From User-Centred Evaluation to Signal-Based Governance

Reshaping the Future of Software Engineering Project Management

INTRODUCTION

Traditional Software Engineering Project Management (SEPM) has long relied on user-centred evaluation (UCE) methodologies—surveys, interviews, focus groups, and perceptual feedback—to assess software quality, user satisfaction, and project success. However, contemporary software architectures, cloud-native deployment patterns, and the emergence of automated telemetry systems are increasingly reshaping how organisations evaluate software quality and manage development lifecycles.

This essay proposes that Epistemic Signal Governance (ESG) represents an emerging analytical construct—synthesising existing signal-based practices observed across contemporary software engineering processes—that will become increasingly dominant in shaping SEPM's future direction. ESG is defined here as the systematic curation and governance of evidence-based behavioural signals derived from infrastructure observability over subjective user interpretation, enabling project managers to ground decisions in real-time, longitudinal system data rather than self-reported perceptions.

This shift is not merely technical; it reflects a structural reorganisation of evaluative authority—increasingly shifting from individual user perception towards infrastructure-derived evidence. By integrating Forsgren et al.'s SPACE framework, the emerging process trends documented by Khan and Le, and the distributed computing paradigms identified by Fahmideh and Laroui, this essay demonstrates that signal-based governance will substantially transform SEPM through:

- (1) real-time quality assessment mechanisms
- (2) decentralised governance structures
- (3) data-informed resource allocation
- (4) hybrid evaluation frameworks balancing quantitative signals with qualitative insights.

CONTEXT: THE LANDSCAPE OF FUTURE TRENDS IN SEPM

Contemporary software engineering faces multiple transformative trends. Khan and Le (2022) identify evolving process patterns spanning Agile maturation, DevOps/DevSecOps consolidation, low-code/no-code platforms, AI-driven project analytics, and distributed ledger technologies. Concurrently, Fahmideh et al. (2023) and Laroui et al. (2021) document the emergence of blockchain-based smart contracts for governance and edge/fog computing architectures that increasingly push decision-making logic to distributed nodes.

Each trend addresses specific SEPM challenges. Low-code platforms reduce time-to-market but introduce governance complexity. AI-driven analytics promise predictive project management but risk algorithmic bias amplification. Blockchain enables transparent, immutable audit trails but requires sophisticated cryptographic infrastructure. Edge/fog computing enables real-time processing but fragments observability across distributed systems.

However, few of these trends directly address a foundational epistemological challenge facing contemporary SEPM: the mismatch between traditional evaluation methodologies and how modern systems actually operate. User-centred evaluation—exemplified by Law et al.’s (2014) survey methodology—treats user perception as ground truth. Yet distributed cloud systems, continuous deployment pipelines, and real-time user interactions increasingly generate observational data at scales and granularities that periodic surveys cannot feasibly capture.

Signal-based governance emerges as a response at this intersection. Rather than positioning ESG as a competing trend, it functions as a meta-level analytical framework that can integrate signals from AI analytics, blockchain audit trails, and edge-computed metrics into coherent quality assessments. ESG thereby provides an epistemological foundation that other technological trends depend upon for coherent project governance.

Forsgren et al.’s; SPACE framework, is instructive here. SPACE measures developer productivity across five dimensions: Satisfaction and well-being, Performance, Activity, Communication and collaboration, Efficiency and flow. Critically, SPACE data derives entirely from system-observable signals—repository commits, code review cycles, deployment frequencies, incident response times—rather than developer self-assessment. This shift from perception to observation exemplifies the analytical approach underlying the UCE-to-signal-governance transition in contemporary practice.

THE CURRENT LIMITATION: USER-CENTRED EVALUATION IN CONTEMPORARY CONTEXTS

User-centred evaluation, though methodologically sound, increasingly faces three structural limitations when applied to contemporary SEPM contexts:

Scalability challenges. Traditional UX research methods—ethnographic observation, structured interviews, A/B testing cohorts—do not scale efficiently across globally distributed teams, heterogeneous user populations, and continuous deployment cycles. Law et al. (2014) document that UCE surveys typically require 4–12 weeks to design, deploy, and analyse, making them increasingly unsuitable for sprint-based decision-making. When organisations such as Netflix shift recommendation algorithms every 48 hours, or Spotify deploys new features to autonomous squads within hours, survey-based evaluation becomes organisationally difficult to implement at required decision frequencies.

Self-report bias. Survey responses suffer from well-documented cognitive biases: social desirability bias (users report preferences aligned with perceived norms), hindsight bias (post-hoc rationalisation of behaviour), and the Hawthorne effect (behaviour modification under observation). Gu, Tang and Xue (2023) demonstrate that subjective UI evaluations deteriorate significantly over time due to halo effects, whereas temporal behavioural signals—session duration, task completion rates, error recovery patterns—prove relatively consistent predictors of actual user engagement across extended periods.

Cost and accessibility barriers. Comprehensive UCE programmes require dedicated user research teams, participant incentives, and recruiting infrastructure. For organisations developing embedded systems, IoT platforms, or business-critical software serving thousands of internal stakeholders, systematic UCE becomes prohibitively expensive to implement comprehensively. Even large enterprises struggle to conduct timely, representative user research across their full product ecosystem.

These limitations do not invalidate UCE methodology; rather, they increasingly expose its constraints when applied to modern software operating conditions—cloud-native architectures, DevOps pipelines progressively minimising manual quality gates, and real-time market feedback channels. Van der Aalst et al. (2018) document how Robotic Process Automation (RPA) and Infrastructure-as-Code tools increasingly replace manual quality gates with automated signal extraction and threshold-based escalation.

The practical consequence is that organisations increasingly default to observable signals as the primary quality substrate, employing UCE as supplementary validation for strategic decisions. This represents not the evolution of UCE, but a significant structural reorientation in evaluative authority.

THE SIGNAL SHIFT: FROM PERCEPTION TOWARDS OBSERVATION

Signal-based governance emerges from recognition that modern software systems increasingly generate rich, continuous, multi-dimensional evaluative signals.
Telemetry as evaluative substrate. Cloud-native architectures instrument virtually every component: API response latencies, database query patterns, exception stack traces, user interaction sequences, and resource utilisation metrics. When Netflix processes billions of telemetry events daily, each event contains signal data—did the user watch the recommended content, how long until disengagement, did playback buffer—that traditional surveys cannot feasibly capture.
Forsgren et al. (2021) formalise this observation through SPACE, demonstrating that developer productivity—traditionally a management blind spot—becomes increasingly observable through infrastructure signals:

- **Satisfaction and well-being**: calendar fragmentation (meeting density), communication tone analysis, incident-driven context switches
- **Performance**: code review throughput, deployment frequency, mean time to recovery (MTTR)
- **Activity**: repository contribution volume, test coverage expansion, documentation updates
- **Communication and collaboration**: pull request discussion patterns, cross-team dependency resolution time
- **Efficiency and flow**: time between code commit and production deployment, batch size of changes

Critically, SPACE metrics derive entirely from infrastructure observability—git logs, CI/CD systems, incident trackers—not manager surveys or employee self-assessment. This shift represents a fundamental reorientation in evaluative authority: from subjects' interpretations towards systems' observable states.

Signal integration across infrastructure layers. Signal-based frameworks increasingly synthesise data across multiple observational dimensions:

- **User-level signals**: session duration, feature adoption, churn probability
- **Developer-level signals**: commit frequency, code review duration, on-call incident load
- **System-level signals**: API latency percentiles, database lock contention, deployment success rates
- **Organisational signals**: team velocity convergence, knowledge distribution, cross-team dependency patterns

These signals are progressively integrated into governance structures that previously relied primarily on managerial judgement. Deployment decisions, for example, increasingly shift from “does the code review panel approve release?” towards “have pre-release signal thresholds been satisfied?” (e.g., error rates < 0.01%, latency p95 < 200ms, zero critical security findings).

Mäkitalo et al. (2020) on opportunistic reuse illustrates practical implications. They document how signal-driven assessment enables organisations to identify reusable components through usage pattern analysis (which components are imported across teams, which versions persist longest) rather than relying solely on architectural mandates. Signal-based identification of reusable code proves more effective than expert design review alone—organisations discover that components developers actually use frequently differ substantially from those architects designated as reusable.

Managing social signal amplification. Concurrently, public software evaluation increasingly occurs through social media, open-source stars, and influencer endorsement—channels generating high-amplitude but often low-credibility signals. Organisations risk amplifying social noise rather than technical evidence, potentially leading to adoption of frameworks based on popularity rather than organisational suitability. ESG mitigates this risk by explicitly weighting signal credibility: social

signals are treated as “supplementary artefacts” (Van der Linden et al., 2019), whilst infrastructure signals receive epistemological priority in governance decisions.
This analytical approach does not eliminate human judgement; rather, it restructures the information environment in which judgement operates. Project managers increasingly shift from asking “what do users report in surveys?” towards “what do systems reveal through continuous observation?” The latter question admits more direct empirical resolution and enables decision-making frequencies aligned with modern deployment cycles.

SEPM IMPLICATIONS: GOVERNANCE TRANSFORMATION

Signal-based frameworks increasingly transform four core SEPM functions:

Quality assessment and release gating

Traditional SEPM has relied on quality assurance teams conducting manual test cycles before release approval. Signal-based approaches increasingly invert this model: automated signal thresholds progressively become the primary quality assessment mechanism.
Example (Spotify squad model): rather than relying on manual acceptance testing, autonomous squads deploy feature flags to production with monitoring thresholds: if user engagement drops below an established baseline, the flag automatically reverts; if error rate exceeds a defined threshold, incident escalation triggers immediately. This signal-driven release gating increasingly compresses cycle time from weeks to hours whilst maintaining quality assurance—quality is assessed against real user behaviour, not test case adequacy alone.
Khan and Le (2022, Chapter 14) document this pattern across the industry: evolving processes increasingly delegate quality decisions to automated signal interpretation rather than purely manual review. The SEPM implication is substantial: quality assurance progressively becomes a signal engineering function—designing which

metrics to monitor, establishing appropriate thresholds, and integrating signals across system layers—rather than manual test execution.

Risk management and incident response

Signal-based SEPM increasingly restructures risk assessment from purely predictive modelling (forecasting failures) towards real-time detection (identifying failures as they emerge).

Laroui et al. (2021) demonstrate this pattern in edge/fog computing contexts: when processing logic distributes across thousands of edge nodes, traditional centralised monitoring becomes operationally difficult. Signal-based governance instead establishes signal propagation patterns—edge nodes emit signals (resource utilisation, packet loss, temperature) that aggregation services integrate into cluster health assessment. Risk is managed through signal-based cascade prevention: if latency signals indicate congestion, traffic sheds to alternative edge zones before cascade failure occurs.

For SEPM, this means projects increasingly transition from “risk register maintained by project manager” towards “signal thresholds triggering automated escalation”. This changes stakeholder communication: rather than presenting risk assessments primarily through subjective narrative, PMs increasingly present objective signal data alongside the governance decisions that thresholds have triggered.

Resource allocation and team optimisation

Forsgren et al. (2021) demonstrate that SPACE signals enable increasingly sophisticated resource allocation: teams exhibiting high context-switching signals (many meetings, fragmented focus) receive protected focus time; teams experiencing high incident response load receive on-call load balancing; teams exhibiting slow code review cycles receive targeted tooling investment.

Signal-based SEPM progressively shifts from “allocate resources based on manager intuition or organisational politics” towards “allocate resources to optimise signal convergence towards target organisational states.” This requires explicitly defining target signal profiles for different team types (platform teams, product teams, and infrastructure teams exhibit different optimal SPACE profiles), then using signal data to identify bottlenecks. Khan and Le (2022, Chapter 9) document this maturation: agile measurement moves beyond story-point velocity towards throughput and flow metrics (e.g., lead time, deployment frequency) that better reflect delivery capacity.

Implications for the SEPM practitioner role

This increasingly implies a fundamental reorientation of SEPM competencies. Traditional PM skills—schedule management, resource negotiation, and risk estimation based on expert judgement—remain relevant but become more dependent on emerging capabilities:

- **Signal literacy**: understanding how systems generate signals, what signals mean, and when thresholds require recalibration
- **Governance design**: defining which signals matter, setting thresholds, and establishing escalation criteria
- **Ethical oversight**: identifying metric gaming, bias amplification, and loss of minority voices as governance risks
- **Human orchestration**: translating multi-dimensional signals into coherent narratives for non-technical stakeholders

This is not the elimination of PM judgement, but its repositioning towards orchestration and ethical stewardship of signal governance frameworks.

COMPARATIVE ANALYSIS: SIGNAL GOVERNANCE WITHIN THE TREND ECOSYSTEM

How does signal-based governance position relative to competing future trends?

Versus AI/ML-driven PM: AI-driven project management promises predictive analytics—forecasting schedule risk, cost overrun probability, and burnout. However, predictive ML models require labelled training data and may struggle with distributional shift (what worked historically may not apply in novel contexts). Signal-based governance can be more robust in practice because it does not require historical models, only real-time observability. When circumstances change dramatically (e.g., unexpected remote-work transition), signal-based decisions adapt immediately, whilst ML models may lag. ESG and predictive ML are complementary—ML can be trained on historical signal data—but signal governance provides the epistemological foundation.

Versus blockchain for governance: blockchain-based smart contracts (Fahmideh et al., 2023) provide cryptographically verifiable, immutable audit trails (e.g., if code does not pass a security scan, contract execution halts). However, blockchain is primarily a trust and verification architecture, whereas ESG is an evaluation framework (determining what quality means). Organisations can combine both: blockchain records signals in immutable ledgers, whilst signal governance interprets those signals. For intra-organisational SEPM, signal governance provides value without blockchain’s computational overhead.

Versus edge/fog computing: edge and fog architectures (Laroui et al., 2021) distribute computation to peripheral nodes, reducing latency and central bottlenecks. However, distributed computing creates signal fragmentation—observability becomes operationally challenging when processing scatters across thousands of nodes. Signal governance frameworks address this by formalising signal aggregation patterns: edge nodes emit standardised signal formats that central coordination services integrate. Signal governance thereby makes edge/fog computing operationally manageable.

In synthesis, signal-based governance functions not as a standalone trend but as a meta-governance framework that other technologies increasingly depend upon for coherent project management.

CRITICAL RISKS AND MITIGATION STRATEGIES

Signal-based governance, whilst potentially beneficial, introduces risks that require disciplined mitigation:

Metric gaming and Goodhart's Law. When signal thresholds become explicit governance criteria, teams may optimise for measured metrics rather than underlying objectives. For example, if deployment frequency becomes an explicit SPACE metric, teams may deploy trivial changes to inflate frequency, degrading overall quality. Mitigation: multi-dimensional signal frameworks where no single signal determines decisions; explicit trade-off documentation; periodic audits for gaming behaviours.

Loss of minority user voices. Signal-based frameworks privilege majority signals. If most users do not use accessibility features, accessibility signals remain low-priority in resource allocation, risking systematic neglect. Mitigation: stratified signal analysis (disaggregate by user segments), dedicated monitoring for under-represented populations, and periodic UCE-based qualitative research for marginalised groups.

Infrastructure dependency. Signal governance depends on functioning telemetry systems. Observability failures create blind spots and force reactive management. Mitigation: redundant observation infrastructure, fallback governance criteria, and periodic validation of signal accuracy and pipeline reliability.

Algorithmic bias amplification. If signal interpretation is automated, biases in data collection (e.g., capturing desktop usage but not mobile) can become systemic biases affecting portfolios. Mitigation: regular bias audits, diverse signal sources, and human-in-the-loop decision-making for high-stakes governance choices.

These risks are manageable but require explicit governance design. Signal governance is not a panacea; it is a framework that demands disciplined implementation and continuous review.

IMPLEMENTATION ROADMAP FOR SEPM TRANSITION

Organisations transitioning towards signal-based SEPM governance should consider a phased approach:

Phase 1 (Months 1–3): Signal discovery and instrumentation

- Audit current systems for available signals (git, CI/CD, monitoring infrastructure)
- Identify observability gaps (which signal dimensions lack instrumentation?)
- Prioritise instrumentation using SPACE as a structuring guide

Phase 2 (Months 4–6): Governance framework definition

- Define target signal profiles for different team types
- Establish explicit thresholds for release gates, resource allocation, and escalation
- Document trade-offs and governance principles (e.g., why deployment frequency is weighted at 20% and quality at 30%)

Phase 3 (Months 7–9): Automation and integration

- Integrate thresholds into CI/CD pipelines
- Automate escalation when thresholds breach
- Establish feedback mechanisms to test whether automated decisions produce intended outcomes

Phase 4 (Months 10–12): Refinement and hybrid integration

- Analyse signal drift: do thresholds remain valid as context evolves?
- Integrate qualitative UCE feedback for validation and bias detection
- Document lessons learned and refine governance iteratively

This timeline assumes mid-sized organisations (50–500 engineers). Large enterprises may require extended Phase 1 due to legacy heterogeneity; start-ups can often compress phases through rapid iteration.

CONCLUSION

The transition from user-centred evaluation towards signal-based governance represents a significant structural transformation in SEPM's near future. This is not primarily a technology trend (like blockchain or edge computing) but a fundamental reorientation of evaluative authority—shifting from individual perception towards collective observation, from interpretive judgement towards data-informed synthesis, and from periodic assessment towards continuous monitoring.

Forsgren et al.'s SPACE framework demonstrates that this transition is technically feasible; Khan and Le's process evolution taxonomy locates it within broader industry transformation; and Fahmideh and Laroui's work on distributed systems shows that signal governance is increasingly necessary for operationalising modern architectures.

Yet signal-based governance is not a panacea. Metric gaming, loss of minority voices, infrastructure dependency, and algorithmic bias remain serious challenges requiring disciplined governance design. Organisations that successfully transition will be those that treat signal governance not as an operational convenience but as a strategic capability.

For project managers, the implications are clear: the future increasingly belongs to those who can design coherent signal governance frameworks, interpret multi-dimensional metrics under uncertainty, and balance quantitative evidence with qualitative judgement. Signal governance does not eliminate human judgement; it repositions it as orchestration—professionals focus on governance design, signal interpretation, ethical oversight, and stakeholder communication, whilst infrastructure handles signal collection and algorithmic decision support.

References

Fahmideh, M., Heidari, F. and Mougouei, D. (2023) 'Engineering Blockchain-Based Software Systems: Foundations, Survey, and Future Directions', ACM Computing Surveys, 55(6), pp. 1–44. Available at: https://doi.org/10.1145/3530813

Forsgren, N., Storey, M.-A., Maddila, C., Zimmermann, T., Houck, B. and Butler, J. (2021) 'The SPACE of Developer Productivity', Communications of the ACM, 64(6), pp. 46–53. Available at: https://doi.org/10.1145/3453928

Gu, Q., Tang, W. and Xue, C. (2023) 'The Effect of Time Lapse on the Halo Effect in the Subjective Evaluation of Digital Interfaces', in Design, User Experience, and Usability (Lecture Notes in Computer Science, vol. 14032). Cham: Springer. Available at: https://doi.org/10.1007/978-3-031-35702-2_12

Khan, A.A. and Le, D-N. (2022) Evolving Software Processes: Trends and Future Directions. Hoboken, New Jersey: John Wiley & Sons. Available at: https://doi.org/10.1002/9781119821779

Laroui, M., Sarhan, S., Zeghlache, D., Al Rammahi, S.K. and Matta, I. (2021) 'Edge and Fog Computing for IoT: A Survey on Current Research Activities & Future Directions', Computer Communications, 180, pp. 210–231. Available at: https://doi.org/10.1016/j.comcom.2021.09.003

Law, E.L.-C. et al. (2014) 'Understanding, scoping and defining user experience: a survey approach', Human–Computer Interaction, 29(3), pp. 263–287. Available at: https://dl.acm.org/doi/10.1145/1518701.1518813

Mäkitalo, N., Taivalsaari, A., Kiviluoto, A., Mikkonen, T. and Capilla, R. (2020) 'On Opportunistic Software Reuse', Computing, 102, pp. 2385–2408. Available at: https://doi.org/10.1007/s00607-020-00833-6

Van der Aalst, W.M.P., Bichler, M. and Heinzl, A. (2018) 'Robotic Process Automation', Business & Information Systems Engineering, 60, pp. 269–272. Available at: https://doi.org/10.1007/s12599-018-0542-4

Van der Linden, J., Amadieu, F., Vayre, E. and Van De Leemput, C. (2019) 'User Experience and Social Influence: A New Perspective for UX Theory', in Design, User Experience, and Usability (Lecture Notes in Computer Science, vol. 11583). Cham: Springer. Available at: https://doi.org/10.1007/978-3-030-23570-3_9