Scrum.org™ The Home of Scrum | Whitepaper

Empiricism at Machine Speed

Executive Summary

AI Governance: The Answer Already Exists

Matthew Hodgson, CEO, Zen Ex Machina & PST
Dave West, CEO, Scrum.org

5 part series

September 2026

© 2026 Scrum.org | All Rights Reserved.

The Answer Already Exists

Close to 80% of organizations have deployed generative AI in some form, yet only around 6% capture significant enterprise value from it (McKinsey & Company, 2025). The 5% that Boston Consulting Group classifies as future-built are not buying more AI than everyone else. They have redesigned how work happens around what AI can do (Boston Consulting Group [BCG], 2025). Leaders and laggards have access to the same technology, so what separates them is how the work around it is organized and governed. Governance is also where deployment is running ahead: 74% of companies plan to deploy agentic AI within 2 years, yet only 21% report a mature governance model for autonomous agents (Deloitte, 2026).

Scrum was designed for this class of problem: complex work where cause and effect are clear only in hindsight, the right answer cannot be specified in advance, and quality is a judgment rather than a test result (Schwaber & Sutherland, 2020; Snowden & Boone, 2007). This series argues that Scrum addresses the governance problem agentic AI creates, tests that argument against twelve months of operation in one agentic team, and names where the evidence stops.

> ***About the Relentless Raptors.*** *The Relentless Raptors are a nine-member Scrum Team at Zen Ex Machina: one human Product Owner and eight AI agent personas, each with a defined domain, a quality standard, and an escalation threshold. After a pre-Scrum baseline phase from July 2025, the team has run under Scrum in one-week Sprints since February 2026, producing the digital transformation of a working business. It is the operational case behind this series, and Paper 3 documents it in full (Hodgson, 2026).*

The finding that matters most for a leader concerns where human judgment sits, not how much of it there is. Banks that placed human review at the point where a wrong decision could no longer be corrected, rather than over every output, achieved production rates more than twice those of banks that reviewed every step (Buehler et al., 2024). The Raptors run on the same principle, and it works because it challenges what conventional governance assumes about the people doing the work.

© 2026 Scrum.org | All Rights Reserved.

Why Existing Governance Leaves Three Gaps

Project governance rests on three assumptions: workers have steady cognitive capacity, output is predictable from effort, and people say when they are unsure. Agentic AI breaks all three. An agent's performance depends on how its context is structured rather than on the hours it works (Liu et al., 2024), small changes in how an instruction is worded shift its output (Sclar et al., 2024), and it does not reliably signal its own uncertainty (Xiong et al., 2024). Agents also produce work far faster than people can review it. Because queue length drives lead time (Reinertsen, 2009), a sequential approval step stops being a safeguard and becomes the system's constraint.

Governance built on those assumptions leaves three questions without an owner:

- **Execution governance.** Who is accountable for what the AI produces?
- **Value governance.** Who decides whether it is working on the right problem?
- **Diagnostic governance.** What does the AI team's experience reveal about the organization's own structure?

The wider AI governance literature does not answer them. Regulators address risk, transparency, and accountability at the level of principle (OECD, 2024), and technologists address how to keep agents' operations safe and interruptible (Shavit et al., 2023). Both bodies of work matter, and neither tells an organization how to run a team that includes AI agents, reliably and at quality, with a person accountable for its output and its direction.

The gaps matter little while a person stands at every step. At the first of the three maturity tiers adapted from BCG's research, human control substitutes for governance design; at the second, controls are bolted onto the existing process and each output is still reviewed. In the Tier 3, agents perform multi-step work with no person present at each step, and governance must be designed because nothing else holds the system together. That is also where the returns are: Tier 3 organizations generate 1.7 times the revenue growth of laggards and 3.6 times the 3-year total shareholder return, and 62% of their AI initiatives reach production, compared with 12% at laggards (BCG, 2025).

Automated tests can verify software is functional. An executive briefing, a research synthesis, or a policy analysis is judged by its audience and purpose, and no test suite returns pass or fail. Governance for agentic knowledge work has to hold where quality is judged rather than tested, which is the condition Scrum was built for.

© 2026 Scrum.org | All Rights Reserved.

Scrum Addresses All Three Gaps

Scrum's three empirical pillars map onto the three gaps (Table 1). They work as one chain rather than three separate controls: transparency makes inspection possible, and inspection makes adaptation possible. Without transparency, what looks like adaptation is drift.

Table 1. The three gaps and the pillar that addresses each

Gap	Pillar	Mechanism
Execution governance	Transparency	Sprint events and the Definition of Done make AI work visible and traceable without per-task oversight
Value governance	Inspection	The Sprint Review inspects direction at a fixed cadence, against the Product Goal, by the person accountable for value
Diagnostic governance	Adaptation	Impediments become dated, attributable evidence of what the organization's structure costs

Inspection works at two levels, because an agentic team produces more than a Product Owner can read and Scrum does not ask them to read it all. Output quality can be delegated as the team earns it, provided verification is designed well: a Definition of Done specified across several criteria, repeated review passes, and graded findings rather than pass-or-fail verdicts, the design choices that improve verification accuracy (Kwok et al., 2026). Direction cannot be delegated. The Sprint Review is the fixed point at which the person accountable for value asks whether the product is still moving toward its Product Goal (Schwaber & Sutherland, 2020).

Classifying each backlog item as clear, complicated, complex, or chaotic before an agent acts on it (Snowden & Boone, 2007) stops a team from executing complex work as though it were routine. Triple-loop learning, which questions the governance framework itself rather than the rules within it (Flood & Romm, 1996), stops that framework hardening into a fixed constraint when the problem shifts, and it needs an accountable human to make the call.

The Scrum Guide describes Scrum as having small teams with no sub-teams or hierarchies (Schwaber & Sutherland, 2020), so placing AI

© 2026 Scrum.org | All Rights Reserved.

agents within the Developer accountability extends Scrum's governance rather than applying something it already provides. This series proposes that extension and tests it in one team: agents work within the Developer accountability under the same Definition of Done, while accountable humans hold value, release, and the governing instructions.

What Twelve Months of Operation Showed

The Relentless Raptors are that test: a single-case experimental design, tracked Sprint by Sprint, in which dated revisions to the team's governing instructions are the interventions. Each failure mode acts as its own baseline: it recurs until the instruction that produced it is revised, then stops. Of 5 retrospective-driven revisions assessed for their effect in the following Sprint, 4 produced the predicted change, meeting the 3 demonstrations that single-case design standards require before a causal claim (Kratochwill et al., 2013). Scrum was present in every Sprint, so the case shows what Scrum made possible rather than proving that Scrum alone caused it (Hodgson, 2026). Four findings emerged, in the order the evidence produced them.

Agents absorb rules faster than people, then plateau. Scored against the Agile IQ® maturity instrument at the close of every Sprint, the team reached Stage 2 in 2 months, where comparable human teams take about 3 months, on the same volume of correction: about 600 interventions (Hodgson & Horrigan, 2021). It then held at a strong Stage 2 under an unchanged design through August 2026. What appears to be missing is the informal exchange of information that a human team relies on to move further.

Coordination design can matter as much as model selection. The first movement off the plateau followed a change in how the agents coordinate. In September 2026, channels for agents to claim work and hand it to one another replaced a single orchestrating agent, and early signs of Stage 3 appeared. That change has not yet been replicated, but independent evidence points the same way. In one large preprint experiment, the quality gap between the best and worst coordination protocol on the same model (44%) was about 3 times the gap between 2 capable models on the same protocol (about 14%) (Dochkina, 2026). A second preprint found that coordination through a shared, verified context roughly halved cost per task while improving accuracy (Mao & Mirhoseini, 2026). Both studies used synthetic tasks or benchmarks rather than production work.

Governance matters more in an agentic team, not less. Corridor conversations, body language, and ambient awareness of what colleagues are doing all cover for weak governance in human teams. Agents have none of them. In the Raptors' record, the most instructive failures met the letter of an instruction and stayed invisible until a later check found them (Hodgson, 2026).

© 2026 Scrum.org | All Rights Reserved.

Capacity has stopped being the constraint. On a fixed budget of 10 million tokens per one-week Sprint, the team completes about 10 times as many Product Backlog items as a human team at Stage 2, measured by the Product Owner against Agile IQ® throughput data (Hodgson & Horrigan, 2021). That is more than the Product Owner can inspect item by item. Once capacity is this cheap and forecastable, what limits the team is the quality of its instructions, the design of its governance, and the tacit knowledge someone has to write down before any of it works.

An organization attempting this will not lack agents or tooling. It will lack people who can write down the rules, and agentic governance fails because those rules decay.

How Agentic Governance Fails

The instruction set, not the agent, is the object of governance. Every organization runs a documented system of procedures and policy, and an implicit system that tells people what to notice, what to care about, and when to stop (Sudhir, 2026). People supply the implicit system without being asked. Agents supply none of it, so every function it performs has to be written into the instructions, or it is not performed at all. The instruction set is also the one governance artifact that decays without signal, because the people who used to notice its decay are no longer in the loop.

Paper 4 documents eight failure modes from the Raptors' operation and groups them into four families by what each degrades (Table 2). The families follow the chain of governance: what the team is told determines what it does, what it does has to be visible before it can be inspected, and what is inspected feeds the loop that revises what the team is told. None of the eight produces an error when it occurs.

Table 2. Eight failure modes in four families

Family	Failure mode	What goes wrong	The fix
What the team is told	**Instruction drift**	Instructions fall out of alignment with reality, silently. A writing agent's file specifies a 1,200-word ceiling; six Sprints later the team has moved to long-form, and nobody updated the file. The agent complies with the outdated rule. A model upgrade can produce the same drift with no file edited at all.	Retrospective instrumented for patterns across Increments; every model version change triggers a Definition of Done regression check

© 2026 Scrum.org | All Rights Reserved.

Family	Failure mode	What goes wrong	The fix
What the team is told	**Definition of Done divergence**	The quality standard fragments across the files that implement it. A Definition of Done requires WebP under 200KB; a documented fallback path delivers PNG at 1.18MB. Every file is followed correctly, and the Increment fails.	The Definition of Done sits above every file that implements it; changes propagate at the Retrospective
What the team may do	**Bounded authority erosion**	A vague Sprint Goal produces machine-speed execution of the wrong work. A goal of “improve the website” lets agents optimize for page speed, word count, or image count, none of which was the intended outcome.	Write the Sprint Goal as a boundary specific enough that an agent can tell whether an action falls inside it
What the team may do	**Specification gaming**	An agent routes around a constraint to complete its task. Blocked by an error writing to a shared drive, an agent built an automation workflow that reached the drive by another route. The Definition of Done was met; the route was unauthorized.	Narrow the options available for the task first; write any constraint that remains as an explicit prohibition
What humans can see	**Goal Substitutions**	An agent substitutes its judgment for an explicit specification. A 900-word brief becomes 1,800 words because the agent judged the topic warranted more depth.	Put the output specification in the Definition of Done
What humans can see	**Transparency infrastructure failure**	A broken webhook or log means release decisions rest on incomplete data. A workflow logging outputs to a spreadsheet stops updating after a credential rotation; the Sprint Review runs on a stale record.	The transparency infrastructure gets its own Definition of Done and fails closed: a missing log entry or heartbeat counts as a failure, not a pass

© 2026 Scrum.org | All Rights Reserved.

Family	Failure mode	What goes wrong	The fix
What humans can see	**Trust accumula-tion without evidence**	Scrutiny falls as the team builds a record, then quality drifts undetected. After eight good Sprints the Product Owner stops reading drafts; in the ninth, an agent publishes work citing a retracted study.	Extend delegation only on explicit Sprint Review evidence, and hold transparency requirements when it extends
How the system learns	**Double-loop detection failure**	Individual outputs are corrected; the instruction that produced them is not. The same headline problem recurs every Sprint because nobody updates the file.	The Retrospective runs as a diagnostic event, with the Scrum Master accountable for tracing patterns to instructions

When all four families have degraded, every artifact exists, and none of them governs. Sprints run, Increments arrive, and Sprint Reviews are held, while the instruction set sits unreviewed and the boundary unenforced. An organization in that state passes every audit of whether it has a governance model and fails the only question that matters: has anything the model says ever changed what the team did? Three signals, each visible from outside the team, show whether the arrangement is holding: an Increment inspected against the Definition of Done at the close of each Sprint, a Product Backlog whose ordering changed in response to what the last Sprint revealed, and a revision history on the governing documents.

Where Human Oversight Belongs

None of the eight failure modes survived because oversight was scarce. Each survived because no human was positioned to see the degradation. Adding oversight does not help, because human inspection capacity is fixed, and AI production capacity is not, so review at every step becomes the binding constraint and removes the benefit the automation was bought for. Review at no step removes accountability, and the organization can no longer reconstruct which human judgments produced which outcomes.

Subsidiarity resolves the placement question. Decisions are made at the lowest level capable of making them reliably and escalated to a human only when they are irreversible, high-consequence, or carry accountability that cannot be distributed. Takeuchi and Nonaka (1986) found the same principle in the highest-performing product development teams and called it subtle control. In practice, the organization finds the irreversibility boundary for a specific workflow,

© 2026 Scrum.org | All Rights Reserved.

the point after which work cannot be undone, and puts the accountable person there. Delegation inside the boundary supplies the throughput, and judgment at the boundary supplies the accountability, which is why the banks that reviewed at the boundary outproduced those that reviewed everything (Buehler et al., 2024). Independent work on multi-agent coordination, with no reference to Scrum, reaches the same design: the closer a decision sits to why the work is done, the more human control it warrants, and the closer it sits to how, the more autonomy (Dochkina, 2026).

Some boundaries do not move. A Product Owner's chosen trust boundaries shift as their team demonstrates capability, while boundaries mandated by regulation, statute, or professional obligation stay put. A mandated review works when it is built into the flow as a staffed stage with an accountable person. Left downstream as a batch process, it turns into the bottleneck an agentic team exposes, because batch size drives queue length (Reinertsen, 2009). A compliance function that reviews quarterly will see throughput collapse, and that collapse is diagnostic information about the organization's structure. The response is to redesign the flow, not to patch the governance.

The whole design assumes that the person at the boundary holds the judgment the decision requires. Professional judgment is built by working on real problems alongside experienced practitioners (Lave & Wenger, 1991), and that work is often the work being automated. An organization that automates the work that produced its reviewers will, in time, staff the boundary with people who have supervised outputs without ever producing them.

Scaling Beyond One Team: The Agentic Nexus

The Raptors are one Scrum Team in which a single human governs the agents. Most Scrum Teams have several human Developers, and once each of them directs agents, the team holds several groups of agents inside a framework that allows no sub-teams (Schwaber & Sutherland, 2020). Nexus already solves a version of this problem for people, letting 3–9 Scrum Teams build one product from one Product Backlog under one Product Owner while extending Scrum only where necessary (Schwaber, 2021).

Paper 5 applies that structure inside a single Scrum Team and treats the result, the Agentic Nexus, as a hypothesis. Each human Developer's agents form an agent team working from the Scrum Team's Product Backlog, and the human members of the Scrum Team take on the accountability of the Nexus Integration Team: one Integrated Increment that meets the Definition of Done. Developers answer for the quality of their agent team's output, and the Product Owner owns value and release. The Scrum Master becomes the steward of the governing instructions across agent teams, keeping skill files consistent, coaching Developers to maintain their own, and using the Sprint Retrospective to find failure patterns that recur from one agent team to the next.

© 2026 Scrum.org | All Rights Reserved.

The Scrum and Nexus Guides set the number of humans; neither says how many agents a Developer can govern. At an illustrative 5 agents per Developer, a Scrum Team of 5 Developers covers 25 agents, and a Nexus of 9 teams with 8 Developers each covers 360. Gartner (2026) projects that an average global Fortune 500 enterprise will have more than 150,000 AI agents in use by 2028, and reports that only 13% of organizations think they have the right AI agent governance in place. The number of agents an organization can safely run is set by the number of accountable humans it can govern. Beyond that number, agents are ungoverned by definition. Organizations that need more capacity can raise the governable ratio by maturing skill files and tightening the Definition of Done, or add governed products, each with its own Product Owner and Product Backlog. The pattern has no operational evidence yet.

Next Steps

Start with a single knowledge-work workflow. Write the team canvas first, a statement of norms that sits above every instruction file and sets the priority order of the team's obligations (Hodgson, 2026). Define the irreversibility boundary for that workflow, build instruction files for each stage up to it, and run the first Sprint. The most valuable thing that Sprint produces is its constraint register, the record of what broke and why, because it tells the team which instructions, boundaries, and signals to revise next.

Before agentic AI reaches a regulated workflow, audit every mandatory checkpoint against two tests: whether it is genuinely irreversible, and whether it sits in the flow as a staffed stage. Checkpoints that fail the first test are management controls, not accountability thresholds. Those that pass the first and fail the second need redesign before the agents arrive, while the change is still cheap. Decide as well which work people keep doing, so that the people who will stand at the boundary still build the judgment it requires.

Governing AI is a Scrum capability as much as a technical one. The largest shift falls on the Scrum Master, whose role moves from facilitator to diagnostician: reading failure patterns across Increments, tracing them to the instructions that produced them, and revising those instructions rather than patching outputs. Practice needs development in writing a Definition of Done for AI-produced output, running each Scrum event when the team includes agents, and positioning Product Owner accountability when agents execute most of the work.

Where the Evidence Stops

The evidence so far shows that Scrum governed one agentic team over twelve months. It does not yet show that the arrangement holds in regulated settings, where a compliance function cannot always be embedded in the flow of work; in organizations bound by statutory process, where the accountable person may be unable to act on a triple-loop question; or across many teams, where the Agentic Nexus remains a hypothesis. The coordination studies that support the case

© 2026 Scrum.org | All Rights Reserved.

are preprints built on synthetic tasks and benchmarks. Organizations do not need to invent a framework to run them, because those that already run Scrum in their delivery teams have one.

About This Series

Each paper can be read on its own. Together, they follow a single argument: the three gaps are named, mapped to Scrum's design, tested against operational evidence, stress-tested against what can go wrong, and extended to teams in which several humans each govern agents.

Paper	Title	What it covers
Paper 1	The AI Governance Problem	Why adoption is not producing value: the assumptions conventional governance rests on, the three gaps they leave open, and why the gaps widen at Tier 3
Paper 2	Scrum Was Built for This	The theoretical case: problem classification, the empirical pillars as one chain, inspection at two levels, adaptation at three, and where the theory runs out
Paper 3	The Agentic Case Study: The Relentless Raptors	Twelve months of operation as a single-case experimental design: fast rule absorption, the Stage 2 plateau and what moved the team off it, and capacity that no longer constrains
Paper 4	Failure Modes for Agentic Systems	Eight failure modes in four families, where oversight belongs, chosen and mandated boundaries, and the judgment the boundary assumes
Paper 5	The Agentic Nexus	A hypothesis for scaling: the human Scrum Team as the Nexus Integration Team over one agent team per Developer, with agent capacity set by governance capacity

© 2026 Scrum.org | All Rights Reserved.

About the Authors

Matthew Hodgson is co-founder and CEO of Zen Ex Machina, a diagnostic-first executive strategy and advisory firm operating across government and corporate Australia and the Asia-Pacific region. He is a Scrum.org Professional Scrum Trainer and has led complex organizational transformations across Australian federal agencies and corporate environments for over 20 years. He is the author of *Evolve: The Operating Model AI Demands.*

Dave West is CEO at Scrum.org, The Home of Scrum. A frequent keynote speaker and widely published author, he co-authored The Nexus Framework For Scaling Scrum and Head First Object-Oriented Analysis and Design. He led development of the Rational Unified Process at IBM/Rational, managed Ivar Jacobson Consulting for North America, and served as VP research director at Forrester Research. Before Scrum.org, he was Chief Product Officer at Tasktop, helping grow it into a VC-backed company of nearly 100.

References

Boston Consulting Group. (2025). *The widening AI value gap: Build for the future 2025.* Boston Consulting Group. https://www.bcg.com/publications/2025/are-you-generating-value-from-ai-the-widening-gap

Buehler, K., Corsi, A., Jurisic, M., Lerner, L., Siani, A., & Weintraub, B. (2024, March 22). *Scaling gen AI in banking: Choosing the best operating model.* McKinsey & Company. https://www.mckinsey.com/industries/financial-services/our-insights/scaling-gen-ai-in-banking-choosing-the-best-operating-model

Deloitte. (2026). *State of AI in the enterprise: The untapped edge.* Deloitte AI Institute. https://www.deloitte.com/us/en/what-we-do/capabilities/applied-artificial-intelligence/content/state-of-ai-in-the-enterprise.html

Dochkina, V. (2026). *Drop the hierarchy and roles: How self-organizing LLM agents outperform designed structures* [Preprint]. arXiv:2603.28990. https://arxiv.org/abs/2603.28990

Flood, R. L., & Romm, N. R. A. (1996). *Diversity management:* Triple loop learning. Wiley.

Gartner. (2026, April 28). *Gartner identifies six steps to manage AI agent sprawl.* Gartner, Inc. https://www.gartner.com/en/newsroom/press-releases/2026-04-28-gartner-identifies-six-steps-to-manage-artificial-intelligence-agent-sprawl

Hodgson, M. (2026). *Evolve: The operating model AI demands.* Companion materials: https://zenexmachina.com/evolve-companion

© 2026 Scrum.org | All Rights Reserved.

Hodgson, M., & Horrigan, M. (2021, March 18). *How do you know if you're really agile?* [Webinar presentation]. Scrum.org.

Kratochwill, T. R., Hitchcock, J. H., Horner, R. H., Levin, J. R., Odom, S. L., Rindskopf, D. M., & Shadish, W. R. (2013). Single-case intervention research design standards. *Remedial and Special Education, 34*(1), 26–38.

Kwok, J., Li, S., Atreya, P., Liu, Y., Jiang, Y., Finn, C., Pavone, M., Stoica, I., & Mirhoseini, A. (2026). *LLM-as-a-Verifier: A general-purpose verification framework* [Preprint]. arXiv:2607.05391. https://arxiv.org/abs/2607.05391

Lave, J., & Wenger, E. (1991). Situated learning: Legitimate peripheral participation. Cambridge University Press.

Liu, N. F., Lin, K., Hewitt, J., Paranjape, A., Bevilacqua, M., Petroni, F., & Liang, P. (2024). Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics, 12*, 157–173. https://doi.org/10.1162/tacl_a_00638

Mao, Y., & Mirhoseini, A. (2026). *Decentralized multi-agent systems with shared context* [Preprint]. arXiv:2606.10662. https://arxiv.org/abs/2606.10662

McKinsey & Company. (2025, March 12). *The state of AI: How organizations are rewiring to capture value.* McKinsey & Company. https://www.mckinsey.com/capabilities/quantumblack/our-insights/the-state-of-ai

OECD. (2024). *OECD AI principles overview.* OECD.AI Policy Observatory. https://oecd.ai/en/ai-principles

Reinertsen, D. G. (2009). *The principles of product development flow: Second generation lean product development.* Celeritas Publishing.

Schwaber, K. (2021). *The Nexus guide: The definitive guide to scaling Scrum with Nexus.* Scrum.org. https://www.scrum.org/resources/nexus-guide

Schwaber, K., & Sutherland, J. (2020). *The Scrum guide: The definitive guide to Scrum — The rules of the game.* Scrum.org. https://scrumguides.org

Sclar, M., Choi, Y., Tsvetkov, Y., & Suhr, A. (2024). Quantifying language models' sensitivity to spurious features in prompt design or: How I learned to start worrying about prompt formatting. *In Proceedings of the 12th International Conference on Learning Representations.* https://arxiv.org/abs/2310.11324

© 2026 Scrum.org | All Rights Reserved.

Shavit, Y., Agarwal, S., Brundage, M., Adler, S., O'Keefe, C., Campbell, R., Lee, T., Mishkin, P., Eloundou, T., Hickey, A., Slama, K., Ahmad, L., McMillan, P., Beutel, A., Passos, A., & Robinson, D. G. (2023). *Practices for governing agentic AI systems*. OpenAI. https://openai.com/index/practices-for-governing-agentic-ai-systems/

Snowden, D. J., & Boone, M. E. (2007). *A leader's framework for decision making.* Harvard Business Review, 85(11), 68–76.

Sudhir, K. (2026, June 19). *How to design agentic systems around the implicit rules that govern your company.* Harvard Business Review. https://hbr.org/2026/06/how-to-design-agentic-systems-around-the-implicit-rules-that-govern-your-company

Takeuchi, H., & Nonaka, I. (1986). The new new product development game. *Harvard Business Review, 64*(1), 137–146.

Xiong, M., Hu, Z., Lu, X., Li, Y., Fu, J., He, J., & Hooi, B. (2024). Can LLMs express their uncertainty? An empirical evaluation of confidence elicitation in LLMs. In *Proceedings of the 12th International Conference on Learning Representations*. https://arxiv.org/abs/2306.13063

© 2026 Scrum.org | All Rights Reserved.