Journal of Business and Management Studies (JBMS)
ISSN: 2709-0876
DOI: 10.32996/jbms
Journal Homepage: www.al-kindipublisher.com/index.php/jbms

JBMS
AL-KINDI CENTER FOR RESEARCH AND DEVELOPMENT

| RESEARCH ARTICLE

Next-Generation Task Trackers as Diagnostic Operating Systems: Structuring and Analyzing Team Performance

Orlov Maxim Sergeevich
Director for Development of Project Management Methodologies LLC "Process technologies and analytics" Kazan
Corresponding Author: Orlov Maxim Sergeevich **E-mail**: maksimserggev.orlov@gmail.com

| ABSTRACT
Task trackers are usually treated as passive record-keeping tools, yet the accumulation of unresolved commitments inside them shapes team performance long before a deadline is missed. This article examines a recent proposal to reframe trackers as diagnostic operating systems that expose commitment saturation, coordination cost, and execution instability as structural conditions in how teams track and admit work. It treats them as engineering problems, not failures of individual discipline. The discussion identifies where this reframing extends existing productivity models such as SPACE and DevEx, where it converges with Kanban and flow-based management, and where the governance risks documented in algorithmic-control research reappear in a new instrumental form. The article closes by outlining the conditions under which diagnostic instrumentation improves execution coherence and the conditions under which it degrades into a surveillance layer.

| KEYWORDS
Task trackers, commitment saturation, coordination overhead, work-in-progress admission control, execution instability, algorithmic workplace monitoring, developer productivity measurement, flow management

| ARTICLE INFORMATION
ACCEPTED: 18 July 2026 **PUBLISHED:** 15 August 2026 **DOI:** 10.32996/jbms.2026.8.9.5

1. Introduction
Software teams, research groups, and operations units have accumulated decades of experience with task trackers, from paper Kanban boards to cloud-based systems that log every state change with a timestamp. Despite this maturity, complaints about overload, missed deadlines, and unpredictable delivery persist across industries that have already adopted sophisticated tooling. The recurring explanation offered inside organizations tends to be behavioral: teams lack discipline, prioritization is weak, or individuals fail to manage their time. Yurkov (2026) challenges this framing directly, arguing that persistent execution problems are frequently structural. They are produced by an asymmetry built into most tracking systems between how easily work is admitted and how much coordination its completion actually requires.
The asymmetry is not a minor design detail. Creating or activating a task carries almost no cost. Closing it depends on dependency resolution, review capacity, and sustained attention that the task's creator rarely controls. Because admission and completion face such different friction, the active set of work grows independently of a team's real capacity to finish it. The monograph describes this expansion as commitment accumulation: surface activity increases, status changes multiply, reassignments and partial updates accumulate, and underneath that motion completion rates and predictability quietly deteriorate. The tracker itself rarely registers the deterioration. Its dominant indicators (task counts, timestamps, categorical states) were designed to record motion.
This article treats that observation as the starting point for a broader question: what would it mean to design and analyze a task tracker as a diagnostic instrument. A diagnostic instrument does not simply catalog what happened. It exposes conditions that are otherwise invisible until they produce failure, in the way a dashboard warning light exposes engine strain before the engine stalls. Several literatures already converge on pieces of this idea without fully assembling it. Goal-setting research has long shown that commitment and feedback specificity shape performance (Locke & Latham, 2002). Organizational control theory

supplies a separate axis relevant here, distinguishing behavior-based from outcome-based regulation of work (Ouchi, 1979). Research on algorithmic management demonstrates that monitoring systems reshape autonomy and contest authority over how work is evaluated (Kellogg, Valentine, & Christin, 2020; Ajunwa, Crawford, & Schultz, 2017). Developer productivity researchers have proposed multidimensional frameworks, SPACE and DevEx, that explicitly reject single-metric productivity accounting (Forsgren et al., 2021; Noda et al., 2023). Flow-based management traditions built around Kanban and value-stream thinking have long argued for limiting work in progress (Anderson, 2010; Kersten, 2018). What has been comparatively underdeveloped is a systematic account of the tracker itself as an epistemic instrument, one whose representational choices determine what a team, and its management, can even perceive about its own saturation.

2. Methods

The analysis proceeds as a structured conceptual synthesis. It does not involve primary data collection or an empirical study design. Three bodies of material were compared. The first is Yurkov's (2026) monograph, which develops a structural account of execution instability built around three constructs: commitments as a scarce, capacity-sensitive resource; a diagnostic triad of persistence, reactivation, and coordination footprint used to detect saturation in execution traces; and a case study of an integrated execution control system that gates task admission against current concurrency and coordination load. The second is the established literature on work motivation and organizational control, which supplies the theoretical vocabulary against which these constructs can be positioned, specifically goal-setting theory (Locke & Latham, 2002), clan and market control mechanisms (Ouchi, 1979), and the job-demands account of burnout (Maslach, Schaufeli, & Leiter, 2001). The third is contemporary research on algorithmic workplace monitoring and developer productivity measurement, used to test whether diagnostic instrumentation of this kind reproduces known failure modes of electronic performance monitoring (Ravid et al., 2023; Siegel, König, & Lazar, 2022) or extends the multidimensional measurement logic already established in software engineering research (Forsgren et al., 2021, 2023; Greiler, Storey, & Noda, 2022).

Comparison was organized around four questions applied consistently across sources, following the structured-comparison logic that design study methodology uses to keep qualitative synthesis across heterogeneous sources auditable (Meyer, Sedlmair, & Munzner, 2018): what does each account treat as the unit of measurement, what failure mode does each account claim to detect, what governance mechanism does each account propose, and what does each account identify as a limit or risk of its own approach. These four questions allowed direct comparison between constructs that use different vocabulary but address overlapping phenomena, for instance comparing the coordination-footprint construct to Cross, Rebele, and Grant's (2016) collaborative overload and to the coordination overhead documented in the SPACE framework's collaboration dimension (Forsgren et al., 2021). Where the monograph reports specific structural findings, most notably the behavioral comparison between a conventional tracker and the integrated execution control system, those findings are treated as case-study evidence subject to the same interpretive caution applied to any single-site study: informative for mechanism, not generalizable by count.

3. Results

Conventional trackers treat the task as the primary unit of representation. A task is created, assigned, moved through states, and closed. Yurkov (2026) proposes a different unit: the commitment, defined as a persistent obligation that continues to occupy cognitive and coordination capacity independently of whether visible work is occurring on it. A task marked as blocked or paused disappears from active attention in most reporting views, yet under this account it continues to require periodic monitoring, explanation to stakeholders, and eventual reintegration, all of which consume capacity that the tracker does not record anywhere. The distinction between task and commitment has a direct analogue in goal-setting theory. Locke and Latham (2002) demonstrated that specific, difficult goals improve performance primarily through the mechanisms of direction, effort, and persistence, but their model assumes a relatively bounded goal set. The contribution here is to show what happens when the number of concurrently accepted goals grows past the point where persistence can be sustained for all of them simultaneously: attention fragments across obligations, a dynamic that goal-setting research did not need to address at the scale of dozens of parallel commitments typical of contemporary knowledge work.

The practical consequence is that commitment formation should be treated as a capacity-sensitive decision. Declaring a commitment is not enough on its own. This reframing has organizational-control implications that map onto Ouchi's (1979) distinction between behavior control, outcome control, and clan control. A conventional tracker exercises a weak form of outcome control: it records whether a task eventually closes, without regulating how many tasks may be open simultaneously. This admission-gated model shifts regulation toward the front of the process, closer to a behavior-control mechanism, by constraining which commitments may enter the active set in the first place. It does not wait to evaluate outcomes after the fact. Whether this shift improves or worsens the political dynamics that Ouchi (1979) associated with each control type is an empirical question the monograph does not fully resolve, since gating decisions still require someone, or some algorithm, to adjudicate contested admission requests.

The most operationally concrete contribution in Yurkov (2026) is a diagnostic triad of three signals. Persistence tracks how long obligations remain unresolved. Reactivation counts how often a nominally closed or paused item must be reopened, and coordination footprint captures the compensatory clarification, handoff, and alignment work required to hold a fragmenting execution state together. The triad argues that when these three signals rise together, a team is likely saturated even while headline throughput indicators, the metrics most dashboards surface by default, remain stable. That claim is consistent with queueing theory as formalized by Little's Law, which establishes that average time in system rises with average work in progress when arrival and service rates hold steady (Little, 1961). The framework applies this relationship directly to tracker data, arguing that a high proportion of partially started commitments in the active set signals rising cycle time even before that rise becomes visible in completion statistics.

This convergence between the diagnostic triad and flow-based management is not coincidental. Anderson's (2010) Kanban method and Kersten's (2018) Flow Framework both argue that limiting work in progress protects cycle-time predictability, and Vacanti (2015) develops cycle-time analytics explicitly for this purpose. What Yurkov (2026) adds to this tradition is a second layer that flow-based approaches tend to underspecify: the coordination footprint generated when concurrency limits are absent. Classical Kanban treats WIP primarily as a throughput variable. Yurkov's framework isolates rising coordination cost as a separate and earlier warning sign, visible in meeting density, clarification volume, and synchronous interaction well before cycle time itself starts to deteriorate. That pattern aligns closely with Cross, Taylor, and Zehner's (2018) account of collaboration overload, and with the SPACE framework's explicit inclusion of collaboration quality as a dimension separate from raw efficiency (Forsgren et al., 2021), suggesting that the coordination footprint is best read as a tracker-native operationalization of a construct software-engineering researchers have already validated through survey instruments.

Yurkov (2026) presents a case study of an integrated execution control system that evaluates proposed commitments against current concurrency and coordination load before admitting them. It does not allow unconstrained creation followed by after-the-fact reporting. The comparison reported there between this system and a conventional tracker spans four dimensions: commitment admission, visibility of latent load, flow stability, and coordination dynamics. In the conventional regime, new work enters through creation and assignment with weak representation of cumulative load, so coordination scales reactively through escalation and meetings once the load becomes unmanageable. Gating changes the entry point itself: admission turns into an explicit negotiation tied to capacity signals, and a commitment that goes dormant stays visible as a first-class entity instead of disappearing from view.

The behavioral shift Yurkov (2026) documents in this case is not primarily technical. Teams operating under the gated system began including capacity discussion in requests for new work, and deferral became an explicit, legitimate outcome instead of something achieved through informal overwork. The shift matches Forsgren, Humble, and Kim's (2018) finding in Accelerate that high-performing technology organizations sustain throughput partly through explicit work-in-progress limits and batch-size discipline, not individual heroics, and it echoes Bellotti et al.'s (2004) earlier observation that personal task-list managers fail when they cannot represent the negotiation of scope and timing that surrounds real task acceptanc. The monograph also reports resistance during early adoption, particularly where responsiveness had previously been treated as a proxy for competence, a pattern directly consistent with the adoption frictions documented in algorithmic-management research when systems begin to constrain discretion that workers previously exercised informally (Kellogg et al., 2020).

A further structural contribution in Yurkov (2026) is a table distinguishing four decision domains, admission of new commitments, deferral and persistence management, overload detection, and coordination restructuring, and specifying for each domain who holds primary authority, how the system participates, and what governance risk requires monitoring. For admission, primary authority is assigned to managers and teams while the system surfaces saturation signals that require explicit justification to override; the associated risk is override normalization, meaning saturation evidence gets erased as exceptions become routine. For overload detection, authority shifts toward the system itself, running longitudinal variability and persistence analysis, with the corresponding risk being false certainty produced by an opaque scoring mechanism that misclassifies exploratory work as overload. Deferral and persistence management sits closer to the middle of this authority spectrum: the individual holding a commitment and the manager reviewing team capacity share the decision, while the system contributes only a threshold flag once an item's persistence crosses a set duration. The risk in this domain is that deferral quietly becomes a default. A commitment can sit unresolved for months without ever being reopened for explicit reprioritization. Coordination restructuring works on a different logic again. The system holds no decision authority here at all, it surfaces rising meeting density and clarification volume as evidence, while the choice to change reporting lines, handoff points, or team boundaries stays entirely with team leads. The corresponding risk is that a restructuring signal generated from aggregate coordination data gets treated by managers as a mandate, which narrows a team's own judgment about how it organizes its work.

This distribution of authority is where the framework intersects directly with the algorithmic-control literature, and where the non-promotional caution required of this article matters most. Ajunwa, Crawford, and Schultz (2017) document how workplace monitoring systems, even when adopted for legitimate productivity purposes, tend to expand surveillance scope over time and create asymmetries of information between monitored workers and the managers who interpret monitoring output. Ravid et al.'s (2023) meta-analysis of electronic performance monitoring found that monitoring workers perceive as evaluative correlates with reduced job satisfaction and increased counterproductive work behavior. Monitoring framed as developmental does not carry the same association, a finding echoed by Siegel, König, and Lazar (2022). The monograph is explicit that the risk of coordination signals being used punitively, driving compensatory work into untracked and therefore invisible channels, is a real governance

failure mode, not a hypothetical one, and the case study reports exactly this workaround behavior when constraint signals felt distant from workers' lived experience of interruption and reactivation.
Yurkov (2026) argues that the underlying shift required is epistemic before it is technical: task trackers must be understood as instruments that determine what can be known about execution, not as passive containers that record it after the fact. This framing resonates with Maslach, Schaufeli, and Leiter's (2001) burnout model insofar as both treat the accumulation of unresolved demands as a structural condition, not an individual deficiency, but it extends the burnout literature's typically individual unit of analysis to the level of the tracking system itself. Burnout research asks what sustained demand does to a person. This diagnostic framing shifts the object of inquiry: it asks what sustained, unregulated commitment admission does to a team's execution trace, treating the trace itself as diagnosable for saturation independently of self-reported exhaustion. The two levels are compatible, and the DevEx literature's explicit inclusion of developer satisfaction alongside flow-efficiency metrics suggests a plausible bridge between them (Noda et al., 2023).

4. Conclusion

Reframing task trackers as diagnostic operating systems, in Yurkov's (2026) terms, shifts the analytical question away from whether individual workers manage their time well and toward whether the system regulates how many unresolved commitments a team is permitted to carry at once. The diagnostic triad of persistence, reactivation, and coordination footprint gives this shift an operational vocabulary grounded in execution traces, and the admission-gating case study demonstrates that constraining commitment entry, not merely recording completion, changes coordination behavior in ways consistent with established flow-management and productivity-measurement research. At the same time, the governance risks this instrumentation introduces are structurally identical to those already documented for algorithmic workplace monitoring. Calling a system diagnostic instead of evaluative does not remove those risks in practice. The most defensible reading of this contribution is therefore conditional: the framework offers a coherent structural account of why execution degrades under commitment saturation and a concrete mechanism for intervening earlier, but its value depends entirely on governance conditions, explicit override accounting, interpretable signals, and non-punitive deferral, that the tracking software itself cannot enforce.

Funding: This research received no external funding.
Conflicts of Interest: The authors declare no conflict of interest.
Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers.

References

[1] Ajunwa, I., Crawford, K., & Schultz, J. (2017). Limitless worker surveillance. California Law Review, 105(3), 735-776. https://doi.org/10.15779/Z38BR8MF94
[2] Anderson, D. J. (2010). Kanban: Successful evolutionary change for your technology business. Blue Hole Press.
[3] Bellotti, V., Dalal, B., Good, N., Flynn, P., Bobrow, D. G., & Ducheneaut, N. (2004). What a to-do: Studies of task management towards the design of a personal task list manager. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (pp. 735-742). ACM. https://doi.org/10.1145/985692.985785
[4] Cross, R., Rebele, R., & Grant, A. (2016). Collaborative overload. Harvard Business Review, 94(1), 74-79.
[5] Cross, R., Taylor, S., & Zehner, D. (2018). Collaboration overload revisited: Managing excessive demands on the time and attention of knowledge workers. California Management Review, 60(4), 40-58. https://doi.org/10.1177/0008125618775461
[6] 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.
[7] Forsgren, N., Storey, M.-A., Maddila, C., Zimmermann, T., Houck, B., & Butler, J. (2021). The SPACE of developer productivity: There's more to it than you think. ACM Queue, 19(1), 20-48. https://doi.org/10.1145/3454122.3454124
[8] Forsgren, N., Kalliamvakou, E., Noda, A., Greiler, M., Houck, B., & Storey, M.-A. (2023). DevEx in action: A study of its tangible impacts. ACM Queue, 21(6), 47-77. https://doi.org/10.1145/3639443
[9] Greiler, M., Storey, M.-A., & Noda, A. (2022). An actionable framework for understanding and improving developer experience. IEEE Transactions on Software Engineering, 49(4), 1411-1425.
[10] Kellogg, K. C., Valentine, M. A., & Christin, A. (2020). Algorithms at work: The new contested terrain of control. Academy of Management Annals, 14(1), 366-410. https://doi.org/10.5465/annals.2018.0174
[11] Kersten, M. (2018). Project to product: How to survive and thrive in the age of digital disruption with the Flow Framework. IT Revolution Press.
[12] 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
[13] Locke, E. A., & Latham, G. P. (2002). Building a practically useful theory of goal setting and task motivation: A 35-year odyssey. American Psychologist, 57(9), 705-717. https://doi.org/10.1037/0003-066X.57.9.705

[14] Maslach, C., Schaufeli, W. B., & Leiter, M. P. (2001). Job burnout. Annual Review of Psychology, 52, 397-422. https://doi.org/10.1146/annurev.psych.52.1.397
[15] Meyer, M., Sedlmair, M., & Munzner, T. (2018). Design study methodology: Reflections from the trenches and the stacks. IEEE Computer Graphics and Applications, 38(3), 62-74. https://doi.org/10.1109/MCG.2018.032421095
[16] Noda, A., Storey, M.-A., Forsgren, N., & Greiler, M. (2023). DevEx: What actually drives productivity: The developer-centric approach to measuring and improving productivity. ACM Queue, 21(2), 35-53. https://doi.org/10.1145/3595878
[17] Ouchi, W. G. (1979). A conceptual framework for the design of organizational control mechanisms. Management Science, 25(9), 833-848. https://doi.org/10.1287/mnsc.25.9.833
[18] Ravid, D. M., White, J. C., Tomczak, D. L., Miles, A. F., & Behrend, T. S. (2023). A meta-analysis of the effects of electronic performance monitoring on work outcomes. Personnel Psychology, 76(1), 5-40. https://doi.org/10.1111/peps.12514
[19] Siegel, R., Konig, C. J., & Lazar, V. (2022). The impact of electronic monitoring on employees' job satisfaction, stress, performance, and counterproductive work behavior: A meta-analysis. Computers in Human Behavior Reports, 8, Article 100227. https://doi.org/10.1016/j.chbr.2022.100227
[20] Vacanti, D. S. (2015). Actionable agile metrics for predictability: An introduction. Daniel S. Vacanti, Inc.
[21] Yurkov, Y. (2026). Next-generation task trackers: Structuring and analyzing the work of teams. LAP LAMBERT Academic Publishing.