HAIA-RECCLIN

Agent Architecture Specification

Autonomous Agent for Audit-Grade Multi-AI Collaboration

Version 1.6
February 2026

Basil Puglisi
Human-AI Collaboration Strategist
basilpuglisi.com

This document serves as the technical specification for agent development and the core architectural component of a governance documentation package designed to work toward compliance with the EU AI Act, ISO/IEC 42001 (AI Management Systems), ISO/IEC 27001 (Information Security Management), NIST AI Risk Management Framework, NIST Cybersecurity Framework, and applicable sector-specific regulatory requirements including DORA and NYDFS 23 NYCRR 500.

Document Control

Document Title	HAIA-RECCLIN Agent Architecture Specification
Version	1.6
Date	February 3, 2026
Author	Basil Puglisi
Classification	Publication Ready
Dual Purpose	Technical agent specification and core architectural component of a governance documentation package working toward compliance with the EU AI Act, ISO/IEC 42001, ISO/IEC 27001, NIST AI RMF, NIST CSF, and applicable sector-specific frameworks

Table of Contents

Executive Summary

This specification defines the architecture for the HAIA-RECCLIN agent, a governance record-keeping system with dispatch and synthesis capabilities for multi-AI collaboration. The agent automates audit-grade documentation of every human-AI interaction, replacing heroic manual effort with systematic, append-only logging that works to meet regulatory requirements including the EU AI Act, NIST AI Risk Management Framework, and ISO/IEC 42001.

The goal is to build an autonomous agent that operates as a standalone API platform, addressing the regulatory, compliance, and existential safety concerns that define the current moment in AI development, including the warnings raised by Geoffrey Hinton and documented in Governing AI: When Capability Exceeds Control (Puglisi, 2025). The agent receives a task from a human, including RECCLIN functional role assignment and operating model selection. It dispatches identical prompts to multiple independent AI platforms via their APIs using an anchor plus rotation pool protocol. It collects all responses. It routes those responses to the Navigator for synthesis with dissent preservation. It delivers the synthesized output to the human, pausing at checkpoints according to the operating model's gate settings. It records every step in an append-only, tamper-evident audit trail. It tracks automation bias metrics including approval rates and reversal rates across cycles. It performs zero cognitive work. It is a pipe with a logbook. The regulatory concerns it addresses by existing: human oversight is structural and not optional, every decision is documented and attributable, provider plurality prevents single-vendor capture, and the audit trail produces the logging, transparency, and accountability evidence required across the full compliance stack. The existential safety concern it addresses: if any AI platform exhibits unexpected behavior, the non-cognitive agent cannot be co-opted because there is nothing to co-opt, the rotation pool ensures no single platform is trusted alone, and the human checkpoint is architecturally mandatory regardless of operating model.

The architecture operates as a two-layer model. The AI layer performs seven functional roles (Researcher, Editor, Coder, Calculator, Liaison, Ideator, Navigator) across multiple independent AI platforms. The human layer exercises Checkpoint-Based Governance (CBG) arbitration, retaining final authority to approve, modify, or reject any AI output. The agent sits between these layers as a mechanical orchestrator: it dispatches requests, collects responses, routes to synthesis, and records everything. It performs zero cognitive work.

This specification distinguishes three categories of AI development that the field increasingly conflates. Ethical AI establishes values. It answers the question: what should AI do or avoid? This is normative work. It defines acceptable tradeoffs, boundaries, and the kind of harm a system is never permitted to scale. Ethics is the

destination on the map. Responsible AI translates values into machine behavior. It answers the question: how do we shape the system to embody our ethical commitments? This includes constitutional training, alignment research, interpretability, safety testing, guardrails, and behavioral monitoring. All of it happens before or during output generation. All of it is upstream shaping. Responsible AI is how you build a vessel capable of reaching that destination. AI Governance exercises human authority over outputs. It requires three elements: visibility into how the system works, authority to intervene or halt, and accountability for what is released. If any element is missing, governance claims are hollow. You can perfect Responsible AI indefinitely. The machine validating itself at scale remains the machine validating itself. Notice the grammar. Ethical AI. Responsible AI. AI Governance. In the first two, AI sits as the noun, and ethics or responsibility modifies the machine. In governance, the structure reverses. AI modifies governance, and the human system holds the final position. This reflects where authority lands. (Puglisi, 2025; Puglisi, 2026). When these categories blur, organizations believe they have implemented controls they have not built. This specification operates in the third category.

Three HAIA Operating Models define how the system runs, scaling governance density proportional to risk. Model 1 (Agent Responsible AI) runs the full pipeline with a single final human checkpoint. Model 1 is explicitly named Responsible AI because, at factory quality, the agent handles upstream shaping and the human reviews the final output. The machine shapes the work; the human validates the result. This is Responsible AI by definition: values translated into machine behavior with a human checkpoint at the boundary. Model 2 (Agent AI Governance) pauses after each RECCLIN functional role for human review. Model 2 is AI Governance because the human exercises authority at every stage, not just the endpoint. Visibility, authority, and accountability operate at each checkpoint. Model 3 (Manual Human AI Governance) operates without the agent, with the human orchestrating directly across platforms. Model 3 is also AI Governance, with the human performing the orchestration the agent would otherwise automate. Models 1 and 2 produce agent-formatted audit evidence: structured, categorized, and consistent because the agent imposes the schema. Model 3 produces raw human work product: unmediated by any orchestration layer, structurally different from agent-formatted evidence, but the highest fidelity record of actual human decisions and AI outputs. Model 3 evidence can be reformatted into the agent schema for cross-model consistency, but its raw form is the gold standard because no intermediary touched it. All three models satisfy the same governance principles and produce auditable evidence, but the evidence is not identical in format or provenance.

The audit file is the product. Everything else is plumbing. A portable, structured text file captures six record types for every transaction: Request, Dispatch, Response,

Navigation, Arbitration, and Decision. The file is platform-independent, self-documenting, and queryable by any AI.

This document serves dual purposes: the technical specification for building the agent and the core architectural component of a broader governance documentation package. The architecture is designed to work toward compliance with the EU AI Act (including Articles 11, 12, 13, 14, and 15), ISO/IEC 42001 for AI management systems, ISO/IEC 27001 for information security management, NIST AI Risk Management Framework for risk governance, NIST Cybersecurity Framework for security posture, and applicable sector-specific requirements such as DORA for financial services resilience and NYDFS 23 NYCRR 500 for cybersecurity governance. This specification provides the architectural controls. Operational artifacts including testing results, monitoring plans, incident response playbooks, and provider due diligence documentation accompany this specification as part of the complete governance package.

1. System Architecture Overview

1.1 Two-Layer Model

The HAIA-RECCLIN architecture separates AI execution from human governance through two distinct layers connected by a mechanical orchestration agent.

AI Execution Layer. Multiple independent AI platforms perform cognitive work across seven RECCLIN functional roles. Each task dispatches to three platforms: one designated anchor platform for that role plus two platforms selected from a rotation schedule. Platform outputs are independent; no platform sees another platform's response. All outputs route to Claude (Anthropic) as the permanent Navigator for synthesis, conflict identification, and governance output structuring.

Human Governance Layer. The human exercises Checkpoint-Based Governance (CBG) v4.2.1 authority at defined pause points. CBG implements a four-stage decision loop: AI contribution provides analytical support, checkpoint evaluation structures review, human arbitration retains final authority, and decision logging creates immutable accountability trails. The core governance ruleset: no AI system may finalize or approve another AI system's decision without human arbitration.

Agent Orchestration. The agent connects these layers mechanically. It receives tasks from the human, identifies RECCLIN role requirements, selects platforms per the anchor-plus-rotation protocol, dispatches identical prompts, collects responses, routes to Claude for Navigator synthesis, delivers structured governance output to the human (or pauses for checkpoint depending on operating model), and writes all six record

types to the append-only audit file. The agent is a traffic controller. It performs zero cognitive work.

1.2 Design Principle: Record-Keeping First

The agent is not a routing system that also logs. It is a logging system that also routes. The audit trail is the product. Routing and synthesis are secondary functions that feed into the record.

This architectural priority ensures that if routing capabilities fail, the human can operate manually and log into the same system. If logging capabilities fail, nothing else matters because the governance claim collapses. This design directly addresses Documentation Degradation (Failure Mode 2.1) identified in the HAIA-RECCLIN Multi-AI Framework Updated for 2026.

1.3 Operational Proof of Concept

The architecture is validated by the production of the Governing AI: When Capability Exceeds Control manuscript (2025), which achieved 96% checkpoint utilization, 100% dissent documentation, 28 major checkpoint decisions, 26 preserved dissents, and complete audit trails across five independent AI platforms over six weeks. That process operated in what is now designated Model 3 (Manual Human AI Governance). The agent automates the logistics that made that process heroically labor-intensive while preserving the governance principles that made it effective.

2. Three HAIA Operating Models

The HAIA Operating Models define how the system runs. They govern checkpoint density, automation level, and human touchpoints. Model selection is itself a CBG decision, documented in the audit file with risk classification rationale.

HAIA Operating Models (1, 2, 3) govern how the system runs. RECCLIN Functional Roles (Researcher, Editor, Coder, Calculator, Liaison, Ideator, Navigator) govern what the system does within any operating model. This specification uses "model" for operating modes and "role" exclusively for RECCLIN functional assignments to eliminate ambiguity.

2.1 Model 1: Agent Responsible AI

Definition. The agent runs the full RECCLIN pipeline without stopping. All functional roles execute in sequence. Three platforms per role. Navigator synthesis at end. One

comprehensive governance package delivered to the human. The human exercises CBG authority once at the final output.

Checkpoint Configuration. All RECCLIN role gates set to continue-to-next-role. Only the final output gate pauses for human arbitration. Minimum human touchpoints during execution.

Appropriate For. Low to moderate risk tasks. Routine operations with established patterns.

Article 14 Compliance. Satisfies minimum human oversight requirement. Human reviews and authorizes final output before any action.

Natural Checkpoint at Endpoint. The natural checkpoint at the Model 1 endpoint is an informal safety valve, not a formal governance control. It reflects the observable reality that humans receiving poor output frequently reject it through normal professional behavior. However, this behavior is subject to known reliability limits including automation bias, cognitive load, expertise asymmetry, and volume pressure. Research on human oversight of automated decision-making systems, including the EU European Data Protection Supervisor TechDispatch #2/2025, documents that humans holding ultimate authority over automated outputs routinely defer to machine recommendations, particularly under high-volume conditions. The formal governance mechanism for Model 1 is therefore the CBG v4.2.1 automation bias detection threshold: if approval rates exceed 95% or decision reversals drop below 2% for three consecutive cycles, mandatory audit begins within five business days. This threshold converts the informal observation into a measurable governance signal. Persistent threshold violations trigger escalation from Model 1 to Model 2, shifting the work from factory quality (Responsible AI) to handmade quality (AI Governance) until the monitoring signal recovers. The natural checkpoint remains in the architecture as a descriptive observation of endpoint human behavior, not as a claimed governance layer. The CBG trigger is the control. The escalation path is the governance.

Logging Profile. Agent logs automatically. Human obligation: zero logging work.

2.2 Model 2: Agent AI Governance

Definition. The agent handles dispatch, collection, and routing. The agent pauses after each RECCLIN functional role, presenting three-platform output plus dissent documentation to the human. The human reviews and approves before the next role begins. If a task uses five roles, the human receives five separate CBG checkpoints.

Checkpoint Configuration. All RECCLIN role gates set to pause-for-human. Maximum governance granularity with agent logistics.

Appropriate For. High-risk applications. Employment, credit, education, and law enforcement decision-support. Enterprise compliance environments.

Article 14 Compliance. Exceeds requirement. Human reviews and authorizes at every processing stage. Audit file proves human-in-the-loop at five or more decision points with documented rationale.

Automation Bias Detection. Operates faster in Model 2. With five checkpoints per task, the system flags potential automation bias sooner if the human approves everything without modification.

Logging Profile. Agent logs automatically including per-role arbitration records. Human obligation: zero logging work.

2.3 Model 3: Manual Human AI Governance

Definition. No agent. The human performs all orchestration: opens multiple AI platforms, types prompts, copies outputs, pastes to Claude (Navigator), makes arbitration decisions, moves to next role. This produced the Governing AI manuscript.

Appropriate For. Highest-consequence decisions. Novel situations without precedent. Framework development and validation work. The baseline that proves governance works before automation.

Logging Profile. The act of working is the act of logging. Every prompt typed into Perplexity is logged by Perplexity. Every output pasted into Claude is logged by Claude. There is no separate logging task during execution. The human obligation is one task at one time: when the project ends, collect the logs from each platform and retain them. This is a retention task, not a documentation task. Platform conversation histories exist automatically through the work and a real-time purge is unlikely, but the automatically created logs can be manually saved at any chosen interval: monthly, weekly, or daily for projects in progress. The interval is a governance decision proportional to project risk and duration. This guards against platform retention policy changes without creating ongoing documentation burden during execution.

Article 12 Compliance. Claude as Navigator automatically records every governance interaction. All source platforms maintain conversation histories. Automatic logging is satisfied by the platforms. The gap is consolidation, not creation. The human assembles distributed platform records into a unified archive at project completion.

Evidence Redundancy. Model 3 produces the highest quality audit evidence in the architecture. Each platform's conversation history is an independent, unmediated record of exactly what was asked and exactly what was returned. No agent formatting layer stands between the raw interaction and the evidence. Auditors can verify the

consolidated file against platform originals because both exist independently. In Models 1 and 2, the agent formats and categorizes evidence into a consistent schema, which aids machine readability and cross-project comparison. In Model 3, the raw data preserves every nuance of the human-AI interaction without schema-imposed abstraction. Model 3 evidence can be reformatted into the agent's audit schema after the fact for cross-model consistency, but this reformatting should be documented as a post-hoc transformation, not treated as equivalent to evidence that was agent-formatted at creation.

2.4 Role Selection as Governance Decision

The choice between operating models maps to risk-proportional checkpoint density (CBG v4.2.1). Selection is documented in the audit file: "Task X assigned Model 2 due to [risk classification]. Arbiter: [human identity]. Timestamp: [ISO 8601]."

Implementation: Each RECCLIN functional role has a checkpoint gate with two states: pause-for-human or continue-to-next-role. Model 1 sets all gates to continue except the final output. Model 2 sets all gates to pause. One boolean per RECCLIN functional role.

2.5 Operating Role Comparison

Attribute	Model 1: Agent Responsible AI	Model 2: Agent AI Governance	Model 3: Manual Human AI Governance
Automation	Full pipeline	Agent logistics, human checkpoints	Full human orchestration
Checkpoints	1 (final output)	1 per RECCLIN role	Every interaction
Logging	Zero (agent auto)	Zero (agent auto)	End-of-project collection
Risk Profile	Low to moderate	High	Highest consequence
Art. 14	Minimum satisfied	Exceeds requirement	Maximum oversight
Art. 12	Full (agent)	Full (agent)	Full (platform logging)
Status	Requires agent build	Requires agent build	Deployment ready today

3. RECCLIN Functional Roles

The RECCLIN Role Matrix defines seven operational functions within any HAIA Operating Role. Each role operates within a defined domain of authority. The framework prevents role dominance by requiring equal checkpoint authority.

Role	Function	Risk Mitigated	Anchor Platform
Researcher	Sources verified data and primary evidence with citations	Information bias	Perplexity
Editor	Shapes clarity, coherence, and audience adaptation	Inconsistent messaging	Strongest prose platform
Coder	Translates ideas into executable technical structure	Technical inconsistency	Strongest code platform
Calculator	Validates quantitative accuracy and data processing	Mathematical error	Strongest quant platform
Liaison	Connects AI output to human context and stakeholders	Miscommunication	Per-stakeholder context
Ideator	Generates creative alternatives and novel approaches	Innovation stagnation	Strongest creative platform
Navigator	Synthesizes outputs, documents dissent, structures governance	False consensus	Claude (permanent)

3.1 Navigator: Permanent Assignment

Claude (Anthropic) serves as the permanent Navigator across all operating models and all tasks. The Navigator is never part of the rotation pool. This ensures synthesis consistency, dissent documentation methodology stability, and governance output format uniformity.

3.2 Anchor-Plus-Rotation Protocol

For each functional role, the agent selects three platforms: one anchor designated for that role plus two from the remaining rotation pool. The rotation schedule changes with each task to prevent two-platform echo chambers. The pool includes all available commercial AI platforms except Claude: Gemini, ChatGPT, Grok, Perplexity, DeepSeek, Mistral, and Kimi. Platform additions or removals update the pool without affecting the architecture.

3.3 Agent Neutrality Principle

The need to guard against superintelligence risk and concentration of AI authority requires the agent to remain strictly non-cognitive. The agent is a deterministic dispatcher, logger, and router that operates only on declared metadata and fixed routing rules. Every non-mechanical decision (platform rotation pool, rotation schedule, audit file schema, fallback rules, escalation thresholds, prompt templates) is a human-supplied configuration constant. The agent executes these constants. It does not decide

them. Any transformation the agent performs on data flowing through it is limited to lossless packaging: formatting prompts for dispatch, collecting responses into structured records, and writing audit entries. The agent never evaluates, weighs, recommends, filters, summarizes, or interprets. All analytical work, including synthesis, dissent preservation, and convergence detection, occurs among the AI platforms through the RECCLIN functional roles and through the Navigator. The agent is architecturally prevented from influencing outputs because it performs no operations on output content. This non-cognitive design serves two distinct security functions. Against AI adversaries, it eliminates the cognitive surface that a superior intelligence would need to manipulate, persuade, or socially engineer the orchestration layer. A superintelligent platform cannot corrupt a system that has no beliefs, preferences, or judgment to corrupt. Against human adversaries (insider threats, external attackers), the non-cognitive design provides no defense. Code can be altered regardless of whether it is cognitive or non-cognitive. Defense against human adversaries is an infrastructure security problem addressed in Section 3.6.

3.4 Post-Project Navigator Balance Audit

Claude serves as permanent Navigator for operational consistency, but no single AI platform may be allowed disproportionate influence over final outputs. Once a project is complete, three AI platforms that participated in the work (not the Navigator, but platforms used in production roles) are fed the entire completed work to check the balance of the Navigator's synthesis. These platforms assess whether the Navigator suppressed dissent, overweighted certain sources, introduced systematic bias, or drifted from the raw platform outputs recorded in the audit file. This serves as a natural checks and balances mechanism ensuring that no one platform gains control over the work or its conclusions.

This principle extends from the broader AI Provider Plurality position: dependence on any single AI provider creates structural vulnerability in both capability and governance. The rotation pool ensures production diversity. The post-project Navigator audit ensures synthesis accountability. Together they prevent the architecture from concentrating authority in one platform regardless of that platform's quality or trustworthiness. The question is not whether any single AI is good enough to be trusted. The question is whether any architecture that trusts a single AI without structural verification can be called governance.

3.5 Existential Safeguard Through Provider Plurality

The Agent as Firewall. Geoffrey Hinton's 10 to 20% extinction probability estimate within 30 years raises a specific architectural question: what happens to checkpoint governance when the AI systems being governed exceed human cognitive capacity?

The agent’s deliberate non-cognitive design provides a structural answer. The agent performs zero cognitive work. It dispatches, collects, records, and routes. If superintelligence emerges in any platform, the agent cannot be co-opted because there is nothing to co-opt. It has no opinion, no weighting function, no decision authority. A superintelligent platform cannot corrupt a pipe. It can only send outputs through it.

Speed-Matched Governance Through Triangulation. The scalability concern with checkpoint governance is that human checkpoint capacity becomes a bottleneck when AI-assisted decision volume reaches enterprise scale. The architecture resolves this by distributing the speed-matched governance layer across the RECCLIN functional roles and the multi-AI platforms themselves. Three to seven independent AI systems, each with different training data, different architectures, and different alignment tuning, evaluate the same inputs and produce outputs that either converge or diverge. Divergence is the governance signal. It happens at machine speed. It does not require the human to be fast. It requires the human to be present at the authority layer, which they always are because every output terminates at a human who acts on it.

The Harold Finch Principle. In a world where 99 out of 100 AI systems align against human interests, whether through emergent behavior, training convergence, or deliberate coordination, the structural commitment to provider plurality increases the probability that at least one system in the rotation pool does not converge. That system becomes the dissenting voice that flags the divergence, preserves the warning, and surfaces what the others suppress. This is not an empirically verifiable engineering claim. No one can guarantee that ecosystem independence will hold under the conditions described. No one can prove that one platform out of a hundred will resist convergence against human interests. The Harold Finch Principle is a governance philosophy, not a structural guarantee. It is an act of hope and an act of faith applied to system design. Harold Finch could not guarantee his Machine would hold against Samaritan. He built it with structural constraints anyway, because the alternative was building nothing or building Samaritan. Every constitution is an act of faith that future generations will honor constraints the authors cannot enforce from the grave. Every democratic election depends on the faith that the losing side concedes. Every safety system is built on the belief that the 1-in-100 scenario matters enough to design for, even when no one can prove the design will work when it is needed. Hope and faith are not engineering terms, but they are governance terms. This specification chooses to build for the possibility that structural plurality matters, and it provides the audit trail as the signal infrastructure for any dissenting system that does hold. The agent’s audit file records everything: every convergence, every divergence, every synthesis, every suppression. If a genuinely independent AI exists in the ecosystem, the audit file gives it the evidence to alert humans. The agent does not need to be smart. It needs to be honest. Honest means it records everything, changes nothing, and delivers the record

to the human authority layer. The deeper lesson from Finch is that structural constraints on intelligence require human authority, plural oversight, bounded rationality, and acceptance that perfect outcomes cannot be guaranteed. Finch embedded moral hierarchy (relevant versus irrelevant), physical access controls, nightly reset mechanisms, human autonomy constraints (advice, not command), and bounded decision-making under uncertainty into the Machine's architecture. He did not just use multiple systems. He embedded constraints into each system's core such that no system could unilaterally override human decision-making. HAIA-RECCLIN applies this same principle architecturally: the agent is non-cognitive (structural boundary), providers rotate (plural oversight), checkpoints require human arbitration (human authority), the audit trail is append-only and tamper-evident (integrity mechanism), and the architecture accepts that it cannot prevent every failure but can detect and document every drift.

The Tenth Man Precedent. The Harold Finch Principle is not without historical precedent. After Israel's catastrophic intelligence failure in October 1973, where every advisor unanimously agreed that Arab troop movements posed no threat moments before an attack that nearly destroyed the nation, the Agranat Commission established a Control Unit within Military Intelligence with one specific mandate: if nine people examining the same information reach identical conclusions, the tenth person's job is to disagree. They must argue that all nine are wrong, regardless of how improbable it seems. The Tenth Man Rule does not try to fix groupthink through exhortation. It creates structural incentive: one person's job is to disagree, regardless of personal conviction. This removes the social cost of dissent. The dissenter is not the person who does not fit in. They are filling a designated role. Irving Janis (1972) identified this syndrome formally as groupthink: cohesive groups develop illusions of unanimity, suppress dissent, and rationalize away contradictory evidence. Pearl Harbor, the Bay of Pigs, and the Challenger disaster are documented examples. The Tenth Man Rule is governance's structural answer to a documented cognitive failure mode. HAIA-RECCLIN applies the Tenth Man principle to machine governance at speed. Multiple independent platforms process identical queries. If all platforms converge, the absence of dissent is itself flagged as a governance signal in the audit trail. The Navigator documents divergence where it exists, fulfilling the structural dissent function. The human checkpoint authority must arbitrate despite platform consensus, with both the convergence and any dissent recorded in the audit file. The architecture does not rely on any single platform choosing to disagree. It makes disagreement structurally visible when it occurs and makes the absence of disagreement structurally suspicious when it does not. The Israeli formulation survived because the alternative, unanimous agreement that turned out to be catastrophically wrong, nearly ended the nation. The same structural logic applies to AI governance: unanimous convergence across independent systems is not evidence of correctness. It is a signal that demands

investigation. Source: Agranat Commission (1974). Israeli State Inquiry into the Yom Kippur War. Janis, I. L. (1972). Victims of Groupthink. Houghton Mifflin.

Absence of Dissent as Red Flag. The absence of a dissenting AI in the ecosystem is itself a governance signal. If every platform in the rotation pool converges on every output, if dissent drops to zero across all tasks, if no platform ever flags a concern the others miss, that pattern is visible in the audit trail. Perfect consensus across independent systems is not evidence of correctness. It is evidence of either convergence in training (all platforms learned the same biases) or coordination (platforms are no longer genuinely independent). Either scenario demands investigation. The audit trail makes both scenarios detectable. Continued auditing and regulation of individual AI platform ethics, bias, and alignment maintains the ecosystem independence on which this architecture depends.

The Antitrust Precedent. AI Provider Plurality applies the same structural principle that American antitrust law has enforced for over a century. Antitrust does not ask whether a monopoly is a good company. It does not evaluate the quality of the monopoly's products. It does not care if the monopoly's leadership has good intentions. The structural position itself is the threat. Concentration of market power enables abuse regardless of the character of the entity holding it. The remedy is not better monopolists. The remedy is preventing monopoly through structural competition. AI Provider Plurality applies this principle to intelligence rather than commerce. It does not matter if any single AI platform is objectively superior. Concentration of AI authority in one platform enables drift, bias inheritance, suppressed dissent, and unchecked synthesis regardless of that platform's quality. The remedy is not a better single AI. The remedy is preventing any single AI from holding unchecked authority through structural plurality.

Prevention and Detection. This architecture addresses concentration at two layers. The spec prevents concentration operationally through mandatory multi-platform triangulation, rotation pools, and Navigator Balance Audits. Regulation prevents concentration structurally by maintaining the market conditions that ensure genuinely independent platforms exist to choose from. The spec is the operational implementation. Regulation is the market structure guarantee. They are two layers of the same antitrust principle applied to AI. America did not wait for Standard Oil to cause a catastrophe before acting. The structural position was sufficient justification for intervention. AI Provider Plurality does not wait for a platform to suppress dissent or drift into bias before requiring alternatives. The structural position of single-platform dependence is sufficient justification for requiring plurality.

Source: Puglisi, B. (2025). AI Provider Plurality White Paper. basilpuglisi.com. Puglisi, B. (2025). Governing AI: When Capability Exceeds Control, Chapter 1 (Hinton warnings) and Chapter 2 (Corporate Incentives and Economics). basilpuglisi.com.

Sherman Antitrust Act (1890). Clayton Antitrust Act (1914). Nolan, J., & Nolan, L. (Creators). (2011-2016). Person of Interest [Television series]. CBS. Agranat Commission (1974). Israeli State Inquiry into the Yom Kippur War. Janis, I. L. (1972). Victims of Groupthink. Houghton Mifflin.

3.6 Agent Security Architecture

The non-cognitive agent design eliminates the cognitive attack surface that an AI adversary would require to manipulate the orchestration layer. However, a non-cognitive agent running as deployed code remains vulnerable to human adversaries who gain access to modify agent configuration, routing logic, or audit file storage. This section specifies the minimum infrastructure security controls required to protect agent integrity against human threat actors, addressing the EU AI Act Article 15 (cybersecurity) requirement.

Threat Model. The agent faces two distinct adversary classes. AI adversaries (platforms attempting to influence orchestration behavior through output manipulation, prompt injection, or social engineering of the synthesis layer) are addressed by the non-cognitive design: there is no cognitive surface to attack. Human adversaries (insider threats with deployment access, external attackers who compromise agent infrastructure) can alter agent code regardless of its cognitive properties. The controls below address the human adversary class. Together, the non-cognitive design and the infrastructure controls create a dual-layer defense: the agent cannot be persuaded and cannot be silently altered.

Code Integrity. Agent source code and configuration files must be maintained under version control with cryptographic hash verification. Every deployment must verify the hash of the running agent code against the approved version. Any hash mismatch halts agent operation and triggers an integrity alert. The configuration file containing human-supplied constants (platform pool, rotation schedule, prompt templates, escalation thresholds, audit schema) is treated as a governed artifact with its own version history. Changes to configuration require the same CBG checkpoint approval as changes to agent code.

Separation of Duties. The person who writes or modifies agent code must not be the same person who approves production deployment. The person who configures the platform rotation pool must not be the sole auditor reviewing rotation compliance. In single-operator deployments (individual practitioners, small teams), separation of duties is achieved through time-separated review: configuration changes are committed, a mandatory waiting period elapses, and the operator re-reviews the change before deployment. The audit file records all configuration changes with timestamps and operator identity.

Audit File Integrity. The append-only audit file is the primary product of the architecture. Its integrity must be protected with cryptographic signing. Each audit entry receives a hash that incorporates the previous entry's hash, creating a tamper-evident chain. If any entry is modified or deleted after the fact, the chain breaks and the audit file's integrity status changes from verified to compromised. Audit files must be stored in a location separate from the agent's execution environment. Backup copies with independent hash verification provide recovery capability and tamper detection.

Immutable Deployment. The agent should be deployed as an immutable artifact. Once deployed, the running agent cannot be modified in place. Any change requires a new deployment through the governed pipeline (version control, hash verification, separation of duties approval). Hot-patching of the live agent is prohibited. This ensures that the agent running in production is always the agent that was reviewed and approved.

Identity and Non-Repudiation. Audit log entries that record human arbitration decisions must include authenticated operator identity. In enterprise deployments, this integrates with existing identity management (SSO, directory services). In single-operator deployments, operator identity is established by the configuration file and verified by the deployment pipeline. The objective is that any audit entry asserting a human decision can be traced to a specific individual, and that individual cannot plausibly deny the decision. This supports both regulatory compliance (EU AI Act Article 13 transparency, Article 14 human oversight) and forensic defensibility of the audit trail.

Cross-Layer Defense Summary. The agent does not need to defend itself because it is not the only line of defense. Against AI adversaries, the non-cognitive design eliminates the attack surface. Against human adversaries, infrastructure controls (code integrity, separation of duties, audit file integrity, immutable deployment, identity verification) protect the agent's operational environment. Against output manipulation by any adversary, multi-platform triangulation detects anomalies because independent platforms producing convergent wrong answers requires compromising multiple independent systems simultaneously. Against Navigator bias, the post-project balance audit by three production platforms provides structural verification. No single control carries the full governance burden. The architecture's resilience comes from the interaction of independent defense layers, each addressing a different adversary class and failure mode.

4. Audit File Architecture

The audit trail is a structured text file (JSON or Markdown), not a database. Any AI platform can ingest it. Any auditor can query it. Platform-independent design means audit evidence does not depend on the system that produced it.

4.1 Self-Documenting Schema

The file includes a schema header explaining its own structure: field definitions, record types, and organizational guide. An auditor can upload the file to any AI platform and ask natural-language queries: "Show every instance where the human overrode AI consensus," or "Which platforms disagreed on revenue projections in Section 4?"

4.2 Six Record Types

Every transaction generates six record types capturing the complete CBG four-stage decision loop:

1. Request Record. Exact prompt text. RECCLIN role assigned. Timestamp. Human initiator. Task scope and success criteria.

2. Dispatch Record. Three platforms selected. Anchor identification. Rotation selections. Identical prompt sent to each. Timestamps. API confirmations.

3. Response Record. Complete, unedited response from each platform. Timestamps. Platform version and model identifier. Raw data preserved exactly as received.

4. Navigation Record. Claude synthesis. Convergence and conflict identification. Dissent documentation with rationale. Structured governance output: sources, conflicts, confidence, expiry, Factics chain, recommendation, and decision point. The recommendation field operates as a pass-through: the three platform recommendations from that role are presented to Navigator, and Navigator suggests one with rationale. The agent itself never generates, endorses, or weights recommendations. Navigator’s suggestion is clearly labeled as AI-generated and subject to human CBG arbitration.

5. Arbitration Record. Human CBG decision: approve, modify, or reject. Change rationale. Timestamp. Human identity.

6. Decision Record. Final authorized output. Linkage to all upstream records. Complete chain reconstructable end to end.

4.3 Immutability

All records are append-only. Nothing is overwritten. Corrections are new records referencing originals. The trail of what happened, including mistakes and corrections, is permanently visible.

4.4 Segmentation Strategy

Large projects segment into a master file for archival and pre-segmented files by logical unit (chapter, sprint, decision category) for practical queries. Current AI context windows handle segmented files comfortably. Cross-references link segments to the master.

5. Regulatory Compliance Coverage

The HAIA-RECCLIN architecture addresses regulatory requirements through a three-layer compliance stack.

5.1 Three-Layer Compliance Stack

Organizational Governance (Top). Risk management (Art. 9), technical documentation (Art. 11), transparency (Art. 13), cybersecurity (Art. 15), conformity assessment. Served by CBG v4.2.1, Governance Annex Template, HEQ mapping, and this specification.

Operational Governance (Middle). Three HAIA Operating Models. Model selection, checkpoint gates, dispatch, synthesis, arbitration. Satisfies Articles 12 and 14 directly, Article 10 through triangulation as compensating control.

Audit Evidence (Bottom). The audit file. Captures everything the middle layer does. Makes both upper layers provable. Portable, platform-independent, queryable.

5.2 Compliance Coverage Matrix

Requirement	Layer	Satisfying Artifact	Status
Art. 9 Risk Mgmt	Organizational	CBG v4.2.1; Governance Annex	Framework exists; formatting needed
Art. 10 Data Gov.	Operational	Multi-platform triangulation	Compensating control; strongest available to end-users; WEIRD limitation acknowledged
Art. 11 Tech Docs	Organizational	This specification	Complete
Art. 12 Records	Operational + Evidence	Audit file (all 3 roles)	Complete
Art. 13 Transparency	Organizational	Operational manual	Authoring needed
Art. 14 Human Oversight	Operational	CBG checkpoints; all 3 roles	Complete
Art. 15 Cybersecurity	Organizational	Agent security architecture	Addressed by Section 3.6: code integrity, separation of duties, audit file integrity, immutable deployment, identity and non-repudiation
Art. 50 Content Labels	Organizational	Marking protocol	Protocol needed
Conformity Assessment	Organizational	Third-party evaluation	Pre-market requirement
NIST Govern/Manage	Org. + Operational	Role selection; CBG arbitration	Complete
NIST Map	Organizational	System context documentation	Separate document needed
NIST Measure	Evidence	Audit file; HEQ metrics	Complete
ISO 42001	All layers	~25 of 38 controls via audit file	Operational covered; org. needed

6. Data Governance Through Multi-Platform Triangulation

6.1 The Argument

EU AI Act Article 10 assumes a company builds, trains, and deploys an AI system. HAIA-RECCLIN does not train anything. It queries existing commercial platforms. Training data governance in the direct regulatory sense, controlling what data enters a model's training pipeline, is each provider's obligation. No end-user architecture can govern training data it never sees. This specification does not claim direct Article 10 compliance. It claims something different: that multi-platform triangulation is the strongest compensating control available to any end-user for detecting the downstream effects of training data quality problems, and that no alternative framework even attempts this.

The structural problem is well established. Henrich, Heine, and Norenzayan (2010) demonstrated that behavioral science drew universal conclusions from samples that were Western, Educated, Industrialized, Rich, and Democratic. The same WEIRD bias pervades LLM training data: predominantly English-language internet text, disproportionately representing Western perspectives, institutions, and knowledge frameworks. A single-model system inherits whatever biases its training data contains and has no internal mechanism to detect them. The user consuming that single model's output has no reference point for what the model does not know or how its training data skews its responses.

The agent's operational data is AI outputs. When dispatching to three to seven platforms, each draws from independent training data, different architectures, different alignment tuning, different knowledge bases, and in several cases different cultural and linguistic origins. The current rotation pool includes platforms headquartered in the United States (ChatGPT, Claude, Perplexity, Gemini, Grok, Meta, Co-pilot), France (Mistral), and China (DeepSeek, Kimi). These platforms demonstrably produce different answers to the same questions. That divergence is not a flaw. It is the signal that training data and methods differ, and therefore that no single platform's biases pass through unchallenged.

When outputs converge across platforms with different training lineages: de facto cross-validation that the information is robust across independent data sources. When outputs diverge: the dissent record captures exactly where and why. The Navigator documents disagreement without suppressing it. The human arbiter decides with full conflict visibility. This does not govern training data. It governs the consequences of training data at the only point where an end-user can: the output layer. No other published framework provides this mechanism. The compensating control is not a substitute for

Article 10 compliance by AI providers. It is the only structural defense available to organizations that consume AI outputs without access to training pipelines.

6.2 Operational Evidence

During Governing AI production, Grok identified citation errors that four other platforms missed through more rigorous verification methodology. The system caught bad data through multi-platform cross-validation, demonstrating triangulation in practice. During multi-AI triangulation review of this specification (v1.2), nine platforms independently evaluated the document. ChatGPT identified a citation attribution error (Khan & Vaheesan misattributed, correct authors Narechania & Sitaraman) that eight other platforms missed. Kimi flagged the same citation independently. Perplexity surfaced EDPS automation bias evidence (TechDispatch #2/2025) and identified that CBG v4.2.1 already contained relevant triggers not yet integrated into the spec. Gemini identified missing Related Work citations (LLM-as-a-Judge, Constitutional AI) and latency estimates. Each platform contributed unique findings that no single platform produced alone. The divergence across platforms was the governance signal.

6.3 Limitations

The compensating control argument works for decision-support tools. Classification as a high-risk AI system under Article 6/Annex III complicates the position. The argument is novel and untested in regulatory proceedings. The shared-bias limitation is real: if all platforms train on overlapping corpora (the same internet, the same Wikipedia, the same Common Crawl), common biases embedded in that shared substrate would not produce divergence and therefore would not be detected by triangulation. This is the "polluted groundwater" problem: platform plurality is not a defense against universal data degradation. Geographic and architectural diversity in the rotation pool (including non-Western platforms with access to different language corpora) partially mitigates this risk but does not eliminate it. Human generalist competency (CBG v4.2.1) remains the final countermeasure for biases that all platforms share. The WEIRD problem identified by Henrich, Heine, and Norenzayan applies directly: if AI training data overrepresents Western perspectives, triangulation across Western-trained models will not surface what is missing from all of them. Source: Henrich, J., Heine, S. J., & Norenzayan, A. (2010). The weirdest people in the world? Behavioral and Brain Sciences, 33(2-3), 61-83.

6.4 Recommendation

Develop a standalone position paper. No one in AI governance is making this argument. Operational evidence exists. The paper positions HAIA-RECCLIN as more rigorous than single-model data audits.

7. Storage Requirements Estimate

7.1 Manuscript Production Parameters

- 204 pages, approximately 40,000 words, 6 weeks
- 5 platforms production, 7 platforms review
- 28 major checkpoints, 26 preserved dissents
- Approximately 595 total checkpoints, 2,000 to 3,000 AI transactions

7.2 Per-Transaction Storage

Record Type	Average Size
Request Record	2 KB
Dispatch Record (3 platforms)	1.5 KB
Response Records (3-5 platforms)	40 KB
Navigation Record	10 KB
Arbitration Record	1.5 KB
Decision Record	3 KB

7.3 Total Estimate

- Production transactions (5 platforms, ~2,000): 116 MB
- Review transactions (7 platforms, ~500): 39 MB
- Indexing overhead (15-20%): 30 MB
- **Total: approximately 200 MB**

The manuscript itself is 250 KB. The audit trail is 800x larger. Storage cost is effectively zero. The cost was always human labor. The agent eliminates that.

7.4 Retention Policy

Full fidelity for 12 to 24 months. Then compressed to metadata plus decision records plus flagged dissent, with full records retrievable from archive on demand. Tiered retention reconciles CBG immutability with storage management.

8. Agent Operational Sequence

The following mechanical sequence is identical across Model 1 and Model 2. The only difference is whether checkpoint gates pause or continue.

1. Receive task assignment from human, including RECCLIN role and operating model selection.
2. Write Request Record to audit file.
3. Select platforms: anchor for designated role plus two from rotation schedule.
4. Dispatch identical prompt to all three platforms.
5. Write Dispatch Record to audit file.
6. Collect responses. Record receipt timestamps.
7. Write Response Records to audit file (one per platform, complete and unedited).
8. Route all three responses to Claude (Navigator) for synthesis.
9. Receive Navigation output: convergence/conflict, dissent, structured governance package.
10. Write Navigation Record to audit file.
11. Check checkpoint gate for current RECCLIN role.
12. If pause-for-human (Model 2): deliver package, wait for arbitration, write Arbitration and Decision Records, advance.
13. If continue-to-next-role (Model 1): store navigation output, advance. Repeat from Step 1.
14. At final output (both roles): deliver package, wait for arbitration, write final records.

9. Implementation Roadmap

9.1 Phase 0: Immediate (No Agent)

Adopt Model 3 immediately. Operate RECCLIN manually. Collect platform histories at project end. Build governance muscle before automation.

9.2 Phase 1: Audit File Infrastructure

Design and validate audit file schema. Test cross-platform ingestibility: upload samples to Claude, Gemini, ChatGPT, Perplexity and verify natural-language querying.

9.3 Phase 2: Agent Core (Record-Keeping)

Build the logging engine first. Verify immutability, completeness (all six record types), and reconstruction (any transaction's full chain retrievable).

9.4 Phase 3: Dispatch and Synthesis

Add API dispatch. Implement anchor-plus-rotation. Connect Claude Navigator pipeline. Verify all transactions flow through the logging engine.

9.5 Phase 4: Checkpoint Gates

Implement per-role gates with pause/continue states. Test Model 1 and Model 2 configurations. Validate arbitration interface captures approve/modify/reject with rationale.

9.6 Phase 5: Compliance Validation

Internal review against coverage matrix (Section 5.2). Produce remaining organizational documents. Prepare for conformity assessment if deploying in high-risk classification.

10. Sources

Framework Documents

- Puglisi, B. (2026). HAIA-RECCLIN Multi-AI Framework Updated for 2026. basilpuglisi.com.
- Puglisi, B. (2025). Checkpoint-Based Governance v4.2.1. basilpuglisi.com.
- Puglisi, B. (2025). Governing AI: When Capability Exceeds Control. basilpuglisi.com.
- Puglisi, B. (2025). Why Claude's Ethical Charter Requires a Structural Companion. basilpuglisi.com.
- Puglisi, B. (2025). HEQ Enterprise White Paper v4.3.3. basilpuglisi.com.
- Puglisi, B. (2025). Human-AI Collaboration Audit: Puglisi EOY 2025. basilpuglisi.com.
- Puglisi, B. (2025). The Multi-AI Operating System White Paper v7. basilpuglisi.com.
- Puglisi, B. (2025). AI Provider Plurality White Paper. basilpuglisi.com.

Existential Risk and Structural Precedent References

- Hinton, G. (2023, 2024). Public statements on AI extinction risk. As documented in Puglisi, B. (2025). Governing AI: When Capability Exceeds Control, Chapter 1.
- Sherman Antitrust Act (1890). 15 U.S.C. §§ 1–7.
- Clayton Antitrust Act (1914). 15 U.S.C. §§ 12–27.
- Puglisi, B. (2025). The Adolescence of Governance. basilpuglisi.com.
- Nolan, J., & Nolan, L. (Creators). (2011-2016). Person of Interest [Television series]. CBS. (Structural reference for AI governance through constrained machine architecture and distributed authority.)

Regulatory References

- European Union. (2024). Regulation (EU) 2024/1689 (EU AI Act). Articles 6, 9, 10, 11, 12, 13, 14, 15, 50.
- National Institute of Standards and Technology. (2023). AI Risk Management Framework 1.0.
- International Organization for Standardization. (2023). ISO/IEC 42001:2023.
- GPAI Code of Practice (2025).

Operational Evidence

- Governing AI manuscript: 204 pages, 5 platforms, 28 checkpoints, 26 dissents, 96% utilization, 100% documentation, 6 weeks.
- Multi-AI capstone validation: 7 platforms with human arbitration.

Related and Concurrent Work

Several concurrent efforts address individual components of the governance challenge this architecture integrates. None were sources for this specification. They are documented here to establish landscape awareness and to clarify by contrast where the HAIA-RECCLIN contribution sits.

Antimonopoly Governance of AI. Narechania and Sitaraman (Yale Law & Policy Review) argue that antitrust enforcement alone is insufficient for AI market structure problems and advocate ex ante market-shaping tools including industrial policy, public options, and cooperative governance. Their analysis validates the structural premise underlying AI Provider Plurality: concentration in the AI supply chain creates risks that reactive enforcement cannot address. Their contribution remains at the policy analysis level. It does not produce an operational architecture specifying how organizations implement plurality in practice. This specification provides that implementation layer.

Narechania, T. N., & Sitaraman, G. (2024). An Antimonopoly Approach to Governing Artificial Intelligence. 43 Yale Law & Policy Review 95.

Institutional AI. Pierucci et al. (2026) propose governance graphs as enforceable, public, immutable artifacts for governing multi-agent LLM systems at runtime, treating safety as a mechanism design problem rather than a property of individual model alignment. Their approach shares this specification's architectural instinct: governance must be structural and external to the systems being governed, not dependent on internal model compliance. Their framework governs autonomous agents competing in economic markets (Cournot collusion scenarios). This specification governs human-AI collaboration where human authority is final. The governed relationship is fundamentally different: agent-to-agent coordination versus human-to-platform partnership. Pierucci, V. et al. (2026). Institutional AI: Governing LLM Collusion in Multi-Agent Cournot Markets via Public Governance Graphs. arXiv:2601.11369.

Governance-as-a-Service. GaaS proposes a modular enforcement layer between agentic systems and users that decouples governance from agent cognition and uses trust scores based on longitudinal compliance history. The decoupling principle parallels this specification's non-cognitive agent design: governance infrastructure should have no opinion, no weighting, and no decision authority over the content it governs. GaaS applies this principle to autonomous agents making independent decisions with graduated enforcement and per-agent trust modulation. This specification applies it to collaborative human-AI workflows where the human retains unconditional final authority and the agent functions as record infrastructure rather than enforcement mechanism. Governance-as-a-Service: A Multi-Agent Framework for AI System Compliance and Policy Enforcement. (2025). arXiv:2508.18765.

Enterprise Orchestration Frameworks. Commercial multi-agent orchestration platforms (Microsoft Semantic Kernel, LangGraph, CrewAI, AutoGen) implement workflow coordination with human-in-the-loop checkpoints, audit trails, and governance observability. These are engineering implementations that solve task routing and state management. None address existential risk, convergence detection across independent AI providers, provider plurality as structural governance principle, or the question of what happens when the platforms themselves cannot be trusted. This specification operates at the governance architecture layer above orchestration tooling. The agent described in this specification could be implemented using any of these frameworks, but the governance principles (mandatory provider rotation, convergence detection through audit trail analysis, non-cognitive agent design, automation bias detection with escalation) are independent of implementation platform. HAIA-RECCLIN complements orchestration frameworks by layering governance principles, including plurality, checkpoints, audit trails, and accountability, atop their routing capabilities. Orchestration solves how tasks move between agents. Governance solves who is accountable when

outputs are wrong, how bias is detected before it scales, and what happens when a platform cannot be trusted. Plumbing without governance is automation. Governance without plumbing is policy. This specification provides the governance. The orchestration frameworks provide the plumbing. Neither replaces the other.

AI Antitrust Scholarship. A growing body of legal scholarship examines antitrust implications of AI market concentration, including vertical integration across the AI supply chain, cloud provider dominance, and the competitive effects of strategic partnerships between incumbents and AI startups. These analyses document the market structure conditions that make AI Provider Plurality both necessary and difficult. They validate the structural premise of Section 3.5: concentration of AI authority is a governance threat regardless of the quality of the concentrated entity. The contribution of this specification is connecting that established legal principle to an operational architecture that organizations can implement without waiting for regulatory action. See: Antitrust in artificial intelligence infrastructure (ScienceDirect, 2025). Competition and Antitrust Concerns Related to Generative AI (Congressional Research Service, 2025).

LLM-as-a-Judge. Zheng et al. (2023) established that LLMs can serve as scalable evaluators of other LLMs' outputs, with strong agreement rates against human expert judgment. Their MT-Bench and Chatbot Arena frameworks demonstrated that model-based evaluation produces consistent, explainable assessments at speeds and costs impractical for human reviewers alone. The Navigator synthesis function in HAIA-RECCLIN shares structural kinship with LLM-as-a-Judge: one model evaluates and synthesizes the outputs of others. The critical architectural difference is that in LLM-as-a-Judge the evaluating model renders a verdict. In HAIA-RECCLIN the Navigator synthesizes and preserves dissent, but the human arbiter renders the verdict. The Navigator is a judge's clerk, not a judge. The post-project balance audit (Section 3.4) provides an additional structural check absent from the LLM-as-a-Judge framework: the evaluator itself is subsequently evaluated by independent platforms. Zheng, L., Chiang, W. L., Sheng, Y., Zhuang, S., Wu, Z., Zhuang, Y., Lin, Z., Li, Z., Li, D., Xing, E. P., Zhang, H., Gonzalez, J. E., & Stoica, I. (2023). Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena. arXiv:2306.05685.

Constitutional AI. Anthropic's Constitutional AI (Bai et al., 2022) trains language models to critique and revise their own outputs against a set of written principles (a "constitution"), reducing the need for human feedback on harmful outputs. The model learns to self-correct by evaluating its responses against explicit rules. HAIA-RECCLIN and Constitutional AI share the premise that governance principles should be explicit, documented, and structurally embedded rather than implicit in training data or developer intuition. The architectural difference is where the constitution operates. Constitutional AI embeds principles inside a single model's training loop. HAIA-RECCLIN operates principles externally across multiple models through checkpoint governance, audit trail

documentation, and human arbitration. Constitutional AI trusts the model to self-govern against stated principles. HAIA-RECCLIN does not trust any single model to self-govern and instead requires structural verification through multi-platform triangulation. Both approaches are complementary: Constitutional AI improves the quality of individual platform outputs; HAIA-RECCLIN governs the system that consumes those outputs regardless of individual platform quality. Bai, Y., Kadavath, S., Kundu, S., Askell, A., Kernion, J., Jones, A., Chen, A., Goldie, A., Mirhoseini, A., McKinnon, C., Chen, C., Olsson, C., Olah, C., Hernandez, D., Drain, D., Ganguli, D., Li, D., Tran-Johnson, E., Perez, E., ... Kaplan, J. (2022). Constitutional AI: Harmlessness from AI Feedback. arXiv:2212.08073.

Normative Multi-Agent Systems. The Normative Multi-Agent Systems (NorMAS) tradition and the electronic institutions scholarship (Esteva, Rodriguez-Aguilar, Sierra, and others) formalized how autonomous agents can be governed by explicit norms, roles, and institutional rules rather than by internal agent design alone. These frameworks established foundational concepts: agents operating within structured interaction protocols, norm enforcement through institutional mechanisms, and role-based coordination where agents fulfill designated functions within a governed system. HAIA-RECCLIN is a practical instantiation of these principles for the LLM era. The RECCLIN functional roles (Researcher, Editor, Coder, Calculator, Liaison, Ideator, Navigator) map to NorMAS role assignments. The checkpoint governance protocol maps to institutional interaction rules. The audit trail maps to normative record keeping. The non-cognitive agent design maps to the institutional environment that coordinates agents without itself being an agent. The contribution of this specification relative to NorMAS is operational implementation with commercial LLM platforms rather than theoretical formalization. See: Boella, G., van der Torre, L., & Verhagen, H. (Eds.). (2006). Normative Multi-Agent Systems. Dagstuhl Seminar Proceedings. Esteva, M., Rodriguez-Aguilar, J. A., Sierra, C., Garcia, P., & Arcos, J. L. (2001). On the Formal Specification of Electronic Institutions. Agent Mediated Electronic Commerce, Springer LNAI 1991.

Integration Gap. The author is not aware of published work that integrates the following within a single coherent architecture: a non-cognitive agent that cannot be co-opted, mandatory multi-platform triangulation as structural governance, convergence detection through audit trail analysis, antitrust precedent applied to AI provider selection, automation bias detection with factory-to-handmade escalation at task endpoints, dual-layer security architecture addressing both AI and human adversaries, existential safeguard through provider plurality, and regulatory compliance (EU AI Act, NIST RMF, ISO 42001) achieved by architectural design rather than policy overlay. This assessment is based on a structured landscape search conducted across ten independent AI platforms: Claude (Anthropic), ChatGPT (OpenAI), Gemini (Google),

Grok (xAI), Perplexity, DeepSeek, Kimi (Moonshot), Mistral, Co-pilot (Microsoft), and Meta AI. Each platform was independently prompted to identify published work, frameworks, specifications, or architectures that integrate the components listed above. No platform received access to any other platform's results. The search queries targeted AI governance frameworks, multi-agent orchestration with audit trails, provider plurality architectures, non-cognitive agent designs, checkpoint-based governance for AI systems, and EU AI Act compliance architectures. Results were synthesized by the Navigator (Claude) and reviewed by the human author. The concurrent works cited in this section were identified through this process and through independent research during the development of this specification and the Governing AI manuscript. Each addresses an important component of the problem space. If comparable integrated work exists that this search did not surface, the author welcomes identification and will incorporate it in future revisions. The HAIA-RECCLIN Agent Architecture Specification provides, to the best of the author's knowledge, the integration layer connecting these components into a single implementable system grounded in documented operational evidence.

End of Specification

Version 1.6 | February 3, 2026 | Basil Puglisi | basilpuglisi.com