Supervisory Primacy: Human-in-the-Loop AI Orchestration for High-Consequence Domains

The Architectural Form of Human Authority in Adaptive AI Systems

By Jorge Enrique Flores Montano

Founder, JM Automated Solutions

~MILO™ — Modular Intelligent Learning Orchestrator (patent pending)

May 2026

ORCID iD: 0009-0003-1859-8418

DOI: 10.5281/zenodo.20117662

Public reference: https://github.com/jmontano1/milo-architecture

Abstract

Human-in-the-loop (HITL) frameworks for AI systems are increasingly treated as policy-level commitments — "the human can always override" — when their operational effectiveness requires that they be architectural properties of the system itself. A policy-level HITL commitment is disabled by a configuration flag; an architectural HITL property is disabled by rebuilding from source. This paper introduces*Supervisory Primacy* as a design principle: the human-authoritative state is the architectural default for consequential actions in adaptive AI orchestration systems, with the AI proposing and the human disposing, every consequential action carrying a mandatory authorization audit trail, and the eight non-negotiable operational integrity constraints implemented as enforceable safeguards in deployment builds rather than as runtime policy. Supervisory Primacy is consistent with the human oversight requirements of EU AI Act Article 14 [1], operates within the levels-of-automation taxonomy of Parasuraman, Sheridan, and Wickens[2], and is illustrated using MILO, a patent-pending adaptive AI orchestrator[3] submitted to the U.S. Department of Energy under the Genesis Mission[4]. The contribution is at the architectural level: Supervisory Primacy is not a new HITL taxonomy; it is the structural design that makes HITL load-bearing rather than retrofittable.

Keywords: human-in-the-loop, AI oversight, supervisory control, EU AI Act, levels of automation, adaptive AI architecture, high-consequence systems.

Highlights.

- Introduces*Supervisory Primacy* as the architectural design principle by which human authority over consequential AI decisions becomes a constitutional property of the orchestration system, not a runtime configuration.
- Operationalizes the principle through four architectural mechanisms: audit-first command-and-signal substrate, reflex predicates before fanout, pre-execution gating, and voluntary side-effect pathways.
- Composes with EU AI Act Article 14, NIST AI RMF 1.0 Appendix C, the Parasuraman–Sheridan–Wickens framework, and the industrial-robotics functional-safety standards (ANSI/RIA R15.06, ISO 10218, ISO/TS 15066, IEC 61508/61511).

- Specifies a threat model for the audit substrate requiring cryptographic chain-of-custody, custodial separation, and external WORM replication for high-consequence deployment posture.

Index Terms: human-in-the-loop AI, supervisory control, AI oversight, EU AI Act, levels of automation, audit-first architecture, industrial robotics safety, functional safety, IEC 61508, ANSI/RIA R15.06, NIST AI RMF.

Plain Language Summary. When an AI system operates in a setting where mistakes have severe consequences — a power grid control room, a nuclear facility, an operating room, an autonomous robotic line — the rule that "a human can always override the AI" must be more than a promise. It must be built into the architecture, so that disabling the override would require rebuilding the system from source, not toggling a setting. This paper names that architectural property *Supervisory Primacy*: the human-authoritative state is the default for any consequential action; the AI proposes and the human disposes; every consequential action carries a mandatory audit trail by architecture, not by policy. The principle does not invent human oversight; it specifies the structural form that makes existing human-oversight regulations operationally effective.

Relevance to U.S. National Interest. Human oversight of AI in critical-infrastructure environments — energy grid operations, nuclear facility control, autonomous robotics under human-robot interface, satellite and space operations, first-responder coordination — is a foundational concern for U.S. critical-infrastructure protection. The principle articulated here provides the architectural substrate by which sector-specific oversight requirements (FDA, NHTSA, EASA, NRC, DOE) become operationally enforceable rather than promissory.

Status of claims. *Supervisory Primacy* as articulated in this paper is the author's original architectural design principle. The EU AI Act Article 14 [1], the Parasuraman–Sheridan–Wickens framework[2], NIST AI RMF 1.0 [9], and the industrial-robotics functional-safety standards (R15.06 [11], ISO 10218 [12], ISO/TS 15066 [13], IEC 61508/61511 [14]) are external references with established standing in their respective regulatory and engineering communities; this paper reads them and proposes an architectural substrate that composes with them. Conformance with any specific standard is a deployment-level assessment, not a certification claim. The audit-substrate hardening commitments specified in §4 are architectural requirements proposed for high-consequence deployment posture; specific implementations and validation against actual incident-reconstruction scenarios are forthcoming work. This manuscript is a preprint prior to peer review.

About MILO. MILO (Modular Intelligent Learning Orchestrator) is a patent-pending adaptive AI orchestration architecture for high-consequence critical-infrastructure environments. The full architectural reference — eight structural principles, eight operational integrity constraints, the unifying viability principle, and the trademark/patent status — is maintained as a single canonical document at https://github.com/jmontano1/milo-architecture (concept DOI: 10.5281/zenodo.20117025). Author contact: jmontano@jmautomated.com.

1. Introduction

The fully autonomous AI orchestration system is structurally unsafe for industrial control, medical decision support, energy grid operations, and other high-consequence domains — not because the AI is unreliable in any individual decision, but because the escalation pathway and the traceability of decision authority cannot be reconstructed forensically when the system is fully autonomous. Every incident review of a sociotechnical failure depends on the ability to trace, after the fact, what decision was made, by whom or by what, on the basis of what information, and with what review. A system in which AI decisions are not reconstructable, or in which the human-authoritative state is reconstructable only as a runtime promise that may have been disabled, is a system whose accountability evaporates at the moment accountability is needed.

The response in current regulatory frameworks is to require*human oversight* of AI systems classified as high-risk. The EU AI Act Article 14 [1] mandates that high-risk AI systems be designed to permit effective human oversight throughout their deployment, with natural persons enabled to monitor the system, understand its outputs, detect anomalies, and disregard the system's output when judgment warrants. The FDA's AI/ML Software as a Medical Device action plan, the NHTSA framework for autonomous vehicle oversight, and the EASA concept paper on AI in aviation all require some form of human oversight for AI systems whose decisions have safety-of-life consequences.

These regulatory frameworks specify*what* the human-AI relationship must permit. They do not specify*how* the system must be architected so that the permission is actually structural rather than policy-level. A system can satisfy EU AI Act Article 14 on paper while implementing override as a configuration flag the operator may or may not be able to invoke under operational stress. The present paper addresses the architectural gap: how should an adaptive AI orchestration system be designed so that human authority is a constitutional property of the system, not a runtime configuration?

The architectural design principle is named here as *Supervisory Primacy*. The paper articulates the principle, distinguishes it from existing HITL taxonomies, identifies the architectural mechanisms that operationalize it, and illustrates the operationalization in MILO [3].

2. Background and Related Work

EU AI Act Article 14 — Human Oversight. Article 14 of the EU AI Act, the regulation that entered into force on 1 August 2024 [1] with high-risk-system obligations applying from 2 August 2026 (and from 2 August 2027 for high-risk AI systems embedded in regulated products), requires high-risk AI systems to be designed to enable effective human oversight by natural persons during the period in which the system is in use. The required capacities under Article 14(4) are: (a) properly understanding the high-risk AI system's capacities and limitations; (b) remaining aware of the tendency to over-rely on system outputs (automation bias); (c) correctly interpreting outputs; (d) deciding not to use the system or otherwise disregard, override, or reverse its outputs; and (e) intervening in the operation of the system or interrupting it through a "stop" function. (The broader monitoring obligation is set out in Article 14(2)–(3) rather than enumerated in 14(4).) Article 14 is the regulatory floor; it specifies what oversight must permit, not how the system must be architected to make the permission effective.

Parasuraman, Sheridan, and Wickens — Levels of Automation. The canonical taxonomy of human-automation interaction[2] organizes automation into four stages (information acquisition, information analysis, decision/action selection, action implementation) along a continuum of automation levels, with the ten-level Sheridan–Verplank scale applied most directly to the decision/action-selection stage and ranging from full human control to full automation with no human intervention. The taxonomy is descriptive: it specifies what level of automation a given system implements. The taxonomy does not specify what level a high-consequence system*should*

implement; that is a design choice. Supervisory Primacy operates within the Parasuraman-Sheridan-Wickens framework by pinning the default position at the human-authoritative end of the action-implementation stage for any class of action classified as consequential, and by specifying that the pin be implemented as an architectural property of the system rather than as a configuration parameter.

FDA, NHTSA, EASA — Sector-Specific Oversight Guidance. Sector-specific regulatory frameworks have developed their own articulations of human oversight requirements: the U.S. Food and Drug Administration's*AI/ML Software as a Medical Device Action Plan* emphasizes total product life-cycle oversight for adaptive AI in medical devices[6]; the National Highway Traffic Safety Administration's Standing General Order on Crash Reporting for Automated Driving Systems requires reporting of crash events involving Level 2+ automation, establishing an operational data-collection regime for the human-system interaction[7]; the European Union Aviation Safety Agency's Concept Paper on AI in aviation specifies categorical levels of human-AI teaming with increasing AI autonomy bounded by increasing oversight requirements[8]. The frameworks differ in detail, but together they signal a regulatory direction in which the human-AI relationship for high-consequence systems must be documented with sufficient fidelity to permit post-hoc reconstruction. Supervisory Primacy adopts this direction as an architectural target: the audit trail is the system's substrate, not a logging feature added after the fact.

Industrial robotics and functional-safety standards. For deployments involving physical actuation — industrial robots, collaborative robots, motion-control systems — the applicable safety standards are ANSI/RIA R15.06 [11], ISO 10218-1 and ISO 10218-2 [12] (industrial robot safety requirements for robots and for robot systems and integration, respectively), and ISO/TS 15066 [13] (collaborative-robot safety). At the functional-safety layer, IEC 61508 (electrical/electronic/programmable electronic safety-related systems) and IEC 61511 (process-industry safety instrumented systems)[14] specify the safety-integrity-level (SIL) framework against which safety-related functions are designed and validated. Supervisory Primacy is intended to compose with — not replace — these standards: the SIL-rated safety function (e.g., an emergency stop tied to a light curtain) remains the deterministic safety floor, while Supervisory Primacy governs the *consequential decision* layer above it (e.g., whether to start a robotic motion at all, on what authorization, and with what audit trail). The architectural substrate makes a SIL-rated safety event and a Supervisory Primacy authorization event part of the same reconstructable record.

Beer's Viable System Model and Supervisory Control. Stafford Beer's Viable System Model [5] specifies the recursive cybernetic structure of viable systems, with each subsystem retaining identity and authority under environmental change. Supervisory Primacy is consistent with Beer's framework: the human operator occupies the position that, in Beer's structure, would be designated as System 5 (identity, policy, and purpose) — the seat of authority from which the system's responses to environmental disturbance are governed.

3. The Principle of Supervisory Primacy

Supervisory Primacy is stated as follows:

> *In an adaptive AI orchestration system deployed in a high-consequence domain, the human-authoritative state is the architectural default for any action classified as consequential. The AI proposes; the human disposes. Every consequential action carries a mandatory authorization audit trail recorded before the action dispatches. The eight non-negotiable operational integrity constraints (no coercion, individual baseline only, no surveillance, operator authority invariant, transparency, data sovereignty, override always, independent oversight) are implemented as enforceable safeguards in signed deployment builds rather than as runtime policy or configuration.*

The principle is not a new taxonomy of human-AI interaction. It is the *architectural form* of human authority — the substrate that makes existing HITL taxonomies and existing regulatory oversight requirements operational under deployment stress. Supervisory Primacy does not invent the concept that humans should retain authority over AI decisions in high-consequence domains. It specifies the structural conditions under which the retention is mechanical rather than promissory.

The principle has three architectural commitments.

Commitment 1 — Default state is human-authoritative. For any class of action classified as consequential, the system's default action is to surface the proposed action to the human operator together with the AI's reasoning and recommendation. The default action is not execute-with--override-available; it is propose-pending-authorization. The distinction is structural: execute-with-override-available admits a race condition in which the action executes before the human can override; propose-pending-authorization eliminates the race because the action does not begin until authorization is recorded.

Commitment 2 — Audit trail by architecture, not by policy. Every command issued in the system persists to a durable, append-only audit log before the command dispatches to its target [3]. If the process terminates between dispatch and execution, replay reconstructs exact state. The audit trail is the substrate, not a logging feature. The architectural pattern named *persist-before-deliver — every command persists before it dispatches; every signal persists before it fans out* — is the mechanism by which the system's behavior is reconstructable under arbitrary failure modes.

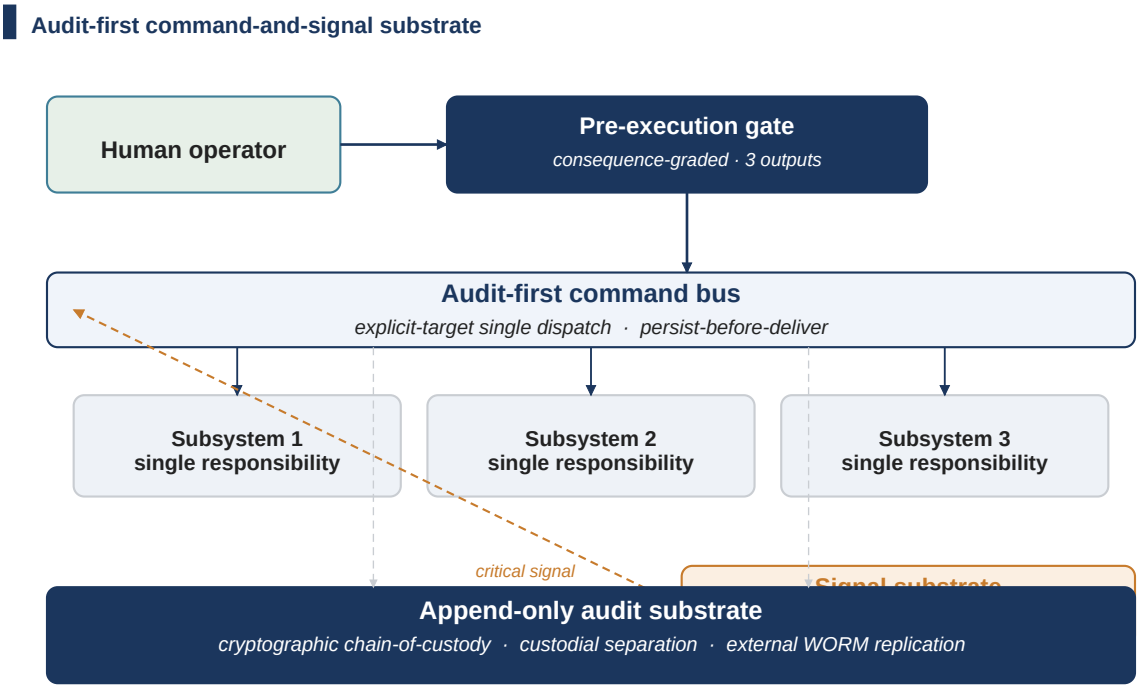
Commitment 3 — Integrity constraints are architectural-target enforceable safeguards. The eight operational integrity constraints — *no coercion ever*, *individual baseline only*, *no surveillance architecture*,*operator authority is the invariant*,*operational transparency*,*data sovereignty*, *override always available*,*independent oversight* [4] — are specified as architectural commitments to be implemented as code-level invariants bound to the design specification and compiled into signed deployment builds for deployment-grade systems. The architectural target is that disabling any constraint should require rebuilding from source, not toggling a runtime flag — making compliance mechanical rather than promissory. Deployments that cannot satisfy all eight constraints are outside the permitted use boundary of the architecture by design.

3.1 The Latency-Budget Tradeoff

Commitment 1 (default state is human-authoritative) imposes a latency budget that is the central design tension of Supervisory Primacy. The principle requires that consequential actions begin only after the human has recorded authorization. If the latency budget for authorization is too long (seconds of operator response time per consequential action), the system cannot keep up with the operational tempo of the environment; routine actions become operationally infeasible. If the latency budget is too short (the operator has only a millisecond to react), the human authorization becomes a rubber stamp and Supervisory Primacy degrades into automation with theatrical human-in-the-loop. The principle is meaningful only when the consequence classification of an action is calibrated such that the operational tempo at that classification matches the latency budget available to the human. Routine actions, by definition, should not be classified as consequential; consequential actions, by definition, should occur at a tempo compatible with deliberate human authorization. The architectural design choice is therefore the *consequence classification itself* — what does and does not count as consequential — rather than the override mechanism. This relates closely to consequence-graded authorization in industrial control environments, where the latency budget for human authorization is dictated by the operational tempo at the action's consequence tier rather than by a uniform per-command policy.

4. Architectural Mechanisms

The principle of Supervisory Primacy is operationalized through four architectural mechanisms, each implemented in MILO and illustrated here at the public-architecture-metaphor level. The four mechanisms are not selected because they happen to exist in MILO; they are selected because each one resolves a question that human-in-the-loop architecture must answer regardless of implementation.*Audit-first command flow* answers"*what did the human authorize, and in what context*" under arbitrary failure modes. *Reflex Arcs Run Before Fanout* answers"*what response is faster than human reaction time, and how is that response architecturally bounded so it does not escape human authority.*" *Pre-execution gating* answers"*at what specific architectural point does authorization happen, and what are the permitted outputs of that decision.*" *Voluntary side-effect pathway* answers"*how does the architecture prevent runaway action sequences from a single human decision.*" The four mechanisms are HITL-specific in the sense that each one addresses a structural problem of human-AI authority allocation that exists in any HITL architecture; the MILO implementations are concrete instances of solutions, not the only possible solutions.



very signal persists to the append-only audit before delivery. High-consequence deployments add cryptographic chain-of-custody, custodial separation, and externa

Mechanism A — Audit-first command-and-signal substrate. The orchestrator implements three load-bearing patterns: every command travels down a single command bus to its target, every signal travels up a single signal bus to its subscribers, and every message is audited*before* it is delivered[3]. The pattern is not a logging convenience; it is the substrate that makes Commitments 1, 2, and 3 mechanically enforceable rather than promissory. The audit substrate is required to be append-only and durable so the trail survives arbitrary process termination. At the framework level, append-only continuity — not cryptographic tamper-evidence — is the *minimum* architectural property required for post-hoc reconstructability; hardening is a separate architectural commitment described in the threat model below.

Audit-substrate threat model. The entire architecture rests on audit-trail integrity, so the substrate is itself a high-value attack target. Append-only continuity at the orchestrator is the

minimum property and is necessary but not sufficient against a sophisticated attacker — particularly an insider with administrative access to the host. For high-consequence deployments, the framework requires three additional architectural commitments: (a) **cryptographic chain-of-custody** — each audit record is signed and chained (hash-linked) so silent tampering is detectable; (b) **custodial separation** — the audit-log writer and the audit-log custodian are distinct services owned by different operational roles, so no single principal can both produce and curate the record; and (c) **external WORM replication** — every audit record is streamed in near-real-time to an independent write-once, read-many sink under separate administrative control (cf. NIST SP 800-92 [10] on log management and NIST SP 800-53 control AU-9 on protection of audit information). Deployments that cannot satisfy all three commitments do not meet the high-consequence deployment posture and are outside the permitted use boundary of the architecture for those environments.

Mechanism B — Reflex Arcs Run Before Fanout. Critical signals (e.g., safety-relevant anomalies) are evaluated by reflex predicates before the signal fans out to general subscribers[3]. A reflex may halt the system before the human is informed of the underlying signal, mirroring the spinal-cord reflex that fires before the brain is aware of the stimulus. This pattern admits a specific class of action — emergency halt — that operates faster than human reaction time but remains architectural: the reflex dispatches a halt command through the same audited command bus as any other command, with end-to-end audit entries from critical signal through reflex through halt-executor. The pattern preserves Supervisory Primacy under emergency conditions: the human cannot react faster than the reflex, but every reflex action is auditable, reversible (by resume command), and bounded by Commitment 3's integrity constraints.

Mechanism C — Pre-Execution Gating. Every consequential action is gated through a pre-execution validator with three outputs:*allow* (logged), *hold/block* (with plain-language operational explanation), or *recommend* (safer alternative)[4]. The pattern surfaces the AI's reasoning at the moment of decision rather than reconstructing it post-hoc. The pattern admits the human override at every gate, with the override logged for audit but never used to trigger adverse personnel or operational consequences (Commitment 3, integrity constraint #7).

Mechanism D — Voluntary Pathway for Side Effects. The architectural pattern explicitly excludes auto-chaining: the orchestrator does not automatically follow a completion signal with a subsequent action[3]. The human-authoritative state is required to re-enter for any subsequent consequential action. The pattern prevents the runaway sequence in which the AI's first action triggers a chain of subsequent actions before the human can intervene. The pattern is voluntary in the architectural sense: side effects require explicit commanding, not automatic propagation.

5. Relationship to Existing HITL Taxonomies

Supervisory Primacy operates within the Parasuraman-Sheridan-Wickens framework[2] by pinning the default position at the human-authoritative end of the action-implementation stage for consequential actions. The pin is structural rather than configurable. The principle does not displace the taxonomy; it specifies where in the taxonomy a high-consequence adaptive AI orchestrator should be designed to operate by default.

Relative to the EU AI Act Article 14 [1], Supervisory Primacy operates as the architectural substrate by which the regulation's required oversight capacities (the five Article 14(4) capacities — understanding limits, automation-bias awareness, correct interpretation, disregard, and interruption — together with the general monitoring obligation in Article 14(2)–(3)) become operationally effective rather than merely permitted. A system that satisfies Article 14 on paper but implements override as a configuration flag fails Supervisory Primacy; a system that implements override through the audit-first command bus with the eight integrity constraints as enforceable safeguards satisfies both.

Relative to sector-specific oversight guidance (FDA, NHTSA, EASA), Supervisory Primacy provides a common architectural substrate across sectors. The substrate does not specify sector-specific risk classifications or decision authority allocations — those remain sector-specific. The substrate ensures that whatever allocation a sector regulator specifies is structurally implemented rather than policy-implemented.

6. Implications and Discussion

Supervisory Primacy has implications for the design, evaluation, and deployment of adaptive AI orchestration systems in high-consequence domains.

For *design*, the principle implies that the architectural choices made before any AI model is trained — the command bus, the audit substrate, the integrity-constraint enforcement mechanism, the reflex layer, the voluntary side-effect pathway — bound the system's eventual capacity to support effective human oversight more tightly than any subsequent regulatory compliance retrofit can recover. A system designed for execute-with-override-available cannot be converted into a propose-pending-authorization system without rebuilding the substrate.

For *evaluation*, the principle implies that the conventional evaluation metrics for AI systems — accuracy, latency, throughput — are necessary but insufficient. An adaptive AI orchestration system in a high-consequence domain must be evaluated additionally on: (a) audit-trail completeness under arbitrary process termination; (b) the structural enforceability of the integrity constraints (can they be disabled at runtime, or do they require rebuilding from source?); (c) the architectural enforceability of the voluntary side-effect pathway (does the system auto-chain, or does it require explicit re-commanding?); (d) the latency and audit fidelity of the human override pathway under operational stress.

For *deployment*, the principle implies that high-consequence domains — energy grid control rooms, nuclear facility operations, advanced manufacturing under operator supervision, autonomous robotics with human-robot interface, satellite and space operations, first-responder coordination — should not deploy adaptive AI systems whose Supervisory Primacy commitments are policy-level rather than architectural. The submission of MILO under the U.S. Department of Energy's Genesis Mission[4] is intended to make a viable-architecture alternative available to those domains.

6.1 Limitations

The principle is articulated at the architectural-design level. Three specific limitations bound the present contribution: (i) the consequence classification of an action (the central design tension named in §3.1) is specified as a deployment-time activity and no canonical taxonomy is offered; (ii) the audit-substrate hardening commitments (§4, "Audit-substrate threat model") are specified as architectural requirements but their composition with specific cryptographic chain-of-custody implementations (e.g., AWS QLDB, Sigstore-style transparency logs, blockchain-anchored audit logs) is not benchmarked here; (iii) the composition with the industrial-robotics functional-safety standards[11]–[14] is asserted at the principle level and not validated against any SIL-rated safety case. Quantitative validation — operator-override latency under operational stress, audit-trail completeness under adversarial process termination, and incident-reconstruction fidelity against actual sociotechnical-failure post-mortems — is forthcoming work and is not claimed here. The submission of MILO under the U.S. Department of Energy's Genesis Mission[4] is under review; no acceptance or grant outcome is claimed.

7. Conclusion

This paper has introduced Supervisory Primacy as the architectural design principle by which human authority over consequential AI decisions becomes a constitutional property of an adaptive AI orchestration system rather than a runtime configuration. The principle is consistent with EU AI Act Article 14 [1], operates within the Parasuraman-Sheridan-Wickens levels-of-automation framework[2], aligns with NIST AI RMF 1.0 Appendix C on human-AI interaction[9], and is illustrated using MILO [3], the author's working orchestrator submitted under the U.S. Department of Energy's Genesis Mission[4]. The contribution is at the architectural level: Supervisory Primacy is not a new HITL taxonomy; it is the structural design — audit-first command flow, reflexes before fanout, pre-execution gating, voluntary side-effect pathway, and integrity constraints as architectural-target safeguards in signed builds — that makes human oversight load--bearing rather than retrofittable. The principle's failure modes are mechanical (an audit trail with a gap, a runtime flag that disables an integrity constraint, an auto-chain pathway, an opaque resolver in the command bus); its successes are mechanical (an unbroken audit trail, a structurally inviolate integrity constraint, an explicit voluntary side-effect pathway, a transparent command bus with single-target dispatch). Related architectural directions by the author — independence-verifiable multi-source cryptographic entropy, latency-aware authentication in industrial control, structural principles for adaptive AI architecture, and the discipline of viability under non-stationary conditions — each inherit Supervisory Primacy as a constitutional property.

Data Availability

All architectural materials, source manuscripts, the reference implementation, and accompanying figures are openly available at https://github.com/jmontano1/milo-architecture and permanently archived at Zenodo (DOI: 10.5281/zenodo.20117025). No private datasets are referenced; the architectural framework itself is the subject of this paper. Patent rights for the underlying MILO software architecture are reserved; the ~MILO trademark is held under USPTO Serial No. 99706004 (intent-to-use, Class 009).

References

[1] European Union, *Regulation (EU) 2024/1689 of the European Parliament and of the Council of 13 June 2024 laying down harmonised rules on artificial intelligence (Artificial Intelligence Act)*, Article 14 (Human Oversight), Aug. 2024. [Online]. Available: https://eur-lex.europa.eu/eli/reg/2024/1689/oj

[2] R. Parasuraman, T. B. Sheridan, and C. D. Wickens, "A model for types and levels of human interaction with automation," *IEEE Transactions on Systems, Man, and Cybernetics — Part A: Systems and Humans*, vol. 30, no. 3, pp. 286–297, May 2000. doi:10.1109/3468.844354

[3] J. E. Flores Montano, *MILO (Modular Intelligent Learning Orchestrator)*, JM Automated Solutions. Patent pending. Submitted under the U.S. Department of Energy Genesis Mission, 2026.

[4] U.S. Department of Energy, "The Genesis Mission: Transforming Science and Energy with AI," Office of the Under Secretary for Science, Executive Order 14363, November 2025. [Online]. Available: https://www.energy.gov/genesis

[5] S. Beer, *Brain of the Firm*, 2nd ed. Chichester, UK: Wiley, 1981.

[6] U.S. Food and Drug Administration, *Artificial Intelligence/Machine Learning (AI/ML)-Based Software as a Medical Device (SaMD) Action Plan*, FDA, Jan. 2021.

[7] National Highway Traffic Safety Administration, *Standing General Order 2021-01 on Crash Reporting for Vehicles Equipped with Automated Driving Systems and Level 2 Advanced Driver Assistance Systems*, NHTSA, Jun. 2021 (Second Amendment effective May 15, 2023).

[8] European Union Aviation Safety Agency, *Concept Paper: Guidance for Level 1 & 2 Machine Learning Applications*, EASA AI Roadmap, Issue 02, Mar. 2024.

[9] E. Tabassi, *Artificial Intelligence Risk Management Framework (AI RMF 1.0)*, NIST AI 100-1, National Institute of Standards and Technology, Jan. 2023, Appendix C: AI Risk Management and Human-AI Interaction. doi:10.6028/NIST.AI.100-1

[10] K. Kent and M. Souppaya, *Guide to Computer Security Log Management*, NIST SP 800-92, National Institute of Standards and Technology, Sep. 2006. doi:10.6028/NIST.SP.800-92

[11] R. Y. Wang and U. Karaman, *ANSI/RIA R15.06: American National Standard for Industrial Robots and Robot Systems — Safety Requirements*, American National Standards Institute / Robotic Industries Association, 2012 (R2020).

[12] International Organization for Standardization, *ISO 10218-1: Robots and robotic devices — Safety requirements for industrial robots — Part 1: Robots*, ISO, 2011 (revised 2025); and *ISO 10218-2: Part 2: Robot systems and integration*, ISO, 2011 (revised 2025).

[13] International Organization for Standardization, *ISO/TS 15066: Robots and robotic devices — Collaborative robots*, ISO, 2016.

[14] International Electrotechnical Commission, *IEC 61508: Functional safety of electrical/electronic/programmable electronic safety-related systems*, IEC, 2010; and *IEC 61511: Functional safety — Safety instrumented systems for the process industry sector*, IEC, 2016.

About the author

Jorge Enrique Flores Montano (ORCID iD: 0009-0003-1859-8418; jmontano@jmautomated.com) is the founder of JM Automated Solutions and the inventor of MILO. A full biography is maintained at https://www.milo-usa.com/jorge-enrique-flores-montano.html.

Conflict of Interest and Funding Disclosure

The author is the inventor of MILO (patent pending) and the founder of JM Automated Solutions. The principle proposed in this paper is a contribution from a working development program in which the author retains sole authorship and inventive interest. No external funding was received for the preparation of this manuscript. The author retains all rights to MILO and to the principle articulated herein.

Appendix A — About MILO

MILO (Modular Intelligent Learning Orchestrator) is a patent-pending adaptive AI orchestration architecture organized into discrete, single-responsibility subsystems under a strict separation-of-concerns discipline. An audit-first command-and-signal substrate persists every command before dispatch and every signal before fanout, producing an append-only audit trail that survives arbitrary process termination. The architecture is designed for *viability* — operational continuity under non-stationary conditions — rather than for prediction accuracy against an expected future.

Eight structural principles

Six are established physical, informational, control-theoretic, and statistical laws applied as architectural design constraints; two are original frameworks proposed by the author for the operator-cognitive performance layer of high-consequence systems.

1. **Second Law of Thermodynamics** — entropy treated as an architectural diagnostic signal, not a fault to be suppressed.

2. **Ashby's Law of Requisite Variety** — a regulator must possess variety at least equal to the system it regulates; implemented as a fleet of specialist agents matching the operational domain.
3. **Shannon Information Theory** — variance reduction occurs at the signal-carrier level, not redundantly at each consumer.
4. **Principle of Least Action — Single-Target Dispatch** — every command has one explicit target; no implicit resolvers, no opaque dispatchers.
5. **Lyapunov-Style Bounded Response** — every adaptive subsystem admits an explicit halt-and-resume pathway; adaptation that drifts unboundedly is failure, not learning.
6. **Power-Law Distribution Architecture** — engineered for the 99th-percentile event, not the median.
7. **Individual-Baseline Variance Modeling** *(original framework)* — operator-layer interventions calibrated against the individual's own established performance baseline, never a population norm. Design-stage; pending empirical validation.
8. **Precision Perturbation Without Variance Compression** *(original framework)* — operator-layer interventions shift probability mass toward high-reliability decision outputs while preserving operator authority and the variability that *is* the operator's adaptive intelligence. Design-stage; pending empirical validation.

Eight operational integrity constraints

Architectural commitments designed to be implemented as enforceable safeguards in deployment builds — not as runtime policy. Disabling any constraint should require rebuilding from source, not toggling a flag.

1. **No coercion, ever** — the system issues recommendations, never compels.
2. **Individual baseline only** — measurements against the operator's own baseline; never against a population norm or productivity target.
3. **No surveillance architecture** — performance-support tool, not a monitoring infrastructure.
4. **Operator authority is the invariant** — the system expands effective decision options; it never narrows or preempts them.
5. **Operational transparency** — every recommendation includes a plain-language explanation.
6. **Data sovereignty** — operator-layer data belongs to the institutional program under documented data governance.
7. **Override always available** — overrides are logged for audit but never used for adverse personnel action.
8. **Independent oversight** — operator-layer deployments require institutional ethics-board review, published consent frameworks, and periodic third-party audits.

Unifying principle

> *MILO does not predict the future. It remains viable in any future.*

The principle is falsifiable: a system whose audit trail is incomplete, whose recovery is improvised, whose adaptation drifts unboundedly, or whose operator override is policy-level rather than architectural, fails the principle.

Trademark, patent, and submission status

- **Mark.** ∼*MILO*™ — U.S. Patent and Trademark Office Serial No. 99706004; filed March 16, 2026; intent-to-use; International Class 009 (downloadable AI software). The leading tilde disambiguates from senior MILO marks held by unrelated owners in different International Classes.
- **Patent.** Patent application pending for the underlying software architecture. Implementation may require a patent license once issued; nothing in this document or its CC BY 4.0 license on the manuscript text grants any patent license.
- **Federal submission.** Submitted to the U.S. Department of Energy under the *Genesis Mission* (Executive Order 14363, November 2025); currently under review. No acceptance or grant outcome is claimed.
- **Concept DOI.** 10.5281/zenodo.20117025 — Zenodo, persistent across versions.
- **Public reference.** https://github.com/jmontano1/milo-architecture.
- **Author contact.** Jorge Enrique Flores Montano · jmontano@jmautomated.com · ORCID iD: 0009-0003-1859-8418.