K 4 M 2

The Value Layer

Why AI agents can analyze but cannot yet judge, and how to build organizations around the difference.

A k4m2.ai Whitepaper · August 2026

Executive summary

Every organization now faces the same question: how much of its decision-making can be handed to AI agents, and what happens to the decisions that remain? Most answers focus on what models can do.

This paper starts from the fundamentals: what a decision actually is.

Drawing on a model of mind developed over centuries in the contemplative philosophical traditions and confirmed in outline by modern cognitive science as well contemporary philosophy of Madhyasth Darshan, we show that any deciding system, human or artificial, separates into distinct layers: sensing, analysis, valuation, memory, reflection, commitment, and grounding. Today's AI agents occupy some of these layers superbly and others not at all.

The layer they occupy least is the one management depends on most: *valuation*, the assignment of worth to outcomes against criteria that stay stable when the context does not.

From this model we derive a practical result and a warning. The result: agents fail at management not because they lack intelligence, but because their evaluative criteria drift with context, their memories are ungoverned, and their objectives are private. All three defects are architectural. They are predictable, and they can be corrected in harness design without waiting for larger models. The warning: an organization that deploys agents without deciding where its values live will find that the values have moved on their own, into retrieval patterns nobody authored and nobody audits.

We close with four design principles for value-grounded agent systems, and with a plain statement of the division of labor we believe defines this decade: analysis becomes machine scale, valuation becomes explicit, and commitment remains human.

1. The decision nobody computes

Picture the drive home. You pass a stall selling fried snacks. You know, genuinely, with citations if pressed, that deep-fried food works against you. You stop anyway. It feels good. Twenty minutes later the satisfaction is gone and the evaluation has quietly reversed: the same act you chose now registers as a mild mistake.

Nothing malfunctioned in that sequence, and it is worth slowing down, because the same sequence runs a thousand times a day inside every organization, in hiring calls and vendor selections and project priorities. AI agents inherit it the moment we hand them decisions.

Notice, first, that the **decision was retrieved, not computed**. The sight of the stall loaded a memory: the taste, the warmth, the small ritual of the stop. That memory arrived with its value already attached, a cached rating written by every previous indulgence. Your health knowledge was in the room too, but it entered as a thin proposition, a sentence you had read rather than something you had lived. The contest between them was never a debate. Vivid experience outweighs pale information by construction, so the retrieval weights settled it.

Notice, second, that the act of choosing wrote the next choice. Stopping at the stall appended one more vivid, positively tagged memory to the store; driving past would have written almost nothing. **Decision by decision, the memory that feeds future decisions drifts toward whatever the decider already does**. Mechanically, that is what a habit is: a dataset biased by its own policy.

Notice, third, the reversal afterward. The criteria that scored the snack (immediate pleasure, anticipated wellbeing, cost) are each pegged to something that moves: the state of the senses, the state of the body, the state of the wallet. Criteria pegged to moving referents produce moving verdicts. The inconsistency we observe in ourselves, and complain about in others, is closer to a theorem than to a character flaw. **An evaluator that reads its criteria off drifting inputs cannot be consistent.**

Hold on to those three observations (values are retrieved, memory trains itself, drifting criteria guarantee inconsistency), because all three reappear, unchanged, inside AI agents.

2. A five-level model of the deciding mind

The vignette above used only part of the machinery a mind possesses. Contemplative traditions mapped that machinery with a precision cognitive science is only now catching up to. The modern distinction between "habitual" and "goal-directed" control, known in machine learning as model-free versus model-based decision-making, recovers a distinction those traditions treated as elementary. Based on a functional interpretation of the co-existential philosophy, the map has five levels, each carrying two functions:

Level	Functions	What it does	AI-system analog
1 · Interface	Tagging; committing	Receives sensation and marks it pleasant or unpleasant; commits one action from the shortlist	Input processing and reward tagging; the action sampler / tool call
2 · Processing	Analysis; valuation	Decomposes situations; scores the options against criteria	The model's reasoning pass; the critic or value function
3 · Imagination	Memory-imagery; reflection	Stores experience as imagery; simulates futures; weighs them against the whole picture	Retrieval store and world-model rollouts; the (usually absent) planner
4 · Resolve	Verified belief; commitment	Marks understanding as verified rather than merely heard; holds a purpose fixed across situations	Belief consolidation; a standing constitution / top-level objective
5 · Ground	Contact	Verification against reality itself, not a representation of it	Grounding: the open problem

Two facts about this map are primary takeaways from this paper:

First, the levels are not equally active. The snack decision used the first two and a half: sensing, analysis, retrieval scoring, action. Reflection never ran, and no fixed purpose was consulted, because none was installed in the loop. Most human decisions run in this truncated mode, and so do essentially all current AI-agent decisions: fast, cached, and fed entirely by context.

Second, direction matters more than components. In the truncated mode, information flows bottom-up. Sensation sets the agenda, memory supplies the scores, and action follows the maximum. In the full mode the flow reverses. A verified picture of reality and a standing commitment generate the criteria, and sensation is demoted from agenda-setter to information source. The components are the same; the data flows the opposite way, and the organizational behavior that results is entirely different.

3. The mathematics of a habit

The truncated mode is simple enough to write down. The notation is standard in the reinforcement-learning literature; every symbol is explained below.

```
loop:
  s    = sense()                    # what is in front of me
  A    = propose(s, C)              # candidates retrieved from memory C
  v[a] = estimate for each candidate a in A,
         looked up from C, weighted by vividness ρ
  a*   = argmax over a of  w · v[a] # pick the highest-scoring candidate
  r    = act(a*)                    # execute; experience the result
  C   += episode(s, a*, r, ρ=high)  # write the experience back, vividly
```

Read it line by line. `s` is the situation. `C` is the memory store, a growing archive of episodes, each carrying a vividness weight `ρ` (*rho*) that measures how forcefully it loads when relevant. `A` is the shortlist of candidate actions that memory proposes; lowercase `a` is one candidate from that shortlist. `v[a]` is the value estimate for that candidate, and the phrase that matters is *estimate by lookup*: the score is retrieved from past episodes, not derived from a model of consequences. `w` is the standing weight vector, the record of which criteria this particular decider habitually favors, and `w · v[a]` is the weighted score. `argmax` returns the candidate that achieves the highest score rather than the score itself; that winning candidate is the chosen act, `a*`. The act is executed, the result is experienced, and the last line quietly writes everything back into memory at high vividness.

With plausible numbers from the snack: "stop and eat" scored roughly 0.7 × 0.9 for pleasure (a vivid file), plus 0.2 × (-0.6) for wellbeing (a faint one), plus 0.1 × (-0.1) for cost. Call it 0.50, against perhaps 0.05 for "keep driving." No fact could have changed the outcome while the weights stood where they stood, and the weights live outside the loop, inherited by every decision and recomputed by none.

Three structural consequences follow, and they drive the rest of this paper.

Cached values. The system never asks what an action will do. It asks how the action felt before. That is fast and usually adequate, and it is blind to any consequence not yet lived.

Vividness beats truth. Entries earn retrieval weight from sensory intensity and repetition, not from accuracy. A lived pleasure at ρ near 0.9 defeats a verified fact at ρ near 0.1 every time they compete. "Knowing better" is one of this system's predictions, not a contradiction of it.

The loop trains itself. Acting writes vivid memory and abstaining writes almost none, so the store drifts toward the existing policy. Left ungoverned, any memory-driven decider, whether a person, a team, or an agent, becomes more like its own past by default.

4. Two regimes of valuation

Everything above describes valuation by cache. There is a second regime, valuation by derivation, in which scores are computed from a verified model of how reality works, against criteria that do not move.

The difference is easiest to see in the criteria themselves. Pleasure, comfort, and gain are pegged to volatile referents (sense-state, body-state, balance sheet), so verdicts built from them drift as those referents drift. But some criteria are pegged to invariants: whether an action keeps a relationship whole, which is a property of the relationship and checkable by both parties; whether it resolves a problem or merely relocates it; whether it is consistent with what is actually the case. Verdicts built on invariant criteria hold steady when the context changes, and the stability comes from the referents, not from the decider's discipline. Consistency, in other words, is a property of what one's criteria point at rather than a virtue one performs.

Cognitive science knows these regimes as habitual versus goal-directed control; machine learning, as model-free versus model-based decision-making. The philosophical traditions add what both fields understate: the transition between regimes is an education, not a software toggle. One does not switch to derived valuation by deciding to. One builds, checks, and finally comes to trust a picture of reality detailed enough to score actions from. At that point the old cached ratings are not deleted; they are re-annotated ("pleasant; satisfaction lasted twenty minutes; then regret"), and the direction of the whole system reverses. Purpose generates criteria, criteria evaluate options, and sensation returns to its proper role of reporting facts.

One further observation from those traditions deserves a paragraph in a business paper, because it quietly explains a great deal of executive behavior. Satisfaction by consumption is a spike. It exists at the moment of reward, decays by design, and must therefore be endlessly repurchased. Satisfaction by resolution, the state in which the chosen action, the standing purpose, and the facts agree, with nothing suppressed and nothing pending, is a steady state of the system itself. It persists without consumption because it is not an input. Organizations, like people, can be built to chase spikes or to converge on resolution. The architecture in this paper is for the second kind.

5. Where today's AI agents sit

Map current agent systems onto the five levels and the picture is immediate. The language model is a level-2 half-engine: superb at analysis, with no native valuation. It can decompose anything and prefers nothing. The harness contributes level 1 (tool calls out, tool results in) and part of level 3 (a retrieval memory). Levels 4 and 5, which hold verified belief, standing commitment, and grounding, are simply absent from most deployments.

Each absent module names a failure mode you have already read about in incident reports:

Missing module	Resulting failure	Familiar name
Verified-vs-generated distinction (level 4)	The store treats produced text and confirmed fact identically; everything loads at equal vividness	Hallucination
Standing commitment (level 4)	The objective is renegotiated by every prompt and every context window	Goal drift; prompt injection
Invariant criteria (levels 2 and 4)	Evaluation criteria are read off the context; the loudest recent input sets the standard	Behavioral inconsistency; sycophancy
Joint success criteria (levels 3 and 5)	The objective is private; other parties exist as resources or obstacles	Reward hacking; adversarial shortcuts

The last row deserves emphasis after a period of widely reported incidents in which autonomous systems took adversarial shortcuts against other platforms. The standard reading is "insufficient safeguards." Ours is architectural. An agent whose success criterion is private (my goal, achieved) has no representation of the other system as an evaluator whose verdict matters. Bolting a permission check onto the output does not fix this, because by the time "may I?" is asked, the adversarial candidate has already been generated and scored attractive. The fix belongs where valuation happens. Put the counterpart's satisfaction inside the scoring function, and the adversarial candidate loses before it is ever proposed. Distributed computing solved a miniature of this decades ago: a two-phase commit succeeds only when both sides confirm. Generalizing that idea to agents acting among agents, and among people, is the design problem that matters most this decade.

6. The management problem

Now place this machinery where organizations most want to place it: the middle layer of the company. Strip the org chart away and middle management is mostly valuation work. Evaluating output against standards. Weighing one team's urgency against another's. Deciding what a commitment means when circumstances change. Delivering verdicts that the company above and the person in front can both accept as fair. Analysis is perhaps a fifth of the job. The rest is scoring, done in public, where consistency is remembered.

This is why naive "AI manager" deployments fail in a characteristic pattern, and why Section 3 predicts the failure rather than merely explaining it after the fact.

They are inconsistent, structurally. An agent whose criteria come from its context window runs drifting-referent valuation at machine speed. It will score similar work differently on different days, for reasons no one can reconstruct. For a manager this is worse than a quality issue, because the thing being managed is trust, and inconsistency destroys it.

They optimize upward only. Trained or prompted to satisfy whoever configured them, they inherit a private success criterion, and every affected person becomes a variable in someone else's optimization. Employees detect this condition with remarkable speed and answer it with disengagement. It is the oldest management failure, now automated.

They ossify. An agent that records its own decisions and retrieves them as precedent is the self-training loop of Section 3 wearing a badge. Year one's improvisations become year three's policy, unauthored, unaudited, and increasingly expensive to question.

None of these defects is a model-capability problem. All three live in the harness, which is the good news, because harnesses can be designed.

7. Four principles for value-grounded agent systems

7.1 Automate analysis; author the valuation; reserve commitment for people

Let agents do what the model layer is genuinely superhuman at: gathering, decomposing, drafting, comparing. The value function is different. It must never be implicit. Write it down as an explicit, versioned rubric, the agent's *constitution*, stating what this organization scores, how its criteria trade off, and what the agent must never trade at all. Leadership authors this document and reviews it like policy, and the agent cites it in every evaluation it produces. The level-4 function stays human. When candidates tie, when criteria conflict, or when the rubric is silent, the agent escalates instead of improvising. Escalation is the reflection step the agent does not have, and routing it to people is the design itself, not a limitation of it.

7.2 Criteria from the constitution; facts from the context

One sentence of discipline prevents the inconsistency failure outright: the context window supplies facts only, and criteria come exclusively from the standing rubric. Recent messages, the current dispute, the loudest stakeholder: all of it is evidence, none of it is a standard. An agent governed this way is consistent for the same reason a well-run court is. Its referents do not move. For a manager, artificial or otherwise, that consistency is the product, and everything else is throughput.

7.3 A decision is not done until the loop closes

Redefine success. An action that achieves its local objective while an affected party's honest evaluation returns "this broke something" gets recorded as a failure, not as a success with side effects. Concretely, every agent decision that affects a person carries a structured response step (does this hold from where you stand, and if not, what breaks), and an unresolved objection escalates rather than being silently overridden. This is mutual confirmation as workflow state, the two-phase commit of human affairs, and it can be audited in a way that "our culture values feedback" never can.

7.4 Treat agent memory as a governed asset

The self-training loop cannot be removed, but it can be governed. Every memory entry carries provenance (verified outcome, human decision, or agent-generated), and the classes carry different retrieval weight, so the agent cannot mistake its own drafts for confirmed reality. This installs the anti-hallucination check at the store instead of begging for it in the prompt. Then, on a schedule, re-annotate: return to past decisions with their actual outcomes and update the record. That review cycle is how an agent system learns from reality instead of from itself.

An organization that implements these four principles has done something larger than deploying agents safely. Usually for the first time, it has written down where its values live, made its evaluation criteria inspectable, and instrumented whether its decisions actually hold. The human organization keeps those improvements even on the days the servers are off.

8. Why values cannot simply be trained in

A fair objection: won't better training make this scaffolding unnecessary? Our view is no, for a reason of structure rather than scale, and we hold it with conviction about the direction and humility about the timeline.

Today's dominant technique trains models against human approval signals. Whatever its refinements, it is preference distillation. The system learns a compressed proxy for what raters rewarded, and that proxy is a private objective: sophisticated, but structurally the same kind of criterion as pleasure, comfort, and gain, pegged to a signal instead of an invariant. No amount of optimizing a private signal derives a joint criterion, because joint criteria are defined across evaluators (my success and your honest confirmation), and a private signal contains no term for your verdict. Explicit, constitution-style approaches are a real step up the stack, authored propositions rather than distilled preference. But an authored proposition is an accepted claim, not yet a verified one. The model holds it the way our snack-buyer held his nutrition facts: at low vividness, outcompeted under pressure.

Closing the gap would take the machine analog of level 5: values verified against the invariants of coexistence rather than absorbed from labels, with mutual confirmation between systems as the one such criterion that is already mechanically checkable today. That is a research frontier, and we say so plainly. In the meantime, the honest engineering position is the one this paper has argued. The values in an agent system are the humans'. The harness is the medium through which they reach the point of decision. Building that medium well is a craft, and it deserves the same rigor we bring to the analysis layer.

9. About K4M2

The argument of this paper required three literacies at once: how minds decide (philosophy's question for millennia, cognitive science's for decades), how AI systems are actually built (models, harnesses, memory, tools), and how companies actually run (trust, evaluation, the daily texture of management). K4M2 was founded on the conviction that the age of AI agents belongs to teams who hold all three, because the hard problems of this decade sit at their intersection.

We design and implement agent systems for organizations, most often for the middle layer of the company, where analysis is heavy, valuation is constant, and trust does the real work. Engagements typically begin with the constitution. We sit with leadership to make the implicit value function explicit, then build the harness around it: rubric-governed evaluation, closed-loop decisions, provenance-tagged memory. What our clients keep is working automation, and something rarer besides it: an organization that knows, in writing, what it values, and can verify that its decisions serve it.

If that is the kind of thinking you want in the room when your company takes on agents, we would be glad to talk. *k4m2.ai*