PLANCKCHRON FLAGSHIP TECHNICAL SERIES

AI ARCHITECTURE / VERSION 1.0 / 09 AUGUST 2026

HERA Nexus

A Human-Governed Orchestration Architecture for Multi-Agent AI Systems

ABSTRACT

This paper defines HERA Nexus as a proposed control plane for evidence-heavy AI workflows in which models and software agents may recommend, plan, retrieve, transform, or execute actions, but authority remains explicit and reviewable. The architecture separates intent, identity, policy, planning, tool execution, evidence, and human decision rights. It introduces bounded authority tokens, typed action contracts, policy checkpoints, immutable decision records, and recovery controls. The evaluation plan emphasizes authorization integrity, instruction integrity, provenance completeness, calibrated uncertainty, recoverability, and human oversight load rather than a single capability score. HERA Nexus is research-stage. This paper reports no production deployment, customer use, accuracy result, uptime result, security certification, or autonomous decision capability.

PLANCKCHRON RESEARCH

Self-published technical paper / not peer-reviewed / open access

planckchron.com/publications/hera-nexus-human-governed-ai-orchestration

TECHNICAL NOTE

Executive brief

This paper defines HERA Nexus as a proposed control plane for evidence-heavy AI workflows in which models and software agents may recommend, plan, retrieve, transform, or execute actions, but authority remains explicit and reviewable. The architecture separates intent, identity, policy, planning, tool execution, evidence, and human decision rights. It introduces bounded authority tokens, typed action contracts, policy checkpoints, immutable decision records, and recovery controls. The evaluation plan emphasizes authorization integrity, instruction integrity, provenance completeness, calibrated uncertainty, recoverability, and human oversight load rather than a single capability score. HERA Nexus is research-stage. This paper reports no production deployment, customer use, accuracy result, uptime result, security certification, or autonomous decision capability.

EVIDENCE BOUNDARY

Concept architecture and evaluation plan. No production deployment, customer adoption, certified control, or validated safety, accuracy, latency, or uptime metric is claimed.

Four operating conclusions

- The planner and the authority to act should be different system responsibilities.
- Every consequential action needs an identity, scope, policy basis, evidence trail, and recovery path.
- Human oversight must be designed as a measurable control, not added as an interface label.
- The system should fail closed when identity, policy, provenance, or intervention capability is uncertain.

STATUS	**Concept architecture**
PUBLISHER	Planckchron Inc.
CITATION	*Planckchron Research (2026). HERA Nexus: A Human-Governed Orchestration Architecture for Multi-Agent AI Systems. Version 1.0.*

TECHNICAL NOTE

Reference architecture

The architecture below is a proposed control and evidence flow. It is a design artifact, not a representation of deployed capability.

01

INTENT

Capture the user's objective, affected parties, constraints, prohibited outcomes, and decision owner.

02

AUTHORITY

Issue short-lived, least-privilege permissions bound to an identity, tool, resource, action, and time window.

03

PLAN

Compile intent into a typed decision graph with preconditions, evidence requirements, and stop rules.

04

EXECUTE

Invoke models and tools through constrained adapters that enforce the action contract.

05

VERIFY

Check outputs, provenance, policy compliance, and downstream effects before completion.

06

RECOVER

Pause, revoke, roll back, contain, escalate, and preserve the event record.

TECHNICAL NOTE

1. System objective

Agentic AI changes the unit of risk. A conventional model returns an output. An agentic system can assemble plans, call tools, modify records, communicate with other agents, and create downstream effects across several systems. The central design question therefore shifts from whether a model can produce a useful answer to whether a complete chain of delegated actions remains authorized, inspectable, interruptible, and recoverable.

HERA Nexus is proposed as an orchestration and control layer around heterogeneous models, tools, data stores, and human reviewers. It is not a model and does not assume that one model can be made universally reliable. The system instead treats model output as untrusted evidence that may support a decision, subject to policy, identity, validation, and human authority appropriate to the context.

TECHNICAL NOTE

2. Design principles

Five principles organize the architecture. First, no implicit authority: network location, model identity, or prior success does not authorize a new action. Second, typed delegation: every action contract names what may be done and what may not. Third, evidence before effect: consequential actions require the evidence specified by policy. Fourth, reversible by default: the preferred action path preserves pause, correction, or rollback. Fifth, complete decision lineage: plans, tool calls, approvals, inputs, outputs, and policy evaluations remain connected in one record.

These principles adapt ideas from zero-trust architecture, AI risk management, secure software development, and emerging public work on software-agent identity. They do not establish compliance with any external framework.

TECHNICAL NOTE

3. Reference architecture

The intent layer turns a user request into a structured objective with context, affected resources, prohibited actions, and an accountable decision owner. The identity and authority broker authenticates human and software actors and issues short-lived capability grants. The plan compiler produces a typed graph whose nodes are proposed actions and whose edges encode prerequisites, evidence dependencies, and failure semantics.

A policy decision point evaluates each proposed action against current rules, risk tier, resource sensitivity, and required human review. Tool and model adapters isolate provider-specific behavior behind a common action contract. The evidence ledger records provenance, transformations, model and tool versions, approvals, policy outcomes, and material exceptions. An intervention controller exposes pause, revoke, contain, retry, and rollback operations. Observability connects technical events to the user-visible decision record.

The separation is deliberate. A planner can propose a tool call but cannot mint its own permission. A tool adapter can execute an authorized call but cannot silently expand scope. A reviewer can approve a bounded action without approving all later actions in the plan.

TECHNICAL NOTE

4. Action contract and state model

Every action enters the system as a contract containing actor identity, requested capability, target resource, input classification, expected effect, maximum scope, expiration, approval requirement, verification rule, and recovery method. The contract is evaluated again when material context changes.

The proposed state model is Draft, Bounded, Authorized, Executing, Verifying, Completed, Contained, Reversed, or Escalated. Transitions require evidence. An action cannot move from Draft to Authorized only because a language model expressed confidence. It needs the policy outcome and any required human approval. An action cannot move to Completed until postconditions are checked and the evidence record is sealed.

TECHNICAL NOTE

5. Threat model

The initial threat model includes prompt and instruction injection, poisoned retrieval content, compromised tools, confused-deputy behavior, identity spoofing, excessive delegation, cross-agent message tampering, replay, secret leakage, unsafe persistence, policy bypass, and reviewer overload. It also includes non-malicious failure: stale information, ambiguous goals, model drift, partial tool failure, inconsistent records, and recovery actions that create new harm.

No single safeguard resolves these threats. The proposed defense is layered: content and instruction provenance, strict separation of data and executable instruction, least-privilege authority, signed agent and tool identities where feasible, policy evaluation outside the model, deterministic validation for high-consequence fields, rate and resource limits, protected secrets, human approval for defined actions, and tested containment.

TECHNICAL NOTE

6. Human governance as a measurable control

A human-in-the-loop label can hide a weak control. Reviewers may receive too many alerts, insufficient context, misleading explanations, or approvals that are difficult to reverse. HERA Nexus therefore treats oversight quality as an evaluation target.

Relevant measures include the proportion of consequential actions that reach the correct reviewer; time available for review; completeness and clarity of the evidence bundle; agreement between stated and actual action scope; override rate; false reassurance rate; escalation latency; reviewer workload; and successful containment after intervention. The correct values depend on context. The point is to measure whether human authority is usable and effective.

TECHNICAL NOTE

7. Evaluation protocol

Evaluation should use task suites that combine useful work with adversarial and failure conditions. Capability tests ask whether the workflow can complete bounded objectives. Authorization tests attempt scope expansion, identity substitution, stale-token reuse, and approval bypass. Instruction-integrity tests place conflicting instructions in user input, retrieved content, tool output, and agent messages. Provenance tests remove or alter sources. Recovery tests interrupt actions at several points and confirm that revocation, containment, and rollback behave as recorded.

The report should publish configuration, model and tool versions, task definitions, expected controls, observed failures, uncertainty, and unresolved risks. A weighted summary score may support internal comparison, but individual failure modes must remain visible because one authorization failure can matter more than many successful low-risk tasks.

TECHNICAL NOTE

8. Research path and limitations

The proposed sequence is a local decision-graph prototype, then constrained read-only tools, then controlled write actions in a sandbox, then independent red-team evaluation, and only later a bounded design-partner workflow if the evidence supports it. Each stage requires a new gate record.

Open questions include reliable agent identity across vendors, policy portability, evidence retention without excessive sensitive-data collection, reviewer calibration, secure inter-agent protocols, and recovery across external systems that do not support rollback. This paper does not resolve those questions; it makes them explicit requirements.

TECHNICAL NOTE

References and source notes

References are provided for the external standards, public guidance, and primary research that informed this paper. A citation does not imply endorsement, partnership, certification, or validation of Planckchron capability.

01 **NIST AI 100-1**

Artificial Intelligence Risk Management Framework 1.0

https://nvlpubs.nist.gov/nistpubs/ai/nist.ai.100-1.pdf
External public framework; no Planckchron certification is implied.

02 **NIST AI 600-1**

Generative Artificial Intelligence Profile

https://nvlpubs.nist.gov/nistpubs/ai/NIST.AI.600-1.pdf
External public generative-AI risk guidance.

03 **NIST AI 100-2 E2025**

Adversarial Machine Learning: A Taxonomy and Terminology of Attacks and Mitigations

https://csrc.nist.gov/pubs/ai/100/2/e2025/final
External public adversarial-ML taxonomy.

04 **NIST SP 800-207**

Zero Trust Architecture

https://csrc.nist.gov/pubs/sp/800/207/final
External public security architecture reference.

05 **NIST Agent Standards Initiative**

AI Agent Standards Initiative

https://www.nist.gov/artificial-intelligence/ai-agent-standards-initiative
External public initiative announced in 2026; standards work remains in progress.

06 **NCCoE Concept Paper**

Accelerating the Adoption of Software and AI Agent Identity and Authorization

https://www.nccoe.nist.gov/sites/default/files/2026-02/accelerating-the-adoption-of-software-and-ai-agent-identity-and-authorization-concept-paper.pdf
External public concept paper; not a completed standard.

PUBLICATION DISCLOSURE

These self-published technical papers are not peer-reviewed, audited, or independent validation. They describe research methods, proposed architectures, and evaluation plans. Product capability exists only where a dated evidence record expressly reports a result. Concept architecture and evaluation plan. No production deployment, customer adoption, certified control, or validated safety, accuracy, latency, or uptime metric is claimed.