Authority Architecture under AI Execution

HIGHLIGHTS

Boards currently oversee Strategy, Financial performance, and Risk.

These assume that execution follows **visible authority structures**.

AI systems change this. Execution can now occur across systems at machine speed **without passing through traditional hierarchies**.

Authority may be exercised without being explicitly visible.

Execution speed now exceeds governance cycles.

The structure the Board governs is no longer the structure through which execution occurs.

The Governance Gap

- Traditional delegation frameworks assume authority flows through people.
- Authority architecture is being formed through systems and workflows.
- Authority can now expand through system behaviour with implicit delegation.

Authority architecture governs **how authority is exercised through systems**.

The Structural Shift

- In Vol. 02: Authority must be explicit.
- In Vol. 03: Instruction formalisation ensures authority becomes operational.
- Now: Authority is **formalised and embedded** in **system design**.

It must be explicitly defined, approved, and enforced **before execution scales**.

This is distinct from risk management.
It defines who or what is allowed to act before risk is assessed.

Authority architecture **evolves as systems learn and scale**.

Authority Architecture

Authority architecture defines:

- Where authority resides.
- Where authority is bounded.
- What activates authority.
- Who holds accountability.

Authority architecture must be explicitly defined, approved, and enforced in system design.

If any element is undefined, authority will be assumed by systems.

Misaligned authority can result in financial loss, regulatory breach, or reputational damage.

Sample Authority Architectures

Customer Refund Process

- **Authority Resides**: System automatically approves and processes refunds.
- **Authority Boundaries**: Capped at a strict financial threshold (e.g., $500). The above threshold requires human intervention.
- **Authority Activation**: Triggered exclusively by a structured customer request via the verified portal.
- **Authority Accountability**: Head of Operations.

What Must Remain Human (Next 3 Years)

Not everything. But not nothing.

Over the next three years, the boundary must be explicitly defined and approved as part of authority architecture.

- **Irreversible decisions**: *actions that cannot be undone.*
- **Accountability-bearing decisions**: *where responsibility cannot be diffused.*
- **Authority-defining decisions**: *where authority is granted or expanded.*
- **Ambiguous or novel situations**: *where rules are incomplete or unstable.*

Changes to this boundary must be explicitly approved by the Board.

AI may support. Authority must remain human.

Board–Executive Responsibility

Executives design authority architecture.

The Board defines acceptability and approves it — including where authority must remain human.

Executives are accountable for design.
The Board is accountable for oversight of its adequacy.

Accountability

Legal and fiduciary responsibility does not transfer to systems.

If authority is not designed in advance, it will be defined by execution.

Authority must be designed before it is exercised.

ACTION: What the Board Must Require

- Explicit definition of where authority resides.
- Clear boundaries for where systems may execute, now and as they scale.
- Authority is activated only through formalised instruction.
- Every system-executed action has a named accountable owner.
- Enforceable ability to override or withdraw system authority.
- Approval before expanding system execution into new domains.
- Periodic review and approval of authority architecture as systems scale.
- Re-evaluation of authority architecture when systems are scaled, interconnected, or repurposed.

Autonomy scales. Authority contains.

Hadi Hendrawan
Advising CEOs on AI Risk, Authority & Accountability
March 2026

SUPPLEMENT 1: Authority Architecture Use Cases

The following models illustrate how Authority Architecture is applied to high-risk, high-speed, and high-impact AI systems. In each case, governance is achieved by strictly coding the boundaries of execution *before* deployment.

Case Study 1: Algorithmic Treasury & FX Hedging

System Name: Project Titan (Autonomous Liquidity Engine)

Strategic Intent: To autonomously execute foreign exchange (FX) trades and hedge against currency volatility at machine speed, exploiting micro-fluctuations that human traders cannot catch.

1. **Residence:** Core Treasury AI Engine (integrated via direct API to primary banking partners).
2. **Boundaries (The Hard Limits)**:
    a. *Financial:* Maximum exposure of $50M per asset class per 24-hour cycle.
    b. *Geopolitical:* Strictly hard-coded to reject any trades involving entities or currencies under emerging global sanctions.
3. **Activation (The Triggers):** Triggered autonomously when real-time currency fluctuations exceed a 1.5% deviation from the 30-day moving average, combined with a confirmed liquidity surplus in the main corporate account.
4. **Accountability:** Chief Financial Officer (CFO).

The "Human-Only" Perimeter: The system **cannot** open new bank accounts, authorize the use of novel financial instruments (e.g., unapproved derivatives), or expand its daily financial limit.

Override Protocol: "Circuit Breaker" toggle on the CFO’s dashboard. If global news sentiment APIs detect extreme, unpredicted market panic, the system automatically degrades to a "suggest only" mode, requiring human approval for all trades.

Case Study 2: Autonomous Cybersecurity Active Defense

System Name: Sentinel Zero (Active Threat Neutralization)

Strategic Intent: To instantly detect, isolate, and neutralize cyber threats (like ransomware) at machine speed, preventing lateral movement across the network before a human security analyst can respond.

1. **Residence:** Cloud-Native Security Operations Center (SOC) AI.
2. **Boundaries (The Hard Limits)**:
    a. *Operational:* Permitted to instantly sever network connections for individual employee endpoints (laptops) and regional branch servers.
    b. *Exclusion Zone:* **Strictly prohibited** from severing the primary customer database cluster or the core payment processing gateway without human authorization.
3. **Activation (The Triggers):** Triggered by the detection of anomalous lateral network movement, unauthorized data exfiltration exceeding 500MB, or the execution of known ransomware cryptographic signatures.
4. **Accountability:** Chief Information Security Officer (CISO).

The "Human-Only" Perimeter: The system **cannot** authorize counter-attacks ("hack backs") against external threat actors, nor can it permanently wipe company servers to prevent data theft.

Override Protocol: Redundant physical and digital override keys held by the CISO and VP of Infrastructure. If the system's own integrity is questioned, it fails to a "monitor, log, and alert" state rather than an active mitigation state.

Case Study 3: Dynamic Workforce Management

System Name: ShiftSync AI (Predictive Labor Allocator)

Strategic Intent: To dynamically adjust retail staffing levels, reallocate shifts, and approve or deny employee Time Off requests in real-time based on foot-traffic predictions, weather data, and supply chain deliveries.

1. **Residence:** Global Human Resources Information System (HRIS) module.
2. **Boundaries (The Hard Limits)**:

    a. *Operational:* Can assign or remove a maximum of 3 shifts per employee per week.
    b. *Compliance:* Hard-coded to never schedule an employee in violation of local labor laws (e.g., mandatory 10-hour rest periods between shifts).
3. **Activation (The Triggers):** Triggered continuously by store-level API data streams (weather forecasts, local event schedules, real-time sales velocity) intersecting with employee availability matrices.
4. **Accountability:** Chief Human Resources Officer (CHRO).

The "Human-Only" Perimeter: The system **cannot** initiate employee terminations, issue formal disciplinary actions, cut base pay rates, or formally demote personnel. (These remain ultimate accountability-bearing decisions).

Override Protocol: Regional HR Directors have dashboard authority to freeze the AI schedule for their specific region. If the system proposes a schedule that affects more than 20% of a region's workforce simultaneously, it is held in a queue for human approval.

SUPPLEMENT 2: Board Approval Template

AI System Authority Architecture

Document Control

- **System Name/Initiative:** [e.g., Project Apex - Dynamic Supply Chain Routing]
- **Requesting Executive (Accountable Owner):** [e.g., Jane Doe, Chief Operating Officer]
- **Target Deployment Date:** [Date]
- **Request Type:** [ ] Initial Deployment | [ ] Expansion of Authority | [ ] Repurposing

1. Executive Summary & Strategic Intent

Briefly describe what the system does, the business value of allowing it to execute autonomously at machine speed, and why human-in-the-loop is no longer viable for this specific workflow.

- **Objective:** [Insert 1-2 sentences on what the system will achieve.]
- **Execution Scope:** [Insert exactly what the system is being authorized to *do* (e.g., re-route global freight, adjust pricing, approve tier-1 claims).]

2. The Authority Architecture (The Four Pillars)

Define the exact parameters of the authority being granted to this system.

Architecture Pillar	**System Specifics** (To be completed by Executive)
1. Residence *Where the authority technically sits.*	**System/Module:** [Name of the specific AI model or software module executing the action.]

Architecture Pillar	**System Specifics** (To be completed by Executive)
2. Boundaries *The hard, coded limits of action.*	**Financial Limit:** [e.g., Up to $50,000 per transaction] **Operational Limit:** [e.g., Only applies to North American logistics networks] **Time/Volume Limit:** [e.g., Maximum of 500 executions per hour]
3. Activation *What triggers the authority.*	**Instruction Formalisation:** [e.g., Triggered only by a weather event severity score > 7 combined with a port delay > 24 hours via API.]
4. Accountability *The named human owner.*	**Primary Accountable Exec:** [Name & Title] **Technical Escalation Owner:** [Name & Title]

3. The "Human-Only" Perimeter Check

Executive confirmation that this system's authority does not breach the Board's mandated boundaries for human decision-making.

By checking these boxes, the Accountable Executive certifies that this system will **NOT** independently execute:

- [ ] **Irreversible Decisions** (Actions with permanent, unrecoverable consequences)
- [ ] **Accountability-Bearing Decisions** (Actions where legal/fiduciary responsibility cannot be diffused)
- [ ] **Authority-Defining Decisions** (The ability to grant, shift, or expand its own or another system's authority)
- [ ] **Ambiguous/Novel Decisions** (Operating outside of strictly defined, formalized instructions)
- *Exceptions/Notes:* [Detail any edge cases here or type "None."]

4. Override Protocols & Incident Response

How the system is controlled when execution exceeds defined authority or intent.

- **The "Kill Switch":** [Explain the exact mechanism a human can use to pause or withdraw the system's execution authority instantly. e.g., "Manual override toggle located in the central Ops dashboard; cuts API access."]
- **Default State on Failure:** [What happens if the system encounters an error or ambiguous data? e.g., "Fails safe to human-in-the-loop queue."]

BOARD RESOLUTION & SIGN-OFF

Based on the defined Authority Architecture above, the Board of Directors hereby:

- [ ] **APPROVES** the deployment and granting of execution authority to the system as defined.
- [ ] **APPROVES WITH CONDITIONS** [List conditions: e.g., reduced financial boundary for the first 90 days].
- [ ] **REJECTS / REQUIRES REVISION** [Reasoning].

Signatures:

_____ (Board Chair) | Date: _____

_____ (Accountable Executive) | Date: _____