Outcome-Driven Operating Model in the Age of AI

How AI Acceleration Changes the Limits of Time-Based Control

Sean Sweeney

First Published: December 13, 2025
Latest Revision: April 15, 2026

Contents

Author's Note

This paper builds directly on ideas first explored in *AI-Driven Development & the Future of Scrum and Agile: A Thought Experiment*. That earlier work examined how AI affects Agile practice and argued that accelerating delivery increases the need for discipline, leadership, and sound systems thinking.

Since then, my own use of AI in development work has deepened, moving from occasional assistance to sustained integration across design, analysis, and delivery. That shift clarified something the earlier paper only hinted at. While Agile principles remain valid, the operating assumptions that underpin time-based control begin to strain once AI materially accelerates the pace of change.

This paper reflects that progression in thinking. It does not revisit Agile practice in detail, nor does it propose incremental adaptations to existing frameworks. Instead, it examines the implications of AI acceleration at the operating-model level and introduces the Outcome-Driven Operating Model as a response to those implications.

The **intent of this paper is not to prescribe immediate adoption**, but to make visible a structural shift that organizations will increasingly encounter as AI becomes embedded in everyday work. Readers familiar with the earlier paper may recognize the continuity. What has changed here is the level of abstraction and the clarity of the conclusion.

The claim that ODOM optimizes learning integrity under acceleration is evaluated in Appendix A, which includes both the mechanism of the claim and a proposed empirical design for testing it.

TL;DR

- AI decouples calendar time from the amount of change teams can produce.
- As AI accelerates delivery, Sprint-based timeboxes stop reliably containing learning and attribution.
- This is not a Scrum or team failure. It is an operating model mismatch.
- When too much change happens too quickly, organizations lose causal clarity even as output increases.
- The scarce resource becomes attribution, not execution capacity.
- Existing Agile methods each address part of this challenge. Kanban provides flow and WIP discipline. XP provides engineering feedback quality. Scrum provides cadenced inspection. None of them individually address how intent should be bounded and evidence should govern decisions at the operating model level under AI acceleration.
- The Outcome-Driven Operating Model (ODOM) replaces time-based control with outcome-based control.
- ODOM constrains intent rather than time, preserving learning while allowing continuous, AI-accelerated delivery.
- Assessment separates truth from direction, enabling executive decisions without executive interference.
- ODOM events are structured conversations, each with a specific purpose: aligning intent, observing signals, interpreting evidence honestly, and improving capability.
- This is an evolution of Agile discipline, not a rejection of Scrum.
- Constraining intent to a single Outcome at a time is what creates organizational agility. When learning is preserved, changing direction becomes an evidence-based decision rather than a political negotiation.
- AI makes it possible to move faster than the organization can understand what is happening.

Executive Summary

As AI dramatically accelerates software delivery, it strains a core assumption behind Sprint-cadenced operating models, namely that timeboxes like Sprints can reliably contain learning, attribution, and decision-making. When AI makes it cheap to explore many options and ship many changes within the same calendar window, organizations generate more activity than they can meaningfully understand. Causal clarity erodes quietly, and confidence can rise even as understanding falls. Removing timeboxes alone, such as by shifting to continuous flow, does not solve this problem. The unit of control must move upward from time and work items to explicitly bounded intent supported by evidence.

This paper treats the challenge as an operating model problem. Team-level discipline matters, yet it cannot carry organizational learning under sustained AI acceleration. Time-based control worked when change was expensive and learning had to be batched. Under sustained acceleration, time becomes a weak boundary for learning. Attribution becomes the scarce resource, and without it, responsible decision-making collapses.

The paper introduces the Outcome-Driven Operating Model (ODOM), an operating model designed to preserve learning, attribution, and decision quality as AI accelerates delivery. ODOM replaces time-based control with outcome-based control. Instead of constraining work by calendar intervals, it constrains intent by focusing each team on a single, explicit Outcome at a time during Build. Work flows continuously, AI can be used extensively, and signals are observed continuously. The unit of control moves upward from time and work items to bounded intent supported by evidence.

Outcomes are shaped through Discovery, where teams and stakeholders align on what change is being attempted, what would constitute evidence of that change, and what is deliberately excluded. By the time an Outcome reaches Kickoff, it is fully formed, specific, observable, and bounded enough to function as a genuine learning container. Kickoff confirms intent and commitment rather than resolving ambiguity. The ODOM Loop is expressed as Kickoff, Build, Assessment, and Reflection. Outcomes move from Build into Assessment for evaluation, and several Outcomes may be evaluated in parallel as signals mature. Assessment happens when evidence is sufficient to support a continue, stop, or pivot decision, not when a calendar window ends. Teams state what the evidence supports without negotiation, and leadership makes portfolio-level decisions based on those conclusions rather than on activity, plans, or reassurance. Reflection improves the team's approach. This structure is what creates organizational agility. When learning is preserved and evidence is clear, changing direction becomes a natural consequence of understanding rather than a costly disruption.

ODOM adapts Deming's PDCA, Plan Do Check Act, as the backbone of its learning loop. It also draws on ideas from across the Agile landscape. Lean's WIP discipline, Kanban's flow mechanics, XP's commitment to fast feedback, Evidence-Based Management's insistence on empirical grounding, and OKR thinking's focus on outcomes over outputs all contribute to its foundation. What none of these individually provide is a coherent operating model that constrains intent, structures evidence, and separates truth from direction at the organizational level. While this paper focuses on the operating model layer, ODOM depends on complementary disciplines at adjacent layers. Jobs to Be Done sharpens the definition of an Outcome by grounding it in observable behavior change. The Theory of Constraints reinforces the need to limit how much intent is in motion by recognizing that system throughput is governed by a small number of bottlenecks. DevOps enables continuous, low-friction execution so that outcome-based control does not introduce delivery drag. Site Reliability Engineering defines the operating boundaries that allow rapid change without destabilizing the system. ODOM does not replace these disciplines. It integrates them at the level where intent, evidence, and decision-making must remain coherent under acceleration.

ODOM is an evolution of operating discipline designed for an environment where execution is fast but understanding is fragile. Its purpose is to preserve learning integrity, accountability, and decision quality as AI adoption moves from experimentation to everyday practice. The key leadership challenge is recognizing the tipping point where time-based control no longer provides clarity, and evolving the operating model before speed outpaces understanding.

Why Operating Models Matter

Every organization, whether it names it or not, operates according to an operating model. An operating model is the set of structural assumptions an organization makes about how work should move, how decisions should be made, how learning should occur, and how risk should be managed. Most operating models are never written down. They are inferred from funding cycles, approval paths, reporting structures, planning rhythms, and what leaders pay attention to.[1]

When an operating model fits the conditions of the environment, the organization feels steady. Decisions feel justified. Progress feels legible. When it does not, leaders experience a persistent sense of friction. Teams appear busy but impact is hard to explain. Confidence increases while understanding decreases. More information is available, yet decision quality does not improve.

Historically, operating models change only when the environment changes enough to expose their limits. The assembly line changed operating models for manufacturing. Electrification changed them again. The internet did the same for information work. Each shift did not eliminate management discipline. It forced a different kind of discipline.[2,3,4]

Artificial intelligence represents the next such shift. Not because it is intelligent in a human sense, but because it alters the economics of action and interpretation at the same time. AI reduces the cost of producing change and the cost of analyzing change. That combination is rare, and it is what makes this moment different from earlier tooling advances.

It is important to be explicit about where most organizations are today. In many places, AI is still treated as an exploratory capability. Individuals experiment. Teams try it in isolated contexts. Leadership observes. This is normal. It mirrors how cloud computing, DevOps, and data platforms first entered organizations.

This paper is not arguing that current operating models are broken today. It is arguing that the future direction is already visible, and that waiting until AI is fully embedded before rethinking operating models will put organizations in a reactive position rather than a

---

[1] Weill, P., & Ross, J. W. (2009). *IT savvy: What top executives must know to go from pain to gain*. Harvard Business Press.

[2] Womack, J. P., Jones, D. T., & Roos, D. (1990). *The machine that changed the world: The story of lean production*. Free Press.

[3] David, P. A. (1990). The dynamo and the computer: An historical perspective on the modern productivity paradox. *American Economic Review*, 80(2), 355–361. https://www.jstor.org/stable/2006600.

[4] Brynjolfsson, E., & Hitt, L. M. (2000). Beyond computation: Information technology, organizational transformation and business performance. *Journal of Economic Perspectives*, 14(4), 23–48. https://doi.org/10.1257/jep.14.4.23

deliberate one. The purpose of this paper is to explain why existing operating models, including those built around Scrum, will eventually strain under widespread AI adoption, and why a different operating model will be required to preserve learning, attribution, and responsible decision-making as speed increases.

Implicit Operating Model	Explicit Operating Model
Time	Intent
Plans	Evidence
Commitments	Decisions
Reporting	Learning

Table 1 - Implicit vs. Explicit Operating Models

The Nature of the Change AI Introduces

Most discussions about AI in software focus on productivity. Faster code. Fewer manual steps. Reduced cycle times. Those effects are real, but they are not the most important change AI introduces.

The deeper change is that AI decouples calendar time from amount of change.[5,6,7] In traditional software delivery, the number of meaningful changes a team could attempt in a given period was limited by human effort. Planning cadences, approval structures, and review cycles evolved to manage that constraint. Learning was batched because it had to be.

AI weakens that constraint. A team can now explore more options, generate more variations, and implement more changes inside the same window of time. Analysis that once followed delivery can now run alongside it. Signals can be surfaced continuously rather than periodically.[8]

This does not remove uncertainty, but it changes form. Instead of uncertainty being driven by lack of information, it becomes driven by excess information and insufficient attribution.

---

[5] Noy, S., & Zhang, W. (2023). Experimental evidence on the productivity effects of generative artificial intelligence. *Science*, 381(6654), 187–192. https://doi.org/10.1126/science.adh2586

[6] Brynjolfsson, E., Li, D., & Raymond, L. R. (2023). *Generative AI at work* (NBER Working Paper No. 31161). National Bureau of Economic Research. https://www.nber.org/papers/w31161.

[7] Dell'Acqua, F., McFowland, E., & Mollick, E. (2023). *Navigating the jagged technological frontier: Field experimental evidence of the effects of AI on knowledge worker productivity and quality* (Working paper). Harvard Business School. https://www.hbs.edu/faculty/Pages/item.aspx?num=64700.

[8] Forsgren, N., Humble, J., & Kim, G. (2018). *Accelerate: The science of lean software and DevOps: Building and scaling high performing technology organizations*. IT Revolution Press.

When many things change quickly, understanding which change mattered becomes harder, not easier. Without structural discipline, acceleration creates confusion rather than clarity.

The real bottleneck is system sensemaking under rapid change.[9] While large-scale empirical data on AI-native operating models is still emerging, the structural dynamics described here are already visible wherever AI materially accelerates delivery.

What Existing Methods Get Right and Where the Gap Remains

Before introducing an alternative operating model, it is important to be explicit about what existing Agile methods already contribute to the problem described above. Any operating model designed for AI-accelerated environments will build on ideas that have been developing across the Agile landscape for decades, and those intellectual debts should be visible before a new model is introduced.

Lean thinking established that limiting work in progress and reducing batch sizes are fundamental to flow, quality, and learning.[10,11] The principle of "stop starting, start finishing" is one of the most powerful ideas in modern management. Any operating model that constrains intent to a single outcome per team is applying a WIP limit at the level of strategic intent.

Kanban brought continuous flow, explicit WIP limits, and evidence-driven policies to software delivery.[12] It did so without prescribing timeboxes, which means it already addresses one of the central problems this paper identifies. The Kanban Maturity Model[13] extends these ideas to organizational governance, including fitness-for-purpose criteria and cadence-tuned feedback loops. Any operating model that proposes continuous flow under outcome-based governance inherits directly from this foundation.

Extreme Programming (XP)[14] established that sustainable speed depends on engineering discipline. Test-driven development, continuous integration, pair programming, and simple design are what make rapid delivery possible without accumulating technical debt. These practices become more important, not less, as AI accelerates the volume of code

---

[9] Meadows, D.H. (2008). *Thinking in systems: A primer*. Chelsea Green Publishing.
[10] Womack, J. P., Jones, D. T., & Roos, D. (1990). *The machine that changed the world: The story of lean production*. Free Press.
[11] Reinertsen, D. G. (2009). *The Principles of Product Development Flow: Second Generation Lean Product Development*. Celeritas Publishing.
[12] Anderson, D. J. (2010). *Kanban: Successful Evolutionary Change for Your Technology Business*. Blue Hole Press.
[13] Anderson, D. J., & Carmichael, A. (2016). *Essential Kanban Condensed*. Lean Kanban University Press.
[14] Beck, K. (2000). *Extreme Programming Explained: Embrace Change*. Addison-Wesley.

produced. Research on high-performing technology organizations has demonstrated empirically that delivery performance, driven by continuous delivery, fast feedback, and lean management practices, predicts organizational outcomes.[15] That finding establishes that engineering discipline and evidence-based governance must work together. Any operating model that assumes continuous delivery must also assume a baseline of engineering maturity sufficient to support it. Without that foundation, outcome-based control cannot function.

Scrum[16] brought cadenced inspection and adaptation to environments that lacked feedback discipline. Its contribution was not speed but legibility. The Sprint created a reliable moment where organizations were required to stop, look, and decide. That contribution remains valuable wherever cadence-based control still fits the pace of change.

Evidence-Based Management (EBM)[17] and OKR thinking[18] both push organizations to measure outcomes rather than outputs and to ground decisions in evidence rather than opinion. These ideas establish the foundation for treating observable behavior change as the measure of progress.

Each of these methods addresses a real constraint. Lean and Kanban address flow and WIP. XP addresses engineering feedback quality. Scrum addresses cadence and inspection discipline. EBM and OKRs address evidence and outcome orientation.

What none of them individually address is the specific problem AI acceleration creates: that when delivery cost collapses, the binding constraint shifts from execution capacity to attribution capacity, and the operating model must be redesigned around that shift. Kanban limits WIP but does not prescribe how intent should be bounded or how evidence should trigger decisions. XP ensures technical quality but does not govern how the organization selects and evaluates what change to attempt. Scrum provides cadence but ties learning to the calendar. OKRs define outcomes but do not provide an operating structure for how teams pursue them or how truth is separated from direction in governance.

---

[15] Forsgren, N., Humble, J., & Kim, G. (2018). *Accelerate: The science of lean software and DevOps: Building and scaling high performing technology organizations*. IT Revolution Press.
[16] Schwaber, K., & Sutherland, J. (2020). *The Scrum Guide*. Scrum Guides. https://scrumguides.org/scrum-guide.html.
[17] Scrum.org. (2020). *Evidence-Based Management Guide*. Scrum.org. https://www.scrum.org/resources/evidence-based-management-guide.
[18] Doerr, J. (2018). *Measure what matters: How Google, Bono, and the Gates Foundation rock the world with OKRs*. Portfolio.

The components are strong. The gap shows up in the integration. Jobs to Be Done sharpens the definition of an Outcome by grounding it in observable behavior change.[19] The Theory of Constraints reinforces the need to limit how much intent is in motion at once by recognizing that system performance is governed by a small number of bottlenecks.[20] DevOps enables continuous, low-friction execution so that outcome-based control does not introduce delivery drag.[21] Site Reliability Engineering defines the operating boundaries that allow rapid change without destabilizing the system.[22] These disciplines do not replace the need for an operating model. They clarify the adjacent conditions ODOM depends on in order to preserve learning and decision quality under acceleration. What none of these individually provide is a coherent operating model that constrains intent, structures evidence, and separates truth from direction at the organizational level.

Why This Forces a Rethink of Scrum

Scrum is a team-level framework. It was never designed to govern organizational learning at scale, and it does not claim to solve portfolio governance, funding allocation, or cross-team coordination.[23] This paper is not arguing that Scrum's design is flawed. It is arguing that most organizations use Scrum as though it were an operating model, and that under AI acceleration, that implicit operating model breaks down.

In practice, Scrum often becomes the primary organizing logic for how work is selected, how progress is reported, how learning is assumed to occur, and how decisions are paced. When that happens, the Sprint cadence functions not just as a team-level discipline but as the organization's default boundary for learning and decision-making. That extension was never part of Scrum's design intent, but it is the lived reality in thousands of organizations.

Scrum was designed as a response to an earlier form of uncertainty. It assumed that learning needed to be protected by cadence. The Sprint created a bounded window in which teams could plan, act, observe, and reflect. That rhythm allowed organizations to

---

[19] Christensen, C. M., Hall, T., Dillon, K., & Duncan, D. S. (2016). *Competing against luck: The story of innovation and customer choice*. Harper Business.
[20] Goldratt, E. M. (1984). *The goal: A process of ongoing improvement*. North River Press.
[21] Forsgren, N., Humble, J., & Kim, G. (2018). *Accelerate: The science of lean software and DevOps: Building and scaling high performing technology organizations*. IT Revolution Press.
[22] Beyer, B., Jones, C., Petoff, J., & Murphy, N. R. (2016). *Site reliability engineering: How Google runs production systems*. O'Reilly Media.
[23] Schwaber, K., & Sutherland, J. (2020). *The Scrum Guide*. Scrum Guides. https://scrumguides.org/scrum-guide.html.

replace long-range prediction with short-range learning while maintaining managerial legibility.

As long as the volume of change inside that window remained manageable, the Sprint functioned as an effective control mechanism. AI adoption changes the volume of change that can occur inside the same window. Over time, this erodes the Sprint's ability to serve as a reliable learning boundary. The result is not immediate failure. It is gradual distortion. Reviews become summaries rather than discovery. Backlogs accumulate changes faster than they can be reasoned about. Confidence increases because activity is visible, while causal understanding quietly declines.

This is the point at which the implicit operating model built around Scrum must evolve. To understand what must evolve, it is first necessary to see how time-based control degrades under sustained AI acceleration.

When Time-Based Control Stops Controlling

Scrum's strength has always been that it creates a disciplined loop between intent, action, and learning. The Sprint exists to prevent organizations from running too far ahead of their understanding. It limits how much can change before people are required to stop, look at what happened, and decide what it means.[24]

That discipline matters. Without it, most organizations revert to push-based behavior. Work piles up, priorities blur, and learning becomes incidental rather than deliberate. Scrum succeeded not because it made teams faster, but because it made learning visible and decisions harder to avoid.

AI rarely breaks Scrum overnight. It gradually weakens the protection time-based operating models provide for learning as the pace and volume of change increase.

Early in AI adoption, this is subtle. A team uses AI to speed up coding tasks. A developer explores alternatives more quickly. A backlog item is completed sooner than expected. The Sprint still holds. The Review still makes sense. The Retrospective still feels grounded.

As AI use expands, the character of the work changes. More options are explored. More changes are attempted. More "small improvements" land in production within the same Sprint window. As AI moves from assistive tools to agent-based execution and orchestration, this parallelism increases again, often without corresponding increases in

---

[24] Schwaber, K., & Sutherland, J. (2020). *The Scrum Guide*. Scrum Guides. https://scrumguides.org/scrum-guide.html.

human sensemaking. Signals begin to move before the Sprint ends, after it ends, and sometimes in ways that no longer align cleanly with the Sprint Goal that was originally articulated.

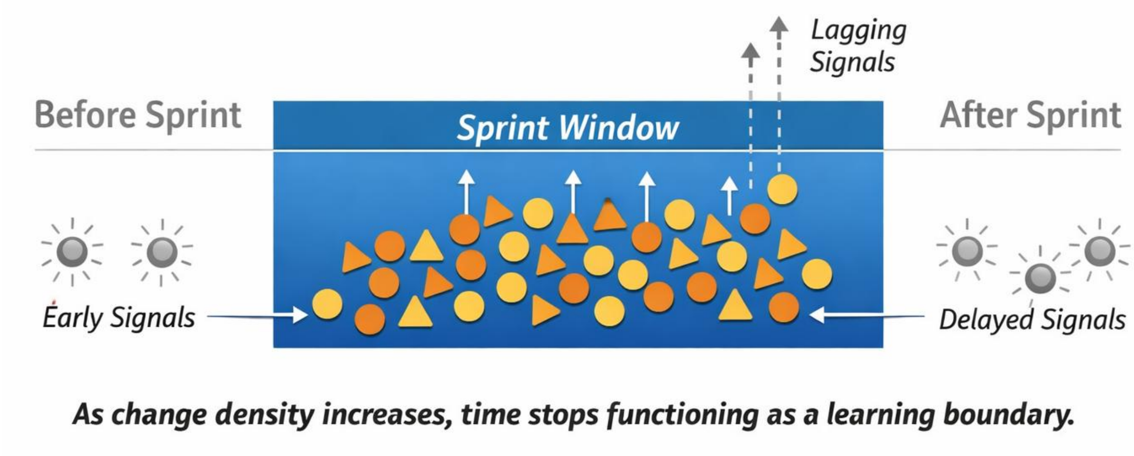

Figure 1 - Change Density and the Limits of Timeboxes

At this point, teams begin compensating without realizing it. They narrate coherence after the fact. They group changes together to explain outcomes that are no longer attributable to a single decision. They adjust Sprint Goals to be broader so they can accommodate what actually happened. None of this feels dishonest. It feels practical.

But this is the moment where learning integrity begins to erode. Scrum assumes that a Sprint Goal functions as a hypothesis and that the Increment provides evidence against that hypothesis. When AI enables many hypotheses to be exercised implicitly inside the same window, the Sprint Goal can no longer serve that function reliably. It becomes a theme rather than a test.

The Sprint Review then shifts its role. Instead of asking "*what did we learn about this bet*," it increasingly answers, "*what did we get done*." Stakeholders still see progress. Teams still feel productive. Leadership still sees motion. What quietly disappears is causal clarity.

Given the incentives and the pace, this drift is predictable. A well-disciplined Scrum team can resist this erosion for a time. Sprint Goals that function as genuine hypotheses, Reviews that interrogate learning rather than demonstrate output, and Retrospectives that address systemic patterns rather than tactical fixes all help. The Scrum Guide explicitly positions the Sprint Review as an inspection of the outcome of the Sprint, not a demo. The degradation described here can be avoided within Scrum's design, yet it becomes harder to sustain as AI compresses delivery cycles and multiplies the number of concurrent changes inside a single timebox. Under acceleration, the strain shifts from team practice to governance. The organizational systems built around Sprint cadence begin to fail in ways that team discipline alone cannot fix.

Goodhart's Law[25] becomes more pronounced here. As output accelerates, metrics that once approximated learning begin to dominate attention. Velocity, predictability, and throughput feel reassuring because they move consistently upward. But they no longer reflect understanding. They reflect activity.[26]

Little's Law[27] asserts itself as well. As AI increases throughput, work in progress expands unless deliberately constrained. This is where Kanban's contribution becomes visible. Kanban makes this dynamic visible by enforcing explicit WIP limits[28]. Many organizations running Sprint-based governance rarely apply WIP discipline at the level of strategic intent. Work item WIP limits and Sprint WIP limits control execution congestion. Intent WIP limits control how much partially understood change stays in motion. The system starts carrying more change than it can interpret.

From the outside, the organization looks healthy. From the inside, decision-making becomes harder. Leaders sense that more is happening but are less confident explaining why outcomes are moving or not moving. Teams feel pressure to keep up with their own output. Retrospectives generate improvements that feel increasingly tactical rather than systemic.

This is the condition many organizations will encounter as AI adoption deepens. Not chaos. Not collapse. But a growing gap between activity and understanding. The claim is not that teams cannot learn within Sprints, but that under sustained acceleration the Sprint boundary no longer reliably bounds attribution across overlapping interventions.

---

[25] Goodhart, C. A. E. (1975). Problems of monetary management: The U.K. experience. In *Papers in Monetary Economics* (Vol. 1). Reserve Bank of Australia. (Goodhart's Law)
[26] Johnson, D. (2002). The unintended consequences of qualitative performance metrics. *Systems Research and Behavioral Science*, 19(3), 249–260. https://doi.org/10.1002/sres.431
[27] Little, J. D. C. (1961). *A proof for the queuing formula: $L = \lambda W$*. Operations Research, 9(3), 383–387. https://doi.org/10.1287/opre.9.3.383 (Little's Law)
[28] Anderson, D. J. (2010). *Kanban: Successful Evolutionary Change for Your Technology Business*. Blue Hole Press.

Why This Is an Operating Model Problem, Not a Team Problem

It is tempting at this stage to look for local fixes. Improve Sprint discipline. Tighten backlog refinement. Add more structure to Reviews. Introduce new metrics to regain clarity. In practice, these responses increase friction without restoring learning.

The reason is that the problem no longer sits at the team level. Scrum governs how a team organizes work within a bounded context. It does not govern how an organization chooses what change is worth making, how much change is safe to attempt at once, or how evidence should be interpreted when signals arrive continuously rather than episodically.[29]

Those questions belong to the operating model. Operating models define the unit of intent an organization works toward. They define the unit of learning an organization trusts. They define where decisions are made and on what basis.[30] Scrum never claimed to answer those questions at scale. The problem is not Scrum's design.[31] It is the common organizational pattern of treating Sprint-based governance as though it provides operating model coverage. Under AI acceleration, that implicit extension breaks down.

As AI lowers the cost of change, the unit of control must move upward, from timeboxed execution to explicitly bounded intent. Without that shift, organizations accumulate learning debt. They ship faster while understanding less. They respond to signals later because signals are harder to interpret. They make larger decisions with weaker evidence.

This is where systems thinking becomes unavoidable. Deming[32] warned that without understanding variation and systems behavior, organizations confuse noise for signal and tamper with their own processes. AI dramatically increases the amount of variation visible in the system. Without a structure designed to interpret it, leaders will react more, not better.

---

[29] Overby, E., Bharadwaj, A., & Sambamurthy, V. (2006). Enterprise agility and the enabling role of information technology. *European Journal of Information Systems*, 15(2), 120–131. https://doi.org/10.1057/palgrave.ejis.3000600

[30] Weill, P., & Ross, J. W. (2004). *IT governance: How top performers manage IT decision rights for superior results*. Harvard Business School Press.

[31] Schwaber, K., & Sutherland, J. (2020). *The Scrum Guide*. Scrum Guides. https://scrumguides.org/scrum-guide.html.

[32] Deming, W. E. (1994). *The new economics for industry, government, education* (2nd ed.). MIT Press.

Conway's Law[33] reinforces the risk. When organizational structures remain oriented around projects, backlogs, or functions, AI accelerates misalignment rather than correcting it. The system architecture will mirror that confusion at speed.[34,35,36]

Gall's Law[37] reminds us that complex systems must evolve from simpler systems that worked. Attempting to scale AI-driven delivery without first stabilizing how learning and decision-making occur invites fragility.

Scrum still matters, but many organizations treat the Sprint cadence as the primary organizing logic for learning and decisions. That approach becomes insufficient in an AI-accelerated environment.

The Shift That Becomes Necessary

As AI adoption crosses a tipping point, organizations must change the question they are optimizing for. Instead of asking, "*How much can we deliver in this timebox?*" they must ask, "*What change are we deliberately trying to create, and how will we know whether it happened?*" On paper the shift looks small, but it changes how learning and decisions happen.

It requires moving from time-bound control to outcome-bound control. It requires constraining work not by calendar intervals, but by intentional, observable change. It requires structuring work so that attribution is preserved even as speed increases.

Sprint-Based	Outcome-Based
Timebox	Intent boundary
Review Cadence	Evidence-triggered decision
Output Focus	Behavior change
Calendar-driven	Learning-driven

Table 2 - Timebox Control vs. Outcome Control

[33] Conway, M. E. (1968). How do committees invent? *Datamation*, 14(4), 28–31. https://www.melconway.com/Home/pdf/committees.pdf. (Conway's Law)
[34] Weill, P., & Ross, J. W. (2009). *IT savvy: What top executives must know to go from pain to gain*. Harvard Business Press.
[35] Ross, J. W., Beath, C. M., & Mocker, M. (2019). *Designed for digital: How to architect your business for sustained success*. MIT Press.
[36] Westerman, G., Bonnet, D., & McAfee, A. (2014). *Leading digital: Turning technology into business transformation*. Harvard Business Review Press.
[37] Gall, J. (2002). *The systems bible: the beginner's guide to systems large and small: being the third edition of Systemantics*. General Systemantics Press. (Gall's Law)

In this context, attribution refers to the organization's ability to connect deliberate actions to observed outcomes with enough confidence to support learning and future decisions.

This is the gap the Outcome-Driven Operating Model (ODOM) is designed to address.

ODOM draws directly on the methods surveyed above. Lean's WIP discipline, Kanban's flow mechanics, XP's engineering rigor, and the evidence orientation of EBM and OKR thinking all contribute to its foundation. ODOM's constraint of one Outcome per team is, at its core, a WIP limit applied at the level of strategic intent. Its insistence on observable behavior change as the measure of progress comes directly from evidence-based and outcome-oriented thinking. None of these methods alone provides integration at the operating model level, where intent must be constrained, evidence must govern decisions, and truth must be separated from direction. ODOM integrates these ideas into an operating model layer that becomes critical under AI acceleration.

ODOM begins from the recognition that, under AI acceleration, attribution is the scarce resource. Without attribution, learning collapses. Without learning, speed becomes dangerous. Without speed, organizations lose relevance.

There is an important consequence to this shift that is easy to overlook. Constraining intent does not reduce agility. It creates it.

Agility, in any meaningful sense, is the ability to change direction quickly and responsibly.[38] Speed alone does not provide that ability. An organization can move fast and still struggle to change direction if it cannot explain what its previous direction actually produced. Without that understanding, every pivot is a guess. Leaders hesitate because evidence is too thin to choose among options with confidence.

This is the condition AI acceleration creates if left unstructured. Organizations generate more change, surface more signals, and accumulate more activity, yet the ability to interpret any of it degrades. Changing direction becomes harder not because of rigid commitments, but because no one can clearly articulate what the current direction is actually achieving.

Outcome-based control addresses this directly. When the organization constrains intent to a single Outcome, with evidence observed continuously and Assessment producing an

[38] Conboy, K. (2009). Agility from first principles: Reconstructing the concept of agility in information systems development. *Information Systems Research*, 20(3), 329–354. https://doi.org/10.1287/isre.1090.0236

honest conclusion, something important becomes available that most operating models struggle to provide. Clarity about what just happened.

That clarity is what makes agility real. When Assessment concludes that the intended change is occurring, leadership can decide to deepen the investment or redirect effort with confidence. When Assessment concludes that the change is not occurring, the organization does not need to debate interpretations or negotiate narratives. The evidence has already spoken. The decision follows naturally.

The pivot is inexpensive not because less was committed to, but because more was understood. Learning was preserved. Attribution was maintained. The organization knows what it tried, what happened, and what the evidence supports doing next.

This is agility in its most honest form. Not the ability to do many things at once, but the ability to learn fast enough to choose well.

An Executive Example: Attribution Loss in Linear TV Scheduling

To make the attribution problem concrete before introducing the model designed to address it, consider the following scenario. Consider a linear TV network responsible for scheduling titles across a portfolio of channels and dayparts. The organization is under pressure to improve ad performance while maintaining audience retention. Ratings softness is appearing in certain time slots. Advertisers are asking sharper questions about reach and consistency. Nothing is broken, but leadership wants improvement.

Several levers are available. Programming teams adjust which titles are scheduled in specific dayparts. Scheduling teams experiment with lead-in and lead-out pairings. Other teams make changes to how advertising is scheduled and structured within programs. Engineering teams adjust playout timing and metadata alignment. AI-assisted tools make it easier to simulate schedules, analyze historical performance, and propose alternative lineups.

Changes begin to roll out. A title is moved earlier in the evening to capture a broader audience. Another is shifted to strengthen a weaker block. Adjustments are made to commercial placement and load within programs. AI-generated recommendations influence which titles are promoted more heavily. All of this happens over the course of several weeks, often overlapping.

Ratings tick up modestly in some slots. Ad performance improves in others. Leadership reviews the data and sees signs of progress. The organization continues refining the schedule, because there is no obvious reason not to. Each change seems directionally correct.

Months later, performance becomes uneven again. Some dayparts perform better than before. Others regress. When asked what specifically drove the earlier improvements, the answers are fragmented. One team points to title placement. Another points to ad load changes. A third points to seasonal effects. No one is wrong, but no one can explain the system's behavior with confidence.

The organization did not fail to execute. It failed to preserve learning.

Change overload made it hard to see what actually drove the outcome. This is attribution loss in linear scheduling. It happens quietly. It is often mistaken for the inherent complexity of media systems. AI accelerates it by making it cheap to explore many scheduling variations simultaneously.

Now imagine the same situation approached with a different operating constraint. Leadership identifies a single outcome to pursue first. Not "*improve ratings*" and not "*optimize the schedule*," but a specific, observable change tied to titles and time slots. For example, increasing audience retention across a defined prime-time block by improving how viewers flow from one title to the next.

That outcome becomes explicit intent. A single team is accountable for that outcome. Other schedule optimizations are deliberately paused. Other teams continue pursuing other Outcomes in parallel, provided their interventions do not change the signals used to evaluate this block. Titles outside the defined block are left unchanged. AI is used heavily, but narrowly, to analyze historical title transitions, simulate alternative sequencing, and test limited adjustments within that block.

Changes still happen. But they are constrained by intent.

When retention improves, the organization can explain why. When it does not, the organization learns something specific. When leadership decides to move on, it does so with evidence that can be reused in other parts of the schedule.

Bounded learning makes the difference. Tools matter less than the boundary that preserves causality.

A Practitioner Example: How ODOM Emerged from AI-Driven Development

ODOM is an operating model I developed while practicing AI-driven development firsthand, specifically while building *AgilePro.AI*, an AI-enabled tool for Agile teams. Early in that work, my assessment aligned with what I described in *AI-Driven Development & the Future of Scrum and Agile*: that AI accelerates delivery and, in doing so, increases the need for discipline, leadership, and sound Agile practice. As my proficiency in AI-driven development deepened and AI became an integrated, primary force in design, analysis, and delivery, a second realization followed. What had been a sequence of discrete tasks, design, then analysis, then build, then test, collapsed into overlapping streams where AI contributed to all of them simultaneously, and the volume of meaningful change in a single working session exceeded what any traditional rhythm could contain. AI does not simply accelerate delivery. It changes the operating assumptions those models rely on, particularly around cadence, the batching of learning, and time-based control.

This was not a rejection of Agile or Scrum. It was precisely because those models have worked well under the conditions they were designed for that their limits under sustained AI acceleration became visible.

ODOM is my response to that shift. It is an operating model that governs how an organization selects intent, structures learning, interprets evidence, and makes decisions when the cost of change is low and the consequences of misunderstanding are high. It replaces timebox-first governance with outcome-bounded learning, and it clarifies what delivery frameworks cannot govern by themselves.

Operating Model Control Assumptions Compared

With the attribution problem now grounded in both a domain example and a practitioner example, it is possible to compare what different methods assume is scarce and see where the gap lies.

Approach	Primary Unit of Control	Primary Optimization Target	Implicit Bottleneck Assumption
Kanban	Work item	Flow efficiency and WIP stability	Workflow congestion limits performance
Extreme Programming (XP)	Engineering change set	Technical feedback quality	Code-level feedback cycles limit quality

Scrum	Timebox (Sprint)	Predictable learning cadence	Cadence and inspection discipline limit uncertainty
ODOM	Outcome (bounded intent)	Attribution integrity and decision clarity	Attribution capacity limits responsible agility under acceleration

Table 3 - Control Assumptions across Agile Methods

This comparison clarifies that ODOM operates at a different control surface than delivery frameworks. It also clarifies the relationship. ODOM replaces Sprint cadence as the governing boundary for learning and decision-making. Teams may still use practices that improve flow, engineering feedback quality, and inspection discipline, but those practices serve the Outcome boundary rather than the calendar. The operating model governs intent and evidence. Delivery practices govern how work is executed within that governance. ODOM assumes a baseline of engineering maturity sufficient to support continuous delivery. Without that foundation, outcome-based control cannot function. Engineering discipline governs the capacity to act on intent and evidence.

ODOM differentiates itself by making outcome-bounded intent the control boundary and using evidence maturity to trigger decisions. Kanban already provides continuous flow and WIP limits. XP already provides fast technical feedback. What neither provides is the governance structure that constrains how much strategic intent is in motion and that separates truth from direction when evidence is evaluated. Under AI acceleration, the binding constraint shifts upward from execution discipline to interpretive capacity. Delivery can accelerate faster than understanding. Under sustained acceleration, attribution becomes the limiting factor for agility.

A Continuation, Not a Reversal

In my earlier paper, *AI-Driven Development & the Future of Scrum and Agile*, I argued that AI would intensify the need for discipline, leadership, and systems thinking rather than eliminate them. That argument remains intact. The conclusion there was that AI accelerates output without supplying judgment, and that human leadership and Agile principles become more important, not less, as speed increases.[39] This paper begins where that thought experiment naturally leads. If AI continues to reduce the cost of change, and if learning and judgment remain human responsibilities, then the question shifts from

[39] Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., Jeffries, R., Kern, J., Marick, B., Martin, R. C., Mellor, S., Schwaber, K., Sutherland, J., & Thomas, D. (2001). *Manifesto for Agile Software Development*. https://agilemanifesto.org.

whether Scrum still matters to whether the implicit operating model built around Scrum is still sufficient. What follows is not a rejection of the ideas that made Scrum valuable, but an examination of how those ideas must be re-expressed at the operating-model level as AI adoption crosses from experimentation into everyday use.[40,41]

The next section introduces ODOM as an operating model built to preserve learning integrity, decision quality, and responsibility as AI changes the economics of work.

Outcomes as the New Control Surface

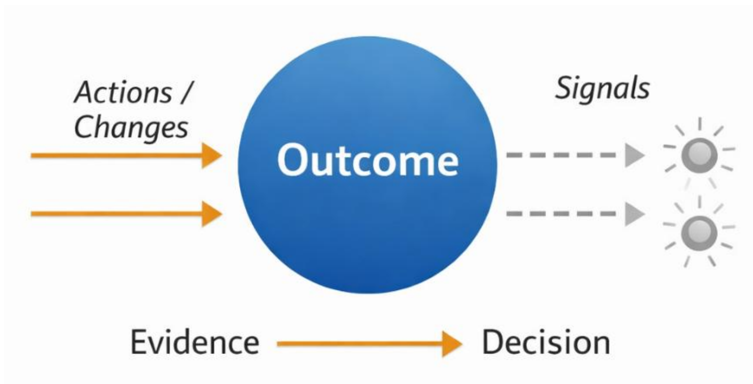

Figure 2 - Outcome as the Unit of Control

Even in organizations that have never spoken about outcomes explicitly, leaders already reason in outcome terms. When executives discuss strategy, they do not start with tasks. They talk about market behavior, customer adoption, operational stability, regulatory posture, risk exposure, and growth trajectories. These are all descriptions of change in the world, not descriptions of work performed. They are outcomes, whether or not they are named that way.

The difficulty is that most operating models lose this clarity once work begins. Strategic intent is translated into initiatives, initiatives into projects, projects into plans, and plans into backlogs. By the time delivery is underway, the original intent has been fragmented into activity. Leaders are then asked to infer progress toward outcomes by inspecting output and proxy metrics. This indirection has always been imperfect, but it was tolerable when the pace of change was slower and learning could be batched.

---

[40] Peng, S., Kalliamvakou, E., Cihon, P., & Demirer, M. (2023). *The impact of AI on developer productivity: Evidence from GitHub Copilot* (arXiv:2302.06590). arXiv. https://arxiv.org/abs/2302.06590.
[41] Brynjolfsson, E., Li, D., & Raymond, L. R. (2023). *Generative AI at work* (NBER Working Paper No. 31161). National Bureau of Economic Research. https://www.nber.org/papers/w31161.

AI makes this indirection dangerous. As the cost of producing change falls, organizations can move further away from intent faster than they can recognize it. Teams can be extremely busy while the original purpose becomes harder to see. Leaders receive more updates, more dashboards, and more signals, yet struggle to answer a simple question: *what, exactly, are we trying to change right now, and is it actually happening*?

This is where outcome-bounded control becomes necessary. An outcome, in this context, is not a slogan or a KPI (Key Performance Indicator).[42] It is a deliberate statement of the change an organization is attempting to create in a specific part of the system, expressed in a way that can be observed. It anchors intent to reality. It makes learning explicit. It gives decision-makers a stable reference point in an environment where activity is accelerating.

The key shift is structural. It goes beyond semantics.

Instead of allowing many changes to proceed in parallel and hoping that periodic reviews will reveal what mattered, an outcome-driven approach constrains attention intentionally. It forces the organization to say, "*This is the change we are attempting now. We will act in service of this change. We will observe how the system responds. And we will decide what to do next based on evidence, not momentum*."

This constraint is what preserves attribution. When AI accelerates delivery, attribution becomes the scarcest resource. Without attribution, organizations cannot learn. Without learning, they cannot adapt responsibly. Outcomes provide a way to preserve that attribution by limiting how much intent is in motion at one time, even as execution becomes faster.[43,44] Any operating model that constrains intent to a single outcome per team is applying a WIP limit at the level of strategic intent.

Teams can keep moving fast. The change is that decisions slow down just enough to stay grounded in reality.

In traditional Scrum environments, the Sprint attempts to play this role indirectly. It bundles intent, action, and learning into a fixed window and assumes coherence will emerge within that boundary. As AI adoption deepens, that assumption weakens. Outcomes replace the timebox as the primary boundary of learning. They allow work to

---

[42] Doerr, J. (2018). *Measure what matters: How Google, Bono, and the Gates Foundation rock the world with OKRs*. Portfolio.
[43] Peng, S., Kalliamvakou, E., Cihon, P., & Demirer, M. (2023). *The impact of AI on developer productivity: Evidence from GitHub Copilot* (arXiv:2302.06590). arXiv. https://arxiv.org/abs/2302.06590.
[44] Brynjolfsson, E., Li, D., & Raymond, L. R. (2023). *Generative AI at work* (NBER Working Paper No. 31161). National Bureau of Economic Research. https://www.nber.org/papers/w31161.

flow continuously while still anchoring interpretation and decision-making to a single, explicit intent.

For executives, this changes the nature of oversight. Instead of asking whether teams are on track against a plan, leaders can ask whether the outcome in focus is actually emerging in the world. Instead of scanning dashboards for reassurance, they can look at signals tied directly to the change they care about. Instead of approving more work when things feel slow, they can decide whether the current outcome deserves further investment or whether learning suggests a change in direction.[45]

This is the shift from managing activity to managing learning. The Outcome-Driven Operating Model (ODOM) formalizes this shift. It treats outcomes as the unit of intent, evidence as the basis for decision, and learning as something that must be structurally protected, not assumed. It does not attempt to optimize delivery speed. It assumes delivery speed will increase. Its purpose is to ensure that increased speed does not outpace understanding.

The next section explains how ODOM structures work around outcomes in a way that remains coherent as AI adoption moves from experimentation to everyday practice, and why this structure becomes necessary once the organization can change faster than it can explain itself.

How ODOM Structures Work and Decisions Around a Single Outcome

ODOM is organized as a small, explicit operating stack.[46] Strategy, Themes, and Initiatives set directional intent above the team level. Outcomes move through a visible Outcome Pipeline from idea to learning.

Discovery is a structured conversation about the future. While the team is building the current Outcome, Discovery runs alongside it, preparing what comes next. Teams and stakeholders examine candidate Outcomes from the Pipeline and work through what each one actually means. *What specific change is being attempted? What behavior in the system would demonstrate that change? What signals would the Evidence Package need to contain? What is deliberately out of scope?* AI makes this preparation faster and more thorough than traditional refinement ever allowed. Analysis that once required weeks can now run in parallel with current work, which means there is no reason for an Outcome to

[45] Schein, E. H. (2013). *Humble inquiry: The gentle art of asking instead of telling*. Berrett-Koehler Publishers.
[46] Ross, J. W., Weill, P., & Robertson, D. C. (2006). *Enterprise architecture as strategy: Creating a foundation for business execution*. Harvard Business School Press.

arrive at the ODOM Loop incomplete. Discovery produces fully formed Outcomes with clear intent, defined evidence criteria, and explicit boundaries. When the team reaches Kickoff, the conversation is about confirming readiness and committing to the work, not debating what the Outcome means. This is what separates ODOM from operating models that accept ambiguity at the point of commitment and hope clarity will emerge during execution.

The ODOM Loop adapts PDCA, often called the Deming Wheel.[47,48] It expresses the loop as four stages, Kickoff, Build, Assessment, and Reflection. The Evidence Package defines how behavior change will be observed and interpreted. It must include disconfirming signals and explicit stop criteria, otherwise outcome control becomes narrative control. Outcomes conclude as Completed, Retired, or Adjusted based on Assessment of the Evidence Package. Teams build one Outcome at a time, while multiple Outcomes may be under evaluation as Signals mature. Evaluation means observing signals and interpreting evidence. It does not mean executing additional change. Cadence supports alignment and learning, but control comes from Outcome readiness and evidence, not from timeboxed cycles.

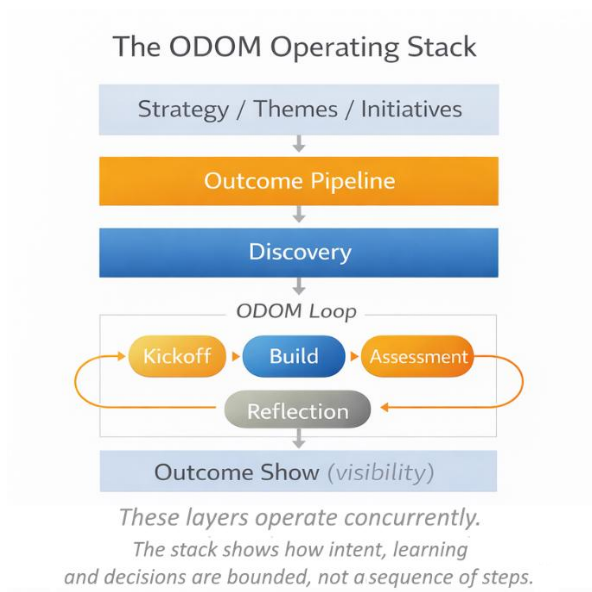

Figure 3 - Layered Structure of the ODOM Model

---

[47] Moen, R. D., & Norman, C. L. (2010, November). *Circling back, clearing up myths about the Deming cycle and seeing how it keeps evolving*. *Quality Progress, 43*(11), 22–28.
[48] Shewhart, W. A. (1939). *Statistical method from the viewpoint of quality control*. The Graduate School, U.S. Department of Agriculture. https://doi.org/10.22004/ag.econ.327285.

ODOM events are structured differently. They are not triggered by the calendar. Cadence may exist for coordination, but it is not the control boundary for learning or decision-making. Instead, events are triggered by the state of the work and the state of the evidence. Each event has a conversational purpose, and that purpose constrains what the conversation is about. This constraint is what makes the conversations productive rather than performative.

Kickoff is a conversation about intent. The team and its stakeholders align on what change is being attempted, how that change will be observed, and what the Evidence Package will consist of. This is where shared understanding is built, not assumed. A Kickoff that does not produce genuine clarity about the Outcome, the signals that will matter, and the boundaries of the work has not succeeded, regardless of how efficiently it was conducted. The question Kickoff exists to answer is *what are we doing, and how will we know if it's working?*

Build is not a conversation-free zone. While execution is the primary activity, Build includes ongoing informal signal conversations as the team works. Teams notice things. Early indicators surface. Assumptions are tested against emerging reality. These conversations are not formalized into ceremonies, but they are recognized as part of the operating rhythm. The discipline during Build is that these conversations remain oriented toward the Outcome—not toward scope expansion, not toward unrelated observations, and not toward premature conclusions. The question Build exists to answer is *what are we seeing as we pursue this change?*

Assessment is a conversation about truth. The team brings together the accumulated evidence and states plainly what the signals show about the Outcome. Assessment is disciplined sensemaking against a clearly defined intent. It replaces report-outs and progress updates with evidence and interpretation. The team focuses on what the evidence supports and where it remains inconclusive. The conclusion from Assessment is what crosses the boundary to leadership for decision-making. The questions Assessment exists to answer is *did the intended change occur, and what does the evidence tell us?*

Reflection is a conversation about capability. Once the Outcome has been assessed and a decision has been made, the team examines how it pursued the work. *What worked well in how the team operated? What created friction? Where did assumptions prove wrong, not about the Outcome itself, but about how the team approached it?* Reflection is not retrospective in the calendar-triggered sense. It is an intentional pause to improve the team's ability to pursue the next Outcome with greater clarity, discipline, and skill. The question Reflection exists to answer is *how do we get better at this?*

Taken together, these four conversations form a complete learning cycle: align on intent, observe while executing, interpret evidence honestly, and improve the approach. What makes them different from traditional Agile ceremonies is not the content (Scrum events address similar concerns) but the trigger and the constraint. ODOM conversations are evidence-triggered and intent-bounded. They happen when the work demands them, and they stay focused on the Outcome that defines their purpose.

Status-driven meetings create the illusion of alignment while causality gets weaker. Everyone stays busy and informed, but fewer decisions get grounded in evidence. As AI accelerates the pace of work, that pattern gets worse because change outpaces explanation. ODOM preserves clarity and alignment by shaping conversations around purpose and evidence rather than schedule. Meetings can still happen often, but they stay tied to what the organization is trying to learn.

This is the core idea behind the Outcome-Driven Operating Model. ODOM assumes that, in AI-accelerated environments, organizations will always be capable of making more changes than they can understand. The model therefore treats learning capacity, not execution capacity, as the primary constraint.

ODOM begins by forcing the organization to choose a single outcome to pursue per team. That outcome defines what change matters now and, just as importantly, what changes must wait. This constraint is uncomfortable for organizations used to parallel optimization, but it is what preserves attribution.

Operational work such as defect remediation, support escalation, or platform maintenance does not disappear under ODOM. When these concerns materially affect behavior in the system, they are elevated into explicit Outcomes. When they do not, they are handled as operational work that does not compete with outcome-level intent, allowing teams to maintain system health without fragmenting learning or decision focus.

Work under ODOM is not timeboxed. It flows continuously. Teams explore solutions, make adjustments, and release changes as needed. AI accelerates exploration and implementation, but it does not determine direction. The outcome defines the boundary of relevance.

Signals are observed continuously, not aggregated at the end of a Sprint.[49] Evidence accumulates as behavior changes in the system. When there is enough evidence to make a

---

[49] Meadows. D.H. (2008). *Thinking in systems: A primer*. Chelsea Green Publishing.

decision, the organization pauses deliberately. This is Assessment. The decision is explicit: continue, adjust, or stop.

Only after a decision is made does Reflection occur. It is about improving how the team pursued it, so the next outcome can be approached with greater clarity.

For executives, this changes what governance looks like. Instead of managing calendars and commitments, leadership governs intent at the level of themes and initiatives and makes decisions based on outcome evidence. Instead of asking whether teams are on track, leaders ask whether the intended change is occurring. Instead of funding streams of activity, they fund learning with clear decision points.

ODOM does not slow organizations down. It assumes AI will continue to accelerate delivery. Its purpose is to ensure that acceleration does not outrun understanding.

How Assessment Enables Executive Decision-Making Without Executive Interference

One of the persistent problems in large organizations is that learning and decision-making are tangled together. Teams generate information. Leaders interpret it. Conclusions are negotiated. Accountability blurs. Over time, evidence loses authority and confidence fills the gap. This pattern predates Agile and has nothing to do with intent. It emerges whenever organizations lack a clear boundary between *sensemaking* and *direction setting*.[50]

ODOM is designed explicitly to separate those responsibilities. Assessment is the mechanism that creates that separation.

When a team reaches Assessment, it performs disciplined sensemaking against a clearly defined Outcome. The team has been collecting signals throughout the work. Those signals are chosen because they trace whether the intended change is occurring in the system.

Assessment is where the team brings those signals together and states, plainly and without hedging, what they mean. This moment matters because it is the first time in many organizations where truth is allowed to stand on its own, without being immediately reshaped to fit a plan or a narrative. The team states what the evidence supports and answers whether the Outcome is emerging, constrained, or not occurring.

[50] Weick, K. E. (1995). *Sensemaking in organizations*. Sage Publications.

That conclusion is the product of Assessment. Because AI dramatically reduces the cost of catching up once a decision is made, teams and leaders can afford to spend more time interpreting evidence and choosing direction without sacrificing delivery speed.

For executives, this changes the nature of decision-making in a fundamental way. Instead of receiving updates framed around progress, plans, or confidence levels, leadership receives an assessment grounded in evidence. The question is no longer "*are we on track,*" because track implies a predetermined destination and timeline. The question becomes "*is the change we intended actually happening*."

This is a higher-quality input into executive decision-making. Executives receive an assessment grounded in evidence, then make portfolio decisions based on it. They are freed from adjudicating delivery reality, arbitrating interpretations of progress, inferring learning from activity, and requesting additional dashboards.

They can focus on decisions that are appropriately theirs. When an Outcome is assessed as achieved, leadership can decide whether that change is sufficient or whether further investment is warranted. When an Outcome is assessed as constrained, leadership can decide whether to accept those constraints or redirect effort elsewhere. When an Outcome is assessed as not emerging, leadership can decide to stop, pivot, or reallocate capacity.

These decisions are portfolio decisions, not delivery decisions. The discipline here is subtle but powerful. Executives are not stepping back from responsibility. They are stepping into a clearer form of it. They are making directional choices based on evidence rather than managing uncertainty through pressure, escalation, or premature commitment.

This is how ODOM replaces calendar-based governance without introducing heavier control structures. There is no need to wait for the end of a Sprint or a quarter to make decisions. Decisions happen when learning justifies them. At the same time, leaders are protected from reacting to noise, because Assessment is not triggered by every signal fluctuation. It occurs when the team believes there is enough evidence to conclude something meaningful.

In this way, Assessment functions as a stabilizing interface between accelerated execution and deliberate leadership. As AI adoption increases, this interface becomes essential. When change is cheap and frequent, leaders need fewer opinions and better conclusions. Assessment provides that, without dragging executives into the mechanics of delivery or turning teams into report generators.

As AI absorbs more execution work, human development increasingly centers on judgment, interpretation, and responsibility, capacities that operating models must deliberately create space for. Assessment conclusions are shared through an Outcome

Show, providing visibility and organizational learning without reopening decisions or reinterpreting evidence.

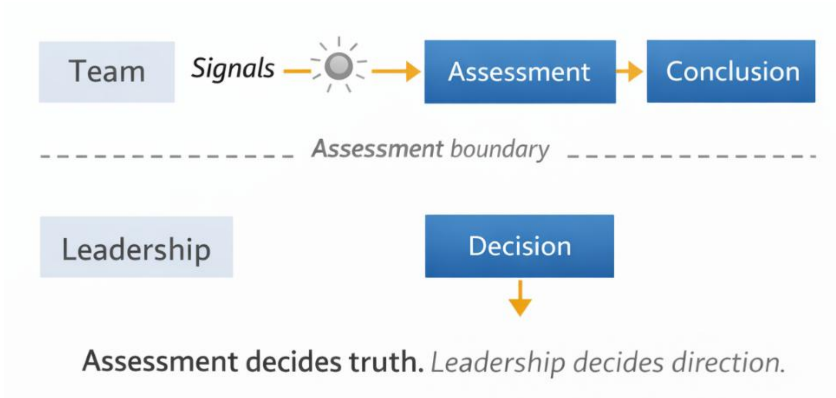

Figure 4 - Separating Truth from Direction

Why This Preserves Accountability on Both Sides

This separation of roles also restores accountability in a way many organizations have lost. Teams are accountable for producing truthful assessments. They cannot hide behind activity, effort, or partial success. They must state what the evidence shows, even when it is uncomfortable.[51]

Executives are accountable for acting on those assessments. They cannot defer decisions indefinitely, demand more work without clarity, or reinterpret evidence to justify predetermined outcomes.

Both sides retain agency. Neither side absorbs the other's responsibility.

This balance is difficult to maintain in Sprint-based governance as AI accelerates work. When learning is batched into timeboxes, pressure builds to reinterpret results rather than act on them. ODOM replaces that pressure with a cleaner contract.[52,53]

Teams own truth.
Leaders own direction.
The operating model protects the boundary.

---

[51] Edmondson, A. (1999). Psychological safety and learning behavior in work teams. *Administrative Science Quarterly*, 44(2), 350–383. https://doi.org/10.2307/2666999
[52] Goodhart, C. A. E. (1975). Problems of monetary management: The U.K. experience. In Papers in Monetary Economics (Vol. 1). Reserve Bank of Australia.
[53] Campbell, D. T. (1979). Assessing the impact of planned social change. Evaluation and Program Planning, 2(1), 67–90 https://doi.org/10.1016/0149-7189(79)90048-X

Preparing for the Tipping Point

It is worth emphasizing again that this model is not required the moment AI appears in development workflows. Early experimentation rarely stresses Sprint-based governance because the volume of change is still low enough to explain. The tipping point comes when AI use becomes routine enough that the volume of change outpaces the organization's ability to reason about it using time-based controls.

This paper is **not an argument to abandon Scrum today**. It is an argument to recognize where Scrum's control mechanism will strain tomorrow, and to begin evolving the operating model before that strain turns into confusion.

How Outcome-Based Control Changes Portfolio and Funding Conversations

In most organizations today, portfolio and funding conversations are organized around initiatives, projects, or programs. Even when leaders speak in strategic terms, money ultimately flows into containers of work that assume delivery will proceed for a defined period of time. Progress is then inferred from milestones, burn rates, and delivery confidence.

This structure exists for a reason. It provides predictability. It allows organizations to plan capacity, manage risk, and create a sense of forward motion. For many years, this was a reasonable trade-off. The cost of change was high enough that committing to a course of action for a period of time felt necessary.

AI alters that trade-off. As the cost of change falls, committing to long-running streams of activity becomes less about enabling progress and more about locking in assumptions. Organizations continue to fund work not because it is clearly producing the intended change, but because the commitment itself has become the justification. Over time, this produces a familiar pattern. Work continues. Investment continues. Learning lags.

Outcome-based control reframes this dynamic. In an outcome-driven operating model, funding becomes a commitment to learn whether a specific change can be made to happen. That distinction is subtle but significant.

When a team selects an Outcome, leadership authorizes the attempt and commits to make continuation decisions through evidence. The duration depends on what the evidence supports. When evidence no longer supports the attempt, funding can be redirected without framing the decision as failure.

This is where ODOM changes the tone of portfolio discussions. Instead of asking which initiatives should be prioritized for the next quarter, leadership asks which outcomes matter most now, given the organization's strategic context. Instead of debating estimates and delivery timelines, leaders look at the evidence emerging from outcomes already in motion. Decisions become comparative and dynamic rather than static and defensive.

This keeps financial discipline intact and often strengthens it. Funding decisions remember why they exist. Money is allocated to produce learning and change, not to protect sunk costs. Outcomes that demonstrate traction can be extended. Outcomes that stall can be stopped without requiring a postmortem to justify the decision. Outcomes that reveal unexpected constraints can inform broader strategy.

This approach also reduces a common form of hidden risk. In activity-based funding models, risk accumulates silently. The longer work continues without clear evidence of impact, the more politically difficult it becomes to stop. Leaders sense this intuitively, which is why they often ask for more reporting, more metrics, and more assurance as initiatives age. These mechanisms provide comfort, but they rarely restore clarity. Outcome-based funding addresses this directly by tying continuation to evidence rather than optimism.

Why This Matters as AI Adoption Accelerates

As AI becomes embedded in everyday development work, the number of plausible initiatives an organization could pursue will increase dramatically. Ideas become cheaper. Experiments become easier. The temptation to fund more things in parallel grows.

Without an outcome-based operating model, this leads to portfolio sprawl. Many initiatives appear active. Few produce reusable learning. Leadership attention fragments. Decision quality declines, even as information volume increases.

ODOM provides a way to remain selective without becoming rigid. By limiting how many outcomes are actively pursued and by making evidence the basis for continuation, the organization preserves its ability to focus. AI is then used to accelerate learning within those constraints rather than to justify broader dispersion of effort.

This is where the model connects back to systems thinking. Little's Law reminds us that increasing throughput without managing work in progress does not reduce cycle time. At the portfolio level, outcomes function as a WIP constraint on intent. They limit how much strategic change is in motion at once. This protects not just teams, but leadership cognition.[54]

Goodhart's Law also plays a role here. When funding is tied to activity metrics, those metrics become targets and see diminishing informational value. When funding is tied to outcome evidence, the organization is incentivized to surface truth rather than performance.[55]

Deming's emphasis on understanding variation becomes actionable. Leaders are no longer reacting to fluctuations in delivery metrics. They are interpreting variation in outcome signals and making informed decisions about where to intervene.[56]

A Different Relationship Between Strategy and Execution

Perhaps the most important shift introduced by an outcome-driven operating model is how strategy and execution relate to one another. In traditional models, strategy is articulated periodically and execution is expected to conform until the next planning cycle. Learning that contradicts strategic assumptions often arrives too late to matter, or seeps upward informally.

Under ODOM, strategy remains directional, but it is continuously informed by outcome evidence.[57] Strategic intent is expressed through outcomes. Execution generates signals. Assessment produces conclusions. Leadership decisions then adjust direction explicitly.

This is where organizational agility stops being an aspiration and becomes a structural property of the operating model. When evidence from Outcomes flows directly into strategic decisions, and when those decisions can redirect effort without unwinding months of commitments, the organization gains something most operating models only claim: the ability to respond to reality faster than reality can outpace the response.

The conversations that make this possible are the mechanism. Assessment produces an honest conclusion for an Outcome when evidence is sufficient. The Outcome Show shares

---

[54] Little, J. D. C. (1961). A proof for the queuing formula: L = λW. Operations Research, 9(3), 383–387. https://doi.org/10.1287/opre.9.3.383
[55] Goodhart, C. A. E. (1975). *Problems of monetary management: The U.K. experience*. In Papers in Monetary Economics (Vol. 1). Reserve Bank of Australia.
[56] Deming, W. E. (1986). *Out of the crisis*. MIT Press.
[57] Teece, D. J., Pisano, G., & Shuen, A. (1997). Dynamic capabilities and strategic management. *Strategic Management Journal*, 18(7), 509–533. https://doi.org/10.1002/(SICI)1097-0266(199708)18:7<509::AID-SMJ882>3.0.CO;2-Z

completed Assessments with leadership. Leadership records a directional decision for each assessed Outcome. The next Outcome Kickoff aligns the team on the next bounded intent while Discovery continues alongside Build. The loop stays tight because every conversation creates the clarity the next conversation requires, even while different Outcomes sit in different stages across teams.

This creates a tighter, more honest loop between intent and reality. It also changes how leaders experience control. Control no longer comes from detailed plans or fixed commitments. It comes from knowing, at any moment, which outcomes the organization is actively pursuing and what the evidence says about them. That form of control is quieter, but stronger. It relies less on reassurance and more on understanding.

Why This Is an Evolution, Not a Rejection

It is worth restating that none of this implies that past operating models were misguided. They were responses to different conditions. Scrum brought cadence and inspection discipline to environments that lacked feedback. Kanban brought flow and WIP management. XP brought engineering discipline and fast feedback loops. Lean thinking brought systems perspective and waste reduction. Each was a response to a real constraint in its era.

Discipline retains its value. AI shifts the level where discipline matters most. As AI adoption continues, discipline must move upstream, from scheduling work to structuring intent, learning, and decision-making. The Outcome-Driven Operating Model is one expression of that shift. It does not invalidate the methods that precede it. It addresses a constraint that emerges above them when AI changes the economics of delivery.

What remains is the question of timing. This is not a question of immediate adoption, but of recognition. Leaders need to identify the point at which Sprint-based control no longer provides sufficient clarity and an outcome-driven operating model becomes necessary to preserve learning, attribution, and responsible decision-making.

Recognizing the Tipping Point

Operating models do not change because a better idea appears. They change because existing structures stop producing clarity under new conditions. AI lets organizations move faster than their controls can keep up. The risk is speed that outruns understanding.

The transition point is rarely dramatic. There is no single moment when Scrum "fails." Instead, leaders begin to notice a pattern of small but persistent signals.

They see more work happening without a corresponding increase in understanding. Reviews contain more explanation than insight. Confidence remains high, but decisions feel harder. Teams report being busy and productive, yet leadership struggles to explain which efforts are actually driving results. When outcomes improve, no one is fully sure why. When outcomes stall, the response is to try more things in parallel.

These signals are often misdiagnosed as communication problems, alignment problems, or execution problems. In reality, they are indicators that the organization's unit of control no longer matches the pace of change.

Another sign is how AI is being discussed internally. Early on, AI is treated as a productivity enhancer. Later, it becomes a strategic lever. When leaders begin asking why certain decisions still take weeks when analysis now takes hours, or why planning cycles remain fixed while delivery accelerates, the organization is approaching the tipping point.

At that stage, attempting to reinforce Sprint-based control usually increases friction rather than restoring clarity. More reporting is requested. More metrics are introduced. Reviews become longer. Retrospectives focus on coordination rather than learning. These responses are understandable, but they are compensatory. They signal that the underlying structure is under strain.

An outcome-driven operating model becomes relevant precisely when these compensations appear. ODOM assumes basic delivery discipline. It is aimed at organizations that can already move and are beginning to question whether they understand the consequences of that movement. It fits environments where AI is widely used but not fully trusted, and where leaders need a way to govern acceleration without reverting to heavy planning or constant intervention.

Importantly, adopting outcome-based control does not require an abrupt shift. It can begin as a lens rather than a mandate. Leaders can start by asking outcome-oriented questions in the governance forums they already use. Teams can experiment with outcome-bounded work in limited contexts. Over time, the organization learns whether this structure produces clearer decisions and more reusable learning.

What matters is not speed of adoption, but readiness of recognition. Organizations that recognize the tipping point early have options. They can evolve deliberately. They can preserve accountability. They can let AI accelerate delivery while keeping learning intact.

Organizations that recognize it late will still adapt, but under pressure. They will respond to confusion rather than prevent it. They will trade understanding for reassurance until the cost becomes unavoidable.

The purpose of this paper is to make visible the structural shift that AI introduces and to offer a coherent way of responding when existing controls begin to falter. It avoids calling for immediate replacement of one operating model with another.

Scrum brought discipline to uncertainty when delivery was slow and learning was expensive. Kanban brought flow when batch-based work created bottlenecks. XP brought technical integrity when code quality was the binding constraint. AI changes that equation. The Outcome-Driven Operating Model is an attempt to bring discipline to acceleration, so that organizations can move faster without losing the ability to understand what they are doing and why.

That is the challenge leaders will face in the near future. The question is not whether it will arrive, but whether the organization will be prepared to meet it with clarity rather than reaction.

References

1. Anderson, D. J. (2010). *Kanban: Successful Evolutionary Change for Your Technology Business*. Blue Hole Press.
2. Anderson, D. J., & Carmichael, A. (2016). *Essential Kanban Condensed*. Lean Kanban University Press.
3. Beck, K. (2000). *Extreme Programming Explained: Embrace Change*. Addison-Wesley.
4. Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., Jeffries, R., Kern, J., Marick, B., Martin, R. C., Mellor, S., Schwaber, K., Sutherland, J., & Thomas, D. (2001). *Manifesto for Agile Software Development*. https://agilemanifesto.org.
5. Beyer, B., Jones, C., Petoff, J., & Murphy, N. R. (2016). *Site reliability engineering: How Google runs production systems*. O'Reilly Media.
6. Brynjolfsson, E., & Hitt, L. M. (2000). Beyond computation: Information technology, organizational transformation and business performance. *Journal of Economic Perspectives*, 14(4), 23–48. https://doi.org/10.1257/jep.14.4.23
7. Brynjolfsson, E., Li, D., & Raymond, L. R. (2023). *Generative AI at work* (NBER Working Paper No. 31161). National Bureau of Economic Research. https://www.nber.org/papers/w31161.
8. Campbell, D. T. (1979). Assessing the impact of planned social change. *Evaluation and Program Planning*, 2(1), 67–90. https://doi.org/10.1016/0149-7189(79)90048-X
9. Christensen, C. M., Hall, T., Dillon, K., & Duncan, D. S. (2016). *Competing against luck: The story of innovation and customer choice*. Harper Business.
10. Conboy, K. (2009). Agility from first principles: Reconstructing the concept of agility in information systems development. *Information Systems Research*, 20(3), 329–354. https://doi.org/10.1287/isre.1090.0236
11. Conway, M. E. (1968). How do committees invent? *Datamation*, 14(4), 28–31. https://www.melconway.com/Home/pdf/committees.pdf.
12. David, P. A. (1990). The dynamo and the computer: An historical perspective on the modern productivity paradox. *American Economic Review*, 80(2), 355–361. https://www.jstor.org/stable/2006600.

13. Dell'Acqua, F., McFowland, E., & Mollick, E. (2023). *Navigating the jagged technological frontier: Field experimental evidence of the effects of AI on knowledge worker productivity and quality* (Working paper). Harvard Business School. https://www.hbs.edu/faculty/Pages/item.aspx?num=64700.

14. Deming, W. E. (1986). *Out of the crisis*. MIT Press.

15. Deming, W. E. (1994). *The new economics for industry, government, education* (2nd ed.). MIT Press.

16. Doerr, J. (2018). *Measure what matters: How Google, Bono, and the Gates Foundation rock the world with OKRs*. Portfolio.

17. Edmondson, A. (1999). Psychological safety and learning behavior in work teams. *Administrative Science Quarterly*, 44(2), 350–383. https://doi.org/10.2307/2666999

18. Fisher, R. A. (1935). *The design of experiments*. Oliver & Boyd.

19. Forsgren, N., Humble, J., & Kim, G. (2018). *Accelerate: The science of lean software and DevOps: Building and scaling high performing technology organizations*. IT Revolution Press.

20. Gall, J. (2002). *The systems bible: The beginner's guide to systems large and small: Being the third edition of Systemantics*. General Systemantics Press.

21. Goldratt, E. M. (1984). *The goal: A process of ongoing improvement*. North River Press.

22. Goodhart, C. A. E. (1975). Problems of monetary management: The U.K. experience. In *Papers in Monetary Economics* (Vol. 1). Reserve Bank of Australia.

23. Johnson, D. (2002). The unintended consequences of qualitative performance metrics. *Systems Research and Behavioral Science*, 19(3), 249–260. https://doi.org/10.1002/sres.431

24. Kohavi, R., Tang, D., & Xu, Y. (2020). *Trustworthy online controlled experiments: A practical guide to A/B testing*. Cambridge University Press. https://doi.org/10.1017/9781108653985

25. Little, J. D. C. (1961). A proof for the queuing formula: L = λW. *Operations Research*, 9(3), 383–387. https://doi.org/10.1287/opre.9.3.383

26. Meadows, D. H. (2008). *Thinking in systems: A primer*. Chelsea Green Publishing.

27. Moen, R. D., & Norman, C. L. (2010, November). *Circling back, clearing up myths about the Deming cycle and seeing how it keeps evolving*. *Quality Progress, 43*(11), 22–28. https://asq.org/quality-progress/articles/circling-back?id=5e4d20950fad4b519133490466ccbf0a.

28. Noy, S., & Zhang, W. (2023). Experimental evidence on the productivity effects of generative artificial intelligence. *Science*, 381(6654), 187–192. https://doi.org/10.1126/science.adh2586

29. Overby, E., Bharadwaj, A., & Sambamurthy, V. (2006). Enterprise agility and the enabling role of information technology. *European Journal of Information Systems*, 15(2), 120–131. https://doi.org/10.1057/palgrave.ejis.3000600

30. Peng, S., Kalliamvakou, E., Cihon, P., & Demirer, M. (2023). *The impact of AI on developer productivity: Evidence from GitHub Copilot* (arXiv:2302.06590). arXiv. https://arxiv.org/abs/2302.06590.

31. Reinertsen, D. G. (2009). *The Principles of Product Development Flow: Second Generation Lean Product Development*. Celeritas Publishing.

32. Rigby, D. K., Sutherland, J., & Takeuchi, H. (2016). Embracing Agile. *Harvard Business Review*, 94(5), 40–50. https://hbr.org/2016/05/embracing-agile.

33. Ross, J. W., Beath, C. M., & Mocker, M. (2019). *Designed for digital: How to architect your business for sustained success*. MIT Press.

34. Ross, J. W., Weill, P., & Robertson, D. C. (2006). *Enterprise architecture as strategy: Creating a foundation for business execution*. Harvard Business School Press.

35. Schein, E. H. (2013). *Humble inquiry: The gentle art of asking instead of telling*. Berrett-Koehler Publishers.

36. Schwaber, K., & Sutherland, J. (2020). *The Scrum Guide*. Scrum Guides. https://scrumguides.org/scrum-guide.html.

37. Scrum.org. (2020). *Evidence-Based Management Guide*. Scrum.org. https://www.scrum.org/resources/evidence-based-management-guide.

38. Shewhart, W. A. (1939). *Statistical method from the viewpoint of quality control*. The Graduate School, U.S. Department of Agriculture. https://doi.org/10.22004/ag.econ.327285.

39. Teece, D. J., Pisano, G., & Shuen, A. (1997). Dynamic capabilities and strategic management. *Strategic Management Journal*, 18(7), 509–533.

https://doi.org/10.1002/(SICI)1097-0266(199708)18:7<509::AID-SMJ882>3.0.CO;2-Z

40. Weick, K. E. (1995). *Sensemaking in organizations*. Sage Publications.
41. Weill, P., & Ross, J. W. (2004). *IT governance: How top performers manage IT decision rights for superior results*. Harvard Business School Press.
42. Weill, P., & Ross, J. W. (2009). *IT savvy: What top executives must know to go from pain to gain*. Harvard Business Press.
43. Westerman, G., Bonnet, D., & McAfee, A. (2014). *Leading digital: Turning technology into business transformation*. Harvard Business Review Press.
44. Womack, J. P., Jones, D. T., & Roos, D. (1990). *The machine that changed the world: The story of lean production*. Free Press.

Appendix A: Defense of the Learning Integrity Claim

Clarifying the Claim

This paper asserts that the Outcome-Driven Operating Model (ODOM) optimizes learning integrity under acceleration. It is a structural claim about what an operating model protects when the pace and parallelism of change increase.

To evaluate the claim, three elements must be explicit:

1. What is meant by learning integrity.
2. The mechanism by which ODOM improves it.
3. The observable consequences that would support or weaken the claim.

Without these elements, the statement remains conceptual. With them, it becomes testable.

Learning Integrity

Learning integrity is the organization's ability to produce decision-relevant, causally credible conclusions from its work, even when delivery is rapid and change is frequent.

Learning integrity is present when the organization can:

- Articulate intent as a specific, testable change in the system.
- Bound which interventions are relevant to that change.
- Define in advance what would constitute meaningful evidence.
- Interpret results with calibrated confidence rather than narrative certainty.
- Reuse conclusions to inform subsequent direction.

This definition treats learning as disciplined inference, not storytelling.[58] Without controlled variation and predefined interpretive boundaries, activity is easily mistaken for impact and coherence is substituted for causality.[59] Trustworthy experimentation requires that evaluation criteria be defined before results are known, not selected afterward to fit a

---

[58] Campbell, D. T. (1979). Assessing the impact of planned social change. *Evaluation and Program Planning, 2*(1), 67–90. https://doi.org/10.1016/0149-7189(79)90048-X
[59] Fisher, R. A. (1935). *The design of experiments*. Oliver & Boyd.

preferred narrative.[60] Learning integrity concerns the credibility of inference, not the volume of output.

The Structural Effect of Acceleration

AI reduces the cost of generating and implementing change. As delivery accelerates, more variations, refinements, and experiments can occur within the same calendar period.

This produces a predictable systems effect. As more interventions overlap, confounding increases. Signals interact. Attribution weakens. The organization experiences more observable movement, yet less clarity about what caused that movement.

Timeboxes regulate cadence. Flow systems regulate work-in-progress. Engineering practices regulate technical quality. None of these mechanisms inherently constrain how many distinct outcome hypotheses are pursued simultaneously.

Under acceleration, intent concurrency becomes the dominant source of attribution loss. The scarce resource shifts from execution capacity to interpretive capacity.

The Mechanism of ODOM

ODOM raises the unit of control from time or work-items to explicitly bounded intent, expressed as a single Outcome in focus per team at a time.

This shift has three structural consequences.

1. Reduction of Intent Concurrency

By limiting teams to one Outcome in Build at a time, ODOM constrains how many independent hypotheses are exercised concurrently within a team's active delivery window. In experimental terms, fewer independent variables move simultaneously within that window, strengthening inferential credibility.[61] Across the portfolio, multiple Outcomes can still be in Assessment in parallel as signals mature.

Execution can remain fast. Exploration can remain aggressive. The constraint operates at the level of intent, not activity.

---

[60] Kohavi, R., Tang, D., & Xu, Y. (2020). *Trustworthy online controlled experiments: A practical guide to A/B testing*. Cambridge University Press.
[61] Fisher, R. A. (1935). *The design of experiments*. Oliver & Boyd.

2. Predefinition of Evidence

Each Outcome includes an Evidence Package defined prior to execution. This reduces post-hoc metric selection and interpretive drift. When measures are selected after results are visible, they tend to align with narrative needs rather than inference discipline, a phenomenon closely related to Goodhart's Law.[62]

Predefining evidence strengthens attribution by clarifying what constitutes meaningful movement before movement occurs.[63]

3. Evidence-Triggered Assessment

Assessment under ODOM is triggered by evidence sufficiency rather than calendar cadence. Decisions occur when signals are mature enough to support inference, not merely because a time interval has elapsed.

This aligns decision timing with learning maturity. At the portfolio level, the Outcome functions as a constraint on strategic work-in-progress, consistent with queueing dynamics described by Little's Law.[64]

Execution can remain fast. Interpretation becomes fragile unless intent is constrained.

Separation of Truth and Direction

ODOM separates the statement of truth from the act of choosing direction. Teams are accountable for articulating what the evidence shows. Leaders are accountable for determining what to do next. This separation reduces pressure to reinterpret weak evidence to protect prior commitments and reduces reactive intervention based on noise rather than signal.[65]

Governance becomes clearer, not louder. This is the point where the unit of control becomes visible. Timeboxes and flow systems can improve delivery discipline, yet attribution can still blur because they do not constrain how much intent is in motion at

---

[62] Goodhart, C. A. E. (1975). Problems of monetary management: The U.K. experience. Reserve Bank of Australia.
[63] Kohavi, R., Tang, D., & Xu, Y. (2020). *Trustworthy online controlled experiments: A practical guide to A/B testing*. Cambridge University Press.
[64] Little, J. D. C. (1961). A proof for the queuing formula: $L = \lambda W$. *Operations Research, 9*(3), 383–387. https://doi.org/10.1287/opre.9.3.383
[65] Deming, W. E. (1986). *Out of the crisis*. MIT Press.

once. ODOM makes intent the controlled variable. It constrains Build to one Outcome per team. Outcomes then move to Assessment for evaluation, and several Outcomes may be evaluated in parallel as signals mature. Evidence is defined before interpretation, and decisions are triggered only when learning is mature enough to justify them. That is what it means to optimize learning integrity under acceleration.

Empirical Evaluation Design

If ODOM optimizes learning integrity under acceleration, the claim should produce observable effects.

Research Question

Does outcome-bounded governance improve attribution quality and decision integrity under accelerated delivery conditions compared to timeboxed or flow-only governance?

Hypotheses

H1: Outcome-bounded governance increases decision traceability to pre-specified evidence at equal or higher delivery throughput.

H2: Outcome-bounded governance reduces intent concurrency and associated attribution ambiguity.

H3: Outcome-bounded governance improves calibration of assessment conclusions over time.

H4: Outcome-bounded governance reduces time-to-decision once evidence is mature without increasing executive interference.

Study Design

A multi-team longitudinal field study provides a realistic evaluation context.

Sample

Recruit 8 to 16 comparable product or platform teams operating under similar domain complexity and delivery maturity. All teams should have access to comparable AI tooling to ensure acceleration is not unevenly distributed.

Grouping

Match teams based on baseline throughput, team size, and volatility. Within each matched pair, one team operates under existing governance. The other adopts outcome-bounded governance.

Random assignment is preferable. If infeasible, matched-pair quasi-experimental design with pre-period covariates can reduce bias.

Intervention Definition

The outcome-bounded condition must be explicit and auditable. Otherwise, the study measures loosely interpreted adoption rather than structural change.

Minimum intervention requirements:

- One declared Outcome in focus per team at a time.
- A documented Evidence Package defined prior to execution.
- A recorded Assessment conclusion triggered by evidence sufficiency.
- A documented directional decision linked to that Assessment and its Evidence Package. This can be a short decision log entry capturing what was decided, why, and what evidence it relied on.

Execution practices remain unchanged. The intervention operates at the governance layer.

Duration

A minimum of 10 to 12 weeks is required to observe multiple outcome cycles and allow signal lags to surface.

Operational Measures

Decision Traceability Index

For each continue, adjust, or retire decision:

- Was intent explicitly defined in advance?
- Was evidence defined in advance?
- Were signal lags and confounders acknowledged?
- Was the decision explicitly tied to observed evidence rather than activity proxies?

- Is the conclusion reusable as documented learning?

Independent raters score decisions using a standardized rubric. Inter-rater reliability is calculated.

Intent Concurrency

Track the number of distinct outcome hypotheses in motion per team, including unofficial efforts that consume meaningful capacity.

Lower concurrency at equivalent throughput indicates improved attribution conditions.

Calibration Accuracy

At Assessment, teams declare confidence levels in their conclusions. After a predefined lag window, actual outcome persistence is compared to declared confidence.

Improved calibration over time indicates disciplined inference.

Throughput Controls

Track deployment frequency, lead time, and throughput. Improvements in learning integrity must not rely solely on reducing delivery speed.

Time from Evidence Maturity to Decision

Define evidence maturity as the point at which predefined criteria are met. Measure elapsed time to recorded decision.

Qualitative Analysis

Collect Assessment records, decision logs, structured interviews, and Evidence Packages.

Code for patterns of narrative repair, metric substitution, ambiguity at commitment, and executive reinterpretation of evidence.

Disconfirmation Criteria

The claim stands or falls on comparison.

If outcome-bounded governance improves traceability and calibration without reducing delivery speed, it supports the argument that ODOM protects learning when acceleration makes attribution fragile.

If ODOM only looks effective because delivery slows, it is reintroducing friction under a different name and it is failing to protect learning integrity.