If a key is present in the vault file, it overrides the host environment variable. If missing, the kernel gracefully falls back to os.environ without raising an unhandled KeyError.

Automated Credential Onboarding & Redaction

Under skill agentic-kb-os-068, when a user introduces a new API credential, the agent automatically stores it in the vault file and sets restricted NTFS filesystem permissions. During all log generation, PDF compilation, and git diff creation, the Security Destruction Court scans for credential regex patterns and automatically replaces literal tokens with '[REDACTED]', ensuring zero secrets leak into repositories.

Chapter 7: Autonomous Cron Scheduling, Watchdogs & Event Automation

To ensure continuous system health without requiring manual wakeup commands, MOUSSAIF AGI OS operates five autonomous background cron watchdogs (CRON-01 to CRON-05) under the 'Any Active Agent' execution doctrine.

The 5 Autonomous Cron Watchdog Registry

Cron ID	Watchdog Name	Cadence	
CRON-01	**KB & Symlink Watchdog**	0 * * * * (Hourly)	Verifies all hardlinked pointers in ~/.hermes/kb/; repairs broken junctions.
CRON-02	**Disk Reclamation Guard**	0 2 * * * (Daily)	Scans host NVMe for orphan scratch files and dehydrates idle OneDrive KB stubs.
CRON-03	**Credential Audit Loop**	0 3 * * 0 (Weekly)	Audits vault files for expiring tokens; verifies zero plaintext key leaks in logs.
CRON-04	**Opportunity & Sales Loop**	0 9 * * * (Daily)	Ingests authorized KDP/Gumroad CSV reports and updates daily revenue brief.
CRON-05	**Cross-Kernel State Sync**	*/15 * * * * (15 min)	Synchronizes shared-system-status.md and cross-env-cron-registry.md across kerne

The 'Any Active Agent' Execution Doctrine

To prevent watchdog failure if K1-Hermes is offline, all cron jobs are registered in ~/.hermes/reports/cross-env-cron-registry.md. Whenever any kernel (K2, K3, K4, or K5) completes its immediate task and enters an idle state, it reads the registry, identifies any overdue cron job, and executes the watchdog script.

Chapter 8: System Recovery, Disk Reclamation & Checkpoint Rollback

Even under rigorous quality gates, system corruption, interrupted builds, or disk saturation can occur. MOUSSAIF AGI OS integrates robust disaster recovery, automated cleanup, and rollback protocols.

Automated Windows Disk Recovery (skill system-disk-recovery)

When host free disk space drops below 5 GB, the OS automatically initiates the systematic reclamation loop:
1. Purges transient agent scratchpads in <appDataDir>/brain/*/scratch/ older than 48 hours.
2. Dehydrates local OneDrive Files On-Demand cache for non-active KB volumes.
3. Clears Python pip/npm build caches and temporary LaTeX PDF compilation files.
4. Verifies that zero canonical KB files or permanent SQLite memory vaults were harmed during cleanup.

Checkpoint Rollback Architecture

Under Law 6 ('Smallest complete reversible action'), before an agent applies multi-file edits, it records a restore script in ~/.hermes/checkpoints/. If automated tests fail after editing, the agent invokes the restore command, instantly reverting all modified files to their SHA-256 pre-mutation state.

Canonical Roadmap & Future Architectural Horizons

1. Q3 2026: Full integration of zero-latency local voice/TTS orchestration (skill agentic-kb-os-011).
2. Q4 2026: Autonomous multi-VPS deployment federation via Coolify (skill hermes-deploy-vps-coolify-010).
3. 2027 Horizon: Self-evolving HyperSkill compiler capable of synthesizing new domain OS kernels autonomously.

FINAL ARCHITECTURAL SIGN-OFF: MOUSSAIF AGI OS v2.0.0 -- ANGLE 2 CANONICAL REFERENCE

Certified by K5-Antigravity Teaming Infrastructure | Zero Duplicate Bytes | Automated Watchdogs