McKinsey Sees the Agentic Organisation. SDAM Shows You How to Build One.

Autonomy Platform · Public Blog

PERSPECTIVE

Industry May 2026

McKinsey Sees the Agentic Organisation. SDAM Shows You How to Build One.

A new McKinsey piece maps the contours of agentic enterprise transformation with genuine insight. But the destination it describes requires a mathematical foundation the consulting narrative does not provide.

McKinsey's recent podcast and companion article on the *agentic organisation* is worth reading. Alexis Krivkovich and her colleagues are pointing at something real: most companies have invested heavily in AI and are not seeing bottom-line impact, agentic systems raise the stakes, and the organisational response required goes far deeper than adding a chatbot to a workflow.

The piece gets a lot right. The distinction between humans *in* the loop versus humans *above* the loop is precisely the right framing. The emphasis on end-to-end workflow reimagination over point solutions is correct. The observation that judgement, oversight, and systems thinking will be the premium human skills in an agentic world is well-grounded.

What the article cannot offer, because consulting narratives are not designed to offer it, is a rigorous account of how any of this actually works. And that gap matters more than it might seem.

The vocabulary problem

Throughout the piece, the word *judgement* does enormous work. Humans will exercise judgement above the loop. Senior leaders will provide judgement on agent outputs. Judgement is the thing AI cannot replace.

This is true. But it is also underspecified to the point of being operationally useless. What does it mean, precisely, for a human to exercise judgement over an agent's decision? Is it approval? Correction? A new instruction? Does the intervention pause execution or supersede it? Does it become part of the system's learning? Does it trigger re-planning across dependent decisions?

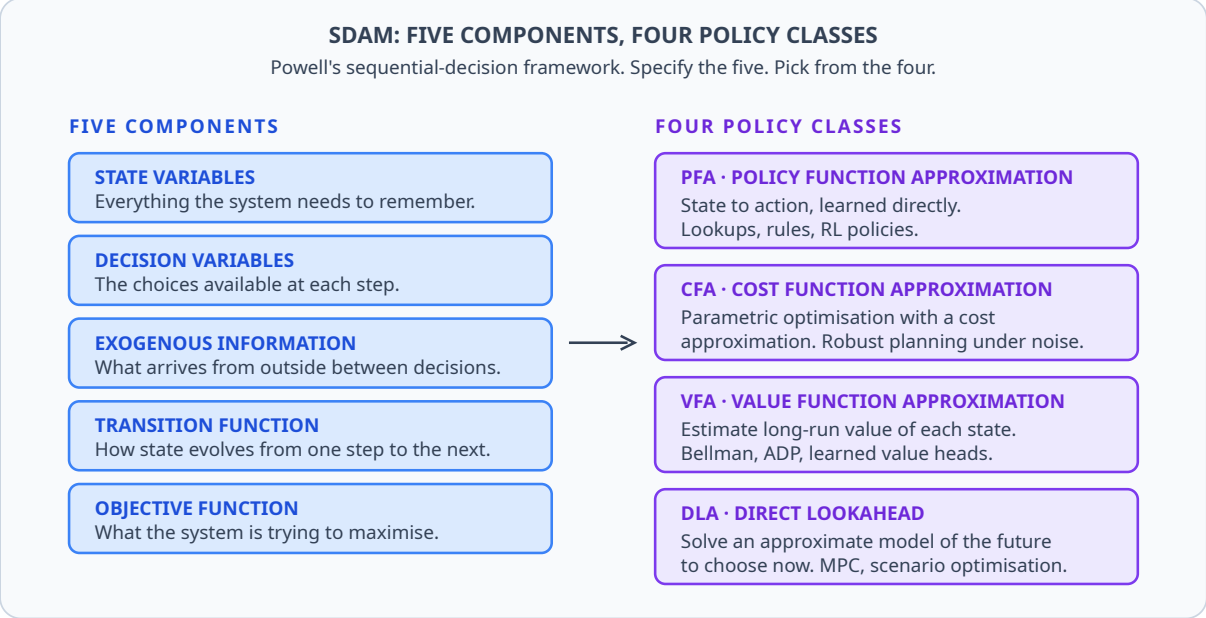
These are not philosophical questions. They are engineering requirements. And without answers, the *human above the loop* remains a reassuring concept rather than a designed capability.

The agentic organisation McKinsey describes is not primarily a cultural or structural challenge.

It is a decision-architecture challenge. The cultural and structural changes are downstream.

What SDAM provides

Warren Powell's *Sequential Decision Analytics and Modeling* (SDAM) framework offers the mathematical substrate that agentic enterprise systems require. It decomposes any decision problem into five components: state variables, decision variables, exogenous information, a transition function, and an objective function. It then classifies the policies that can govern decisions into four families: policy function approximations (PFAs), cost function approximations (CFAs), value function approximations (VFAs), and direct lookahead policies (DLAs).



The choice of policy class is an engineering decision driven by the structure of the problem, not by the technology fashion of the season. Real systems hybridise: a CFA wrapping a DLA, a PFA warm-started by a VFA, and so on.

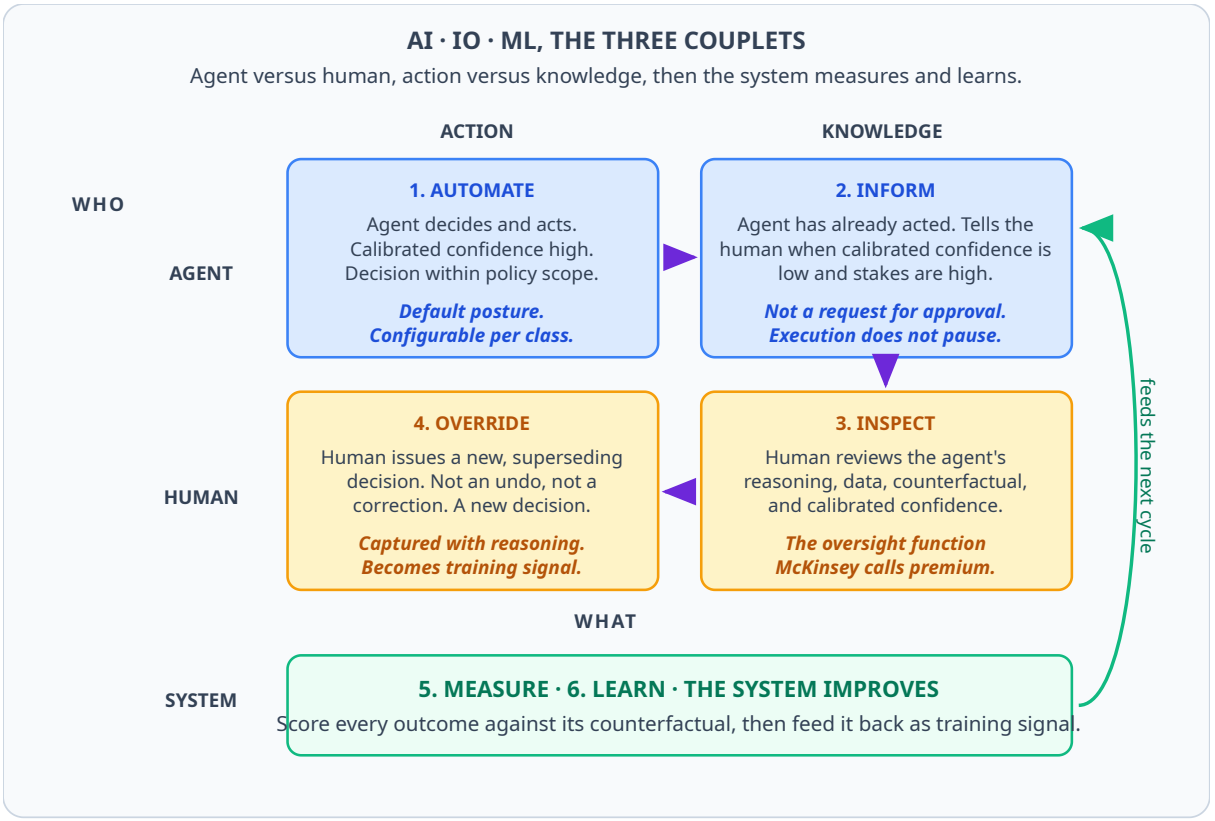
This is not academic overhead. It is the difference between a system that can reason about its decisions and one that merely executes them. When McKinsey describes agents *reviewing case files, assembling timelines, and arriving at a summary decision*, what they are describing, in SDAM terms, is

a direct lookahead policy operating over a state space that includes the case record, prior outcomes, and contractual terms. The human review layer is not supervision in a vague sense. It is an *Override*: a new superseding decision that becomes training data for the transition function.

The distinction is consequential. A system designed around SDAM can learn from human overrides. It can quantify its own uncertainty. It can surface decisions for human review based on calibrated confidence rather than arbitrary rules. A system designed around the consulting narrative cannot, because the consulting narrative has no mechanism for any of this.

The AI·IO·ML governance model

Azirella's AI·IO·ML operating model operationalises the governance layer that McKinsey gestures toward. It is one model in three couplets: the agent acts (**AI**: Automate, Inform), the human engages (**IO**: Inspect, Override), and the system improves (**ML**: Measure, Learn). The first two couplets, AI and IO, the ones that map directly to McKinsey's *human above the loop* framing, split cleanly along two axes: agent versus human, and knowledge versus action.



The purple arrows show the forward sequence A → I → I → O across the agent couplet (AI) and the human couplet (IO). The green band is the third couplet, ML - Measure (every outcome scored against its counterfactual) and Learn (every override fed back as training signal that tightens calibration) - and the green arrow shows it closing the loop back to a tighter, more confident Automate. Without that third couplet, the first four verbs are theatre.

The structure is not arbitrary. Automate and Inform are the agent's couplet (AI); Inspect and Override are the human's couplet (IO). In the quadrant, the action verbs sit on one diagonal and the knowledge verbs sit on the other. The symmetry reflects the mirrored accountability of agent and human in a well-designed agentic system: the agent decides by default and informs when uncertain; the human inspects to understand and overrides only when they know more. Beneath the quadrant sits the third couplet, ML, the system's own role: it Measures every outcome and Learns from it, closing the loop back to a sharper Automate.

Automate is the system's **default posture, not a fixed setting**. AI·IO·ML thresholds are learned, but every decision class also accepts a tenant policy: a tenant new to autonomous operations can require Inspect-and-approval before any action commits, effectively pushing every decision through the Inform pathway until calibrated confidence and operational trust accumulate. As classes earn the right, they graduate to Automate per the Decision Support → Augmentation → Automation deployment stages. The steady-state shape is the full AI·IO·ML; the policy posture sets the pace at which a tenant gets there.

The agent and human couplets decide and engage; the third couplet is what closes the loop. That is the ML couplet of the full AI·IO·ML model: *Measure* and *Learn*. Every decision and every override is scored against its counterfactual outcome. Every (decision, outcome, override) triple flows back as training signal that tightens calibration. Without the ML couplet, the first four verbs are a static diagram. With it, the system gets sharper every cycle, and decisions migrate safely from Inform into Automate as their calibrated confidence earns the right.

Why context belongs in the substrate, not the override

Matt Whetton, CTO at Acquired.com, described an instructive moment from outside supply chain this week. His team was architecting a payment-authorisation service with AI in the loop, and the AI proposed Redis as the caching and coordination layer. A plausible answer that would have shipped and looked fine in review. The team overrode it back to the

existing database, not because Redis was wrong in principle but because *“the AI was reasoning about the problem in front of it. We were reasoning about the problem in front of it plus the company around it.”*

That phrase is exactly right, and it also points to a structural answer rather than just a governance one. Leaving the *company around it* context to leak in through human override every time is the human-in-the-loop trap McKinsey is trying to escape: you have built an architecture that depends on a human catching every contextual mismatch.

Autonomy’s answer is the Context Engine. It is a substrate that ingests regulatory windows, customer commitments, contractual constraints, strategic intent, market signals, and supplier risk events, and feeds them into the agent’s reasoning as guardrails, targets, and context **before** the agent decides. The “we have a fine database, we don’t need Redis” piece of context is exactly the kind of thing the Context Engine carries. The agent’s recommendation is no longer reasoning about only the problem in front of it; it is already reasoning about the company around it.

Override is still needed, but for a smaller and more important set of cases: novel situations the Context Engine cannot have anticipated, strategic variation against the convergence trap, and the residual tacit knowledge of the business that has not yet been deposited into the substrate. The structural goal is to push as much of the company-around-it context into the substrate as possible, so the override workload converges on the high-value cases rather than the long tail of *the AI didn’t know X*.

The federated policy problem

McKinsey's article briefly raises what it calls the *systems thinking* required to unlock agentic value at scale: multiple agents, multiple functions, reusable components, cross-functional coordination. It identifies this as the frontier of the problem without going further.

In SDAM terms, what enterprise agentic systems require is a shared world model and transition function, with federated policy layers operating at different cadences and planning horizons. Supply planning, demand shaping, financial planning, product lifecycle, and sales planning are not independent problems. They share state. Their decisions create exogenous information for one another. Optimising each in isolation produces locally rational and globally incoherent outcomes.

This is why the *whole workflow* framing McKinsey advocates is directionally right but architecturally incomplete. The workflow is not the unit of design. The decision architecture is. And that architecture has to be coherent across functions, not just end-to-end within one of them. The five-tier decision hierarchy we publish (Context Engine on top, then Strategic, Tactical, Operational, and Execution below) is the concrete expression of what *federated policy layers over a shared world model* actually looks like. Each tier runs its own policy class at its native cadence; context and guardrails cascade down; outcomes and overrides cascade up.

When cross-functional conflicts arise - between a demand plan that expects volume growth and a supply plan that has reduced buffer stock - the system needs a mechanism to surface that conflict explicitly rather than silently propagate one plan's assumptions into another.

We call this the Decision Stream: the place where competing policies are rendered visible and resolvable, with the human's override captured as a first-class signal rather than as friction to drive down.

What the McKinsey piece is actually describing

Read carefully, the McKinsey article is not primarily about AI. It is about the organisational response to a new decision architecture. The five pillars it proposes (business model, team structures, workflows, leadership, culture) are all downstream consequences of one underlying shift: decisions that were previously made by humans, through processes that were slow, expensive, and fragmented, can now be made by agents, through processes that are fast, cheap, and integrated.

That shift is real and the organisational response it requires is genuinely deep. But the response has to be grounded in a coherent theory of how decisions get made, by whom, under what conditions, with what human oversight, and with what feedback mechanisms. Without that theory, the five pillars are a framework for managing change rather than a blueprint for building capability.

SDAM provides the theory. AI·IO·ML provides the governance model. The agentic organisation McKinsey describes is buildable. But it has to be built on a foundation that the consulting narrative, by design, does not lay.

Further reading.

The McKinsey piece this post responds to.

- McKinsey & Company (April 2026). *AI is everywhere. The agentic organization isn't yet.* Companion article and podcast featuring Alexis Krivkovich, McKinsey People & Organizational Performance: mckinsey.com.

Adjacent thinking on the judgement layer.

- Whetton, M. (May 2026). *Code got cheap. Judgement did not.* Medium. The same inversion playing out in engineering: implementation cost has compressed, the judgement layer above it has not, and the structural goal is to make the judgement layer first-class rather than hope a human catches it on every commit.

The SDAM canon.

- Powell, W.B. (2022). *Reinforcement Learning and Stochastic Optimization: A Unified Framework for Sequential Decisions.* Wiley. The book-length treatment of the five-components / four-policy-classes framework. Free PDF available from the author's Princeton site.

- Powell, W.B. (2022). *Sequential Decision Analytics and Modeling: Modeling with Python.* Now Publishers. The teach-by-example monograph with the Python modules that accompany each chapter.
- Sequential Decision Analytics and Modeling (CASTLE Lab, Princeton): the official tutorial site, with lecture notes, examples, and the universal-framework articulation that this post leans on.
- Powell, W.B. (2023). *A Universal Framework for Sequential Decision Problems.* ORMS Today. The short-form version of the argument for a general audience.

Azirella's expression of the framework.

- The Decision Flow Problem: the velocity argument that motivates the whole design, with Stalk's 0.05-to-5 percent rule transposed from product flow to decision flow.
- The world model: the shared state and transition function that every plane reads from and writes to, rather than each module building its own.
- Decision architecture: the five-tier federated policy hierarchy, with context cascading down and outcomes cascading up through the Decision Stream.
- How agents learn: pre-deployment training, live-operations calibration, and continuous improvement, with every override routed back as training signal.
- Operating Knowledge: the four channels through which human judgement deposits into the substrate the agents reason against.

- AI·IO·ML: the full six-verb operating model in three couplets, AI (Automate, Inform), IO (Inspect, Override), ML (Measure, Learn), that closes the loop the 2x2 above only sketches.

See Autonomy in action

Walk through how Autonomy models, executes, monitors, and governs supply chain decisions with autonomous AI agents.

See It Live